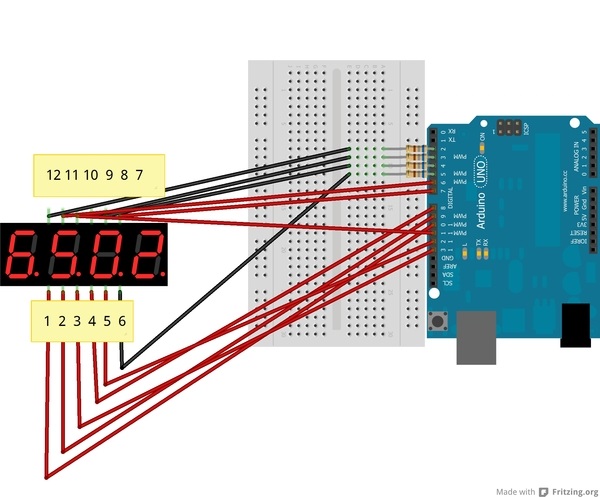
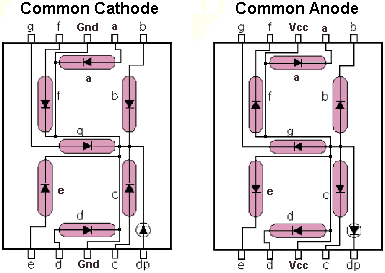
**ARDUİNO İLE 4 DİGİT 7 SEGMENT DİSPLAY UYGULAMASI**

İlk olarak , 7 segment displayler plastik kalıp içine 8 adet LED’in yerleştirilmesiyle oluşturulmuştur. Display’in büyüklüğüne göre birden fazla LED seri olarak bağlanabilmektedir. Aşağıda segment isimleri ve display görülmektedir.

Arduino ile 4 digit 7 segment display uygulamasını gerçekleştireceğiz





Bizim elimizde ki display ortak katottur.

**Arduino ile Display bağlantı tablosu:**

|  |  |  |
| --- | --- | --- |
| Display Pin | Karşılığı | Arduino Pin |
| 12 | D1 | 2 |
| 9 | D2 | 3 |
| 8 | D3 | 4 |
| 6 | D4 | 5 |
| 11 | A | 6 |
| 7 | B | 7 |
| 4 | C | 8 |
| 2 | D | 9 |
| 1 | E | 10 |
| 10 | F | 11 |
| 5 | G | 12 |
| 3 | Nokta | 13 |

4 digit 7 segment display için kütüphaneye de [buradan](http://www.ornekyazilim.com/SevSeg-master.zip) ulaşabilirsiniz.

Kütüphaneleri taslak -> library ekle -> zip kitaplığı ekle bölümünden direk ekleyebilirsiniz.

Kütüphaneleri indirip ekledikten sonra yapmamız gereken bağlantı ve kodlama kalıyor.

4 digit 7 segment displayin bağlantısını resimde gösterildiği gibi yapabilirsiniz.

**Arduino kodumuz aşağıdadır. Sayaç olarak çalışmaktadır.**

#include "SevSeg.h"

SevSeg sevseg; //Instantiate a seven segment controller object

void setup() {

byte numDigits = 4;

// d1 d2 d3 d4

byte digitPins[] = {2, 3, 4, 5};

//a b c d e f g .

byte segmentPins[] = {6, 7, 8, 9, 10, 11, 12, 13};

bool resistorsOnSegments = false; // 'false' means resistors are on digit pins

byte hardwareConfig = COMMON\_CATHODE; // See README.md for options

bool updateWithDelays = false; // Default. Recommended

bool leadingZeros = false; // Use 'true' if you'd like to keep the leading zeros

sevseg.begin(hardwareConfig, numDigits, digitPins, segmentPins, resistorsOnSegments, updateWithDelays, leadingZeros);

sevseg.setBrightness(10);

}

void loop() {

static unsigned long timer = millis();

static int deciSeconds = 0;

if (millis() - timer >= 100) {

timer += 100;

deciSeconds++; // 100 milliSeconds is equal to 1 deciSecond

if (deciSeconds == 10000) { // Reset to 0 after counting for 1000 seconds.

deciSeconds=0;

}

sevseg.setNumber(deciSeconds, 1);

}

//sevseg.setNumber(8888); // Displays '3.141'

//delay(1000);

sevseg.refreshDisplay(); // Must run repeatedly

}