

Phi -1 shield Documentation



*Last reviewed on 01/03/11
John Liu*

Table of content

1.	Introduction:	2
2.	List of functions:	2
3.	List of possible projects:	2
4.	Parts list:	3
5.	Shield pin usage:	3
6.	List of accessories:	4
7.	Before assembling:	4
8.	Steps to assemble the shield:	5
9.	Testing the shield:	6
10.	Pictures:	7
11.	FAQ:	11

1. Introduction:

Congratulations on getting the best multi-functional Arduino shield! The Phi-1 shield fits Arduino Deumilanove, UNO, MEGA1280, or MEGA2560. It also fits most Arduino clones. I have developed many complete projects with fully-functional codes, available for download at this location: <http://liudr.wordpress.com/phi-1-shield/>. The individual project links are at the end of each paragraph describing the project. Several projects such as the Morse code encoder/decoder and the alarm clock don't require anything other than an Arduino board and the Phi-1 shield so you can load up the code and you're ready to enjoy! You will also get a head start with these codes or simply take parts of the code, such as buttons class, menus, input panel, GPS, etc to quickly construct your own project. The possibility is endless.

2. List of functions:

The Phi-1 shield has the following functions:

- 16X2 LCD character display
- 6 push buttons - four arranged in arrow keys and two more on the side
- 2 RJ11 ports for robust connections with sensors or control devices
- Optional buzzer and LED in place of the RJ11 ports
- Real time clock with battery backup keeps the time when Arduino is turned off
- EEPROM for easy data logging keeps data when Arduino is turned off
- GPS connector and breakout for a popular GPS module
- Reset button for Arduino
- All Arduino pins are brought out for maximal flexibility.
- Hackable for more functionalities (see the end of the assembling)

3. List of possible projects:

- ✓ Alarm clock
- ✓ Standalone or PC data logger
- ✓ Lab data acquisition system (Physics, Chemistry etc)
- ✓ Weather station
- ✓ Input or operating panel, like security panels or garage door opener
- ✓ Handheld GPS
- ✓ Mores code generator
- ✓ The list goes on...

4. Parts list:

An updated list is at <http://dipmicro.com/store/JL-PHI-1>

Link	Item	Qty
http://www.dipmicro.com/store/LCD-1602A-Y	Character LCD	1
http://www.dipmicro.com/store/DS1307N	Real time clock	1
http://www.dipmicro.com/store/XC4-32768	32.768KHz crystal	1
http://www.dipmicro.com/store/HDR40X1F	Female pin headers	2
http://www.dipmicro.com/store/HEADS40T	Male pin headers	1
http://www.dipmicro.com/store/ICS308	8-pin DIP sockets	2
http://www.dipmicro.com/store/BH2025	CR2032 battery holder	1
http://www.dipmicro.com/store/MJTP1230	Buttons	7
http://www.dipmicro.com/store/R4V10-3296	10KOhm Trimpot	1
http://www.dipmicro.com/store/JACK-FRJA-614	RJ11 jacks	2
http://www.dipmicro.com/store/BAT-CR2032	CR2032 coin battery	1
http://www.dipmicro.com/store/LED3RD	LED	1
http://www.dipmicro.com/store/DMT-11B-15	Speaker	1

Table 1. Parts list.



Fig. 1 Unassembled shield with all parts.

5. Shield pin usage:

To use the LCD: First jump pin 0 to pin 7, jump pin 1 to pin 6 (Fig. 3), Then use *LiquidCrystal lcd(8, 9, 7, 6, 2, 4)*; to initialize the display.

Button	Arduino pin	LCD	Arduino pin	Channels	Arduino pin
Up	5	RS	8	Ch1	digital 16 or analog 2
Down	10	EN	9	Buzzer	digital 16
Left	11	D4	7	Ch2	digital 17 or analog 3
Right	3	D5	6	LED	digital 17
B	14	D6	2	I2C	Analog 4, 5
A	15	D7	4	Unused	Digital 5,12,13

Table 2. Pin usage.

6. List of accessories:

The compatible EEPROMs are 24LC256, 24LC512, or 24LC1024.

The compatible GPS module is the EM-406A, sold at sparkfun.com. You need a connector from them as well:

GPS connector: <http://www.sparkfun.com/products/579>

EEPROM: <http://www.sparkfun.com/products/525>

GPS module: <http://www.sparkfun.com/products/465>

To make use of the RJ11 connections, you can choose the following boards. You may download their EAGLE files from <http://liudr.wordpress.com/> and have them made at your favorite PCB house or you may just purchase them from us. We have them manufactured in large quantity and pass the savings to you.

Board	Function
Phi-1 connector board	Provides prototype space for Ics, screw terminals and two LED indicators
Phi-1 connector board small	Provides 0.1" pins to be plugged on to a breadboard
Phi-1 photogate connector board	Provides breakout for a Sharp photogate
Phi-1 stereo connector board	Provides a 3.5mm stereo jack. Can be used for Pasco photogate.
Phi-1 relay board	Provides space for a relay and screw terminals to control high power stuff
	This board also has a power barrel to easily power an electromagnet or else

Table 3. Accessories.

7. Before assembling:

You will need a basic soldering iron with a sharp tip to assemble the shield like this on from RadioShack:

<http://www.radioshack.com/product/index.jsp?productId=2062758>

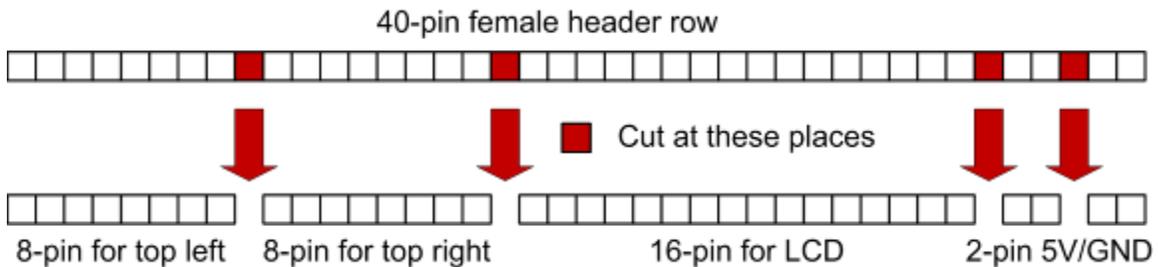
Purchase some 0.032" diameter solder as well as the included solder runs out fast.

Do you want a buzzer and an LED or two RJ11 ports? Buzzer and RJ11-port1 share Arduino analog channel 2. LED and RJ11-port2 share Arduino analog channel 3. You could use the buzzer as an alarm or maybe play a few mono-tunes (not MP3 quality).

You could use the LED as an indicator. These are great for beginners. On the other hand, the RJ11 ports give you two robust connections to external sensors and controls. Jumper wires fall off very easily. You can always de-solder if you desire. Buy some solder wick for de-soldering.

8. Steps to assemble the shield:

1. Step to step pictures can be found at <http://liudr.wordpress.com/phi-1-shield/>
2. Break away five 6-row and one 8-row male headers from the row of 40.
3. Break away one 16-row, two 8-row, two 2-row female headers from a row of 40 as shown below:



4. Break away another row of 8 female headers from the other row of 40.
5. Identify the front side with the LCD outline, a big white rectangle.
6. Female headers for the analog and digital pins (yellow rectangles in fig. 2). Use masking tape to secure the female headers in place. To secure the headers, wrap tape around it, even if it means covering most of its pins. Solder one pin to the board to secure it and then you can remove the tape to access the rest of the pins. The optional female header brings out 3.3V besides other pins.
7. Male headers next to the female headers (red rectangles in fig. 2). Please leave pins 0 and 1 alone. Don't put any male headers pins in those holes (blue rectangle in fig. 2). Tape the male headers onto the board. Solder the first and the last pin to secure it. Remove tape and solder the rest of the pins.
8. Tape the LCD female header to the board and solder it to the board.
9. Two 2- row female headers (5V and GND) on top left, and on right side of the board.
10. Push all buttons in and solder them. Stick the tip of the iron between the prone and the solder pad to quickly heat up the part.
11. Tape the 8-pin DIP sockets and the coin battery holder to the **BACK SIDE** of the board (Fig. 4). Be patient with the battery holder.
12. Carefully tape the GPS connector, leaving space to solder its side. Once soldered on, remove tape and use extreme care while soldering in order not to short the connections or melt plastic.
13. Solder a 6-pin female header next to the GPS connector.
14. The resistor for LCD back light (220Ohm), the crystal (insert and bend it down), the resistors for the buzzer (330Ohm) and LED (150Ohm) if you use them. (FAQ)
15. The trim potentiometer. Use tape to secure it.
16. Solder male headers on LCD pins 1-6, and 11-16. You don't need pins 7-10.

17. Jump pin 0 to pin 7 (purple jumper on fig. 3). Jump pin 1 to pin 6 (green jumper on fig. 3). **This enables the LCD.**(FAQ)
18. If you are using an Arduino Mega, jump analog pin 4 to digital pin 20, and analog pin 5 to digital pin 21. This transfers the I2C clock and data pins to the right pins. After that, set the analog pins 4 and 5 as input and enable internal pull-up resistors. (FAQ)
19. RJ11 jacks. After soldering, bend the pins from the metal shield towards PCB.
20. **Optional:** If you want to hack the RJ11 jacks, here's how to: The RJ11 jack has 4 connections in it. Two of them are 5V and GND, useful to power sensors. Only one of the other two is used, I call it Y. Look to the side of the RJ11 jack, you can see two holes with X and Y marked to them. If you want to use the other connection on the RJ11 jack so you have two I/O lines running on one phone cable, connect the X to an Arduino I/O port. Make sure you're not already using that I/O port for something else!!!
21. **Optional:** If you're not happy with analog 2 going into the RJ11-port1, look for a white circle on the back side of the board. There is a wire running through it. Carefully cut the wire open. Then solder a different I/O port to the Y hole besides the RJ11 jack. Say you want soft serial to run on Arduino pin 10 and 11 and need the RJ11-port1 for a robust connection to your destination, you can cut open the wire inside of the circle, solder pin 10 and 11 to X and Y.
22. **Optional:** You can do the same with the RJ11-port2. The circle is on the front side of the board, at the bottom right.

9. Testing the shield:

Now that you're done, download the testing code to test out all functions:
<http://www.mediafire.com/?v2b08ju0tffxamu>

You need to set the clock before you can use the code.

1. Uncomment this line "#define setRTC" in the main program.
2. Update the time in setup() to the current time.
3. Upload the sketch to arduino to set the time.
4. Then comment the line "#define setRTC" and recompile and upload to arduino.
5. If you don't do step 1, you will be stuck on the clock test. If you don't do 4, the clock will be reset every time arduino resets.

Follow the steps in this video: <http://www.youtube.com/watch?v=sYxAij8gQj0>

Download more codes at <http://liudr.wordpress.com/>. Use the test code as a template and develop your project!

10. Pictures:

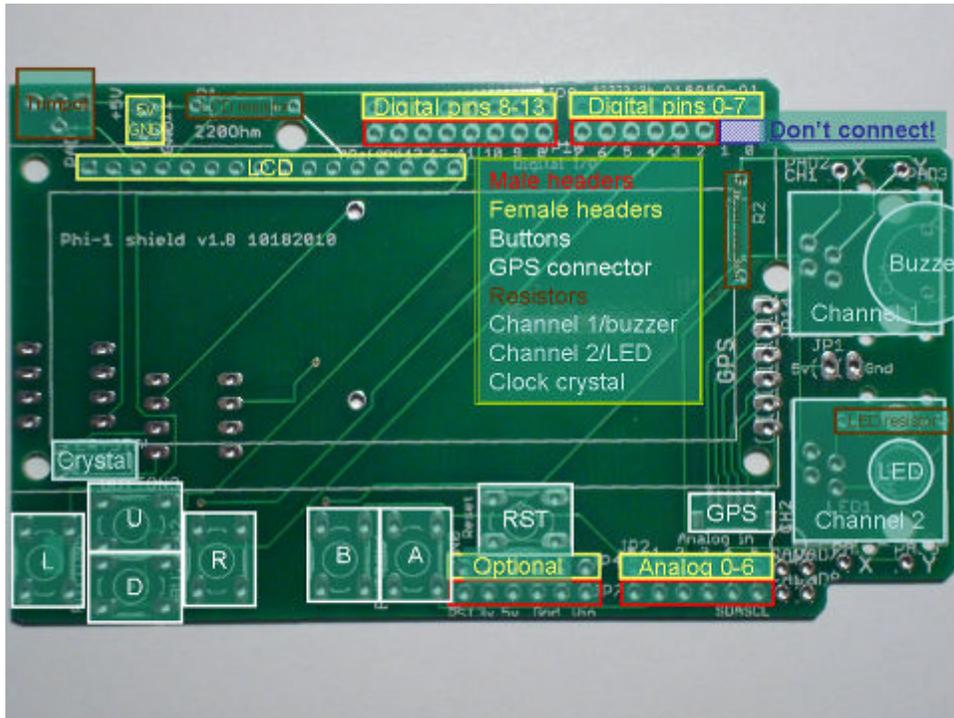


Fig. 2 Front side of the shield PCB with all parts marked.

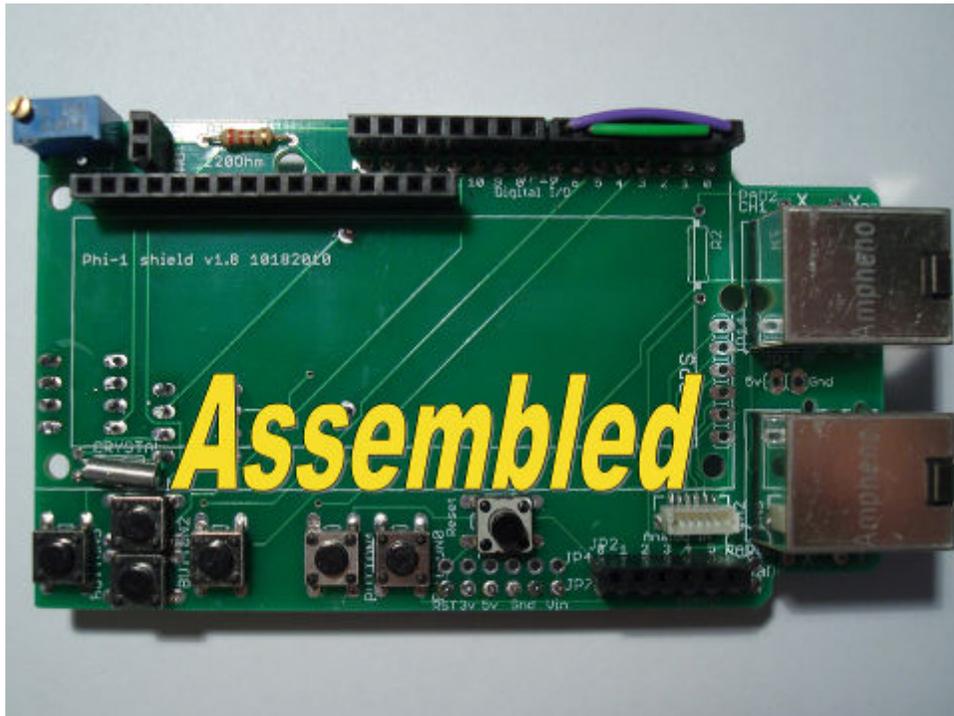


Fig. 3 Front side of a completely assembled shield with RJ11 and GPS connectors

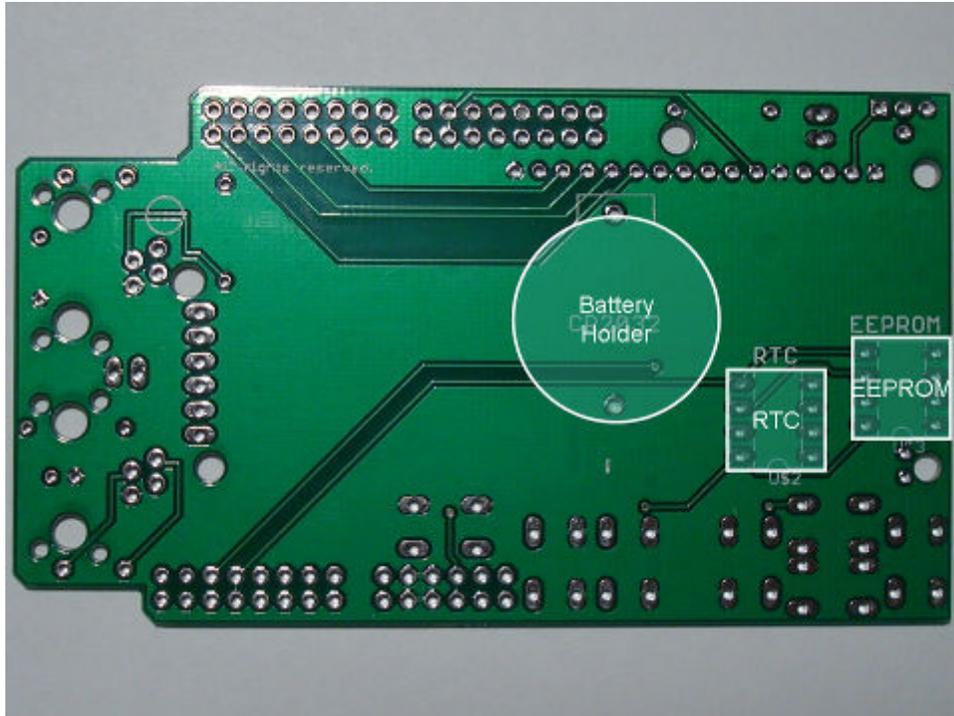


Fig. 4 Bottom side of the shield PCB with all parts marked.

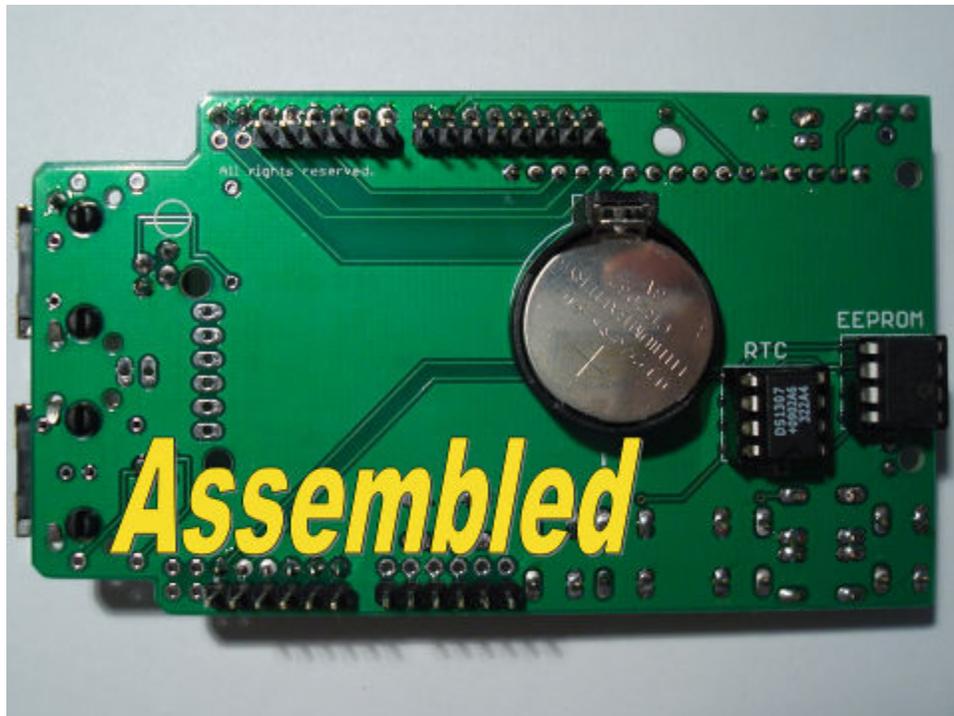


Fig. 5 Back side of a completely assembled shield with Real time clock and EEPROM

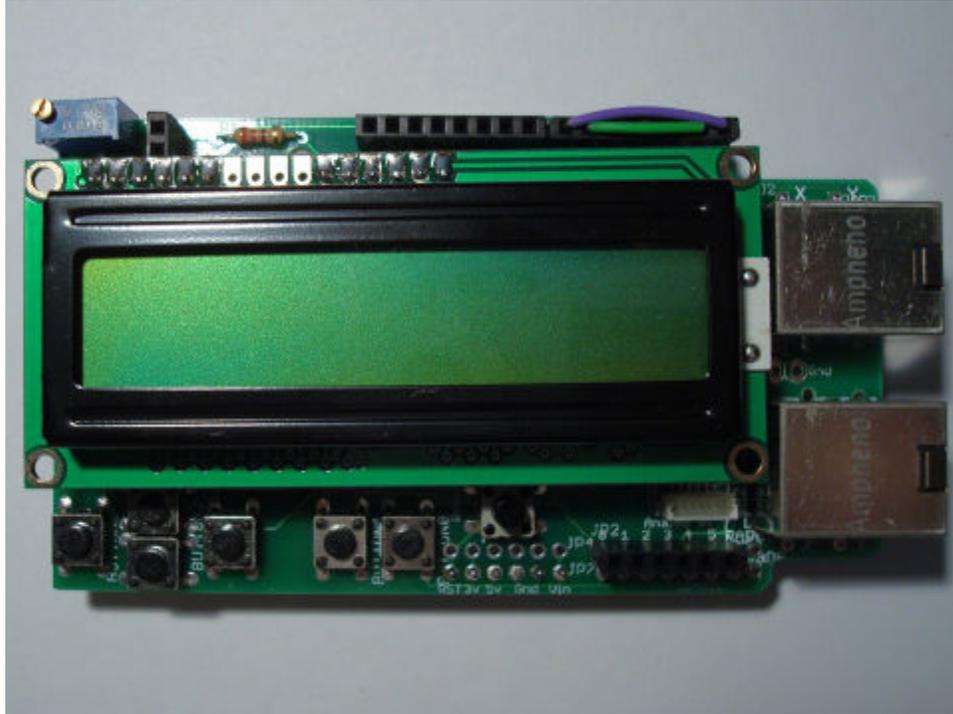


Fig. 6 A completely assembled shield with LCD and RJ11 plugs

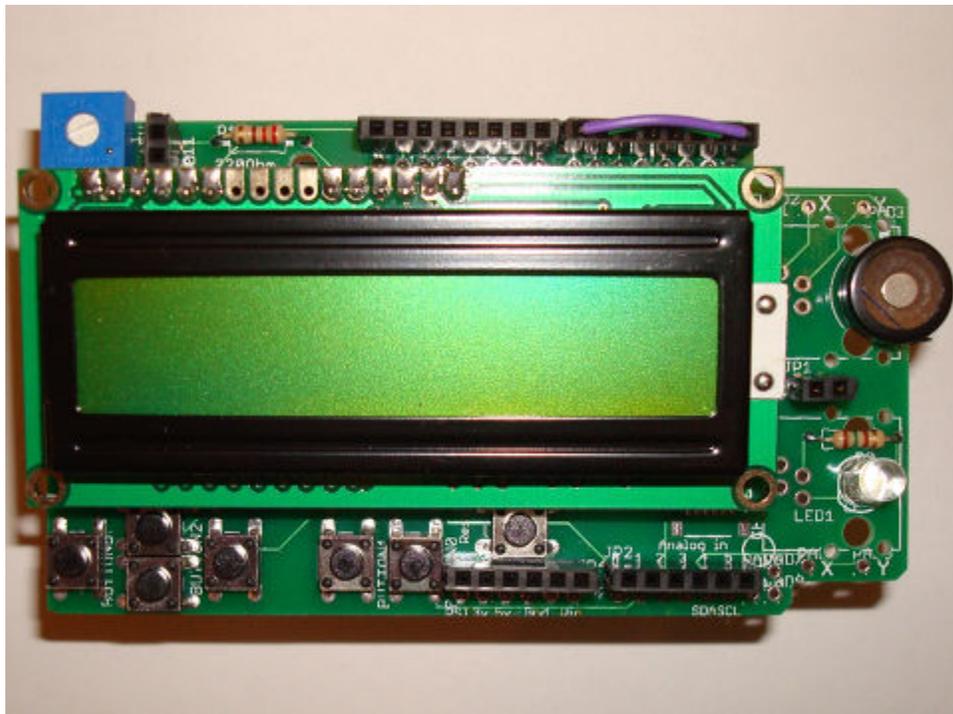


Fig. 7 A completely assembled shield with buzzer and LED

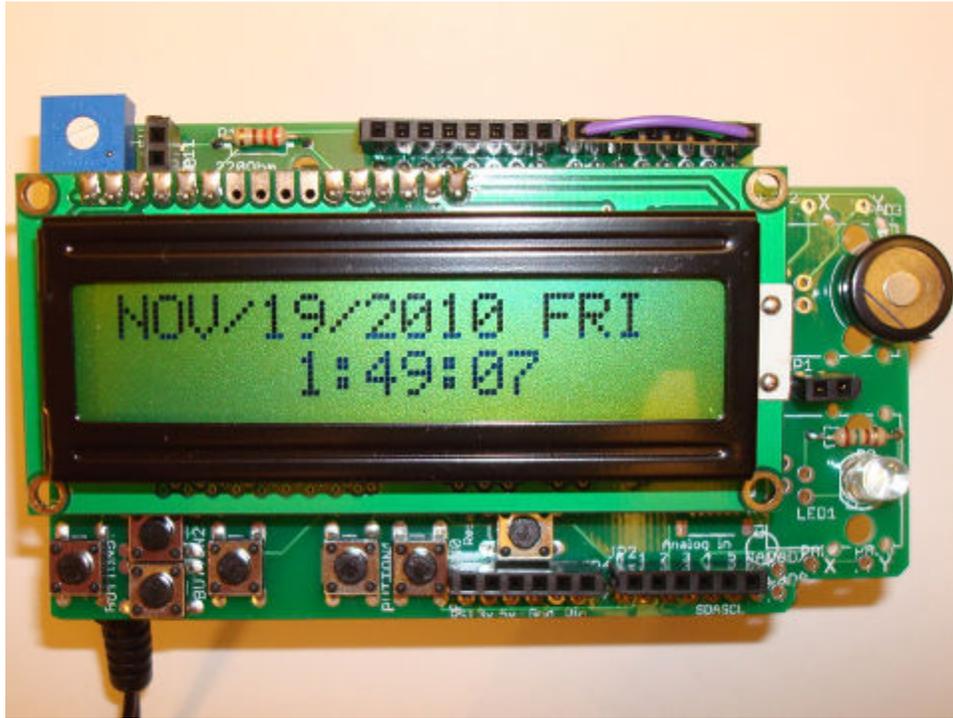


Fig. 8 A completely assembled shield running a clock (with LED and buzzer)



Fig. 9 A shield with an EM 406A GPS module (LCD is removed to show the GPS)

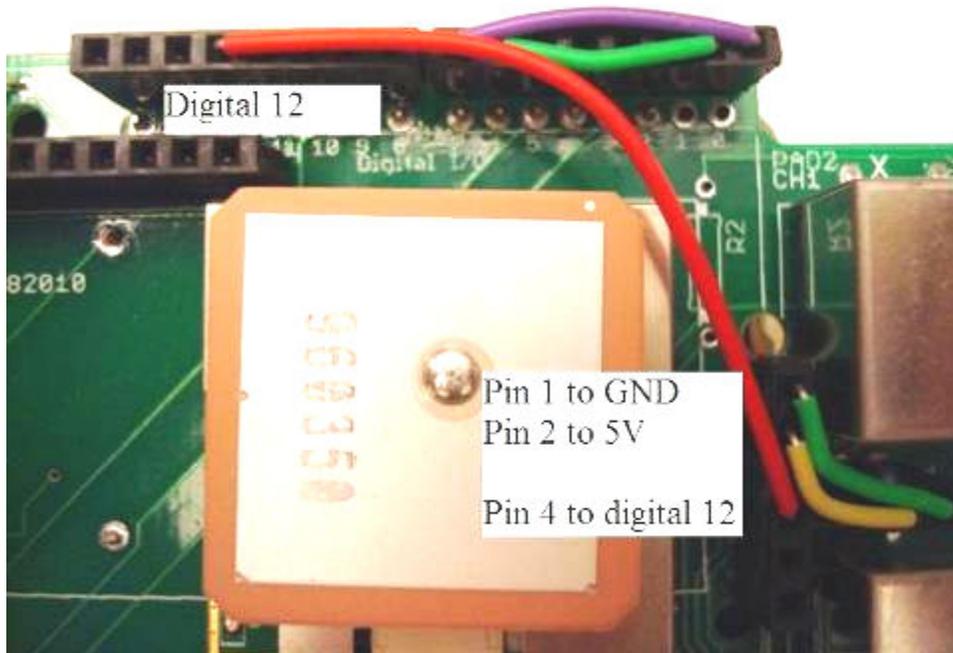


Fig. 10 The detailed connections of the GPS module (EM 406A)

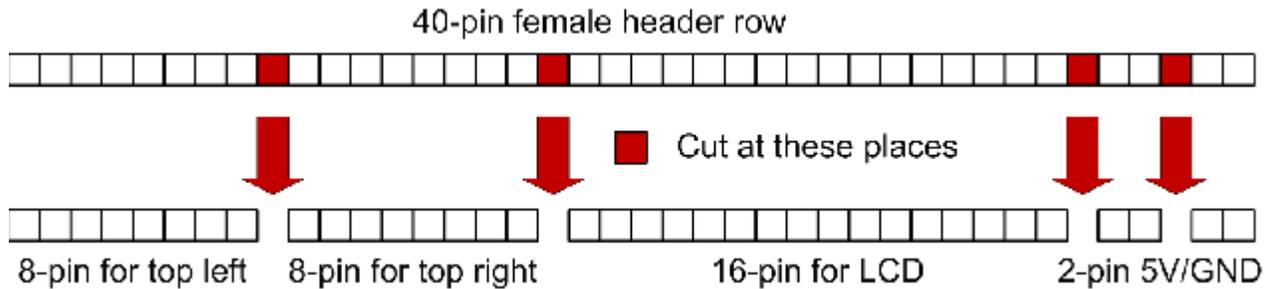
11. FAQ:

1. [How do I get the 2-pin, 6-pin, and 8-pin female headers pictured in your parts photo?](#)
2. [Where do I get an EEPROM, the GPS connector, and the GPS module?](#)
3. [Where to get a decent soldering iron for the assembling?](#)
4. [How do I solder the headers so they don't end up tilted or rotated?](#)
5. [How do I enable the LCD?](#)
6. [How to choose LCD back light resistor?](#)
7. [How do I secure the LCD to the shield, besides pushing the headers together?](#)
8. [How do I test my assembled shield?](#)
9. [How do I make use of the RJ11 connectors?](#)

How do I get the 2-pin, 6-pin, and 8-pin female headers pictured in your parts photo?

I take one 40-pin female header (hope you bought two as required on the parts list), then I use a regular 45 degree cutter (\$2USD at a hardware store) to cut it to the size I need. Say I need a 6-pin female header, I take a row of female headers, clip on the 7th pin on both the pin side and the hole side, this destroys the 7th pin but after trimming the edges, I get a 6-pin header. I will then trim off the edge on the rest of the row of female pins, then cut one more time to get another header out. Remember, if you need a header with 6 pins, cut on the 7th pin so you get a complete 6-pin header. So for a 40-pin female header you

will get say 8(keep)-1(break)-8(keep)-1(break)-16(keep for lcd)-1(break)-2(keep for 5V/GND)-1(break)-2(keep). Break indicates you destroy the pin to separate into the size you need. Then use the other 40-pin header row to get one more 8 and two 6. Here is an illustration:



Where do I get an EEPROM, the GPS connector, and the GPS module?

Here is a list of the links to the vendor:

GPS connector: <http://www.sparkfun.com/products/579>

EEPROM: <http://www.sparkfun.com/products/525>

GPS module: <http://www.sparkfun.com/products/465>

Where to get a decent soldering iron for the assembling?

I have used both a corded and a cordless iron. I like the cordless one slightly better because it is not as intimidating as the corded one.

Corded: radioshack basic soldering set for \$7.99

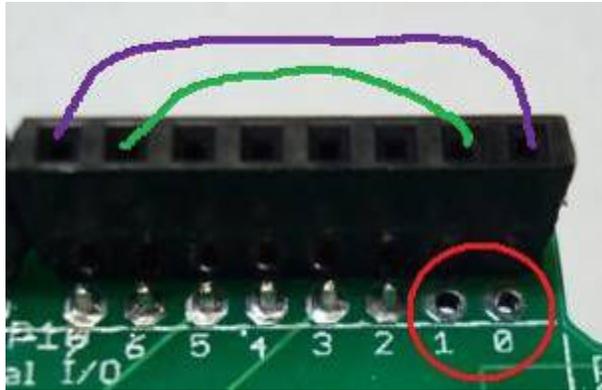
Cordless: Weller BP860MP cordless dual-power soldering iron for less than \$20 depending on where you buy.

How do I solder the headers so they don't end up tilted or rotated?

The best way is to tape the headers to the board using masking tape. You will wrap the header with tape only leaving one pin exposed for soldering. Then solder this one pin so the header will not move when you remove the tape. After that, solder all rest pins, starting from the opposite end to the first soldered pin.

How do I enable the LCD?

Jump pin 0 to pin 7 (purple jumper on picture). Jump pin 1 to pin 6 (green jumper on picture). This enables the LCD. Make sure that there are no male pins on arduino pins 0 or 1. If there are, then cut them off at the bottom with a pair of wire cutter. See the following picture, the arduino pins 0 and 1 are NOT supposed to be connected.



DO NOT solder pins at 0 and 1.



I once forgot this and I put pins on the arduino pin 0 and 1. I had to cut them off with a wire cutter, not too hard to do.

How to choose LCD back light resistor?

My basic rule of thumb: start with a large resistor (300Ohm), reduce it until you're satisfied with the brightness of the back light. Remember the back light doesn't have to glow visibly in a well-lit room. You just need it bright enough when all lights are out. This saves battery and life time of the back light.

Say you bought this from dipmicro (<http://www.dipmicro.com/store/LCD-1602A-B>), its spec sheet is here: (<http://www.gzliyedz.com/download/yjpfz/162-4.pdf>) It says $V=4.1$ $I=120\text{mA}$ about the back light. This current is kind of big but with calculation, you need 7.5Ohm resistor. I would use a larger value like 150 Ohms. I don't want my backlight to burn so bright that it reduces its life time. All you need is when the lights are turned off in the room, the back light makes the display totally visible.

<http://www.sparkfun.com/products/709>

This is very similar to the one dipmicro sells. The comments are quite helpful too.

How do I secure the LCD to the shield, besides pushing the headers together?

There are three mounting holes on the shield that line up with the LCD mounting holes. Since the LCD mounting holes can vary a bit from manufacturer to manufacturer, I didn't put a specific screw size there. I would go to a local hardware store (like acs) and go to the screw/nut sector to hand pick some. You will need a standoff between the LCD and the shield. I ended up using M4 standoff and M4 screws. They're a bit too large but are available to me. I just screwed the screw through the board holes to open it up by just enough. Then screw down the standoff till it fits. Nothing fancy.

How do I test my assembled shield?

First, you want to visually inspect everything to make sure there are not short circuits. Then carefully plug the shield into arduino while the arduino is not powered. Supply power to arduino. If arduino will not be recognized by the computer, then you have a short, which draws too much current that it trips the resettable fuse on arduino (bless the arduino designers for this feature). Check again. Remove excessive solder with some solder wick (radioshack) or solder sucker.

Once arduino powers up, load the test program from the list of codes on this page, test everything. If you don't see any message on the LCD, adjust the trimpot all the way to find the best spot for a good contrast. Without a proper setting on the pot, the LCD messages will not be visible.

How do I make use of the RJ11 connectors?

I will be posting connector boards for the RJ11 connectors. At the same time, I will try to ask dipmicro to carry these small boards as well. Check on my blog and dipmicro on updates. You may save time and money if I make like 100 of them and have dipmicro offer them for sale, other than you making 2 of them at a higher cost.