

Adresleme Modları

1. Arında Adresleme = Operand doğrudan komutun içinde yer alır.
LDA A, #25

2. İndirgi Adresleme = Bir adrese bir indis eklenerek operand adresi belirlenir. LDA A, \$05, X (Dongü)

3. Doğrudan = Komut operandın adresini doğrudan belirtir. (Veriye ulaşmak için adres bilgisini kullanır.) LDA A, \$90

4. Genişletilmiş = ~~Adres~~ Pointer yapısını oluşturmak için \$0045

5. Anlaşılır = Matematiksel formül veya algoritma kullanılarak hesaplanır.

Arduino Niteleyiciler

1- Global Değişkenler = Kod içerisinde her yerden ulaşılabilir.
Örn: int ledPin;
:

2- Static = Bellekte tutulur, örn: int topla {int a;}
static int b=0;
b=b+a;
return b; }

3- Volatile = RAM'e kaydedilir. Değerin değişmesi kesme (interrupt) işlemi ile yapılır

4- Const = Değiştirilemez değer.

5- #define = Ön işlemci komutu. Bir isim yerine başka isim

6- include = Başka kütüphanelere erişmek için kullanılır.

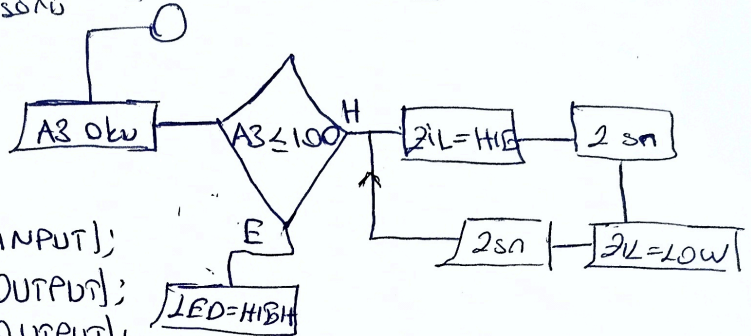
Eğer bir şeye basılıyorsa digital Read
" " değere bakılıyorsa Analog Read

Işık şiddetinin değeri 100 ve altında ise LED yanacak
Üstünde ise 2'li 2'ser saniye aralıkla gidecek

A3 = Işık sensörü

3 = LED

12 = 2'li



```
void setup() {
```

```
  pinMode(A3, INPUT);
```

```
  pinMode(3, OUTPUT);
```

```
  pinMode(12, OUTPUT);
```

```
}
```

```
void loop() {
```

```
  if (analogRead(A3) <= 100) { digitalWrite(3, HIGH); else {
```

```
  * digitalWrite(12, HIGH);
```

```
    delay(2000);
```

```
    digitalWrite(12, LOW);
```

```
    delay(2000);
```

```
    goto xx : 3 ?
```

Buton ile LED kontrolü :

```
int led = 8;
```

```
int buton = 9;
```

```
int butondurum = 0;
```

```
void setup() {
```

```
  pinMode(led, OUTPUT);
```

```
  pinMode(buton, INPUT);
```

```
}
```

```
void loop() {
```

```
  butondurum = digitalRead(buton);
```

```
  if (butondurum == HIGH) {
```

```
    digitalWrite(led, HIGH);
```

```
  else {
```

```
    digitalWrite(led, LOW);
```

```
}
```

Mesafe Ölçümü

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
const int pingPin = 7;
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  lcd.begin(16, 2);
```

```
  lcd.print("Cisim Uzaklık");
```

```
void loop() {
```

```
  lcd.setCursor(0, 1);
```

```
  lcd.print(millis() / 1000);
```

```
  long duration; inches = cm;
```

```
  pinMode(pingPin, OUTPUT);
```

```
  digitalWrite(pingPin, LOW);
```

```
  delay(2000);
```

```
  digitalWrite(pingPin, HIGH);
```

```
  delay(5000);
```

```
  pinMode(pingPin, INPUT)
```

```
  duration = pulseIn(pingPin, HIGH);
```

```
  digitalWrite(13, HIGH);
```


Von Neuman

- * İlk mimari
- * Veriler tek bir yoldan getirilir.

Avantaj

- * Esneklik
- * Basitlik
- * Programlanabilirlik

Dezavantaj

- * Aynı bellekte
- * Performans kısıtlı
- * Güvenlik

Tasarım Mimarileri

1. CISC = Amacı karmaşık işlemleri daha az satırda yazılmasını sağlar.

2. Risc = Basit işlemleri, her bir komutun tek bir saat çevriminde tamamlanması.
* Daha yüksek performans sağlanması.

3. Epic = Talimatlar paralel yürür.

4. DSP = Dijital sinyallerin işlenmesi için.

Program Sayacı (PC) = 16 bit. O anda çalışacak olan komutun adresini tutar.

Komut Kayıtarısı (IR) = 8 bit. O anda çalışan komutu üzerinde bulundurur.

Komut Kod Gözetici = K. kayıtarısından gelen bilgileri sinyal oluşturacak şekilde kodlar.

Akümülatör = A ve B. 8 bit. ALU tarafından. Dataları ve işlem sonuçlarını bulundurur.

Harvard

- * Aynı yollar, aynı bellek.
- * Hızlı.

Avantaj

- * Performans
- * Bellek
- * Güvenlik

Dezavantaj

- * Maliyet
- * Esneklik
- * Karmaşıklık.

Farkları

	<u>CISC</u>	<u>RISC</u>
Komut =	çok	az
İşlem Süresi =	uzun	kısa
Enerji =	Düşük	Yüksek
Danım =	Karmaşık	Basit
Karmaşıklık,		
Kolaylık =	Kolay	Zor

İndis Kayıtarısı (X) =

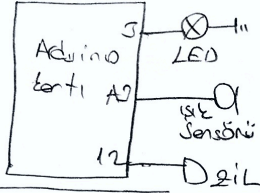
16 bit. Başlık hafıza yerini belirlemek için belirtilen adrese.

Yığın Gözetici (SP) =

16 bit. Hafızadaki herhangi bir hücre adresini bulundurur.

1) Lık siddetinin değeri:

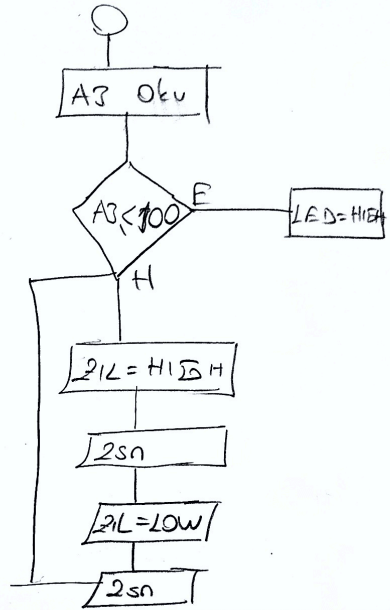
100 ve altında Led yanacak
Üstünde ise Zil ikiser saniye
aralıklarla çalacak.



```
b) void setup() {
    pinMode(12, output);
    pinMode(3, output);
}

void loop() {
    if (AnalogRead(A0) <= 100)
        digitalWrite(3, HIGH);
    else {
        xx digitalWrite(12, HIGH);
        delay(2000);
        digitalWrite(12, LOW);
        delay(2000);
        go to xx;
    }
}
```

a) Akış Diyagramı



2) Bir hafızanın \$1000 - \$1500

aralığında; verileri silen

LDX # \$1000

xx CLR \$00, X

INX

CPX # \$1500

BNE xx

SWI



SOWACODEHK.COM.TR

