

Upboard Lab04

tags: Upboard Embedded

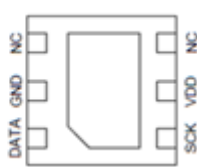
- [Upboard Lab04](#)
- [I. Introduction](#)
- [II. HTU21D](#)
 - [HTU21D – Output Conversion](#)
- [III. I²C](#)
- [IV. Serial port](#)
- [V. Serial/Parallel communication](#)
 - [Serial programming in Linux](#)
 - [Serial programming in x86 ASM](#)
- [VI. TTY](#)
- [VII. ANSI escape code](#)
- [VIII. Demonstration](#)
 - [PuTTY](#)
 - [Connection diagram](#)

I. Introduction

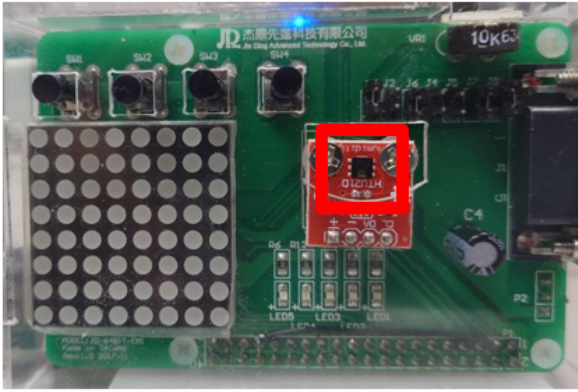
In this lab, we cover

1. Serial programming
2. Accessing sensors with over I2C

II. HTU21D



- Digital Relative Humidity sensor with Temperature output (datasheet)
- 以 I2C 控制
- UP board PIN:
 - DATA – 3 (I2C1-SDA)
 - SCK – 5 (I2C1-SCL)



N°	Function	Comment
1	DATA	Data bit-stream
2	GND	Ground
3	NC	Must be left unconnected
4	NC	Must be left unconnected
5	VDD	Supply Voltage
6	SCK	Serial clock input
PAD		Ground or unconnected

HTU21D – Output Conversion

The raw output (16 bits) need to be converted
Relative humidity conversion

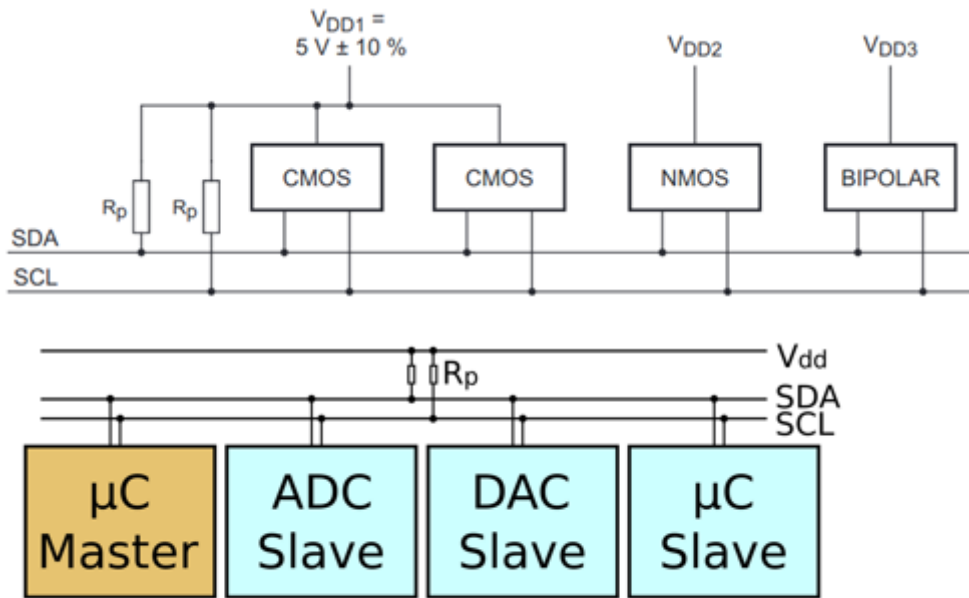
Temperature conversion

$$Temp = -46.85 + 175.72 \times \frac{S_{Temp}}{2^{16}}$$

RH: Relative Humidity (%)
 S_{RH} : RH signal output
 Temp: Temperature (°C)
 S_{Temp} : Temp signal output

III. I²C

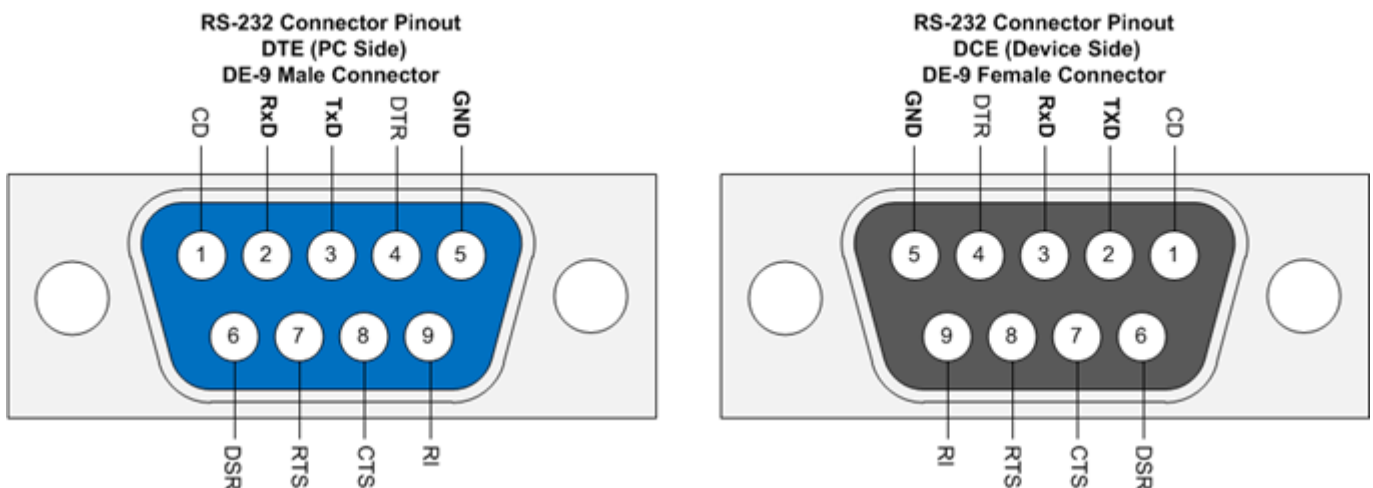
- I²C (Inter-Integrated Circuit, I-squared-C) (spec)
- 一種多主從的序列通信協定
- 使用兩條接上拉電阻的線 (open-drain/collector)
 - SDA: serial data
 - SCL: serial clock
- 在埠上的裝置有一個 7-bit* 的位址
- master 以位址選取 slave 裝置後讀寫其資料



* 通常是 7-bit, 也有較罕見的 10-bit

IV. Serial port

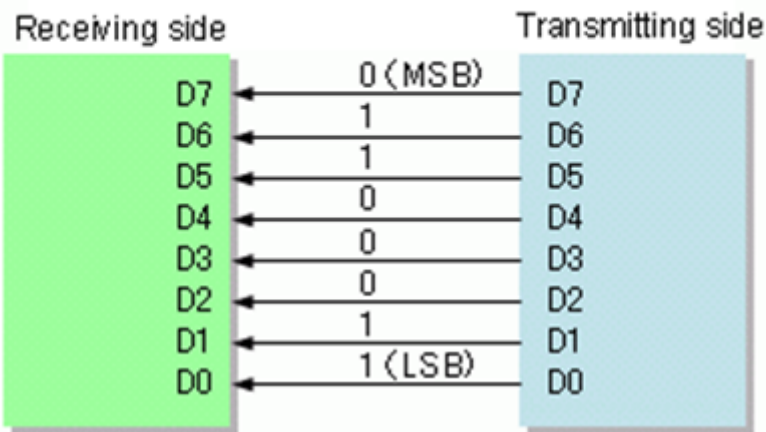
- A serial communication interface
- Commonly exposed as an RS-232 DE-9 connector
- Commonly implemented with a UART
 - Universal Asynchronous Receiver/Transmitter
- Speed is determined by the Baud rate setting (bps)



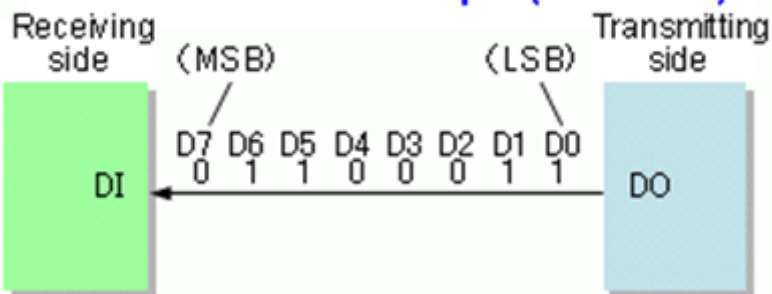
V. Serial/Parallel communication

- Serial 序列：一次一個 bit
- Parallel 平行：多條線同時傳

Parallel interface example



Serial interface example (MSB first)



Serial programming in Linux

- 序列界面在 Linux 中以裝置檔案 `/dev/ttyS[0-3]` 呈現
- 往界面傳輸 = 寫東西到檔案
- 從界面取資料 = 從檔案讀資料
- 使用一般讀寫檔案的函式即可 (`fprintf`, `read`, `write`, ...)
- Serial Programming Guide for POSIX Operating Systems
- UP 上的 Serial Port 對應到裝置檔 `/dev/ttyS0`

Serial programming in x86 ASM

- Serial is port-mapped on x86
- Uses `IN` and `OUT` instructions
- Example code: Initialization (OSDev Wiki)

```

#define PORT 0x3f8          // COM1

static int init_serial() {
    outb(PORT + 1, 0x00);    // Disable all interrupts
    outb(PORT + 3, 0x80);    // Enable DLAB (set baud rate divisor)
    outb(PORT + 0, 0x03);    // Set divisor to 3 (lo byte) 38400 baud
    outb(PORT + 1, 0x00);    //                               (hi byte)
    outb(PORT + 3, 0x03);    // 8 bits, no parity, one stop bit
    outb(PORT + 2, 0xC7);    // Enable FIFO, clear them, with 14-byte threshold
    outb(PORT + 4, 0x0B);    // IRQs enabled, RTS/DSR set
    outb(PORT + 4, 0x1E);    // Set in loopback mode, test the serial chip
    outb(PORT + 0, 0xAE);    // Test serial chip (send byte 0xAE and check if serial returns same byte)

    // Check if serial is faulty (i.e: not same byte as sent)
    if(inb(PORT + 0) != 0xAE) {
        return 1;
    }

    // If serial is not faulty set it in normal operation mode
    // (not-loopback with IRQs enabled and OUT#1 and OUT#2 bits enabled)
    outb(PORT + 4, 0x0F);
    return 0;
}

```

- I/O address and the register map can be found on the SoC's datasheet (vol1, vol2)

Table 16. Fixed I/O Ranges in the Platform Controller Unit (PCU)

Device	I/O Address	Comments
8259 Master	20h-21h, 24h-25h, 28h-29h, 2Ch-2Dh, 30h-31h, 34h-35h, 38h-39h, 3Ch-3Dh	
8254s	40h-43h, 50h-53h	
PS2 Control	60h, 64h	
NMI Controller	61h, 63h, 65h, 67h	
RTC	70h-77h	
Port 80h	80h-83h	
INIT Register	92h	
8259 Slave	A0h-A1h, A4h-A5h, A8h- A9h, ACh-ADh, B0h-B1h, B4h-B5h, B8h-B9h, BCh- BDh, 4D0h-4D1h	
PCU UART	3F8h-3FFh	
Reset Control	CF9h	Overlaps PCI I/O registers
Active Power Management	B2h-B3h	

18.5 Register Map

Table 105. Register Access List

Register Address (Offset to Base IO Address)	COM1_LCR.DLA B Value	Register Access Type	Register Accessed
0h	0b	RO	Receiver Buffer ¹
0h	0b	WO	Transmitter Holding ¹
0h	1b	RW	Divisor Latch LSB (Lowest Significant Bit) ¹
1h	0b	RW	Interrupt Enable ²
1h	1b	RW	Divisor Latch MSB (Most Significant Bit) ²
2h	xb	RO	Interrupt Identification ³
2h	xb	WO	FIFO Control ³
3h	xb	RW	Line Control
4h	xb	RW	Modem Control ⁴
5h	xb	RO	Line Status
6h	xb	RO	Modem Status ⁴
7h	xb	RW	Scratchpad

VI. TTY

- man 3 termios
- The TTY demystified



○

Real teletypes in the 1940s.

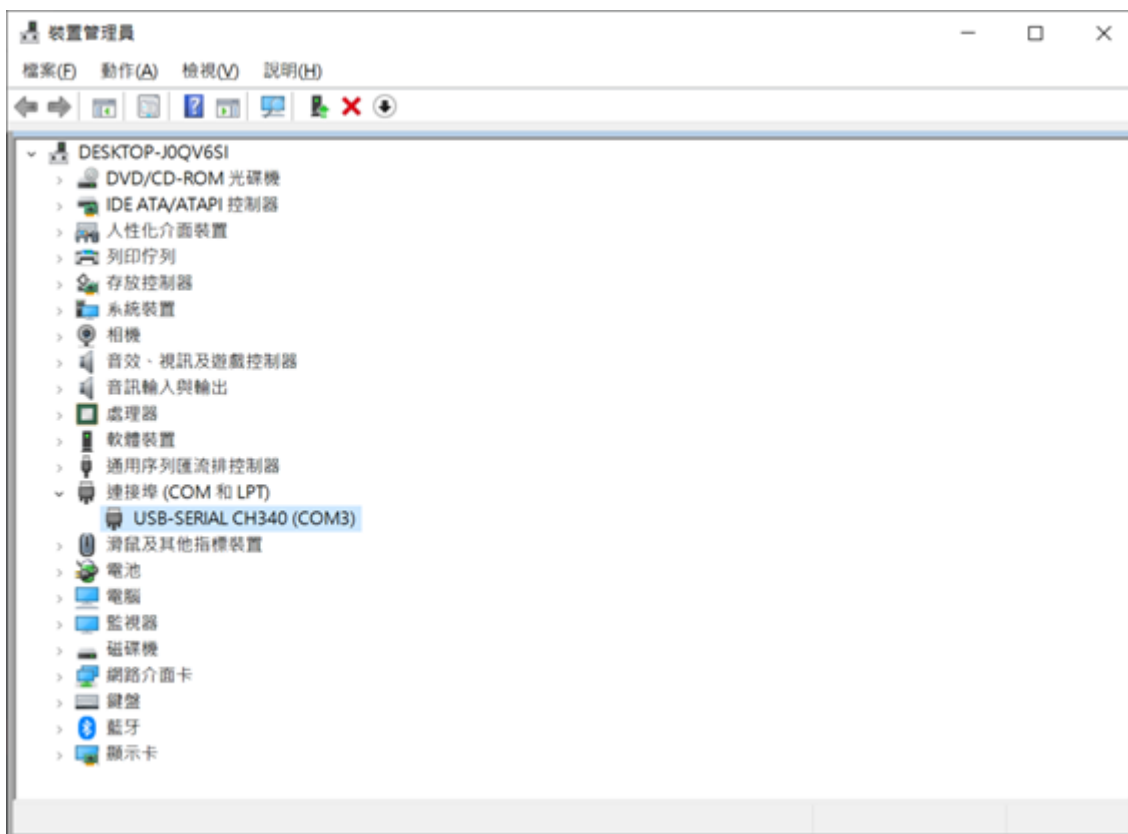
- tty一詞源於Teletypes，或者teletypewriters，原來指的是電傳打字機，是通過串行線用打印機鍵盤通過閱讀和發送信息的東西，後來這東西被鍵盤與顯示器取代，所以現在叫終端比較合適。
- 終端是一種字符型設備，它有多種類型，通常使用tty來簡稱各種類型的終端設備。
- TTY 大概與 terminal 同義
- 每個 TTY 有 terminal attributes，用以設定這個 TTY 的行為，例如：
 - ECHO: Echo input characters
 - ICANON: Enable canonical mode
- Linux 裡，/dev/ttyS0 是一個 TTY，其 terminal attributes 可透過 tcgetattr, tcsetattr 等函數存取修改

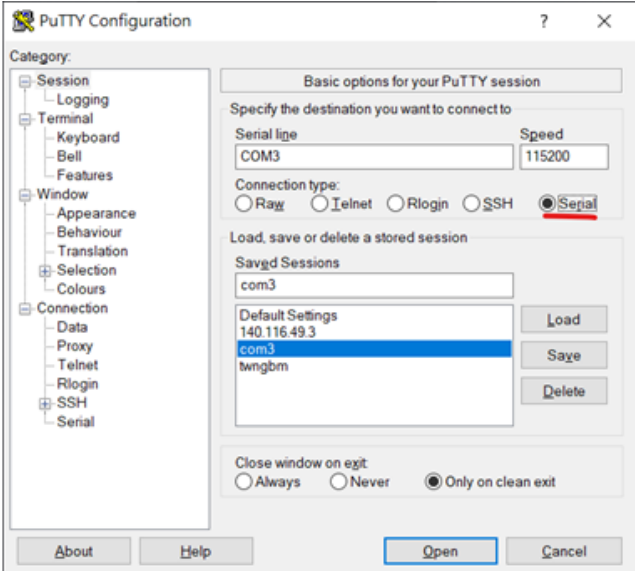
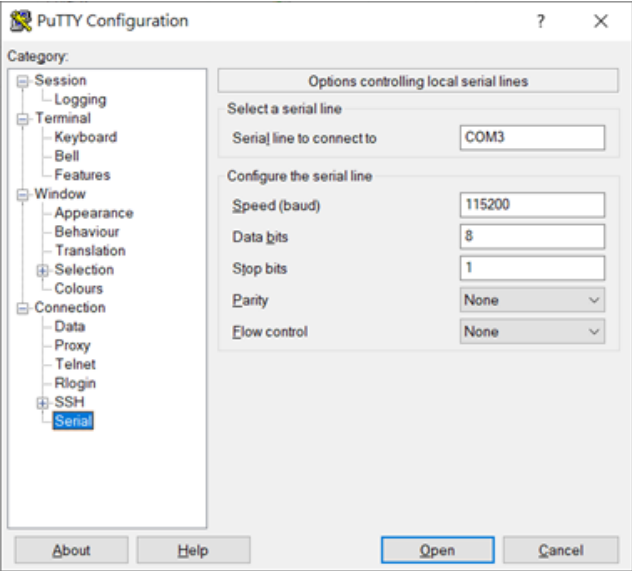
VII. ANSI escape code

- 在 terminal 輸出某些特定字串時，terminal 會做出單純印出該字串以外的事情，如移動游標，改變顏色等
- 例：
 - ^H (BS): 將游標向左移
 - ESC [A: 將游標向上移
 - ESC [J: 將游標以後全部清空
- 在按某些按鍵時，terminal 收到的可能收到不只一個字
 - 例：按 方向鍵下 會傳送 "ESC [B"
 - Why does the terminal show ^[[A?
<https://reurl.cc/gW2487> (<https://reurl.cc/gW2487>).

VIII. Demonstration

PuTTY





Connection diagram



DEV: Virtualbox dev environment
HOST: Windows host