

Arduino Chat project

Mini Project proposal submission 1

010123119 Embedded System Design

รายชื่อสมาชิก

- นายกิตติศักดิ์ เชี่ยวเชิงชล 54-010126-3002-9
- นายคณิตกร จันเชื้อ 54-010126-3005-3
- นายสรวิศ ทองอินทร์ 54-010126-3027-4

รายการงานย่อย

work breakdown structure (WBS)

- 1. ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard
- 2. รับข้อมูลจาก PS2 keyboard มาแสดงบน LCD
- 3. รับข้อมูลจาก PS2 keyboard เข้า Arduino Uno
- 4. ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link
- 5. รับข้อมูลจาก PS2 keyboard โดยส่งผ่าน wireless link มาแสดงบน PC

ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard

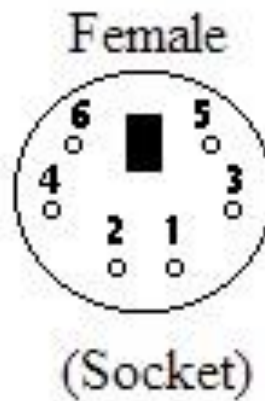
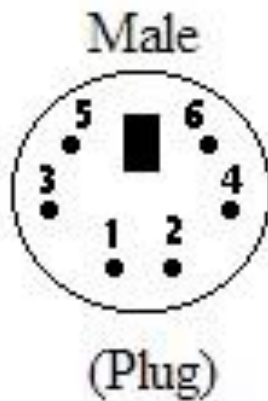
- จากการศึกษาข้อมูลการใช้งาน PS2 Keyboard ต้องใช้ DIN CONNECTOR ในการเชื่อมต่อกับ PS2 Keyboard



รูปอุปกรณ์ DIN Connector

ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- โดยรายละเอียดต่างๆของ **DIN CONNECTOR** มีดังนี้

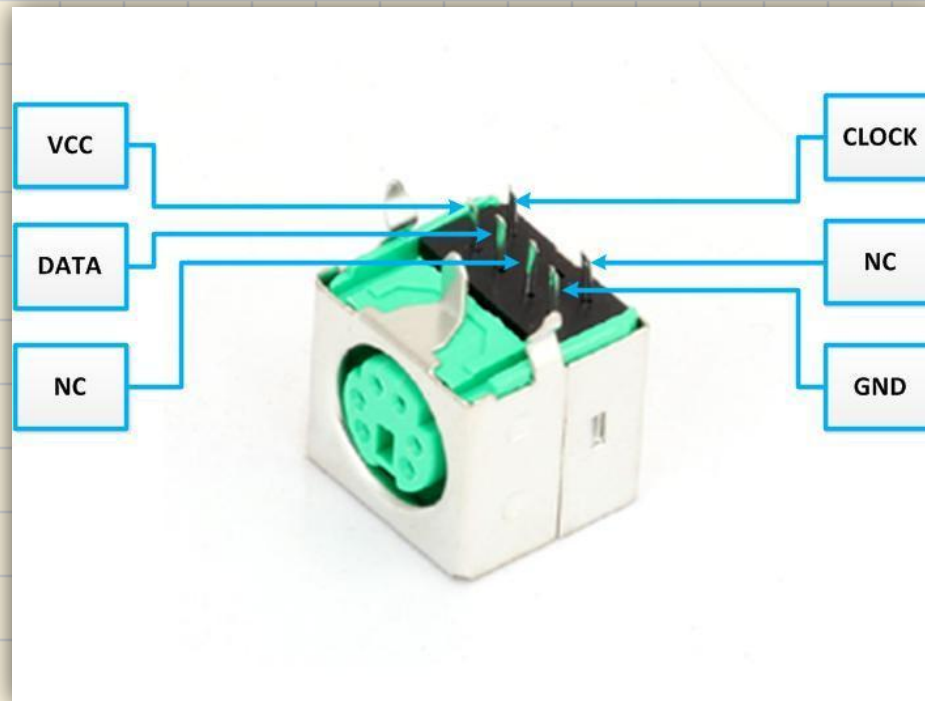


6-pin Mini-DIN (PS/2):

- 1 - Data
- 2 - Not Implemented
- 3 - Ground
- 4 - Vcc (+5V)
- 5 - Clock
- 6 - Not Implemented

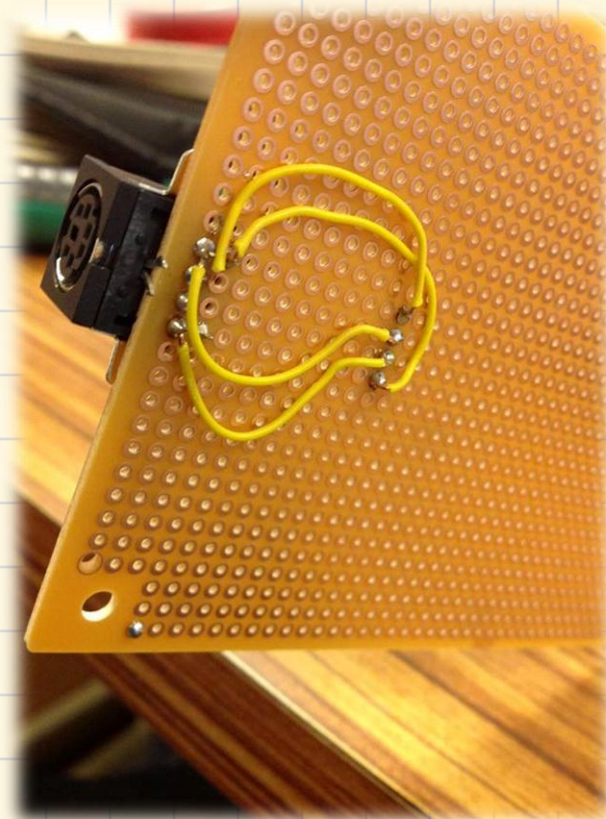
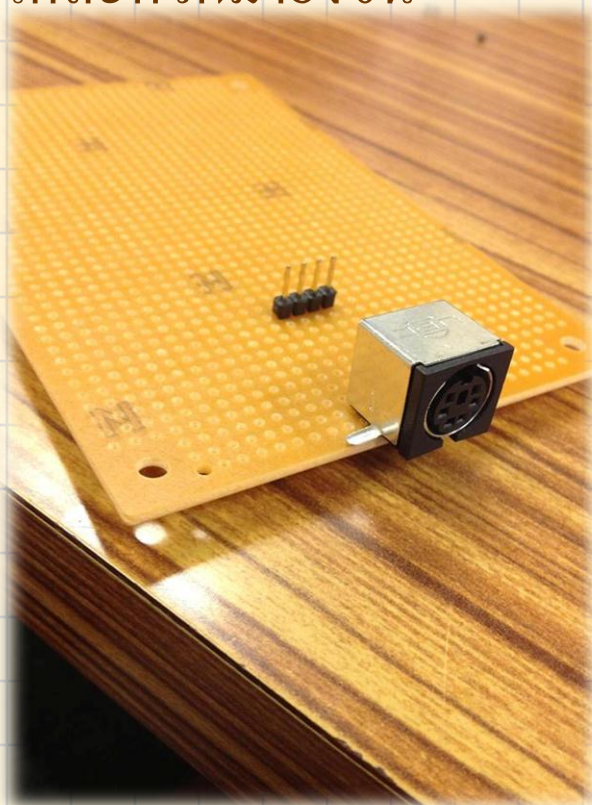
ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- ภาพอธิบายขาต่างๆจากอุปกรณ์จริง



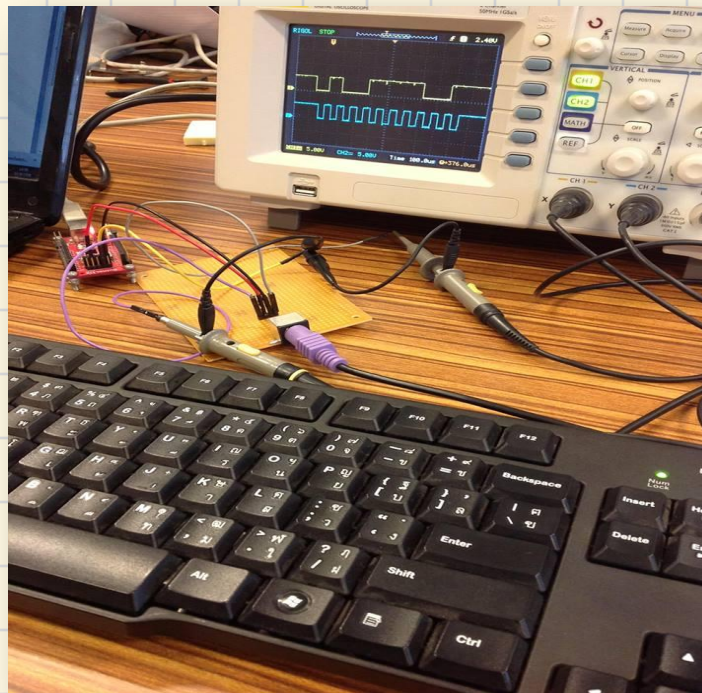
ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- ทางทีมงานได้ดำเนินการเชื่อมต่ออุปกรณ์ลง PCB เพื่อนำมาใช้งานได้สะดวกมากยิ่งขึ้น



ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- ต่อ oscilloscope เพื่อตรวจสอบลักษณะการส่งสัญญาณของ keyboard(PS2)



ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)



โดย CH1(สีเหลือง) คือ สัญญาณของพอร์ต DATA
และ CH2(สีฟ้า) คือ สัญญาณของพอร์ต CLK

ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- โดยในการเชื่อมต่อ PS2 keyboard กับ Arduino board นั้นทาง **www.arduino.cc** มี Library ของ PS2Keyboard ซึ่งสามารถศึกษาได้จาก <http://playground.arduino.cc/Main/PS2Keyboard>
- โดยการใช้งาน Library นั้นต้องทำการ include ก่อน
`#include <PS2Keyboard.h>`

ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- โดยในการ Library นั้นจะรับ input 2 ตัวจาก PS2 keyboard คือ Data และ Clock
- จากโค้ดที่นำมาใช้ไม่สามารถใช้ pin PWM อื่นในการส่ง CLK ได้ ใช้ได้เพียง pin 3(PWM) เท่านั้น

```
#include <PS2Keyboard.h>

const int DataPin = 8;
const int IRQpin = 3;

PS2Keyboard keyboard;

void setup() {
  delay(1000);
  keyboard.begin(DataPin, IRQpin);
  Serial.begin(9600);
  Serial.println("Keyboard Test:");
}
```


ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- จากโค้ดเดิมจะยังไม่สามารถใช้ปุ่ม F1-F12 หรือ ปุ่มอื่นๆได้
ต้องเพิ่ม **else if** ตรวจสอบปุ่มที่จะใช้นั้นๆแล้วต้องทำให้ปุ่มนั้น
enable ด้วยการไปแก้ไขไฟล์
library PS2Keyboard.h และกำหนดเลขเฉพาะตัวของ
แต่ละปุ่มซึ่งจะต้องไม่ซ้ำกับปุ่มอื่น
- เพิ่ม **else if** เพื่อตรวจสอบปุ่ม F1-F1

ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- โค้ดที่ทำการเพิ่มปุ่ม
F1-F12

```
} else if (c == PS2_DELETE) {  
    Serial.print("[Del]");  
}  
} else if (c == PS2_F1) {  
    Serial.print("[F1]");  
}  
} else if (c == PS2_F2) {  
    Serial.print("[F2]");  
}  
} else if (c == PS2_F3) {  
    Serial.print("[F3]");  
}  
} else if (c == PS2_F4) {  
    Serial.print("[F4]");  
}  
} else if (c == PS2_F5) {  
    Serial.print("[F5]");  
}  
} else if (c == PS2_F6) {  
    Serial.print("[F6]");  
}  
} else if (c == PS2_F7) {  
    Serial.print("[F7]");  
}  
} else if (c == PS2_F8) {  
    Serial.print("[F8]");  
}  
} else if (c == PS2_F9) {  
    Serial.print("[F9]");  
}  
} else if (c == PS2_F10) {  
    Serial.print("[F10]");  
}  
} else if (c == PS2_F11) {  
    Serial.print("[F11]");  
}  
} else if (c == PS2_F12) {  
    Serial.print("[F12]");  
}  
} else {
```

ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

```
#ifndef PS2Keyboard_h
#define PS2Keyboard_h

#if defined(ARDUINO) && ARDUINO >= 100
#include "Arduino.h" // for attachInterrupt, FALLING
#else
#include "WProgram.h"
#endif

#include "utility/int_pins.h"

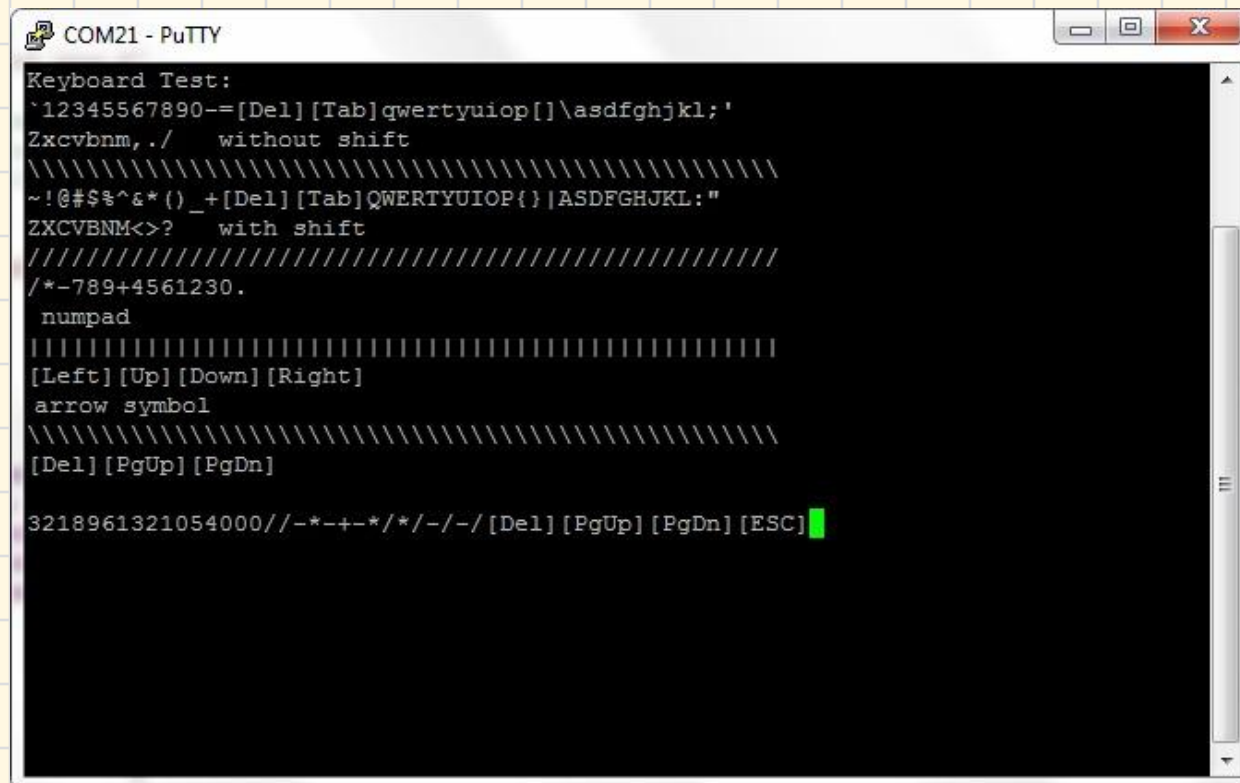
// Every call to read() returns a single byte for each
// keystroke. These configure what byte will be returned
// for each "special" key. To ignore a key, use zero.
#define PS2_TAB 9
#define PS2_ENTER 13
#define PS2_BACKSPACE 127
#define PS2_ESC 27
#define PS2_INSERT 11
#define PS2_DELETE 127
#define PS2_HOME 3
#define PS2_END 2

#define PS2_PAGEUP 25
#define PS2_PAGEDOWN 26
#define PS2_UPARROW 11
#define PS2_LEFTARROW 8
#define PS2_DOWNARROW 10
#define PS2_RIGHTARROW 21
#define PS2_F1 40
#define PS2_F2 41
#define PS2_F3 42
#define PS2_F4 43
#define PS2_F5 44
#define PS2_F6 45
#define PS2_F7 46
#define PS2_F8 47
#define PS2_F9 48
#define PS2_F10 49
#define PS2_F11 50
#define PS2_F12 51
#define PS2_SCROLL 52
```

- enable ด้วยการไปแก้ไข
ไฟล์
library PS2Keyboar
d.h
และกำหนดเลขเฉพาะตัวของ
แต่ละปุ่ม

ศึกษาและทดลองรับข้อมูลจาก PS2 keyboard (ต่อ)

- เมื่อทำ 2 ขั้นตอนข้างต้นเรียบร้อยแล้วจะสามารถใช้ปุ่ม F1-F12 ได้



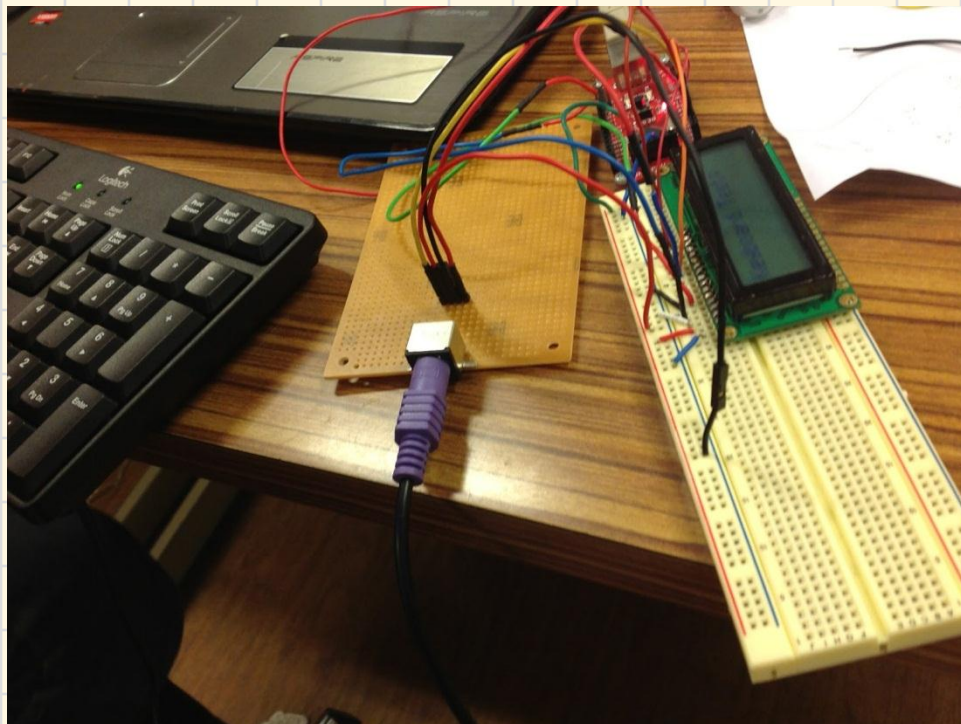
```
COM21 - PuTTY
Keyboard Test:
`12345567890--[Del] [Tab]qwertyuiop[]\asdfghjkl;'
Zxcvbnm,./  without shift
////////////////////////////////////
~!@#$%^&*()_+[Del] [Tab]QWERTYUIOP{}|ASDFGHJKL:"
ZXCVBNM<>?  with shift
////////////////////////////////////
/*-789+4561230.
  numpad
|||||
[Left] [Up] [Down] [Right]
  arrow symbol
////////////////////////////////////
[Del] [PgUp] [PgDn]

3218961321054000//-*+--*//*/-/-/[Del] [PgUp] [PgDn] [ESC]
```

ภาพการส่งข้อมูลจาก keyboard

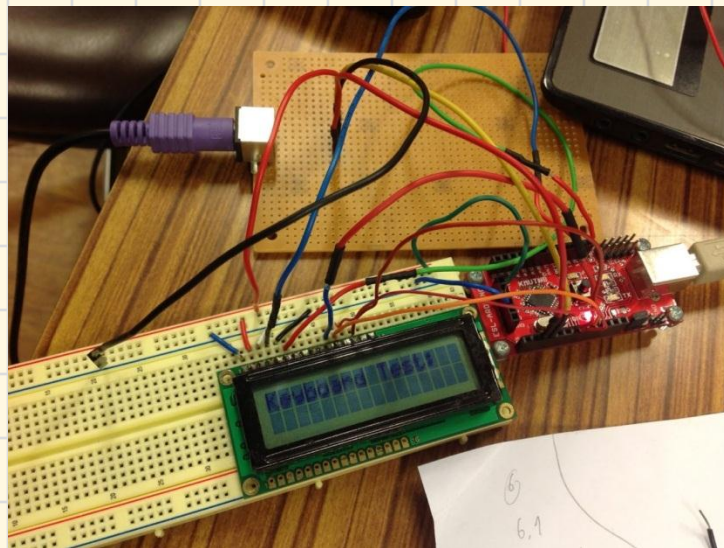
รับข้อมูลจาก PS2 keyboard มาแสดงบน LCD

- นำข้อความจาก keyboard มาแสดงผลใน LCD 16x2
 - เริ่มจากการต่อ LCD และ keyboard กับ Arduino



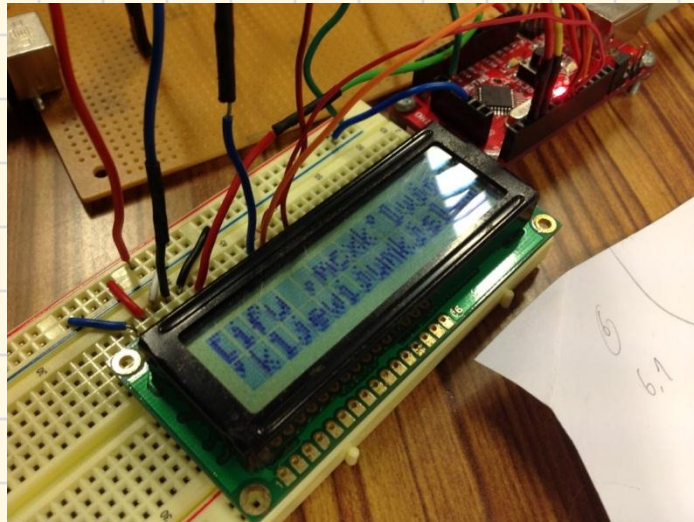
รับข้อมูลจาก PS2 keyboard มาแสดงบน LCD (ต่อ)

- อัฟโหลดโปรแกรมลงบอร์ด เมื่ออัฟโหลดเสร็จ LCD จะแสดงข้อความที่ได้จาก keyboard
- โดยเริ่มแรกจะมีข้อความเพื่อทดสอบการพิมพ์คือ “keyboard test”



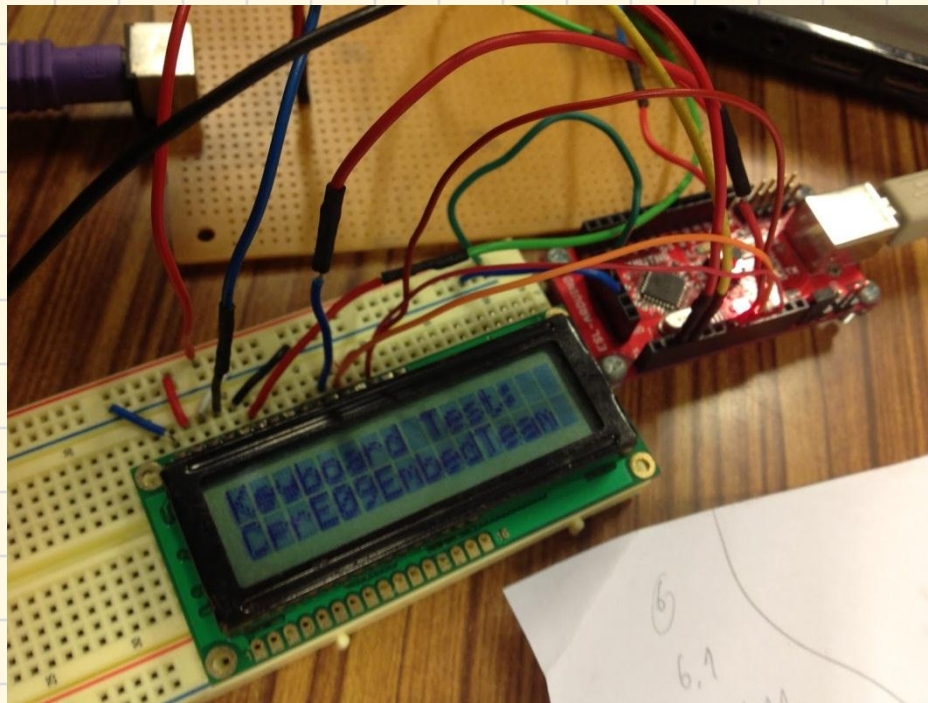
รับข้อมูลจาก PS2 keyboard มาแสดงบน LCD (ต่อ)

- โดยเมื่อพิมพ์ keyboard จะมาแสดงผลที่บรรทัดที่2ของ LCD แต่เมื่อเกินขนาดของบรรทัด ตัวอักษรจะไปปรากฏที่บรรทัดที่1 แทน และเมื่อพิมพ์ไปได้ระยะหนึ่งเช่นกันตัวอักษรจะไปแสดงผลบรรทัดที่ 2 วนอย่างนี้ไปเรื่อยๆ



รับข้อมูลจาก PS2 keyboard มาแสดงบน LCD (ต่อ)

- รูปแสดงการพิมพ์ PS2 keyboard แสดงบน LCD

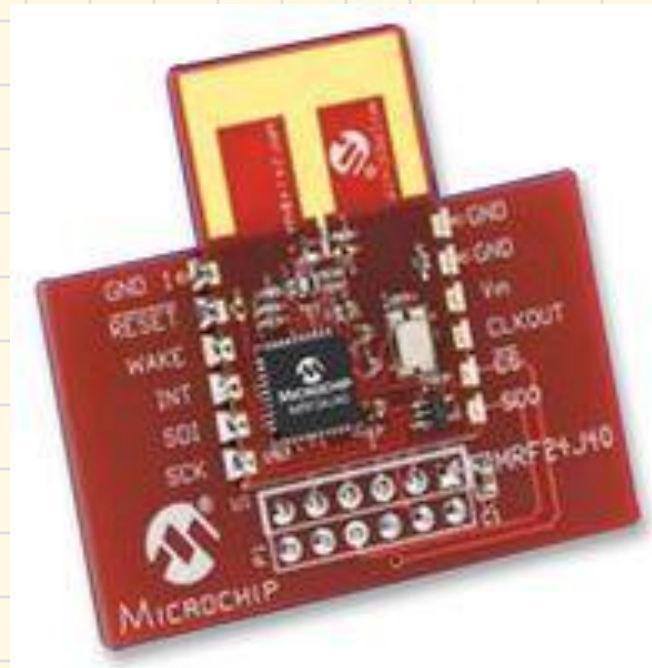


รับข้อมูลจาก PS2 keyboard มาแสดงบน LCD (ต่อ)

- เมื่อสามารถแสดงข้อมูลบน LCD ได้แล้ว เพื่อความสะดวกในการใช้งาน LCD ทางทีมงานจึงเพิ่มฟีเจอร์การแสดงผลบน LCD เพิ่มเติม
 - สามารถลบตัวอักษรได้จากครั้งแรกแค่ตรวจได้ว่ากดปุ่ม Del เหยๆ
 - เพิ่ม cursor ระบุตำแหน่ง
 - มี scroll เลื่อนเมื่อข้อความเกิน 16 ช่อง

ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

- Wireless module ที่ใช้ในการส่งข้อมูลใช้ module MRF24J40MA/MB



ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

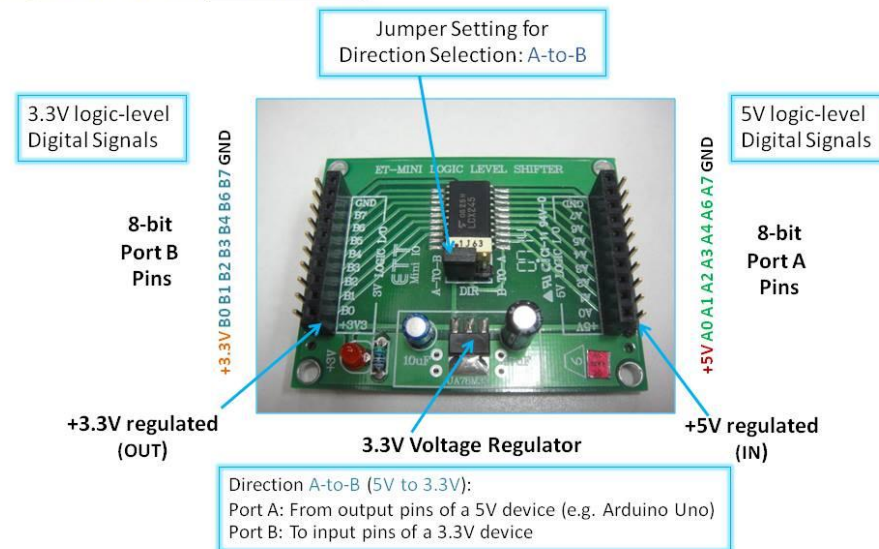
- MRF24J40MA/MB มีขาที่ใช้เชื่อมต่อทั้งหมด 12 ขา
- มีรายละเอียดดังนี้

MRF24J40MA/MB	
GND	GND
RS	GND
WAKE	3.3V
INT	CLKOUT
SDI	CS
SCK	SDO

ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

- เนื่องจากโมดูล MRF24J40MA/MB ใช้ไฟเลี้ยงที่
ประมาณ 3.3 V แต่ไฟเลี้ยงที่ใช้ในการต่อวงจรใช้ไฟเลี้ยงที่
5V จึงจำเป็นต้องใช้ Logic Level Shifter เพื่อแปลง
ไฟเลี้ยง

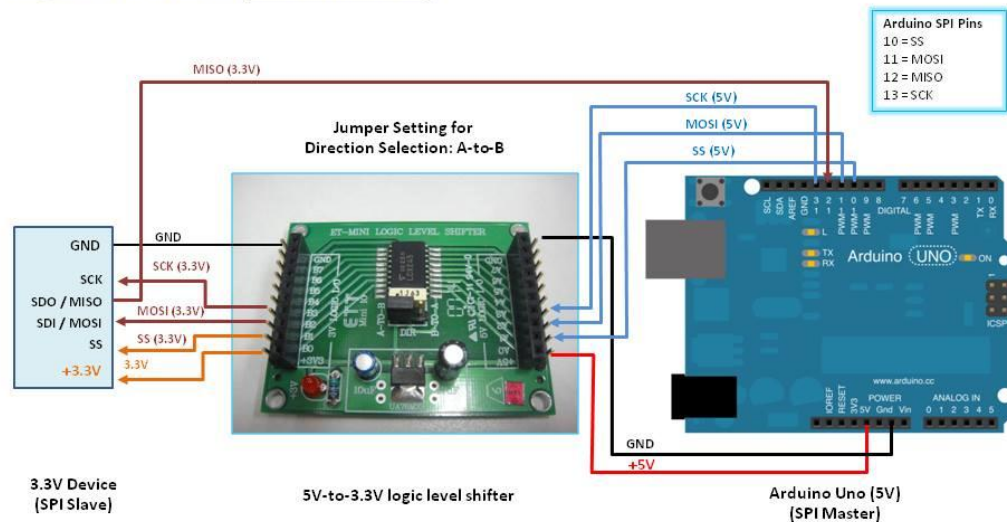
ET-Mini Logic Level Shifter
Logic Level Shifter (Unidirectional)



ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

- ภาพวงจรการต่อโมดูล MRF24J40MA/MB , Logic Level Shifter และ Arduino เข้าด้วยกัน

ET-Mini Logic Level Shifter
Logic Level Shifter (Unidirectional)



ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

- ในการใช้งาน MRF24J40MA/MB นั้นมี Library สำเร็จรูปที่สามารถเชื่อมต่อและส่งข้อมูลได้เลย นั่นคือ mrf24j.h สามารถศึกษาและ download ได้ที่ <https://github.com/karlp/Mrf24j40-arduino-library>
- โดย MRF24J40MA/MB นั้นส่งข้อมูลในรูปแบบ SPI

ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

อธิบายโค้ดพอสังเขป

- include Library
 - #include <SPI.h>
 - #include <mrf24j.h>
- กำหนดขา input port แต่ละขา
 - const int pin_reset = 6;
 - const int pin_cs = 10;
 - const int pin_interrupt = 2;

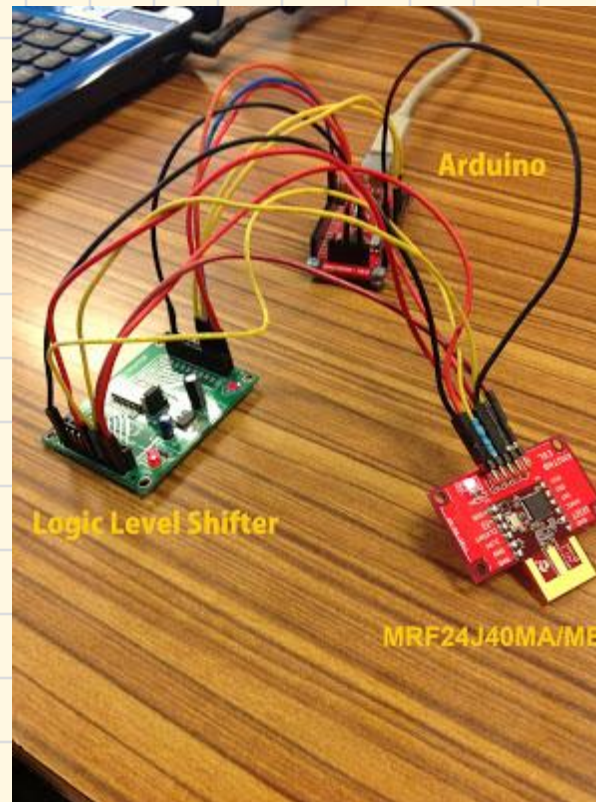
ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

อธิบายโค้ดพอสังเขป

- กำหนดวงแลน
 - `mrf.set_pan(0xcafe);`
- กำหนด IP Address
 - `mrf.address16_write(0x6001);`
- ฟังก์ชันที่ใช้ส่งข้อมูล
 - `mrf.send16(0x4202, "abcd");`
- ฟังก์ชันที่ใช้อ่านข้อมูล
 - `handle_rx()`

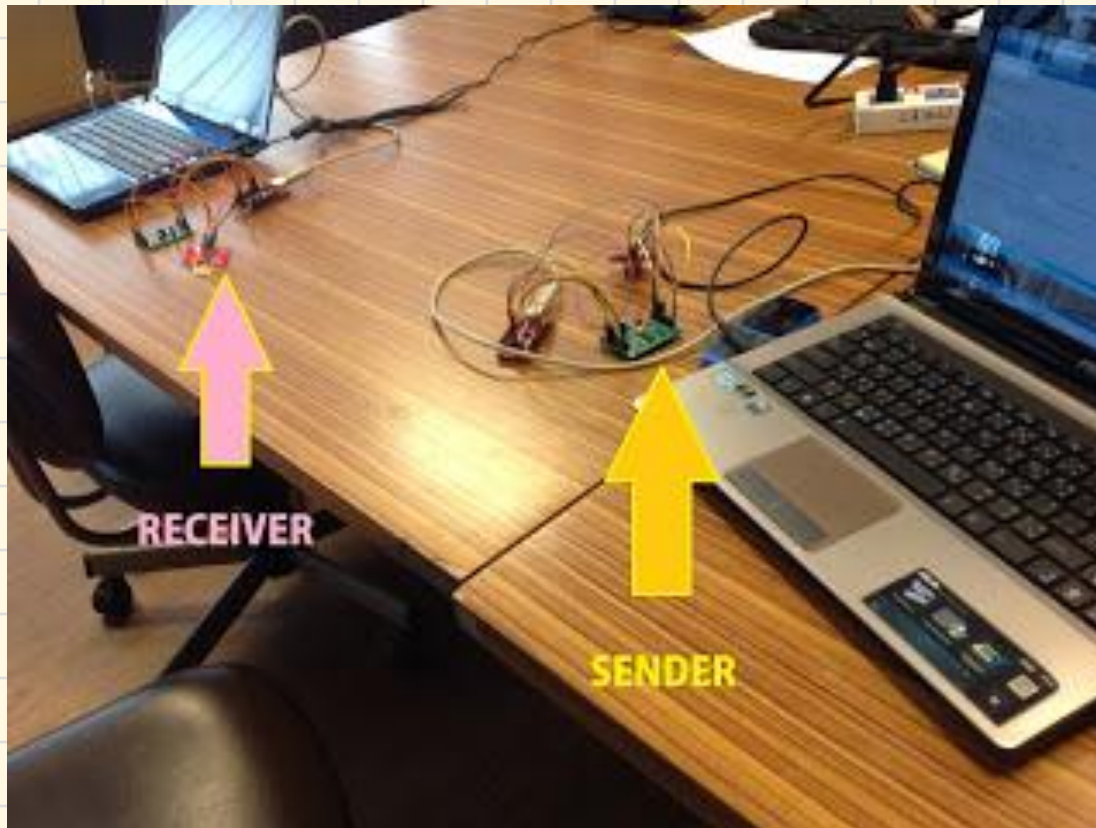
ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

- เมื่อต่อตามรูปร่างจะได้อุปกรณ์รับส่งไร้สาย 1 ชุด



ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

- รูปทดลองส่งวงจจริง

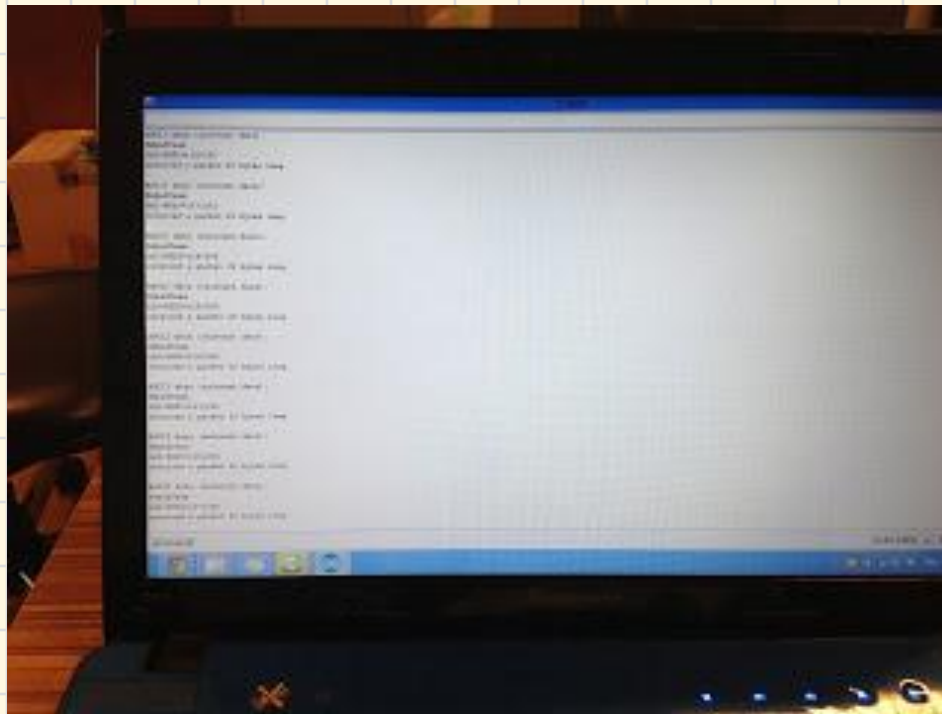


Protocol

- ❑ Frame control 2 byte
- ❑ sequence number 1 byte
- ❑ panid 2 byte
- ❑ shortAddr Destination 2 byte
- ❑ shortAddr Soucre 2 byte
- ❑ FCS 2 byte
- ❑ DATA 116 byte

ศึกษาและทดลองใช้ wireless module ที่เลือกใช้ ส่งและรับข้อมูลผ่าน wireless link

- OUTPUT จาก RECEIVER ที่ได้รับจาก SENDER
ซึ่งข้อมูลที่ส่งก็คือ EmbedTeam



สรุปความคืบหน้าProject

- สามารถรับข้อมูลจาก PS2 Keyboard เข้า arduino board ได้
- สามารถนำจาก PS2 Keyboard มาแสดงบน LCD ได้
- สามารถส่ง และ รับข้อมูลด้วย module wireless รุ่น MRF24J40MA/MB ได้

ส่วนที่เหลือ

- เพิ่มพีเจอร์ให้กับจอ **LCD** เพื่อความสะดวกในการใช้งานมากขึ้น
- นำข้อมูลที่ได้จาก **PS2 Keyboard** มาส่งผ่าน **module wireless MRF24J40MA/MB**