SINH HOẠT HỌC THUẬT THÁNG 1/2017:

# GIAO THỨC TRUYỀN DỮ LIỆU NỐI TIẾP

NGUYỄN VĂN HÂN
Bộ môn: Điện tử - Tự động
Khoa: Điện – Điện tử
Trường Đại học Nha Trang
Email: hannv@ntu.edu.vn

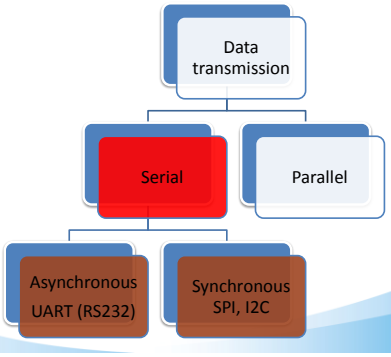
www.ntu.edu.vn

## NỘI DUNG

- GIỚI THIỆU
- GIAO THỨC TRUYỀN DỮ LIỆU BẤT ĐỒNG BỘ
  - UART (RS232)
- GIAO THỨC TRUYỀN DỮ LIỆU ĐỒNG BỘ
  - I2C
  - SPI

www.ntu.edu.vn

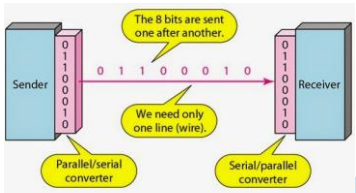
## GIỚI THIỆU




www.ntu.edu.vn

## TRUYỀN DỮ LIỆU NỐI TIẾP

- …Serial communication is the process of sending data one bit at a time, sequentially, over a communication channel or computer bus*…




*https://en.wikipedia.org/wiki/Serial_communication

www.ntu.edu.vn

# TRUYỀN DỮ LIỆU NỐI TIẾP

**Ưu điểm**

- Tốn ít dây dẫn và cổng kết nối → giảm giá thành
- Khoảng cách truyền dẫn lớn
- Nhiều IC và thiết bị ngoại vi tích hợp sẵn module truyền dữ liệu nối tiếp

**Nhược điểm**

- Tốc độ truyền dẫn thấp (tại thời điểm hiện tại)

**Ứng dụng**

- Giao tiếp với IC với IC (microcontroller), giữa máy tính với IC và thiết bị ngoại vi

# UART, SPI, I2C

| | Synchronous | | Asynchronous |
|---|---|---|---|
| **Peripheral** | SPI | I²C | UART |
| **Max Bit Rate** | 10 Mbit/s | 1 Mbit/s | 500 kbit/s |
| **Max Bus Size** | Limited by no. of pins | 128 devices | Point-to-point (RS232)<br>256 devices (RS485) |
| **Number of Pins** | 3 + n × CS | 2 | 2 |
| **Pros** | Simple, low cost, high speed | Small pin count, allows multiple masters | Longer distance, improved noise immunity (requires transceivers) |
| **Cons** | Single master, short distance | Slowest, short distance | Requires accurate clock frequency |
| **Typical Application** | Direct connection to ASICs and other peripherals on same PCB | Bus connection with peripherals on same PCB | Interface with terminals, personal computers and other data acquisition systems |
| **Examples** | Serial EEPROMs (25CXXX series), MCP320X A/D converter, ENC28J60 Ethernet controller, MCP251X CAN controller... | Serial EEPROMs (24CXXX series), MCP98XX tempera-ture sensors, MCP322x A/D converters... | RS232, RS422, RS485, LIN bus, MCP2550 IrDA interface... |

# TRUYỀN/NHẬN DỮ LIỆU NỐI TIẾP BẤT ĐỒNG BỘ (UART)

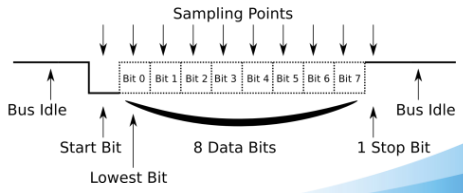
- **Universal asynchronous receiver/transmitter** (UART /ˈjuːɑːrt/), là một thiết bị của máy tính để truyền/nhận dữ liệu nối tiếp bất đồng bộ với định dạng và tốc độ có thể thiết lập được*
- Định dạng thường gặp: 8 bit dữ liệu + start bit + stop bit + (parity bits); mức điện áp tín hiệu phụ thuộc vào phần cứng bên ngoài
- Tốc độ thường sử dụng 9600 bauds (9600 bit per second); không cần tín hiệu clock

*https://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter

# UART

- Định dạng khung dữ liệu (8 bit dữ liệu, 1 bit start, 1 bit stop, không có bit parity)

UART with 8 Databits, 1 Stopbit and no Parity



# UART

**Truyền (Ví dụ trên VĐK 8051)**

- Thiết lập tốc độ baud (9600) → Khởi tạo Timer
- Kiểm tra đường truyền → Thanh ghi SCON (TI)
- Truyền 1 byte → Thanh ghi SBUFF

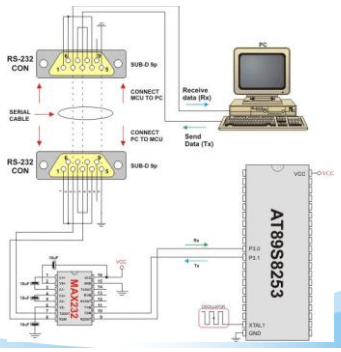
**Nhận (Ví dụ trên VĐK 8051)**

- Thiết lập tốc độ baud (9600) → Khởi tạo Timer
- Kiểm tra đường truyền → Thanh ghi SCON (RI)
- Nhận 1 byte → Thanh ghi SBUFF



# UART: GIAO TIẾP MÁY TÍNH

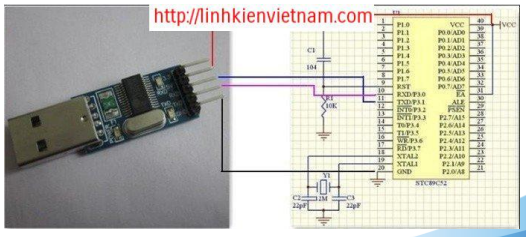
8051(TX, RX)
↕
MAX232
↕
PC
(COM PORT)






# UART: GIAO TIẾP MÁY TÍNH

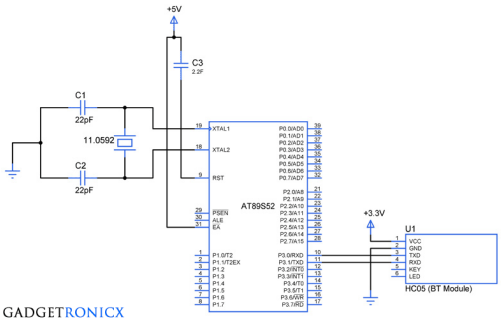
- PC (USB) ↔ PL2303HX ↔ 8051 (TX,RX)




PL2303HX (USB to UART converter)

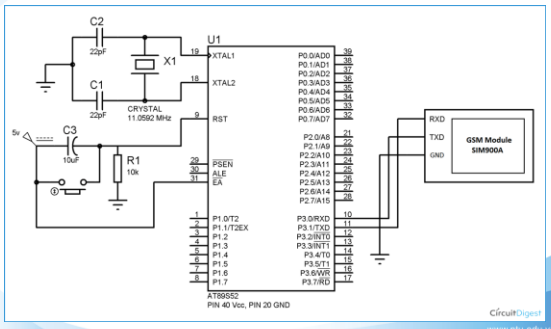


# UART: BLUETOOTH MODULE




GADGETRONICX

# UART: GSM MODULE



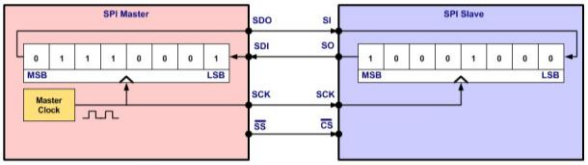



# SPI

- **Serial Peripheral Interface** (SPI) là giao thức truyền/nhận dữ liệu nối tiếp đồng bộ do Motorola phát triển
- Là giao thức song công (full-duplex), sử dụng kiến trúc master-slave(s), tốc độ truyền dữ liệu cao lên đến 10 Mbps
- Đôi khi còn gọi là giao thức 4 dây (four-wire) vì sử dụng 4 dây kết nối (MOSI, MISO, SCK, SS)



# SPI: 1 MASTER 1 SLAVE



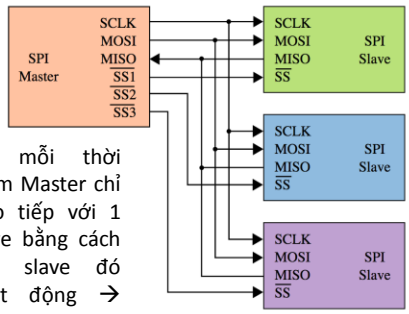

| PIN | Description | Direction |
|---|---|---|
| SDO | Serial Data Out, This pin also called as MOSI (Master Out Serial In) in AVR Type Microcontroller | Out |
| SDI | Serial Data In, This pin also called as MISO (Master In Serial Out) in AVR Type Microcontroller | In |
| SCK | Serial Clock (SPI Master) | Out |
| SS | Serial Select (could be any I/O Port) | Out |

| PIN | Description | Direction |
|---|---|---|
| SI | Serial In | In |
| SO | Serial Out | Out |
| SCK | Serial Clock (SPI Slave) | In |
| CS | Chip Select (usually active low) | In |

The SPI Master and SPI Slave Basic Connection

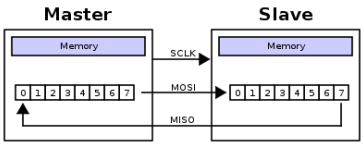


# SPI: 1 MASTER MULTI-SLAVE




Tại mỗi thời điểm Master chỉ giao tiếp với 1 slave bằng cách cho slave đó hoạt động → $\overline{SSi} = 0$

# SPI: NGUYÊN LÝ HOẠT ĐỘNG

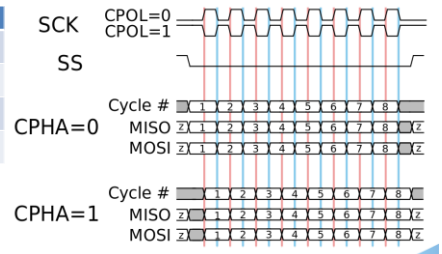



- Master khởi tạo kết nối và điều khiển kết nối (tạo xung SCLK, tín hiệu SS), truyền/nhận dữ liệu; Slave chỉ truyền/nhận dữ liệu
- Mỗi chu kỳ xung SCLK, 1 bit từ Master truyền tới Slave trên đường MOSI; đồng thời 1 bit từ Slave truyền đến Master trên đường MISO



# SPI: NGUYÊN LÝ HOẠT ĐỘNG

| Mode | CPOL | CPHA |
|---|---|---|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 2 | 1 | 0 |
| 3 | 1 | 1 |




CPOL (Clock Polarity): Cực tính xung đồng bộ

CPHA (Clock Phase): Pha của xung đồng bộ



# SPI: NGUYÊN LÝ HOẠT ĐỘNG

- **At CPOL=0 (the idle state is 0 and active state is 1):**
  - For CPHA=0, data are captured on the clock's rising edge (low→high transition) and data is output on a falling edge (high→low clock transition).
  - For CPHA=1, data are captured on the clock's falling edge and data is output on a rising edge.
- **At CPOL=1 (the idle state is 1 and active state is 0):**
  - For CPHA=0, data are captured on clock's falling edge and data is output on a rising edge.
  - For CPHA=1, data are captured on clock's rising edge and data is output on a falling edge.



# SPI: GIAO TIẾP VỚI EEPROM 25LC256

```
//Khoi tao SPI mode 0
//(lay mau DL o canh len; chot DL o canh xuong)
void Soft_SPI_Init()
{
    SPI_CS = 1;
    SPI_SCLK = 0;
    SPI_MISO = 1;
    SPI_MOSI = 0;
}
```

```
//Gui mot by
void Soft_SP
{
    unsigned

    for(i=0;i<8;i++)
    {
        if((b&0x80)==0x80)
            SPI_MOSI = 1;
        else
            SPI_MOSI = 0;

        SPI_SCLK = 1; //Lay mau DL
        b <<= 1;
        SPI_SCLK = 0; //Chot DL de truyen di
    }
}
```

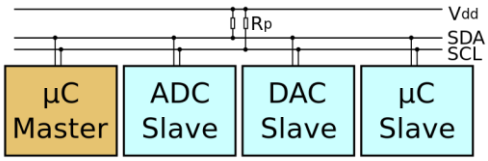
```
void)

;i++)

;
K = 1; //Lay mau DL
MISO)
= 0x01;
K = 0; //Chot DL nhan vao
```

# I2C

- **Inter-Integrated Circuits (I2C) bus** là chuẩn truyền/nhận dữ liệu nối tiếp đồng bộ do Philips phát triển những năm 1980,
- Là giao thức multi-master, multi-slave → trên cùng 1 bus có thể nhiều master, nhiều slave, tốc độ dữ liệu lên đến vài Mbps (thông thường 100 kbps)
- Còn gọi là giao thức 2 dây (Two-wire: SDA, SCL)



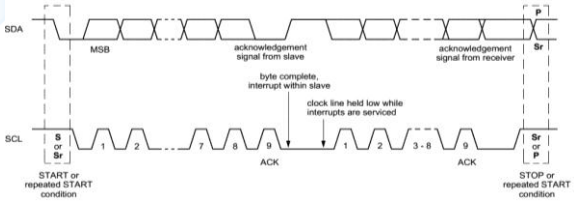
# I2C




- Master: Thiết lập phiên và truyền/nhận dữ liệu, truyền xung clock
- Slave: Truyền/nhận dữ liệu
- SDA (Serial Data): Bus dữ liệu 2 chiều
- SCL (Serial Clock): Bus xung đồng bộ chiều từ master → slave
- Bus SDA và SCL được kéo lên nguồn bằng điện trở treo (thường 4,7kΩ với Vdd = 5 V)



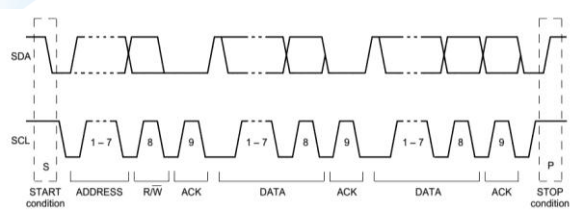
# I2C: NGUYÊN LÝ HOẠT ĐỘNG




- Chế độ 1 master – 1 slave
  - START (master) → ADDRESS(7/10bit) + R/W (1 bit) ← ACK → DATA (8 bit) ← ACK → STOP (master)
  - START (master) → ADDRESS(7bit) + R/W …← NACK (slave) → STOP/REPEAT START (master)



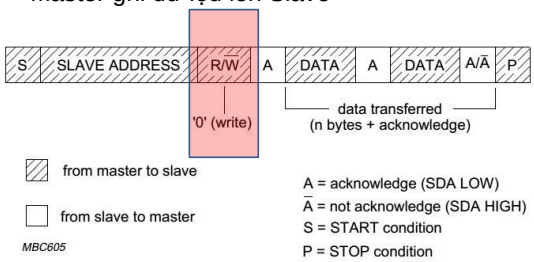
# I2C: NGUYÊN LÝ HOẠT ĐỘNG




- Định dạng dữ liệu:
  - START: SDA cao → thấp + SCL cao
  - STOP: SDA thấp → cao +SCL cao
  - Salve ADRRESS: 7 bit (hoặc 10 bit) → tối đa 128 thiết bị
  - DATA: 1 byte (8 bit)
  - R/W: 0 → ghi DL từ master → slave; 1 → đọc dữ liệu từ slave → master
  - ACK/NACK: 0: ACK → thành công; 1: NACK → không thành công

# I2C: NGUYÊN LÝ HOẠT ĐỘNG

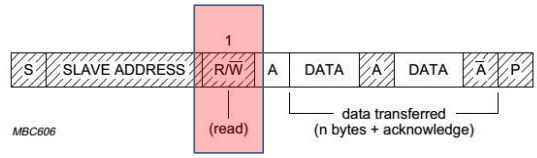
- Master ghi dữ lệu lên Slave






# I2C: NGUYÊN LÝ HOẠT ĐỘNG

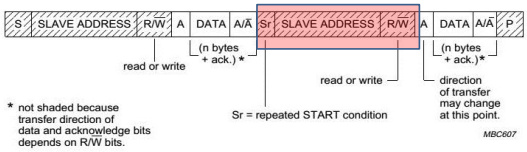
- Master đọc dữ liệu từ Slave





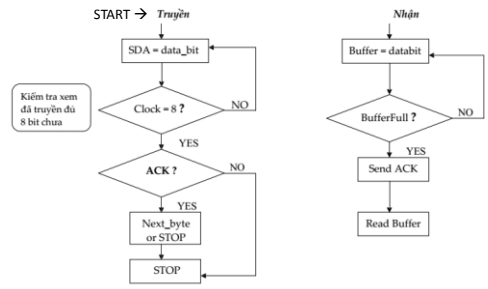

# I2C: NGUYÊN LÝ HOẠT ĐỘNG

- MASTER KẾT HỢP ĐỌC/ GHI DỮ LIỆU TỪ SLAVE; HOẶC GIAO TIẾP SLAVE KHÁC
  - Trong chu trình START và STOP, master có thể đọc/ghi nhiều lần, hoặc giao tiếp slave khác bằng cách repeated START → Gửi lại trạng thái START



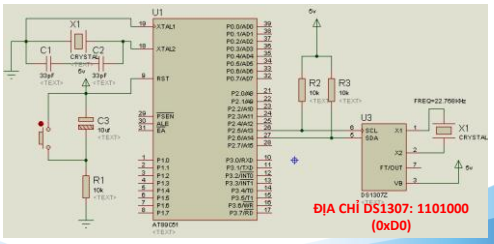



# I2C: NGUYÊN LÝ HOẠT ĐỘNG



# I2C: GIAO TIẾP VỚI DS1307

- Cài đặt thời gian: Ghi dữ liệu từ master → slave
- Đọc và hiển thị thời gian: Đọc DL từ slave → master






# THANK YOU