Experimental Incubator

Soldering and connection guide

Thank you for purchasing the prototype version of the Experimental Incubator. We hope that it will be of use to you and your fermentation projects.
We provide this guide and more info via https://wiki.techinc.nl/index.php/Incubator_v21b
If some things are unclear in this guide, we might have updates/fixes/FAQ's provided there that'll answer your question. Note that the arduino comes **WITHOUT SOFTWARE** and needs to be PROGRAMMED first; ask one of the people selling this kit to you to do that, or find the software on the URL above. Note: you will need a USB-to-serial connector for this.

First of all, in the package you received, you should have found the following parts as well as the names used on the PCB to indicate where they should go.

- A Single-sided PCB (green thing with lines, holes and white text)
- IC1: An Arduino Pro Mini , 5v 16Mhz, Atmega 328 on it (recangular PCB with chip.)
- An 16x2 LCD module with I2C 'backpack' soldered on the back. (hooks up to P5)
- A DS1307 Real-Time-Clock module (with battery-clip) (solders onto P4)
- A battery (goes in clip)
- A DS18B20 temperature probe (metal thing on a black wire) for connecting to K5 or K6
- J1: A USB Type-B PCB connector
- K5+K6: 2x 3-pin PCB screw-terminals (blue, three screws, can't miss it)
- SW1-SW6: 6x 'PCB-mounted Mini Micro Momentary Tactile Push Switches'.... AKA: buttons
- R1-R8: 8 Resistors, in several values. It is good to identify these and sort them before starting work

– They have colored rings around them that indicate their value
– Since some of them look very alike, we've highlighted the key difference in telling the
– 330Ohm, 3k3Ohm resistors apart, as well as the 1K and the 10K resistors, marked **BOLD** below

– R1 : 2K2 (2200) Ohm	RED-RED-BLACK-BROWN-BROWN
– R2: 330 Ohm	ORANGE-ORANGE-BLACK-**BLACK**-BROWN
– R3: 680 Ohm	BLUE-GREY-BLACK-BLACK-BROWN
– R4: 1K (1000) Ohm	BROWN-BLACK-BLACK-**BROWN**-BROWN
– R5: 3.3K (3300) Ohm	ORANGE-ORANGE-BLACK-**BROWN**-BROWN
– R6: 10K (10.000) Ohm	BROWN-BLACK-BLACK-**RED**-BROWN
– R7 and R8: 470 Ohm	YELLOW-PURPLE-BLACK-BLACK-BROWN

- A set of SHORT and LONG wires (aka: Dupont Wires) with single-pin 'female' connectors
- A long strip of SINGLE ROW pins; this needs to be carefully broken into suitable lengths.

– K1+K2: Three pin headers, optional connector. Not required unless you are experimenting
– K3+K4: Three pin headers, optional. Only required if you would like to SPLIT the controller and have a seperate KEYBOARD
– P3: Five pin header for connecting signals from Arduino
– P4: 7pin header, used to solder DS1307 to
– P5: 4pin header, used for connecting the LCD

- Two shorter DOUBLE ROW pin strips; sometimes one of them needs to be shortened to fit.

– P1 + P2: 2x12 pin headers for making connections to switches, relays, etc.

Notes and Techniques

TOP and BOTTOM

One IMPORTANT thing to note with this PCB is that, unlike what you may be used to, the COPPER layer of the PCB is on the TOP. But SOLDERING is done on the BOTTOM, as is ‘normal’.

The TOP side is the side with the lettering on it.

SPLIT KEYBOARD BUILD

K3+K4 are optional and require a bit of explaining: You will perhaps have noticed the long horizontal line that runs from left to right and cuts right in between K3 and K4. This is to allow you to CUT the PCB into two parts at that line and to make it possible to have the keyboard seperate from the rest of the logic/circuit. This can be handy if you want to make it pretty with the buttons right below the LCD on the front of some control box. However, it will require some steady cutting of the PCB either with a (very) precise saw or by using a tool-knife and scoring the PCB deeply on both sides until you can SNAP the board into two parts. IF you would like to go down this road, please cut the board **BEFORE** soldering anything to it. It will make your life MUCH easier. Also, please be careful while doing it. PCB-material is tough; the human body typically is not... IF you have cut the PCB into two parts, you can use two 3-pin headers at position K3 and K4 to connect the two halves together again using long or short ‘dupont wires’. You can even solder K4 onto the ‘back’ of the PCB so they stick out into a convenient direction. If you do NOT want to go for the seperate keyboard, you can skip soldering K3 and K4

Soldering rows of PIN-headers

It’s not always easy to get a pin-header to be neatly soldered so that it sticks out STRAIGHT.

An approach that always works is to first solder only 1 pin of any row of multiple pins. Then, flipping the board over, you check if the row is ‘straight’. If it is, continue soldering the rest. If it is **not**, you can simply adjust the pins by re-heating the single soldering-joint while holding onto some of the OTHER pins (warning; small pins get hot **fast**). This approach only really works well when only one or two pins have been soldered yet. Re-adjusting a line of 12 pins after they’ve all been soldered is NOT a fun experience; so please assure yourself that things are in straight and ‘flush’ before continuing.

Another approach is to use a ‘guide’; like in the case of the Arduino (step 3) , you could use the holes of the IC1 footprint to stick the pins into and then settle the Arduino PCB ontop of it and then solder the pins in place on the top of the board. This will ensure that everything will fit perfectly when putting things together (as IC1 is exactly where the Arduino Pro Mini will be going!)

Steps of construction

1) The small parts always go first

When putting any electronic project together, it's best to always start with the most 'flat' parts first. In our case, this involves the 8 RESISTORS (R1-R8) , and the 6 SWITCHES (SW1-SW6)

In the pack-list above, you will find the color-codes with which to identify the different resistors used in this project. The names of the resitors (R1 – R8) correspond with the labels used on the PCB for each of them. You can use a multi-meter to double-check the values of the resistors before soldering them, if you have one available and you dont trust the color-codes alone.

To solder them, simply bend the wires until they can go through the holes and make sure that they're flat against the PCB. It is a good idea to bend the wires outwards on the UNDERSIDE of the PCB so that, when you turn the PCB over, they wont 'drop out'.

To save work, it's advised to first make sure that you have all the resistors identified; then insert them all into the proper places. Double-check. Then flip the board over and solder all the resistor-wires neatly in place.

When done; clip the wires nice and short; just above the solder-joint. Watch your eyes when clipping wires.

Next up are the six switches.
They are all the same; they only fit in one way as the pin-layout is slightly rectangular.
You will notice that there is a 5th hole next to each switch-position on the PCB. This hole is not used in the models of the switches used in this kit. You can safely ignore it.

The holes for the pins to go through are a bit tight; but they should fit with a little careful manipulation. Try and get the switches to sit as neatly 'flat' (flush) against the PCB as possible. They should 'snap' into the PCB and not fall out when you turn the board over.
When they're all in, solder them in place. No wire-trimming required here.

2) Pins, Headers and Connectors, Oh My!

The next set of things to solder into place are the PIN-headers, Screw-terminals and USB connector.

First a short explanation:
K1+K2 are OPTIONAL and are only really needed if you plan to not use the screw-terminals (K5/K6) for connecting the temperature-probes to. They can be safely left out but they dont hurt if you solder them in place and never end up using them, either.

K3+K4 are explained in the section about the SPLIT KEYBOARD BUILD at the beginning; we wont repeat that here.

It’s best to start with the pin-headers as they are the ‘lowest’ of all the parts we’ll be soldering into place in this section.
The pin-headers P1, P2, P3, P4 and P5 are all **required** and need to be soldered in place. The K1-K4 pins are discussed above. Decide if you would like to have them included too.

To do so, the best approach is to insert them into the PCB and then lay a flat object onto the top of the PCB to help them not falling out when flipping the PCB over onto the solder-side (bottom).
The sheet of paper you are reading right now can already be a great help!

It is important that the pins are soldered in STRAIGHT. This is not as easy as it may seem at first.

The ‘Notes and Techniques’ section at the beginning provides suggestions on how to ensure that the pin-headers will be in straight and neat. Please refer to it if you are unsure about your technique.

After the pin-headers come the Terminals K5 + K6. These are used to connect the temperature-sensors to. Simple insert them with the OPEN sides sticking OUTWARDS (left). Solder one pin of each; check if they are ‘flat’ against the board; solder the rest of the pins.

The USB-connector should be easy to fit in. It only fits in one way. Snap it into the board; solder one pin, check if it’s neat and then solder the rest. The two large holes require quite a bit of solder and heat to have it secured properly to the board. Be careful though; the outside of the connector gets quite hot and will stay that way for a while after.

Great; we’re almost there with the soldering. Two more things to go.

3) The Arduino Pro Mini (IC1)

First thing to do is to put the Arduino Pro Mini together.
When you take it from it's packaging, you will see that it has a TOP side (with the chip on it) and a blank BOTTOM side. It comes with two strips of STRAIGHT pins and one set of BENT pins.

The BENT pins go on the short side with 6 holes.
You will need to break the STRAIGHT pins into sections:
– Two sections of 12 pins (along the long edges)
– One section of 3 pins (for the pins close to the switch: GND, A6, A7. THESE ARE OPTIONAL and are only required if you plan to experiment with a lot of ANALOG sensors.
– One section of 2 pins for the two holes marked A4 and A5 on the UNDERSIDE of the board (these are IMPORTANT AND SHOULD NOT BE FORGOTTEN!

Putting it all together:
The two long strips of 12-pins should stick out DOWNWARDS, so soldering is done on the TOP of the board.
The BENT pins and the two short STRAIGHT sections of pins (3 + 2) should stick up UPWARDS and soldering should thus be done on the BOTTOM of the board.
In BOTH cases, the 'short' length of the pins should be what sticks through the Arduino. The LONG section is what should be 'sticking outwards'.

To make sure that the rows of pins will be soldered in straight and that everything will fit properly, later on, it is good to use a 'guide'. You can use the PCB itself as such a guide. Sticking the pins into the area marked 'IC1' and then putting the arduino onto will ensure that it'll fit perfectly, later.

The other approach outlined in 'NOTES AND TECHNIQUES' at the beginning of this guide also works, of course.

Great: you should now have an Arduino Pro Mini with 2 rows of 12 pins sticking out DOWNWARDS.

Now it is time to solder the 6 BENT pins and the 3 STRAIGHT pins at the other side. Use the same technique as before but solder on the BOTTOM of the board and make sure the pins stick out UPWARDS! . You can also solder the 3 Pins at A6/A7/GND if you so wish; but they are normally not required. Doing so will, of course, not interfere with anything later.

Now, to insert it onto the PCB correctly, make sure you have the side with the six BENT pins sticking out to the RIGHT side of the PCB. The three STRAIGHT pins sticking up should be on the LEFT side; close to where P3 is.

Flip the board over; solder one pin first; check alignment, re-adjust if needed; then solder the rest of the 2x12 pins so that the arduino is fastened in place securely.

Congrats; nearly there !

4) The RTC MODULE

You will notice the text ‘RTC MODULE’ written on the board on the left side of P4.

You guessed it, this is where the DS1307 Real Time Clock module goes (the thing with the battery holder).

Turn it so that the battery-holder is at the TOP and the row of 7 holes is on the RIGHT.

To secure it to the board; simply stick it onto the board so that the 7 pins of P4 stick right through the 7 holes of the RTC module; keeping the batteryclip on the TOP , towards you.

It’s good to make sure that it’s flush against the PCB; aligned as straight as possible.

When done, it is smart to insert the battery in case you forget to do so later.

You are now done with soldering. The rest of the connections are all done with screw-terminals or dupont-wires.

5) Arduino wires between A5/A6 and P3

Because not all Arduino Pro Mini’s are built the same and the position of some of the ‘extra’ pins change between models, we’ve chosen a flexible approach that will ensure that the required pins A4+A5 on the Arduino can always be connected to the PCB in some manner. We provide the connector P3 on the PCB to do this.

The only pins we require that are not already soldered to the PCB are Pin A4 and A5 .
With the PCB infront of you, A4 is the pin just next to the black chip on the arduino, on the bottom-right of it. Pin A5 is the one on the RIGHT of pin A4

These two pins provide access to the so-called ‘I2C bus’ on the arduino. We use these in our design to interact with both the DS1307 RTC module and the LCD-module. These connections are thus very important.

We connect these pins to P3 by using the SHORT DUPONT wires. You will need TWO of them.
Notice that below P3 there is a ‘1’ marked on the PCB; this is ‘pin 1’ of 5 pins in total.

Connect A4 (the left pin) to pin 4 of P3 Connect A5 (the right pin) to pin 3 of P3

6) Connect the LCD-module

Unpack the LCD module and make sure that the included jumper (little black plastic thing that comes in the package) doesnt get lost. Directly slide the jumper over the two pins labelled ‘LED’ to connect them together. This will enable the BACKLIGHT on the LCD.

Now, using the LONG DUPONT WIRES, connect the 4 bent pins on the LCD-module to the P5 of the PCB. Note that the numbering of P5 starts with the 1 on the RIGHT side of the connector.

P5 Pin 1: GND Pin 2: VCC Pin 3: SDA Pin 4: SCL

The easiest way to get it right is to simply connect the 4 Dupont wires to P5 and locate pin 1. Connect that one to GND on the LCD. Then simply connect the rest of the wires in a row; without mixing them up.

The most important thing is to NOT mix up GND and VCC here. This will break the LCD.

7) Connect a temperature sensor.

It is now time to hook up the ‘business end’ of the controller; the thing that actually monitors the temperature.

There are two screw-terminals (K5 and K6) on the PCB for easy connecting and disconnecting. If you use only one sensor, please connect that one to K5.

The tempature-sensor supplied with the kit comes with three wires; Black, Red, Yellow.
The BLACK and RED wires are for supplying power to the sensor (0 and 5 Volt), the YELLOW wire is the one that is used to communicate with the fermentor logic. It uses the so-called ‘1 wire’ protocol; hence only using one signal wire.

Both K5 and K6 have the same pinout. With the PCB infront of you and the connectors on the LEFT side of the board, the BOTTOM pin is pin 1!

K5/K6:	
Pin1: Signal	(Yellow)
Pin2: +5Volt	(Red)
Pin3: Ground/0 volt	(Black)

It is important to note that the RED one should ALWAYS be in the middle. As long as you make sure that this is the case, you cannot break the sensor by swapping around the other two wires. It will simply not work correctly, but nothing will end up broken beyond repair.

7) Hooking up relays, actuators, etc.

The dual-row pin-headers P1 and P2 are used together with Dupont-wires to hook up either extra sensors/switches/etc, or, more importantly, to control relays and such that , in turn, drive hot-plates, immersion heaters, drive fans, switch on Peltier coolers/heaters, etc.

The kit , deliberately, does not come with relays and such by default. The specific use that anybody might want to put the fermentor to may require rather specific circuitry to drive.

The software provided (see webpage) does have a number of options available, however, and we would like you to check the info provided there about what the advantages/disadvantages are of some of these approaches, and how you can go about hooking them up to the incubator.

Here is a short overview of some of the options that have been tried so far:

Solid State Relay

These devices are magic; they take a 5 volt signal and allow you to control a **much** higher voltage at a high current with relative ease and maximum safety. They come with 2 screws on one side (hook up to a GND and signal pin), and 2 screws on the other side that you use to 'insert' the relay into a power-line as if it were a normal switch. Prices vary between 5 and 50 euro depending on quality and current.

H-Bridge for Peltier

An H-Bridge is a device normally used for driving motors. It has one signal pins to control the 'direction' and another to control if the motor is spinning at all. It also has an input for the MOTOR voltage and an output connector for the motor itself. It uses a smart configuration of 4 powerful transistors and some control logic to control them.
It happens to be the case that a PELTIER ELEMENT is rather like a motor in the sense that you can drive it in any of two 'directions'. The element is a flat piece of material with two wires sticking out. Depending on how you apply the + and – voltage to the wires, the device will 'magically' transfer heat from one side to the other, or the other way around. This can thus be used to either COOL or HEAT ; depending on the 'direction'. Note that peltier-elements always benefit from having a FAN mounted on them; especially on the HOT side.

ATX-PSU driver

A rather smart and simple method to drive a powerful 12Volt device is to power it using an ATX power supply and then use a single part to control if the ATX power-supply should be on or off. This prevents having to use a high-current relay or transistor to control power to , for example, a 12-Volt heater, Peltier-element, lightbulb, hotplate, whatever.
This is achieved by using a single-transistor circuit that controls the so-called 'POWER-ON' signal in the ATX-connector (normally connected to the motherboard of a PC)

433Mhz transmitter

It should be quite possible to remote-control one of the commonly available remote-controlled power-sockets that one can by in DIY-shops nowadays. It should be noted that it's likely only 'easy' to do with the 433Mhz/833Mhz devices, as there are modules available on ebay to transmit and receive on these frequencies. This approach has so far ,however, not been used by anyone using the incubator; so no software support exists (yet). It is listed here as a convenience.