

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

MİKRODENETLEYİCİ TABANLA ISI KONTROL SİSTEMİNİN TASARIMI

YÜKSEK LİSANS TEZİ

Elektrik ve Elektronik Müh. SİNAN SARIKAYA

Enstitü Anabilim Dalı : ELK.-ELEKTRONİK MÜH.
Enstitü Bilim Dalı : ELEKTRONİK
Tez Danışmanı : Doç. Dr. Ayhan ÖZDEMİR

Eylül 2008

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

MİKRODENETLEYİCİ TABANLA ISI KONTROL
SİSTEMİNİN TASARIMI

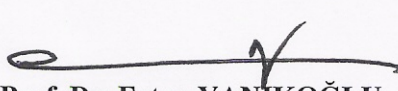
YÜKSEK LİSANS TEZİ

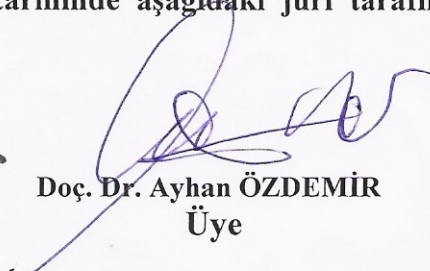
Elektrik ve Elektronik Müh. Sinan SARIKAYA

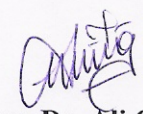
Enstitü Anabilim Dalı : ELK.-ELEKTRONİK MÜH.

Enstitü Bilim Dalı : ELEKTRONİK

Bu tez 05/ 09 /2008 tarihinde aşağıdaki jüri tarafından Oybirliği ile kabul edilmiştir.


Prof. Dr. Ertan YANIKOĞLU
Jüri Başkanı


Doç. Dr. Ayhan ÖZDEMİR
Üye


Yrd. Doç. Dr. Ali GÜLBAĞ
Üye

ÖNSÖZ

Günümüzde yaşamımızın her safhasında ısıtma sistemlerine duyulan ihtiyaç giderek artmakta ve bu konuya yönelik çalışmalar büyük bir önem arz etmeye devam etmektedir. Isıtma enerjisinde tasarruf elde etmek ve bu enerjiyi verimli bir şekilde kullanarak ekonomiye katkıda bulunmak, ülkelerin temel politikası haline gelmiştir. Bu amaçla sanayinin birçok alanında ve evlerimizde kullanılan ısıtma sistemlerinde değişik ısı kontrol yöntemleri geliştirilmiştir. Isı kontrol sistemleri içerisinde doğruluğu, hassasiyeti ve kontrol yeteneği güçlü mikrodnetleyiciler kullanılarak mükemmelliğe yakın ısı kontrol sistemleri tasarlanabilmiştir. Bu çalışmada evlerimizde sıkça kullanmaya başladığımız örnek bir kombi ısıtma sistemi detaylı olarak irdelenmiş ve buna yönelik mikrodnetleyici tabanlı PID kontrol algoritmalı ısı kontrol sistemi tasarımı uygulamalı olarak gerçekleştirilmiştir.

TEŐEKKÜR

Bu projeyi seçmemde öncü olan ve projenin her aşamasında yardımlarını esirgemeyen danışmanım Doç. Dr. Ayhan ÖZDEMİR'e, bu çalışmayı gerçek ve uygulanabilir kılan ve ticari hayattaki özgün tecrübelerini aktaran Mak. Müh. Murat TEKER'e, Mak. Müh. Remzi ÇALIŐKAN'a, Mak. Öğretmeni Emin Erkin BİLALOĞLU'na (AIRFEL A.Ő.) ve proje uygulamasında kombi test ve geliştirme laboratuvarını kullanmama izin veren AIRFEL A.Ő. firmasına, iş hayatımda tezimi hazırlamamda fırsat ve imkan sağlayan müdürüm Mak. Müh. Fahrettin BİLGE'ye, manevi desteklerini esirgemeyen eşim Aslı SARIKAYA'ya ve aileme teşekkür ederim.

İÇİNDEKİLER

ÖNSÖZ.....	ii
TEŞEKKÜR	iii
İÇİNDEKİLER.....	iv
SİMGELERVE KISALTMALAR LİSTESİ.....	viii
ŞEKİLLER LİSTESİ.....	ix
TABLolar LİSTESİ.....	xii
ÖZET.....	xiii
SUMMARY.....	xiv
BÖLÜM 1.	
GİRİŞ.....	1
BÖLÜM 2.	
MİKRODENETLEYİCİLER.....	3
2.1. Mikrodenetleyiciler ve Seçimi.....	3
2.2. Mikrodenetleyici Yazılım Araçları.....	3
2.3. Mikrodenetleyici Mimari Yapısı.....	4
2.4. Mikrodenetleyici Teknik Özellikleri.....	5
2.5. Mikrodenetleyici Pin Konfigürasyonu.....	6
2.6. Mikrodenetleyici Donanım Mimarisinin Blok Diyagramı.....	7
2.7. Mikrodenetleyici İşlem Komutları.....	8
BÖLÜM 3.	
SICAKLIK SENSÖRLERİ.....	9
3.1. Sensörler.....	9
3.2. Sıcaklık Sensörleri ve Çeşitleri.....	9

3.2.1. Termistörler	10
3.2.2. Termocouple sensör	13
3.2.3. RTD (Resistive Temperature Detector)	13
3.2.4. Entegre yapıya sahip sensörler	13

BÖLÜM 4.

PID KONTROL	14
4.1. Oransal Kontrolör	16
4.2. İntegral Kontrolör.....	17
4.3. Türev Kontrolör.....	18
4.4. S-Domenİnde ve Z-Domeninde PID Kontrolör	19
4.5. Ziegler-Nichols Tuning Algoritması	19
4.5". 1. Ziegler-Nichols Açık-Çevrim Tuning	20
4.5.1. Ziegler-Nichols Kapaî-Çevrim Tuning.....	23
4.6. Ayrık Zaman İntegral ve Türevin Elde Edilmesi	23
4.7..Aynk Zaman PID Kontrolör Tasannu	25
4.8. PID Kontrolör Örnekleme Zamanım Belirleme	28
4.9. Standart ve Modifiye Edilmiş'PID Kontrolörlerin Karşılaştırılması	29
4.9.1 Standart PID tasarımı	29
4.9.2.Modi-fiye edilmiş PID kontrolör-ile tasarım	30

BÖLÜM 5.

BİTERMİK KOMBİ ISITMA SİSTEMİ VE SİSTEM DEVRE ELEMANLARI	32
5.1. Kombi- Isıtma Sistemi Devre Şeması ve Devre Elemanları.....	32
5.2. Kombi Sistem Elemanları ve Görevleri	33
5.2.1. Su sirkülasyon pompası	34
5.2.2. Sabasınç prosestatı.....	35
5.2.3. Limit sıcaklık sensörü	36
5.2.4. NTC sıcaklık sensörâ	36
5.2.5. Akışölçer.....	37
5.2.6. Fan	38

5.2.7. Atık gaz basınç prosestatı	39
5.2.8. Gaz valfi ve gaz valfi modülasyon bobini.....	40
5.2.9. Ateşleme elektrodu.....	40
5.2.10. İyonizasyon elektrodu	41
5.2.11. Eşanjör	41
5.2.12. Brülör	42

BÖLÜM 6.

KOMBİ ISITMA SİSTEMİ ELEKTRONİK KONTROL KARTI-TASARIMI.....

6. 1. Elektronik Kontrol Kartı Blok Diyagramı	43
6.1.1. Elektronik kontrol kartı DC güç kaynağı ve filtre tasarımı.....	45
6.1.2. Mikrodenetleyici ve çevre bileşenleri uygulama devresi	55
6.1.3. NTC teraüstör kontrol devresi	58
6.1.4. Açık çevrimde su debisini ölçen flovvmetre sinyal işleme uygulama devresi.....	62
6.1.5. Elektronik kontrolkartı-PC RS-232 haberleşme portu uygulama devresi	63
6.1.6. 220V AC beslemeli ekipmanların röle kontrolü .uygulama devresin.....	64
6.1 .7. Mekanik sensörlerin-on/off kontak durumlarını algılama devresi..	67
6.1.8. Gaz valfi modülasyon bobini kontrol devresi	68
6.1.9. LCD ve tuş takımı uygulama devresi.....	70
6.2. Elektronik Kart Resimleri.....	73

BÖLÜM 7.

SİSTEMİN MATEMATİKSEL MODELİNİN ELDE EDİLMESİ

7.1. Sistemin Modellenmesi	78
7.2. Deneysel Olarak Sistemin Açık Çevrim Transfer Fonksiyonunun Elde Edilmesi.....	79
7.3. PID Kontrolör Tasarımı ve Uygulama Sonucu	87

KAYNAKLAR.....	91
EKLER.....	92
ÖZGEÇMİŞ.....	162

SİMGELER VE KISALTMALAR LİSTESİ

PID	: Oransal integral türev
NTC	: Negatif sıcaklık katsayılı
PTC	: Pozitif sıcaklık katsayılı
ADC	: Analog dijital çevirici

ŞEKİLLER LİSTESİ

Şekil 2.1.	PIC16F877A DIP pin konfigürasyonu.....	6
Şekil 2.2.	PIC16F877A QFN pin konfigürasyonu.....	6
Şekil 2.3.	PIC16F877A donanım mimari yapısının blok diyagramı.....	7
Şekil 3.1a.	NTC sensör.....	10
Şekil 3.1b.	NTC sensör.....	11
Şekil 3.1c.	NTC sensör.....	11
Şekil 3.2.	Tipik bir 10 k Ω NTC sensör sıcaklık-direnç eğrisi.....	12
Şekil 4.1.	Tipik bir PID kontrolör blok diyagramı.....	14
Şekil 4.2.	Dijital ısı kontrol sistemi.....	15
Şekil 4.3.	Oransal Kontrolör giriş çıkış grafiği.....	16
Şekil 4.4.	İntegral Kontrolör giriş çıkış grafiği.....	17
Şekil 4.5.	Türevsel Kontrolör giriş çıkış grafiği.....	18
Şekil 4.6.	Açık-çevrim yöntemiyle bir prosesin K , T_d ve T_i parametrelerini bulmak.....	20
Şekil 4.7.	Termal bir sistemin açık-çevrim cevabı.....	21
Şekil 4.8.	Termal bir sistemin kapalı-çevrim blok diyagramı.....	22
Şekil 4.9.	Ziegler-Nichols kapalı-çevrim testi.....	23
Şekil 4.10.	Sayısal olarak integral alma yöntemi.....	24
Şekil 4.11.	PID kontrolör paralel gerçekleştirme diyagramı.....	25
Şekil 4.12.	Pratikte kullanılacak bir PID kontrolör gerçekleştirme diyagramı	27
Şekil 4.13.	Standart PID kontrolör tasarımı.....	29
Şekil 4.14.	Modifiye edilmiş PID kontrolör tasarımı.....	30
Şekil 5.1.	Bitermik kombi sistem şeması ve devre elemanları.....	31
Şekil 5.2.	Günümüzde üretilen farklı kombi tipleri.....	34
Şekil 5.3.	Bir kombinin iç görünüşü.....	34
Şekil 5.4.	Kapalı çevrim su sirkülasyon pompası.....	35

Şekil 5.5.	Limit sıcaklık sensörü.....	36
Şekil 5.6.	NTC sıcaklık sensörü.....	36
Şekil 5.7.	Honeywell T7335 NTC termistör sıcaklık-direnç tablosu.....	37
Şekil 5.8.	FUGAS-TURBINA-SPT102 flowmetre.....	38
Şekil 5.9.	FUGAS-TURBINA-SPT102 flowmetre su akış miktarı-pulse ilişkisi.....	38
Şekil 5.10.	FIME atık gaz fanı.....	39
Şekil 5.11.	HUBA 80/60 atık gaz basınç prosestatı.....	39
Şekil 5.12.	HONEYWELL Moduplus VK 4105 gaz valfi.....	40
Şekil 5.13.	Ateşleme/iyonizasyon elektrodu.....	41
Şekil 5.14.	Eşanjör.....	42
Şekil 5.15.	Tipik bir brülör.....	42
Şekil 6.1.	Kombi ısıtma sistemi elektronik kontrol kartı blok diyagramı.....	44
Şekil 6.2.	Sistem güç kaynağı devresi.....	46
Şekil 6.3.	Dalgalı DC gerilimin düzgün hale getirilmesi.....	47
Şekil 6.4.	TNY278PN blok diyagramı.....	51
Şekil 6.5.	TNY278PN pin konfigürasyonu.....	52
Şekil 6.6.	TNY278PN entegresi maksimum yüke yakın değerinde davranışı..	53
Şekil 6.7.	PC817 optokuplör iç devre yapısı.....	53
Şekil 6.8.	Elektronik kontrol kartı mikrokontrolör ve çevre bileşenleri, NTC sensör, RS-232, flowmetre, röle kontrol entegresi, LCD ve tuş takımı uygulama devreleri.....	56
Şekil 6.9.	MCLR uygulama devresi.....	57
Şekil 6.10.	10K NTC uygulama devresi.....	58
Şekil 6.11.	Flowmetre uygulama devresi.....	62
Şekil 6.12.	FUGAS-TURBINA-SPT102 flowmetre su akış miktarı-pulse ilişkisi.....	62
Şekil 6.13.	Stamp Plot Lite Version 7.0 veri izleme programı.....	64
Şekil 6.14.	On/off sivici algılama devresi, röle kontrol uygulama devresi devamı ve gaz valfi modülasyon bobin PWM uygulama devresi	65
Şekil 6.15.	ULN2803A entegre blok diyagramı.....	66
Şekil 6.16.	ULN2803A darlington transistör devre şeması.....	66
Şekil 6.17.	On-off kontak durumu algılama devresi.....	67

Şekil 6.18.	PWM sinyali.....	69
Şekil 6.19.	LCD display ön ve arka yüzey genel görünümü.....	71
Şekil 6.20.	LCD display ve tuş takımı uygulama devresi.....	72
Şekil 6.21.	Ana kart ve keypad.....	74
Şekil 6.22.	Ana kart PCB ön yüzey.....	75
Şekil 6.23.	Ana kart PCB arka yüzey.....	76
Şekil 6.24.	Keypad.....	77
Şekil 7.1.	Sistemin açık çevrim ünitesi transfer fonksiyonun elde edilmesi..	78
Şekil 7.2.	Açık-çevrim yöntemiyle bir prosesin K , Td ve T_1 parametrelerini bulmak.....	79
Şekil 7.3.	Kombi test laboratuvarı kontrol istasyonu.....	81
Şekil 7.4.	Elektronik kart-kombi sistem elemanları kablo bağlantıları.....	82
Şekil 7.5.	Sistem test çalışmasından edilen bilgilerin incelenmesi.....	83
Şekil 7.6.	Sıcaklık bilgisinin bilgisayar ortamında takip edilmesi ve değerlendirilmesi.....	84
Şekil 7.7.	Sistemin açık çevrim transfer fonksiyon cevabı.....	85
Şekil 7.8.	PID kontrolör sonucu elde edilen proses sıcaklık grafiği.....	89

TABLULAR LİSTESİ

Tablo 2.1.	PIC16F877A mikrodenetleyicisinin diğer PIC16F ailesi mikrodenetleyicilerle kıyaslanması.....	5
Tablo 2.2.	PIC16F877A komut düzeni.....	8
Tablo 3.1.	Sıcaklık sensörlerinin mukayesesi.....	9
Tablo 4.1.	Ziegler-Nichols tarafından önerilen kontrolör ayar tablosu.....	21
Tablo 6.1.	Honeywell T7335 NTC termistör sıcaklık-direnç tablosu.....	61
Tablo 7.1.	Açık çevrim transfer fonksiyon cevabı sıcaklık değerleri.....	86

ÖZET

Anahtar kelimeler: NTC sensörlü ısı kontrol sistemi ve tasarımı, PID ısı kontrol sistemi ve tasarımı, mikrodenetleyici tabanlı ısı kontrol sistemi ve tasarımı,

Mikrodenetleyici tabanlı ve dijital PID kontrol algoritmaları yazılım kullanılarak tasarlanan ısı kontrol sistemi kullanıcıya birçok avantaj ve üstünlük sağlamaktadır. Sanayide veya evlerimizde sıcaklığı kontrol edilen prosesin ısı değişimlerine, dijital tabanlı PID kontrol algoritmasıyla lineerlik kazandırılarak hassasiyeti ve doğruluğu yüksek, kontrolü kolay bir kontrol sistemi elde edilir. Bu yöntemle elde edilen kazanımları örneklersek; endüstride ekonomik ve katma değeri yüksek kaliteli ürünlerin elde edilmesini, evlerimizde kullanılan ısıtma amaçlı sistemlerin kararlı bir sıcaklıkta çalışmasını sağlayarak insan sağlığına olan olumlu ve üstün yanlarını sayabiliriz.

Sıcaklık kontrolünün, insan hayatında ve sanayideki birçok süreçte önemli bir yeri olduğundan, bu alanda yapılan ve yapılmaya devam edilen çalışmaların değeri her zaman artmaya devam edecektir ve bilimsel anlamda önemini her zaman koruyacaktır.

THE DESIGN OF TEMPERATURE CONTROL SYSTEM BASED ON MICROCONTROLLER

SUMMARY

Key Words: Temperature control system and design with NTC sensor, PID temperature control system and design, The design of temperature control system based on microcontroller

Temperature control system designed by using digital PID control algorithm and based on microcontroller supplies to user many advantages and superiority. We obtain a temperature control system with high accuracy and sensitivity by using digital PID control algorithm. This control system supplies a linearity to the temperature change in systems controlled its temperature in industry and our home. For example, with this control ,we can count these utilities : producing economic goods with high quality, obtaining process with stable temperature, supplying that people live in an environment with stable temperature in heating their house for a good health.

Because of the temperature control is very important subject for our life and for many processes in industry, the importance and value of scientific workings that will be done in temperature control area will always continue to increase.

BÖLÜM 1. GİRİŞ

Isıtma sistemleri gün geçtikçe farklı enerji kaynaklarının kullanıldığı ve kontrol edildiği birçok proste karşımıza çıkmaktadır. Bu enerji kaynakları, elektrik enerjisi, LPG, doğalgaz, petrol ve petrol türevleri gibi kaynaklardır. Enerjinin verimli şekilde kullanılması temel amaç olduğundan, gelişmiş ısı kontrol denetleyicilerine ihtiyaç duyulmaktadır.

Bu projede maksimum ısı verimliliğini sağlamak için kontrol açısından esneklik ve doğruluğun kolaylıkla elde edilebildiği mikrodnetleyici kullanılmıştır. Mikrodnetleyiciler, esnek bir kontrol yapısına ve donanım olarak tüm devre üzerinde farklı kontrol yapılarına sahip olduğundan günümüzdeki tüm akıllı sistemlerde vazgeçilmez hale gelmiştir. Bu yolla birçok karmaşık işlemler kolay kontrol edilebilir hale getirilmiştir. Mikrodnetleyicide kullanılan kontrol algoritması bu projede en önemli yeri tutmaktadır. Dijital PID kontrolör kullanılarak istenen dinamik performans ve sürekli-hal cevabı elde edilmeye çalışılmıştır.

Sıcaklık kontrolünde, kontrolör kadar sistem ısısının doğru şekilde ölçülmesi ve ortam sıcaklığı algılayan sensör seçimi de ısı kontrol sisteminin önemli bir ayağını oluşturmaktadır. Bu amaçla algılayıcı olarak, doğruluğu yüksek ve maliyet açısından uygun olan, maliyeti artıracak harici devre bileşenlerine gereksinim duymayan NTC sensör seçilmiştir.

Ülkemizde doğalgaz kullanımının yaygınlaşmaya başladığı günümüzde, evlerimizde ısıtma sistemi olarak kullandığımız kombi ismini sıkça duymaya başladık. Kombi ısıtma sistemi alanında, kontrolör tasarımı açısından ülkemizde son derece bakir bir

alan olduđu ve tamamen dıřa bađımlı bir yapının olduđu tespit edilmiřtir. Bu ađıđı kapatmak adına kombi ısıtma sistemi bu projede konu olarak ele alınmıřtır ve bu konu detaylı olarak irdelenmiřtir.

Proje teorik bilgiler yanında uygulama devresiyle 6n plana ıkmıřtır ve gerek sistem elemanlarıyla birlikte kullanılarak bilimsel ve ticari bir boyut kazandırılmaya alıřılmıřtır.

BÖLÜM 2. MİKRODENETLEYİCİLER

2.1. Mikrodenetleyiciler ve Seçimi

Mikrodenetleyici seçiminde piyasada kolay bulunabilirlik, performans/fiyat oranının yüksekliği, analog bilgileri dijital bilgiye çeviren tüm devre üzeri ADC giriş sayısı, kullanıcı değişken bilgilerinin kalıcı olarak hafızada tutan EEPROM hafıza ve hafıza boyutu, kullanıcı programının tutulduğu program hafıza boyutu, harici devrelerle ve PC gibi elektronik ekipmanlarla haberleşmeyi sağlayan asenkron haberleşme portu, bilgilerin geçici olarak saklandığı RAM hafıza boyutu, giriş ve çıkış port sayısının çokluğu gibi birçok kriter göz önüne alınarak Microchip firmasının üretmiş olduğu PIC16F877A işlemcisi seçilmiştir [1].

Mikrodenetleyicilerin programlanması için kullanılan programlama kartının piyasada kolayca elde edilmesi ve makine kodlarının mikrodenetleyiciye gönderilmesini sağlayan yazılımın ve simülasyon programının ücretsiz olması mikrodenetleyici seçiminde ön plana çıkmıştır.

2.2. Mikrodenetleyici Yazılım Araçları

Yazılımın derlenmesi ve simülasyonun yapılması amacıyla, Microchip firması tarafından geliştirilen ve ücretsiz sağlanan MPASM derleyici programı ve MPLAB simülasyon programı kullanılmıştır. Makine kodlarının işlemciye gönderilmesi amacıyla IC-PROG yazılımı kullanılmıştır.

2.3. Mikrodenetleyici Mimari Yapısı

PIC16F877A mikrodenetleyicisi RISC(Reduced Instruction Set) mimarili yapıya sahiptir. Bu yapının avantajı adından da anlaşılacağı gibi “azaltılmış komut seti mimarisi” ile tasarlanarak, toplam 35 adet komutla yazılım yapılabilmesidir. Komut sayısı az olduğundan işlem kodu 14 bit uzunluğundadır. İşlem kodunun düşük değerli 8 bit’i data, yüksek değerli 6 bit’i işlem kodudur. İşlem kodunun 6 bit olması maksimum 64 adet komut kullanılmasına imkan sağlarken, sadece 35 adet komut kullanılmıştır. Komut sayısının az olması yeni kullanıcıların programlamayı kısa sürede öğrenmesini ve program yazmalarını kolaylaştırmaktadır. Fakat komut sayısının az olması bazı işlemleri yapmada, program yükünü ve program yazma süresini arttırmaktadır.

2.4. Mikrodenetleyici Teknik Özellikleri

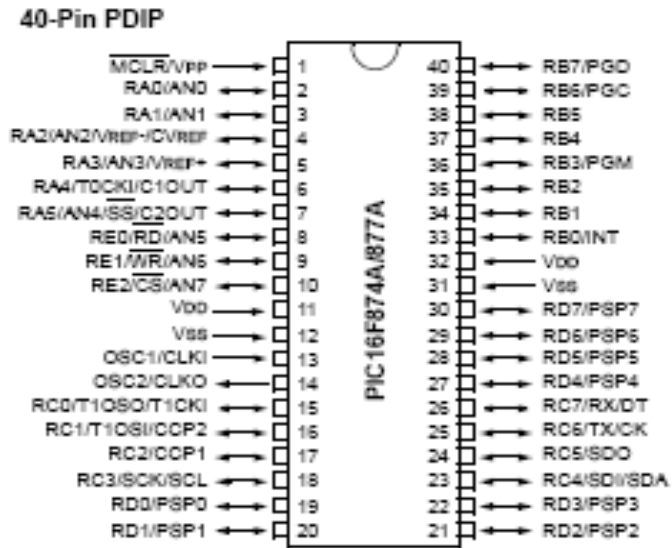
1. 35 adetlik komut seti
2. CALL, GOTO, BTFSS-BTFSC-INCFSZ-DECFSZ(atlama yapılmazsa), RETLW, RETURN, RETFIE komutları haricindeki tüm komutların tek saykılta yapılması
3. Maksimum 20 MHz osilatör frekansı
4. 8Kx14 program hafıza alanı
5. 366x8 data hafıza (RAM)
6. 256x8 elektrikle yazılabilir silinebilir hafıza alanı (EEPROM)
7. 3 adet zamanlayıcı ve sayıcı: Timer0: 8-bit zamanlayıcı/sayıcı, Timer1: 16-bit zamanlayıcı/sayıcı, Timer2: 8/16-bit zamanlayıcı
8. 2 adet Compare (12 bit) / Capture (12 bit) / PWM modül (10 bit)
9. PSP, SPI ve I²C haberleşme portu

10. Universal asenkron haberleşme portu
11. 10-bit çözünürlüklü 8 adet ADC kanalı
12. Brown-out reset (BOR), power-on reset (POR)
13. Tüm devre üzeri RC osilatörlü 18msn*128 watch-dog timer (WDT)
14. Güç tasarrufu için Sleep modu
15. Tüm devre üzeri 2 adet analog karşılaştırıcı giriş kanalı
16. Tüm pinler elektrostatik deşarj korumalı
17. Düşük güç tüketimi
18. Tüm pinler 25 mA çıkış akımlı
19. Düşük gerilim aralığında çalışma (2V..5V)
20. Yazılım kod koruması

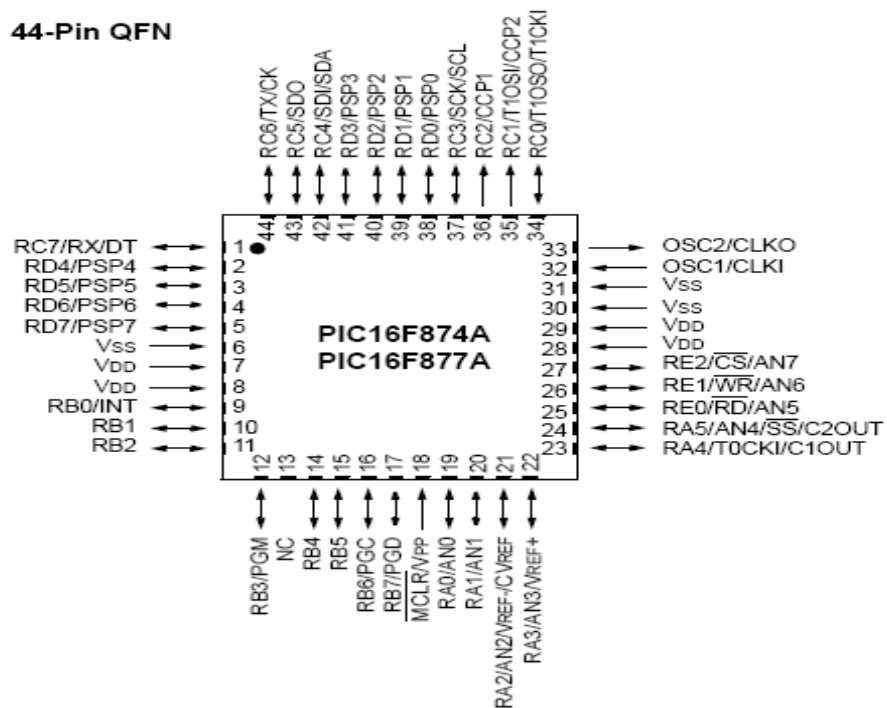
Tablo 2.1. PIC16F877A mikrodenetleyicisinin diğer PIC16F ailesi mikrodenetleyicilerle kıyaslanması

Key Features	PIC16F873A	PIC16F874A	PIC16F876A	PIC16F877A
Operating Frequency	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz	DC – 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
Flash Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory (bytes)	128	128	256	256
Interrupts	14	15	14	15
I/O Ports	Ports A, B, C	Ports A, B, C, D, E	Ports A, B, C	Ports A, B, C, D, E
Timers	3	3	3	3
Capture/Compare/PWM modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Analog Comparators	2	2	2	2
Instruction Set	35 Instructions	35 Instructions	35 Instructions	35 Instructions
Packages	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN	28-pin PDIP 28-pin SOIC 28-pin SSOP 28-pin QFN	40-pin PDIP 44-pin PLCC 44-pin TQFP 44-pin QFN

2.5. Mikrodenetleyici Pin Konfigürasyonu

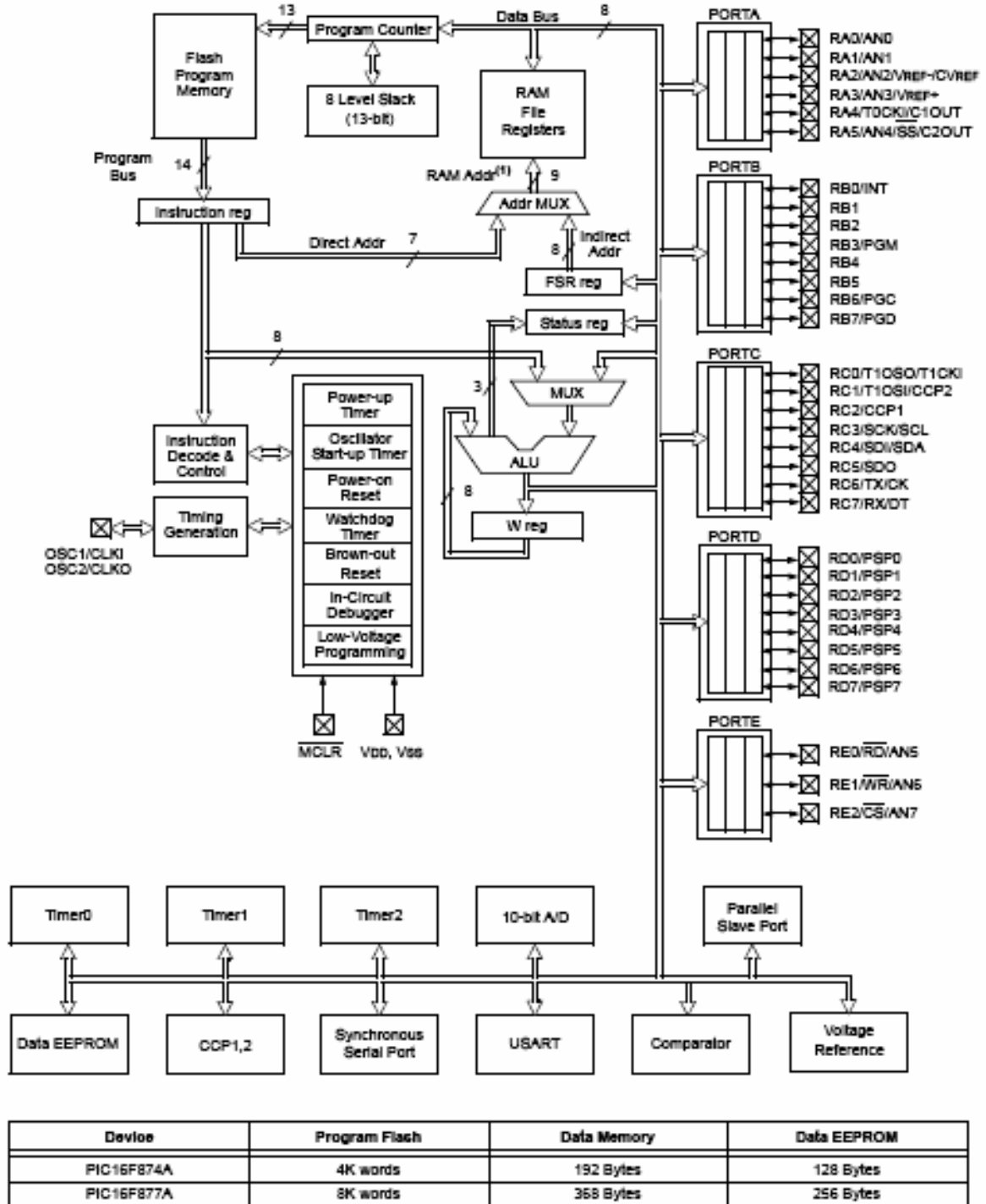


Şekil 2.1. PIC16F877A DIP pin konfigürasyonu



Şekil 2.2. PIC16F877A QFN pin konfigürasyonu

2.6. Mikrodenetleyici Donanım Mimarısının Blok Diyagramı



Şekil 2.3. PIC16F877A donanım mimari yapısının blok diyagramı

2.7. Mikrodenetleyici İşlem Komutları

Tablo 2.2. PIC6F877A komut düzeni

Mnemonic, Operands	Description	Cycles	14-Bit Opcode		Status Affected	Notes	
			MSb	LSb			
BYTE-ORIENTED FILE REGISTER OPERATIONS							
ADDWF	f, d	Add W and f	1	00 0111	ffff ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00 0101	ffff ffff	Z	1,2
CLRF	f	Clear f	1	00 0001	1fff ffff	Z	2
CLRWF	-	Clear W	1	00 0001	0xxx xxxxx	Z	
COMF	f, d	Complement f	1	00 1001	ffff ffff	Z	1,2
DECF	f, d	Decrement f	1	00 0011	ffff ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00 1011	ffff ffff		1,2,3
INCF	f, d	Increment f	1	00 1010	ffff ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00 1111	ffff ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00 0100	ffff ffff	Z	1,2
MOVF	f, d	Move f	1	00 1000	ffff ffff	Z	1,2
MOVWF	f	Move W to f	1	00 0000	1fff ffff		
NOP	-	No Operation	1	00 0000	0xxx0 0000		
RLF	f, d	Rotate Left f through Carry	1	00 1101	ffff ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00 1100	ffff ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00 0010	ffff ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00 1110	ffff ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00 0110	ffff ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS							
BCF	f, b	Bit Clear f	1	01 00bb	bfff ffff		1,2
BSF	f, b	Bit Set f	1	01 01bb	bfff ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1(2)	01 10bb	bfff ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1(2)	01 11bb	bfff ffff		3
LITERAL AND CONTROL OPERATIONS							
ADDLW	k	Add literal and W	1	11 111x	kkkk kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11 1001	kkkk kkkk	Z	
CALL	k	Call subroutine	2	10 0kkk	kkkk kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00 0000	0110 0100	$\overline{TO}, \overline{PD}$	
GOTO	k	Go to address	2	10 1kkk	kkkk kkkk		
IORLW	k	Inclusive OR literal with W	1	11 1000	kkkk kkkk	Z	
MOVLW	k	Move literal to W	1	11 00xx	kkkk kkkk		
RETFIE	-	Return from interrupt	2	00 0000	0000 1001		
RETLW	k	Return with literal in W	2	11 01xx	kkkk kkkk		
RETURN	-	Return from Subroutine	2	00 0000	0000 1000		
SLEEP	-	Go into Standby mode	1	00 0000	0110 0011	$\overline{TO}, \overline{PD}$	
SUBLW	k	Subtract W from literal	1	11 110x	kkkk kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11 1010	kkkk kkkk	Z	

BÖLÜM 3. SICAKLIK SENSÖRLERİ

3.1. Sensörler

Sensörler sıcaklık, basınç, ışık, nem, hareket vb. fiziksel bir büyüklüğü elektriksel bir işarete dönüştüren elemanlardır. Elde edilen elektriksel büyüklük analog bir büyüklüktür ve bu büyüklük elektronik devrelerle anlam kazandırılarak bizlerin anlayabileceği şekle dönüştürülür.

3.2. Sıcaklık Sensörleri ve Çeşitleri

Sistem sıcaklığını ölçmek için farklı fonksiyonel yapıya sahip, aktif ve pasif sıcaklık sensörleri kullanılmaktadır. Sıcaklık sensörleri; termistör, termocouple, RTD (resistive temperature detector), entegre sensör gibi farklı yapılarda imal edilmektedir. Günümüzde sıcaklık sensörlerinin çıkış sinyalleri lojik çıkışlı, voltaj çıkışlı (analog gerilim) veya dijital seri data sinyal çıkışlıdır. Bu sensör türleri arasında seçim yapmak gerektiğinde bazı kriterler göz önünde bulundurularak amaca uygun sensör seçilir. Bu kriterler; yüksek doğruluk, düşük güç harcaması, harici devre elemanlarının azlığı ve kontrol edilen sistemin ortam sıcaklık aralığı.

Tablo 3.1. Sıcaklık sensörlerinin mukayesesi

Sensör Tipi	Sıcaklık Ölçüm Aralığı, °C	Tolerans ± °C	Maliyet	Dayanıklılık
Termokuplör	-270 / + 2600	1	Yüksek	Yüksek
RTD	-200 / + 600	0.2	Yüksek	Düşük
Entegre Sensör	-40 / + 125	1	Orta	Düşük
Termistör	-50 / + 200	0.2	Düşük	Orta

3.2.1. Termistörler

Termistörler direnç değeri sıcaklıkla değişen elemanlardır. İki farklı yapıda bulunmaktadır: negatif ısı katsayısına sahip, sıcaklık arttıkça direnç değeri azalan NTC sensör ve pozitif ısı katsayısına sahip, sıcaklık arttıkça direnç değeri artan PTC sensör. Termistörlerin sıcaklık-direnç eğrisi non-lineer yapıda olduğundan dolayı, doğruluk ve lineerleştirme açısından, düzeltme tablosu kullanmak gerekmektedir.

Termistörlerin tipik sıcaklık aralığı genelde $-25\text{ }^{\circ}\text{C}$ ile $150\text{ }^{\circ}\text{C}$ 'dir. Termistörler fiyat ve harici devre elemanları açısından maliyeti en ucuz sensör çeşididir. Termistörlerin ani sıcaklık değişimlerine yanıtı hızlıdır.

PTC sensörler birçok uygulamada yüksek sıcaklığın istenmediği devrelerde, uzun kablolu sistemlerde kullanılmaktadır. PTC sensör örneğin sanayide kullanılan yüksek akımlı trafoların ve tristörlerin çalışma sıcaklığının belirlenen limit sıcaklık değerinin üzerine çıktığı durumlarda, sistemi koruma amaçlı limit siviç olarak kullanılmaktadır. Normalde düşük direnç, yüksek sıcaklıkta yüksek direnç gösterir

NTC sensörler ise daha çok elektronik devrelerde kullanılmaktadırlar. Sıcaklık arttığında direnç değeri azaldığı için mikrodenetleyicili uygulamalarda kullanımı, devre tasarımı açısından daha uygundur. Aşağıda şekillerde değişik yapıdaki NTC sensörler ve NTC sensör sıcaklık-direnç eğrisi gösterilmiştir:



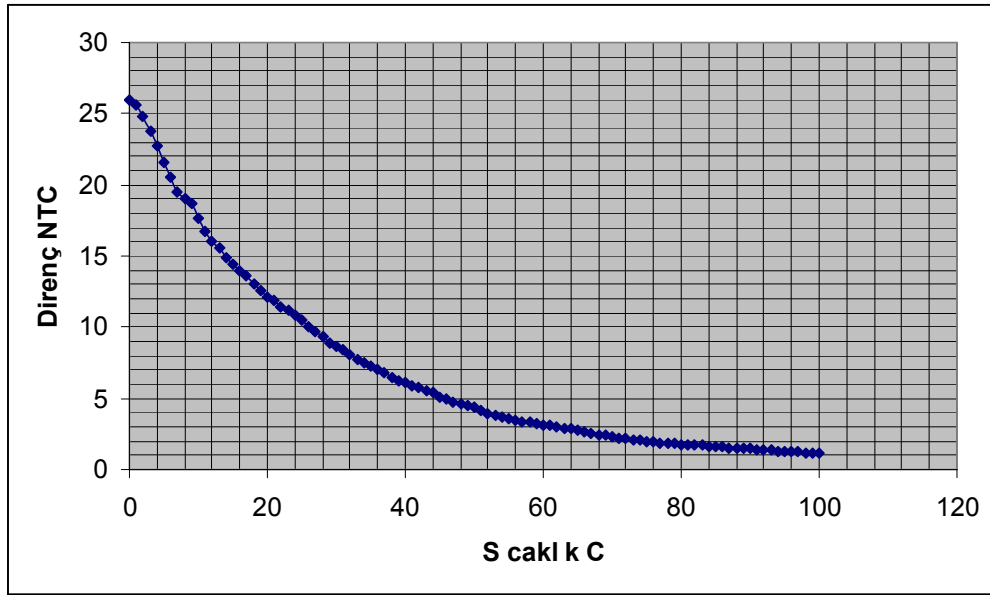
Şekil 3.1a. NTC sensör



Şekil 3.1b. NTC sensör

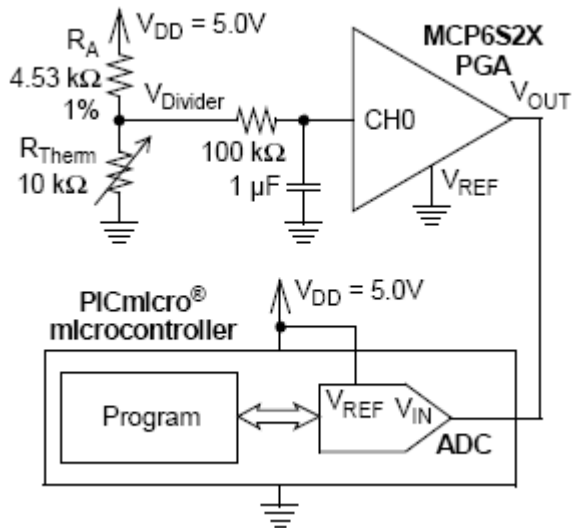


Şekil 3.1c. NTC sensör



Şekil 3.2. Tipik bir 10 kΩ NTC sensör sıcaklık-direnç eğrisi

Aşağıda tipik bir NTC sensör uygulama devresi gösterilmiştir [2].



Şekil 3.3. NTC uygulama devresi

Yukarıdaki devrede NTC sensör çıkış gerilimi:

$$V_{\text{Divider}} = (R_{\text{ntc}} / (4.53k + R_{\text{ntc}})) * V_{\text{DD}} \quad [3.1]$$

100 kohm'luk direnç ve 1uF'lık kondansatörden oluşan paralel devre, ani sıcaklık değişimlerine ve parazitlere karşı RC alçak geçiren filtre olarak kullanılmıştır.

3.2.2. Termocouple sensör

Termokupıl sensör genelde geniş ve yüksek sıcaklık değerlerinde çalıştığından dolayı tercih edilmektedirler. Çalışma sıcaklıkları -270 °C ile 1750 °C arasındadır. Dayanıklılık ve uygun fiyat da ayrı bir tercih sebebidir. Bununla beraber sıcaklık-direnç eğrisi yüksek derecede non-lineer bir yapıya sahiptir. Bu sebeple detaylı bir lineerleştirme algoritmasına gereksinim duyar.

3.2.3. RTD (Resistive Temperature Detector)

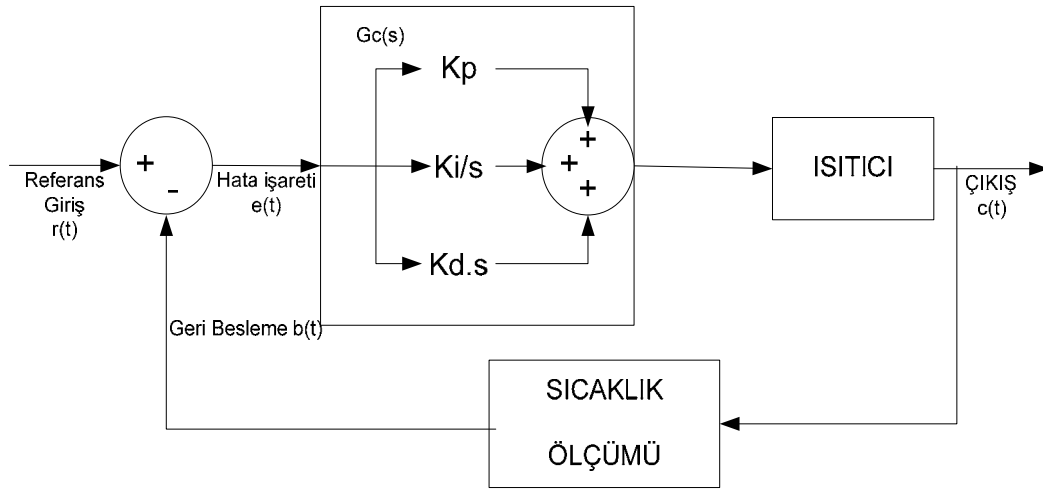
RTD'ler orta değerdeki sıcaklık ölçümlerinde kullanılır. Ölçüm aralığı -200 °C ile +850 °C arasındadır. Ölçüm doğruluğu termokuplörlerden daha hassastır. Toleransı +/- 0.2 °C dır. Termokuplör kadar dayanıklı değildir. Non-lineer bir yapıya sahiptir.

3.2.4. Entegre yapıya sahip sensörler

Sıcaklık ölçüm aralığı -40 °C ile + 125 °C arasındadır. Toleransı +/- 1 °C civarındadır. Değişik elektriksel çıkış sinyallerine sahiptir. SPI, I2C gibi seri data haberleşme yapısına, lojik çıkış verebilme özelliklerine sahiptirler. Örneğin LM35A sensörü 10mv/ 1°C bir çıkış voltaj değeri üretmektedir. Harici devre elemanlarına gereksinim yok denecek kadar azdır.

BÖLÜM 4. PID KONTROL

PID kontrol endüstride sık olarak kullanılan bir kontrol algoritmasıdır. Oransal(Proportional), integral(Integral) ve türevsel(Derivative) kelimelerinin baş harfleriyle adlandırılmıştır.



Şekil 4.1. Tipik bir PID kontrolör blok diyagramı

PID kontrolör istenen referans sıcaklık ile gerçek sistem sıcaklığı arasındaki farkı ölçer ve bu fark değeriyle sürücüye ne kadar bir gücün aktarılacağını hesaplar. Fark değeri ne kadar az ise çıkışa aktarılacak güç o kadar az olacaktır. Referans sıcaklığına yakın bir değerde güç kademeli olarak azaltılır.

PID kontrolöründe genel olarak kullanılan üç terim vardır. Bunlar: K_p , T_i ve T_d terimleridir. PID kontrolör mevcut kontrol edilen sinyalin hata değerini, belli bir zaman aralığındaki hata sinyalinin integralini ve türevini hesaplar. Bu

hesaplamalarla, ne kadarlık bir düzeltmenin ve düzeltme zamanının uygulanacağı belirlenir [4].

Sürekli zaman domeninde PID algoritması şu formülle belirtilir:

$$u(t) = K_p.e(t) + \frac{K_p}{T_i} \int_0^t e(t)dt + K_p.T_D \frac{de(t)}{dt} \quad [4.1]$$

$u(t)$: Kontrolör çıkışı

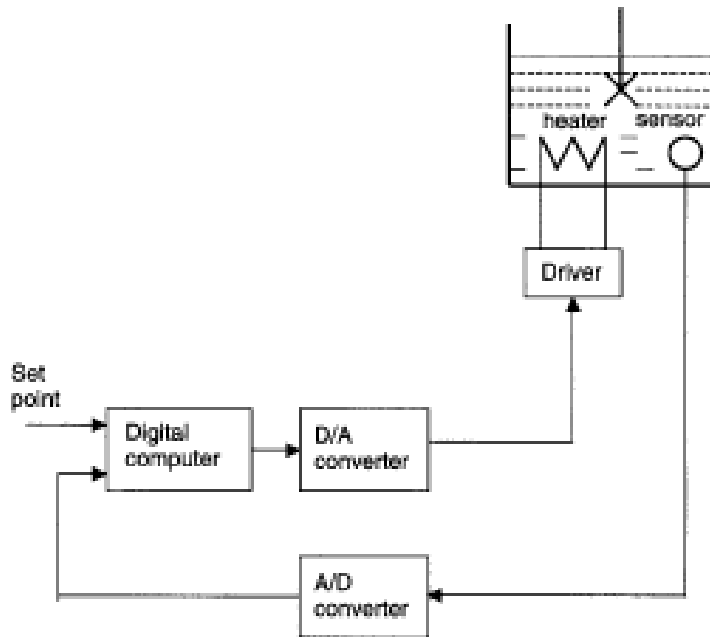
$e(t)$: Hata sinyali

K_p : Oransal kazanç

T_i : İntegral zaman sabiti

T_d : Türev zaman sabiti

Şekil 4.2’de dijital tabanlı bir ısı kontrol sistemi gösterilmiştir.



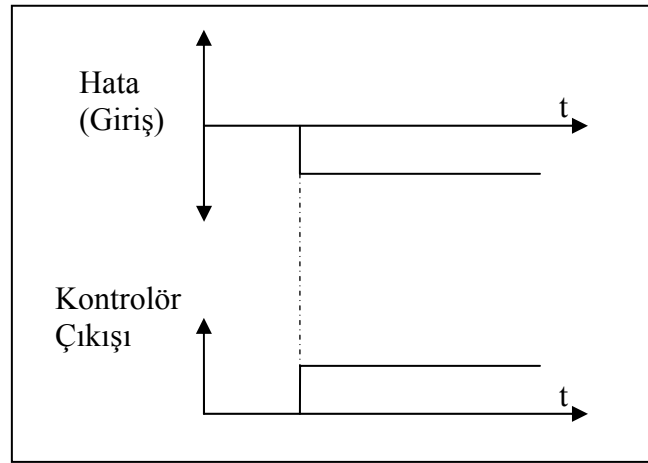
Şekil 4.2. Dijital ısı kontrol sistemi

4.1. Oransal Kontrolör

Oransal kontrolörün çıkışı, kontrolör girişindeki hata değerinin belli bir katsayı (K_p) ile çarpımı sonucunda elde edilir. Giriş sinyali olarak hata sinyali, $e(t)$ kullanılır.

$$X(t) = K_p.e(t) \quad [4.2]$$

Matematiksel olarak [4.2] denkleminle ifade edilebilir. K_p oransal kontrolörün kazancını gösterir.



Şekil 4.3. Oransal Kontrolör giriş çıkış grafiği

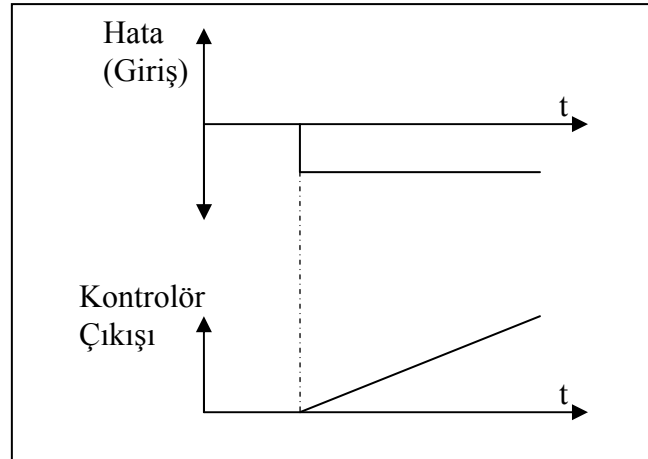
Oransal kontrolörü sabit kazançlı bir kuvvetlendirici gibi düşünebiliriz. Uygulaması basit olan oransal kontrolörde her zaman kararlı hal hatası vardır ve hatanın boyutu sistemden sisteme değişir. Oransal kontrolörün transfer fonksiyonu [4.3] denklemdir.

$$G_c(s) = K_p \quad [4.3]$$

Oransal kazancı artırmak, yükselme zamanını azaltır fakat sürekli durum hatasını yok etmez. Aynı zamanda aşımı artıracaktır.

4.2. İntegral Kontrolör

İntegral kontrolörde çıkış hata değerinin integralinin belirli bir katsayı ile çarpımıdır.



Şekil 4.4. İntegral Kontrolör giriş çıkış grafiği

Matematiksel olarak ;

$$X(t) = \frac{Kp}{Ti} \int_0^t e(t) dt = Ki \int_0^t e(t) dt \quad [4.4]$$

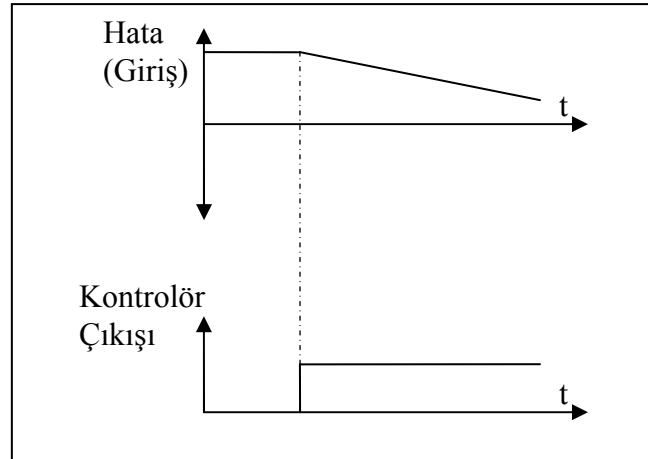
[4.4] eşitliğiyle gösterilebilir. Ki katsayısı integralin kazancıdır. İntegral kontrolör kararlı hal hatasını yok edebilmek için sistem tipini 1 artırır. İntegral kontrolörün transfer fonksiyonu [4.5] denklemleriyle gösterilir.

$$Gc(s) = \frac{Ki}{s} \quad [4.5]$$

İntegral kontrolör sürekli durum hatasını azaltma etkisi gösterir, fakat geçici durum cevabını kötüleştirir. Aşırı integral geniş aşımara yol açar ve osilasyon oluşturur.

4.3. Türev Kontrolör

Türev kontrolörde çıkış hata değerinin türevinin belirli bir katsayı ile çarpımıdır.



Şekil 4.5. Türevsel Kontrolör giriş çıkış grafiği

Matematiksel olarak ;

$$X(t) = K_p.TD \frac{de(t)}{dt} = K_d \frac{de(t)}{dt} \quad [4.6]$$

[4.6] eşitliğiyle gösterilebilir. K_d katsayısı türevin kazancıdır. Türev kontrolör hata sabit ise veya çok az değişim gösteriyorsa bu hatayı giderecek işaret üretmez. İntegral kontrolör yada oransal kontrolör ile beraber kullanılır. Türev kontrolörü gürültülere karşı duyarlıdır, aşımı ve osilasyonu azaltır fakat kararlı hal hatasına etkisi yoktur.

Türev kontrolörün transfer fonksiyonu [4.7] denklemiyle gösterilir.

$$G_c(s) = K_d.s \quad [4.7]$$

Türev kontrolör sistem kararlılığını arttırır, aşmayı azaltır ve sistemin geçici durum cevabını düzeltir. Türevi arttırmak hem aşmayı hemde yerleşme zamanını azaltacaktır.

4.4. S-Domeninde ve Z-Domeninde PID Kontrolör

S-domeninde PID kontrolör aşağıdaki formülle gösterilir:

$$U(s)=K_p.[1+\frac{1}{T_i.S}+TD.S].E(s) \quad [4.8]$$

K_p , T_i , TD PID kontrolör parametreleri birbiriyle etkileşim halindedir. Bu parametrelere uygun değerler vererek en iyi performansı elde edebilmek zor bir görev olabilir.

Ayrık PID kontrolör formu, z-transfer denklemini bularak oluşturulabilir. Bu denklem:

$$U(z)=E(z).K_p.[1+\frac{T}{T_i.(1-z^{-1})}+TD.\frac{(1-z^{-1})}{T}] \quad [4.9]$$

4.5. Ziegler-Nichols Tuning Algoritması

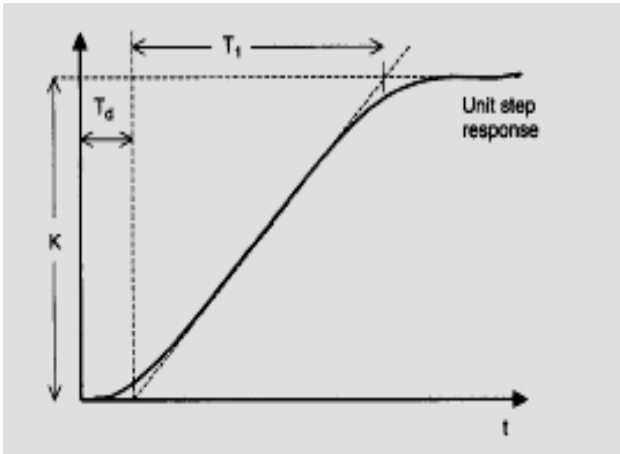
Ziegler-Nichols algoritması açık-çevrim ve kapalı-çevrim testleri üzerine kurulu PID parametre değerlerini önerir.

4.5.1. Ziegler-Nichols Açık-Çevrim Tuning

Ziegler-Nichols'a göre açık-çevrim işlemi yaklaşık transfer fonksiyonu;

$$G(s) = \frac{K.e^{-s.T_d}}{(1 + s.T_1)} \quad [4.10]$$

K , T_d ve T_1 parametreleri aşağıda gösterilen kontrol edilen sistemin açık-çevrim cevabıyla bulunabilir.



Şekil 4.6. Açık-çevrim yöntemiyle bir prosesin K , T_d ve T_1 parametrelerini bulmak

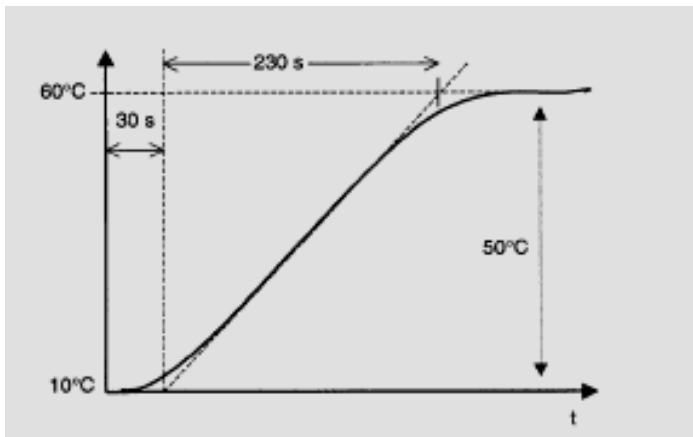
Ziegler-Nichols bir sonraki adımda, referans noktasında bir adım değişimi uygulamasından sonra mutlak hatanın integralini minimum yapmak için Tablo 4.1.'de gösterilen kontrolör ayarlarını önerir.

Tablo 4.1. Ziegler-Nichols tarafından önerilen kontrolör ayar tablosu

KONTROLÖR	Kp	Ti	TD
P	$\frac{T_1}{K.Td}$		
PI	$\frac{0,9.T_1}{K.Td}$	3,3. Td	
PID	$\frac{1,2.T_1}{K.Td}$	2.Td	0,5.Td

PID kontrolör transfer fonksiyonu:

$$U(s)=K_p.[1+\frac{1}{T_i.s}+TD.s].E(s) \quad [4.11]$$



Şekil 4.7. Termal bir sistemin açık-çevrim cevabı

Yukarıdaki şekilde $Td=30sn$, $T_1=230sn$ bulunur. $K=50$ varsayalım.

Transfer fonksiyonu:

$$G(s)=\frac{K.e^{-s.Td}}{(1+s.T_1)}=\frac{50.e^{-30s}}{(1+230s)}$$

Tablo 4.1’de Ziegler-Nichols’un önerdiği ayarlara göre;

$$K_p = \frac{1,2.T_1}{K.Td}, \quad T_i = 2.Td, \quad TD = 0,5.Td \text{ ‘dir.}$$

Bu durumda elde edilen ayarlar:

$$K_p = \frac{1,2.T_1}{K.Td} = \frac{1,2 \times 230}{50 \times 30} = 0,184$$

$$T_i = 2 \times Td = 2 \times 30 = 60$$

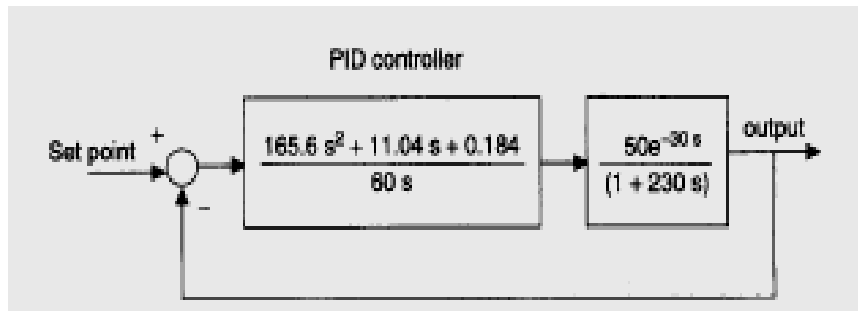
$$TD = 0,5 \times Td = 0,5 \times 30 = 15$$

PID transfer fonksiyonu bu durumda:

$$U(s) = K_p \cdot \left[1 + \frac{1}{T_i \cdot s} + TD \cdot s \right] \cdot E(s)$$

$$U(s)/E(s) = K_p \cdot \left[1 + \frac{1}{T_i \cdot s} + TD \cdot s \right] = 0,184 \times \left[1 + \frac{1}{60s} + 15s \right] = \frac{165,6s^2 + 11,04s + 0,184}{60s}$$

Sistemin kapalı-çevrim blok diyagramı aşağıdaki şekilde gösterilmiştir.

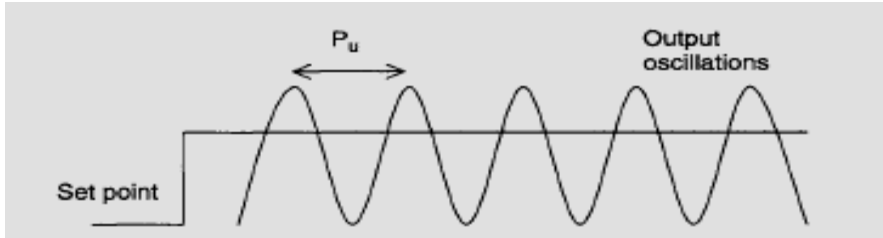


Şekil 4.8. Termal bir sistemin kapalı-çevrim blok diyagramı

4.5.2. Ziegler-Nichols Kapalı-Çevrim Tuning

Z-N tuning kapalı-çevrim testleri üzerine kuruludur. Yöntem şu şekildedir:

1. I, D kontrolör devre dışı bırakılarak sadece P kontrolör kullanılır.
2. Bir referans ayar noktası verilerek, cevap gözlemlenir.
3. Kararlı bir osilasyona sahip bir dalga elde edilinceye kadar kontrolör kazancı artırılır. Bu kazanca “ultimate kazanç” denir ve K_u olarak gösterilir..



Şekil 4.9. Ziegler-Nichols kapalı-çevrim testi

P_u : Kararlı osilasyon dalga periyodu

Tablo 4.1’ de gösterilen formüllere göre kontrolör parametreleri hesaplanır.

$$K_p=0,6xK_u, T_i=P_u/2, T_D=P_u/8 \quad [4.12]$$

4.6. Ayrık Zaman İntegral ve Türevin Elde Edilmesi

İntegral ifadesini yamuk toplamını cinsinden ifade edersek;

$$\frac{K_p}{T_i} \int_0^t e(t) dt = \frac{K_p}{T_i} \left[\frac{e(0) + e(T)}{2} T + \frac{e(T) + e(2T)}{2} T + \dots + \frac{e(k-1) + e(kT)}{2} T \right] \quad [4.13]$$

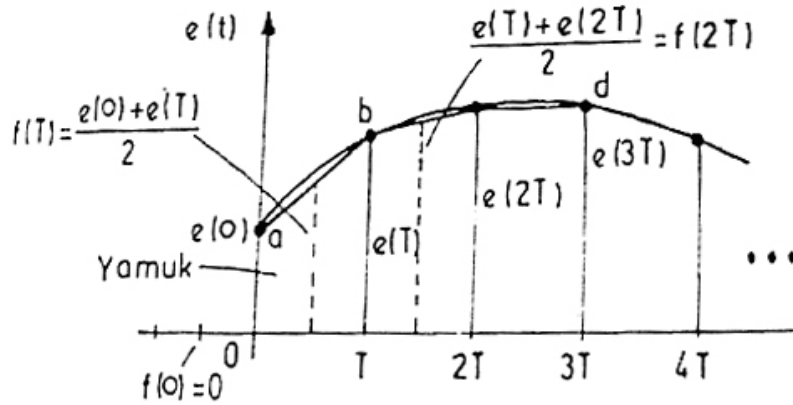
$$Z \left\{ \frac{Kp}{Ti} \int_0^t e(t) dt \right\} = \frac{Kp.T}{2Ti} \cdot \frac{z+1}{z-1} \quad [4.14]$$

Türev terimini de iki değer farkı cinsinden ifade edersek;

$$K_D \cdot \frac{de(t)}{dt} \Big|_{t=kT} \cong \frac{e(kT) - e[(k-1)T]}{T} K_D \quad [4.15]$$

$$Z \left\{ Kp \cdot \frac{de(t)}{dt} \right\} = \frac{E(z) - z^{-1}E(z)}{T} Kp \cong \frac{1-z^{-1}}{T} E(z) Td = \frac{z-1}{zT} KpTd \quad [4.16]$$

Şekil 4.10 'da sayısal olarak integral alma gösterilmiştir [3].



Şekil 4.10. Sayısal olarak integral alma yöntemi

Bu ifadelerden yola çıkarak transfer fonksiyonunu [4.17] denklemi ile gösterebiliriz.

$$G_{PID} = \frac{X(z)}{E(z)} = Kp + \frac{Kp.T}{2Ti} \cdot \frac{z+1}{z-1} + \frac{Kp.Td}{T} \cdot \frac{z-1}{z} \quad [4.17]$$

4.7. Ayırık Zaman PID Kontrolör Tasarımı

Ayrık zaman PID kontrolör denklemini aşağıdaki formülde gösterilmiştir:

$$U(z)=E(z).K_p.[1+ \frac{T}{T_i.(1-z^{-1})}+ TD. \frac{(1-z^{-1})}{T}] \quad [4.18]$$

$$a=K_p, \quad b=\frac{K_p.T}{T_i}, \quad c=\frac{K_p.TD}{T}$$

$$U(z)/E(z)=a+\frac{b}{(1-z^{-1})}+c.(1-z^{-1})$$

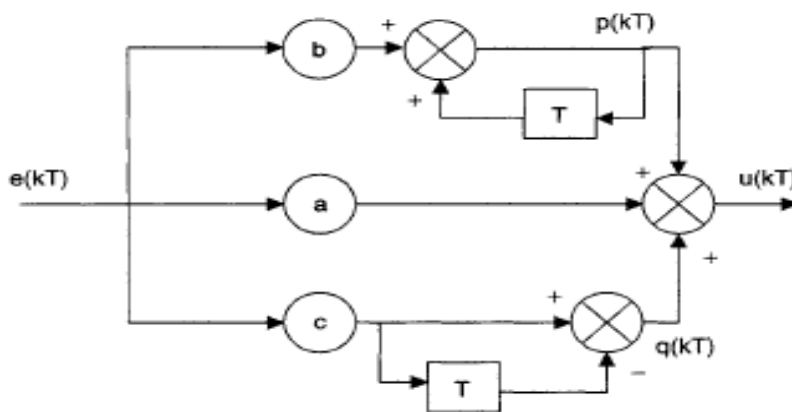
Şekil 4.11 diyagramı referans alınarak aşağıdaki denklemler yazılabilir. $p(kT)$ ve $q(kT)$ geçici değişkenlerdir.

$$w(kT) = a.e(kT) \quad \text{oransal kontrolör(P)}$$

$$p(kT) = b.e(kT)+ p(kT-T) \quad \text{integral kontrolör(I)}$$

$$q(kT) = c.(e(kT)- e(kT-T)) \quad \text{türev kontrolör(D)}$$

$$u(kT) = w(kT)+q(kT)+ p(kT) \quad \text{PID kontrolör çıkışı}$$



Şekil 4.11. PID kontrolör paralel gerçekleştirme diyagramı

Tüm bu bilgiler dahilinde standart bir PID algoritmasını sırasıyla aşağıdaki gibi yazabiliriz:

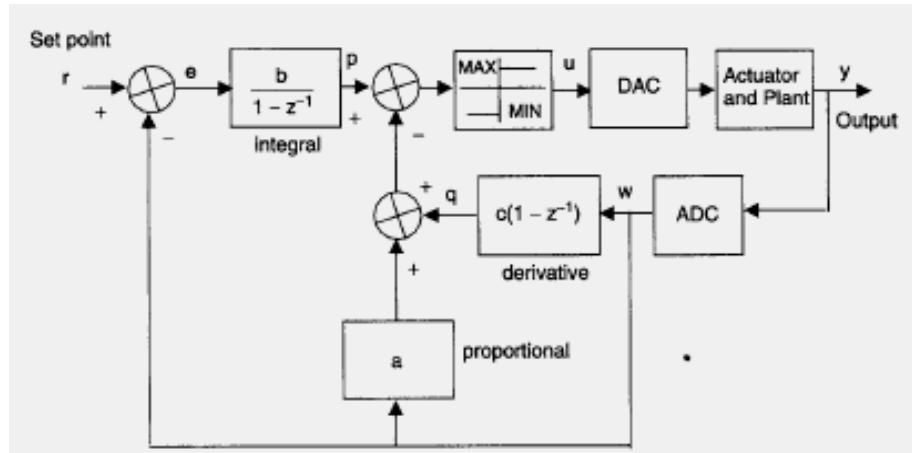
1. Ayar sıcaklığı: $r(kT)$
2. Sistem çıkışı: $y(kT)$
3. Hata: $e(kT) = r(kT) - y(kT)$
4. Oransal terim(P): $n(kT) = a \cdot e(kT)$
5. İntegral terimi(I): $p(kT) = b \cdot e(kT) + p(kT-T)$
6. Türev terimi(D): $q(kT) = c \cdot (e(kT) - e(kT-T))$
7. PID kontrolör çıkış: $u(kT) = n(kT) + p(kT) + q(kT)$
8. If ($u(kT) > \text{MAX}$)
 - { $p(kT) = p(kT-T)$;
 - $u(kT) = \text{MAX}$;
 - }
- else if ($u(kT) < \text{MIN}$)
 - { $p(kT) = p(kT-T)$;
 - $u(kT) = \text{MIN}$;
 - }
9. Kontrol sonucunu sürücüye gönder
10. Değişkenleri kaydet: $p(kT-T) = p(kT)$

$$e(kT-T) = e(kT)$$
11. Sonraki döngüyü bekle

Standart PID algoritmasında pratikte mevcut hatalarla karşılaşmaktadır. Bu hatalar:

1. Hata sinyali uzun zaman periyotlarında integral edildiği zaman, bir taşma durumu oluşabilir. Bundan sakınmak için kontrolör çıkışına maksimum ve minimum sınırlar konulmalıdır.
2. Referans sıcaklığı keskin olarak değiştirilince, hata sinyali de değişmektedir. Bu durumda türevsel kontrolör, kontrol çıkışını yükseltebilir. Bunu önlemenin yolu türevsel terimi geri besleme döngüsüne katmaktır.
3. Oransal terim de çıkışa ani bir yükseltme etkisi gösterebilir. Bu yüzden oransal terim de geri besleme döngüsüne katılmalıdır.

Tüm bu bilgiler doğrultusunda pratikteki bir PID kontrolörü Şekil 4.12'teki gibi oluşturabiliriz:



Şekil 4.12. Modifiye edilmiş bir PID kontrolör gerçekleştirme diyagramı

Pratik olarak gerçekleştirilecek bir PID algoritmasını tekrar yazarsak:

1. Ayar sıcaklığı: $r(kT)$
2. Sistem çıkışı: $w(kT)$
3. Hata: $e(kT) = r(kT) - w(kT)$
4. Oransal terim(P): $n(kT) = a \cdot w(kT)$

5. İntegral terimi(I): $p(kT)=b.e(kT)+ p(kT-T)$
6. Türev terimi(D): $q(kT)=c.(w(kT)- w(kT-T))$
7. PID kontrolör çıkışı: $u(kT)= n(kT)+p(kT)+ q(kT)$
8. If ($u(kT)>MAX$)
 - { $p(kT)= p(kT-T)+MAX- u(kT);$
 - $u(kT)=MAX;$
 - }
- else if ($u(kT)<MIN$)
 - { $p(kT)= p(kT-T)+MIN- u(kT);$
 - $u(kT)=MIN;$
 - }
9. Kontrol sonucunu sürücüye gönder
10. Değişkenleri kaydet: $p(kT-T)= p(kT)$

$$w(kT-T)= w(kT)$$
11. Sonraki döngüyü bekle

4.8. PID Kontrolör Örnekleme Zamanını Belirleme

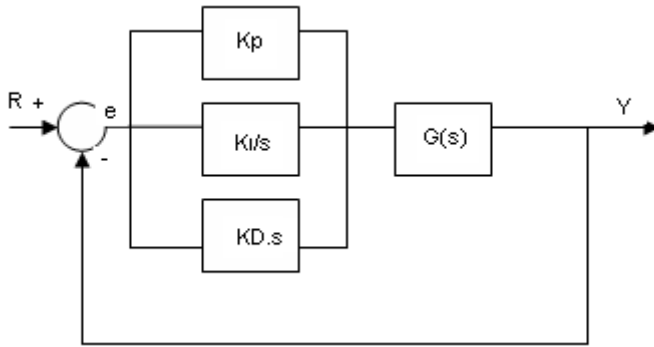
Ziegler-Nichols'a göre PID kontrol örnekleme zamanı, sistemin açık-çevrim fonksiyonunda elde edilen T_1 parametre değerinin dörtte birinden küçük olmalıdır.

$$T_{\text{örnekleme_zamanı}} < T_1/4 \text{ olmalıdır.}$$

4.9. Standart ve Modifiye Edilmiş PID Kontrolörlerin Karşılaştırılması

Standart bir PID kontrolör ile pratikte oluşabilecek hataları göz önünde bulundurarak oluşturulmuş modifiye edilmiş PID kontrolör arasındaki farkı irdelememizde yarar vardır.

4.9.1. Standart PID kontrolör tasarımı



Şekil 4.13. Standart PID kontrolör tasarımı

Şekil 4.13'ten aşağıdaki denklemleri elde edebiliriz:

$$Y = G(s) \cdot (K_p + K_i/s + K_D \cdot s) \cdot e$$

$$e = R - Y$$

Bu iki denklemden;

$$Y = G(s) \cdot (K_p + K_i/s + K_D \cdot s) \cdot (R - Y)$$

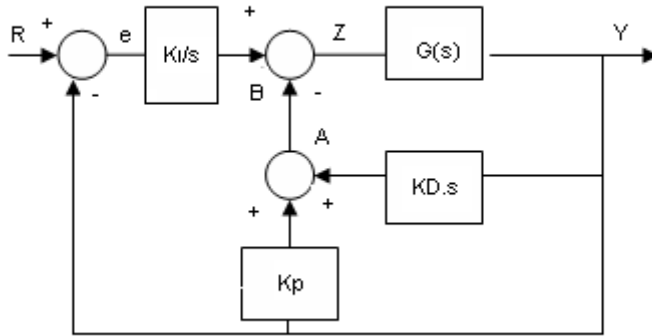
$$Y \cdot (1 + G(s) \cdot (K_p + K_i/s + K_D \cdot s)) = G(s) \cdot (K_p + K_i/s + K_D \cdot s) \cdot R$$

$$Y \cdot (s + G(s) \cdot (K_p \cdot s + K_i + K_D \cdot s^2)) = G(s) \cdot (K_p \cdot s + K_i + K_D \cdot s^2) \cdot R$$

$(K_p \cdot s + K_i + K_D \cdot s^2)$ ifadesini $(s+a) \cdot (s+b)$ şeklinde yazarsak [4.19] denklemini elde ederiz;

$$\frac{Y}{R} = \frac{G(s).(s+a).(s+b)}{s+G(s).(s+a).(s+b)} \quad [4.19]$$

4.9.2. Modifiye edilmiş PID kontrolör ile tasarım



Şekil 4.14. Modifiye edilmiş PID kontrolör tasarımı

Şekil 4.14'ten aşağıdaki denklemleri elde edebiliriz:

$$e=R-Y$$

$$B=(Ki/s).e=Ki/s.(R-Y)$$

$$A=(Kp+KD.s).Y$$

$$Z=B-A= Ki/s.(R-Y)- (Kp+KD.s).Y$$

$$Y = Z.G(s) = (Ki/s.(R-Y)- (Kp+KD.s).Y).G(s)$$

$$Y+Y. ((Ki/s)+ Kp+KD.s)G(s) = (Ki/s).R.G(s)$$

$$Y/s.(s+(Ki+Kp.s+KD. s^2).G(s)) = (Ki/s).R.G(s)$$

Tüm denklemleri bir araya getirdiğimizde $(Kp.s+Ki+KD. s^2)$ ifadesini $(s+a).(s+b)$ şeklinde yazarsak [4.20] denklemini elde ederiz;

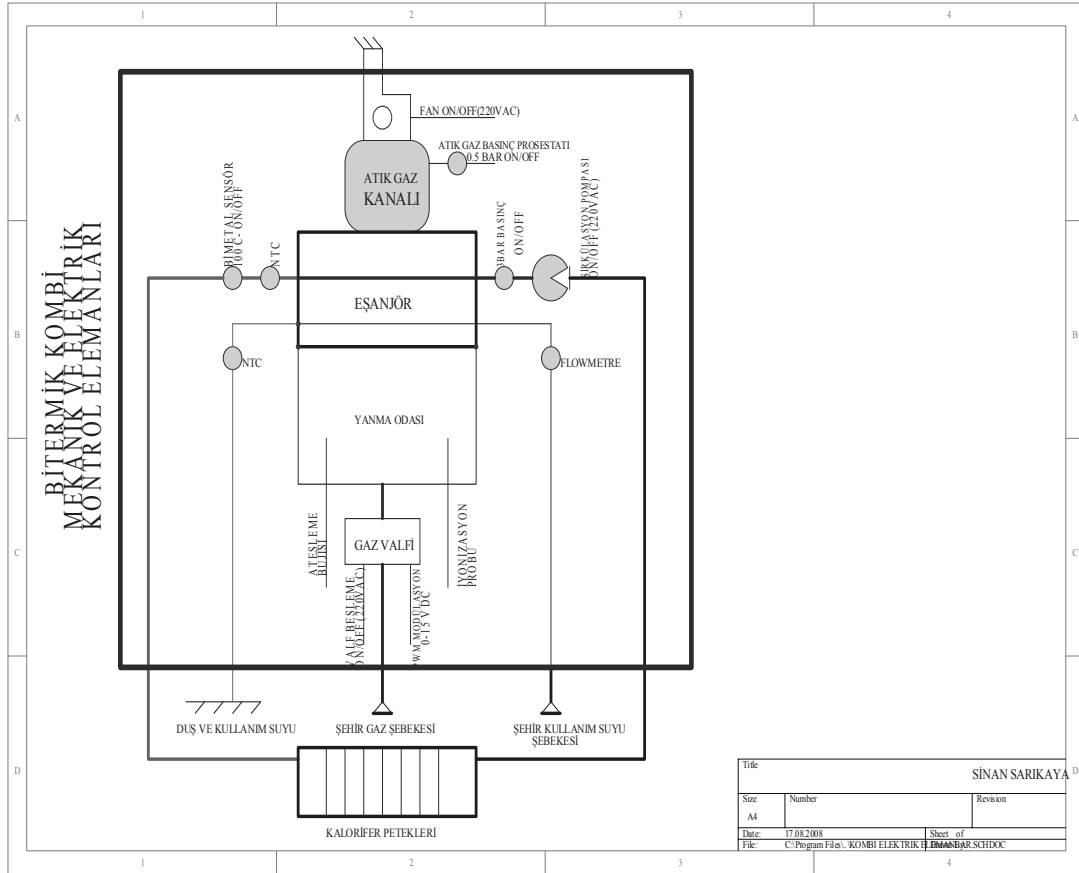
$$\frac{Y}{R} = \frac{G(s).Ki}{s+G(s).(s+a).(s+b)} \quad [4.20]$$

[4.19] standart PID durum denklemindeki $(s+a).(s+b)$ çarpanları, $s=-a$ ve $s=-b$ değerlerinde, kapalı-çevrim transfer fonksiyonunda sıfır getirmektedir. Bu sıfırlar sistem çıkış cevabında aşımın artmasına neden olmaktadır. Bu istenmeyen bir durumdur. [4.20] modifiye edilmiş PID kapalı-çevrim transfer fonksiyonunda ise aşımın sebep olan bu sıfırlar gelmez.

BÖLÜM 5. BİTERMİK KOMBİ ISITMA SİSTEMİ ve SİSTEM DEVRE ELEMANLARI

5.1. Kombi Isıtma Sistemi Devre Şeması ve Devre Elemanları

Kombi ısıtma sistemi, çok sayıda mekanik, elektronik ve elektromekanik elemanların bir araya getirilmesiyle oluşturulur. Evlerimizde veya işyerlerimizde ısınma amaçlı kullanmakta olduğumuz bir kombinin iç devre şeması ve sistemi oluşturan ana devre elemanları Şekil 5.1’de gösterilmiştir.



Şekil 5.1. Bitermik kombi sistem şeması ve devre elemanları

Şekil 5.1’de kombi ısıtma sistemini oluşturan devre kontrol elemanları gösterilmiştir. Her bir devre elemanı, elektronik kart ile kontrol edilir. Elektrik sinyalleri ile kontrol edilen devre elemanları aşağıda sıralanmıştır:

1. Kapalı devre su sirkülasyon pompası besleme gerilimi,(220V AC)
2. Kapalı devre su akışını algılayan basınç prosestatı-(on/off switch),
3. Kapalı devre 100 °C bimetal limit sensör-(on/off switch),
4. Kapalı devre su sıcaklığını ölçen 10k NTC sensör,
5. Kullanım suyu(açık devre) debisini ölçen flowmetre, (0-5V DC PULSE)
6. Kullanım suyu sıcaklığını ölçen 10k NTC sensör,
7. Yanma odasında biriken gazı sistemden uzaklaştıran fana ait besleme gerilimi,(220 V AC)
8. Fanın devreye girişini algılayan 0.5 bar atık gaz basınç prosestatı-(on/off switch)
9. Gaz valfi besleme gerilimi,(220V AC)
10. Gaz valfi modülasyon bobini kontrol sinyali,(0-15 VDC)
11. Gaz ateşleme elektrodu(buji) ,(20 KV)
12. Ateşlemeyi algılayan alev iyonizasyon algılama elektrodu,
13. Bu devre elemanlarına ek olarak, gelişmiş monotermik tip kombilerde, su geçişini eşanjördeki kapalı veya açık devreye ileten üç yollu vana vardır.

5.2. Kombi Sistem Elemanları ve Görevleri

Bu bölümde bir kombi sistemini oluşturan ana devre elemanlarının görevleri açıklanmıştır. Şekil 5.2’de günümüzdeki üretilen farklı tiplerdeki kombiler gösterilmiştir.



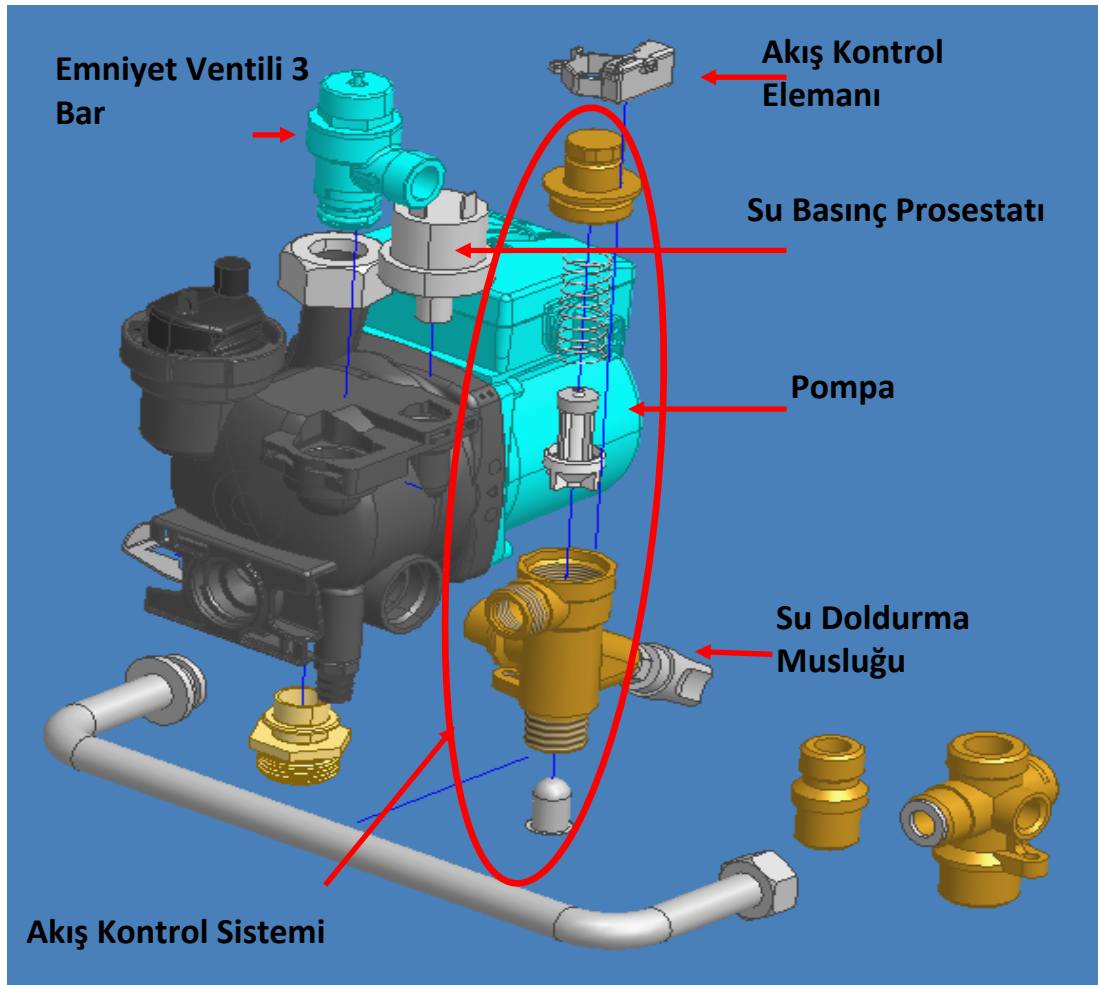
Şekil 5.2. Günümüzde üretilen farklı kombi tipleri



Şekil 5.3. Bir kombinin iç görünüşü

5.2.1. Su sirkülasyon pompası

Su sirkülasyon pompası kapalı çevrimdeki suyu sirküle etmeye yarar. Böylece kombi tarafından ısıtılan sıcak su, ortamda bulunan tüm kalorifer sistemine ulaştırılmış olur.



Şekil 5.4. Kapalı çevrim su sirkülasyon pompası

5.2.2. Su basınç prosestatı

Su basınç prosestatı, kapalı devre su sirkülasyon pompası devreye girdiğinde kalorifer sistemindeki suyun uygun basınç seviyesine ulaşp ulaşmadığını kontrol eden mekanik bir anahtardır. On/Off kontak prensibine göre çalışır. Su basınç prosestatı Şekil 5.4'te gösterilmiştir.

5.2.3. Limit sıcaklık sensörü

Limit sıcaklık sensörü, kapalı devredeki su sıcaklığı 100 C°'nin üzerine çıktığında genişerek birbirinden ayrılan metallere oluşan bir devre elemanıdır. On/Off kontak prensibine göre çalışır. Şekil 5.5'te gösterilmiştir.



Şekil 5.5. Limit sıcaklık sensörü

5.2.4. NTC sıcaklık sensörü

NTC sensör, kapalı ve açık çevrim devresindeki su sıcaklık değerini algılayan devre elemanıdır. Genelde 10 kOhm'luk ntc sensör kullanılmaktadır. Şekil 5.6'da NTC sensör ve Şekil 5.7'de ntc sensör sıcaklık-direnç tablosu gösterilmiştir [6].



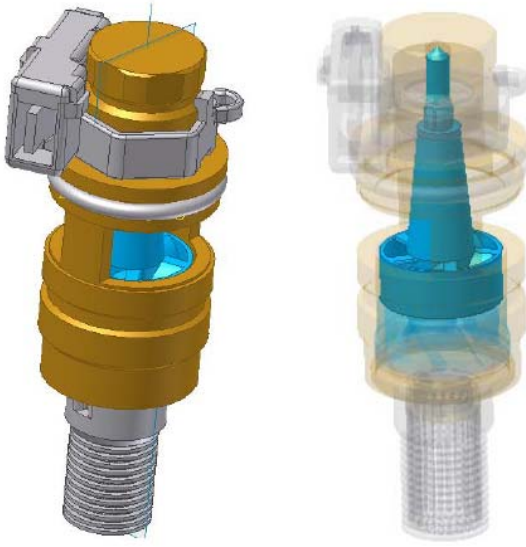
Şekil 5.6. NTC sıcaklık sensörü

Steinhard & Hart Coeff.	Curves							
	2.7k		10k-A		10k-B		12k-A	
C	1.94567E-18		8.77547E-08		1.89916E-07		1.28318E-07	
B	2.51842E-04		2.34108E-04		2.52293E-04		2.40423E-04	
A	1.36421E-03		1.12924E-03		8.84204E-04		9.89458E-04	
f	1		1.01707E+00		1		1	
T [°C]	R [Ω]	α [1/°C]	R [Ω]	α [1/°C]	R [Ω]	α [1/°C]	R [Ω]	α [1/°C]
-20	29.30k	-6.2%	98.66k	-5.8%	67.38k	-4.8%	98.97k	-5.4%
-10	16.14k	-5.7%	56.25k	-5.4%	42.14k	-4.6%	58.88k	-5.0%
0	9.292k	-5.3%	33.21k	-5.1%	27.06k	-4.3%	36.13k	-4.7%
10	5.561k	-5.0%	20.24k	-4.8%	17.82k	-4.1%	22.80k	-4.5%
20	3.446k	-4.6%	12.71k	-4.5%	12.00k	-3.8%	14.77k	-4.2%
25	2.746k	-4.5%	10.17k	-4.4%	9.92k	-3.7%	12.00k	-4.1%
30	2.205k	-4.3%	8.194k	-4.3%	8.251k	-3.6%	9.804k	-4.0%
40	1.451k	-4.0%	5.416k	-4.0%	5.786k	-3.5%	6.652k	-3.8%
50	980.1	-3.8%	3.663k	-3.8%	4.132k	-3.3%	4.607k	-3.6%
60	677.8	-3.6%	2.530k	-3.6%	3.000k	-3.1%	3.252k	-3.4%
70	478.9	-3.4%	1.782k	-3.4%	2.213k	-3.0%	2.337k	-3.2%
80	345.1	-3.2%	1.278k	-3.2%	1.656k	-2.8%	1.707k	-3.1%
85	295.0	-3.1%	1.089k	-3.2%	1.441k	-2.8%	1.467k	-3.0%
90	253.2	-3.0%	931.6	-3.1%	1256.9	-2.7%	1.266k	-2.9%
100	188.9	-2.9%	690.0	-2.9%	965.9	-2.6%	952.3	-2.8%
110	143.1	-2.7%	518.5	-2.8%	751.2	-2.5%	726.0	-2.6%
120	109.9	-2.6%	395.0	-2.7%	590.8	-2.3%	560.4	-2.5%
125	96.83	-2.5%	346.4	-2.6%	526.0	-2.3%	494.6	-2.5%

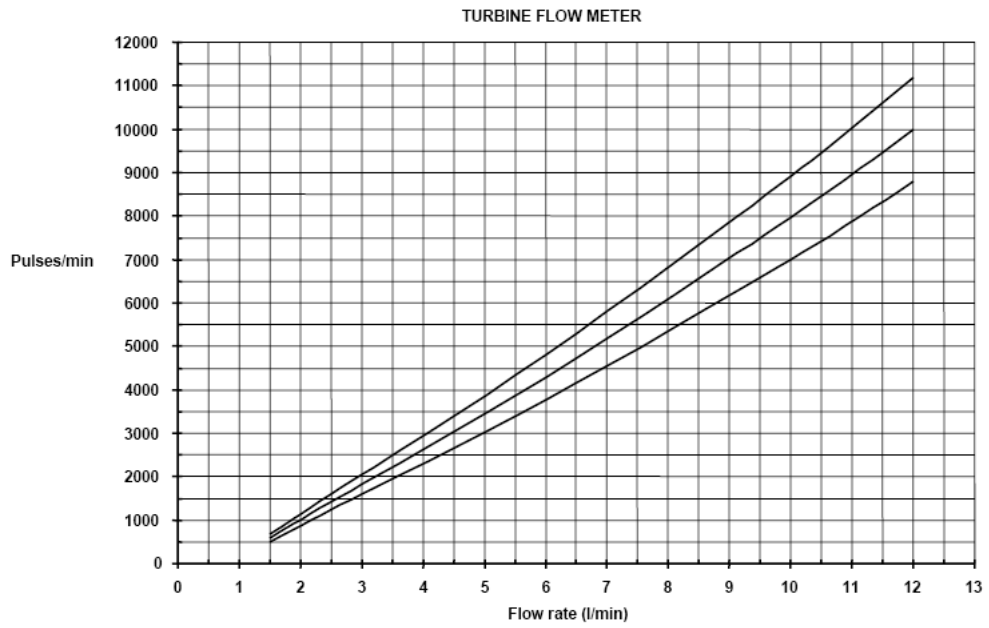
Şekil 5.7. Honeywell T7335 NTC termistör sıcaklık-direnç tablosu

5.2.5. Akışölçer

Akışölçer, kullanım suyu debisini ölçen bir devre elemanıdır. Piyasada farklı prensipte çalışan birçok su akış debisi ölçer vardır. Kullanım suyu akış kanallına bağlanır ve su akış hızına göre DC 5V pulse üretir. Pulse sayısına göre dakikada geçen su miktarı litre/dakika olarak ölçülür. Geçen su debisine göre pulse sayısını gösteren tablo, üreticinin vermiş olduğu teknik katalogda gösterilir. Şekil 5.8'de projemizde kullanılan flowmetre ve şekil 5.9'da flowmetre DC pulse/akış miktarı tablosu gösterilmiştir [9].



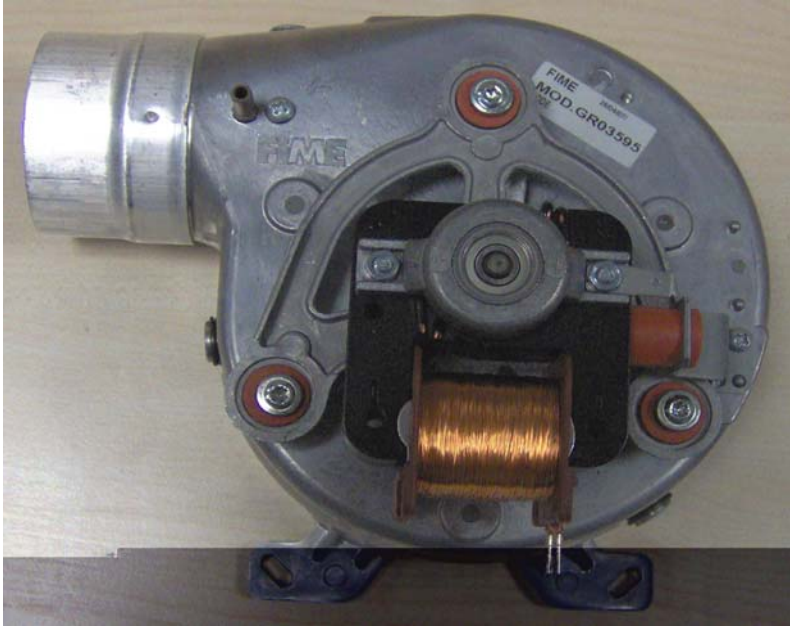
Şekil 5.8. FUGAS-TURBINA-SPT102 flowmetre



Şekil 5.9. FUGAS-TURBINA-SPT102 flowmetre su akış miktarı-pulse ilişkisi

5.2.6. Fan

Fan, kombi yanma odasındaki yanmış gazı sistemden uzaklaştıran devre elemanıdır. 220V AC besleme gerilimi ile açılıp kapatılır. Şekil 5.10'da atık gaz fanı gösterilmiştir.



Şekil 5.10. FIME atık gaz fanı

5.2.7. Atık gaz basınç prosestati

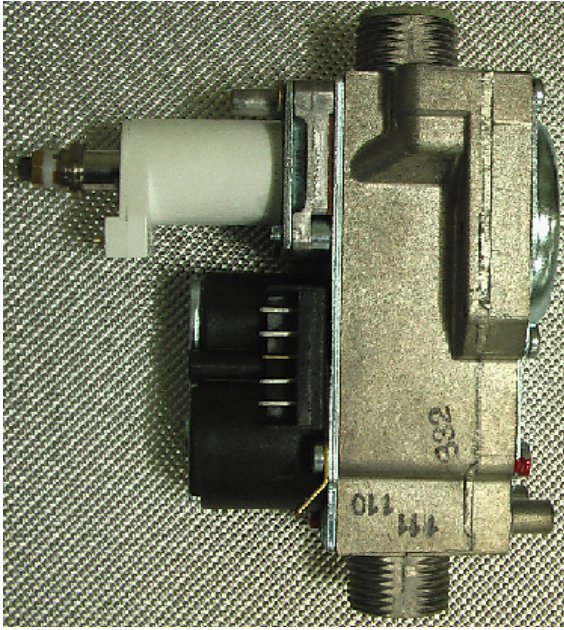
Yanma odasında fan aracılığıyla uzaklaştırılan gazın sistemden atılıp atılmadığını algılayan devre elemanıdır. Sistemden uzaklaştırılan gazın oluşturduğu basınca göre kontaklar açılır veya kapanır. On/off kontak prensibine göre çalışan mekanik bir anahtardır. Şekil 5.11’de atık gaz basınç prosestati gösterilmiştir.



Şekil 5.11. HUBA 80/60 atık gaz basınç prosestati

5.2.8. Gaz valfi ve gaz valfi modülasyon bobini

Gaz valfi, kombi girişine bağlanan gazı kontrol eden en önemli devre elemanıdır. Gaz valfi vasıtasıyla yanma odasına gaz gönderilir. Gaz valfi kapısı 220 VAC gerilim ile açılır ve 0-15 VDC gerilim ile modülasyon bobini kontrol edilerek gaz debisi değiştirilir. Modülasyon bobini gelişmiş sistemlerde genelde PID kontrolörün ürettiği kontrol işareti ile sürülür. Gaz valfi modülasyon bobininin kontrol gerilimi üreticiye göre değişebilir. Şekil 5.12’de bir gaz valfi gösterilmiştir.



Şekil 5.12. HONEYWELL Moduplus VK 4105 gaz valfi

5.2.9. Ateşleme elektrodu

Ateşleme elektrodu, elektronik ateşleme yapmayı sağlar. Ateşleme elektrodu yaklaşık 20kV’luk yüksek çıkış gerilimine sahip trafonun sekonder uçlarından birine bağlanarak gaz odasına taşınır ve yanma odasının şaseyiyle birleşip, yüksek gerilim yoluyla ark yaparak ateşleme yapar. Şekil 5.13’te dış yapısı seramikle izole edilmiş bir ateşleme elektrodu gösterilmiştir.



Şekil 5.13. Ateşleme/iyonizasyon elektrodu

5.2.10. İyonizasyon elektrodu

Yapı olarak ateşleme elektroduyla aynıdır. Yanma odasında ateşlemenin oluşup oluşmadığını algılayan devre elemanıdır. Ateşleme oluştuğunda, yanma odasındaki gazın yanması sonucu iyonlar yoluyla bir akım oluşur ve bu akım iyonizasyon elektrodu vasıtasıyla elektronik devreye taşınır. Oluşan iyonizasyon akımı yaklaşık 300uA civarındadır.

5.2.11. Eşanjör

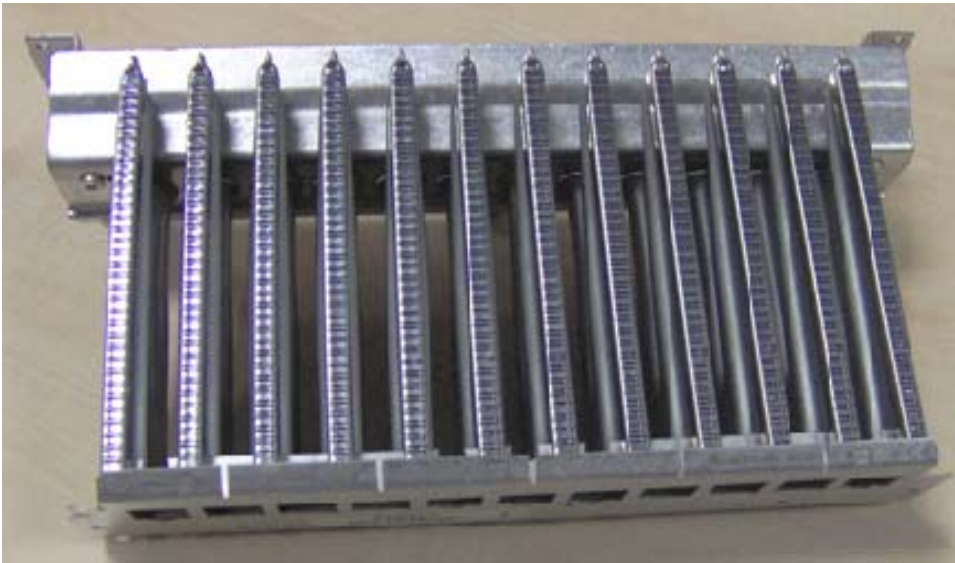
Eşanjör, ısı iletimi yüksek çok sayıda ince metal kanatçığın bir araya getirilmesiyle oluşan mekanik bir devre elemanıdır. Yanma odasındaki gazın yanmasıyla oluşan alev, geniş metal yüzeyi ısıtıp eşanjör içerisinden geçen kullanım suyunun ve kapalı devre suyunun ısıtılmasını sağlar. Kapalı ve açık devre su geçişi, eşanjör içerisinde, birbirinden bağımsız iki kanal vasıtasıyla sağlanır. Şekil 5.14'te bir eşanjör gösterilmiştir.



Şekil 5.14. Eşanjör

5.2.12. Brülör

Gaz valfinden geçen gaz, çok sayıda kanalın bir araya getirilmesi ile oluşturulan gözlerle ulaşır. Bu gözlerle oluşturulan mekanik yapıya brülör denir. Bu devre elemanı vasıtasıyla ateşleme yüzeyi artırılır ve daha yüksek bir ateşleme yüzeyi oluşturulur. Brülör yanma odasında bulunur. Şekil 5.15’te bir brülör gösterilmiştir.



Şekil 5.15. Tipik bir brülör

BÖLÜM 6. KOMBİ ISITMA SİSTEMİ ELEKTRONİK KONTROL KARTI TASARIMI

6.1. Elektronik Kontrol Kartı Blok Diyagramı

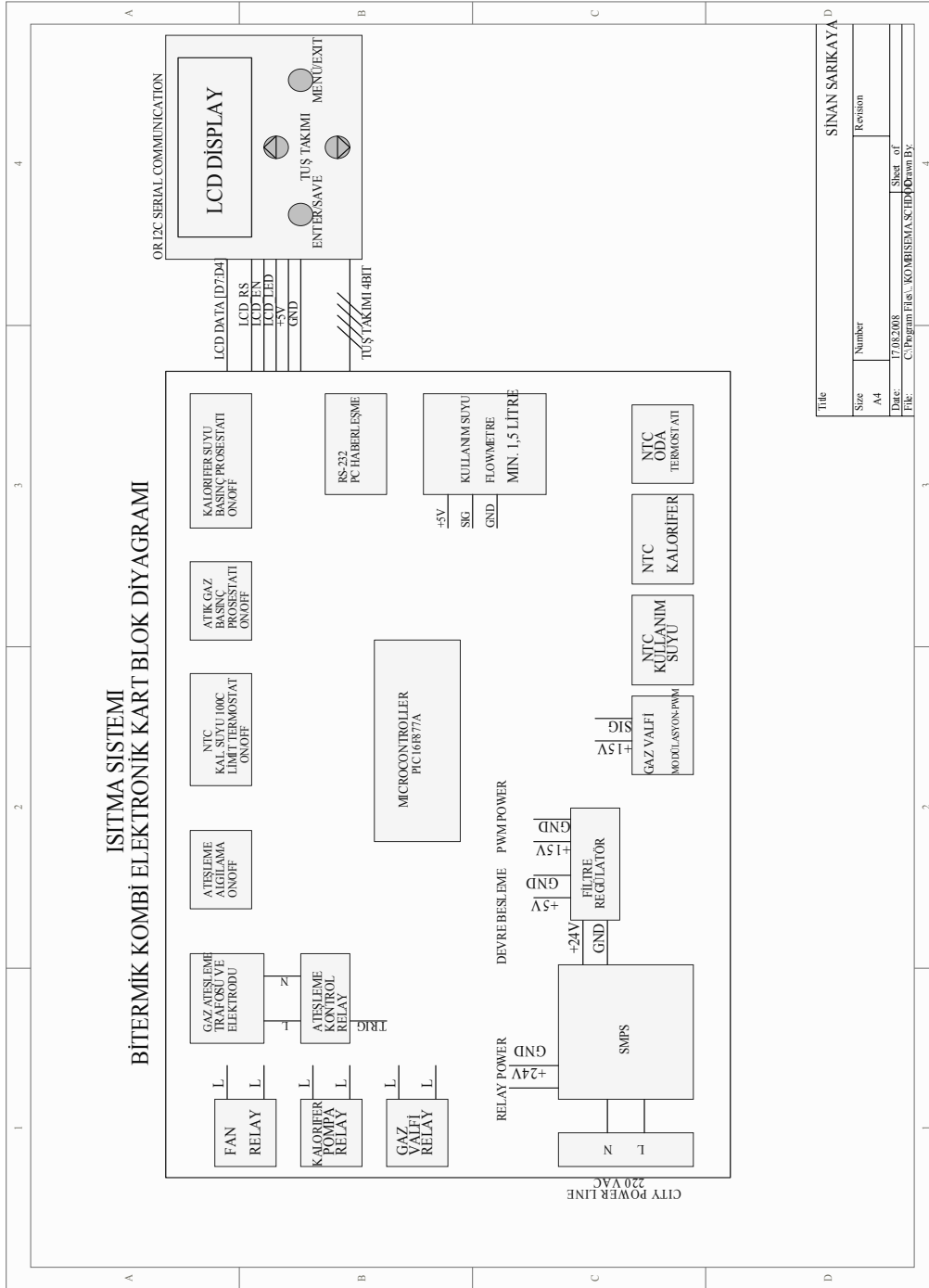
Elektronik kart tasarımına başlanmadan önce elektronik kontrol kartı blok diyagramını hazırlamak, proje oluşturmanın birinci ve temel ayağıdır. Bu plan, tasarımın mimarisini belirlemeye yarar ve sistemi parçalara ayırarak proje basamaklarını belirlememizi ve projeye hakim olmamızı sağlar. Aynı zamanda blok diyagramı, bir projenin rehberidir.

Blok diyagramını belirlemeden önce sistem kontrol elemanlarının özelliklerini ve elektrik-elektronik kontrol yöntemlerini bölüm 5'te ayrıntılı olarak işledik. Şekil 6.1'de sistemin blok diyagramı gösterilmiştir.

Isıtma sistemi elektronik kartı blok diyagramını oluşturan özellikler aşağıda sıralanmıştır:

1. Elektronik kontrol kartı dc güç kaynağı ve filtre tasarımı,
2. Mikrodenetleyici ve çevre bileşenleri devresi,
3. NTC termistör kontrol devresi,
4. Mekanik sensörlerin on/off kontak durumlarını algılama devresi,
5. Gaz valfi ve modülasyon bobini kontrol devresi,
6. Sistemi dışarıdan izleme ve kontrol etme amaçlı kontrol kartı-PC RS-232 haberleşme arayüz devresi,
7. 220VAC beslemeli ekipmanların röle kontrol devresi,
8. Açık çevrim su debisini ölçen flowmetre sinyal işleme devresi,
9. Gaz ateşleme trafosu kontrol devresi, (projede gaz ateşleme trafosu kontrol devresi tasarlanmayıp, piyasada bulunan harici ateşleme kartı kullanılmıştır.)

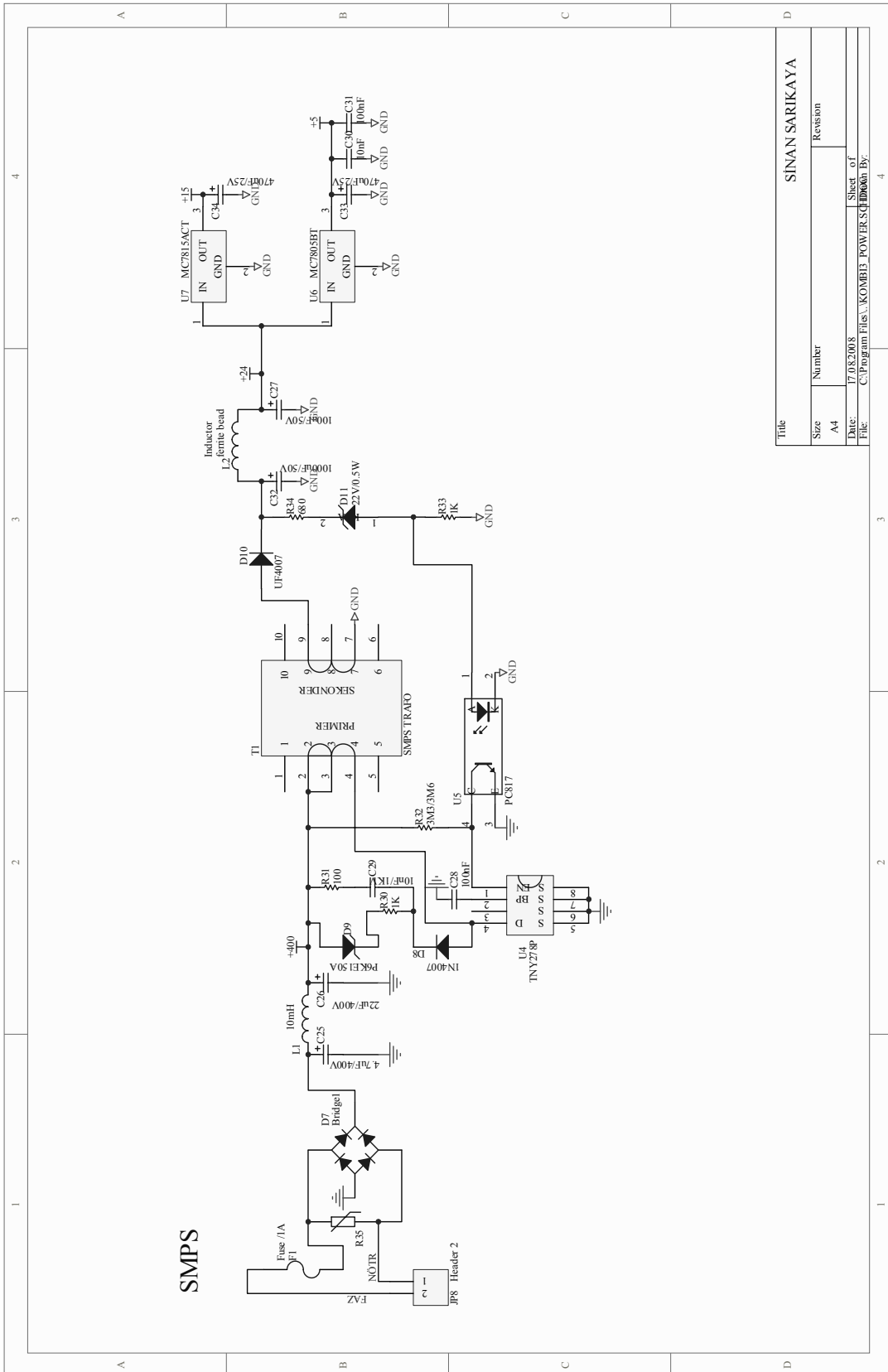
10. Ateşleme algılama devresi, (bu özellik için kontrol devresi tasarlanmamıştır.)
11. Kullanıcının arayüz bilgilerini girmesini ve izlemesini sağlayan tuş takımı ve LCD display kontrol devresi. Projede tuş takımı ve LCD display kontrol devresi ana kontrol kartından bağımsız, ikinci bir kart olarak tasarlanmıştır.



Şekil 6.1. Kombi ısıtma sistemi elektronik kontrol kartı blok diyagramı

6.1.1. Elektronik kontrol kartı DC güç kaynağı ve filtre tasarımı

DC güç kaynağı tasarımı ana kart tasarımının en önemli kısmını teşkil etmiştir. Tasarımda güç kaynağı seçiminde; elektronik kontrol kartının hacmini ve ağırlığını azaltan, elektrik koruma fonksiyonları olarak kısa devre korumalı, aşırı akım-sıcaklık sınırlayıcı, sekonder devre gerilimini sabitleyici koruma devrelerine sahip SMPS (switch mode power supply- anahtarlama mod güç kaynağı) güç kaynağı tasarlanmıştır. Tüm devre korumaları göz önünde bulundurularak tasarlanan sistem güç kaynağı devresi Şekil 6.2’de gösterilmiştir.

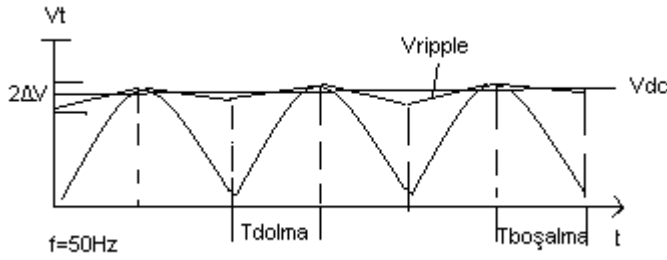


Title		SINAN SARIKAYA	
Size	Number	Revision	
A4			
Date:	17.08.2008	Sheet of	
File:	C:\Program Files\KOMBE\POWERS\170808	Sheet of	
		Drawn By:	

Şekil 6.2. Sistem güç kaynağı devresi

Güç kaynağı tasarımında birincil giriş koruma devresi olarak; şebeke gerilimindeki geçici yüksek frekanstaki ve genlikteki EFT(electrical fast transient) sinyallerini TNY278P entegresine ulaşmasını engellemek amacıyla 275V AC eşik gerilimine sahip MOV(metal oxid varistor) kullanılmıştır. R35 MOV devre elemanını sırt sırta ters bağlanmış yüksek güce sahip zener diyotlar gibi düşünebiliriz. Bu sayede eşik gerilimini aşan yüksek gerilimler kırılarak TNY278PN entegresinden uzaklaştırılacaktır. Ayrıca TNY278PN entegresi mosfet akım tüketimi maksimum 0,7A ile sınırlandırıldığından ve diğer devre elemanları akım sarfiyatından dolayı, 1A'lık sigorta kısa devre önleyici olarak devreye eklenmiştir [7].

AC giriş gerilimi, D7 köprü diyot aracılığıyla doğrultulmuştur. Doğrultma devresinden sonra smps trafonun primer kısmından ortalama maksimum 0,1 A sürekli akım gereksiniminden dolayı ve dalgalı dc gerilimin düzgün hale getirilmesi için C1, C2 kondansatörleri ve L1 bobini kullanılmıştır. C1,C2 ve L1 devre elemanlarının elektriksel değerlerinin hesaplanması aşağıda açıklanmıştır. Hesaplamalarda yaklaşık değerler göz önüne alınmıştır.



Şekil 6.3. Dalgalı DC gerilimin düzgün hale getirilmesi

Vdc: Dc gerilim değeri

Vd: Köprü diyot devresindeki diyot gerilim düşümü

ΔV: Ripple gerilim

ΔQc: Kondansatör üzerinde biriken yük

Δt : Yükün boşaldığı süre

ΔQ_{dc} : Birim zamanda harcanan yük

T: AC giriş gerilimi periyodu

$$V_{dc} = V_m - 2V_d - \Delta V \quad [6.1]$$

Kondansatör üzerinde biriken yük = birim zamanda harcanan yük

$$\Delta Q_c = \Delta Q_{dc} \quad [6.2]$$

$$C = \frac{\Delta Q_c}{\Delta V} \text{ ve } I_{dc} = \frac{\Delta Q_{dc}}{\Delta t} = \frac{V_{dc}}{R} \quad [6.3]$$

$$C \cdot 2\Delta V = I_{dc} \cdot \Delta t \quad [6.4]$$

Yükün boşaldığı periyot = AC gerilim periyodu/2

$$\Delta t = T/2 = \frac{1}{2f} \quad [6.5]$$

Tüm denklemleri bir araya getirdiğimizde;

$$C \cdot 2 \cdot \Delta V = \frac{V_{dc}}{R} \cdot \frac{1}{2f}$$

$$\Delta V = \frac{1}{4fRC} \cdot V_{dc} \quad \text{bulunur.} \quad [6.6]$$

DC gerilim yüzde dalgalılık oranı:

$$\delta = \frac{\Delta V}{V_{dc}} \cdot 100 = \frac{1}{4fRC} \cdot 100 \quad [6.7]$$

Sonuç olarak:

$V_{dc} = V_m - 2V_d - \Delta V$ denklemindeki ΔV eşitliği yerine [6.6] denklemini koyduğumuzda;

$$V_{dc} = (V_m - 2V_d) \cdot \frac{4fRC}{4fRC + 1} \quad \text{bulunur.} \quad [6.8]$$

SMPS trafounun primer tarafında harcanan güç, sekonder tarafta harcanan güce eşit olduğundan;

$$P_p = P_s \quad [6.9]$$

$$I_p \cdot V_p = I_s \cdot V_s \quad [6.10]$$

$$I_p = I_s \cdot \frac{V_s}{V_p}$$

I_p : SMPS trafo primer devre akımı

V_p : SMPS trafo primer devre gerilimi

I_s : SMPS trafo sekonder devre akımı

V_s : SMPS trafo sekonder devre gerilimi

Sistem tasarımında ihtiyaç duyulan gereksinimler:

Sekonder devre akımı ihtiyacı: maksimum 1 A

Primer devre gerilimi: 220 VAC

Sekonder devre gerilimi: 17 VAC

DC giriş gerilimi yüzde dalgalılık oranı: 10

$$I_p = I_s \cdot \frac{V_s}{V_p} = 1 \cdot \frac{17}{220} = 0,077 \text{ A}$$

Akım ihtiyacının %40 fazlasını emniyet için alırsak;

$$I_p = \sim 0,1 \text{ A}$$

$$\text{Primer yük direnci } R_p = \sim \frac{V_m}{I_p} = 310/0,1 = 3100 \Omega$$

DC gerilim yüzde dalgalılık oranı çok küçük değer seçildiği zaman kondansatör değeri artar ve haliyle kondansatör boyutunda hacim olarak bir büyüme olacaktır. Bu da istenmeyen bir durum olduğundan projeye uygun %10 değer seçilmiştir.

DC gerilim yüzde dalgalılık oranı:

$$\delta = \frac{\Delta V}{V_{dc}} \cdot 100 = \frac{1}{4fRC} \cdot 100$$

$\delta = 10$ istendiğinden;

$$\frac{1}{4fRC} \cdot 100 = 10$$

$$C = \frac{2,5}{fR}$$

$$R = R_p = 3100 \Omega$$

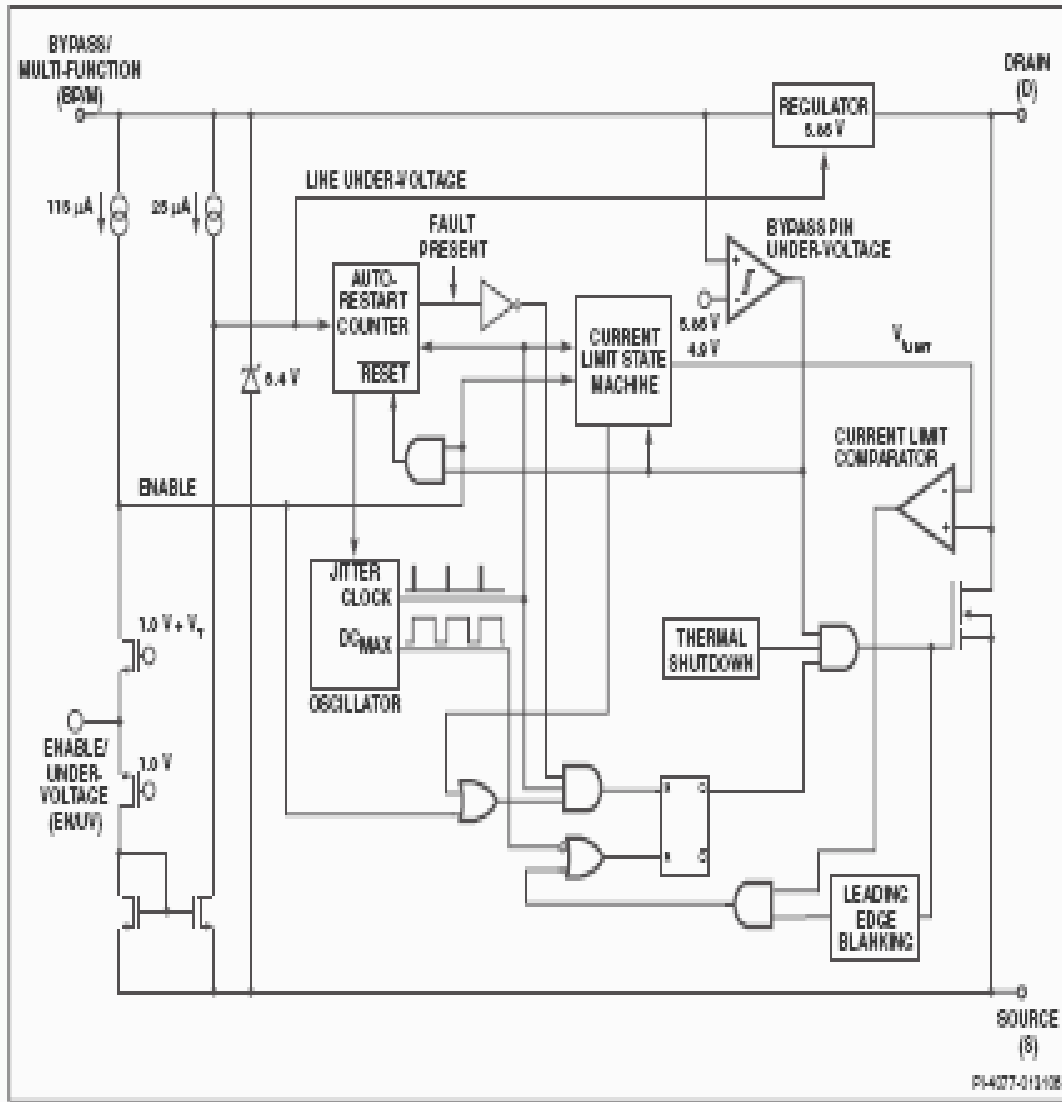
$$f = 50\text{Hz}$$

$$C = \frac{2,5}{fR} = \frac{2,5}{3100 \cdot 50} = 16\mu\text{F}$$

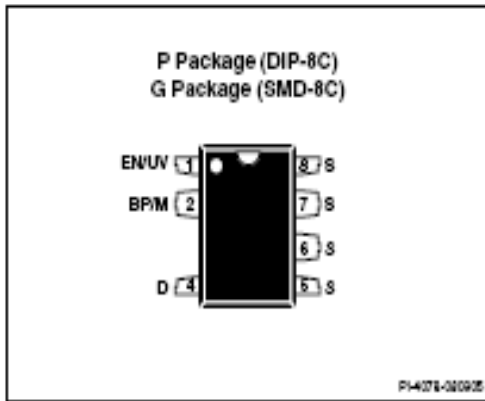
$C=16 \mu\text{F}$ değerine en yakın kondansatör için $22\mu\text{F}-400\text{VDC}$ elektrolitik kondansatör seçilmiştir. Devrede $C1+C2=22\mu\text{F} + 4,7\mu\text{F} = 26,7 \mu\text{F}$ kondansatörleri kullanılmıştır. $L=10\text{mH}$ bobini hem ripple gerilimi azaltmak hemde yüksek frekanslı bileşenlerin etkisini azaltmak için devreye eklenmiştir.

SMPS kontrolör olarak Powerint firması tarafından üretilen TNY278PN smps kontrol entegresi kullanılmıştır. TNY278PN entegre seçiminde, minimum sayıda harici devre elemanlarının kullanımı ön plana çıkmıştır. TNY278PN blok diyagramı Şekil 6.3'de ve entegre pin konfigürasyonu Şekil 6.4'te gösterilmiştir.

TNY278PN entegresindeki mosfet kapalı konuma geçtiğinde, smps trafonun primer uçlarında depolanan akım, bir akım kaynağı gibi davranacağından mosfetin DS uçlarında çok yüksek gerilim oluşturacaktır. Yüksek gerilim olasılığını engellemek için D8, D9, R29, R30, C29 devre elemanları kullanılmıştır.



Şekil 6.4. TNY278PN blok diyagramı



Şekil 6.5. TNY278PN pin konfigürasyonu

TNY278PN smps kontrolör entegre özellikleri:

1. 700V güç mosfeti,
2. 132 kHz osilatör,
3. Seçilebilir yüksek akım sınırlayıcı devre,
4. Termal mosfet kapama koruma devresi,
5. 16 Watt sürekli güç iletimi,
6. Harici mosfet on/off kontrol girişi,
7. Yüksek giriş voltajı koruma devresi,
8. Minimum sayıda harici devre elemanı gereksinimi,
9. Sadece DC giriş gerilimi altında çalışabildiğinden doğrultma devresi gereksinim duyar,

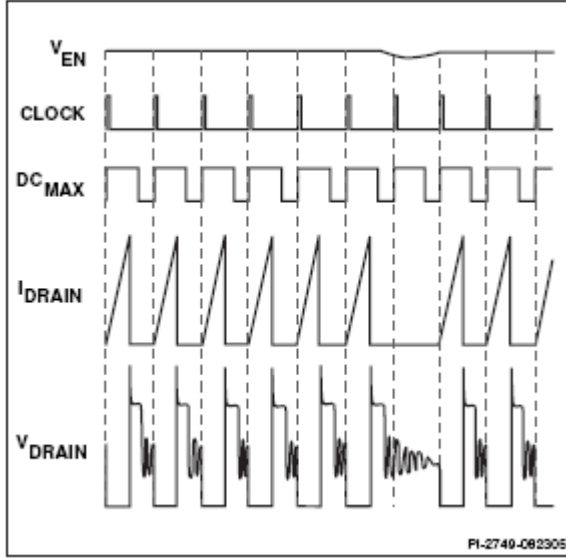
TNY278PN pin görevleri:

D (Drain) ve S (source) pin: Güç mosfeti bacak bağlantısı.

BP/M (bypass/multi-function) pin: Eklenen kapasite değerine bağlı olarak akım limit değerini belirler. 0.1uF kapasite değeri mosfetten maksimum 588mA akım geçmesine izin verir.

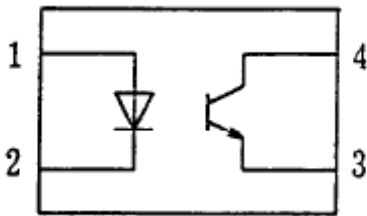
EN/UV (enable/under-voltage) pin: Bu pinin iki görevi vardır: mosfeti on/off kontrol eder ve 60-115 uA eşik akımında mosfeti kapatarak koruma görevi görür.

TNY278PN entegresinin maksimum yük voltajına yakın davranışa Şekil 6.5'teki grafikte gösterilmiştir [7].



Şekil 6.6. TNY278PN entegresi maksimum yüke yakın değerlerde davranışı

R34, D11, R33 ve U5 devre elemanları sabit 24V DC gerilim elde etmek için geribesleme elemanları olarak kullanılır. 24V üzeri gerilimde U5 PC817 optokuplor, aktif hale geldiğinde TNY278PN kontrolörü kapatılır. PC817 devre elemanı, 310V hat gerilimi ile 24V devre çalışma gerilimi arasında izolasyon sağlamak için kullanılır.



Şekil 6.7. PC817 optokuplör iç devre yapısı

PC817 optokuplör led akımı ile transistör akımı arasında bağımlı bir oran söz konusudur.

PC817 optokuplör elektriksel özellikleri aşağıdaki gibidir:

If: Led akımı

Ic: Transistör akımı

$$I_c = \%50-600 \times I_f$$

$$I_{fmax} = 50mA$$

$$V_{led} = \max. 1,4 V$$

$$V_{ce} = \max. 35V$$

Led ile transistör arasındaki izolasyon gerilimi ;

$$V_{isolation} = 5kV$$

R34, D11, R33 ve U5 devre elemanlarının elektriksel değerlerinin hesaplanması aşağıda açıklanmıştır:

$$V_{zener} = 22V$$

$$I_{zmin} = 0,25mA$$

Zenerin devreye girmesi için gerekli minimum akım 0,25mA'dir. Zenerin minimum akımda devreye girmesi durumunda R33 1K'lık direnç üzerinde oluşacak gerilim:

$$V_{R33} = I \cdot R33 = 0,25mA * 1K = 0,25V$$

$$V_{toplam} = V_z + I \cdot (R34 + R33) = 22V + 0,25mA * (688 + 1000)$$

$$V_{toplam} = 22,42 V$$

22,42 V gerilim değerinde zener devreye girecektir. Fakat R33 direnci üzerindeki gerilim $V_{R33} = V_{led} = 0,25V$ olduğundan U5 optokuplör devreye giremeyecektir. U5 optokuplörün devreye girmesi için led yayılım gerilimi $V_{led} = 1,4 V$ olmalıdır. Bu durumda minimum I_{toplam} akım değeri:

$$I_{R33} = \frac{V_{led}}{R33} = \frac{1,4V}{1K} = 1,4mA$$

$$I = I_{R33} + I_f$$

$$I_f = 0 \text{ olduğundan } I = I_{R33}$$

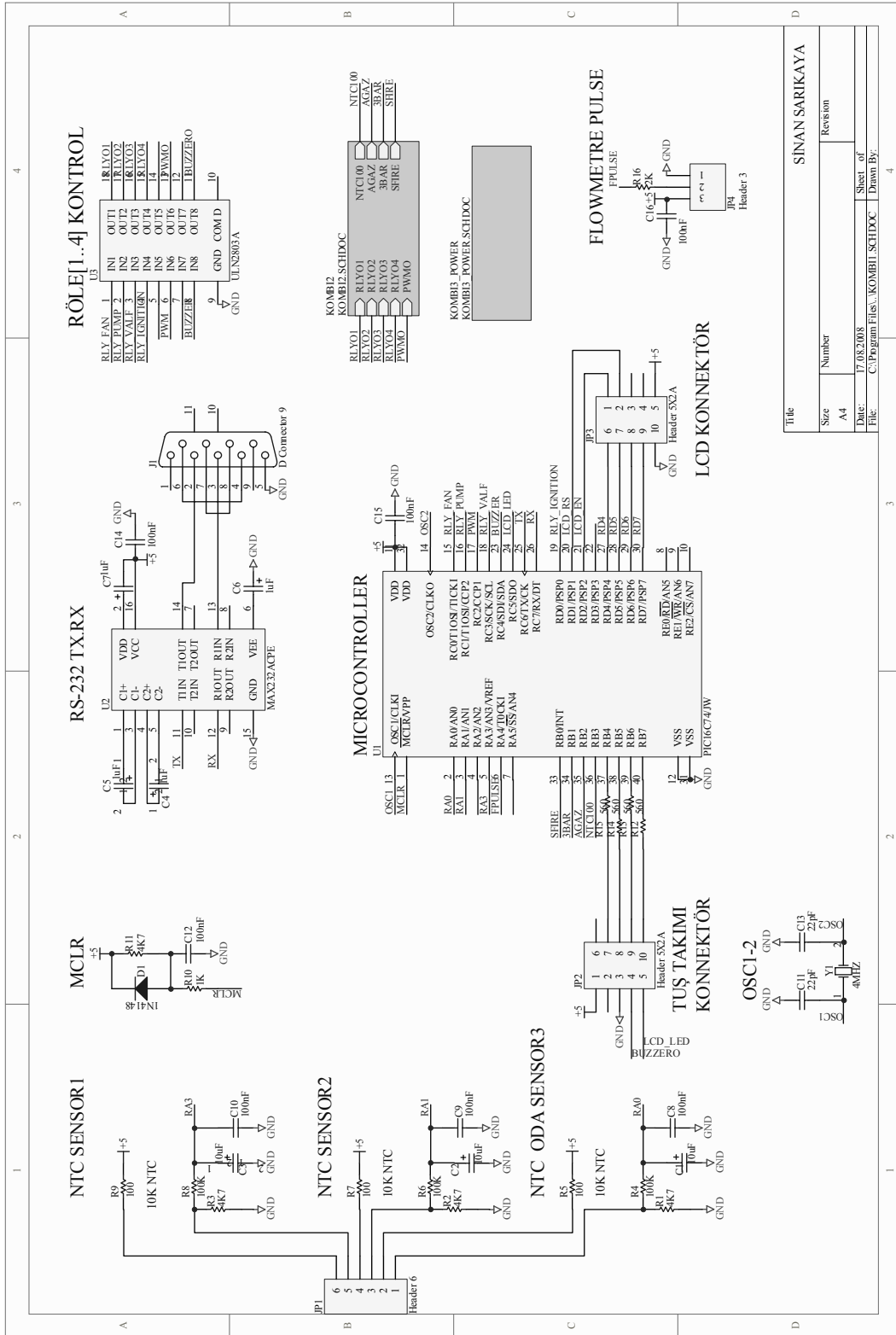
Bu durumda;

$$V_{\text{toplam}} = V_z + I.(R34 + R33) = 22 + 1,4\text{mA} * 1688 = 24,36\text{V}$$

$V_{\text{input}} > 24,36 \text{ V}$ olduğunda, I_f led akımı 0'dan farklı değere ulaşacağından U5 aktif olup TNY278PN kontrolörünü kesime sokacaktır. Böylece 24V besleme geriliminin sabit durumda kalması sağlanmış olacaktır.

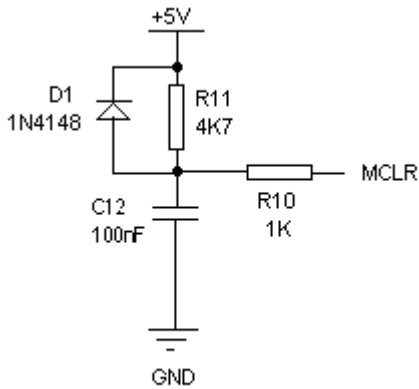
6.1.2. Mikrodenetleyici ve çevre bileşenleri uygulama devresi

Mikrodenetleyici ve genel özellikleri bölüm 4'te incelendi. PIC16F877A mikrodenetleyici pin görevleri, osilatör ve MCLR devresi Şekil 6.8'de gösterilmiştir.



Şekil 6.8. Elektronik kontrol kartı mikrokontrolör ve çevre bileşenleri, NTC sensör, RS-232, flowmetre, röle kontrol entegrasi, LCD ve tuş takımı uygulama devreleri

Şekil 6.9’da MCLR (master clear reset) devresinde PIC16F877A’nın MCLR pini normal çalışmada 5V olması gerekmektedir. Dışarıdan bu pin 0V yapıldığında PIC16F877A’da yüklü yazılım resetlenecektir ve bu pin üzerindeki gerilim tekrar 5V olduğunda program 1. satırdan itibaren yürütülmeye başlayacaktır. Yazılımın güvenle çalışması, MCLR pininin besleme gerilimi ani dalgalanmalarından ve parazitlerinden etkilenip resetlenmemesi için D1, R11, C12, R10 devre elemanlarının kullanılması büyük önem arz etmiştir. Elektriksel olarak bu devre elemanlarını çalışma şekli aşağıda açıklanmıştır.



Şekil 6.9. MCLR uygulama devresi

R11 ve C12 devre elemanları RC alçak geçiren filtre olarak tasarlanmıştır. Yüksek frekanslı bileşenler C12 aracılığıyla toprağa akacaktır, MCLR pini bu durumdan etkilenmeyecektir. D1 diyotu, +5V besleme gerilimi 0V’a düştüğünde, C12 kapasitesinde biriken 5V’luk gerilimin hızlı bir şekilde deşarj olması için kullanılmıştır. 5V besleme gerilimi varken D1 diyotunun devreye bir etkisi yoktur. C12 kapasitesi boşaldığı esnada MCLR pininin kısa süreli yüksek akım çekmemesi için R10 direnci akım sınırlayıcı olarak kullanılmıştır. Aynı zamanda bu devre, 5V besleme geriliminin R11 direnci aracılığıyla C12 kondansatörüne iletildiği ilk anda, MCLR pini üzerinde güvenli bir emniyet gerilimi oluşuncaya kadar PIC16F877A’nın reset konumunda kalmasını sağlayacaktır.

C12 kondansatörünün kaynak gerilimine şarj olması için geçen süre;

$$\tau = R_{11}.C_{12} = 4K7.100nF = 470\mu s$$

$$T_{\text{şarj}} = 5 \tau = 5.470\mu s = 2.35ms$$

C12 kondansatörü 2.35msn'de 5V kaynak gerilimine eşit olacaktır.

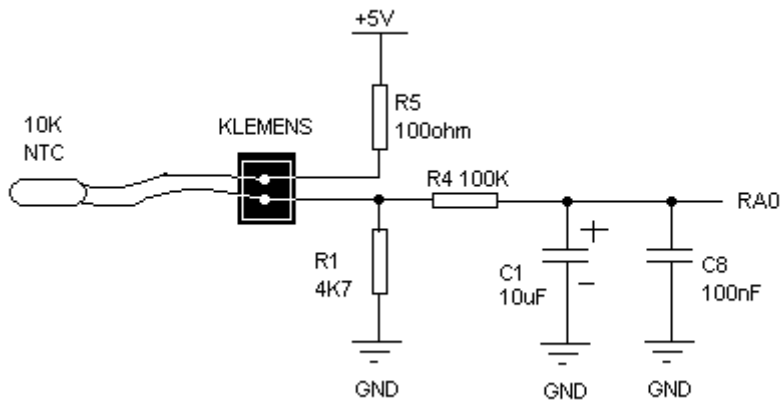
Kaynak gerilimi 0V olduğunda C12 deşarj süresi;

$$T_{\text{deşarj}} = 5\tau = 5.*R_{\text{ileri_diod_ic_direnci}}*C_{12} = \sim 5-50\mu s$$

C15 kondansatörü by-pass kapasite ve PIC16F877A entegre iç yapısındaki mosfetlerin başlangıç ani şarj akımlarını sağlamak için devreye eklenmiştir.

PIC16F877A osilatör devresi için Y1 kristali ve C11, C13 kondansatörleri kullanılmıştır. Y1 = 4MHz için Microchip firması tarafından önerilen C11, C13 kapasite değerleri 22pF olarak seçilmiştir.

6.1.3. NTC termistör kontrol devresi



Şekil 6.10. 10K NTC uygulama devresi

Uygulama devresinde kullanılan R5 direnç, gerçek zamanda uygulama sırasında NTC termistör kablosunun besleme gerilimi ucunun şaseye dokunma ihtimaline karşı, kaynak gerilimini kısa devrelere karşı korumak için kullanılmıştır. R1 direnci,

10k NTC termistör direnç değeriyle gerilim bölücü oluşturarak NTC termistör direnç değerini gerilim yoluyla elde etmemizi sağlar. R4 ve C1 devre elemanları RC alçak geçiren filtre olarak tasarlanmıştır. Bu şekilde ani sıcaklık değişimleri ve elektriksel gürültüler absorbe edilmiş olur. C8 kondansatörü seramik yapıda olduğundan yüksek frekanslı gürültüleri elektrolitik kondansatörlere göre daha iyi absorbe ettiği için kullanılmıştır.

Devre elemanlarının elektriksel değerleri aşağıda açıklanmıştır:

R5 direnç çıkış ucunun kısa devre olması durumunda oluşacak maksimum güç tüketimi;

$$P_{R5} = \frac{V^2}{R1} = \frac{5^2}{100} = 0,25 \text{ watt}$$

NTC termistör direncinin ölçümü için R1 direnci üzerindeki gerilim değeri;

$$V_{R1} = \frac{R1}{R1 + R_{NTC} + R5} \cdot V_{kaynak} = \frac{4700}{4700 + R_{NTC} + 100} \cdot 5 \quad [6.11]$$

V_{R1} gerilim değerinin ölçülmesinde; Tablo 6.1’de her bir sıcaklığa denk gelen NTC direnç değeri tek tek girilerek V_{R1} gerilim değeri hesaplanır. Elde edilen gerilim değeri PIC16F877A ADC(Analog-digital converter) birimi tarafından sayısal değerlere çevrilir. Tüm sayısal değerler bir araya getirilerek bir tablo oluşturulur. Böylece NTC termistör direncinin sayısal sonucuna denk gelen sıcaklık değeri mikrodenetleyici tarafından belirlenerek kullanıcıya sunulur. Projede kullanılan NTC, Honeywell 10k-B serisine aittir [6].

ÖRNEK 1:

Tablo 6.1’den 10k-B direnç-sıcaklık tablosundan 10C°’ye denk gelen direnç değeri bulunur. Bulunan değer [6.1] denkleminde R_{NTC} yerine konular ve gerilim değeri hesaplanır.

Tablo 6.1'den 10 C° için $R_{NTC} = 17820\Omega$

[6.11] denkleminde;

$$V_{RI} = \frac{4700}{4700 + R_{NTC} + 100} \cdot 5 = \frac{4700}{4700 + 17820 + 100} \cdot 5 = 1,039V \text{ bulunur.}$$

$V_{RI} = 1,039 V$ gerilim değeri PIC16F877A mikrodenetleyicisinin ADC birimi tarafından sayısal değere çevrilir. ADC birimi 10 bit sayısal çözünürlüğe sahiptir. Vreferans gerilim değeri 5 volt seçilmiştir.

$$2^{10} = 1024$$

$$1 \text{ sayısal değeri} = \frac{5V}{1024} = 4,88 \text{ mV}$$

$$V_{RI} \text{ gerilimi sayısal değeri} = \frac{1039mV}{4,88mV} = \sim 213 = 0D5H$$

ÖRNEK 2: Tablo 6.1'den 11 C° ve 12 C°'ye denk gelen R_{NTC} değerlerini bulunuz?

$$10 \text{ C}^\circ \text{deki } R_{NTC} = 17820\Omega$$

10 -20 C° arasındaki her bir sıcaklık derecesi değişim yüzdesi = - %4,1

$$\text{Bu durumda 11 C}^\circ \text{ için } R_{NTC} = 17820 - 17820 \cdot 0,041 = 17089 \Omega$$

$$12 \text{ C}^\circ \text{ için } R_{NTC} = 17089 - 17089 \cdot 0,041 = 16388 \Omega$$

Tablo 6.1. Honeywell T7335 NTC termistör sıcaklık-direnç tablosu

Steinhard & Hart Coeff.	Curves							
	2.7k		10k-A		10k-B		12k-A	
C	1.94567E-18		8.77547E-08		1.89916E-07		1.28318E-07	
B	2.51842E-04		2.34108E-04		2.52293E-04		2.40423E-04	
A	1.36421E-03		1.12924E-03		8.84204E-04		9.89458E-04	
f	1		1.01707E+00		1		1	
T [°C]	R [Ω]	α [1/°C]	R [Ω]	α [1/°C]	R [Ω]	α [1/°C]	R [Ω]	α [1/°C]
-20	29.30k	-6.2%	98.66k	-5.8%	67.38k	-4.8%	98.97k	-5.4%
-10	16.14k	-5.7%	56.25k	-5.4%	42.14k	-4.6%	58.88k	-5.0%
0	9.292k	-5.3%	33.21k	-5.1%	27.06k	-4.3%	36.13k	-4.7%
10	5.561k	-5.0%	20.24k	-4.8%	17.82k	-4.1%	22.80k	-4.5%
20	3.446k	-4.6%	12.71k	-4.5%	12.00k	-3.8%	14.77k	-4.2%
25	2.746k	-4.5%	10.17k	-4.4%	9.92k	-3.7%	12.00k	-4.1%
30	2.205k	-4.3%	8.194k	-4.3%	8.251k	-3.6%	9.804k	-4.0%
40	1.451k	-4.0%	5.416k	-4.0%	5.786k	-3.5%	6.652k	-3.8%
50	980.1	-3.8%	3.663k	-3.8%	4.132k	-3.3%	4.607k	-3.6%
60	677.8	-3.6%	2.530k	-3.6%	3.000k	-3.1%	3.252k	-3.4%
70	478.9	-3.4%	1.782k	-3.4%	2.213k	-3.0%	2.337k	-3.2%
80	345.1	-3.2%	1.278k	-3.2%	1.656k	-2.8%	1.707k	-3.1%
85	295.0	-3.1%	1.089k	-3.2%	1.441k	-2.8%	1.467k	-3.0%
90	253.2	-3.0%	931.6	-3.1%	1256.9	-2.7%	1.266k	-2.9%
100	188.9	-2.9%	690.0	-2.9%	965.9	-2.6%	952.3	-2.8%
110	143.1	-2.7%	518.5	-2.8%	751.2	-2.5%	726.0	-2.6%
120	109.9	-2.6%	395.0	-2.7%	590.8	-2.3%	560.4	-2.5%
125	96.83	-2.5%	346.4	-2.6%	526.0	-2.3%	494.6	-2.5%

Sıcaklık değişiminin hesaplanması için minimum sıcaklık kararlılık süresi (C1 kondansatörünün V_{R1} gerilimini tamamen şarj etmesi için gerekli süre) ;

$$\tau = \left(\frac{R5 + R_{NTC}}{R1} + R4 \right) \cdot C1 \quad [6.12]$$

$$\frac{R5 + R_{NTC}}{R1} = \frac{(R5 + R_{NTC}) \cdot R1}{R5 + R_{NTC} + R1}$$

$$\tau = \left(\frac{(R5 + R_{NTC}) \cdot R1}{R5 + R_{NTC} + R1} + R4 \right) \cdot C1$$

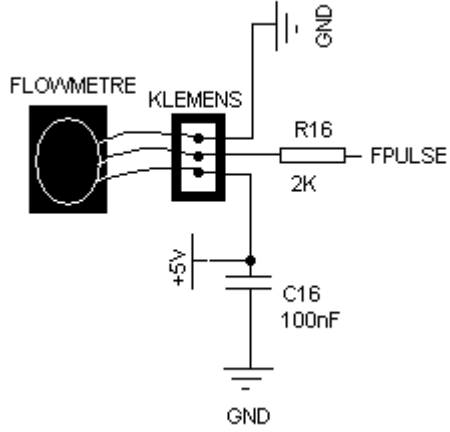
$$T = 5 \cdot \tau = 5 \cdot \left(\frac{(R5 + R_{NTC}) \cdot R1}{R5 + R_{NTC} + R1} + R4 \right) \cdot C1 \quad [6.13]$$

10C° için minimum sıcaklık kararlılık süresi;

$$T = 5 \cdot \left(\frac{(R5 + R_{NTC}) \cdot R1}{R5 + R_{NTC} + R1} + R4 \right) \cdot C1 = 5 \cdot \left(\frac{(100 + 17820) \cdot 4700}{100 + 17820 + 4700} + 100000 \right) \cdot 10\mu F$$

$$T = 5sn$$

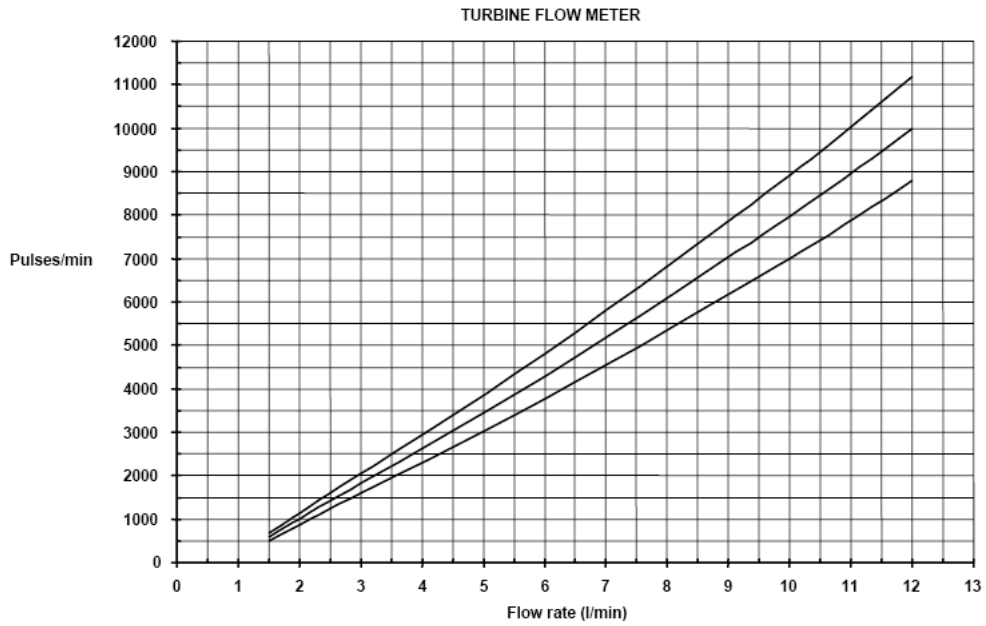
6.1.4. Açık çevrim sistemindeki su debisini ölçen flowmetre sinyal işleme uygulama devresi



Şekil 6.11. Flowmetre uygulama devresi

Bölüm 5’te FUGAS-TURBINA-SPT102 flowmetre özellikleri açıklandı. Devre elemanlarının çalışma şekli aşağıda açıklanmıştır:

C16 kondansatörü bypass kapasite olarak kullanılmıştır. R16 2K direnç PIC16F877A fpulse pinini akım sınırlayıcı olarak korumak için devreye eklenmiştir.



Şekil 6.12. FUGAS-TURBINA-SPT102 flowmetre su akış miktarı-pulse ilişkisi

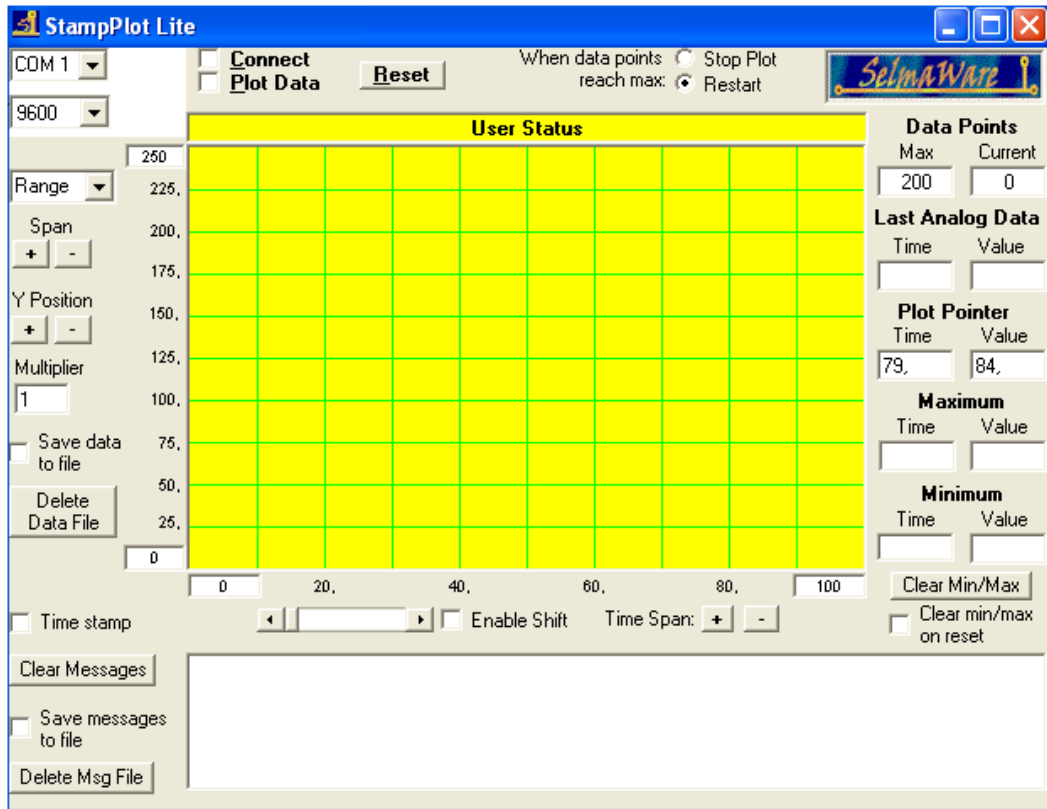
FUGAS-TURBINA-SPT102 flowmetrenin ölçüm yapması için minimum akış debisi 1,5 litre/dakika'dır, ölçülebilen maksimum akış debisi ise 12 litre/dakika'dır. Minimum sınırın altındaki değerlerde flowmetre pulse üretmeyecektir. Şekil 6.12'den bir dakikada geçen su akış miktarı tablodan bakılarak öğrenilebilir. Örneğin 2 litre/dakika için 1200 pulse /dakika -20 Hz'lik 5V DC sinyal FUGAS-TURBINA-SPT102 flowmetre tarafından üretilir. Bu sinyal PIC16F877A sayıcı birimi tarafından sayılır ve akış miktarı sayısal değer olarak ölçülür. 1.5- 12 litre akış aralığında her bir akış miktarı için bir tablo oluşturularak, mikrodenetleyici tarafından mevcut akış miktarı kolaylıkla öğrenilir.

6.1.5. Elektronik kontrol kartı-PC RS-232 haberleşme portu uygulama devresi

Uygulama devresi Şekil 6.8'de gösterilmiştir. İlgili tüm devre elemanları MAX-232 entegre üreticisi firma tarafından önerilmiştir. Projede RS-232 haberleşme portu başta sıcaklık değeri olmak üzere çeşitli verilerin, elektronik karttan bilgisayar ortamına veya verilerin bilgisayar ortamından elektronik karta iletilmesi için kullanılması amaçlanmıştır.

Projede PID açık çevrim fonksiyonunun elde edilmesinde elektronik karttan sıcaklık değerleri 1sn periyotlarla bilgisayar ortamına aktarılmıştır. Verilerin değerlendirilmesinde ve grafik haline getirilmesinde Selmaware Solutions firması tarafından geliştirilen Stamp Plot Lite Version 1.7 programı kullanılmıştır. Verilerin izlenmesi için elektronik karttan sayısal rakamların ASCII karakterleri seri porta gönderilmiştir.

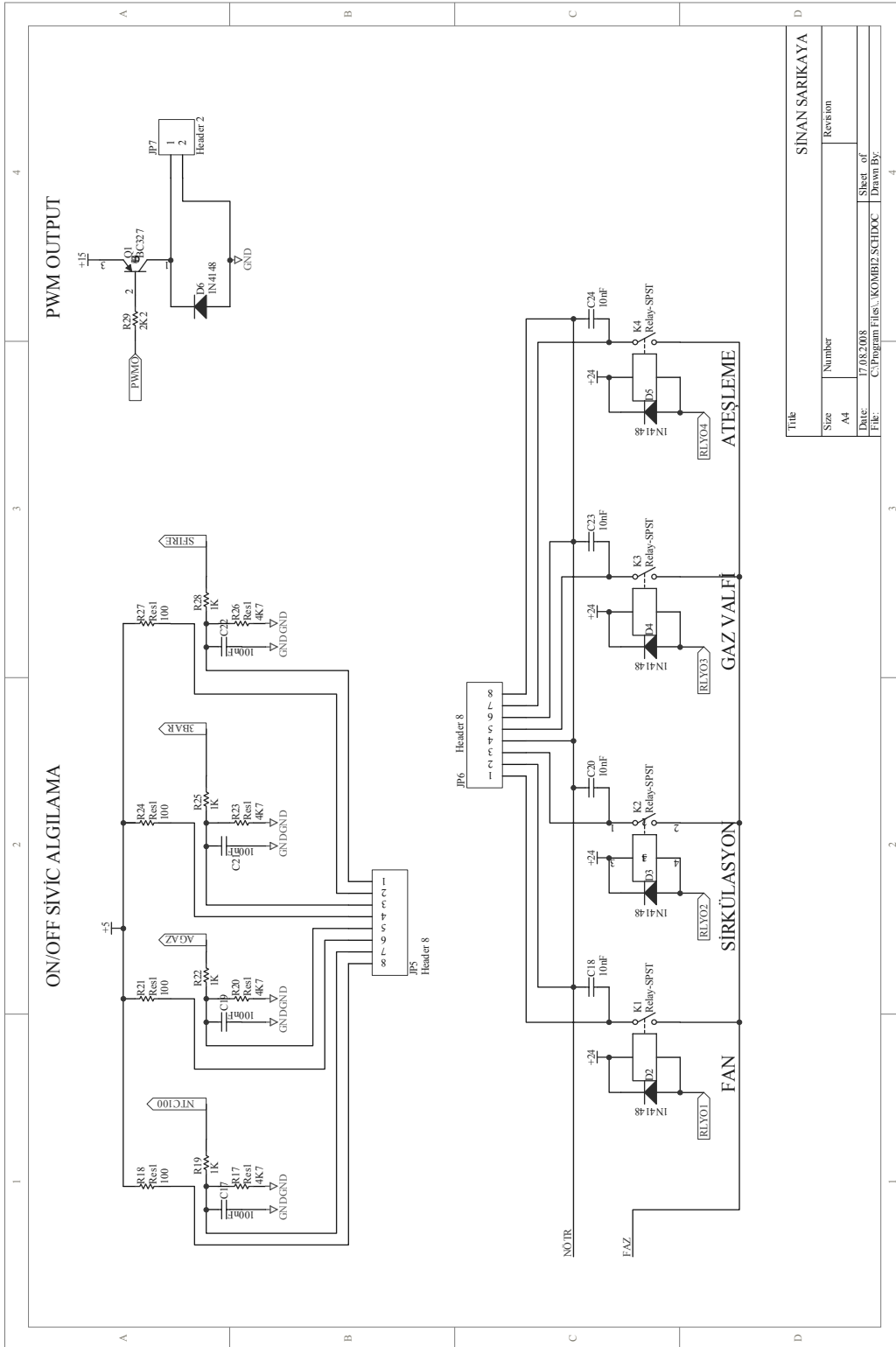
Örneğin 20 C° sıcaklık değerini seri porta aktarmak için iki karakter arasında gecikme olmaksızın önce 1. veri: 32H (30H+02H) onlar basamağını, sonra 2.veri: 30H (30H+00H) birler basamağı bilgileri seri porttan bilgisayar programına aktarılmıştır.



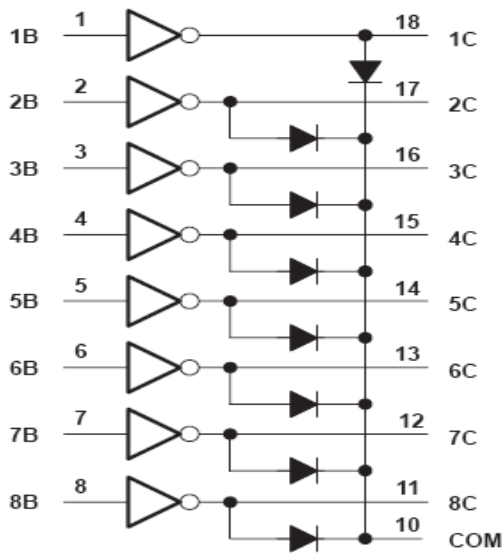
Şekil 6.13. Stamp Plot Lite Version 7.0 veri izleme programı

6.1.6. 220VAC beslemeli ekipmanların röle kontrolü uygulama devresi

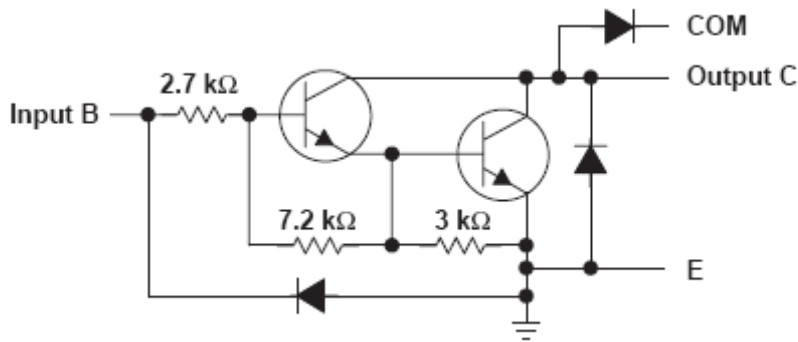
Kombi ısıtma sisteminde fan, gaz valfi, kapalı çevrim su sirkülasyon pompası ve gaz ateşleme trafosu kontrol gerilimleri 220 VAC olduğundan 24 Volt DC tetiklemeli röleler kullanılmıştır. Röle kontrol entegresi olarak ULN2803A entegresi kullanılmıştır, Şekil 6.8 ve Şekil 6.14'te röle kontrol uygulama devreleri gösterilmiştir. Bu entegrenin kullanılma sebebi; her bir röle kontrolü için bir direnç ve bir transistör kullanılması gerektiğinden, entegre kullanarak elektronik kart üzerinde eleman yoğunluğunu önlemek amaçlanmıştır. ULN2803A entegresi 8 giriş ve 8 çıkışa sahip bir sürücü entegredir [8].



Şekil 6.14. On/off siviçi algılama devresi, röle kontrol uygulama devresi devamı ve gaz valfi modülasyon bobin PWM uygulama devresi



Şekil 6.15. ULN2803A entegre blok diyagramı



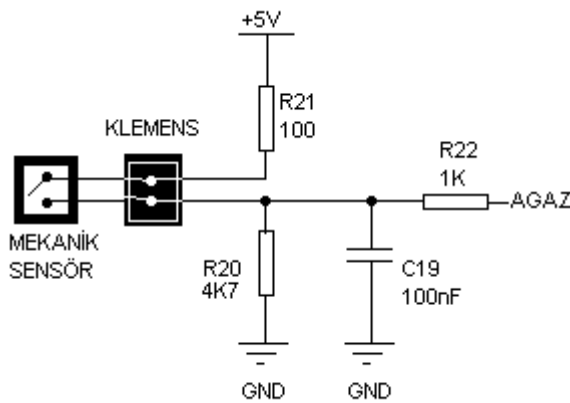
Şekil 6.16. ULN2803A darlington transistör devre şeması

ULN2803A entegre teknik özellikleri;

- 1- Maksimum 50Volt VCE gerilimi,
- 2- 500mA transistör kolektör akımı,
- 3- $V_{input} = 3,85 \text{ V}$ 'ta maksimum 1,25mA base akımı gereksinimi,
- 4- $T_{plh} = 130\text{ns}$, (low-to-high yayılım gecikmesi)
- 5- $T_{phl} = 20\text{ns}$, (high-to-low yayılım gecikmesi)

Şekil 6.13'te röle kontrol uygulama devresinde kullanılan D2 diyodu röle kapandığı anda röle bobini tarafından indüklenen ters emk gerilimini söndürmek için kullanılmıştır. C18-1000VAC kondansatörü yük geriliminin kesilmesinden sonra, yük bobini üzerindeki enerji tam boşalmadan tekrar ani besleme yapılması durumunda röle kontakları arasında oluşabilecek arkı engellemek için kullanılmıştır.

6.1.7. Mekanik sensörlerin on/off kontak durumlarını algılama devresi



Şekil 6.17. On-off kontak durumu algılama devresi

Projede atık gaz basınç prosestatı, bimetel sıcaklık algılama sensörü, su basınç prosestatı genel özellikleri bölüm 5'te açıklandı. Kontaklar normalde açık pozisyonundadır. Algılama durumunda bu sistem elemanları kontakları kapatır. Kontak durumunu algılamak için Şekil 6.17'deki devre tasarlanmıştır. Devre elemanlarının elektriksel çalışma şekli aşağıda açıklanmıştır.

R21 direnci kaynak gerilimini korumak için seri olarak mekanik sensör kontaklarına bağlanmıştır. R20 devre elemanı gerilim bölücü olarak kontak durumunu belirleyebilmek için devrede kullanılmıştır. C19 kondansatörü elektriksel gürültülerden ve kaynak gerilimi ani dalgalanmalarından AGAZ pininin etkilenmemesi için R20 direncine paralel bağlanmıştır. R22 direnci dışarıdan gelebilecek tehlikeli genlikteki sinyallerinin AGAZ pini üzerinde etkisini azaltır. Mikrodenetleyici pinlerinin her biri için entegre üzerinde ESD diyotlar

yerleştirilmiştir. Dışarıdan gelebilecek negatif yada 5V üzeri pozitif genlikteki sinyaller R22 direncinin olmadığı durumda bu diyotları bozacaktır.

Kontak açık durumu;

$$V_{R20} = 0 \text{ V}$$

Kontak kapalı durumu;

$$V_{R20} = \frac{R20}{R20 + R21} \cdot V_{kaynak} = \frac{4700}{4700 + 100} \cdot 5$$

$$V_{R20} = 4,9 \text{ Volt}$$

6.1.8. Gaz valfi modülasyon bobini kontrol devresi

Uygulama devresi Şekil 6.14'te gösterilmiştir. ULN2803A entegresi çıkış pini R29 direncine bağlanmıştır. ULN2803A entegresi giriş pinine mikrodenetleyici tarafından 5V uygulandığında ULN çıkış pini sıfır olacaktır ve R29 direnci aracılığıyla Q1 pnp transistörünü aktif edecektir. ULN2803A entegresi giriş pini 0V yapıldığında entegre çıkış pini yüksek direnç konumuna geçeceğinden Q1 transistörü kesime gidecektir. ULN2803A giriş pinine PWM sinyal uygulanarak 15V'lük gerilim frekansa bağlı olarak modülasyon bobinine verilir. Modülasyon bobini maksimum akım tüketimi 350mA' dir ve bu sebeple maksimum 500mA kolektör akımına sahip BC327 pnp transistörü seçilmiştir. D6 diyodu Q1 transistörü kesime gittiği anda modülasyon bobinin oluşturacağı ters emk gerilimini söndürür ve böylece Q1 transistörünü de korur. Q1 transistörü, PWM sinyalinin kapalı olduğu durumlarda 15V kaynak gerilimini dış dünyadan izole etmek için kullanılmıştır.

Devre elemanları elektriksel değerlerinin belirlenmesi:

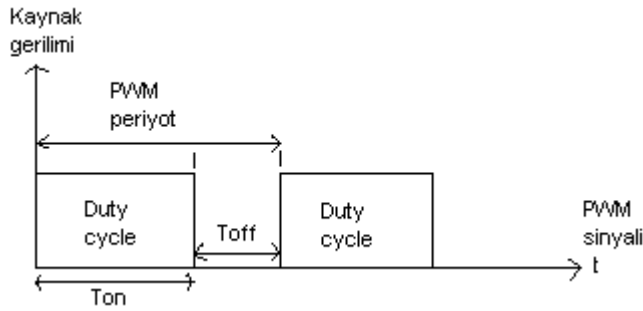
ULN2803A entegresi çıkış pini aktif olduğunda Q1 transistörü base akımı;

$$-15V + I_{base} \cdot R_{29} + V_{Q1BE} + V_{ULN_CE} = 0$$

$V_{Q1BE} = V_{ULN_CE} = \sim 0,7 \text{ V}$ kolektör-emiter doyum gerilimleri

$$I_{\text{base}} = \frac{15 - 2 \cdot 0,7}{2K2} = 6,18\text{mA}$$

6,18mA I_{base} akımı Q1 transistörünü doyumda çalıştırabilecektir.



Şekil 6.18. PWM sinyali

PWM sinyali ortalama gerilimi;

$$V_{\text{ort}} = \frac{T_{\text{ON}}}{T_{\text{ON}} + T_{\text{OFF}}} \cdot V_{\text{kaynak gerilimi}} \quad [6.14]$$

PIC16F877A RC2 pini PWM sinyal üretici çıkış pini olarak ayarlanmıştır. İstenen gerilimin elde edilmesi için PWM sinyalin duty cycle(T_{ON})'ı değiştirilir. Projede PWM frekansı 16KHz olarak gaz valfi modülasyon bobinine uygulanmıştır. Gaz valfi modülasyon bobini kontrol gerilimi 0- 15 Volt aralığında olduğu için kaynak gerilimi 15 V olarak uygulanmıştır.

Örneğin gaz valfi modülasyon bobinine 9V gerilim uygulanmak istenirse T_{ON} süresini bulalım;

PWM frekans = 16 KHz

$$\text{PWM periyot} = \frac{1}{f} = \frac{1}{16\text{Khz}} = 62\mu\text{sn}$$

[6.14] denkleminde;

$$9 \text{ V} = \frac{T_{\text{ON}}}{T_{\text{ON}} + T_{\text{OFF}}} \cdot 15 \text{ V}$$

$$\frac{T_{\text{ON}}}{T_{\text{ON}} + T_{\text{OFF}}} = \frac{9}{15} = \frac{3k}{5k}$$

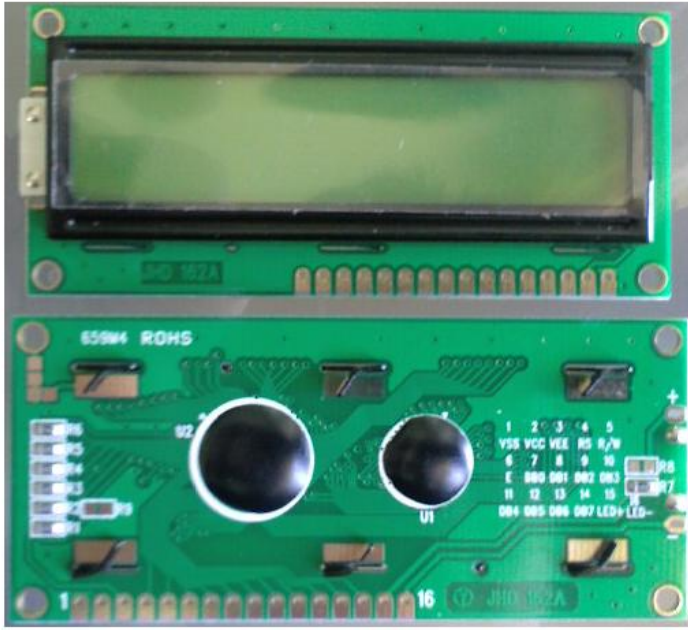
$$\text{PWM periyot} = T_{\text{ON}} + T_{\text{OFF}} = 62 \mu\text{s}$$

$$T_{\text{ON}} = 37 \mu\text{s} \text{ bulunur.}$$

6.1.9. LCD ve tuş takımı uygulama devresi

LCD display olarak 16 pinli JHD 162A display kullanılmıştır. LCD display özellikleri ve pin görevleri:

- 4/8 bit'lik veri ile bilgi alışverişi,
- 2 satırlı, her satırda 16 karakter gösterebilme (2x16 LCD),
- Ledlerle onboard iç aydınlatma kontrolü,
- Kontrast ayarlama özelliği,
- 2 bit'lik komut pini ile kontrol edilebilme,
- Her satırda maksimum 40 karakter olmak üzere toplam 2x40=80 karakter gösterebilme ve 64 byte-CGRAM(character generator RAM) kullanıcıya özgü 8 karakter oluşturabilme,



Şekil 6.19. LCD display ön ve arka yüzey genel görünümü

LCD bacak bağlantılarının görevi:

VSS, VCC = LCD besleme gerilimi,

EN(ENABLE) = LCD data/komut aktif etme,

RS(Read/save) = 1/0 (LCD'den okuma /LCD'ye gönderme),

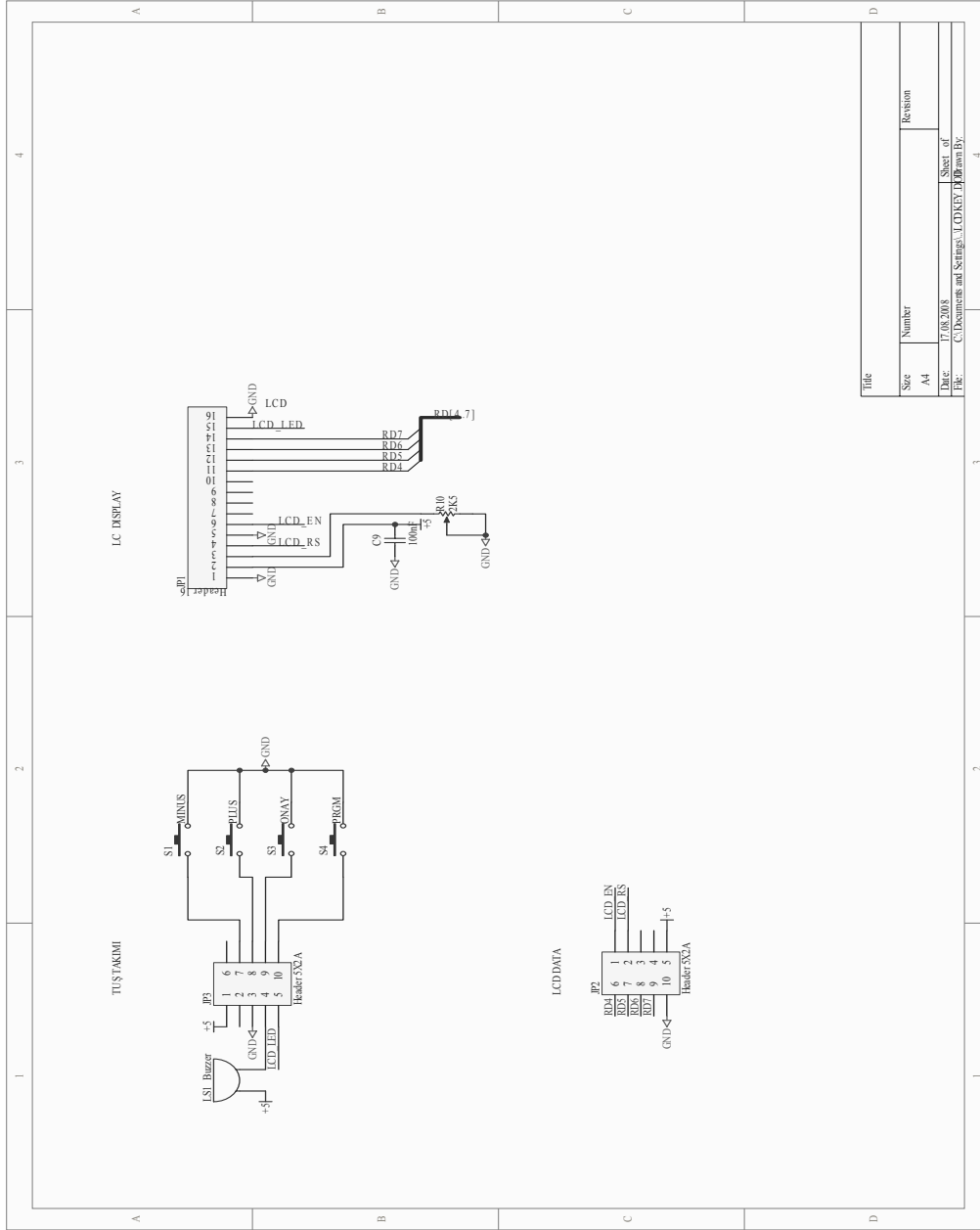
Tasarımda bu pin toprağa bağlanmıştır. Tek yönlü bilgi alışverişi var,

R/W = 1/0 (Data/Komut),

LED+, LED- = LCD onboard ledlerle aydınlatmayı aktif etme,

DB7::DB0 = 8 bit data

Tasarımda 4 bit [DB7:DB6:DB5:DB4] data kullanılmıştır.



Şekil 6.20. LCD display ve tuş takımı uygulama devresi

Şekil 6.20’de gösterilen LCD uygulama devresinde R10 trimpotu LCD ekran kontrastını ayarlamak için kullanılmıştır. C9 kondansatörü by-pass kapasite olarak ve LCD entegre iç devre yapısında bulunan mosfetlerin ani şarj akımlarını sağlamak için eklenmiştir.

Tuş takımı uygulama devresi Şekil 6.20’de gösterilmiştir. Tuş takımı kullanıcının, sıcaklık set değerlerini kaydetmesini ve isteğe bağlı olarak menü yazılımında çeşitli bilgileri kaydetmesini veya okumasını sağlamak için tasarlanmıştır. Tuş takımı uygulama devresindeki tuş algılama bağlantıları PIC16F877A mikrodenetleyicisinin PORTB portunun RB7, RB6, RB5, RB4 bacaklarına bağlanmıştır. PORTB RB7::RB4 pinlerinin pull-up dirençleri aktif edilmiştir. Herhangi bir tuşun basılı olmadığı durumda bu pinlerin gerilimleri 5V’tur, tuş basıldığında ilgili pin gerilimi 0V seviyesine düşecektir. Bu şekilde yazılımla hangi tuşun basıldığı kolaylıkla algılanmaktadır. Tuş takımının parazitlere karşı koruması yazılımla yapılmıştır. Buzzer, herhangi bir tuşa basıldığını veya oluşan bir alarm durumunu sesli uyarım yoluyla kullanıcıya iletmek için kullanılmıştır.

4 tuşla bilgi kontrolü yapılmaktadır:

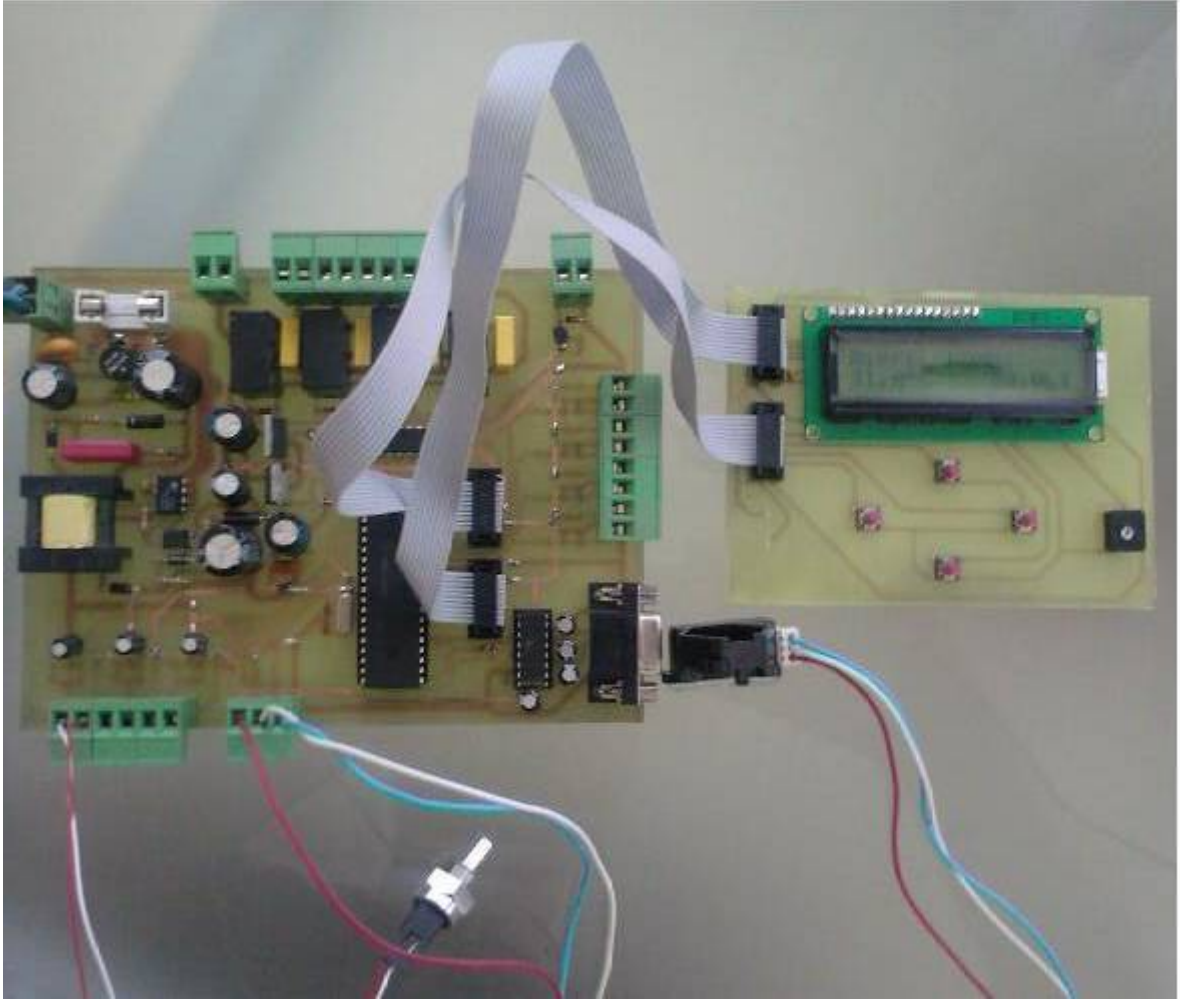
“ENTER/SAVE” tuşu: “Enter/Save” tuşu seçilen menü alt programına girmek ve girilen menüde ayarlanan set değerini kaydetmeyi sağlar,

“MENU/EXIT” tuşu: “Menu/Exit” tuşu normal bilgi ekranından ana menüye girmek veya gelinen menüye geri dönmek ve aynı zamanda girilen herhangi bir set değeri kaydetmeden çıkma görevlerini sağlar,

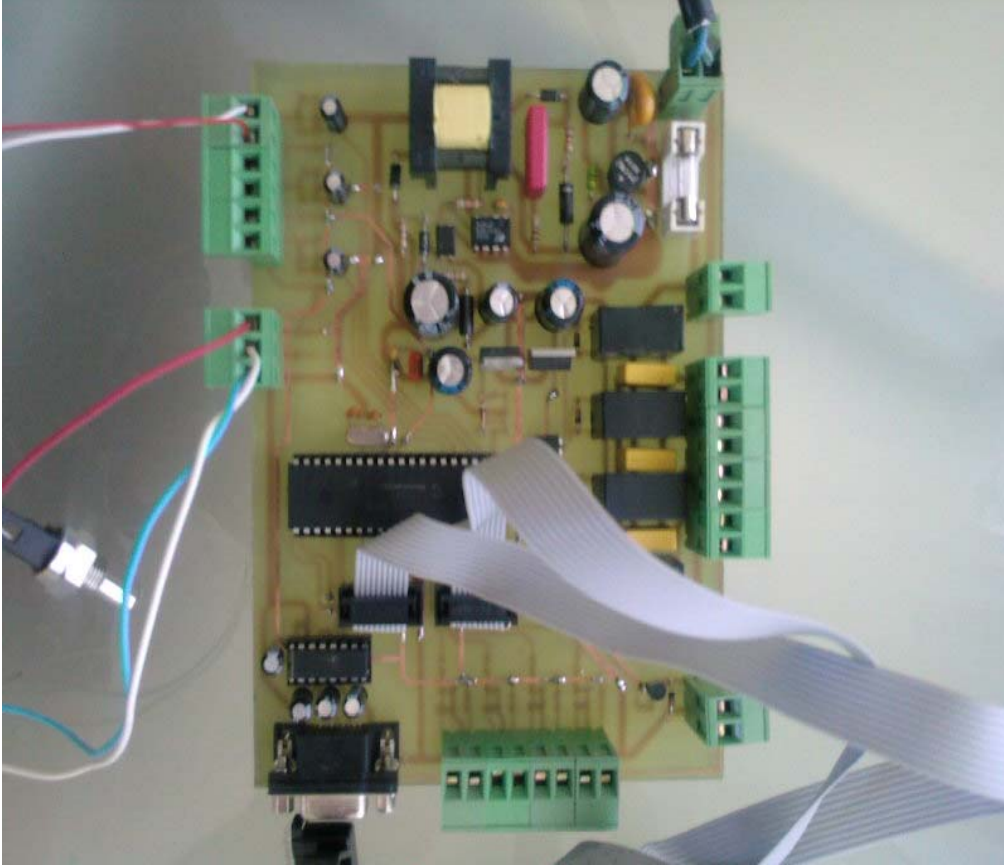
“Yukarı/aşağı ok” tuşları: İlgili menüde girilen set değerleri artırmak / azaltmak veya ana menüdeki alt menüleri dolaşmak için kullanılır.

6.2. Elektronik Kart Resimleri

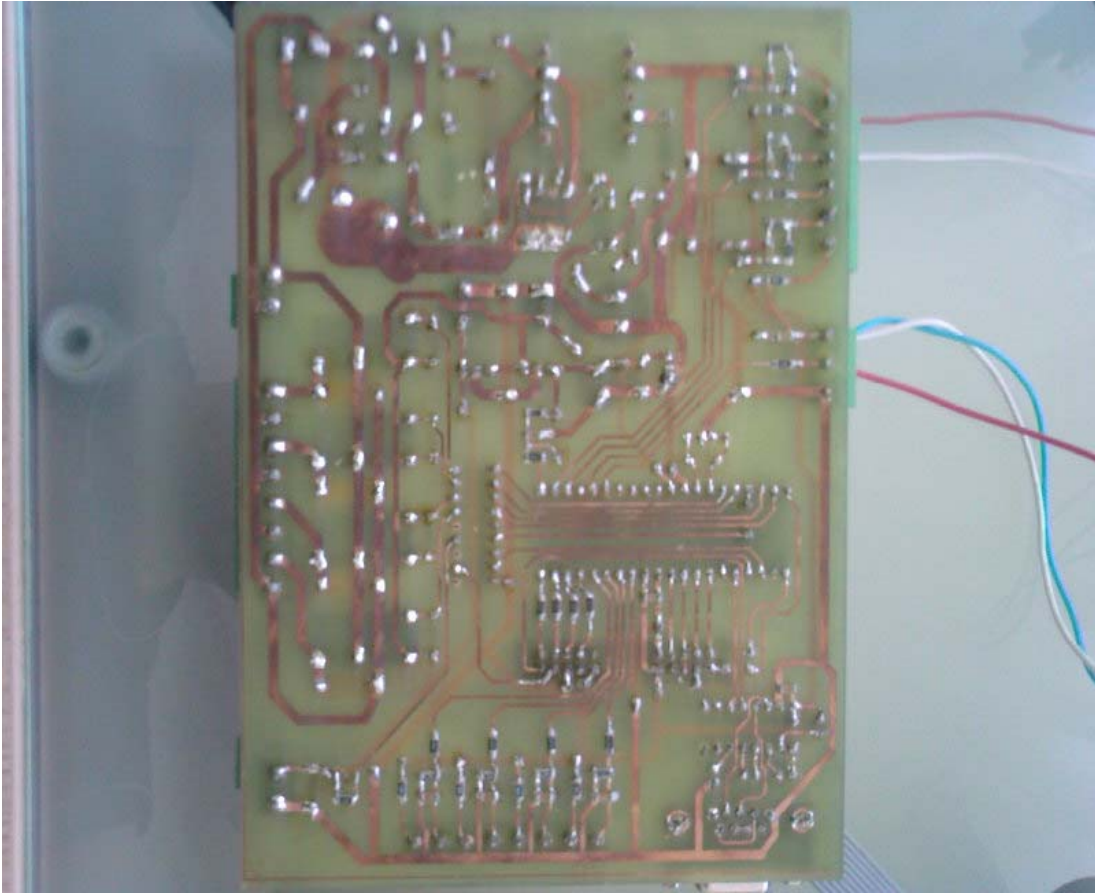
Tasarlanan elektronik kartın genel görünümleri Şekil 6.21, Şekil 6.22, Şekil 6.23 ve Şekil 6.24’de gösterilmiştir. Ana kart, elektronik devre elemanlarının PCB kartın ön ve arka yüzeyine lehimlenmesi ile oluşturulmuştur. Kart tasarımında, assembly yüzey montaj devre elemanları yoğunluklu olarak kullanılmıştır. İlgili kartların tüm fonksiyonları denenmiştir.



Şekil 6.21. Ana kart ve keypad



Şekil 6.22. Ana kart PCB ön yüzey



Şekil 6.23. Ana kart PCB arka yüzey

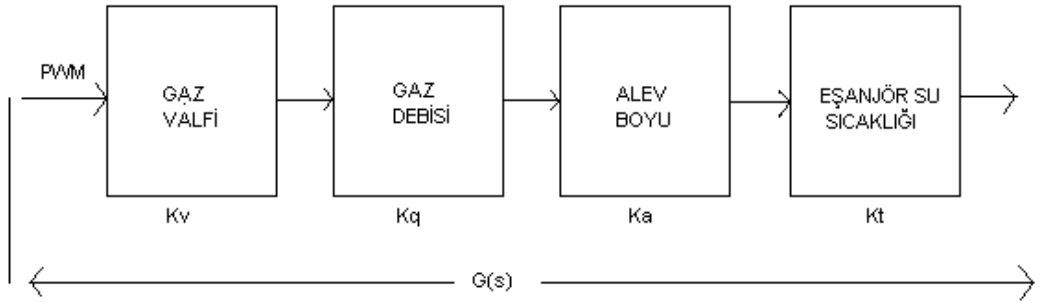


Şekil 6.24. Keypad

BÖLÜM 7. SİSTEMİN MATEMATİKSEL MODELİNİN ELDE EDİLMESİ

7.1. Sistemin Modellenmesi

PID kontrolör tasarımı Ziegler-Nichols yöntemi bölüm 4'te detaylı olarak incelendi. Bölüm 4'te elde edilen bilgiler doğrultusunda açık çevrim ünitesi blok diyagramını Şekil 7.1'deki oluşturabiliriz.



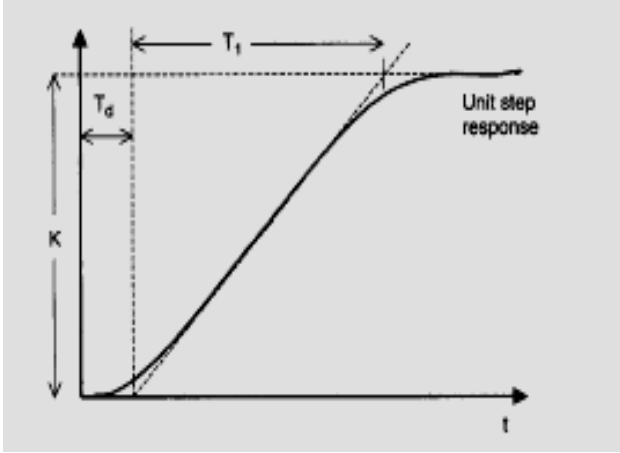
Şekil 7.1. Sistemin açık çevrim ünitesi transfer fonksiyonunun elde edilmesi

Şekil 7.1'de gösterildiği gibi sisteme dışarıdan PWM sinyali gaz valfi modülasyon bobinine uygulanır. Gaz valfi modülasyon bobini şehir gaz şebekesinden gelen gazın debisini kontrol ederek kombi yanma odasında oluşan ateşin alev boyunu kontrol eder. Kombi yanma odasında oluşan alevin büyüklüğüyle orantılı olarak eşanjör içerisinde geçen soğuk su ısıtılır ve ısıtılan su kapalı çevrim su sirkülasyon pompasıyla istenilen noktaya ulaştırılır.

Ziegler-Nichols'a göre açık-çevrim işlemi yaklaşık transfer fonksiyonu;

$$G(s) = \frac{K.e^{-s.T_d}}{(1 + s.T_1)} \quad [7.1]$$

K , T_d ve T_1 parametreleri Şekil 7.2'de gösterilen açık-çevrim ünitesi cevabıyla bulunabilir.



Şekil 7.2. Açık-çevrim yöntemiyle bir prosesin K , T_d ve T_1 parametrelerini bulmak

7.2. Deneysel Olarak Sistemin Açık Çevrim Transfer Fonksiyonunun Elde Edilmesi

Önceki bölümlerde elde edilen tüm bilgiler doğrultusunda ve elektronik kart tasarımının tamamlanmasından sonra sistemin PID kontrol açık çevrim cevabı incelenmiştir.

Sistemin açık çevrim fonksiyonun elde edilmesi için dizayn edilen elektronik kartla gerçek kombi sistem elemanlarının kontrolü yapılmıştır. Test ortamı için AIRFEL A.Ş. firmasının kombi test laboratuvarı kullanılmıştır. Gerçek ortamda elektronik kartın tüm fonksiyonları denenerak elde edilen bilgiler ile kombi test kontrol istasyonundaki bilgilerin karşılaştırılması yapıp sistemin doğruluğu test edilmiştir. Karşılaştırılan fonksiyonlar; sıcaklık algılama, flowmetre akış kontrol, gaz valfi modülasyon sonucu elde edilen alev boyutu. Bu üç fonksiyonun doğruluğu bir kombinin kalitesini ve güvenliğini doğrudan etkilediği için çok önemli yer teşkil etmektedir.

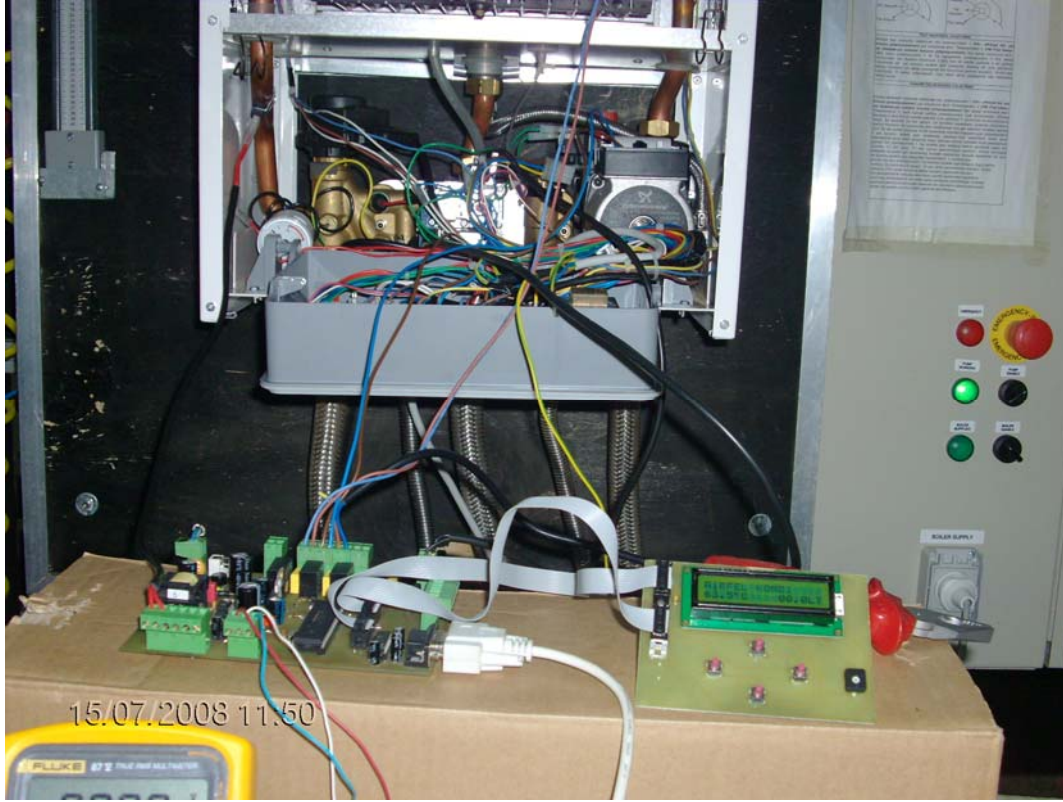
Atık gaz basınç prosestatı, su sirkülasyon pompası, fan, su basınç prosestatı başta olmak üzere diğer sistem elemanlarının kontrolü test edilerek elektronik kart üzerindeki uygulama devrelerinin doğruluğu test edilmiştir ve herhangi olumsuz bir sonuçla karşılaşılmamıştır.

Şekil 7.3'te kombi test laboratuvarı gösterilmiştir. Kombi test istasyonu deney ortamı olarak ve aynı zamanda sistem tasarımındaki değerlerin karşılaştırılmasında referans olarak kullanılmıştır. Kombi test istasyonunda bir kombinin tüm fonksiyonları denenmektedir. Deney sırasında elde edilen tüm fonksiyonlar harici olarak supervisor bilgisayarında gözlenebilmekte ve bazı fonksiyonlar simülasyon için kontrol edilebilmektedir. Gerçek simülasyon için kombi çıkışındaki su sıcaklığı ile dönüş suyu sıcaklığı arasındaki sıcaklık farkı istenen set değerinde sabit tutulabilmektedir. Bu yolla kombi tarafından ısıtılan sıcak su, gerçek ortamda kullanılmıyormuş gibi simülasyon imkanı sağlanabilmektedir.



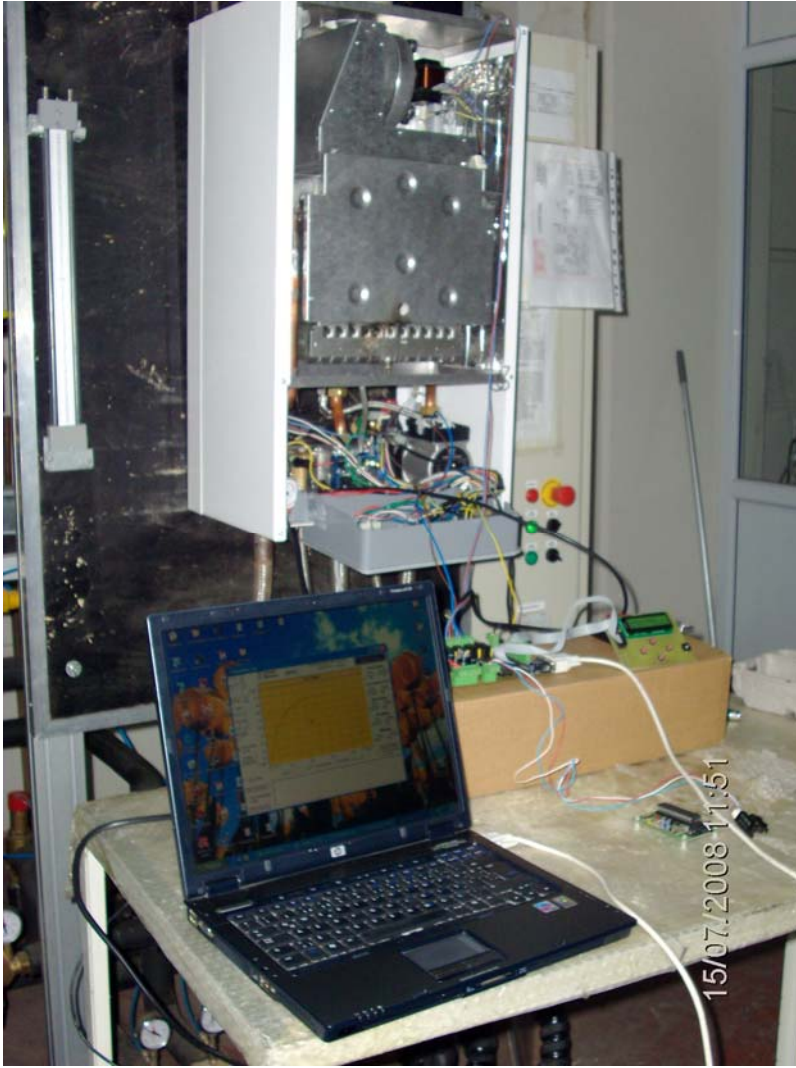
Şekil 7.3. Kombi test laboratuvarı kontrol istasyonu

Şekil 7.4’te tasarlanan elektronik kart ile kombi sistem elemanları arasındaki kablo bağlantıları gösterilmiştir. Açık çevrim fonksiyon cevabının elde edilmesi için fan, gaz valfi besleme gerilimi, gaz valfi modülasyon bobini PWM sinyali, flowmetre sinyal, kapalı çevrim su sirkülasyon pompası ve kombi çıkış suyu sıcaklığını algılamak için NTC sensör kablo bağlantıları yapılmıştır. LCD ekranda, sıcaklık değeri her 1.5 sn’de bir görülebilmektedir. Ayrıca LCD ekranda kapalı çevrim su akış miktarı her saniyede bir litre/dak olarak gözlenebilmektedir. Test çalışmasının başlatılması ve bitirilmesi işlemi için keypad üzerindeki tuşlar kullanılmıştır. Test ortamında elde edilen sıcaklık bilgisinin her 1.5 sn’de bir bilgisayar ortamına gönderilmesi ve elde edilen bilginin grafik haline getirilmesi için RS232 seri port bağlantısı bilgisayar ile elektronik kart arasında kurulmuştur.



Şekil 7.4. Elektronik kart-kombi sistem elemanları kablo bağlantıları

Şekil 7.5'te ve Şekil 7.6'da test sırasında elde edilen sıcaklık bilgisinin bilgisayar ortamında grafik haline getirilmesi gösterilmiştir. Bilgisayar ekranında Stamp Plot Lite Version 7.0 veri izleme programı görülmektedir.

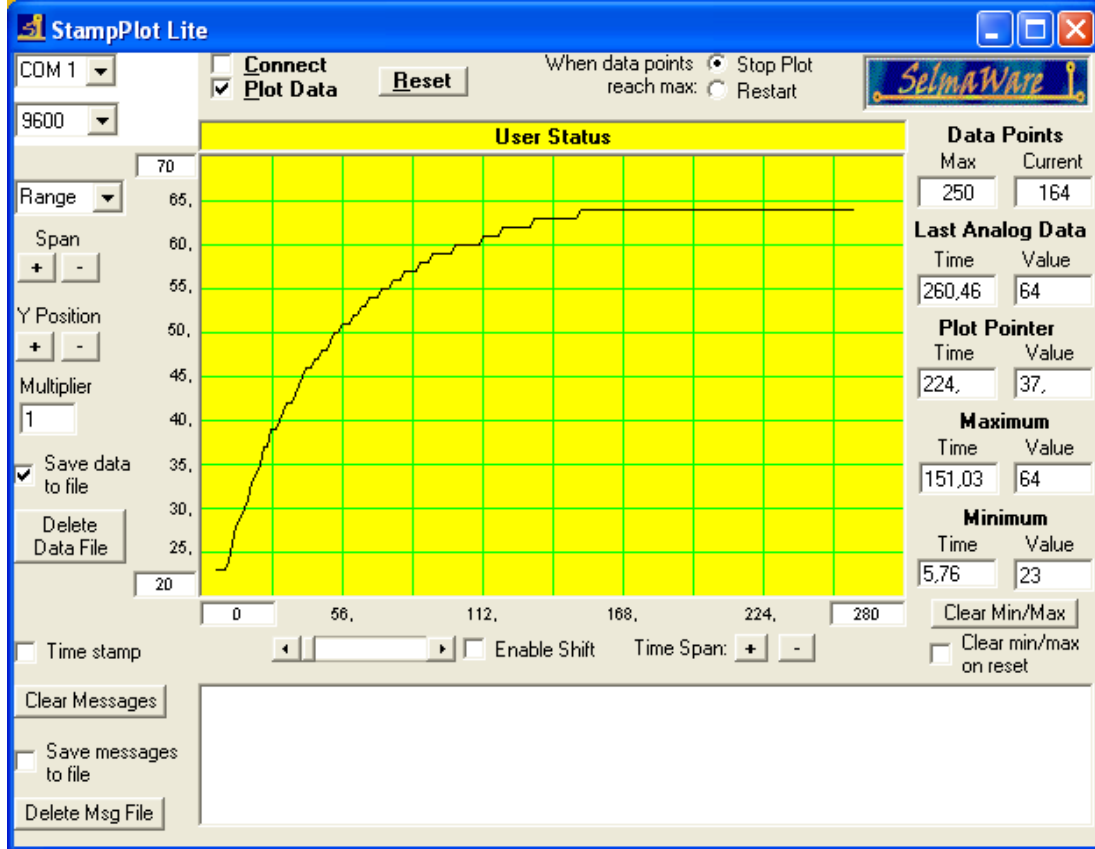


Şekil 7.5. Sistem test çalışmasından elde edilen bilgilerin incelenmesi



Şekil 7.6. Sıcaklık bilgisinin bilgisayar ortamında takip edilmesi ve değerlendirilmesi

Sistem açık çevrim fonksiyon cevabının elde edilmesi için PWM sinyalinin duty cycle'ı %5 ayarlanarak gaz valfi modülasyon bobinine uygulanmıştır ve elde edilen sonuç Şekil 7.7'de grafik halinde gösterilmiştir.



Şekil 7.7. Sistemin açık çevrim transfer fonksiyon cevabı

Şekil 7.7’de sistemin açık çevrim transfer fonksiyon cevabı grafik haline getirilmiştir. Grafikteki bilgiler doğrultusunda [7.1] denklemindeki K , T_d ve T_1 parametreleri kolaylıkla bulunabilir. Bunun için grafikteki gerçek sıcaklık değer bilgilerine ihtiyaç duyulur. Bunun için Stamp Plot Line programı tarafından alınan sıcaklık bilgileri otomatik olarak eşzamanlı olarak bir dosyaya kaydedildiğinden, bu bilgiler derlenip tablo haline getirilmiştir. Elde edilen değerler Tablo 7.1’ de gösterilmiştir. Tablo 7.1’de her sütunda elde edilen değerler sıralı olarak gösterilmiştir ve iki bilgi arasındaki süre 1.5 sn’dir. Örneğin Tablo 7.1’den toplam test süresinin $246\text{sn}(164 \cdot 1.5)$ olduğunu görebiliriz.

Tablo 7.1. Açık çevrim transfer fonksiyon cevabı sıcaklık değerleri

1, 23, 0,	31, 50, 0,	61, 59, 0,	91, 63, 0,	121, 64, 0,	151, 64, 0,
2, 23, 0,	32, 50, 0,	62, 60, 0,	92, 63, 0,	122, 64, 0,	152, 64, 0,
3, 23, 0,	33, 51, 0,	63, 60, 0,	93, 63, 0,	123, 64, 0,	153, 64, 0,
4, 24, 0,	34, 51, 0,	64, 60, 0,	94, 64, 0,	124, 64, 0,	154, 64, 0,
5, 26, 0,	35, 51, 0,	65, 60, 0,	95, 64, 0,	125, 64, 0,	155, 64, 0,
6, 28, 0,	36, 52, 0,	66, 60, 0,	96, 64, 0,	126, 64, 0,	156, 64, 0,
7, 29, 0,	37, 52, 0,	67, 60, 0,	97, 64, 0,	127, 64, 0,	157, 64, 0,
8, 30, 0,	38, 53, 0,	68, 60, 0,	98, 64, 0,	128, 64, 0,	158, 64, 0,
9, 31, 0,	39, 53, 0,	69, 61, 0,	99, 64, 0,	129, 64, 0,	159, 64, 0,
10, 33, 0,	40, 54, 0,	70, 61, 0,	100, 64, 0,	130, 64, 0,	160, 64, 0,
11, 34, 0,	41, 54, 0,	71, 61, 0,	101, 64, 0,	131, 64, 0,	161, 64, 0,
12, 35, 0,	42, 54, 0,	72, 61, 0,	102, 64, 0,	132, 64, 0,	162, 64, 0,
13, 37, 0,	43, 55, 0,	73, 61, 0,	103, 64, 0,	133, 64, 0,	163, 64, 0,
14, 37, 0,	44, 55, 0,	74, 62, 0,	104, 64, 0,	134, 64, 0,	164, 64, 0,
15, 39, 0,	45, 55, 0,	75, 62, 0,	105, 64, 0,	135, 64, 0,	
16, 39, 0,	46, 56, 0,	76, 62, 0,	106, 64, 0,	136, 64, 0,	
17, 40, 0,	47, 56, 0,	77, 62, 0,	107, 64, 0,	137, 64, 0,	
18, 41, 0,	48, 56, 0,	78, 62, 0,	108, 64, 0,	138, 64, 0,	
19, 42, 0,	49, 57, 0,	79, 62, 0,	109, 64, 0,	139, 64, 0,	
20, 42, 0,	50, 57, 0,	80, 62, 0,	110, 64, 0,	140, 64, 0,	
21, 43, 0,	51, 57, 0,	81, 62, 0,	111, 64, 0,	141, 64, 0,	
22, 44, 0,	52, 57, 0,	82, 63, 0,	112, 64, 0,	142, 64, 0,	
23, 45, 0,	53, 58, 0,	83, 63, 0,	113, 64, 0,	143, 64, 0,	
24, 46, 0,	54, 58, 0,	84, 63, 0,	114, 64, 0,	144, 64, 0,	
25, 46, 0,	55, 58, 0,	85, 63, 0,	115, 64, 0,	145, 64, 0,	
26, 47, 0,	56, 59, 0,	86, 63, 0,	116, 64, 0,	146, 64, 0,	
27, 47, 0,	57, 59, 0,	87, 63, 0,	117, 64, 0,	147, 64, 0,	
28, 48, 0,	58, 59, 0,	88, 63, 0,	118, 64, 0,	148, 64, 0,	
29, 48, 0,	59, 59, 0,	89, 63, 0,	119, 64, 0,	149, 64, 0,	
30, 49, 0,	60, 59, 0,	90, 63, 0,	120, 64, 0,	150, 64, 0,	

Tablo 7.1'den açık çevrim fonksiyon parametrelerini aşağıdaki gibi elde edebiliriz:

1-) T_d parametresinin hesaplanması için eğrinin keskin olarak değiştiği 0-8 sütun aralığını kabul edersek;

$$T_d = 8 * 1.5 \text{sn} = 12 \text{ sn}$$

2-) T_1 parametresi için eğrinin sabit hale geldiği 8-62 sütun aralığını kabul edersek;

$$T_1 = (62-8) \times 1,5 = 81 \text{ sn}$$

3-) K parametresini şu formülle gösterirsek;

$$K = (\text{Sıcaklık değişim yüzdesi}) / (\text{PWM duty cycle \% son değer} - \text{PWM duty cycle \% ilk değer})$$

Sıcaklık ilk değeri: 23 °C

Sıcaklık son değeri: 64 °C

Sıcaklık değişim yüzdesi= %178

PWM duty cycle ilk değer % = 0

PWM duty cycle son değer % = 5

Bu durumda;

$$K = 178 / (5-0) = 35,6$$

Bu işlemler sonucunda Ziegler-Nichols açık-çevrim işlemi yaklaşık transfer fonksiyonunu elde ederiz:

$$G(s) = \frac{K \cdot e^{-s.T_d}}{(1 + s.T_1)} = \frac{35,6 \cdot e^{-12s}}{(1 + 81s)}$$

7.3. PID Kontrolör Tasarımı ve Uygulama Sonucu

Açık çevrim transfer fonksiyon cevabında elde edilen parametrelerle, Bölüm 4.5.1 Tablo 4.1'de gösterilen Ziegler-Nichols kontrolör ayar tablosundaki değerler hesaplanarak ayrık zaman PID kontrolör denklemi elde edilir.

Tablo 4.1'de Ziegler-Nichols'un önerdiği ayarlara göre;

$$K_p = \frac{1,2 \cdot T_1}{K \cdot T_d}, \quad T_i = 2 \cdot T_d, \quad T_D = 0,5 \cdot T_d \text{ 'dir.}$$

Bu durumda elde edilen sonuçlar:

$$K_p = \frac{1,2 \cdot T_1}{K \cdot T_d} = \frac{1,2 \cdot 81}{35,6 \cdot 12} = 0,227$$

$$T_i = 2 \cdot T_d = 2 \cdot 12 = 24$$

$$T_D = 0,5 \cdot T_d = 0,5 \cdot 12 = 6$$

Bölüm 4.8'de elde edilen bilgiler doğrultusunda örnekleme zamanı;

$$T_{\text{örnekleme_zamanı}} < T_1/4$$

$$T_{\text{örnekleme_zamanı}} < 81/4 = 20,25 \text{ sn olmalıdır.}$$

Bu koşula uygun olarak örnekleme zamanı $T = 5 \text{ sn}$ seçilmiştir.

Bölüm 4.7'de işlenen bilgiler doğrultusunda [4.18] denklemi;

$$U(z) = E(z) \cdot K_p \cdot \left[1 + \frac{T}{T_i \cdot (1 - z^{-1})} + T_D \cdot \frac{(1 - z^{-1})}{T} \right]$$

$$a = K_p, \quad b = \frac{K_p \cdot T}{T_i}, \quad c = \frac{K_p \cdot T_D}{T}$$

$$U(z)/E(z) = a + \frac{b}{(1 - z^{-1})} + c \cdot (1 - z^{-1})$$

Ziegler-Nichols kontrolör ayar tablosundaki formüller kullanılarak aşağıdaki değerler hesaplanır:

$$a = K_p = 0,227$$

$$b = \frac{K_p \cdot T}{T_i} = \frac{0,227 \cdot 5}{24} = 0,047$$

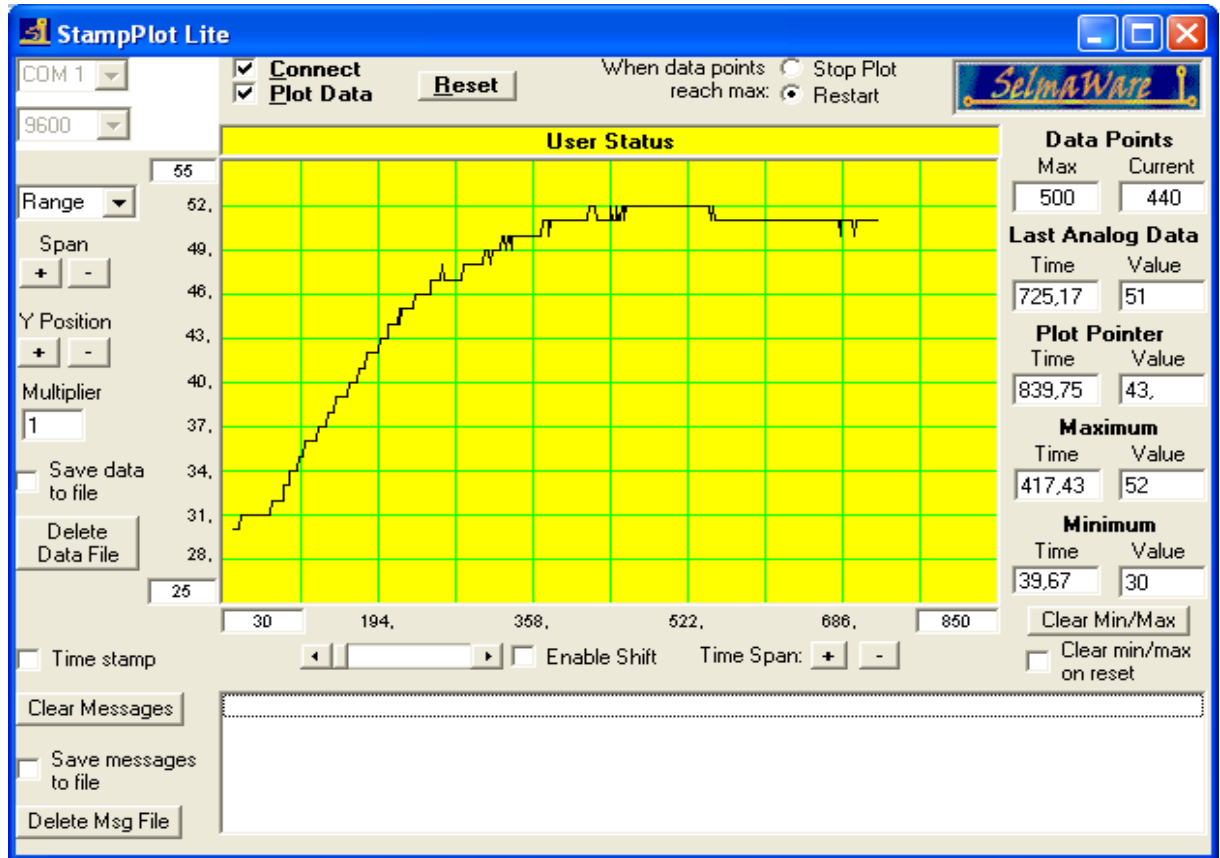
$$c = \frac{K_p \cdot T_D}{T} = \frac{0,227 \cdot 6}{5} = 0,27$$

Ayrık zaman PID kontrolör denklemi:

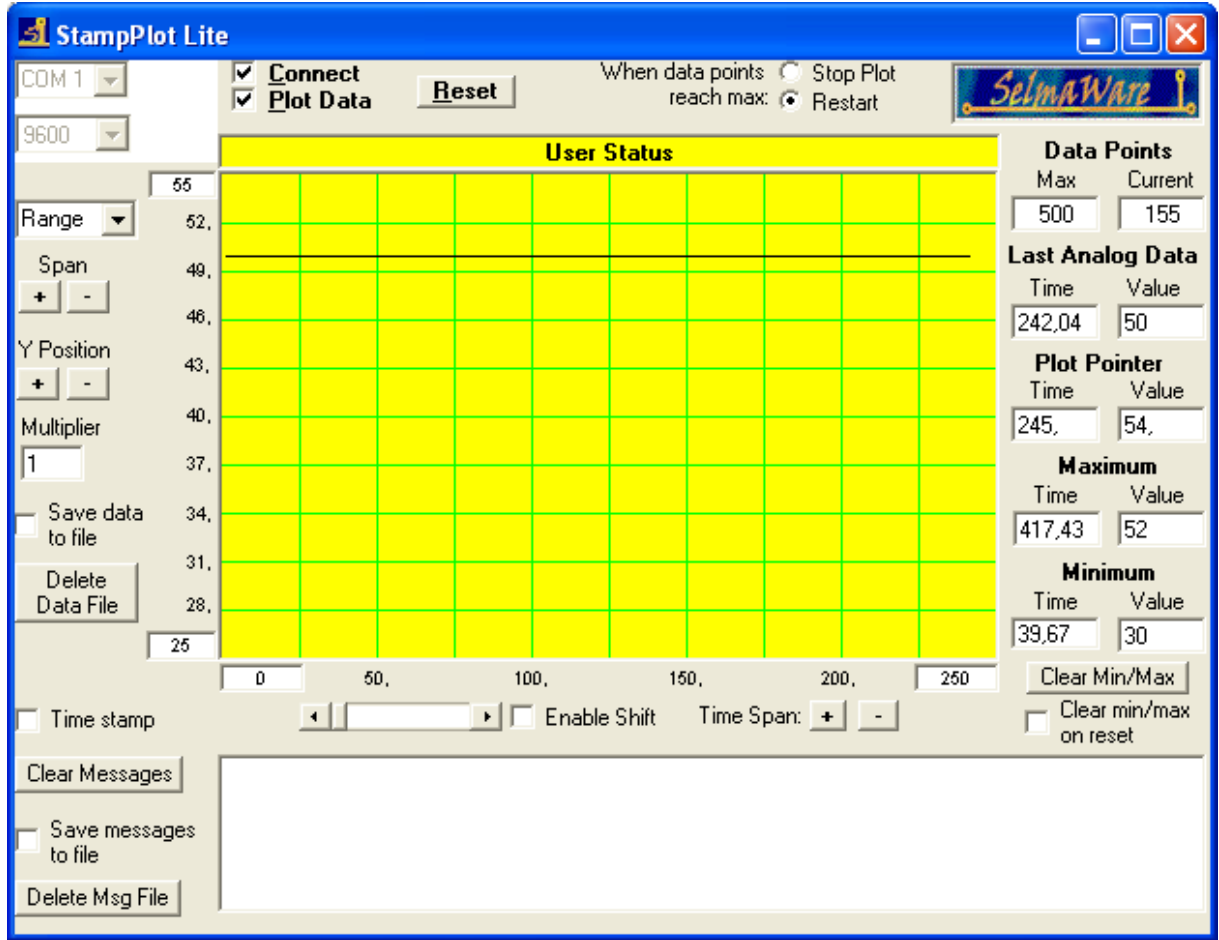
$$U(z)/E(z) = a + \frac{b}{(1-z^{-1})} + c \cdot (1-z^{-1}) = 0,227 + \frac{0,047}{(1-z^{-1})} + 0,27 \cdot (1-z^{-1})$$

elde edilir.

Yukarıda açıklanan tüm bilgiler doğrultusunda ve bölüm 4.7'de açıklanan standart PID algoritması mikrokontrolör tarafından yürütülerek Şekil 7.8 ve Şekil 7.9'daki grafik elde edilmiştir. Sıcaklık set değeri 50 °C olarak ayarlanmıştır.



Şekil 7.8. PID kontrolör sonucu elde edilen proses sıcaklık grafiği



Şekil 7.8. devamı

Şekil 7.8’de elde edilen çalışma sonucuyla, bu tezde tasarlanan donanım ve yazılım ile istenen sistem cevap performansı elde edilmiştir.

Bu çalışmanın bir sonraki aşaması Auto-tuning yöntemlerinden uygun olanın seçilerek çalışmaya ilave edilmesidir. Böylelikle cihaz farklı çalışma ortamları için gerekli olan PID parametrelerini otomatik olarak ayarlayacaktır.

KAYNAKLAR

- [1] Microcontroller and Applications, www.microchip.com, 04.2008
- [2] Temperature Sensors and Applications, www.microchip.com, 05.2008
- [3] KÜÇÜKER, A., NTC Sensörlü Endüstriyel Bir Fırının Mikrodenetleyici İle Kontrolü, 2005
- [4] DOĞAN, İ., Microcontroller Based Temperature Monitoring and Control, Elsevier Science & Technology Books, 2002
- [5] IEEE 754 Compliant Floating Point Routines, www.microchip.com, 06.2008
- [6] Thermistor Temperature Sensor, <http://europe.hbc.honeywell.com>, 06.2008
- [7] TNY278PN, www.powerint.com, 05.2008
- [8] ULN2803A, www.alldatasheet.com, 04.2008
- [9] Turbine Flow Meter With Hall Effect Sensor Cartdrige Version, www.fugas.com, 05.2008
- [10] SARIOĞLU, K., Dijital Kontrol Sistemleri, 2007
- [11] BİR, A., Otomatik Kontrol Sistemleri, 2007

EKLER

Tasarımda kullanılan mikrodenetleyici assembly program

```
.*****
;
;***Author&Designer: SINAN SARIKAYA***
;*****
;////////////////////////////////////
;////////////////////////////////////
;////////////////////////////////////
; PROGRAM DEĞİŞİKLİK TARİHİ: 02.09.2008- 21:00
;////////////////////////////////////
LIST P=16F877A
#include "P16F877A.INC"
__CONFIG _WDT_ON ;"watch dog timer" etkin

;////////////////////////////////////
;////////////////////////////////////

.*****
;***** PIC PIN TANIMLAMALARI *****
;
#define ISI1 PORTA,0 ;ISI SENSÖRÜ ALGILAMA GIRISI1
#define ISI2 PORTA,1 ;ISI SENSÖRÜ ALGILAMA GIRISI1
#define ISI3 PORTA,3 ;LM35A ODA TERMOSTATI
#define FPULSE PORTA,4 ;FLOWMETRE PULSE

#define RB0 PORTB,0
#define RB1 PORTB,1
#define RB2 PORTB,2
#define RB3 PORTB,3
#define RB4 PORTB,4
#define RB5 PORTB,5
#define RB6 PORTB,6
#define RB7 PORTB,7

#define SFIRE PORTB,0 ;ATEŞLEME ALGILAMA
#define _3BAR PORTB,1
#define AGAZ PORTB,2
#define NTC100 PORTB,3

#define RELAY_FAN PORTC,0 ;FAN ON/OFF
#define RELAY_PUMP PORTC,1 ;POMPA ON/OFF
#define PWM PORTC,2 ;PWM MODULATION
#define RELAY_VALF PORTC,3 ;GAZ VALFİ ON/OFF
#define BUZZER PORTC,4
#define LCD_LED PORTC,5
```

```

#DEFINE RELAY_IGNITION PORTD,0 ;ATEŞLEME ON/OFF
#DEFINE LCD_RS PORTD,1
#DEFINE LCD_EN PORTD,2

.*****
;
.***** TANIMLAMALAR *****
;

#DEFINE BANK0 BCF STATUS,5
#DEFINE BANK1 BSF STATUS,5
#DEFINE ZERO STATUS,Z
#DEFINE CARRY STATUS,C
#DEFINE PAGE0 BCF PCLATH,3
#DEFINE PAGE1 BSF PCLATH,3
;-----
#DEFINE FLW FLAG1,0 ;(1/0)0.5/0.0 LT
#DEFINE ACK_KONTROL FLAG1,1 ;(1/0)ARDIŞIL/TEK OKUMA
#DEFINE HATA FLAG1,2 ;(1/0)(RS232 RECEIVE DATA HATA VAR/YOK)
#DEFINE POINT FLAG1,3 ;(1/0)ONLAR/BIRLER BASAMAĞI
#DEFINE LIMIT FLAG1,4 ;(1/0)ÜST/ALT LİMİT
#DEFINE GENF FLAG1,5 ;(1/0)DENEME AMAÇLI FLAG
#DEFINE ZEROFIVE FLAG1,6 ;(1/0) (+0.5C /+0.0 C)
#DEFINE NOISE FLAG1,7 ;(1/0) (GÜRÜLTÜ VAR/YOK)

#DEFINE BUZ FLAG2,0 ;BUZZER ON/OFF(1/0)
#DEFINE OTO_ON FLAG2,1 ;SİSTEM AÇIK/KAPALI(1/0)
#DEFINE MANUEL FLAG2,2 ;MANUEL/OTOMATİK(0/1)

#DEFINE PRGM TUS_FLAG,0
#DEFINE OKEY TUS_FLAG,1
#DEFINE PLUS TUS_FLAG,2
#DEFINE MINUS TUS_FLAG,3 ;HANGI TUS BASILDIYSA DEGER:1
#DEFINE LEFT TUS_FLAG,4
#DEFINE RIGHT TUS_FLAG,5
#DEFINE ONNOFF TUS_FLAG,6

#DEFINE OK_VALID TUS_KONTROL,0 ;(1/0)YÖN TUSLARI GEÇERLİ/GEÇERSİZ

#DEFINE STR_SIRA TUS_KONTROL,1 ;(1/0)2.SATIR
#DEFINE CIFTSTR TUS_KONTROL,2 ;(1/0)ÇİFT SATIR/TEKSATIR
#DEFINE GORUN TUS_KONTROL,3 ;(1/0)GÖRÜN/GÖRÜNME
#DEFINE Y_INVLD TUS_KONTROL,4 ;(1/0)YÖN TUSLARI TARAMA DISINDA

#DEFINE TUS_ON TUS_KONTROL,7 ;(1/0)TUS TAKIMINA GIRILDIGINI
;GÖSTERİR

.*****
;
.*****EQU ESITLIKLERI*****
;
.*****
;
.*****GENEL SAYACLAR*****
;

SAYAC0 EQU 20H
SAYAC1 EQU 21H
SAYAC2 EQU 22H
SAYAC3 EQU 23H
SAYAC4 EQU 24H
SAYAC5 EQU 25H
SAYACX EQU 26H

```

```

SAYACA          EQU   27H
SAYACB          EQU   28H

;*****GENEL FLAGLER*****

FLAG1          EQU   29H
;FLAG2          EQU   2AH
FLAG3          EQU   2BH
TUS_FLAG       EQU   2CH
TUS_KONTROL    EQU   2DH
;*****
KANAL          EQU   2EH   ;NTC
TEMPH         EQU   2FH
TEMPL         EQU   30H
POINTER       EQU   31H
;***** INTERRUPT *****
INT_SAYAC1    EQU   32H
INT_SAYAC2    EQU   33H
;*****

LCD_DATA      EQU   34H   ;2
UZUNLUK      EQU   35H

CHR_POINT     EQU   36H   ;3
WORD_POINT   EQU   37H

W_TEMP       EQU   38H   ;4
FSR_TEMP     EQU   39H
STATUS_TEMP  EQU   3AH
PCLATH_TEMP  EQU   3BH

RAKAM_HEX    EQU   3CH   ;5
KARAKTER     EQU   3DH
SINIR        EQU   3EH   ;ÜST SINIR
SINIR_SAYAC  EQU   3FH   ;ALT SINIR

BIN          EQU   40H
HUNDRED      EQU   41H
TENS_ONES    EQU   42H

TOP_L        EQU   43H
TOP_H        EQU   44H
RAKAM        EQU   45H
YEDEK        EQU   46H
TEMPX        EQU   47H

ALT_LIMIT    EQU   48H
UST_LIMIT    EQU   49H

DIGIT1       EQU   4AH
DIGIT2       EQU   4BH
DIGIT3       EQU   4CH

PAGEADRES    EQU   4DH
LCD_ADRES    EQU   4EH
RC_TEMP      EQU   4FH

SYC1         EQU   50H
TEMPF        EQU   51H

```

```
TMR1_INT    EQU    52H
SAYACY      EQU    53H
```

```
*****
;
FLAG2       EQU    7DH
E_ADR       EQU    7EH ;EEPROM ADRES
E_DATA      EQU    7FH ;EEPROM DATA
```

```
*****MACROLAR*****
```

```
BANK_0      MACRO
    BCF     STATUS,5
    BCF     STATUS,6
ENDM
```

```
BANK2       MACRO
    BCF     STATUS,5
    BSF     STATUS,6
ENDM
```

```
BANK3       MACRO
    BSF     STATUS,5
    BSF     STATUS,6
ENDM
```

```
*****
```

```
; BANK2
; GENERAL MATH LIBRARY DEFINITIONS
; 20H-31H RAM ALANI(18 BYTE)
; general literal constants
; define assembler constants
B0 equ 0
B1 equ 1
B2 equ 2
B3 equ 3
B4 equ 4
B5 equ 5
B6 equ 6
B7 equ 7
MSB equ 7
LSB equ 0
; define commonly used bits
; STATUS bit definitions
#define _C STATUS,0
#define _Z STATUS,2
;
; general register variables
;
```

```
ACCB7 equ 0x20
ACCB6 equ 0x21
ACCB5 equ 0x22
ACCB4 equ 0x23
ACCB3 equ 0x24
ACCB2 equ 0x25
ACCB1 equ 0x26
```

```

ACCB0 equ 0x27
ACC  equ 0x27      ; most significant byte of contiguous 8 byte accumulator
;
SIGN  equ 0x29     ; save location for sign in MSB
;
TEMPB1 equ 0x30
TEMP  equ 0x31     ; temporary storage
;

; binary operation arguments
;
AARGB7 equ 0x20
AARGB6 equ 0x21
AARGB5 equ 0x22
AARGB4 equ 0x23
AARGB3 equ 0x24
AARGB2 equ 0x25
AARGB1 equ 0x26
AARGB0 equ 0x27
AARG  equ 0x27     ; most significant byte of argument A
;
BARGB3 equ 0x2B
BARGB2 equ 0x2C
BARGB1 equ 0x2D
BARGB0 equ 0x2E
BARG  equ 0x2E     ; most significant byte of argument B
;
; Note that AARG and ACC reference the same storage location
;
;*****
;
;
; FLOATING POINT SPECIFIC DEFINITIONS
;
; literal constants
;
EXPBIAS equ 7fh
;
; biased exponents
;
EXP equ 0x28       ; 8 bit biased exponent
AEXP equ 0x28      ; 8 bit biased exponent for argument A
BEXP equ 0x2F      ; 8 bit biased exponent for argument B
;
; floating point library exception flags
;
FPFLAGS equ 0x2A; floating point library exception flags
IOV equ 0          ; bit0 = integer overflow flag
FOV equ 1          ; bit1 = floating point overflow flag
FUN equ 2          ; bit2 = floating point underflow flag
FDZ equ 3          ; bit3 = floating point divide by zero flag
NAN equ 4          ; bit4 = not-a-number exception flag
DOM equ 5          ; bit5 = domain error exception flag
RND equ 6          ; bit6 = floating point rounding flag, 0 = truncation
; 1 = unbiased rounding to nearest LSB
SAT equ 7          ; bit7 = floating point saturate flag, 0 = terminate on
; exception without saturation, 1 = terminate on
; exception with saturation to appropriate value

;*****
;

```

```

.*****
;
.*****
;
;PID YAZILIMINDA KULANILAN ÖZEL ALANLAR
SETP EQU 7AH
TMPR EQU 7BH
COUNTER EQU 7CH
#DEFINE _MINUS FLAG2,3
#DEFINE PID FLAG2,4 ;(1/0) PID ON/OFF
;-----
;PID FLOAT SABİTLER

MAX EQU 0FFH
MIN EQU 00H

;a=Kp = 0,227= 0x7C6872
;b=Kp*T/Ti = 0,047 =0x7A4083
;c=Kp*TD/T = 0,27 =0x7D0A3D

A_EXP equ 7Ch
A_B0 equ 68h
A_B1 equ 72h ;SABİT float a

B_EXP equ 7Ah
B_B0 equ 40h
B_B1 equ 83h ;SABİT float b

C_EXP equ 7Dh
C_B0 equ 0Ah
C_B1 equ 3Dh ;SABİT float c

.*****
;
;PID FLOAT HAFIZA BİRİMLERİ
;BANK2 RAM ALANI

Y_EXP equ 40h
Y_B0 equ 41h
Y_B1 equ 42h ;BANK2 float Y (GERÇEK sıcaklık değeri)

E_EXP equ 43h
E_B0 equ 44h
E_B1 equ 45h ;BANK2 float e (sıcaklık hata değeri)

E_1_EXP equ 46h
E_1_B0 equ 47h
E_1_B1 equ 48h ;BANK2 float E_1 (ÖNCEKİ sıcaklık hata değeri)

P_EXP equ 49h
P_B0 equ 4Ah
P_B1 equ 4Bh ;BANK2 float n (Propertional)-ORANSAL

I_EXP equ 4Ch
I_B0 equ 4Dh
I_B1 equ 4Eh ;BANK2 float p (İntegral) - İNTEGRAL

I_1_EXP equ 4Fh
I_1_B0 equ 50h

```



```

I_1_B1 equ    51h           ;BANK2 p-1 (önceki integral)

D_EXP equ    52h
D_B0 equ    53h
D_B1 equ    54h           ;BANK2 float q (Derivative) - TÜREV

U_EXP equ    55h
U_B0 equ    56h
U_B1 equ    57h           ;BANK2 float U (PID sonucu) - PID SONUCU

```

```

.*****
,

```

```

ORG 000H
GOTO MAIN

```

```

ORG 004H

```

```

MOVWF      W_TEMP
SWAPF     STATUS,W
CLRF     STATUS           ;< IRP,RP1 ve RP0 > sıfır ve Z de 1 olsun
MOVWF     STATUS_TEMP
MOVF     PCLATH,W
MOVWF     PCLATH_TEMP
MOVF     FSR,W
MOVWF     FSR_TEMP
GOTO     INTERRUPT

```

```

.*****
,
.*****
,
.***** LCD ALT PROGRAMLARI *
,
.*****
,

```

```

; GIRIS : W (DATA/ADRES)
,
-----

```

```

LCD_GONDER
  BCF     LCD_RS           ;KOMUT MODU
  MOVWF   LCD_DATA

  MOVLW   0FH
  ANDWF   PORTD,F         ;ilk 4-bit (LSb'ler) sıfır
  MOVF   LCD_DATA,W ;
  ANDLW   0F0H           ;son 4-bit (MSb'ler) sıfır
  IORWF   PORTD,F         ;DÜŞÜK BYTE KORUNDU

```

```

BSF     LCD_EN           ;ENABLE
NOP
BCF     LCD_EN           ;DISABLE

```

```

MOVLW   0FH
ANDWF   PORTD,F         ;ilk 4-bit (LSb'ler) sıfır
SWAPF   LCD_DATA,W

```

```

        ANDLW      0F0H      ;son 4-bit (MSb'ler) sıfır
        IORWF PORTD,F
                                ;---SWAPF      LCD_DATA,W

        BSF      LCD_EN      ;ENABLE
        NOP
                                ;DÜSÜK BYTE GÖNDERİLDİ
        BCF      LCD_EN      ;DISABLE

GECIKME_40US

        CLRWDT
        MOVLW     12H        ;~50us
        MOVWF     SAYAC3
BEKLE0
        DECFSZ    SAYAC3,F
        GOTO     BEKLE0
        RETURN
;*****
;
; GIRIS : W (DATA/ADRES)
;-----
LCD_PUTC
        BSF      LCD_RS      ;DATA MODU
        GOTO     LCD_GONDER+0x1
;*****
;

CLEAR_DISP

        MOVLW     01H        ;DISPLAY SILME KOMUTU
        CALL     LCD_GONDER
        CALL     GECIKME_1_64MS ;WDT <- 0
        RETURN
;*****
;
;STR2'YI TEMİZLER, CURSOR'U STR2'NIN BASINA GETİRİR VE
;LCD'YI DATA GIRISINE HAZIR HALE GETİRİR (LCD_RS=1)

CLEAR_STR2
        MOVLW     10H        ;2.SATIR UZUNLUĞU d'16'
        MOVWF     SAYAC5

        MOVLW     0xC0       ;2.SATIR BASLANGIC ADRESI YUKLENDİ
        CALL     LCD_GONDER

STR2SIL
        MOVLW     20H        ;BOSLUK KARAKTERI ' '
        CALL     LCD_PUTC
        DECFSZ    SAYAC5,F
        GOTO     STR2SIL

        MOVLW     0xC0       ;2.SATIR BASLANGIC ADRESI YUKLENDİ
        CALL     LCD_GONDER
        RETURN
;*****
;
;*** imleci bir sola kaydırır

CrShiftLeft
        MOVLW     10H        ;IMLECI SOLA KAYDIR
        CALL     LCD_GONDER
        RETURN
;*****
;

```

```
;*** imleci bir saga kaydırır
```

```
CrShiftRight
```

```
    MOVLW    14H           ;IMLECI SAGA KAYDIR
    CALL    LCD_GONDER
```

```
    RETURN
```

```
.*****
```

```
;*** imleci str2'nin basına alır
```

```
CrGotoRow2
```

```
    MOVLW    0C0H         ;STR2 BAS. ADRESI
    CALL    LCD_GONDER
```

```
    RETURN
```

```
.*****
```

```
blink_ON
```

```
    MOVLW    0DH           ;blink on, cursor off
    CALL    LCD_GONDER
    RETURN
```

```
.*****
```

```
blink_OFF
```

```
    MOVLW    0CH           ;blink off, cursor off
    CALL    LCD_GONDER
    RETURN
```

```
.*****
```

```
;*** LCD'YE BIR KELIME YAZAN ALTPROG. ***
```

```
;*** ( RAM' DEN LCD'YE) ***
```

```
;*** GIRIS : LCD_ADR, FSR ***
```

```
.*****
```

```
wordRAM2LCD
```

```
    MOVF    INDF,W         ;ILK BELLEK HUCRESI UZUNLUK BILGISINI
;ICERIR
    MOVWF   UZUNLUK        ;UZUNLUK <- UZUNLUK
```

```
word_RAMLOOP           ;DONGU
```

```
    INCF   FSR,F
    MOVF   INDF,W
    CALL   LCD_PUTC
    DECFSZ UZUNLUK,F
    GOTO   word_RAMLOOP
    RETURN
```

```
;DONUŞTE FSR YAZILAN EN SON KARAKTERE
```

```
IŞARET EDİYOR
```

```
.*****
```

```
;*** LCD'YE BIR KELIME YAZAN ALTPROG. ***
```

```
;*** ( ROM' DAN LCD'YE) ***
```

```
;*** GIRIS : LCD_ADR, WORD_POINT ***
```

```
.*****
```

```
wordROM2LCD
```

```
    CLRF   CHR_POINT
```

```

PAGESEL    CHR_TABLE
CALL  CHR_TABLE   ;ILK BELLEK HUCRESI UZUNLUK BILGISINI ICERIR
CLRF  PCLATH
MOVWF   UZUNLUK           ;UZUNLUK <- UZUNLUK

word_ROMLOOP           ;DONGU
INCF  CHR_POINT,F
PAGESEL    CHR_TABLE
CALL  CHR_TABLE
CLRF  PCLATH
CALL  LCD_PUTC
DECFSZ   UZUNLUK,F
GOTO  word_ROMLOOP
RETURN

.*****
;
;*** LCD2'nin 2.satirinin ilk 8
;*** karakterini siler
;*** (0xC0'dan 0xC7'ye kadar)

CLR_C0TOC7
    MOVLW    0x08           ;silenecek kısmın UZUNLUĞU d'8'
    MOVWF   SAYAC5

    MOVLW    0xC0           ;2.SATIR BASLANGIC ADRESI YUKLENDI
    CALL  LCD_GONDER

C0TOC7_SIL
    MOVLW    0x20           ;BOSLUK KARAKTERI ' '
    CALL  LCD_PUTC
    DECFSZ   SAYAC5,F
    GOTO  C0TOC7_SIL
;-----

    MOVLW    0xC0           ;2.SATIR BASLANGIC ADRESI YUKLENDI
    CALL  LCD_GONDER

RETURN

.*****
;
;*****
;
CGRAM_TABLE
    MOVLW    HIGH (CGRAM_TABLE+0X04)
    MOVWF   PCLATH
    MOVF    SAYAC0,W
    SUBLW   20H           ;w = 20H - SAYAC0 [4x8=32 (=20h), 4 KARAKTER ]
    ADDWF   PCL,F       ;SIRALAMA ç->00, ğ-> 01, ş->02, THICK->03

    RETLW0EH           ;Ç ->00h
    RETLW11H
    RETLW10H
    RETLW10H
    RETLW11H
    RETLW0EH
    RETLW04H
    RETLW00H
;-----
    RETLW0EH           ;Ğ ->01h
    RETLW00H
    RETLW0FH
    RETLW10H

```

```

RETLW17H
RETLW11H
RETLW0FH
RETLW00H
;-----
RETLW0FH           ;§ ->02h
RETLW10H
RETLW0EH
RETLW01H
RETLW1EH
RETLW00H
RETLW04H
RETLW00H
;-----
RETLW00H           ;THICK ->03h
RETLW00H
RETLW01H
RETLW02H
RETLW14H
RETLW08H
RETLW00H
RETLW00H
;-----
.*****
;
PREPARE_LCD

;   BCF   LCD_RW           ;YAZ (DONANIM)
LCD
   BCF   LCD_EN           ;ISLEM YOK
   BCF   LCD_RS           ;KOMUT GIRISI MODU
   CALL  GECIKME_1_64MS

   MOVLW   28H           ;FUNCTION SET(40US), (4-bit, 2 satir, 5X7 nokta)
   CALL  LCD_GONDER
   CALL  GECIKME_1_64MS

   CALL  CLEAR_DISP      ;AYNI ZAMANDA CURSOR HOME OLUR(1.64MS)
   MOVLW   06H           ;ENTRY MOD SET(40US), INCREMENT CURSOR
;(saga kaydir)
   CALL  LCD_GONDER
   MOVLW   0CH           ;DISPLAY ON/OFF KONTROL(40US), (display on,
;cursor off)
   CALL  LCD_GONDER

;.....[-- CGRAM HAZIRLIK KOMUTLARI --].....

   MOVLW   20H           ;TOPLAM KARAKTER SAYISI
   MOVWF  SAYAC0

WRITE_CGRAM
   MOVF  SAYAC0,W
   SUBLW  20H           ;W = 20H - W
   ADDLW  40H           ;CGRAM BASLANGIC ADRESI TOPLAM 64 BYTE

   CALL  LCD_GONDER
   CALL  CGRAM_TABLE

```

```

CALL LCD_PUTC
CALL GECIKME_40US
CALL GECIKME_40US
CALL GECIKME_40US ;CGRAM E DATA YAZMA 120US
DECFSZ SAYAC0,F
GOTO WRITE_CGRAM

CALL GECIKME_1_64MS
CALL CLEAR_DISP ;AYNI ZAMANDA CURSOR HOME OLUR(1.64MS)
;CALL GECIKME_1_64MS
RETURN

.*****
;
GECIKME_1_64MS
CLRWDI
MOVLW 30h ;~7,1ms
MOVWF SAYAC3
BEKLE2
MOVLW 30h
MOVWF SAYAC4
BEKLE1
DECFSZ SAYAC4,F
GOTO BEKLE1
DECFSZ SAYAC3,F
GOTO BEKLE2
RETURN

.*****
;
;*** GIRIS : FSR
;*** INDF'DEN ALDIGI 1 BYTE'LIK BCD DEGERI
;*** 2 DIGIT SEKLINDE LCD'YE GINDERIR.
;-----
;_PRINTF
MOVWF FSR
BCF STATUS,IRP ;BANK0
SWAPF INDF,F
MOVLW 0FH
ANDWF INDF,W ;MSD, W'DE
ADDLW 30H ;RAKAM -> ASCII
CALL LCD_PUTC ;1.DIGIT (MSD) LCD'YE

SWAPF INDF,F
MOVLW 0FH
ANDWF INDF,W ;LSD, W'DE
ADDLW 30H ;RAKAM -> ASCII
CALL LCD_PUTC ;2.DIGIT (LSD) LCD'YE
;-----

RETURN

.*****
;
;*** giriş : WORD_POINT, CHR_POINT, FLAGS
;*** çıkış : -
;*** çağırım : wordROM2LCD, CrShiftLeft, CrGotoRow2,
;*** CHR_TABLE, PANEL_GONDER, PANELDEN_DATA, LCD_GONDER
;*** değişen : -
;*** amaç :daha kısa ve anlaşılır bir alt prog.
.*****
;
TABLO
CLRF CHR_POINT

```



```

*****
;
;*** İŞLEM: 0..255 HEX -> DECİMAL ÇEVİRME İŞİ YAPAR
;*** GIRIS :W
;*** CIKIS : HUNDRED, TENS_ONES
;-----
HEXTOBCD
    MOVWF    BIN

    CLRF    HUNDRED
    SWAPF   BIN,W
    ADDWF   BIN,W
    ANDLW   0FH
    BTFSC   STATUS,DC
    ADDLW   16H
    BTFSC   STATUS,DC
    ADDLW   06H
    ADDLW   06H
    BTFSS   STATUS,DC
    ADDLW   -06H
    BTFSC   BIN,4
    ADDLW   16H-1+06H
    BTFSS   STATUS,DC
    ADDLW   -06H

    BTFSC   BIN,5
    ADDLW   30H
    BTFSC   BIN,6
    ADDLW   60H
    BTFSC   BIN,7
    ADDLW   20H
    ADDLW   60H
    RLF     HUNDRED,F
    BTFSS   HUNDRED,0
    ADDLW   -60H
    MOVWF   TENS_ONES
    BTFSC   BIN,7
    INCF    HUNDRED,F

    RETURN
*****
;
;*****
;*** İŞLEM: 0..255 BCD/ASCII -> HEX ÇEVİRME İŞİ YAPAR
;*** GIRIS :DIGIT1/2/3
;*** CIKIS : BIN/W
;-----
ASCII2HEX
    MOVLW   0X30
    SUBWF   DIGIT1, F    ;YÜZLER BASAMAĞI
    SUBWF   DIGIT2, F    ;ONLAR BASAMAĞI
    SUBWF   DIGIT3, F    ;BİRLER BASAMAĞI
;-----
;-----
BCD2HEX
_DIGI1
    MOVF    DIGIT1,W
    BTFSC   ZERO
    GOTO    X0
    MOVF    DIGIT1, W
    SUBLW   0X02

```



```

CLRF TUS_FLAG          ;BÜTÜN BAYRAKLAR SIFIR
BTFS GORUN             ;karakter LCD'de gösterilecek mi?
GOTO $+3               ;hayır,

;-----
MOVLW 0DH              ;cursor off, blink on
CALL LCD_GONDER

MOVLW 9AH              ;154*65msn=10 sn
MOVWF INT_SAYAC2      ;TUŞ TAKIMI TİMER
CALL SET_65msn
;=====
TARA
CLRWDI
BTFS RB7
GOTO _ONAY
BTFS RB6
GOTO _PRGM
BTFS RB5
GOTO OK_YUKARI        ;plus
BTFS RB4
GOTO OK_ASAGI         ;minus

;-----
BTFS PIR1,TMR2IF
GOTO TARA
DECFSZ INT_SAYAC2,F
GOTO TARA-1
GOTO $                 ;WDT RESET
;=====

OK_YUKARI

CALL CHECK_NOISE
BTFS NOISE             ;GÜRÜLTÜ VAR MI?
GOTO TARA              ;EVET

MOVLW 03H
MOVWF KARAKTER
BTFS Y_INVLD          ;YÖN TUSLARI TARAMA DISINDAMI
GOTO TARA              ;EVET

BSF PLUS               ;OK_YUKARI'YA BASILDI
BTFS OK_VALID         ;OKLAR SADECE TABLO DIŞINDA MI AKTIF?
GOTO OK_YUKARI_CIK    ;EVET, çık

DECFSZ SINIR_SAYAC,F  ;SINIR_SAYAC ONCEDEN YUKLU
GOTO $+0x4
MOVF SINIR,W
MOVWF SINIR_SAYAC
ADDWF WORD_POINT,F

DECF WORD_POINT,F
BSF STR_SIRA          ;SIRA 2. SATIRDA (SADECE 2.SATIR YAZILACAK)
CALL TABLO
CALL blink_OFF

OK_YUKARI_CIK

```

```
CALL TUSSES_VER
GOTO TUSTARA_CIK
```

```
;-----
OK_ASAGI
```

```
CALL CHECK_NOISE
BTFSC NOISE      ;GÜRÜLTÜ VAR MI?
GOTO TARA       ;EVET

MOVLW 04H
MOVWF KARAKTER
BTFSCY_INVLD    ;YÖN TUSLARI TARAMA DISINDAMI(SADECE _SCANFDE
                ; KULLANILİYOR)
GOTO TARA       ;EVET

BSF  MINUS      ;OK_ASAGI'YA BASILDI
BTFSS OK_VALID  ;OKLAR SADECE TABLO DIŞINDA MI AKTIF?
GOTO OK_ASAGI_CIK ;EVET

MOVF SINIR,W
SUBWF SINIR_SAYAC,W
BTFSS CARRY     ;SINIR_SAYAC >= SINIR?    (C=1?)
GOTO $+0x4      ;hayır
CLRF SINIR_SAYAC ;evet
MOVF SINIR,W
SUBWF WORD_POINT,F

INCF WORD_POINT,F
INCF SINIR_SAYAC,F ;SUREKLI ASAGI TUSA BASILMA DURUMUNDA ;
                  ;SINIR_SAYAC ARTTIRILMASIN
BSF  STR_SIRA   ;SIRA 2. SATIRDA (SADECE 2.SATIR YAZILACAK)
CALL TABLO
CALL blink_OFF
```

```
OK_ASAGI_CIK
CALL TUSSES_VER
GOTO TUSTARA_CIK
```

```
;-----
_ONAY
```

```
CALL CHECK_NOISE
BTFSC NOISE      ;GÜRÜLTÜ VAR MI?
GOTO TARA       ;EVET

MOVLW 01H
MOVWF KARAKTER
BSF  OKEY

CALL TUSSES_VER
GOTO TUSTARA_CIK
```

```
;-----
_PRGM
```

```
CALL CHECK_NOISE
BTFSC NOISE      ;GÜRÜLTÜ VAR MI?
```

```

GOTO TARA ;EVET

MOVLW 02H
MOVWF KARAKTER
BSF PRGM

CALL TUSSES_VER
GOTO TUSTARA_CIK

;=====
TUSTARA_CIK
BIRAK
    CLRWDT
    BTFSC GORUN ;ZAMANA BAĞLI DEĞİŞİM MI?
    GOTO TUSZAMAN
OUT_KEY
    CALL DELAY_10msn
    MOVF PORTB,W
    ANDLW 0F0H
    XORLW 0F0H
    BTFSS ZERO
    GOTO OUT_KEY
    RETURN ;hayır, çık

;=====
;GİRİŞ :TEMP(HEX DEĞER)
;ÇIKIŞ :LCDADDRESS,

TUSZAMAN

    BTFSC PLUS ;+ TUŞU MU BASILDI?
    GOTO LOOPKEY ;EVET
    BTFSS MINUS ;- TUŞU MU BASILDI?
    GOTO OUT_KEY ;HAYIR, PRGM/ONAY TUŞU

;0.5SN'DE 1 DEĞERLER DEĞİŞTİRİLEREK GÖSTERİLECEK
;*****

LOOPKEY

    CLRF INT_SAYAC1
    CLRF INT_SAYAC2 ;ZAMANLAYICI SIFIR

    CALL CHANGE_VALUE

LOOPKEYX

    CALL DELAY_10msn

    INCF INT_SAYAC2,F
    MOVF INT_SAYAC2,W
    XORLW 32H ;0.5 SN DOLDU MU?
    BTFSS ZERO
    GOTO CHECK_KEY ;HAYIR

    CLRF INT_SAYAC2
    INCF INT_SAYAC1,F ;TAŞMA KONTROLÜ OLMALI!!!
    CALL CHANGE_VALUE

```

CHECK_KEY

```

MOVFB PORTB,W
ANDLW    0F0H
XORLW    0F0H
BTFSS ZERO
GOTO LOOPKEYX
CALL DELAY_10msn      ;PARAZİT Mi?
MOVFB PORTB,W
ANDLW    0F0H
XORLW    0F0H
BTFSS ZERO
GOTO LOOPKEYX      ;EVET
RETURN          ;hayır, çık

```

```

;-----
;-----
;-----

```

```

;ARTIS DEGERLERİ:

```

```

;0.00-0.99 SN      :1
;1.00-1.499 SN    :2
;1.50- 2.499 SN   :3
; >=2.5SN         :5

```

```

;-----
;-----
;-----
CHANGE_VALUE

```

```

CLRF  DIGIT1
MOVFB TEMPX,W
ANDLW 0FH
MOVWF DIGIT3
SWAPF TEMPX,W
ANDLW 0FH
MOVWF DIGIT2
CALL  BCD2HEX
MOVWF TEMPX

```

```

MOVLW    02H
SUBWF    INT_SAYAC1,W      ;0.00-0.99 SN 'DEN ÇOK BASILDI MI?

```

```

BTFSC CARRY
GOTO DEGİSİM_2      ;EVET

```

```

DEĞİSİM_1

```

```

MOVLW    01H
GOTO DEGİSİM_X

```

```

DEĞİSİM_2

```

```

MOVLW    03H
SUBWF    INT_SAYAC1,W      ;1-1.49 SN 'DEN ÇOK BASILDI MI?

```

```

BTFSC CARRY
GOTO DEGİSİM_3      ;EVET

```

```

MOVLW    02H
GOTO DEGİSİM_X

```

```

DEĞİSİM_3

```

```

MOVLW    05H
SUBWF    INT_SAYAC1,W      ;1.50-2.49 SN 'DEN ÇOK BASILDI MI?

```

```

BTFSC CARRY

```

```

GOTO SONDEGISIM ;EVET

MOVLW    03H
GOTO DEGISIMX

SONDEGISIM

    MOVLW    05H

;.....

DEGISIMX

    BTFSS PLUS           ;+TUŞUNA MI BASILDI?
    SUBWF    TEMPX,F     ;HAYIR, - TUŞUNA BASILDI ÇIKARMA İŞLEMİ
    BTFSC PLUS
    ADDWF    TEMPX,F     ;EVET + TUŞUNA BASILDI, TOPLAMA İŞLEMİ
    BTFSC PLUS
    GOTO    $+3

    BTFSS CARRY          ;ÇIKARMA İŞLEMİ Mİ YAPILDI?
    GOTO    MINVALUE    ;EVET, ÇIKARMA İŞLEMİ YAPILDI SONUÇ NEGATİF

    MOVLW    64H         ;SONUÇ POZİTİF
    SUBWF    TEMPX,W
    BTFSC CARRY
    GOTO    MAXVALUE    ;SONUÇ 99'DAN BÜYÜK
    GOTO    SHOW_LCD

MINVALUE

    MOVLW    00H         ;MİNİMUM DEĞER
    MOVWF    TEMPX
    GOTO    SHOW_LCD

MAXVALUE

    MOVLW    63H         ;MAKSİMUM DEĞER
    MOVWF    TEMPX

;=====
SHOW_LCD

    MOVF    TEMPX,W
    CALL    HEXTOBCD     ;HEX-> BCD ÇEVİRİMİ
    MOVF    TENS_ONES,W
    MOVWF    TEMPX      ;ONLAR BASAMAĞI BCD DEĞER
    MOVLW    TEMPX
    CALL    _PRINTF     ;DATAYI LCD'DE GÖSTER
    MOVF    LCD_ADRES,W
    CALL    LCD_GONDER  ;LCD BAŞLANGIÇ ADRESİ GÖNDER
                        ;blink_ON

    RETURN

.*****
;
.*****
;
.*****
;
;Parazit sinyal kontrolü yapılıyor
CHECK_NOISE

```

```

MOVF PORTB,W
ANDLW      0F0H
XORLW      0F0H
BTFSC ZERO
GOTO P_NOISE

CALL DELAY_10msn

MOVF PORTB,W
ANDLW      0F0H
XORLW      0F0H

BCF NOISE
BTFSC ZERO
P_NOISE
BSF NOISE          ;GÜRÜLTÜ VAR
RETURN

.*****
;
;*****TUS TAKIMI HAZIRLIK KOMUTLARI*
;
;*****
TUSSES_VER
CLRWDT
BSF BUZZER          ;BUZZER ON
;-----
GECIK_158ms        ;~158 ms'lik (SAYAC03 = 0CFH için) GECIKME

MOVLW      0AFH
MOVWF      SAYACX
MOVLW      0FFH
WAIT2
MOVWF      SAYAC4
WAIT1
DECFSZ     SAYAC4,F
GOTO WAIT1
DECFSZ     SAYACX,F
GOTO WAIT2
;-----
BCF BUZZER          ;BUZZER OFF
RETURN

.*****
;
UYARI_SES
MOVLW      20H          ;320ms boyunca, BUZZER 10ms'lik periyot
;... ile çalacak
MOVWF      INT_SAYAC1
UYARI_LOOP
BSF BUZZER          ;BUZZER ON
CALL DELAY_5msn     ;WDT <- 0
BCF BUZZER          ;BUZZER OFF
CALL DELAY_5msn     ;WDT <- 0
DECFSZ     INT_SAYAC1,F
GOTO UYARI_LOOP
RETURN

.*****
;
;*****
;*** BOLGELERDE KOD GONDERMEDEN ONCE SADECE ONAY TUSUNA BASILMASINI
;BEKLER
;*** PRGM'YE BASILMA DURUMUNDAYSAYSA PRGM BAYRAGINI (FLAGS2,0) SET EDEREK
;*** ANA PROG.'A DONER

```



```

        MOVLW      3AH
        MOVWF     SAYACB

        MOVLW      3AH
        MOVWF     SAYACA
        DECFSZ    SAYACA,F
        GOTO     $-1

        DECFSZ    SAYACB,F
        GOTO     $-5
        RETURN

;*****
;
SET_65msn
        CLRWDT
        MOVLW     7BH           ;PRESCALER :16, POSTSCALER:16
        MOVWF     T2CON        ;TMR2ON:OFF
        CLRF     TMR2
        BCF     PIR1,TMR2IF ;PR2:0FFH
        BSF     T2CON,TMR2ON
;
;       BTFSS    PIR1,TMR2IF
;
;       GOTO     $-1
;
;       BCF     T2CON,TMR2ON
        RETURN

;*****
;
DELAY_500msn
        CALL     SET_500msn
        BTFSS    PIR1,TMR1IF
        GOTO     $-1
        CLRF     T1CON
        CLRWDT
        RETURN

;*****
;
SET_500msn
        CLRWDT
        MOVLW     30H
        MOVWF     T1CON        ;PRESCALER:8, TMR1:OFF
        MOVLW     0DBH
        MOVWF     TMR1L
        MOVLW     0BH
        MOVWF     TMR1H
        BCF     PIR1,TMR1IF    ;BAYRAK=0
        BSF     T1CON,TMR1ON   ;TMR1 ON
        RETURN

;*****
;
;GİRİŞ:W
DELAY_X10msn
        MOVWF     SAYACY
        CALL     DELAY_10msn
        DECFSZ    SAYACY,F
        GOTO     $-2
        CLRWDT
        RETURN

;*****
;
;GİRİŞ:W
DELAY_Xsn
        MOVWF     SAYACY
        CALL     DELAY_500msn
        DECFSZ    SAYACY,F

```



```

        DECFSZ    SAYAC2,F
        GOTO $-0X04
        RETURN
;-----
ADC_TABLE
        MOVLW    HIGH ADC_TABLE
        MOVWF    PCLATH
        MOVF     KANAL,W
        ADDWF    PCL,F
        RETLW00H
        RETLW08H
        RETLW18H
;*****
;
DELAY_60us
        CLRWDT
        MOVLW    0x14
        MOVWF    SAYAC1                ;3*20=60

        DECFSZ    SAYAC1,F
        GOTO $-1
        RETURN
;*****
;
READ_ADC
        BSF     ADCON0,ADON                ;ADC ON
        CALL    DELAY_60us

        BSF     ADCON0,2                    ;GO!/DONE <- 1 (dönüşüm baslasin)
        BTFSC  ADCON0,2
        GOTO $-1
        BCF     ADCON0,ADON                ;ADC OFF
        RETURN
;*****
;
SHOW_ISI
        BCF     INTCON,GIE
        MOVF    POINTER,W                ;POINTER->BCD
        BSF     INTCON,GIE

        CALL    HEXTOBCD
        CALL    CLEAR_STR2
        MOVLW    TENS_ONES
        CALL    _PRINTF
        CALL    LCD_ZEROFIVE                ;ONDALIK DERECEYİ GÖSTER
        MOVLW    TENS_ONES
        CALL    SEND_TEMP
        RETURN
;-----
LCD_ZEROFIVE
        MOVLW    '.'
        CALL    LCD_PUTC
        BTFSS  ZEROFIVE
        MOVLW    '0'
        BTFSC  ZEROFIVE
        MOVLW    '5'
        CALL    LCD_PUTC
        CALL    IMDERECE
        CALL    blink_OFF

        RETURN

```

```

SEND_TEMP
    MOVWF    FSR
    BCF     STATUS,IRP           ;BANK0
    SWAPF   INDF,F
    MOVLW   0FH
    ANDWF   INDF,W             ;MSD, W'DE
    ADDLW   30H                ;RAKAM -> ASCII
    CALL    SEND_DATA          ;1.DIGIT (MSD) RS232'YE

    SWAPF   INDF,F
    MOVLW   0FH
    ANDWF   INDF,W             ;LSD, W'DE
    ADDLW   30H                ;RAKAM -> ASCII
    CALL    SEND_DATA          ;2.DIGIT (LSD) RS232'YE
    ;-----

RETURN

;*****
;*****
;GİRİŞ: KANAL
;ÇIKIŞ: POINTER, ZEROFIVE
;-----
READ_NTC
;    BCF     INTCON,GIE
;    CALL    READ_ADC8         ;ADC SONUC=TOP_H:TOP_L
;    CLRF   POINTER           ;NTC DEĞER İŞARETÇİSİ
;-----
;SICAKLIK<0 C İSE SICAKLIK 0 C OLARAK GÖSTERİLECEKTİR
;SICAKLIK>=200 C İSE SICAKLIK 99.5 C OLARAK GÖSTERİLECEKTİR
;-----
    CLRWDT
    CALL    NTC_TABLE
    MOVWF   TEMPL             ;NTC TABLE DÜŞÜK BYTE
    CALL    NTC_TEMPH         ;NTC TABLE YÜKSEK BYTE----->TEMPH:TEMPH
;-----
COMPARE_ADC_NTCTABLE         ;TOP_H:TOP_L>=TEMPH:TEMPH?

    MOVF    TEMPH,W
    SUBWF   TOP_H,W
    BTFSS   CARRY             ;TOP_H>=TEMPH?
    GOTO    SICAKLIK_SONUC    ;HAYIR
    BTFSS   ZERO              ;TOP_H=TEMPH?
    GOTO    LOOP_NTC

    MOVF    TEMPL,W
    SUBWF   TOP_L,W
    BTFSS   CARRY             ;TOP_H>=TEMPH?
    GOTO    SICAKLIK_SONUC    ;HAYIR
    BTFSC   ZERO              ;TOP_H=TEMPH?
    GOTO    SICAKLIK_SONUC

;-----
LOOP_NTC
    INCF    POINTER,F
    MOVF    POINTER,W
    XORLW   0C8H              ;200?
    BTFSS   ZERO
    GOTO    READ_NTC+2

```

```

;-----
    DECF POINTER,F          ;SON İŞARETÇİDEN ÖNCEKİ DEĞER
                             ;GERÇEK SICAKLIK
SICAKLIK_SONUC

    BTFSS POINTER,0
    BCF  ZEROFIVE          ;+0.0 C
    BTFSC POINTER,0
    BSF  ZEROFIVE          ;+0.5 C

    BCF  CARRY
    RRF  POINTER,F         ;POINTER=POINTER/2
;    BSF  INTCON,GIE
    RETURN
;////////////////////////////////////
;-----
NTC_TEMP

;X_0_32
    MOVF POINTER,W
    SUBLW 20H              ;0-32
    BTFSS CARRY
    GOTO XA
    CLRF TEMPH
    RETURN

;X_33_93
XA
    MOVF POINTER,W
    SUBLW 5DH              ;33-93
    BTFSS CARRY
    GOTO XB
    MOVLW 01H
    MOVWF TEMPH
    RETURN

XB
;X_94_168
    MOVF POINTER,W
    SUBLW 0A8H             ;94-168
    BTFSS CARRY
    GOTO XC
    MOVLW 02H
    MOVWF TEMPH
    RETURN

XC
;X_169_200
    MOVLW 03H
    MOVWF TEMPH
    RETURN

;-----
    ORG 0320H
NTC_TABLE
    MOVLW HIGH NTC_TABLE
    MOVWF PCLATH
    MOVF POINTER,W
    ADDWF PCL,F
; "---" ;0 --9.5 C
    RETLW97H
    RETLW9AH

```

```

RETLW9DH
RETLW9FH
RETLW0A2H
RETLW0A5H
RETLW0A8H
RETLW0ABH
RETLW0AEH
RETLW0B1H
RETLW0B4H
RETLW0B7H
RETLW0BAH
RETLW0BDH
RETLW0C0H
RETLW0C3H
RETLW0C7H
RETLW0CAH
RETLW0CDH
RETLW0D1H
;"----"
RETLW0D5H ;10 --19.5 C
RETLW0D8H
RETLW0DBH
RETLW0DFH
RETLW0E2H
RETLW0E5H
RETLW0E9H
RETLW0ECH
RETLW0F0H
RETLW0F3H
RETLW0F7H
RETLW0FAH
RETLW0FEH
RETLW02H ;16.5 C
RETLW05H
RETLW09H
RETLW0DH
RETLW11H
RETLW14H
RETLW18H
;"----"
RETLW1FH ;20 --24.5 C
RETLW22H
RETLW26H
RETLW2AH
RETLW2EH
RETLW32H
RETLW36H
RETLW3AH
RETLW3EH
RETLW42H
;"----"
RETLW47H ;25 --29.5 C
RETLW4BH
RETLW4FH
RETLW53H
RETLW57H
RETLW5BH
RETLW60H
RETLW64H

```



```

RETLW68H
RETLW6CH
;"---"
RETLW71H ;30 --39.5 C
RETLW75H
RETLW79H
RETLW7DH
RETLW82H
RETLW86H
RETLW8AH
RETLW8EH
RETLW92H
RETLW97H
RETLW9BH
RETLW9FH
RETLW0A4H
RETLW0A8H
RETLW0ACH
RETLW0B1H
RETLW0B5H
RETLW0B9H
RETLW0BEH
RETLW0C2H
;"---"
RETLW0C7H ;40 --49.5 C
RETLW0CBH
RETLW0CFH
RETLW0D3H
RETLW0D7H
RETLW0DCH
RETLW0E0H
RETLW0E4H
RETLW0E8H
RETLW0ECH
RETLW0F0H
RETLW0F5H
RETLW0F9H
RETLW0FDH
RETLW01H ;47 C
RETLW05H
RETLW0AH
RETLW0EH
RETLW12H
RETLW16H
;"---"
RETLW1BH ;50 --59.5 C
RETLW1FH
RETLW23H
RETLW27H
RETLW2BH
RETLW2FH
RETLW32H
RETLW36H
RETLW3AH
RETLW3EH
RETLW42H
RETLW46H
RETLW49H
RETLW4DH

```

```
RETLW51H
RETLW55H
RETLW59H
RETLW5CH
RETLW60H
RETLW64H
;"---"
RETLW69H ;60 --69.5 C
RETLW6DH
RETLW71H
RETLW74H
RETLW78H
RETLW7BH
RETLW7FH
RETLW82H
RETLW86H
RETLW89H
RETLW8DH
RETLW90H
RETLW93H
RETLW97H
RETLW9AH
RETLW9EH
RETLW0A1H
RETLW0A4H
RETLW0A8H
RETLW0ABH
;"---"
RETLW0AFH ;70 --79.5 C
RETLW0B2H
RETLW0B5H
RETLW0B8H
RETLW0BBH
RETLW0BEH
RETLW0C1H
RETLW0C4H
RETLW0C7H
RETLW0C9H
RETLW0CCH
RETLW0CFH
RETLW0D2H
RETLW0D5H
RETLW0D8H
RETLW0DBH
RETLW0DDH
RETLW0E0H
RETLW0E3H
RETLW0E6H
;"---" ;80 --84.5 C
RETLW0EAH
RETLW0EDH
RETLW0EFH
RETLW0F2H
RETLW0F5H
RETLW0F7H
RETLW0FAH
RETLW0FCH
RETLW0FFH
RETLW01H
```



```

; 10 litre/dak --> 9000 rpm --> 150 Hz -6.6
; 9.5 litre/dak --> 8400 rpm --> 140 Hz- 7.1
; 9 litre/dak --> 7850 rpm --> 130.8 Hz- 7.6
; 8.5 litre/dak --> 7400 rpm --> 123.3 Hz -8.1
; 8 litre/dak --> 6800 rpm --> 113.3 Hz -8.8
; 7.5 litre/dak --> 6300 rpm --> 105 Hz- 9.5
; 7 litre/dak --> 5800 rpm --> 96.6 Hz- 10.3
; 6.5 litre/dak --> 5300 rpm --> 88.3 Hz- 11.3
; 6 litre/dak --> 4800 rpm --> 80 Hz -12.5
; 5.5 litre/dak --> 4400 rpm --> 73.3 Hz- 13.6
; 5 litre/dak --> 3900 rpm --> 65 Hz -15.3
; 4.5 litre/dak --> 3400 rpm --> 56.6 Hz- 17.6
; 4 litre/dak --> 2900 rpm --> 48.3 Hz - 20
; 3.5 litre/dak --> 2500 rpm --> 41.6 Hz - 24
; 3.0 litre/dak --> 2025 rpm --> 33.7 Hz - 29.6
; 2.5 litre/dak --> 1625 rpm --> 27 Hz - 37 msn
; 2.3 litre/dak --> 1500 rpm 25 Hz - 40
; 2 litre/dak --> 1200 rpm --> 20 Hz - 50 msn
; 1.8 litre/dak --> 1000 rpm - 16.6 Hz - 60
; 1,5 litre/dak --> 750 rpm --> 12.5 Hz - 80 msn

```

```

;.....
FLOWMETRE_FUGAS_ENABLE

```

```

    CLRF TMR1_INT
    CALL SET_500msn
    CLRF TMR0          ;SAYICI SIFIR
    BANK1
    BSF PIE1,TMR1IE   ;TMR1 INTERRUPT ENABLE
    BANK0
    RETURN

```

```

;.....
LCD_ZEROFIVEF

```

```

    MOVLW    ':'
    CALL LCD_PUTC
    BTFSS FLW
    MOVLW    '0'
    BTFSC FLW
    MOVLW    '5'
    CALL LCD_PUTC
    CALL blink_OFF
    RETURN

```

```

;.....
SHOW_FLOW

```

```

    BCF INTCON,7
    MOVF SYC1,W          ;SYC1->BCD
    BSF INTCON,7
    CALL HEXTOBCD
    MOVLW    0CAH
    CALL LCD_GONDER
    MOVLW    TENS_ONES
    CALL _PRINTF
    CALL LCD_ZEROFIVEF   ;ONDALIK DERECEYİ GÖSTER
    MOVLW    'L'
    CALL LCD_PUTC
    MOVLW    'T'
    CALL LCD_PUTC
    RETURN

```

```

;.....

```

READ_FUGAS

```

    CLR  SYC1           ;tablo işaretçisi
    MOVF TMR0,W
    MOVWF    TEMPF
    SUBLW    0CH
    BTFSC CARRY
    GOTO XFUGAS+2

```

XF

```

    CALL FUGAS_FLOW_TABLE
    SUBWF    TEMPF,W
    BTFSS CARRY
    GOTO XFUGAS

```

```

    INCF  SYC1,F
    MOVF  SYC1,W
    SUBLW    16H
    BTFSS ZERO
    GOTO XF
    DECF  SYC1,F

```

XFUGAS

```

    MOVLW    03H
    ADDWF    SYC1,F

```

```

    BTFSS SYC1,0
    BCF  FLW           ;+0.0 C
    BTFSC SYC1,0
    BSF  FLW           ;+0.5 C

```

```

    BCF  CARRY
    RRF  SYC1,F       ;SYC1=SYC1/2
    RETURN

```

.....
FUGAS_FLOW_TABLE

```

    MOVLW  HIGH FUGAS_FLOW_TABLE
    MOVWF  PCLATH
    MOVF   SYC1,W
    ADDWF  PCL,F
    RETLW  0DH ;1.5
    RETLW  14H
    RETLW  1BH ;2.5
    RETLW  22H
    RETLW  2AH
    RETLW  30H
    RETLW  39H
    RETLW  41H ;5
    RETLW  49H
    RETLW  50H
    RETLW  58H
    RETLW  61H
    RETLW  69H ;7.5
    RETLW  71H
    RETLW  7BH
    RETLW  82H
    RETLW  8CH
    RETLW  96H ;10

```



```

*****LCD HAZIRLIK*****
;
*****
;
OOP
    CALL  PREPARE_LCD      ;LCD P. YÜKLE

*****
;
***** HABERLEŞME HAZIRLIK*****
;
*****
;
COMMUNICATION_SETTING

    CALL  TX_RX_PREPARETION

*****
;
*****FABRİKA DEĞERLERİ KAYIT YAZMA-OKUMA*
;
*****
;
    CLRF  E_ADR
    CALL  OKU_EEDATA
    XORLW    0FFH
    BTFSS ZERO          ;ILK KULLANILMA ANIMI
    GOTO  SYSTEM_UPDATE ;HAYIR DAHA ÖNCE AYARLAR KAYDEDİLMİS

;-----
;-----
;
    CLRF  E_ADR          ;FABRİKA DEĞERLERİ
    MOVLW    0AAH
    MOVWF   E_DATA
    CALL  YAZ_EEDATA    ;EEADR=00H

    CALL  YAZ_EEDATA

    INCF  E_ADR,F
    MOVLW    40H        ;BCD ALT LİMİT=40 C E_ADR:01H
    MOVWF   E_DATA
    CALL  YAZ_EEDATA    ;EEADR=01H

    INCF  E_ADR,F
    MOVLW    90H        ;BCD ÜST LİMİT=90 C E_ADR:02H
    MOVWF   E_DATA
    CALL  YAZ_EEDATA    ;EEADR=02H

    INCF  E_ADR,F        ;EEADR=03H
    CLRF  E_DATA        ;SESLİ UYARI OFF,manuel,kapalı
    CALL  YAZ_EEDATA    ;EEADR=09H

;-----
;-----
;
SYSTEM_UPDATE          ;KAYITLI OLAN GÖSTERGELER OKUNUYOR

    MOVLW    01H        ;BCD ALT LİMİT
    MOVWF   E_ADR
    CALL  OKU_EEDATA    ;EEADR=01H
    MOVWF   ALT_LIMIT

    INCF  E_ADR,F        ;BCD ÜST LİMİT
    CALL  OKU_EEDATA    ;E_ADR:02H
    MOVWF   UST_LIMIT

    INCF  E_ADR,F        ;OTOMATİK
    CALL  OKU_EEDATA    ;E_ADR:03H

```



```
MOVWF FLAG2 ;BUZ,OTO_ON,MANUEL
```

```
*****
;*****
;***** KESME HAZIRLIK*****
;*****
BSF INTCON,PEIE ;ÇEVRESEL KESMELER AKTIF
BSF INTCON,GIE ;GLOBAL KESMELER AKTIF
CALL CLEAR_DISP ;EKRANI SİL
CALL FLOWMETRE_FUGAS_ENABLE ;TMR1 INT ENABLE
BTFSC LCD_LED
GOTO $+3
MOVLW 96H
CALL DELAY_X10msn ;1.5 sn bekle
```

```
#####
```

```
BEGIN1
```

```
BCF LCD_LED ;LED SÖNÜK
BCF TUS_ON ;MENÜYE GİRİLMEDİ
CLRF TUS_KONTROL ;BÜTÜN KONTROLLER DISABLE
```

```
;PID POWER RESET
```

```
BCF PID ;PID KONTROL KAPALI
CLRF COUNTER ;PID INTERRUPT
```

```
BANK2
```

```
CLRF E_1_EXP
CLRF E_1_B0
CLRF E_1_B1
CLRF I_1_EXP
CLRF I_1_B0
CLRF I_1_B1
BANK_0
```

```
MOVLW 32H
MOVWF SETP ;SICAKLIK SET DEĞERİ 50 C
```

```
BCF INTCON,RBIF ;PORTB KESME BAYRAGI SIFIR
BSF INTCON,RBIE ;PORTB KESMESİ AKTIF
```

```
.* * * * * * * * * * * *
;*****
;***** KONTROL ALGORİTMASI*****
;*****
.* * * * * * * * * * * *
```

```
MAIN_LOOP
```

```
MOVLW 06H
MOVWF WORD_POINT ;AİRFEL KOMBİ
MOVLW 80H
CALL LCD_GONDER ;1.SATIR
CALL wordROM2LCD
```

```
MAIN_LOOP1
```

```
CALL SHOW_ISI
CALL SHOW_FLOW
MOVLW 96H
CALL DELAY_X10msn ;1.5 sn bekle
```

```
BTFSS TUS_ON
GOTO MAIN_LOOP1
```



```

.** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** **
;
;////////////////////////////////////
;*****MENÜYE GIRIS YAPILDI*****
;

;////////////////////////////////////
.** ** ** ** ** ** **^
;

ENTRY
    CLRF TUS_KONTROL      ;OKLAR GEÇERSİZ, BASILAN RAKAM LCD'DE
GÖZÜKMEYECEK
    CALL TUS_TARAMA
    BTFSC PRGM            ;PROGRAM TUSUNA BASILDIMI
    GOTO PROGRAM
;
; BTFSC ONNOFF          ;SİSTEM ON MU?
; GOTO SISTEM_ONOFF    ;SİSTEM OTOMATİK MODDA ONN
; GOTO BEGIN1          ;HAYIR DİGER TUSLARA BASILDI ES GEÇ

;-----
;
;-----
SISTEM_ONOFF

    MOVLW    03H
    MOVWF    E_ADR

    MOVLW    02H          ;TERSLEME
    XORWF    FLAG2,F     ;FLAG2,1:OTO_ON

    MOVF    FLAG2,W
    MOVWF    E_DATA
    CALL    YAZ_EEDATA
    GOTO    BEGIN1

;-----
;
;-----
PROGRAM
    BSF     LCD_LED      ;LEDİ YAK

;=====
;
;=====
;
;=====
;
;=====
;
;=====
MENU

    MOVLW    00H
    MOVWF    WORD_POINT ;MENÜ

    MOVLW    01H
    MOVWF    SINIR_SAYAC
    MOVLW    03H
    MOVWF    SINIR      ;3 ADET ALT MENÜ

    CLRF    TUS_KONTROL
    BCF     STR_SIRA     ;1.SATIR
    BSF     CIFTSTR      ;ÇİFT SATIR
    CALL    TABLO        ;"MENÜ" YAZILACAK ve "ALT LİMİT" yazılacak
    CALL    SHOW_KAYIT
    CALL    blink_OFF

```

```

;-----
;-----
BAS_KODGIR
  CLRF TUS_KONTROL      ;LCD GÖRÜNMESİN, OKLAR GEÇERSİZ
  CALL TUS_TARAMA

  BTFSC PRGM           ;PRGM?
  GOTO SYSTEM_UPDATE  ;evet, MENU'DEN ÇIK

  BTFSC OKEY           ;ONAY?
  GOTO kodgir_TABLO   ;evet

ASAGI_KODGIR
  BTFSS MINUS          ;OK AŞAĞI?
  GOTO YUKARI_KODGIR  ;hayır

  MOVF SINIR,W
  SUBWF SINIR_SAYAC,W
  BTFSC CARRY          ;SINIR_SAYAC >= SINIR? (C=1?)
  CLRF SINIR_SAYAC    ;evet
  INCF SINIR_SAYAC,F  ;hayır
  GOTO SECOND_ROW

YUKARI_KODGIR
  BTFSS PLUS           ;OK YUKARI?
  GOTO BAS_KODGIR     ;hayır

  DECFSZ SINIR_SAYAC,F ;SINIR_SAYAC ONCEDEN YUKLU
  GOTO SECOND_ROW
  MOVF SINIR,W
  MOVWF SINIR_SAYAC

SECOND_ROW
  CALL CLEAR_STR2
  CALL KODGIRX_WORDX
  MOVWF WORD_POINT
  CALL wordROM2LCD
  CALL SHOW_KAYIT
  CALL blink_OFF
  GOTO BAS_KODGIR

;-----
kodgir_TABLO
  MOVLW HIGH (kodgir_TABLO+0x03)
  MOVWF PCLATH
  DECF SINIR_SAYAC,W
  ADDWF PCL,F         ; <KOD NO:>
  GOTO ALT_LIMITX     ; 00
  GOTO UST_LIMITX    ; 01
  GOTO SESLI_UYARIX  ; 02

KODGIRX_WORDX
  MOVLW HIGH (KODGIRX_WORDX+0x03)
  MOVWF PCLATH
  DECF SINIR_SAYAC,W
  ADDWF PCL,F
  RETLW01H

```

```

RETLW02H
RETLW03H

;=====
;=====
;=====
;LIMIT 0/1(ALT/ÜST LİMİT)
;-----

ALT_LIMITX

    BCF    LIMIT                ;ALT LİMİT

    MOVLW   01H                ;ALT LİMİT
    BTFSC  LIMIT
    MOVLW   02H                ;ÜST LİMİT
    MOVWF   WORD_POINT

    BSF    STR_SIRA            ;SADECE 2.SATIR
    CALL   TABLO                ;WORD_POINT DEĞİŞMEDİ
                                ;ALT MENU OLMADIĞI İÇİN HIÇ DEĞİŞMEYECEK

    MOVLW   ':'
    CALL   LCD_PUTC

    MOVLW   UST_LIMIT          ;ALT LİMİT DEĞERİ BCD OLARAK GÖSTERİLDİ
    BTFSS  LIMIT
    MOVLW   ALT_LIMIT          ;ALT LİMİT DEĞERİ BCD OLARAK GÖSTERİLDİ
    CALL   _PRINTF
    CALL   İMDERECE            ;DERECE TİKİ OK

    MOVLW   0CAH
    MOVWF   LCD_ADRES
    CALL   LCD_GONDER          ;CURSOR 1.RAKAMI GÖSTERİYOR
    CALL   blink_ON

    MOVF   UST_LIMIT,W         ;ÜST LİMİT
    BTFSS  LIMIT
    MOVF   ALT_LIMIT,W         ;BCD DEĞER
    MOVWF   TEMPX             ;KAYITLI DEĞER TEMP'D

    CLRF   TUS_KONTROL        ;OKLAR GEÇERSİZ, BASILAN RAKAM LCD'DE
    GÖZÜKMEYECEK
    BSF    GORUN                ;BASILAN RAKAM LCD'DE GÖZÜKECEK
;-----
LOOP_TUS

    CALL   TUS_TARAMA

;-----
TUS_PRGM
    BTFSC  PRGM                ;PRGM TUŞU MU?
    GOTO   MENU
;-----
;-----
;-----
TUS_OKEY
    BTFSS  OKEY                ;OKEY TUŞU MU?
    GOTO   LOOP_TUS

```

```

        BTFSS LIMIT          ;ÜST LİMİT Mi?
        GOTO CHECK_MIN      ;HAYIR
;-----
CHECK_MAX
        MOVF  TEMPX,W        ;YENİ KAYIT EĞER ÜST LİMİTSE
        SUBWF  ALT_LIMIT,W   ;ALT LİMİT'TEN DAHA KÜÇÜK OLAMAZ
        BTFSC CARRY
        GOTO  ALT_LIMITX+1
        GOTO  RKM
;-----
CHECK_MIN

        MOVF  UST_LIMIT,W    ;YENİ KAYIT EĞER ALT LİMİTSE
        SUBWF  TEMPX,W       ;ÜST LİMİT'TEN DAHA BÜYÜK OLAMAZ
        BTFSC CARRY
        GOTO  ALT_LIMITX+1
;-----
RKM
        MOVF  TEMPX,W
        BTFSC LIMIT          ;ÜST LİMİT Mi?
        MOVWF  UST_LIMIT     ;SON DEĞER RAM'DE
        BTFSS LIMIT
        MOVWF  ALT_LIMIT     ;SON DEĞER RAM'DE

        MOVWF  E_DATA
        MOVLW  01H
        MOVWF  E_ADR        ;ALT LİMİT
        BTFSC LIMIT
        INCF  E_ADR,F        ;ÜST LİMİT
        CALL  YAZ_EEDATA
        GOTO  MENU

;=====
;=====
;=====
;=====

UST_LIMITX
        BSF   LIMIT          ;ÜST LİMİT
        GOTO  ALT_LIMITX+1

;=====
;=====
;=====
;=====
;GENF :ON/OFF(1/0)
;-----

SESLI_UYARIX

        BSF   STR_SIRA      ;SADECE 2.SATIR
        CALL  TABLO         ;WORD_POINT DEĞİŞMEDİ
                                ;ALT MENU OLMADIĞI İÇİN HIÇ DEĞİŞMEYECEK

        MOVLW  ':'
        CALL  LCD_PUTC

        MOVLW  04H          ;ON
        BTFSS BUZ
        MOVLW  05H          ;OFF
        MOVWF  WORD_POINT
        CALL  wordROM2LCD

```

```

        MOVLW      0CCH
        CALL  LCD_GONDER      ;13.KARAKTERDE CURSOR
        CALL  blink_ON

        BTFSS BUZ
        BCF  GENF      ;BUZZER OFF
        BTFSC BUZ
        BSF  GENF      ;BUZZER ON

;-----
LOOP_TUS1

        CLRF  TUS_KONTROL      ;OKLAR GEÇERSİZ, BASILAN RAKAM LCD'DE
        GÖZÜKMEYECEK
        CALL  TUS_TARAMA

;-----
        BTFSS PLUS      ;+ TUŞU MU?
        GOTO TUS_M1      ;HAYIR

;-----
TUS_P1

        MOVLW      20H
        XORWF      FLAG1,F      ;GENF EXOR

        MOVLW      04H      ;ON
        BTFSS GENF
        MOVLW      05H      ;OFF
        MOVWF      WORD_POINT
        CALL  wordROM2LCD

        MOVLW      0CCH
        CALL  LCD_GONDER      ;13.KARAKTERDE CURSOR
        GOTO  LOOP_TUS1

;-----
TUS_M1
        BTFSC MINUS      ;- TUŞU MU?
        GOTO TUS_P1

;-----
TUS_PRGM1
        BTFSC PRGM      ;PRGM TUŞU MU?
        GOTO MENU

;-----
TUS_OKEY1
        BTFSS OKEY      ;OKEY TUŞU MU?
        GOTO LOOP_TUS1

        BTFSS GENF
        BCF  BUZ      ;BUZZER OFF
        BTFSC GENF
        BSF  BUZ      ;BUZZER ON

        MOVF  FLAG2,W      ;BUZ=FLAG2,0
        MOVWF E_DATA

        MOVLW      03H
        MOVWF      E_ADR

```

```

CALL YAZ_EEDATA
;CALL OUTPUT_KONTROL
GOTO MENU
;=====
;=====
;=====

ORG 0700H

CHR_TABLE
    MOVLW    HIGH (CHR_TABLE+0x03)
    MOVWF    PCLATH
    MOVF     WORD_POINT,W
    ADDWF    PCL,F
    GOTO     _00
    GOTO     _01
    GOTO     _02
    GOTO     _03
    GOTO     _04
    GOTO     _05
    GOTO     _06
;   GOTO     _07
;   GOTO     _08
;   GOTO     _09
;   GOTO     _0A
;   GOTO     _0B

_00
MENU1
    MOVLW    HIGH (MENU1+0x03)
    MOVWF    PCLATH
    MOVF     CHR_POINT,W
    ADDWF    PCL,F
    RETLW04H    ;SATIR UZUNLUĞU=4
    RETLW'M'    ;M
    RETLW'E'    ;E
    RETLW'N'    ;N
    RETLW'Ü'    ;Ü
;-----
_01
ALT_LIMITY
    MOVLW    HIGH (ALT_LIMITY+0x03)
    MOVWF    PCLATH
    MOVF     CHR_POINT,W
    ADDWF    PCL,F
    RETLW09H    ;SATIR UZUNLUĞU=9
    RETLW'A'    ;ALT LİMİT
    RETLW'L'
    RETLW'T'
    RETLW' '
    RETLW'L'
    RETLW'i'
    RETLW'M'
    RETLW'i'
    RETLW'T'
;-----
_02

```



```

UST_LIMITY
  MOVLW      HIGH (UST_LIMITY+0x03)
  MOVWF     PCLATH
  MOVF  CHR_POINT,W
  ADDWF     PCL,F
  RETLW09H      ;SATIR UZUNLUĞU=9
  RETLW0F5H     ;ÜST LİMİT
  RETLW'S'
  RETLW'T'
  RETLW' '
  RETLW'L'
  RETLW'i'
  RETLW'M'
  RETLW'i'
  RETLW'T'

```

```

;-----
_03
SESLI_UYARI
  MOVLW      HIGH (SESLI_UYARI+0x03)
  MOVWF     PCLATH
  MOVF  CHR_POINT,W
  ADDWF     PCL,F
  RETLW0BH      ;SATIR UZUNLUĞU=11
  RETLW'S'      ;SESLİ UYARI
  RETLW'E'
  RETLW'S'
  RETLW'L'
  RETLW'i'
  RETLW' '
  RETLW'U'
  RETLW'Y'
  RETLW'A'
  RETLW'R'
  RETLW'I'

```

```

;-----
_04
_ON
  MOVLW      HIGH (_ON+0x03)
  MOVWF     PCLATH
  MOVF  CHR_POINT,W
  ADDWF     PCL,F
  RETLW03H      ;SATIR UZUNLUĞU=3
  RETLW'O'
  RETLW'N'
  RETLW' '

```

```

;-----
_05
_OFF
  MOVLW      HIGH (_OFF+0x03)
  MOVWF     PCLATH
  MOVF  CHR_POINT,W
  ADDWF     PCL,F
  RETLW03H      ;SATIR UZUNLUĞU=3
  RETLW'O'      ;O
  RETLW'F'      ;F
  RETLW'F'      ;F

```

```

;-----
_06

```

```

_AIRFEL
    MOVLW        HIGH (_AIRFEL+0x03)
    MOVWF       PCLATH
    MOVF  CHR_POINT,W
    ADDWF       PCL,F
    RETLW0CH           ;SATIR UZUNLUĞU=12
    RETLW'A'           ;A
    RETLW'i'           ;İ
    RETLW'R'           ;R
    RETLW'F'           ;F
    RETLW'E'           ;E
    RETLW'L'           ;L
    RETLW' '           ;
    RETLW'K'           ;K
    RETLW'O'           ;o
    RETLW'M'           ;m
    RETLW'B'           ;b
    RETLW'i'           ;i

```

```

;%%%%%%%%%%
;%%%%%%%%%%
;***** INTERRUPT PROGRAM *****
;%%%%%%%%%%
;%%%%%%%%%%

```

INTERRUPT

```

    CLRWDI
    BTFSS INTCON,RBIE
    GOTO COMMUNICATION_INTERRUPT
    BTFSS INTCON,RBIF           ;PORTB KESMESİ Mİ?
    GOTO COMMUNICATION_INTERRUPT;HAYIR ,DİĞER INTERUPT

```

```

;=====

```

PORTB_INTERRUPT

```

    BCF  TUS_ON
    BCF  INTCON,RBIF           ;BAYRAK RESET

    CALL DELAY_5msn           ;5MSN BEKLE
    MOVF PORTB,W
    ANDLW 0F0H
    XORLW 0F0H
    BTFSC ZERO                 ;HERHANGİ BİR TUS BASILI MI?
    GOTO EXIT_INTERRUPT       ;hayır, değil

    BCF  INTCON,RBIE           ;PORTB KESMESİ DISABLE
    BSF  TUS_ON                ;evet, TUSA BASILDI
    GOTO EXIT_INTERRUPT       ;hayır, çık (bir tuşa basıldı)

```

```

;=====

```

COMMUNICATION_INTERRUPT

```

    BANK1
    BTFSS PIE1,RCIE           ;receive kesmesi aktif mi?
    GOTO TMR1_INTERRUPT       ;hayır, diğer kesmelere...
    BANK0
    BTFSS PIR1,RCIF           ;receive kesmesi mi?
    GOTO TMR1_INTERRUPT

```



```

GOTO EXIT_INTERRUPT
CLRF COUNTER

PAGE1
CALL PID_ALGORITHM
BTFSC ZERO
GOTO PWMOFF
CALL SEND_PWM
GOTO EXIT_INTERRUPT
PWMOFF
CALL PWM_OFF
.
;
; MOVF POINTER,W
; CALL SEND_DATA

;=====
EXIT_INTERRUPT
BANK0
MOVF FSR_TEMP,W
MOVWF FSR ;FSR'yi geri yükle
MOVF PCLATH_TEMP,W
MOVWF PCLATH ;PCLATH'i geri yükle

SWAPF STATUS_TEMP,W ;STATUS'u geri yükle
MOVWF STATUS
SWAPF W_TEMP,F ;W'yi geri yükle
SWAPF W_TEMP,W
RETFIE
;%%%%%%%%%%
;%%%%%%%%%%

ORG 0800H

.*****
;
; Integer to float conversion
; Input: 16 bit 2's complement integer right justified in AARGB0, AARGB1
; Use: CALL FLO1624 or CALL FLO24
; Output: 24 bit floating point number in AEXP, AARGB0, AARGB1
; Result: AARG <-- FLOAT( AARG )
; Max Timing: 11+72 = 83 clks SAT = 0
; 11+77 = 88 clks SAT = 1
; Min Timing: 7+14 = 21 clks AARG = 0
; 7+18 = 25 clks
; PM: 11+26 = 37 DM: 6
;-----
FLO1624
FLO24
MOVLW 0fh+EXPBIAS ; initialize exponent and add bias
MOVWF EXP
MOVF AARGB0,W
MOVWF SIGN
BTFSS AARGB0,MSB ; test sign
GOTO NRM2424
COMF AARGB1,F ; if < 0, negate and set MSB in SIGN

```

```

    COMF AARGB0,F
    INCF AARGB1,F
    BTFSC _Z
    INCF AARGB0,F
;*****
;
; Normalization routine
; Input: 24 bit unnormalized floating point number in AEXP, AARGB0, AARGB1,
; with sign in SIGN, MSB and other bits zero.
; Use: CALL NRM2424 or CALL NRM24
; Output: 24 bit normalized floating point number in AEXP, AARGB0, AARGB1
; Result: AARG <-- NORMALIZE( AARG )
; Max Timing: 10+6+7*7+7 = 72 clks SAT = 0
; 10+6+7*7+1+11 = 77 clks SAT = 1
; Min Timing: 14 clks AARG = 0
; 5+9+4 = 18 clks
; PM: 26 DM: 6
;-----
NRM2424
NRM24
    CLRF TEMP                ; clear exponent decrement
    MOVF AARGB0,W           ; test if highbyte=0
    BTFSS _Z
    GOTO NORM2424
    MOVF AARGB1,W           ; if so, shift 8 bits by move
    MOVWF AARGB0
    BTFSC _Z                 ; if highbyte=0, result=0
    GOTO RES024
    CLRF AARGB1
    BSF TEMP,3
NORM2424
    MOVF TEMP,W
    SUBWF EXP,F
    BTFSS _Z
    BTFSS _C
    GOTO SETFUN24
    BCF _C                   ; clear carry bit
NORM2424A
    BTFSC AARGB0,MSB        ; if MSB=1, normalization done
    GOTO FIXSIGN24
    RLF AARGB1,F            ; otherwise, shift left and
    RLF AARGB0,F            ; decrement EXP
    DECFSZ EXP,F
    GOTO NORM2424A
    GOTO SETFUN24           ; underflow if EXP=0
FIXSIGN24
    BTFSS SIGN,MSB
    BCF AARGB0,MSB         ; clear explicit MSB if positive
    RETLW 0
RES024
    CLRF AARGB0             ; result equals zero
    CLRF AARGB1
    CLRF AARGB2             ; clear extended byte
    CLRF EXP
    RETLW 0
;*****
;
; Integer to float conversion

```

```

; Input: 24 bit 2's complement integer right justified in AARGB0, AARGB1, AARGB2
; Use: CALL FLO2424
; Output: 24 bit floating point number in AEXP, AARGB0, AARGB1
; Result: AARG <-- FLOAT( AARG )
; Max Timing: 14+94 = 108 clks RND = 0
; 14+103 = 117 clks RND = 1, SAT = 0
; 14+109 = 123 clks RND = 1, SAT = 1
; Min Timing: 6+28 = 34 clks AARG = 0
; 6+22 = 28 clks
; PM: 14+51 = 65 DM: 7
;
;-----

```

FLO2424

```

    MOVLW      17h+EXPBIAS ; initialize exponent and add bias
    MOVWF     EXP
    CLRF      SIGN
    BTFSS    AARGB0,MSB      ; test sign
    GOTO     NRM3224
    COMF     AARGB2,F        ; if < 0, negate and set MSB in SIGN
    COMF     AARGB1,F
    COMF     AARGB0,F
    INCF     AARGB2,F
    BTFSC    _Z
    INCF     AARGB1,F
    BTFSC    _Z
    INCF     AARGB0,F
    BSF      SIGN,MSB

```

```

;*****

```

```

; Normalization routine
; Input: 32 bit unnormalized floating point number in AEXP, AARGB0, AARGB1,
; AARGB2, with sign in SIGN,MSB
; Use: CALL NRM3224
; Output: 24 bit normalized floating point number in AEXP, AARGB0, AARGB1
; Result: AARG <-- NORMALIZE( AARG )
; Max Timing: 21+6+7*8+7+4 = 94 clks RND = 0
; 21+6+7*8+20+4 = 103 clks RND = 1, SAT = 0
; 21+6+7*8+19+11 = 109 clks RND = 1, SAT = 1
; Min Timing: 22+6 = 28 clks AARG = 0
; 5+9+4+4 = 22 clks
; PM: 51 DM: 7
;
;-----

```

NRM3224

```

    CLRF      TEMP          ; clear exponent decrement
    MOVF     AARGB0,W       ; test if highbyte=0
    BTFSS    _Z
    GOTO     NORM3224
    MOVF     AARGB1,W       ; if so, shift 8 bits by move
    MOVWF    AARGB0
    MOVF     AARGB2,W
    MOVWF    AARGB1
    CLRF     AARGB2
    BSF      TEMP,3        ; increase decrement by 8
    MOVF     AARGB0,W       ; test if highbyte=0
    BTFSS    _Z
    GOTO     NORM3224
    MOVF     AARGB1,W       ; if so, shift 8 bits by move
    MOVWF    AARGB0
    CLRF     AARGB1
    BCF      TEMP,3        ; increase decrement by 8
    BSF      TEMP,4

```

```

        MOVF AARGB0,W          ; if highbyte=0, result=0
        BTFSC      _Z
        GOTO RES024
NORM3224
        MOVF TEMP,W
        SUBWF      EXP,F
        BTFSS     _Z
        BTFSS     _C
        GOTO SETFUN24
        BCF       _C          ; clear carry bit
NORM3224A
        BTFSC     AARGB0,MSB ; if MSB=1, normalization done
        GOTO NRMRND3224
        RLF       AARGB2,F   ; otherwise, shift left and
        RLF       AARGB1,F   ; decrement EXP
        RLF       AARGB0,F
        DECFSZ    EXP,F
        GOTO NORM3224A
        GOTO SETFUN24      ; underflow if EXP=0
NRMRND3224
        BTFSC     FPFLAGS,RND
        BTFSS     AARGB1,LSB
        GOTO FIXSIGN24
        BTFSS     AARGB2,MSB ; round if next bit is set
        GOTO FIXSIGN24
        INCF      AARGB1,F
        BTFSC     _Z
        INCF      AARGB0,F
        BTFSS     _Z          ; has rounding caused carryout?
        GOTO FIXSIGN24
        RRF       AARGB0,F   ; if so, right shift
        RRF       AARGB1,F
        INCF      EXP,F
        BTFSC     _Z          ; check for overflow
        GOTO SETFOV24
        GOTO FIXSIGN24
;
;*****
;*****
; Float to integer conversion
; Input: 24 bit floating point number in AEXP, AARGB0, AARGB1
; Use: CALL INT2416 or CALL INT24
; Output: 16 bit 2's complement integer right justified in AARGB0, AARGB1
; Result: AARG <-- INT( AARG )
; Max Timing: 29+6*6+5+13 = 83 clks RND = 0
; 29+6*6+5+19 = 89 clks RND = 1, SAT = 0
; 29+6*6+5+22 = 92 clks RND = 1, SAT = 1
; Min Timing: 18+5+7 = 30 clks
; PM: 63 DM: 6
;-----
INT2416
INT24
        MOVF     EXP,W          ; test for zero argument
        BTFSC     _Z
        RETLW     0x00
        MOVF     AARGB0,W      ; save sign in SIGN
        MOVWF    SIGN
        BSF      AARGB0,MSB    ; make MSB explicit
        MOVLW    EXPBIAS+0fh ; remove bias from EXP
        SUBWF    EXP,F

```

```

    BTFSS EXP,MSB
    GOTO SETIOV16
    COMF EXP,F
    INCF EXP,F
    MOVLW      8           ; do byte shift if EXP >= 8
    SUBWF      EXP,W
    BTFSS _C
    GOTO TSHIFT2416
    MOVWF      EXP
    RLF  AARGB1,F         ; rotate next bit for rounding
    MOVF AARGB0,W
    MOVWF      AARGB1
    CLRF AARGB0
    MOVLW      8           ;do byte shift if EXP >= 8
    SUBWF      EXP,W
    BTFSS _C
    GOTO TSHIFT2416
    MOVWF      EXP
    RLF  AARGB1,F         ; rotate next bit for rounding
    CLRF AARGB1
    MOVF EXP,W
    BTFSS _Z
    BCF  _C
    GOTO SSHIFT2416OK
TSHIFT2416
    MOVF EXP,W           ; shift completed if EXP = 0
    BTFSC _Z
    GOTO SHIFT2416OK
SHIFT2416
    BCF  _C
    RRF  AARGB0,F         ; right shift by EXP
    RRF  AARGB1,F
    DECFSZ EXP,F
    GOTO SHIFT2416
SHIFT2416OK
    BTFSC      FPFLAGS,RND
    BTFSS AARGB1,LSB
    GOTO INT2416OK
    BTFSS _C           ; round if next bit is set
    GOTO INT2416OK
    INCF AARGB1,F
    BTFSC _Z
    INCF AARGB0,F
    BTFSC      AARGB0,MSB ; test for overflow
    GOTO SETIOV16
INT2416OK
    BTFSS SIGN,MSB     ; if sign bit set, negate
    RETLW  0
    COMF AARGB1,F
    COMF AARGB0,F
    INCF AARGB1,F
    BTFSC _Z
    INCF AARGB0,F
    RETLW  0

SETIOV16
    BSF  FPFLAGS,IOV    ; set integer overflow flag
    BTFSS FPFLAGS,SAT   ; test for saturation
    RETLW  0xFF         ; return error code in WREG

```



```

        CLRF  AARGB0          ; saturate to largest two's
        BTFSS SIGN,MSB      ; complement 16 bit integer
        MOVLW    0xFF
        MOVWF    AARGB0     ; SIGN = 0, 0x 7F FF
        MOVWF    AARGB1     ; SIGN = 1, 0x 80 00
        RLF     SIGN,F
        RRF     AARGB0,F
        RETLW   0xFF        ; return error code in WREG
;*****
;*****
; Float to integer conversion
; Input: 24 bit floating point number in AEXP, AARGB0, AARGB1
; Use: CALL INT2424
; Output: 24 bit 2's complement integer right justified in AARGB0, AARGB1, AARGB2
; Result: AARG <-- INT( AARG )
; Max Timing: 41+6*7+6+16 = 105 clks RND = 0
; 41+6*7+6+24 = 113 clks RND = 1, SAT = 0
; 41+6*7+6+26 = 115 clks RND = 1, SAT = 1
; Min Timing: 5 clks
; PM: 82 DM: 6
;-----
INT2424
        CLRF  AARGB2
        MOVF  EXP,W          ; test for zero argument
        BTFSC    _Z
        RETLW   0x00
        MOVF  AARGB0,W      ; save sign in SIGN
        MOVWF    SIGN
        BSF   AARGB0,MSB    ; make MSB explicit
        MOVLW   EXPBIAS+17h ; remove bias from EXP
        SUBWF   EXP,F
        BTFSS  EXP,MSB
        GOTO   SETIOV24
        COMF   EXP,F
        INCF   EXP,F
        MOVLW   8           ; do byte shift if EXP >= 8
        SUBWF   EXP,W
        BTFSS  _C
        GOTO   TSHIFT2424
        MOVWF   EXP
        RLF    AARGB2,F     ; rotate next bit for rounding
        MOVF   AARGB1,W
        MOVWF   AARGB2
        MOVF   AARGB0,W
        MOVWF   AARGB1

        CLRF  AARGB0
        MOVLW   8           ; do another byte shift if EXP >= 8
        SUBWF   EXP,W
        BTFSS  _C
        GOTO   TSHIFT2424
        MOVWF   EXP
        RLF    AARGB2,F     ; rotate next bit for rounding
        MOVF   AARGB1,W
        MOVWF   AARGB2
        CLRF  AARGB1
        MOVLW   8           ; do another byte shift if EXP >= 8
        SUBWF   EXP,W
        BTFSS  _C

```

```

GOTO TSHIFT2424
MOVWF    EXP
RLF  AARGB2,F      ; rotate next bit for rounding
CLRF  AARGB2
MOVF  EXP,W
BTFSS _Z
BCF   _C
GOTO  SHIFT2424OK
TSHIFT2424
MOVF  EXP,W      ; shift completed if EXP = 0
BTFSC _Z
GOTO  SHIFT2424OK
SHIFT2424
BCF   _C
RRF  AARGB0,F    ; right shift by EXP
RRF  AARGB1,F
RRF  AARGB2,F
DECFSZ    EXP,F
GOTO  SHIFT2424
SHIFT2424OK
BTFSC    FPFLAGS,RND
BTFSS AARGB2,LSB
GOTO  INT2424OK
BTFSS _C
GOTO  INT2424OK
INCF  AARGB2,F
BTFSC _Z
INCF  AARGB1,F
BTFSC _Z
INCF  AARGB0,F
BTFSC    AARGB0,MSB ; test for overflow
GOTO  SETIOV24
INT2424OK
BTFSS SIGN,MSB    ; if sign bit set, negate
RETLW 0
COMF  AARGB0,F
COMF  AARGB1,F
COMF  AARGB2,F
INCF  AARGB2,F
BTFSC _Z
INCF  AARGB1,F
BTFSC _Z
INCF  AARGB0,F
RETLW 0
IRES024
CLRF  AARGB0    ; integer result equals zero
CLRF  AARGB1
CLRF  AARGB2
RETLW 0
SETIOV24
BSF   FPFLAGS,IOV ; set integer overflow flag

BTFSS FPFLAGS,SAT ; test for saturation
RETLW    0xFF      ; return error code in WREG
CLRF  AARGB0      ; saturate to largest two's
BTFSS SIGN,MSB   ; complement 24 bit integer
MOVLW    0xFF
MOVWF    AARGB0   ; SIGN = 0, 0x 7F FF FF
MOVWF    AARGB1   ; SIGN = 1, 0x 80 00 00

```

```

        MOVWF    AARGB2
        RLF     SIGN,F
        RRF     AARGB0,F
        RETLW   0xFF           ; return error code in WREG
;*****
;*****
; Floating Point Multiply
; Input: 24 bit floating point number in AEXP, AARGB0, AARGB1
; 24 bit floating point number in BEXP, BARGB0, BARGB1
; Use: CALL FPM24
; Output: 24 bit floating point product in AEXP, AARGB0, AARGB1
; Result: AARG <-- AARG * BARG
; Max Timing: 25+15*16+15+18 = 298 clks RND = 0
; 25+15*16+15+29 = 309 clks RND = 1, SAT = 0
; 25+15*16+15+33 = 313 clks RND = 1, SAT = 1
; Min Timing: 6+5 = 11 clks AARG * BARG = 0
; 24+15*11+14+15 = 218 clks
; PM: 80 DM: 11
;-----
FPM24
    MOVF AEXP,W           ; test for zero arguments
    BTFSS _Z
    MOVF BEXP,W
    BTFSC _Z
    GOTO RES024
M24BNE0
    MOVF AARGB0,W
    XORWF BARGB0,W
    MOVWF SIGN           ; save sign in SIGN
    MOVF BEXP,W
    ADDWF EXP,F
    MOVLW EXPBIAS-1
    BTFSS _C
    GOTO MTUN24
    SUBWF EXP,F
    BTFSC _C
    GOTO SETFOV24       ; set multiply overflow flag
    GOTO MOK24
MTUN24
    SUBWF EXP,F
    BTFSS _C
    GOTO SETFUN24
MOK24
    MOVF AARGB0,W

    MOVF AARGB0,W
    MOVWF AARGB2       ; move result to AARG
    MOVF AARGB1,W
    MOVWF AARGB3
    BSF AARGB2,MSB     ; make argument MSB's explicit
    BSF BARGB0,MSB
    BCF _C
    CLRF AARGB0        ; clear initial partial product
    CLRF AARGB1
    MOVLW 10h
    MOVWF TEMP        ; initialize counter
MLOOP24
    BTFSS AARGB3,LSB   ; test next bit
    GOTO MNOADD24

```

```

MADD24
    MOVF BARGB1,W
    ADDWF AARGB1,F
    MOVF BARGB0,W
    BTFSC _C
    INCFSZ BARGB0,W
    ADDWF AARGB0,F
MNOADD24
    RRF AARGB0,F
    RRF AARGB1,F
    RRF AARGB2,F
    RRF AARGB3,F
    BCF _C
    DECFSZ TEMP,F
    GOTO MLOOP24
    BTFSC AARGB0,MSB ; check for postnormalization
    GOTO MROUND24
    RLF AARGB2,F
    RLF AARGB1,F
    RLF AARGB0,F
    DECF EXP,F
MROUND24
    BTFSC FPFLAGS,RND
    BTFSS AARGB1,LSB
    GOTO MUL24OK
    BTFSS AARGB2,MSB ; round if next bit is set
    GOTO MUL24OK
    INCF AARGB1,F
    BTFSC _Z
    INCF AARGB0,F
    BTFSS _Z ; has rounding caused carryout?
    GOTO MUL24OK
    RRF AARGB0,F ; if so, right shift
    RRF AARGB1,F
    INCF EXP,F
    BTFSC _Z ; check for overflow
    GOTO SETFOV24
MUL24OK
    BTFSS SIGN,MSB
    BCF AARGB0,MSB ; clear explicit MSB if positive
    RETLW 0
SETFOV24
    BSF FPFLAGS,FOV ; set floating point underflag
    BTFSS FPFLAGS,SAT ; test for saturation
    RETLW 0xFF ; return error code in WREG
    MOVLW 0xFF
    MOVWF AEXP ; saturate to largest floating
    MOVWF AARGB0 ; point number = 0x FF 7F FF
    MOVWF AARGB1 ; modulo the appropriate sign bit

    RLF SIGN,F
    RRF AARGB0,F
    RETLW 0xFF ; return error code in WREG
;*****
;*****
; Floating Point Divide
; Input: 24 bit floating point dividend in AEXP, AARGB0, AARGB1
; 24 bit floating point divisor in BEXP, BARGB0, BARGB1
; Use: CALL FPD24

```

```

; Output: 24 bit floating point quotient in AEXP, AARGB0, AARGB1
; Result: AARG <-- AARG / BARG
; Max Timing: 32+13+15*26+25+12 = 472 clks RND = 0
; 32+13+15*26+25+34 = 494 clks RND = 1, SAT = 0
; 32+13+15*26+25+38 = 498 clks RND = 1, SAT = 1
; Min Timing: 7+5 = 12 clks
; PM: 120 DM: 11
;-----

```

```
FPD24
```

```

    MOVF BEXP,W           ; test for divide by zero
    BTFSC _Z
    GOTO SETFDZ24
    MOVF AEXP,W
    BTFSC _Z
    GOTO RES024

```

```
D24BNE0
```

```

    MOVF AARGB0,W
    XORWF BARGB0,W
    MOVWF SIGN           ; save sign in SIGN
    BSF AARGB0,MSB      ; make argument MSB's explicit
    BSF BARGB0,MSB

```

```
TALIGN24
```

```

    CLRF TEMP           ; clear align increment
    MOVF AARGB0,W
    MOVWF AARGB2       ; test for alignment
    MOVF AARGB1,W
    MOVWF AARGB3
    MOVF BARGB1,W
    SUBWF AARGB3, f
    MOVF BARGB0,W
    BTFSS _C
    INCFSZ BARGB0,W
    SUBWF AARGB2, f
    CLRF AARGB2
    CLRF AARGB3
    BTFSS _C
    GOTO DALIGN24OK
    BCF _C             ; align if necessary
    RRF AARGB0,F
    RRF AARGB1,F
    RRF AARGB2,F
    MOVLW 0x01

```

```

    MOVWF TEMP           ; save align increment

```

```
DALIGN24OK
```

```

    MOVF BEXP,W           ; compare AEXP and BEXP
    SUBWF EXP,F
    BTFSS _C
    GOTO ALTB24

```

```
AGEB24
```

```

    MOVLW EXPBIAS-1
    ADDWF TEMP,W
    ADDWF EXP,F
    BTFSC _C
    GOTO SETFOV24
    GOTO DARGOK24

```

```

    ; set overflow flag

```

```
ALTB24
```

```

    MOVLW EXPBIAS-1
    ADDWF TEMP,W

```

```

        ADDWF      EXP,F
        BTFSS     _C
        GOTO     SETFUN24      ; set underflow flag
DARGOK24
        MOVLW     10h          ; initialize counter
        MOVWF     TEMPB1
DLOOP24
        RLF      AARGB3,F      ; left shift
        RLF      AARGB2,F
        RLF      AARGB1,F
        RLF      AARGB0,F
        RLF      TEMP,F
        MOVF     BARGB1,W      ; subtract
        SUBWF     AARGB1,F
        MOVF     BARGB0,W
        BTFSS     _C
        INCFSZ   BARGB0,W
        SUBWF     AARGB0,F
        RLF      BARGB0,W
        IORWF    TEMP,F
        BTFSS    TEMP,LSB     ; test for restore
        GOTO     DREST24
        BSF      AARGB3,LSB
        GOTO     DOK24
DREST24
        MOVF     BARGB1,W      ; restore if necessary
        ADDWF     AARGB1,F
        MOVF     BARGB0,W
        BTFSC     _C
        INCF     BARGB0,W
        ADDWF     AARGB0,F
        BCF      AARGB3,LSB
DOK24
        DECFSZ   TEMPB1,F
        GOTO     DLOOP24
DROUND24
        BTFSC     FPFLAGS,RND
        BTFSS     AARGB3,LSB
        GOTO     DIV24OK
        BCF      _C
        RLF      AARGB1,F      ; compute next significant bit
        RLF      AARGB0,F      ; for rounding
        RLF      TEMP,F
        MOVF     BARGB1,W      ; subtract
        SUBWF     AARGB1,F

        MOVF     BARGB0,W
        BTFSS     _C
        INCFSZ   BARGB0,W
        SUBWF     AARGB0,F
        RLF      BARGB0,W
        IORWF    TEMP,W
        ANDLW    0x01
        ADDWF     AARGB3,F
        BTFSC     _C
        INCF     AARGB2,F
        BTFSS     _Z          ; test if rounding caused carryout
        GOTO     DIV24OK
        RRF      AARGB2,F

```

```

        RRF  AARGB3,F
        INCF EXP,F
        BTFSC  _Z          ; test for overflow
        GOTO SETFOV24
DIV24OK
        BTFSS  SIGN,MSB
        BCF  AARGB2,MSB   ; clear explicit MSB if positive
        MOVF  AARGB2,W
        MOVWF AARGB0      ; move result to AARG
        MOVF  AARGB3,W
        MOVWF AARGB1
        RETLW 0
SETFUN24
        BSF  FPFLAGS,FUN   ; set floating point underflag
        BTFSS FPFLAGS,SAT  ; test for saturation
        RETLW 0xFF         ; return error code in WREG
        MOVLW 0x01         ; saturate to smallest floating
        MOVWF AEXP         ; point number = 0x 01 00 00
        CLRF  AARGB0       ; modulo the appropriate sign bit
        CLRF  AARGB1
        RLF  SIGN,F
        RRF  AARGB0,F
        RETLW 0xFF         ; return error code in WREG
SETFDZ24
        BSF  FPFLAGS,FDZ   ; set divide by zero flag
        RETLW 0xFF
;*****
;*****
; Floating Point Subtract
; Input: 24 bit floating point number in AEXP, AARGB0, AARGB1
; 24 bit floating point number in BEXP, BARGB0, BARGB1
; Use: CALL FPS24
; Output: 24 bit floating point sum in AEXP, AARGB0, AARGB1
; Result: AARG <-- AARG - BARG
; Max Timing: 2+197 = 199 clks RND = 0
; 2+208 = 210 clks RND = 1, SAT = 0
; 2+213 = 215 clks RND = 1, SAT = 1
; Min Timing: 2+12 = 14 clks
; PM: 2+112 = 114 DM: 11
;-----
FPS24
        MOVLW 0x80
        XORWF BARGB0,F
;*****
; Floating Point Add
; Input: 24 bit floating point number in AEXP, AARGB0, AARGB1
; 24 bit floating point number in BEXP, BARGB0, BARGB1
; Use: CALL FPA24
; Output: 24 bit floating point sum in AEXP, AARGB0, AARGB1
; Result: AARG <-- AARG + BARG
; Max Timing: 25+28+6*6+5+31+72 = 197 clks RND = 0
; 25+28+6*6+5+42+72 = 208 clks RND = 1, SAT = 0
; 25+28+6*6+5+42+77 = 213 clks RND = 1, SAT = 1
; Min Timing: 8+4 = 12 clks
; PM: 112 DM: 11
;-----
FPA24
        MOVF  AARGB0,W      ; exclusive or of signs in TEMP
        XORWF BARGB0,W

```

```

MOVWF    TEMP
CLRF    AARGB2    ; clear extended byte
CLRF    BARGB2
MOVF    AEXP,W    ; use AARG if AEXP >= BEXP
SUBWF    BEXP,W
BTSS    _C
GOTO    USEA24
MOVF    BEXP,W    ; use BARG if AEXP < BEXP
MOVWF    AARGB4    ; therefore, swap AARG and BARG
MOVF    AEXP,W
MOVWF    BEXP
MOVF    AARGB4,W
MOVWF    AEXP
MOVF    BARGB0,W
MOVWF    AARGB4
MOVF    AARGB0,W
MOVWF    BARGB0
MOVF    AARGB4,W
MOVWF    AARGB0
MOVF    BARGB1,W
MOVWF    AARGB4
MOVF    AARGB1,W
MOVWF    BARGB1
MOVF    AARGB4,W
MOVWF    AARGB1
USEA24
MOVF    BEXP,W    ; return AARG if BARG = 0
BTFS    _Z
RETLW    0x00

MOVF    AARGB0,W
MOVWF    SIGN    ; save sign in SIGN
BSF    AARGB0,MSB ; make MSB's explicit
BSF    BARGB0,MSB
MOVF    BEXP,W    ; compute shift count in BEXP
SUBWF    AEXP,W
MOVWF    BEXP
BTFS    _Z
GOTO    ALIGNED24
MOVLW    8
SUBWF    BEXP,W
BTSS    _C    ; if BEXP >= 8, do byte shift
GOTO    ALIGNB24
MOVWF    BEXP
MOVF    BARGB1,W    ; keep for postnormalization
MOVWF    BARGB2
MOVF    BARGB0,W
MOVWF    BARGB1
CLRF    BARGB0
MOVLW    8
SUBWF    BEXP,W
BTSS    _C    ; if BEXP >= 8, BARG = 0 relative to AARG
GOTO    ALIGNB24
MOVF    SIGN,W
MOVWF    AARGB0
RETLW    0x00
ALIGNB24
MOVF    BEXP,W    ; already aligned if BEXP = 0
BTFS    _Z

```



```

        GOTO ALIGNED24
ALOOPB24
    BCF    _C                ; right shift by BEXP
    RRF    BARGB0,F
    RRF    BARGB1,F
    RRF    BARGB2,F
    DECFSZ    BEXP,F
    GOTO ALOOPB24
ALIGNED24
    BTFSS  TEMP,MSB        ; negate if signs opposite
    GOTO AOK24
    COMF  BARGB2,F
    COMF  BARGB1,F
    COMF  BARGB0,F
    INCF  BARGB2,F
    BTFSC  _Z
    INCF  BARGB1,F
    BTFSC  _Z
    INCF  BARGB0,F
AOK24
    MOVF  BARGB2,W
    ADDWF    AARGB2,F
    MOVF  BARGB1,W
    BTFSC  _C
    INCFSZ    BARGB1,W
    ADDWF    AARGB1,F
    MOVF  BARGB0,W
    BTFSC  _C
    INCFSZ    BARGB0,W
    ADDWF    AARGB0,F
    BTFSC    TEMP,MSB

    GOTO ACOMP24
    BTFSS  _C
    GOTO NRMRND3224
    RRF    AARGB0,F        ; shift right and increment EXP
    RRF    AARGB1,F
    RRF    AARGB2,F
    INCFSZ    AEXP,F
    GOTO NRMRND3224
    GOTO SETFOV24
ACOMP24
    BTFSC  _C
    GOTO NRM3224        ; normalize and fix sign
    COMF  AARGB2,F
    COMF  AARGB1,F        ; negate, toggle sign bit and
    COMF  AARGB0,F        ; then normalize
    INCF  AARGB2,F
    BTFSC  _Z
    INCF  AARGB1,F
    BTFSC  _Z
    INCF  AARGB0,F
    MOVLW    0x80
    XORWF    SIGN,F
    GOTO NRM24

```

```

*****
;
;=====
PID_ALGORITHM
  BANK2
  PAGE1

;pkt_l = 0.0 ;
;ekt l=0.0 ;
; /* Calculate error */
;ekt=rkt-ykt;
; /* Calculate I term */
;pkt=b*ekt+pkt_l;
; /* Calculate D term */
;qkt=c*(ekt-ekt l);
; /* Calculate PID output */
;ukt=pkt+a*ekt+qkt;
;if(ukt > MAX)
;{
;pkt=pkt_l;
;ukt=MAX;
;}
;else if(ukt < MIN)
;{
;pkt=pkt_l;
;ukt=MIN;
;}
; /* Send control to heater driver */
;ukt=ukt-3.0;
;ukt=ukt/4.0;
;control=ukt;
;CCPRIL=control;
; /* Save variables */
;pkt_l=pkt;
;ekt l=ekt;
; /* Wait for 5 seconds */
;=====

*****
;
;*****
;Açık çevrim fonksiyon parametreleri
;K: %pv/%output= 178/5=35,6
;Td: 8x1,5=12
;T1: (62-8)x1,5=81
;*****
;Ziegler-Nichols kontrolör ayar parametreleri
;Kp=1.2*T1/(K*Td) = 0,227
;Ti=2*Td = 24
;TD=0.5*Td = 6
;T <T1/4 = 5sn
;*****
;
; a=Kp = 0,227= 0x7C6872
; b=Kp*T/Ti = 0,047 =0x7A4083
; c=Kp*TD/T = 0,27 =0x7D0A3D
;*****
;KULLANILAN ALT PROGRAMLAR
;FPM24: MICROCHIP FLOAT ÇARPMA İŞLEMİ
;FPD24: MICROCHIP FLOAT BÖLME İŞLEMİ
;FPA24: MICROCHIP FLOAT TOPLAMA İŞLEMİ

```

```

;FPS24: MICROCHIP FLOAT ÇIKARMA İŞLEMİ
;FLO1624: INTEGER 16-BİT --> MICROCHIP FLOAT 24-BİT ÇEVİRİM
;INT2416: MICROCHIP 24-BIT FLOAT--> 16-BIT INTEGER ÇEVİRİM
;=====
;GERÇEK SICAKLIK->FLOAT ÇEVİRİM İŞLEMİ
;y(kt)
    CLRF AARGB0
    MOVF TMPR,W
    MOVWF AARGB1
    CALL FLO1624 ;16bit int-->24 bit float

    MOVF AEXP,W
    MOVWF Y_EXP
    MOVF AARGB0,W
    MOVWF Y_B0
    MOVF AARGB1,W ;y(kt)
    MOVWF Y_B1 ;SONUÇ Y_EXP:Y_B0:Y_B1
;-----
;SICAKLIK HATA DEĞERİ İŞLEMİ
;ekt=rkt-ykt
    BCF _MINUS ;0/1 (POZİTİF/NEGATİF)
    MOVF TMPR,W ;y(kT)
    SUBWF SETP,W ;r(kT)
    MOVWF E_B0 ;E_B0=R-Y (INTEGER)
    BTFSS CARRY
    BSF _MINUS ;NEGATİF SONUÇ
    BTFSS _MINUS ;NEGATİF SONUÇ MU?
    GOTO $+4 ;HAYIT ATLA
    COMF E_B0,W ;EVET
    ADDLW 01H
    MOVWF E_B0

    CLRF AARGB0
    MOVF E_B0,W
    MOVWF AARGB1 ;INTEGER INPUT=AARGB0:AARGB1
    CALL FLO1624 ;FLOAT OUTPUT: AEXP:AARGB0:AARGB1
    BTFSC _MINUS
    BSF AARGB0,MSB ;MICROCHIP NEGATİF

    MOVF AEXP,W
    MOVWF E_EXP
    MOVF AARGB0,W
    MOVWF E_B0
    MOVF AARGB1,W ;e(kT)=r(kT)-y(kT)
    MOVWF E_B1 ;SONUÇ E_EXP:E_B0:E_B1
;-----
;PROPERTIONAL CONTROL: ORANSAL KONTROL İŞLEMİ
;n(kT)=a*e(kT)
    MOVLW A_EXP
    MOVWF AEXP
    MOVLW A_B0
    MOVWF AARGB0
    MOVLW A_B1
    MOVWF AARGB1

    MOVF E_EXP,W
    MOVWF BEXP
    MOVF E_B0,W
    MOVWF BARGB0

```

```

MOVF E_B1,W
MOVWF BARGB1
CALL FPM24 ;a*e(kT)

MOVF AEXP,W
MOVWF P_EXP
MOVF AARGB0,W
MOVWF P_B0
MOVF AARGB1,W ;n(kT)=a*e(kT)
MOVWF P_B1 ;SONUÇ P_EXP:P_B0:P_B1
;-----
;INTEGRAL CONTROL: İNTEGRAL KONTROL İŞLEMİ
;p(kT)=b.e(kT)+p(kT-1)

MOVLW B_EXP
MOVWF AEXP
MOVLW B_B0
MOVWF AARGB0
MOVLW B_B1
MOVWF AARGB1

MOVF E_EXP,W
MOVWF BEXP
MOVF E_B0,W
MOVWF BARGB0
MOVF E_B1,W
MOVWF BARGB1
CALL FPM24 ;b.e(kT)

MOVF I_1_EXP,W
MOVWF BEXP
MOVF I_1_B0,W
MOVWF BARGB0
MOVF I_1_B1,W
MOVWF BARGB1
CALL FPA24 ;b.e(kT)+p(kT-1)

MOVF AEXP,W
MOVWF I_EXP
MOVF AARGB0,W
MOVWF I_B0
MOVF AARGB1,W ;p(kT)=b.e(kT)+p(kT-1)
MOVWF I_B1 ;SONUÇ I_EXP:I_B0:I_B1
;-----
;DERIVATIVE CONTROL: TÜREV KONTROL İŞLEMİ
;qkt=c*(ekt-ekt-1);
MOVF E_EXP,W
MOVWF AEXP
MOVF E_B0,W
MOVWF AARGB0
MOVF E_B1,W
MOVWF AARGB1

MOVF E_1_EXP,W
MOVWF BEXP
MOVF E_1_B0,W
MOVWF BARGB0
MOVF E_1_B1,W

```

```

MOVWF    BARGB1
CALL    FPS24                ;e(kT)-e(kT-1)

MOVLW    C_EXP
MOVWF    BEXP
MOVLW    C_B0
MOVWF    BARGB0
MOVLW    C_B1
MOVWF    BARGB1
CALL    FPM24                ;c*(e(kT)-e(kT-1))

MOVF    AEXP,W
MOVWF    D_EXP
MOVF    AARGB0,W
MOVWF    D_B0
MOVF    AARGB1,W            ;q(kT)=c*(e(kT)-e(kT-1))
MOVWF    D_B1                ;SONUÇ D_EXP:D_B0:D_B1
;-----
;PID CONTROLLER OUTPUT:PID KONTROLÖR ÇIKIŞI
;u(kT)=n(kT)+p(kT)+q(kT)=P+I+D
MOVF    P_EXP,W
MOVWF    AEXP
MOVF    P_B0,W
MOVWF    AARGB0
MOVF    P_B1,W
MOVWF    AARGB1

MOVF    I_EXP,W
MOVWF    BEXP
MOVF    I_B0,W
MOVWF    BARGB0
MOVF    I_B1,W
MOVWF    BARGB1
CALL    FPA24                ;PI = n(kT)+p(kT)

MOVF    D_EXP,W
MOVWF    BEXP
MOVF    D_B0,W
MOVWF    BARGB0
MOVF    D_B1,W
MOVWF    BARGB1
CALL    FPA24                ;PID = n(kT)+p(kT)+q(kT)

MOVF    AEXP,W
MOVWF    U_EXP
MOVF    AARGB0,W
MOVWF    U_B0
MOVF    AARGB1,W            ;u(kT)=n(kT)+p(kT)+q(kT)
MOVWF    U_B1                ;SONUÇ U_EXP:U_B0:U_B1
;-----
;MAX=255
;MIN=0

;if(ukt > MAX)
;{
;pkt=pkt_I;
;ukt=MAX;
;}

```

```

;else if(ukt < MIN)
;{
;pkt=pkt_I;
;ukt=MIN;
;}
;-----
          BTFSS U_B0,MSB      ;if (u(kT)<MIN)
          GOTO POS_RESULT ;POSITIVE

;-----
;else if(ukt < MIN)
;{
;pkt=pkt_I;
;ukt=MIN;
;}
;-----
NEG_RESULT                                ;u(kT)<MIN

          CLRF AARGB0
          MOVLW      MIN
          MOVWF     AARGB1      ;MIN-->FLOAT
          CALL  FLO1624

          MOVF AEXP,W
          MOVWF     U_EXP
          MOVF AARGB0,W
          MOVWF     U_B0
          MOVF AARGB1,W
          MOVWF     U_B1      ;u(kT)=MIN;

          MOVF I_1_EXP,W
          MOVWF     I_EXP
          MOVF I_1_B0,W
          MOVWF     I_B0
          MOVF I_1_B1,W
          MOVWF     I_B1      ;p(kT)=p(kT-1)

          GOTO PID_RESULT

;-----
;if(ukt > MAX)
;{
;pkt=pkt_I;
;ukt=MAX;
;}
;-----
POS_RESULT
          CALL  INT2416      ;FLOAT -->INT

          MOVF AARGB0,F      ;AARGB0<>0?
          BTFSS ZERO
          GOTO PMAX

          MOVF AARGB1,W
          SUBLW      MAX      ;NORMALDE GEREKSİZ
          BTFSC CARRY
          GOTO PID_RESULT      ;MIN<U(kT)<MAX

```

```

;-----
;u(kT)>MAX
PMAX
    CLRF  AARGB0
    MOVLW  MAX
    MOVWF  AARGB1      ;MAX-->FLOAT
    CALL  FLO1624

    MOVF  AEXP,W
    MOVWF  U_EXP
    MOVF  AARGB0,W
    MOVWF  U_B0
    MOVF  AARGB1,W
    MOVWF  U_B1      ;u(kT)=MAX;

    MOVF  I_1_EXP,W
    MOVWF  I_EXP
    MOVF  I_1_B0,W
    MOVWF  I_B0
    MOVF  I_1_B1,W
    MOVWF  I_B1      ;p(kT)=p(kT-1)
;=====
;=====
PID_RESULT

    MOVF  U_EXP,W
    MOVWF  AEXP
    MOVF  U_B0,W
    MOVWF  AARGB0
    MOVF  U_B1,W      ;u(kT)
    MOVWF  AARGB1
    CALL  INT2416

;    MOVF  AARGB1,W
;    CALL  SEND_PWM      ;OUTPUT->PWM
;=====
;SAVE VARIABLES
;Değişkenleri kaydet:
;pkt_1=pkt;
;ekt_1=ekt;
;=====
    MOVF  I_EXP,W
    MOVWF  I_1_EXP
    MOVF  I_B0,W
    MOVWF  I_1_B0
    MOVF  I_B1,W
    MOVWF  I_1_B1      ;p(kT-1)= p(kT)

    MOVF  E_EXP,W
    MOVWF  E_1_EXP
    MOVF  E_B0,W
    MOVWF  E_1_B0
    MOVF  E_B1,W
    MOVWF  E_1_B1      ;e(kT-1)= e(kT)

    MOVF  AARGB1,W      ;OUTPUT->PWM
    BANK_0

```

PAGE0
RETURN

;%%%%%%%%%%;
END
;%%%%%%%%%%;

ÖZGEÇMİŞ

Sinan SARIKAYA, 10.06.1981 tarihinde Ağrı'nın Taşlıçay ilçesinde doğdu. İlköğrenimini 1990 yılında Aş. Toklu Köyü İlkokulu'nda, orta öğrenimini 1993 yılında Iğdır Atatürk Lisesi Orta Bölümü'nde ve lise eğitimini 1996 yılında Taşlıçay Lisesi'nde tamamladı. 1997 yılında başladığı Sakarya Üniversitesi Elektrik-Elektronik Mühendisliği bölümünü 2001 yılında tamamladı. 2002 -2003 yıllarında Ağrı-Patnos 34. İç Güvenlik Tugayı'nda Muhabere Asteğmen rütbesiyle, Bilgi İşlem Merkezi'nde BİM sorumlusu olarak çalıştı ve askerlik hizmetini tamamladı. Eylül 2003-Ocak 2005 tarihleri arasında İstanbul Sefaköy'de Gül Kablo ve Elektronik Ltd. Şti.'nde güvenlik sistemleri alanında Ar-Ge mühendisi olarak çalıştı. Şubat 2005 tarihinde Sakarya'nın Hendek İlçesi'nde kurulu AIRFEL ISITMA-SOĞUTMA SİSTEMLERİ SAN. TİC. A.Ş. Panel Radyatör Fabrikası'nda üretim mühendisi olarak yeni görevine başladı. Haziran 2007 tarihinde, aynı firmada Üretim Bakım Şefi görevine atandı ve halen aynı görevde çalışmaya devam etmektedir. 2005 yılında Sakarya Üniversitesi Fen Bilimleri Enstitüsü Elektrik ve Elektronik Mühendisliği EABD Elektronik EBD'de yüksek lisans eğitimine başlamıştır.