



# IntelliBrain™

## Robotics Controller

# User Guide

[www.ridgesoft.com](http://www.ridgesoft.com)

## **IMPORTANT NOTICE**

Information in this document is furnished under license and may only be used in accordance with the terms of the license.

RIDGESOFT PRODUCTS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

RIDGESOFT PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS WHERE FAILURE COULD RESULT IN THE LOSS OF LIFE OR HAVE SERIOUS LIFE THREATENING OR ECONOMIC IMPACT.

### **Regulatory Warning**

IntelliBrain robotics controllers, their constituent components, add-on boards and accessories sold by RidgeSoft, LLC are not FCC approved because they are not in finished product form. If you wish to obtain FCC approval, you must first design the controller and/or accessory into a product, and then seek FCC approval of the whole product.

### **Trademarks**

RidgeSoft™, RoboJDE™ and IntelliBrain™ are trademarks of RidgeSoft, LLC.

Java™ and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. All other brand or product names are trademarks of their respective owners.

### **Copyright**

Copyright © 2004, 2005 by RidgeSoft, LLC. All rights reserved.

RidgeSoft, LLC  
PO Box 482  
Pleasanton, CA 94566  
[www.ridgesoft.com](http://www.ridgesoft.com)

Document Revision 1.2

# Table of Contents

<b>Introduction</b> .....	<b>3</b>
Main Board Features .....	4
Expansion Board Features .....	4
<b>Getting Started</b> .....	<b>5</b>
Connecting Power .....	5
Installing and Removing the Expansion Board .....	6
Installing the Expansion Board.....	6
Removing the Expansion Board.....	7
Establishing a Host Connection.....	7
Changing the Host Port Baud Rate.....	8
Programming IntelliBrain .....	8
Application Programming Interface Documentation .....	8
Using RoboJDE .....	9
Error Messages.....	9
Mounting IntelliBrain on a Robot Chassis .....	9
IntelliBrain-Bot Chassis Mounting .....	10
Lego Mounting .....	10
Mark III Mini-Sumo Mounting .....	10
Custom Mounting.....	10
Connecting Sensors and Effectors .....	10
Connecting Handy Board Sensors.....	12
<b>Main Board</b> .....	<b>14</b>
CPU and Memory .....	14
Flash Memory .....	14
RAM.....	14
CPU EEPROM.....	15
Direct Memory Access .....	15
Main Board Power Connections .....	15
Choosing a Battery and/or a Power Supply .....	16
Main Power Supply Loading Considerations.....	16
Battery Indicator.....	17
Host Port (COM1).....	17
LCD Display.....	18
Main Board LEDs .....	18
Buzzer .....	18
Push Buttons .....	19
Thumbwheel.....	19
Main Board Servo Ports.....	20
Serial / CMUcam Port (COM2) .....	20
Main Board Analog / Digital Input Ports.....	22

Analog Ports 1 - 2 .....	22
Analog Port 3 .....	22
Analog Ports 4 - 7 .....	22
Main Board Digital Input / Output Ports .....	23
I <sup>2</sup> C Ports .....	24
Infrared Receiver .....	25
Bootstrap Loader .....	25
Reverting to Factory Settings .....	26
Viewing the Serial Number .....	26
Expansion Board Connector .....	26
Main Board In-System Programming (ISP) Header .....	27
<b>Expansion Board .....</b>	<b>28</b>
Expansion Board Power .....	29
Choosing How to Power Your Robot .....	29
Setting Power Jumpers and Connecting a Second Battery .....	29
Motor Ports .....	29
Expansion Board Servo Ports .....	31
Expansion Board Digital Input / Output Ports .....	31
Expansion Board Analog / Digital Input Ports .....	31
Analog Ports 8-11 .....	31
Analog Ports 12-15 .....	32
Expansion Board LEDs .....	32
Infrared Transmitter LEDs .....	32
Infrared LED Modulation on Digital Outputs 17 & 18 .....	33
Expansion Board EEPROM .....	33
Expansion Board In-System Programming Header .....	33
<b>Appendix A - ATmega128 Port/Pin Usage .....</b>	<b>35</b>

## Introduction

The IntelliBrain™ robotics controller is a microcontroller board designed to enable robotics enthusiasts like you to build intelligent robots using a modern programming language, Java™, and a powerful microcontroller, the Atmel ATmega128. The IntelliBrain controller's design makes it easy for you to interface with popular sensors and effectors such as the CMUcam, Fairchild IR photo-reflector sensors, Sharp IR range sensors, Devenatech sonar and compass sensors, infrared remote controls, hobby servo motors, DC motors, and speech synthesizers. In addition, the IntelliBrain incorporates an LCD display, push buttons, a thumbwheel, a buzzer, and LEDs to provide a flexible human interface. Finally, the IntelliBrain includes a high-speed RS232 host port to enable you to program and debug your robot's intelligence using the RoboJDE™ Java-enabled robotics software development environment on your personal computer.

The IntelliBrain is made up of two circuit boards, a main board and an expansion board. You can purchase and use the IntelliBrain main board alone or together with the expansion board. The dimensions of the boards and positioning of the mounting holes make it easy for you to mount an IntelliBrain controller on the IntelliBrain-Bot chassis, a robot chassis constructed using Lego® bricks, the Portland Robotics Society's Mark III mini-sumo chassis or another robot chassis.

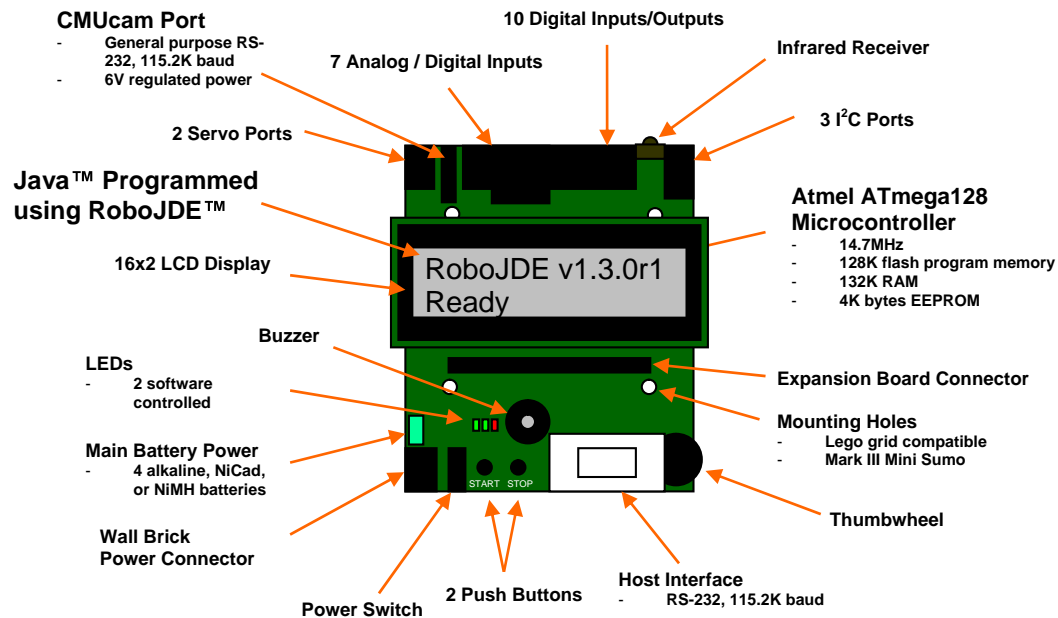


Figure 1 - IntelliBrain Main Board Features

## Main Board Features

Figure 1 shows the main features of the IntelliBrain main board. The main board dimensions are 3.2 x 2.8 inches. The LCD module dimensions are 1.4 x 3.1 inches.

## Expansion Board Features

Figure 2 shows the main features of the IntelliBrain expansion board. The expansion board dimensions are 2.2 x 2.8 inches. The expansion board mounts on the main board via its expansion connector and the LCD connector. The expansion board includes an LCD connector which allows the LCD module to be attached to the expansion board.

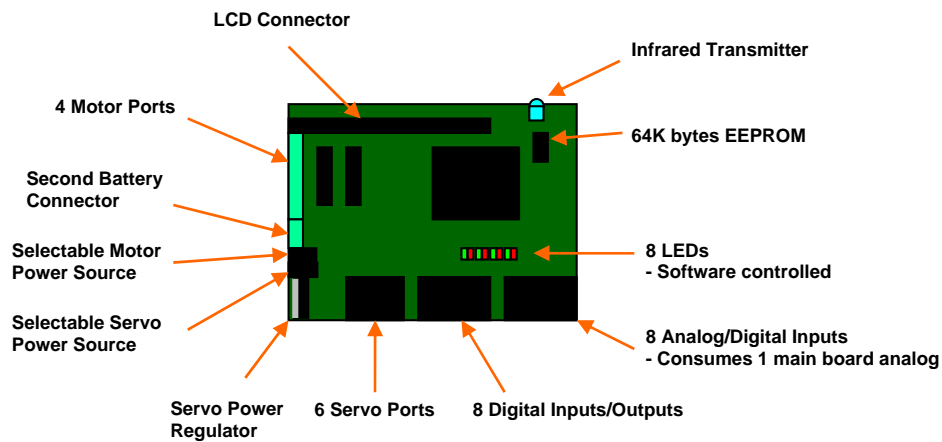


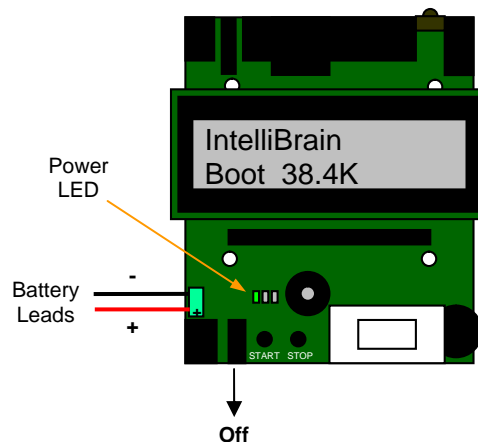
Figure 2 - IntelliBrain Expansion Board Features

## Getting Started

The following sections describe steps you will need to complete to get started with your IntelliBrain™ robotics controller.

### Connecting Power

The IntelliBrain supports a variety power options, described in later sections. The following procedure describes how to power your IntelliBrain with 4 AA batteries.

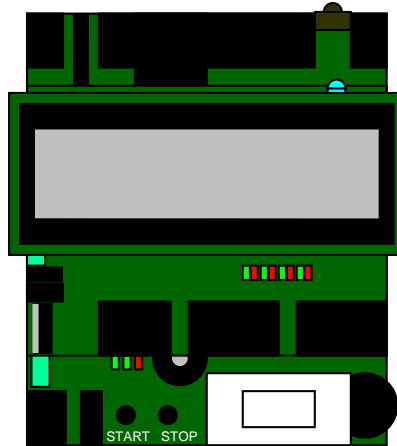


**Figure 3 – Making Battery Connections**

1. Check that the main board power switch is in the off position, as shown in Figure 3.
2. If your IntelliBrain has an expansion board, remove it by following the directions in the *Installing and Removing the Expansion Board* section.
3. Attach the red lead from the battery holder to the positive terminal on the main board battery terminal block and the black lead from the battery holder to the negative terminal, as shown in Figure 3.
4. Insert four fresh AA NiMH (recommended), Ni-Cad or alkaline batteries in the battery holder in the orientation indicated on the battery holder.
5. If your IntelliBrain has an expansion board, install it by following the directions in the *Installing and Removing the Expansion Board* section.
6. Switch the power switch on. The green power LED will illuminate and a welcome message will display on the LCD screen.

### ***Installing and Removing the Expansion Board***

The expansion board mounts to the main board via two connectors, the 40 pin expansion board connector on the lower half of the board and the 14 pin LCD module connector. The LCD module mounts in the 14 pin connector on the expansion board when the expansion board is installed.



**Figure 4 - IntelliBrain with Expansion Board Installed**

### **Installing the Expansion Board**

1. Check that the power for both the main board and expansion board is switched off.
2. Carefully remove the LCD module from the main board, rocking it very slightly from end to end to slide the pins out of the connector without bending them.
3. Remove any jumpers that are installed on the main board expansion header. (See Figure 12 - Expansion Board Connector Pin Out.)
4. Note the orientation of the installed expansion board – shown in Figure 4.
5. Hold the expansion board by its left and right edge in your right hand with the expansion connectors facing out (components facing your palm) and your thumb near the light green motor port terminal block.
6. Hold the main board in your left hand with the connectors and components facing out.
7. Align the connectors on the main board and the expansion board being careful to check that all pins are properly aligned.
8. Slowly and evenly press the two boards together, rocking the boards very slightly if necessary to slide the connectors completely together.



9. Once the pins are fully inserted, press firmly on both connectors to ensure they are fully seated.
10. Align the LCD module pins with the 14 pin LCD connector on the expansion board.
11. Slowly and evenly press the LCD module to fully seat the connector.

**Removing the Expansion Board**

1. Check that the power for both the main board and expansion board is switched off.
2. Carefully remove the LCD module from the expansion board, rocking it very slightly from end to end to slide the pins out of the connector without bending them.
3. Grip the expansion board left and right edges with one hand and the main board edges directly below the expansion board with the other hand.
4. Carefully rock the expansion board very slightly while pulling the two boards apart evenly, being careful to avoid bending the connector pins.
5. If you will be using the main board without the expansion board, insert jumpers at the locations marked on the main board expansion header, as shown in Figure 12 - Expansion Board Connector Pin Out.
6. Align the LCD module pins with the 14 pin LCD connector on the main board.
7. Slowly and evenly press the LCD module to fully seat the connector.

***Establishing a Host Connection***

The IntelliBrain connects to a host computer via a DB9 female to DB9 male extension cable (wired straight through).

1. Attach the male end of the cable to the DB9 connector on the IntelliBrain main board.
2. Attach the female end of the cable to the host PC. If your PC has USB ports, but not serial ports you will need to install a USB to serial adapter cable, such as the IOGear GUC232A, on the host first.
3. Install RoboJDE on the host computer.

4. As described in the *RoboJDE User Guide*, Configure RoboJDE's settings (Tools->Settings menu item) to select the controller type (IntelliBrain), the COM port IntelliBrain is attached to, and 38400 baud.

Note: The default baud rate for the IntelliBrain is 38.4K baud. It is best to establish communication between the host and the IntelliBrain before attempting to switch to higher baud rates. The IntelliBrain displays the currently configured baud rate when the boot loader is started (see next step).

5. Following the instructions in the *RoboJDE User Guide*, build, load and run the "HelloWorld" example program.

## Changing the Host Port Baud Rate

1. Start the IntelliBrain in bootstrap mode by holding the STOP button down while turning power on or by pressing the STOP button immediately after switching power on.
2. Observe the bootstrap loader's welcome message – similar to the message shown in Figure 3. The bootstrap loader indicates the currently set baud rate. The factory default is 38.4K (38400) baud.
3. Press the STOP button repeatedly until the baud rate you would like to use is displayed.
4. Press the START button. The IntelliBrain will store the new baud rate setting so it is retained when you power the board off.
5. Use RoboJDE's Tools->Settings menu to configure the same baud rate on the host.
6. Press the START button to start the virtual machine, or power the controller off and back on again to start the virtual machine.

## Programming IntelliBrain

You program the IntelliBrain using the Java language. RoboJDE makes it easy for you to create a program, compile it, download it to the IntelliBrain and execute it.

## Application Programming Interface Documentation

RoboJDE includes full on-line documentation for its Application Programming Interface (API). The API allows your program to interface with the IntelliBrain hardware, as well as sensors and effectors attached to the IntelliBrain. Click on the blue book icon on the RoboJDE tool bar to view the API documentation.

Browse to the documentation for the `IntelliBrain` class for full documentation on the methods that allow you to access the IntelliBrain controller's features.

RoboJDE also includes an API quick reference graphic for the IntelliBrain (see file `IntelliBrainAPI.pdf` in the "docs" folder where you install RoboJDE or on the docs folder on the RoboJDE CD).

This document provides brief programming examples of using the IntelliBrain API to access IntelliBrain features. These programming examples are intended to provide basic information to help you get started using various features. Consult the API documentation for more in-depth information on using each feature.

## Using RoboJDE

Consult the *RoboJDE User Guide* for information on using RoboJDE.

## Error Messages

All run-time errors, except internal virtual machine errors, are reported to your Java application as exceptions. Consult the *RoboJDE User Guide* for more information on exceptions.

If the virtual machine determines it has encountered an internal error it will turn the red fault LED on and then print "Internal Error" on the first line of the LCD screen followed by additional information on the second line. Finally the virtual machine will execute a breakpoint.

When at a breakpoint, the LCD screen displays "Breakpoint" on the first line followed by "ip: <address>", where <address> is the hexadecimal value of the virtual machine's instruction pointer at the time of the breakpoint. This indicates the point of execution in your application when the breakpoint occurred. You can toggle between the breakpoint display screen and the program output by pressing the STOP button. If the breakpoint is a result of an internal error, the program output screen will display the internal error message.

Internal errors should rarely or never occur. They are the result of errors in the virtual machine. You can't fix internal errors by changing your application, though you may be able to avoid them by changing your application. If your application encounters an internal error, refer to the RidgeSoft web site for support information or send email to [support@ridgesoft.com](mailto:support@ridgesoft.com) to report the error.

## Mounting IntelliBrain on a Robot Chassis

The IntelliBrain main board has four mounting holes that enable you to mount it on the IntelliBrain-Bot chassis, a chassis constructed from Lego bricks, the Portland Robotics Society's Mark III mini-sumo chassis (see [www.junun.org](http://www.junun.org)) or another chassis.

## IntelliBrain-Bot Chassis Mounting

Follow the directions in the IntelliBrain-Bot Assembly Guide to mount your IntelliBrain controller on the IntelliBrain-Bot chassis.

### Lego Mounting

The mounting holes in the IntelliBrain main board are spaced to fit the Lego grid. The holes are spaced six Lego grid units apart. You can create mounting posts using four antenna whip pieces, shown in Figure 5. The wide base portion of the antenna fits snugly in the IntelliBrain mounting holes, allowing the IntelliBrain to be mounted on a Lego chassis. You can make the antenna into a short mounting post by cutting an antenna at the point above the base where it narrows.

Slide an antenna through each hole from the bottom of the main board and push it snugly into place. This will give the main board four Lego connectors that will align with other Lego components.



Figure 5 - Lego Antenna Whip 8H

These and many other Lego components can be found for sale on [www.bricklink.com](http://www.bricklink.com).

### Mark III Mini-Sumo Mounting

You can mount the IntelliBrain directly on the Mark III mini-sumo chassis using the hardware that comes with the Mark III chassis kit. The kit is available from [www.junun.org](http://www.junun.org).

Fasten the main board to the top of the chassis mounting posts using the screws and washers provided with the Mark III chassis kit.

### Custom Mounting

You can mount the IntelliBrain on almost any chassis by drilling mounting holes in a square pattern, 1 7/8 inches (48 mm) on center. The diameter of the IntelliBrain mounting holes is 1/8 inch (~3 mm).

### *Connecting Sensors and Effectors*

The IntelliBrain main board has 23 ports that allow you to connect a wide variety of sensors and effectors to your IntelliBrain. Adding an IntelliBrain expansion board increases the number of sensor and effector ports to 48.

Port headers are arranged side-by-side along the forward (top) edge of the main board and along the rear (bottom) edge of the expansion board, as shown in Figure 7. The ground pin for each port is always the pin nearest the edge of the

board. The power pin is always next to the ground pin. The power pin is +5V for all ports except the servo ports (see servos section) and the COM2 port. The power pin on the COM2 port is +6V to supply power to a CMUcam. The next pin – third from the board edge – is always a signal pin. On four-pin ports, the function of the fourth pin depends on the port. Additional details for each port are provided in later sections.

Each port, except for the motor ports, has a 3 or 4 pin male header that provides power, ground and signal connections for the sensor or effector attached to the port, as shown in Figure 6. The sensor connectors can be constructed using the parts and tools listed in Table 1.

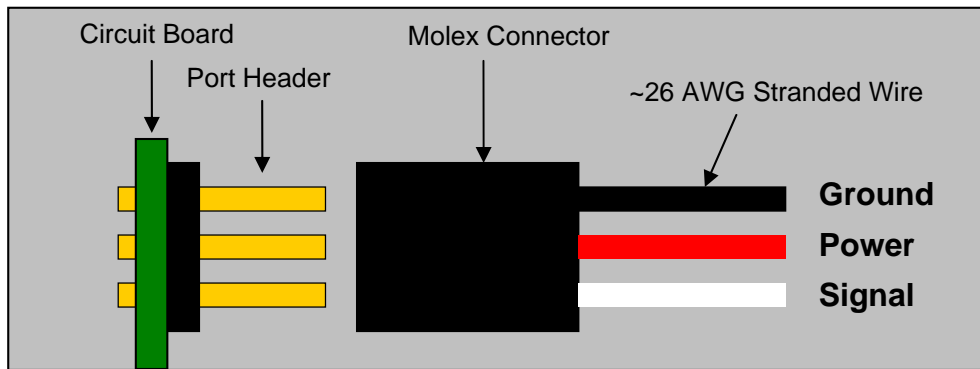


Figure 6 - Typical Sensor / Effector Connection

Table 1 – Sensor / Effector Connector Parts and Tools

Part	Part Number	Source
Crimp Terminals	WM2555-ND	<a href="http://www.digikey.com">www.digikey.com</a>
3 Circuit Housing	WM2801-ND	<a href="http://www.digikey.com">www.digikey.com</a>
4 Circuit Housing	WM2802-ND	<a href="http://www.digikey.com">www.digikey.com</a>
Universal Crimp Tool	WM9999-ND	<a href="http://www.digikey.com">www.digikey.com</a>
Insertion Tool	WM9911-ND	<a href="http://www.digikey.com">www.digikey.com</a>

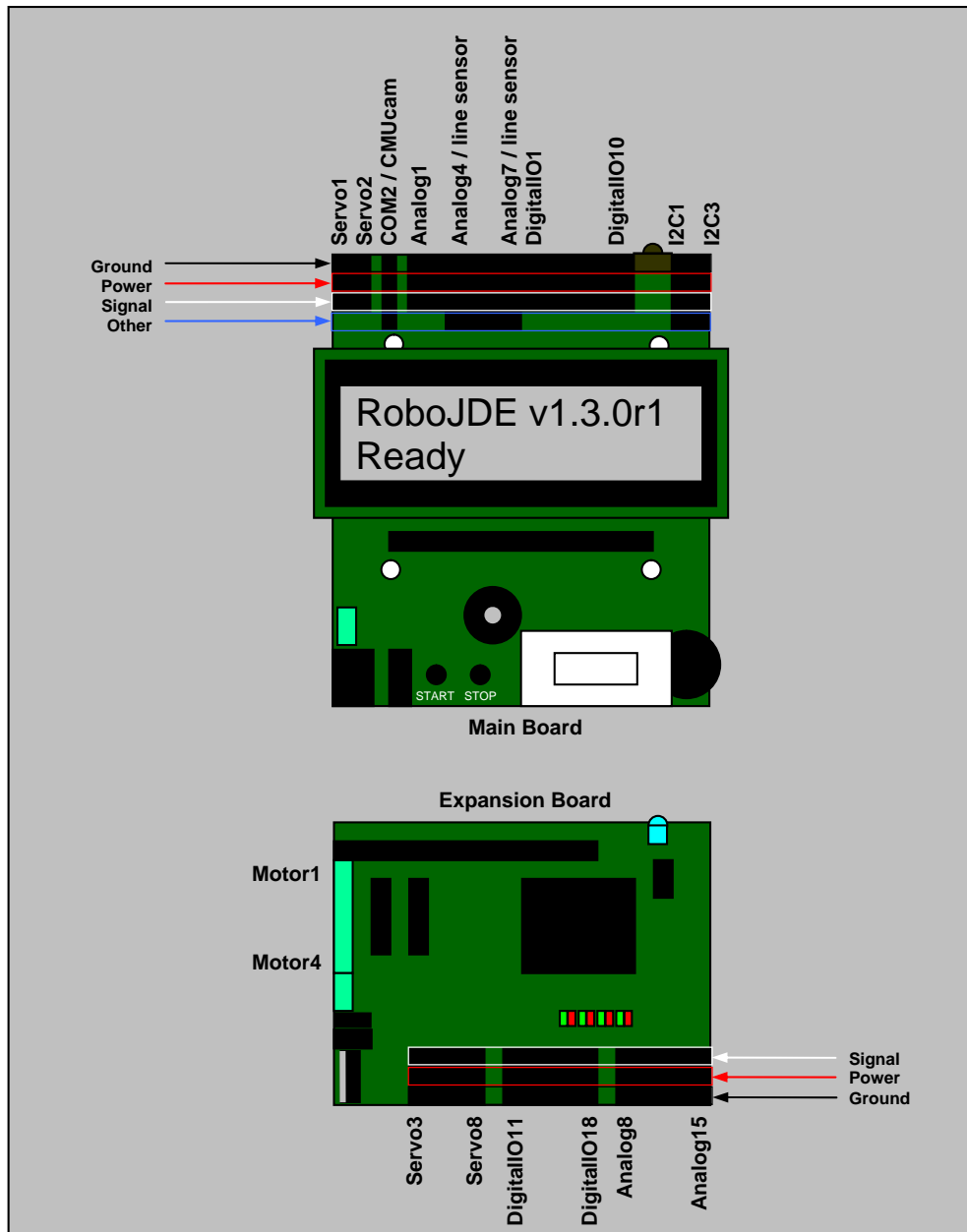


Figure 7 - Ports and Pin Arrangement

### Connecting Handy Board Sensors

Most sensors designed for the MIT Handy Board controller will work with the IntelliBrain but require an adapter cable as shown in Figure 8. The adapter cable can be constructed using the parts listed in Table 1 and 26 AWG stranded wire.

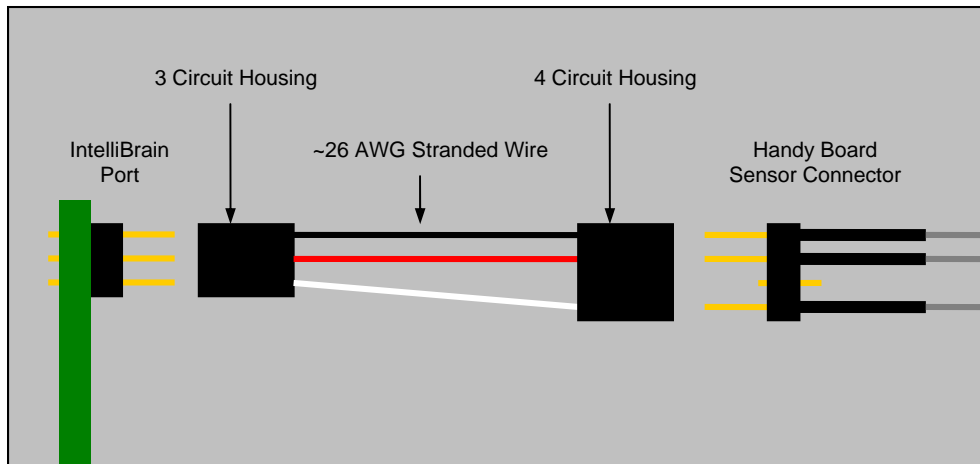


Figure 8 - Handy Board Sensor Adapter Cable

## **Main Board**

The IntelliBrain™ main board provides a powerful robot controller with features that will allow you to build an intelligent robot. The main board provides ports for connecting up to 23 sensors and effectors, including hobby servos, the CMUcam, an assortment of I<sup>2</sup>C devices, analog sensors, digital sensors, shaft encoders, infrared remote controls, and more.

### ***CPU and Memory***

An Atmel ATmega128 microcontroller operating at 14.7 MHz is the central component of the IntelliBrain main board. In addition to the microcontroller, the main board has 128K bytes of external RAM.

### **Flash Memory**

The ATmega128 includes 128K bytes of on chip flash memory. IntelliBrain and RoboJDE™ use this memory for the IntelliBrain bootstrap loader, the RoboJDE virtual machine and for 60K bytes of persistent storage for your Java™ application.

The flash memory has a limited life (minimum 10,000 writes according to ATmega128 specifications). Minimizing downloads to the flash memory by using RAM while developing, testing and debugging your Java applications will extend the life of your IntelliBrain.

### **RAM**

The ATmega128 has 4K bytes of on chip Random Access Memory (RAM). This memory is used by the bootstrap loader and the RoboJDE virtual machine.

Another 128K bytes of external RAM on the IntelliBrain main board provides space for your Java application data and for temporarily loading your Java application while developing and testing. You can load a Java application of up to 30K bytes into RAM, but it will be lost when power is switched off or the batteries drain. Using RAM for program storage while developing your robot application reduces the wear on the flash memory and results in faster program downloading, but the RAM it occupies is not available for your application's objects and stacks. Downloading your program to flash allows you to load larger applications and have the most memory available for its data and stacks.

When you have an application loaded in RAM, it will be the application that runs when you press the START button. If you want to run the application that is in flash memory when there is also an application stored in RAM, switch the IntelliBrain off then back on to erase the application in RAM. This will allow you to run the application stored in flash memory.

The virtual machine manages the allocation of memory as the program creates new objects. The virtual machine also recovers memory (garbage collects) when



the program no-longer references an object. The virtual machine's garbage collector uses a reference counting garbage collector rather than a mark-and-sweep algorithm to provide predictable real-time performance.

## CPU EEPROM

The ATmega128 includes 4K bytes of EEPROM memory with a life of at least 100,000 writes, according to ATmega128 specifications. The first 96 bytes of the EEPROM memory are reserved by IntelliBrain and RoboJDE and are not visible to the application. The remaining 4000 bytes are available to the application.

### *Programming Example*

```
static final int ADDRESS_OF_SOMEDATA = 0;
:
EEPROM cpuEEPROM = IntelliBrain.getCpuEEPROM();
:
cpuEEPROM.write(ADDRESS_OF_SOMEDATA, 1234);
:
int someData = cpuEEPROM.readInt(ADDRESS_OF_SOMEDATA);
```

## Direct Memory Access

Under most circumstances you should access features of the IntelliBrain controller via the Java API rather than attempting to access the microcontroller's registers or memory directly. RoboJDE does provide direct memory access via methods in the `com.ridgesoft.vm.VM` class; however, direct use of these methods by end users is not recommended. Should you choose to use direct memory access methods, you must be careful to avoid interfering with RoboJDE. All RAM memory is managed by the virtual machine; therefore, it is not advisable to use the direct memory access methods to access RAM.

## ***Main Board Power Connections***

The IntelliBrain main board provides connectors for two DC power sources, allowing you to power the board using a battery (pack) or a regulated DC power supply. Using a DC power supply while developing and debugging your robot on the desktop will extend the life of your battery by disconnecting it when the power supply is attached.

The IntelliBrain controller does not charge the battery when plugged in, it simply disconnects it. You may choose to use a rechargeable or non-rechargeable battery. If you use a rechargeable battery, you must recharge your battery with a separate battery charger.

A terminal block allows you to connect a battery holder, as discussed in the *Connecting Power* section. The power supply jack allows you to connect a regulated DC power supply. The following discussion and Table 2 provide information to help you select a battery arrangement and a power supply that suit your application.

Table 2 – Power Source Notes

Power Source	Notes
Main Battery	<ul style="list-style-type: none"> <li>▪ 4 NiMH, Ni-Cad or alkaline cells (6V) cells are recommended for low to modest power applications</li> <li>▪ up to 6 NiMH, Ni-Cad or alkaline cells (9V) can be used to support higher power applications, however, do not exceed the maximum input voltage of servos if powering servos</li> <li>▪ NiMH or Ni-Cad batteries are recommended when powering servos</li> <li>▪ other options are available when using the expansion board (see Expansion Board section)</li> </ul>
Power Supply	<ul style="list-style-type: none"> <li>▪ regulated 5V or 6V DC recommended</li> <li>▪ female, 2.1 mm x 5.5 mm, center positive plug</li> </ul>

### Choosing a Battery and/or a Power Supply

The IntelliBrain main board includes a two stage voltage regulator designed to fully and efficiently use the charge in four standard NiMH, Ni-Cad, or alkaline battery cells. Depending on the sensor and effector load, the board can operate on four to six NiMH, Ni-Cad or alkaline cells (9V maximum recommended). However, if you use the servo ports YOU MUST ENSURE THE MAXIMUM VOLTAGE RATING OF YOUR SERVOS IS NOT EXCEEDED, OTHERWISE THEY MIGHT BE DAMAGED.

If you are not using an expansion board, the main board servo ports are powered directly from the main battery or DC power supply. You must ensure that each power source does not exceed the maximum voltage rating of the servos. Normally four NiMH, Ni-Cad or alkaline cells or a 5V or 6V regulated power supply work well. A NiMH or Ni-Cad battery is recommended when powering servos or motors because they can deliver higher current and maintain higher output voltage under loading conditions produced by servos and motors than can a similar alkaline battery.

If you are using an expansion board, you will have several more servo power options, which are discussed in the *Expansion Board* section.

### Main Power Supply Loading Considerations

There are many factors that affect the number of sensors and effectors you can use with your IntelliBrain controller. One primary consideration is the total load the main board power regulator can support. The main board power regulator supplies the regulated power for the main board, the expansion board and any sensors and effectors, other than servos and motors. The amount of current the main regulator can supply depends on the input voltage and the thermal limits of the regulator. If the power supply is overloaded it will either enter thermal shutdown, or not be able to maintain the +5 volt output. In either case, the controller will reset.

Table 3 provides approximate values of the maximum current output available from the main board's regulated power supply after deducting the current consumed by the main board circuitry. The expansion board consumes approximately 50 mA, which will reduce the values in Table 3 when the expansion board is present.

**Table 3 – Main Power Regulator Output**

Main Battery	Approximate maximum total current available to sensors and effectors
4 alkaline cells	450 mA
4 NiMH or NiCad cells	600 mA
6 NiMH or NiCad cells	800 mA

### Battery Indicator

Digital input / output port3 (IO3) may be used as a battery indicator for the main battery by installing jumper J2, under the LCD module. Port IO3 must not be used for a sensor or effector if jumper J2 is installed.

#### *Programming Example*

```

DigitalInput batteryGood = IntelliBrain.getBatteryInput();
:
if (!batteryGood.isSet()) {
    // low battery handler
:
}

```

### Host Port (COM1)

The IntelliBrain main board provides a high speed RS232 serial connection to your host computer for software download and debugging. This port can operate at baud rates between 115.2K baud and 9600 baud. The highest baud rate you can use may be limited by the host computer and the quality or length of the cable you use. See the *Establishing a Host Connection* section for more information on connecting IntelliBrain to your host computer.

You can also make use of the COM1 port to interface with other serial devices such as GPS receiver.

#### *Programming Example*

```

// Print some text to the RoboJDE run window
System.out.println("some text");

```

By default `System.out` goes to both the LCD and the Run window. You can direct `System.out` to just the Run window as follows:

```

System.out = new PrintStream(VM.getDebugOutputStream());

```

*Programming Example – Using COM1 to Interface with Other Devices*

```

SerialPort com1 = IntelliBrain.getCom1();
com1.setSerialPortParams(115200,
                        SerialPort.DATABITS_8,
                        SerialPort.STOPBITS_1,
                        SerialPort.PARITY_NONE);
InputStream inputStream = com1.getInputStream();
OutputStream outputStream = com1.getOutputStream();
:
int inData = inputStream.read();    // receive data
:
outputStream.write(outData);        // transmit data

```

**LCD Display**

The LCD display allows your application to output text messages to its two-line by sixteen character LCD screen.

*Programming Example*

```

// Address the display lines separately
Display display = IntelliBrain.getLcdDisplay();
display.print(0, "first line");
display.print(1, "second line");

// Write scrolling text to the LCD screen
System.out.println("This is more than 16 characters");

```

**Main Board LEDs**

The IntelliBrain main board has three LEDs. The left most LED is green and indicates power is on. The middle LED is a green “status” LED. You may use this LED as you choose in your program. This LED always illuminates while the START button is pressed. The right most LED is a red “fault” LED. The virtual machine illuminates this LED if it encounters a problem. You may also use this LED in your program, but you can’t prevent it from being illuminated if the virtual machine detects a fault.

*Programming Example*

```

LED statusLED = IntelliBrain.getStatusLed();
LED faultLED = IntelliBrain.getFaultLed();
:
statusLED.toggle();
:
faultLED.on();

```

**Buzzer**

The magnetic buzzer may be used to provide user feedback, such as clicking when a button is pressed, beeping when your robot detects certain events, or playing tunes to entertain observers.

### *Programming Example*

```
Speaker buzzer = IntelliBrain.getBuzzer();
:
buzzer.beep();
:
buzzer.click();
:
// play C note for one quarter second
buzzer.play(262, 250);
```

### **Push Buttons**

The START and STOP buttons start and stop the execution of your application. These buttons are also used to interact with the bootstrap loader (see the *Bootstrap Loader* section).

Once your program is executing, you can use the START button for any purpose you choose. You can also gain control of the STOP button as follows:

```
IntelliBrain.setTerminateOnStop(false);
```

### *Programming Example*

```
PushButton startButton = IntelliBrain.getStartButton();
:
if (startButton.isPressed()) {
    // wait for the button to be released
    startButton.waitReleased();

    // do something based on START being pressed
    :
}
```

### **Thumbwheel**

The thumbwheel provides an analog user input. It may be used for many functions such as calibration, scrolling through a list of sensor readings or controlling the speed of a motor. Depending on its position, the thumbwheel will return a value between 0 and 1023 when your program samples it.

### *Programming Example*

```
AnalogInput thumbwheel = IntelliBrain.getThumbWheel();
:
// generate 0 - 9 index based on thumbwheel position
int index = (thumbwheel.sample() + 57) / 114;
```

### **Main Board Servo Ports**

The IntelliBrain main board has two servo ports for hobby servos (see Figure 7 - Ports and Pin Arrangement). These ports may be used to control positioning servos or servos modified for continuous rotation. The ports support Futaba compatible servos with a 1 to 2 ms pulse control signal (1.5 ms center position).

The servo power and control lines are routed across the expansion header to provide additional powering options and additional servo ports when the expansion board is installed. When the expansion board is not present, three jumpers must be inserted between pins 1-2, 5-6, and 7-8 on the expansion connector, as shown in Figure 12 - Expansion Board Connector Pin Out.

Attach your servos to the main board by plugging them into the servo port as shown in Figure 6 - Typical Sensor / Effector Connection.

The servo ports are powered directly from the main battery or DC power supply when the expansion board is not present. **YOU MUST ENSURE THE BATTERY AND / OR POWER SUPPLY VOLTAGE DOES NOT EXCEED THE MAXIMUM SPECIFIED FOR YOUR SERVOS.** See the main board power section for more information on powering when not using an expansion board. See the expansion board power section for more information on powering servos when the expansion board is present.

The main board servo ports are fused through a single 1 amp automatic resetting fuse.

#### **Programming Example**

```
// Positioning servo
Servo servo = IntelliBrain.getServo(1);
:
servo.setPosition(75);

// Continuous rotation servo
Motor motor = new ContinuousRotationServo(
    IntelliBrain.getServo(2),
    false);
:
motor.setPower(Motor.MAX_FORWARD);
```

### **Serial / CMUcam Port (COM2)**

The COM2 port (see Figure 7 - Ports and Pin Arrangement) is an RS232 serial port. (Note: The COM2 port uses RS232 signal levels, not TTL signal levels). In addition to transmit, receive and ground pins, the port also provides a regulated +6V power supply with a 1 amp automatic resetting fuse to power the CMUcam or CMUcam2.

For information on using the CMUcam or CMUcam2 with the IntelliBrain controller see the RidgeSoft application note titled *Vision Sensing with the CMUcam*.

In addition to the CMUcam, the COM2 port can be used to interface with other devices that provide an RS232 interface, such as a Global Positioning System (GPS) receiver. Figure 9 shows how to construct an adapter cable to connect to a device that is designed to connect to a standard 9 pin COM port on a PC. Table 4 lists the parts and tool required to attach the 4 pin connector to the cable shown in Figure 9. In addition, you will need to solder the wires to the DB9 connector.

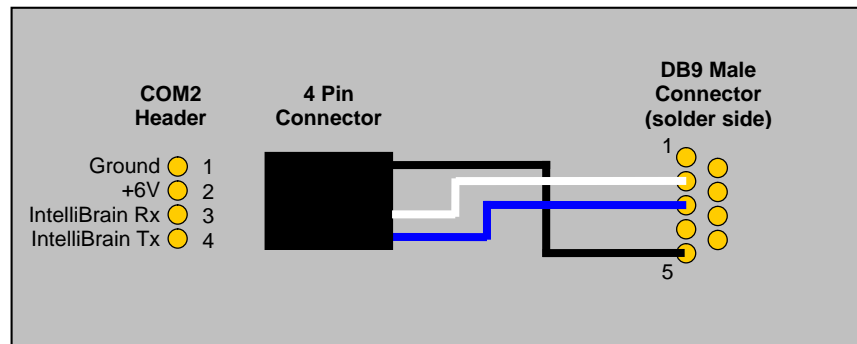


Figure 9 – COM2 DB9 Adapter Cable

Table 4 – DB9 Adapter Cable Parts and Tools

Qty	Description	Part Number	Source
4	Crimp Terminals	WM2555-ND	<a href="http://www.digikey.com">www.digikey.com</a>
1	4 Circuit Housing	WM2802-ND	<a href="http://www.digikey.com">www.digikey.com</a>
1	DB9 Male Connector	209M-ND	<a href="http://www.digikey.com">www.digikey.com</a>
	26 AWG stranded wire	A3049x-100-ND	<a href="http://www.digikey.com">www.digikey.com</a>
1	Universal Crimp Tool	WM9999-ND	<a href="http://www.digikey.com">www.digikey.com</a>

### Programming Example

```

SerialPort com2 = IntelliBrain.getCom2();
com2.setSerialPortParams(115200,
                        SerialPort.DATABITS_8,
                        SerialPort.STOPBITS_1,
                        SerialPort.PARITY_NONE);
InputStream inputStream = com2.getInputStream();
OutputStream outputStream = com2.getOutputStream();
:
int inData = inputStream.read();    // receive data
:
outputStream.write(outData);        // transmit data

```

## ***Main Board Analog / Digital Input Ports***

Seven ports on the IntelliBrain main board provide for Analog-to-Digital conversion (see Figure 7 - Ports and Pin Arrangement). All seven ports may also be used as digital inputs. All seven ports have a software controlled pull-up resistor built into the ATmega128 chip. When you sample an analog port it will return a value between 0 and 1023, proportional on the voltage (0 – 5 volts) on the signal pin (pin 3).

### **Analog Ports 1 - 2**

Analog ports 1 – 2 include a .1 uF capacitor on the signal line for noise filtering that is particularly useful when interfacing with Sharp infrared range sensors such as the GP2D12. Sensors should be connected to these ports as shown in Figure 6 - Typical Sensor / Effector Connection.

### **Analog Port 3**

Analog port 3 is used as an analog multiplexing port for the expansion board. You should not use it if the expansion board is installed.

### **Analog Ports 4 - 7**

Analog ports 4 – 7 have an additional pin and optional electronics on the signal pin (see Figure 10) to support interfacing to infrared emitter-detector sensors, such as the Fairchild QRB1134. The fourth pin connects to ground through a 220 ohm resistor to provide a current limited power circuit for the infrared emitter.

Figure 10 illustrates the additional electronics and shows how to connect QRB1134 sensors to these ports. See Figure 12 - Expansion Board Connector Pin Out, to determine where to insert the jumper. When the expansion board is present, the jumper is not needed because the expansion board always connects the infrared detector electronics.

You can connect other sensors to pins 1 through 3 of these ports, leaving pin 4 unconnected, just as you would with the other analog ports; however, you should remove the jumper on the expansion header. If you have an expansion board installed, attach your other analog sensors to the expansion board and use these ports for infrared emitter-detector sensors.

### ***Programming Example***

```
AnalogInput lineSensor = IntelliBrain.getAnalogInput(4);  
:  
int lineSensorReading = lineSensor.sample();
```



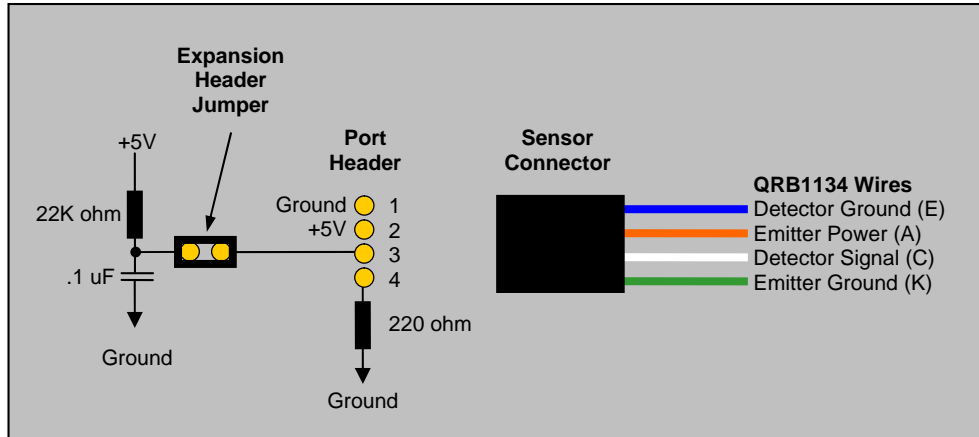


Figure 10 - Connecting QRB1134 Line Sensors to Analog Ports 4 - 7

### Main Board Digital Input / Output Ports

There are ten digital input / output (IO) ports on the IntelliBrain main board (see Figure 7 - Ports and Pin Arrangement).

Each port is software configurable as an input or an output.

All ports have a software selectable pull-up resistor in the ATmega128 chip when configured as an input.

You can use digital IO ports 3 through 6 for measuring input pulse durations. You will find this useful for interfacing with sensors such as the Devantech SRF04 sonar range finder which use pulse duration modulation to indicate the measured value.

You can use digital IO port 3 as a battery indicator, as described in the *Battery Indicator* section.

Sensor and effectors should be attached to these ports as shown in Figure 6 - Typical Sensor / Effector Connection.

### Programming Example

```
// A digital input
IntelliBrainDigitalIO digitalIn = IntelliBrain.getDigitalIO(1);
digitalIn.setDirection(false);
digitalIn.setPullUp(true);
:
if (digitalIn.isSet()) {
    // input is high
}

// A digital output
IntelliBrainDigitalIO digitalOut = IntelliBrain.getDigitalIO(2);
digitalOut.setDirection(true);
```

```

        :
        // set the output high
        digitalWrite.set()
        :
        // pull the output low
        digitalWrite.clear();

```

## I<sup>2</sup>C Ports

The IntelliBrain has a single I<sup>2</sup>C bus with 3 headers that allow you to interface it to a wide variety of I<sup>2</sup>C slave devices. Some popular robotics sensors and effectors that you may want to interface your IntelliBrain to are:

- Devantech SRF08 sonar range finder
- Devantech CMPS03 magnetic compass
- Devantech SP03 text to speech synthesizer
- Devantech MD03 motor driver
- Devantech MD22 dual motor driver

The IntelliBrain is an I<sup>2</sup>C master and the sensors and effectors you attach are I<sup>2</sup>C slaves. Each I<sup>2</sup>C slave must have a unique device address which is used to communicate to the device. The IntelliBrain treats the device address as 8 bits consisting of 7 address bits followed by a zeroed bit as the least significant bit. The least significant bit is used as a read-write indicator when the IntelliBrain communicates with slave devices. The IntelliBrain virtual machine manages the read write bit so you only need to be concerned with the value of the 7 address bits when using the API. Consult the technical documentation for the I<sup>2</sup>C devices you use for more information on their I<sup>2</sup>C addresses.

Figure 11 shows how to connect your I<sup>2</sup>C devices to the IntelliBrain headers.

The I<sup>2</sup>C bus is also routed to the expansion header (see Figure 12 - Expansion Board Connector Pin Out).

### Programming Examples

```

// Devantech SP03 example
DevantechSP03 sp03 =
    new DevantechSP03(IntelliBrain.getI2CMaster());
sp03.speak(0, 5, 10, "Hello");

```

```

// Direct I2C device access example
I2CMaster i2cMaster = IntelliBrain.getI2CMaster();
:
// read the byte in the device at "address" by
// writing the 2 byte address and reading back 1
// byte into resultBuffer array
byte[] writeBuffer = new byte[] {
    (byte)(address >> 8),

```

```

        (byte)address
    };
    byte[] resultBuffer = new byte[1];
    i2cMaster.transfer(deviceAddress, writeBuffer, resultBuffer);

```

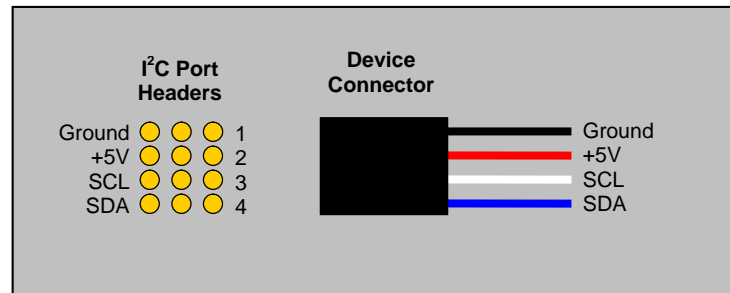


Figure 11 - Connecting I<sup>2</sup>C Devices

### ***Infrared Receiver***

The infrared receiver on the IntelliBrain main board is modulated at 38 kHz to be compatible with many television remote controls. The infrared transmitter LED on the expansion board is also modulated at 38 kHz allowing you to use the transmitter and receiver together for object sensing and low bandwidth communication between robots.

### ***Programming Example***

```

    IrRemote irRemote =
        new SonyIrRemote(IntelliBrain.getIrReceiver());
    :
    int data = irRemote.read();
    if (data != -1) {
        // data received from IR receiver
        :
    }

```

### ***Bootstrap Loader***

Bootstrap loader software is programmed into a protected area of flash memory when IntelliBrain main boards are manufactured. The bootstrap loader allows you to configure the host port baud rate, download the RoboJDE virtual machine to flash memory and download your Java applications into flash memory.

You may enter the bootstrap loader by pressing the STOP button while switching power on or by pressing the STOP button immediately after switching power on. The bootstrap loader displays “IntelliBrain” on the first line of the LCD screen followed by “Boot: <baudRate>” on the second line. The <baudRate> parameter is the currently configured baud rate (9.6K, 19.2K, 38.4K, 57.6K, or 115.2K), used by both the bootstrap loader and the RoboJDE virtual machine to communicate with the host computer. Pressing the START button exits the bootstrap loader and starts the virtual machine.

In addition to setting the host port baud rate (see the *Changing the Host Port Baud Rate* section), the bootstrap loader allows you to revert to the configuration data in the reserved portion of the microcontroller's EEPROM to the factory settings. The bootstrap loader also allows you to view the main board's serial number.

### **Reverting to Factory Settings**

Reverting to the factory settings will set the baud rate to 38.4K and clear all other configuration data in the reserved portion of the EEPROM. (Currently the baud rate is the only configurable parameter stored in EEPROM. If other parameters are added in the future, they will be cleared by this operation.) Use the following procedure to revert to the factory default configuration:

1. Switch the power off and back on.
2. Press the STOP button repeatedly until you see the message "CLEAR CONFIG?" displayed on the LCD screen.
3. Press the START button to return to the factory configuration settings.

Note: You may exit the configuration mode without making changes by pressing the STOP button repeatedly until the initial bootstrap loader message reappears on the LCD screen.

### **Viewing the Serial Number**

You may view the board's serial number, hardware revision and model number using the following procedure:

1. Switch the power off and back on.
2. Press the STOP button repeatedly until you see the serial number displayed.
3. Press the STOP button again to return to exit the configuration mode and display the initial bootstrap loader message.

### ***Expansion Board Connector***

The expansion board attaches to the main board via two connectors, the Expansion Board Connector, shown in Figure 12 and the LCD connector. When the expansion board is installed, the LCD module attaches to the LCD connector on the expansion board.

See the *Installing and Removing the Expansion Board* section for information on installing and removing the expansion board.

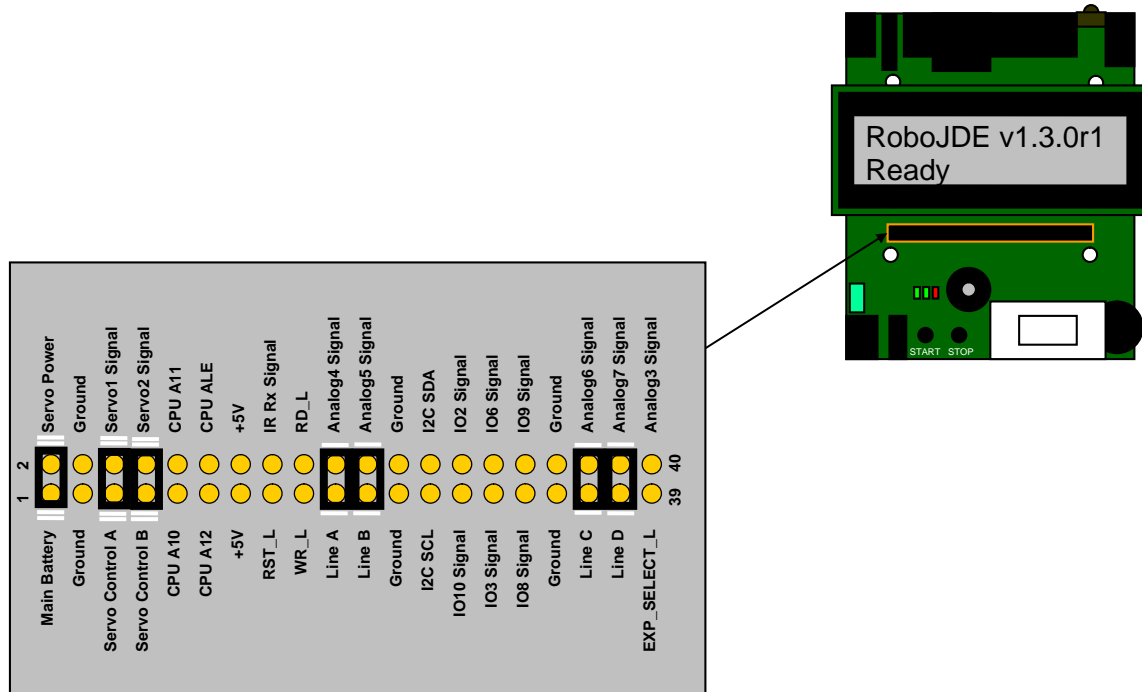


Figure 12 - Expansion Board Connector Pin Out

**Main Board In-System Programming (ISP) Header**

The header on the IntelliBrain main board just above the thumbwheel is an In-System Programming port (see Figure 13). This port is used when the board is manufactured and you should not attempt to use it.

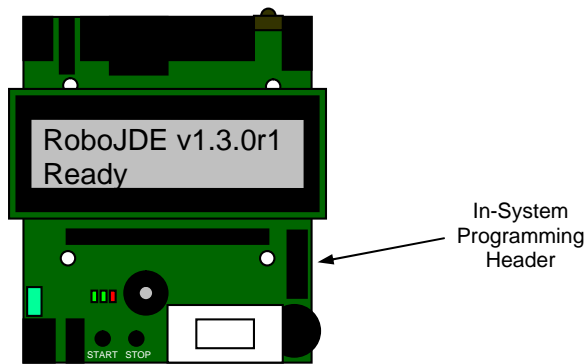


Figure 13 – Main Board In-System Programming Header

## Expansion Board

The IntelliBrain™ expansion board, shown in Figure 2, adds the following additional features to your IntelliBrain:

- 4 DC motor ports
- 6 additional hobby servo ports
- 7 additional analog / digital input ports
- 8 additional digital input / output ports
- 8 additional software controlled LEDs
- 1 infrared transmitter LED
- 64K bytes of additional EEPROM storage
- support for a second battery
- optional servo power regulation
- flexible powering options

The expansion board connects to the main board via two connectors as described in the *Expansion Board Connector* section. See *Installing and Removing the Expansion Board* for instructions on installing and removing the expansion board.

**Table 5 - Battery Recommendations**

Power Requirements	Recommendation
Low	<ul style="list-style-type: none"> <li>▪ one main battery</li> <li>▪ 4 cells</li> </ul>
Moderate <ul style="list-style-type: none"> <li>▪ 2 or 3 motors and/or servos</li> <li>▪ low current demand</li> <li>▪ 9V or lower motor voltage</li> </ul>	<ul style="list-style-type: none"> <li>▪ one main battery</li> <li>▪ up to 6 cells (maximum 9V)</li> <li>▪ regulate servo power if main battery is over 6V</li> <li>▪ add a heat-sink to servo regulator if necessary</li> </ul>
Moderate to High <ul style="list-style-type: none"> <li>▪ 3 to 5 motors and/or servos</li> <li>▪ high current demand</li> <li>▪ up to 12V – 1A motors</li> </ul>	<ul style="list-style-type: none"> <li>▪ two batteries</li> <li>▪ 4 to 6 cell main battery</li> <li>▪ motors and servos powered from second battery</li> <li>▪ regulate servo power if second battery is over 6V</li> <li>▪ add a heat-sink to servo regulator if necessary</li> </ul>
High <ul style="list-style-type: none"> <li>▪ many motors and servos</li> <li>▪ high current demand</li> <li>▪ up to 15V – 1A motors</li> </ul>	<ul style="list-style-type: none"> <li>▪ two batteries</li> <li>▪ 6 cell main battery</li> <li>▪ up to 15V second battery</li> <li>▪ servos powered from main battery through regulator</li> <li>▪ add a heat-sink to servo regulator if necessary</li> <li>▪ motors powered from second battery</li> </ul>

## ***Expansion Board Power***

The IntelliBrain expansion board provides support for a secondary battery to power motors and/or servos. The logic and sensor ports are all powered by the main battery, attached to the main board. The motors and servos may also be powered from the main battery. However, you may choose to use a second battery attached to the expansion board for one of the following reasons:

- motors and/or servo power demands are too high for the main battery
- motors require a higher voltage than the main battery supplies
- better isolate logic circuitry from motor noise
- maximize the power available to logic and sensors powered by the main battery

The expansion board also provides a voltage regulator that you may use to regulate voltage to the servos if your battery voltages are not suitable for servo powering. You may choose either battery as the power source for your servos and you may also choose to regulate or not regulate the power to the servos.

**WARNING: THE SERVO POWER REGULATOR ON THE EXPANSION BOARD CAN BECOME VERY HOT UNDER HIGH LOAD. BE CAREFUL NOT TO TOUCH THE REGULATOR WHEN IT IS HOT.** If your use of the IntelliBrain expansion board causes the regulator to become too hot to touch, attach a heat sink to the regulator or use a different powering arrangement.

## **Choosing How to Power Your Robot**

Table 5 provides recommendations to help you choose how to power your robot.

## **Setting Power Jumpers and Connecting a Second Battery**

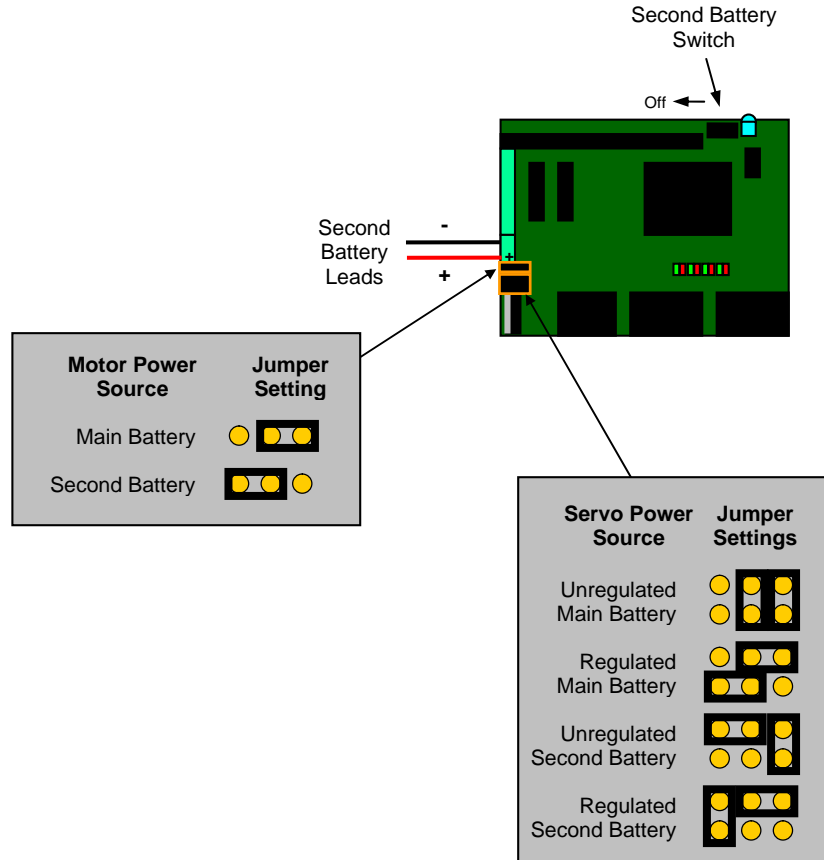
Figure 14 shows how to set the power jumpers and where to connect your second battery.

The servo power source jumpers apply to the servo ports on both the main board and the expansion board.

## ***Motor Ports***

The expansion board provides four pulse width modulated DC motor ports capable of supplying up to 1 amp per port at up to 15 volts. The motor ports provide braking as well as variable forward and reverse power. A jumper allows selection of the main battery or second battery as the power source for the motor ports (see Figure 14 – Power Jumper Settings and Second Battery Connection).

Each motor port consists of two adjacent terminals on the motor terminal block. The motor ports are arranged on the terminal block as shown in Figure 7 - Ports and Pin Arrangement.



**Figure 14 – Power Jumper Settings and Second Battery Connection**

To connect your motors you will need to first remove the LCD module and then connect the motors by inserting each motor's leads into the port's terminal connectors and tightening the screws. If you find a motor spins in the opposite direction of what your software intends, swap the positions of its leads in the terminal block.

By default the eight LEDs on the expansion board show the state of each motor port (see *Expansion Board LEDs*).

### *Programming Example*

```
Motor motor = IntelliBrain.getMotor(1);
:
motor.setPower(Motor.MAX_FORWARD);
:
motor.brake();
:
motor.setPower(Motor.STOP);
```



### ***Expansion Board Servo Ports***

The expansion board adds six servo ports, ports 3 through 8 (see Figure 7 - Ports and Pin Arrangement). The expansion board also adds several servo powering options which are discussed in the *Expansion Board Power* section.

Before attaching servos, first review the expansion board powering options and choose how you will power your servos. **POWERING SERVOS AT THE INCORRECT VOLTAGE MAY DAMAGE THE SERVOS.**

Note: The servo power source jumpers on the expansion board affect the power source for both the main board and expansion board servo ports.

The expansion board servo ports are fused through a 1 amp automatic resetting fuse.

See the main board servo section for information on connecting and using the servo ports. Note: The ground pin is always the pin nearest the edge of the circuit board (see Figure 7 - Ports and Pin Arrangement). **CONNECTING A SERVO INCORRECTLY MAY DAMAGE THE SERVO.**

### ***Expansion Board Digital Input / Output Ports***

The expansion board includes eight digital input / output ports (see Figure 7 - Ports and Pin Arrangement). These ports are numbered 11 through 18. They are similar in function to the main board ports except their pull-up resistor is not under software control and is always active. Each port is software configurable as an input or an output. Digital ports 17 and 18 also provide the ability to drive 38 kHz modulated infrared LEDs as discussed in the *Infrared Transmitter LED* section.

Sensor and effectors should be attached to these ports as shown in Figure 6 - Typical Sensor / Effector Connection.

See the main board discussion of digital input / output ports for more information on using these ports.

### ***Expansion Board Analog / Digital Input Ports***

The expansion board adds eight analog ports, numbered 8 through 15. These ports are multiplexed through analog port 3 on the main board. You should not use analog port 3 on the main board when the expansion board is present. You may also use these ports as digital input ports.

#### **Analog Ports 8-11**

Analog ports 8 through 11 each have .1 uF capacitor between the signal pin and ground.

## Analog Ports 12-15

These ports function similar to analog port 3 on the main board. They have a software controlled pull-up resistor; however, the resistor is on the main board port and will only be active on a particular expansion board port while the port is being accessed by your software.

## Expansion Board LEDs

The expansion board has eight, software-controlled LEDs (see Figure 2 - IntelliBrain Expansion Board Features). By default these LEDs indicate the state of the motor ports. The left most pair of LEDs corresponds to motor port 1 and the right most pair corresponds to motor port 4.

These LEDs can each be used for a different purpose of your choosing by calling the method:

```
IntelliBrain.getExpansionLed(int ledNumber)
```

which disassociates the LED from its motor status function, giving your software full control of it.

The LEDs are numbered 1 to 8 from left to right. The odd numbered LEDs are green and the even numbered LEDs are red.

Table 6 - Motor Port States

Green LED	Red LED	Motor Port State
Off	Off	Off, coast
On	Off	On, forward
Off	On	On, reverse
On	On	On, brake

### Programming Example

```
LED greenLED = IntelliBrain.getExpansionLed(7);
greenLED.on();
```

## Infrared Transmitter LEDs

The infrared transmitter LED on the expansion board is modulated at 38 kHz to be compatible with the infrared receiver on the main board. You may use this LED in conjunction with the infrared receiver to sense objects in front of your robot or to communicate between robots at low bandwidth.

If you use the infrared LED in conjunction with the infrared receiver for object detection, you will need to restrict the infrared light beam it transmits by placing a short piece (~1 inch) of heat shrink tubing over it. Be sure to slide the tubing all the way over the base of the LED.

### *Programming Example*

```
LED irLED = IntelliBrain.getExpansionIrLed();
irLED.on();
:
irLED.off();
```

## **Infrared LED Modulation on Digital Outputs 17 & 18**

You may add two additional 38 kHz modulated infrared LEDs attached to digital ports 17 and 18. When attaching an LED to either of these ports, attach the anode (longer lead) to the power pin and the cathode (shorter lead) to a current limiting resistor (~220 ohm) connected to the signal pin. The signal is active low.

### *Programming Example*

```
IntelliBrainExpDigitalIO digitalOutput =
    (IntelliBrainExpDigitalIO)IntelliBrain.getDigitalIO(18);
digitalOutput.setModulatedOutput();
LED irLED = new BasicLED(digitalOutput, true);
:
:
irLED.on();
```

## ***Expansion Board EEPROM***

The expansion board includes 64K bytes of EEPROM memory accessible via the I<sup>2</sup>C bus (device address 0xA0). The entire EEPROM is available to your application for persistent storage. The EEPROM is specified by the manufacturer to have an endurance of at least 100,000 writes.

### *Programming Example*

```
EEPROM expansionEEPROM = IntelliBrain.getExpansionEEPROM();
:
// write byte 123 to address 0 in the EEPROM
expansionEEPROM.write(0, (byte)123);
:
// retrieve the byte at address 0 in the EEPROM
byte someData = expansionEEPROM.readByte(0);
```

## ***Expansion Board In-System Programming Header***

The header on the IntelliBrain expansion board in the upper right corner is an In-System Programming port (see Figure 15). This port is used when the board is manufactured and you should not attempt to use it.

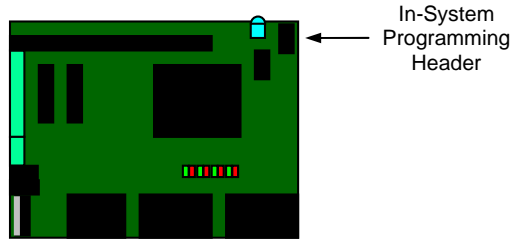


Figure 15 - Expansion Board In-System Programming Header

## Appendix A - ATmega128 Port/Pin Usage

The following table documents the usage of each of the I/O pins on the ATmega128 microcontroller.

**Table 7 - ATmega128 Pin Usage**

Port	Usage
Port A0-7	External RAM & LCD
Port B0	LCD
Port B1	Digital IO8
Port B2	Digital IO9
Port B3	Digital IO10
Port B4	Buzzer
Port B5	Digital IO7
Port B6	Servo control – Controls Servo 1 when expansion board is not present. Controls all 8 servo ports when expansion board is present.
Port B7	Main board servo 2 / Expansion IR LED – Controls Servo 2 when expansion board is not present. Controls expansion board infrared LED when expansion board is present.
Port C0-7	External RAM
Port D0	I <sup>2</sup> C SCL
Port D1	I <sup>2</sup> C SDA
Port D2	COM2 receive
Port D3	COM2 transmit
Port D4	Infrared receiver
Port D5	External RAM
Port D6	External RAM
Port D7	External RAM
Port E0	COM1 (host) receive
Port E1	COM1 (host) transmit
Port E2	Digital IO1
Port E3	Digital IO2
Port E4	Digital IO4
Port E5	Digital IO5
Port E6	Digital IO3
Port E7	Digital IO6
Port F0	Thumbwheel
Port F1	Analog 1
Port F2	Analog 2
Port F3	Analog 3 – Used to multiplex expansion board analogs inputs when expansion board is present.
Port F4	Analog 4
Port F5	Analog 5
Port F6	Analog 6

Port	Usage
Port F7	Analog 7
Port G0	External RAM
Port G1	External RAM
Port G2	External RAM
Port G3	START button & green (status) LED
Port G4	STOP button & red (fault) LED