



## เอกสารประกอบชิ้นงาน

รายวิชา Digital System Fundamentals รหัสวิชา 1076006

Smart lock box

### สมาชิก

- |                |              |                                |
|----------------|--------------|--------------------------------|
| 1.นายธัญธร     | พรสวัสดิ์ชัย | รหัสนักศึกษา 61010497 Sec เข้า |
| 2.นายณพนธ์     | พันลุดัน     | รหัสนักศึกษา 61010541 Sec เข้า |
| 3.นายพิพิธพงศ์ | จิตภักดีไทย  | รหัสนักศึกษา 61010750 Sec เข้า |
| 4.นายธนรัช     | อัครรัตนกร   | รหัสนักศึกษา 62015048 Sec บ่าย |
| 5.นายพีรวัส    | อำโต         | รหัสนักศึกษา 62015087 Sec บ่าย |

### นำเสนอ

รศ. ดร. เจริญ วงษ์ชุ่มเย็น

Idea/Useful	Techniques	Completeness	Report/Clip	Present

## คำนำ

ชิ้นงานนี้เป็นส่วนหนึ่งของรายวิชา Digital System Fundamentals รหัสวิชา 1076006 โดยมีจุดประสงค์เพื่อนำเอาความรู้ที่ได้จากรายวิชา Digital System Fundamentals มาประยุกต์ใช้ในชิ้นงานนี้ โดยมีความสำคัญในการนำความรู้ที่ได้รับมาใช้ควบคู่กับความคิดสร้างสรรค์ของผู้จัดทำเพื่อนำเสนอชิ้นงานซึ่งมีความเหมาะสมแก่ความรู้ซึ่งได้รับจากรายวิชานี้

ซึ่งในเอกสารประกอบชิ้นงานนี้จะมีเนื้อหาเกี่ยวกับรายละเอียดในการออกแบบการทำงานของระบบดิจิทัลในการทำ Smart lock box โดยใช้ FPGA รวมถึง TTL ที่ใช้ในการจัดทำชิ้นงานนี้

ทั้งนี้ทางคณะผู้จัดทำหวังเป็นอย่างยิ่งว่าเอกสารนี้จะเป็นประโยชน์แก่ผู้ที่สนใจที่จะศึกษาเกี่ยวกับชิ้นงานที่คณะผู้จัดทำได้จัดทำขึ้นไม่มากนักน้อย หรือใช้เป็นแรงบันดาลใจไปต่อยอดในชิ้นงานอื่นๆได้

คณะผู้จัดทำ

## สารบัญ

บทที่ 1 ที่มา หลักการ และเหตุผล .....	4
ที่มาและความสำคัญ .....	4
วัตถุประสงค์ของชิ้นงาน .....	4
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง .....	5
บทที่ 3 การดำเนินการชิ้นงาน .....	11
รายละเอียดของชิ้นงาน .....	11
กระบวนการและเทคนิคการออกแบบ .....	11
วัสดุอุปกรณ์ที่ใช้ .....	21
ขั้นตอนการดำเนินงาน .....	21
บทที่ 4 ผลการดำเนินงาน.....	22
วิธีการใช้ชิ้นงาน .....	22
Source code : .....	22
บทที่ 5 สรุปผลการดำเนินงาน .....	32
สรุปผลการดำเนินงาน .....	32
ปัญหา อุปสรรค และการแก้ไข .....	32
แหล่งอ้างอิง.....	33

## บทที่ 1 ที่มา หลักการ และเหตุผล

### ที่มาและความสำคัญ

ปัจจุบันนี้ความเป็นส่วนตัวและความปลอดภัยเป็นเรื่องที่ใครๆก็ต้องการ หากเรานึกถึงที่เก็บของที่มีความเป็นส่วนตัวและปลอดภัย เราก็นึกถึงตู้เซฟกัน แต่ทว่าตู้เซฟปกตินั้นยากต่อการพกพาไปไหนมาไหน และลำบากเวลาที่เราจะเปิดตู้ด้วย ดังนั้น Smart Lock จึงเข้ามามีความสำคัญ ด้วยขนาดที่สามารถพกพาไปที่อื่นๆได้ รวมกับระบบป้องกันความปลอดภัย 2 ชั้น RFID และ Password

### วัตถุประสงค์ของชิ้นงาน

- 1.เพื่อให้ผู้ใช้ทำกิจกรรมระหว่างวันได้โดยไม่ต้องกังวลเรื่องสัมภาระ
- 2.เพื่อเพิ่มความสบายใจให้แก่ผู้ใช้ด้วยระบบล็อค 2 ชั้น ด้วยรหัสผ่านและบัตร RFID



การทำงานของ FPGAs จะยังมีความเร็วที่ด้อยกว่าapplication-specific integrated circuit (ASIC) , และเมื่อเปรียบเทียบขนาดทางกายภาพ พบว่าจะมีความหนาแน่นของวงจรที่น้อยกว่า รวมทั้งใช้กำลังงานมากกว่า ASIC อย่างไรก็ตาม FPGA มีข้อได้เปรียบตรงที่ใช้เวลาในการพัฒนาผลิตภัณฑ์ (time to market) ที่น้อยกว่า สามารถแก้ไขวงจรได้หลังจากที่ใช้งานจริงในภาคสนาม ,และมีค่าแรงในการดำเนินการที่ต่ำกว่า (non-recurring engineering) . นอกจากนี้ ยังมี FPGA ชนิดที่โปรแกรมได้ครั้งเดียว (OTP) ซึ่งมีราคาต่ำกว่าโดย FPGA ชนิดนี้เมื่อโปรแกรมแล้วจะคล้ายกับ ASIC นอกจากนี้ยังมีการรวมหน่วยความจำ config เข้าไว้ในอุปกรณ์ FPGA ซึ่งจะยังคงอยู่แม้ปิดไฟเลี้ยง เรียกว่า Complex programmable logic devices

### LCD 16x2



LCD หรือ Liquid Crystal Display คือหน้าจอแสดงผลตัวอักษร ตัวเลขหรืออักขระต่างๆ (Character LCD) รวมถึงบางรุ่นที่สามารถแสดงภาพกราฟิกได้ด้วย (Graphic LCD)

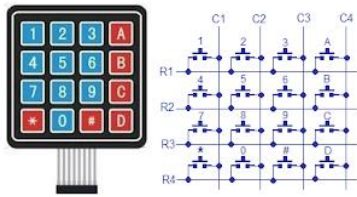
จอ LCD นี้จำเป็นมาก สำหรับงานที่ต้องการแสดงผลการทำงานต่างๆ เพื่อให้ผู้ใช้งานสามารถเชื่อมเป็นส่วนหนึ่งกับวงจรอิเล็กทรอนิกส์นั้นๆได้

### RFID



RFID ย่อมาจาก Radio Frequency Identification เป็นระบบระบุลักษณะของวัตถุด้วยคลื่นความถี่วิทยุ ที่ได้ถูกพัฒนามาตั้งแต่ปี ค.ศ. 1980 มีวัตถุประสงค์หลักเพื่อนำไปใช้งานแทนระบบบาร์โค้ด (Barcode) โดยจุดเด่นของ RFID อยู่กับการอ่านข้อมูลจากแท็ก (Tag) ได้หลายๆ แท็กแบบไร้สัมผัสและสามารถอ่านค่าได้แม้ในสภาพที่ทัศนวิสัยไม่ดี ทนต่อความเปียกชื้นแรงสั่นสะเทือน การกระแทกกระแทก สามารถอ่านข้อมูลได้ด้วยความเร็วสูง โดยข้อมูลจะถูกเก็บไว้ในไมโครชิปที่อยู่ในแท็ก

## Keypad



Keypad 4x4เป็นการจัดสวิตช์กดติดปล่อยดับจำนวน 16 ปุ่มในรูปแบบเมตริกซ์ 4 X 4 โดยปกติจะเป็นปุ่มสำหรับหมายเลข 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 และตัวอักษร A, B, C, D, \*, # Keypad 4x4จะมี 8 การเชื่อมต่อสายสัญญาณ คือ R1, R2, R3, R4 และ C1, C2, C3, C4 เป็นตัวแทนของแถวและคอลัมน์ตามลำดับ

## Servo



เซอร์โวมอเตอร์ (Servo Motor) เป็นมอเตอร์ที่มีการควบคุมการเคลื่อนที่ของมัน (State) ไม่ว่าจะเป็นระยะ ความเร็ว มุมการหมุน โดยใช้การควบคุมแบบป้อนกลับ (Feedback control) เป็นอุปกรณ์ที่สามารถควบคุมเครื่องจักรกล หรือระบบการทำงานนั้นๆ ให้เป็นไปตามความต้องการ เช่น ควบคุมความเร็ว (Speed), ควบคุมแรงบิด (Torque), ควบคุมแรงตำแหน่ง (Position), ระยะทางในการเคลื่อนที่(หมุน) (Position Control) ของตัวมอเตอร์ได้ ซึ่งมอเตอร์ทั่วไปไม่สามารถควบคุมในลักษณะงานเบื้องต้นได้ โดยให้ผลลัพธ์ตามความต้องการที่มีความแม่นยำสูง

## Arduino

Arduino เป็นชื่อเรียกของ platform micro controller ชนิดหนึ่งซึ่งก็ตามความหมายครับ micro=เล็ก, controller=ชุดควบคุม ดังนั้นหน้าที่ของมันก็คือชุดควบคุมขนาดเล็กที่สามารถนำไปเชื่อมต่อเพื่อสั่งการเครื่องมืออิเล็กทรอนิกส์ต่างๆได้นั่นเอง

สาเหตุหลักๆที่ทำให้ Arduino มีชื่อเสียงโด่งดัง อีกทั้งยังนับได้ว่าราคาย่อมเยามากกว่าบรรดา micro controller อื่นๆในตลาดก็คือการที่มันเป็น Open Source ทั้งในเชิง software และ hardware นั้นหมายความว่าใครๆก็สามารถนำระบบของ Arduino ไปดัดแปลงแก้ไขแล้วนำออกมาขายในตลาดได้โดยไม่ผิดกฎหมายครับ

## Arduino ในส่วนของ Software



Arduino ได้พัฒนาส่วนของระบบการเขียนโปรแกรม หรือที่เรียกว่า IDE (Integrated Development Environment) ซึ่งจะช่วยให้เราฝังคำสั่งลงในบอร์ด Arduino ชนิดต่างๆได้ ซึ่งในส่วนของ software นี้ เรายังสามารถดาวน์โหลด library เพิ่มได้จาก internet ซึ่งทำให้เราเขียนโปรแกรมกับ controller ชนิดอื่นๆ ไม่จำกัดอยู่แค่ Arduino อีกด้วย

## Arduino ในส่วนของ hardware

Arduino board แบ่งออกเป็นหลายชนิดด้วยกัน ซึ่งจะมีคุณลักษณะเหมาะสมกับการใช้งานไปคนละแบบ





## Maker Uno



บอร์ด Maker UNO จาก Cytron เป็นบอร์ดไมโครคอนโทรลเลอร์ ATmega328 ที่ถูกออกแบบมาให้ใช้งานง่าย ใช้ชิพ USB to serial CH340G พร้อมให้ on board LED เพิ่มมาถึง 12 ดวง และยังมี Push Button กับ Piezo Buzzer แบบโปรแกรมควบคุมได้ เสริมมาให้ เพื่อให้ใช้งานง่ายยิ่งขึ้น

### Technical Specifications

- SMD ATmega328P microcontroller
- USB Programming facilitated by the CH340.
- Input voltage: USB 5V, from computer, power bank or standard USB adapter.
- 500mA (maximum) 3.3V voltage regulator.
- 0-5V outputs with 3.3V compatible inputs.
- 14 Digital I/O Pins (6 PWM outputs).
- 6 Analog Inputs.
- ISP 6-pin Header.
- 32k Flash Memory.
- 16MHz Clock Speed.
- R3 Shield Compatible.
- LED array for 5V, 3.3V, TX, RX and all digital pins.
- On board programmable push button (pin 2, need to configure as INPUT\_PULLUP).

- On board piezo buzzer (pin 8).
- Utilize USB Micro-B socket.
- PURPLE PCB!

Dimension: 69.09mm(L) x 60.96(W)

### บทที่ 3 การดำเนินการชิ้นงาน

#### รายละเอียดของชิ้นงาน

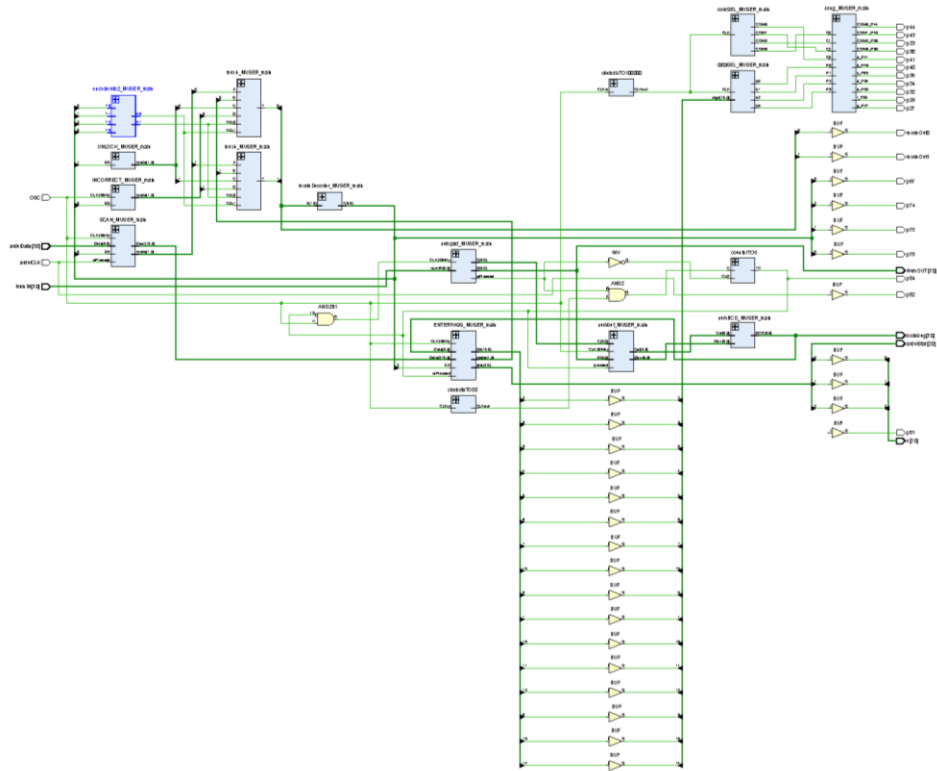
- 1) LCD แสดงสถานะของการทำงานตามการใช้งาน คือ Lock และ Unlock
- 2) สถานะการทำงานเปลี่ยน เมื่อมีการกระทำกับ Numpad หรือ แท้การ์ด RFID
- 3) มีเสียงเตือนดัง เมื่อมีการเปิดกล่อง Smart lock box อย่างผิดปกติ
- 4) หากแท้การ์ด RFID และป้อนรหัสผ่านถูกต้อง Smart lock box จะเปลี่ยนสถานะ และกล่องถูกเปิดออก

#### กระบวนการและเทคนิคการออกแบบ

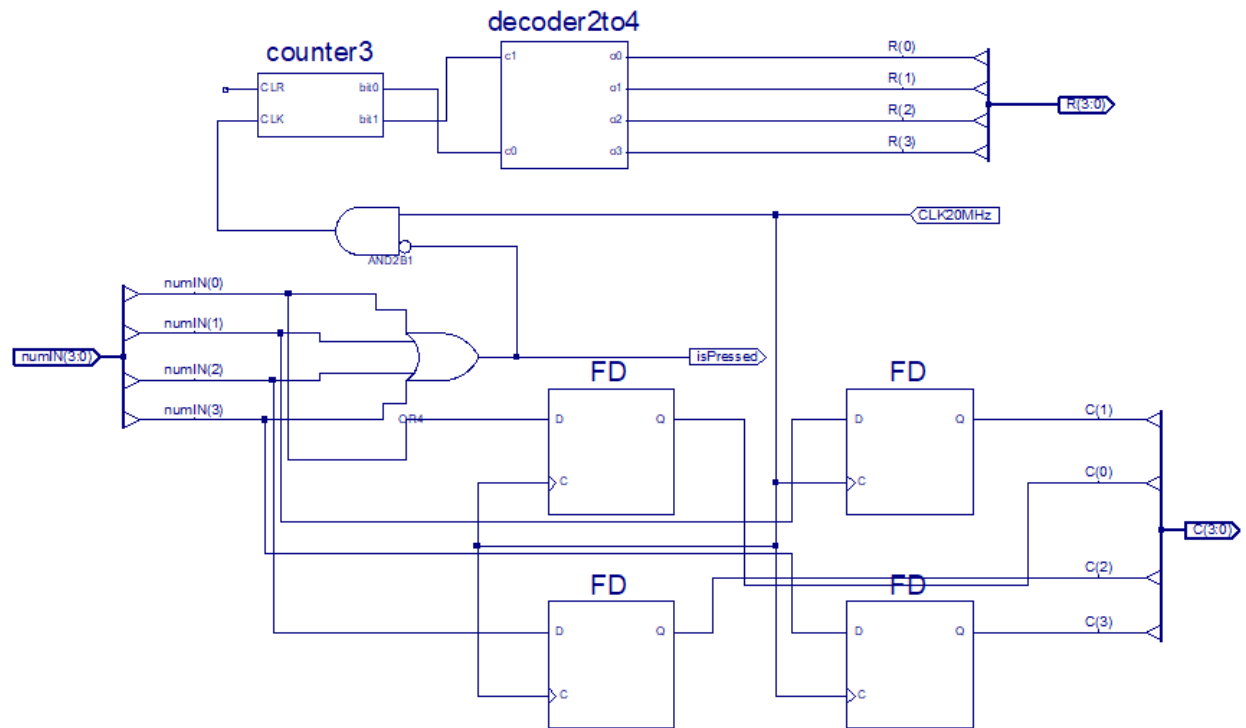
- 1) ใช้อุปกรณ์อ่าน RFID มารับค่าบัตร RFID เพื่อแสดงผลไปที่ LCD
- 2) ใช้ Magnet switch ในการรับค่าการ เปิด-ปิด ของกล่อง Smart lock box
- 3) ใช้ LCD ในงานแสดงสถานะของกล่อง Smart lock box
- 4) สร้างกล่อง Smart lock box จากกล่องไม้ โดยมีการเจาะเพื่อติดตั้งแผงป้อนข้อมูล
- 5) มีเสียงจาก Buzzer หากสถานะ Lock แต่มีการเปิดกล่องออก



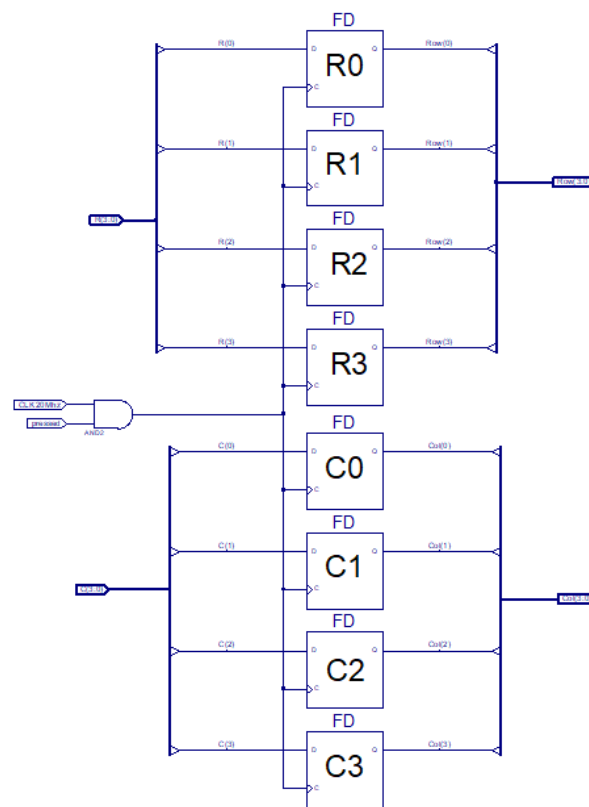
รูปที่ 1 Top-down ของวงจร



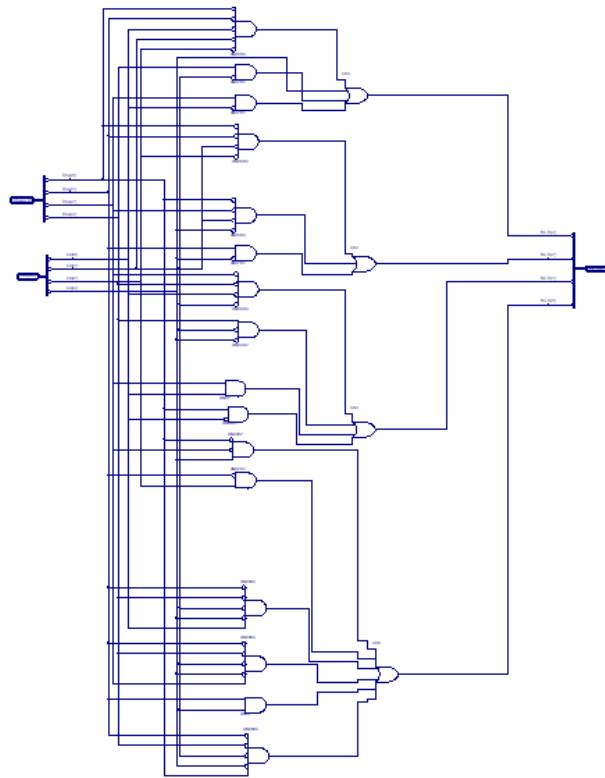
รูปที่ 2 แสดงแบบจำลองของวงจรทั้งหมด



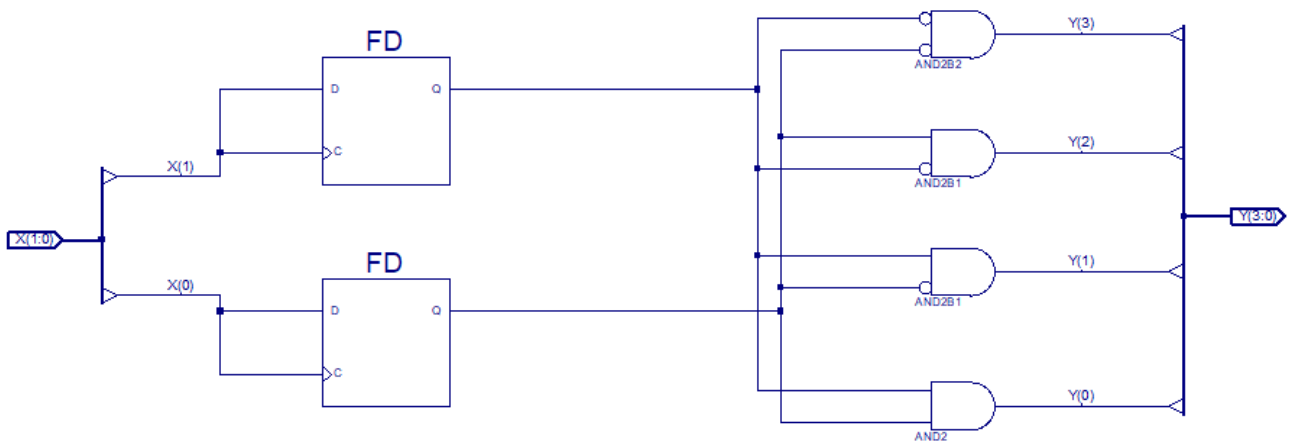
รูปที่ 3 แสดงวงจร Numpad



รูปที่ 4 แสดงวงจร Numbuf



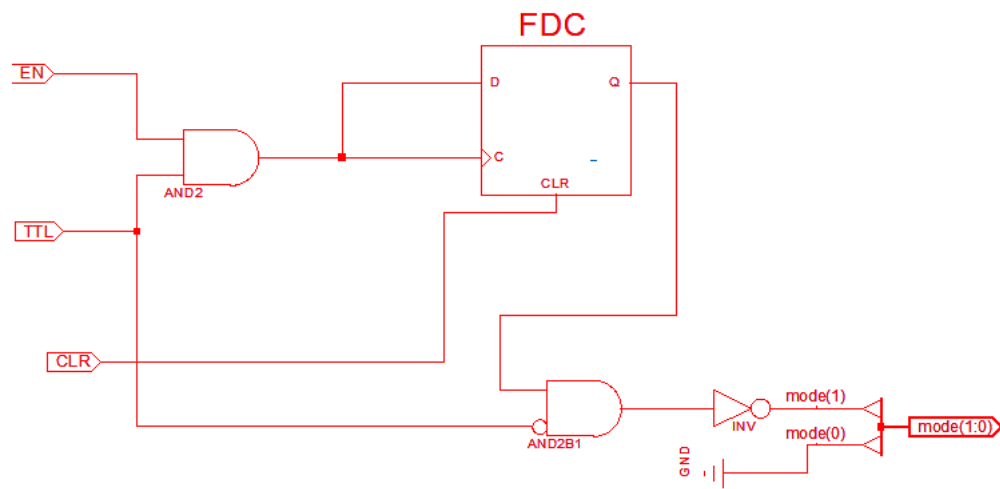
รูปที่ 5 แสดงวงจร NumBCD



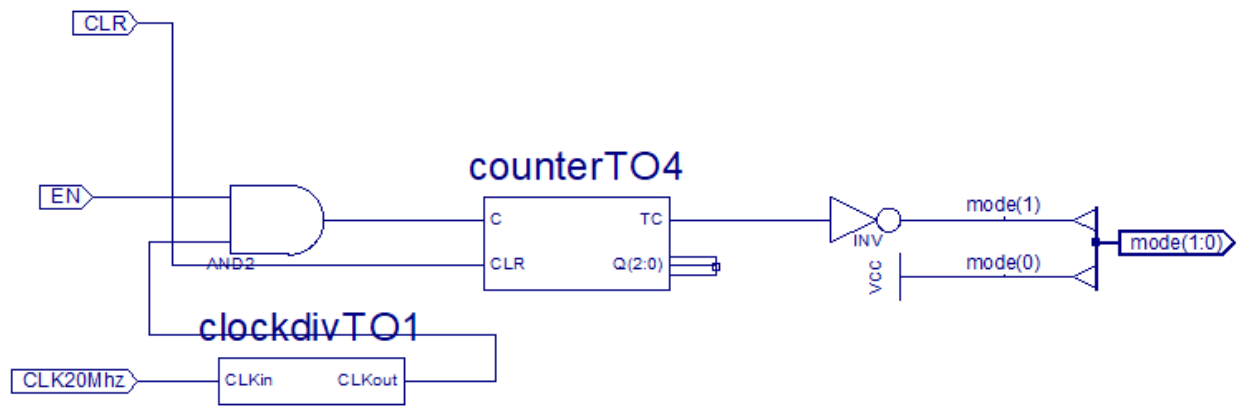
รูปที่ 6 แสดงวงจร ModeDecoder

รูปที่ 7 แสดงวงจร Scan

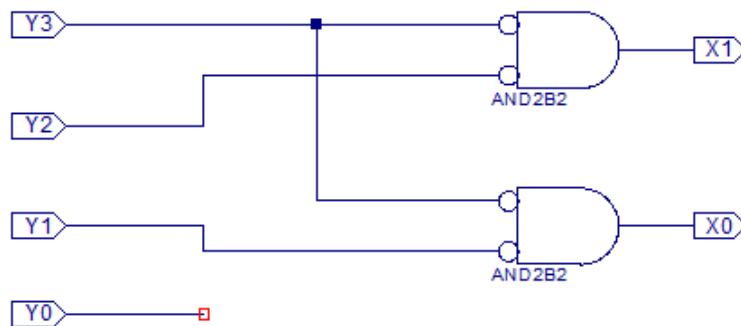
รูปที่ 8 แสดงวงจร Enterpass



รูปที่ 9 Unlock

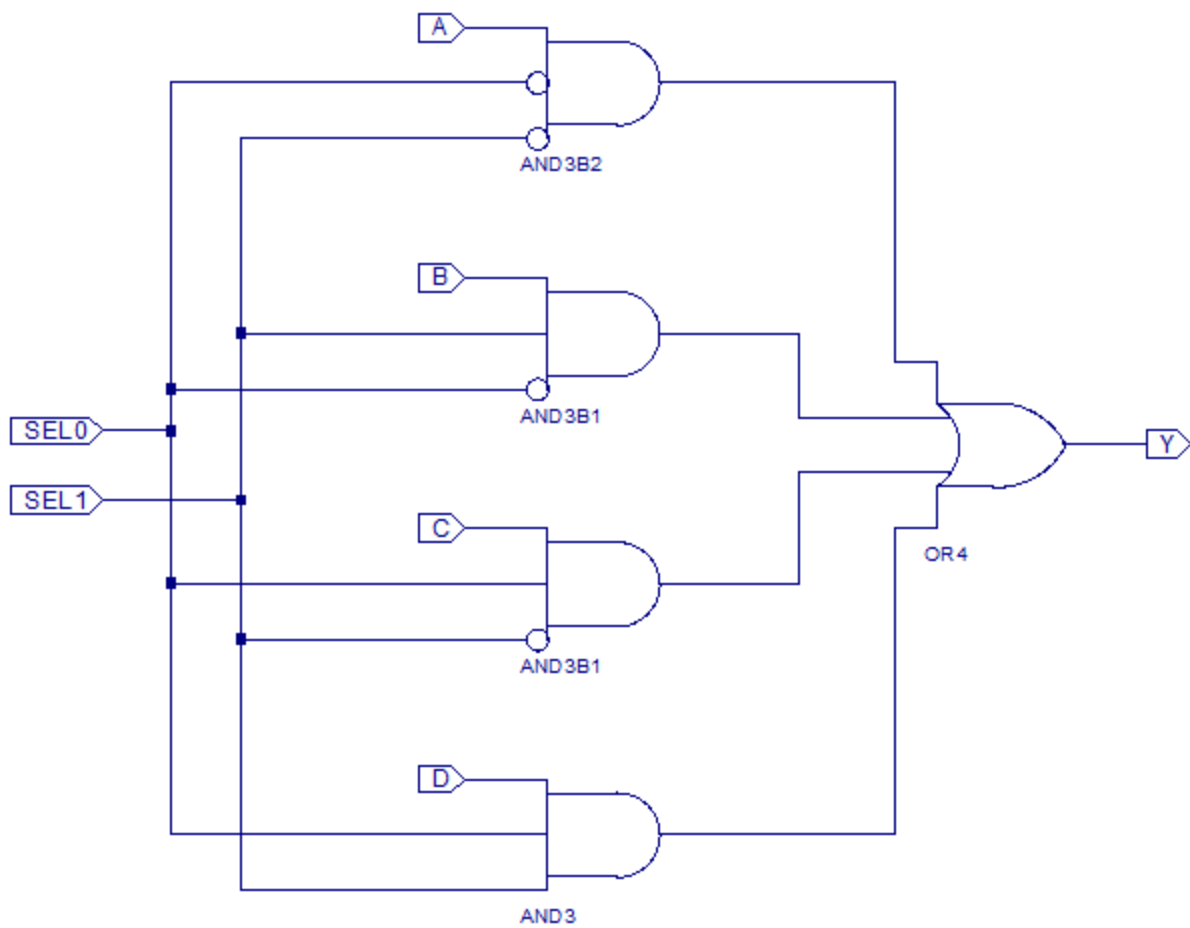


รูปที่ 10 แสดงวงจร Incorrect



รูปที่ 11 Encoder 4 to 2





รูปที่ 12 Mux

# PlanAhead Generated physical constraints

NET "arduiData[3]" LOC = P100;

NET "arduiData[2]" LOC = P101;

NET "arduiData[1]" LOC = P102;

NET "arduiData[0]" LOC = P104;

# PlanAhead Generated physical constraints

NET "arduiStar[2]" LOC = P105;

NET "arduiStar[1]" LOC = P111;

NET "arduiStar[0]" LOC = P112;

NET "NumIN[0]" LOC = P137;

NET "NumIN[1]" LOC = P133;

NET "NumIN[2]" LOC = P131;

NET "NumIN[3]" LOC = P126;

NET "NumOUT[0]" LOC = P2;

NET "NumOUT[1]" LOC = P143;

NET "NumOUT[2]" LOC = P141;

NET "NumOUT[3]" LOC = P139;

NET "OSC" LOC = P123;

NET "arduiCLK" LOC = P99;

# mode

NET "p67" LOC = P67;

NET "p74" LOC = P74;

NET "p75" LOC = P75;

NET "p78" LOC = P78;

#modeOut

NET "modeOut1" LOC = P114;

NET "modeOut0" LOC = P115;

NET "p82" LOC = P82; # arduick

NET "p81" LOC = P81; # enable

NET "p84" LOC = P84; # numpad

# star

NET "s[2]" LOC = P93;

NET "s[1]" LOC = P94;

NET "s[0]" LOC = P95;

# bcd

NET "bcdebug[3]" LOC = P85;

NET "bcdebug[2]" LOC = P87;

NET "bcdebug[1]" LOC = P88;

NET "bcdebug[0]" LOC = P92;

# sseg

NET "p41" LOC = P41;

NET "p40" LOC = P40;

NET "p35" LOC = P35;

NET "p34" LOC = P34;

NET "p32" LOC = P32;

NET "p29" LOC = P29;

NET "p27" LOC = P27;

NET "p44" LOC = P44;

NET "p43" LOC = P43;

NET "p33" LOC = P33;

NET "p30" LOC = P30;

### วัสดุอุปกรณ์ที่ใช้

- 1) บอร์ด FPGA รุ่น SERVEYOR-6 XCS6SLX9
- 2) Servo Motor 1 อัน
- 3) Arduino Maker Uno 2 อัน
- 4) Breadboard 1 อัน
- 5) LCD 16x2 1อัน
- 6) Keypad 4x4 1 อัน
- 7) Buzzer 1 อัน
- 8) กล้องไม้ 1 อัน
- 9) แผ่นอะคริลิค
- 10) Arduino RFID RC522A MFRC522 1 อัน

### ขั้นตอนการดำเนินงาน

- 1) ประชุมกลุ่มเพื่อหาหัวข้อชิ้นงานที่จะจัดทำ
- 2) จัดทำเอกสารข้อเสนอโครงการและนำเสนออาจารย์
- 3) ออกแบบระบบของชิ้นงานแบบ Top down Design
- 4) เขียนโปรแกรมตามที่ได้ออกแบบไว้
- 5) จัดทำชิ้นงาน
- 6) ทดสอบ ปรับปรุงแก้ไขชิ้นงาน
- 7) จัดทำรายงาน วิทยุทัศน์ พร้อมนำเสนออาจารย์

## บทที่ 4 ผลการดำเนินงาน

### วิธีการใช้ชิ้นงาน

เมื่อต้องการเปิด Smart lock box จะต้องนำ การ์ด RFID นำมาแตะที่แผงอ่าน RFID ในส่วนหน้าของ Smart lock box จากนั้นป้อนรหัสไปยัง Smart lock box ผ่าน Numpad จากนั้น อุปกรณ์ Smart lock box จะปลดล็อกออก และสามารถเปิดกล่องได้ หากนำของเข้าไปในกล่องแล้ว สามารถปิดฝาของ Smart lock box เพื่อให้อุปกรณ์ เข้าสู่โหมด Lock ดั้งเดิม

Source code :

Servo myservo;

SoftwareSerial sr(0, 1); // RX, TX

String state = "WAIT\_RFID";

int data[16];

int i = 0;

void setup() {

  sr.begin(9600);

  Serial.begin(9600);

  pinMode(CLK, OUTPUT);

  pinMode(D1, OUTPUT);

  pinMode(D2, OUTPUT);

  pinMode(D3, OUTPUT);

  pinMode(D4, OUTPUT);

  pinMode(STARb2, INPUT);

```
_pinMode(STARb1, INPUT);  
  
_pinMode(STARb0, INPUT);  
  
_pinMode(MODEb1, INPUT);  
  
_pinMode(MODEb0, INPUT);  
  
}  
  
void loop() {  
  
  if (state == "WAIT_RFID") {  
  
    myservo.write(0);  
  
    int count = 0;  
  
    int tmp1;  
  
    int tmp2;  
  
    while (1) {  
  
      if (sr.available()) {  
  
        if (count == 0) {  
  
          tmp1 = sr.read();  
  
          Serial.println(tmp1);  
  
          count++;  
  
        }  
  
      else if (count == 1) {
```

```
_____ tmp2 = sr.read();  
_____ Serial.println(tmp2);  
_____ break;  
_____ }  
_____ else {  
_____ int tmpFool = sr.read();  
_____ count++;  
_____ }  
_____ }  
_____ }  
_____ }  
_____ int i = 0;  
_____ int arr1[8] = {0, 0, 0, 0, 0, 0, 0, 0};  
_____ int arr2[8] = {0, 0, 0, 0, 0, 0, 0, 0};  
_____ while (tmp1 > 0) {  
_____ arr1[i] = tmp1 % 2;  
_____ tmp1 = tmp1 / 2;  
_____ i++;  
_____ }  
_____ i = 0;
```



```
____ while (tmp2 > 0) {  
____     arr2[i] = tmp2 % 2;  
____     tmp2 = tmp2 / 2;  
____     i++;  
____ }  
____ for (int i = 0; i < 8; i++ ) {  
____     data[15 - i] = arr1[i];  
____ }  
____ for (int i = 0; i < 8; i++ ) {  
____     data[7 - i] = arr2[i];  
____ }  
____ send2FPGA();  
____ for (int i = 0; i < 16; i++) {  
____     Serial.print(data[i]);  
____ }  
____ Serial.println();  
____ state = "WAIT_NUMPAD";  
____ }  
____ else if (state == "WAIT_NUMPAD") {
```

```
____myservo.write(0);

____int prev[3] = {0, 0, 0};

____while (1) {

____    int tmp[3];

____    tmp[0] = digitalRead(STARb2);

____    tmp[1] = digitalRead(STARb1);

____    tmp[2] = digitalRead(STARb0);

____    //    Serial.print(tmp[0]);

____    //    Serial.print(tmp[1]);

____    //    Serial.println(tmp[2]);

____    if (tmp[0] == 0 and tmp[1] == 0 and tmp[2] == 0 and (prev[0] != tmp[0] or prev[1]
!= tmp[1] or prev[2] != tmp[2])) {

____        //        sr.print('0');

____        Serial.println("Nothing");

____    }

____    else if (tmp[0] == 0 and tmp[1] == 0 and tmp[2] == 1 and (prev[0] != tmp[0] or
prev[1] != tmp[1] or prev[2] != tmp[2])) {

____        sr.print('a');

____        Serial.println('a');

____    }
```

```
_____ else if (tmp[0] == 0 and tmp[1] == 1 and tmp[2] == 0 and (prev[0] != tmp[0] or  
prev[1] != tmp[1] or prev[2] != tmp[2])) {
```

```
_____ sr.print('b');
```

```
_____ Serial.println('b');
```

```
_____ }
```

```
_____ else if (tmp[0] == 0 and tmp[1] == 1 and tmp[2] == 1 and (prev[0] != tmp[0] or  
prev[1] != tmp[1] or prev[2] != tmp[2])) {
```

```
_____ sr.print('c');
```

```
_____ Serial.println('c');
```

```
_____ }
```

```
_____ else if (tmp[0] == 1 and tmp[1] == 0 and tmp[2] == 0 and (prev[0] != tmp[0] or  
prev[1] != tmp[1] or prev[2] != tmp[2])) {
```

```
_____ sr.print('d');
```

```
_____ Serial.println('d');
```

```
_____ }
```

```
_____ else if (tmp[0] == 1 and tmp[1] == 0 and tmp[2] == 1 and (prev[0] != tmp[0] or  
prev[1] != tmp[1] or prev[2] != tmp[2])) {
```

```
_____ while (1) {
```

```
_____ int tmp2[2];
```

```
_____ tmp2[0] = digitalRead(MODEb1);
```

```
____ tmp2[1] = digitalRead(MODEb0);

____ if (tmp2[0] == 1 && tmp2[1] == 0) {

____     state = "UNLOCK_BOX";

____     sr.print('U');

____     Serial.println('U');

____     break;

____ }

____ else if (tmp2[0] == 1 && tmp2[1] == 1) {

____     state = "INCORRECT";

____     sr.print('I');

____     Serial.println('I');

____     break;

____ }

____ }

____ break;

____ }

____ prev[0] = tmp[0];

____ prev[1] = tmp[1];

____ prev[2] = tmp[2];
```

```
____}  
  
____}  
  
____else if (state == "UNLOCK_BOX") {  
  
____int pos = 45;  
  
____myservo.write(pos);  
  
____while (1) {  
  
____int tmp[2];  
  
____tmp[0] = digitalRead(MODEb1);  
  
____tmp[1] = digitalRead(MODEb0);  
  
____if (tmp[0] == 0 && tmp[1] == 0) {  
  
____state = "WAIT_RFID";  
  
____sr.print("S");  
  
____break;  
  
____}  
  
____}  
  
____}  
  
____else if (state == "INCORRECT") {  
  
____myservo.write(0);  
  
____while (1) {
```

```
int tmp[2];

tmp[0] = digitalRead(MODEb1);

tmp[1] = digitalRead(MODEb0);

if (tmp[0] == 0 && tmp[1] == 0) {

    state = "WAIT_RFID";

    sr.print('S');

    break;

}

}

}

}
```

```
void send2FPGA() {

    for (int i = 0; i < 16; i++) {

        digitalWrite(D1, data[i++]);

        digitalWrite(D2, data[i++]);

        digitalWrite(D3, data[i++]);

        digitalWrite(D4, data[i++]);

        digitalWrite(CLK, HIGH);

    }

}
```

```
____//Serial.print("HIGH");  
____delayMicroseconds(1);  
____digitalWrite(CLK, LOW);  
____//Serial.print("LOW");  
____delayMicroseconds(1);  
____}  
____}
```

## บทที่ 5 สรุปผลการดำเนินงาน

### สรุปผลการดำเนินงาน

จากการดำเนินการทำชิ้นงานและทดสอบแล้ว พบว่า Smart lock box ที่กลุ่มข้าพเจ้าจัดทำขึ้น ไม่สามารถทำงานได้ตามที่ตั้งเป้าหมายไว้ได้จริง กล่าวคือ ทำได้เพียงอ่านข้อมูลจาก RFID และอ่านข้อมูลจาก Keypad แต่ไม่สามารถเปรียบเทียบข้อมูลทั้งสองได้อย่างถูกต้อง อีกทั้งตัวอุปกรณ์ยังไม่เสถียรเท่าที่ควร



Link video ชิ้นงาน : <https://youtu.be/HEuca6TCslQ>

### ปัญหา อุปสรรค และการแก้ไข

- 1) สายไฟมีจำนวนเยอะ และพันกันยุ่งเหยิง  
วิธีแก้ จัดเรียงสายไฟให้เป็นระเบียบ มัดส่วนที่รวมกันได้
- 2) อุปกรณ์พัง เสียหาย  
วิธีแก้ ซื้อเผื่อไว้ก่อน และก่อนใช้งานต้องตรวจสอบให้ดีกว่าต่อถูกไหม



## แหล่งอ้างอิง

“FPGA” [ออนไลน์]. เข้าถึงได้จาก :

<https://th.wikipedia.org/wiki/เอฟพีจีเอ>

“RFID” [ออนไลน์]. เข้าถึงได้จาก :

<https://th.wikipedia.org/wiki/อาร์เอฟไอดี>

“LCD 16x2” [ออนไลน์]. เข้าถึงได้จาก :

<https://www.thaieasyelec.com/article-wiki/review-product-article/how-to-use-character-lcd-display-arduino-ch1-parallel-version.html>

“Keypad 4x4” [ออนไลน์]. เข้าถึงได้จาก :

<http://elec2web.blogspot.com/2016/04/arduino-uno-keypad-4x4.html>

“Servo motor” [ออนไลน์]. เข้าถึงได้จาก :

[http://www.advance-electronic.com/blog/detail/86/th/เซอร์โวมอเตอร์-\(Servo-Motor\).html](http://www.advance-electronic.com/blog/detail/86/th/เซอร์โวมอเตอร์-(Servo-Motor).html)

“Arduino 4x4” [ออนไลน์]. เข้าถึงได้จาก :

<https://thaiarduino.club/what-is-arduino/>

“Maker Uno” [ออนไลน์]. เข้าถึงได้จาก :

<https://www.arduitronics.com/product/2572/cytron-maker-uno>