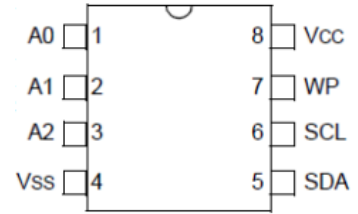




EEPROM'lar (Electrically Erasable Programmable Read Only Memory), elektriksel olarak yazılıp silinebilen ve sadece okunabilen yongalardır. Bilindiği gibi Mikrodenetleyici tarafından saklanması gereken Veri, Mikrodenetleyici güç kaynağı çalıştığı sürece RAM Bellek'de saklanır. Güç kesildiği anda RAM Bellek'deki tüm bilgiler kaybolur. Bu nedenle daha sonra tekrar kullanılması gereken verilerin sürekli veri saklayabilen (Permanent) bellekler'de saklanması gerekir. Bu amaçla kullanılabilen bellek entegre türlerinden biri Seri EEPROM'dur. Ayrıca EEPROM bellekler veri hızının yüksek olmadığı ve pin sayısının önem kazandığı uygulamalarda da kullanılır. Bu tip hafıza elemanları I<sup>2</sup>C, SPI veya 1-wire adı verilen protokoller ile iletişim sağlamaktadır. EEPROM Easy PIC7 üzerinde I<sup>2</sup>C protokolü ile harici depolama gereksinimi için kullanılmaktadır.

24C02, 2048 Byte=2K seri EEPROM'a ait pin numaraları Şekil 1'de gösterilmektedir. Burada A0, A1, A2 adres hatları, SCL Seri saat sinyali, SDA seri veri sinyali olmakla beraber, bellek mikro denetleyiciye bu hatlar üzerinden bağlanmaktadır. WP (Write Protect) yazma koruması olup +5 V'a bağlanırsa ilk 1K Byte'ı koruma altına alınır. V<sub>SS</sub> toprak hattı, V<sub>CC</sub> 4.5-5.5 V'luk güç girişidir.

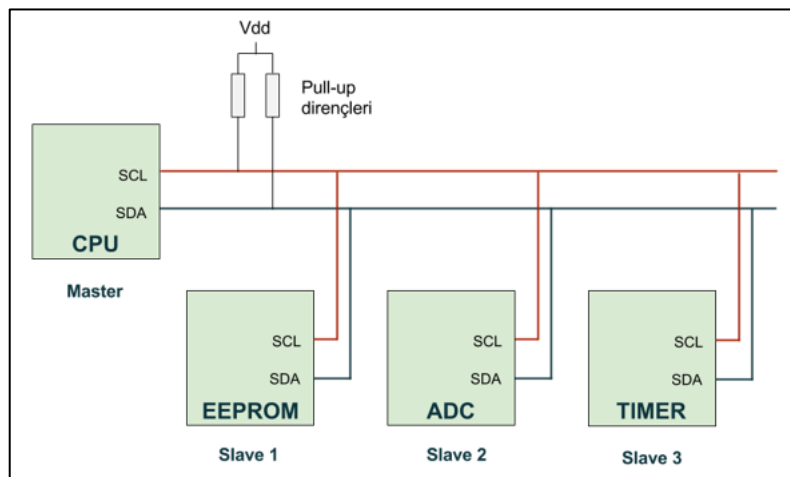


Şekil 1. 24C02 yapısı

## I<sup>2</sup>C protokolü

Aynı hat üzerinde birden fazla cihazın ekstra bir adres yoluna ihtiyaç duymaksızın, iletişimini mümkün kılan seri bir haberleşme protokolüdür. Bu protokole göre SDA ve SCL yolları üzerinde birden fazla cihaz bulunabilir (Şekil 2).

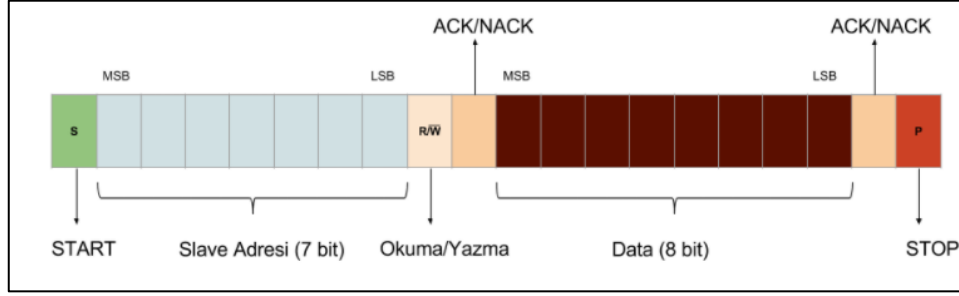
Protokole göre; haberleşmeyi başlatan ve aynı zamanda sonlandıran aygıtta **master** (MİB veya mikrodenetleyici), SDA ve SCL yolları üzerinden bulunan ve master cihazın saat işaretine uyan diğer aygıtlara **slave** adı verilir. Slave cihazlar 7 bitlik bir adrese sahip olmakla beraber bu adresler üzerinden cihazlara erişim sağlanabilir. Master ve slave cihazlar arasındaki iletişim çift yönlüdür ve bu yönü master belirler.



Şekil 2. I2C'ye ait haberleşme diyagramı

Bu protokol için haberleşme işlemi sırasıyla şu şekilde gerçekleşir:

Herhangi bir veri aktarımı gerçekleşmediği durumda hat boşdur (Idle durumu). Bu durumda SCL ve SDA hatları lojik 1 seviyesindedir. Master tarafından, SDA'nın lojik 1'den 0'a çekilmesiyle iletişim başlar. Adresleme aşamasında, master iletişime geçeceği slave aygıtı ve iletişim modunu (okuma veya yazma) belirler. Bit düzeyinde veri iletişimi gerçekleşir ve sonrasında master tarafından SDA'nın lojik 0'dan 1'e çekilmesiyle iletişim sonlandırılır.



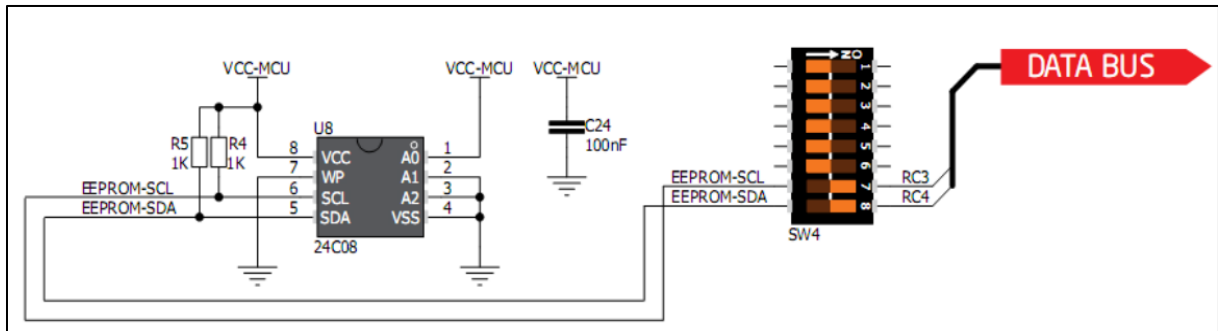
Şekil 3. I2C'ye ait çerçeve yapısı

I<sup>2</sup>C ile gönderilen veri paketine ait çerçeve yapısı Şekil 3'te gösterilmektedir. Slave adresi 7 bittir oluşmakla beraber, R/W biti okuma veya yazma işlemini seçer, okuma işlemi için 1 yazma işlemi için ise 0'a setlenir. Veri 8 bittir oluşur ve ACK ile NACK bitleri onay işlemi için kullanılmaktadır. Lojik 0 ACK ve Lojik 1 ise NACK'ı temsil etmektedir.

Örneğin bir yazma işlemi için master, slave adresi ve kontrol bitini iletir, belirtilen adres herhangi bir slave ile eşleşmesi durumunda, master bir onay biti (ACK) alır. Master veri gönderdiği sürece slave cihazdan ACK almaya devam edecektir.

## EEPROM DENEYİ

EasyPIC7 üzerindeki I2C EEPROM modülünün aktif olması için SW4 dip anahtarının 7. ve 8. Pinleri "ON" konumuna getirilmelidir. Böylelikle EEPROM'un Seri Clock ve Seri Data Pinleri mikro denetleyici ile haberleşmiş olur. I2C seri arabirimini kullanarak belirli bir veriyi EEPROM'a yazma ve benzer şekilde EEPROM byte içeriğini okuyarak bu bilgiyi PortB LED'lerinde "10101010" şeklinde gözlemlemek amaçlanmıştır.



### Örnek Kod:

EasyPic7 kartı üzerindeki EEPROM entegresine veri yazıp, yazılan veriyi okuyarak, okunan verinin PortB LED'lerinde gözlemlenmesi

/\*Konfigürasyon:

MCU : 18F45K22

Dev. Board : Easy PIC7

SW : mikroC PRO for PIC

Notlar (Board Spesifikasyonu):

- SW 4.7 ve SW 4.8 portları açık konuma getirilmeli
- PORTC'ye ait LED'ler kapalı konuma getirilmeli (SW 3.3)
- PORTB'ye ait LED'ler açık konuma getirilmeli (SW 3.2)

**Repeated Start:** Bir iletişim kurulduktan sonra mevcut iletişimi sonlandırmadan, yeni bir başlangıç durumu oluşturmak ve slave'i yeniden adreslemek için kullanılır. Bu durum hattın kullanımını kaybetmemeyi, aygıtlar arasındaki iletişimin kesilmemesini ve birbirini takip eden okuma ve yazma işlemlerini yapabilmeyi sağlar. Burada EEPROM'a yeni komutlar gönderilmesi amacıyla başlangıç durumunun yeniden oluşturulması için kullanılmıştır. \*/

```
void main(){
```

```
ANSELB = 0; // PORTB'ye ait pinleri dijital olarak ayarla
```

```
ANSELC = 0; // PORTC'ye ait pinleri dijital olarak ayarla
```

```
TRISB = 0; // PORTB'yi çıkış olarak ayarla
```

```
LATB = 0; // PORTB'yi temizle
```

```
I2C1_Init(100000); // I2C haberleşmeyi başlat
```

```
I2C1_Start(); // I2C başlatma sinyali ver
```

```
I2C1_Wr(0xA2); // I2C ile bayt gönder (cihaz adresi + W)
```

```
I2C1_Wr(2); // EEPROM adresine bayt gönder
```

```
I2C1_Wr(0xAA); // Yazılacak veriyi gönder
```

```
I2C1_Stop(); //I2C durdurma sinyali ver
```

```
Delay_100ms();
```

```
I2C1_Start(); // I2C başlatma sinyali ver
```

```
I2C1_Wr(0xA2); // I2C ile bayt gönder (cihaz adresi + W)
```

```
I2C1_Wr(2); // Bayt gönder(veri adresi)
```

```
I2C1_Repeated_Start(); // I2C sinyali tekrar başlat
```

```
I2C1_Wr(0xA3); // bayt gönder ( cihaz adresi + Okuma Biti)
```

```
LATB = I2C1_Rd(0u); // Datayı Oku
```

```
I2C1_Stop(); //I2C durdurma sinyali ver }
```

**Deneve Hazırlık:** Soruların araştırılarak rapor halinde sunulması gerekmektedir.

1-) Bellek nedir, bellek türleri nelerdir?

2-) Belleğin haritalanmasını araştırarak örneklendiriniz.

3-) ROM türlerini araştırınız.

4-) Ön belleğe neden ihtiyaç vardır?

5-) I/O transfer modlarının (Programlanmış G/Ç, Kesme ile başlatılan G/Ç ve Doğrudan Bellek Erişim-DMA) ne olduğunu araştırarak örnekleyiniz.