PROJECTS WITH PLUTO

Using Pluto Blocks

Version 2.0

pluto blocks

Table of Contents

Introduction

Pluto is not just a drone, it is a development tool. It enables the user to develop numerous projects across different areas. With the Pluto Platform, you can create and innovate to bring your ideas into reality.

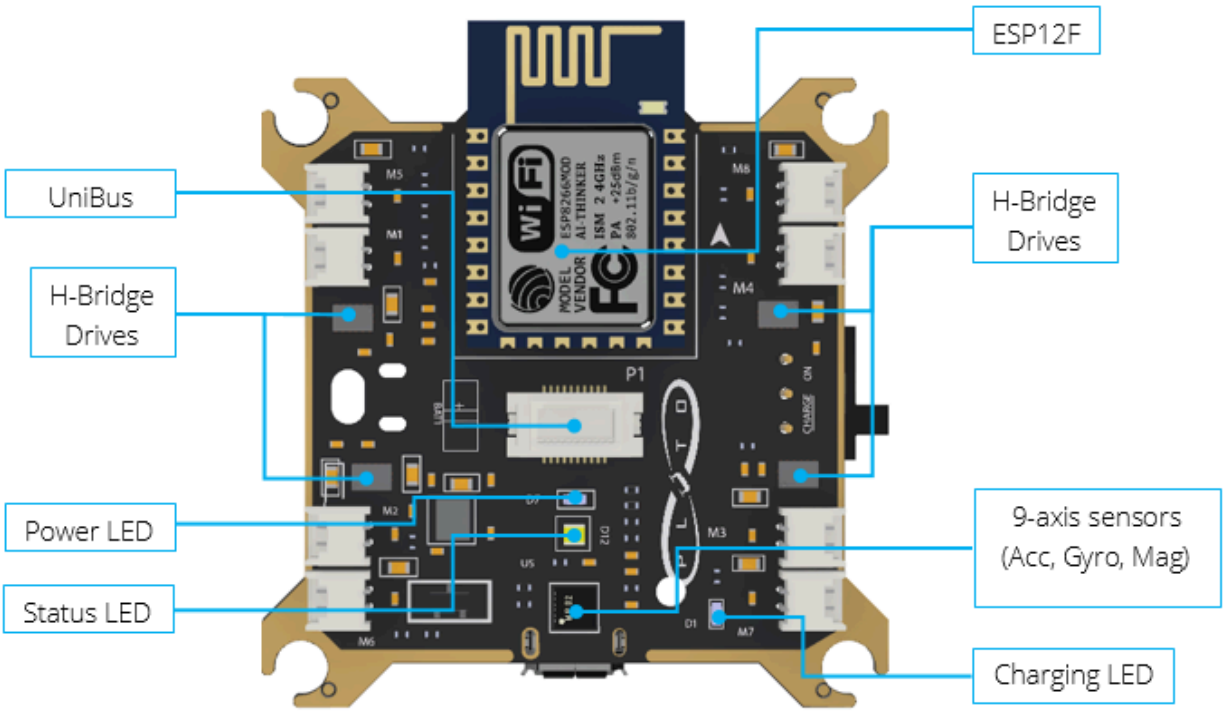
From simply glowing LEDs, to understanding complicated scientific concepts, to developing fun games - everything is possible with the Pluto Platform. The sensor data and the remote control data are all accessible with this platform, making it one of a kind.

With the platform being open source, tinkering is made a lot easier with the API based coding in C++. And if you have never coded in your life, we have something for you as well - Pluto Blocks.

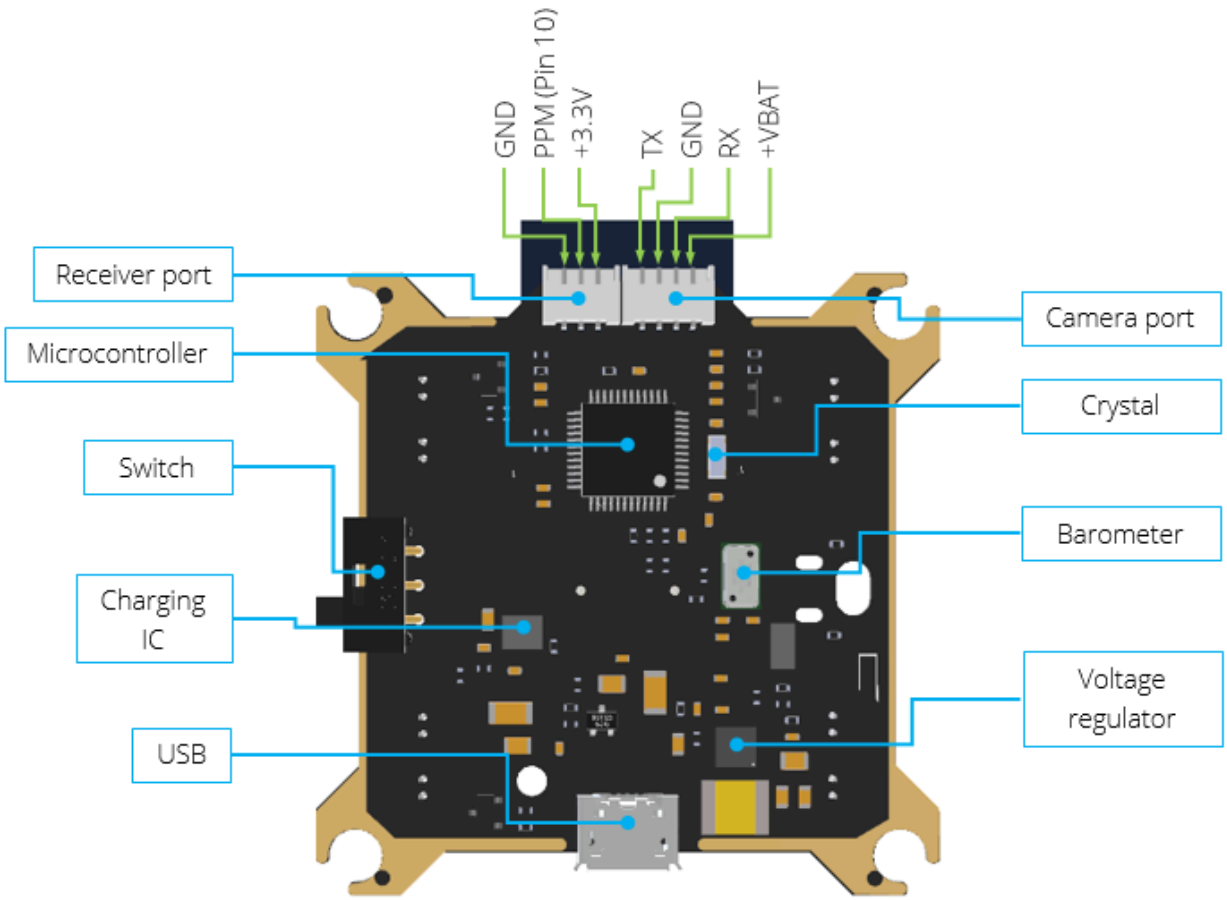
Pluto Blocks is an easy drag and drop block programming application which does not require any prior knowledge of programming languages. It provides a comprehensible platform to tinker with Pluto 1.2 and Pluto X nano drones. With PlutoBlocks, we aim to motivate the users to begin their journey into the universe of drones. Some of the key features of Pluto Blocks are:

- ✓ Easy drag and drop blocks
- ✓ Real time C++ code in separate tab helps in learning the programming language
- ✓ No separate software required to build and flash the code
- ✓ Wireless flashing makes everything much easier and faster
- ✓ User friendly UI makes it exciting for users of all ages

Board Details

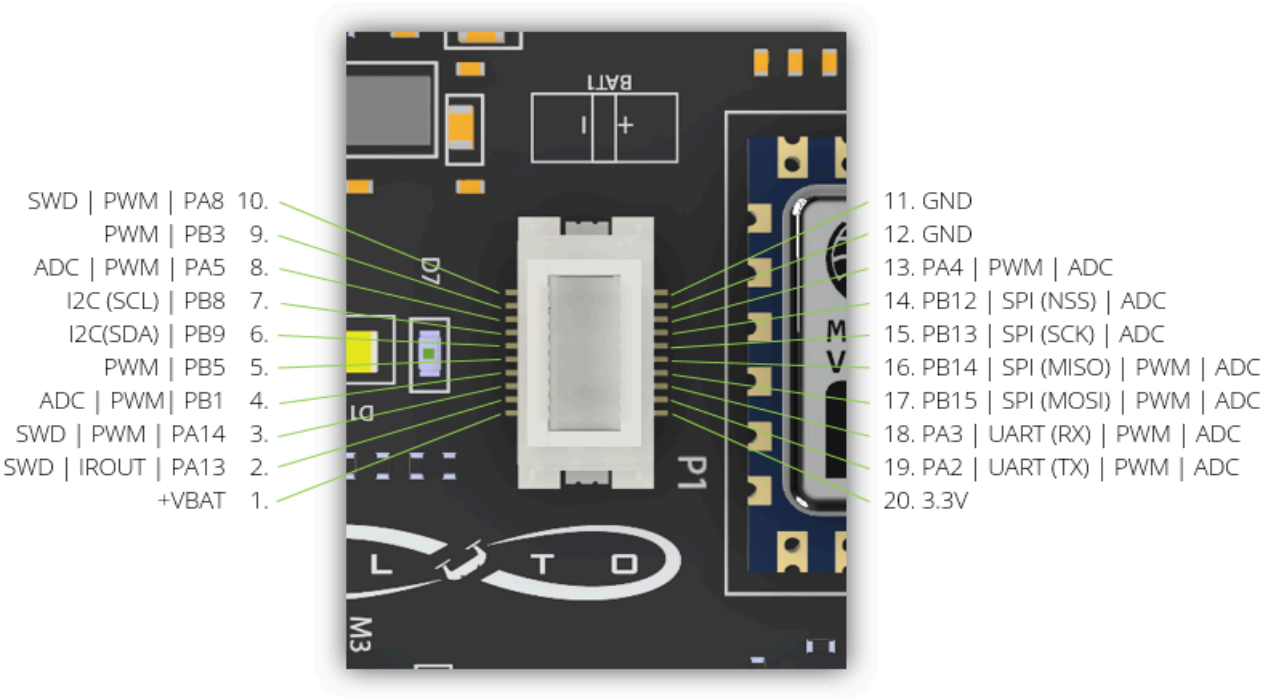


Primus X (TOP)


GND
PPM (Pin 10)
+3.3V
TX
GND
RX
+VBAT
Receiver port
Microcontroller
Switch
Charging IC
USB
Camera port
Crystal
Barometer
Voltage regulator

Primus X (BOT)

UniBus pin-out



Compatibility

The Projects with Pluto: Using Pluto Blocks (version 1.0) contains projects that are built on Pluto Blocks software and executed using Pluto nano drone. The compatibility for both are given below.

Software:

- Pluto Blocks 1.0.0
- Guide to Pluto Blocks 1.0.0

Drones:

- Pluto X nano drone
- Pluto 1.2 nano drone

Preparing to Take Off

Given below are a few prerequisites for using Pluto Blocks. Please make sure to fulfill them for a smooth experience with Pluto Blocks.

Pluto Blocks Software

- Download and install the latest version of Pluto Blocks Software (Beta) using the link provided below.

Pluto Blocks Software Download Link

NOTE: The current version will work only with 64 bit Windows PC

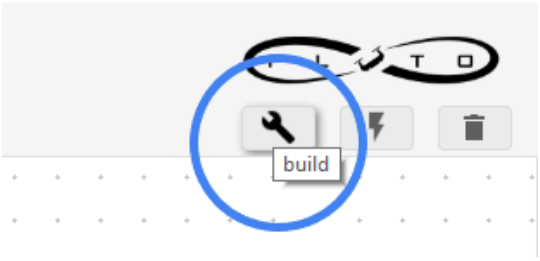
- Please refer to the video provided in the link given below for support regarding the installation of Pluto Blocks.

Pluto Blocks Installation Video Link

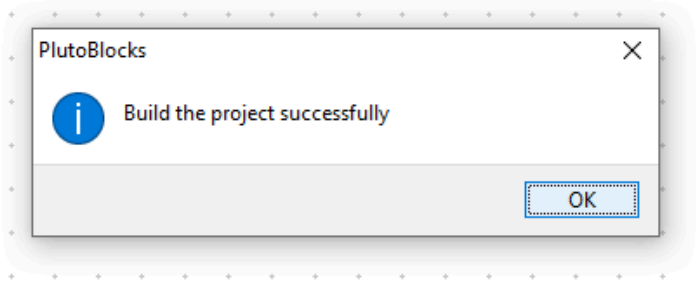
Building and Flashing

After coding any project, you will have to *build* and *flash* it on your Pluto drone. Programs can be flashed on Pluto wirelessly, making it an effortless experience. To get an idea, please build and flash a blank project as guided in the below given steps:

- In a new project, without adding any block, click on the '*build*' option as shown

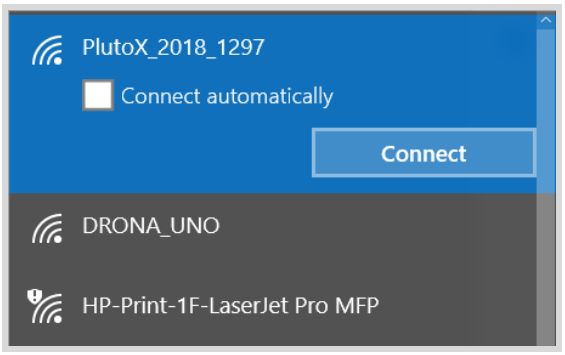


- The project building process will begin

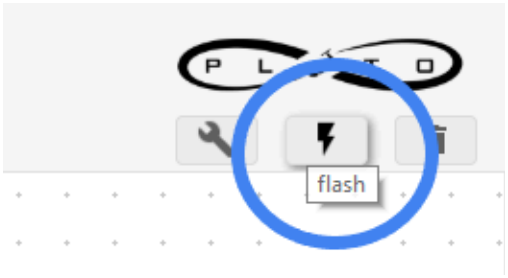


- After the successful building process, you will get a message confirming the same

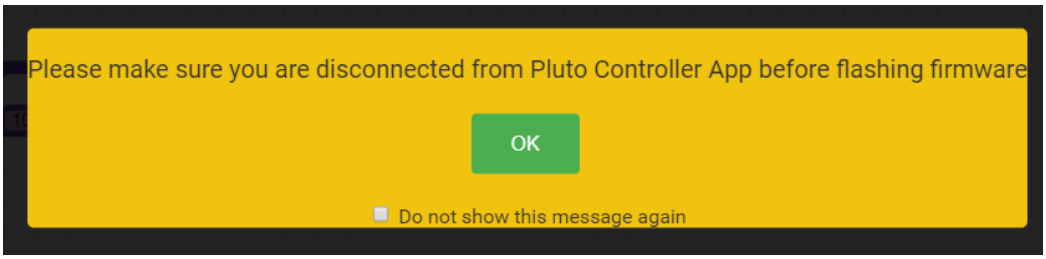
- Connect your Pluto drone with your computer Wi-Fi as shown



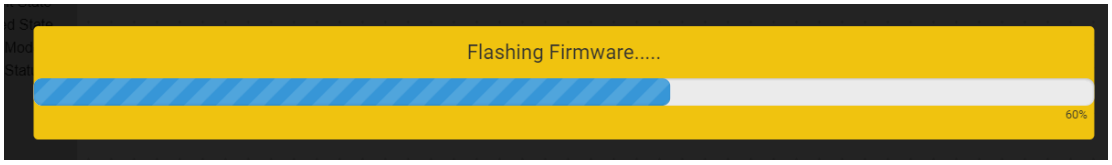
→ Now, click on the '*flash*' option as shown



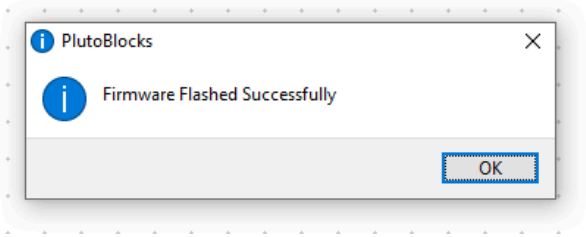
→ While flashing a program on Pluto, the drone should not be connected with the Pluto Controller App. If you are already disconnected, click on '*OK*' option



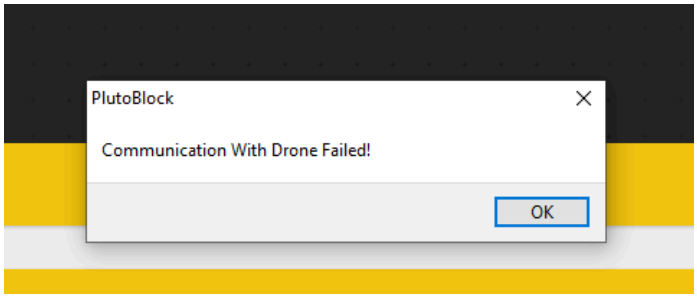
→ The program will flash on your Pluto



→ After successful flashing, you will get a message confirming the same



- In case you get the below message while flashing, it means that the connection between your computer and Pluto has been lost.

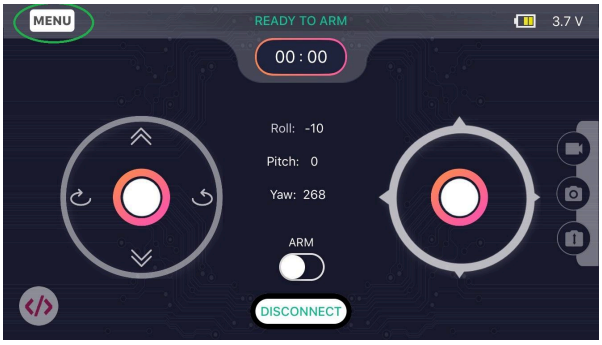
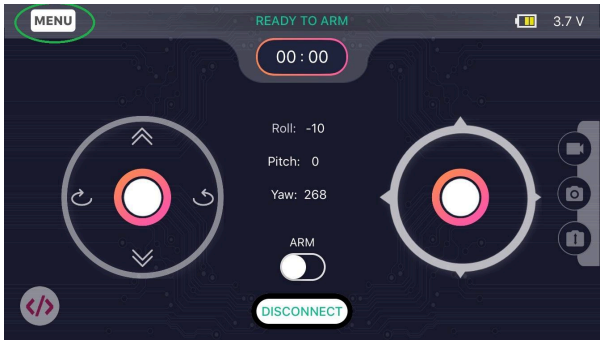
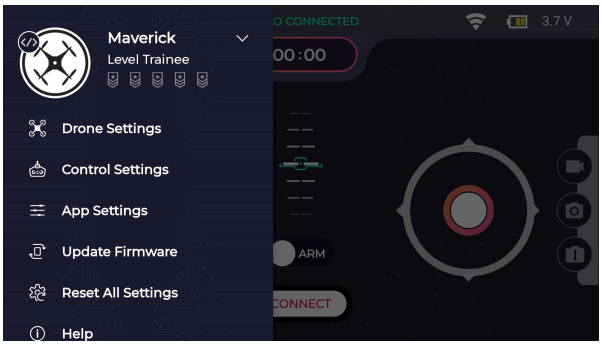
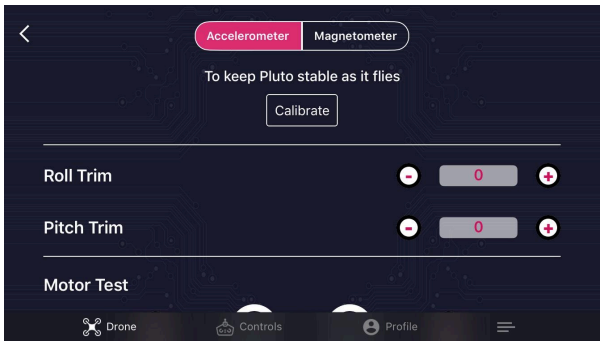
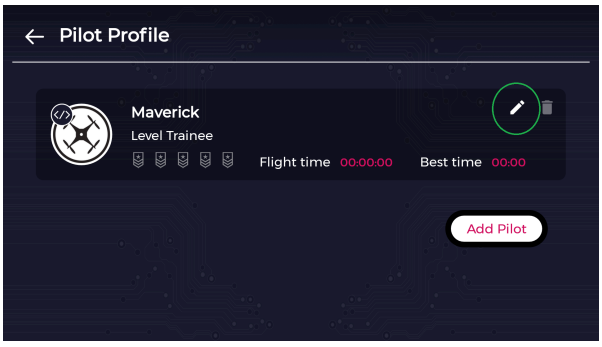
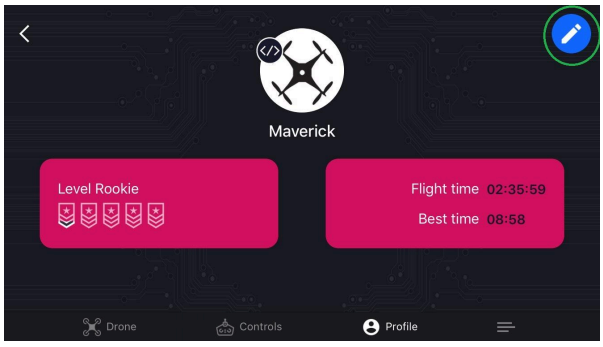


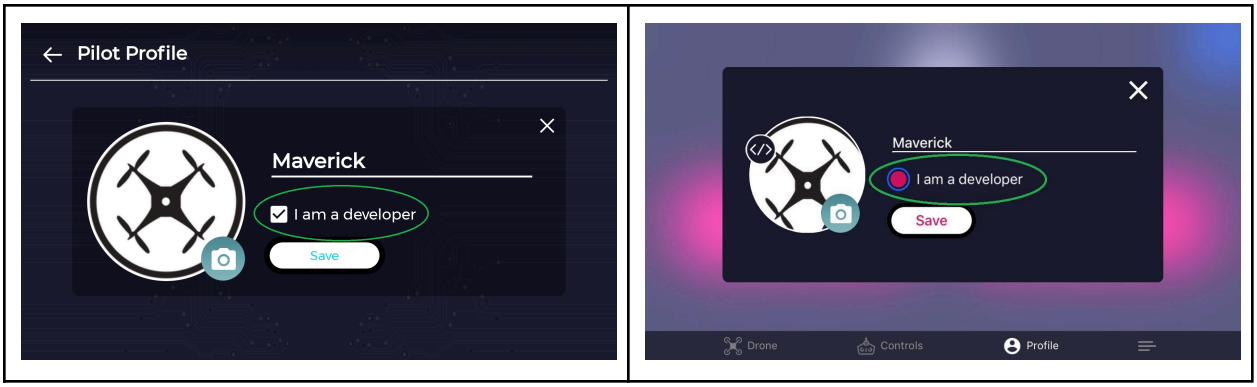
This could happen because of various reasons. We suggest you reconnect your drone with the computer and try flashing the program again.

Developer Mode

The projects that will be built and flashed through Pluto Blocks can be executed only in Developer Mode. Follow the steps given below to enable Developer Mode button on the main screen of the Pluto Controller App

Go to “menu” → select your profile → edit → check “I am a developer” option.

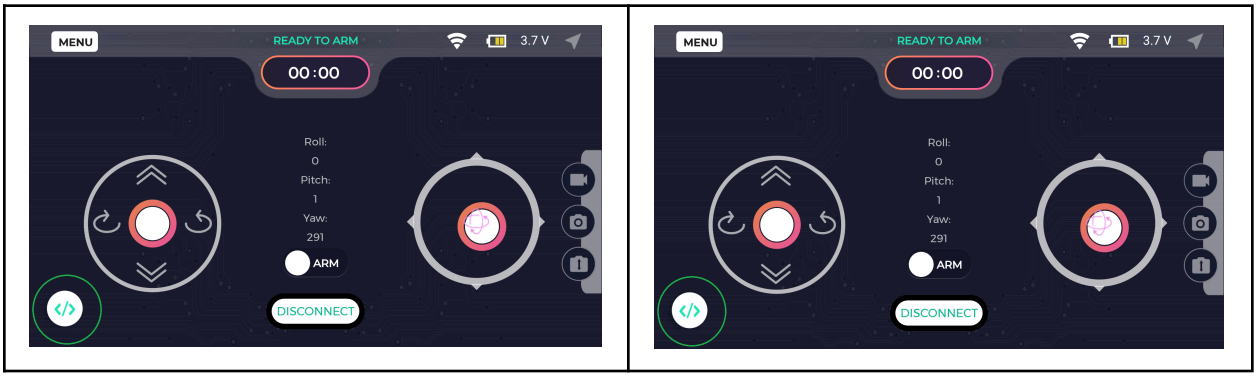
Android	IOS
	
	
	



After selecting the “I am a developer” option, you can see the Developer Mode button on your main screen.

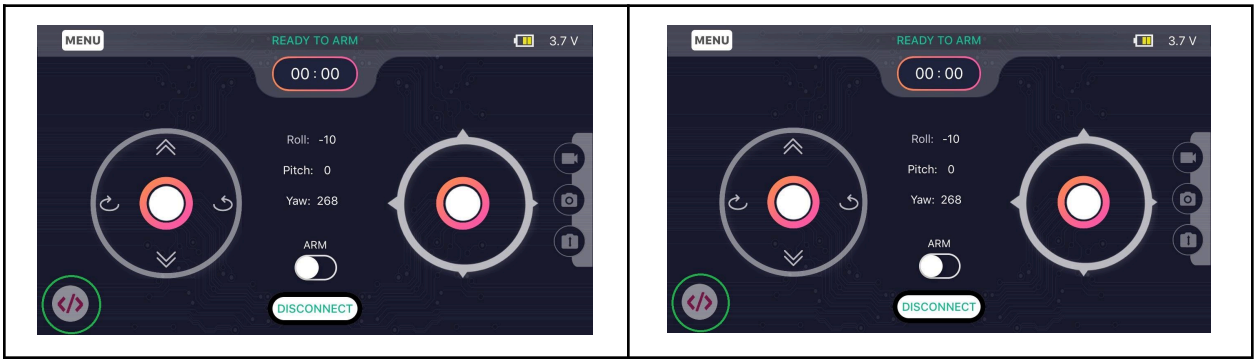
Using Developer Mode

- Fade in (green) indicates that the Developer Mode is ON. The blocks that are included in the ‘On Start’ block, will work when the Developer Mode is switched ON.



- The blocks that are included in ‘Pluto Loop’ work during the time the Developer Mode is kept ON.

- Fade out (red) indicates that the Developer Mode is OFF. The blocks that are included in ‘On Stop’ block, will work when the Developer Mode is switched OFF.



Let's Begin

The first step in this enlightening and exciting journey of using Block Programming is to understand the working of various blocks. By knowing the functions of every block, you will be able to create your own innovative programs and see them in action using your Pluto drone.

This section contains the beginner's level projects, aimed at helping you to get used to Pluto Blocks and understanding the working of different blocks. Please feel free to refer to the Guide to Pluto Blocks alongside.

Guide to Pluto Blocks Link

To begin with, let us have a look at Project 0 given on the next page!

Project 0

Challenge

Build a project to turn the Red LED ON.

Solution

Running the project:

- Build and flash the project (Refer *Building and Flashing* section for steps)
- Connect the Pluto Controller App with the drone and turn the Developer Mode ON to execute the project
- The Red LED will turn ON
- The LED will turn OFF when the Developer Mode will be turned OFF

REMEMBER:

- **All the projects have to be built and flashed in the same manner**
- **The projects will run in the Developer Mode only**

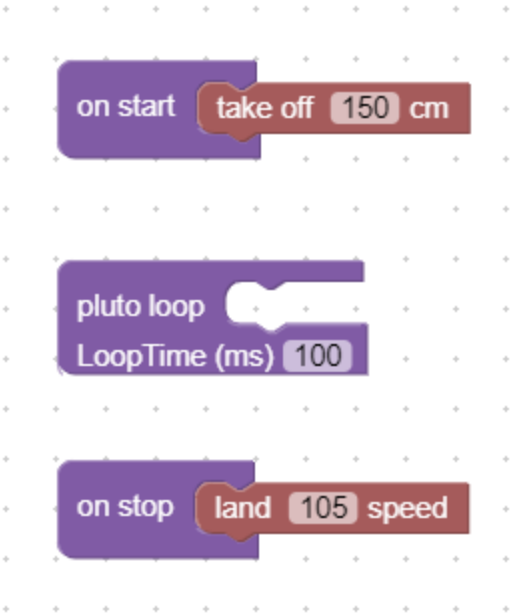
After understanding Project 0, let us try more block programming with the help of the next projects. We recommend you to first try building the projects on your own by reading the challenge and later refer to the solution to review your project.

Project 1

Challenge:

Build a program such that the drone will take off when the Developer Mode is turned ON and the drone will land when the Developer Mode is turned OFF.

Solution:



Pro tip:

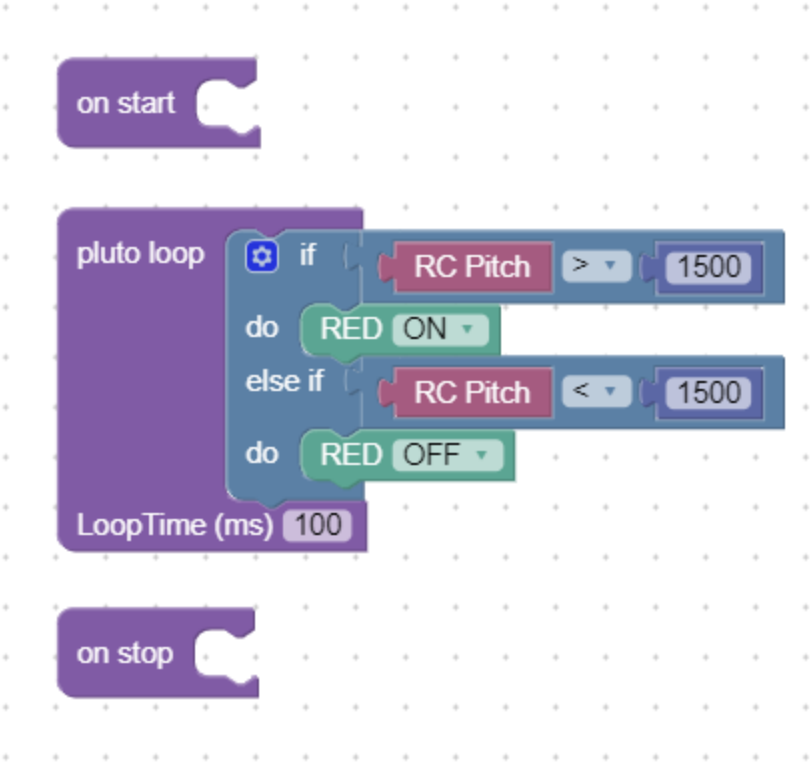
If a project involves flying the drone, make sure you have enough space around the drone while taking off.

Project 2

Challenge:

Build a program such that if the Pitch input from the Pluto Controller App (RC Pitch) is more than 1500, the Red LED should turn ON and if it is less than 1500, then the Red LED should turn OFF.

Solution:



Pro tip:

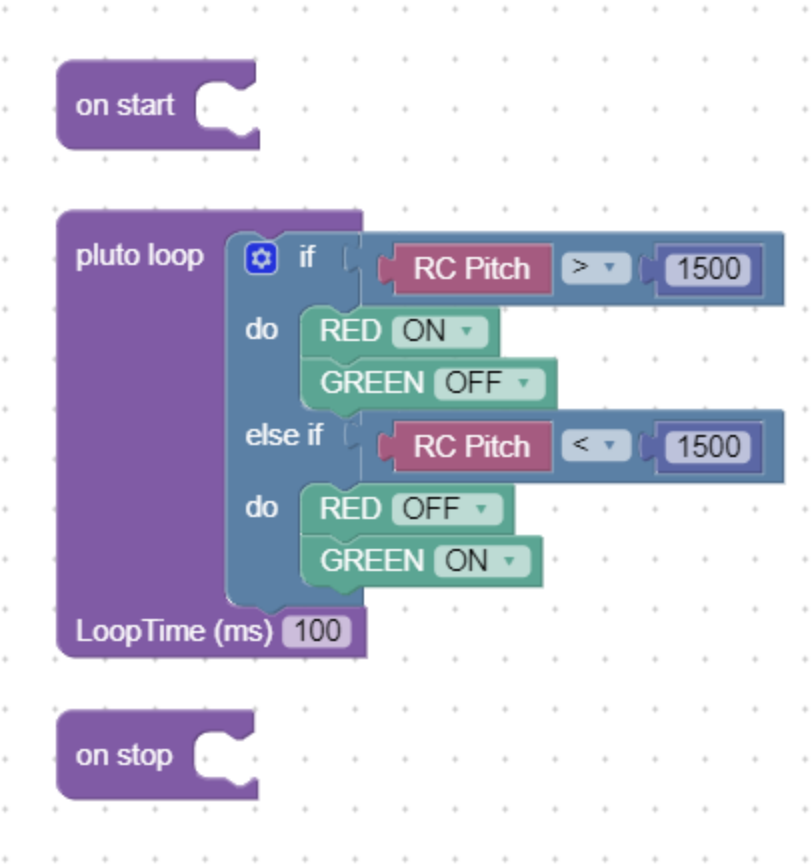
Did you fly the drone to execute this project? This project can run even without flying or arming. Try it!

Project 3

Challenge:

Build a program such that if the Pitch input from the Pluto Controller App (RC Pitch) is more than 1500, the Red LED should turn ON and if it is less than 1500, then the Green LED should turn ON.

Solution:



Pro tip:

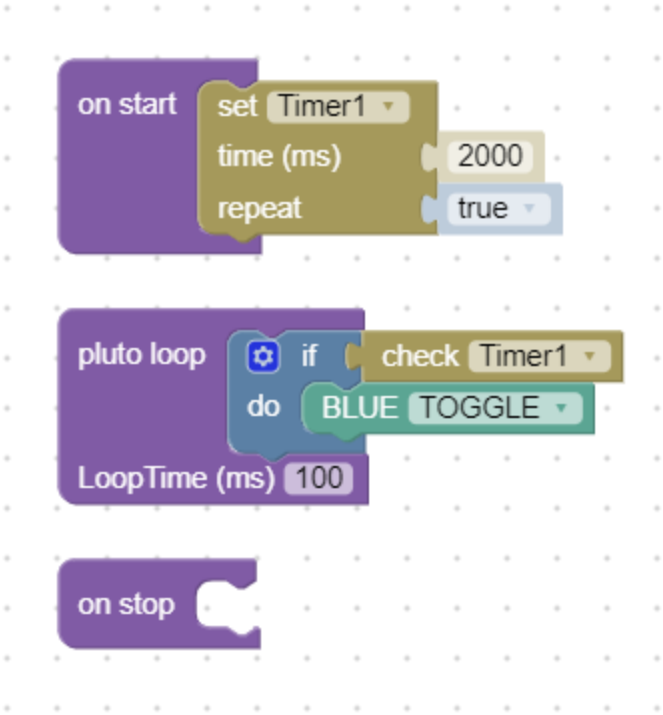
Remember to turn the first LED OFF when you want another LED to turn ON. If not turned OFF, the first LED will remain ON along with the second LED. Test it yourself!

Project 4

Challenge:

Build a program such that the Blue LED will TOGGLE after every 2 seconds.

Solution:



Pro tip:

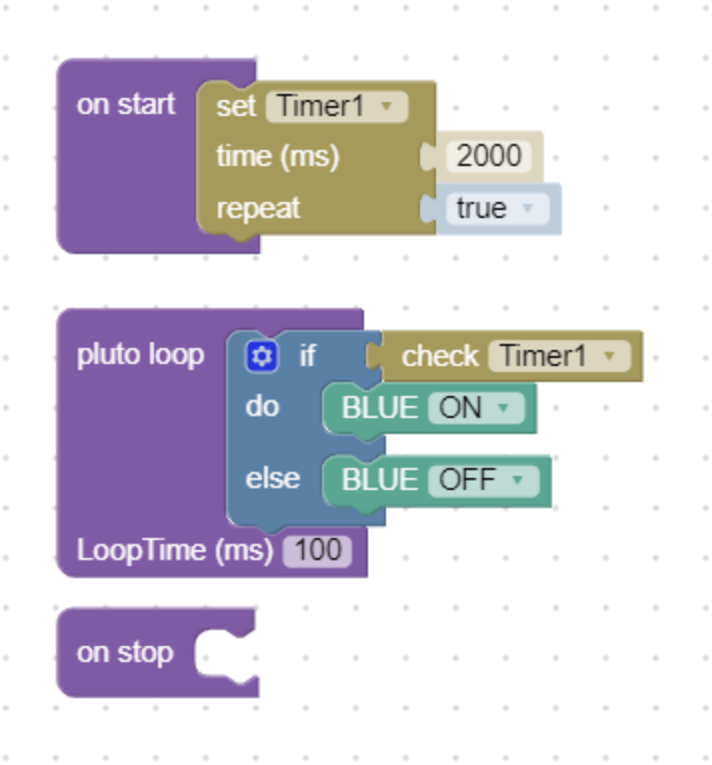
Toggle means to change the state. This makes the LED switch from ON to OFF and OFF to ON every two seconds. Try changing the time and understand how the timer block works.

Project 5

Challenge:

Build a program to blink an LED after every two seconds.

Solution:



Pro tip:

If the LED is not switched OFF, it will remain ON continuously and hence will not blink. The above program makes the LED turn ON every 2 seconds. Try changing the time to a very small number and see how fast the LED blinks!

Project 6

Challenge:

Build a program such that all the motors will get 1200 PWM when developer mode is turned ON, and all the motors will get 1000 PWM when developers mode is turned OFF.

Solution:

Pro tip:

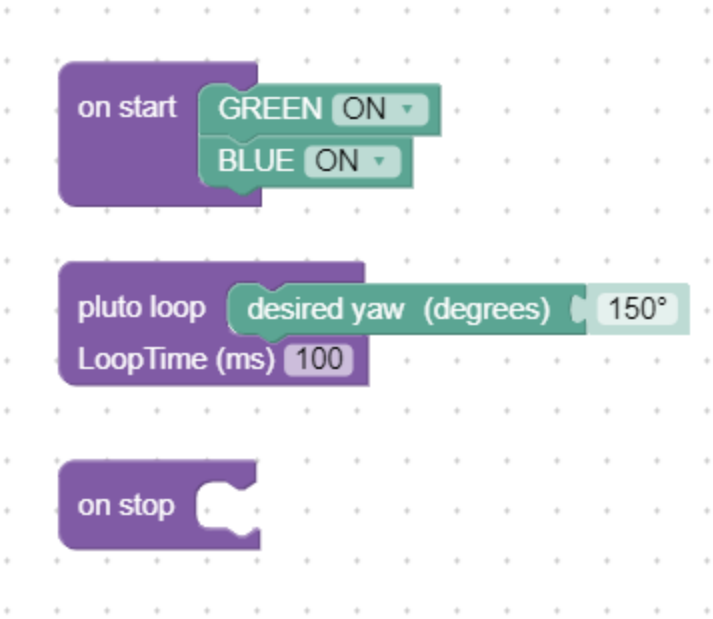
Whenever giving PWM to motors in a program, remember to shut them off during On Stop by giving 1000 PWM. This will ensure that the motors do not run after Developer Mode is turned OFF or the app is disconnected.

Project 7

Challenge:

Build a program such that Green and Blue LEDs will turn ON and the drone should Yaw to 150 degrees. (Hint: Use ‘desired yaw’ block)

Solution:



Pro tip:

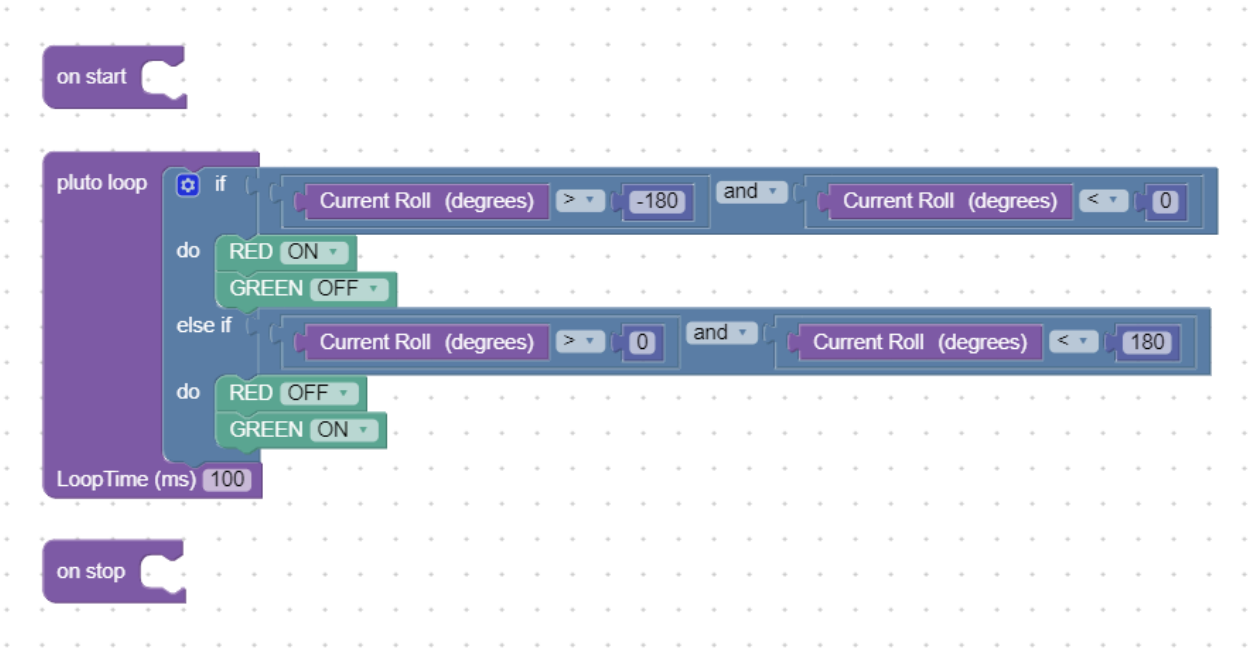
The execution of ‘desired state’ blocks can be tested only while flying the drone.

Project 8

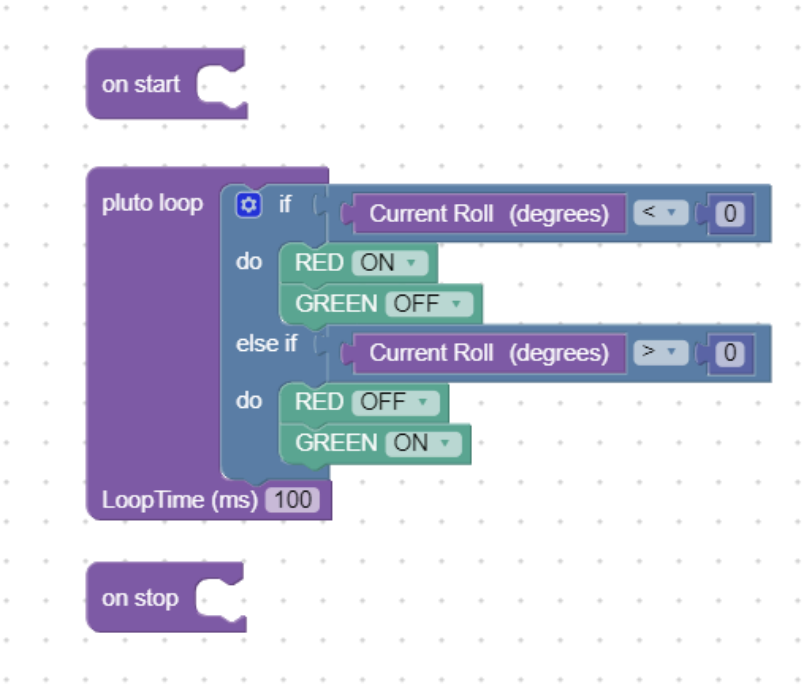
Challenge:

Build a program such that if the current roll angle is between -180 and 0 degrees then the Red LED will turn ON and if the current roll angle is between 0 and 180 degrees then the Green LED should turn ON.

Solution:



OR



Pro tip:

A single Challenge may have multiple Solutions. Try to find the most optimum Solution while programming.

Leveling Up

With the above projects, we hope that you have become familiar with the various types of blocks. If required, we suggest you try building a few projects on your own using different blocks to get a better understanding of Pluto Blocks.

Next in line are the projects which will help you level up. These projects can be used as - interesting activities, tools to understand scientific concepts and fun games as well! Let's level up!

Project 01: Color Wheel

As the drone yaws 360˚, it will generate different primary and secondary colors across different angles!

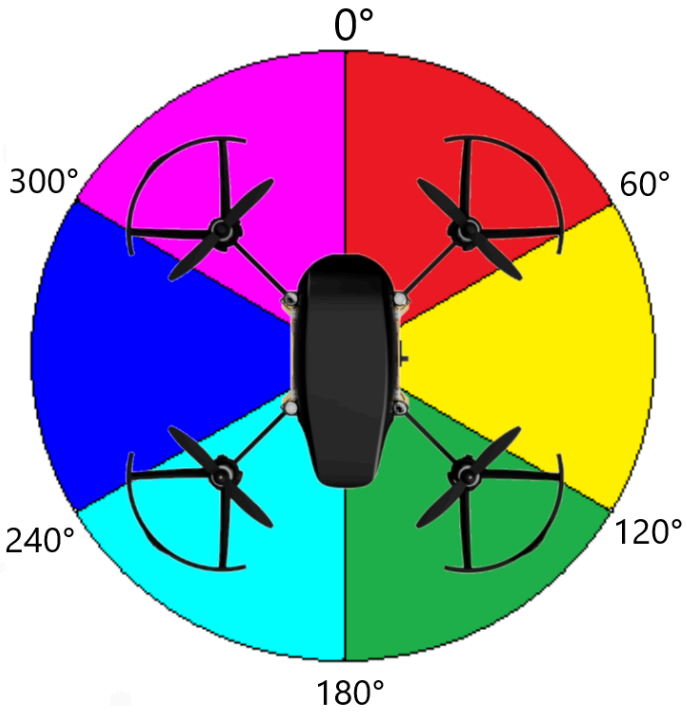
The Idea

You must have seen the color wheel showing different primary colors combining to form secondary colors. With this project, you will turn your Pluto into a color wheel and visually see how the primary colors blend in to produce a secondary color!

Approach

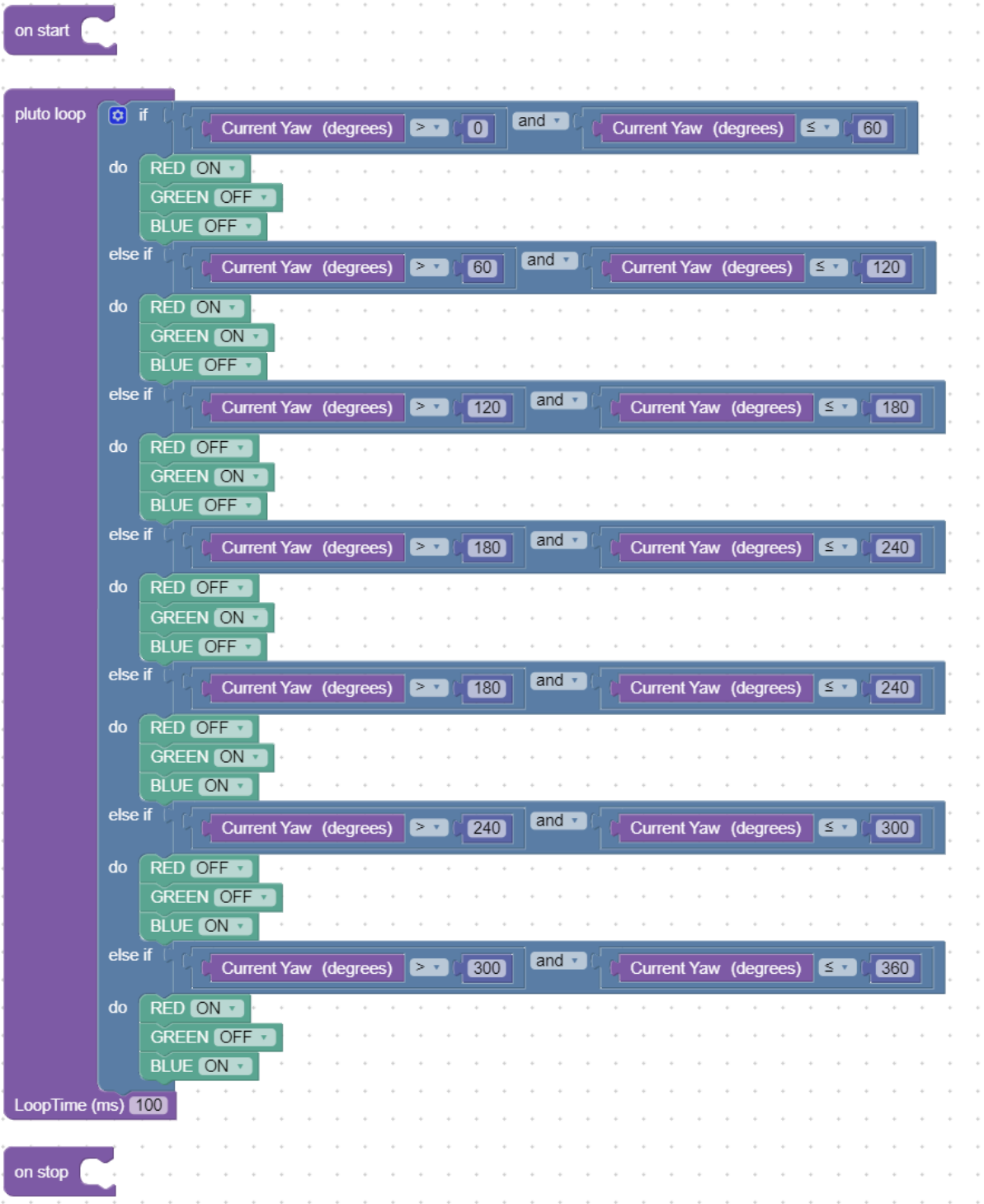
The Primus V4/Primus X Flight Controller contains three LEDs - Red, Green and Blue. These primary colors can be combined to form different secondary colors. At a given time, if two LEDs are turned ON, they will combine to produce a secondary color.

If you condition the LEDs to turn ON depending on the current angle of yaw of the drone, you will be able to use your Pluto as a color wheel. Refer to the below image and start coding!



Colors to be produced between different Yaw angles

Code


on start
pluto loop
if
Current Yaw (degrees) > 0 and Current Yaw (degrees) ≤ 60
do
RED ON
GREEN OFF
BLUE OFF
else if
Current Yaw (degrees) > 60 and Current Yaw (degrees) ≤ 120
do
RED ON
GREEN ON
BLUE OFF
else if
Current Yaw (degrees) > 120 and Current Yaw (degrees) ≤ 180
do
RED OFF
GREEN ON
BLUE OFF
else if
Current Yaw (degrees) > 180 and Current Yaw (degrees) ≤ 240
do
RED OFF
GREEN ON
BLUE OFF
else if
Current Yaw (degrees) > 180 and Current Yaw (degrees) ≤ 240
do
RED OFF
GREEN ON
BLUE ON
else if
Current Yaw (degrees) > 240 and Current Yaw (degrees) ≤ 300
do
RED OFF
GREEN OFF
BLUE ON
else if
Current Yaw (degrees) > 300 and Current Yaw (degrees) ≤ 360
do
RED ON
GREEN OFF
BLUE ON
LoopTime (ms) 100
on stop

Insights

- Magnetometer is the most important sensor for measuring Yaw angle of the drone
- If you do not get proper Yaw angles, then try Magnetometer Calibration from the Pluto Controller App
- Running this project will not require you to fly/arm the drone

Project 02: Compass Drone

Use your Pluto as a compass to point to the North, East, South and West directions!

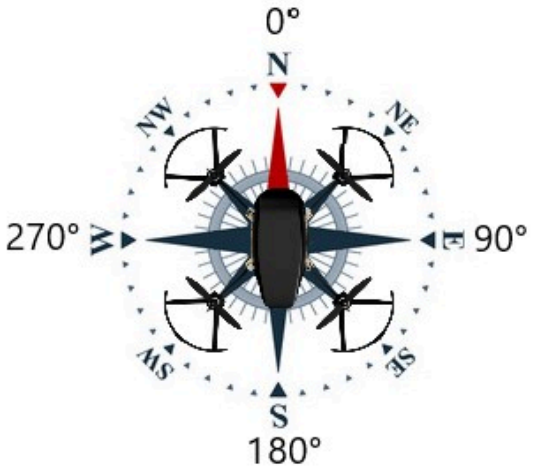
The Idea

Compasses have been used since ancient times to figure out directions. A typical compass points to the North direction. Using this project you can make your Pluto to point to North, East, South, West, or any other direction you want!

Approach

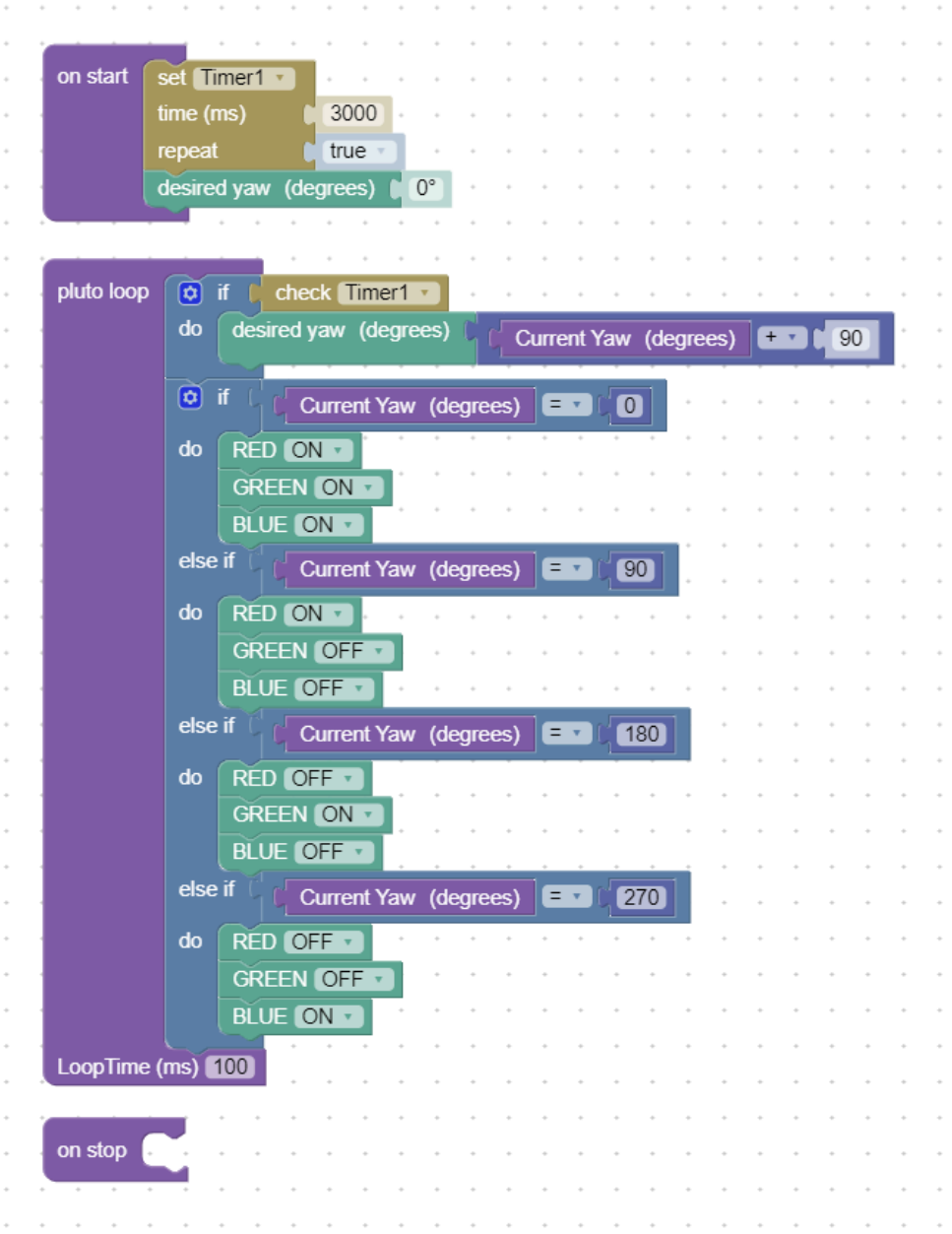
A properly Mag Calibrated drone when shows 0° as Yaw angle, it essentially points to the North direction. With every 90° added, it will change the direction. Thus, 90° will point to East, 180° will point to South and 270° will point to West.

If you set the desired Yaw angle of the drone to these angles, you will convert your Pluto into a compass! A suggestion, use different LEDs for different directions.



Directions with Yaw angles

Code



Activity

Along with a few friends, try to guess the directions in a room with no sunlight. Then, use this project to verify the directions. The one with the correct guess wins!

Project 03: Spirit Level

Upgrading from the traditional spirit level, use Pluto to detect if a surface is perfectly level!

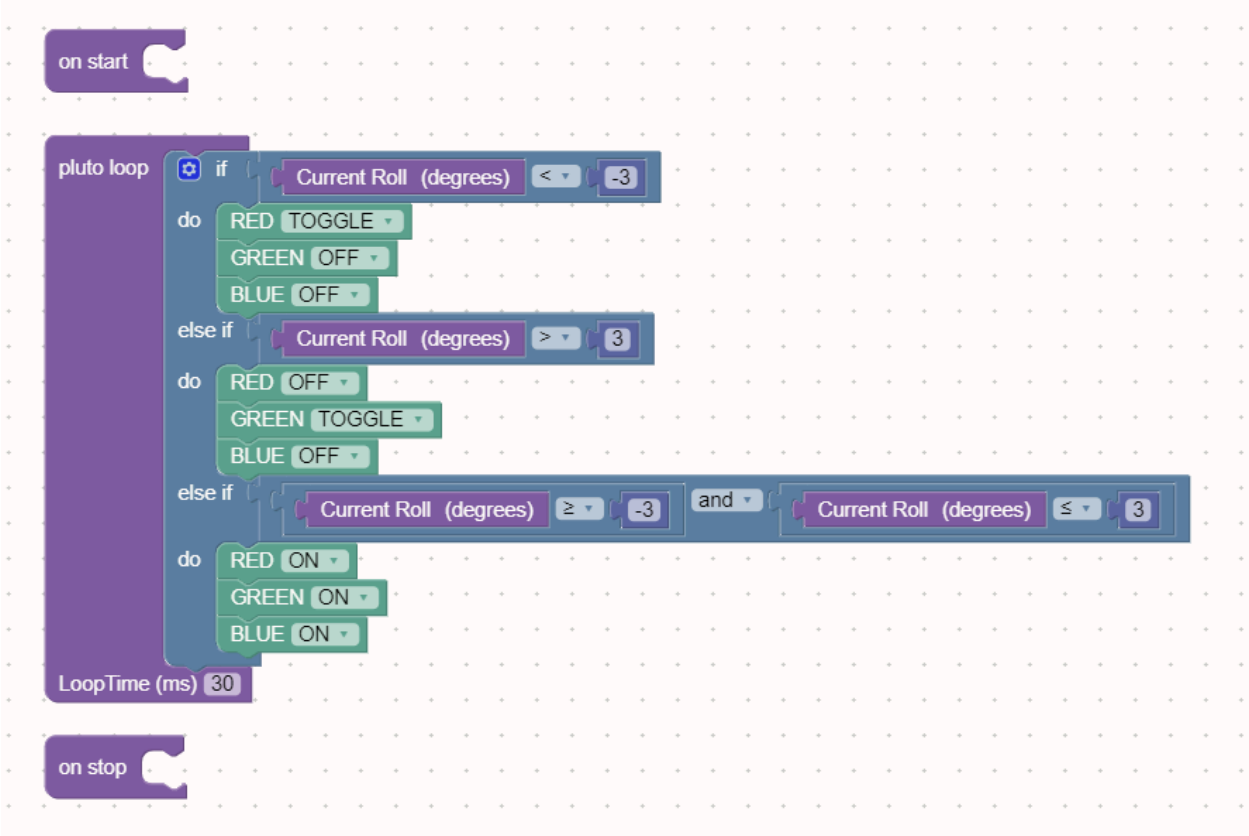
The Idea

Spirit Levels are used to check whether the surface is perfectly horizontal and vertical. Pluto enables you to replace this age old traditional tool and make the application more interesting by involving a drone.

Approach

The sensors give the current values of Roll, Pitch and Yaw angles. If we use this real time data of angles along with the on board LEDs, we can create our own customizable Spirit Level. Use the current Roll angle and program the LEDs to blink if the value of Roll angle is outside the given range. Change the loop time for better toggle frequency.

Solution



Insights

- By varying the loop time, you can change the frequency of LED toggle
- You can customize the accuracy of the Spirit Level by changing the range of angle

Activity

Use this project to check if all the components at your home are perfectly horizontal. Simply keep the drone on the surface to be tested. Blinking LED will indicate that the surface is not perfectly horizontal but White LED will indicate that it is perfectly horizontal.

Game

- Change the range of angle to ± 1 instead of ± 3
- Try to hold the done in your hand and maintain the white light.
- Whoever is able to hold the drone perfectly flat for the most amount of time is the winner!

Project 04: Truth or Dare

Now you can play this classic game with a drone instead of a bottle!

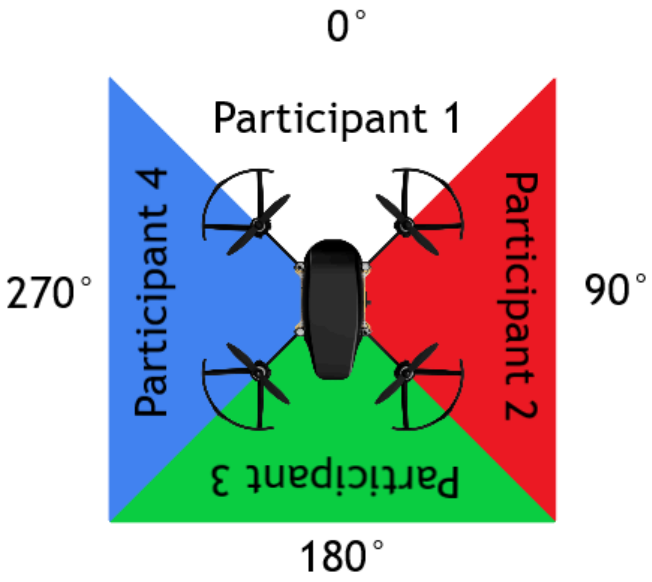
The Idea

This game originally uses a spinning bottle. With this project it will be replaced by a spinning drone! The spinning drone will point the participants and when it stops spinning, the participant to which it points will have to choose between Truth or Dare!

Approach

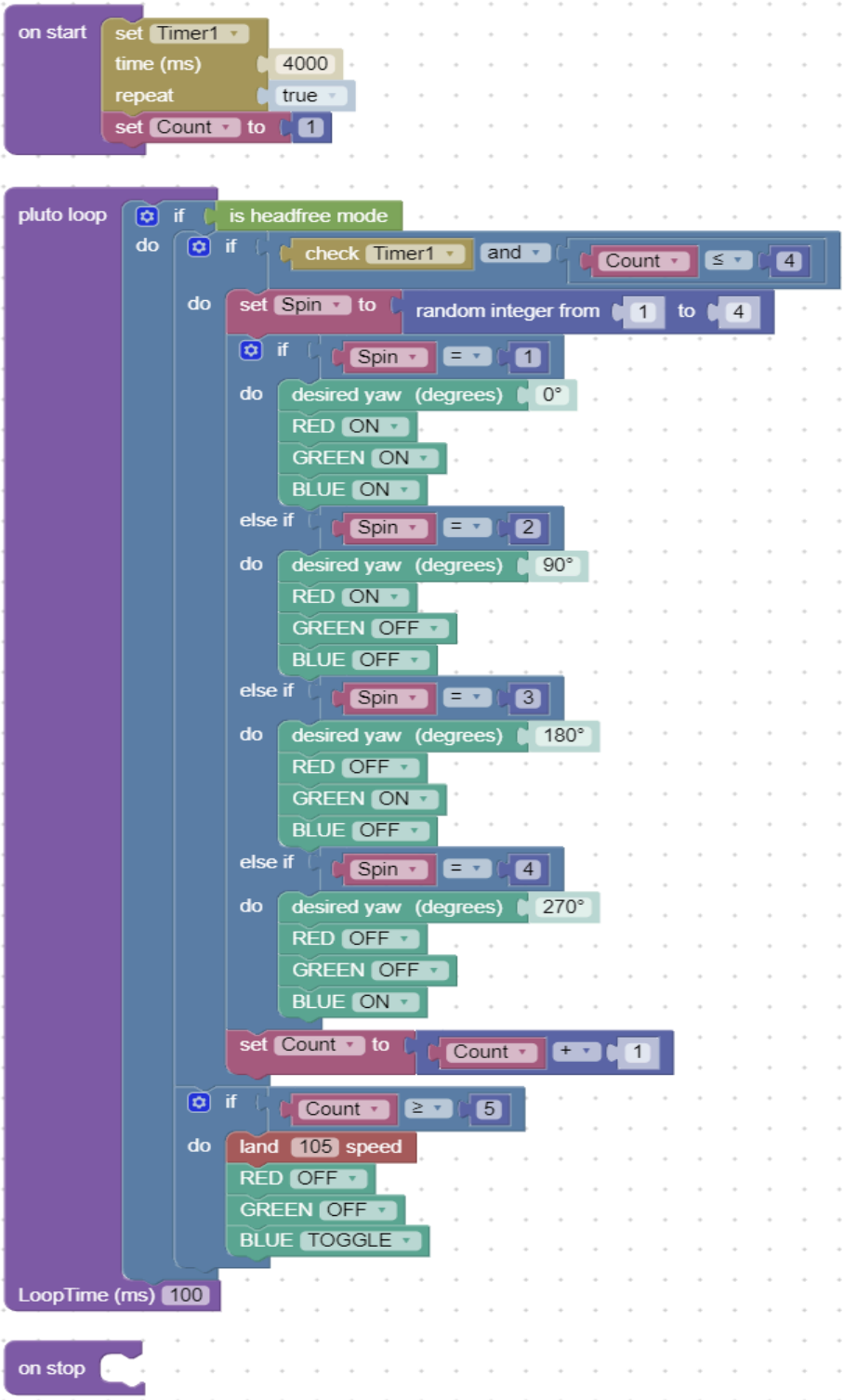
Pluto Blocks allows you to set the desired Yaw angle of the drone. Whatever angle you set, the drone will turn to that angle. Using this idea, we will program the drone to turn to a specific Yaw angle chosen at random. Use a timer block as well to ensure that the drone gets enough time to reach an angle before getting a new command to turn to a new angle.

Fly this project in Headfree mode to get better control over the drone.



Yaw angles for participants

Solution


on start
set Timer1
time (ms) 4000
repeat true
set Count to 1
pluto loop
if is headfree mode
do if check Timer1 and Count ≤ 4
do set Spin to random integer from 1 to 4
if Spin = 1
do desired yaw (degrees) 0°
RED ON
GREEN ON
BLUE ON
else if Spin = 2
do desired yaw (degrees) 90°
RED ON
GREEN OFF
BLUE OFF
else if Spin = 3
do desired yaw (degrees) 180°
RED OFF
GREEN ON
BLUE OFF
else if Spin = 4
do desired yaw (degrees) 270°
RED OFF
GREEN OFF
BLUE ON
set Count to Count + 1
if Count ≥ 5
do land 105 speed
RED OFF
GREEN OFF
BLUE TOGGLE
LoopTime (ms) 100
on stop

Game

Use this project to play Truth or Dare among 4 participants. Each participant can stay at 0°, 90°, 180° and 270° Yaw angle of the drone.

In the traditional game, the surprise element is the time for which the bottle spins. In this upgraded version, the surprise element is random spinning of the drone to select a participant. So start playing!

Activity

By making a few changes in the code, convert it so that it can be played by six players.

Project 05: Chuck to Arm

Arm your Pluto by simply tossing it in the air!

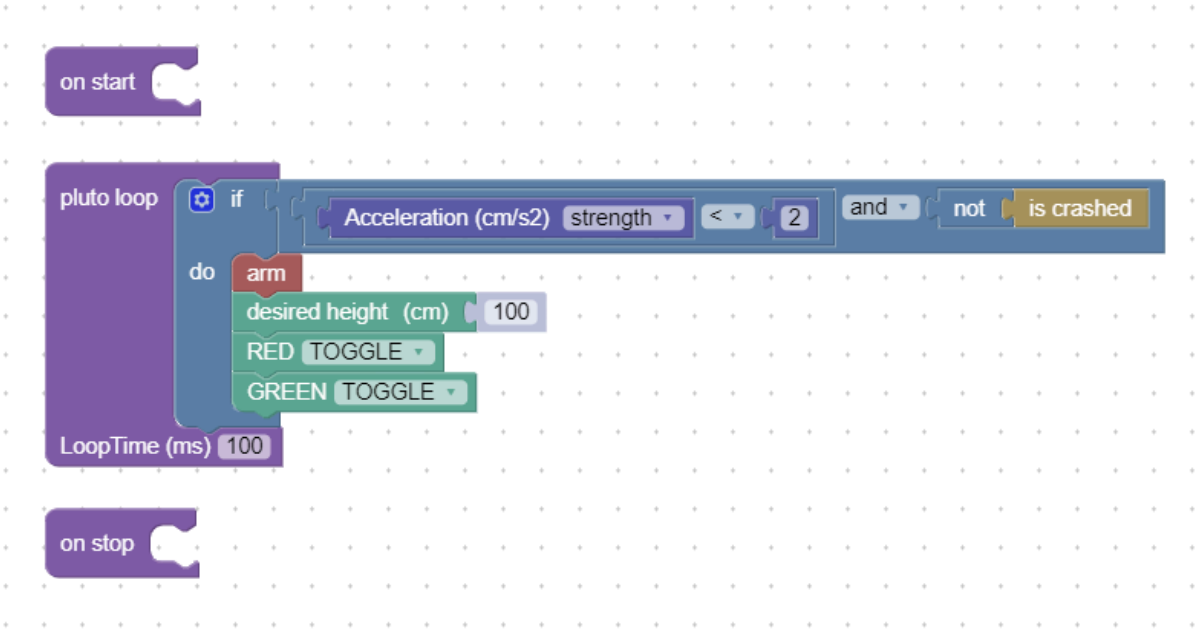
The Idea

Normally, you arm your drone by keeping it on the ground/floor. Instead of this typical method, you can now arm your drone by tossing it in the air.

Approach

When you toss the drone in the air, it needs to arm itself at that moment. Thus, you need to program the drone to detect that moment. When the drone is tossed in the air, the Accelerometer data should reach zero. If this value is not because of a crash, then the drone should arm itself. Remember to arm the drone from the app before running the project.

Solution



Project 06: Open Sesame

Blow air under Pluto to take off!

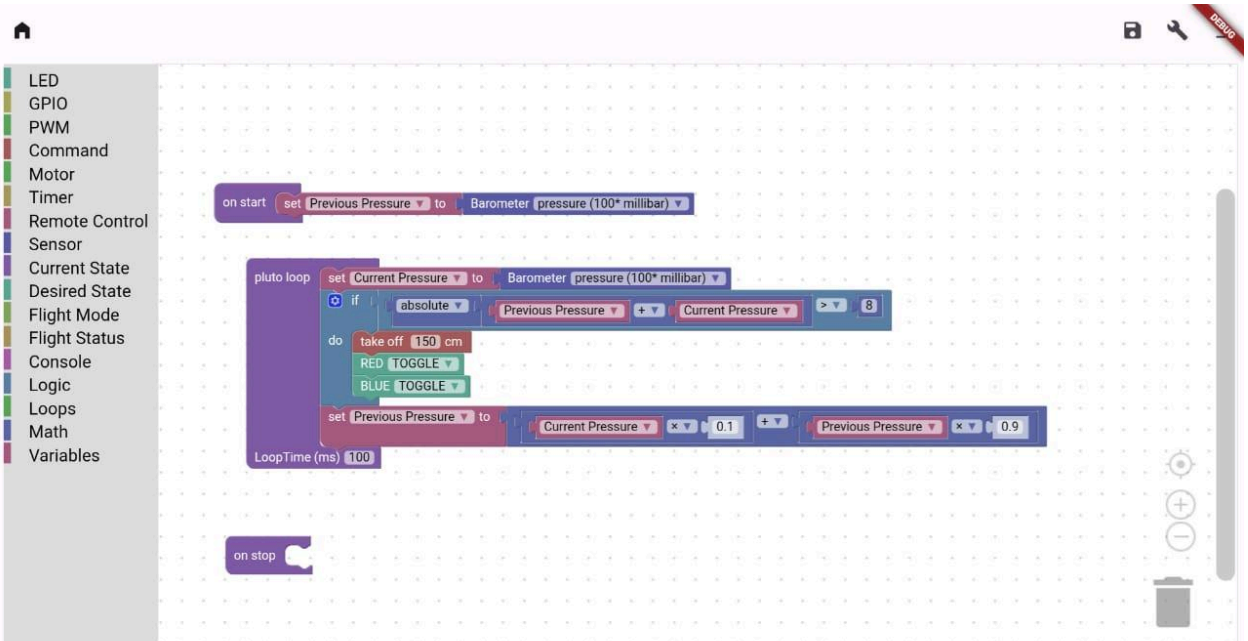
The Idea

Instead of using the take off button on the Pluto Controller App, you can use the Barometer sensor present on the Primus V4/ Primus X board to take off. Just blow some air on the drone and it will take off!

Approach

The barometer sensor measures the pressure in the air. If you blow air over it, the sensor senses it and for that instant, the pressure reading changes. This moment can be used to command the drone to take off. Remember to arm the drone from the app before running the project.

Solution



Activity

Keep the drone near a door and open/close the door at a fast speed to make the drone take off!

Project 07: Gyroscopic Drone

Experience the effect of the Gyroscope Sensor by holding Pluto in your hand!

The Idea

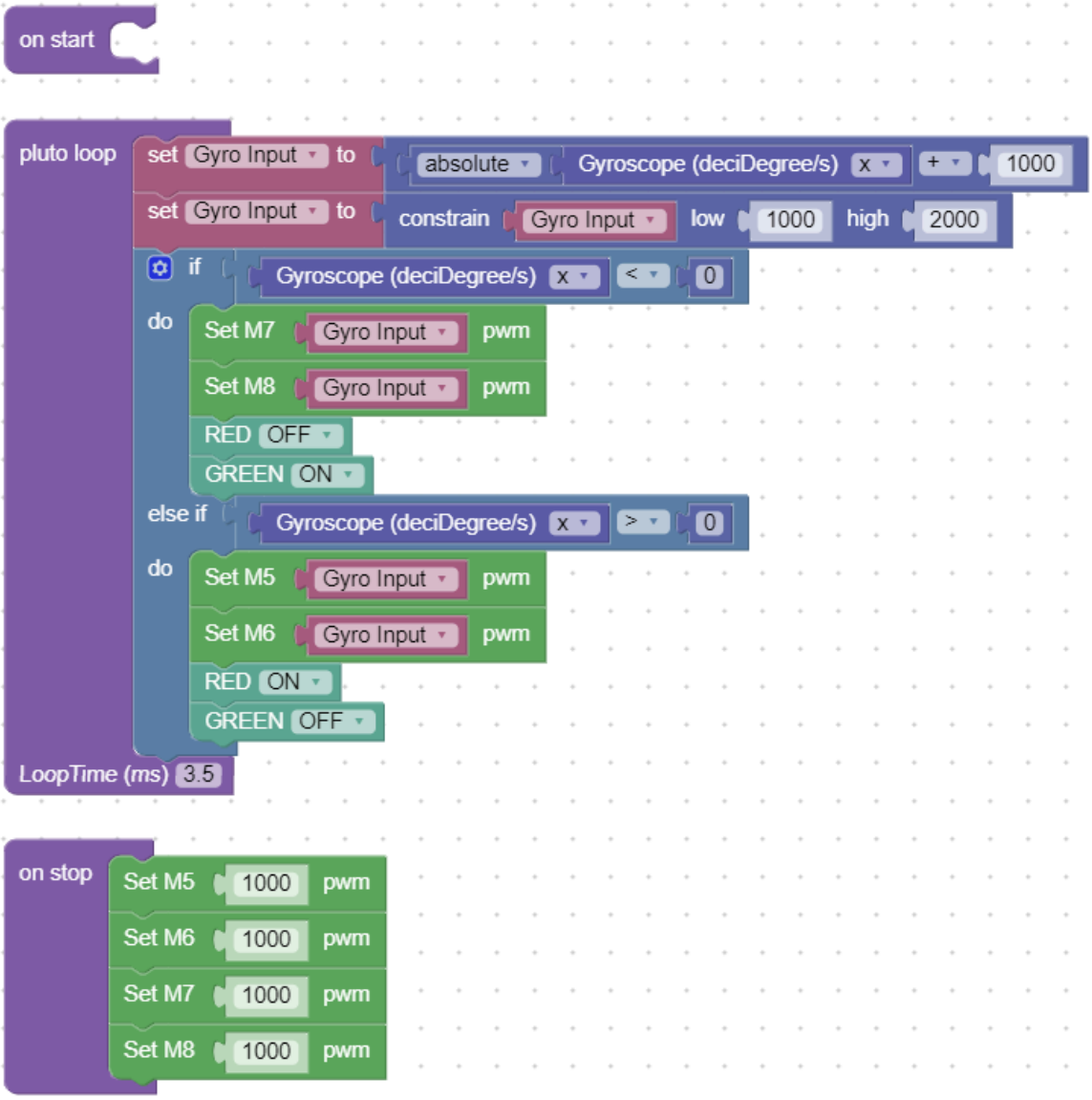
Gyroscope is one of the most important sensors found in applications ranging from smartphones to submarines and Hubble Telescope. Through this project, let us understand the working of the Gyroscope sensor by experiencing its effect by holding the drone in your hand.

Approach

In this project, we will use the data from only the Gyroscope sensor to power the motors. This will make the motors rotate depending on the data of the Gyroscope sensor. Positive values of the data should power one set of motors and the negative values should power the other set.

Moving Pluto to experience Gyroscope sensor

Solution



Activity

Try to change the inclination of Pluto about the roll axis without making the motors rotate. This will have to be done very slowly!

Project 08: Logic Drone

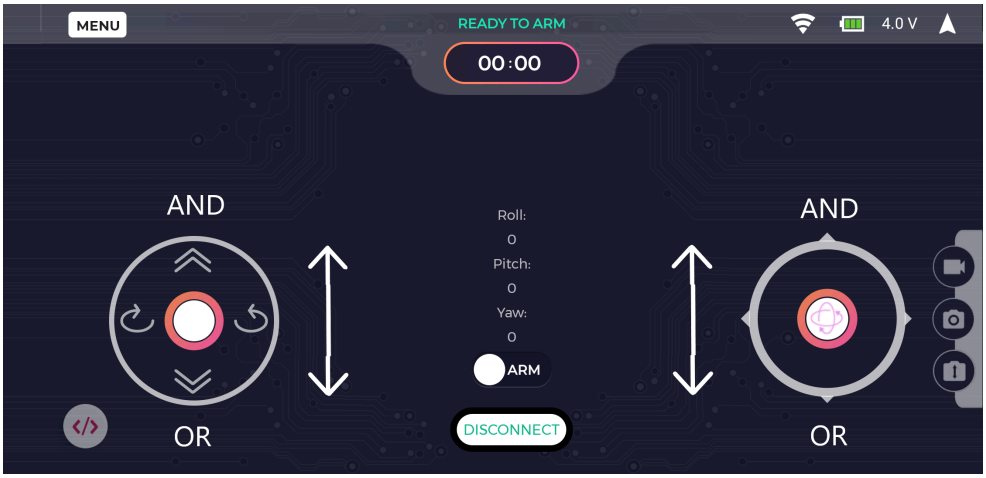
Understand the basics of Logic Gates using Pluto!

The Idea

Pluto Blocks contains the logical operators AND and OR, which are the basic logic gates. Using specific input and output, you can see how these logic gates work!

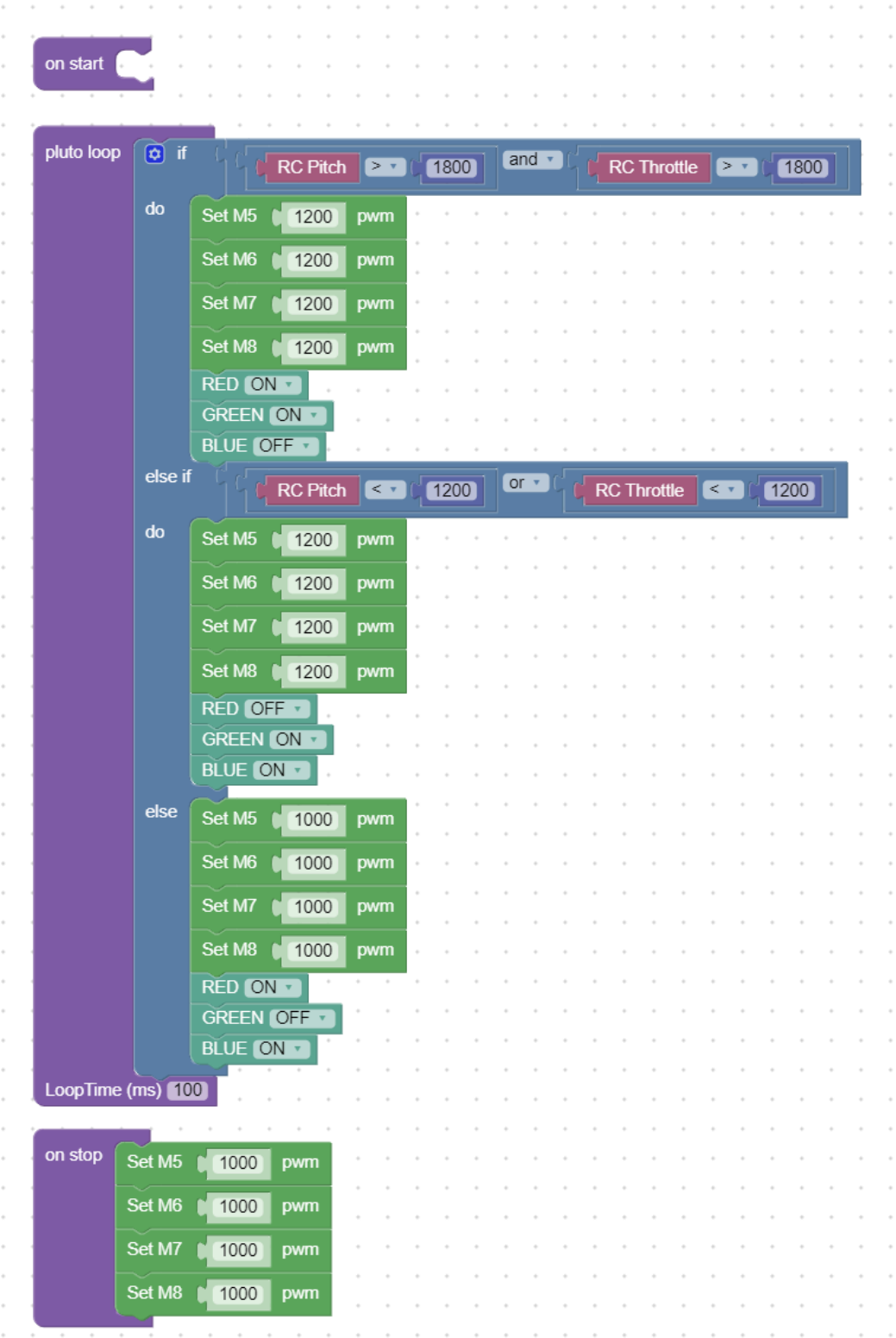
Approach

Using the logic blocks AND and OR, give the input from the app itself. Use the Throttle and Pitch sticks as input. For AND, use Throttle up and Pitch up and for OR, use Throttle down and Pitch down. For output, power all the four motors along with different LED combinations.



Inputs for AND and OR from Pluto Controller App

Solution


on start
pluto loop
if
RC Pitch > 1800 and RC Throttle > 1800
do
Set M5 1200 pwm
Set M6 1200 pwm
Set M7 1200 pwm
Set M8 1200 pwm
RED ON
GREEN ON
BLUE OFF
else if
RC Pitch < 1200 or RC Throttle < 1200
do
Set M5 1200 pwm
Set M6 1200 pwm
Set M7 1200 pwm
Set M8 1200 pwm
RED OFF
GREEN ON
BLUE ON
else
Set M5 1000 pwm
Set M6 1000 pwm
Set M7 1000 pwm
Set M8 1000 pwm
RED ON
GREEN OFF
BLUE ON
LoopTime (ms) 100
on stop
Set M5 1000 pwm
Set M6 1000 pwm
Set M7 1000 pwm
Set M8 1000 pwm

Project 09: One Key Return

Use Developer mode to call back the drone

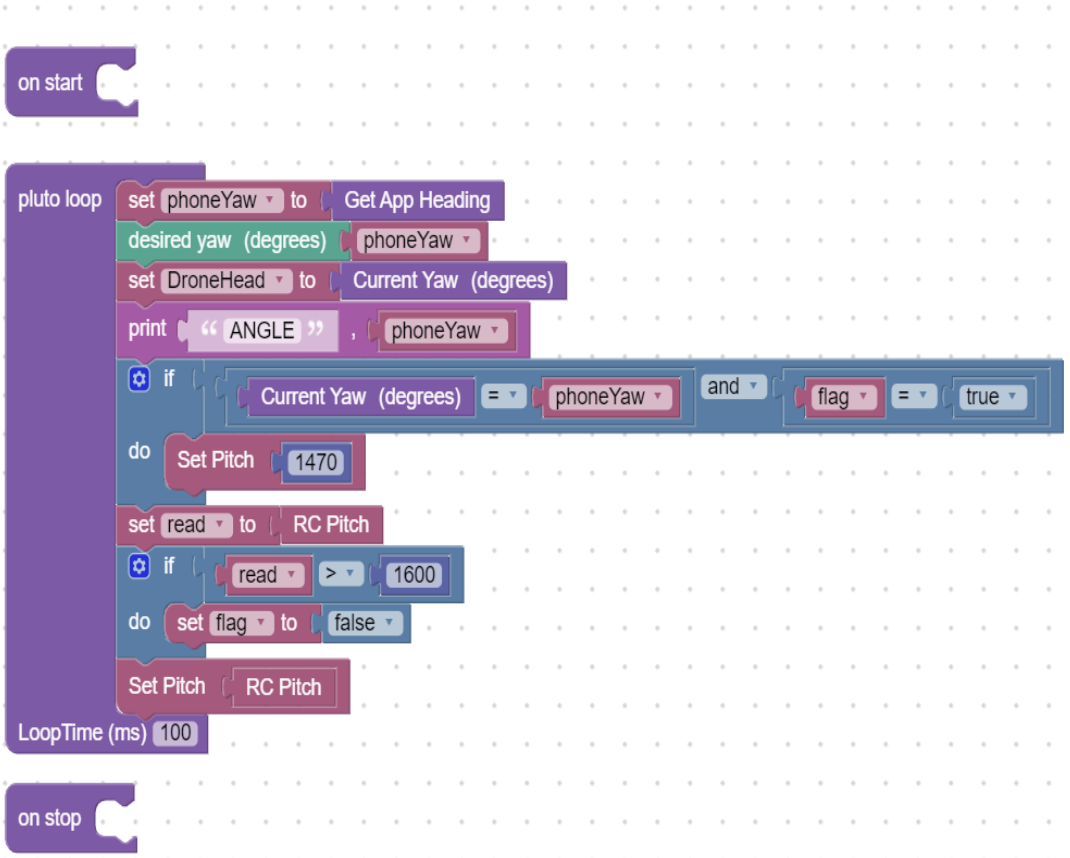
The idea:

To bring the drone back to the owner by tracking the phone heading. Without using GPS technology

Approach:

When the drone is running too far away from the pilot it can be called back. Thus you need to program the drone to come back to its pilot when the Developer mode is turned ON

Solution:



Note:

- Connect the drone with PlutoBlocks
- Turn ON the 'send heading' in the Developer settings

- Create Integer name:-
- 1. DroneHead
- 2. phoneYaw
- 3. Read
- Create Boolean:-
- 1. Flag

Project 10: LED Module

Introduction:

Pluto 1.2 and X, can accomadate one of the add-ons that is the LED module. This module allows the Primus 1.2 or X flight controller to use and control an external LED.

Objectives:

- Exploring additional functionalities of Pluto drones

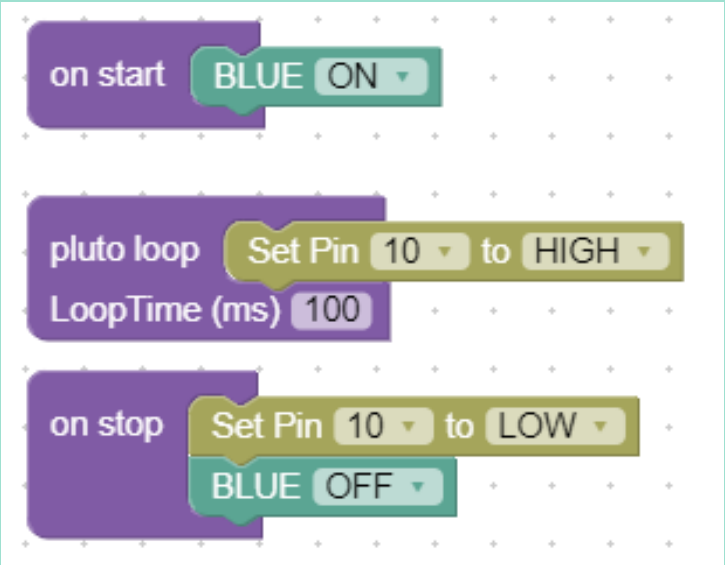
Problem Statement:

To use the LED module on the Pluto drone

Approaching the Problem:

For this project, we can make use of the Developer Mode button to turn on the LED, and turn off the LED by turning off the Developer Mode. This is just a simple project for the purpose of demonstration..

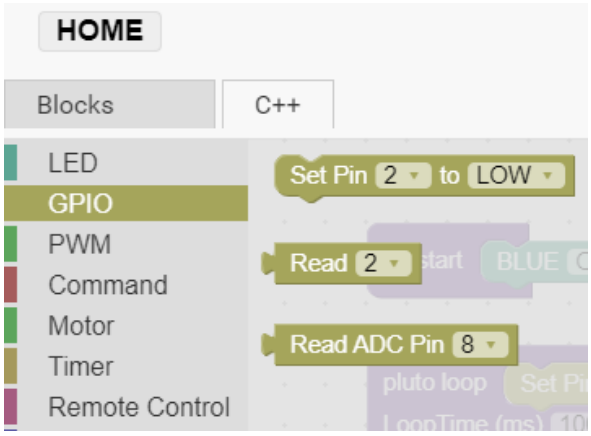
Solution:



Explanation of Solution:

On Start:
The on start code block executes when the Developer mode is turned on.
We turn on the in-built Blue LED on the Primus flight controller to indicate this

Pluto Loop:
We connect the LED model to the port (3-pin) on the front of Primus 1.2 or X. The GPIO pin designation of this port is 'Pin 10'. Thus, we provide the command to this port, using the 'Set PIN ___ to ___' block. This block can be found in the 'GPIO' block group.



To turn on the LED module we set pin 10 to HIGH.

On Stop:
The on stop code block executes when the Developer mode is turned off.
To turn on the LED module we set pin 10 to HIGH. We also turn off the Blue LED indicator, to signify that the developer mode is turned off.

Activity:

Have the LED module toggle at different time rates. Experiment around with the code blocks to create your own challenges. You can even transmit Morse Code with the help of the LED module! You just need to code for it.

Project 11: Sound Sensor

Pluto X, One of the add-ons is Sound Sensor. This is a type of sensor which detects a sound from the surrounding.

Objectives:

- Exploring additional functionalities of Pluto drones
- Learning about Sound Sensor.

Problem Statement:

- To use Sound Sensor to detect the Sound which is passing around Drone.

What is a Sound Sensor?

A sound sensor is defined as a module that detect sound waves through its intensity and converting it to electrical signals. The sound sensor consists of an in-built capacitive microphone, a peak detector, and an amplifier that's highly sensitive to sound.

Connection of Sound Sensor?

There are four terminal in sound sensor

GND: Negative terminal

VCC: Positive terminal

D0: Digital PIN

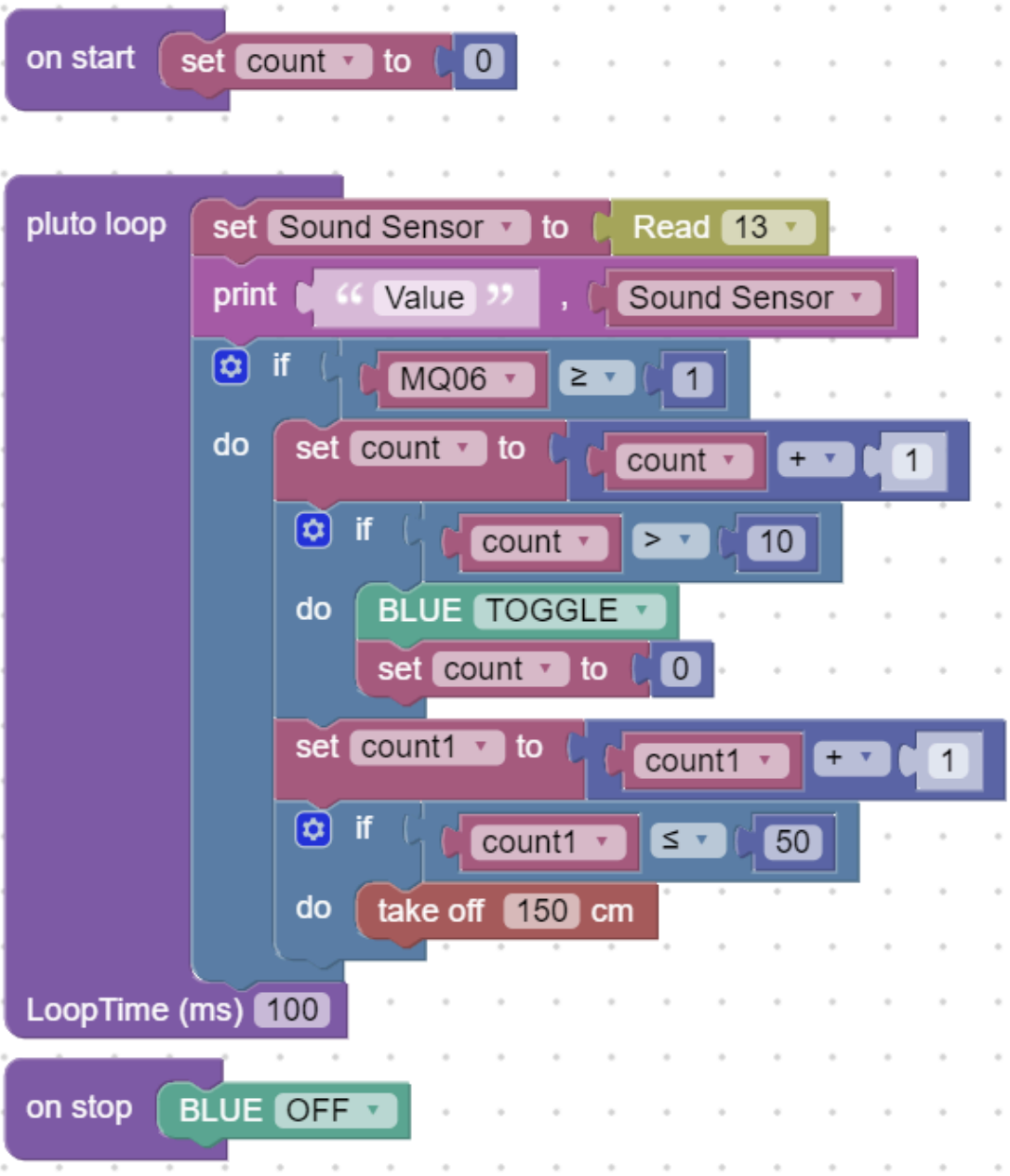
A0: Analog PIN

- Connect GND of sound sensor to GND of breakout
- Connect VCC of sound sensor to VCC of breakout
- Connect D0 of sound sensor to PIN number 13 of breakout

Approaching the Problem:

When Sound Sensor Detect the Sound Wave near the Drone then the Flight controller gives a command to Take OFF the Drone and the Blue LED toggle.

Solution:



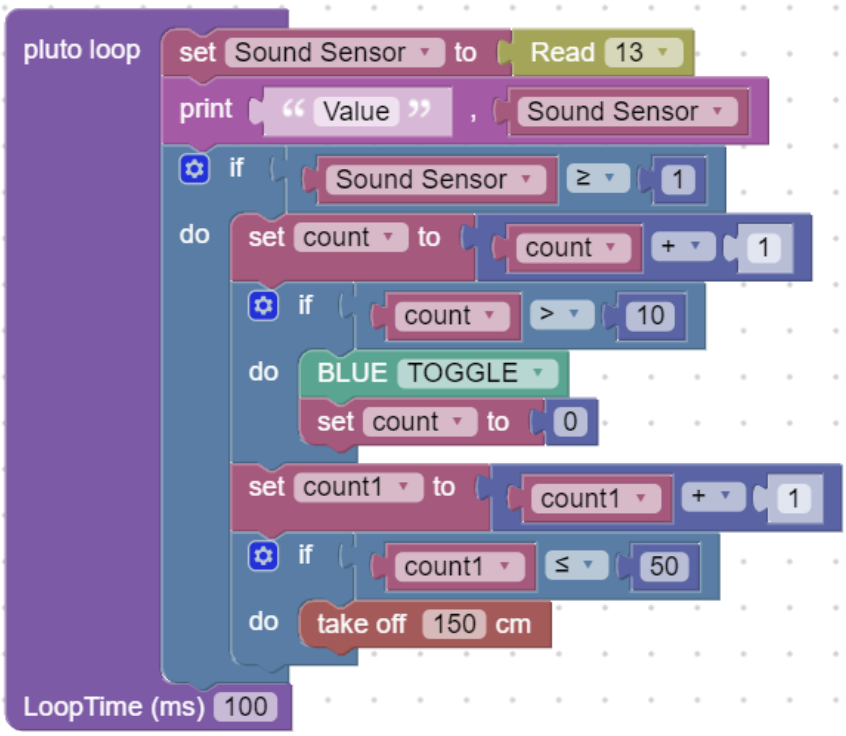
Explanation of Solution:

On Start Loop:

On Start Loop, we have declared a variable name count. The value of the variable count is 0.

Pluto Loop:

Declared Second Variable Sound Sensor and set PIN13. Display the value of sound sensor on console box. If Sound Sensor is greater and equal to then the count will Add by 1 so the value of the count is now counting = 1. Nested if, if the count is greater than 10 then BLUE will toggle and set count value = 0. Declared third variable count1 add by 1, so count1 = 1. If count1 is lesser than 50 then flight will Take OFF.



On Stop Loop: Blue LED will Turn OFF.

Project 12: ADC Read - MQ06 Sensor (PlutoX)

Introduction:

Pluto X has a valuable add-ons to help achieve tasks that would otherwise not be possible. One of these add-ons is the MQ06 Sensor. It is also known as a GAS Sensor. The MQ06 (LPG Gas Sensor) is used to detect the leakage of Gas from the LPG cylinder, etc.

Objectives:

- Exploring additional functionalities of Pluto drones
- Learning about MQ06 sensor.

Problem Statement:

- To use MQ06 to detect the leakage of gas by passing around the Drone and to use ADC pins

What is an MQ06 Sensor?

MQ06 is a kind of sensor that Detect the leakage of gas that flows in the air or around the sensor. These sensors are mostly used in industry. This sensor helps to alert worker who works in the industry and many other fields that use this sensor.

Connection of MQ06 Sensor?

There are Four terminals present in the sensor:

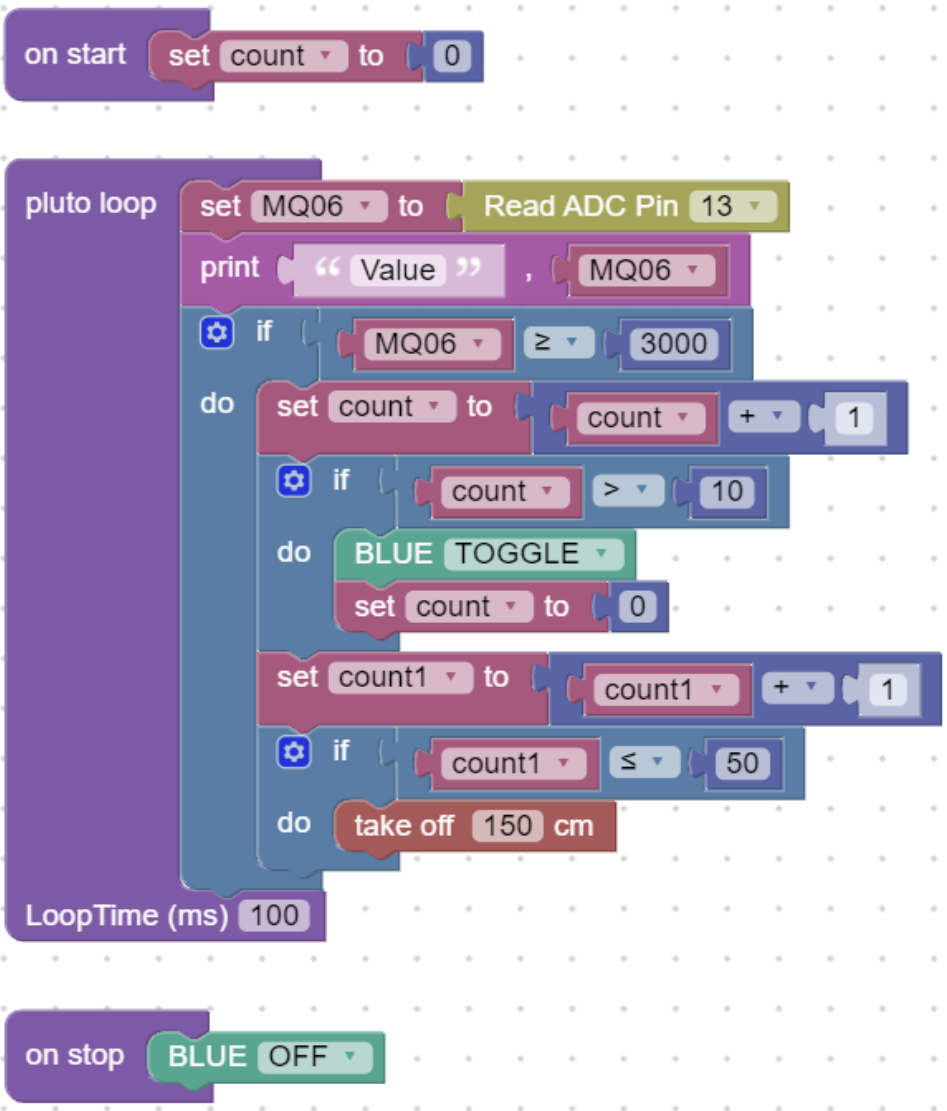
VCC: Positive Terminal, GND: Negative Terminal, D0: Digital Pin, A0: Analog Pin

- Connect a Breakout board on Pluto X.
- Connect VCC of the sensor to VCC of Breakout board.
- Connect the GND of the sensor to the GND of the Breakout board.
- Connect the A0 terminal of the sensor to the ADC pin of the Breakout board.

Approaching the Problem:

When LPG Gas is Leak around the Drone then the MQ06 sensor will detect Gas and Give the command to the flight controller for Taking OFF and Blue Light Toggle.

Solution:


on start
set count to 0
pluto loop
set MQ06 to Read ADC Pin 13
print " Value " , MQ06
if MQ06 ≥ 3000
do
set count to count + 1
if count > 10
do
BLUE TOGGLE
set count to 0
set count1 to count1 + 1
if count1 ≤ 50
do
take off 150 cm
LoopTime (ms) 100
on stop
BLUE OFF

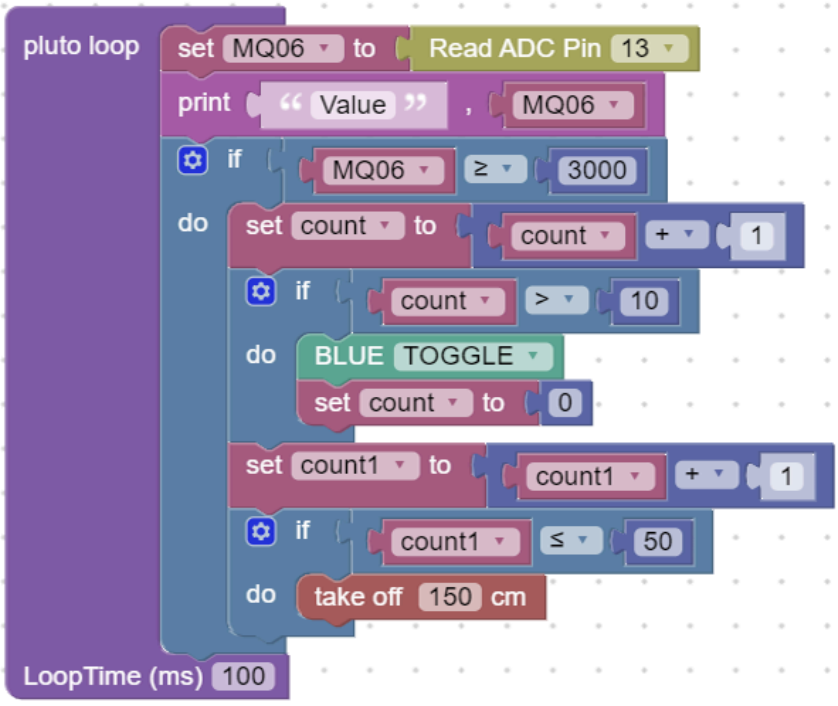
Explanation of Solution:

On Start Loop:

On Start Loop, we have declared a variable name count. The value of the variable count is 0.

Pluto Loop:

In Pluto Loop declared second variable MQ06 and Set Pin 13. Then display the value of the MQ06 sensor in the console box. If the MQ06 sensor value is greater than 3000 then the count will be +1, Value of the count will become 1. If the count is greater than 10 then Blue light will toggle. And the value of the count will become 0. Then 3 variable count1 = 0+1. Then the value of count1 will become 1. If count1 is lesser and equal to 50, Then the Drone will Take OFF.



On Stop Loop: Blue LED will Turn OFF.

Project 13: Gripper Module

Introduction:

Pluto 1.2 and X has one of the add-ons that is the Gripper module. The Gripper module consists of a servo motor (with servo horn attached), along with two gripping claws. The claws are linked to each other with a pair of spur gears (the base of each claw is composed of a gear). One of the claws is directly driven by the servo motor. When the servo motor is powered, it rotates the motor shaft and servo horn, moving one of the claws. This claw in turn moves the other claw (due to the gear linkage). Thus, the gripper can either open or close in order to clasp objects.

Objectives:

- Exploring additional functionalities of Pluto drones
- Learning about servo motors

Problem Statement:

To use the Gripper module to pick and release objects with the Pluto drone

What is a Servo Motor?

A servo motor is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision. If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor which runs through a servo mechanism. If the motor is powered by a DC power supply then it is called DC servo motor, and if it is AC-powered motor then it is called AC servo motor. We will be discussing the working of DC servo motors. Apart from these major classifications, there are many other types of servo motors based on the type of gear arrangement and operating characteristics. A servo motor usually comes with a gear arrangement that allows us to get a very high torque servo motor in small and lightweight packages. Due to these features, they are being used in many applications like toy cars, RC helicopters and planes, Robotics, etc.
Servo motors are rated in kg/cm (kilogram per centimeter) most hobby servo motors are rated at 3kg/cm or 6kg/cm or 12kg/cm. This kg/cm tells you how much weight your servo motor can lift at a particular distance. For example: A 6kg/cm Servo motor should be able to lift 6kg if the load is suspended 1cm away from the motors shaft, the greater the distance the lesser the weight carrying capacity. The position of a servo motor is decided by electrical pulse and its circuitry is placed beside the motor.

Servo Motor Working Mechanism:

It consists of three parts:

1. Controlled device
2. Output sensor
3. Feedback system

It is a closed-loop system where it uses a positive feedback system to control motion and the final position of the shaft. Here the device is controlled by a feedback signal generated by comparing output signal and reference input signal.

Reference input signal is compared to the reference output signal and the third signal is produced by the feedback system. And this third signal acts as an input signal to control the device. This signal is present as long as the feedback signal is generated or there is a difference between the reference input signal and reference output signal. So the main task of servo mechanism is to maintain the output of a system at the desired value in the presence of noises.

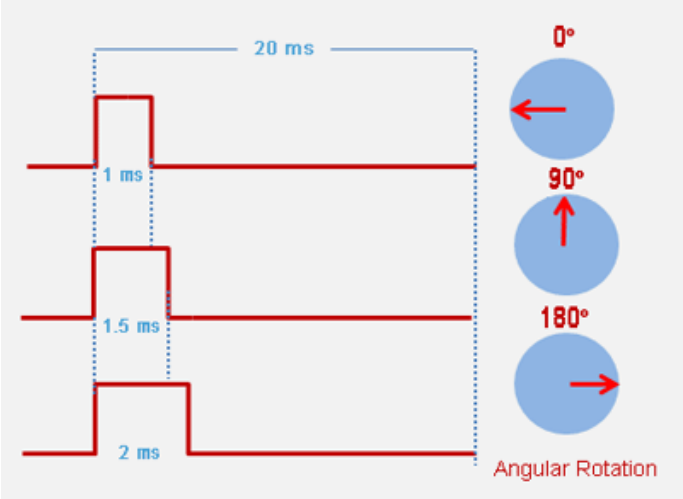
Servo Motor Working Principle:

A servo consists of a Motor (DC or AC), a potentiometer, gear assembly, and a controlling circuit. First of all, we use gear assembly to reduce RPM and to increase torque of the motor. Say, at the initial position of the servo motor shaft, the position of the potentiometer knob is such that there is no electrical signal generated at the output port of the potentiometer. Now an electrical signal is given to another input terminal of the error detector amplifier. Now the difference between these two signals, one comes from the potentiometer and another comes from other sources, will be processed in a feedback mechanism and output will be provided in terms of error signal. This error signal acts as the input for the motor and motor starts rotating. Now the motor shaft is connected with the potentiometer and as the motor rotates so does the potentiometer and it will generate a signal. So as the potentiometer's angular position changes, its output feedback signal changes. After sometime the position of potentiometer reaches a position where the output of potentiometer is the same as the external signal provided. At this condition, there will be no output signal from the amplifier to the motor input as there is no difference between external applied signal and the signal generated at potentiometer, and in this situation the motor stops rotating.

Controlling Servo Motor:

All motors have three wires coming out of them. Out of which two will be used for Supply (positive and negative) and one will be used for the signal that is to be sent.

The servo motor is controlled by PWM (Pulse width Modulation) which is provided by the control wires. There is a minimum pulse, a maximum pulse and a repetition rate. Servo motor can turn 90 degrees in either direction from its neutral position. The servo motor expects to see a pulse every 20 milliseconds (ms) and the length of the pulse will determine how far the motor turns. For example, a 1.5ms pulse will make the motor turn to the 90° position, such as if the pulse is shorter than 1.5ms the shaft moves to 0° and if it is longer than 1.5ms then it will turn the servo to 180°.
Servo motor works on PWM (Pulse width modulation) principle, meaning its angle of rotation is controlled by the duration of applied pulse to its Control PIN. Basically a servo motor is made up of a DC motor which is controlled by a variable resistor (potentiometer) and some gears. High speed force of the DC motor is converted into torque by Gears. We know that WORK= FORCE X DISTANCE, in DC motor Force is less and distance (speed) is high and in Servo, force is High and distance is less. The potentiometer is connected to the output shaft of the Servo, to calculate the angle and stop the DC motor on the required angle.

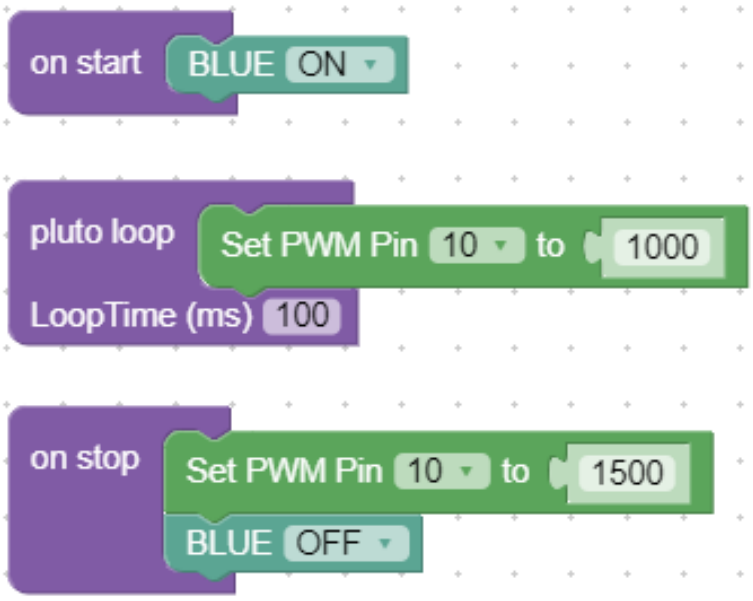


Servo motors can be rotated from 0 to 180 degrees, but it can go up to 210 degrees, depending on the manufacturing. This degree of rotation can be controlled by applying the Electrical Pulse of proper width to its Control pin.

Approaching the Problem:

We can make use of the Developer Mode button to control the gripper. We can thus prepare a simple code, such that the gripper closes (to grip objects) when we turn on Developer mode, and opens (to release objects) when we turn off the Developer mode.

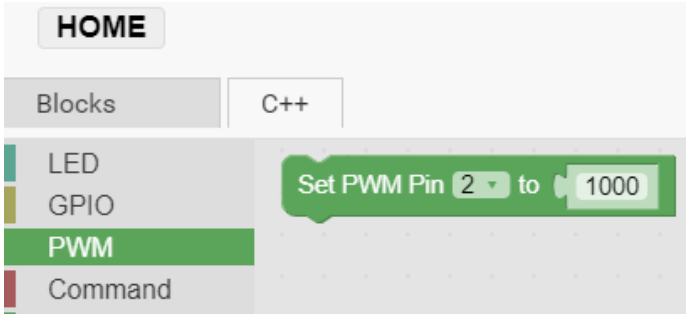
Solution:



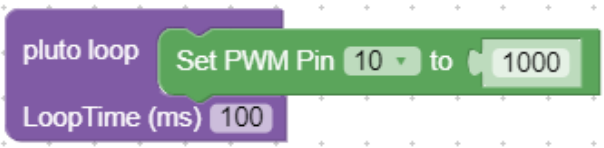
Explanation of Solution:

On Start:
The on start code block executes when the Developer mode is turned on.
We turn on the in-built Blue LED on the Primus flight controller when we turn on the Developer Mode. This will indicate that the gripper will be in the closed, or grasping position.

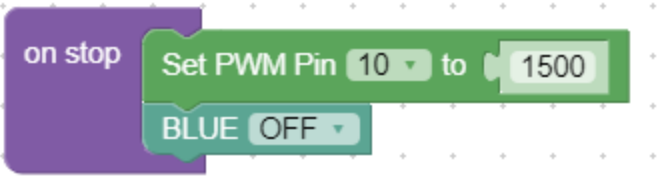
Pluto Loop:
We connect the gripper model to the port (3-pin) on the front of Primus 1.2 or X. The GPIO pin designation of this port is 'Pin 10'. Thus, we provide input to this port, using the 'Set PWM Pin' block. This block can be found in the 'PWM' block group.



To set the position of the servo motor shaft (and thus the whole gripper), we have to provide it with a PWM value. For the closed position, we are to provide the servo motor with a low PWM value of '1000'.



On Stop:
The on stop code block executes when the Developer mode is turned off.
In order to open the gripper (to release objects) we provide the servo motor with a higher PWM value of '1500'.
We also turn off the Blue LED indicator, to signify that the gripper is in the open or released position.



Gripper Photos