



**DesignNews**

Raspberry Pi 4 Automation

# DAY 1 : Physical Computing and Automation Basics

Sponsored by



## Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Group Chat’ by maximizing the chat widget in your dock.
- Submit questions for the lecturer using the Q&A widget. They will follow-up after the lecture portion concludes.



## Don Wilcher

Visit 'Lecturer Profile' in your console for more details.

## Course Kit

**Freenove Ultimate Starter Kit for Raspberry Pi 4 B 3 B+**



## Agenda:

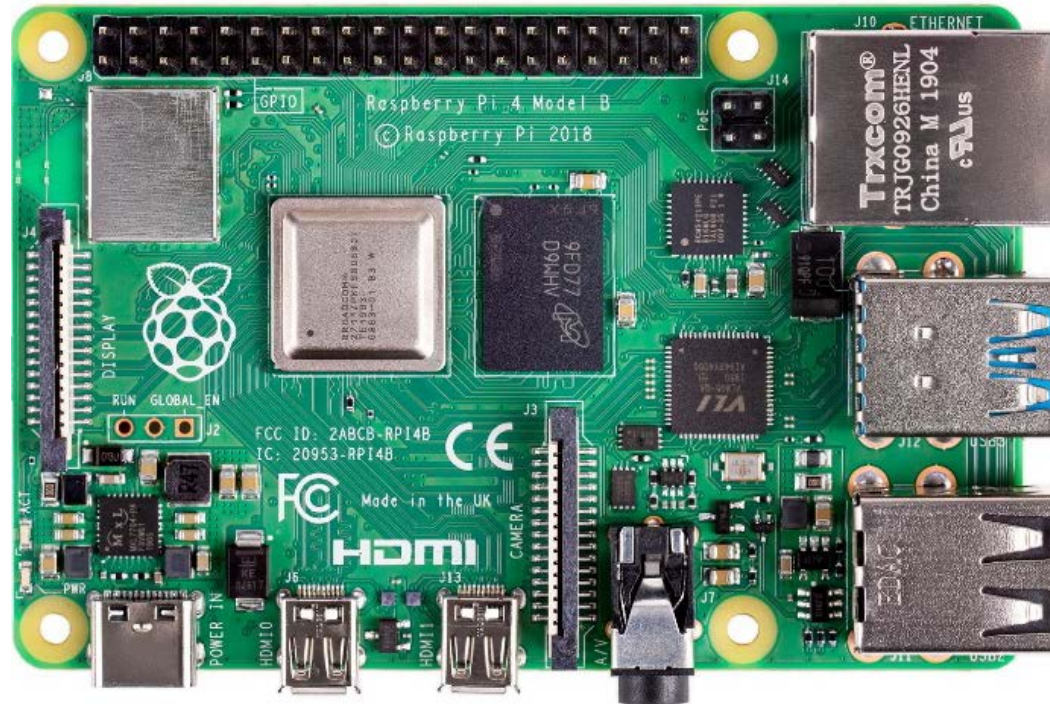
- Introduction to the Raspberry Pi 4 (RPi4)
- What is Physical Computing?
- What is Physical Computing Automation?
- What is Processing?
- Lab: A Processing LED Flasher



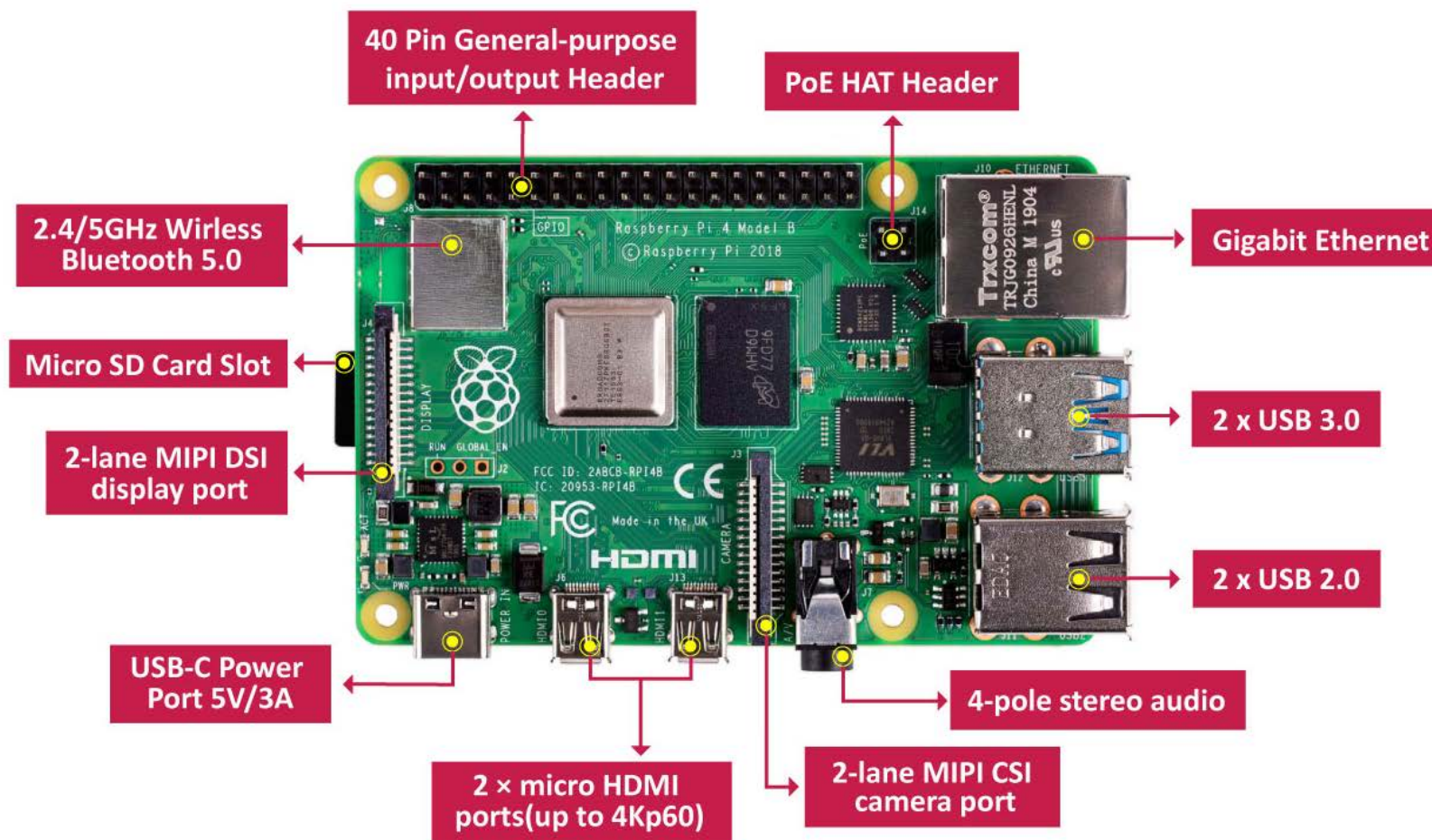




## Introduction to the RPi4...

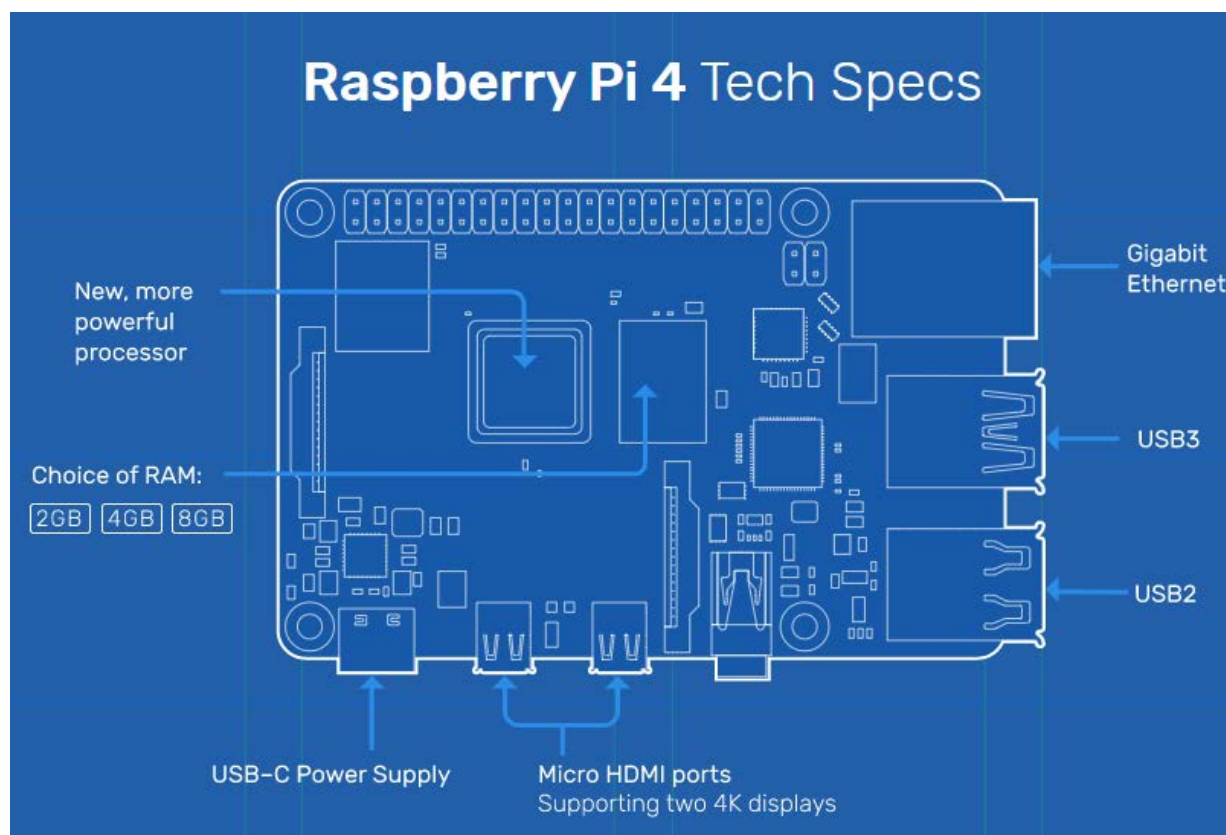


## Introduction to the RPi4...





## Introduction to the RPi4. . .



Source:

<https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>



## Question 1



**What Bluetooth specification is implemented on a RPI4?**

## Introduction to the RPi4. . . Specifications



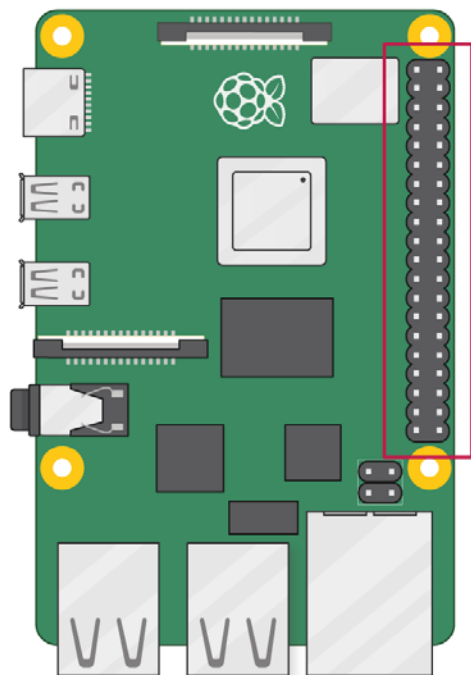
- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- 2GB, 4GB or 8GB LPDDR4-3200 SDRAM (depending on model)
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 USB 3.0 ports; 2 USB 2.0 ports.
- Raspberry Pi standard 40 pin GPIO header (fully backwards compatible with previous boards)
- 2 × micro-HDMI ports (up to 4kp60 supported)
- 2-lane MIPI DSI display port
- 2-lane MIPI CSI camera port
- 4-pole stereo audio and composite video port
- H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode)
- OpenGL ES 3.0 graphics
- Micro-SD card slot for loading operating system and data storage
- 5V DC via USB-C connector (minimum 3A\*)
- 5V DC via GPIO header (minimum 3A\*)
- Power over Ethernet (PoE) enabled (requires separate PoE HAT)
- Operating temperature: 0 – 50 degrees C ambient

\* A good quality 2.5A power supply can be used if downstream USB peripherals consume less than 500mA in total.

Source:

<https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>

## Introduction to the RPi4. . .



3V3 power	1	2	5V power
GPIO 2 (SDA)	3	4	5V power
GPIO 3 (SCL)	5	6	Ground
GPIO 4 (GPCLK0)	7	8	GPIO 14 (TXD)
Ground	9	10	GPIO 15 (RXD)
GPIO 17	11	12	GPIO 18 (PCM_CLK)
GPIO 27	13	14	Ground
GPIO 22	15	16	GPIO 23
3V3 power	17	18	GPIO 24
GPIO 10 (MOSI)	19	20	Ground
GPIO 9 (MISO)	21	22	GPIO 25
GPIO 11 (SCLK)	23	24	GPIO 8 (CE0)
Ground	25	26	GPIO 7 (CE1)
GPIO 0 (ID_SD)	27	28	GPIO 1 (ID_SC)
GPIO 5	29	30	Ground
GPIO 6	31	32	GPIO 12 (PWM0)
GPIO 13 (PWM1)	33	34	Ground
GPIO 19 (PCM_FS)	35	36	GPIO 16
GPIO 26	37	38	GPIO 20 (PCM_DIN)
Ground	39	40	GPIO 21 (PCM_DOUT)

Source:

<https://www.raspberrypi.org/documentation/usage/gpio/>

## Introduction to the RPi4. . . Specifications. . .



### What is MIPI?

- a) MIPI is the Mobile Industry Processor Interface.
- b) The MIPI is a standard that defines industry specifications for the design of mobile devices such as
  - i. smartphones
  - ii. tablets
  - iii. laptop computers
  - iv. hybrid devices

Source:

<https://www.mipi.org/specifications/dsi>



## Introduction to the RPi4. . . Specifications. . .



### What is MIPI?

There are two standard MIPI ports packaged on a RPi4.

- a) Camera Serial Interface (CSI)
  - i. A specification of the MIPI Alliance
  - ii. Defines an interface between a camera and a host processor
- b) Display Serial Interface (DSI)
  - i. A specification of the MIPI Alliance
  - ii. A versatile, high-speed interface for displays in smartphones, tablets, laptop computers, automotive and other platforms.

Source:

<https://www.mipi.org/specifications/dsi>

# Introduction to the RPi4. . . Specifications. . .

## What is MIPI?

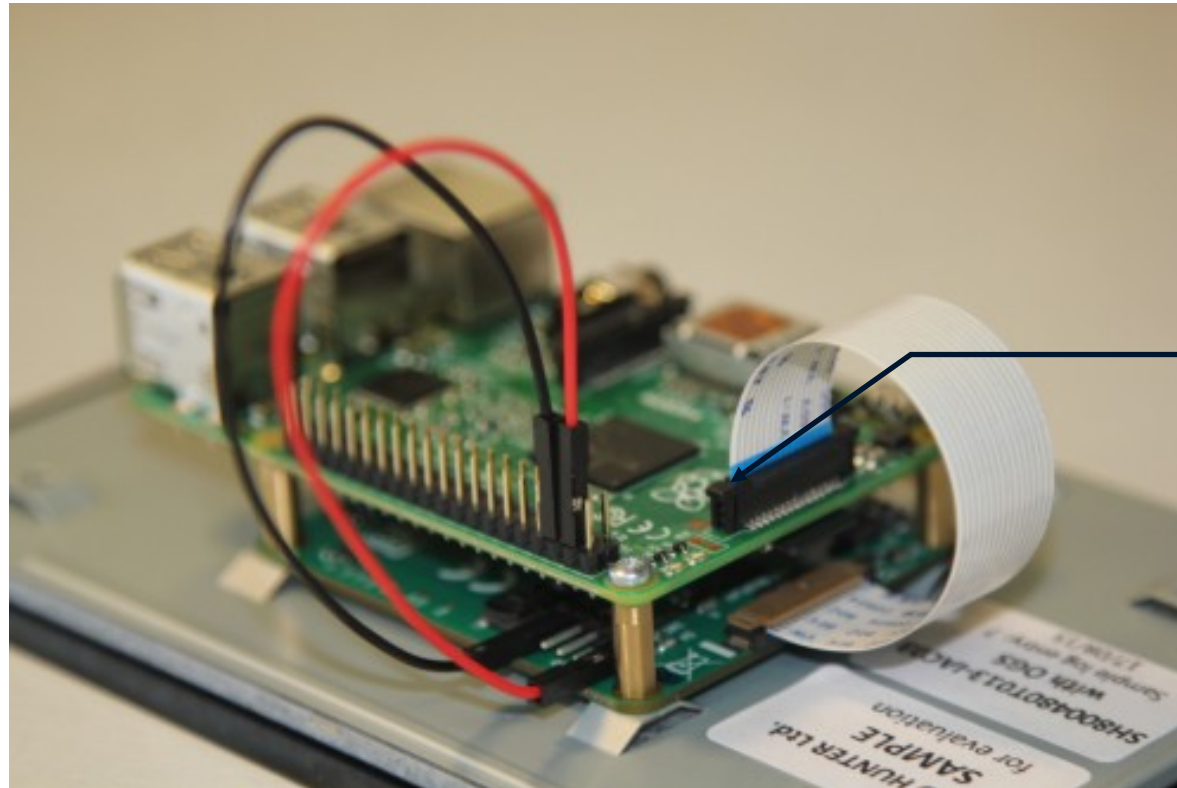


The screenshot shows the MIPI Alliance website. The header includes social media links, Member Login, Contact Us, and Visit our Blog. The main navigation bar has links for About Us, Membership, Specifications, Resources, News, and Events. A search bar and a 'Join MIPI' button are also present. The main content area is titled 'MIPI Display Serial Interface (MIPI DSI)' and describes it as a versatile, high-speed interface for displays in smartphones, tablets, laptops, automotive, and other platforms. It includes sections for Quick Facts, Physical Layer (MIPI D-PHY), Use Cases (UHD resolution, Embedded displays, Smart meters, Video game devices, Smart watches, Virtual or augmented reality Head-mounted devices, In-sight (glass) devices), Industries (represented by icons for mobile, automotive, IoT, and healthcare), Fundamental Features (High performance, Low power, Low EMI), Get the Specification (MIPI DSI<sup>SM</sup> v1.3.1, Member version), Versions (Current Version: v1.3.1 (December 2015)), and Developed by (Display Working Group).

Source:  
<https://www.mipi.org/specifications/dsi>

## Introduction to the RPi4. . . Specifications. . .

### What is MIPI?



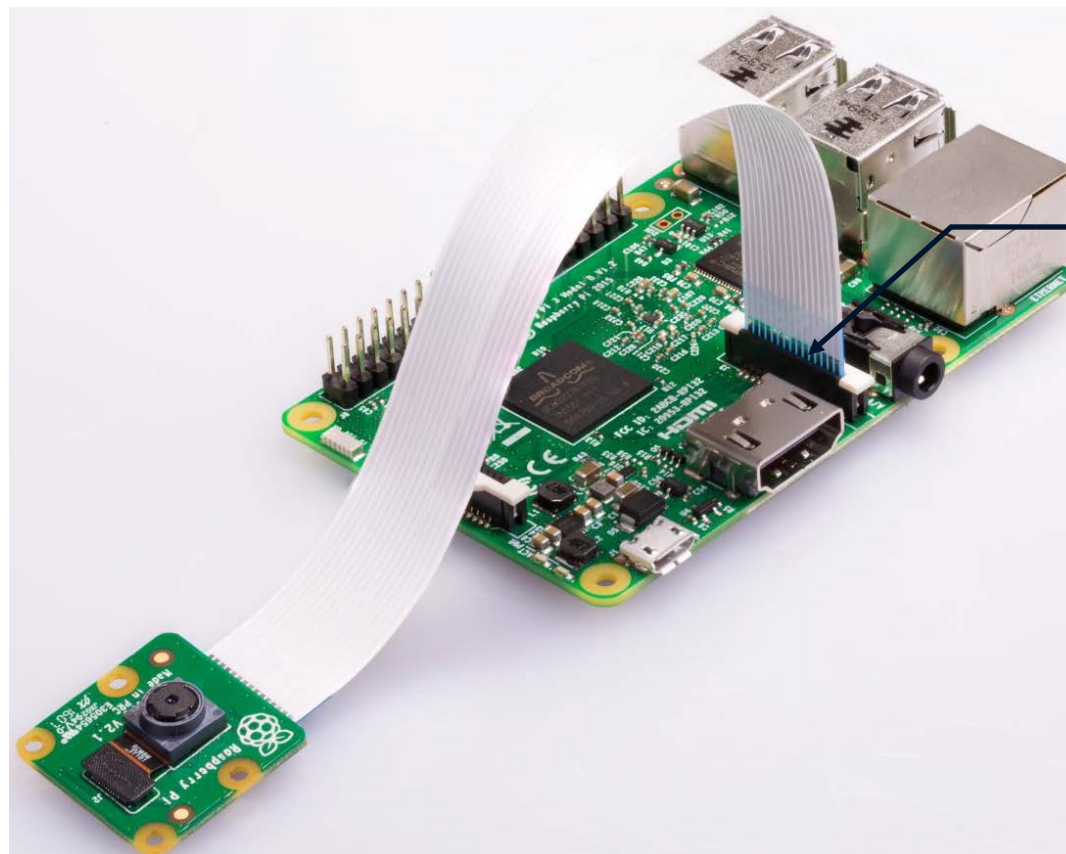
DSI Port

Source:

<https://www.raspberrypi.org/documentation/hardware/display/>

## Introduction to the RPi4. . . Specifications. . .

### What is MIPI?



CSI Port

Source:

<https://projects.raspberrypi.org/en/projects/getting-started-with-picamera>



## Question 2



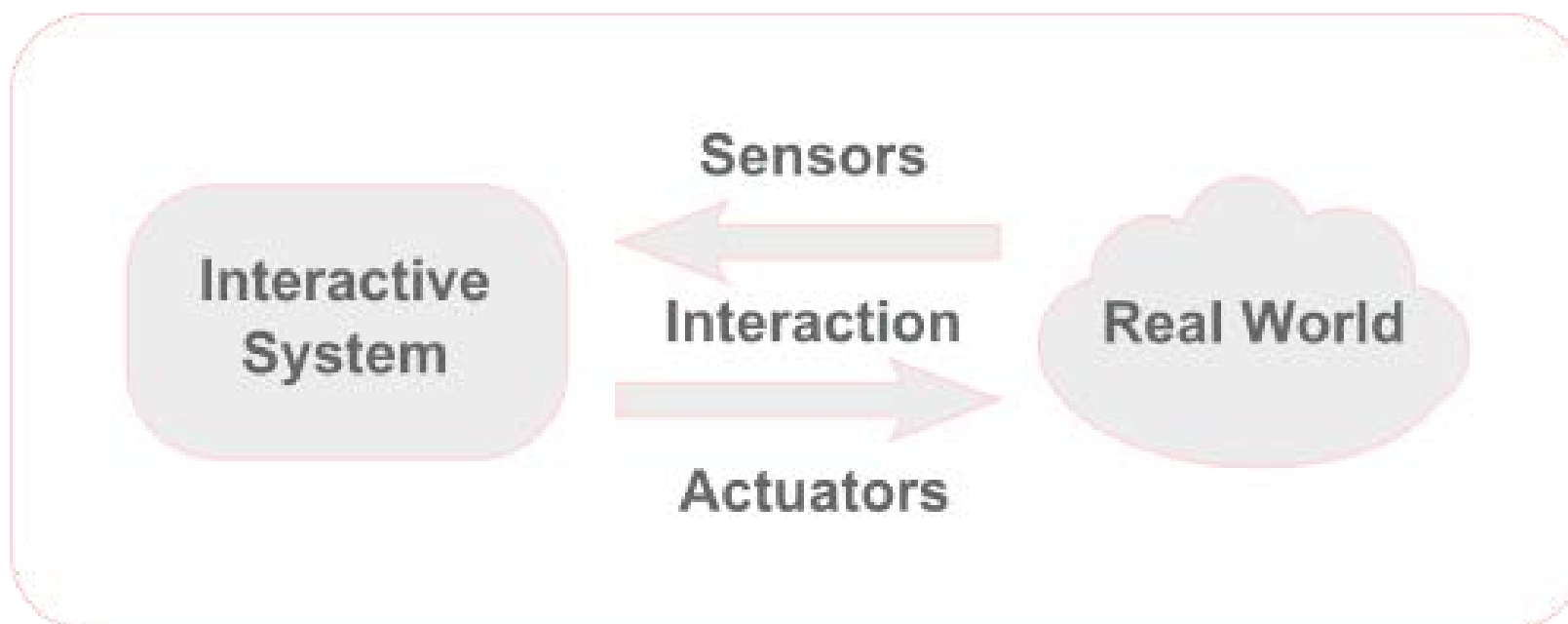
**What is MIPI?**

## What is Physical Computing?...



### Physical Computing

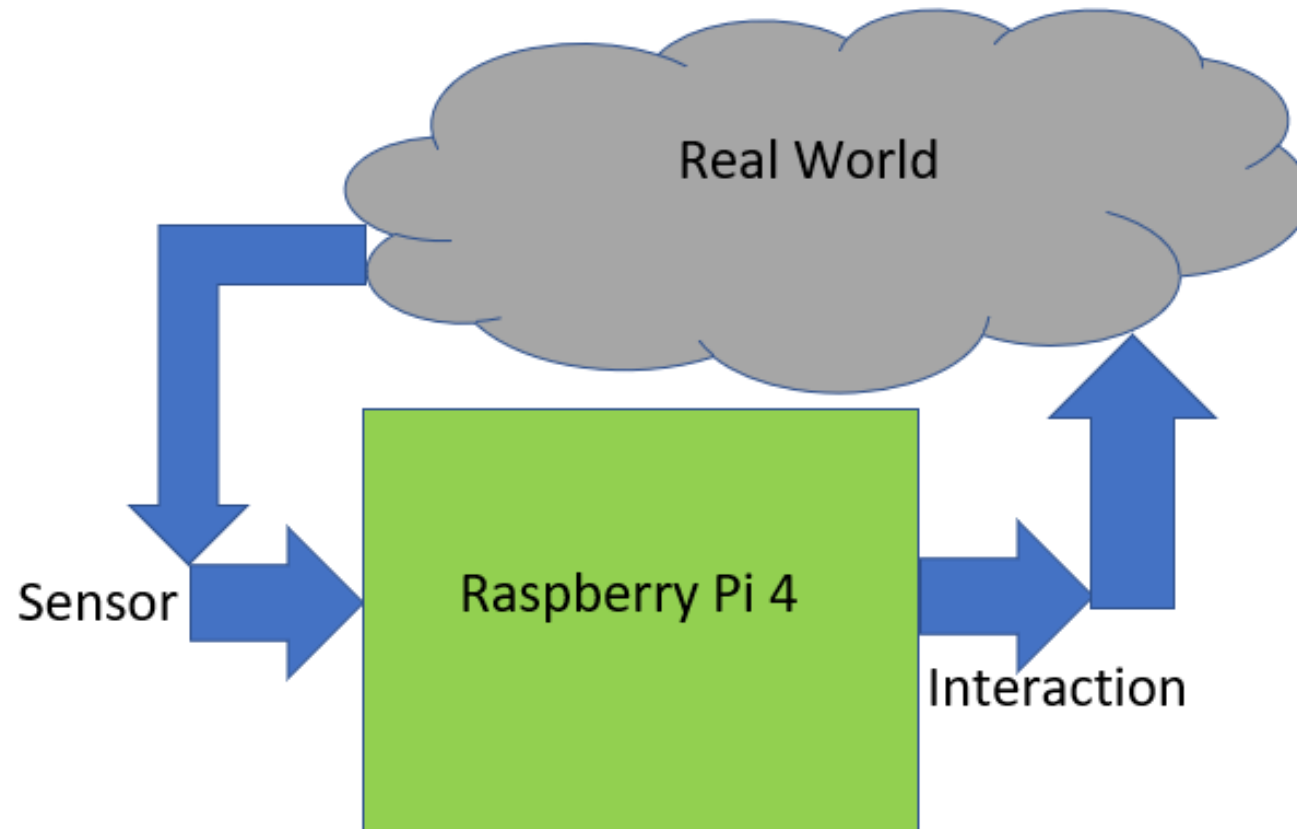
Physical computing, in the broadest sense, means building interactive physical systems by the use of software and hardware that can sense and respond to the analog world





## What is Physical Computing?...

### Concept



## Question 3



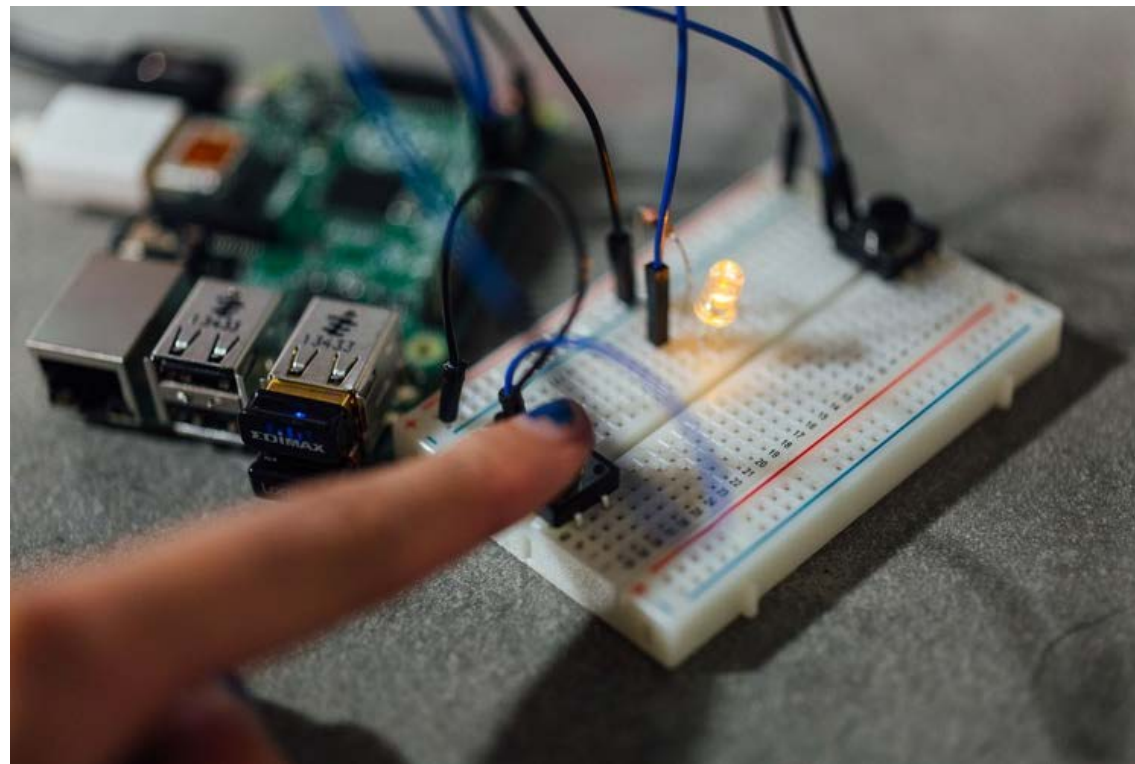
**What world does Physical Computing interact with?**



## What is Physical Computing?



### Concept



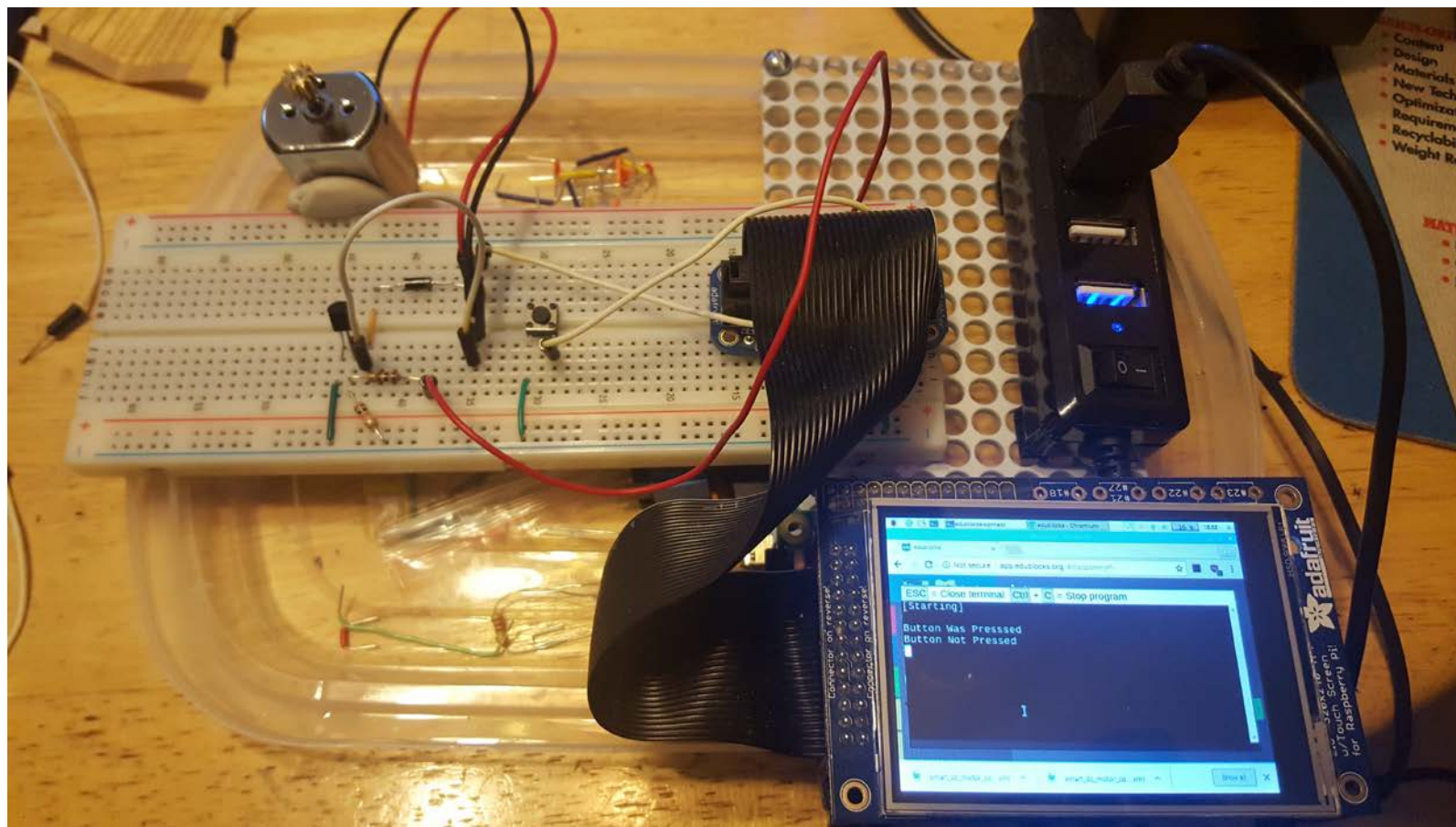
Source:

<https://projects.raspberrypi.org/en/projects/getting-started-with-picamera>

## What is Physical Computing?...

### Concept

**A Basic Physical  
Computing  
Device: DC Motor  
Controller**





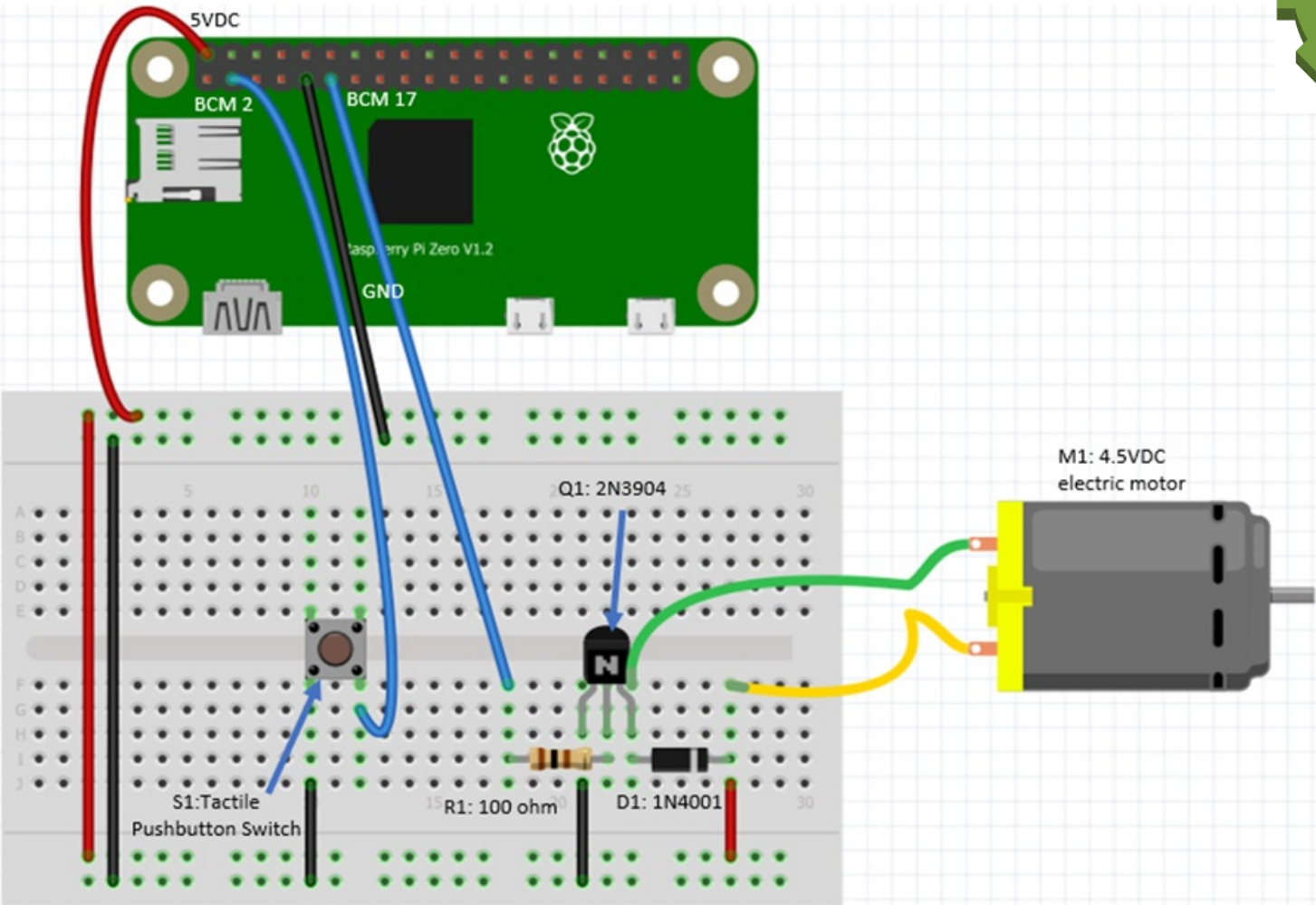


# What is Physical Computing?...

**Concept**

**A Basic Physical Computing Device:  
DC Motor Controller  
Block Diagram**

**Breadboard Wiring  
Diagram**





## What is Physical Computing?...

### Concept

A Basic Physical  
Computing Device:  
DC Motor Controller  
Block Diagram

### Python Code

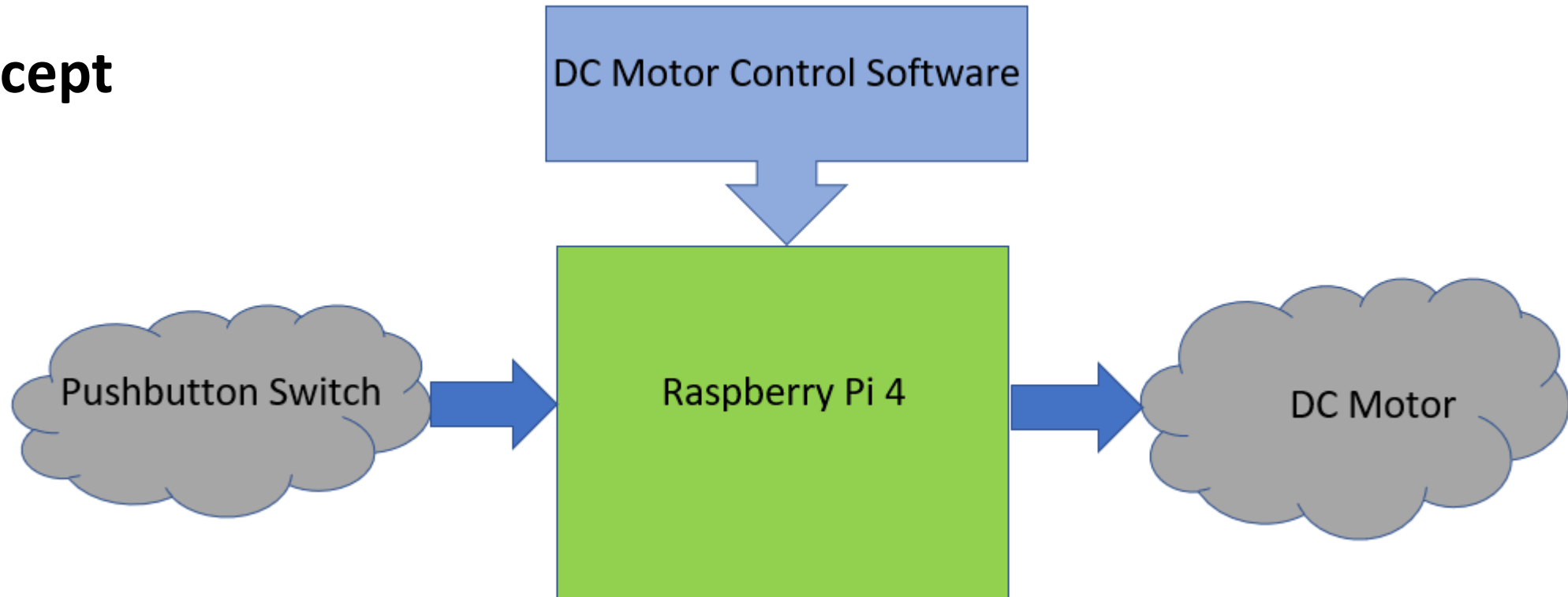
```
1 button = None
2 motor = None
3
4
5 from gpiozero import *
6 import time
7 button = Button(2)
8 motor = Motor(17,18)
9 while True:
10     button.wait_for_press()
11     if True:
12         motor.forward()
13         print("Button Was Presssed")
14         time.sleep(10)
15         print("Button Not Pressed")
16         motor.stop()
```



## What is Physical Computing?...



### Concept



**A Basic Physical Computing Device:  
DC Motor Controller Block Diagram**

## What is Physical Computing Automation?



**Automation** – use of control systems and information technologies to reduce the need for human work in the production of goods and services. In the scope of industrialization, automation is a step beyond mechanization [1].

### Physical Computing Automation

Creating control systems that use sensors and information technologies that augment the human worker (aka production operator).

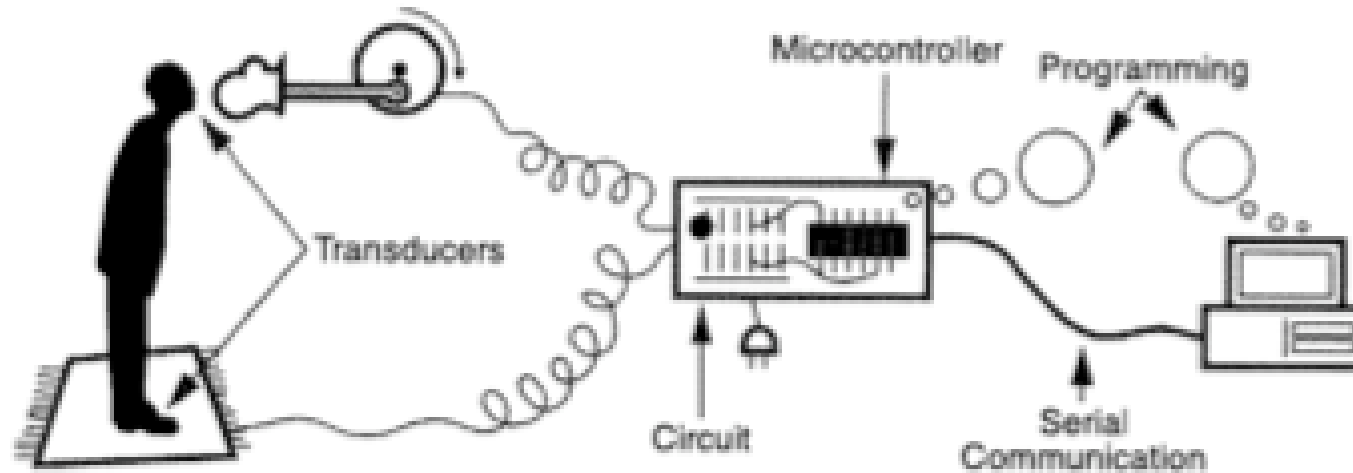
Source:

<https://en.wikipedia.org/wiki/Automation>

# What is Physical Computing Automation?



**Figure 1.4**  
The parts of a physical computing system.



## Physical Computing Concept

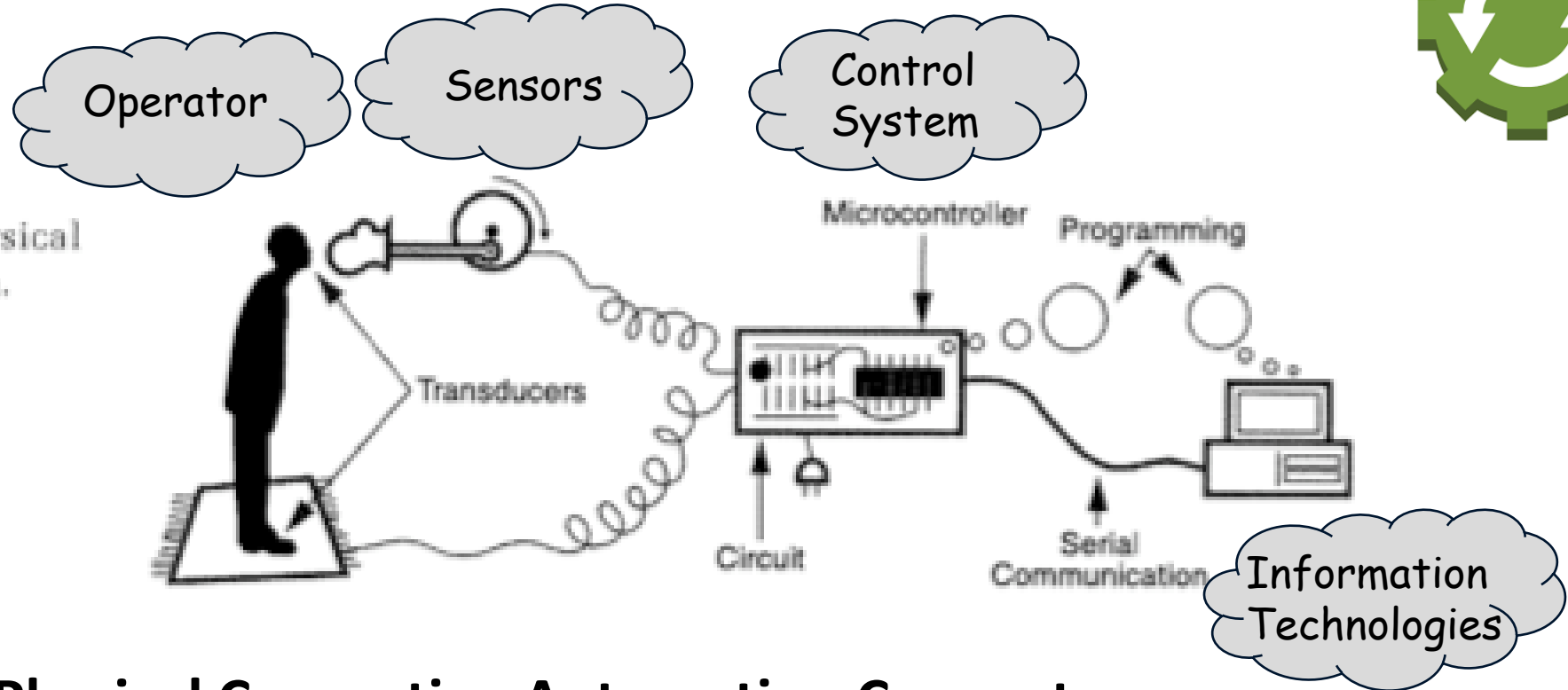
Source:

O'Sullivan, D., & Igoe, T. (2004). *Physical computing: Sensing and controlling the physical world with computers*. Boston, MA: Thompson.

## What is Physical Computing Automation?



**Figure I.4**  
The parts of a physical computing system.



## Physical Computing Automation Concept

Source:

O'Sullivan, D., & Igoe, T. (2004). *Physical computing: Sensing and controlling the physical world with computers*. Boston, MA: Thompson.

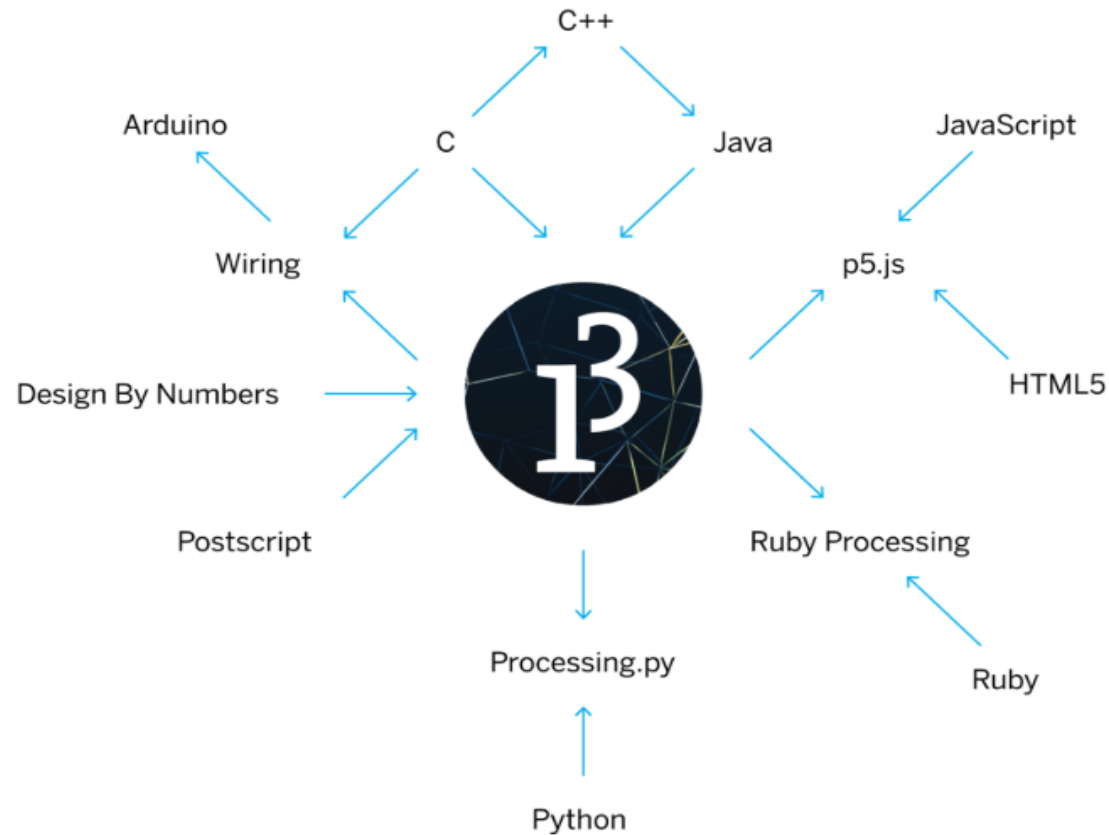


# What is Processing?

A programming language created to make programming interactive graphics easier.



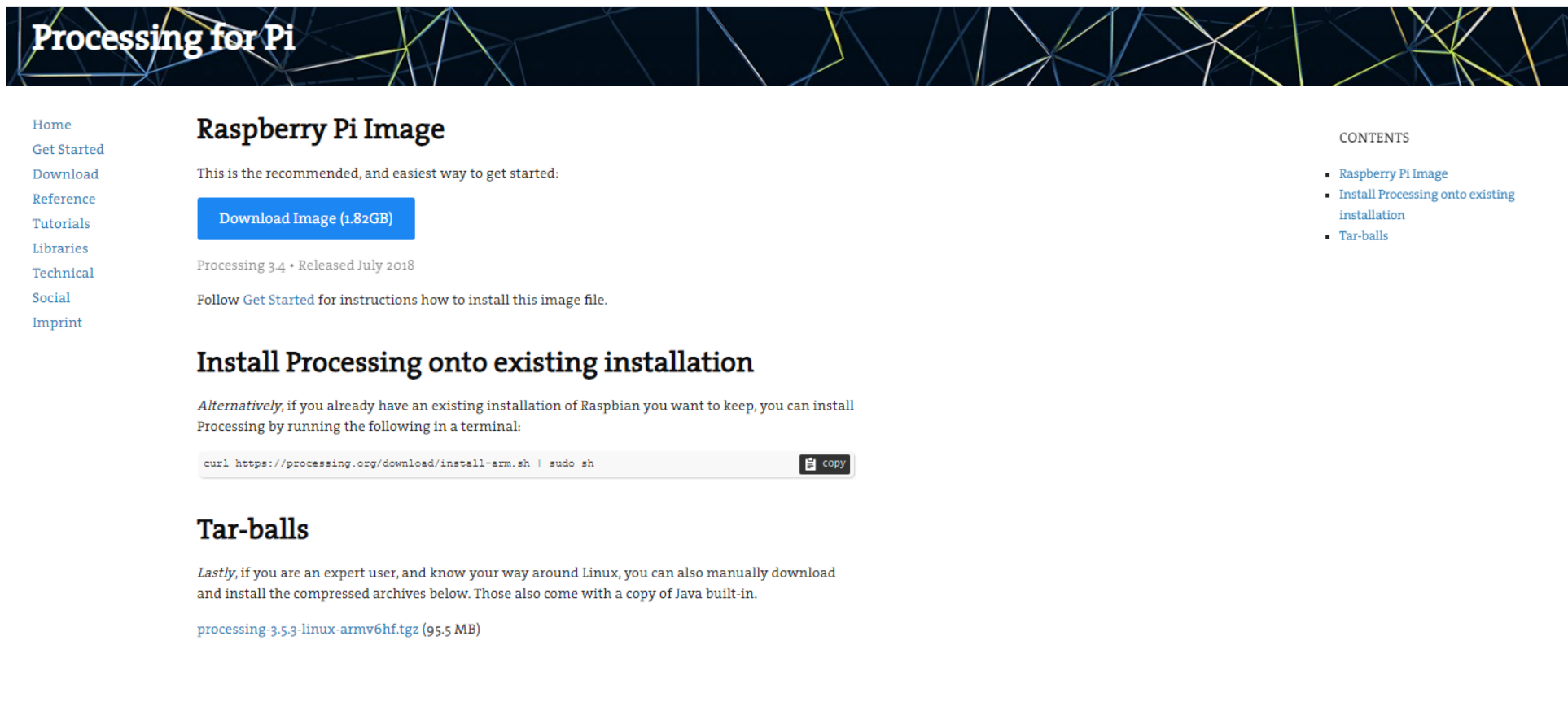
**Processing is comprised of multiple computer languages**



Source:

Reas, C., & Fry, B. (2015). *Getting started with processing* (2nd ed). Make: Community.

# What is Processing?...



The screenshot shows the 'Processing for Pi' website. The header has a dark blue background with a geometric pattern and the text 'Processing for Pi'. On the left is a navigation menu with links: Home, Get Started, Download, Reference, Tutorials, Libraries, Technical, Social, and Imprint. The main content area has a section titled 'Raspberry Pi Image' with the text 'This is the recommended, and easiest way to get started:' followed by a blue button 'Download Image (1.82GB)'. Below this, it says 'Processing 3.4 • Released July 2018' and 'Follow Get Started for instructions how to install this image file.' The next section is 'Install Processing onto existing installation' with the text 'Alternatively, if you already have an existing installation of Raspbian you want to keep, you can install Processing by running the following in a terminal:' and a code block containing 'curl https://processing.org/download/install-arm.sh | sudo sh' with a 'copy' button. The final section is 'Tar-balls' with the text 'Lastly, if you are an expert user, and know your way around Linux, you can also manually download and install the compressed archives below. Those also come with a copy of Java built-in.' and a link 'processing-3.5.3-linux-armv6hf.tgz (95.5 MB)'. On the right side, there is a 'CONTENTS' section with a list of links: 'Raspberry Pi Image', 'Install Processing onto existing installation', and 'Tar-balls'.

## Processing for Pi

- Home
- Get Started
- Download
- Reference
- Tutorials
- Libraries
- Technical
- Social
- Imprint

### Raspberry Pi Image

This is the recommended, and easiest way to get started:

[Download Image \(1.82GB\)](#)

Processing 3.4 • Released July 2018

Follow [Get Started](#) for instructions how to install this image file.

### Install Processing onto existing installation

Alternatively, if you already have an existing installation of Raspbian you want to keep, you can install Processing by running the following in a terminal:

```
curl https://processing.org/download/install-arm.sh | sudo sh
```

[copy](#)

### Tar-balls

Lastly, if you are an expert user, and know your way around Linux, you can also manually download and install the compressed archives below. Those also come with a copy of Java built-in.

[processing-3.5.3-linux-armv6hf.tgz](#) (95.5 MB)

#### CONTENTS

- [Raspberry Pi Image](#)
- [Install Processing onto existing installation](#)
- [Tar-balls](#)

## Downloading Processing

Source:  
<https://pi.processing.org/download/>

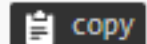
## What is Processing?...



### Install Processing onto existing installation

*Alternatively, if you already have an existing installation of Raspbian you want to keep, you can install Processing by running the following in a terminal:*

```
curl https://processing.org/download/install-arm.sh | sudo sh
```



copy

Source:  
<https://pi.processing.org/download/>

## Question 4



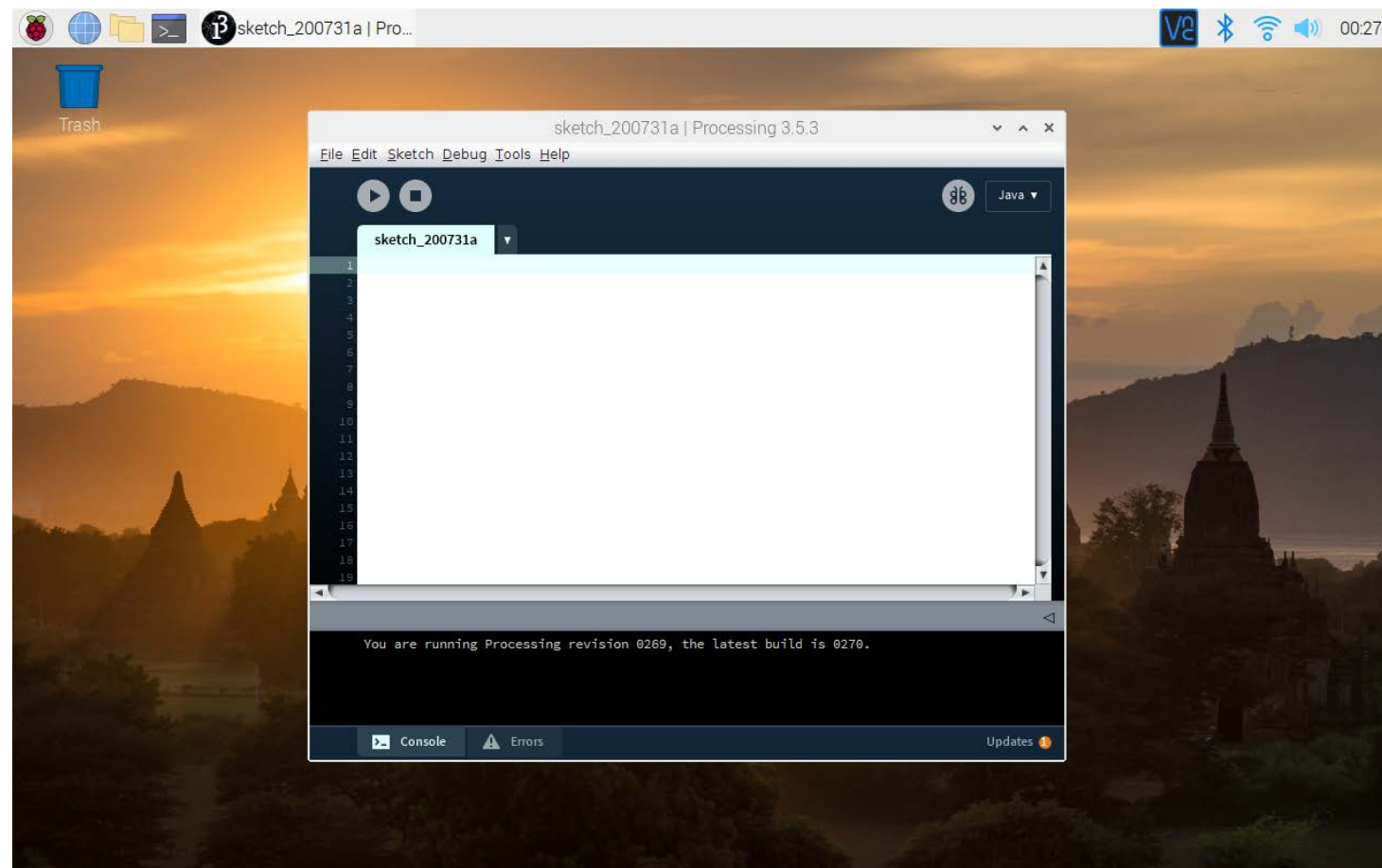
**What is the design concept behind the Processing Language?**



## What is Processing?...



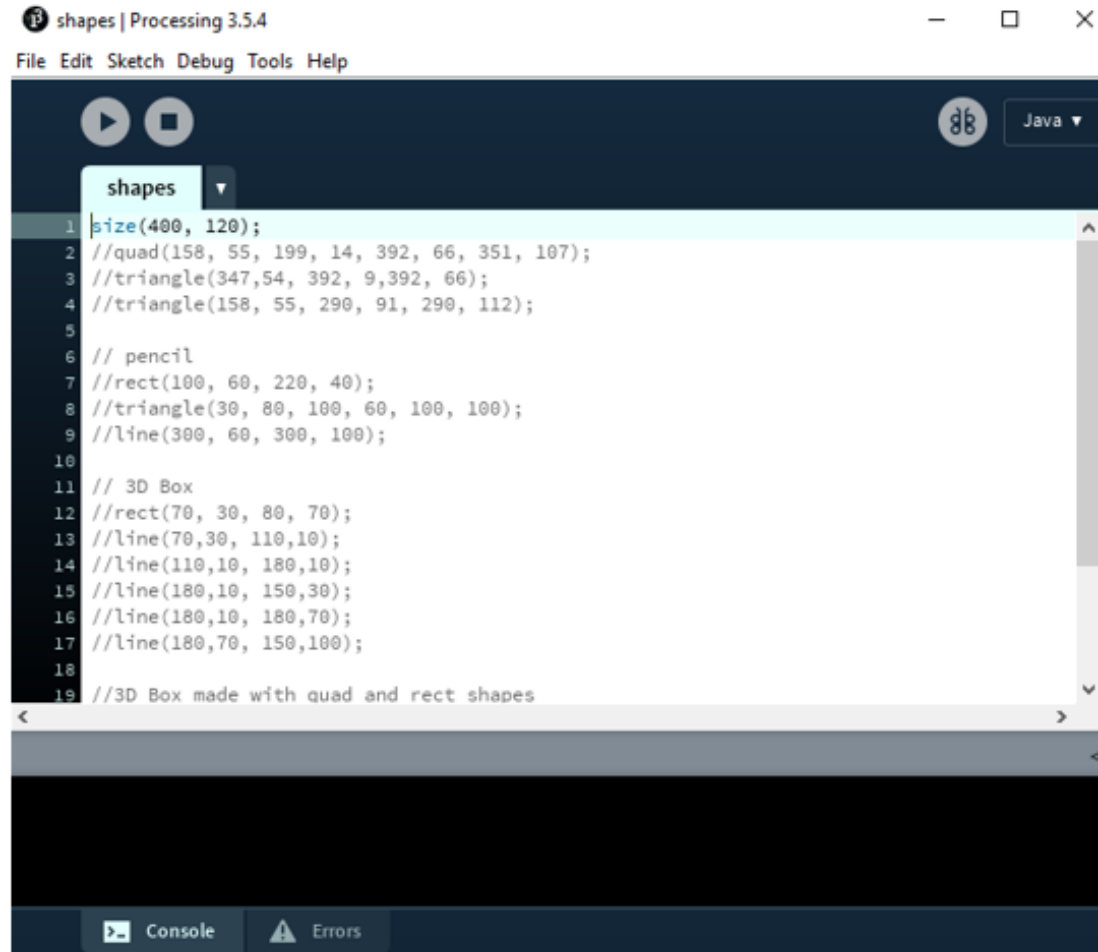
**Processing installed  
successfully on a  
RPi4 !**



# What is Processing?...



Processing Integrated Development Environment (IDE)



Toolbar  
Tabs

Text Editor

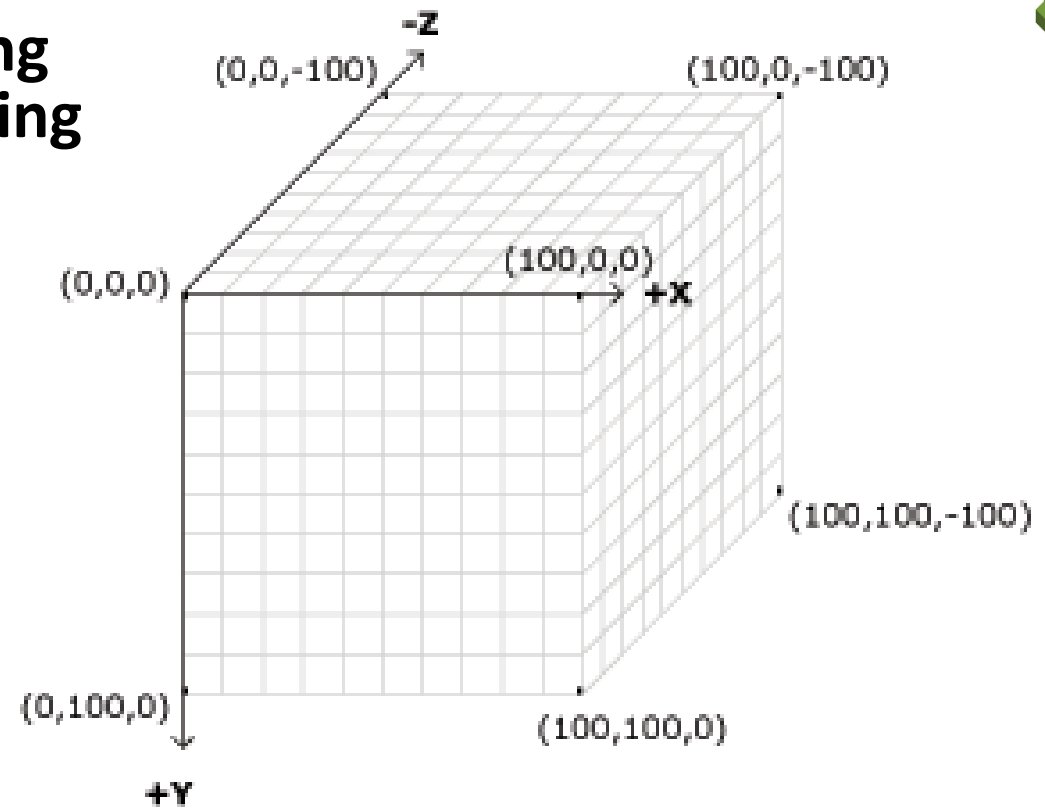
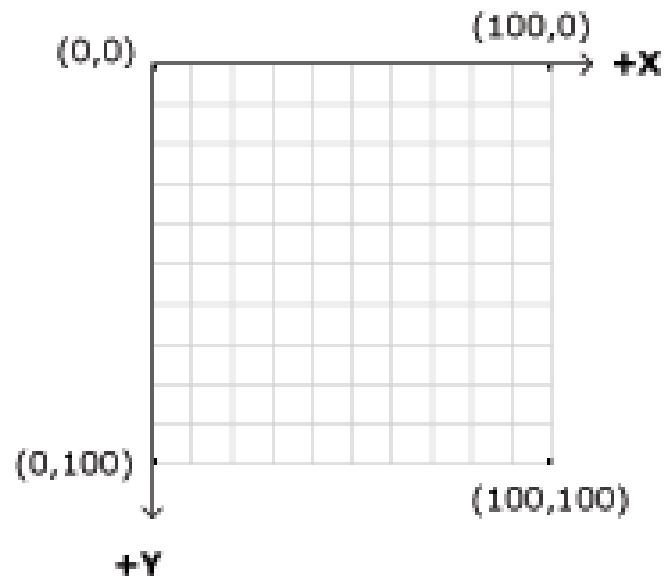
Message Area

Console/Errors

## What is Processing?...



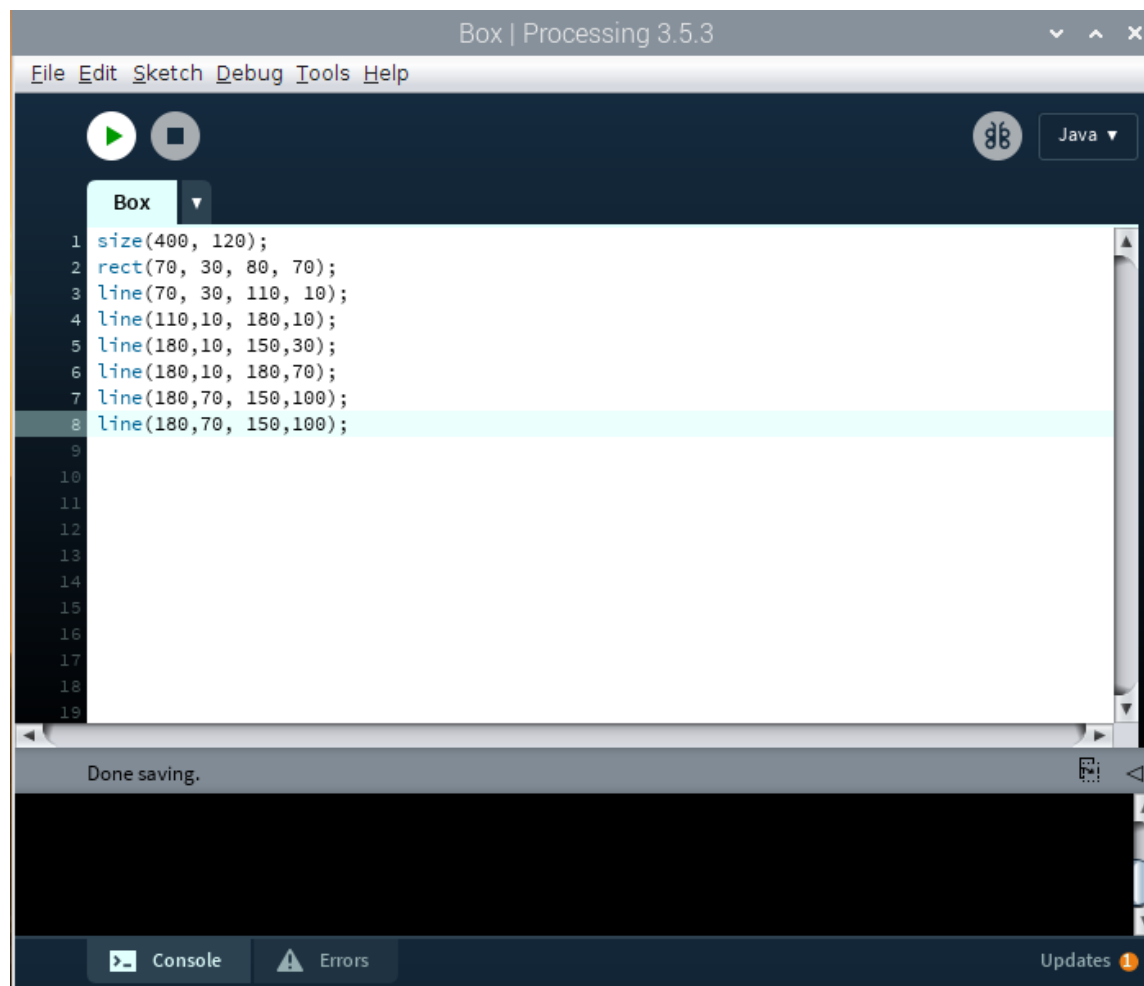
### Processing Coordinating System



# What is Processing?...



How to use the  
Coordinate System  
to create Shapes in  
Processing







## What is Processing?...

Processing  
Code for  
creating a Box

```
Box ▼  
1 size(400, 120);  
2 rect(70, 30, 80, 70);  
3 line(70, 30, 110, 10);  
4 line(110, 10, 180, 10);  
5 line(180, 10, 150, 30);  
6 line(180, 10, 180, 70);  
7 line(180, 70, 150, 100);  
8 line(180, 70, 150, 100);
```

## What is Processing?...

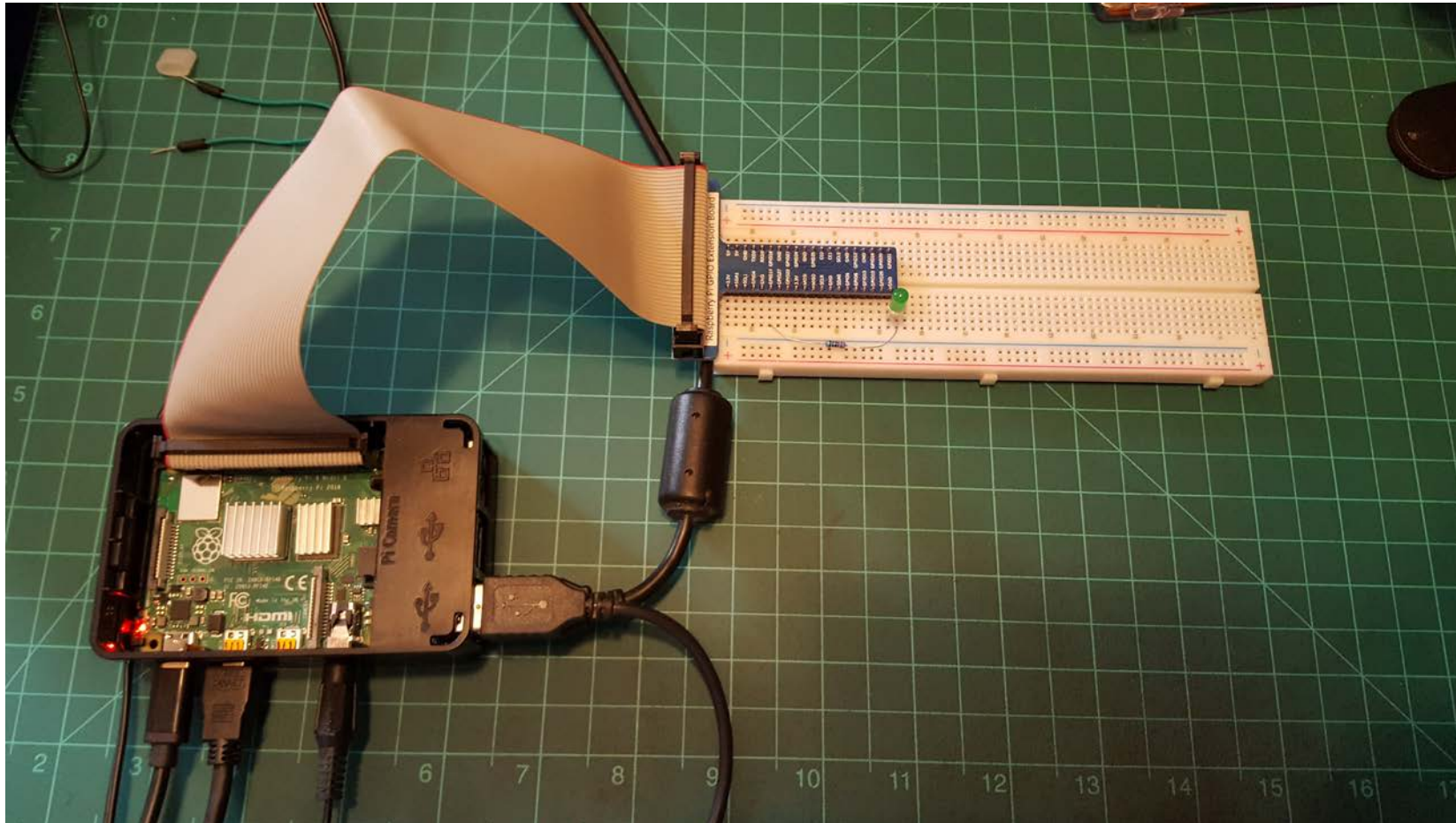
### Processor Coordinate System Example

```
(x , y, width, height);  
rect(70, 30, 80, 70);
```

```
(x1 , y1, x2, y2);  
line(70, 30, 110, 10);
```



## Lab: A Processing LED Flasher



## Big IDEAS:

### Lab: A Processing LED Flasher...



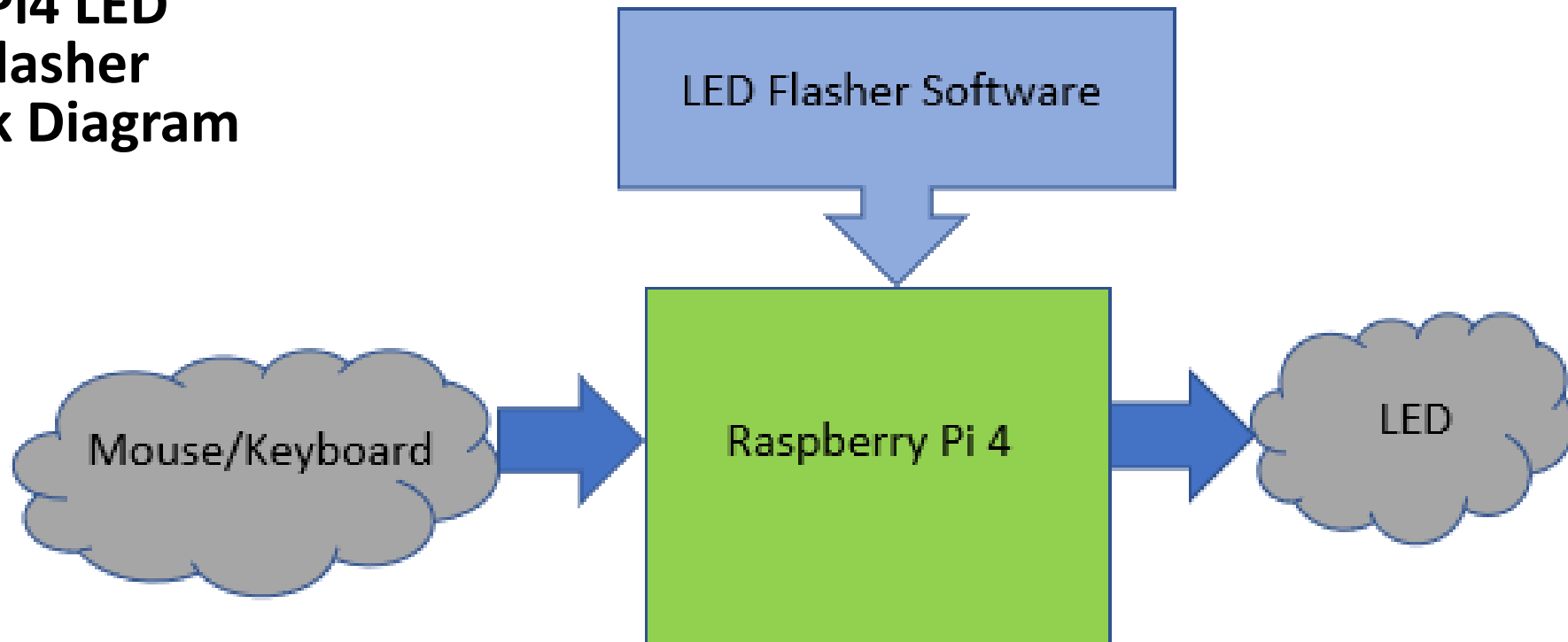
1. Learners will be able to explain to Physical Computing and Automation Concepts.
2. Learners will be able to explain Constructs.
3. Learners will be able to install the Processing programming language.
4. Learners will be able to upload and run a basic shape processing code on a Raspberry Pi 4 computer.
5. Learners will be able to upload and run a digital LED flasher processing code on a Raspberry Pi computer.



## Lab: A Processing LED Flasher...



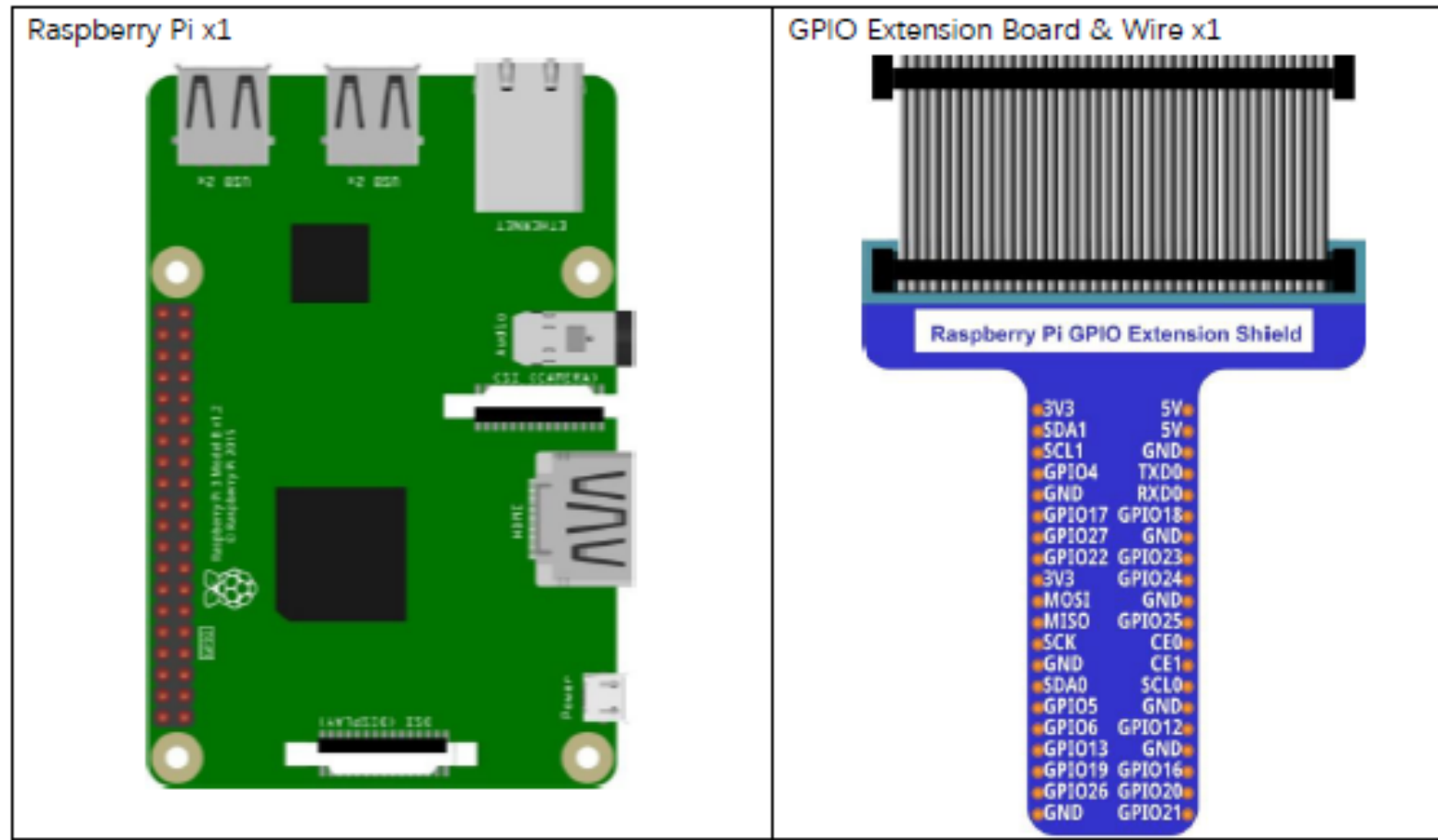
### RPi4 LED Flasher Block Diagram



## Lab: A Processing LED Flasher...



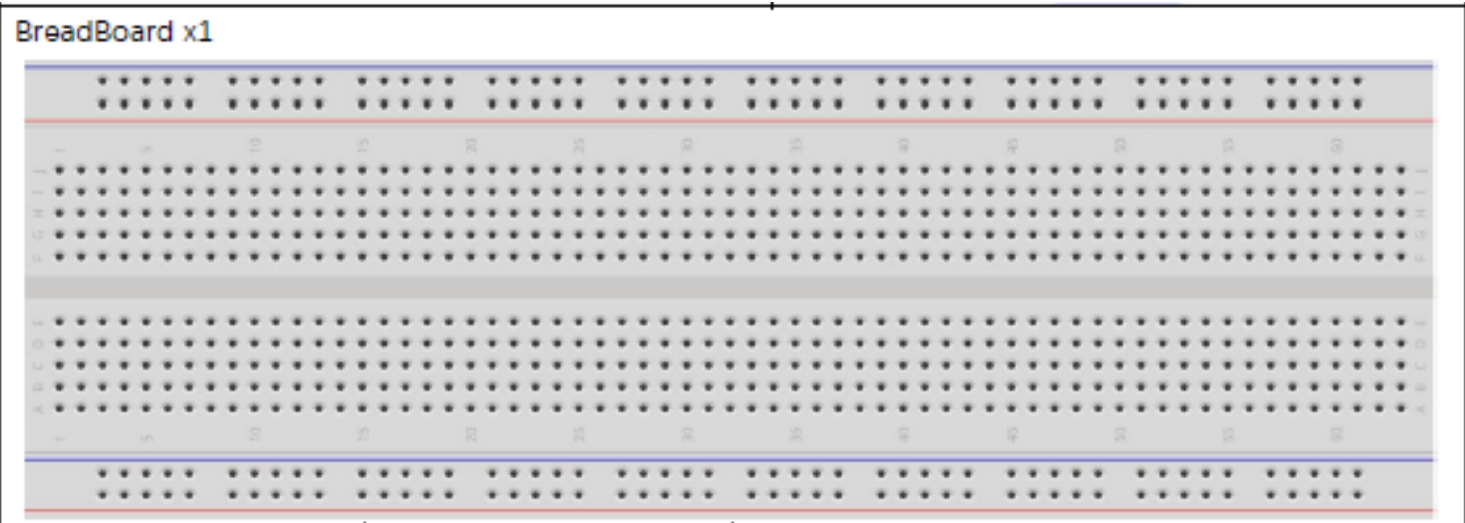



### RPi4 LED Flasher Components



# Lab: A Processing LED Flasher...



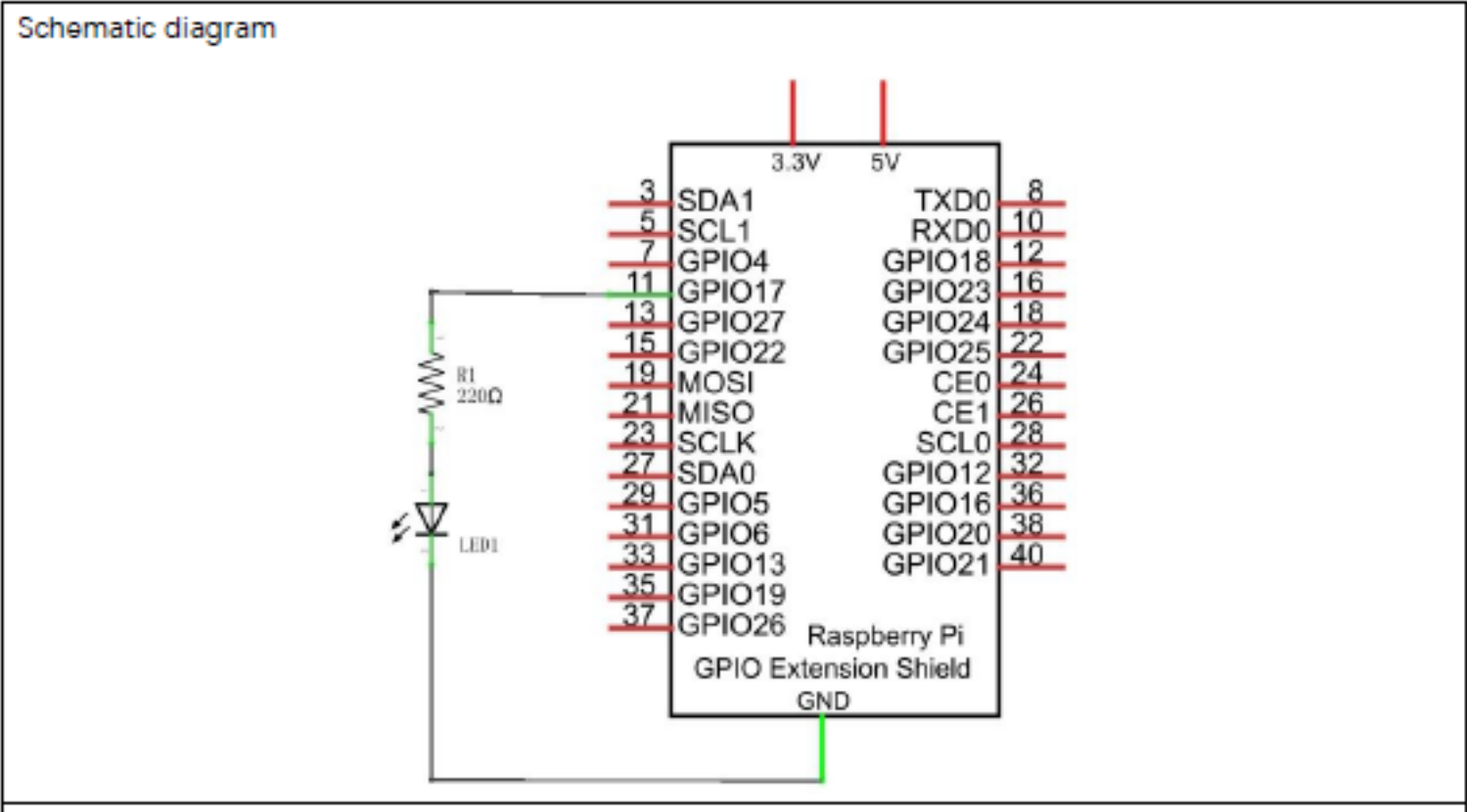
## RPi4 LED Flasher Components

		
<p>LED x1</p> 	<p>Resistor 220Ω x1</p> 	<p>Jumper Wire M/M x2</p> 

# Lab: A Processing LED Flasher...



**RPi4 LED Flasher:  
Circuit Schematic  
Diagram**

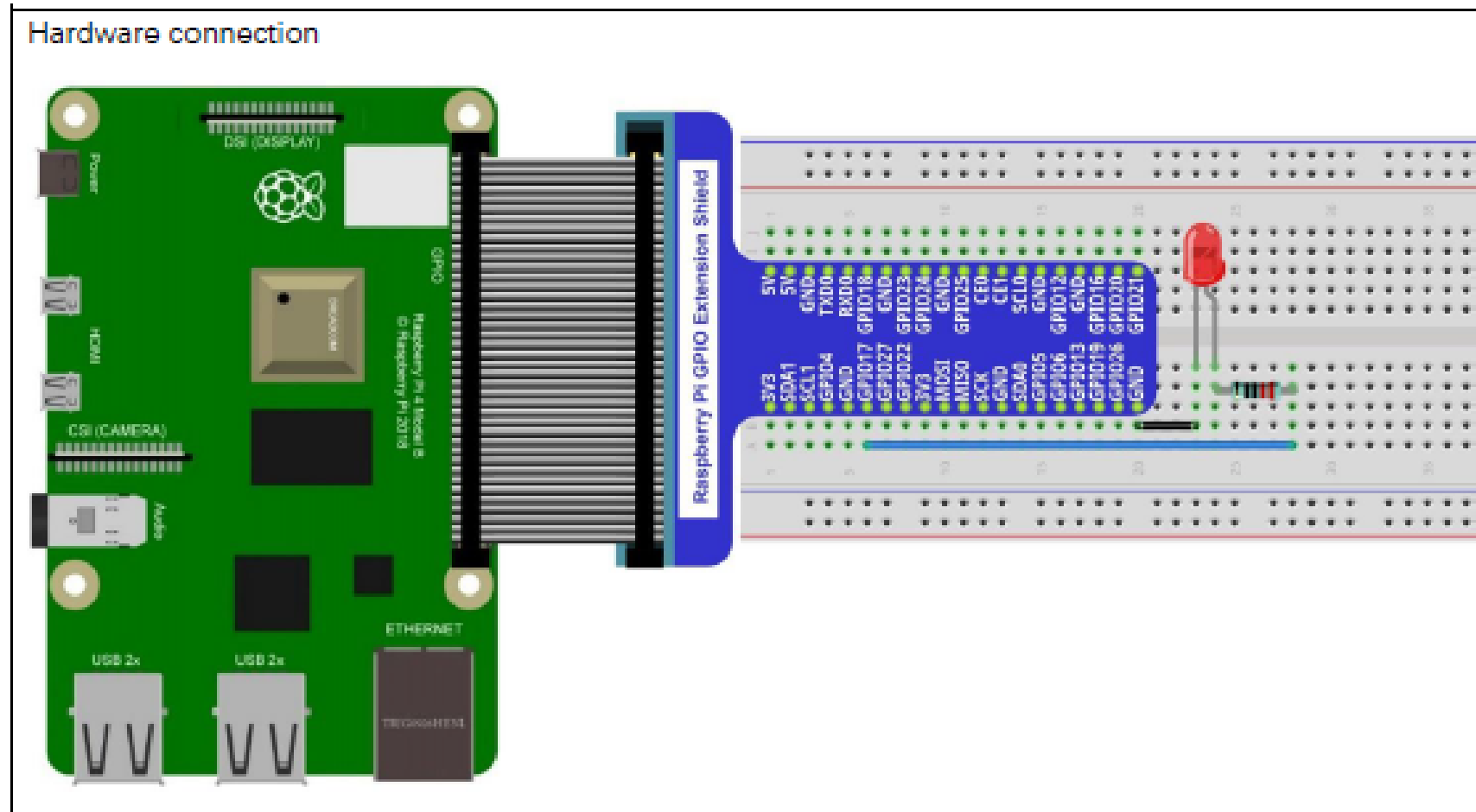




## Lab: A Processing LED Flasher...



### RPi4 LED Flasher: Breadboard Wiring Diagram



## Lab: A Processing LED Flasher...



### RPi4 LED Flasher: Processing Code Breakdown

```
import processing.io.*;
```

I/O Library for Raspberry Pi

```
int ledPin = 17;    //define ledPin  
boolean ledState = false;    //define ledState
```

```
void setup() {  
  size(100, 100);  
  frameRate(1);    //set frame rate  
  GPIO.pinMode(ledPin, GPIO.OUTPUT);    //set the ledPin to output mode  
}
```

Switching (in secs) between  
backgrounds and flashing external LED

```
void draw() {  
  ledState = !ledState;  
  if (ledState) {  
    GPIO.digitalWrite(ledPin, GPIO.HIGH);    //led on  
    background(255, 0, 0); //set the fill color of led on  
  } else {  
    GPIO.digitalWrite(ledPin, GPIO.LOW);    //led off  
    background(102); //set the fill color of led off  
  }  
}
```



RED background



Gray background

## Lab: A Processing LED Flasher...



### Processing LED Flasher: Demo



<https://youtu.be/OLGJvImOUGE>

## Question 5



**What Processing Language instruction allows background switching?**



# Thank you for attending

Please consider the resources below:

- O'Sullivan, D., & Igoe, T. (2004). *Physical computing: Sensing and controlling the physical world with computers*. Boston, MA: Thompson.
- Reas, C., & Fry, B. (2015). *Getting started with processing* (2nd ed). Make: Community.
- Freenove Ultimate Starter Kit for Raspberry Pi 4 B 3 B+
- [https://www.amazon.com/Freenove-Raspberry-Processing-Tutorials-Components/dp/B06W54L7B5/ref=sr\\_1\\_1?crd=3P2WAY3NHNYUU&dchild=1&keywords=freenova&qid=1596553527&srefix=frenova%2Caps%2C153&sr=8-1](https://www.amazon.com/Freenove-Raspberry-Processing-Tutorials-Components/dp/B06W54L7B5/ref=sr_1_1?crd=3P2WAY3NHNYUU&dchild=1&keywords=freenova&qid=1596553527&srefix=frenova%2Caps%2C153&sr=8-1)
- MIPI DSI:  
<https://www.raspberrypi.org/documentation/hardware/display/>
- Processing Language:  
<https://pi.processing.org/download/>  
<https://processing.org/tutorials/drawing/>
- Raspberry Pi 4 GPIO:  
<https://www.raspberrypi.org/documentation/usage/gpio/>
- Raspberry PiCamera Project:  
<https://projects.raspberrypi.org/en/projects/getting-started-with-picamera>
- Raspberry Pi 4 Specification:  
<https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>





**DesignNews**

Thank You

Sponsored by

