

CAPSTONE Meeting Notes

JULY 5

- Camera Store Homeroom and Take Picture
- External power source of all the components

Notes:

-You can only initialize a variable once in a program, so cannot put in a loop because it will initialize again and crash

pointer

```
int x = 3;  
int* i = &x = 1775 (address of x)  
*i = 3
```

fs::FS (fs is the library in which FS is stored in, FS is a class)

variable

- global variable
- local variable

int x: write x here becomes global

```
void function1 {  
int x;  
x=5;  
}
```

```
void function2 {  
print(x) doesn't work because x is local variable  
}
```

JUNE 14

Lesson:

-When looping, sometimes when the loop isn't finished and the button is pressed, it won't work because the loop is still trying to get back. To fix this problem, they use event.

- Event-driven programming - flow of program is determined by sensors, keyboard inputs, etc
- Starts loop, and triggers call-back function = listens to event and when

- something happens, does some action
- Wire.begin = starts listener loop

1 byte = 8 bit

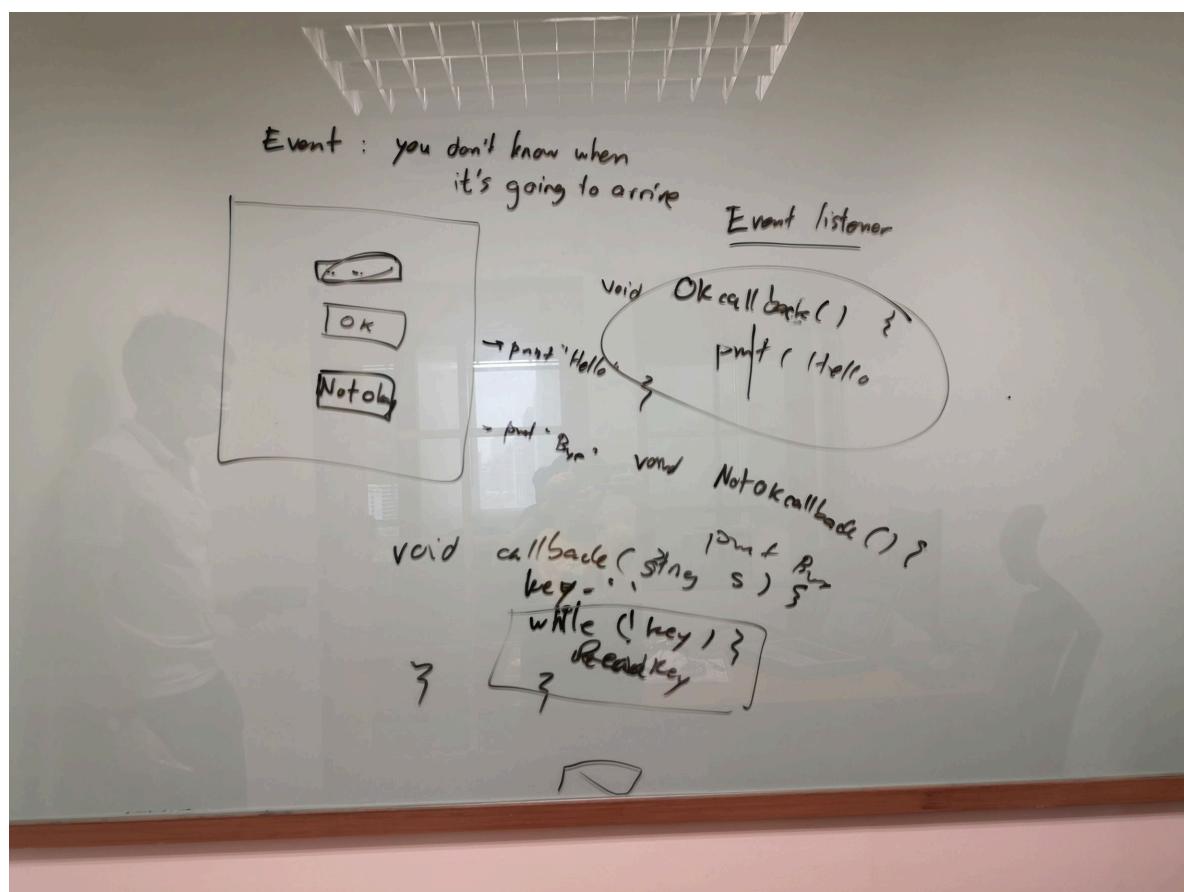
byte = written in binary from computer

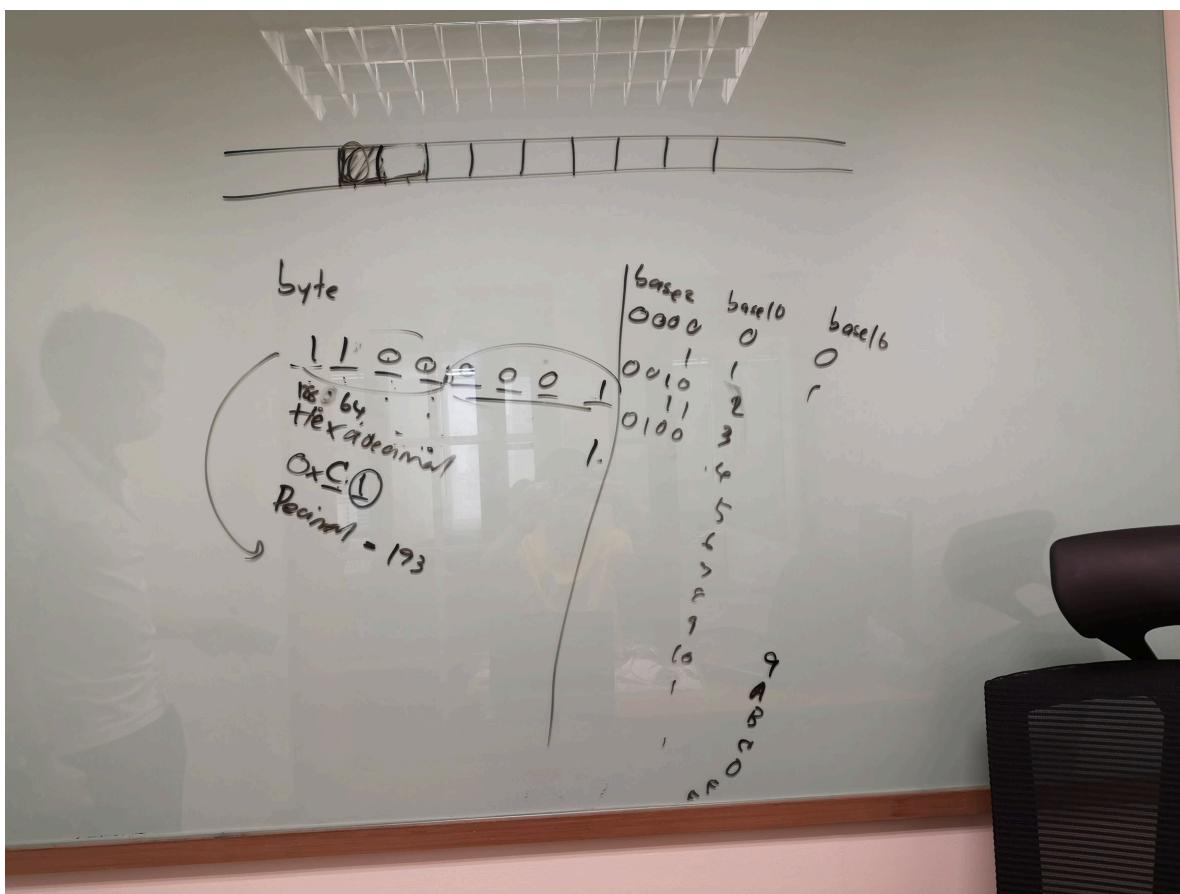
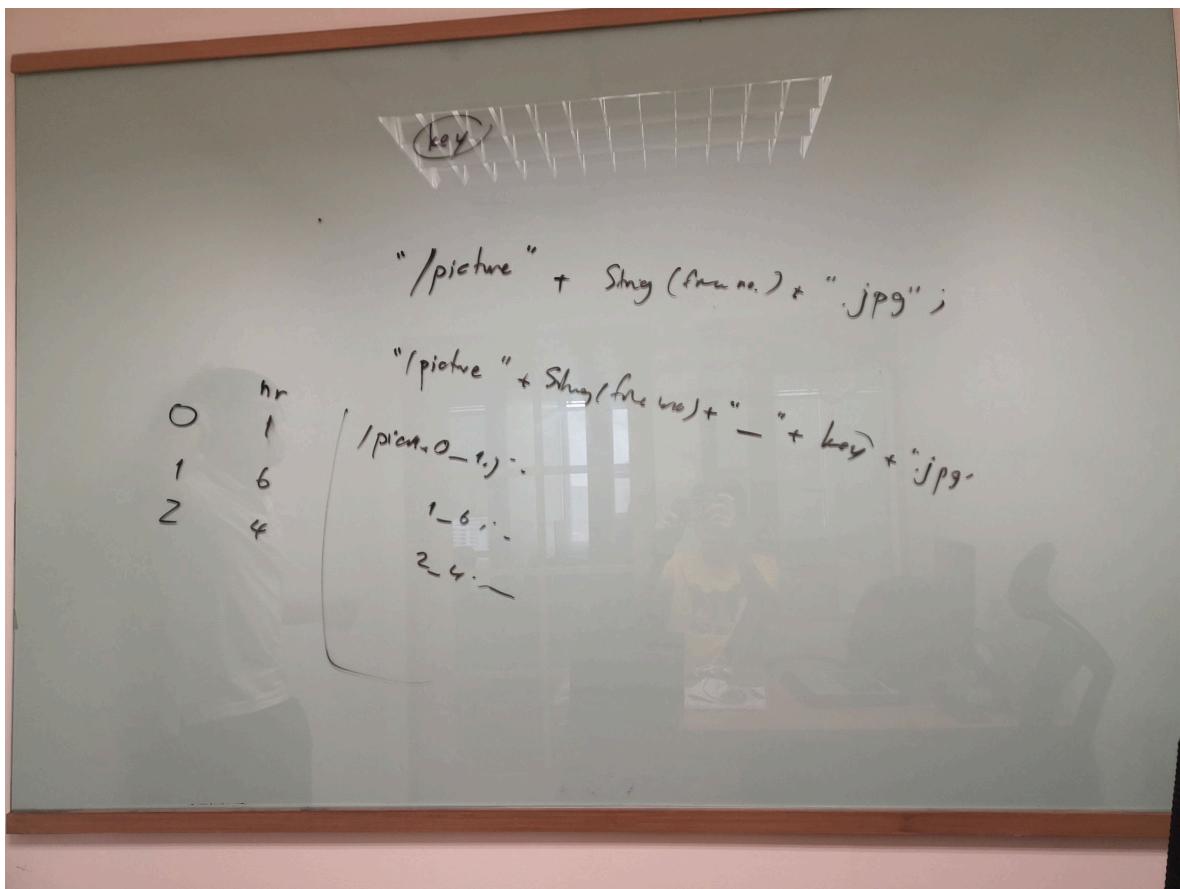
Wire.write(data, length) length = number of bytes

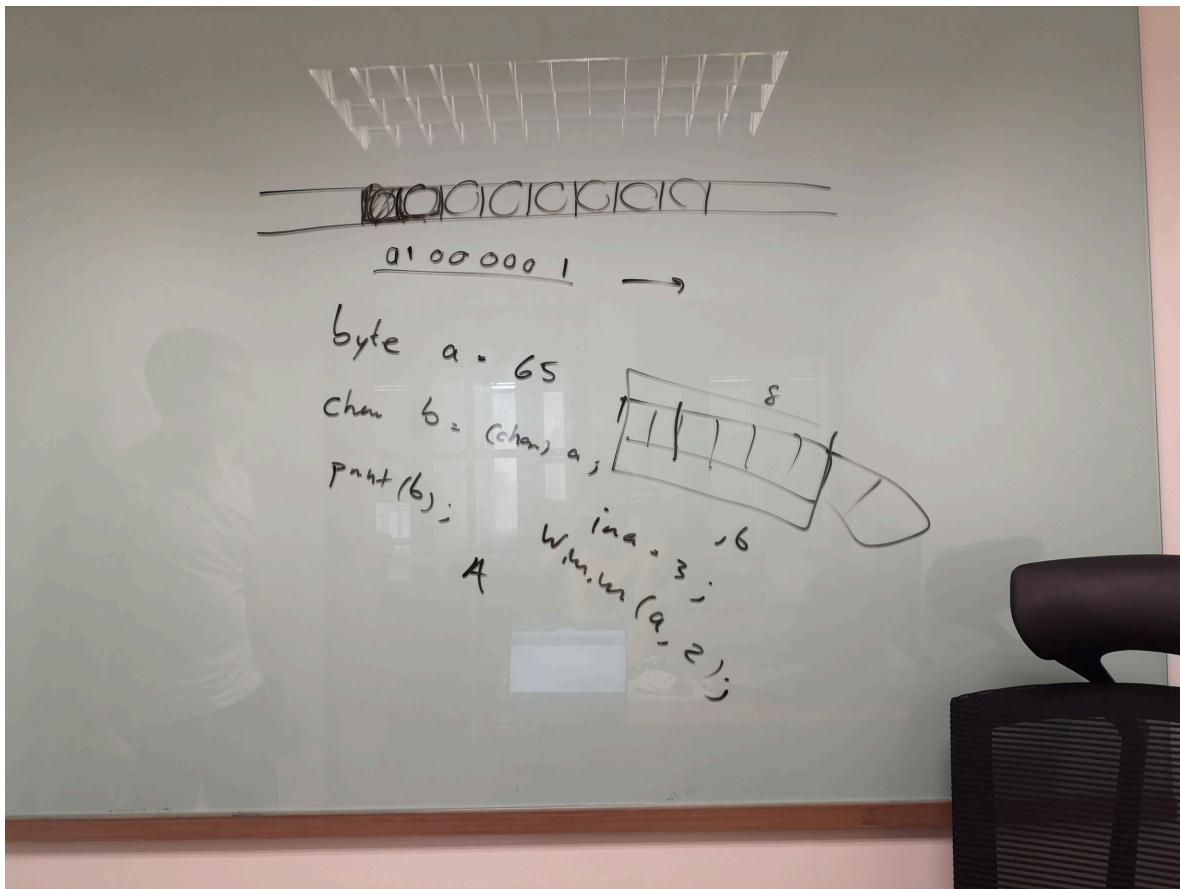
onEventKeypadNumber(callback function TakePicture)

TakePicture(homeroom)

1. communication between master and slave
2. change callback from print SerialMonitor to take picture with new filename







11 JUNE

- Using door latch instead of lock and motor (12V Solenoid Lock)
<https://shopee.co.th/product/38626007/2756036884?v=535&smtt=0.0.4>

<https://www.youtube.com/watch?v=NyRXzKMM6UE>

Steps:

Initial: Top door locked, layer unlocked, camera and flashlight turned off

1. Key in number pad - unlocks top door
2. Lift top door (string attached to lock layer)
3. Throw trash in
4. Close top door (hall sensor locks top door)
5. Flashlight blinks
6. Snap image, save
7. Layer unlocked, falls down

8. Restart

30 MAY

Tips for soldering: heat up both objects first, then connect them up (text p Stock for details if forgot)

- Got light emitter to work (solder the wires and light together, use the right resistance for brighter light)
- For the ESP32 CAM external power source, wires also have to be soldered for them to work.
- Got hall sensor to work (use digital pins to detect presence of magnet, and digital read on Arduino code)
- Overall code details (how to integrate all components of the box, use variable named steps)
- Learn to use motor and lock
- Learn to use keypad
- Sending data from ESP32-CAM directly to Arduino may be difficult, ESP32 CAM serves as its own board and connecting it directly to keypad would work, where it receives data from the keypad and names the images accordingly. Then, the ESP32 CAM just connects to the Arduino through one wire and sends data that the picture has been taken, so that other steps can follow through
- Order the correct dimensions of the box and design it properly (talk with dad and conclude with p Stock)