

Module Learning Kit

Contents:

- Duinotech Mega (XC4420)
- Solderless Breadboard
- USB Cable
- 4 Digit 7-Segment Display (Common Cathode)
- 1 Digit 7-Segment Display (Common Cathode)
- 8x8 Dot Matrix Display
- 6AA Battery Holder
- Assorted Resistors (8x 220, 5x 1k, 5x 10k)
- Assorted LEDs
- RGB LED Module – Common Anode
- 4 Tactile Switches
- 2 Piezo Sounders
- Assorted Jumpers
- 74HC595 (ZC4895)
- 50k potentiometer
- 2x Light dependant resistors
- Infra-red Photodiode
- Infra-red Receiver
- Analog Temperature Sensor
- 17 Button IR Remote Control
- 40 pin Header
- 2 Channel relay module, (Similar to XC4440)
- LCD display (2x16 characters)



WARNING – KEEP BUTTON BATTERIES OUT OF REACH OF CHILDREN

- If swallowed a lithium button battery may lead to serious or fatal injury in as little as 2 hours, due to chemical burns and potential perforation of the oesophagus
- All button batteries must be kept away from children regardless if they are new or used.
- If you suspect your child has swallowed or inserted a button battery immediately call the 24-hour Poisons Information Centre on 13 11 26 (Australia) or 0800 764 766 (New Zealand) for fast, expert advice.
- If your child is having difficulty breathing contact 000 (Australia) or 111 (New Zealand).
- Dispose of used button batteries immediately and safely out of the reach of children. A battery can still be dangerous even when it can no longer operate the device.
- As soon as you have finished using a button battery, put sticky tape around both sides of it. This will make it harder for children to swallow the button battery and avoid the risk of the battery catching fire.
- Immediately dispose of button batteries immediately out of reach of children, or recycle using a child resistant container and take used cells to your local battery collection centre.

How-tos and guide

There are a lot of tutorials online for most common components in this kit. A good resource is the Arduino.cc website and forums, as well as google; sometimes even a simple “ [item] datasheet” search would be enough to get you going. A quick starter for each of the components is found below but you are encouraged to experiment and do some research in what is possible with the components in this kit.

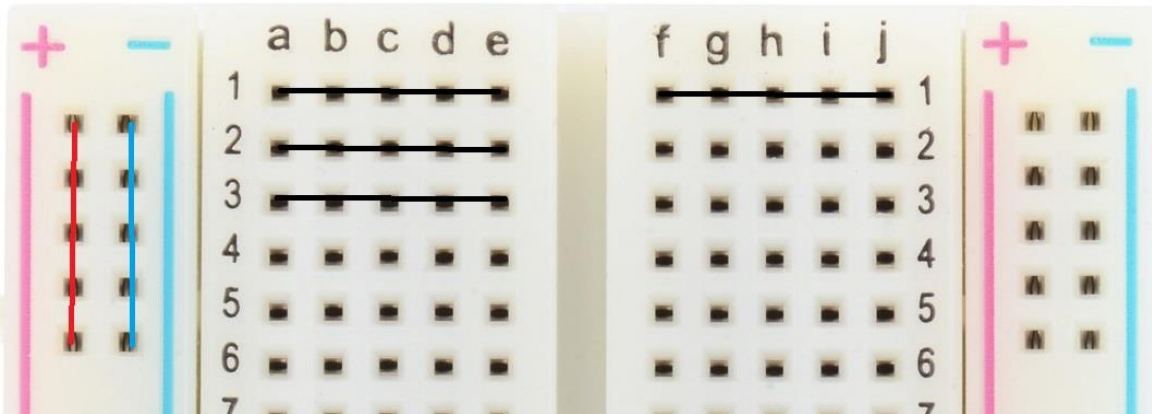
Duinotech Mega

The duinotech Mega (hereforth ‘Mega’) is a *microcontroller* device that is designed to control your electronic circuits. This is our version of the very popular Arduino set-up, and you will find most tutorials and modules plug right into the Arduino (or duinotech) boards. To start, you can download the Arduino program from <http://arduino.cc/> - some background into programming is recommended, or you can look into <http://S4A.cat/> which is the scratch, graphical programming language, which might be a little easier for young and old minds alike.

A good resource is "Practical Arduino" (BM7132 – \$54.95) for a good basis of understanding, or you could try "getting started with sketches" (BM7133 - \$24.95) to understand programming a "sketch," which is the main code file that runs on the Mega.

Solderless breadboard

Solderless breadboard is an easy way to test out circuits without soldering. Each row in the breadboard is connected to each hole, where leads and modules plug into.



See with the above picture, the red and blue "rails" down the side of the breadboard are connected in a vertical fashion, and each row (1, 2, 3) are connected a-e and f-j. (note that e and f are not connected.)

Anything connected on row 3 for instance, is connected to the letters on that side of the board.

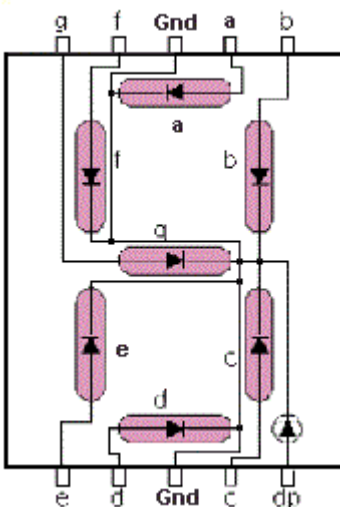
USB Cable

Used to program your Mega.

4 and 1 Digit 7-Segment displays.

Both of these work in a similar fashion. These are common cathode devices, which means the device shares a single ground connection, and each "bar" is lit up when the particular pin is powered.

Common Cathode

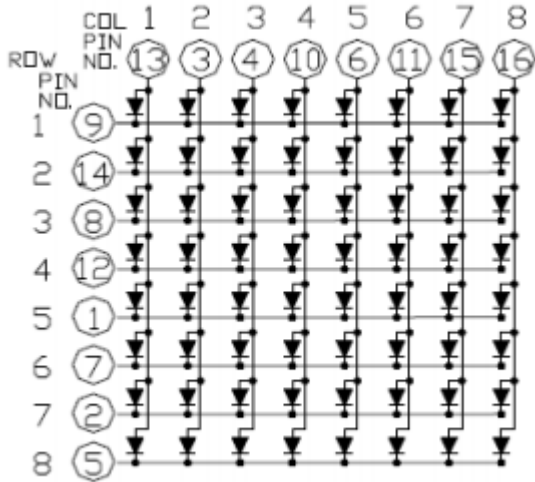


Consider for the picture on the left, if you wanted a "nine" to show up, you would connect Gnd to ground, then connect a,b,c,d,f,g to positive voltages. "dp" is also present as a decimal point indicator.

Be careful with the power, as current limited resistors should be used for each segment of the display. You should have 8, 220R resistors in your kit, so use that to connect to 5v supply.

The 4 digit display works similar, however there is a common cathode for each digit.

Most similar to [ZD1800] – the dot matrix display consists of a positive and negative for each light in the matrix.



With this set up, to turn on the top left light, you would connect positive to pin 13 and negative to pin 9. Again, as with all LEDs, be careful of the voltage and if you can, use a resistor (220R, only needed on one side of the led).

If you find the wrong square lighting up, or none at all, try turning around the module.

6AA battery holder

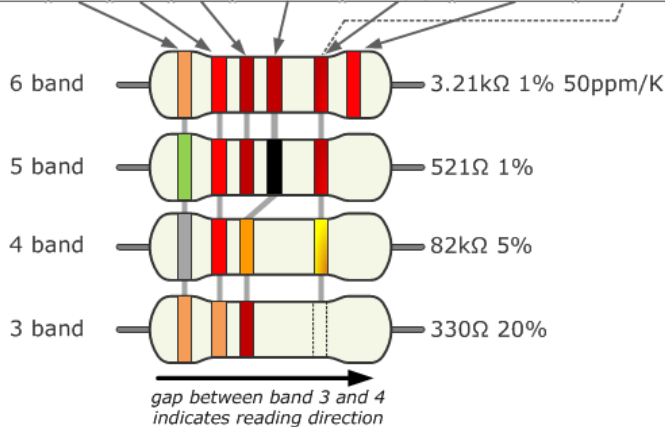
Holds 6xAA batteries. The supplied voltage is the sum of each of the battery voltages.

Assorted resistors

To find the correct size resistors, use the following chart. Usually there are 2-3 numbers and a multiplier; for instance, if the colour code was “yellow violet orange” then the value of the resistor would be 47k.

www.resistorguide.com

	Color	Significant figures	Multiply	Tolerance (%)	Temp. Coeff. (ppm/K)	Fail Rate (%)
Bad	black	0 0 0	x 1		250 (U)	
Beer	brown	1 1 1	x 10	1 (F)	100 (S)	1
Rots	red	2 2 2	x 100	2 (G)	50 (R)	0.1
Our	orange	3 3 3	x 1K		15 (P)	0.01
Young	yellow	4 4 4	x 10K		25 (Q)	0.001
Guts	green	5 5 5	x 100K	0.5 (D)	20 (Z)	
But	blue	6 6 6	x 1M	0.25 (C)	10 (Z)	
Vodka	violet	7 7 7	x 10M	0.1 (B)	5 (M)	
Goes	grey	8 8 8	x 100M	0.05 (A)	1(K)	
Well	white	9 9 9	x 1G			
Get	gold		x 0.1	5 (J)		
Some	silver		x 0.01	10 (K)		
Now!	none			20 (M)		



Assorted LEDs

Each coloured LED uses a different amount of current to power them; for ease though, you can use the provided 220ohm resistors on each of them safely.

Providing too high of a resistor would limit the current to the LED, providing a dim light but will not blow the LEDs. Providing too low of a resistance will flood the LED with current and burn it out. In our catalogue (why haven't you got one on your desk?) there is a handy chart to find the correct resistor for common voltage values, or you could look for an "LED resistor calculator" online to find the correct resistor for maximum power output.

RGB LED Module Common Anode

Similar theory with the 7 segment displays. The RGB module requires 1 common connection (which is "anode" this time) connected to positive voltage, and each other pin is then grounded to activate the Red, Green, and Blue LED lights. Grounding both Red and Blue would produce a violet colour, and so forth.

Tactile Switches

Switches and buttons are used to get input from the user, such as when to start/stop something, or to activate certain items. To connect a button, you must use what's called a "tie up" or "tie down" resistor of a high value (shown here is a tie down resistor of 10k.)

This resistor settles the pin to a steady voltage so that it is not "floating." Further information and descriptions of what these terms mean are left as an exercise for the reader but are irrelevant to the functioning of a circuit. Simply connect the resistor from the designated pin of your choice, going to ground, and then put one side of the button (or switch) in the pin, and the other side to a positive voltage.

When looking for a button in Arduino, declare the pin as an input and then wait until the button has been pressed to run your code.

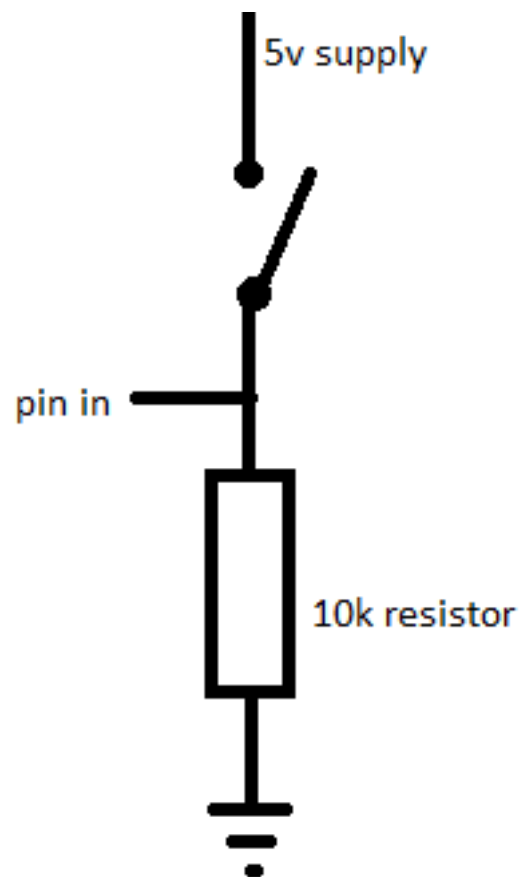
For a more detailed run down, check out <https://www.arduino.cc/en/Tutorial/Button>

Piezo Sounders

These are simple buzzers that are used to create sounds. You must send a *square wave* to the buzzer to activate it, as you will only hear one "tick" if you connect it straight to positive and negative. Check out the Arduino example called "toneMelody" in the 02.Digital folder of the inbuilt examples. This requires a buzzer connected between pin 8 and ground.

74HC595

This is an 8-bit shift register. Remember the 7-Segment digit? This would allow you to run the entire thing off 3-4 wires, instead of using all 8 or 9. The easiest way to learn about this would be to look at



the datasheet, which is both on our website or you can google “74HC595 datasheet” and find one; different manufacturers all follow the same standard and pin-out of what the chip should be so that people can use them in their devices.

This has two main lines of interest, you must “clock” a signal (take the voltage high then low) to latch in whatever is on the serial line (a high or a low) – which is recorded onto the parallel port. If you were to clock a signal as “high, low, low, high, high” it will show up on the parallel port as 10011000.

If you haven’t got your head around that yet, don’t worry, because the Mega has quite a lot of data lines on it and you shouldn’t have to worry about saving too much space for now.

50k potentiometer

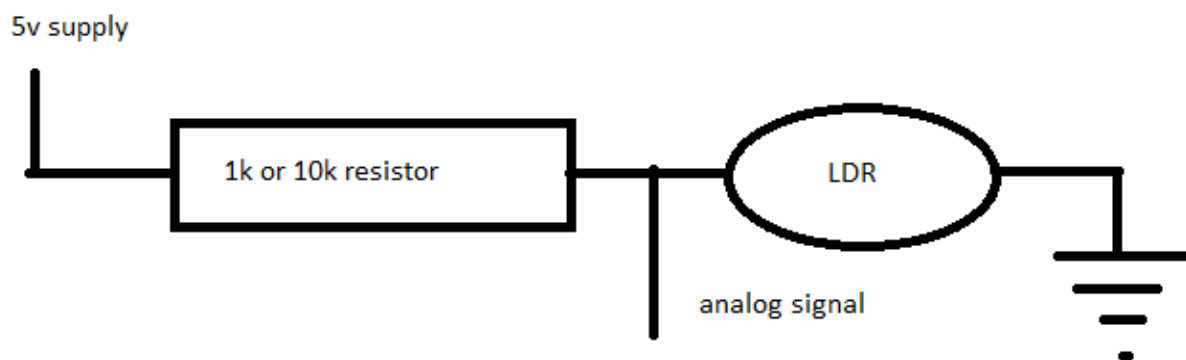
The potentiometer can best be thought of as a 50k resistor, but with a “window wiper” that slides across it.

When the potentiometer is turned all the way to the left, there will be no resistance between the left and middle pins, with 50k resistance between them and the right pin. At mid-way, you should read 25k resistance between the left/middle and right/middle pins. Then when it is to the right, there should be no resistance between the right and middle pins.

You can use this to set an adjustment setting, such as how slow you want an LED to blink, in your Arduino code. Place the resistor’s middle pin to an analogue in pin, such as A0. For more information, check out <https://www.arduino.cc/en/tutorial/potentiometer>

Light dependant resistors

Light dependant resistors work similar to a normal resistor, however the value changes depending on the light. Try them out and show the results in the Serial Monitor of the Arduino IDE. Use a 1k resistor from one side of them, so that they can behave like a potentiometer.



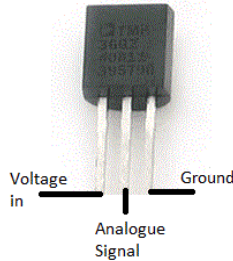
Infra-red Photodiode

The Infra-red photodiode works just like a normal LED, however it’s infrared, which is invisible to the human eye. Try lighting it up then looking at it through your phone camera, and you should easily see the effects.

Infra-red Receiver

The receiver gives an analogue signal of the IR light that it senses. This could be used in conjunction with your photodiode as a way to communicate wirelessly to another Arduino. OR you could use your provided IR remote to allow for 17 buttons through one analogue pin.

This sensor is very easy to use, just use it as a potentiometer above. Make sure you have the polarity right however, as it will not work in any other way. When looking at the flat side, the leftmost is your positive, and the rightmost is the ground.



2 Channel Relay module (similar to XC4440)

Relays are useful devices that act like “electrically activated switches” – They are used in cases where you would want to isolate the “control voltage” (coming from your mega) to another circuit, say to a car engine or similar.

To activate the relays, simply provide power to the module (VCC= positive, GND = negative/ground) and send a voltage to IN1 and 2 for the first and second relays.

The relays themselves have 3 connections; NC, NO, and COM. This stands for Normally Closed, Normally Open, and Common, which when you activate the relay, the Normally Closed is opened, and the Normally Open is closed, with respect to common.

If you look at the diagram printed on the circuit board, it should begin to make sense.

LCD display

The LCD display is the last module in this kit and is one of the more interesting and somewhat challenging ones. we are going to be connecting it in a 4 bus connection. Aside from providing power to the module, you must use 7 data lines to control different aspects of the device. These are labelled RS, RW, E, and DB4-DB7.

Once you connect these to different pins on the Mega, use the LCD library (or see the LCD example in the Arduino IDE) and specify which pins you have used. Arduino has hidden a lot of the hard work that needs to be done, so it should be relatively easy to set up.

Pin Functions

Signal	No. of Lines	I/O	Device Interfaced with	Function
RS	1	I	MPU	Selects registers. 0: Instruction register (for write) Busy flag: address counter (for read) 1: Data register (for write and read)
R/W	1	I	MPU	Selects read or write. 0: Write 1: Read
E	1	I	MPU	Starts data read/write.
DB4 to DB7	4	I/O	MPU	Four high order bidirectional tristate data bus pins. Used for data transfer and receive between the MPU and the HD44780U. DB7 can be used as a busy flag.
DB0 to DB3	4	I/O	MPU	Four low order bidirectional tristate data bus pins. Used for data transfer and receive between the MPU and the HD44780U. These pins are not used during 4-bit operation.

Distributed by:
Electus Distribution Pty. Ltd.
320 Victoria Rd,
Rydalmere NSW 2116 Australia
Ph: 1300 738 555
Int'l: +61 2 8832 3200
Fax: 1300 738 500
www.electusdistribution.com.au