

3HB07

RTOS : Boards communication

วัตถุประสงค์

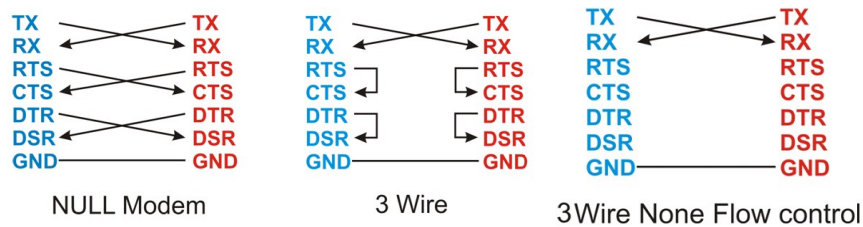
- สามารถใช้งาน Real-time os บน ระบบ Embedded ขนาดเล็กในการสื่อสารตอบโต้กันระหว่างอุปกรณ์ได้
- เข้าใจการสื่อสารแบบ Uart

การสื่อสาร

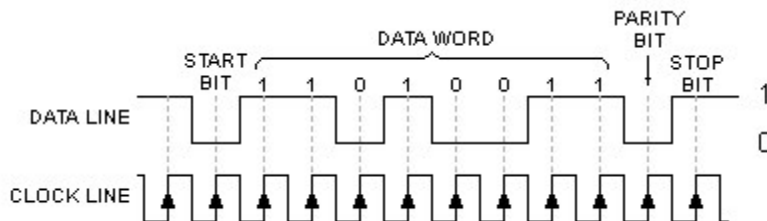
การสื่อสารแบบ Uart (Universal Asynchronous Receiver and Transmitter) เป็นมาตรฐานการสื่อสารข้อมูลแบบ Serial โดยมาตรฐานการสื่อสารแบบ Uart นั้นยังแบ่งออกเป็นรูปแบบการสื่อสารออกเป็น RS-232, RS-422, RS-485 และอื่น ๆ โดยรูปแบบการสื่อสารดังกล่าวข้างต้นจะมีความแตกต่างกันในเรื่อง แรงดันในการสื่อสาร ระยะทางในการสื่อสาร รูปแบบการสื่อสาร (Full-duplex/Half duplex) โดยความแตกต่างการสื่อสารแบบ Uart นั้นสามารถแสดงได้ดังรูปที่ 1

Port name	RS-232	RS-422	RS-485
Transfer type	Full duplex	Full duplex	Half duplex (2 wires),
Maximum distance	15 meters at 9600 bps	1200 meters at 9600 bps	1200 meters at 9600 bps
Contacts in use	TxD, RxD, RTS, CTS, DTR, DSR,	TxA, TxB, RxA, RxB, GND	DataA, DataB, GND
Topology	Point-to-Point	Point-to-Point	Multi-point
Max. Number of connected devices	1	1 (10 devices in receive mode)	32 (with repeaters larger, usually up to 256)

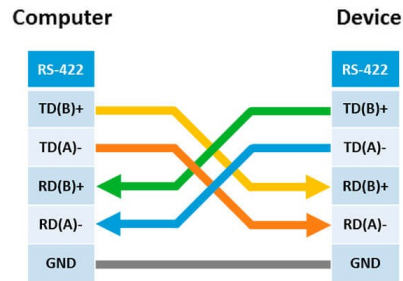
รูป 1 ตารางแสดงความแตกต่าง มาตรฐานการสื่อสารของ Uart



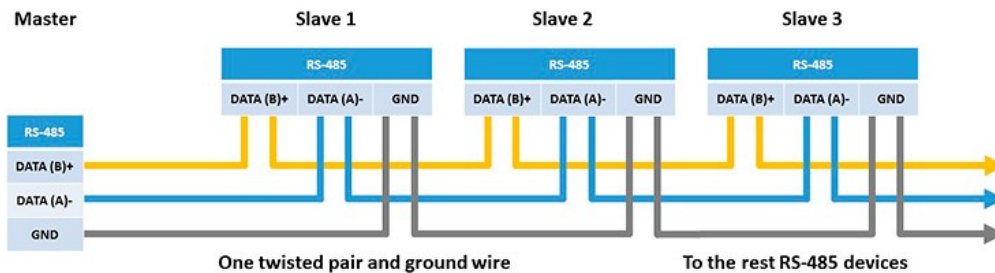
รูป 2 รูปแบบการเชื่อมโยง RS-232



รูป 3 สัญญาณการสื่อสารแบบ RS-232



รูป 4 การเชื่อมโยงแบบ RS-422



รูป 5 การเชื่อมโยงแบบ RS-485

ความเร็วในการสื่อสารใน Uart นั้นจะระบุความเร็วเป็น Baud rate เช่น 1200, 9600, 19200, 38400, 115200 โดย Baud rate หมายถึงจำนวน ตัวอักษร/สัญลักษณ์ ที่ส่งในหนึ่งวินาที โดยจำนวนบิตของ 1 ตัวอักษร/สัญลักษณ์ ขึ้นอยู่กับข้อกำหนดในแต่ละระบบที่แตกต่างกันออกไป ซึ่งมาตรฐานในการส่งข้อมูล Uart ในปัจจุบันนิยมใช้มาตรฐาน Ascii ในการสื่อสารซึ่งจะแทน 1 ตัวอักษรด้วย ข้อมูล 8 Bit โดยผู้ใช้งานสามารถกำหนดการเชื่อมโยงได้ เช่น 9600,N,8,1 สามารถแปลข้อกำหนดในการสื่อสารเป็น Baud rate = 9600, Parity =None, Data Bits 8, Stop Bits=1



รูป 6 ตัวอย่างข้อมูลในการสื่อสารแบบ Uart

คำถามที่ 1 _____

คำตอบ _____

การทดลอง

- กำหนดให้ Arduino board ติดต่อเชื่อมโยงความเร็ว uart ด้วยความเร็ว 115200 จงเขียนโปรแกรมให้ Board Arduino ส่งข้อมูล Digital Pin 1-13 ทุก ๆ 300 ms. และ Analog pin 0-5 ทุก ๆ 900 ms. โดยใช้ FreeRTOS

(Check point 1)

ตัวอย่างโค้ด

```
void TaskWriteAnalog( void *pvParameters ); //priority = 1
```

```
{ (void) pvParameters;
```

```
int i;
```

```
for (;;) {
```

```
for(i=0;i<5;i++)
```

```
{ Serial.println(analogRead(i));
```

}

}

}

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

- (Check point 2)

ตัวอย่างโค้ด

```
void TaskSerialwrite( void *pvParameters ); //priority = 2
```

```
{ (void) pvParameters;
```

```
{ if (Serial.available()) {
```

..... }

```
vTaskDelay(10);
```

}

}

[illegible]

