

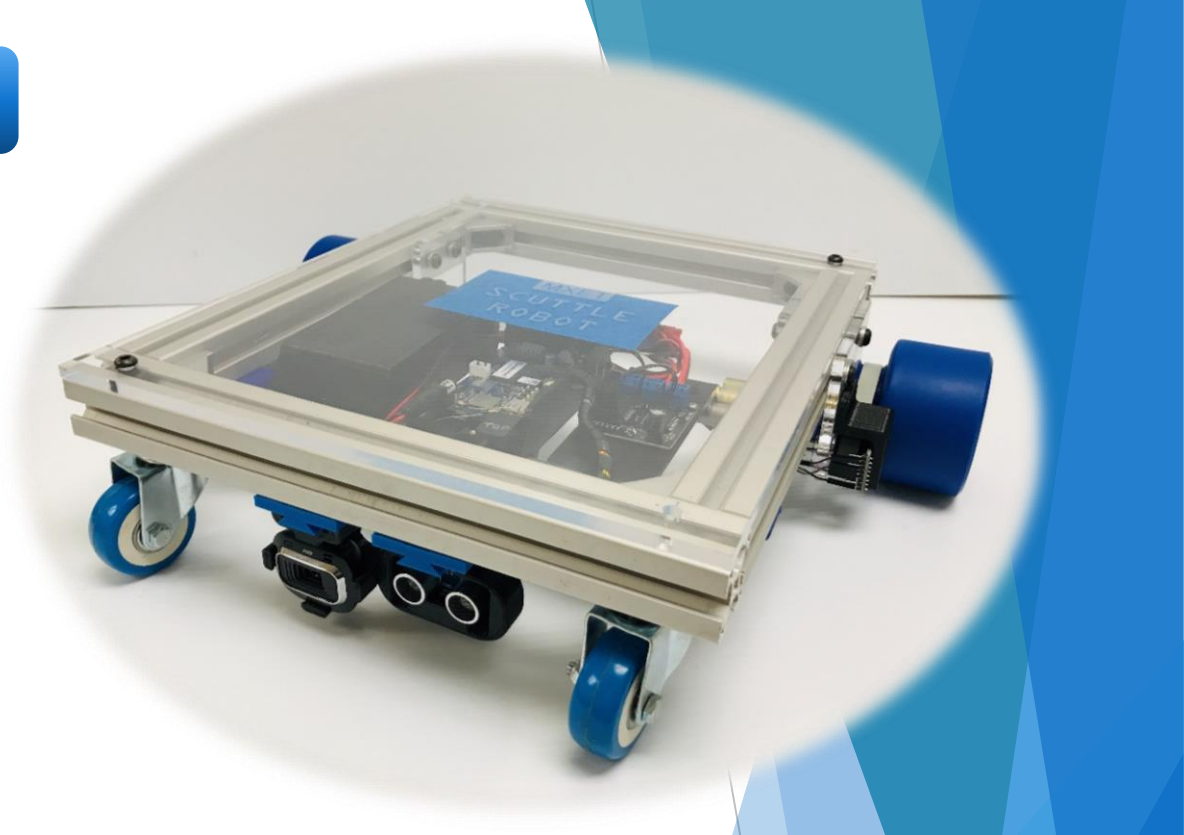
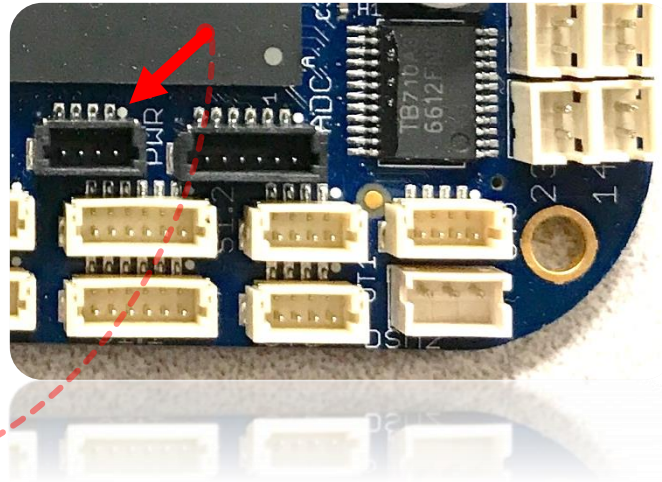
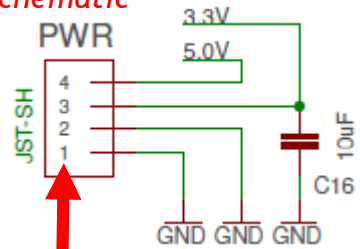
# Scuttle robot Wiring Guide (rev 2019.11.20)

## Important Info:

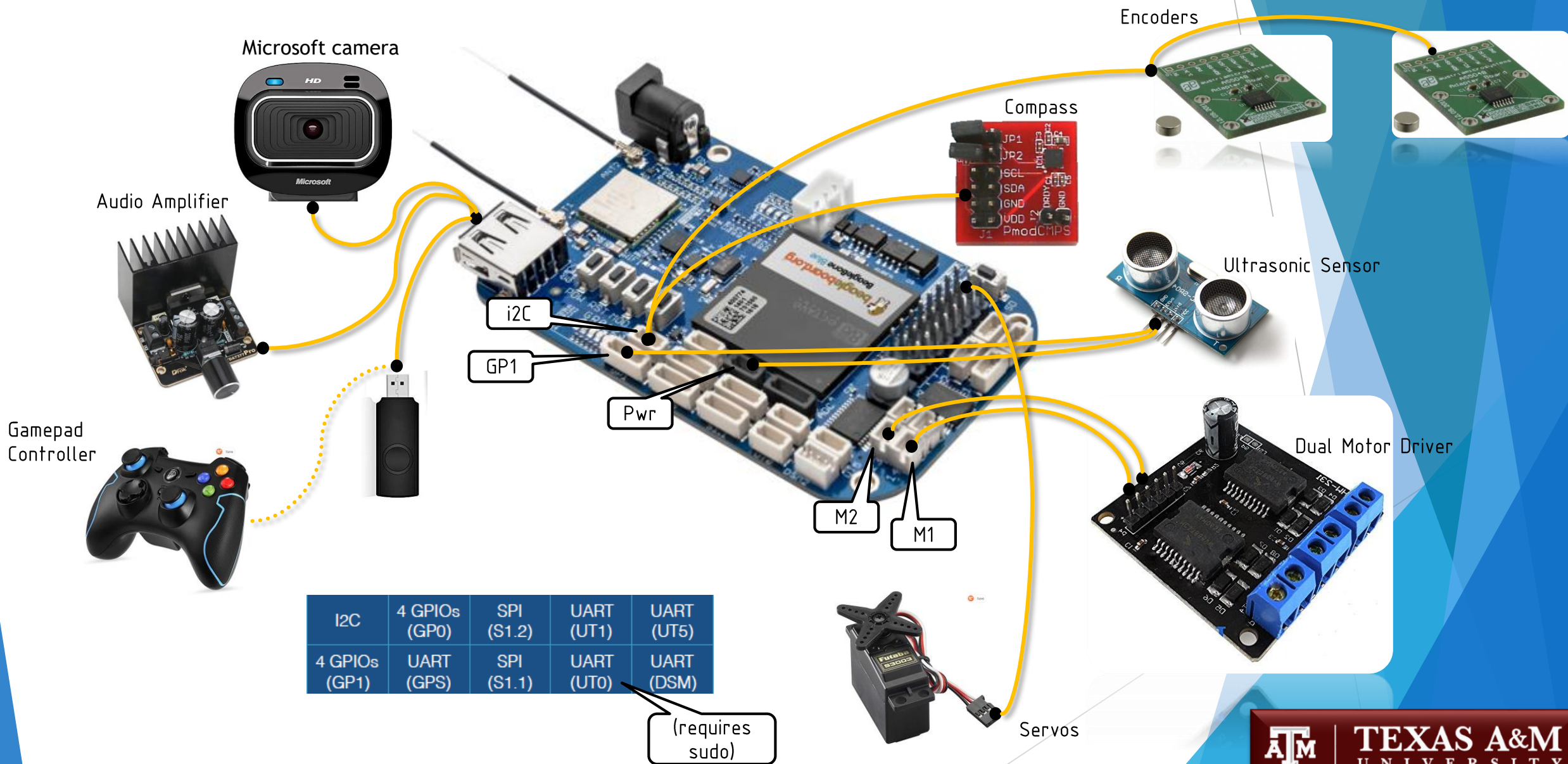
To match the beaglebone pins to the pin numbers on the diagram:

The tiny white circle on the silkscreen at each connector indicates “pin1”

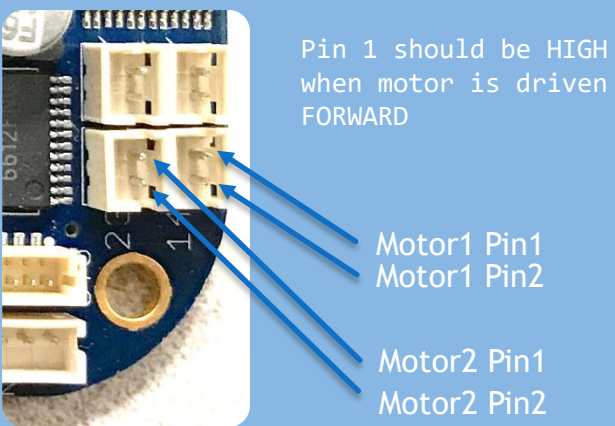
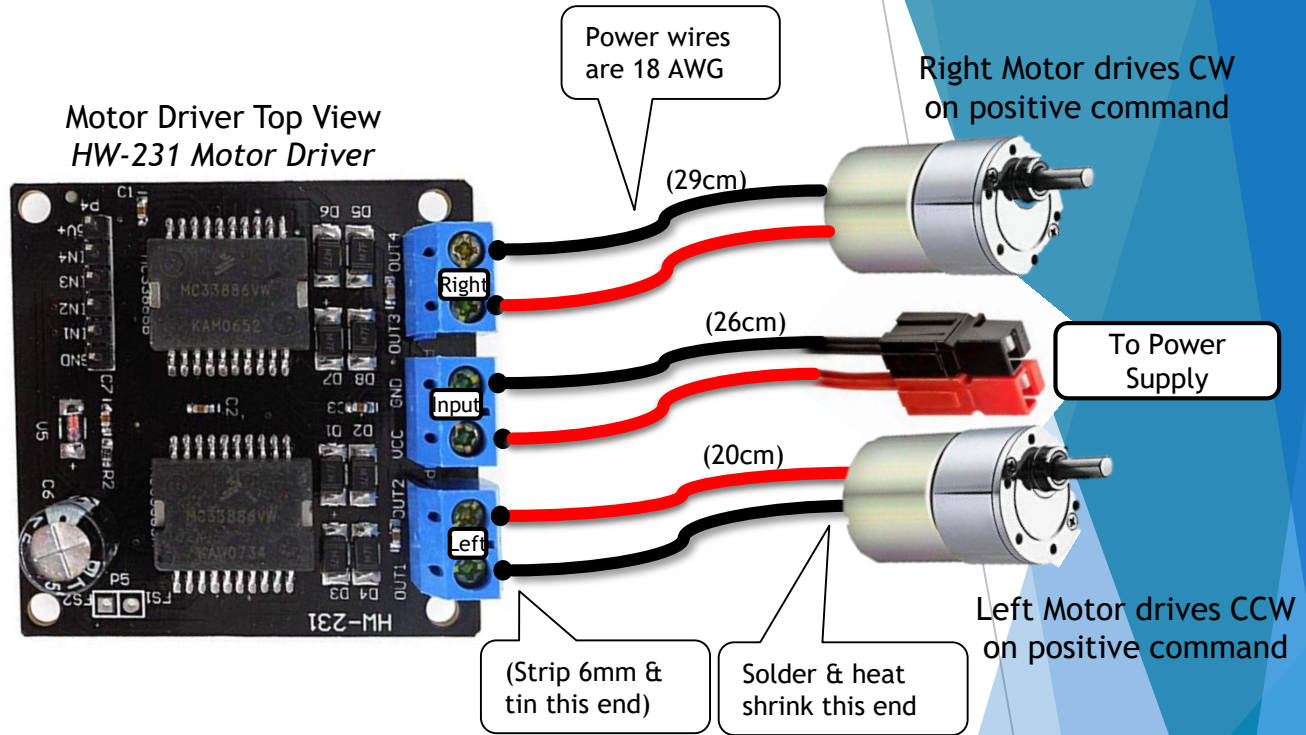
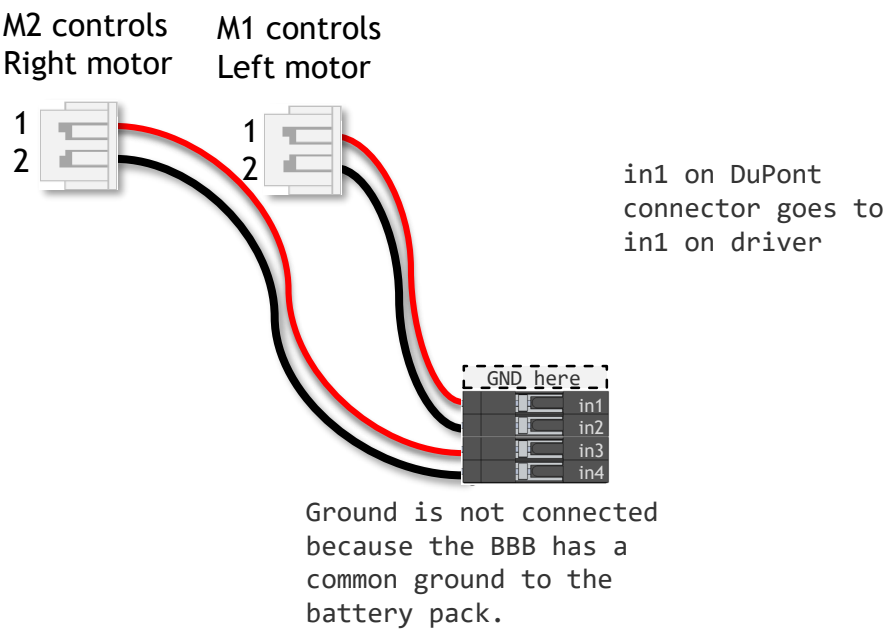
*All images of this style are copied directly from the beaglebone schematic*



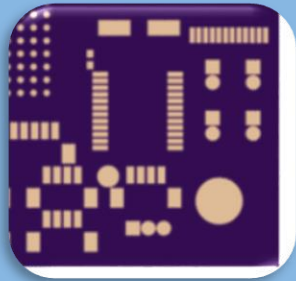
# Available Sensors & Actuators



# Motor Driver Signal Cables



The hardware design convention is pin 1 gets the square solder pad.

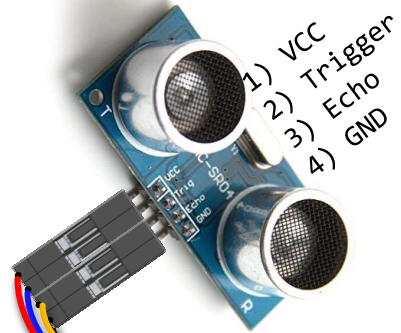


Connector vector image reserved.

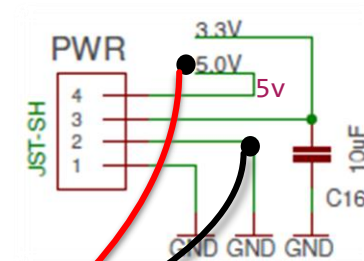




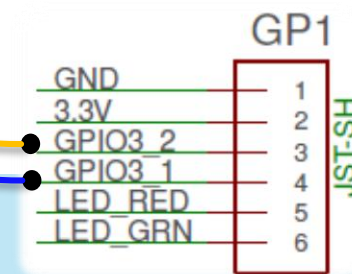
# Ultrasonic Distance Sensor (GPIO)



Connector Style: JST-SH 4-pin



Connector Style: JST-SH 6-pin

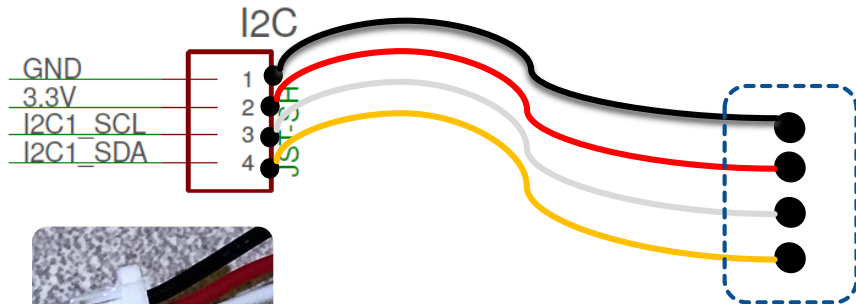


*NOTE: For JST connectors out-of-box,  
the colors are not in the correct order.  
You need to rearrange them.*

# Beaglebone to I2C bus cable

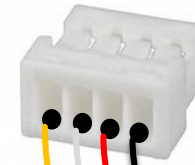
## Diagram

BeagleBone I2C Connector  
Style: JST-SH 4-pin

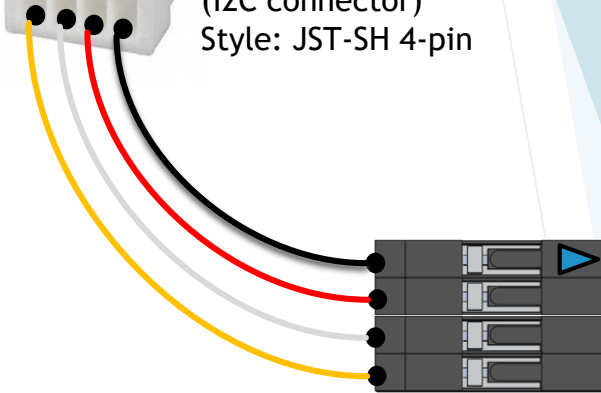


Plugs into I2C Bus PCB

## Cable Design



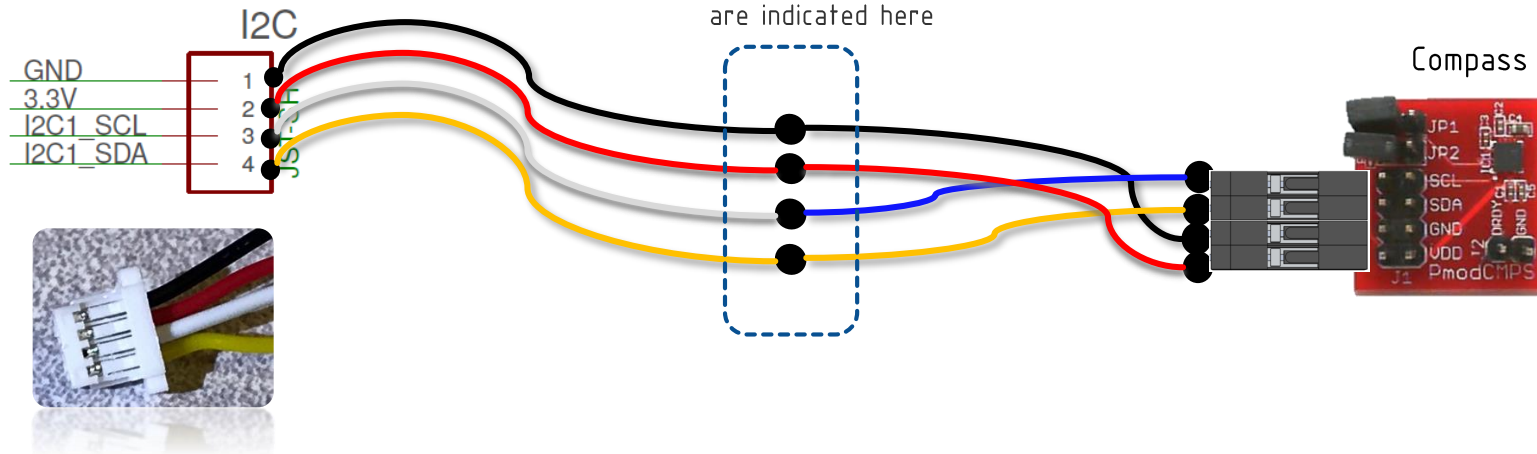
(I2C connector)  
Style: JST-SH 4-pin



DuPont style  
2.54mm female 4-pin

# Compass CMPS or CMPS2 (I2C)

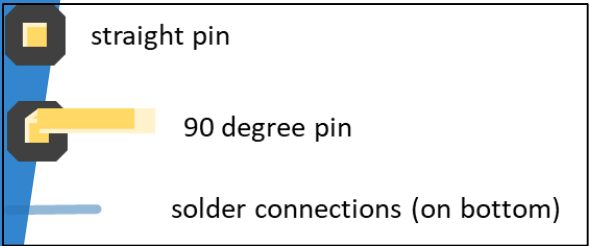
BeagleBone I2C Connector  
Style: JST-SH 4-pin



This compass is not necessary since you can access the compass on the beaglebone blue. Be sure to calibrate the compass on the blue since it lies within close proximity of magnetic hardware on the robot.

# I2C Bus Board

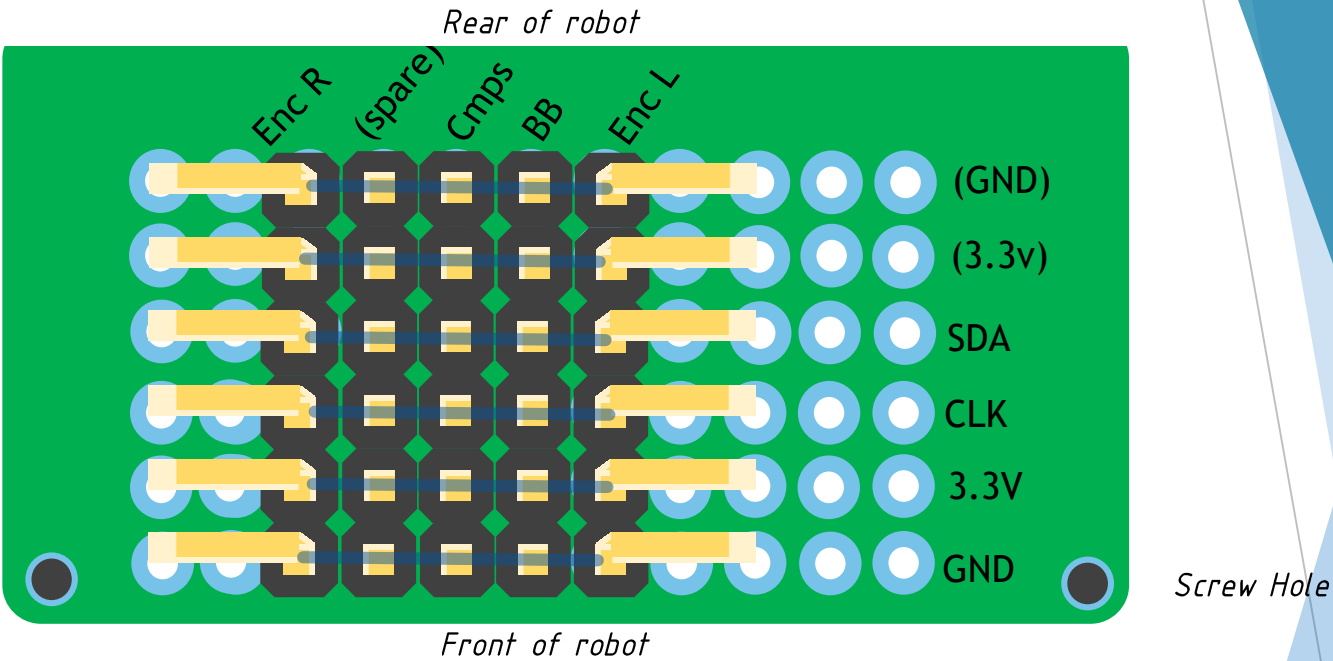
The board is made from a breadboard and soldered manually. The board can be cut between rows J & K. The solder bridges all pins from left to right.



straight pin

90 degree pin

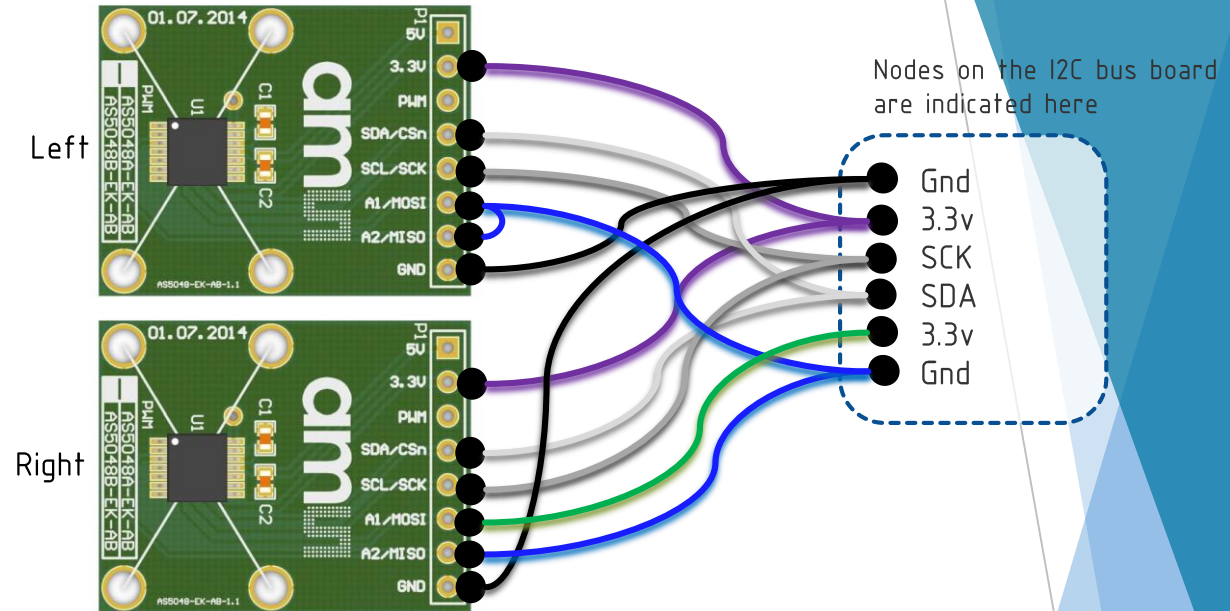
solder connections (on bottom)



# Encoder AS5048 (I2C)

Left Hand Encoder  
A1 is pulled **down** to GND  
I2C address is 0x40

Right Hand Encoder  
A1 is pulled **up** to 3.3v  
I2C address is 0x41

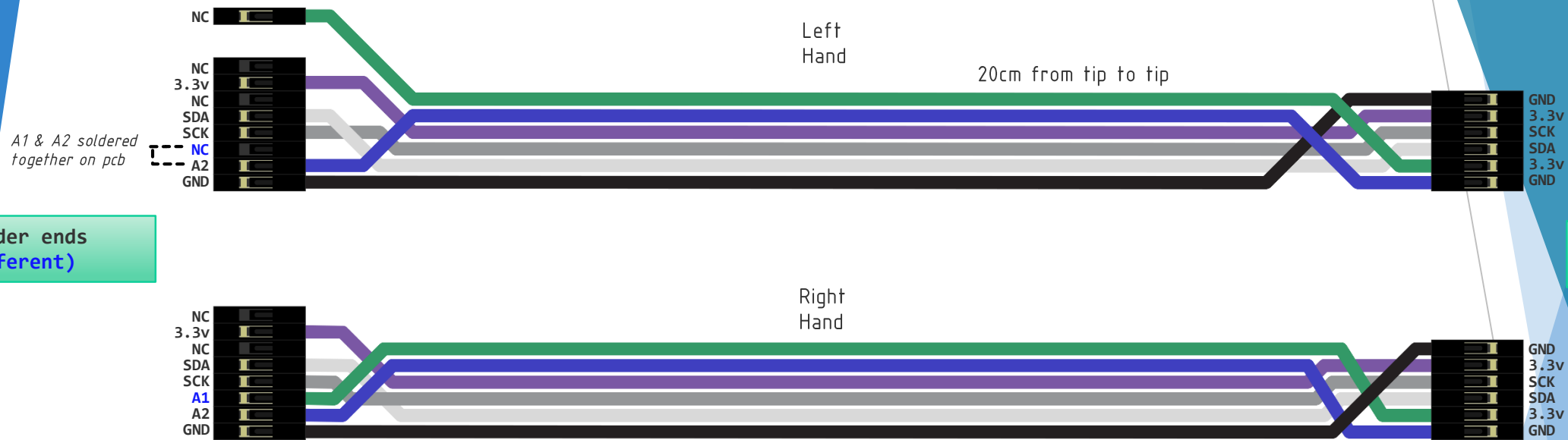


PIN	Left	Right
A1	0 (low)	1 (high)
A2	0 (low)	0 (low)
i2C Address	0x40	0x41

On the Left Hand Encoder PCB, bridge the pins A1 and A2 using solder, to each other.

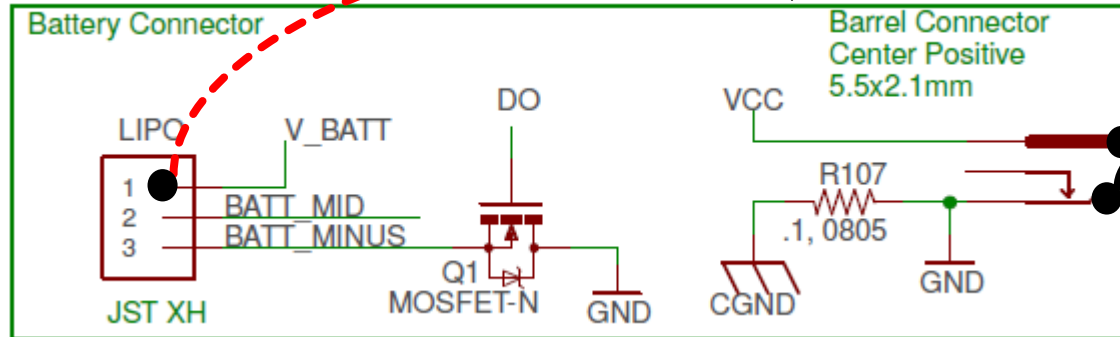


# Encoder Cables

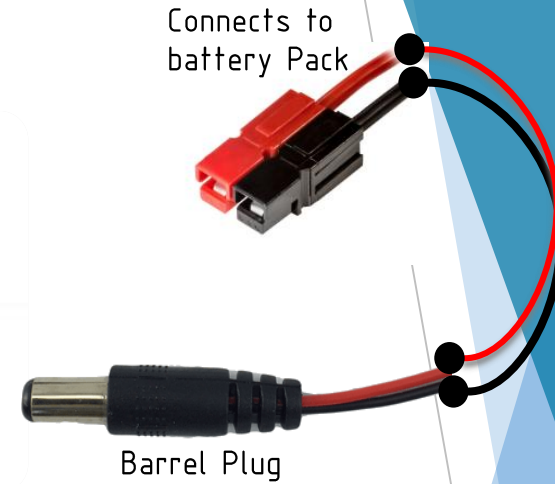


# Battery

*As an option, bridge  
LiPo terminal to 12v  
positive terminal (see  
Servo Slide)*

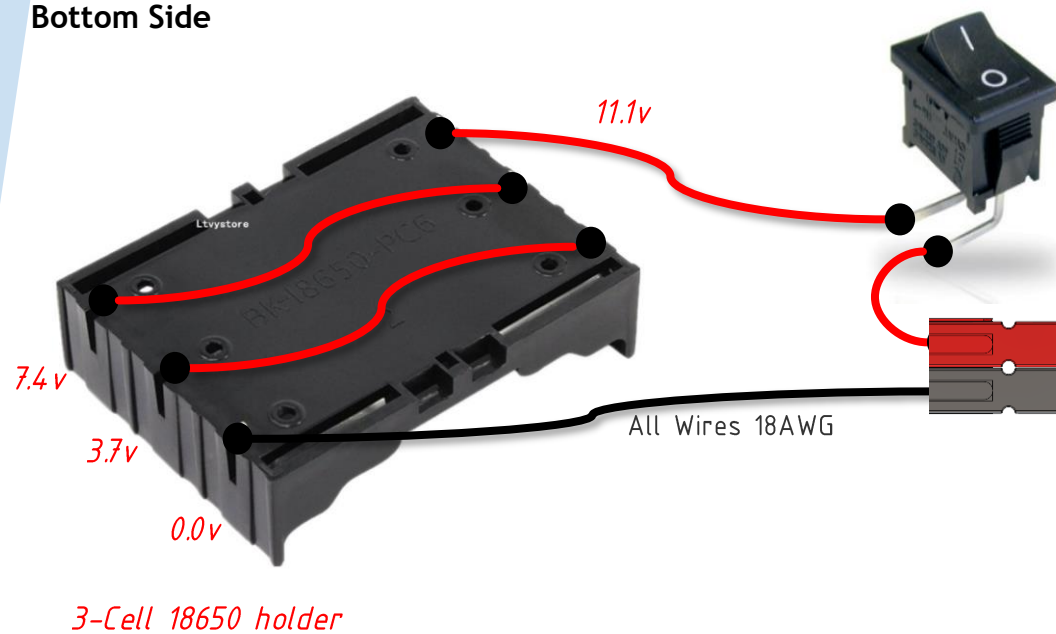


*The "Battery Connector" is disconnected. Actual battery uses Barrel Connector.*



# Battery Pack

Bottom Side



Switch PN:SRB22A2FBBNN  
Carries 10A max

Two pairs of Anderson  
connectors are attached here.

# LIDAR

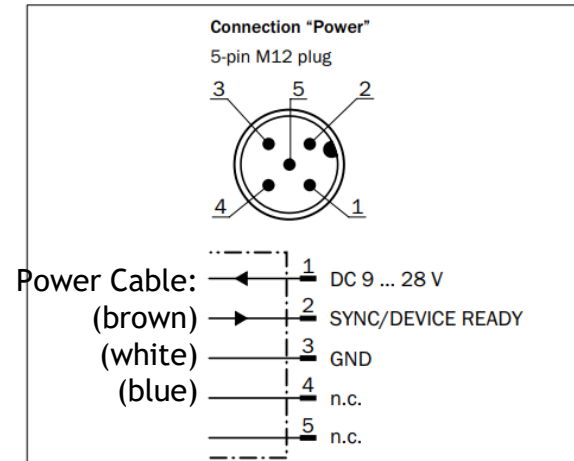
Lidar Device



TiM 561

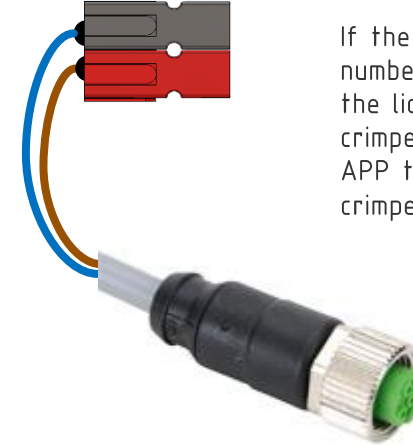
Power Connector Diagram (lidar side)

**POWER connection (supply voltage)**



*LIDAR-side connector (male pins)*

Power Cable Diagram (plugs into lidar)



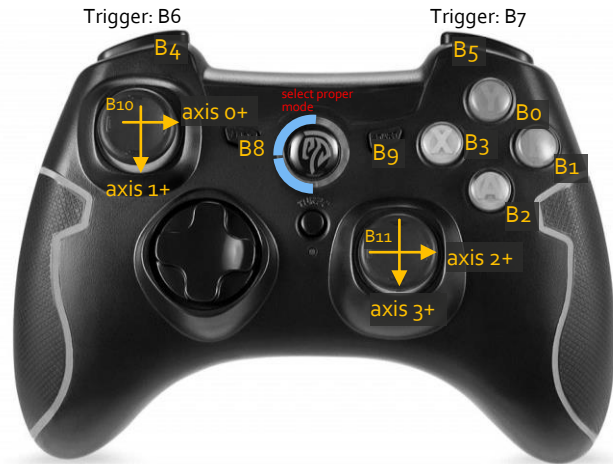
If the indicated cable part number is used for power to the lidar, brown will be crimped into the 12v positive APP terminal and blue is crimped into the negative.

Cable: 7000-12241-2150300

*Cable-side connector (female pins)*

Typical Lidar power consumption: 2.1w

## Gamepad Controls Mapping



### Button Behavior:

- not pressed: 0
- Pressed: 1

### Axis behavior:

- Right returns positive values
- down returns positive values



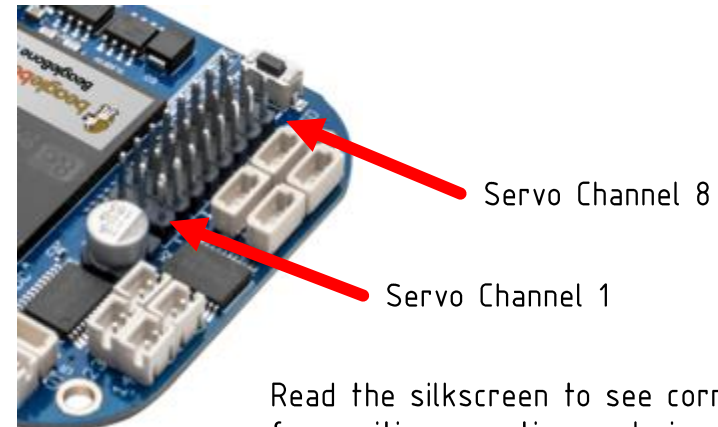
# Servos

## Bridge Power to the LiPo connector



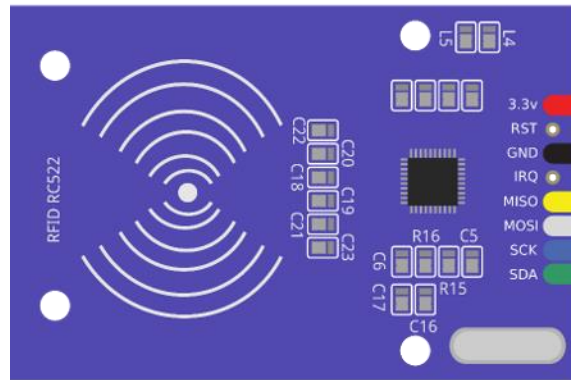
Without a power source available at the positive (third pin) input of the LiPo connector, the board has insufficient current available to the servos to drive servos at full torque or to drive multiple servos.

A safe fix is to solder the positive terminal of the DC jack to the third pin of the connector shown. When a battery is connected, the pins correspond to 0.0v, 3.7v, and 7.2v terminals of a 2-cell lipo.

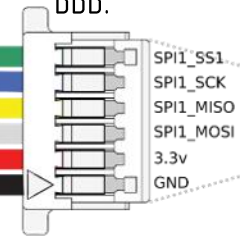


# RFID reader

RC522 low-cost  
RFID Scanner



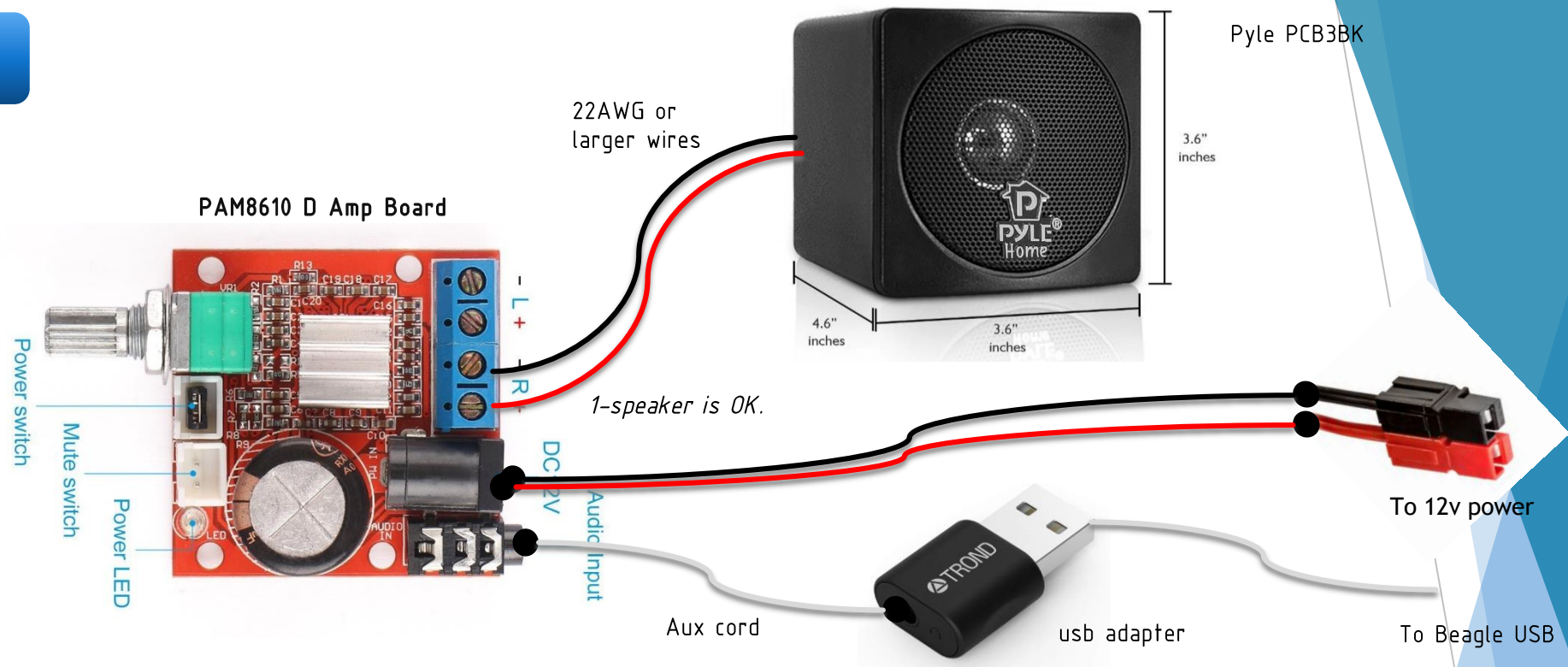
Plug into 6-pin  
JST-SH port on  
bbb.



SPI1\_SS1  
SPI1\_SCK  
SPI1\_MISO  
SPI1\_MOSI  
3.3v  
GND



# Audio Amp



# GPIO Connections

Example call for writing to this pin:  
`write(0,3,1)` # arguments: port, pin, state



output / Controls Green LED  
output / Controls Red LED

output  
output

3.3V

GND

3.3V

GND

input

output

input

output

3.3V

GND

3.3V

GND

GPIO connectors:  
JST-SH 6-pin

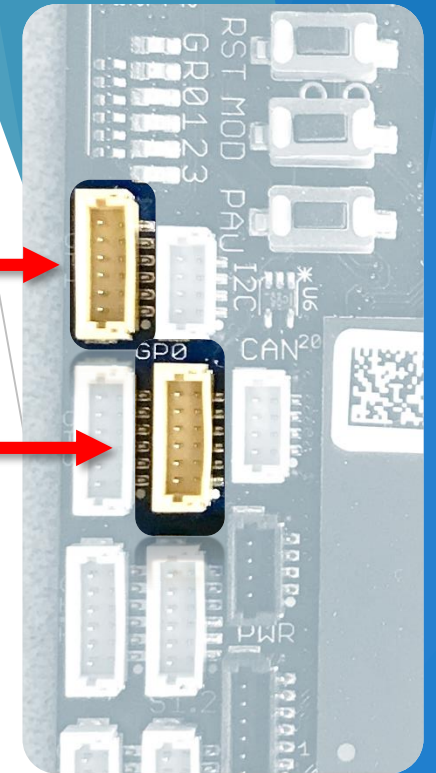
GP0

GP1

Example call for reading this pin:  
`read(0, 1)` #arguments: port, pin



SCUTTLE naming convention  
(used in L1\_gpio.py)



Connector vector image  
preserved for later use.



*Note: JST wires don't come  
with the proper color  
sequence. They must be  
rearranged.*



## GPIO Example - Relay

