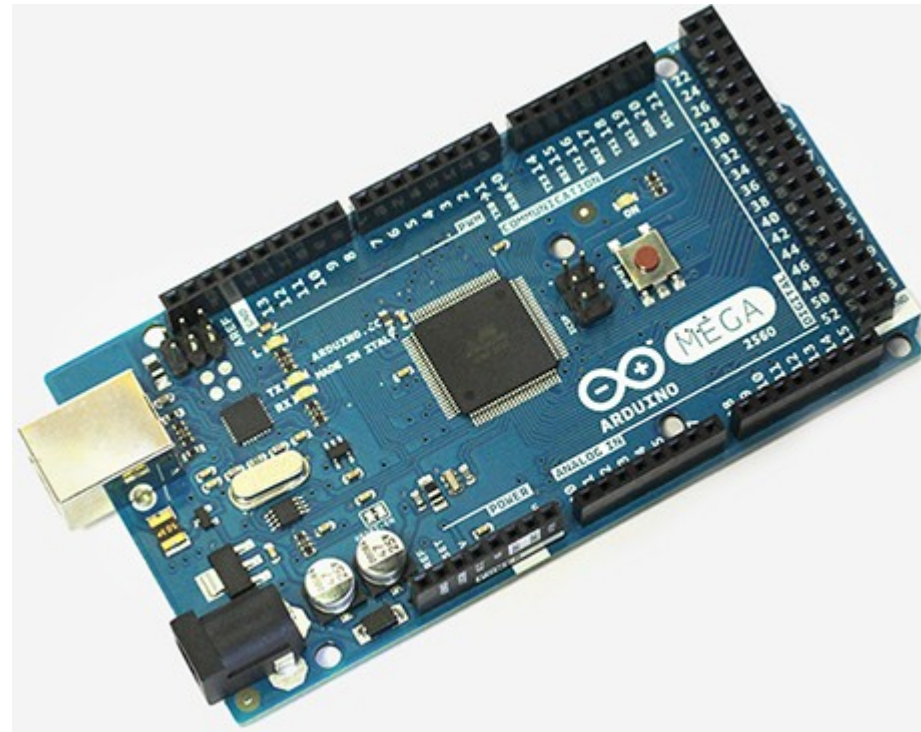


ARDUINO

Dr. Fahri VATANSEVER

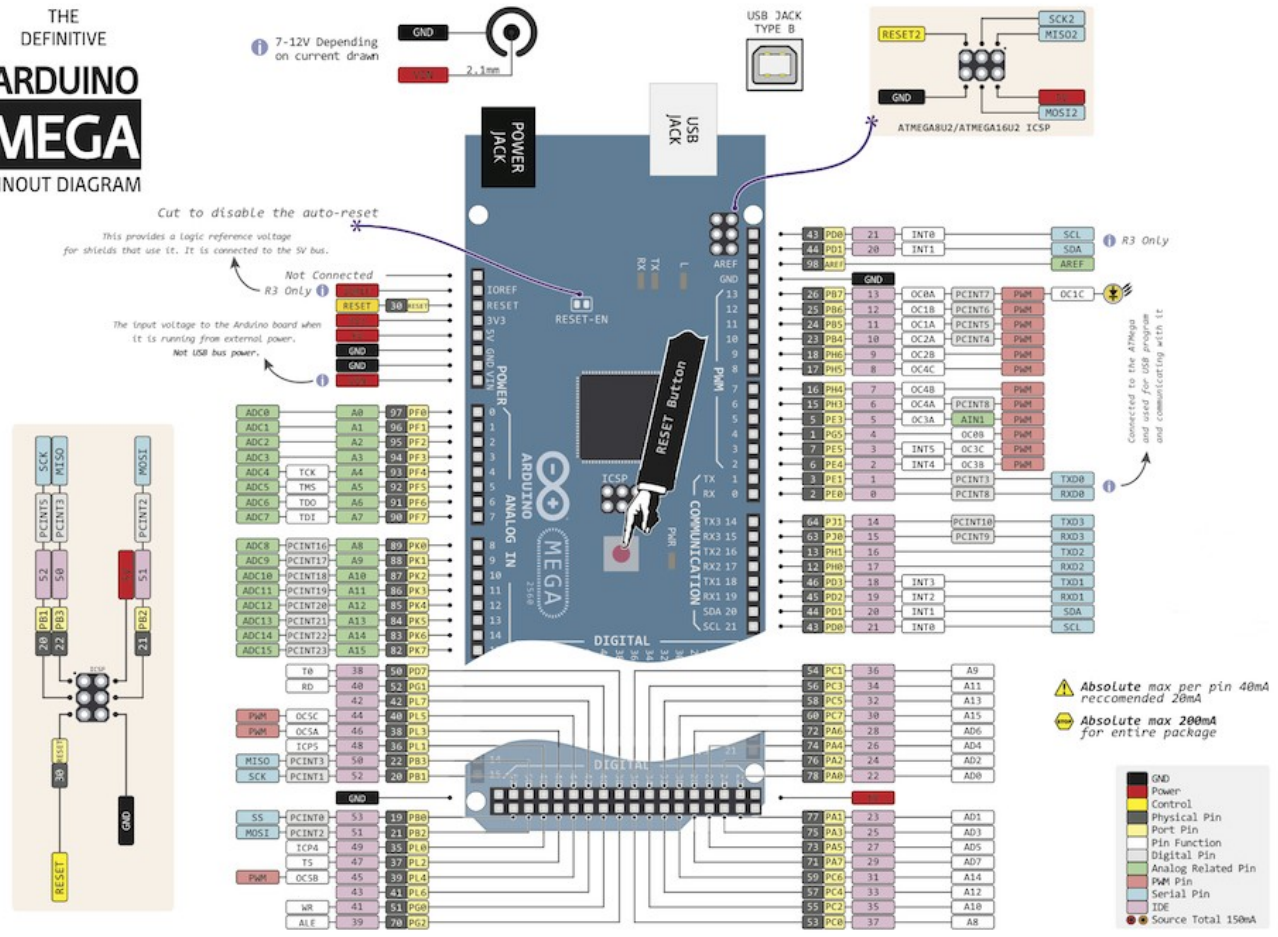
Arduino hakkında daha fazla bilgi için F. Vatansever, "Algoritma Geliştirme ve Programlamaya Giriş", 14. baskı, Seçkin Yayıncılık, Ankara, 2020 kitabından faydalanabilirsiniz.

Arduino Mega 2560 R3



Arduino Mega 2560 R3

THE DEFINITIVE ARDUINO MEGA PINOUT DIAGRAM

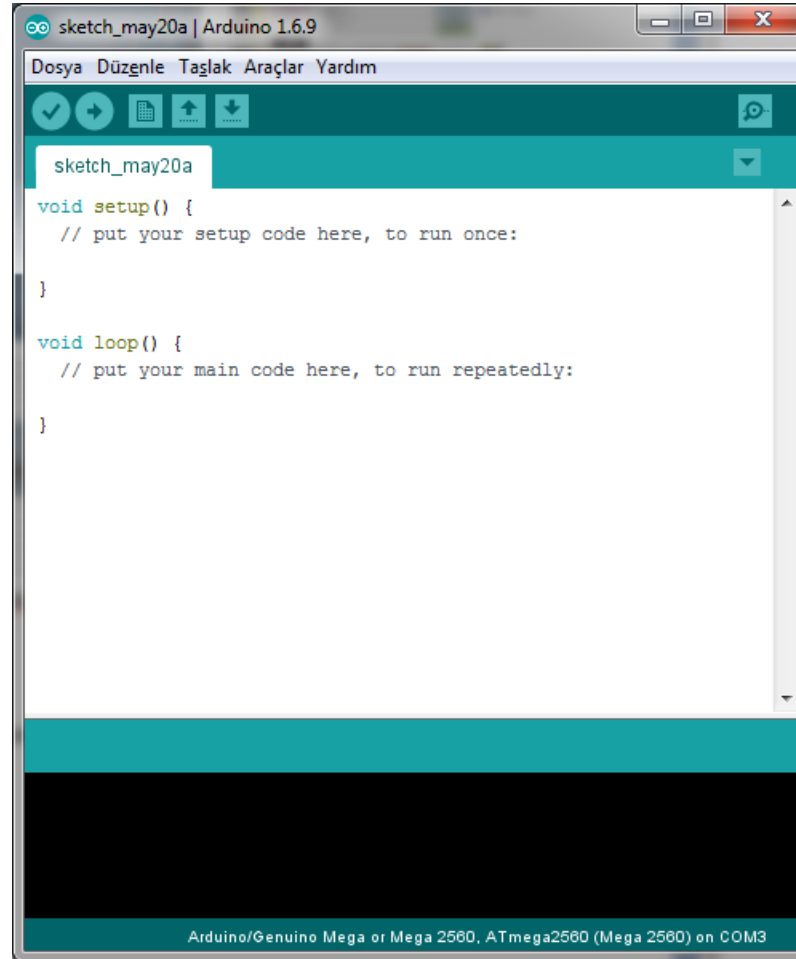


Arduino Mega 2560 R3

Tablo-1. Arduino Mega 2560 R3 teknik özellikler

İşlemci	ATmega2560 (8-bit Atmel mikroişlemci)
Çalışma gerilimi	5 V
Giriş gerilimi (önerilen)	7-12 V
Giriş gerilimi (sınır değerler)	6-20 V
Dijital I/O pinleri	54 (15 tanesi PWM çıkış)
Analog giriş pinleri	16
Her I/O pini için DC akım	20 mA
3.3 V pini için DC akım	50 mA
Flash bellek	256 KB (8 KB bootloader tarafından kullanılmaktadır)
SRAM	8 KB
EEPROM	4 KB
Saat (Clock) hızı	16 MHz
Uzunluk	101.52 mm
Genişlik	53.3 mm
Ağırlık	37 gr

Arduino Yazılım Geliştirme Ortamı



The image shows a screenshot of the Arduino IDE (version 1.6.9) interface. The window title is "sketch_may20a | Arduino 1.6.9". The menu bar includes "Dosya", "Düzenle", "Taçlak", "Araçlar", and "Yardım". The toolbar contains icons for saving, undo, redo, and other functions. The main editor area shows a sketch named "sketch_may20a" with the following code:

```
void setup() {  
  // put your setup code here, to run once:  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
}
```

The status bar at the bottom indicates the board and port: "Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM3".

Arduino Programları

```
// Genel (global) tanımlamalar

void setup() {

// kurulum kodları yazılır ve bir defa çalışır

}

void loop() {

// ana kodlar yazılır ve sürekli (tekrarlamalı) çalışır

}
```

Arduino Programlarının Temel Yazım Özellikleri

- Program yazımları belirli kalıpta, bloklar halinde olur.
- Bloklar, { } parantezleri ile oluşturulur.
- Komutlar aynı veya alt alta satırlara yazılabilirler.
- Tüm komutlar, noktalı virgül (;) ile biter. Yalnız blok başlatan ifadelerden sonra noktalı virgül kullanılmaz.
- Programda kullanılan tüm değişkenler ve bilgi tipleri bildirilmelidir.
- Programın başında kütüphaneler aktifleştirilir/çağrılır.
- Açıklamalar “//” (tek satır açıklama) ve “/* */ ” (birden fazla satır açıklama) ile yazılır.
- Büyük-küçük harf ayrımı vardır (karakter duyarlıdır)

Değişken/Sabit Bildirimi

```
veri_tipi değişken adı ;  
veri_tipi değişken1 , değişken2 ;  
veri_tipi değişken_adı = değeri ;  
const veri_tipi değişken_adı = değeri;
```


Veri Tipleri

Tablo-2. Arduino veri tipleri

Veri tipi	Özellik	Min	Max	Bellek Alanı (Byte)
boolean	Mantıksal veri (true/false)			
char	Tek karakter			1
unsigned char	byte ile aynı	0	255	1
byte	0-255 arası işaretsiz tamsayı	0	255	1
int	Tamsayı	-2^{15}	$2^{15}-1$	2
unsigned int	İşaretsiz tamsayı	0	$2^{16}-1$	2
word	İşaretsiz tamsayı	0	$2^{16}-1$	2
long	Uzun tamsayı	-2^{31}	$2^{31}-1$	4
unsigned long	İşaretsiz uzun tamsayı	0	$2^{32}-1$	4
short	Kısa tamsayı	-2^{15}	$2^{15}-1$	2
float	Ondalıklı sayı	$-3,4028235 \cdot 10^{38}$	$3,4028235 \cdot 10^{38}$	4
double	Ondalıklı sayı			4

Sabitler

Tablo-3. Arduino'daki bazı sabitler

Sabit	Değerleri
Mantıksal	true , false
Pin seviyesi	HIGH , LOW
Pin veri yönü	INPUT , OUTPUT

Veri Dönüşümleri

Tablo-4. Veri tipi dönüşümleri

Sabit	Değerleri
<code>char()</code>	<i>Karaktere</i>
<code>byte()</code>	<i>Tamsayıya</i>
<code>int()</code>	<i>Tamsayıya</i>
<code>word()</code>	<i>Tamsayıya</i>
<code>long()</code>	<i>Tamsayıya</i>
<code>float()</code>	<i>Ondalıklı sayıya</i>

Operatörler

Tablo-5. Aritmetik operatörler

Operatör	İşlem
+	<i>Toplama</i>
-	<i>Çıkarma</i>
*	<i>Çarpma</i>
/	<i>Bölme</i>
%	<i>Mod alma</i>

Operatörler

Tablo-6. Karşılaştırma operatörleri

Operatör	İşlem
==	Eşit
!=	Eşit değil
>	Büyük
<	Küçük
>=	Büyük eşit
<=	Küçük eşit

Operatörler

Tablo-7. Bitsel operatörler

Operatör	İşlem
&	VE
	VEYA
~	DEĞİL
^	Özel VEYA
>>	Bir bit sağa kaydır
<<	Bir bit sola kaydır

Operatörler

Tablo-8. Birleşik operatörler

Operatör	İşlem
++	<i>Bir arttır</i>
--	<i>Bir azalt</i>
+=	<i>Topla-aktar</i>
-=	<i>Çıkar-aktar</i>
*=	<i>Çarp-aktar</i>
/=	<i>Böl-aktar</i>
%=	<i>Mod-aktar</i>
&=	<i>VE-aktar</i>
=	<i>VEYA-aktar</i>

Operatörler

Tablo-9. İşaretçi operatörler

Operatör	İşlem
*	<i>Adresteki değer (de-referans)</i>
&	<i>Referans</i>

Karar Yapıları

- if

- if-else

- switch-case

Döngüler

- for
- while
- do-while
- break - continue - return - goto

Sayısal Giriş/Çıkış Fonksiyonları

Tablo-10. Sayısal giriş/çıkış fonksiyonları

Fonksiyon	Kullanım şekli	Açıklama
pinMode	<code>pinMode(pin no, mod)</code> mod: INPUT, OUTPUT, INPUT_PULLUP	İlgili pin numarasını giriş-çıkış için ayarlar.
digitalWrite	<code>digitalWrite(pin no, değer)</code> değer: HIGH, LOW	İlgili pin numarasına değeri yazar/gönderir.
digitalRead	<code>değer = digitalRead(pin no)</code>	İlgili pin numarasındaki değeri (HIGH, LOW) okur/alır.

Analog Giriş/Çıkış Fonksiyonları

Tablo-11. Analog giriş/çıkış fonksiyonları

Fonksiyon	Kullanım şekli	Açıklama
analogReference	<code>analogReference (tip)</code> tip: DEFAULT (5 V) EXTERNAL (Aref girişine 0-5 V)	Analog giriş için kullanılan referans gerilimini ayarlar.
analogWrite	<code>analogWrite (pin no, değer)</code> değer: duty cycle (0-255)	İlgili pin numarasına değeri (PWM dalga) yazar/gönderir.
analogRead	<code>değer = analogRead (pin no)</code>	İlgili pin numarasındaki değeri okur/alır. 16 kanal, 10 bit ADC [0 V -> 0, 5 V -> 1023]

Bazı Sayısal Fonksiyonlar

- $\min(a,b)$
- $\max(a,b)$
- $\text{abs}(a)$
- $\text{pow}(a,b)$
- $\text{sqrt}(a)$
- $\sin(a)$
- $\cos(a)$
- $\tan(a)$

Zaman Fonksiyonları

<code>delay</code>	<code>delay(süre)</code>	ms cinsinden program akışını bekletir
<code>delayMicroseconds</code>	<code>delayMicroseconds(süre)</code>	Mikrosaniye cinsinden program akışını bekletir