FOUNDATION

LESSON PLAN Contents
for Teaching and Learning

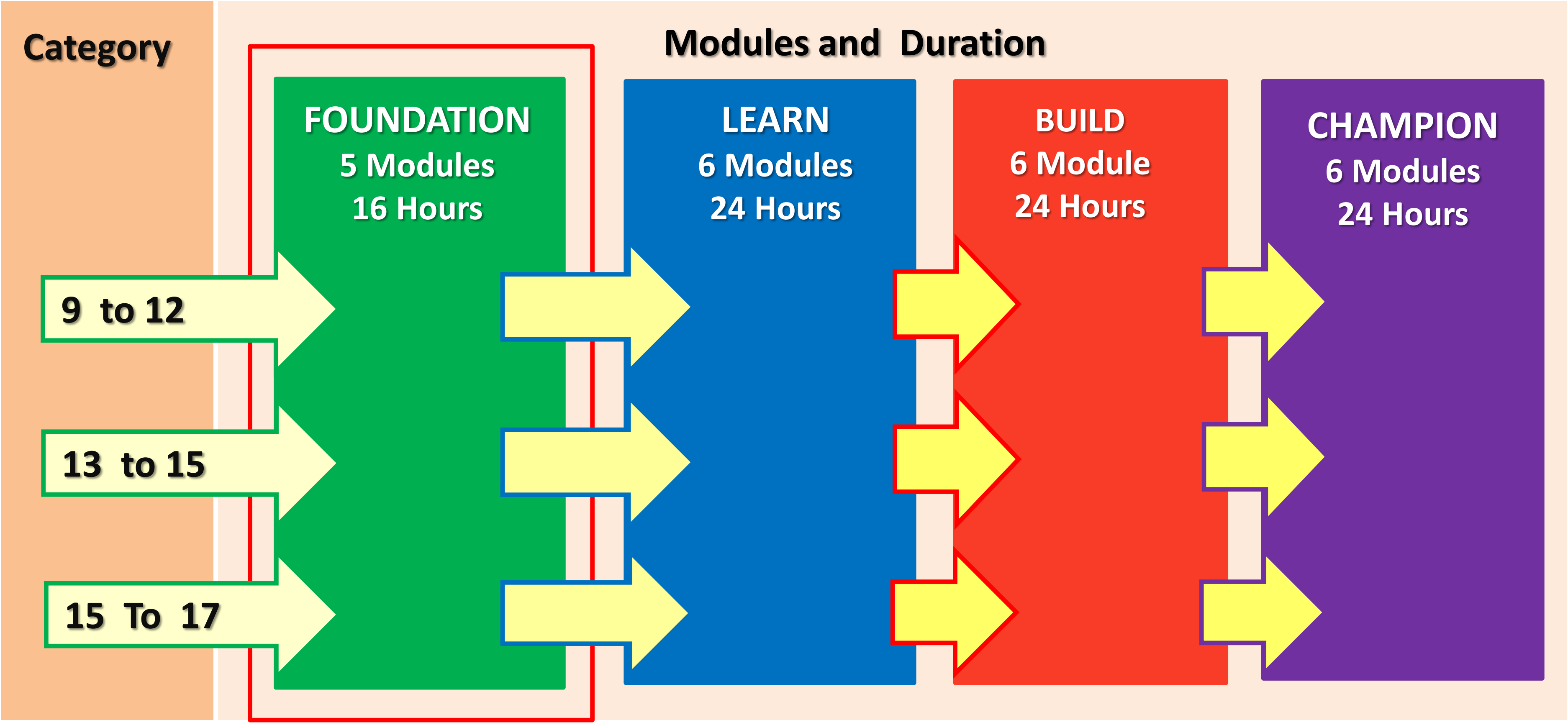
Category : 9 to 15

Flowchart programming
Using FlowLogic 6
and Introduction to
ARDUINO UNO

Mahsat acknowledge that there may be errors or omissions in this publication for which responsibility cannot be assumed. No liability will be accepted for loss or damage resulting from the use of information contained in this documentation or from uses as described.

myFlowlab™
www.myflowlab.com

Digital STEM Programs and Duration



Category

Modules and Duration

FOUNDATION
5 Modules
16 Hours

LEARN
6 Modules
24 Hours

BUILD
6 Module
24 Hours

CHAMPION
6 Modules
24 Hours

9 to 12

13 to 15

15 To 17

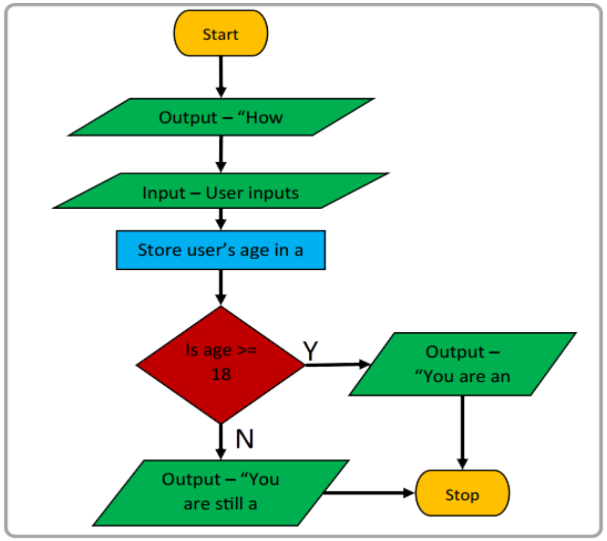
myFlowlab™

Module #1

Algorithm

What is Flowchart programming and Why

A flowchart programming is a arrangement of a type of diagram that represents an algorithm, workflow or process using Flowchart blocks.



What is Algorithm

An **algorithm** is a step by step method of solving a problem or task.

Why Flowchart

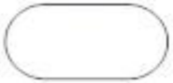
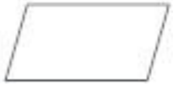
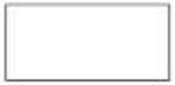
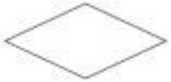
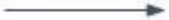
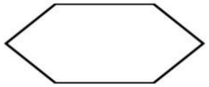
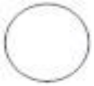
Flowcharts are used in analyzing, designing, documenting or managing a process in various fields.

myFlowLab™ STEM Education uses Flowchart programming tool FLOWLOGIC 6 to learn computational thinking by using Flowchart blocks to construct FlowProgram {algorithm] by:-

1. Developing Console applications
2. Controlling and analyzing built-in On scree mimic model (virtual projects)
3. Controlling, Monitoring, analyzing real-world applications using Arduino UNO board

Flowchart Blocks to develop Algorithm

myFlowlab™

	Terminal(Stop/Start)	Used to represent start and end of flowchart.
	Input / Output	Used for input and output operation.
	Processing	Used for arithmetic operations and data-manipulations.
	Decision	Used to represent the operation in which there are two alternatives, true and false.
	Flow line	Used to indicate the flow of logic by connecting symbols.
	Predefined Process/Function	Used to represent a group of statements performing one processing task.
	On-page Connector	Used to join different flow line

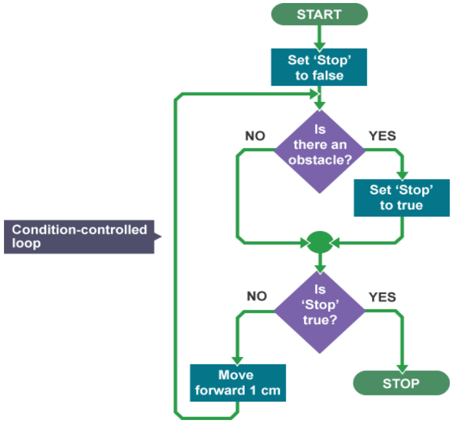
Algorithm VS Computer Program

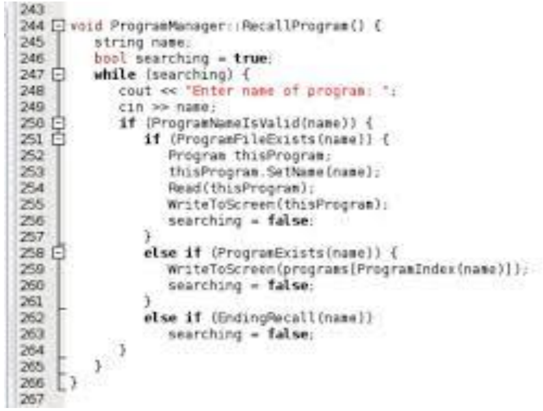
inputs
the ingredients
outputs
the results

Processes

Algorithm

Computer Program





```
void ProgramManager::RecallProgram() {
    string name;
    bool searching = true;
    while (searching) {
        cout << "Enter name of program: ";
        cin >> name;
        if (ProgramNameIsValid(name)) {
            if (ProgramFileExists(name)) {
                Program thisProgram;
                thisProgram.SetName(name);
                Read(thisProgram);
                WriteToScreen(thisProgram);
                searching = false;
            }
            else if (ProgramExists(name)) {
                WriteToScreen(programs[ProgramIndex(name)]);
                searching = false;
            }
            else if (EndingRecall(name))
                searching = false;
        }
    }
}
```

Algorithm design

An algorithm design is logical flow or sequence of instructions
That must accomplish a result

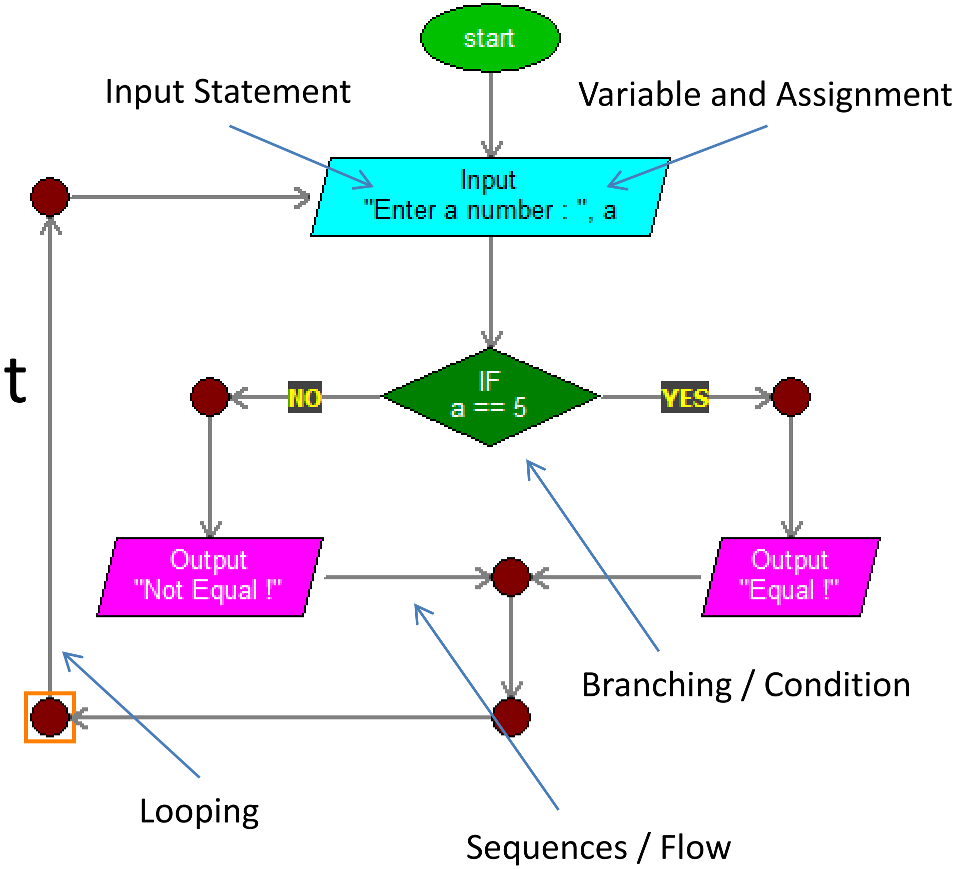
Important points:

1. *Instructions must be well ordered*
2. *Instructions must not be the same*
3. *The process must eventually end or loop*
4. *The actions must be doable*
5. *The algorithm must produce the required result.*

Core elements of an algorithm

Algorithms have some subset of the following critical elements:

1. Input Statement
2. Output Statement
3. Variable and Assignment
4. Branching / Condition
5. Sequences / Flow
6. Looping



Input Statement

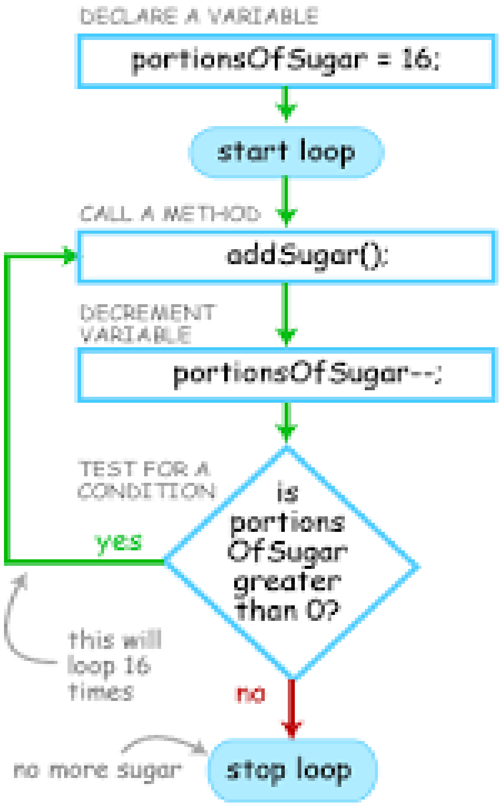
Variable and Assignment

Branching / Condition

Looping

Sequences / Flow

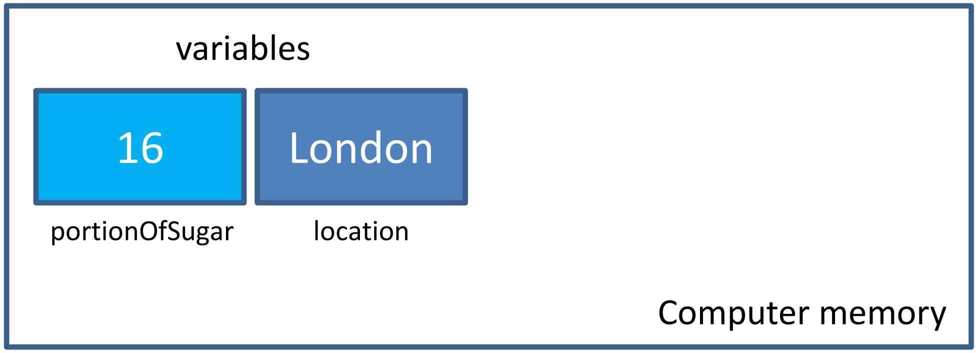
Variable



What is Variable ?

Variables are used to store information

Variables are like containers that hold information. It must be given a unique name and assigned a data. This data can then be used throughout your program by referencing to variable name.



variables

16	London
portionOfSugar	location

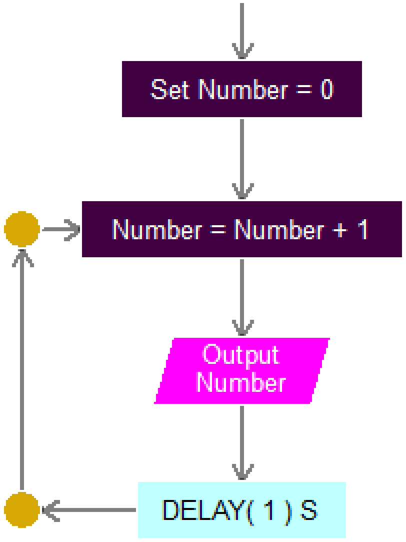
Computer memory

Assigning data to variable

portionOfSugar = 16 **location =** London

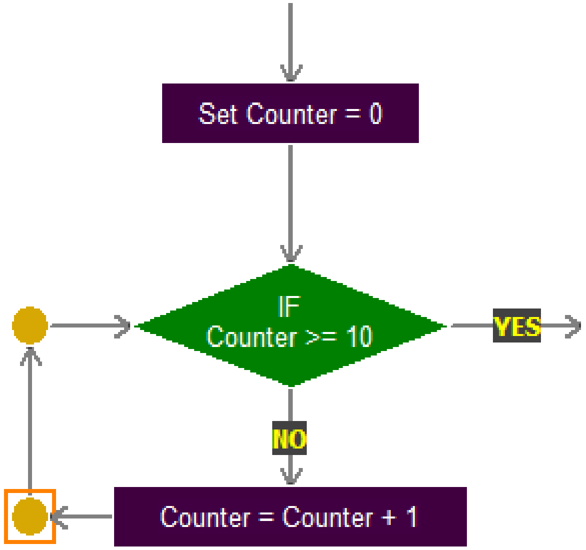
Loops and Conditional

Repeat loop



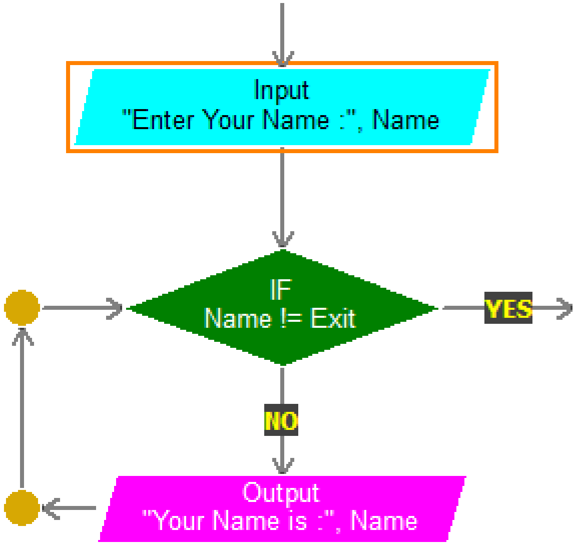
ULANGAN
TANPA HENTI

For loop



ULANG
UNTUK

While loop



ULANG SELAGI
SE - HINGGA

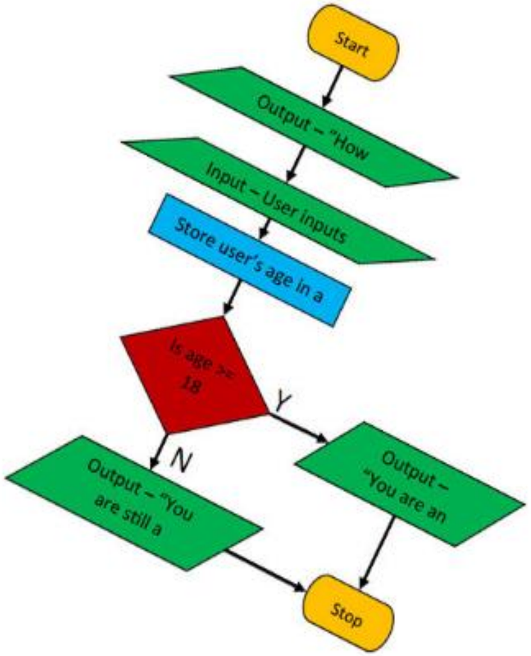
Computational Thinking

Computational Thinking (CT) is a problem solving process like computer does. CT is essential to the development of computer applications, but it can also be used to support problem solving across all disciplines, including the humanities, math, and science.

Process Elements of Computational Thinking

1. Decompositon:

Breaking down a big problem in smaller chunks.



2. Pragmatic Thinking:

Approaching the problem using programmatic thinking techniques such as iteration, symbolic representation, and logical operations

3. Pattern generalisaton & abstracton:

Putting a pattern in its simplest terms and creating a piece which can be used whenever needed. Abstracton means focusing on important information and ignoring irrelevant detail.

4. Algorithm design:

Stepwise solution to a problem where the above elements are used when deemed necessary..A visual method of depicting algorithms are **flowcharts**.

Module #2

Introduction to

FlowLogic 6

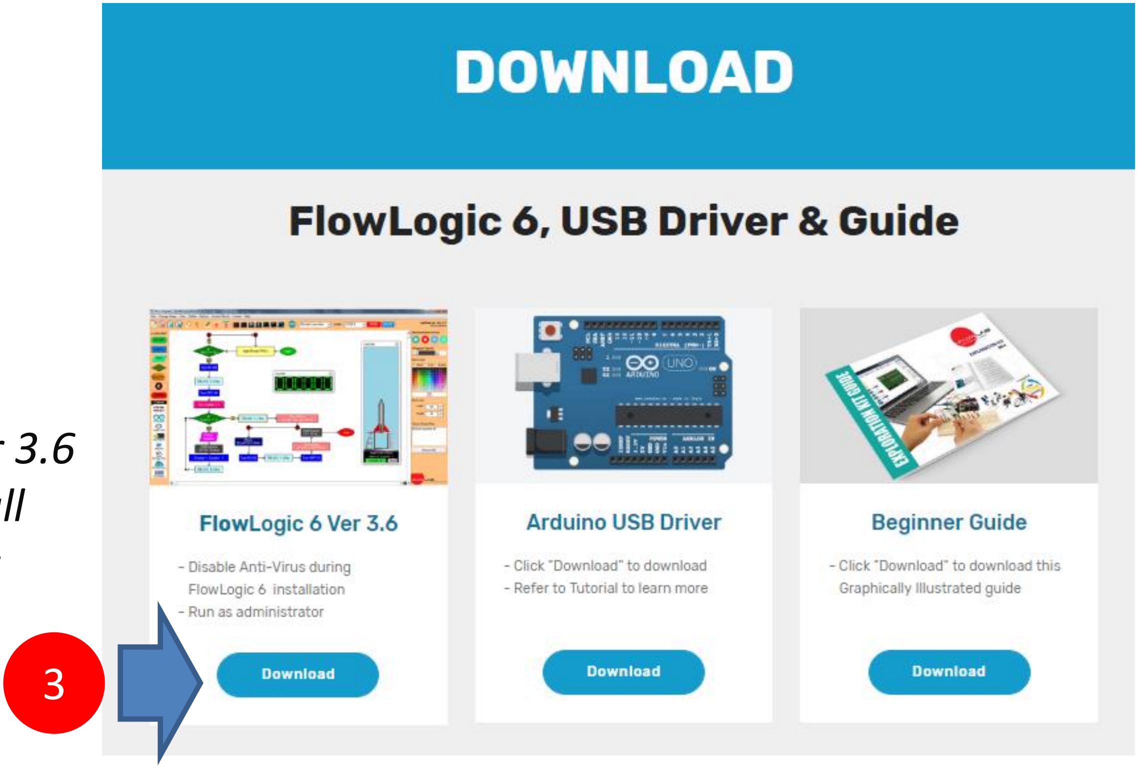
myFlowlab™

DIY - #1

FlowLogic 6

Download

1. Go to *www.myflowlab.com*
2. *Click Download*
3. *Click Download – FlowLogic 6 Ver 3.6*
4. *Click the Downloaded file to install FlowLogic 6 Version 3.6 into your computer*



DOWNLOAD

FlowLogic 6, USB Driver & Guide

FlowLogic 6 Ver 3.6

- Disable Anti-Virus during FlowLogic 6 installation
- Run as administrator

Arduino USB Driver

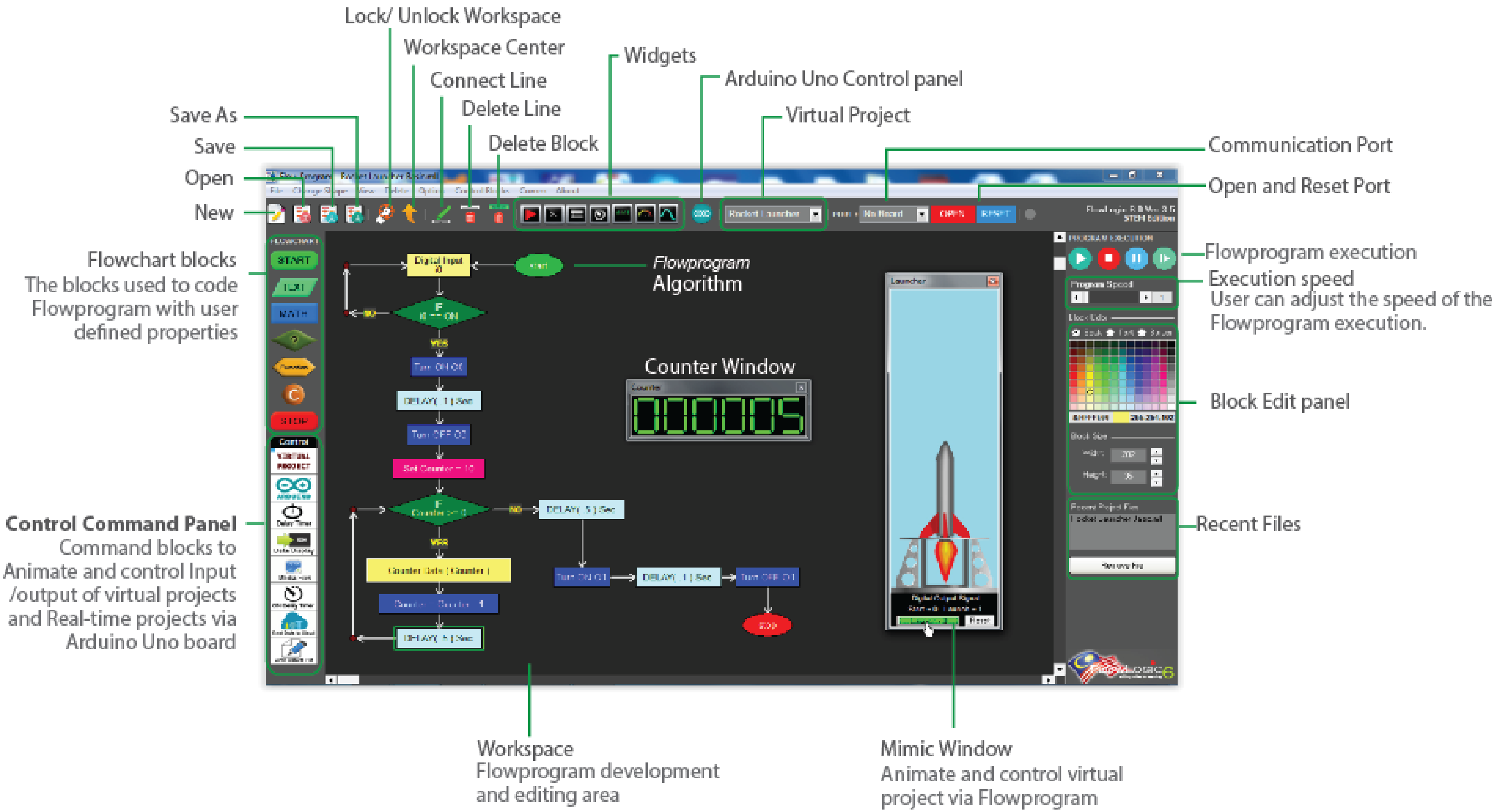
- Click "Download" to download
- Refer to Tutorial to learn more

Beginner Guide

- Click "Download" to download this Graphically Illustrated guide

FlowLogic 6 Version 3.6

myFlowlab™



Lock/ Unlock Workspace

Workspace Center

Connect Line

Delete Line

Delete Block

Widgets

Arduino Uno Control panel

Virtual Project

Communication Port

Open and Reset Port

Save As

Save

Open

New

Flowchart blocks
The blocks used to code Flowprogram with user defined properties

Flowprogram Algorithm

Counter Window

Flowprogram execution

Execution speed
User can adjust the speed of the Flowprogram execution.

Block Edit panel

Recent Files

Control Command Panel
Command blocks to Animate and control Input /output of virtual projects and Real-time projects via Arduino Uno board

Workspace
Flowprogram development and editing area

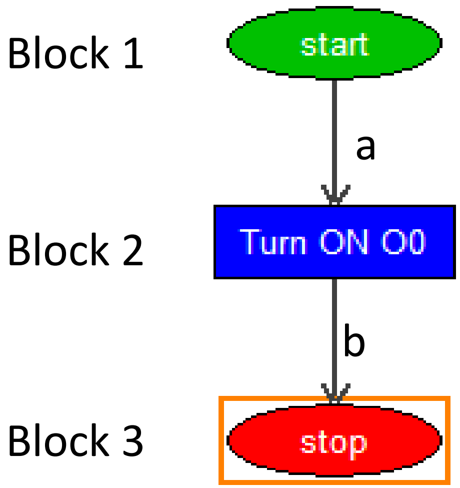
Mimic Window
Animate and control virtual project via Flowprogram

Introduce to student the FlowLogic 6 and Guide them on how to construct a FlowProgram

Editing Command Blocks

Activity - #1 – Practice Select the Blocks , define the property, Connect Line, Delete Line and Delete block to

To delete Line and Blocks



Block 1

start

a

Block 2

Turn ON O0

b

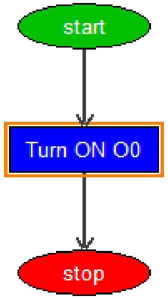
Block 3

stop

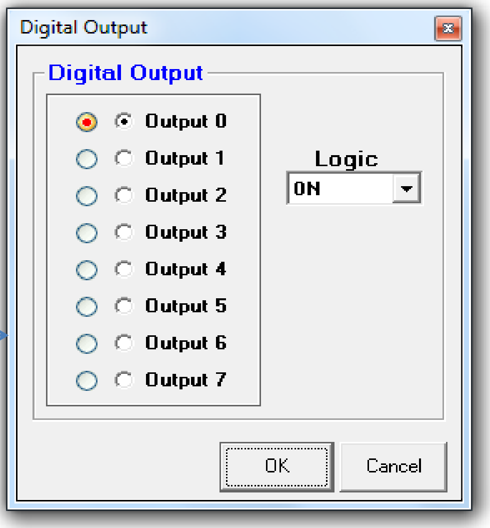
To delete line “a”, click on Block 1 and then Block 2, while mouse pointer on Block 2, right click and select “Delete Line” option from the pop-up menu.

To delete Blocks, delete all connecting line, right Click on the block and select “Delete block” option from the Pop-menu.

Double click the block to Edit



Property Window →



Digital Output

Digital Output

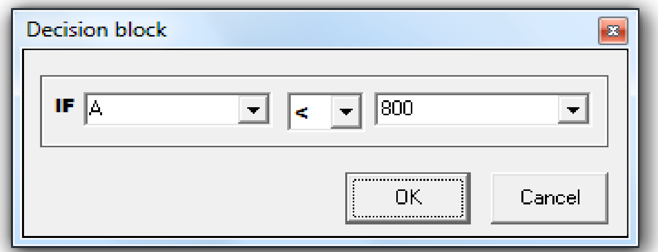
● Output 0
○ Output 1
○ Output 2
○ Output 3
○ Output 4
○ Output 5
○ Output 6
○ Output 7

Logic: ON

To edit blocks, double click on the block and make the necessary changes on the pop-up property Windows and click “Ok” when done.

Working with Decision Blocks

Activity - #2 - Practice Select the Blocks , define the property, Connect Line to try out



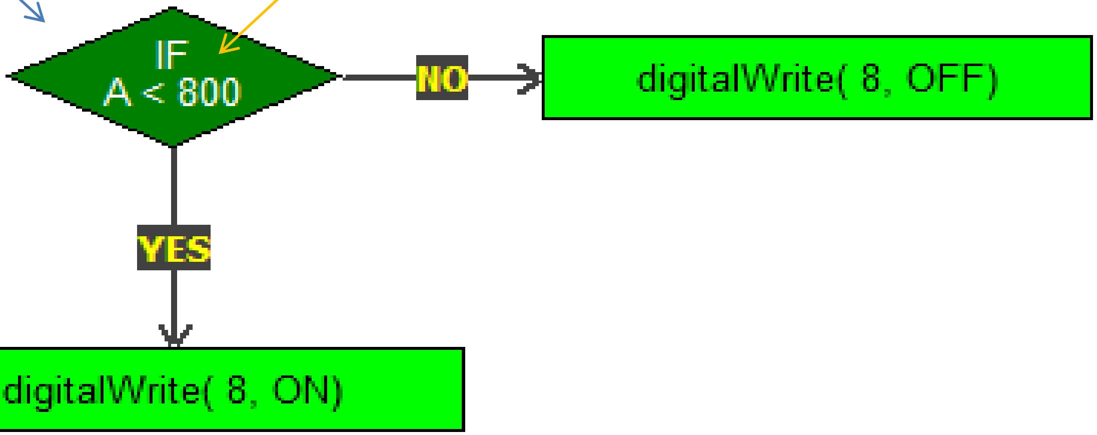
Decision block

IF A < 800

OK Cancel

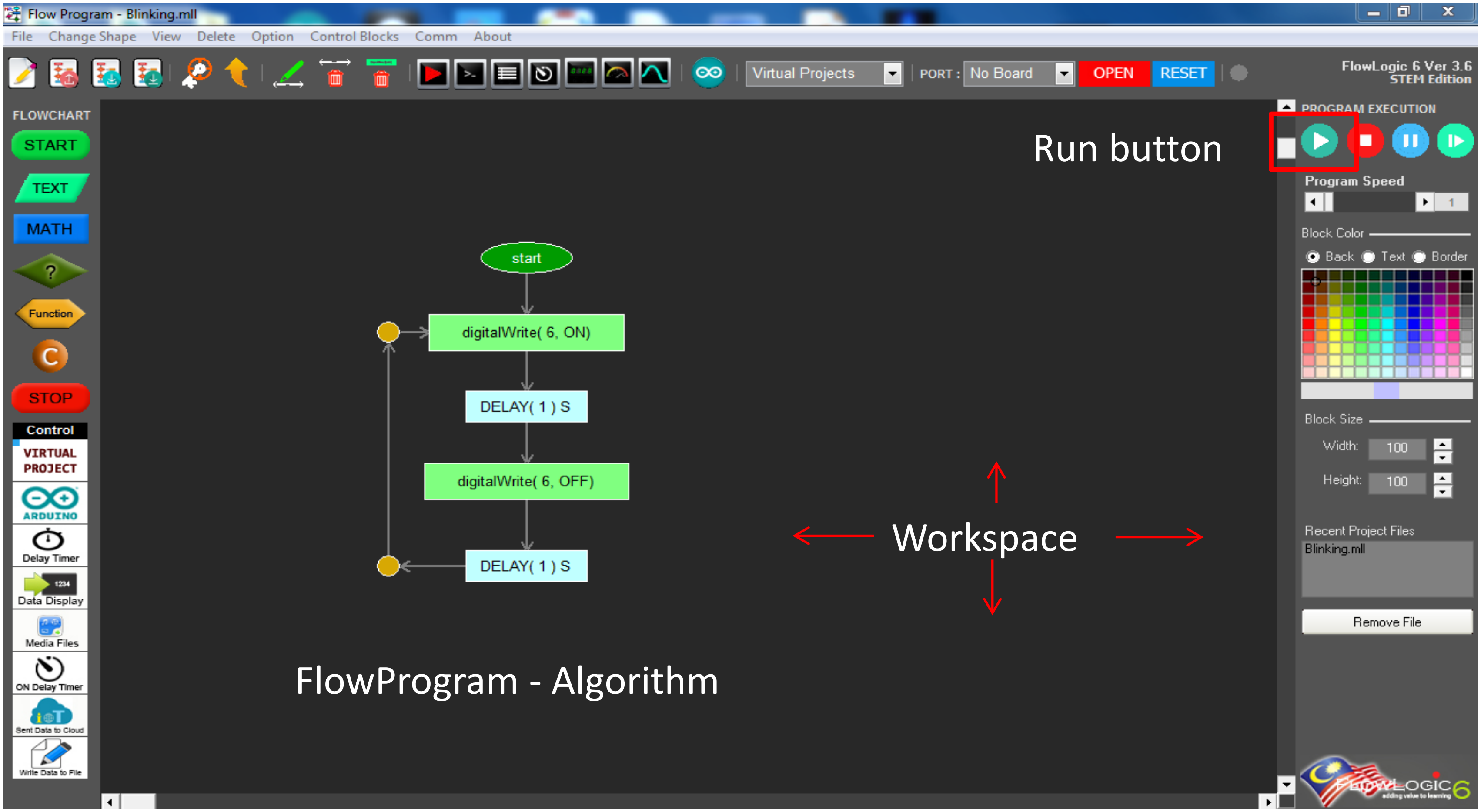
Decision block
Property Window

Condition statement as specified
in the property window



Decision block requires two (2) connecting point, the first connection to a block
will be “YES” and the next connection will be “NO”.
Connect the lines at your discretion based on the condition statement on the decision block.

Running a FlowPogram



Run button

Workspace

FlowProgram - Algorithm

Explain to Students the steps required to run a FlowProgram that is loaded into the Workspace.

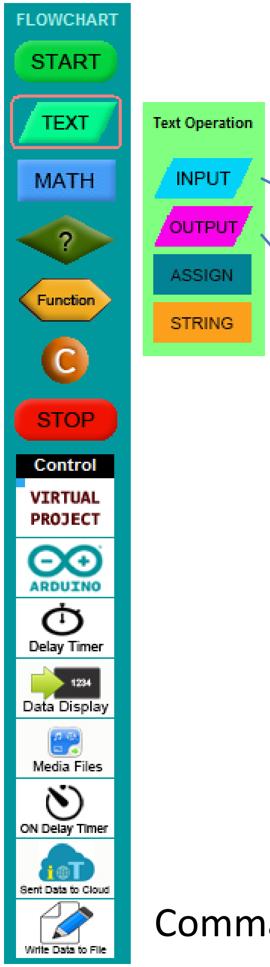
Module #3

Developing

Console Applications

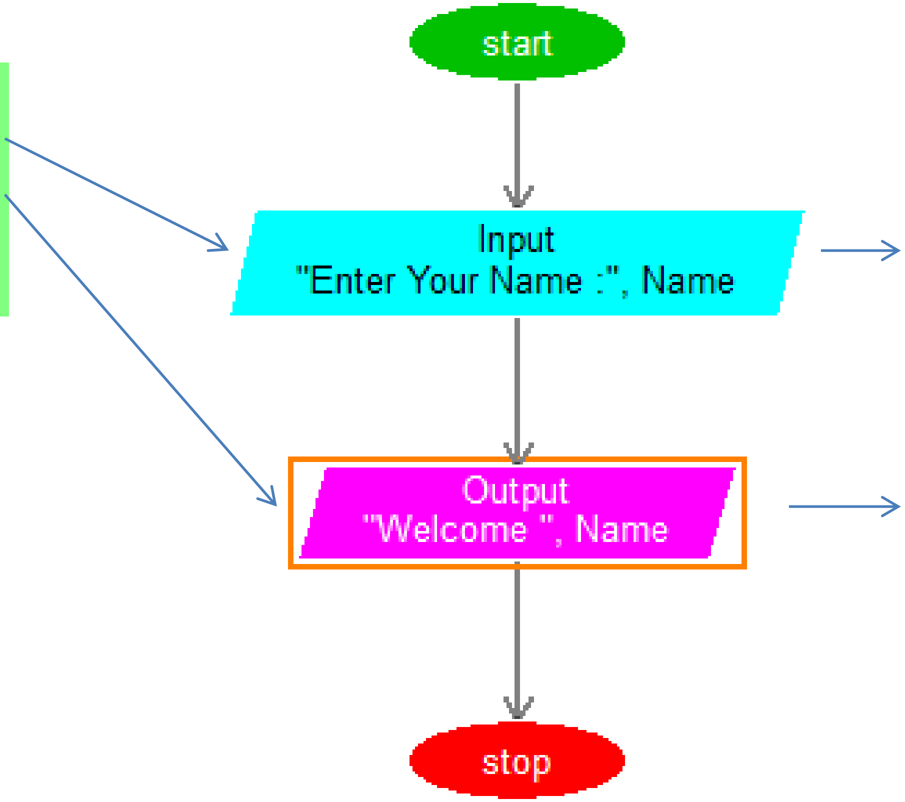
FlowProgram / Algorithm – Activity #3

Console applications



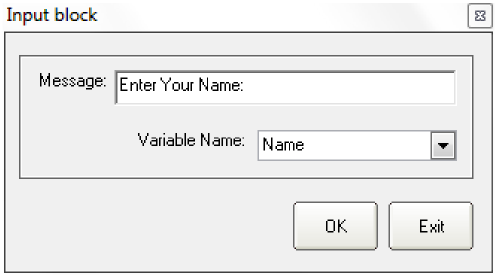
Command Blocks

FlowProgram/Algorithm



Run the FlowProgram to view the output..

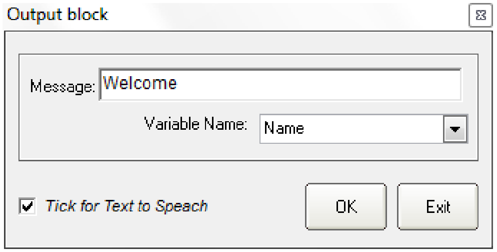
Block's Property Windows



Input block

Message: Enter Your Name:

Variable Name: Name



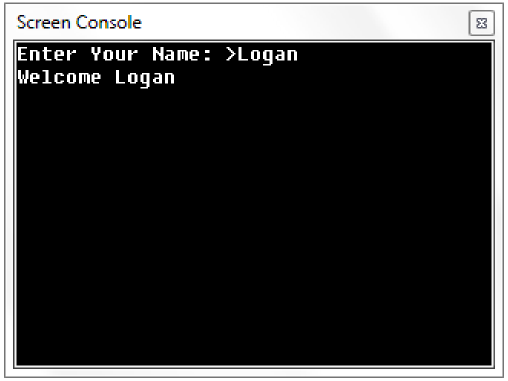
Output block

Message: Welcome

Variable Name: Name

☑ *Tick for Text to Speach*

Screen Console widget



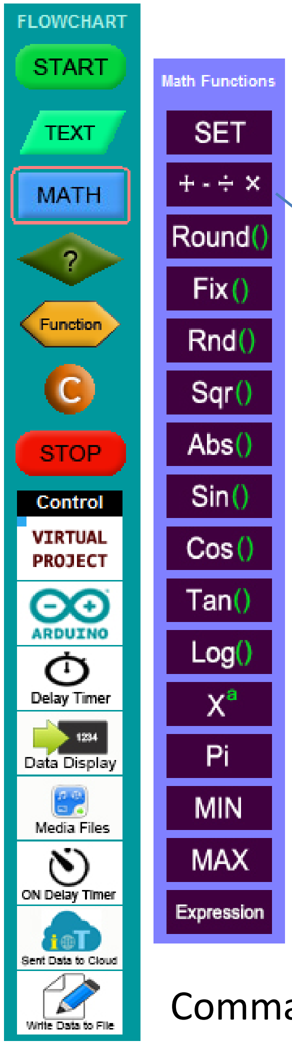
Screen Console

```
Enter Your Name: >Logan
Welcome Logan
```

myFlowlab™

FlowProgram / Algorithm – Activity #4

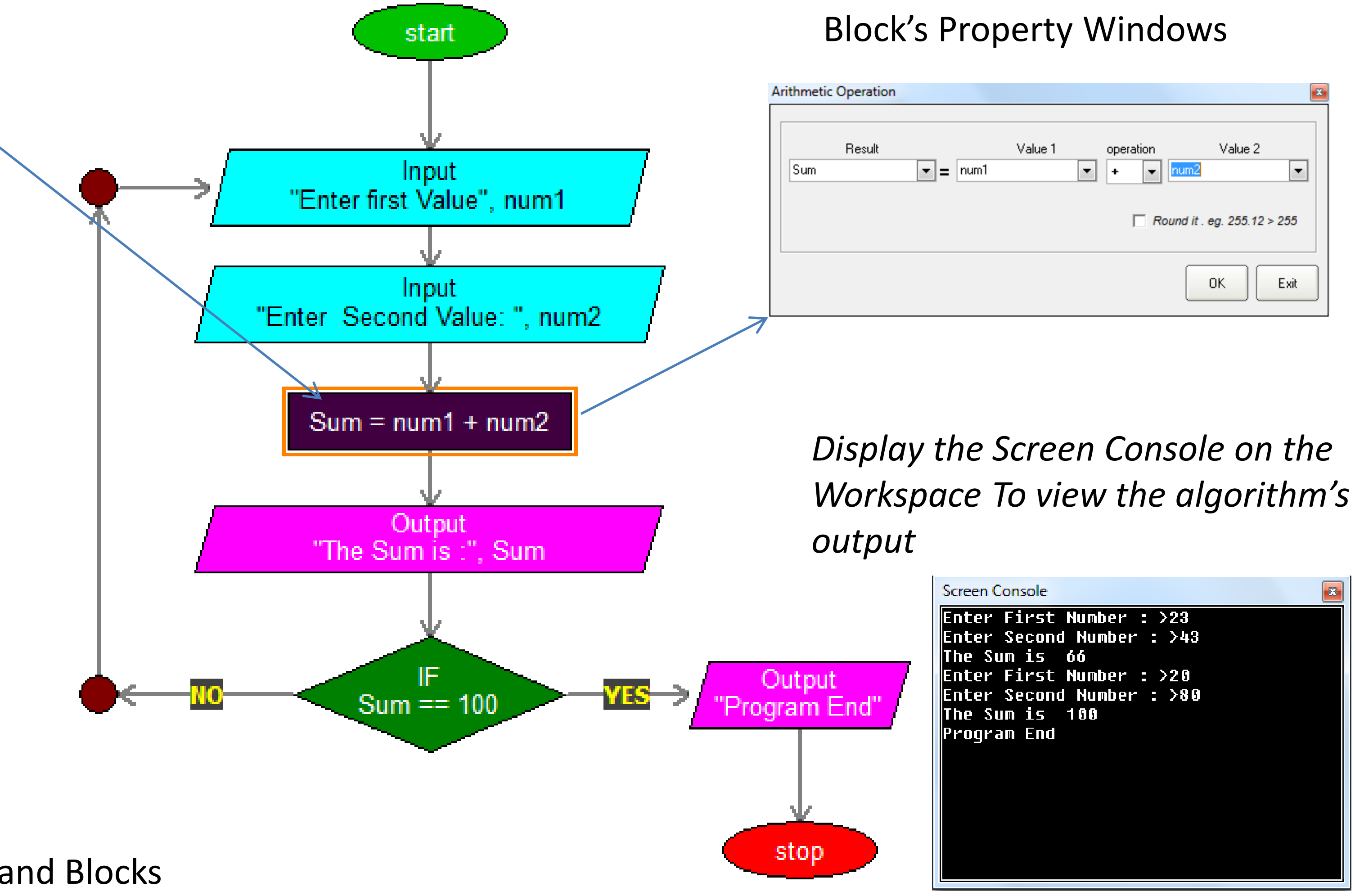
Console applications



Command Blocks

FlowProgram/Algorithm

Block's Property Windows



Arithmetic Operation

Result		Value 1	operation	Value 2
Sum	=	num1	+	num2

□ Round it . eg. 255.12 > 255

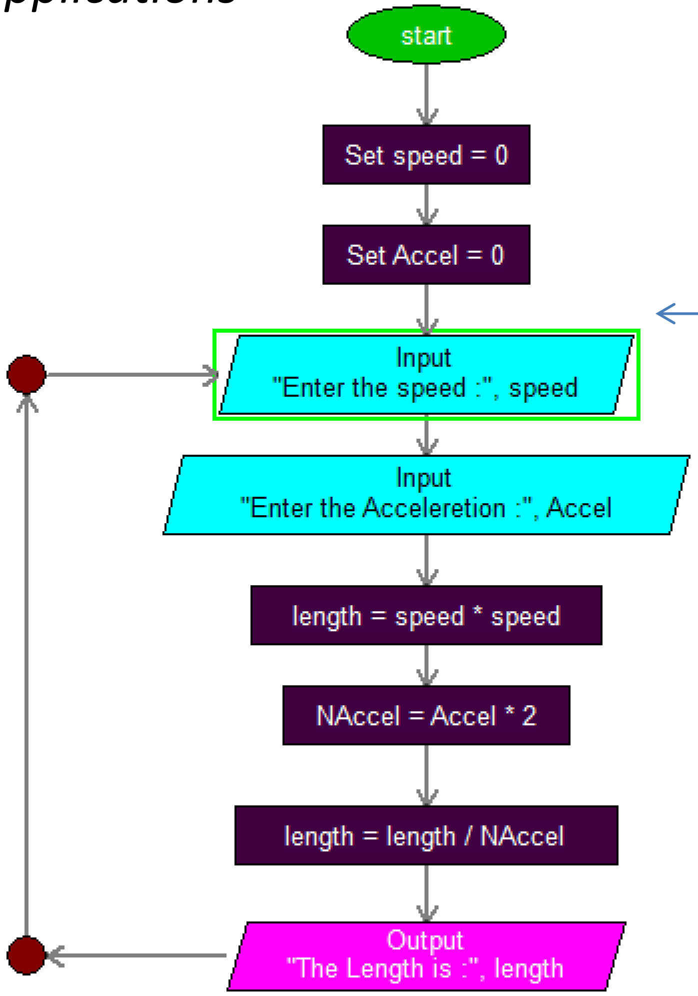
Display the Screen Console on the Workspace To view the algorithm's output

Screen Console

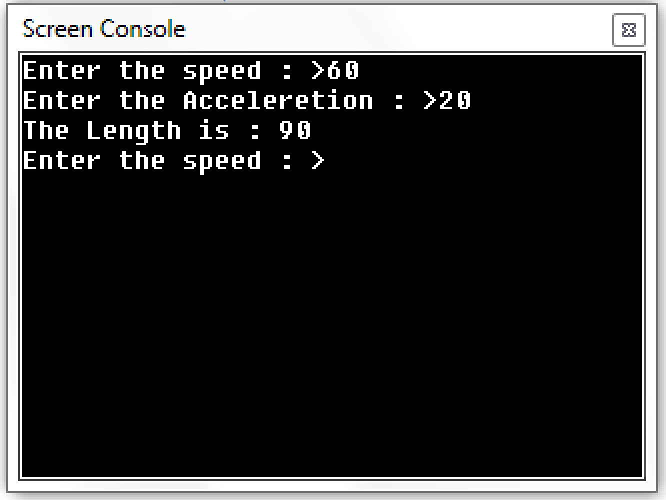
```
Enter First Number : >23
Enter Second Number : >43
The Sum is  66
Enter First Number : >20
Enter Second Number : >80
The Sum is  100
Program End
```

FlowProgram / Algorithm – DIY #2

Console applications



Construct the FlowProgram to produce the output shown below



```
Enter the speed : >60
Enter the Acceleretion : >20
The Length is : 90
Enter the speed : >
```

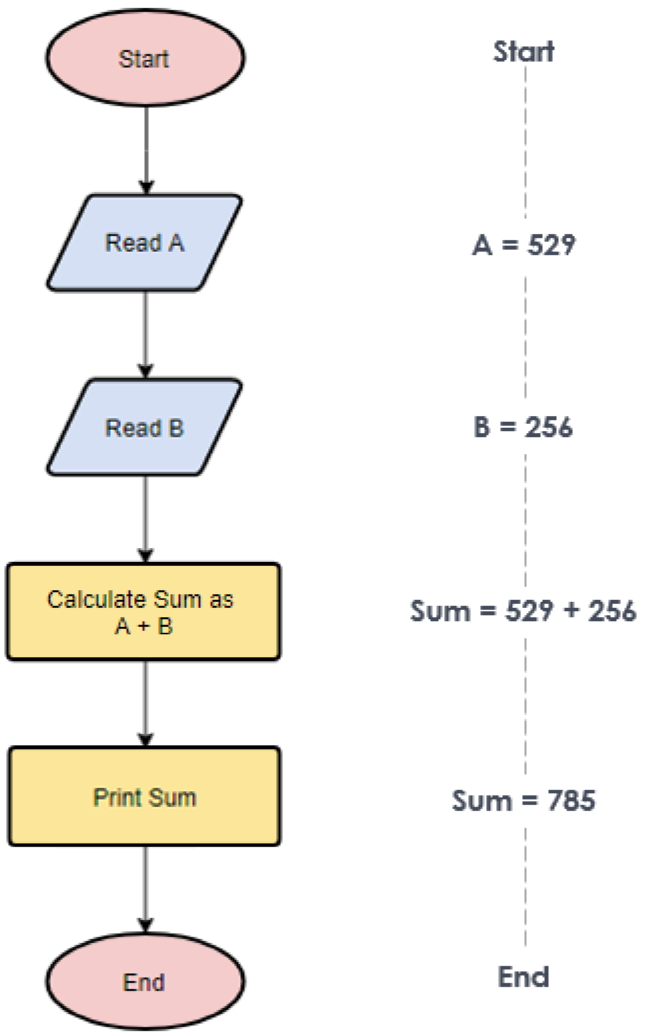
Computational Thinking –DIY #3

myFlowlab™

Console applications

Construct the FlowProgram to solve the given problem based on the flowchart reference →

Find the sum of 529 and 256



Start

Read A

Read B

Calculate Sum as A + B

Print Sum

End

Start

A = 529

B = 256

Sum = 529 + 256

Sum = 785

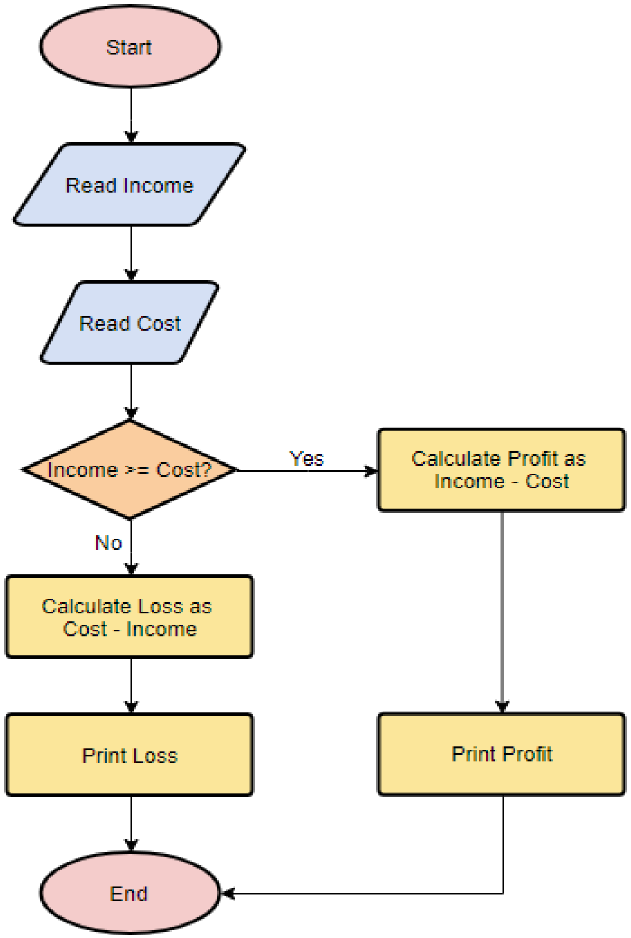
End

M myFlowlab™

Computational Thinking –DIY #4

Console applications

Construct the FlowProgram to solve the given problem based on the flowchart reference



Start

Read Income

Read Cost

Income >= Cost?

Yes

Calculate Profit as Income - Cost

No

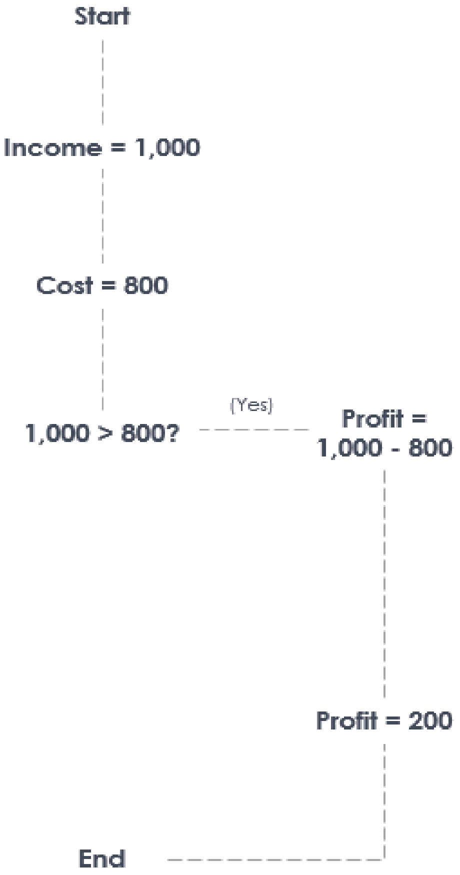
Calculate Loss as Cost - Income

Print Loss

Print Profit

End

Find the profit/loss when income = 1,000, cost = 800



Start

Income = 1,000

Cost = 800

1,000 > 800?

(Yes)

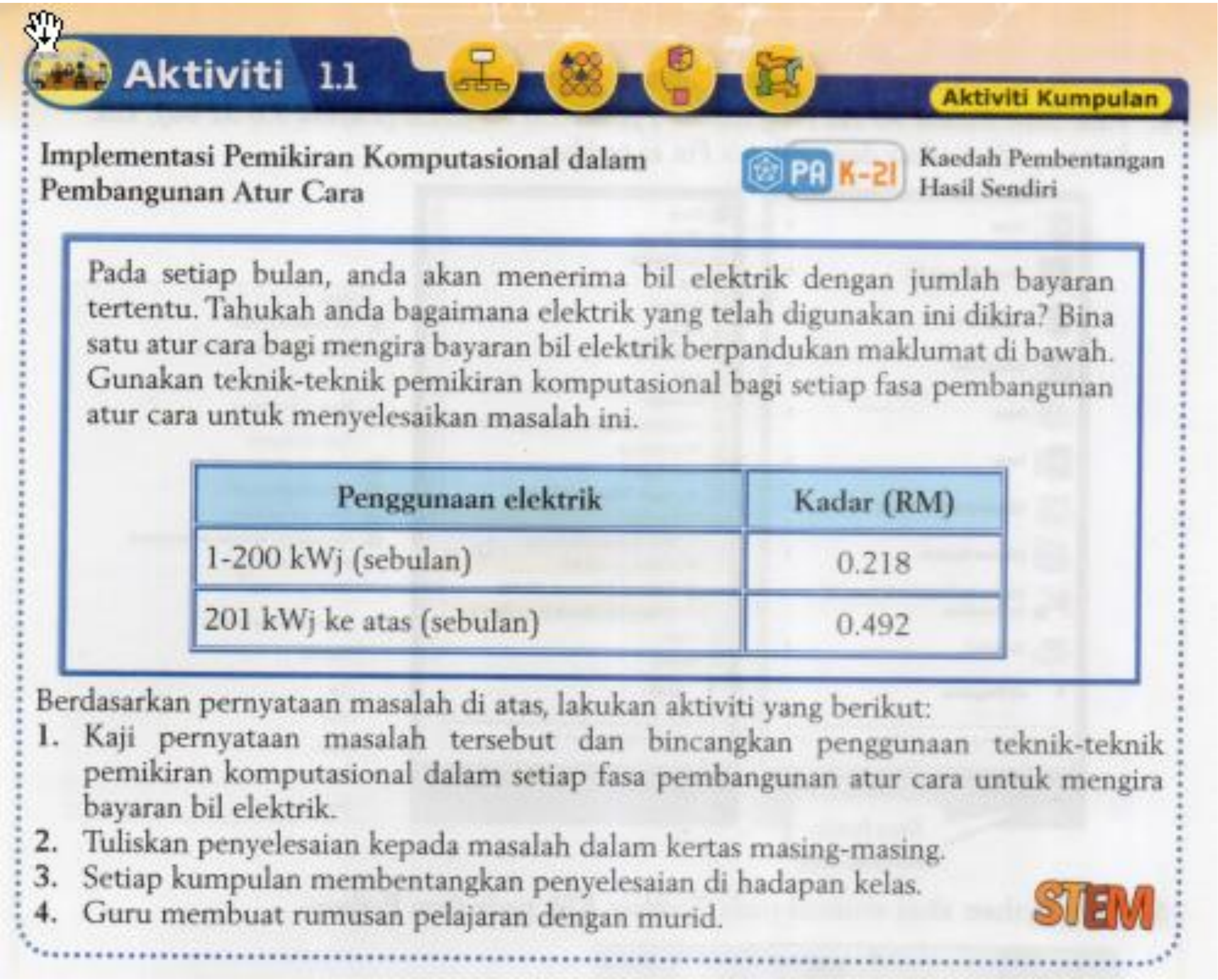
Profit = 1,000 - 800

Profit = 200

End

Computational Thinking –DIY #5

Console applications



Aktiviti 1.1

Aktiviti Kumpulan

Implementasi Pemikiran Komputasional dalam Pembangunan Atur Cara

PA K-21 Kaedah Pembentangan Hasil Sendiri

Pada setiap bulan, anda akan menerima bil elektrik dengan jumlah bayaran tertentu. Tahukah anda bagaimana elektrik yang telah digunakan ini dikira? Bina satu atur cara bagi mengira bayaran bil elektrik berpandukan maklumat di bawah. Gunakan teknik-teknik pemikiran komputasional bagi setiap fasa pembangunan atur cara untuk menyelesaikan masalah ini.

Penggunaan elektrik	Kadar (RM)
1-200 kWj (sebulan)	0.218
201 kWj ke atas (sebulan)	0.492

Berdasarkan pernyataan masalah di atas, lakukan aktiviti yang berikut:

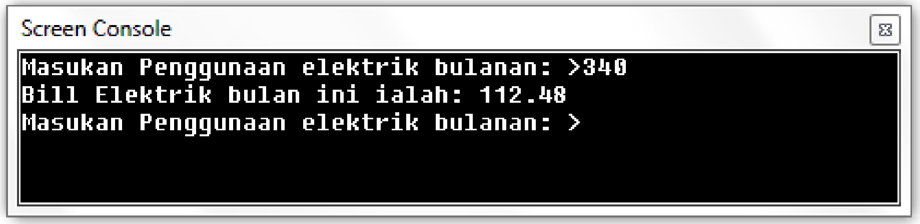
1. Kaji pernyataan masalah tersebut dan bincangkan penggunaan teknik-teknik pemikiran komputasional dalam setiap fasa pembangunan atur cara untuk mengira bayaran bil elektrik.
2. Tuliskan penyelesaian kepada masalah dalam kertas masing-masing.
3. Setiap kumpulan membentangkan penyelesaian di hadapan kelas.
4. Guru membuat rumusan pelajaran dengan murid.

STEM

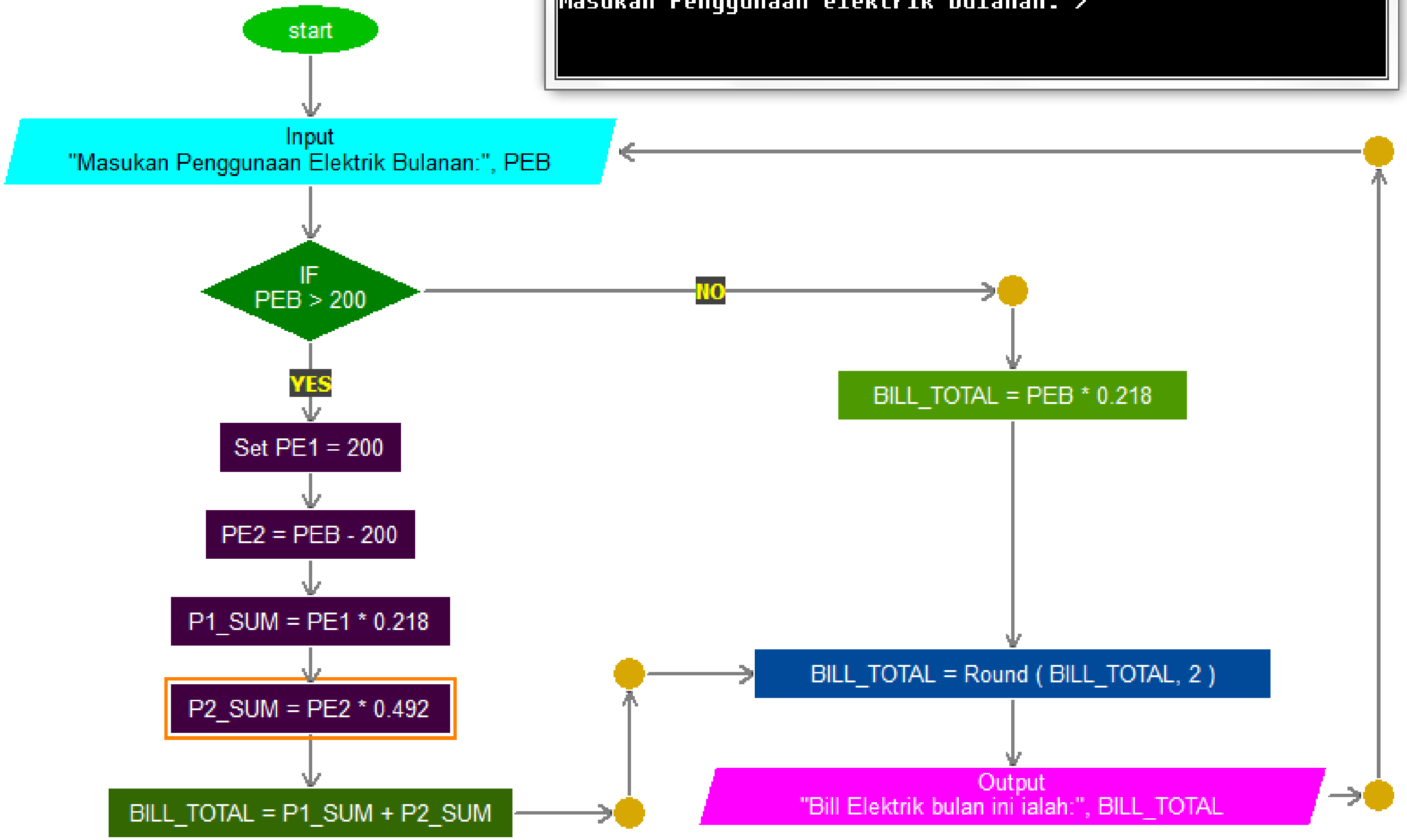
DIY #5 - Solution

Console applications

myFlowlab™



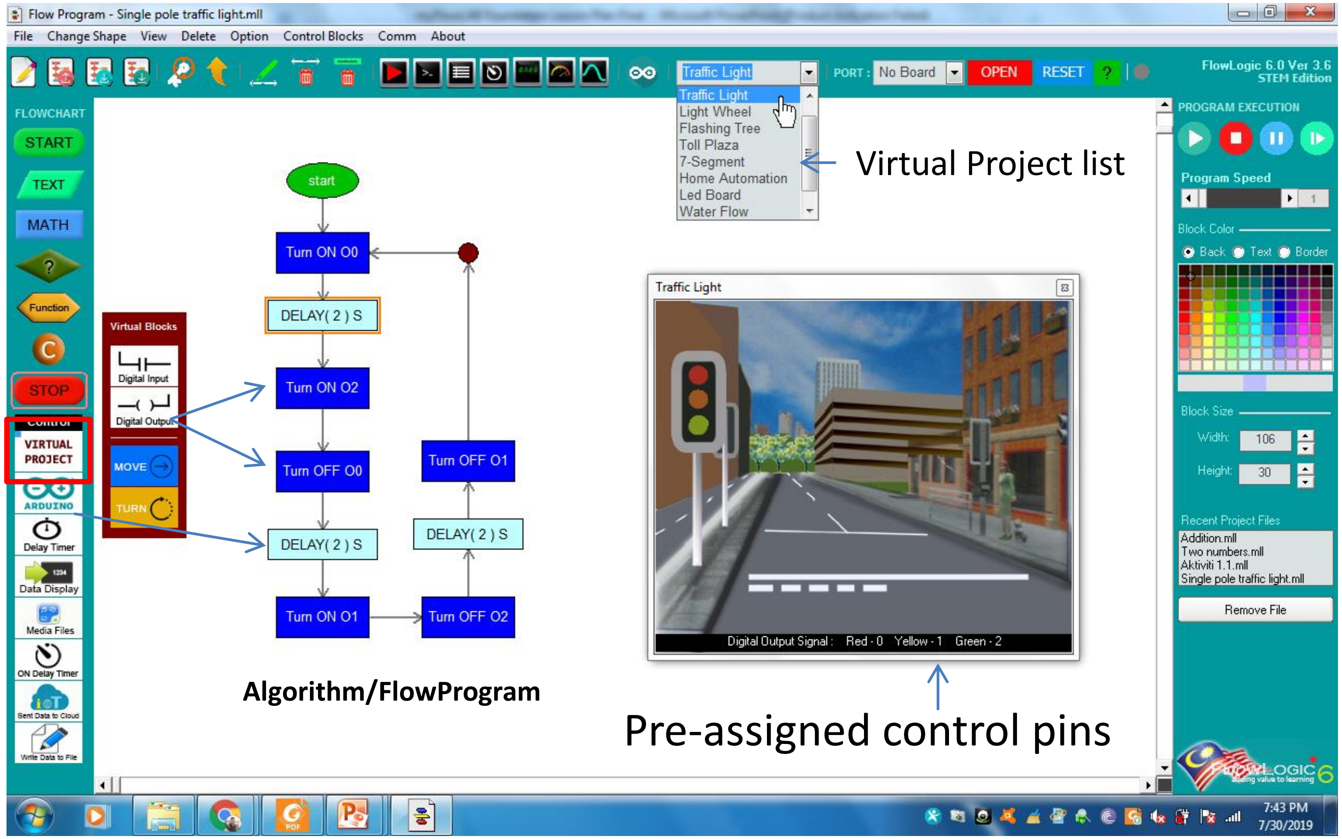
```
Masukan Penggunaan elektrik bulanan: >340
Bill Elektrik bulan ini ialah: 112.48
Masukan Penggunaan elektrik bulanan: >
```


start
Input
"Masukan Penggunaan Elektrik Bulanan:", PEB
IF
PEB > 200
NO
YES
Set PE1 = 200
PE2 = PEB - 200
P1_SUM = PE1 * 0.218
P2_SUM = PE2 * 0.492
BILL_TOTAL = P1_SUM + P2_SUM
BILL_TOTAL = PEB * 0.218
BILL_TOTAL = Round (BILL_TOTAL, 2)
Output
"Bill Elektrik bulan ini ialah:", BILL_TOTAL

Module #4
Building Algorithm using Virtual Projects

Virtual Projects

Is a On-screen mimics with Pre-assigned control pins and animations that can be programmed by using the virtual command blocks



Command Block

Virtual Project list

Algorithm/FlowProgram

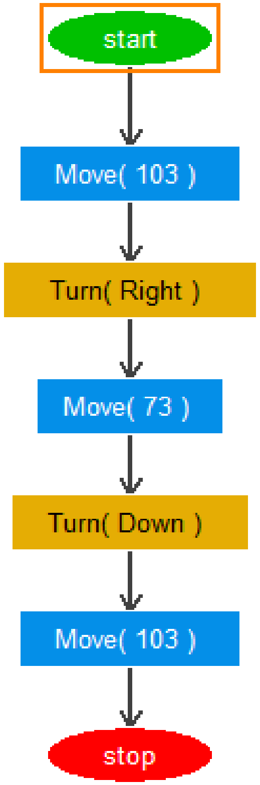
Pre-assigned control pins

When running Virtual project programs, select the appropriate project from the list and place it on the Workspace.

myFlowlab™

Activity #5 : Robot Maze (Virtual Project)

In this project student will construct FlowProgram using FlowLogic 6 to navigate a Virtual Robot thru a maze.


start

Move(103)

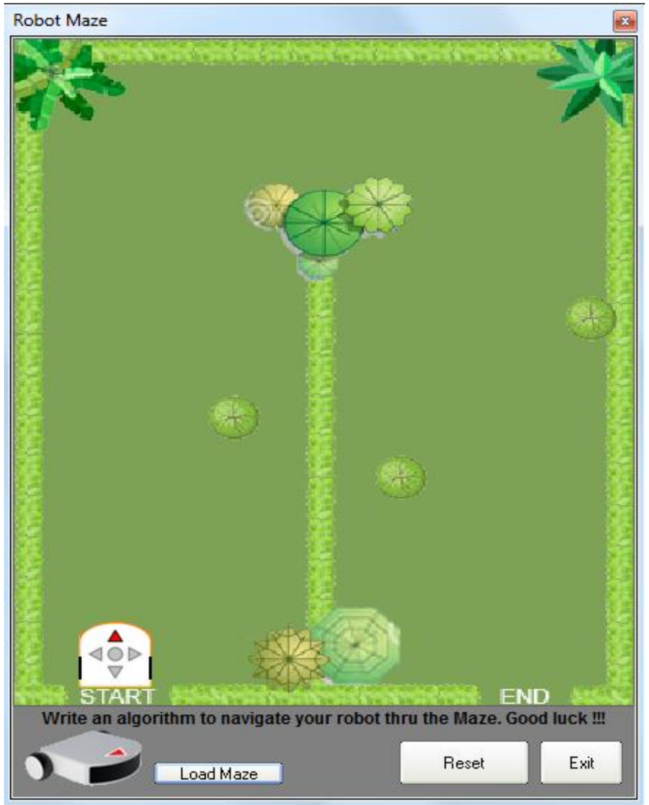
Turn(Right)

Move(73)

Turn(Down)

Move(103)

stop



Write an algorithm to navigate your robot thru the Maze. Good luck !!!

Step 1: Select Robot Maze from the Virtual Project List

Step 2: Click Load Maze to Load Rmaze1 image from the folder

Step 3: Construct the FlowProgram as shown

Step 4: Click Run icon to execute the FlowProgram

Step 5: Click Reset to place Robot to its origin location

Execute again to test again

Activity : Get the students to load other Maze images and construct the FlowProgram to practice their skill in Flowchart programming and Computational Thinking.

DIY #6 : Robot Maze (Virtual Project)

Step 1: Select Robot Maze from the Virtual Project List

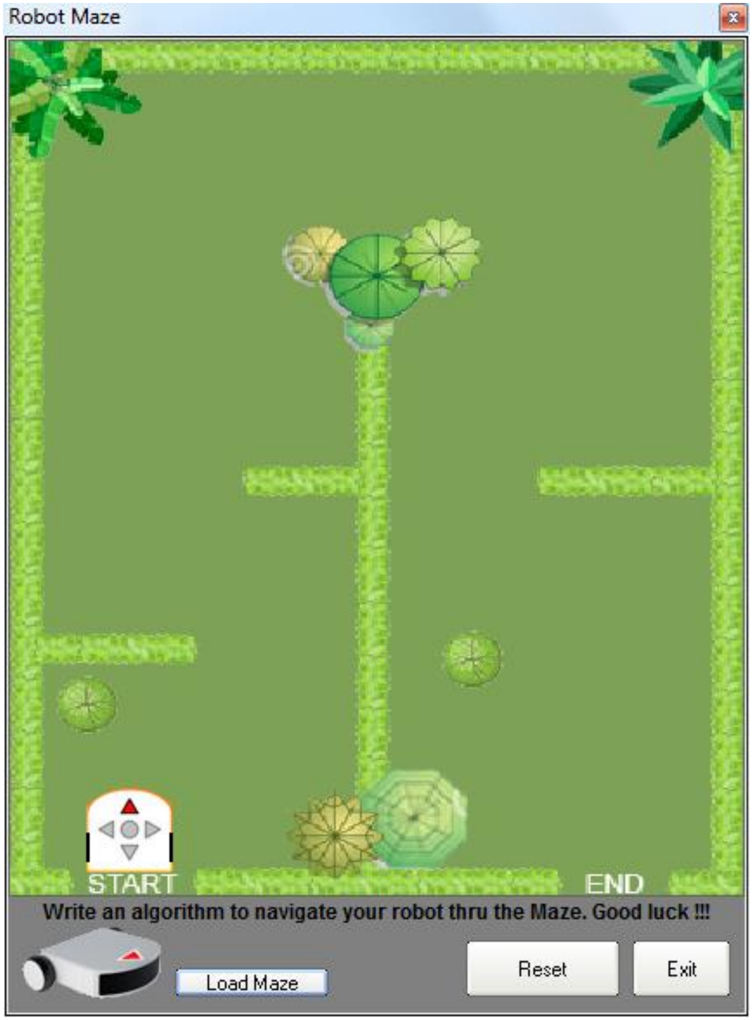
Step 2: Click Load Maze to Load Rmaze2 image from the folder

Step 3: Construct the FlowProgram to navigate the Robot thru the Maze

Step 4: Click Run icon to execute the FlowProgram

Step 5: Click Reset to place Robot to its origin location

Execute again to test again

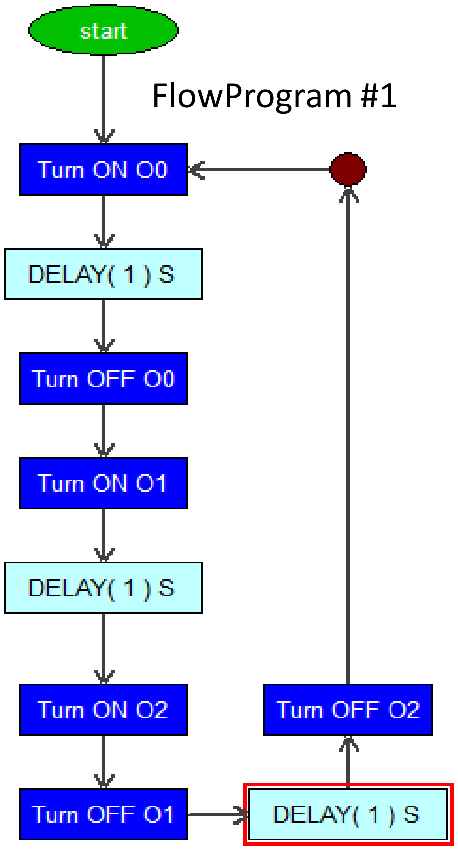


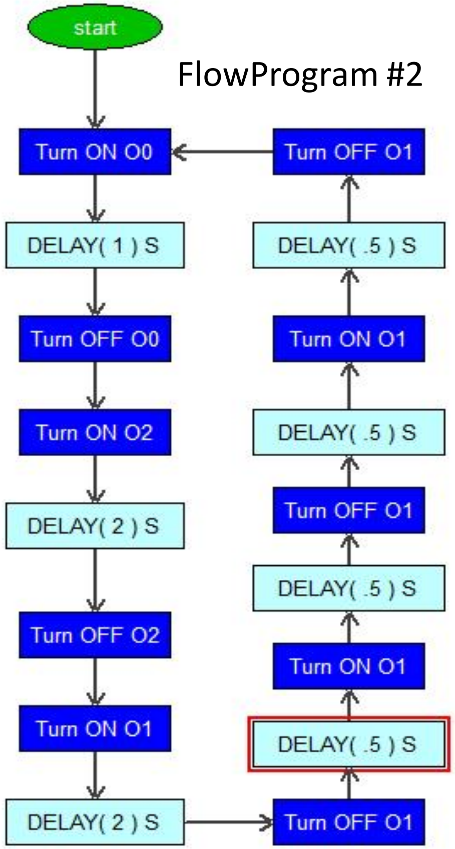
Write an algorithm to navigate your robot thru the Maze. Good luck !!!

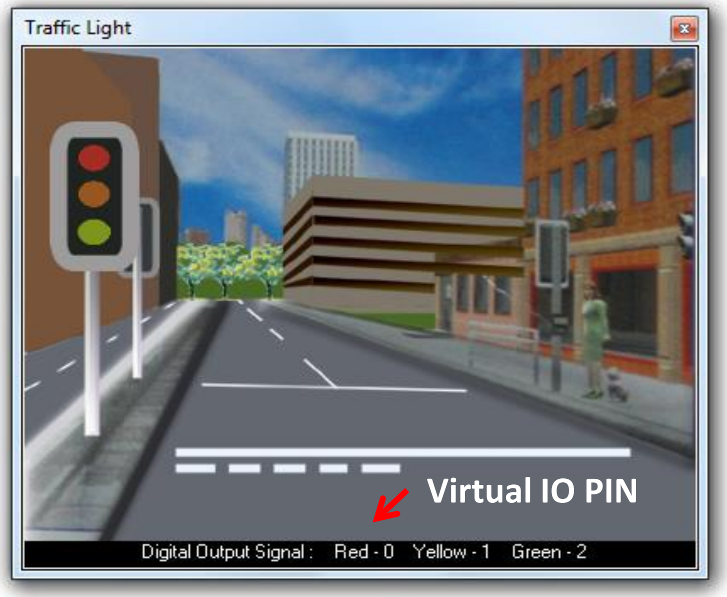
myFlowlab™

Activity #6 : Traffic Light (Virtual Project)

Develop FlowProgram / Algorithm to control a Virtual Traffic Light system.







Load Traffic Light Virtual Model from the Virtual project list

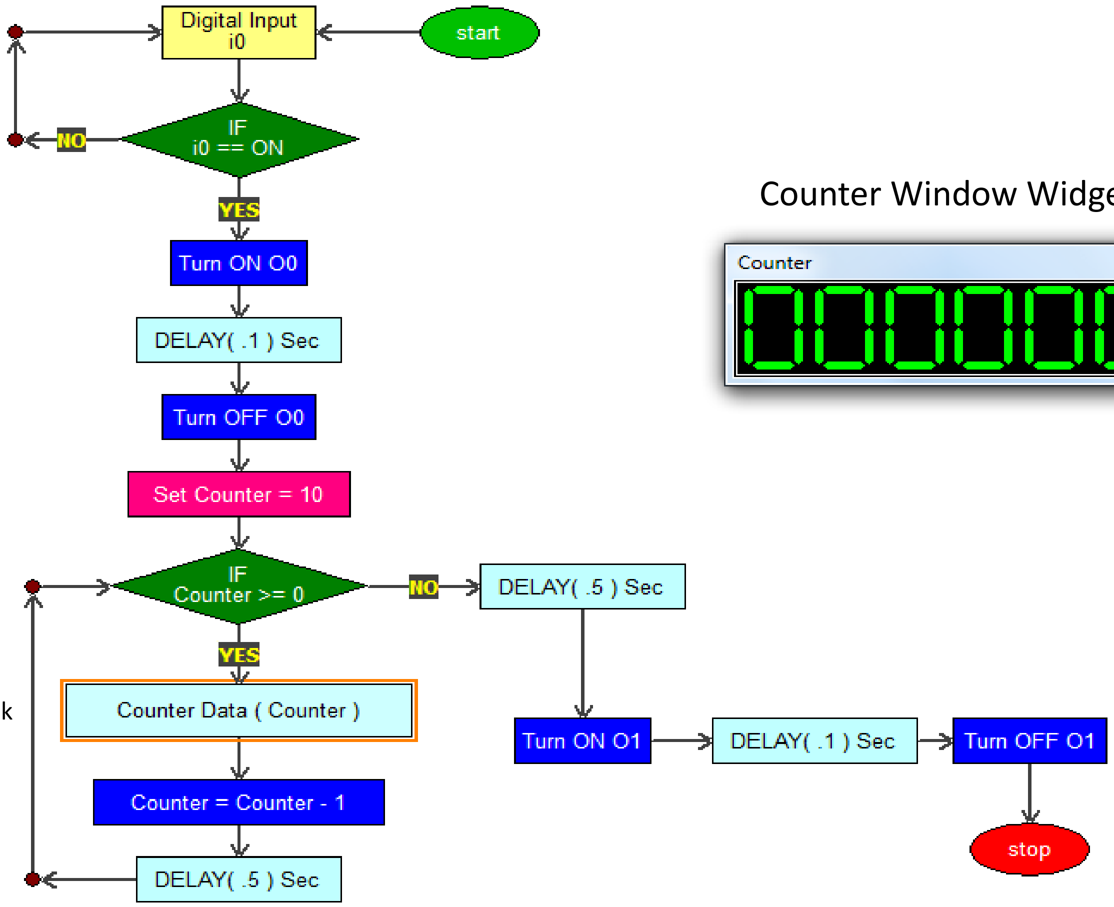
Get Student to construct the FlowProgram #1 Using virtual IO pin, Test and Present.

Variant : Get the students to construct a new FlowProgram (FlowProgram #2) to Blink the Yellow Light to enhance their skill in Flowchart programming and Computational Thinking.

Activity #7 : Rocket Launcher (Virtual Project)

Develop FlowProgram / Algorithm to Launch a Virtual Rocket. (BASIC operation)

Data Display Block



start

Digital Input i0

IF i0 == ON

NO

YES

Turn ON O0

DELAY(.1) Sec

Turn OFF O0

Set Counter = 10

IF Counter >= 0

NO

DELAY(.5) Sec

YES

Counter Data (Counter)

Counter = Counter - 1

DELAY(.5) Sec

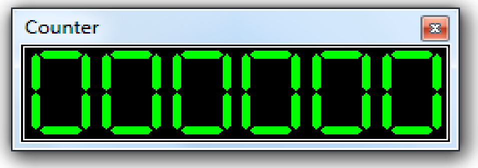
Turn ON O1

DELAY(.1) Sec

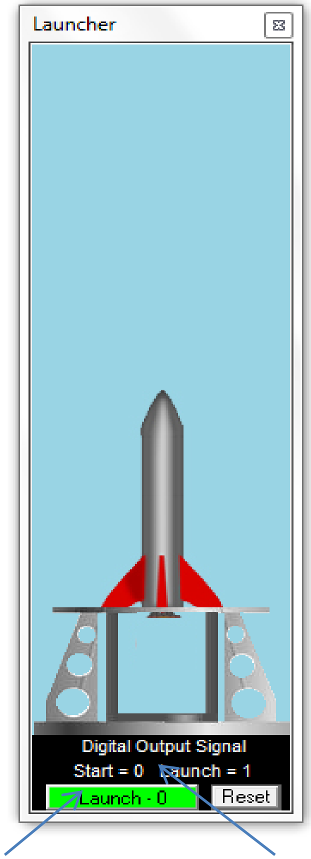
Turn OFF O1

stop

Counter Window Widgets



Select from Virtual project list
– Rocket Launcher



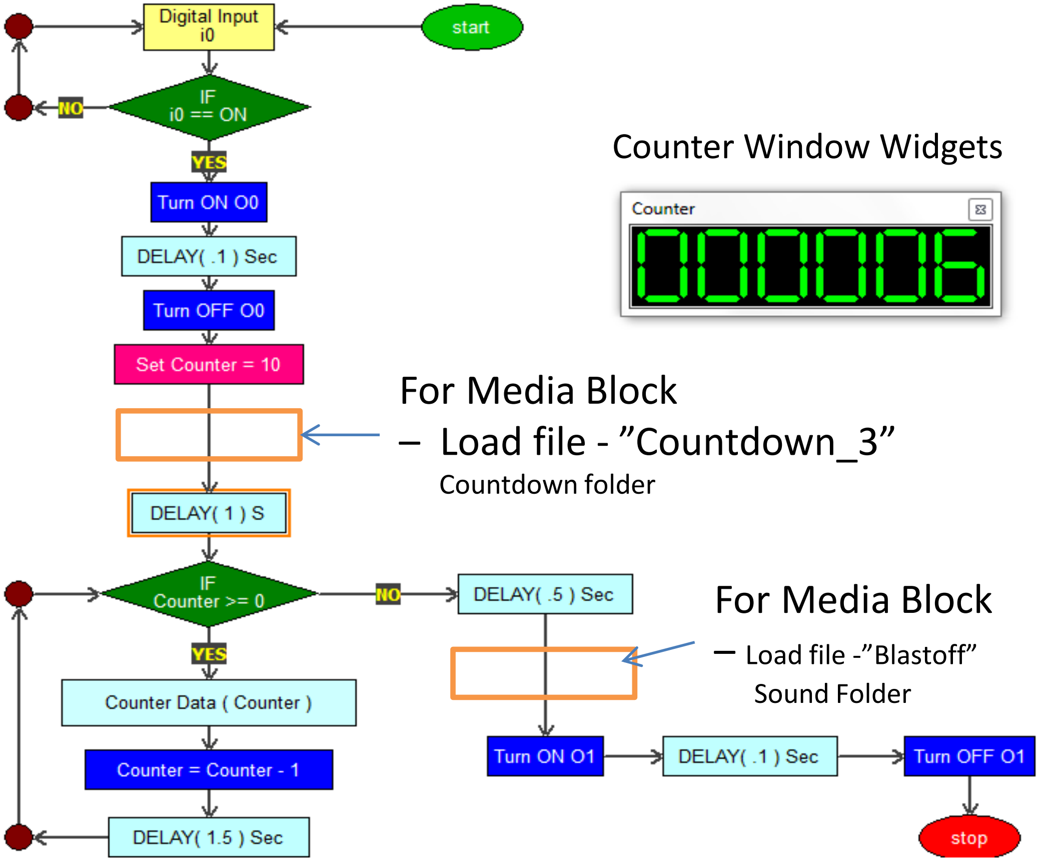
Input Virtual I/Os

Output Virtual I/Os

Activity #8 : Rocket Launcher (Virtual Project)

Develop FlowProgram / Algorithm to Launch a Virtual Rocket with background audio using Media blocks

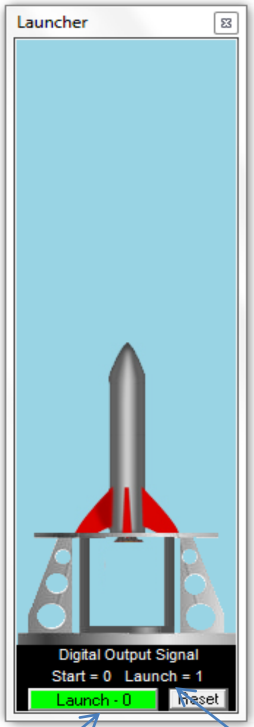
Select from Virtual project list
– Rocket Launcher



Counter Window Widgets

For Media Block
– Load file - ”Countdown_3”
Countdown folder

For Media Block
– Load file -”Blastoff”
Sound Folder

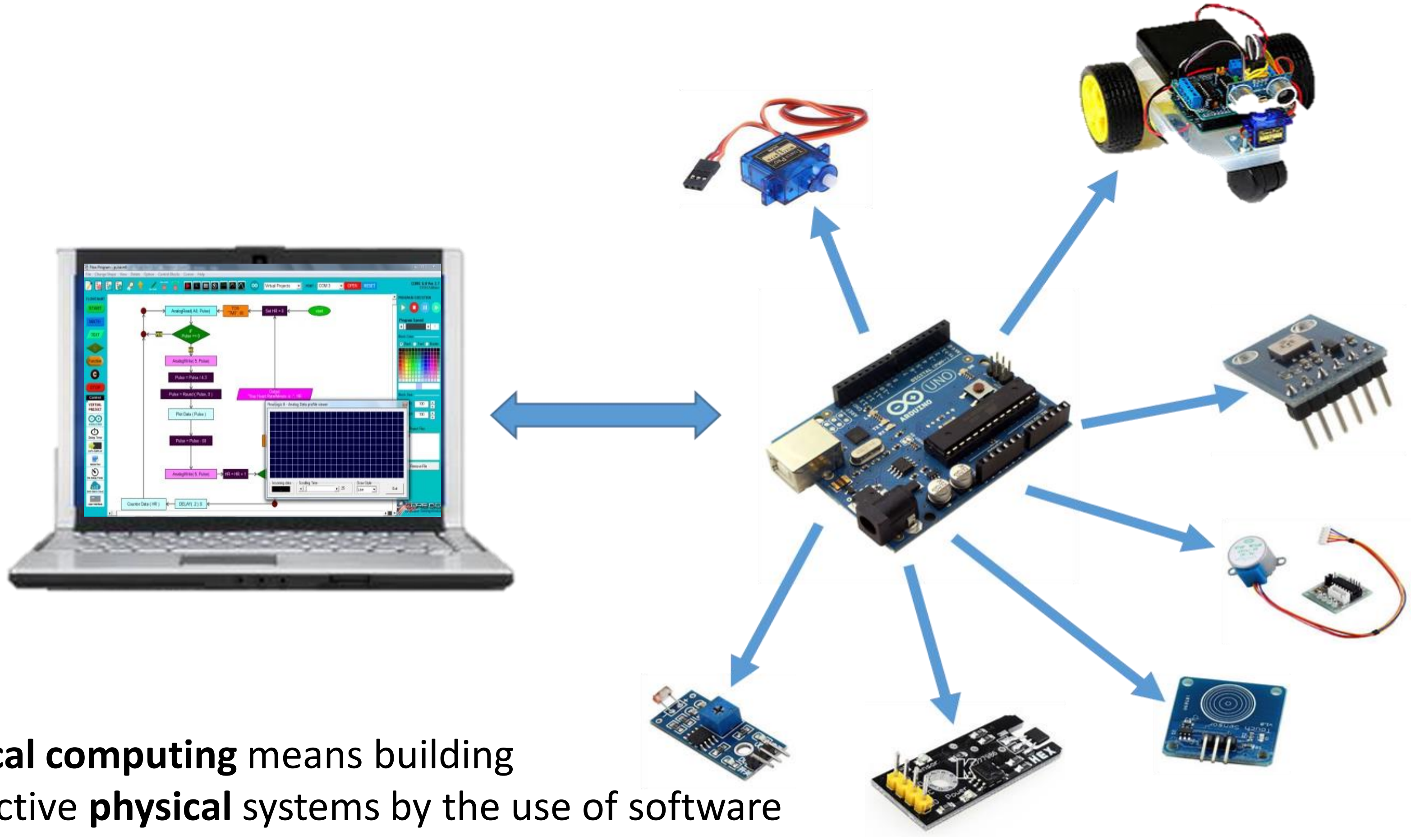


Input Virtual I/Os

Output Virtual I/Os

Module #5
Controlling, Monitoring, analyzing real-world applications using Arduino UNO board

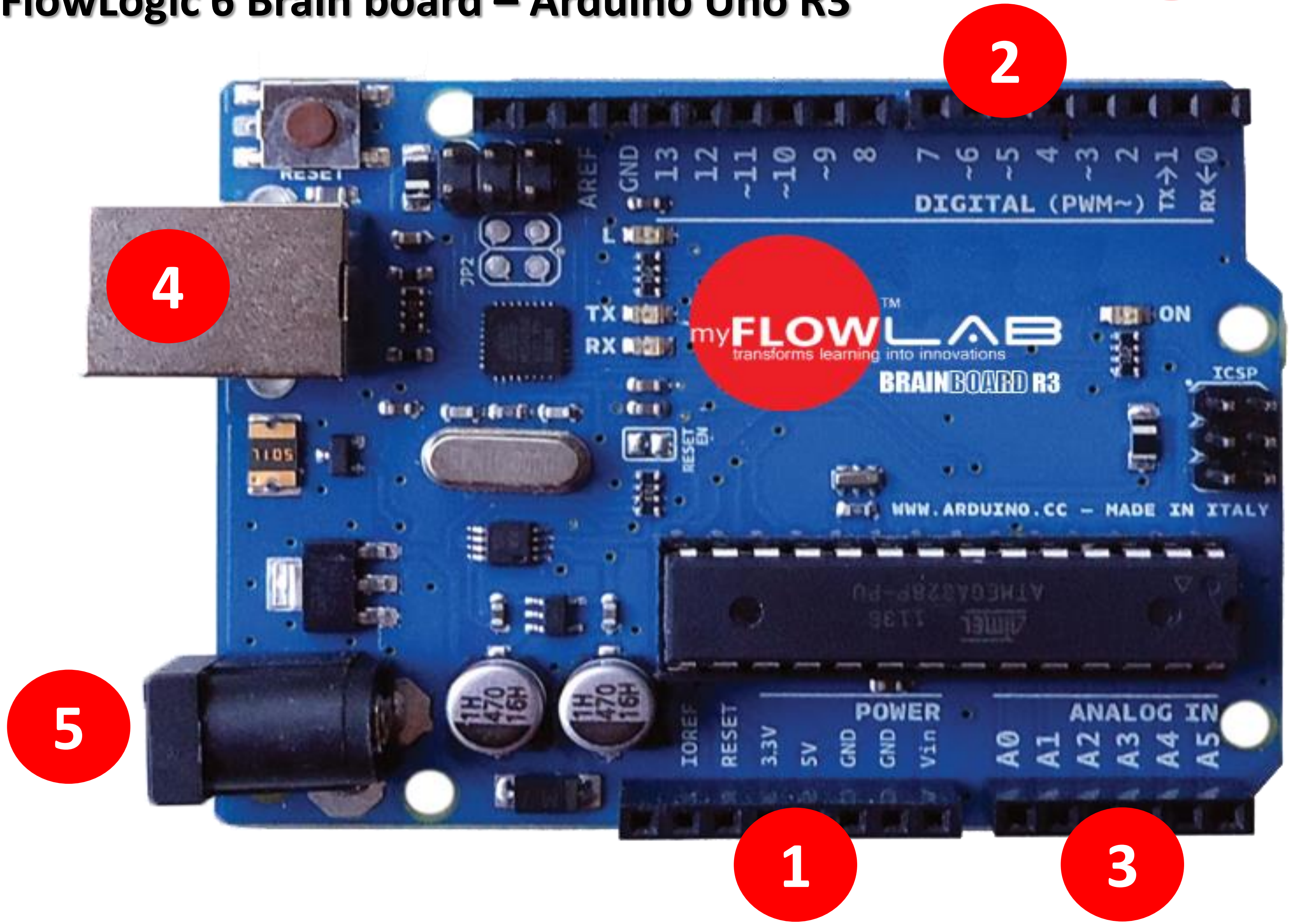
What Is Physical Computing



Physical computing means building interactive **physical** systems by the use of software and hardware that can sense and respond to the Real-World

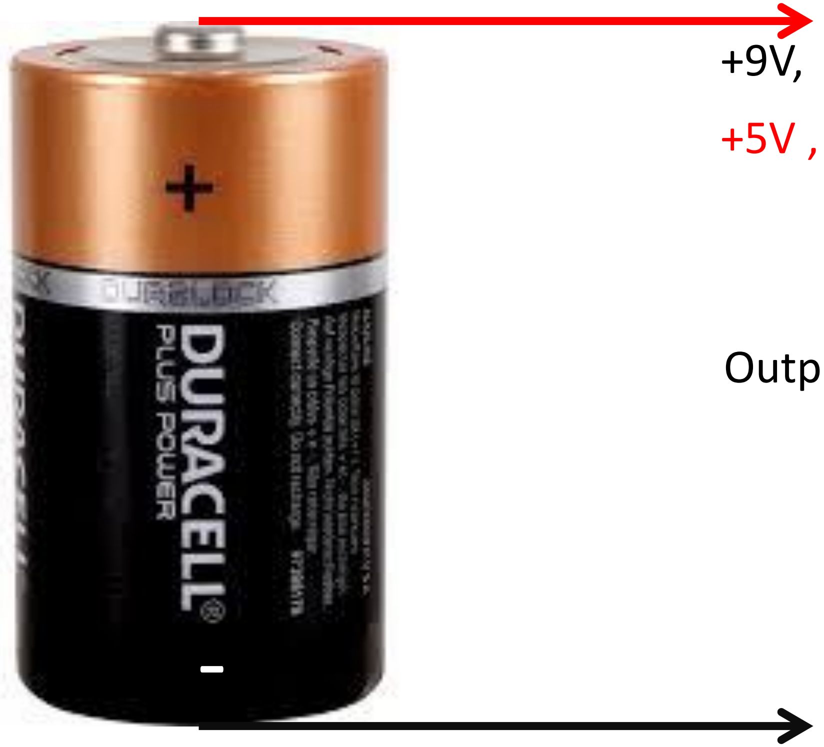
myFlowlab™

FlowLogic 6 Brain board – Arduino Uno R3



Power Source

Positive Terminal



+9V, VCC

+5V , VCC

9 volt battery

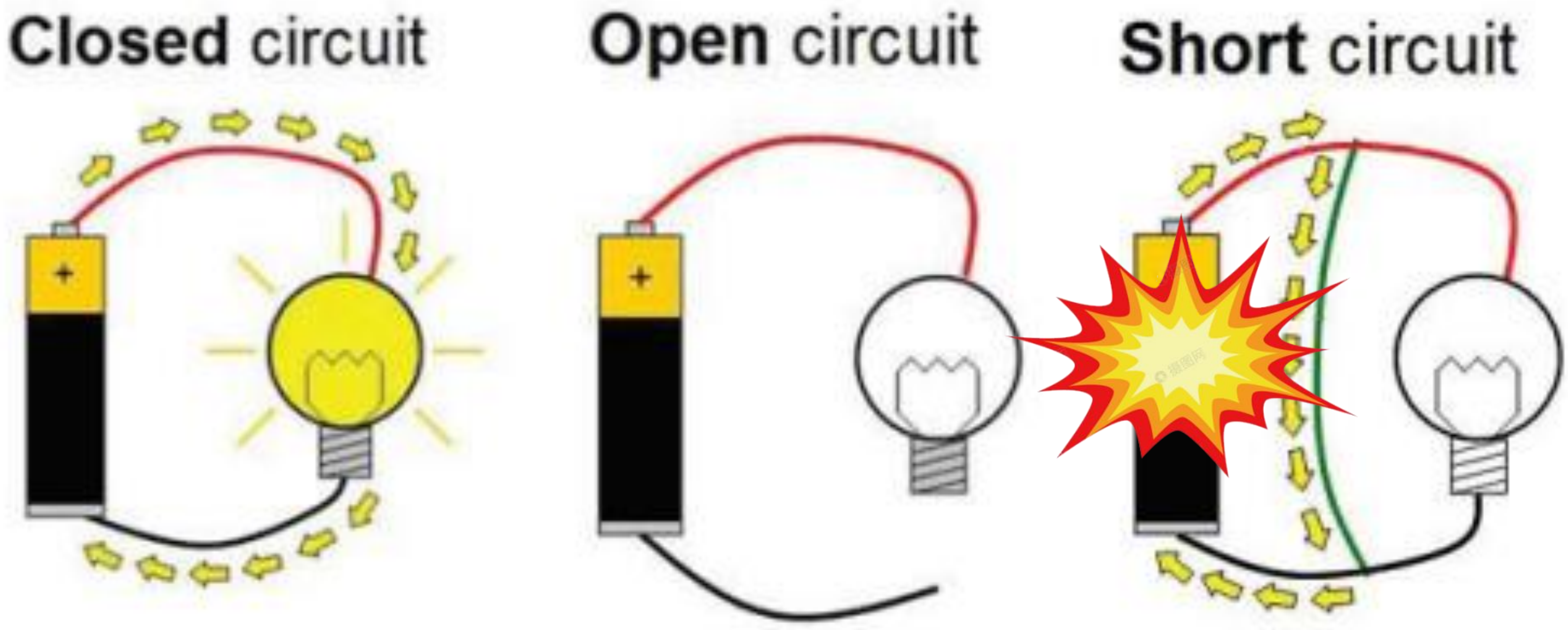
5 volt battery

Output

Negative Terminal

GND

Power Source



Power Source

myFlowlab™

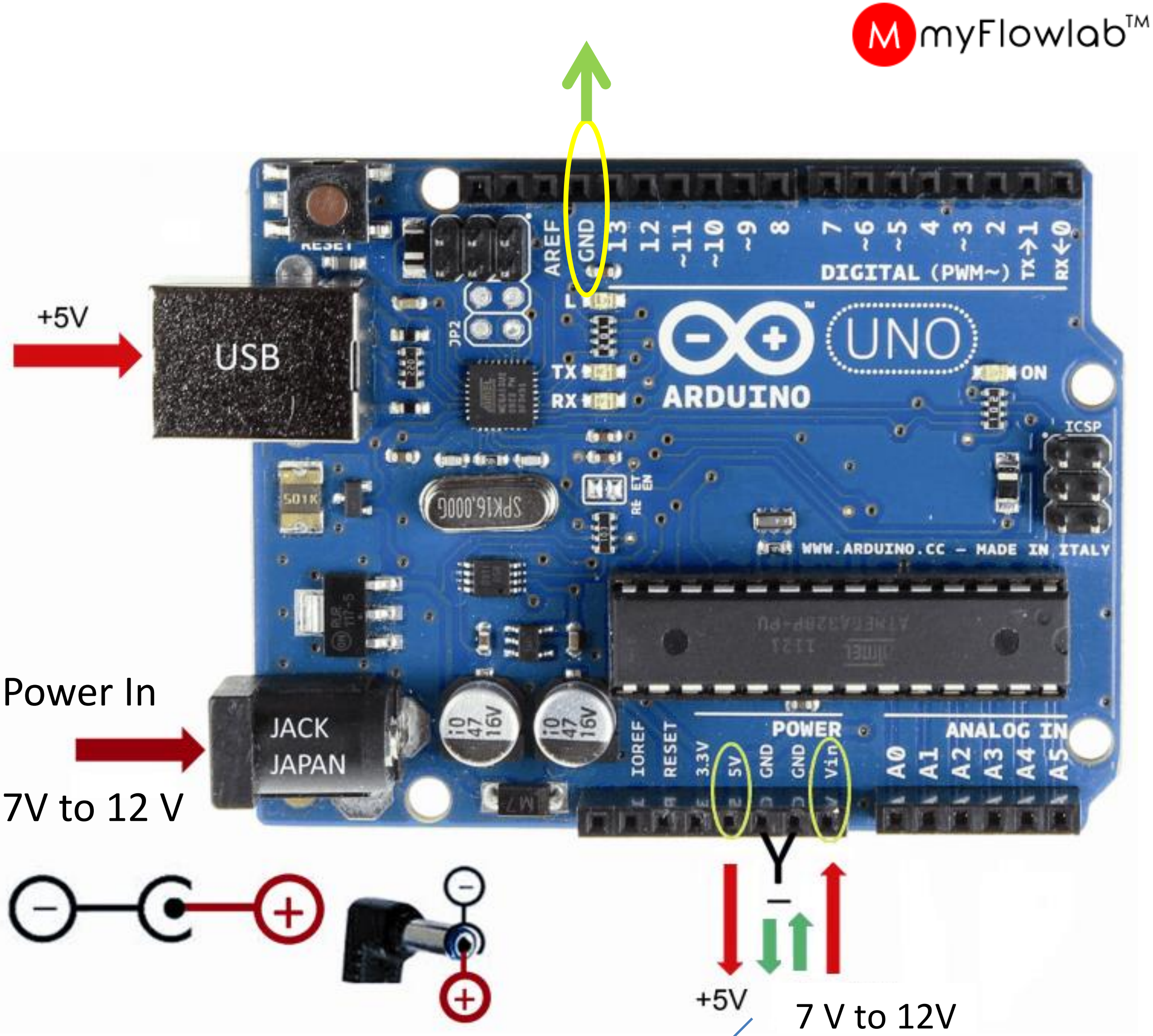
USB Cable

TO Arduino Board

To Computer

Communication and Power In

+5V



Power In

7V to 12 V

+5V

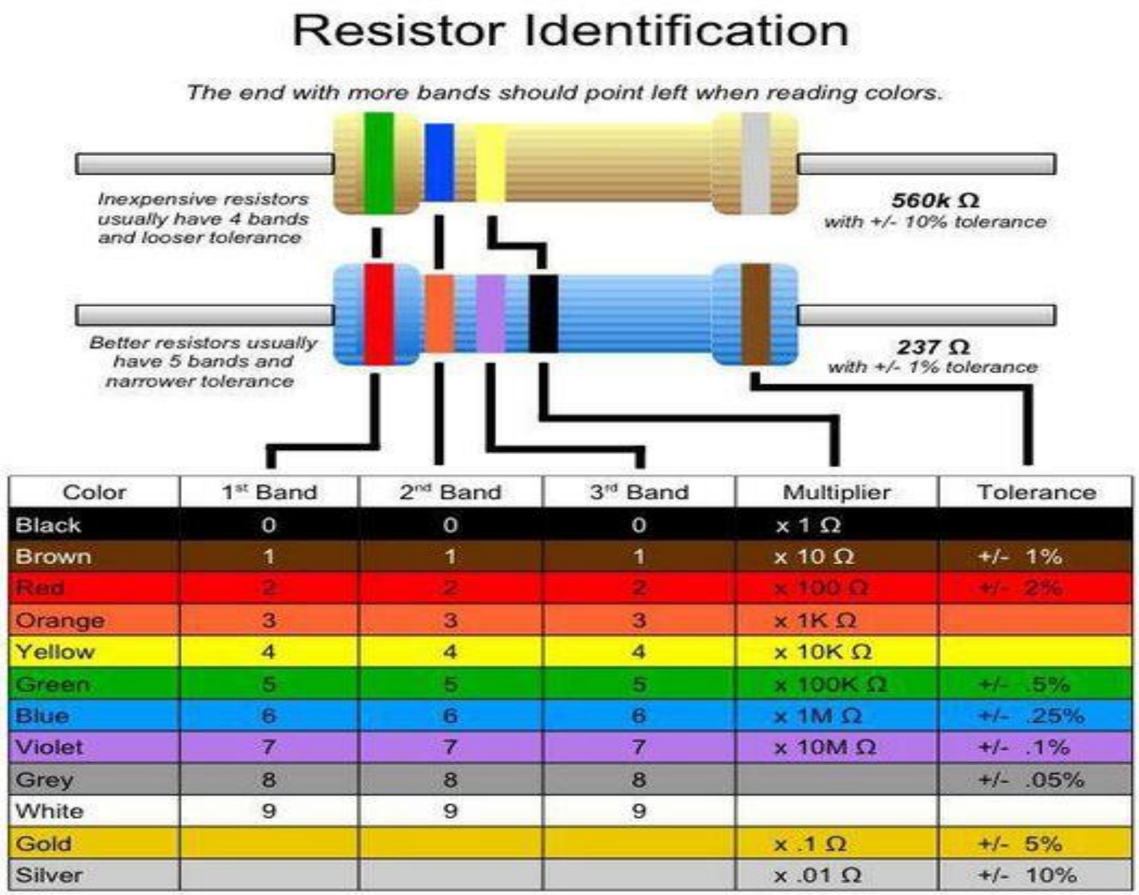
7 V to 12V
Power In

Power out to Breadboard or
Components LED, sensors, Motor….

Battery

Power Adaptor

Resistor



Resistor Identification

The end with more bands should point left when reading colors.

Inexpensive resistors usually have 4 bands and looser tolerance

560k Ω *with +/- 10% tolerance*

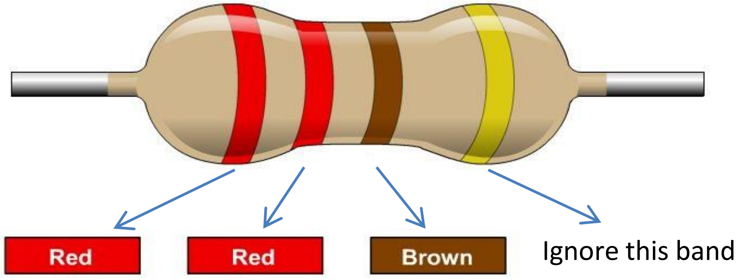
Better resistors usually have 5 bands and narrower tolerance

237 Ω *with +/- 1% tolerance*

Color	1st Band	2nd Band	3rd Band	Multiplier	Tolerance
Black	0	0	0	x 1 Ω	
Brown	1	1	1	x 10 Ω	+/- 1%
Red	2	2	2	x 100 Ω	+/- 2%
Orange	3	3	3	x 1K Ω	
Yellow	4	4	4	x 10K Ω	
Green	5	5	5	x 100K Ω	+/- .5%
Blue	6	6	6	x 1M Ω	+/- .25%
Violet	7	7	7	x 10M Ω	+/- .1%
Grey	8	8	8		+/- .05%
White	9	9	9		
Gold				x .1 Ω	+/- 5%
Silver				x .01 Ω	+/- 10%

Resistor value calculation

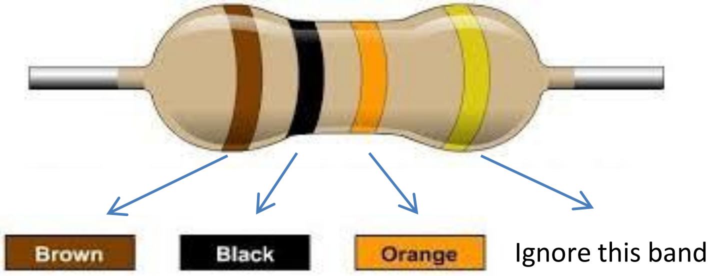
220 Ohm 4 band resistor



Red Red Brown Ignore this band

2 2 x 10 = 220 ohm

10K Ohm 4 band resistor

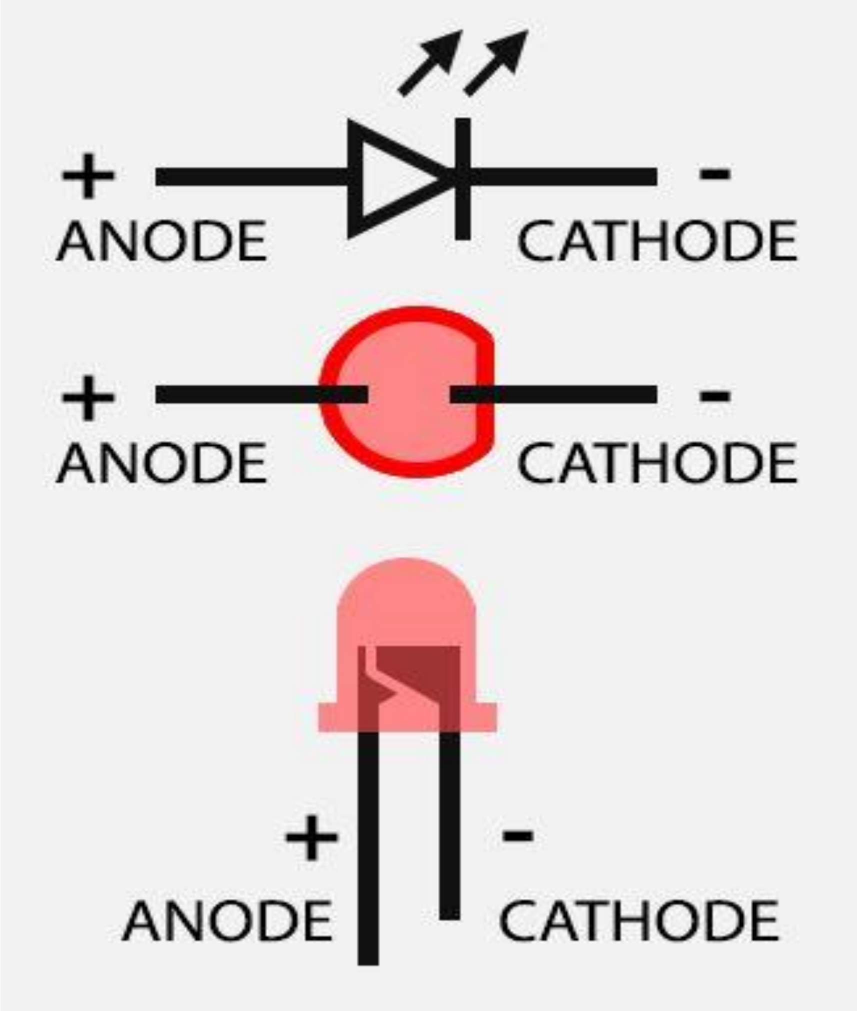


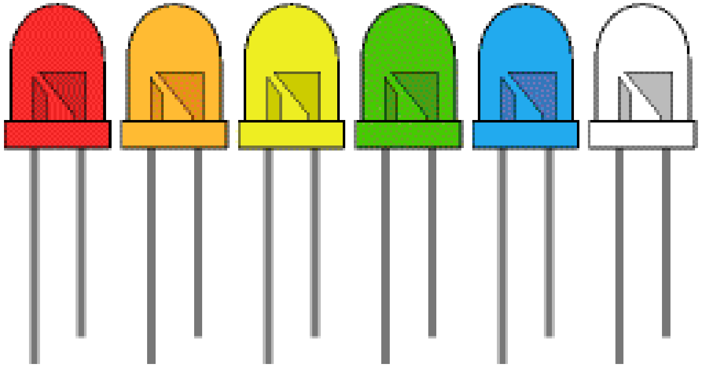
Brown Black Orange Ignore this band

1 0 x 1k = 10K ohm

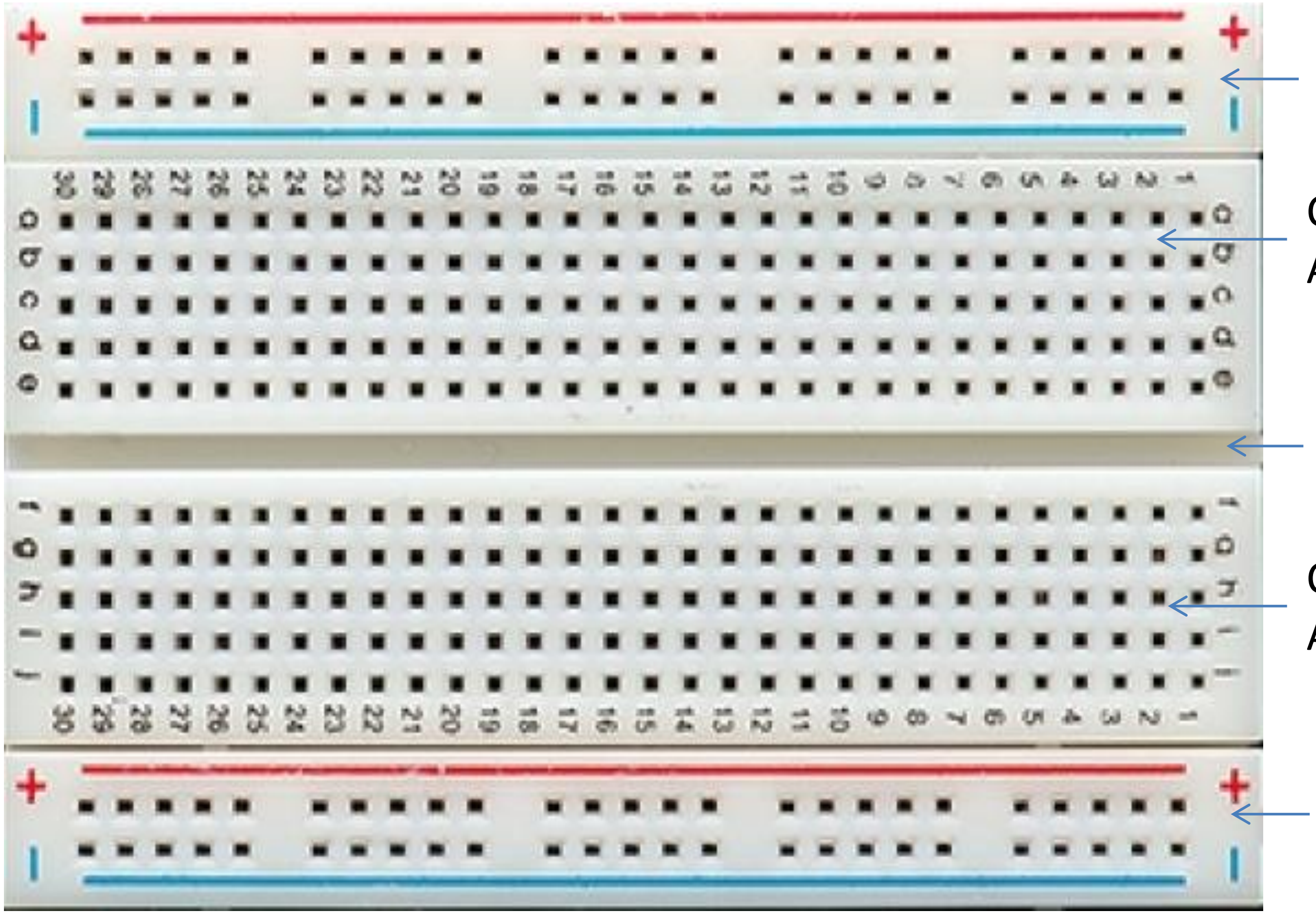
myFlowlab™

LEDs – Output Devices





Solder Less Breadboard – Half + Size



Power Rail / Bus bar

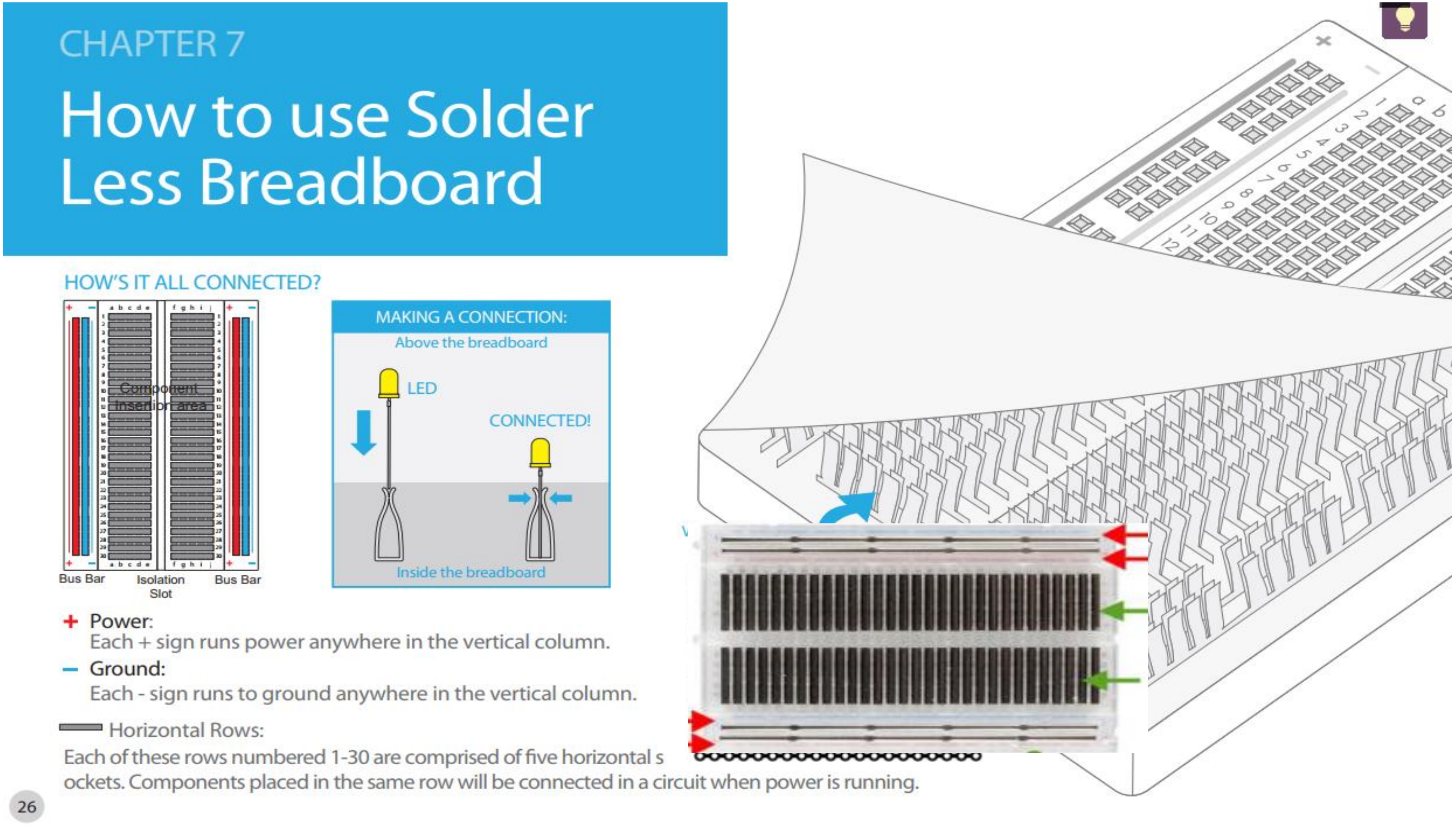
Component Insertion Area

Isolation slot

Component Insertion Area

Power Rail / Bus bar

Building Circuit using Solder Less Breadboard



CHAPTER 7

How to use Solder Less Breadboard

HOW'S IT ALL CONNECTED?

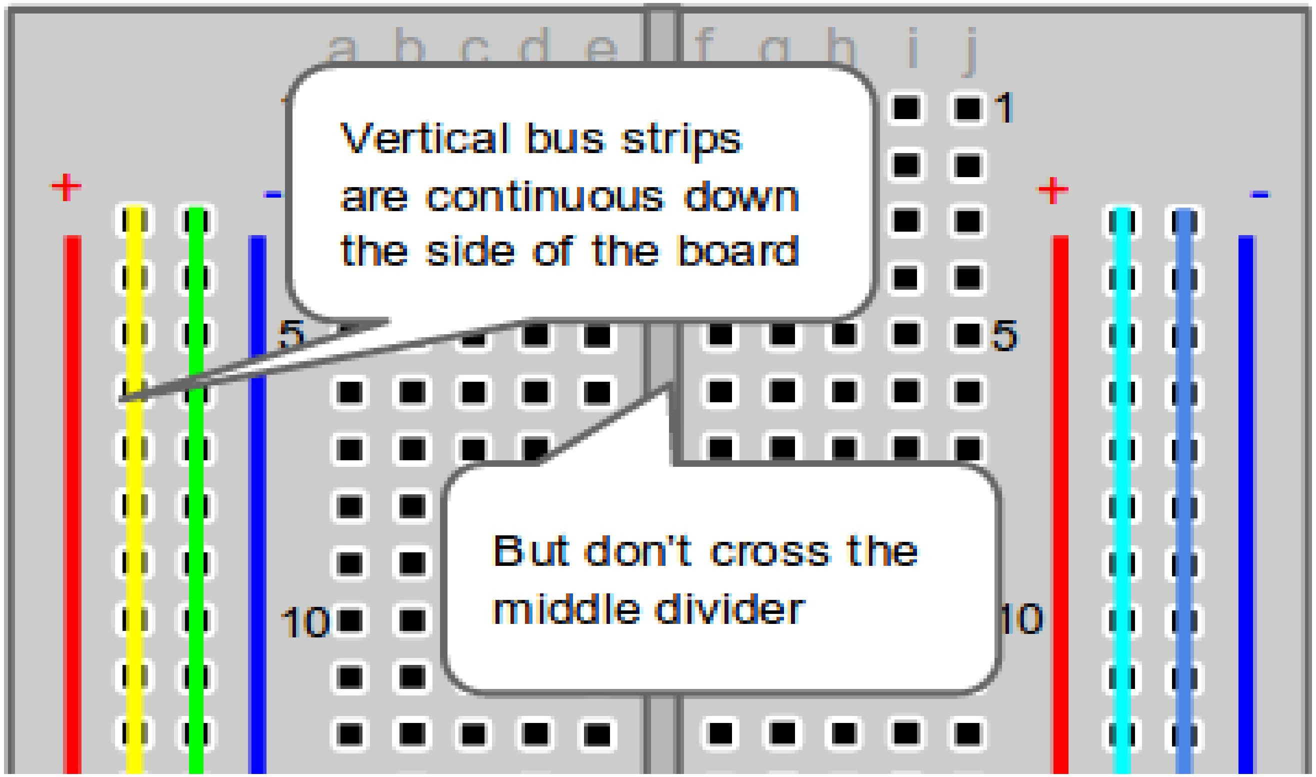
+ Power:
Each + sign runs power anywhere in the vertical column.

– Ground:
Each - sign runs to ground anywhere in the vertical column.

▬ Horizontal Rows:
Each of these rows numbered 1-30 are comprised of five horizontal s ockets. Components placed in the same row will be connected in a circuit when power is running.

myFlowlab™

Power Rails / Bus Bar

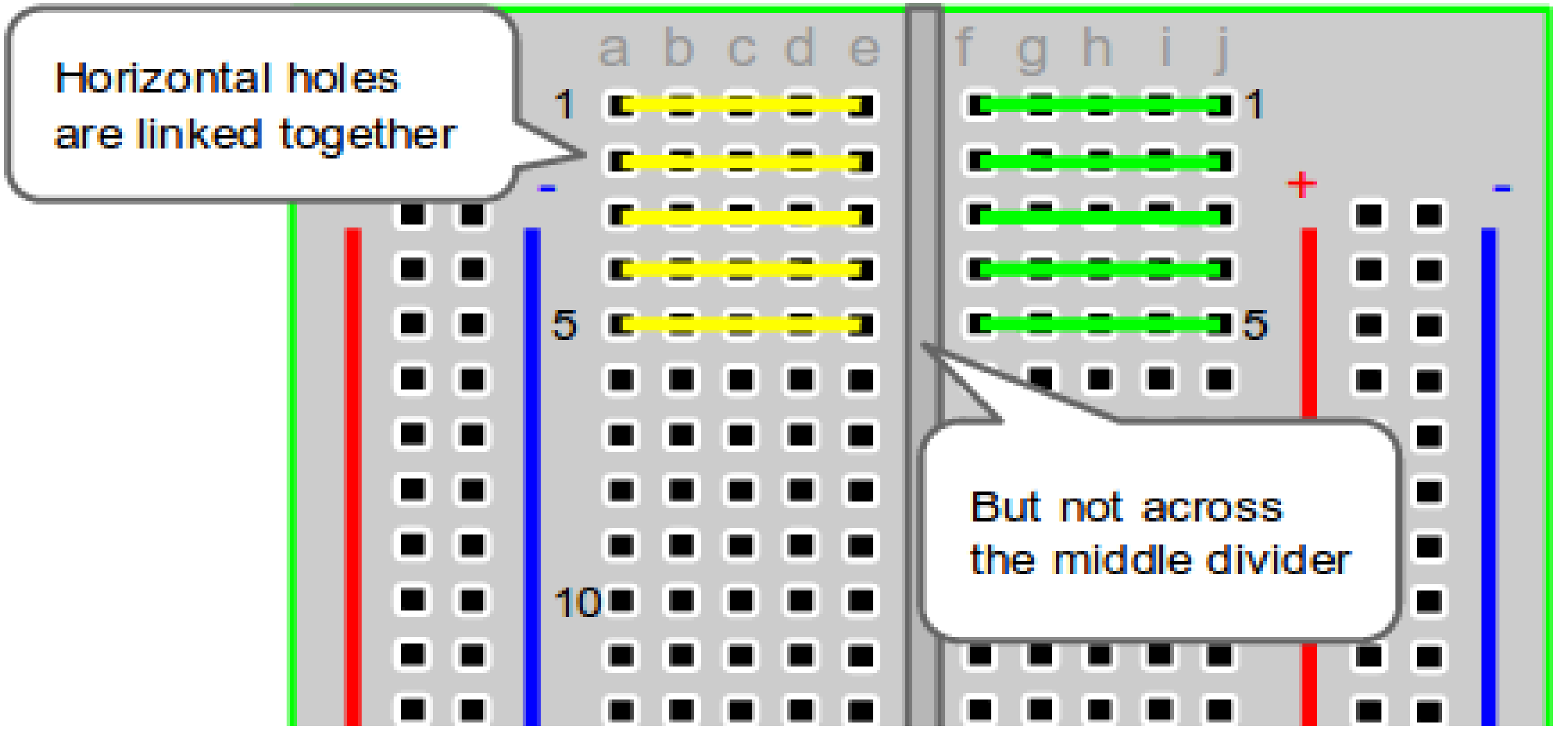


Vertical bus strips are continuous down the side of the board

But don't cross the middle divider

myFlowlab™

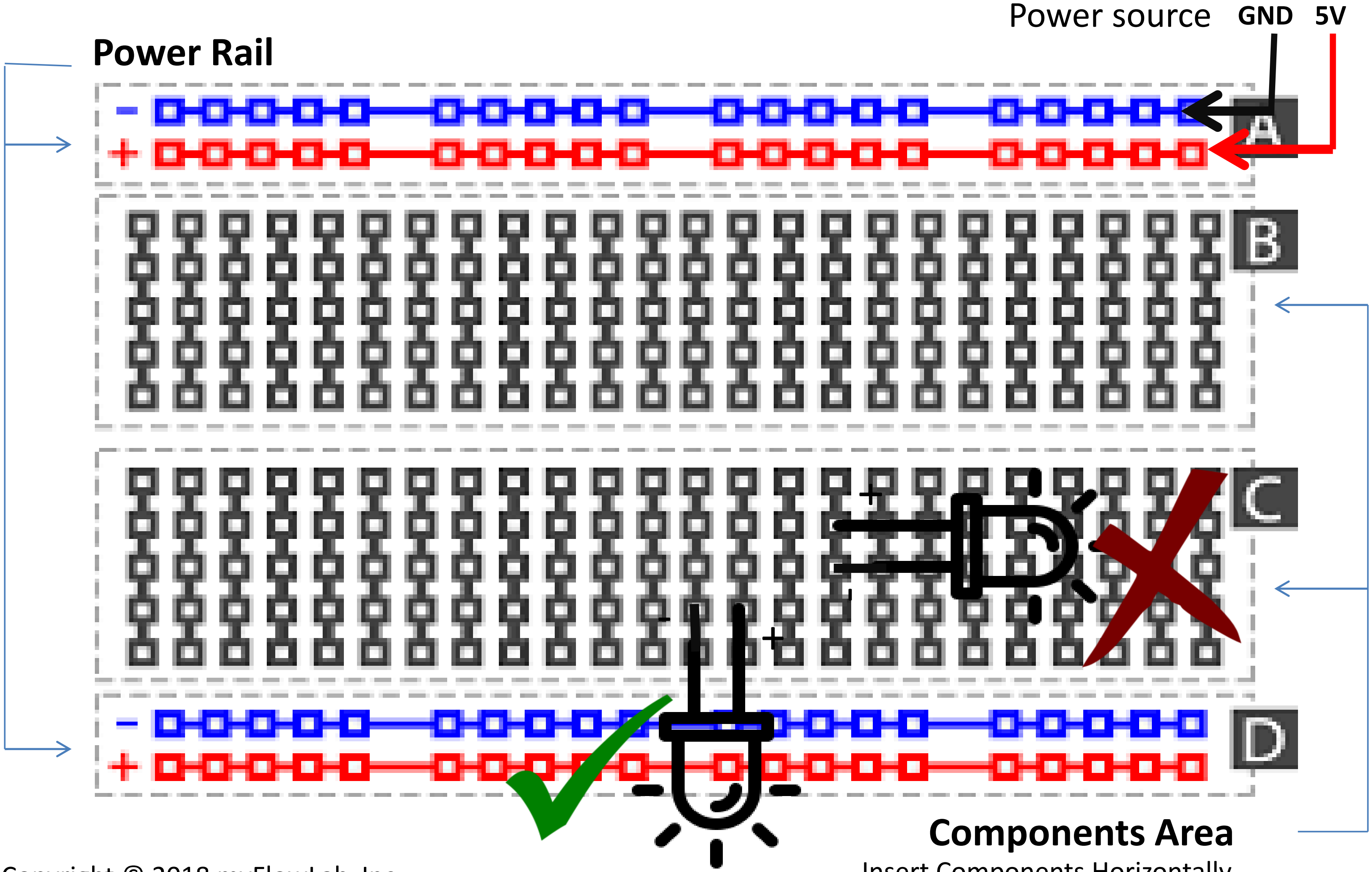
Components Terminal



Horizontal holes are linked together

But not across the middle divider

Breadboard Connection

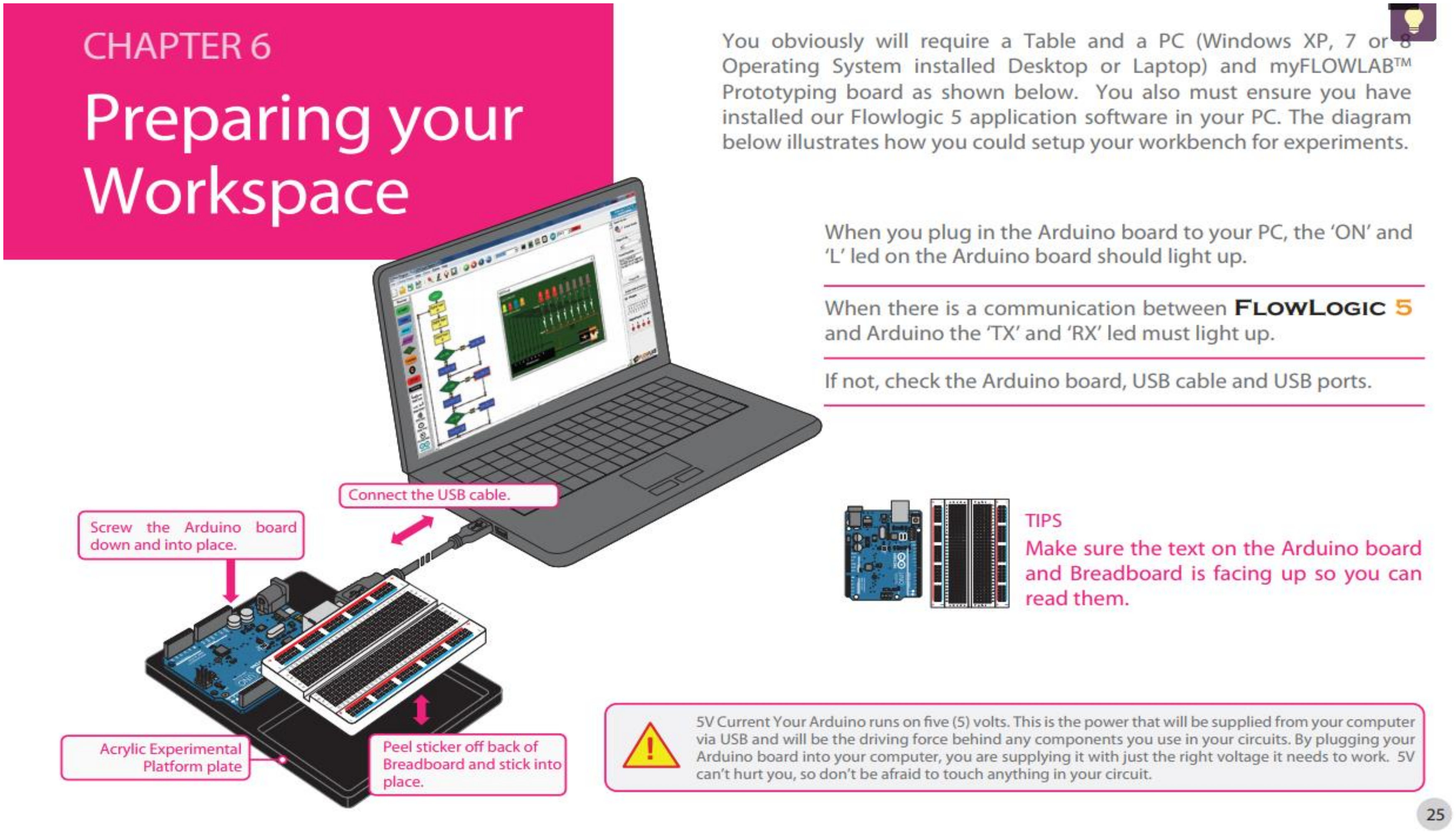


Components Area

Insert Components Horizontally

Preparing your Workspace for prototyping

myFlowlab™



CHAPTER 6

Preparing your Workspace

You obviously will require a Table and a PC (Windows XP, 7 or 8 Operating System installed Desktop or Laptop) and myFLOWLAB™ Prototyping board as shown below. You also must ensure you have installed our Flowlogic 5 application software in your PC. The diagram below illustrates how you could setup your workbench for experiments.

When you plug in the Arduino board to your PC, the 'ON' and 'L' led on the Arduino board should light up.

When there is a communication between **FLOWLOGIC 5** and Arduino the 'TX' and 'RX' led must light up.

If not, check the Arduino board, USB cable and USB ports.

Connect the USB cable.

Screw the Arduino board down and into place.

Acrylic Experimental Platform plate

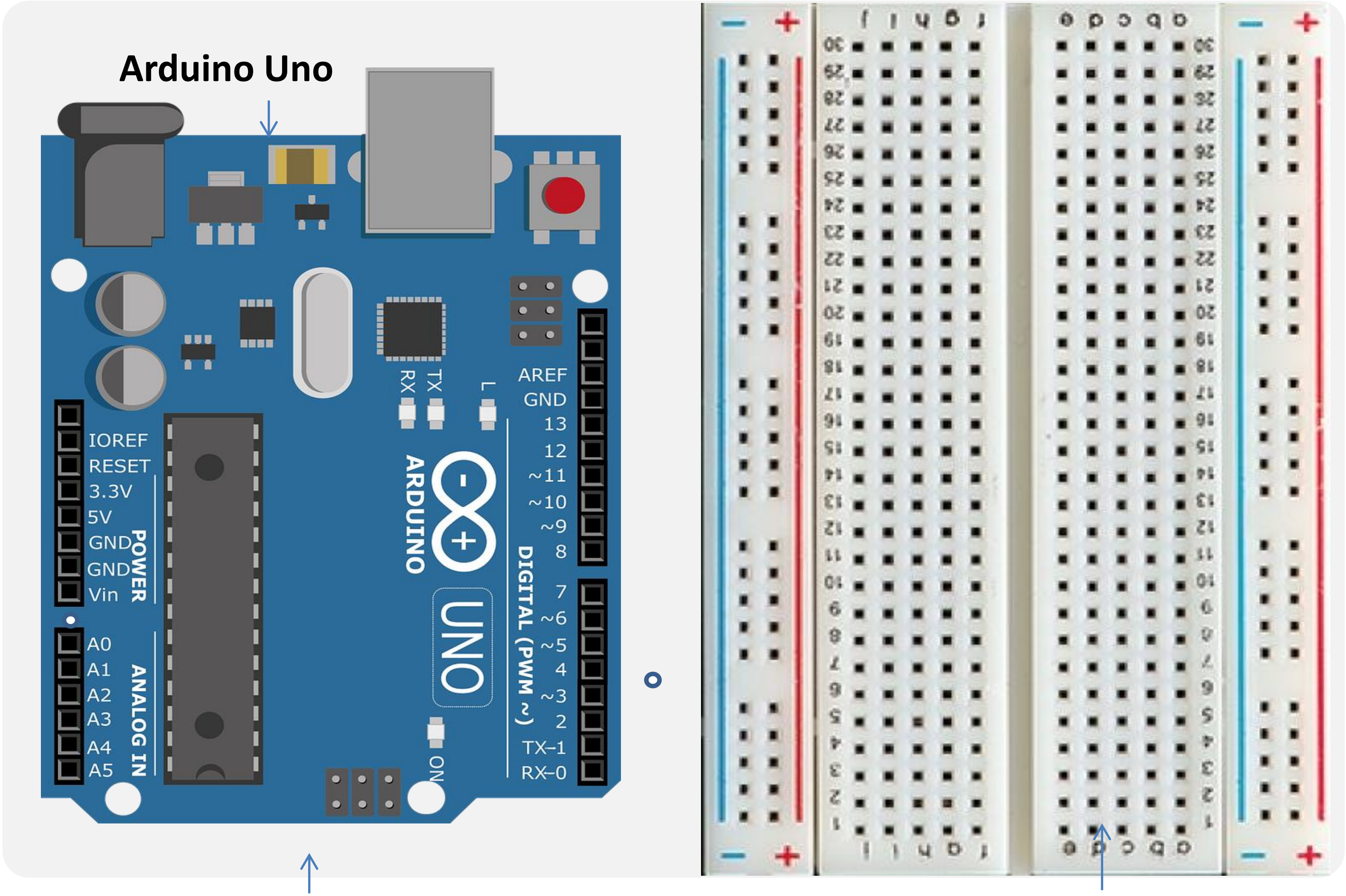
Peel sticker off back of Breadboard and stick into place.

TIPS

Make sure the text on the Arduino board and Breadboard is facing up so you can read them.

5V Current Your Arduino runs on five (5) volts. This is the power that will be supplied from your computer via USB and will be the driving force behind any components you use in your circuits. By plugging your Arduino board into your computer, you are supplying it with just the right voltage it needs to work. 5V can't hurt you, so don't be afraid to touch anything in your circuit.

Activity # 9 – Assemble the Prototyping Workspace as per Layout shown.



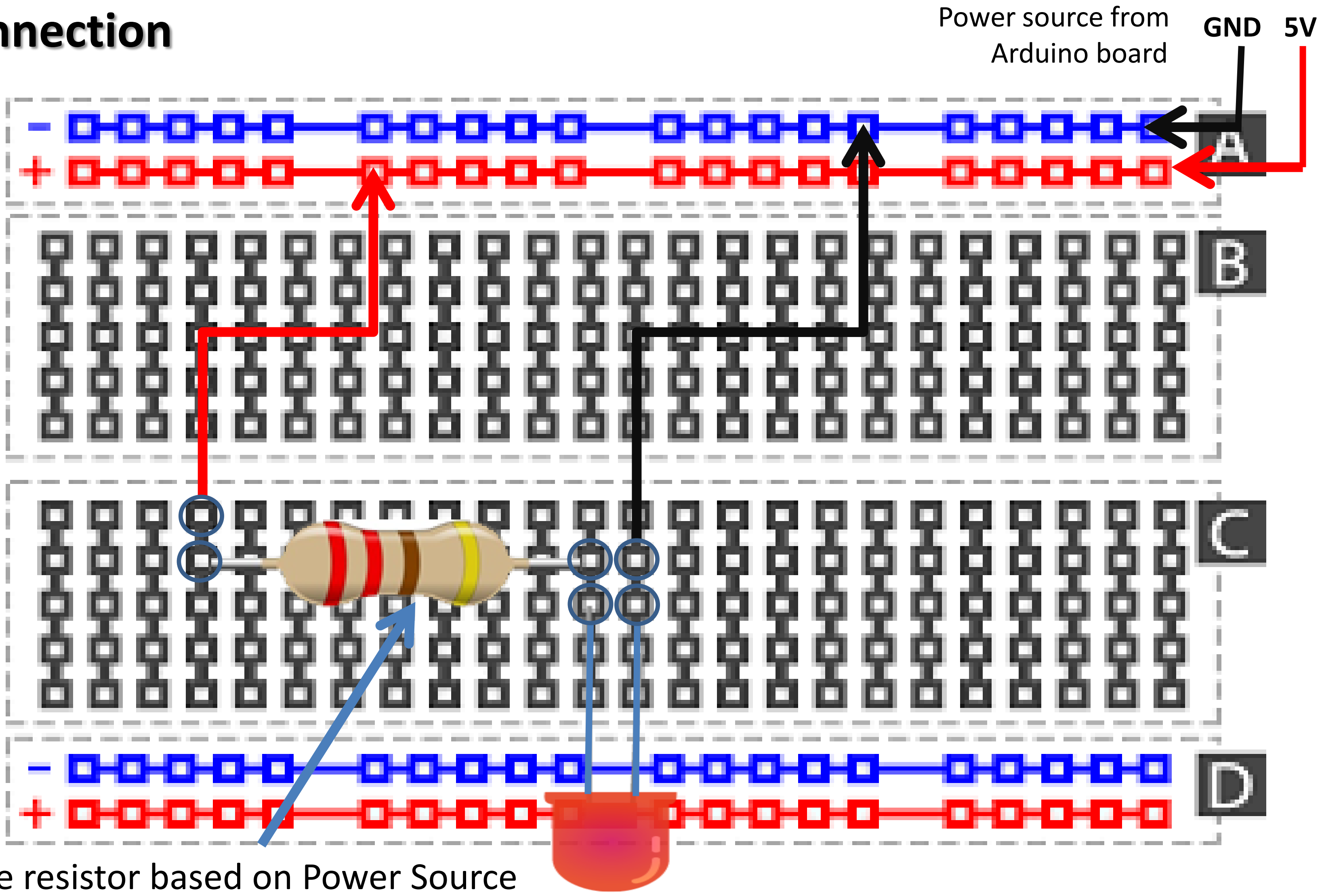
Arduino Uno

Acrylic Base Plate

400 Hole Breadboard

Activity # 10 - Circuit on Breadboard connection

myFlowlab™



Change resistor based on Power Source

220 Ohm for 3V, 470 Ohm for 5V, 1k for 9V

myFlowlab™

DIY # 7

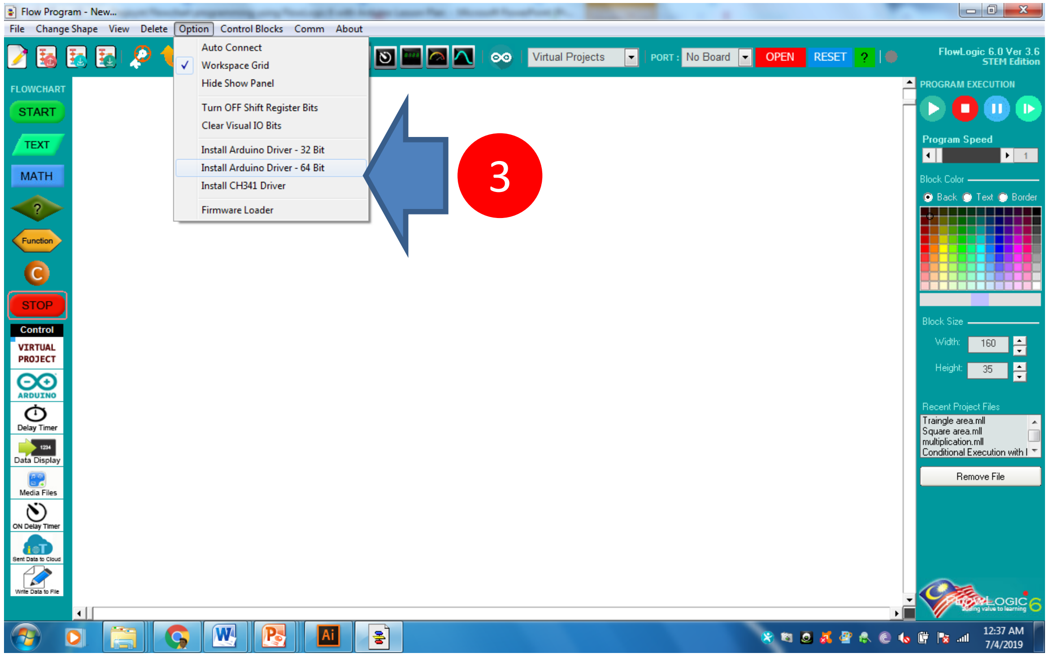
FlowLogic 6

Arduino USB Driver installation

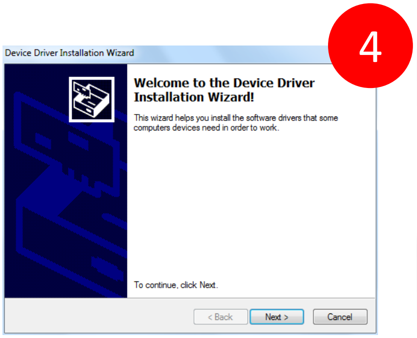
1. Launch FlowLogic 6 Version 3.6
2. *From the menu, click option*
3. *Select Install Arduino USB Driver*

   Select either 32 Bit or 64 Bit
4. *The USB Driver Installation window Should appear as shown below, if NOT, Exit FlowLogic 6 and Run it as Administrator.*

Right click on FlowLogic 6 desktop Icon and Select "Run as Administrator from the pop-menu



4



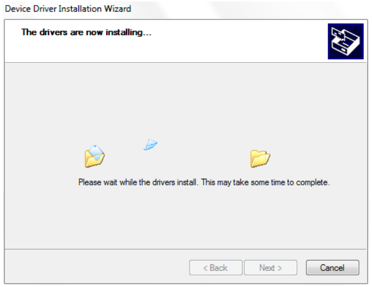
Device Driver Installation Wizard

Welcome to the Device Driver Installation Wizard!

This wizard helps you install the software drivers that some computers devices need in order to work.

To continue, click Next.

< Back | Next > | Cancel

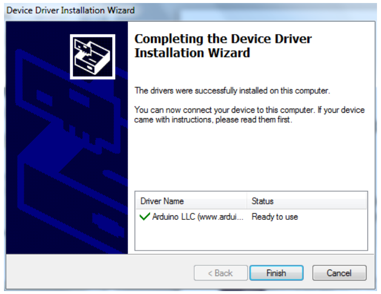


Device Driver Installation Wizard

The drivers are now installing...

Please wait while the drivers install. This may take some time to complete.

< Back | Next > | Cancel



Device Driver Installation Wizard

Completing the Device Driver Installation Wizard

The drivers were successfully installed on this computer.

You can now connect your device to this computer. If your device came with instructions, please read them first.

Driver Name	Status
✓ Arduino LLC (www.ardui...	Ready to use

< Back | Finish | Cancel

DIY # 8

myFlowlab™

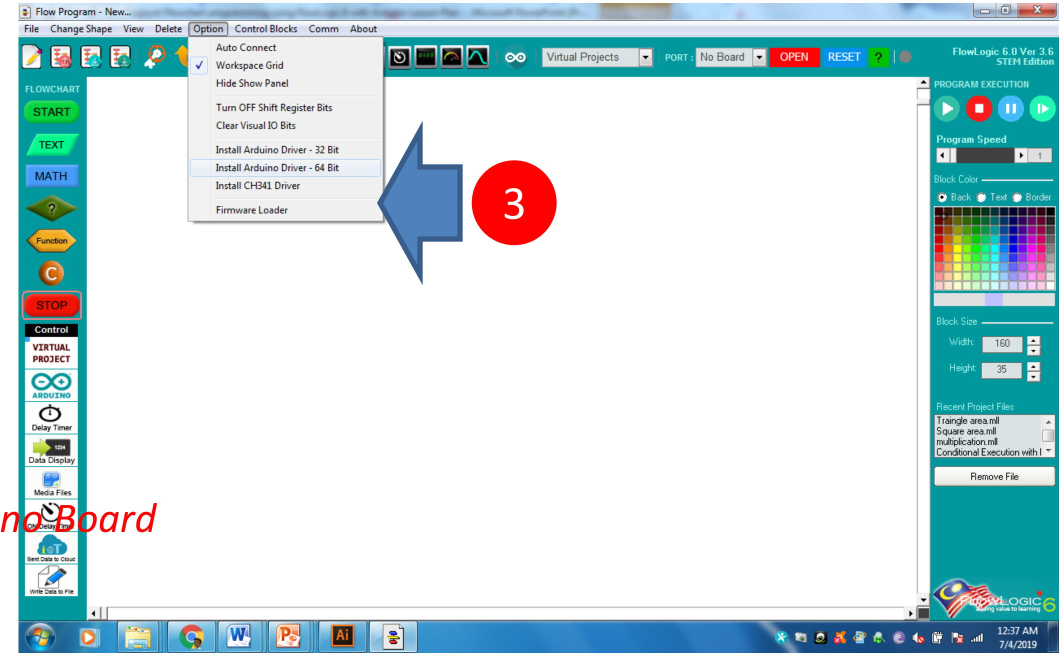
FlowLogic 6

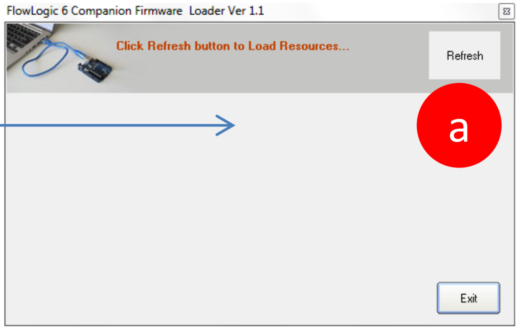
Companion Firmware upload

1. Launch FlowLogic 6 Version 3.6
2. *From the menu, click option*
3. *Select Firmware Loader*
4. *The Firmware Loader window Should appear as shown*

a. Click Refresh button to connect the Arduino Board

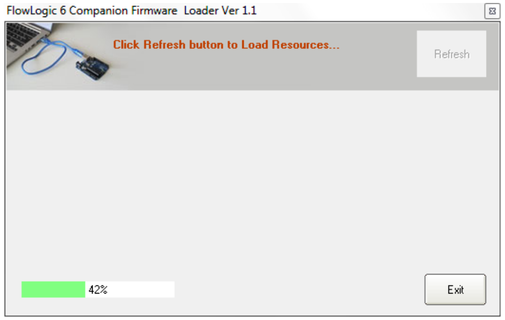
*b. Fill in your Name, email and valid **Access Code that** you have purchased and click Upload button*





FlowLogic 6 Companion Firmware Loader Ver 1.1

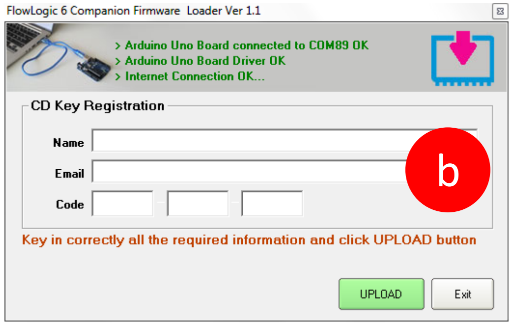
Click Refresh button to Load Resources...



FlowLogic 6 Companion Firmware Loader Ver 1.1

Click Refresh button to Load Resources...

42%



FlowLogic 6 Companion Firmware Loader Ver 1.1

> Arduino Uno Board connected to COM89 OK
> Arduino Uno Board Driver OK
> Internet Connection OK...

CD Key Registration

Name

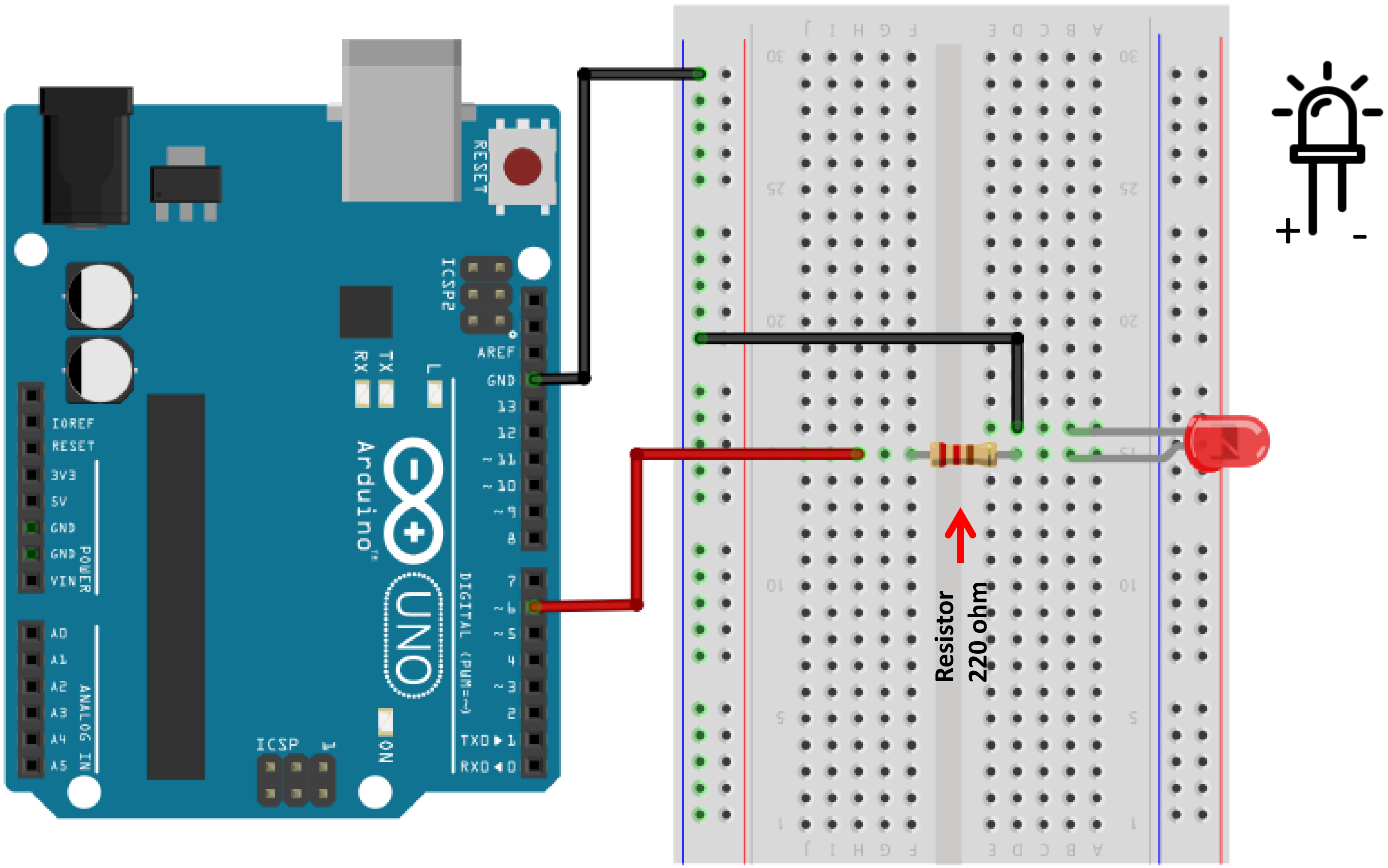
Email

Code

Key in correctly all the required information and click UPLOAD button

myFlowlab™

Activity # 11 - Single LED Connection



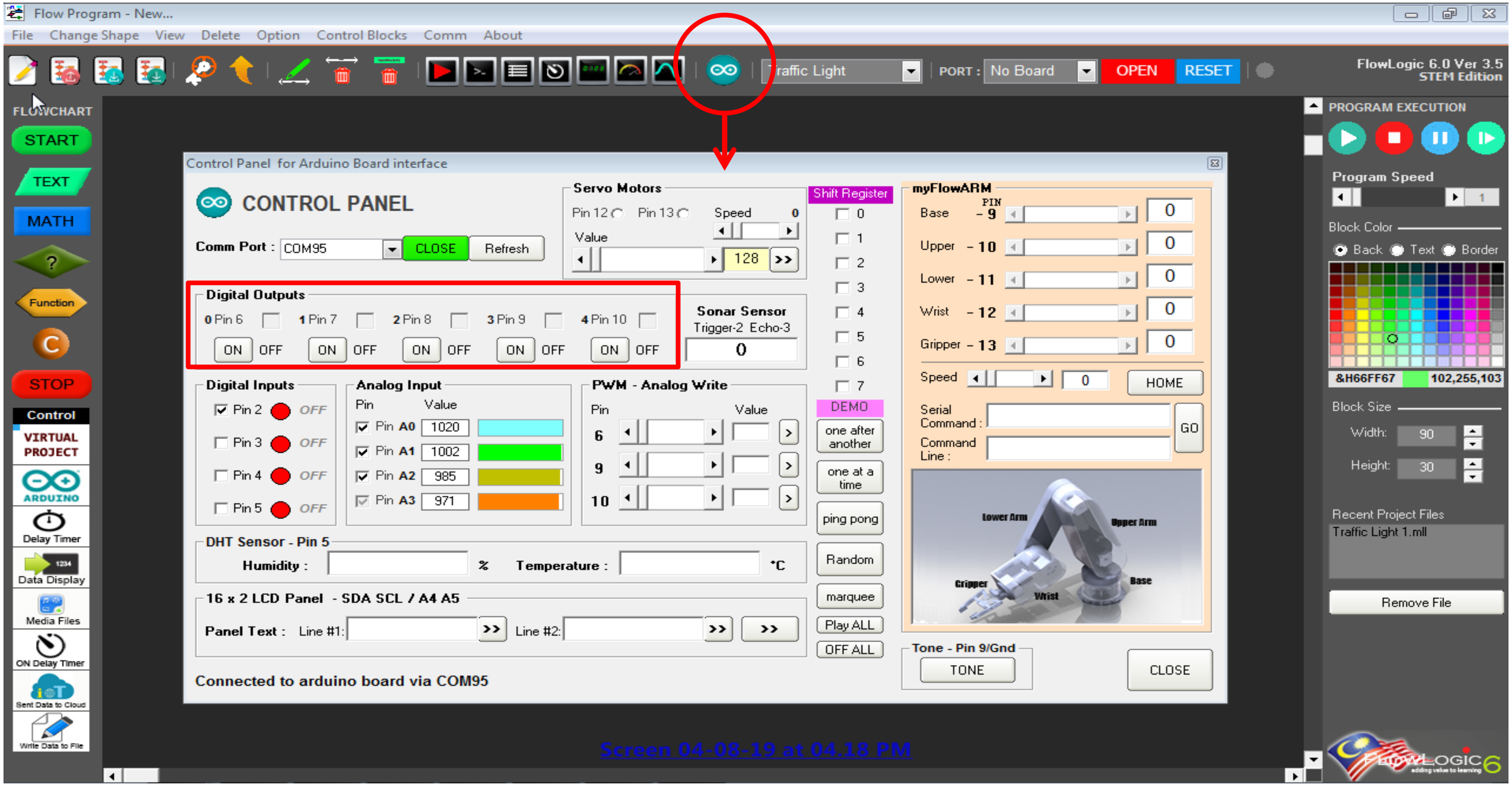
Fritzing

myFlowlab™

TESTING AND DEBUGGING

ARDUINO UNO board

To ensure connections and Components functionality are corrects before building algorithm/ Programming



Comm Port : COM95

Servo Motors — Speed 0 — Value 128

Sonar Sensor — 0

Digital Inputs: ☑ Pin 2 ● OFF; □ Pin 3 ● OFF; □ Pin 4 ● OFF; □ Pin 5 ● OFF

Pin	Value
☑ Pin A0	1020
☑ Pin A1	1002
☑ Pin A2	985
☑ Pin A3	971

PWM - Analog Write — Pin 6, 9, 10

Shift Register: □ 0 □ 1 □ 2 □ 3 □ 4 □ 5 □ 6 □ 7

myFlowARM	PIN	Value
Base	- 9	0
Upper	- 10	0
Lower	- 11	0
Wrist	- 12	0
Gripper	- 13	0
Speed		0

Connected to arduino board via COM95

Screen 04-08-19 at 04.18 PM

Block Color: ● Back ○ Text ○ Border — &H66FF67 — 102,255,103

Width: 90 — Height: 30

Recent Project Files: Traffic Light 1.mll

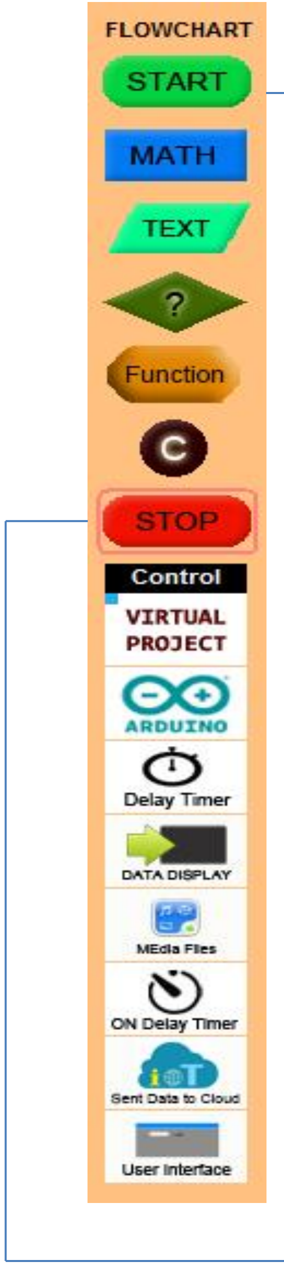
Activity # 12

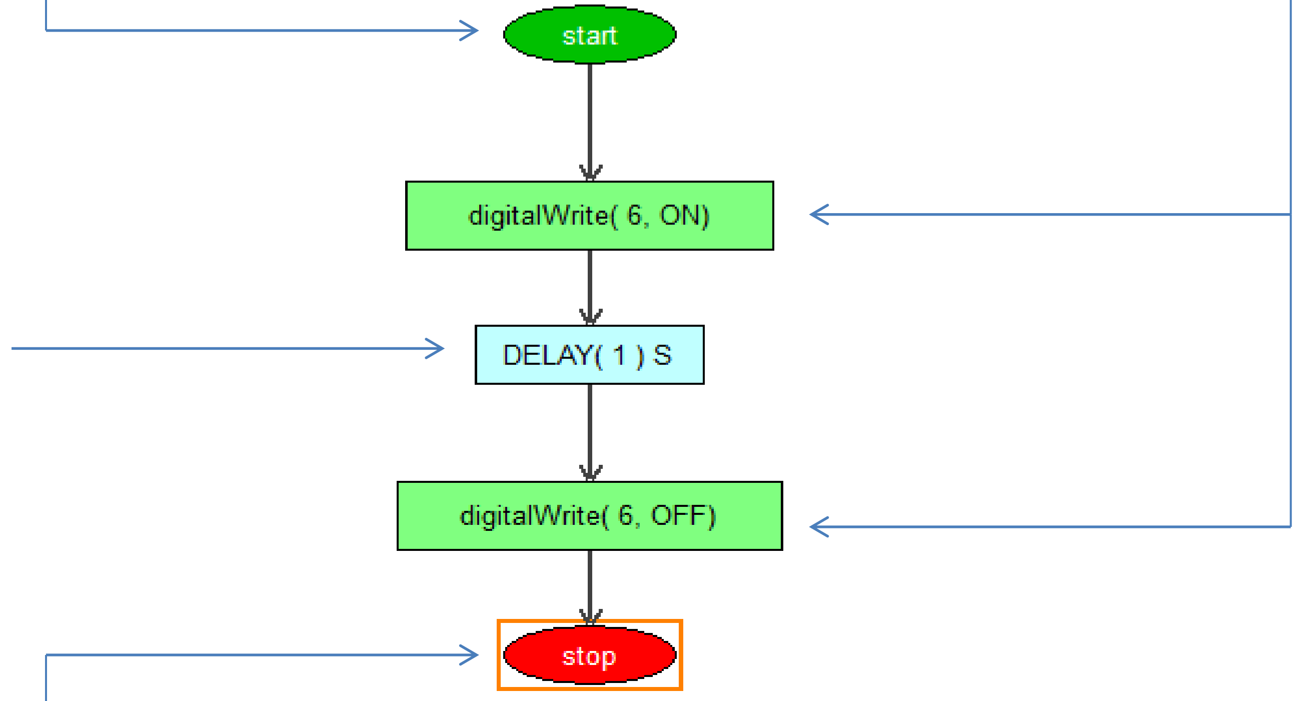
Algorithm/ FlowProgram

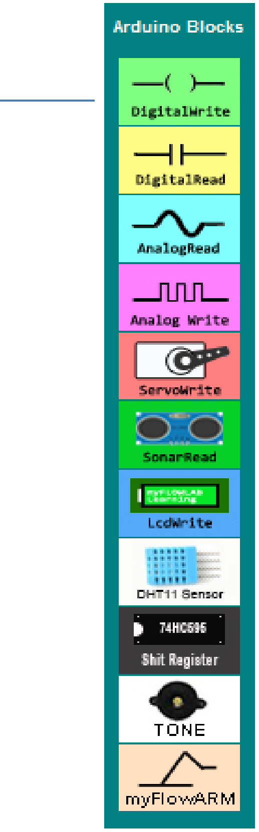
LED ON/OFF (Digital Output)

Single Loop

Arduino Command blocks







Activity # 13

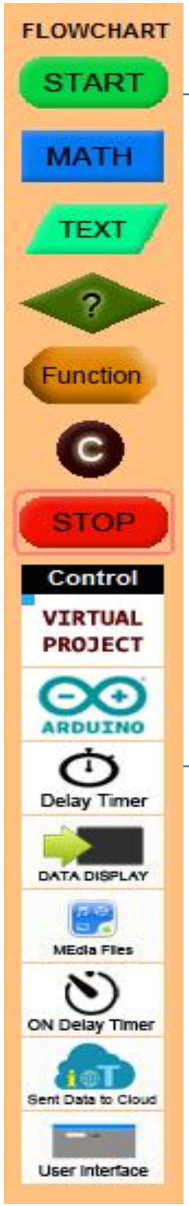
Algorithm/ FlowProgram

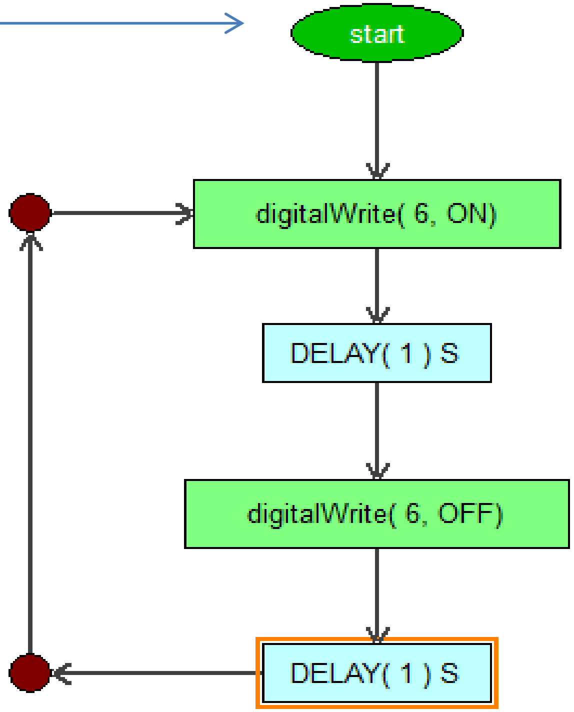
LED Blink

(Digital Output)

Continuous Loop

MmyFlowlab™





start

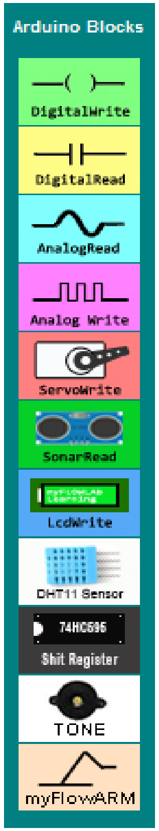
digitalWrite(6, ON)

DELAY(1) S

digitalWrite(6, OFF)

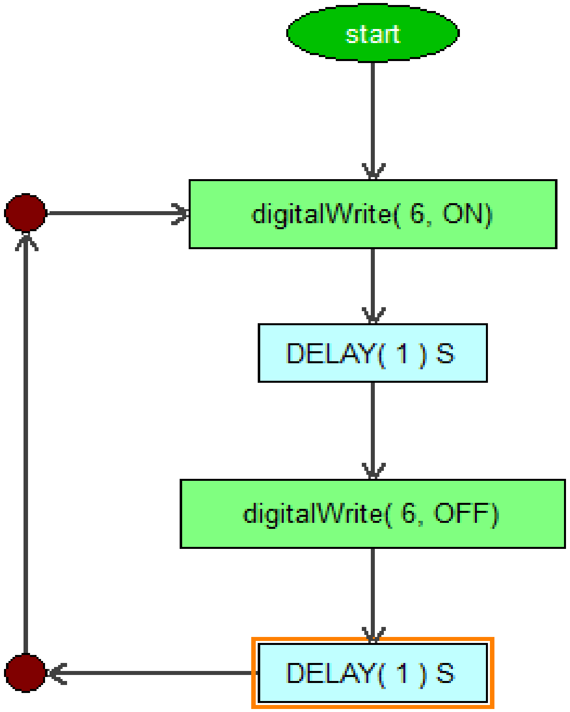
DELAY(1) S

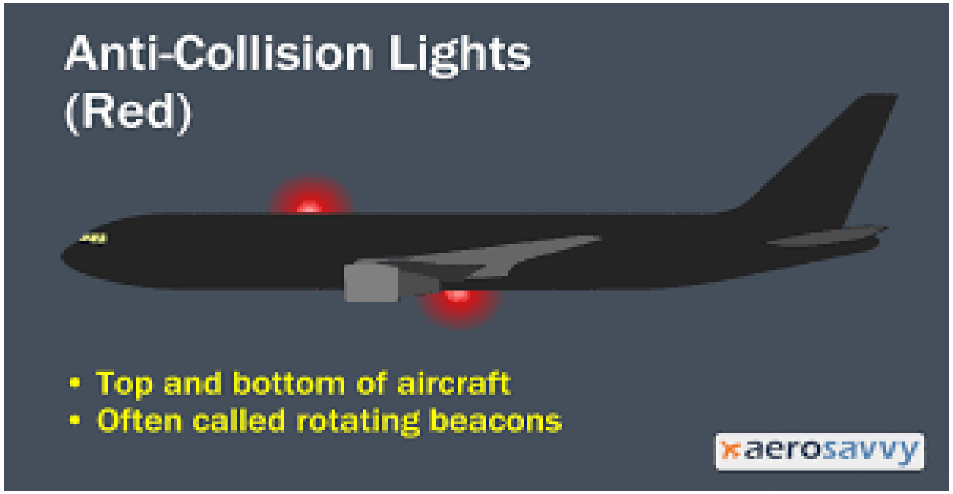
Arduino Command blocks



myFlowlab™

DIY # 9 - Modify the Algorithm/FlowProgram for Aircraft Anti-Collision Strobe Light




Anti-Collision Lights (Red)
Top and bottom of aircraft
Often called rotating beacons
aerosavvy

Let the students watch the Aircraft Beacon strobe light video before attempting this project

Explain the aircraft anti-collision light project, get them to change the first delay value to 0.02 sec and second delay value to 2 sec and Run it.

Activity # 14

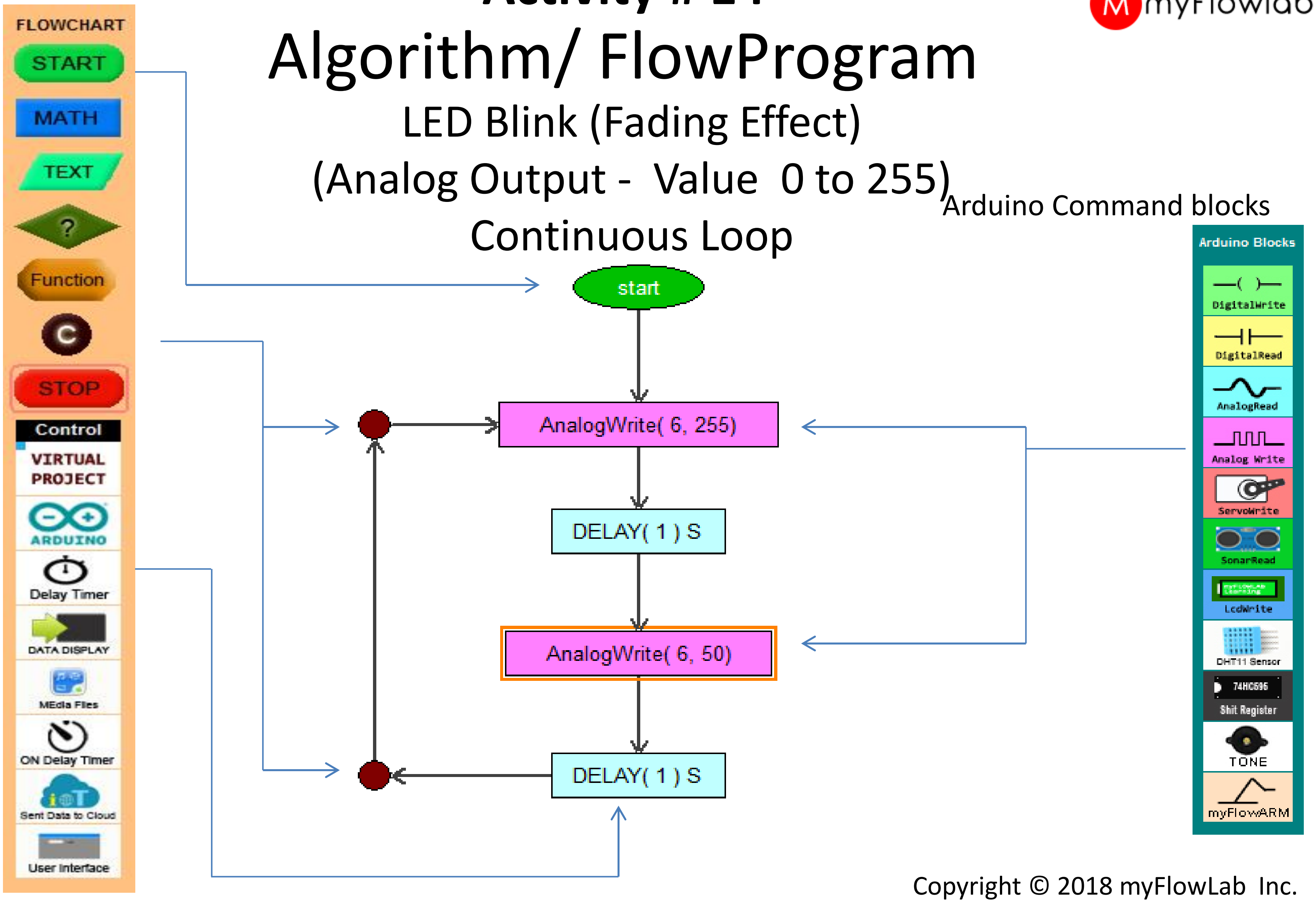
Algorithm/ FlowProgram

LED Blink (Fading Effect)

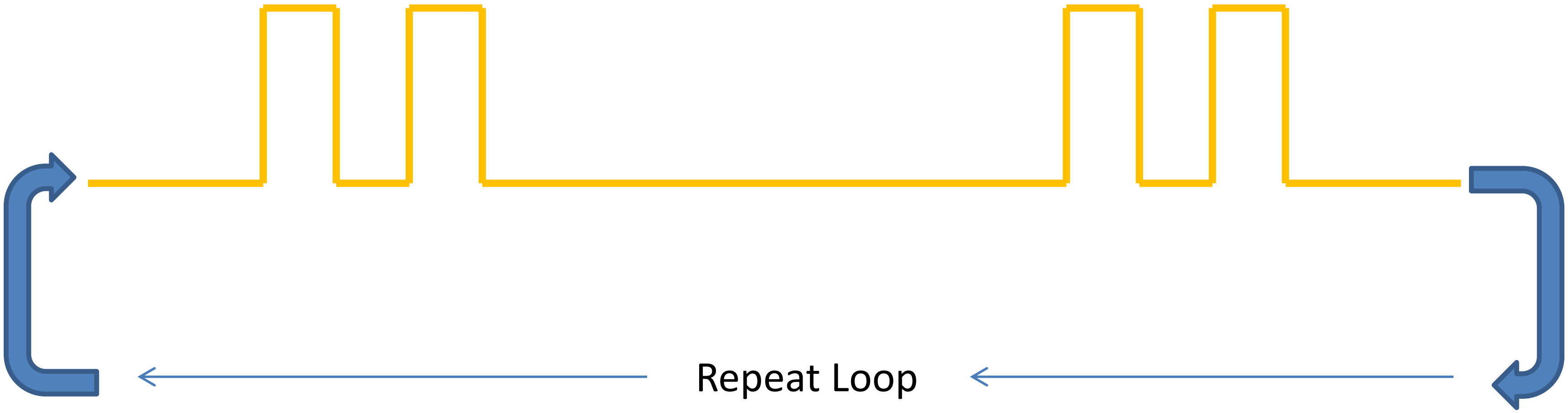
(Analog Output - Value 0 to 255)

Continuous Loop

myFlowlab™



DIY # 10 – Develop an Algorithm / FlowProgram to blink the Led at a interval based on the profile below



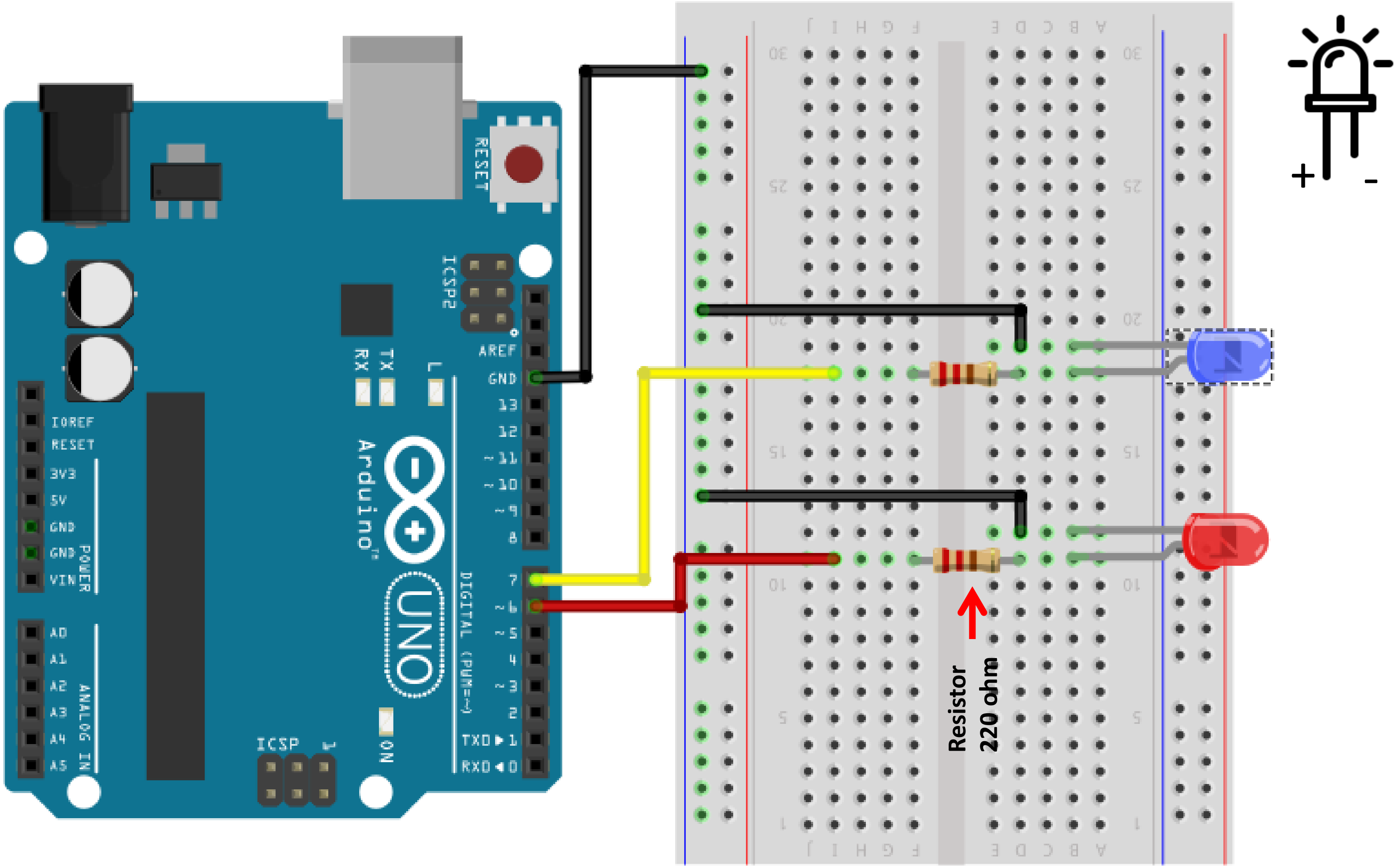
Variations : Create the same effect by adjusting the delay frequency

Building Real-World Digital Output project

2- LEDs
Police car Siren

myFlowlab™

Activity # 15 - 2 LED Connection



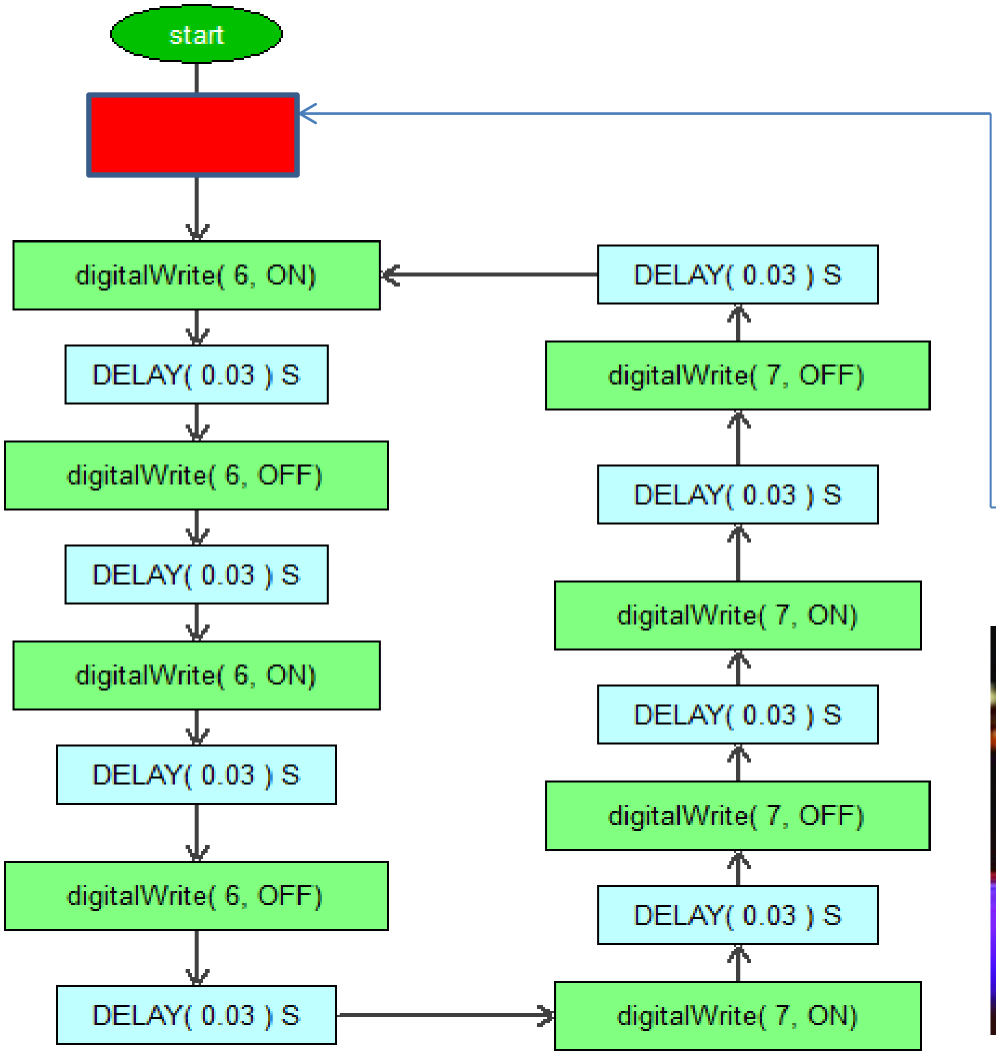
Resistor
220 ohm

Fritzing

Police Car Siren Project

Activity # 16 - Algorithm/FlowProgram for Police Car with Audio

myFlowlab™



start

digitalWrite(6, ON)

DELAY(0.03) S

digitalWrite(6, OFF)

DELAY(0.03) S

digitalWrite(6, ON)

DELAY(0.03) S

digitalWrite(6, OFF)

DELAY(0.03) S

digitalWrite(7, ON)

DELAY(0.03) S

digitalWrite(7, OFF)

DELAY(0.03) S

digitalWrite(7, ON)

DELAY(0.03) S

digitalWrite(7, OFF)

DELAY(0.03) S

Police Car Siren Light Project

Guide student to write a program to double blink each LED with slow delay

Edit the Delay for the LEDs to blink like Police car siren light.

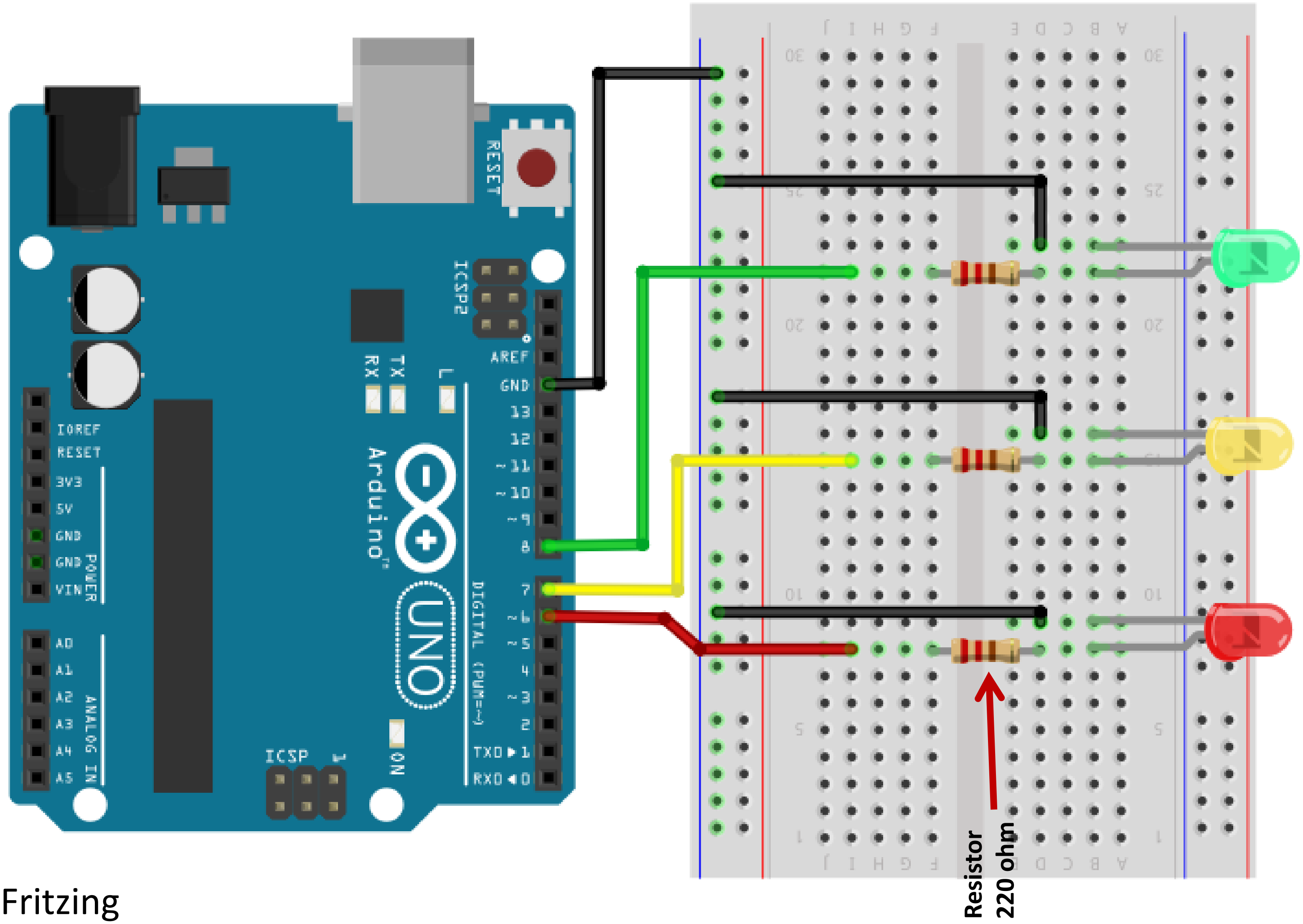
Get them to place the Media command block (Police-Siren) in a correct flow.

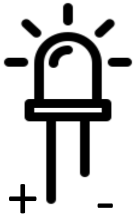
Building Real-World Digital Output project

3- LEDs
Traffic Light system

Activity # 17 - 3 LED Connection

myFlowlab™

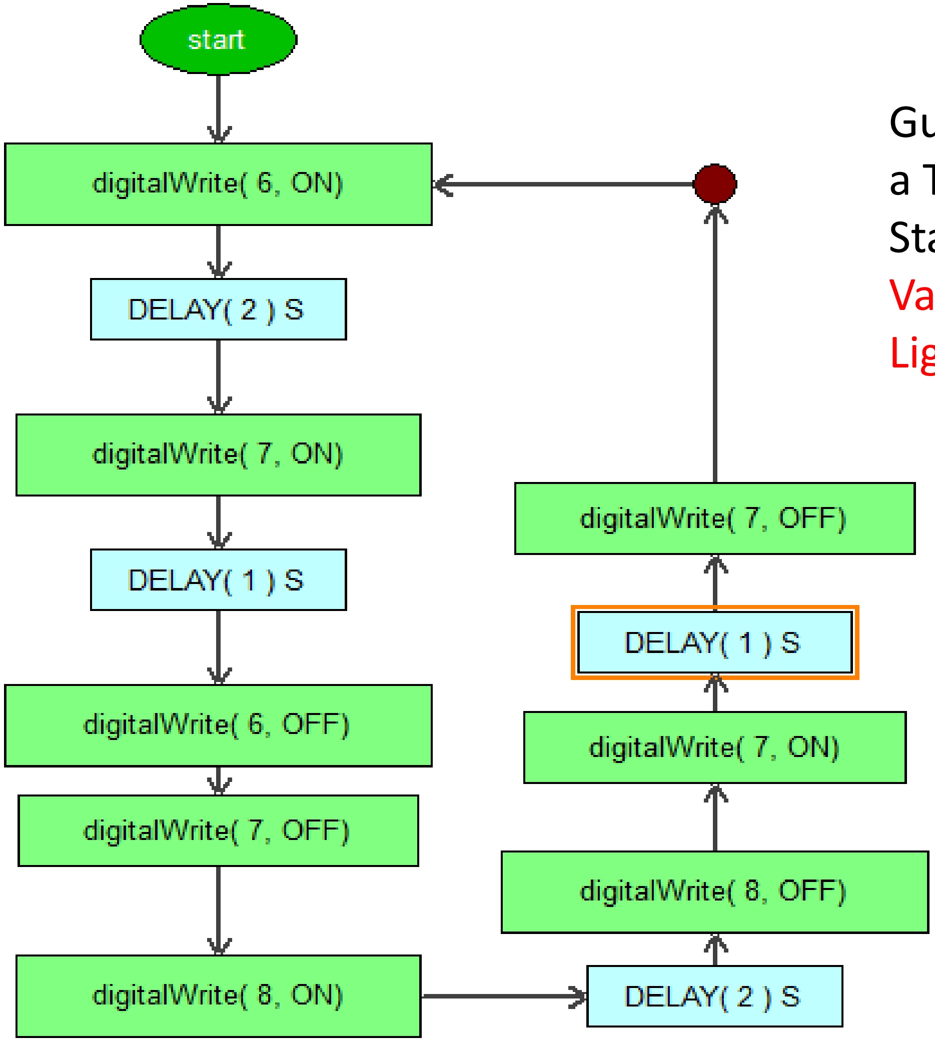




Fritzing

Activity # 19 - Algorithm/FlowProgram for Traffic Light system

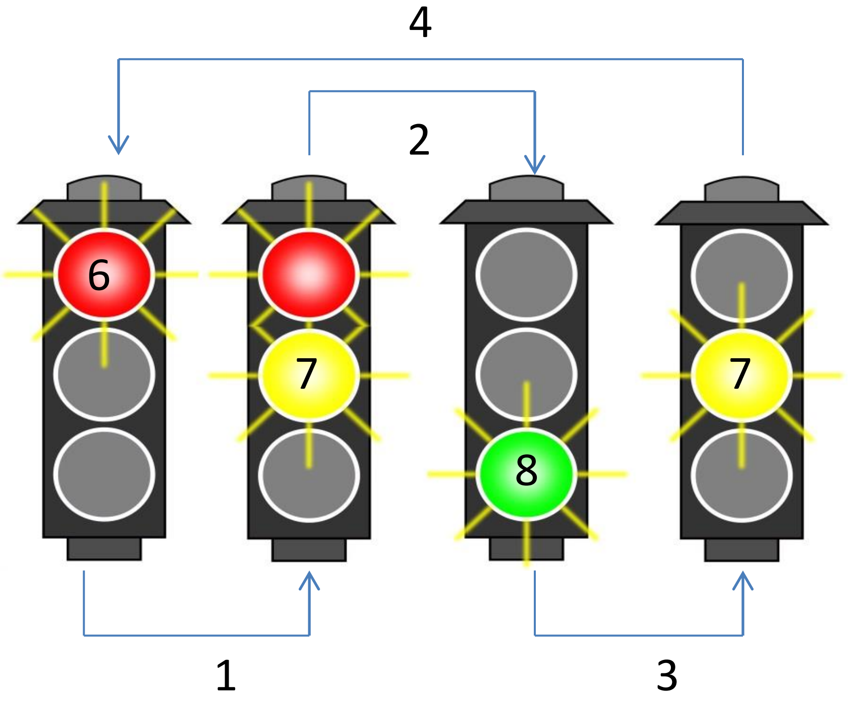
myFlowlab™



Traffic Light Control Project

Guide student to write a program to control a Traffic as per sequence below – The UK Standards

Variation: Get the students to program the Traffic Light based on American Standards

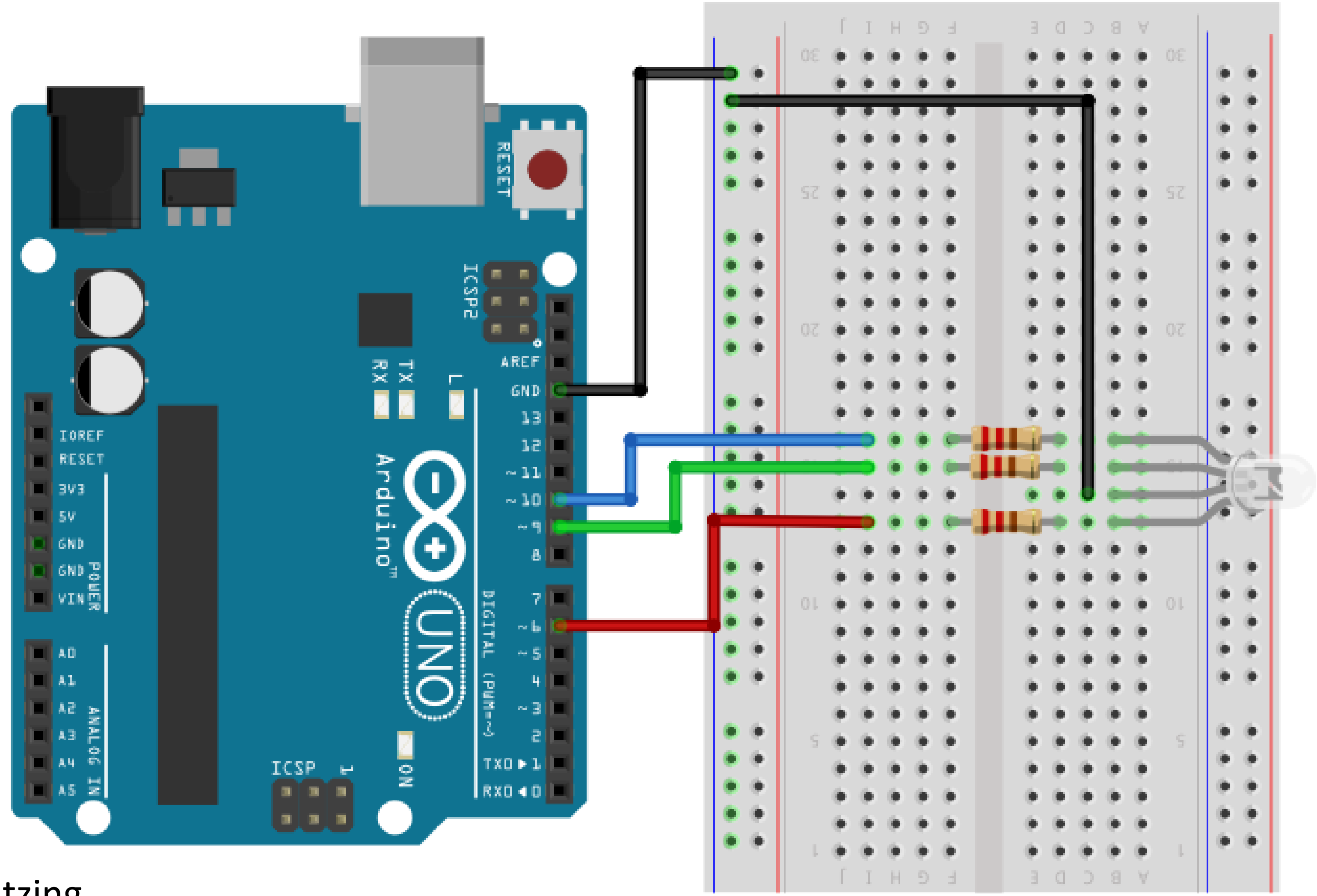


Building Real-World Digital Output project

RG LED Digital Color Mixing Mood Lamp

Activity # 18 - RGB LED Connection

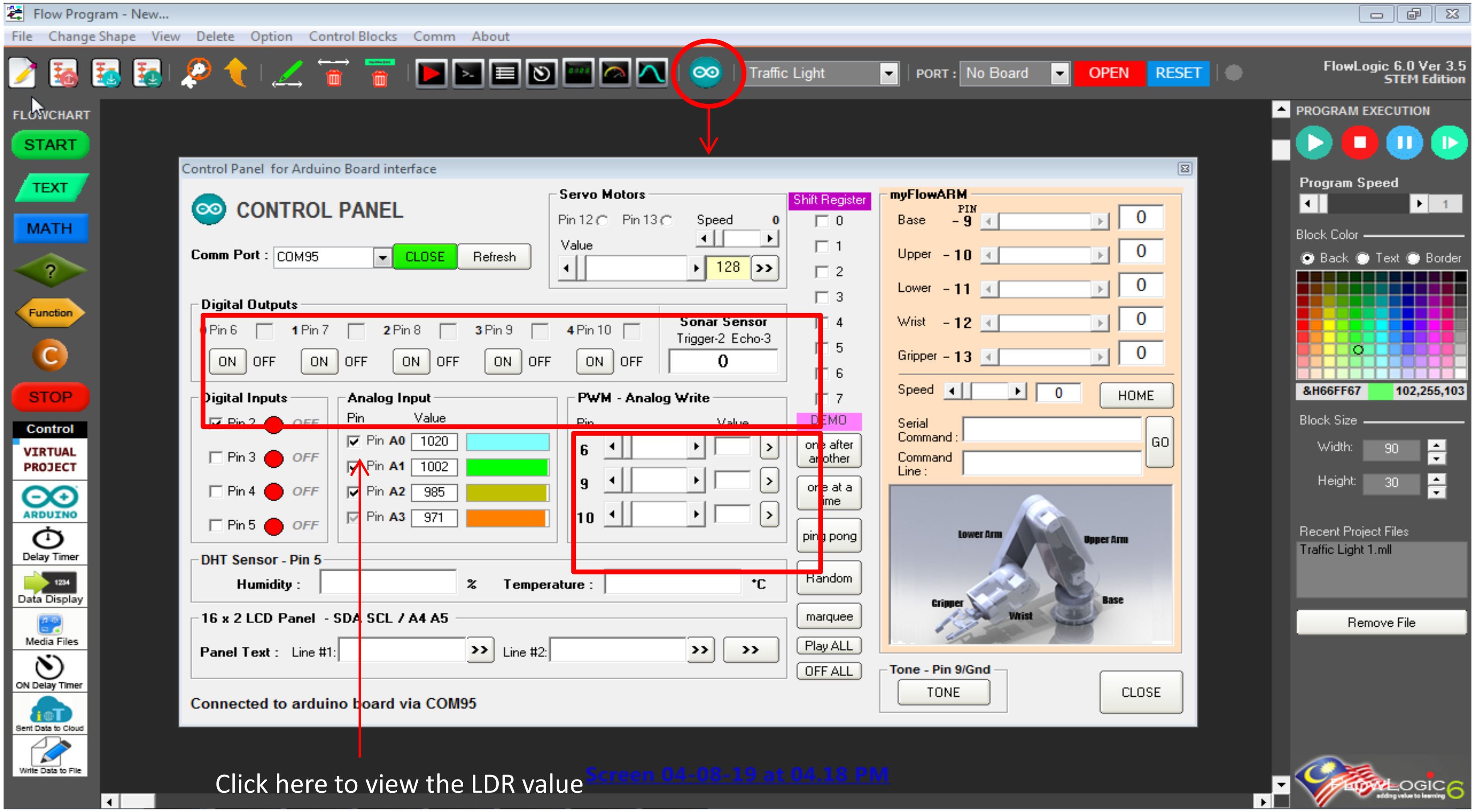
myFlowlab™



Fritzing

myFlowlab™

Activity # 19 -Manual Color Mixing using Control Panel



Traffic Light

PORT : No Board

FlowLogic 6.0 Ver 3.5 STEM Edition

Comm Port : COM95

Speed 0

Value 128

Sonar Sensor 0

☑ Pin A0 1020

☑ Pin A1 1002

☑ Pin A2 985

☑ Pin A3 971

Base 0, Upper 0, Lower 0, Wrist 0, Gripper 0, Speed 0

&H66FF67 102,255,103

Width: 90

Height: 30

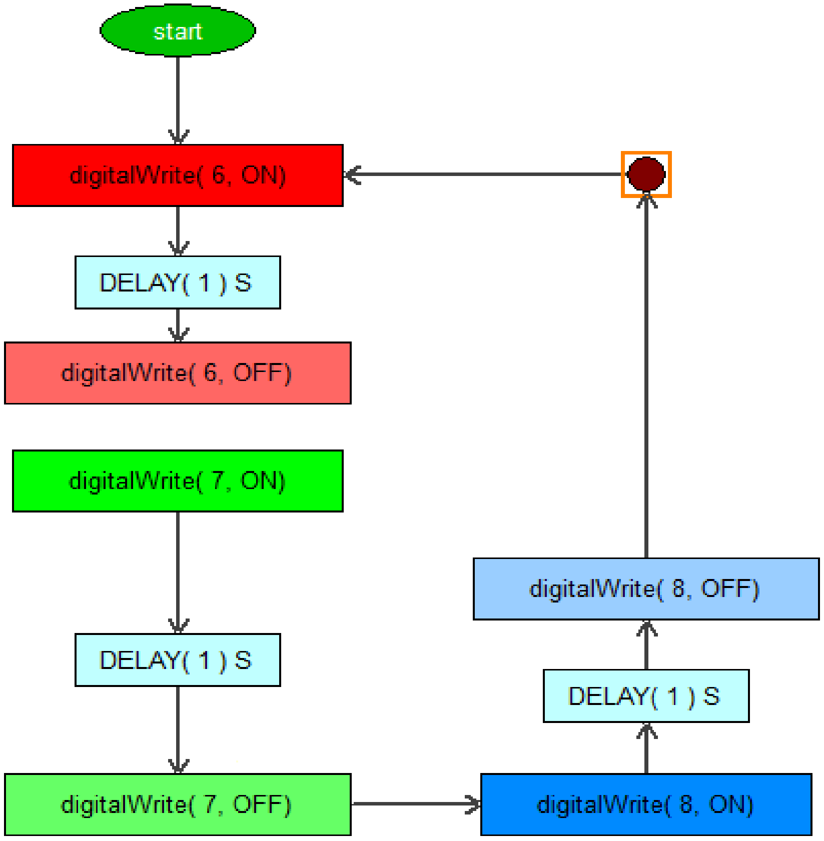
Recent Project Files
Traffic Light 1.mll

Connected to arduino board via COM95

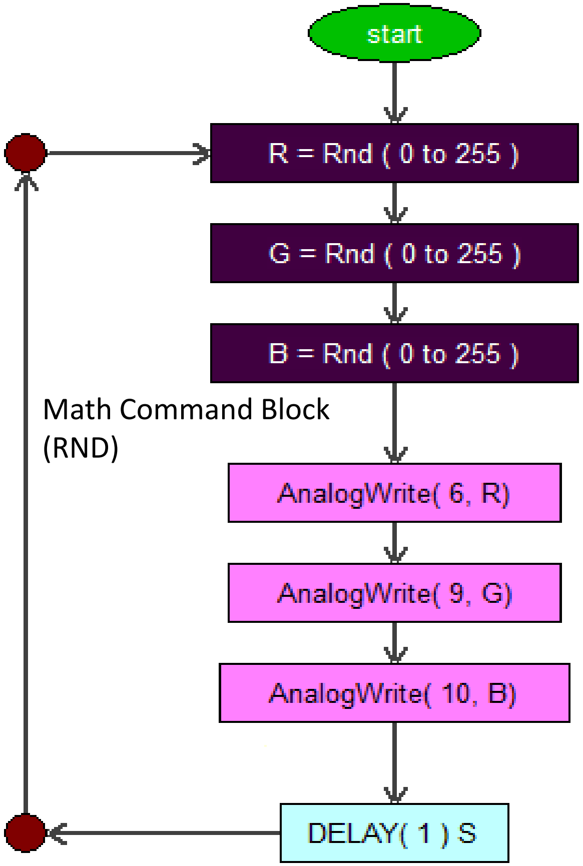
Click here to view the LDR value Screen 04-08-19 at 04.18 PM

Activity : Get the students to explore manually to test their circuit and connection. Guide them to use the DigitalWrite and PWM-Analog Write command to turn ON/OFF the RGB LED and perform Digital Color Mixing thinkering.

Activity # 20 -Algorithm/FlowProgram for Color Mixing using RGB Led – Using Digital , Analog and Math block

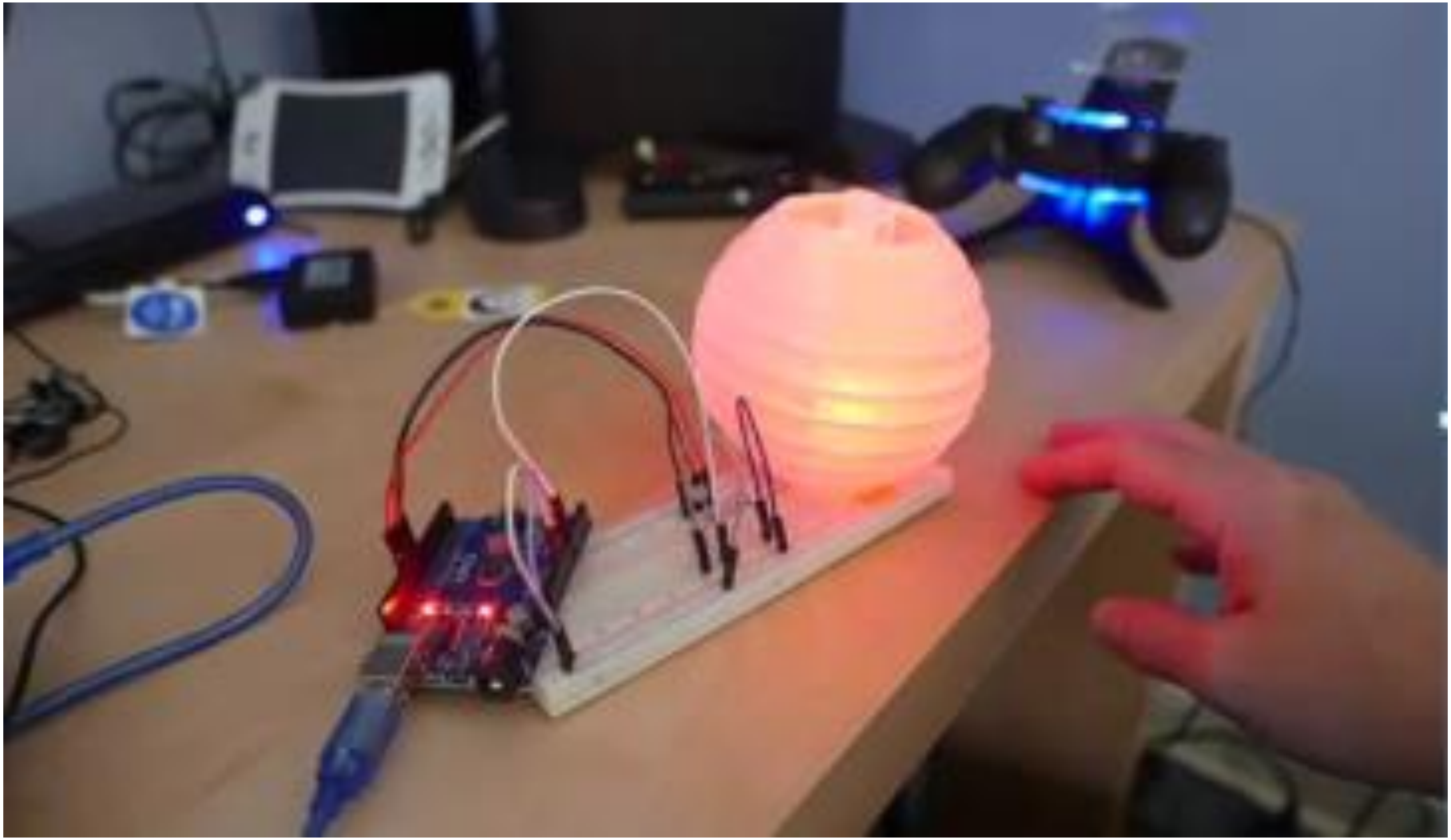


FlowProgram using DigitalWrite () Command Block



FlowProgram using AnalogWrite () Command Block

Mood Lamp – Prototype Model Sample



Final Project

Design a Creative Mood Lamp and
Develop FlowProgram / Algorithm to animate the colors

END OF

Flowchart programming using FlowLogic 6 And Introduction to Arduino UNO

FOUNDATION

LESSON PLAN CONTENTS

Visit www.myflowlab.com to download Beginners Guide
And video tutorials and

M myFlowlab™