

PRAKTIKUM

KOMUNIKASI SPI MASTER SLAVE

TUJUAN :

1. Memahami prinsip mengirim dan menerima data melalui SPI pada mikrokontroler.
2. Mampu membuat program untuk mengirim dan menerima data dalam bentuk *byte* melalui SPI.
3. Mampu membuat program untuk mengirim dan menerima data dalam bentuk teks melalui SPI.

PERALATAN :

- | | |
|-----------------------|---------|
| 1. Arduino Uno | : 2 pcs |
| 2. Project Board | : 2 pcs |
| 3. Switch Push Button | : 3 pcs |
| 4. LCD 2x16 | : 1 pcs |
| 5. Potensiometer 10K | : 1 pcs |
| 6. Kabel Jumper | : 1 set |

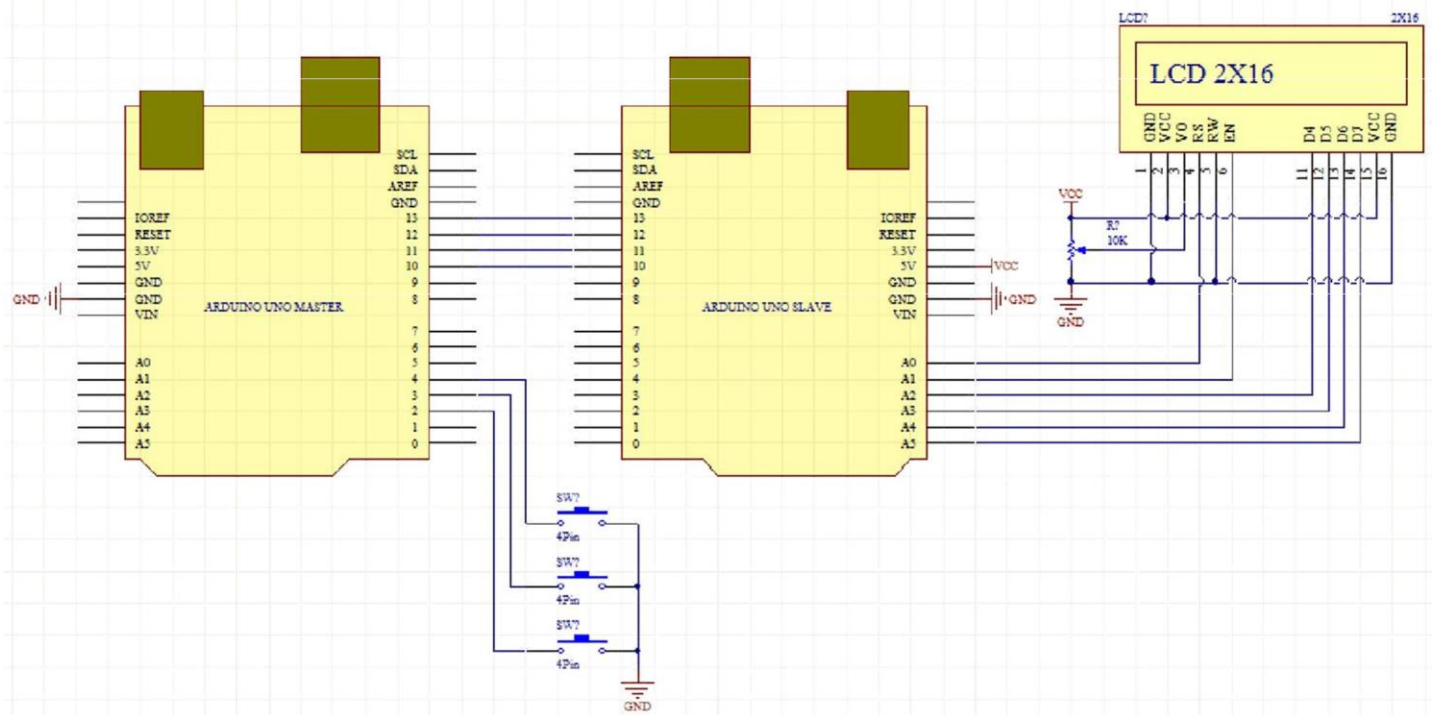
PERCOBAAN :

1 Mengirim dan Menerima Data *Byte* Pada Komunikasi SPI Master Slave Arduino

Pada percobaan kali ini akan dibuat dua buah program mikrokontroler. Program tersebut nantinya berfungsi untuk mengirimkan dan menerima data melalui jalur komunikasi SPI. Salah satu mikrokontroler akan bertindak sebagai *master*. Tugas mikrokontroler *master* adalah mengirimkan data dalam bentuk *byte* yang nilainya disesuaikan oleh penekanan tombol. Mikrokontroler lainnya akan bertindak sebagai *slave* yang tugasnya menerima data dalam bentuk *byte* dari *master* kemudian menampilkan data tersebut ke LCD 2x16.

Prosedur Praktikum :

1. Buatlah rangkaian untuk mikrokontroler *master* seperti Gambar 1.



Gambar 1 Rangkaian Percobaan 1

2. Tuliskan sintaks program berikut ini pada Arduino IDE, kemudian *upload* program tersebut ke Arduino *master*.

```
void initSPI_Master() {
    DDRB |= (1<<2)|(1<<3)|(1<<5); // SCK, MOSI dan SS menj adi output
    DDRB &= ~(1<<4); // MISO menjadi input
    SPCR |= (1<<MSTR); // SPI sebagai master
    SPCR |= (1<<SPR0)|(1<<SPR1); // Pembagi clock = 128
    SPCR |= (1<<SPE); // Aktifkan SPI
}

void setup() {
    initSPI_Master();
    pinMode(2, INPUT);    pinMode(3, INPUT);    pinMode(4, INPUT);
    digitalWrite(2, HIGH); digitalWrite(3, HIGH);    digitalWrite(4, HIGH);
}

void loop() {
    if(digitalRead(2)==LOW)
        kirimData(5);
    else if(digitalRead(3)==LOW)
        kirimData(10);
    else if(digitalRead(4)==LOW)
        kirimData(15);
}

void kirimData(unsigned char data) {
    SPDR = data; // Kirim data
    while(!(SPSR & (1<<SPIF))); // Tunggu sampai pengiriman selesai
    delay(50);
}
```

3. Tuliskan sintaks program berikut ini pada Arduino IDE, kemudian *upload* program tersebut ke Arduino *slave*.

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);

unsigned char data;

void initSPI_Slave() {
    DDRB &= ~((1<<2)|(1<<3)|(1<<5)); // SCK, MOSI dan SS menjadi input
    DDRB |= (1<<4); // MISO menjadi output
    SPCR &= ~(1<<MSTR); // SPI sebagai slave
    SPCR |= (1<<SPR0)|(1<<SPR1); // Pembagi clock = 128
    SPCR |= (1<<SPE); // Aktifkan SPI
}

void setup() {
    initSPI_Slave();
    lcd.begin(16, 2);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("SPI Slave Begin");
}

void loop() {
    while(!(SPSR & (1<<SPIF))); // Tunggu data masuk
    data = SPDR; // Menyimpan isi register SPDR ke variabel data
    lcd.setCursor(0,0);
    lcd.print("Data Receive ==> ");
    lcd.setCursor(0,1);
    lcd.print(data,HEX);
}
```

Tugas dan Pertanyaan :

1. Tekanlah salah satu tombol yang terhubung pada Arduino *master*, kemudian isilah data pada Tabel 1 berikut ini.

Tabel 1 Data yang dikirimkan Arduino master

Tombol	Data Pada LCD
2	5
3	A
4	F

2 Mengirim dan Menerima Data Teks Pada Komunikasi SPI Master Slave Arduino

Pada percobaan kali ini akan dibuat dua buah program mikrokontroler. Program tersebut nantinya berfungsi untuk mengirimkan dan menerima data dalam bentuk teks melalui jalur komunikasi SPI. Salah satu mikrokontroler akan bertindak sebagai *master*. Tugas mikrokontroler *master* adalah mengirimkan data dalam bentuk teks yang isinya disesuaikan oleh penekanan tombol. Mikrokontroler lainnya akan bertindak sebagai *slave* yang tugasnya menerima data yang dikirimkan dalam bentuk teks dari *master* kemudian menampilkan data tersebut ke LCD 2x16.

Prosedur Praktikum :

1. Gantilah program pada Arduino master menjadi seperti sintaks program berikut ini. Kemudian *upload* program tersebut ke Arduino yang bertindak sebagai *master*.

```
void initSPI_Master() {
    DDRB |= (1<<2)|(1<<3)|(1<<5);    // SCK, MOSI dan SS menjadi output
    DDRB &= ~(1<<4);                  // MISO menjadi input
    SPCR |= (1<<MSTR);                // SPI sebagai master
    SPCR |= (1<<SPR0)|(1<<SPR1);      // Pembagi clock = 128
    SPCR |= (1<<SPE);                // Aktifkan SPI
}
void setup() {
    initSPI_Master();
    pinMode(2, INPUT);               pinMode(3, INPUT);               pinMode(4, INPUT);
    digitalWrite(2, HIGH);           digitalWrite(3, HIGH);           digitalWrite(4, HIGH);
}
void loop() {
    if(digitalRead(2)==LOW) {
        kirimData("Hallo\r\n");
    }
    else if(digitalRead(3)==LOW) {
        kirimData("Apa\r\n");
    }
    else if(digitalRead(4)==LOW) {
        kirimData("Kabar\r\n");
    }
}
void kirimData(char *string) {
    int panjangString = strlen(string);
    for(int i=0; i<panjangString; i++) {
        SPDR = string[i];
        while(!(SPSR & (1<<SPIF)));
        delay(10);
    }
}
```

2. Gantilah program pada Arduino Slave menjadi seperti sintaks program berikut ini.

Kemudian *upload* program tersebut ke Arduino yang bertindak sebagai *slave*.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(A0, A1, A2, A3, A4,
A5); String dataString = "";
char data;

void initSPI_Slave() {
  DDRB &= ~((1<<2)|(1<<3)|(1<<5)); // SCK, MOSI dan SS me njadi input
  DDRB |= (1<<4); // MISO menjadi output
  SPCR &= ~(1<<MSTR); // SPI sebagai slave
  SPCR |= (1<<SPR0)|(1<<SPR1); // Pembagi clock = 128
  SPCR |= (1<<SPE); // Aktifkan SPI
}

void setup() {
  initSPI_Slave();
  lcd.begin(16, 2); lcd.clear();
  lcd.setCursor(0,0); lcd.print("SPI Slave Begin");
}

void loop() {
  while(!(SPSR & (1<<SPIF))); // Tunggu data masuk
  data = SPDR; // Menyimpan isi regis ter SPDR ke variabel data
  if(data!='\n') {
    if(data!='\r') {
      dataString += data; // Simpan data ke dala m string
    }
  }
  else { lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Data Receive ==> ");
  lcd.setCursor(0,1);
  lcd.print(dataString); // Menampilkan string ke LCD
  dataString = "";
  }
}
```

Tugas dan Pertanyaan :

1. Tekanlah salah satu tombol yang terhubung pada Arduino *master*, kemudian isilah data pada Tabel 2 berikut ini.

Tabel 2 Data yang dikirimkan Arduino master

Tombol	Data Pada LCD
2	Hallo
3	Apa
4	Kabar

2. Cobalah untuk menghubungkan beberapa arduino dengan komunikasi SPI. Arduino master mengirimkan data ke slave, kemudian slave mengirimkan data yang diterima master ke hyper Terminal / Serial monitor komputer.