

1. GİRİŞ

Sıcaklık, nem, basınç gibi değişkenler yürüyüş, koşu, dağcılık, bisiklet sporları gibi açık hava sporları yapan sporcular için çok önemlidir. Bu nedenle sayısal veya analog ölçümler yapılarak bu verilerin gerektiği gibi belirli arayüzler vasıtasıyla sporculara gösterilmesi fayda sağlar. Bu projede belirtilen ihtiyaçları karşılayabilecek ve tüm fonksiyonları bir arada bulunan elektronik metre tasarlanmıştır.

1.1 PROJENİN AMACI

Bu projenin amacı; acık havada yapılan spor, yürüyüş ve benzeri aktivitelerde kullanılmak maksadıyla sıcaklık, nem basınç, zaman gibi parametreleri tek bir sayısal ekranda spor yapan şahsa gösteren bir elektronik metre tasarlamaktır.

1.2 PROJENİN KAPSAMI

Bitirme projesi yedi bölüm, sonuç, kaynaklar, 20 adet şekiller dizini, 11 adet çizelgeler dizini ve deneysel çalışma için yazılan program ekleri olmak üzere toplam 31 sayfadan ibarettir.

Projenin giriş bölümünde konunun hayatımızla olan ilişkisi gösterilip, projenin amacı ve bu amaç doğrultusunda yapılması gereken çalışmalar belirtilmiştir.

Projenin ikinci bölümünde Arduino kartlara giriş yapılmıştır. Arduino kart günümüzde elektronik tabanlı bütün projelerde sıkça kullanılmaktadır. Arduino kartın tercih sebepleri açıklanmıştır. Arduino kartın kullanımı için gerekli aşamalar açıklanmış, Arduino kartın donanımsal incelemesi yapılmıştır. Ayrıca bu bölümde Arduino kartın iç yapısı incelenmiş içeriğindeki hafıza birimleri ve üzerinde bulunan giriş ve çıkış pinleri tanıtılmış, bilgisayarda hazırlanan programın Arduino karta nasıl yükleneceği gösterilmiştir.

Projenin üçüncü bölümünde sıcaklık değişkeninin ölçülmesi için sıcaklıkla ilgili sensörler incelenmiştir. Bu bölümde DHT22 sıcaklık ve nem sensörü hakkında genel bilgiler verilmiş, sensörün üzerinde bulunan pinler tanıtılmış ve bu pinlerin Arduino kart ile olan bağlantıları açıklanmıştır.

Projenin dördüncü bölümünde basınç değişkeninin ölçülmesi için basınçla ilgili sensörler incelenmiştir.

Projenin beşinci bölümünde cihazda gerçek zaman ve tarihin gösterilmesi için gereken bütünleşmiş birimler incelenmiştir.

Projenin altıncı bölümünde bu bölüme kadar yapılan çalışmaları desteklemek amacıyla Arduino UNO kartı ile 20x4'lük LCD'de bağlantıları yapılarak ara yüzü tasarlanmıştır.

Projenin kaynaklar bölümünde incelenen makaleler ve projenin hazırlanmasında başvuru kitap ve dokümanlar referans olarak yer almaktadır.

1.3 PROJE TASARIMI

1.3.1 Proje Tasarımında Kullanılacak Temel Birimler

Çok amaçlı metre tasarımında, kolay bir şekilde çevresiyle etkileşime girebilen sistemlerin tasarımında kullanılan Arduino kart kullanılmaktadır. Arduino kart açık kaynaklı bir geliştirme platformudur ve Arduino kütüphaneleri ile mikrodenetleyicileri kolaylıkla programlanabilir. Ayrıca Arduino kartta bulunan analog ve dijital giriş ve çıkışlar ile sensörler den gelen veriler kolay bir şekilde alınır ve gerekli arayüzler vasıtasıyla kullanıcıya yansıtılabilir.

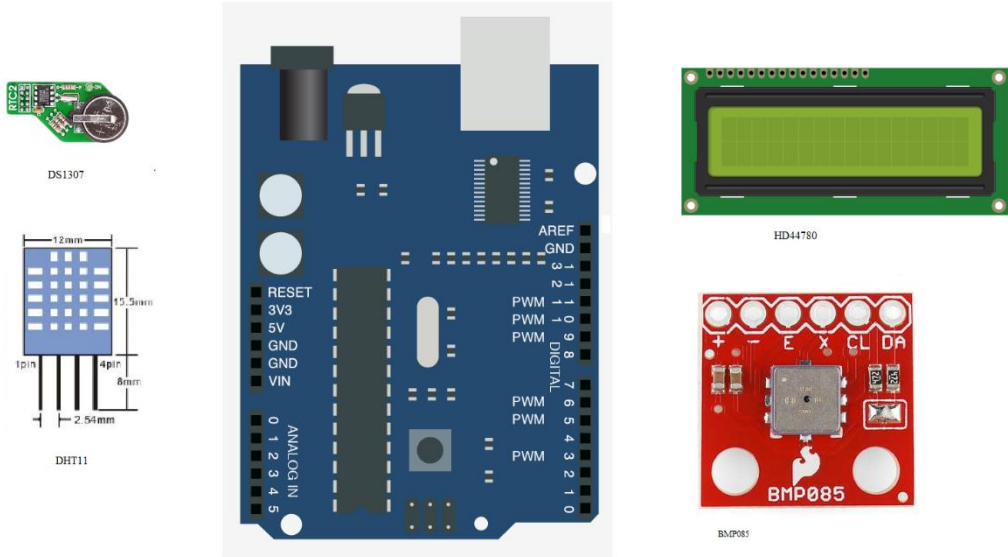
Sıcaklık ve nem ölçümü ayrı ayrı yapılacağı gibi iki alanda birden ölçüm yapıp veri alabilecek dijital sıcaklık ve nem sensörü olan DHT11 veya DHT22 sensörü kullanılmıştır.

Basınç ölçümünde BMP085 analog basınç sensörü kullanılacaktır.

Zaman gösterimi için DS1307 gerçek zamanlı tarih ve saat gösteriminde kullanılan bütünleşmiş devreden faydalanılmıştır. Ayrıca bu entegrenin kullanılabilmesi için tarih ve zaman bilgilerinin cihazın içinde veri olarak tutulması gerekmektedir. Bunun için CR2032 pili kullanılmıştır.

Kullanıcıyla iletişimi sağlamak için 16x2'lik boyutlarda gösterim yapan LCD ekran ve ölçülüp ekranda gösterilen değerler arasında geçişi yapmak için bir buton kullanılmıştır.

Kullanılan donanımlar Şekil 1.1'de gösterilmiştir.



Şekil 1.1 Projede Kullanılacak Temel Birimler

1.4 PROJE MALİYETİ

Projede Arduino platform, DHT22 sıcaklık nem sensörü, BMP085 basınç sensörü ve DS1307 zaman entegresi kullanılmış ve 136 lira harcanmıştır. Kullanılan donanımların maliyeti Çizelge 1.1’de verilmiştir.

Sıra No	Malzeme	Adet	Maliyet (TL)
1	DS1307 (zaman entegresi)	1	11,80
2	0.1 uF Kondansatör	1	0,83
3	10k Direnç	3	1,02
4	CR2032 pil yuvası	1	2,18
5	CR2032 pil	1	2,50
6	32.768 KHz kristal	1	2,36
7	DHT22 (sıcaklık ve nem sensörü)	1	31,86
9	BMP085 (basınç sensörü)	1	38,25
8	10k Potansiyometre	2	2,18
10	16 Pin Erkek Header	1	8,85
11	BMP085 Basınç Sensörü	1	0,00
12	Arduino UNO R3	1	57,82
13	Kablo	x	1,00
14	20x4 LCD Ekran Mavi	1	19,50
15	Döner Enkoder	1	8,50
Toplam Maliyet			215,150

Çizelge 1.1 Proje Maliyet Tablosu

1.5 PROJE UYGULAMA TAKVİMİ

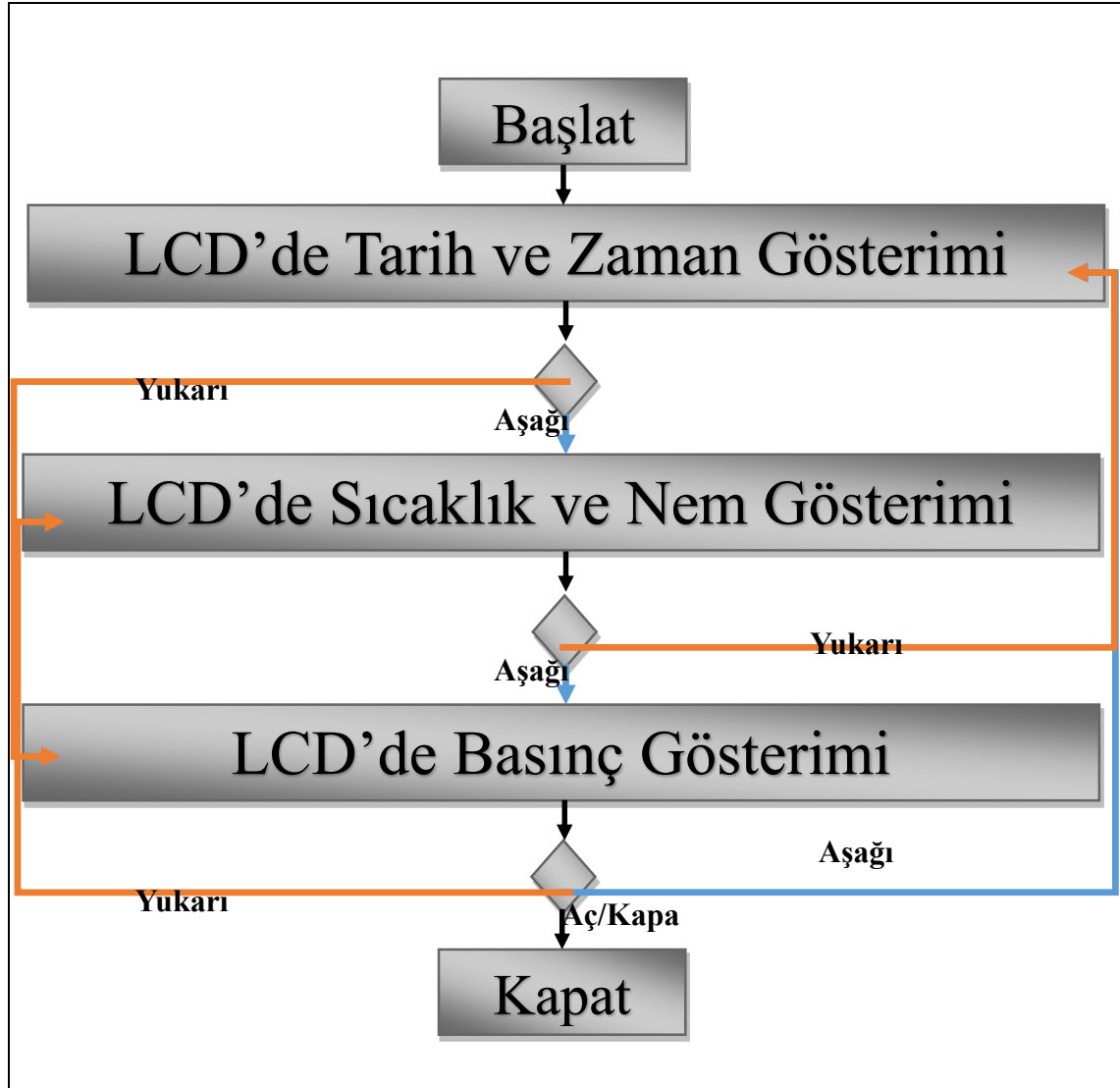
Projenin alınmasından, sunuma kadar yapılanlar Çizelge 1.2’de gösterilmiştir.

AY	ŞUBAT	MART	NİSAN
İş	15 16 17 18 19 20 21 22 23 24 25 26 27 28 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Bitirme Projesi Konu Seçimi			
Tasarımda Kullanılacak Birimlerin Seçimi			
Entegre ve Kartların Tedariği			
Proje Raporunun Hazırlanması			
Yazılımın Geliştirilmesi			
Devrenin Kurulması ve Yazılımın Yükleme			
Vize Sunumu			
Final Sunumu			
AY	NİSAN	MAYIS	HAZİRAN
İş	17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Bitirme Projesi Konu Seçimi			
Tasarımda Kullanılacak Birimlerin Seçimi			
Entegre ve Kartların Tedariği			
Proje Raporunun Hazırlanması			
Yazılımın Geliştirilmesi			
Devrenin Kurulması ve Yazılımın Yükleme			
Uygun Kullanıcı Ara Yüzünün Hazırlanması			
Vize Sunumu			
Final Sunumu			

Çizelge 1.2 Proje Uygulama Takvimi

1.6 PROJE ŞEMASI

Çok amaçlı elektronik metre projesinde, hazırlanan cihazın üzerinde bulunan “BAŞLAT”, “AŞAĞI” ve “YUKARI” düğmeleri kullanılacaktır. Başlat düğmesine basıldığı andan itibaren sensörler çalışmaya başlayacak, sıcaklık nem ve basınç değerleri ölçülecektir. Kullanıcı aşağı ve yukarı düğmelerine bastıkça sıcaklık, nem ve basınç parametreleri arasında geçiş yapacaktır. Projenin akış şeması Şekil 1.2’de gösterilmektedir.



Şekil 1.2 Proje Akış Şeması

2. ARDUINO

Arduino; kolay kullanımlı, esnek, donanım ve yazılım tabanlı, açık kaynak kodlu elektronik prototip platformudur. Arduino sanatçılar, tasarımcılar, hobi ve etkileşimli nesneleri veya ortamları oluşturmakla ilgilenen kişiler için tasarlanmıştır.

Arduino sensörler ile çeşitli girişler olarak çevresini hissedebilir veya kontrol ışıkları, motorlar ve diğer aktüatörler ile çevresini etkileyebilir.

Arduino üzerinde bulunan mikrodenetleyicinin programlanmasında Arduino Programing Language kullanılmıştır. Bu program Wiring tabanlıdır. Wiring mikrodenetleyiciler için bir açık kaynak programlama çerçevesidir. Arduino projeleri tek başına işlevsel oldukları gibi çalışan bir bilgisayarda bulunan yazılımlar ile de kullanılabilir (Flash, Processing, MaxMSP).

Arduino 2005 yılında bir proje olarak Massimo Banzi ve David Cuartielles tarafından, Arduino of Ivrea adı altında geliştirilmiştir.

2.1 ARDUINO ÖZELLİKLERİ

Arduino platformlarında genel olarak Atmel firmasının çeşitli mikrodenetleyicileri kullanılmaktadır. Arduino UNO, Mini gibi modellerinde Atmega328p kullanılmaktadır.

Sketch – Arduino platformlarının program için kullandığı addır. Windows, Mac, Linux işletim sistemlerinde yazılan program gerekli arabirimler vasıtasıyla Arduino platforma yüklenir ve koşturulması sağlanır.

Dijital Pinler – Bütün Arduino platformlarında dijital giriş ve çıkış pinleri bulunmaktadır.

Analog Giriş Pinleri – Bütün Arduino platformlarında analog giriş pinleri bulunmaktadır. Analog olarak alınan veriler platformun içinde tümleşik devre halinde bulunan ADC'ler (analog to digital converter – analogtan dijital çeviriciler) sayesinde dijital verilere çevirilir.

PWM (pulse width modulation) – Arduino platformları, dalga boyu modülasyonu için gerekli olan PWM girişlerini içermektedir. Bu girişler dalga boyu ölçümü, istenilen genlikte dalga üretimi, saat sinyali üretimi gibi yerlerde kullanılabilir.

Bellek – Arduino platformları, EEPROM, SRAM, Flash bellek gibi gerekli bellek birimlerini içermektedir.

Programlama Teknikleri – Değişken tanımlama, fonksiyonlar ve kütüphaneler ile çalışılmaktadır.

2.1.1 Arduino Temel Kavramları

Arduino platformunu oluşturan bileşenlere göz atacak olursak.

- Arduino Geliştirme Ortamı (IDE)
- Arduino Bootloader (Optiboot) – Bir mikrodnetleyiciyi programlamak için kullanılan cihaza yönelik geliştirilmiş olan programlayıcıları kullanmak gerekmektedir. Bu işlemin bir diğer yolu da seri haberleşme üzerinden mikrodnetleyiciyi seri haberleşme modülü üzerinden programlamamızdır. Bunun için mikrodnetleyicinin kendi program belleğinin programlama özelliğine sahip olması gerekmektedir. Arduino da bu programlama işlemini yapacak OptiBoot bootloader'ıdır. Mikrodnetleyici çalışmaya başladıktan hemen sonra belleğin sıfıncı adresinden itibaren programlamaya başlar. Eğer bu veriler gelmediyse mikrodnetleyiciye yüklenmiş olan programa verilen isim OptiBoot'dur.
- Arduino Kütüphaneleri – Arduino platformunun bu kadar çok kullanılmasındaki en büyük nedenlerden bir tanesi de sahip olduğu geniş kütüphane haznesidir. Bu kütüphaneler sayesinde mikrodnetleyiciyi çok iyi bilmesek de kolay bir şekilde programlayabiliriz.

Standart Kütüphaneler;

- i. EEPROM – kalıcı depolamada okuma ve yazma işlemlerini üstlenen kütüphane birimidir.
- ii. Ethernet – Arduino Ethernet Shield kullanarak internete bağlanmada kullanılır.
- iii. Firmata – Standart seri protokolleri kullanarak çalışan bilgisayarlar ile iletişimi sağlayan kütüphaneleri içerir.
- iv. GSM – GSM Shield kullanarak GSM/GRPS networküyle iletişimi sağlayan kütüphanelerdir.
- v. LiquidCrystal – Çok amaçlı metre tasarımı projesinde de kullanılacağı gibi LCD'lerin kontrolünü sağlayan kütüphaneler.
- vi. SD – SD kartlara okuma yazma işlemlerini kolaylaştıran kütüphaneler.
- vii. Servo – Servo motorların kontrolü
- viii. SPI (Serial Peripheral Interface) – Seri haberleşmede dört hat üzerinde SPI kullanan aletlerin kontrolünü kolaylaştıran kütüphaneler.
- ix. SoftwareSerial – Seri haberleşmede dijital pinleri kullanan aletler.
- x. Wifi – Arduino WiFi Shild kullanara networke bağlanmada.
- xi. Wire – TWI/I²C bağlantılarında kullanılan kütüphaneler.

USB Kütüphaneleri;

- i. Klavye
- ii. Fare (mouse)

Bunlar gibi birçok kütüphane seçeneği mevcuttur.

- AVR Dude – Arduino üzerindeki mikrodnetleyici programlayan yazılım
- AVR-GCC – Derleyici C ve C++ için.

Bir Arduino platformu bütün bu bileşenleri bir araya getirip kolayca kullanımı sağlamaktadır.

Arduino'nun günümüzde bu kadar sık kullanılmasının temel nedenlerini sıralayacak olursak;

- Açık kaynak kodlu donanım/yazılım – Devre şemasından programlama ara yüzüne kadar her birimin açık kaynak kodlu olmasıdır. Yani Arduino üzerinde bulunan birimleri (mikrodenetleyici, bellekler, USB ve güç bağlantı noktaları gibi) elinizle birleştirdikten sonra Arduino ile ilgili sitelerden bedava olarak indirilebilen yazılımlar ile Arduino kart tasarlanabilir.
- Ucuz – Piyasada bulunan diğer mikrodenetleyici platformlarına oranla nispeten daha ucuzdur. Piyasada bulunabilecek en pahalı Arduino yaklaşık olarak 110 TL civarındadır.
- Cross – Platform (çapraz platform) – Arduino yazılımı Windows, Macintosh OSX ve Linux gibi işletim sistemlerinde sorunsuz bir şekilde çalışmaktadır.
- Net, basit programlama ortamı – Arduino programlama ortamı yeni başlayanlar için basit ve kullanışlıdır. Ayrıca ileri seviye kullanıcılar için yeteri seviyede esnektir.

2.1.2 Arduino Çeşitleri

Arduino Platformları

- Arduino Uno – Çok amaçlı metre projesinin yapımında kullanılan Arduino platformu.
- Arduino Leonardo
- Arduino Due – Diğer Arduino platformlarına karşın bu kart 3,3V'da çalışmaktadır. Ayrıca üzerinde ekstrasından bir tane programlama girişi bulunmaktadır.
- Arduino Esplora – Mikrofon, LCD girişleri, LDR ışık sensörü, potansiyometre, RGB LED, Sıcaklık sensörü, Analog joystick girişi, mikro USB, buzzer, dört tane düğme, bulunduran programlaması en basit Arduino platformudur.
- Arduino Mega 2560 – Üzerinde en fazla dijital giriş ve çıkış pini bulunan (54) Arduino platformudur. Ayrıca 14 tane de analog giriş olmak üzere toplam 60 pin bulunmaktadır.
- Arduino ADK – Arduino Mega 2560 platformu baz alınarak geliştirilmiştir. Özellikleri benzerdir.
- Arduino Ethernet – Dijital pinlerinden 10,11,12,13 girişleri Ethernet modülü için ayrılmıştır ve başka amaçlar için kullanılamaz. Platformun çalıştırılmasında ekstra güç Ethernet modülünün çalıştırılması için eklenebilir. Wiznet Ethernet arayüzü sayesinde USB girişine ihtiyaç duymamaktadır.
- Arduino BT (bluetooth) – Platform üzerinde bulunan Bluegiga WT₁₁ modülü sayesinde bluetooth ile iletişim sağlayabilir.
- Arduino LilyPad – Takılabilir aksesuarlar ve e-kumaşlar için uygun bir mikrodenetleyici kartıdır. Bu platform çeşidi kumaşa dikilir ve benzer iletken iplik ile güç kaynakları, sensörler ve aktüatörler monte edilebilir.

- LilyPad Arduino USB – Arduino LilyPad platformunu USB modülüne sahip versiyonudur.
- LilyPad Arduino Simple
- LilyPad Arduino SimpleSnap
- Arduino Micro – En küçük Arduino platformudur. Küçük boyutu sayesinde breadboardlar üzerine kolaylıkla yerleştirilebilir. Ayrıca üzerinde bulunan mikro USB girişi sayesinde diğer Arduino platformlar gibi kolaylıkla programlanabilir
- Arduino Nano – Üzerinde mini USB girişi bulunmaktadır.
- Arduino Pro Mini
- Arduino Pro
- Arduino Fio
- Arduino Mini

Arduino Shields

- GSM, Ethernet, WiFi, Wireless SD, Motor, Wireless Proto, Proto

Arduino Kits

- Arduino Starter Kit

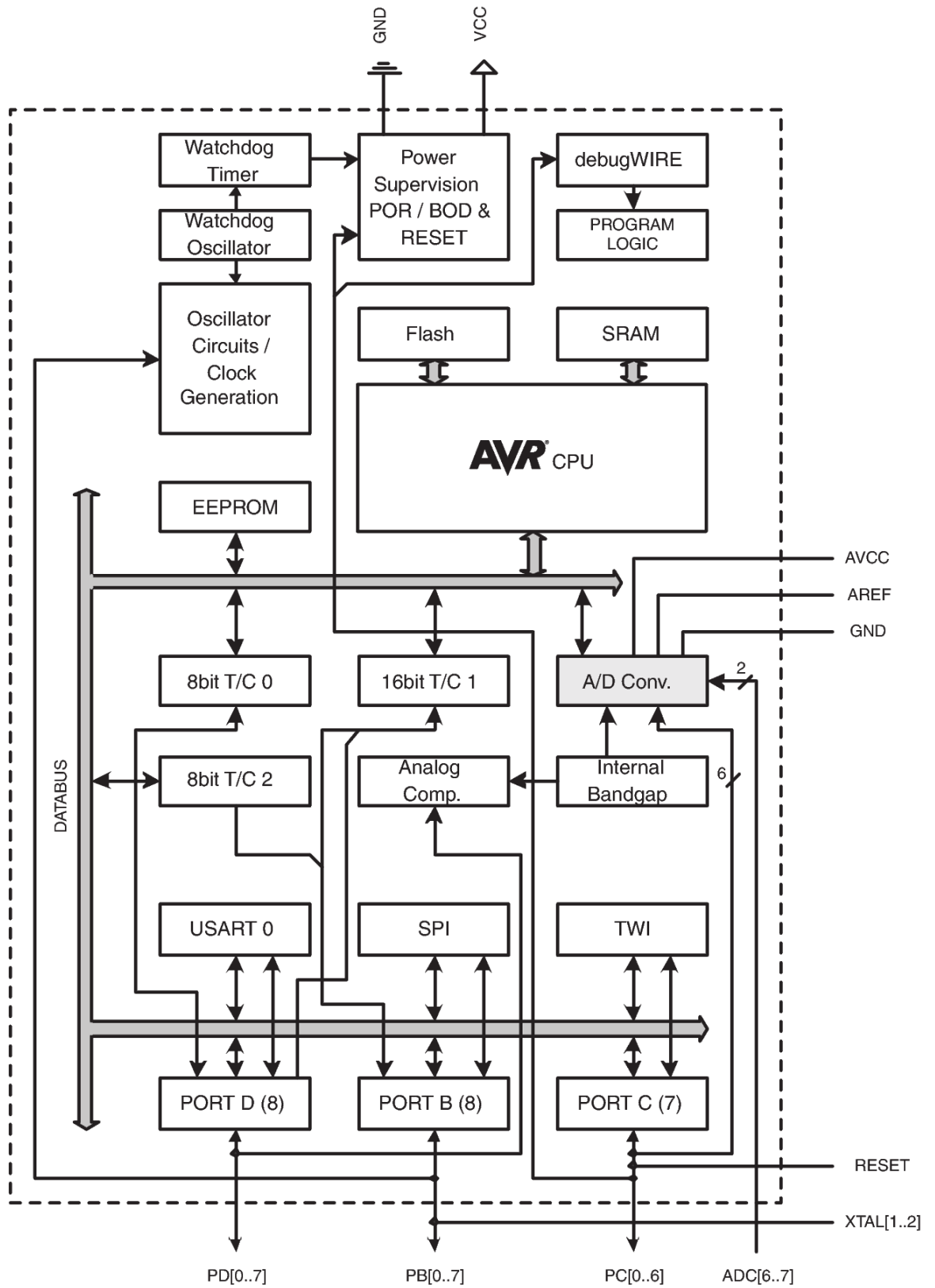
Arduino Accessories

- USB/Serial Light Adapter, Mini USB/Serial Adapter

Arduino Mega, Uno, Mini, Nano ve LilyPad platformlarının gerçek boyutları Şekil 2.1’de gösterilmektedir.



Şekil 2.1 Arduino Mega – Uno – Nano – Mini – LilyPad



Şekil 2.3

ATmega328w Blok Diyagramı

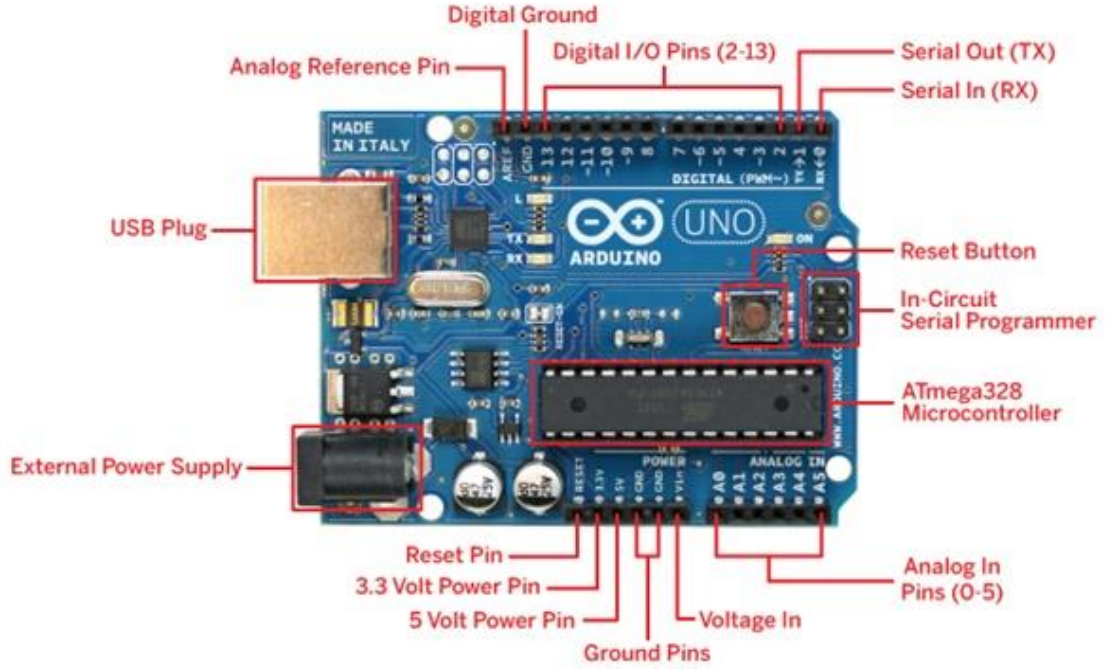
2.2.1 Arduino UNO Özellikleri

Arduino UNO platformu, Atmel firmasının ATmega328 mikrodenetleyicisini kullanmaktadır. Arduino'nun çalışma voltajı 5 V tavsiye edilen çalışma voltaj aralığı 7-12 volt arasındadır. Ayrıca Arduino üzerinde programlamada kullanılmak üzere SRAM, EEPROM ve Flash bellek bulunmaktadır. Arduino UNO'nun ayrıntılı bilgileri Çizelge 2.1'de verilmiştir.

Mikrodenetkeyici	ATmega328
Çalışma voltajı	5 V
Giriş Voltajı (Önerilen)	7-12 V
Giriş Voltajı (Limitleri)	5-20 V
Dijital Giriş/Çıkış Pinleri	14 (6 tanesi PWM desteklidir)
Analog Giriş Pinleri	6
Flash Bellek	32 KB (0,5 KB bootloader kullanmakta)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

Çizelge 2.1 Arduino UNO Özellikleri

Arduino'nun üzerinde 14 tane sayısal, 6 tane analog ve 6 tane güçle ilgili pin olmak üzere toplam 26 pin bulunmaktadır. Sayısal 14 pinden belirli 6 tanesi PWM'de kullanılmaktadır. Ayrıca kartın interaktifliği için USB bağlantısı, güç tedariki için girişler bulunmaktadır. Arduino platformun üzerinde bulunan bütün pin ve birimlerin tanıtımı Şekil 2.4'te yapılmıştır.



Şekil 2.4 Arduino UNO Pinler ve Birimler

- Güç

Arduino UNO USB bağlantısından veya güç bağlantı noktasından harici güç kaynağı ile beslenebilmektedir. Takılan güç kaynağı Arduino tarafından otomatik olarak seçilir.

Harici güç kaynağı olarak AC-to-DC adaptör veya batarya kullanılabilir. Eğer batarya kullanılacaksa GND ve V_{in} pinlerinin kafaları güç sağlayıcısına ve toprağa yerleştirilmelidir.

Platform 6 ila 20 volt arasında çalışabilmektedir. Bu değerlerin dışında çalışan devreler kararsız sonuçlar çıkartacaktır ve karta zarar verebilecektir.

Arduino UNO üzerinde güçle ilgili beş pin bulunmaktadır. Bunlar V_{in} giriş voltajı için, 5v pini karttan 5v sağlamak için, 3v3 pini 3,3 volt kaynak sağlamak için, GND toprak bağlantısı için kullanılmaktadır. Ayrıca IOREF pini bulunmaktadır.

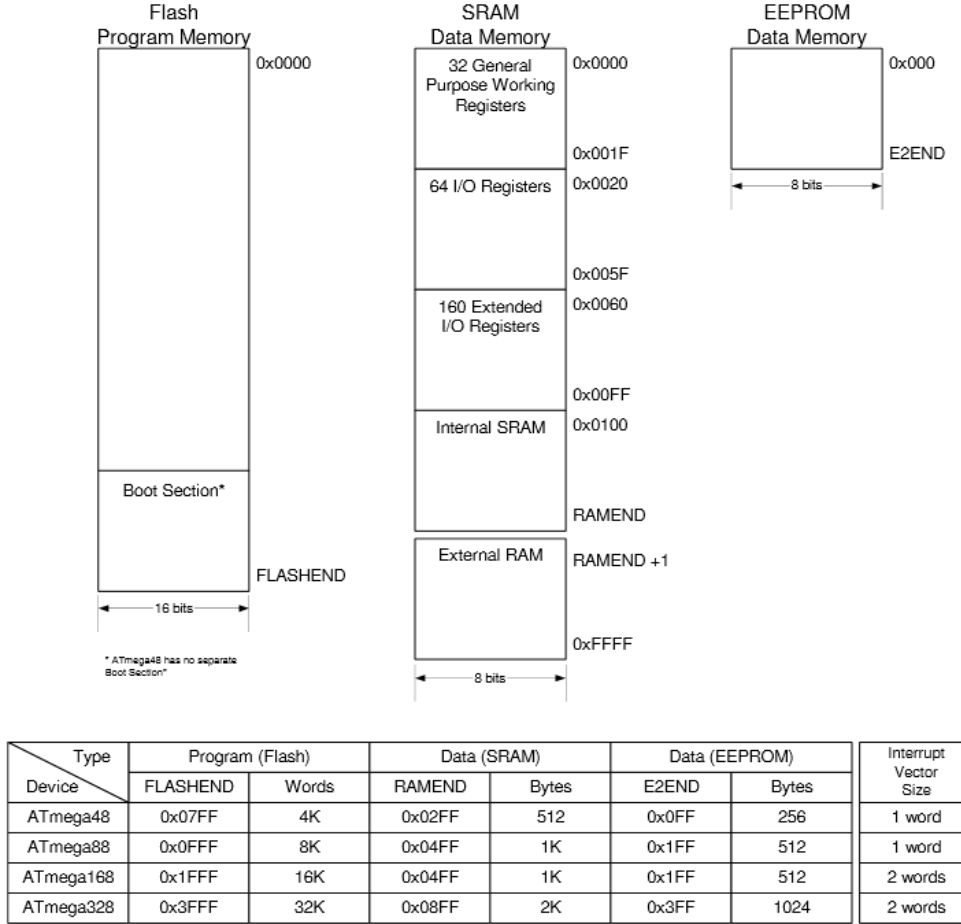
- Bellek

ATmega328, 0,5 KB'ı bootloader tarafından kullanılmak üzere 32 KB hafızaya sahiptir. Ayrıca 2 KB SRAM, 1KB EEPROM bulunmaktadır.

SRAM (Static Random-Access Memory) – Her bir biti saklamak için iki durumlu kilitlemeli devre kullanan yarı iletken bellek tipidir. SRAM'ler uçucu (volatile) belleklerdir. Devreye uygulanan güç kesildiği anda içeriğini kaybeder. Gerçek zamanlı saat ve tarih gösteriminin gerçekleştirilmesi için devrede CR2032 pil ve CR2032 pil yuvası kullanılmıştır.

ATmega328w mikrodnetleyicisinin sahip olduđu bellek organizasyonu Şekil 2.5'te gösterilmektedir.

ATmega Memory Map
ATmega48/88/168/328 etc.



Şekil 2.5 ATmega328w Bellek Organizasyonu

- Giriş ve Çıkış

Arduino UNO'da bulunan 14 dijital pinden her biri giriş veya çıkış olarak kullanılabilir. Bu işlem *pinMode()*, *digitalWrite()* ve *digitalRead()* fonksiyonlarını kullanarak gerçekleştirilir. Her pinden maksimum 40mA alınabilir veya her pine maksimum 40mA sağlanabilir. Analog giriş olmak üzere toplam 6 tane pin bulunmaktadır. Ayrıca özel durumlara sahip pinler de bulunmaktadır. Bu pinler;

- Serial 0 (RX) ve 1 (TX) : Alma ve gönderme (Receive/Transmit) işlemlerinde kullanılırlar
- Harici kesme (interrupt) pinleri, dijital 2 ve 3. Birden fazla Arduino platform, mikrodnetleyici veya dışardan interrupt alınması gereken durumlarda kullanılacak pinler.
- PWM (pulse width modulation) 3,5,6,9,10,11; *analogWrite()* komutuyla 8-bit PWM çıkışı sağlar.

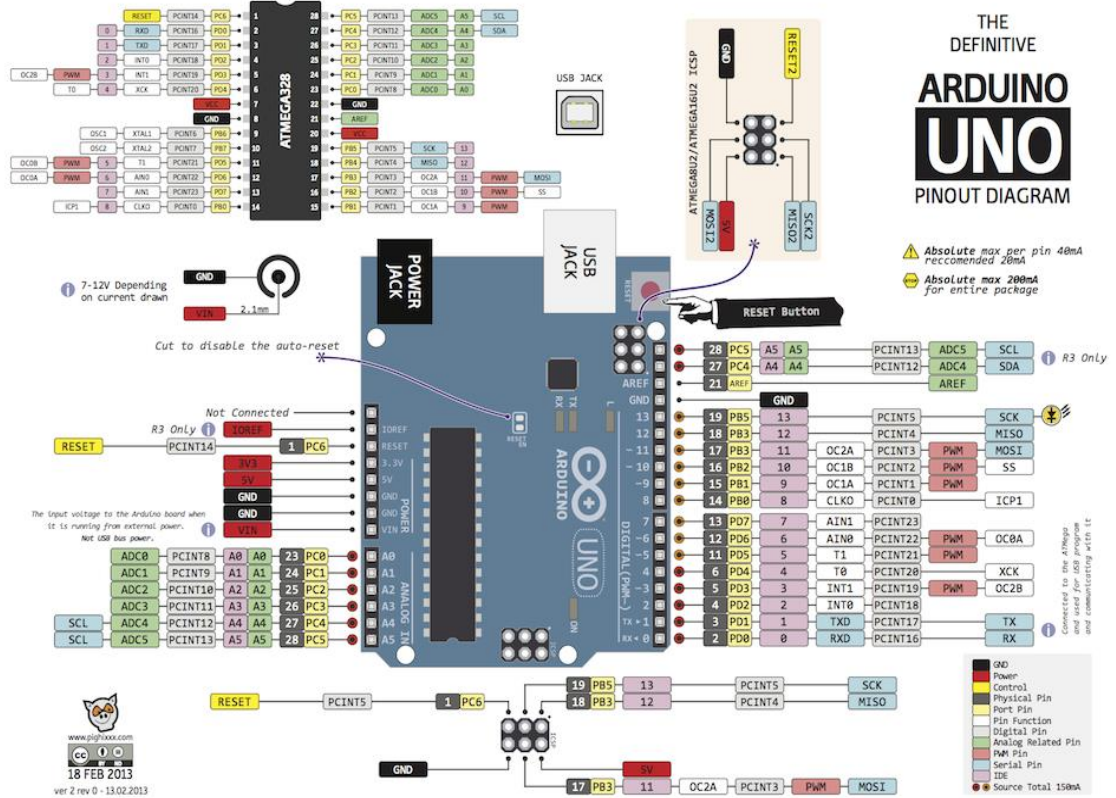
- iv. SPI 10(SS) cihaz seçimi, 11(MOSI) data girişi, 12 (MISO) data çıkışı ve 13 (SCK) saat (clock) sinyali iletiminde kullanılır.
- v. LED 13 numaralı pin üzerinden Arduino UNO üzerinde bulunan LED kontrol edilir.
- vi. TWI (I²C) Analog 4 (SDA) data iletimi ve Analog 5 (SCL) saat (clock) sinyalinin iletimi için bu iki pin kullanılmaktadır I²C.
- vii. AREF Analog inputlar için referans voltajının verildiği pindir.
- viii. Reset LOW a çekildiğinde mikrodenetleyiciyi baştan başlatan pindir.

• İletişim

Arduino UNO platformunun bilgisayarla, diğer bir Arduino platform ile veya başka bir mikrodenetleyiciyle iletişimi için birçok yöntem bulunmaktadır. ATmega328 UART TTL sayesinde dijital pin 0 ve 1 aracılığıyla seri iletişim sağlamaktadır. Ayrıca Atmega16U2 sayesinde USB üzerinden bilgisayarla seri iletişim sağlamaktadır.

Ayrıca ATmega328 I²C (TWI) ve SPI seri iletişim yöntemlerini desteklemektedir [bknz. Giriş ve Çıkış].

Arduino UNO'nun üzerindeki bütün bağlantılar Şekil 2.6'da gösterilmiştir.



Şekil 2.6

Arduino UNO Pin Diyagramı

- Programlama

Arduino UNO, Arduino yazılımı tarafından programlanabilmektedir. Bu program ayrıca diğer bütün Arduino platformlarının desteklemektedir.

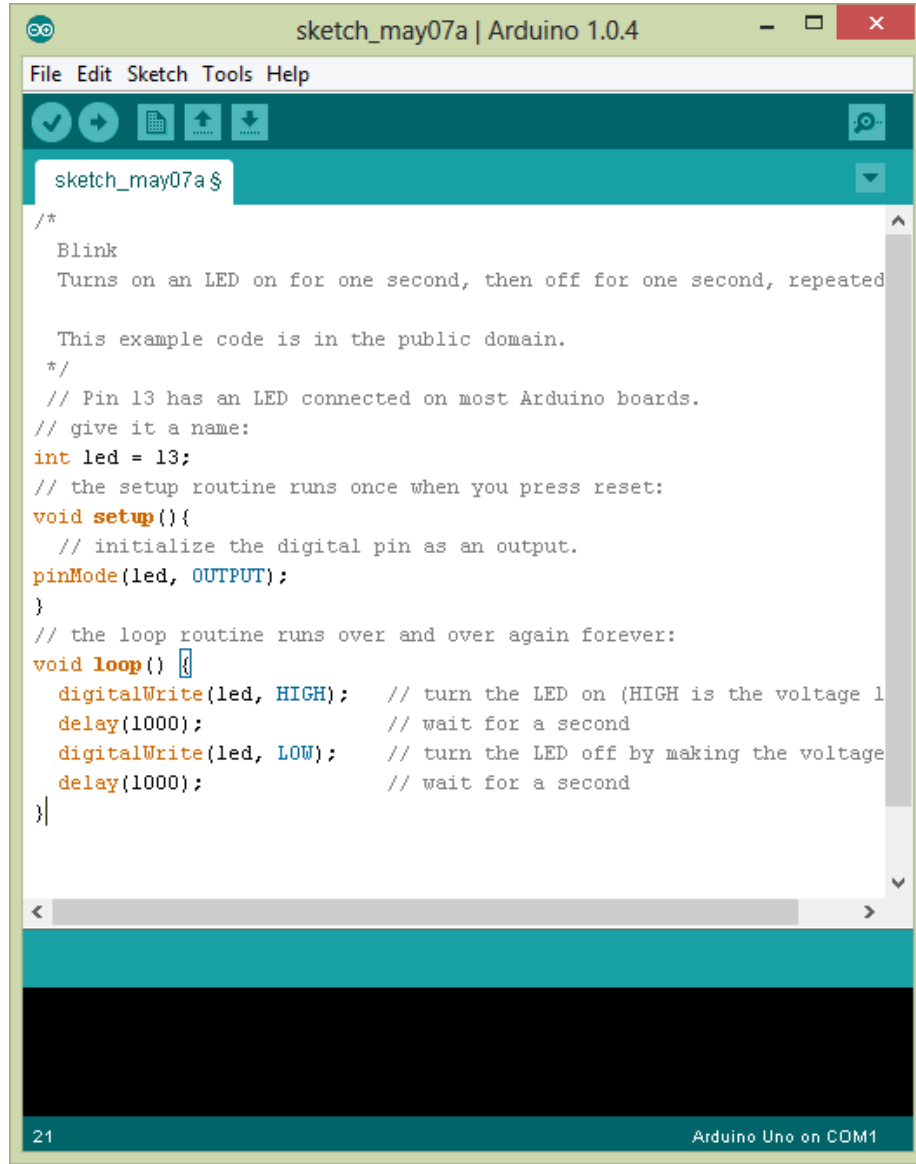
2.3 ARDUİNO PROGRAMLAMA DİLİ

Arduino platformlarını açık kaynak kodlu olması kod yazımını kolaylaştırır ve yazılan programın i/o kartlarına yüklenmesini kolaylaştırmaktadır. Arduino yazılım dili Windows, Mac OS X ve Linux işletim sistemlerinde çalışmaktadır. Projede kullanıcı dostu olması açısından Windows da yazılımlar gerçekleştirilmiştir.

Arduino; Wiring tabanlı kendine has bir programlama dili kullanılır. Mikrodenetleyicisinde bir Bootloader yüklü halde geldiği için harici bir programlayıcıya gerek yoktur. Bootloader; Arduino üzerindeki mikrodenetleyicinin EEPROM'una yazılmış, harici bir programlama kartı kullanmadan seri iletişim yoluyla yazdığımız komutları direkt olarak mikrodenetleyiciye yüklememizi sağlayan yazılımların genel adıdır.

Arduino yazılım ortamı Java'da geliştirilmiş ve Processing yazılım dili temel alınmıştır. Söz dizimi (sentaks) olarak C,C++ veya Java dillerine benzemektedir. Örneğin satır sonları noktalı virgül (;) ile biter ve (// veya /* */) komutları ile yazılan satırlara yorumlar eklenebilmektedir. Arduino yazılım dilinde bütün işlemler veri tipleri (int char vs.) üzerinden gerçekleştirilmektedir.

Arduino UNO üzerinde bulunan ve dijital 13 pinine bağlı bulunan LED'in yakılıp söndürülmesi işleminin Arduino yazılım programında kodlanmış hali Şekil 2.7'de görülmektedir.



Şekil 2.7 Arduino Yazılım Programı (IDE)

3. SICAKLIK ve NEM ÖLÇÜMLERİ

3.1 SICAKLIK ÖLÇÜMÜ

Sıcaklık; belirli bir değerler aralığına göre tanımlanmış sıcaklık veya soğukluk derecesidir. Sıcaklık aynı zamanda bir cihazın veya sistemin ısı enerjisi olarak da tanımlanabilir. Isı enerjisi doğrudan moleküler enerji ile ilişkili olduğundan ısı enerjisi yüksek ise moleküler enerji de yüksektir.

Sıcaklık, açık alanda faaliyet gösteren çalışanlar veya sporcular için en önemli işlem değişkenlerinden birisidir. Dolayısıyla ölçümde sıcaklığı daha hızlı, daha kararlı ve bir süre boyunca sürekli ölçebilen termometrelerin gelişmesi gerekmiştir.

Sıcaklık sensörleri, sıcaklıkta meydana gelen değişime karşılık direnç veya çıkış voltajı gibi fiziksel parametrelerdeki değişimi algırlar.

3.1.1 Ölçüm Metotları

Genel olarak iki farklı tip sıcaklık algılayıcı sensör çeşidi vardır. Bunlar;

- **Temaslı Termometreler**

Sıcaklığı algılanacak ortam veya nesne ile doğrudan fiziksel temas halinde bulunan sıcaklık algılayıcı sensörlerdir. Bu metotla katı, sıvı ve gaz halinde bulunan maddelerin büyük bir genlikte sıcaklıkları ölçülebilir. Temaslı termometreler kendi içinde üç kısma ayrılmaktadır.

- i. Genişleme Tipli Termometreler : Sıvı genişlemeli (civalı), Bimetalik ve gaz termometreler
 - ii. Termistorler – Projenin tamamlanmasında üzerinde bulunan termistör ile sıcaklık ölçümü yapan DHT22 cihazı kullanılmıştır.
 - iii. Direnç Termometreleri – nikel, bakır, platinyum, ungsten
 - iv. Isıl Çiftler – termoeleman, thermocouple
 - v. Sıvı Kristal Termometreler
 - vi. Birleşik Devreli Termometreler
- **Temassız Termometreler**

Isı kaynağından yayılan ısı enerjisini ölçen termometrelerdir. Bu metotla yansımının olmadığı katı ve sıvıların sıcaklıkları ölçülebilir, gazlarda uygun değildir. Temassız termometreler iki çeşittir.

- i. Optik termometreler
- ii. İnfrared (Kızıl ötesi) Termometreler

3.2 NEM ÖLÇÜMÜ

Nem, gaz veya havadaki su buharı miktarı olarak tanımlanmaktadır. Genellikle mutlak nem olarak ölçülmektedir. Mutlak nem, su buharı kütlesinin havanın veya gazın hacmine oranı olarak tanımlanmaktadır. %35 – 55 arasındaki bağıl nem oranı normal olarak kabul edilir. Alt değerden düşük nem oranları kuru, üst değerden yüksek nem oranları yaş olarak kabul edilmektedir.

3.2.1 Ölçüm Metotları

Nem ölçümünde genel olarak elektronik tip sensörler kullanılmaktadır. Elektronik tip sensörler ikiye ayrılır.

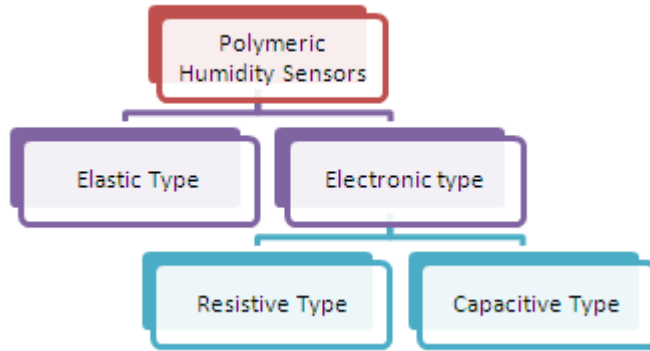
- Rezistif Tip

Rezistif nem sensörleri empedanstaki değişimi ölçerler. Bu değişim genellikle nem ile üstel bir şekilde ters orantılıdır. Empedansı değişen ortam ise iletken polimer veya tuzdur.

- Kapasitif Tip

Kapasitif RH sensörler endüstride ve ticari amaçlı olarak sıkça kullanılır. Sıcaklığın bu sensörlerin çalışmasına etkisi çok az olduğundan, oldukça büyük bir sıcaklık genliğinde sıcaklık kompanzasyonu yapılmadan kullanılabilir. Kapasitif bir RH sensörde di-elektrik sabitindeki değişime bağıl nemi ile doğru orantılıdır. Projenin gerçekleştirilmesinde kullanılan DHT22 modülünün nem birimi kapasitif tiptedir.

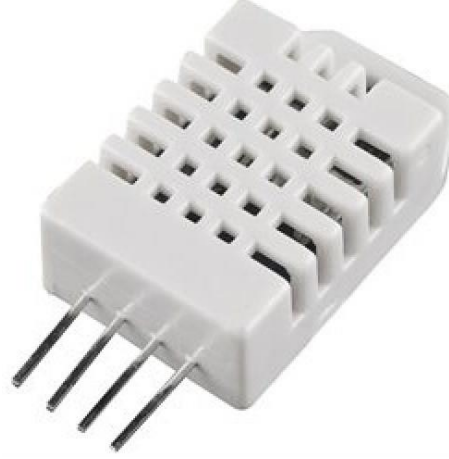
Nem ölçüm metotları Şekil – 3.1’de gösterilmiştir.



Şekil 3.1 Nem Ölçüm Tipleri

3.3 DHT22 SICAKLIK ve NEM SENSÖRÜ

Dijital-output bağıl nem ve sıcaklık sensörü/modülü aynı zamanda AM2302 olarak da adlandırılır. kapasitif-tip nem ve sıcaklık sensörü Şekil 3.2’de gösterilmiştir.



Şekil 3.2 DHT22 Sıcaklık ve Nem Modülü

DHT22, kendine özel dijital sinyal toplama tekniği ve nem farkındalık teknolojisi kullanmaktadır. DHT22’nin algılama elemanları bilgisayara 8-bitlik bir tek çip ile bağlanmaktadır.

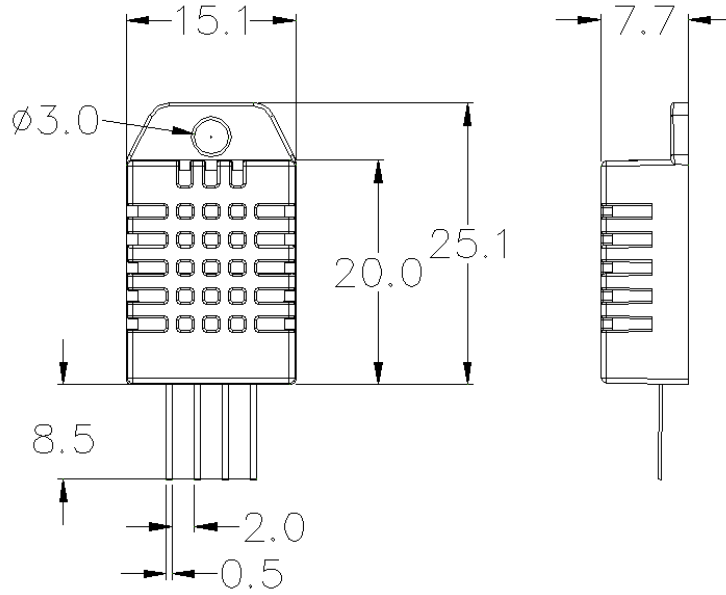
3.3.1 Sensör Özellikleri

DHT22’nin özellikleri ayrıntılı bir şekilde Çizelge 3.1’de gösterilmiştir.

Model	DHT22
Voltaj Çalışma Aralığı	3,3 – 6 V DC
Çıkış Sinyali	Tek bus yapısı üzerinden dijital sinyal
Nem Ölçüm Elemanı	Polimer Kapasitör
Nem Ölçüm Aralığı	% 0 – 100 RH
Nem Ölçüm Keskinliği	± %2 RH (Maksimum ± %5 RH)
Nem Hassasiyeti	%0,1
Nem Tekrarlanabilirliği	± %1 RH
Sıcaklık Ölçüm Aralığı	-40~80 Celsius
Sıcaklık Ölçüm Keskinliği	± 0,5 Celsius
Sıcaklık Hassasiyeti	± 0,1 Celsius
Sıcaklık Tekrarlanabilirliği	± 0,2 Celsius
Nem Ölü Bölgesi	± %0,3 RH
Uzun-dönem Kararlılığı	± %0,5 RH/yıl
Ölçüm Aralığı	2 saniye
Değiştirilebilirlik	Tamamen değiştirilebilir
Boyutları	Küçük boyut 14*18*5,5mm

Çizelge 3.1 DHT22 Özellikleri

Küçük boyutta DHT22 modülünün gerçek boyutları Şekil 3.3'te gösterilmiştir. Projede bu sensör kullanılmıştır (birimler mm cinsinden).



Şekil 3.3 Küçük Boyut DHT22 Modülü

DHT22 Pinlerinin (Şekil 3.3'te soldan sağa) işlevleri Çizelge 3.2'de gösterilmiştir.

Pin	Fonksiyon
1	V _{DD} – güç kaynağı
2	Data – sinyal
3	NULL – dikkate alınmamıştır
4	GND – Toprak

Çizelge 3.2 DHT22 Pinleri

3.3.2 İşlem Özellikleri

- Güç ve Pinler

DHT22 modülünün güç voltajı 3,3 ila 6 V DC arasında değişmelidir. Modüle güç verildiğinde, sensör kararlı duruma geçene kadar bir saniye boyunca veri gönderilmemesi gerekmektedir. Dolayısıyla yazılım aşamasında çeşitli gecikme (delay) fonksiyonlarıyla gereken bekleme süresi sağlanmıştır. Bu gecikmeyi oluşturmanın diğer bir yöntemi de modülün V_{DD} pini ile GND pini arasına konulacak 100nF'lık kapasite yeterli olacaktır.

- İletişim ve Sinyal

DHT22 sensörünün mikrodenetleyici veya platformla iletişimde tek-bus data (single-bus data) hattı kullanılmaktadır. Bu yöntemle tek seferlik iletişim 5mS sürmektedir.

Data, ondalık ve tam kısımdan oluşmaktadır. Datanın formülü aşağıdaki gibidir.

DHT22 yüksek veri bitini ilk göndermektedir.

DATA=8 bit integral RH data+8 bit decimal RH data+8 bit integral T data+8 bit decimal T data+8 bit check-sum
If the data transmission is right, check-sum should be the last 8 bit of "8 bit integral RH data+8 bit decimal RH data+8 bit integral T data+8 bit decimal T data".

Mikrodenetleyici veya platform DHT22'ye başlama sinyalini göndermesinden itibaren, DHT22 durumunu low-power-consumption-mode'dan (az güç harcayan), running-mode'a çevirir. Mikrodenetleyici başlama sinyalini göndermeyi kesmesiyle DHT22, 40-bitten oluşan nem ve sıcaklık bilgilerinden oluşan yanıt sinyalini gönderir. Başlama sinyali olmadan DHT22 modülü hiçbir türlü yanıt sinyali dönmez. Mikrodenetleyicinin başlat sinyali tekrar geldiği takdirde yeniden ölçüm yapmaya başlar. Ölçüm sıklığı DHT22 nin ölü bölgesi (hysteresis) dâhilinde yapılmaktadır.

3.3.3 Arduino UNO & DHT22

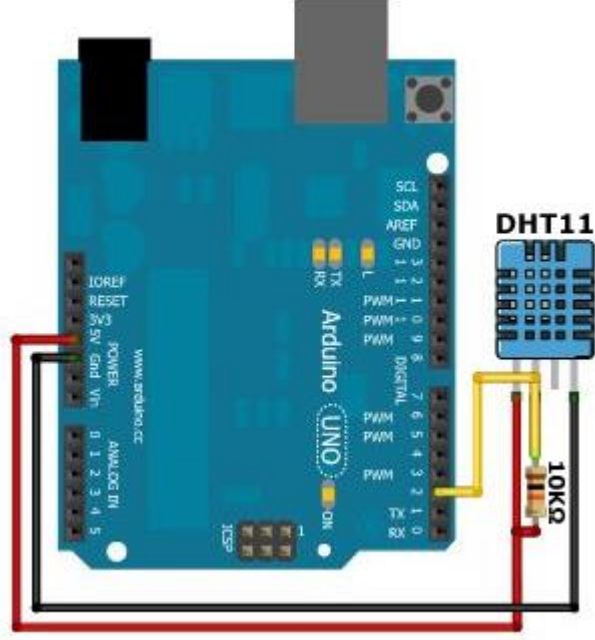
DHT22 modülü, Arduino UNO platform ile bağlantısı için üç pinin kullanılması yeterlidir. Bağlantı pinlerinin tanımlamaları Çizelge 3.3'te yapılmıştır.

DHT22 Pinleri	Arduino UNO Pinleri
V _{DD}	5V
Data out	Dijital 2 (D2)
Null	Bağlantı yok
Ground (toprak)	GND

Çizelge 3.3 DHT22 & Arduino UNO Bağlantıları

Kurulum aşamasında V_{DD} ile DATA bağlantıları arasına 10kΩ'luk bir direnç konarak Data pini üzerinde orta güçte bir pull-up oluşturulabilir. Arduino UNO'nun kendi pull-up'ları olmasına karşın bu değerler 100kΩ ile desteklenmiştir. Bu nedenle çok zayıf kalmaktadır.

DHT22'nin Arduino UNO ile bağlantısı açık bir şekilde Şekil 3.4'te gösterilmiştir.



Şekil 3.4 Arduino UNO & DHT22 Bağlantısı

4. BASINÇ

Basınç, bir yüzey üzerine etkide bulunan dik kuvvetin, birim alana düşen miktarı olarak tanımlanır. Katı, sıvı ve gazların bulundukları yüzeye uyguladıkları basıncı düşünecek olursak, projede açık hava aktivitelerinde sporcu veya çalışanların açık hava basıncını ölçmelerinde kullanışlı bir cihaz geliştirilmesi hedeflenmiştir.

Kısaca birim alana etki eden kuvvettir. Buda $P = \frac{F}{A}$ denklemi ile hesaplanır. SI birimi Pascal = N/m²'dir

İçinde bulunduğumuz atmosferde hali hazırda mevcut olan bir atmosfer basıncı vardır. Bu basınç, atmosfer açık olan gaz ve sıvı sistemlerine tesir eder. Dolayısıyla atmosfere açık sistemlerde basınç gerçek basınç değildir. Buna bir de atmosferin yaptığı basıncı eklemek gerekir. Bu toplam basınca mutlak basınç denir.

4.1 ÖLÇÜM METOTLARI

Manometre, gaz veya sıvı akışkanlarının basıncını ölçmede kullanılmaktayken, barometre atmosfer basıncını ölçen alettir.

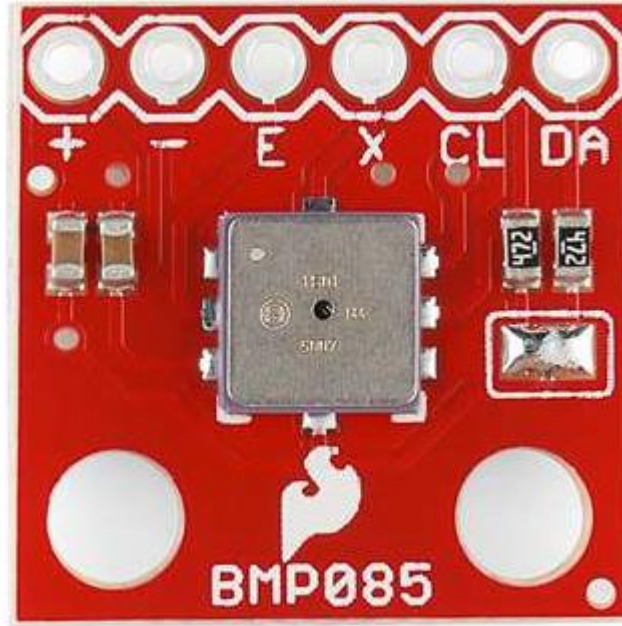
- Mekanik Basınç Ölçme Cihazları
Akışkanlı Manometreler: İkiye ayrılmaktadır bunlar manometre ve barometerdir. Projede kullanacağımız sensör barometrik basınç sensörüdür.
Ölü Ağırlık Test Cihazı
Bourdon Tüplü Basınç Ölçer
Diyaframlı ve Körüklü Basınç Ölçerler
Bridgman Cihazı
- Vakum Ölçme Cihazları

4.2 BMP085 BASINÇ SENSÖRÜ

BMP085, yüksek hassasiyette, çok düşük seviyede enerji tüketen ve dijital çıkış veren basınç sensörüdür. Barometrik BMP085 sensörü bu iki özelliği sayesinde mobil uygulamalarda sıklıkla kullanılmaktadır.

Herhangi bir mobil uygulama için tercih edilen BMP085 sensörü hassas barometrik basınç ölçümü gerektirmektedir. Örnek olarak mobil telefonlar, PDA, kişisel GPS tabanlı navigasyon cihazları ve proje konusunda da olduğu gibi gelişmiş dış mekân ekipmanlarıdır.

BMP085 barometrik basınç sensörü Şekil 4.1’de gösterilmektedir.



Şekil 4.1 BMP085 Basınç Sensörü

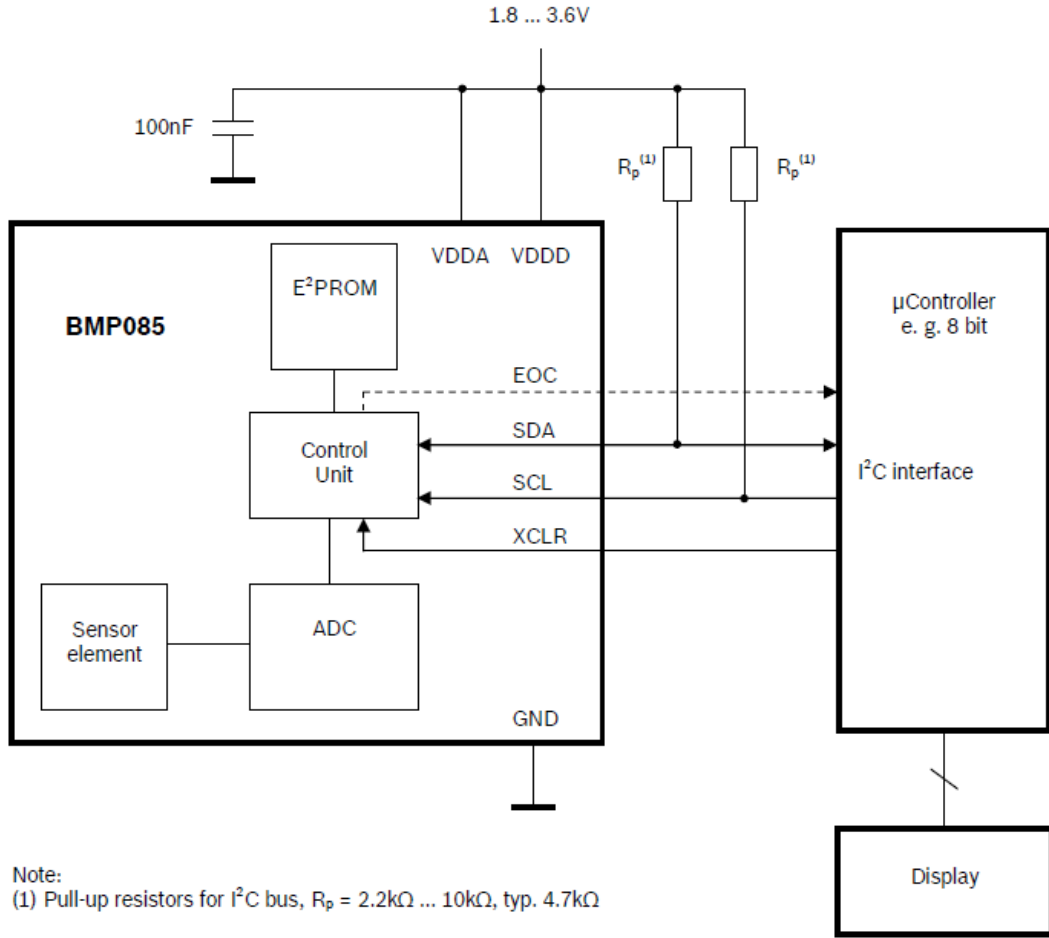
4.2.1 Sensör Özellikleri

BMP085 sensörü basıncın yanında sıcaklık ölçümü içermektedir. Tam kalibrasyonludur. Bağladığı mikrodenetleyici veya platformlar ile I²C seri bağlantı türünü kullanarak haberleşir. BMP085 basınç sensörünün çalışma sıcaklığı, besleme voltajı gibi bilgiler Çizelge 4.1’de verilmiştir.

Parametre	Sembol	Min.	Mak.	Birim
Çalışma sıcaklığı	T _A	-40	+85	°C
Besleme Voltajı	V _{DD}	1,8	3,6	V
Peak akımı	I _{PEAK}		1000	µA
Çalışma Frekansı	F _{SCL}		3,4	MHz
Ölçüm Aralığı (V _{DD} = 3,3V)		700	1100	hPa
Ölçüm Keskinliği (V _{DD} = 3,3V)		-2,5	+2,5	
Çıkış veri çözünürlüğü (basınç)			0,01	
Çıkış veri çözünürlüğü (sıcaklık)			0,1	°C

Çizelge 4.1 BMP085 Basınç Sensörü Özellikleri

BMP085 modülü ile bir mikrodenetleyici arasındaki pin bağlantıları Şekil 4.2’de gösterilmiştir.



Şekil 4.2 BMP085 Pin Tanımlamaları

Ölçülen değere göre deniz seviyesi basınç hesaplaması formülü.

$$p_0 = \frac{p}{\left(1 - \frac{Yükseklik}{44330}\right)^{5,255}}$$

4.2.2 BMP085 & Arduino UNO

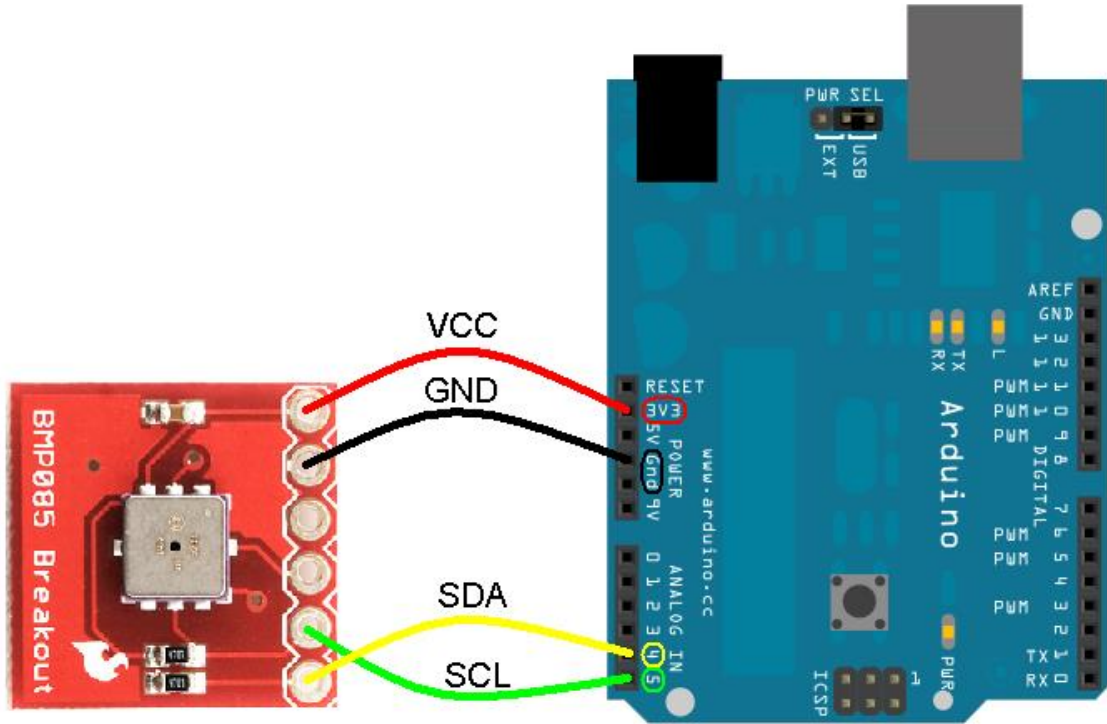
BMP085 sensörüyle Arduino UNO arasındaki bağlantının sağlanması dört hat üzerinden sağlanmıştır.

Arduino ile BMP085 arasındaki bağlantılar Çizelge 4.2’de tanımlanmış ve bu iki cihaz arası bağlantılar Şekil 4.3’te gösterilmiştir. Sensör ve Arduino arasındaki iletişim seri I²C bağlantısıyla sağlanmıştır. Bunun için Sensörün Data verilerini taşıyan hat Arduino da A4 pinine, platformun saat sinyalini taşıyan hat A5 pinine

bağlanmıştır. 3,3V ile çalışan sensöre güç sağlanması 3v3 pininden toprakta GND pininden sağlanmaktadır. Şekilde gösterilmemesine karşın SDA ve SCL girişlerine 4,7k Ω 'luk pull-up direnci bağlanmıştır.

BMP085 Pin	Arduino UNO pin
V _{CC}	3,3V
GND	GND
SCL	A5
SDA	A4

Çizelge 4.2 BMP085 & Arduino



Şekil 4.3 Arduino & BMP085

Yuakrındaki Şekil 4.3'de ve Çizelge 4.2'de de görüldüğü üzere

5. GERÇEK ZAMANLI SAAT TÜMLEŞİK DEVRESİ

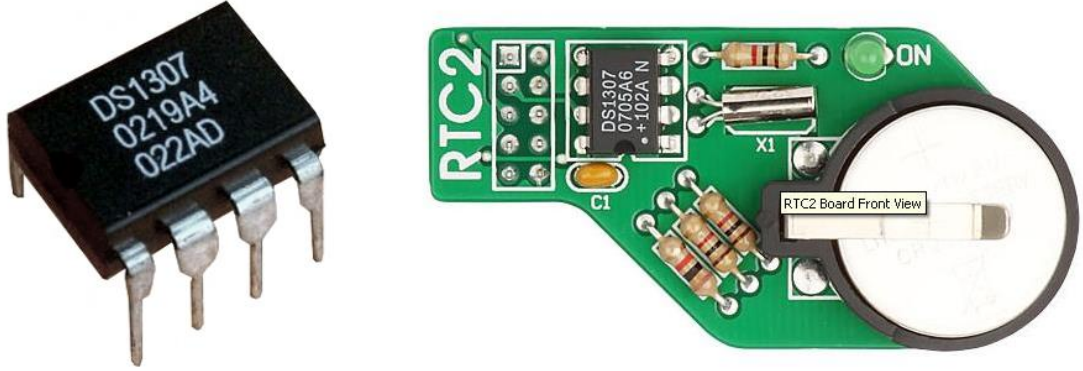
Gerçek zamanlı saat – GTÇ (RTC – Real-Time Clock), güncel zamanı gösteren bilgisayar saati olarak tanımlanır. Genellikle tümleşik devreler biçiminde tasarlanan bu saatler, kişisel bilgisayarlar, sunucular ve gömülü sistemler başta olmak üzere tüm elektronik aygıtlarda kullanılmaktadır.

Tasarlanan sistemde gerçek zamanlı saat kullanmanın avantajları; düşük güç tüketimine sahip olması ve süre kısıtlaması gerektiren işlemlere öncelik tanınmasına yardımcı olmasıdır

5.1 DS1307

Gerçek zamanlı saatler genellikle ayrı bir güç kaynağından beslenmektedirler. Böylelikle sistem kapalı dahi olsa saat verisinde herhangi bir yanlışlık oluşmaz.

DS1307'nin dış görünüşü Şekil 5.1'de gösterilmiştir.(soldaki resim DS1307 modülü, sağdaki resim gerçek zamanlı saat tümleşik devresi.)



Şekil 5.1 DS1307

5.2 Donanım Özellikleri

Projede kullanılacak gerçek zamanlı saatte kristal osilatörü kullanılmaktadır. DS1307 GÇS'sinin osilatör frekansı 32768 kHz olduğundan, GÇS tümleşik devresinde bir 32768 kHz'lik kristal kullanılmıştır.

DS1307 donanımı;

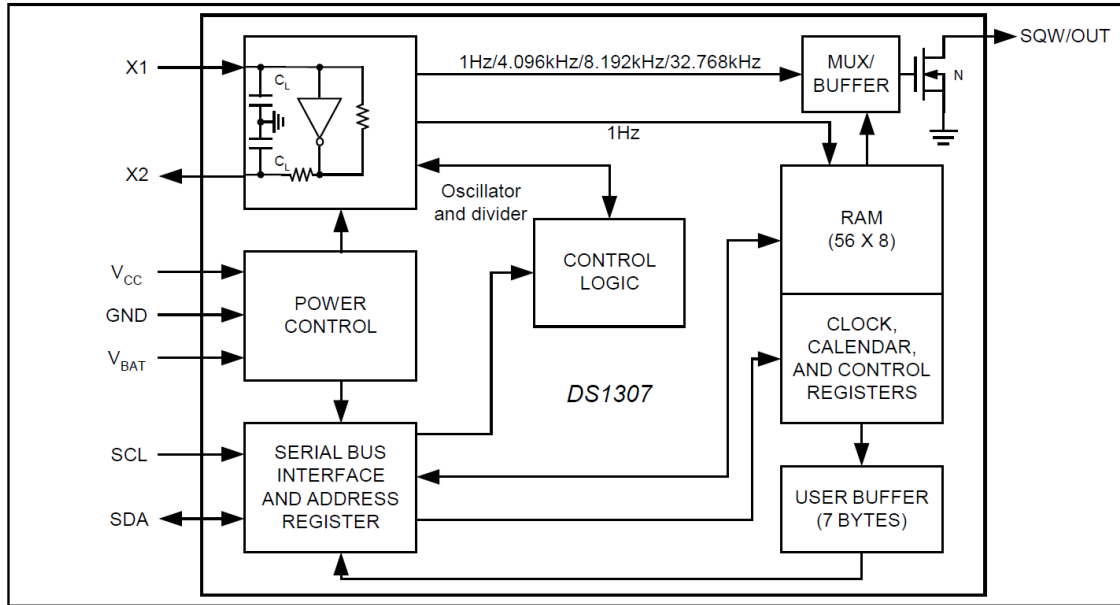
- Saniye, dakika, saat, gün, ay ve yıl (2100'e kadar) değerlerini sayabilmektedir.
- Mikrodenetleyici ve platformlarla iletişimi I²C seri bağlantısını kullanarak gerçekleştirmektedir.
- Programlanabilir kare dalga çıkış sinyali üretebilmektedir.
- Otomatik güç kesintisi algılaması ve anahtarlama devresine sahiptir.
- Osilatör çalışırken 500nA'den daha az akım modu bulunmaktadır.
- Opsiyonlu sanayi sıcaklık aralığı -40 ~ -50 °C'dir.

DS1307'nin çalışma gerilimi, sızıntı miktarı gibi değerler Çizelge 5.1'de verilmiştir.

Parametreler	Sembol	Min. ~ Mak.	Birim
Çalışma Sıcaklık Aralığı	T_A	0 ~ 70	°C
Besleme Gerilimi	V_{CC}	4,5 ~ 5,5	V
Batarya Voltajı	V_{BAT}	2,0 ~ 3,5	V
SCL Giriş Sızıntısı	I_{LI}	-1 ~ 1	μA
SDA, SQW I/O Sızıntısı	I_{LO}	-1 ~ 1	μA
SCL Saat Frekansı	f_{SCL}	0 ~ 100	kHz
Pin Kapasite (SDA, SCL)	$C_{I/O}$	10	pF

Çizelge 5.1 DS1307 Özellikleri

Ds1307 modülünün blok diyagramı Şekil 5.2'de gösterilmiştir.



Şekil 5.2 DS1307 Blok Diyagram

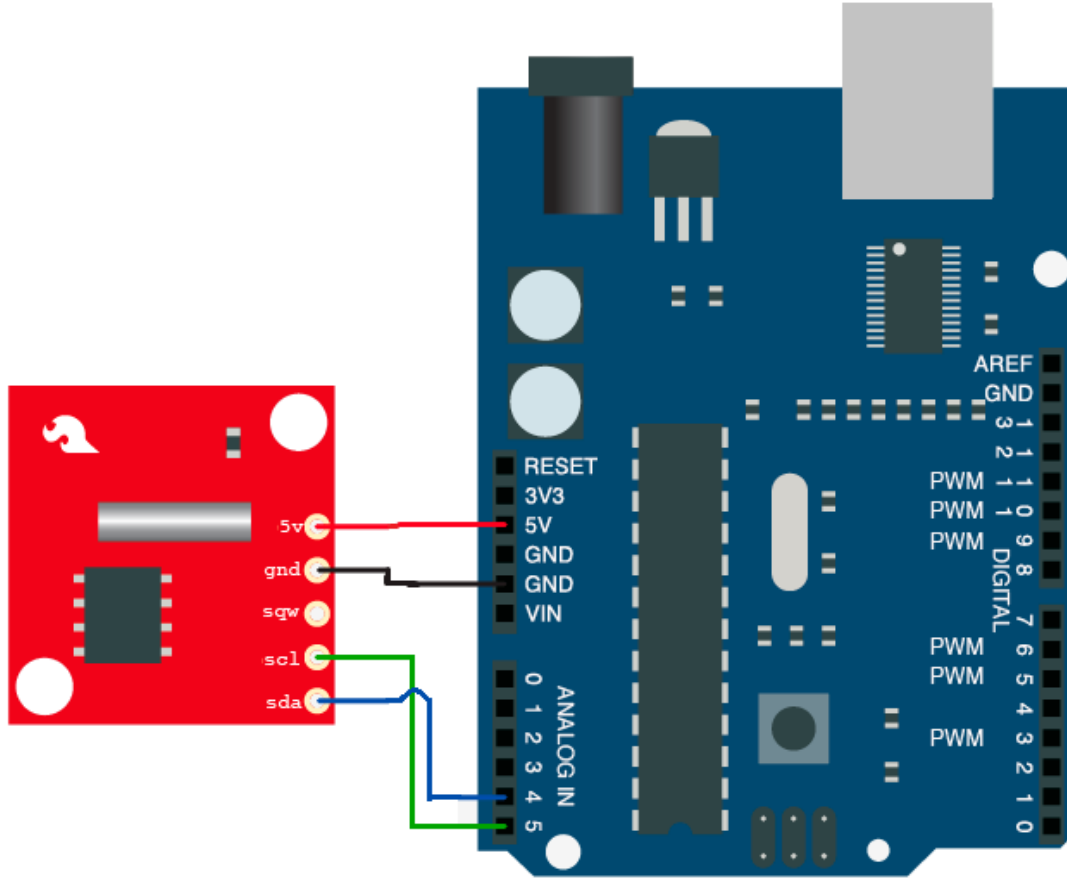
5.3 DS1307 & Arduino UNO

DS1307 modülü I²C seri bağlantı tipini kullandığından, Arduino ile iletişimde veri (SDA) ve saat (SCL) pin bağlantılarını A4,A5 ve gerekli güç pinleri üzerinden yapmalıyız. Modül ve platform arasındaki bağlantı Çizelge 5.2’de verilmiştir.

DS1307 Pin	Arduino UNO Pin
SDA	A4
SCL	A5
GND	GND
5v	5v

Çizelge 5.2 DS1307 & Arduino UNO

DS1307 ile Arduino arasındaki bağlantı Şekil 5.3 te gösterilmiştir.



Şekil 5.3 DS1307 & Arduino UNO

6. LCD

LCD sıvı kristal ekran (liquid-crystal display), elektrikle polarize edilebilen sıvının ışığı tek fazlı geçirmesi ve önüne eklenen bir kutuplanma filtresi ile gözle görülebilmesi ilkesine dayanan bir görüntü teknolojisidir. Sıvı kristal ekranlar düşük enerji tüketimi özelliğinden dolayı sıkça kullanılmaktadırlar.

6.1 LCD & Arduino

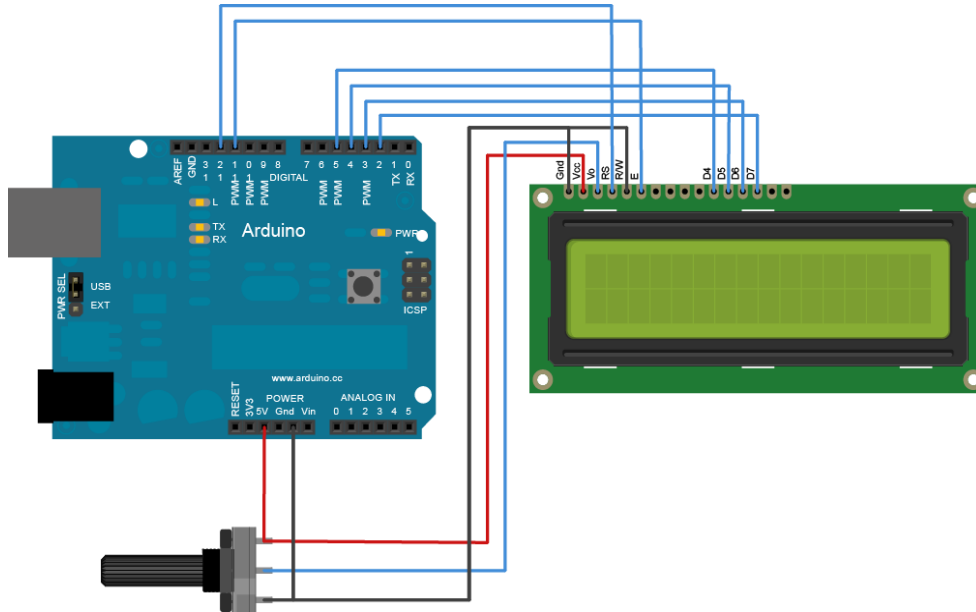
LCD'nin pinleri ile Arduino'nun pinleri Çizelge 6.1'de tanıtılmıştır.

LCD	Arduino UNO
V _{SS}	GND
R/W	GND
GND LCD	GND
HIGH LCD	5V
DB4	9
DB5	10
DB6	11
DB7	12
DB1,2,3	Bağlantı Yok
E	12
RS	11
V _{DD}	5V

Çizelge 6.1 LCD & Arduino

V_{DD} ve HIGH LCD pinlerine birer pot bağlanarak LCD'nin verdiği arka plan ışık şiddeti ve yazı ışık şiddeti değiştirilmektedir.

Arduino UNO ile LCD arasındaki bağlantılar Şekil 6.1'de gösterilmiştir.



Şekil 6.1 LCD & Arduino

7. KAYNAKÇA

- Jon S. Wilson (2004), **Sensor Technology Handbook Book**, Newness
- Maik Schmidt (2011), **Arduino: A Quick Start Guide (Pragmatic Programmers)**, Pragatic Bookshelf
- Joshua Noble (2009), **Programming Interactivity: A Designer's Guide to Processing, Arduino, and Openframeworks**, O'Reilly Media
- <http://arduino.cc/> (18.04.2013)
- <http://www.robots101.com/> (18.04.2013)

EK – 1: KOD

```
int sensePin = 5;
```

```
#include "DHT.h"
```

```
#include <LiquidCrystal.h>    //LCD  
kütüphanesi
```

```
#include <Wire.h>            // Wire  
kütüphanesini
```

```
#include <DS1307new.h>        //  
DS1307new kütüphanesini
```

```
// Değişkenler ve sabitler
```

```
uint16_t startAddr = 0x0000; // NV-  
RAM'de saklamak için başlangıç  
adresi
```

```
uint16_t lastAddr;           // NV-  
RAM'de saklamak için yeni adres
```

```
uint16_t TimelsSet = 0xaa55;  //  
Saatin tekrar ayarlanmamasına  
yardımcı olur.
```

```
int ac1; // BMP085 için kalibrasyon  
değişkenleri
```

```
int ac2;
```

```
int ac3;
```

```
unsigned int ac4;
```

```
unsigned int ac5;
```

```
unsigned int ac6;
```

```
int b1;
```

```
int b2;
```

```
int mb;
```

```
int mc;
```

```
int md;
```

```
long b5;
```

```
short temperature;
```

```
long pressure;
```

```
const unsigned char OSS = 0;
```

```
const int buttonPin = 2;      //  
pushbutton pininin numarası
```

```
LiquidCrystal lcd(7,8,9,10,11,12);
```

```
#define DHTTYPE DHT22        // DHT  
22 (AM2302)
```

```
#define BMP085_ADDRESS 0x77
```

```
DHT dht(sensePin, DHTTYPE);
```

```
// ÖNEMLİ NOT BASINÇ  
yükseklikten önce algılanmalıdır !!!
```

```
// yükseklik hesaplamaları
```

```
const float p0 = 101325;      // deniz  
seviyesindeki basınç (Pa)
```

```
float altitude;
```

```
void setup() {
```

```
    lcd.begin(20, 4);          // 20x4 lük  
LCD
```

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

```
    Wire.begin();
```

```
    bmp085Calibration();
```

```
    dht.begin();
```

```
    RTC.setRAM(0,              (uint8_t  
*)&startAddr, sizeof(uint16_t));
```

```
    RTC.getRAM(54,             (uint8_t  
*)&TimelsSet, sizeof(uint16_t));
```

```

    lcd.setCursor(4,0);          // Position
    cursor on line x=3,y=1

    lcd.print("VOLKAN ERKAN");    //
    Print a message to the LCD

    lcd.setCursor(2,2);
    lcd.print("COK AMACLI METRE");
    lcd.setCursor(7,3);
    lcd.print("Arduino");
    delay(2000);

    if (TimelsSet != 0xaa55)      //saat
    önceden ayarlandı mı ?
    {
        RTC.stopClock();
        RTC.fillByYMD(2013,5,13);
        RTC.fillByHMS(20,59,0);
        RTC.setTime();
        TimelsSet = 0xaa55;
        RTC.setRAM(54,            (uint8_t
        *)&TimelsSet, sizeof(uint16_t));
        RTC.startClock();
    }
    else
    {
        RTC.getTime();
    }
}

void loop() {
    float t = dht.readTemperature();

```

```

    float h = dht.readHumidity();

    temperature                =
    bmp085GetTemperature(bmp085Re
    adUT());

    pressure                    =
    bmp085GetPressure(bmp085Read
    UP());

    altitude = (float)44330 * (1 -
    pow(((float)                pressure/p0),
    0.190295));

    /* Bu kodlar bilgisayar üzerinden
    kontrol yapıldığında kullanılacak

    Serial.print("Temperature: ");
    Serial.print(temperature, DEC);
    Serial.println(" *0.1 deg C");
    Serial.print("Pressure: ");
    Serial.print(pressure, DEC);
    Serial.println(" Pa");
    Serial.print("Altitude: ");
    Serial.print(altitude, 2);
    Serial.println(" m");
    Serial.println();

    Serial.print("SICAKLIK: ");
    Serial.print(t);
    Serial.print("NEM");
    Serial.print(h);
    Serial.println(" Pa");
    Serial.print("Altitude: ");
    Serial.print(altitude, 2);
    Serial.println(" m");
    Serial.println();

```

```

*/
    RTC.getTime();    // Saat ve Tarih
verilerini al
    lcd.clear();
    lcd.setCursor(7,0);
    if (RTC.hour < 10)
    {
        lcd.print("0");
        lcd.print(RTC.hour, DEC);
    }
    else
    {
        lcd.print(RTC.hour, DEC);
    }
    lcd.print(":");
    if (RTC.minute < 10)
    {
        lcd.print("0");
        lcd.print(RTC.minute, DEC);
    }
    else
    {
        lcd.print(RTC.minute, DEC);
    }
    lcd.print(":");
    if (RTC.second < 10)
    {
        lcd.print("0");
        lcd.print(RTC.second, DEC);
    }
    else
    {
        lcd.print(RTC.second, DEC);
    }
    lcd.setCursor(3,1);
    if (RTC.day < 10)
    {
        lcd.print("0");
        lcd.print(RTC.day, DEC);
    }
    else
    {
        lcd.print(RTC.day, DEC);
    }
    lcd.print("-");
    if (RTC.month < 10)
    {
        lcd.print("0");
        lcd.print(RTC.month, DEC);
    }
    else
    {
        lcd.print(RTC.month, DEC);
    }
    lcd.print("-");
    lcd.print(RTC.year, DEC); // Yılda
herhangi bir değişikliğe gerek yok
    lcd.print(" ");
    switch (RTC.dow)        // Haftanın
günü

```

```

{
    case 1:
        lcd.print("PZT");
        break;
    case 2:
        lcd.print("SAL");
        break;
    case 3:
        lcd.print("CAR");
        break;
    case 4:
        lcd.print("PER");
        break;
    case 5:
        lcd.print("CUM");
        break;
    case 6:
        lcd.print("CTS");
        break;
    case 7:
        lcd.print("PAZ");
        break;
}

Serial.println();
lcd.setCursor(0,2);
lcd.print (t);
lcd.print (" C");
lcd.setCursor(14,2);
lcd.print ("%");

        lcd.print (h);
        lcd.setCursor(0,3);
        lcd.print(pressure, DEC);
        lcd.print(" Pa");
        lcd.setCursor(13,3);
        lcd.print(altitude, 2);
        lcd.print("m");
        delay(900);          // 800mS saniye
        bekle
    }
    void bmp085Calibration()
    {
        ac1 = bmp085ReadInt(0xAA);
        ac2 = bmp085ReadInt(0xAC);
        ac3 = bmp085ReadInt(0xAE);
        ac4 = bmp085ReadInt(0xB0);
        ac5 = bmp085ReadInt(0xB2);
        ac6 = bmp085ReadInt(0xB4);
        b1 = bmp085ReadInt(0xB6);
        b2 = bmp085ReadInt(0xB8);
        mb = bmp085ReadInt(0xBA);
        mc = bmp085ReadInt(0xBC);
        md = bmp085ReadInt(0xBE);
    }

    short
    bmp085GetTemperature(unsigned
    int ut) // bmp085 ile sıcaklık
    hesaplaması
    {
        long x1, x2;

```

```

x1      =      (((long)ut      -      p = (b7<<1)/b4;
(long)ac6)*(long)ac5)>> 15;
else
x2 = ((long)mc << 11)/(x1 + md);
p = (b7/b4)<<1;
b5 = x1 + x2;

return ((b5 + 8)>>4);

}

long bmp085GetPressure(unsigned
long up)

{
    long x1, x2, x3, b3, b6, p;
    unsigned long b4, b7;

    b6 = b5 - 4000;
    // Calculate B3
    x1 = (b2 * (b6 * b6)>>12)>>11;
    x2 = (ac2 * b6)>>11;
    x3 = x1 + x2;
    b3 = (((((long)ac1)*4 + x3)<<OSS)
+ 2)>>2;

    // Calculate B4
    x1 = (ac3 * b6)>>13;
    x2 = (b1 * ((b6 * b6)>>12))>>16;
    x3 = ((x1 + x2) + 2)>>2;
    b4 = (ac4 * (unsigned long)(x3 +
32768))>>15;

    b7 = ((unsigned long)(up - b3) *
(50000>>OSS));
    if (b7 < 0x80000000)

        p = (b7<<1)/b4;
    else
        p = (b7/b4)<<1;

    x1 = (p>>8) * (p>>8);
    x1 = (x1 * 3038)>>16;
    x2 = (-7357 * p)>>16;
    p += (x1 + x2 + 3791)>>4;

    return p;
}

char bmp085Read(unsigned char
address)
{
    unsigned char data;

    Wire.beginTransmission(BMP085_A
DDRESS);
    Wire.write(address);
    Wire.endTransmission();

    Wire.requestFrom(BMP085_ADDRE
SS, 1);
    while(!Wire.available())
        ;

    return Wire.read();
}

```

```

int bmp085ReadInt(unsigned char
address)
{
    unsigned char msb, lsb;

Wire.beginTransmission(BMP085_A
DDRESS);

    Wire.write(address);

    Wire.endTransmission();

Wire.requestFrom(BMP085_ADDRE
SS, 2);

    while(Wire.available()<2)

        ;

    msb = Wire.read();
    lsb = Wire.read();

    return (int) msb<<8 | lsb;
}

unsigned int bmp085ReadUT()
{
    unsigned int ut;

Wire.beginTransmission(BMP085_A
DDRESS);

    Wire.write(0xF4); // Write 0x2E into
Register 0xF4

    Wire.write(0x2E); // This requests a
temperature reading

    Wire.endTransmission();

```

```

    delay(5);

    ut = bmp085ReadInt(0xF6);

    return ut;
}

unsigned long bmp085ReadUP()
{
    unsigned char msb, lsb, xlsb;

    unsigned long up = 0;

Wire.beginTransmission(BMP085_A
DDRESS);

    Wire.write(0xF4);

    Wire.write(0x34 + (OSS<<6));

    Wire.endTransmission();

    delay(2 + (3<<OSS));

Wire.beginTransmission(BMP085_A
DDRESS);

    Wire.write(0xF6);

    Wire.endTransmission();

Wire.requestFrom(BMP085_ADDRE
SS, 3);

    while(Wire.available() < 3);

    msb = Wire.read();

    lsb = Wire.read();

    xlsb = Wire.read();

    up = (((unsigned long) msb << 16) |
((unsigned long) lsb << 8) | (unsigned
long) xlsb) >> (8-OSS);

    return up;
}

```