(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 202 234 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
02.05.2002 Bulletin 2002/18

(51) Int Cl.[7]: **G08G 1/127**

(21) Application number: **01308997.4**

(22) Date of filing: **23.10.2001**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE TR
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **24.10.2000 US 696722**

(71) Applicant: **@Road, Inc.**
Freemont, California 94538 (US)

(72) Inventor: **Taylor, Sean Dominic**
Fremont, California 94538 (US)

(74) Representative: **Allman, Peter John et al**
MARKS & CLERK,
Sussex House,
83-85 Mosley Street
Manchester M2 3LG (GB)

(54) Targeted impending arrival notification of a wirelessly connected location device

(57) A rover determines a destination and periodically checks the location of the rover against one or more active alert thresholds. One alert threshold is the boundary of an alert area surrounding a destination. The rover sends an alert to a service center alerts a designated party of an impending arrival of the rover at the destination when the rover is within an alert area surrounding the destination. The service center can also alert a designated party when the rover leaves an alert area surrounding a location. The alerts can be communicated via an automated telephone message, a pager message, an e-mail message, or any other means of communication. The alerts can be used for deliveries by directing an alert to a customer expecting a delivery or to a site expecting a vehicle for loading or unloading. In a private context, the alerts can automatically activate systems such as a home appliances or systems in anticipation of a resident's arrival.

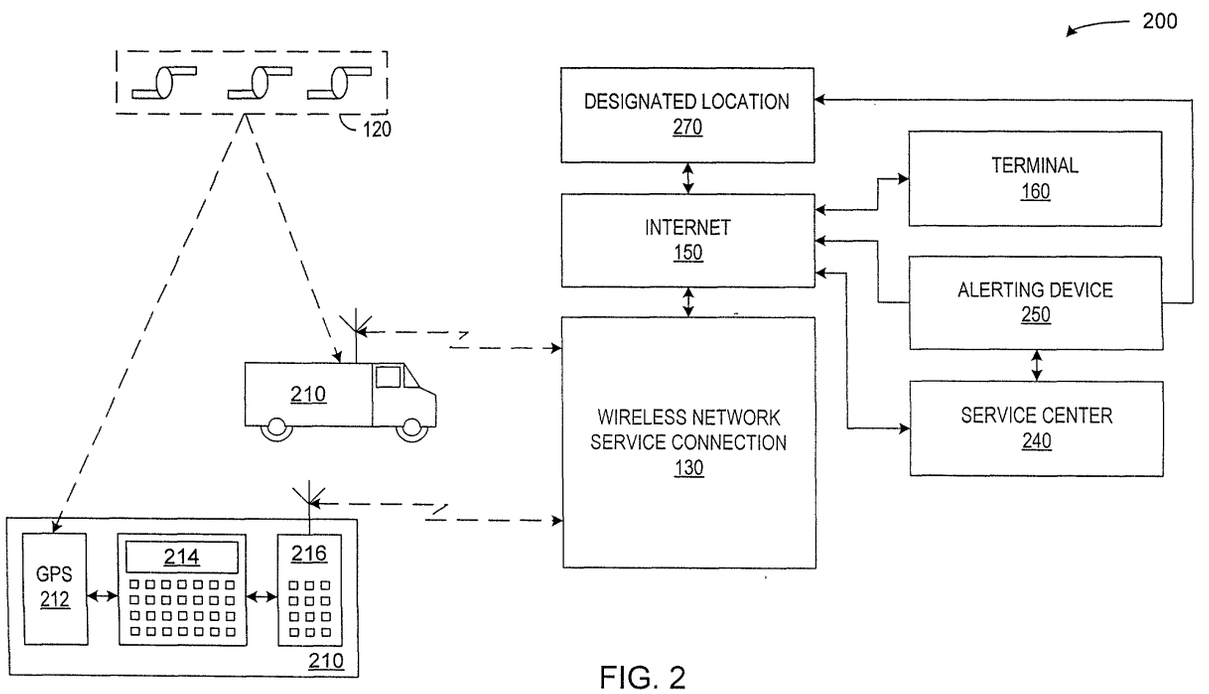

FIG. 2

EP 1 202 234 A1

Description

COPYRIGHT NOTICE

BACKGROUND

Field of the Invention

[0002] This invention relates to communications and tracking systems that track the location, movement, and destination of vehicles or individuals.

Description of Related Art

[0003] Many businesses deliver packages to customers and may have one or a fleet of vehicles for such deliveries. A nearly universal problem for these businesses is tracking and planning the movement of the delivery people and vehicles. Fig. 1 shows a tracking system similar to that described in U.S. patent application No. 5,959,577, which is hereby incorporated by reference in its entirety. The system of Fig. 1 includes multiple mobile units (also referred to herein as rovers) 110. Each rover 110 can be a handheld device that a delivery person carries or a device mounted in a vehicle. Each rover 110 contains a global positioning system (GPS) receiver, a wireless device, and a control circuit. When activated, the GPS receiver receives signals from GPS satellites 120 and from the signal identifies the position (e.g., longitude and latitude) and velocity of a rover 110. The wireless device, typically a wireless modem, transmits the position and velocity information via a wireless network 130 and the Internet 150 to a service center 140.

[0004] Service center 140 receives and collects data from mobile units 110 and makes the collected data available to a user of a terminal 160. Terminal 160 is typically a computer that connects via the Internet 150 to a web site associated with service center 140. The user can view the data or send a query or message to service center 140 for relaying to a specific rover 110. In response to communications from terminal 160, service center 140 sends the query or message to the selected rover 110.

[0005] Although the system of Fig. 1 provides a convenient method for a business to track deliveries, the system is generally not accessible to the business' customers who may also be interested in knowing when a delivery will arrive. Additionally, the business expecting an arrival of a vehicle at a particular location such as a loading dock must periodically check the position of the vehicle to determine when the vehicle can be expected. This monitoring may be subject to errors since the vehicle while near the destination may be headed elsewhere. Accordingly, a possible improvement of the system of Fig. 1 would be to provide better information indicating expected arrival times and to provide such information not only to the business but also to customers without requiring the business or customers to constantly poll the position of the delivery vehicle.

SUMMARY

[0006] In accordance with an aspect of the invention, a rover identifies a destination or travel threshold, periodically checks the rover's location relative to the destination or threshold, and sends an alert to a service center upon nearing the destination or crossing the threshold. The service center relays the alert to one or more designated party. Accordingly, the service center can alert a designated party when the rover nears a destination or when the rover leaves an alert area surrounding a location. The alerts can be relayed via an automated telephone message, a pager message, an e-mail message, or any other communication means. The alerts can be used for deliveries by directing an alert to a customer expecting a delivery or to a site expecting a vehicle for loading or unloading. In a private context, the alerts can automatically activate systems such as a home appliances or systems in anticipation of a resident's arrival.

[0007] One embodiment of the invention is an alert generating method. The alert generating method includes: providing to a mobile unit information that identifies conditions for an alert; monitoring by the mobile unit, of the position of the mobile unit; and alerting a designated location when the monitoring indicates the mobile units satisfies the conditions for the alert. The conditions for an alert typically include that the mobile unit is headed to a specific destination and is within an alert area surrounding that destination or the mobile unit has entered or left an alert area. Typically, alerting the designated location includes: sending a signal from the mobile unit to a service center when the mobile unit satisfies the conditions for the alert; and generating the alert from the service center to the designated location in

response to the signal from the mobile unit. The alert can be sent to a telephone, a pager, or an e-mail address.

[0008] Another embodiment of the invention is a delivery method that includes creating a list of destinations for deliveries. The list includes a threshold distance and other information for one or more destination for which an alert should be generated. A next destination that a delivery vehicle is headed towards is selected from the list for the delivery vehicle, and the delivery vehicle monitors distance between the delivery vehicle and the selected destination. The delivery vehicle generates an alert when the distance is less than a threshold distance associated with the destination. More specifically, generating the alert includes sending a message from the delivery vehicle to a service center, looking-up a designated location that corresponds to the destination, and sending the alert from the service center to the designated location.

[0009] Yet another embodiment of the invention is a mobile unit that includes a location system, a wireless device, and a control circuit. In one mode, the control system periodically activates the location system to determine a current location of the mobile unit, determines whether the mobile unit has satisfied an alert condition (e.g., has entered or left an alert area), and activates the wireless device to send an alert signal if the mobile unit has satisfied the alert condition. The location system is typically a GPS receiver, and the wireless device is typically a wireless modem or telephone. The control circuit can determine whether the current location is within the alert area by determining whether a distance between the current location and a central point or destination in the alert area is less than or greater than a threshold distance associated with the alert area.

[0010] Still another embodiment of the invention is a system including a communication connection, an alerting device, and a service center. The communication connection allows the service center to receive messages from a mobile unit, and the alerting device allows the service center to send alerts. Generally, the service center maintains contact information for the mobile unit, and in response to a signal from the mobile unit, the service center activates the alerting device to send an alert to a designated location identified in the contact information. The contact information can be kept at the service center or forwarded from the mobile unit to the service center. The service center is generally Internet accessible to allow multiple, geographically disparate people to set the conditions for an alert and the designated location to which the alerting device sends the alert.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011]

Fig. 1 shows a block diagram of a tracking system.

Fig. 2 shows a tracking and impending arrival notification system in accordance with an embodiment of the invention.

Fig. 3 is a flow diagram of a process for generating an alert in accordance with an embodiment of the invention.

[0012] Use of the same reference symbols in different figures indicates similar or identical items.

DETAILED DESCRIPTION

[0013] In accordance with an aspect of the invention, a tracking system automatically alerts a designated party before a mobile unit (or rover) arrives at a destination. Fig. 2 illustrates a system 200 implementing alerts in accordance with the invention. System 200 is similar to system 100 described above and includes rovers 210, GPS satellites 120, a wireless network service connection 130, a service center 240, an alerting device 250, the Internet 150, and terminal 160. Rovers 210 and service center 240 of Fig. 2 are similar to rovers 110 and service center 140 of Fig. 1. In particular, service center 240 communicates with and tracks multiple rovers 210, as describe above. However, rovers 210 and service center 240 have additional features for generating impending arrival alerts to designated locations 270. These additional features can be implemented in software or firmware performing the functions described further below.

[0014] Each rover 210 includes a communications package such as the iLM 2000 available from @Road, Inc. The communications package includes a GPS receiver 212, a control circuit 214, and a wireless device 216. GPS receiver 212 when activated interprets signals from GPS satellites 120 to identify the position and velocity of the rover 110. Control circuit 214 has a user interface including a keyboard or other input device for operator control of the communications package, a display or another output device for conveying information such as the status of the package and received messages, and a processing circuit. The processing circuit implements automated operation and commands from the operator or from service center 240.

[0015] Wireless device 216 and service connection 130 handle communications between service center 240 and rovers 210. Connection 130 can be any wireless service, whether analogue or digital, which supports data transfers between service center 240 and rover 210. For example, the communication can operate through a CDPD, AMPS, CDMA, GSM, or Nextel system, using both OEM modem modules internal to the rover or cellular phones that are

separable from the rover and attached to the remainder of the communication package via a data cable. Although the communications package could use a removable or separate cellular telephone, this is not the ideal solution since removing the telephone disables the connection between the communications package and service center 240. The wireless service 132 preferably offers packetized data and direct connection to the Internet because these capabilities facilitate sporadic data transfers between rovers 210 and service center 240 with least use of the wireless resources and therefore the least cost to the operator of the communications package.

[0016] Fig. 3 is a flow diagram of a method 300 of generating impending arrival alerts in accordance with an embodiment of the invention. An initial step 310 of method 300 is creation of a destination list that can be forwarded or downloaded to service center 240 and/or rover 210. Each rover 210 has a destination list, which contains a set of destination entries corresponding to the intended destinations of the rover 210. (The Appendix contains a C-language data structure for one embodiment of a destination list.) Each destination entry includes a location (e.g., an address or longitude and latitude of a destination), an alert condition, contact information for the alert or alerts, and any other information related to the destination, the alert, or the contact. Service center 240 can easily convert an address provided in a destination list to longitude and latitude values, which are commonly used in GPS systems. The alert condition typically includes threshold information, which is typically a distance or radius that defines an alert area around the destination. The alert condition may additionally include fields indicating circumstances for the alert. Typical circumstances surrounding an alert include whether the alert should be sent at all, a direction of traversal of the threshold that triggers an alert, an order of the destinations, and any time limitations on the alert. The contact information identifies designated location 270 and the method for contacting location 270 when an issuing an alert. The contact information can include, for example, a telephone number, a pager number, or IP address and port.

[0017] The destination list can be maintained in service center 240 or rover 210 or a combination of the two. However, keeping and manipulating the destination list at service center 240 has advantages. For example, a central dispatcher through terminal 160 and data connection 150 can organize destinations into an ordered destination list and download the list to service center 240. Since service center 240 is ideally Internet based, multiple people could access and manage the destination list via the Internet 150, even after rovers 210 have started their routes. End customers, who are anticipating a delivery, could also be given limited access to their destination entry to determine an estimated arrival time, schedule an alert immediately before arrival at their address, or change the contact information. Besides a web interface, the service center can also provide a telephone or e-mail interface for creating, changing, or reviewing a destination list.

[0018] Each destination entry is at least partially downloaded to rover 210, so that rover 210 at least has the destination and threshold information for the next destination. The destination location can be latitude and longitude (probably from geocoding the street address) as used in the GPS receiver. Generally, when service center 240 keeps the destination list, rover 210 does not require the contact information, but other information may be useful in rover 210. For example, text sent to rover 210 could indicate any information related to the destination, for example, a destination name and address that permits a driver/operator to more readily identify destinations when scrolling through the destination list. An appointment time associated with a destination provides the driver additional information for scheduling of deliveries.

[0019] Additional fields used by control circuit 214 in the communications package can indicate whether to generate an alert when the rover 210 crosses into the alert area, out of the alert area, or both. Generally, an alert is generated when rover 210 moves into the alert area to indicate an impending arrival. However, an alert can also be generated when rover 210 leaves an alert area indicating rover 210 is headed elsewhere, for example, possibly indicating that rover 210 is outside its intended range and may have been stolen or misdirected. Other information that can be included with a destination entry indicates an expiration time (when the entry is automatically deleted whether or not one or more alerts were generated), whether the entry should auto-delete after an alert is sent or is retained until its expiration time, and alternate contact information that may be indexed for use at different times.

[0020] In step 320, when rover 210 begins heading to the next destination, the operator/driver of rover 210 selects that destination for use in generating alerts. In particular, some alerts may only be issued if the alert is for the next destination. Other alerts could be issued regardless of the next destination. Optionally, the rover informs service center 240 of the selection. The selection of the destination can be automatic according to the order in the destination list or subject to the choice and judgement of the driver. If the list is downloaded to the rover 210 instead of being managed exclusively at service center 240, the in-vehicle system can display candidate destinations for the driver's selection. In some cases, the driver may override the order of destinations in the destination list because of traffic or other reasons not anticipated by the list's creators. Where the driver overrides the order, the rover 210 can send a message directing the service center to update the destination list to reflect the new order and update estimates of delivery times. The driver could even create new entries, although the limits of the user interface available on an in-vehicle system might make this difficult.

[0021] After rover 210 forwards or confirms the next destination, service center 240 in step 330 confirms the next destination and instructs the rover 210 to keep checking the location until rover 210 an alert condition is met, e.g., the

rover enters an alert area for the selected destination. If necessary, service center 240 supplies a location for the selected destination and the alert conditions during step 330, but that information may have been previously downloaded into rover 210. In step 340, rover 210 determines whether rover 210 has crossed the boundary of the alert area. The appendix includes functions that record route hysteresis to determine whether a boundary has been crossed and whether the crossing direction is into or out of to the area within the boundary. In particular, if the alert condition indicates a bounding radius and a direction into the bounded area, the rover determines whether the rover has crossed from a distance from the destination that is greater than the bounding radius to a distance from the destination that is less than the threshold radius. Other more complicated threshold area descriptions could also be used. For example, the alert condition could identify an alert area other than a circle or include some variations based on the current or recent average speed of rover, but such identifications could significantly complicate the required calculations in rover 210.

[0022] A decision step 350 determines whether rover 210 reached the threshold for an alert. If not, rover 210, typically after a delay, again determines whether it has reached the threshold for an alert. In step 350, when rover 210 determines that it has crossed the threshold in the target direction, rover 210 in step 360 sends an alert to service center 240. In step 370, service center 240 forwards the alert to designated location 270 using the contact information from the destination list. The designated location 270 can be the destination or to another location. For example, if the destination is a customer's home and the designated location 270 is the customer's work address, the customer upon receiving the alert can return home to accept the delivery. As noted above, the designated location 270 can change according to factors such as the time.

[0023] In system 200 of Fig. 2, service center 240 activates alerting device 250 to send the alert to designated location 270. Alerting device 250 is a communication device such as an automated telephone messaging system or an automated e-mail system. The alerted party can be alerted in a variety of ways, including but not limited to by e-mail, paging message, text message to a cellular phone, or a telephone call with a synthesized speech or recorded message.

[0024] Method 300 can be widely varied without departing from the scope of the invention. For example, instead of providing service center 240 with a destination list, rovers 210 can keep their own destination lists. When the operator/driver of a rover 210 selects the next destination in step 320, the communications device in the rover accesses threshold and contact information in the stored on board in the destination list. In step 360, when rover 210 sends an alert to service center 240, the rover forwards the contact information for designated location 270, and service center 240 forwards the alert to the designated location 270.

[0025] In another variation of method 300, a rover 210 sends an alert whenever the rover is near particular destinations regardless of whether any of these destinations is the next destination. For this variation, the rover in step 340 compares the current location to the location and threshold information of all destinations that are eligible for alerts, and in step 350 decides to issue an alert if conditions for the alert are met. For example, a rover can issue an alert if the rover leaves a designate area or range from its base location (indicating perhaps that the rover is misdirected or stolen). Each destination entry can include a tag number that identifies the destination, and the rover 210 incorporates the tag number in the alert message sent to service center 240. From the tag number, service center 240 decides where (e.g., which designated location) and how to send the alert.

[0026] In some cases, the alerted party might be the truck dispatcher (for example, to alert the dispatcher that a truck is returning to base or that the truck has crossed a perimeter, heading away, and might be stolen). Service center 240 can forward an alert from a rover 210 to multiple designated locations, for example, to a customer, and to the dispatcher to alert the dispatcher as to the progress of their drivers.

[0027] Arrival alerting systems such as described above can be employed in a variety of applications. In warehousing, an alert indicating an impending arrival allows a delivery manager to deploy resources to a particular loading bay, and even to call ahead to the driver tell them which bay has been allocated for the delivery. In construction, when a supply truck is about to arrive, the site manager can ready loading/unloading equipment and people, to minimize turn-around time. Similarly, if a truck is delayed, resources are not wasted waiting for arrival, and time is not consumed periodically polling the position of the truck.

[0028] For consumer delivery, the alert can go to a location other than the destination, for example, to alert a consumer at work or via the consumer's pager that an impending delivery is destined for the consumer's home. If the consumer is near enough to home, the consumer can return home to meet the delivery or ask the delivery driver to wait. Another consumer application is automated activation of household equipment such as heating or cooling systems or appliances when the resident nears home. Energy can be saved by leaving heating and cooling systems in a low power mode while the resident is away and activating the system so that the home is at a comfortable temperature when the resident arrives. Alternatively, an alert can activate a coffeepot so that the resident has freshly brewed coffee when arriving home. The full range of applications of the invention is of course not limited to these few examples.

[0029] Although the invention has been described with reference to particular embodiments, the description is only an example of the invention's application and should not be taken as a limitation. For example, although the above embodiments employ GPS devices to identify locations, other locating systems using, for example, triangulation based on terrestrial signals or landmarks are also suitable for use in the embodiments described above. Various other adap-

tations and combinations of features of the embodiments disclosed are within the scope of the invention as defined by the following claims.

APPENDIX

This appendix contains listings for functions and data structures in C programming language, for use in a rover in an embodiment of the invention such as described above. The functions typical would be periodically executed in a rover to check any alert conditions of entries in a proximity queue and update the proximity queue to remove expired entries.

```
/*********************************************************/
/* prx_alrt.c                                            */
/* Function calls for proximity alert monitoring in the  */
/* rover.                                                */
/* S Taylor, @Road Inc.                                  */
/*********************************************************/

#include"rover_ds.h"

#define D_HYSTERESIS (float) 0.2  // 10% either side hysteresis
                                  // for points tripping in either
                                  // direction
#define TRIP_HYSTERESIS_TIME 120  // seconds


/*********************************************************/
// Function to check for tripping of entries in the ordered
// proximity queue, and deleting any expired points.  This
// function would typically be called every 10 seconds
/*********************************************************/
void check_ordered_proximity(long current_lat, long current_lon,
             ULONG current_time){

t_proximity_Q_entry *p_entry;
short need_resort;

    need_resort = 0; // assume we don't need to resort the queue
    p_entry = ordered_proximity_queue; // point at head of queue

    // first check for empty queue
    if(p_entry->tag_id){
       // check the head of the queue to see if it was tripped
       // see listing below for check_point_tripped function
       if(check_point_tripped(p_entry, current_time)){
            need_resort = 1;
       }

        // check the remainder of the queue to see if any points
        // have time expired
        for(i=1;i<SZ_ORDERED_PROX_Q;i++){
            p_entry = &ordered_proximity_queue[i];
            // check for time expiry of point
            if((p_entry->tag_id) &&
                    (p_entry->expiry_time < current_time)){
               // delete point (set to all zero)
               memset(p_entry, 0x00, sizeof(t_proximity_Q_entry));
               need_resort = 1;
            }
        }

        if(need_resort){  // resort the queue
           // sort by E.T.A. (smallest ETA at top of list),
```

```
            // placing empty entries at end of queue.
            sort_by_ETA(ordered_proximity_queue);
        }
    }
}

/*********************************************************/
// Function to check for tripping of entries in the unordered
// proximity queue, and deleting any expired points.  This
// function would typically be called every 30 seconds
/*********************************************************/
void check_unordered_proximity(long current_lat, long_current_lon,
            ULONG current_time){

t_proximity_Q_entry *p_entry;

    // go through all points in queue checking for trip
    for(i=0;i<SZ_UNORDERED_PROX_Q;i++){
        p_entry = &unordered_proximity_queue[i];     // point at entry
        // check for empty entry
        if(p_entry->tag_id){
           // see listing below for check_point_tripped function
           check_point_tripped(p_entry, current_time);  // check entry
        }
    }
}

/*********************************************************/
// Function to check data in a point, and add to the
// appropriate queue, resorting if necessary.
// Returns 0 if successful, 1 if point bad, -1 if queue full
/*********************************************************/
short add_new_point(t_proximity_Q_entry *pnew_entry,
            ULONG current_time){

short return_val;
t_proximity_Q_entry *p_entry;

    return_val = 0;

    // check that expiry time has not already been met,
    // that non-zero tag_id was assigned and that
    // will trip on at least one day
    if( (pnew_entry->expiry_time < current_time) ||
        (pnew_entry->tag_id == 0) || (pnew_entry->days_to_trip == 0) ){
       printf("Attempt to add expired/null tag entry to prox Q\n");
       return(1);  // point data was bad
    }

    // Also need to check for duplicate tag_id :
    // especially if resynchronizing with server
    // If duplicate is found then delete it.
    for(i=0;i<SZ_UNORDERED_PROX_Q;i++){
       p_entry = &(unordered_proximity_queue[i]);
       // identical tag_id indicates duplicate point
       if( (p_entry->tag_id == pnew_entry->tag_id) ){
           // delete existing entry
           memset(p_entry, 0x00, sizeof(t_proximity_Q_entry));
       }
    }

    for(i=0;i<SZ_ORDERED_PROX_Q;i++){
       p_entry = &(ordered_proximity_queue[i]);
       // identical tag_id indicates duplicate point
       if( (p_entry->tag_id == pnew_entry->tag_id) ){
           // delete existing entry
           memset(p_entry, 0x00, sizeof(t_proximity_Q_entry));
       }
```

```
    }

    // Now determine which queue entry must be added to
    // if trip_when_head is set then add to the ordered Q
    // so that point is only monitored when head of queue.
    // if trip_when_head == 0 then add to unordered queue
    // for constant monitoring.
    if(pnew_entry->trip_when_head){

       // add entry to ordered queue
        for(i=0;i<SZ_ORDERED_PROX_Q;i++){
           p_entry = &(ordered_proximity_queue[i]);
           if(p_entry->tag_id){   // queue entry is in-use
              continue;
           } else {
              // add entry to the queue
              memcpy(p_entry, pnew_entry, sizeof(t_proximity_Q_entry));
              // resort the queue
               sort_by_ETA(ordered_proximity_queue);
              return(0);  // success
           }
       }

    } else {
       // add entry to unordered queue
        for(i=0;i<SZ_UNORDERED_PROX_Q;i++){
           p_entry = &(unordered_proximity_queue[i]);
           if(p_entry->tag_id){   // queue entry is in-use
              continue;
           } else {
              // add entry to the queue
              memcpy(p_entry, pnew_entry, sizeof(t_proximity_Q_entry));
              return(0);  // success
           }
       }
    }
    return(-1);  // indicate queue was full
}

/*********************************************************/
// Function to find a point in one of the queues and
// delete it.  Allows server to delete points for
// whatever reason.  Zero tag id indicates delete all points.
// Returns 0 if successful, 1 if point not found
/*********************************************************/
short delete_existing_point(ULONG match_tag_id){

short return_val;
t_proximity_Q_entry *p_entry;

    return_val = 1;  // assume failure
    // this flag allows server to specify deletion of
    // all proximity points from both queues
    if(match_tag_id == 0){
       memset(ordered_proximity_queue, 0x00,
             SIZE_ORDERED_PROX_Q*sizeof(t_proximity_Q_entry) );
       memset(unordered_proximity_queue, 0x00,
             SIZE_UNORDERED_PROX_Q*sizeof(t_proximity_Q_entry) );
       return_val = 0;
    }

    // else find point in either queue : if found then delete
    else {
        for(i=0;i<SZ_ORDERED_PROX_Q;i++){
           p_entry = &(ordered_proximity_queue[i]);
           // if queue entry is in-use
           if(p_entry->tag_id == match_tag_id){
               // delete entry from the queue
```

```
                memset(p_entry, 0x00, sizeof(t_proximity_Q_entry));
                sort_by_ETA(ordered_proximity_queue); // sort by ETA
              return_val = 0;
              break;
            }
        }
        if(return_val){
            for(i=0;i<SZ_UNORDERED_PROX_Q;i++){
              p_entry = &(unordered_proximity_queue[i]);
             // if queue entry is in-use
              if(p_entry->tag_id == match_tag_id){
                  // delete entry from the queue
                  memset(p_entry, 0x00, sizeof(t_proximity_Q_entry));
                return_val = 0;
                break;
              }
            }
        }
    }
    return(return_val);
}

/*********************************************************/
// Function to check whether point is tripped/expired.
// If point is deleted then non-zero is returned, otherwise
// function returns zero.
/*********************************************************/
short check_point_tripped(t_proximity_Q_entry *p_entry,
            ULONG current_time){
ULONG distance = 0;  // records current distance
ULONG inner_radius, outer_radius; // used to implement hysteresis
short return_val;  // 1 if point was deleted
short point_tripped;  // 1 if point is tripped

    return_val = 0;
    point_tripped = 0;

    // check for time expiry of point
    if(p_entry->expiry_time < current_time){
      // delete point (set to all zero)
      memset(p_entry, 0x00, sizeof(t_proximity_Q_entry));
      return(1);
    }

    // Next check that it is a sufficiently long time since the
    // point was last tripped
    if ((current_time - p_entry->time_last_tripped) <
        TRIP_HYSTERESIS_TIME ){
      // still within hysteresis time : just return
      return(0);
    }

    // If you get to here then need to check for tripping of point

    // calculate current distance from point
    distance = calculate_distance( current_lat, current_lon,
      p_entry->center_lat, p_entry->center_lon);

    // allow a %age hysteresis where delete_when_tripped == 0
    // to prevent multiple rapid alerts for rover moving
    // tangentially to radius.
    if( delete_when_tripped == 0x00 ){
        inner_radius = p_entry->radius * ( 1 - D_HYSTERESIS/2);
        outer_radius = p_entry->radius * ( 1 + D_HYSTERESIS/2);
    } else {
        // use unadultered value if point will be deleted when tripped
        inner_radius = p_entry->radius;
```

```
            outer_radius = p_entry->radius;
        }

        // look for perimeter being traversed by an approaching
        // rover.  The previous cross of the perimeter must have
        // been in the opposite direction (to add hysteresis).
        if( (distance < inner_radius) &&    // current dist < radius
            ( (p_entry->direction_last_crossed == RECEDING) ||
              (p_entry->direction_last_crossed == NOT_YET_CROSSED) ) ){

          // record that the perimeter has been broken in this direction
          // this is used for future hysteresis calculations
          p_entry->direction_last_crossed = APPROACHING;

          // check if entry is activated by traverse in this direction
          if(p_entry->trip_direction & APPROACHING)
              point_tripped = 1;

        } else

        // look for perimeter being traversed by a receding
        // rover.  The previous cross of the perimeter must have
        // been in the opposite direction (to add hysteresis)
        if( (distance > inner_radius) &&    // current dist > radius
            ( (p_entry->direction_last_crossed == APPROACHING) ||
              (p_entry->direction_last_crossed == NOT_YET_CROSSED) ) ){

          // record that the perimeter has been broken in this direction
          // this is used for future hysteresis calculations
          p_entry->direction_last_crossed = RECEDING;

          // check if entry is activated by traverse in this direction
          if(p_entry->trip_direction & RECEDING)
              point_tripped = 1;

        }

        //check for tripping of point
        if(point_tripped){
            tx_prox_alert(p_entry->tag_id);  // transmit alert to server

            // check if point must be deleted
            if(delete_when_tripped){
              memset(p_entry, 0x00, sizeof(t_proximity_Q_entry)); // kill
              return_val = 1;  // so that calling function can resort
            }
        }

        return(return_val);
    }

    /*********************************************************/
    /* rover_ds.h                                            */
    /* Datastructure definitions for the rover for proximity */
    /* alert indications.                                    */
    /* S Taylor, @Road Inc.                                  */
    /*********************************************************/

    typedef unsigned long ULONG;  // 32 bit unsigned integer
    typedef unsigned short WORD;  // 16 bit unsigned integer
    typedef unsigned char BYTE;  // 8 bit unsigned integer

    // Bit patterns for days_to_trip field
    #define BP_SUNDAY    0x01
    #define BP_MONDAY    0x02
    #define BP_TUESDAY   0x04
    #define BP_WEDNESDAY 0x08
    #define BP_THURSDAY 0x10
```

```
#define BP_FRIDAY   0x20
#define BP_SATURDAY 0x40

// Bit patterns for direction variables
#define APPROACHING 0x01
#define RECEDING    0x02
#define NOT_YET_CROSSED 0x00

/************** Proximity queue entry ************************/
// Examples of field combinations for typical applications :
//
// For home / warehouse delivery :
//      center_lat, lon = destination
//      expiry_time = end of today
//      radius = 1-10 miles (in meters)
//      estimated_time_arrival = set by dispatcher reflects delivery order.
//      delete_when_tripped = 1
//      trip_when_head = 1
//      trip_direction = APPROACHING
//      Days to trip = either set to today, or to 0xFF
//      Description = customer name, address, order number
//
// For stolen vehicle alarm : (vehicle in normal use Mon-Fri)
//      center_lat, lon = truck depot
//      expiry_time = 0xFFFFFFFF
//      radius = X miles (in meters)
//      estimated_time_arrival = N/A since trip_when_head = 0
//      delete_when_tripped = 0
//      trip_when_head = 0  always trip
//      trip_direction = 0x02 when receding, or 0x03 either direction
//      Days to trip = 0x41 (binary) 01000001 (Sat and Sun)
//      Description = Stolen vehicle alarm (threshold distance + days)
//
// When downloading points from the server to the rover the data
// packet must give a value for each of these fields.  It is expected
// that the data structure in the server is a superset of the
// information in the data structure presented here.
// The server would store all information below, and would additionally
// store the contact information for the alert, whether, e-mail,
// telephone call, of page etc.  The server could additionally link
// this data to a database of customer orders, the format of which is
// beyond the scope of this file.
//
typedef struct proximity_Q_entry{

    ULONG tag_id;  // uniquely identifies an alert : 0 equals empty Q
entry
    char  description[SZ_PRX_TXT];  // for use of the driver
    long  center_lat;  // latitude of destination in MAS
    long  center_lon;  // longitude of destination in MAS
    ULONG radius;      // radius of trip in meters
    // all times in seconds, relative to 0/0/97
    ULONG estimated_time_arrival;  // used for sorting points in queue
    ULONG expiry_time;             // point is deleted at this time
    BYTE  delete_when_tripped;  // if 1 then point is deleted when tripped
                                // if 0 then point deleted at expiry time
only
    BYTE  trip_when_head;  // if 1 then point is only tripped when top of
queue
                           // if 0 then point is tripped whenever crossed
    BYTE  trip_direction;   // 0x01 : trip when rover is approaching
                            // 0x02 : trip when rover is receding
                            // 0x03 : trip in both directions
    BYTE  days_to_trip;     // bit field for 7 days of week, Sun-Sat

    // These final fields are not be downloaded from the server.
    // They are used locally in the rover to generate hysteresis
    // to prevent multiple trips on a point in rapid succession.
```

```
        // Hysteresis values may or may not be sent from the server
        // on a per-entry basis.
        ULONG time_last_tripped;  // time hysteresis
        BYTE  direction_last_crossed;  // direction in which rover was moving
                                       // when perimeter was last crossed
                                       // 0x01 : approaching
                                       // 0x02 : receding
                                       // 0x00 : has never been tripped (init)
} t_proximity_Q_entry;

// Extern declarations for two queues :
// one for points with trip_when_head == 1 (i.e. only check top entry)
// the second for points with trip_when_head == 0 (i.e. check every entry)
extern t_proximity_Q_entry ordered_proximity_queue[SZ_ORDERED_PROX_Q];
extern t_proximity_Q_entry unordered_proximity_queue[SZ_UNORDERED_PROX_Q];
```

Claims

1. An alert generating method comprising:

   providing to a mobile unit over a wireless network service connection information that identifies conditions for an alert, the wireless network service connection linking the mobile unit and a service center over a wide area network;
   monitoring in the mobile unit a position of the mobile unit;
   providing the service center a signal indicating that the conditions for an alert are satisfied; and
   alerting a designated location from the service center upon receiving the signal.

2. The method of claim 1, wherein the signal from the mobile unit is sent via the wireless network service connection.

3. The method of claim 1, wherein the alerting comprises telephoning the designated location.

4. The method of claim 1, wherein the alerting comprises sending e-mail to the designated location.

5. The method of claim 1, wherein providing the information that identifies the conditions for the alert, comprises downloading the information from the service center to the mobile unit.

6. The method of claim 5, further comprising downloading a destination list, including the information that identifies the conditions for the alert, to a web site corresponding to the service center.

7. The method of claim 1, wherein the information that identifies the conditions identifies an area around a destination for the mobile unit.

8. The method of claim 7, wherein the information that identifies the conditions indicates that the alert should be generated when the mobile unit enters the area around the destination.

9. The method of claim 7, wherein the information that identifies the conditions indicates that the alert should be generated when the mobile unit leaves the area around the destination.

10. The method of claim 7, wherein the information includes a location and a threshold radius that respectively correspond to a center and a radius of the area surrounding the destination.

11. The method of claim 1, further comprising selecting a selected destination for the mobile unit, wherein the information that identifies the conditions for the alert, requires that the selected destination be a destination that is identified in the information.

12. The method of claim 11, wherein an operator of the mobile unit selects the selected destination when the mobile unit is directly proceeding to the selected destination.

13. A delivery method comprising:

creating a list of destinations for deliveries at a service center, the list including a threshold distance for one or more destination for which an alert should be generated;
downloading a portion of the list of destinations to a mobile unit installed in a delivery vehicle, the downloading being effectuated over a wireless network connection which links the mobile unit to the service center over a wide area network;
selecting a destination from the list as a next destination for a delivery vehicle;
monitoring distance between the delivery vehicle and the selected destination; and
generating an alert from the delivery vehicle when the distance is less than a threshold distance.

14. The method of claim 13, wherein generating the alert comprises:

sending a message from the delivery vehicle to the service center, the message including a tag identifying the destination;
looking-up a designated location that corresponds to the destination; and sending the alert from the service center to the designated location.

15. A mobile unit comprising:

a location system;
a wireless device linking the mobile unit with a service center over a wireless network connection of a wide area network; and
a control circuit, wherein the control system automatically activates the location system to determine a current location of the mobile unit, determines whether the mobile unit has crossed a threshold, and activates the wireless device to send an alert signal if the mobile unit has crossed the threshold.

16. The mobile unit of claim 15, wherein the location system is a GPS receiver.

17. The mobile unit of claim 15, wherein the wireless device is a wireless modem.

18. The mobile unit of claim 15, wherein the wireless device is an attached data-capable cellular telephone.

19. The mobile unit of claim 15, wherein the control circuit determines whether the mobile unit has crossed the threshold by determining whether the current location of the mobile unit is within an alert area.

20. The mobile unit of claim 19, wherein the control circuit calculates a distance between the current location and a central point in the alert area and determines whether the distance is less than a threshold distance associated with the alert area.

21. A system comprising:

a data connection to wide area network;
an alerting device; and
a service center connected to the data connection to enable receipt of messages from a mobile unit over a wireless network connection and connected to the alerting device to enable the service center to activate the alerting device and send alerts, the service center maintaining contact information for the mobile unit, wherein in response to a signal from the mobile unit, the service center activates the alerting device to send an alert to a designated location identified in the contact information.

22. The system of claim 21, wherein the service center comprises a server that permits internet access to the service center for setting of the designated location to which the alerting device sends the alert.

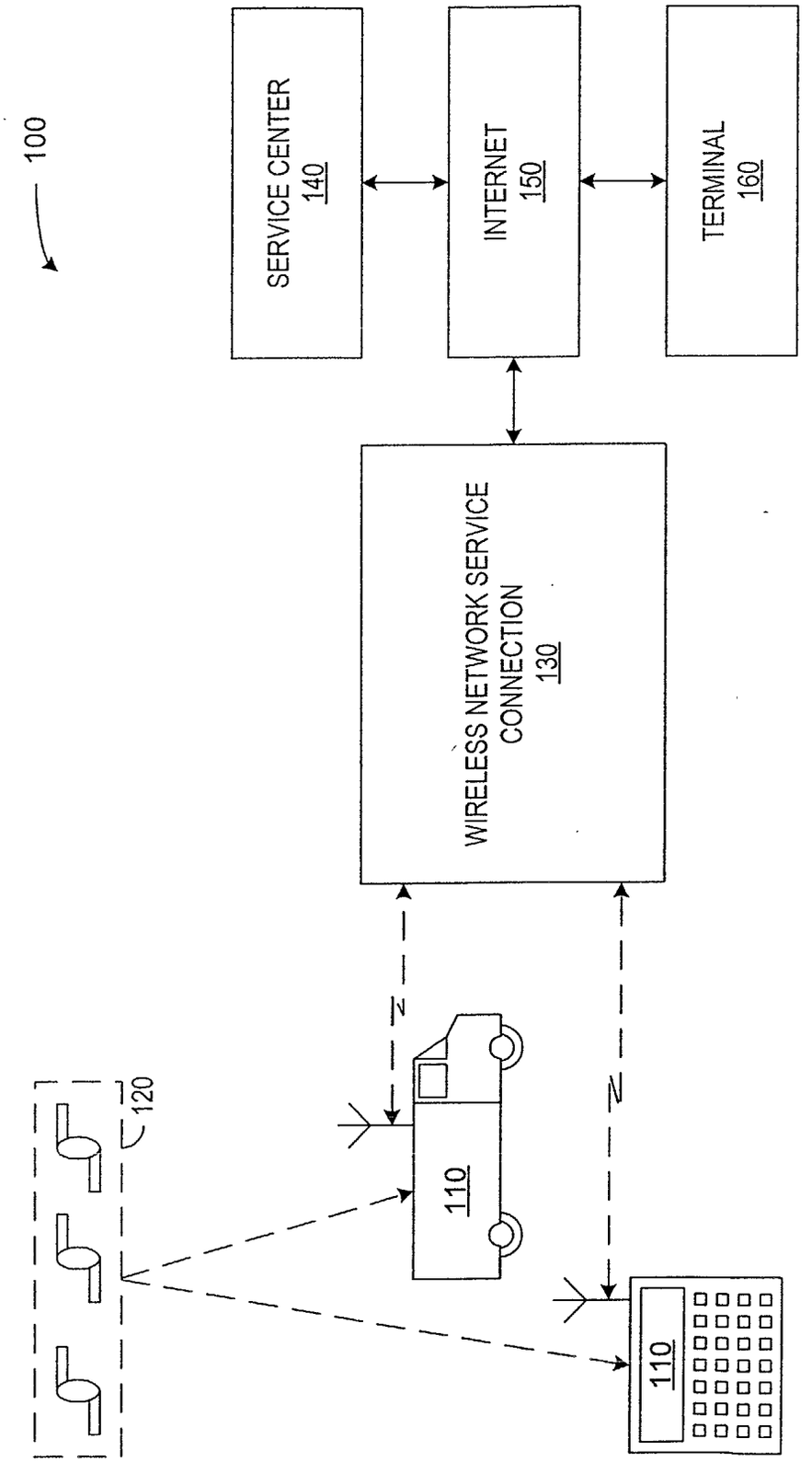
100
SERVICE CENTER
140
INTERNET
150
TERMINAL
160
WIRELESS NETWORK SERVICE CONNECTION
130
120
110
110

FIG. 1

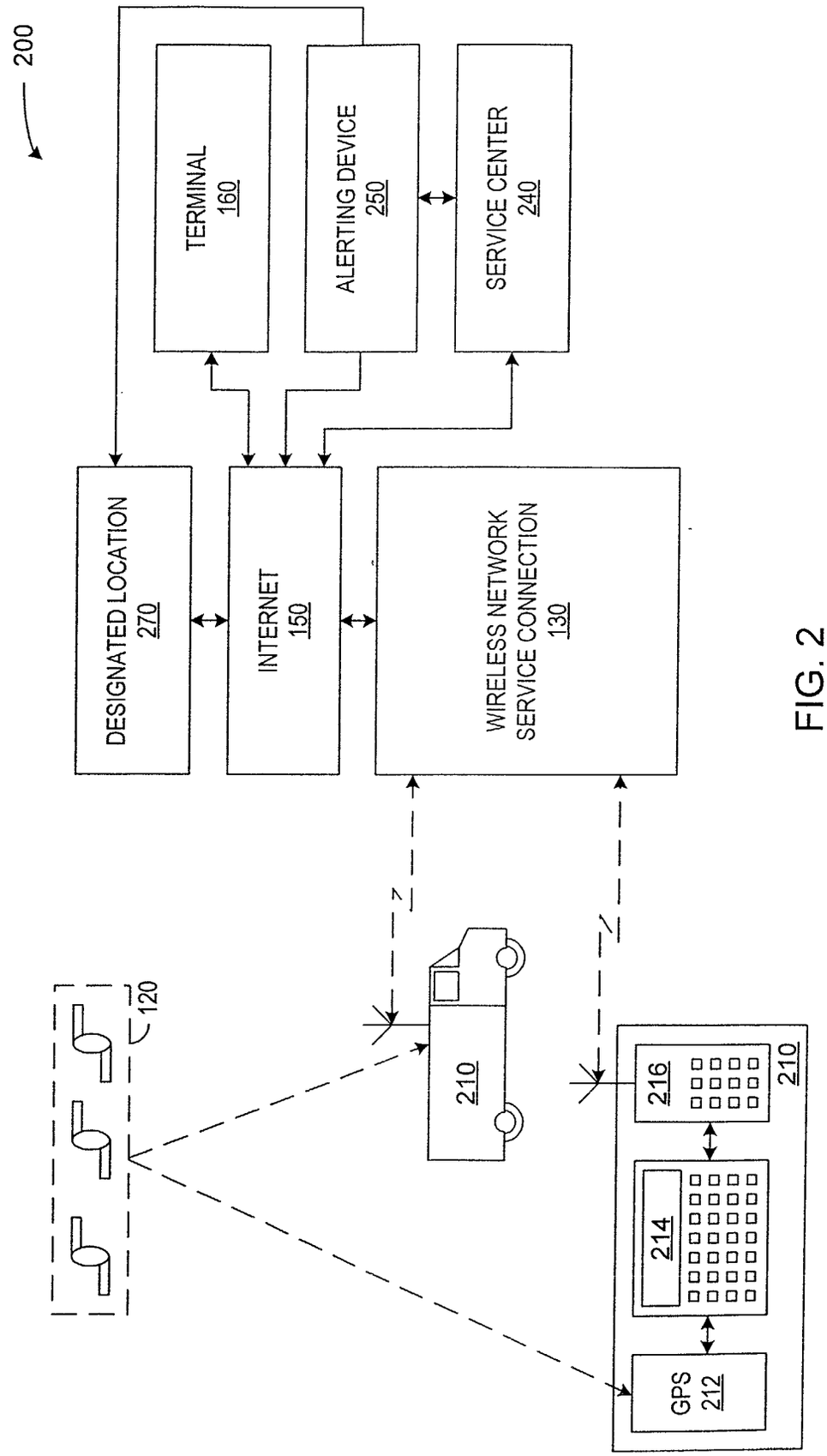
200
TERMINAL
160
ALERTING DEVICE
250
SERVICE CENTER
240
DESIGNATED LOCATION
270
INTERNET
150
WIRELESS NETWORK
SERVICE CONNECTION
130
120
210
210
216
214
GPS
212

FIG. 2

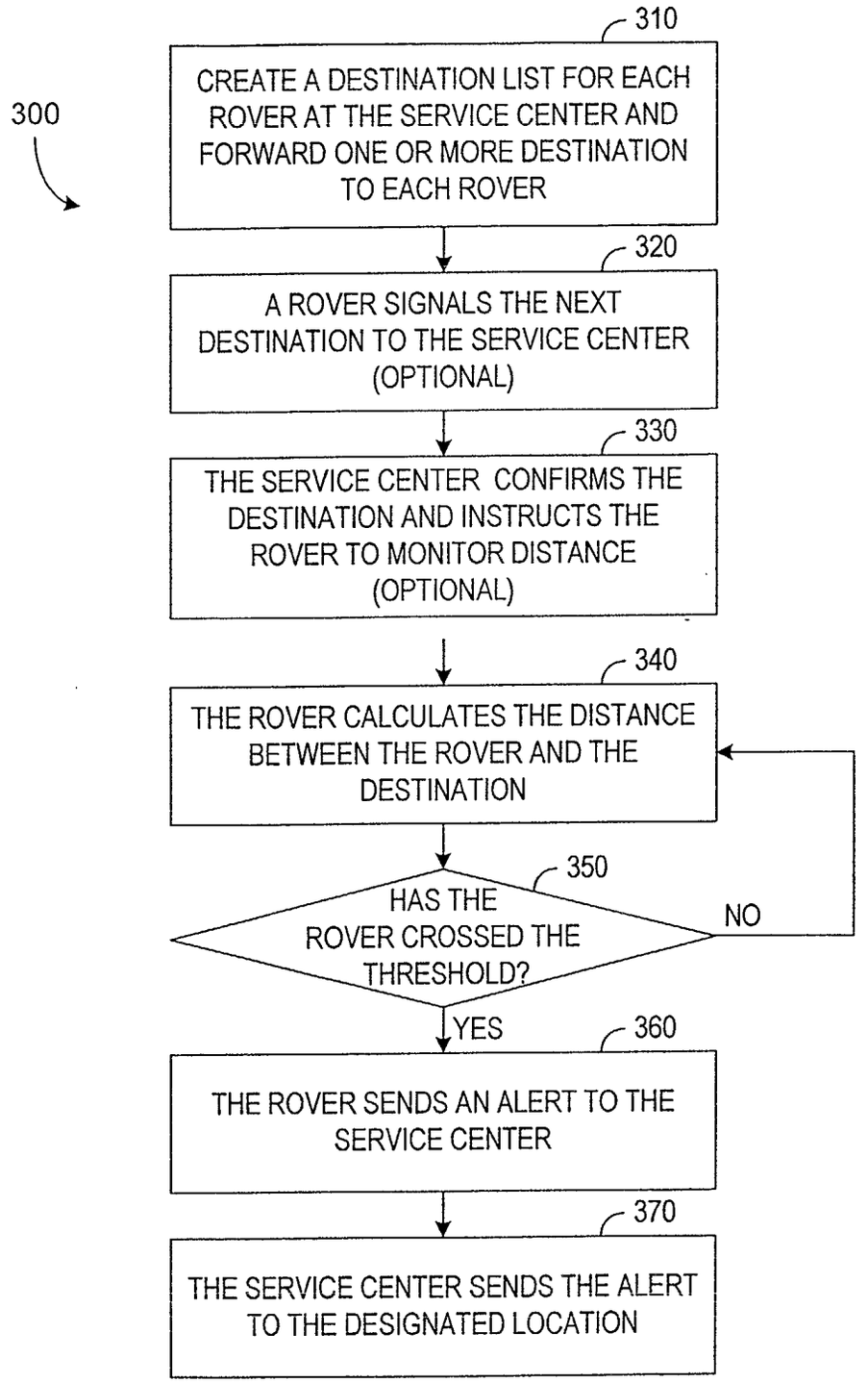
300
310
CREATE A DESTINATION LIST FOR EACH ROVER AT THE SERVICE CENTER AND FORWARD ONE OR MORE DESTINATION TO EACH ROVER
320
A ROVER SIGNALS THE NEXT DESTINATION TO THE SERVICE CENTER (OPTIONAL)
330
THE SERVICE CENTER CONFIRMS THE DESTINATION AND INSTRUCTS THE ROVER TO MONITOR DISTANCE (OPTIONAL)
340
THE ROVER CALCULATES THE DISTANCE BETWEEN THE ROVER AND THE DESTINATION
350
HAS THE ROVER CROSSED THE THRESHOLD?
NO
YES
360
THE ROVER SENDS AN ALERT TO THE SERVICE CENTER
370
THE SERVICE CENTER SENDS THE ALERT TO THE DESIGNATED LOCATION

FIG. 3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 01 30 8997

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	US 6 124 810 A (DE PAOLO KATHLEEN R ET AL) 26 September 2000 (2000-09-26) * column 3, line 53 - line 60 * * column 5, line 52 - line 62 * * column 9, line 9 - column 10, line 27; figure 1 *	1-22	G08G1/127
X	US 5 657 010 A (JONES MARTIN KELLY) 12 August 1997 (1997-08-12) * column 4, line 27 - line 36 * * column 5, line 52 - line 57 * * column 6, line 29 - column 8, line 15 * * column 9, line 24 - line 33 *	1-22	
A	WO 00 13339 A (QUALCOMM INC) 9 March 2000 (2000-03-09) * page 5, line 34 - page 6, line 12 *	1-22	
A	US 5 987 377 A (KENNEDY III WILLIAM C ET AL) 16 November 1999 (1999-11-16) * abstract *	1-22	
			TECHNICAL FIELDS SEARCHED (Int.Cl.7)
A	US 5 719 771 A (HAUPT DAVID E ET AL) 17 February 1998 (1998-02-17) * abstract *	1-22	G08G G01S
A	EP 0 494 499 A (ROCKWELL INTERNATIONAL CORP) 15 July 1992 (1992-07-15) * abstract *	1-22	
A,D	US 5 959 577 A (FAN RODRIC C ET AL) 28 September 1999 (1999-09-28) * abstract *	1-22	
A	US 5 648 770 A (ROSS JOHN) 15 July 1997 (1997-07-15) * abstract *	1-22	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
MUNICH	21 January 2002	Johansson, R

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 01 30 8997

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

21-01-2002

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6124810 A	26-09-2000	AU 6045999 A	03-04-2000
		BR 9906949 A	03-10-2000
		CN 1277706 T	20-12-2000
		EP 1031123 A1	30-08-2000
		WO 0016293 A1	23-03-2000
US 5657010 A	12-08-1997	US 5400020 A	21-03-1995
		US 6278936 B1	21-08-2001
		US 6313760 B1	06-11-2001
		US 5668543 A	16-09-1997
		US 5623260 A	22-04-1997
WO 0013339 A	09-03-2000	US 6181931 B1	30-01-2001
		AU 5903499 A	21-03-2000
		WO 0013339 A1	09-03-2000
US 5987377 A	16-11-1999	US 5724243 A	03-03-1998
		AU 4858396 A	27-08-1996
		CA 2212330 A1	15-08-1996
		EP 0808492 A1	26-11-1997
		JP 11505642 T	21-05-1999
		WO 9624915 A1	15-08-1996
US 5719771 A	17-02-1998	US 5475597 A	12-12-1995
EP 0494499 A	15-07-1992	US 5068656 A	26-11-1991
		CA 2056045 A1	22-06-1992
		DE 69121872 D1	10-10-1996
		DE 69121872 T2	17-04-1997
		EP 0494499 A2	15-07-1992
		MX 9102201 A1	01-06-1992
US 5959577 A	28-09-1999	NONE	
US 5648770 A	15-07-1997	US 5444444 A	22-08-1995
		AU 5538094 A	12-12-1994
		GB 2293071 A ,B	13-03-1996
		WO 9427264 A1	24-11-1994
		US 5673305 A	30-09-1997

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82