

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**TEK ROTORLU WANKEL MOTORUNUN ATEŞLEME VE PÜSKÜRTME
ÜNİTELERİNİN (KONTROL ÜNİTESİNİN) TASARIMI, İMALATI VE
BİLGİSAYAR ARAYÜZÜ İLE KONTROLÜ**

YÜKSEK LİSANS TEZİ

Mehmet İlter ÖZMEN

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

MAYIS 2015

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**TEK ROTORLU WANKEL MOTORUNUN ATEŞLEME VE PÜSKÜRTME
ÜNİTELERİNİN (KONTROL ÜNİTESİNİN) TASARIMI, İMALATI VE
BİLGİSAYAR ARAYÜZÜ İLE KONTROLÜ**

YÜKSEK LİSANS TEZİ

**Mehmet İlter ÖZMEN
518101024**

Mekatronik Mühendisliği Anabilim Dalı

Mekatronik Mühendisliği Programı

Tez Danışmanı: Doç. Dr. Cemal BAYKARA

MAYIS 2015

İTÜ, Fen Bilimleri Enstitüsü'nün 518101024 numaralı Yüksek Lisans Öğrencisi **Mehmet İlter ÖZMEN**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**TEK ROTORLU WANKEL MOTORUNUN ATEŞLEME VE PÜSKÜRTME ÜNİTELERİNİN (KONTROL ÜNİTESİNİN) TASARIMI, İMALATI VE BİLGİSAYAR ARAYÜZÜ İLE KONTROLÜ**” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Doç. Dr. Cemal BAYKARA**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Doç. Dr. İlker Murat KOÇ**
İstanbul Teknik Üniversitesi

Doç. Dr. Cüneyt FETVACI
İstanbul Üniversitesi

Teslim Tarihi : **04 Mayıs 2015**
Savunma Tarihi : **29 Mayıs 2015**

Biricik Aileme,

ÖNSÖZ

Wankel motorunun çalışma prensibinden, sistemin elektronik ve yazılım gereksinimlerine kadar tüm bilgi birikimlerini ve çalışmamın temelini oluşturan araştırmalarını benimle paylaşan, öğreten, bir danışmandan öte yaklaşımı ile her türlü sorunumda yardımcı olup yol gösteren yüksek lisans tez danışmanım Sayın Doç.Dr. Cemal Baykara'ya ve Sayın Yard.Doç.Dr. Osman Akın Kutlar'a teşekkürlerimi sunarım.

Çalışmalarında karşılaştığım teknik sorunlarda bana yardımcı olan ve zamanında yetiştirebilmem için ellerinden geleni yapan Ömer Cihan ve Abdurrahman Demirci'ye çok teşekkür ederim.

Geçmişte benzer konularda hazırladıkları bitirme tasarım projeleri ve yüksek lisans tezleriyle katkıda bulunan tüm İTÜ öğrencilerine teşekkür ederim.

Adını taşımaktan onur duyduğum, sayesinde üniversiteye kadar özel okullarda okuduğum, hayatı boyunca (ve hala daha) maddi manevi desteğini hep yanımda hissettiğim rahmetli dedem Hacı Mehmet İlter'e, her fırsatta “sen olmasaydın buralara gelemezdim ben” dediğim, kendi yemeyip bana yediren, hastayken dahi önceliği ben olan, hayatını her anlamda bana adayan canım anneme, tezime faydası olur diye eşe dosta haber salan, bana destek olabilmek için elinden gelenin fazlasını yapan, düşüğüm her an elimden tutan babama, “çekirdek ailemizin” bir parçası olan, anlayışlı tutumları ve ufkumu açan fikirleriyle her zaman destekçim olan teyzem ve enişteme ve “diplomayı görmeden ölmeyeceğim” diyerek çalışmalarında ve hayatımın her anında beni kamçılaman “yaşam sponsorum” canım anneanneme minnettarım.

Tez çalışmamın başından sonuna kadar yanımda olan, derslerinden geri kalma pahasına bana yardım eden, adeta kendi tezini yapıyormuş gibi sahiplenen yoldaşım Tunç Köse'ye, takıldığım her noktada yardımcı olan, yaratıcı fikirleriyle çalışmalarına ışık tutan Altay Arbak'a, askerde olmasına rağmen izin alıp gerektiğinde sabaha kadar benimle çalışan, “uğurlu elleriyle” yaşadığım tüm teknik sorunları kolayca aşmamı sağlayan kardeşim Metin Usan'a ve tez sürecim boyunca her zaman yanımda olup moral veren kız arkadaşım Giulia Perri'ye ayrıca sonsuz teşekkürlerimi sunarım.

Mayıs 2015

Mehmet İlter ÖZMEN
Makina Mühendisi

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	vii
İÇİNDEKİLER	ix
KISALTMALAR	xi
ÇİZELGE LİSTESİ.....	xiii
ŞEKİL LİSTESİ.....	xv
SEMBOL LİSTESİ	xvii
ÖZET.....	xix
SUMMARY	xxi
1. GİRİŞ	1
1.1 Tezin Amacı	4
1.2 Araştırma Motorunun Tanıtılması.....	5
1.3 Wankel Motorunda Zamanlar	9
2. ATEŞLEME VE PÜSKÜRTME ELEMANLARININ TANITIMI	13
2.1 Yakıt Enjeksiyon Sistemi	13
2.1.1 Elektrikli yakıt pompası	13
2.1.2 Yakıt filtresi	13
2.1.3 Enjektör	15
2.2 Ateşleme Sistemi.....	16
2.2.1 Ateşleme voltajı ve enerjisi.....	17
2.2.2 Ateşleme zamanlaması.....	18
2.2.3 Buji.....	18
2.2.3.1 Bujiden beklenen özellikler	20
2.2.3.2 Buji seçimi	21
2.2.4 Ateşleme bobini	23
2.2.5 Buji kabloları.....	25
2.2.6 Akü.....	25
2.2.7 Ateşleme modülü	26
2.2.8 Solid state röle.....	28
2.3 Arttırımsal Enkoder.....	29
3. KONTROL SİSTEMİNİN DONANIMI.....	31
3.1 Sürücü Kartı Tasarım Ve İmalat Çalışmaları	31
3.1.1 Sürücü kartı üzerindeki elemanlar	32
3.2 Solid State Röle	34
3.2.1 Ateşleme bobininin solid state röle ile kullanımı.....	37
3.3 Deneyisel Veriyi Hafızalama Çalışmaları	39
3.3.1 Eeprom	41
3.4 Kontrol Cihazı	45
4. KONTROL SİSTEMİ YAZILIMI	49
4.1 Kontrolcü Seçimi.....	49
4.1.1 Arduino modeli seçimi.....	51
4.1.2 Arduino mega 2560 özellikleri ve kullanımı	52

4.2 Kontrolcü Seçimi.....	54
4.3 Mikrokontrolörün Programlanması	60
4.4 C# Kontrol Arayüzü Yazılımı	68
4.4.1 C# yazılımı ile arduino'nun seri haberleşmesi.....	69
4.4.2 C# kontrol arayüzünün tanıtılması	71
4.5 Eeprom Yazılımı Ve Arduino'ya Kurulumu	73
4.6 Ek Yazılımlar.....	78
4.6.1 Çevrim atlatma yönteminin tanıtılması	78
4.6.2 Çevrim atlatma yönteminin dört zamanlı motora uygulanışı	79
4.6.3 Güncellenen kodlar ve arayüzün tanıtılması	81
5. DENEYLER.....	79
5.1 Ön Çalışmalar.....	79
5.1.1 Mikro enkoder ile pals okuma ve okunan palsi işleme	79
5.1.2 C# ile arduino arasında seri haberleşme ile veri gönderme ve alma	80
5.2 Enjektör Debi Karakteristiği Deneyi	85
5.3 Osiloskop Ölçümleri Ve Matkap Deneyi	88
6. SONUÇLAR VE ÖNERİLER.....	93
KAYNAKLAR.....	95
EKLER.....	97
ÖZGEÇMİŞ	107

KISALTMALAR

ACK	: Acknowledge (Onay Biti)
AÖN	: Alt Ölü Nokta
I2C	: Two wire Interface (İki Kablolulu Arabirim)
İTÜ	: İstanbul Teknik Üniversitesi
KMA	: Krank Mili Açısı
M	: Metrik
ÜÖN	: Üst Ölü Nokta
PWM	: Darbe Genişlik Modülasyonu
R/W	: Read/Write (Okuma/Yazma)
N	: Normal Çevrim
S	: Atlatılmış Çevrim
SCL	: Serial Clock Line (Seri Saat Hattı)
SDL	: Serial Data Line (Seri Veri Hattı)

ÇİZELGE LİSTESİ

Sayfa

Çizelge 1.1 : RENESİS RX-8 13B motorunun teknik özellikleri	7
Çizelge 2.1 : NGK NY34-18 S30 İridyum bujilerin özellikleri	22
Çizelge 2.2 : AC Delco R4602 bujinin özellikleri.....	28
Çizelge 2.3 : Mako BAE207A ateşleme bobinin direnç ve endüktans değerleri	28
Çizelge 2.4 : Heidenhain Rod426 enkoder özellikleri.....	30
Çizelge 3.1 : Norton SDI-1105D Solid-State röle özellikleri [26]..	37
Çizelge 3.2 : Enkoder bağlantısının Numaralandırması	47
Çizelge 3.3 : Motor bağlantısının Numaralarına Göre	47
Çizelge 4.1 : Arduino Mega 2560 karakteristik özellikleri	52
Çizelge 4.2 : Değişken Tablosu [13].	62
Çizelge 5.1 : 1-5 bar basınçta Denso enjektör debi karakteristiği	87

ŞEKİL LİSTESİ

Sayfa

Şekil 1.1 : İşleyiş Şeması	5
Şekil 1.2 : Çift rotorlu RENESİS motoru	7
Şekil 1.3 : İki rotorlu High Power motorda portlar	8
Şekil 1.4 : Motor moment eğrileri	9
Şekil 1.5 : Tek rotorlu Wankel motorunun prototipi	10
Şekil 1.6 : Wankel motorunda zamanlar [7].....	11
Şekil 2.1 : Püskürtme akış şeması [4].....	14
Şekil 2.2 : Bosch elektrikli yakıt pompası ve AVL yakıt şartlandırma cihazı [4].....	14
Şekil 2.3 : Mazda RX8 Denso 026450x12 enjektör	16
Şekil 2.4 : Ateşleme işleyiş şeması.....	17
Şekil 2.5 : Buji kesiti [18].....	19
Şekil 2.6 : Sıcak ve soğuk buji kesitleri [8]	21
Şekil 2.7 : NGK iridyum bujiler	22
Şekil 2.8 : Mazda RX8 ateşleme bobini	24
Şekil 2.9 : Mazda RX8 ateşleme bobininin çalışma mekanizması.....	24
Şekil 2.10 : Karbon dirençli ateşleme kablolarından bir kesit [23]	25
Şekil 2.11 : GM DR100 ateşleme modülü.....	27
Şekil 2.12 : Mazda RX8 bobininin içerisindeki anahtarlama elemanı	28
Şekil 2.13 : Ateşleme bobini ve ateşleme bobini ile kıvılcım oluşturma denemeleri	29
Şekil 3.1 : Proteus devre simulasyonundan bir Görüntü	31
Şekil 3.2 : Işınsal bağlayıcı TLP521-1 [Ek A.1]	33
Şekil 3.3 : İlk denemelerde kullanılan sürücü kartı	34
Şekil 3.4 : Solid-State rölenin çalışma mekanizması.....	36
Şekil 3.5 : Solid-State rölenin boyutları [11].....	36
Şekil 3.6 : Norton SDI-1105D kart tipi Solid-State röle	37
Şekil 3.7 : Sürücü röle – ateşleme bağlantı şeması.....	38
Şekil 3.8 : İmal edilmesi planlanan SRAM'in baskı devre çizimi	39
Şekil 3.9 : Örnek SRAM kartı komponentleri	40
Şekil 3.10 : Örnek SRAM kartın son hali.....	40
Şekil 3.11 : Atmel 24LC EEPROM.....	42
Şekil 3.12 : 24LC256 nın adres değerleri	42
Şekil 3.13 : EEPROM ve Arduino bağlantısı	43
Şekil 3.14 : EEPROM'un Arduino'ya bağlandıktan sonraki görünümü	44
Şekil 3.15 : Kontrol cihazının Solidworks'te çizimi	45
Şekil 3.16 : Kontrol cihazının dış görünüşü	45
Şekil 3.17 : Kontrol cihazının iç görünüşü	46
Şekil 3.18: Enkoder motor tesisatına bağlanmadan önceki bağlantısı	47
Şekil 4.1 : Gömülü sisteme ait bir blok diyagramı [13]	49
Şekil 4.2 : Sırasıyla; ARM, Beagleboard ve Cubieboard marka mikrokontrolcüler.	50
Şekil 4.3 : Arduino Mega 2560.....	53
Şekil 4.4 : Arduino IDE arayüzü [4].....	55

Şekil 4.5 : File menüsü	57
Şekil 4.6 : Edit menüsü	57
Şekil 4.7 : Sketch menüsü.....	58
Şekil 4.8 : Tool menüsü	59
Şekil 4.9 : Timer/Counter kurulumu ve sayıcıyı sıfırlamak için yazılan kod.....	61
Şekil 4.10 : Değişken tanımlamaları ile ilgili yazılan kodun bir parçası	62
Şekil 4.11 : Sürekli püskürtmenin önüne geçmek için yazılmış kodun bir parçası ...	63
Şekil 4.12 : Ateşleme ve püskürtme sürelerini kontrol etmek için yazılan kod	64
Şekil 4.13 : Aygıt yöneticisinden Arduino'nun seri haberleşme için COM portu	65
Şekil 4.14 : Serial Monitor'un görünümü	66
Şekil 4.15 : Seri haberleşmenin başlatılması ve verilerin serial monitor ile gösterimi	66
Şekil 4.16 : Arayüzden gelen verinin alınması ile ilgili yazılan kod	67
Şekil 4.17 : Paket halinde gelen verilerin değişkenlere atanması için kodun bir kısmı	67
Şekil 4.18 : Gelen string halindeki karakterlerin, integer'a çevrildiği kodun bir kısmı	68
Şekil 4.19 : 2 ms püskürtme süresiyle dakikada 1000 püskürtme için yazılan kod ..	68
Şekil 4.20 : Örnek arayüzde seri portun kurulması	69
Şekil 4.21 : Bilgisayar ile sistemin haberleşmesi	71
Şekil 4.22 : C# arayüzü.....	71
Şekil 4.23 : I2C veriyolunun şematik görünümü	73
Şekil 4.24 : SDL ve SCL hatlarından komut dizisi başlatma ve durdurma	74
Şekil 4.25 : Adres, R/W ve ACK bitlerinin gösterimi	75
Şekil 4.26 : EEPROM'a veri yazdırmak için yazılan kodun bir parçası	77
Şekil 4.27 : EEPROM'dan veri okumak için yazılan kodun bir parçası	77
Şekil 4.28 : Kurulum ayarlarını içeren kodun bir parçası.....	77
Şekil 4.29 : NS çevrimi.....	79
Şekil 4.30 : NSS çevrimleri	80
Şekil 4.31 : Güncellenen seri haberleşme kodu	81
Şekil 4.32 : Sistemi kapatmak için yazılan kodun bir parçası	82
Şekil 4.32 : Güncellenen kontrolcü arayüzü	82
Şekil 5.1 : Enkoderden pals okumak ve işlemek için kurulan düzenek.....	83
Şekil 5.2 : Enkoderden pals okuma ve işleme ile ilgili kodun bir parçası	84
Şekil 5.3 : Seri haberleşme denemelerinde kullanılan LED düzeneği.....	85
Şekil 5.4 : Seri haberleşme denemelerinde kullanılan arayüz	85
Şekil 5.5 : Enjektör debi karakteristiği deney düzeneği	86
Şekil 5.6 : Denso enjektör debi karakteristiği.....	87
Şekil 5.7 : Osiloskop ölçümü denemesinden bir kesit	89
Şekil 5.8 : 600 d/d için osiloskop ekran görüntüsü.....	89
Şekil 5.9 : 1000 d/d için osiloskop ekran görüntüsü.....	90
Şekil 5.10 : 3000 d/d için osiloskop ekran görüntüsü.....	90
Şekil 5.11 : Matkap deneyi	91
Şekil A.1 : TLP521 ışınsal bağlayıcı teknik detayı	88
Şekil A.2 : IRFZ44n mosfet teknik detayı	89
Şekil A.3 : Heidenhain ROD 426 enkoder teknik detayı.....	90
Şekil A.4 : Bu323A transistör teknik detayı	91
Şekil A.5 : BAE207A ateşleme bobini teknik detayı	92
Şekil A.6 : GM DR100 ateşleme modülü teknik resmi	93
Şekil A.7 : GM DR100 modül içindeki MC3334 transistör teknik detayı	94

SEMBOL LİSTESİ

d/d	: Devir/dakika
hp	: Beygir gücü
Hz	: Hertz
J	: Joule
Kbps	: Kilo byte per second (Saniyede bir bit)
kV	: Kilovolt
KW	: KiloWatt
L	: Litre
mm	: Milimetre
ms	: Milisaniye
n	: Motor devir sayısı
N_i	: İndike güç
Nm	: Newtonmetre
p_{mi}	: Ortalama indike basınç
rad	: Radyan
s	: Saniye
T	: Sıcaklık
V	: Hacim
V_H	: Bir silindirin strok hacmi
λ	: Hava fazlalık katsayısı
μs	: Mikrosaniye

TEK ROTORLU WANKEL MOTORUNUN ATEŞLEME VE PÜSKÜRTME ÜNİTELERİNİN (KONTROL ÜNİTESİNİN) TASARIMI, İMALATI VE BİLGİSAYAR ARAYÜZÜ İLE KONTROLÜ

ÖZET

Bir enerji türünü mekanik işe çeviren makinelere genel olarak motor denir. Dönüştürülen enerji türlerine göre motorlar Isı(Termik), Elektrik, Nükleer veya Hidrolik olabilmektedirler. Isı (Termik) motorlarda yakıtların kimyasal enerjisi yanma ve oksidasyon sonucu önce ısı enerjisine dönüşür, bunun sonucunda gazların sıcaklığı ve basıncı hızla yükselir ve bu gazların genişlemesi ile mekanik iş elde edilir. Yakıtın kimyasal enerjisinin motorun (silindirin) içinde ısı enerjisine dönüştüğü İçten Yanmalı Motorlarda doğrudan basıncı ve sıcaklığı yükseltilmiş yanma ürünleri kullanılmaktadır. Bu motorlara örnek olarak Gaz Tribünleri, Jet, Dizel, Otto ve Wankel Motorları verilebilir.

Silindir yapılarına göre karşılaştırıldığında, yanma ile pistonların aşağı yukarı itilmesi yerine düz bir diskin döndüğü Wankel Motorları dünyanın en verimli, devirli ve güçlü motorları arasında gösterilmektedir. 1950 ve 60'lı yıllarda Alman Mühendis Felix Wankel tarafından geliştirilen, şimdilerde ise daha teknolojik versiyonları sadece bazı Mazda modellerinde kullanılan Wankel motorları, klasik motorlara göre üçte iki oranında daha az hareketli parçaya sahiptir. İçten yanmalı olmasına karşın, daha farklı bir iç yapısı bulunmaktadır. Döner pistonlu motor olarak da bilinen bu motorlarda diğer içten yanmalı motorlardan farklı olarak, kenarları yayvanlaştırılmış üçgen şeklinde döner pistonlar kullanılır. Güç iletiminin doğrudan piston üzerine bağlı mil yardımı ile gerçekleştirilmesi sayesinde yapıları diğer motorlara göre daha az karmaşıktır. Düşük hacimden yüksek miktarda güç elde eden, basit, hafif bir motor olarak bilinen Wankel Motoru 1957'nin dünya hız rekorunu egale etmiştir. Yakıt tüketiminin fazlalığı ve ekonomik olmaması sebepleriyle yaygın olarak kullanılamamasına karşın yüksek motor devirlerine çıkabilmesi, verimli ve dayanıklı olması ve kompakt boyutları sayesinde spor otomobillerde tercih edilmiştir. Günümüzde de patentini elinde bulunduran Mazda Firmasının RX7 ve RX8 modellerinde Wankel Motoru kullanılmaktadır.

Bu çalışmada deney standartları açısından tek rotorlu hale çevrilmiş bir Wankel Motoru için püskürtme ve ateşleme zamanlamaları (avansları) ile enjektör püskürtme süresi ve ateşleme bobini dolma süresi (dwell süresi) değişkenlerinin bilgisayardan kontrolü yapılacak bir kontrol mekanizması tasarlanmış ve imal edilmiştir. Mikrodenetleyici kontrol kartı olarak Arduino kullanılmış, ateşleme ve püskürtme sinyallerini güçlendirmek için sürücü rölelerden oluşan bir kontrol kutusu imal edilmiştir.

Wankel motoru çevrimlerinin kontrolü ve motordan geri besleme almak için krank miline bir enkoder bağlanarak, sistemin işleyişi enkoder başlangıç sinyali sıkıştırma üst ölü noktasına gelecek şekilde tasarlanmıştır. Enjektör debisini hesaplamak için bir deney düzeneği oluşturulmuştur. Wankel motorunun çalışma prensibi hakkında bilgi verilmiş, ateşleme ve püskürtme elemanlarından ve bu elemanları seçerken hangi kriterlerin uygulanacağından bahsedilmiştir. Kontrol sistemi yük ve motor hızındaki değişimlere uyum sağlayacak şekilde tasarlanmıştır.

Kontrol sistemi deney motoruna bağlanmadan önce kurulan gerçek zamanlı bir deney düzeneğinde denenmiş ve tasarım kriterlerine bağlı olarak sorunsuz şekilde çalıştığı gözlemlenmiştir. Ayrıca ileride sistem üzerinde yapılacak çeşitli deneylerde veri depolamak ve gerektiğinde bu verileri kullanabilmek için harici bir hafızalama elemanı (EEPROM) devresi imal edilmiş, imal edilen devre elektronik sisteme eklenmiştir. İleride üzerinde çalışacak araştırmacılara kaynak olması açısından hafızalama elemanının kontrolcü programa kurulumu ve üzerine veri yazıp okumak için gerekli yazılım çalışmaları hakkında detaylı bilgi verilmiştir.

DESIGNING, PRODUCING AND COMPUTER BASED CONTROLLING IGNITION AND INJECTION UNITS (CONTROL UNITS) OF A SINGLE ROTOR WANKEL ENGINE

SUMMARY

An engine or motor is a machine designed to convert one form of energy into mechanical energy. In terms of converted energy types, engines are classified as Heat (Thermic), Electrical, Nuclear or Hydraulic. Electric motors are machines that convert electrical energy into mechanical energy. Electric motors are used to produce linear or rotary force (torque), and should be distinguished from devices such as magnetic solenoids and loudspeakers that convert electricity into motion but do not generate usable mechanical powers, which are respectively referred to as actuators and transducers. A hydraulic motor is a mechanical actuator that converts hydraulic pressure and flow into torque and angular displacement (rotation). The hydraulic motor is the rotary counterpart of the hydraulic cylinder. In nuclear engines a working fluid, usually liquid hydrogen, is heated to a high temperature in a nuclear reactor and then expands through a rocket nozzle to create thrust. Thus it converts nuclear energy into mechanical energy.

In heat motors, chemical energy of fuels transform to heat energy by burning and oxidation, then the temperatures and pressures of gases fastly increase and as a result of the distant of gases, mechanical work occurs. Where a working fluid, contained internally, is heated by combustion in an external source, through the engine wall or a heat exchanger is called external combustion engine. where the combustion of a fuel occurs with an oxidizer (usually air) in a combustion chamber that is an integral part of the working fluid flow circuit is called internal combustion engine. Internal combustion engines use fuels that produce high temperature and pressure. Gas turbines, Jet, Diesel, Otto and Wankel engines can be given as examples of internal combustion engines.

If we need to compare in the way of cylinder moulds, instead of the common piston engines which are reciprocating pistons, an eccentric rotary design converts pressure into rotating motion is in use in Wankel engines. Over the commonly used reciprocating piston designs, the Wankel engine delivers advantages of simplicity, smoothness, compactness, high revolutions per minute and a high power to weight ratio. In Wankel motors (namely rotary engines), parts rotate in one direction as opposed to the common piston engines which have pistons that moves in two opposite directions. The four-stroke cycle occurs in a moving combustion chamber inside of an oval shaped housing, and a rotor that is similar in shape to a Reuleaux triangle with sides somewhat flatter.

In the Wankel engine, the four strokes of a Otto cycle piston engine occur in the space between a three-sided symmetric rotor placed in a housing. In a Wankel engine, the oval-like epitrochoid-shaped housing surrounds a rotor which is triangular with bow-shaped flanks (often confused with a Reuleaux triangle, a three-pointed curve of constant width, but with the bulge in the middle of each side a bit

more flattened). The theoretical shape of the rotor between the fixed corners is the result of a minimization of the volume of the geometric combustion chamber and a maximization of the compression ratio, respectively. The symmetric curve connecting two arbitrary apexes of the rotor is maximized in the direction of the inner housing shape with the constraint that is not in touch with the housing at any angle of rotation (an arc is not a solution of this optimization problem).

The central drive shaft, called the "eccentric shaft" or "E-shaft", passes through the center of the rotor and is supported by fixed bearings. The rotors ride on eccentrics (analogous to crankpins) integral to the eccentric shaft (analogous to a crankshaft). The rotors both rotate around the eccentrics and make orbital revolutions around the eccentric shaft. Seals at the corners of the rotor seal against the periphery of the housing, dividing it into three moving combustion chambers. The rotation of each rotor on its own axis is caused and controlled by a pair of synchronizing gears. A fixed gear mounted on one side of the rotor housing engages a ring gear attached to the rotor and ensures the rotor moves exactly 1/3 turn for each turn of the eccentric shaft. The power output of the engine is not transmitted through the synchronizing gears. The force of gas pressure on the rotor (to a first approximation) goes directly to the center of the eccentric part of the output shaft.

Wankel engine developed by German engineer Felix Wankel in 1950's and 60's, has less moving parts (at the ratio of 2/3) than classical engines. And it's less complicated compared to the classical engines realizing power transmission by the help of shaft directly connected to the piston. Wankel engines known as getting maximum power in minimum volume conditions, equalized World Speed Record in 1957. The Wankel engine has advantages of compact design, low weight and high revolutions per minute. These advantages have given rotary engine applications in a variety of vehicles and devices, including automobiles (mostly sports cars), motorcycles, racing cars, aircraft, go-karts, jet skis, snowmobiles, chain saws and auxiliary power units. It's still used by the type of RX7 and RX8 engines in Mazda.

In Mazda's RX-8 with the Renesis engine (that is used in my study converted into single rotor type), fuel economy met California State requirements, including California's low emissions vehicle (LEV) standards. This was achieved by a number of innovations. The exhaust ports, which in earlier Mazda rotaries were located in the rotor housings, were moved to the sides of the combustion chamber. This solved the problem of the earlier ash buildup in the engine, and thermal distortion problems of side intake and exhaust ports. A scraper seal was added in the rotor side, and by use of some ceramic-made parts in the engine. This approach allowed Mazda to eliminate overlap between intake and exhaust port openings, while simultaneously improves the endurance of the exhaust area. The side port trapped the unburned fuel in the chamber, decreased the oil consumption, and improved the combustion stability in the low-speed and light load range. The HC emissions from the side exhaust port Wankel engine are 35–50% less than those from the peripheral exhaust port Wankel engines, because of near zero intake and exhaust port opening overlap.

Because of the existing advantages, in this research especially electronical experiments were done with the mechanisms which mechanical engineers Taner Yıldırım and Özgür Tekeli built in their master dissertations. For the timing of injection and ignition also duration of injection and dwell time (that current is flowing through the primary winding and the magnetic field is building up in the coil) variables of the Wankel engine (which converted into single rotor type in terms

of testing standards) computer based electronic unit were designed and produced. Arduino is used as microprocessor control unit and to strengthen the signal of injection and ignition, an electronic control box consisted of driver relays was produced. To change injection and ignition parameters and to get feedback from the system, control program is written on PC with Visual Studio C#. To receive and send data between the system and computer, serial communication codes are written both in Arduino control program and C# interface program. To control of the cycles of engine and to get feedback, an incremental encoder was mounted on crank shaft and the starter signal of encoder has become equivalent to the compression of the top dead center of engine crank shaft. To evaluate the flow rate of injector, a testing mechanism was designed. It's given information about working principle of Wankel engine and referred about injection and ignition equipments and which criterions should be cared when they were chosen. Control system was able to design for adapting the conversions at different load and engine speed. Before connecting to the research engine, control system was tested on the real time testing mechanism which driven by a drill, and it is observed that the control system is working without problem depending upon design criterions.

Also for the tests to be held on the system later on, to store the data and to be used when it is required, an external memory device unit (EEPROM - Electrically Erasable Programmable Read-Only Memory) was added and incorporated into the electronic system. The main goal of storing data is to map ignition advance and injector flow rate due to throttle opening and revolution per minute. It's given detailed informations about how to set external memory up to the Arduino software and the way of write and read data, with regards to a source for the new researchers. To sum up this study, it's given detailed informations about mechanical working principle of internal combustion Wankel engine. An electronic control unit designed and produced to control injection and ignition parameters. To do that, a control program and computer based interface were created. The software and hardware submitted with minimal prices and designed as easier as it can to make for the future researchers easy to understand. A memory device founded for new researches.

1. GİRİŞ

Önemli bir teknolojik buluş olan otomobilin tarihi 19. yüzyılda enerji kaynağı olarak buharın kullanılmasıyla başlar ve içten yanmalı motorlarda petrolün kullanılmasıyla devam eder. Günümüzde alternatif enerji kaynakları ile çalışan otomobillerin imalatı konusunda çalışmalar hız kazanmıştır [14].

19. yüzyıla gelinmesiyle birlikte içten yanmalı motorlar, gelişen mühendislik çalışmaları ile birlikte yaygınlaşmaya başlamıştır. İlk içten yanmalı motor 1859 yılında Etienne Lenoir tarafından geliştirilmiştir. Geliştirilen ilk motorda kömür-gazı ile hava karışımını atmosfer basıncında yakılmıştır. Ateşleme ise bir elektrik kıvılcımı ile sağlanmıştır. Egzoz stroku ilave edilen motor 1860 – 65 arasında yaklaşık %5 verim ve 6 BG güçte 5.000 adet imal edilmiştir. İçten yanmalı pistonlu motorların bugünkü temel yapısı için öneriler ilk defa Nicolaus August Otto tarafından 1867 yılında sunulmuş, 1876 yılında da ilk prototip motor çalıştırılmıştır. 1890'a kadar yaklaşık 50.000 adet motor Amerika ve Avrupa'da satılmıştır. Bu motor, sabit hacimde yanma çevrimi ile çalışan 4-zamanlı benzin motoru olup efektif verimi daha 1894 yıllarında %20-25'e kadar yükseltilmiştir. Bugün ise efektif verim %30 - 37 arasında değişmektedir [15].

Otto çevrimine göre; pistonun silindir içinde üst ölü noktadan alt ölü noktaya doğru hareketiyle (1. Strok) artan hacme emilen yakıt-hava karışımı, pistonun tekrar üst ölü noktaya gelmesiyle (2. Strok) sıkıştırılır, sıkıştırma sonu basıncının en üst seviyeye çıktığı anda elektrik kıvılcımıyla yakılan karışımın sağladığı yüksek basınç, pistonu bu kez iş üreterek, tekrar aşağı iter (3. Strok). Piston üst ölü noktaya ikinci kez gelirken (4. Strok), yanmış olan gazları yanma odasından tahliye ederek çevrimini tamamlar.

1892 yılına gelindiğinde, Alman mühendis Rudolf Diesel, içten yanmalı motor tarihinin ikinci dönüm noktası sayılan, rasyonel termal motor adını verdiği, ancak daha sonraları kendi adıyla anılan Diesel (dizel) motorunun patentini alarak dünya çapında büyük yankı uyandırmıştır. Bu motorun kullandığı, günümüzde de dizel çevrimi olarak bilinen çevrime göre, motora emme strokunda sadece hava emilmekte, sıkıştırma sonunda yanma odasına enjekte edilen yakıtın sıkıştırmadan kaynaklanan yüksek sıcaklıkla kendi kendine tutuşması sağlanmaktadır. Rudolf Diesel'in bu motorda yakıt olarak yer fıstığı yağı kullandığı, ancak daha sonra fuel-oil'in dizel yakıtı olarak kullanıldığı bilinmektedir.

19. yüzyılın ikinci yarısında içten yanmalı motorlar alanında kaydedilen bu önemli gelişmeler otomotiv sektörünün de hızla ilerlemesini sağlamıştır. 20. yüzyılın başından itibaren motorlar konusunda yapılan araştırmalar ve gösterilen çabalar yeni bir strateji geliştirmekten çok, temeli atılmış olan Otto ve Diesel motorlarının otomotiv sektörünün ihtiyaçlarına cevap verecek şekilde geliştirilmesi yönünde olmuştur. Ancak 1954 yılında, Felix Wankel'in geliştirdiği ve kendi adıyla anılan Wankel motoru da, içten yanmalı motorlar tarihinde önemli bir başarı elde etmiştir [1].

1954 yılında Dr.Felix Wankel (Batı Almanya), NSU işbirliği ile dünyada döner motoru (Wankel Motor) başarıyla geliştiren ilk kişi olmuştur. Geçmişteki başarısız denemelerden sonra ortaya çıkan bu buluş, mühendislik alanında ve özellikle otomotiv endüstrisinde gitgide artan yüksek beklentiler oluşturmuştur. Dünyanın her yanından mühendisler tarafından takip edilen yeni döner motor, dirençli ve kuvvetli oluşu sayesinde pratik bir motor haline gelmiştir. Gelişim sürecinde yaşanan sorunlara karşın başarıyla geliştirilmiş Wankel Motoru endüstriyel uygulamalarda özellikle otomotiv endüstrisinde uygulanabilir ve yenilikçi bir motor olarak tanınacaktır [2].

NSU ile birlikte yapılan geliştirme çalışmaları sonucunda, 1954 yılında birçok teknik sorunu çözümlenmiş DKM54 (Almanca: Drehkolbenmotor) motoru ortaya çıkmıştır. Bu motorun kullanıldığı NSU yapımı bir taşıt 1957 yılında 50 cm³ sınıfında dünya hız rekorunu kırmıştır.

1963 yılında ilk defa bu motor, piyasaya sürülecek Audi Spider'e yerleştirilmiştir. 1967'de motoru geliştiren Wankel bu motora çift rotor eklemeyi başarmıştır. 900 cm³ iç hacmi olan motorun 103 kg ağırlığı vardır ve 136 beygirlik bir güç üretmeyi başarmıştır. Bu da o dönemin motorlarına göre geçilemeyecek bir hız sağlamıştır [16]. Daha sonra Wankel motorunun bu performansını gören Japon ve Amerikan firmaları kendi imal ettikleri otomobillerde bu motoru kullanmaya başlamışlardır. 1972de, 50 milyon dolarlık bir yatırımla bu motorun gücünü 185 beygire kadar çıkaran Amerikalılar bu motorun gelişmesinde Audi markası kadar büyük bir pay sahibi olmuşlardır. Bu yatırımdan sonra Rolls-Royce, Mercedes ve Alfa Romeo da Wankel motorunu kullanmışlar, 1970'lerin sonuna doğru ise bu motorun tüm haklarını Mazda'nın almasıyla diğer markaların Wankel motoru dönemi son bulmuştur.

Wankel motorunun normal bir motordan çok daha basit bir yapısı vardır. Oval bir gövde içerisinde merkezden kaçık olarak dönen bir rotor (döner piston) (tasarıma göre 2-3-4 rotor da olabilir) ve eksantrik milidir (eksantrik mili 4 zamanlı motorlarda bulunan krank milinin işini yapmaktadır). Motor, 4 silindirli, 16 supaplı, 2 eksantrikli, günümüz motoruna göre çok daha az karmaşık hareketli parçaya sahiptir. Rotor bir iç ve bir dış dişli yardımı ile motorun ana miline bağlıdır. Çalıştığı sürece emme, sıkıştırma, iş ve egzoz zamanları rotorun çevresinde oluşur. Wankel motoru dört zaman ilkesine göre çalışmaktadır. Rotorun her bir tam devrinde her haznede dört zamanlı motora göre bir iş meydana gelir. Eksantrik mili bu sırada üç devir dönmüş olur. Motorun kumandası muhafaza gövdesindeki kanal üzerinden sağlanır. Rotor eksantrik milinin bir kamına yataklanmıştır. Muhafaza gövdesinin içinde sabitlenen pinyon dişli rotorun iç tarafına açılmış dişli ile temasdadır. Rotor sabit pinyon dişli üzerinde yuvarlanır. Yuvarlanırken eksantrik mili üzerinde bir döndürme kuvveti oluşturur. Eksantrik milinde oluşan bu dönme hareketi dişli kutusuna iletilir. Su ile soğutulan motor gövdesinin bir yanında emme ve egzoz kanalları karşı tarafında ateşleme bujileri yer alır [16]. Wankel motorları hız, moment, performans gibi değerler yönünden, diğer içten yanmalı motorlarla karşılaştırıldığında birçok üstünlüğe sahiptir. Bu motorların ağırlıkları daha az ve doğrudan dairesel hareket üretilmektedir. Bu sebeple motor milinden az yakıtla daha çok güç alınabilmektedir [3].

Mazda, Wankel motorunu, Mazda 110 S Cosmo Sport (1967–1972), Mazda RX-2 (1971–1974), Mazda R100 (1968–1975), Mazda RX-5 (1975–1981), Mazda RX-3 (1972–1977), Mazda 787B (1991), Mazda RX-7 (1978–2002) modellerinde kullanmıştır. Şu anda Wankel motorunu Mazda'nın RX-8 modelinde kullanmaktadır [17].

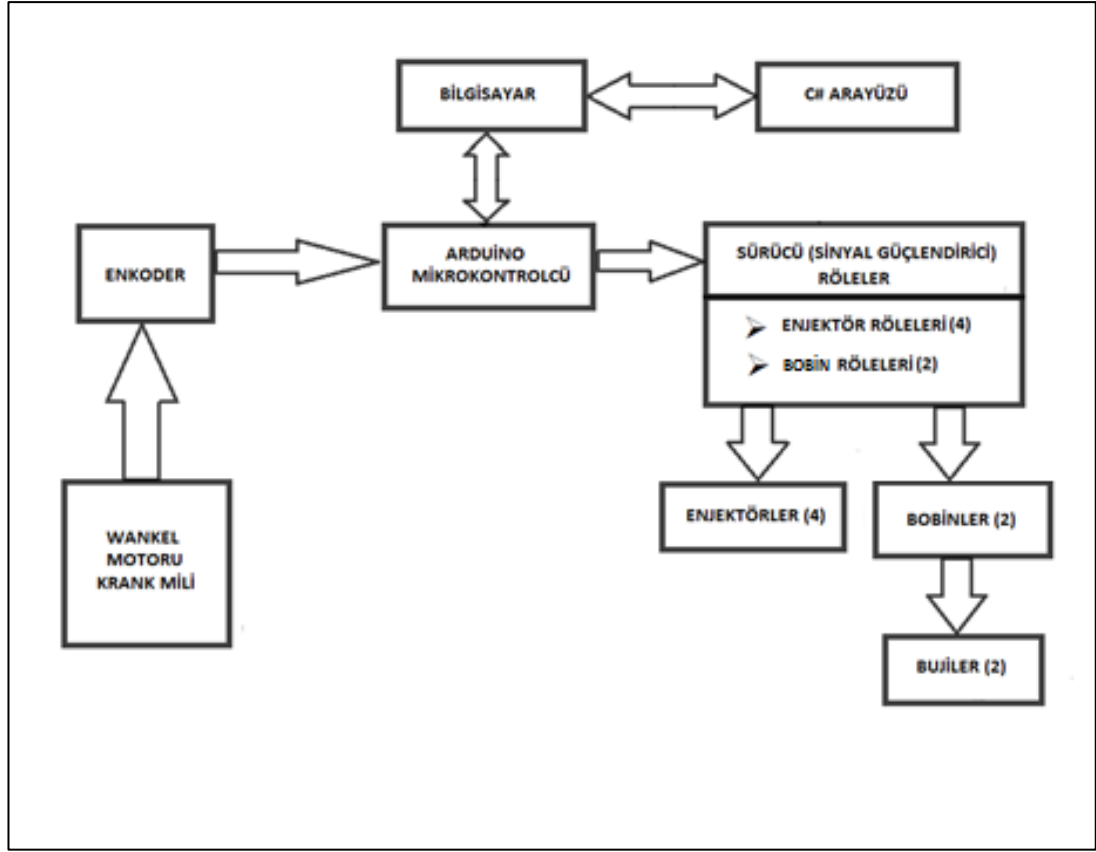
1.1 Tezin Amacı

Motor araştırma ve geliştirmesi çoğu zaman pahalı, zaman gerektiren, karmaşık bir iştir. Bu alandaki deneysel araştırmalar bir çok farklı konstrüksiyon ve ölçüm içermektedir. Araştırmaların en önemli ve gerekli parçası ise deney motorlarıdır. Deney masraflarını azaltması, geliştirme zamanlarını en aza indirmesi, esnek olması ve parçaların daha kolay adapte edilebilmesi için, araştırma motoru olarak genellikle tek silindirli deney motorları kullanılmaktadır [4].

Bu çalışmada, deney standardı oluşturmak adına tek rotorlu hale dönüştürülen bir wankel motorunun ateşleme ve püskürtme süreleri, bir mikrokontrolcü ile bilgisayar arayüzü üzerinden kontrol edilmiştir. Bu sayede yeni yanma odası geometrileri, farklı ateşleme ve püskürtme şartları için enjektör ve bujilerin her birinin birbirinden bağımsız kontrolü, optimum ateşleme ve püskürtme şartlarının oluşturulması sağlanmaya çalışılarak doğruluğu yüksek sonuçlar elde edilmesi hedeflenmiştir.

Ateşleme ve püskürtme sistemindeki donanımların seçim kriterlerinden bahsedilmiş, performansı yüksek ve ileride yaşanabilecek arızaların kolay giderilerilmelerine olanak tanıyan elektronik ekipmanlar seçilmiş, yine yaşanabilecek olası sorunların kolay anlaşılması, çözümlenmesi ve sistemin geliştirilebilmesi için kullanıcı dostu bir arayüz tasarlanmıştır. Sistemin farklı enkoder çözünürlüklerinde, farklı motor tipleri, enjektör ve bujiler ile çalışacak şekilde uyarlanabilmesinin de mümkün olduğunca kolay olması hedeflenmiştir.

Sistemin işleyişi Şekil 1.1'deki gibi olacaktır:



Şekil 1.1: İşleyiş şeması

1.2 Araştırma Motorunun Tanıtılması

Wankel motorları dört zamanlı çalışma prensibine sahip döner pistonlu motorlardır. Bu motorlar çalışma prensibi bakımından lineer pistonlu motorlara göre farklılık gösterir. Motor üzerinde hareketli parça olarak rotor ve eksantrik mil bulunmaktadır. Emme ve egzoz zamanlarında lineer pistonlu motorlar gibi subap tertibatı bulunmamaktadır. Emme ve egzoz süreci, gövde veya yan gövdelerde bulunan portlar sayesinde gerçekleştirilir. Portların açılma ve kapanma zamanları ise rotorun hareketi ile yan yüzü sınırlamaktadır. Portlar aracılığı ile dolgu değişimi sağlanmaktadır. Aynı zamanda portun geometrisi, avansı belirlemektedir. Ayrıca lineer pistonlu motorlar gibi krank biyel mekanizması olmadığı için daha basit bir yapıya sahiptir.

Wankel motorunda rotor, doğrusal hareketli pistonlu motorlardaki piston ve biyel mekanizmasının görevini üstlenmekte ve yanan gazların oluşturduğu basınç sayesinde dönme hareketini sağlayarak güç iletimini çıkış miline doğrudan iletmektedir. Wankel motorlarında krank biyel mekanizması ve supap tertibatı

bulunmadığı için pistonlu motorlara göre daha yüksek devirlere çıkabilme imkânı tanımaktadır. Wankel motorlarında rotorun bir turunda ekzantrik mili üç tur dönmektedir. Burada milin her turunda bir iş elde edilmektedir. Motor çalışma esnasında krank milinin sadece yukarı aşağıya hareketi (180°'lik) yerine, eksantrik mili 270°'lik dönme hareketi yaparak daha çok güç üretimi sağlanmaktadır. Diğer taraftan, daha az parçadan oluştuğu için hafif olması, daha az yer kaplaması, piston ve biyel gibi karşılıklı hareket eden kütleler olmadığından dolayı yumuşak bir şekilde çalışması, NOx emisyon değerlerinin daha düşük olması ve aynı ağırlık oranına sahip pistonlu motora göre daha fazla güç üretebilmesi wankel motorlarının avantajları arasındadır [5].

Gerçekleştirilen deneysel çalışmaların en önemli parçası olan araştırma motorunun seçiminde, aşağıda belirtilen kriterler göz önüne alınmıştır.

- Deney standardı oluşturmak ve parçaları kolay adapte etmek adına tek rotorlu bir Wankel motoru olmalıdır.
- Motor sıcaklığını hassas bir şekilde kontrol edebilmek için soğutma su ile gerçekleştirilmelidir.
- İndike özellikleri ön plana çıkarabilmek için, yardımcı üniteler (alternatör, su pompası, yağ pompası vs.) motor tarafından tahrik edilmemelidir.
- Yakıt - hava karışımının hassas ayarı için, karışım püskürtme ile sağlanmalıdır.
- Motor üzerinde kolay değişiklik yapılabilmelidir.
- Ölçüm elemanları için yeterli ve uygun yer olmalıdır [4].

Yukarıdaki kriterlerin ışığında tek rotorlu hale çevrilecek olan deney motorunun çalışma prensibi, Mazda'nın RX8 için geliştirdiği en yeni motor olan RENESİS RX-8 13B' den uyarlanmıştır. Gerek parça temini, gerekse son yıllarda araştırma potansiyelinin bu motor tipine doğru ilerlemesi nedeniyle bu motor üzerinde çalışılmasına karar verilmiştir.

RENESİS döner pistonlu motorlarda MİLENYUM anlamına gelir. Çizelge 1.1'de motorun teknik özelliklerinden bahsedilmektedir.



Şekil 1.2: Çift rotorlu RENESİS motoru

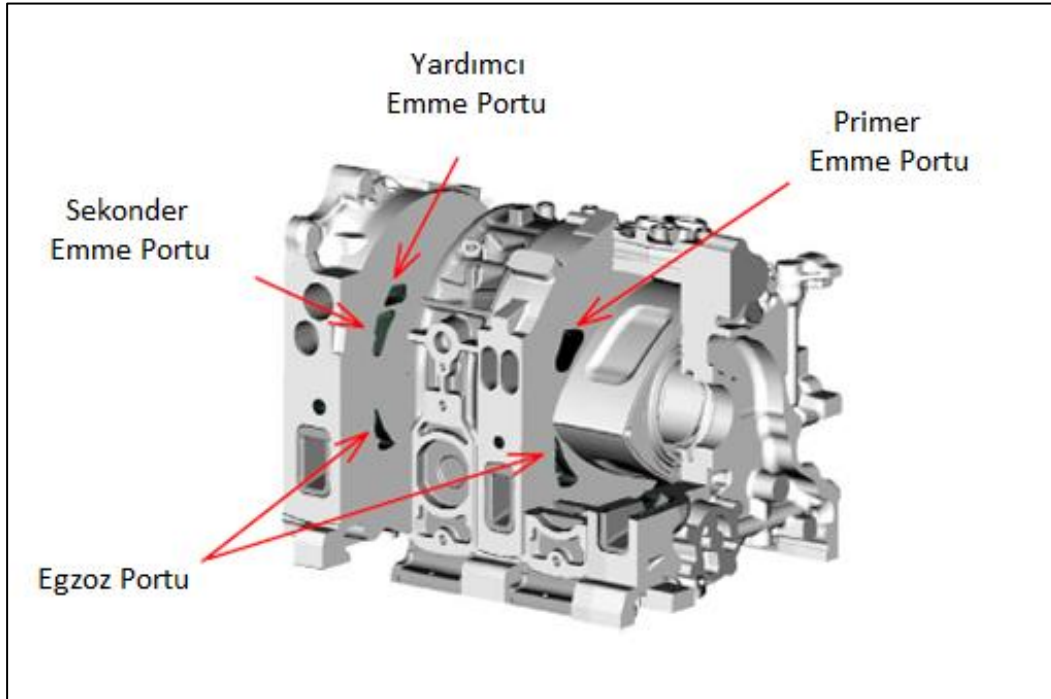
Çizelge 1.1: RENESİS RX-8 13B motorunun teknik özellikleri.

RENESİS RX-8 13B	
Piston köşe-merkez uzunluğu	R = 105 mm
Eksantriklik	e = 15 mm
Gövdenin genişliği	b = 80 mm
Piston boyu	L = 180 mm
Sıkıştırma oranı	$\varepsilon = 10$
Silindir Hacmi	654x1 cc
Emme Portu açılma zamanı Primary(Primer)	3° (ÜÖN' dan sonra)
Emme Portu kapanma zamanı Primary(Primer)	65° (AÖN' dan sonra)
Emme Portu açılma zamanı Secondary(Sekonder)	12° (ÜÖN' dan sonra)
Emme Portu kapanma zamanı Secondary(Sekonder)	36° (AÖN' dan sonra)
Emme Portu açılma zamanı Auxiliary(Yardımcı)	38° (ÜÖN' dan sonra)
Emme Portu kapanma zamanı Auxiliary(Yardımcı)	80° (AÖN' dan sonra)
Egzos Portu açılma zamanı	50° (AÖN' dan önce)
Egzos Portu kapanma zamanı	3° (ÜÖN' dan önce)

RENESİS motorunun High Power (Yüksek Güç) ve Standart Power (Standart Güç) olmak üzere iki ayrı versiyonu vardır. Her ikisinin de imalat karakteristikleri ve motor hızları farklıdır.

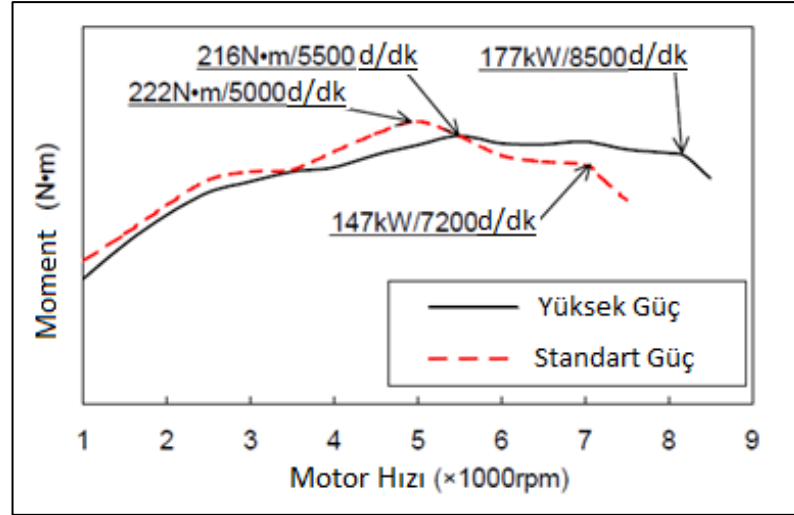
Standart Power (Standart Güç) RENESİS motorda her rotor için Primary (Primer) ve Secondary (Sekonder) olmak üzere iki adet emme portu vardır.

Tezdeki çalışmalarımızda kullanılan High Power (Yüksek Güç) motorda ise, bir rotor için Primary (Primer), Secondary (Sekonder) ve Auxiliary (Yardımcı) olmak üzere üç ayrı emme portu bulunmaktadır. Her üç portun da açılma ve kapanma zamanları farklıdır. Standart iki rotorlu Wankel motorlarında ise toplamda altı adet emme portu bulunmaktadır. Şekil 1.3 iki rotorlu High Power motorda portların yerleşimini göstermektedir [6].



Şekil 1.3: İki rotorlu High Power motorda portlar

Motor performanslarına bakılacak olursa, High Power (Yüksek Güç) RENESİS motoru 8500d/d'da 177KW, 5500d/d'da 216 N.m güç üretirken, Standart Power (Standart Güç) RENESİS motoru 7200d/d'da 147KW, 5000d/d'da 222 N.m güç üretmektedir. Moment eğrileri Şekil 1.4 'teki gibidir.



Şekil 1.4: Motor moment eğrileri

Tek rotorlu Wankel motorunun imalatı öncesinde doktora öğrencisi Yük. Müh. Ömer Cihan'ın doktora tezi çalışmaları kapsamında motorun prototipi hazırlanmış ve çalışabilirliği test edilmiştir. Prototip Wankel motorunun hazır hale getirilmesi sürecinde yağlama sisteminde yağ filtre kütüğü imal edilerek yağlama sistemi rekor bağlantısıyla pompa ile yağ kütüğü arasında yağın akışı sağlanmıştır. Daha sonra üretilen mil üzerinde, motorun montajı sağlanabilmesi açısından yeniden boyutlandırma çalışmalar yapılmıştır. Tüm eksiklikleri giderilen prototip motor için bir tabla yapılarak üzerine sabitlenmiştir. Hazır hale getirilen prototip motor üzerinde kompresyon basıncı, sürtünme basıncı ve ve gücü gibi parametrelerin ölçümü yapılmıştır. Motoru tahrik edebilmek için kayış kasnak bağlantısı ile farklı iki elektrik motoru kullanılmıştır. Elektrik motorları üzerine farklı çaplarda kasnaklar monte edilmiştir. Buradaki amaç farklı çap ile motor devrini değiştirebilmektir. Farklı kasnak çaplarında motorda mile devir takometresi tutularak devir ölçülmüştür. Daha sonra bujinin biri çıkartılıp yerine manometre bağlanarak kompresyon basıncına bakılmıştır. Sonuç olarak yapılan prototip motorun düzgün bir şekilde çalışması, motorun kompresyon yapabiliyor olması, sürtünme değerlerinin makul bir seviyede oluşu ile motorun tek rotorlu hale dönüşebileceği kanaatine varılmıştır.

Yapılan bir diğer çalışma ise eksantrik milin dengeleme hesabıdır. Daha önce yapılan bu hesaplama, tek rotorlu motorun eksantrik mili imal edilirkenki rotor muylusunun eni dikkate alınarak tekrar hesaplanmıştır. Orijinal eksantrik milinde rotor muylusunun devamında, rotorun deliklerine yağ püskürtmesi için yağ jeti bulunmaktadır. Bu yağ jeti mil tek rotorlu motora göre imal edildiğinde

orijinalindeki gibi kalırsa motorun montajında hata verecektir ve arka yan gövde gövdeye oturmayacaktır. Bu yüzden rotorun yağ jeti kısmı torna ile alınmıştır. Yağ jetinin yeri mil üzerinde arka yatak ile rotor muylusu arasına konulmuştur. Dolayısı ile rotor muylusunun en ölçüsü değiştiği için, motorun önündeki ve volan üzerindeki dengeleme ağırlığına tek rotorlu motor için ne kadar kütle ekleneceği tekrar hesaplanmıştır. Dengeleme hesabı motorun düzgün bir şekilde çalışması için oldukça önemlidir. Daha sonra milin mukavemet hesapları yapılmıştır.



Şekil 1.5 : Tek rotorlu Wankel motorunun prototipi

Gelinen aşamada motor imalat aşamaları halen devam etmektedir. Emme manifold kanalları yapılacak, üzerlerine küresel vanalar eklenecek ve enjektörler yerleştirilecektir. Son olarak motorun geri kalan kısmının montajı ve kurulumu yapıp deney aşamasına hazır hale getirilecektir.

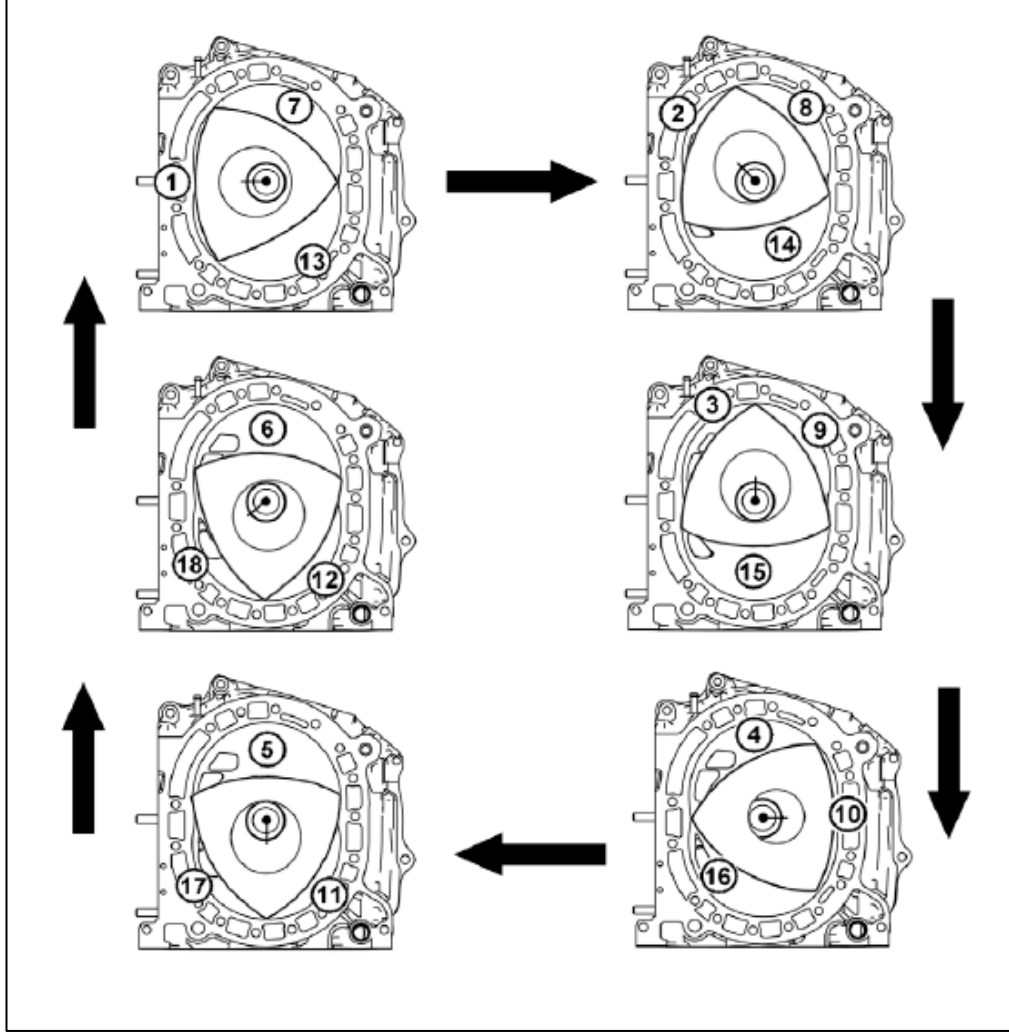
1.3 Wankel Motorunda Zamanlar

Pistonlu motorlarda bulunan alt ölü nokta ile üst ölü nokta arasında pistonun kat ettiği yola strok denilmektedir. Ancak bu motorlarda piston olmadığı için bu motorlara 4 zamanlı (stroklı) motor yerine, 4 fazlı motor da denilmektedir.

Wankel motorlarında zamanlar:

1. zaman: Emme,
2. zaman: Kompresyon (sıkıştırma),
3. zaman: Ateşleme (iş),
4. zaman: Egzoz

olarak isimlendirilmektedir. Bu motorlarda döner piston, motor bloğu içinde yaptığı bir dönme hareketi ile dört zamanı tamamlar. Fakat döner piston üzerinde 120°'lik açı farkı ile üç ateşleme yüzeyi vardır. Yani döner piston, bir devrini tamamladığında emme, sıkıştırma, ateşleme zamanlarını yapar. Böylece bu çalışma prensibi ile az hacimde, çok güç elde edebilmektedir [3].



Şekil 1.6: Wankel motorunda zamanlar [7].

- Hava Emme Stroğu: 1 numaralı konumda çalışma odasının hacmi en küçüktür, böylece pistonlu motorun emme üst ölü noktasına karşılık gelir. Rotor 2 numaralı konumdan 4 numaraya doğru dönmeye devam ettikçe portlar gitgide büyürler ve emme stroku başlarken hava/yakıt karışımını içeri çekerler. Aynı zamanda emme odası hacmini genişleterek 5 numaralı konumda maksimum seviyesine ulaşır. Bu anda rotor konumu pistonlu motorun alt ölü noktasına karşılık gelir ve bunun için emme alt ölü noktası denir.

- Sıkıştırma Stroğu: 5 numaralı konumdan sonra emme portları rotor tarafından kaplanarak emme stroğu biter. Çalışan odanın kapasitesi derece derece azalarak kompresyon stroğu başlarken hava/yakıt karışımını sıkıştırır. Rotor dönerken, çalışan odanın kapasitesi 6 numaralı konumdan 9 numaraya kadar daha da küçülür ve 10 numaralı konumda kompresyon hemen hemen tamamlanmışken karışım bujiler tarafından tutuşturulur. 10 numaralı konumda çalışan odanın kapasitesi minimum seviyesindedir. Bu kompresyon stroğu nun üst ölü noktasıdır.
- Yanma Stroğu: 10 numaralı konumda tutuşturulan karışım yanar, basınç ve sıcaklık artar ve çalışan odadaki genleşme 11 numaradan 13'e doğru devam eder. Bu süre zarfında karışımın yanma enerjisi rotorun yüzeyine uygulanır ve eksantrik mili aracılığıyla, dönel enerjiye çevrilir ve itici güç olarak kullanılır 15 numarada odanın kapasitesi yanma stroğunun alt ölü noktasına ulaşırken maksimuma çıkar.
- Egzoz Stroğu: Yanmanın tamamlanmasından sonra çalışma odasının kapasitesi 15 numaradan 18 numaraya azalırken yanmış gazlar egzoz portlarından dışarı atılırlar. Bu egzoz stroğudur ve bu strok bittiğinde döngüyü yeniden başlatacak olan 1 numaraya bir geri dönüş vardır [7].

2. ATEŞLEME VE PÜSKÜRTME ELEMANLARININ TANITIMI

2.1 Yakıt Enjeksiyon Sistemi

Yakıt sistemi, motorun çalışması için gereken yakıtı enjektörlere yaklaşık 3 bar basıncında sağlar. Sistem aşağıdaki elemanlardan oluşur;

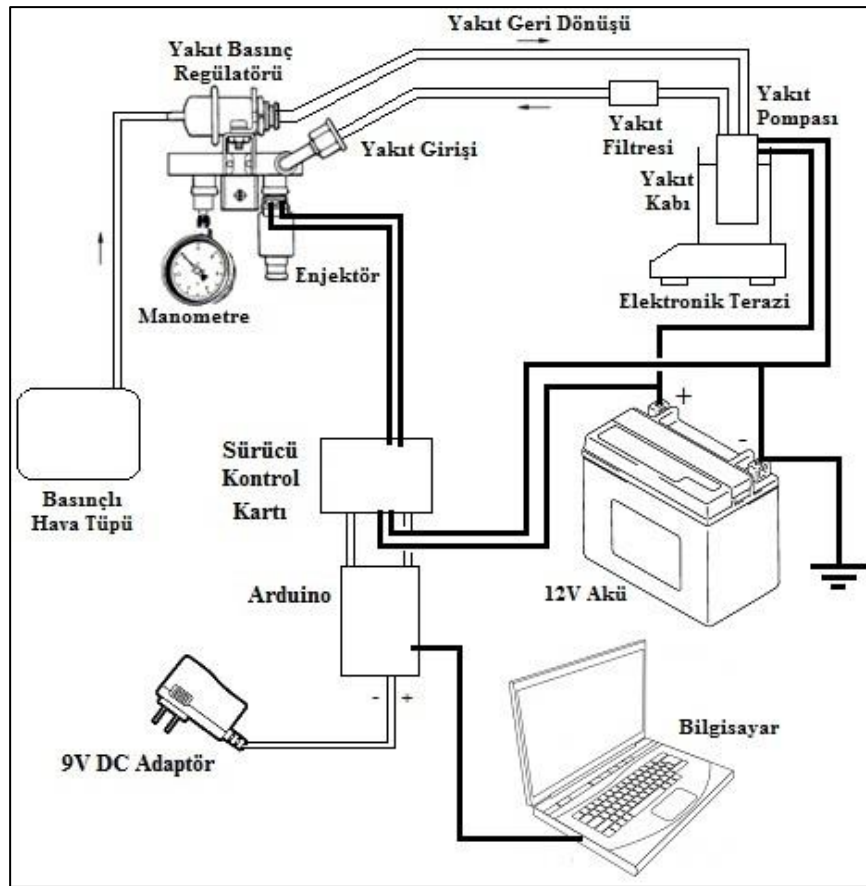
- Deponun içindeki yakıt pompası
- Yakıt filtresi
- Basınçlı yakıt geliş ve dönüş hortumları
- Manifold
- Enjektör
- Yakıt kütüğü ve basınç regülatörü.

2.1.1 Elektrikli yakıt pompası

Elektrikli yakıt pompası benzin deposundan enjektör kütüğüne yakıt pompalamaktadır. Deneylerde ilk etapta depodan çekebilen Bosch marka bir dış pompa kullanılmıştır, kapasitesi yaklaşık olarak 90 lt/sa'dır. Ölçüm alınacak deneylerde ise yakıt miktarını gram cinsinden ölçerek Test Yazılımı programına aktaran AVL firmasının yakıt tartısı ve pompası kullanılacaktır.

2.1.2 Yakıt filtresi

Kâğıttan yapılan güçlü yakıt filtresi, çok hassas toleranslarla çalışan enjektörler için oldukça önemlidir. Normal kullanımda yaklaşık olarak 30.000 km'lik bir ömre sahiptir.



2.1.3 Enjektör

Enjektörler yakıt püskürtme sisteminin en önemli elemanlarıdır. Çok düşük toleranslarla işlendiklerinden çok hassastırlar. Çoğu benzin püskürtmeli sistemlerde enjektörler 3 bar civarında çalışır. Enjektörün açma kapama kontrolü tasarlanırken enjektörün bir solenoid valf gibi çalıştığı düşünülebilir. Gerilim gelince solenoid içindeki yaya direnç göstererek enjektörü açar ve yakıt püskürtülür. Gerilim kesildiğinde enjektör kapalıdır.

Çalışmada Mazda RX-7/RX-8, Mitsubishi, Mazda, Toyota, Nissan'ın çeşitli modellerinde ve yüksek performanslı otomobillerde kullanılan Denso 026M450X12 enjektörleri kullanılmıştır. Enjektörlerin direnci 13,5 Ω , nominal çalışma gerilimleri 13,5 Volt'dur. Minimum 2,5 bar , maximum 7 bar basıncında çalışabilmektedirler.

$$i = \frac{13,5}{13,5} = 1 A \quad (2.1)$$

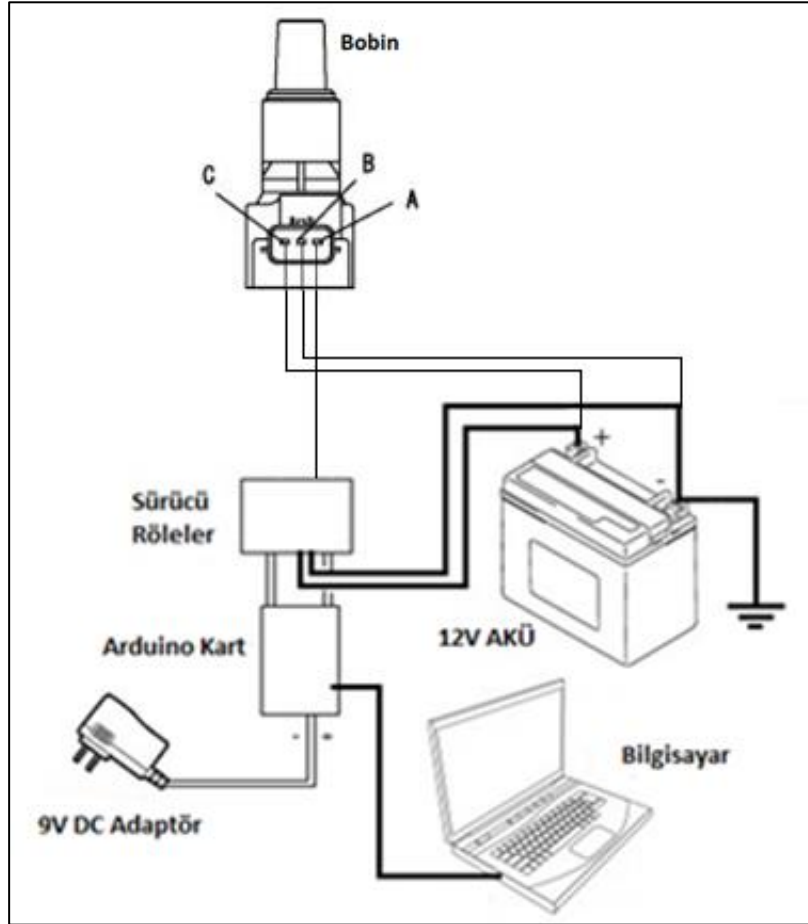
Enjektör tam açıkken 1 amper akım çekmektedir. Enjektör sürücü devresi tasarlanırken çektiği akıma dikkat edilerek kullanılacak anahtarlama elemanı seçilmiştir.



Şekil 2.3: Mazda RX8 Denso 026450x12 enjektör

2.2 Ateşleme Sistemi

Bir Otto veya Wankel motorunun çalışabilmesi için tutuşabilir karışım, sıkıştırma ve tutuşabilir enerji seviyesi gereklidir. İçten yanmalı benzinli motorlarda ateşleme sistemi, silindir içine alınan hava yakıt karışımının patlaması için gerekli ilk enerjiyi sağlayan sistemdir. Ateşleme sistemi bunun yanında motorun değişen devir ve yük koşullarına göre de kıvılcım çakma zamanını yani avans ayarını ayarlamakla da görevlidir. Motor krank milinin 2000 d/dk'da döndüğü düşünülüğünde, Wankel motorunda rotorun her turunda bir ateşleme yapıldığına göre, iki ateşleme zamanı arasında 30 ms gibi çok kısa bir süre mevcuttur. Ateşleme sisteminin düzgün çalışabilmesi için sistemdeki elemanlarla tamamen uyumlu olması gerekmektedir.



Şekil 2.4: Ateşleme işleyiş şeması.

2.2.1 Ateşleme voltajı ve enerjisi

Stokiyometrik orandaki karışımı elektrik arkı ile ateşlenmek için 0,2 mJ, zengin ve fakir karışımlarda yaklaşık 3 mJ enerji yeterlidir. Fakat güvenle ateşleme yapılabilmesi için uygulamada daha fazla enerji ile ateşleme yapılır. Ateşlemenin güvenle yapılabilmesi için silindirde oluşturulan arkın daha uzun süreli olması gereklidir. Bu da daha büyük buji elektrodu tırnak aralığıyla sağlanır. Motor şartlarında güvenli ateşleme yapılabilmesi için yeterli ateşleme voltajı 25- 30 kV mertebesindedir.

Ateşleme geriliminin hava-yakıt karışımını ateşleyebilmek için, sürekli belli bir seviyede olması gerekmektedir. Ateşleme gerilimi bujinin elektrotları arasındaki gerilimdir. Yüksek gerilim buji elektrotları arasındaki sıkıştırılmış hava ve yakıtın iyonize olmasını sağlar. İletken hale gelen hava sayesinde elektrotlar arası oluşan şerare hava-yakıt karışımını ateşleyen kıvılcım olarak kullanılır [4].

Paschen kuralına göre bir aralıkta meydana gelen boşalma gerilimin fonksiyonu oluşan gaz basıncı ve hava aralığının değerine bağlı olarak non-lineerdir. Yani motorda bu gerilim elektrod aralığına ve hava yakıt karışımının ateşleme anındaki basıncına bağlıdır. Hava yakıt karışımının ateşleme anındaki basıncı da doğal olarak motorun sıkıştırma oranıyla doğrudan ilgilidir [4].

Ateşleme gerilimini etkileyen faktörler aşağıda sıralanmıştır:

- Hava yakıt karışım özelliği (gaz yoğunluğunu etkileyeceğinden)
- Dolgu giriş sıcaklığı
- Motor sıcaklığı
- Ateşleme anında hava-yakıt karışımının basıncı
- Buji elektrodu arasındaki mesafe
- Buji elektrodlarının çapları ve geometrisi

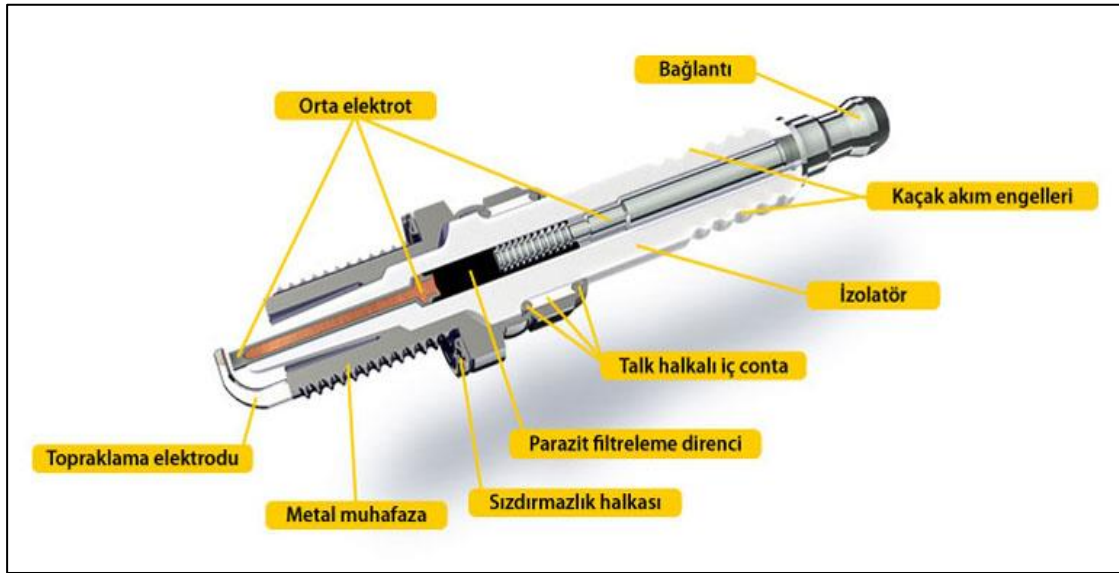
2.2.2 Ateşleme zamanlaması

Stokiyometrik buji ateşlemeli motorlarda piston üst ölü noktaya gelmeden biraz önce ateşleme yapılmaktadır. Bujide kıvılcımın çakması ile ateşlemenin tam anlamıyla gerçekleşmesi arasında yaklaşık 2 milisaniyelik bir süre vardır. Bu süre motor hızı değiştikçe ateşleme başlangıcı farklı krank mili açılarına denk geldiğinden ve kıvılcımın üst ölü nokta yakınlarında olması istendiğinden sürekli olarak ateşleme

zamanı kontrolü gereklidir. Ateşlemenin, piston sıkıştırma ÜÖN'na gelmeden kaç krank mili açısı önce gerçekleştirileceğine ateşleme avansı denir. Motor hızı arttıkça da açısal hız arttığından avans büyümeli, düşük motor hızlarında ise erken ateşleme olmaması için avans değeri düşürülmelidir. Ateşleme avansı için kontrol yöntemleri mekanik ve elektronik olarak geliştirilebilmektedir. Deney motorunda avans kontrolü bilgisayar üzerinden manuel olarak ayarlanacaktır [4].

2.2.3 Buji

Stokiyometrik buji, ateşleme bobininden orta elektroduna gelen yüksek voltajı, sıkıştırılmış hava yakıt karışımında, toprak elektroduna ark yaptırarak çıkan enerjiyle sıkıştırılmış hava yakıt karışımının patlaması için ilk hareketi veren ateşleme sisteminin son parçasıdır.



Şekil 2.5: Buji kesiti [18].

Bağlantı SAE konektörü ya da 4 mm vida dişi şeklindedir. Buraya ateşleme kablosu ya da bir çubuk tipi ateşleme bobini takılır. Her iki durumda da buraya bağlı bir yüksek gerilim bujinin öteki ucuna iletilmek zorundadır. Seramik izolatörün iki görevi vardır. Asıl olarak izolasyon görevi yapar, böylelikle yüksek gerilimin araç topraklamasına atlamasını önler ve yanma sıcaklığını silindir kapağına iletir. İzolatörün dış tarafında dalga biçimli kaçak akım engelleri gerilimlerin araç topraklamasına akmasını önler. Bu engeller kat edilmesi gereken yolu uzatarak

elektrik direncini yükseltir. Böylece enerjinin daha düşük dirençli yolu yani orta elektrottan geçen yolu seçmesi sağlanır [18].

Araç elektroniğinin sorunsuz çalışmasını güvence altına almak için, bujinin içinde parazit filtreleme rezistans olarak bir cam eriyiği tabakası kullanılır. Standart bir bujinin orta elektrodu çoğunlukla bir nikel alaşımından oluşur. Bu elektrodun ucundan kıvılcım toprak elektroduna atlamak zorundadır. Metal gövde bujinin sıcaklığın iletilerek uzaklaştırılmasında da önemli rol oynar. Sızdırmazlık halkası, yüksek yanma basınçlarında bile bujiden yanma gazları çıkmasını önler. Böylece basınç kayıplarının önüne geçer. Ayrıca silindir kapağına ısı iletir ve silindir kapağının ve buji gövdesinin genleşme davranışlarındaki farkları dengeler. İç contalar izolatör ile metal gövde arasında gaz geçirmez bir bağlantı oluşturur.

Bunun için bir talk halkası iki başka sızdırmazlık halkasının arasına gömülüdür, bu halka bujinin imalatı sırasında parçalanır ve böylece en iyi sızdırmazlık düzeyini sağlar. Standart bir bujinin topraklama elektrodu nikel alaşımından yapılmıştır. Normal çalışmada bu elektrot orta elektrodun karşı kutubunu oluşturur [18].

2.2.3.1 Bujiden beklenen özellikler

- Elektriksel performans
- Mekanik performans
- Isıl performans

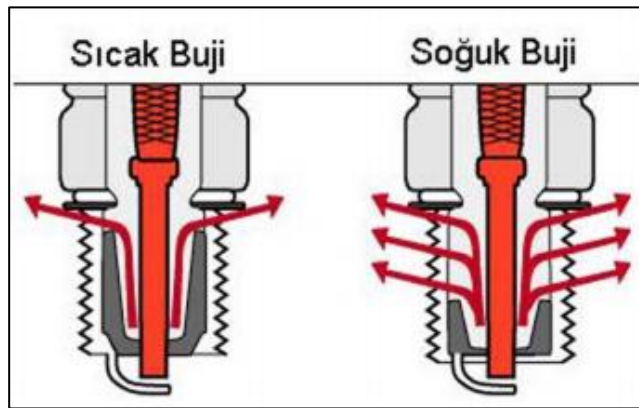
Benzin motorunun doğru ve verimli çalışabilmesi için, buji kıvılcımı mutlak surette güvenilir bir şekilde ateşlenmek zorundadır. Örneğin bir ateşleme atlaması, bütün bir silindir dolusu yakıt-hava karışımının yanmadan egzoz sistemine ulaşmasına ve ancak orada, katalizatörün içinde yanmasına yol açabilir. Sadece az sayıdaki birkaç ateşleme atlaması bile, katalizatöre hasar vermeye ve hatta katalizatörü tamamen kullanılmaz hale getirmeye yetebilir. Ayrıca, kıvılcım tam doğru anda ateşlenmek zorundadır. İdeal ateşleme noktası, pistonun en yüksek noktasına ulaşmasından hemen önce ve sıkıştırmanın en fazla olduğu andır. Fazla erken ya da fazla geç bir ateşleme noktası, motorun verimliliğini olumsuz etkiler ve sonuçta yakıt tüketiminin artmasına ve egzoz emisyonlarının yükselmesine neden olur. Ateşleme kıvılcımının kendisi, yanma odasındaki yakıt-hava karışımını doğrudan ateşlemez. Kıvılcımın gaz karışımı içerisinde izlediği yol, sonuçta karışımın patlamasına neden olan, yoğun bir bölgesel ısınma yaratır. Bir bujiyi tasarlarlarken, bu değerli ısının çok çabuk kaçmasını

önlemek önemlidir. Buji aynı zamanda, çalışma koşullarını kontrol altında tutabilmek için ısıнын silindir kapağına iletilmesinde de önemli bir rol oynar [22].

Elektriksel performans olarak incelendiğinde buji ateşleme işlemi esnasında, herhangi bir topraklama imkanı olmadan direkt olarak, yaklaşık 30 kV'luk bir gerilime maruz kalır. Bujinin bu yüksek gerilimler altında iletkenliğini koruması gerekmektedir [4].

Mekanik olarak incelendiğinde yanma odası içerisinde 100 bara kadar çıkan basınçlar yanma odasında bulunan bujilere etkimektedir. Bir bujinin dinamik olarak etkileyen bu yüke karşı gelebilecek mukavemete sahip olması beklenir. Ayrıca mekanik gerilmelere karşı özellikle kırılğan yapıyla bilinen seramik materyalin dayanıklı olması beklenir. Buji, yanma odasının atmosfere açılan tek açıklığı olduğundan bağlantı esnasında, ön gerilme amacı ile yüksek moment değerleriyle sıkılırlar. Buji bu baskıya herhangi bir fiziksel deformasyon göstermeden dayanabilmelidir [4].

Isıl olarak incelendiğinde yanma odasının içerisinde yanma sonucu oluşan sıcak gazlar egsozla atılırken içeriye daha soğuk karışım alınmakta ve bu süreç çok hızlı bir şekilde işlemektedir. Bu nedenle bujinin termal şoklara karşı özel dayanımı olması beklenir. Ayrıca yanma sonucu sıcaklıkların anlık da olsa 2000-3000⁰C derecelere çıktığı bir yanma odasında bulunan bujinin, erimemesi deformasyona uğramaması ve bunu tekrarlanan süreçlerde istikrarlı bir şekilde ve uzun ömürle başarabilmesi için iyi bir iletim hattı ile sürekli soğuk kalması sağlanmalıdır. Fakat yapılan çalışmalarda sıcaklığı yüksek olan bujilerin ateşleme hızının ve veriminin daha iyi olduğu görülmüştür. Bu yüzden detonasyon yaptırmayan en sıcak buji seçilmelidir [4].



Şekil 2.6: Sıcak ve soğuk buji kesitleri [8].

2.2.3.2 Buji seçimi

Bujiden beklenen özellikler ve deney motorunun gereksinimleri değerlendirildiğinde, Mazda RX8 RENESIS motorlarında halihazırda kullanılmakta olan NGK NY34-18 S30 İridyum bujilerin kullanılmasına karar verilmiştir. Karakteristikleri açısından birbirinden az bir farkla ayrılan bu iki buji, Leading (Ön) ve Trailing (Arka, İzleyen) buji olarak adlandırılırlar ve sırasıyla önce Leading sonra Trailing devreye girerek ateşleme yaparlar. Tırnak aralıkları farklı olduğundan Leading bujinin kıvılcımı Trailing bujinin kıvılcımından daha şiddetli olacaktır.

Çizelge 2.1 : NGK NY34-18 S30 İridyum bujilerin özellikleri

Bujinin vida dış çapı	14mm
Oturma yüzeyi	Düz
Standart Tırnak aralığı	1,15-1,25 mm
Maksimum Tırnak aralığı (Leading)	1,5 mm
Maksimum Tırnak aralığı (Trailing)	1,4 mm
Bujinin vida dış uzunluğu (Leading)	22,5 mm
Bujinin vida dış uzunluğu (Trailing)	20,8 mm
Sıkma Torku	12,8-17,7
Elektrot tipi	J1 tipi
Şasi elektrotu	Platinyum (Pt)
Merkez elektrot	İridyum (Ir)



Şekil 2.7 : NGK iridyum bujiler

İridyum bujiler halihazırda teknolojik olarak var olan en yüksek evsafli çözümü oluşturmaktadır. Bu bujilerin orta elektrotları üzerinde bir iridium alaşımlı ucu vardır. Bu alaşım lazer yardımıyla özel bir yöntemle elektrodun ucuna kaynaklanır. İridyum asil metali dünyanın en sert metallerinden biridir. İridyum ancak 2450°C 'den itibaren erimeye başlar ve kıvılcım erezyonuna karşı çok dirençlidir. Çok sert bir yüzey sağlaması , sürekli kıvılcımlamada ve yüksek ısıda erezyonu azaltması özellikleriyle döner pistonlu motorlarda tercih sebebidir [9]. İridyum kullanımıyla bujinin kullanım ömrü, standart bujilere kıyasla ortalama iki katına çıkarılır. Bu asil metal ayrıca, orta elektrodu 0,6 mm kalınlık ile çok daha ince tasarlamaya imkan verir. Bu tedbir ateşleme gerilimi gereksinimini önemli ölçüde azaltır ve yanma odasındaki alev cephesinin daha iyi yayılmasına katkıda bulunur [21].

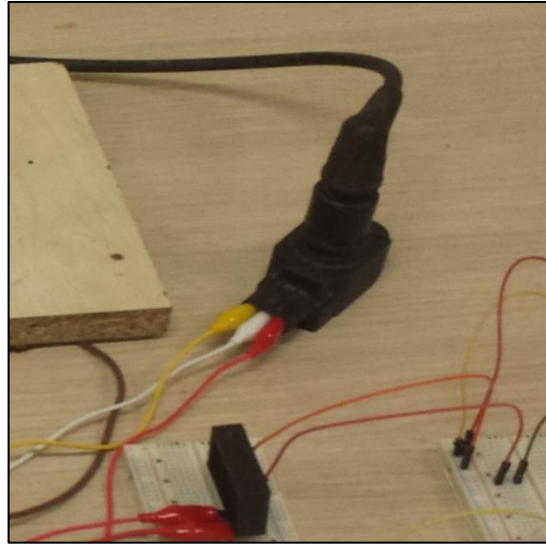
2.2.4 Ateşleme bobini

Ateşleme bobini, diğer adıyla endüksiyon bobini, benzinli motorlarda silindir içinde sıkıştırılan hava-yakıt karışımını ateşlemekte kullanılan bujiye yüksek gerilim sağlayan sistemdir [19]. İç içe iki bobinden oluşmaktadır. Dış tarafta birincil sargı, iç tarafta ikincil sargı vardır. Birincil devre yüksek akım taşıma, ikincil devre ise yüksek gerilim iletme özelliklerine sahiptir. Bu nedenle ortalama değerler olarak ateşleme bobininin birincil (primer) sargısı 0.7-1,0 mm çapında, yüksek gerilim çıkışı sağlayan ikincil (sekonder) sargısı ise çok ince (yaklaşık 0,1 mm çapında) bakır telden sarılır ve her iki bobin manyetik bir çekirdek üzerinde, izolasyon sağlayan trafo yağı ile dolu bir silindir içine yerleştirilerek kapatılır [4]. Kuvvetli bir transformatör etkisi yaratmak için birincil sargının sarım sayısı yaklaşık 200, sekonder sargının ise 20000 civarı olup 1:100 gibi bir orana sahiptirler. Kare, silindir, kalem gibi farklı tiplerdeki tüm bobinlerin içeriği aynıdır.

Ateşleme bobininin birincil sargılarını kontrol eden kesici kontaklar kapalı olduğunda birincil sargıdan akım geçer ve birincil sargının etrafında manyetik alan oluşur. Sargılar iç içe olduğundan ikincil sargıda da bir gerilim indüklenir, fakat bu gerilim bujide ark yaptıracak mertebede değildir. Kesici kontaklar birincil devreyi açtığı anda manyetik alan değişimi maksimum düzeyde olduğundan ikincil devrede indüklenen yüksek gerilim bujide kıvılcım oluşturur [4].

Güvenilir bir ateşleme sağlamak için ateşleme bobinlerinin, devir sayısına göre dakikada 100-1000 defa, yüksek gerilim çıkışından bir yüksek gerilim darbesini bujiye göndermesi gerekir. Bir ateşleme bobininin konstrüksiyonu ve kullanılan malzemeler, motor bölmesinde sıcağa ve neme karşı koymak ve değişken hava koşullarının etkisi altında da güvenilir bir şekilde çalışmak zorundadır. Ateşleme bobininin en önemli özellikleri arasında, bobinin yerine otomobil imalatçısı tarafından öngörülen şekilde oturması için mutlak uyma hassasiyetinin yanında, kısa devre ve elektrik darbesi direnci, ısı iletimi için soğutma alanlarının yeterli ölçeklendirilmiş olması ve sıcaklık istikrarı yer alır. Ateşleme bobinlerin ayrıca, sıçrayan sulara ve tuzlu suya karşı dayanımı da yüksek olmalıdır [23].

Deneylerde Mazda RX8 Wankel motorlarında kullanılan UF501 N3H118100 model ateşleme bobinin kullanılmasına karar verilmiştir.

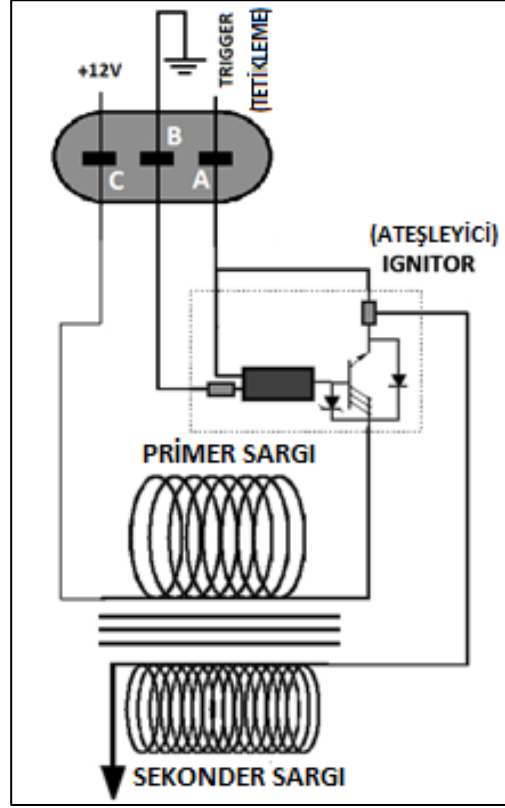


Şekil 2.8: Mazda RX8 ateşleme bobini

Ateşleme bobini, görünüm olarak çubuk tipi ateşleme bobini (Pencil Coil) adıyla da bilinen buji yuvalı ateşleme bobinlerine benzemektedir. Kablo ile veya doğrudan bujinin üzerine takılır. Bu tipte yüksek gerilim darbesi doğrudan bujinin üzerinde üretilir. Hemen hemen hiç güç kaybı olmaması buji yuvalı ateşleme bobinlerinin en önemli avantajlarından. Ayrıca kompakt yapı biçimi sayesinde özellikle modern motorlarda büyük önemi olan yer kazanları elde edilir [24].

Ateşleme bobininin birincil sargısından akım geçme süresi veya bobinin dolma süresi olarak bilinen dwell süresi deneylerde kontrol edilecek değişkenlerden biri

olduğu için önemlidir. UF501 ateşleme bobininin önerilen dwell süresi 5ms'dir. Hazırlanan kontrolcü arayüzünde motorun farklı çalışma aralıklarında optimum dwell süresinin bulunabilmesi için dwell süresi krank mili açısı cinsinden kullanıcının tercihine sunulmuştur. Kontrolcü arayüzü ile ilgili detaylı bilgi Bölüm 4.4'te verilmiştir.

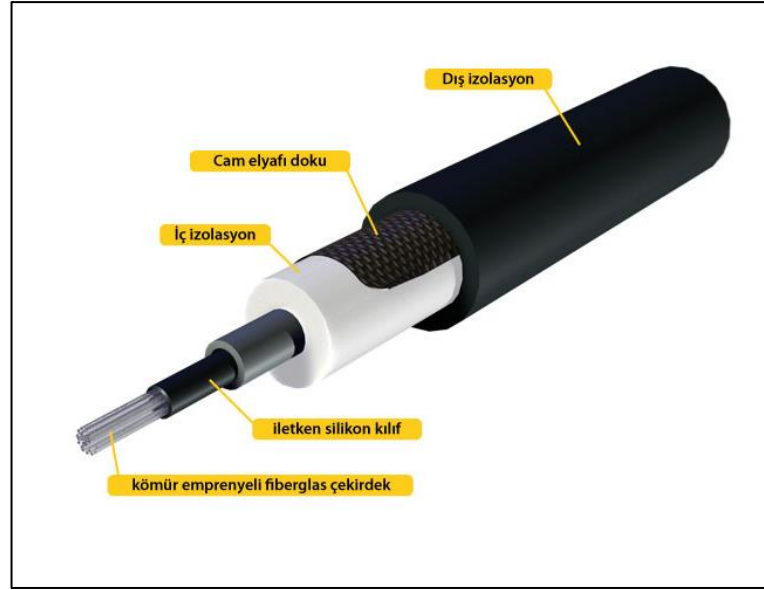


Şekil 2.9: Mazda RX8 ateşleme bobininin çalışma mekanizması

2.2.5 Buji kabloları

Bujide kıvılcımın çıkabilmesi için, ateşleme bobininde üretilen yüksek gerilimin (25.000 Volta kadar) ateşleme kabloları yoluyla bujiye akması gerekmektedir. Bunun güvenilir bir şekilde gerçekleşmesi için, bu kablolarda bir dizi önemli teknik şart aranır. Birinci sınıf izolasyon özellikleri, yüksek sıcaklık dayanımı (200°C'ye kadar), titreşimlere, sıcaklık ve nem oranı değişikliklerine karşı direnç, yüksek evsafli ateşleme kablolarının yerine getirmek zorunda olduğu başlıca önkoşullardır. Ayrıca bunları sürekli ve güvenilir bir şekilde, uzun bir zaman aralığı boyunca, en aşırı koşullar altında yapmaları gerekmektedir.

Deneylerde NGK ZE81 karbon dirençli ateşleme kabloları kullanılmıştır.



Şekil 2.10: Karbon dirençli ateşleme kablolarından bir kesit [25].

Silikondan iç izolasyon, kabloya daha fazla rijitlik kazandırır ve elektrik izolasyonu görevi yapar. Cam elyafı dokusu kabloya daha büyük çekme dayanımı sağlar. Silikon kauçuktan dış izolasyon 220°C'ye kadar sıcaklığa dayanıklıdır ve benzin ve yağa karşı da dayanıklıdır. Karbon dirençli ateşleme kablolarında parazit filtreleme direnci 1 m kablo için 10 k Ω - 23 k Ω arasında değişmektedir [25].

2.2.6 Akü

Akünün görevi marş motorunu, ateşleme sistemini, doğru akımla çalışan tüm devreleri, ışık ve alıcıları beslemektir. Benzinli motorlarda kullanılan 12 voltluk akü, birbirine seri olarak bağlanmış altı adet elemandan meydana gelir. Genellikle her eleman içerisinde, yine birbirlerine seri olarak bağlanmış 4 adet pozitif, 5 adet negatif yüklü plakalar bulunur. Bu plakalar, kurşun-antimuan alaşımı petek üzerine, aktif maddelerin sıvanarak fırınlanmasından oluşur. Pozitif plakalar aktif madde olarak, kurşundioksit içerir. Negatif plakalar aktif madde olarak, saf kurşun içerir. Bu tür plakalar arasına, kısa devreyi önlemek için plakaları izole eden ayırıcılar yerleştirilir. Ayırıcılar, plakalar arasındaki kimyasal tepkimeyi engellemeyecek şekilde çok küçük gözenekleri bulunan plastiklerden yapılır. Akünün içinde Sülfürik asit saf su karışımı olan elektrolit koyulur. Karışımında %39 asit, %61 su vardır. Elemanlar seri köprülerle bağlanır [16].

Deneylerde 12 voltluk, sürekli doluluk oranı yüksek olmasına özen gösterilecek olan standart bir akü kullanılacaktır. Bu şekilde çalışacak bir enerjilendirme sisteminin Wankel motorunda da sorunsuz olarak çalışacağı düşünülmüştür.

2.2.7 Ateşleme modülü

Ateşleme modülü, transistör kontrollü ateşleme sistemlerinde temelde ateşleme bobininin birincil sargısını anahtarlayan, bazı tiplerinde dwell süresi ayarlama, devir saati çıkışı gibi özellikleri de bulunan ünitedir.

Ateşleme sürücüsünde kullanılmak üzere literatürde ve piyasada bulunabilecek taşıtların üzerinde transistörlü ateşleme sistemleri araştırılmıştır. Müh. Özgür Tekeli'nin de tezindeki çalışmalarında kullandığı , TOFAŞ grubu taşıtlarda rastlanan, düzgün soğutulduğunda sorunsuz kullanılabilecek bir ateşleme modülüne rastlanmıştır. Wells DR100 marka bu ateşleme modülü internette yapılan araştırmalarda ilk olarak Amerikan araçlarında, daha sonra Avrupa piyasasındaki araçlarda platinli ateşleme sisteminden transistörlü ateşleme sistemine geçişte kullanılmıştır [4].

GM DR100 yüksek enerjili ateşleme modülü 4 tane bıçak terminal çıkış ve bir soğutucu plakayla birleştirilecek metal yüzeye sahiptir. Soğutucu plaka toprak çıkışı olarak kullanılmaktadır. İki adet vidalama deliği bulunmaktadır ve bu deliklerin iç tarafına ve bağlanacak soğutucu plakaya değen vidalar sayesinde toprak hattı deliklerin iç yüzünden soğutucu plakaya geçmektedir. Çalışmama durumunda öncelikle topraklanmasına dikkat edilmelidir [4].

Bir giriş sinyali verilmediğinde DR100 ateşleme modülü 13V ta 140mA akım çekmektedir. Bu da 1.8 W a tekabül eder. Yukarıda bahsedilen dört adet bıçak terminal çıkışı "B", "C", "G" ve "W" 'dir. B besleme gerilimine bağlanır. Normalde bu yaklaşık 10 ila 15 volt DC olacaktır. C ateşleme bobini bağlantısıdır. Bu ateşleme bobinin birincil sargısının eksi tarafına bağlanır ve ateşleme bobinini eksi kutbunu topraklayarak ateşleme bobinini açık ve kapalı olarak anahtarlar. Kapalı konumdayken, C terminali yaklaşık olarak 600 volt indüktif gerilimi durdurabilir. Bobin açık olduğunda, C gerilim düşümü 1 volt civarındadır [4].



Şekil 2.11: GM DR100 ateşleme modülü

G terminali ateşleme modülü için giriş sinyalıdır. Bu uç hall sensörün pozitif çıkışına bağlanır. Hall sensör çıkış voltajı DR100 ateşleme modülünü kontrol eder. Gerilim belli bir eşik değerin üzerinde olduğunda C terminali topraklanarak ateşleme bobini açık tutar. Giriş gerilim seviyesi eşik değerin altına düştüğünde C terminali açılır ve ateşleme bobini sonra kapalı konuma geçer ve bir kıvılcım oluşur. Bu anda, C terminalinde 100-150 voltluk bir gerilim sıçraması olur. W terminalinin gerilim değeri motor devri ve onunla ilişkili hall sensörün ortasındaki milin devri arttıkça yükselir. İçindeki dwell süresi kapasitesi sayesinde dwell süresini sabit tutmaya çalışır.

Ateşleme modülünün G sinyal ucu toprak hattına bağlıyken ateşleme bobininin (-) kutubuna giden C terminali toprak bağlantısı kesik durumdadır. Bu durumda ateşleme bobininin birincil sargısından akım geçmediğinden bobin dolmamaktadır. Ateşleme modülünün G sinyal ucu toprak hattından ayrıldığı durumda ise ateşleme modülünün C terminali toprak hattına bağlıdır. Bu durumdayken ateşleme bobininin (-) kutubu toprak hattına bağlı olduğundan bu esnada ateşleme bobini dolmaktadır. Ateşleme modülünün G sinyal ucu tekrar toprak hattına bağlandığında ateşleme bobininin (-) kutubu toprak hattı aniden kesilmekte ve bu ani değişim sonucu ateşleme bobinin primer sargısında endüklenen yaklaşık olarak 150 – 200 voltluk gerilim, ateşleme bobininin sarım sayısı 10- 15 kat fazla olan ikincil sargısında 15000- 30000 volt değerlerine ulaşarak sırasıyla sekonder sargı, buji kablosu, buji ve buji orta elektrodundan toprak elektroduna atlayarak harcanmaktadır [4].

2.2.8 Solid state röle

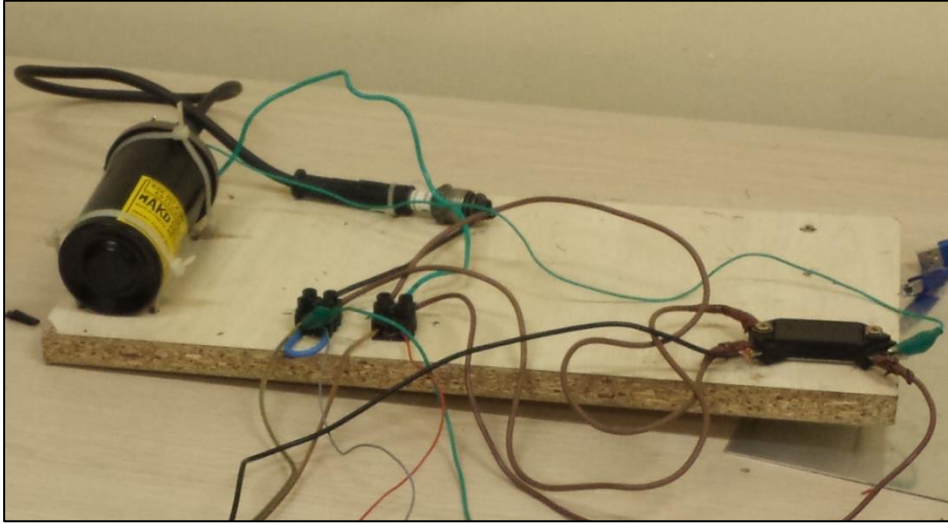
İlk denemelerde Arduino mikrokontrolcü karttan alınan sinyalin ateşleme sistemine ulaştığını gözlemlemek için, GM DR100 ateşleme modülü ve laboratuarda daha önceki çalışmalarda kullanılmış olan Mako BAE207A ateşleme bobini ve AC Delco R4602 buji ile kıvılcım oluşturulması amaçlanmıştır.

Çizelge 2.2: AC Delco R4602 bujinin özellikleri.

Bujinin vida dış çapı	14mm
Oturma yüzeyi	Düz
Tırnak aralığı	0,8 mm
Bujinin vida dış uzunluğu (paso)	17,5 mm=(11/16)"
Direnç	Var
Elektrot tipi	J1 tipi
Şasi elektrotu	Nikel (Ni)
Merkez elektrot	Bakır (Cu)
Yüksek voltaj direnci (HVR)	9 Ω
Düşük voltaj direnci (LVR)	3 Ω

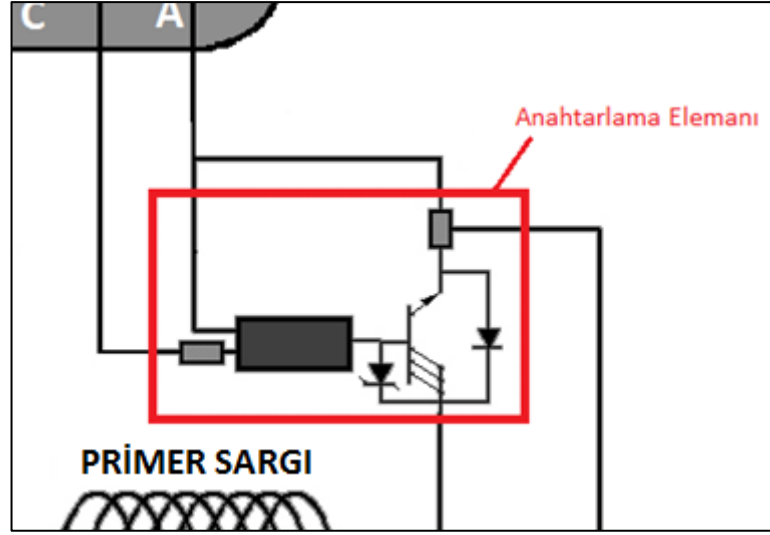
Çizelge 2.3: Mako BAE207A ateşleme bobinin direnç ve endüktans değerleri[A.5].

Birincil direnç ($m\Omega$)	712
İkincil direnç ($k\Omega$)	10,33
Birincil endüktans	3,23 mili Henry
İkincil endüktans	54,4 Henry



Şekil 2.12: Ateşleme bobini ve ateşleme bobini ile kıvılcım oluşturma denemeleri

Denemeler esnasında GM DR100 ateşleme modülünün ısındığı gözlemlenmiştir. Daha önce Müh. Özgür Tekeli'nin tez çalışmalarında da uzun süreli deneylerde soğutucu metal plakaya rağmen aşırı ısındığı bilgisi alınan ateşleme modülünün düzgün çalışabilmesi için ekstra bir soğutucu sisteme ihtiyaç duyduğu kanısına varılmıştır. Ayrıca Wankel motoru deneylerimizde kullanacağımız Mazda RX8 UF501 N3H118100 ateşleme bobininin kendi içinde transistörlü bir anahtarlama sistemi barındırması da GM DR100 ateşleme modülünün gerekliliğini ortadan kaldırmıştır.



Şekil 2.13: Mazda RX8 Bobinin içerisindeki anahtarlama elemanı

Sonuç olarak bobinin, Arduino'dan aldığı 5V'luk sinyali güçlendirerek 12V'luk ateşleme sistemine ileten transistörlü bir anahtarlama elemanı olan Solid-State röleye direkt olarak bağlanması kararına varılmıştır. Solid-State röleden Bölüm 3.1.4'te detaylı olarak bahsedilecektir.

2.3 Arttırımsal Enkoder

Tasarlanan sistemde Wankel motorunun açısal konumu, krank miliyle aynı hızda dönen bir enkoder ile saptanmaktadır. Ateşleme ve püskürtme zamanlamasındaki hassasiyeti belirleyen etken enkoderin çözünürlüğüdür. Bir turun 360° olduğu düşünüldüğünde ihtiyaç olunan enkoderin minimum her derece için 1 pals, toplamda 1 turda 360 pals sayabilen bir enkoder olması yeterlidir. Ancak zamanla değişen deney şartları ve gereksinimleri göz önüne alındığında, ateşleme ve püskürtme hassasiyetinin daha yüksek olması istenen durumlar için, enkoder pals sayısı ne kadar yüksek olursa, ateşleme ve püskürtme sistemlerinin kontrolü bir o kadar hassas, deney sonuçları da verimli olacaktır.

Enkoderler ne kadar kaliteli olsalar da kendi içlerinde hata payları olmaktadır. Binde bir hata payıyla bile çalışılsa 2000 d/d hızda dönen motorda bir dakika sonrasında 1 tur kaçıklığa uğrama ihtimali yüksektir. Bunun nedeni bir turdaki hataların diğer turlara artarak geçmesidir. Toplanarak artan bu hataların önüne geçebilmek için arttırımsal enkoder kullanılmasına karar verilmiştir. Arttırımsal enkoderler normal enkoderlerden farklı olarak her turda aynı konumda olmak üzere bir sinyal daha

üretirler. Z sinyali de denilen bu sinyal geldiğinde enkoderin palslarının sıfırlanarak tekrar saydırılmaya başlaması toplanarak artan bu hataların önüne geçmektedir. [4]

Ateşleme ve püskürtme zamanını hassas olarak kontrol edebilmek için bir turda 3600 pals veren Heidenhain Rod426 enkoderi alınmıştır. Çizelge 2.4'te seçilen enkoderin temel özellikleri belirtilmiştir. Elektrik bağlantıları ve daha fazla detay için Rod426 teknik verilerini içeren Şekil A.3'e bakılabilir.

Çizelge 2.4: Heidenhain Rod426 enkoder özellikleri

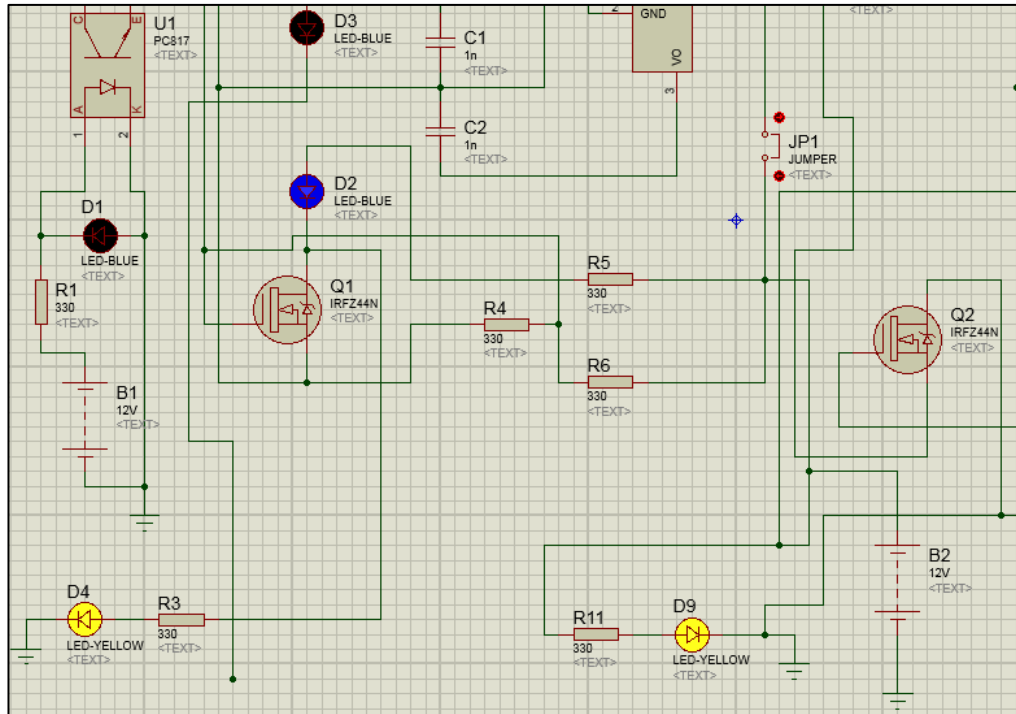
Tur başına sinyal sayısı	3600 (TTL)
Güç kaynağı	5V, 120 mA
Şaft çapı	6 mm
Elektrik bağlantısı	Kablo
Pin düzeni	12 adet pin (M23 konnektörlü)
Şaft bağlantısı	K17 diyafram kaplin ($d = 6/10 \text{ mm}$)

3. KONTROL SİSTEMİNİN DONANIMI

3.1 Sürücü Kartı Tasarım Ve İmalat Çalışmaları

Sürücü kartı tasarım ve imalat aşamasında öncelikle literatürde daha önce yapılan benzer çalışmalar üzerinde araştırma yapılmıştır. Araştırmalar sonucunda Müh. Özgür Tekeli'nin tezinde [4] kullandığı sürücü devre kartı incelenmiş ve çeşitli değişiklikler ile aynı çalışma mantığına sahip yeni bir sürücü kartı tasarlanması amaçlanmıştır.

Bu aşamada öncelikle PROTEUS devre tasarım programında yeni bir sürücü devre tasarlanmış ve simülasyonu yapılmıştır. Güncellenen yeni sürücü kartı imal edilmiş ve çalışabilirliği test edilmiştir.



Şekil 3.1: Proteus devre simülasyonundan bir görüntü

Sürücü kartı enjektör ve bobin sürücülerini içermektedir. Mikrokontrolcuden gelen sinyalleri güçlendirerek bobin ve enjektörü yüksek hızda anahtarlamaktadır. Sürücü kartının diğer bir görevi içinde bulunan ışınsal bağlayıcı (oktokuplör) vasıtasıyla

bujinin orta elektrodundan buji topraklama elektroduna akan yüksek gerilimin; mikrokontrolcü karta, mikrokontrolcü kartın seri haberleşme için bağlı bulunduğu bilgisayara ve enkodere zarar verme olasılığını ortadan kaldırmasıdır. Ayrıca sürücü karta devre elemanlarında meydana gelebilecek olası arızaları tespit edebilmek için led ışıklar yerleştirilerek, bu ledlerin yanıp sönmelerine göre mevcut durumun anlaşılıp sorunların çözülmesi amaçlanmıştır.

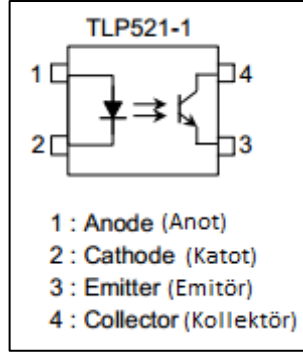
3.1.1 Sürücü kartı üzerindeki elemanlar

- Işınsal bağlayıcı (2 adet)
- 2. 7805 Pozitif lineer voltaj regülatörü
- IRFZ44n mosfet (2 adet)
- 2'li devre klemensi (4 adet)
- 2'li elektrik klemensi (1 adet)
- LED (2 mavi, 1 beyaz, 3 kırmızı)
- 330Ω Direnç (5 adet)
- 100kΩ Direnç (2 adet)
- 10kΩ Direnç (2 adet)
- 1 nF Kondensatör (2 adet)
- Jumper ve soketi (2 adet)
- 1N4007 Diyot
- 4' lü entegre soketi (2 adet)
- Atlatmalar için tek damarlı ince kablo

Dirençler: Devre elemanları üzerine gelen akımı sınırlamak için dirençler kullanılmıştır. Elektronik devreler sabit bir gerilim ile çalışır, ancak devrede kullanılan elemanların hepsi farklı gerilimlere ihtiyaç duyarlar. Tüm devre içinde her eleman için farklı gerilimler kullanmak yerine uygun dirençler kullanarak her elemana gerekli gerilimler, direnç üzerinde düşürülerek temin edilir. Ledlere gerekli olan gerilimi sağlamak ve ışınsal bağlayıcıları tetikleyebilmek için 330 ohm'luk dirençler kullanılmıştır.

Işınsal Bağlayıcı (optokupler): Optoizolatör, fotokuplör veya optikal izolatör olarak da bilenen optokuplörler, birbiri ile optik bağlantılı ışın verici ve fotoalıcıdan oluşan, elektriksel bir bağlantı olmadan düşük gerilimlerle, yüksek gerilim ve akımları kontrol edebilen ve iki devrenin elektriksel olarak izolasyonunu (yalıtılmasını)

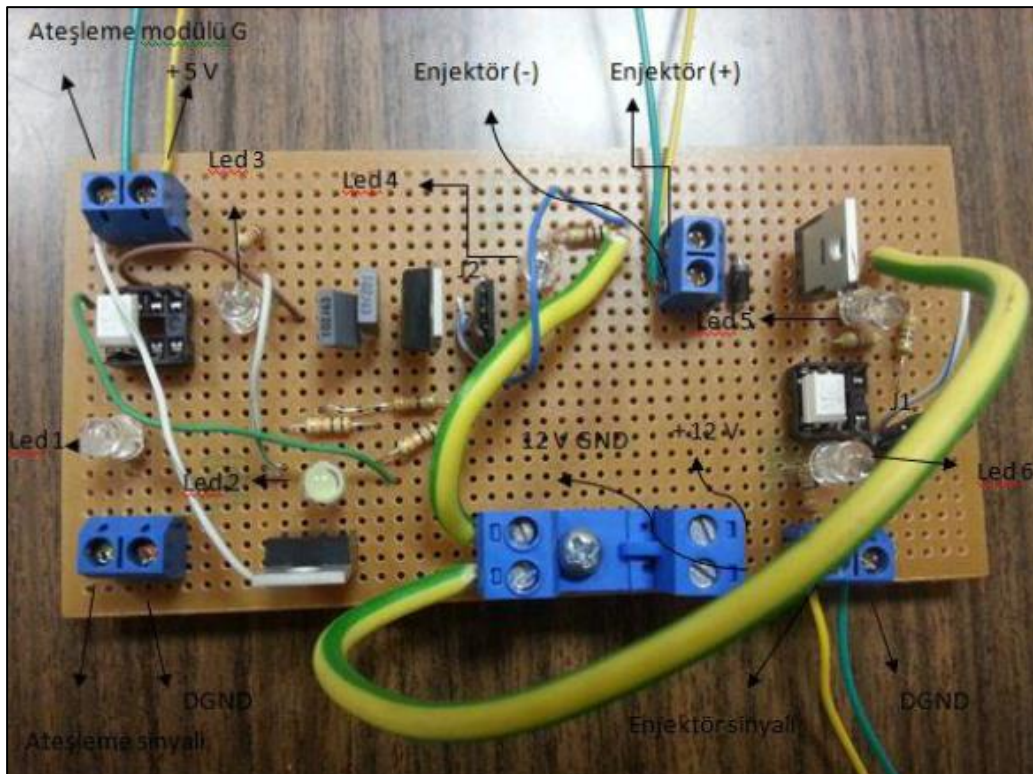
sağlayan, bir devre elemanıdır. Optokuplörlerin görevi, iki devreyi birbirinden fiziksel olarak ayırmak, yani elektriksel olarak yalıtmaktır [26]. Devrede tepki süresi çok kısa olan TLP521-1 ışınsal bağlayıcı kullanılmıştır. Ateşleme ve püskürtme tarafında birer tane bulunmaktadır.



Şekil 3.2: Işınsal bağlayıcı TLP521-1 [Ek A.1]

Mosfet: Mosfet (Metal Oxide Semiconductor Field Effect Transistör - Metal oksit yarı iletken alan etkili transistör) analog ve dijital devrelerde sık kullanılan bir alan etkili transistördür. Mosfetler, JFET transistörler gibi üç bacaklıdır. Bu bacaklar ; G (gate, normal transistörün base bacağı), S (source, kaynak) ve D (drain, normal transistörün kollektörü) bacaklarıdır. D ucu ile S ucunun çıkarıldığı bölgeye kanal denir. Mosfetlerde gate bacağı ile kanal bölgesi arasında silisyum nitrat ve silisyum oksit ile yalıtım yapılmıştır. Bu metal oksit tabaka çok ince olduğundan statik elektriğe karşı oldukça hassastır. Mosfetlerin giriş empedansı yüksek, elektrodları arasında iç kapasitansları ise çok düşüktür. Mosfet transistörler, JFET transistörlerden ve normal transistörlerden daha yüksek frekanslarda çalışabilirler. Mosfet transistörlerin güç tüketimleri düşüktür ve mekanik dayanımları fazladır. Ayrıca, mosfet kanal bölgelerinde kullanılan maddelere göre N tipi mosfet ve P tipi mosfet olmak üzere iki çeşittir. Çalışma şekillerine göre ise mosfetler; enhancement (çoğaltan - arttıran kanallı) mosfetler ve depletion (deplasyon - azaltan kanallı) mosfetler olarak iki çeşittir [4]. Kart üzerinde iki adet IRFZ44N mosfeti kullanılmıştır.

1N4007 Diyot: Enjektör solenoidi üzerinden geri dönüp mosfete zarar verebilecek ters tepe gerilimlerini önlemek amacıyla kullanılmıştır.



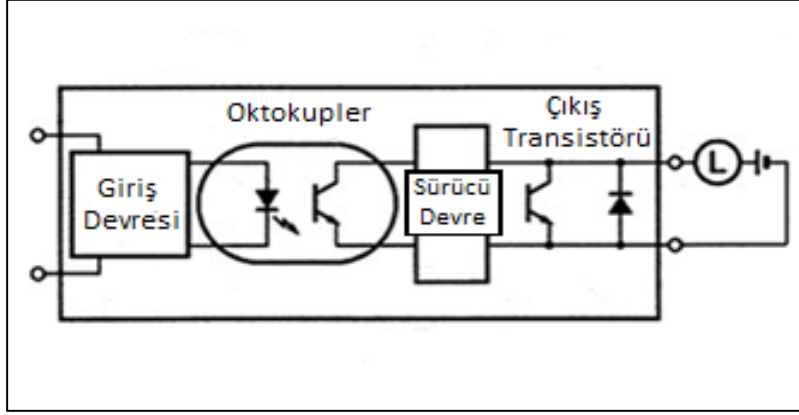
Solid-State röle (kısaca SSR), tamamen elektronik parçalardan oluşmuş bir anahtarlama düzenidir. Klasik röle ve kontaktörlerle aynı işi yapar. Kontaktör ve röle gibi kumanda ve güç devrelerine sahiptir. Solid-State rölede kumanda devresi transistör benzeri elemanın enerjilendirilmesi ile güç devresi transistör benzeri elemanı açık konumdan kapalı konuma gelerek sisteme enerji verir. Bu bir elektrik rölesi veya kontaktörün bobinine enerji verilerek, kontaklarının açık konumdan kapalı konuma gelmesi gibi bir uygulamadır. Kontaktör ve röleden farkı, bobin ve kontaklar yerine elektronik devre elemanlarının olmasıdır.

Kullanımda, röle ve kontaktörden en temel karakteristik farklarından biri güç devresinin Solid-State rölenin tipine göre yalnızca belirli alt ve üst limit gerilim aralıklarında çalışabilmesidir. Bir kontaktörün açılıp kapanması, basitçe iki iletken metalin birbirine teması veya temas etmemesidir ve bu mekanik bir olaydır. Bu mekanik hareket, kontaklar üzerindeki gerilimden bağımsızdır. Örneğin bir kontaktör güç devresi 0,1 voltla 600 volt arasında DC veya AC herhangi bir sistemi anahtarlıyor olabilir. Solid-State rölelerde böylesine geniş bir güç devresi voltaj çalışma aralığından, AC veya DC herhangi bir gerilimi anahtarlama bahsetmek mümkün değildir. Solid-state röleler, güç devrelerinde kullanılan transistör benzeri elemanların karakterine göre değişen güç devresi çalışma aralığına sahiptir.

Transistör benzeri elektronik elemanlar, çalışma için asgari bir gerilim değerine ihtiyaç duydukları için mekanik kontaklarda olduğu gibi 0.1, 1, 2 volt gibi bir gerilimi solid-state rölelerle anahtarlama mümkün değildir. Bir solid-state röleyle minimum voltaj değerinin altında bir güç devresi anahtarlama istenirse solid-state röle zarar görmez. Yalnızca anahtarlama yapmaz. Ancak solid-state rölenin üst gerilim değerinden daha yüksek voltaj değerinde anahtarlama yapılmak istenirse transistör benzeri elektronik anahtarlama elemanının dayanabileceği üst gerilim değeri aşılmış olur, cihaz hasar görür ve kullanılmaz hale gelir. Böyle bir hasarın oluşması için 1 saniyenin altında, çok kısa süreler uygulanacak bir gerilim aşımı yeterlidir [27].

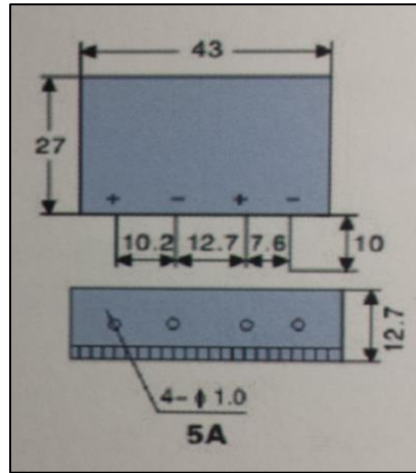
Solid-State rölelerin çalışma mekanizması genel olarak şu şekildedir; Girişe bağlı küçük güçlü ve küçük tetikleme gerilimli bir yarı iletken (Tristör yada Transistör), çıkışa bağlı yüksek güçlü küçük tetikleme gerilimli diğer bir yarı iletkeni tetikler. Çıkışa bağlı olan tristör yada transistör üzerinden akım geçirmeye başlar. Girişe uygulanan küçük gerilim kesildiği zaman çıkışa bağlanan tristör ya da transistör

yalıtkan olur. Üzerinden akım geçmez. Solid-State röle içerisinde herhangi bir kontak yoktur. Kontak olarak belirtilen çıkışa bağlı olan tristör ya da transistörün akım geçiren uçlarıdır.



Şekil 3.4: Solid-State rölenin çalışma mekanizması

Solid-State rölelerin kontak yerine yarı iletken malzeme bulundurması normal rölelere oranla çok daha hızlı açılıp kapanmasını sağlar. Arksız açma kapama yapması nedeniyle parlayıcı ve patlayıcı ortamlar için kullanılabilir olması, bu sebeple çok uzun ömürlü olması gibi avantajları vardır. Ayrıca küçük boyutları sayesinde portatif bir elektronik kutu tasarımına imkan vermesi, üzerinde bulunan sinyal ledi sayesinde oluşabilecek olası arıza ve hataların tespit ve onarımının kolaylığı, kolay sökülüp takılması sayesinde ileride sistem üzerine yapılabilecek ekleme ve çıkarmaları kolaylaştırması ve daha önce tasarlanan devrenin imalat maliyetine çok yakın bir maliyetle temin edilebilmesi Solid-State rölenin başlıca kullanım sebepleridir.

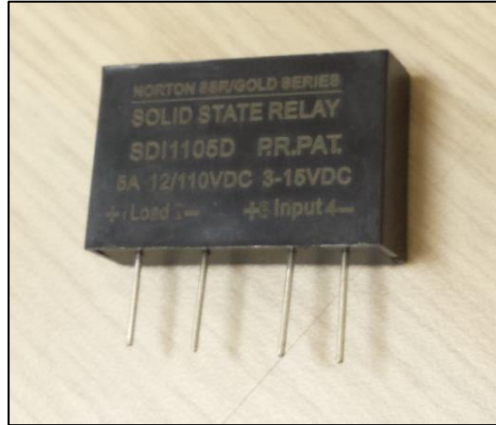


Şekil 3.5: Solid-State rölenin boyutları [11].

Yapılan arařtırmalar sonucunda deneylerimizde doęru akımı yine doęru akıma çevirebilen (DC-DC), monofaze 5A akım çeken Norton SDI-1105D Kart Tipi Solid-State rölenin kullanılmasına karar verilmiřtir. Deneylerde enjektörler için 4 adet, bujiler için 2 adet olmak üzere toplamda 6 adet Solid-State röle kullanılacaktır.

Çizelge 3.1: Norton SDI-1105D Solid-State röle özellikleri [28].

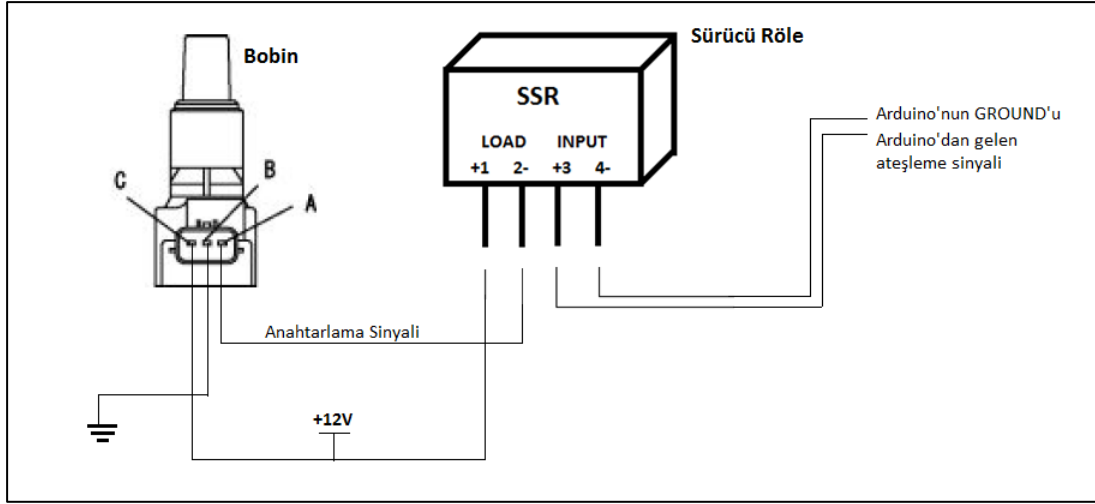
Yük Voltajı	12-300VDC
Amper	5A
Kontrol voltajı	3-15V
Kontrol Akımı	3-30mA
Çalıştırma Voltajı	$\leq 1,3V$
Kaçak Toleransı	$< 0,05mA$
Açma Kapama Süresi	$\leq 0,1ms$
Gösterge	LED
Çalışma Sıcaklığı	$-40\sim 80^{\circ}C$



Şekil 3.6: Norton SDI-1105D kart tipi Solid-State röle

3.2.1 Ateşleme bobininin solid state röle ile kullanımı

Solid-State (sürücü) rölenin Input girişinin 4 numaralı bacağı (-), Arduino mikrokontrolcü kartın toprak hattına bağlanır. Input girişinin 3 numaralı bacağı (+), Arduino mikrokontrolcü kartın ateşleme sinyali çıkış pinine bağlanır. Solid-State rölenin Load çıkışının 1 numaralı bacağı ve bobinin C ucu AKÜ'nün artı (+) kutubuna, bobinin B ucu da toprağa bağlanır. Son olarak Solid-State rölenin Load çıkışının 2 numaralı bacağı (+) da ateşleme bobininin A sinyal ucuna bağlanarak anahtarlama devresi tamamlanır.



Şekil 3.7: Sürücü röle – ateşleme bağlantı şeması

Ateşleme bobinin birincil sargısı üzerinden akım geçtiği süre, yani bobinin şarj olduğu süre ateşleme bobininin dwell süresidir. Bu dwell süresinin, ateşleme bobininin A ucu anahtarlanarak mikrokontrolcü tarafından kontrol edilmesi amaçlanmaktadır. Ateşleme sinyalinin Solid-State röleye gelip devrenin kapanmasından, ateşleme sinyalinin kesilip devrenin tekrar açılmasına kadar geçen süre, ateşleme bobininin dolması; başka bir deyişle bujide kıvılcım oluşturacak enerjiye sahip olabilmesi için geçmesi gereken süre olarak da tabir edilen dwell süresi dolduğunda, ateşleme sinyali kesilecek ve Solid-State röle tekrar açık konuma gelecektir. Solid-State rölenin açık duruma gelmesiyle ateşleme bobininin A ucu ile toprak hattı arasındaki bağlantı kesilecektir. Bu sayede bobinin çalışma karakteristiğine uygun olarak bobinin enerjisi bujiye aktarılacak ve bujiden istenilen kıvılcım elde edilmiş olacaktır.

3.3 Deneysel Veriyi Hafızalama Çalışmaları

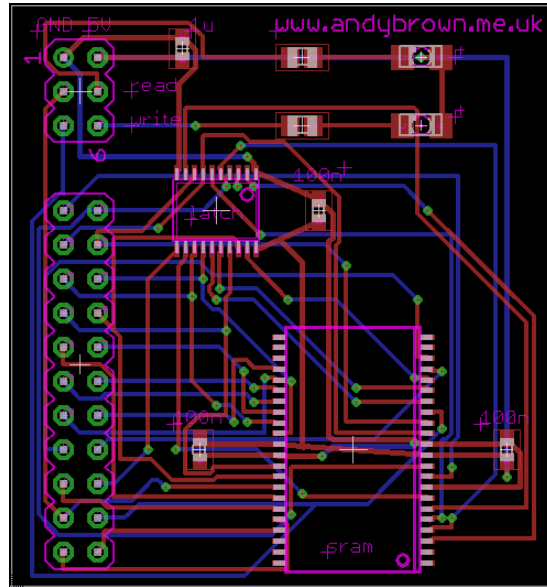
Tasarlanan ateşleme ve püskürtme üniteleri, Wankel motoru üzerinde çalışan Yük.Müh. Ömer Cihan'ın doktora tezi çalışmalarında yapacağı deneylerde kullanılacaktır. Bu kapsamda hem Ömer Cihan'ın hem de sistem üzerinde ileride çalışacak araştırmacıların yapacağı deneylerde toplayacakları verilerin depolanabilmesi ve gerektiğinde kullanılabilmesi için sisteme donanımsal olarak bir veya birden fazla harici hafıza elemanı eklenmesi araştırmalarına başlanmıştır.

Veri depolamada amaç, ileride yapılacak deneylerde değişen motor devri ve gaz keleşliği açıklığı değerlerinde optimum ateşleme avansı ve yakıt püskürtme

miktarlarını kaydetmek, bu değerleri içeren bir harita oluşturmak ve bu haritayı hafızalama elemanına gömerek motorun hafızalama elemanından aldığı veriler ile en verimli şekilde çalışmasını sağlamaktır.

Veri depolama denildiğinde akla ilk gelen komponentler RAM (Random Access Memory) ve ROM (Read Only Memory) olmuştur. RAM, rastgele erişimli bellek anlamına gelmektedir. İçeriğine bilgi yazmak ve okumak için tasarlanmıştır. ROM ise sadece okunabilir bellek anlamına gelir. RAM gibi yazılıp silinebilen bir depolama birimi değildir. ROM içeriği sadece imalat anında yazılır. Hafızalama elemanı için öncelikle RAM, ROM ve kontrolcü kartımıza uyumları incelenmeye başlanmıştır.

Literatürde Arduino mikrokontrolcü karta RAM veya ROM'un entegre edildiği nadir uygulamalara rastlanmıştır. Bu uygulamalardan birinde İngiliz programcı ve mikrokontrolcü kart tasarımcısı Andy Brown'un kendi internet sitesinde [29] yayınladığı 512KB hafızalı SRAM'in (Static Random Access Memory) Arduino karta entegre edildiği görülmüş ve aynı kartı tasarlayıp imal edebilmek için çalışmalara başlanmıştır.



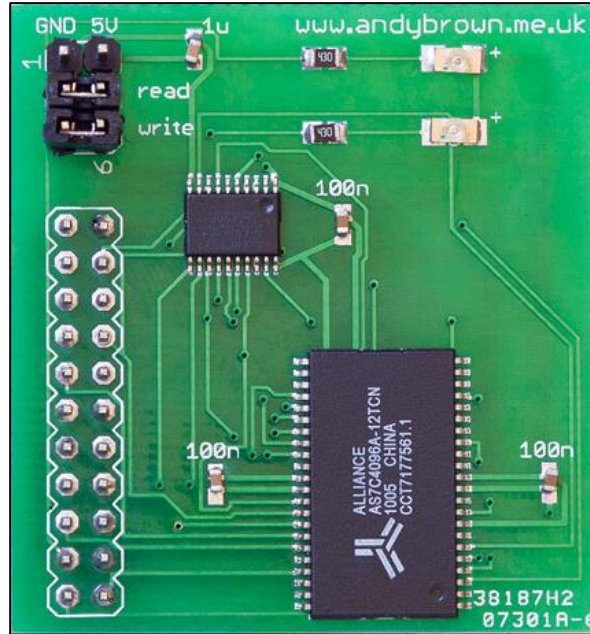
Şekil 3.8: İmal edilmesi planlanan SRAM'in baskı devre çizimi

SRAM kartı imal etmek için, öncelikle Eagle devre tasarım programında çizilmiş baskı devresi ve üzerindeki komponent listesi temin edilmiştir. İmal edilmesi amaçlanan çift katlı baskı devre kartın boyutları 4-4,3 cm, sinyal çizgileri (ince) 0,02 cm, güç çizgileri (kalın), 0,04 cm'dir.

Komponent	Model	Adet
SRAM	Alliance AS7C4096A 12ns	1
Latch (Mandal)	NXP 74AHC373	1
Kapasitör	100nF	3
Kapasitör	1µF	1
Direnç	approx 430Ω	2
LED – sarı	Kingbright	1
LED – kırmızı	Kingbright	1

Şekil 3.9: Örnek SRAM kartı komponentleri

Devrede en önemli iki eleman olarak Alliance SRAM ve NXP Latch (Mandal) kullanıldığı görülmüştür. SRAM'in en önemli özelliği sürekli tazelenmesi gereken RAM'in aksine, belleğe güç verildiği sürece belleğin içeriğinin korunduğu bir RAM çeşidi olmasıdır. Latch (mandal) ise bir giriş sinyali ile durumu değişmedikçe ikili bir bilgiyi (sinyal ve değerler) güç verildiği müddetçe saklayabilen devre elemanıdır. Devrenin imal edildikten sonraki hali Şekil 3.10'daki gibi olacaktır.



Şekil 3.10: Örnek SRAM kartın son hali

Edinilen bilgiler ışığında SRAM kartın imal edilebilmesi için çalışmalara başlanmıştır. Türkiye elektronik piyasasında bulunamayan SRAM ve Latch için distribütör firma ile irtibata geçilmiş ve sadece SRAM için toplu miktarda (19 adet) temin etmek ve Mayıs 2015'te teslim almak şartıyla 101 USD (Amerikan Doları) fiyat alınmıştır. Baskı devre kartı için PrintPCB isimli imalatçı firmadan 1 hafta teslimli 180 USD, 3 hafta teslimli 140 USD fiyat alınmıştır. Çok ince lehim işçiliği gerektiren komponent dizme işlemi için, EMS Elektronik firması, çoklu imalat yaptığından dolayı dizgi işleminin tek bir kart için yapılamayacağını, yapıldığı takdirde ücretinin normalin çok üzerinde olacağını belirtmiştir.

Sonuç olarak tek bir kartın imalat ve dizgi işlemlerinin çok pahalı oluşu, üzerindeki komponentlerin yurtdışı menşeli oluşu, teslim süreleri ve yüksek ücretleri, Türkiye'den temin edilebilecek benzeri komponentler için yeni bir baskı devre tasarımı gerekliliği ve ayrıca ileride SRAM'ın Arduino mikrokontrolcüde programlanması üzerinde çalışacak kişiler için yeteri kadar kaynak barındırmaması gibi dezavantajları değerlendirilecek SRAM kartı imalinden vazgeçilmiş, Arduino ile uyumlu, düşük maliyetli başka bir komponent aranmaya başlanmıştır.

3.3.1 Eeprom

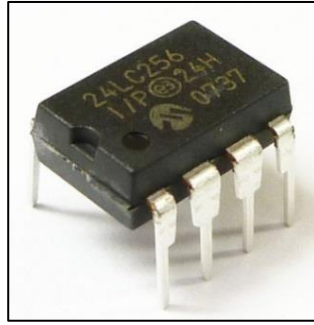
Yapılan araştırmalar sonucunda Arduino'nun kendi bünyesinde de barındırdığı (4KB) ve literatürde kaynaklarına rastlanabilen EEPROM'un (Electrically Erasable Programmable Read Only Memory) harici bir veri depolama elemanı olarak kullanılabileceği düşünülmüştür.

EEPROM elektriksel olarak silinebilen ve programlanabilen bir bellek türüdür. Bu bellek türünde bilgilerin yazılması için ve silinmesi için normal elektrik voltajı uygulanır. Diğer ROM türlerinde olduğu gibi içerdiği veriyi elektrik kesilse de saklar. EEPROM'lar birkaç byte dan 128 KB'a kadar olan aralıklarda, tipik olarak düzen parametrelerini saklamada rol oynarlar. Modern bilgisayarlarda şimdiye dek kullanılan CMOS kalıcı BIOS bellek teknolojisi (CMOS nonvolatile BIOS memory) ile yer değiştirmiştir. Ram üzerine tekrar tekrar yazmak mümkünken, EEPROM'lar üzerlerinde bulunan ince yalıtkan madde zarar görene dek yazma ve silme işlemi yapabilmekle sınırlandırılmıştır.

İlk kullanılan EEPROM'lar 100 defa silme-yazma işlemi gerçekleştirirken, günümüz teknolojisinde kullanılan EEPROM'lar 1.000.000'dan fazla yazma-silme işlemini

yerine getirebilmektedirler. Flash bellekler EEPROM'dan daha ucuz olmasına karşın daha çabuk yıpranır ve zarar görürler (yaklaşık 10.000 yazma-silme işleminden sonra). EEPROM'un daha etkili olduğunun düşünülmesinin diğer bir sebebi; flash bellek birden fazla bellek bölgesini aynı anda silmek zorundadır. Sadece bir baytı değiştirmek için tüm bloğu silmek ve tekrar yazmak gerekir. Bu da flash belleğin çabuk yıpranmasının sebebidir [30].

Sonuç olarak bilgisayarda kullanılan Flash bellekler gibi çalışan, program işlerken müdahale edebilen, okunabilen veya üzerine yazabilen, elektrik kesildiğinde de verileri saklamaya devam eden EEPROM'un, deneysel verileri kaydetmek ve gerektiğinde kullanabilmek açısından gereksinimleri karşılayacak bir ekipman olduğu kanısına varılmıştır. Ayrıca Arduino'ya donanımsal ve yazılımsal olarak entegre edilebilmesi, kolay temin edilebilirliği ve uygun maliyeti de EEPROM'un seçilmesindeki önemli etkenlerdir.

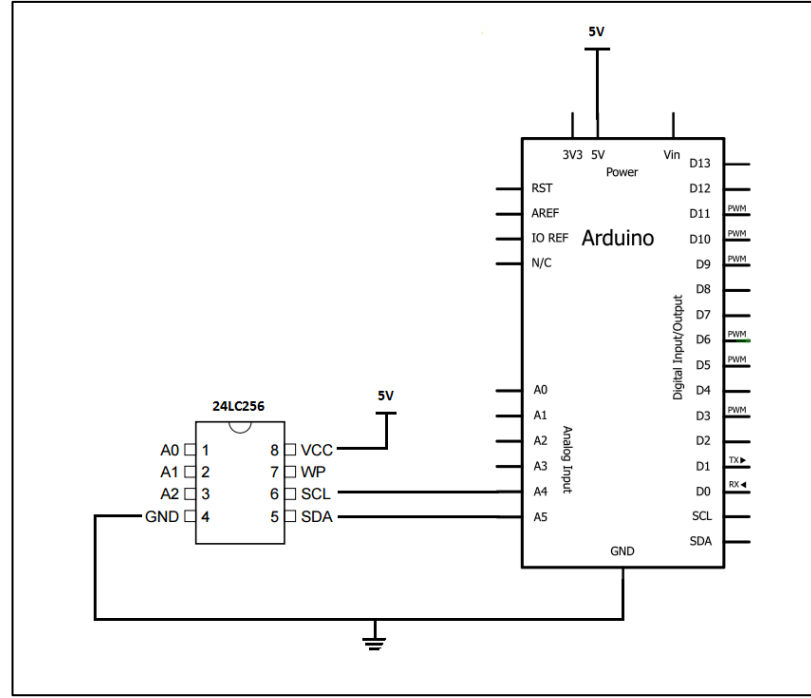


Şekil 3.11: Atmel 24LC EEPROM

A0	A1	A2	Adress
Gnd	Gnd	Gnd	0X50
+5V	Gnd	Gnd	0X51
Gnd	+5V	Gnd	0X52
+5V	+5V	Gnd	0X53
Gnd	Gnd	+5V	0X54
+5V	Gnd	+5V	0X55
+5V	+5V	Gnd	0X56
+5V	+5V	+5V	0X57

Şekil 3.12: 24LC256'nın adres değerleri

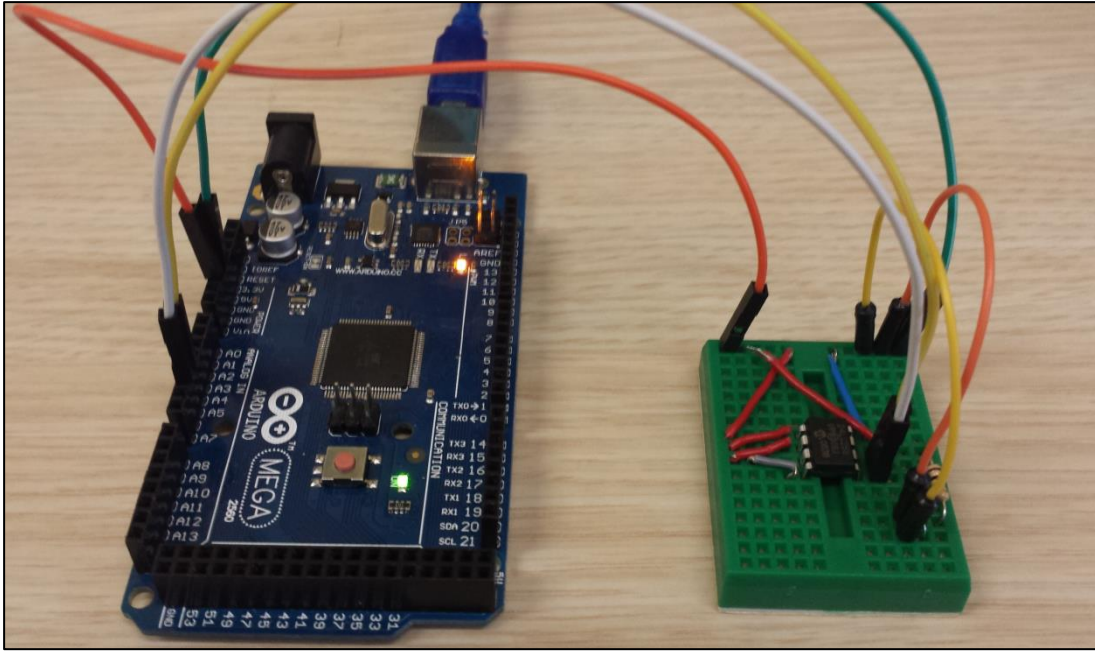
Sistemde piyasadaki en yüksek hafızalı EEPROM olan Atmel 24LC256 kullanılmıştır. 256 KB hafızaya sahiptir. A0, A1 ve A2 bacakları ile EEPROM'un adres bilgisi Şekil 3.12' de gösterilen değerlere göre değiştirilebilir. Herhangi bir değişiklik yapılmazsa default adresi 0x50'dir. A0, A1 ve A2 pinleri +5V'a bağlandığında adres 0x57 olacaktır. Write Protect (WP) bacağı HIGH yapıldığında yazmaya karşı korumalı duruma geçer. Bu pin boşta bırakılmalı veya toprak hattına bağlanmalıdır. SDA veri transfer hattıdır. SCL ise clock hattıdır. Pozitif kenar geldiğinde veri yazılır, negatif kenar geldiğinde veri okunur [12].



Şekil 3.13: EEPROM ve Arduino bağlantısı

Sisteme EEPROM eklenmesi, ileride yapılacak çeşitli deneylerde veri depolamak ve bu verileri kullanmak isteyen araştırmacılara kolaylık sağlamak amacıyla uygulanmıştır. Her deney için değişkenler ve depolanması istenen veriler değişeceği için, yazılan kodlar da değişiklik gösterecektir. EEPROM'un programlanması tez kapsamında değildir. EEPROM'un sadece donanımsal olarak sisteme entegre edilmesi ve Arduino mikrokontrolcüye kurulması amaçlanmıştır. Bu doğrultudaki çalışmalardan Bölüm 4.3' te bahsedilmiştir.

EEPROM devresi ileride yaşanabilecek arızalarda veya daha fazla hafıza gerektiğinde kolay sökülüp takılabilmesi için mini bir breadboard üzerine kurulmuştur.

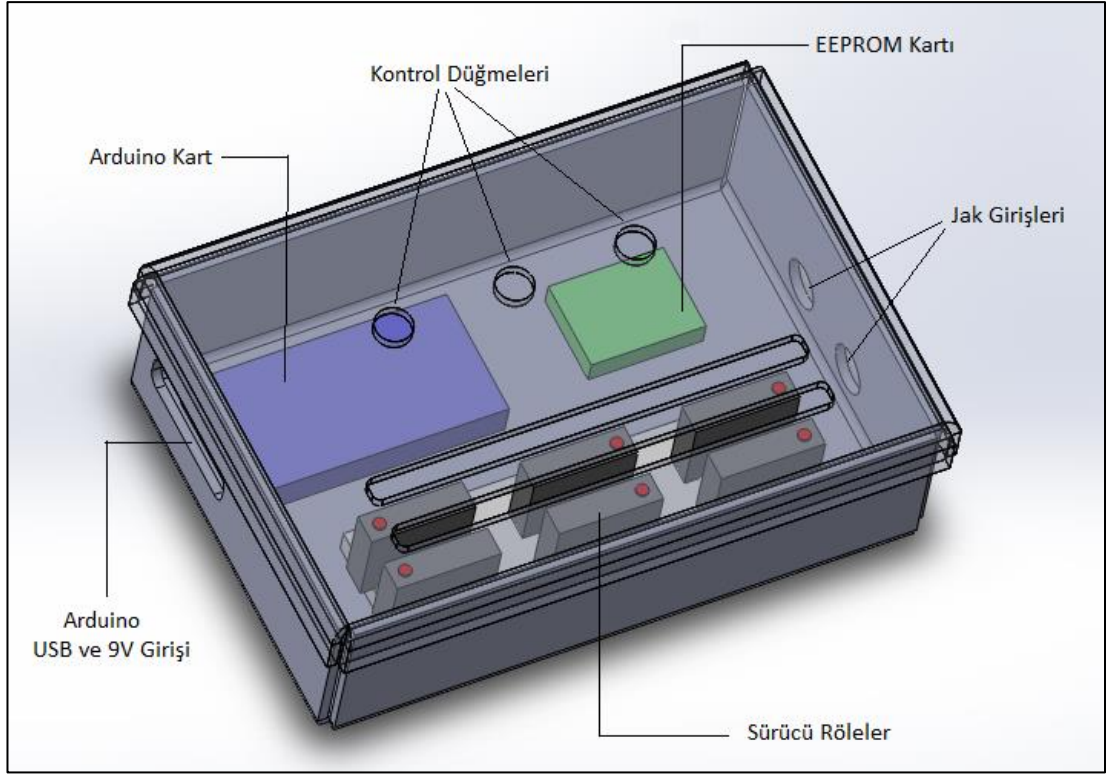


Şekil 3.14: EEPROM'un Arduino'ya bağlandıktan sonraki görünümü

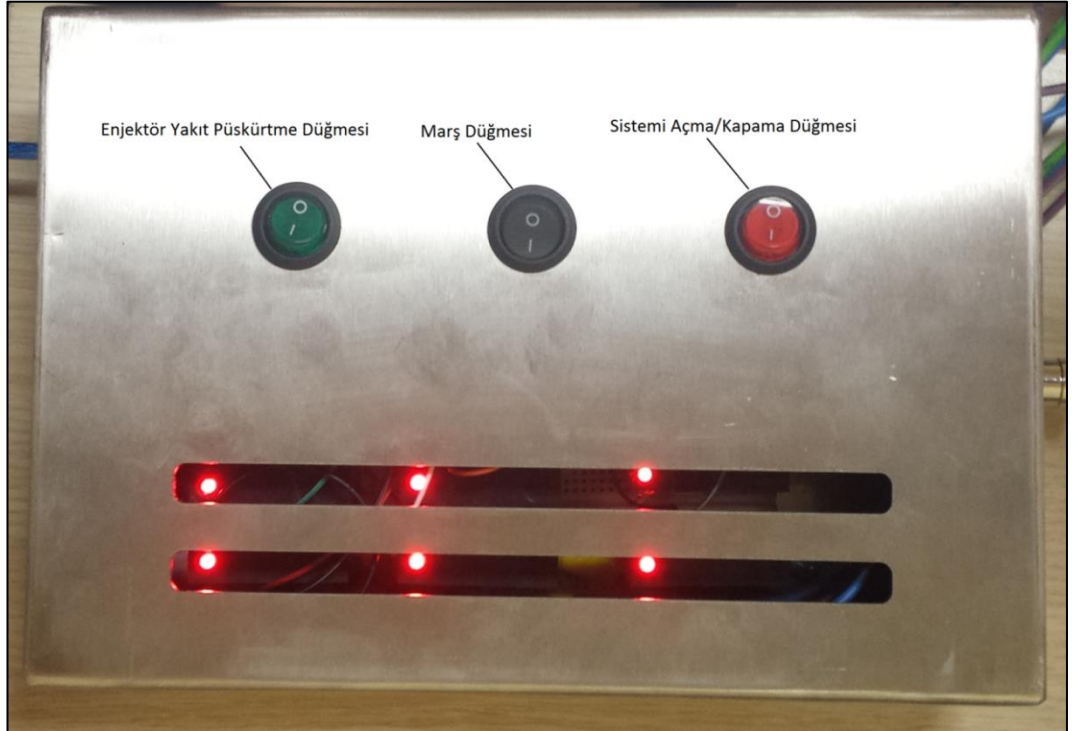
3.4 Kontrol Cihazı

Kontrol cihazı, içeriğinde Arduino mikrokontrolcü kartını, sürücü röleleri, kontrol düğmelerini, kablo soketlerini ve EEPROM kartını barındıran bir kutudur. Profesyonel bir görüntüsü olması ve montajının kolay olması açısından Solidworks programında tasarlanıp imal edilmesi amaçlanmıştır.

Tasarım aşamasında öncelikle komponentlerin boyutları ve yerleşim yerleri belirlenmiştir. Mini breadboard boyutları 47,5 x 37 mm, büyük breadboard boyutları 166 x 56 mm, breadboard ve Solid-State röle kalınlıkları toplamı 45 mm' dir. 1mm paslanmaz çelik saçtan imal edilmesi planlanan kutunun boyutları 170-250 mm ve yüksekliği 73 mm olarak belirlenmiş ve teknik resmi çıkartılarak saç lazer kesim ve büküm işlemleri için imalatçı firmaya gönderilmiştir. Şekil 3.15'de kontrol cihazının Solidworks çizimi bulunmaktadır.



Şekil 3.15: Kontrol cihazının Solidworks'te çizimi



Şekil 3.16: Kontrol cihazının dış görünüşü

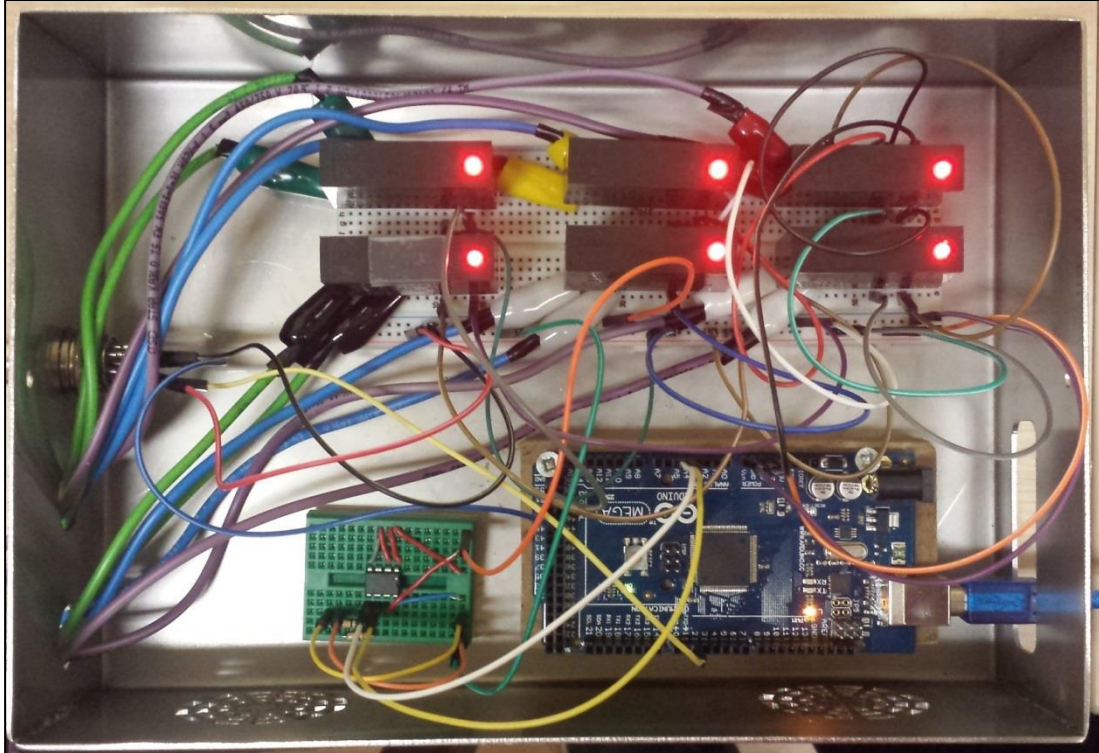
Açma – Kapama düğmesi: Açık – Kapalı konumları olan bu anahtar ile kontrol cihazına gelen gerilim sürücü rölelere ve bobine giden gerilimi anahtarlamaktadır.

Marş düğmesi: Marş düğmesi ile motor tesisatı üzerindeki marş rölesinin anahtarlanması ile marş motoru çevirilmektedir.

Yakıt püskürtme düğmesi: Özellikle ilk çalıştırma için yakıt az gelebilmektedir. Motor yol almadığı zaman bu düğme kullanılarak emme portuna yakıt püskürtülebilmektedir.

Arduino USB bağlantısı: Arduino ile bilgisayar arasındaki iletişimi sağlamak için kullanılan USB B tipi kablodur.

Arduino güç girişi: Arduino 5 volt ile çalışmaktadır. Ancak içerisindeki voltaj regülatörü sayesinde daha yüksek voltajlı adaptörler kullanılmasına izin vermektedir. Arduino mikrokontrolcüyü ve mikrokontrolcüye bağlı olan enkoderi beslemeye yarayan 9 volt 1 amper değerindeki güç adaptörü buraya takılmalıdır.

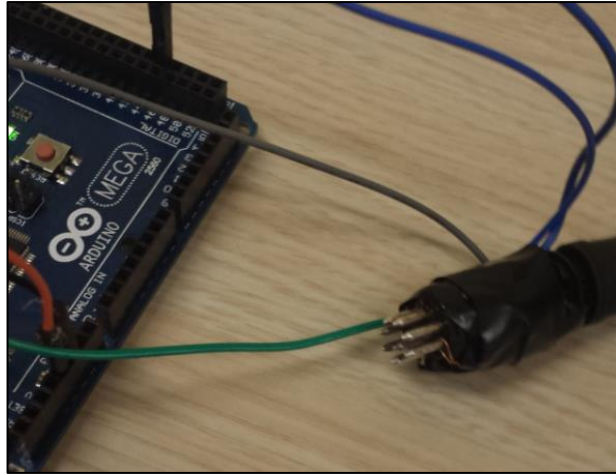


Şekil 3.17: Kontrol cihazının iç görünüşü

Kontrol cihazının içerisinde 4 adet enjektör ve 2 adet bobin olmak üzere 6 adet sürücü röle bulunmaktadır. Bu röleler kart tipi olduklarından ve kolay çıkarılıp takılabilmeleri açısından bir breadboard üzerine yerleştirilmiştir. Arduino ile sürücü röle girişlerinin bağlantısı için 5 volt taşıyan 0,25 mm'lik data kabloları kullanılmıştır. Sürücü röle çıkışlarından motora giden 12 volt taşıyan 1,5 mm'lik kalın kablolar içi boş 2 adet jak ile kontrol cihazına bağlanacaktır. Hafızalama elemanı olarak kullanılan EEPROM devresi, yine bir mini breadbord üzerine kurulmuştur. Ve Arduino ile bağlantısı 0,25 mm'lik data kablolarıyla sağlanmıştır.

Kontrol cihazı içerisindeki elektronik komponentlerin havalandırma ve soğutma gereksinimlerini karşılayabilmek amacıyla cihazın yaz yüzüne 2 adet havalandırma ızgarası açılmıştır. Bu ızgaraların birinden Arduino kartın power (güç) ledinin yanıp yanmadığı gözlemlenebilmekte ve bu sayede kullanıcıya gerekli enerjinin gidip gitmediğinin kontrolü imkanı sunulmaktadır.

Enkoder kablosu bağlantısı: Daha önceki çalışmalarda normalde 12 pinli olması gereken Heidanhain enkoder kablosu 8 pinli 15mm'lik bir bağlantı ile kontrol cihazına bağlanacaktır. Dolayısıyla enkoder bağlanmadan önce Heidanhain Türkiye distribütörü ile irtibata geçilmiş ve enkoder çıkış kablo renkleri ve bağlantılarıyla ilgili teknik destek alınmıştır.



Şekil 3.18: Enkoder motor tesisatına bağlanmadan önceki bağlantısı

Çizelge 3.2: Enkoder bağlantısının numaralandırması

1 numaralı pin	Enkoder negatif kutbu (-)
4 numaralı pin	Enkoder Z sinyali
7 numaralı pin	Enkoder pozitif kutbu (+)
5 numaralı pin	Sayıcı pin

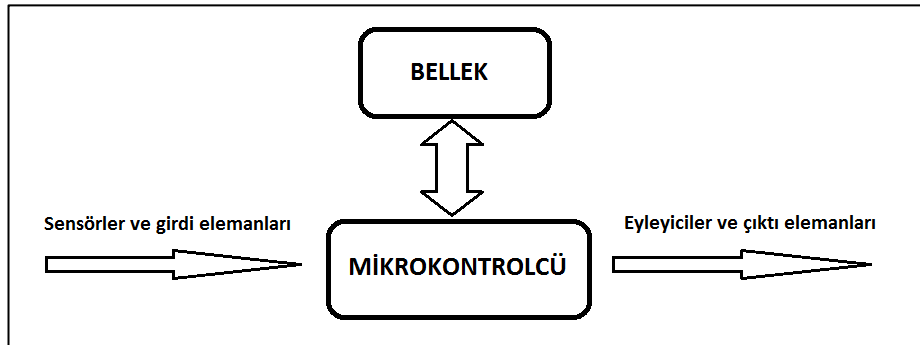
4. KONTROL SİSTEMİ YAZILIMI

4.1 Kontrolcü Seçimi

Kontrolcü seçimi için öncelikle literatürde yapılan benzeri çalışmalar incelenmiştir. Dr. Akın Kutlar'ın doktora tezinde [10], Müh. Ahmet AKIN'ın PC tabanlı kişisel bilgisayarın ISA genişleme portuna yaptığı kart ve kişisel bilgisayar ile çalışan bir sistem kullanılmıştır. Ancak günümüzde kullanılabilecek ISA genişleme portlu bilgisayarlar bulunmadığından farklı bir yöntem arayışı içine girilmiştir.

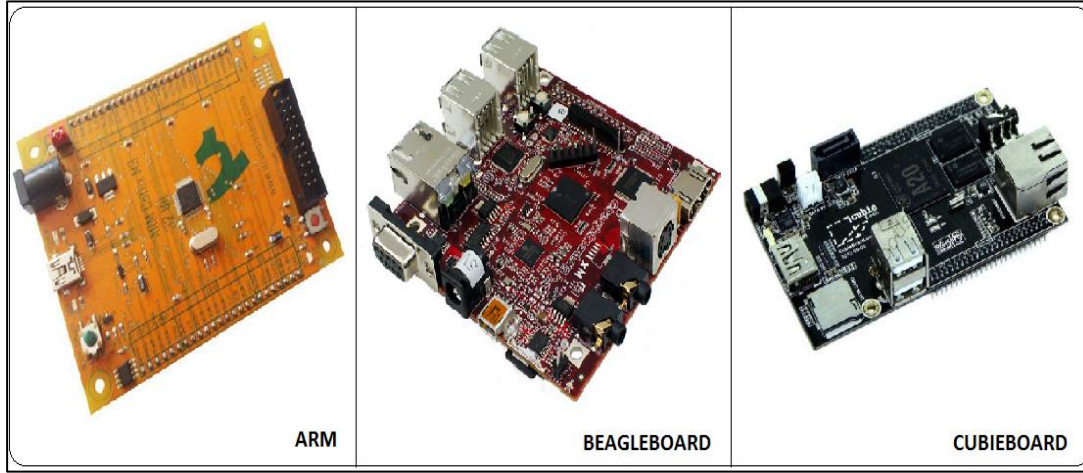
Müh. Taner Yıldırım'ın [1] yüksek lisans tezinde Müh. Ahmet AKIN'ın tasarladığı PIC16F877A mikrodeneleyici ile çalışan bir kontrol kullanılmıştır. Bilgisayar ile haberleşmesi ise günümüz bilgisayarlarında yerini USB porta bırakmış güncelliğini yitirmiş seri porttan gerçekleştirilmiştir. Ayrıca PIC entegresini kullanabilmek için temel olarak bazı elektronik devre parçalarının (kristal, direnç, kondansatör gibi) bulunduğu yeni bir devre kartı tasarım ve imalat uğraşı gerektirdiği düşünülerek hazır mikrokontrolcü kartlara yönelinmiştir.

Belirli bir işi gerçekleştirmek için tasarlanmış donanım, yazılım ve bazen de mekanik bileşenlerden oluşan ve çoğu zaman kendisinden büyük bir sistemin içine gömülü olarak çalışan sistemlere gömülü sistemler denir. Mikrokontrolcüler cihazların içinde gömülü olarak çalışırlar. Böylece o cihazın hareketlerini ve özelliklerini kontrol edebilirler.



Şekil 4.1: Gömülü sisteme ait bir blok diyagramı [13].

Günümüzde hobi robotiğinde ve akademik uygulamalarda sistemlerin kontrol edilmesinde yaygın olarak mikrokontrolcü kartlar kullanılmaktadır. Mikroişlemcilere göre avantajları, mikrokontrolcü kartların içeriğinde mikro işlemci, RAM, ROM, program belleği, sayıcılar, iletişim modülü, programlanabilir I/O portlar bulunmasıdır. Sonuç olarak sistemde uygun bir mikrokontrolcü kartın seçilerek kullanımına karar verilmiştir.



Şekil 4.2: Sırasıyla; ARM, Beagleboard ve Cubieboard marka mikrokontrolcüler

Mikrokontrolcü kartlar ve modelleri hakkında yapılan araştırmalar sonucunda, çeşitli fiyat ve işlevlerde Arduino, Beagleboard, ARM, Cubieboard gibi çok sayıda imalatçının kartları olduğu görülmüştür. Müh. Özgür TEKELİ'nin [4] tez çalışmasında Arduino marka mikrokontrolcü kart kullandığı ve herhangi bir sorunla karşılaşmadığı görülmüştür. Ayrıca diğer kartlara nazaran uygun fiyatları ve daha yaygın kullanımı sebebiyle Arduino kart ile ilgili çalışmalara başlanmıştır.

Yapılan araştırmalar sonucunda Arduino kartın geliştirme ortamının ve sürücülerinin kurulumunun kolay olduğu ve Windows, Linux ve Mac gibi platformlarında çalışabildiği görülmüştür. Geniş kütüphane desteği sayesinde birçok karmaşık işlemi kolaylıkla gerçekleştirebilmektedir. Programlar başka bir platform üzerinde koşmadığından ve yorumlamalı bir dil olmadığından oldukça hızlı çalışabilmektedirler. Arduino kart kendisiyle birlikte çalışabilecek bir çok ek donanım desteği (shield) sunmaktadır. Bununla birlikte piyasadaki sensörlerin çoğu Arduino karta bağlanabilmektedir. Benzerlerine göre fiyatları oldukça uygun olmakla beraber imalatçı firma tarafından kullanıcılara kendi Arduino kartlarını imal edip satma özgürlüğü verilmiştir. Açık kaynaklı olduğundan yazılımı için lisans ücreti söz konusu değildir. Ayrıca çok yaygın kullanıldığından Arduino ile ilgili literatürde ve

internette çok sayıda kaynak bulmak mümkündür. Yukarıda sayılan avantajları nedeni ile sistemde Arduino mikrokontrolcü kart kullanılmaya karar verilmiştir.

4.1.1 Arduino modeli seçimi

Arduino kartlarının birçok çeşidi bulunmaktadır. Temel olarak bütün kartlarda benzer bileşenler olmakla beraber mikrodenetleyici modelleri, giriş/çıkış pinleri, dahili modüllerin sayısı, boyut ve çalışma gerilimleri gibi farklara sahiptirler [13].

Arduino'nun kalbi mikrodenetleyicilerdir. Çoğu Arduino kartı Atmega mikrodenetleyicilerini kullanır. Arduino kartı ile derlenen program mikrodenetleyiciye aktarılabilir. Arduino programlama dili ile kullanıcılara mikrodenetleyicinin çevre birimlerine erişim sağlayabilme, ADC (Analog/Dijital Dönüştürücü) kullanabilme, genel amaçlı giriş/çıkış pinlerinden verip alıp gönderebilme ve iletişim hatlarını (SPI ya da Seri port) kullanabilme imkanı sunmaktadır. Tüm bu özellikleri kullanabilmek için kartın üzerinde standart olarak bulunan dişi pinlerden veya shield denilen yardımcı kartlardan yararlanmak mümkündür [12].

Kullanılacak Arduino modelinin seçimi için dikkate alınacak kriterler:

- İnterrupt (Donanımsal Kesme) pini bulundurması
- Timer (Zaman Sayıcı) ve Counter (Sayıcı) Kesme pini bulundurması
- 6 adet PWM çıkış pini bulundurması
- Genel amaçlı giriş/çıkış pinlerinin fazla olması

Donanımsal kesmeler bağlı oldukları pinin lojik durumundaki değişimler ile tetiklenirler. Kesme sinyali geldiğinde, mikrokontrolcü başka bir işlemle meşgulken, o işlemi bırakıp kesme fonksiyonuna girer ve kesme fonksiyonunu uyguladıktan sonra işlemine kaldığı yerden devam eder. Tasarlanan sistemde donanımsal kesme fonksiyonu enkoderin her turda kendisini sıfırlaması için kullanılacaktır.

Donanımsal kesmeler kendi kendilerine tetiklenmez. Kesme pininin dışarıdan bir lojik değişime tabi tutulması gerekir. Timer/Counter kesmeleri ise belirli bir süre ile programlanabilir ve istenilen zamanda tetiklenebilir. Sistemde Timer ve Counter kesme pinlerinden, enkoderin palsleri sayılırken ve motor devri hesaplanırken yararlanılmıştır.

Enjektör ve ateşleme çıkışları için mikrokontrolcude yüksek frekansta anahtarlama sağlayabilen, darbe genişlik modülasyonu (PWM) olarak da kullanılabilen pinler sinyal çıkışı olarak tanımlanacaktır. Ayrıca sisteme ileride eklenebilecek harici donanımlar düşünülerek genel amaçlı giriş/çıkış pin sayısı fazla olan bir kart seçilmesi amaçlanmıştır.

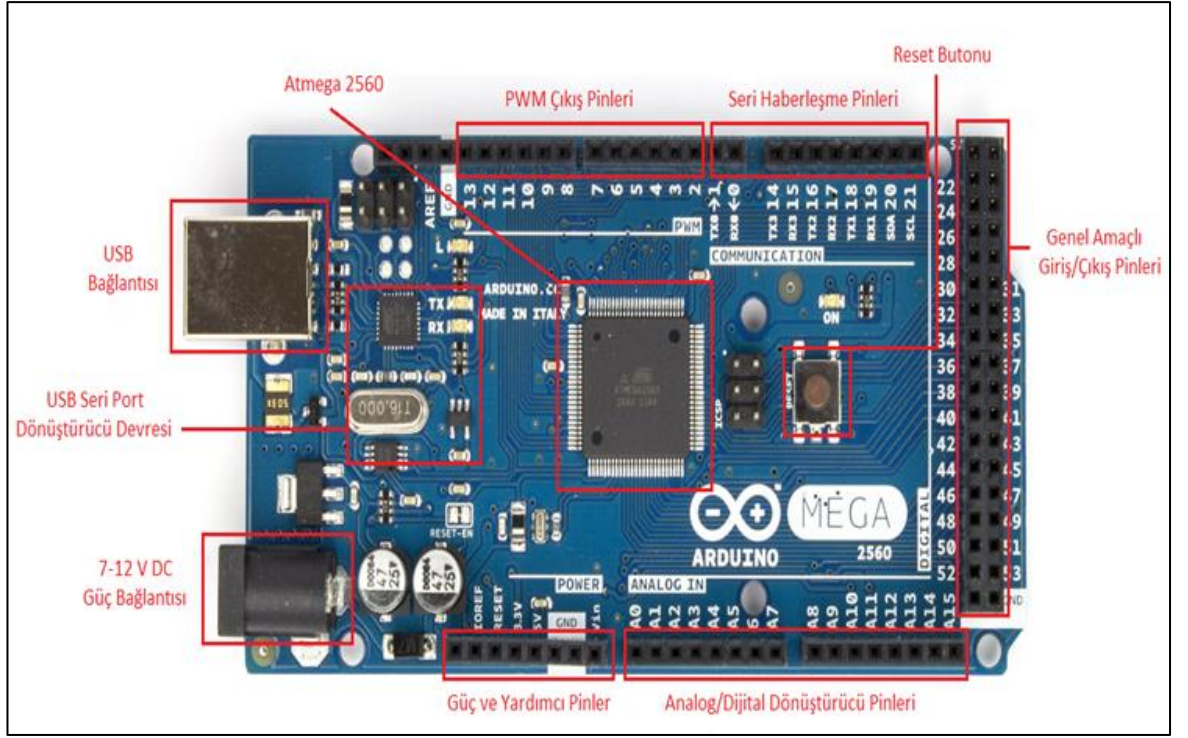
Sistemin gereksinimleri düşünüldüğünde, Arduino kartlar arasında en çok kullanılan 2 model olan Arduino Uno ve Arduino Mega isimleri öne çıkmıştır. Arduino Uno Atmega 328, Arduino Mega ise Atmega 2560 mikrodnetleyicisini kullanmaktadır. İşlemci frekansları, donanımsal ve Timer/Counter kesmeleri açısından her iki mikrodnetleyici de gereksinimlerimizi karşılamaktadır. Ancak bellek hafızasının yüksek olması ve genel amaçlı giriş çıkış pinleri sayısının fazla olması sebebiyle Arduino Mega 2560 mikrokontrolcü kart kullanılmasına karar verilmiştir.

4.1.2 Arduino mega 2560 özellikleri ve kullanımı

Arduino Mega üzerinde Atmega 2560 mikrodnetleyicisi bulunan, 54 dijital giriş/çıkış pinine (14 tanesi PWM olarak kullanılabilen), 16 analog girişe, 4 donanımsal seriporta (UART) ve 16 MHz kristal osilatöre sahip bir mikrokontrolcü karttır. Bütün bu özellikleriyle daha fazla giriş/çıkış pini gerektiren uygulamalar için idealdir. Hem USB üzerinden hem de DC adaptör üzerinden beslenebilmektedir [13]. Önerilen voltaj aralığı 7-12 V olsa da 6-20 V arasında da çalışabilir. 5V ve 3,3V'luk regülatör entegreleri ile çalışma voltajı sınırlanmıştır. Sistemde mikrokontrolcü beslemesinde 9V 1A bir DC adaptör kullanılacaktır.

Çizelge 4.1: Arduino Mega 2560 karakteristik özellikleri.

Mikrodnetleyici	ATmega2560
Çalışma Gerilimi	5V
Besleme Voltajı (Önerilen)	7-12V
Besleme Voltajı (Limit)	6-20V
Dijital I/O Pinleri	54 (14ü PWM çıkışı)
Analog Giriş Pinleri	16
I/O Pinlerinin Akımı	40 mA
3.3V Pini Akımı	50 mA
Flash Bellek	256 KB (8kB'ını bootloader kullanıyor)
SRAM	8 KB
EEPROM	4 KB
Clock Frekansı	16 MHz



Şekil 4.3: Arduino Mega 2560

Arduino Mega 2560 kartlarında ikinci bir mikrodenetleyici kullanılır. Bu mikrodenetleyicinin görevi; ana mikrodenetleyici olan Atmega 2560 denetleyicilerinin USB desteği olmadığı için bu denetleyiciler ile USART bağlantısı üzerinden haberleşerek USB porta arayüz sunarlar [12].

0 (RX) - 1 (TX), 19 (RX) – 18 (TX), 17 (RX) – 16 (TX), 15 (RX) – 14 (TX) pinleri seri data almak (RX) ve göndermek (TX) için kullanılmaktadır [4].

Arduino Mega 2560'ta bulunan 54 dijital pinin herbiri, IDE kod editöründe yazılan pinMode(), digitalWrite(), digitalRead() komutları sayesinde giriş/çıkış olarak seçilebilmektedir. Her pin dahili olarak 20 – 50 K Ω pull-up dirençleri ile bağlantılıdır ve maksimum 40 mA akım çekmekte ve 5 V ile çalışmaktadır [4].

Arduino Mega 6 adet donanım kesmesi pinine sahiptir. Gerekli ayarlama yapıldıktan sonra 2, 3, 18, 19, 20, 21 numaralı pinlerin lojik seviyesi attachInterrupt() komutu ile değiştirildiğinde kesme (interrupt) tetiklenir ve ilgili kesme fonksiyonu çalıştırılır.

Arduino üzerinde bulunan mikrodenetleyicide 10 bitlik bir analog dijital çevirici (ADC) bulunmaktadır. 10 bitlik ADC ile girişe uygulanan 0-5V arasındaki gerilimler 0 ile 1023 arasındaki sayısal değerlere çevrilir. (1024 farklı değer). Bu da

$5V/1024=4,88$ Mv ‘luk bir çözünürlük elde edilmesini sağlar. Yani girişteki gerilim değişimleri 4,88 Mv ‘luk bir hassasiyet ile okunabilir.

Arduino Mega üzerinde 15 adet (A0-A15) analog giriş pini bulunur. Bu pinlere uygulanan 0-5V arasındaki gerilimleri analogRead() fonksiyonu ile ölçülebilir. AnalogRead() fonksiyonu 0 ile 1023 arasında bir değer döndürür. Bu değerden giriş gerilimi (çevrim sonucu/1023)*5 V formülüyle elde edilir [13].

Vin : Harici güç kaynağı kullanıldığında Arduino’nun beslenmesi için kullanılmaktadır.

5V : Harici güç kaynağı kullanıldığında mikroişlemcinin ve karttaki diğer elemanların gücünün sağlanması için kullanılmaktadır. Bu güç, karttaki regülatör vasıtasıyla Vin’den de gelebilmektedir.

3.3V : Karttaki FTDI çip tarafından 3.3 V üretilmektedir.

GND : Topraklama pinidir.

0’dan 13’e tüm pinler analogWrite() fonksiyonu ile 8 bitlik PWM çıkışı sağlamaktadır.

50, 51, 52 ve 53 pinleri SPI kütüphanesi kullanılarak SPI iletişim kurulmasını sağlamaktadır.

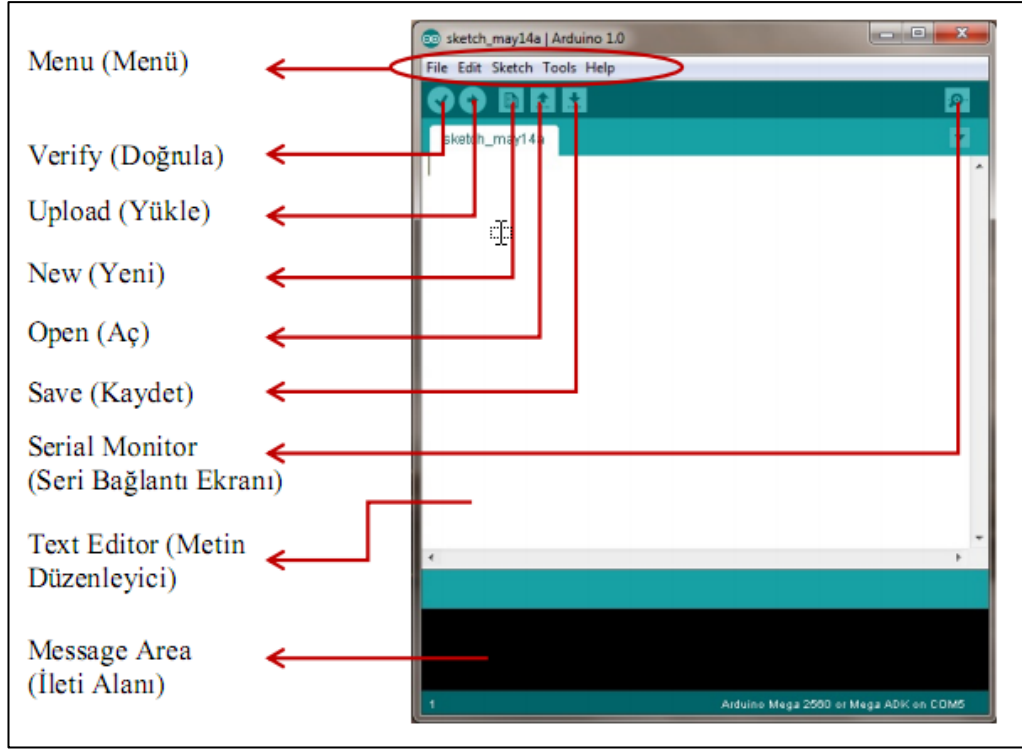
13. pin LED bağlantısı mevcuttur. 13. pinden alınan çıkış 1 olunca (High) LED yanmakta, 0 olunca da (Low) LED sönmektedir.

AREF piniyle analogReference() komutuyla analog girişlerin referans gerilim aralığı değiştirilebilmektedir.

Reset butonu işlemciyi sıfırlamak için kullanılmaktadır.

4.2 Arduino IDE Yazılımı

Arduino geliştirme ortamı proje geliştirmek için ihtiyacımız olan kodlama editörü, derleyici, program yükleyici gibi bileşenleri üzerinde barındıran komple bir yazılımdır. Arduino IDE yazılım dili C tabanlı bir dil olan Wiring’dır. Arduino IDE arayüzü Şekil 4.4 ‘deki gibidir.



Şekil 4.4: Arduino IDE arayüzü [4].

Verify (Doğrula), kodları derleyerek hata olup olmadığını kontrol eder.

Upload (Yükle), kodları derler ve Arduino kartına yükler. Eğer harici bir programlayıcı kullanılacaksa File menüsünden “ Upload Using Programmer ” ya da Shift tuşuyla Upload butonuna tıklanır.

New (Yeni), yeni bir sketch (boş sayfa) oluşturur.

Open (Aç), önceden oluşturulan sketch’leri açmak için kullanılır.

Save (Kaydet), sketch’i kaydeder.

Serial Monitor (Seri Bağlantı Ekranı), seri haberleşmeden gelen giden verileri görmeyi sağlayan Arduino IDE’nin kendi serial görüntüleme ekranını açar.

Text Editor (Metin Düzenleyici), Arduino’ya yazılan programlar (sketch) metin düzenleyicide yazılır.

Message Area (İleti Alanı), hataları, uyarıların detaylarını ve varsa düzeltilmesi gerekenleri gösterir.

Menu (Menü), içeriğinde File, Edit, Sketch, Tools ve Help menülerini barındıran ve Arduino IDE’yi yapılandırmaya yarayan içeriklere sahip alandır.

➤ **File menüsü**

Bu bölümde genel bir kelime işlemci programından alışık olunan yeni dosya, açma, varolan dosyayı açma, kaydetme, kapatma, yazıcıdan çıktı alma gibi dosya işlemleri bulunmaktadır. Ayrıca Arduino ile birlikte gelen örnek programlara da bu menüden ulaşılabilir ve programın karta yüklenmesi işlemleri gerçekleştirilebilir.

New: yeni bir Arduino program dosyası (sketch) açar.

Open: varolan bir program dosyasını açar.

Sketchbook: Preferences (Tercihler) menüsünde belirtilen ve Arduino programlarının kaydedildiği klasördeki program listesini görüntüler.

Examples: Arduino ile birlikte gelen örnek projelerin bulunduğu menüdür.

Close: açık olan projeyi kapatır.

Save: üzerinde çalışılan program dosyasını kaydeder.

Save As: üzerinde çalışılan program dosyasını başka bir isimle kaydeder.

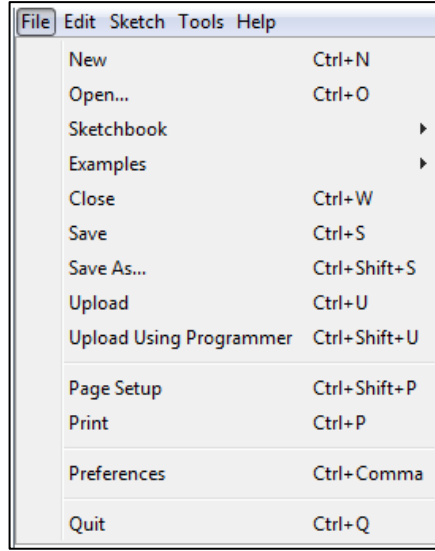
Upload to I/O Board: programın derlenip karta yüklenmesini sağlar.

Page Setup: yazıcı çıktısı almak için sayfa ayarlarının yapıldığı pencereyi açar.

Print: programın yazıcıdan çıktısını alır.

Preferences: Arduino geliştirme ortamı ile ilgili bazı ayarların yapıldığı pencereyi açar. Buradan Arduino programlarının kaydedileceği klasör seçimi, Arduino editörünün font büyüklüğünün ayarlanması, derleme ve yükleme esnasında bütün mesajların görüntülenip görüntülenemeyeceğinin seçimi gibi işlemler yapılabilir.

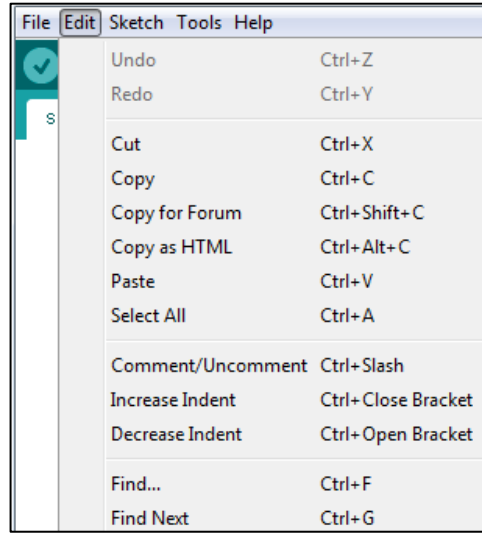
Quit: Arduino geliştirme ortamını kapatır.



Şekil 4.5: File menüsü

➤ Edit menüsü

Bu menüde de yine kelime işlemci programlara benzer olarak yapılan son işlemi geri alma veya tekrar gerçekleştirme, metin kesme, kopyalama, yapıştırma, seçili hale getirme işlemleri, kodların başlangıcına girinti ekleme veya kaldırma, kelime aratma gibi metin düzenleme komutları yer alır.



Şekil 4.6: Edit menüsü

Undo: kod editörü üzerinde yapılan son işlemlerin geri alınmasını sağlar.

Redo: son geri alınan işlemi tekrar gerçekleştirir.

Cut: seçili metin parçasını keserek hafızaya alır.

Copy: seçili metin parçasını hafızaya kopyalar.

Copy for Forum: forumlarda paylaşmak üzere renk kodlarıyla birlikte hafızaya alır.

Copy as HTML: web sitelerinde kullanılmak üzere kod renklendirmelerini HTML kodlarıyla birlikte hafızaya alır.

Paste: hafızadaki metni imlecin bulunduğu noktaya yapıştırır.

Select All: kod penceresindeki bütün herşeyi seçili hale getirir.

Comment/Uncomment: otomatik olarak bir satırın yorum haline getirilmesini veya eski haline dönmesini sağlar.

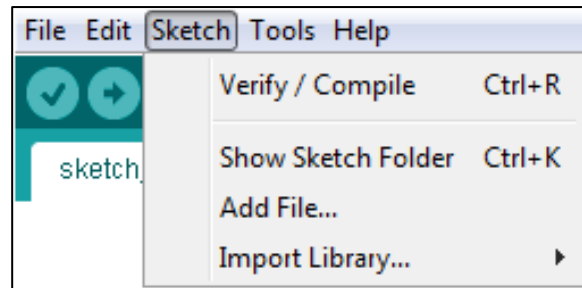
Increase Indent: satır başı boşluğu ekler.

Decrease Indent: satır başı boşluğunu azaltır.

Find: kod içerisinde metin araması yapma penceresini açar.

Find Next: en son aranılmış metnin başka bir örneğini arar.

➤ Sketch menüsü



Şekil 4.7: Sketch menüsü

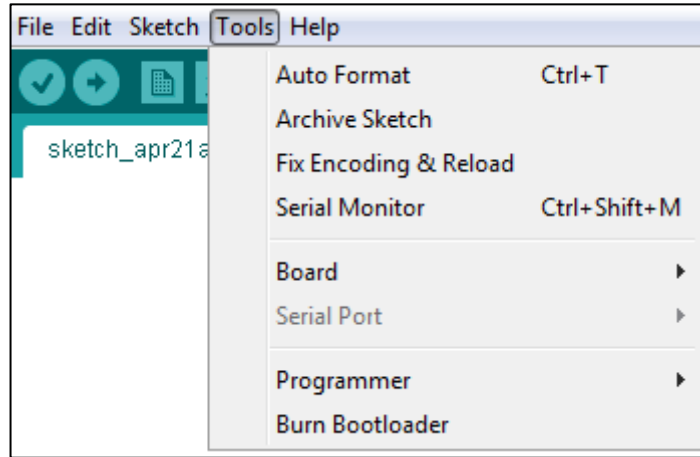
Verify/Compile: kodu derler.

Show Sketch Folder: programların kaydedildiği klasörü açar.

Add File: programa başka bir program dosyası eklemek için dosya açma diyalog kutusunu görüntüler.

Import Library: programda kullanılacak kütüphaneler ile ilgili başlık dosyasını programın başına ekler (.h uzantılı).

➤ Tools menüsü



Şekil 4.8: Tool menüsü

Auto Format: koddaki girintiler ve boşlukları okunabilirliğini artıracak şekilde ayarlar.

Archive Sketch: programı sıkıştırılmış zip dosyası olarak kaydeder.

Fix Encoding & Reload: programda karakter kodlamalarıyla ilgili bir problem varsa bunu düzelterek program dosyasını tekrar açar.

Serial Monitor: seri iletişim üzerinden gelen verileri görüntüler.

Board, kullanılan Arduino board tipi burdan seçilir. Sistem için bu sekmeden Arduino Mega 2560 modeli seçilmiştir.

Serial Port: bilgisayarda Arduino ile haberleşmek için oluşturulmuş COM portun numarasıdır. C# arayüzü ile seri haberleşmek için C# programına da aynı COM port tanımlanmalıdır.

Burn Bootloader: Arduino karta yeni bir Atmega mikrodenetleyici takıldığında önyükleyici (bootloader) programın yüklenmesini sağlar.

➤ Help menüsü

Help menüsünden Arduino'nun web sitesinden kullanıcıların ulaşabilecekleri bilgi sayfalarının offline (çevrimdışı) versiyonlarına atanmış linkleri bulunmaktadır.

4.3 Mikrokontrolörün Programlanması

Bir problemi çözmek veya belirli bir amaca ulaşmak için çizilen yola algoritma denir. Algoritma, bir başlangıç durumundan başladığında, açıkça belirlenmiş bir son durumunda sonlanan, sonlu işlemler kümesidir. Tüm programlama dillerinin temeli algoritmaya dayanır. Wankel motorunun ateşleme ve püskürtme sistemlerinin kontrolü için de mikrokontrolcünün stabil bir şekilde izlemesi planlanan bir algoritma hazırlanmıştır.

Mikrokontrolcü Algoritması:

1. Sıkıştırma ÜÖN referans olarak alınır.
2. Bu referans 0 krank mili açısı olarak kabul edilir.
3. Ateşleme avansı, dwell süresi, püskürtme avansı ve püskürtme süresi önceden belirlendiği değerde dönmeye başlar.
4. Ateşleme avansı, dwell süresi, püskürtme avansı ve püskürtme süresi için gelecek komutlar için seri haberleşme sürekli alıma açık olarak bekler.
5. Enkoderden gelen palslar, sıfır değeri sıkıştırma sonu üst ölü noktası olacak şekilde sayılmaya başlanır.
6. Püskürtme ve ateşleme noktaları, ilk 3600 palsın içindedir, püskürtme ve ateşleme gerçekleştirilir. Bu sırada motor devir bilgisi seri haberleşme ile COM porttan bilgisayara gönderilir.
7. Sıkıştırma sonu üst ölü noktası geldiğinde mikroişlemci enkoder sayıcı değerini sıfırlar.
8. Seri haberleşme ile yeni değişken değerleri (Ateşleme avansı, dwell süresi, püskürtme avansı ve püskürtme süresi) gönderildiğinde bir sonraki çevrim yeni gelen değerlerle gerçekleşir.

Mikrokontrolcü algoritmasının içeriği şu şekildedir. Enkoder krank miline bağlanmıştır. Krank milinin her bir turunda ateşleme ve püskürtme olması için, enkoderin 0. palsı sıkıştırma sonu üst ölü noktası olarak kabul edilir ve enkoderden gelen palsler sayılmaya başlanır. Sayıcı önceden belirlenen değerlere geldiğinde sırasıyla püskürtme ve ateşleme işlemleri gerçekleştirilir. Püskürtme ve ateşleme işlemleri ilk 3600 palsın içinde gerçekleşmiştir. Tekrar sıkıştırma üst ölü noktasına gelindiğinde Z ucundan alınan sinyal ile enkoder kendini sıfırlar ve yeniden saymaya

başlar. Yeni çevrim, püskürtme ve ateşleme ile ilgili değişkenler değiştirilmiş ise yeni değerlerle, değiştirilmez ise bir önceki değerlerle gerçekleştirilir.

```
TCCR5A=0;
TCCR5B=0b00000111;

void sifirlama()
{
    TCNT5=0;
}
```

Şekil 4.9: Timer/Counter kurulumu ve sayıcıyı sıfırlamak için yazılan kod

Timer'lar(zaman sayıcı) counter register denen bir register'a sahiptir. Bu register'ın görevi Timer'ın sayacı, yani counter (sayıcı) olarak çalışmaktır. Timer her adımda maksimum değerine ulaşınca kadar bu sayacı arttırır. Sayaç maksimum değeri aşıldığında değeri sıfırlanır. Bu işlem sistemde arttırımsal enkoderin Z ucundan alınan sinyal ile TCNT5 sayıcı değeri sıfırlanarak gerçekleştirilecektir.

Düzenli aralıklarla sayaç değerini arttırmak amacıyla timer bir clock kaynağına yani osilatöre ihtiyaç duyar. Osilatör tekrar eden sabit frekansta sinyal üretir. Timer her clock palsi aldığı anda sayacını bir arttırır. Timer'lar osilatöre bağlıdır. Ölçeklenebilir tekrar eden en küçük clock değerine periyot denir. 1Mhz osilatör kullanıldığında her 1 mikrosaniyede timer sayacını bir arttırır. Arduino kartında 16Mhz osilatör kullanıldığından periyot süresi yaklaşık olarak 63ns'dir [12].

Arduino Mega'da Timer0, Timer1, Timer2, Timer3, Timer4 ve Timer5 olmak üzere 6 adet timer interrupt'ı bulunmaktadır. Timer0 ve Timer2 8 bitlik olduğu için 0-255 arası değerler verilerek kullanılabilir. Diğer timer'lar 16 bitliktir ve 0-65535 arası değer verilerek kullanılabilir. Enkoderimiz 3600 pals saydığından 8 bitlik timer'lar yetersiz kalacaktır. Dolayısıyla sayıcı olarak 16 bitlik timer'lardan Timer5 seçilerek kullanılmıştır.

Arduino IDE programlanırken her mikrokontrolcü programlamada olduğu gibi önce değişken tipleri sınıflandırılmıştır.

```

int a;
int b;
int c;
int d;
int e;
unsigned long counter
boolean stringComplete=false;
String inputString = "";
int dwellbasla1=3100;
int ateslemebasla1=3150;
int enjektorbasa1= 2200;
int enjektorsuresil= 3000;
int tek_atim_enjektora1=0;

```

Şekil 4.10: Değişken tanımlamaları ile ilgili yazılan kodun bir parçası

Çizelge 4.2: Değişken Tablosu [13].

Tip	Boyut(byte)	Aralık	Açıklama
İnt	2	-32768 ile 32767	İşaretli tamsayılar
Float	4	-3,4+38 ile 3,4+38	Ondalık sayılar
Unsigned long	4	0 ile	Çok büyük pozitif tamsayılar
Boolean	1	4294967295 aralığı True(1) veya False(0)	Mantık ifadelerinde kullanılır
String	Özel karakter dizisidir, sayısal veri olarak işlem yapılmazlar.		

a, b, c,...k, l değişkenleri, tasarlanan arayüzden seri haberleşme ile gelen verilerin (12'lik paketler halinde) her birinin atandıkları integer tipindeki değişkenlerdir.

counter değişkeni enkoderden gelen palsları sayan değişkendir. Aslında maksimum değeri 3600 olduğundan integer olarak tanımlanabilir. Ancak sistemde daha yüksek çözünürlüklü bir enkoder kullanılabime veya kodun çevrim atlatma vb. uygulamalar için kullanılabilme ihtimaline karşılık 0 ile çok yüksek pozitif tam sayılar arasında tanımlanan unsigned long değişkeni olarak kullanılmasına karar verilmiştir.

dwellbasla, ateslemebasla değişkenleri integer sınırlarında kaldığından o şekilde tanımlanmıştır. 2 adet buji için dwellbasla1, dwellbasla2 ve ateslemebasla1, ateslemebasla2 şeklinde her biri bir bujiyi temsil edecek şekilde ikişer adet değişken kümesi tanımlanmıştır. dwellbasla değişkeni, arayüzden aç olarak girilen ve dwell

süresinin sıkıştırma üst ölü noktasından kaç derece önce başlayacağını belirleyen değişkendir. Ateslemebasla değişkeni ise ateslemenin sıkıştırma üst ölü noktasından kaç derece önce başlayacağını gösteren değişkendir.

enjektorbasla, enjektorsuresi değişkenleri de ateşleme değişkenleri gibi integer olarak tanımlanmıştır. 4 adet enjektör için enjektörbasla1, enjektorsuresi1, enjektörbasla2, enjektorsuresi2, enjektörbasla3, enjektorsuresi3 ve enjektörbasla4, enjektorsuresi4 olmak üzere her bir enjektörü temsil edecek şekilde toplamda 8 ayrı değişken tanımlanmıştır. Enjektörbasla değişkeni püskürtme işleminin başladığı anı, enjektorsuresi de püskürtme işleminin kaç ms süreceğini belirten değişkenlerdir.

stringcomplete değişkeni sadece True(1) – False(0) değerlerini aldığından boolean olarak tanımlanmıştır.

tek_atim_enjektör, püskürtme işleminin her turda sadece bir defa gerçekleştirilmesini garantilemek için oluşturulmuş değişkendir. Sayısal değeri küçük olduğundan integer olarak tanımlanmıştır.

Püskürtme işlemi süre ile belirtilmiş ve daha önceki çalışmalarda [4] karşılaşılan sürekli püskürtme gibi bir problemle karşılaşmamak amacıyla tek_atim_enjektör adında bir değişken tanımlanmıştır. Bu değişkenin amacı püskürtmeyi sadece belirlenen zamanlarda aktive etmesi ve çevrimin diğer noktalarında devreden çıkarmasıdır. İlgili kodun bir bölümü Şekil 4.11 de gösterilmiştir.

```
tek_atim_enjektör1=1;
}
if(counter >= 0 && counter <= 2 )tek_atim_enjektör1=0;
```

Şekil 4.11: Sürekli püskürtmenin önüne geçmek için yazılmış kodun bir parçası

Yazılan kontrol programında ateşleme işlemi için bobinin dolma süresi olan dwell süresi krank mili açısı cinsinden ifade edilecektir. Dwell süresi çoğu bobin için önerilen sabit bir değer olmakla birlikte motor devri ile birlikte değişmesi ve her devirde motor verimi açısından ideal bir değer alması beklenmektedir. Yard.Doç.Dr Akın Kutlar ile yapılan kişisel görüşmeler sonucunda dwell süresini değişen motor devri ile birlikte değiştirip ideal dwell süresini bulmak için bu sürenin krank mili açısı cinsinden tanımlanması kararı alınmıştır.

Püskürtme işlemi süre bazlı çalışmaktadır. Püskürtmenin başlangıcı ve bitişi arasında geçen süre ms (milisaniye) mertebesinde ve delay (gecikme) komutu ile kontrol edilir. Püskürtme sürelerinin değişiminin deneyler esnasında ondalık değerler ile ifade edilmesi gerekmektedir. Dolayısıyla püskürtme değişkenlerinin Çizelge 4.1’ deki float komutuyla tanımlanması gerektiği düşünülebilir. Ancak delay komutunun içine sadece integer (tamsayı) değişkenler yazılabildiğinden püskürtme süreleri delayMicroseconds komutu ile µs(mikrosaniye) cinsinden yazılmıştır. Ateşleme ve püskürtme işlemleri ile ilgili yazılan kodların bir bölümü Şekil 4.12’ de belirtilmiştir.

```
if(counter >= dwellbasla2 && counter<=ateslemebasla2)
{
    digitalWrite(atesleme2,HIGH);
}
else
{
    digitalWrite(atesleme2,LOW);
}

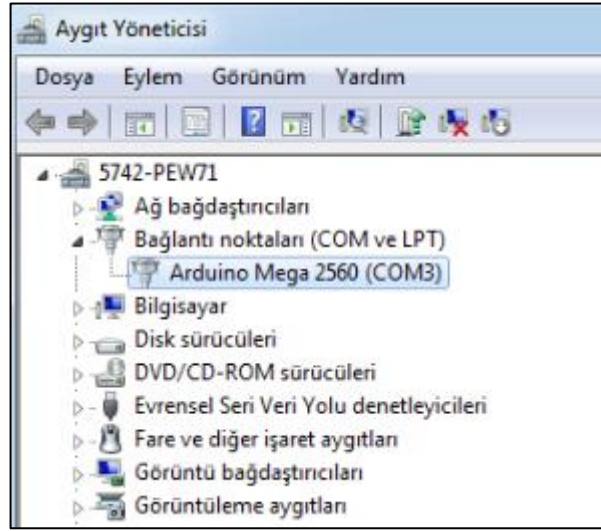
if(counter >= enjektörbasla1 && !tek_atim_enjektör1)
{
    digitalWrite(enjektör1,HIGH);
    delayMicroseconds(enjektörsuresil);
    digitalWrite(enjektör1,LOW);
}
```

Şekil 4.12: Ateşleme ve püskürtme sürelerini kontrol etmek için yazılan kod

Elektronik birimler aralarında iletişim kurmak istediklerinde birbirleriyle çeşitli haberleşme protokolleri üzerinden haberleşirler. Protokol iki veya daha fazla birimin kendi aralarında haberleşmekte kullandıkları kurallar bütünüdür. Dijital haberleşmede temel olarak iki yöntem bulunmaktadır. Bu ikisi seri ve paralel haberleşmedir. Seri haberleşmede dijital veriler (bitler) tek bir hat üzerinden peşi sıra gönderilirken paralel haberleşmede ise belirli gruplar halinde eş zamanlı olarak iletilirler. Daha az veri hattı gerektirdiğinden seri haberleşme elektronik projelerde çok sık kullanılır. Günümüz bilgisayarlarındaki USB, SATA vs. gibi terimler seri haberleşme standartlarına birer örnektir. Arduino ile bilgisayarı haberleştirmekte seri haberleşme birimi kullanılmıştır [13].

Normalde Arduino kartları bilgisayara 9 pinli RS232 seri portu üzerinden bağlanacak şekilde donatılmıştır. Ancak günümüzde USB’nin daha yaygın kullanımından dolayı Arduino, USB üzerinden de bağlantı sağlamak için RS232-USB dönüştürücü aparatı kullanır. Arduino Mega mikrodenetleyicisi seri port donanımına sahiptir. TX

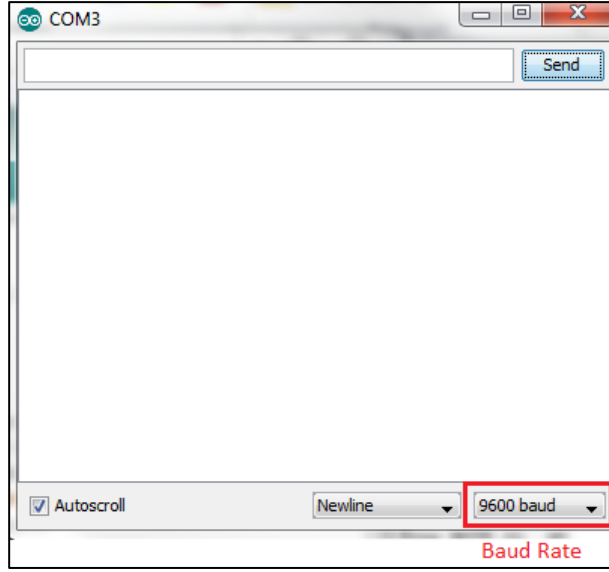
(Transmit) ve RX (Receive) pinleri ile sayısal iletişim sağlanır. RS232 kullanılarak doğrudan bilgisayara bağlantı sağlayabilir. USB ile seri iletişimi sağlamak için ise RS232-USB dönüştürücü ikinci bir mikrodnetleyici kullanır [12]. Bu dönüştürücünün Arduino kart içindeki konumu Bölüm 3.4 Şekil 3.4 de verilmiştir. İkinci bir mikrodnetleyici kullanılması mikrodnetleyicinin tüm özelliklerinin programlanabilir olmasına ve cihazın farklı bir sanal seri port üzerinde tanıtılmasına imkan verir. Böylece seri port çıkışı olmayan bilgisayarlarda USB üzerinden sanal bir seri port (COMx) yaratılır ve haberleşme bu seri port üzerinden gerçekleştirilir. Kullanılan COM portun hangisi olduğu bilgisayarın Aygıt Yöneticisinden veya Arduino IDE Tools>Serial Port menüsünden kontrol edilmeli ve arayüz yazılımıyla aynı COM portun kullanıldığına dikkat edilmelidir.



Şekil 4.13: Aygıt yöneticisinden Arduino’nun seri haberleşme için COM portu

Seri haberleşme için Arduino üzerinde bulunan seri haberleşme ünitesine UART (Universal asynchronous receiver/transmitter: Evrensel asenkron alıcı/verici) adı verilir. Arduino Mega modelinde 3 adet UART bulunmaktadır.

Seri haberleşme için gönderme TX ve alma RX olmak üzere iki pin kullanılır. (D0, D1 pinleri) Arduino Mega üzerinde bu pinler ayrıca TX, RX ledlerine ve USB dönüştürücü ikinci mikrodnetleyiciye de bağlıdır. Programlama esnasında Arduino ile PC arasındaki iletişim bu pinler üzerinden sağlanır. Arduino geliştirme ortamındaki Serial Monitor aracı Arduino ile seri haberleşme üzerinden veri alışverişinde kullanılır. Serial Monitör’den gelen veriler Arduino’ya iletilir. Aynı şekilde Arduino’dan gönderilen veriler de Serial Monitör penceresinden görüntülenebilir.



Şekil 4.14: Serial Monitor’un görünümü

Serial Monitör’de sol alttaki kutucuk işaretlendiğinde (autoscroll) yeni veriler geldikçe pencere aşağı doğru kaydırılır. Sağ alttaki seçim kutusunda ise iletişim hızı (Baud Rate) seçilir. Buradaki hız değeri Arduino IDE programında yazılan Serial.begin() fonksiyonu ile belirlenen değerle aynı olmalıdır. Aynı zamanda Arduino ile haberleşen programın veya cihazın kabul ettiği iletişim hızı da Arduino’nunkine eşit olmalıdır [12]. Standart seri haberleşme hızları içinden Arduino’nun desteklediği maksimum hız değeri olan 115.200 seçilmiştir. Bu değerlerin birimi bps (bit per second)’dir. Her bir bayt için 10 bit gönderildiğinden gönderilen bayt miktarı bu değerlerin 10’a bölümüyle hesaplanır. 115.200 bps hızında saniyede 11520, milisaniyede ise yaklaşık 11 karakter gönderilmektedir.

```
Serial.begin(115200);  
Serial.println(counter);
```

Şekil 4.15: Seri haberleşmenin başlatılması ve verilerin serial monitör ile gösterimi

Yazılan seri haberleşme algoritmasında arayüzden Arduino’ya COM port vasıtasıyla gönderilecek veri paketi örneği (1 buji ve 1 enjektör için) aşağıdaki gibidir.

“dwellbasla1:ateslemebasla1:enjektörbasla1:enjektörsuresi1:\n”

Sayısal olarak ifade etmek gerekirse;

“3200:3400:2000:2500:\n”

```

while (Serial.available())
{
    char inChar = (char)Serial.read();
    inputString += inChar;

    if (inChar == '\n')
    {
        stringComplete = true;
    }
}

```

Şekil 4.16: Arayüzden gelen verinin alınması ile ilgili yazılan kod

Arayüzden gelen byte'lar inputString değişkenine atanır. Karakter halinde gelen bu değişkenler için “\n” (satır sonu) karakteri görüldüğünde inputString değişkeni, yani veri alma işlemi tamamlanmış demektir. 2 buji ve 4 enjektörün hepsi için ikişer adet değişken olacağından veriler 12'lik paketler halinde gelmektedir. Değişkenler “:” karakteri ile birbirinden ayrılmıştır. “:” karakteri gelinceye kadar kadarki byte'lar string olarak kaydedilmiştir. Her iki “:” karakteri arasındaki stringler integer'a çevrilip bir sayısal değişkene atanmıştır.

```

if(stringComplete)
{
    a = inputString.indexOf(":");
    b = inputString.indexOf(":", a + 1);
    c = inputString.indexOf(":", b + 1);
    d = inputString.indexOf(":", c + 1);
    e = inputString.indexOf(":", d + 1);
    f = inputString.indexOf(":", e + 1);
    g = inputString.indexOf(":", f + 1);
    h = inputString.indexOf(":", g + 1);
    i = inputString.indexOf(":", h + 1);
    j = inputString.indexOf(":", i + 1);
    k = inputString.indexOf(":", j + 1);
    l = inputString.indexOf(":", k + 1);
}

```

Şekil 4.17: Paket halinde gelen verilerin değişkenlere atanması için kodun bir kısmı

```
String Dwell_baslal=inputString.substring(0, a);
dwellbaslal=Dwell_baslal.toInt();
String Ates_baslal=inputString.substring(a + 1, b);
ateslemebaslal=Ates_baslal.toInt();
String Enjektör_baslal=inputString.substring(b + 1, c);
enjektörbaslal=Enjektör_baslal.toInt();
String Enjektör_surel=inputString.substring(c + 1, d);
enjektörsuresil=Enjektör_surel.toInt();
```

Şekil 4.18: Gelen string halindeki karakterlerin, integer'a çevrildiği kodun bir kısmı

Enjektör debi deneyi için de Arduino kart kullanılmıştır. Belirli püskürtme sürelerinde (ms) deney enjektörünün 1.000 püskürtmesinde elde edilen toplam yakıttan bir püskürtmedeki yakıt miktarının hesaplanması amaçlanmıştır.

```
void loop () {

    if(a == 0)
    {
        for(int i=1; i<1001; i++){
            digitalWrite(10, HIGH);
            delayMicroseconds(2000);
            //delay(2);
            digitalWrite(10, LOW);
            //delayMicroseconds(58000);
            delay(58);
        }
        a = 1;
    }
}
```

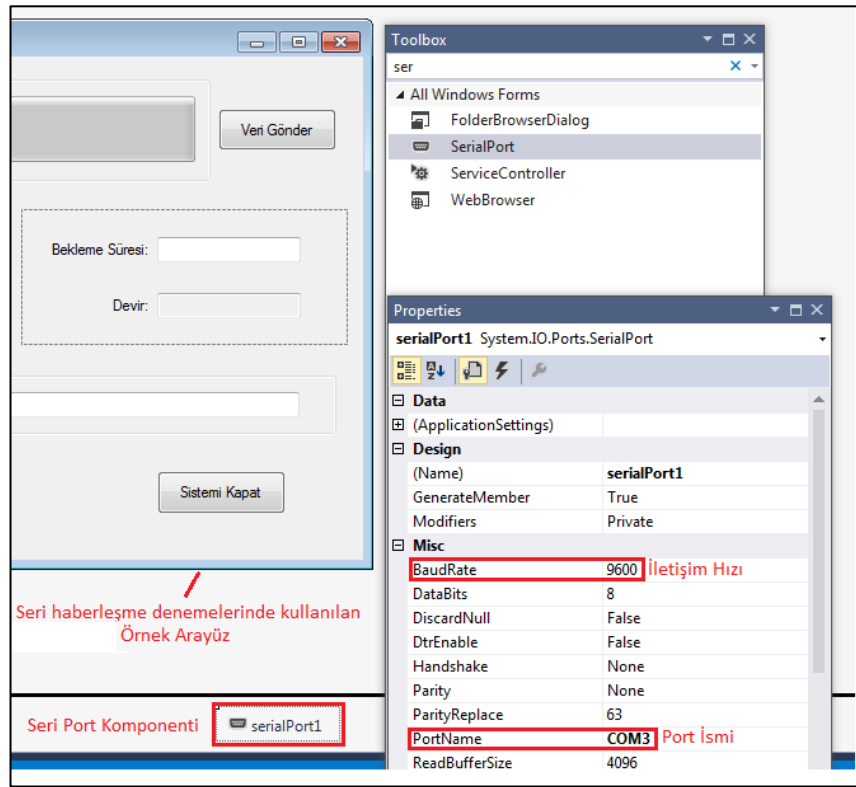
Şekil 4.19: 2 ms püskürtme süresiyle dakikada 1.000 püskürtme için yazılan kod

4.4 C# Kontrol Arayüzü Yazılımı

Ateşleme ve püskürtme sürelerini bilgisayardan kontrol edebilmek için bir arayüz tasarımına gerek duyulmaktadır. Bu doğrultuda yapılan çalışmalarda Arduino ile ilgili uygulamalarda yaygın olarak kullanılan C# programı tercih edilmiştir. C# dili Microsoft'un .net platform için tasarladığı, C++ ve java dillerinin bir çok iyi ve güçlü yönlerini kullanarak, object oriented programming (nesne tabanlı programlama) için ideal olan dillerden olan, internet, mobil ve masaüstü uygulamalara çözüm üreten bir programlama dilidir.

4.4.1 C# yazılımı ile arduino'nun seri haberleşmesi

C# ile geliştirilen arayüzün Arduino ile iletişimi seri port vasıtasıyla sağlanmaktadır. Arduino Serial Monitor'u kullanarak gönderilen verileri C# arayüzünde oluşturacağı bir sanal seri port komponenti ile alabilir, aynı şekilde C# arayüzüne yazılan yeni değişkenleri Arduino mikrokontrolcüye gönderilebilir. Bunu yapmak için Toolbox (Araçlar) menüsünden SerialPort seçilir ve Properties (Özellikler) menüsünden kullanılacak seri port ile ilgili ayarlamalar yapılır. Daha önce de bahsedildiği gibi ayarlamalar yapılırken seçilen port isminin (COMx) ve iletişim hızının (Baud Rate) Arduino yazılımında seçilen ile aynı olması gerekmektedir. Aksi takdirde seri iletişim sağlanamaz.



Şekil 4.20: Örnek arayüzde seri portun kurulması

Daha önce de bahsedildiği gibi seri haberleşme algoritması şu şekildedir.

```
“dwellbasla1:ateslemebasla1:enjektorbasla1:enjektorsuresi1:  
dwellbasla2:ateslemebasla2:enjektorbasla2:enjektorsuresi2:  
enjektorbasla3:enjektorsuresi3::enjektorbasla4:enjektorsuresi4:\n”
```

Arduino ile kişisel bilgisayar arasındaki seri iletişim algoritması, gönderilen sayısal verileri “\n” (satır sonu) işareti gelene kadar string şeklinde veri olarak kaydetme mantığına dayanır. Her “:” işareti görüldüğündeki veri de substring olarak tek tek

püskürtme ve ateşleme değişkenlerine atanacak ve ardından integer'a çevrilerek işlenecektir.

Integer'a çevrilmiş sayısal değerlere bakılacak olursa; (1 buji ve 1 enjektör için)

“3200:3400:2000:2500:\n”

şeklinde olmaktadır.

dwellbasla: Dwell süresinin sıkıştırma üst ölü noktasından kaç krank mili açısı önce başlayacağı anlamına gelen enkoderin pals değeridir. Örneğin 3.200 yazıldığında;

$$360 - \frac{3200}{3600} \times 360 = 40^\circ \text{ krank mili açısı}$$

Dwell süresi sıkıştırma üst ölü noktasından 40° önce başlayacaktır.

ateslemebasla: Ateşleme avansıdır. Ateşleme işleminin sıkıştırma üst ölü noktasından kaç derece önce başladığını gösteren pals değeridir. Örneğin 3.400 yazıldığında;

$$360 - \frac{3400}{3600} \times 360 = 20^\circ \text{ krank mili açısı}$$

Ateşleme işlemi sıkıştırma üst ölü noktasından 20° önce başlayacaktır. Dwell süresi dwellbasla ile ateslemebasla değişkenleri arasındaki süreye tekabül etmektedir. Dwell süresi bittiği anda ateşleme başlayacaktır. Yukarıdaki örnekte dwell süresi 200 pals, yani 20° krank mili açısıdır.

enjektörbasla: Püskürtme işleminin sıkıştırma üst ölü noktasından kaç derece önce başladığını gösteren pals değeridir. Örneğin 2.000 yazıldığında;

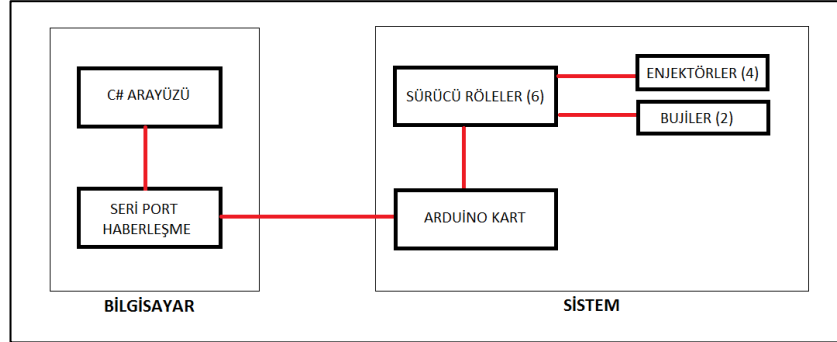
$$360 - \frac{2000}{3600} \times 360 = 160^\circ \text{ krank mili açısı}$$

Püskürtme işlemi sıkıştırma üst ölü noktasından 160° önce başlar.

enjektorms: Püskürtme işleminin kaç mikrosaniye boyunca başlayacağını gösteren değerdir. Arduino delay komutuna ondalık sayılar yazılamadığı için

delayMicroseconds komutu ile mikrosaniye mertebesinde değerler atanacaktır. Örneğin 2.500 yazıldığında;

$$2500 \times 10^{-3} = 2.5 \text{ ms püskürtme süresince enjektör açık kalacaktır.}$$



Şekil 4.21: Bilgisayar ile sistemin haberleşmesi

4.4.2 C# kontrol arayüzünün tanıtılması

Kontrol arayüzü mümkün olduğu kadar sade ve kullanıcı tarafından anlaşılabilir şekilde tasarlanmıştır. Arayüzde 2 adet buton bulunmaktadır. Veri Gönder butonu ateşleme ve püskürtme değişkenleri değiştirildiğinde Arduino'ya yeni verileri göndermek amacıyla kullanılmıştır. Sistemi Kapat butonu ise Arduino ile yapılan seri iletişimi, bununla birlikte tüm sistemi kapatmaya yarar.

Dwell süresi, ateşleme avansı, püskürtme başlangıcı ve püskürtme süresi değerleri NumericUpDown kutucuklarıyla kullanıcıya sunulmuştur. Bu kutucuklara elle değer girilebildiği gibi, yanlarındaki oklarla birer birer artırılıp azaltılabilmektedirler. Dwell başlangıcı, ateşleme avansı ve püskürtme başlangıcı değerleri kullanıcı tarafından krank mili açısı cinsinden 1° hassasiyet ile belirlenebileceklerdir. Püskürtme süresi ise 0,01 ms hassasiyetle değiştirilebilecektir.

TextBox ile oluşturulan Gelen Veri bölümüne enkoderden alınan motor devir bilgisi yazılır. ProgressBar isimli gösterge bölümü de Gelen Veri bölümünden alınan devir bilgisini 200 ms hassasiyetle kullanıcıya sunar. Doktora öğrencisi Yük.Müh. Ömer Cihan ile yapılan kişisel görüşmede Wankel deney motorunun maksimum 6000 d/d'da çalışacağı bilgisi alınmış ve bu bilgi ışığında devir göstergesi ölçeklendirilmiştir.

Şekil 4.22: C# arayüzü

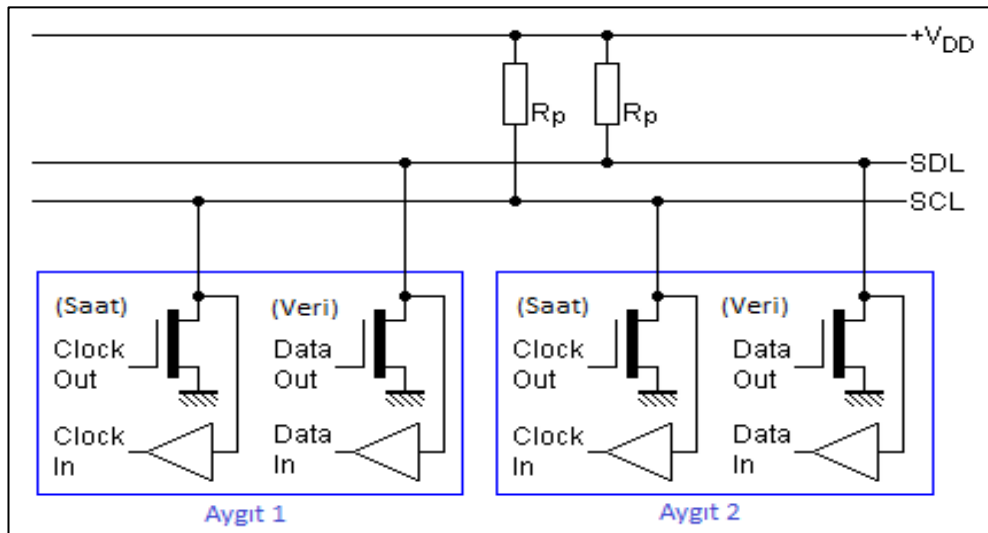
Giden Veri bölümü yine TextBox ile oluşturulmuş, ateşleme ve püskürtme değişkenleri değiştirildikten sonraki son hallerinin veri gönderilirken gözlemlenmesi amaçlanmıştır. COM Port kutucuğu ile, arduino ile bilgisayarın haberleşmesini sağlayan sanal seri portun bilgisayardaki adı olan ve her bilgisayarda farklı bir adrese tanımlanan COM Port'un kullanıcı tarafından arayüzden seçilebilmesi amaçlanmıştır. Baudrate kutucuğu ile arduino iletişim hızı seçilebilmektedir. İleride farklı pals sayısında bir enkoder ile çalışılabilme ihtimaline karşı, enkoder pals sayısının kullanıcı tarafından 1 pals hassasiyetle girilebileceği bir enkoder pals sayısı kutucuğu oluşturulmuştur.

Sistem şu şekilde çalışır. Dwell süresi, ateşleme başlangıcı, püskürtme başlangıcı ve püskürtme süresi değerleri arayüzden anlık olarak sistem çalışır durumda olsa dahi değiştirilebilir. Değiştirilen veriler Giden Veri kısmında görüntülenir ve giden verilerin doğruluğu kontrol edilir. Yapılan değişiklikler Veri Gönder butonu ile Arduino mikrokontrolcü karta, Arduino'dan da sürücü Solid-State röleler yardımıyla ateşleme ve püskürtme elemanlarına iletilir. Aynı anda enkoderden alınan motor devir bilgisi Gelen veri kısmında ve buna bağlı olarak gösterge kısmında kullanıcıya sunulur. Sistem kapatılmak istendiğinde Sistemi Kapat butonu kullanılır ve hem seri haberleşme hem de tüm elektronik sistem elemanları deaktif hale gelmiş olur.

4.5 Eeprom Yazılımı Ve Arduino'ya Kurulumu

Birçok entegre ve elektronik komponent mikrodnetleyici ile iletişim kurmak için farklı dijital iletişim protokolü kullanır. Arduino, bu protokollerin çoğunu destekler. Arduino ile harici EEPROM iletişimi için en yaygın olarak kullanılan I2C veri yolu protokolüdür. I2C veriyolu güçlü, yüksek hızlı (400Kbps), iki yönlü iletişimi destekleyen cihazlar arasındaki iletişim için en az sayıda pin kullanılmasını sağlayan bir veri türüdür. I2C veriyolu bir master (genellikle mikrodnetleyici) cihaz tarafından kontrol edilir [12].

I2C (Inter Integrated Circuit) 1970'li yıllarda Philips firmasının yarıiletken bölümü tarafından devreler üzerindeki komponentlerin birbirleriyle kolayca haberleşebilmesi amacıyla geliştirilmiş bir seri haberleşme protokolüdür. I2C'de aynı veriyolu üzerine birden fazla birim bağlanabilir ve bu birimler birbirleriyle haberleşebilirler. Haberleşme için iki adet hatta sahip olduğundan 2 kablolu arabirim (Two Wire Interface) adıyla da anılan I2C protokolünde hatta bağlı her birimin bir adresi bulunur ve hattaki ana kontrolcü diğer birimlere bu adresler üzerinden erişir. Haberleşme veriyolu seri veri hattı (SDL-Serial Data Line) ve seri saat hattı (SCL-Serial Clock Line) olmak üzere iki hattın sağlanır. I2C birimlerinin çıkışları open-drain tipinde olduğundan her bir hatta 10Kohm'luk pull-up dirençleri bağlanması gerekir [13].

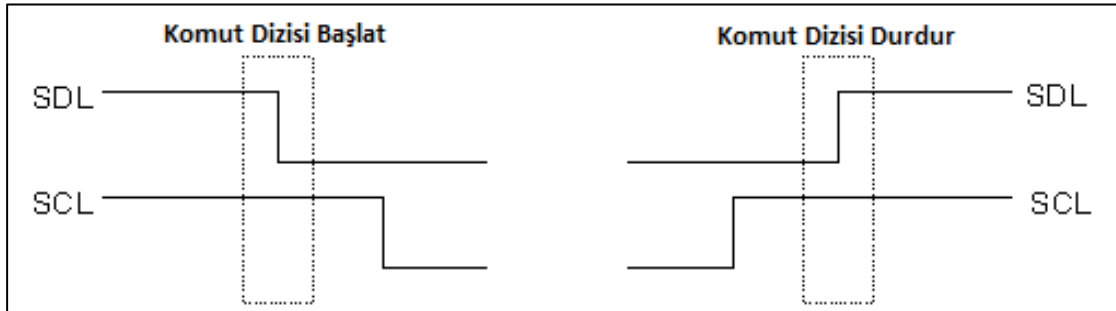


Şekil 4.23: I2C veriyolunun şematik görünümü

I2C teorik olarak biraz karışık gibi görünse de Arduino kütüphane fonksiyonlarıyla oldukça kolay bir şekilde kullanılabilir. I2C arabirimiyle haberleşen birçok sensör donanımı ve çevrebirim bulunmaktadır. Harici Analog-Dijital çeviriciler, giriş-çıkış çoklayıcılar, sıcaklık, hareket sensörleri bunlara örnek olarak verilebilir.

I2C haberleşmesinde ana (master) ve uydu (slave) modüller olmak üzere iki tip birim bulunur. Haberleşme ana modül tarafından başlatılır ve yönetilir. I2C haberleşmesi ana modülün başlangıç sinyalini veriyoluna iletmesiyle başlatılır. Böylelikle hat üzerindeki bütün modüller dinlemeye geçer. Ardından ana modül haberleşmek istediği uydu modülünün adresini veriyoluna yollar. Veriyolu üzerindeki her bir modülün kendine özgü bir adresi bulunur. Bu adres 7 bit uzunluğundadır. (8. Bit yazma veya okuma komut seçimi için kullanılır.) Kendi adresini veriyolu üzerinden okuyan uydu modül daha sonra ana modülün gönderdiği komutu alır. Bu komut bir kaydediciyi okuma veya yazma komutu olabilir. Komutun içeriği uydu modülünün tipine göre değişir. Örneğin EEPROM için bu belirli bir bellek adresinin yazılma ve okunma komutu olabilirken, bir analog dijital çevirici için çevrime başla komutu olabilir.

Master aygıtın slave aygıtlarla haberleşebilmesi için, veri yolu üzerinden bir komut dizisi gönderilmelidir. Bu komut dizisinin başlatılması ve sonlandırılması için SDL hattına veri gönderilirken, SCL hattının HIGH konumunda olması gerekir. Şekil 4.24'te SDL ve SCL hatlarından veri gönderimi gösterilmiştir.

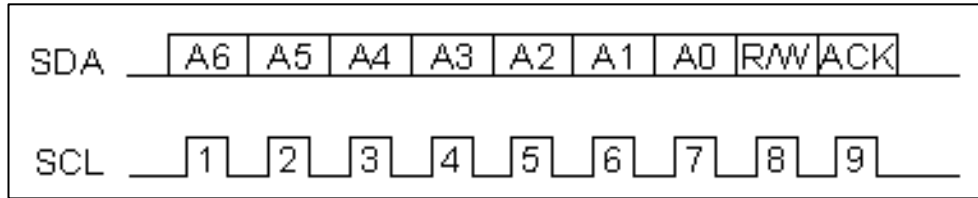


Şekil 4.24: SDL ve SCL hatlarından komut dizisi başlatma ve durdurma

Veri 8 bitlik dizi şeklinde gönderilir. Bu bitler SDL hattına en yüksek değerli bitten başlayarak gönderilir. SCL hattı ise HIGH/LOW yapılarak verinin gönderilmesini sağlar. SDL ve SCL hatlarındaki pull-up dirençleri bu hatlara gönderilen veri LOW olarak belirtilmediğinde HIGH konumuna çekmek için kullanılır. Her 8 bitlik veri aldığı anda slave aygıt, “ veriyi aldım” demek için 9. (ACK-Acknowledge) biti

gönderir. Böylece 8 bitlik (1 byte) veriyi gönderebilmek için 9 adet clock pulsı gerekir. Eğer slave aygıt 9. Bit (ACK) olarak LOW gönderirse, veriyi aldığını ve başka bir veri almaya hazır olduğunu söyler. Eğer slave aygıtının gönderdiği ACK biti, HIGH ise veri almadığını söyler. Bu durumda master aygıtının komut dizisi sonlandırılmalıdır.

Daha önce de bahsedildiği gibi I2C aygıtlarının her birinin 7 bitlik bir adresi vardır. Tüm aygıtların aynı hat üzerinde olması ve adres bilgisinin 7 bit olmasından dolayı hatta bağlanacak en fazla 127 slave aygıt olabilir. Master 7 bitlik adres gönderdiğinde slave 8 bitlik veri alır. Sonuncu bit (R/W) biti 0 olursa master, slave aygıtı veri yazılacağını, 1 olursa slave aygıttan veri okunacağını belirtir.



Şekil 4.25: Adres, R/W ve ACK bitlerinin gösterimi

Arduino ile I2C aygıtı arasındaki iletişim için gereken temel adımlar şöyledir:

- Master bir başlangıç biti gönderir. Tüm slave aygıtlar iletişim için hazır hale gelir.
- Master, iletişim kurmak istediği slave aygıtın 7 bitlik adresini gönderir.
- Master veri yazacaksa (0), veri okuyacaksa (1) olarak R/W bit değerini gönderir.
- Slave aygıt, veri aldığını onaylamak için ACK bitini 0 yapar.

İletişim tamamlandığında master aygıt bir stop biti göndererek iletişimi kapatır.

I2C fonksiyonlarının kullanılabilmesi için Wire kütüphanesi Arduino'ya eklenmelidir. Kütüphane internet ortamında indirildikten sonra Arduino IDE yazılımında şu şekilde çağırılır;

#include “wire.h”

I2C haberleşmesini başlatmak için setup() fonksiyonu içerisinde Wire.begin() fonksiyonu çağrılmalıdır. Bazı durumlarda Arduino’nun uydu modül olarak ayarlanması gerekebilir. Bu durumda begin() fonksiyonuna parametre olarak uydu modül adresi verilebilir.

I2C haberleşmesinde ana modül her zaman haberleşme işleminin başlangıcında veriyoluna başlangıç işaretini gönderir. Bunun için Wire.beginTransmission() fonksiyonu kullanılır. Bu fonksiyona parametre olarak haberleşme yapılacağı uydu modülünün adresi yazılır. I2C üzerinden gönderilecek veriler Wire.send() fonksiyonu ile tampona kaydedilir ve Wire.endTransmission() fonksiyonunun çağrılması ile gönderilir. Wire.endTransmission() fonksiyonu sonuç olarak 0 ile 4 arasındaki değerleri döndürür. Bu değerler şu anlama gelir:

- 0 : Veri gönderimi başarılı
- 1 : Gönderilmek istenen veri, gönderim tamponunun boyutunu aşıyor
- 2 : Adres gönderiminden sonra NACK (Not Acknowledge-Olumsuz cevap) alındı
- 3 : Veri gönderiminden sonra NACK (Not Acknowledge-Olumsuz cevap) alındı
- 4 : Başka bir hata oluştu

Ana modül uydu modüle veri alma isteğini Wire.requestFrom() fonksiyonu ile yollar. Bu fonksiyona ilgili uydu modülün adresi ve istenilen bayt sayısı parametre olarak verilir. Uydu modülden alınan bayt olup olmadığı Wire.available() fonksiyonu ile kontrol edilebilir.

Edinilen bilgiler ışığında ileride veri depolama ve kullanma çalışmalarına fayda sağlayacağı düşünüldükçe EEPROM’un Arduino’ya kurulumu, EEPROM’dan veri okumak ve yazmak ile ilgili ön çalışmalar yapılmış ve ASCII kodlarının EEPROM’a yazılması ve okunması ile ilgili bir örnek denemesi gerçekleştirilmiştir.

```
// EEPROM'a veri yazdırmak için:
void I2CEEPROM_Write (unsigned int bellek_adres, byte data)
{
    Wire.beginTransmission(EEPROM_ADRES); // Aygıtla iletişim başlatılıyor.
    Wire.write( (int)highByte(bellek_adres));
    Wire.write( (int)lowByte(bellek_adres));
    Wire.write (data); // byte türünden veri yazılıyor.
    Wire.endTransmission(); // iletişim sonlandırılıyor.
    delay(5); // I2C, EEPROM yazma görevi tamamlanana kadar 5ms bekle
}
```

Şekil 4.26: EEPROM'a veri yazdırmak için yazılan kodun bir parçası

```
//EEPROM'dan veri okumak için:
byte I2CEEPROM_Read (unsigned int bellek_adres)
{
    byte data; // bellekten okunacak verinin kaydedileceği değişken
    Wire.beginTransmission(EEPROM_ADRES);
    Wire.write( (int)highByte(bellek_adres));
    Wire.write( (int)lowByte(bellek_adres));
    Wire.endTransmission();
    Wire.requestFrom(EEPROM_ADRES, (byte)1);
    while(Wire.available() == 0 ); // veriyi bekle
    data = Wire.read();
    return data ;
}
```

Şekil 4.27: EEPROM'dan veri okumak için yazılan kodun bir parçası

```
// KURULUM AYARLARI

void setup()
{
    Serial.begin(9600);
    Wire.begin();

    Serial.println("EEPROM'A 1024 BYTE DEGER YAZILIYOR"); // EEPROM'A VERİ YAZILIYOR.
    for(int i=0; i<1024; i++)
    {
        I2CEEPROM_Write(i, KarakterDeger); // sonraki karaktere gidilir.
        KarakterDeger++;
        if(KarakterDeger == 126) // ASCII 126.karakterde başa dönülüyor.
            KarakterDeger=33; // Tekrar başlatılıyor.
    }
}
```

Şekil 4.28: Kurulum ayarlarını içeren kodun bir parçası

4.6 Ek Yazılımlar

Tez kapsamında tek rotorlu Wankel motoru haricinde laboratuarda bulunan çevrim atlatmalı tek silindirli 4 zamanlı benzinli bir motor için de kontrolcü prgoramı ve arayüzü güncellenerek kullanıma hazır hale getirilmiştir. Müh. Özgür Tekeli'nin yüksek lisans tezinde kullandığı ve Yük.Müh. Abdurrahman Demirci'nin doktora tezinde üzerinde deneyler yapacağı, dizelden benzinliye çevrilmiş Lombardini 3LD450 tek silindirli motor için hazırlanan algoritmadaki amaç, Özgür Tekeli'nin sistemindeki eksikleri gidermek ve çeşitli basitleştirme işlemleri ile kullanıcıya yeniden aktarmaktır.

4.6.1 Çevrim atlatma Yönteminin tanıtılması

Kıvılcım ateşlemeli motorlarda kısmi yüklerde efektif verim gaz keleşbeğinin kısılması sonucuyla doğan pompalama kayıplarıyla hızlıca düşmektedir. Bu düşüşü önlemek amacıyla, üzerinde en fazla durulan yöntemlerden biri de değişken strok hacmi yöntemidir.

$$N_i = P_{mi} \cdot V_H \cdot n \quad (4.1)$$

N_i : İndike güç

P_{mi} : Ortalama indike basınç

V_H : Bir silindirin strok hacmi

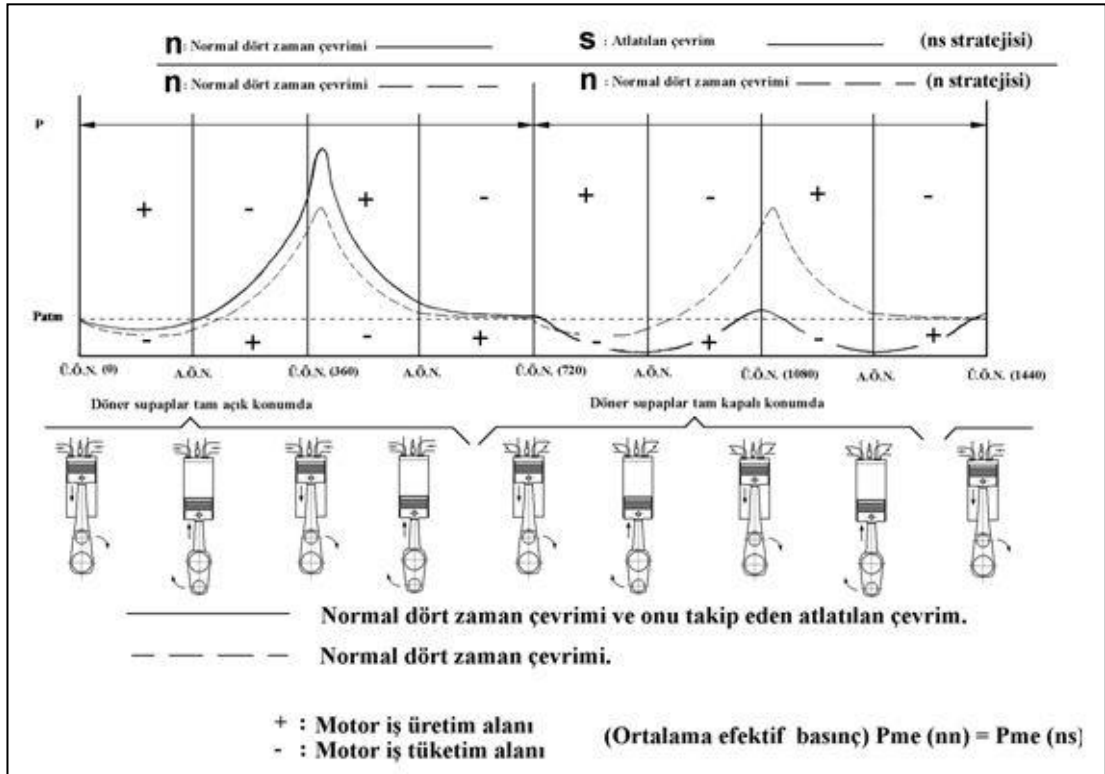
n : Motor devir sayısı

Bu yöntemde, motordan çekilen güç azaldığında etkin strok hacmi belirli bir oranda azaltılır, aynı devir koşulunda istenen gücün sağlanabilmesi için ortalama efektif basınç (çevrim başına dolgu girişı) aynı oranda arttırılır. Böylece, gaz keleşbeğı açıklığında sağlanan artış ile indike verim ve dolayısıyla toplam efektif verim yükseltilmiş olur[4].

Çevrim atlatma yöntemi, motordan çekilen güç azaldığında, değişken strok hacmi yönteminde belirli sayıda silindiri devre dışı bırakmamakta, her bir silindiri, birbirini izleyen çevrimlerde peşpeşe devreden çıkarıp devreye almayı amaçlamaktadır. Gerek çok silindirli, gerekse tek silindirli motorlara uygulanabilecek bu yöntem ile efektif verimde artış sağlanmasının yanında, ani moment değişimlerinin sebep olduğu titreşimlerin de büyük ölçüde giderilmesi beklenmektedir[4].

4.6.2 Çevrim atlatma yönteminin dört zamanlı motora uygulanışı

4 zamanlı motor çevrimi krank milinin 720° , yani iki defa dönmesinde tamamlamaktadır. Normalde, motordan çekilen güç azaldığında, gaz kelebeği kısılır ve yakıt sistemine kumanda edilerek, silindir içine çevrim başına giren dolgu miktarı azaltılır. Çevrim atlatma yönteminin uygulandığı motorda ise, motordan çekilen güç azaldığında, gaz kelebeğini kısıp silindir içine çevrim başına giren dolgu miktarını azaltmak yerine, bir çevrimin dört zamanı tamamlandıktan sonra takip eden dört zamanda iş üretilmeyerek, iş üretilen çevrim için gaz kelebeği açıklığı ve dolgu miktarı artırılır. Yani motor krank milinin ancak dört kez dönmesinde bir kez iş üretilir. Bu şekilde motor sekiz stroklu hale getirilmiş olur [4].

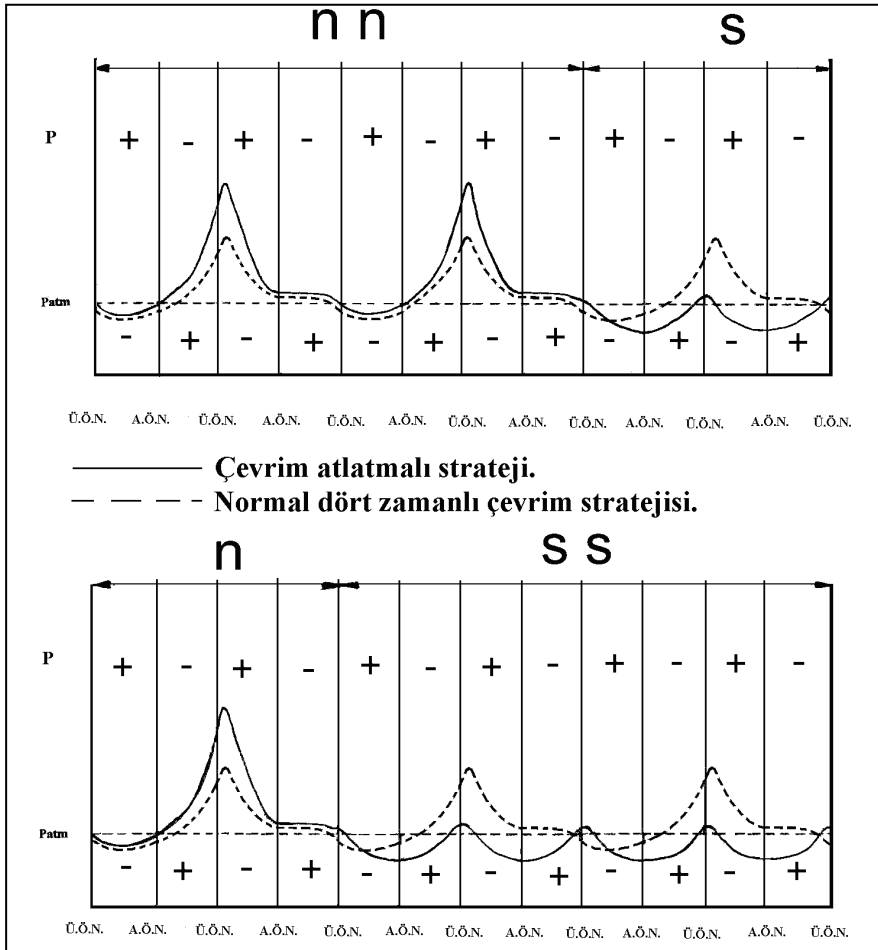


Şekil 4.29 : NS çevrimi

İki çevrimde bir defa iş üretebilmek için öncelikle iş üretilmeyen çevrimde silindire giren yakıtın kesilmesi gerekir. Ancak sadece yakıtın kesilmesi, pompalama kayıplarını ortadan kaldırmayacaktır. İş üretilmeyen çevrimde silindire hava emilmesi ve havanın tahliye edilmesi motordan iş çekilmesine neden olacağından, iş üretilen çevrimde pompalama kayıplarının azaltılmasıyla sağlanan verim kazancı kaybedilecek, hatta normal çalışmaya göre verim eksi değerlere düşecektir. Bu

sebeplerden dolayı, iş üretilmeyen çevrimlerde, emme zamanında silindire hava girişine engel olmak gerekmektedir [4].

Çevrim atlatma yönteminde, gerçekleşen çevrimler ‘N’ harfi ile, atlatılan çevrimler ise ‘S’ harfi ile temsil edilmektedir. Şekil 4.29’da, gerçekleşen bir çevrimi takip eden bir adet atlatılmış çevrim (NS) uygulaması temsili olarak p-KMA diyagramında gösterilmiştir. Tanımlanan çevrim atlatma stratejileri motorun yüküne ve hızına bağlı olarak genişletilebilir. Böylece çevrim atlatma işlemine geçişte, atlatılan çevrim / iş çevrimi çok küçük oranlardan başlatılarak motor yükü azaldıkça büyük oranlara doğru çıkarılabilir. Şekil 4.30’de, ‘12 stroklu’ olarak da adlandırılan, iki gerçekleşen çevrimi takip eden bir adet atlatılmış çevrim (NNS) ve bir gerçekleşen çevrimi takip eden iki adet atlatılmış çevrim (NSS) hali temsili olarak p-KMA diyagramında gösterilmiştir[4].



Şekil 4.30 : NSS çevrimleri

4.6.3 Güncellenen Kodlar ve arayüzün tanıtılması

Tek silindirli Lombardini motorda 1 adet buji ve 1 adet enjektör bulunmaktadır. Dolayısıyla tek rotorlu Wankel motoru için tasarlanan kontrol cihazında ateşleme ve püskürtme için bulunan 6 adet sürücü röleden sadece 2 tanesi kullanılarak (diğer 4'ü deaktif edilerek) elektronik açıdan tek silindirli motorun enjektör ve bobinleri enerjilendirmek mümkün olacaktır.

Mikrokontrolcü programı güncellenmeden önce doktora öğrencisi Yük.Müh. Abdurrahman Demirci'den çevrim atlatma mekanizmasının mekanik olarak kurulacağı dolayısıyla kontrolcü programının sadece normal çevrim için tekrar güncellenmesi gerektiği bilgisi alınmış ve Müh. Özgür Tekeli'nin kontrol algoritmasına sadık kalınarak sistem kodlarının güncellenmesi çalışmalarına başlanmıştır. Bu kapsamda seri haberleşme kodları Arduino ve C# arayüzü yazılımı için güncellenmiş ve yeni kodların daha basit bir algoritma ile daha az satırdan oluşmaları sağlanmıştır. Bununla birlikte mikroişlemcinin daha az satır işlemesi ve daha az yorulması amaçlanmıştır. Ayrıca Müh. Özgür Tekeli'nin sisteminde kapat butonu ile sadece seri haberleşme kapanırken, güncellenen kod ile birlikte kapat butonun tüm sistemi kapatması amaçlanmıştır. Böylece sistemi kapat butonuna basıldıktan sonra enjektör ve bobinlerin enerjilenmeye devam etmesi sorunu ortadan kaldırılmıştır.

```
if(stringComplete)
{
    a = inputString.indexOf(":");
    b = inputString.indexOf(":", a + 1);
    c = inputString.indexOf(":", b + 1);
    d = inputString.indexOf(":", c + 1);

    String Enjektör_basla=inputString.substring(0,a);
    enjektörbasla=Enjektör_basla.toInt();
    String Enjektör_sure=inputString.substring(a + 1, b);
    enjektörsuresi=Enjektör_sure.toInt();

    String Dwell_basla=inputString.substring(b + 1, c);
    dwellbasla=Dwell_basla.toInt();
    String Dwell_sure=inputString.substring(c + 1, d);
    dwellsuresi=Dwell_sure.toInt();
}
```

Şekil 4.31 : Güncellenen seri haberleşme kodu

```

if (inputString == "kapat")
{
    digitalWrite(atesleme,LOW);
    digitalWrite(enjektör,LOW);

}

stringComplete=false;
inputString = "";
}

```

Şekil 4.32 : Sistemi kapatmak için yazılan kodun bir parçası

Şekil 4.33 : Güncellenen kontrolcü arayüzü

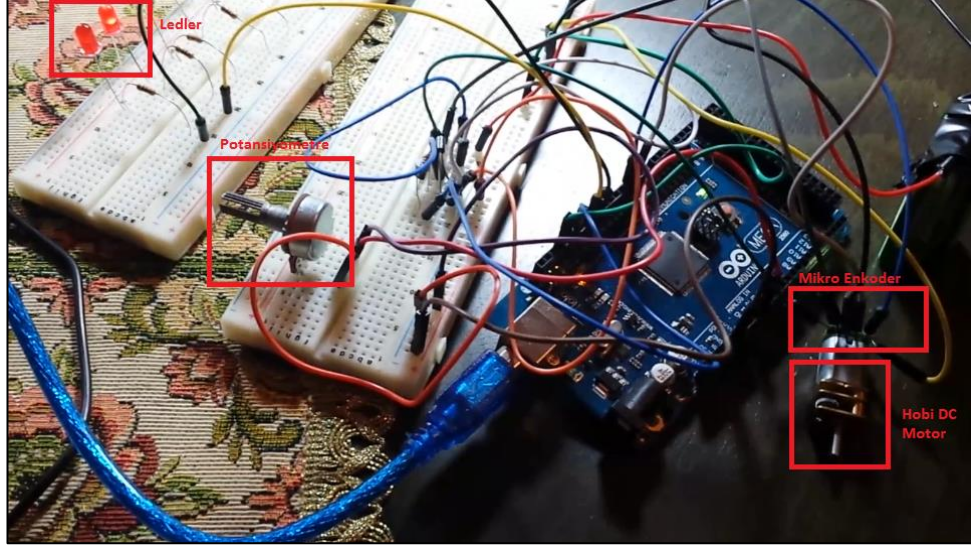
C# kontrol arayüzü de Wankel motoru için yapılan arayüze benzer bir mantıkla tek silindirli benzinli motorun normal çevrimi için güncellenmiştir. Doktora öğrencisi Yük.Müh. Abdurrahman Demirci'nin deneylerinde bobinin dolma süresi olan dwell süresi, ms olarak girileceğinden arayüzde de kullanıcıya dwell süresini ms cinsinden değiştirme imkanı tanınmıştır. Gelen veri kısmında Arduino'dan alınan motor devir bilgisi sayısal olarak okunabilirken, gösterge bölümünde ise görsel olarak kullanıcıya sunulmuştur. Giden veri kısmında veriler değiştirildikten ve veri gönder butonuna basıldıktan sonra paket halinde görüntülenebilmekte ve doğru verilerin gönderildiği kontrol edilebilmektedir. Kapat butonu ise seri haberleşme ile birlikte tüm elektronik sistemin kapanmasını sağlamaktadır.

5. DENEYLER

5.1 Ön Çalışmalar

Ateşleme ve püskürtme deneyleri yapılmadan önce hem Arduino ve C# bilgilerini pekiştirmek hem de adım adım ilerleyip sisteme hakim olmak için çok sayıda deneme yapılmış ve sonuca bu şekilde gidilmiştir. Bu denemelere örnek olarak Led yakma, string komutunu deneme, elektronik zar uygulaması, Arduino sinyaliyle potansiyometre vasıtasıyla DC motor hızı değiştirme, mikro enkoder pals okuma ve işleme, C# ile Arduino seri haberleşmesinin Ledler ile simülasyonu, ateşleme ve püskürtme elemanlarının ledler ile simüle edilerek yazılan kodun denenmesi verilebilir. Aşağıda bu örneklerden bazıları bulunmaktadır.

4.6.4 Mikro enkoder ile pals okuma ve okunan palsi işleme



Şekil 5.1: Enkoderden pals okumak ve işlemek için kurulan düzenek

Bu denemede sistemde önemli bir yer tutan enkoderden pals okunması çalışmaları yapılmıştır. Bunun için hazırlanan düzenek ve düzeneğin işleyişi şöyledir. 2 adet LED, biri buji diğeri enjektör olarak simüle edilmiş ve 1 turda 900 pals üreten mikro enkoder, farklı pals aralıklarında ledlere sinyal göndererek ledlerin periyodik olarak yanması sağlamıştır. Potansiyometre ile de DC motorun devri değiştirilerek, ledlerin

yanma aralıklarının hız arttıkça artması, hız azaldıkça azalması gözlemlenmiştir. Bu deneme ile ana sistemdeki enkoderden alınan palslerin farklı zamanlarda ateşleme ve püskürtme elemanlarına doğru sinyalleri göndereceği kanısına varılmıştır. İlgili kodun bir parçası Şekil 5.2’de gösterilmiştir.

```
long newPosition = myEnc.read();

if (newPosition != oldPosition)
{
    oldPosition = newPosition;
    //Serial.println(newPosition);
}

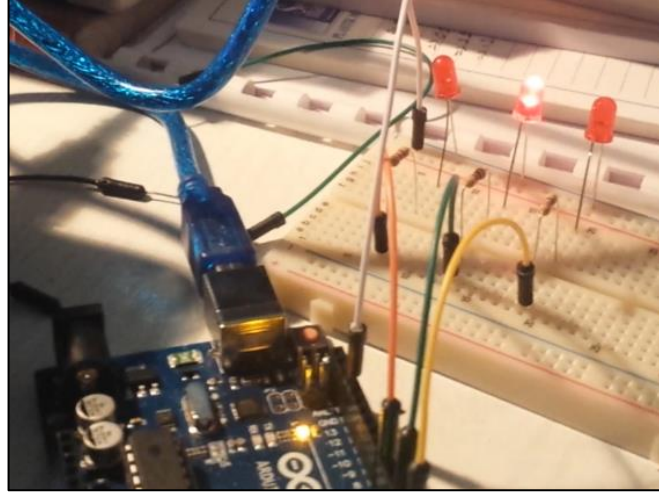
if (newPosition >= 200 && newPosition < 500)
{
    digitalWrite(ledPin1, HIGH);
}
else
{
    digitalWrite(ledPin1, LOW);
}

if (newPosition >= 550 && newPosition < 850)
{
    digitalWrite(ledPin2, HIGH);
}
else
{
    digitalWrite(ledPin2, LOW);
}
```

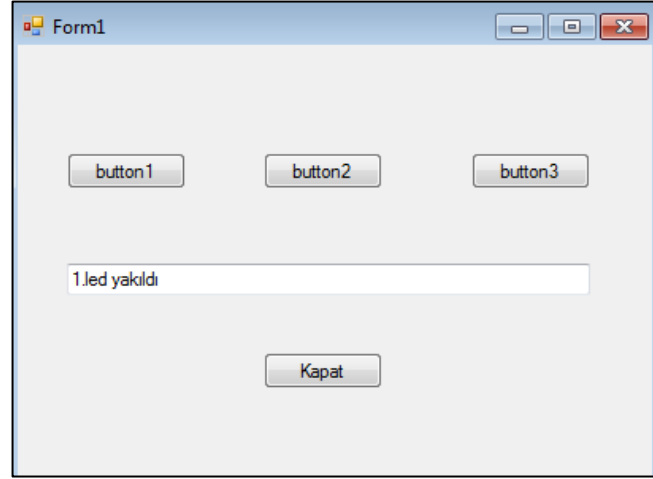
Şekil 5.2: Enkoderden pals okuma ve işleme ile ilgili kodun bir parçası

4.6.5 C# ile Arduino arasında seri haberleşme ile veri gönderme ve alma

Sistem açısından bir diğer önemli aşama da Arduino ile C# arayüzü arasındaki seri haberleşmedir. Seri haberleşme ile C#’dan Arduino’ya veri göndermek ve Arduino’dan alınan veriyi arayüzde görüntülemek için 3 adet LED’i arayüzdeki 3 adet buton ile yakıp söndürmek ve bu bilgileri görüntülemek için Şekil 5.3’deki düzenek ve 5.4’teki arayüz oluşturulmuştur. 1. Buton 1. Led’i, 2. Buton 2. Led’i, 3. Buton 3. Led’i hem yakıp hem söndürmek için kullanılmış ve Arduino’dan alınan ledlerin durumu bir Textbox aracılığıyla arayüzde gösterilmiştir. Böylece ana sistemde giden verilerin gönderileceği buton, sistemi kapatma butonu ve motor devrinin yazılacağı Textbox kutucuğunun sorunsuz çalıştırılacağı kanısına varılmıştır.



Şekil 5.3: Seri haberleşme denemelerinde kullanılan LED düzeneği



Şekil 5.4: Seri haberleşme denemelerinde kullanılan arayüz

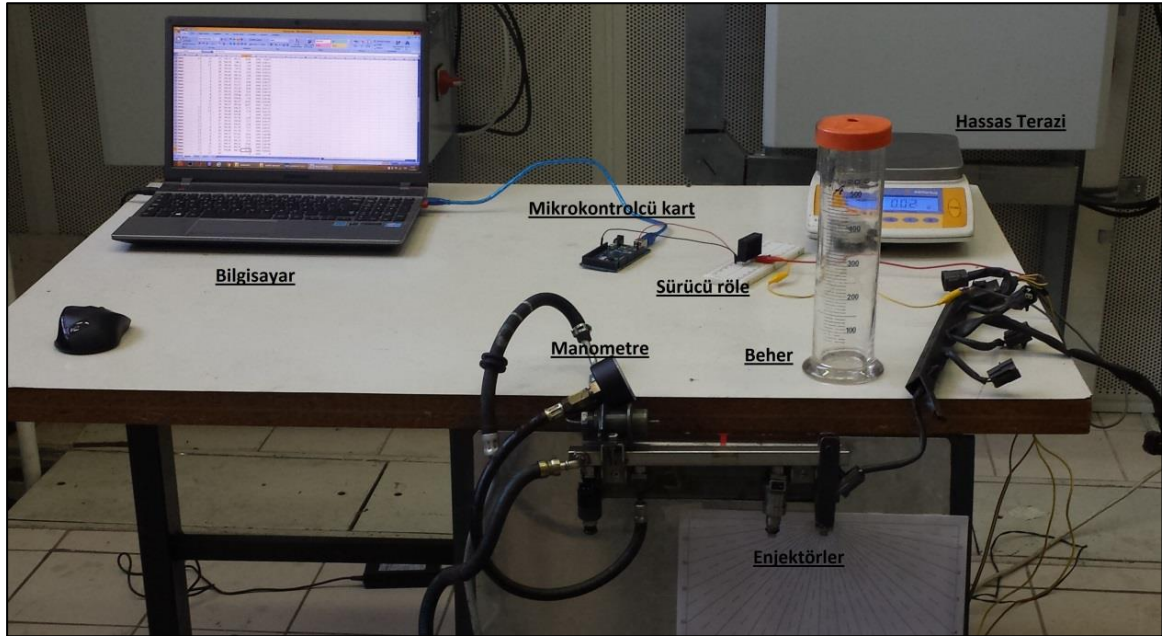
5.2 Enjektör Debi Karakteristiği Deneyi

İleride motor üzerinde doktora öğrencisi Yük.Müh. Ömer Cihan'ın yapacağı deneylerde farklı basınç ve püskürtme sürelerinde çalışılacaktır. Bu şartlarda yakıt püskürtme miktarlarının, dolayısıyla enjektör debilerinin bilinmesi önemlidir. Bu kapsamda Dr. Akın Kutlar'ın doktora tezinde kullandığı [ek] Tofaş Tempra 1.6 ve 2.0 motorlarında kullanılan enjektörler, BMW 3.0 enjektörler, Bosch direkt püskürtme enjektörleri ve Mazda RX8 Denso enjektörlerinin farklı basınç ve püskürtme süresi şartlarında debi miktarlarını ölçmek, debi karakteristiklerini karşılaştırmak ve enjektörleri birbirleriyle karşılaştırmak için bir deney düzeneği oluşturulmuştur.

Deney düzeneđi için ařađıdaki malzemeler kullanılmıřtır;

- Tempa çok nokta enjeksiyonlu yakıt kütüğü
- Enjektör girişine uygun manometre
- Hassas terazi
- AVL yakıt şartlandırma cihazı ve kompresör
- Basınçlı yakıt hortumları
- Beher
- Bilgisayar
- Arduino mikrokontrolcü kart
- Sürücü röleler

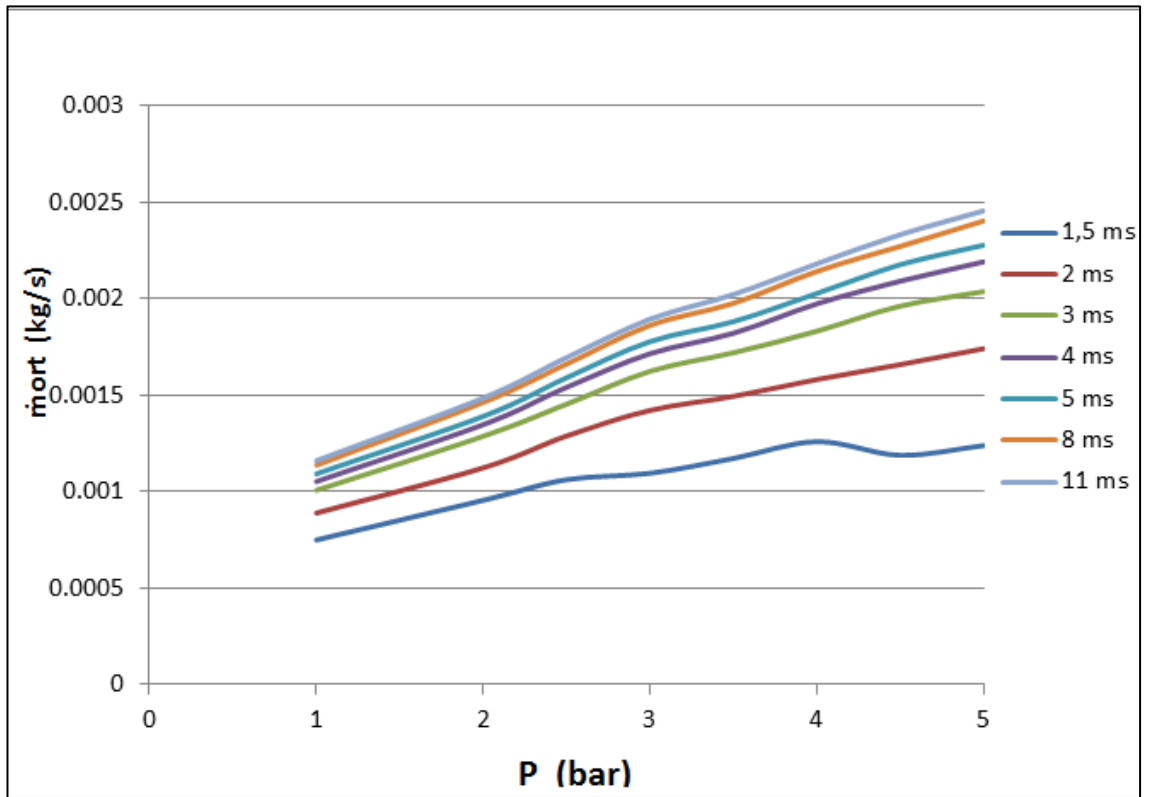
Deney düzeneğinde enjektörleri istenilen püskürtme süreleri ve sayılarında püskürtebilmek için Arduino mikrokontrolcü karta gönderilen bir program ile sürücü röleler vasıtasıyla enjektörler tetiklenmiştir. Bu işlem için Arduino’da yazılan kod hakkında Bölüm 4.3’te detaylı bilgi verilmiştir.



Şekil 5.5 : Enjektör debi karakteristiđi deney düzeneđi

Çizelge 5.1 : 1-5 bar basınçta Denso enjektör debi karakteristiği

Basınç, P [bar]	Püskürtme süresi [ms]	Ortalama debi, $\dot{m}_{ort}$ [kg/s]
1	1.5	0.000750726
1	2	0.000890028
1	3	0.001008358
1	4	0.00105387
1	5	0.001094015
1	8	0.001138751
1	11	0.001161819
5	1.5	0.001240143
5	2	0.001742504
5	3	0.002038335
5	4	0.002192511
5	5	0.002279022
5	8	0.002404375
5	11	0.00245682



Şekil 5.6 : Denso enjektör debi karakteristiği

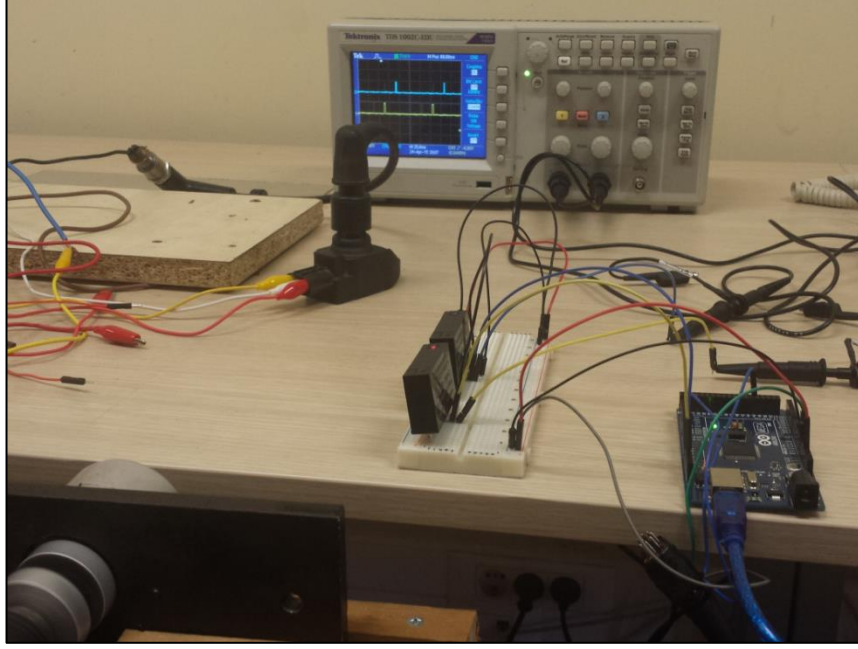
Enjektörlerin açık kalma, yani püskürtme süreleri olarak 1,5, 2, 3, 4, 5, 8, 11 ms için deneyler yapılmıştır. Mikrokontrolcünden 30 saniyede 1000 adet püskürtme sinyali gönderilmiştir. Püskürtülen yakıt bir behere toplanmış ve hassas terazide tartılarak sonuçlar kaydedilmiştir. AVL yakıt şartlandırma cihazından alınan yakıtın basıncı yükseltmek için ekstra bir kompresör kullanılmış ve deneyler 1, 1,5, 2, 2,5, 3, 3,5, 4, 4,5, 5 bar basınç aralıklarında yapılmıştır. Mazda RX8 Denso enjektörler için 1 ile 5 bar ölçüm sonuçları Çizelge 5.1’de, tüm sonuçlar ile elde edilen grafik Şekil 5.5’te verilmiştir. Sonuç olarak hem üzerinde çalışılacak Wankel deney motorunun orjinal enjektörü olması, hem de düşük püskürtme sürelerinde (1,5 ve 2 ms) diğer enjektörlerden daha kararlı çalışması Denso enjektörlerin kullanım sebepleri olmuştur.

5.3 Osiloskop Ölçümleri ve Matkap Deneyi

Arduino mikrokontrolcüye yazılan kodun ve C# arayüzünün hazırlanmasından sonra kontrol edilecek ateşleme ve püskürtme elemanları ile birlikte son bir deney düzeneği kurulmuştur. Mikrokontrolcü yazılımının istenenlere cevap vermesi, arayüzdeki geliştirmeler, sürücü rölelerin doğru şekilde çalışıp çalışmadığının testi bu deney düzeneğinde denenmiş ve son şeklini almıştır.

Deney düzeneğinde 2 adet buji ve 1 adet enjektör bulunmaktadır. Motor krank miline bağlanacak enkoder, deney motoru hazır olmadığı için laboratuarda bulunan ve dakikada 600 devir yapan bir el matkabıyla döndürülmüştür.

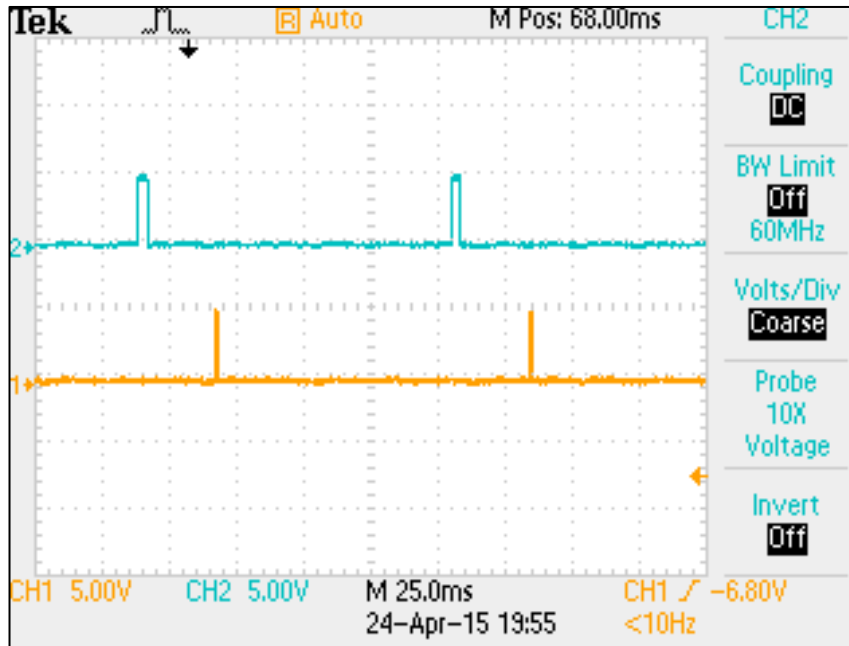
Öncelikli olarak sürücü rölelerin ateşleme ve püskürtme çıkışları osiloskopta incelenerek uygulanan PWM sinyalinin doğruluğu kontrol edilmek istenmiştir.



Şekil 5.7: Osiloskop ölçümü denemesinden bir kesit

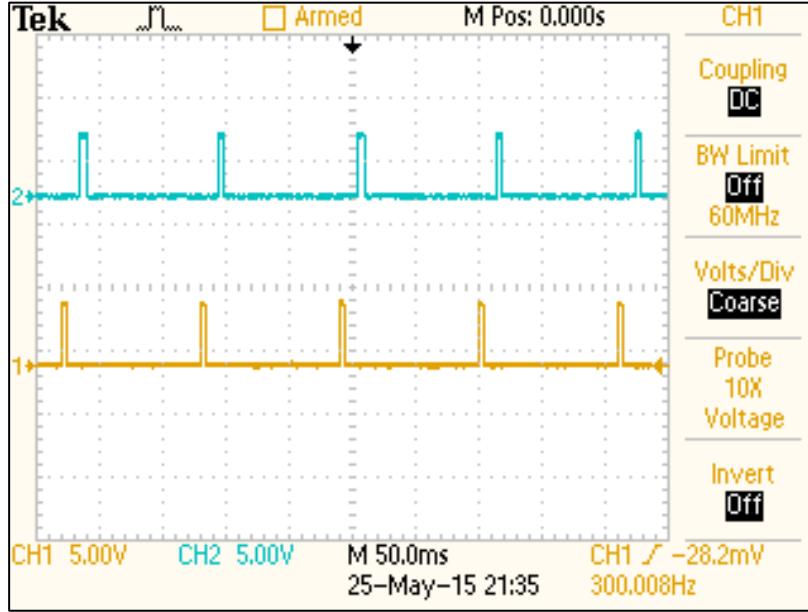
Sistemin deney motoru üzerinde çalışıp çalışmayacağı hakkındaki yorumlar sürücü rölelerin ateşleme ve püskürtme girişleri osiloskopta incelenerek yapılmıştır. Osiloskopun mavi renk ile gösterilen kanalı sürücü kartı ateşleme girişine, sarı renk ile gösterilen kanalı ise enjektör girişine bağlanmıştır.

Matkap ile çevirilen enkoder ile yapılan osiloskop ölçüm sonuçları Şekil 5.7 Şekil 5.8 ve Şekil 5.9 'da gösterilmiştir.

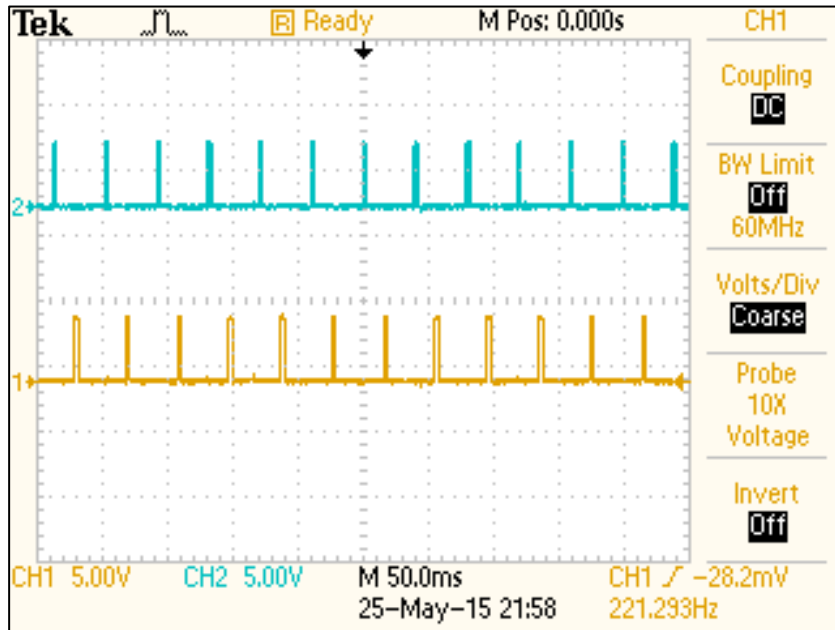


Şekil 5.8: 600 d/d için osiloskop ekran görüntüsü

Matkap , sonuçları farklı devirlerde gözlemleyebilmek için 600, 1000 ve 3000 d/d'da döndürülmüştür. Osiloskop ekranında bir kutucuğun düşey eksende yükseliği 5 volt, yatay eksendeki genişliği ise 600d/d için 25 ms, yüksek devirlerde 50 ms olarak belirlenmiştir. Ateşleme sinyali krank mili açısı cinsinden (yaklaşık 5ms), püskürtme sinyali süresi ise 2 ms verilmiştir. Ölçüm sonucunda ateşleme ve püskürtme sinyallerinin periyodik olarak ve birbirini takip ederek eksiksiz bir şekilde oluştuğu gözlemlenmiştir.

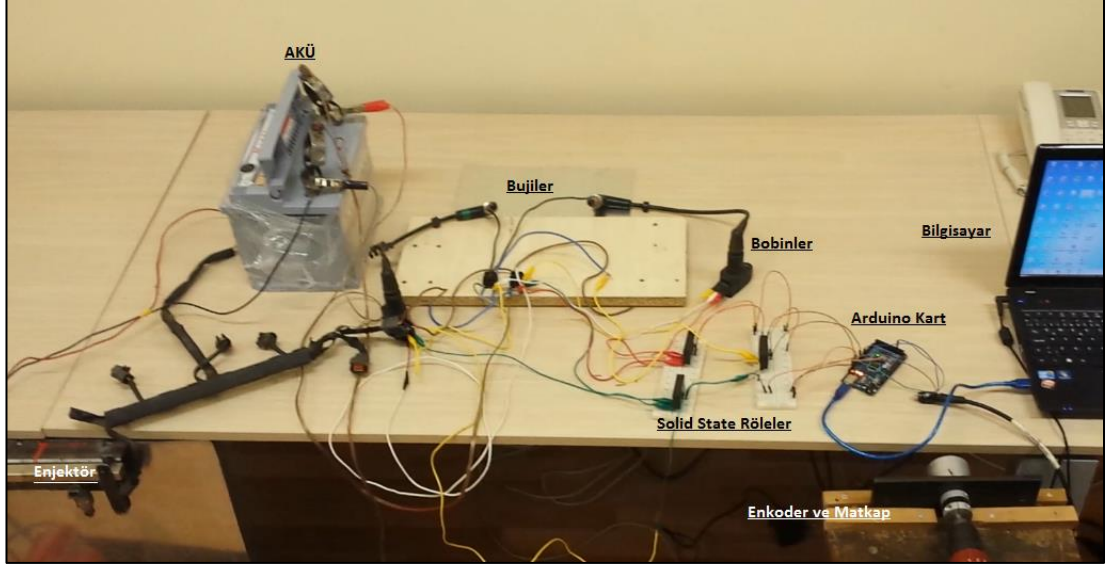


Şekil 5.9 : 1000 d/d için osiloskop ekran görüntüsü



Şekil 5.10 : 3000 d/d için osiloskop ekran görüntüsü

Osiloskop ölçümünde alınan sonuçlar ışığında matkap deneyi düzeneği kurulmuş, bujilerde oluşan kıvılcımın ve enjektördeki püskürtmenin kendilerini kontrol eden Solid-State (sürücü) röleler ile uyumlu olarak sorunsuz çalıştığı gözlemlenmiştir.



Şekil 5.11: Matkap deneyi

6. SONUÇLAR VE ÖNERİLER

Bu çalışmada tek rotorlu hale çevrilmiş bir Wankel motorunun ateşleme ve püskürtme mekanizmaları için mikrokontrolcü kart ve sinyal güçlendirici (sürücü) rölelerden oluşan bir elektronik kontrol cihazı tasarlanıp imal edilmiştir. Mikrokontrolcü kart olarak hem uygulanabilirliği hem de açık kaynaklı oluşu sebepleriyle Arduino tercih edilmiştir. Sürücü röle olarak, Arduino'dan alınan sinyal ile ateşleme ve püskürtme elemanlarını tetiklemek için klasik bir anahtarlama elemanı gibi çalışan ve tamamen elektronik komponentlerden oluşmasıyla milisaniye mertebesinde anahtarlama izin veren Solid-State röleler kullanılmıştır.

C tabanlı bir kontrol algoritması (Arduino IDE) oluşturulmuş ve yine C tabanlı bir kontrol arayüzü (C#) tasarlanıp kullanıcıya ateşleme ve püskürtme değişkenlerini bilgisayar üzerinden kontrol etme imkanı sunulmuştur.

Değişken basınç ve püskürtme süreleri için enjektör yakıt püskürtme miktarlarının hesaplandığı enjektör debi karakteristiği deney düzeneği kurulmuş ve Mazda RX8 Denso, BMW 3.0, Tempra 1.6, Tempra 2.0 ve BOSCH direkt enjektörleri için debi karakteristiği tabloları oluşturulmuştur.

Son olarak matkap deneyi düzeneği kurulmuş, imal edilen elektronik kontrol cihazının, yazılan kontrol algoritmasının ve arayüzünün başarıyla çalıştığı gözlemlenmiş ve sistem motor üzerinde çalışmaya hazır hale getirilmiştir.

Referans olarak çalışılacak motor tipi yerine ateşleme ve püskürtme elemanlarının çalışma zamanları alındığı için, sistem sadece Wankel motoru için değil, içten yanmalı benzinli herhangi bir motoru da kontrol edebilecek şekilde tasarlanmıştır. Bu kapsamda laboratuarda daha önce çevrim atlatma mekanizması ile birlikte kullanılan tek silindirli dört zamanlı bir benzin motoru için kontrol algoritması ve arayüzü oluşturulmuş ve elektronik kontrol cihazı ile uyumlu bir şekilde çalışacak hale getirilmiştir.

Ayrıca ileride motor üzerinde yapılacak deneylerde elde edilen verileri depolamak ve gerektiğinde kullanmak amacıyla harici bir hafızalama elemanı (EEPROM) donanımsal olarak sisteme eklenmiştir.

Sistemin geliştirilebilirliği açısından, motor toplanıp sistemin çalıştırılmasıyla elde edilecek sonuçların; özellikle ölçülen gaz keleşeđi açısı ve motor devri deđerlerine göre, control edilen ve optimum deđerleri bulunan ateşleme avansı ve yakıt püşkürtme miktarı deđişkenlerinin haritası çıkarılabilir. Bu harita Arduino mikrokontrolcüye yazılacak bir algoritma ile harici hafızalama elemanına gömülebilir. Hafızalama elemanı bilgisayar arayüzünden müdahaleye gerek kalmadan hafızasındaki optimum deđerleri döndürerek motoru en verimli şekilde çalıştırabilir.

KAYNAKLAR

- [1]**Yıldırım, T.** 2010. Tek zamalı 4 silindirli benzinli bir motor için çevrim atlatma mekanizmasının tasarlanması, imalatı ve motora adapte edilmesi, *Yüksek Lisans Tezi*, İTÜ, Fen Bilimleri Enstitüsü, İstanbul, Türkiye.
- [2]**Yamamoto, K.** 1981 Rotary Engine, Sankaido
- [3]**Megep.** Motorlu Araçlar Teknolojisi Alternatif Motorlar ve Yakıtlar Alındığı tarih: 18.04.2015, adres: http://hbogm.meb.gov.tr/modulerprogramlar/kursprogramlari/motorlu_aracilar/moduller/alternatif_motorlar_ve_yakitlar.pdf
- [4]**Tekeli, Ö.** 2013 Tek Çevrim Atlatmalı bir benzin motorunun ateşleme ve püskürtme ünitelerinin tasarımı ve üretimi, *Yüksek Lisans Tezi*, İTÜ, Fen Bilimleri Enstitüsü, İstanbul, Türkiye
- [5]**Cihan Ö.** 2014 İki zamanlı Wankel motorlarının deneysel ve teorik olarak incelenmesi, *Doktora Tezi 2.İlerleme Raporu*, İTÜ, Fen Bilimleri Enstitüsü, İstanbul, Türkiye
- [6]**Ohkubo, M., Tashima, S., Shimizu, R., Fuse, S., Ebino, H.** 2004 Developed Technologies of the new rotary engine (RENESIS).pdf
- [7]**Mazda** Döner Pistonlu motorlar için motor yönetim teşhis ve mekanik.pdf, *Mazda Servis verileri, Alındığı tarih: 23.04.2015*
- [8]**Megep.** Motorlu Araçlar Teknolojisi Benzinli Motorların Yakıt ve Ateşleme Sistemleri I. Alındığı tarih: 18.04.2015, adres: http://hbogm.meb.gov.tr/modulerprogramlar/kursprogramlari/motorlu_aracilar/moduller/benzinli_motorlari_yakit_ve_atesleme_sistemleri1.pdf
- [9]**NGK** Spark plug wire instalation.pdf, *NGK teknik verileri, Alındığı tarih: 10.04.2015*
- [10]**Kutlar, O. A.,** 1999. Dört zamanlı otto (rochas) çevrimli motorlarda kısmi yükte yakıt tüketimini azaltmak için yeni bir yöntem (periyot atlatmalı motor), *Doktora Tezi*, İTÜ, Fen Bilimleri Enstitüsü, İstanbul, Türkiye.
- [11]**Norton** Solid State Rölelerin özellikleri, *katalog* , Alındığı tarih: 18.04.2015
- [12]**Delebe, E.,** 2015. Arduino, İnkılap Yayınevi
- [13]**Taşdemir, C.,** 2012. Arduino, Dikeyksen Yayınevi
- [14]**Url-1** <http://tr.wikipedia.org/wiki/Otomobilin_tarihi>, alındığı tarih: 06.04.2015
- [15]**Url-2**<<http://www.elektrikport.com/teknik-kutuphane/icten-yanmali-motorlar-1bolum/9017#ad-image-0>>, alındığı tarih: 06.04.2015
- [16]**Url-3** <http://tr.wikipedia.org/wiki/Wankel_motoru>, alındığı tarih: 06.04.2015

- [17]**Url-4** <<http://ieee.bilkent.edu.tr/teknoloji101/?p=1407>>, alındığı tarih: 08.04.2015
- [18]**Url-5** <<http://www.ngk.de/tr/ayrintilariyla-teknoloji/bujiler/bujiler-hakkinda-temel-bilgiler/bir-bujinin-yapisi/>>, alındığı tarih: 08.04.2015
- [19]**Url-1** <http://tr.wikipedia.org/wiki/End%C3%BCksiyon_bobini>, alındığı tarih: 11.04.2015
- [20]**Url-1** <<http://tr.wikipedia.org/wiki/Ak%C3%BCm%C3%BClat%C3%B6r>>, alındığı tarih: 12.04.2015
- [21]**Url-1** <<http://www.ngk.de/tr/ueruenler-ve-teknolojiler/bujiler/buji-teknolojileri/asil-metal-bujileri/>>, alındığı tarih: 12.04.2015
- [22]**Url-1** <<http://www.ngk.de/tr/ueruenler-ve-teknolojiler/bujiler/teknik-sartlar/>>, alındığı tarih: 13.04.2015
- [23]**Url-1** <<https://www.ngk.de/tr/ueruenler-ve-teknolojiler/atesleme-bobinleri/atesleme-bobinlerinde-aranan-sartlar/>>, alındığı tarih: 14.04.2015
- [24]**Url-1** <<https://www.ngk.de/tr/ueruenler-ve-teknolojiler/atesleme-bobinleri/atesleme-bobini-teknolojileri/buji-yuvali-atesleme-bobinleri/>>, alındığı tarih: 14.04.2015
- [25]**Url-1** <<http://www.ngk.de/tr/ueruenler-ve-teknolojiler/atesleme-kablolari/atesleme-kablosu-teknolojileri/karbon-direncli-atesleme-kablolari/>>, alındığı tarih: 14.04.2015
- [26]**Url-1** <<http://www.elektrikport.com/teknik-kutuphane/optokuplor-nedir/12346#ad-image-0>>, alındığı tarih: 15.04.2015
- [27]**Url-1** <<http://www.3eelectrotech.com.tr/arsiv/yazi/137-solid-state-roleler-kullanim-alanlari-ve-mekanik-roleler-ile-karsilastirilmesi>>, alındığı tarih: 15.04.2015
- [28]**Url-1** <<http://www.nortonelectronic.com/0/solid-state-roleler/dc-solid-state-roleler/monofaze/standart-amper-3a-60a/kart-tipi-2.html>>, alındığı tarih: 15.04.2015
- [29]**Url-1** <<http://andybrown.me.uk/wk/2011/08/28/512kb-sram-expansion-for-the-arduino-mega-build/>>, alındığı tarih: 21.03.2015
- [30]**Url-1** <<http://tr.wikipedia.org/wiki/EEPROM>>, alındığı tarih: 20.04.2015

EKLER

EK A.1 : Teknik Detaylar

EK B.1 : C# kodu (kısmi olarak)

TOSHIBA

TLP521-1, TLP521-2, TLP521-4

TOSHIBA PHOTOCOUPLER GaAs IRED & PHOTO-TRANSISTOR

TLP521-1, TLP521-2, TLP521-4

PROGRAMMABLE CONTROLLERS

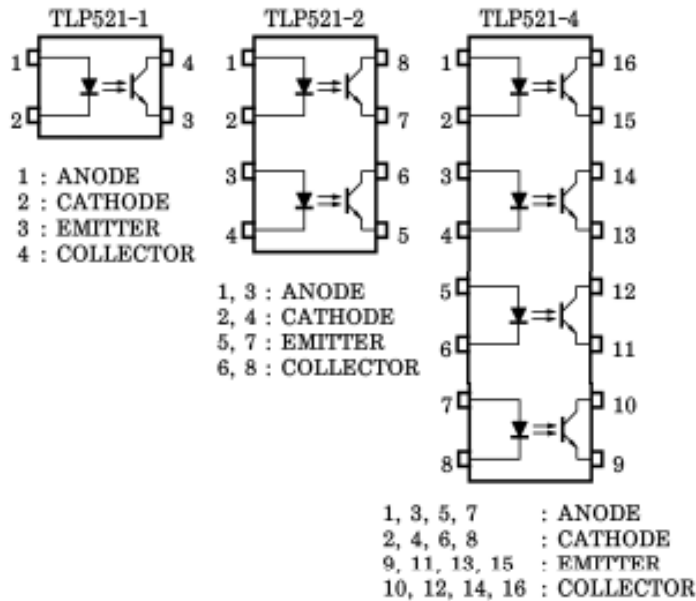
AC/DC-INPUT MODULE

SOLID STATE RELAY

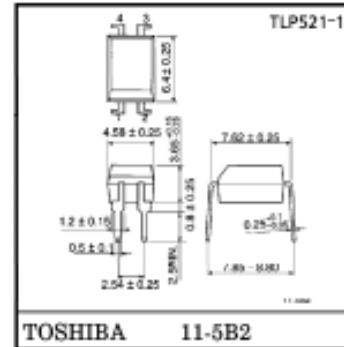
The TOSHIBA TLP521-1, -2 and -4 consist of a photo-transistor optically coupled to a gallium arsenide infrared emitting diode. The TLP521-2 offers two isolated channels in an eight lead plastic DIP package, while the TLP521-4 provides four isolated channels in a sixteen plastic DIP package.

- Collector-Emitter Voltage : 55 V (min)
- Current Transfer Ratio : 50% (min)
Rank GB : 100% (min)
- Isolation Voltage : 2500 Vrms (min)
- UL Recognized
made in Japan : UL1577, File No. E67349
made in Thailand : UL1577, File No. E152349

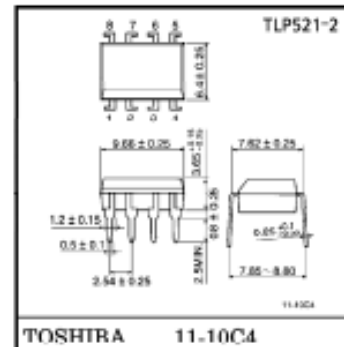
PIN CONFIGURATIONS (TOP VIEW)



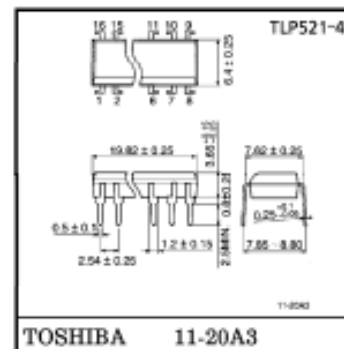
Unit in mm



Weight : 0.26 g



Weight : 0.54 g



Weight : 1.1 g

000707B9C3

- TOSHIBA is continually working to improve the quality and reliability of its products. Nevertheless, semiconductor devices in general can malfunction or fail due to their inherent electrical sensitivity and vulnerability to physical stress. It is the responsibility of the buyer, when utilizing TOSHIBA products, to comply with the standards of safety in making a safe design for the entire system, and to avoid situations in which a malfunction or failure of such TOSHIBA products could cause loss of human life, bodily injury or damage to property. In developing your designs, please ensure that TOSHIBA products are used within specified operating ranges as set forth in the most recent TOSHIBA products specifications. Also, please keep in mind the precautions and conditions set forth in the "Handling Guide for Semiconductor Devices," or "TOSHIBA Semiconductor Reliability Handbook" etc.
- The TOSHIBA products listed in this document are intended for use in general electronics applications (computer, personal equipment, office equipment, measuring equipment, industrial robotics, domestic appliances, etc.). These TOSHIBA products are neither intended nor warranted for use in equipment that requires extraordinarily high quality and/or reliability or a malfunction or failure of which may cause loss of human life or bodily injury ("Unintended Usage"). Unintended Usage include atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, medical instruments, all types of safety devices, etc.. Unintended Usage of TOSHIBA products listed in this document shall be made at the customer's own risk.

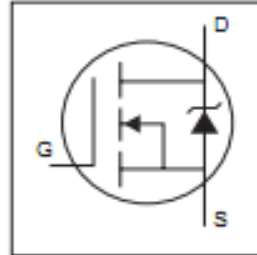
2000-08-28 1/8

Şekil A.1 : TLP521 ışınal bağlayıcı teknik detayı

IRFZ44N

HEXFET® Power MOSFET

- Advanced Process Technology
- Ultra Low On-Resistance
- Dynamic dv/dt Rating
- 175°C Operating Temperature
- Fast Switching
- Fully Avalanche Rated

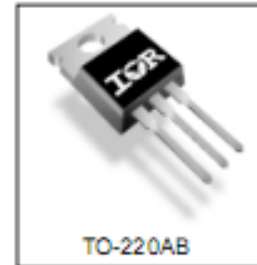


$V_{DS} = 55V$
 $R_{DS(on)} = 17.5m\Omega$
 $I_D = 49A$

Description

Advanced HEXFET® Power MOSFETs from International Rectifier utilize advanced processing techniques to achieve extremely low on-resistance per silicon area. This benefit, combined with the fast switching speed and ruggedized device design that HEXFET power MOSFETs are well known for, provides the designer with an extremely efficient and reliable device for use in a wide variety of applications.

The TO-220 package is universally preferred for all commercial-industrial applications at power dissipation levels to approximately 50 watts. The low thermal resistance and low package cost of the TO-220 contribute to its wide acceptance throughout the industry.



Absolute Maximum Ratings

	Parameter	Max.	Units
$I_D @ T_C = 25^\circ C$	Continuous Drain Current, $V_{GS} @ 10V$	49	A
$I_D @ T_C = 100^\circ C$	Continuous Drain Current, $V_{GS} @ 10V$	35	
I_{DM}	Pulsed Drain Current ①	180	
$P_D @ T_C = 25^\circ C$	Power Dissipation	94	W
	Linear Derating Factor	0.63	W/°C
V_{GS}	Gate-to-Source Voltage	± 20	V
I_{AR}	Avalanche Current ②	25	A
E_{AR}	Repetitive Avalanche Energy ③	9.4	mJ
dv/dt	Peak Diode Recovery dv/dt ④	5.0	V/ns
T_J	Operating Junction and	-55 to $+175$	°C
T_{stg}	Storage Temperature Range		
	Soldering Temperature, for 10 seconds	300 (1.8mm from case)	
	Mounting torque, 6-32 or M3 screw	10 lbf•in (1.1N•m)	

Thermal Resistance

	Parameter	Typ.	Max.	Units
$R_{\theta JC}$	Junction-to-Case	---	1.5	°C/W
$R_{\theta CS}$	Case-to-Sink, Flat, Greased Surface	0.50	---	
$R_{\theta JA}$	Junction-to-Ambient	---	62	

www.irf.com

1
01/03/01

Şekil A.2 : IRFZ44n mosfet teknik detayı

	Absolute Singleturn				Multiturn				Incremental			
	ROC 425	ROC 413	ROC 412 ROC 413	ROC 413	ROC 425	ROC 424 ROC 425	ROC 425	ROC 425	ROC 426	ROC 436	ROC 436	ROC 436
Absolute position values*	Endat 2.2	Endat 2.2	SSI	PROFIBUS-DP	Endat 2.2	SSI	PROFIBUS-DP	PROFIBUS-DP	–	–	–	–
Ordering designation	Endat 2.2	Endat 01	–	–	Endat 2.2	–	–	–	–	–	–	–
Positions per rev	33554/432 (25 bits)	8192 (13 bits)	1024 (10 bits) 4096 (12 bits) 8192 (13 bits)	8192 (13 bits) ²⁾	33554/432 (25 bits)	8192 (13 bits)	4096 (12 bits) 8192 (13 bits)	8192 (13 bits) ²⁾	–	–	–	–
Revolutions	–	–	–	–	4096	–	–	4096 ²⁾	–	–	–	–
Code	Pure binary	Gray	Gray	Pure binary	Pure binary	Gray	Gray	Pure binary	–	–	–	–
Res. permissible speed/ at accuracy	512 lines: \$12000 rpm for continuous position value 2048 lines: \$15000 rpm/50 LSB	512 lines: \$12000 rpm/50 LSB 2048 lines: \$15000 rpm/50 LSB	512 lines: \$12000 rpm/50 LSB 2048 lines: \$15000 rpm/50 LSB	512 lines: \$12000 rpm/50 LSB 2048 lines: \$15000 rpm/50 LSB	512 lines: \$12000 rpm/50 LSB 2048 lines: \$15000 rpm/50 LSB	512 lines: \$12000 rpm/50 LSB 2048 lines: \$15000 rpm/50 LSB	512 lines: \$12000 rpm/50 LSB 2048 lines: \$15000 rpm/50 LSB	512 lines: \$12000 rpm/50 LSB 2048 lines: \$15000 rpm/50 LSB	–	–	–	–
Calculation time t_{calc}	≤ 5 µs	≤ 0.25 µs	≤ 0.5 µs	–	≤ 5 µs	≤ 0.25 µs	≤ 0.5 µs	–	–	–	–	–
Incremental signals	Without	Without	Without	Without	Without	Without	Without	Without	–	–	–	–
Line counts*	–	512 2048	512	512 (internal only)	–	512 2048	512	512 (internal only)	–	–	–	–
Cutoff freq. ~3 dB Scanning frequency Edge separation s_e	–	512 lines/200 kHz; 2048 lines/200 kHz	–	–	–	512 lines/200 kHz; 2048 lines/200 kHz	–	–	–	–	–	–
System accuracy	± 20"	512 lines/60"; 2048 lines/20"	512 lines/60"; 2048 lines/20"	± 60"	± 20"	512 lines/60"; 2048 lines/20"	–	–	–	–	–	–
Power supply*	3.6 to 5.25 V	5 V ± 5 %	5 V ± 5 % or 10 to 30 V	10 to 30 V	3.6 to 5.25 V	5 V ± 5 %	5 V ± 5 % or 10 to 30 V	10 to 30 V	5 V ± 10 %	10 to 30 V	10 to 30 V	5 V ± 10 %
Current con. without load	≤ 160 mA	≤ 160 mA	≤ 160 mA	≤ 125 mA at 24 V	≤ 160 mA	≤ 200 mA	≤ 200 mA	≤ 125 mA at 24 V	120 mA	100 mA	160 mA	120 mA
Electrical connection*	• Flange socket M12, radial • Cable 1 m, with coupling M12	• Flange socket M12, radial • Cable 1 m, with coupling M23	• Flange socket M12, radial • Cable 1 m, with coupling M23	Screw terminals; radial cable exit	• Flange socket M12, radial • Cable 1 m, with coupling M12	• Flange socket M12, radial • Cable 1 m, with coupling M23	• Flange socket M12, radial • Cable 1 m, with coupling M23	Screw terminals; radial cable exit	• Range socket M23, axial and axial • Cable 1 m/5 m, with or without coupling M23	–	–	–
Shaft	Solid shaft D = 6 mm	–	–	–	–	–	–	–	Solid shaft D = 6 mm	–	–	–
Mech. permissible speed n	≤ 12000 rpm	–	–	–	–	–	–	–	≤ 16000 rpm	–	–	–
Starting torque	≤ 0.01 Nm (at 20 °C)	–	–	–	–	–	–	–	≤ 0.01 Nm (at 20 °C)	–	–	–
Moment of inertia of rotor	2.7 · 10 ⁻⁸ kgm ²	–	–	3.8 · 10 ⁻⁸ kgm ²	2.7 · 10 ⁻⁸ kgm ²	–	–	–	2.7 · 10 ⁻⁸ kgm ²	–	–	–
Shaft load ⁵⁾	Axial 10 N/radial 20 N at shaft end	–	–	–	–	–	–	–	Axial 10 N/radial 20 N at shaft end	–	–	–
Vibration 55 to 2000 Hz Shock 8 ms/2 ms	≤ 300 m/s ² (IEC 60068-2-6) ≤ 1000 m/s ² (IEC 60068-2-27)	–	–	–	–	–	–	–	≤ 300 m/s ² (IEC 60068-2-6) ≤ 1000 m/s ² (IEC 60068-2-27)	–	–	–
Max. operating temp.	U _p = 5 V/100 °C; U _p = 10 to 30 V/85 °C	–	–	60 °C	U _p = 5 V/100 °C; U _p = 10 to 30 V/85 °C	–	–	60 °C	100 °C	70 °C	100 °C	100 °C
Min. operating temp.	Range socket or fixed cable: -40 °C Moving cable: -10 °C	–	–	-20 °C	Range socket or fixed cable: -40 °C Moving cable: -10 °C	–	–	-20 °C	Range socket or fixed cable: -40 °C Moving cable: -10 °C	–	–	–
Protection EC 80/529	IP 67 at housing; IP 64 at shaft end ³⁾	–	–	–	–	–	–	–	IP 67 at housing; IP 64 at shaft end ³⁾	–	–	–
Weight	Approx. 0.35 kg	–	–	–	–	–	–	–	Approx. 0.3 kg	–	–	–

Sekil A.3 : Heidenhain ROD426 enkoder teknik detayı

isc Silicon NPN Power Transistor

BU323A

DESCRIPTION

- Collector-Emitter Sustaining Voltage-
: $V_{CE(sus)} = 400V(\text{Min.})$
- DARLINGTON
- High Reliability

APPLICATIONS

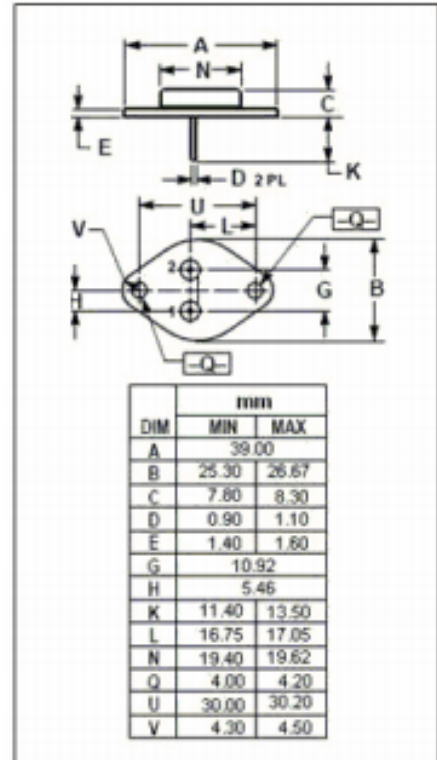
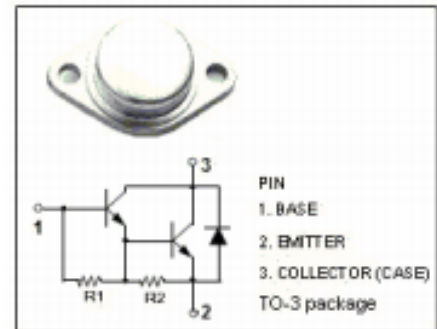
- Automotive ignition
- Switching regulator
- Motor control applications

ABSOLUTE MAXIMUM RATINGS ($T_s=25^\circ\text{C}$)

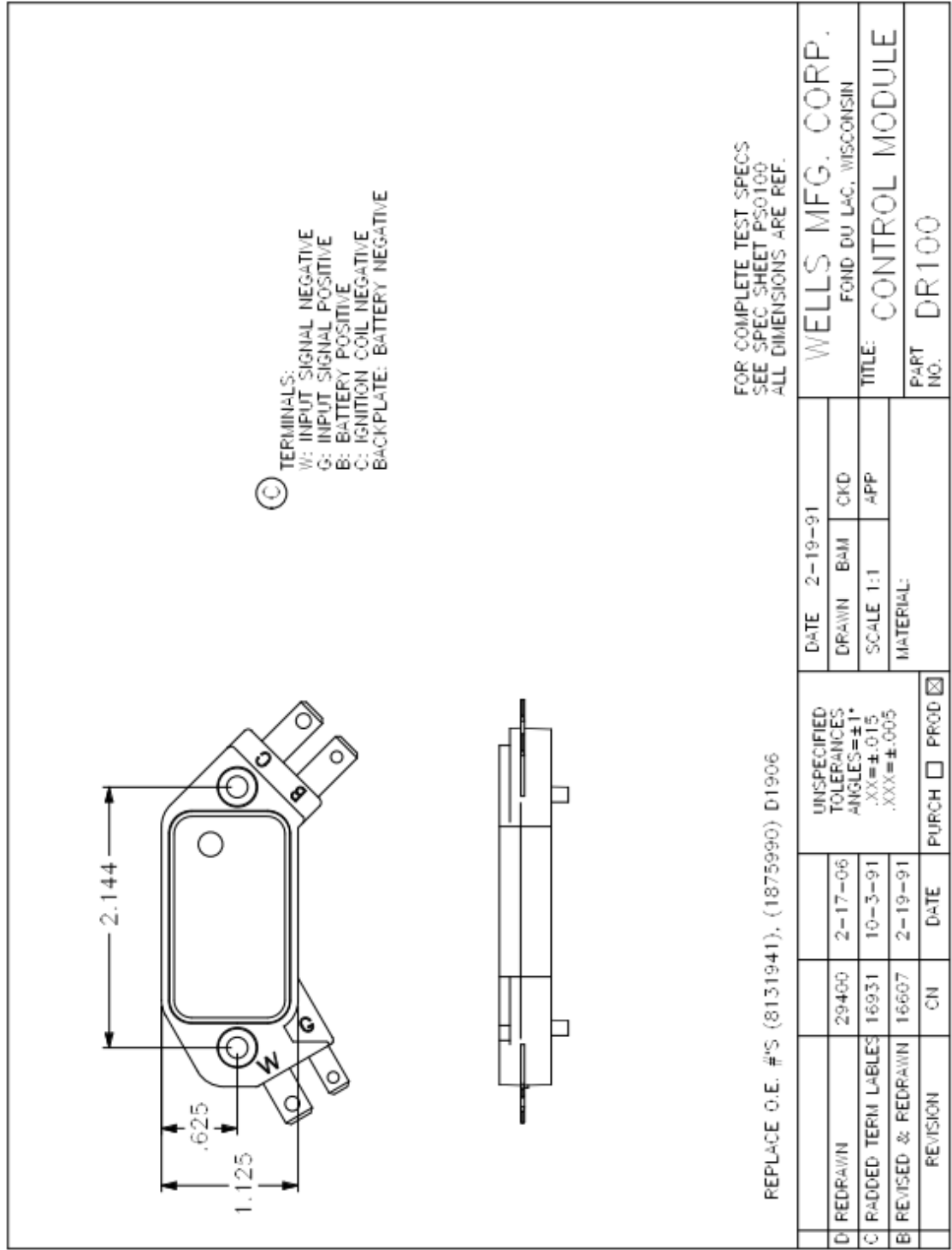
SYMBOL	PARAMETER	VALUE	UNIT
V_{CB0}	Collector-Base Voltage	600	V
V_{CE0}	Collector-Emitter Voltage	400	V
V_{EB0}	Emitter-Base Voltage	8	V
I_C	Collector Current- Continuous	10	A
I_{CM}	Collector Current-Peak	16	A
I_B	Base Current	3	A
P_C	Collector Power Dissipation @ $T_c=25^\circ\text{C}$	175	W
T_j	Junction Temperature	200	$^\circ\text{C}$
T_{stg}	Storage Temperature Range	-65~200	$^\circ\text{C}$

THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	MAX	UNIT
$R_{th(j-c)}$	Thermal Resistance, Junction to Case	1.0	$^\circ\text{C/W}$



Şekil A.4 : Bu323A transistör teknik detayı



Şekil A.6 : GM DR100 ateşleme modülü teknik resmi

High Energy Ignition Circuit

This device is designed to use the signal from a reductor type ignition pickup to produce a well controlled output from a power Darlington output transistor.

- Very Low Peripheral Component Count
- No Critical System Resistors
- Wide Supply Voltage Operating Range (4.0 V to 24 V)
- Overvoltage Shutdown (30 V)
- Dwell Automatically Adjusts to Produce Optimum Stored Energy without Waste
- Externally Adjustable Peak Current
- Available in Chip and Flip-Chip Form
- Transient Protected Inputs and Outputs

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Power Supply Voltage—Steady State	V_{bat}	24	V
Transient 300 ms or less		90	
Output Sink Current—Steady State	$I_O(\text{Sink})$	300	mA
Transient 300 ms or less		1.0	A
Junction Temperature	$T_{J(\text{max})}$	150	°C
Operating Temperature Range	T_A	-40 to +125	°C
Storage Temperature Range	T_{stg}	-65 to +150	°C
Power Dissipation, Plastic Package, Case 628	P_D	1.25	W
Derate above 25°C		10	mW/°C

MC3334
MCC3334
MCCF3334

HIGH ENERGY
IGNITION CIRCUIT

SEMICONDUCTOR
TECHNICAL DATA



P SUFFIX
PLASTIC PACKAGE
CASE 628

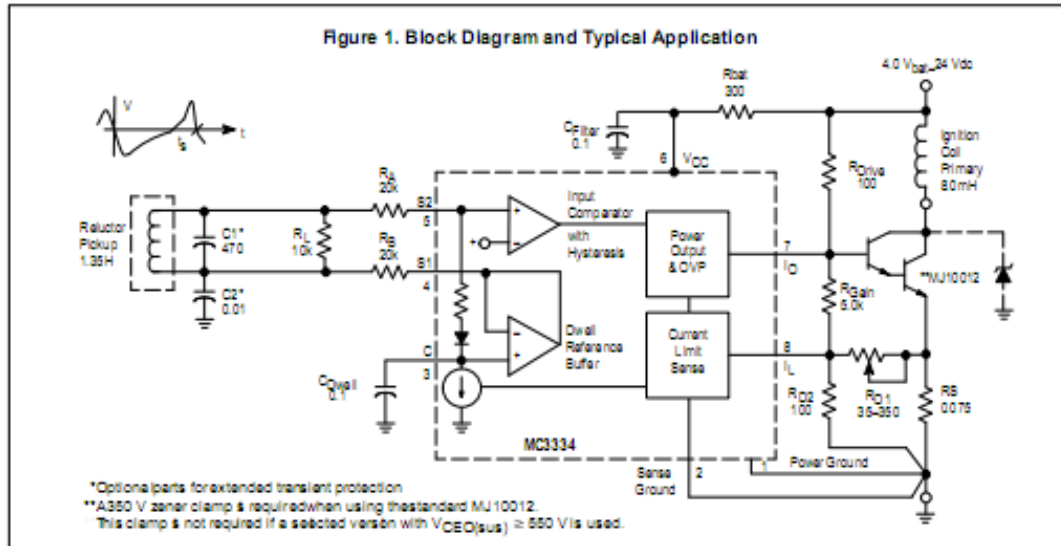
D SUFFIX
PLASTIC PACKAGE
CASE 751
(SO-8)



ORDERING INFORMATION

Device	Operating Temperature Range	Package
MC3334P	$T_A = -40^\circ \text{ to } +125^\circ \text{C}$	Plastic DIP
MC3334D		SO-8
MCC3334		Chip
MCCF3334		Flip-Chip

Figure 1. Block Diagram and Typical Application



Şekil A.7 : GM DR100 modül içindeki MC3334 transistör teknik detayı

EK B.1

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;
using System.IO.Ports;
using System.Threading;

namespace WindowsFormsApplication2
{
    public partial class Form1 : Form
    {
        public SerialPort sp;
        string dataReceived = string.Empty;
        private delegate void SetTextDeleg(string text);
        public Form1()
        {
            InitializeComponent();
        }

        private void groupBox1_Enter(object sender, EventArgs e)
        {

        }

        private void numericUpDown1_ValueChanged(object sender, EventArgs e)
        {

        }

        private void verigonderButton_Click(object sender, EventArgs e)
        {
            gidentext.Text = ((Convert.ToInt16(3600 - numericUpDownEnjekorPB1.Value *
10)) + ":" + (Convert.ToInt16(3600 - numericUpDownEnjekorPB2.Value * 10)) +
":" + (Convert.ToInt16(3600 - numericUpDownEnjekorPB3.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownEnjekorPB4.Value * 10)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS1.Value * 1000)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS2.Value * 1000)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS3.Value * 1000)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS4.Value * 1000)) + ":" +
(Convert.ToInt16(3600 - numericUpDownBujiDB1.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownBujiDB2.Value * 10)) + ":" +
```

```
(Convert.ToInt16(3600 - numericUpDownBujiAA1.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownBujiAA2.Value * 10)) + ":\n");
```

```
try
{
    sp.Write((Convert.ToInt16(3600 - numericUpDownEnjekorPB1.Value * 10))
+ ":" + (Convert.ToInt16(3600 - numericUpDownEnjekorPB2.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownEnjekorPB3.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownEnjekorPB4.Value * 10)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS1.Value * 1000)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS2.Value * 1000)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS3.Value * 1000)) + ":" +
(Convert.ToInt16(numericUpDownEnjekorPS4.Value * 1000)) + ":" +
(Convert.ToInt16(3600 - numericUpDownBujiDB1.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownBujiDB2.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownBujiAA1.Value * 10)) + ":" +
(Convert.ToInt16(3600 - numericUpDownBujiAA2.Value * 10)) + ":\n");
}
catch
{
    MessageBox.Show("Veri Gonderme Hatasi!!");
}
}
```

```
private void sistemiKapatBtn_Click(object sender, EventArgs e)
{
    sp.Write("kapat" + "\n");
    sp.Close();
    Application.Exit();
}
```

```
private void progressBarDevir_Click(object sender, EventArgs e)
{
}
```

```
private void gidentext_TextChanged(object sender, EventArgs e)
{
}
```

```
private void label35_Click(object sender, EventArgs e)
{
}
}
}
```

ÖZGEÇMİŞ



Ad Soyad : Mehmet İlter Özmen
Doğum Yeri ve Tarihi : 30.05.1990 / İstanbul
E-Posta : m.ilterozmen@gmail.com

ÖĞRENİM DURUMU:

- **Lisans** : 2010, İstanbul Üniversitesi, Mühendislik Fakültesi, Makine Mühendisliği