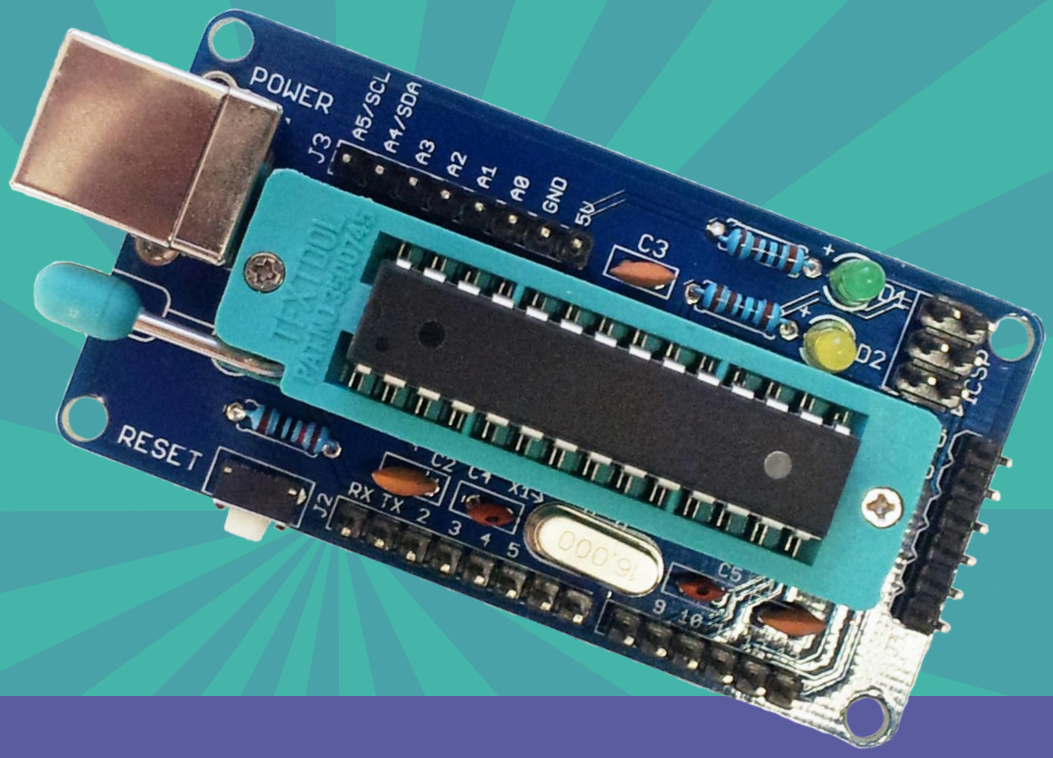
D.I.Y 8BIT ARDUINO MICROCONTROLLER



Build your own 8-bit Microcontroller Board.

- Learn the fundamentals of a microcontroller.
- Learn basic through-hole soldering techniques.
- Comes with FTDI Interface Connector.

ABRA
ÉLECTRONIQUES • ELECTRONICS CORP.

Table of Contents

1.0 Component Checklist

Please ensure you have the following components included in your kit.

Component	Quantity	Remarks	PCB Reference number
10kΩ Resistor	1	Brown, Black, Black, Red, Brown	R1
1kΩ Resistor	2	Brown, Brown, Black, Black, Brown	R2, R3
16Mhz Crystal	1	External Oscillator	X1
22pF Capacitors	2	Ceramic Based	C4, C5
100nF Capacitors	3	Ceramic Based	C1, C2, C3
Push Button	1	N/A	K1
Green LED	1	Power Indicator, Anode towards R2	D1
Yellow LED	1	Status Indicator, Anode towards R3	D2
6 pin Header	2	1 Row, 6 Columns	J1, J4
8 pin Header	2	1 Row, 8 Columns	J2, J3
3 pin Header	1	2 Rows, 3 Columns	ICSP
USB – B	1	Female USB B Socket	USB
Locking Seat	1	N/A	ZX1

2.0 Soldering Through-hole Components

Solder is composed of two or more metals that have a lower melting point compared to standard metallic substance. They are found commonly in spools or tubes and the solder is conductive (generates electricity), we can classify them as conductive glue for the moment. Solder is also a verb in which we can use to solder something together to create a solder joint.

There are two types of solder, leaded solder and lead-free solder. Leaded solder contains lead and tin, the ratio of the substance is read as the following: 60/40 denoting that it contains 60% tin, 40% lead. In some cases the product is advertise with rosin core, this material is embedded within the solder and acts as a cleaner to clean surfaces. For electronics we use Rosin based core and not Acid based core, acid is for plumbing not electronics as it could damage the connection. Rosin and acid core cleaners are called flux. Flux helps clean the surface and if the contact point isn't too dirty it will make a very strong solder joint.

Lead-free solder contains 99% tin and may contain both silver and copper or solely copper. The difference between the two is that lead-free solder melting point is at 227°C (440.6°F) and leaded solder melting point is at 188°C (370°F) or 63/37 at 183°C (361°F). These are the most common solders, there are many more variants for different applications.

You would need a lighter to melt the glue from solid form to liquid form, the only issue is using a lighter you would have no accuracy of melting the glue and it could pour everywhere. A soldering iron solves this issue, there are two types of soldering irons. Those with a base and those without a base. Soldering irons that come with a base have temperature control and in some cases removable connectors. Soldering irons without a base cannot be temperature controlled, they are often sold as 25W – 40W. They usually will hover around 400°C (752°F), not very good if you're starting to learn soldering. Some soldering irons come with replaceable tips, tips are useful for different applications of soldering. To change a tip, ensure the soldering iron is powered off and cooled down. Twist the knurled nut counter clockwise and remove the sheath, you can now change the tip and refasten the sheath. Always position the soldering iron to the side where you soldering hand is, so you don't have the wire running across you. To hold the soldering iron when not in use, you use the cradle. A device that holds the iron for you, some cradles come with a sponge holder to clean your tip.

The three most common tips are Bevel tips (Hoof Tips), Chisel Tips and Conical Tips.
Bevel are used for holding large amount of solder on the tip for small-gauge wires or dragging across surface-mount chips to solder multiple pins at once. Chisel are used for evenly delivering heat to the component and pad. Great for wires, through-hole components, large surface-mounts components and desoldering. Conical are used for precision soldering and are very common. The pointy tip helps deliver heat to small areas like tiny surface-mount components.

Mildly wet Sponges are helpful for removing excess solder from the iron tip, however over time they can damage the tip from the contraction of high to low temperature. Brass sponges do not need to be wet and are abrasive to trap excess solder when it is not required, if you can't find a brass sponge please use a mildly wet sponge, not a soaking wet sponge.

Solder wick is composed of stranded wires to remove excess solder on through hole components. Place the wick next to the component and heat with the iron, watch as the wick absorbs the solder. Remove the wick and cut the silver side, leaving the copper colored wick for future use.

Tip tinner is used to clean the tip of your soldering iron. Useful for removing those accidental plastic residue, it also prevents oxidation that accumulate on your soldering tip when not in use.

Soldering vacuum pump are great for removing solder left behind in the through-holes when desoldering components.

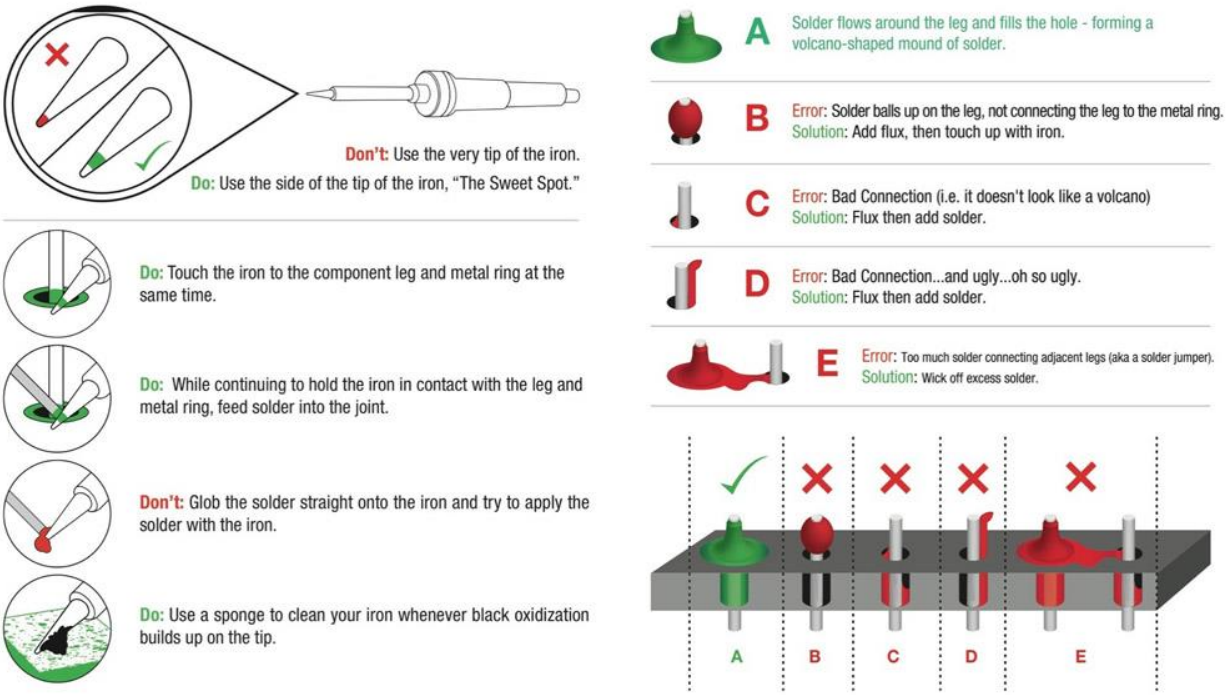
Third hand or helping hands are used to hold the board in place when you solder.

Flush cutters are used to trim the legs of components.

It is recommend to tin the tip of the soldering iron before and after use. Turn on the soldering station or plug in the iron, after a few minutes take the iron and place it over the moist sponge and do a wiping motion for both side of the tips, you should be trying to remove the old solder that was left over. Do not leave the soldering iron in the sponge it will damage the tip over time. Take some solder and place some over the iron to make a nice coating, use this coating to solder components to the pads and add more solder if needed. When placing or leaving the soldering iron alone for a long duration clean the iron and place a coating of solder on it for future use, it will help protect the tip.

When heating the soldering iron a good indicator that the heat is too high is if the solder when applied to the iron starts forming vaper, you should lower the heat. Recommend heat ranges from (325 – 375°C) or (617 – 707°F).

Your solder joint should look like the following:



(Source: Sparkfun)

3.0 Component Orientation

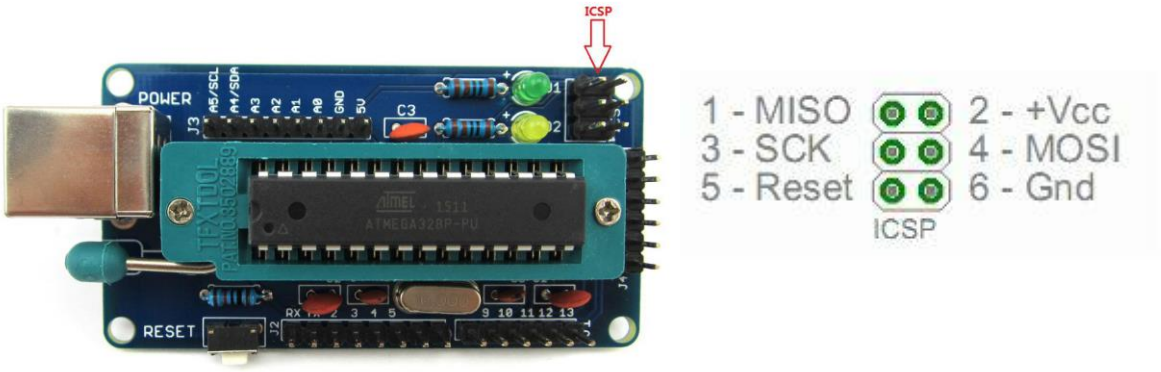
When soldering any board or kit, you must start with low overhead connections and then build up to high overhead connections.

Solder in order: R1, C2, C4, X1, C5, C1 and K1. Then solder D1, D2, R2, R3, C4 and J3.
Proceed in soldering J4, J1 and J2. For the large overhead components solder ZX1 and USB.
To find out what the component reference numbers refer to please look at 1.0 component check list.

4.0 Loading the Bootloader

Refer to this section if in case your bootloader cause issues or you changed the IC for a new empty ATmega328P-PU IC. You should have a preloaded version from the abra kit.

- Download and install the Arduino IDE
- Borrow or use your Arduino Uno R3 to burn the bootloader to the DIY Controller Board.
- Ensure the ATmega328P is fastened to the locking seat and the handle is positioned downwards.
- Connect using a female to male header the ICSP pins to the Arduino Uno.



ICSP	Arduino UNO R3
1 - MISO	PIN 12
2 - VCC	PIN 5v
3 - SCK	PIN 13
4 - MOSI	PIN 11
5 - RESET	PIN 10
6 - GND	PIN GND

- Connect the USB cable to the Arduino Uno and open the Arduino IDE. Click Tools > Board > Arduino Uno. Also select the port in which it is connected to under Tools > Port, in my case COM25.
- Now under Tools> Programmer change from “AVRISP MKll” to “Arduino as ISP”.
- Right click on the open or upload button and select “ArduinoISP”. Now click the compile and upload sketch button.
- Now we are done go to Tools > Board and ensure that Arduino Uno is selected.
- Go to Tools > Burn Bootloader and wait for the DIY board to be loaded with the bootloader, it will say “Done burning bootloader”.
- As the final step go to Tools > Programmer and set as “AVRISP MKll”.

5.0 Connecting FTDI

If you look at the Arduino Microcontroller on the opposite side of the USB port you should notice a header out of place compared to a standard Arduino Uno. If you flip this board you will see a label calling this header a program. You will connect your FTDI to this header for future programming.

Arduino Uno does not have this Header exposed because it is connected to another microprocessor (ATmega16U2-MU) that does the communication for you when you plug in your USB cable. This chip will translate USB protocol from the computer to Serial TTL Communication format. Old standard Serial communication formats where used with printer ports or the odd looking vga like ports, they sent information very slowly at high voltage ranging from -13v to +13v to increase the distance of the cable using the rs-232 protocol.

The old serial ports transfer data slowly, whereas the new USB ports could do this job very quickly. The issue is that the USB uses a protocol that send information like packets rather than the serial port stating the incoming line as high or low.

To fix this issue they developed a USB to Serial converter chip that would do the signal level shifting and manage communication protocols required to communicate the pc to a serial device.

If you find a serial port on your computer, you could purchase a serial to USB cable adapter but the issue is then the voltage is really high and can damage the microcontroller. Microcontrollers like to work from 3.3v – 5, to fix this issue we use something called MAX232 chips. It would convert 26v line to a 5v line.

We are using FTDI device based on the level shifting logic, connect the following pins to the Arduino Microcontroller.

Program Header	FTDI
DTR – Data Terminal Ready	DTR
TX - Transmit	RX
RX – Receive	TX
5V – Do not plug in USB if plugging in 5V	5V
GND	CTS – Clear to Send
GND	GND

- FTDI based on ARD-FTDI @ abra-electronics.com

Connect the USB B mini to the FTDI Chip and USB A to the Computer. You should see the device automatically detected and assigned a COM port # that can be found in your device manager.

6.0 Programming

To program the Arduino Microcontroller, open the Arduino IDE. If you do not have the Arduino ide please download it from arduino.cc

Please go to Tools > Board and select “Arduino Uno”.
Please go to Tools > Port and select “the com port that is assigned to your device by your O/S”
Type in or open an example code like you would with a normal Arduino.
Click upload and wait for the program to load onto the Arduino Microcontroller.
It will take a few seconds but should result in “Done Uploading”.

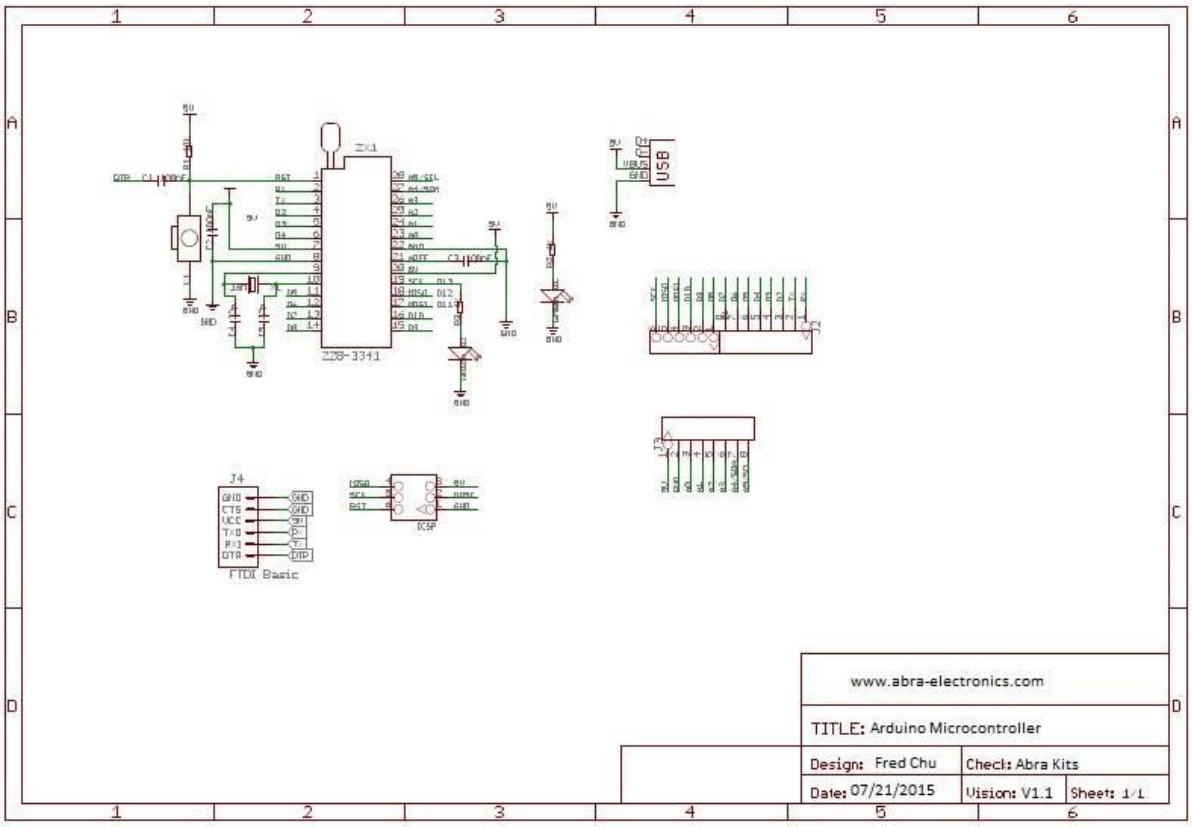
If it has failed please check the error message. If the device states “busy” disconnect the device and remove the CTS line, reconnect and attempt to load code.
If it says com port error message please go to Tools > Port and select the correct COM port.
If in device manager you do not see the COM port please go to:
http://www.ftdichip.com/Drivers/VCP.htm and download the appropriate driver.

7.0 Schematic diagram and Circuit analysis

Crystal: The crystal is used for time based operations, it will generate a frequency or pulse time that the microprocessor will understand and convert into timer value. Using some mathematics and some formula you can make the microprocessor which communicates in a very fast time slow down to 1 second for a single task. Some codes like the led blinking for 1 second would not normally happen with a very high timescale. The timescale from this mathematics and signal reading/scaling will give the function to blink every one second. The microprocessor does come with an internal oscillator but to turn that on you must custom program it. If you look at section

4.0 it is the same processor however we must change the code from Arduino to turn some fuses within the microprocessor that will activate the internal clock, not recommend for the time being. It is best to attempt this using a PIC based processor and later move on to the Atmel based processors.

Capacitors: The capacitors are used to smoothen voltage for a very clean output. They are also configured as decoupling capacitors. Decoupling capacitors are used to eliminate noise from one section of the circuit to the other. So you can see the oscillator is isolated from the entire circuit so the microprocessor only receives the oscillator signal at those pins.

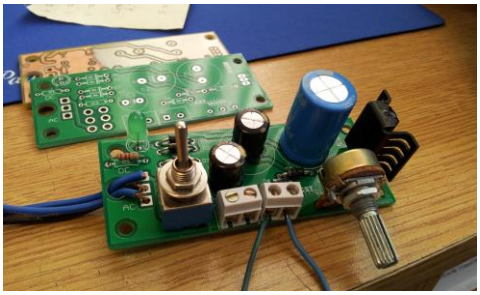


8.0 Feedback

Noticed some grammatical errors, theory that don't make sense or diagrams incorrectly drawn? Need further explanation or reduced explanation on certain aspects? Got ideas to improvement on certain kits? Want to leave feedback about our services or suggest ideas that would make great kits? Well then simply e-mail us at kits@abra-electronics.com, for general inquiries about product and pricing please contact our store at 1-800-361-5237 from Monday to Friday from 9am to 4pm.

9.0 Related Products

Looking for similar related products, here is what we have made so far.



Linear Power Supply Kit 0 -> ±18 VDC, 18 VAC, 1A

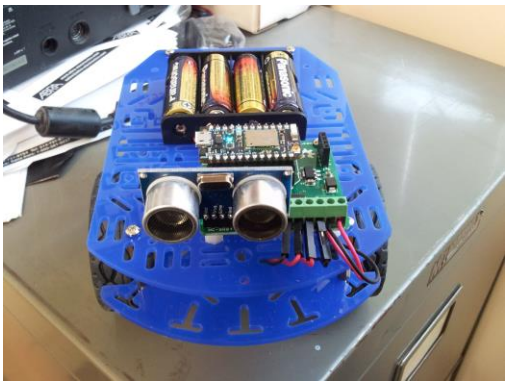
Great kit for starters as an essential lab bench tool, learn the basics about electricity AC vs DC, bridge rectifications and basic soldering skills.

Code: AK-10 (safer than the AK-47)

Tracking Robot Car Kit

Super simple to assemble kit comes with all necessary parts and instruction manual get your very own low powered line tracking robot. Good for experienced beginner level soldering students and beginner robotics students.

Code: KIT-ROBOCAR (not Robocop sadly)



PHOTON 2WD Robot Instructional Pack (includes Photon)

IOT Internet of things device of the year with the super simple over the internet installation you can have your very own computer wireless controlled robot that only uses four AA batteries! Make sure you download the 2WD Photon Driver and installation guide located on our webpage.

Code: PHOTON-2WD-KIT (comes with a photon)

We even have a Raspberry pi 2 version that teaches SSH and

Python Code: PI-2WD-KIT (comes with a raspberry pi 2)