**FAQ**

**Basics**

1) *How do I know what end of the LED/capacitor is '****+****' and which is '****-****'?*

Look at which side is not like the other, and that would be the ground. I.e for a capacitor there is a blue strip which points to the ground.

2) *What is the front side of the voltage regulator, and where is the input?*

The front side is where the circle is closer to you & the input is on the right, at that view.

3) *What is the difference between analog and digital signals?*

The difference between analog and digital technologies is that in analog technology, information is translated into electric pulses of varying amplitude. In digital technology, translation of information is into binary format (zero or one) where each bit is representative of two distinct amplitudes.

<http://www.diffen.com/difference/Analog_vs_Digital>

4) *What does VDD and VSS mean?*

VDD, VCC, V+: The positive power supply, from a battery or other source; wire it **RED**.

VSS: Ground, GND, or the negative supply; wire it **BLACK**. So, VDD is Input, VSS is ground (trick for VSS: think of South)

5) *What does TRIS, ANSEL & PORT mean*?

TRIS is to set ports as input/output

ANSEL is to turn on/off analog ports

PORT is to set/clear ports

usage:

#define LED PORTAbits.RA4

TRISAbits.TRISA2 = 1;

6) *How do you upload your code to the microcontroller?*

-**build**, then **program** https://lh4.googleusercontent.com/yaf7WuUUe7TSFKJ7POu8f67r2zuUkvmq2Vr5g3tbUKJBBUaOJ8R28bk3fGzMlkzLmssFeNFRQsxnykl5x5dWdQJ7eX7xnSJP_5GXUNITZIcXNOjS_vxqp2TBl8gooxFFAw(or look for this button in MPLABX)

7) *Does having different 'grounds' matter?*

Yes!

8) *How do you connect a Pickit3 to a microcontroller?*

PIC16F88 Microcontroller: PP 2-4 <http://ww1.microchip.com/downloads/en/DeviceDoc/30487D.pdf>

Pickit3 Programmer: PP 2-7 [http://ww1.microchip.com/downloads/en/DeviceDoc/52116A.pdf pg.15](http://ww1.microchip.com/downloads/en/DeviceDoc/52116A.pdf%20pg.15)

9) *What is an ADC?*

Analog digital converter

10) *How do I program in C?*

<http://www.learn-c.org/>

or

<http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-087-practical-programming-in-c-january-iap-2010/>

11) *Useful Websites*

<http://www.digikey.ca/?WT.srch=1&WT.medium=cpc&WT.mc_id=IQ65223341-VQ2-g-VQ6-49021330985-VQ15-1t1-VQ16-c>

<http://www.microchip.com/>

<http://www.google.com/shopping>

<http://www.engr.sjsu.edu/bjfurman/courses/ME106/mechatronicstutorials.htm>

<http://www.engineeringtoolbox.com/>

<https://learn.sparkfun.com/tutorials>

**Programing/C stuff**

12) *What is in your ‘header’?*

-Your prototypes of your functions

-.h

13) *What is in your ‘source code’?*

-Your functions

-.c

14) *What is in the ‘library’?*

-The library is the output from your header and source code (which has been compiled). The header and source code is information in binary, while the library is information that is human readable(actual words)

-.lpp

Therefore, h + c = lpp, where “=” is the microcontroller compiling.

15) *How to make a library?*

-create a source code

-go to cmd

-type: dir & cd multiple times until you find the path to your source code file(dir to search directory; cd to change folder location)

-then type a modified version of this into cmd: xc8 source.c --chip=MYCHIP --output=lpp

16)*What are the different brackets in your ‘include’ section?*

-”stuff”, is a local file, inside your folder; in other words, you created the file ‘stuff’

-<stuff>, is a global file, inside any other folders; in other words, someone else created the file ‘stuff’

17) *What is a prototype?*

-It’s the first line of each function, usually like :

‘void name(void)’

or

‘int name(int stuff)’

18) *C tips*:

%d = integer (placeholder)

\n = new line

& = address-of ...  gets the address of the variable

\* = gets the value of a pointer

19) ADC:  voltage → bits → integer(distance)

**Motors/H-bridge**

20) *The project is not building?*

-double check if all the wires are in the right position

-Always, ensure that VCC2 is disconnected before programming the microcontroller; otherwise it won’t build. (And then put VCC2 wire back into position.)

-disconnect PIC3 from computer and reconnect

21)

EN = enabled = allows the motors to run or not, through impedance

A = input to output(Y)

Y = input to the motor

A--->>----Y--->>---motor

A will be the same value as Y

1A & 2A need to be different values to put the motors in motion (i.e 1A = 0, 2A = 1; therefore, 1Y = 0, 2Y = 1)

1,2 is one pair for one motor; 3,4 is another pair for another motor

Vcc1 = input to circuitry (to power microcontroller)

Vcc2 = input to circuitry ( doesn’t power microcontroller, but powers the motors, with the voltage going to the Y terminal)

Y voltage = Voltage of (=Vcc1 + Vcc2)

<http://www.ti.com/lit/ds/symlink/l293d.pdf>

22) *Datasheets*:

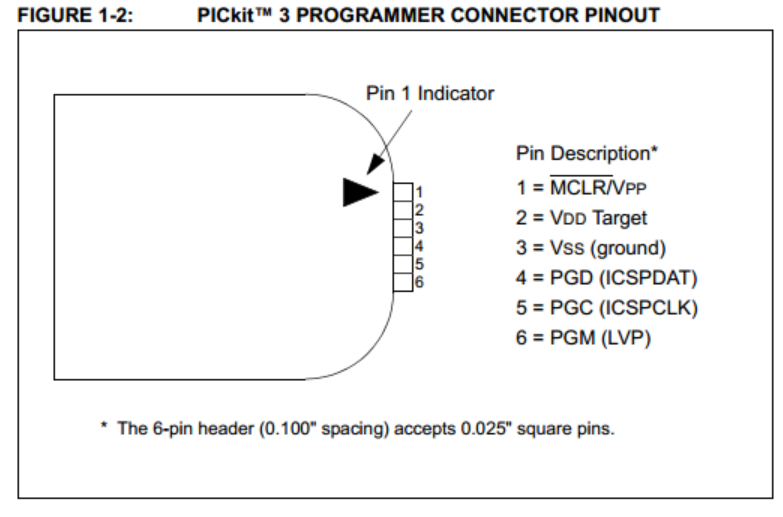
http://www.ti.com/lit/ds/symlink/l293d.pdf

http://ww1.microchip.com/downloads/en/DeviceDoc/30487D.pdf

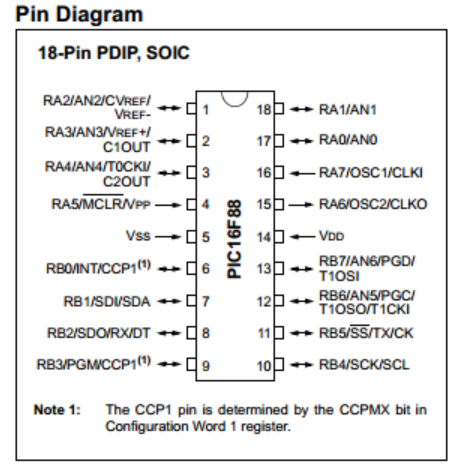
<http://ww1.microchip.com/downloads/en/DeviceDoc/52116A.pdf>

<http://www.fairchildsemi.com/ds/LM/LM7805.pdf>

PICkit3 → microcontroller



microcontroller:



H-bridge:

