Midterm Project

Team 4

開發環境

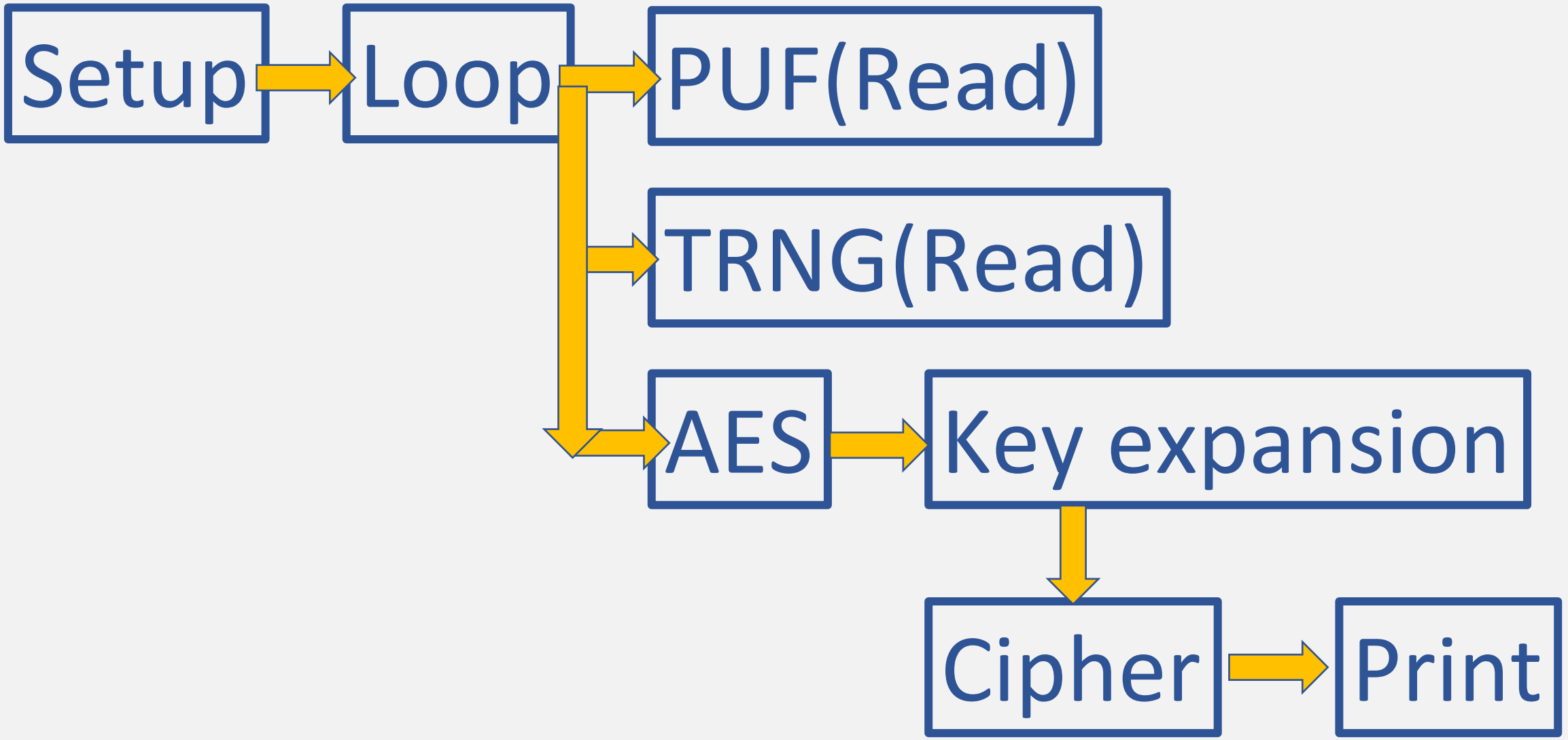
macOS Catalina 10.16.7

Sublime text 3 with Arduino-like IDE

Platform : Arduino SAM(32bits)

Arduino DUE(Programming port)

Overall Flowchart



Part 0 Setup

Set baud = 9600 times/sec

input :

pinMode(m , INPUT)

output :

pinMode(n , OUTPUT)

Part 1 (PUF data讀取)

PUF with 4096 bits.

TRNG with 8 bits.

.

Part 1 (PUF data讀取)

pinMode=0 -> Read_PUF()
8bits a time, 512 times.
pinMode=1 -> Read_TRNG()
8bits a time, 1 time.

Part 1 (PUF data讀取)

PUF_flag = 0 originally.

PUF_flag = 1 if all the 4096 bits being read.

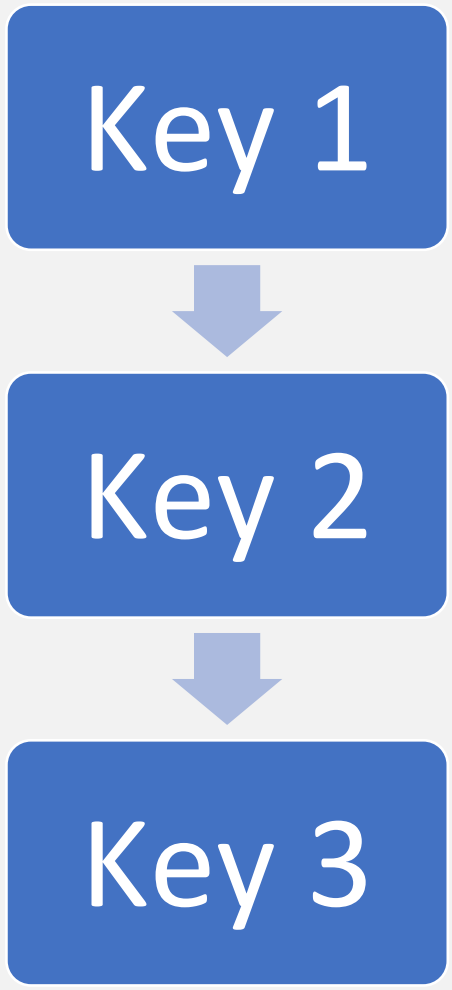
Part2 (AES-128 coding & 加密演練)

- Being activated only if PUF_flag = 1.
- AES with 128 bits.
- Generate **plain text** from random()
- Generate **Key** from PUF_data[0:127](changable)

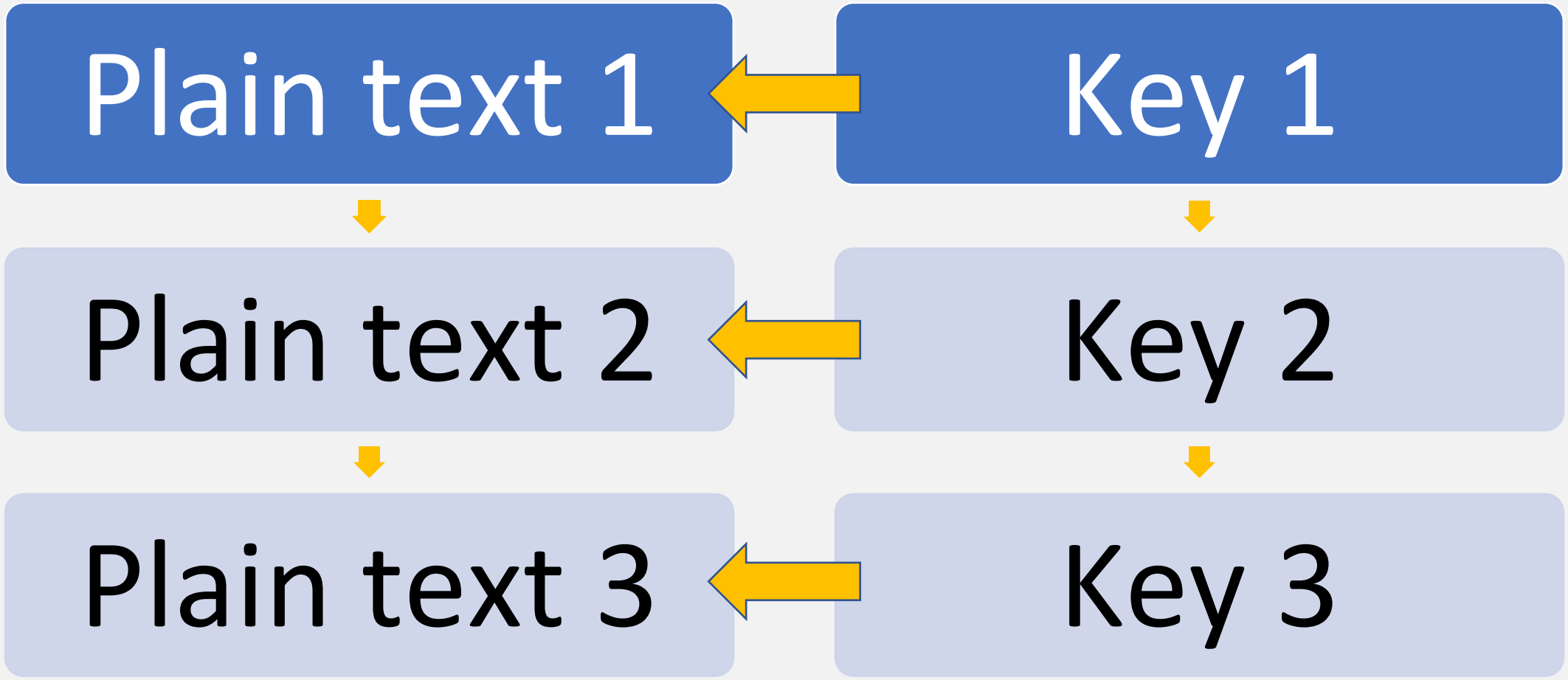
Part2 (AES-128 coding & 加密演練)

- int binary_plain text[128] -> char **in[16]**
- int binary_key -> char **key[16]**

Key expansion



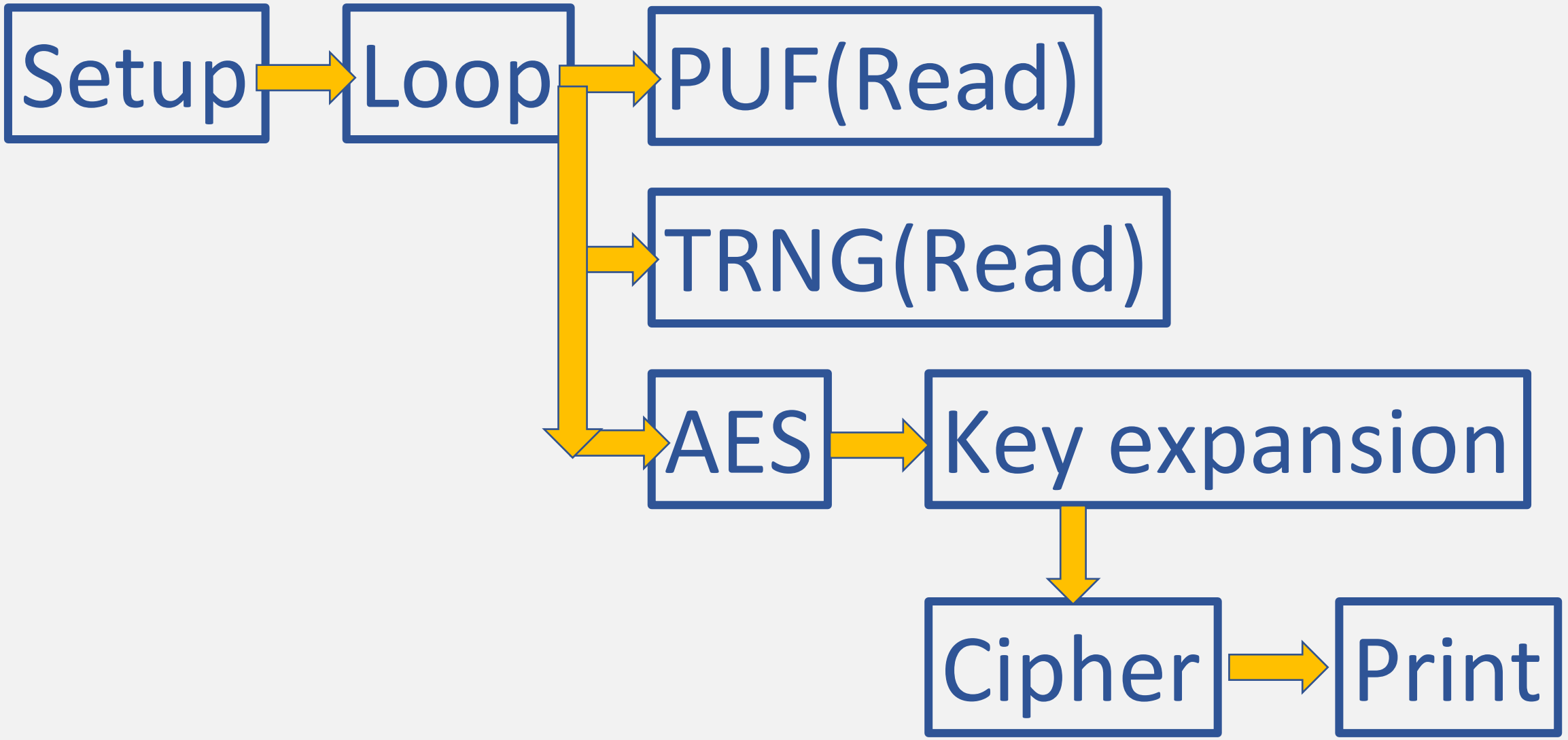
Cipher



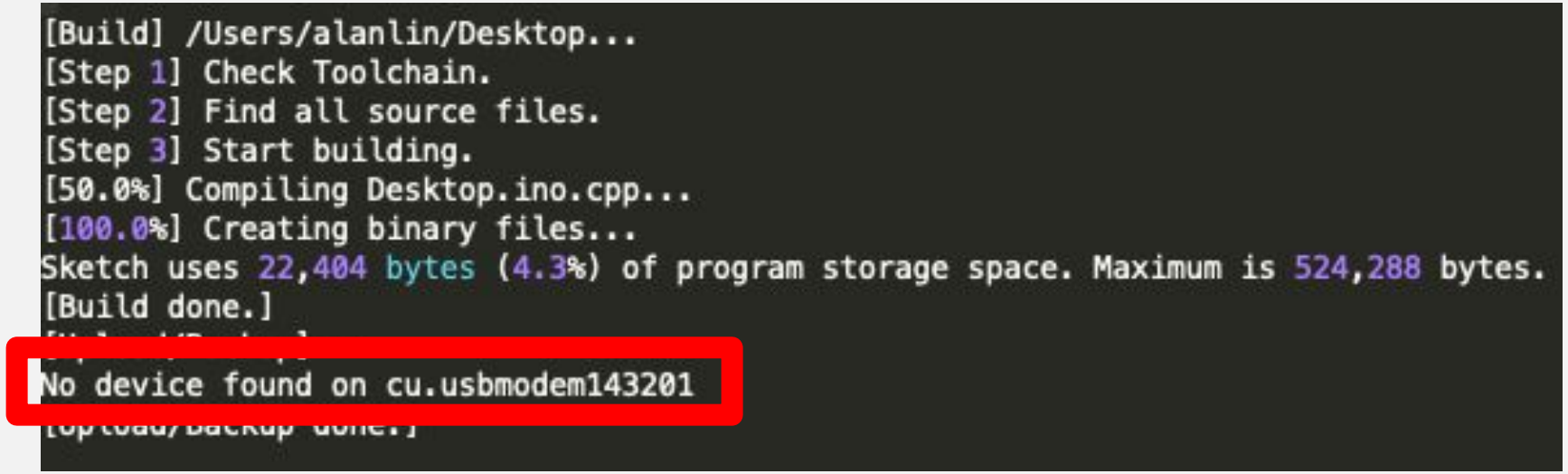
Print Out

```
Serial.print(out[0],HEX)
```

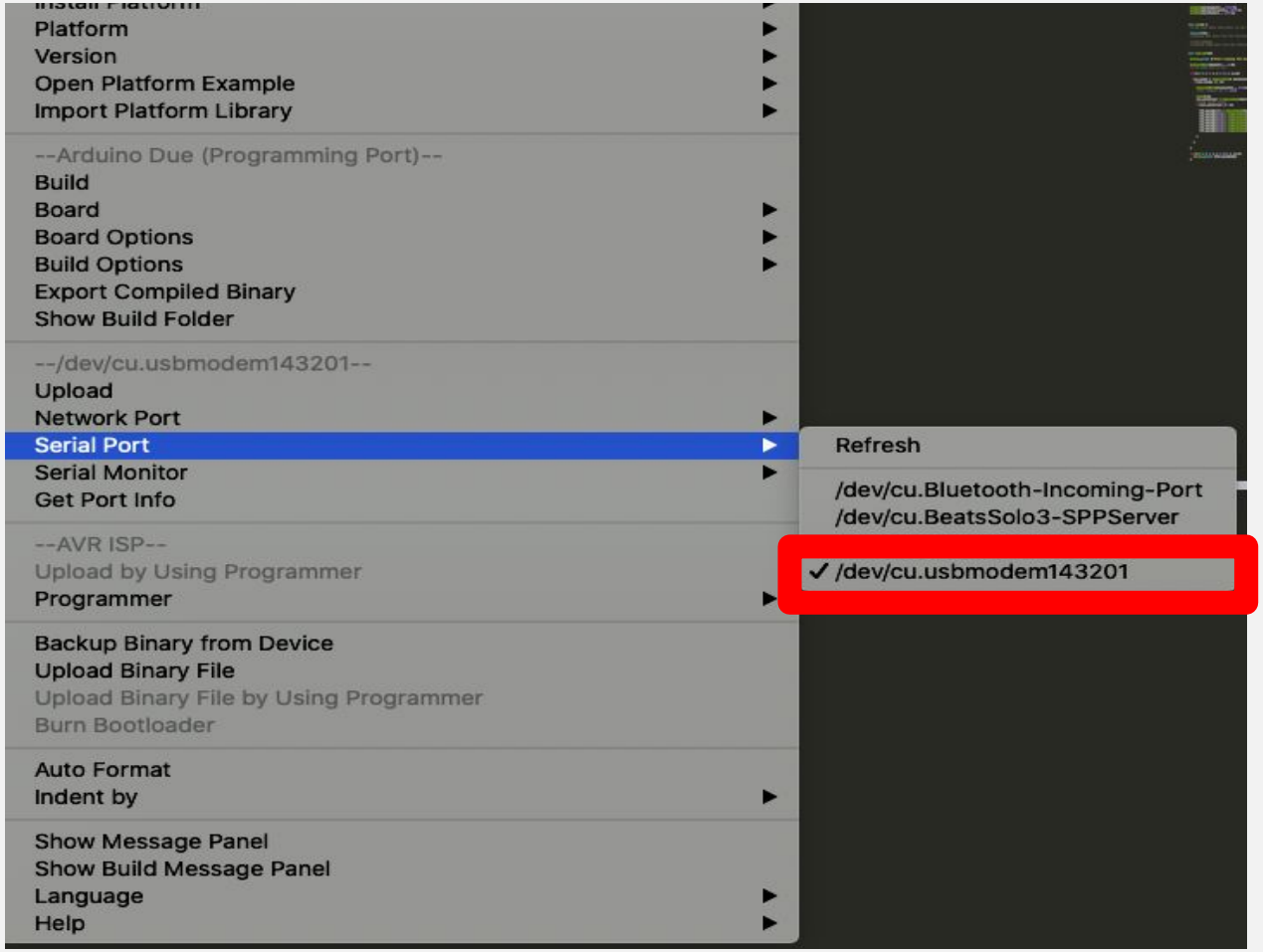
Summary


Setup
Loop
PUF(Read)
TRNG(Read)
AES
Key expansion
Cipher
Print

遇到的問題-1


[Build] /Users/alanlin/Desktop...
[Step 1] Check Toolchain.
[Step 2] Find all source files.
[Step 3] Start building.
[50.0%] Compiling Desktop.ino.cpp...
[100.0%] Creating binary files...
Sketch uses 22,404 bytes (4.3%) of program storage space. Maximum is 524,288 bytes.
[Build done.]
No device found on cu.usbmodem143201

遇到的問題-1


Platform
Version
Open Platform Example
Import Platform Library
--Arduino Due (Programming Port)--
Build
Board
Board Options
Build Options
Export Compiled Binary
Show Build Folder
--/dev/cu.usbmodem143201--
Upload
Network Port
Serial Port
Serial Monitor
Get Port Info
--AVR ISP--
Upload by Using Programmer
Programmer
Backup Binary from Device
Upload Binary File
Upload Binary File by Using Programmer
Burn Bootloader
Auto Format
Indent by
Show Message Panel
Show Build Message Panel
Language
Help
Refresh
/dev/cu.Bluetooth-Incoming-Port
/dev/cu.BeatsSolo3-SPPServer
✓ /dev/cu.usbmodem143201

遇到的問題-1

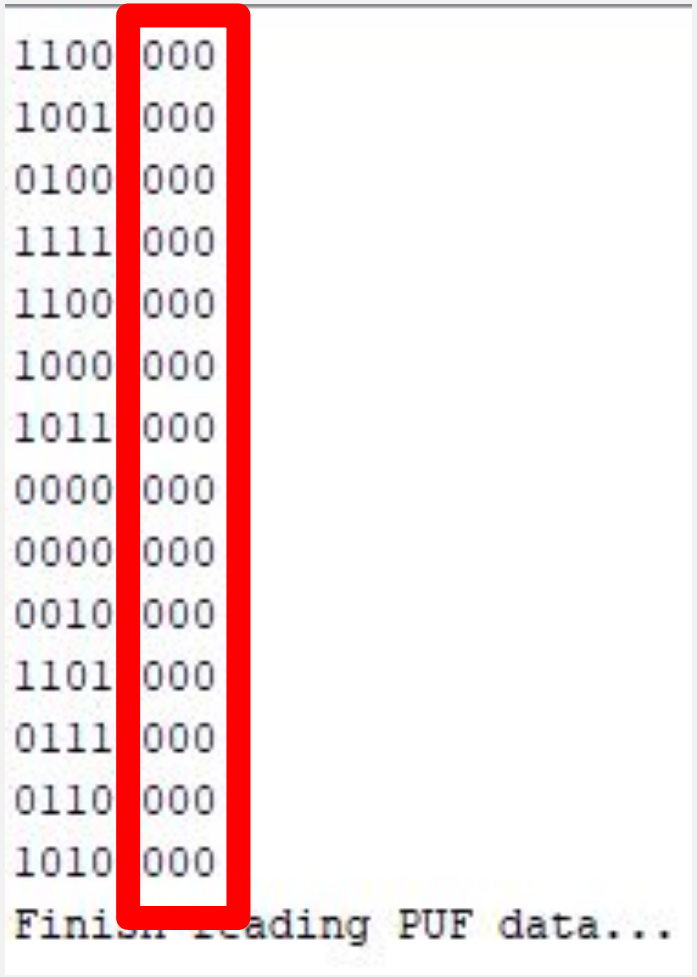

MacBook Pro
硬體
ATA
Apple Pay
FireWire
NVMExpress
PCI
SAS
SATA/SATA Express
SPI
Thunderbolt
USB
乙太網路卡
儲存裝置
光纖通道
印表機
平行 SCSI
控制器
燒錄光碟
相機
藍牙
記憶體
診斷
讀卡機
電源
音訊
顯示卡/顯示器
網路
WWAN
Wi-Fi
卷宗
USB 裝置樹
Ambient Light Sensor
FaceTime HD 攝影機（內建）
Apple T2 Controller
USB 3.1 匯流排
Arduino Due Prog. Port
USB 3.1 匯流排
Arduino Due Prog. Port：
產品識別碼： 0x003d
廠商識別碼： 0x2341
版本： 0.01
序號： 55839313638351A031A2
速度： 最高每秒12 Mb
位置識別碼： 0x14320000 / 6
所需電流（mA）： 100
額外的操作電流（mA）： 0
Alan的MacBook Pro › 硬體 › USB › USB 3.1匯流排 › USB2.1 Hub › Arduino Due Prog. Port

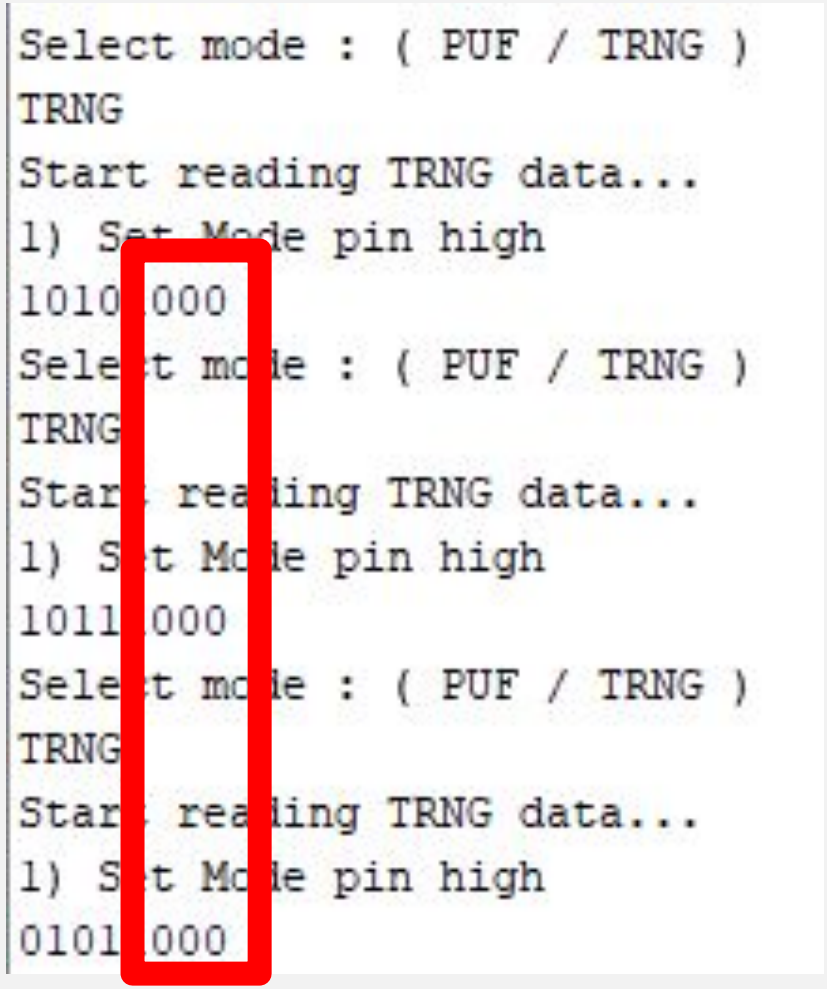
遇到的問題-2

```
const byte dataPin[8] = {6 , 7 , 8 , 9 , 10 , 11 , 12 , 13};  //8-bit data
//const byte dataPin[8] = {13 , 12 , 11 , 10 , 9 , 8 , 7 , 6};  //8-bit data
```

```
pinMode(dataPin[8] , INPUT);
```

遇到的問題-2


1100 000
1001 000
0100 000
1111 000
1100 000
1000 000
1011 000
0000 000
0000 000
0010 000
1101 000
0111 000
0110 000
1010 000
ading PUF data...


Select mode : (PUF / TRNG)
TRNG
Start reading TRNG data...
1) Set Mode pin high
1010 000
Select mode : (PUF / TRNG)
TRNG
Start reading TRNG data...
1) Set Mode pin high
1011 000
Select mode : (PUF / TRNG)
TRNG
Start reading TRNG data...
1) Set Mode pin high
0101 000

遇到的問題-2

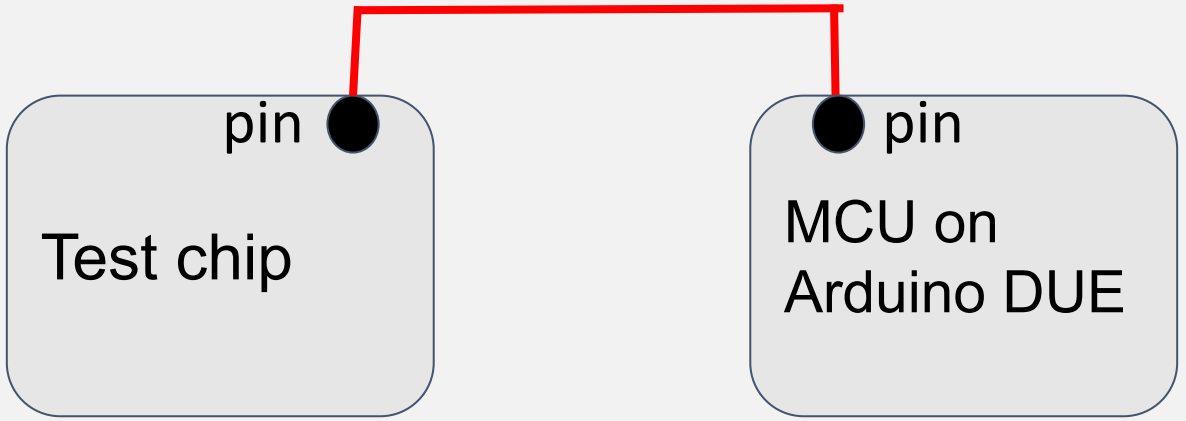
```
//8-bit data

const byte dataPin_0 = 6;
const byte dataPin_1 = 7;
const byte dataPin_2 = 8;
const byte dataPin_3 = 9;
const byte dataPin_4 = 10;
const byte dataPin_5 = 11;
const byte dataPin_6 = 12;
const byte dataPin_7 = 13;
```

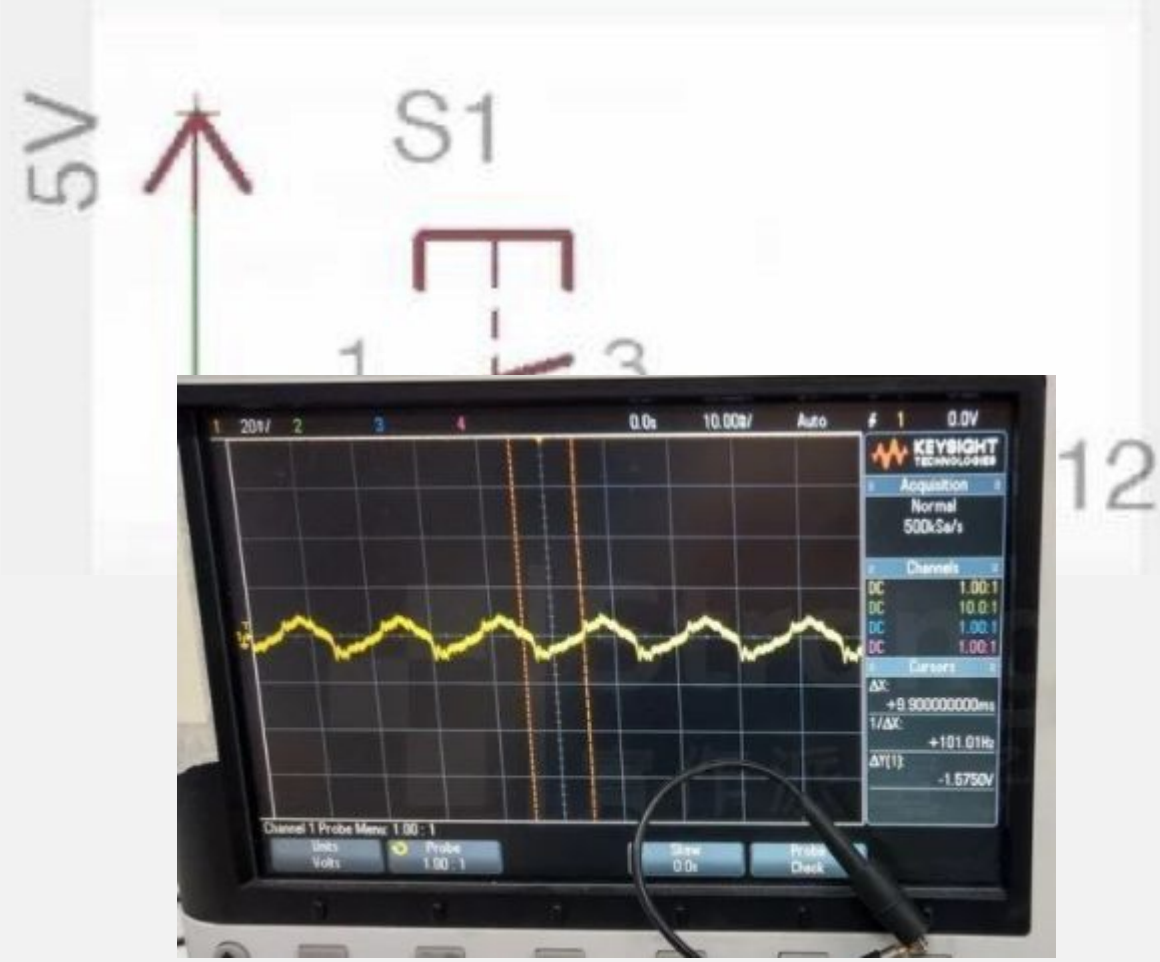
```
pinMode(dataPin_0 , INPUT);
pinMode(dataPin_1 , INPUT);
pinMode(dataPin_2 , INPUT);
pinMode(dataPin_3 , INPUT);
pinMode(dataPin_4 , INPUT);
pinMode(dataPin_5 , INPUT);
pinMode(dataPin_6 , INPUT);
pinMode(dataPin_7 , INPUT);
```

Polling

輪詢(Polling)是一種CPU決策如何提供週邊裝置服務的方式，又稱「程式控制輸入輸出」(Programmed I/O)。輪詢法的概念是：由CPU定時發出詢問，依序詢問每一個週邊裝置是否需要其服務，有即給予服務，服務結束後再問下一個週邊，接著不斷週而復始。



Polling

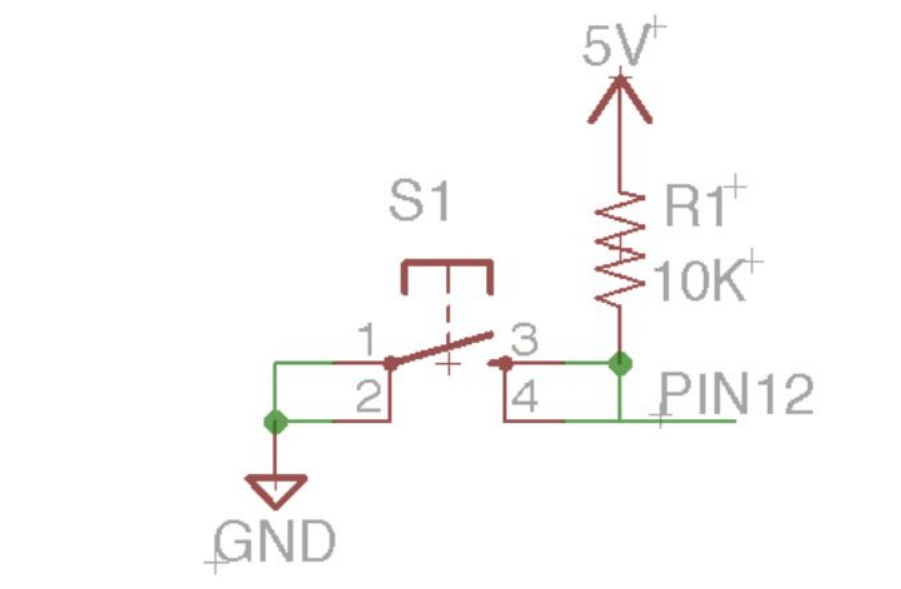



```
In setup():
pinMode( mypin , INPUT);

In loop():
if( digital.read ( mypin )  == HIGH){
   // do something
}
```

mypin is a Floating pin !

Polling

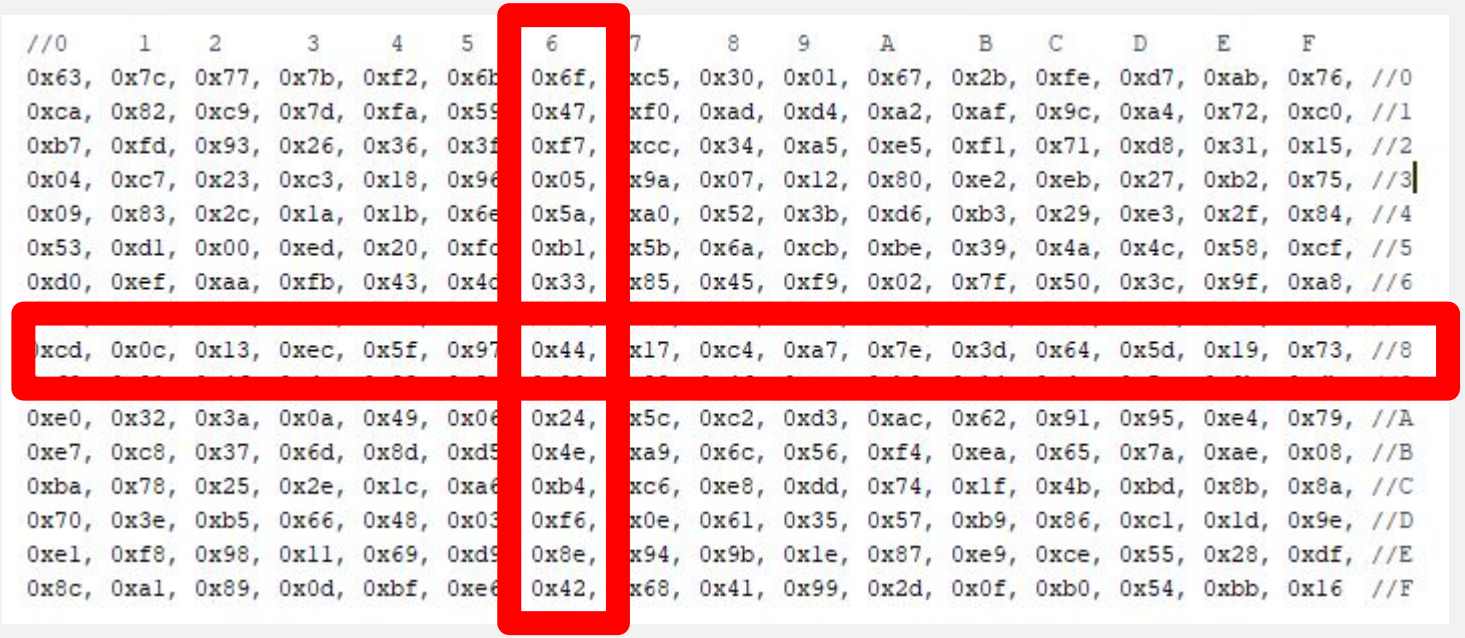



```
In setup():
pinMode( mypin , INPUT_PULLUP);

In loop():
if( digital.read ( mypin ) == HIGH){
    // do something
}
```

參考資料
:https://www.baldengineer.com/arduino-internal-pull-up-resistor-tutorial.html

S box generation



```
//0    1    2    3    4    5    6    7    8    9    A    B    C    D    E    F
0x63, 0x7c, 0x77, 0x7b, 0xf2, 0x6b, 0x6f, 0xc5, 0x30, 0x01, 0x67, 0x2b, 0xfe, 0xd7, 0xab, 0x76, //0
0xca, 0x82, 0xc9, 0x7d, 0xfa, 0x59, 0x47, 0xf0, 0xad, 0xd4, 0xa2, 0xaf, 0x9c, 0xa4, 0x72, 0xc0, //1
0xb7, 0xfd, 0x93, 0x26, 0x36, 0x3f, 0xf7, 0xcc, 0x34, 0xa5, 0xe5, 0xf1, 0x71, 0xd8, 0x31, 0x15, //2
0x04, 0xc7, 0x23, 0xc3, 0x18, 0x96, 0x05, 0x9a, 0x07, 0x12, 0x80, 0xe2, 0xeb, 0x27, 0xb2, 0x75, //3
0x09, 0x83, 0x2c, 0x1a, 0x1b, 0x6e, 0x5a, 0xa0, 0x52, 0x3b, 0xd6, 0xb3, 0x29, 0xe3, 0x2f, 0x84, //4
0x53, 0xd1, 0x00, 0xed, 0x20, 0xfc, 0xb1, 0x5b, 0x6a, 0xcb, 0xbe, 0x39, 0x4a, 0x4c, 0x58, 0xcf, //5
0xd0, 0xef, 0xaa, 0xfb, 0x43, 0x4d, 0x33, 0x85, 0x45, 0xf9, 0x02, 0x7f, 0x50, 0x3c, 0x9f, 0xa8, //6
0xcd, 0x0c, 0x13, 0xec, 0x5f, 0x97, 0x44, 0x17, 0xc4, 0xa7, 0x7e, 0x3d, 0x64, 0x5d, 0x19, 0x73, //8
0xe0, 0x32, 0x3a, 0x0a, 0x49, 0x06, 0x24, 0x5c, 0xc2, 0xd3, 0xac, 0x62, 0x91, 0x95, 0xe4, 0x79, //A
0xe7, 0xc8, 0x37, 0x6d, 0x8d, 0xd5, 0x4e, 0xa9, 0x6c, 0x56, 0xf4, 0xea, 0x65, 0x7a, 0xae, 0x08, //B
0xba, 0x78, 0x25, 0x2e, 0x1c, 0xa6, 0xb4, 0xc6, 0xe8, 0xdd, 0x74, 0x1f, 0x4b, 0xbd, 0x8b, 0x8a, //C
0x70, 0x3e, 0xb5, 0x66, 0x48, 0x03, 0xf6, 0x0e, 0x61, 0x35, 0x57, 0xb9, 0x86, 0xc1, 0x1d, 0x9e, //D
0xe1, 0xf8, 0x98, 0x11, 0x69, 0xd9, 0x8e, 0x94, 0x9b, 0x1e, 0x87, 0xe9, 0xce, 0x55, 0x28, 0xdf, //E
0x8c, 0xa1, 0x89, 0x0d, 0xbf, 0xe6, 0x42, 0x68, 0x41, 0x99, 0x2d, 0x0f, 0xb0, 0x54, 0xbb, 0x16  //F
```

Translate rows and column into binary

(row | column) = (8 | 6)

= (1000 0110)

S box generation

Calculate the multiplicative inverse of (row | column) in GF(2^8)

initial : p = 1 , q = 1

$p * q \equiv 1 \bmod x^8$

$(p * (x + 1)) * (q / (x + 1)) \equiv p * q \equiv 1 \bmod x^8$

$(p * (x + 1)^2) * (q / (x + 1)^2) \equiv p * q \equiv 1 \bmod x^8$

…

$(p * (x + 1)^{255}) * (q / (x + 1)^{255}) \equiv (p * 1) * (q / 1) \equiv p * q \equiv 1 \bmod x^8$

$(x + 1)^{255} \equiv 1 \bmod x^8$

Generate 255 pairs of (p , q) of 8 bits binary number + special solution(p = 0)

S box generation

Affine transformation

S_box[p]

q (multiplicative inverse of p)