2021-1568 (LEAD), -1569, -1570, -1571, -1573

United States Court of Appeals for the Federal Circuit

UNILOC USA, INC., UNILOC LUXEMBOURG, S.A.,
Plaintiffs-Appellants

UNILOC 2017 LLC,
Plaintiff

v.

APPLE INC.,
Defendant-Appellee

ELECTRONIC FRONTIER FOUNDATION,
Intervenor-Appellee

Appeal from the United States District Court for the Northern District of California
Case Nos. 3:18-cv-00358-WHA, 3:18-cv-00360-WHA,
3:18-cv-363-WHA, 3:18-cv-00365-WHA, 3:18-cv-00572-WHA
before Judge William H. Alsup

FIRST CORRECTED BRIEF OF PLAINTIFFS-APPELLANTS

Aaron S. Jacobs
Prince Lobel Tye LLP
One International Place, Suite 3700
Boston, MA 02110
(617) 456-8000
ajacobs@princelobel.com

April 2, 2021

Counsel for Plaintiffs-Appellants

CERTIFICATE OF INTEREST

UNITED STATES COURT OF APPEALS
FOR THE FEDERAL CIRCUIT

AMENDED

CERTIFICATE OF INTEREST

Case Number	21-1568, -1569, -1570, -1571, -1573
Short Case Caption	Uniloc USA, Inc. v. Apple Inc.
Filing Party/Entity	Uniloc USA, Inc.; Uniloc Luxembourg S.A.

Instructions: Complete each section of the form. In answering items 2 and 3, be specific as to which represented entities the answers apply; lack of specificity may result in non-compliance. **Please enter only one item per box; attach additional pages as needed and check the relevant box**. Counsel must immediately file an amended Certificate of Interest if information changes. Fed. Cir. R. 47.4(b).

I certify the following information and any attached sheets are accurate and complete to the best of my knowledge.

Date: April 2, 2021

Signature: */s/ Aaron S. Jacobs*

Name: Aaron S. Jacobs

1. Represented Entities. Fed. Cir. R. 47.4(a)(1).	**2. Real Party in Interest.** Fed. Cir. R. 47.4(a)(2).	**3. Parent Corporations and Stockholders.** Fed. Cir. R. 47.4(a)(3).
Provide the full names of all entities represented by undersigned counsel in this case.	Provide the full names of all real parties in interest for the entities. Do not list the real parties if they are the same as the entities. ☐ None/Not Applicable	Provide the full names of all parent corporations for the entities and all publicly held companies that own 10% or more stock in the entities. ☐ None/Not Applicable
Uniloc USA, Inc.	Uniloc 2017 LLC	Uniloc Corporation Pty. Ltd.
Uniloc Luxembourg S.A.	Uniloc 2017 LLC	None

☐ Additional pages attached

4. Legal Representatives. List all law firms, partners, and associates that (a) appeared for the entities in the originating court or agency or (b) are expected to appear in this court for the entities. Do not include those who have already entered an appearance in this court. Fed. Cir. R. 47.4(a)(4).

☐ None/Not Applicable ☑ Additional pages attached

Nelson Bumgardner Albritton, P.C.	Edward R. Nelson, III	Anthony Michael Vecchione
Prince Lobel Tye LLP	Kevin Gannon	Tyrus C. Cartwright
Daniel McGonagle	Dean G. Bostock	Matthew David Vella

5. Related Cases. Provide the case titles and numbers of any case known to be pending in this court or any other court or agency that will directly affect or be directly affected by this court's decision in the pending appeal. Do not include the originating case number(s) for this case. Fed. Cir. R. 47.4(a)(5). See also Fed. Cir. R. 47.5(b).

☐ None/Not Applicable ☑ Additional pages attached

6. Organizational Victims and Bankruptcy Cases. Provide any information required under Fed. R. App. P. 26.1(b) (organizational victims in criminal cases) and 26.1(c) (bankruptcy case debtors and trustees). Fed. Cir. R. 47.4(a)(6).

☑ None/Not Applicable ☐ Additional pages attached

ADDENDUM

4. Legal Representatives (continued)

Prince Lobel Tye LLP (continued):	Michael James Ercolini	Robert R. Gilman
Paul J. Hayes	Brian A. Tollefson	Tyrus S. Cartwright

5. Related Cases

Pursuant to Uniloc's unopposed Motion, *see* Appeal No. 21-1568, Dkt. No. 14 (Motion), the present appeals were deconsolidated from *Uniloc USA, Inc. v. Apple Inc.*, Appeal No. 21-1572, on February 25, 2021. *See* Appeal No. 21-1568, Dkt. No. 15 (Order). Although now-deconsolidated Appeal No. 21-1572 arises from the same underlying case as Appeal No. 21-1573, as described in Uniloc's Motion, the issues, orders on appeal and interested parties in the present appeals are different from Appeal No. 21-1572.

The following appeals, while perhaps not "related cases" within the meaning of this Court's rules, were designated as companion cases to the deconsolidated Appeal No. 21-1572:

Uniloc USA, Inc. v. Motorola Mobility LLC, No. 21-1555 (Fed. Cir.)	*Uniloc 2017 LLC v. Google LLC*, Nos. 21-4198, -1500, -1501, -1502, -1503, -1504, -1505, -1506, -1507, -1508, -1509 (Fed. Cir.) (consolidated)

See Appeal No. 21-1568, Dkt. No. 15 (Order).

The following appeals, while perhaps not "related cases" within the meaning of this Court's rules, arise from *inter partes* review proceedings involving some of the same patents asserted in these cases:

Uniloc 2017 LLC v. Unified Patents, LLC, Nos. 20-1666, -1667 (Fed. Cir.) (consolidated)	Apple Inc. v. Uniloc 2017 LLC, Nos. 20-1575, -1638 (Fed. Cir) (consolidated)	Uniloc 2017 LLC v. Apple Inc., No. 20-1038 (Fed. Cir.)
Iancu v. Fall Line Patents, LLC, No. 20-853 (U.S.)	Uniloc 2017 LLC v. Apple Inc., No. 20-1228, -1229 (Fed. Cir.)	

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Cases

Statutes

Rules

Other

STATEMENT OF RELATED CASES

This is a collateral appeal regarding the district court's refusal to seal third-party confidential information arising out of five patent-infringement actions between (mostly) the same parties:

- *Uniloc USA Inc. et al. v. Apple Inc.*, No. 3:18-cv-00358-WHA (N.D. Cal.)
- *Uniloc 2017 LLC et al. v. Apple Inc.*, No. 3:18-cv-00360, -00363, -00365 & -00572-WHA (N.D. Cal.)[1]

Appellants Uniloc USA, Inc. ("Uniloc USA") and Uniloc Luxembourg S.A. ("Uniloc LUX") are the plaintiffs in the -358 case. Uniloc 2017 LLC ("Uniloc 2017"), Uniloc USA and Uniloc LUX (collectively "Uniloc") are the plaintiffs in the -360, -363, -365 and -572 cases ("-360 *et seq.* cases").[2] Appellee Apple Inc. ("Apple") is the defendant in all cases. Third-party Electronic Frontier Foundation ("EFF") is an intervenor in all cases.

The -360, -365 and -572 cases are stayed due to instituted *inter partes* reviews. Uniloc moved to dismiss without prejudice the -363 case on September 5,

[1] Cases will be referred to by their non-zero digits, *e.g.*, "the -360 case." All relevant pleadings in the -360, -363, -365 and -572 cases were filed in parallel. To avoid quadruplicate entries in the Joint Appendix, all items from the record below for these cases are from the docket of the -360 case, unless otherwise noted. The -358 case is an exception, as it took a different path.

[2] The district court allowed Uniloc 2017 to joint as plaintiff in the -360 *et seq.* cases. Appx674. Uniloc 2017 subsequently moved to join the -358 case, but the motion was denied. Appx903.

2018, which motion was granted on August 7, 2019; the to-be-sealed documents in the -363 case were filed between those dates.

The -358 case was dismissed on December 4, 2020. The substance of that dismissal is on appeal to this Court in Appeal No. 2021-1572. The -1572 appeal was briefly related to the instant appeals, *see* -1568 Appeal, Order (Feb. 1, 2021), but the Court deconsolidated the -1572 appeal pursuant to Uniloc's unopposed motion, *see id.*, Dkt. No. 15 (Feb. 25, 2021).

Some of the same to-be-sealed information at issue here was also submitted in eleven cases between Uniloc 2017 and Google LLC ("Google"): *Uniloc 2017 LLC v. Google LLC*, Nos. 4:20-cv-04355, -05330, -05333, -05334, -05339, -05341, -05342, -05343, -05344, -05345 & -05346-YGR (N.D. Cal.) (collectively "the *Google* cases"). The information was ordered sealed in the *Google* cases. Those cases were dismissed on December 22, 2020. The substance of those dismissals is on appeal to this Court in Appeal Nos. 2021-1498, -1500, 1501, -1502, -1503, -1504, -1505, 1506, -1507, -1508 & -1509.

Some of the same to-be-sealed information at issue here was also submitted in a case brought by Uniloc USA and Uniloc LUX against Motorola Mobility, LLC ("Motorola"): *Uniloc USA, Inc. v. Motorola Mobility, LLC*, C.A. No. 17-1658 (CFC) (D. Del.). The relevant information remains under seal in the

Motorola case. The *Motorola* case was dismissed on December 30, 2020. The substance of that dismissal is on appeal to this Court in Appeal No. 2021-1555.

The following table lays out the cases and appeals, and how they are related:

Case	Uniloc(s)	Defendant	Appeal	Subject Matter
-358 (N.D. Cal.)	USA, LUX	Apple	-1572	Standing
			-1573	Sealing (present appeal)
-360 (N.D. Cal.)	2017, USA, LUX	Apple	-1568	Sealing (present appeal)
-363 (N.D. Cal.)	2017, USA, LUX	Apple	-1569	
-365 (N.D. Cal.)	2017, USA, LUX	Apple	-1570	
-572 (N.D. Cal.)	2017, USA, LUX	Apple	-1571	
-4355 (N.D. Cal.)	2017	Google	-1498	Standing
-5330 (N.D. Cal.)	2017	Google	-1500	
-5333 (N.D. Cal.)	2017	Google	-1501	
-5334 (N.D. Cal.)	2017	Google	-1502	
-5339 (N.D. Cal.)	2017	Google	-1503	
-5341 (N.D. Cal.)	2017	Google	-1504	
-5342 (N.D. Cal.)	2017	Google	-1505	
-5343 (N.D. Cal.)	2017	Google	-1506	
-5344 (N.D. Cal.)	2017	Google	-1507	
-5345 (N.D. Cal.)	2017	Google	-1508	
-5346 (N.D. Cal.)	2017	Google	-1509	
-1658 (D. Del.)	USA, LUX	Motorola	-1555	Standing

This Court's determination of the present appeals should not impact the outcome of the -1572 appeal, the -1498 *et al.* appeals or the -1555 appeal, and vice versa. The Court's determination will, however, influence whether the materials filed in the underlying cases remain under seal.

STATEMENT OF PRIOR APPEALS

Some of the same issues were raised in a prior appeal of in the -360 *et seq.* cases in *Uniloc 2017 LLC v. Apple, Inc.*, 964 F.3d 1351 (Fed. Cir. 2020). *See* Appx484-504. In that opinion, this Court affirmed-in-part, vacated-in-part and remanded for further consideration of the to-be-sealed information. After further briefing, the district court below issued an order, Appx30-36, which forms the basis for the current collateral appeals.

The -358 case was separately appealed to this Court following dismissal on Section 101 grounds in *Uniloc USA, Inc. v. Apple Inc.*, 784 F. App'x 763 (Fed. Cir. 2019). *See* Appx476-483. This Court vacated and remanded for further consideration of standing issues which are not (directly) relevant to the current collateral appeals. Instead, these collateral appeals relate to whether the third-party licensing information filed in conjunction with the standing motions in all of the cases will remain under seal.

PRELIMINARY STATEMENT

This appeal relates to narrowly tailored redactions covering confidential business and patent-licensing information of more than 100 third-parties. On December 22, 2020, Judge Gonzalez Rogers of the Northern District of California issued an order sealing much of this information:

> Uniloc 2017 seeks to seal portions of two exhibits that identify third-party licensees and the amounts they paid for each license, as well as

> their confidential payment information. Pricing terms and confidential financial information are routinely sealed as materials that may be used to harass or harm a party's competitive standing. The requests are narrowly tailored and do not prevent the public from understanding the issues in this motion. Accordingly, Uniloc 2017's motion seal is GRANTED.

Uniloc 2017 LLC v. Google LLC, ___ F. Supp. 3d ___, No. 4:20-cv-04355-YGR, 2020 WL 7626430, at *13 n.23 (N.D. Cal. Dec. 22, 2020) (citation omitted). However, this is not the order on appeal.

Instead, earlier on December 22, 2020—literally the same day—Judge Alsup of the Northern District of California refused to seal some of the same documents and information, despite identical arguments presented to both judges of the same court:

> This order addresses the sealing of evidence submitted in a patent infringement suit. Accepting that several courts of appeal have held certain licensing and financial records sealable at times, on the record provided, the sealing motions are DENIED.

Appx30. This is the order on appeal.

The disparate treatment was even noted in the press, such as by Docket Navigator, which included squibs of the contradicting orders at the top its Patent Docket Report for December 28, 2020:

From: docketreport@docketnavigator.com
Sent: Monday, December 28, 2020 7:00 AM
To: Jacobs, Aaron
Subject: Patent Docket Report for December 28, 2020

DocketReport

Patent Report *December 28, 2020*

Identity of Third-Party Patent Licensees and Financial Details of the Licenses *Should* be Sealed

The court granted plaintiff's motion to seal portions of two exhibits that contained its licensing and financial information. " [Plaintiff] seeks to seal portions of two exhibits that identify third-party licensees and the amounts they paid for each license, as well as their confidential payment information. Pricing terms and confidential financial information are routinely sealed as materials that may be used to harass or harm a party's competitive standing. The requests are narrowly tailored and do not prevent the public from understanding the issues in this motion."

Uniloc 2017 LLC et al v. Google LLC, 4-20-cv-04355 (NDCA 2020-12-22, Order) (Yvonne Gonzalez Rogers)

Read Order | Docket Sheet

Identity of Third-Party Patent Licensees and Financial Details of the Licenses *Should Not* be Sealed

On remand, the court again denied plaintiff's motion to seal portions of a motion to dismiss that contained licensing and financial information. "Federal courts are public tribunals, not private mediators. They belong to the people, not to corporate America. . . . A patent is not a private agreement between private parties. . . . [Plaintiff's] reliance on the numerous district court orders which have sealed similar information underscores the larger problem of indiscriminate oversealing in patent and commercial cases nationwide. . . . [T]he theme among [plaintiff's] third party licensees who did previously request sealing is a concern that disclosure of their identities and license details will expose them to other non-practicing patent holders. . . . A patent owner is a tenant on a plot within the realm of public knowledge, and a licensee is her sub-tenant. The public has every right to account for all its tenants, all its sub-tenants, and (more broadly) anyone holding even a slice of the public grant. . . . The public has an interest in inspecting the valuation of the patent rights as revealed by [plaintiff's] transactions, particularly given secrecy so often plays to the patentee's advantage in forcing bloated royalties. . . . Conclusive here, though, is the fact that the dates and dollar amounts involved in [plaintiff's] patent licenses 'go to the heart of' the primary dispute, that of [plaintiff's] standing (or lack of) to sue."

Uniloc USA, Inc. et al v. Apple Inc., 3-18-cv-00358 (NDCA 2020-12-22, Order) (William H. Alsup)

Read Order | Docket Sheet

Appx1063.

Appellants respectfully submit that the district court on appeal—Judge Alsup, not Judge Gonzalez Rogers—made several mistakes of law and fact in refusing to seal the confidential information of more than 100 third-parties.

STATEMENT OF JURISDICTION

Uniloc appeals from the district court's December 22, 2020 Order re Sealing. Appx30-36. The district court has jurisdiction over the proceedings below pursuant to 28 U.S.C. §§ 1331, 1338(a) and 1367. This Court would have jurisdiction over an appeal from a final judgment in this case under 28 U.S.C. § 1295(a)(1), because Uniloc asserted claims for patent infringement. Indeed, as noted above, this Court currently has jurisdiction over just such an (unrelated) appeal in the -358 case.

This Court has jurisdiction over these non-final appeals pursuant to the collateral order doctrine. *Uniloc 2017 v. Apple*, 964 F.3d at 1357-58; *see also*, *e.g.*, *Oliner v. Kontrabecki*, 745 F.3d 1024, 1025 (9th Cir. 2014) (recognizing that in the Ninth Circuit "an order denying a motion to unseal or seal documents is appealable either as a final order under 28 U.S.C. § 1291 or as a collateral order").

STATEMENT OF THE ISSUES

1. Whether the district court made a mistake of law in concluding that all of the courts of the Northern District of California have been consistently wrong in sealing third-party patent-licensing information, based upon the novel theory that "[t]he public has every right to account for all its tenants, all its sub-tenants, and (more broadly) anyone holding even a slice of the public grant" of a patent. Appx34.

2. Whether the district court made a mistake of law in ignoring the evidence already before it, following remand from this Court with the instruction to consider that evidence.

3. Whether the district court abused its discretion in refusing to seal discrete portions of third-party information and documents, which information and documents relate to valuable and fiercely protected trade secrets.

STATEMENT OF THE CASE

This appeal relates to the district court's denial of a motion to redact or seal trade secret information belonging to more than 100 third-parties. The to-be-sealed information includes (1) a table with the financial terms of 109 licenses between third-party licensees and Uniloc; (2) excerpts of a declaration expressing the requests and concerns of twenty-three third-party licensees who asked the district court to maintain their information under seal; (3) eight individual declarations from third-party licensees who asked the district court to maintain their information under seal; (4) a non-party's memorandum which discloses that non-party's business analyses, as well as some of the third-party licensees' information at issue in the other documents; and (5) scattered filings which reference this confidential information. Attachment A to this Brief identifies and cross-references the few remaining items in the record.

To understand why these few items should be redacted or filed under seal, further background is required. Uniloc USA and Uniloc LUX brought suit against Apple to pursue their patent rights in five separate cases in the Eastern District of Texas. Apple moved to transfer these cases—along with others between the parties—to the Northern District of California, which motion was granted in December 2017. The cases in these collateral appeals were transferred in January 2018 and assigned to Judge Alsup. From there, the cases took two disparate paths to arrive before this Court.

I. The -360 *et seq.* cases.

A. The district court granted Uniloc's motion to add Uniloc 2017 to the cases and denied Apple's motion to dismiss.

In mid-2018, several Uniloc entities entered a series of corporate transactions which resulted in a new entity, Uniloc 2017, as the assignee of the patents-in-suit. So, in August 2018, the Uniloc parties in the -360 *et seq.* cases filed a Rule 25 motion to join Uniloc 2017 as the patent owner. Appx89 (Dkt. No. 119). Separately, in September 2018, the district court *sua sponte* stayed these cases pending IPRs, but allowed Apple to file a motion challenging Uniloc's standing and as to subject matter issues. Appx90 (Dkt. No. 131).

On October 25, 2018, Apple moved to dismiss the four -360 *et seq.* cases for lack of subject matter jurisdiction. Appx91 (Dkt. No. 135). In short, Apple argued that the Uniloc entities had granted their creditor, Fortress Credit Co. LLC

("Fortress"), a license with the right to sublicense in the event of a default. Apple further argued that there had been a default because the agreement required Uniloc to obtain at least $20,000,000 in licensing revenue by March 31, 2017, while Uniloc had only obtained about $14,000,000 by that time. Apple argued that, as a result, Uniloc lacked the right to exclude Apple from practicing the patents.

On January 17, 2019, the district court denied Apple's motion to dismiss and granted Uniloc's motion to add Uniloc 2017 as a plaintiff. *See* Appx666-675. The substantive correctness of the district court's ruling denying Apple's motion to dismiss is not at issue in this appeal. Rather, the question is whether the district court made a mistake of law in refusing to seal third-party information filed in association with Apple's motions to dismiss.

B. The parties submitted motions to seal third-party confidential information associated with Apple's motions to dismiss.

Apple's motion to dismiss was filed with documents and information that disclosed, *inter alia*, the individual licensing information of more than 100 third-parties, *see infra* Statement of the Case § I.E, including information that had been produced and designated by Uniloc as Highly Confidential under the Protective Order, *see* Appx1-29. So, concurrent with its motion, Apple filed an administrative motion to seal or redact the designated information. Appx349-355. As required by Local Rule 79-5(e)(1), Uniloc—as the "Designating Party"—filed a

declaration in support of Apple's administrative motion four days later. Appx356-359. Uniloc's opposition was accompanied by an administrative motion and declaration to seal additional information. Appx360-365. Apple's reply was accompanied by an administrative motion to seal a few more items, Appx366-372, for which Uniloc submitted a declaration in support, Appx373-375.

C. The district court denied the parties' motions to seal and the district court and denied EFF's motion to intervene.

On January 9, 2019, one day before oral arguments on Apple's motion to dismiss, third-party Electronic Frontier Foundation ("EFF") moved to intervene to oppose the parties' motions to seal. Appx93 (Dkt. No. 152).

On January 17, 2019, four days before Uniloc's deadline to respond to EFF's motion to intervene, the district court denied the parties' motions to seal and denied EFF's motion to intervene. Appx38-39.

D. The district court denied Uniloc's motion for leave to file a motion for reconsideration regarding the motions to seal and denied EFF's second motion to intervene.

On February 15, 2019, Uniloc filed a motion for leave to file a motion for reconsideration regarding the motions to seal. Appx95 (Dkt. No. 168). The substantive motion for reconsideration that Uniloc proposed to file was included as an exhibit. Appx418-435 ("motion for reconsideration"). Therein, Uniloc retrenched the proposed redactions and documents to be filed under seal, such that upwards of 90% of the previously confidential materials would be made public.

Uniloc's motion for reconsideration was accompanied by a fifteen-page, 5000-plus-word declaration that detailed, on an item-by-item basis, the individual grounds for redacting or sealing the remaining 10%. Appx761-776. Several of the exhibits accompanying Uniloc's motion for reconsideration were the subject of one more motion to file under seal and declaration, Appx392-409, to address, *inter alia*, eight of the thirteen third-party declarations asking the district court to keep their information under seal, Appx805-837 (sealed declarations); and statements from twenty-three third-parties who asked Uniloc to relay specific, sealed requests regarding their information to the district court, Appx767-772, ¶¶ 9-9.w.i.

Apple did not oppose Uniloc's motions.

On March 11, 2019, EFF filed its "Second Motion to Intervene for Limited Purpose of Opposing Uniloc's Motion for Reconsideration." Appx96 (Dkt. No. 177). Uniloc opposed. Appx96 (Dkt. No. 180).

On May 7, 2019, the district court acknowledged that "Apple's motion to dismiss for lack of standing did not directly depend upon information regarding the specific dollar amounts, financial terms, and names of the licensees in the various agreements (with Fortress or third-party licensees)." Appx42. Nonetheless, it denied Uniloc's motion for reconsideration. Appx43. The district court again denied EFF permission to intervene, other than as to an appeal. *Id*.

E. The few documents still at issue disclose more than 100 third-parties' confidential information.

With the motions to seal from the -360 *et seq.* cases now laid out, it is time to discuss the documents and information still at issue.

To establish its point that Uniloc had not reached the $20,000,000 threshold by March 31, 2017—a fact Uniloc did not dispute—Apple filed a number of Uniloc's confidential documents, including, *inter alia*: (1) documents that contain detailed financial information regarding Uniloc's licenses with third-parties; (2) Uniloc's loan agreements with Fortress; (3) cross-Uniloc-entity contracts; and (4) deposition transcripts that addressed some of these items. Uniloc's motion for reconsideration added several confidential items: (5) a declaration from Uniloc's counsel describing the documents and requests for confidentiality from more than thirty third-party licensees; and (6) declarations from third-party licenses.[3] Although these confidential materials will be discussed individually below, *see infra* Argument §§ III.B & III.C, one document should be brought to the fore.

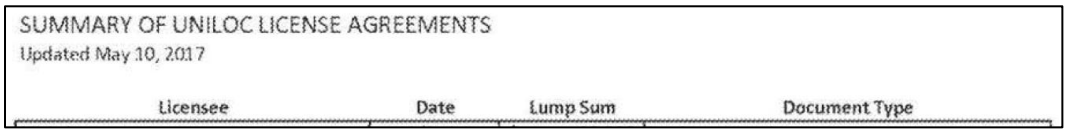
---

[3] More granularly:

Eight third-party licensees agreed to disclose their identities but asked Uniloc to relay to the district court their requests that the financial terms of their licenses remain under seal. Appx765-767 ¶¶ 8-8.h. Of these eight third-parties, five separately submitted unsealed declarations. Appx436-450.

Twenty-three third-party licensees asked Uniloc to relay to the district court—under seal—their individual requests for complete confidentiality and the reasons therefor. Appx767-772 ¶¶ 9-9.w.i. Of these twenty-three third-parties, eight also submitted sealed or redacted declarations. Appx805-837.

Exhibit A accompanying Apple's motion to dismiss was the Conformed Revenue Sharing and Note and Warrant Purchase Agreement. Appx708-734.[4] The last three pages include a table of 109 licenses between third-party licensees and Uniloc. Appx732-734. Each line identifies the licensee (*i.e.*, the third-party), the date of the license, the amount paid and the license type for each license:


SUMMARY OF UNILOC LICENSE AGREEMENTS
Updated May 10, 2017

Licensee	Date	Lump Sum	Document Type

Each such set of information is covered by a separate license agreement. Appx764 ¶ 4. Almost every one of these license agreements includes a confidentiality provision. *Id.* And nearly all of these agreements arose out of cases in which district courts issued protective orders. *See, e.g.*, Appx941-962 *Uniloc USA, Inc. v. Microsoft Corp*., No. 03-cv-440 (WES), Dkt. No. 23 (D.R.I. Apr. 30, 2004). So, this one Uniloc document includes trade secret information of

[4] Most of the documents still at issue were originally attached to the Winnard Declarations accompanying Apple's motion to dismiss and Apple's reply brief, *see* Appx353-355; Appx370-372, and the pleadings generally refer to them using those exhibit designations.

Local Rule 79-5(d)(1) requires to-be-sealed documents to be attached to declarations accompanying motions to seal. So, every time there was a motion to seal related to a given document, that document was (re)filed. This led to duplication and nested exhibit-numbering in the record. To avoid duplication in the Joint Appendix, only those versions attached to the last motion to seal are included. *See* Appx676-707.

more than 100 third-parties. Still, the district court denied the parties' motions to seal this third-party information and Uniloc's motion for reconsideration.

F. This Court affirmed-in-part, vacated-in-part and remanded for further action.

Uniloc filed an interlocutory appeal to this Court regarding the district court's denial of the motions to seal. EFF intervened. This Court heard oral arguments on April 8, 2020 and issued its opinion on August 9, 2020. Appx484-504 (published at 964 F.3d 1351 (Fed. Cir. 2020)).

First, this Court concluded that the district court did not abuse its discretion in denying "Uniloc's requests to seal its purportedly confidential information and that of its related entities" due to originally overbroad requests. Appx502.

Second, though, this Court concluded that the confidential information belonging to the more-than 100 licensees and third-parties was entitled to a closer look:

> Such third-parties were not responsible for Uniloc's filing of an overbroad sealing request. Their information calls for an analysis not dependent on the overbreadth rationale just discussed.
>
> The district court rejected Uniloc's attempt to prevent disclosure of information related to its third-party licensees, including the licensees' names, the duration of their licenses, and the specific royalty rate each licensee paid. Uniloc asserts that almost all of its third-party license agreements included a confidentiality provision, indicating that the information in the agreements was "proprietary and confidential," and that "the vast majority of these agreements were entered into under the auspices of protective orders signed by district court judges." Significantly, moreover, many of Uniloc's licensees

> have submitted declarations stating that they wish their licensing information to remain confidential and that the disclosure of such information would cause them material competitive injury.
>
> As to these third-party materials, we conclude that the district court failed to make findings sufficient to allow us to adequately assess whether it properly balanced the public's right of access against the interests of the third-parties in shielding their financial and licensing information from public view. In this regard, there is no indication in the record that the court assessed whether any of the third-party information was "protectable as a trade secret or otherwise entitled to protection under the law." We therefore vacate those portions of the district court's orders which denied sealing or redaction of the purportedly confidential information of third-parties and remand so that the court may make particularized determinations as to whether and, if so, to what extent, the materials of each of these parties should be made public.

Appx502-504 (citations omitted).

And, this Court recognized that non-party Fortress, as Uniloc's lender, presented an intermediate situation between Uniloc and the other third-parties. So, this Court left to the district court's discretion the question of whether Fortress's materials should be kept under seal. Appx502 at n.8.

G. Uniloc and Fortress unsealed their information and Uniloc filed a motion asking the district court to seal the third-parties' information.

Following issuance of the mandate, Uniloc identified all purely Uniloc information and materials for filing into the public record. And, rather than drag out the issue, Fortress agreed to submit its materials into the public record as well. So, on November 19, 2020, Uniloc filed two motions. First, Uniloc filed a motion to place most of the documents into the public record. Appx661-664. And,

second, Uniloc filed a motion to seal the few remaining documents with third-party information. Appx676-700. The outcome of that second motion is part of the basis for the current appeals, as will be discussed further below.

II. The -358 case.

The -358 case took a different path to end up in mostly the same place.

On May 18, 2018—five months before Apple moved to dismiss on standing grounds in the -360 *et seq.* cases—the district court held that the patent-at-issue in the -358 case was directed to patent-ineligible subject matter and granted Apple's motion for judgment on the pleadings. Appx70 (-358 case, Dkt. No. 99). Uniloc appealed but, during that appeal, Apple raised the question of standing. So, this Court remanded to address the standing issue without touching upon patentability. Appx476-483.

Following remand and after further discovery, on October 1, 2020, Uniloc moved to add Uniloc 2017 as a party to the -358 case and for a declaration that the court had subject matter jurisdiction. Appx66 (-358 case, Dkt. No. 158). On October 22, 2020, Apple filed an opposition to Uniloc's motion, Appx66 (-358 case, Dkt. No. 163); and separately filed a motion to dismiss on standing grounds in the -358 case, Appx66 (-358 case, Dkt. No. 165). Apple filed motions to seal associated with each. Appx505-518.

Uniloc timely filed two Local Rule 79-5(e)(1) declarations. In the first, Uniloc stated that none of the documents accompanying Apple's opposition to the motion to add Uniloc 2017 needed to be sealed. Appx617-618. In the second, Uniloc and Fortress asked the court to seal just (1) three lines from a deposition transcript that disclosed third-party licensee information; (2) a Fortress Memorandum that disclosed Fortress's confidential information, as well as dozens of third-party licensees' information; and (3) four lines of Apple's memorandum which disclosed details of the Fortress Memorandum. Appx619-625. The outcome of that second request to seal is the other part of the basis for the current appeals, as will be discussed further below.

On December 4, 2020, the district court granted Apple's motion to dismiss and denied Uniloc's motion to add Uniloc 2017 as a party. Appx891-904. That order will be addressed in the -1572 appeal; the substance is not at issue here.

* * *

This brings the matters in the present appeals to the same place, with Uniloc's motion to seal in the -360 *et seq*. cases lining up (in time and some substance) with Apple's motion to seal similar information in the -358 case.

III. EFF moved to intervene, again.

On November 12, 2020, EFF filed its "Third Motion of Electronic Frontier Foundation to Intervene for Limited Purpose of Opposing Motions to Seal," in

each of the cases, now including the -358 case. Appx99 (Dkt. No. 217). Uniloc opposed. Appx102 (Dkt. No. 229).

IV. The district court denied the parties' motions to seal the third-parties' confidential information and permitted EFF to intervene in the cases below.

On December 17, 2020, the district court held a hearing regarding (1) Uniloc's motion to seal third-party information following remand from this Court in the -360 *et seq.* cases; (2) Apple's motion to seal similar information from the -358 case; and (3) EFF's motion to intervene. *See* Appx924-940.

The district court began the hearing by stating that it did not intend to seal anything because "we are dealing with the public right here . . . . [¶] And ownership of that public right ought to be known. And anyone who has any slice of it ought to be known and open to public view because it is a public right, not a private right created like a trade secret agreement." Appx925-926.

Uniloc noted more than thirty third-party licensees had submitted declarations and statements which explained why they viewed their licensing information as trade secrets, but the district court did not believe this was sufficient:

> THE COURT: These licensees took these licenses, way back when, whenever they took them. They were -- they knew the day would come when someone would want to know their identities. It wasn't as if they got tricked into oh, my goodness, my identity is going to come out now; and other people are going to sue me and try to soak me too.

> No. They knew the day would come when their names would become public, and now is the time because of the summary judgment motion. So I'm sorry but I feel very strongly about this.

Appx930.

The district court then pressed Apple to take a position regarding whether the materials should be sealed:

> THE COURT: Are you saying to me that on appeal, you are just going to stand mute? That's what I hear you saying.
>
> Whatever order I come out with, you are going to say: Judge, we wash our hands of it. That judge down there, no, let -- it is just up to -- you know, the federal -- we don't care, Judge. We have already won our case. That is Apple's position?

Appx935. Apple's counsel confirmed that it was Apple's position that the information was sealable. *Id.* The district court stated that it was "disappointed" that Apple would not defend the district court's refusal to seal these materials and allowed EFF to intervene in the case—not just on appeal, as before, but fully in the cases below—because of Apple's recognition that this information is sealable:

> THE COURT: . . . . I'm going to allow EFF to intervene in this case because Apple in a -- I'm disappointed in Apple. After all the trouble I have gone to, to scour this record on this brand new theory of Apple's about lack of standing -- and I had to scour the record in order to understand it -- and finally I give Apple the relief it wants.
>
> And when it comes to the ancillary issue of whether or not the material that I relied on and reviewed should be under seal, All-in-all we don't care, Judge. We have already won. Oh, Judge, we will just stand mute on appeal because there are going to be times when we want to take the other side of this position because we want to keep our information under seal. So thank you, Apple.

> EFF is going to stand in for the correct argument here. EFF you are now in the case as a party.

Appx935-936.

On December 22, 2020, the district court issued its written order denying the parties' motions to seal and allowing EFF to intervene. Appx30-36. This order is the basis for the current appeal and will be discussed in greater detail below. *See infra* Argument § III.A.

V. Subsequent sealing orders from the Northern District of California.

Before turning to the arguments regarding the order on appeal, two subsequent orders from the Northern District of California should be addressed.

A. *Uniloc v. Google*: Judge Gonzalez Rogers of the Northern District of California sealed some of the same information on the same day.

The order on appeal in these cases was filed at 10:57 a.m. PT on December 22, 2020. But, that was not the last order of the day to address some of this information.

In its copy-cat motion to dismiss—the substance of which is at issue in the -1498 *et seq.* appeals—Google filed some of the same materials, including Exhibit A, with its list of 109 licenses. Those materials were the subject of an October 2, 2020, motion to seal, Appx1056-1059, and a declaration from Uniloc's counsel that paralleled the declaration from the *Apple* cases, Appx1060-1062. The motion

and declaration in support laid out the details of the *Apple* cases with respect to the same documents and information. *Compare* Appx1056-1062 *with* Appx761-776.

At 3:30 p.m. PT on December 22, 2020, Judge Gonzalez Rogers ordered sealed the materials submitted on her docket, including some of the very same materials that the order on appeal refused to seal:

> Uniloc 2017 seeks to seal portions of two exhibits that identify third-party licensees and the amounts they paid for each license, as well as their confidential payment information. (Dkt. No. 354.) Pricing terms and confidential financial information are routinely sealed as materials that may be used to harass or harm a party's competitive standing. *See In re Elec. Arts, Inc.*, 298 F. App'x 568, 569 (9th Cir. 2009); *In re Qualcomm Litig.*, No. 3:17-cv-0108-GPC-MDD, 2017 WL 5176922, at *2 (S.D. Cal. Nov. 8, 2017). The requests are narrowly tailored and do not prevent the public from understanding the issues in this motion. Accordingly, Uniloc 2017's motion seal is GRANTED.

Uniloc 2017 LLC v. Google LLC, ___ F. Supp. 3d ___, No. 4:20-cv-04355-YGR, 2020 WL 7626430, at *13 n.23 (N.D. Cal. Dec. 22, 2020).

B. *Finjan v. Juniper Network*: The district court below cited its order in this case in another instance where it denied a motion to seal.

One final order worth mentioning was issued by the district court below in another patent case, *Finjan v. Juniper Network*, a few months after the order on appeal. *See* Appx1065-1067, *Finjan, Inc. v. Juniper Network, Inc.*, No. 17-cv-05659-WHA, Dkt. No. 656 (N.D. Cal. Feb. 10, 2021). As in the present appeals, the district court's order in *Finjan v. Juniper Network* followed a remand from this

Court regarding that district court's refusal to seal documents. *See Finjan, Inc. v. Juniper Network, Inc.*, 826 F. App'x 928 (Fed. Cir. 2020).

In short, in *Finjan v. Juniper Network*, this Court considered an appeal wherein the district court below previously denied Finjan's motion to redact eight lines of the district court's *Daubert* order "that Finjan asserts disclose confidential licensing terms discussed between Finjan and third-party licensees." *Id.* at 929. In that appeal, this Court—just as in the *Uniloc 2017 v. Apple* appeal—explained that the district court below had not provided any analysis regarding the third-party licensee's interest in its trade secrets. And so, this Court— just as in the *Uniloc 2017 v. Apple* appeal—remanded for further consideration:

> In *Uniloc*, for example, we vacated and remanded a portion of an order that "failed to make findings sufficient to allow us to adequately assess whether [the district court] properly balanced the public's right of access against the interests of the third parties in shielding their financial and licensing information from public view." We do the same here. The district court did not perform the required analysis.

Id. (quoting *Apple Inc. v. Samsung Elecs. Co.*, 727 F.3d 1214, 1220 (Fed. Cir. 2013)) (citation omitted).

Following remand, on February 10, 2021, the district court below—the court on appeal here—denied the Finjan's renewed motion to seal based upon the same theories found in the order on appeal. The only order cited in support of its decision to unseal those materials was the order on appeal here. Appx1065-1067, *Finjan, Inc. v. Juniper Network, Inc.*, No. 17-cv-5659, Dkt. No. 656 (N.D. Cal.

Feb. 10, 2021). In other words, the only support the district court below could find in *Finjan v. Juniper Network* for the proposition that licensing information should be unsealed was the same court's prior order in these cases.

SUMMARY OF THE ARGUMENT

The district court's order is flawed in three respects.

First, the district court made a mistake of law in concluding that all of the courts of the Northern District of California—and Ninth Circuit and this Court—have uniformly been wrong to seal third-party patent-licensing information. The district court came to this erroneous conclusion by devising the new theory that patent licenses are both special and exempt from the universally recognized principal that licensing details are trade secrets that are sealable. In so doing, the district court contravened not only the overwhelming weight of precedent, but every single case cited by any party, the intervenor and the district court itself.

Second, the district court made a mistake of law in ignoring this Court's express instructions in its previous remand to consider the many declarations and statements already in the record from third-parties "stating that they wish their licensing information to remain confidential and that the disclosure of such information would cause them material competitive injury." Appx503.

Third, the district court abused its discretion by refusing to seal the third-parties' licensing information and other trade secrets. Financial and licensing

information indisputably qualify as trade secrets, as does confidential business information. Disclosure of this detailed information would indelibly injure more than 100 third-parties by giving their future licensing and contracting partners an unfair, asymmetric advantage in negotiations. The district court made a mistake of law by failing to give the appropriate weight to these third-parties' compelling interests.

ARGUMENT

I. Law

Documents filed with courts are presumed to be accessible to the public to allow the public to hold courts accountable for their reasoning. In short, the public should presumptively be able to confirm in the given case that the court came to the right conclusion. There is, however, no irrebuttable right to access documents just because the information contained in them is interesting.

In *Center for Auto Safety v. Chrysler Group, LLC*, 809 F.3d 1092, 1096 (9th Cir. 2016), the Ninth Circuit stated:

> "It is clear that the courts of this country recognize a general right to inspect and copy public records and documents, including judicial records and documents." *Nixon v. Warner Commnc'ns, Inc.*, 435 U.S. 589, 597 (1978). Following the Supreme Court's lead, "we start with a strong presumption in favor of access to court records." *Foltz v. State Farm Mut. Auto. Ins. Co.*, 331 F.3d 1122, 1135 (9th Cir. 2003).

Continuing—and this is key—the Ninth Circuit explained the touchstone of public's interest: "The presumption of access is 'based on the need for federal

courts, although independent—indeed, particularly because they are independent—to have a measure of accountability and for the public to have confidence in the administration of justice.'" *Id.* (quoting *U.S. v. Amodeo*, 71 F.3d 1044, 1048 (2d Cir. 1995)); *see also, e.g.*, *Pintos v. Pac. Creditors Ass'n*, 605 F.3d 665, 679 (9th Cir. 2010) ("[The] relevant factors" include the "public interest in understanding the judicial process and whether disclosure of the material could result in improper use of the material for scandalous or libelous purposes or infringement upon trade secrets.") (quoting *Hagestad v. Tragesser*, 49 F.3d 1430, 1434 (9th Cir. 1995)); *Phillips ex rel. Estates of Byrd v. Gen. Motors Corp.*, 307 F.3d 1206, 1213 (9th Cir. 2002) ("[T]he public policy reasons behind a presumption of access to judicial documents [are] judicial accountability [and] education about the judicial process . . . .").

Of course, "the right to inspect and copy judicial records is not absolute." *Nixon*, 435 U.S. at 598. Rather, "'the common-law right of inspection has bowed before the power of a court to insure that its records' are not . . . sources of business information that might harm a litigant's competitive standing." *Id.* (quoting *In re Casewell*, 18 R.I. 835, 836, 29 A. 259 (1893)). In the Ninth Circuit, the question is whether there are "compelling reasons" to maintain the documents under seal in matters that are "more than tangentially related to the merits of the case." *Center for Auto Safety*, 809 F.3d at 1096; *see also Kamakana v. City & Cty.*

of Honolulu, 447 F.3d 1172, 1178 (9th Cir. 2006). Even if the public has a legally cognizable interest in access to a document, the presumption of public access is rebutted where the document's owner establishes a compelling reason to keep it sealed.

The Ninth Circuit recognizes that there are compelling reasons to seal where the release of particular "'court files *might have become* a vehicle for improper purposes,' such as the use of records to gratify private spite, promote public scandal, circulate libelous statements, or *release trade secrets*." *Kamakana*, 447 F.3d at 1179 (quoting *Nixon*, 435 U.S. at 598) (emphasis added); *see also Apple v. Samsung*, 727 F.3d at 1228 (applying Ninth Circuit law) (reversing the district court; ordering sealed "market research reports [that] contain information that Apple's competitors could not obtain anywhere else"); *Apple Inc. v. Psystar Corp.*, 658 F.3d 1150, 1162 (9th Cir. 2011) ("publication of materials that could result in infringement upon trade secrets has long been a factor that would overcome" public access).

"The most commonly accepted definition of trade secrets," *Aronson v. Quick Point Pencil Co.*, 440 U.S. 257, 266 (1979), is found in comment (b) to section 757 of the first Restatement of Torts. *See, e.g.*, *In re Elec. Arts, Inc.*, 298 Fed. App'x 568, 569 (9th Cir. 2008). It defines a "trade secret" as "any formula, pattern, device or compilation of information which is used in one's business, and which

gives him *an opportunity* to obtain an advantage over competitors who do not know or use it." Rest. of Torts § 757, cmt. b (1939) (emphasis added); *see also id.* (listing factors).

The irreparable harm that would result from disclosure of trade secrets is as undeniable as it is obvious. *See, e.g.*, *Am. Standard Inc. v. Pfizer Inc.*, 828 F.2d 734, 741 (Fed. Cir. 1987) (recognizing harms of disclosure of confidential business information to competitors and collecting cases). "A trade secret once lost is, of course, lost forever." *North Atl. Instruments, Inc. v. Haber*, 188 F.3d 38, 49 (2d Cir. 1999) (quotation omitted); *see also, e.g.*, *In re Elec. Arts, Inc.*, 298 Fed. App'x at 570 (recognizing that once trade secret information is made public, a party may be "irreparably damaged").

Courts within the Ninth Circuit regularly find compelling reasons to seal documents containing valuable, competitive business information, because they are recognized trade secrets. For example, in *In re Electronic Arts*, the Ninth Circuit reversed a district court's denial of a request to seal "pricing terms, royalty rates, and guaranteed minimum payment terms found in a license agreement which were plainly within the definition of 'trade secrets.'" *In re Elec. Arts*, 298 Fed. App'x. at 569-70; *see also, e.g.*, *Apple v. Samsung*, 727 F.3d at 1222.

Other non-public information regarding pricing strategy, business decision-making and financial records also constitute trade secrets that may be sealed.

Rodman v. Safeway, Inc., No. 3:11-cv-03003-JST, 2014 WL 12787874, at *2 (N.D. Cal. Aug. 22, 2014); *see, e.g.*, *Arista Networks, Inc. v. Cisco Sys., Inc.*, No. 5:16-cv-00923-BLF, 2018 WL 2010622, at *2-3 (N.D. Cal. Apr. 30, 2018) (sealing, *inter alia*, "highly confidential and sensitive information relating to Cisco's financial information and internal development strategies," "highly confidential and sensitive information relating to Arista's financial and customer information," and "confidential settlement terms between Cisco and third-party, Huawei Technologies"); *Juicero, Inc. v. iTaste Co.*, No. 5:17-cv-01921-BLF, 2017 WL 8294276, at *2 (N.D. Cal. Jun. 28, 2017) (sealing, *inter alia*, "confidential financial and business information"); *Van v. Language Line Servs., Inc.*, No. 5:14-cv-03791-LHK, 2016 WL 3566980, at *2 (N.D. Cal. Jun. 6, 2016) (ordering sealed "the identities of Defendants' clients, billing rates, billing amounts, and the subject matter of calls"); *Transperfect Global, Inc. v. MotionPoint Corp.*, No. 4:10-cv-02590-CW, 2014 WL 4950082, at *1 (N.D. Cal. Sept. 25, 2014) (sealing, *inter alia*, "confidential financial and marketing information"); *see also, e.g.*, *McDonnell v. Southwest Airlines Co.*, 292 F. App'x 679, 680 (9th Cir. 2008) (affirming finding that "compelling reasons" supported denying public access to "documents contain[ing] trade secrets and confidential procedures and communications").

As is most relevant in these appeals, licensing information in patent cases is invariably sealed "because disclosure could create an asymmetry of information in the negotiation of future licensing deals." *Apple Inc. v. Samsung Elecs. Co., Ltd.*, No. 5:11-cv-01846-LHK, 2012 WL 4933287, at *2 (N.D. Cal. Oct. 16, 2012) ("*Apple v. Samsung* (N.D. Cal. October Order)"). In practice, licensing information is almost a *per se* basis upon which to seal. *See, e.g.*, *In re Elec. Arts*, 298 F. App'x at 569-570. As Judge Koh explained:

> The Ninth Circuit has held, and [the Northern District of California] has previously ruled, that pricing terms, royalty rates, and minimum payment terms of licensing agreements *plainly constitute trade secrets and thus are sealable.*

Apple Inc. v. Samsung Elecs. Co., Ltd., No. 5:11-cv-01846-LHK, 2012 WL 5988570, at *4 (N.D. Cal. Nov. 29, 2012) ("*Apple v. Samsung* (N.D. Cal. November Order)") (emphasis added). So, it should come as no surprise that the judges of the Northern District of California, including, *inter alia*, Judges Chen,[5]

[5] *See, e.g.*, Appx994-996, *Abbvie Inc. v. Novartis Vaccines & Diagnostics, Inc.*, No. 3:3:17-cv-01815-EMC, Dkt. No. 64 at 1 (July 11, 2017).

Davila,[6] Freeman,[7] Gilliam,[8] Gonzalez Rogers,[9] Hixson,[10] Illston,[11] Koh,[12] Keulen,[13] LaPorte,[14] Orrick,[15] Spero,[16] Tigar,[17] White,[18] and Wilken,[19] routinely seal licenses and licensing information.

---

[6] *See, e.g.*, Appx997-1001, *PersonalWeb Techs LLC v. IBM Corp.*, No. 5:16-cv-01226-EJD, Dkt. No. 347 at 4 (N.D. Cal. July 27, 2017) (sealing pleadings and exhibits that "contain[] confidential business information, confidential financial information related to [the defendant] and third-parties, and confidential settlement and license terms relating to third-parties").

[7] *See, e.g.*, Appx1005-1007, *Finjan v. Blue Coat Sys., LLC*, No. 5:15-cv-03295-BLF, Dkt. No. 398 at 2 (N.D. Cal. Oct. 31, 2017) (sealing "information relating to [plaintiff's] confidential business and licensing practices").

[8] *See, e.g.*, *In re Koninklijke Philips Patent Litig.*, No. 4:18-cv-01885-HSG, 2020 WL 1865294, at *1 (N.D. Cal. Apr. 13, 2020) ("Courts have found that 'confidential business information' in the form of "license agreements, financial terms, details of confidential licensing negotiations, and business strategies' satisfies the 'compelling reasons' standard." (quoting *In re Qualcomm Litig.*, No. 3:17-cv-0108-GPC-MDD, 2017 WL 5176922 (S.D. Cal. Nov. 8, 2017)); *Plexxikon Inc. v. Novartis Pharm. Corp.*, No. 4:17-cv-04405-HSG, 2020 WL 1233881, at *2 (N.D. Cal. Mar. 13, 2020) ("[T]he parties have narrowly tailored their requested redactions to confidential and proprietary business, scientific, manufacturing, sales, or licensing information. The public release of these documents could give non-party competitors an unfair advantage in the development or marketing of rival products."); *Big Baboon, Inc. v. SAP Am., Inc.*, No. 4:17-cv-02082-HSG, 2019 WL 1791421, at *4 (N.D. Cal. Apr. 24, 2019), *aff'd*, 819 F. App'x 928 (Fed. Cir. 2020) (sealing appendices to a license agreement between defendant and third-parties); *Synchronoss Techs., Inc. v. Dropbox, Inc.*, No. 4:16-cv-00119-HSG, 2018 WL 6002319, at *3 (N.D. Cal. Nov. 15, 2018) (sealing documents that "contain highly confidential, trade secret, and sensitive business information and practices of [plaintiff] and third-parties . . . including specific terms of confidential license and settlement agreements between [plaintiff] and third-party entities"); Appx1069-1072, *TVIIM, LLC v. McAfee, Inc.*, No. 4:13-cv-04545-HSG, Dkt. No. 218 (N.D. Cal. June 23, 2015) (sealing "patent, software, and/or technology licensing information" that "could be used by McAfee's competitors to McAfee's disadvantage, particularly because it reveals McAfee's licensing history").

[9] *See, e.g.*, *Uniloc v. Google*, 2020 WL 7626430, at *21 n.23 ("Pricing terms and confidential financial information are routinely sealed as materials that may be used to harass or harm a party's competitive standing."); *Windy City Innovations, LLC v. Facebook, Inc.*, No. 4:16-cv-01730-YGR, Dkt. No. 216 at 6 (N.D. Cal. Sept. 25, 2019) (sealing "terms of licenses").

[10] *See, e.g.*, Appx1051, *Finjan, Inc. v. Juniper Network, Inc.*, No. 3:17-cv-05659-WHA (TSH), Dkt. No. 570 (N.D. Cal. July 1, 2019) (granting redactions regarding licenses). Magistrate Judge Hixson's ruling in the *Finjan v. Juniper Network* case is noteworthy for several reasons. First, the case was assigned to Judge Alsup and referred to Judge Hixson for discovery matters. Second, just as in the present case, the defendant filed the plaintiff's documents under seal. *See* Appx1045-1046, *id.*, Dkt. No. 542 (defendant's motion). Third, those sealed documents included information regarding plaintiff's licenses and licensing discussions. *See* Appx1047-1049, *id.*, Dkt. No. 552 (plaintiff's declaration).

[11] *See, e.g.*, *Verinata Health, Inc. v. Sequenom, Inc.*, No. 3:12-cv-00865 SI, 2014 WL 12789020, at *1 (N.D. Cal. Aug. 8, 2014) (sealing deposition excerpts because of the "risk to Sequenom's licensing negotiations with third parties and Sequenom's competitive interests"); Appx970-971, *Bluestone Innovations LLC v. Nichia Corp.*, No. 3:12-cv-00059-SI, Dkt. No. 285 at 2 (N.D. Cal. Apr. 15, 2013) (sealing "documents [that] discuss confidential information, such as explicit details regarding negotiations in licencing [sic] agreements and internal decision-making processes").

[12] *See, e.g.*, *Apple Inc. v. Samsung Elecs. Co., Ltd.*, No. 5:11-cv-01846-LHK, 2012 WL 3283478, at *6 (N.D. Cal. Aug. 9, 2012) ("*Apple v. Samsung* (N.D. Cal. August Order") ("[T]he Court will follow the Ninth Circuit's guidance and seal all information related to the payment terms of Apple's licensing agreements."); *Abaxis, Inc. v. Cepheid*, No. 10-cv-02840-LHK, 2011 WL 6002522, at *1 (N.D. Cal. Nov. 30, 2011) (sealing "a term sheet from 2005 licensing negotiations between Cepheid and Abaxis").

[13] *See, e.g.*, Appx1052-1055, *X One Inc. v. Uber Techs., Inc.*, No. 5:16-cv-06050-LHK (SVK), Dkt. No. 243, at 4 (N.D. Cal. Aug. 2, 2019) (sealing information related to defendant's "third-party patent license agreements, including the identities of the confidential third parties, the pricing terms and licensing fees, and the specific intellectual property assets subject to the licenses").

[14] *See, e.g.*, *Dodocase VR, Inc. v. MerchSource, LLC*, No. 3:17-cv-07088-AGT, 2018 WL 5619799, at *1 n.1 (N.D. Cal. May 22, 2018) (sealing documents "containing confidential settlement information in the form of sensitive pricing

information that could be used to Plaintiff's disadvantage by existing or potential licensees").

[15] *See, e.g.*, *Intel Corp. v. Tela Innovations, Inc.*, No. 3:18-cv-02848-WHO, 2021 WL 783560, at *14 (N.D. Cal. Mar. 1, 2021) (sealing "discrete pieces of confidential business, financial, and licensing information"); *Huawei Techs., Co., Ltd. v. Samsung Elecs. Co., Ltd.*, No. 3:16-cv-02787-WHO, 2018 WL 1784065, at *12 (N.D. Cal. Apr. 13, 2018) (sealing pleadings and exhibits related to licensing); Appx1002-1004, *Autodesk, Inc. v. Alter*, No. 3:16-cv-04722-WHO, Dkt. No. 108 at 2 (N.D. Cal. Sept. 5, 2017) (sealing in its entirety a license agreement between plaintiff and Walt Disney Pictures ("WDP") because "publication of these terms would put WDP at a notable negotiating disadvantage in future licensing negotiations. In addition, WDP is not a party."); *see also, generally*, Appx1028-1034, *Autodesk, Inc. v. Alter*, No. 3:16-cv-04722-WHO, Dkt. No. 174 (July 19, 2018) (sealing licensing information).

[16] *See, e.g.*, Appx1068, *SmugMug, Inc. v. Virtual Photo Store LLC*, No. 4:09-cv-02255 CW (JCS), Dkt. No. 69 (Nov. 6, 2009) (sealing information regarding amount of third-party license).

[17] *See, e.g.*, Appx1083-1086, *Droplets, Inc. v. Yahoo!, Inc.*, No. 4:12-cv-03733-JST, Dkt. No. 638 at 3 (N.D. Cal. Jan. 28, 2021) (sealing information regarding licenses); Appx1080-1082, *Symantec Corp. v. Zscaler, Inc.*, No. 4:17-cv-04426-JST, Dkt. No. 356 at 2 (N.D. Cal. Oct. 29, 2019) (sealing information regarding licensing agreements); Appx974-988, *Icon-IP Pty Ltd. v. Specialized Bicycle Components, Inc.*, No. 4:12-cv-3844-JST, Dkt. No. 217 at 5 (Mar. 4, 2015) ("This exhibit contains information about assignments, and consulting and license agreements between a third party consultant and Specialized. The Court is satisfied that release of this information would result in an invasion of the third party's privacy, that Specialized would suffer competitive harm if this material were made public, and that there are therefore compelling reasons to file this exhibit in its entirety under seal. . . . More specifically, the Court is satisfied that disclosing the terms of these agreements would put Specialized at a disadvantage in future negotiations for similar agreements.") (citations omitted).

[18] *See, e.g.*, Appx991-993, *ChriMar Sys. Inc. v. Cisco Sys. Inc.*, No. 4:13-cv-01300-JSW, Dkt. No. 413 at 3 (Aug. 12, 2016) (sealing entire license agreement).

[19] *See, e.g.*, *Digital Reg of Texas, LLC v. Adobe Sys., Inc.*, No. 4:12-cv-01971-CW, 2014 WL 6986068, at *1 (N.D. Cal. Dec. 10, 2014) ("[T]he redacted portions disclose details of Digital Reg's patent licenses and that public disclosure of this information would harm Digital Reg by placing it at a disadvantage in future

Not to belabor the point, but examples from Northern District of California sealing licensing information are *legion*. Appellants stopped adding citations to the footnotes above to avoid running into the word-limit for this Brief. This is not a "both sides" situation; other than the district court below, the cases are all on the side of sealing this information.

In fact, the judge below had—until these cases—consistently recognized that that disclosure of patent licensing information "would cause great and undue harm" to both litigants and their licensees. Appx963-964, *Oracle Am., Inc. v. Google Inc.*, No. 3:10-cv-03561-WHA, Dkt. No. 687 at 2 (N.D. Cal. Jan. 10, 2012). For example, the district court below ordered sealed the entirety of a defendant's license in another patent case just one week after denying Uniloc's motion for reconsideration:

> [Plaintiff Finjan sought to seal Exhibit 7. Defendant] Juniper declares that Exhibit 7, which consists of a confidential license agreement, constitutes a trade secret (*id.* ¶ 10). *See In re Elec. Arts, Inc.*, 298 F. App'x 568, 569 (9th Cir. 2008). . . . Compelling reasons having been shown, Finjan's motion to seal Exhibit 7 in its entirety . . . is GRANTED.

---

licensing negotiations. The Court finds good cause to grant the motion."); Appx968-969, *Tessera, Inc. v. Advanced Micro Devices, Inc.*, No. 4:05-cv-04063-CW, Dkt. No. 1036 at 2 (N.D. Cal. Aug. 10, 2012) (sealing "royalty reports with financial information about payments"); *Powertech Tech., Inc., v. Tessera, Inc.*, No. 4:11-cv-06121-CW, 2012 WL 1969039, at *1-2 (N.D. Cal. May 31, 2012) (granting motion to seal details of license agreement).

Appx1042-1044, *Finjan, Inc. v. Juniper Network, Inc.*, No. 3:17-cv-05659-WHA, Dkt. No. 485 at 2 (N.D. Cal. May 22, 2019).

The order on appeal denigrated Uniloc's counsel's declaration as "hearsay" in refusing to seal third-party information, Appx33, but declarations from the designating party's counsel are the standard evidence used to support motions to seal in the Northern District of California. *See* L.R. 79-5(d)(1) & 79-5(e)(1). Nearly all the orders cited above at footnotes 5-19 were based upon counsels' declarations. So too, the district court below had—until these cases—routinely relied upon counsels' declarations. For example, in *Finjan v. Juniper Network*, a partner at Juniper Network's outside counsel submitted a declaration in support of the motion to seal, with only the following references to Exhibit 7:

Document	Finjan's Designations of Portions to Be Sealed	Juniper's Designations of Portions to Be Sealed	Juniper's Basis for Sealing
* * *			
Exhibit 7 to Kastens Declaration	Entire Exhibit	Entire Exhibit	Confidential License Agreement
* * *			

10. Additionally, I am informed and believe that Exhibit 7 is a full, unredacted confidential license agreement between Juniper and a third party, which is information that falls within the definition of a "trade secret." *In re Elec. Arts, Inc.*, 298 F. App'x 568, 569 (9th Cir. 2008).

Appx1038-1041, *Finjan v. Juniper Network*, No. 3:17-cv-05659-WHA, Dkt. No. 418 (N.D. Cal. Apr. 5, 2019) (highlighting added). The district court below held that this constituted "[c]ompelling reasons having been shown," and so sealed

Exhibit 7 in its entirety. Appx1042-1044, *Finjan v. Juniper Network*, No. 3:17-cv-05659-WHA, Dkt. No. 485 at 2.

In *Oracle*, an associate at defendant Google's outside counsel submitted a declaration in support of a motion to seal. The following shows the only mention of licensing information in that declaration:

> 3. Exhibit J to the Dearborn Decl. (Dkt. No. 573) is an excerpt from the Expert Report of Dr. Gregory K. Leonard ("the Leonard Report"). The Court previously has granted a request to file the Leonard Report under seal (Dkt. No. 583). The Leonard Report contains information that has been designated HIGHLY CONFIDENTIAL – ATTORNEY'S EYES ONLY pursuant to the stipulated protective order in this case. The report and the underlying documents contain Google's sensitive, non-public financial data, such as costs, revenues, and profits associated with Android. The report and underlying documents also contain non-public information about Google's consideration of and potential financial impact from alternatives to the intellectual property at issue in this lawsuit. Additionally, the report contains non-public information about licensing arrangements with third-parties, which are protected by confidentiality clauses with those third-parties. Google does not make this information available to the public. Public disclosure of this confidential information would cause great and undue harm to Google, and place it at a competitive disadvantage.

Appx965-967, *Oracle*, No. 3:10-cv-03561-WHA, Dkt. No. 600 (N.D. Cal. Jan. 10, 2012) (highlighting added). Again, the district court below concluded that this declaration identified compelling reasons to seal the entire exhibit and, in fact, its order quoted verbatim (without attribution) the highlighted text as the basis to seal. Appx963-964, *Oracle*, No. 3:10-cv-03561-WHA, Dkt. No. 687 at 2.

So too, in *Rembrandt Patent Innovations, LLC v. Apple Inc.*, on defendant Apple's motion, the district court below ordered sealed information from plaintiff

Rembrandt including "Rembrandt's business model, and the amounts of royalty payments negotiated in licenses for the patents-in-suit." Appx989-990, No. 3:14-cv-05094 WHA, Dkt. No. 144144 at 2 (N.D. Cal. Jan. 12, 2016).

And in *Plantronics, Inc. v. Aliph, Inc.*, the district court below granted plaintiff's motion to seal an entire license agreement. Appx972-973, No. 3:09-cv-01714-WHA, Dkt. No. 295 (N.D. Cal. Dec. 5, 2013), Dkt. No. 295 at 2. It did so even though the defendant (not the moving plaintiff) was the "Designating Party" and even though defendant did not file *any* declaration in support of sealing it.

II. Standard of Review

As this appeal does not involve substantive issues of patent law, this Court applies the law of the regional circuit in which the district court sits, *i.e.*, the Ninth Circuit. *Apple v. Samsung*, 727 F.3d at 1220. When reviewing a district court's order sealing or unsealing judicial records, the Ninth Circuit reviews *de novo* whether the district court used the correct legal standard. *In re Midland Nat'l Life Ins. Co. Annuity Sales Practices Litig.*, 686 F.3d 1115, 1119 (9th Cir. 2012).

A district court's decision to seal or unseal judicial records is reviewed for abuse of discretion, provided the district court applied the correct law. *Id.* Relying upon an erroneous legal standard vitiates the abuse of discretion standard of review. *Foltz v. State Farm Mut. Auto. Ins. Co.*, 331 F.3d 1122, 1135 (9th Cir. 2003) ("*If* the district court conscientiously balances the competing interests and

articulates compelling reasons supported by specific factual findings, its decision will be reviewed only for an abuse of discretion.") (emphasis added).

A district court abuses its discretion if it "bases its decision on an erroneous legal standard or clearly erroneous findings of fact," *Earth Island Inst. v. Carlton*, 626 F.3d 462, 468 (9th Cir. 2010), or if the reviewing court "has a definite and firm conviction that the court below committed a clear error of judgment in the conclusion it reached upon a weighing of the relevant factors," *Smith v. Jackson*, 84 F.3d 1213, 1221 (9th Cir. 1996).

In sum, this Court must first determine whether the district court below followed the correct law; if not, then it is entitled to no deference. If the district court applied the correct law, then the question is whether the district court abused its discretion in its application of the law to the facts.

III. Discussion

The district court below made several mistakes of law and fact, which in turn infected its decision to unseal the remaining items of third-party information. This abrogated application of the abuse of discretion standard in these appeals.

We will begin by going through the district court's order and then turn to the compelling grounds to seal the information at hand.

A. The district court erred on the law and facts.

The district court below made several reversible mistakes of law and fact in denying the motion to seal.

To start, the district court improperly focused on the point that the third-party licensing information here at issue relates to patents. For example, the district court quoted from its own prior (reversed-in-part) order:

> [T]he public in turn has a strong interest in knowing the full extent of the terms and conditions involved in [the patentee's] exercise of its patent rights and in seeing the extent to which [the patentee's] exercise of the government grant affects commerce.

Appx31-32 (quoting Appx41). The district court did not cite a single case—other than its own—in support of this proposition. And, to close the loop, that earlier order cited nothing at all for this proposition. *See* Appx41-42.

So too, the district court below wrote:

> [A] patent is a *public* grant of rights. A patent owner is a tenant on a plot within the realm of public knowledge, and a licensee is her sub-tenant. The public has every right to account for all its tenants, all its sub-tenants, and (more broadly) anyone holding even a slice of the public grant.

Appx35 (Dkt. No. 233 at 5) (emphasis in original); *see also, e.g.*, *id.* ("[P]atent licenses carry unique considerations."); Appx925-926 ("[W]e are dealing with the public right here . . . . [¶] And ownership of that public right ought to be known."). Again, the district court cited no precedents in support of this theory. *Cf.* *Uzuegbunam v. Preczewski*, No. 19-968, ___ S. Ct. ___, 2021 WL 850106, at *6

(U.S. Mar. 8, 2021) ("The dissent's contrary assertion is unaccompanied by any citation.").

The foundation of patent law is the bargained exchange of public disclosure for temporary monopoly rights to the disclosed invention. *Cf.* U.S. Const., Art. I, Sect. 8, Cl. 8 ("The Congress shall have the Power To . . . promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."). But there is nothing in the Patent Act (or Constitution) that requires disclosure of licensing transactions related to patents. There are untold hundreds of financial transactions conducted every day with respect to patent rights and, beyond public recordation of *assignment* interests in the public register, those transactions are almost always confidential.

As a matter of public policy, it should not be the case where a patent owner elects—or is forced—to protect its rights by way of litigation, that the patent owner is thereby obliged to disclose the terms of every license it has done. Such a policy would chill the interests of both patentees and licensees in entering license agreements, thereby hampering the practice of licensing. Licensing should be encouraged, not discouraged as would happen under the district court's theory.

Whatever the merits of the district court's theory might be, such public interest would apply equally to *every single licensing case* cited herein. *See supra*

nn.5-19. Every one of those cases involved patent licenses. And yet, the licensing terms were sealed in every case. In other words, other than the district court on appeal, the other judges of the Northern District of California have uniformly found that, under Ninth Circuit law, the interest in maintaining the confidentiality of universally acknowledged trade secrets related to patent licensing outweighs the public's interest in knowing the specific amounts paid for those licenses. There has yet to be a single citation adduced to the contrary.

Moreover, in this instance, disclosure of the licensing information will not advance the district court's purported goal. None of the documents and information at issue identify the licensed patents or portfolios. Rather, they identify the licensee, amount paid, payment date and license type. Appx732-734. So, the public will be no more informed of what entity has a license to what patent(s) than before.

In any event, after identifying its theory, the district court turned to "the crux of the problem," which it described as "whether the *particular bases* offered by our third parties outweigh the presumption of public access." Appx32 (emphasis in original). Focusing first on the law, the district court acknowledged that Uniloc cited many cases in which this sort of information had been sealed, and for which not a single *contra* citation has ever been found. But, the district court concluded that those other courts had all simply and uniformly been wrong. This is not an

exaggeration. Indeed, the judge had to state that he too had been consistently wrong on the subject:

> Uniloc's reliance on the numerous district court orders which have sealed similar information underscores the larger problem of indiscriminate oversealing in patent and commercial cases nationwide. This order readily acknowledges that this Court, among the many Uniloc cites, has at times fallen into the habit, as a concession to the shortness of life, of oversealing information that in truth should be made public. But the culprits seem plain. Our adversarial system collapses when, as often occurs in these suits, both parties seek to seal more information than they have any right to and so do not police each other's indiscretion. Perhaps no case better illustrates this collusive oversealing so much as the one upon which Uniloc continually relies, *Apple v. Samsung*. There, *both* Apple and Samsung bombarded the Honorable Lucy H. Koh with excessive and *unopposed* sealing motions, took full advantage of the judge's patience in permitting them to revise and narrow their motions, and then appealed the partial denial of their sealing motions to the Federal Circuit, where they again refused to oppose each other's motions, leaving that task to amici curiae. 727 F.3d 1218-20.
>
> True to form, given the opportunity to oppose Uniloc's oversealing at oral argument here, Apple abandoned the Court in its enforcement of the local rules and circuit precedent, further confirmed that it would not oppose the sealing on appeal, and opted instead to grab its December 4 victory on the standing issue and head for the hills. This is why EFF must be permitted to intervene.

Appx33 (emphasis in original). Yet again, the district court did not cite any other cases in support of overturning this uniform precedent.

The above bears repeating: The district court below asserted that it and its fellow judges of the Northern District of California have consistently sealed this exact sort of information in the past. The Ninth Circuit and this Court too have

sealed this sort of information. *See, e.g.*, *In re Elec. Arts*, 298 Fed. App'x. at 569-70; *Apple v. Samsung*, 727 F.3d at 1228 (applying Ninth Circuit law). But the district court said that these courts were all wrong to have done so because patent licenses are special. Yet, the district court could not cite a single precedent—at any level—in support of this novel position, other than its own prior ruling which this Court had vacated with respect to this exact issue. With all due respect, the district court below could not overrule the Ninth Circuit. The fact that there is a uniform practice on one side, with only one example on the other, suggests that the district court made a mistake of law. *E.g.*, *Earth Island Inst.*, 626 F.3d at 468 (9th Cir. 2010) (explaining that a district court abuses its discretion if it "bases its decision on an erroneous legal standard or clearly erroneous findings of fact").

Focusing next on the evidence presented, the district court acknowledged that twenty-three licensees explicitly asked the court to keep their information under seal, and that they provided reasons for their requests by way of Uniloc's declaration. But the district court discarded this evidence because "[a]ll of this is hearsay." Appx33 (Dkt. No. 233 at 4). The district court did not explain why a third-party must submit a separate declaration in support; there is no such requirement in the local rules.[20] The district court also rejected, *inter alia*, the

[20] Local Rule 79-5(e) requires that the "Designating Party" file a declaration. Uniloc—which was the Designating Party for the document containing the list of

thirteen (not hearsay) declarations filed by third-parties. Appx33-34. It rejected this evidence—thousands upon thousands of words explaining the interests of the third-parties—because the district court was bothered by the absence of any additional evidence submitted *after* remand from this Court:

> Given the Federal Circuit's public remand to more carefully consider their interests, the third parties' silence speaks volumes. We are left to balance the public interest against either stale declarations from nearly two years ago (for those eight [sic.[21]] who provided them) or Uniloc's hearsay.

Appx34. Appellants submit that this was a clear misreading of this Court's directions on remand.

Again, in its July 2020 opinion, this Court recognized that "many of Uniloc's licensees *have submitted declarations* stating that they wish their licensing information to remain confidential and that the disclosure of such information would cause them material competitive injury," Appx503 (emphasis

---

109 licenses—did so, and therein included the pleas of the third-party licensees. *See* Appx765-772 ¶¶ 8-9.w.i.

The undersigned counsel suggested to the licensees that they file their own declarations. Thirteen did so. But many others declined, based upon the very fear that was realized in this case—in addition to their licensing information, the district court might put their declarations into the public record too. The district court's orders will likely chill future third-parties from coming forward to protect their compelling interests.

[21] The district court miscounted the third-party declarations in support of the motions to seal. There were eight filed under seal, Appx805-837, and another five filed in the public record, Appx436-450.

added), and further that other third-parties had asked Uniloc to relay their requests and concerns. So, this Court wrote:

> As to these third-party materials, we conclude that the district court failed to make findings sufficient to allow us to adequately assess whether it properly balanced the public's right of access against the interests of the third parties in shielding their financial and licensing information from public view.

Id. In short, this Court recognized that the district court failed to consider the wealth of information already before it and remanded for consideration of those materials. Yet, the district court denigrated Uniloc's attempt to argue on behalf of those third-parties and the evidence they already submitted because nothing new was added to the already voluminous record. This was a mistake of law.

The district court next held that it was "[c]onclusive" that "the dates and dollar amounts involved in Uniloc's patent licenses 'go to the heart of' the primary dispute, that of Uniloc's standing (or lack thereof) to sue." Appx34. But that is demonstrably untrue. There was no relevance whatsoever to *who* paid Uniloc licensing fees in the past. While the total amount paid was relevant to whether there was a breach of the creditor agreement with Fortress, the *individual payments* was not at issue.[22] There was no "dispute" about the date and dollar amount under

[22] As the district court previously recognized, "Apple's motion to dismiss for lack of standing did not directly depend upon information regarding the specific dollar amounts, financial terms, and names of the licensees in the various agreements (with Fortress or third-party licensees)." Appx42. Instead, it was the aggregate of them all that mattered.

the agreement. As the district court stated in its order dismissing the -358 case: "Our facts are uncontested." Appx897. This is because no one disputed that the licensing target was $20,000,000. No one disputed that the deadline was March 31, 2017. And no one disputed that by March 31, 2017, Uniloc only gathered about $14,000,000 in aggregate revenues. *See, e.g.*, Appx892 & Appx897. By the district court's own account, all that remained in dispute was a question of law: In light of the uncontested facts, did Uniloc lack standing to enforce its patents?

The district court then turned to the materials at issue. Based upon the foregoing, it concluded that the table of licensing information would be unsealed, as would be all references to any of those licenses in the deposition transcripts and briefing. Appx35. The compelling reasons to seal these documents will be discussed *infra* at Argument § III.B.

Lastly, the district court considered the Fortress Memorandum, submitted in the -358 case. The district court concluded that it should be unsealed because "Fortress has not submitted a declaration in support of its sealing request. Instead, Uniloc filed the hearsay declaration here, merely reporting what Fortress's counsel apparently said (No. C 18-00358, Dkt. No. 173 at ¶¶ 3, 19-22)." Appx35. This was a mistake of fact, as is evident from the cited paragraphs; the declaration was expressly submitted on behalf of Uniloc *and Fortress*. The compelling reason to seal this document will be discussed *infra* at Argument § III.C.

In sum, the district court made mistakes of law—and thereby vitiated the abuse of discretion standard—by (1) relying upon a new theory that patent licenses hold a special place in the firmament of filing materials under seal; (2) ignoring the dictates of the Ninth Circuit (and this Court's application of Ninth Circuit law); (3) disregarding the uniform holdings of the other judges of the Northern District of California; (4) contravening its own prior holdings on the subject; (5) asserting that all of those prior decisions were wrong; (6) announcing a new requirement for third-parties seeking to keep their information confidential, to wit, that they must individually file declarations close in time to the court's consideration; (7) ignoring this Court's directions on remand; (8) ignoring (again) the evidence presented by the more than 100 third-party licensees; (9) bypassing its own statement that the facts were uncontested; and (10) misreading the declaration in support of Fortress. Any one of these should warrant reversal, in part or whole. The correct outcome was reached by Judge Gonzalez Rogers—also of the Northern District of California—just five hours later, when she ordered sealed the same list of 109 licensees: "Pricing terms and confidential financial information are routinely sealed as materials that may be used to harass or harm a party's competitive standing." *Uniloc v. Google*, 2020 WL 7626430, at *21 n.23.

* * *

We will now turn to the individual documents and information to be sealed, starting with the third-party licenses and related information, and then the Fortress Memorandum.

B. There are compelling reasons to redact the references to third-party licensees and their licensing information.

1. The Conformed Revenue Sharing and Note and Warrant Purchase Agreement should remain redacted.

The Conformed Revenue Sharing and Note and Warrant Purchase Agreement—submitted as Exhibit A to Apple's motion to dismiss in the -360 case—relates to the financial relationship between non-party Fortress and Uniloc. Appx708-734. The first twenty-two pages have been made public, so the only remaining issue relates to the last three pages. Appx732-734. Those pages include a table listing 109 licenses. Each row discloses the third-party licensee's name, the date of the license and the amount paid for the license. Each such set of information is covered by a separate agreement. Most of these agreements include express confidentiality provisions. And, these confidentiality provisions are in almost every instance founded upon court-issued protective orders.

Disclosure of these three pages would make public the confidential financial and business information of more than 100 third-parties. In the cases cited herein, *see supra* nn.5-19, the to-be-sealed information almost always related to just one

third-party. So, there will be more third-party licensing information made public through this one event than was at issue in all of the other cited cases combined.

Following the district court's original refusal to seal this licensing information, Uniloc's counsel reached out to the licensees to ask for their positions as to publication of this information. Appx764 ¶ 5. Just two of the more than 100 agreed to the disclosure of their information. Appx764-765 ¶¶ 7-7.b. Eight others agreed to disclose their identities, but asked to maintain the confidentiality of their license payments; they described their reasons in declarations submitted by them or by Uniloc with their explicit requests. Appx436-450 (third-party declarations); Appx765-767 ¶¶ 8-8.h.

Twenty-three licensees asked that *all* information about them remain confidential; they described their reasons in the accompanying declarations submitted by them or by Uniloc with their explicit requests. Appx805-837 (sealed third-party declarations); Appx767-772 ¶¶ 9-9.w.i. These entities explained that confidentiality, including of their identities, was important to the licensing negotiations. *See Huawei Techs., Co, Ltd v. Samsung Elecs. Co, Ltd.*, 340 F. Supp. 3d 934, 1004 (N.D. Cal. 2018) ("Percentages of royalties sought or secured in negotiations or resulting licensing agreements may remain under seal at this juncture, if those terms are not otherwise publicly known. *This includes references to the identities of third-parties to those agreements, assuming the existence of the*

agreement itself is not otherwise publicly known.") (emphasis added). They further asserted that the disclosure of their identities and the existence and terms of the licenses would cause competitive harm.

As all the third-party licensees explained, they go to great lengths to maintain the confidentiality of their licensing information. For, among other reasons, disclosure of their licensing information would be used by other patentees in license negotiations with those third-parties. This information asymmetry would put these third-parties at a permanent disadvantage. Appx436-450; Appx805-837; Appx765-772 ¶¶ 8-9.w.i; *see, e.g.*, *Finjan, Inc. v. Sophos, Inc.*, No. 14-cv-01197-WHO, 2016 WL 7911365, at *1 (N.D. Cal. Aug. 30, 2016) (granting patentee's request to protect the identity of its licensees during trial by using codenames when referring to the licensee and redacting the licensee's name from the license agreements submitted as evidence in the trial).

Uniloc is obliged to similarly protect the information of any entity that did not respond or whose response was not sufficiently definite. Appx764 ¶ 6. That some licensees did not respond does not mean they do not care, nor does it vitiate Uniloc's contractual obligations to abide by those third-parties' previously expressed wishes (and court-ordered protective orders). Rather, it could be that the licensees moved from the addresses identified in the agreements; or, the in-house

or outside counsel no longer receive email at the listed addresses; or, any number of other alternatives. Silence cannot be taken as assent for disclosure.

Further, the identity of any given third-party and the terms of its individual license agreement are not relevant to the issues considered by the district court, so the public's interest in this information is substantially outweighed by the third-party's interest in maintaining its confidentiality. *See Foltz*, 331 F.3d at 1135. Whether the proverbial Blackacre Corporation paid Uniloc a peppercorn or $10,000,000 was irrelevant to Apple's motion. The issue behind Apple's motion was whether, *in the aggregate*, Uniloc's licenses hit a certain threshold. Appx42 ("Apple's motion to dismiss for lack of standing did not directly depend upon information regarding the specific dollar amounts, financial terms, and names of the licensees in the various agreements (with Fortress or third-party licensees) . . . ."). And that issue was not in dispute.

Nonetheless, the district court ordered this information unsealed.

If this Court will forgive the rhetorical questions, how could the district court below hold that the declarations it considered in *Oracle*, *Finjan v. Juniper Network* and *Rembrandt* were sufficient, *see supra* at 34-36, but the declarations in this case were not? And how could the district court below conclude that the license in *Plantronics* should be filed under seal in its entirety—even without a declaration in support, *see supra* at 36—but the third-parties' requests here were

insufficient? The district court gave its answer: The district court concluded that it and every other court must have been wrong. Appx33.[23] If a district court cannot cite a single appellate case in support of its conclusion that it and every other district court's uniform practice to date has been wrong, then that district court must have made a mistake of law.

To the extent there is any question of whether the district court abused its discretion, Uniloc asks this Court to compare the declarations that were found sufficient—by the same district court—in *Finjan v. Juniper Network*, Appx1038-1041, and in *Oracle*, Appx965-967, with the declarations submitted in this case that were deemed lacking, Appx436-450 (unsealed third-party declarations); Appx805-833 (sealed third-party declarations); Appx766-772 ¶¶ 8-10 (Uniloc declaration). Sealing the financial terms in the earlier cases, but not these cases, is not an exercise in discretion; it is an abrogation of precedent and practice.[24]

---

[23] The district court seemed to suggest that this Court came to the wrong conclusion in *Apple v. Samsung*, 727 F.3d 1214. *See* Appx33.

[24] Again, in *Finjan v. Juniper Network*, the same district court sealed the "confidential license agreement between Juniper and a third-party, which is information that falls within the definition of 'trade secret,'" with far less evidence. Appx1042-1044, *Finjan v. Juniper Network*, No. 3:17-cv-05659-WHA, Dkt. No. 485 at 2. In *Oracle*, the same district court sealed "non-public information about licensing arrangements with third-parties, which are protected by confidentiality clauses with those third-parties," with far less evidence. Appx963-964, *Oracle*, No. 3:10-cv-03561-WHA, Dkt. No. 687 at 2. In *Rembrandt*, the same district court sealed "the amounts of royalty payments negotiated in licenses," with far less evidence. Appx989-990, No. 3:14-cv-05094 WHA, Dkt. No. 144 at 2 (N.D. Cal.

In light of the thirteen third-party declarations and the more than 3500 words of the Uniloc declaration which address this document in particular, Appellants asks the Court to recognize that the third-parties and Uniloc identified more than simple, "generalized assertions of potential competitive harm." Appx43; *see, e.g.*, *Apple v. Samsung* (N.D. Cal. November Order), 2012 WL 5988570, at *4 ("The Ninth Circuit has held, and [the Northern District of California] has previously ruled, that pricing terms, royalty rates, and minimum payment terms of licensing *agreements plainly constitute trade secrets* and thus are sealable.") (emphasis added). Rather, as the district court below wrote in *Oracle*, these pages "contain[] non-public information about licensing arrangements with third-parties, which are protected by confidentiality clauses with those third-parties. [The third-parties] do[] not make this information available to the public. Public disclosure of this confidential information would cause great and undue harm to [those third-parties], and place [them] at a competitive disadvantage." Appx963-964, *Oracle*, No. 3:10-cv-03561-WHA, Dkt. No. 687 at 2; *cf. Microsoft Corp. v. Hon Hai Precision Indus. Co.*, No. 5:19-cv-01279-LHK, 2020 WL 4901610, at *3 (N.D. Cal. Aug. 20, Jan. 12, 2016). And, in *Plantronics*, the same district court sealed an entire license agreement with *absolutely no evidence*.

Even if the district court were wrong in each of those earlier cases, the third-party licensees were at least entitled to rely upon those prior, consistent precedents at the time they submitted their evidence and requests in the cases on appeal.

2020) ("Exhibit 3 is a small, one-page excerpt from Microsoft's damages expert's supplemental expert report. Exhibit 3 contains a single table replete with confidential trade secret information concerning royalty rates and pricing terms. As a result, the Court holds that Exhibit 3 is sealable in its entirety."). This constitutes compelling reasons shown and so Appellants respectfully request that this Court overrule the district court and order that these lines remain redacted.

2. The excerpts of the Settlement and License Agreement between Microsoft and Uniloc should remain redacted.

Apple filed a two-page excerpt from the confidential settlement and license agreement between third-party Microsoft and Uniloc. Appx759-760. The document is redacted to cover only the license fees.

Even EFF did not ask that these redactions be lifted. Nonetheless, the district court ordered it unsealed.

As detailed in the Microsoft and Uniloc declarations, terms of the license agreement constitute Microsoft's trade secret information, disclosure of which would cause competitive harm to Microsoft. Appx449-450; Appx766 ¶ 8.e. Disclosure of this information would lead to information asymmetry between third-party Microsoft and potential licensors, indelibly damaging Microsoft. *See, e.g.*, Appx963-964, *Oracle*, No. 3:10-cv-03561-WHA, Dkt. No. 687 at 2. This information is also covered by a Protective Order from another court. Appx941-

962 *Uniloc USA, Inc. v. Microsoft Corp*., No. 03-cv-440 (WES), Dkt. No. 23 (D.R.I. Apr. 30, 2004). These constitute compelling reasons shown to redact the identified terms.

As such, Appellants respectfully request that the Court overrule the district court and order that these few words remain redacted. *See, e.g.*, *Apple v. Samsung* (N.D. Cal. November Order), 2012 WL 5988570, at *4 ("The Ninth Circuit has held, and [the Northern District of California] has previously ruled, that pricing terms, royalty rates, and minimum payment terms of licensing agreements plainly constitute trade secrets and thus are sealable.").

3. Apple's Reply Brief should remain redacted.

After remand from this Court, Uniloc proposed to redact only a few phrases in Apple's reply brief, specifically on page 10 regarding Uniloc's licenses with Microsoft. Appx 853. The to-be-redacted information does not disclose the exact dollar figures in the Uniloc-Microsoft licenses. But, in conjunction with other information that has been made public, the disclosure of these lines would effectively disclose the exact dollar figures of those licenses.

Even EFF did not seek the disclosure of this information. Nonetheless, the district court ordered it unsealed.

For the reasons discussed immediately above with respect to the Uniloc-Microsoft agreement, *see supra* Argument § III.B.2, Appellants respectfully

request that the Court overrule the district court and order that these few words remain redacted.

4. The Jacobs Revised Redactions Declaration should remain redacted.

The Jacobs Revised Redactions Declaration was submitted with Uniloc's motion for reconsideration. Appx761-776. The portions that Uniloc sought to redact are those that relate solely to information from third-parties who expressly asked that their information and requests remain confidential. In short, the to-be-redacted sections are the pleas and explanations from twenty-three third-party licensees, asking the district court to seal their licensing information.

Even EFF did not seek the disclosure of the redacted information. Nonetheless, the district court ordered it unsealed.

As detailed in the declarations, this information constitutes trade secrets of those third-party licensees, the disclosure of which would cause competitive harm to them. Appx764-772 ¶¶ 4, 8-11; Appx436-450 (unsealed third-party declarations); Appx805-833 (sealed third-party declarations). Disclosure of this information would lead to information asymmetry between those third-parties and potential licensors. *See, e.g.*, Appx963-964, *Oracle*, No. 3:10-cv-03561-WHA, Dkt. No. 687 at 2. This information is also covered by Protective Orders from other courts. And, refusing to seal this information would likely result in a chilling effect on future third-parties coming forward to advocate for their interests.

Unsealing this information will provide no benefit at all to anyone and only harm those third-parties that petitioned the district court for its assistance. These constitute compelling reasons shown to redact the identified information. As such, Appellants respectfully request that the Court overrule the district court and order this declaration remain redacted. *Apple v. Samsung* (N.D. Cal. November Order), 2012 WL 5988570, at *4.

5. The sealed declarations should remain under seal.

Exhibits G through N of Uniloc's motion for reconsideration were sealed and redacted declarations from third-party licensees. Appx805-833 (sealed third-party declarations).

Even EFF did not seek the disclosure of these declarations. Nonetheless, the district court ordered them unsealed.

In each declaration, an executive from the given licensee explained in detail the damage that would be caused by publication of the information that relates to it. This information constitutes trade secrets of those third-parties, the disclosure of which would cause competitive harm to those third-parties. Disclosure of this information would lead to information asymmetry between the third-parties and potential licensors. *See, e.g.*, Appx963-964, *Oracle*, No. 3:10-cv-03561-WHA, Dkt. No. 687 at 2. This information is also covered by Protective Orders from other courts. And refusing to seal this information would likely result in a chilling

effect on future third-parties coming forward to protect their interests. Unsealing it will provide no benefit at all to anyone and only harm those third-parties that petitioned the district court for its assistance. These constitute compelling reasons shown to seal and redact these documents. As such, Appellants respectfully request that the Court overrule the district court and order these declarations sealed and redacted. *Apple v. Samsung* (N.D. Cal. November Order), 2012 WL 5988570, at *4.

6. The Palmer deposition excerpts should remain redacted.

Apple's motion to dismiss in the -358 case was accompanied by sixty-three pages of transcript from the deposition of James Palmer, a Managing Director at Fortress Investment Group.[25] Appx549-612. Uniloc sought to seal just a few words across three lines, which disclose the identity of third-party licensees. Appx555. For these reasons and those discussed above, *see supra* Argument § III.B.1, Appellants respectfully request that this Court overrule the district court and order these three lines redacted.

[25] Fortress Investment Group is an affiliate of Fortress Credit Co. LLC, *i.e.*, "Fortress."

C. There are compelling reasons to seal the Fortress Memorandum and redact Apple's Motion to Dismiss in the -358 case.

The Fortress Memorandum is a detailed analysis of Uniloc created by non-party Fortress, Appx613-616, which Apple submitted as an exhibit with its motion to dismiss the -358 case. That motion discussed the Fortress Memorandum at page 15, from lines 13 to 17. Appx538. Fortress and Uniloc sought to seal the Memorandum and the four lines of excerpts regarding Fortress in Apple's brief.

Starting with the first two pages of the Fortress Memorandum, the document summarizes Fortress's prior interactions and business dealings with Uniloc; analyzes Uniloc in depth; and proposes additional investments in Uniloc, based upon Fortress's internal, proprietary analyses of Uniloc. Appx614-615. This information has not been shared outside of Fortress. Fortress's proposal includes terms which are not publicly available and which are considered proprietary to Fortress. Disclosure of the information contained in the Fortress Memorandum would harm Fortress's ability to negotiate and further deal with Uniloc—which has not seen the information—as well as other third-parties with whom Fortress might seek to deal. Fortress considers its investment criteria among the most valuable—and thus confidential—information available to it. Appx623-624 ¶¶ 19-23; *see, e.g.*, *In re Koninklijke*, 2020 WL 1865294, at *2 (sealing "information related to business operations"); Appx989-990, *Rembrandt*, No. 14-cv-05094-WHA, Dkt.

No. 144 at 2 (N.D. Cal. Jan. 12, 2016) (sealing "business model" information); *Arista Networks*, 2018 WL 2010622, at *2-3 (sealing "highly confidential and sensitive information relating to Cisco's financial information and internal development strategies," "highly confidential and sensitive information relating to Arista's financial and customer information," and "confidential settlement terms between Cisco and third-party, Huawei Technologies"); *Juicero*, 2017 WL 8294276, at *2 (sealing "confidential financial and business information"); *Transperfect Global*, 2014 WL 4950082, at *1 (sealing "confidential financial and marketing information").

Moreover, disclosure of this information would not aid the public. Whatever personal interest one might have in reading non-party Fortress's analyses of its existing and potential investments, these particulars will play no part in verifying the propriety of the district court's order. *Center for Auto Safety*, 809 F.3d at 1096 ("The presumption of access is 'based on the need for federal courts, although independent—indeed, particularly because they are independent—to have a measure of accountability and for the public to have confidence in the administration of justice.'"). Unsurprisingly, the district court did not cite the Fortress Memorandum its order dismissing the case. *See* Appx891-904. As such, the public's interest in it is *de minimis*. *Cf. Kamakana*, 447 F.3d at 1179.

The third page of the Fortress Memorandum includes a list of fifty-five third-party licenses taken from the larger list of 109 licenses in the Conformed Revenue Sharing and Note and Warrant Purchase Agreement. For the reasons discussed above with respect to the larger list, this licensing information should remain under seal. *See supra* Argument § III.B.1.

Finally, the district court was mistaken when it stated that the document should not be sealed because Fortress did not submit a declaration in support. As the district court recognized, a declaration was submitted by (at least) counsel for Uniloc. Appx35. However, Prince Lobel Tye LLP, counsel for Uniloc, also represents Fortress with respect to production of the Fortress Memorandum in these cases. As such, the declarant spoke with an individual at Fortress and another attorney at the firm regarding the Fortress Memorandum in preparing the declaration. And, on this basis, and *on behalf of Fortress*, that declarant identified the confidential information, the harm that could come from disclosure of it, and requested *on behalf of Fortress* that the court seal it. Appx619-625 ¶¶ 3, 19-23.

These constitute compelling reasons shown to seal the Fortress Memorandum. As such, Appellants and Fortress respectfully request that the Court overrule the district court and order these declarations sealed and redacted.

CONCLUSION

For the foregoing reasons, this Court should reverse the district court's order and remand with instructions to redact and seal the documents as proposed in Uniloc's motion.

April 2, 2021

Respectfully submitted,

UNILCO USA, INC. AND UNILOC LUXEMBOURG, S.A., by their attorneys,

/s/ Aaron S. Jacobs

Aaron S. Jacobs
Prince Lobel Tye LLP
One International Place, Suite 3700
Boston, MA 02110
(617) 456-8000
ajacobs@princelobel.com

Counsel for Plaintiffs-Appellees

CERTIFICATE OF SERVICE

I hereby certify under penalty of perjury that on this 2d day of April, 2021, a copy of the foregoing document was filed electronically.

This filing was served electronically to all parties by operation of the Court's electronic filing system.

/s/ Aaron S. Jacobs
Aaron S. Jacobs

CERTIFICATE OF COMPLIANCE

This brief complies with the type-volume limitation of Federal Circuit Rule 32(b). This brief contains 13,828 words, excluding the parts of the brief exempted by Federal Rule of Appellate Procedure 32(f) and Federal Circuit Rule 32(b)(2).

This brief complies with the typeface requirements of Federal Rule of Appellate Procedure 32(a)(5) and the type style requirements of Federal Rule of Appellate Procedure 32(a)(6). This brief has been prepared in proportionally spaced typeface using Microsoft Office Professional Plus 2013 in 14-point Times New Roman type style.

April 2, 2021 */s/ Aaron S. Jacobs*

Aaron S. Jacobs

ATTACHMENT A

Docket Item	Description	Proposed Treatment
-360 case, 222-3 & -360 case, 222-11	Winnard Decl. Exhibit A: Conformed Revenue Sharing and Note and Warrant Purchase Agreement. Jacobs Decl. re Revised Redactions Ex. B: Revised redactions for Conformed Revenue Sharing and Note and Warrant Purchase Agreement	Redactions to the third-party licensing information from UNILOC_APPLE_2017_18337 to UNILOC_APPLE_2017_18339. *See* Argument § III.B.1.
-360 case, 222-5 & -360 case, 222-29	Apple's Reply Jacobs Decl. re Revised Redactions Ex. U: Revised redactions for Apple Reply in Support of Motion to Dismiss	Redactions to the dollar values and percentages related to the Microsoft agreements, on page 10 at lines 15, 16, 17, 24 and 25, which shall remain redacted. *See* Argument § III.B.3.
-360 case, 222-7 & -360 case, 222-25	Winnard Reply Decl. Exhibit DD: Settlement and License Agreement between Microsoft and Uniloc Jacobs Decl. re Revised Redactions Ex. W: [Sealed] Settlement and License Agreement between Microsoft Corporation and Uniloc	Redactions to cover the financial terms of this agreement with Microsoft. *See* Argument § III.B.2.
-360 case, 222-9	Decl. of Aaron S. Jacobs ("Jacobs Revised Redactions Decl.") in Support of Plaintiff's Motion for Reconsideration of the Court's January 17, 2019,	Redactions to third-party information from page 5, line 16, through page 10, line 8. *See* Argument § III.B.4

Docket Item	Description	Proposed Treatment
	Order re Sealing of Order on Motion to Dismiss and Motion to Join Party, and Order re Administrative Motions to File Under Seal and Motion to Intervene	
-360 case, 222-13	Jacobs Decl. re Revised Redactions Ex. G: [Sealed] Decl.	Third-party information to remain sealed. *See* Argument § III.B.5.
-360 case, 222-15	Jacobs Decl. re Revised Redactions Ex. H: [Sealed] Decl.	Third-party information to remain sealed. *See* Argument § III.B.5.
-360 case, 222-17	Jacobs Decl. re Revised Redactions Ex. I: [Redacted] Decl.	Third-party information to remain redacted. *See* Argument § III.B.5.
-360 case, 222-19	Jacobs Decl. re Revised Redactions Ex. J: [Sealed] Decl.	Third-party information to remain sealed. *See* Argument § III.B.5.
-360 case, 222-21	Jacobs Decl. re Revised Redactions Ex. K: [Sealed] Decl.	Third-party information to remain sealed. *See* Argument § III.B.5.
-360 case, 222-23	Jacobs Decl. re Revised Redactions Ex. L: [Sealed] Decl.	Third-party information to remain sealed. *See* Argument § III.B.5.
-360 case, 222-25	Jacobs Decl. re Revised Redactions Ex. M: [Sealed] Decl.	Third-party information to remain sealed. *See* Argument § III.B.5.
-360 case, 222-27	Jacobs Decl. re Revised Redactions Ex. N: [Sealed] Decl.	Third-party information to remain sealed. *See* Argument § III.B.5.
-358 case, No. 165	Apple's Motion to Dismiss	Third-party information to remain redacted at 15:13-17. *See* Argument § III.C.
-358 case, No. 165-2	Winnard Decl. Ex. A: Palmer Deposition Excerpts	Third-party information to remain redacted at 119:14-16. *See* Argument § III.B.6.
-358 case, No. 165-10	Fortress Memorandum	Third-party information to remain sealed. *See* Argument § III.C.

ADDENDUM

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

UNILOC USA, INC., et al.,

Plaintiffs,

v.

APPLE, INC.,

Defendant.

No. C 18-00358 WHA
No. C 18-00360 WHA
No. C 18-00363 WHA
No. C 18-00365 WHA
No. C 18-00572 WHA

ORDER RE SEALING

INTRODUCTION

This order addresses the sealing of evidence submitted in a patent infringement suit. Accepting that several courts of appeal have held certain licensing and financial records sealable at times, on the record provided, the sealing motions are **DENIED**.

STATEMENT

Previous orders detail the background (No. C 18-00358, Dkt. No. 186, No. C 18-00360, Dkt. No. 157). In brief, these patent infringement suits have been funded by an entity called Fortress Credit Co. LLC, which took a broad license in the asserted patents as security and imposed annual monetization goals on plaintiff Uniloc. Accused-infringer Apple later discovered that when Uniloc filed these cases, it had failed to meet its monetization goals for the preceding twelve months, which released the sole condition on Fortress's license, letting it sub-license the asserted patents to the world. After several rounds of briefing and decision, and

a remand by the Court of Appeals for the Federal Circuit to take a fresh look, a December 4 order concluded that Uniloc indeed lacked standing to sue here (No. C 18-00358, Dkt. No. 186).

Several sets of sealing motions now converge here because of the piecemeal manner in which Uniloc's lack of standing has surfaced across these cases. We address both Uniloc's request to seal portions of Apple's most recent motion to dismiss and its requests to seal similar material in the related cases, Nos. C 18-00360 *et seq.*, where Apple first raised the standing issue. The Federal Circuit largely affirmed denial of those sealing requests, but has remanded for our reconsideration of third party confidentiality interests. *Uniloc 2017 LLC v. Apple, Inc.*, 964 F.3d 1351, 1364 (Fed. Cir. 2020). The Electronic Frontier Foundation again moves to intervene in opposition to Uniloc's sealing requests. This order follows full briefing and a hearing (held telephonically due to COVID-19).

ANALYSIS

Federal courts are public tribunals, not private mediators. They belong to the people, not to corporate America. The public has legitimate interests in looking over our shoulders to see why and how we grant relief (or not) and specifically what record justified (or not) that relief. So our court of appeals has recognized a strong presumption of public access to our records. On a dispositive motion, sealing any part of a record requires, without "hypothesis or conjecture," a compelling factual basis which *outweighs the public interest in disclosure*. *Ctr. for Auto Safety v. Chrysler Grp.*, 809 F.3d 1092, 1096–97 (9th Cir. 2016); *Kamakana v. City & County of Honolulu*, 447 F.3d 1172, 1178–79 (9th Cir. 2006).

In our present case, a *second* public interest also favors access. The United States Supreme Court "has long recognized that the grant of a patent is a matte[r] involving public rights." A patent is not a private agreement between private parties. Rather, as a creature of statute, *the national government grants* the patent in derogation of the usual free flow of goods and ideas. *See Oil States Energy Servs. v. Greene's Energy Grp.*, 584 U.S. ___, 138 S. Ct. 1365, 1373 (2018) (quotation marks omitted). Thus, the undersigned has recognized that:

> Because [a patentee's] rights flow directly from this government-conferred power to exclude, the public in turn has a strong interest in knowing the full extent of the terms and conditions involved in

> [the patentee's] exercise of its patent rights and in seeing the extent to which [the patentee's] exercise of the government grant affects commerce.
>
> The impact of a patent on commerce is an important consideration of public interest. One consideration is the issue of marking by licensees. Another is recognition of the validity (or not) of the inventions. Another is in setting a reasonable royalty. In the latter context, patent holders tend to demand in litigation a vastly bloated figure in "reasonably royalties" compared to what they have earned in actual licenses of the same or comparable patents. There is a public need to police this litigation gimmick via more public access. *We should never forget that every license has force and effect only because, in the first place, a patent constitutes a public grant of exclusive rights*.

(No. C 18-00360, Dkt. No. 187) (emphasis added).

We turn first to the Federal Circuit's task on remand, to reconsider the "sealing or redaction of the purportedly confidential information of third parties" and to "make particularized determinations as to whether and, if so, to what extent, the materials of each of these parties should be made public." *Uniloc*, 964 F.3d at 1364. Uniloc seeks to seal the details of its patent licensing agreements, taking the form of: (i) tables containing the names of Uniloc's licensees and the dates and dollar amounts of the deals; (ii) brief references to data from that table in an Apple brief; (iii) a licensing agreement between Uniloc and Microsoft; (iv) eight declarations from various licensees requesting the Court keep their license details under seal; and (v) a declaration summarizing similar requests from many licensees.

Uniloc argues that judges, including many in this district, routinely find patent licensing data sealable. Our court of appeals has found a videogame publisher's licensing agreements with a professional athletes' union sealable; and the Federal Circuit has found profit and cost data for patented and infringing goods sealable. But the crux of the problem is not just whether the information falls within categories of sealable information. It is, rather, whether the *particular bases* offered by our third parties outweigh the presumption of public access. If these bases exist here, they come not from Uniloc, but from the third parties. After all, the Federal Circuit affirmed the rejection of Uniloc's requests for sealing and remanded solely for this Court to consider third party interests. *See Uniloc*, 964 F.3d at 1364; *Chrysler*, 809 F.3d at 1096–97; *see, e.g.*, *Apple v. Samsung*, 727 F.3d 1214, 1225–26 (Fed. Cir. 2013); *In re Elect.*

Arts, 298 Fed. App'x 568 (9th Cir. 2008); *Parrish v. Nat'l Football League Players Ass'n*, No. C 07-0943 WHA, Dkt. No. 498 (N.D. Cal. Oct. 21, 2008) (EA's motion to seal).

Moreover, Uniloc's reliance on the numerous district court orders which have sealed similar information underscores the larger problem of indiscriminate oversealing in patent and commercial cases nationwide. This order readily acknowledges that this Court, among the many Uniloc cites, has at times fallen into the habit, as a concession to the shortness of life, of oversealing information that in truth should be made public. But the culprits seem plain. Our adversarial system collapses when, as often occurs in these suits, *both parties* seek to seal more information than they have any right to and so do not police each other's indiscretion. Perhaps no case better illustrates this collusive oversealing so much as the one upon which Uniloc continually relies, *Apple v. Samsung*. There, *both* Apple and Samsung bombarded the Honorable Lucy H. Koh with excessive and *unopposed* sealing motions, took full advantage of the judge's patience in permitting them to revise and narrow their motions, and then appealed the partial denial of their sealing motions to the Federal Circuit, where they again refused to oppose each other's motions, leaving that task to amici curiae. 727 F.3d 1218–20.

True to form, given the opportunity to oppose Uniloc's oversealing at oral argument here, Apple abandoned the Court in its enforcement of the local rules and circuit precedent, further confirmed that it would not oppose the sealing on appeal, and opted instead to grab its December 4 victory on the standing issue and head for the hills. This is why EFF must be permitted to intervene. Without EFF, the public's right of access will have no advocate. Our court of appeals has permitted similar permissive intervention by parties seeking record access. *See Beckman Indus. v. Int'l Ins. Co.*, 966 F.2d 470, 473–74 (9th Cir. 1992). EFF's timely motion to intervene is thus granted.

Uniloc solicited the views of all one hundred nine licensees regarding the sealing of their patent license details. It reports that two agreed to disclosure, eight offered to disclose their identities but asked to keep the remaining details under seal, and twenty three asked to keep all information under seal. All of this is hearsay, and again, moreover, Uniloc's argument *on behalf* of third parties rings hollow. Uniloc's own interests appear sprinkled throughout its

argument and, unlike Electronic Arts, who appeared and argued its own confidentiality interests before both the district court and court of appeals, no third party has, despite notice, filed any request or statement on our docket in connection with the instant sealing motions. *Contra In re Elect. Arts*, 298 Fed. App'x 568; *see Parrish*, Dkt. No. 498. Given the Federal Circuit's public remand to more carefully consider their interests, the third parties' silence speaks volumes. We are left to balance the public interest against either stale declarations from nearly two years ago (for those eight who provided them) or Uniloc's hearsay.

That said, the theme among Uniloc's third party licensees who did previously request sealing is a concern that disclosure of their identities and license details will expose them to other non-practicing patent holders. It remains true that information tending to harass may support sealing. *Chrysler*, 809 F.3d at 1097. Nevertheless, the identities of the patent licensees and the dates and natures of their patent licenses should and will be disclosed here. Again, a patent is a *public* grant of rights. A patent owner is a tenant on a plot within the realm of public knowledge, and a licensee is her sub-tenant. The public has every right to account for all its tenants, all its sub-tenants, and (more broadly) anyone holding even a slice of the public grant.

It also remains true that image licensing or product financial information may be sealed where circumstances warrant. *See Apple*, 727 F.3d at 1225–26; *Elect. Arts*, 298 Fed. App'x 568. Again, however, patent licenses carry unique considerations. The public has an interest in inspecting the valuation of the patent rights as revealed by Uniloc's transactions, particularly given secrecy so often plays to the patentee's advantage in forcing bloated royalties. It may even be that disclosure of prior patent licenses better illuminates the parties' positions, offering up-front cost evaluations of potentially infringing conduct and driving license values to a more accurate representation of the technological value of the patent. In addition, the patent license values here may inform reasonable royalties in other courts.

Conclusive here, though, is the fact that the dates and dollar amounts involved in Uniloc's patent licenses "go to the heart of" the primary dispute, that of Uniloc's standing (or lack of) to sue. *Cf. Chrysler*, 809 F.3d at 1098. The dates and amounts of the licenses revealed Uniloc's failure to meet its time-based monetization goals which, in turn, resulted in Uniloc's default,

released the restriction on Fortress's broad license, and divested Uniloc of standing to sue. Uniloc argues that only the aggregate value of its patent licensing revenue over the relevant time period matters to the analysis of the December 4 order. This misses the point. The public owes little deference to this Court's statements of *fact* and has every right to inspect the bases for those statements. Review of the parties' and the Court's calculation of Uniloc's actual monetization requires public access to the underlying amounts and dates of Uniloc's patent licenses. This information will be unsealed in full.

Turning to Apple's most recent motion to dismiss, Uniloc first seeks to seal references to three specific patent licensees in the excerpts of a deposition transcript. As above, this information will be unsealed.

Uniloc next seeks to seal a three-page extract of an internal Fortress memorandum and brief references to it in Apple's motion. Uniloc's counsel describes the memorandum as Fortress's internal deliberations on whether to invest more in Uniloc's litigation campaign. Thus, counsel asserts, disclosure of the memorandum would broadcast Fortress's business model and criteria for investment to the world, placing it at a competitive disadvantage in negotiations with future investment prospects.

But, as EFF correctly notes, this sealing interest is *Fortress's*, not Uniloc's. Yet Fortress has not submitted a declaration in support of its sealing request. Instead, Uniloc filed the hearsay declaration here, merely reporting what Fortress's counsel apparently said (No. C 18-00358, Dkt. No. 173 at ¶¶ 3, 19–22). Rule 79-5 does not require much to seal. But it does require that "*the Designating Party* must file a declaration . . . establishing that all of the designated material is sealable." Civ. L.R. 79-5(e)(1) (emphasis added). Uniloc and Fortress are more than on notice of the requirements of our local rules regarding sealing. Their continued incapacity to follow or take seriously our local rules has already caused the bulk of their sealing headaches herein. *Uniloc*, 964 F.3d at 1361. Their noncompliance with our local rules has struck again. The investment memorandum and references to it in Apple's brief will be disclosed, and the table of licenses will be disclosed also for the reasons above.

CONCLUSION

EFF's motion to intervene is **GRANTED**. The Court thanks EFF for its most helpful briefing and willingness to vindicate the public's right of access.

Apple's administrative motion to seal its motion to dismiss (No. C 18-00358, Dkt. Nos. 164, 173) is **DENIED**. Uniloc's renewed motion to seal (No. 18-00360, Dkt. No. 222) is **DENIED**. Uniloc waived sealing of Apple's opposition to Uniloc's renewed motion for declaration of subject matter and joinder of Uniloc 2017 (No. C 18-00358, Dkt. Nos. 162, 172). This motion is **DENIED**.

As the Federal Circuit affirmed the denial of Uniloc's previous sealing requests, the motion to place documents in the public record (No. 18-00360, Dkt. No. 220) is **GRANTED**.

The contested denials herein shall be **STAYED** until 28 days after all appeals of this order are exhausted. The parties shall please advise the Court when this period has run and remind the Court to effect the unsealing.

IT IS SO ORDERED.

Dated: December 22, 2020.

WILLIAM ALSUP
UNITED STATES DISTRICT JUDGE

US006661203B2

(12) **United States Patent**
Wolin et al.

(10) **Patent No.: US 6,661,203 B2**
(45) **Date of Patent: Dec. 9, 2003**

(54) **BATTERY CHARGING AND DISCHARGING SYSTEM OPTIMIZED FOR HIGH TEMPERATURE ENVIRONMENTS**

(75) Inventors: **Dale Wolin**, Boise, ID (US); **Eugene Cohen**, Eagle, ID (US); **Richard G. Sevier**, Boise, ID (US)

(73) Assignee: **Hewlett-Packard Development Company, L.P.**, Houston, TX (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 44 days.

(21) Appl. No.: **10/011,140**

(22) Filed: **Nov. 12, 2001**

(65) **Prior Publication Data**

US 2003/0090238 A1 May 15, 2003

(51) **Int. Cl.**[7] **H02J 7/00**
(52) **U.S. Cl.** **320/134**; 320/128
(58) **Field of Search** 320/134, 128, 320/127, 112, 144, 150, 153; 324/426, 427

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,572,110	A *	11/1996	Dunstan	320/106
5,789,899	A *	8/1998	van Phuoc et al.	320/133
6,160,377	A *	12/2000	Fujii	320/117
6,271,643	B1 *	8/2001	Becker et al.	320/112
6,286,109	B1 *	9/2001	Pirdy	713/340
6,366,056	B1 *	4/2002	Podrazhansky et al.	320/141

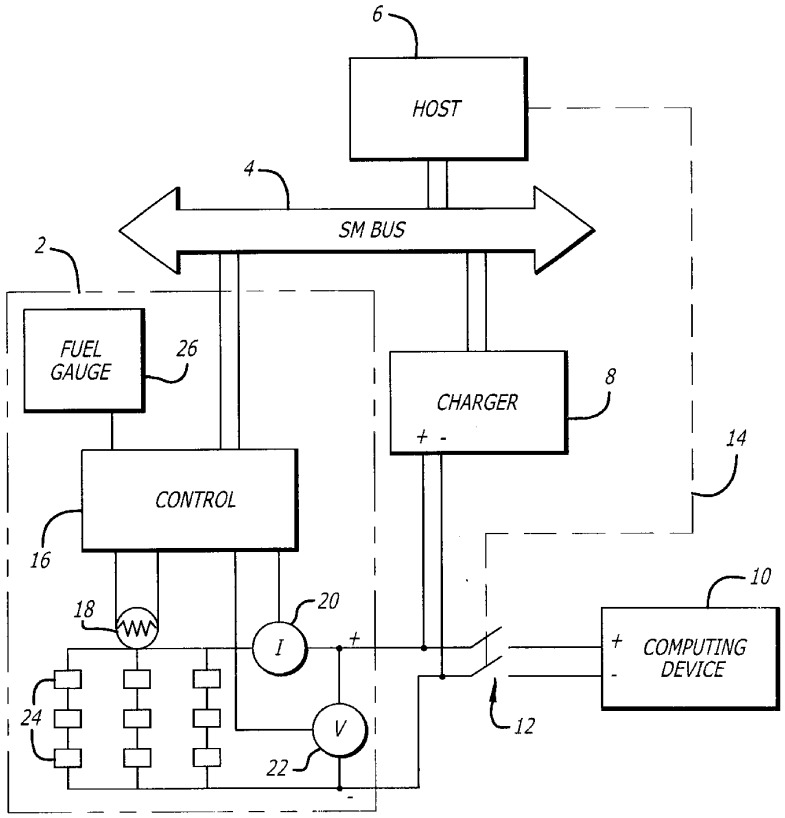
* cited by examiner

Primary Examiner—Michael Sherry
Assistant Examiner—Lawrence Luk

(57) **ABSTRACT**

A method and apparatus for controlling the charge and discharge currents in a battery **(2)** as a function of temperature. When a battery **(2)** is charged or discharged in an environment that approaches its design operating temperature extreme, the currents are reduced to limit self-heating of the battery and thus extend the useful operating environment temperature range. A temperature sensor **(18)** is coupled to a controller **(6)** to sense the battery **(2)** temperature. The temperature information is used to set a suitable charging or discharging current **(8)**.

31 Claims, 2 Drawing Sheets



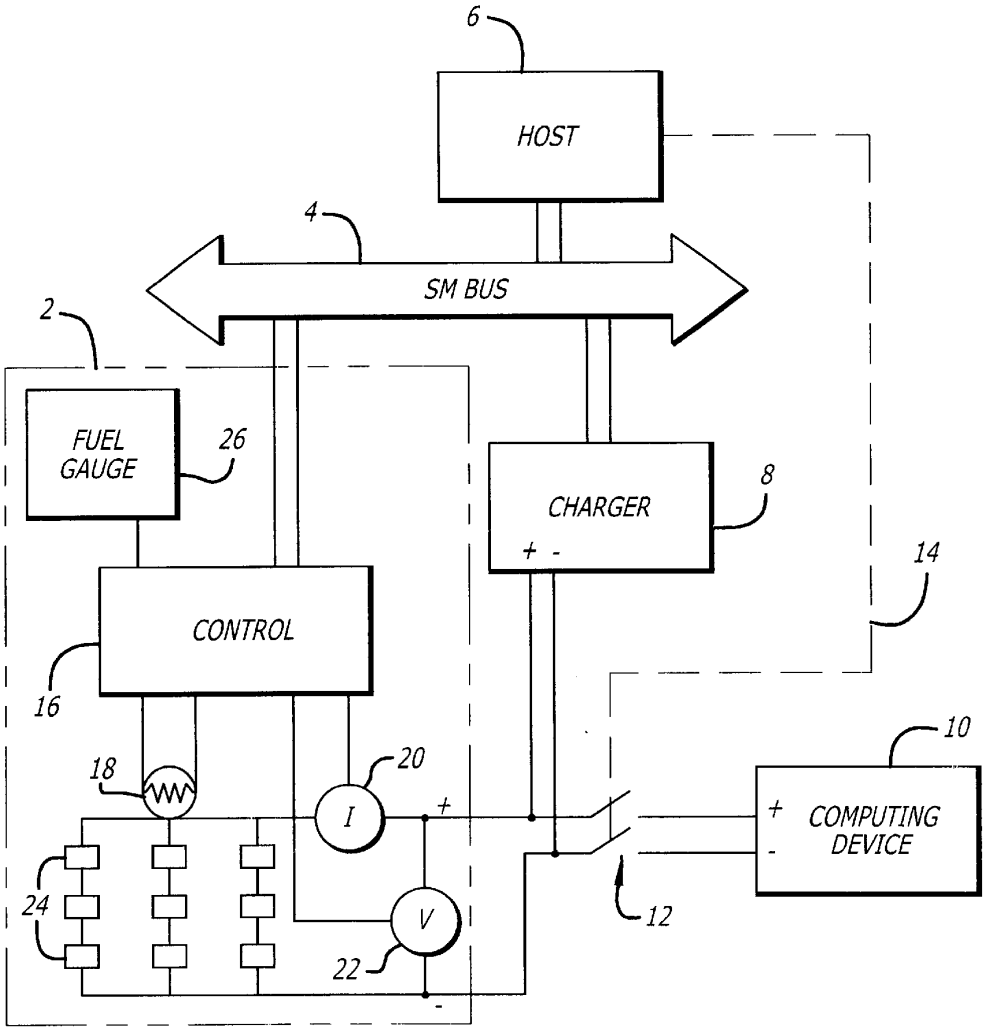

6
HOST
4
SM BUS
2
FUEL GAUGE
26
CHARGER
8
+ -
14
CONTROL
16
10
18
20
I
+
+
COMPUTING DEVICE
-
24
V
12
22
-

FIG. 1

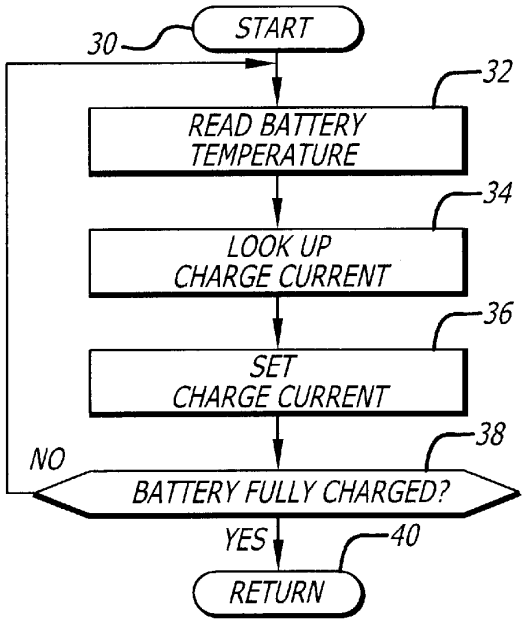

START
30
32
READ BATTERY TEMPERATURE
34
LOOK UP CHARGE CURRENT
36
SET CHARGE CURRENT
38
NO
BATTERY FULLY CHARGED?
YES
40
RETURN

FIG. 2

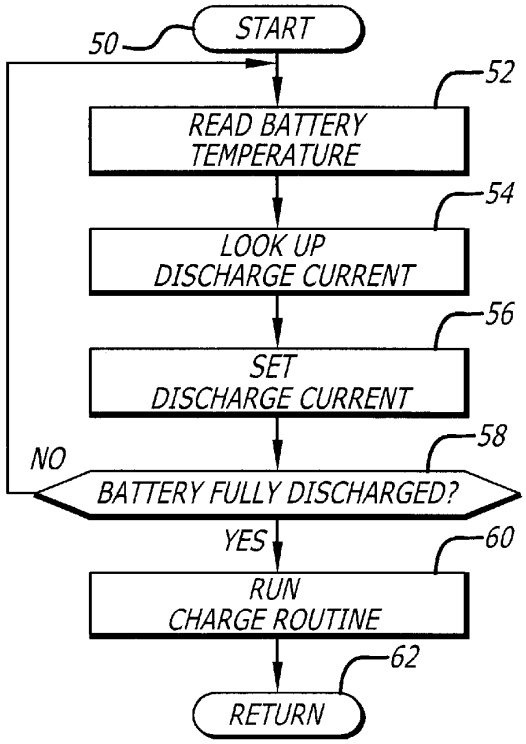

START
50
52
READ BATTERY TEMPERATURE
54
LOOK UP DISCHARGE CURRENT
56
SET DISCHARGE CURRENT
58
NO
BATTERY FULLY DISCHARGED?
YES
60
RUN CHARGE ROUTINE
62
RETURN

FIG. 3

BATTERY CHARGING AND DISCHARGING SYSTEM OPTIMIZED FOR HIGH TEMPERATURE ENVIRONMENTS

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to apparatus and methods for charging rechargeable batteries. More specifically, the present invention relates to apparatus and method for charging, discharging and recharging rechargeable batteries under adverse thermal conditions.

2. Description of the Related Art

Reliable electric power sources are needed to meet the continued growth of electric and electronic business, commercial and personal applications. For portable applications, the chemical storage battery is most commonly employed. For fixed location applications, the public power grid is the most common source of electrical power. Also, alternative sources of power are often employed to produce electric power, such as solar-voltaic, thermal, wind, water and other power sources.

For many applications, a high degree of reliability is required. Although public power grids are highly reliable, these grids are not perfect. Nor are alternative sources of electric power. Therefore, storage batteries are frequently employed in conjunction with, and as a back-up to, the public power grid and alternative sources of electrical power.

Chemical storage batteries have been produced using a variety of technologies. Each technology comprises a number of defining characteristics that should be considered in selecting a suitable technology for a particular application. These include, but are not limited to, size, weight, cost, power density, environmental constraints, voltage, current, power, and so forth.

In many applications, the ability to be recharged is a critical requirement of a chemical storage battery. Rechargeability reduces cost, extends useful life, and adds reliability to both battery and system design. Some common chemical technologies employed in rechargeable batteries are Nickel-Metal Hydride, Lithium Ion, Lithium Ion Polymer, Lead-Acid, and Nickel Cadmium among other unique and hybrid technologies.

Rechargeable batteries are charged by delivering electric current to positive and negative terminals of the battery for a duration of time sufficient to fully charge the battery. Later, current is drawn from the battery as a power source to some particular device or application.

However, the conditions of charging and discharging are not without limitations. The limitations are typically defined by the battery manufacturer or supplier. In applications where a battery is maintained as a back up to another primary source of electrical power, the battery may rest for long periods of time in a fully charged ("standby") state, awaiting an interruption of the primary power source. When this occurs, the electric power stored in the battery is consumed in lieu of the primary power source.

A chemical battery resting in the standby state for long periods of time may degrade due to various factors. The total power available may be reduced, the terminal voltage may change, and the ability to determine the amount of power available may be compromised.

Smart battery charge algorithms have been developed to alleviate some of the problems associated with long term

standby operation of a battery. Such chargers periodically 'condition' the battery by applying an artificial load to discharge the battery to some predetermined level, and then recharge the battery to full charge. During such a conditioning process, certain metrics may be measured and used to calibrate the battery for later determination of the available power during a battery discharge cycle. It is desirable to process a discharge cycle in as short a period of time as possible so that the battery can quickly be returned to standby operation. Similarly, it is generally desirable to charge a battery as quickly as possible so that it can be readied for use as quickly as possible.

When a battery is being charged or discharged, a certain amount of internal heat is generated as current flows through the battery. This heat is proportional to the amount of current flowing within the battery. In ambient conditions where the amount of heat generated is small compared to the heat loss from the battery, the internal heat generation is usually not significant. Often, a battery is located in close physical proximity to the device it powers or to which it provides standby service. An example of this is occurs when a battery is used to provide standby power to a computing device. In most instances, the device with which the battery operates also generates heat during operation.

Electrical energy discharged from the battery can cause thermal problems at high temperature, for both the battery and the adjacent circuitry. For example, a battery may be subjected to heat energy produced by the device it powers as well as the heat the battery produces internally. In addition, the components adjacent to the battery conditioning circuit (often a resistive load) may be pushed close to thermal limits due to joule heating of the discharge load at high temperature.

In addition, other heat sources in the vicinity of the battery may affect ambient conditions and raise the operating temperature of the environment. Thus, it is not uncommon for a battery to be operated at substantially elevated temperatures.

When a battery is operating at or near its maximum operating temperature, designers are faced with a dilemma. If the battery charge and discharge currents are maintained at levels normally applied for the lower ranges of expected operating temperatures, the battery life and reliability can be greatly compromised when temperatures become elevated. On the other hand, if the designer takes a conservative approach, and sets the charge and discharge currents at levels consistent with a reasonable maximum operating temperature, then charge and discharge currents may be so low that the time required to accomplish these operations become unacceptably long.

Alternatives presently available to address this dilemma include locating the battery in a cooler environment, usually distant from the device being powered and providing additional cooling equipment. Each of these alternatives is typically undesirable due to increased cost, greater systems complexity, or reduced reliability, inter alia.

Thus there is a need in the art for an apparatus and method for efficiently charging, discharging and recharging batteries in environments with variable thermal conditions.

SUMMARY OF THE INVENTION

The need in the art is addressed by the apparatus and methods taught by the present invention. An apparatus for charging a battery according to its temperature is taught. The apparatus includes a charging circuit adapted to charge a battery and a temperature sensor positioned to sense a

battery temperature, i.e., adjacent environmental temperature. The apparatus includes a controller coupled to the temperature sensor and the charging circuit. The controller operates to set the charging current in accordance with the sensed temperature.

In a refinement, the controller continuously sets the charging current in accordance with the sensed temperature. In a further refinement, the controller periodically sets the charging current in accordance the sensed temperature. In a further refinement, the apparatus further includes a memory coupled to the controller having a temperature and charging current look up table stored therein. In this embodiment, the controller accesses the look up table to set the charging current. In a further refinement, the controller operates to set the charging current to a maximum value when the temperature is lower than a first predetermined threshold value. In a further refinement, the maximum value is the battery's maximum specified charging current and the first predetermined threshold value is the battery's maximum charging temperature. In a further refinement, the controller sets the charging current to zero when the temperature is higher than a second predetermined threshold value. In a further refinement, the battery is coupled to a load and the temperature sensor senses the temperature of the battery and the load.

The present invention also teaches an apparatus for exercising or conditioning a battery. This apparatus includes the charging circuit and a temperature sensor. Also, a discharging circuit is coupled to the battery while a controller is coupled to the temperature sensor, the charging circuit, and the discharging circuit. The controller operates to set the charging and discharging currents in accordance with temperature.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a functional block diagram of an illustrative embodiment of the present invention.

FIG. **2** is a flow diagram of an illustrative embodiment of the present invention.

FIG. **3** is a flow diagram of an illustrative embodiment of the present invention.

DESCRIPTION OF THE INVENTION

Illustrative embodiments and exemplary applications will now be described with reference to the accompanying drawings to disclose the advantageous teachings of the present invention.

While the present invention is described herein with reference to illustrative embodiments for particular applications, it should be understood that the invention is not limited thereto. Those having ordinary skill in the art and access to the teachings provided herein will recognize additional modifications, applications, and embodiments within the scope thereof and additional fields in which the present invention would be of significant utility.

The present invention advantageously utilizes a temperature sensor in combination with a battery charger, or a battery conditioner, to control charging and discharging current flow as a function of the battery temperature. As is understood by those skilled in the art, rechargeable batteries are characterized by a number of operational constraints. Among these are terminal and charging voltage, maximum charging current flow, maximum current draw, and a range of environmental constraints, including maximum operation, charging and discharging temperatures. Implementation of a battery in a system that operates outside the bounds of such constraints will lead to a number of deleterious effects. These include reduced battery life, reduced battery capacity, and certain potentials for dangerous situations including overheating, fire, and chemical leakage. Thus, designers strive to maintain operational factors within design constraints. Yet, in certain practical applications, designers are forced to implement batteries in environments that push the limits of these constraints. The present invention allows designers to move closer to the absolute limits, while still maintaining maximum performance from the battery and the system into which it is installed.

As discussed above, a number of chemical technologies are employed in modern rechargeable batteries. Each technology is constrained as noted above. In an illustrative embodiment, a lithium ion battery is employed. Nonetheless, it will be understood by those of ordinary skill in the art that the teachings provided herein are not limited to a particular battery technology.

When a battery is charged or discharged, current flows through the battery and a certain amount of internal heat is produced. When the battery is being charged or discharged in an environment near its maximum operating temperature, the internal heat generated can push the battery beyond it design constraints, leading to the aforementioned deleterious effects. Operating environments that are near a battery's operating extremes are rather common. For example, a battery back-up system for a computing device, such as a computer or mass storage system, is often times located in close proximity to the computing device. The heat produced by the computing device contributes to the heat of the environment that the battery operates within. Also ambient conditions may be warm or hot, exacerbating the thermal environment. There are many other applications that push the thermal constraints, including outdoor, mobile, industrial, non-air conditioned, and other similar environments. The present invention advantageously balances the current flow in the battery, thus balancing the internal heat generation and build-up, with the battery and local environmental temperature.

In an illustrative embodiment, a lithium ion smart battery is employed in a computer storage disk array system and the present invention is implemented to allow the system to extract maximum performance from the battery without exceeding safe operational constraints for the battery.

The Smart Battery industry standard describes one or more battery cells in conjunction with a controlling device that enables the battery to measure and communicate certain information about its operation to a user or an external device. An implementation of a Smart Battery, which is the battery employed in an illustrative embodiment of the present invention, is the Moltech Power Systems model NI2040A17 Rechargeable Lithium Ion Battery, specifications for which are available from Moltech Power Systems, Inc., 12801 NW Highway 441, Alachua, Fla. 32615. This Smart Battery employs lithium ion chemistry in nine storage cells that are arranged in a three by three series-parallel configuration to yield a nominal terminal voltage of 10.8 volts and a power rating of 5000 milli-Ampere hours ("mAhr"). The Smart Battery employs a controller and a "fuel gauge" which is coupled to a display that indicates the battery's power reserve in twenty percent increments. The Smart Battery comprises a thermistor temperature sensor within its housing. The Smart Battery also comprises an SMBus two-wire serial communications port, as is understood by those possessing ordinary skill in the art. The SMBus interface generally applies the industry standard I^2C

signaling levels. The SMbus is operable to communicate the smart battery's terminal voltage, the rate of current flow into or out of the battery, the charges state, including whether the battery is fully charged or fully discharged, and the temperature of the battery, according to the aforementioned thermistor temperature sensor. In the illustrative embodiment, the SMBus is coupled to a host controller, as will be more fully discussed herein after. The illustrative embodiment Smart Battery specifications provide a maximum charge current to 3 amperes at 12.6 volts in the range of temperatures from 0° C. to 45° C. Discharge is rated at 3 amperes from 0° C. to 50° C. Full charge is realized when the charging current drops below 150 milli-amperes.

Reference is directed to FIG. **1**, which is a functional block diagram of an illustrative embodiment of the present invention. The aforementioned Smart Battery **2** comprises a plurality of lithium ion cells **24** that are arranged in a series-parallel configuration. A thermistor **18** is located within the battery **2** at a position enabling it to sense the temperature immediately adjacent to the battery cells **24**. The thermistor **18** is coupled to a controller **16**, which is operable to read the temperature via the thermistor **18**. The controller **16** is coupled to a current sensor **20** that enables the controller **16** to monitor the current flow through the battery **2**. The controller **16** is also coupled to a voltage sensor **22** that enables the controller **16** to monitor the battery **2** terminal voltage. A fuel gauge **26** is provided that displays the remaining battery capacity, as well as making this information available to the controller **16**.

The battery **2** controller **16** is coupled to an SMBus **4** enabling communications of the aforementioned parameters through the SMBus **4**. The positive and negative output terminals of battery **2** are coupled though relay **12** to a load, which is a computing device **10** in this illustrative embodiment. A programmable charger **8** is coupled to the battery **2** and enables the supply of charging current to the battery **2**. The charger **8** comprises an SMBus interface coupled to SMBus **4**, which interface allows the charger **8** to be programmed to deliver a specified current and voltage to the battery **2** for charging thereof. A host controller **6** is coupled to the SMBus **4** and is operable to control the operation of this embodiment of the present invention. The host controller **6** is also coupled **14** to actuate relay **12**, which may be accomplished either directly (as shown) or through an SMBus interface (not shown). The host controller **6** may be any of a variety of processors, microprocessors, controllers, microcontrollers, or other programmable devices as are presently understood, or later become available, to those possessing ordinary skill in the art. The host controller includes an amount of random access memory in the illustrative embodiment. The temperature sensor may be a thermistor, a thermocouple, an infrared sensor, or any other sensor having an output proportional to temperature that is understood by those possessing ordinary skill in the art.

The host controller **6** memory is programmed with a look up table of charging and discharging currents related to temperatures. In the illustrative embodiment, these relations are determined through empirical measurements. Table 1 below shows illustrative charging current and temperature values:

TABLE 1

Temperature	Current
Less than 45° C.	2.0 Amps
45° C. to 55° C.	1.0 Amps
55° C. to 60° C.	0.5 Amps
Greater then 60° C.	0.0 Amps

In operation, the host controller **16** periodically requests the battery temperature from the smart battery **2** and uses this value to access the memory look up table to select a charging current associated with that temperature. By applying these reduced current values, a corresponding reduction in the self-heating of the battery cells is caused by the current flow. This reduction allows the battery to function in a correspondingly warmer environment at the system level. For example, a reduction of the charge current by 50% will reduce the power, and hence the self-heating term, by the square of the charge, or 75%. This readily provides an improvement of 5° C. compared to the battery suppliers recommend extreme temperatures of operation.

As may be expected with respect to the charging cycle, a self-heating term is associated with cell temperatures due to the discharging cycle. When a battery is conditioned, or exercised, the system discharges the battery to a predetermined level. This allows the system to calibrate the battery and assess capacity and useful life, as is understood by those skilled in the art. The battery is then recharged, as discussed above. The discharge rate is reduced in like fashion to the charge rate, thus reducing self-heating and extending the battery's useful life. Also note that the discharge current is directed to a load, such as a resistive load, that converts the battery energy into heat as it is discharged. In the illustrative embodiment, a variable impedance load, under control of the host controller, is employed. A look up table in the memory is used to recall empirically derived factors for suitable discharge current rates, in like fashion with respect to the charging approach. The load is typically located in close proximity to the battery and thus the heat produced affects the battery's environment. The temperature sensor should be positioned to detect this heat, thereby allowing the system to respond accordingly.

Reference is directed to FIG. **2**, which is a flow diagram of an illustrative embodiment of a charging operation according to the present invention. The process is called by the host controller at step **30** and proceeds to read the battery temperature at step **32**. The battery temperature returned is used to access the look up table in the memory at step **34**. The current associated with that temperature is recalled and used to set the output current of the charger at step **36**. At step **38**, the host controller reads the charge state over the SMBus to determine whether the battery is fully charged or not. If the battery is fully charged at step **38**, then the process returns to the calling routine at step **40**. On the other hand, if the battery is not fully charged at step **38**, then the flow returns to step **32** to repeat the process.

The foregoing describes an operation where the battery temperature is effectively continuously tested by the reiterative loop. In a practical application, it may be preferred to add a fixed time delay because the thermal mass of the battery will prevent sudden jumps in temperature. Thus, the process can readily be adapted from a continuous test to a periodic test, suitable for a given application and environment.

Reference is directed to FIG. **3**, which is a flow diagram of an illustrative embodiment of the conditioning, or

exercise, operation taught by the present invention. The process is called by the host controller at step **50** and proceeds to read the battery temperature at step **52**. The battery temperature returned is used to access the look up table in the memory at step **54**. The discharge current associated with that temperature is recalled and used to set the load impedance, or discharge current at step **56**. At step **58**, the host controller reads the charge state over the SMBus to determine whether the battery is fully discharged or not. If the battery is fully discharged at step **58**, then the process proceeds to step **60** where the charging process of FIG. **2** is executed. After the charging process is completed, flow returns to the calling routine at step **62** in FIG. **3**.

On the other hand, if the battery is not fully discharged at step **58** in FIG. **2**, then the flow returns to step **52** to repeat the process. The foregoing describes an operation where the battery temperature is effectively continuously tested by the reiterative loop. In a practical application, it may be preferred to add a fixed time delay because the thermal mass of the battery will prevent sudden jumps in temperature. Thus, the process can readily be adapted from a continuous test to a periodic test, suitable for the application and environment at hand.

Thus, the present invention has been described herein with reference to a particular embodiment for a particular application. Those having ordinary skill in the art and access to the present teachings will recognize additional modifications applications and embodiments within the scope thereof.

It is therefore intended by the appended claims to cover any and all such applications, modifications and embodiments within the scope of the present invention.

What is claimed is:

1. An apparatus for charging a battery comprising

a charging circuit for providing a charging current to the battery;

a temperature sensor positioned to sense a temperature of said battery; and

a controller coupled to said temperature sensor and said charging circuit and operable to control said charging circuit in accordance with said temperature, said controller operable to set said charging current to zero when said temperature is higher than a first predetermined threshold value.

2. The apparatus of claim **1** wherein said controller continuously sets said charging current in accordance with said temperature.

3. The apparatus of claim **1** wherein said controller periodically sets said charging current in accordance with said temperature.

4. The apparatus of claim **1** further comprising a memory coupled to said controller having a temperature and charging current look up table stored therein, and wherein said controller accesses said look up table to set said charging current.

5. The apparatus of claim **1** wherein said controller is operable to set said charging current to a maximum value when said temperature is lower than a second predetermined threshold value.

6. The apparatus of claim **5** wherein said maximum value is the battery's maximum specified charging current, and said second predetermined threshold value is the battery's maximum charging temperature.

7. The apparatus of claim **1** wherein the battery is coupled to a load, and wherein said temperature sensor senses that temperature of the battery and the load.

8. An apparatus for exercising a battery, comprising

a charging circuit having a charging current output coupled to the battery;

a temperature sensor positioned to sense a temperature related to the battery temperature;

a discharging circuit having a discharging current input coupled to the battery; and

a controller coupled to said temperature sensor, said charging circuit, and said discharging circuit, said controller operable to set said charging current in accordance with said temperature, and operable to set said discharging current in accordance with said temperature, said controller being operable to set said discharging current to zero when said temperature is higher than a first predetermined threshold value.

9. The apparatus of claim **8** and wherein said controller continuously sets said discharging current in accordance with said temperature.

10. The apparatus of claim **8** and wherein said controller periodically sets said discharging current in accordance with said temperature.

11. The apparatus of claim **8** further comprising a memory coupled to said controller having a temperature versus discharging current look up table stored therein, and wherein said controller accesses said look up table to set said discharging current.

12. The apparatus of claim **11** and wherein said discharging circuit comprises a variable impedance load and wherein said look up table values correspond to values of said variable impedance load.

13. The apparatus of claim **8** and wherein said controller is operable to set said discharging current to a maximum value when said temperature is lower than a second predetermined threshold value.

14. The apparatus of claim **8** wherein said maximum value is the battery's maximum specified discharging current and said second predetermined threshold value is the battery's maximum discharging temperature.

15. The apparatus of claim **8** wherein said temperature sensor senses the temperature of the battery and said discharging circuit.

16. A method of charging a battery, comprising the steps of:

sensing a temperature related to the battery temperature;

setting a charging current in accordance with said sensed temperature and setting said charging current to zero when said temperature is higher than a first predetermined threshold value; and

charging the battery at said charging current.

17. The method of claim **16** and wherein said sensing and setting steps are repeated continuously during said charging step.

18. The method of claim **16** and wherein said sensing and setting steps are repeated periodically during said charging step.

19. The method of claim **16** and wherein said setting step further comprises the step of recalling a charging current corresponding to said sensed temperature from a look up table.

20. The method of claim **16** and wherein set setting step includes setting said charging current to a maximum value if said temperature is lower than a second predetermined threshold.

21. The method of claim **20** and wherein said maximum value is the battery's maximum specified charging current, and said second predetermined threshold is the battery's maximum charging temperature.

22. The method of claim 16 wherein the battery is coupled to a load, and wherein said sensing step includes sensing the temperature of the battery and the load.

23. A method of exercising a battery, comprising the steps of:

sensing a temperature related to the battery temperature;

setting a discharging current in accordance with said temperature;

discharging the battery at said discharging current;

discontinuing said discharging step when a predetermined battery voltage is reached;

setting a charging current in accordance with said temperature, said setting step further including the step of setting said discharging current to zero when said temperature is higher than a first predetermined threshold value; and

charging the battery at said charging current.

24. The method of claim 23 and wherein said sensing and setting a discharge current steps are repeated continuously during said discharging step.

25. The method of claim 23 and wherein said sensing and setting a discharge current steps are repeated periodically during said discharging step.

26. The method of claim 23 and wherein said setting step further comprises the step of recalling a discharging current corresponding to said sensed temperature from a look up table.

27. The method of claim 23 and wherein said setting step includes setting said discharging current to a maximum value if said temperature is lower than a second predetermined threshold.

28. The method of claim 27 and wherein said maximum value is the battery's maximum specified discharging current, and said second predetermined threshold is the battery's maximum discharging temperature.

29. The method of claim 23 wherein the battery is coupled to a load, and wherein said sensing step includes sensing the temperature of the battery and the load.

30. An apparatus for exercising a battery, comprising

a charging circuit having a charging current output coupled to the battery;

a temperature sensor positioned to sense a temperature related to the battery temperature;

a discharging circuit having a discharging current input coupled to the battery; and

a controller coupled to said temperature sensor, said charging circuit, and said discharging circuit, said controller operable to set said charging current in accordance with said temperature, and operable to set said discharging current in accordance with said temperature, said controller being operable to set said discharging current to a maximum value when said temperature is lower than a first predetermined threshold value, said maximum value being the battery's maximum specified discharging current, and said first predetermined threshold value being the battery's maximum discharging temperature.

31. A method of exercising a battery, comprising the steps of:

sensing a temperature related to the battery temperature;

setting a discharging current in accordance with said temperature;

discharging the battery at said discharging current;

discontinuing said discharging step when a predetermined battery voltage is reached;

setting a charging current in accordance with said temperature, said setting step further including the step of setting said discharging current to a maximum value if said temperature is lower than a first predetermined thresholds said maximum value being the battery's maximum specified discharging current and said first predetermined threshold value being the battery's maximum discharging temperature; and

charging the battery at said charging current.

* * * * *

US007092671B2

(12) **United States Patent**
Lunsford et al.

(10) **Patent No.: US 7,092,671 B2**
(45) **Date of Patent: Aug. 15, 2006**

(54) **METHOD AND SYSTEM FOR WIRELESSLY AUTODIALING A TELEPHONE NUMBER FROM A RECORD STORED ON A PERSONAL INFORMATION DEVICE**

(75) Inventors: **E. Michael Lunsford**, San Carlos, CA (US); **Steve Parker**, Centerville, UT (US); **David Kammer**, Seattle, WA (US); **David Moore**, Riverton, UT (US)

(73) Assignee: **3Com Corporation**, Marlborough, MA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 615 days.

(21) Appl. No.: **09/727,727**

(22) Filed: **Nov. 30, 2000**

(65) **Prior Publication Data**

US 2002/0065041 A1 May 30, 2002

(51) **Int. Cl.**
H04B 7/00 (2006.01)

(52) **U.S. Cl.** **455/41.2**; 455/412.1; 455/418; 455/419; 455/414.3; 455/466; 455/565; 455/557; 455/563; 455/410; 455/411

(58) **Field of Classification Search** 455/41, 455/411, 418, 41.2, 41.3, 412.1, 419, 414.3, 455/557, 563
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,109,403 A * 4/1992 Sutphin 455/419
5,655,219 A * 8/1997 Jusa et al. 455/338
5,878,339 A * 3/1999 Zicker et al. 455/419
5,961,600 A * 10/1999 Ono et al. 709/228
5,983,100 A * 11/1999 Johansson et al. 455/426.1
6,069,588 A * 5/2000 O'Neill, Jr. 343/713
6,122,523 A * 9/2000 Zicker et al. 455/551
6,282,433 B1 * 8/2001 Holshouser 455/556.2
6,331,972 B1 * 12/2001 Harris et al. 370/313
6,374,079 B1 * 4/2002 Hsu 455/11.1
6,424,820 B1 * 7/2002 Burdick et al. 455/41.1
6,484,027 B1 * 11/2002 Mauney et al. 455/421
6,515,575 B1 * 2/2003 Kataoka 340/5.8
6,577,877 B1 * 6/2003 Charlier et al. 455/557
6,584,080 B1 * 6/2003 Ganz et al. 370/315
6,600,902 B1 * 7/2003 Bell 455/411
6,650,871 B1 * 11/2003 Cannon et al. 455/41.2
6,728,531 B1 * 4/2004 Lee et al. 455/419

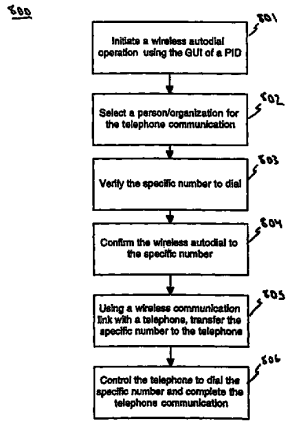
* cited by examiner

Primary Examiner—Marceau Milord

(57) **ABSTRACT**

An automated telephone dialing system. The system enables a handheld PID to automatically dial a telephone number stored its memory by interacting with a telephone. The telephone is equipped with a wireless port for short-range wireless data transfer. Similarly, the PID is equipped with a wireless port for short-range wireless data transfer. The PID establishes a wireless communication with the telephone. The PID is configured to control the telephone via the wireless communications such that the telephone dials a telephone number stored on the PID. The telephone number can be dialed in response to the user interacting with application executing on the PID. The application can be a contact management or address management program. The user can interact with the program, select a contact, address, phone number, or the like, through a GUI of the PID, and have this number automatically dialed by the telephone. In this manner, the user's PID seamlessly interacts with the user's telephone to dial numbers and establish phone calls without requiring the user to access controls of the telephone. The wireless communication between the PID and the telephone can be compatible with a version of the Bluetooth specification. The wireless communication between the PID and the telephone can also be compatible with a version of the IrDA specification.

16 Claims, 9 Drawing Sheets



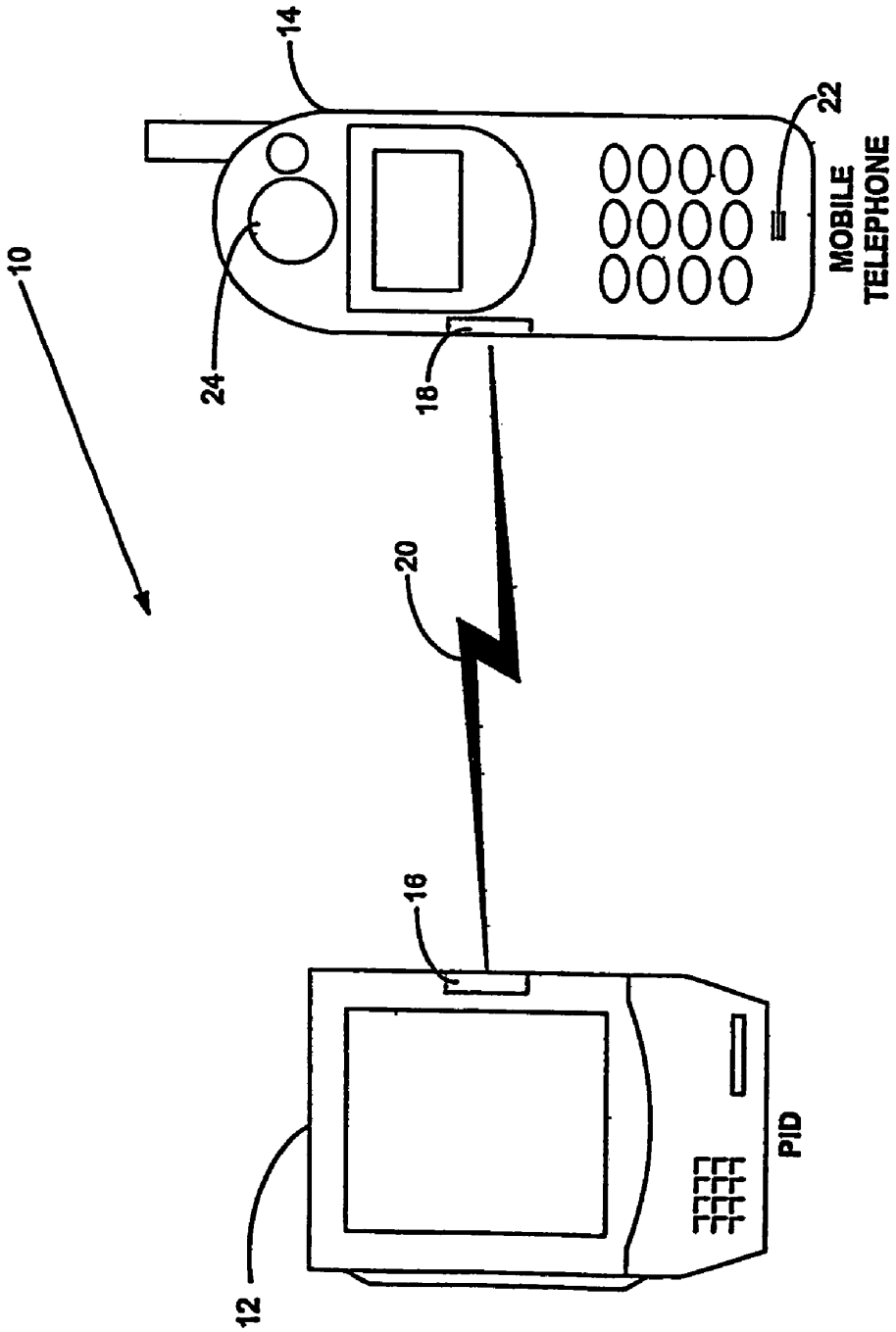


FIG. 1

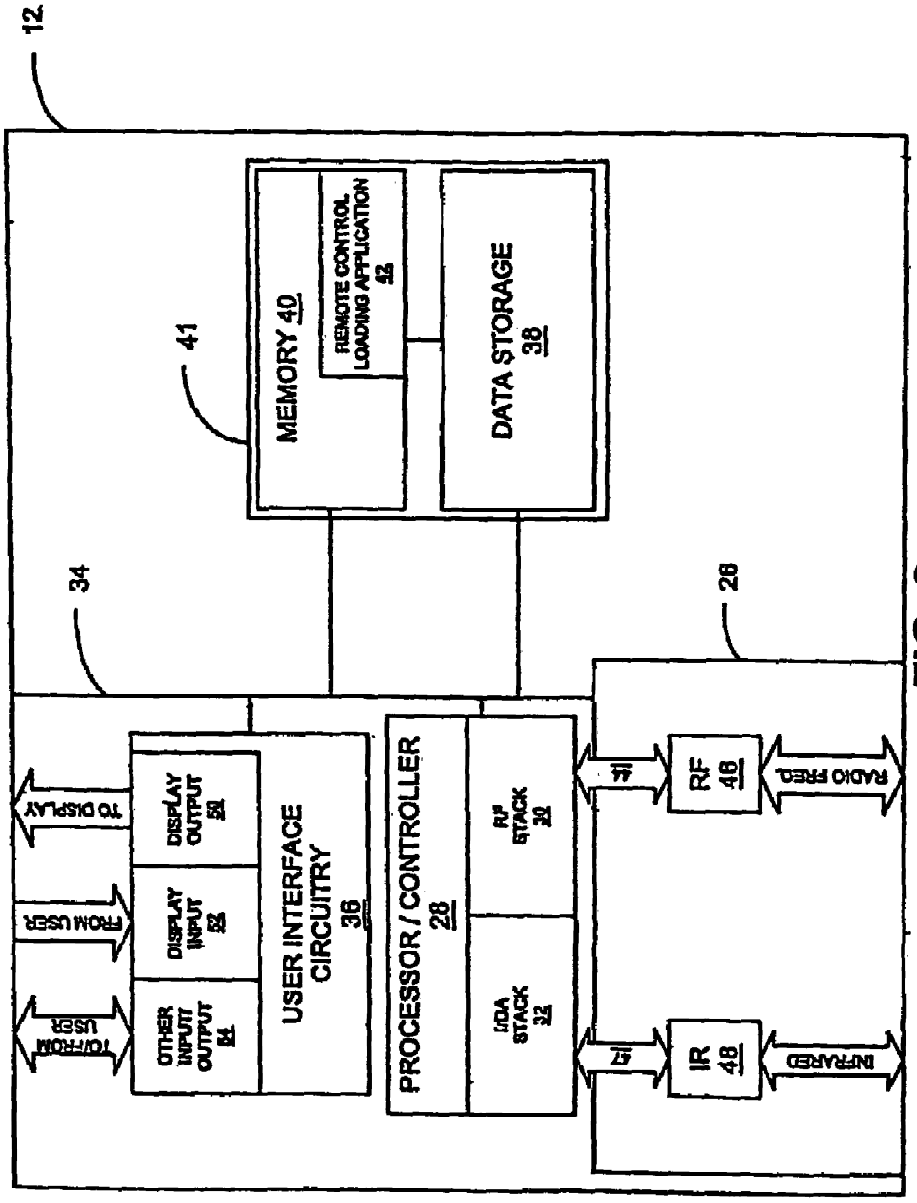

12
41
MEMORY 40
REMOTE CONTROL LOADING APPLICATION 42
DATA STORAGE 38
34
26
USER INTERFACE CIRCUITRY 36
DISPLAY OUTPUT 50
DISPLAY INPUT 52
OTHER INPUT OUTPUT 54
TO DISPLAY
FROM USER
TO/FROM USER
PROCESSOR / CONTROLLER 28
RF STACK 30
IrDA STACK 32
RF 46
IR 48
RADIO FREQ.
INFRARED

FIG. 2

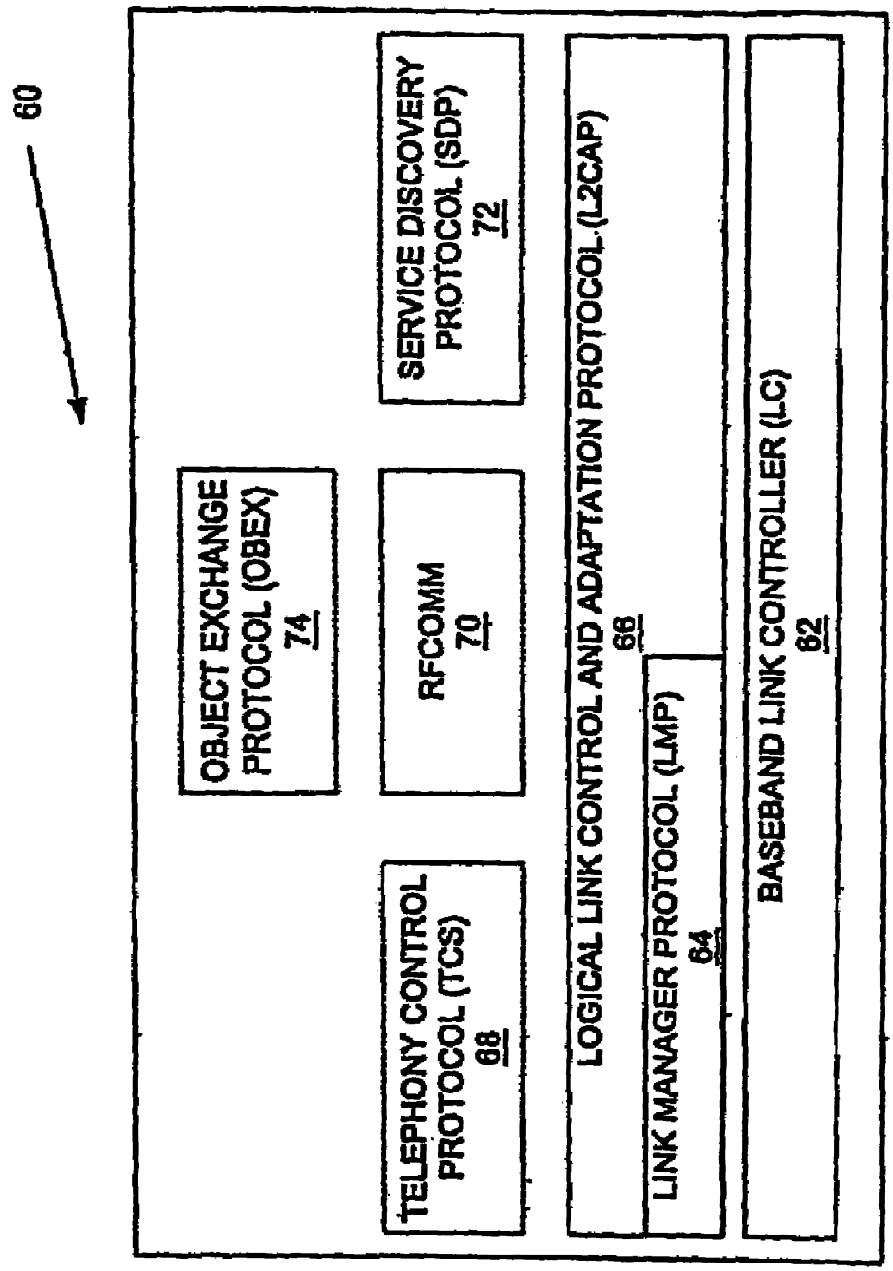

60
OBJECT EXCHANGE PROTOCOL (OBEX) 74
RFCOMM 70
SERVICE DISCOVERY PROTOCOL (SDP) 72
TELEPHONY CONTROL PROTOCOL (TCS) 68
LOGICAL LINK CONTROL AND ADAPTATION PROTOCOL (L2CAP) 66
LINK MANAGER PROTOCOL (LMP) 64
BASEBAND LINK CONTROLLER (LC) 62

FIG. 3

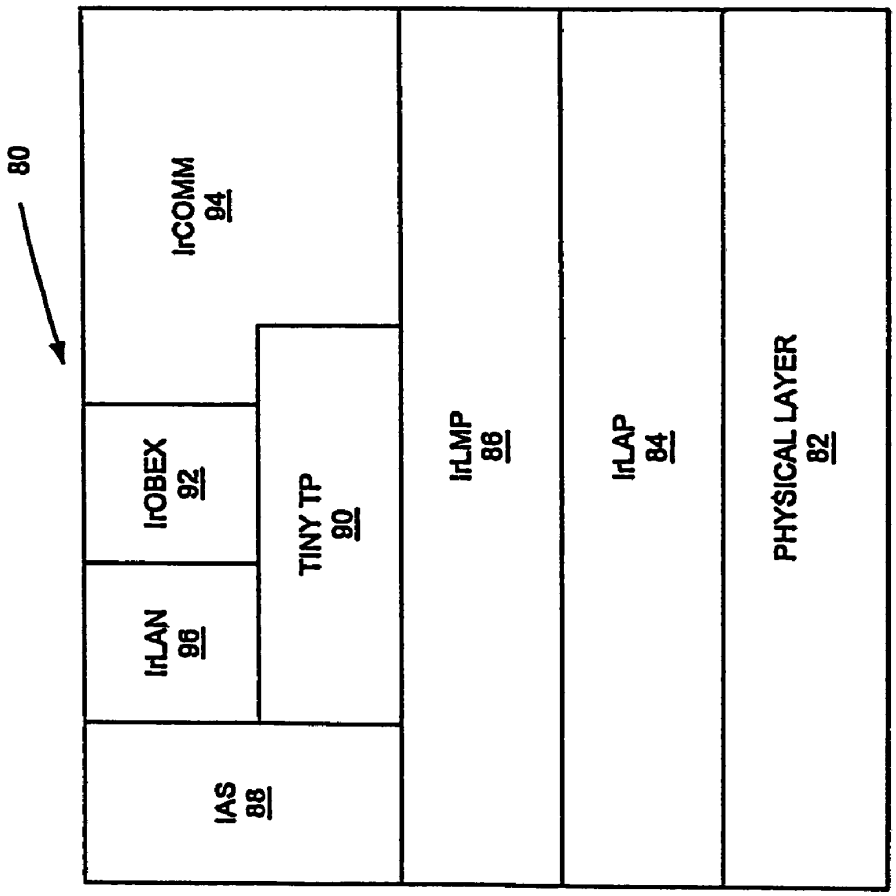

80
IrCOMM
94
IrOBEX
92
IrLAN
96
TINY TP
90
IAS
88
IrLMP
86
IrLAP
84
PHYSICAL LAYER
82

FIG. 4

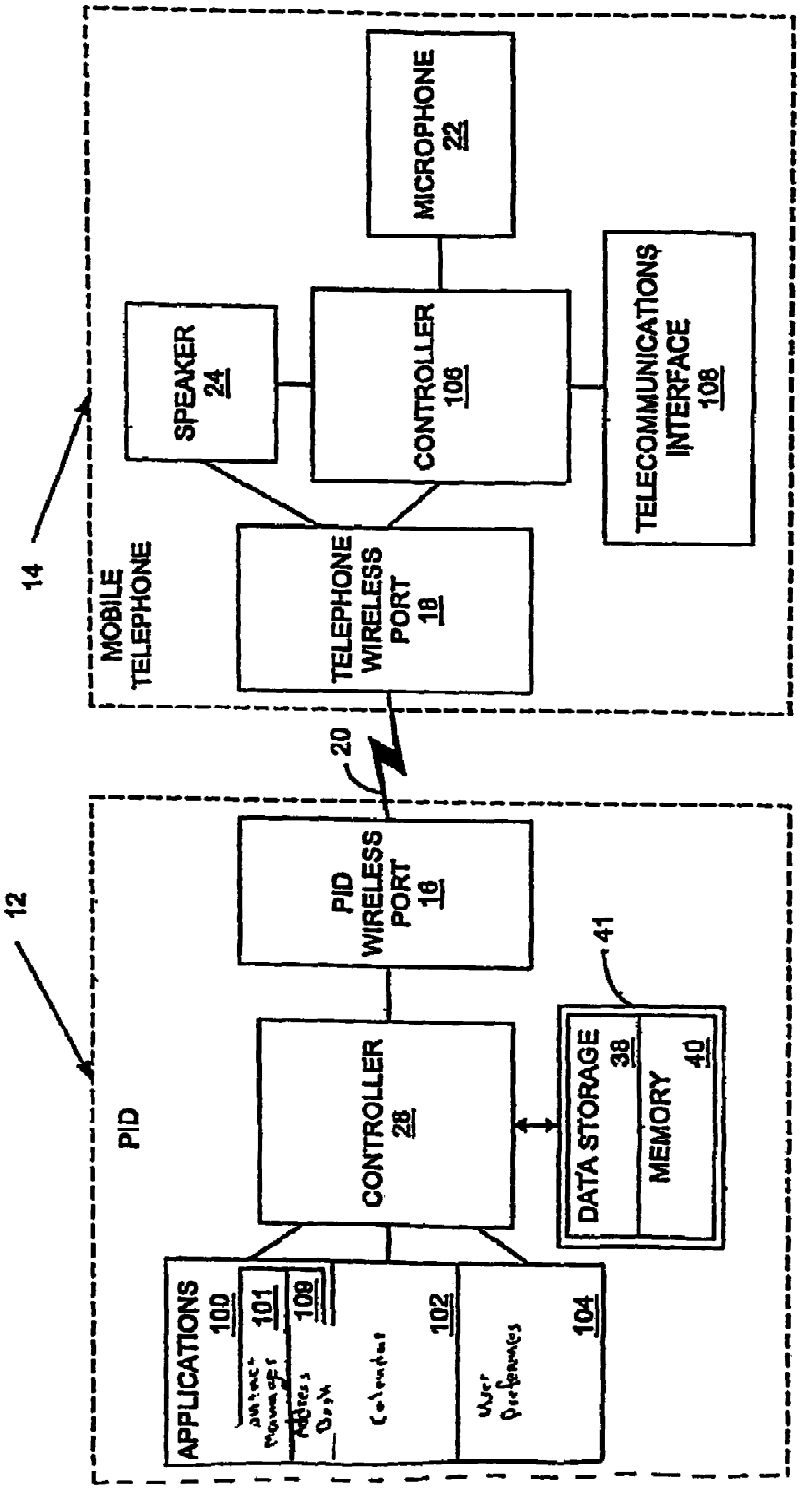

MICROPHONE 22
SPEAKER 24
CONTROLLER 106
TELECOMMUNICATIONS INTERFACE 108
MOBILE TELEPHONE
TELEPHONE WIRELESS PORT 18
14
20
PID WIRELESS PORT 16
12
PID
CONTROLLER 28
41
DATA STORAGE 38
MEMORY 40
APPLICATIONS 100
101
102
104

FIG. 5

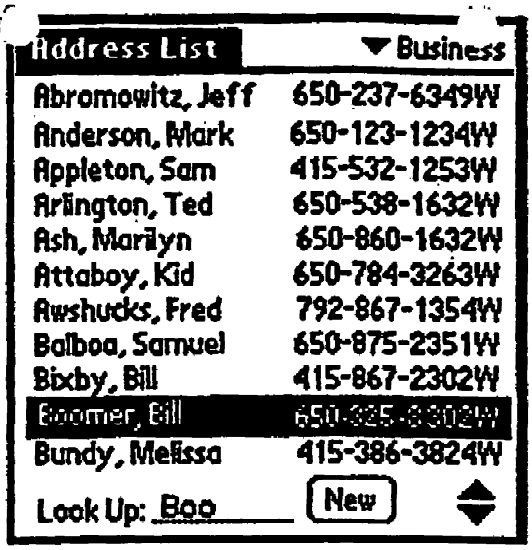

Address List
Business
Abromowitz, Jeff 650-237-6349W
Anderson, Mark 650-123-1234W
Appleton, Sam 415-532-1253W
Arlington, Ted 650-538-1632W
Ash, Marilyn 650-860-1632W
Attaboy, Kid 650-784-3263W
Awshucks, Fred 792-867-1354W
Balboa, Samuel 650-875-2351W
Bixby, Bill 415-867-2302W
Boomer, Bill
Bundy, Melissa 415-386-3824W
Look Up: Boo
New

Fig 6A

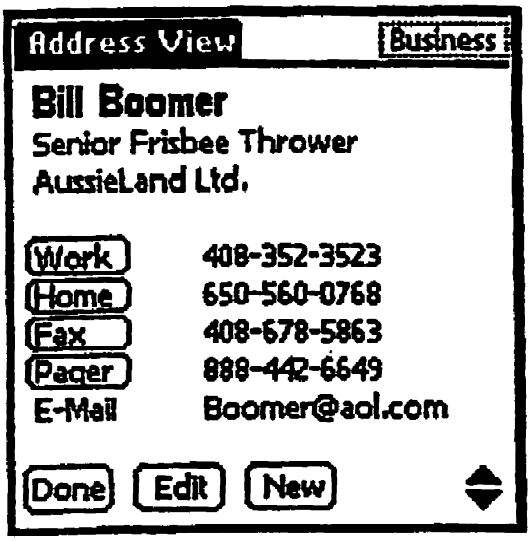

Address View
Business
Bill Boomer
Senior Frisbee Thrower
AussieLand Ltd.
Work 408-352-3523
Home 650-560-0768
Fax 408-678-5863
Pager 888-442-6649
E-Mail Boomer@aol.com
Done
Edit
New

Fig 6B

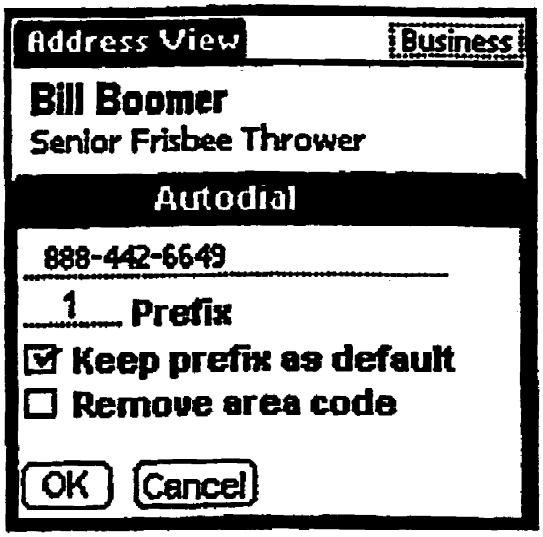

Address View
Business
Bill Boomer
Senior Frisbee Thrower
Autodial
888-442-6649
1 Prefix
Keep prefix as default
Remove area code
OK
Cancel

Fig 6C

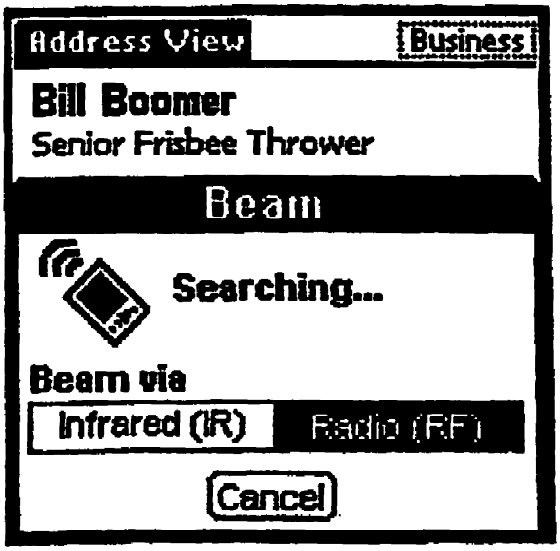

Address View
Business
Bill Boomer
Senior Frisbee Thrower
Beam
Searching...
Beam via
Infrared (IR)
Radio (RF)
Cancel

Fig 6D

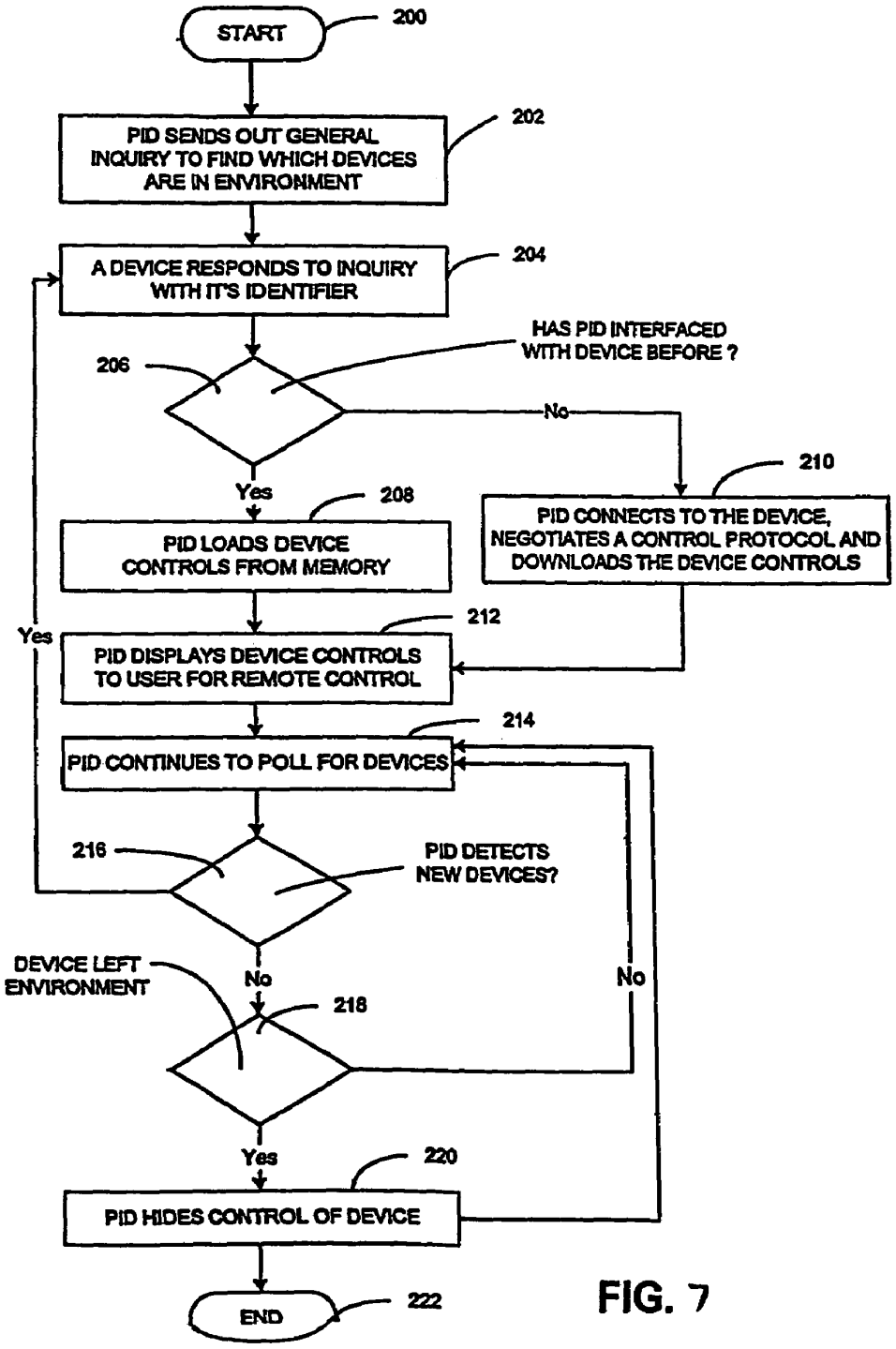

START
200
PID SENDS OUT GENERAL INQUIRY TO FIND WHICH DEVICES ARE IN ENVIRONMENT
202
A DEVICE RESPONDS TO INQUIRY WITH IT'S IDENTIFIER
204
206
HAS PID INTERFACED WITH DEVICE BEFORE ?
No
Yes
208
PID LOADS DEVICE CONTROLS FROM MEMORY
210
PID CONNECTS TO THE DEVICE, NEGOTIATES A CONTROL PROTOCOL AND DOWNLOADS THE DEVICE CONTROLS
212
PID DISPLAYS DEVICE CONTROLS TO USER FOR REMOTE CONTROL
Yes
214
PID CONTINUES TO POLL FOR DEVICES
216
PID DETECTS NEW DEVICES?
DEVICE LEFT ENVIRONMENT
No
No
218
Yes
220
PID HIDES CONTROL OF DEVICE
END
222

FIG. 7

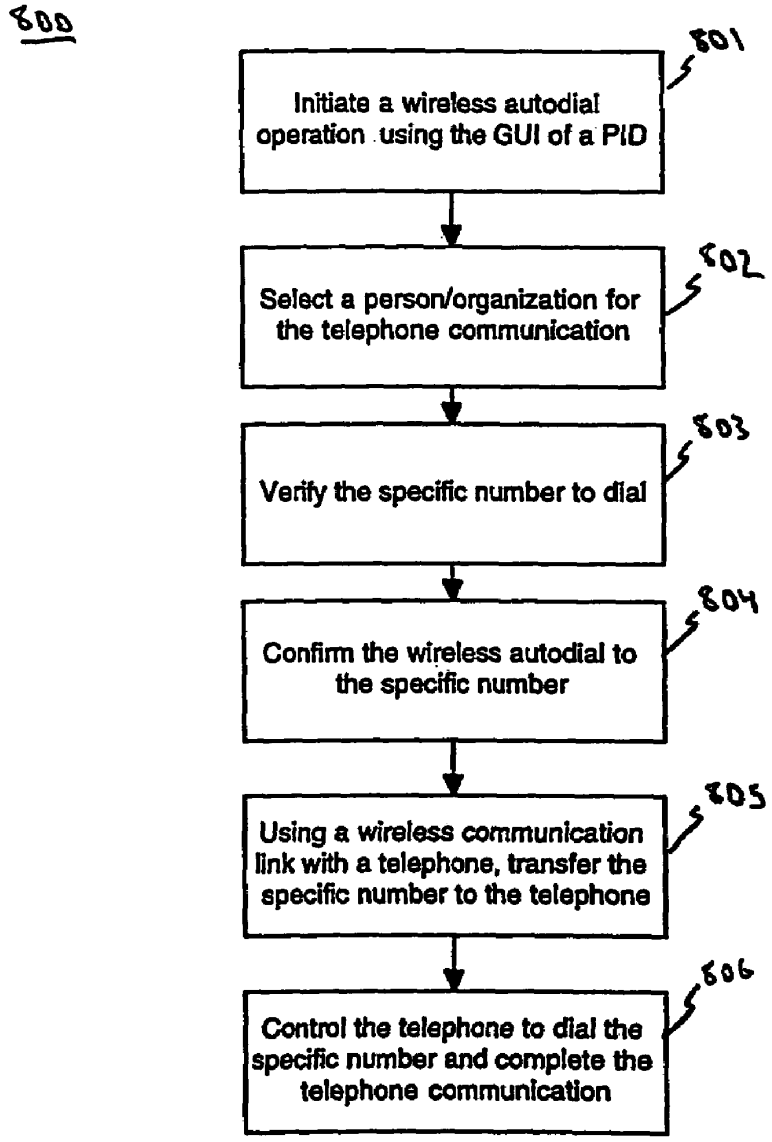

800
801
Initiate a wireless autodial operation using the GUI of a PID
802
Select a person/organization for the telephone communication
803
Verify the specific number to dial
804
Confirm the wireless autodial to the specific number
805
Using a wireless communication link with a telephone, transfer the specific number to the telephone
806
Control the telephone to dial the specific number and complete the telephone communication

FIG. 8

METHOD AND SYSTEM FOR WIRELESSLY AUTODIALING A TELEPHONE NUMBER FROM A RECORD STORED ON A PERSONAL INFORMATION DEVICE

TECHNICAL FIELD

The present invention relates to a system and method by which a personal information device ("PID") may directly control the operation functions of a mobile telephone.

BACKGROUND ART

Personal Information Devices include the class of computers, personal digital assistants and electronic organizers that tend both to be physically smaller than conventional computers and to have more limited hardware and data processing capabilities. PIDs include, for example, products sold by Palm, Inc. of Santa Clara, Calif., under such trademark as Pilot, and Pilot 1000, Pilot 5000, PalmPilot, PalmPilot Personal, PalmPilot Professional, Palm, and Palm III, Palm V, Palm VII, as well as other products sold under such trade names as WorkPad, Franklin Quest, and Franklin Convey.

PIDs are generally discussed, for example, in U.S. Pat. Nos. 5,125,0398; 5,727,202; 5,832,489; 5,884,323; 5,889,888; 5,900,875; 6,000,000; 6,006,274; and 6,034,686, which are incorporated herein by reference. PIDs typically include a screen and data processor, allowing the PID user to operate a substantial variety of applications relating to, for example: electronic mail, a calendar, appointments, contact data (such as address and telephone numbers), notebook records, expense reports, to do lists, or games. PIDs also often include substantial electronic memory for storing such applications as well as data entered by the user. Due to their substantial variety of applications and uses, personal information devices are becoming increasingly widely used.

Another widely used handheld digital device is the cellular telephone, or simply referred to as the cellphone. Cellphones are portable handheld devices that share a number of attributes of the PIDs. Cellphones, of course, include microphones and speakers with which to receive and send sound signals, typically the user's voices on either end of a connection. When in association with a modem or other such devices, the cellphone also may send and receive data signals, (including audio data signals). Many cellphones often include a screen in which to display information. However, cellphones typically have, or can accommodate, substantially fewer applications than many PIDs and user's find them much more difficult to use entering data, such as names and phone numbers, than PIDs. Further, some cellphones have less available memory for storing data signals than of many PIDs.

Because of this more limited functionality, cellphones are typically used for communication as opposed to personal information management (e.g., maintaining and updating appointments, contact lists, addresses, telephone numbers, and the like). Although many modern cellphones include computer resources to run some form of contact or telephone number lists, most users manage such information using other resources, such as, for example, a PID, and access cellphone (e.g., dial the correct number) once the proper telephone number is obtained.

There is a problem with this approach in that telephone numbers can be rather lengthy to dial, especially considering the small keypads on some cellphones. If the user makes a mistake while dialing, but doesn't notice the mistake untill

the call goes through, the user must typically terminate the call (e.g., hang up) and start all over again. In addition, the user must devote attention to devices, for example, obtaining the correct telephone number from a PID and then correctly punching in the number on the keypad of the cellphone. One prior art solution to this problem involves the use an automatic audible-tone dialer in conjunction with the cellphone. However, audible-tone dialers have met with only limited success in the marketplace and do not work on all cellphones.

Accordingly, what is required is a method whereby a user's handheld PID can automatically dial a telephone number stored in its memory. What is required is a solution that allows applications executed on the user's PID, such as, for example, an address book program, to access the user's telephone and automatically dial members stored in the program. What is further required is a solution which enables a user's PID to seamlessly interact with the user's telephone to dial members and establish phone calls without requiring the user to access controls of the telephone. The present invention provides a novel solution to the above requirements.

SUMMARY OF THE INVENTION

The present invention provides a method whereby a user's handheld PID can automatically dial a telephone number stored in its memory. The present invention provides a solution that allows applications executed on the user's PID, such as, for example, an address book program, to access the user's telephone and automatically dial members stored in the program. The present invention further provides a solution which enables a user's PID to seamlessly interact with the user's telephone to dial members and establish phone calls without requiring the user to access controls of the telephone.

In one embodiment, the present invention is implemented as an automated telephone dialing system. The system enables a handheld PID to automatically dial a telephone number stored its memory by interacting with a telephone. The telephone (e.g., a handheld cellular telephone) is equipped with a wireless port for short range wireless data transfer. Similarly, the PID is equipped with a wireless port for short range wireless data transfer. The PID establishes a wireless communication with the telephone. The PID is configured to control the telephone via the wireless communications such that the telephone dials a telephone number stored on the PID. The telephone number can be dialed in response to the user interacting with an application executing on the PID. The application can be a contact management or address management program. The user can interact with the program, select a contact, address, phone number, or the like, through a GUI of the PID, and have this number automatically dialed by the telephone.

In this manner, the user's PID seamlessly interacts with the user's telephone to dial numbers and establish phone calls without requiring the user to access controls of the telephone. The wireless communication between the PID and the telephone can be compatible with a version of the Bluetooth specification. The wireless communication between the PID and the telephone can also be compatible with a version of the IrDA specification.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not by way of limitation in the figures of the accompa-

nying drawings, in which like reference numerals refer to similar elements, and in which:

FIG. **1** is a diagram illustrating an exemplary preferred embodiment of the present system.

FIG. **2** is a block diagram illustrating the layers of a radio frequency protocol stack used in the PID of FIG. **2**.

FIG. **3** shows a stack layer diagram illustrating the layers of an RF protocol stack in accordance with one embodiment of the present invention.

FIG. **4** is a stack layer diagram illustrating layers of an Infrared Data Association protocol stack used in the PID of FIG. **2**.

FIG. **5** is a block diagram of the system of FIG. **1**.

FIG. **6**A shows a diagram of a first GUI dialog box in accordance with one embodiment of the present invention.

FIG. **6**B shows a second GUI dialog box in accordance with one embodiment of the present invention.

FIG. **6**C shows a third GUI dialog box in accordance with one embodiment of the present invention.

FIG. **6**D shows a fourth GUI dialog box in accordance with one embodiment of the present invention.

FIG. **7** is a flowchart illustrating on exemplary method for a PID in accordance with one embodiment of the present invention to establish a wireless link.

FIG. **8** is a flowchart illustrating the steps of an automatic wireless dialing process in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

In the following detailed description of the present invention, a method and system for wirelessly autodialing a telephone number from a record stored on a personal information device, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it will be obvious to one skilled in the art that the present invention may be practiced without these specific details. In other instances well known methods, procedures, components, and circuits have not been described in detail as not to obscure aspects of the present invention unnecessarily.

Some portions of the detailed descriptions which follow are presented in terms of procedures, logic blocks, processing, and other symbolic representations of operations on data bits within a computer memory. These descriptions and representations are the means used by those skilled in the data processing arts to convey most effectively the substance of their work to others skilled in the art. A procedure, logic block, process, step, etc., is here, and generally, conceived to be a self-consistent sequence of steps or instructions leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated in a computer system. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussions, it is appreciated that throughout the present invention, discussions utilizing terms such as "communicating" or "implementing," "transferring," "executing," "controlling," "configuring," "storing," "initializing," or the like, refer to the actions and processes of an embedded computer system, or similar embedded electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

The present invention is directed towards a method whereby a user's handheld PID can automatically dial a telephone number stored in its memory. The present invention provides a solution that allows applications executed on the user's PID, such as, for example, an address book program, to access the user's telephone and automatically dial members stored in the program. The present invention further provides a solution which enables a user's PID to seamlessly interact with the user's telephone to dial members and establish phone calls without requiring the user to access controls of the telephone. Embodiments of the present invention and its benefits are further described below.

FIG. **1** shows an exemplary embodiment of a system **10** in accordance with one embodiment of the present invention. The system **10** includes a hand-held PID **12** and a mobile telephone **14**. In one exemplary embodiment, the mobile telephone **14** is a hand-held cellular telephone.

As described above, the preferred embodiment utilizes a PID communicatively coupled to a mobile telephone. However, many electric devices, such as digital cameras, pagers, limited capability laptop computers, and the like, are similar to many PIDs in that they are unaware of the scheduling information contained within a user personal information device. Such limited-feature devices may also be enhanced by coupling the devices with a personal information device in accordance with present invention to implement the functionality for automatically updating the operating modes of the devices.

As shown in FIG. **1**, the PID **12** of the present system **10** includes a wireless port, or transceiver, **16** (used herein to mean some combination of a receiver and/or transmitter). The telephone **14** has a corresponding wireless port, or transceiver, **18** such that a wireless link **20** is established between the telephone **14** and PID **12**. The telephone **14** further includes a microphone **22** and speaker **24**.

In one preferred embodiment, the wireless ports **16**, **18** each include a short-range radio frequency ("RF") transceiver. The wireless transceiver **16**, **18** establish an RF link, such as that defined by the Bluetooth communications specification. However, the link **20** can also take other forms, including an infrared communication link such as that as defined by the Infrared Data Association (IrDA).

FIG. **2** is a function block diagram showing an exemplary embodiment of the PID **12** that can communicate with the telephone **14** or other such devices. The link interface circuitry **26** illustrates, but is not limited to, two alternative link interfaces for establishing a wireless link to another device. One wireless link interface (or more than two link interfaces) may, of course, be used with the present system **10**.

The PID **10** includes a processor, or controller, **28** that is capable of executing an RF stack **30** and an IrDA stack **32**. The stacks **30**, **32** communicate with data interface circuitry **26** through a bus **34**. The processor **28** is also connected through the bus **34** to user interface circuitry **36**, a data storage module **38** and memory **40**. As used herein, the data

storage module **38** and memory **40** may both generally be referred to as part of the PID memory **41**.

The memory **40** may contain a specific remote control loading application **42**. The remote control loading application **42** may operate, for example, after the processor **28** receives a message for the user to establish a wireless link with the telephone **14** in the nearby environment. Alternatively, the remote control loading application **42** may operate in a PID default mode.

The data interface circuitry **26** includes, in this exemplary embodiment, a first and second port, such as, infrared and RF interface ports. The first wireless link interface, the RF link interface, may include first connection **44** which includes radio-frequency (RF) circuitry **46** for converting signals into radio-frequency output and for accepting radio-frequency input. The RF circuitry **46** can send and receive RF data communications via a transceiver that is part of the communication port **16**. The RF communication signals received by the RF circuitry **46** are converted to electrical signals and relayed to the RF stack **30** in processor **28** via the bus **34**.

The mobile telephone **14** includes a corresponding port, or transceiver, **18** for RF signals. Thus, the RF **24** and wireless link **20** between the PID **12** and telephone **14** may be implemented according to the Bluetooth specification, described at www.bluetooth.com, which is incorporated in its entirety into this document.

Bluetooth is the name for a short-range radio link intended to replace the cable(s) connecting portable and/or fixed electronic devices. Bluetooth technology features low power, robustness, low complexity and low cost. It operates in the 2.4 Ghz unlicensed ISM (Industrial, Scientific and Medical) band. Devices equipped with Bluetooth are capable of exchanging data at speeds up to 720 kbps at ranges up to 10 meters. It should be noted that higher power devices other than the typical Bluetooth enabled personal information device, such as, for example, a network access point, may communicate via Bluetooth with an RF-enabled PID over a greater range, such as, for example, approximately 100 meters.

A frequency hop transceiver is used to combat interface and fading. A shaped, binary FM modulation is applied to minimize transceiver complexity. A slotted channel is applied with a nominal slot length of 625 μs. For full duplex transmission, a Time Division Duplex scheme is used. On the channel, information is exchanged through packets. Each packet is transmitted in a different hop frequency. A packet nominally covers a single slot, but can be extended to cover up to five slots.

The Bluetooth protocol uses a combination of circuit and packet switching. Slots can be reserved for synchronous packets. Bluetooth can support an asynchronous data channel, up to three simultaneous voice channels, or a channel that simultaneously supports asynchronous data and synchronous voice. Each voice channel supports a 64 kb/s synchronous (voice) channel in each direction. The asynchronous channel can support maximum 723.2 kb/s asynchronous, or 433.9 kb/s symmetric.

The Bluetooth system consists of a radio unit, a link control unit, and a support unit for link management and host terminal interface functions. The link controller carries out the baseboard protocol and other low-level routines.

The Bluetooth system also provides a point-to-point connection (only two Bluetooth units involved) or a point-to-multipoint connection. In the point-to-multipoint connection, the channel is shared among several Bluetooth units. Two or more units sharing the same channel form a piconet. One Bluetooth unit acts as the master of the piconet, whereas the other units act as slaves. Up to seven slaves can be active in a piconet.

The Bluetooth link controller has two major states: STANDBY and CONNECTION. In addition, there are seven substances: page, page scan, inquiry, inquiry scan, master response, slave response, and inquiry response. The substances are interim states that are used to add new slaves by piconet.

The STANDBY state is the default state in the Bluetooth unit. In this state, the Bluetooth unit is in a low-power mode. The controller may leave the STANDBY state to scan for page or inquiry messages, or to page or inquiry itself When responding to a page message, the unit enters the CONNECTION state as a master.

In order to establish new connections, the inquiry procedures and paging are used. The inquiry procedures enable a unit to discover which units are in range, and what their device address and clocks are during an inquiry substrate. The discovering unit collects the Bluetooth device addresses and clocks of all units that respond to the inquiry message. It can then, if desired, make a connection to any one of them. The inquiry message broadcasted by the source does not contain any information about the source. However, it may indicate which class of devices should respond.

There is one general inquiry access code (GIAC) to inquire for any Bluetooth device, and a number of dedicated inquiry access codes (DIAC) that only inquire for a certain type of devices. A unit that wants to discover other Bluetooth units enters an inquiry substance. In this substance, it continuously transmits the inquiry message (which is an identification packet) at different hop frequencies. A unit that allows itself to be discovered, regularly enters the inquiry scan substance to respond to inquiry messages.

As noted above, the system **10** may operate with only one wireless link. However, additional, or alternative, wireless link interfaces may also be included in the system **10**. A second connection **46** includes infrared circuitry **48** for converting signals into infrared output and for accepting infrared input. Thus, the wireless link **28** may include an infrared interface. The infrared circuitry **48** can send and receive infrared data communications via the port, or transceiver, **16**.

Infrared communication signals received by infrared circuitry **48** are converted into electrical signals that are relayed to the IrDA stack **32** in the processor, or controller, **28** via the bus **34**. The telephone **14** may include a corresponding infrared transceiver. The infrared circuitry **48** operates according to the IrDA specifications available at www.IrDA.org.

The two, alternative link interfaces described above are merely exemplary, and additional means for implementing the interface between a PID and telephone or other such device may also be utilized. Although not required, more than one wireless link interface may be included to improve flexibility and to provide redundancy in case of failure of one of the link interfaces.

User interface circuitry **36** in the PID **12** includes hardware and software components that provide user input and output resources for functions in the processor **28**. The user interface circuitry **36** includes display output **50**, display input **52**, and additional input/output interface circuitry **54**.

The display output **50** preferably receives digital information representing graphical data from the processor **28** and converts the information to a graphical display, such as text and or/images, for display on a display screen. The display input **52** may receive data inputs, such as graphical

data inputs, from a user of the PID **12**. The graphical data inputs are preferably entered by the user with a stylus on a pressure sensitive display screen, and may include text, drawings, or other objects that are capable of being graphically presented.

Typically, the additional input/output interface **54** permits user input and commands to be input through buttons and similar devices on the PID, e.g., buttons for scrolling through data entries and activating applications. Alternatively, the input/output interface **54** may allow the PID **12** to accept audio data as well as other types of non-graphical data. For example, audio data signals (or picture telephone video input) may be entered through the additional input/output interface **54**.

FIG. **3** shows a diagram illustrating the layers of the Bluetooth (RF) protocol stack **60** in accordance with one embodiment of the present invention. An RF protocol stack is implemented at each end of the connection endpoints of an RF link. For example, a PID **12** and a telephone **14** could each implement an RF stack to enable a link. The required layers of the RF link using the Bluetooth system are the Baseband layer **62**, the Link Manager Protocol Layer (LMP) **64**, the Logical Link Control and Adaptation Layer **68**, RFCOMM Layer **70**, Service Discovery Protocol Layer **72**, and Object Exchange Protocol (OBEX) layer **74**.

FIG. **4** is a protocol diagram **80**, illustrating the layers of the IrDA protocol stack that may be used with the system **10**. For example, the PID and the telephone **41** each implement an IrDA protocol stack to enable the wireless link **20**.

The required layers of an IrDA protocol stack are the physical layer **82**, the IrLMP layer **84**, the IrLMP layer **86** and the IAS layer **88**. The physical layer **82** specifies optical characteristics of the link, encoding of the data, and framing for various speeds. The IrLAP (Link Access Protocol) layer **84** establishes the basic reliable connection between the two ends of the link. The IrLMP (Link Management Protocol) layer **86** multiplexes services and applications on the IrLAP connection. The IAS (Information Access Service) layer **88** provides a directory of services on an IrDA device.

The IrDA protocol also specifies a number of optional protocol layers, these protocol layers being TINY TP90, IrOBEX 92, IrCOMM 94 AND IrLAN 96. TINY TP (Tiny Transport Protocol) 90 adds per-channel flow control to keep traffic over the link **20** moving smoothly. IrOBEX (Infrared Object Exchange Protocol) 92 provides for the easy transfer of files and other data objected between the IrDA devices at each end of the applications that use serial and parallel communications to use IrDA without change. IrLAN (Infrared Object Exchange Protocol) 92 provides for the easy transfer of files and other data objects between the IrDA devices at each end of the link **20**. IrCOMM 94 is a serial and parallel communications to use IrDA without change. IrLAN (Infrared Local Area Networks) 96 enables walk-up infrared LAN access.

The use of the optional layers depends upon the particular application in the IrDA device. The IrDA protocol stack is defined by such standard documents as "IrDA Serial Infrared Physical Layer Link Specification", "IrDA 'IrCOMM': Serial and Parallel Port Emulation over IR (wire replacement)", "IrDA Serial Infrared Link Access Protocol (IrLAP)", "IrDA Infrared Link Management Protocol (IrLMP)", and "IrDA 'TINY TP": A Flow-Control Mechanism for use with IrLMP,: and related specifications published by the IrDA. Such documents are available at www.irda.org/standards/specifications.asp and are incorporated in their entirety in this document.

As shown in FIG. **5**, the PID **12** may include resident applications **100**, such as, for example, a contact management program **101** for managing contact information. The PID **12** may include as well, for example, an address book program **109** for managing address and telephone number information, a calendar program **102** for assisting in managing scheduling and events, and a user preferences program **104** for configuring personal information device **12** in accordance with the requirements of the user.

As described above, PID **12** can accept input from a user, such as selecting a specific contact using contact management program **101**, and automatically dial a telephone number stored in its memory via a wireless communication with telephone **14**. The wireless link **20** enables applications executed on PID **12** (e.g., address book program **109**) to access the telephone **14** and automatically dial the number stored in the application (e.g., within memory **40**). The wireless link **20** enables an application executing on PID **12** to access telephone **14**, communicate the desired telephone number, and control telephone **14** to dial the number and established the telephone call. In the present embodiment, the interactions are seamless, wherein the telephone **14** is controlled by PID **12** without requiring any intervening steps or actions by the user with any controls of telephone **14** (e.g., punching any keys on a keypad). The user can interact with the program (e.g., address book program **109**), select a person, or the like, through the program's GUI, and have this number automatically dialed by the telephone **14**. The wireless communication link **20** between the PID **12** and the telephone **14** can be an RF based Bluetooth link or and IR based IrDA link.

Referring now to FIG. **6**A through FIG. **6**D, a user interaction process with the GUI of an application in accordance with one embodiment of the present invention is shown. FIGS. **6**A–**6**D depict an example process through which a user has a telephone number stored within, for example, address book program **109** in PID **12** that will be automatically dialed by telephone **14**.

FIG. **6**A shows an address list as presented by address book program **109** on PID **12**. For example, address book **109** executes a "lookup" feature to call up contact information for various people/organizations stored within memory. The user can then select the person/organization to place a call to. In this example, the user has selected "Bill Boomer" by, for example, tapping on the entry with a stylus. In response to the selection, address book program **109** then presents information associated with the selection made in FIG. **6**A. The user can then confirm the correct selection is presented in the GUI, and then select the appropriate button for the specific telephone number to dial (e.g., work, home, fax, or pager). Then, as shown in FIG. **6**C, the user can once again the confirm the correct number is presented in the GUI, and select "OK" to initiate the automatic dialing process. If the number shown is incorrect, the user can modify the prefix by, for example, removing or altering the area code, or correcting any of the other digits of the number, or cancel the entire process. Subsequently, as shown in FIG. **6**D, the automatic dialing process begins, with PID **12** setting up a communications link with telephone **14** for the execution of the autodial. The communication can be IR based or RF based, depending upon the last such communication, previous selection, default setting, etc.

FIG. **7** illustrates a process by which the PID **12** detects and communicates with another device, such as the telephone **14** that is in its environment (or other additional mobile computing devices within range) using standard RF protocols (e.g., Bluetooth). The process begins at Step **200**

when either the user manually initiates the process or the process is implemented as a default. At step **202**, the PID **12** sends out a general query to determine whether a corresponding telephone is present in the nearby environment (for example, in the room). According to Bluetooth terminology, this step is analogous to the process of discovery.

At Step **204**, the telephone **14** responds to the query with its unique identifier using, but not limited to, the discovery resins protocol defined by the Bluetooth specification. The PID **12** then loads the controls of each device using at least two methods. If it is determined at Step **206** that the PID **12** has interfaces with the telephone **14** before and the controls had previously been downloaded, then, at step **208**, the PID **12** accesses the controls from its memory. However, if it is determined at Step **206** that the PID **12** has not interfaced with the telephone **14** before, then, at Step **201**, the PID **12** connects with the telephone **14** and negotiates a control protocol. As an alternative, the PID **12** may be manufactured with pre-stored controls for certain devices. These controls may be accessed whenever the PID **12** detects the respective devices or class of devices.

At Step **212**, the PID **12** presents a notification to the user, indicating that the link between the PID **12** and telephone **14** has been established. The PID **12** then continues to poll for devices at regular time intervals, as set out in Step **214**, querying the environment to determine if any devices, like the telephone **14**, have appeared or any such devices have left the PID environment. If the PID **12** detects a new device in its environment at Step **218**, the PID **12** hides the controls for the telephone **14** from the user and transfers the controls to a storage location, such as the memory **40** or data storage module **38** (Step **220**).

If the telephone **14** is still in the PID's environment, as determined at step **218**, then the process returns to Step **214** and continues. The process ends at Step **222** if the user disables the remote control loading applications. However, if the user does not disable the application, then the process returns to Step **214** and follows the process again.

FIG. **8** shows a flowchart of the steps of the wireless autodialing process **800** in accordance with one embodiment of the present invention. Process **800** shows the operating steps of the autodial process after a wireless link (e.g., wireless link **20**) has been established between the PID (e.g., PID **12**) and the telephone (e.g., telephone **14**) as described in FIG. **7**.

Process **800** begins in step **801**, where the user initiates a wireless autodial operation using the GUI of PID **12**. As described above, the user can access a large amount of information stored and maintained within the memory of PID **12**. The information is maintained by one or more applications resident within PID **12**, such as address book program **109** and contact manager program **101**. Upon initiation, the particular program can present a list of individuals/organizations from which the user can select the desired individual/organization to contact. In step **802**, from the list, the user selects the desired contact, for example, as depicted in FIGS. **6**A and **6**B above. In step **803**, after selection of the desired contact, the user verifies the specific number to dial (e.g., as shown in FIG. **6**C).

Referring still to FIG. **8**, in step **804**, the user confirms the wireless autodial to the specific number, for example, as shown in FIG. **6**D. In step **805**, using a wireless communication link (e.g., link **20** of FIG. **5**) with the telephone **14**, the PID **12** transfers the specific number to telephone **14**. Subsequently, in step **806**, PID **12** controls telephone **14** to dial the specific number and complete the telephone communication.

Thus, the present invention provides a method whereby a user's handheld PID can automatically instruct a cellphone to dial a telephone number stored in the PID's memory. The present invention provides a solution that allows applications executed on the user's PID, such as, for example, an address book program, to access the user's telephone and automatically dial members stored in the program. The present invention further provides a solution which enables a user's PID to seamlessly interact with the user's telephone to dial members and establish phone calls without requiring the user to access controls of the telephone.

It should be noted that the programs, processes, methods and systems described herein are not related or limited to any particular type of computer or network system (hardware or software), unless indicated otherwise. In view of the variety of embodiments to which the principles of the present invention can be applied, it should also be understood that the illustrated embodiments are exemplary only and should not be taken as limiting the scope of the present invention.

For example, the steps of the flow diagrams may be taken in sequences other than those described, and more or fewer elements may be used in the block diagrams. In addition, protocols of various types are referenced throughout. While preferred and alternate embodiments may implement selected protocols, any suitable replacement protocol not mentioned, or any function not part of a protocol used to replace a corresponding function from a protocol, may be implemented without departing from the scope of the invention. While various elements of the preferred embodiments have been described as being implemented in software, in other embodiments hardware or firmware implementations may alternatively be used, and vice-versa. Also, while the present invention has been described in the context of a mobile telephone in communication with a PID, the principles of the present invention may be applied to other combinations of devices, such as a PID in communication with a land-line telephone, without departing from the teachings of the present invention.

The foregoing descriptions of specific embodiments of the present invention have been presented for purposes of illustration and description. They are not intended to be exhaustive or to limit the invention to the precise forms disclosed, and obviously many modifications and variations are possible in light of the above teaching. The embodiments were chosen and described in order best to explain the principles of the invention and its practical application, thereby to enable others skilled in the art best to utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the claims appended hereto and their equivalents.

What is claimed is:

1. An automated telephone dialing system, comprising:

a telephone having a wireless port for short range wireless data transfer; and

a handheld computer system having a wireless port for communication with the wireless port on the telephone, wherein a specific telephone number is selectable from a list displayed on the handheld computer system and wherein the handheld computer system is operable to transfer the specific telephone number to the telephone using a wireless communication, and wherein the handheld computer system is configured to control the telephone via the wireless communication such that the telephone dials the specific telephone number.

2. The system of claim 1 wherein the dialing of the specific telephone number by the telephone is automatically effected in response to a user interacting with the information stored on the handheld computer system.

3. The system of claim 2 wherein the information stored in the handheld computer system includes contact information.

4. The system of claim 2 wherein the list is presented as a list of contacts and the telephone number dialed by the telephone corresponds to one of the contacts selected by the user.

5. The system of claim 2 wherein the information stored on the handheld computer system is maintained by a management program executing on the handheld computer system and the management program controls the telephone via the wireless communication.

6. The system of claim 5 wherein the management program is an address book program.

7. The system of claim 1 wherein the wireless communication is compatible with a version of the Bluetooth specification.

8. The system of claim 1 wherein the wireless communication is compatible with a version of the IrDA specification.

9. An automatic wireless telephone dialing method, comprising the steps of:

a) establishing a wireless communications link for a short range data transfer between a telephone and a handheld computer system;

b) receiving a user input identifying a specific telephone number from a list displayed on the handheld computer system;

c) transferring the specific telephone number from the handheld computer system to the telephone using a wireless communication; and

d) controlling the telephone using the handheld computer system to cause the telephone to dial the specific telephone number.

10. The method of claim 9 wherein step d) further includes the step of dialing the specific telephone number automatically in response to a user interacting with the information stored on the handheld computer system.

11. The method of claim 10 wherein the information stored in the handheld computer system includes contact information.

12. The method of claim 10 wherein the list is presented as a list of contacts and the telephone number dialed by the telephone corresponds to one of the contacts selected by the user.

13. The method of claim 10 wherein the information stored on the handheld computer system is maintained by a management program executing on the handheld computer system and the management program controls the telephone via the wireless communication.

14. The method of claim 13 wherein the management program is an address book program.

15. The method of claim 9 wherein the wireless communication is compatible with a version of the Bluetooth specification.

16. The method of claim 9 wherein the wireless communication is compatible with a version of the IrDA specification.

* * * * *

US006622018B1

(12) **United States Patent**
Erekson

(10) **Patent No.: US 6,622,018 B1**
(45) **Date of Patent: Sep. 16, 2003**

(54) **PORTABLE DEVICE CONTROL CONSOLE WITH WIRELESS CONNECTION**

(75) Inventor: **Rich Erekson**, Ogden, UT (US)

(73) Assignee: **3Com Corporation**, Santa Clara, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/558,413**

(22) Filed: **Apr. 24, 2000**

(51) **Int. Cl.**[7] **H04M 3/00**
(52) **U.S. Cl.** **455/420**; 455/419
(58) **Field of Search** 455/420, 456, 455/457, 445, 404, 75, 566, 575, 96, 99, 145; 342/457.1; 340/825.36, 825.37, 825.49

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,917,405 A * 6/1999 Joao 340/426
6,091,957 A * 7/2000 Larkins et al. 455/456
6,161,066 A * 12/2000 Wright et al. 701/36
6,339,706 B1 * 1/2002 Tillgren et al. 455/419

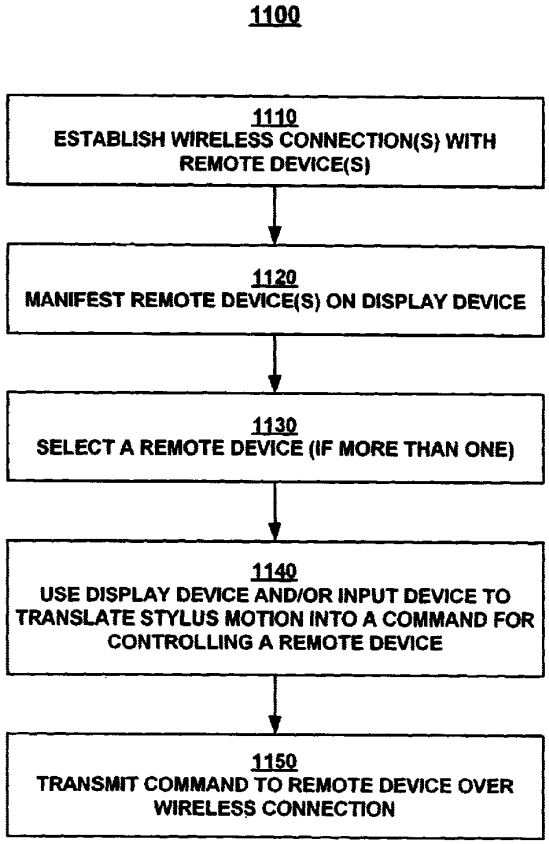
* cited by examiner

Primary Examiner—William Trost
Assistant Examiner—Naghmeh Mehrpour
(74) *Attorney, Agent, or Firm*—Wagner, Murabito & Hao LLP

(57) **ABSTRACT**

A system and method for controlling a remote device over a wireless connection. In one embodiment, a hand-held computer system having a Bluetooth-enabled transceiver is used to control other Bluetooth-enabled devices. A wireless connection between a transceiver and a remote device is established. A position where a stylus makes contact with a surface of an input device of the hand-held computer system is registered. The particular position where the stylus element makes contact with the input device is translated into a particular command for controlling the remote device. The command is then transmitted to the remote device over the wireless connection.

27 Claims, 12 Drawing Sheets



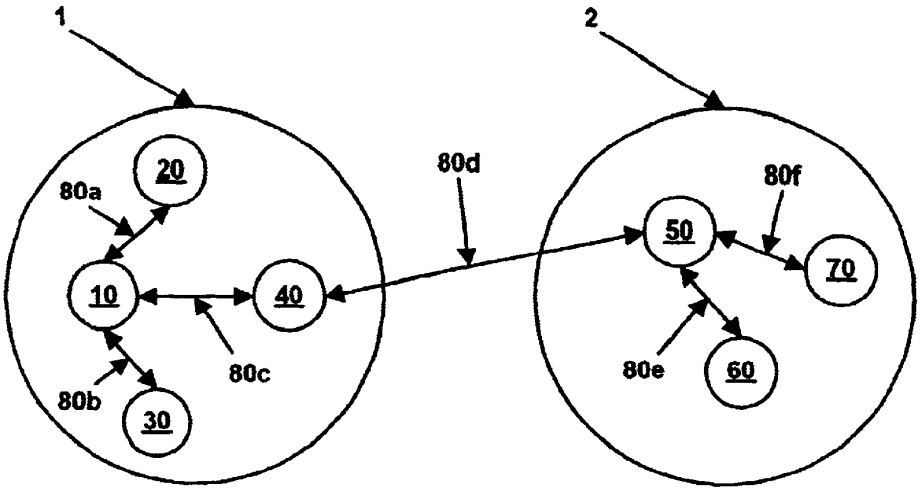


FIG. 1

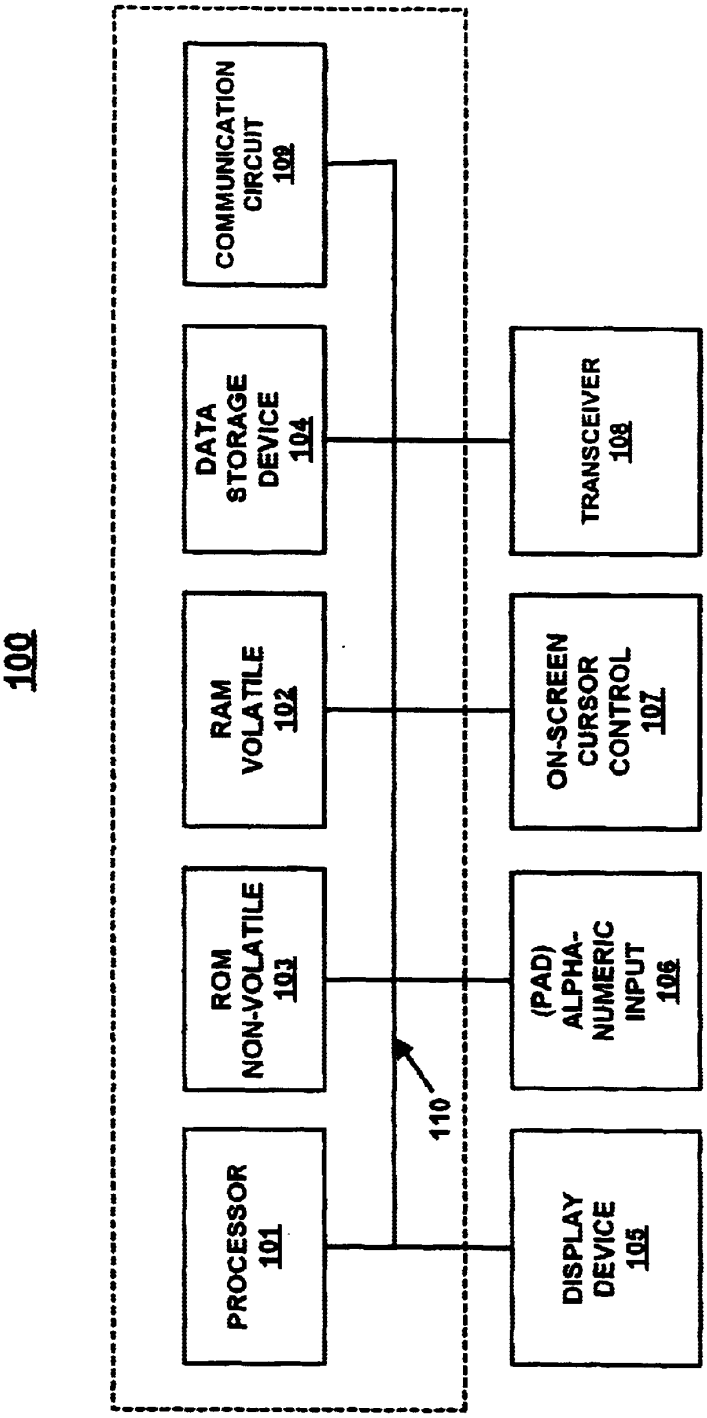

100
PROCESSOR 101
ROM NON-VOLATILE 103
RAM VOLATILE 102
DATA STORAGE DEVICE 104
COMMUNICATION CIRCUIT 102
110
DISPLAY DEVICE 105
(PAD) ALPHA-NUMERIC INPUT 106
ON-SCREEN CURSOR CONTROL 107
TRANSCEIVER 108

FIG. 2

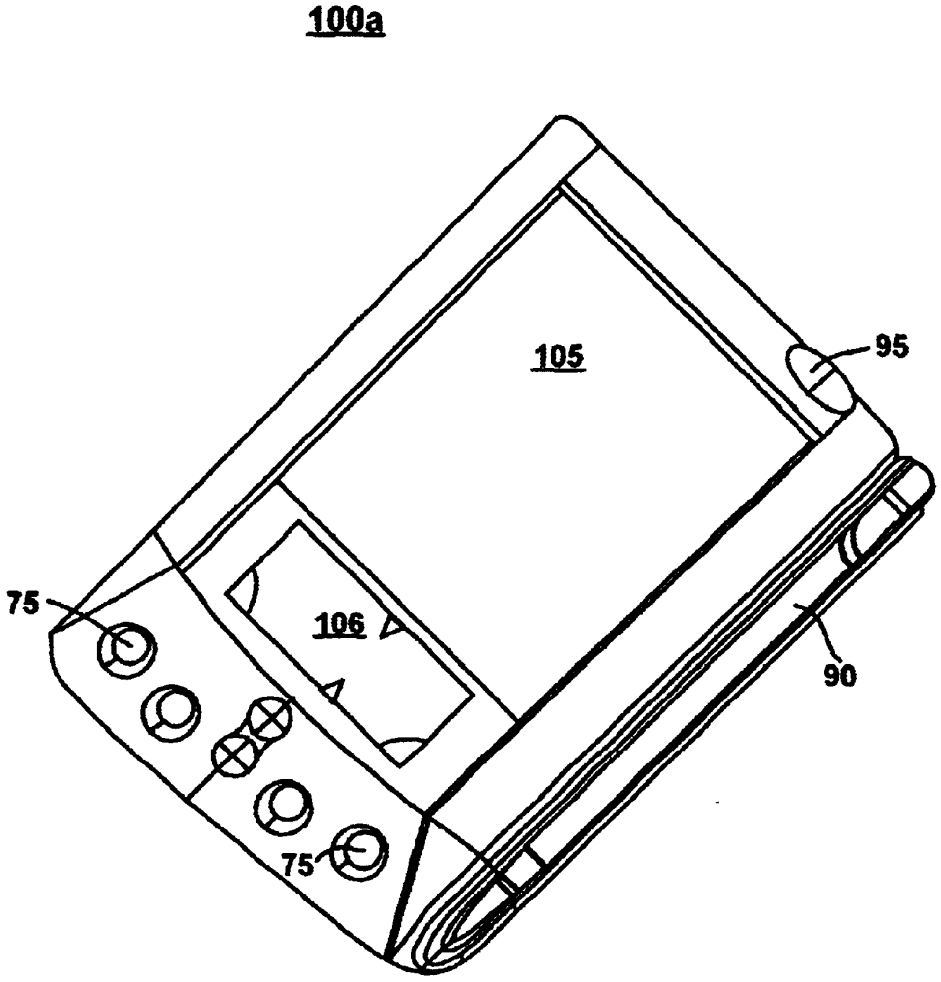


FIG. 3

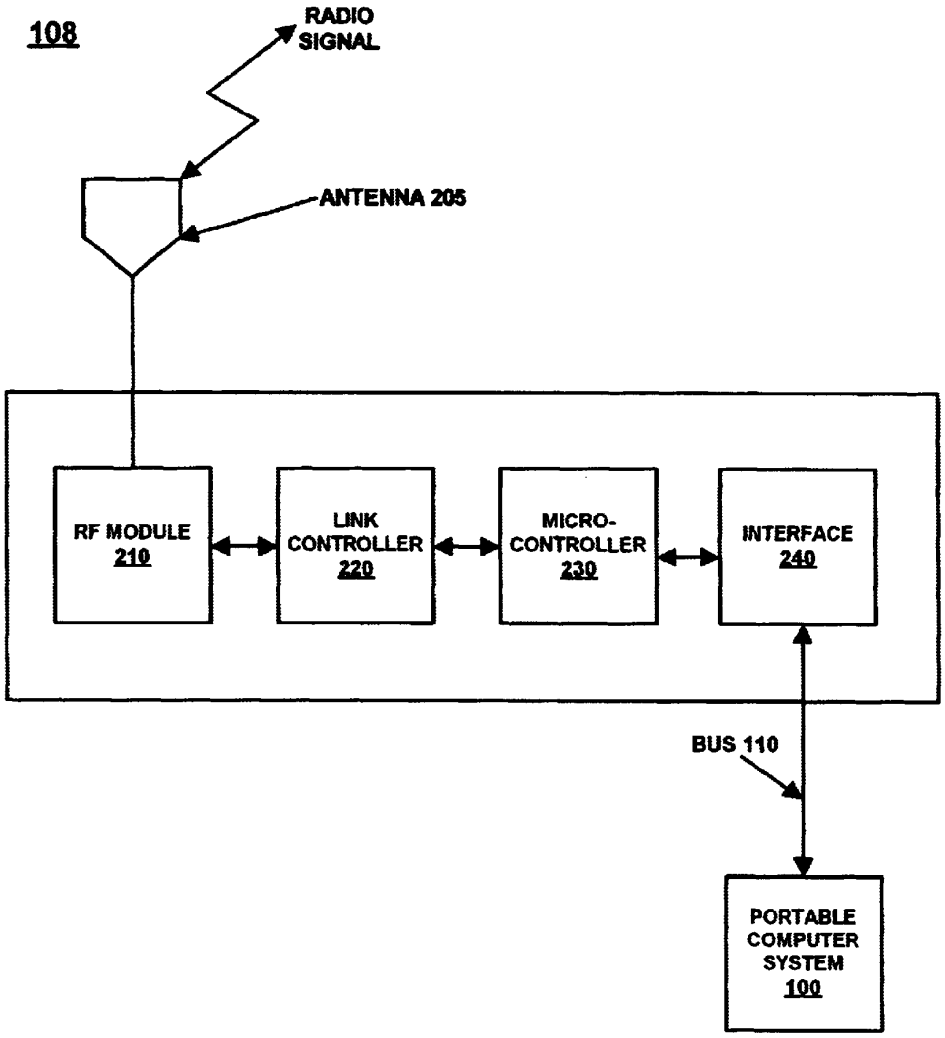

108
RADIO SIGNAL
ANTENNA 205
RF MODULE 210
LINK CONTROLLER 220
MICRO-CONTROLLER 230
INTERFACE 240
BUS 110
PORTABLE COMPUTER SYSTEM 100

FIG. 4A

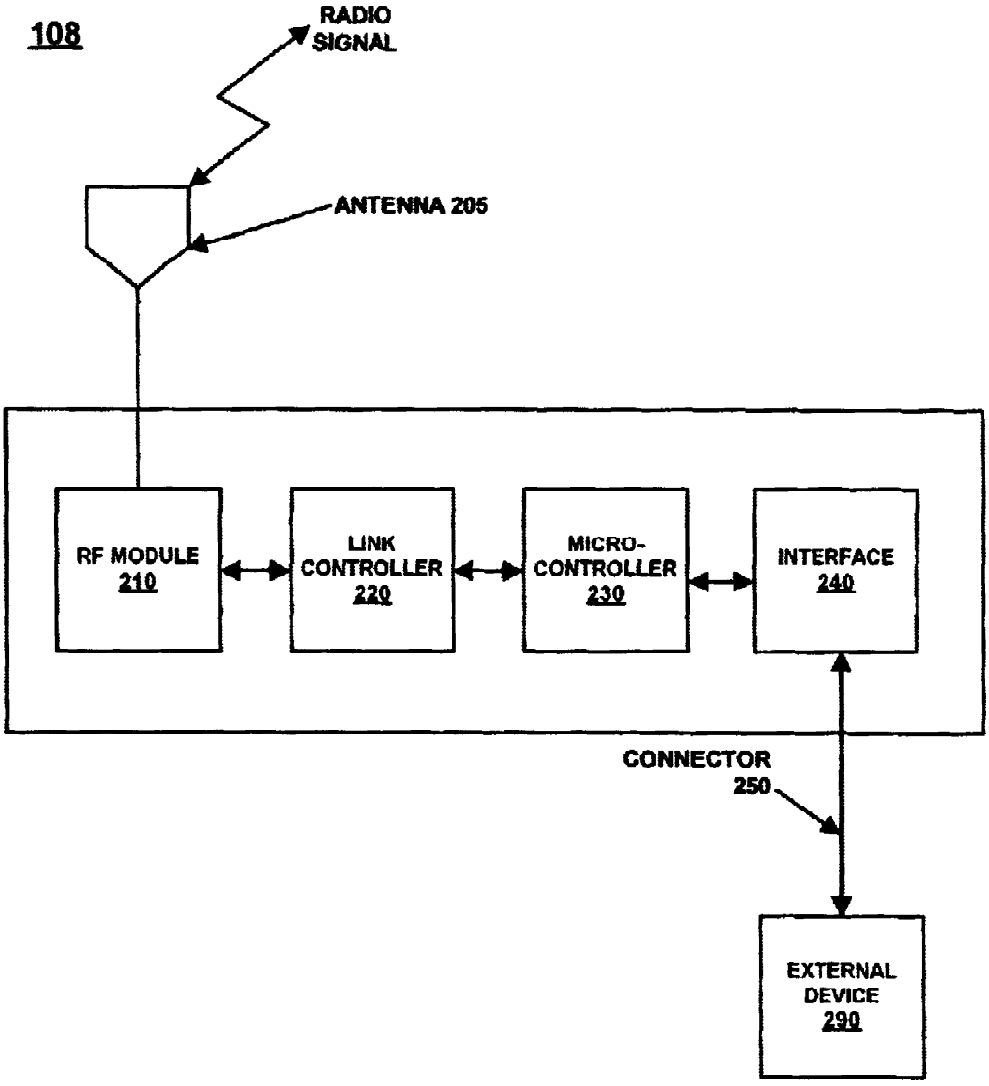

108
RADIO SIGNAL
ANTENNA 205
RF MODULE 210
LINK CONTROLLER 220
MICRO-CONTROLLER 230
INTERFACE 240
CONNECTOR 250
EXTERNAL DEVICE 290

FIG. 4B

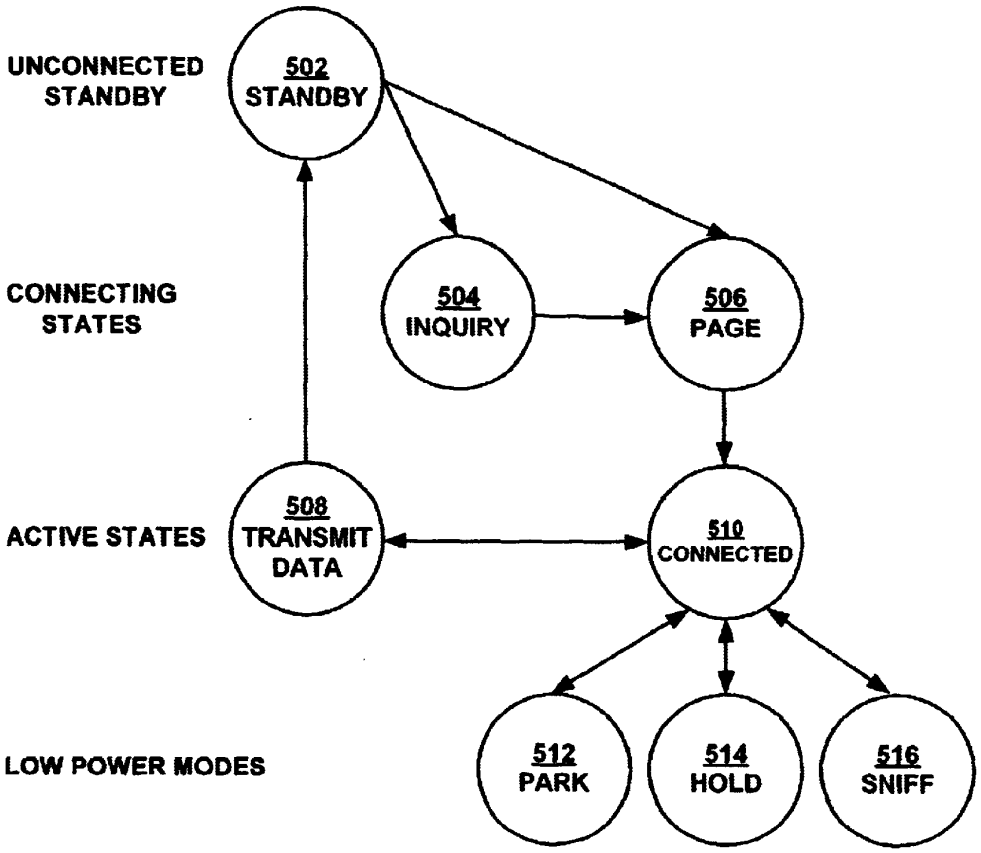

UNCONNECTED STANDBY
502 STANDBY
CONNECTING STATES
504 INQUIRY
506 PAGE
ACTIVE STATES
508 TRANSMIT DATA
510 CONNECTED
LOW POWER MODES
512 PARK
514 HOLD
516 SNIFF

FIG. 5

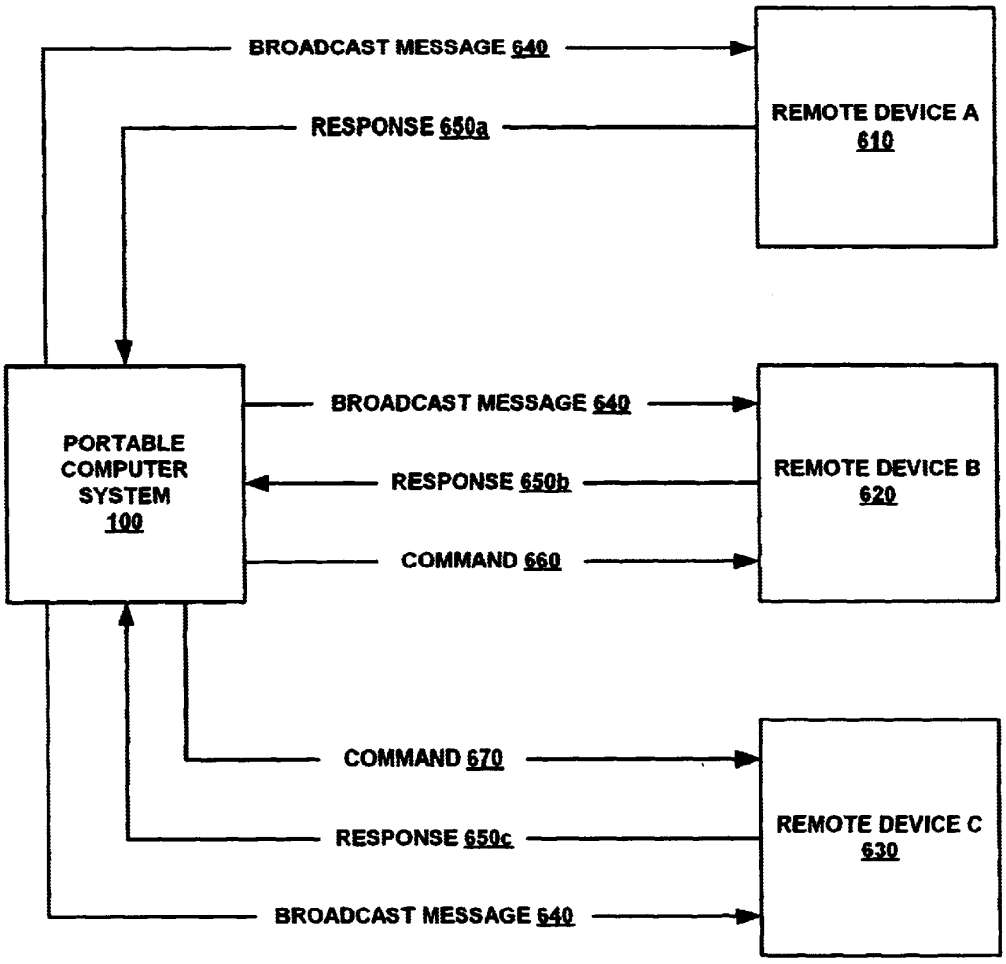

BROADCAST MESSAGE 640
RESPONSE 650a
REMOTE DEVICE A
610
PORTABLE
COMPUTER
SYSTEM
100
BROADCAST MESSAGE 640
RESPONSE 650b
COMMAND 660
REMOTE DEVICE B
620
COMMAND 670
RESPONSE 650c
BROADCAST MESSAGE 640
REMOTE DEVICE C
630

FIG. 6

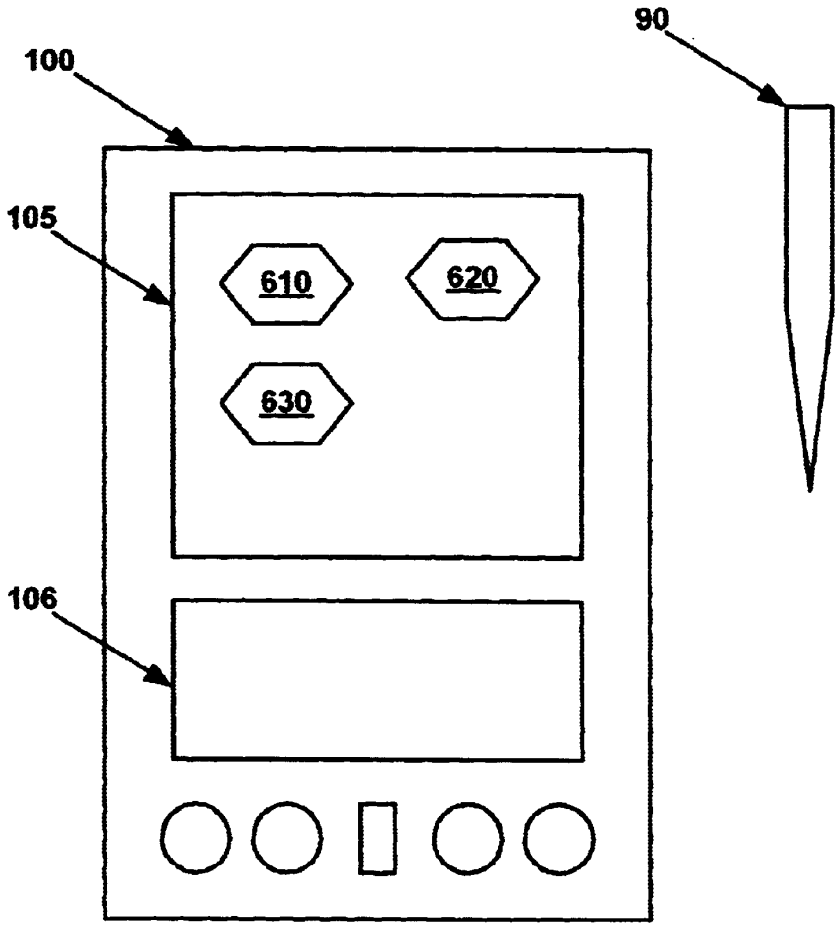


FIG. 7

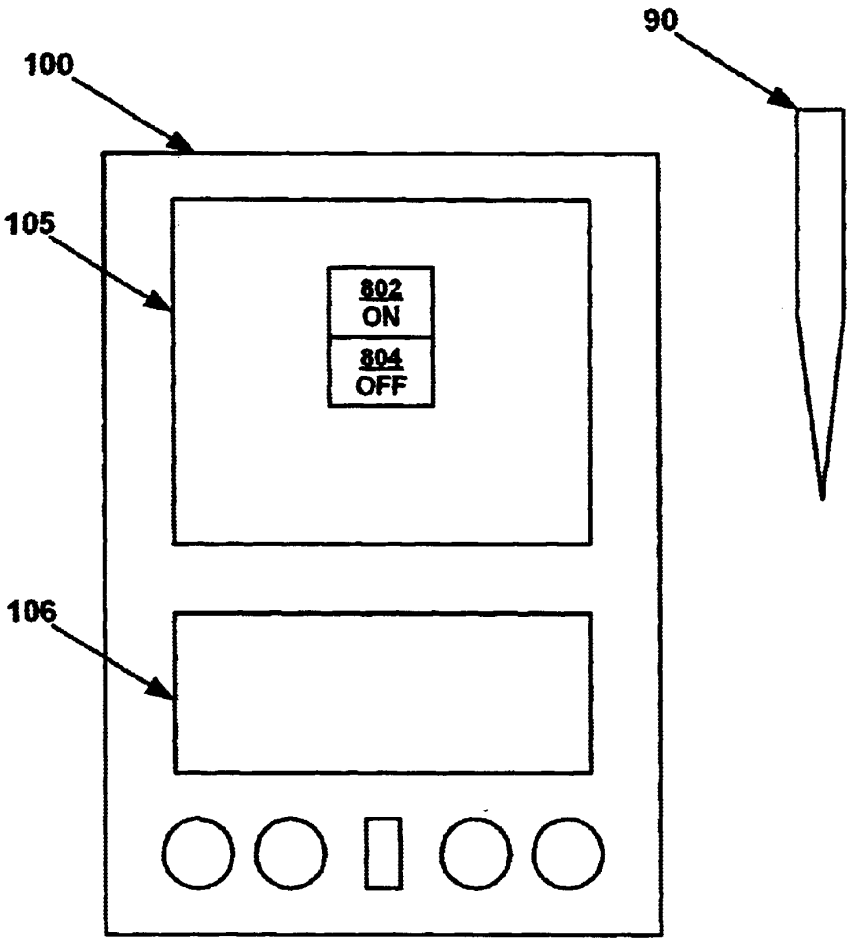


FIG. 8

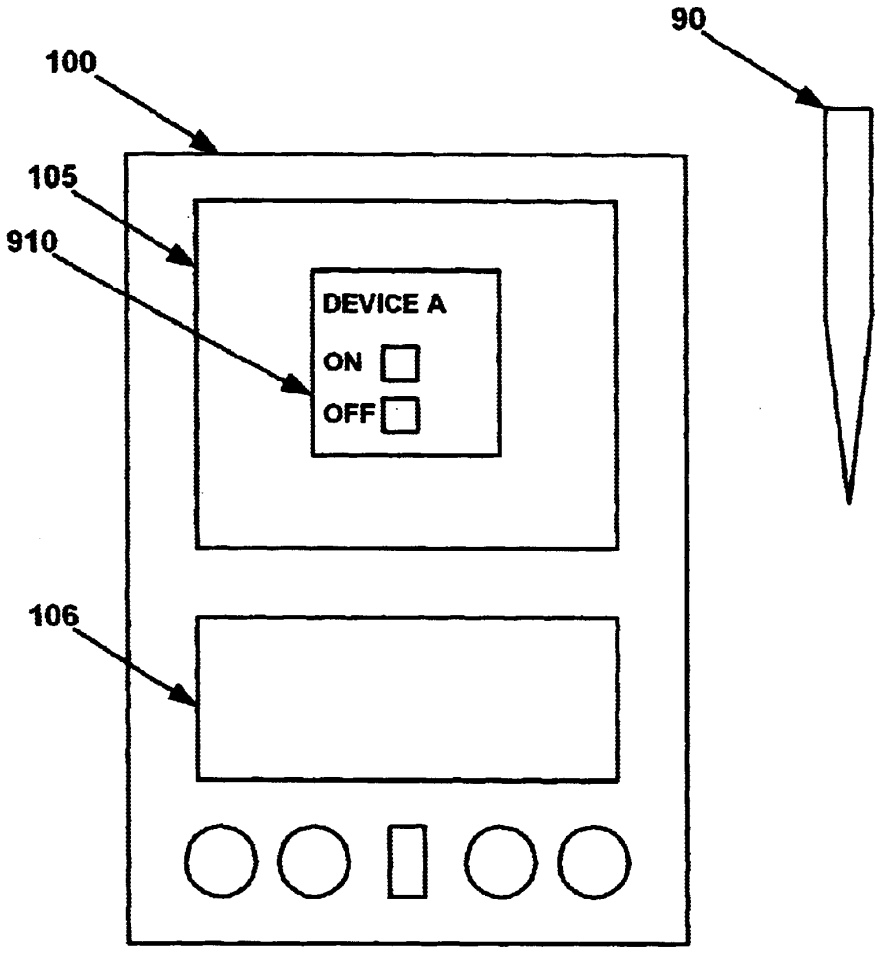


FIG. 9

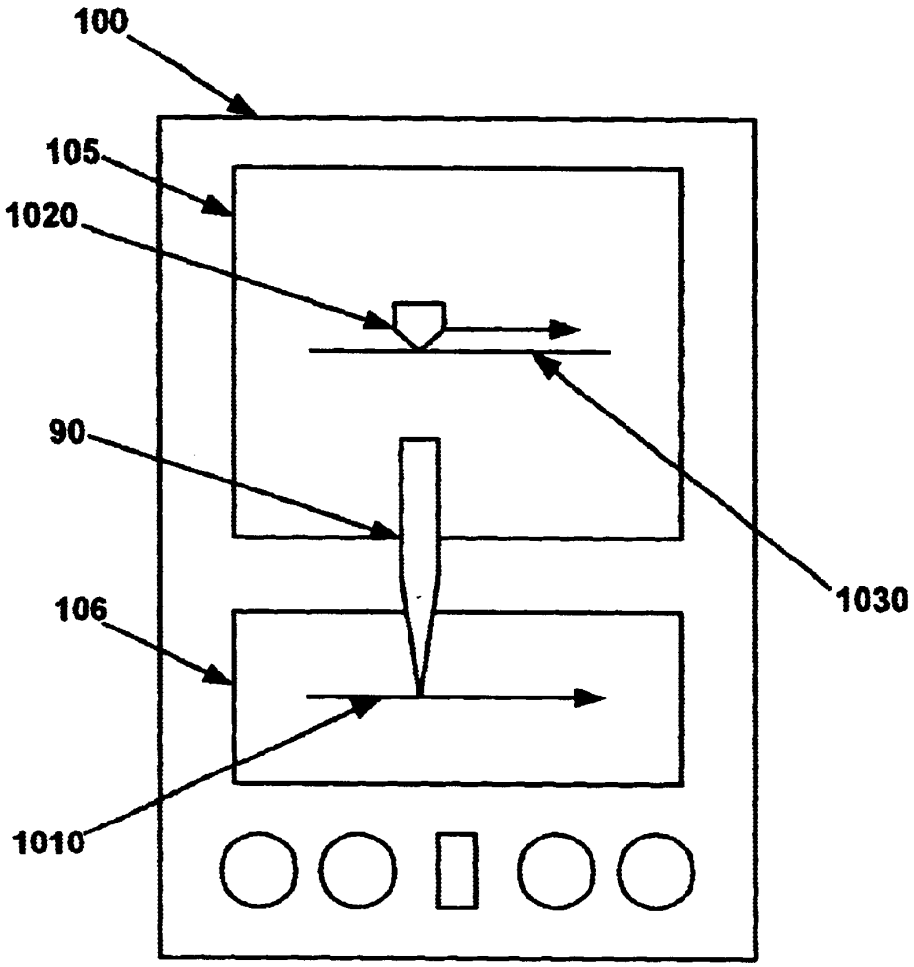


FIG. 10

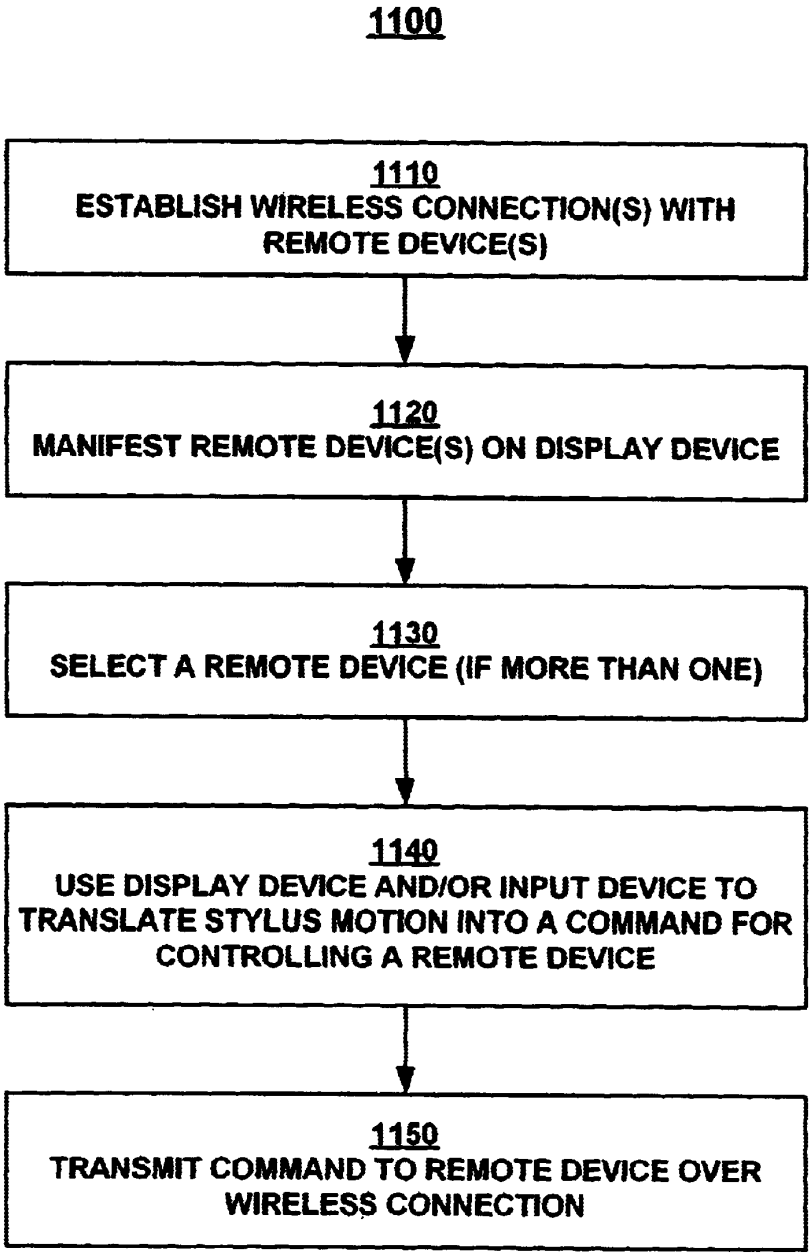

1100
1110
ESTABLISH WIRELESS CONNECTION(S) WITH REMOTE DEVICE(S)
1120
MANIFEST REMOTE DEVICE(S) ON DISPLAY DEVICE
1130
SELECT A REMOTE DEVICE (IF MORE THAN ONE)
1140
USE DISPLAY DEVICE AND/OR INPUT DEVICE TO TRANSLATE STYLUS MOTION INTO A COMMAND FOR CONTROLLING A REMOTE DEVICE
1150
TRANSMIT COMMAND TO REMOTE DEVICE OVER WIRELESS CONNECTION

FIG. 11

PORTABLE DEVICE CONTROL CONSOLE WITH WIRELESS CONNECTION

TECHNICAL FIELD

The present invention relates to systems and devices connected using wireless links, such as systems and devices that use the Bluetooth technology. In particular, the present invention pertains to a method and system for controlling remote devices over a wireless connection.

BACKGROUND ART

Consider, for example, the number of devices and appliances in the typical living room or family room of a residential dwelling: lamps, light switches, a thermostat, and consumer electronic devices such as televisions, video cassette recorders, and stereos, some of these devices themselves comprising multiple devices such as compact disk players, tape players, etc. Each of these devices requires manual interaction by a user in order to turn them off or on, to raise or lower levels, and so on. Other rooms of the house, as well as factories and places of business, also have countless devices and appliances that require manual interaction in order to use and control them.

Of course, many of these devices are or can be remotely controlled. In the home, for example, remote control devices for televisions, stereos and the like are very common. Devices for controlling lights, etc., are also available although generally not as commonly used.

Remote control devices in each of their present forms have a number of associated shortcomings. For example, a separate remote control device may be required for each device to be controlled. In some instances the separate remotes can be replaced with a universal remote control; however, universal remotes still have their shortcomings. Generally, current universal remotes often do not have the resources (e.g., memory and computational logic) to allow them to be used with all devices, or they may not be capable of controlling a new device. In addition, in order to accommodate the variety of devices to be controlled, universal remotes usually have a multiplicity of buttons and thus can be difficult to use.

Another shortcoming associated with current remote control devices is their limited range. Commonly, remote control devices use infrared beams to communicate commands to the device that is to be controlled, and so the remotes can only be used for line-of-sight applications. Devices behind an object, around a corner, or in another room cannot be controlled if they are not in the line of sight of an infrared remote.

A more modern solution is to wire devices together into a network of some sort, so that they can be controlled from a central location such as a personal computer. However, this approach also has a number of shortcomings. For example, the connections and cabling needed may be quite cumbersome and complex. In addition, this approach is difficult and expensive to backfit into existing homes and businesses. Furthermore, such an approach is not necessarily convenient. For instance, the central computer system is not a practical or convenient replacement for a television remote, nor is it portable enough to allow it to be easily moved from one room to another whenever it is necessary to do so.

Accordingly, a need exists for a device and/or method that can be used to remotely control a variety of different devices and appliances, including new devices. A need also exists for a device and/or method that can satisfy the above need, that is relatively simple to introduce into existing homes and businesses, and that is user-friendly. In addition, a need exists for a device and/or method that satisfies the above needs, is portable, and is not limited to line-of-sight applications.

DISCLOSURE OF THE INVENTION

The present invention provides a system and method that can be used to remotely control a variety of different devices, including new devices. The present invention also provides a system and method that is relatively easy to introduce into existing homes and businesses and that is user-friendly. In addition, the present invention provides a system and method that is portable and that is not limited to line-of-sight applications.

In the present embodiment, the present invention pertains to a system and method for controlling remote devices over a wireless connection (e.g., using a radio signal). In one embodiment, a portable computer system (e.g., a palmtop or hand-held computer) having a transceiver is used to control compliant devices. In a preferred embodiment, the transceiver and the remote devices are Bluetooth-enabled devices.

In the present embodiment of the present invention, a wireless connection between the portable computer system and one or more remote devices is established. Each of the remote devices is manifested on a display device of the portable computer system, and one of the devices is selected using, for example, a stylus element.

In one embodiment, the stylus element can also be used to specify commands for controlling the remote device. A position where the stylus element makes contact with a surface of the display device of the portable computer system is registered. The particular position where the stylus element makes contact with the display device is translated into a particular command for controlling the remote device. The command is then transmitted to the remote device over the wireless connection.

In one embodiment, a rendering of the remote device or of a mechanism that can be used to control the remote device is displayed on the display device. The contact of the stylus element with a position in the rendering is translated into a particular command for controlling the remote device. In another embodiment, a menu of commands for controlling the remote device is displayed on the display device. The contact of the stylus element with a position in the menu is translated into a particular command for controlling the remote device.

In yet another embodiment, the movement of the stylus element over the surface of an input device is recognized and translated into a particular command for controlling the remote device. In another embodiment, by moving the stylus element over the surface of the input device, motion is imparted to the rendering on the display device of the remote device or the mechanism for controlling the remote device.

The present invention thus provides a system (e.g., a Bluetooth-enabled device, specifically a portable computer system) that can be used to remotely control compliant devices (e.g., other Bluetooth-enabled devices) over a wireless (radio) connection. With a radio connection, the system of the present invention is not limited to line-of-sight applications. Remote devices can be adapted to receive commands over the wireless connection, obviating the need for hardwire connections and making the system relatively easy to implement in homes and businesses. The processing

power and other features of the portable computer system enable user-friendly interfaces, and also allow a variety of remote devices to be controlled, including new devices introduced into the home or business.

These and other objects and advantages of the present invention will become obvious to those of ordinary skill in the art after having read the following detailed description of the preferred embodiments which are illustrated in the various drawing figures.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and form a part of this specification, illustrate embodiments of the invention and, together with the description, serve to explain the principles of the invention:

FIG. **1** illustrates one embodiment of a network of devices coupled using wireless connections in accordance with the present invention.

FIG. **2** is a block diagram of one embodiment of a portable computer system in accordance with the present invention.

FIG. **3** is a top-side perspective view of a portable computer system in accordance with one embodiment of the present invention.

FIGS. **4**A and **4**B are block diagrams showing one embodiment of a wireless transceiver coupled to, respectively, a portable computer system and an external device in accordance with the present invention.

FIG. **5** illustrates the different operating modes of a wireless transceiver in accordance with one embodiment of the present invention.

FIG. **6** illustrates the flow of messages between a controlling device and remote devices in accordance with one embodiment of the present invention.

FIG. **7** illustrates one embodiment of a display used on a controlling device in accordance with the present invention.

FIG. **8** illustrates another embodiment of a display used on a controlling device in accordance with the present invention.

FIG. **9** illustrates another embodiment of a display used on a controlling device in accordance with the present invention.

FIG. **10** illustrates a display on a controlling device responding to movement on an input device in accordance with one embodiment of the present invention.

FIG. **11** is a flowchart of the steps in a process for controlling a remote device over a wireless connection in accordance with one embodiment of the present invention.

BEST MODE FOR CARRYING OUT THE INVENTION

Reference, will now be made in detail to the preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings. While the invention will be described in conjunction with the preferred embodiments, it will be understood that they are not intended to limit the invention to these embodiments. On the contrary, the invention is intended to cover alternatives, modifications and equivalents, which may be included within the spirit and scope of the invention as defined by the appended claims. Furthermore, in the following detailed description of the present invention, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it will be obvious to one of ordinary skill in the art that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, and circuits have not been described in detail so as not to unnecessarily obscure aspects of the present invention.

Some portions of the detailed descriptions which follow are presented in terms of procedures, logic blocks, processing, and other symbolic representations of operations on data bits within a computer memory. These descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. A procedure, logic block, process, etc., is here, and generally, conceived to be a self-consistent sequence of steps or instructions leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated in a computer system. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, bytes, values, elements, symbols, characters, terms, numbers, or the like.

It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussions, it is appreciated that throughout the present invention, discussions utilizing terms such as "establishing," "registering," "recognizing," "broadcasting," "receiving," "manifesting," "transmitting," "displaying," or the like, refer to the action and processes (e.g., process **1100** FIG. **11**) of a computer system or similar intelligent electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

The present invention is discussed primarily in a context in which devices and systems are coupled using wireless links, and specifically with regard to devices and systems compliant with the Bluetooth technology. Bluetooth is the code name for a technology specification for small form factor, low-cost, short-range radio links between personal computers (PCs), mobile phones and other devices and appliances. However, it is appreciated that the present invention may be utilized with devices and systems compliant with standards different from Bluetooth, such as the IEEE (Institute of Electronic and Electrical Engineering) 802.11 standard.

The Bluetooth technology allows cables that connect one device to another to be replaced with short-range radio links. Bluetooth is targeted at mobile and business users who need to establish a link, or small network, between their computer, cellular phone and other peripherals. The required and nominal range of Bluetooth is thus set to approximately ten (10) meters. To support other uses, for example the home environment, Bluetooth can be augmented to extend the range to up to 100 meters.

The Bluetooth technology is based on a high-performance, yet low-cost, integrated radio transceiver. For instance, Bluetooth transceivers built into both a cellular telephone and a laptop computer system would replace the cables used today to connect a laptop to a cellular telephone.

Bluetooth radio technology can also provide: a universal bridge to existing data networks, a peripheral interface, and a mechanism to form small private ad hoc groupings of connected devices away from fixed network infrastructures.

FIG. **1** illustrates the topology of a network of devices coupled using wireless connections in accordance with one embodiment of the present invention. In the parlance of Bluetooth, a collection of devices connected in a Bluetooth system are referred to as a "piconet" or a "subnet." In the present embodiment, a piconet starts with two connected devices, and may grow to eight connected devices. All Bluetooth devices are peer units; however, when establishing a piconet, one unit will act as a master and the other(s) as slave(s) for the duration of the piconet connection.

A Bluetooth system supports both point-to-point and point-to-multi-point connections. Several piconets can be established and linked together in a "scatternet," where each piconet is identified by a different frequency hopping sequence. All devices participating on the same piconet are synchronized to their respective hopping sequence.

Accordingly, devices **10**, **20**, **30** and **40** are coupled in piconet **1** using wireless connections **80***a–c*. Similarly, devices **50**, **60** and **70** are coupled in piconet **2** using wireless connections **80***e–f*. Piconet **1** and piconet **2** are coupled using wireless connection **80***d*. Devices **10–70** can be printers, personal digital assistants (PDAs), desktop computer systems, laptop computer systems, cell phones, fax machines, keyboards, and joysticks equipped with a Bluetooth radio transceiver or adapted to communicate with Bluetooth devices ("Bluetooth-enabled"). In accordance with the present invention, devices **10–70** can also be virtually any type of device, including mechanical devices and appliances, equipped with a Bluetooth radio transceiver or Bluetooth-enabled. The Bluetooth radio transceiver may be integrated into the device, or it may be coupled to the device.

FIG. **2** is a block diagram of a portable computer system **100** (e.g., a PDA, a hand-held computer system, or palmtop computer system) upon which embodiments of the present invention can be implemented. Computer system **100** includes an address/data bus **110** for communicating information, a central processor **101** coupled with the bus for processing information and instructions, a volatile memory **102** (e.g., random access memory, RAM) coupled with the bus **110** for storing information and instructions for the central processor **101** and a non-volatile memory **103** (e.g., read only memory, ROM) coupled with the bus **110** for storing static information and instructions for the processor **101**. Computer system **100** also includes an optional data storage device **104** (e.g., memory stick) coupled with the bus **110** for storing information and instructions. Data storage device **104** can be removable.

Computer system **100** also contains a display device **105** coupled to the bus **110** for displaying information to the computer user. The display device **105** utilized with computer system **100** may be a liquid crystal display device, a cathode ray tube,(CRT), a field emission display device (also called a flat panel CRT) or other display device suitable for generating graphic images and alphanumeric characters recognizable to the user. In the preferred embodiment, display device **105** is a flat panel display.

Computer system **100** also includes a cursor control or directing device (on-screen cursor control **107**) coupled to bus **110** for communicating user input information and command selections to processor **101**. In one implementation, on-screen cursor control device **107** is a touch-screen device incorporated with display device **105**. On-screen cursor control device **107** is capable of registering a position on display device **105** where a stylus makes contact.

In accordance with the present invention, a stylus can be used to select a command for controlling a remote device by touching the stylus to display device **105**. In one embodiment, a command can be selected from a menu of commands displayed on display device **105**. In another embodiment, a rendering of the remote device or of a mechanism for controlling the remote device may be manifested on display device **105**, and a command can be selected by touching the stylus to a prescribed location in the rendering. The position where the stylus contacts display device **105** is registered and fed to processor **101**, which translates this information into a command for controlling the remote device. The command is then transmitted to the remote device over a wireless connection using signal transmitter/receiver device ("transceiver") **108**.

Also included in computer system **100** of FIG. **2** is an input device **106** that in one implementation is a stroke or character recognition pad (e.g., a "digitizer"). Input device **106** can communicate information and command selections to processor **101**. Input device **106** is capable of registering a position where a stylus (or an element having the functionality of a stylus) makes contact. Input device **106** also has the capability of registering movements of a stylus (or an element having the functionality of a stylus) across or above the surface of input device **106**.

In accordance with the present invention, in one embodiment, a stylus can be used for making a stroke or inscribing a character on the surface of input device **106**. The stroke or character information is then fed to a processor **101** for automatic character recognition. Once the stroke or character information is recognized, it can be displayed on display device **105** for verification and/or modification.

In accordance with the present invention, stroke information entered onto input device **106** can correspond to a command that can be used to control a remote device. That is, particular strokes or characters can correspond to a respective command. A stroke or character is recognized by processor **101** and translated by processor **101** into a command for controlling a remote device. The command is then transmitted to the remote device over a wireless connection using transceiver **108**.

With reference still to FIG. **2**, transceiver **108** is coupled to bus **110** and enables computer system **100** to communicate wirelessly with other electronic devices coupled in a piconet or scatternet (refer to FIG. **1**). It should be appreciated that within the present embodiment, transceiver **108** is coupled to an antenna and provides the functionality to transmit and receive information over a wireless communication interface. In one embodiment, transceiver **108** is a Bluetooth device. Additional information with regard to the Bluetooth embodiment is provided in conjunction with FIGS. **4**A and **4**B.

FIG. **3** is a perspective illustration of the top-side face **100***a* of one embodiment of the portable computer system **100** (FIG. **2**) in accordance with the present invention. The top-side face **100***a* contains one or more dedicated and/or programmable buttons **75** for selecting information and causing the computer system to implement functions. The on/off button **95** is also shown.

In the present embodiment, the top-side face **100***a* contains a display device **105** typically surrounded by a bezel or cover. A removable stylus element **90** is also shown. The

display device **105** is a touch screen capable of registering contact between the screen and the tip of the stylus element **90**. The top-side face **100***a* also contains an input device **106** that in one implementation is a stroke or character recognition pad. Input device **106** is a touch screen type of device capable of registering contact with a tip of stylus element **90**, and also can register movements of the stylus element. The stylus element **90** can be of any shape and material to make contact with the display device **105** and input device **106**.

FIGS. **4**A and **4**B are block diagrams of one embodiment of a transceiver **108** in accordance with the present invention. In a preferred embodiment (the "Bluetooth embodiment", transceiver **108** is a Bluetooth device comprising a digital component (e.g., a Bluetooth controller) and an analog component (e.g., a Bluetooth radio). In accordance with the present invention, a transceiver **108** is coupled via a system bus **110** to a system or device that will be used to control remote devices (e.g., portable computer system **100** of FIG. **2**). Similarly, a transceiver **108** is coupled via a connector **250** to each remote device that is to be controlled (e.g., external device **290**).

With reference to both FIGS. **4**A and **4**B, in the present embodiment, transceiver **108** comprises an antenna **205** for receiving or transmitting radio signals, a radio frequency (RF) module **210**, a link controller **220**, a microcontroller (or central processing unit) **230**, and an external interface **240**.

In the Bluetooth embodiment, RF module **210** is a Bluetooth radio. Bluetooth radios operate in the ISM (Industrial, Scientific, Medical) band at 2.4 GHz. A frequency hop transceiver is applied to combat interference and fading. Bluetooth uses a packet-switching protocol based on a frequency hop scheme with 1600 hops/second. Slots can be reserved for synchronous packets. A packet nominally covers a single slot, but can be extended to cover up to five slots. Each packet is transmitted in a different hop frequency. The entire available frequency spectrum is used with 79 hops of one (1) MHz bandwidth, defined analogous to the IEEE 802.11 standard. The frequency hopping scheme is combined with fast ARQ (Automatic Repeat Request), cyclic redundancy check (CRC) and Forward Error Correction (FEC) for data.

In the present embodiment, link controller **220** is a hardware digital signal processor for performing baseband processing as well as other functions such as Quality-of-Service, asynchronous transfers, synchronous transfers, audio coding, and encryption.

In one embodiment, microcontroller **230** is an application specific integrated circuit (ASIC). In the Bluetooth embodiment, microcontroller **230** is a separate central processing unit (CPU) core for managing transceiver **108** and for handling some inquiries and requests without having to involve the host device. In the Bluetooth embodiment, microcontroller **230** runs software that discovers and communicates with other Bluetooth devices via the Link Manager Protocol (LMP). The LMP provides a number of services including sending and receiving of data, inquiring of and reporting a name or device identifier, making and responding to link address inquiries, connection setup, authentication, and link mode negotiation and setup. The LMP also can be used to place transceiver **108** in "sniff" mode, "hold" mode, "park" mode or "standby" mode (refer to FIG. **5** below).

With reference still to FIGS. **4**A and **4**B, in the present embodiment, interface **240** is for coupling transceiver **108** to portable computer system **100** or to external device **290** in a suitable format (e.g., USB, PCMCIA, PCI, CardBus, PC Card, etc.). In the present embodiment, interface **240** runs software that allows transceiver **108** to interface with portable computer system **100** or external device **290**.

FIG. **5** illustrates the different operating modes of a wireless transceiver **108** (FIGS. **4**A and **4**B) in accordance with one embodiment of the present invention. In the Bluetooth embodiment, before any connections between Bluetooth devices are created, all devices are in standby mode **(502)**. In this mode, an unconnected unit "listens" for messages at a regular rate (e.g., every 1.28 seconds) on a set of hop frequencies defined for that unit. The hold mode **(514)** is a power saving mode that can be used for connected units if no data need to be transmitted. The sniff mode **(516)** and park mode **(512)** are also low power modes. In the sniff mode, a device listens to the piconet at a reduced rate (relative to the regular rate), thus reducing its duty cycle. The sniff interval is programmable and depends on the application. In the park mode, a device is still synchronized to the piconet but does not participate in the traffic.

A connection between devices is made by a "page" message **(506)** if the address is already known, or by an "inquiry" message **(504)** followed by a subsequent page message if the address is unknown. When connected **(510)**, data can be transmitted **(508)** between devices.

FIG. **6** illustrates the flow of messages between a controlling device (e.g., portable computer system **100**) and remote devices to be controlled (**610**, **620** and **630**) in accordance with the present embodiment of the present invention. In one embodiment, portable computer system **100** and remote devices **610–630** are Bluetooth devices or Bluetooth-enabled devices.

In the present embodiment, when it is necessary to locate and identify compliant devices, portable computer system transmits a broadcast message **640** (e.g., an inquiry **504**) that is received by compliant remote devices **610–630**. For example, a user with portable computer system **100** enters a room containing remote devices **610–630**. Portable computer system **100**, either automatically or in response to a user input, transmits broadcast message **640** for the purpose of discovering compliant devices in the room.

As compliant devices, remote devices **610–630** respond to broadcast message **640** via responses **650***a*, **650***b* and **650***c*, respectively. In the present embodiment, responses **650***a–c* include the Medium Access Control (MAC) address for remote devices **610–630**. Typically, each remote device is assigned a temporary MAC address for the duration of the connection. All communications between portable computer system **100** and a remote device carry the MAC address of the remote device. Responses **650***a–c* can also include information characterizing, for example, the type and capabilities of each remote device. This information may include an identifier that can be used by portable computer system **100** to characterize the remote device based on information stored in a database or lookup table.

Portable computer system **100** can then transmit a command **660** to a selected remote device (e.g., remote device B **620**). Command **660** is a command for controlling the remote device in some prescribed manner (e.g., turning the device off or on, raising or lowering a level, etc.) based on the type of device and its capabilities. In accordance with the present invention, a second device can be selected (e.g., remote device C **630**), and a command **670** can be transmitted to that device.

In the present embodiment, when a connection between portable computer system **100** and a remote device has already been established, or when the MAC address of the

remote device is known, broadcast message **640** is a page **506** (FIG. **5**) instead of an inquiry **504**.

FIG. **7** illustrates one embodiment of a display used on a controlling device (e.g., portable computer system **100**) in accordance with the present invention. As described above, portable computer system **100** includes a display device **105**, an input device **106**, and a stylus element **90**.

In this embodiment, with reference also to FIG. **6**, each of the remote devices **610–630** have sent a response **650***a–c*, respectively, to portable computer system **100** in response to broadcast message **640**. Accordingly, each of remote devices **610–630** are indicated on display device **105**. For example, an icon can be used to represent each remote device, each remote device can be identified by its name in a menu, etc. It is appreciated that other mechanisms may be used to indicate a remote device on display device **105** in accordance with present invention.

In the present embodiment, a user can then select one of the remote devices by touching stylus element **90** to display device **105**. It is appreciated that an element other than stylus element **90** can be used to make a selection, or that another mechanism may be used to make a selection. For example, the user may simply touch the screen, or an on-screen cursor of some type may be used.

FIG. **8** illustrates another embodiment of a display used on a controlling device (e.g., portable computer system **100**) in accordance with the present invention. In this embodiment, a connection has been established between the controlling device and the device to be controlled, and the characteristics otfthe device to be controlled have been identified. In this embodiment, display device **105** displays a rendering of a mechanism that can be used to control the remote device, such as an on/off switch. In the present embodiment, a user can turn the remote device on by touching stylus element **90** to position **802** in the rendering, and can turn the remote device off by touching stylus element **90** to position **804**. However, it is appreciated that an element other than stylus element **90** can be used to make a selection, or that another mechanism may be used to make a selection.

In one embodiment, a user can also control the remote device using input device **106**. As described above, input device **106** is adapted to recognize movements of stylus **90** on or above the surface of input device **106**, and to translate particular movements into particular commands. Thus, for example, a user might turn on the remote device by writing the word "on" using input device **106**. Alternatively, a user might instead write a character that represents the command "on" in some type of shortened version, or might make a stroke that represents this command. It is appreciated that other mechanisms, styles, and methods can be used to input a command using input device **106** in accordance with the present invention.

FIG. **9** illustrates another embodiment of a display used on a controlling device (e.g., portable computer system **100**) in accordance with the present invention. In this embodiment, a connection has been established between the controlling device and the device to be controlled, and the characteristics of the device to be controlled have been identified. In this embodiment, display device **105** displays a menu **910** of commands for the remote device, such as an "on" command and an "off" command. In the present embodiment, a user can turn the remote device on by touching stylus element **90** to a particular position in the menu **910**. It is appreciated that an element other than stylus element **90** can be used to make a selection, or that another mechanism may be used to make a selection.

FIG. **10** illustrates a display on a controlling device (e.g., portable computer system **100**) responding to movement on an input device **106** in accordance with one embodiment of the present invention. In this embodiment, a connection has been established between the controlling device and the device to be controlled, and the characteristics of the device to be controlled have been identified. In this embodiment, for example, a variable level of the remote device is to be controlled (e.g., a level of brightness if the remote device is a light).

In the present embodiment, the level to be controlled is indicated by an indicator **1020** displayed on display device **105**. A user touches stylus element **90** to input device **106**, and moves the stylus as indicated by stroke **1010**. In response to the.motion of the stylus across input device **106**, indicator **1020** moves in a corresponding manner along the path **1030**. That is, motion is imparted to indicator **1020** by moving stylus element **90** on input device **106**. It is appreciated that an element other than stylus element **90**, or another type of mechanism, can be used with input device **106**.

It is appreciated that, in accordance with the present invention, different mechanisms for controlling the remote device can be rendered on display device **105**, different types of movement can be used with input device **106**, and different types of motion can be imparted to the rendering on display device **105** in response to the movements on input device **106**. In accordance with the present embodiment of the present invention, either the motion of the stylus itself or the corresponding motion of the rendering can be translated into a command for controlling the remote device.

FIG. **11** is a flowchart of the steps in a process **1100** for controlling a remote device over a wireless connection in accordance with one embodiment of the present invention. In step **1110**, a wireless connection is established between the controlling device (e.g., portable computer system **100** of FIG. **6**) and a remote device or remote devices to be controlled (e.g., remote devices **610**, **620** and **630** of FIG. **6**). As described above, if the MAC addresses of remote devices **610–630** are known, then a page **506** (FIG. **5**) is used by portable computer system **100**; otherwise, an inquiry **504** (FIG. **5**) is used. In response to the broadcast message, each of remote devices **610–630** sends a response to portable computer system **100**. In the Bluetooth embodiment, the broadcast message and the responses are transmitted using radio signals.

In one embodiment, the characteristics and capabilities of remote devices **610–630** are identified in the response. In another embodiment, the characteristics and capabilities of various types of devices are stored in a database or lookup table in a memory unit of portable computer system **100**. In this latter embodiment, the responses from the remote devices include an identifier that can be used by portable computer system **100** to retrieve the characteristics and capabilities of remote devices **610–630** from memory.

In step **1120** of FIG. **11**, each of the remote devices (e.g., remote devices **610–630**) responding to the broadcast message is manifested on portable computer system **100**. In one embodiment, each remote device is indicated on display device **105** of portable computer system **100** (refer to FIG. **7**). The characteristics and capabilities of each remote device **610–630** are linked to the indications (e.g., icons) on display device **105**.

In step **1130**, one of the remote devices **610–630** is selected by a user. In one embodiment, the user makes a selection by touching a stylus (e.g., stylus element **90** of

FIG. 7) to the screen of display device **105**. It is appreciated that, if only one remote device is present or if a response is received from only one remote device, then step **1130** may be bypassed.

In step **1140** of FIG. **11**, as described above in conjunction with FIGS. **8**, **9** and **10**, a user can input a command for controlling any of remote devices **610**–**630** using display device **105** and/or input device **106**. In various embodiments, display device **105** displays a rendering of the remote device, a rendering of a mechanism for controlling the remote device, or a menu of commands for controlling the remote device. In one embodiment, a user can use stylus element **90** to make contact with the surface of display device **105**. The position where stylus element **90** contacts the surface of input device **105** is registered and translated into a particular command.

In another embodiment, a user can input a command using input device **106** and stylus element **90**, by inscribing a command or by using a character or stroke that represents a command. In these cases, input device **106** registers the movement of the stylus and translates the movement into a particular command. In another embodiment, a user can impart motion to the rendering of the remote device displayed on display device **105** by moving stylus element **90** on input device **106**. The motion of the stylus or the corresponding motion of the rendering can be translated into a command for controlling the remote device.

In step **1150** of FIG. **11**, the command (e.g., command **660** of FIG. **6**) is transmitted to the remote device. In accordance with the present invention, additional commands can also be sent to the remote device. In the Bluetooth embodiment, commands are transmitted via a radio signal.

In the case in which more than one remote device is to be controlled, another remote device can be selected as in step **1130**, and commands can be input and transmitted to that device as described above. Furthermore, portable computer system **100** can be transported to a new location (e.g., another room), and process **1100** can be repeated to locate and identify compliant remote devices in the new location, establish connections with those devices, and specify and transmit commands for controlling those devices. The processing power and intelligence of portable computer system **100** in combination with the processing power and intelligence of each transceiver **108** (in both portable computer system **100** and in the remote device; refer to FIGS. **4**A and **4**B) permit portable computer system **100** to be updated as needed, so that it can operate as a universal remote control device for a multiplicity of different devices, including new devices.

Thus, the present invention provides a system and method that can be used to remotely control a variety of different devices. In one embodiment, the present invention provides a system (e.g., a Bluetooth-enabled device, specifically a portable computer system) that can be used to remotely control compliant devices (e.g., other Bluetooth-enabled devices) over a wireless (radio) connection. With a radio connection, the system of the present invention is not limited to line-of-sight applications. Remote devices can be adapted to receive commands over the wireless connection, obviating the need for hardwire connections and making the system relatively easy to implement in homes and businesses. The processing power and other features of the portable computer system enable user-friendly interfaces, and also allow a variety of remote devices to be controlled, including new devices introduced into the home or business.

The preferred embodiment of the present invention, portable device control console with wireless connection, is thus described. While the present invention has been described in particular embodiments, it should be appreciated that the present invention should not be construed as limited by such embodiments, but rather construed according to the following claims.

What is claimed is:

1. A method for controlling a remote devices over a wireless connection, said method comprising:

a) establishing said wireless connection between a transceiver and said remote device by:

broadcasting a message, said message for locating remote devices within range of said transceiver; and

receiving a response from said remote device;

b) manifesting said remote device on a display device;

c) registering a position where contact is made with a surface of an input device, wherein a particular position on said input device is translated into a particular command for controlling said remote device; and

d) transmitting a command to said remote device over said wireless connection.

2. The method as recited in claim **1** wherein said step c) comprises the step of:

registering a position where a stylus element makes contact with said surface of said input device.

3. The method as recited in claim **2** wherein said step c) further comprises the step of:

recognizing a movement of said stylus element over said surface of said input device, wherein a particular movement of said stylus element is translated into a particular command for controlling said remote device.

4. The method as recited in claim **2** further comprising the step of:

registering a position where said stylus element makes contact with a screen of said display device, wherein a particular position on said screen is translated into a particular command for controlling said remote device.

5. The method as recited in claim **1** further comprising the steps of:

receiving responses from a plurality of remote devices;

manifesting each of said plurality of remote devices on said display device; and

selecting one of said plurality of remote devices.

6. The method as recited in claim **1** further comprising the step of:

displaying on said display device a rendering of a mechanism for controlling said remote device.

7. The method as recited in claim **6** further comprising the step of:

contacting a particular position in said rendering, wherein said contacting is translated into a particular command corresponding to said particular position.

8. The method as recited in claim **6** further comprising the step of:

imparting motion to said rendering in response to movement of a stylus element over said surface of said input device.

9. The method as recited in claim **1** further comprising the steps of:

displaying on said display device a menu of commands for controlling said remote device; and

contacting a particular position in said menu, wherein said contacting is translated into a particular command corresponding to said particular position.

10. The method as recited in claim **1** wherein said transceiver and said remote device are Bluetooth-enabled devices.

11. A computer system comprising:

a bus;

a processor coupled to said bus;

a transceiver coupled to said bus;

a display device coupled to said bus; and

an input device coupled to said bus;

said processor for performing a method for controlling a remote device over a wireless connection, said method comprising the computer-implemented steps of:

a) establishing said wireless connection between said transceiver and said remote device by:

broadcasting a message, said message for locating remote devices within range of said transceiver; and

receiving a response from said remote device;

b) manifesting said remote device on a display device;

c) registering a position where contact is made with a surface of an input device, wherein a particular position on said input device is translated into a particular command for controlling said remote device; and

d) transmitting a command to said remote device over said wireless connection.

12. The computer system of claim 11 wherein said step c) of said method comprises the step of:

registering a position where a stylus element makes contact with said surface of said input device.

13. The computer system of claim 12 wherein said step c) of said method further comprises the step of:

recognizing a movement of said stylus element over said surface of said input device, wherein a particular movement of said stylus element is translated into a particular command for controlling said remote device.

14. The computer system of claim 12 wherein said method further comprises the steps of:

registering a position where said stylus element makes contact with a screen of said display device, wherein a particular position on said screen is translated into a particular command for controlling said remote device.

15. The computer system of claim 11 wherein said method further comprises the steps of:

receiving responses from a plurality of remote devices;

manifesting each of said plurality of remote devices on said display device; and

selecting one of said plurality of remote devices.

16. The computer system of claim 11 further comprising the step of:

displaying on said display device a rendering of a mechanism for controlling said remote device.

17. The computer system of claim 16 wherein said method further comprises the step of:

contacting a particular position in said rendering, wherein said contacting is translated into a particular command corresponding to said particular position.

18. The computer system of claim 16 wherein said method further comprises the step of:

imparting motion to said rendering in response to movement of a stylus element over said surface of said input device.

19. The computer system of claim 11 wherein said method further comprises the steps of:

displaying on said display device a menu of commands for controlling said remote device; and

contacting a particular position in said menu, wherein said contacting is translated into a particular command corresponding to said particular position.

20. The computer system of claim 11 wherein said transceiver and said remote device are Bluetooth-enabled devices.

21. A hand-held computer system for controlling a remote device over a radio connection, said system comprising:

a bus;

a processor coupled to said bus;

a transceiver coupled to said bus, said transceiver for transmitting commands for controlling said remote device over said radio connection, wherein said transceiver is adapted to broadcast a message for locating remote devices within range of said transceiver, wherein each remote device responding to said message is indicated on said display device;

a display device coupled to said bus, said display device adapted to register a position where a stylus element makes contact with a screen of said display device, wherein a particular position on said screen is translated into a particular command for controlling said remote device; and

an input device coupled to said bus, said input device adapted to register a position where a stylus element makes contact with a surface of said input device, wherein a particular position on said input device is translated into a particular command for controlling Said remote device.

22. The computer system of claim 21 wherein said input device is adapted to recognize a movement of said stylus element over said surface of said input device, wherein a particular movement of said stylus element is translated into a particular command for controlling said remote device.

23. The computer system of claim 21 wherein said transceiver and said remote device are Bluetooth-enabled devices.

24. The computer system of claim 21 wherein said display device is adapted to display a rendering of a mechanism for controlling said remote device.

25. The computer system of claim 24 wherein said display device is adapted to register a position where said stylus element makes contact within said rendering, wherein a particular position within said rendering is translated into a particular command for controlling said remote device.

26. The computer system of claim 24 wherein said display device is adapted to impart motion to said rendering in response to movement of said stylus element over said surface of said input device.

27. The computer system of claim 21 wherein said display device is adapted to display a menu of commands for controlling said remote device.

* * * * *

US006216158B1

(12) **United States Patent**
Luo et al.

(10) **Patent No.: US 6,216,158 B1**
(45) **Date of Patent: Apr. 10, 2001**

(54) **SYSTEM AND METHOD USING A PALM SIZED COMPUTER TO CONTROL NETWORK DEVICES**

(75) Inventors: **Wenjun Luo**, Fremont; **Elaine P. Lusher**, Pleasanton, both of CA (US)

(73) Assignee: **3COM Corporation**, Santa Clara, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/237,609**

(22) Filed: **Jan. 25, 1999**

(51) **Int. Cl.**[7] **G06F 13/00**

(52) **U.S. Cl.** **709/217**; 709/219; 709/225; 709/313; 709/329

(58) **Field of Search** 709/202, 203, 709/217, 219, 223, 225, 230, 250, 313, 319, 328, 329

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,564,070	*	10/1996	Want et al.	455/507
5,790,977	*	8/1998	Ezekiel	702/122
5,832,506	*	11/1998	Kuzma	707/200
5,909,545	*	6/1999	Frese II et al.	709/208
5,923,885	*	7/1999	Johnson et al.	395/712
5,964,836	*	10/1999	Rowe et al.	709/221
5,973,696	*	10/1999	Agranat et al.	345/357

OTHER PUBLICATIONS

M.G. Lamming, *Institute for Personalized Information Environment 1994* "Towards Future Personalised Information Environments" *Technical Report EPC—1994—104*, Dec. 21, 1993.

McCandless, Michael, "The PalmPiolt and the Handhled Revolution", IEEE Expert, Nov./Dec. 1997, pp. 6–8.

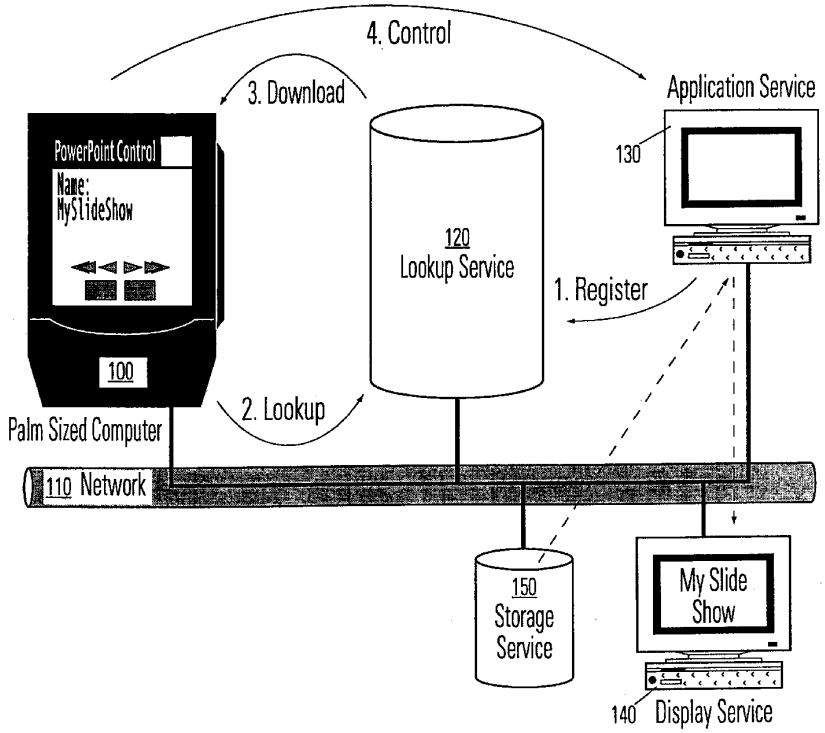
* cited by examiner

Primary Examiner—Viet D. Vu
(74) *Attorney, Agent, or Firm*—Mark A. Haynes; Haynes & Beffel LLP

(57) **ABSTRACT**

Controlling network services using palm sized computers is described. A program on the palm sized computer is used to access a registry of network services that may be available. The registry includes descriptions for various services. Each description includes at least a reference to program code that can be downloaded to the palm sized computer. Executing this program causes the palm sized computer to issue commands directly to the specific network services needed. In some cases, these network services include application services for running desktop applications that the palm sized computer could not execute.

20 Claims, 7 Drawing Sheets



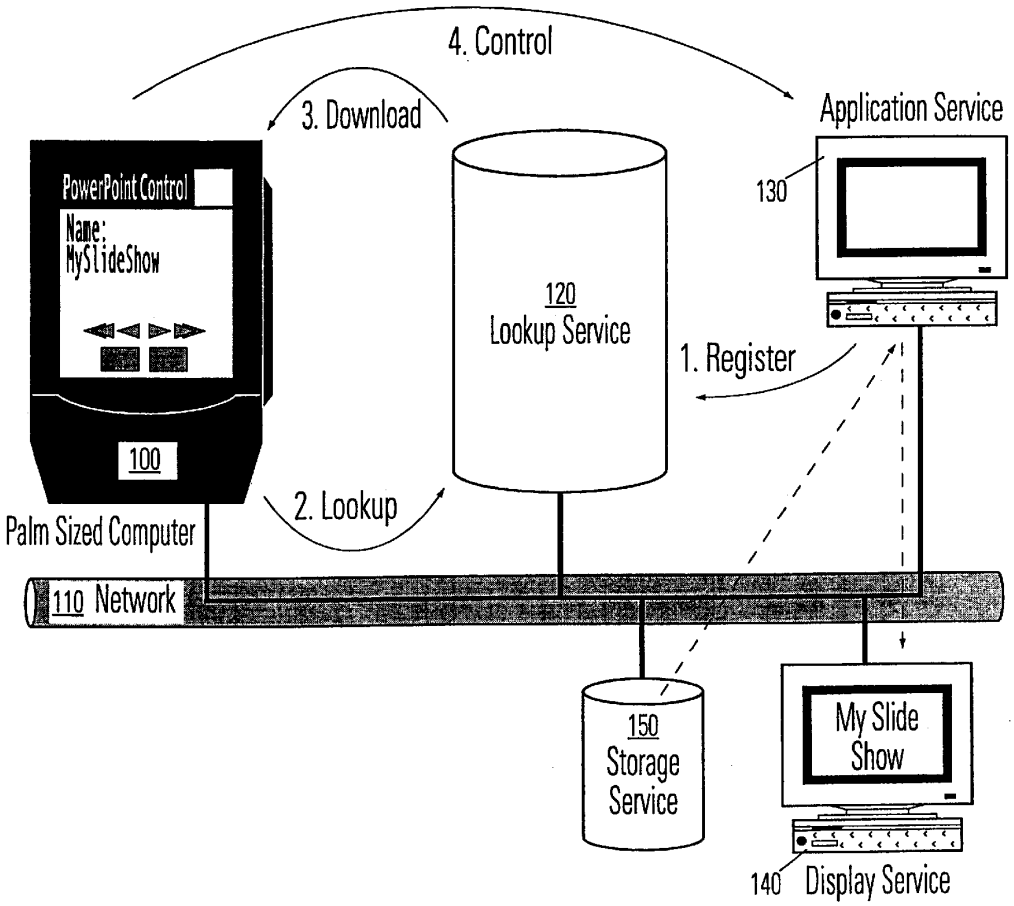

4. Control
3. Download
Application Service
130
PowerPoint Control
Name:
MySlideShow
120
Lookup Service
1. Register
100
2. Lookup
Palm Sized Computer
110 Network
150
Storage
Service
My Slide
Show
140 Display Service

FIG. 1

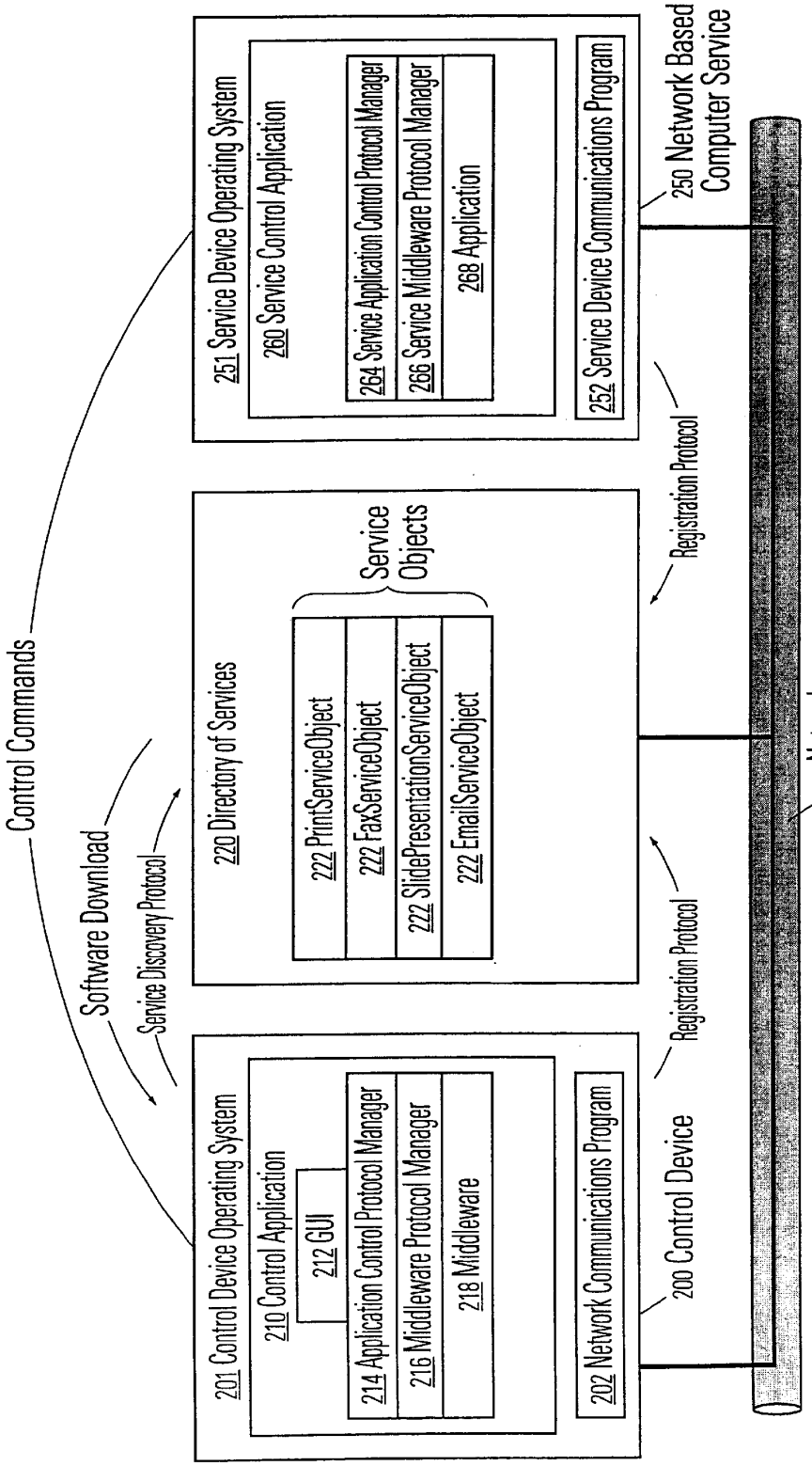

Control Commands
Software Download
Service Discovery Protocol
201 Control Device Operating System
210 Control Application
212 GUI
214 Application Control Protocol Manager
216 Middleware Protocol Manager
218 Middleware
202 Network Communications Program
200 Control Device
220 Directory of Services
222 PrintServiceObject
222 FaxServiceObject
222 SlidePresentationServiceObject
222 EmailServiceObject
Service Objects
Registration Protocol
Registration Protocol
251 Service Device Operating System
260 Service Control Application
264 Service Application Control Protocol Manager
266 Service Middleware Protocol Manager
268 Application
252 Service Device Communications Program
250 Network Based Computer Service
110 Network

FIG. 2

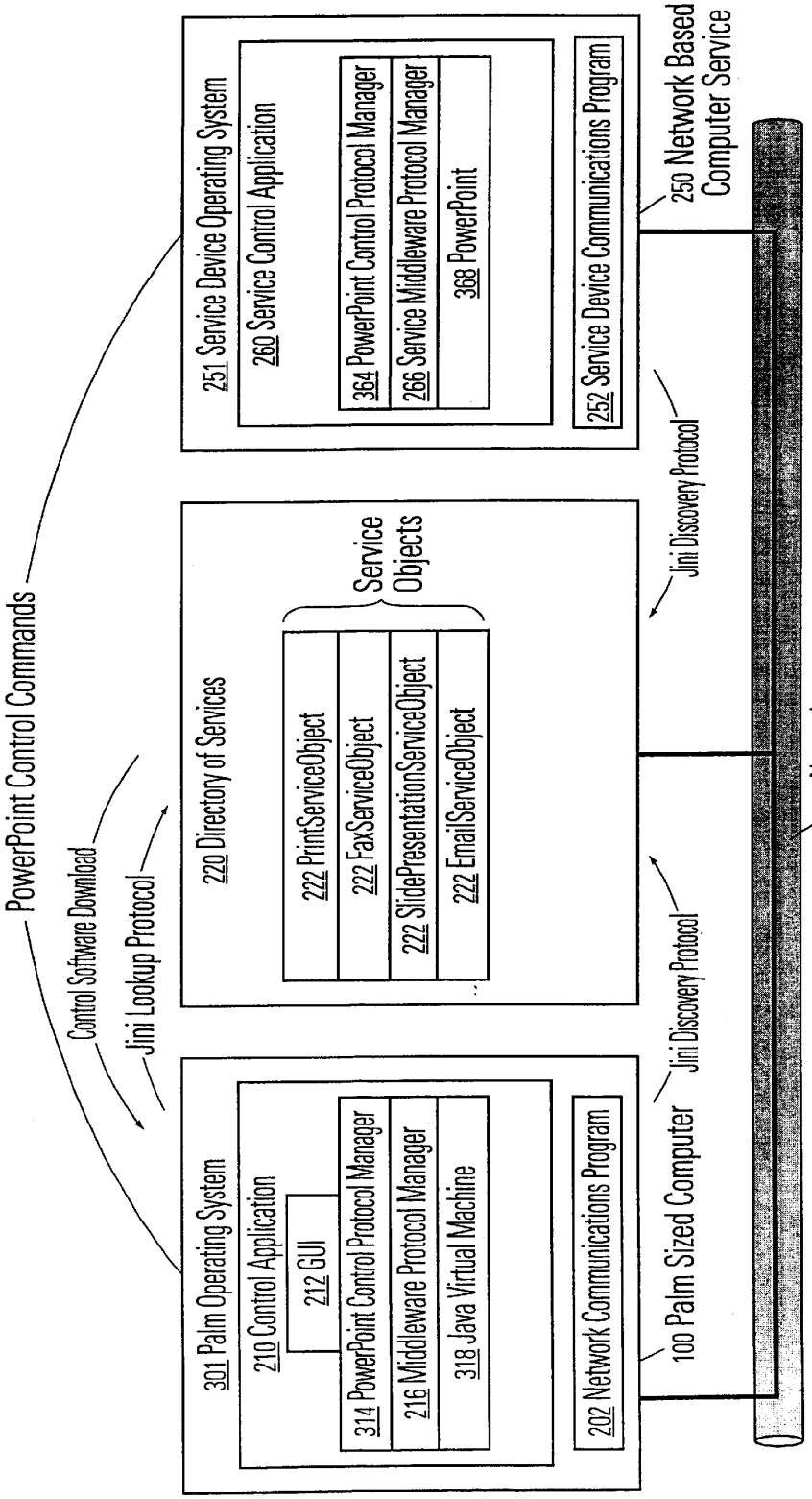

PowerPoint Control Commands
Control Software Download
Jini Lookup Protocol
301 Palm Operating System
210 Control Application
212 GUI
314 PowerPoint Control Protocol Manager
216 Middleware Protocol Manager
318 Java Virtual Machine
202 Network Communications Program
100 Palm Sized Computer
Jini Discovery Protocol
220 Directory of Services
Service Objects
222 PrintServiceObject
222 FaxServiceObject
222 SlidePresentationServiceObject
222 EmailServiceObject
Jini Discovery Protocol
110 Network
251 Service Device Operating System
260 Service Control Application
364 PowerPoint Control Protocol Manager
266 Service Middleware Protocol Manager
368 PowerPoint
252 Service Device Communications Program
250 Network Based Computer Service

FIG. 3

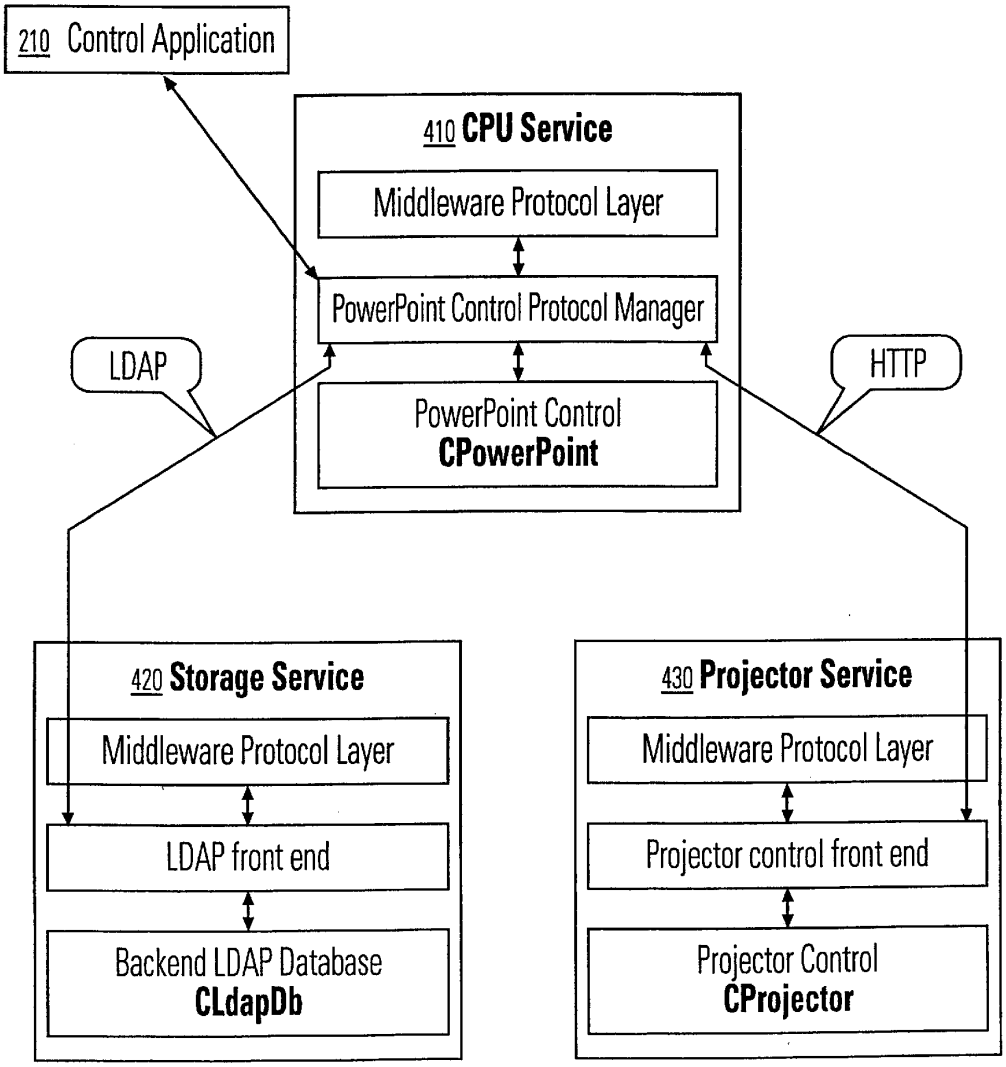

210 Control Application
410 CPU Service
Middleware Protocol Layer
PowerPoint Control Protocol Manager
PowerPoint Control
CPowerPoint
LDAP
HTTP
420 Storage Service
Middleware Protocol Layer
LDAP front end
Backend LDAP Database
CLdapDb
430 Projector Service
Middleware Protocol Layer
Projector control front end
Projector Control
CProjector

FIG. 4

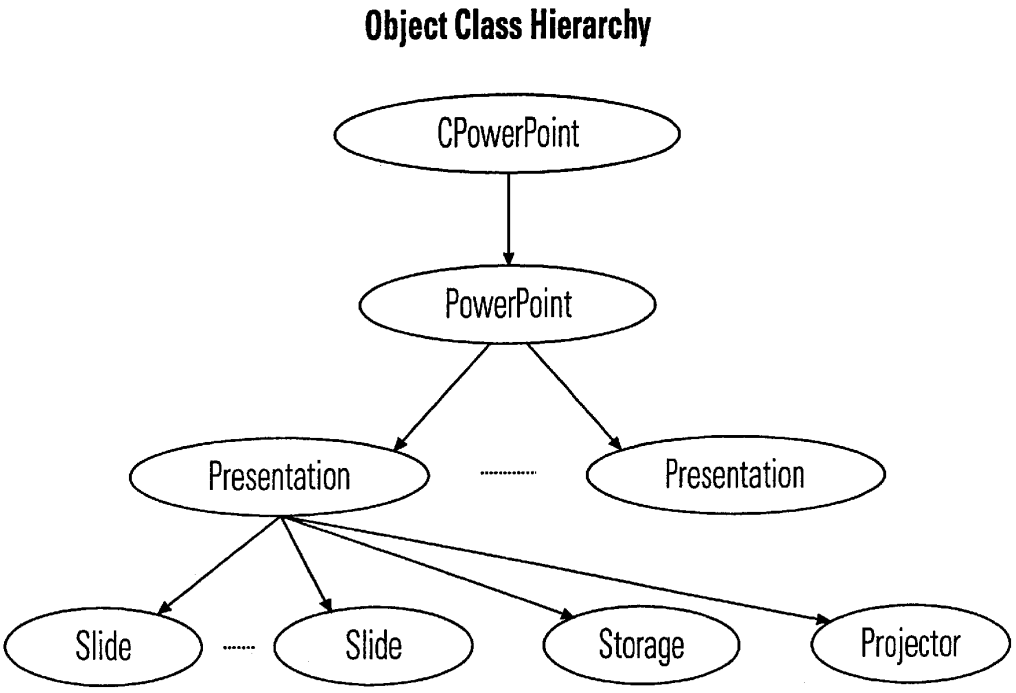

Object Class Hierarchy
CPowerPoint
PowerPoint
Presentation
Presentation
Slide
Slide
Storage
Projector

FIG. 5

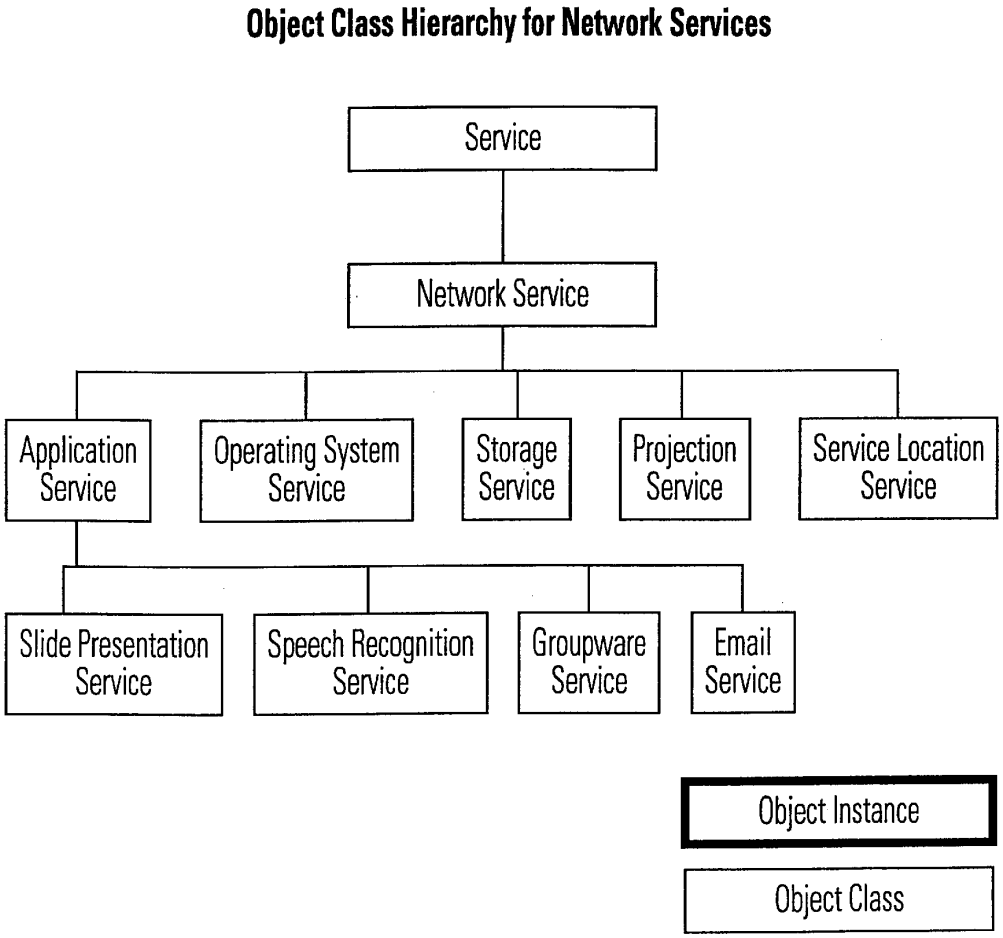

Object Class Hierarchy for Network Services
Service
Network Service
Application Service
Operating System Service
Storage Service
Projection Service
Service Location Service
Slide Presentation Service
Speech Recognition Service
Groupware Service
Email Service
Object Instance
Object Class

FIG. 6

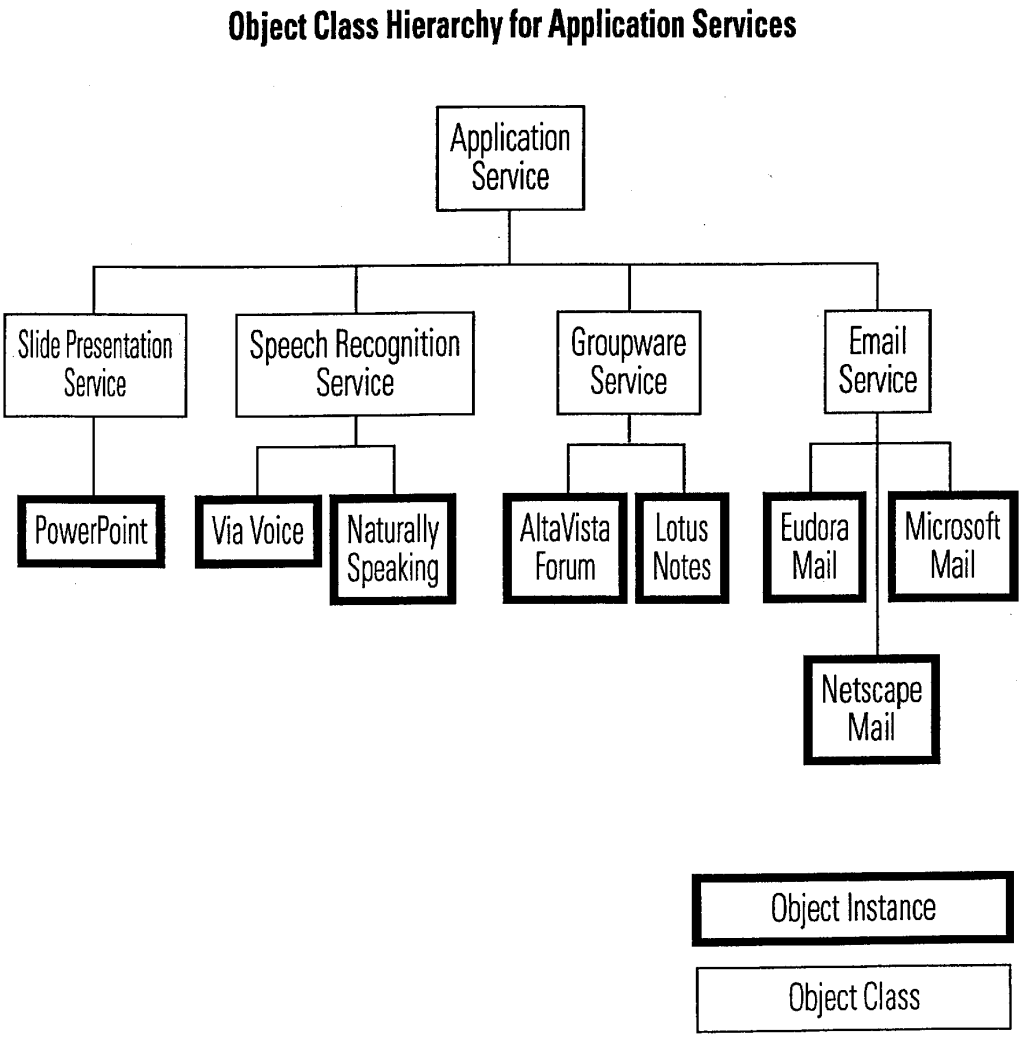

Object Class Hierarchy for Application Services
Application Service
Slide Presentation Service
Speech Recognition Service
Groupware Service
Email Service
PowerPoint
Via Voice
Naturally Speaking
AltaVista Forum
Lotus Notes
Eudora Mail
Microsoft Mail
Netscape Mail
Object Instance
Object Class

FIG. 7

SYSTEM AND METHOD USING A PALM SIZED COMPUTER TO CONTROL NETWORK DEVICES

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates to the field of networked computer. In particular, the invention relates to a system and method to control network devices using a palm sized, or otherwise reduced functionality, computer.

2. Description of the Related Art

Palm sized computers, also referred to as Personal Digital Assistants (PDAs), are portable devices which perform an array of personal management tasks such as calendar management and address book storage. The adoption of palm sized computers has been rapid. Some palm sized computers are able to interface with conventional computing devices, such as PCs, on an as-needed basis. For example, palm sized computers such as 3Com's Palm Platform™ computers can upload personal appointments to a PC-based calendar.

Palm sized computers generally have the following characteristics. Relative to desktop and laptop computers, palm sized computers have limited processing, display and input capabilities. As a result of these limitations, palm sized computers do not run the same applications as desktop or laptop computers. Other limitations of palm sized computers include limited battery life and lower bandwidth communications with other devices.

One big advantage of palm sized computers is their portability. Therefore, it is desirable to be able to access desktop functionality from palm sized computers.

SUMMARY OF THE INVENTION

Controlling network services using palm sized computers is described. A program on the palm sized computer is used to access a registry of network services that may be available. The registry includes descriptions for various services. Each description includes at least a reference to program code that can be downloaded to the palm sized computer. Executing this program causes the palm sized computer to issue commands directly to the specific network services needed. In some cases, these network services include application services for running desktop applications that the palm sized computer could not execute.

In some embodiments, the device executing the network services and the palm sized computer are executing middleware applications for communicating with the registry. In some embodiments, this middleware includes Jini technology from Sun Microsystems. Additionally, the programs downloaded can include Java program code.

BRIEF DESCRIPTION OF THE FIGURES

FIG. **1** illustrates a system having a palm sized computer controlling operations of various network devices.

FIG. **2** illustrates a software architecture for use in the system of FIG. **1**.

FIG. **3** illustrates an example architecture for controlling a PowerPoint presentation.

FIG. **4** illustrates a detailed software architecture for the example of FIG. **3**.

FIG. **5** illustrates a class hierarchy that can be used in the example of FIG. **3**.

FIG. **6** illustrates an example class hierarchy for network services that can be provided in the system of FIG. **1**.

FIG. **7** illustrates an example class hierarchy for application services that can be provided in the system of FIG. **1**.

DETAILED DESCRIPTION

A. System Overview

A palm sized computer can serve as a network portal to usher in a new generation of mobile computing. Palm sized computers are the ideal next generation computing device in that they are inherently mobile and have the lightweight form factor necessary for mobile computing. The fundamental obstacle to palm sized computers as the next generation computing device can be removed by viewing the network as an extension of the palm sized computer's resources. Functions can be downloaded into the device as needed, and overlaid after they have been used. This allows the palm sized computer to adapt to a changing environment (as the mobile user's location changes) and to access exactly the set of services it needs. These services are stored on the network and can be used at will. Many of these services may never reside on the device and are more suitable for execution by a conventional computer. However, they are accessible and can be controlled via a lightweight computing device, such as a palm sized computer.

B. Building Blocks of Network-Enabled Palm Sized Computers

The building blocks for lightweight mobile computing include a palm sized computer, a compute/memory/storage-intensive device(s), and a network.

Middleware allows palm sized computers to discover network-based computing resources. Once discovered, this middleware provides a mechanism for the palm sized computer to use these resources. This middleware typically includes a directory of resources (or services), a protocol for storing and retrieving from the directory, and mechanisms to transfer software from the directory to a palm sized computer.

Control applications reside on a lightweight computing device, such as a palm sized computer, but manipulate computing services on the network. Control applications assume the use of middleware, such as Sun Microsystems Jini, for access to network-based resources. (Note in some embodiments, the control application is distributed between the palm sized computer and a control proxy server.)

Example Registry and Control Infrastructure: Jini

Jini™ is a technology developed by Sun Microsystems which addresses the problem of computing and network complexity. It eases the burden of accessing services by providing seamless access and automatic registration of network resources. Jini accomplishes this by adopting a model in which devices announce their presence and capabilities to the network, and access the same type of information in order to locate services they wish to use.

The Jini approach relies on Java and a Jini "registry" (i.e. database of services) as the underlying infrastructure. Each device is expected to run a Java Virtual Machine (JVM), or rely on a Jini proxy which runs a JVM on the device's behalf. Key to Java is the idea that software (as well as data) can be dynamically downloaded to a device. The Java model assumes a distributed, network-centric model in which the behavior of a device can be dynamically altered to accommodate changing conditions.

Jini eases network connectivity problems. Jini acts as middleware to access network resources, as it lets devices locate services and download software for those services. Other middleware could be substituted for Jini if it provides discovery and software download for network-based services.

C. Definitions

A control device is a device which runs Jini (or some other discovery and software download technology) and is capable of accessing a wide range of network-centric services, including services which are incapable of running on the control device. The control device is the access device for a wide class of computing services otherwise inaccessible to devices with restricted computing power. 3Com's palm sized computer, the Palm Computing platform, is an example of such a control device. Windows CE compatible devices may also be used.

A control application is as an application that relies on resources located off of the control device (e.g. on a network), but uses a control device to initiate and control the application. The resources are accessed and controlled, but not resident, on the control device. Examples of such compute/memory-intensive services include PowerPoint slide presentations and speech recognition systems. These services can be both invoked and controlled via a control device.

Network-based services can be used by any control device. Services offer a discrete task to the control device. This contrasts with a traditional model in which devices, such as a PC, house the entirety of resources a control device might utilize. Services are fine-grained and represent a distributed set of capabilities residing on a network. Services may or may not be co-located with other services on the same physical device. Services are offered (via a network and a Directory of Services, such as the Jini Lookup) in as small a unit as is justifiable given the application tasks users will typically want to accomplish. Fine-grained services can be used by a variety of consumers who need precisely that function. The model that emerges is a network-based model that offers a wide range of narrowly-defined computing services of interest to a wide range of applications. Services will be physically distributed across devices but accessible via a central repository (e.g. database) of services. This model of distributed computing is inherently scalable in that the set of services offered via the network grows seamlessly as devices "plug in" and announce their capabilities.

D. An Example of the Control Application

To help illustrate the use of a control device and a control application, a PowerPoint slide presentation control is described herein. (FIG. **1** illustrates a generalized view of such an example.) In this example, a palm sized computer **100** accesses network **110** resources (via a Directory of Services, such as the lookup service **120**) to locate the elements it requires to remotely control a presentation located on the network. The palm sized computer **100** uses three services to function as a control device to display a PowerPoint presentation:

an application service **130** (to run PowerPoint)

a persistent storage service **150** (to store the presentation)

a display service **140** (to display the presentation images)

Each of these services will have registered with the lookup service **100**. A service in this example, is tied to different hardware devices. However, this is not required. Storage, processing, and/or display could be supported by one or more computer systems.

None of these services are resident on the palm sized computer **100**. Once the palm sized computer **100** has located the necessary services, it downloads the code required to control those services (using the lookup and download protocols). Middleware, such as Sun's Java/Jini technology, is used to move the code.

The palm sized computer **100** is then capable of directly controlling the services it requires.

In this example, all the devices can communicate on the network **110**, however, they need not all be connected to the network in the same way.

Functionally, the devices play the following roles:

palm sized computer **100**: the remote control device

application services: a set of resources and services, such as compute power and applications

network **110**: the physical medium which connects devices and services

lookup service **120**: a database of network services

E. Architecture

As noted above, the palm sized computer **100** functions as the remote control device for the PowerPoint presentation. It is a multi-function control device in that it can control a host of resources accessible via the network. The palm sized computer **100** accomplishes this via middleware (e.g. Jini) and a generic control protocol capable of issuing control commands to an offboard resource. The palm sized computer **100** implements this control via the software components depicted in FIG. **2**. These software components represent a generic architecture for control of any network-based resource using a reduced functionality computer such as a palm sized computer. The software architecture, middleware, and control protocol implement a new model for lightweight mobile computing. This model of lightweight mobile computing is particularly well-served by devices such as a palm sized computer, as they possess the requisite size and portability.

In FIG. **2**, the control device **200** is able to control services on the network **110**, such as the network based service **250**. The directory of services **220** provides the registry functions used to enable the system.

The control device **200** includes a control device operating system **201** which supports a network communications program **202** and a control application **210**. The network communications program **202** allows the control device **200** to communicate with other devices on the network. The control application **210** includes the following elements:

a GUI **212** to display the available network-based services and accept user input

an application control protocol manager **214** which interfaces between the control device **200** and the network based computer service **250** by requesting tasks from that service (e.g. slide manipulation). This module is responsible for generating the application control protocol to command the selected service,

a middleware protocol manager **216** to transfer data between the control device **200** and the directory of services **220** (e.g. communication between Palm platform and the Jini Lookup). This module is responsible for generating the syntax necessary to communicate based on the type of middleware being used.

The service device operating system **251** includes a service device communications program **252** and a service control application **260**. The service device communications program **252** is responsible for communicating with the network **110**. The service control application **260** includes the following components:

a service application control protocol manager **262** that interfaces between the network based computer service **250** and the control device **200** and accepts control tasks (e.g. slide manipulation) issued from the control device **200**. This module is capable of sending back a response (e.g. status) to the control device **200**,

a middleware protocol manager **266** to transfer data between the network based computer service **250** and

the directory service (e.g. communication between the application service **130** and the Jini Lookup). This module is responsible for generating the syntax used to communicate based on the type of middleware being used.

an application **268** to perform functions on the network. The application **268** can be a desktop application that would not execute on a palm sized computer.

Returning to the specific example of the PowerPoint presentation, FIG. **3** illustrates the architecture that could be used to implement such a system. In FIG. **3**, the control device **200** has been replaced with a palm sized computer **100** executing the Palm OS **301**. The GUI **212** is specifically for PowerPoint control (see FIG. **1** for an example of such a GUI). The Java Virtual Machine **318** is executing on the palm sized computer **100** and replaces the middleware **218**. The middleware protocol manager **216** supports Jini discovery, lookup and download protocols. PowerPoint control commands are issued to the network based computer service **250**. The PowerPoint control protocol manager **364** provides the interface for these commands and controls the PowerPoint application **368**.

F. Control Flow

The process for accessing and controlling network-based services is described below. The specific implementation control of a PowerPoint Service via a 3Com Palm Computing platform is used as an example. Refer to FIG. **1**.

1. Configure a computer hosting the directory of services **220** and connect it to the network **110**. For instance, a Jini lookup service is configured to listen at a preset TCP, or UDP, port for service registration or service lookup requests.
2. Establish a network connection from the network based computer service **250** to the network **110**.
3. Register the computing service with a directory of services **220**. For example, in the Jini model, the computing service initially sends out a multicast packet announcing its presence on the network. Once the directory service **220** receives the packet, it sends a unicast packet back to the announcer. The packet includes an interface (e.g. Java code) for uploading code back to the directory service **220** as well as for searching and downloading code from the directory service **220**.
4. Upload service description to the directory service **220**. If Jini is adopted as the middleware, the application service **130** receives the response from the directory service **220** and uses the included interface to upload its service interface to the directory service **220**. When the service interface is called, it contacts the directory service **220** which in turn creates an entry (represented by an object) for this new service and sets the proper fields such as service name, attributes and optionally the corresponding service interface. Other middleware may choose to use protocol-based approaches such as FTP or TFTP for the uploading process.
5. Register the storage service **150** and display service **140** via the same process.
6. Establish a network connection from the control device **200** to the network **110**. For Palm computers, there are multiple options for network connectivity. Possible solutions include using the infrared (IR) port to talk to a IR-LAN bridge or router, using the serial port to talk to a serial-to-LAN bridge or router, using either the IR or the serial port to talk to a digital cell phone and dial up a modem server, and/or using wireless data communications.
7. Launch the service control graphical user interface (GUI) **212** on the control device **200**.
8. Via the service control GUI **212**, accept user input, such as the selection of an application (for example, a PowerPoint application) to be controlled.
9. Optionally register the control device **200** with the directory service via a registration protocol, such as the Jini Discovery Protocol. This step is the same as the above one for the other services. It is executed only if the control device **200** has resources to offer.
10. Search the directory service **220** and download the desired service descriptor. In the case of Jini, after the control device **200** receives the response from the directory service **220**, it uses the included interface to search the directory service **220** for an application service using the object type representing the service (such as an object type of PowerPoint presentation service) and the desired service attributes (such as the name and the physical location of the service). Once the directory service **220** finds such a service entry, the control device **200** is notified, which in turn uses the downloading interface to download the application service descriptor. One example of these services is the GUI code for controlling a PowerPoint presentation.
11. Send requests from the control device **200** to the network based computer service **250** to control the desired application. For example, a "next slide" request could be sent from a palm sized computer **100** to an application service **130** running PowerPoint. The communication can be based on a protocol such as the following one:

Control Type	Application	File Name	Control Function

Where:

Control Type={Request, Reply}

Application={PowerPoint, FAX, Print, Email, Phonebook, . . . }

File Name={3ComPalmVIISpecification}

Control function={File Load, Slide Forward, Slide Backward, File Close, . . . }

Alternatively, techniques such as Java's Remote Method Invocation (RMI) can be used to achieve the same goal. In this case, the control device makes a local function call such as doForwardSlide( ). The RMI mechanism will transfer the call to a remote machine which implements and carries out the function call. The PowerPoint presentation service may in turn use other services such as the storage service **150** and the display service **140**. The procedure to employ these services is similar to steps 10–11 above.

12. Accept any response to requests sent from the control device **200** to the network based computer server **250** and process any errors.

G. Control Device GUI

An important element of the control application **210** is a GUI front-end which accepts user input for controlling the PowerPoint presentation (or other application) and a control protocol manager backend which takes user input and translates it into commands to the CPU service. An example GUI is depicted in FIG. **1**. The example GUI allows the user to click on "forward", "backward", "go-to-first-page" or "go-to-last-page" buttons to control the slide show. The user can also click a "get-list" button to get a full list of the slide titles in the current presentation and choose to go to a particular slide. By clicking the "scribble" button, the window switches to graphics mode. In this mode, the user can draw at random on the panel, and the result of the drawing will be sent to the CPU service and eventually displayed on the projection service.

As explained in the software flow section, there are multiple ways to implement the application control protocol manager **214**, the following illustrates one approach. The protocol takes the following form:

Control Type	Application	File Name	Control Function

Where:

Control Type={Request, Reply}

Application={PowerPoint, FAX, Print, Email, Phonebook, . . . }

File Name={3ComPalmVIISpecification}

Control function ={File Load, Slide Forward, Slide Backward, File Close,. . . . }

When the user clicks any button or draws something on the GUI, the application control protocol manager **214** generates the corresponding field in the protocol and sends a command to the network based computer service **250** via a TCP/IP channel.

H. Service Control Application Design

FIG. **4** illustrates a detailed software architecture for the service control application of FIG. **3**. The architecture includes three main elements: a CPU service **410** (corresponding to the application service **130**), a storage service **420** (corresponding to the storage service **150**) and a projector service **430** (corresponding to the display service **140**). Each of the elements include middleware protocol layer management modules. Each module has a corresponding service application control module (e.g., PowerPoint Control Protocol Manager). To control specific network services, instances of those services are instantiated and are used by the corresponding control protocol manager. The following describes examples of such classes that can be instantiated for specific network services.

PowerPoint Control

The PowerPoint control, through OLE automation, instantiates and controls an instance of PowerPoint application. The PowerPoint control also communicates with the storage service **420** to store/retrieve presentations and the projector service **430** to view slides.

The PowerPoint control has the ability to have many presentations open at the same time and is capable of switching between presentations. A presentation has a collection of slides in it. Once open, the PowerPoint control allows easy traversal of the slides either by commands like previous, next slide or by direct access (e.g. slide number or slide title). When any change in slide position occurs, the PowerPoint application automatically generates the image that needs to be shown by the projector control and invokes the method on the projector control to show the changed slide.

Once a presentation is open, the PowerPoint control can have facilities such as Add Comments and Add Scribbled Graphics to the current slide. Additionally, it can allow the adding of new slides to the presentation.

FIG. **5** illustrates an example object class hierarchy for the PowerPoint control class. The following describes the elements of FIG. **5** in more detail.

Property/Method Name	Type	Description
	PowerPoint	
Presentations	Collection	List of open presentations
	Presentation	
Slides	Collection	Array of slides in presentation
Location	Storage	Location where presentation loaded from
View	Projector	Projector where presentation is viewed
	Slide	
Title	String	Title for the slide
AddComments	Method	Add given comments to slide at given coordinates
AddScribble	Method	Add given graphics to slide at given coordinates
	Storage	
Host	String	LDAP Server host where the presentation stored
FileName	String	File name and other details for presentation
	Projector	
Host	String	Projector host where slides shown
ShowSlide	Method	Show slide image on projector

LDAP Database Control

The LDAP database control provides file system services to store and retrieve presentations. This control gets commands from the PowerPoint control through the LDAP protocol, to search for presentations and return presentations.

An object class hierarchy for CLdapDB could include CLdapDb having to a specific storage system reference.

Storage		
Property/Method Name	Type	Description
FileFilter	String	File filter used to get file list
FileList	Collection	List of files found on storage

Projector Control

The projector control provides image viewing services and has a simple image viewer that shows the image on a projection screen, monitor, display device or canvas. The control gets commands from the PowerPoint control. Specifically, to display images, the projector control could cause frames generated by the PowerPoint control to be displayed at the device controlled by the projector control. Other embodiments use more sophisticated techniques for displaying the PowerPoint information (e.g., support windowing system API calls that an application may make).

An object class hierarchy for CProjector could include CProjector having to a specific image view reference.

ImageView

Property/Method Name	Type	Description
FileName	String	File name to be displayed
View	Method	Show the image on the projector
Mode	Long	FullScreen mode

I. Directory of Services Design

The directory of services **220** encodes the set of services available on the network **110**. The directory of services **220** describes the characteristics of these services and provides a means to locate those services. To illustrate this concept, an object-oriented directory service is used. The directory service will have objects whose attributes describe the features of available services and optionally include either code to invoke those services or a reference to such code. A directory service will typically be one of several components offered in the middleware a control device **200** will use.

FIG. **6** illustrates an object class hierarchy which models several network-based services. These network-based services are Application Service, Operating System Service, Storage Service, Projection Service, and Service Location Service. Examples of each of these services are now given.

Define-Class Service
  Superclass: root
  Attributes:
    Name
    Status
Define-Class NetworkService
  Superclass: Service
  Attributes:
    Location
    PhysicalAddress
    PhysicalMachine
    Owner
    Vendor
    Version
Define-Class ApplicationService
  Superclass: NetworkService
  Attributes:
    CodeLocation
    SerialNumber
    SupportedFeatures
Define-Class OperatingSystemService
  Superclass: NetworkService
  Attributes:
    RealTimeOS?
Define-Class StorageService
  Superclass: NetworkService
  Attributes:
    DatabaseType
    Schema
Define-Class ProjectionService
  Superclass: NetworkService
  Attributes:
    Resolution
Define-Class ServiceLocationService
  Superclass: NetworkService
  Attributes:
    QueryProtocol
    Schema

FIG. **7** illustrates a partial object class hierarchy which models Application Services, including the PowerPoint Application Service:

Define-Class SlidePresentationService
  Superclass: ApplicationService
Define-Class GroupwareService
  Superclass: ApplicationService
Define-Class EmailService
  Superclass: ApplicationService
Define-Class SpeechRecognitionService
  Superclass: ApplicationService

Based on this object class hierarchy, objects which represent network services can be defined. These objects are stored in a directory of services **220**, such as a Jini Lookup.

An object to instantiate a SlidePresentationService could look as follows:

Make-Instance SlidePresentationService
  Name "PowerPoint"
  Status "Active"
  Location "3Com Intranet"
  PhysicalAddress "Building300.Floor2.Cube323"
  PhysicalMachine "PowerBook G3"
  Owner "Elaine Lusher"
  Vendor "Microsoft"
  Version "98"
  CodeLocation "system/applications/office/powerpoint:"
  SerialNumber "169-43-4666"
  SupportedFeatures "Scribble"

Other examples of objects which represent application services include:

Make-Instance SpeechRecognitionService
  Name "Naturally Speaking"
  Status "Active"
  Location "3Com Intranet"
  PhysicalAddress "Building300.Floor2.Cube100"
  PhysicalMachine "Solaris"
  Owner "Wenjun Luo"
  Vendor "Dragon Systems"
  Version "4.1"
  CodeLocation "system/applications/research/dragonspeech"
  SerialNumber "157-89-4323"
  SupportedFeatures "Dictation for Microsoft Word"
Make-Instance GroupwareService
  Name "Alta Vista Forum"
  Status "Active"
  Location "3Com Intranet"
  PhysicalAddress "Building300.Floor2.Cube220"
  PhysicalMachine "Windows NT"
  Owner "Paul Huard"
  Vendor "Microsoft"
  Version "3.1"
  CodeLocation "http://3Community/code/groupware/latest"
  SerialNumber "444-56-7777"
  SupportedFeatures "Virtual Chat Room"
Make-Instance EmailService
  Name "Netscape Mail"
  Status "Active"
  Location "3Com Intranet"

PhysicalAddress "Building300.Floor2.Cube300"

PhysicalMachine "Solaris"

Owner "Rick Nottingham"

Vendor "Netscape"

Version "5.0"

CodeLocation "http://3Community/code/email/latest

SerialNumber "456-34-6786"

If middleware resides on a proxy device (rather than on the control device **200**), the control device **200** will need to locate such a proxy service. An Operating System Service class can encode services such as a JVM Service, a Linux Service, or a Jini proxy.

Make-Instance OperatingSystemService

Name "Jini Proxy"

Status "Active"

Location "3Com Intranet"

PhysicalAddress "Building300.Floor2.Cube120"

PhysicalMachine "Solaris"

Owner "Rick Nottingham"

Vendor "Sun"

Version "5.0"

RealTimeOS? "no"

Finally, directory services **220** (such as the Jini Lookup) are modeled. This could be encoded in a class such as the Service Location Service.

Make-Instance ServiceLocationService

Name "Jini Lookup"

Status "Active"

Location "3Com Intranet"

PhysicalAddress "Building300.Floor2.Cube150"

PhysicalMachine "Solaris"

Owner "Rick Nottingham"

Vendor "Sun"

Version "5.0"

QueryProtocol "Jini Lookup Protocol"

Schema "Service Directory Schema 1.0"

J. Variations on the Network-based Control Application

Several variations of the general control paradigm can be defined:

The middleware (Jini, in our example) may not be physically resident on the control device **200**. In this case, a proxy is used which runs middleware on behalf of the control device **200**. Functionally speaking, the same design will still apply. One change to the design can be used to support a modified GUI for the control device. In these embodiments, the GUI program is modified by the proxy device to account for specific limitations of the control device. Also, in some embodiments, the control device may include preset applications that interface directly with the proxy device. Such an architecture would support a limited set of network services but would likely result in very small and lightweight applications on the control device.

The control paradigm is not limited to palm sized computers. Any computing device with restricted computing power could be used as a control device **200** for any network-based resource. Switches, hubs, routers, and other networking devices are candidates for a control device **200**. The network-based resources they use could include any service that cannot physically reside on the networking device due to restrictions such as limited memory.

Some embodiments of the invention include only the control device and the programs on the control device; other embodiments include some and/or all of the programs in computer readable media, or in electromagnetic waveforms.

K. Conclusion

The foregoing description of various embodiments of the invention has been presented for purposes of illustration and description. It is not intended to limit the invention to the precise forms disclosed. Many modifications and equivalent arrangements will be apparent.

What is claimed is:

1. A method of controlling a service on a network using a palm sized computer, the palm sized computer being coupled in communications with the network, the method comprising:

accessing a description of the service from a directory of services, the description of the service including at least a reference to program code for controlling the service;

downloading the program code to the palm sized computer;

the palm sized computer executing at least a portion of the program code; and

sending control commands to the service from the palm sized computer in response to the executing, wherein the service controls an application that cannot be executed on the palm sized computer.

2. The method of claim **1** further comprising registering the service in the directory of services by storing the description of the service in the directory of services.

3. The method of claim **1** wherein the directory of services includes a Jini Lookup directory, wherein the accessing the description includes executing a Jini discovery protocol to locate the Jini Lookup directory and executing a Jini Lookup protocol to retrieve the description of the service.

4. The method of claim **1** wherein the description of the service includes an object reference corresponding to an object representing the service and a set of service attributes including the name of the service and the physical location of the service.

5. The method of claim **1** wherein the program code includes Java code and wherein the palm sized computer is executing a Java Virtual Machine to execute at least a portion of the program code.

6. The method of claim **1** wherein the program code includes code to implement a graphical user interface on the palm sized computer.

7. The method of claim **1** wherein the application includes a desktop program.

8. A method of controlling a program on a network device from a palm sized computer, the computer is not capable of executing the program by itself, the network device and computer being coupled in communications via a network, the method comprising:

accessing a directory of services, a service in the directory of services corresponding to the program, the description of the service including at least a reference to program code for controlling the service;

loading the program code;

issuing control commands to the network device using the program code, the control commands causing the network device to control the program.

9. The method of claim **8**, wherein loading the program code includes loading the program code onto the palm sized computer and the issuing the control commands includes the palm sized computer issuing the control commands.

10. The method of claim 8, wherein a proxy device is coupled to the network and wherein accessing the directory of services includes the palm sized computer accessing the proxy device, and the proxy device accessing the directory of services, and wherein the loading the program code includes loading the program code onto the proxy device, and wherein the issuing the control commands includes the palm sized computer issuing a set of first set of commands to the proxy device and the proxy device issuing the control commands.

11. The method of claim 10, wherein the program code includes a user interface program and wherein the proxy device receives the user interface program and generates a second user interface for the palm sized computer.

12. The method of claim 8 wherein loading the program code includes loading the program code onto the palm sized computer from the directory of services.

13. The method of claim 8, wherein the palm sized computer includes a Palm OS compatible computer, wherein the program code includes Java code and wherein the directory of services includes a Jini directory of services.

14. The method of claim 8 further comprising the network device registering the description of the service with the directory of services.

15. The method of claim 8 wherein the program includes a desktop program.

16. The method of claim 8 wherein the program includes a desktop program and wherein the services controlled include a CPU service for executing the program, a storage service for providing data to the CPU service and a display service for displaying information generated from the CPU service.

17. A system for controlling a network service comprising:

a network based computer service for controlling an application that cannot exectuted by a control device;

a directory of services including a registry of services, each service in the registry of services corresponding to a service on the network;

the control device having a program for sending control commands to the network based computer service, the program being loaded onto the control device as a result of locating a reference to the network based computer service in the directory of services;

a network coupled in communications with the network based computer service, the directory of services and the control device;

wherein the control device includes a palm sized computer having an operating system, a network communications program, a protocol program for communicating with the directory of services and wherein the program includes a graphical user interface.

18. The method of claim 17 wherein the network based computer service includes a computer having an operating system, a network communications program, a protocol program for communicating with the directory of services and the application.

19. A method for controlling a service in a network using a palm sized computer, the method comprising:

accessing a description of a service, the description of the service including at least a reference to program code for controlling the service;

downloading the program code;

executing at least a portion of the program code; and

sending control commands to the service in response to the executing, wherein the service includes a CPU service.

20. A system comprising:

means for accessing a description of a service, the description of the service including at least a reference to program code for controlling a service;

means for downloading the program code;

means for executing at least a portion of the program code; and

means for sending control commands to the service in response to the means for executing, wherein the service controls an application that cannot be executed on the means for executing.

* * * * *

US006161134A

United States Patent [19]

Wang et al.

[11] Patent Number: 6,161,134

[45] Date of Patent: Dec. 12, 2000

[54] **METHOD, APPARATUS AND COMMUNICATIONS SYSTEM FOR COMPANION INFORMATION AND NETWORK APPLIANCES**

[75] Inventors: **Peter Si-Sheng Wang**, Cupertino, Calif.; **Ismail Dalgic**, Mountain View, Calif.

[73] Assignee: **3Com Corporation**, Santa Clara, Calif.

[21] Appl. No.: **09/181,431**

[22] Filed: **Oct. 30, 1998**

[51] **Int. Cl.**[7] **G06F 13/00**

[52] **U.S. Cl.** **709/220**

[58] **Field of Search** 709/200, 205, 709/212, 218, 220, 222; 708/109; 455/550

[56] **References Cited**

U.S. PATENT DOCUMENTS

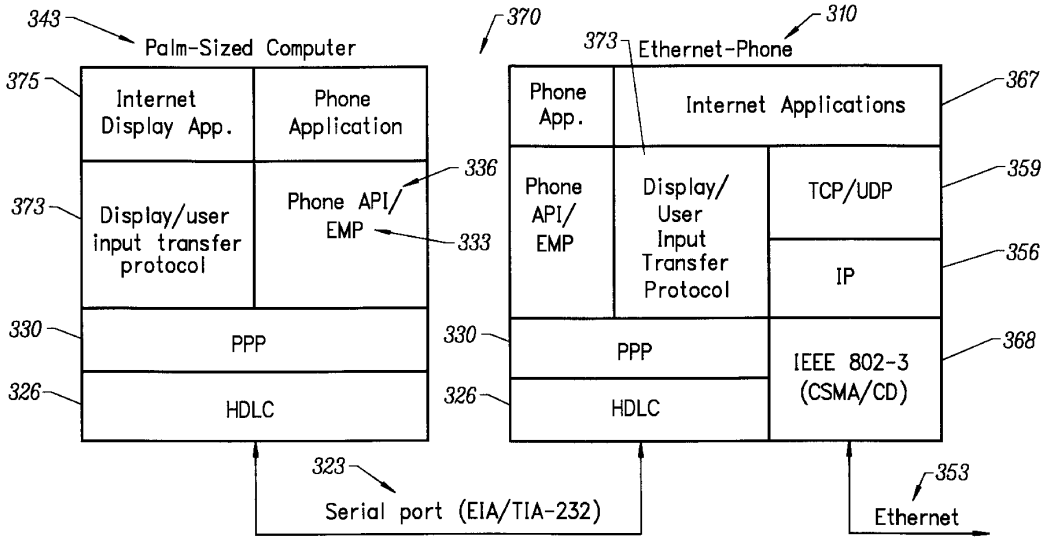
5,497,339	3/1996	Bernard	708/109
5,606,594	2/1997	Register et al.	455/550

Primary Examiner—Robert B. Harrell
Attorney, Agent, or Firm—David J. Weitz; Wilson Sonsini Goodrich & Rosati

[57] **ABSTRACT**

The invention provides an information appliance and a network appliance (or telephone) that function independently as well as with each other as companion appliances. The information appliance stores user information corresponding to a particular user. The telephone is linked to network. In some embodiments, the companion appliances are capable of simultaneously exchanging voice and data messages with devices connected to the network. The appliances are connected to each other physically through a communications port, and exchange data link layer formatted data corresponding to user personalized information, commands from the user, and responses including message status information corresponding to action of the network connected devices. The user information enables the telephone to perform network communications according to user specified settings, and enables the telephone to assume the user specific information appliance network identification. The information appliance is typically a portable computer and in some embodiments is a palm-sized computer. In some embodiments, the telephone is an Ethernet telephone. One aspect of the invention provides a method for transmitting data from a portable computer to a telephone. Other aspects of the invention include: a method for exchanging voice and data messages between a telephone and devices connected to a network, a portable computer adapted for connection to a telephone, a telephone adapted for connection to a portable computer, and a communications system including the telephone connected to the portable computer.

75 Claims, 26 Drawing Sheets



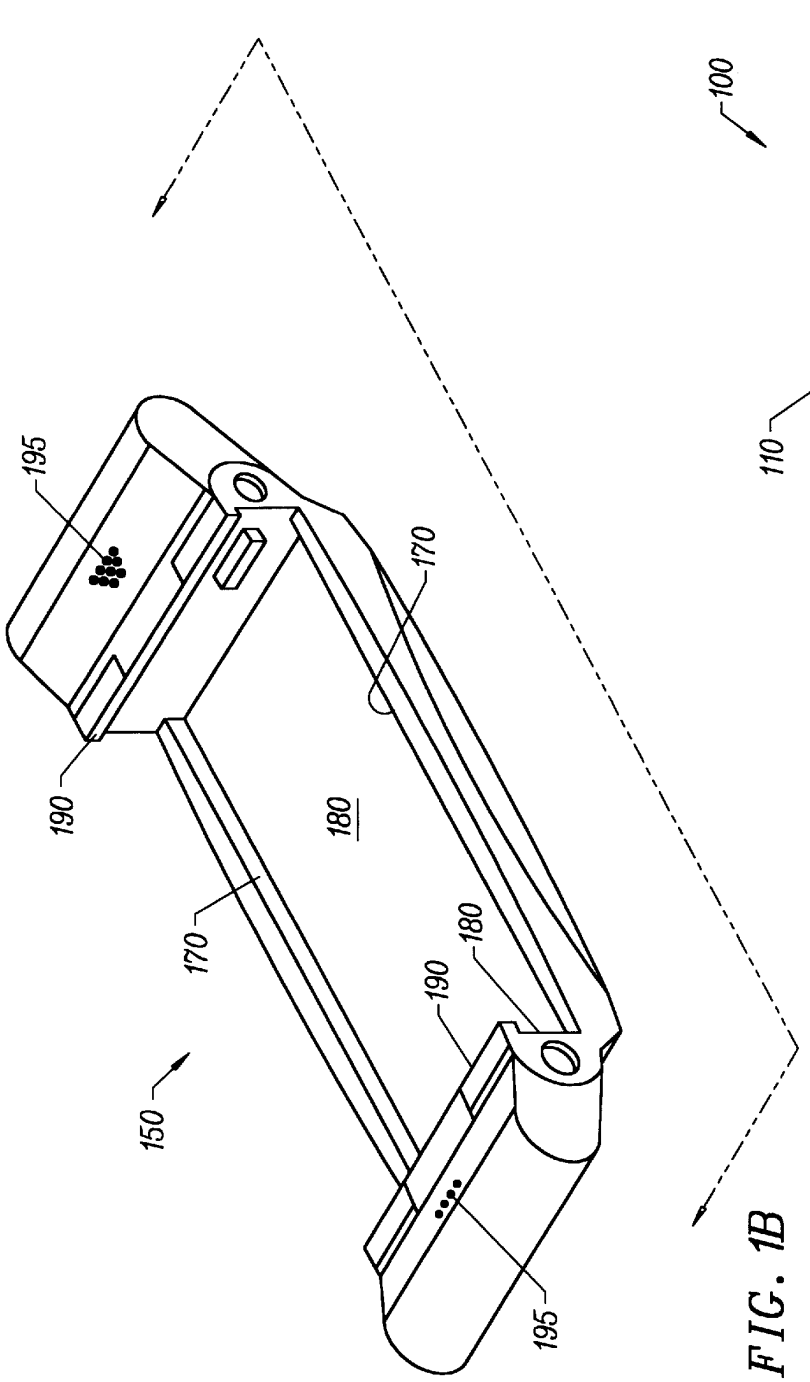


FIG. 1B

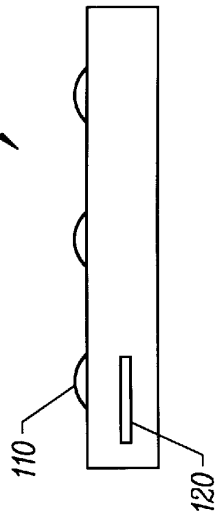


FIG. 1A

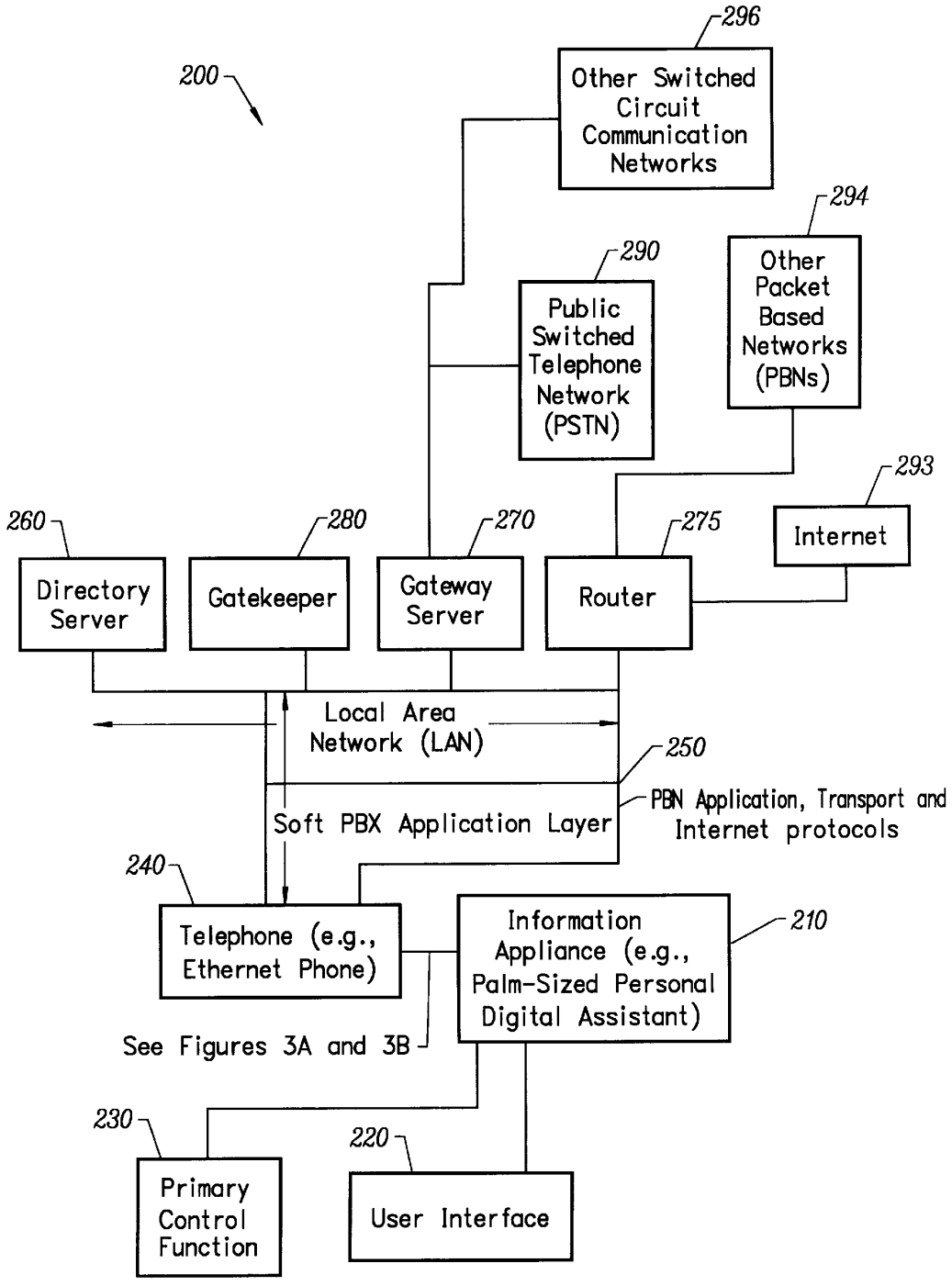

200
296
Other Switched Circuit Communication Networks
294
290
Public Switched Telephone Network (PSTN)
Other Packet Based Networks (PBNs)
293
Internet
260
Directory Server
280
Gatekeeper
270
Gateway Server
275
Router
Local Area Network (LAN)
250
Soft PBX Application Layer
PBN Application, Transport and Internet protocols
240
Telephone (e.g., Ethernet Phone)
Information Appliance (e.g., Palm-Sized Personal Digital Assistant)
210
See Figures 3A and 3B
230
Primary Control Function
220
User Interface

FIG. 2

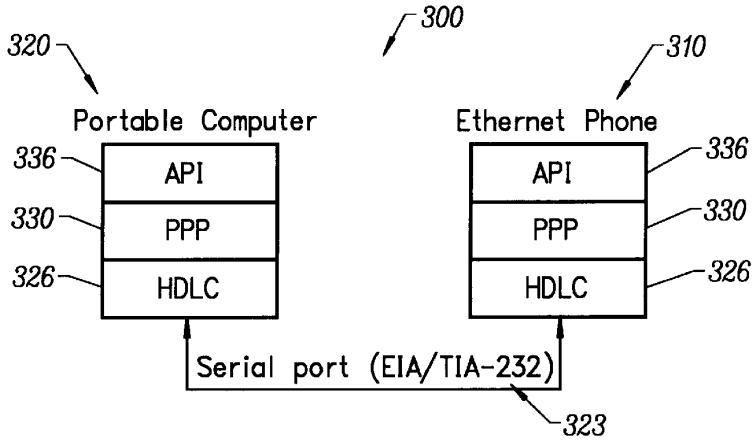

300
320
310
Portable Computer
Ethernet Phone
336
API
API
336
330
PPP
PPP
330
326
HDLC
HDLC
326
Serial port (EIA/TIA-232)
323

FIG. 3A

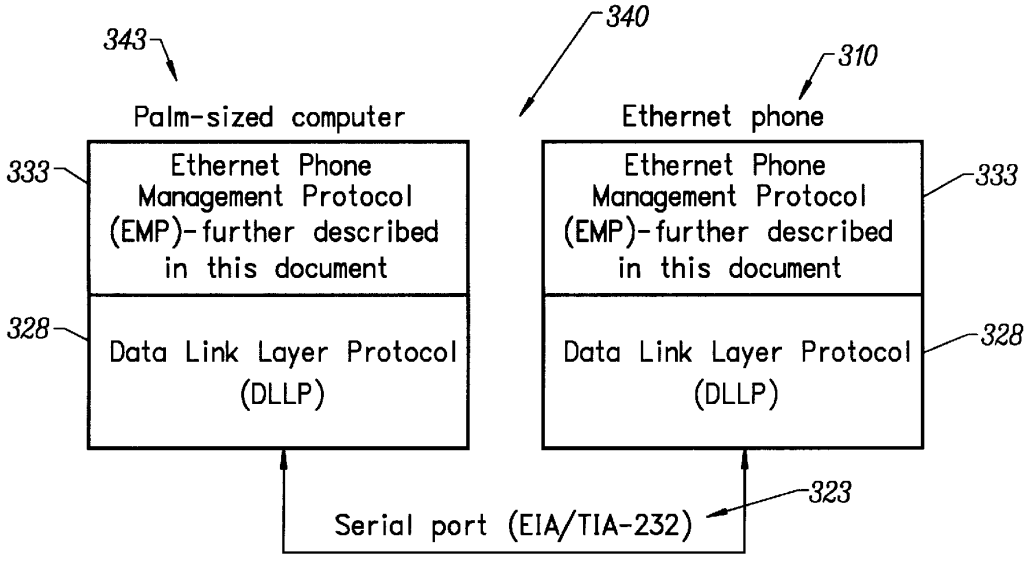

343
340
310
Palm-sized computer
Ethernet phone
333
Ethernet Phone Management Protocol (EMP)-further described in this document
Ethernet Phone Management Protocol (EMP)-further described in this document
333
328
Data Link Layer Protocol (DLLP)
Data Link Layer Protocol (DLLP)
328
323
Serial port (EIA/TIA-232)

FIG. 3B

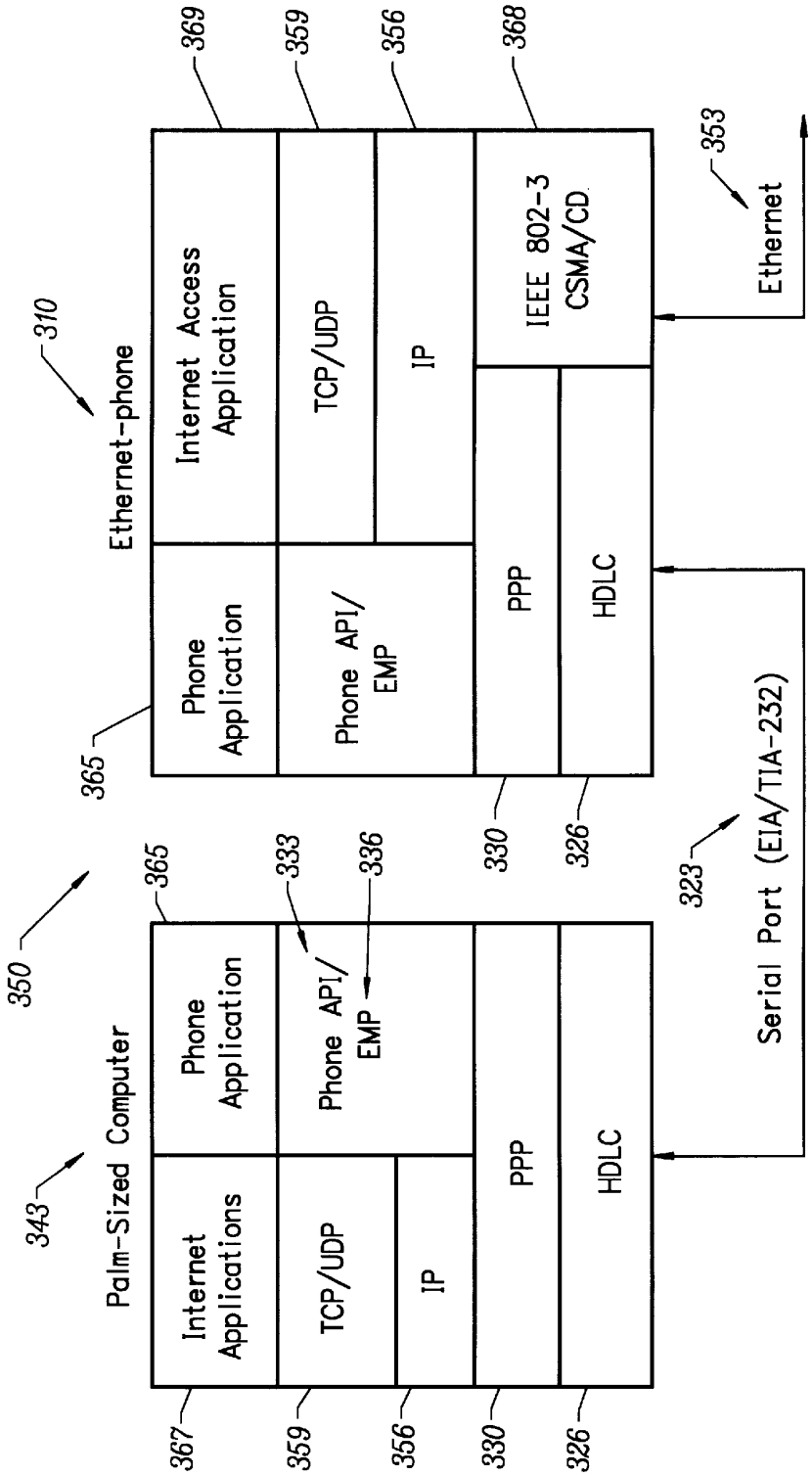

Palm-Sized Computer
Internet Applications
TCP/UDP
IP
Phone Application
Phone API/ EMP
PPP
HDLC
Ethernet-phone
Internet Access Application
TCP/UDP
IP
Phone Application
Phone API/ EMP
PPP
HDLC
IEEE 802-3 CSMA/CD
Ethernet
Serial Port (EIA/TIA-232)
310
343
350
353
365
367
359
356
330
326
333
336
323
369
368

FIG. 3C

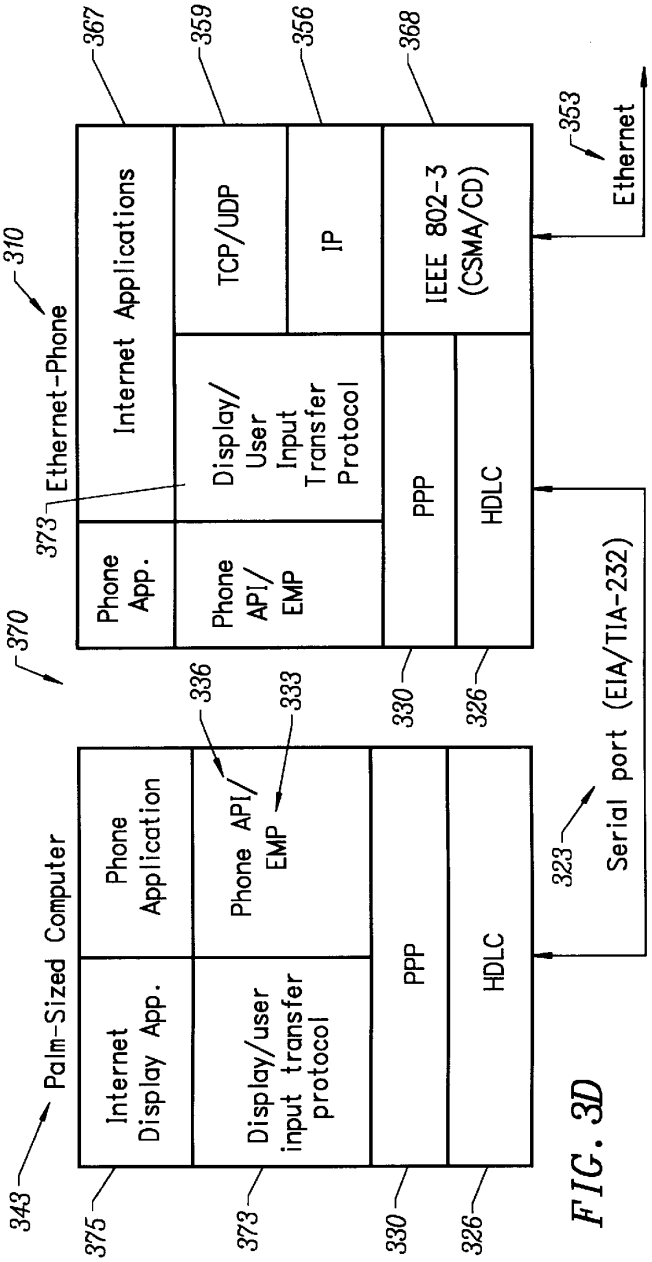

343
Palm-Sized Computer
Internet Display App.
Phone Application
375
Display/user input transfer protocol
Phone API/ EMP
373
336
333
PPP
330
HDLC
326
323
Serial port (EIA/TIA-232)
370
Ethernet-Phone
310
Phone App.
Phone API/ EMP
373
Display/ User Input Transfer Protocol
PPP
HDLC
Internet Applications
TCP/UDP
IP
IEEE 802-3 (CSMA/CD)
367
359
356
368
330
326
353
Ethernet

FIG. 3D

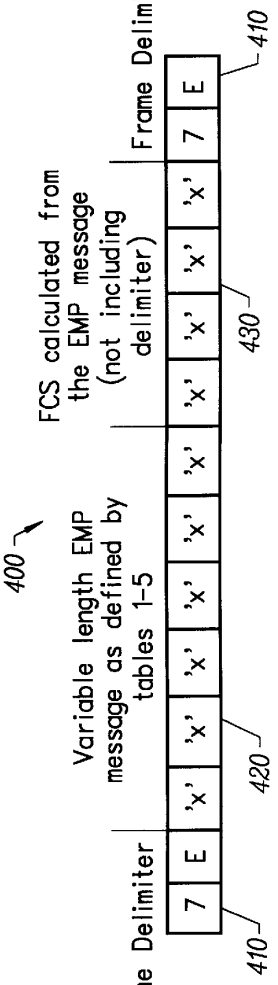

400
Variable length EMP message as defined by tables 1-5
FCS calculated from the EMP message (not including delimiter)
Frame Delimiter
Frame Delimiter
7
E
'x'
'x'
'x'
'x'
'x'
'x'
'x'
'x'
7
E
410
420
430
410

FIG. 4

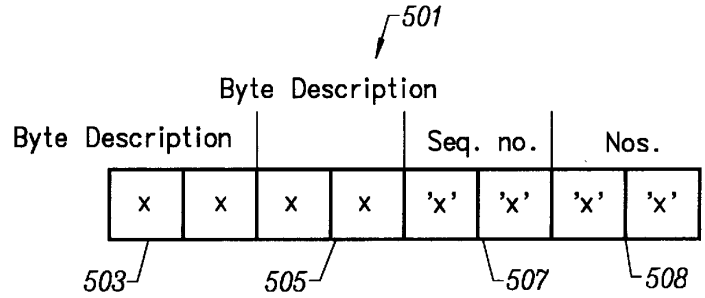


FIG. 5A

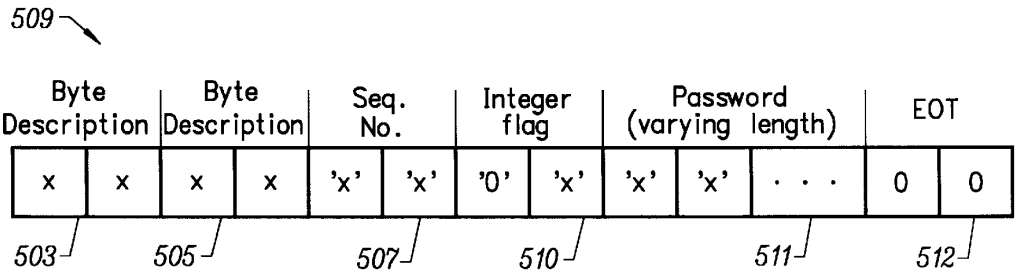


FIG. 5B

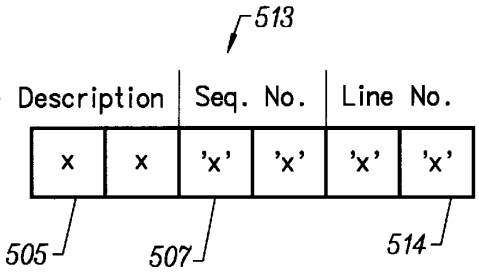


FIG. 5C

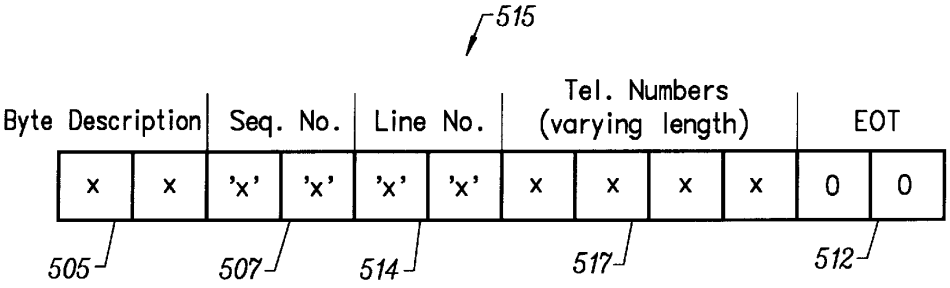


FIG. 5D

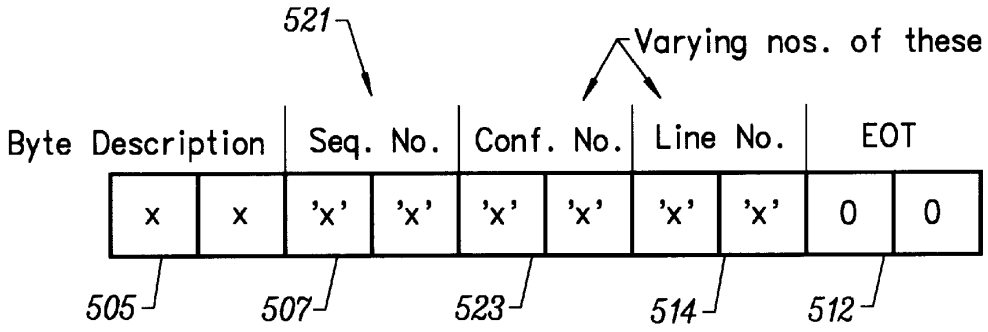

521
Varying nos. of these
Byte Description
Seq. No.
Conf. No.
Line No.
EOT
x
x
'x'
'x'
'x'
'x'
'x'
'x'
0
0
505
507
523
514
512

FIG. 5E

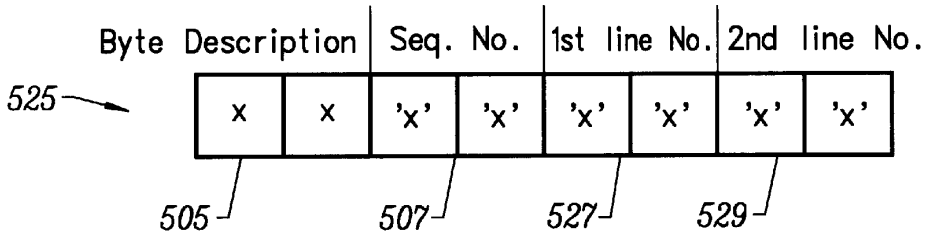

Byte Description
Seq. No.
1st line No.
2nd line No.
525
x
x
'x'
'x'
'x'
'x'
'x'
'x'
505
507
527
529

FIG. 5F

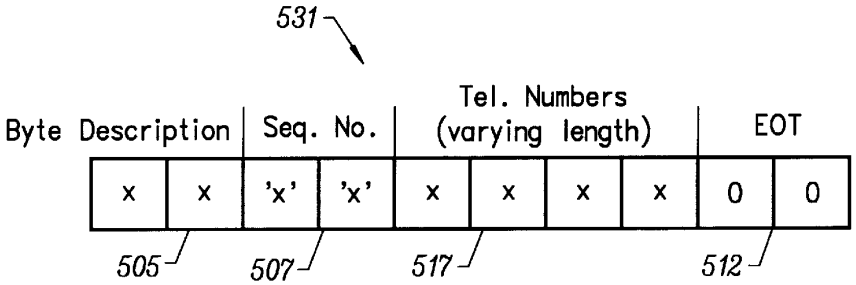


FIG. 5G

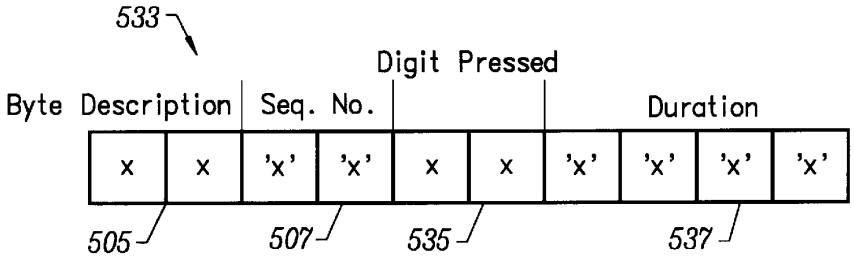


FIG. 5H

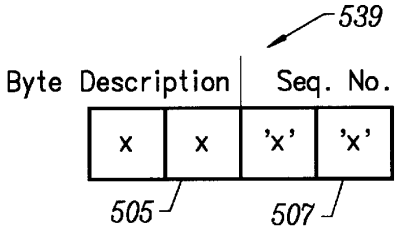


FIG. 5I

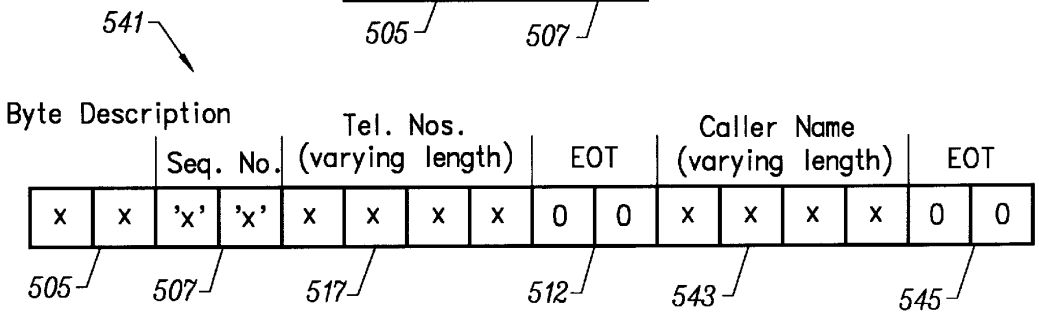


FIG. 5J

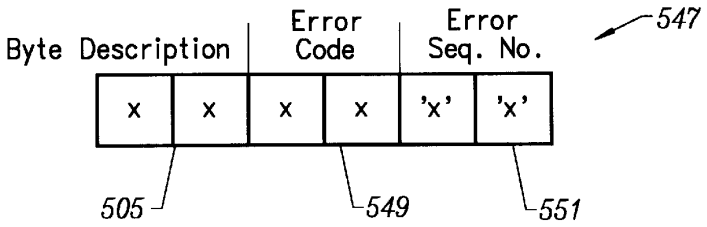


FIG. 5K

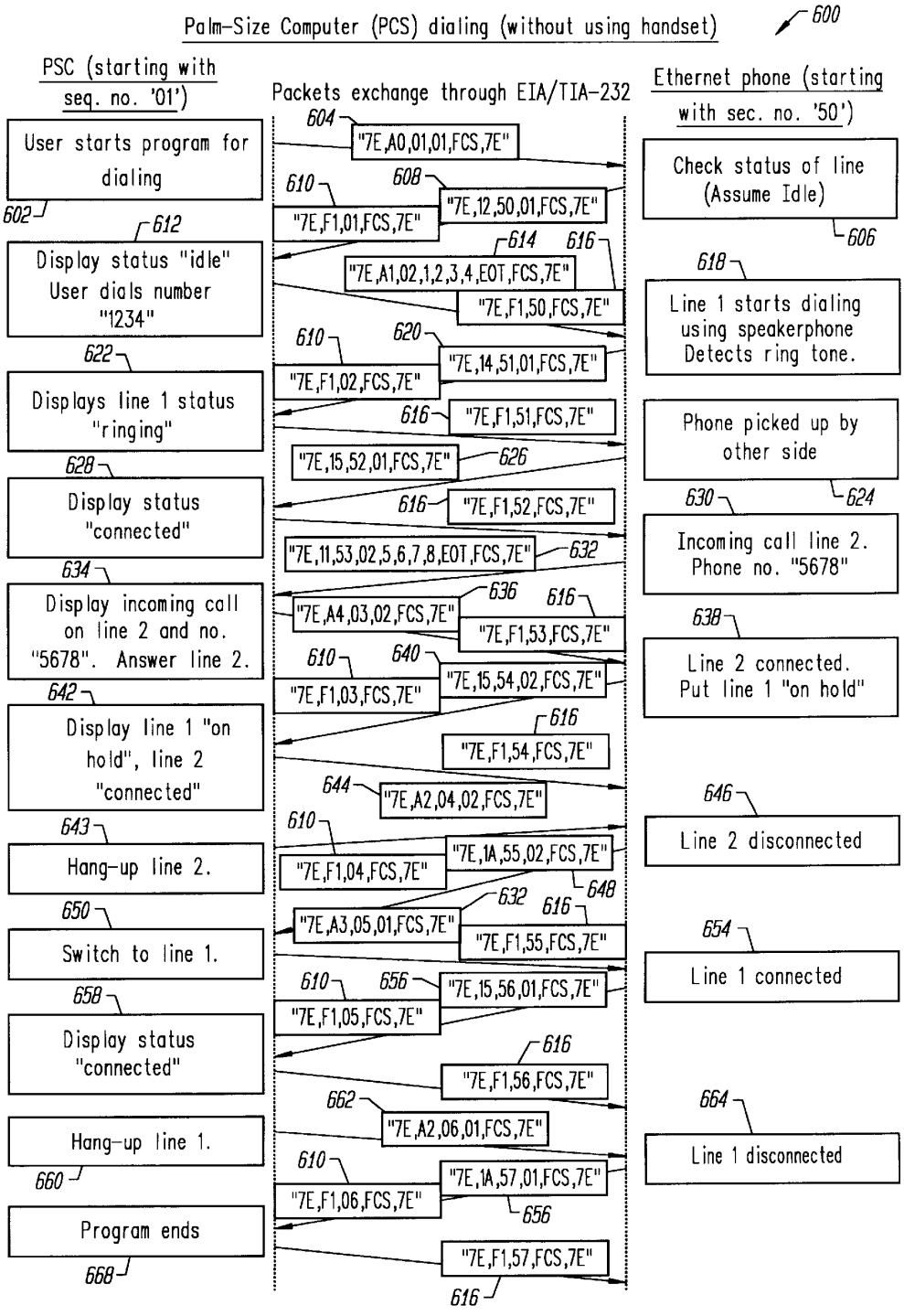

Palm-Size Computer (PCS) dialing (without using handset)
600
PSC (starting with seq. no. '01')
Packets exchange through EIA/TIA-232
Ethernet phone (starting with sec. no. '50')
User starts program for dialing
602
"7E,A0,01,01,FCS,7E"
604
Check status of line (Assume Idle)
606
"7E,12,50,01,FCS,7E"
608
"7E,F1,01,FCS,7E"
610
Display status "idle" User dials number "1234"
612
"7E,A1,02,1,2,3,4,EOT,FCS,7E"
614
"7E,F1,50,FCS,7E"
616
Line 1 starts dialing using speakerphone Detects ring tone.
618
"7E,F1,02,FCS,7E"
610
"7E,14,51,01,FCS,7E"
620
Displays line 1 status "ringing"
622
"7E,F1,51,FCS,7E"
616
Phone picked up by other side
624
"7E,15,52,01,FCS,7E"
626
Display status "connected"
628
"7E,F1,52,FCS,7E"
616
Incoming call line 2. Phone no. "5678"
630
"7E,11,53,02,5,6,7,8,EOT,FCS,7E"
632
Display incoming call on line 2 and no. "5678". Answer line 2.
634
"7E,A4,03,02,FCS,7E"
636
"7E,F1,53,FCS,7E"
616
Line 2 connected. Put line 1 "on hold"
638
"7E,F1,03,FCS,7E"
610
"7E,15,54,02,FCS,7E"
640
Display line 1 "on hold", line 2 "connected"
642
"7E,F1,54,FCS,7E"
616
"7E,A2,04,02,FCS,7E"
644
Line 2 disconnected
646
"7E,F1,04,FCS,7E"
610
"7E,1A,55,02,FCS,7E"
648
Hang-up line 2.
643
"7E,A3,05,01,FCS,7E"
632
"7E,F1,55,FCS,7E"
616
Line 1 connected
654
Switch to line 1.
650
"7E,F1,05,FCS,7E"
610
"7E,15,56,01,FCS,7E"
656
Display status "connected"
658
"7E,F1,56,FCS,7E"
616
"7E,A2,06,01,FCS,7E"
662
Line 1 disconnected
664
Hang-up line 1.
660
"7E,F1,06,FCS,7E"
610
"7E,1A,57,01,FCS,7E"
656
Program ends
668
"7E,F1,57,FCS,7E"
616

FIG. 6

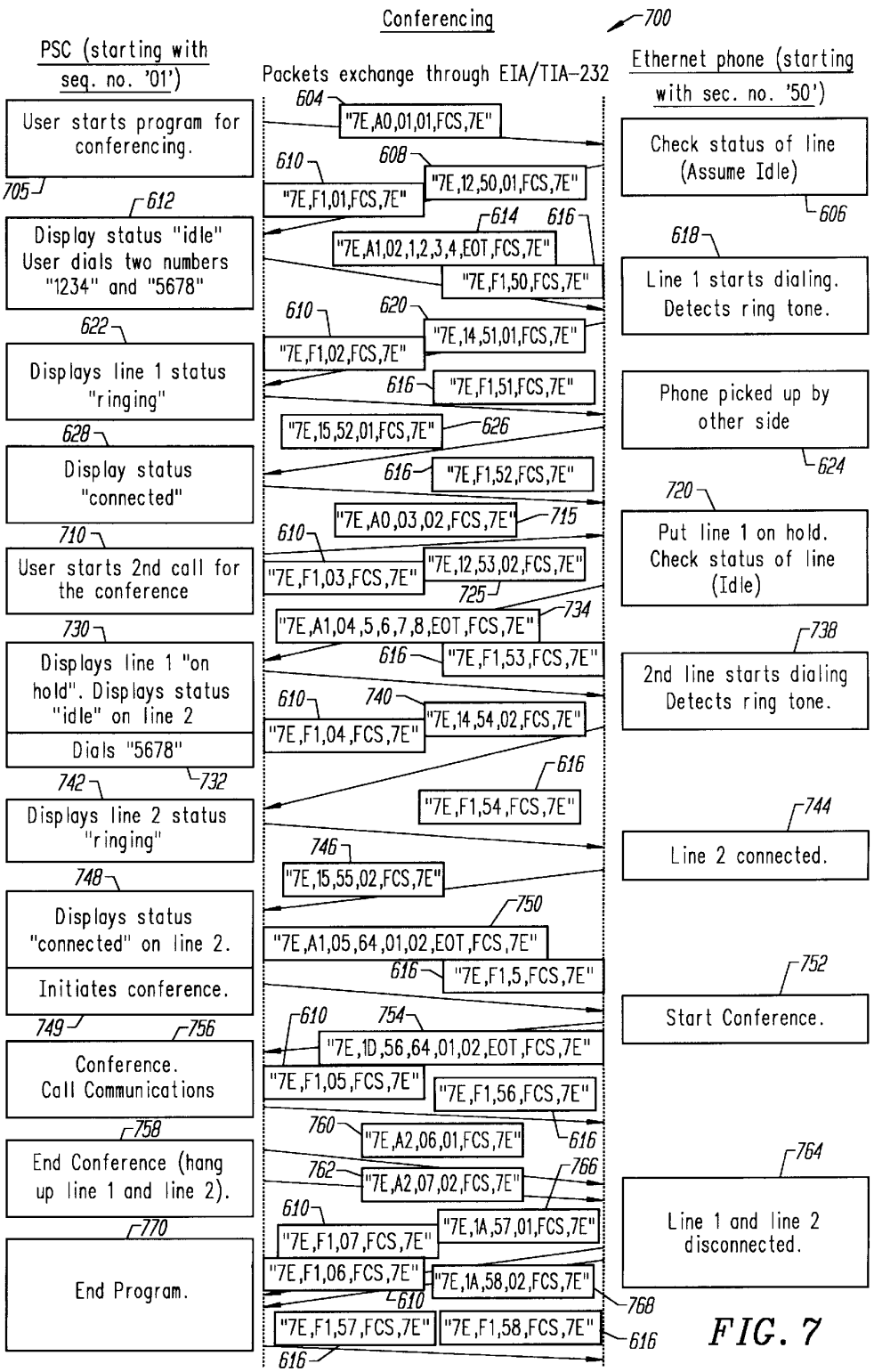

Conferencing
700
PSC (starting with seq. no. '01')
Packets exchange through EIA/TIA-232
Ethernet phone (starting with sec. no. '50')
User starts program for conferencing.
705
"7E,A0,01,01,FCS,7E"
604
Check status of line (Assume Idle)
606
"7E,12,50,01,FCS,7E"
608
"7E,F1,01,FCS,7E"
610
612
Display status "idle" User dials two numbers "1234" and "5678"
"7E,A1,02,1,2,3,4,EOT,FCS,7E"
614
"7E,F1,50,FCS,7E"
616
618
Line 1 starts dialing. Detects ring tone.
"7E,14,51,01,FCS,7E"
620
"7E,F1,02,FCS,7E"
610
622
Displays line 1 status "ringing"
"7E,F1,51,FCS,7E"
616
Phone picked up by other side
624
"7E,15,52,01,FCS,7E"
626
628
Display status "connected"
"7E,F1,52,FCS,7E"
616
720
Put line 1 on hold. Check status of line (Idle)
"7E,A0,03,02,FCS,7E"
715
"7E,12,53,02,FCS,7E"
"7E,F1,03,FCS,7E"
610
710
User starts 2nd call for the conference
725
"7E,A1,04,5,6,7,8,EOT,FCS,7E"
734
"7E,F1,53,FCS,7E"
616
738
2nd line starts dialing Detects ring tone.
730
Displays line 1 "on hold". Displays status "idle" on line 2
Dials "5678"
732
"7E,14,54,02,FCS,7E"
740
"7E,F1,04,FCS,7E"
610
"7E,F1,54,FCS,7E"
616
742
Displays line 2 status "ringing"
744
Line 2 connected.
"7E,15,55,02,FCS,7E"
746
748
Displays status "connected" on line 2.
Initiates conference.
"7E,A1,05,64,01,02,EOT,FCS,7E"
750
"7E,F1,5,FCS,7E"
616
752
Start Conference.
749
Conference. Call Communications
756
"7E,1D,56,64,01,02,EOT,FCS,7E"
754
610
"7E,F1,05,FCS,7E"
"7E,F1,56,FCS,7E"
616
"7E,A2,06,01,FCS,7E"
760
758
End Conference (hang up line 1 and line 2).
"7E,A2,07,02,FCS,7E"
762
766
764
Line 1 and line 2 disconnected.
"7E,1A,57,01,FCS,7E"
610
"7E,F1,07,FCS,7E"
"7E,F1,06,FCS,7E"
"7E,1A,58,02,FCS,7E"
770
End Program.
610
768
"7E,F1,57,FCS,7E"
"7E,F1,58,FCS,7E"
616
616

FIG. 7

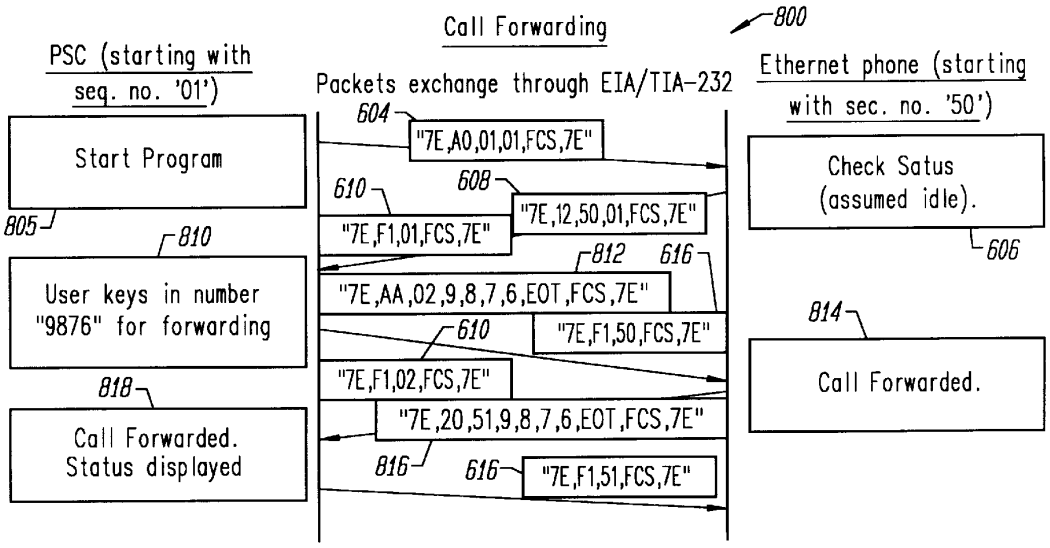

Call Forwarding
800
PSC (starting with seq. no. '01')
Packets exchange through EIA/TIA-232
Ethernet phone (starting with sec. no. '50')
Start Program
805
604
"7E,A0,01,01,FCS,7E"
608
"7E,12,50,01,FCS,7E"
610
"7E,F1,01,FCS,7E"
Check Satus (assumed idle).
606
810
User keys in number "9876" for forwarding
812
616
"7E,AA,02,9,8,7,6,EOT,FCS,7E"
610
"7E,F1,50,FCS,7E"
814
Call Forwarded.
"7E,F1,02,FCS,7E"
818
Call Forwarded. Status displayed
"7E,20,51,9,8,7,6,EOT,FCS,7E"
816
616
"7E,F1,51,FCS,7E"

FIG. 8

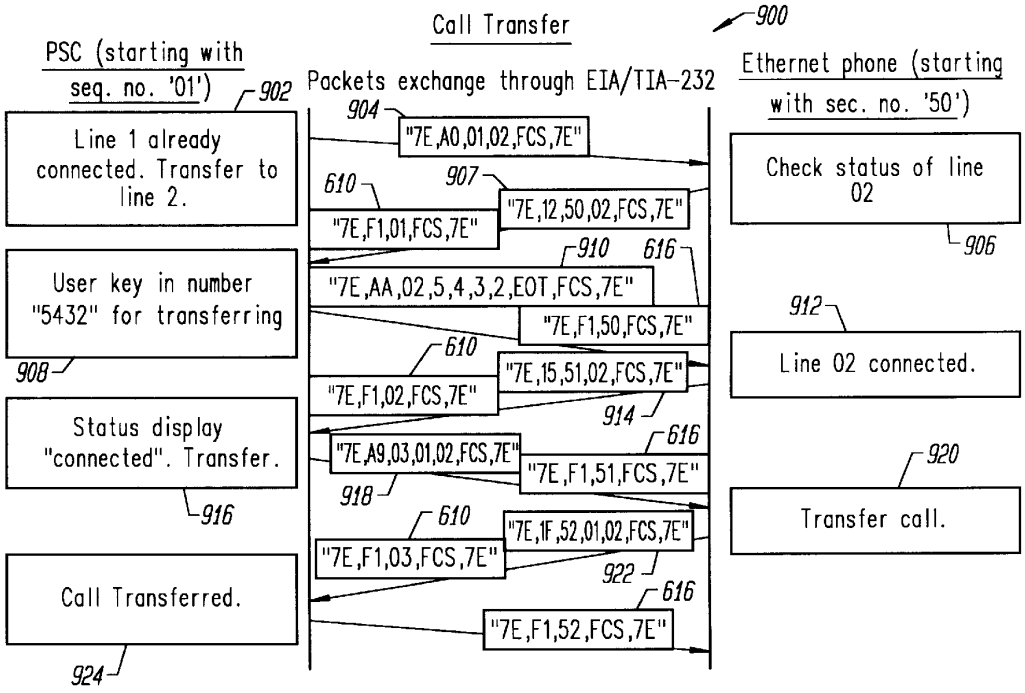

Call Transfer
900
PSC (starting with seq. no. '01')
902
Packets exchange through EIA/TIA-232
Ethernet phone (starting with sec. no. '50')
Line 1 already connected. Transfer to line 2.
904
"7E,A0,01,02,FCS,7E"
Check status of line 02
906
907
"7E,12,50,02,FCS,7E"
610
"7E,F1,01,FCS,7E"
910
616
"7E,AA,02,5,4,3,2,EOT,FCS,7E"
User key in number "5432" for transferring
908
"7E,F1,50,FCS,7E"
912
610
"7E,15,51,02,FCS,7E"
"7E,F1,02,FCS,7E"
Line 02 connected.
914
Status display "connected". Transfer.
616
"7E,A9,03,01,02,FCS,7E"
"7E,F1,51,FCS,7E"
920
916
918
610
"7E,1F,52,01,02,FCS,7E"
Transfer call.
"7E,F1,03,FCS,7E"
922
Call Transferred.
616
"7E,F1,52,FCS,7E"
924

FIG. 9

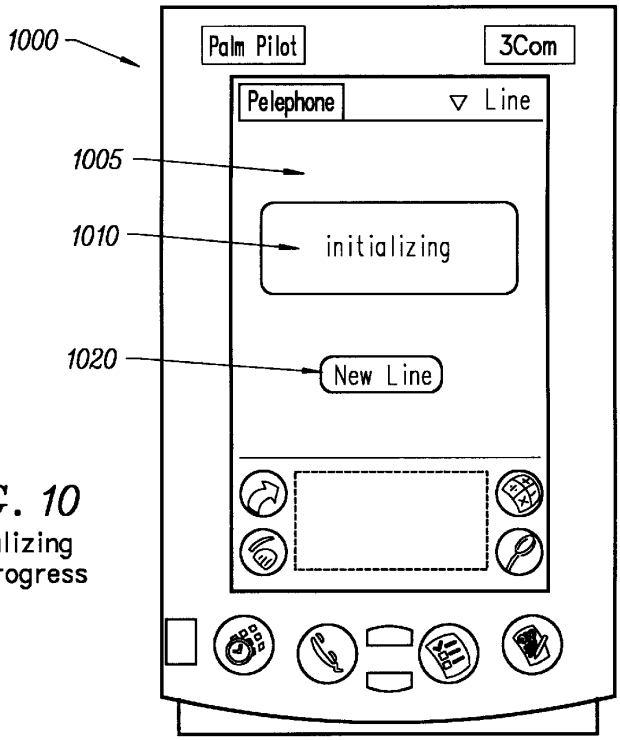


FIG. 10
Initializing in Progress

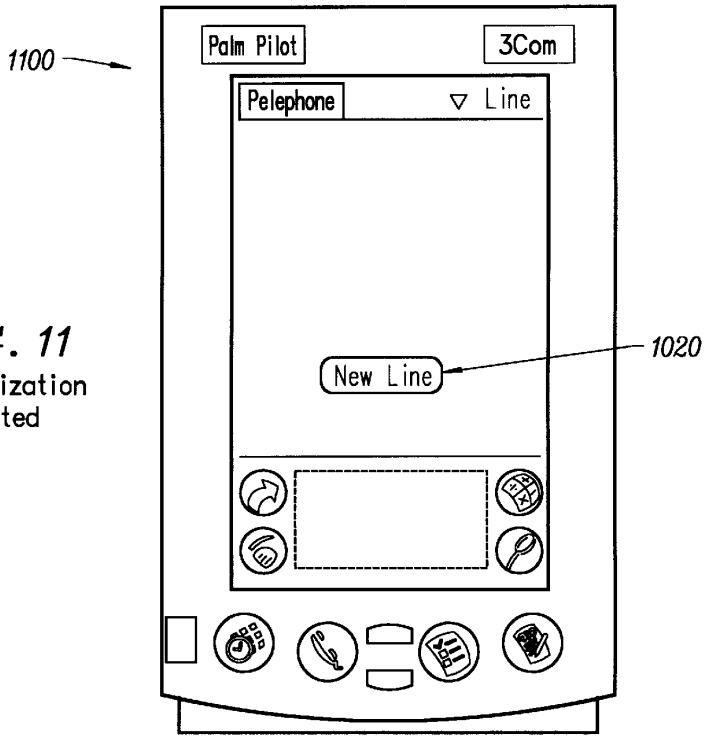


FIG. 11
Initialization Completed

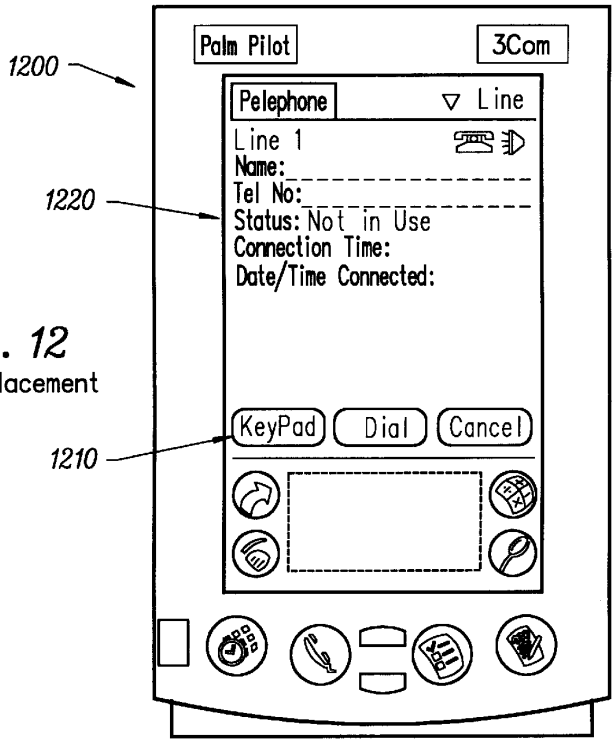


FIG. 12
Call Placement

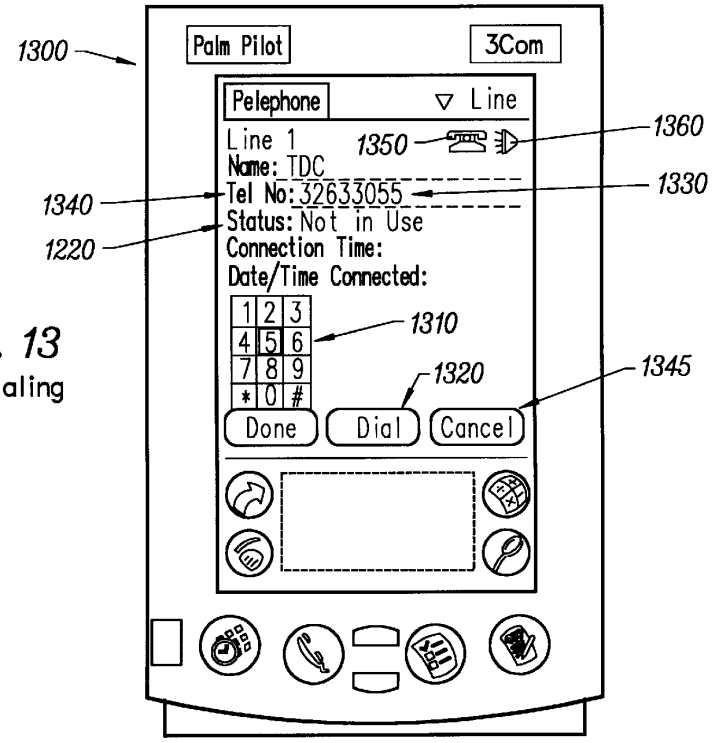


FIG. 13
User Dialing

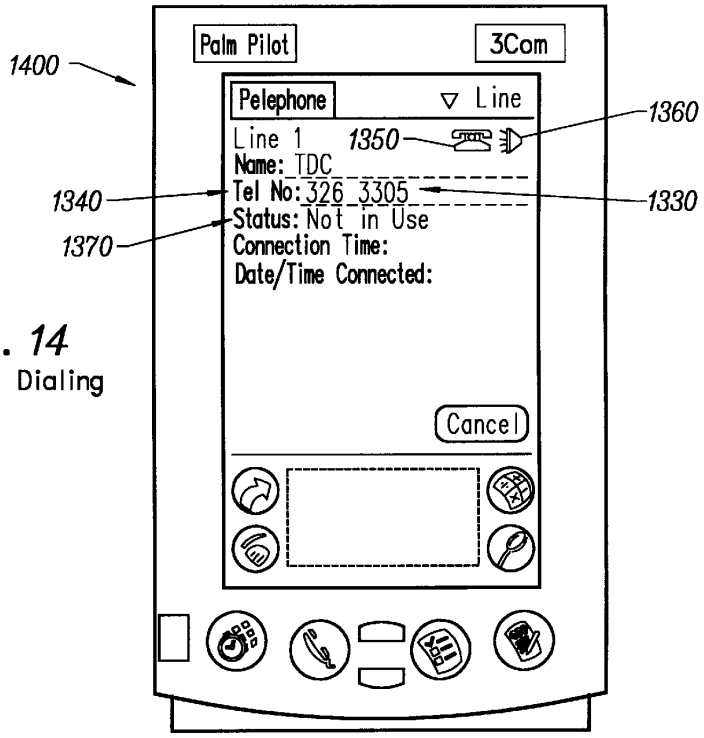


FIG. 14
Telephone Dialing

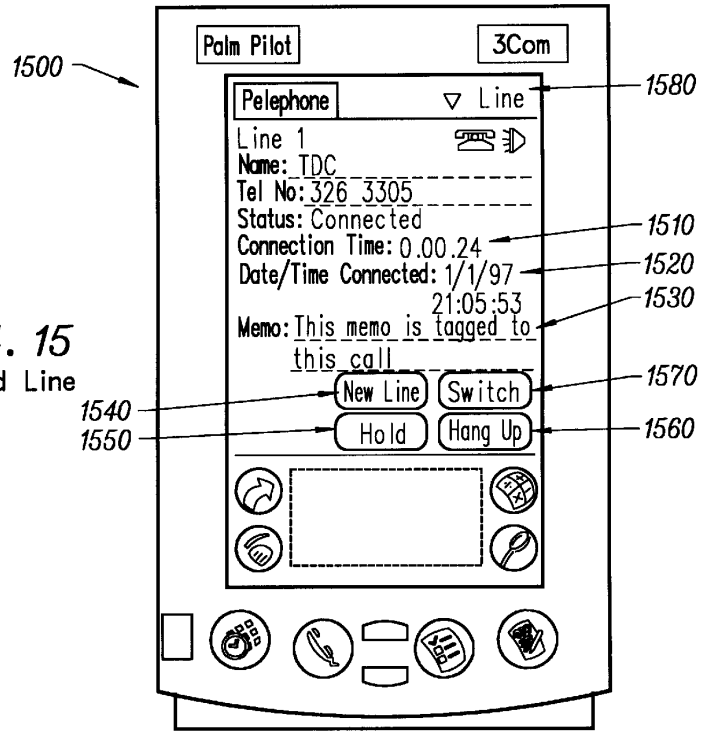


FIG. 15
Connected Line

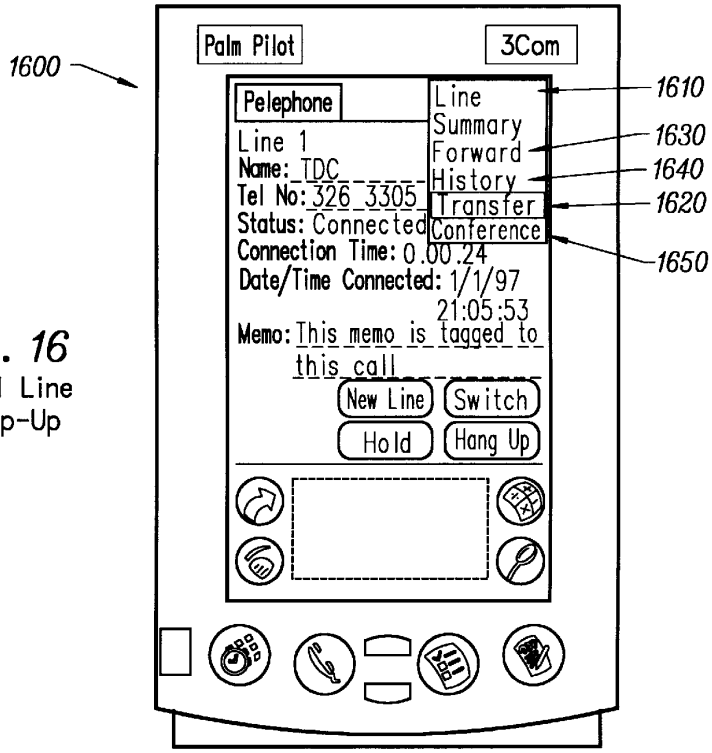


FIG. 16
Connected Line with Pop-Up Menu

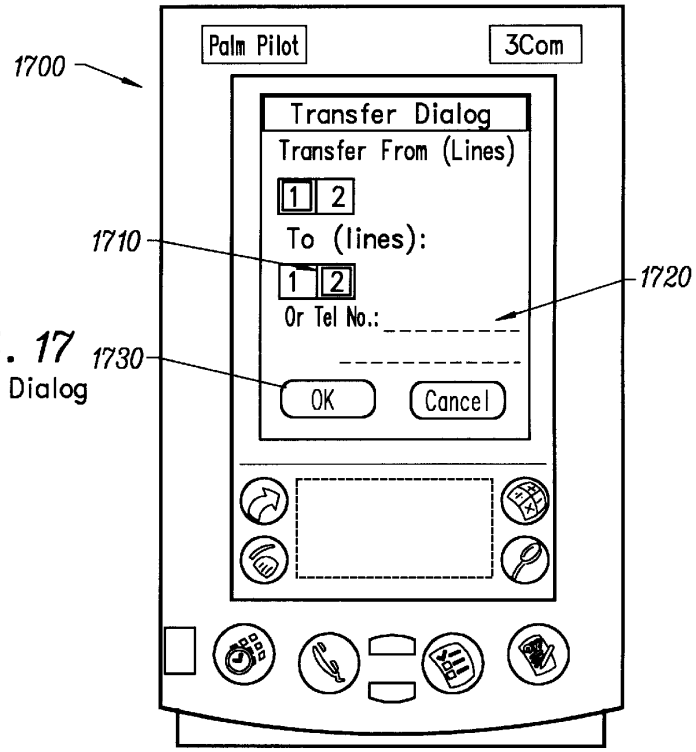


FIG. 17
Transfer Dialog

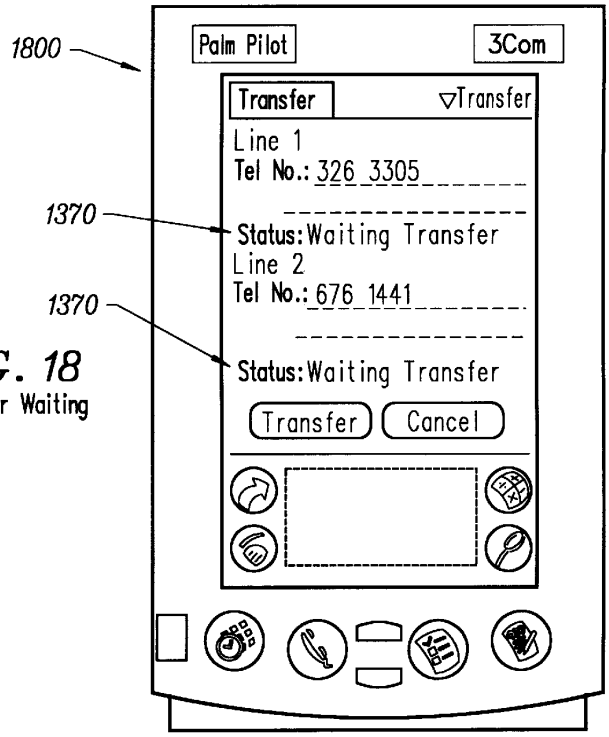


FIG. 18
Transfer Waiting

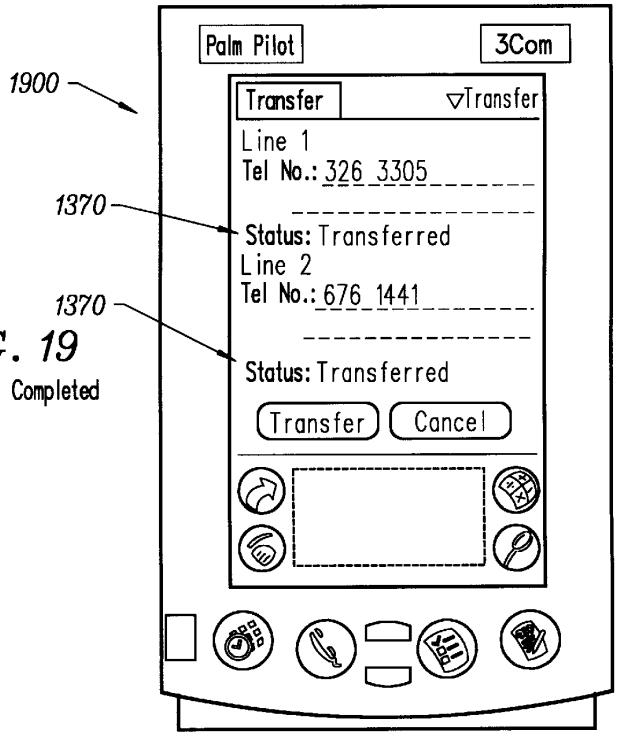


FIG. 19
Transfer Completed

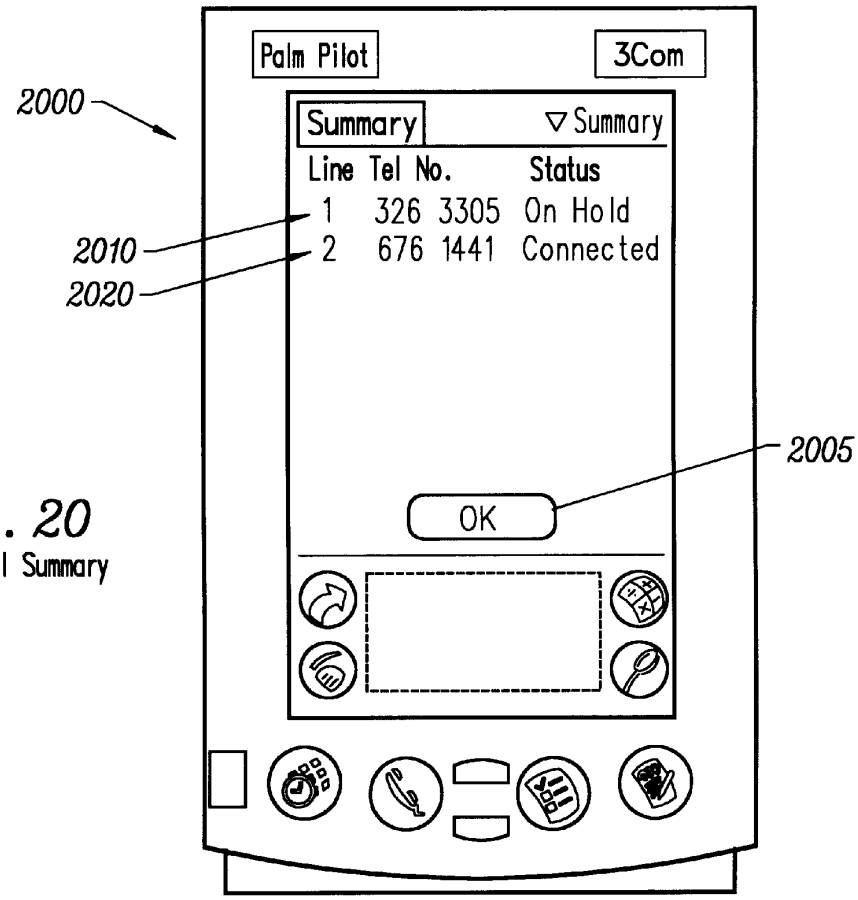


FIG. 20
Active Call Summary

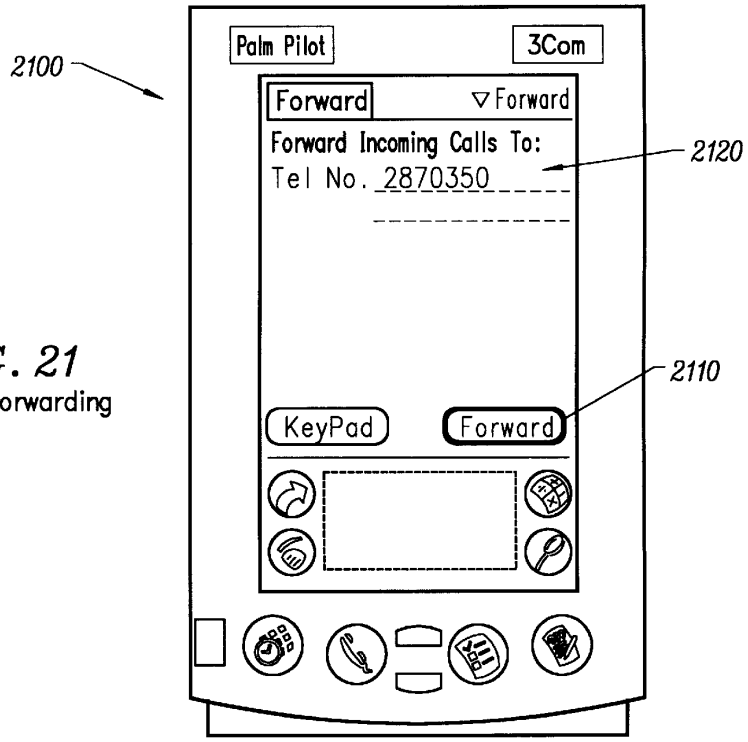


FIG. 21
First Forwarding

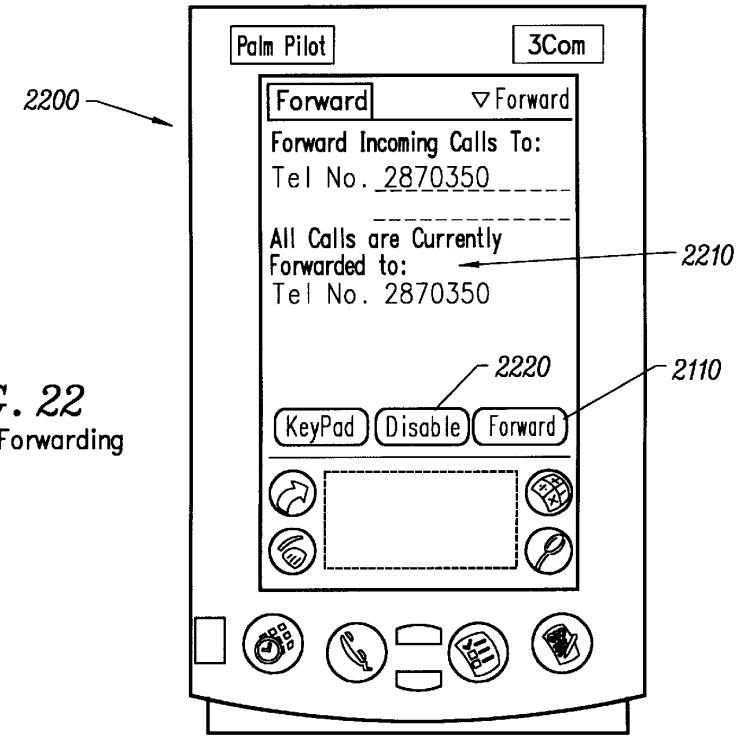


FIG. 22
Second Forwarding

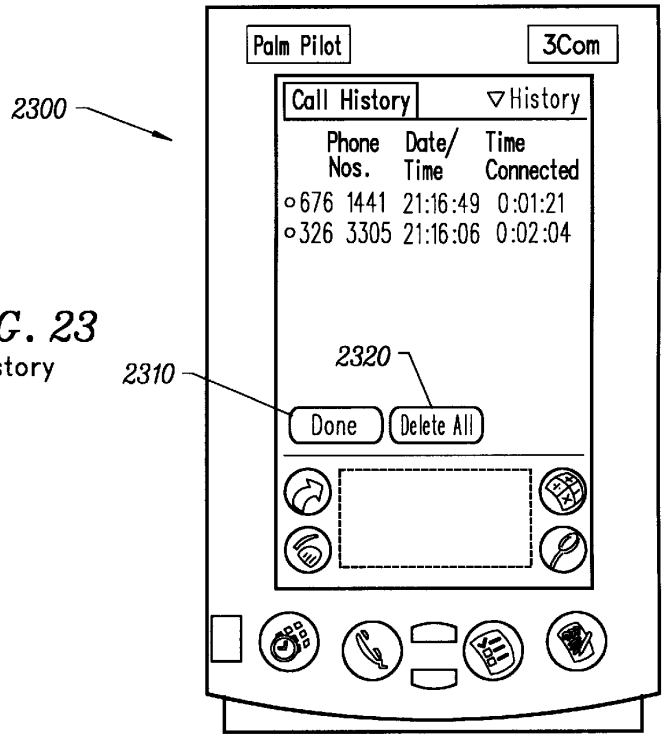


FIG. 23
History

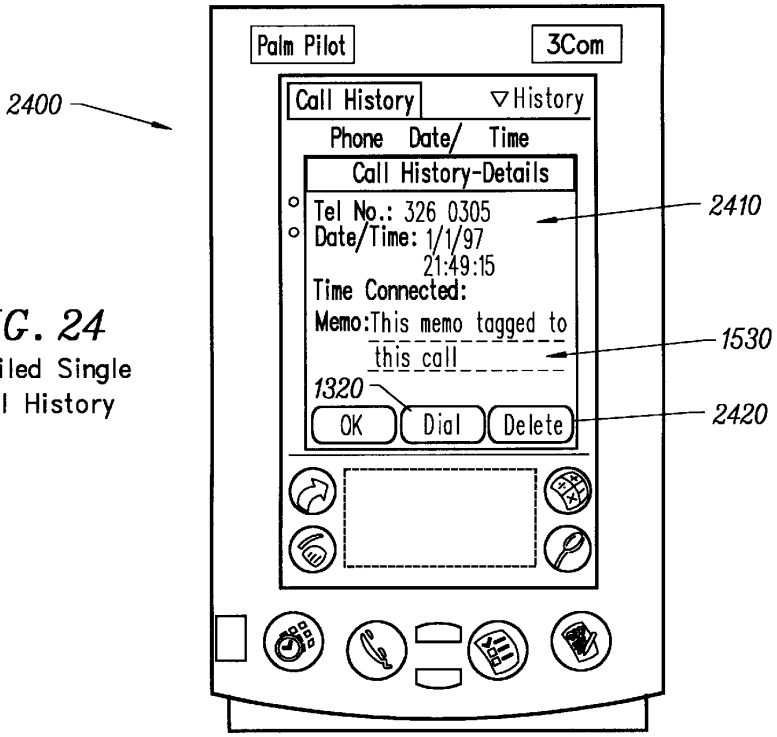


FIG. 24
Detailed Single Call History

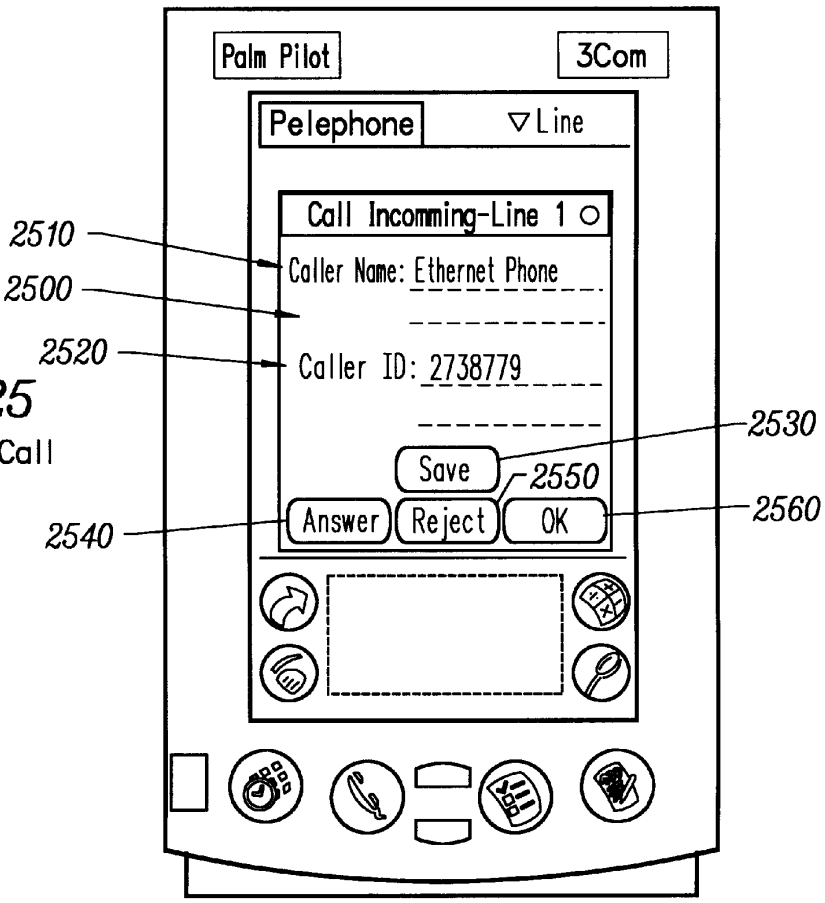


FIG. 25
Incomming Call

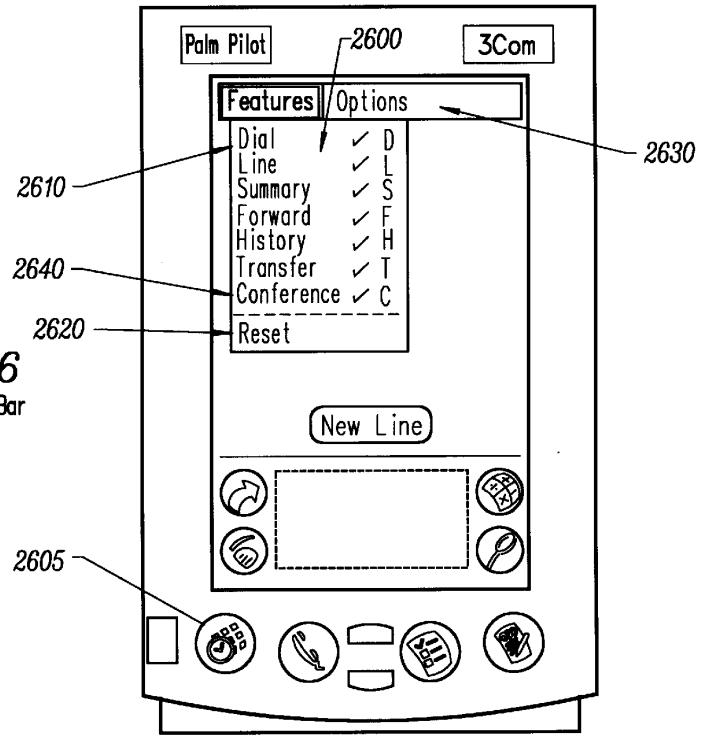


FIG. 26
Features Menu Bar

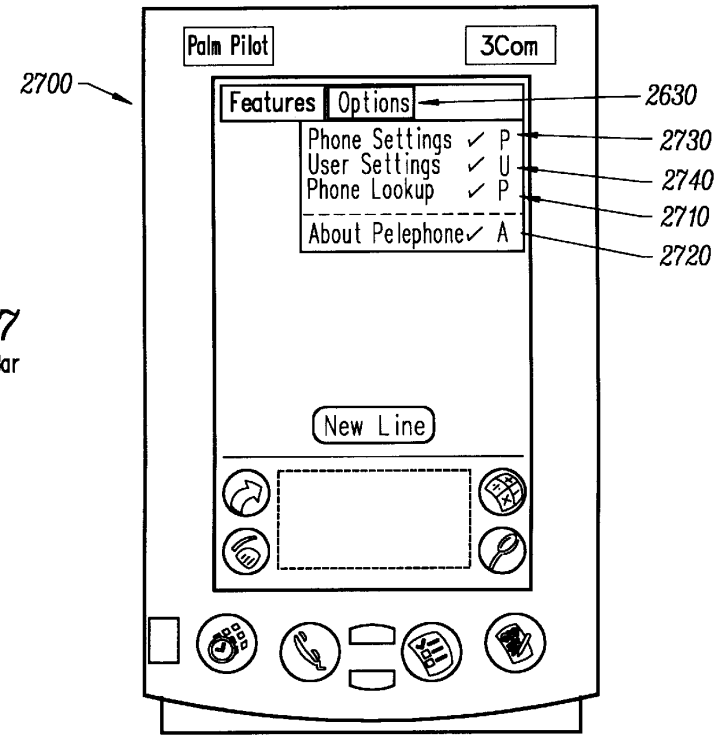


FIG. 27
Options Menu Bar

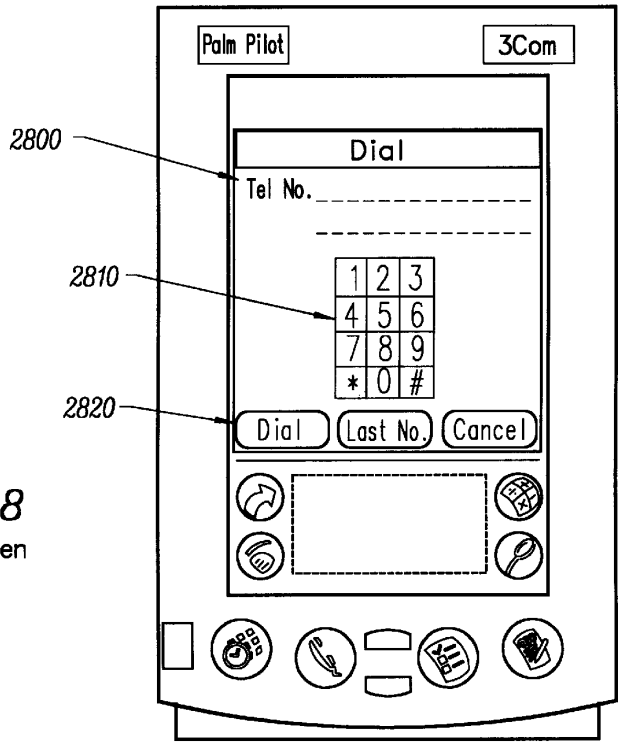


FIG. 28
Dial Screen

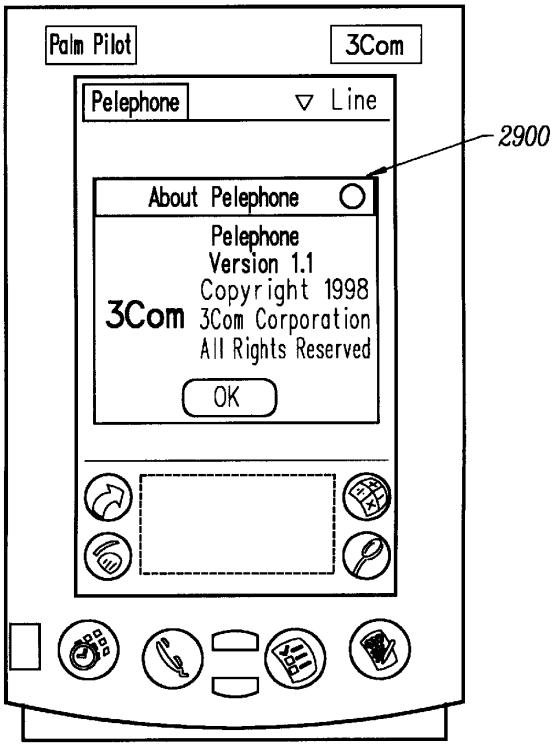


FIG. 29
Copyright Information

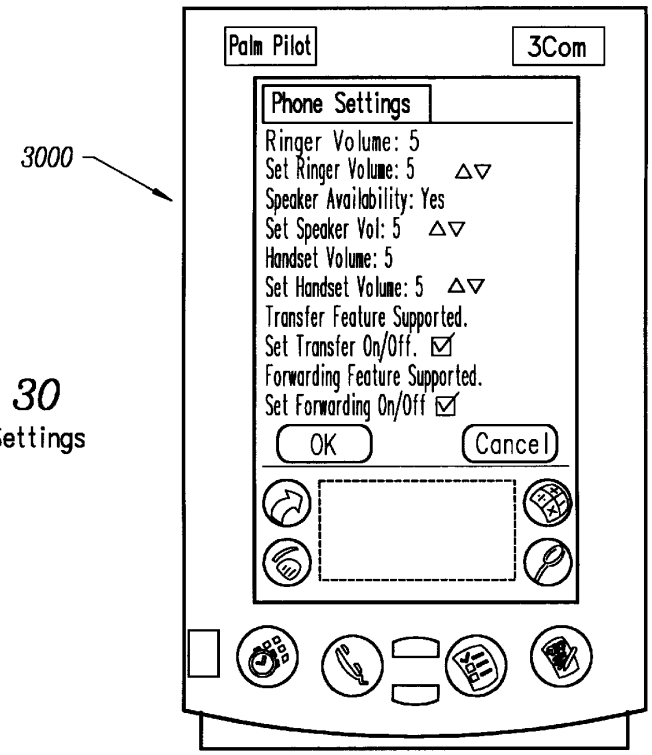


FIG. 30
Phone Settings

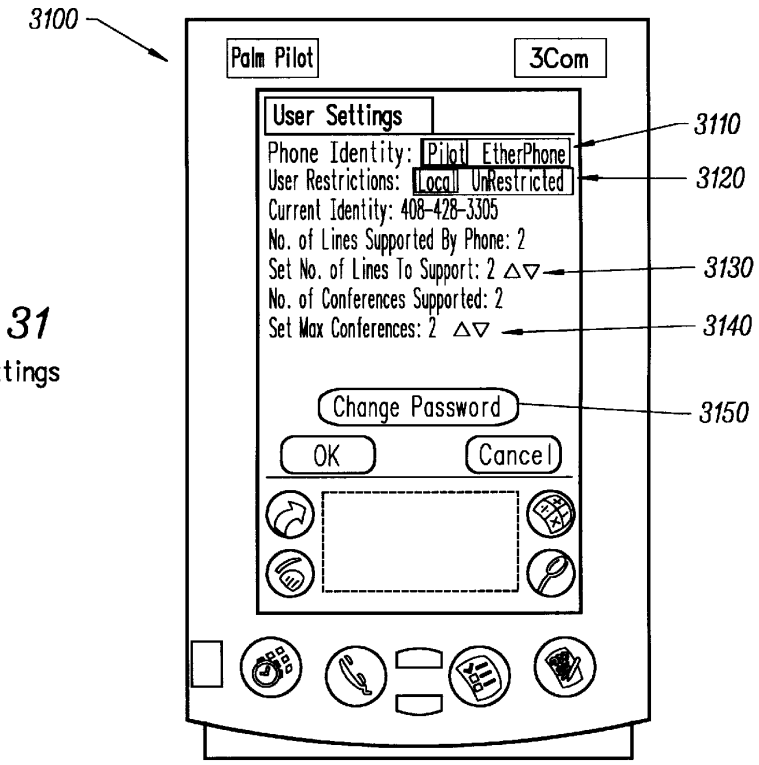


FIG. 31
User Settings

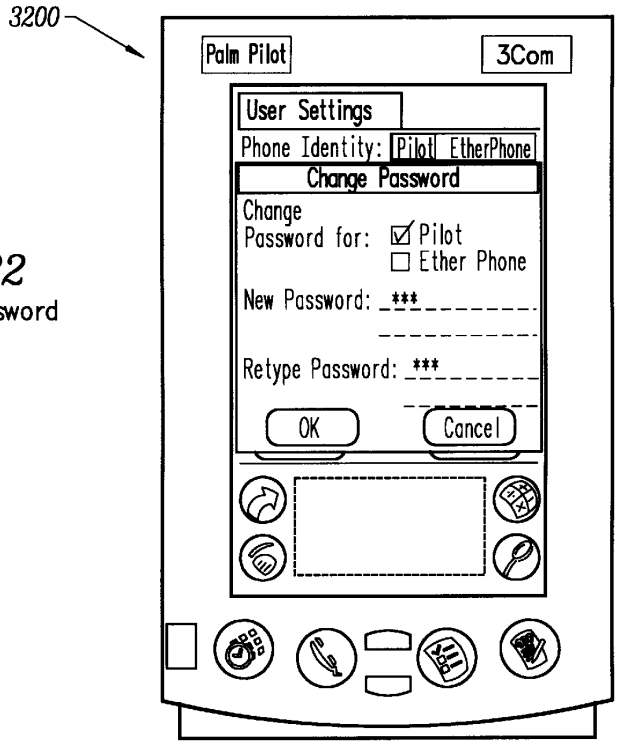


FIG. 32
Change Password

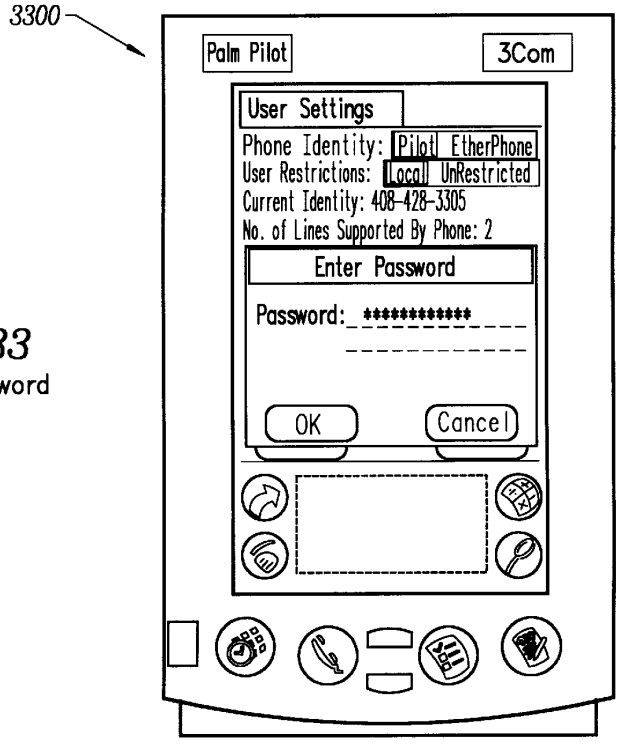


FIG. 33
Enter Password

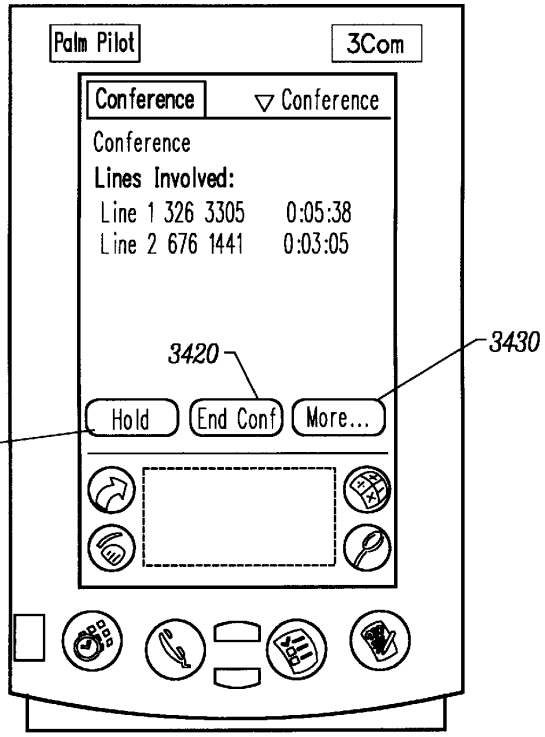


FIG. 34
Conferencing Screen

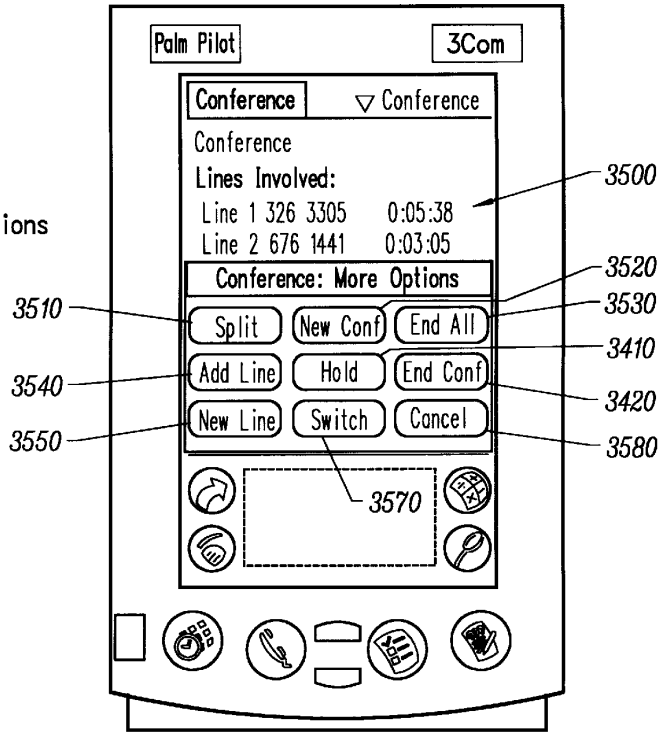


FIG. 35
Conference Options

METHOD, APPARATUS AND COMMUNICATIONS SYSTEM FOR COMPANION INFORMATION AND NETWORK APPLIANCES

BACKGROUND OF THE INVENTION

1. Field of Invention

The invention generally relates to network communications devices. More specifically, the invention relates to the combination of a portable computer with a communications device to form a more capable compound network apparatus.

2. Description of Related Art

Recent advances in the manufacture and design of integrated circuits have enabled technology producers to provide portable instruments with ever-increasing processing capabilities. Advances in liquid crystal diode displays, stylus based input devices, and handwritten character recognition have also resulted in the availability of palm-sized computers [or personal digital assistants (PDAs)], such as the Palm III and PalmProfessional from Palm Computing, Inc., Mountain View, Calif. The primary advantage of these devices is the combination of small size, lightweight, and stored information that can be customized for a particular user. These computers provide functions such as note taking, data retrieval and storage, application program execution, and interfacing with external devices. The palm-sized computers have been very successful in calendar and telephone directory utilities, and also enable users to have access to e-mail, and to even play games.

The prior art includes accessories that allow a portable computer to become part of a telecommunications device. One such accessory is described in U.S. Pat. No. 5,606,594, granted to Register et al. on Feb. 25, 1997, entitled "Communication Accessory and Method of Telecommunicating for a PDA". A top-edge view of the PDA **100** adapted for insertion into the accessory is provided in FIG. **1**A. The PDA **100** has one or more buttons **110** and an electronic link connector **120**. As shown in FIG. **1**B, the accessory **150** is specially adapted to receive a particular PDA. The PDA **100** can be releasably inserted into the accessory **150**. The PDA **100** electronic link connector **120** mates with the accessory **150** electronic link connector **160**. The external surfaces of the PDA **100** fit within the retaining wall **170**, and concave surface **180** of the accessory **150**. The retention ridges **190** are disposed to cover the top and bottom portion of the exterior of the PDA **100** by rotating the rotatable body extensions **195** of the accessory **150**.

The accessory **150** allows the PDA to play a part in managing voice communications for the user, and to send and receive data. Similarly, U.S. Pat. No. 5,497,339, granted to Bernard on Mar. 5, 1996 provides for PDA that mounts within a communications device. However, none of the communications devices in the prior art provide methods for software implementation of telephone call processing functions. Also missing from the prior art is the method for storing the phone number and user parameters in the PDA and then deploying the communications device with the PDA/user phone number and user characteristics.

Accordingly, what is needed in the art is a method and apparatus for companion information and network appliances that incorporates software implementation of telephone call processing functions, and storage of phone number and user parameters in the PDA for network appliance deployment.

SUMMARY OF THE INVENTION

The invention provides an information appliance and a network appliance that function independently as well as

with each other. The information appliance stores information corresponding to a particular user. The network appliance is linked to a local area network and is capable of simultaneously exchanging voice and data messages with devices connected to the local area network.

The appliances are connected to each other physically through a communications port, and exchange specially formatted data corresponding to user personalized information, commands from the user, and responses including message status information from the network connected devices. The user personalized information enables the network appliance to perform network communications according to user specified settings and enables the network appliance to assume the user specific information appliance identification. The information appliance is typically a portable computer and in some embodiments is a palm-sized computer. The network appliance is typically a network attached phone and in some embodiments is an Ethernet telephone.

A first aspect of the invention provides a communications system comprising a telephone having capabilities, a portable computer connected to the telephone, and a network link connecting the telephone to network connected devices. The portable computer includes a port for connecting to the telephone, a memory storing user information corresponding to a user; and processing resources adapted to exchange data with the telephone. The data includes the user information and data corresponding to the telephone capabilities. The exchange of the data enables the portable computer to: discover capabilities of the telephone; provide the user information to the telephone; and establish telephone operating parameters for telephone communications with devices connected to the telephone based on the user information and the telephone capabilities.

In some embodiments, the communications system includes a gateway server connected to the network link, and switched circuit network devices connected to the gateway server.

In some embodiments, the communications system includes a router connected to the network link, and packet based network devices connected to the router. For some of these embodiments, the portable computer includes processing resources for Internet access, and the telephone includes processing resources for Internet access.

For some of the embodiments with Internet access capability, the portable computer includes processing resources for Internet access including Internet applications, transmission control software, and Internet protocol software. The telephone includes processing resources for Internet access including an Internet access application, transmission control software, Carrier Sense Multiple Access/Collision Detection software, and Internet protocol software.

For some of the embodiments with Internet access capability, the portable computer includes processing resources for Internet access including Internet display applications and display/user input transfer software. The telephone includes processing resources for Internet access including Internet applications, display/user input transfer software, transmission control software, Internet protocol software, and Carrier Sense Multiple Access/Collision Detection software.

In some embodiments, the portable computer comprises a palm-sized computer; and the telephone comprises an Ethernet telephone. In some embodiments, the data exchanged with the telephone corresponding to the telephone capabili-

ties and the user information are formatted according to an applications layer protocol. The applications layer protocol has frame formats for telephony functions.

In some embodiments, the portable computer includes processing resources for user interface support of video data. The telephone includes a display, and processing resources for video display and capture. In some embodiments, the portable computer includes processing resources for user interface support of video data, video data decoding, and video display. In some embodiments, the portable computer includes processing resources for user interface support of video data, video data decoding, video display, and video camera image data.

A second aspect of the invention provides a method for transmitting data from a portable computer to a telephone. The telephone has operating capabilities. The telephone is connected to network connected devices. The method comprises connecting the portable computer with the telephone, supplying the portable computer with telephone operating data, the portable computer exchanging the telephone operating parameter data and the operating capabilities with the telephone, and the portable computer establishing telephone operating parameters for a communications sessions. In some embodiments, the method includes the user starting a telephony program with the portable computer controlling execution of the telephony program.

The telephone operating parameter data is for a communications session including an exchange of messages with one or more of the network connected devices. In some embodiments, the communications session includes simultaneous exchanges of voice and packet data messages.

The telephone operating parameters for the communications session are based on the telephone operating parameter data and the operating capabilities. The telephone operating parameters provide options and features for the communications session. In some embodiments, the operating parameter data comprises constructs formatted according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions.

In some embodiments, prior to connecting the portable computer with the telephone, the portable computer stores user information. The user information can include an identification corresponding to the portable computer, user access parameters, and user characteristics corresponding to the telephone operating parameter data. For some of these embodiments, establishing telephone operating parameters includes the user selecting user setting inputs. The user setting inputs corresponding to the portable computer identification, user access parameters, and user characteristics, the user selecting change the corresponding telephone operating parameter data. In one embodiment, the user setting inputs include the network address of the telephone, so that the user can change the network address provided by the telephone to be the network address of the portable computer.

In some embodiments, after the establishing step, the method includes one or more of the following programs: conferencing, dialing, receiving an incoming call, forwarding, transferring, and placing a call. For some of these embodiments, the telephone receives input data from the portable computer. The input data is formatted according to a data link layer protocol. The data link layer protocol encapsulates frames formatted according to an application layer protocol adapted for telephony functions. The telephone transforms the input data into transport data formatted according to a transport protocol for a packet switched network. The telephone transmits the transport data to a gateway server. The gateway server is connected by a local area network link to at least one switched circuit network including a public switched telephone network.

In some embodiments, the network connected devices include a gateway server. The gateway server provides access for the user to a public switched telephone network. For some of these embodiments, the telephone is also connected to a gatekeeper and a directory server by a local area network link. Communications between the telephone and the gatekeeper, gateway server, and directory server are formatted according to a soft private branch exchange telephony application layer protocol.

In some embodiments, the telephone is connected by a local area network link to a router. The router is connected to at least one packet based network including an Internet source. Communications between the router and the telephone are formatted according to packet based network application protocols.

A third aspect of the invention includes a method for exchanging voice and data messages between a telephone and devices connected to a network. The telephone is connected to the network. The method starts by connecting the telephone with a portable computer. The portable computer then exchanges telephone operating parameter data with the telephone. The operating parameter data provides options for communications between the telephone and the network connected devices.

Then, in response to a user indication of a desired communication, the portable computer exchanges call data with the telephone. The call data corresponds to the desired communication. The call data is formatted according to an application layer protocol, and the underlying transport, network, and data link layer protocols. The application layer protocol has frame formats for telephony functions.

The telephone then exchanges messages with an addressed network connected device. The messages correspond to the desired communication. The addressed network connected device has a network address. The message includes data corresponding to the address of the addressed network connected device.

In some embodiments, the method includes, prior to connecting the portable computer with the telephone, the portable computer storing user information. The operating parameter data comprises the user information. The user information comprises an identification corresponding to the portable computer and user access parameters. For some of these embodiments, the telephone has an identification. The method includes the telephone presenting and the identification corresponding to the portable computer to the network connected devices in place of the telephone identification.

In some embodiments, the telephone comprises an Ethernet telephone, and the portable computer comprises a palm-sized computer.

In some embodiments, the telephone is connected to a gatekeeper, a directory server and a gateway server by a local area network link. Communications between the telephone and the gatekeeper, the gateway server, and the directory server are formatted according to a soft private branch exchange telephony application layer protocol.

In some embodiments, the telephone is connected by a local area network link to a router. The router is connected to at least one packet based network including an Internet source. Communications between the router and the telephone are formatted according to packet based network application protocols.

In some embodiments, the exchange of messages includes simultaneous exchanges of voice and packet data messages.

A fourth aspect of the invention provides a portable computer adapted for connection to a telephone. The telephone has capabilities. The portable computer comprises a port for connecting to the telephone, a memory storing user information corresponding to a user; and processing resources adapted to exchange data with the telephone. The data exchanged with the telephone includes the user information and data corresponding to the telephone capabilities. The exchange of the data enables the portable computer to discover capabilities of the telephone, provide the user information to the telephone; and establish telephone operating parameters for telephone communications with devices connected to the telephone based on the user information and the telephone capabilities.

In some embodiments, the portable computer includes a display providing user interface graphic elements corresponding to data exchanged with the telephone, and a user interface enabling the user to input data supplementing the user information provided to the telephone. For some of these embodiments, the data exchanged with the telephone includes data corresponding to portable computer control of telephony programs, and data corresponding to the status of the devices connected to the telephone.

In some embodiments, the user information comprises an identification corresponding to the portable computer, user characteristics, and user access parameters. For some of these embodiments, the telephone has an identification, and the identification corresponding to the portable computer is presented by the telephone instead of the telephone identification to devices connected to and communicating with the telephone.

In some embodiments, the data exchanged with the telephone corresponding to portable computer control of the execution of the telephony programs, the telephone capabilities, and the user information are formatted according to an applications layer protocol. The applications layer protocol has frame formats for telephony functions.

In some embodiments, the portable computer comprises a palm-sized computer. In some embodiments, the portable computer is adapted to provide data processing and user interface functions without connection to the telephone. In some embodiments, the exchange of the data enables the portable computer to control execution of telephony programs.

In some embodiments, the portable computer includes processing resources for Internet access. The processing resources for Internet access can include Internet applications, transmission control software, and Internet protocol software. Alternatively, the processing resources for Internet access can include Internet display applications and display/user input transfer software.

In some embodiments, the portable computer includes processing resources for user interface support of video data. The processing resources for user interface support of video data can include video data decoding, and video display. Alternatively, the processing resources for user interface support of video data, video data decoding, video display, and video camera image data.

A fifth aspect of the invention provides a telephone adapted for connection to a portable computer. The portable computer has user information corresponding to a user. The telephone comprises a port for connecting to the portable computer, network communication capabilities, portable computer companion capabilities, and processing resources adapted to exchange data with the portable computer. The network communication capabilities include a communication port. The data includes the user information, data corresponding to the network communication capabilities; and the portable computer companion capabilities. The exchange of data enables the telephone to: discover user information and capabilities of the portable computer, provide the network communication capabilities and the portable computer companion capabilities to the portable computer, and indicate the network communication capabilities to devices connected to the telephone via a network.

In some embodiments, the portable computer is adapted to control execution of telephony programs. The data exchanged with the portable computer includes data corresponding to portable computer control of the telephony programs. Responsive to commands from the portable computer, the exchange of the data enables the telephone to communicate with devices connected to the telephone.

In some embodiments, the telephone has an identification and the portable computer has an identification. The portable computer identification is presented by the telephone in place of the telephone identification to devices connected to and communicating with the telephone.

In some embodiments, the telephone comprises an Ethernet telephone.

In some embodiments, the data exchanged with the portable computer corresponding to the telephone capabilities and the user information are formatted according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions.

In some embodiments, the telephone includes processing resources adapted to receive incoming call data from a gateway server indicating that a first network connected device is waiting to start a call with a user, and processing resources adapted to transform the incoming call data into an incoming call message formatted in a data link layer protocol for transmission to the portable computer. The data link layer protocol encapsulates frames formatted according to the application layer protocol.

In some embodiments, the telephone provides the network communication capabilities to a user without connection to the portable computer.

In some embodiments, the telephone includes processing resources for Internet access. The processing resources for Internet access can include Internet applications, display/user input transfer software, transmission control software, Internet protocol software, and Carrier Sense Multiple Access/Collision Detection software. Alternatively, the processing resources for Internet access can include an Internet access application, transmission control software, Carrier Sense Multiple Access/Collision Detection software, and Internet protocol software.

In some embodiments, the telephone includes a display; and processing resources for video display and capture.

BRIEF DESCRIPTION OF THE FIGURES

FIG. **1A** is a top-edge view of a portable computer adapted for insertion into a prior art telephone accessory.

FIG. **1B** is a perspective view of a prior art telephone accessory adapted to receive a portable computer.

FIG. **2** is a schematic diagram illustrating the connections between the telephone, the portable computer, the local area network, and services connected to the local area network.

FIG. **3A** is communications protocol hierarchy for an embodiment of the invention with Application Program

Interface (API), Point to Point protocol (PPP), and High-level Data Link Control (HDLC) layers for telephony applications.

FIG. **3**B is a communications protocol hierarchy for an embodiment of the invention with Ethernet Telephone Management Protocol (EMP) and data link layer protocol (DLLP) layers for telephony applications.

FIG. **3**C is a protocol stack for the companion appliances for an embodiment where the palm-sized computer runs the Internet applications protocol stack.

FIG. **3**D is a protocol stack for the companion appliances for an embodiment where the Ethernet telephone is running the Internet applications protocol stack.

FIG. **4** illustrates a frame format according to one embodiment of the invention, referred to as the Data Link Layer Protocol.

FIGS. **5**A, **5**B, **5**C, **5**D, **5**E, **5**F, **5**G, **5**H, **5**I, **5**J, and through **5**K illustrate a plurality of Ethernet Telephone Management Protocol (EMP) frame body formats according to one embodiment of the invention.

FIG. **6** provides a state diagram for a palm-sized computer dialing a phone number according to one embodiment of the invention.

FIG. **7** provides a state diagram for conferencing according to one embodiment of the invention.

FIG. **8** provides a state diagram for call forwarding according to one embodiment of the invention.

FIG. **9** provides a state diagram for call transfer according to one embodiment of the invention.

FIG. **10** shows an initialization in progress screen for a palm-sized computer according to one embodiment of the invention.

FIG. **11** shows an initialization completed screen for a palm-sized computer according to one embodiment of the invention.

FIG. **12** shows a call placement screen for a palm-sized computer according to one embodiment of the invention.

FIG. **13** shows a user dialing screen for a palm-sized computer according to one embodiment of the invention.

FIG. **14** shows a telephone dialing screen for a palm-sized computer according to one embodiment of the invention.

FIG. **15** shows a connected line screen for a palm-sized computer according to one embodiment of the invention.

FIG. **16** shows a connected line screen with a line pop-up menu window for a palm-sized computer according to one embodiment of the invention.

FIG. **17** shows a transfer dialog screen for a palm-sized computer according to one embodiment of the invention.

FIG. **18** shows a transfer waiting screen for a palm-sized computer according to one embodiment of the invention.

FIG. **19** shows a transfer completed screen for a palm-sized computer according to one embodiment of the invention.

FIG. **20** shows an active call summary list screen for a palm-sized computer according to one embodiment of the invention.

FIG. **21** shows a first forwarding screen for a palm-sized computer according to one embodiment of the invention.

FIG. **22** shows a second forwarding screen for a palm-sized computer according to one embodiment of the invention.

FIG. **23** shows a call history screen for a palm-sized computer according to one embodiment of the invention.

FIG. **24** shows a detailed single call history screen for a palm-sized computer according to one embodiment of the invention.

FIG. **25** shows an incoming call window for a palm-sized computer according to one embodiment of the invention.

FIG. **26** shows a features menu bar for a palm-sized computer according to one embodiment of the invention.

FIG. **27** shows an options menu bar for a palm-sized computer according to one embodiment of the invention.

FIG. **28** shows a dial screen for a palm-sized computer according to one embodiment of the invention.

FIG. **29** shows a copyright information screen for a palm-sized computer according to one embodiment of the invention.

FIG. **30** shows a phone settings screen for a palm-sized computer according to one embodiment of the invention.

FIG. **31** shows a user settings screen for a palm-sized computer according to one embodiment of the invention.

FIG. **32** shows a change password screen for a palm-sized computer according to one embodiment of the invention.

FIG. **33** shows an enter password screen for a palm-sized computer according to one embodiment of the invention.

FIG. **34** shows a conferencing screen for a palm-sized computer according to one embodiment of the invention.

FIG. **35** shows a conference option window for a palm-sized computer according to one embodiment of the invention.

DETAILED DESCRIPTION

Connection Scheme for Communications System

A first aspect of the invention provides a communications system for companion information and network appliances. The information appliance is typically a portable computer (shown in FIG. **3**A as reference number **320**), and the network appliance is typically a network attached, or connected telephone **240**. A connection scheme **200** for one embodiment of the communications system is shown in FIG. **2**. The communications system includes a network appliance shown in FIG. **2** as a telephone **240**, a information appliance **210** (e.g., a portable computer) connected to the telephone **240**, and a network link connecting the telephone **240** to network connected devices.

The network appliance is typically a network connected or attached telephone **240**, and can be an Ethernet telephone (shown in FIG. **3**A as reference number **310**), a cable television set top box, a personal computer, a workstation, or any other network connected device that communicates with other network connected devices providing telephony communications. The telephone **240** has capabilities such as a maximum number of connected lines, and a maximum number of conference lines.

The network link is shown in FIG. **2** as a local area network link **250**. The portable computer **320** includes a physical layer communications port [shown in FIG. **3**A through FIG. **3**D as an Electronic Industries Association/Telecommunications Industry Association (EIA/TIA) 232 serial port **323**] for connecting to the telephone **240**, a memory storing user information corresponding to a user; and processing resources adapted to exchange data with the telephone **240**.

The data includes the user information and data corresponding to the telephone **240** capabilities. The exchange of the data enables the portable computer **320** to discover

capabilities of the telephone **240**, provide the user information to the telephone **240**, and establish telephone **240** operating parameters for telephone communications with devices connected to the telephone **240** based on the user information and the telephone capabilities. The telephone **240** capabilities are typically transferred from the telephone **240** to the information appliance **210** during initialization of the companion appliances. Initialization typically occurs upon connecting and power-up of the companion appliances.

The different types of data exchanged between the telephone **240** and the information appliance **210** of the communications system are discussed in greater detail below in the Data Link Layer Protocol section in conjunction with FIG. **4**, and in the Ethernet Telephone Management Protocol Frame Descriptions section in conjunction with FIGS. **5**A through **5**K. State diagrams illustrating selected exchanges are provided in FIGS. **6** through **9** and are discussed below in the Interactions between the Telephone and the Information Appliance section. In some embodiments of the communication system, the data exchanged between the telephone **240** and the information appliance corresponds to the telephone **240** capabilities and the user information, and is formatted according to an applications layer protocol having frame formats for telephony functions.

FIG. **2** illustrates the connections between the telephone, the portable computer **320**, the telephone, the local area network, and services connected to the local area network. The connection scheme **200** illustrates that for some embodiments, the information appliance **210** provides a user interface **220** and a primary control function **230**.

The information appliance **210** can be any device capable of storing user information and exchanging information with a network connected appliance according to the communications protocol hierarchy with Application Program Interface (API), Point to Point protocol (PPP), and High-level Data Link Control (HDLC) layers for telephony applications **300** described in conjunction with FIG. **3**A below. More typically the information appliance **210** is a portable computer **320** (such as a palm-sized computer shown in FIG. **3**B as reference number **343**) or a desktop computer.

The user interface **220** typically comprises a display, however any device, or element thereof, that can provide sensory cues adapted for perception by the user can be used as a user interface according to the invention. The user can provide inputs for the display by tapping a user interface graphic element image on the screen with a stylus, pressing a button or key on the information appliance **210**, or by any other means known in the art. Details of the display and certain user interface graphical elements and user inputs related thereto for some embodiments are illustrated in FIGS. **10** through **35** below and are discussed in the User Interface Graphical Elements and Inputs section below.

In some embodiments the telephone **240** provides simultaneous data and telephony communications. In order for the telephone **240** to act simultaneously as an Internet access device and a telephone **240**, the information appliance provides multiprocessing.

While the telephony application is running in the foreground, a packet data communications program runs in the background to check if any packets have been received on the telephone **240** communications port. If any packets have been received, the packet data communications program forwards the packets to the appropriate application depending on whether the packets are formatted according to the Ethernet Telephone Management Protocol (EMP) **333** (discussed below with reference to FIGS. **3** through **5**) or IP. If the received packet is an Internet Protocol (IP) packet, then the program forwards it to the appropriate network application corresponding to the application protocol, such as simple mail transfer protocol (SMTP) for email packets, or hyper text transfer protocol (HTTP) for packets received from the World Wide Web.

The primary control function **230** enables the user of the information appliance **210** to establish telephone operating parameters for the telephone **240**, and to control the execution of telephony programs with network connected devices. The advantages of placing the primary control function **230** a portable information appliance **210**, such as a palm-sized computer **343** (sometimes referred to as personal digital assistant, or a PDA) are commensurate with the portability of the palm-sized computer **343** and the personalization of the telephone operating parameters that can be accomplished for a palm-sized computer **343** that is used by a single user. The information appliance **210** communicates user settings for telephony operations to the telephone **240** during initialization of the companion appliances. The user settings established during initialization can be supplemented by the user entering user settings for an upcoming session before beginning a particular set of communications with network connected devices using the telephone **240**.

The communications protocols used for data exchanged between the information appliance **210** and the telephone **240** are discussed in more detail in the "protocol design" and "protocol layer scheme" sections of this application, in conjunction with FIGS. **3**A and **3**B.

In some embodiments, the communications system includes a gateway server **270** connected to the network link, and switched circuit network devices connected to the gateway server **270**. In some embodiments, as shown in FIG. **2**, the telephone **240** is connected to network services and network connected devices through a local area network (LAN) link **250**. The LAN link **250** connects the telephone **240** to a directory server **260**, a gateway server **270**, a router **275**, and a gatekeeper **280**.

The LAN link can be through a digital subscriber line (DSL), a twisted-pair cable, an integrated services digital network (ISDN) link, or any other link that supports packet switched communications with a LAN, including Internet Protocol (IP)/Transmission Control Protocol (TCP) communications using an Ethernet. The gateway server **270** is connected to regional, national, and global services across wide area network links to public switched telephone networks (PSTN) **290**, and other switched circuit networks (SCNs), included in the other communications networks **296** shown in FIG. **2**. The other communications networks **296** can include, broadband distribution channels, wireless networks, restricted access government and corporate networks, or any other communications network capable of transmitting data formatted for SCNs or packet based networks (PBNs).

In some embodiments, the telephone **240** is an Ethernet telephone (shown in FIG. **3**A as reference number **310**) and the gateway server **270** transforms the data formatted for a packet based network received from the telephone **240** (or any other gateway server **270** client or terminal), to data formatted for a switched circuit network for transmission to the PSTN **290** or any other SCN. The gateway server **270** also transforms the data formatted for switched circuit network devices, such as the data received from the PSTN **290** or any other SCN, to data formatted for a packet based network device such as the telephone **240**.

In some embodiments, the communications system includes a router connected to the network link, and packet based network (PBN) devices connected to the router. For the PBN connected devices no gateway server **270** data transformation is needed. Therefore, data transmitted from the telephone **240** proceeds from the LAN link **250** to a router **275**, and then to the desired PBN such as an Internet source **293** or other PBN **294** and then to network connected devices accessible therefrom. The telephone **240** can also transmit the data directly through the router **275** to devices connected directly to the router **275**. Network and transmission layer formats for these communications can be according to the IP/TCP protocols.

The Ethernet telephone **310** communicates with the gatekeeper **280**, the directory server **260**, the gateway server **270**, and other Ethernet telephones using a telephony application layer protocol, otherwise referred to as a soft private branch exchange (PBX), for example an protocol compliant with the International Telecommunication Union ITU-T recommendation H.323 "Packet-based multimedia communication systems", or a Session Initiation Protocol (SIP).

The information appliance **210** stores address information according to a particular user database. The address information stored in the user database and address information entered by the information appliance **210** user is provided to the telephone **240** as an alias (or high-level) address such as a telephone number using the protocols discussed in the protocol design section below. The telephone **240** transmits the alias address to the gatekeeper **280** using a registration, admission and status (RAS) compliant protocol. Certain details of an RAS protocol are provided by the International Telecommunication Union (ITU) in the ITU-T recommendation H.323 and for transmission formats in the ITU-T recommendation H.225 "Media stream packetization and synchronization on non-guaranteed quality of service LANs". The RAS signaling function performs registration, admission, bandwidth changes, status and disengagement procedures between endpoints (e.g., the telephone **240**) and the Gatekeeper **280**.

For embodiments operating according to the H.323 recommendation, a gatekeeper **280** can service endpoints such as gateways (such as the gateway server **270**), terminals (also referred to as client terminals, e.g., the telephone **240**), and multipoint control units. Each multipoint control unit supports conferences between three or more endpoints. The collection of all terminals, gateways, and multipoint control units managed by a single gatekeeper **280** operating according to the H.323 protocol is referred to as an H.323 zone. The gatekeeper **280** acts as the central point for all calls within its zone and provides call control services to a plurality of registered endpoints.

Call control function communications between the telephone **240** and the gatekeeper **280** include control messages governing operation of the telephone **240**. These messages address capabilities exchange, opening and closing of logical channels, preference requests, flow control messages, and general commands and indications and are transmitted over a reliable ITU H.245 control channel. The companion appliances enable the user to provide appropriate inputs into the control channel messages using communications from the information appliance **210** to the telephone **240** in messages formatted according to the data link layer protocol (DLLP **328**) and EMP discussed below.

For H.323 embodiments, the telephone **240** uses an ITU Q.931 call signaling channel to establish a connection with another network connected device through the gatekeeper **280**.

The gatekeeper **280** performs two important call control functions. The first is address translation from LAN aliases, such as those provided by the telephone **240**, to Internet Protocol (IP) or Internet Packet Exchange (IPX) addresses, as defined in the RAS specification. The gatekeeper **280** transmits a message including the LAN alias addresses provided by the telephone **240** to the directory server **260**.

The directory server **260** includes a registry table having alias addresses and corresponding network (e.g., SCN, IP or IPX) addresses for the registered network connected devices. In response to the message from the gatekeeper, the directory server **260** accesses the registry table and transmits a message including the corresponding network addresses to the gatekeeper **280**. The gatekeeper **280** then provides the network addresses to the telephone **240**. The telephone **240** then places the network address in the message to be sent to the network connected device and transmits the message to the device through the gateway server **270** or router **275**.

The directory server **260** also enables the user to properly address messages sent to network connected device addresses that are not included in the address information stored in the information appliance **210**. The user can retrieve and store such addresses in the information appliance **210** for future use. Similarly, the user can provide new address information for inclusion in the directory server **260** registry table, so that other users can determine the corresponding network addresses for network connected devices that were not previously included in the directory server **260** database.

Method for Transmitting Data from a Portable Computer to a Telephone

The second aspect of the invention provides a method for transmitting data from a portable computer **320** to a telephone **240**. The telephone **240** has operating capabilities. The telephone **240** is connected to network connected devices. The method comprises connecting the portable computer **320** with the telephone **240**, supplying the portable computer with telephone operating data, the portable computer exchanging the telephone operating parameter data and the operating capabilities with the telephone, and the portable computer establishing telephone operating parameters for a communications session.

In some embodiments, the method includes the user starting a telephony program with the portable computer **320** controlling execution of the telephony program. In some embodiments, the telephone **240** accepts the telephone operating parameters after the establishing step. In other embodiments, the telephone **240** can accept or reject the telephone operating parameters after the establishing step, depending on network conditions or whether more than one portable computer **320** is connected to the telephone.

The telephone **240** operating parameter data is for a communications session including an exchange of messages with one or more of the network connected devices. In some embodiments, the communications session includes simultaneous exchanges of voice and packet data messages.

The telephone **240** operating parameters for the communications session are based on the telephone operating parameter data and the operating capabilities. The telephone **240** operating parameters provide options and features for the communications session. In some embodiments, the operating parameter data comprises constructs formatted according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions. In some embodiments, the operating parameter data com-

prises constructs formatted according to a high-level data link control protocol (shown in FIG. **3**A as reference number **326**).

In some embodiments, prior to connecting the portable computer **320** with the telephone **240**, the portable computer stores user information. The user information can include an identification corresponding to the portable computer **320**, user access parameters, and user characteristics corresponding to the telephone **240** operating parameter data. For some of these embodiments, establishing telephone **240** operating parameters includes the user selecting user setting inputs. The user setting inputs corresponding to the portable computer **320** identification, user access parameters, and user characteristics, the user selecting change the corresponding telephone **240** operating parameter data. In one embodiment, the user setting inputs include the network address of the telephone, so that the user can change the network address provided by the telephone to be the network address of the portable computer.

In some embodiments, after the establishing step, the method includes one or more of the following programs: conferencing, dialing, receiving an incoming call, forwarding, transferring, and placing a call. For some of these embodiments, the telephone **240** receives input data from the portable computer **320**.

The input data is formatted according to a data link layer protocol (shown in FIG. **3**B as reference number **328**). The data link layer protocol **328** encapsulates frames formatted according to an application layer protocol adapted for telephony functions, i.e., the Ethernet telephone management protocol (EMP) shown in FIG. **3**B as reference number **333**. The telephone **240** transforms the input data into transport data formatted according to a transport protocol for a packet switched network. The telephone **240** transmits the transport data to a gateway server **270**. The gateway server **270** is connected by a local area network link **250** to at least one switched circuit network including a public switched telephone network **290**.

In some embodiments, the network connected devices include a gateway server **270**. The gateway server **270** provides access for the user to a public switched telephone network **290**. For some of these embodiments, the telephone **240** is also connected to a gatekeeper **280** and a directory server **260** by a local area network link **250**. Communications between the telephone **240** and the gatekeeper **280**, gateway server **270**, and directory server **260** are formatted according to a soft private branch exchange telephony application layer protocol.

In some embodiments, the telephone **240** is connected by a local area network link **250** to a router **275**. The router **275** is connected to at least one packet based network including an Internet source **293**. Communications between the router **275** and the telephone **240** are formatted according to packet based network application protocols.

In some embodiments the method includes a user placing a phone call after the establishing telephone **240** operating parameters step. Placing the phone call includes the user starting a dialing program and the user inputting values to the portable computer **320**. The values correspond to a recipient network connected device. The portable computer **320** displays the values and phone call status information. The portable computer **320** transforms the values into input data formatted according to a data link layer protocol **328**. The data link layer protocol **328** encapsulates frames formatted according to an application layer protocol (such as the EMP **333**). The application layer protocol is adapted for telephony functions. The portable computer **320** transmits the input data to the telephone **240**.

For some of the embodiments including a dialing program, the user enters text data to form a memo corresponding to the telephone **240** call after the transmitting step. The portable computer **320** creates a data record corresponding to the telephone **240** call, and the portable computer attaching the memo to data record.

For some of the embodiments including a dialing program, the portable computer **320** includes a user interface **220** and a display (such as the palm-sized computer **343** display **1005** shown in FIG. **10**). The inputting is accomplished through the user interface **220**. The portable computer **320** displays a telephone number entry field in which the user inputs the values.

In some embodiments the method includes a user placing a conference call after the establishing telephone **240** operating parameters step. Placing the conference call includes the user starting a conferencing program and the user inputting values to the portable computer **320**. The values correspond to a plurality of conference participant network connected devices. The portable computer **320** displays the values and conference call status information. As for placing the call, the portable computer **320** transforms the values into input data formatted according to a data link layer protocol **328**. The data link layer protocol **328** encapsulates frames formatted according to an application layer protocol. The application layer protocol is adapted for telephony functions. The portable computer **320** transmits the input data to the telephone **240**.

For some of the embodiments including a conferencing program, the portable computer **320** includes a display **1005**. The starting includes a user selection of a user interface element for a conferencing feature (such as the "conference option" element shown in FIG. **26** as reference number **2640**). The user interface element is disposed on the display **1005**. In response to the user selection, the portable computer **320** displays a list of conference actions (such as those shown in FIG. **34**) for subsequent user selection.

In some embodiments the method includes a user forwarding a call after the establishing telephone operating parameters step. The forwarding includes the user starting a forwarding program, and the user inputting a number to the portable computer **320**. The number corresponds to a forwarding destination network connected device. The portable computer **320** displays the number and forwarding status information. The portable computer **320** transforms the values into input data formatted according to a data link layer protocol **328**. The data link layer protocol **328** encapsulates frames formatted according to an application layer protocol. The application layer protocol is adapted for telephony functions. The portable computer **320** transmits the input data to the telephone **240**.

In some embodiments the method includes connecting a user to a first network connected device after the establishing telephone **240** operating parameters step. The first network connected device having a first number. The user transfers a call from the first network connected device to a second network connected device. The transferring includes the user inputting a second number into the portable computer **320** corresponding to the second network connected device. The portable computer **320** then displays the first and second numbers, and status information corresponding to the first network connected device and the second network connected device. The portable computer **320** then transforms into input data formatted according to a data link layer

protocol **328**. The data link layer protocol **328** encapsulates frames formatted according to an application layer protocol. The application layer protocol is adapted for telephony functions. The portable computer **320** transmits the input data to the telephone **240**.

In some embodiments, the portable computer **320** corresponds to a user and has a display **1005**. After the establishing telephone operating parameters step, the method includes the portable computer **320** receiving an incoming call. Receiving the incoming call includes the portable computer **320** receiving an incoming call message from the telephone **240**, the incoming call message (such as the incoming call message **632** for the second 02 line, as shown in FIG. **6**) indicating that a first network connected device is waiting to start a call with the user. The portable computer **320** displays an incoming call screen (such as the call incoming window shown in FIG. **25** as reference number **2500**) on the display **1005**. For some of these incoming call embodiments, the portable computer **320** includes an address database. The incoming call message includes a caller name, and a caller identification. The incoming call screen includes a user selection for saving the caller name and the caller identification to the address database (such as the "save" button shown in FIG. **25** as reference number **2530**). For some of these incoming call embodiments, the incoming call screen includes user selections for processing the incoming call. The user selections include rejecting the incoming call (shown in FIG. **25** as the "reject" button **2550**), answering the incoming call (shown in FIG. **25** as the "answer" button **2540**), and answering the call at a later time (shown in FIG. **25** as the "OK" button **2560**).

In some embodiments, after the establishing telephone **240** operating parameters step, the method includes a user placing a phone call. Placing the phone call includes the telephone **240** receiving input data from the portable computer **320**. In some embodiments, after the establishing telephone **240** operating parameters step, the method includes a user placing a conference call. Placing the conference call includes the telephone **240** receiving input data from the portable computer **320**. In some embodiments, after the establishing telephone **240** operating parameters step, the method includes a user forwarding a call. Forwarding the call includes the telephone **240** receiving input data from the portable computer **320**. In some embodiments, after the establishing telephone **240** operating parameters step, the method includes connecting a user to a first line, the first line having a first number. The user then transfers a call from the first line to a second line. The transferring includes the telephone **240** receiving input data from the portable computer **320**.

For the embodiments described in the preceding paragraph, the input data is formatted according to a data link layer protocol **328**. The data link layer protocol **328** encapsulates frames formatted according to an application layer protocol. The application layer protocol is adapted for telephony functions. The telephone **240** transform is the input data into transport data formatted according to a transport protocol for a packet switched network. The telephone **240** transmits the transport data to a gateway server **270**. The gateway server **270** is connected to at least one switched circuit network including a public switched telephone network **290**.

In some embodiments, after the establishing telephone **240** operating parameters step, the method includes the telephone **240** receiving an incoming call from a first network connected device. The receiving includes the telephone **240** receiving data from a gateway server **270** indicating that a first network connected device is waiting to start a call with a user. The telephone **240** transforms the data into an incoming call message formatted in the data link layer protocol **328**. The data link layer protocol **328** encapsulates frames formatted according to an application layer protocol. The application layer protocol is adapted for telephony functions. The telephone **240** transmits the incoming call message to the portable computer **320**.

In some embodiments, after the connecting, the method includes powering up the portable computer **320**. In response to the powering up of the portable computer **320**, the portable computer and the telephone **240** are initialized. The initializing include the exchanging and establishing steps.

In some embodiments, the portable computer includes a display **1005**. After the establishing step, the method includes a user starting a telephony program. The starting includes the portable computer displaying a user interface element corresponding to a first menu (such as the line pop-up menu shown in FIG. **16** as reference number **1610**) on the display **1005**. The first menu includes a list of telephony programs available for a particular connected line. The user selects the first menu user interface element (such as the "line" prompt in FIG. **16**). The portable computer **320** displays a first (line pop-up) menu **1610** list, the first menu list including user interface elements corresponding to the telephony programs. The user selects a program from the list. For some of these embodiments, the portable computer **320** simultaneously displays information corresponding to the connected line and the connection thereto on the display **1005** with the first (line pop-up) menu **1610** list.

In some embodiments, the portable computer **320** includes a display **1005**. After the establishing step, the method includes the user starting a telephony program. The starting includes the user selecting a button corresponding to a second menu. In one embodiment, the second menu is the features bar menu (shown in FIG. **26** as reference number **2600**) and the button is the features bar. The button is provided on the display **1005**. The second menu includes a list of programs. Execution of the programs is controlled by the portable computer **320**. The user then selects a user interface element corresponding to the program (such as the "dial" option shown in FIG. **26** as reference number **2610**) from the list of programs.

In some embodiments, the method includes prior to connecting the portable computer **320** with the telephone **240**, the portable computer **320** storing user information. The user information includes an identification corresponding to the portable computer **320**, user access parameters, and user characteristics corresponding to the telephone **240** operating parameter data. Establishing telephone **240** operating parameters includes the user selecting user setting inputs. The user setting inputs corresponding to the portable computer **320** identification, user access parameters, and user characteristics, the user selecting change the corresponding telephone **240** operating parameter data. For some of these embodiments, the portable computer **320** includes a display **1005**. The user selecting of user setting inputs includes the portable computer **320** providing user interface elements corresponding to the user setting inputs on the display **1005**. The user interface elements include level indicators, selection buttons, and a set button. The user raises and lowers the level indicators, selects the selection buttons; and the set button. For some of these embodiments, the user setting inputs include the network address of the telephone **240**.

In some embodiments, establishing telephone **240** operating parameters includes the user selecting phone setting

inputs. For some of these embodiments, the portable computer includes a display **1005**. The user selecting phone setting inputs includes the portable computer **320** providing user interface elements corresponding to the phone setting inputs on the display **1005**. The user interface elements including level indicators, selection buttons, and an OK button. The user raising and lowering the level indicators, selecting the selection buttons, and selecting the OK button.

In some embodiments, the portable computer includes a display **1005**. The display **1005** has a features menu bar. In response to user selection of the features menu bar, a selection list having a plurality of features is provided on the display **1005**.

In some embodiments, the portable computer includes a display **1005**. The display has an options menu bar. In response to user selection of the options menu bar, a selection list having a plurality of options is provided on the display **1005**.

In some embodiments, the network connected devices include a gateway server **270**, the gateway server provides access to a public switched telephone network **290**.

In some embodiments, the method includes, prior to the connecting step, the telephone **240** exchanging voice messages with at least one of the network connected devices.

In some embodiments, the telephone **240** is connected to a gatekeeper **280**, a directory server **260** and a gateway server **270** by a local area network link **250**. Communications between the telephone **240** and the gatekeeper **280**, gateway server **270**, and directory server **260** are formatted according to a soft private branch exchange telephony application layer protocol.

In some embodiments, the telephone **240** is connected by a local area network link **250** to a router **275**. The router **275** is connected to at least one packet based network including an Internet source **293**. Communications between the router **275** and the telephone **240** are formatted according to packet based network application protocols. For some of these embodiments, the portable computer **320** includes processing resources for Internet access and the telephone **240** includes processing resources for Internet access.

For some of the packet based network connected embodiments as shown in FIG. **3**C, the portable computer **320** includes processing resources for Internet access including Internet applications **367**, transmission control **359** software, and Internet protocol **356** software. The telephone **240** includes an Internet access application **369**, transmission control **359** software, carrier sense multiple access/collision detection **368** software, and Internet protocol **356** software.

For some of the packet based network connected embodiments as shown in FIG. **3**D, the portable computer **320** includes processing resources for Internet access including Internet display applications **375** and display/user input transfer software **373**. The telephone **240** includes processing resources for Internet access including Internet applications **367**, display/user input transfer software **373**, transmission control **359** software, Internet protocol **356** software, and carrier sense multiple access/collision detection **368** software.

In some embodiments, the telephone **240** is connected to a gatekeeper **280**, a directory server **260** and a gateway server **270** by a local area network link **250**. Communications between the telephone **240** and the gatekeeper **280** are formatted according to registration admission and status signaling function control. Communications between the telephone **240** and the gateway server **270** and the directory server **260** are formatted according to an H.323 compliant protocol.

In some embodiments, the method includes the user starting a telephony program. The portable computer **320** controls execution of the telephony program.

Method for Exchanging Voice and Data Messages Between a Companion Telephone and Devices Connected to the Network

The third aspect of the invention provides a method for exchanging voice and data messages between a telephone **240** and devices connected to a network. The telephone **240** is connected to the network. The method comprises connecting the telephone **240** with a portable computer **320**. The portable computer **320** exchanges telephone **240** operating parameter data with the telephone. The operating parameter data provides options for communications between the telephone **240** and the network connected devices. Responsive to a user indication of a desired communication, the portable computer **320** exchanges call data with the telephone **240**. The call data corresponds to the desired communication. The call data is formatted according to an application layer protocol, and the underlying transport, network, and data link layer protocols. The application layer protocol has frame formats for telephony functions. The telephone **240** exchanges messages with an addressed network connected device. The messages corresponding to the desired communication. The addressed network connected device has a network address. The message includes data corresponding to the address of the addressed network connected device.

In some embodiments, the method includes, prior to connecting the portable computer **320** with the telephone **240**, the portable computer storing user information. The operating parameter data comprises the user information. The user information comprises an identification corresponding to the portable computer **320** and user access parameters.

In some embodiments, the method includes the telephone **240** requesting a connection to a first network connected device. The first network connected device responds to the connection request. In response to a user input, the message data comprise a phone number corresponding to the first network connected device transmitted from the portable computer **320** to the telephone **240**. Upon receipt by the telephone **240** of the first network connected device response to the connection request, the message data comprise a first connection made response transmitted from the telephone **240** to the portable computer **320**. For some of these embodiments, the portable computer **320** further comprises a display **1005**. The method further comprises, upon receipt of the first connection made response, the portable computer **320** providing on the display **1005** a representation of a date/time connected, and a connected time corresponding to the first network connected device response.

In some embodiments, the telephone **240** comprises an Ethernet telephone **310**.

In some embodiments, the telephone **240** is connected to a gatekeeper **280**, a directory server **260** and a gateway server **270** by a local area network link **250**. Communications between the telephone **240** and the gatekeeper **280**, the gateway server **270**, and the directory server **260** are formatted according to a soft private branch exchange telephony application layer protocol.

In some embodiments, the telephone **240** connected by a local area network link **250** to a router **275**. The router **275** is connected to at least one packet based network including an Internet source **293**. Communications between the router **275** and the telephone **240** are formatted according to packet based network application protocols.

In some embodiments, the telephone **240** is connected to a gatekeeper **280**, a directory server **260** and a gateway server **270** by a local area network link **250**. Communications between the telephone **240** and the gatekeeper **280** are formatted according to a registration admission and status signaling function control. Communications between the telephone **240** and the gateway server **270** and the directory server **260** are formatted according to an H.323 compliant protocol.

In some embodiments, the exchange of messages includes simultaneous exchanges of voice and packet data messages.

In some embodiments, the telephone **240** has an identification, and an identification corresponding to the portable computer **320** is presented to the network connected devices by the telephone in place of the telephone identification.

Protocol Design

A communications protocol hierarchy with API/PPP/HDLC layers for telephony applications **300** for companion information and network appliances is provided below in FIG. **3**A. The communications protocol communications protocol hierarchy with API/PPP/HDLC layers for telephony applications **300** can be implemented for any information appliance **210** combined with any telephone **240**. The telephone **240** can be an Ethernet telephone **310** as shown in FIG. **3**A. The information appliance **210** is typically a portable computer **320** as shown in FIG. **3**A. The portable computer **320** can be a laptop, note book, or palm-size computer. Exemplary palm-sized computers **343** for which this invention is particularly well-suited include the PalmPilot Professional, and the Palm III, available from Palm Computing, Mountain View, California.

The communications protocol communications protocol hierarchy with API/PPP/HDLC layers for telephony applications **300** is specially adapted for implementation with a palm-sized computer **343** and an Ethernet telephone **310**.

The communications protocol communications protocol hierarchy with API/PPP/HDLC layers for telephony applications **300** enables the telephone **240** and the information appliance **210** to carry out several telephony features, including call forwarding and conferencing. In order to establish the capabilities of the combined companion appliances, the information appliance **210** is connected to the telephone **240** according to a physical layer protocol.

The information appliance **210** exchanges operating parameter data with telephone **240**. For some embodiments, the operating parameter data include the portable computer **320** identification and user information. The identification and user access information establish a network address for the combined companion appliances, so that messages transmitted to the user have a network destination address that corresponds to the information appliance **210**, not the telephone **240**.

As the user moves among many different locations, the use of the information appliance **210** destination address allows network connected devices to contact the user as soon as the information appliance **210** is connected to the telephone **240**, and the information appliance **210** exchanges the operating parameter data with the telephone **240**. Use of the information appliance **210** destination address is especially useful for embodiments where the information appliance **210** comprises a portable computer **320**, or a palm-sized computer **343**.

The operating parameter data includes user information that describes other user characteristics. User characteristics can relate to user access, security and service level profiles, smart card functionality for purchases, and/or financial account information. The profiles can be used to determine user access to long distances services, firewall protected data, or user access to secure communication networks. Prior to connecting information appliance **210** the with the telephone **240**, the information appliance **210** stores the user information.

Companion Appliance Communications Protocol Hierarchy

The communications protocol communications protocol hierarchy with API/PPP/HDLC layers for telephony applications **300** (or software architecture) for one embodiment of the invention is shown in FIG. **3**A. The physical layer communications port connecting the Ethernet telephone **310** and the portable computer **320** is shown in FIG. **3**A as an EIA/TIA-232 serial port **323** (formerly RS-232-C) serial port. The invention works with other ports including EIA/TIA-422 and EIA/TIA-423 serial ports, parallel ports, or wireless connections such as infrared ports, connecting the two companion appliances.

For this embodiment, the data link layer protocol comprises the high-level data link control (HDLC) protocol **326** and the Point-to-Point Protocol (PPP) **330**. Typically, for the purposes of HDLC the information appliance **210** is designated as the primary station and the network connected telephone **240** is designated as the secondary device. The information appliance is shown in FIG. **3**A as a portable computer **320**, and the network connected telephone **240** is shown in FIG. **3**A as an Ethernet telephone **310**. The companion appliances can operate in a normal unbalanced mode, an asynchronous mode, or an asynchronous balanced mode.

The Point-to-Point Protocol (PPP) **330** part of the data link protocol, as described in the Internet Engineering Task Force Request for Comments **1661**, provides a method for encapsulating multi-protocol datagrams, a link control protocol for establishing, configuring, and testing the data-link connection, and a family of network control protocols for establishing and configuring different network-layer protocols. PPP **330** supports simultaneous multi-protocol transport of TCP/IP, Internetwork Packet Exchange (IPX), DECnet, and Appletalk traffic on the same connection. The Point to Point protocol (PPP) **330** layer provides a method for connecting the information appliance **210** to an Internet source **293**.

An Application Program Interface (API) **336** layer acts as a boundary across which application software available to the companion appliances uses facilities of programming languages to invoke services provided by the network connected servers. The API **336** layer specification provides a mapping of functions that are made available by the network connected service providers into the syntax and semantics of the programming languages used by the application platform. The API **336** layer thereby provides methods enabling the telephone **240** to connect the information appliance **210** to the Internet, or an Internet Source **293** and the other LAN link **250** connected devices, services, and networks.

In one embodiment, the communications protocol hierarchy is as shown in FIG. **3**B. In this embodiment's communications protocol hierarchy with EMP/DLLP **328** for telephony applications **340**, data link layer protocol (DLLP) **328** communications are formatted as shown in FIG. **4**. The DLLP **328** transforms data provided via the communication port into data formatted for processing by the applications

layer protocol also referred to herein as the Ethernet Telephone Management Protocol (EMP) **333**. The EMP is discussed in detail below.

Internet Access

The combination of the information appliance **210** and the telephone **240** can also be used to provide Internet access. The communications protocol hierarchy for Internet access, otherwise referred to as the software architecture, used to support Internet access depends on the capabilities that exist in the information appliance **210**. In one embodiment, the information appliance **210** is a palm-sized computer **343** that includes a Transmission control protocol (TCP)/Internet Protocol (IP)/PPP stack (as is the case for the **3**Com Palm III and the PalmPilot Professional). This embodiment is referred to herein as the palm-sized computer **343** Internet access execution option. For the first case, as shown in FIG. **3**C, the protocol stack can be represented as a palm-sized computer **343** running the Internet applications protocol stack **350**. The protocol stack can be located in any information appliance **210** to provide Internet access according to the invention.

The palm-sized computer **343** running the Internet applications protocol stack **350** includes the following software layers, in descending order, for Internet access for the palm-sized computer **343**: an Internet applications **367** layer, a TCP **359**/User Datagram Protocol (UDP) layer, an Internet protocol **356** layer, a PPP **330** layer, and an HDLC **326** layer. The Internet applications **367** include electronic mail, web browsing, terminal emulation (telnet), file transfer protocol (ftp) and other applications providing access to data provided by the Internet. The palm-sized computer **343** running the Internet applications protocol stack **350** for the Ethernet telephone **310** is the same as the protocol stack for the palm-sized computer except that: (1) the PPP **330** and HDLC layers are replaced by an IEEE 802-3 Carrier Sense Multiple Access/Collision Detection (CSMA/CD) **368** layer, and (2) the top layer of the Ethernet telephone **310** stack includes only Internet access applications **369**.

As before the Ethernet telephone **310** is shown as connected to the palm-sized computer **343** by an EIA/TIA-232 serial port **323**. The Ethernet telephone **310** is shown as connected to the network connected devices by an Ethernet link **353**. More generally a telephone **240** connected to a switched circuit network (SCN) by any appropriate LAN link will work according to the invention.

The palm-sized computer **343** running the Internet applications **367** protocol stack **350** shown in FIG. **3**C also includes the following software layers for access to telephony applications: a phone applications layer **365**, a phone API **336**/EMP layer, a PPP **330** layer, and an HDLC layer **326**. Note that the phone API **336**/EMP layer is shown to cover both the communications protocol hierarchy with API/PPP/HDLC for telephony applications **300**, and the communications protocol hierarchy with EMP/DLLP **328** for telephony applications **340**.

However, the communications protocol hierarchy with EMP/DLLP **328** for telephony applications **340** actually has DLLP **328** replacing the PPP **330** layer and the HDLC layer **326**. For Internet access using the DLLP **328**, an additional field is added to the DLLP frame format shown in FIG. **4** below, to indicate whether the next upper layer protocol is IP **356**, or EMP **333**.

Given a palm-sized computer **343** with multitasking capability, the Internet access (or other data communications) and telephony communications (including voice communications) can occur simultaneously using the software provided in the palm-sized computer **343** running the Internet applications **367** protocol stack **350**.

Another option for providing Internet access to the user is to have the Ethernet telephone **310** run the Internet applications **367** while the palm-sized computer act as a user interface. This option is referred to herein as the Ethernet telephone **310** Internet access execution option. The Ethernet telephone **310** Internet access execution option arrangement can be advantageous for palm-sized computers having very limited storage capacity to accommodate the TCP **359**/IP **356** stack and the Internet applications **367**. In one embodiment, the Ethernet telephone **310** Internet access execution option is achieved by providing another protocol next to EMP **333**. This protocol is shown in the Ethernet telephone **310** running the Internet applications **367** protocol stack **370** as the display/user input transfer protocol **373**. The display/user input transfer protocol **373** sends display information to the palm-sized computer **343** from the telephone **240** and sends user inputs (such as entered text, selected links, selected buttons, etc.) to the Internet application **368** running on the telephone **240**. Note also that an Internet display application **375** is the top level software operating on the palm-sized computer **343** for the Ethernet telephone **310** Internet access execution option.

Operations

The EMP **333** and DLLP **328** hierarchy supports a number of operations and features related thereto including those listed below.

1) Placing a phone call: telephone **240** number autodialing from the address book;
2) Receiving an incoming call: caller identification (ID) and caller name, interrupt other applications with "HotSync" key, answer, reject, cancel (ignore), save (record to "Address" application);
3) A user interface for PBX-like features such as:
   a) Forwarding a call: when forwarding is set, all calls are automatically diverted to the forwarded phone (both to one or more phone numbers and from another phone, i.e., go to roaming mode), Features: activate/de-activate, forwarded information
   b) Conference calls: Features include conference, add parties, hang up, split conference, hold conference, switch;
   c) Transferring a call (i.e., during a call, after the parties have conversed for a period of time);
4) Keeping a history of past numbers communicated with (both incoming and outgoing calls), which can also be used for redialing;
5) Voice mail user interface;
6) Capability exchange with the network appliance, e.g., the telephone **240** and the subsequent adjustment of the user interface features during initialization, information including the number of lines and conferences supported; speaker, ringer and handset volume; transfer and forward feature support; Ethernet telephone **310** and Palm-sized computer **343** Ids; call restrictions for the user; and line status are exchanged;
7) Changes to the user interface and the telephone **240** operating parameters can also be made by user adjustment of user settings through the information appliance user interface after initialization for earphone/ringer volume levels, etc.;
8) Internet access;

9) Simultaneous transmission of Internet, e-mail, and other non-telephony data with the telephony voice data using the companion appliances; and

10) Incorporating video data in received and transmitted messages.

Feature Sets

A large number of feature sets can be implemented for a variety of voice and data communications using the companion information appliance **210** and the telephone **240**. Five of these feature sets are described briefly below for the following telephony operations: placing a phone call, receiving an incoming call, forwarding a call, conference calls, and transferring a call. User interface displays corresponding to the feature sets are described in the "User Interface Description" section of this application.

A first feature set enables the information appliance **210** user to direct the telephone **240** to place a phone call. For one embodiment, as the call is placed, information is transferred from the telephone **240** to the information appliance **210** so images are shown on the information appliance display to inform the user of call status information. The call status information includes dialing, idle, no dial tone, ringing, connected, busy, on hold, disconnected, hang-up, and re-dialing. The user can respond to the status information display and provide input through the information appliance **210** to cause the telephone **240** to re-dial, switch a connected line to on hold, switch calls (e.g., hold to active for line 1 and active to hold for line 2), place a phone call using an address book application data entry, or transmit a memo to the recipient of the phone call over a data network link.

The memo and the phone call can be transmitted simultaneously by the companion information appliance **210** and the telephone **240**, thus providing an example of the appliances capability to transmit both voice and data communications. User input to the information appliance **210** communicated to the telephone **240** can cause the telephone to provide connection time and date data, and/or connected time data for display on the information appliance display.

A second feature set enables the information appliance **210** user to respond to incoming calls received by the telephone **240**. After the telephone **240** detects an incoming call, the telephone transmits a message to the information appliance **210** indicating that a incoming call has been received. The information appliance **210** then provides the message information to the user on the display. The information provided can include the Caller ID and the Caller Name.

When establishing the telephone **240** operating parameters, the user can establish that any incoming call will result in an interrupt of other applications, or that an incoming call from a particular network connected device will interrupt the other applications. For one embodiment, where the PalmPilot, PalmProfessional or Palm III is the information appliance **210**, the incoming call interrupt can be provided through a "HotSync" key.

The user can respond to the incoming call information provided on the information appliance display by entering a response through the information appliance user interface. Such responses include answering the incoming call, rejecting the incoming call, canceling (or ignoring) the incoming call, and saving the incoming call (i.e., recording the alias address information corresponding to the incoming caller to the information appliance address database).

A third feature set enables the information appliance **210** user to put call forwarding (or call diversion) in effect for selected lines on the telephone **240**. The call forwarding automatically diverts all calls to the forwarded phone. The forwarding can be to more than one forwarded number, and can continue forwarding from another phone, i.e., go to a roaming mode. Other call diversion services include call forwarding busy, call forwarding unconditional, call forwarding no reply, and call deflection. In one embodiment these services are provided as defined by the ITU H.450 series standards.

A fourth feature set enables the information appliance **210** user to control the execution of conference calls through the telephone **240**. The user can add parties to the conference, hang-up on the conference, place connected conference lines on hold, switch from one conference to another, and provide white board type information to conference participants.

A fifth feature set enables the information appliance **210** user to control the transfer of a connected call.

Video Capability

Communications including video data can be exchanged with network connected devices and displayed using the combined information appliance **210** and the telephone **240**. If the telephone **240** has video display and capture capabilities, the information appliance **210** can be used as an interface to control the video aspects of the communication.

In one embodiment, the EMP **333** protocol is extended to support video capability exchanges and commands to control the video, such as turning the video on and off, panning, zooming, etc. For this embodiment, the information appliance **210** can be a palm-sized computer **343**. The user interface on the palm-sized computer **343** also provides support for such capabilities.

In another embodiment, the palm-sized computer **343** has sufficient computing and display capabilities; as well as sufficient communications capacity on the link (e.g., the EIA/TIA 232 serial port **323** link) between the palm-sized computer **343** and the telephone **240**; to decode and display the video on the palm-sized computer's **343** own screen, or a screen corresponding to a peripheral device connected to the palm-sized computer **343**. In this embodiment the palm-sized computer's **343** screen provides video display and user interface functionality.

The video camera for sending video information to network connected devices can reside either on the palm-sized computer **343** or the telephone **240**. For embodiments where the video camera resides on the palm-sized computer **343**, sufficient computing resources are provided in the palm-sized computer **343** to capture and encode the video (e.g., for transmission as an MPEG formatted data stream), and sufficient communications capacity between the palm-sized computer **343** and the telephone **240** to transfer the captured video information.

Data Link Layer Protocol Frame Format

The DLLP **328** frame **400** format is illustrated in FIG. **4**. The frame is the unit of transmission at the data link layer. The frame includes a header and/or a trailer, along with some number of units of data. Each box in the frame format represents four binary digits (bits), or a nibble. Basically, DLLP **328** serves the combined roles of the HDLC **326** and PPP **330** layers in the communications protocol communications protocol hierarchy with API/PPP/HDLC layers for telephony applications **300** with respect to the API **336** data. DLLP **328** encapsulates an applications layer EMP frame and provides error detection as described below.

The basic unit of encapsulation that is passed across the interface between the network layer and the data link layer is referred to as a packet. Each packet is typically mapped to a frame. The exceptions to this mapping are when data link layer fragmentation is being performed, or when multiple packets are incorporated in a single frame.

Frame delimiters are flag sequences that indicate the beginning or the end of a frame. The frame delimiters **410** used in the DLLP frame **400** format are hexadecimal 0x7E, or binary 01111110, like those used for Point to Point Protocol (PPP) packets.

A message, or EMP frame **420** having variable length is placed in the DLLP frame **400** after the left, or first, frame delimiter **410**. The EMP frame **420** can be according to any of the byte descriptions provided in Tables 1 through 5 below, or any other byte description that corresponds to information exchanged between the companion appliances.

Each transmitted DLLP frame **400** includes a numerical value calculated using a formula applied to the EMP frame **420**. The calculated value formatted for the DLLP frame **400**, as shown in FIG. **4**, is referred to as a frame check sequence (FCS), **430**. The FCS **430** is used as checksum for error-detection. The FCS **430** is similar to that used for PPP frames. After calculating the FCS **430** value, the bits are reversed (XOR-ed) and the least significant byte is sent out first. The receiving station then applies the same calculation to the EMP frame **420** and checks to make sure the accompanying numerical value of the FCS **430** is the same. If not, the receiver can assume that the message has been garbled, and proceed to ensure that the transmitting device resends the garbled frame.

For embodiments that use octet stuffing, a control escape octet is defined as hexadecimal 0x7D, or binary 01111101. Also similar to PPP frames, each 0x7D or 0x7E byte in the EMP frame **420** or FCS **430**, is replaced by a two byte sequence <0x7D><original byte XOR 0x7E>.

Ethernet Telephone Management Protocol Frame Descriptions

In this section, descriptions of EMP frame **420** formats are provided for the data types listed in each of the following tables. Tables 1–5 provide status titles for operations, information types for capabilities, byte descriptors, and comments for some of the different data types contemplated for the Ethernet Management Protocol.

Duration fields, conference numbers, line numbers, sequence numbers, and other number fields (shown as encapsulated by' 'in the frames shown in FIGS. **4**, and **5**A through **5**K) are formatted as integers. Telephone numbers, passwords, and the command/status are formatted as ASCII codes (shown with no encapsulation). For one embodiment, conference numbers are integers starting from 64, while line numbers start from 1.

Table 1 lists information types corresponding to the exchange of get/set capability information from a palm-sized computer **343** to the Ethernet telephone **310**. Table 2 lists information types corresponding to the exchange of send capability information from the Ethernet telephone **310** to a palm-sized computer **343**. Table 3 lists status types for data sent from a palm-sized computer **343** to the Ethernet telephone **310**. Table 4 lists status types for data sent from the Ethernet telephone **310** to a palm-sized computer **343**. Table 5 lists information types corresponding to error codes along with the action taken by one embodiment of the invention.

Different types of information describing some of the get/set capabilities and preferred settings for a palm-sized computer **343** that are exchanged with a connected Ethernet telephone **310** are shown in Table 1. The EMP frame **420** formats for some of the information types are described in the following paragraphs. Other information types corresponding to get/set capabilities not included in Table 1, will have similar formats to those discussed below, and will have differences according the nature of the information required to be exchanged.

The format of the EMP frame **420** for get/set capability types indicated by a superscript "1", in Table 1 is shown in FIG. **5**A, as reference number **501**. There are two byte descriptions in each of the EMP frames formatted according to FIG. **5**A for the get/set capabilities. The first byte description **503** is provided in the first two EMP frame **420** nibbles and typically corresponds to the get (AC) and set (AB) status byte descriptions shown in Table 3. The second byte description **505** is provided in the third and fourth EMP frame **420** nibbles and corresponds to the byte descriptions provided in Table 1. For example, the first four nibbles for a get number of lines exchange of capabilities from the palm-sized computer are ACC0.

TABLE 1

Capabilities Exchanged from Palm-sized Computer to Ethernet telephone Get/Set (Capabilities)

Type of Information	Byte Description	Comments	Get	Set
Number of lines	$C0^1$	To get/set number of lines to support. Currently, "Get" is sent to denote initialization.	✓	✓
Speaker	$C1^2$	To get/set speaker settings of Ethernet telephone.	✓	✓
Conference Call	$C2^1$	To get/set number of conferences to support.	✓	
Call Transfer	$C3^1$	To get/set transfer capability. ('0' for disable, '1' for enable.)	✓	✓
Call Forward	$C4^1$	To get/set forwarding capability. ('0' for disable, '1' for enable.)	✓	✓
Call Restriction	$C5^1$	To get/set user's restriction. (Integer '0' for local, '1' for unrestricted access. Internal calls are possible without the password.)	✓	✓
Phone Number	$C6^3$	For "Get": Gets the Ethernet telephone's 'ID' (phone no). When this packet is sent, phone assumes its own 'ID'. For "Set": Sets the Palm-sized computer's 'ID' on the phone.	✓	✓
Password	$C7^4$	Sent before some of the settings are possible. Integer '00' for Ethernet telephone password, and '01' for Palm-sized computer's, placed before the ASCII formatted password. "Get": Used for sending original password. "Set": Used for setting new password.	✓	✓
Ringer Volume	$C8^2$	To get/set ringer volume of Ethernet telephone.	✓	✓
Handset Volume	$C9^2$	To get/set handset volume of Ethernet telephone.	✓	✓

The sequence number (Seq. no.) **507** corresponds to the placement order of a frame in a sequence of frames used for a particular operation, such as dialing by the palm-sized computer **343**. A detailed discussion of the frames

exchanged in a palm-sized computer **343** dialing operation is provided in conjunction with the dialing state diagram shown in FIG. **6** below. The sequence number **507** is placed as the fifth and sixth nibbles in the get/set capability type **501** EMP frame **420** format.

The EMP frame **420** format for the number of lines (C0) and conference call (C2) capability types includes the number of lines that the user is allowed or that the Ethernet telephone **310** supports, as a one byte integer. These numbers are placed in the numbers ("Nos.") field **508** of the get/set capability type **501** EMP frame **420** format.

The format of the EMP frame **420** for capability types indicated by a superscript "2", in Table 1 is similar to the format for the capability types indicated by a superscript "1". The key difference is that the numbers [i.e., data placed in the numbers ("Nos.") portion **508**] are in ASCII code format. A character of"0" means that the speaker, ringer, or handset is off, or the volume adjustment capability is not supported by the Ethernet telephone **310**. A character indicating a value in the range of "1"through "9" sets the speaker, ringer, or handset volume.

The format of the EMP frame **420** for capability types indicated by a superscript "3", in Table 1 is similar to the format for the capability types indicated by a superscript "1". The key differences are that the integers representing the phone number in the "Nos." field **508** are in ASCII code format, and the phone number field can be of varying length. Note that for the purposes of the "Nos" field **508**, phone numbers are restricted to numerals.

When the "Get" indication is provided for the phone number (C6) information type, the Ethernet telephone **310** will assume its own identification. When this occurs, a single Ethernet telephone **310** can act as the network appliance companion for a plurality of palm-sized computers **343**. When the "Set" indication is provided, the palm-sized computer's **343** is used by the Ethernet telephone **310**, enabling others to contact the palm-sized computer user using the same identification, or phone number, without concern for the user's actual location, i.e., analogous to a roaming cellular phone.

The format of the EMP frame **420** for get/set capability types indicated by a superscript "4", in Table 1 is shown in FIG. **5**B, as reference number **509**. The FIG. **5**B EMP frame **509** format is particularly well-suited for the password capability type (C7) and includes the first byte description **503**, followed by the second byte description **505**, and the sequence number. After the sequence number **507**, an integer flag **510** is provided to indicate which password is included in the FIG. **5**B EMP frame **509**. An integer flag **510** value of "00" indicates that the Ethernet telephone **310** password is included and a value of "01" indicates that the palm-sized computer **343** password is included. The password is then provided in the password field **511** as a variable length alpha-numeral. After the password field **511**, the last field in the FIG. **5**B EMP frame **509** is an end of transmission indication **512**, shown in FIG. **5**B as "00".

Different types of information describing the some of the send capabilities and preferred settings for an Ethernet telephone **310** that are exchanged with a connected palm-sized computer **343** are shown in Table 2. The EMP frame **420** formats for the information types in Table 2 are indicated by the superscripts following the byte descriptions. The formats are essentially the same as for the information types having the same superscripts in Table 1 above.

Some of the different types of data transmitted from a palm-sized computer **343** to an Ethernet telephone **310** for one embodiment of the invention are listed in Table 3. The EMP frame **420** formats for these types of data are discussed in the following paragraphs. Other data types transmitted from a palm-sized computer **343** to an Ethernet telephone **310** not included in Table 3, will have similar formats to those discussed below, and will have differences therefrom according the nature of the transmitted data. Many of the formats that are used for transmitting data from the palm-sized computer **343** to the Ethernet telephone **310**, are also used for transmitting similar data from the Ethernet telephone **310** to the palm-sized computer **343**. These data types are referred to herein as palm/phone transmitted EMP data types.

The format of the EMP frame **420** for first types of data transmitted from a palm-sized computer **343** to an Ethernet telephone **310** indicated by a superscript[1], in Table 3 is shown in FIG. **5**C, as reference number **513**. The first palm/phone transmitted EMP data type **513** format includes a second byte description **505**, followed by a sequence number **507**, and a line number **514**. Status types having the first palm/phone transmitted EMP data type **513** format include new line/query (A0), hang-up (A2), switch calls (A3), answer (A4), and put on hold (A5).

The second palm/phone transmitted EMP data type **515** format, shown in FIG. **5**D, is indicated by a superscript[2], in Table 3 and includes a second byte description **505**, followed by a sequence number **507**, a line number **514**, a telephone number **517** having variable length, and an end of transmission indication **512**. The telephone number **517** field supports the use of numerals and the "#", "*", and "!" characters. The second palm/phone transmitted EMP data type **515** format is used for dialing numbers (A1).

The third palm/phone transmitted EMP data type **521** format, shown in FIG. **5**E, is indicated by a superscript[3], in Table 3 and includes a second byte description **505**, followed by a sequence number **507**, one or more conference numbers in a conference number **523** field, one or more line numbers **514**, and an end of transmission indication **512**. This format is used to split a conference (A8). The first numbers listed in the conference number **523** field are the new conference numbers if another conference is started. The numbers that follow the new conference numbers in the conference number **523** field are the line numbers affected by the split.

TABLE 2

Capabilities exchanged from Ethernet telephone to Palm-sized Computer
Send Capabilities

Type of Information	Byte Description	Bytes Following
No. of lines	30[1]	1 byte integers to give number of lines supported
Speaker	31[2]	Character "0" to indicate speaker off, "1"–"9" to indicate speaker volume. ASCII Code NAK (0x15) means no speaker on phone.
Conference Call	32[1]	1 byte integer to give number of conferences supported.
Call Transfer	33[1]	To indicate if such feature is supported. (Integer '0' indicates non-support, '1' for supported.)
Call Forward	34[1]	To indicate if such capability is supported. (Integer '0' for non-support, '1' for supported, '2' for enabled.)

TABLE 2-continued

Capabilities exchanged from Ethernet telephone to Palm-sized Computer
Send Capabilities

Type of Information	Byte Description	Bytes Following
Call Restriction	35[1]	User's restriction of calls (Integer '0' for local, '1' for unrestricted access. Internal calls are possible without the password.)
Phone No	36[3]	For sending the Ethernet telephone's identity to Palm-sized computer.
Password	37[3]	For password requests (Integer '0' indicates request, '1' indicates password error, '2' indicates password is correct.)
Ringer Volume	38[1]	Character "0"–"9" to indicate ringer volume.
Handset Volume	39[1]	Character "0"–"9" to indicate handset volume.

TABLE 3

Protocol for Data Sent from Palm-sized Computer to Ethernet telephone
Palm-sized Computer → Ethernet-Phone

Status	Byte Description	Comments
New Line/Query Status	A0[1]	To call another party (phone should put the current active line, if any, on hold.)
Dial	A1[2]	Supports numerals, "#", "*", "!".
Hang-up	A2[1]	
Switch calls/line	A3[1]	
Answer	A4[1]	
Put on hold	A5[1]	
Off-Hook	A6	
Conference Call	A7[3]	The numbers that follow the first (new conference number) number can be conference numbers (to combine conferences) or simply line numbers.
Split Conference	A8[3]	Split the stated lines from the current conference. Fields that follow include (in the indicated order) new conference numbers (if starting another conference) and line numbers affected.
Call Transfer	A9[4]	Two line numbers should be present in the packets transferred. The line to be transferred precedes.
Call Forwarding	AA[5]	Negative acknowledgement ASCII code NAK (0x15) signifies disabling the feature.
Set	AB[6]	See Table 1 for more information.
Get	AC[6]	See Table 1 for more information.
Conference Phone Nos.	AD	Getting phone numbers of other parties involved in the conference.
Data	AE	Data transfer
Dual Tone Multi-Frequency (DTMF)	AF[7]	
EOT	00	To signify end of message/transmission.
Reset	F0	Effectively resets the phone.
Acknowledge	F1[8]	For error control. Packets ID from 1–255(1 byte). Time-out set at approximately 3 seconds.
Reject	F2[9]	For error control. See Table 5 for descriptions of the codes.
Escape	FF	

The fourth palm/phone transmitted EMP data type **525** format, shown in FIG. **5**F, is indicated by a superscript[4], in Table 3 and includes a second byte description **505**, followed by a sequence number **507**, a first line number **527**, and a second line number **529**.

The fifth palm/phone transmitted EMP data type **531** format, shown in FIG. **5**G, is indicated by a superscript[5], in Table 3 and includes a second byte description **505**, followed by a sequence number **507**, one or more telephone numbers **517**, and the end of transmission (EOT) indication 00, indicated in FIG. **5**G as reference number **512**.

The sixth palm/phone transmitted EMP data type is indicated by a superscript[6], in Table 3 and has the format described above for FIG. **5**A, i.e. reference number **501**.

The seventh palm/phone transmitted EMP data type **533** format, shown in FIG. **5**H, is indicated by a superscript[7], in Table 3 and includes a second byte description **505**, followed by a sequence number **507**, a digit pressed **535**, and a duration **537**. The digit pressed **535** is a one-byte value, and the duration **537** is a two byte hexadecimal number corresponding to the duration in milliseconds.

The eighth palm transmitted EMP data type **539** format, shown in FIG. **5**I, is indicated by a superscript[8], in Table 3 and includes a second byte description **505**, followed by a sequence number **507**.

The ninth palm/phone transmitted EMP data type is indicated by a superscript[9], in Table 3 and has the format described below in conjunction with FIG. **5**K and Table 5. This data type is also referred to as the EMP error code data type **547**.

The format of the EMP frame **420** for first palm/phone transmitted EMP data type **513** transmitted from an Ethernet telephone **310** to a palm-sized computer **343** is indicated by a superscript[1] in Table 4, and is shown in FIG. **5**C. As discussed above, the EMP frames **420** for the first palm/phone transmitted EMP data type **513** have the same format whether the frame is transmitted from the Ethernet telephone **310** or from the palm-sized computer **343**. The key differences associated with source of the message are the byte descriptions and the sequence numbers. The formats for the EMP frames **420** for the second through ninth phone transmitted EMP data types, indicated by superscripts[2-9] in Table 4, are also the same as for the palm/phone transmitted EMP data types having the same superscript designations in Table 3, above.

TABLE 4

Protocol for Data Sent from Ethernet telephone to Palm-Sized Computer
Ethernet-Phone → Palm-Sized Computer

Status	Byte Description	Comments
New Line	10[1]	
Incoming Call	11[2]	
Idle	12[1]	
No Dial Tone	13[1]	
Ringing	14[1]	
Connected	15[1]	
Busy	16[1]	
Speaker Answer	17	Answering using Speaker. The palm-sized computer program assumes this automatically unless the "Using Handset" frame is sent.
Using Handset	18	Answering using the handset.
On Hold	19[1]	
Disconnected	1A[1]	
Hang-up	1B[1]	
Re-dial	1C[1]	

TABLE 4-continued

Protocol for Data Sent from Ethernet telephone to Palm-Sized Computer
Ethernet-Phone → Palm-Sized Computer

Status	Byte Description	Comments
Conference Call	$1D^{3}$	Procedure is same as for present PBX phone, i.e. multiple conferences are set up individually.
Incoming Caller ID	$1E^{10}$	Unknown number represented by ASCII character NAK (0x15). Line number precedes telephone number in the packet. Caller Name follows the telephone number.
Call Transfer	$1F^{4}$	ASCII code NAK (0x15) indicates canceling. If only one line number is sent, it indicates that the user had pressed the transfer button on the phone - this is to enable the palm-sized computer to go to the transfer screen.
Call Forward	20^{5}	ASCII code NAK (0x15) indicates disabling. If no telephone number is sent, it indicates that the user had pressed the forward button on the phone - this is to enable the palm-sized computer to go to the forward screen.
Send	21^{6}	See Table 2 for more information
Phone No.	22^{5}	Sends telephone numbers of other lines in the conference to the Palm-Sized Computer.
Data	23	Data Transfer.
DTMF	24^{7}	
Switch Line	25^{1}	To change to another line.
EOT	00	End of message.
Acknowledge	$F1^{8}$	For error control. Each packet has an ID in the range from 1 to 127.
Reject	$F2^{9}$	For error control. See Table 5 for descriptions for the codes.
Reset	F0	
Escape	FF	

TABLE 5

Error codes
Error Specifications

Type of Information	Byte Description	Current Action Implemented On Palm-Sized Computer
Default/Unknown	90	Warning Message to User
Server not ready	91	Warning Message to User
Capability not supported	92	Ignored
Packet/Bit Error	93	Re-send Packet (and those with Sequence Number after it)
Repeated/Duplicate Message	94	Check if message was already acknowledged
Repeated/Duplicate Acknowledgment	95	Ignored
Password not set yet	96	Ignored
Line/Conference already in use	97	Warning Message - Recommend Reset of Palm-Sized Computer
Missing Sequence No	98	Re-send Packet (and those with Sequence Number after it)

The format of the EMP frame **420** for the tenth phone transmitted EMP data type is indicated by a superscript[10] in Table 4, and is shown in FIG. **5**J, as reference number **541**. The tenth phone transmitted EMP data type **541** includes a second byte description **505**, followed by a sequence number **507**, a telephone numbers **517** field, an end of transmission field **512**, a caller name field **543**, and a second end of transmission field **545**. If the caller name is not available, the caller name field will have a zero length, but the "EOT" will still be present, i.e. two "EOT"s will be transmitted next to each other.

The format of the EMP error code data type **547** used for the error messages listed in Table 5 is shown in FIG. **5**K. The error code data is typically transmitted from the Ethernet telephone **310** to the palm-sized computer **343**. The EMP error code data type **547** format includes a second byte description **505**, followed by an error code **549** and an error sequence number **551**.

Interactions Between the Telephone and the Information Appliance

Many types of communications between the telephone **240** and the information appliance **210** are possible. Using the EMP frame **420** formats described above, state diagrams describing four types of exchanges between the telephone **240** and the information appliance **210** are provided in FIGS. **6** through **9**. Each of these state diagrams is discussed in detail below.

As shown in the dialing state diagram **600**, the conferencing state diagram **700**, the call forwarding state diagram **800** and the call transfer state diagram **900** below, the packets, formatted according to the EMP and DLLP **328** protocols, are exchanged though a communications port that connects the telephone **240** and the information appliance **210**. In FIGS. **6** through **9**, the communications port is shown as a serial EIA/TIA-232 port, the information appliance **210** is shown as a palm-sized computer (or PSC) **343**, and the telephone **240** is shown as an Ethernet telephone **310**. In the descriptions below, the two character values in parenthesis after each message provide the corresponding byte description **503** corresponding to the message. The palm-sized computer **343** sequence numbers for the exchange begin with '01', while the Ethernet telephone **310** sequence numbers begin with '50'.

Palm-Sized Computer Dialing

FIG. **6** provides a state diagram illustrating the DLLP **328** formatted packets of data exchanged between the telephone **240** and the information appliance **210** when the information appliance **210** dials a phone number to place a call through the telephone **240** to a device connected to the LAN link **250**.

The dialing session begins when the user starts the dialing program **602**. The steps required to start the dialing program **602** are discussed in detail below in conjunction with FIGS. 10–13. The palm-sized computer **343** sends a new line/status query (A0) message for the first line **604** formatted according to FIG. **5**C to the Ethernet telephone **310**. The new line/status query (A0) message for the first line **604** requests access to line 01 on the Ethernet telephone **310** for communication with a network connected device.

The Ethernet telephone **310** checks the status of the first 01 line **606** and returns an idle (12) message **608** to the palm-sized computer **343** indicating that line 01 is available for dialing, or other telephony operations controllable by the palm-sized computer **343**. The idle (12) frame body is formatted according to FIG. **5**C.

Note that an Ethernet telephone **310** acknowledgement (F1) message **610** indicating receipt by the Ethernet telephone **310** of the previous packet from palm-sized computer is sent in the same data stream as the idle message. The simultaneous acknowledgement is well known in the art as "piggybacking" and reduces the number of message

exchanges required for a particular telephony operation. The acknowledgement packet is formatted according to FIG. **5**I.

As described in greater detail below in conjunction with FIGS. **13** and **14**, the palm-sized computer **343** displays the idle status **612** of line 01 on the palm-sized computer **343** screen, e.g., as "not in use". The palm-sized computer **343** then sends a dial (A1) message **614** to the Ethernet telephone **310**. The dial message **614** is formatted according to FIG. **5**D, and provides a command to the Ethernet telephone **310** to dial the number corresponding to the desired network connected device, e.g., **1234**. Note that a simultaneous palm-sized computer **343** acknowledgement (F1) message **616** indicating receipt by the palm-sized computer **343** of the previous frame from the Ethernet telephone **310** is sent in the same data stream as the dial message.

The Ethernet telephone **310** acknowledgement messages **610** and the palm-sized computer **343** acknowledgement messages **616** are typically sent in a piggyback fashion. However, there are certain exceptions that will be noted herein where the acknowledgement messages are sent by themselves.

In response to the dial message **614**, the Ethernet telephone **310** dials the "1234" number **618** indicated therein, i.e., in the telephone numbers **517** field of the second palm/phone-transmitted EMP data type **515** as shown in FIG. **5**D, corresponding to the network connected device. Also in the Ethernet telephone **310** dials the "**1234**" number **618** block, the Ethernet telephone **310** detects that the network-connected device is ringing and then sends a line 01 ringing (14) message **620** to the palm-sized computer **343**. The line 01 ringing (14) message **620** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. Upon receipt of the line 01 ringing (14) message by the palm-sized computer **343**, an indication of the ringing status is provided **622** on the display.

When the network connected device **1234** picks up **624** its phone receiver, or otherwise indicates that the communication can begin, the Ethernet telephone **310** sends a line 01 connected (15) message **626** to the palm-sized computer **343** for network-connected device **1234**. The line 01 connected (15) message **626** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. Upon receipt of the line 01 connected (15) message **626** by the palm-sized computer **343**, an indication of the connected status of line 01 is provided **628** on the display.

Note that two uncoupled palm-sized computer acknowledgement messages **616** are sent back to the Ethernet telephone **310**. These uncoupled palm-sized computer acknowledgement messages **616** are sent because there is no intervening "dialing" protocol message from the palm-sized computer **343** upon which they can readily piggyback before the next message is sent by the Ethernet telephone **310**.

The Ethernet telephone **310** receives an incoming call **630** for the user on line 02 from a network-connected device having a number 5678. The Ethernet telephone **310** responds to the incoming call by sending an incoming call (11) message **632**. The incoming call (11) message **632** is formatted according to the second palm/phone-transmitted EMP data type **515** as shown in FIG. **5**D. The number of the network connected device making the call, 5678, is indicated by the telephone number **517** portion of the second palm/phone-transmitted EMP data type **515**.

Upon receipt of the incoming call (11) message **632** by the palm-sized computer **343**, an indication of the incoming call on line 02 from the network connected device having the 5678 number is provided **634** on the display. The palm-sized computer **343** also sends an answer (A4) message **636** to the Ethernet telephone **310**. The answer (A4) message **636** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. The Ethernet telephone **310** responds to the answer (A4) message **636** by connecting **638** the 5678 network connected device to line 02. The Ethernet telephone **310** response to the answer (A4) message **636** is set to place line 01 on hold before switching to line 02, because for this embodiment only one Ethernet telephone **310** line can be active at a particular time.

The Ethernet telephone **310** then sends a line 02 connected (15) message **640** to the palm-sized computer **343**. The line 02 connected (15) message **640** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. Upon receipt of the line 02 connected (15) message **640** by the palm-sized computer **343**, an indication of line 02 connected status and the line 01 on hold status are provided **642** on the display.

After completing the conversation or other communication on line 02, the user hangs-up **643** line 02. The palm-sized computer **343** then sends a line 02 hang-up (A2) message **644** to the Ethernet telephone **310**. The hang-up (A2) message **644** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. In response to the line 02 hang up (A2) message, the Ethernet telephone **310** disconnects **646** the network connected device having the number **5678** from line 02.

The Ethernet telephone **310**, then sends a line 02 disconnected (1A) message **648** to the palm-sized computer **343**. The line 02 disconnected (1A) message **648** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C.

The palm-sized computer **343** the switches **650** to line 01 and sends a switch calls/line (A3) message **652** to the Ethernet telephone **310**. The switch calls/line (A3) message **652** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. The Ethernet telephone **310** the connects **654** line 01, and sends a line 01 connected (15) message **656** to the palm-sized computer **343**. The line 01 connected (15) message **656** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. Upon receipt of the line 01 connected (15) message **656** by the palm-sized computer **343**, an indication of line 01 connected status is provided **658** on the display.

After completing the conversation or other communication on line 01, the user hangs-up **660** line 01The palm-sized computer **343** then sends a line 01 hang-up (A2) message **662** to the Ethernet telephone **310**. The hang-up (A2) message **662** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. In response to the line 01 hang up (A2) message **662**, the Ethernet telephone **310** disconnects **664** the network connected device having the number **1234** from line 01.

The Ethernet telephone **310**, then sends a line 01 disconnected (1A) message **666** to the palm-sized computer **343**. The line 02 disconnected (1A) message **648** is formatted according to the first palm/phone transmitted EMP data type **513** as shown in FIG. **5**C. Then the user ends the "dialing" program **668**.

Conferencing

FIG. **7** provides a state diagram illustrating the DLLP **328** formatted packets of data exchanged between the Ethernet telephone **310** and the palm-sized computer **343** when the palm-sized computer **343** requests the Ethernet telephone

310 to connect two lines to a conference call. First, the two Ethernet telephone 310 lines are connected to LAN link 250 connected devices.

The conferencing session begins when the user starts the conferencing program 705. The steps required to start the conferencing program 705 are described in detail below in conjunction with FIGS. 16, 26, 34, and 35. The palm-sized computer 343 and the Ethernet telephone 310 then proceed to exchange the same messages as described above for the dialing state diagram 600 including: the new line/Status Query (A0) message for the first line 604, the idle (12) message 608, the dial (A1) message 614, the ringing (14) message 620, and the connected (15) message 626, along with the accompanying Ethernet telephone 310 acknowledgement (F1) messages 610 and the palm-sized computer 343 acknowledgement (F1) messages 616.

The user then starts a second call 710 for the conference by sending a second call new line/Status Query (A0) message 715 formatted according to FIG. 5C to the Ethernet telephone 310. The new line/status query (A0) message 710 requests access to line 02 on the Ethernet telephone 310 for communication with a network connected device.

The Ethernet telephone 310 puts the first 01 line on hold and checks the status of second 02 line 720. Upon finding that line 02 is idle, the Ethernet telephone 310 transmits a line 02 idle (12) message 725 formatted according to FIG. 5C. The palm-sized computer 343 then provides information on the screen informing the user that line 01 is on hold and that line 02 is idle 730. The user then inputs (dials) the number "5678"732 to place a call to the corresponding network connected device on line 02. The palm-sized computer 343 then sends a dial message (A1) corresponding to network connected device "5678" 734 formatted according to FIG. 5D.

The Ethernet telephone 310 dials the 5678 number on the second 02 line 738 and detects a ring tone from the 5678 device. The Ethernet telephone 310 then sends a line 02 ringing (14) message 740 formatted according to FIG. 5C to the palm-sized computer 343. The palm-sized computer 343 then displays the line 02 status as ringing 742.

When the network connected device 5678 picks up its phone receiver, or otherwise connects the second 02 line 744 for communication, the Ethernet telephone 310 sends a line 02 connected (15) message 746. formatted according to FIG. 5C. The palm-sized computer 343 then displays the connected status of second 02 line 748.

The user then initiates the conference 749 by tapping either the conference prompt (shown in FIG. 16 as reference number 1650) or the "Conference" option (shown in FIG. 26 as reference number 2640). The palm-sized computer 343 then sends a PSC conference call (A7) message 750 formatted according to FIG. 5E to the Ethernet telephone 310. Note that the first conference number 523 as previously discussed is provided a number of "64" that as shown in both the PSC conference call message 750, and the Ethernet telephone 310 (ET) conference call message 754 discussed below.

The Ethernet telephone 310 then starts the conference 752 and sends an ET conference call (1D) message 754 formatted according to FIG. 5E to the palm-sized computer 343.

The conference call communications progress 756 under the control of the palm-sized computer 343 until the user hangs up both the first 01 line and the second 02 line 758. The palm-sized computer 343 then sends a line 01 hang-up (A2) message 760 and a line 02 hang-up (A2) message 762 to the Ethernet telephone 310. Both of the hang-up messages are formatted according to FIG. 5C.

The Ethernet telephone 310 then disconnects both the first 01 and the second 02 lines 764, and sends a first 01 line disconnected (1A) message 766 and second 02 line disconnected message 768 to the palm-sized computer 343, and the conferencing program is thereby ended 770. Note that the hang-up messages are sent to the Ethernet telephone 310 in response to the user tapping the end conference button 3420.

Call Forwarding

FIG. 8 provides a state diagram illustrating the DLLP 328 formatted packets of data exchanged between the Ethernet telephone 310 and the palm-sized computer 343 when the palm-sized computer 343 requests the Ethernet telephone 310 to forward a call to another device connected to the LAN link 250.

The forwarding begins when the user starts the forwarding program 805. The steps required for the user to start the forwarding program 805 are described in detail below in conjunction with FIGS. 16, 21, 22, and 26. As was done for the previous two exchanges the program begins with the palm-sized computer 343 sending a new line/Status Query (A0) message for the first line 604 to the Ethernet telephone 310. As before for the dialing and conferencing sequences, the Ethernet telephone 310 checks the status of the first 01 line 606 and we assume, for the purposes of this discussion, that line 01 is idle. Therefore, the Ethernet telephone 310 responds by sending an idle (12) message 608 to the palm-sized computer 343.

The user then keys in the number "9876" for forwarding 810 in the forward incoming calls line shown in FIG. 21 as reference number 2120. The palm-sized computer 343 then sends a call forwarding (AA) message 812 formatted according to FIG. 5G to the Ethernet telephone 310.

The Ethernet telephone 310 then forwards all calls to the "9876" device 814 through line 01, and sends a call forward (20) message 816 formatted according to FIG. 5G to the palm-sized computer 343. The palm-sized computer 343 then displays the call forwarded status 818 as shown in FIG. 22 including the currently forwarded line 2210.

Call Transfer

FIG. 9 provides a state diagram illustrating the DLLP 328 formatted packets of data exchanged between the telephone 240 and the information appliance 210 when the information appliance 210 requests the telephone 240 to transfer a call to another device connected to the LAN link 250.

The transfer begins when the user starts the transfer program 902 to transfer a call connected to line 01 to line 02. The steps required for the user to start the transfer program 902 are described in detail below in conjunction with FIGS. 16 through 19. As indicated in FIG. 9, line 01 is already connected when the user starts the transfer program 902.

The palm-sized computer 343 sends a new line/Status Query (A0) message for the second line 904 formatted according to FIG. 5C to the Ethernet telephone 310. The Ethernet telephone 310 checks the status of the second line 906 and finds that line 02 is idle. The Ethernet telephone 310 then sends a second line Idle (12) message 907 formatted according to FIG. 5C.

The user then keys in "5432" 908 as the number to which the call connected on line 01 will be transferred. The palm-sized computer 343 then sends a dial (A1) "5432" message 910 formatted according to FIG. 5D to the Ethernet telephone 310. The Ethernet telephone 310 then connects line 02 to the "5432" device 912 and sends a "5432" connected (15) to line 02 message 914 formatted according to FIG. 5C to the palm-sized computer 343.

The palm-sized computer **343** displays the connected status **916** and the user selects the "5432" device for transferring the call on line 01. The palm-sized computer **343**, then sends a PSC call transfer (A9) message from the first 01 line to the second 02 line **918** formatted according to FIG. **5**F to the Ethernet telephone **310**.

The Ethernet telephone **310** then transfers the call from line 01 to the 5432 device **920**, and sends an ET call transfer (1F) message **922** formatted according to FIG. **5**F to the palm-sized computer **343**. The palm-sized computer then displays the transferred status of the call **924**.

The Portable Computer

The fourth aspect of the invention provides a portable computer **320** adapted for connection to a telephone **240**. The portable computer **320** comprises a port for connecting to the telephone **240**, a memory storing user information corresponding to a user, and processing resources adapted to exchange data with the telephone. The telephone **240** has capabilities. The data includes the user information and data corresponding to the telephone **240** capabilities. The exchange of the data enables the portable computer **320** to discover capabilities of the telephone **240**, provide the user information to the telephone, and establish telephone operating parameters for telephone communications with devices connected to the telephone based on the user information and the telephone capabilities.

In some embodiments, the telephone **240** capabilities comprise network communication capabilities, and portable computer **320** companion capabilities.

In some embodiments, the portable computer **320** includes a display **1005** providing user interface **220** graphic elements corresponding to data exchanged with the telephone **240**, and a user interface enabling the user to input data supplementing the user information provided to the telephone.

In some embodiments, the user information comprises an identification corresponding to the portable computer **320**, user characteristics; and user access parameters.

In some embodiments, the portable computer **320** comprises a palm-sized computer **343**.

In some embodiments, the data exchanged with the telephone **240** corresponding to portable computer **320** control of the execution of the telephony programs, the telephone capabilities, and the user information are formatted according to an applications layer protocol. The applications layer protocol having frame formats for telephony functions.

In some embodiments, the portable computer **320** is adapted to provide data processing and user interface **220** functions without connection to the telephone **240**.

In some embodiments, the portable computer **320** includes processing resources for Internet access. For some of these embodiments, the processing resources for Internet access include Internet applications **367**, transmission control **359** software, and Internet protocol **356** software. For some of these embodiments, the processing resources for Internet access include Internet display applications **375** and display/user input transfer **373** software.

In some embodiments, the portable computer includes processing resources for user interface **220** support of video data. For some of these embodiments, the processing resources for user interface **220** support of video data include video data decoding, and video display. For some of these embodiments, the processing resources for user interface **220** support of video data, video data decoding, video display, and video camera image data.

For some embodiments, the exchange of the data enables the portable computer **320** to control execution of telephony programs.

For some embodiments, the data exchanged with the telephone **240** includes data corresponding to portable computer control of telephony programs, and data corresponding to the status of the devices connected to the telephone.

For some embodiments, the telephone **240** has an identification, and the identification corresponding to the portable computer **320** is presented in place of the telephone identification to devices connected to and communicating with the telephone **240**.

User Interface Graphical Elements and User Inputs

A user interface provided by the information appliance **210** enables users to control various telephone **240** features. These features include conferencing, transfer, and forwarding (diversion). A variety of screens provided for one embodiment of the invention including a palm-sized computer **343** and an Ethernet telephone **310** are described below. The screens provide user interface graphic elements that can be manipulated by the user through actions such as providing key strokes, tapping, clicking a mouse or comparable device, or providing voice entered data or commands. Similar screens could be provided on any other information appliance **210** display. The information could also be provided through user interface audio elements depending on the user's preferences.

As shown in FIG. **10**, the palm-sized computer **343** displays user interface graphic elements on a display **1005**. In the following user interface description, the displays provided at particular moments during a telephony operation are referred to as "screens".

When the user starts the telephone **240** control program on the palm-sized computer **343**, the first screen that greets the user on the palm-sized computer display is the initialization in progress screen **1000** as shown in FIG. **10**. During initialization, the palm-sized computer **343** queries the Ethernet telephone **310** and determines the Ethernet telephone's capabilities. The palm-sized computer **343** then establishes telephone **240** operating parameters based on the received capabilities. During initialization, information such as the number of lines and conferences supported, speaker, ringer and handset volume, transfer and forward feature support, Ethernet telephone **310** and Palm-sized computer **343** IDs, call restrictions for the user, and line status are exchanged.

The "Initializing" box **1010** disappears after the initialization is complete. The "New Line" button **1020** is also shown on the initialization in progress screen **1000**. The initialization completed screen **1100** is shown in FIG. **11**. The following description of the "New Line" button **1020** and placing a call feature includes references to messages described in the palm-sized computer **343** dialing state diagram section above.

When the user taps the "New Line" button **1020** on the initialization completed screen **1100**, the palm-sized computer **343** sends a new line/status query message (such as the new line/status query message for the first 01 line **604**) to the Ethernet telephone **310**. If line 01 is idle, the Ethernet telephone **310** sends a line 01 idle message **608** to the palm-sized computer **343** which causes the display of the call placement screen **1200**, including "not in use" at the status prompt **1220**.

The user can either enter the phone number, or network address alias, by using a stylus in conjunction with a handwriting recognition application, such as "Graffiti" or by

using the soft keypad after tapping on the "KeyPad" button **1210** so the keypad **1310** appears on the user dialing screen **1300** as shown in FIG. **13**. The user can also provide the network address alias through the user address database in the palm-sized computer **343**.

When the user taps the "Dial" button **1320** after entering the phone number **1330** in the "Tel. No." line **1340**, the palm-sized computer **343** sends a dial message (such as the dial message corresponding to the **1234** network device **614**). Note that alphabet entries in the "Tel. No." line **1340** will be converted by the palm-sized computer **343** to numbers before the dial message is sent to the Ethernet telephone **310**.

If calls were made using the palm-sized computer **343** telephone application before, the "Last No." button (shown in FIG. **28** as reference number **2820**) would also appear. In that case, the user can also dial the last number by tapping the "Last No." button **2820**. Tapping the "Cancel" button **1345** in the user dialing screen **1300** returns the user to the call placement screen **1200**.

A call status icon **1350** and a sound interface icon **1360** are disposed at the top right corner of the user dialing screen **1300**. The call status icon **1350** indicates whether the call has been hung up. A "hang up" message will also be indicated at the status prompt **1220**. The sound interface icon **1360** indicates whether the user is using a handset or a speaker.

In response to the dial message, the Ethernet telephone **310** dials the indicated number and sends a ringing message (such as the line 01 ringing message **620**) to the palm-sized computer **343**. The telephone **240** dialing screen **1400**, shown in FIG. **14**, appears after the "Dial" button **1320** or the "Last No." button **2820** is pressed by the user. After receiving the ringing message, the "ringing" status is provided on the display **1005**, as shown in FIG. **14**.

When the other party answers, the Ethernet telephone **310** sends a line connected message (such as the line 01 connected message **626** for the **1234** network connected device) to the palm-sized computer **343**, which responds by displaying the connected line screen **1500**, shown in FIG. **15**. The "Connection Time" line **1510** shows the time connected for the particular call, while the "Date/Time Connected" line **1520** shows the date and time that the call was connected. In addition, the user can write a short message down in the "Memo" field **1530** by means of "Graffiti" or by using a soft keyboard. This message can be retrieved when the user views the "Call History" details window (reference number **2410**, in FIG. **24**).

In the connected line screen **1500**, the user can choose to place another call (thereby placing the current one on hold naturally) by tapping the "New Line" button **1540**. The "Hold" button **1550** and the "Hang Up" button **1560** hold and hang up the call respectively. If there are two or more calls active at the same time, the "Switch" button **1570** will also appear on the connected line screen **1500** so that the user can switch from one active call to another active call.

The line pop-up menu button **1580** is disposed at the top right comer of the connected line screen **1500**, and appears as a downward arrow next to and to the left of "Line". When the line pop-up menu button **1580** is tapped, a line pop-up menu **1610** list of choices appears, as shown in the connected line screen with line pop-up menu window **1600**, shown in FIG. **16**. User selection of the "transfer" prompt **1620** will result in the display of the transfer dialog screen **1700**, shown in FIG. **17**.

The transfer feature is described here while the other features are explained later in the document. In the transfer dialog screen **1700**, the user is prompted to select the number corresponding to the line(s) to which the call is to be transferred in the "transfer to" prompts **1710**. Alternatively, the line to which the call is to be transferred can be dialed by entering the entire phone number in the "transfer to" telephone number field **1720**. The following description of the transfer feature user interface element interaction includes references to messages described in the call transfer state diagram section above.

If the user clicks the transfer dialog "OK" button **1730**, the appropriate lines are connected and placed on hold per the dial message sent from the palm-sized computer **343** to the Ethernet telephone **310** (such as the dial **5432** message **910**). When the appropriate line is connected to effect the transfer, the Ethernet telephone **310** sends a connected to message (such as the connected to second 02 line message **914**). Upon receipt of the connected to message, the palm-sized computer **343** displays the transfer waiting screen **1800**, shown in FIG. **18** with the status lines **1370** in the transfer waiting screen **1800** indicating that both line 1 and line 2 are waiting for their respective transfer to be completed. The palm-sized computer **343** then sends a call transfer message (such as the PSC **343** call transfer message from the first 01 line to the second 02 line **918**) to the Ethernet telephone **310**.

Once the transfer is complete, the Ethernet telephone **310** sends an ET call transfer message **922** to the palm-sized computer **343** and the transfer status display indication is updated. This is seen in the transfer completed screen **1900**, shown in FIG. **19**, where the status lines **1370** for both line 1 and line 2 indicate that their respective transfers have been completed.

An active call summary list screen **2000**, shown in FIG. **20**, summarizes all the active calls for the information appliance **210** at a given moment. Tapping the active call summary "OK" button **2005** returns the user to the previous screen. Tapping on the record itself for a particular line will allow the user to browse the call information corresponding to the selected call. For example, a connected line screen **1500** for line 01 will appear in response to tapping the "line 1" record **2010**, and a connected line screen **1500** for line 02 will appear in response to tapping the "line 2" record **2020**.

Choosing the forward prompt **1630** from the connected line pop-up menu **1610** will lead the user into either the first forwarding screen **2100**, shown in FIG. **21**, if there are no calls being forwarded currently. If calls are already being forwarded when the user chooses the forward prompt **1630** from the connected line pop-up menu **1610**, the second forwarding screen **2200** appears, as shown in FIG. **22**.

To forward a call, the user enters the number and taps the "Forward" button **2110** on the first forwarding screen **2100** or the second forwarding screen **2200**. An example of the call forwarding is shown in FIG. **8**'s DLLP call forwarding state diagram **800**. In response to a user inputting a number of "9876" **810** as the number for the forwarded to network connected device, the palm-sized computer **343** sends a call forwarding message **812** to the Ethernet telephone **310**. The Ethernet telephone **310** establishes the call forwarding feature with the gateway server **270**. For this example, the gateway server **270** then indicates its acceptance of the forwarding request to the Ethernet telephone **310**. The Ethernet telephone **310** responds by sending a call forward message **816** to the palm-sized computer **343**. The palm-sized computer **343** then displays the call forwarded status **818**, as shown in the currently forwarded line **2210**. From the second forwarding screen **2200**, the user can also choose

to disable the feature by tapping the "Disable" button **2220**. The "KeyPad" button **1210** performs essentially the same function as described in conjunction with the call placement screen **1200**.

The call history screen **2300**, shown in FIG. **23**, appears when the user chooses the history prompt **1640** from the connected line pop-up menu **1610**. Information about the calls made is summarized in the list provided on the call history screen **2300**. In the first column, "O" indicates outgoing calls while "I" (not shown) indicates incoming calls. The second column displays the phone number for each call listed. The third column shows the time the corresponding call was made, or if the call was made more than a day before the history prompt **1640** was selected, the third column will show the date of the corresponding call instead. Finally, in the last column, the duration of each call is displayed.

Tapping the "Done" button **2310** returns the user to the previous screen, i.e., the screen having the connected line pop-up menu **1610**. To delete all the records, the user taps the "Delete All" button **2320**. A warning screen will appear first before the user is allowed to delete all the records.

If the user taps on the record itself, a detailed single call history window **2410** corresponding to the tapped record will appear, as seen in the detailed single call history screen **2400**, shown in FIG. **24**. Detailed information, including the memo **1530** that had been written for the call, is displayed. The user can choose to dial the number by tapping the "Dial" button **1320**, or delete the record by tapping the "Delete" button **2420**.

Whenever there is an incoming call, the "Call Incoming" window **2500** will pop-up as shown in FIG. **25**. One example of the incoming call process is shown in the DLLP palm-size computer **343** dialing state diagram **600** in FIG. **6**. Upon receiving an incoming call from a network connected device on line 02, the Ethernet telephone **310** sends an incoming call message **632** for the second 02 line to the palm-sized computer **343**. Upon receiving the incoming call message **632** for the second 02 line, the palm-sized computer **343** displays the "Caller Name" **2510** and "Caller ID" **2520**. Tapping the "Save" button **2530** will save the record to the Address database. Tapping the "Answer" button **2540** or the "Reject" button **2550** answers or rejects the call respectively. For example, in response to the user tapping the "Answer" button **2540**, the palm-sized computer sends an answer message **636** for the second 02 line to the Ethernet telephone **310**, and the Ethernet telephone connects the network connected device to line 02. The call incoming "OK" button **2560** is used if the user decides to answer the call at a later and more convenient time.

A features menu bar **2600**, shown in FIG. **26**, is also available for several functions. The features menu bar **2600** provides most of the features that are available in the connected line pop-up menu **1610** list. These features are repeated to allow added convenience to the user. To access the features menu bar **2600**, the user taps the silkscreen menu button **2605**. The information appliance then displays a plurality of graphical user interfaces (not shown) including an image corresponding to the features menu bar **2600** on the screen. The user then taps the appropriate image to display the features menu bar **2600**.

The "Dial" option **2610** displays a dial screen **2800** with a larger keypad **2810**, but performs the same functions as described in conjunction with FIG. **13** for placing a call section. This dial screen **2800** is shown in FIG. **28**. A "Reset" option **2620** is also available to reset the palm-sized computer **343** telephone program.

Tapping the "Options" bar **2630** results in display of the options menu bar **2700** as shown in FIG. **27**. A "Phone Lookup" option **2710** provided in the options menu bar **2700** listing, allows the user to look up a certain person from an Address application/database in the palm-sized computer **343**. The phone lookup application performs the same function as phone lookup applications in many other standard palm-sized computer **343** applications, including the PalmPilot. The "About" option **2720** shows the Copyright information as shown in the copyright information screen **2900**, in FIG. **29**. The rest of the choices are further described later in this section.

After selecting the "Phone Settings" option **2730**, the user is allowed to adjust the Ethernet telephone's **310** ringer volume, speaker volume (where applicable) and handset volume by means of tapping the up or down arrowhead displayed on the phone settings screen **3000**, shown in FIG. **30**. If the transfer and forwarding feature are available, the user can also choose to turn them on or off.

By selecting the "User Settings" option **2740**, the user can "fine tune" the operation of the Ethernet telephone **310** beyond the settings established during initialization. The user can choose to use the telephone's **240** identity or the palm-sized computer's **343** identity (i.e. the user's mobile identity) by selecting the corresponding indication on the phone identity prompts **3110** as shown on the user settings screen **3100**, shown in FIG. **31**. The access level of the user can also be adjusted to local calls only or unrestricted calls by selecting the corresponding indication on the user restrictions prompts **3120**. The number of lines to support for the user can be set by pressing up and/or down on the set number of lines to support arrows **3130**. The number of conferences to support can be set up and/or down using the set max conferences arrows **3140**. Since the maximum number of conferences and the number of lines to be supported settings are somewhat more sensitive to resource restraints, a password will be required and the user will be prompted to enter it in an enter password screen **3300**, such as that shown in FIG. **33**.

If the user decides to change the password, he/she can tap the "Change Password" button **3150**. A change password screen **3200**, shown in FIG. **32**, then pops up asking the user for the new Password. The change is double confirmed by requiring the user to enter the new password twice.

A conferencing screen **3400** is shown in FIG. **34**. The "Hold" button **3410** is used to put the entire conference on hold, while the "End Conf" button **3420** ends the conference. The "More" button **3430** summons more commands from a conferencing options window **3500** as shown in FIG. **35**. The options window **3500** enables the user to select commands corresponding to the following: a "Split" button **3510** (to split the conference), a "New Conf" button **3520** (to start a new conference), an "End All" button **3530** (to end all the current conferences), an "Add Line" button **3540** (to add another party to the conference), a "Hold" button **3410**, an "End Conf" button **3550** (to end the conference), a "New Line" button **3560** (to make a new, separate call) and a "Switch" button **3570** (to switch to another active line or conference). Tapping the "Cancel" button **3580** returns the user to the previous screen.

The companion appliance telephone **240** program can be used in conjunction with voice mail messages, to allow the user to access voice mail with a touch of the screen. Voice mail information (user name, phone number and date/time called) can be viewed from the companion appliance telephone **240** program whereby users can sort or search for a particular message.

The companion appliance telephone **240** program can be used in conjunction with a Smart Card to access the user's billing or charging information stored and viewed in the palm-sized computer. Access to such information can require user access "tokens" and security information. The charging information can be updated on a per call basis.

Another reset option that resets the Ethernet telephone **310** as well as the companion appliance telephone **240** program can be provided.

A "dialog" window or a "white-board" window can be provided to share and transmit messages and drawings across the network.

A pop-up window to generate a Keypad entry screen can be provided.

The "Call History" information generated by the companion appliance telephone **240** program can be transmitted from the information appliance **210** to a personal computer for storage through a synchronization process such as the HotSync process for the PalmPilot.

The memo field text related to the call can be cut and pasted into a Memo application in the portable computer **320**.

The Telephone

The fifth aspect of the invention provides a telephone **240** adapted for connection to a portable computer **320**. The telephone **240** comprises a port for connecting to the portable computer **320**, network communication capabilities including a communication port, portable computer companion capabilities, and processing resources adapted to exchange data with the portable computer. The portable computer **320** has user information corresponding to a user. The data includes the user information, data corresponding to the network communication capabilities, and the portable computer companion capabilities. The exchange of data enables the telephone **240** to discover user information and capabilities of the portable computer **320**, provide the network communication capabilities and the portable computer companion capabilities to the portable computer **320**, and indicate the network communication capabilities to devices connected to the telephone via a network.

In some embodiments, the portable computer **320** is adapted to control execution of telephony programs. The data exchanged with the portable computer **320** includes data corresponding to portable computer control of the telephony programs. Responsive to commands from the portable computer **320**, the exchange of the data enables the telephone **240** to communicate with devices connected to the telephone.

In some embodiments, the network communication capabilities include a number of communications lines supported, a number of conferences supported, forwarding feature support, call transfer feature support, voice mail support, an identification corresponding to the telephone **240**. The portable computer **320** companion capabilities comprise audible component volumes.

In some embodiments, the telephone **240** has an identification and the portable computer **320** has an identification. The portable computer **320** companion capabilities include the telephone **240** presenting the portable computer identification to the network connected devices in place of the telephone identification.

In some embodiments, the telephone **240** comprises an Ethernet telephone **310**.

In some embodiments, the data exchanged with the portable computer **320** corresponding to the telephone **240** capabilities and the user information are formatted according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions.

In some embodiments, the telephone **240** includes processing resources adapted to receive incoming call data from a gateway server **270** indicating that a first network connected device is waiting to start a call with a user. The telephone **240** also includes processing resources adapted to transform the incoming call data into an incoming call message formatted in a data link layer protocol **328** for transmission to the portable computer **320**. The data link layer protocol **328** encapsulates frames formatted according to the application layer protocol.

In some embodiments, the telephone **240** provides the network communication capabilities to a user without connection to the portable computer **320**.

In some embodiments, the telephone **240** includes processing resources for Internet access. For some of these embodiments, the processing resources for Internet access include Internet applications **367**, display/user input transfer software **373**, transmission control **359** software, Internet protocol **356** software, and carrier sense multiple access/collision detection **368** software. For some of these embodiments, the processing resources for Internet access include an Internet access application **369**, transmission control **359** software, carrier sense multiple access/collision detection **368** software, and Internet protocol **356** software.

In some embodiments, the telephone **240** includes a display **1005**, and processing resources for video display and capture.

The foregoing description of embodiments of the invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise forms disclosed. Obviously, many modifications and variations will be apparent to practitioners skilled in this art. It is intended that the scope of the invention be defined by the following claims and their equivalents.

What is claimed is:

1. A method for transmitting data from a portable computer to a telephone comprising:

connecting the portable computer with the telephone, the telephone having operating capabilities, the telephone connected to network connected devices;

supplying the portable computer with telephone operating parameter data for a communications session between the telephone and one or more of the network connected devices, the communications session including an exchange of messages with the one or more of the network connected devices;

the portable computer exchanging the telephone operating parameter data and the operating capabilities with the telephone; and

the portable computer establishing telephone operating parameters for the communications session based on the telephone operating parameter data and the operating capabilities, the telephone operating parameters providing options and features for the communications session.

2. The method of claim **1**, wherein after the establishing telephone operating parameters step, the method includes a user placing a phone call, the placing including:

the user starting a dialing program;

the user inputting values to the portable computer, the values corresponding to a recipient network connected device;

the portable computer displaying the values and phone call status information;

the portable computer transforming the values into input data formatted according to a data link layer protocol, the data link layer protocol encapsulating frames formatted according to an application layer protocol, the application layer protocol adapted for telephony functions; and

the portable computer transmitting the input data to the telephone.

3. The method of claim **2**, wherein after the transmitting step the method includes:

the user entering text data to form a memo corresponding to the phone call;

the portable computer creating a data record corresponding to the phone call; and

the portable computer attaching the memo to data record.

4. The method of claim **2**, wherein the portable computer includes a user interface and a display, the inputting is accomplished through the user interface, and the portable computer displays a telephone number entry field in which the user inputs the values.

5. The method of claim **1**, wherein:

the telephone connected by a local area network link to a router;

the router connected to at least one packet based network including an Internet source; and

communications between the router and the telephone formatted according to packet based network application protocols.

6. The method of claim **5**, wherein:

the portable computer includes processing resources for Internet access; and

the telephone includes processing resources for Internet access.

7. The method of claim **5**, wherein:

the portable computer includes processing resources for Internet access including Internet applications, transmission control software, and Internet protocol software; and

the telephone includes an Internet access application, transmission control software, carrier sense multiple access/collision detection software, and Internet protocol software.

8. The method of claim **5**, wherein:

the portable computer includes processing resources for Internet access including Internet display applications and display/user input transfer software; and

the telephone includes processing resources for Internet access including Internet applications, display/user input transfer software, transmission control software, Internet protocol software, and carrier sense multiple access/collision detection software.

9. The method of claim **1**, wherein after the establishing telephone operating parameters step, the method includes a user placing a conference call, the placing including:

the user starting a conferencing program;

the user inputting values to the portable computer, the values corresponding to a plurality of conference participant network connected devices;

the portable computer displaying the values and conference call status information;

the portable computer transforming the values into input data formatted according to a data link layer protocol, the data link layer protocol encapsulating frames formatted according to an application layer protocol, the application layer protocol adapted for telephony functions; and

the portable computer transmitting the input data to the telephone.

10. The method of claim **9**, wherein:

the portable computer includes a display;

the starting includes a user selection of a user interface element for a conferencing feature, the user interface element disposed on the display;

in response to the user selection, the portable computer displays a list of conference actions for subsequent user selection.

11. The method of claim **1**, wherein:

the portable computer corresponds to a user and has a display; and

after the establishing telephone operating parameters step, the method includes the portable computer receiving an incoming call, the receiving includes:

the portable computer receiving an incoming call message from the telephone, the incoming call message indicating that a first network connected device is waiting to start a call with the user; and

the portable computer displaying an incoming call screen on the display.

12. The method of claim **11**, wherein:

the portable computer includes an address database;

the incoming call message includes a caller name, and a caller identification;

the incoming call screen includes a user selection for saving the caller name and the caller identification to the address database.

13. The method of claim **1**, wherein the method includes:

prior to connecting the portable computer with the telephone, the portable computer storing user information, the user information includes an identification corresponding to the portable computer, user access parameters, and user characteristics corresponding to the telephone operating parameter data; and

establishing telephone operating parameters includes the user selecting user setting inputs, the user setting inputs corresponding to the portable computer identification, user access parameters, and user characteristics, the user selecting changing the corresponding telephone operating parameter data.

14. The method of claim **13**, wherein the user setting inputs include the network address of the telephone.

15. The method of claim **1**, wherein the exchange of messages includes simultaneous exchanges of voice and packet data messages.

16. The method of claim **1**, wherein the method includes, prior to connecting the portable computer with the telephone, the portable computer storing user information, the user information includes user characteristics, the operating parameter data includes user information.

17. The method of claim **1**, wherein the method includes, prior to connecting the portable computer with the telephone, the portable computer storing user information, the user information includes an identification corresponding to the portable computer and user access parameters.

18. The method of claim **1**, wherein the operating parameter data comprises constructs formatted according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions.

19. The method of claim 1, wherein the telephone comprises an Ethernet telephone.

20. The method of claim 1, wherein after the establishing telephone operating parameters step, the method includes a user forwarding a call, the forwarding including:

the user starting a forwarding program; and

the user inputting a number to the portable computer, the number corresponding to a forwarding destination network connected device;

the portable computer displaying the number and forwarding status information;

the portable computer transforming the number into input data formatted according to a data link layer protocol, the data link layer protocol encapsulating frames formatted according to an application layer protocol, the application layer protocol adapted for telephony functions; and

the portable computer transmitting the input data to the telephone.

21. The method of claim 1, wherein after the establishing telephone operating parameters step, the method includes a user placing a phone call, the placing including:

the telephone receiving input data from the portable computer, the input data formatted according to a data link layer protocol, the data link layer protocol encapsulating frames formatted according to an application layer protocol, the application layer protocol adapted for telephony functions; and

the telephone transforming the input data into transport data formatted according to a transport protocol for a packet switched network; and

the telephone transmitting the transport data to a gateway server, the gateway server connected to at least one switched circuit network including a public switched telephone network.

22. The method of claim 1, wherein after the establishing telephone operating parameters step, the method includes a user placing a conference call, the placing including:

the telephone receiving input data from the portable computer, the input data formatted according to a data link layer protocol, the data link layer protocol encapsulating frames formatted according to an application layer protocol, the application layer protocol adapted for telephony functions; and

the telephone transforming the input data into transport data formatted according to a transport protocol for a packet switched network; and

the telephone transmitting the transport data to a gateway server, the gateway server connected to at least one switched circuit network including a public switched telephone network.

23. The method of claim 1, wherein after the establishing telephone operating parameters step, the method includes a user forwarding a call, the forwarding including:

the telephone receiving input data from the portable computer, the input data formatted according to a data link layer protocol, the data link layer protocol encapsulating frames formatted according to an application layer protocol, the application layer protocol adapted for telephony functions; and

the telephone transforming the input data into transport data formatted according to a transport protocol for a packet switched network; and

the telephone transmitting the transport data to a gateway server, the gateway server connected to at least one switched circuit network including a public switched telephone network.

24. The method of claim 1, wherein after the connecting, the method includes:

powering up the portable computer; and

in response to the powering up of the portable computer, initializing the portable computer and the telephone, the initializing including the exchanging and establishing steps.

25. The method of claim 1, wherein:

the portable computer includes a display;

after the establishing step, the method includes a user starting a telephony program, the starting including:

the portable computer displaying a user interface element corresponding to a first menu on the display, the first menu including a list of telephony programs available for a particular connected line;

the user selecting the first menu user interface element;

the portable computer displaying a first menu list, the first menu list including user interface elements corresponding to the telephony programs; and

the user selecting a program from the list.

26. The method of claim 1, wherein establishing telephone operating parameters includes the user selecting phone setting inputs.

27. The method of claim 1, wherein the network connected devices include a gateway server, the gateway server providing access to a public switched telephone network.

28. The method of claim 1, wherein the method includes, prior to the connecting step, the telephone exchanging voice messages with at least one of the network connected devices.

29. The method of claim 1, wherein:

the telephone connected to a gatekeeper, a directory server and a gateway server by a local area network link; and

communications between the telephone and the gatekeeper, gateway server, and directory server formatted according to a soft private branch exchange telephony application layer protocol.

30. The method of claim 1, wherein:

the method includes the user starting a telephony program; and

the portable computer controls execution of the telephony program.

31. A method for exchanging voice and data messages between a telephone and devices connected to a network, the telephone connected to the network, the method comprising:

connecting the telephone with a portable computer;

the portable computer exchanging telephone operating parameter data with the telephone, the operating parameter data providing options for communications between the telephone and the network connected devices; and

responsive to a user indication of a desired communication, the portable computer exchanging call data with the telephone, the call data corresponding to the desired communication, the call data formatted according to an application layer protocol, and the underlying transport, network, and data link layer protocols, the application layer protocol having frame formats for telephony functions;

the telephone exchanging messages with an addressed network connected device, the messages corresponding to the desired communication, the addressed network connected device having a network address, the mes-

sage including data corresponding to the address of the addressed network connected device.

32. The method of claim **31**, wherein, the method includes, prior to connecting the portable computer with the telephone, the portable computer storing user information, the operating parameter data comprises the user information, the user information comprises an identification corresponding to the portable computer and user access parameters.

33. The method of claim **32**, wherein:

the telephone has an identification; and

the method includes the telephone presenting the identification corresponding to the portable computer in place of the telephone identification to devices connected to and communicating with the telephone.

34. The method of claim **31**, the method includes:

the telephone requesting a connection to a first network connected device; and

the first network connected device responding to the connection request; wherein the message data comprise:

in response to a user input, a phone number corresponding to the first network connected device transmitted from the portable computer to the telephone; and

upon receipt by the telephone of the first network connected device response to the connection request, a first connection made response transmitted from the telephone to the portable computer.

35. The method of claim **31**, wherein the telephone comprises an Ethernet telephone, and the portable computer comprises a palm-sized computer.

36. The method of claim **31**, wherein:

the telephone is connected to a gatekeeper, a directory server and a gateway server by a local area network link; and

communications between the telephone and the gatekeeper, the gateway server, and the directory server formatted according to a soft private branch exchange telephony application layer protocol.

37. The method of claim **31**, wherein:

the telephone connected by a local area network link to a router;

the router connected to at least one packet based network including an Internet source; and

communications between the router and the telephone formatted according to packet based network application protocols.

38. The method of claim **31**, wherein the exchange of messages includes simultaneous exchanges of voice and packet data messages.

39. A portable computer adapted for connection to a telephone, the portable computer comprising:

a port for connecting to the telephone, the telephone having capabilities;

a memory storing user information corresponding to a user; and

processing resources adapted to exchange data with the telephone, the data including the user information and data corresponding to the telephone capabilities;

the exchange of the data enabling the portable computer to:

discover capabilities of the telephone;

provide the user information to the telephone; and

establish telephone operating parameters for telephone communications with devices connected to the telephone based on the user information and the telephone capabilities.

40. The portable computer of claim **39** including:

a display providing user interface graphic elements corresponding to data exchanged with the telephone; and

a user interface enabling the user to input data supplementing the user information provided to the telephone.

41. The portable computer of claim **40**, wherein the data exchanged with the telephone includes:

data corresponding to portable computer control of telephony programs; and

data corresponding to the status of the devices connected to the telephone.

42. The portable computer of claim **39**, wherein the user information comprises:

an identification corresponding to the portable computer;

user characteristics; and

user access parameters.

43. The portable computer of claim **42**, wherein:

the telephone has an identification, and

the identification corresponding to the portable computer presented by the telephone in place of the telephone identification to devices connected to and communicating with the telephone.

44. The portable computer of claim **39**, wherein the portable computer comprises a palm-sized computer.

45. The portable computer of claim **39**, wherein the data exchanged with the telephone corresponding to portable computer control of the execution of the telephony programs, the telephone capabilities, and the user information formatted according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions.

46. The portable computer of claim **39**, wherein the portable computer adapted to provide data processing and user interface functions without connection to the telephone.

47. The portable computer of claim **39** includes processing resources for Internet access.

48. The portable computer of claim **39** including processing resources for Internet access including Internet applications, transmission control software, and Internet protocol software.

49. The portable computer of claim **39** including processing resources for Internet access including Internet display applications and display/user input transfer software.

50. The portable computer of claim **39** including processing resources for user interface support of video data.

51. The portable computer of claim **39** including processing resources for user interface support of video data including video data decoding, and video display.

52. The portable computer of claim **39** including processing resources for user interface support of video data, video data decoding, video display, and video camera image data.

53. The portable computer of claim **39**, wherein the exchange of the data enables the portable computer to control execution of telephony programs.

54. A telephone adapted for connection to a portable computer, the telephone comprising:

a port for connecting to the portable computer, the portable computer having user information corresponding to a user;

network communication capabilities including a communication port;

portable computer companion capabilities;

processing resources adapted to exchange data with the portable computer, the data including:

the user information,

data corresponding to the network communication capabilities; and

the portable computer companion capabilities;

the exchange of data enabling the telephone to:

discover user information and capabilities of the portable computer;

provide the network communication capabilities and the portable computer companion capabilities to the portable computer; and

indicate the network communication capabilities to devices connected to the telephone via a network.

55. The telephone of claim **54**, wherein:

the portable computer adapted to control execution of telephony programs;

the data exchanged with the portable computer includes data corresponding to portable computer control of the telephony programs; and

responsive to commands from the portable computer, the exchange of the data enabling the telephone to communicate with devices connected to the telephone.

56. The telephone of claim **54**, wherein:

the telephone has an identification;

the portable computer has an identification; and

the portable computer identification presented by the telephone in place of the telephone identification to devices connected to and communicating with the telephone.

57. The telephone of claim **54**, wherein the telephone comprises an Ethernet telephone.

58. The telephone of claim **54**, wherein the data exchanged with the portable computer corresponding to the telephone capabilities and the user information formatted according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions.

59. The telephone of claim **54**, including processing resources adapted to:

receive incoming call data from a gateway server indicating that a first network connected device is waiting to start a call with a user;

transform the incoming call data into an incoming call message formatted in a data link layer protocol for transmission to the portable computer, the data link layer protocol encapsulating frames formatted according to the application layer protocol.

60. The telephone of claim **54**, wherein the telephone provides the network communication capabilities to a user without connection to the portable computer.

61. The telephone of claim **54** including processing resources for Internet access.

62. The telephone of claim **54**, including processing resources for Internet access including Internet applications, display/user input transfer software, transmission control software, Internet protocol software, and Carrier Sense Multiple Access/Collision Detection software.

63. The telephone of claim **54**, including processing resources for Internet access including an Internet access application, transmission control software, Carrier Sense Multiple Access/Collision Detection software, and Internet protocol software.

64. The telephone of claim **54**, including:

a display; and

processing resources for video display and capture.

65. A communications system comprising:

a telephone having capabilities;

a portable computer connected to the telephone, the portable computer including:

a port for connecting to the telephone;

a memory storing user information corresponding to a user; and

processing resources adapted to exchange data with the telephone, the data including the user information and data corresponding to the telephone capabilities;

the exchange of the data enabling the portable computer to:

discover capabilities of the telephone;

provide the user information to the telephone; and

establish telephone operating parameters for telephone communications with devices connected to the telephone based on the user information and the telephone capabilities; and

a network link connecting the telephone to network connected devices.

66. The communications system of claim **65** including a router connected to the network link, and packet based network devices connected to the router.

67. The communications system of claim **66**, wherein:

the portable computer includes processing resources for Internet access; and

the telephone includes processing resources for Internet access.

68. The communications system of claim **66**, wherein:

the portable computer includes processing resources for Internet access including Internet applications, transmission control software, and Internet protocol software; and

the telephone includes processing resources for Internet access including an Internet access application, transmission control software, Carrier Sense Multiple Access/Collision Detection software, and Internet protocol software.

69. The communications system of claim **65**, wherein:

the portable computer comprises a palm-sized computer; and

the telephone comprises an Ethernet telephone.

70. The communications system of claim **69**, wherein:

the portable computer includes processing resources for Internet access including Internet display applications and display/user input transfer software; and

the telephone includes processing resources for Internet access including Internet applications, display/user input transfer software, transmission control software, Internet protocol software, and Carrier Sense Multiple Access/Collision Detection software.

71. The communications system of claim **65** including a gateway server connected to the network link, and switched circuit network devices connected to the gateway server.

72. The communications system of claim **65**, wherein the data exchanged with the telephone corresponding to the telephone capabilities and the user information formatted

according to an applications layer protocol, the applications layer protocol having frame formats for telephony functions.

73. The communications system of claim **65**, wherein:

the portable computer includes processing resources for user interface support of video data; and

the telephone includes:

a video display; and

processing resources for video display and capture.

74. The communications system of claim **65**, wherein the portable computer includes processing resources for user interface support of video data, video data decoding, and video display.

75. The communications system of claim **65**, wherein the portable computer includes processing resources for user interface support of video data, video data decoding, video display, and video camera image data.

* * * * *

US006446127B1

(12) **United States Patent**
Schuster et al.

(10) **Patent No.: US 6,446,127 B1**
(45) **Date of Patent: Sep. 3, 2002**

(54) **SYSTEM AND METHOD FOR PROVIDING USER MOBILITY SERVICES ON A TELEPHONY NETWORK**

(75) Inventors: **Guido M. Schuster**, Des Plaines; **Ikhlaq S. Sidhu**, Vernon Hills; **Jerry J. Mahler**, Prospect Heights; **Frederick D. Dean**; **Jacek A. Grabiec**, both of Chicago, all of IL (US); **Ismail Dalgic**, Mountain View, CA (US)

(73) Assignee: **3Com Corporation**, Santa Clara, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/451,388**

(22) Filed: **Nov. 30, 1999**

Related U.S. Application Data

(63) Continuation-in-part of application No. 09/181,431, filed on Oct. 30, 1998, now Pat. No. 6,161,134.

(51) **Int. Cl.**[7] **G06F 13/00**

(52) **U.S. Cl.** **709/227**

(58) **Field of Search** 709/200, 217, 709/218, 219, 223, 227, 228, 229

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,953,198	A	8/1990	Daly et al.	379/61
5,497,339	A	3/1996	Bernard	364/705.05
5,557,658	A	9/1996	Gregorek et al.	379/67
5,606,594	A	2/1997	Register et al.	379/58
5,742,905	A	4/1998	Pepe et al.	455/461
5,838,665	A	11/1998	Kahn et al.	370/260
5,850,433	A	12/1998	Rondeau	379/201
5,894,595	A	4/1999	Foladare et al.	455/414
5,915,008	A	6/1999	Dulman	379/201
5,933,778	A	8/1999	Buhrmann et al.	455/461

FOREIGN PATENT DOCUMENTS

DE	198 13 179 A1	9/1999
EP	0 704 788 A2	4/1996
EP	0 869 688 A2	10/1998
EP	0 881 848 A2	12/1998
WO	WO 97/31492	8/1997
WO	WO 97/33421	9/1997
WO	WO 98/04065	1/1998
WO	WO 98/16051	4/1998
WO	WO 98/30008	7/1998
WO	WO 98/37665	8/1998
WO	WO 99/12365	3/1999
WO	WO 99/19988	4/1999

OTHER PUBLICATIONS

3COM SIP Solutions 1.0 benefits brochure. (4 total pages).
Sidhu, Ikhlaq and Bezaitis, Andrew, Eat or be eaten, www.americasnetwork.com/issues/99issues/991101/991191_eat.htm, printed May 10, 2000. (6 total pages).
Myers, Brad A.; Stiel, Herb; and Gargiulo, Robert, Collaboration Using Multiple PDAs Connected to a PC, Proceedings of the ACM 1998 conference on Computer supported cooperative work, Nov. 14–18, 1998, Seattle, WA. (total 11 pages).

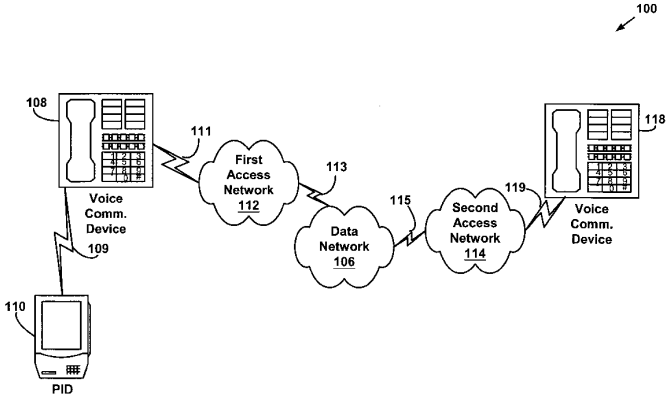
(List continued on next page.)

Primary Examiner—Robert B. Harrell
(74) *Attorney, Agent, or Firm*—McDonnell Boehnen Hulbert & Berghoff

(57) **ABSTRACT**

A system and method for providing user mobility services on a data network telephony system. User attributes may be transmitted from a portable information device, such as a personal digital assistant, to a voice communication device, such as an Ethernet-based telephone. The voice communication device receives the user attributes from the portable information device and may transmit a registration request to a registration server. The registration request may include the user attributes, and is used by the registration server to register the user to the voice communication device in a registration data base. When a call is placed to the user, the registration server may reference the registration data base to direct the call to the voice communication device.

31 Claims, 14 Drawing Sheets



OTHER PUBLICATIONS

Dalgic, Ismail; Borella, Michael; Dean, Rick; Grabiec, Jacek; Mahler, Jerry; Schuster, Guido; and Sidhu, Ikhlaq, True Number Portability and Advanced Call Screening in a SIP–Based IP Telephony System, *IEEE Communications Magazine*, vol. 37, No. 7, Jul. 1999, pp. 96–101. (8 total pages).

Handley/Schulzrinne/Schooler/Rosenberg, SIP: Session Initiation Protocol, Network Working Group, Request for Comments (RFC) 2543, Mar. 1999. (153 pages).

Handley/Schulzrinne/Schooler/Rosenberg, SIP: Session Initiation Protocol, Internet Engineering Task Force, draft–ietf–sip–rfc2543bis–02.ps. Sep. 4, 2000. (131 pages).

Hansson, Allan et al., *Phone Doubler –A Step Towards Integrated Internet and Telephone Communities*, Ericsson Review, No. 4, 1997, pp. 142–152.

International Search Report for PCT Application Serial Number PCT/US00/26618, Dated Feb. 19, 2001.

Pepper, David J. et al., *The Call Manager System: A Platform for Intelligent Telecommunications Services*, Speech Communication, vol. 23, (1997), pp. 129–139.

Dalgic, Ismail et al., *True Number Portability and Advanced Call Screening in a SIP–Based IP Telephony System*, IEEE Communications Magazine, vol. 37, No. 7, (Jul. 1999), pp. 96–101.

International Search Report for PCT Application Serial Number PCT/US00/26094, Dated Jan. 31, 2001.

International Search Report for PCT Application Serial Number PCT/US00/26594, Dated Feb. 6, 2001.

Watanabe, H. et al., *Development of the BTRON–BrainPad*, Proceedings 13th Tron Project International Symposium, Online!, (Dec. 4–7, 1996), pp. 95–103.

Gessler, Stefan et al., *PDAs as Mobile WWW Browsers*, Computer Networks and ISDN Systems, vol. 28, No. 1, (Dec. 1995), pp. 53–59.

International Search Report for PCT Application Serial Number PCT/US00/26650, Dated Feb. 19, 2001.

International Search Report for PCT Application Serial Number PCT/US00/41020, Dated Feb. 21, 2001.

Anquetil, L. P. et al., *Media Gateway Control Protocol and Voice Over IP Gateways. MGCP and VoIP Gateways Will Offer Seamless Interworking of New VoIP Networks with Today's Telephone Networks*, Electrical Communication, (Apr. 1, 1999), pp. 151–157.

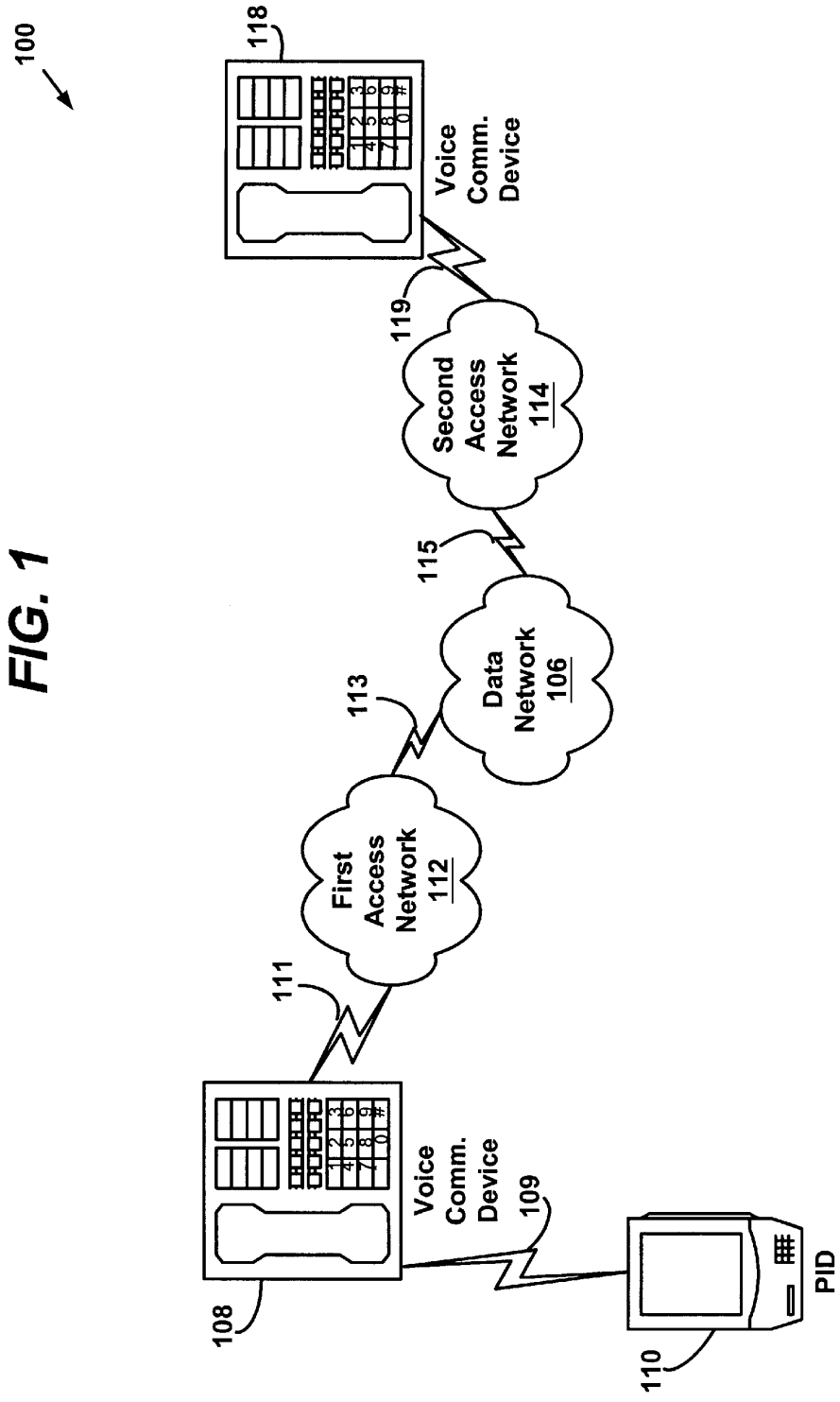
International Search Report for PCT Application Serial Number PCT/US00/26649, Dated Feb. 6, 2001.

Zellweger, Poile T. et al. *An Overview of the Etherphone System and Its Applications*, Xerox Palo Alto Research Center (Mar. 1988), pp. 160–168, XP 000617541.

Terry, Douglas B. et al. *Managing Stored Voice In the Etherphone System*, Xerox Palo Alto Research Center vol. 1, (Feb. 1988), pp. 3–27, XP 000032477.

Terry, Douglas B. et al. *Managing Stored Voice In the Etherphone System*, Operating Systems Review (SIGOPS), US, ACM Head Quarter, New York, NY. vol. 21, No. 5 (Nov. 8, 1987), pp. 103–104, XP 000005196.

FIG. 1


100
118
Voice Comm. Device
119
Second Access Network 114
115
Data Network 106
113
First Access Network 112
111
Voice Comm. Device
109
PID
108
110

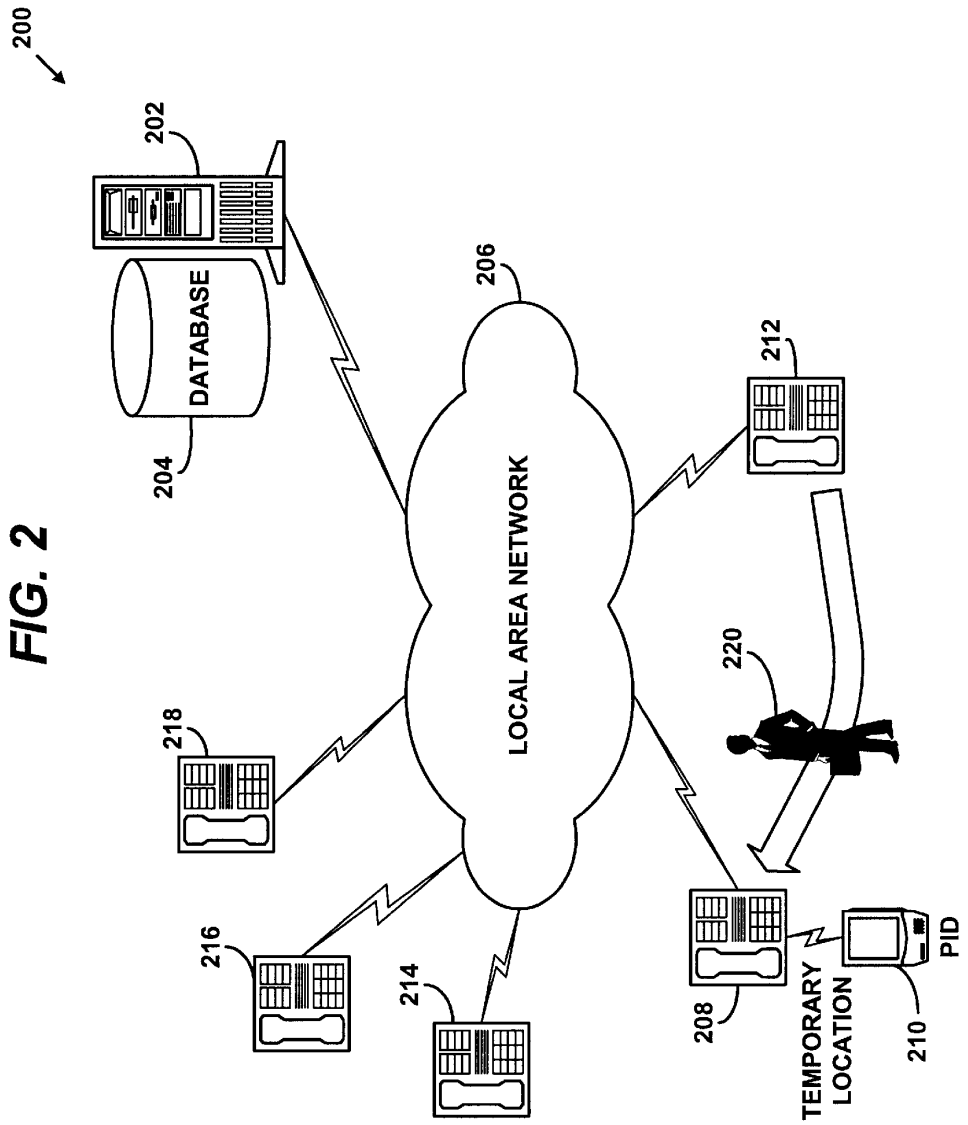

200
202
DATABASE
204
206
LOCAL AREA NETWORK
212
220
218
216
214
208
TEMPORARY LOCATION
210
PID

FIG. 2

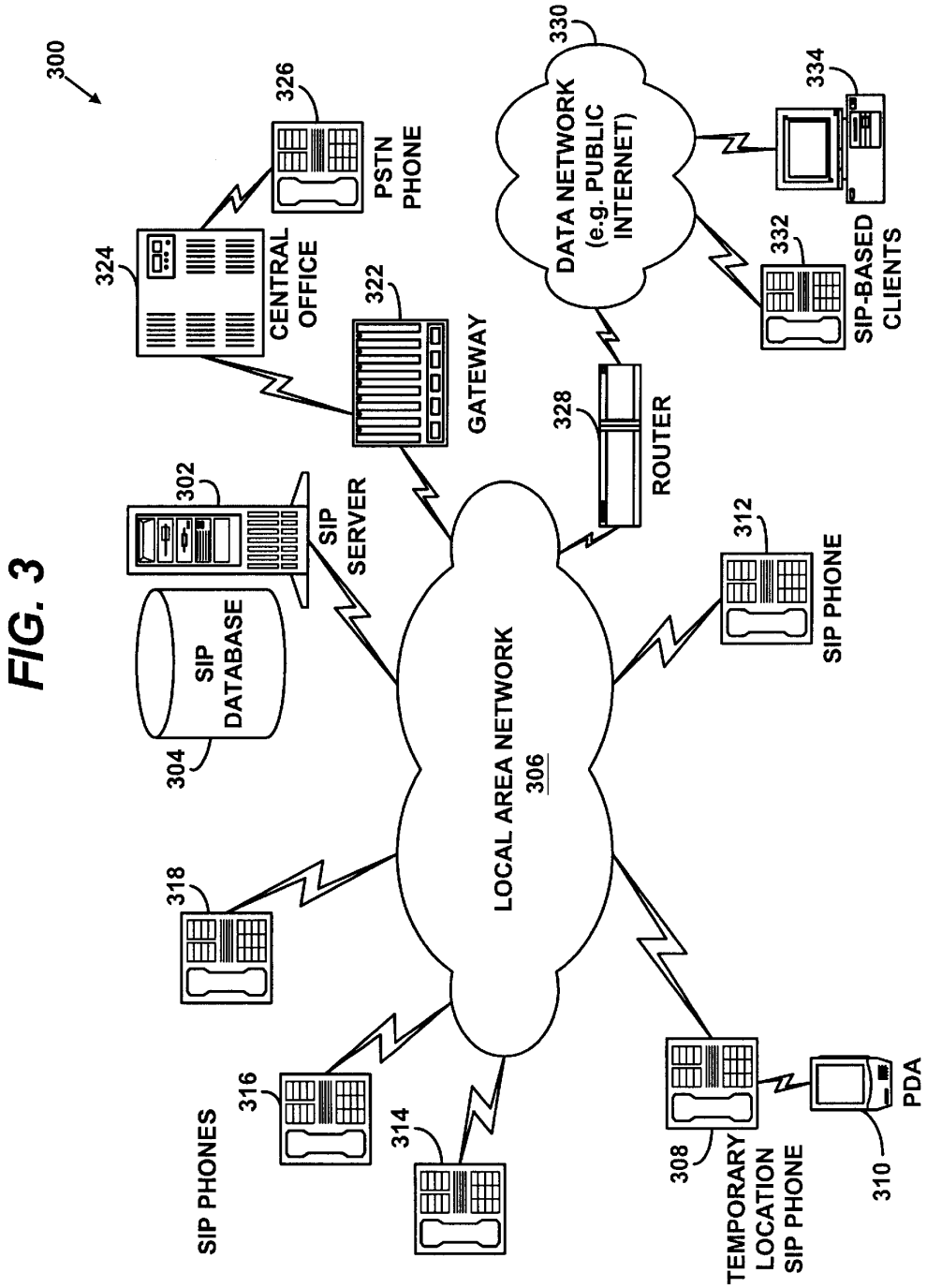

300
326
PSTN PHONE
330
DATA NETWORK (e.g. PUBLIC INTERNET)
334
SIP-BASED CLIENTS
332
324
CENTRAL OFFICE
322
GATEWAY
328
ROUTER
302
SIP SERVER
SIP DATABASE
304
LOCAL AREA NETWORK
306
312
SIP PHONE
318
316
SIP PHONES
314
308
TEMPORARY LOCATION SIP PHONE
310
PDA

FIG. 3

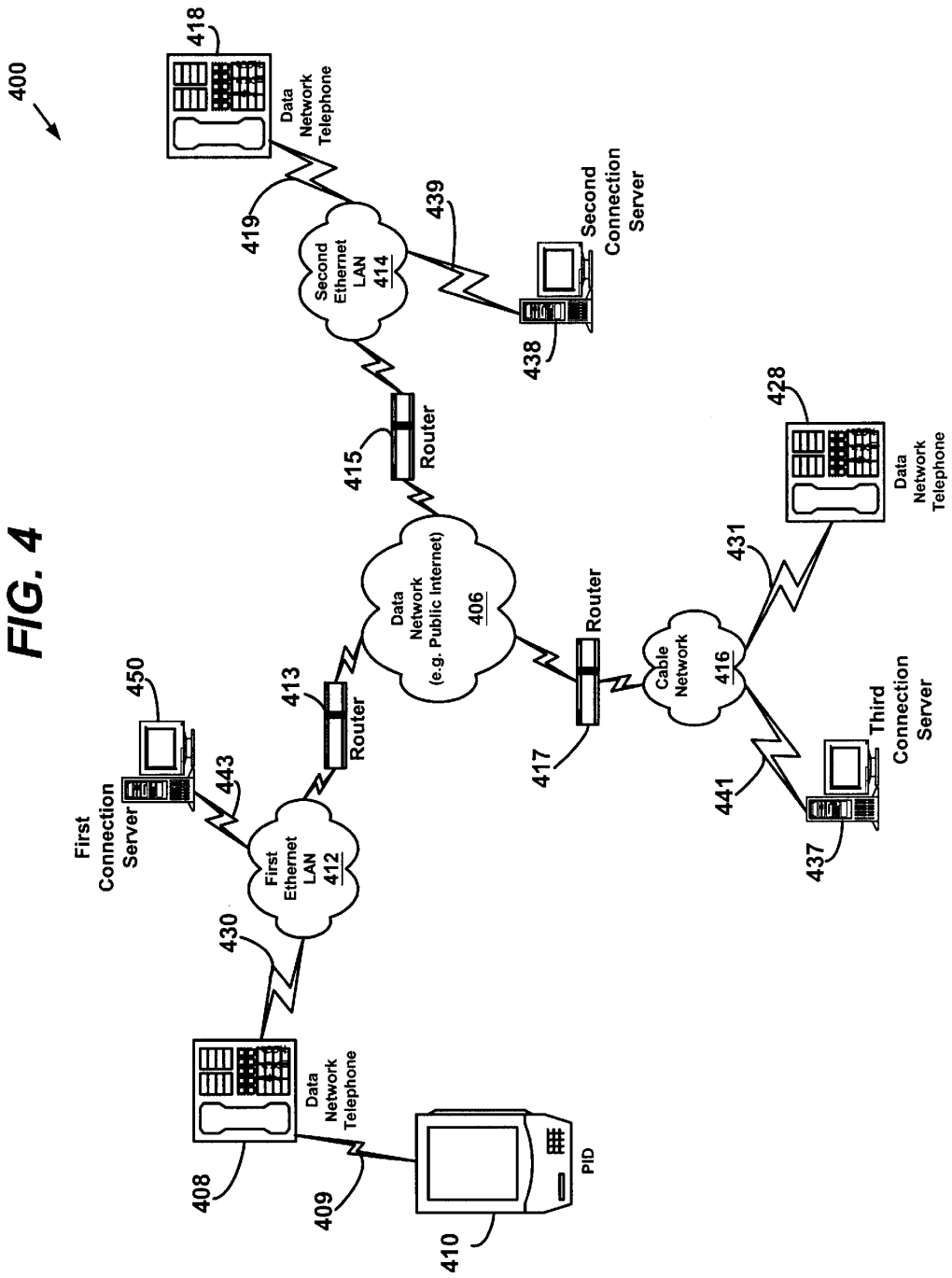
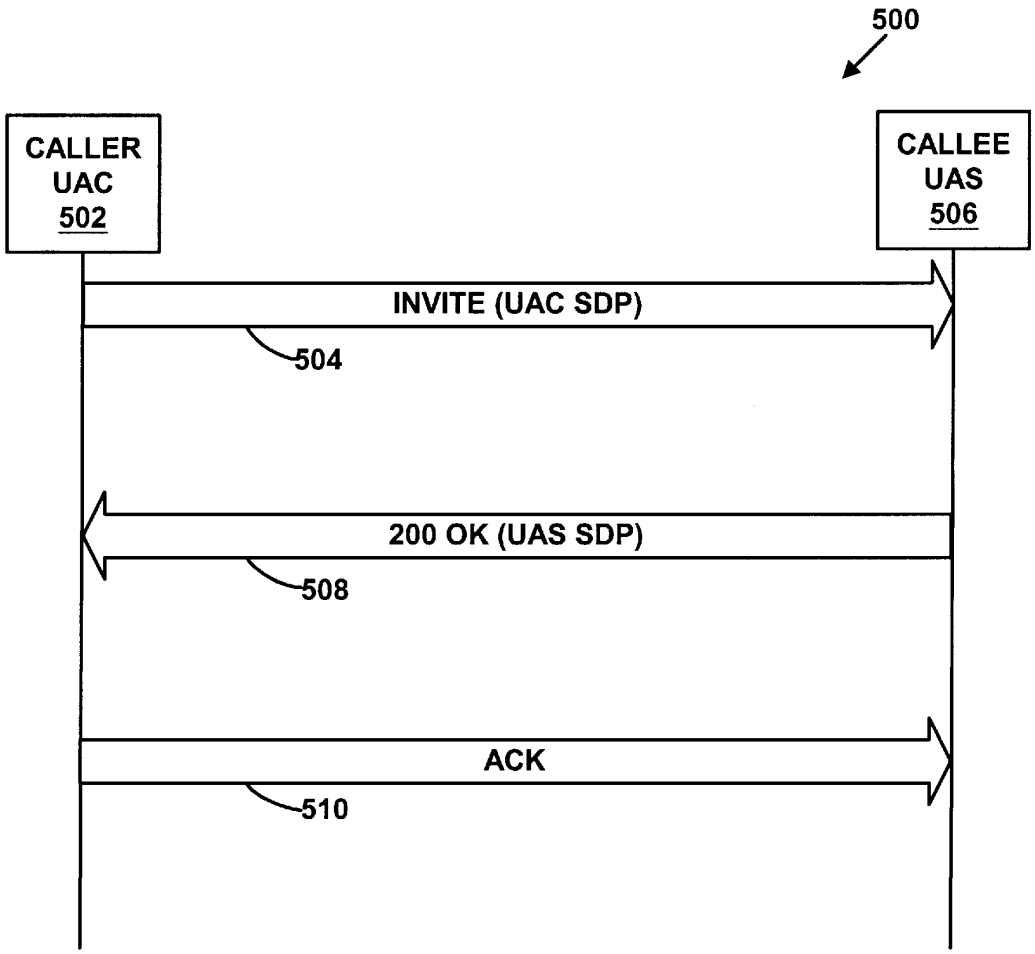

400
418
Data Network Telephone
419
Second Ethernet LAN 414
439
Second Connection Server
438
415
Router
Data Network (e.g. Public Internet) 406
Router
417
Cable Network 416
428
Data Network Telephone
431
Third Connection Server
441
437
450
First Connection Server
443
First Ethernet LAN 412
413
Router
430
Data Network Telephone
408
409
PID
410

FIG. 4

FIG. 5


500
CALLER UAC 502
CALLEE UAS 506
INVITE (UAC SDP)
504
200 OK (UAS SDP)
508
ACK
510

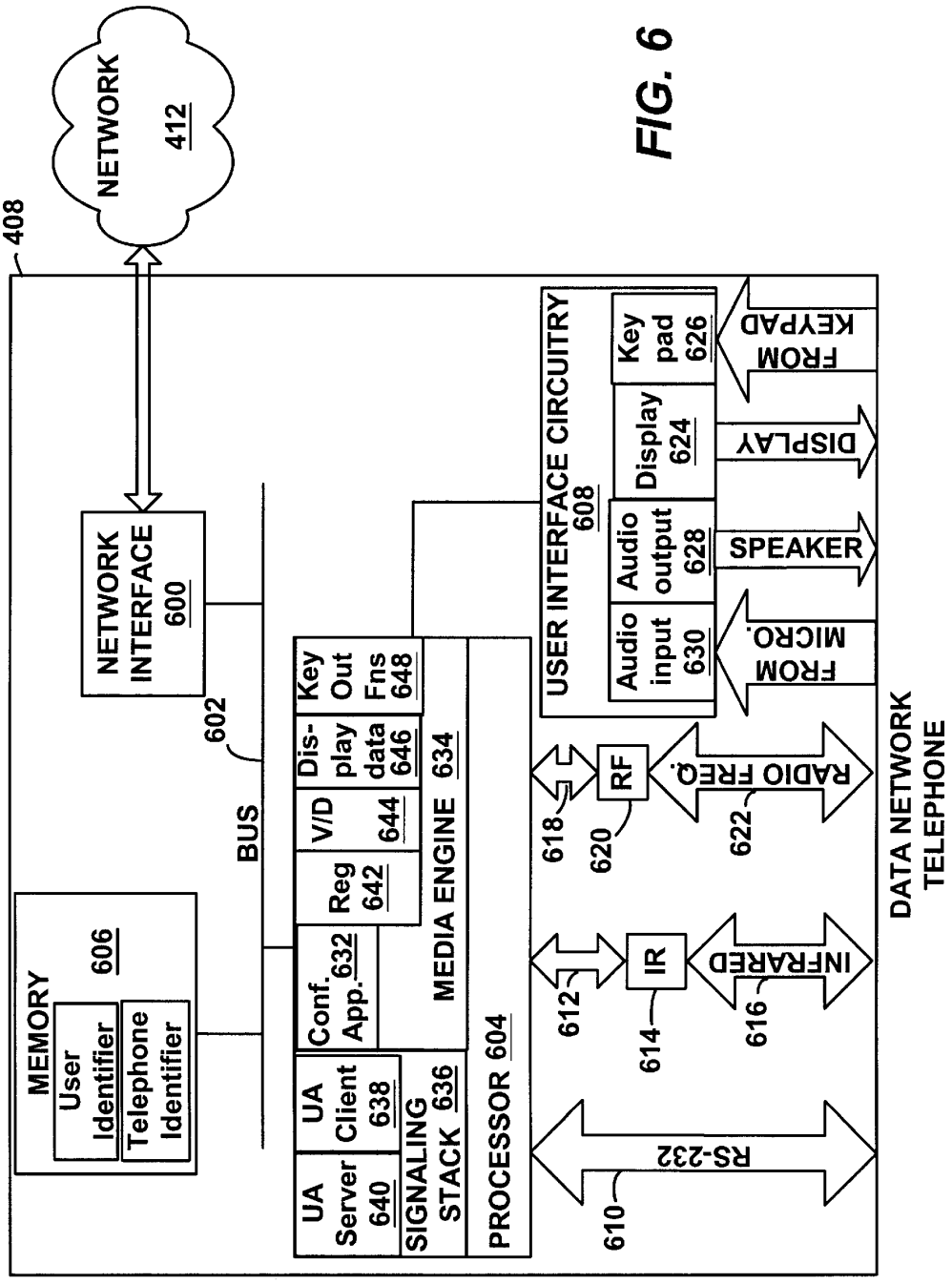
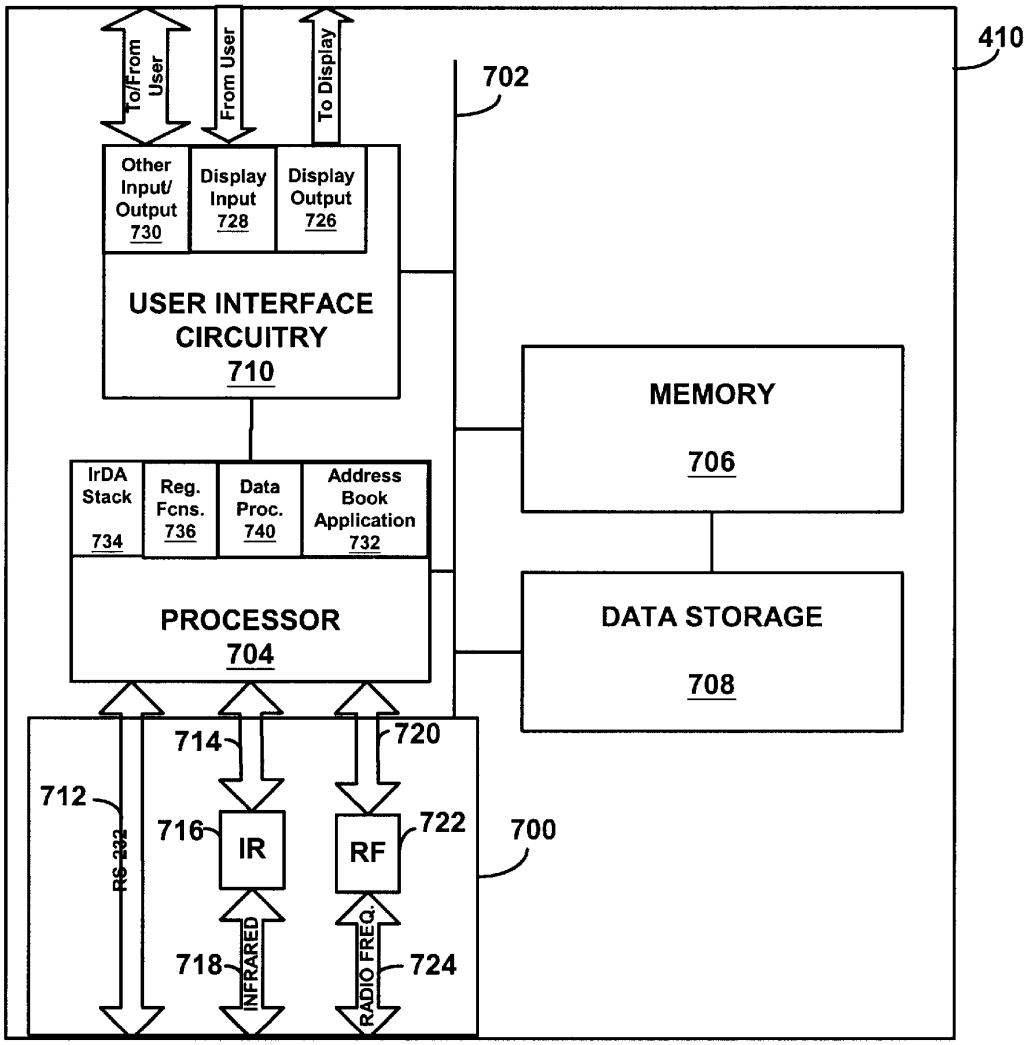

NETWORK
412
408
NETWORK INTERFACE 600
602
BUS
MEMORY
606
User Identifier
Telephone Identifier
UA Server 640
UA Client 638
SIGNALING STACK 636
Conf. App. 632
Reg 642
V/D 644
Dis-play data 646
Key Out Fns 648
MEDIA ENGINE 634
PROCESSOR 604
USER INTERFACE CIRCUITRY 608
Audio input 630
Audio output 628
Display 624
Key pad 626
FROM MICRO.
SPEAKER
DISPLAY
FROM KEYPAD
RS-232
610
IR
612
614
INFRARED
616
RF
618
620
RADIO FREQ.
622
DATA NETWORK TELEPHONE

FIG. 6

FIG. 7


410
To/From User
From User
To Display
702
Other Input/ Output 730
Display Input 728
Display Output 726
USER INTERFACE CIRCUITRY 710
MEMORY 706
IrDA Stack 734
Reg. Fcns. 736
Data Proc. 740
Address Book Application 732
PROCESSOR 704
DATA STORAGE 708
714
720
712
716
IR
RF
722
700
RS-232
INFRARED
RADIO FREQ.
718
724

PID

FIG. 8

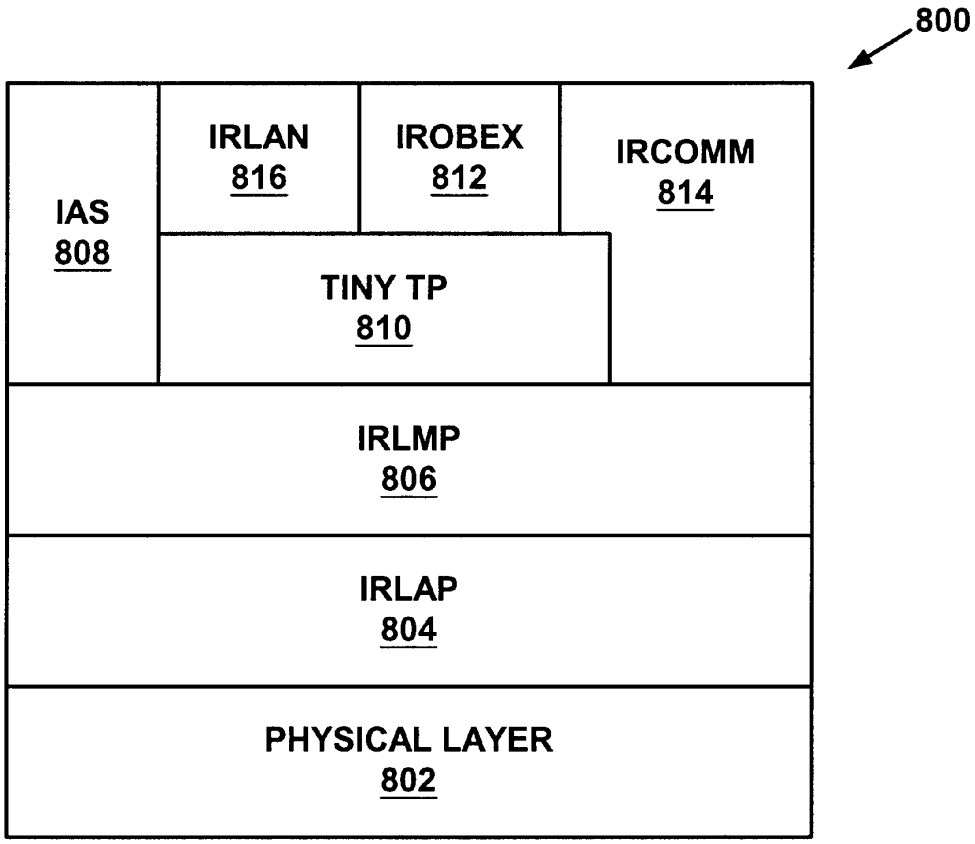

800
IAS
808
IRLAN
816
IROBEX
812
IRCOMM
814
TINY TP
810
IRLMP
806
IRLAP
804
PHYSICAL LAYER
802

FIG. 9

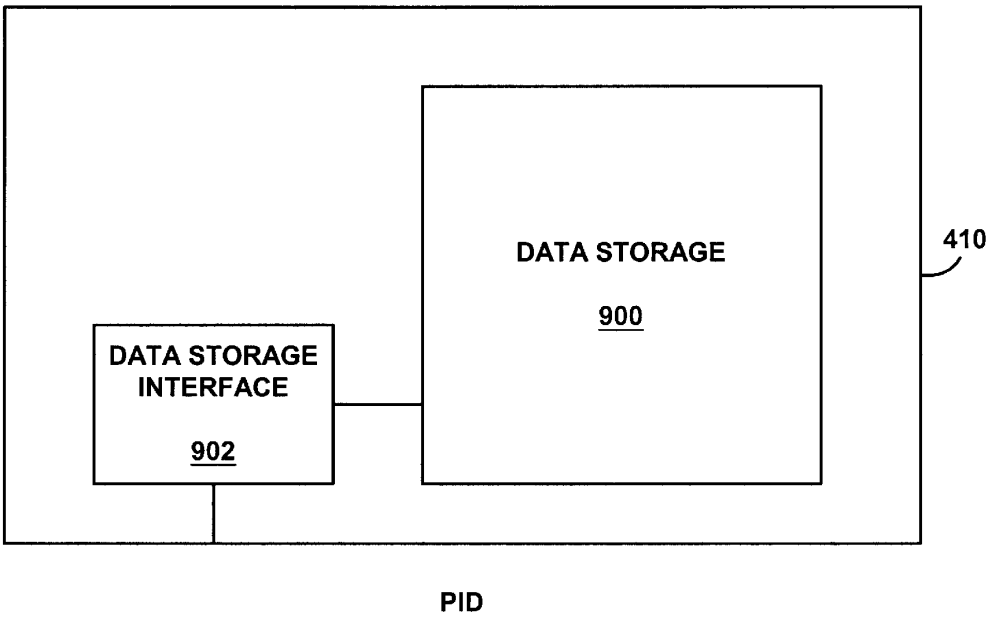


FIG. 10

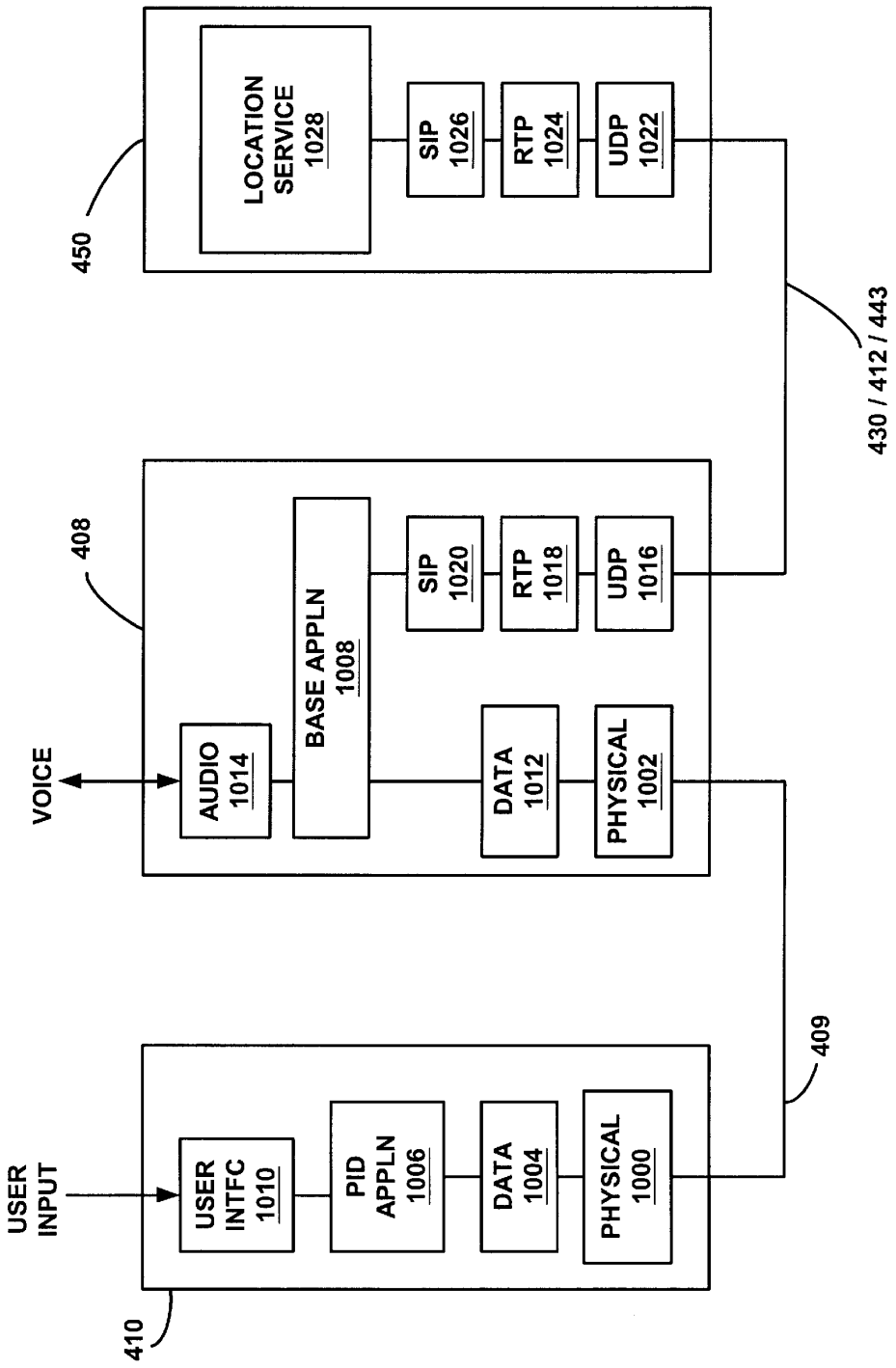

LOCATION SERVICE 1028
SIP 1026
RTP 1024
UDP 1022
450
430 / 412 / 443
408
BASE APPLN 1008
SIP 1020
RTP 1018
UDP 1016
VOICE
AUDIO 1014
DATA 1012
PHYSICAL 1002
409
USER INPUT
USER INTFC 1010
PID APPLN 1006
DATA 1004
PHYSICAL 1000
410

FIG. 11

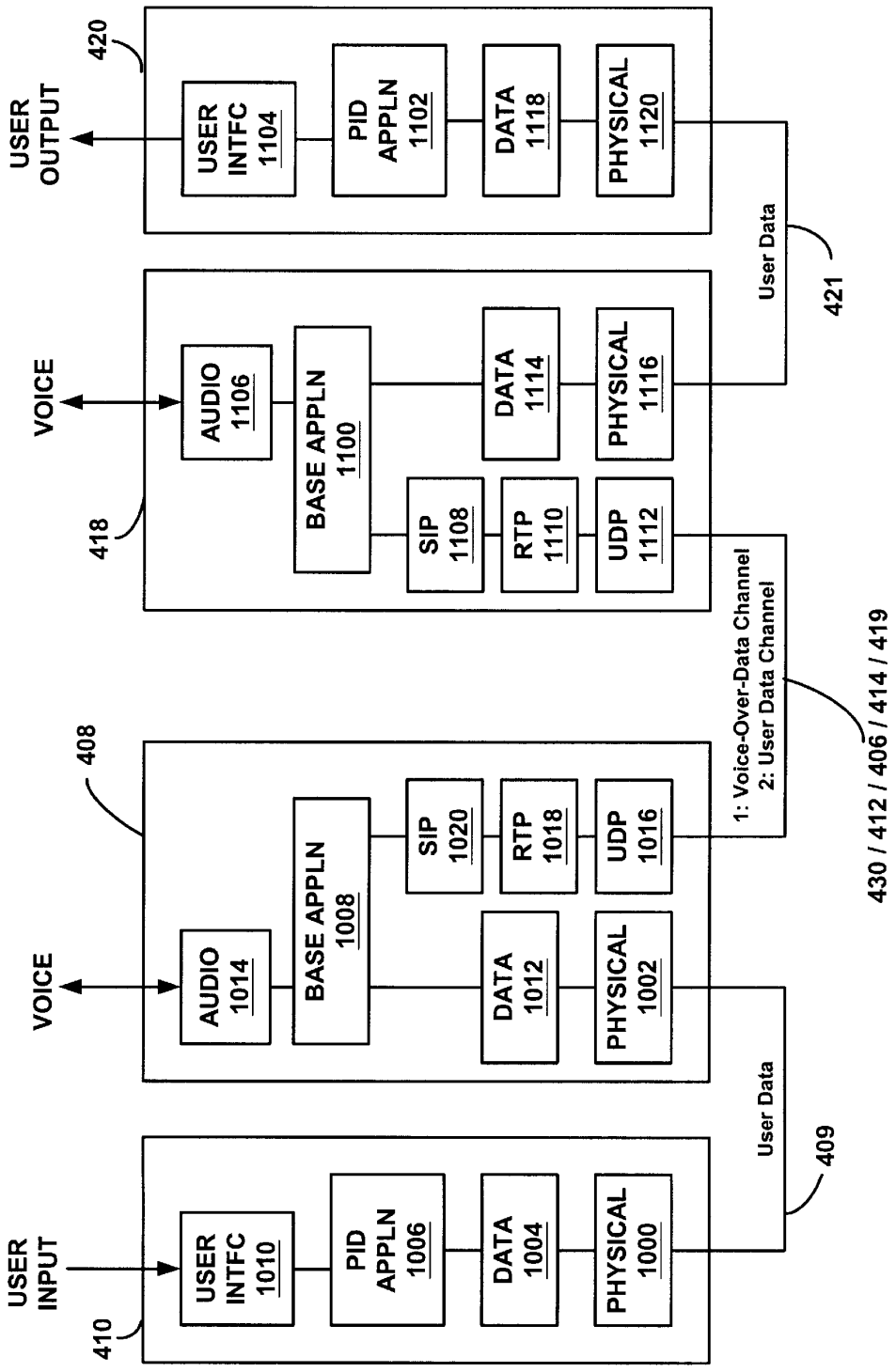

USER INPUT
410
USER INTFC 1010
PID APPLN 1006
DATA 1004
PHYSICAL 1000
User Data
409
VOICE
408
AUDIO 1014
BASE APPLN 1008
SIP 1020
RTP 1018
UDP 1016
DATA 1012
PHYSICAL 1002
1: Voice-Over-Data Channel
2: User Data Channel
430 / 412 / 406 / 414 / 419
VOICE
418
AUDIO 1106
BASE APPLN 1100
SIP 1108
RTP 1110
UDP 1112
DATA 1114
PHYSICAL 1116
User Data
421
USER OUTPUT
420
USER INTFC 1104
PID APPLN 1102
DATA 1118
PHYSICAL 1120

FIG. 12

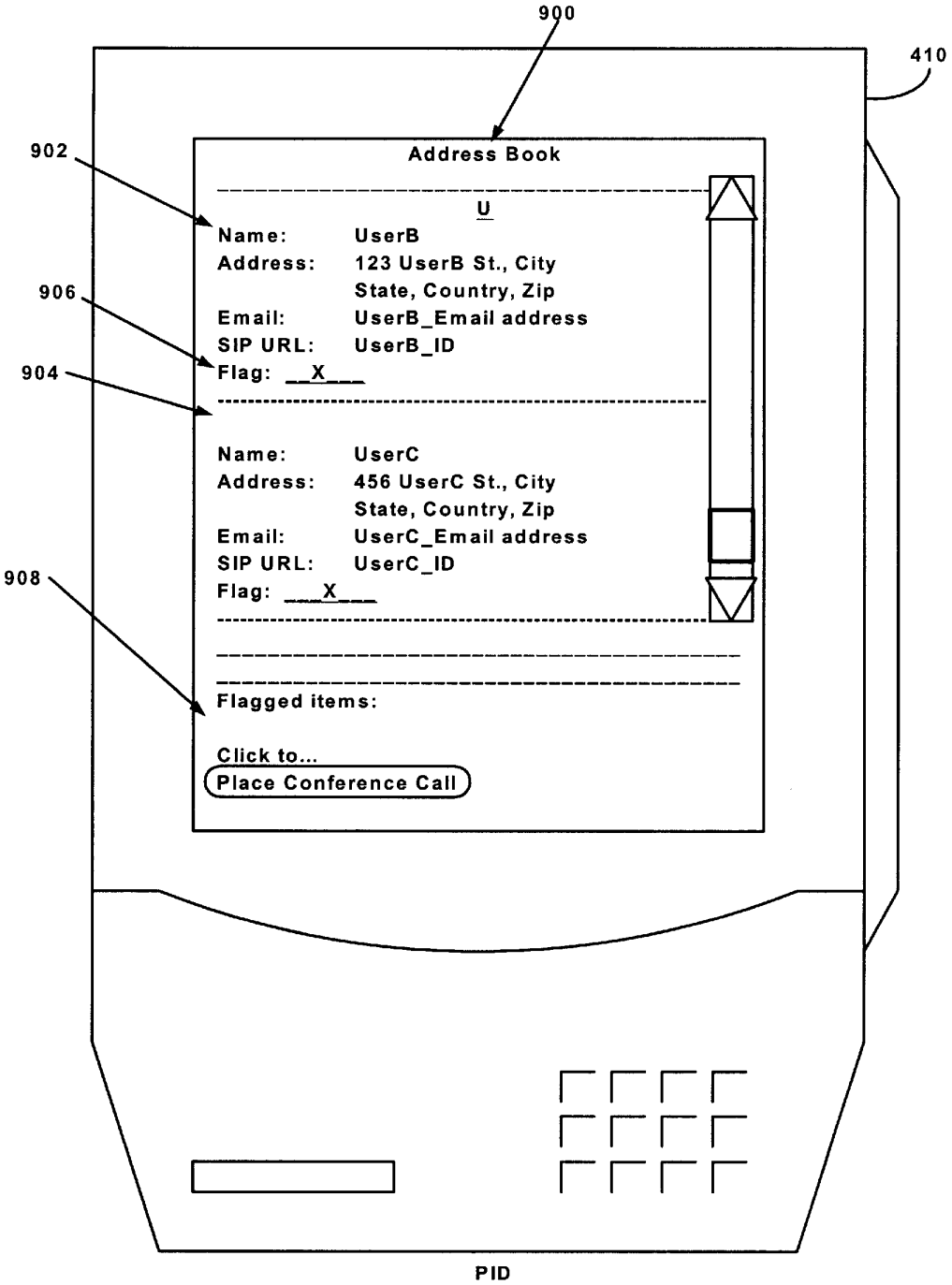

900
410
902
906
904
908
Address Book
U
Name: UserB
Address: 123 UserB St., City
State, Country, Zip
Email: UserB_Email address
SIP URL: UserB_ID
Flag: __X___
Name: UserC
Address: 456 UserC St., City
State, Country, Zip
Email: UserC_Email address
SIP URL: UserC_ID
Flag: ___X___
Flagged items:
Click to...
Place Conference Call
PID

FIG. 13

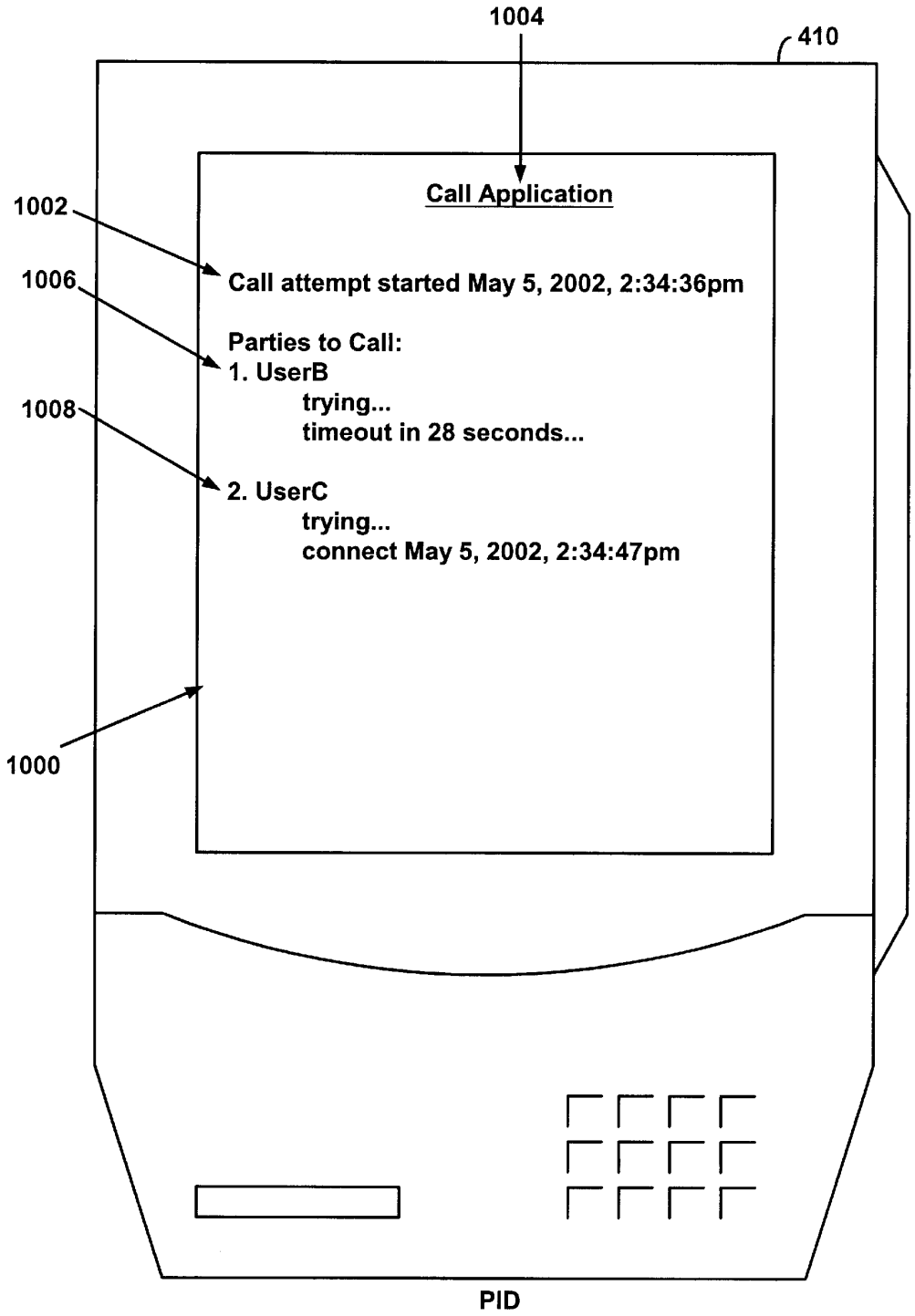

1004
410
Call Application
1002
Call attempt started May 5, 2002, 2:34:36pm
1006
Parties to Call:
1. UserB
trying...
timeout in 28 seconds...
1008
2. UserC
trying...
connect May 5, 2002, 2:34:47pm
1000
PID

FIG. 14

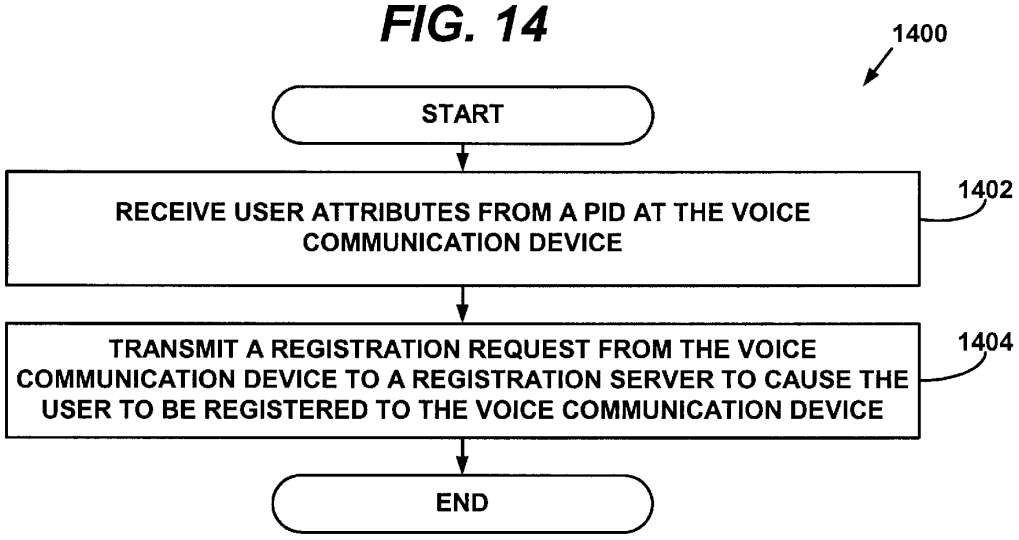
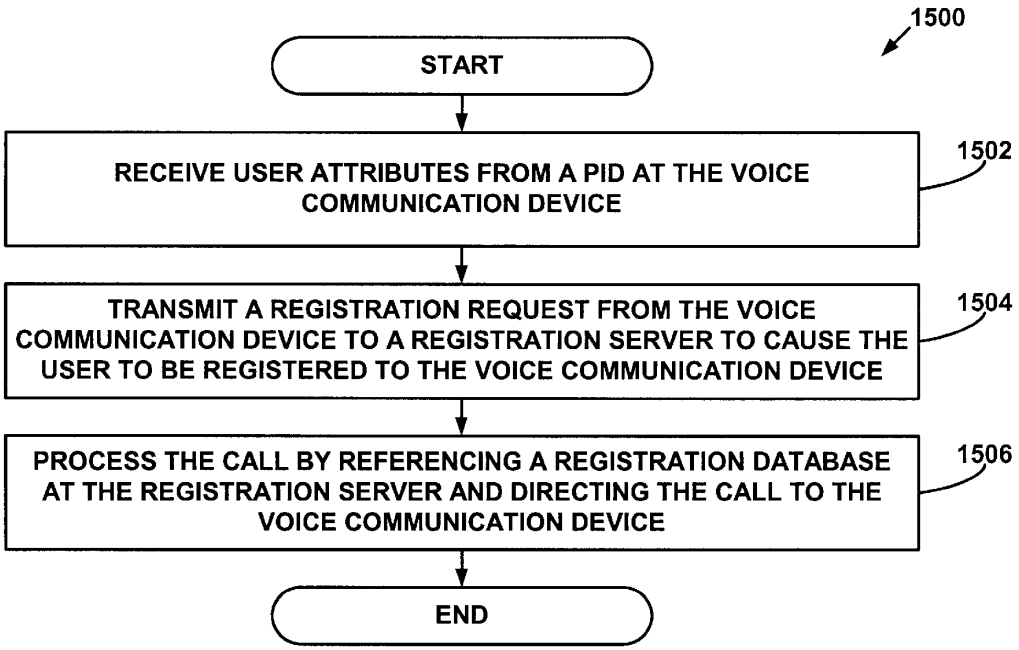

1400
START
RECEIVE USER ATTRIBUTES FROM A PID AT THE VOICE COMMUNICATION DEVICE
1402
TRANSMIT A REGISTRATION REQUEST FROM THE VOICE COMMUNICATION DEVICE TO A REGISTRATION SERVER TO CAUSE THE USER TO BE REGISTERED TO THE VOICE COMMUNICATION DEVICE
1404
END

FIG. 15


1500
START
RECEIVE USER ATTRIBUTES FROM A PID AT THE VOICE COMMUNICATION DEVICE
1502
TRANSMIT A REGISTRATION REQUEST FROM THE VOICE COMMUNICATION DEVICE TO A REGISTRATION SERVER TO CAUSE THE USER TO BE REGISTERED TO THE VOICE COMMUNICATION DEVICE
1504
PROCESS THE CALL BY REFERENCING A REGISTRATION DATABASE AT THE REGISTRATION SERVER AND DIRECTING THE CALL TO THE VOICE COMMUNICATION DEVICE
1506
END

SYSTEM AND METHOD FOR PROVIDING USER MOBILITY SERVICES ON A TELEPHONY NETWORK

PRIORITY

This application is a continuation-in-part of U.S. patent application Ser. No. 09/181,431, filed Oct. 30, 1998 now U.S. Pat. No. 6,161,134, by inventors Peter Si-Sheng Wang and Ismail Dalgic, titled "Method, Apparatus and Communications System for Companion Information and Network Appliances," assigned to 3Com Corporation.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention is related to a method and system for user mobility services on a network. In particular, the present invention relates to a method and system for providing user mobility services that can be configured by user-operated portable information devices.

2. Description of the Related Art

For many years, telephone service providers on the Public Switched Telephone Network (PSTN) provided their customers nothing more than a telephone line to use to communicate with other subscribers. Over time, telephone service providers have enhanced their service by providing Custom Local Area Signaling Service (CLASS) features to their customers. Similar communication services are provided by a Private Branch Exchange (PBX), which is typically implemented in a nonresidential setting.

The CLASS features permit customer subscribers of the features to tailor their telephone service according to individual needs. Some of the more popular CLASS features are:

- Call blocking: The customer may specify one or more numbers from which he or she does not want to receive calls. A blocked caller will hear a rejection message, while the callee will not receive any indication of the call.
- Call return: Returns a call to the most recent caller. If the most recent caller is busy, the returned call may be queued until it can be completed.
- Call trace: Allows a customer to trigger a trace of the number of the most recent caller.
- Caller ID: The caller's number is automatically displayed during the silence period after the first ring. This feature requires the customer's line to be equipped with a device to read and display the out-of-band signal containing the number.
- Caller ID blocking: Allows a caller to block the display of their number in a callee's caller ID device.
- Priority ringing: Allows a customer to specify a list of numbers for which, when the customer is called by one of the numbers, the customer will hear a distinctive ring.
- Call forwarding: A customer may cause incoming calls to be automatically forwarded to another number for a period of time.

A customer subscriber to a CLASS feature may typically activate and/or deactivate a CLASS feature using "*" directives (e.g., *69 to automatically return a call to the most recent caller). CLASS features may also be implemented with the use of out-of-band data. CLASS feature data is typically transmitted between local Class-5 switches using the Signaling System #7 (SS7).

Local Exchange Carriers (LECs) and other similar organizations maintain CLASS offices that typically contain a database entry for each customer. The database allows specification of the CLASS features a customer has subscribed to, as well as information, such as lists of phone numbers, associated with those features. In some cases, customers may edit these lists on-line via a touch-tone interface. A list of all phone numbers that have originated or terminated a call with each customer is often included in the CLASS office database. For each customer, usually only the most recent number on this list is stored by the local Class-5 switch.

A Private Branch Exchange (PBX), is a stored program switch similar to a Class-5 switch. It is usually used within a medium-to-large-sized business for employee telephony service. Since a PBX is typically operated by a single private organization, there exists a wide variety of PBX services and features. Custom configurations are common, such as integration with intercom and voice mail systems. PBX's typically support their own versions of the CLASS features, as well as other features in addition to those of CLASS. Most PBX features are designed to facilitate business and group communications.

A summary of typical PBX features includes:

- Call transfer: An established call may be transferred from one number to another number on the same PBX.
- Call forwarding: In addition to CLASS call forwarding, a PBX number can be programmed to automatically transfer a call to another number when the first number does not answer or is busy.
- Camp-on queuing: Similar to PSTN call return, a call to a busy number can be queued until the callee can accept it. The caller can hang up their phone and the PBX will ring them when the callee answers.
- Conference calling: Two or more parties can be connected to one another by dialing into a conference bridge number.
- Call parking: An established call at one number can be put on hold and then reestablished from another number. This is useful when call transfer is not warranted.
- Executive override: A privileged individual can break into an established call. After a warning tone to the two participants, the call becomes a three-way call.

While the CLASS and PBX features have enhanced the offerings of service providers that use the PSTN, the features are nevertheless limited in their flexibility and scope. The effect to the user is that the features become clumsy and difficult to use. For example, in order to use the Call Forwarding function, the user must perform the steps at the user's own phone prior to moving to the location of the telephone to which calls will be forwarded. A more desirable approach, from the standpoint of usefulness to the user, would be to perform the steps at the telephone to which calls will be forwarded.

Much of the lack of flexibility of the PSTN features is due to the lack of flexibility in the PSTN system itself. One problem with the PSTN is that the terminal devices (e. g. telephones) lack intelligence and operate as "dumb" terminals on a network having the intelligence in central offices. Most PSTN telephones are limited in functional capability to converting the analog signals they receive to sound, converting the sound from the handset to analog signals, generating the appropriate dial tones when a key on the keypad is pressed, and ringing when there is an incoming call.

Some PSTN telephones have a display device and a display function to display specific information communi-

cated from intelligent agents in the PSTN network using the PSTN signaling architecture. For example, some PSTN telephones have a display function to enable the Caller ID feature. Even such PSTN telephones are limited however by the closed PSTN signaling architecture, which prohibits access by the PSTN telephones to the network signaling protocols. A PSTN telephone having a display function is effectively limited to displaying text, again, as a "dumb" terminal.

The Internet presents a possible solution for distributing intelligence to telephony terminal devices. In Internet telephony, digitized voice is treated as data and transmitted across a digital data network between a telephone call's participants. One form of Internet telephony uses a telephony gateway/terminal where IP telephony calls are terminated on the network. PSTN telephones are connected by a subscriber line to the gateway/terminal at the local exchange, or at the nearest central office. This form of Internet telephony provides substantial cost savings for users. Because the PSTN portion used in Internet telephony calls is limited to the local lines on each end of the call, long distance calls may be made for essentially the cost of a local call. Notwithstanding the costs savings provided by this form of Internet telephony, it is no more flexible than the PSTN with respect to providing enhancements and features to the basic telephone service.

In another form of Internet telephony, telephones are connected to access networks that access the Internet using a router. The telephones in this form of Internet telephony may be substantially more intelligent than typical PSTN telephones. For example, such a telephone may include substantially the computer resources of a typical personal computer.

Data network telephones and the data network (e.g. Internet) system in which they operate, however, lack a substantial infrastructure and service providers for providing telephone service.

It would be desirable to incorporate CLASS and PBX features into a data network telephony system that uses a data network such as the Internet.

It would be desirable to provide new features and enhancements to telephony service that accommodate and conform to users' needs.

It would also be desirable to provide features and capabilities to telephone service that create new opportunities for users and for service providers.

It would also be advantageous to allow a user to utilize information stored in a user's Portable Information Device (PID), e.g. a Personal Digital Assistant (PDA), to assist in providing user mobility services.

SUMMARY OF THE INVENTION

The present invention is directed toward a system and method for interfacing a personal information device (PID), such as a PDA, with a network device, such as a data network telephone, to provide user access to voice and data network services. Applications on the PID, in combination with the network device and network servers, provide for a variety of user services, including user mobility services.

One aspect of the present invention is user portability for a user identifier, such as a user's telephone number or character sequence. By synchronizing user identification information contained on a PID with a data network telephone, the owner of the PID registers with the data network telephone, which causes the user identification information to be registered in a database along with the address of the data network telephone. A calling device queries the database server for an entry corresponding to the user's identification information in order to obtain the address of the data network telephone where the user is registered. The call to the user is thus routed to the data network telephone where the PID user was registered most recently. Unlike conventional telephone systems, the user's identification is separate from a particular physical address.

In another aspect of the present invention, the PID may store and download to the phone the preferences of the user about the phone operation, such as the ringer volume and tone.

In yet another aspect of the present invention, the PID may act as a smart card, providing authentication information for making toll calls.

In still another aspect of the present invention, the user may program the system through the PID so that depending on the time of day, and on the date-book information in the PID, the phone forwarding information is dynamically updated. For example, during business hours, the default location to forward calls could be set to be the user's office, and during other hours, the user's cellular phone or their pager.

In one embodiment of the invention, a system for providing user mobility services on a data network telephony system is disclosed. A data network provides data connectivity for a plurality of data communication channels using data transport protocols. A plurality of data network telephones within the data network may be used to communicate a voice signal as voice-over-data packets on a voice-over-data channel, where the voice-over-data channel is one of the plurality of data communication channels on the data network. A user, having a portable information device, may transfer user attributes from the portable information device to one of the data network telephones. The data network telephone receiving the user attributes transmits a registration request to a registration server to register the user with the data network telephone.

In another embodiment of the present invention, the data transmitted from the portable information device to the data network telephone includes communication partner data, allowing the user to invite at least one communications partner to a call across the data network.

In another embodiment of the invention, a system for providing user mobility services to a user associated with the portable information device includes a first set of machine language instructions for causing the portable information device to transmit user attributes to a first voice communication device, a second set of machine language instructions to cause the voice communication device to accept the user attributes and to transmit a registry request to a registration server, and a third set of machine language instructions to cause the registration server to accept the register request and to register the user associated with the portable information device to the data network telephone. A method for providing user mobility services at a voice communication device is also disclosed. According to one embodiment, the method includes the steps of receiving user attributes from a portable information device at the voice communication device and transmitting a registration request from the voice communication device to a registration server. The registration server registers the user to the voice communication device.

In another embodiment, the method could further include processing a call by referencing a registration data base maintained by the registration server and directing the call to the voice communication device to which the user is registered.

BRIEF DESCRIPTION OF THE DRAWINGS

Presently preferred embodiments of the invention are described below in conjunction with the appended drawing figures, wherein like reference numerals refer to like elements in the various figures, and wherein:

FIG. **1** is block diagram of a network telephony system according to one embodiment of the present invention;

FIG. **2** is a block diagram showing a system for providing user mobility services on a telephony network according to an exemplary embodiment of the present invention;

FIG. **3** is a block diagram showing a system for providing user mobility services on a telephony network according to a preferred embodiment of the present invention;

FIG. **4** is a block diagram showing a system for providing user mobility services on a telephony network according to an exemplary embodiment of the present invention;

FIG. **5** is a message flow diagram showing an exemplary SIP call setup operation;

FIG. **6** is a block diagram of a data network telephone according to an exemplary embodiment of the present invention;

FIG. **7** is a block diagram of a portable information device (PID) according to an exemplary embodiment of the present invention;

FIG. **8** is a stack layer diagram showing the layers of an IrDA stack;

FIG. **9** is a block diagram of a portable information device (PID) according to an alternative embodiment of the present invention;

FIG. **10** is a block and stack layer diagram illustrating the protocol stacks in an exemplary embodiment of a PID linked to a data network telephone;

FIG. **11** is block and stack layer diagram illustrating an embodiment of the present invention in which a SIP call may be established;

FIG. **12** is a pictorial diagram of an address book application on a PID according to an embodiment of the present invention;

FIG. **13** is a pictorial diagram of a call application on a PID according to an embodiment of the present invention;

FIG. **14** is a flow diagram illustrating a method for providing user mobility services at a voice communication device in a network according to an embodiment of the present invention; and

FIG. **15** is a flow diagram illustrating a method for providing user mobility services at a voice communication device in a network according to an embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The following references to patent applications are incorporated by reference herein:

- "Method Apparatus and Communication System for Companion Information and Network Appliances" to Wang, et al., Ser. No. 09/181,431.
- "System and Method for Controlling Telephone Service Using a Wireless Personal Information Device" to Schuster, et al.
- "System and Method for Advertising Using Data Network Telephone Connections" to Schuster, et al.
- "System and Method for Providing User-Configured Telephone Service in a Data Network Telephony System" to Sidhu, et al.
- "System and Method for Accessing a Network Server Using a Portable Information Device Through a Network Based Telecommunication System" to Schuster, et al.
- "System and Method for Interconnecting Portable Information Devices Through a Network Based Telecommunication System" to Schuster, et al.
- "System and Method for Enabling Encryption on a Telephony Network" to Schuster, et al.
- "System and Method for Associating Notes with a Portable Information Device on a Network Telephony Call" to Schuster, et al.
- "System and Method for Providing Shared Workspace Services Over a Telephony Network" to Schuster, et al.
- "System and Method for Providing Service Provider Configurations for Telephones in a Data Network Telephony System" to Schuster, et al.
- System and Method for Using a Portable Information Device to Establish a Conference Call on a Telephone Network" to Schuster, et al.
- "Multiple ISP Support for Data Over Cable Networks" to Ali Akgun, et al.
- "Method and System for Provisioning Network Addresses in a Data-Over-Cable System" to Ali Akgun, et al., Ser. No. 09/218,793.
- "Network Access Methods, Including Direct Wireless to Internet Access" to Yingchun Xu, et al., Ser. No. 08/887,313.

A. PID-Enabled Data Network Telephony System

FIG. **1** is a block diagram showing an exemplary embodiment of a system **100** for providing user mobility services on a telephony network according to the present invention. The system includes a data network **106**. A first voice communication device **108** linked to a first access network **112** via connection **111** may communicate over the data network **106** by connecting via the first access network **112**. A second voice communication device **118** is linked to a second access network **114** through connection **119** and may communicate over the data network **106** by connecting via the second access network **114**.

The data network **106** in the system **100** typically includes one or more Local Area Networks (LANs) connected to one another or to a Wide-Area Network (WAN), such as an Internet Protocol (IP) network, to provide wide-scale data connectivity. The data network **106** may use Voice-Over-Packet (VOP) schemes in which voice signals are carried in data packets. The network **106** may also include a connection to the Public Switched Telephone Network (PSTN) to allow for voice connections using traditional circuit switching techniques. In one embodiment, the data network **106** may include one or more LANs such as Ethernet LANs and support data transport protocols for performing Voice-over-Internet-Protocol (VoIP) techniques on the Internet. For further details regarding VoIP, see the information available through the Internet Engineering Task Force (IETF) at www.ietf.org. In addition, an Internet Telephony gateway may be included within the system **100** to allow for voice connections to users connected by subscriber lines at a PSTN Central Office. Other data besides voice data may also be communicated over the data network **106**.

The voice communication devices **108** and **118** typically include a voice input, a voice output, and a voice processing system and may be data network telephones (described further below with reference to FIG. **6**). The voice processing system converts voice sound to digital data signals that are communicated on a voice connection over the data

network. The voice processing system also converts digital data signals received from the voice connection to voice sound. The voice communication devices **108** and **118** typically include a central processing unit and memory to store and process computer programs. Additionally, each voice communication device typically includes a unique network address, such as an IP address, in memory to uniquely identify it to the data network **106** and to permit data packets to be routed to the device.

A PID **110** is shown linked to the first voice communication device **108** via link **109**, and may enable communications over the data network **106** via the first access network **112**. The PID **110** includes user attributes stored in a user information database. The user attributes may contain such information as a user identifier, schedule information, information about contacts, and other information that is associated with a user of the PID **110**. The PID **110** preferably includes a user interface allowing a user to easily enter and retrieve data. In a preferred embodiment, the user interface includes a pressure-sensitive display that allows a user to enter input with a stylus or other device. An example of a PID with such an interface is a PDA (Personal Digital Assistant), such as one of the Palm™ series of PDAs offered by 3Com® Corporation. Alternatively, the PID **110** may be a form of smart card, in which the user attributes are programmed into the card with the assistance of a programming device. In such a case, the user attributes might not be easily modified by the user when the user is not in the presence of the programming device. The PID **110** may include other functionality, such as wireless phone, two-way radio, digital camera, or digital audio recording functionality, for example.

Link **109** is a point-to-point link, and may be entirely or partially wireless, or may be a hard-wired connection. Preferably, the link **109** is a wireless link, such as an infrared link specified by the Infrared Data Association (IrDA) (see irda.org for further information) or a radio frequency (RF) link such as the Bluetooth system (see www.bluetooth.com for further information). However, the point-to-point link can also be a hardwired connection, such as an RS-232 or Universal Serial Bus (USB) serial port connection. An example of a serial port connection is a docking cradle or a synchronizing cable connection.

In one embodiment, the voice communication devices **108** and **118** each include a handset with a receiver and transmitter similar or identical to handsets of traditional circuit-switched telephones. A console on which the handset sits may include the voice processing system, a display, and a keypad, for example.

In a preferred embodiment, a portion of each of the voice communication devices **108** and **118** utilizes an NBX 100™ communication system phone offered by 3Com® Corporation. In alternative embodiments, the voice communication devices **108** and **118** may include any device having voice communications capabilities. For example, a personal computer having a microphone input and speaker output may also be used to implement the voice communication devices **108** and **118**. Other configurations are also intended to be within the scope of the present invention.

The details relating to operation of the voice communication devices **108** and **118** depend on the nature of the data network **106** and the nature of the access networks **112** and **114** connecting the voice communication devices **108** and **118** to each other and/or to other network entities. The access networks **112** and **114** typically include any high bandwidth network adapted for data communications, i.e. a network having greater than 64,000 bits-per-second (bps) bandwidth. The access networks **112** and **114** may link to the voice communication devices **108** and **118** using an Ethernet LAN, a token ring LAN, a coaxial cable link (e.g. CATV adapted for digital communication), a digital subscriber line (DSL), twisted pair cable, fiberoptic cable, an Asynchronous Transfer Mode (ATM) link, an integrated services digital network (ISDN) link, and wireless links, for example. In embodiments that may not require bandwidth greater than 64,000 bps, the access networks **112** and **114** may also include the PSTN and link the voice communications devices **108** and **118** by an analog modem. Further details regarding specific implementations are described below, with reference to FIGS. **2** through **13**.

B. System for Providing User Mobility Services on a Data Network Telephony System

One advantage of the network telephony system **100** is that it may be used to provide user mobility services to users of the network telephony system. In one embodiment, the PID **110** allows a user to select the communications partner or partners to be included in the call. The PID **110** then transfers information about the communication partner(s) to the first voice communication device **108** through the link **109**. The first voice communication device **108** then sets up the call with voice communication devices associated with the communication partners selected by the PID user associated with the first voice communication device **108**.

Once a call is set up, data can be transferred between the voice communication devices. PIDs, such as the PID **110**, associated with the parties to the call may also be used to communicate information. For example, the PID **110** linked to the first voice communication device **108** may be able to accept and display PID data entered by a user through a user interface on the PID **110**. The PID data can then be communicated across the link **109** to the voice communication device **108** for transport across the first access network **112**, the data network **106**, and the second access network **114** to the second voice communication device **118**. The PID **110** can also receive PID data and other data across the link **109** for display on the PID **110**. A voice-over-data channel for communicating voice-over-data can concurrently exist with this communication of PID data over a PID data channel. Preferably, all parties to the call have PIDs linked to the voice communication devices associated with the parties. In this way, a user of the PID **110** can communicate PID data to other parties to the call while voice signals are communicated between the voice communication devices.

1. Providing User Mobility Services on a Local Area Network

FIG. **2** is a block diagram showing a system **200** for providing user mobility services on a LAN according to one embodiment of the present invention. System **200** includes a registration server **202** having access to a registration database **204**. The registration server **202** is linked to a packet-switched local area network (LAN) **206**. A voice communication device **208** is also a part of the network **206**. Also shown are additional voice communication devices **212**, **214**, **216**, and **218**, which may or may not be identical to each other and voice communication device **208**. The voice communication devices **208**, **212**, **214**, **216**, and **218** are each preferably able to accept information from a PID **210**. A user **220** is shown as having recently moved from the voice communication device **212** to the voice communication device **208**. The PDA **210** is associated with the user **220**. The connections shown in FIG. **2** may be entirely or partially wireless, or they may be hard-wired connections. The LAN **206**, the voice communication device **208**, and the PID **210** correspond respectively to the first access network

112, the voice communication device 108, and the PID 110 shown in FIG. 1.

The LAN 206 is preferably an Ethernet LAN operating according to the IEEE 802.3 specification, which is incorporated by reference herein. The voice communication devices 208, 212, 214, 216, and 218 are preferably modified Ethernet phones. An Ethernet phone is a telephone capable of communicating through an Ethernet port.

In most cases, Ethernet phones support Internet Protocol (IP), using an IP address that is either statically configured or obtained via Dynamic Host Configuration Protocol (DHCP). An exemplary Ethernet phone, such as voice communication device 208, contains two basic parts: the signaling-stack and the media-engine. While currently two different standards (SIP and H.323) and several proprietary approaches exist for the signaling stack, the media is almost exclusively transported via the Real Time Protocol (RTP), which itself is carried inside of User Datagram Protocol (UDP). RTP is described in H. Schulzrinne et al., "RTP: A Transport Protocol for Real-Time Applications," IETF RFC 1889, January 1996, which is incorporated herein by reference. UDP is described in J. Postel, "User Datagram Protocol," IETF RFC 768, August 1980, and IP is described in J. Postel, ed., "Internet Protocol," IETF RFC 791, September 1981, both of which are incorporated by reference herein.

The purpose of the signaling stack in an exemplary Ethernet phone, such as the voice communication device 208, is to set up, manage, and tear down a call. During the setup phase, the location of the endpoint is discovered, communication parameters, such as the supported voice CODEC types are determined, the voice channel is established, and other parties are invited to the call if needed. During the management phase, for example, other parties are invited to the call or the existing CODEC can be changed. During the teardown phase, the call is terminated. The preferred call-management protocol for the present invention is Session Initiation Protocol (SIP), which is described in M. Handley et al., "SIP: Session Initiation Protocol," IETF RFC 2543, March 1999, incorporated by reference herein. Alternative call-management protocols, such as the ITU-T H.323 protocol and others, may also be used to implement the present invention.

The purpose of a media engine in an exemplary Ethernet phone is to sample the voice, encode the samples, and build the RTP packets on the sending side. On the receiver side, in addition to performing the reverse operations, the media engine also typically manages a receiver buffer to compensate for network jitter. The media engine includes the features discussed with reference to the user interface of the voice communication device 108.

The user 220 is shown as being recently relocated from voice communication device 212 to voice communication device 208. In the example illustrated by FIG. 2, voice communication device 212 may be in the user's office and voice communication device 208 may in be a conference room, for example. Prior to leaving the office, the user 220 may have been registered (associated) with the voice communication device 212 in the office, so that calls were routed to that location or were processed according to attributes associated with the user being located in the office. When the user moves to the voice communication device 208 in the conference room, it would be desirable for the user to be registered with the voice communication device 208 instead of the voice communication device 212. According to one embodiment of the present invention, the user 220 is able to register with the voice communication device 208 by using a portable information device 210 to transmit user attributes to the voice communication device 208, which may then transmit all or some of the user attributes to the registration server 202, so that the registration database 204 may be updated with the revised user communication-location information. If the PID 210 is a PDA, the process may be initiated by synchronizing the PDA 210 with an Ethernet telephone, for example.

2. Providing User Mobility Services on a LAN using the Session Initiation Protocol

FIG. 3 is a block diagram showing an exemplary user mobility system 300 according to a preferred embodiment of the present invention, in which SIP is used as the call-management protocol. Portions of the system 300 are similar to the system 200 illustrated in FIG. 2. The system 300 includes a SIP server 302 having access to a SIP database 304. The SIP server is shown with a link to a LAN 306, which is preferably an Ethernet LAN. SIP phones 308, 312, 314, 316, and 318 are Ethernet phones, and are also linked to the LAN 306. A PDA 310 serves as a PID for customizing the communication system 300 according to a preferred embodiment of the present invention. The number of SIP phones in the system 300 can vary to meet the needs of the users of the system 300.

Also shown in the LAN 306 is a gateway 322 with a SIP client. The gateway 322 is preferably a VoIP gateway and is in communication with a PSTN central office 324, which provides PSTN service to a PSTN phone 326. The PSTN phone 326 is likely to be one of many PSTN phones serviced by the central office 324. Additional portions of a PSTN network have been omitted from FIG. 3 to improve clarity. The PSTN network is well known by those having skill in the art of telecommunications. The gateway 322, the central office 324, and the PSTN 326 are optional and need not be included within the system 300.

A router 328 may also be connected to the LAN 306. The router 328 connects the LAN 306 to a data network 330, such as a public internet. The data network preferably includes connections to additional SIP-based clients, such as additional SIP phone 332 and a personal computer 334 operating as a SIP client. SIP will be described in more detail with reference to FIGS. 4, 5, 10, and 11. The router 328, the data network 330, and the SIP-based clients 332 and 334 are optional and need not be included within the system 300.

3. Local Area Network as an Exemplary Access Network

FIG. 4 is a block diagram showing one example of the system 100 of FIG. 1 for providing user mobility services according to the present invention. The system 400 in FIG. 4 includes a local area network 412 connected to a data network 406 by a first router 413. A second local area network 414 is connected to the data network 406 by a second router 415. A cable network 416 is connected to the data network 406 by a third router 417. Those of ordinary skill in the art will appreciate that while FIG. 4 illustrates the access networks as two local area networks 412 and 414, and a cable network 416, other types of networks may be used. For example, the local area networks and the cable network may be replaced by ISDN, DSL, or any other high-speed data link.

The local area networks 412 and 414 provide data connectivity to their respective network elements. For example, the first LAN 412 provides data connectivity to at least a first data network telephone 408 and a first network telephony connection server 450. The second LAN 414 provides data connectivity to at least a second data network telephone 418 and a second network telephony connection server 438.The local area networks 412 and 414 in FIG. 4 are, for example,

Ethernet LANs operating according to the IEEE 802.3 specification, which is incorporated by reference herein; however, other types of local area networks may also be used. The first local area network **412** uses the router **413** to provide the first data network telephone **408** and the first network telephony connection server **450** with access to the data network **406**. For example, the router **413** may perfonn routing functions using protocol stacks that include the Internet Protocol and other protocols for communicating on the Internet. Similarly, the second local area network **414** uses the router **415** to provide the second data network telephone **418** and the second network telephony connection server **438** with access to the data network **406**.

The first, second, and third network telephony connection servers **450**, **438**, and **437** provide telephony registration, location, and session initiation services for voice connections in which at least one of their members is a party. For example, a user of the first data network telephone **408** may register for telephony service with an administrator of the first network telephony connection server **450** and receive a user identifier and a telephone identifier. The user identifier and telephone identifier may be sequences of unique alphanumeric elements that callers use to direct voice connections to the user. The network telephony connection servers register users by storing user records in registration databases (not shown in FIG. **4**) associated with each of the network telephony connection servers, in response to registration requests.

The call setup process and the user and telephone identifiers preferably conform to requirements defined in a call-management protocol. The call-management protocol is used to permit a caller anywhere on the data network to connect to the user identified by the user identifier in a data network telephone call. A data network telephone call includes a call setup process and a voice exchange process. The call setup process includes steps and message exchanges that a caller and callee perform to establish the telephone call. The actual exchange of voice signals is performed by a voice data communications channel. The voice data communications channel incorporates other data transport and data formatting protocols, and preferably includes well-known data communications channels typically established over the Internet.

The call management protocol used in FIG. **4** is the Session Initiation Protocol (SIP), which is described in M. Handley et al., "SIP: Session Initiation Protocol," IETF RFC 2543, March 1999, incorporated by reference herein; however, any other such protocol may be used. Other protocols include H.323, MEGACO, the Media Gateway Control Protocol (MGCP), etc.

The network telephony connection servers **450**, **438**, and **437** may be used to provide telephony service for mobile users. For example, a user may be registered to use the first network telephone **408** (which is identified by its telephone identifier), but the user may move to a location near a second network telephone (not shown) on the first local area network **412**. The user may re-register as the user of the second network telephone. The user would then become associated with the second network telephone. Calls that identify the user by the user's user identifier may then reach the user at the second network telephone.

4. Cable Network as an Exemplary Access Network

The system **400** in FIG. **4** also shows the cable network **416** connected to the data network **406** by a router **417**. The cable network **416** provides data network access to its network elements, which in FIG. **4** include the third data network telephone **428** and the third network telephony connection server **437**. A user of the third data network telephone **418** connected to the cable network **416** may communicate by telephone over the data network **406** with the users of the first and second data network telephones **408** and **418** connected to the first and second local area networks **412** and **414**.

The cable network **416** may include any digital cable television system that provides data connectivity. In the cable network **416**, data is communicated by radio frequency in a high-frequency coaxial cable. The cable network **416** may include a headend and/or a central termination system that permits management of the cable connections to the users.

5. Providing Telephony Services

The third network telephony connection server **437** is preferably a SIP-based server that performs call initiation, maintenance, and teardown for the third data network telephone **428** connected to the cable network **416**. The third network telephony connection server **437** may be similar or identical to the first and second network telephony connection servers **450** and **438** connected to the first and second local area networks **412** and **414**.

The system **400** shown in FIG. **4** includes a data network telephony system that permits the first and second data network telephones **408** and **418** connected to the local area networks **412** and **414** to communicate through the data network **406** with the third data network telephone **428** connected to the cable network **416**. The system shown in FIG. **4** uses SIP in order to establish, maintain, and tear down telephone calls between users.

There are two major architectural elements to SIP: the user agent (UA) and the network server. The UA resides at the SIP end stations, (e.g. the data network telephones), and contains two parts: a user agent client (UAC), which is responsible for issuing SIP requests, and a user agent server (UAS), which responds to such requests. There are three different network server types: a redirect server, a proxy server, and a registrar. The various network server types may be combined into a single server, such as the network telephony connection servers **450**, **437**, and **438**. Not all server types are required to implement the various embodiments of the present invention. The communication services to be provided will determine which servers are present in the communication system. Preferred embodiments of the present invention may be carried out using proxy servers.

One example of a SIP operation involves a SIP UAC issuing a request, a SIP proxy server acting as end-user location discovery agent, and a SIP UAS accepting the call. A successful SIP invitation consists of two requests: INVITE followed by ACK. The INVITE message contains a user identifier to identify the callee, a caller user identifier to identify the caller, and a session description that informs the called party what type of media the caller can accept and where it wishes the media data to be sent. User identifiers in SIP requests are known as SIP addresses. SIP addresses are referred to as SIP Uniform Resource Locators (SIP-URLs), which are of the form sip:user@host.domain. Other addressing conventions may also be used.

FIG. **5** is a message flow diagram showing an exemplary SIP call setup operation **500**. A SIP caller UAC **502** sends an INVITE message **504** to a SIP callee UAS **506**. (The proxy server is not shown in this illustration). The INVITE message **504** contains session description information (UAC SDP) for the caller UAC **502**. The callee UAS **506** sends a 200-OK message **508** to the caller UAC **502**. The 200-OK message **508** contains session description information (UAS SDP) for the callee UAS **506**. The caller UAC **502** sends an

ACK message **510** to the callee UAS **506** to complete the session initiation operation.

Redirect servers may be used to process an INVITE message by sending back the SIP-URL where the callee is reachable. Proxy servers perform application layer routing of the SIP requests and responses. A proxy server can either be stateful or stateless. A stateful proxy holds information about the call during the entire time the call is up, while a stateless proxy processes a message without saving information contained in the message. Furthermore, proxies can either be forking or non-forking. A forking proxy can, for example, ling several phones at once until somebody takes the call. Registrar servers are used to record the SIP address (called a SIP URL) and the associated IP address. The most common use of a registrar server is for the UAC to notify the registrar where a particular SIP URL can be reached for a specified amount of time. When an INVITE request arrives for the SIP URL used in a REGISTER message, the proxy or redirect server forwards the request correctly.

At the first local area network **412**, the central registrar/proxy server, such as the first network telephony connection server **450**, is the primary destination of all SIP messages trying to establish a connection with users on the first local area network **412**. Preferably, the first network telephony connection server **450** is also the only destination advertised to the SIP clients outside the first local area network **412** on behalf of all the SIP clients residing on the first local area network **412**. The network telephony connection server **450** relays all SIP INVITE messages to the appropriate final destination (or another SIP proxy), based on a database lookup using a first SIP database (not shown) associated with the first network telephony connection server **450**. This allows mobile clients to be registered to their current locations.

Similarly, the second network telephony connection server **438** is the primary destination of all SIP messages trying to establish a connection with SIP clients such as the data network telephone **418**, connected to the second local area network **414**. Preferably, the second network telephony connection server **438** is also the only destination advertised to the SIP clients outside the second local area network **414** on behalf of all the SIP clients (e.g. data network telephones) residing on the second local area network **414**. The second network telephony connection server **438** relays all SIP INVITE messages to the appropriate final destination (or another SIP proxy), based on a database lookup using a second SIP database. The third network telephony server **437** behaves similarly to the first and second network telephony connection servers **450** and **438**. The use of three servers is for illustrative purposes only, and other server configurations may also be used.

The data network telephones **408**, **418**, and **428** in the system **400** preferably have pre-programmed device identifiers (e.g. phone numbers), represented as SIP-URL's that are of the form sip:user@domain. An example is sip:1234567890@3Com.com. After power-up, each of the data network telephones **408**, **418**, and **428** sends a SIP REGISTER message to the default registrar, such as the network telephony servers **450**, **438**, and **437**. When a call arrives at one of the network telephony servers **450**, **438**, or **437** for any of the registered SIP URLs, the server will forward the call to the appropriate destination. If a data network telephone is moved to a new location, all calls to the associated SIP URL will still be properly routed to that device. In other words, the system in FIG. **4** provides device mobility in the sense that calls will "follow" the data network telephone according to its SIP URL. This is especially useful if the data network telephone **408**, **418**, or **428** is running the DHCP (Dynamic Host Configuration Protocol) so that when the location is changed, the IP address is also automatically changed.

An advantage of the system in FIG. **4** is that once the call is established between data network telephones, the data network **406** provides data connectivity for up to a plurality of data communications channels. For example, the data network telephones **408**, **418**, and **428** can communicate voice signals as voice-over-data packets on a voice-over-data channel. The data network telephones **408**, **418**, and **428** may also be able to communicate PID data as PID data packets on a PID data channel. An example of PID data is graphical drawing data that is input into a PDA with a stylus device. Another example of PID data is one of the call participants' contact information, which may be passed on to the other participants. Other data types may also be communicated. If PID data is input into the PID **410**, the PID data may be communicated to and from the PID **410** across link **409** to the data network telephone **408**, where the PID data may be assembled into packets and disassembled from packets as part of the process for communicating the PID data packets across the data network **406** and any access networks, such as the first Ethernet LAN **412**, the second Ethernet LAN **414**, and the cable network **416**. For example, the PID data may be communicated to and from at least one other PID (not shown) through a network device (such as a data network telephone) located in the system **400**.

6. The Data Network Telephones

The data network telephones **408**, **418**, and **428** are preferably telephones that include an Ethernet communications interface for connection to an Ethernet port. The Ethernet phones in FIG. **4** support the Internet Protocol (IP), using an IP address that is either statically configured or obtained by access to a Dynamic Host Configuration Protocol (DHCP) server.

FIG. **6** is a block diagram showing the first data network telephone **408** connected to the local area network **412** in FIG. **4**. The voice communication devices **108**, **118**, **208**, **212**, **214**, **216**, **218** may be implemented using the concepts shown in FIGS. **4** and **6**. The data network telephone **408** in FIG. **6** is connected to the LAN **412** by a network interface **600**. The network interface **600** may, for example, be a network interface card, and may be in the form of an integrated circuit. A bus **602** may be used to connect the network interface **600** with a processor **604** and a memory **606**. Also connected to the processor are user interface circuitry **608** and three alternative interfaces **610**, **612**/**614**/**616**, and **618**/**620**/**622** to a PID, such as the first PID **410**.

The first alternative interface **610** includes an RS-232 serial connection and associated coupling hardware and mechanisms. The first alternative interface **610** may, for example, be a docking cradle or cable for a PID, such as a PDA (Personal Digital Assistant), in which information may be transferred between the PID and the first data network telephone **408**.

The second alternative interface comprises a first connection **612**, such as an RS-232 connection, along with infrared circuitry **614** for converting signals into infrared output and for accepting infrared input. An infrared interface **616** may also be included within the second alternative interface.

The third alternative interface comprises a first connection **618**, such as an RS-232 connection, along with radio-frequency circuitry **620** for converting signals into radio frequency output and for accepting radio frequency input. A radio frequency interface **622** may also be included as part of the third alternative interface.

The three alternative interfaces described above are merely examples, and additional means for implementing the interface between the data network telephone **408** and the PID may also be used. Although three interfaces are shown in FIG. **6**, there may be only one such interface in the data network telephone **408**. More than one interface may be included to improve flexibility and to provide redundancy in case of failure of an interface.

The user interface circuitry **608** includes hardware and software components that access the functions of the handset, display, and keypad to provide user input and output resources for functions in the processor **604**. It is important to note that none of the handset, display, and keypad need be present in the data network telephone **408**, and alternative user interface mechanisms may be used. As an example, the user interface circuitry may include a display interface **624**, a keypad interface **626**, an audio output interface **628**, and an audio input interface **630**.

For some applications, the user interface circuitry **608** may only need to support sending or receiving, but not both. The user interface circuitry **608** preferably supports the sending and receiving of at least audio information. For example, in the case where the data network telephone **408** is a voice communication device, the user interface circuitry may include a microphone, a speaker, and analog interface circuitry. A videophone implementation might also include a camera and monitor. The data network telephone **408** is not limited to telephones or videophones—additional user interface types, for example, such as the ones needed for computer games, (e.g. a joystick, or virtual reality headset) are also contemplated as being within the scope of the present invention.

The audio input interface **630** may receive voice signals from a microphone or other audio input device and convert the signals to digital information. The conversion preferably conforms to the G.711 ITU-T Standard. Further processing of the digital signal may be performed in the audio input interface **630**, such as to provide compression (e.g. using the ITU-T G.723.1 standard) or to provide noise reduction, although such processing may also be performed in the processor **604**. Alternatively, the audio input interface **630** may communicate an analog voice signal to the processor **604** for conversion to digital information.

The audio output interface **628** receives digital information representing voice from the processor **604** and converts the information to sound. In one embodiment, the audio output interface **628** receives information in the form of G.711 although other processing such as decompression may be performed in the audio output interface **628**. Alternatively, the processor **604** may convert digital information to analog voice signals and communicate the analog voice signals to the audio output interface **628**.

The keypad interface **626** and the display interface **624** include well-known device interfaces and respective signal processing techniques. The user interface circuitry **608** may support other hardware and software interfaces.

The processor **604** may consist of one or more smaller processing units, including, for example, a programmable digital signal processing engine. In the preferred embodiment, the processor is implemented as a single ASIC (Application Specific Integrated Circuit) to improve speed and to economize space. The processor **604** also may include an operating system, and application and communications software, firmware, or hardware for implementing the functions of the first data network telephone **408**. For example, the processor may include a conferencing application to assist a user of the data network telephone **408** in gathering communication partner data from a PID and to establish a conference call by connecting the conference call parties. Other applications may also be processed by the data network telephone **408**. The operating system may be any suitable commercially available embedded or disk-based operating system, or any proprietary operating system.

The processor **604** preferably includes a media engine **634** and a signaling stack **636** to perform the primary communications and application functions of the data network telephone **408**. The purpose of the signaling stack in an exemplary data network telephone **408** is to set up, manage, and tear down a call. During the setup phase, a user may use the keypad to enter a user identifier to call. The signaling stack **636** receives the user entry and formats a request message to send to the user identified by the user identifier to initiate a telephone call. When the request message is sent, the location of the user identified by the user identifier is discovered, communication parameters, such as the supported voice CODEC types are exchanged, and a voice over data channel is established. During the management phase, for example, other parties are invited to the call if needed. During the teardown phase, the call is terminated.

The call-management protocol used in the exemplary data network telephone **408** in FIG. **3** is the SIP protocol. In particular, the signaling stack implements a User Agent Client **638** and a User Agent Server **640**, in accordance with the SIP protocol. Alternative call-management protocols, such as the ITU-T H.323 protocol and others, may also be used to implement the present invention.

Once the call is set up, the media engine **634** manages the communication over a data communications channel using a network transport protocol and the network interface **600**. The media engine **634** sends and receives data packets having a data payload for carrying data and an indication of the type of data is being transported. The media engine **634** in the data network telephones **408** may sample the voice signals from the audio input **630** (or receive voice samples from the audio input **630**), encode the samples, and build data packets on the sending side. On the receiver side, in addition to performing the reverse operations, the media engine also preferably manages a receiver buffer to compensate for network jitter.

The media engine **634** preferably includes hardware and software components for conferencing **632**, performing registration functions **642**, voice-over-data functions **644**, display data functions **646**, and keypad output functions **648**. The media engine **634** processes data that is received from the first local area network **412**, and data that is to be sent over the first local area network **412**.

For data that is received from the first local area network **412**, the media engine **634** may determine from the type of data in the packet whether packets contain sampled voice signals or data for performing other functions. For example, packet headers or trailers may contain an indication of data type. Packets containing sampled voice signals are processed by voice over data function **644**. The voice over data function **644** preferably conforms to a protocol for formatting voice signals as digital data streams. While any suitable protocol may be used, the media (voice signal) is preferably transported via the Real Time Protocol (RTP), which itself is carried inside of User Datagram Protocol (UDP). RTP is described in H. Schulzrinne et al., "RTP: A Transport Protocol for Real-Time Applications," IETF RFC 1889, January 1996, which is incorporated herein by reference. UDP is described in J. Postel, "User Datagram Protocol," IETF RFC 768, August 1980, and IP is described in J. Postel, ed., "Internet Protocol," IETF RFC 791, September 1981, both of which are incorporated by reference herein.

Packets containing data for use in registering the data network telephone **408** with a network telephony service are processed by the registration function **642**. By registering the data network telephone **408**, a user may establish with the network telephony service provider that calls addressed to the user's user identifier may be connected to the data network telephone **408**. Registration may occur when the data network telephone **408** sends a request to register to a service provider host. The service provider host may respond by setting the user's user identifier to correspond to the telephone identifier of the data network telephone **408**, and by acknowledging the request with a status message to the data network telephone **408**. In one embodiment, a request to register the data network telephone **408** to a default user is automatically sent during power-up of the data network telephone **408**. As a result, the user becomes associated with the data network telephone **408**.

Other features may be added to the registration functions **642**, or implemented as extensions to the registration functions **642**. For example, the data network telephone **408** may be provisioned to provide selected network telephony services by establishing a data connection with a service provider, requesting the selected services, and receiving data that ensures that the services have been successfully provisioned. Such services may include, for example, caller identification, call forwarding, voice mail, and any other service offered by the network telephony service provider to enhance the capabilities of the data network telephone **408**. One advantage of such provisioning functions is that services may be ordered for temporary use in a manner that is convenient to the user.

Packets containing data for display on the display device are processed by the display data function **646**. The display data function **646** may be used for displaying, for example, the name(s) and user identifier(s) of the other party(-ies) to the call, the status of the telephone call, billing information, and other information

For data that is to be sent over the data network **406**, the media engine **634** formats the data as data packets in accordance with a selected protocol. The placement of data into packets may also be performed elsewhere in the data network telephone **408**. The selected protocol is preferably the protocol that is supported by the data network telephone that will receive the data for the particular type of data being transported.

The voice-over-data function **644** formats voice samples according to the protocol used by the receiving data network telephone. In one preferred embodiment, the voice over data function **644** formats voice samples as RTP packets. The registration function **642** and the keypad output function **648** may control the transport of data that does not represent voice signals.

The second and third data network telephones **418** and **428** are preferably similar or identical to the first data network telephone **408**.

7. The Portable Information Device (PID)

FIG. **7** is a block diagram showing one embodiment of the PID **410**. The PID **410** may be linked to the data network telephone **408** through a point-to-point interface **700**. A bus **702** may be used to connect the point-to-point interface **700** with a processor **704**, a memory **706**, data storage **708**, and user interface circuitry **710**.

The point-to-point interface **700** shown in FIG. **4** illustrates three alternative interfaces **712**, **714**/**716**/**718**, and **720**/**722**/**724** to a data network telephone.

A first alternative interface **712** includes an RS-232 serial connection and associated coupling hardware mechanisms. The first alternative interface **712** may, for example, be a docking cradle or cable for a PID, in which information can be transferred between the PID **410** and the first data network telephone **408**.

The second alternative interface comprises a first connection **714**, such as a RS-232 serial connection, along with infrared circuitry **716** for converting signals into infrared output and for accepting infrared input. An infrared interface **718** may also be included within the second alternative interface.

The third alternative interface comprises a first connection **720**, such as an RS-232 connection, along with radio-frequency circuitry **722** for converting signals into radio frequency output and for accepting radio frequency input. A radio frequency interface **724** may also be included as part of the third alternative interface.

The three alternative interfaces described above are merely examples, and additional means for implementing the interface between the PID **410** and the data network telephone **408** may also be used. Although three interfaces are shown in FIG. **4**, there may be only one such interface in the PID **410**. More than one interface may be included to improve flexibility and to provide redundancy in case of failure of an interface.

The user interface circuitry **710** includes hardware and software components that provide user input and output resources for functions in the processor **704**. The user interface circuitry preferably includes a display output **726**, a display input **728**, and an additional input/output interface **730**.

The display output **726** preferably receives digital information representing graphical or other data from the processor **704** and converts the information, such as text and/or images, for display on a graphical display, such as an LCD or TFT screen.

The display input **728** may receive PID data inputs from a user of the PID **410**. The PID data inputs are preferably entered by the user with a stylus on a pressure-sensitive display screen. Alternately, a keyboard may be used to accept user input. Similarly, the display output **726** preferably displays the PID data on the display screen.

The additional input/output interface **730** allows the user to enter other types of data besides PID data into the PID **410**. For example, audio data, additional PID data, or additional input may be entered through the additional input/output interface **730**. Touch-sensitive screen buttons are an exemplary method for a user to enter control data into the PID **410**.

The processor **704** includes an operating system and application and communication software, firmware, or hardware to implement the functions of the PID **410**. The operating system may be any suitable commercially available operating system, or any proprietary operating system. The operating system and software may be stored on data storage **708**. An address book application **732** is preferably included to manage the user's contact information. Similarly, a scheduling application (not shown) may be included to manage the user's schedule information. A conferencing application (not shown) may be included to allow a user to select communication partners to be invited to a conference call initiated by the user of the PID **410**. Many other applications are also possible, and further examples of applications suitable for a PID may be found at http://www.palm.com, http://www.palmcentral.com, or http://www.tucows.com. Although the processor **704** is shown connected to the data storage **708** through a bus **702**, other configurations may also be used. Similarly, the

memory **706** may be alternatively configured, and may be embedded within the processor **704**.

The PID **410** may be able to send data to and receive data from the data network telephone **408** across a point-to-point link, such as the point-to-point link **409**. A user enters PID data at the display input **728**. The PID data may be processed in the user interface circuitry **710** or it may go directly to the processor **704** or the memory **706**. The processor **704** may also perform such processing functions as compression. A PID data application may be used to implement the display input, the display output, and the processing functions.

As an example, a drawing application may be used to accept PID data input at the display input **728** from a user drawing with a stylus on a display screen (if one exists) of the PID **410**. A drawing application could then display the drawing through the display output **726** to enable the user to see a visual representation of the drawing. If the user desires to share the drawing with a second user on the system **400**, where the second user is using a second PID, the PID data from the drawing application can be transmitted through one of the point-to-point interfaces **700**, allowing the data to be received by the data network telephone **408**. An application in the data network telephone **408** receives the PID data across the point-to-point link, and the PID data is prepared for transmission across the data network **406**, such as by the media engine **634** shown in FIG. **6**. Preferably the PID data is converted to PID data packets and is communicated on a PID data channel across the first LAN **412** through the router **413** across the data network **406** and eventually to a network device at which the second PID is located.

The point-to-point link **409** may be implemented as a serial bit stream between an application in the PID **410** and an application in the first data network telephone **408**. For example, the link **409** could be an infrared link that is implemented with minimal stack interpretation. However, the infrared link **409** between PID **410** and the first data network telephone **408** can alternatively be implemented using all or parts of a specialized protocol, such as the Infrared Data Association (IrDA) protocol stack, where data is interpreted through the stack between application-layer processes at each end of the link.

FIG. **8** is a protocol diagram illustrating the layers of the IrDA protocol stack. An IrDA stack is implemented at each of the connection endpoints of an IrDA link. The required layers of an IrDA protocol stack are the physical layer **802**, the IrLAP layer **804**, the IRLMP layer **806** and the IAS layer **808**. The physical layer **802** specifies optical characteristics of the link, encoding of data, and framing for various speeds. The IrLAP (Link Access Protocol) layer **804** establishes the basic reliable connection between the two ends of the link. The IrLMP (Link Management Protocol) layer **806** multiplexes services and applications on the IrLAP connection. The IAS (Information Access Service) layer **808** provides a directory or "yellow pages" of services on an IrDA device.

The IrDA protocol also specifies a number of optional protocol layers, these protocol layers being TinyTP **810**, IrOBEX **812**, IrCOMM **814** and IrLAN **816**. TinyTP (Tiny Transport Protocol) **810** adds per-channel flow control to keep traffic over the IrDA link moving smoothly. This important function is required in many cases. IrOBEX (Infrared Object Exchange protocol) **812** provides for the easy transfer of files and other data objects between the IrDA devices at each end of the link. IrCOMM **814** is a serial and parallel port emulation that enables existing applications that use serial and parallel communications to use IrDA without change. IrLAN (Infrared Local Area Network) **816** enables walk-up infrared LAN access for laptops and other devices. The use of the optional layers depends upon the particular application in the IrDA device. The IrDA protocol stack is defined by such standards documents as "IrDA Serial Infrared Physical Layer Link Specification", "IrDA 'IrCOMM': Serial and Parallel Port Emulation over IR (Wire Replacement)", "IrDA Serial Infrared Link Access Protocol (IrLAP)", "IrDA Infrared Link Management Protocol (IrLMP)", and "IrDA 'Tiny TP': A Flow-Control Mechanism for use with IrLMP", and related specifications published by the IrDA and available at http://www.irda.org/standards/specifications.asp and is incorporated by reference herein.

In one embodiment, the data network telephones **408**, **418**, and **428** merely provide a data tunnel for the data channel attendant to the infrared links, while the IrDA protocol stack is implemented at endpoint PID devices, such as PID **410**. Alternatively, IrDA stacks can be implemented in the data network telephones as well. By implementing additional layers of the IrDA protocol stack, the PID applications and the base applications in the data network telephones can be simplified because the IrDA protocol layers take over certain functions. For example, the IrDA protocol stack can be implemented at each PID used in a conference call, and the IrOBEX layer **812** can be used to transfer text and graphics object files, such as drawings or electronic business cards, end-to-end between PID devices connected via data network telephones and networks.

FIG. **9** shows a second embodiment of PID **410** according to the present invention. The PID **410** may be part of a more complex device, such as a portable phone. The PID **410** might also be a simple data storage object, such as a smart card or a computer disc. Included within the PID **410** are a data storage unit **900** and a data storage interface **902**.

The data storage unit **900** contains a user information database. The user information database contains user information such as personal address and schedule information, for example.

The data storage interface **902** provides access to the data stored in the data storage unit **900**. The complexity of the data storage interface **902** will depend on what reading or modifying tasks are performed by an outside device, such as a voice communication device or data network telephone, as compared with which tasks are performed by the PID **410**. If the PID **410** is a simple computer disk or smart card, the data storage interface may be primarily mechanical in nature, so that the PID **410** is in position to read or modify user information contained in the data storage unit **900**. If the PID **410** is more complex, then the data storage interface may include circuitry, possibly for reading or modifying the stored information. Infrared, magnetic, or radio frequency technology may be used to implement the data storage interface **902**, for example.

Other implementations of PIDs may be used besides those described with reference to FIGS. **7** and **9**. The PID will preferably include a user information database stored in data storage or memory, and should include a means for allowing an outside device to read and possibly modify the user information contained in the user information database.

Many alternative embodiments are also made possible by utilizing the PID **410**. For example, the PID **410** may store and download to the data network telephone **408** the preferences of the user about the phone operation, such as the ringer volume and tone. The PID **410** may also act as a smart card, providing authentication information for making toll calls. In another embodiment, the user of the PID **410** may program the system through the PID **410** so that, depending on the time of day, and on the datebook information in the

PID **410**, the phone forwarding information is dynamically updated. For example, during business hours, the default location to forward calls could be set to be the user's office, and during other hours, their cellular phone or their pager. If the PID **410** has voice playback capability, it can download voice mail and play it back off-line. On a LAN, this would be implemented as a file transfer, which is much faster than playing audio back. This feature would be useful if the user cannot spend too much time on the phone to check their voice mail. For example, a traveler at an airport may download their 30 minutes worth of voice mail in a few minutes, just before taking their flight, and may listen to those messages during the flight.

8. Providing User Mobility Services

FIG. **10** is a functional block diagram and protocol stack diagram illustrating an embodiment of the protocol stacks in the PID **410** and the first data network telephone **408** that support link **409**. In the infrared RS-232 embodiment, the point-to-point interface circuitry **700** in the PID **410** provides the physical layer **1000**, such as that specified by the Infrared Data Association (IrDA), that connects via link **409** to the point-to-point interface circuitry **608** implementing a physical layer **1002** in the first data network telephone **408**. The data link layer **1004** in PID **410** provides data link control for link **409** in transferring data to and from a PID application client **1006**. Similarly, the first data network telephone **408** includes a data link layer **1008** and a base application server **1008** that is configured to synchronize connection and other functions with the PID application **1006** in PID **410**.

When PID **410** is activated, either through power-up or through a user input at the user interface **1010**, the application client **1006** in the PID **410** may send the user's SIP URL across the link **409** to the first data network telephone **408**, where it is received by the application server **1008**. The base application server **1008** sends the SIP URL received from the PID **410** across connection **430** and the Ethernet LAN **412** through connection **443** to the network telephony connection server **450**. The network telephony connection server **450** may store the SIP URL and the IP address of the associated data network telephone **408** in a SIP database (not shown) so that the SIP URL is listed as being resident at the IP address of the data network telephone **408**. (If the network telephony connection server **450** uses a location server for registration/location tasks, the registration information might instead be stored with such a location server). SQL (Structured Query Language) is preferred for implementing and maintaining the database. Once the PID **410** is registered with the network telephony connection server **450**, calls to the SIP URL for PID **410** (or the user of the PID **410**) will be directed to the first data network telephone **408**.

FIG. **11** is a functional block and protocol stack diagram illustrating an embodiment of the present invention where a SIP connection is established from the first data network phone **408** to the second data network phone **418** through network connection **430**, first access network **412**, data network **406**, second access network **414** and network connection **419**. The routers **413** and **415** and associated connections are not shown to simplify the block diagram representation. Although only two data network telephones are shown in FIG. **11**, a three-party conference call would look very similar to what is shown in FIG. **11**, with the addition of an additional data network telephone. The first PID **410** and a second PID **420** are also shown for exemplary purposes, but need not be included for most embodiments.

The diagram of FIG. **11** shows how PID user data can be communicated from onc PID to another PID during a call in one aspect of the present invention. The PID application **1006** in PID **410** is configured to send PID data received through the user interface **1010** through link **409** to base applications **1008** in the first data network phone **408**. In this embodiment, base applications **1008** are configured to define data channels for transport to the second data network telephone **418**.

Multiple data channels in SIP may be defined through the Session Description Protocol described in RFC 2327, herein incorporated by reference. Included in a SIP INVITE request are options for the requested connection that describe the number and type of media streams. Each media stream is described by an "m=" line in the INVITE request. For example, a request for a connection that includes an audio stream and a bidirectional video stream using H.261 might look like this:

TABLE 1

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.anywhere.com
c=IN IP4 host.anywhere.com
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 51372 RTP/AVP 31
a=rtpmap:31 H261/90000
```

If the called device includes functionality to receive the connection as described in Table 1, then the called device will respond to the INVITE request with a 200 OK response that includes the same option values. If the called device or party is unable or unwilling to receive such a connection, then it will respond with alternative option values for the connection. See RFC 2543 for further details regarding the negotiation of connection parameters in SIP.

In FIG. **11**, a first data channel for voice data and a second data channel for PID user data have been negotiated by the base applications **1008** in the first data network telephone **408** and the base applications **1100** in the second data network telephone **418**. The base applications **1008** and **1100** transfer voice data between the AUDIO applications, such as applications including G.711 encoders, in each phone via the first data channel. The base applications **1008** in phone **408** are also configured to send the PID data received via link **409** from PID **410** to the base applications **1100** in phone **418** via the second data channel. The base applications **1100** in phone **418** may be configured to forward the PID data received via the second data channel to a second PID **420** via a second link **421**. The PID application **1102** in PID **420** then outputs the PID data received from phone **418** to the user interface **1104** for output to the user of PID **420**.

The PID data in FIG. **11** can take a variety of forms. For example, the PID data can be a text file containing information about the user of PID **410**, such as an electronic business card. The PID data can also be drawing data generated by graphical applications in the PIDs **410** and **420** whereby a user drawing on a touchscreen of the user interface **1010** in PID **410** generates corresponding PID data that is transmitted via the second data channel to PID **420** for display on the user interface **1104** of PID **420**. The media description for the media stream can be defined during connection setup to establish a connection appropriate to the type of data being transferred. These examples represent just a few of the applications for this aspect of the present invention and should not be viewed as limiting the present invention.

In one embodiment, RTP data packets for two or more types of data are exchanged between the first data network

telephone **408** and the second data network telephone **418** according to one of three possible methods. In the first method, one RTP data channel (or RTP stream) on UDP carries data packets in which both data types are present in a single split packets. Each such split packet contains (1) a source port number and a destination port number in the UDP portion, and (2) a special payload sequentially including each of the data types in the RTP portion. The special payload type can be defined in the SDP described above. Other information is also contained in each packet as well. In the second method for transmitting two or more data types, a separate RTP over UDP data channel is created for each of the different data types, and the RTP header indicates which type of data is contained in each packet. For example, voice data coded as G.711 might be assigned a payload type code of 0, while PID data is assigned a payload type code of 190. In the third method for transmitting two or more data types, a single RTP/UDP data channel (RTP/UDP stream) is created that contains data packets of two or more different types. In this method, the data types are identified in a payload type field in the RTP header of each packet, enabling an underlying application to identify which data packets are voice data packets and which data packets are PID data packets, for example.

FIG. **12** is a pictorial diagram showing an exemplary display screen of a PID **410** displaying entries from an address book application **900** according to one embodiment of the present invention. Shown are a first contact entry **902** and a second contact entry **904**. The entries each contain contact information, such as name, address, email, SIP URL, and other information. In the preferred embodiment, the user of the PID **410** is given the option to flag entries in the address book, such as by checking a flag field **906** in the first contact entry **902**. When the communication parties to the impending conference call have been flagged (UserB and UserC in FIG. **12**), the user of the PID **410** can click on a box **908** to cause a conference call to be placed to people, numbers, or locations referred to by the flagged contact entries. Additional features may also be implemented in the address book application **900**.

FIG. **13** is a pictorial diagram showing an exemplary display screen **1000** of a PID **410** displaying a conference call attempt **1002** according to one embodiment of the present invention. Such a screen **1000** may be part of a conference call application **1004** executed after contact entries are flagged in the example of FIG. **12** to initiate a conference call. The conference call application **1004** is shown attempting a conference call to two communication partners: UserB **1006** and UserC **1008**. Note that a timeout period may be set to end a call attempt to a communication partner that does not respond. Other methods for handling unanswered requests may also be implemented, and are intended to be within the scope of the present invention.

FIGS. **12** and **13** illustrate an embodiment of the present invention in which a PID is used to perform the actual dialing operations. In other embodiments, the PID may be used to register the user with a particular data network telephone. The user may then use the data network telephone user interface (such as a keypad) to input the user identifier or device identifier (such as a phone number) of the party the user wishes to call. In such a case, the PID would not be necessary for communications to proceed. Additional details for implementing embodiments in which a PID is used to initiate calls may be found in Dalgic, et al., "Through Number Portability and Advanced Call Screening in a SIP-Based IP Telephony System," IEEE Communications Magazine, July, 1999, pp. 96–101; and U.S. patent application Ser. No. 09/181,431 "Method, Apparatus and Communications System for Companion Information and Network Appliances," filed on Oct. 30, 1998, by Inventors Peter Si-Sheng Wang and Ismail Dalgic, assigned to 3Com Corporation, both of which are incorporated by reference herein.

FIG. **14** is a flow diagram illustrating a method **1400** for providing user mobility services at a voice communication device in a network according to an embodiment of the present invention. In step **1402**, user attributes are received from a portable information device at the voice communication device. The user attributes are associated with a user of the portable information device. In step **1404**, a registration request is transmitted from the voice communication device to a registration server. The registration server registers the user to the voice communication device and a registration data base.

FIG. **15** is a flow diagram illustrating a method **1500** for providing user mobility services at a voice communication device in a network according to an embodiment of the present invention. Steps **1502** and **1504** are similar to steps **1402** and **1404**, described with reference to FIG. **14**. In step **1506**, a call is processed by referencing the registration data base and directing the call to the voice communication device. The registration server is preferably used to reference the registration data base.

The methods **1400** and **1500** are preferably implemented with an Ethernet network, Ethernet-based SIP phones, and a PID that is a personal digital assistant. The steps of **1402** and **1502** or receiving user attributes from the portable information device at the voice communication device may be performed during a synchronization operation between the PID and the voice communication device. Alternatively, the PID could be a smart card, and the user attributes could be transferred to the voice communication device during a card-read operation.

While the invention has been described in conjunction with presently preferred embodiments of the invention, persons of skill in the art will appreciate that variations may be made without departure from the scope and spirit of the invention. For example, the access networks shown in FIG. **2** may comprise any other suitable type of local area network or service infrastructure.

In addition, protocols of various types are referenced throughout. While preferred and alternative embodiments may implement selected protocols, any suitable replacement protocol not mentioned, or any function not part of a protocol used to replace a corresponding function from a protocol may be implemented without departing from the scope of the invention.

This true scope and spirit is defined by the appended claims, interpreted in light of the foregoing.

We claim:

1. A system for providing user mobility services on a data network telephony system comprising:

a data network to provide data connectivity for a plurality of data communications channels using data transport protocols;

a plurality of data network telephones connected to the data network, each data network telephone operable to communicate a voice signal as voice-over-data packets on a voice-over-data channel, the voice over data channel being one of the plurality of data communications channels on the data network, the data network telephones each operable to convert voice-over-data packets communicated on the voice-over-data channel to voice signals; and

a portable information device, the portable information device associated with a user and comprising a user interface and a data network telephone interface, the user interface operable to accept PID data from the user, the data network telephone interface operable to communicate PID data to and from a first data network telephone, the PID data including user attributes;

wherein the PID data is transmitted by the portable information device to the first data network telephone, the PID data used to register the user with the first data network telephone.

2. The system of claim **1**, wherein the PID data includes communication partner data, and wherein the user invites at least one communication partner to a call, the communication partner specified in the communication partner data, each of the communication partners registered with one of the plurality of data network telephones.

3. The system of claim **2** wherein:

at least a first and second user communicate on the voice-over-data channel, each user identified by a user identifier that includes a unique sequence of alpha numeric elements.

4. The system of claim **3** wherein each data network telephone includes a device identifier that corresponds to the user identifier.

5. The system of claim **4** wherein the device identifiers include Internet Protocol (IP) addresses.

6. The system of claim **3** wherein the user identifiers include Session Initiation Protocol (SIP) addresses.

7. The system of claim **3** wherein the user identifiers include E.164 telephone numbers.

8. The system of claim **1** further comprising:

a network telephony user database connected to the data network to store a user identifier and a telephone identifier corresponding to the user identifier for each of a plurality of users, wherein:

the user identifier includes a first sequence of alphanumeric elements that identify a corresponding user;

the telephone identifier includes a second sequence of alphanumeric elements that identifies a corresponding data network telephone; and

a network telephony connection server operable to receive a request message from the second data network telephone to initiate the voice over data channel and the encrypted data channel with the first data network telephone, and to send a response message in response to the request message.

9. The system of claim **8**, wherein the response and request messages are communicated by the network telephony connection server in accordance with the Session Initiation Protocol (SIP).

10. The system of claim **8**, wherein the response and request messages are communicated by the network telephony connection server in accordance with the H.323 Protocol.

11. The system of claim **8** wherein the response and request messages are communicated by the network telephony connection server in accordance with the MEGACO protocol.

12. The system of claim **8** wherein the response and request messages are communicated by the network telephony connection server in accordance with the MGCP protocol.

13. The system of claim **8** wherein:

the request message includes a callee user identifier; and

wherein the network telephony connection server determines the telephone identifier for the callee user identifier and includes the telephone identifier in the response message.

14. The system of claim **8** wherein:

the request message includes a callee user identifier; and

wherein the network telephony connection server determines the telephone identifier for the callee identified in the call user identifier and sends the response message to the callee at the telephone identifier.

15. A system for providing user mobility services to a user of a portable information device, the portable information device having user attributes stored in a user information database, wherein the user attributes are associated with the user, comprising, in combination:

a voice communication device connected to a network, wherein the voice communication device includes a user interface for accepting user inputs and delivering user outputs, and wherein the user attributes are transmitted from the portable information device to the voice communication device; and

a registration server connected to the network, wherein the registration server registers the user to the voice communication device in a registration database in response to the voice communication device issuing a register request, the register request including the user attributes.

16. The system of claim **15**, further comprising a gateway to a public switched telephone network.

17. The system of claim **15**, further comprising a router to a public internet.

18. The system of claim **16**, wherein the gateway includes a SIP client, the voice communication device is a SIP phone, the registration server is a SIP server, and the registration database is a SIP database.

19. The system of claim **17**, wherein the router includes a SIP client, the voice communication device is a SIP phone, the registration server is a SIP server, and the registration database is a SIP database.

20. The system of claim **15** wherein the registration server processes a call addressed to the user by referencing the registration database and directing the call to the voice communication device.

21. A system for providing user mobility services to a user associated with a portable information device, comprising, in combination:

a first set of machine language instructions for causing the portable information device to transmit user attributes from a user information database stored in a memory of the portable information device to a first voice communication device;

a second set of machine language instructions for causing the first voice communication device to accept the user attributes from the portable information device and to transmit a register request to a registration server, the register request including the user attributes; and

a third set of machine language instructions accessible by the registration server for causing the registration server to accept the register request from the first voice communication device and to update a registration database maintained by the registration server to register the user with the first voice communication device.

22. The system of claim **21**, wherein the second set of machine language instructions further causes the first voice communication device to place a call to a second communication device responsive to receiving input from the user, the input including a device identifier corresponding to the second communication device.

23. The system of claim 22, wherein the third set of machine language instructions further causes the registration server to locate the second communication device to which the call is placed.

24. The system of claim 21, wherein the first voice communication device is a personal computer having a microphone and a speaker.

25. The system of claim 21, wherein the register request is a REGISTER message according to the Session Initiation Protocol.

26. The system of claim 21, wherein the third set of machine language instructions further causes the registration server to forward a call to the first voice communication device, the call initiated at a second voice communication device.

27. A method for providing user mobility services at a voice communication device in a network, comprising the steps of:

receiving user attributes from a portable information device at the voice communication device, wherein the user attributes are associated with a user of the portable information device; and

transmitting a registration request from the voice communication device to a registration server, wherein the registration server registers the user to the voice communication device in a registration database.

28. The method of claim 27, further comprising the step of processing a call by referencing the registration database and directing the call to the voice communication device.

29. The method of claim 27, wherein the network includes an Ethernet network, the voice communication device is an Ethernet-based SIP phone, and the portable information device is a personal digital assistant.

30. The method of claim 27, wherein the portable information device is a personal digital assistant, and wherein the step of receiving user attributes from the portable information device at the voice communication device is performed during a synchronization operation between the portable information device and the voice communication device.

31. The method of claim 27, wherein the portable information device is a smart card.

* * * * *