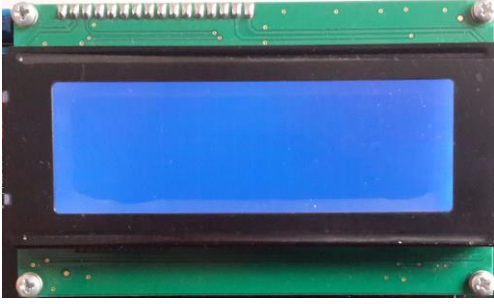


LCD EKRANIN SÜRÜLMESİ



Şekil 1: Deneyde kullanılacak olan LCD 2x20



Şekil 2: Grafik LCD

LCD bir görüntüleme teknolojisidir. Bu teknolojiyi kullanan cihazlar ise LCD gösterge olarak adlandırılmaktadır. 7-Segment (7 Parçalı Gösterge) göstergelerin fazla akım çekmesi ve kullanım zorluğu nedeniyle, son yıllarda LCD göstergelerin kullanımı popüler hale gelmiştir.

Bu deneyde kullanacağımız LCD (ve çoğu LCD), aslında sadece bir göstergeden ibaret değildir. Bünyesinde,

1. Kendisine ait bir **mikroişlemcisi**,
2. RAM ve ROM'LARI,
3. Giriş çıkışları,

olan bir cihazdır. Yani bir nevi **bütünleşik mikrodenetleyici+LCD** cihazdır.

LCD'ler 1 satırdan 4 satıra kadar, 16 karakterden 80 karaktere kadar ve 5X7, 5X10 nokta font gibi değişik ölçülerde üretilip satılmaktadır. Bazılarında ise tüm ekran tek bir karakter gibi yapılandırılmıştır, bu türlerine grafik LCD gösterge adı verilmektedir. LCD gösterge ile iletişim, TTL standardında 4 veya 8 veri hattı ile yapılır. 4 bit iletişim G/Ç hatlarının başka işler(görevler, amaçlar) için kullanımını kolaylaştırırken, iletişim süresini iki kat uzatmaktadır.

2.2. LCD Hafıza Haritası (Memory Map)

LCD göstergeler üzerinde kullanılan denetleyiciler, Hitachi firması tarafından üretilen **HD44780U** mikrodenetleyicisidir ve bu mikrodenetleyici standart bir hale gelmiştir.

LCD'lerin mimarisinde 3 adet hafıza yapısı bulunmaktadır. Bunlar DDRAM, CGROM ve CGRAM'dir.

DDRAM: LCD göstergeler, 40 karaktere ve 4 satıra kadar değişik seçenekler sunar. LCD göstergeler 80 adet karaktere kadar kodu saklayabilmek için dâhili bir RAM bulundurlar. Bu RAM'a Gösterge Veri RAM'i(Display Data RAM-DDRAM) denir. Örneğin bir satırında 16 karakteri olan iki satırlık bir göstergelyi, her birinde 40 karakteri olan 2 sanal satır olarak düşünebiliriz. 40 karakterlik bir satır bulunmaktadır ancak biz onu 16 karakterlik bir pencere ile görebilmekteyiz. Sanal satırdaki diğer karakterleri görebilmek için gösterge kaydırılmalıdır. Örneğin, bu göstergenin birinci satırına aşağıdaki 40 karakterli diziyi yazalım.

9876543210QWERTYUIOPLKJHGFDSA ZXCVBNMsedn

Göstergede sadece ilk 16 tanesi gözükecektir.

9876543210QWERTY

Bu gösterimde, aşağıda bahsedileceği gibi Entry (Giriş) moduna bağlı olarak ekran kayabilir veya kaymayabilir. Burada ekran kaydırılmamıştır. "U" karakterini görüntülemek istediğimizde, aşağıdaki gibi ekran bir karakter sağa kayacak ve "9" karakteri gizlenecektir.

876543210QWERTYU

CGROM: LCD göstergede önceden programlanmış veya kullanıcı tarafından karakterleri tanımlanan gösterebilmektedir. LCD kontrolörü, 192 adet karakter içeren bir Karakter Üretici ROM'a (Character Generator ROM –CGROM) sahiptir. Karakterler belirli kodlarla seçilirler. Bu karakterlerden 96 tanesi ASCII karakter (ASCII kodlarla seçilir) , 64 adeti japon karakterleri (Kana Alfabeti) ve 32 adeti Yunan harfleri gibi özel karakterlerdir.

CGRAM: LCD kontrolörü aynı zamanda, Karakter Üretici RAM (Character Generator RAM – CGRAM) olarak adlandırılan ve kullanıcı tarafından tanımlanabilen 8 karakter içerebilen bir hafızaya sahiptir. Bu karakter kullanılmadan önce tanımlanmalı, CGRAM'a yüklenmeli ve gösterilmek için gereken yerlerde çağrılmalıdır.

ASCII KOD TABLOSU: **CGROM** hafızasında dâhili olarak bulunur. Latin alfabesi üzerine kurulu 7 bitlik bir karakter setidir. ASCII'de 33 tane basılmayan (ekranda görülmeyen) kontrol karakteri ve 95 tane basılan (ekranda görülen) karakter bulunur. Kontrol karakterleri metnin akışını kontrol eden, ekranda çıkmayan karakterlerdir. Basılan karakterler ise ekranda görünen, okuduğumuz metni oluşturan karakterlerdir.

Tablo 1: ASCII Kod Tablosu

*	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	TAB	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

Tablo 1'den bir karakterin ASCII kodunu bulmak için önce karakterin bulunduğu satır numarası sonrada sütun numarasına bakılır. Aşağıda örnek olarak çeşitli karakterlerin ASCII kodları verilmiştir.

Karakter	ASCII Kodu
A	41'H
a	61'H
5	35'H
=	3D'H

LCD Ekranın Adreslenmesi

(LCD Göstergeye bu dokümanda bundan sonra kısaca LCD denilecektir.)

LCD'lerde aşağıda gösterildiği gibi her karakterin ayrı bir adresi vardır. Aşağıdaki resimlerde 2x16 ve 4x20 karakter boyutundaki LCD'lerin her karakter için belirlenen adresleri verilmiştir.

2x16									4x20											
	80	81	8E		8F	80	81	92	93
	C0	C1	CE		CF	C0	C1	D2	D3
											94	95	A6	A7
										D4	D5	E6	E7	

Değişik firmalar tarafından üretilen LCD'lerde, LCD ekranın sürücü devresini de içerir. Dolayısıyla ilave devre kurmadan doğrudan mikrokontrolör ile LCD modüller sürülebilir. LCD'ler farklı firmalar tarafından üretilmelerine rağmen erişim protokolleri çoğunlukla aynıdır. LCD'lerin 3 kontrol hattı (RS, R/W, E) ve 8 veri hattı (DB0...DB7) vardır.

2.2. LCD Bağlantı Uçları

Şekil 2’de görüldüğü gibi günümüzde üretilen LCD panellerin çoğunda tek sıra halinde 16 pin bulunur. Bazı LCD 'lerde kontrol için kullanılan 14 pin 2 adet 7’li sıra halinde de bulunabilir. LCD bağlantı uçları Tablo 2’de verilmiştir. Bağlantı uçlarını besleme, denetim ve veri olmak üzere üç grupta inceleyebiliriz.

Tablo 2: Pinlerin Görevleri

Pinlerin Görevleri					
No	Sembol	Fonksiyon	No	Sembol	Fonksiyon
1	Vss	GND	9	DB2	Komut veya Veri Yolu
2	Vdd	+5V	10	DB3	Komut veya Veri Yolu
3	Vee	LCD Sürmek için	11	DB4	Veri Yolu
4	RS	Fonksiyon Seçimi	12	DB5	Veri Yolu
5	R/W	Okuma/Yazma	13	DB6	Veri Yolu
6	E	Sinyal Aktifleş.	14	DB7	Veri Yolu
7	DB0	Komut veya Veri Yolu	15	LEDA	Işık +5v
8	DB1	Komut veya Veri Yolu	16	LEDA	Işık 0V

1.1.1. Besleme Gerilimleri

HD44780U standardında besleme ile ilgili üç uç yer almaktadır. Bunlar **Vcc**, **Vee (V0)**, **Vss (GND)**’dir. Vss ve Vcc standart TTL gerilimi 0 ve 5 Volttur. Vee ise ekranın parlaklığını belirleyen bir gerilimdir ve değeri en çok Vcc’dir.

1.1.2. Kontrol ve Veri Pinleri

LCD’yi kontrol etmek amaçlı üç uç yer almaktadır. Bunlar **RS**, **R/W**, **E** uçlarıdır. Ayrıca 8 tane de veri ucu bulunmaktadır. Bunlar;

Tablo 3: Kontrol Uçları

Sembol	Görevi
RS	LCD’ye komut mu yoksa veri mi gönderileceğini belirler. LCD ekrana veri aktarılacaksa RS= 1 , komut gönderilecekse RS= 0 LCD ekranı silme, kursör on/off, kursör başa dön, yazma başlangıç adresinin belirtilmesi gibi işlemler komut olarak adlandırılır. LCD’lerde kullanılan komutlar ve ilgili komutlar için pin değerleri aşağıda tablo halinde verilmiştir. LCD ekrana yazılan (örneğin “SAU”, “12+3=15”, vb.) değerler ise veri olarak adlandırılır.
R/W	Lcd den okuma mı yoksa lcd ye yazma yapılacağını belirler. Lojik R/W=1 seviyesi LCD’lerden okuma, lojik R/W=0 ise LCD’ye yazma işlemini gösterir. Deneylerde LCD’den okuma işlemi yapılmayacağı için bu pin Şekil 1 de gösterildiği gibi donanımsal olarak GND pinine bağlanarak Lojik 0 seviyesinde tutulmuştur.
E	Enable (Aktifleştirme) ucu LCD ve pinler arasındaki gerçek veri alışverişini sağlayan bacadır. Bu girişi mikrodenetleyiciye program aracılığıyla tanıttıktan sonra mikrodenetleyici kendisi veri gönderileceği zaman bu bacağa enable (aktifleştir) darbesi gönderir. Yani bu uca 0-1-0 darbesi üretilir.
DB0-DB7	Data hattı olan bu pinler doğrudan mikrodenetleyicinin bir portuna bağlanır. Veri 4 ya da 8 bitlik veri yolu ile gönderilebilir.

1.2. LCD Komut Tipleri ve Zaman Çizelgesi

Bir LCD işlemi, ya kontrol ya da veri işlemidir. Kontrol işlemleri, ya LCD'ye gönderilen komutlardır, ya da LCD'den okunan bir hafıza adresidir (kaydedici-register). RS (Register Select) ucu lojik 0'a çekilirse yapılacak işlem, kontrol işlemidir. Bütün komutlar ve sinyaller, harici bir mikrodenetleyici, bilgisayar vb. tarafından üretilir. Komutlar LCD'ye 8-bit olarak yazılır. Bunun için R/W lojik 0'da tutulmalıdır. Bu uç lojik 1'e çekilirse, LCD'den veri okunabilir. Komut çeşitleri ve ayar bitleri Tablo 4'te verilmiştir.

Tablo 4: LCD Komut Gönderimi

KOMUT	KOD										İŞLEM SÜRESİ
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
Ekranı Sil	0	0	0	0	0	0	0	0	0	1	1,64 ms
Kursör basa don	0	0	0	0	0	0	0	0	1	*	1,64 ms
Giriş kipini seç	0	0	0	0	0	0	0	1	I/D	S	40µs
Ekran aç/kapa	0	0	0	0	0	0	1	D	C	B	40µs
Kursör kaydır	0	0	0	0	0	1	S/C	R/L	*	*	40µs
Fonksiyon seç	0	0	0	0	1	DL	N	F	*	*	40µs
Meşgul bayrağını oku	0	1	BF	DDRAM ADRESİ							0µs
Veri yaz	1	0	VERİ YAZ							40µs	
Veri oku	1	1	VERİ OKU							40µs	

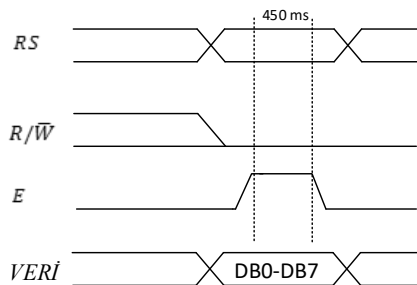
Komut seti; (* : Aldığı değer (1-0) önemsiz, DDRAM: Ekran Veri Belleği) Not: Tabloda belirtildiği gibi LCD modüle yazma işlemi minimum 40µs sürmektedir. Dolayısı ile ard arda yapılan yazma işlemlerinde bir önceki verinin yazılabilmesi için en az belirtilen süre kadar beklenmelidir.

Tablo 5: Kontrol bitlerinin görevleri

Kod	AÇIKLAMA	
I/D	0 = Her yazma işleminden sonra kursör pozisyonunu azalt	1 = Her yazma işleminden sonra kursör pozisyonunu arttır
S	0 = Ekran kaydırma modu kapalı	1 = Ekran kaydırma modu açık
D	0 = Ekran kapalı	1 = Ekran açık
C	0 = Kursör kapalı	1 = Kursör açık
B	0 = Kursör blink kapalı	1 = Kursör blink açık
S/C	0 = Kursör taşınması gerekir. Manuel.	1 = Kursör kaydır
R/L	0 = Sola kaydır	1 = Sağa kaydır
DL	0 = Veri hattı 4 bit	1 = Veri hattı 8 bit
N	0 = 1 satır	1 = 2 satır
F	0 = 5x7 pixel	1 = 5x10 pixel
BF	0 = Komut kabul edebilir	1 = LCD Meşgul

1.2.1. Komut/Veri Gönderim Zaman Çizelgesi

Tablo 3'teki LCD kontrol pinlerinin zamanlama çizelgesi aşağıda verilmiştir.



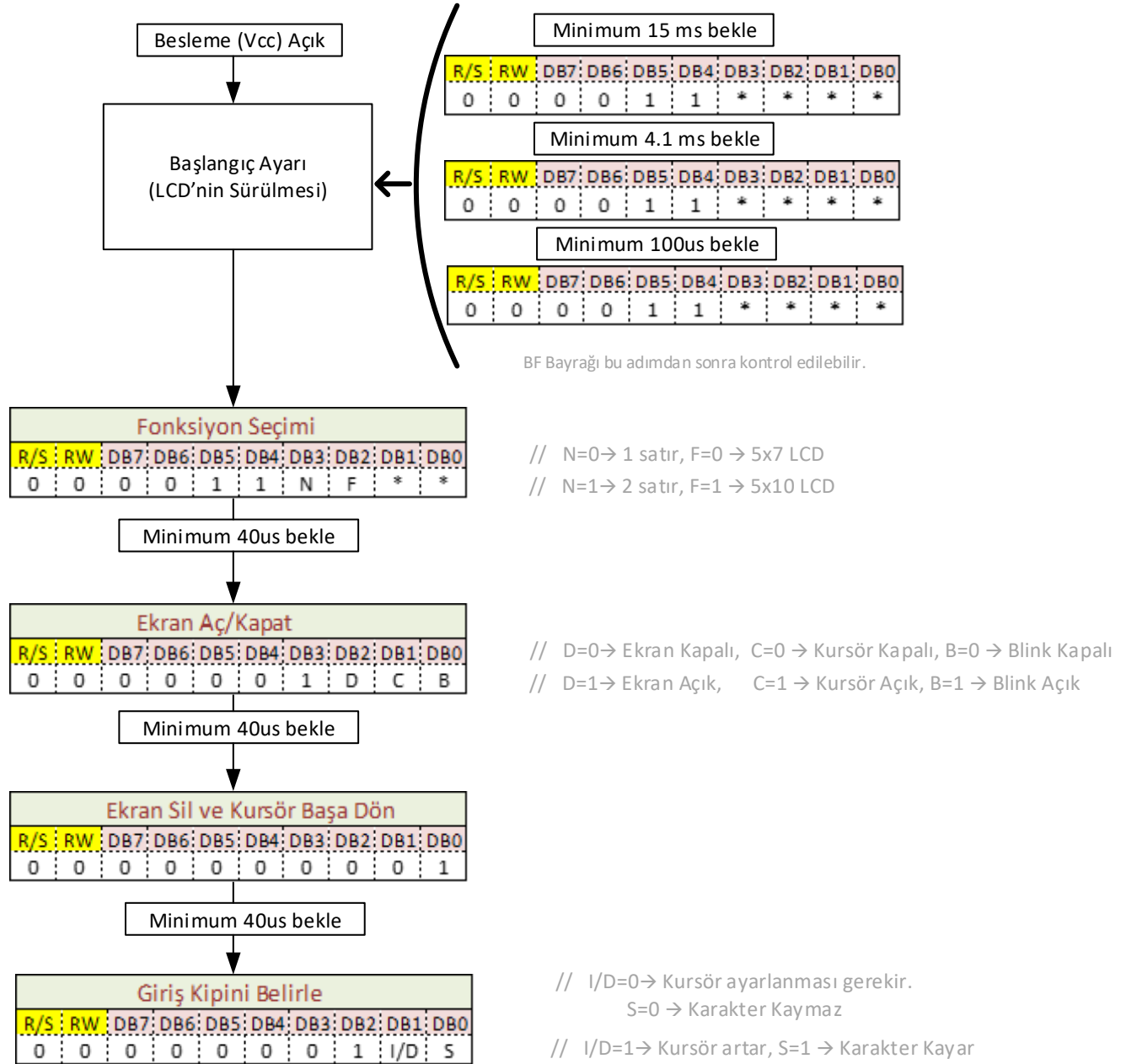
Şekil 3: LCD'ye veri yazmak için kontrol pinlerinin zamanlanması

2.3. LCD'nin Başlangıç Ayarlarının Yapılması

LCD'ye ilk enerji verilmesinden ardından, hazırlanan program ile arka arkaya 3 defa 30h komutu LCD'ye gönderilir. Burada bu komutun düşük dörtlüsü (son dört biti) ihmal edilir; DB7 ve DB6=0, DB5 ve DB4=1 olarak atanır.

Yukarıdaki başlangıç ayarının ardından LCD için **fonksiyon belirleme** işlemi Tablo 4 ve Tablo 5 ile gerçekleştirilir yani veri yolunun büyüklüğü (4 bit veya 8 bit), göstergedeki satır sayısı (1-2-3-4) ve font büyüklüğü (5x7 veya 5x10 gibi) belirlenir. Ardından sırası önemli olmamak üzere giriş kipi, ekranın, kursörün, blink'in (imlecin) açık/kapalı ayarları yapılır. Giriş kipi; her karakter okuma veya yazma işlemini takiben imlecin veya göstergenin yerini belirler.

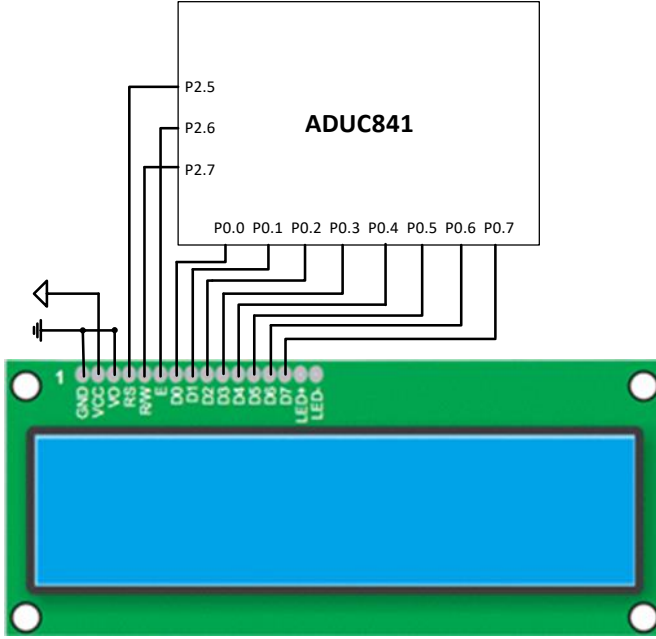
LCD ilk başlangıç ayarı aşağıdaki şekilde verilmiştir.



Şekil 4: LCD Başlangıç Ayarı

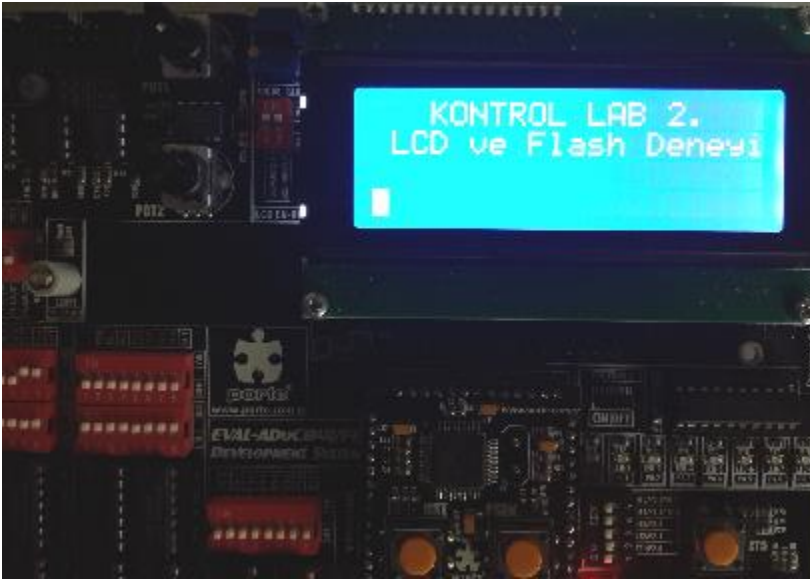
2.4. Deney Setindeki Aduc841 İle Lcd Bağlantı Şeması Ve Pinleri

ADUC841 deney setindeki mikrodenetleyici ile LCD arasında aşağıdaki şekilde bağlantılar gerçekleştirilmiştir.



UYGULAMA 1:

Deneyde aşağıdaki şekilde gözüktüğü gibi LCD ekranın ilk satırına 4. koldan itibaren “KONTROL LAB. 2” ikinci satırına ise 2. koldan itibaren “LCD ve Flash Deneyi” yazdırılacaktır.



Uygulamalara Ait Kodlar

Uygulama 1:

```
#include<aduc841.h>
```

```
LRS EQU P2.5  
LEE EQU P2.6  
LCD EQU P0  
RW EQU P2.7
```

```
ORG 00H  
sjmp BASLA
```

```
BASLA:
```

```
clr RW ; LCD'den okuma modunu iptal et.  
mov r0, #07fH ; Temizleme  
temiz: mov @r0, #00H ; Temizleme  
djnz r0, temiz ; Temizleme  
  
lcall lcd_ayar ; LCD'nin Başlangıç Ayarları Yapılıyor.  
mov dptr, #Data1
```

```
; _____ 1. Satir _____  
clr LRS ;komut girişi  
mov a, #83H ;birinci satır  
lcall yaz ;başlangic adresi  
setb LRS ;veri girişi
```

```
mov r0, #00H  
str1: mov a, r0  
movc a, @a+dptr  
cjne a, #'0', go1  
sjmp str2  
go1: lcall yaz  
inc r0  
sjmp str1
```

```
; _____ 2. Satir _____  
str2: clr LRS ;komut girişi  
mov a,#0c1H ;ikinci satır  
lcall yaz ;başlangic adresi  
setb LRS ;veri girişi
```

```
mov dptr, #Data2  
mov r0, #00H  
go: mov a, r0  
movc a, @a+dptr  
cjne a, #'0', go2  
sjmp dur  
go2: lcall yaz  
inc r0  
sjmp go  
DUR: SJMP DUR
```

; _____ LCD Ayarı _____

lcd_ayar:

```
    clr    LEE
    clr    LRS                ;komut girişı
;-----
; Minimum 15 ms bekleme
    lcall  gecik
    lcall  gecik
    lcall  gecik
;-----
;-----
; LCD'nin Sürülmesi için Gerekli Kod Parçacığı
; RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0
; 0 0 0 0 1 1 * * * * => 30H
    mov    a,    #30H
    lcall  yaz
    lcall  gecik
    mov    a,    #30H
    lcall  yaz
    lcall  gecik
    mov    a,    #30H
    lcall  yaz
;-----
;-----
; LCD Ayarları
```

```
; RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0
; 0 0 0 0 1 1 N F * *
; 0 0 0 0 1 1 1 1 * * => 3CH => N=1 için 2 satır, F=1 için 5x10 LCD
    mov    a,    #3cH        ;2 satır, 5x10 pixel
    lcall  yaz
```

```
; RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0
; 0 0 0 0 0 0 1 D C B
; 0 0 0 0 0 0 1 1 1 1 => OFH => D=1 Ekran Açık, C=1 Kursör Açık, B=1 Blink Açık.
    mov    a,    #0fh        ;Ekran, Kursör ve Blink açık
    lcall  yaz
    mov    a,    #01H        ;Ekranı sil, kursör başa dön.
    lcall  yaz
```

```
; RS RW DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0
; 0 0 0 0 0 0 0 1 I/D S
; 0 0 0 0 0 0 0 1 1 0 => 06H => I/D=1 Kursör Arttır, S=0 Ekran Kaydırma Kapalı
    mov    a,    #06H        ;Giriş Modu => Kursör pozisyonunu artır
    lcall  yaz
    ret
```

; _____ LCD Ayarı Bitti _____

```
yaz:  setb    LEE
      lcall  gecik
      mov    LCD,a    ; LCD = PortO (P0)
      clr    LEE
      ret
```

; _____

```
gecik: mov    r3,    #4fh
w2:    mov    r4,    #0ffh
w1:    djnz   r4,    w1
      djnz   r3,    w2
      ret
```


;

```
Data1: DB 'MIKRO LAB.0'  
Data2: DB 'SAU EEM BOLUMU0'  
end
```