Save As

Wire Label Control

(Macros must be enabled)

Turn Wire Label On

Turn Wire Label Off

Wire Gauge Legend

- 12-S — **12 AWG Silicone (Stranded)** (Up to 84.7Amps)
- 14-S — **14 AWG Silicone (Stranded)** (Up to 55.6 Amps)
- 20-S — **20 AWG Silicone (Stranded)** (Up to 13.7 Amps)
- 24 — **24 AWG (Solid Core)** (Up to 3.5 Amps)
- Str — ?? AWG (Stranded)

DOME

Front HoloProjector

Rear HoloProjector

Top HoloProjector

Front Servo Vertical

Front Servo Horizontal

Rear Servo Vertical

Rear Servo Horizontal

Top Servo Vertical

Top Servo Horizontal

R2 Servo Controller

Installed but not used

ROTOPOD Dome Button Controller

R2 Logics Controller

Front PSI

Rear PSI

R2 Dome Controller

MAG-TAG V3

R2 Dome Plate Controller

DB-25 Connector

MAG TAG V4

Small 1 of 3

Small 2 of 3

Small 3 of 3

Medium Side Panel

Medium Side Panel

Large Side Panel

Pie Panel 1

Pie Panel 2

Pie Panel 3

Pie Panel 4

Dome Warp Core Baseplate

Rotary Motor

Periscope Lighting Kit

Periscope Lifter Motor

R2 Uppity Spinner

Control Board

Sabertooth 2x32

Syrene 10

Solid State Relay (60Amp)

Solid State Relay (60Amp)

Solid State Relay (60Amp)

DC/DC Converter

Dome Electronics

RF Kill Switch

Ground Loop Isolator

vMusic

vMusic2

Stealth Controller

Slip Ring

MOLEX Mini-Fit Sr (Plug)

MOLEX Mini-Fit Sr (Receptacle)

MOLEX Mini-Fit Sr (Receptacle)

MOLEX Mini-Fit Sr (Plug)

Power In from Sled

Mounted to Skins (Rear Octagon Port)

Power Out to Sled

Mounted to Skins(12V or 5V??)

14.8V 10AH

14.8V 10AH

14.8V 10AH

Dome Motor

NPC

Right Foot Motor

NPC

Left Foot Motor

Ground Loop Isolator

SparkFun Spectrum Shield

The Dream Mixer Passive Edition

MP3 Trigger

Ground Loop Isolator

R2 Body Controller

Body LED Controller

Body Servos Controller

RADH Controller

RADH Sensor Ring

Dome Controller

Dome Plate Controller

Stealth Controller

Roam a dome home

R2 Servo Controller

Upper Utility Arm

Lower Utility Arm

Right Large Door

Left Large Door

Data Panel Door

CBI Door

Unknown at this time

This may replace the RF Kill Switch on the flip down control board

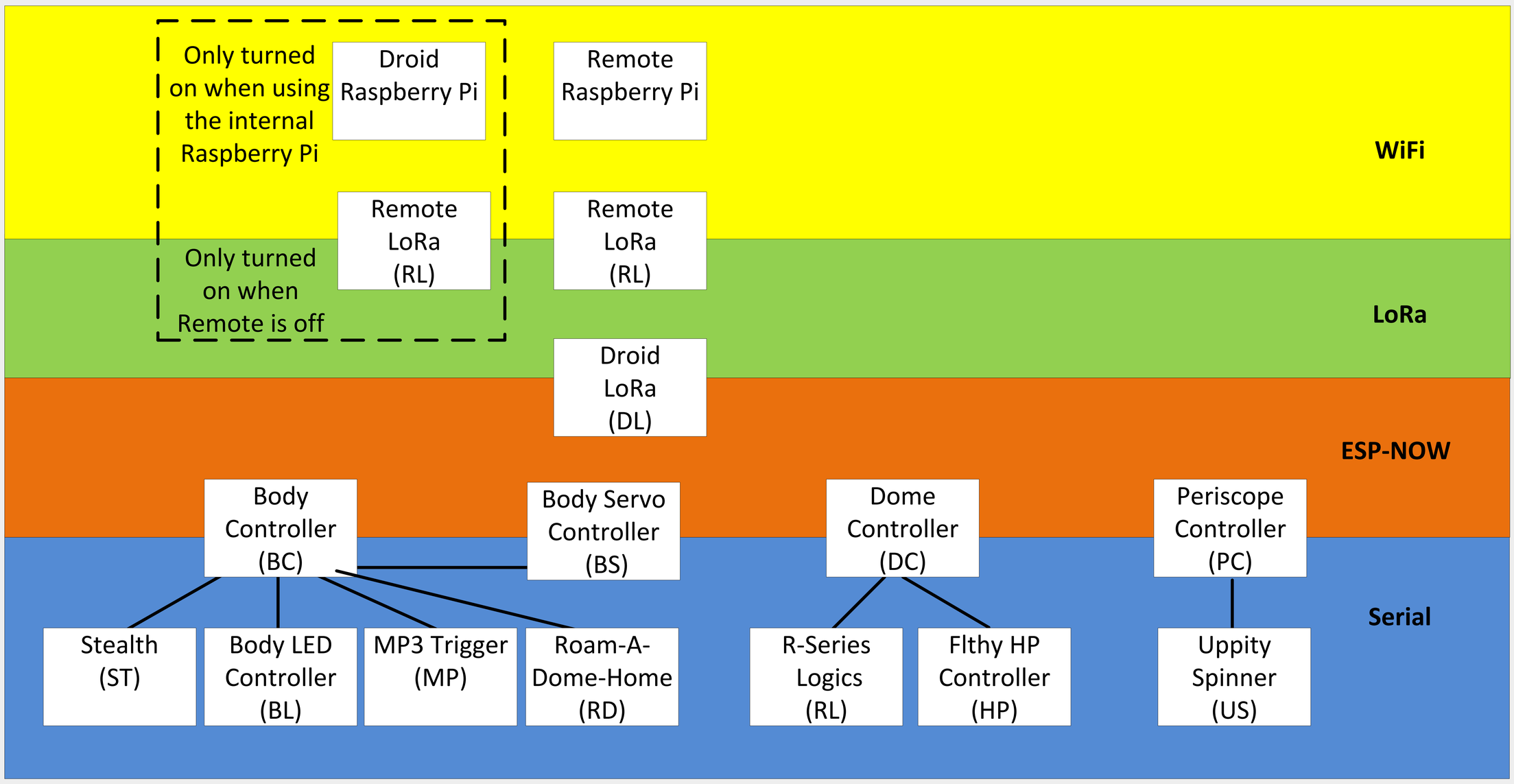
LoRa to ESP-NOW Bridge

Ground Loop Isolator

Actions	Details	Actions	Details	Actions	Details	Actions	Details
Open/Close Dome 1 – Opens the Dome 2 – Changes the Camaera Lens LED Colors to a random color each time it closes a few panels 3 – Sets the HP's to random pulse Right, Right	Double Click Double Click to get To the Details Page	**HP Rainbow** 1 - Waves the Dome Panels 2 - Flashes the Camera LEDs to Random Colors, 3 - Turns the HP LEDs to Rainbow Up, Up	Double Click Double Click to get To the Details Page	**Utility Arms Open/Close** 1 – Opens and closes the Utility Arms Button 4	Double Click Double Click to get To the Details Page		
Quick Wave 1 – Opens the Dome panels 2 – Changes the Camera Lens LED Colors to a random color each time it closes a few panels 3 – Changes the HPs to a random color each time it runs through a cycle of opening and closing Left, Left	Double Click Double Click to get To the Details Page	**HP Cyclone Blue** 1 - Waves the Dome Panels 2 - Flashes the Camera LEDs to Random Colors, 3 - Turns the HP LEDs to Rainbow Down, Down	Double Click Double Click to get To the Details Page				
Random Jittery Wave 1 – Opens the Dome panels 2 – Changes the Camera Lens LED Colors to a random color each time it closes a few panels 3 – Changes the HPs to a random color each time it runs through a cycle of opening and closing Left, Right,	Double Click Double Click to get To the Details Page	**HP Random Color Pulse** 1 – Pulses the HP's a random Color at medium speed Down, Up	Double Click Double Click to get To the Details Page				
Right, Left,		**HP Leia Message** 1 – Plays the HP's at the Leia function Up. Down	Double Click Double Click to get To the Details Page				
Wave 1 1 - Waves the Dome Panels 2 - Flashes the Camera LEDs to Random Colors, 3 - Turns the HP LEDs to Rainbow Right, Right, Right	Double Click Double Click to get To the Details Page						
Wave 2 1 - Waves the Dome Panels 2 - Flashes the Camera LEDs to Random Colors, 3 - Turns the HP LEDs to Rainbow Left, Left, Left	Double Click Double Click to get To the Details Page						
Left, Right, Left							
Right, Left, Right							

Back to Quick Reference Tab

Action	Gesture	Stealth Config.txt	Body Servo Expander I2C = 9	Body Arduino Pro Mini I2C = 8	Body Arduino Mega I2C = 7	Dome Expander I2C = 10	Flthy HP I2C = 25	HP Servo Board I2C = 65	Teeces (Serial)	Teeces (Serial)
			Space Holer							


Only turned on when using the internal Raspberry Pi
Droid Raspberry Pi
Remote Raspberry Pi
Remote LoRa (RL)
Remote LoRa (RL)
Only turned on when Remote is off
Droid LoRa (DL)
WiFi
LoRa
ESP-NOW
Serial
Body Controller (BC)
Body Servo Controller (BS)
Dome Controller (DC)
Periscope Controller (PC)
Stealth (ST)
Body LED Controller (BL)
MP3 Trigger (MP)
Roam-A-Dome-Home (RD)
R-Series Logics (RL)
Flthy HP Controller (HP)
Uppity Spinner (US)

Back to Quick Reference Tab

Action	Gesture	Stealth Config.txt	Body Servo Expander I2C = 9	Body Arduino Pro Mini I2C = 8	Body Arduino Mega I2C = 7	Dome Servo Expander I2C = 10	Flthy HP I2C = 25	HP Servo Board I2C = 65	Teeces (Serial)	Dome Button Cntr I2c = 11

Open/Close Dome
1 – Opens the Dome
2 – Changes the Camaera Lens LED Colors to a random color each time it closes a few panels
3 – Sets the HP's to random pulse
	
	Right, Right
		g=656,4,10,5				case 5:
						OpenClosePanels();
						break;
						void OpenClosePanels() {
						
						//digitalWrite(STATUS_LED, HIGH); // turn on STATUS LED so we can visually see we got the command on the board
						//Open or close All Pie Panels
						
						Serial.print("Pie Panels: ");
						colorWipe(C_RED, 255); // Red
						if (panelsOpen) { // Close the Pie Panels
						Serial.println("Closing");
						panelsOpen=false;
						// Attach to PANELS so we can move them
						Servos[LTL1].attach(LITTLE_PANEL_1_OF_3);
						Servos[LTL2].attach(LITTLE_PANEL_2_OF_3);
						Servos[LTL3].attach(LITTLE_PANEL_3_OF_3);
						Servos[MED1].attach(MED_PANEL_1_OF_2);
						Servos[MED2].attach(MED_PANEL_2_OF_2);
						Servos[LRG1].attach(LARGE_PANEL1);
						Servos[PIE1].attach(PIE1_SERVO_PIN);
						Servos[PIE2].attach(PIE2_SERVO_PIN);
						Servos[PIE3].attach(PIE3_SERVO_PIN);
						Servos[PIE4].attach(PIE4_SERVO_PIN);
						
						// Close them in a non-sequencial order and not at the same time to make it more interesting
						// Basically 2, then 2
						Servos[PIE1].write(NEUTRAL,CLOSESPEED);
						colorWipe(basicColors[random(1,10)], 255); // Turn Camera Lens LED Random Colors
						Servos[PIE3].write(NEUTRAL,CLOSESPEED,true); // wait
						Servos[PIE4].write(NEUTRAL,CLOSESPEED);
						colorWipe(basicColors[random(1,10)], 255); // Turn Camera Lens LED Random Colors
						Servos[PIE2].write(NEUTRAL,CLOSESPEED,true); // wait
						Servos[LTL1].write(NEUTRAL,CLOSESPEED);
						colorWipe(basicColors[random(1,10)], 255); // Turn Camera Lens LED Random Colors
						Servos[MED1].write(NEUTRAL,CLOSESPEED,true); //wait
						Servos[LTL2].write(NEUTRAL,CLOSESPEED);
						colorWipe(basicColors[random(1,10)], 255); // Turn Camera Lens LED Random Colors
						Servos[LRG1].write(NEUTRAL,CLOSESPEED,true); //wait
						Servos[LTL3].write(NEUTRAL,CLOSESPEED);
						colorWipe(basicColors[random(1,10)], 255); // Turn Camera Lens LED Random Colors
						Servos[MED2].write(NEUTRAL,CLOSESPEED,true); //wait
						
						// Detach from the Pies
						Servos[LTL1].detach();
						Servos[LTL2].detach();
						Servos[LTL3].detach();
						Servos[MED1].detach();
						Servos[MED2].detach();
						Servos[LRG1].detach();
						Servos[PIE1].detach();
						Servos[PIE2].detach();
						Servos[PIE3].detach();
						Servos[PIE4].detach();
						
						Serial.println("Closed");
						colorWipe(C_BLUE, 255); // Turn Camera Lens LED Blue
						
						} else { // Open Pie Panels
						Serial.println("Opening");
						panelsOpen=true;
						colorWipe(C_GREEN, 255); // Turn Camera Lens LED Green
						// Attach to all the panel so we can move them
						Servos[LTL1].attach(LITTLE_PANEL_1_OF_3);
						Servos[LTL2].attach(LITTLE_PANEL_2_OF_3);
						Servos[LTL3].attach(LITTLE_PANEL_3_OF_3);
						Servos[MED1].attach(MED_PANEL_1_OF_2);
						Servos[MED2].attach(MED_PANEL_2_OF_2);
						Servos[LRG1].attach(LARGE_PANEL1);
						Servos[PIE1].attach(PIE1_SERVO_PIN);
						Servos[PIE2].attach(PIE2_SERVO_PIN);
						Servos[PIE3].attach(PIE3_SERVO_PIN);
						Servos[PIE4].attach(PIE4_SERVO_PIN);
						
						// Open them all at once.
						Servos[LTL1].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[LTL2].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[LTL3].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[MED1].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[MED2].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[LRG1].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[PIE1].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[PIE2].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[PIE3].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED);
						Servos[PIE4].write(NEUTRAL-PIE_ADD_DEGREES,OPENSPEED, true);
						
						// Detach from the Pies
						Servos[LTL1].detach();
						Servos[LTL2].detach();
						Servos[LTL3].detach();
						Servos[MED1].detach();
						Servos[MED2].detach();
						Servos[LRG1].detach();
						Servos[PIE1].detach();
						Servos[PIE2].detach();
						Servos[PIE3].detach();
						Servos[PIE4].detach();
						
						Serial.println("Opened");
						
						}
						i2cCommand=-1; // always reset i2cCommand to -1, so we don't repeatedly do the same command
						//digitalWrite(STATUS_LED, LOW);
						}				

Back to Quick Reference Tab

Action	Gesture	Stealth Config.txt	Body Servo Expander I2C = 9	Body Arduino Pro Mini I2C = 8	Body Arduino Mega I2C = 7	Dome Expander I2C = 10	Flthy HP I2C = 25	HP Servo Board I2C = 65	Teeces (Serial)	Teeces (Serial)
Quick Wave 1 – Opens the Dome 2 – Changes the Camaera Lens LED Colors to a random color each time it closes a few panels 3 – Sets the HP's to random pulse	 Left, Left	g=454,4,10,8 →				case 8: quickWave(); break;				

```
void Wave1() {
  sendFlthy2Cond("A007");
  Serial.println("Wave Panels 1");
  // Wave 1
                // Attach to all the panel so we can move them
                Servos[LTL1].attach(LITTLE_PANEL_1_OF_3);
                Servos[LTL2].attach(LITTLE_PANEL_2_OF_3);
                Servos[LTL3].attach(LITTLE_PANEL_3_OF_3);
                Servos[MED1].attach(MED_PANEL_1_OF_2);
                Servos[MED2].attach(MED_PANEL_2_OF_2);
                Servos[LRG1].attach(LARGE_PANEL);
                Servos[PIE1].attach(PIE1_SERVO_PIN);
                Servos[PIE2].attach(PIE2_SERVO_PIN);
                Servos[PIE3].attach(PIE3_SERVO_PIN);
                Servos[PIE4].attach(PIE4_SERVO_PIN);

                //Move the Servos to Nuetral position
                    Servos[LTL1].write(NEUTRAL,150);
                Servos[LTL2].write(NEUTRAL,150);
                Servos[LTL3].write(NEUTRAL,150);
                Servos[MED1].write(NEUTRAL,150);
                Servos[MED2].write(NEUTRAL,150);
                Servos[LRG1].write(NEUTRAL,150);
                    Servos[PIE1].write(NEUTRAL,150);
                    Servos[PIE2].write(NEUTRAL,150);
                    Servos[PIE3].write(NEUTRAL,150);
                    Servos[PIE4].write(NEUTRAL,150);


  for (int i=0; i<4; i++) {                                   //RUN THIS 4 TIMES
    Serial.println("Wave");
                colorWipe(basicColors[random(0,10)], 255);     // Turn Camera Lens LED Random Colors
                Servos[LTL1].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[LTL2].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[LTL3].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[MED1].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[MED2].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[LRG1].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE1].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE2].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE3].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE4].write(NEUTRAL-PIE_ADD_DEGREES,WAVESPEED);
                delay(WAVEDELAY * 2.5);
                Servos[LTL1].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[LTL2].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[LTL3].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[MED1].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[MED2].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[LRG1].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE4].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE3].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE2].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY);
                Servos[PIE1].write(NEUTRAL,WAVESPEED);
                delay(WAVEDELAY * 2);
  }

  // Disconnect from our servos
                Servos[TL1].detach();
                Servos[LTL2].detach();
                Servos[TL3].detach();
                Servos[MED1].detach();
                Servos[MED2].detach();
                Servos[LRG1].detach();
                Servos[PIE1].detach();
                Servos[PIE2].detach();
                Servos[PIE3].detach();
                Servos[PIE4].detach();

    colorWipe(C_BLUE, 255);                    // Turn Camera Lens LED Blue
    sendFlthy2Cmd("A00354");
Serial.println("Wave Done");


  i2cCommand=-1; // always reset i2cCommand to -1, so we don't repeatedly do the same command
  //digitalWrite(STATUS_LED, LOW);
}
```

Turn all HPs to Rainbow

Turn all HPs to pulse blue and speed of 4