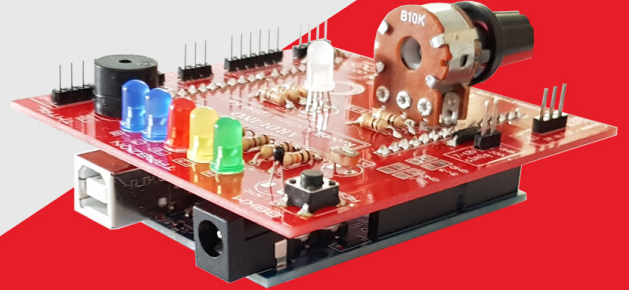
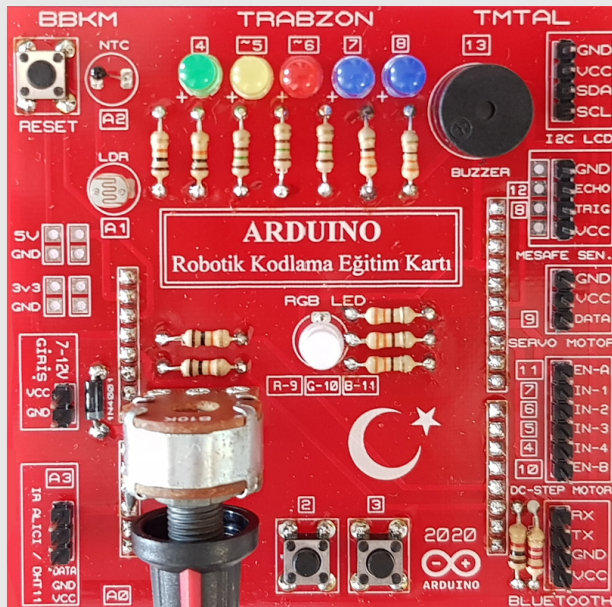


BİLGİSAYAR BİLİMLERİ KODLAMA EĞİTİM MERKEZİ

TRABZON

MESLEKİ VE TEKNİK ANADOLU LİSESİ



ROBOTİK KODLAMA EĞİTİM SETİ UYGULAMA KLAVUZU

- * TEMEL KONU ANLATIM
- * BLOK VE C TABANLI
100 UYGULAMA ÖRNEĞİ

**TRABZON BİLGİSAYAR BİLİMLERİ
VE
KODLAMA EĞİTİMİ MERKEZİ**

**TRABZON MESLEKİ VE TEKNİK
ANADOLU LİSESİ**

**ARDUINO EĞİTİM SETİ
UYGULAMA KLAVUZU**

HAZIRLAYANLAR

Emre UZUN

(Teknik Öğretmen-Bilişim Teknolojileri Alanı)

Mustafa GÜNEŞ

(Teknik Öğretmen-Elektrik-Elektronik Teknolojileri Alanı)

EYLÜL - 2020

İÇİNDEKİLER

A. ARDUINO TANITIM

1. Genel Tanıtım
2. İç Yapısı ve Bileşenleri
3. Geniştirme Elemanları
4. Çeşitleri ve Özellikleri
5. Programlama Türleri ve Geliştirme Ortamları

B. ARDUINO İLE DEVRE KURMA

1. Elektronik Elemanlar ve Devre Teorisi
 - a. Direnç, Diyot, Transistör
 - b. LED, RGB-LED, Buzzer, Buton
 - c. Breadboard, Bağlantı Kablosu
2. Algılayıcı ve Dönüştürücüler
 - a. Mesafe Algılayıcı Elemanlar
 - b. LCD Ekranlar ve I2C Sürücüler
 - c. DC Motor Çeşitleri ve Sürücüleri
 - d. Bluetooth ve Kızılötesi (IR) İletişim Elemanları
3. Örnek Devre Uygulamaları

C. ARDUINO EĞİTİM SETİ TANITIMI

1. Genel Tanıtım ve Kullanım Amacı
2. Çeşitleri ve Özellikleri
3. İç Yapısı ve Bileşenleri
4. Geniştirme Elemanları

D. ARDUINO EĞİTİM SETİ UYGULAMALARI (TEMEL SEVİYE)

1. LED Uygulamaları
2. BUTTON Uygulamaları
3. SERİ İLETİŞİM Uygulamaları
4. POTANSİYOMETRE Uygulamaları
5. RGB-LED Uygulamaları
6. ISI ve IŞIK Uygulamaları

E. ARDUINO EĞİTİM SETİ UYGULAMALARI (İLERİ SEVİYE)

1. Arduino IDE Temel Uygulamaları
2. 16x2 I2C-LCD Ekran Uygulamaları
3. Ultrasonic Mesafe Algılayıcı Uygulamaları
4. Servo Motor Ve Röle Uygulamaları
5. DC Motor Ve Sürücü Uygulamaları
6. Kızılötesi (IR) Uygulamaları
7. Bluetooth (HC-06) Uygulamaları (Windows & Android)
8. 7 Segment Display Modül Uygulamaları

A. ARDUINO TANITIM



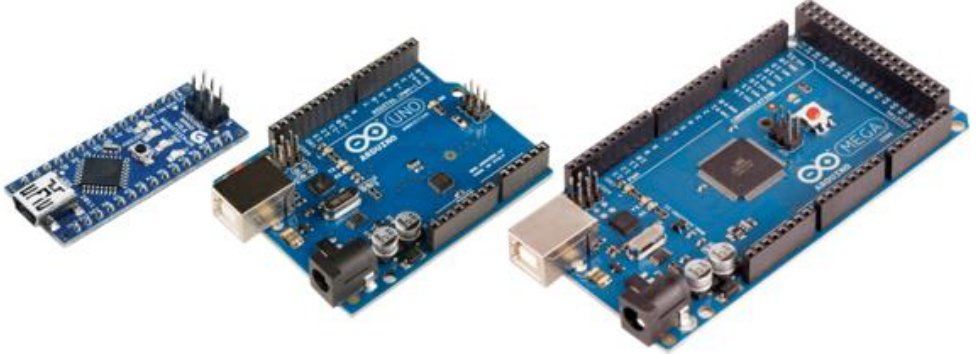
A.1. Genel Tanıtım

ARDUINO, üzerinde **OSİLATÖR**, **REGÜLATÖR**, **RESET**, **USB PROGRAMLAMA PORTU** ve **GİRİŞ/ÇIKIŞ BİRİMLERİ** bulunan bir Mikrodenetleyici geliştirme kartıdır.

ARDUINO, kullanıcıyı elektronik devre ile fazla uğraştırmadan kontrol yeteneği, kolay programlanabilmesi ve bolca kaynak bulunabilmesi sebebiyle günümüzde çok tercih edilir.

ARDUINO ÜRETİCİLERİ, ürettikleri arduinoların devre şemalarını paylaşımına açmıştır. Bu sayede **ARDUINO**'nun klonları üretilmiştir. Fakat programlanması ve kullanımı aynıdır. **Bu sebeple internette çok farklı fiyatlarda ARDUINO'lar bulabilirsiniz.**

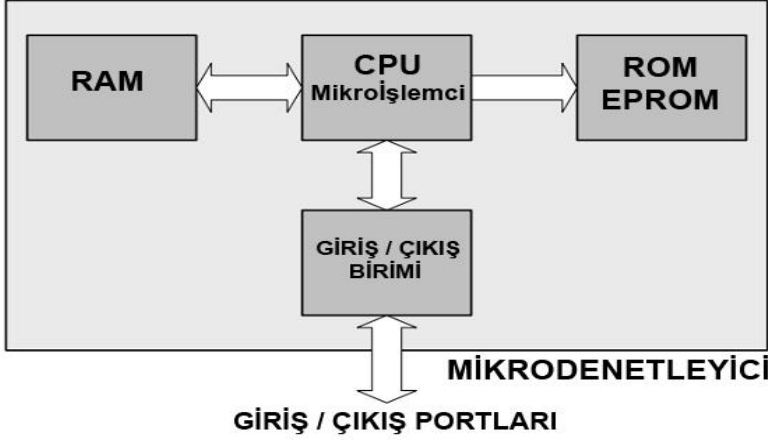
ARDUINO kartlarının donanımında bir adet Atmel AVR mikrodenetleyici (ATmega328, ATmega2560 gibi) ve programlama ile diğer devrelere bağlantı için gerekli yan elemanlar bulunur. Her Arduino kartında en azından bir 5 voltluk regüle ünitesi ve bir 16MHz osilatör vardır. Arduino kartlarında programlama için harici bir programlayıcıya ihtiyaç duyulmaz, çünkü karttaki mikrodenetleyiciye önceden bir bootloader programı yüklenmiştir.



ARDUINO kütüphaneleri ile kolaylıkla programlama yapılabilir, analog ve digital sinyalleri işleyebilir, algılayıcılardan gelen sinyalleri kullanarak, çevre birimleri ile etkileşim içerisinde olan robotlar ve sistemler tasarlanabilmektedir. Tasarlanan projeye özgü olarak dış dünyaya hareket, ses, ışık gibi tepkiler oluşturulabilmektedir.

ARDUINO nun farklı ihtiyaçlara çözüm üretebilmek için tasarlanmış çeşitli eklentileri ve modülleri mevcuttur. Bu modüller kullanılarak proje geliştirilebilmektedir.

A.2. İç Yapısı ve Bileşenleri



Bu noktada, **Mikroişlemci ve Mikrodenetleyici** kavramlarını detaylandırmak gerekmektedir. Mikroişlemci ile Mikrodenetleyici birbirine karıştırılmamalıdır. Mikrodenetleyiciler, yapılarında Mikroişlemcileri içermektedirler. Mikroişlemcilerin yapısında CPU, ön bellek ve I/O portları olmasına karşın, Mikrodenetleyicilerde ayrıca seri ve paralel portlar, sayıcılar ve çevirciler bulunmaktadır. EEPROM, RAM ve ROM birimleri, Mikrodenetleyicilerde dahili olmasına karşın, Mikroişlemcilerde harici birimlerdir. Mikrodenetleyiciler, gerçek zamanlı uygulamalarda daha başarılı çalışmaktadırlar. Mikrodenetleyiciler çok küçük boyutlarda ve daha az enerji harcayarak çalışabilmektedirler. Bunun yanı sıra, Mikroişlemciler aynı anda çoklu işlem yapabilirken, Mikrodenetleyiciler aynı anda tek bir işlem yapabilmektedirler.

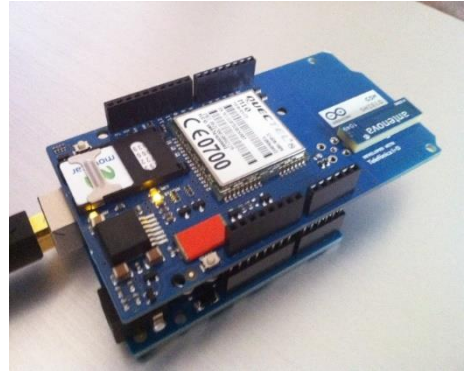
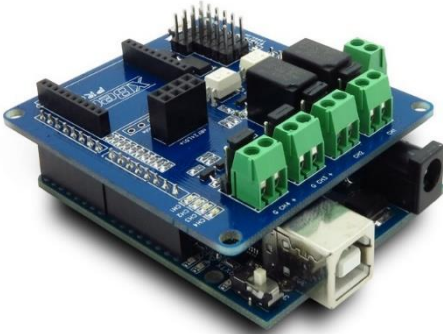
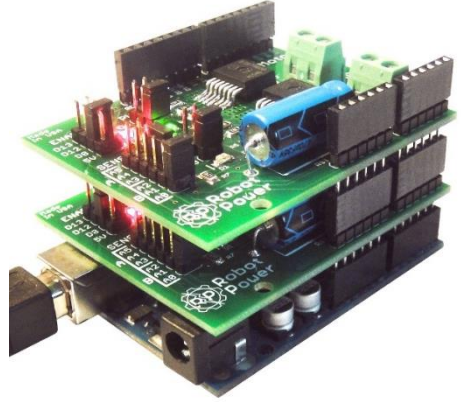
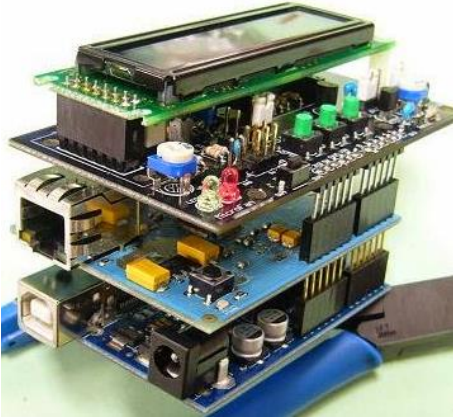


A.3. Geniřletme Elemanları



ARDUINO da farklı iřlevler iin retilmiř zel geliřtirme kartları bulunmaktadır. Bu kartlara **SHIELD** adı verilir. **ARDUINO** ile alıřabilen ve Arduino'yu geliřtirmeyi saėlayan ve zerine takılabilen paralardır. Diėer bir adıyla modl olarak da bilinir.

ARDUINO'yu geliřtirmek ve yeni donanımlar ekleyebilmek iin bir adet **SHIELD** kullanılabilceėi gibi, birden ok **SHIELD** de kullanılabilir. Birden fazla **SHIELD** kullanılırken, **SHIELD** lerin pinlerine dikkat etmek gerekmektedir. **SHIELD** lerin Arduino zerindeki aynı pini kullanmamasına zen gsterilmelidir. Byle bir durum varsa, **SHIELD** ler birlikte kullanılmamalıdır. Ařaėıda bazı rnek **SHIELD** kullanımları gsterilmiřtir.



A.4. Çeşitleri ve Özellikleri



ARDUINO kartlarının birçok çeşidi mevcuttur. Temel olarak bütün kartlar benzer bileşenlere sahip olmakla beraber, Mikrodenetleyici modelleri, Giriş/Çıkış pinleri ve dahili modüllerin sayısı, ebat ve çalışma gerilimleri gibi farklılıklara sahiptirler. En çok bilinen ve kullanılan çeşitleri aşağıda belirtilmiştir.

1. Arduino NANO

Arduino NANO, ATmega328 mikrodenetleyici (Nano 3.x) veya Atmega168 (Nano 2.x) mikrodenetleyici barındıran, küçük ve breadboard dostu bir karttır.



Arduino Nano V3 ön yüz



Arduino Nano V3 arka yüz

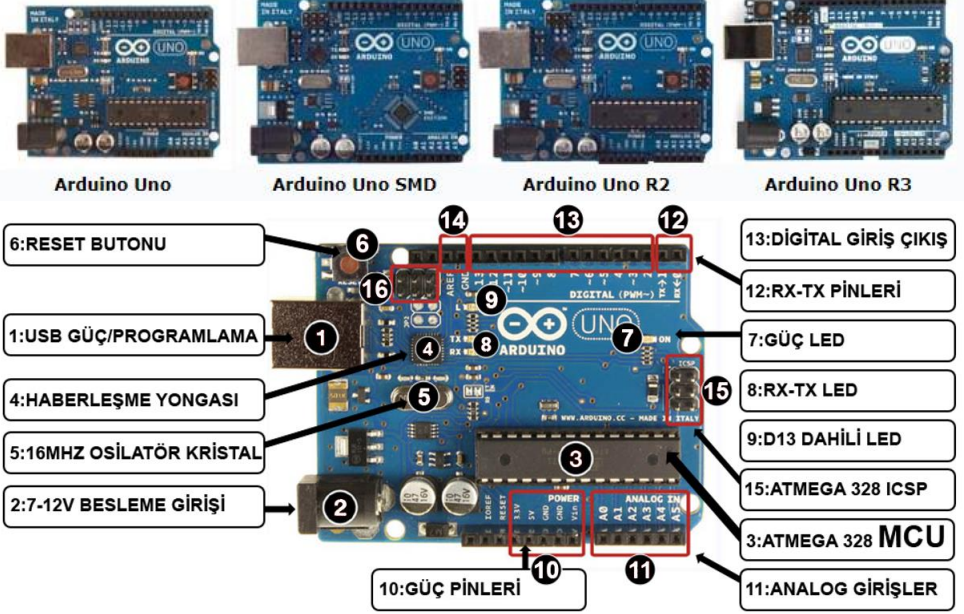
Teknik Özellikleri

Mikrodenetleyici	Arduino Nano V3 Te Atmega328 Önceki Versiyonlarda Atmega168
Çalışma Gerilimi	+5 V DC
Tavsiye Edilen Besleme Gerilimi	7 - 12 V DC
Dijital Giriş / Çıkış Pinleri	14 ADET / 6 ADET PWM Çıkış
Analog Giriş Pinleri	8 ADET
Giriş / Çıkış Pini Başına Düşen DC Akım	40 mA
Flash Hafıza	Atmega328 32KB-Atmega168 16KB
SRAM	Atmega328 2KB-Atmega168 1KB
EEPROM	Atmega328 1KB-Atmega168 512B
Saat Frekansı	16 MHz
Boyutları	18 mm X 45 mm
Ağırlık	5 Gram

2. Arduino UNO



Arduino UNO, ATmega328 mikrodenetleyici içeren bir karttır. Arduino'nun en yaygın kullanılan kartı olduğu söylenebilir. Arduino UNO, ilk modelinden sonra, Arduino UNO R2, Arduino UNO SMD ve son olarak Arduino UNO R3 çıkmıştır. Arduino'nun kardeş markası olan Genuino markasını taşıyan Genuino UNO kartı ile tamamen aynı özelliklere sahiptir.



Teknik Özellikleri

Mikrodenetleyici	Atmega328
Çalışma Gerilimi	+5 V DC
Tavsiye Edilen Besleme Gerilimi	7 - 12 V DC
Dijital Giriş / Çıkış Pinleri	14 ADET / 6 ADET PWM Çıkış
Analog Giriş Pinleri	6 ADET
Giriş / Çıkış Pini Başına Düşen DC Akım	40 mA
Flash Hafıza	Atmega328 32KB
SRAM	Atmega328 2KB
EEPROM	Atmega328 1KB
Saat Frekansı	16 MHz

3. Arduino MEGA

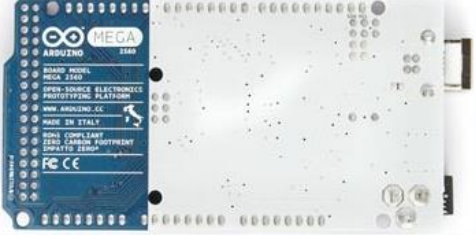


Arduino MEGA, ATmega2560 mikrodenetleyici içeren bir Arduino kartıdır. Arduino Uno 'dan sonra en çok tercih edilen Arduino kartı olduğu söylenebilir. Arduino'nun kardeş markası olan Genuino markasını taşıyan Genuino Mega 2560 kartı ile tamamen aynı özelliklere sahiptir.

Genuino markası, Avrupa'da geçerli olmakla birlikte, Amerika'da Arduino olarak kullanılmaktadır.



Arduino Mega 2560 R3 ön yüz



Arduino Mega 2560 R3 arka yüz

<p>AREF</p> <p>D13 : PB 7</p> <p>D12 : PB 6</p> <p>D11 : PB 5</p> <p>D10 : PB 4</p> <p>D9 : PH 6</p> <p>D8 : PH 5</p> <p>D7 : PH 4</p> <p>D6 : PH 3</p> <p>D5 : PE 3</p> <p>D4 : PG 5</p> <p>D3 : PE 4</p> <p>D2 : PE 0</p> <p>D0 :</p> <p>D14 : PJ 1</p> <p>D15 : PJ 0</p> <p>D16 : PH 0</p> <p>D17 : PH 1</p> <p>D18 : PD 3</p> <p>D19 : PD 2</p> <p>D20 : PD 1</p> <p>D21 :</p> <p>D22 :</p> <p>D23 :</p> <p>D25 :</p> <p>D27 :</p> <p>D29 :</p> <p>D31 :</p> <p>D33 :</p> <p>D35 :</p> <p>D37 :</p> <p>D39 :</p> <p>D41 :</p> <p>D43 :</p> <p>D45 :</p> <p>D47 :</p> <p>D49 :</p> <p>D51 :</p> <p>D53 :</p>	<p>GND</p> <p>PWM T0A, Pin Int 7</p> <p>PWM T1B, Pin Int 6</p> <p>PWM T1A, Pin Int 5</p> <p>PWM T2A, Pin Int 4</p> <p>PWM T2B</p> <p>PWM T4C</p> <p>PWM T4B</p> <p>PWM T3A</p> <p>PWM T3B</p> <p>PWM T3C, INT5</p> <p>PWM T3B, INT4</p> <p>USART0 TX</p> <p>USART0 RX, pin Int 8</p> <p>USART3 TX, Pin Int 10</p> <p>USART3 RX, Pin Int 9</p> <p>USART2 TX</p> <p>USART2 RX</p> <p>USART1 TX, Ext Int 3</p> <p>USART1 RX, Ext Int 2</p> <p>I2C SDA, Ext Int 1</p> <p>I2C SCL, Ext Int 0</p> <p>Ext Memory addr bit 0</p> <p>Ext Memory addr bit 2</p> <p>Ext Memory addr bit 4</p> <p>Ext Memory addr bit 6</p> <p>Ext Memory addr bit 15</p> <p>Ext Memory addr bit 13</p> <p>Ext Memory addr bit 11</p> <p>Ext Memory addr bit 9</p> <p>RD Ext Mem</p> <p>*PWM 5C</p> <p>*PWM 5A</p> <p>ICP T5</p> <p>SPI MISO</p> <p>SPI SCK</p> <p>RESET</p> <p>3.3V</p> <p>5V</p> <p>GND</p> <p>Vin</p> <p>AIN0</p> <p>AIN1</p> <p>AIN2</p> <p>AIN3</p> <p>AIN4</p> <p>AIN5</p> <p>AIN6</p> <p>AIN7</p> <p>PK 0</p> <p>PK 1</p> <p>PK 2</p> <p>PK 3</p> <p>PK 4</p> <p>PK 5</p> <p>PK 6</p> <p>PK 7</p> <p>(D62)</p> <p>(D63)</p> <p>(D64)</p> <p>(D65)</p> <p>(D66)</p> <p>(D67)</p> <p>(D68)</p> <p>(D69)</p> <p>(Pin Int 16)</p> <p>(Pin Int 17)</p> <p>(Pin Int 18)</p> <p>(Pin Int 19)</p> <p>(Pin Int 20)</p> <p>(Pin Int 21)</p> <p>(Pin Int 22)</p> <p>(Pin Int 23)</p> <p>PA 0</p> <p>PA 2</p> <p>PA 4</p> <p>PA 6</p> <p>PA 7</p> <p>PA 8</p> <p>PA 9</p> <p>PA 10</p> <p>PA 11</p> <p>PA 12</p> <p>PA 13</p> <p>PA 14</p> <p>PA 15</p> <p>PA 16</p> <p>PA 17</p> <p>PA 18</p> <p>PA 19</p> <p>PA 20</p> <p>PA 21</p> <p>PA 22</p> <p>PA 23</p> <p>PA 24</p> <p>PA 25</p> <p>PA 26</p> <p>PA 27</p> <p>PA 28</p> <p>PA 29</p> <p>PA 30</p> <p>PA 31</p> <p>PA 32</p> <p>PA 33</p> <p>PA 34</p> <p>PA 35</p> <p>PA 36</p> <p>PA 37</p> <p>PA 38</p> <p>PA 39</p> <p>PA 40</p> <p>PA 41</p> <p>PA 42</p> <p>PA 43</p> <p>PA 44</p> <p>PA 45</p> <p>PA 46</p> <p>PA 47</p> <p>PA 48</p> <p>PA 49</p> <p>PA 50</p> <p>PA 51</p> <p>PA 52</p> <p>PA 53</p> <p>PC 0</p> <p>PC 1</p> <p>PC 2</p> <p>PC 3</p> <p>PC 4</p> <p>PC 5</p> <p>PC 6</p> <p>PC 7</p> <p>PC 8</p> <p>PC 9</p> <p>PC 10</p> <p>PC 11</p> <p>PC 12</p> <p>PC 13</p> <p>PC 14</p> <p>PC 15</p> <p>PC 16</p> <p>PC 17</p> <p>PC 18</p> <p>PC 19</p> <p>PC 20</p> <p>PC 21</p> <p>PC 22</p> <p>PC 23</p> <p>PC 24</p> <p>PC 25</p> <p>PC 26</p> <p>PC 27</p> <p>PC 28</p> <p>PC 29</p> <p>PC 30</p> <p>PC 31</p> <p>PC 32</p> <p>PC 33</p> <p>PC 34</p> <p>PC 35</p> <p>PC 36</p> <p>PC 37</p> <p>PC 38</p> <p>PC 39</p> <p>PC 40</p> <p>PC 41</p> <p>PC 42</p> <p>PC 43</p> <p>PC 44</p> <p>PC 45</p> <p>PC 46</p> <p>PC 47</p> <p>PC 48</p> <p>PC 49</p> <p>PC 50</p> <p>PC 51</p> <p>PC 52</p> <p>PC 53</p> <p>PD 0</p> <p>PD 1</p> <p>PD 2</p> <p>PD 3</p> <p>PD 4</p> <p>PD 5</p> <p>PD 6</p> <p>PD 7</p> <p>PD 8</p> <p>PD 9</p> <p>PD 10</p> <p>PD 11</p> <p>PD 12</p> <p>PD 13</p> <p>PD 14</p> <p>PD 15</p> <p>PD 16</p> <p>PD 17</p> <p>PD 18</p> <p>PD 19</p> <p>PD 20</p> <p>PD 21</p> <p>PD 22</p> <p>PD 23</p> <p>PD 24</p> <p>PD 25</p> <p>PD 26</p> <p>PD 27</p> <p>PD 28</p> <p>PD 29</p> <p>PD 30</p> <p>PD 31</p> <p>PD 32</p> <p>PD 33</p> <p>PD 34</p> <p>PD 35</p> <p>PD 36</p> <p>PD 37</p> <p>PD 38</p> <p>PD 39</p> <p>PD 40</p> <p>PD 41</p> <p>PD 42</p> <p>PD 43</p> <p>PD 44</p> <p>PD 45</p> <p>PD 46</p> <p>PD 47</p> <p>PD 48</p> <p>PD 49</p> <p>PD 50</p> <p>PD 51</p> <p>PD 52</p> <p>PD 53</p> <p>PE 0</p> <p>PE 1</p> <p>PE 2</p> <p>PE 3</p> <p>PE 4</p> <p>PE 5</p> <p>PE 6</p> <p>PE 7</p> <p>PE 8</p> <p>PE 9</p> <p>PE 10</p> <p>PE 11</p> <p>PE 12</p> <p>PE 13</p> <p>PE 14</p> <p>PE 15</p> <p>PE 16</p> <p>PE 17</p> <p>PE 18</p> <p>PE 19</p> <p>PE 20</p> <p>PE 21</p> <p>PE 22</p> <p>PE 23</p> <p>PE 24</p> <p>PE 25</p> <p>PE 26</p> <p>PE 27</p> <p>PE 28</p> <p>PE 29</p> <p>PE 30</p> <p>PE 31</p> <p>PE 32</p> <p>PE 33</p> <p>PE 34</p> <p>PE 35</p> <p>PE 36</p> <p>PE 37</p> <p>PE 38</p> <p>PE 39</p> <p>PE 40</p> <p>PE 41</p> <p>PE 42</p> <p>PE 43</p> <p>PE 44</p> <p>PE 45</p> <p>PE 46</p> <p>PE 47</p> <p>PE 48</p> <p>PE 49</p> <p>PE 50</p> <p>PE 51</p> <p>PE 52</p> <p>PE 53</p> <p>PJ 0</p> <p>PJ 1</p> <p>PJ 2</p> <p>PJ 3</p> <p>PJ 4</p> <p>PJ 5</p> <p>PJ 6</p> <p>PJ 7</p> <p>PJ 8</p> <p>PJ 9</p> <p>PJ 10</p> <p>PJ 11</p> <p>PJ 12</p> <p>PJ 13</p> <p>PJ 14</p> <p>PJ 15</p> <p>PJ 16</p> <p>PJ 17</p> <p>PJ 18</p> <p>PJ 19</p> <p>PJ 20</p> <p>PJ 21</p> <p>PJ 22</p> <p>PJ 23</p> <p>PJ 24</p> <p>PJ 25</p> <p>PJ 26</p> <p>PJ 27</p> <p>PJ 28</p> <p>PJ 29</p> <p>PJ 30</p> <p>PJ 31</p> <p>PJ 32</p> <p>PJ 33</p> <p>PJ 34</p> <p>PJ 35</p> <p>PJ 36</p> <p>PJ 37</p> <p>PJ 38</p> <p>PJ 39</p> <p>PJ 40</p> <p>PJ 41</p> <p>PJ 42</p> <p>PJ 43</p> <p>PJ 44</p> <p>PJ 45</p> <p>PJ 46</p> <p>PJ 47</p> <p>PJ 48</p> <p>PJ 49</p> <p>PJ 50</p> <p>PJ 51</p> <p>PJ 52</p> <p>PJ 53</p> <p>PF 0</p> <p>PF 1</p> <p>PF 2</p> <p>PF 3</p> <p>PF 4</p> <p>PF 5</p> <p>PF 6</p> <p>PF 7</p> <p>PF 8</p> <p>PF 9</p> <p>PF 10</p> <p>PF 11</p> <p>PF 12</p> <p>PF 13</p> <p>PF 14</p> <p>PF 15</p> <p>PF 16</p> <p>PF 17</p> <p>PF 18</p> <p>PF 19</p> <p>PF 20</p> <p>PF 21</p> <p>PF 22</p> <p>PF 23</p> <p>PF 24</p> <p>PF 25</p> <p>PF 26</p> <p>PF 27</p> <p>PF 28</p> <p>PF 29</p> <p>PF 30</p> <p>PF 31</p> <p>PF 32</p> <p>PF 33</p> <p>PF 34</p> <p>PF 35</p> <p>PF 36</p> <p>PF 37</p> <p>PF 38</p> <p>PF 39</p> <p>PF 40</p> <p>PF 41</p> <p>PF 42</p> <p>PF 43</p> <p>PF 44</p> <p>PF 45</p> <p>PF 46</p> <p>PF 47</p> <p>PF 48</p> <p>PF 49</p> <p>PF 50</p> <p>PF 51</p> <p>PF 52</p> <p>PF 53</p>	<p>5V</p> <p>Ext Memory addr bit 1</p> <p>Ext Memory addr bit 3</p> <p>Ext Memory addr bit 5</p> <p>Ext Memory addr bit 7</p> <p>Ext Memory addr bit 14</p> <p>Ext Memory addr bit 12</p> <p>Ext Memory addr bit 10</p> <p>Ext Memory addr bit 8</p> <p>ALE Ext Mem</p> <p>Wr Ext Mem</p> <p>PL 6</p> <p>*PWM 5B</p> <p>T5 external counter</p> <p>ICP T4</p> <p>SPI MOSI</p> <p>SPI SS</p>
---	--	---

Mikrodenetleyici	Atmega2560
Çalışma Gerilimi	+5 V DC
Tavsiye Edilen Besleme Gerilimi	7 - 12 V DC
Dijital Giriş / Çıkış Pinleri	54 ADET / 15 ADET PWM Çıkış
Analog Giriş Pinleri	16 ADET
Giriş / Çıkış Pini Başına Düşen DC Akım	40 mA
Flash Hafıza	Atmega2560 256KB
SRAM	Atmega2560 8KB
EEPROM	Atmega2560 4KB
Saat Frekansı	16 MHz

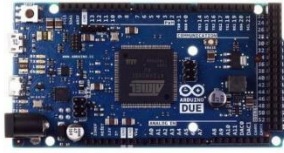
4. Diğer Arduino Çeşitleri



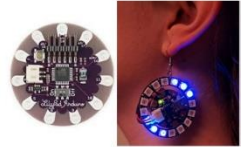
Arduino Uno



Arduino Leonardo



Arduino Due



Arduino LilyPad



Arduino Yun



Arduino Micro



Arduino Robot



Arduino Fio



Arduino Esplora



Arduino Mini



Arduino Nano



Arduino Pro Mini

A.5. Programlama Türleri ve Geliştirme Ortamları

ARDUINO'nun programlanması, temel olarak iki ana başlıkta incelenebilir. Kod yazma süreçlerine dahil olmadan, BLOCK tabanlı algoritma geliştirme ortamlarından, ARDUINO programlamaya uyumlu olan araçları kullanılabilir. Eğitim-Öğretim süreçleri bağlamında kullanışlı ve öğrenci veya yeni öğrenen yetişkinlerde ilgi çekici ve pratik olarak hızlı netice vermesi, kolay öğrenilmesi bakımından olumlu olarak değerlendirilmesine rağmen, kullanıcının seviyesi arttıkça ve yapılan uygulamaların karmaşıklık düzeyi yükseldikçe, sorunlar oluşturabilmektedir. Başlangıç düzeyi için uygun olmakla beraber, ileri seviyede kaçınılmaz olan sonuç, diğer tüm Mikrodenetleyici türevlerinde olduğu gibi, üretici tarafından geliştirilen, kod tabanlı editör ortamlarına geçiş yapmaktır.

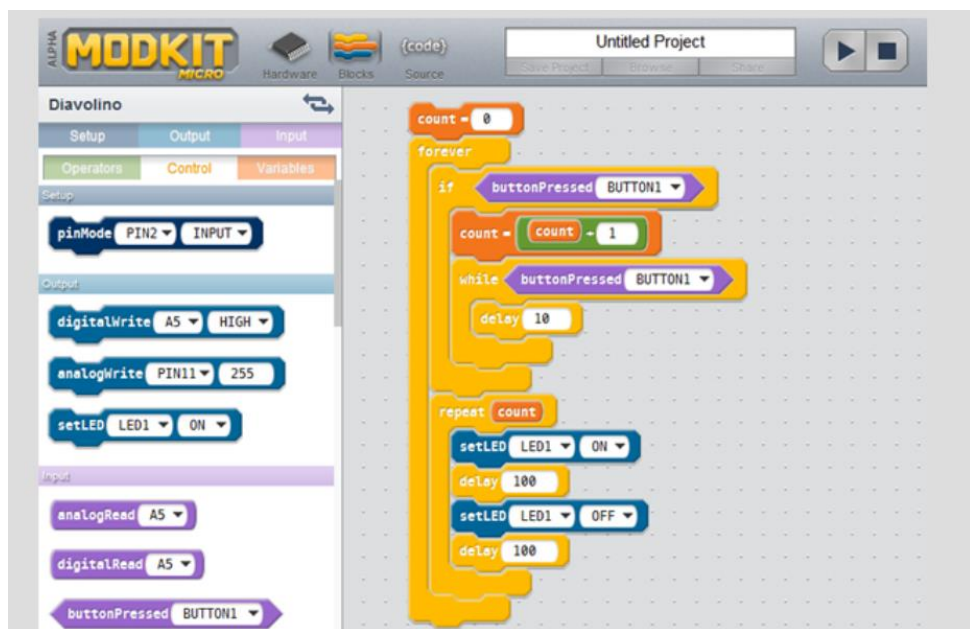
Bu durum, ARDUINO ve diğer tüm Mikrodenetleyici türevlerinde geçerli olan temel programlama yaklaşımıdır.

BLOCK tabanlı ARDUINO programlama araçlarının, tamamında bulunmamasıyla birlikte, bir diğer dikkat edilesi gereken noktası da, ARDUINO üzerine yazılan program kodlarının kalıcılığıdır. Birçok BLOCK tabanlı geliştirme ortamında yazılan kodlar, PC-ARDUINO iletişimi kesildiğinde, çalışmaya devam etmemektedir. Bu bilgiler ışığında ARDUINO üzerine programlama yapılabilecek birkaç BLOCK tabanlı geliştirme ortamı aşağıda listelenmiştir.

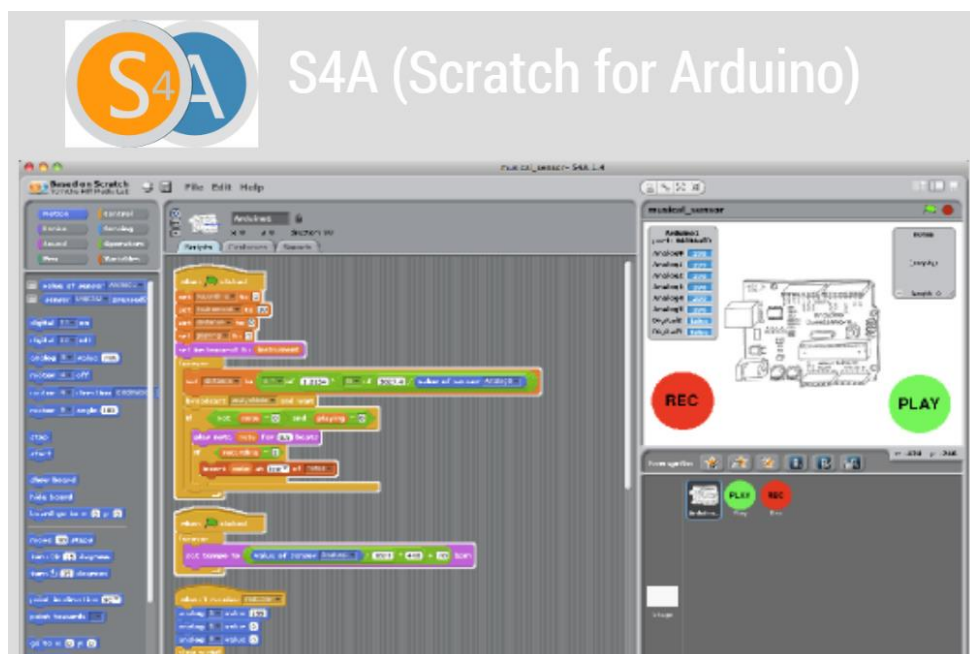
1. ARDUBLOCK

The screenshot displays the ARDUBLOCK programming environment. The interface includes a top menu bar with buttons for 'New', 'Save', 'Save As', 'Open', 'Upload to Arduino', and 'Serial Monitor'. On the left, there is a vertical toolbar with various categories: Control, Pins, Tests, Math Operators, Variables/Constants, Generic Hardware, Communication, SCoop (Multitask), Storage, Networking, Code Blocks, TinkerKit, DFRobot, Sseed Studio Grove, DuinoEDU Grove Add, Adafruit Motorsshield, Makeblock, Insect Bot, 4Drawing, and LittleBits. The main workspace is divided into three sections: 'setup', 'loop', and 'program'. The 'setup' section contains two 'set decimal number variable' blocks for 'celsius' and 'fahrenheit', both with a value of 0. The 'loop' section contains a 'set decimal number variable' block for 'celsius', followed by an 'analog pin' block (pin 0) with a multiplier of 0.00488 and a divider of 0.5. This is followed by a 'serial println' block for 'celsius'. The 'program' section contains a 'set decimal number variable' block for 'fahrenheit' with a calculation: $celsius \times 9 + 5 + 32$. This is followed by a 'serial println' block for 'fahrenheit'. A 'test' block checks if 'fahrenheit' is greater than or equal to 80.0. If true, it sets digital pins 9 and 11 to HIGH. If false, it sets digital pins 9 and 11 to LOW. Finally, a 'delay MILLIS' block is set to 1000. The bottom of the interface has buttons for 'Save as image...', 'Go to Web Site', and a version number 'v 20140704 (beta)'.

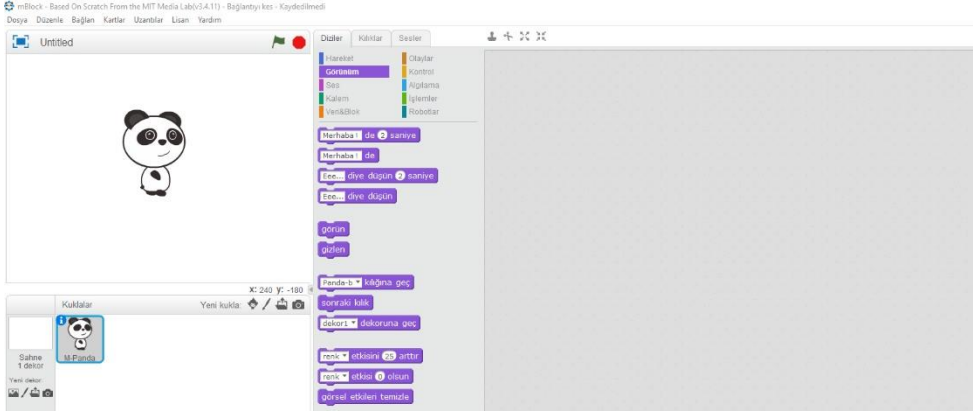
2. MODKIT



3. S4A (SCRATCH FOR ARDUINO)



4. MBLOCK



MBLOCK, birkaç farklı robotik ürünü programlayabilmesi, kullanım kolaylığı ve kodlama uyumluluğu bakımından, BLOCK tabanlı ARDUINO programlamada en uygun araç olma özelliğine sahiptir. Ayrıca, ARDUINO uygulamalarına başlamadan, oyun tabanlı temel kod yazma becerisi ve programlama mantığı kazandırmak amacıyla kullanım özellikleri olan bir BLOCK tabanlı geliştirme ortamıdır. Program kurulumu gerçekleştirildikten sonra, ARDUINO bağlantısı ve programlaması için gerekli ayarlamalar, aşağıdaki biçimdedir.

<http://www.mblock.cc>



mBlock



mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Bağlantı kes - Kaydedilmedi

Dosya Düzenle Bağlan Kartlar Uzantılar **Lisan** Yardım

Untitled

English
正體中文
簡體中文
Српски
Русский
Português
Polski
Nederlands
मराठी
हिन्दी
Türkçe

Diziler Kılıklar Sesler

Hareket Görünüm Ses Kalem Veri&Blok Olaylar Kontrol Algılama İşlemler Robotlar

Merhaba! de 2 saniye
Merhaba! de
Eee... diye düşün 2 saniye
Eee... diye düşün

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Bağlantı kes - Kaydedilmedi

Dosya Düzenle Bağlan Kartlar **Uzantılar** Lisan Yardım

Untitled

Uzantıları Yönet Ctrl+Shift+T
Uzantıları Eski haline Getir
Zulayı Temizle
Smart Servo Tools
Arduino
Smart Servo
Communication

Diziler Kılıklar Sesler

Hareket Görünüm Ses Kalem Veri&Blok Olaylar Kontrol Algılama İşlemler Robotlar

Merhaba! de 2 saniye
Merhaba! de
Eee... diye düşün 2 saniye

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Bağlantı kes - Kaydedilmedi

Dosya Düzenle Bağlan **Kartlar** Uzantılar Lisan Yardım

Untitled

Arduino Uno
Arduino Nano (mega328)
Arduino Mega 1280
Arduino Mega 2560

Diziler Kılıklar Sesler

Hareket Görünüm Ses Kalem Veri&Blok Olaylar Kontrol Algılama İşlemler Robotlar

mBlock - Based On Scratch From the MIT Media Lab(v3.4.11) - Bağlantı kes - Kaydedilmedi

Dosya **Düzenle** Bağlan Kartlar Uzantılar Lisan Yardım

Silineni Geri Al
Sahneyi Gizle
Küçük Sahne
Arduino Kipi

Arduino Programı

geri al Arduinoya Yükle

```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7
8 void setup() {
9 }
```


Diziler

Hareket
Görünüm
Ses
Kalem
Veri&Blok

Olaylar
Kontrol
Algılama
İşlemler
Robotlar

10 adım git

15 derece dön

15 derece dön

90 yönüne dön

'ye doğru dön

x: -4 y: 11 noktasına git

fare oku 'na git

1 sn.de x: -4 y: 11 a süzül

Arduino Programı

sürekli tekrarla

4 sayisal pini YÜKSEK yap

1 saniye bekle

4 sayisal pini DÜŞÜK yap

1 saniye bekle

Arduino Programı

sürekli tekrarla

4 sayisal pini YÜKSEK yap

1 saniye bekle

4 sayisal pini DÜŞÜK yap

1 saniye bekle

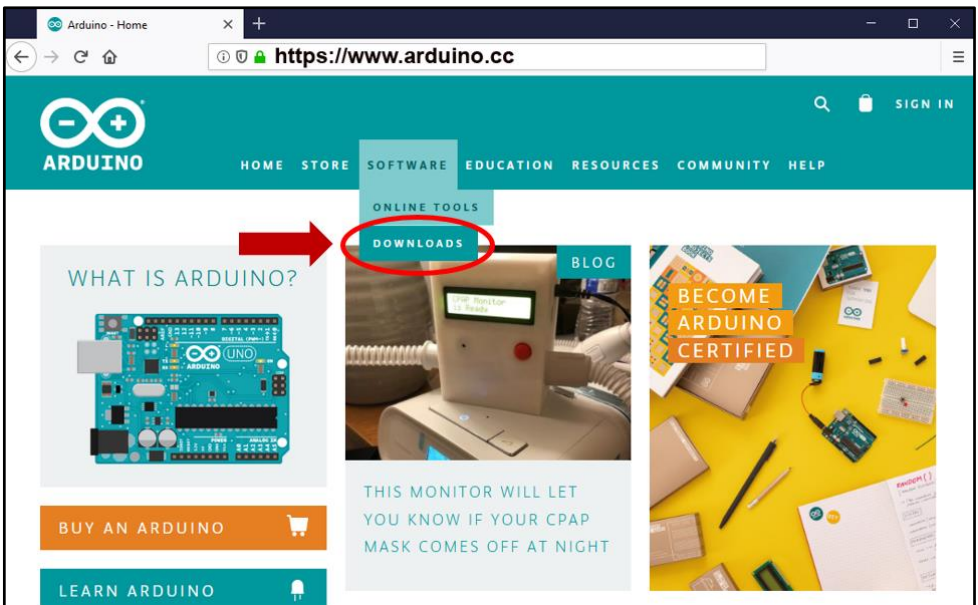
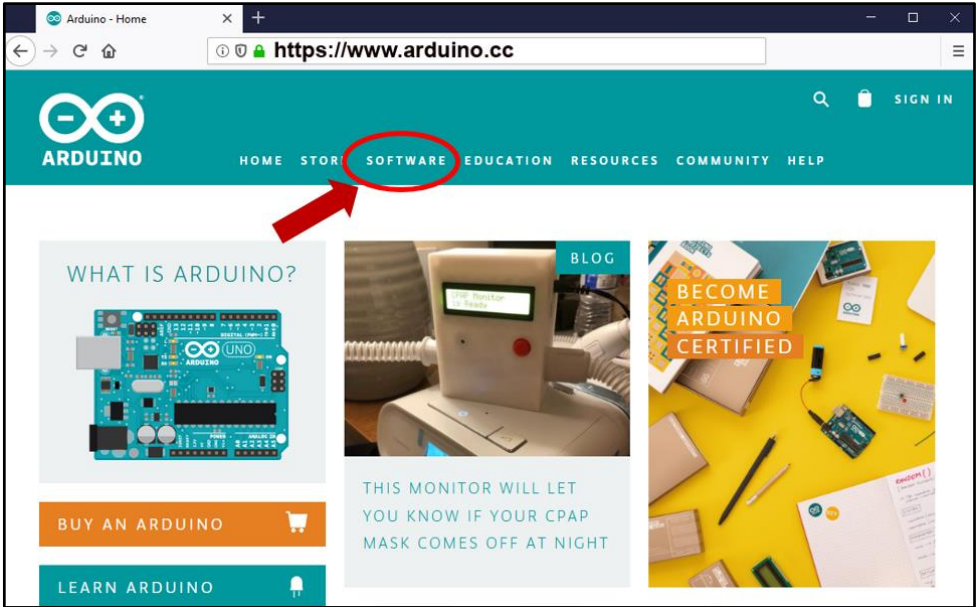
geri al **Arduinoya Yükle**

```

1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 double angle_rad = PI/180.0;
6 double angle_deg = 180.0/PI;
7
8 void setup() {
9   pinMode(4,OUTPUT);
10 }
11
12 void loop() {
13   digitalWrite(4,1);
14   _delay(1);
15   digitalWrite(4,0);
16   _delay(1);
17   _loop();
18 }
19
20 void _delay(float seconds) {
21   long endTime = millis() + seconds * 1000;
22   while(millis() < endTime)_loop();
23 }
24
25 void _loop() {
26 }

```

5. ARDUINO IDE



Arduino - Home

https://www.arduino.cc

HOME STORE SOFTWARE **EDU** RESOURCES COMMUNITY HELP

SEARCH SIGN IN

Download the Arduino IDE



ARDUINO 1.8.9

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

- Windows installer, for Windows XP and up
- Windows ZIP file for non admin install
- Mac OS X 10.8 Mountain Lion or newer
- Linux 32 bits
- Linux 64 bits
- Linux ARM 32 bits
- Linux ARM 64 bits

Arduino - Home

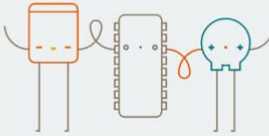
https://www.arduino.cc

HOME STORE SOFTWARE **EDU** RESOURCES COMMUNITY HELP

SEARCH SIGN IN

Contribute to the Arduino Software

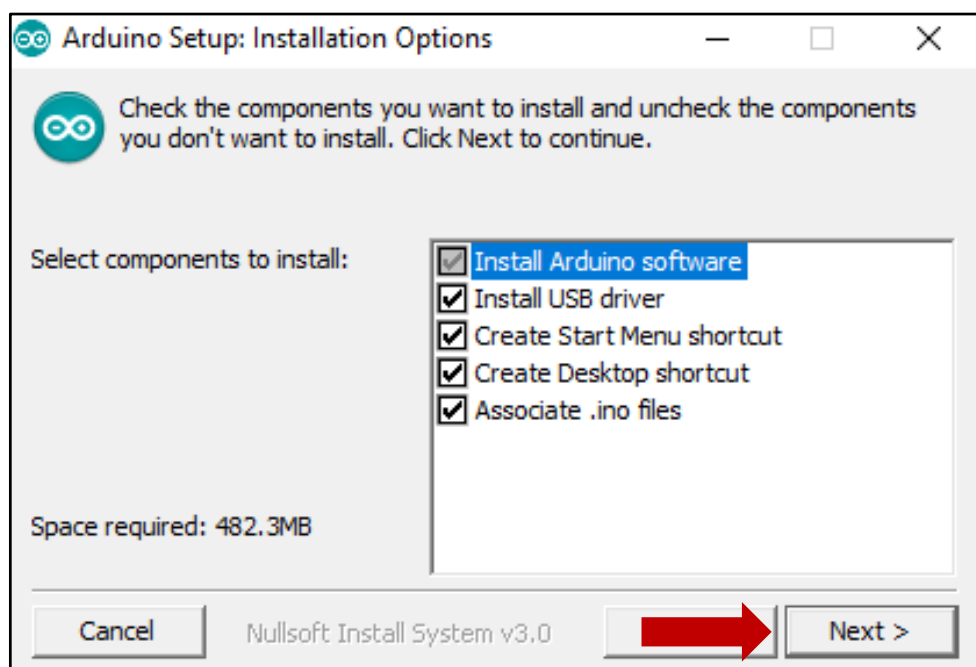
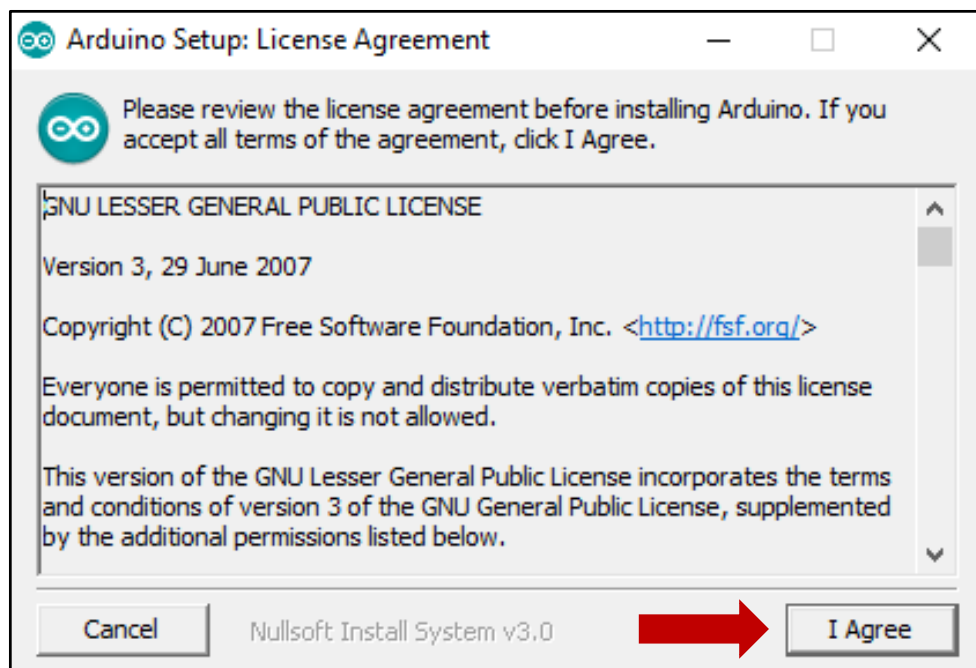
Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)

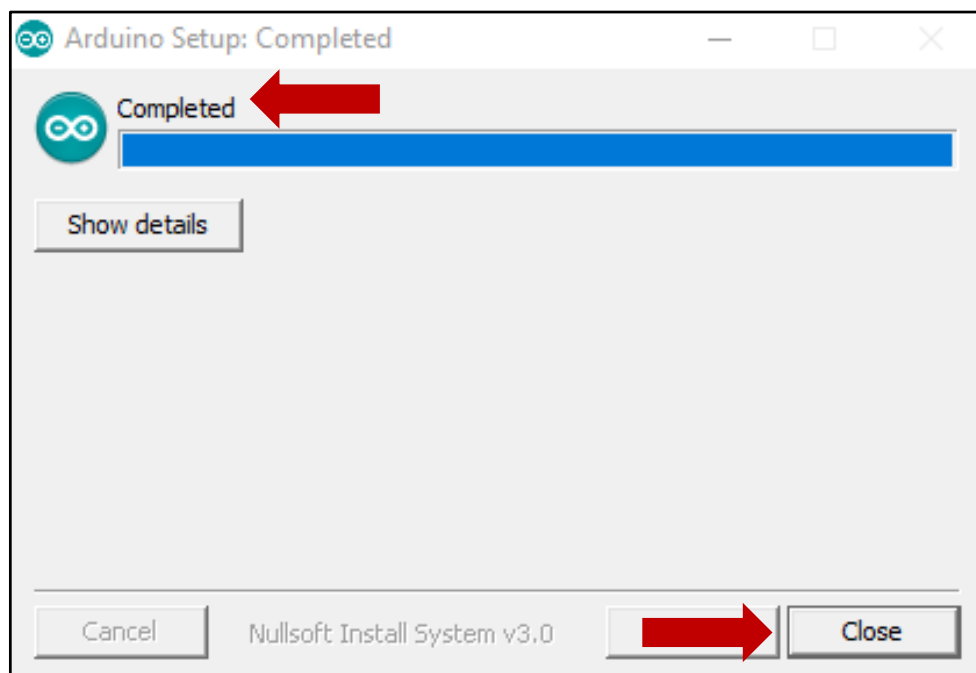
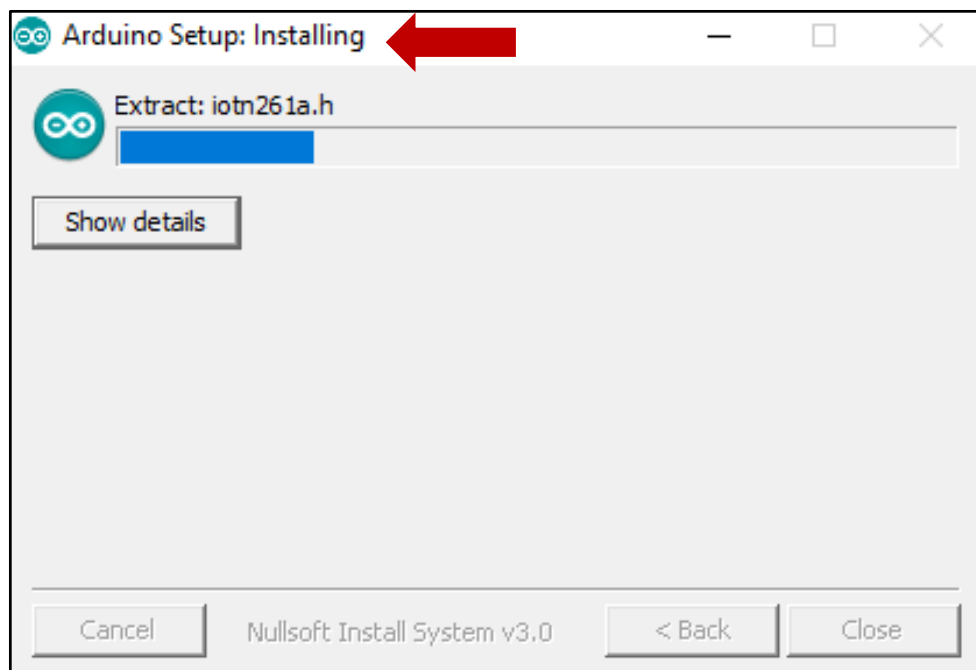


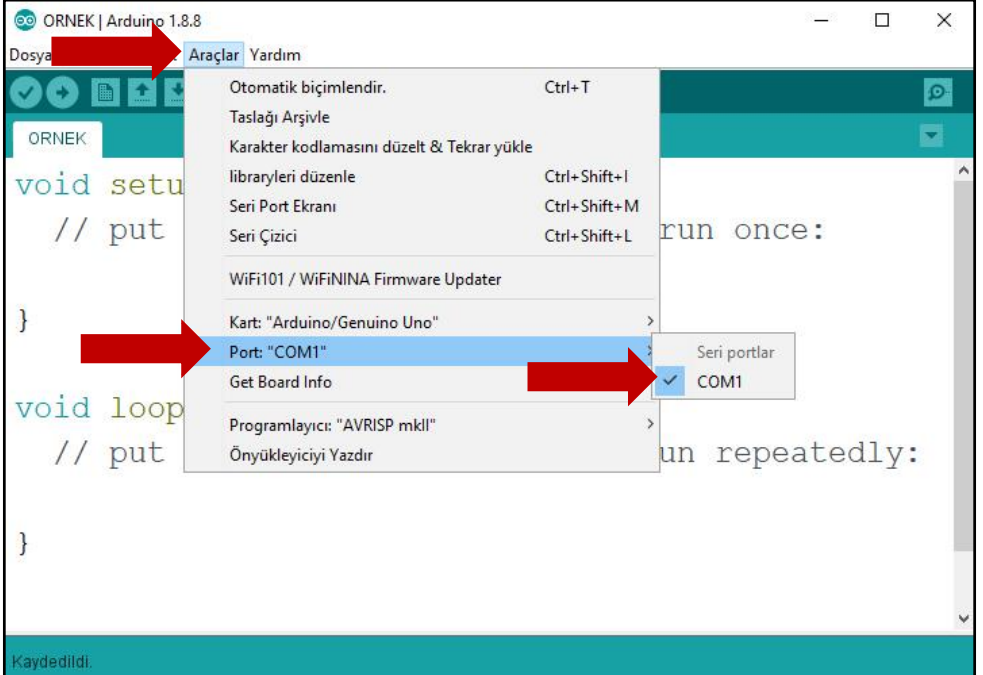
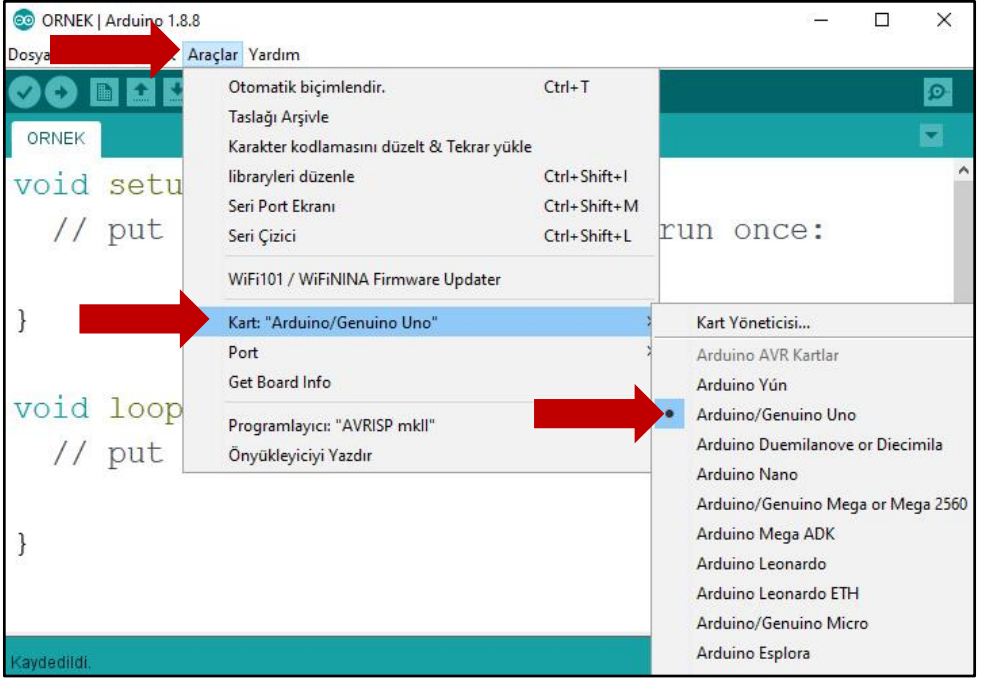
SINCE MARCH 2015, THE ARDUINO IDE HAS BEEN DOWNLOADED **34,277,254** TIMES. (IMPRESSIVE!) NO LONGER JUST FOR ARDUINO AND GENUINO BOARDS, HUNDREDS OF COMPANIES AROUND THE WORLD ARE USING THE IDE TO PROGRAM THEIR DEVICES, INCLUDING COMPATIBLES, CLONES, AND EVEN COUNTERFEITS. HELP ACCELERATE ITS DEVELOPMENT WITH A SMALL CONTRIBUTION! REMEMBER: OPEN SOURCE IS LOVE!

\$3 \$5 \$10 \$25 \$50 OTHER

JUST DOWNLOAD CONTRIBUTE & DOWNLOAD







ARDUINO IDE « TEMEL C KODLAMA »

setup() : Programın başlangıç ayarları içindir. Başlangıçta bir defa çalışır. Bu fonksiyon değişkenler, pin modları, seri iletişim ve kütüphaneler içindir.

loop() : « Loop » Türkçesi « Döngü » dür. Adından anlaşılacağı gibi **setup()** fonksiyonundan sonra sonsuz döngü şeklinde sürekli çalışır. Ana program kodları bu fonksiyon içine yazılır.

```
// Tanımlamalar
void setup() {
    // Kurulum kodları buraya yazılır. (Bir defa çalışır)
}
void loop() {
    // Ana program kodları buraya yazılır. (Sürekli çalışır)
}
```

- Program yazımı belirli kalıpta, bloklar halinde olur.
- Bloklar, { } parantezleri ile oluşturulur.
- Komutlar aynı veya alt alta satırlara yazılabilirler.
- Tüm komutlar, noktalı virgül (;) ile biter.
- Programda kullanılan tüm değişkenler ve bilgi tipleri bildirilir.
- Programın başında kütüphaneler aktifleştirilir/çağrılır.
- Açıklamalar “//” ve “/* */” (Birden fazla satır için) ile yazılır.
- #define ile eşdeğer ifade atanır.
- #include ile kütüphane çağrılır.

IF / ELSE

ARDUINO programlamada temel karar komutudur. « if » ten sonra verilen koşul doğru ise bu « if » bloğundaki işlemler, yanlış ise “else” bloğundaki işlemler yapılır.

```
if (koşul) {  
    // Koşul sağlanıyor ise yapılacak işlem  
}  
else {  
    // Koşul sağlanmıyor ise yapılacak işlem  
}
```

FOR

ARDUINO programlama dilinde temel döngü komutudur. « for » içinde yer alan koşul doğru olduğu sürece, döngüdeki işlemler gerçekleştirilir.

```
for (başlangıç değeri; koşul; artım)  
{  
    // işlemler  
}
```

WHILE

« while » ile belirtilen koşul doğru olduğu sürece, döngü içindeki işlemler gerçekleştirilir.

DO – WHILE

« while » ile belirtilen koşul doğru olduğu sürece, döngüdeki işlemler yapılır.

```
while (koşul) {  
    // Koşul doğru olduğu sürece yapılacak işlemler  
}  
do {  
    // Koşul doğru olduğu sürece yapılacak işlemler  
} while (koşul);
```

SWITCH / CASE

Seçim yapılarak program akışının istenilen bloklara atlamasını sağlar. « switch » teki değişken, « case » deki hangi değeri alırsa, karşılığındaki işlem yapılır.

```
switch (seçim değişkeni) {  
    case 1:  
        // Seçim değişkeni «1» olduğunda yapılacak işlem  
        break;  
    case 2:  
        // Seçim değişkeni «2» olduğunda yapılacak işlem  
        break;  
    default:  
        // Varsayılan bağımsız işlem (isteğe bağlı)  
}
```

ARDUINO IDE « I/O KODLAMA »

pinMode(Pin_No, Pin_Tür);

Pinleri giriş veya çıkış olarak yapılandırma işlemi yapar. (13, INPUT)

digitalWrite(Pin_No, Lojik_Değer);

Çıkış olarak ayarlanan pinlerin değerlerini, HIGH veya LOW olarak ayarlar.

digitalRead(Pin_No);

Belirtilen digital pin değerini okur.

analogRead(Pin_No);

Belirtilen analog pin değerini okur. (0-1023 arasında değerdir. **10BitADC**)

analogWrite(Pin_No, Değer);

Ayarlanan pinden analog çıkış almayı sağlar. LED parlaklığı, motor hızı ayarlama gibi işlemlerde kullanılır. (0-255 arasında değerdir. **8BitDAC**)

delay (Milisaniye);

Milisaniye biriminde zaman gecikmesi verir.

**map (Giriş_Değeri, Giriş_Değeri_Minimum,
Giriş_Değeri_Maksimum, Çıkış_Değeri_Minimum,
Çıkış_Değeri_Maksimum);**

map (Giriş_Değeri, 0, 1023, 0, 255);

Giriş_Değeri değişkeni ile gönderdiğimiz sayı 0-1023 arasında değişebilir ve bu değişimle orantılı olarak fonksiyon 0-255 değerleri arasında geri dönüş yapar.

B. ARDUINO İLE DEVRE KURMA

B.1. Elektronik Elemanlar ve Devre Teorisi

Arduino her ne kadar elektronik devre sistemleri ile çok fazla uğraşmadan, birçok uygulama yapılmasını sağlasa da, uygulamalar çeşitlendikçe ek elektronik elemanlar, algılayıcılar ve dönüştürücüler içeren elektronik devreler kurmak ve Arduino üzerinden bu devreleri programlayıp yönetmek ihtiyacı kaçınılmazdır. Bu sebeple, temel elektrik ve elektronik bilgisi yanında, devre elemanları ile temel seviyede devre teorisi ve çözümü yaklaşımına ihtiyaç duyulmaktadır. Bu bölümde, temel seviye elektrik ve elektronik bilgisi, devre elemanları açıklanacak ve bu elemanlar ile devre kurulumu sırasında ihtiyaç duyulacak olan devre teorisi üzerinde durulacaktır.

Elektrik

Elektrik, yüklü parçacıkların hareketi ile ilgili olan fiziksel olaylara verilen genel isimdir. Elektrik, bilinen anlamda iletken maddelerin akım iletmesi ile ilgili olabileceği gibi, statik elektrik, radyo dalgaları (elektromanyetik dalgalar) ve yıldırım gibi daha farklı fiziksel olaylar da elektrik ile ilgilidir.

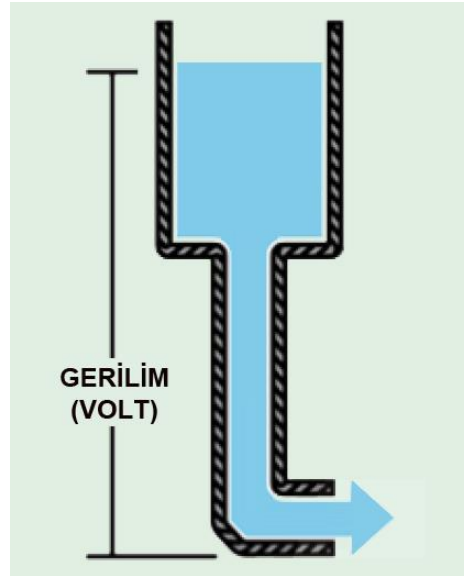
İletken Ve Yalıtkan Maddeler

Elektrik akımını ileten maddelere iletken adı verilir. Metallerin iletken oldukları bilinmektedir (Altın, Bakır, Demir, Gümüş vs). Buna karşılık elektrik akımını taşıyacak serbest elektronlara sahip olmayan, yani elektrik akımını iletmeyen maddelere yalıtkan maddeler adı verilir (Plastik, Kauçuk, Cam, Porselen vs).

Gerilim (Voltaj)

Elektrik gerilimi, akımın oluşması, yani elektronların hareketi için gerekli olan elektrik alan kuvvetidir. Elektrik akımın “akmasını” yani oluşmasını sağlar. Birimi “VOLT” tur. “V” harfi ile gösterilir. Gerilim (Voltaj), elektrik akımını itme gücü olarak tanımlanabilir. Gerilim voltmetre ile ölçülür. Voltmetre ölçüm yapılacak devre elemanına paralel olarak bağlanır.

Kavramları daha kolay anlayabilmek adına elektrik enerjisinin su ile benzerliğine bakılabilir. Gerilimi (Volt) su seviyesinin yüksekliği olarak düşünebiliriz. Su yüksekliği ne kadar fazla olursa, akış o kadar şiddetli olacaktır. Yani gerilim yükseldikçe akacak olan akım şiddeti o kadar artacaktır.



Elektrik Akımı

Güç kaynağı seçimi yaparken, devrenin çalışacağı gerilime uygun bir kaynak kullanılması gerekir. Örneğin 6V motorlara sahip bir robot var ise, bu robotu çalıştırmak için 6V gerilim sağlayabilen bir adaptöre veya bataryaya ihtiyaç olacaktır.

Elektrik akımı, potansiyel fark (gerilim) etkisi sonucunda iletken bir madde üzerinden elektrik yüklerinin hareketi olarak tanımlanabilir. Hareket eden yükler, madde içerisindeki elektronlardır. Birimi "AMPER" dir. "A" harfi ile gösterilir. Denklemlerde "I" harfi ile ifade edilir.

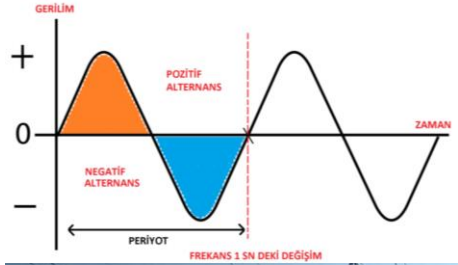
Elektrik devrelerinde asıl işi yapan akımdır. Gerilim akımın akmasını sağlar. Devreyi besleyecek olan pili veya güç kaynağını seçerken devrenin çekeceği en yüksek akımı düşünmek gerekir.

Robot örneğinde olduğu gibi, 6V ile çalışan robotun en yüksek durumda 1A akım çekeceği varsayılırsa, bu durumda en az 6V-1A akım verebilen bir adaptör veya pil kullanılması gerekecektir. Adaptörün verebileceği akımın yüksek olması bir sorun değildir.

Robot ihtiyacı olan 1A akımı çekecek, adaptör ise bu değeri rahatlıkla karşılayabilecektir. 6V-10A bir güç kaynağı da kullanılabilir. Tek fark maliyet ve güç kaynağının boyutu artacaktır. Akım, ampermetre ile ölçülür. Ampermetre devreye seri olarak bağlanır.

AC / DC

AC, Alternating Current (Alternatif Akım) anlamına gelmektedir. İsmi, akımının belirli bir periyot ile pozitif ve negatif alternanslar arasında değişmesinden almaktadır. Üretim ve dağıtım sürecinden, evlere kadar gelen şebeke elektriği, enerji kayıplarının daha az olması sebebiyle alternatif akım şeklindedir.



DC ise, Direct Current (Doğru Akım) kelimelerinin kısaltmasından oluşur. Kalem piller, şarj aletleri, güneş panelleri ve daha birçok elektrikli cihazın çıkışı DC'dir.



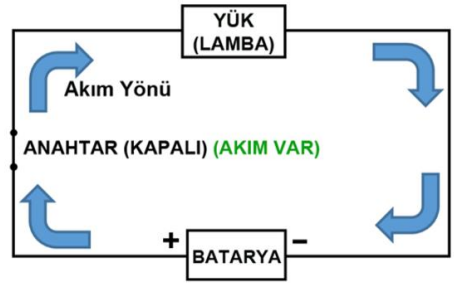
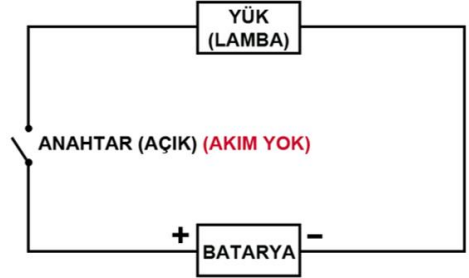
Güç Kaynakları

Bilgisayar, cep telefonu gibi çoğu elektronik cihaz, şebeke gerilimi olan 220V AC'den çok daha düşük gerilim seviyeleri ile çalıştığından adaptörler ile prizden alınan 220V AC gerilim, uygun DC gerilime çevrilir. Günümüzde elektrik enerjisini elde etmenin birçok farklı yöntemi mevcuttur. Hidroelektrik Santralleri (SU), Termik Santraller (YAKIT), Nükleer Santraller (ATOM) ve özellikle günümüzde çok daha önem kazanan yenilenebilir enerji üretim yöntemleri olan Güneş ve Rüzgar Santralleri başlıca elektrik enerjisi üretim yöntemlerindedir.



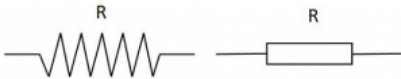
Açık/Kapalı Devre

Bir elektrik devresinde anahtar açık ise, devrede akım dolaşmaz. Bu tip devrelere açık devre adı verilir ve Yük çalışmaz, Lamba ışık vermez. Anahtarın kapalı olduğu devreye kapalı devre adı verilir. Bu devrede akım dolaşır, Yük çalışır ve Lamba ışık verir.



a. Direnç, Diyot, Transistör

DİRENÇ, elektrik devrelerinde, iletken üzerinden geçen akımının karşılaştığı zorlanmadır. Elektrik akımına zorluk göstererek, akımı sınırlandıran elemana DİRENÇ denir. Mekanik sistemlerdeki sürtünmeye benzer özellikler gösterir. Direncin birimi Ohm (Ω)'dur. Denklemlerde R harfi ile gösterilir. Elektronik devrelerde direncin sembolü iki farklı şekilde gösterilebilir.

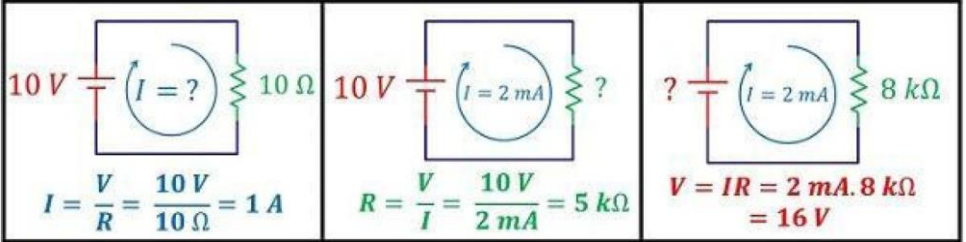
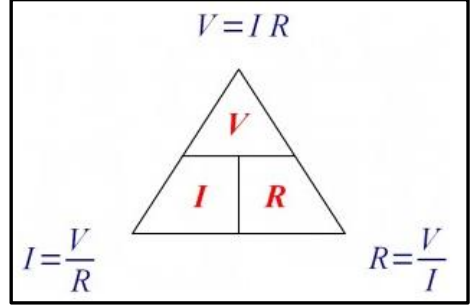


Dirençler, elektrik devrelerinde akımı sınırlandırarak belli bir değerde tutmaya yararlar. Bunun haricinde hassas devre elemanlarının üzerlerinden yüksek akım geçmesini önlerler, besleme gerilimini ve akımı bölmek için kullanılırlar. Farklı tipteki bazı dirençler pasif algılayıcı görevi görerek ortamdaki fiziksel değişimleri algılayabilirler. Çeşitlerine göre dirençlerin farklı kullanım alanları vardır. Direnç, yukarıda belirtilen görevleri sebebiyle neredeyse tüm elektronik devrelerde bulunmaktadır.

Kullanım alanlarına göre diğer tipteki dirençler farklı görevlerde de kullanılırlar. Potansiyometre gibi ayarlı dirençler, çıkış sinyalinin kontrol edilmesi istenilen devrelerde sıklıkla kullanılırlar. LDR tipi dirençler, üzerlerine düşen ışık şiddetine göre davranan bir algılayıcı görevi görürler; ışığa duyarlı devrelerde kullanılırlar. NTC-PTC termistörleri ise üzerlerine uygulanan ısıya göre davrandıklarından dolayı yine bir algılayıcı görevi görerek ısıya duyarlı devrelerde kullanılırlar.

Bir elektrik devresinde direnç değeri Ohm Yasası sayesinde hesaplanır. İki uçlu bir devre elemanının direnci, üzerindeki gerilimin (V), eleman üzerinden geçen akıma (I) bölümü ile hesaplanır. Elektronik devrelerde akım, gerilim ve direnç arasındaki ilişki ise Ohm yasası ile hesaplanmaktadır.

Ohm yasası, Alman fizikçi George Simon Ohm tarafından 1827 yılında bulunmuştur. Bu yasa, elektriğin temel yasalarındandır. Bir elektronik devrede iki nokta arasında bulunan iletkenin üzerinden geçen akım, üzerindeki potansiyel fark ile doğru orantılı, sahip olduğu direnç ile ters orantılıdır.



Direnç Renk Kodları Hesaplama

1. Şerit: İlk Basamak
2. Şerit: İkinci Basamak
3. Şerit: Çarpan Katsayısı
4. Şerit: Tolerans



1. Şerit: İlk Basamak
2. Şerit: İkinci Basamak
3. Şerit: Üçüncü Basamak
4. Şerit: Çarpan Katsayısı
5. Şerit: Tolerans



Renk	1. Şerit (ilk basamak)	2. Şerit (ikinci basamak)	3. Şerit (üçüncü basamak)	4. Şerit (katsayı)	5. Şerit (tolerans)	6. Şerit (sıcaklık katsayısı)
Siyah	0	0	0	$\times 10^0$		
Kahve	1	1	1	$\times 10^1$	$\pm \%1$	100
Kırmızı	2	2	2	$\times 10^2$	$\pm \%2$	50
Turuncu	3	3	3	$\times 10^3$	$\pm \%3$	15
Sarı	4	4	4	$\times 10^4$	$\pm \%4$	25
Yeşil	5	5	5	$\times 10^5$	$\pm \%0.5$	
Mavi	6	6	6	$\times 10^6$	$\pm \%0.25$	10
Mor	7	7	7	$\times 10^7$	$\pm \%0.1$	5
Gri	8	8	8	$\times 10^8$	$\pm \%0.05$	
Beyaz	9	9	9	$\times 10^9$	$\pm \%1$	
Altın					$\pm \%5$	
Gümüş					$\pm \%10$	

İlk şerit **KAHVERENGİ**, ilk basamak **1**. İkinci şerit **SİYAH**, ikinci basamak **0**. Elde edilen sayı **10**. Üçüncü basamak **KIRMIZI**, Çarpan Katsayısı **2**, 10^2 yani **100** değerini ifade ediyor. Yani $10 \times 10^2 = 1000$ Ohm değerinde bir dirençtir. Son şerit **ALTIN**, olduğundan direncin $\%5$ 'lik bir hata payı ile üretildiğini gösteriyor. Direnç **1000 Ohm** değerinin $\%5$ 'i yani 50 Ohm daha yüksek veya düşük dirence sahip olabilir.

Direnç 5 şeritli olduğunda, katsayı değerinden önce bir hane daha vardır. Yine 1000 Ohm $\pm \%5$ değerine sahip 5 şeritli bir direncin renkleri yukarıdaki gibi olacaktır.

Direnç Çeşitleri

Üretildikleri malzeme çeşidi, kılıf tipleri, değişken-sabit gibi farklı parametrelere göre direnç çeşitleri aşağıdaki gibidir.

Karbon Dirençler

Karbon karışımı veya karbon direnç, toz halindeki karbon ve reçinenin ısıtılarak eritilmesi yolu ile elde edilir. Karışımdaki karbon oranı direncin değerini belirler. Büyüklüklerine göre $\frac{1}{4}$, $\frac{1}{2}$, 1, 2, 3 watt / 1Ω dan $22\text{ M}\Omega$ 'a kadar değerlerde üretilirler. Üzerindeki çizgilerden değerleri belirtir. Hobi elektronikinde en yaygın kullanılan direnç çeşididir.



Ayarlı Direnç (Potansiyometre)

Direnç değeri isteğe bağlı değiştirilebilen dirençlerdir. Kılıfının üstünde en yüksek değeri yazar ve sıfır ile maksimum değer arasında bir istenilen değere ayarlanabilir.



Termistörler (NTC-PTC)

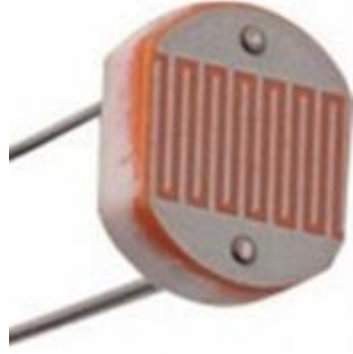
Ortam sıcaklığına göre direnç değerleri değişir. Positive Temperature Coefficient (PTC) sıcaklık artınca, Negative Temperature Coefficient (NTC) sıcaklık azalınca dirençleri yükselen dirençlerdir.



Foto Direnç (LDR)

LDR, Light Dependent Resistor kelimelerinin baş harflerinden oluşan bir kısaltmadır. Bu elemanların yapısında yarı iletken madde olarak kadmiyum sülfat (CdSO4) kullanılmaktadır.

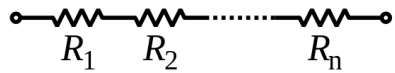
Aydınlık ortamlarda direnci 10Ω 'a kadar düşer. Karanlık ortamda direnci $200M\Omega$ 'a kadar çıkar. Işığa bağlı olarak direncinin değişmesi özelliği sayesinde sensör olarak kullanılabilir.



Direnç Bağlantıları – Eşdeğer Direnç

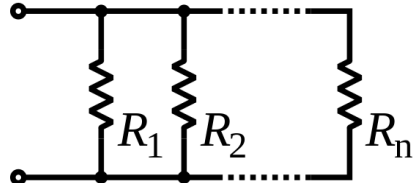
Dirençler, diğer iki uçlu devre elemanları gibi seri veya paralel şekilde birbirlerine bağlanabilirler. Böylelikle mevcut olmayan direnç değerleri elde edilebilir. Farklı bağlantı çeşitlerine göre toplam direnç -eşdeğer direnç- değeri elde edilir.

Seri Direnç Hesaplama



$$R_{eş} = R_1 + R_2 + R_3 + \dots + R_n$$

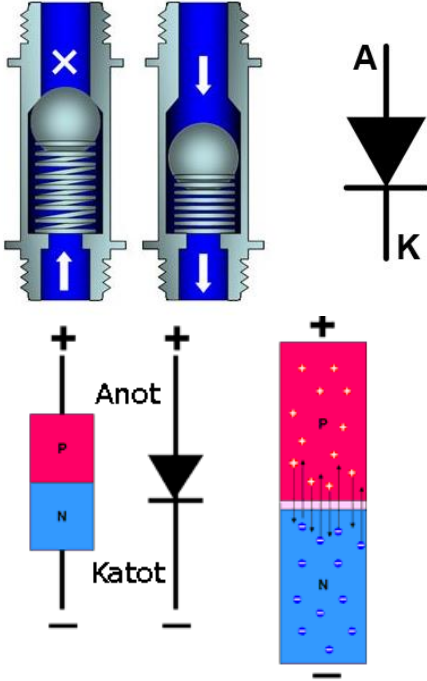
Paralel Direnç Hesaplama



$$\frac{1}{R_{eş}} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \dots + \frac{1}{R_n}$$

DIYOT, elektrik akımının yalnızca bir yönde geçişine izin veren, yarı iletken maddelerden yapılmış iki uçlu bir devre elemanıdır. Devrelerde aşağıdaki diyot sembolü ile gösterilir.

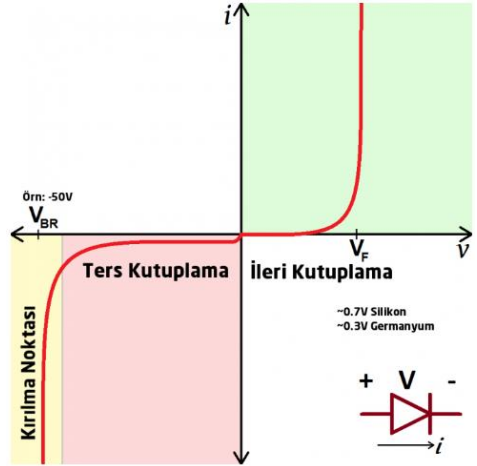
Diyotun **ANOT** ve **KATOT** olmak üzere iki bacağı bulunur. Diyotlar, akımı üzerlerinden yalnızca anottan katoda doğru iletirler. Su ile benzerliğinden örnek vermek gerekirse diyotlar, bir boru üzerindeki çek-valf gibi davranırlar.



Diyot Çeşitleri

LED Diyot	Varaktör Diyot
Zener Diyot	IMPATT Diyot
Schottky Diyot	Pin Diyot
Lazer Diyot	Köprü Diyot
Kristal Diyot	Silikon Diyot
Tünel Diyot	(1N4001-1N4007)
Foto Diyot	Germanyum Diyot

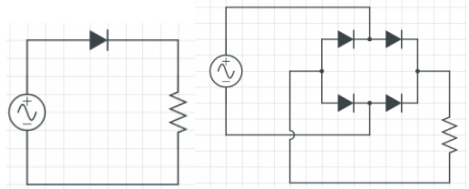
Bir elektronik devre elemanının davranışını (karakteristiğini) anlamak için **AKIM-GERİLİM** grafiğine bakılır.



Diyotun Kullanım Alanları

Diyotların kullanımına örnek olarak verebileceğimiz ilk devre tipi doğrultuculardır.

Doğrultucu, AC gerilimi DC gerilime dönüştürmede kullanılan devrenin ismidir.

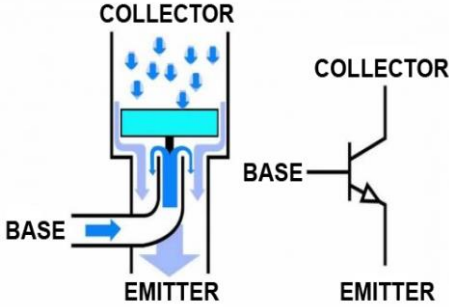


Yarım Dalga Doğrultmaç

Köprü Tipi Tam Dalga Doğrultmaç

Ayrıca Diyotlar, doğrultucu haricinde gerilim regülatörü, ters polarite koruması, lojik devre kapıları olarak, seri bağlı güneş panellerinde baypas amaçlı ve indüktif devrelerde gerilim sıçramalarına karşı koruma amaçlı olarak kullanılırlar.

TRANSİSTÖR, küçük elektrik sinyallerini yükseltmek veya anahtarlamak amacıyla kullanılabilen bir yarı iletken devre elemanıdır. 3 veya daha fazla bacağı bulunan transistörün bacaklarından birisine uygulanan elektrik sinyali ile diğer bacakları arasındaki elektrik akımını kontrol edebiliriz.



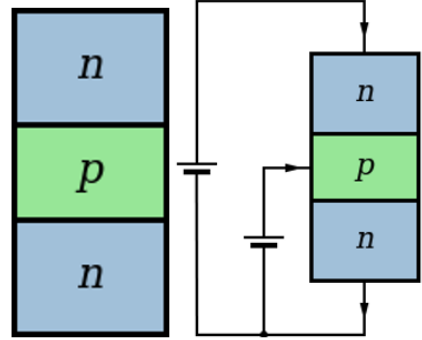
Yarı-iletken Maddeler

Bakır ve demir gibi maddeler iletken, plastik ve seramik gibi maddeler yalıtkandır. Yarı-iletken maddeler ise, normalde yalıtkan olmalarına karşın elektrik gerilimi gibi dış etkilere maruz kaldıklarında iletken olarak davranırlar. Elektronikte genellikle silisyum, germanyum, galyum, arsenit gibi yarı-iletken maddeler kullanılır. Bu maddeler doğadaki saf halleri yerine, işleminden geçirilerek **P-Tipi** ve **N-Tipi** olmak üzere iki farklı çeşide dönüştürülerek devre elemanlarının yapılarını oluşturur.

P-Tipi maddede, pozitif yüklere sahip "delik"ler, maddede yer alan negatif yüklerden daha fazla sayıdadır. N-Tipi yarı-iletkenlerde ise negatif yüklere sahip elektronlar, pozitif yük taşıyıcılardan daha fazla sayıdadır. Bu yük taşıyıcıları, madde üzerinden geçen elektrik akımının hareketini sağlar.

Transistör Nasıl Çalışır?

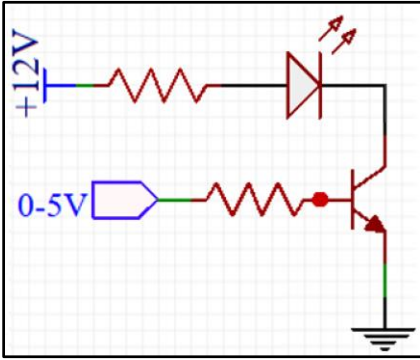
En basit yarı-iletken devre elemanı olan ve akımın yalnızca tek bir yönde akmasını sağlayan devre elemanı olan **DIYOT**, iki farklı şekilde kutuplanabilir. Diyotun anot ucuna pozitif, katot ucuna negatif bir gerilim uygulanması durumuna ileri kutuplama ismi verilir. İleri kutuplanmış bir diyotun anot ucundan katot ucuna akım geçişi olur. Gerilimlerin yerleri tam tersine çevrildiğinde ise diyot ters kutuplama durumundadır. Bu durumda diyot üzerinden akım geçmez. İki adet diyot anot uçlarından birleştirilirse, aşağıdaki gibi benzer bir yapı elde edilir.



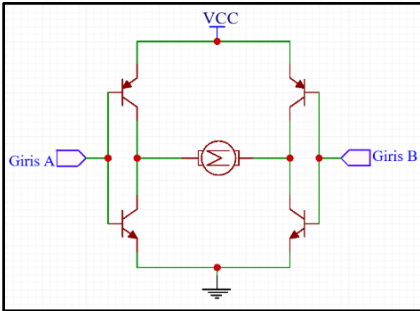
Bu yapının iki ucuna gerilim uygulandığında, güç kaynağı nasıl bağlanırsa bağlansın, **P-N** çiftlerinden bir tane mutlaka ters kutuplu olacaktır. İkinci bir güç kaynağını şekildeki gibi bağladığımızda, orta ve altta yer alan **P-N** çifti ileri kutuplanmış bir diyotu anımsatır. Alttaki **P-N** çiftinin oluşturduğu elektron hareketi, üstteki **P-N** çiftinin de elektronlarını harekete geçirecektir. Ve bir elektron akışı gerçekleşecektir. Bu şekildeki yarı-iletken maddelerden oluşturulan yapıya **Bipolar Junction Transistor (Çift Birleşim Yüzeyleli Transistör-BJT)** adı verilir.

Transistörlerin Kullanım Alanları

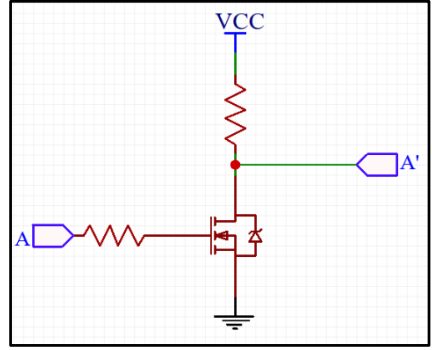
Transistörler sayesinde Arduino gibi mikrodenetleyiciler aracılığıyla LED şeritler, motorlar gibi yüksek güç tüketecek yükleri anahtarlamak mümkündür. Transistör ile LED anahtarlama devresi aşağıdaki gibidir. Devrede 0-5V ile gösterilen bağlantıya PWM sinyali uygulayarak, anahtarlama yüksek hızda yapılabilir, dolayısıyla bağlı olan LED'in parlaklığı veya bağlanan motorun hızı kontrol edilebilir. Motorun yalnızca hızını değil, yönünü de kontrol etmek gerekiyorsa, **H-KÖPRÜSÜ** adı verilen bir devre gereklidir.



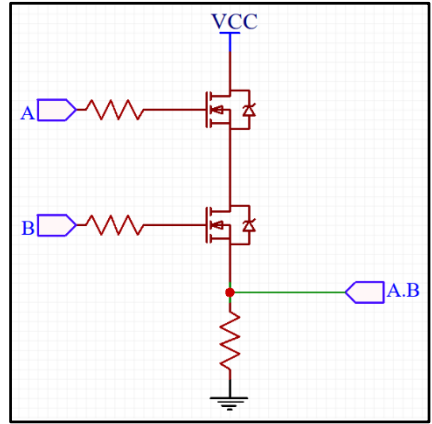
Transistörlü – Led Kontrol Devresi



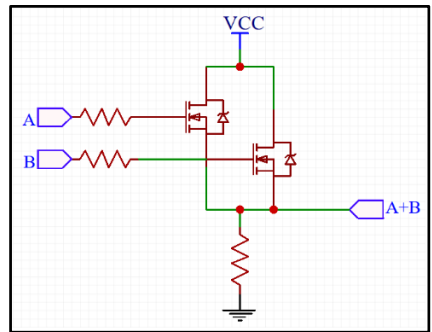
Transistörlü – H-KÖPRÜSÜ



NOT Kapısı (Inverter – Tersleyici)



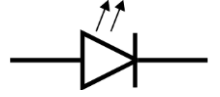
AND (VE) Kapısı



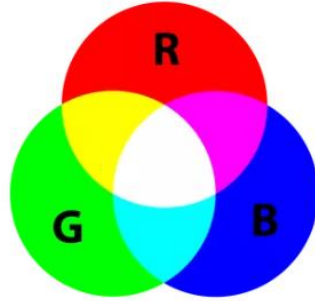
OR (VEYA) Kapısı

b. LED, RGB-LED, Buzzer, Button

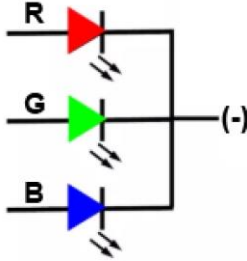
LED-LED Diyot, (Light Emitting Diode – Işık Saçan Diyot) kelimelerinin baş harflerinden oluşan bir kısaltma olan LED, günlük hayatımızda en çok kullandığımız diyot tiplerinden birisidir. Elektronik cihazlarımızın güç ve durum göstergelerinde kullanıldığı gibi, verimli olmalarından dolayı günümüzde aydınlatma amaçlı olarak da sıkça kullanılırlar. Devre sembolü yandaki gibidir.



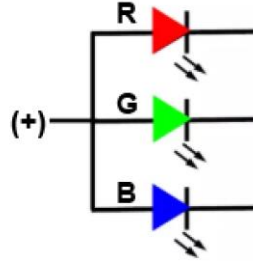
RGB-LED, normal LED'lerden farklı olarak tek paket içerisinde 3 farklı renk (Kırmızı-Yeşil-Mavi) LED'i bir arada bulundurur. Normal LED'lerin ANOT ve KATOT uçları olmasına karşın, RGB-LED'lerde, üretim şekline göre ANOT veya KATOT bağlantıları ortak olarak bulunmaktadır.



COMMON CATHODE(-)



COMMON ANODE(+)



BUZZER, devre elemanı küçük bir hoparlör olarak düşünülebilir. Hoparlörler kadar yüksek ve detaylı ses üretmeseler de, "bip" leme seslerinde oldukça iyidirler.

BUTTON, devre elemanı, devre bağlantılarını açıp-kapatmak amacıyla kullanılmaktadır. İki veya dört bacaklı olmakla birlikte, bağlantı ve çalışma biçimleri aynıdır. Elektronik devrelere tuş sinyali üreterek, farklı görev ve altyardamlar çalıştırılmasını sağlamak amacıyla kullanılırlar.



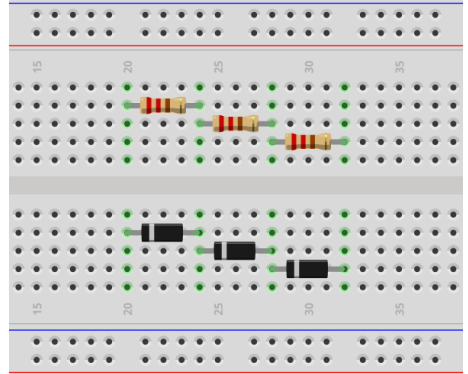
c. Breadboard, Bağlantı Kablosu

Breadboard, üzerinde devrelerin test edildiği bir araçtır. Kurulan devrelerin birbirlerine lehimlenmeden kolaylıkla test edilmesini sağlar. Tasarlanan devreleri baskı devre veya delikli plakette üzerine aktarmadan önce denemeye imkanı sunar. Bu sayede devre bağlantılarını kontrol ederek bir hata olup olmadığı gözlemlenmiş olur. Devreler tak-çıkart şeklinde kurulabildiği için, kullanılan elektronik bileşenleri başka projelerde tekrar kullanma imkanı verir. Breadboard iç yapısı dik ve yatay şekilde birbirlerine bağlı halde konumlanmış metal kısıklardan oluşur. Dışarıdan bakıldığında görünen kırmızı ve mavi kısımları breadboard-un satır kısımlarıdır. Bu kısımlar boydan boya bir satır şeklinde iletim halindedir. Bu kısımlar breadboard-un iki yanında görülebilir. Burada dikkat edilmesi gereken nokta, bu satır bazı breadboard-larda ortadan ikiye ayrılmış durumdadır. Yani baştan sona kısa devre değildir. Breadboard-un ortada kalan kısımları da sütun boyunca yerleştirilmiş iletkenlerden oluşur. Bu kısımlar da tıpkı satırlarda olduğu gibi breadboard-un iki tarafında bulunur. Tüm bu iletkenlerin üstü elektronik bileşenlerin ayaklarının yerleştirilmesi için açılmış deliklerden oluşan bir plastik ile kapalıdır.

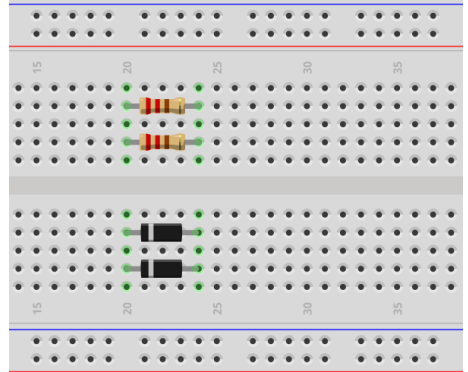


Breadboardlar ebatlarına göre birkaç çeşitten oluşurlar. Hangi breadboardun kullanılacağı, projedeki elektronik bileşenlerin miktarı ve bacak sayıları ile alakalıdır. Projenin büyüklüğüne göre istenilen breadboard tercih edilebilir. Karmaşık yapılı ve çok bileşenli elektronik projelerde büyük breadboard bile yeterli olmayabilir. Bu durumlarda birden fazla breadboardu yanlarındaki çentikler yardımıyla birleştirerek istenilen boyutta bir board elde edilebilir. Genel olarak breadboard-lar **Mini Boy**, **Orta Boy** ve **Büyük Boy** şeklinde sınıflandırılır.

Seri Bağlantılar

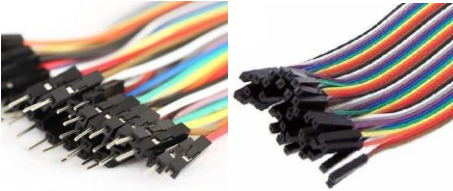
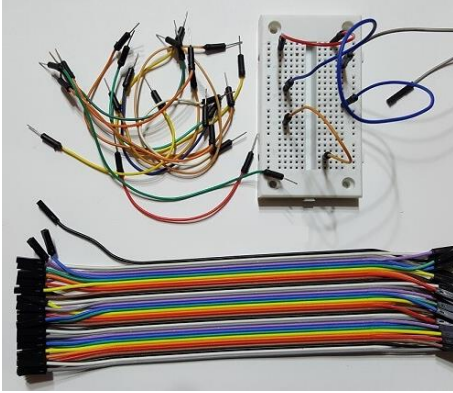


Paralel Bağlantılar



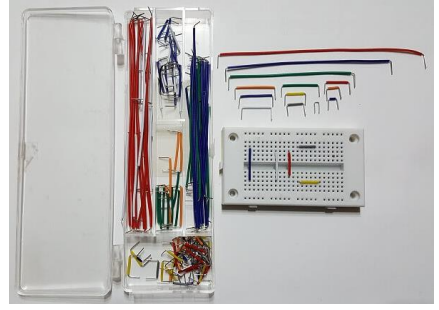
Bağlantı Kabloları (Jumper), özellikle breadboard ile Arduino gibi geliştirme kartlarının bir arada kullanıldığı devreler için oldukça uygundur. Uçlarında **Dişi** ve **Erkek** girişlerin olduğu üç çeşidi bulunmaktadır.

Erkek-Erkek / Erkek-Dişi / Dişi-Dişi

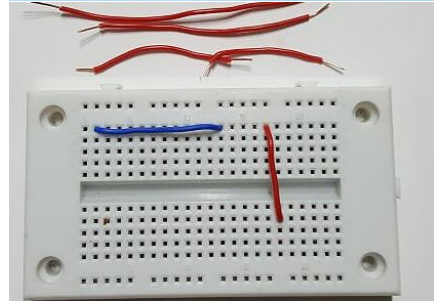


Bağlantı yapılacak girişlere göre, bu çeşitlerden uygun olanları seçilebilir. Ayrıca, farklı çeşitlerin bir arada kullanılması (bu kabloları birbirine bağlayarak), daha uzun iletim kabloları oluşturulmasını sağlar. Devre tahtaları için hazırlanmış, içerisinde değişik boylarda tek damarlı kabloların olduğu kutulu Jumper-lar bulunmaktadır.

Breadboard-a tam oturduğu için, bu seçenek bir öncekine göre kurulan devrenin daha düzenli görünmesine yol açar.



Eğer kutulu Jumper Kablolar pahalı geliyorsa, 100 metre rulolar halinde satılan tek damarlı bakır kablardan alıp, istenilen boylarda kesip, uçlarını açmak yoluyla, Jumper kablolar daha ucuza hazırlanabilir.

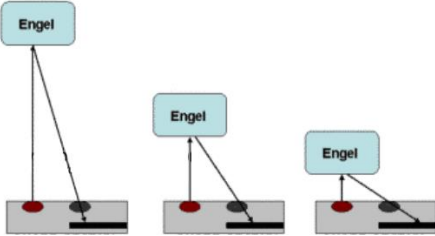


Tek damarlı bakır kablo gibi, ucuza mal edilebilecek bir diğer alternatif ise, CAT-5, CAT-6 gibi internet kablolarını Jumper kablo haline getirmektir. Yalnız, kablonun AWG değerine dikkat etmek gerekir. 23-24 AWG gibi daha kalın türleri breadboard (devre tahtası) için daha uygun olacaktır. 26 AWG fazla incedir.

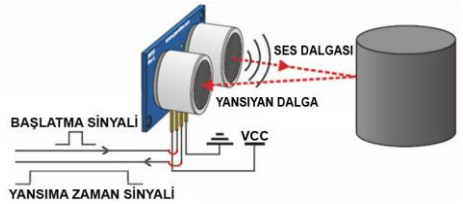
B.2. Algılayıcı ve Dönüştürücüler

a. Mesafe Algılayıcı Elemanlar

Robot ve otomasyon projelerinde bazen sadece engeli algılamak yeterli olmayabilir. Mesafe algılaması gerektiren durumlarda SHARP mesafe algılayıcıları (Sensör) kullanılabilir. SHARP algılayıcıların mesafeyi nasıl algıladıkları aşağıdaki şekilde gösterilmektedir. SHARP algılayıcının yaydığı IR ışık engelle çarpıp geri yansır ve yansıyan IR ışık SHARP algılayıcının alıcısı tarafından algılanır. Engel ile SHARP algılayıcı arasındaki mesafeye bağlı olarak IR ışığın yansıma açısı ve ışığın dedektör üzerinde düştüğü nokta değişir. Dedektör bu veriyi okur ve mesafeyi hesaplar.



Kızılötesi (IR) dışında ultrasonik ses dalgaları yoluyla mesafe ölçülebilen algılayıcılar da mevcuttur. Ultrasonik ses dalgaları 20kHz-500kHz arasında frekanslara sahip ses dalgalarıdır. İnsanların duyabildiği 300Hz-14000Hz bandının üzerindedir. Ultrasonik algılayıcılar, ultrasonik ses dalgaları yayan ve bunların engellere çarpıp geri dönmesine kadar geçen süreyi hesaplayarak aradaki uzaklığı belirleyebilen algılayıcılardır. Bu algılayıcılarda bu kadar yüksek frekanslarda ses dalgalarının yayılmasının nedeni; bu frekanslardaki dalgaların düzgün doğrusal şekilde ilerlemeleri, enerjilerinin yüksek olması ve sert yüzeylerden kolayca yansmasıdır.



b. LCD Ekranlar ve I2C Sürücüler

LCD (Liquid Crystal Display - Sıvı Kristal Görüntü), sıvı kristal yöntemi ile görüntü veren ekranlardır. Bu kristal görüntüleri oluşturan her bir noktaya piksel adı verilir. Bu piksellerin birleştirilmesi ile gerçek bir görüntü elde edilir.

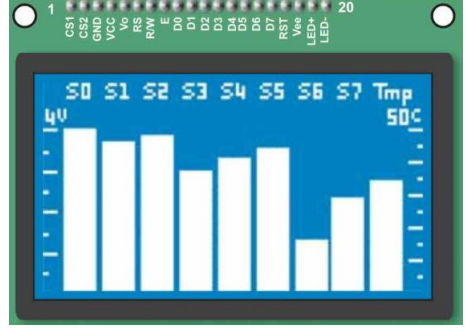
Alfanümerik LCD Ekranlar

Bu ekranlar üzerinde sadece harf ve rakam yazdırılabilir. Genellikle basit elektronik devrelerde kullanılırlar. Ölçülen bir değeri veya herhangi bir hesaplamayı göstermek için kullanılırlar. Anlık değer alınan bazı uygulamalarda ve basit elektronik devrelerde kullanılırlar. Alfanümerik LCD modelleri karakter ve satır sayısına göre ifade edilirler. Bunlar **Karakter_Sayısı x Satır_Sayısı** olarak şöyledir. 16x2, 20x2, 20x4.



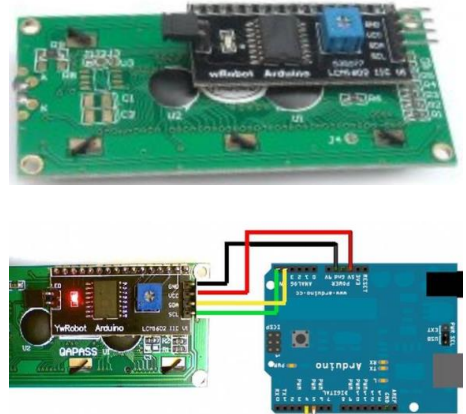
Grafik LCD Ekranlar

Bu tür ekranlar ile grafik görüntüler alınabilir. Yani ekran alanına uygun olarak istenilen görüntü bu ekrana alınabilir. TV ve telefon ekranları Grafik LCD ekranlara örnek verilebilir.



I2C Sürücüsü

Arduino ile LCD ekranı çalıştırmak için 9 adet bağlantı gereklidir. Bu durum bağlantılarda soruna yol açabileceği gibi, Arduino üzerinde çok fazla pin kullanır. Bu sorun I2C protokolü kullanılarak çözülebilir. Bazı LCD ekranlar I2C sürücü modülü dahili olarak bulunmaktadır.



c. DC Motor Çeşitleri ve Sürücüleri

DC motor, düz elektrik akımı enerjisini mekanik enerjiye dönüştüren makinedir. Motorun içinde yer alan sargılara elektrik akımı uygulandığında, yine motorun içerisinde bulunan sabit mıknatlara zıt yönde oluşan manyetik kuvvetin etkisi ile hareket etme prensibiyle çalışır. Bu akımın yönünün, sürekli olarak sabit mıknatısa ters manyetik alan oluşturacak şekilde değiştirilmesi gereklidir. Bu değişim, fırçalı motorlarda motorun sarımlarına temas eden fırçalar ile, fırçasız motorlarda ise elektronik hız kontrol devresi tarafından yapılır.

Fırçalı DC Motorlar

En temel çeşit DC motor tipidir. Redüktör ile beraber veya redüktörsüz şekilde birçok projede kullanılırlar. Avantajları kolay bir şekilde sürülebilmeleri, dezavantajları ise fırça ya da kömür ismi verilen aşınan parçalarının periyodik olarak değiştirilmesi gerekliliğidir.

Fırçasız DC Motorlar

Fırçalı DC motorların yerini almaları için tasarlanmıştır. Çalışmaları için özel sürücü devreleri kullanılır. Sürtünmenin en az düzeyde olması sayesinde verimliliklerinin çok yüksek olması ve fırça gibi aşınan parça olmaması sebebiyle yüksek performans ihtiyaç duyulan uygulamalarda kullanılabilirler.



Redüktörlü DC Motorlar

DC motorlar çoğunlukla yüksek devir özelliğine sahiptir. Yüksek devir yerine yüksek tork tercih edilen uygulamalarda, motorun miline bağlanan bir dişli seti sayesinde örnek olarak çıkış hızını 30'da 1'e düşürerek, torkun 30 katına çıkması sağlanabilir.

Step Motorlar

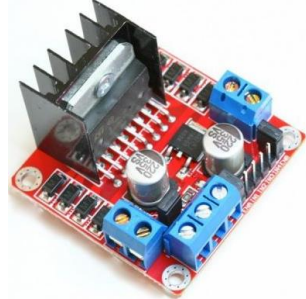
Konumlama hassasiyeti en yüksek motor çeşididir. 2 ve 3 boyutlu yazıcılarda kullanılırlar. Çok hassas konumlama yapabilmelerine karşın, fazla akım çekerler ve hareket hızları fırçalı ve fırçasız motorlara göre daha yavaştır.

Servo Motorlar

0 ile 180 derece açısal aralıkta, istenilen biçimde konumlama yapabilen motorlardır. Sürekli dönebilen tipte servo motorlar bulunsa da, asıl amaçları uçak, helikopter, araba gibi araçlarda kanat, direksiyon, kontrol yüzeyi gibi parçaların açısını kontrol etmektir. Harici bir sürücü devresine ihtiyaç duymazlar. PWM ile çalışırlar.

DC Motor Sürücü Kartları

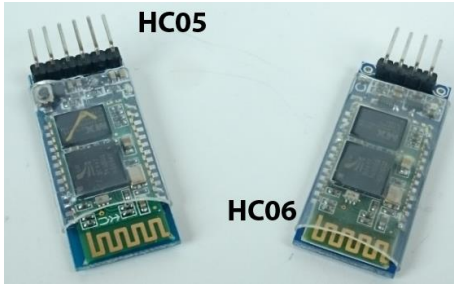
Redüktörlü veya redüktörsüz, fırçalı DC motorları sürmek için H-Köprüsü adı verilen devreler kullanılır. Hazır olarak H-Köprüsü devresi entegresini ve bu entegrelerin ihtiyaç duyacağı devre elemanlarını barındıran motor sürücü kartları tercih edilebilir.



d. Bluetooth ve Kızılötesi (IR) İletişim Elemanları

HC05-HC06 Bluetooth Modülleri

Mikrodenetleyici projelerinde en pratik biçimde kullanılabilen kablosuz bağlantı Bluetooth tekniğidir. Bluetooth haberleşmede, özellikle fiyatından dolayı en sık kullanılan modüller HC05 ve HC06 Bluetooth modülleridir. Her iki modülün fiziksel görünüşleri hemen hemen aynıdır.



HC06 AT Komut Yönergesi

HC06 modüllerini bilgisayar ile konfigüre etmek için şu şekilde bağlantı yapılmalıdır.

<u>HC06 Modül</u>	>>>	<u>USB-Seri Kart</u>
VCC	>>>	3.3V veya 5V
GND	>>>	GND
TXD	>>>	RXD
RXD	>>>	TXD

Arduino IDE üzerinde doğru COM portunun seçili olduğunun kontrolünden sonra, dikkat edilmesi gereken nokta, kartın baud rate'inin, AT komutlarını göndermek için varsayılan baud rate olan 9600'e ayarlanmasıdır. Bağlantıyı denemek için, "AT" komutu gönderilebilir. Eğer seri port ekranında "OK" cevabı görünüyorsa, bağlantı düzgün çalışıyor demektir.

Modülün ismini değiştirmek için;
AT+NAMEYeniKartAdı

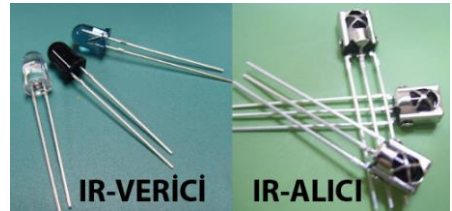
Modülün Şifresini değiştirmek için;
AT+PINYeniŞifre

Modülün Baud rate'ini değiştirmek için;
AT+BAUD4

(1:1200, 2:2400, 3:4800, 4:9600, vs.)

IR (KızılÖtesi)

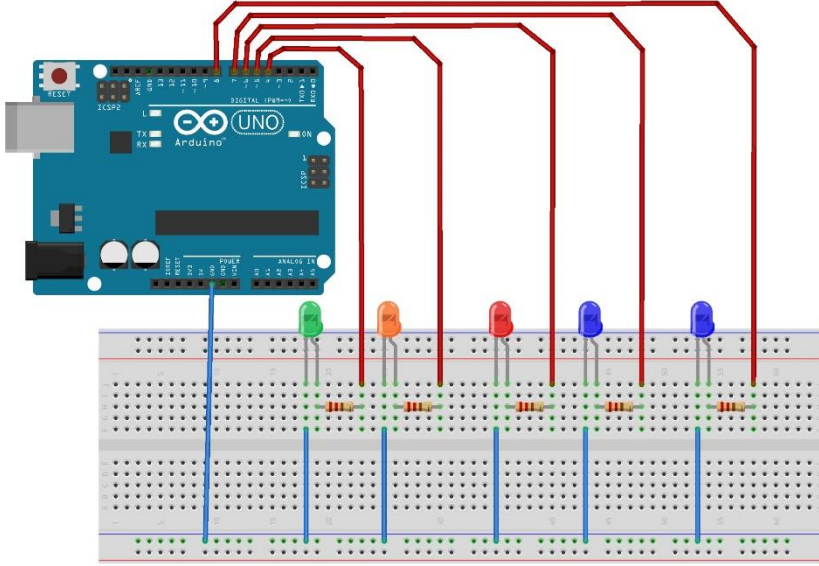
IR, kızılötesi dalga boyunda ışık yayarak veri transferi sağlayan bir sistemdir. Kızılötesi ışık Infrared Led'ten ilgili Infrared Alıcı'ya gönderilir. Bu sistem uzun zamandır birçok alanda mevcut ve bu donanımlara ulaşmak çok kolay ve ucuzdur. Arduino için bu uygulamalarda kullanılmak üzere yapılmış hazır modüller (Shield) olsa da bu malzemeleri elektronik devre elemanları satan mağazalarda bulmak mümkündür.



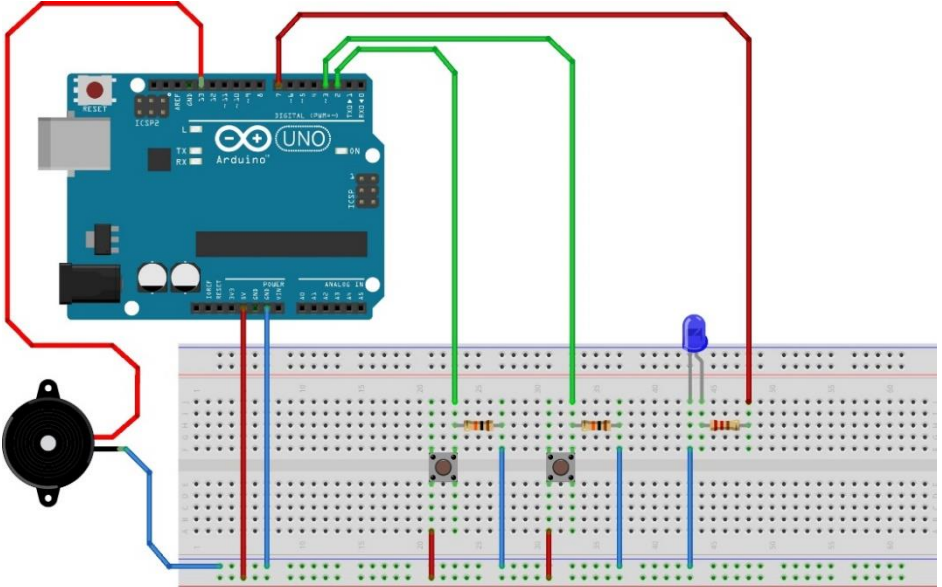
Birçok çeşit IR Alıcı-Verici bulunmaktadır. Burada önemli olan nokta, IR Alıcı pinlerinin çeşitlere göre farklılık göstermesidir. IR tekniği ile, birçok uygulama gerçekleştirilebilir. Yapılması gereken tek şey, kullanılacak olan kumandanın herhangi bir tuşuna basıldığında hangi kodu gönderdiğinin öğrenilmesidir.

B.3. Örnek Devre Uygulamaları

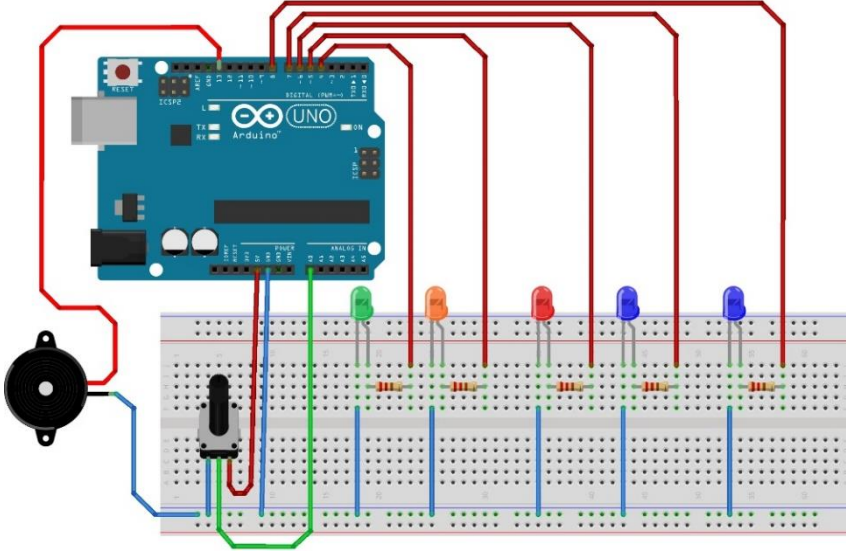
1. Arduino İle 4-5-6-7-8 Numaralı Sayısal Çıkış Pinleri Üzerinden Led Uygulaması



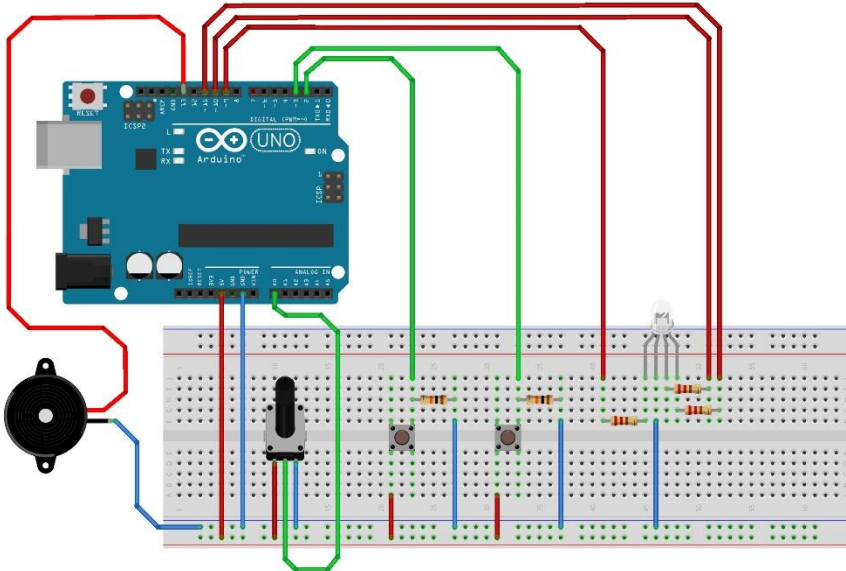
2. Arduino İle 2-3 Numaralı Sayısal Pinlere Button Ve 7-13 Numaralı Sayısal Çıkış Pinleri Üzerinden Led-Buzzer Uygulaması



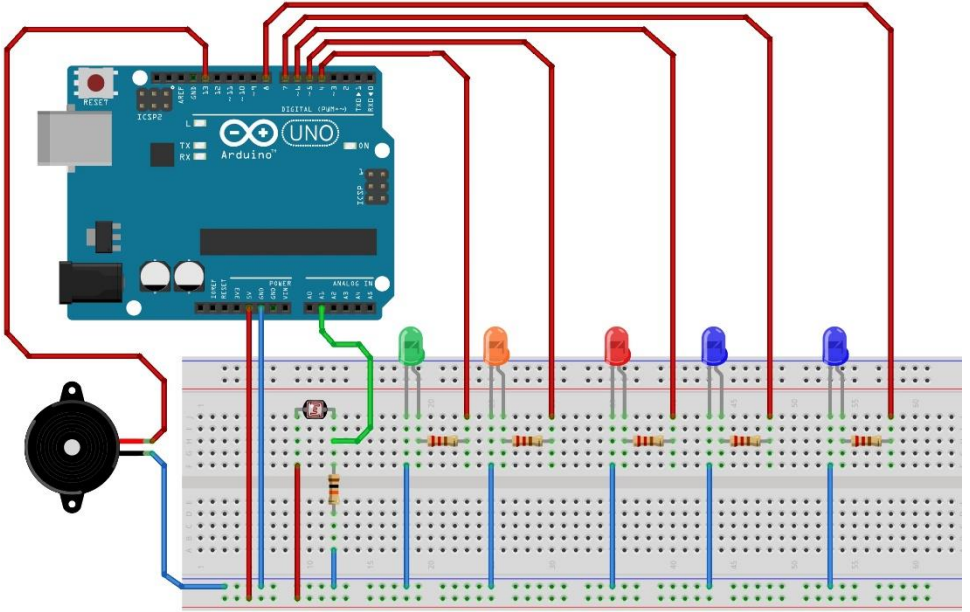
3. Arduino İle 4-5-6-7-8 Numaralı Sayısal Çıkış Pinleri Üzerinden Led Ve 13 Numaralı Sayısal Çıkış Pini Üzerinden Buzzer, A0 Analog Giriş Pini Üzerinden Potansiyometre Uygulaması



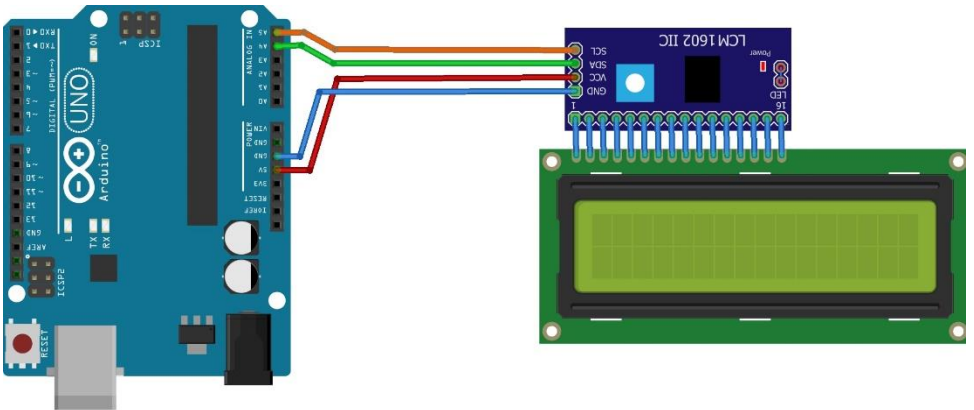
4. Arduino İle 2-3 Numaralı Sayısal Pinlere Buton, 13 Numaralı Sayısal Çıkış Pini Üzerinden Buzzer, A0 Analog Giriş Pini Üzerinden Potansiyometre Ve 9-10-11 Sayısal Çıkış Pinleri Üzerinden RGB-LED Uygulaması



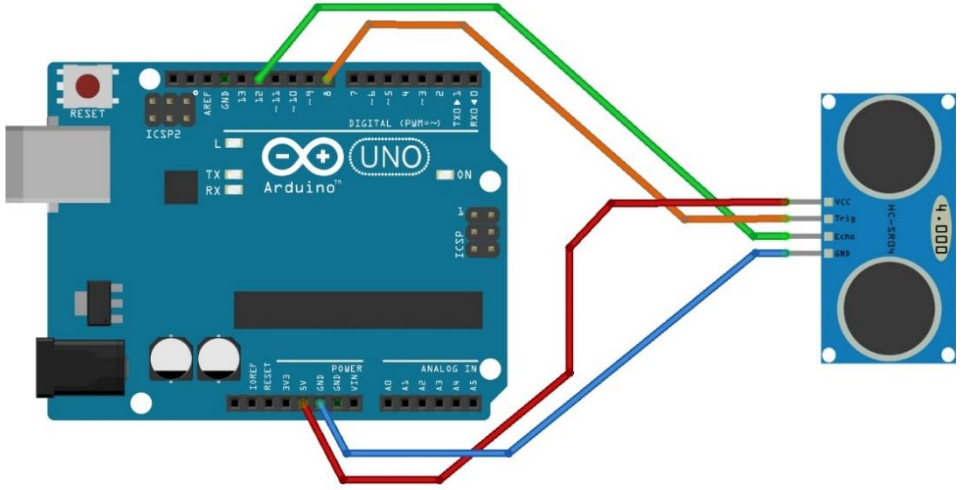
5. Arduino İle 4-5-6-7-8 Numaralı Sayısal Çıkış Pinleri Üzerinden Led Ve 13 Numaralı Sayısal Çıkış Pini Üzerinden Buzzer, A1 Analog Giriş Pini Üzerinden LDR Uygulaması



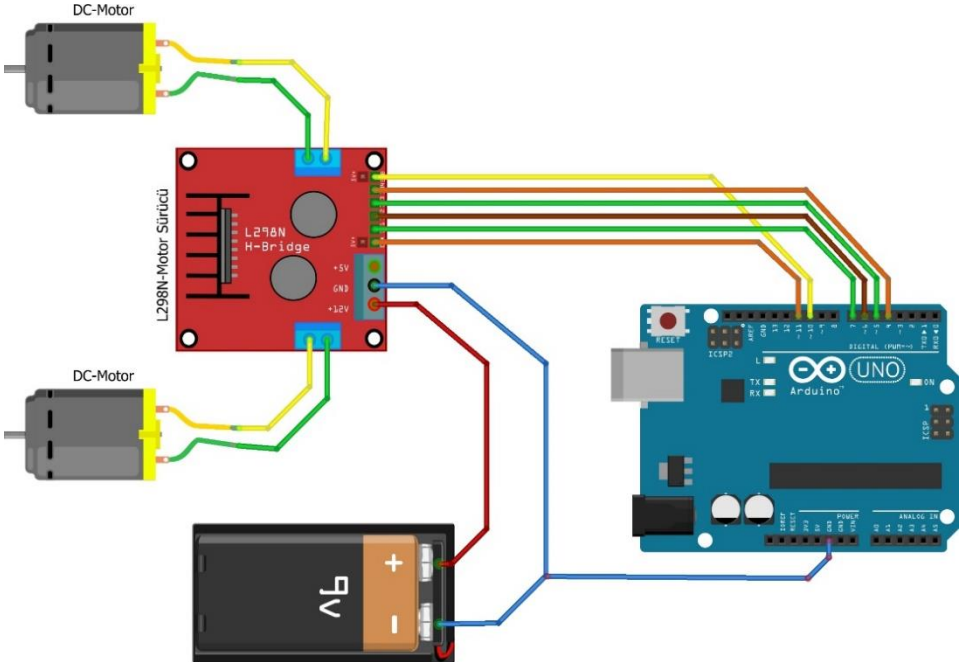
6. Arduino İle I2C – LCD Uygulaması



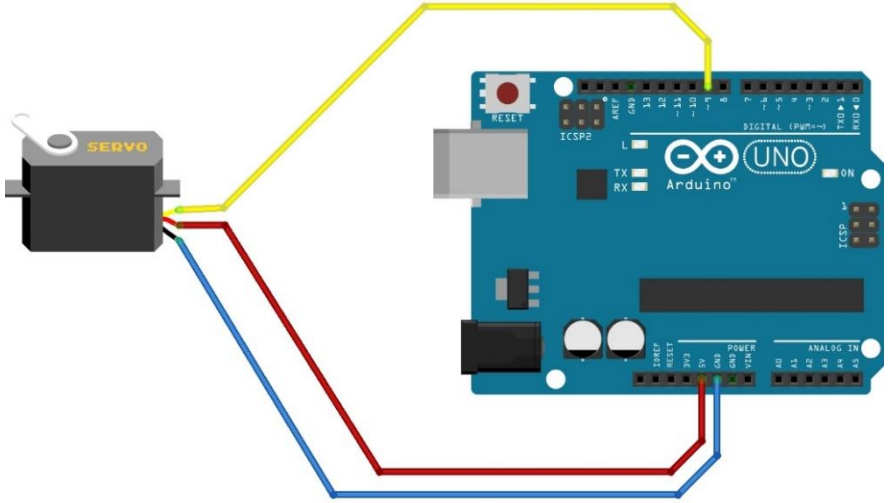
7. Arduino İle Mesafe Algılayıcı (HC-SR04) Uygulaması



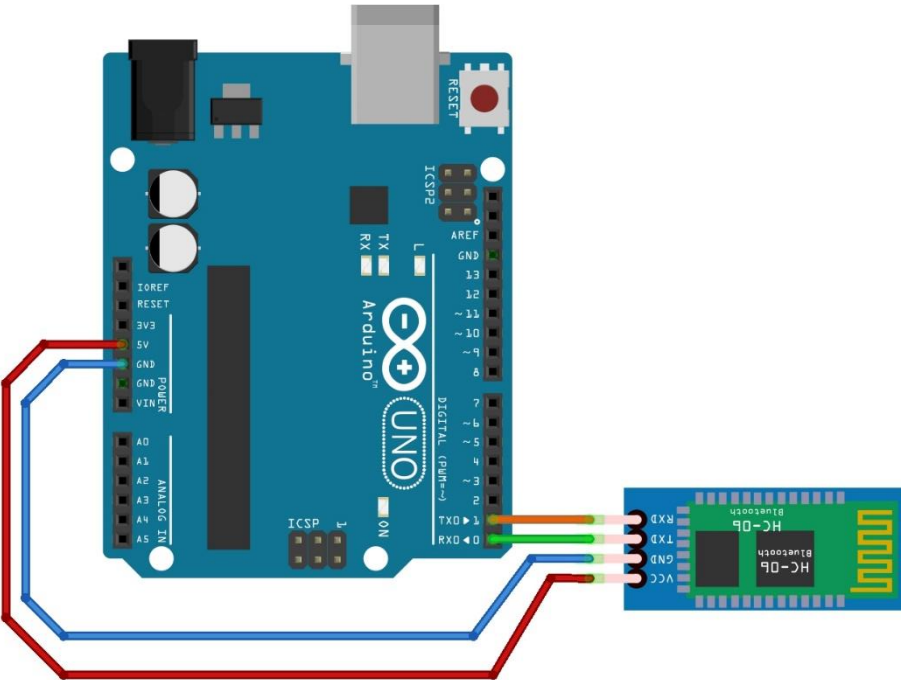
8. Arduino İle DC-Motor Uygulaması



9. Arduino İle Servo-Motor Uygulaması



10. Arduino İle Bluetooth (HC-06) Uygulaması



C. ARDUINO EĞİTİM SETİ TANITIMI

C.1. Genel Tanıtım ve Kullanım Amacı

Önceki bölümlerde belirtilen açıklamalar doğrultusunda, Trabzon İl Millî Eğitim Müdürlüğü AR-GE Birimi ile Trabzon Bilgisayar Bilimleri ve Kodlama Merkezi koordinasyonu kapsamında geliştirilen, “ARDUINO TEMEL EĞİTİM SETİ”, adı geçen tüm elektronik ve devre uygulama süreçlerini üzerinde bulunduran bir ürün olarak tasarlanmış ve içeriklendirilmiştir.

Sonraki aşamada ise, Trabzon Mesleki ve Teknik Anadolu Lisesi (TMTAL) Döner Sermaye İşletmesi bünyesinde, seri üretimine başlanmış ve halen devam edilmektedir.

Elektronik devre kurulum, bağlantı ve programlama aşamalarının tamamı göz önünde bulundurulduğunda, özellikle yeni başlayan kullanıcılar açısından, eş zamanlı bir şekilde, elektronik ve programlama uygulamalarının yürütülmesi büyük bir zorluk teşkil etmektedir. Bu amaçla, kullanıcıyı/programcıyı, elektronik iş ve işlem yükünden kurtararak, özellikle daha etkili kodlamaya yönlendirmek ve gelişim süreci içerisinde, kullanıcının ilgisini kaybetmesine imkan vermeden, gerekli durumlarda, gerekli miktarda elektronik altyapı ile karşı karşıya kalmasını sağlamak yoluyla, daha etkili bir “Robotik Kodlama” süreci meydana gelmesine sebep olmaktadır.

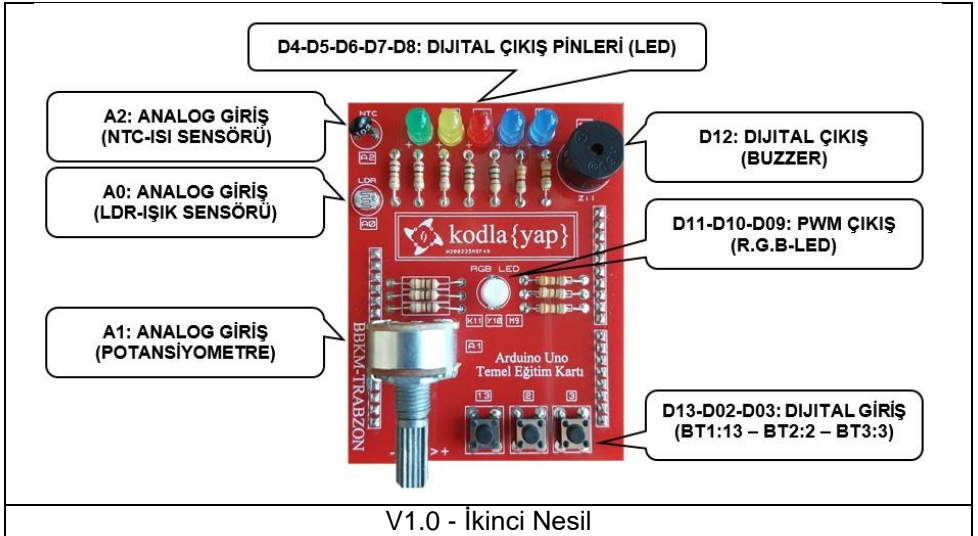
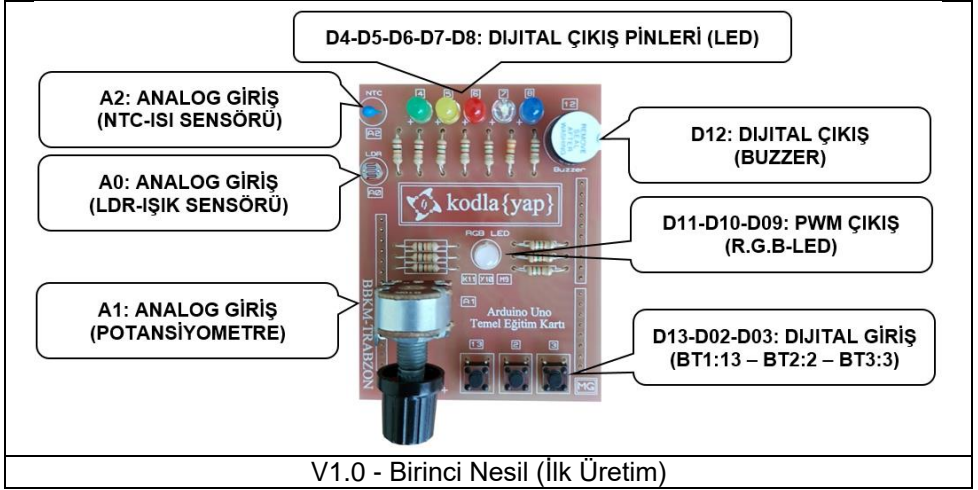
Bu amaçlar doğrultusunda, ilerleyen kısımlarda detaylı teknik tanıtımı yapılacak ve uygulama örnekleri sunulacak eğitim seti, özellikle Robotik Kodlama alanına yeni başlayanlar ve bu alanda eğitim süreçleri uygulayanlar için önemli bir eğitim materyali olabilecektir.

C.2. Çeşitleri ve Özellikleri

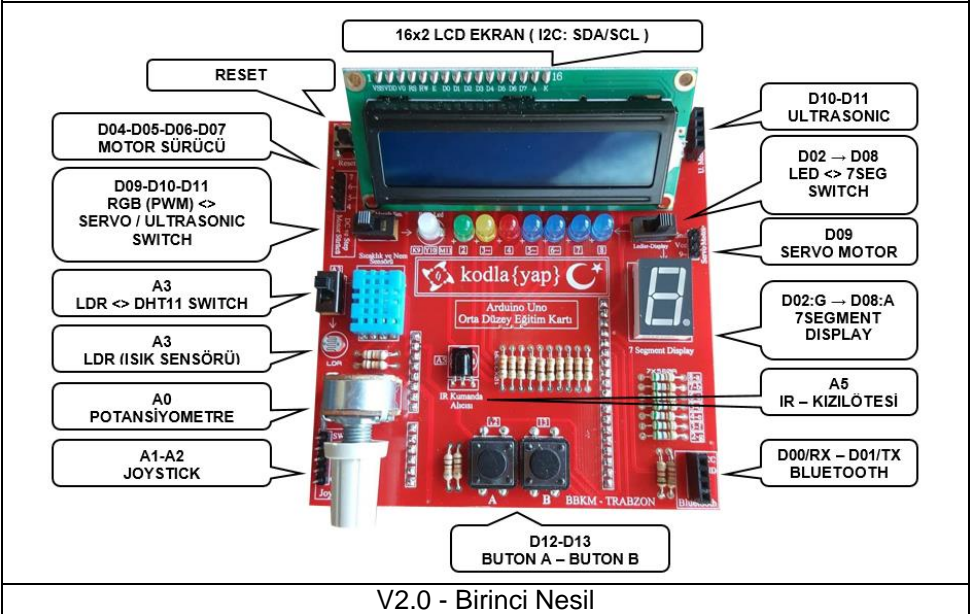
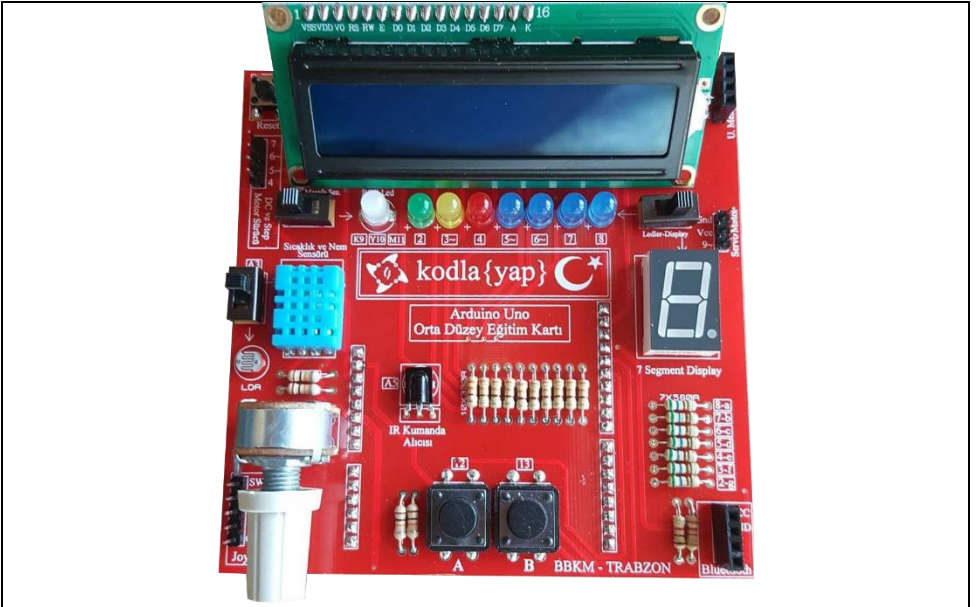
Özellikle kodlama/programlama süreçlerinde hızlı ilerleme sağlaması ve sonraki aşamalarda kullanıcının ilgisini kaybetmeden gerekli oranda elektronik süreçlerle karşı karşıya kalmasını sağlaması sebebiyle, eğitim süreçlerinde çok faydalı olduğu görülen “ARDUINO TEMEL EĞİTİM SETİ”, üç aşamalı bir ürünleştirme altyapısına sahiptir.

Birinci aşamada Versiyon 1.0 eğitim kartını barındıran materyal, genişletme elemanlarına uyumluluk taşınamaması sebebiyle, geliştirilmesine devam edilmiş ve ikinci aşamada Versiyon 2.0 ile dahili olarak birçok bileşeni üzerinde barındırmakla birlikte, tasarım, üretim, montaj, maliyet ve son kullanıcı taraflı arıza giderme noktasında zorluk çıkarmıştır. Son olarak üçüncü aşamada Versiyon 1.1 ile, tasarım, üretim, montaj ve son kullanıcı taraflı arıza giderme süreçleri optimize edilerek, özellikle V1.0 maliyetinde ve V2.0 yeteneklerinde, genişletme bağlantıları ile birçok ek algılayıcı ve anahtarlama elemanı bağlantısı yapılabilen bir ürün kimliğine kavuşmuştur.

İlerleyen kısımlarda bu materyallerin detaylı tanımlamaları yapılacak olup, gelenen son durumda V1.1 özelinde, TEMEL SEVİYE ve İLERİ SEVİYE olmak üzere iki aşamada, yaklaşık 100 farklı örnek BLOK ve KOD(C) tabanlı çözümlerle uygulanacaktır.

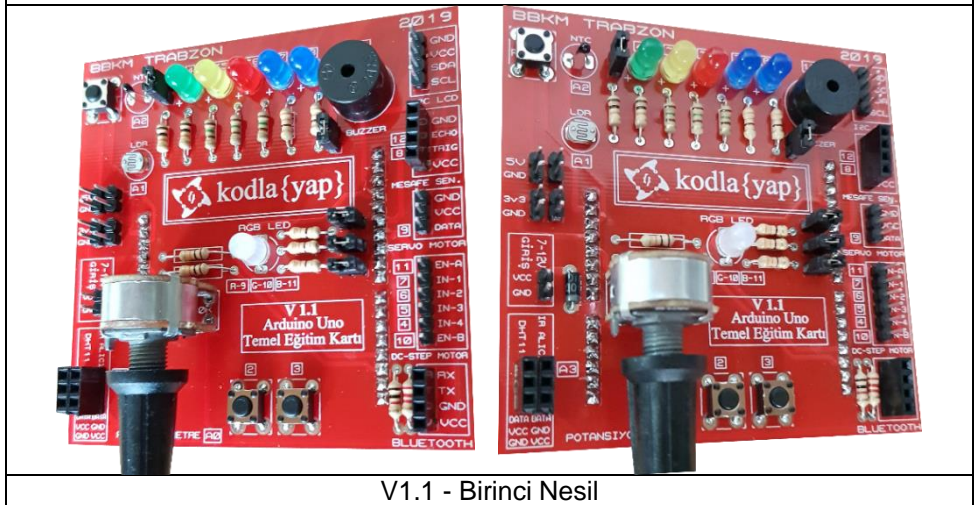
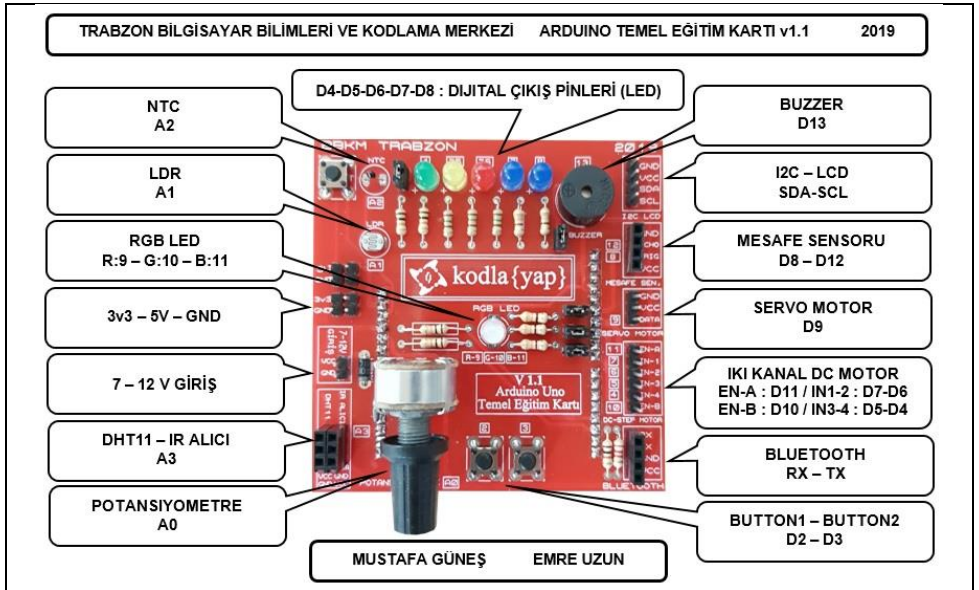


Şekilde belirtildiği üzere, düşük maliyet, kolay montaj ve kullanıcı dostu hızlı programlama eğitimi uygulamalarına uygun fakat geliştirme bağlantıları olmayan V1.0 eğitim kartı, okul öncesi, ilk okul ve orta okul seviyeleri için, ilgi çekici ve kolay uygulama imkanları sebebiyle ideal bir eğitim materyalidir.



V2.0 - Birinci Nesil

Birçok donanımsal programlama bileşenini, başta LCD ve 7SEG modüllerini dahili olarak sunan V2.0, yüksek maliyetinin yanında, montaj süresinin fazla olması ve son kullanıcı tarafı arıza ve hasar gidermedeki zorlukları sebebiyle, bireysel kullanım, lise veya üniversite seviyesi robotik kodlama eğitimlerinde daha ideal bir çözüm sunmaktadır.

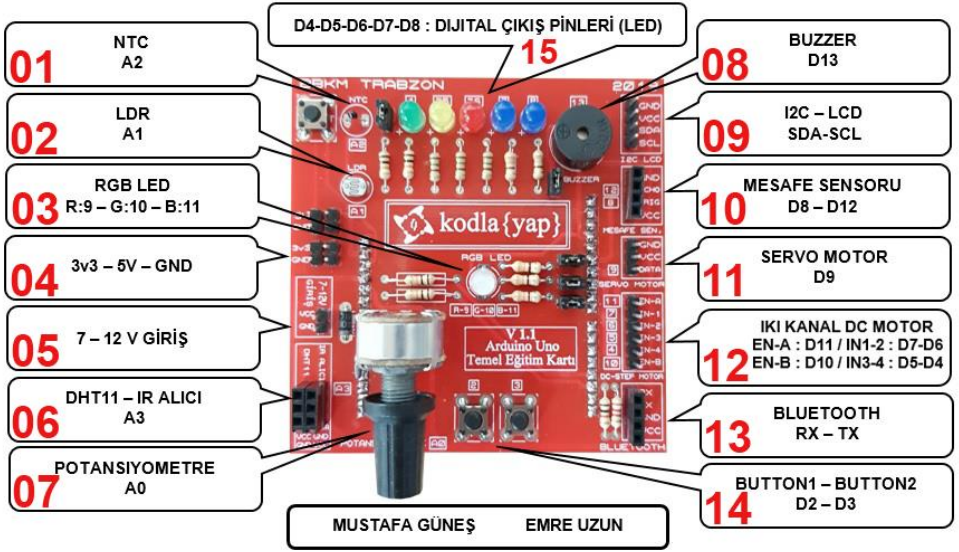


V1.1 - Birinci Nesil

Son olarak geliştirilen ve V1.0 ile V2.0 arası bir donanım özelliğine sahip olan V1.1, V1.0 basitliğinde ve V2.0 genişletme çeşitliliğine sahip, tam olarak okul öncesi ile üniversite aralığında her tür robotik kodlama seviyesine uygun kullanımda bir üründür. Tüm harici algılayıcı ve anahtarlama elemanları bağımsız olarak her tür kaynaktan temin edilebilmekte ve gerekli uygulamalarda, standart bağlantı noktalarına takılarak, gerekli yazılım geliştirme sürecine hızlı bir şekilde devam edilebilmektedir.

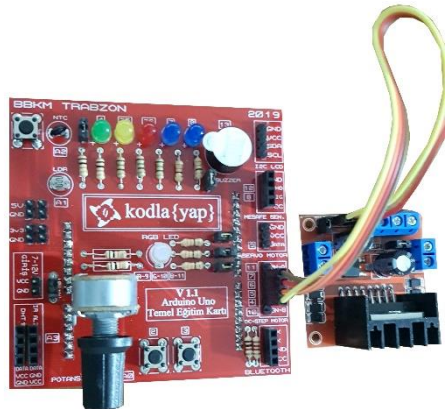
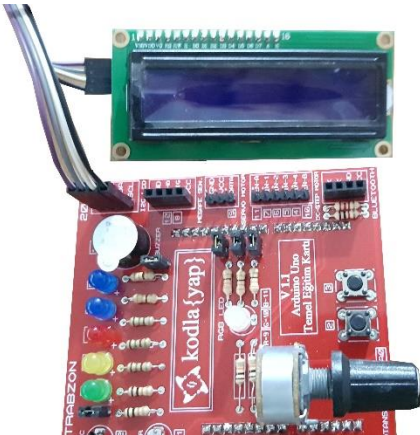
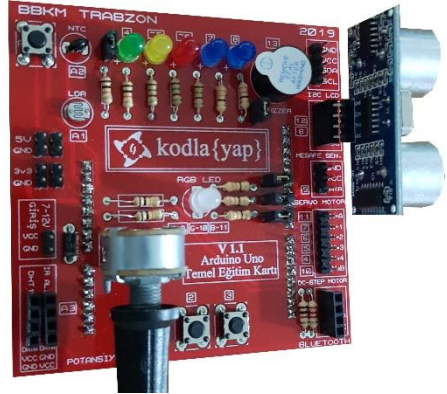
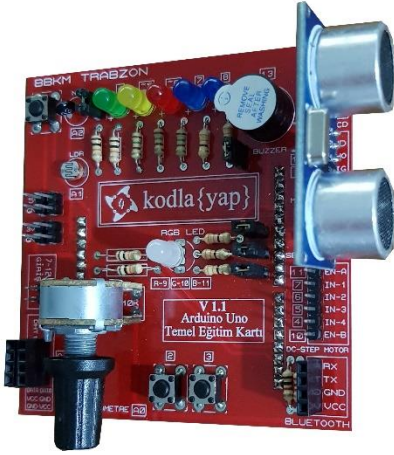
Bu kılavuz, V1.1-Birinci Nesil model robotik kodlama eğitim kartı uygulamaları üzerine hazırlanmıştır.

C.3. İç Yapısı ve Bileşenleri



- 01 ► Analog "A2" bağlantı noktasında, dahili NTC (ISI Algılayıcı)
- 02 ► Analog "A1" bağlantı noktasında, dahili LDR (IŞIK Algılayıcı)
- 03 ► Digital "9-10-11" bağlantı noktasında, RGB-LED
- 04 ► Harici 3v3 – 5V – GND bağlantı noktası
- 05 ► Harici 7 – 12 V besleme girişi bağlantı noktası
- 06 ► Analog "A3" bağlantı noktasında, genişletme soketi DHT11 (ISI-NEM Algılayıcı) veya IR (KızılÖtesi) bağlantısı
- 07 ► Analog "A0" bağlantı noktasında, dahili POTANSİYOMETRE
- 08 ► Digital "13" bağlantı noktasında, dahili BUZZER
- 09 ► I2C bağlantı noktasında, genişletme soketi LCD bağlantısı
- 10 ► Digital "8-12" bağlantı noktasında, genişletme soketi Ultrasonic Mesafe Algılayıcı (HC-SR04) bağlantısı
- 11 ► Digital "9" bağlantı noktasında, genişletme soketi SERVO Motor bağlantısı
- 12 ► Digital "4-5-6-7-10-11" bağlantı noktasında, genişletme soketi İki kanal DC Motor Sürücü bağlantısı
- 13 ► Digital "0-1" bağlantı noktasında, genişletme soketi Bluetooth (HC-06) bağlantısı
- 14 ► Digital "2-3" bağlantı noktasında, dahili BUTTON
- 15 ► Digital "4-5-6-7-8" bağlantı noktasında, dahili 5 adet LED

C.4. Geniřletme Elemanları



D. ARDUINO EĞİTİM SETİ UYGULAMALARI (TEMEL SEVİYE)

D.1. LED Uygulamaları

1. Bir LED Yak-Söndür (Blink)

```
Arduino Programı
sürekli tekrarla
4 sayısal pini YÜKSEK yap
1 saniye bekle
4 sayısal pini DÜŞÜK yap
1 saniye bekle
```

2. İki LED Yak-Söndür (Flip-Flop)

```
Arduino Programı
sürekli tekrarla
4 sayısal pini YÜKSEK yap
1 saniye bekle
4 sayısal pini DÜŞÜK yap
1 saniye bekle
5 sayısal pini YÜKSEK yap
1 saniye bekle
5 sayısal pini DÜŞÜK yap
1 saniye bekle
```

3. Yürüyen Işık Uygulaması (Eklenerek Yanan LED-ler)

```
Arduino Programı
sürekli tekrarla
4 sayısal pini YÜKSEK yap
1 saniye bekle
5 sayısal pini YÜKSEK yap
1 saniye bekle
6 sayısal pini YÜKSEK yap
1 saniye bekle
7 sayısal pini YÜKSEK yap
1 saniye bekle
8 sayısal pini YÜKSEK yap
1 saniye bekle
4 sayısal pini DÜŞÜK yap
1 saniye bekle
5 sayısal pini DÜŞÜK yap
1 saniye bekle
6 sayısal pini DÜŞÜK yap
1 saniye bekle
7 sayısal pini DÜŞÜK yap
1 saniye bekle
8 sayısal pini DÜŞÜK yap
1 saniye bekle
```


4. Kara Şimşek Uygulaması (Tek Tek Yanarak Sağa-Sola İlerleme)

```
Arduino Programı
sürekli tekrarla
4 sayısal pini YÜKSEK yap
1 saniye bekle
4 sayısal pini DÜŞÜK yap
5 sayısal pini YÜKSEK yap
1 saniye bekle
5 sayısal pini DÜŞÜK yap
6 sayısal pini YÜKSEK yap
1 saniye bekle
6 sayısal pini DÜŞÜK yap
7 sayısal pini YÜKSEK yap
1 saniye bekle
7 sayısal pini DÜŞÜK yap
8 sayısal pini YÜKSEK yap
1 saniye bekle
8 sayısal pini DÜŞÜK yap
7 sayısal pini YÜKSEK yap
1 saniye bekle
7 sayısal pini DÜŞÜK yap
6 sayısal pini YÜKSEK yap
1 saniye bekle
6 sayısal pini DÜŞÜK yap
5 sayısal pini YÜKSEK yap
1 saniye bekle
5 sayısal pini DÜŞÜK yap
```

D.2. BUTTON Uygulamaları

5. Bir BUTTON İle LED Yakma (Bırakınca Sönen)

```
Arduino Programı
sürekli tekrarla
eğer 2 sayısal pini oku = 1 ise
8 sayısal pini YÜKSEK yap
değilse
8 sayısal pini DÜŞÜK yap
```

6. Bir BUTTON İle BUZZER Çalıştırma (Bırakınca Duran)

```
Arduino Programı
sürekli tekrarla
eğer 3 sayısal pini oku = 1 ise
13 sayısal pini YÜKSEK yap
değilse
13 sayısal pini DÜŞÜK yap
```

7. Bir BUTTON İle Yakılan LED-in İkinci BUTTON İle Söndürülmesi

```
Arduino Programı
sürekli tekrarla
eğer 2 sayısal pini oku = 1 ise
8 sayısal pini YÜKSEK yap
eğer 3 sayısal pini oku = 1 ise
8 sayısal pini DÜŞÜK yap
```

8. Bir BUTTON İle BUZZER Çalıştırma, İkinci BUTTON İle Durdurulması

```
Arduino Programı
sürekli tekrarla
eğer 2 sayısal pini oku = 1 ise
  13 sayısal pini YÜKSEK yap
eğer 3 sayısal pini oku = 1 ise
  13 sayısal pini DÜŞÜK yap
```

9. Bir BUTTON İle Yakılan LED-in, Aynı BUTTON İle Söndürülmesi

```
Arduino Programı
LED_DURUM 0 olsun
sürekli tekrarla
eğer 2 sayısal pini oku = 1 ise
  LED_DURUM 1 - LED_DURUM olsun
  8 sayısal pini LED_DURUM yap
  0.005 saniye bekle
```

10. Bir BUTTON İle BUZZER Çalıştırma, Aynı BUTTON İle Durdurulması

```
Arduino Programı
BUZZER_DURUM 0 olsun
sürekli tekrarla
eğer 3 sayısal pini oku = 1 ise
  BUZZER_DURUM 1 - BUZZER_DURUM olsun
  13 sayısal pini BUZZER_DURUM yap
  0.005 saniye bekle
```

D.3. SERİ İLETİŞİM Uygulamaları

11. Serial Monitor Giriş-Çıkış Uygulaması (PRINT)

```
Arduino Programı
sürekli tekrarla
seri porta TRABZON yaz
```

12. Serial Monitor Giriş-Çıkış Uygulaması (READ)

```
Arduino Programı
sürekli tekrarla
eğer seri portta byte var > 0 ise
  seri porta seri porttan byte oku yaz
  1 saniye bekle
```


D.4. POTANSİYOMETRE Uygulamaları

13. POTANSİYOMETRE İle Okunan Analog Değerin Yazdırılması

```
Arduino Programı
sürekli tekrarla
seri porta (A) 0 analog pini oku yaz
0.1 saniye bekle
```

14. POTANSİYOMETRE İle LED Yanıp-Sönme Hızının Değiştirilmesi

```
Arduino Programı
sürekli tekrarla
POT_DEGER, (A) 0 analog pini oku olsun
seri porta POT_DEGER yaz
BEKLEME_SURESI, POT_DEGER / 1023 olsun
seri porta BEKLEME_SURESI yaz
8 sayisal pini YÜKSEK yap
BEKLEME_SURESI saniye bekle
8 sayisal pini DÜŞÜK yap
BEKLEME_SURESI saniye bekle
```

15. POTANSİYOMETRE İle Analog Çıkış LED Parlaklık Ayarı

```
Arduino Programı
sürekli tekrarla
POT_DEGER, (A) 0 analog pini oku olsun
seri porta POT_DEGER yaz
PARLAKLIK_DEGERI, POT_DEGER / 4 olsun
6 pwm pini PARLAKLIK_DEGERI yap
```

16. POTANSİYOMETRE İle BUZZER Çalışma Zaman Aralığının Değiştirilmesi

```
Arduino Programı
sürekli tekrarla
POT_DEGER, (A) 0 analog pini oku olsun
seri porta POT_DEGER yaz
BEKLEME_SURESI, POT_DEGER / 1023 olsun
seri porta BEKLEME_SURESI yaz
13 sayisal pini YÜKSEK yap
BEKLEME_SURESI saniye bekle
13 sayisal pini DÜŞÜK yap
BEKLEME_SURESI saniye bekle
```

17. POTANSİYOMETRE İle LED Seviyesi (Eklenerek Yanan)

```
Arduino Programı
sürekli tekrarla
POT_DEGER (A) 0 analog pini oku olsun
seri porta POT_DEGER yaz
eğer POT_DEGER == 0 ise
  4 sayısal pini DÜŞÜK yap
  5 sayısal pini DÜŞÜK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 0 ve POT_DEGER < 206 ise
  4 sayısal pini YÜKSEK yap
  5 sayısal pini DÜŞÜK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 205 ve POT_DEGER < 411 ise
  4 sayısal pini YÜKSEK yap
  5 sayısal pini YÜKSEK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 410 ve POT_DEGER < 616 ise
  4 sayısal pini YÜKSEK yap
  5 sayısal pini YÜKSEK yap
  6 sayısal pini YÜKSEK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 615 ve POT_DEGER < 821 ise
  4 sayısal pini YÜKSEK yap
  5 sayısal pini YÜKSEK yap
  6 sayısal pini YÜKSEK yap
  7 sayısal pini YÜKSEK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 820 ve POT_DEGER < 1024 ise
  4 sayısal pini YÜKSEK yap
  5 sayısal pini YÜKSEK yap
  6 sayısal pini YÜKSEK yap
  7 sayısal pini YÜKSEK yap
  8 sayısal pini YÜKSEK yap
```

18. POTANSİYOMETRE İle Tek Tek Yanarak Sağa-Sola İlerleyen LED

```
Arduino Programı
sürekli tekrarla
POT_DEGER (A) 0 analog pini oku olsun
seri porta POT_DEGER yaz
eğer POT_DEGER == 0 ise
  4 sayısal pini DÜŞÜK yap
  5 sayısal pini DÜŞÜK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 0 ve POT_DEGER < 206 ise
  4 sayısal pini YÜKSEK yap
  5 sayısal pini DÜŞÜK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 205 ve POT_DEGER < 411 ise
  4 sayısal pini DÜŞÜK yap
  5 sayısal pini YÜKSEK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 410 ve POT_DEGER < 616 ise
  4 sayısal pini DÜŞÜK yap
  5 sayısal pini DÜŞÜK yap
  6 sayısal pini YÜKSEK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 615 ve POT_DEGER < 821 ise
  4 sayısal pini DÜŞÜK yap
  5 sayısal pini DÜŞÜK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini YÜKSEK yap
  8 sayısal pini DÜŞÜK yap
eğer POT_DEGER > 820 ve POT_DEGER < 1024 ise
  4 sayısal pini DÜŞÜK yap
  5 sayısal pini DÜŞÜK yap
  6 sayısal pini DÜŞÜK yap
  7 sayısal pini DÜŞÜK yap
  8 sayısal pini YÜKSEK yap
```

D.5. RGB-LED Uygulamaları

19. Sıra İle R-G-B 3 Ve 7 Renk Ayrı Ayrı Yakma

Arduino Programı

sürekli tekrarla

9 sayısal pini DÜŞÜK yap

10 sayısal pini DÜŞÜK yap

11 sayısal pini DÜŞÜK yap

1 saniye bekle

9 sayısal pini YÜKSEK yap

10 sayısal pini DÜŞÜK yap

11 sayısal pini DÜŞÜK yap

1 saniye bekle

9 sayısal pini DÜŞÜK yap

10 sayısal pini YÜKSEK yap

11 sayısal pini DÜŞÜK yap

1 saniye bekle

9 sayısal pini DÜŞÜK yap

10 sayısal pini DÜŞÜK yap

11 sayısal pini YÜKSEK yap

1 saniye bekle

20. BUTTON Kontrollü R-G-B 3 Ve 7 Renk Ayrı Ayrı Yakma

Arduino Programı

RGB_SIRA_NO , 0 olsun

sürekli tekrarla

eğer 2 sayısal pini oku = 1 ise

RGB_SIRA_NO 'i -1 arttır

eğer RGB_SIRA_NO < 0 ise

RGB_SIRA_NO , 0 olsun

2 sayısal pini oku = 0 olana kadar bekle

0.01 saniye bekle

eğer 3 sayısal pini oku = 1 ise

RGB_SIRA_NO 'i 1 arttır

eğer RGB_SIRA_NO > 3 ise

RGB_SIRA_NO , 3 olsun

3 sayısal pini oku = 0 olana kadar bekle

0.01 saniye bekle

eğer RGB_SIRA_NO = 0 ise

9 sayısal pini DÜŞÜK yap

10 sayısal pini DÜŞÜK yap

11 sayısal pini DÜŞÜK yap

eğer RGB_SIRA_NO = 1 ise

9 sayısal pini YÜKSEK yap

10 sayısal pini DÜŞÜK yap

11 sayısal pini DÜŞÜK yap

eğer RGB_SIRA_NO = 2 ise

9 sayısal pini DÜŞÜK yap

10 sayısal pini YÜKSEK yap

11 sayısal pini DÜŞÜK yap

eğer RGB_SIRA_NO = 3 ise

9 sayısal pini DÜŞÜK yap

10 sayısal pini DÜŞÜK yap

11 sayısal pini YÜKSEK yap

21. POTANSİYOMETRE İle R-G-B 3 Ve 7 Renk Ayır Ayrı Yakma

```
Arduino Programı
sürekli tekrarla
POT_DEGER, (A) 0 analog pini oku olsun
eğer POT_DEGER = 0 ise
  RGB_SIRA_NO, 0 olsun
eğer POT_DEGER > 0 ve POT_DEGER < 341 ise
  RGB_SIRA_NO, 1 olsun
eğer POT_DEGER > 340 ve POT_DEGER < 682 ise
  RGB_SIRA_NO, 2 olsun
eğer POT_DEGER > 681 ve POT_DEGER < 1024 ise
  RGB_SIRA_NO, 3 olsun
eğer RGB_SIRA_NO = 0 ise
  9 sayısal pini DÜŞÜK yap
  10 sayısal pini DÜŞÜK yap
  11 sayısal pini DÜŞÜK yap
eğer RGB_SIRA_NO = 1 ise
  9 sayısal pini YÜKSEK yap
  10 sayısal pini DÜŞÜK yap
  11 sayısal pini DÜŞÜK yap
eğer RGB_SIRA_NO = 2 ise
  9 sayısal pini DÜŞÜK yap
  10 sayısal pini YÜKSEK yap
  11 sayısal pini DÜŞÜK yap
eğer RGB_SIRA_NO = 3 ise
  9 sayısal pini DÜŞÜK yap
  10 sayısal pini DÜŞÜK yap
  11 sayısal pini YÜKSEK yap
```

22. Döngü İle R-G-B 3 Ve 7 Renk Ayır Ayrı Yakma

```
Arduino Programı
sürekli tekrarla
RGB_SIRA_NO, 0 olsun
RGB_SIRA_NO > 3 olana kadar tekrarla
eğer RGB_SIRA_NO = 0 ise
  9 sayısal pini DÜŞÜK yap
  10 sayısal pini DÜŞÜK yap
  11 sayısal pini DÜŞÜK yap
eğer RGB_SIRA_NO = 1 ise
  9 sayısal pini YÜKSEK yap
  10 sayısal pini DÜŞÜK yap
  11 sayısal pini DÜŞÜK yap
eğer RGB_SIRA_NO = 2 ise
  9 sayısal pini DÜŞÜK yap
  10 sayısal pini YÜKSEK yap
  11 sayısal pini DÜŞÜK yap
eğer RGB_SIRA_NO = 3 ise
  9 sayısal pini DÜŞÜK yap
  10 sayısal pini DÜŞÜK yap
  11 sayısal pini YÜKSEK yap
RGB_SIRA_NO, 1 arttır
0.5 saniye bekle
```

D.6. ISI ve IŞIK Uygulamaları

23. LDR İle Analog Işık Şiddeti Değeri Serial Monitörde Yazdırma



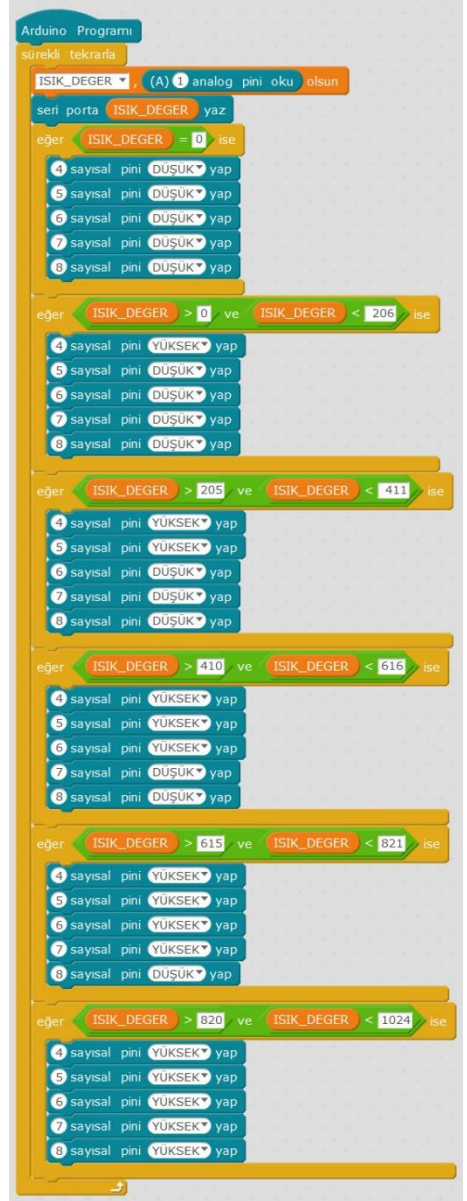
```
Arduino Programı
sürekli tekrarla
  ISIK_DEGERI , (A) 1 analog pini oku olsun
  seri porta ISIK_DEGERI yaz
  0.5 saniye bekle
```

24. LDR İle Işık Şiddetine Bağlı LED Kontrolü (Yak-Söndür)



```
Arduino Programı
sürekli tekrarla
  ISIK_DEGERI , (A) 1 analog pini oku olsun
  seri porta ISIK_DEGERI yaz
  eğer ISIK_DEGERI < 300 ise
    9 sayısal pini YÜKSEK yap
  değilse
    9 sayısal pini DÜŞÜK yap
```

25. LDR İle Işık Şiddetine Bağlı LED Kontrolü (Eklenerak Yanan LED)



```
Arduino Programı
sürekli tekrarla
  ISIK_DEGERI , (A) 1 analog pini oku olsun
  seri porta ISIK_DEGERI yaz
  eğer ISIK_DEGERI = 0 ise
    4 sayısal pini DÜŞÜK yap
    5 sayısal pini DÜŞÜK yap
    6 sayısal pini DÜŞÜK yap
    7 sayısal pini DÜŞÜK yap
    8 sayısal pini DÜŞÜK yap
  eğer ISIK_DEGERI > 0 ve ISIK_DEGERI < 206 ise
    4 sayısal pini YÜKSEK yap
    5 sayısal pini DÜŞÜK yap
    6 sayısal pini DÜŞÜK yap
    7 sayısal pini DÜŞÜK yap
    8 sayısal pini DÜŞÜK yap
  eğer ISIK_DEGERI > 205 ve ISIK_DEGERI < 411 ise
    4 sayısal pini YÜKSEK yap
    5 sayısal pini YÜKSEK yap
    6 sayısal pini DÜŞÜK yap
    7 sayısal pini DÜŞÜK yap
    8 sayısal pini DÜŞÜK yap
  eğer ISIK_DEGERI > 410 ve ISIK_DEGERI < 616 ise
    4 sayısal pini YÜKSEK yap
    5 sayısal pini YÜKSEK yap
    6 sayısal pini YÜKSEK yap
    7 sayısal pini DÜŞÜK yap
    8 sayısal pini DÜŞÜK yap
  eğer ISIK_DEGERI > 615 ve ISIK_DEGERI < 821 ise
    4 sayısal pini YÜKSEK yap
    5 sayısal pini YÜKSEK yap
    6 sayısal pini YÜKSEK yap
    7 sayısal pini YÜKSEK yap
    8 sayısal pini DÜŞÜK yap
  eğer ISIK_DEGERI > 820 ve ISIK_DEGERI < 1024 ise
    4 sayısal pini YÜKSEK yap
    5 sayısal pini YÜKSEK yap
    6 sayısal pini YÜKSEK yap
    7 sayısal pini YÜKSEK yap
    8 sayısal pini YÜKSEK yap
```

26. LDR İle Işığın Şiddetine Bağlı LED Kontrolü (Tek Tek Sağa-Sola İlerleme)

```
Arduino Programı
sürekli tekrarla
ISIK_DEGER , (A) 1 analog pini oku olsun
seri porta ISIK_DEGER yaz
eğer ISIK_DEGER = 0 ise
4 sayısal pini DÜŞÜK yap
5 sayısal pini DÜŞÜK yap
6 sayısal pini DÜŞÜK yap
7 sayısal pini DÜŞÜK yap
8 sayısal pini DÜŞÜK yap
eğer ISIK_DEGER > 0 ve ISIK_DEGER < 206 ise
4 sayısal pini YÜKSEK yap
5 sayısal pini DÜŞÜK yap
6 sayısal pini DÜŞÜK yap
7 sayısal pini DÜŞÜK yap
8 sayısal pini DÜŞÜK yap
eğer ISIK_DEGER > 205 ve ISIK_DEGER < 411 ise
4 sayısal pini DÜŞÜK yap
5 sayısal pini YÜKSEK yap
6 sayısal pini DÜŞÜK yap
7 sayısal pini DÜŞÜK yap
8 sayısal pini DÜŞÜK yap
eğer ISIK_DEGER > 410 ve ISIK_DEGER < 616 ise
4 sayısal pini DÜŞÜK yap
5 sayısal pini DÜŞÜK yap
6 sayısal pini YÜKSEK yap
7 sayısal pini DÜŞÜK yap
8 sayısal pini DÜŞÜK yap
eğer ISIK_DEGER > 615 ve ISIK_DEGER < 821 ise
4 sayısal pini DÜŞÜK yap
5 sayısal pini DÜŞÜK yap
6 sayısal pini DÜŞÜK yap
7 sayısal pini YÜKSEK yap
8 sayısal pini DÜŞÜK yap
eğer ISIK_DEGER > 820 ve ISIK_DEGER < 1024 ise
4 sayısal pini DÜŞÜK yap
5 sayısal pini DÜŞÜK yap
6 sayısal pini DÜŞÜK yap
7 sayısal pini DÜŞÜK yap
8 sayısal pini YÜKSEK yap
```

27. NTC İle Analog Isı Değerini Serial Monitörde Yazdırma

```
Arduino Programı
sürekli tekrarla
ISI_DEGERI , (A) 2 analog pini oku olsun
seri porta ISI_DEGERI yaz
0.5 saniye bekle
```

28. NTC İle Sıcaklığa Bağlı LED Kontrolü (Yak-Söndür)

```
Arduino Programı
sürekli tekrarla
ISI_DEGERI , (A) 2 analog pini oku olsun
seri porta ISI_DEGERI yaz
eğer ISI_DEGERI > 600 ise
9 sayısal pini YÜKSEK yap
değilse
9 sayısal pini DÜŞÜK yap
```