ER-400 AGV

Mobile Robot

User Manual

Catalog #100394-D

March 2011

intelitek

Catalog #100394-D
March 2011

Every effort has been made to make this book as complete and accurate as possible. However, no warranty of suitability, purpose, or fitness is made or implied. Intelitek is not liable or responsible to any person or entity for loss or damage in connection with or stemming from the use of the software, hardware and/or the information contained in this publication.

Intelitek bears no responsibility for errors that may appear in this publication and retains the right to make changes to the software, hardware and manual without prior notice.

INTELITEK INC.
444 East Industrial Park Drive
Manchester NH 03109-5317
Toll-free (sales): (800) 777-6268
Tel: (603) 625-8600
Fax: (603) 625-2137
Web site: www.intelitek.com
e-mail address: info@intelitek.com

Intelitek ER Series Limited Warranty

Intelitek warrants to the original purchaser that the ER series ‘Product’ is free from defects in materials and workmanship when used under normal purposes for a period of 12 months, beginning from the date of purchase. Product accessories, including replacement batteries, are warranted for a period of ninety days from the date of purchase. This warranty provides for the cost of parts and labor to repair covered defects when performed by an authorized Intelitek service and warranty facility. A valid proof of purchase is required for warranty repairs. The limited warranty does not cover transportation costs of any kind. The owner bears all responsibility for transportation costs to an authorized Intelitek service and warranty facility.

Items and Conditions Not Covered

This express warranty does not cover the following:

- Any product or part that has been altered, misused, abused or requires replacement or repair due to accidents or lack of proper maintenance.
- Normal wear and tear including fading of paint or plastic parts.
- Any costs or damages associated with improper use of product.
- Any product that has been opened, repaired, modified or altered by anyone other than a Intelitek authorized repair facility.
- Repairs necessary due to improper battery care and/or improper charging process such as charging in wet conditions, electrical supply irregularities, or failure to properly prepare the robotic mobile robot or battery prior to any period of non-use.
- Repairs necessary due to water damage or any acts of God.

Instructions for Obtaining Warranty Service

Should you feel your Intelitek product contains a defect in materials or workmanship, contact the retailer who sold you the product.

Owner Responsibilities

You must maintain and care for your Intelitek product by following the maintenance and care procedures described in this manual. Routine maintenance, whether performed by a service provider or by you, is at your expense.

General Conditions

Repair by an authorized Intelitek service and warranty repair facility is your sole remedy under this warranty. There is no other express or implied warranty. All implied warranties of merchantability and fitness for use are limited to the duration of this express warranty. Intelitek is not liable for indirect, incidental or consequential damages in connection with the use of the Intelitek Product covered by this warranty, including any cost or expense of providing substitute equipment or service during reasonable periods of malfunction or non-use pending completion of repairs under this warranty. Some states do not allow exclusions of incidental or consequential damages, or limitations on how long an implied warranty lasts, so the above exclusion and limitations may not apply to you. This warranty gives you specific legal rights, and you may also have other rights, which vary from state to state.

Always follow the safety instructions specified in this Manual

Intelitek® products are manufactured by F. Robotics Acquisitions Ltd.

Table of Contents

1

General Information

The instructions in this manual are for the ER-400 AGV (Automated Guided Vehicle) Mobile Robot only.

Description

The ER-400 Mobile Robot is a versatile and cost-effective Automated Guided Vehicle (AGV). With its open architecture, superior performance, and complete reliability, the ER-400 performs either independently or integrated as part of a multi-robot application. It is ideal for a wide variety of research projects and functional applications.

The ER-400 offers a ready-made platform for student projects, advanced robotic education, and competitions. Research topics are limited only by the student's imagination: artificial intelligence, digital communications, path planning, real-time programming, and remote viewing and operation. The ER-400 provides a highly versatile alternative means of moving parts within the workspace. Its capabilities are perfect for prototyping heavy applications such as hazardous material handling, agricultural manufacturing, rescue, security and military uses.

The open architecture of the ER-400 allows for the integration of limitless advanced technologies, including GPS, gyroscope, compass, laser, and stereo vision. This expands the unit's uses far beyond its original incarnation as an automated vacuum cleaner.

For autonomous navigation and obstacle avoidance, the ER-400 uses various sensor technologies such as infrared and ultrasound. These features are operational both when the ER-400 is controlled remotely and when it functions independently.

System Overview

The AGV System consists of three basic parts:

- ER-400 AGV Mobile Robot
- On-board laptop PC (not provided by Intelitek)
- Optional Desktop workstation PC (not provided by Intelitek)

Safety

READ ALL INSTRUCTIONS BEFORE USING THIS DEVICE

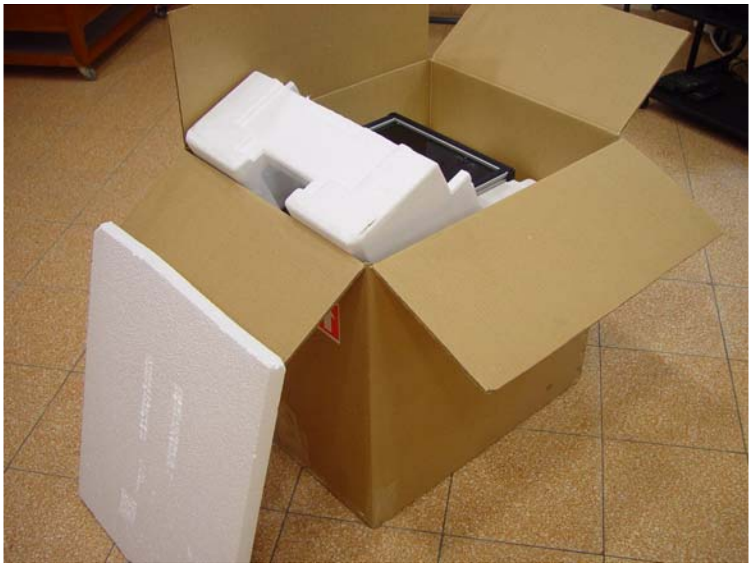
WARNING: To reduce the risk of fire, electric shock, or injury:

- Do not use outdoors or on wet surfaces.
- Close supervision is necessary when any appliance is used by or near children.
  Do not allow the robot to be used as a toy or operated in an area with unsupervised children.
- Keep hair, loose clothing, fingers, feet and all parts of body away from openings and any moving parts. Do not operate Robot in bare feet or when wearing open-toe shoes or sandals.
- Do not use in a room with any form of open flame, such as lit candles, burning fireplaces, or lit oil lamps.
- Do not use to pick up flammable or combustible materials such as gasoline.
- Ensure that the floor of the mobile robot's workspace is completely free of electrical cords, wires and cables. Secure all such obstacles so that they cannot be accidentally moved.
- Before servicing the mobile robot, always turn off all controls and remove the batteries.
- Remove batteries from the mobile robot if it will not be used for periods of time greater than two weeks. Keep batteries fully charged for storage.
- Do not unplug by pulling on charger cord. To unplug, grasp the plug, not the cord. Never handle plug, battery or appliance with wet hands.
- Do not use with damaged charger cord, plug or batteries. If appliance is not working as it should, has been dropped, damaged, left outdoors, or dropped into water, take it to your dealer.
- Use mobile robot only for its intended use as described in the instructions.
- Use only attachments recommended by Intelitek; others may cause hazards.
- This product contains a sealed lead acid battery. Battery must be recycled properly and must not be incinerated.
- Do not incinerate this appliance even if it is severely damaged. The batteries can explode in a fire.
- Avoid touching the electrical terminals on the batteries.
- Keep the charger and batteries away from heated surfaces.
- Use only the batteries and charger supplied by the manufacturer.
- Do not sit or stand on this device.
- Hold the Pull-Line when rewinding. Do not allow the Pull-Line to whip when rewinding.
- Always use the carry handle to lift the unit, not the bumper guard.

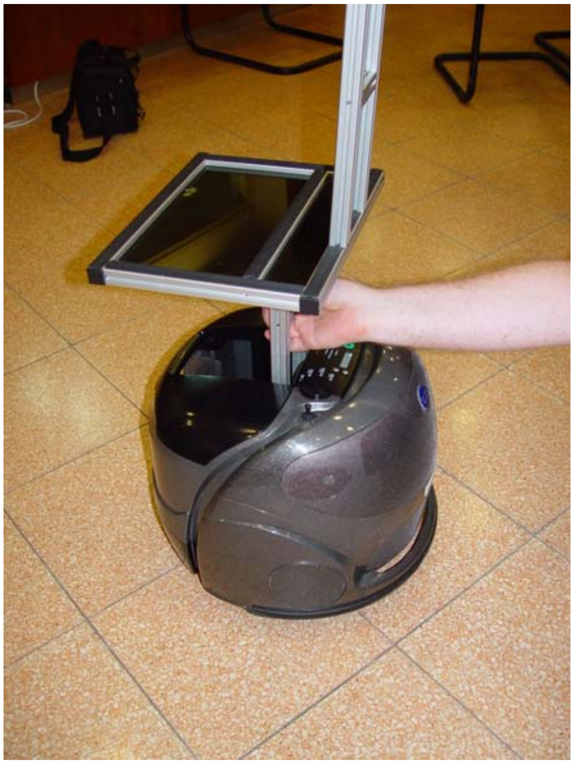
Unpacking

Use care when unpacking your new ER-400 Mobile Robot. Do not cut the carton.

1. Remove small parts from the top of the shipping carton.
2. Gently remove the large, flat Styrofoam packing insert from the front of the box, and remove the two foam spacer bars from underneath the On-Board PC shelf.



3. There are two large pieces of formed Styrofoam protecting the body of the Mobile Robot. Tilt the smaller of the two pieces toward you and guide it gently out past the On-Board PC shelf.



4. Gently lift the Mobile Robot out of the carton. Use only the aluminum lifting handle to support the weight of the Mobile Robot. Do not put any pressure on the Bump Guard (also called Bumper or Bump Sensor), which occupies most of the front of the unit. See page 6 for a diagram identifying the parts.
5. Gently tip the unit onto its back. Remove the two protective plastic strips from the Bumper Guard.

Inspection for Acceptance

Verify that you have received all parts listed in the Parts List on page 5. If anything is missing, contact your supplier immediately.

Inspect all parts for scratches, cracks or dents. If you find any damage, contact your shipping company and the manufacturer before attempting to install or operate the system.

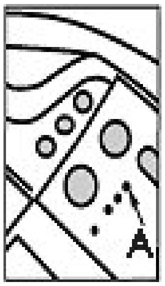
Installation of Batteries

The battery compartment is protected by a plastic dust cover. Squeeze the latch and lift the cover to its Open position. There is a small utility compartment in the middle, with a place for one battery on each side.

The batteries are keyed so that they will only go in one way. At the top of the key, you can see the battery fuse. This side is turned toward the center of the Mobile Robot unit. Gently slide the batteries into place. When the both batteries are installed properly, the unit plays a short, audible melody.

If you do not hear the melody, either the batteries are not installed properly, or they need to be charged. If the system still does not respond after charging, contact Intelitek technical support.

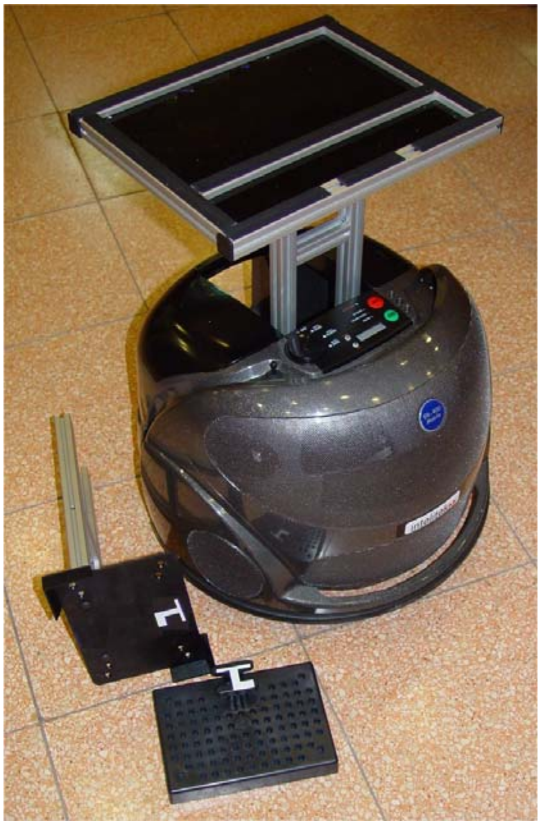
Battery Charging



Manually plug the AC Power Supply into the battery charger port, located in the bottom center of the back of the unit. While the charger is working, the Battery Light (indicated by Arrow A) flashes green.

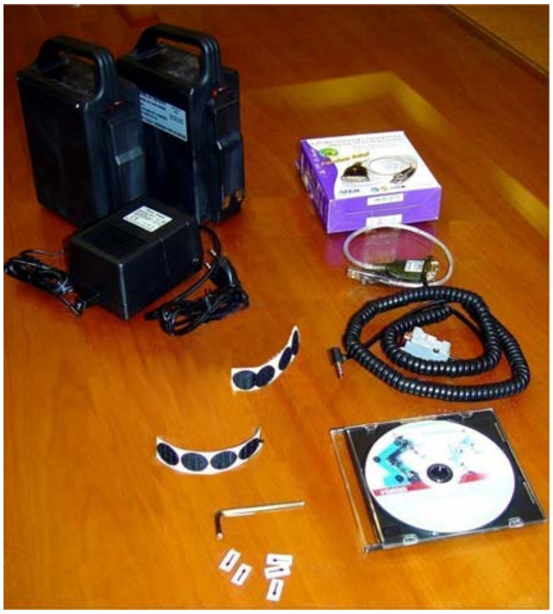
The typical time to fully charge the batteries is 8 to 12 hours.

Parts List



A. ER-400 Mobile Robot
   - Assembled with On-Board PC Shelf

B. Parts for Payload Stand
   - Payload Shelf, assembled with 2 supports, with 2 screws (M5x20) for attaching to On-Board PC Shelf
   - One plastic template



C. 2 Batteries

D. AC Power Supply:
   1. 220V Europe
   2. 220V UK
   3. 110V USA

E. Serial Communication Cable (RS-232) and USB to RS-232 Adapter

F. 3mm Allen wrench

G. Packet of adhesive arrows

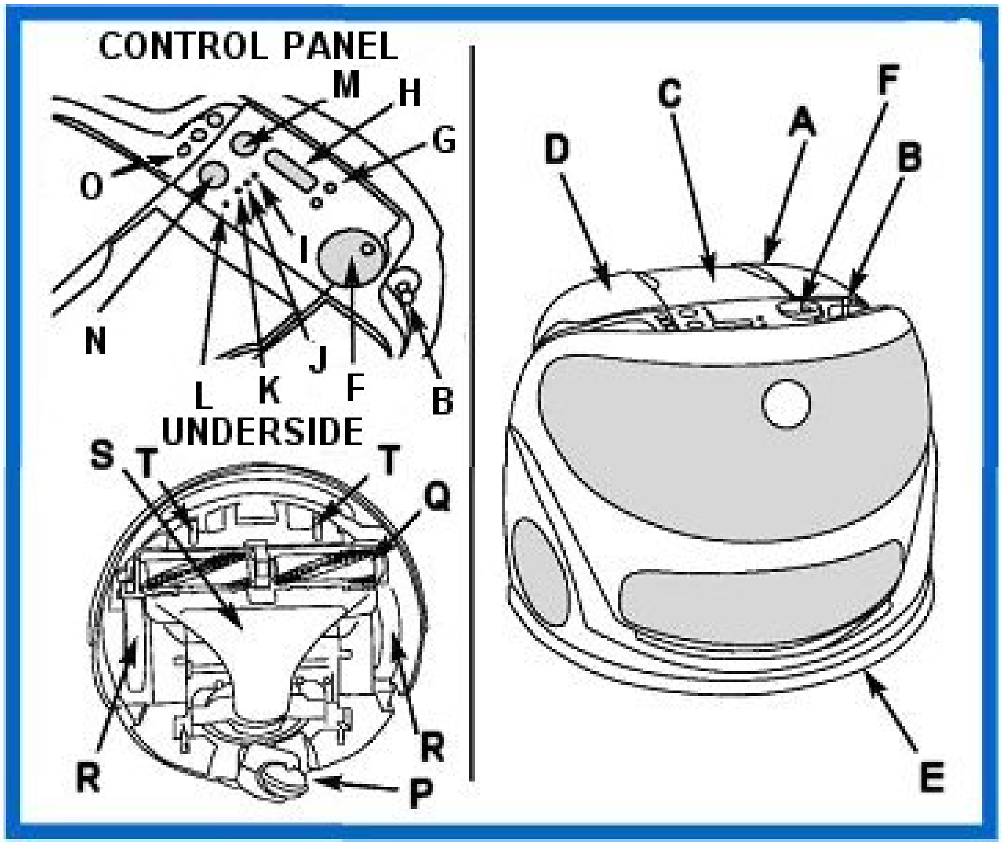
H. Velcro strips

I. CD-ROM

J. This User Manual

Identification of Parts

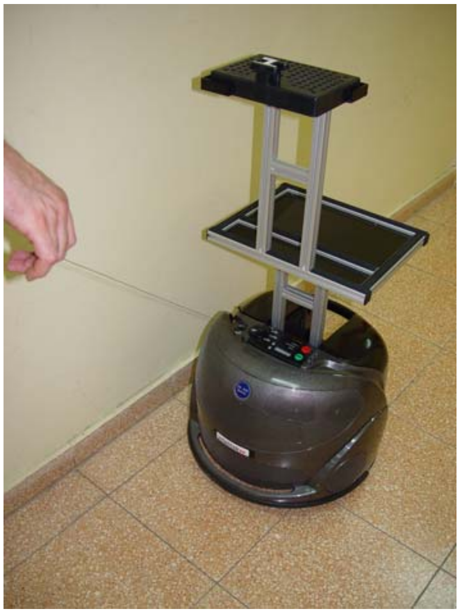
The following diagram will help you identify the various parts and controls on the body of the ER-400 AGV Mobile Robot. Several of the parts are used only when the unit is configured as an automatic vacuum cleaner.



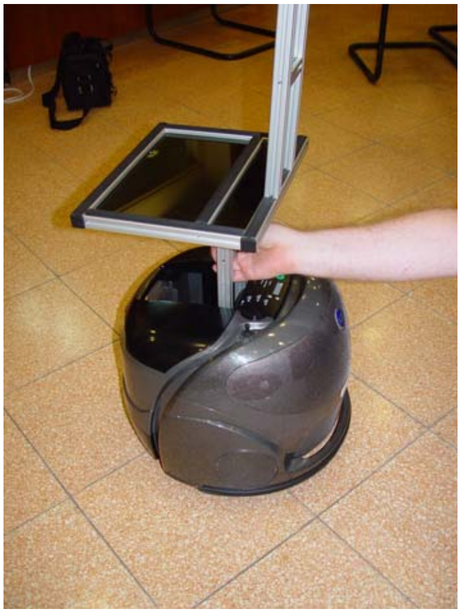
- A. Carry Handle
- B. Pull-Line Cord
- C. (removed)
- D. Battery Compartment (under handle)
- E. Bump Guard
- F. Operation Mode Selector
- G. Scroll Buttons
- H. Display Window
- I. Battery Indicator Light
- J. Check Display Light
- K. Stand-by Light
- L. Child Lock Light
- M. Start Button
- N. Stop Button
- O. Door and Table Sensors (disabled)
- P. Caster
- Q. Vacuum Brush
- R. Drive Wheels
- S. Intake Nozzle
- T. Stair Sensors

Handling Instructions

There are several ways to move the AGV by hand, without the use of computer control programs.



1. If the unit is functional and the batteries have power, you can move it by towing it with the Pull-Line cord (also known as the Leash). The Pull-Line cord has four positions: OFF (fully retracted), FORWARD (fully extended in the AGV's forward direction), RIGHT (fully extended to the right of the AGV) and LEFT (fully extended to the left of the AGV). Turn the Operation Mode Selector to **ON** (position 1). To move the AGV, pull on the cord in the direction desired, and the AGV will move at a pre-set speed. To stop, release the tension on the cord. Do not allow the spring-loaded retractor mechanism to whip the cord back into its retracted position. This avoids jamming or other damage. Instead, hold the end of the cord and guide it manually until it is fully retracted.



2. The AGV may be lifted and moved using the handle portion of the payload stand, as shown in the illustration. This is the safest way to move the AGV without using its motors.
3. If you use a hand-truck, be sure not to allow any pressure to be placed on the Bump Guard.

You must always be very careful not to apply any pressure to the Bump Guard, which covers the entire front of the AGV. The Bump Guard is a delicate moving part and is not designed to support the entire weight of the unit. Similarly, you must never use the Bump Guard as a handle for lifting.

Specifications

Physical Characteristics

- Dimensions:
  - Diameter: 43 cm
  - Height without payload stand: 33 cm
  - Height with payload stand: 87 cm
- Weight: 19 kg (including two 3-kg batteries)
- Payload: Up to 1.0 kg

Power

- Two 12V, 7AH, sealed, lead acid, rechargeable batteries
- Average work time per charge: 5 hours of continuous operation
- Charging time: 8-12 hours
- Battery life: 250 hours of operation

Controller

- Hitachi microcontroller

Drive

- 2-wheel drive – two 2-Amp motors with odometers
- Rear caster with elevation sensors
- Maximum speed: 40 cm/sec
- Minimum turn radius: 0 cm

Communications

- RS-232 serial port, with USB adapter
- 5 IR receivers for remote control
- TCP/IP (from on-board PC or Pocket PC)

Devices Available for User Programs

- Analog and digital I/O
- 3 PWM connections

Control Software

- Workstation: SCORBASE for Mobile Robot, version 5.0.1, or newer
- On-board PC: AGV Device Driver, version 1.0.0.17, or newer
- Diagnostic Toolkit, version 1.1, or newer
- On-board control software, version 06.06.05, or newer

Sensors

- 10 ultrasonic sensors with resolution ±5mm
- 18 analog sensors, including 1 optical Position Sensitive Detector (PSD)
- 9 infrared sensors
- 15 digital sensors

Safety Features

- Touch-sensing bumper – responds to force of less than 100 g
- 3 stair detectors - robot stops operation upon detecting a stair
- CE-approved

On-Board Laptop PC Specifications

The on-board PC must have the following minimum characteristics:

- **Weight**: less than 4 kg
- **Physical dimensions**: maximum 23.5 cm x 31.5 cm
- **Operating System**: Windows XP, Windows Vista or Windows 7.
- **Processor**: in accordance with the requirements for the version of Windows you will be running.
- **Memory**: in accordance with the requirements for the version of Windows you will be running.
- **Removable drive**: CD-ROM drive
- **Hard drive**: minimum 20 MB of free disk space.
- **Ports**:
  - USB (or RS-232)
  - Wireless LAN
- **Battery life**: minimum 2 hours
- **Color**: black (important only for integration into Open CIM)

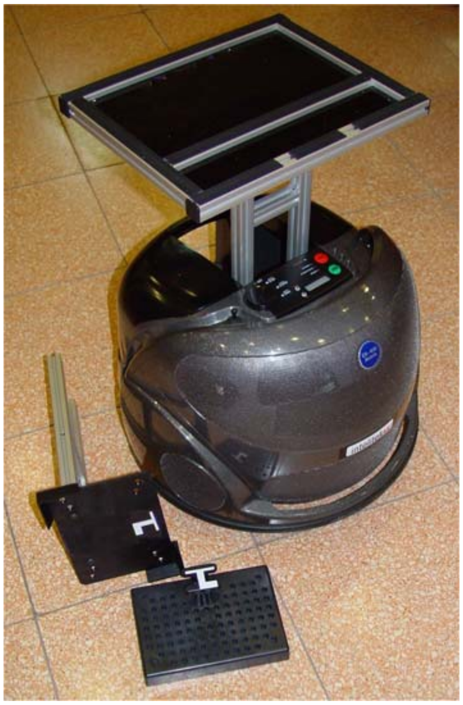
Desktop Workstation PC Specifications

- **Processor**: in accordance with the requirements for the version of Windows you will be running.
- **Memory**: in accordance with the requirements for the version of Windows you will be running.
- **Hard drive**: minimum 60 MB of free disk space.
- **Operating System**: Windows XP, Windows Vista or Windows 7.
- **Monitor**: A Super VGA or better graphics display, minimum 256 colors.
- **Mouse** or other pointing device.
- **Removable drive**: CD-ROM drive
- **Ports**:
    - USB (or RS-232)
    - Wireless LAN

2

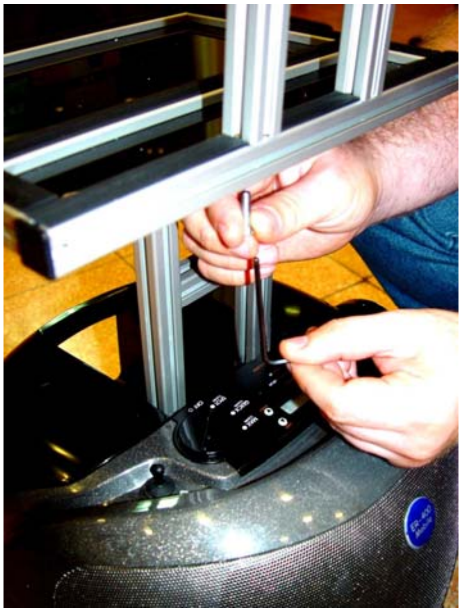
Assembly and Installation

Assembly of Payload Stand



The ER-400 comes with a specially-engineered stand to support an on-board laptop computer and a payload. The stand includes a horizontal crossbar to be used as a handle for lifting, a large shelf for the laptop computer, and a small shelf for the payload.

The Mobile Robot is shipped with the on-board PC shelf already attached to the body of the robot. The Payload Shelf must be installed by the user.



Attach the Payload Shelf supports to the top of the on-board PC shelf using the two M5x20 Allen screws provided. The Payload Shelf itself comes already assembled with its two vertical supports. The screws must be driven with an Allen wrench. Hold the Allen wrench as shown, using only the short end to tighten the screws. Be careful to avoid overtightening.

Note that, if desired, the Payload Shelf can also be installed at an orientation 180 degrees from what is illustrated.

If desired, install a non-skid surface for the on-board PC to rest on, or secure it in place with Velcro strips or a strap, belt, or elastic band.

Attach two adhesive arrows to opposite sides of the AGV, and two more to the floor at the AGV's home location. See page 22 for more detailed instructions.

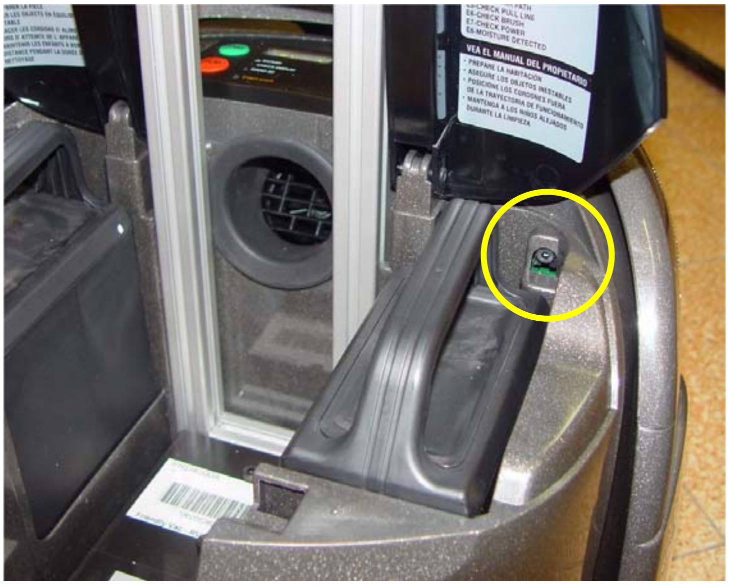
Installation of On-Board Computer

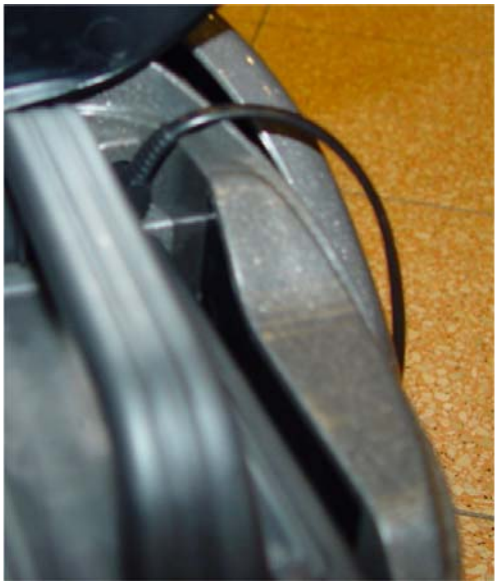
Follow the manufacturer’s instructions for the setup and installation of the on-board laptop computer. Charge the batteries, if necessary.

Secure the laptop in place on the on-board computer shelf. Open it and boot it up.

Connection to On-Board PC

Make sure that the ER-400 is not charging, and that the Mode Selector is in the **PC Control** or SPOT position (Position 1). One end of the serial communications cable resembles an audio plug. Plug it into the ER-400. The connection is located above the right-hand battery. Lift the left-hand battery and re-set it to force a reset of the ER-400 system. The unit goes through a Power-On Self-Test for about 4 seconds, and plays a short melody to indicate success.





Connect the other end of the RS-232 cable to the RS-232/USB Adapter provided with your robot. Tighten the screws with a small slotted-head screwdriver. Finally, plug the USB connector into the appropriate port on the on-board computer.

Note: if your on-board laptop computer does not have a USB connection, consult Intelitek technical support for the proper installation procedure.

Connection to Network

The user must have some means of connecting the AGV Mobile Robot to the workstation. In a small system, this may be accomplished by a simple peer-to-peer connection by means of wireless interface cards (not provided by Intelitek).

If you intend to integrate the AGV into a CIM system or other LAN network, you must use a Wireless Access Point and enable all necessary devices.

If you want to control the AGV from a desktop computer, you can operate it within the reach of the flexible USB cable provided. However, it will be more convenient to operate it over a wireless connection.

Software Installation

Software Overview

Generally speaking, software which can control your ER-400 Mobile Robot falls into three categories:

- Software for the on-board laptop PC.
- Software for a single workstation that controls the ER-400 Mobile Robot.
- Software for a master control computer that manages several independently-functioning workstations.

The software bundle provides a basic working system on the first two levels. To provide you with the greatest flexibility, it is designed to install the same set of software on each computer in your network. On-screen instructions simplify the task of installation.

AGV Device Driver

The AGV Device Driver is the software that directly controls the AGV (Automated Guided Vehicle), also called the Mobile Robot. It is normally run from the on-board laptop PC, but can also be run from a desktop PC. The communications cable provided with the Mobile Robot is long enough to permit this type of operation, if desired.

SCORBASE

SCORBASE for Mobile Robot is a programmable software control package. It is designed to run on a desktop computer, issuing its commands over the wireless LAN to the AGV Device Driver. However, it can also be run on the on-board laptop PC, sending the commands directly to the AGV Device Driver running on the same computer.

Other Software

Other software can be substituted at any of the three levels, as long as the programmer knows which commands are expected as input and output. Examples of available software programs include:

- MATLAB
- Lab View
- CIM Manager

Examples of suitable programming languages include the following:

- C#/C/C++
- Visual Basic
- Java

Please contact Intelitek Sales for current availability of optional software to work with the ER-400.

On-board (laptop) software

ATEN USB to Serial Bridge

This software contains the drivers for the RS-232/USB adapter. It is only needed if you are using the USB port on your on-board laptop or your desktop PC.

Make all the physical connections and wait to See whether Windows detects a new USB device. If it succeeds, make note of the COM port which is assigned to the device. Restart the computer.

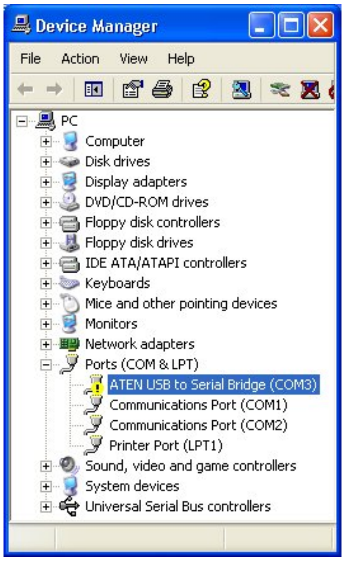
To install the adapter driver manually from the CD-ROM, follow the on-screen instructions. You may also download the driver from the manufacturer’s website:

http://www.aten-usa.com/?support_df&Item=UC232A#display

Refer to the manufacturer’s user manual or contact Intelitek technical support for more information.

Software Configuration (USB)

The AGV Device Driver communicates with the Mobile Robot through a COM port. However, the AGV-DD is not able to automatically detect which port is being used by the ATEN USB to Serial Bridge. The following procedure ensures correct identification of the COM port and configuration of the AGV Device Driver.



On the on-board PC, open the Control Panel. Click on the System icon to access the System Properties dialog box. Click on the Hardware tab. In the Device Manager subpanel, click the button for Device Manager. This opens up a directory tree. Select the node that reads PORTS and open it. Take note of the COM port number that has been assigned to the ATEN USB to Serial Bridge.

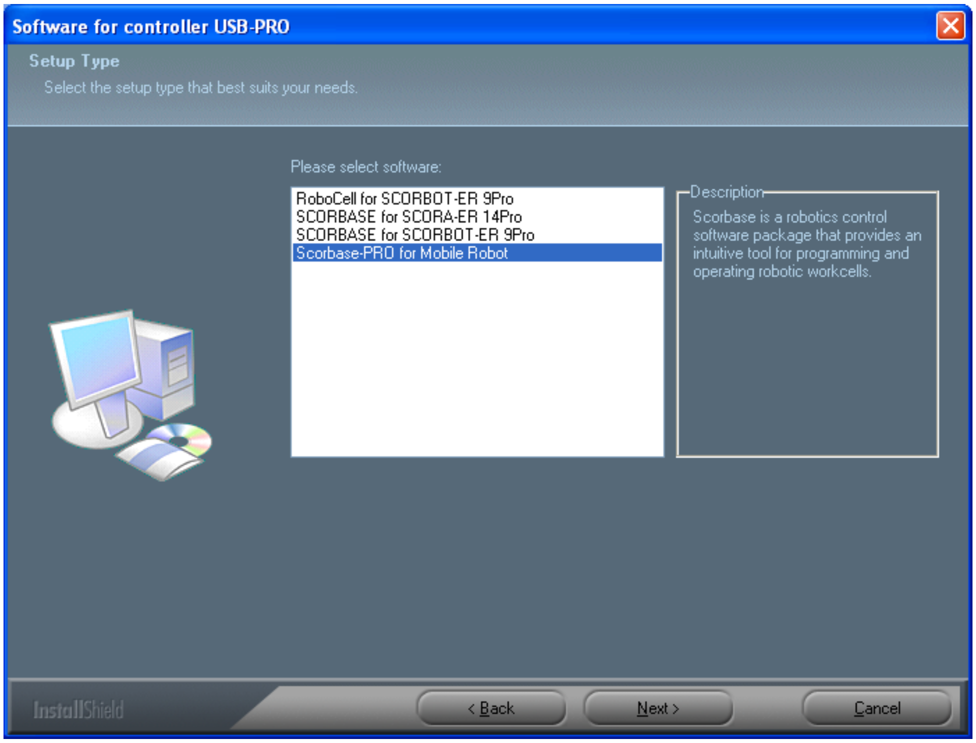
If the Device Manager displays a yellow exclamation mark, either you have not rebooted your machine or there is a problem with the port. If rebooting does not eliminate that mark, contact Intelitek technical support.

AGV Device Driver

This is also called the Mobile Robot Device Driver. Its purpose is to transmit commands to the AGV from either a human operator or a control program.

Installation of the AGV Device Driver is done automatically along with the installation of SCORBASE.

Insert the CD-ROM and follow the on-screen instructions. When prompted to select software to install, select **SCORBASE for Mobile Robot** or **SCORBASE-PRO for MOBILE Robot**.


Software for controller USB-PRO

Setup Type

Select the setup type that best suits your needs.

Please select software:

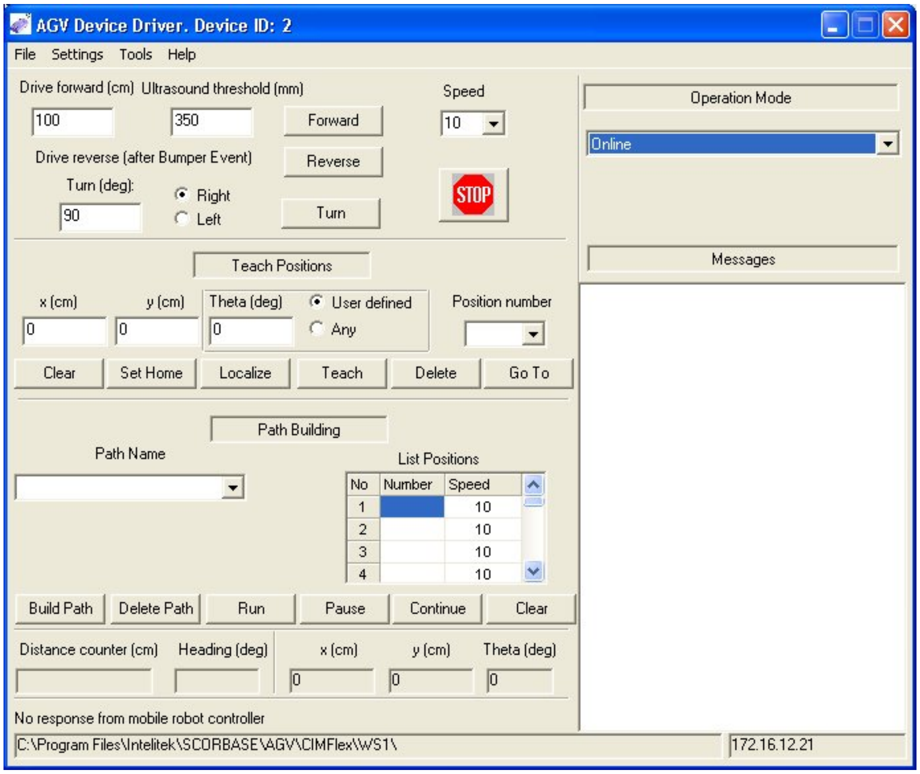
RoboCell for SCORBOT-ER 9Pro
SCORBASE for SCORA-ER 14Pro
SCORBASE for SCORBOT-ER 9Pro
Scorbase-PRO for Mobile Robot

Description

Scorbase is a robotics control software package that provides an intuitive tool for programming and operating robotic workcells.

By default, the installation program copies the file AGVDD.EXE into the AGV\BIN subdirectory of the Scorbase directory.

Open the AGV Device Driver. The following screen is displayed:



AGV Device Driver. Device ID: 2

Drive forward (cm): 100

Ultrasound threshold (mm): 350

Speed: 10

Drive reverse (after Bumper Event)

Turn (deg): 90

● Right ○ Left

Operation Mode: Online

Teach Positions

x (cm): 0 y (cm): 0 Theta (deg): 0

● User defined ○ Any

Position number:

Path Building

Path Name:

List Positions

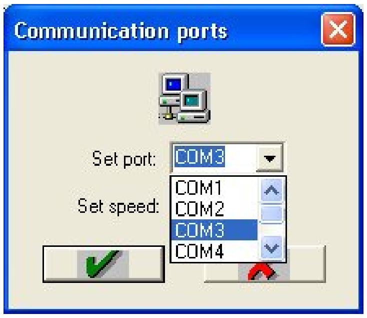
No	Number	Speed
1		10
2		10
3		10
4		10

Distance counter (cm): Heading (deg): x (cm): 0 y (cm): 0 Theta (deg): 0

No response from mobile robot controller

C:\Program Files\Intelitek\SCORBASE\AGV\CIMFlex\WS1\

172.16.12.21



Communication ports

Set port: COM3

Set speed:

From the menu bar, select Settings, and then the drop-down menu item Communications Port. In the subpanel labeled Set Port, select the COM port you identified in the previous step. Do not change the speed setting from the default (38,400).

This setting will be automatically saved for future use by the Device Driver each time it is loaded.

AGV Project Files

The AGV project files contain the definitions of the positions and paths available to the Mobile Robot.

The project files are not part of the installation disc, but are created by the human operator and saved by default into the subdirectory from which AGVDD.EXE is launched. These files will be assigned an *.APF extension. They are plain text files and can therefore be viewed and edited with a simple word processor such as Wordpad.

AGV Device Driver Messages file

The AGV Device Driver uses a special configuration file for the text of audio alerts played through the speakers of the PC which is running the Device Driver, and also status messages displayed at the bottom of the user interface window. This file is named AGVDD.ENG, and is located in the same directory that the AGVDD program is launched from. It is a plain-text file, which means it can be viewed and edited using WordPad.

The audio alerts are generated automatically from the text in the [voice_messages] section. However, you can substitute an actual sound file using the following procedure:

1. Create a \Voice subdirectory at the location where AGVDD.ENG is located (by default the AGV\BIN subdirectory of the Scorbase directory.)
2. Place the .WAV or .MP3 file in the \Voice subdirectory.
3. Insert the name of the sound file after the = sign following the applicable key whose name ends with **_file**

Example:

```
ControllerNotRespond_file = NotRespond.mp3
```

To change the text of any status message, first back up the file with a different filename. Then make the changes you need in the [status_messages] section and save the file.

Workstation software

SCORBASE for Mobile Robot

SCORBASE is a versatile control program which runs scripts to send messages to various robotic devices. The commands are sent to the controllers which actually drive the specific pieces of equipment. SCORBASE is fully featured, containing all program flow control and input/output communications commands. SCORBASE for Mobile Robot is a custom version, containing commands which are specifically designed to communicate with the AGV Device Driver.

SCORBASE is installed automatically together with the AGV Device Driver. See page 15.

AGV Test Files

Your installation disc also contains the AGV_TEST project to be run by SCORBASE. It consists of the following files:

AGV_TEST.WS

AGV_TEST.PNT

AGV_TEST.SBP

These files are copied by the installation program into the Projects subdirectory of the Scorbase directory, which is the default project location. They are used by SCORBASE as a system diagnostic tool for the on-board PC's control of the Mobile Robot. Be sure SCORBASE is in Pro mode before attempting to open this project.

CIM Software

CIM (Computer-Integrated Manufacturing) software is used to control a number of independent workstations from a single, central computer. It sends control messages to the SCORBASE software running on the workstation, and receives status messages which it can use in making program control decisions.

Contact Intelitek Sales for availability. See page 32 for more information.

3

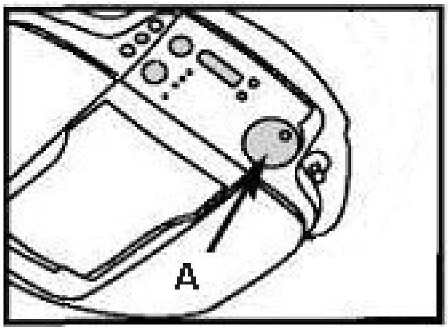
Operation

The ER-400 AGV Mobile Robot can be operated either manually or under the control of a computer. In either event, the Operation Mode Selector must be set to **ON** (position 1).

The unit may not be operated while it is being charged. Be sure to disengage the AC power supply before attempting to run the Mobile Robot.

Manual Mode

Turn the Operation Mode Selector to **ON** (position 1). Pull on the Pull-Line cord until the AGV either moves forward or turns. In this mode, the AGV can be towed to any unobstructed area on the work floor. The AGV design can negotiate past small variations in the floor, such as between a bare floor and a carpet. Loose electrical cords, however, can cause the unit to turn slightly, even though it may succeed in getting past them. This can interfere with tracking and navigation. Therefore, care must be taken to clear the floor of such items and secure them out of the way so that they cannot be dislodged.



The Operation Mode Selector is indicated by Arrow A in the illustration.

This model of the AGV is designed for use as a mobile robot. However, it still has some parts, circuitry and on-board programming which are needed only when it is used as an automated vacuum cleaner. Two of the settings on the Operation Mode Selector (Positions 2 and 3) fall into this category. On some models, these positions are labeled QUICK and MAX. Experimentation with these settings may lead to undesirable results. To avoid damaging circuitry or mechanical supports, users who wish to remove the superfluous vacuum cleaner parts should consult with Intelitek technical support.

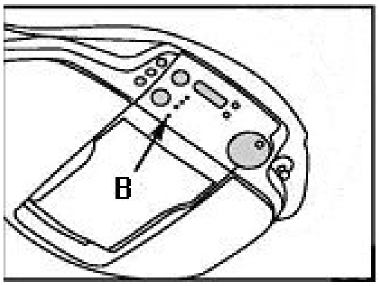
Emergency Stop

The red button marked **STOP** on the AGV's control panel can be used to manually interrupt the operation of the Mobile Robot, whether set in motion manually or by means of computer commands. When stopped with the red button, the unit emits a 2-tone beep.

Child Safety Lock

The unit is equipped with a Child Safety Lockout. To engage, set the Operation Mode Selector to OFF (position 0).

Press **[STOP] [UP] [UP]** on the ER-400 Control Panel. The Child Lock indicator light comes on and the unit beeps three times. See Lamp B, as indicated here:



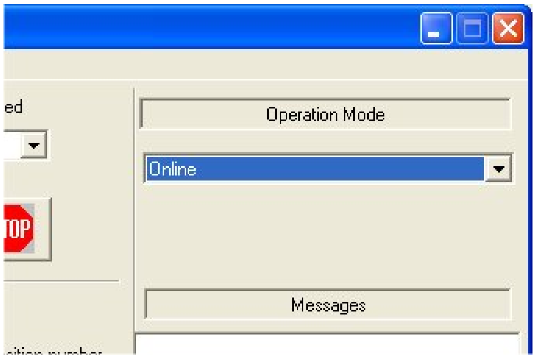
To disengage, ensure that the Operation Mode Selector is still set to **OFF** (position 0). Press **[STOP] [DOWN] [DOWN]**. The unit beeps three times and the Child Lock indicator light goes out.

On-Board PC Control

The ER-400 is designed to be controlled by computer. Ordinarily, the on-board laptop computer is used for this purpose. However, by connecting the USB cable directly to your desktop workstation, you can bypass the on-board PC for troubleshooting or setup. When issuing distance commands to the Mobile Robot, pay attention to the length of the cable.

To control the AGV with a computer, set the Operation Mode Selector to the first position, labeled **PC Control**. On some models, this position is labeled "SPOT". When set to PC Control, the LED display window reads "CL1". While a PC-controlled program is in progress, the display window changes to a moving wave pattern.

The Mobile Robot Device Driver is the software used by the on-board laptop computer. Its dialog box interface provides several methods of interacting with the AGV. Ensure that the operation mode is set to **Online**. The other settings, labeled “Simulation” and “Standalone,” are reserved for future use.

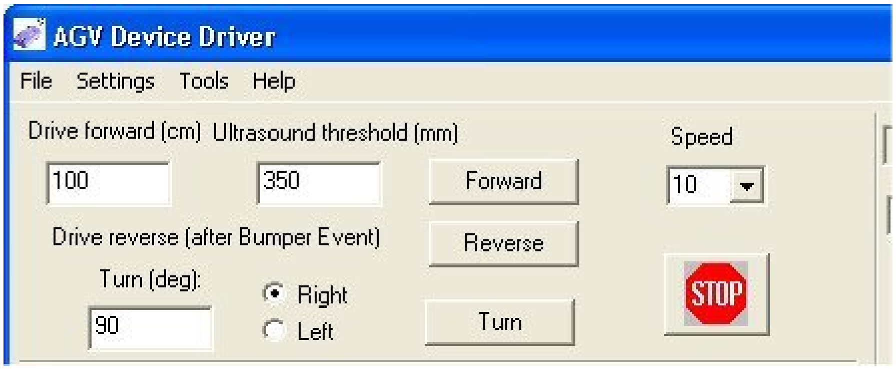


Operation Mode

Online

Individual Movements

By clicking on the appropriate buttons in the upper left subpanel, you can issue individual motion commands to move, turn or stop.



Drive forward (cm) 100

Ultrasound threshold (mm) 350

Speed 10

Drive reverse (after Bumper Event)

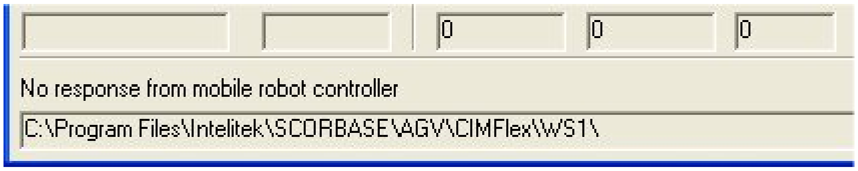
Turn (deg): 90

● Right
○ Left

Forward

Select the distance and the speed and click the button labeled **Forward**. You can also set the ultrasound threshold, which is the distance at which the Mobile Robot stops and sends an Ultrasound Event message to the AGV Device Driver on the laptop computer.

When the AGV detects an obstacle with its Bumper Guard or its ultrasound sensors, it stops, broadcasts an audio voice message and waits three seconds. An appropriate status message is displayed at the bottom of the screen. When the obstacle moves out of the way, the AGV resumes its assigned course. Without custom programming, it does not go around.



0 0 0

No response from mobile robot controller

C:\Program Files\Intelitek\SCORBASE\AGV\CIMFlex\WS1\

Turn

Select the number of degrees to turn, a positive integer up to 180. Select a direction (right or left) and a number between 1 and 10 for Speed. Click **Turn** to send the command to the Mobile Robot.

Stop

Click the **STOP** button to stop the AGV before it reaches the destination you have set for it. As an alternative, you can press the **[F5]** key on the PC which is running the AGV Device Driver.

Reverse

The **Reverse** button functions only after a Bumper Event has been detected. It causes the AGV to back up 10 cm on its path, without turning 180 degrees. The speed is pre-set at level 4. You can cause a Bumper Event by positioning a small, heavy object in the path of the Mobile Robot. If the object has the right dimensions, it will escape detection by the ultrasound sensors.

Homing

Place the AGV at the position you wish to use for its home base. You must have some means of physically defining this location. This is normally done by lining up marks on the floor with marks on the Mobile Robot, such as the adhesive arrows provided. This can also be done by making the Mobile Robot (or its Payload Shelf) physically touch a stationary object. However, you must keep in mind that the ultrasound sensors and Bump Guard will not allow the body of the robot to touch any objects, and that the Reverse command only works after a Bumper Event.

By definition, the (*x*,*y*) coordinates at the Home position are (0,0) and the theta is 0. Make sure the AGV has the proper orientation. If the home position is at a workstation, it is recommended that the *x* axis of the Mobile Robot should be the same line as the *x* axis of the stationary robot. This will simplify the procedures for localizing. Press the button marked **Set Home** to label this as the home position.

All other points are defined by the AGV Device Driver in relation to the Cartesian (*x*,*y*) coordinate system defined by the Home position. The Mobile Robot itself will keep track of the coordinates as long as it is not towed by the Pull-Line cord or physically moved with the handle on the payload stand.

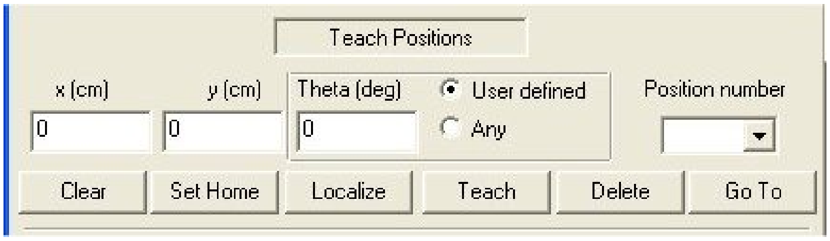
Each time the AGV is moved by hand, it must be either homed or localized before receiving commands which relate to defined points or paths in an AGV Position File. See page 25 for more information about localization and AGV Position Files.

Programmed paths

The AGV can be programmed to move along predetermined paths, each composed of one or more segments. Many different paths can be defined, each with its own unique name. You can then cause the AGV to move from point to point simply by selecting the name of that path.

Defining Points Manually

You must manually enter the coordinates of the points. Be mindful of the size of your workspace and the orientation of the Cartesian coordinates you have chosen. In the Teach Positions subpanel, enter positive or negative integers in the boxes labeled x (cm) and y (cm).


x (cm): 0
y (cm): 0
Theta (deg): 0
● User defined
○ Any

The theta setting is the orientation of the Mobile Robot when it reaches the destination point, not the heading it must follow to get there. Accordingly, it is only useful to designate a theta setting for the very last point on a particular path. Entering theta values for intermediate points may cause bizarre behavior, such as pausing in mid-journey and looking right and left before continuing.

If the point is an intermediate point on a path, select **Any** for the theta setting. If the point is the final point, specify the angular offset from the zero degree orientation of the Home Point, using the number of degrees to the left (counter-clockwise).

Assign a position number by typing an integer value in the box labeled Position number. You can use any number of digits up to six, but you should use the same number of digits for all position numbers in a single project. This will ensure proper sorting in the dialog boxes, while allowing you to develop your own numbering conventions.

When you are satisfied with all the values, press the button labeled **Teach**. This places the values in memory. To go to that point at a future time, select the position number and click the button labeled **Go To**. The Mobile Robot will go to that point in a straight line from its current location. The Speed drop-down box controls the speed.

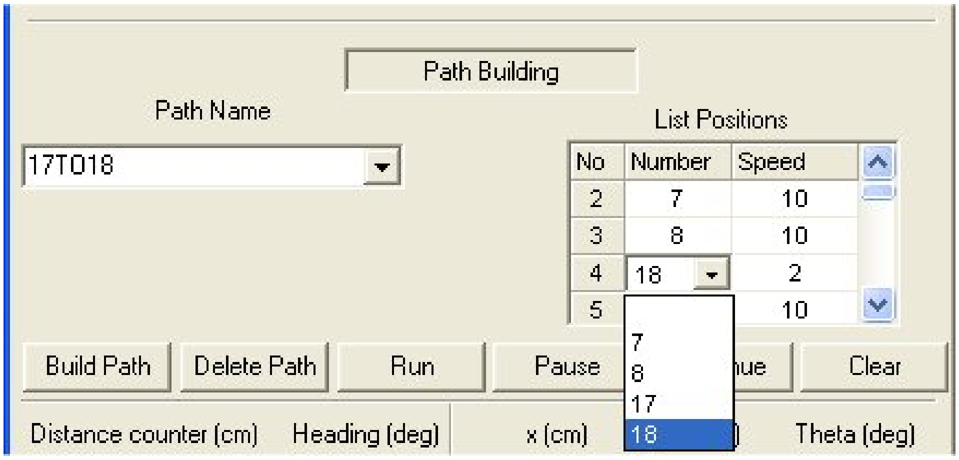
Without any special higher-level programming, the ER-400 will not be able to go around obstacles. For example, if you define four points in a square, but place an obstacle in the center, the robot will be unable to proceed to an opposite corner until the obstacle is moved out of its path.

If you make a mistake, you can simply **Teach** the correct values for the desired point, which overwrites the incorrect data. Alternatively, you can press the **Delete** button to remove the point from the list of points already defined.

Building Paths

A path is a set of one or more defined points, with speeds selected for the segments that connect them.

By definition, the first point on any path is the Mobile Robot's CURRENT location. Following the numbering scheme in the Path Building subpanel, the current location would be List Position number zero.

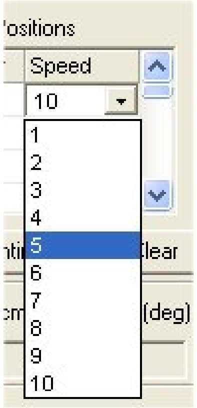


Path Name: 17TO18

No	Number	Speed
2	7	10
3	8	10
4	18	2
5		10

7
8
17
18

Select a descriptive name for the path and type it into the Path Name box. In the List Positions box, click in the cell in the column labeled Number which corresponds to the position you want to add. This causes the appearance of a down arrow (▾Click on it to See a drop-down menu with all of the positions you have defined with the Teach command. Select the desired position number. The number is then automatically copied into the cell. If you need to delete a point, select the blank from the drop-down menu for that position.



Speed: 10

1
2
3
4
5
6
7
8
9
10

You must also select a speed on the path segment TO the selected point. The default is 10, which is the fastest speed. To select a slower speed, click on the corresponding cell in the Speed column. A down arrow (▾will be displayed. Clicking on that will produce a drop-down menu of the available speeds between 1 and 10. Because the Mobile Robot generally has more precise position control at slower speeds, the last segment on a path should usually be slower than 10. By experimenting, you will find the best values for your application.

If you make a mistake at a position, you can select the correct point for that position and the mistake will be automatically replaced. To clear all the points listed for a path, click the button labeled **Clear**. It may be a good idea to delete any points that are not needed for any defined path.

When you are satisfied that all points are listed in the correct order, click the button labeled **Build Path**. This action saves the path in the laptop's memory but not on its hard disk. You can then cause the Mobile Robot to follow the entire path by selecting the path name in the Path Name drop-down box, and then clicking the button labeled **Run**.

As the Mobile Robot pursues the path, you can click on the buttons labeled **Pause** and **Continue**. **Pause** makes the AGV stop at the end of the current path segment, and **Continue** allows it to resume following the rest of the path. The bSTOPon causes an emergency abort. Clicking on the **Continue** button after the **Stop** button causes the AGV to resume its previously-assigned course.

Naming Paths

Ordinarily, the paths you define will be of two types: station-to-station and docking. A station-to-station path may have as many intermediate points as necessary, and is used to get the Mobile Robot as close as possible to its actual destination. A docking path basically consists of a single point, and is used only after the Mobile Robot has been localized.

Accordingly, a logical naming convention is to name the station-to-station paths with the numbers of the start and end points, for example, **21TO52**. Since the docking paths only have a destination point, a name with only the number of that point is best, for example, **DOCK52**. Remember that the first point on any path is actually its current location. Any navigational errors are adjusted by the localization procedure.

Saving AGV Position File

All of the data collected in positions and paths can be saved in a file on your laptop computer's hard disk. From the File menu, select Save or Save As, and choose a unique filename. By default, the resulting file is saved in the subdirectory the Device Driver was launched from. The Device Driver automatically appends an .APF file extension. The file is a plain text file, which can be viewed and edited with a simple word processor such as Wordpad.

Note that the Device Driver can only hold one file open at a time, though there is no limit to the number of points and paths that a single file can hold.

Localizing

As the Mobile Robot moves along its designated paths, minor navigation errors affect its precise position. These errors may result from humans, irregularities in the floor surface, changes in balance or center of gravity, stops caused by unexpected obstacles, or mechanical imprecision. The errors accumulate more quickly when the Mobile Robot is moving at higher speeds. The system must have a means of correcting these errors. This process is called **Localizing**.

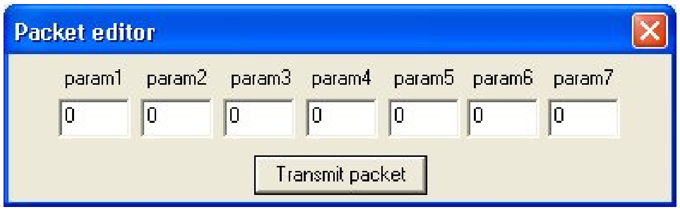
Instruct the Mobile Robot to move to a known point. When it has completed the movement, a close check will show that it has made a small error. Enter the AGV's actual coordinates and click on **Localize**. Then instruct the AGV to **Run** the docking path for that point.

Navigation errors can also be corrected by software solutions. A workstation running ViewFlex can process visual information from a camera to calculate the Mobile Robot's actual coordinates. Other effective solutions might utilize magnetic strips, lasers or ultrasound.

For more information on localizing, see page 31.

Packet Editor

The Packet Editor is for advanced users only. It is used to manually send a single command to the AGV. The command is a packet, or a short string of bytes, sent through the communications ports. Different commands require different byte-string lengths. The first byte is always 170, and the third is always 11. The second byte is the total number of bytes in the packet, and the last byte is a checksum.



param1	param2	param3	param4	param5	param6	param7
0	0	0	0	0	0	0

The Packet Editor performs several functions. It provides numeric fields for the programmer to manually insert DECIMAL values for all the bytes in the string after the first three and before the final checksum (CHK) byte. Thus, the first parameter (param1) in the Packet Editor is actually the fourth byte in the packet. The Packet Editor conveniently counts the total number of bytes and inserts the value into the second byte. Finally, it calculates the value required for the checksum, inserting it at the end of the packet. Note that if you calculate and enter your own checksum, the computer-generated checksum will be zero, which by itself would not affect the packet, but the count of bytes will be off by one.

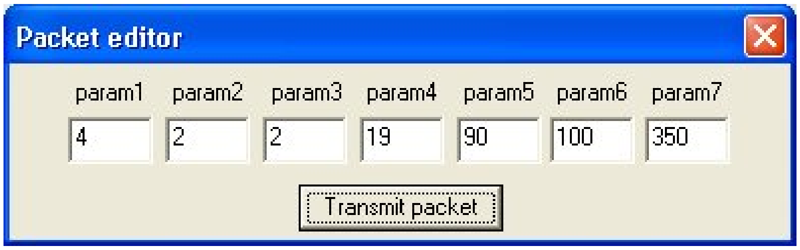
When you are satisfied that all the values entered are correct, press the button labeled **Transmit packet**.

The checksum byte is used to determine whether there have been any data transmission errors. Its value is calculated as follows: the numeric value of all the previous bytes in the packet are summed. If the value is greater than 255, such that it must be expressed in more than one byte, the higher order byte is discarded. The resulting value is subtracted from 255, and that value is the checksum. The following C-language code for this process is provided for your convenience.

```
byte Check_sum(byte* packet, byte length)
{
    byte i=0;
    long sum=0;
    for (i=0;i<length;i++)
            sum += packet[i];
    return((byte)(~sum));
}
```

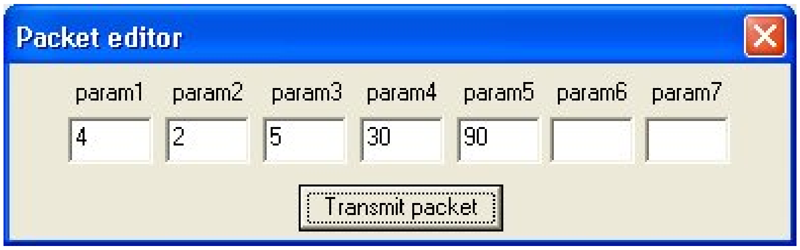
The following are examples of three commands, based on Appendix A.

Forward 100 cm at 90 degree heading, speed 19, with 350 mm ultrasound limit. Observe the path the Mobile robot takes. Note the differences between these parameters and the default settings in the AGV Device Driver.



param1	param2	param3	param4	param5	param6	param7
4	2	2	19	90	100	350

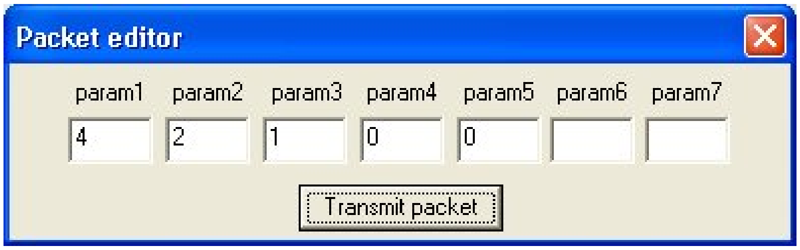
Turn 90 degrees to the right at speed 30:



param1	param2	param3	param4	param5	param6	param7
4	2	5	30	90		

If you change parameter 5 to zero, the instruction is Endless Motion. You can stop it with the following command:

Stop motor:



param1	param2	param3	param4	param5	param6	param7
4	2	1	0	0		

The packet editor allows you to fill in unused parameters with zeroes, with no change in function. Note: many of the commands used for the robotic vacuum cleaner are still operational.

Integration with Computer Workstation

The ER-400 can be configured to interact with a single computer workstation, which is typically a desktop computer. The control software is SCORBASE for Mobile Robot, which runs under Microsoft Windows. The Mobile Robot and its on-board laptop PC must be connected as provided at page 13, and must be running the AGV Device Driver as provided at page 15.

Possible applications may include delivery of a piece of raw material to a robotic lathe. In this case, the SCORBASE software will be running on the workstation which controls the Mobile Robot, the lathe and the robotic arm that manipulates parts at that location. The workstation script will issue one set of commands to the Mobile Robot, another set of commands to the robotic arm, and still other commands to the lathe.

Another possible application is packing a finished item into a container. The workstation running SCORBASE will issue commands to the Mobile Robot, the robotic arm and the machinery that delivers empty boxes and takes away full ones.

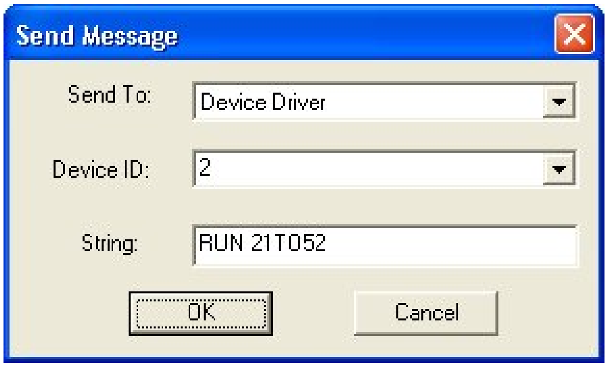
SCORBASE commands

There are only four commands sent by SCORBASE to the AGV Device Driver. This configuration frees the SCORBASE script from the low-level details of the paths and the conditions of the sensors. The commands are sent as ordinary text strings.

Run

This command is used in conjunction with the path names defined in the *.APF file which is currently open in the AGV Device Driver. Use of any unlisted names will generate an error message from the Mobile Robot, which is then displayed on the workstation in the Messages window of SCORBASE.

The following example demonstrates use of the SM dialog box in SCORBASE to send a RUN message to the AGV. The designated path is **21TO52**, the example given on page 25.



Send Message

Send To: Device Driver

Device ID: 2

String: RUN 21TO52

Stop

If the AGV happens to be within arm's reach, the user can press the red **STOP** button on the AGV's control panel or the **[F5]** key on the on-board laptop PC. In other situations, the user must interact through the workstation.

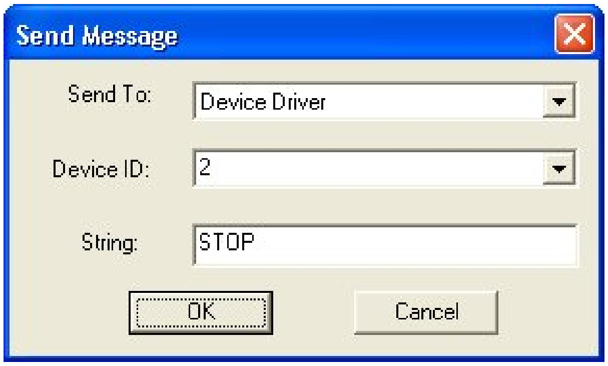
This command can be sent from the workstation to the Mobile Robot in one of two ways.

- The user may click on the i STOP the SCORBASE interface window.
- The user may press **[F9]** on the workstation computer keyboard.

However, neither action will have any effect if the user has not defined a subroutine called ON_STOP, which sends the appropriate message to the AGV Device Driver. That subroutine must be currently available for execution, which means that it must be part of every AGV SCORBASE project.

Note that this is a different key from the one used by the AGV Device Driver for the same function. If you are running both SCORBASE and the AGV Device Driver on the same computer, you can press **[F5]** to stop the AGV Device Driver, but the SCORBASE program continues to run. **[F9]** stops both.

The following example demonstrates use of the SM dialog box in SCORBASE to send a STOP message to the AGV.

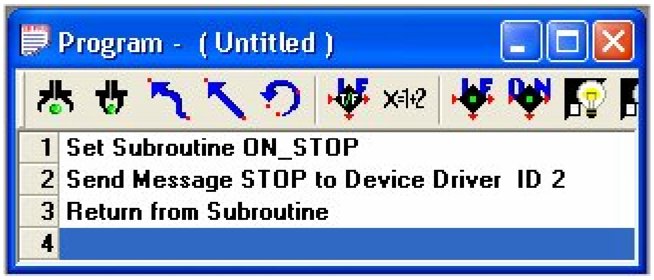


Send To: Device Driver

Device ID: 2

String: STOP

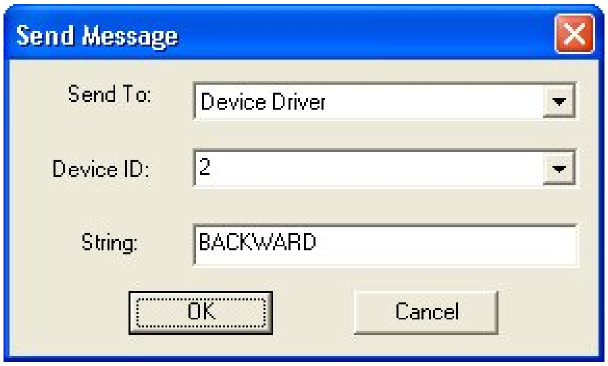
The following example shows how this command appears in the ON-STOP subroutine, as displayed in the Program Window of SCORBASE.



1 Set Subroutine ON_STOP
2 Send Message STOP to Device Driver ID 2
3 Return from Subroutine
4

Backward

The Backward command can be sent after a Bumper Event has been detected. It functions exactly like the **Reverse** button in the AGV Device Driver interface window. See page 22.



Send To: Device Driver

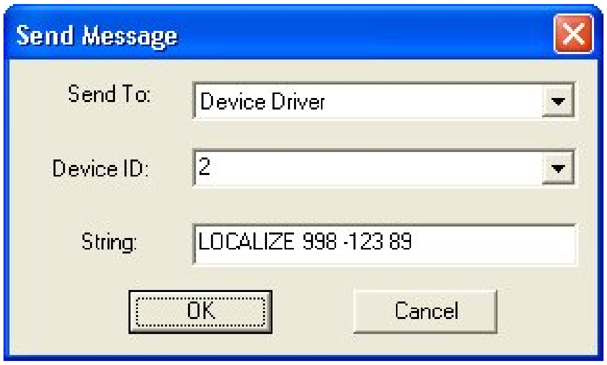
Device ID: 2

String: BACKWARD

Localize

SCORBASE uses this command to correct accumulated navigation errors. It requires three parameters: X, Y, and Theta. These parameters represent the ACTUAL location and orientation of the Mobile Robot in reference to the Cartesian coordinate system defined by the Home position. X and Y are given in millimeters (note the difference from the AGV Device Driver), and Theta is given in degrees. This data can be obtained by using a camera and ViewFlex software.

The following example demonstrates use of the SM dialog box in SCORBASE to send a message to the AGV. In plain English, the message means “AGV: your current location is x = 998 mm; y = -123 mm and you are turned at 89 degrees heading.”



Send Message

Send To: Device Driver

Device ID: 2

String: LOCALIZE 998 -123 89

Once the Mobile Robot has thus been instructed as to its actual location, it can be directed to move to its originally intended destination by means of running a docking path. See page 25.

The system designer needs to determine how often the Mobile Robot needs to be localized.

For more information on localizing, see page 26.

Mobile Robot messages

On receipt of a command from SCORBASE, the Mobile Robot responds with a transparent Command Acknowledged message. This does not show up in the Messages window. When the Mobile Robot has completed its assigned task, it sends a RUN_AGV_DONE message to the workstation. On receipt of this message, SCORBASE runs a user-defined subroutine such as AGV_DONE.

If the AGV encounters a problem which prevents it from completing the task, the AGV Device Driver sends two error messages to SCORBASE, one signaling that an error has occurred, and the other consisting of a code number which indicates the type of error. SCORBASE stores the error number in the variable AGV_ERRNUM and runs the subroutine named AGV_ERR. The following table provides the meaning of each of the error codes:

1	Bumper Event	6	Wheels Stuck Event
2	Stair Event	100	Robot Controller Not Responding
4	Ultrasound Event	101	Invalid Pathname

The following simple situations can cause the generation of Error 100 (Robot Controller Not Responding):

- Batteries for either the Mobile Robot or the on-board computer may be low, dead or not properly seated.
- The data cable between the Mobile Robot and the on-board computer may be dislodged.
- The AGV Device Driver may be not loaded properly, or not configured for the proper COM port.
- The RS-232/USB Device Driver may not be loaded properly.

Communications between the Mobile Robot and the computer workstation can also be disrupted if there is a problem with the LAN network. In this event, however, there will be no message from the Mobile Robot at all.

It is the responsibility of the SCORBASE script to resolve whatever problem is reported by the Mobile Robot.

Integration with CIM

Overview

A Computer-Integrated Manufacturing (CIM) facility typically consists of several workstations where work is performed by stationary machines and some means of transporting the work from station to station. The transport system may be a computer-controlled conveyor belt, a mobile robot such as the ER-400 AGV, a robotic arm on a slidebase, or some combination.

The layout of the work area must be carefully planned for the requirements of the manufacturing operation. Each stationary robot must have enough room to perform its tasks without interference from other robots or human operators. Each mobile robot must have adequate clearance to allow safe passage from station to station. The path must be wide enough to allow for cumulative errors in navigation and the passage of two-way traffic. The manufacturer recommends that a one-way path be one meter in width, level and free of all types of obstacles. There must also be one or more designated places where the system may perform a periodic localization operation to reset the mobile robot's coordinates. Finally, adequate provision must be made for charging and/or replacement of the batteries of each mobile robot and its on-board computer.

In a normal configuration, the CIM Manager has no direct communications with the Mobile Robot. This function is performed exclusively by one or more of the workstations. For testing or debugging purposes, the CIM Manager can be configured to send commands directly, without an intermediary.

The CIM administrator may design a configuration which places the AGV at the disposal of two or more workstations.

Network Setup

Please consult your systems administrator for assistance in setting up network communications. The CIM Manager computer should be named CIM-MANAGER. The workgroup should be named CIM. Each workstation should be named CIM-PCx, where x is a unique number assigned to each workstation.

If there is more than one AGV on a CIM network, each must have its own device ID number, assigned by a trained installation technician. The device ID numbers for CIM are distinct from the device ID numbers for other purposes, such as LAN identification.

CIM installations require a Wireless Access Point to integrate the AGV into the CIM network. Other types of networks may or may not need such a device. Please consult with Intelitek technical support for assistance in purchasing, installation and configuration.

4
Optional Accessories

Currently Available	Check for Availability
Spare Batteries (pair)	GPS
Stand-alone battery charger	Compass
3-Year Extended Warranty	Laser Rangefinder
Extended Project Support calls, in sets of 5 calls (after the first 5 complimentary calls). See page 37.	On-board robotic arm
ViewFlex Machine Vision system, including ViewFlex software, color PC camera, camera mount stand	Software interface for MatLab or LabView
ViewFlex Color Pan-Tilt Camera system(PAL or NTSC), including ViewFlex software, color Pan-Tilt camera, camera shelf bracket	Simulation, Navigation and Control software
CIM Integration package	Safety Lamp

5

Optional Research Projects

The following projects are suggested as ideas for students of high-school or college level computer programming or robotics courses. Ideas suggested for one project may be appropriate for other projects as well.

The source code for the AGV Device Driver is provided on your installation CD in the subdirectory named \Mobile Robot ER-400\Programming interface. To build an executable file, you must have Microsoft Visual C++ version 6.0 or newer.

The instructor should advise students carefully, considering the goals of the course, the scope of the work to be undertaken and the materials available. Some projects are relatively easy and others are quite difficult. The instructor should devise an appropriate scheme for determining whether the student has learned the required material.

Project Support

As part of the ER-400 AGV Mobile Robot package, Intelitek offers professional assistance to students in designing and implementing their projects.

Though armed with enthusiasm and creativity, students occasionally find themselves frustrated by technical problems. Intelitek feels that the best way to surmount such obstacles is with the assistance of professional engineers who work with the same issues on a daily basis. The engineer may:

- suggest one or more solutions, including solutions from third-party sources,
- advise the student to modify the project in one or more particulars,
- counsel the student to find a project of a more suitable level of difficulty.

Included with the AGV package is a virtual ‘card’ entitling the customer to five such calls, free of charge. Additional calls are available from Intelitek Sales. Issues which actually relate to AGV installation, maintenance, technical support or service will not be charged against the Project Support calls. Intelitek, in its sole discretion, retains the right to allocate a call to the proper category.

Contact Intelitek by e-mail at info@intelitek.com. Include the following words in the subject line: ER-400 AGV Student Project Support. Include a detailed description of the proposed project and the technical issue causing the problem. Intelitek will send a confirming e-mail to the purchaser of the unit. When the purchaser provides authorization, the call will be referred to the appropriate personnel for response.

Software-oriented projects

Add behaviors

Attempt to bypass obstacles

An obstacle may be a human, an object inadvertently left by a worker, or an object left by a prankster. The response of the robot to one of these situations is presently to wait until the obstacle is removed. However, there may be other solutions, such as attempting to bypass the obstacle by moving to one side of the path, by backtracking and selecting an alternate route. If either of these solutions is used, should the computer 'remember' the solution, or should it be resolved on every single occasion? Is there a way of bulldozing the obstacle out of the way? What sort of cry for help should be issued to human operators?

The following solutions to this problem are suggested for varying levels of difficulty.

Simple Solution #1

Change the default wait period from 3 seconds to a longer period after the second iteration. Once the obstacle has been removed, be sure to reset the delay to the original default value.

Simple Solution #2

Use SCORBASE to run a script which rewrites and saves part of the AGVDD.ENG file. This file can be edited on the fly, and the AGV Device Driver does not have to be restarted. Instead of broadcasting the same message repeatedly, the Mobile Robot should broadcast one message when it first encounters the obstacle, a second message if the obstacle is not removed within the default 3 seconds, and so on. The last response should be to notify the human operator. Be sure to replace the original AGVDD.ENG file for use the next time an obstacle is encountered.

Simple Solution #3

Rewrite the SCORBASE script so that upon encountering an obstacle, the Mobile Robot goes by a straight line to the beginning of its current path, and then follows an alternative route to get to the same destination.

Intermediate Solution #1

Rewrite the SCORBASE script so that upon encountering an obstacle, the Mobile Robot retraces its actual path back to the beginning, and then follows an alternative route to get to the same destination.

Intermediate Solution #2

Use SCORBASE to interact with Excel, keeping track of which paths are blocked, and directing the Mobile Robot to use only unblocked paths.

Advanced Solution #1

Use SCORBASE and Excel to map the entire workspace in the form of a suitably-sized grid. Use the data thus gathered to set up paths around the workspace. Consider the additional problems discussed at page 40.

Advanced Solution #2

Use SCORBASE and suitable path descriptions to allow the Mobile Robot to back up to the last unobstructed node point on the path, and then choose an alternate route to the destination.

Advanced Solution #3

Modify the source code provided for the device driver so that the mobile robot will map its environment, and use this mapping to avoid mapped obstacles. Create your own software or interface to any third-party mapping software.

Follow walls (count openings ...)

If the robot is being used to patrol hallways as part of a security operation, it might be used to follow walls instead of a prearranged program. It may bypass closed doors and investigate opened rooms. What provision should be made for pranksters trying to lure the robot into a trap? How can ultrasound sensors be used for an operation like this? Are there any drawbacks to using the Mobile Robot’s built-in functions for following walls?

Control motion velocity

Are there applications where you would want to change the velocity of the robot? If so, how would the velocity be modified, and what logic will be used to determine when modifications are needed? Consider not only linear velocity but also angular velocity on turns.

Implement circular and other non-linear trajectory (e.g., spline curves)

The Mobile Robot has a built-in function for curve trajectories. Are there advantages in using such functions to make turns instead of the sharp-angle turns involved in a point-to-point path definition? Are there limitations on when such a trajectory should be used, and what control logic should be used? Can you find a way to change the orientation of the Mobile Robot while it is still in motion, thus permitting travel along curves with changing radius?

Implement map creation and map-based navigation

How would you use the robot to collect and record data by patrolling a previously-unseen workspace? Are there limitations such as available lighting which would impact this data-collection process? Can you use an on-board video camera to collect visual data as well? How would you organize and display this data for maximum readability?

Remote control over the web

How would you interface the control software to direct the Mobile Robot from a remote location? Can you direct the Mobile Robot to automatically dial the telephone and issue an audio command? Under what circumstances would you want to use a video camera mounted on the Mobile Robot to transmit images over the web?

Interface with Matlab, LabView

Can you design an application which uses Matlab, LabView or ViewFlex to direct the motion of the Mobile Robot? Would your application be limited to periodic spot-checks, or could it be used to control the Mobile Robot over long paths? What applications could benefit from this sort of control?

Change in Center of Gravity

If the Mobile Robot takes on a new load, its center of gravity changes. It may be higher or lower than normal, and it may be more or less off-center. How will you modify the movement controls to deal with these situations?

Optimization of speeds

How would you determine the best speed for operation for various tasks, such as transport of objects of varying weights, and homing or localization? Can you write your program to adapt without human intervention?

Communications-oriented projects

Communication and control of multiple robots on one system

What steps are necessary to run more than one robot on a single system? If a second mobile robot is introduced to take half the workload, what changes must be made to the workstation control scripts?

Sensor polling and reporting

Write a control script to periodically query the Mobile Robot's sensors. Develop methods of receiving both analog and digital replies.

Command normalization

Write a routine to convert integer degree measurements to the values required by the on-board specifications, and to convert speeds of 1-10 to the numbers required for left and right turns.

Initialization

Write a routine to perform diagnostic checks at power-up, and to shut down unused parts that have no function for your application, such as the vacuum motor.

Monitoring of data packets

Write a program to view the data stream between the on-board PC and the Mobile Robot. This should show either decimal or hexadecimal bytes in the data stream, as well as what the data represents in human language.

Hardware-oriented projects

Add video camera

How will you resolve issues related to adding a video camera to the Mobile Robot? Where should it be mounted and why? How will you handle bumps and jiggles while the Mobile Robot is in motion? Can you mount an independent tilt/pan control so that the entire robot need not be turned? How will you implement control of the camera and reception of the data stream from it? Do you need a routine equivalent to localizing? Can you change various color and polarized filters?

Add new positional or navigational sensors

Can you add laser-guided sensors for navigation? Should these sensors be mounted on the robot or in the workspace? Should the control of these sensors be local, that is, handled by the robot, or should it be handled by the workstation running SCORBASE? Can you implement sensors to follow magnetic striping on the floor? Can you integrate other technologies for positioning, such as electrostatic sensors, optical sensors and zebra striping or solid-color striping?

Add other sensors

Can you integrate sensors for various environmental conditions, such as ambient light, magnetic interference, electric fields, infrared radiation at various frequencies, air temperature, humidity, smoke detector, or a sniffer for specific chemicals? How will you deal with the differences between analog and digital sensors? Can you design a sort of universal bus to accommodate numerous different types of sensors as might be required by different users of the robot?

Move existing sensors

The Mobile Robot presently has dozens of various sensors distributed throughout its body. Can you find other uses for any of them, and move them to a more suitable location? What changes in the polling and reporting routines are needed for communications between the Mobile Robot and the workstation? Can any of the sensors be modified for use by the on-board computer? How do you plan to deal with problems associated with disconnection of sensors from the existing system?

Experiment with travel surfaces and wheel materials

Is there a difference in the recording of positional data by the on-board odometers when the Mobile Robot moves on different surfaces, such as concrete or wood or carpet? Is there a way to modify the wheels to minimize these differences? Do you get different results with tread made of rubber or silicone, or plastic spikes from furniture cups or casters? Does the margin of error in the odometers change in a predictable way such that software programming can reliably compensate for it?

Experiment with trailer

The Mobile Robot’s specifications indicate a maximum payload of 1 kilogram. Can you devise a system to circumvent this limitation by towing a trailer? Where would you mount the trailer hitch? How will you deal with 90-degree turns? Is there an appreciable effect on the traction of the Mobile Robot’s wheels? Does the speed of the unit in travel affect the cumulative errors in navigation as recorded by the on-board odometers? Same question in relation to weight of the payload in the trailer? How will you deal with reversal of direction? Can you devise an AI routine to deal with making 3-point turns?

Add on-board robotic arm

Where would you mount a robotic arm for maximum effectiveness of its work envelope? How will you deal with changes in the center of gravity of the Mobile Robot? Can you modify communications on the existing robot to transmit messages to and from the robotic arm? Can you implement a separate infrared port to treat the robotic arm as a separate network device? What uses do you envision for this add-on? Where would you store a tool kit for use by this robot arm, and what tools would you put in it? What sort of power supply and controller should be used, and where should they be mounted?

Add laser bar-code reader

Can you integrate a laser bar-code reader and mount it on the Mobile Robot? Where would you locate it? Would you mount it on a robotic arm or make it part of a toolkit available for the robotic arm?

AI-oriented projects

Interaction with human

Can you program the robot to learn the different habits of each human it encounters, such as where in the building he or she is likely to be found at any given hour? Can you program it to deliver beverages according to the taste preferences of each person, and at the times each person usually responds “yes” to queries? Can you program it to interface with humans via an office telephone system or by face-to-face voice commands? Can you program it to ask questions by means of sound files, and collect the answers by means of a hand-held infrared device such as a television remote control? Can you program it to send calls for help to cell phones, and accept input from the telephone keypad?

Multi-agent group robotics

Can you transmit data collected by one robot to another, then use that data to make decisions? If one robot detects an intruder, how will you optimize the response of the other robots to go from their current location to their emergency station? If one robot drops out of communication with the network, what responses will you develop for the remaining robots? Can you develop diagnostic routines which cause multiple robots to play Follow The Leader, with varying degrees of spacing, different speeds and different routes? Develop criteria for determining whether it is better to follow the precise same path as the leader or to follow a line-of-sight path. Can you devise a routine for the leader to evade the robot following? Can you devise routines to deal with temporary loss of communication with the leader, or temporary disappearance from sight?

Data collection and optimization

Can you use different types of sensors to collect different types of data, then use the data to develop and optimize a response? Develop AI routines to determine whether changes in ambient light are better than ultrasound or infrared detectors at locating intruders. Compare this data to data collected from other types of sensors.

Perform task in natural environment (deliver, assist)

Can you devise a method for the robot to perform a task with uncertain outcome, and determine whether the task has succeeded? Can you give the robot some routines for determining an alternate course of action? If one tool fails to achieve the desired result, how will the robot determine which other tool to try? If one tool could ruin the material, under what circumstances would you want to try it anyway? Under what circumstances would you want the robot to refuse to begin the operation? If the robot is given a mostly repetitive task, what routines would you use to determine whether prior successful strategies should be tried again?

Identifying bottlenecks

What routines could you develop to identify bottlenecks in a production line? If one of the bottlenecks is a mobile robot, how can the problem be solved? What trade-offs are there in terms of robot down time and recharging of batteries? How is your analysis affected by batteries working at marginal power levels? How would you use AI routines to develop alternate routes and sequences?

Environmental changes

If your robot application is designed to navigate or collect data by means of a video camera or other optical sensor, how will your application adapt to changes in the ambient light? Such changes could come from nightfall, failure of one or more light bulbs, general power failure, malicious mischief, or changes in color of wall paint.

Path resumption

A security robot could be deliberately moved off of its path by an intruder. What routines would you devise for it to return to its path? Could these routines be generalized to accommodate any type of navigation system such as magnetic striping, laser-guidance or optical landmark recognition? How would you determine which navigation system is most successful at recovery? How will you deal with the situation where a robot is simply “turtled” (turned on its side)?

Sensor-oriented projects

Use on-board camera or other sensors for obstacle avoidance

Use a variety of sensors in conjunction with an on-board camera to determine the most effective way to detect different types of objects. Use AI routines to test alternate methods of avoiding such objects and determine which is most efficient.

Use on-board camera for landmark navigation

Use image processing routines to identify unique landmarks and note their locations on a workspace map. Create routines to identify the objects from different angles and distances. Use on-board camera to determine when an obstacle has been encountered.

Implement line-following application (with existing or new sensors)

The Mobile Robot has sensors which permit it to follow a wall or contour. Develop a routine to map routes along walls, allowing for the fact that doors may be open or closed. Adapt the stair sensors to allow the robot to follow some type of guide marking along the floor. Use an on-board camera to follow a painted stripe or cracks in a tiled floor.

Implement precise homing or docking

Using existing or new sensors, develop a routine to place the robot at a precise point, with a precise orientation. Do not use brute force. Or use brute force. Can you develop an AI routine to keep track of the adjustments required, and make more efficient adjustments? For example, if the robot always ends up at its HOME station 10 cm to the north, your first adjustment may be to simply turn, move 10 cm and turn to the right orientation. If that repeats enough times, however, it may be more efficient to have the robot turn slightly on the last leg of its approach.

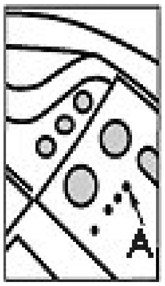
6

Maintenance

Daily Maintenance

Battery Charging

Manually plug the AC Power Supply into the battery charger port, located in the bottom center of the front of the unit. While the charger is working, the Battery Light (indicated by Arrow A) flashes green.



The normal time to fully charge the batteries is 8 to 12 hours, which typically means overnight.

If the unit experiences heavy use, the batteries may have to be charged more often. You may also order a spare set of batteries and charge them in the stand-alone charger (see page 11) while the Mobile Robot is in use.

Start-Up Checklist

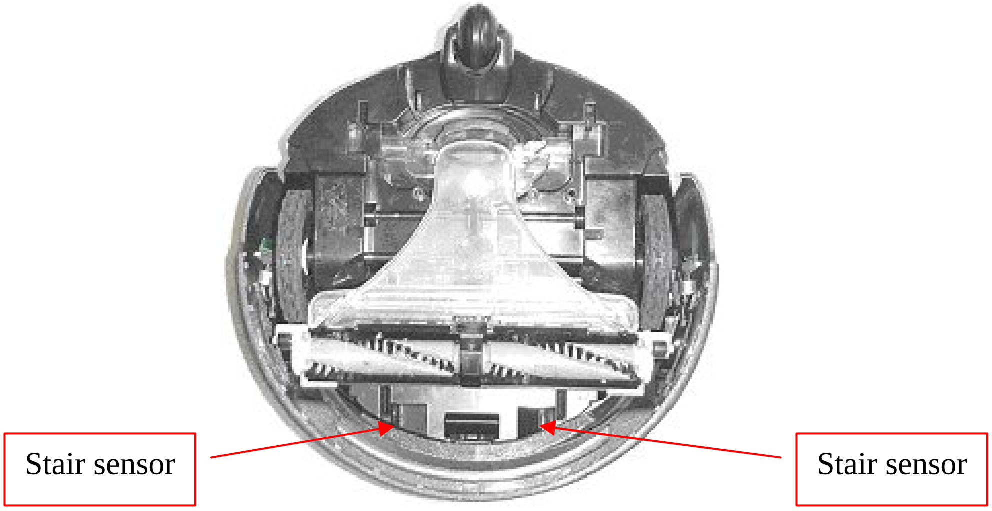
1. Remove the charger cables from the front of the AGV Mobile Robot and from the on-board laptop computer.
2. Move the Mobile Robot to its Home position, if that is different from the charging station. Refer to Manual Operation procedures detailed on page 19.
3. Set the Operation Mode Selector to the first position, labeled **PC Control**.
4. Boot up the on-board laptop computer and the desktop workstation.
5. Launch SCORBASE on the workstation. Launch the AGV Device Driver on the on-board computer. Load the AGV Position File named **3POINTS**. Run the **AGV_TEST** file from SCORBASE to verify that the Mobile Robot is in proper working order.

6. In the AGV Device Driver, load the AGV Position File you need to use, and in SCORBASE, load the script file you need.
7. If you wish to conserve battery power in the on-board laptop PC, turn off the monitor screen.

Periodic Inspection

Stair Sensors

If the stair sensors have been enabled, accumulated dirt may interfere with AGV operation. The two sensors are located on the bottom of the ER-400. Wipe with a clean, soft cloth.



Wheels

Check the wheels for signs of undue wear, particularly signs of flat spots. Clean the contact surface of dirt and any foreign objects that may have gotten lodged in the treads. Turn the AGV gently onto its back and inspect for any hair, fiber or other material that may interfere with the axles.

Bumper Guard

Place a small object weighing at least 100 grams in front of the AGV and tow the AGV with the Pull-Line cord until it makes contact. If the Bumper Guard fails to stop the AGV, check to see that it can move freely in and out. If this does not resolve the problem, call your service center.

Noises

Follow the unit as it pursues one of its assigned paths and listen for any unusual noises.

Battery Storage

If you plan to leave the ER-400 AGV out of operation for a few days, charge the batteries fully and station the AGV in a secure location. When the batteries are fully charged, remove the charger cable and disconnect it from the wall AC power supply.

If you plan to leave the ER-400 out of operation for more than two weeks, charge the batteries fully and remove them, storing them separately from the unit.

Keep a log showing battery usage, and be sure to order replacement batteries well in advance of when they will be needed.

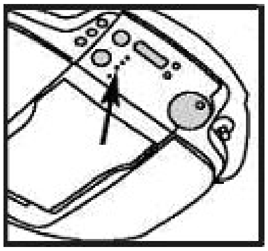
Dispose of used batteries in accordance with applicable laws. If available, recycle the batteries by returning them to your supplier. Batteries are hazardous waste and must not be disposed of in ordinary trash collection.

Troubleshooting

Important: Do not attempt to make repairs or perform procedures for which you are not qualified. If you have any doubt about how to proceed, call Intelitek technical support.

Display Message	What It Means	What You Should Do
Call Help (01-15)	Robot requires service by an authorized technician.	Remove batteries for 3 seconds, then replace. This forces a system reset. If this does not resolve the problem, note the number of the error message and call your dealer.
E1	The battery voltage has reached a critically low level.	Completely charge the batteries immediately to avoid permanent damage.
E2 (right after pressing START)	You are trying to start the robot too close to a stair or door.	Start the robot in a clear location.
E2 (during operation)	The robot is trapped and can't find its way out. Drive wheels are jammed.	Remove the robot from its trapped location. Clear drive wheels.

E3	The Bumper Guard is caught or snagged on something, or robot is sitting on an uneven surface.	Reposition robot and make sure the Bumper Guard is free.
E5	The Pull-Line cord is not fully retracted.	Make sure the Pull-Line cord is fully retracted before pressing **START**.
E7	Charging problem. The charger is plugged into the port, but no voltage is going to the batteries.	Make sure the charger is plugged into the wall; make sure the fuse or circuit breaker is set for the wall plug. Try a new charger. If the problem persists, call your dealer.

Problem	**Possible Cause**	**Solution**
Robot is non-responsive and nothing displays in the message window; there is a flashing red light on the Control Panel. 	Robot has entered Sleep Mode (Stand-By) after a period of 60 seconds of inactivity.	Press any button on the Control Panel, or pull the Pull-Line cord. Revise workstation control program to send a Dummy command periodically.
Short battery run time after a full charging session.	Battery/batteries are damaged.	Replace both batteries. Contact Service Center.
Robot not waking up.	Battery fuse is missing or burnt; batteries are weak or dead.	Contact Service Center.
Robot avoids dark areas or threshold.	Malfunctioning stair sensor.	Clean or disable stair sensor.
Robot rotates and changes direction in the middle of the room, for no reason.	Directional system false alarm.	Clean the bumper screen.

Pull-Line cord does not retract.	Faulty Pull-Line cord.	Contact Service Center.
Pull-Line cord does not pull out.	Pull-Line cord has been disabled during startup.	Retract the Pull-Line cord, remove and re-install the batteries.
Pull-Line cord does not pull out.	Faulty Pull-Line cord.	Contact Service Center.

Repacking for Shipment

There are several occasions that may require the Mobile Robot to be shipped. The user may be moving to a new location, transporting the Mobile Unit for display or demonstration in a remote location, or returning it to the manufacturer or dealer for service. If the unit is to be stored for long periods of time, the manufacturer highly recommends packing it as if it is to be shipped.

The best way to pack the unit for shipment is in its original box, with its original packing materials. For this reason, you are advised to keep those materials.

Unless instructed otherwise by your supplier, you must pack all parts together, including the AC power supply, batteries and cables. Each unit must be wrapped in heavy plastic and placed so that movement in shipment will not cause them to rub together or contact one another. The unit must be surrounded by a layer of packing material at least 7 cm thick to protect it from damage in shipment. The packing material may be plastic peanuts, bubble pack or other commercially suitable material. Do not use crumpled newspapers.

The shipping carton must be at least 15 cm longer than the parts in all three dimensions. The dimensions of the original carton are 610mm (w) x 490mm (d) x 600mm (h). The carton must be securely fastened with packing tape, not cellophane tape or masking tape. Contact your supplier if you have any questions.

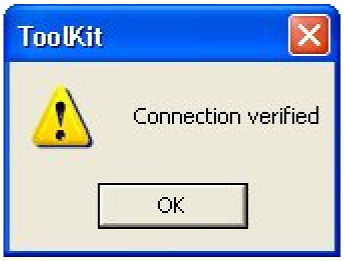
Advanced Troubleshooting

The ER-400 AGV Mobile Robot is shipped with software tools for the advanced user. This software is used for diagnosing errors and also for fine-tuning user projects.

Mobile Robot Toolkit

The Mobile Robot Toolkit is for advanced users only. It is not installed on your system, but may be run from the CD-ROM or a suitable directory selected by the user. The filename is TOOLKIT.EXE. The toolkit requires that the designated COM port be free for its exclusive use. If you have the AGV Device Driver loaded, you must close it before opening the toolkit. The AGV Mode Selector may be set in Position 0 or Position 1.

The toolkit contains four separate utilities: Diagnostics, Settings, COM Port and Upgrade. When first loaded, it displays a control panel with buttons to select those four options. To return to the main control panel, click on the button labeled **Menu**.


ToolKit

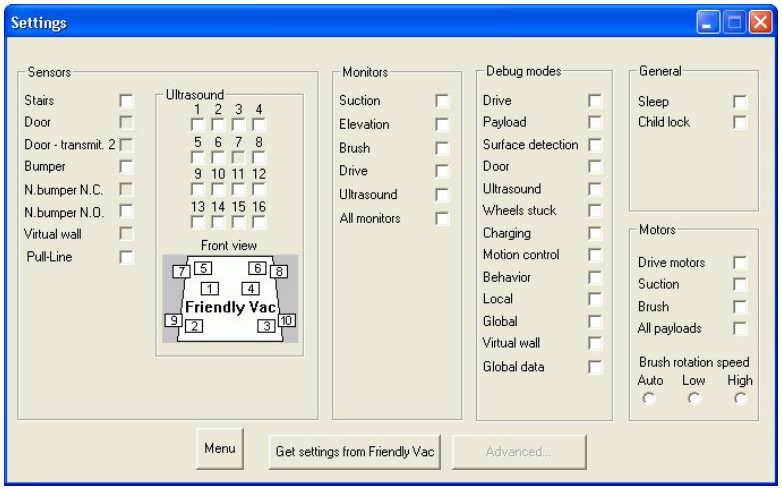
Connection verified

The COM Port utility provides an alternate means of assigning the communications port to the Mobile Robot, similar to the procedure described on page 16. Press **Check Connection** to ensure that you have communications between the PC and the AGV. If successful, the software will report Connection Verified.

In order to establish a good connection, you may have to run the AGV Device Driver, issue a single command to the AGV, and exit. Re-load the toolkit, select **Configure COM Port**, press the button marked **Set COM Port** and then **Check Connection**.

The Diagnostics and Upgrade utilities are needed only for purposes of repair. If you need repair, please contact your Intelitek service center.

The internal microprocessor in the Mobile Robot runs its own software. When the Mobile Robot is turned on, the processor boots up that software, including many predetermined default settings. The **Settings** utility is used to modify those defaults according to the needs of the user. For example, the user may need to disable the stair sensor because of the texture or coloration of the floor surface in the workspace where the ER-400 will be used. A check mark (✓) indicates that a sensor is active.


Settings

Sensors
Stairs ☐
Door ☐
Door - transmit. 2 ☐
Bumper ☐
N.bumper N.C. ☐
N.bumper N.O. ☐
Virtual wall ☐
Pull-Line ☐

Ultrasound
1 ☐ 2 ☐ 3 ☐ 4 ☐
5 ☐ 6 ☐ 7 ☐ 8 ☐
9 ☐ 10 ☐ 11 ☐ 12 ☐
13 ☐ 14 ☐ 15 ☐ 16 ☐

Monitors
Suction ☐
Elevation ☐
Brush ☐
Drive ☐
Ultrasound ☐
All monitors ☐

Debug modes
Drive ☐
Payload ☐
Surface detection ☐
Door ☐
Ultrasound ☐
Wheels stuck ☐
Charging ☐
Motion control ☐
Behavior ☐
Local ☐
Global ☐
Virtual wall ☐
Global data ☐

General
Sleep ☐
Child lock ☐

Motors
Drive motors ☐
Suction ☐
Brush ☐
All payloads ☐

Brush rotation speed
Auto ○ Low ○ High ○

Appendices

Appendix A
Research Kit - Control Functions

* See page 26 for details

Function	Param1	Param2	Param3	Param4	Param5	Param6	Param7	Param8	Notes
*	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8-9	Byte 9-11	Byte 10-13	Byte 11-14	
Navigation – left	2	1	CHK						CHK = 65
Navigation – right	2	2	CHK						CHK = 64
Navigation – forward	2	4	CHK						CHK = 62
Navigation - reverse	2	8	CHK						CHK = 58
Vacuum motor	4	0	0-98	CHK					Param3: speed; 0 = stop
Vacuum brush	4	1	0-98	0-1	CHK				Param3: speed; 0 = stop Param4: 0 = forward, 1 = reverse
Drive motor – stop	4	2	1	0	0	CHK			CHK=59
Forward	4	2	2	0-19	0-359	0-65535	1-1000	CHK	Param4: speed Param5: heading in degrees (2 bytes) Param6: distance in cm (2 bytes); 0 = endless Param7: ultrasound threshold in mm; 0 = ignore
Reverse 10 cm	4	2	3	0	0	CHK			CHK = 56
Turn to	4	2	4	0	0-359	CHK			Param5: heading in degrees (2 bytes)
Turn	4	2	5	15-120	0-180	CHK			Param4: 15-60 = velocity right 75-120 = velocity left Param5: degrees to turn; 0 = continuous
Follow wall contour	4	2	6	0	0	CHK			CHK = 53
Homing	4	2	7	0	0	CHK			CHK = 52
Motors PWM	4	2	8	0-100	0-100	CHK			Param4: Left motor PWM Param5: Right motor PWM
Motors direction	4	2	9	0-1	0-1	CHK			Param4: Left motor direction; 0 = forward; 1 = reverse Param5: Right motor direction
Curve	4	2	10	0-120		CHK			Param4: 0-60 = velocity right; 61-120 = velocity left Param5: radius (cm)
Buzzer	6	1-29	1-10	CHK					Param2: number of buzzes Param3: frequency
Dummy	7	CHK							CHK = 62 Used to avoid communications timeout
End program mode	8	CHK							CHK = 61
Communications timeout	9	0-255	CHK						Param2: time in seconds; 0 = no timeout
Wait	10	0-255	CHK						Param2: time in units of 50 ms (1/20 sec)

Appendix B
Telemetry Functions

Digital Sensor Polling Functions

Function	**Param1**	**Param2**	**Param3**	**Param4**	**Param5**	**Param6**	**Param7**	**Param8**	**Notes**
	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10		
Odometer	3	*	2	0	0	0	CHK		* Param2: message rate (50 ms increments)
Left nozzle bumper	3	*	16	0	0	0	CHK		
Right nozzle bumper	3	*	32	0	0	0	CHK		
Stair sensor	3	*	0	0	2	0	CHK		2 infrared sensors
Control Panel Buttons	3	*	0	0	4	0	CHK		
Leash (Pull-cord)	3	*	0	0	8	0	CHK		
Door	3	*	0	0	16	0	CHK		2 infrared sensors
Infrared receivers	3	*	0	0	32	0	CHK		5 sensors for remote control
Charging Plug	3	*	0	0	64	0	CHK		

Analog Sensor Polling Functions

Function	Param1	Param2	Param3	Param4	Param5	Param6	Param7	Param8	Notes
	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10		
Ultrasound sensors	3	*	1	0	0	0	CHK		* Param2: message rate (50 ms increments)
Left bumper	3	*	4	0	0	0	CHK		
Right bumper	3	*	8	0	0	0	CHK		
Nozzle height	3	*	64	0	0	0	CHK		
Charging voltage	3	*	128	0	0	0	CHK		
Vcc voltage	3	*	0	1	0	0	CHK		
Ambient temperature	3	*	0	2	0	0	CHK		
Brush elevation motor current	3	*	0	4	0	0	CHK		
Right motor current	3	*	0	8	0	0	CHK		
Left motor current	3	*	0	16	0	0	CHK		
Selector position	3	*	0	32	0	0	CHK		
Front stair PSD	3	*	0	64	0	0	CHK		
Humidity	3	*	0	128	0	0	CHK		
Battery Voltage	3	*	0	0	1	0	CHK		
Charging temperature	3	*	0	0	128	0	CHK		
Left battery thermistor	3	*	0	0	0	1	CHK		
Right battery thermistor	3	*	0	0	0	2	CHK		

Appendix C
Robot Messages to PC

Stop notification is sent when any of the following occurs:

- STOP button in the IR remote was pressed
- STOP button in the robot's control panel was pressed
- Stair or Bumper events were detected during navigation commands

Function	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Notes
Stop	170	5	11	1	CHK	CHK = 68

Acknowledgements

Function	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Notes
Drive command received	170	7	11	5	1	0	CHK	CHK = 68
Drive command completed	170	7	11	5	2	0	CHK	CHK = 67
	170	7	11	5		*	CHK	Byte 6: 1 = bumper event 2 = stair event 3 = desired distance reached 4 = ultrasound event 5 = desired heading reached 6 = wheels stuck event
Global command received	170	7	11	5	3	0	CHK	CHK = 66
Global command completed	170	7	11	5	4	0	CHK	CHK = 65

Telemetry responses (digital)

Function	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Bytes 10-n	Byte n+1	Notes
Odometer	170	34	11	3	*	2	0	0	0	10-11: heading 0-360 12-15: cumulative distance (cm) 16-19: L odometer fwd ticks 20-23: R odometer fwd ticks 24-27: L odometer rev ticks 28-31: R odometer rev ticks 32: L motor direction; 0=fwd,1=rev 33: R motor direction; 0=fwd,1=rev	CHK	
Left nozzle bumper	170	11	11	3	*	16	0	0	0		CHK	Byte 10: 1 = pressed
Right nozzle bumper	170	11	11	3	*	32	0	0	0		CHK	Byte 10: 1 = pressed

Stair sensor	170	11	11	3	*	0	0	2	0		CHK	Byte 10: 1 = left stair detected 2 = right stair detected
Control Panel Buttons	170	11	11	3	*	0	0	4	0		CHK	Byte 10: 1 = Go 2 = Stop 3 = Up 4 = Down
Leash (Pull-cord)	170	11	11	3	*	0	0	8	0		CHK	Byte 10: 1 = in/out 2 = Left 3 = Right 4 = Forward
Door	170	11	11	3	*	0	0	16	0		CHK	Byte 10: 1 = vertical 2 = inclined
Infrared receivers	170	16	11	3	*	0	0	32	0	Byte 10: Right Front Byte 11: Left Front Byte 12: Right Byte 13: Left Byte 14: Rear	CHK	Values for bytes 10-14: 1 = Stop 10 = Unlock 11 = Left 12 = Right 13 = Reverse 14 = Forward 15 = Call me 99 = no signal/illegal transmission
Charging Plug	170	11	11	3	*	0	0	64	0		CHK	Byte 10: 1 = detected

* In this table, Byte 2 is the number of bytes sent in the packet, and Byte 5 is the message transmission rate (50 ms increments)