**#define - const**

**AMAÇ:**  Değişken tanımlarken kullanılan define ve const arasındaki farkı öğrenir.

### **Const ve define kullanımı**

Sık kullanılan değerleri tanımlamada ve kodun anlaşılır hale getirilmesinde değişkenleri kullanırız. Program içinde değeri değişmeyen ifadelerde ise sabitler kullanılır. Sabit tanımlamak için **const** ifadesi değişken tanımlama mantığının başına yazılır.

**Örneğin:**Const double pi=3.14;

Ancak kullanılan her değişken 2KB kapasiteye sahip RAM ‘in hafızasını kullanır. Oldukça az olan RAM kapasitesini verimli kullanmak için define komutu kullanılabilir. Çünkü #define komutunda tanımlanan ifade derleme işlemi esnasında program hafızasına değer olarak yazılır.

Yani **#define** bir programlama ifadesi değildir. Bunun yerine, kod derlenmeden önce bir metin değiştirmesinin gerçekleşmesine neden olan bir makro oluşturur. Örneğin bu kodda:

#define pin 13

void setup() {

pinMode(pin, OUTPUT);

}

void loop() {

digitalWrite(pin, HIGH);

delay(500);

digitalWrite(pin, LOW);

delay(500);

}

IDE, kodu derleyiciye göndermeden önce, bir ön işlemden geçecektir. 'pin' makrosunu tüm örneklerinde 13 tamsayı değerine getirecektir. Yani, bu kod aslında derlenmiş gibi görünüyor:

#define pin 13

void setup() {

pinMode(13, OUTPUT);

}

void loop() {

digitalWrite(13, HIGH);

delay(500);

digitalWrite(13, LOW);

delay(500);

}