UCLA ENGINEERING
Mechatronics and Controls Laboratory

MAE162E
Week 5: Serial Communications

Lecture 4

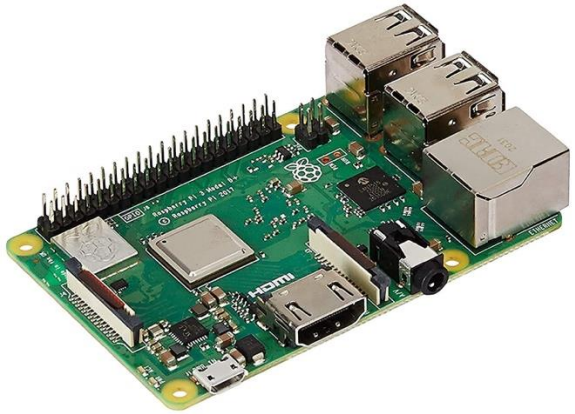
Instructor: Tsu-Chin Tsao
TAs: Will Shih, Toby Chen, Kevin Lai, Jing-Yung Huang.
Mechanical and Aerospace Engineering Department, UCLA

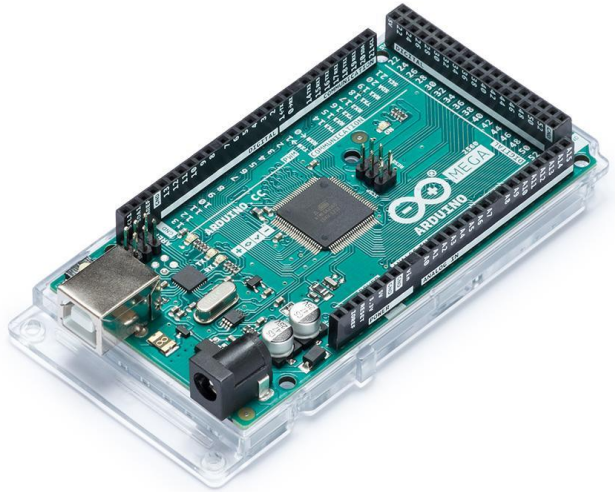
Spring 2025

Schedule

Week	Lecture	Lab	Assignment
1 (4/3)	Course overview; Electronics	Raspberry Pi and camera	Submit all the PAC order forms by 4/15
2 (4/10)	Team meeting: electronics review (final orders)	Python image processing and object detection	**Interim Report 1**
3 (4/17)	Programming	Path planning algorithms and (GPS)	Submit any additional orders to TA by 4/22
4 (4/24)	Team meeting: control strategy review	Device fabrication and assembly	**Interim Report 2**
5 (5/1)	Serial Communication	Develop/Implement code and module test (Program for navigation, pickup, etc)	Same as lab
6 (5/8)	Team meeting: progress status; assembly review	Develop/Implement code and module test (Program for navigation, pickup, etc)	**Interim Report 3**
7 (5/15)	TBD	Develop/Implement code and module test (Program for navigation, pickup, etc)	Same as lab
8 (5/22)	Team meeting: final review	Develop/Implement code and module test (Program for navigation, pickup, etc)	Same as lab
9 (5/29)	Holiday	**Project Demonstration** (demo videos will be graded)	Final design report and oral presentation
10 (6/5)	**Oral Presentation** (Each team 15 mins)	**Competition** (Each team 30 mins)	**Final Design Report (Integrate the three reports)**

Objective



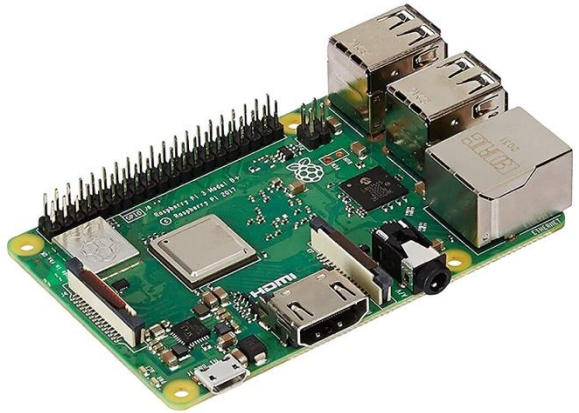


Detected

Serial Communication

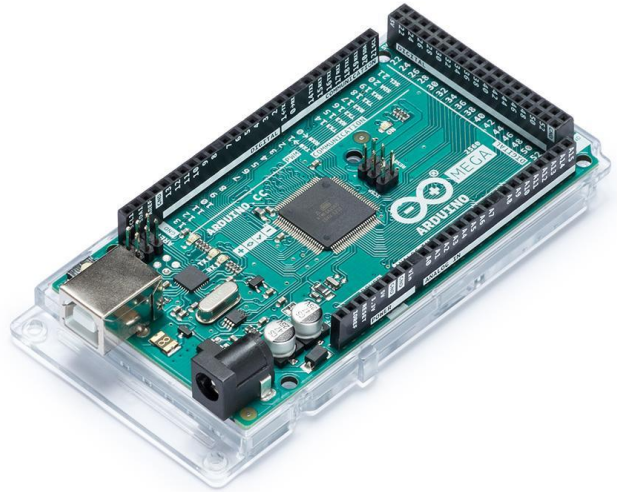
Stop the rover

Physical Protocol and Data Encoding



1. Physical Protocol
2. Data encoding

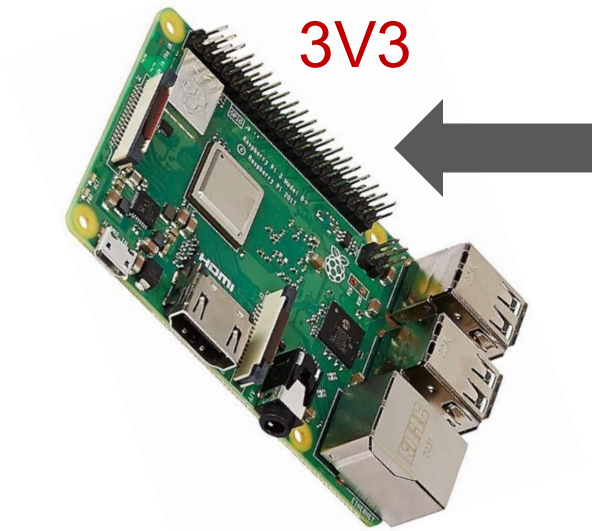
Serial Communication



Serial Communication: Physical Protocol

- Examples: IIC, SPI, UART, USB, RS232, RS485, etc.
- Considerations:
  - Availability
  - Transmission speed
  - Noise Immunity

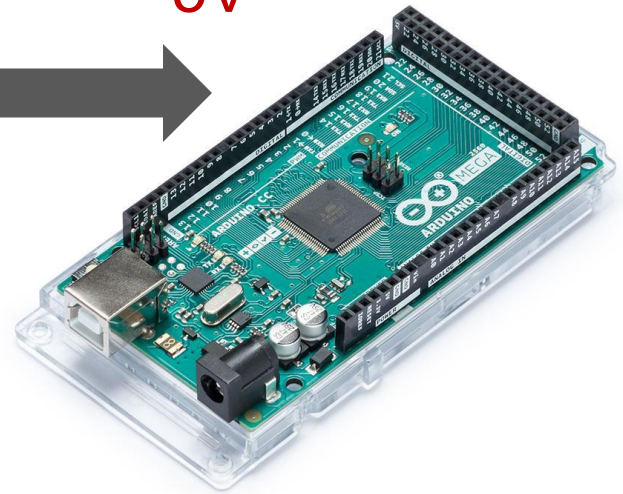
Serial Communication in Today's Lab



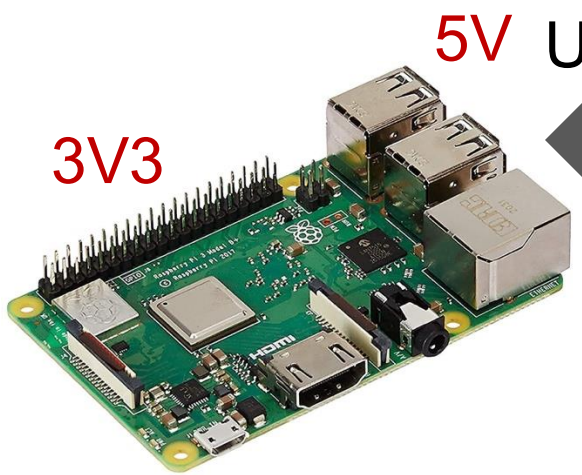
3V3

UART

5V



Serial Communication in Today's Lab



5V

3V3

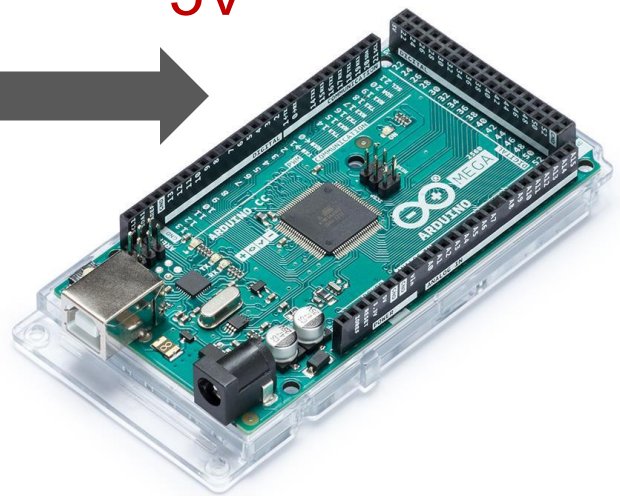
USB

CP2102

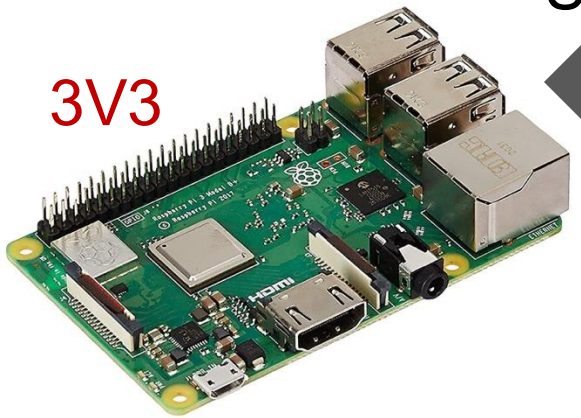
USB/UART converter

UART

5V



Serial Communication in Today's Lab



3V3

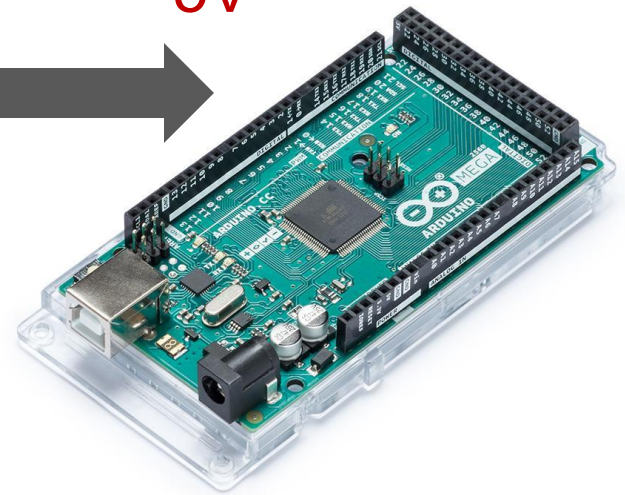
USB

CP2102

USB/UART converter

UART

5V



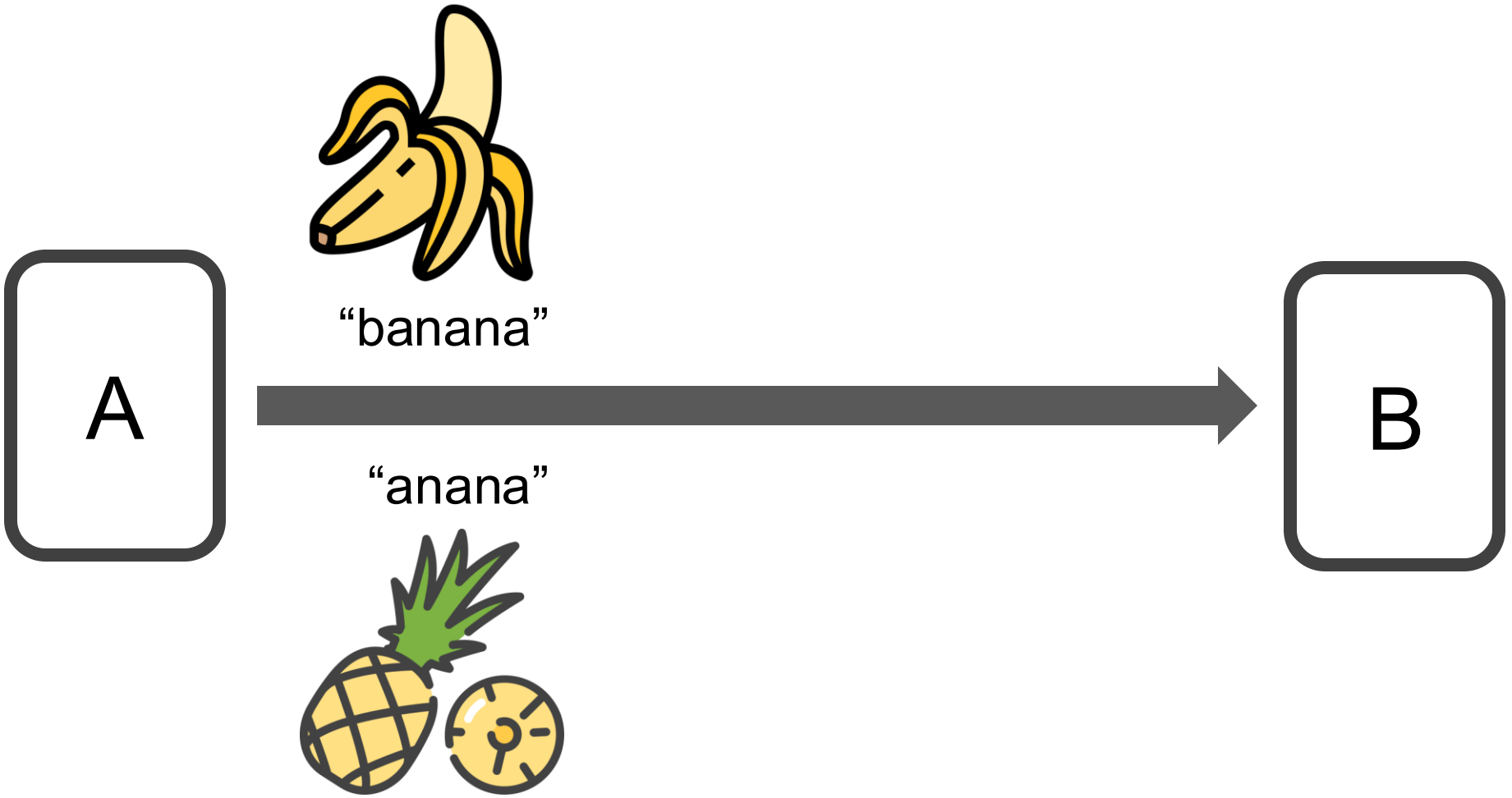
```
import serial
ser = serial.Serial("USB0", 9600)

ser.write(0xff)
data = ser.read(ser.in_waiting)
```

0xFF

0xFC

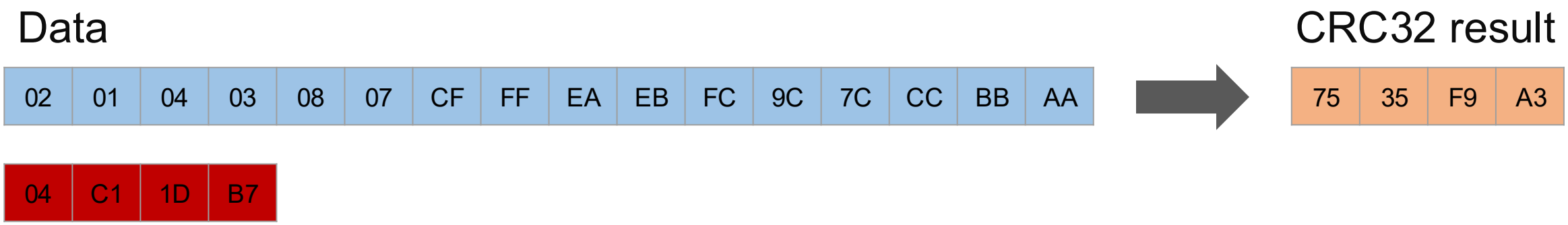
```
Serial1.begin(9600);

Serial1.write(0xfc);
char data = Serial.read();
```

Data Encoding: Error Detection



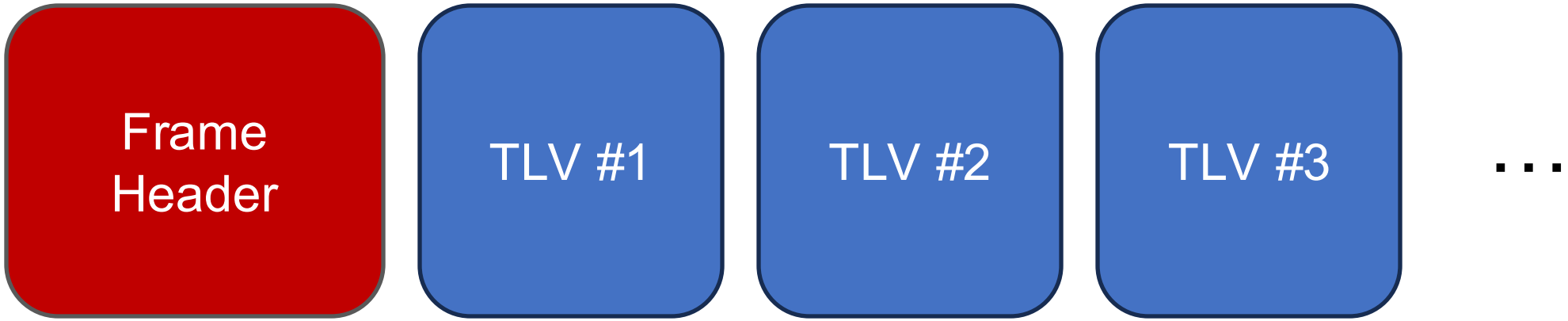
Error Detection: CRC32

- CRC32
  - Perform **binary polynomial division** using a fixed generator (e.g., 0x04C11DB7 for CRC-32)
  - Detects
    - All single-bit errors
    - All double-bit errors
    - All odd number of bit errors
    - All burst errors up to 32 bits
    - Most burst errors longer than 32 bits

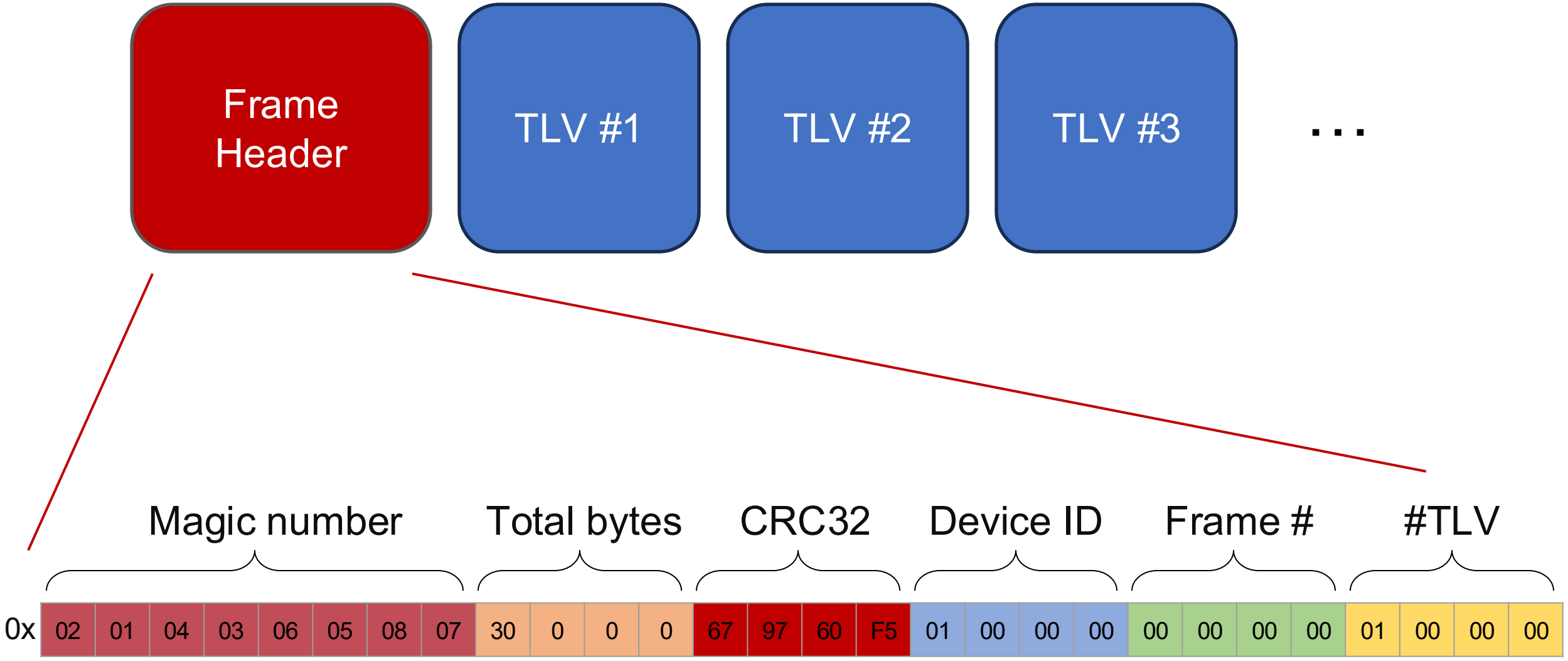


Data Encoding: TLV format

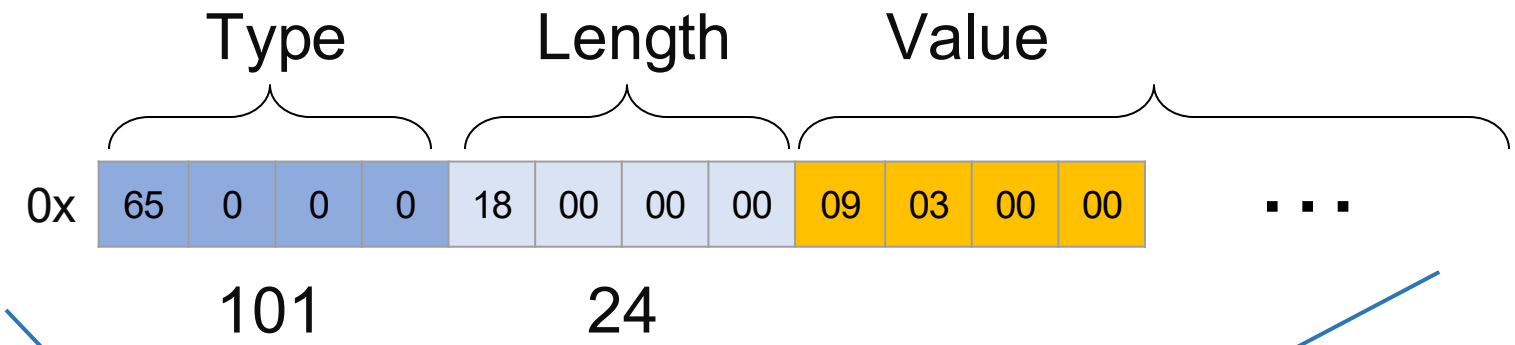
- Design objective
  - Efficiency
  - Flexible data format
  - Noise immunity: Magic number, CRC32 check

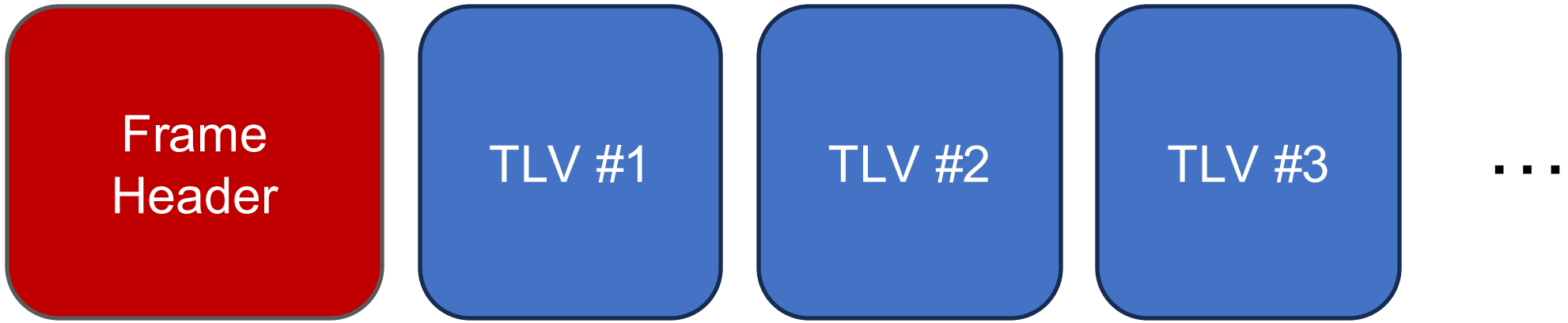


Data Encoding: TLV format

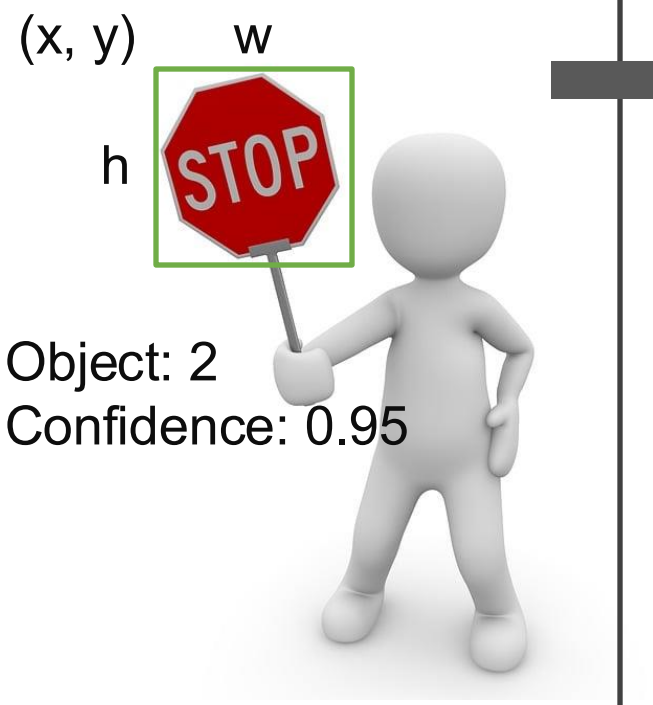


Data Encoding: TLV format





Data Encoding and Decoding



MessageCenter.py

add_yolo_detection()

```
detection = Detection()
detection.object = ctypes.c_int(object)
detection.bbox.x = ctypes.c_int(bbox[0])
detection.bbox.y = ctypes.c_int(bbox[1])
detection.bbox.w = ctypes.c_int(bbox[2])
detection.bbox.h = ctypes.c_int(bbox[3])
detection.confidence = ctypes.c_float(confidence)
```

add_message()

```
def add_message(self, type, length, value):
    """Add a message to the encoder."""
    self.encoder.addPacket(type, length, value)
    self.messageCount += 1
```

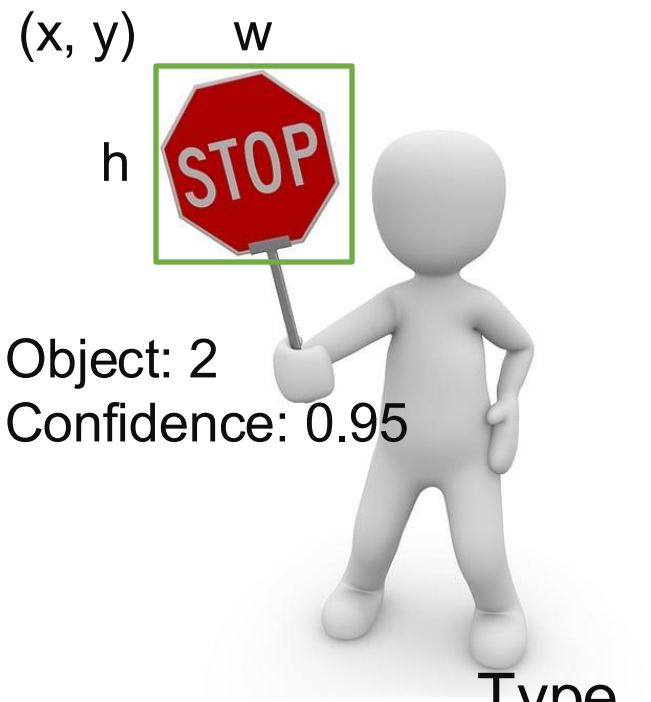
Encoder

Frame Header

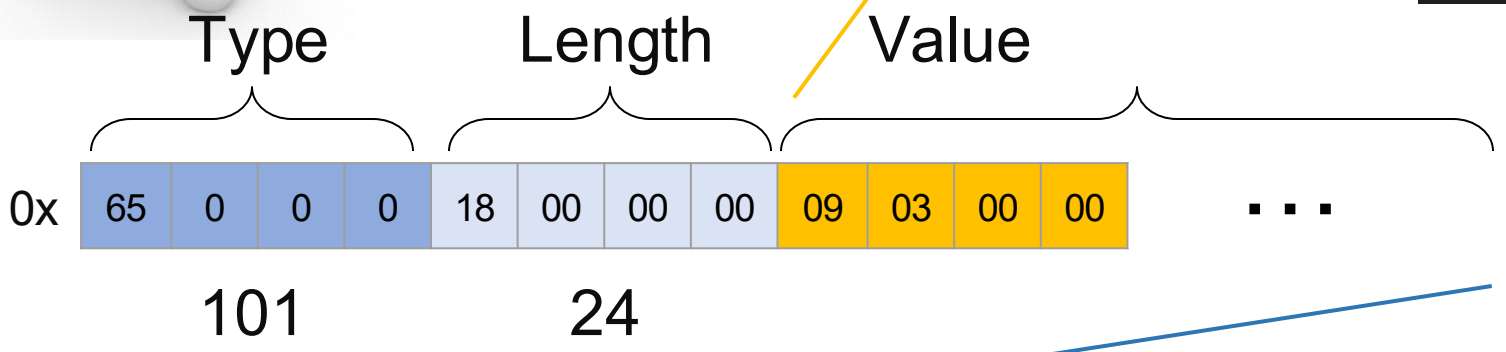
TLV #1

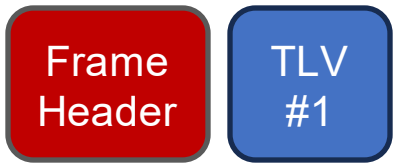
Send serial

Data Encoding and Decoding

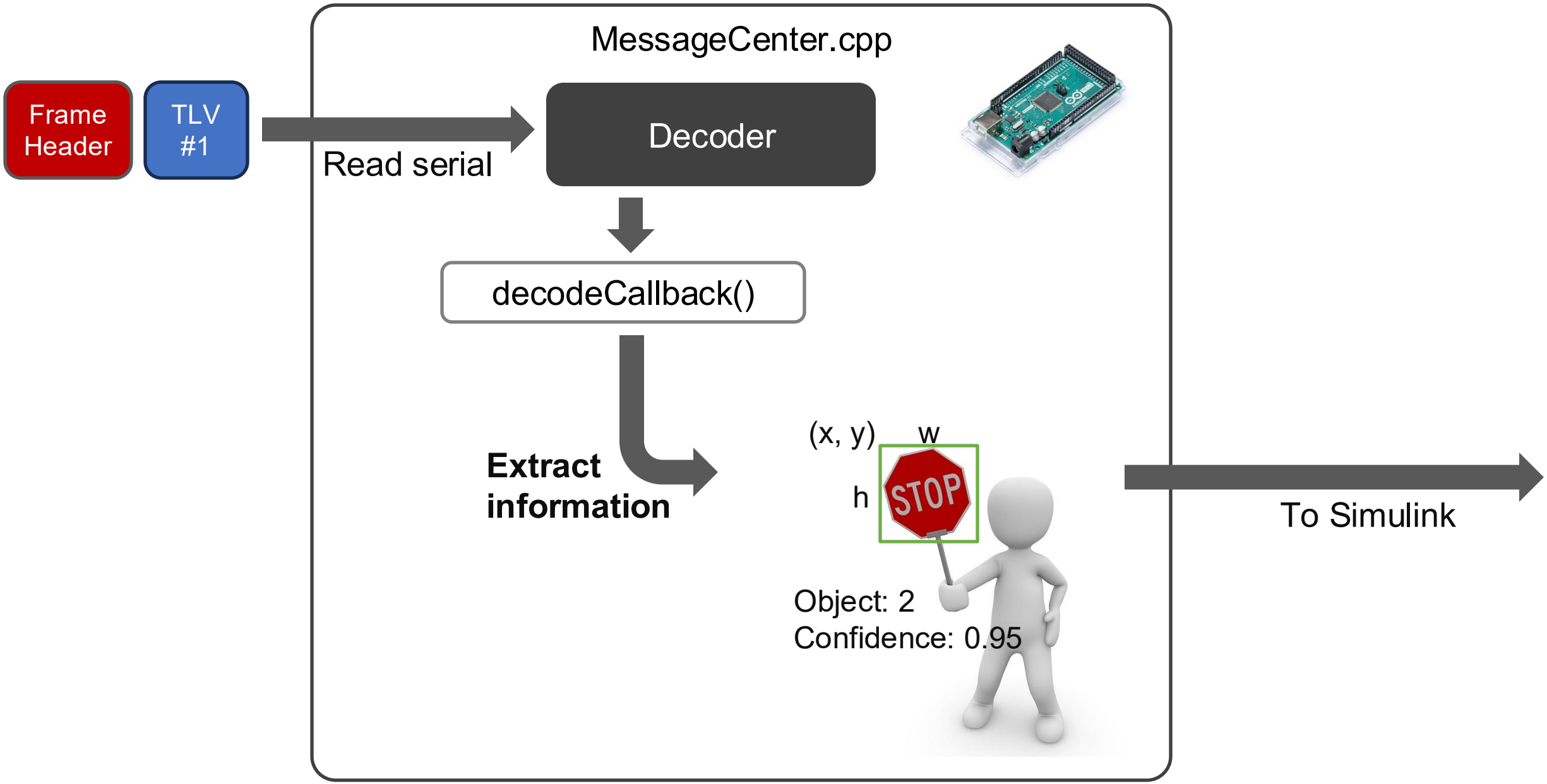



```
struct Detection
{
    int32_t object;
    struct BBox
    {
        int32_t x;
        int32_t y;
        int32_t w;
        int32_t h;
    } bbox;
    float confidence;
};
```

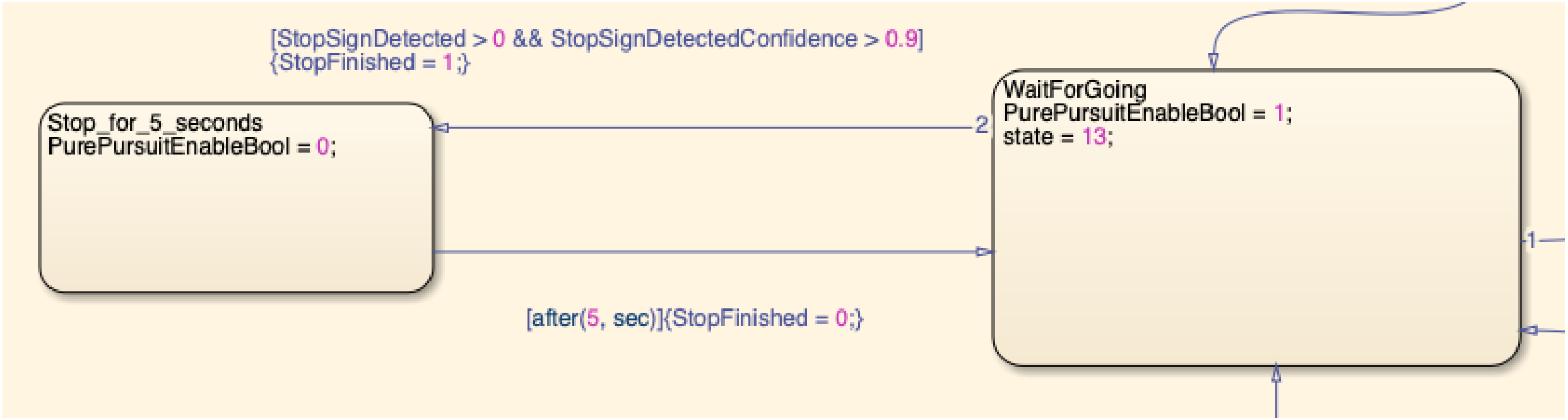




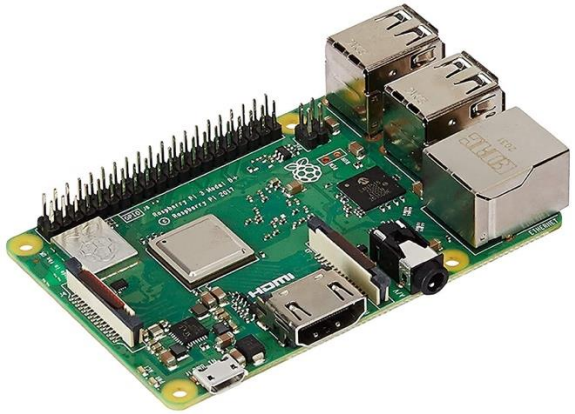
Data Encoding and Decoding

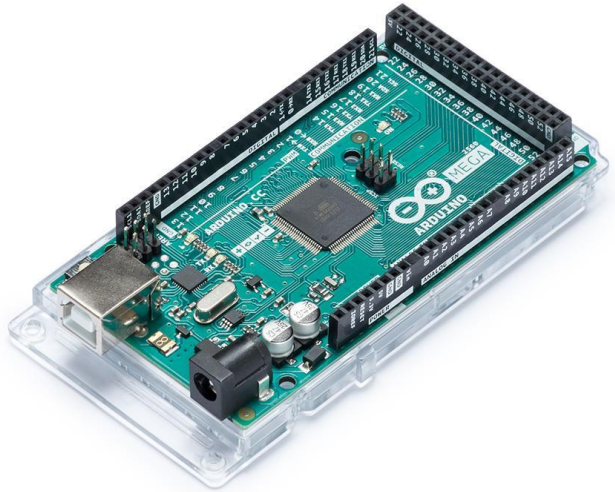


Data Encoding and Decoding



Back to the Objective





Detected

Serial Communication

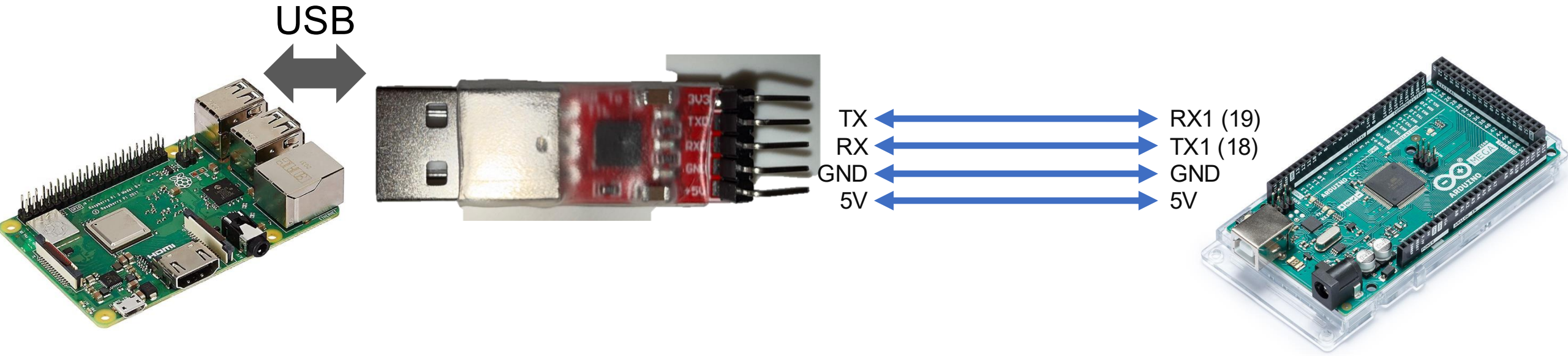
Stop the rover

The Lab

- Clone the code on both Raspberry Pi and your laptop

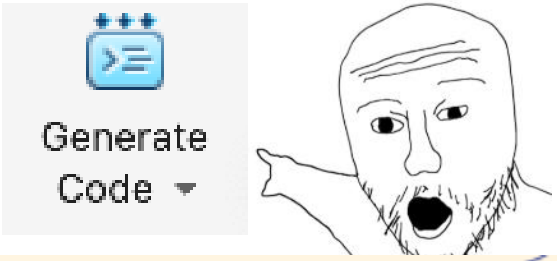
```
> git clone https://github.com/pochihh/MAE_162D-E_2025.git
```

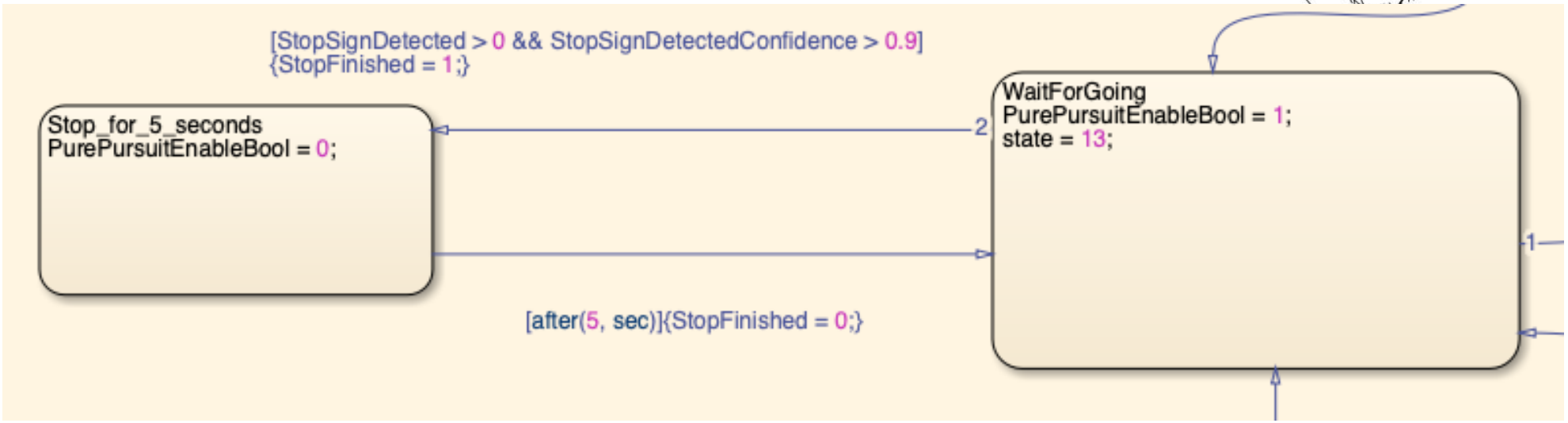
- Or you can visit the GitHub page: https://github.com/pochihh/MAE_162D-E_2025



For your Arduino

- Under **MAE_162D-E_2025/Arduino/src/**
- Open and run **“SimulinkGenerate.m”**
  - Select **Change Directory** if prompted
- Generate the codes (ControlLoop and StateflowBlock)
- Upload the code to Arduino





For your Raspberry Pi

- **Connect your laptop to the same Wi-Fi**
  - SSID: **MAE162_5G**
  - Password: **mae162mae162**
- Connect to your Raspberry Pi **(ssh recommended)**

```
> ssh pi@192.168.8.YOUR_IP
```

- Clone the project on RPi

```
> git clone https://github.com/pochihh/MAE_162D-E_2025.git
```

- Enter rpi/ folder

```
> cd MAE_162D-E_2025/rpi
```

For your Raspberry Pi

- Under rpi/, create a filder named weights/

```
> mkdir weights/
```

- Copy the **yolov4.weights** from week2's lab to **rpi/weights/**
  - Or else…

For your Raspberry Pi

- **Activate your conda environment**

```
> conda env list
```

```
> conda activate YOUR_ENV
```

- **Install pyserial and matplotlib if it's not installed already**

```
> conda install pyserial
```

```
> conda install matplotlib
```

For your Raspberry Pi

- **Identify your USB device**

```
> dmesg | grep tty
```

- **You should see something like this**

```
[   23.456789] usb 1-1.3: pl2303 converter now attached to **ttyUSB0**
```

- **In main.py, modify USB device name if needed**

```
65      # initialize the message center
66      message_center = MessageCenter('/dev/ttyUSB0', 9600, args.debug)
```

For your Raspberry Pi

- **Run the code**

```
> python main.py
```

- **Options for the example code**

```
> python main.py [-d] [-gps]
```

  - **-d:** debug mode: prints debug messages
  - **-gps:** use GPS

- **Shut the Rpi down** before powering off to avoid damage

UCLA ENGINEERING
Mechatronics and Controls Laboratory

Thank you for your attention!

- Before you leave the classroom
  - Start your project already (time flies)
  - Show your results to TA
  - Remember your CHARGER
  - Remember your WATER BOTTLE