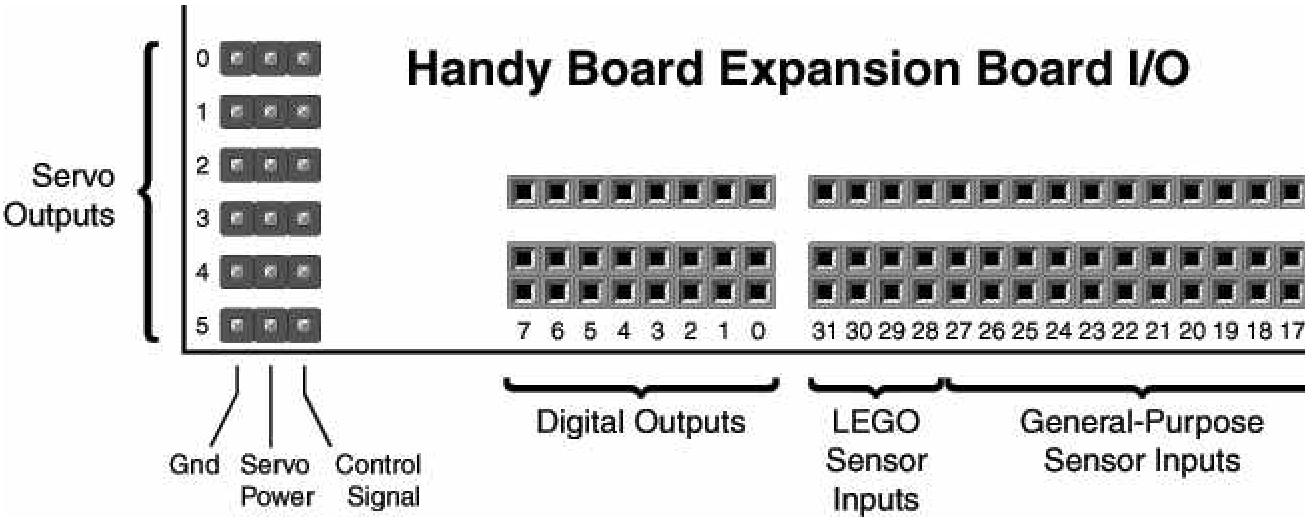
Expansion Board Software Libraries



Sensor Inputs

The Expansion Board sensor inputs are numbered from 16 to 31. The standard `analog()` and `digital()` functions have been augmented allow arguments from 2 to 31:

- Analog sensor ports 0 and 1 on the Handy Board itself may no longer be used. The Expansion Board routes analog signals into these inputs.
- Sensor ports 2 through 6 are the Handy Board's analog inputs;
- Sensor ports 7 through 15 are the Handy Board's digital inputs;
- Sensor ports 16 through 27 are the Expansion Board's general-purpose analog inputs;
- Sensor ports 28 through 31 are the Expansion Board's LEGO-compatible sensor inputs, for use with the LEGO reflectance sensor and the LEGO angle sensor. (These sensor ports may be used as normal inputs, with the caveat that they have a 10K pullup resistor, rather than the normal 47K on other Handy Board sensor inputs.)

Digital Outputs

Software

There are nine digital output pins, numbered from 0 to 8. Digital outputs 0 through 7 are located on headers on the edge of the board (see diagram above); digital output 8 is a PCB pad located in the upper left corner of the prototyping area.

The digital outputs are controlled with the following functions:

`set_digital_out(int pin)`
Sets the selected output pin, producing a +5v level on the output pin. `pin` must be a value from 0 to 8.

`clear_digital_out(int pin)`
Clears the selected output pin, producing a 0v level on the output pin. `pin` must be a value from 0 to 8.

`test_digital_out(int pin)`
Queries the setting of the specified pin. Returns 1 if the pin is set, or 0 if the pin is cleared. `pin` must be a value from 0 to 8.

Electrical

The output signals are provided on the upper row of pins (labeled *I/O Signal* in the diagram). These are CMOS outputs, capable of driving just 5 mA per output.

Servo Motor Outputs

Software

The Expansion Board provides outputs for six servo motors, numbered from 0 to 5. The following function is used to enable and disable the servo control signals:

`init_expbd_servos(int action)`

> If `action` is zero, servo control signals are disabled.
> If `action` is non-zero, servo control signals are enabled.
> Control signals are disabled by default every time the Handy Board is reset.

In addition, six global variables are provided for controlling the position of the six servo motors. These globals are named `servo0`, `servo1`, `servo2`, `servo3`, `servo4`, and `servo5`, for servo outputs 0 through 5 respectively.

When the Expansion Board servo drivers are loaded, these variables are defined as globals in the Interactive C environment. Modify these globals with simple assignment statements; e.g., `servo3= 1500;`

The values determine the length of the timing pulse provided to each servo, which in turn corresponds to a rotational position of the servo motor. Valid values range from 0 to about 3000. *The software does not protect against out-of-range numbers, which will cause the servo to attempt to turn to positions it cannot reach.*

Electrical

Servo motors connect via a three-wire plug. The three signals (from left to right on the Expansion Board) are ground, power, and control. These are typically indicated with black, red, and white wires (respectively) on the servo motor. Make sure to plug the servo motor in properly.

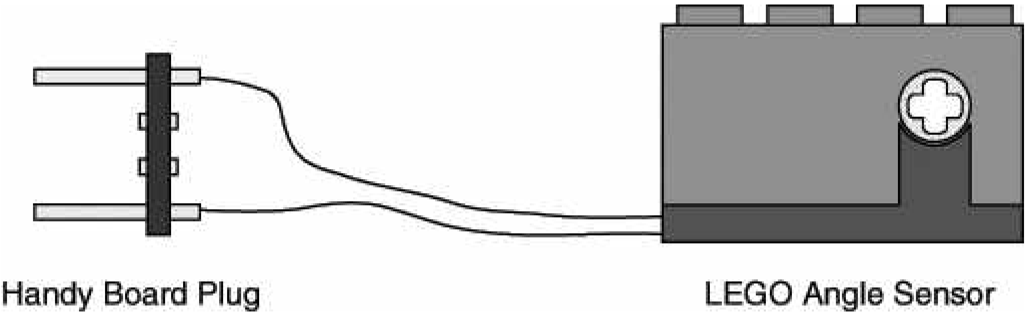
Servo motors come with different kinds of connectors. The Expansion Board is designed for servos with a female header connector on tenth-inch spacing. The Futaba S148 is an example of a good basic servo that uses this connector.

Power is provided to the servo from the Handy Board's internal battery, through the five large diodes near the servo connectors. These diodes will get warm during servo operation. If the diodes get too hot to touch, an external servo power supply will be required.

On the flip side of the Expansion Board is a trace labeled “CUT HERE FOR EXT SERVO PWR.” Using a sharp razor knife, slice through this trace, making sure it is completely severed. Then, connect an external power supply of 5 to 6 volts (e.g., four C or D cells) to the External Servo Power PCB pads. Be sure to get the `+` and `-` polarity correct.

LEGO-Compatible Sensor Inputs

Sensor ports 28 through 31 are compatible with the LEGO reflectance sensor and LEGO angle sensor. Wire either of these sensors to the Handy Board plug as shown below (the angle sensor is depicted):



LEGO Reflectance Sensor

The LEGO reflectance sensor contains a red emitter LED and a visible detector LED. In normal usage, the device is positioned about 1/2 inch away from a surface, and its reading indicates the reflectivity of the surface. Used in this fashion, typical readings range from 180 to 210 (depending on ambient lighting), with smaller values indicating more light (and greater reflectivity).

The analog values are read using the `lego` globals. These should be used to obtain readings from the LEGO reflectance sensor, or any other resistive sensor used in these ports when the LEGO driver is loaded.

`lego0`
: Global variable which is analog reading on port 28 while LEGO drivers are active.

`lego1`
: Analog reading on port 29 while LEGO drivers are active.

`lego2`
: Analog reading on port 30 while LEGO drivers are active.

`lego3`
: Analog reading on port 31 while LEGO drivers are active.

LEGO Angle Sensor

The LEGO angle sensor is a quadrature (direction--sensitive) rotation sensor. It can be used to measure the angular position of a LEGO axle with a precision of 16 counts per revolution. The drivers count upward when turning in one direction and downward when counting in the opposite direction. Hence, the angle sensor can be used to keep track of the precise position of a mechanical system even as its direction of rotation changes.

The drivers also provide a velocity measurement. The velocity readings are updated every 1.024 seconds.

`lego0_counts`
: Global variable which is rotational count of LEGO angle sensor in port 28.

`lego0_velocity`
: Global variable which is rotational velocity of LEGO angle sensor in port 28.

`lego1_counts`
: Rotational count of LEGO angle sensor in port 29.

`lego1_velocity`
: Rotational velocity of LEGO angle sensor in port 29.

`lego2_counts`
: Rotational count of LEGO angle sensor in port 30.

`lego2_velocity`
: Rotational velocity of LEGO angle sensor in port 30.

`lego3_counts`
: Rotational count of LEGO angle sensor in port 31.

`lego3_velocity`
Rotational velocity of LEGO angle sensor in port 31.

The `counts` values are reset to zero every time the Handy Board is reset. To reset the counts to zero under software control, simply assign the global variable to zero; e.g.:

```
lego0_counts= 0;
```

Caveats

The encoder values are polled by the software at a rate of 250 samples per second. This allows a maximum rotational velocity of about 16 revolutions per second. If the axle turns any faster than that, the software will not be able to keep up and counts will be missed.

Because the software must read the four LEGO-compatible inputs 250 times a second, some CPU power is lost to this operation. If you are not using any LEGO sensors, it is advisable to unload the `explego.icb` driver. Simply remove it from the `lib_hb.lis` file.

Please note that direct analog readings using the `analog()` routine will not work properly for ports 28 through 31 (the LEGO-compatible inputs) when the `explego.icb` driver is loaded. Use the `lego0` through `lego3` globals instead.

Polaroid Ultrasonic Driver

The Expansion Board implements the same Polaroid 6500 driver circuit as previously published. To connect the 6500 board into the Expansion Board, perform the following steps:

1. Install the 9-pin flex cable adapter in the position provided.
2. Solder a 1.5" length of solid core hookup wire (22 gauge) to the PCB pad labeled **J12**, thread it between the digital output and analog input headers, and plug it into digital input 7 on the main Handy Board.
3. Use the `sonar.c` driver provided on the Handy Board sonar web page, http://el.www.media.mit.edu/projects/handy-board/software/sonar.html

Library Names and Functions

The Handy Board and Expansion Board library routines are split across multiple files. These are:

`lib_hb.lis`
Root library file. When set up for the Expansion Board, should include the subsequent files listed.

`lib_hb.c`
Motor, switch, and miscellaneous Handy Board functions.
Please note that the analog and digital sensor functions are now contained in a separate files. When using the Expansion Board, `expsens.c` must be loaded; when not using the Expansion Board, load `hbsensor.c` instead.

`expsens.c`
Contains `analog()` and `digital()` functions to subsume both the main Handy Board and the Expansion Board. *If the Expansion Board is not being used, the file named `hbsensor.c` should be used instead.*

`lib_hb.icb`
Contains `_raw_analog()`, an assembly-language driver for reading the analog sensors on the main board. This function is called by the user-level functions `analog()` and `digital()` in `expsens.c` or `hbsensor.c`.

`libexpbd.icb`

Contains `_exp_analog()`, which is used in `expsens.c`. Also has `set_digital_out()`, `clear_digital_out()`, and `test_digital_out()` routines.

`expservo.icb`
Contains driver, user routine, and global definitions for Expansion Board servo motors.

`explego.icb`
Contains driver and global definitions for use with LEGO reflectance and angle sensors.

Download

Download the complete set of libraries for Wintel or for Macintosh.

The Expansion Board libraries were last modified March 3, 1999.

HB site by fredm at handyboard.com / last page update at Wed Sep 20 10:22:32 2000