

C Programlama Diliyle

8051

MİKRODENETLEYİCİ  
UYGULAMALARI

ÖZCERİT, ÇAKIROĞLU ve BAYILMIŞ

PAPATYA YAYINCILIK EĞİTİM

İstanbul, Ankara, İzmir ve Adana

Papatya Yayıncılık Eğitim

© Papatya Yayıncılık Eğitim - Mart 2008  
BİLGİSAYAR SİS. SAN. VE TİC. A.Ş.  
İnönü Cad. Hacıhanım Sok. 10/6, 80090, Gümüşsuyu/İstanbul

Tel : (212) 245 37 40 (Merkez)  
Faks : (212) 245 37 41  
e-Posta : bilgi@papatya.gen.tr  
Web : www.papatya.gen.tr  
www.papatya.info.tr  
Dağıtım : İstanbul : (212) 527 52 96  
Adana : (322) 432 00 73

C ile 8051 Mikrodenetleyici Uygulamaları - Dr. Ahmet Turan ÖZCERİT, Murat ÇAKIROĞLU ve Cüneyt BAYILMIŞ

2. Basım Mart 2008

Yayın Danışmanı : Dr. Rifat ÇÖLKESEN  
Akademik Danışman : Dr. Cengiz UĞURKAYA  
Türk Dili : Necdet AVCI  
Üretim : Ziya ÇÖLKESEN  
Sayfa Düzenleme : Papatya & Kelebek Tasarım  
Kapak Tasarım : Papatya & Kelebek Tasarım  
Basım ve Ciltleme : Altan Basım San. Ltd. Şti. / İstanbul

©Bu kitabın her türlü yayın hakkı Papatya Yayıncılık Eğitim A.Ş.'ye aittir. Yayınevinden yazılı izin alınmaksızın alıntı yapılamaz, kısmen veya tamamen hiçbir şekil ve teknikle ÇOĞALTILAMAZ, BASILAMAZ, YAYIMLANAMAZ. Kitabın, tamamı veya bir kısmının fotokopi makinesi, ofset gibi teknikle çoğaltılması, hem çoğaltan hem de bulunduranlar için yasadışı bir davranıştır.

Lütfen kitabımızın fotokopi yöntemiyle çoğaltılmasına engel olunuz. Fotokopi hızsızlıktır.

Özcerit, Ahmet Turan.

C ile 8051 Mikrodenetleyici Uygulamaları / Ahmet Turan ÖZCERİT, Murat ÇAKIROĞLU ve Cüneyt BAYILMIŞ  
- İstanbul: Papatya Yayıncılık Eğitim, 2008.

XII, 416 s.; 24 cm.

Kaynakça ve dizin var.

ISBN 975-6797-64-9

1. Mikroişlemci .2. Kontrol 3.Endüstriyel 4. Programlama 5.Assembly I. Title.

Kitabımızı,  
bizlere büyük emekleri geçmiş olan  
Mehmet-Emine ÇAKIROĞLU, Sevim BAYILMIŞ  
ve Merhum Mustafa-Hafize ÖZCERİT ve Merhum Fevzi BAYILMIŞ'ın anısına,  
ayrıca bizleri her zaman destekleyen sevgili eşlerimiz  
Esra ÖZCERİT ve Hatice ÇAKIROĞLU'na  
İthaf Ediyoruz.

## TEŞEKKÜR

Kitabımızın hazırlanması sürecinde bizlere yol gösteren ve önerileriyle daha da kapsamlı olmasını sağlayan Prof.Dr.Hüseyin EKİZ'e, Doç.Dr.Akif KUTLU'ya ve Doç.Dr.İsmail ERTÜRK'e;

Maddi manevi desteklerini hiç bir zaman esirgemeyen değerli meslektaşlarımız ve dostlarımız Özdemir ÇETİN'e ve Adnan SONDAŞ'a;

Görüş ve düşüncelerinden yararlandığımız meslektaşlarımız Sedat ATMACA'ya ve Ahmet KARACA'ya;

Kaynak CD ve [www.8051turk.com](http://www.8051turk.com) sitesinin hazırlanmasında fedakarca çalışan Sakarya Üniversitesi'nden Ali DURDU'ya;

Video derslerin hazırlanmasında büyük emeği geçen Kocaeli Üniversitesi'nden Çağlar KIYMET'e;

Keil µVision ve ISIS/Proteus simülasyon programlarının deneme sürümlerini kaynak CD'ye koymamıza izin veren ve teknik desteklerini esirgemeyen Keil Software ve Labcenter firmalarına teşekkür ederiz.

Herşeyin daha da güzel olması dileğiyle...

## İÇİNDEKİLER

<b>ÖNSÖZ</b>	<b>8</b>
<b>Bölüm 1: MİKRODENETLEYİCİLERE GENEL BAKIŞ</b>	<b>13</b>
1.1. Giriş	14
1.2. Mikrodenetleyici Mimari Özellikleri	17
1.3. Mikrodenetleyici Seçiminde Dikkat Edilecek Ölçütler	20
1.4. Mikrodenetleyicinin Rakipleri	22
1.5. Özet	23
1.6. Sorular	24
<b>Bölüm 2: 8051 MİKRODENETLEYİCİSİNE GİRİŞ</b>	<b>25</b>
2.1. MCS-51 Ailesi	26
2.2. 8051 Mikrodenetleyicisinin Genel Yapısı	26
2.3. 8051 Mikrodenetleyicisinin Uç Fonksiyonları	32
2.4. Ayrıntılı Port Yapısı	36
2.5. 8051 Mikrodenetleyicisinin Ayrıntılı İç Mimarisi	38
2.6. Özet	38
2.7. Sorular	39
<b>Bölüm 3: 8051'İN BELLEK ORGANİZASYONU</b>	<b>41</b>
3.1. Giriş	42
3.2. Program (Kod) Belleği	42
3.3. Veri Belleği	44
3.4. Bellek Organizasyonu Tasarımı ve Adres Çözümleme	57
3.5. Özet	69
3.6. Sorular	69
<b>Bölüm 4: ADRESLEME MODLARI VE KOMUT KÜMESİ</b>	<b>71</b>
4.1. Giriş	72
4.2. Assembly Dili ve Özellikleri	72
4.3. Adresleme Yöntemleri	75
4.4. 8051 Komut Türleri	83
4.5. Özet	101
4.6. Sorular	102

<b>Bölüm 5: ZAMANLAYICILAR / SAYICILAR</b>	<b>105</b>
5.1. Giriş	106
5.2. Zamanlayıcı/Sayıcı Saklayıcıları	107
5.3. Zamanlayıcı/Sayıcı Çalışma Modları	111
5.4. Zaman ya da Durum Sayma Seçimi	117
5.5. Zamanlayıcıların Kullanımı ve Örnekler	117
5.6. Zamanlayıcı/Sayıcı 2	123
5.7. Özet	129
5.8. Sorular	129
<b>Bölüm 6: SERİ PORT İŞLEMLERİ</b>	<b>131</b>
6.1. Giriş	132
6.2. Seri Port Kontrol Saklayıcısı (SCON)	133
6.3. Seri Port Çalışma Modları	134
6.4. Seri Port Saklayıcılarının Kurulumu ve Saklayıcılara Erişim	137
6.5. Çok İşlemcili Haberleşme	140
6.6. Seri Port Baud Hızlarının Ayarlanması	141
6.7. Seri Port Uygulama Örnekleri	143
6.8. Özet	146
6.9. Sorular	146
<b>Bölüm 7: KESMELER (INTERRUPTS)</b>	<b>149</b>
7.1. Giriş	150
7.2. 8051 Kesme Düzenlemesi	151
7.3. Kesme Vektör Adresleri	156
7.4. Kesmelerin Çalışması	158
7.5. Kesme Örnekleri	158
7.6. Özet	161
7.7. Sorular	162
<b>Bölüm 8: 8051 YENİ TEKNOLOJİLER</b>	<b>163</b>
8.1. Giriş	164
8.2. AT89C51RC2 Mikrodenetleyicisi	165
8.2/a. Hızlı Çalışma Modu	166
8.3. Çift Veri İşaretçisi (Dual DPTR)	169
8.4. Genişletilmiş RAM (XRAM)	171
8.5. Gelişmiş Sayıcı Birimleri (PCA)	174
8.6. Tuş Takımı Arayüzü	185
8.7. Seri Port Arayüzü (SPI)	189
8.8. Gelişmiş Kesme Birimi	199
8.9. Donanımsal Kısır-Döngü Sayacı (Watchdog Timer)	204

8.10. Dahili Emülasyon (ONCE-On Chip Emulation)	206
8.11. EEPROM Veri Belleği	207
8.12. Flash Bellek ve ISP (Sistem Üzerinde Programlama)	208
8.13. Saklayıcılar	209
8.14. Özet	211
8.15. Sorular	212
<b>Bölüm 9: MİKRODENETLEYİCİDE C PROGRAMLAMA</b>	<b>213</b>
9.1. Giriş	214
9.2. Geliştirme Araçları	215
9.3. 8051'in C Diliyle Programlanması	225
9.4. Özet	250
9.5. Sorular	251
<b>Bölüm 10: C İLE 8051 UYGULAMALARI</b>	<b>253</b>
10.1. Giriş	254
10.2. Basit Giriş/Çıkış İşlemleri	254
10.3. 7-Parçalı Gösterge (Display) Uygulamaları	267
10.4. Zamanlayıcı/Sayıcı Uygulamaları	278
10.5. Seri Haberleşme Uygulamaları	286
10.6. Kesme Uygulamaları	290
10.7. LCD Uygulamaları	297
10.8. Bellek Uygulamaları	323
10.9. Tuş takımı (Keypad) Kullanımı	328
10.10. ADC/DAC Kullanımı	334
10.11. Motor Uygulamaları	346
10.12. Özet	361
10.13. Sorular	361
<b>Bölüm 11: 8051 PROJE UYGULAMALARI</b>	<b>363</b>
11.1. Giriş	364
11.2. LCD Kullanarak Tarih-Saat Uygulaması	364
11.3. I <sup>2</sup> C Protokolünü Kullanarak Termometre Uygulaması	374
11.4. 5x7 Matris LED Kullanarak Kayan Yazı Uygulaması	382
11.5. PS/2 Klavyesinden Metinsel Ekran LCD'ye Yazı Uygulaması	392
11.6. Programlanabilir Güç Kaynağı (0-5 V) Uygulaması	399
11.7. Özet	407
11.8. Sorular	408
<b>KAYNAKLAR</b>	<b>409</b>
<b>DİZİN</b>	<b>421</b>

## ÖNSÖZ

Özellikle günümüzde, elektrik-elektronik mühendisliği alanında oldukça yaygın olarak kullanılan tasarım araçlarından birisi de mikrodenetleyicilerdir. Geçmişe nazaran oldukça fazla sayıda olan üretici firma ve bunların sonucu ortaya çıkan onlarca geliştirme sistem yazılımları bu alanda çalışan tasarımcı sayısını hızla artırmıştır. Daha da önemlisi, yeni teknolojilerle zenginleşen ve maliyetleri son on yıl içinde hızla düşen mikrodenetleyiciler, tasarımcılar için büyük bir cazibe odağı haline gelmiştir.

Sürekli gelişen ve kendini yenilemek zorunda olan elektrik-elektronik alanlarındaki konularda kitap yazmak, yazarları kaygılandıran önemli bir etkidir. Buna rağmen, böyle kapsamlı ve yararlı bir kaynağın kendi dilimizde ifade edilmesi, öğrenci ve tasarımcıların hizmetine sunulması takdir edilecek bir çabadır.

Kitabı oluşturan bölümler ve içerdiği konular son derece güncel olup kitabın günümüzde bu alanda çalışacak tasarımcıların en önemli başvuru kaynaklarından biri olacağına inanıyorum. Özellikle içerdiği kaynak CD, video dersler ve kitabın resmi İnternet sitesi eğitim sektöründe çalışan tüm öğretmen ve öğretim elemanları için yeni bir çığır açacaktır. Böylelikle kitap, durağan bir kaynaktan ziyade zenginleşen, gençleşen, güncellenen ve yaşayan bir başvuru kaynağına dönüşecektir.

Tümü geçmişte öğrencim olan yazar kadrosunu böyle bir ürünü ortaya çıkarttıkları için tebrik eder, akademik hayatlarında kalite odaklı tavırlarını devam ettirmelerini ve kitabın tüm eğitimcilere/tasarımcılara yararlı olmasını dilerim.

Prof. Dr. Hüseyin EKİZ



## Kitap Hakkında

Elektrik-Elektronik alanındaki endüstriyel uygulamalarda vazgeçilemeyen iki temel araç olan 8051 mikrodenetleyicisi ve C programlama dili, bu kitap ile bir araya getirilmiştir. Endüstriyel uygulamalar geliştiren hemen hemen tüm tasarımcılar ve elektronik alanında eğitici olarak görev yapanlar, kitap içerisindeki kaynakları doğrudan uygulama imkanına sahip olacaklardır.

Kaynak CD’de sunulan projelere ait kaynak dosyaları ve uygulamaları geliştirmek üzere deneme sürümleri verilen yazılım araçları, hem endüstriye tasarım üretenleri hem de tasarımcı yetiştiren eğitimcilere büyük kolaylıklar sağlayacaktır. Ayrıca eğitim sunuları ve eğitici video filmleri konuların çok kısa sürede kavranmasını sağlayacaktır. [www.8051turk.com](http://www.8051turk.com) sitesi kitabın sürekli olarak güncellenmesini sağlayarak, okuyucu ve tasarımcıların yeni materyallere erişimlerine de imkan tanıyacaktır.

Kitap, 11 bölümden oluşmaktadır ve ilk yedi bölüm 8051 mikrodenetleyicisinin genel mimari ve programlama ayrıntıları hakkında bilgi verirken, kalan bölümler daha ileri düzeyde ve uygulamaya dönük bilgi talep eden okuyuculara hitap etmektedir.

Bölüm 1, 8051 mikrodenetleyicisine ait genel özelliklerden ziyade endüstride yaygın kullanılan kontrol ve denetim elemanlarının mimarisi ve özellikleri üzerine odaklanmıştır. Çeşitli mimariler ayrıntılı ve karşılaştırmalı olarak incelenmiştir. Ayrıca mikrodenetleyicinin mikroişlemci ve PLC ile yapısal farkları ortaya çıkartılmıştır.

8051’in genel olarak anlatıldığı Bölüm 2’de, mikrodenetleyicilerin tarihi gelişimi ile birlikte 8051 mikrodenetleyicisinin uç isimleri ve fonksiyonları, ayrıntılı port yapısı ve çalışması ayrıntılı olarak verilmiştir. Standart bir 8051 mikrodenetleyicisinin sahip olduğu birimler tanıtılmıştır.

Bellek organizasyonu ve bellek haritası tasarımı çoğu kaynak tarafından zayıf bırakılan bir konudur. Bölüm 3’de, görülen bu zayıf noktanın ayrıntılı ve çeşitli örneklerle verilmesi amaçlanmıştır. Ayrıca 8051’ait dahili ve harici bellek alanları ve birimlerinin ayrıntılı tanıtımı yapılmıştır.

Bölüm 4’de 8051 mikrodenetleyicisinin kullandığı adresleme modları ve *assembly* dili özellikleri anlatılmıştır. Teorik konular çeşitli örneklerle ve zengin *assembly* komutlarıyla okuyucuya sunulmuştur. Tüm komut türlerinin dökümü yapılarak, yazılacak *assembly* programlarda kullanılacak olası komutlar tanıtılmıştır.

Zamanlayıcı ve sayıcılar hem donanımsal içerikleri ile hem de yazılımsal ayarları ile Bölüm 5’de tanıtılmıştır. Konu ile ilgili tüm uygulamalar, önceki bölümde öğrenilen *assembly* komutları ile uygulanarak, komutların kullanımı pekiştirilmeye çalışılmıştır.

Bölüm 6’da mikrodenetleyicinin dış ortamla iletişimini sağlayan RS-232 seri arayüz sistemi tanıtılmıştır. Çeşitli çalışma modları ve seri haberleşmeye ait bazı teknik bilgiler okuyucuya sunulmuştur.

Bölüm 7, 8051 mikrodenetleyicisine ait donanım kaynaklarının anlatıldığı son bölümdür ve endüstriyel uygulamaların vazgeçilmez çözümü olan Kesme mekanizmalarının çalışma ilkeleri ile ilgili ayrıntıları içermektedir. Kesmenin gerekliliği ve kesme tabanlı yazılım geliştirmenin incelikleri *assembly* seviyesinde verilmektedir.

Yeni nesil 8051 mikrodenetleyicisine ait fonksiyon ve birimlerin tanıtıldığı bu bölüm, uzman tasarımcılara ileri seviyede sistem tasarımında gerekli olacak birimleri tanıtmaktadır. Bölüm 8’de verilen ayrıntılar, önceki konularda olduğu gibi ayrıntılı programlama örnekleri ile verilmemiştir. Program örneklerinin verilmemesinde kullanılan benzetim araçlarının ilgili birimleri desteklememesi büyük bir etkidir. Ancak kitabın daha sonraki basımlarında bu bölümdeki konuların uygulamalı kodlar ile anlatımı düşünülmektedir.

Bölüm 9’da 8051-C dilinin, standart ANSI-C dilinden farkları örnek C fonksiyonları kullanılarak anlatılmıştır. Projelerin geliştirildiği *Keil  $\mu$ Vision®* programı ve bu programa ait ayrıntılar görsel materyaller kullanılarak tanıtılmıştır.

Bölüm 10’da mikrodenetleyici ile uygulama geliştirmek isteyenlerin ihtiyaç duyacağı çevre birimlerin kullanımını pekiştiren uygulamalar, C dilinde yazılmış kaynak kodları ve devre şemaları ile birlikte verilmiştir. Basit giriş/çıkış işlemlerinden tuş takımı uygulamalarına, 7-parçalı göstergelerden metin ve grafik ekranlı LCD’lerin kullanımına, ADC/DAC’lerden motor uygulamalarına kadar çok çeşitli çevre birimlerinin kullanımı örnekler üzerinde ayrıntılı olarak anlatılmıştır.

Son bölümde, gerçek yaşamda karşımıza çıkan mikrodenetleyici ile yapılabilen endüstriyel uygulamalardan örnekler verilmiştir. Uygulamalara ait devre şemaları ve C konularının iyice pekiştirilmesinin amaçlanmasından dolayı kodları herhangi bir kısıtlamaya gidilmeden olduğu gibi verilmiştir. Bölüm 10 ve 11’de verilen tüm uygulamaların ve kodların kaynaklarına verilen CD’den ulaşılabilir.

**Not:** Kaynak CD’den gerekli olan kodları Türkçe karakter içeren ve isimleri çok uzun klasörlere kopyalayamayız. Klasör isimleri Türkçe karakterlerden arındırılmış ve kısa olmalıdır. Aksi halde ISIS programı beklenen benzetim işlemlerini yapamayabilir.

# MİKRODENETLEYİCİLERE

## GENEL BAKIŞ

### 1. BÖLÜM

#### Amaçlar

- Mikrodenetleyici terimini kavramak
- Mikrodenetleyicilerin kullanım alanları hakkında bilgi sahibi olmak
- Bir mikrodenetleyiciyi oluşturan birimleri tanımak
- Modern mikrodenetleyicilerin mimari yapıları hakkında bilgi sahibi olmak
- Mikrodenetleyici seçiminde dikkat edilmesi gereken ölçütleri öğrenmek

#### Başlıklar

- 1.1. Giriş
- 1.2. Mikrodenetleyici Mimari Özellikleri
- 1.3. Mikrodenetleyici Seçiminde Dikkat Edilecek Ölçütler
- 1.4. Mikrodenetleyicinin Rakipleri
- 1.5. Özet
- 1.6. Sorular

#### 1.5. Özet

Mikrodenetleyiciler, endüstriyel denetim sistemlerinin tasarımında kullanılmakla birlikte taşınabilir elektronik cihazlar ve ev elektroniği alanlarında daha çok kullanılmaktadır. Özellikle düşük güç tüketimli ve ekonomik cihazların üretimi, mikrodenetleyiciler ile mümkün olmuştur. Mikrodenetleyicilerde, olabildiğince fazla sayıda tasarım ihtiyacını karşılamak üzere çeşitli işleve sahip dahili birimlere yer verilmiştir. Tasarımcı, kendi projesine en uygun olan kullanıma hazır mikrodenetleyiciyi seçerek sistemin gerçekleştirilmesini sağlar.

Mikroişlemci ve mikrodenetleyiciler bellek organizasyonu açısından iki temel mimariden birine sahiptir. Donanım yapılanmasını etkileyen bu mimariler *Harvard* ve *Von Neuman* olarak bilinirler. *Harvard* mimaride kod ve veri ayrı fiziksel alanlarda saklanırken, *Von Neuman* mimaride hem kod hem de veriler tek bir bellek biriminde saklanır.

8051 mikrodenetleyicileri her iki mimaride de çalıştırılabilme esnekliğine sahiptir. Ancak *Harvard* mimari, 8051 mikrodenetleyicilere ait özgün bir mimaridir ve bu yüzden tercih edilmelidir.

Komut işleme yöntemine göre *RISC* ve *CISC* olarak ikiye ayrılan mikroişlemci/mikrodenetleyici mimarileri, işlemcilerin mikrokod seviyesindeki çalışma tekniklerini belirler. *RISC* çekirdekler daha az donanım kaynağı kullanarak daha az sayıda komutun çalışmasına izin verirken, *CISC* mimariler karmaşık donanım yapısı ile çok sayıda kod formatına destek vermektedir.

Günümüzde çok farklı mimariler ve karma mimari çeşitleri ile üretilen 8051 veya diğer mikrodenetleyici aileleri mevcuttur. Tasarımcının aklına gelen ilk soru, amacına hizmet edecek mikrodenetleyicinin piyasada satılan ürünlerden hangisi olduğudur. Aslında bu sorunun cevabı çok da basit değildir. Her hangi bir mikrodenetleyici ailesine bağlı kalmak gelişmelere gözünüzü kapatmak ile eşdeğerdir. Cevabı oluşturmanıza yardım edecek birçok ölçüt sayılabilir: maliyet, işlevsellik, ürün desteği, geliştirme yazılımları, bilgi birikimi ve deneyim, esneklik gibi daha birçok kriter göz önünde bulundurulmalıdır. Bu arada mikrodenetleyicilerin endüstriyel uygulamaları gerçekleştirmek için gerekli olan tek çözüm olmadığı da akıldan çıkarılmamalıdır.

## 1.6. Sorular

- 1.6.1) Mikrodenetleyici ve mikroişlemci arasındaki fark nelerdir?
- 1.6.2) Mikrodenetleyici ve PLC arasındaki fark nelerdir?
- 1.6.3) Bir mikrodenetleyicinin temel birimleri nelerdir?
- 1.6.4) Mikrodenetleyicilerin popüler olma sebepleri nelerdir?
- 1.6.5) *Harvard* mimarisi ile *Von Neuman* mimarisi arasındaki yapısal fark nelerdir?
- 1.6.6) *RISC* ve *CISC* mimarisi arasındaki farklar nelerdir?
- 1.6.7) *RISC* ve *CISC* mimarilerden farklı olarak hangi tür mimariler vardır? Açıklayınız?
- 1.6.8) Mikrodenetleyici seçiminde dikkat edilecek kriterleri listeleyiniz?
- 1.6.9) 8051 mikrodenetleyicisi neden çok sayıda firma tarafından üretilmektedir?
- 1.6.10) 8051 mikrodenetleyicisi ve diğer mikrodenetleyiciler arasındaki temel fark nelerdir?
- 1.6.11) Mikrodenetleyicili sistemlerin endüstriyel alanlarda kullanılırken karşılaşılabilecek zorlukları ve uygulama sınırlandırmalarını maddeler halinde yorumlayınız?

**8051**

**MİKRODENETLEYİCİSİNE**

**GİRİŞ**

**2.  
BÖLÜM**

**Amaçlar**

- 8051 mikrodnetleyicisinin tarihi gelişimini açıklamak
- 8051 mikrodnetleyicisinin mimari yapısını kavramak
- 8051 mikrodnetleyicisinin fiziksel özelliklerini tanımlamak
- 8051 mikrodnetleyicisinin iç yapısını kavramak

**Başlıklar**

- 1.7. MCS-51 Ailesi
- 1.8. 8051 Mikrodnetleyicisinin Genel Yapısı
- 1.9. 8051 Mikrodnetleyicisinin Uç Fonksiyonları
- 1.10. Ayrıntılı Port Yapısı
- 1.11. 8051 Mikrodnetleyicisinin Ayrıntılı İç Mimarisi
- 1.12. Özet
- 1.13. Sorular

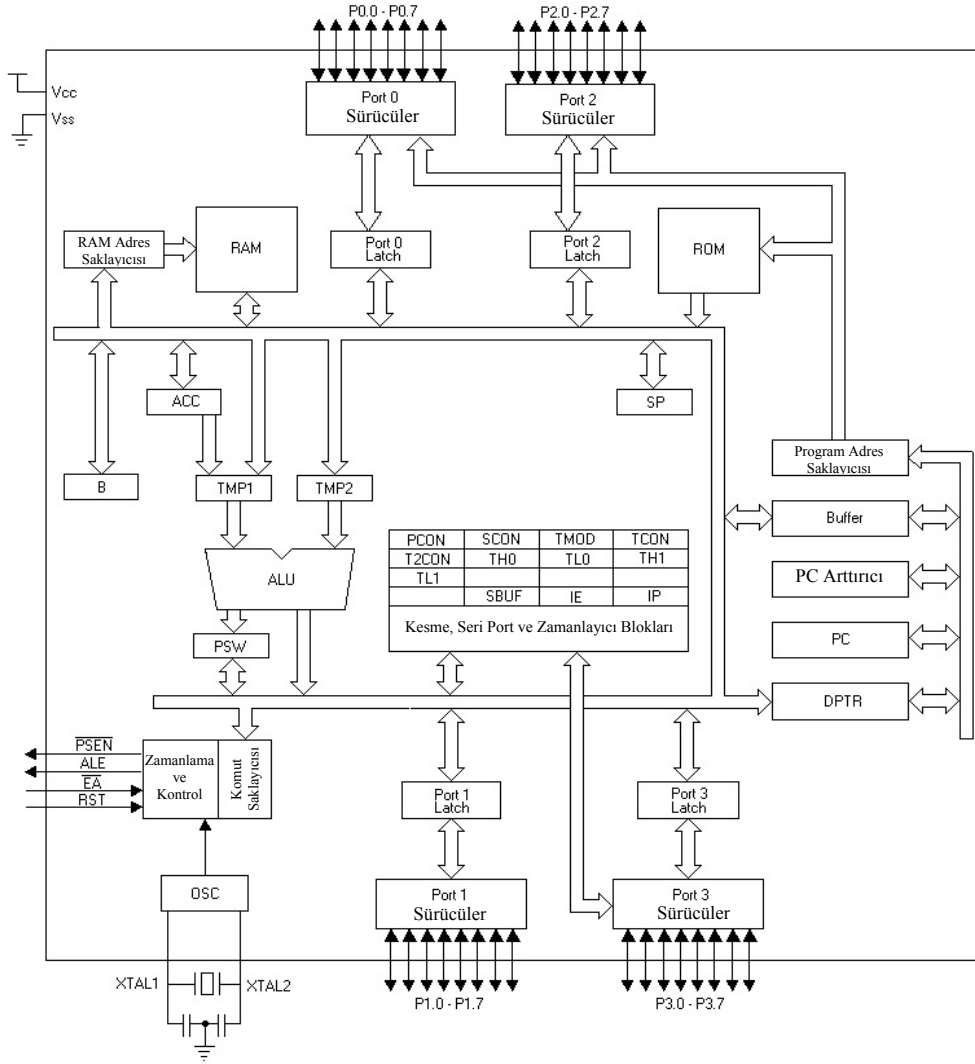
**2.6. Özet**

8051 mikrodnetleyicisi tüm dünya tarafından kabul görmüş ve bir endüstri standardı haline gelmiştir. Elektronik ve yarı-iletken sanayinin devleri olan *Intel*, *Philips*, *Atmel* ve *Dallas* gibi üreticiler 8051 mikrodnetleyicilerini üreten yirmiden fazla firmadan sadece bir kaçıdır.

8051, standart olarak en az 128 Byte'ı genel amaçlı ve 128 Byte'ı SFR olmak üzere 256 Byte'lık bir dahili RAM bellek birimine sahip olmalıdır. Program belleği için böyle bir varsayım doğru değildir. Endüstriyel kontrol düzeneklerinde sıkça ihtiyaç duyulan zamanlama ve sayma işlemleri için de 8051 ailesi en az iki adet dahili zamanlayıcı/sayıcı birimi bulundurur.

## C ile 8051 Mikrodenetleyici Uygulamaları

8051 mikrodenetleyicilerin giriş/çıkış portları, endüstriden gelen taleplerin çoğuna cevap verebilmek için çok işlevli olarak tasarlanmıştır. Böylece, hem yerden tasarruf edilebilir hem de ekonomik bir çözüm üretme fırsatı yakalanabilir. Reset işlemi ve saat darbesinin üretimi yine dahili birimler ile gerçekleştirildiğinden, fazladan harici elektronik elemanların bağlanması önüne geçilmiş olur. 8051, harici bellek birimleri ile kullanılırken gerekli olacak arayüz uçlarına da sahiptir ve bu yönüyle, profesyonel tasarımların gerçekleştirilmesinde de kullanılabilir.



**Şekil-2.7. 8051 mikrodenetleyicisinin ayrıntılı blok diyagramı**

[Alıntı: IC20: 8051 Data Book, Philips,1994]

## 2.7. Sorular

- 2.7.1) İlk mikrodenetleyici üretimi hangi firma tarafından ne amaçla yapılmıştır?
- 2.7.2) Belli başlı 8051 üreticilerini listeleyiniz?
- 2.7.3) Standart 8051 mikrodenetleyicisinde dahili bellek türleri ve büyüklüklerini açıklayınız?
- 2.7.4) Standart 8051 mikrodenetleyicisinin temel özellikleri nelerdir?
- 2.7.5) 8051 MİB'nin temel işlevlerini sıralayınız?
- 2.7.6) Üreticiler arasındaki çekirdek farklılıkları nelerdir?
- 2.7.7) Program ve veri belleği ne demektir?
- 2.7.8) Program bellek türlerini sayınız?
- 2.7.9) Program bellek türlerini üstünlüklerine göre kıyaslayınız?
- 2.7.10) Program bellek türünü seçerken nelere dikkat etmeliyiz?
- 2.7.11) Veri bellek türlerini sayınız?
- 2.7.12) Veri bellek türünü seçerken dikkat edilecek kriterleri belirtiniz?
- 2.7.13) 8051 mikrodenetleyicisinin sahip olduğu çevresel birimlerin amacı nedir?
- 2.7.14) 8051 mikrodenetleyicisi ile birlikte kullanılan seri haberleşme protokollerini listeleyiniz?
- 2.7.15) Dahili saat devresinin bileşenlerini çiziniz?
- 2.7.16) Standart 8051 mikrodenetleyicisinin devre simgesini çiziniz?
- 2.7.17) Standart 8051 mikrodenetleyicisinin çalışması için hangi uçlara kaç Volt uygulamak gerekir?
- 2.7.18) Standart 8051 mikrodenetleyicisinin kontrol uçlarını listeleyiniz?
- 2.7.19) Aşağıdaki kontrol uçlarının görevlerini kısaca açıklayınız?  
a) ALE b)  $\overline{\text{PSEN}}$  c)  $\overline{\text{EA}}$  d) Reset
- 2.7.20) Harici bellekten kod okumak gerektiğinde hangi uç nasıl kullanılmalıdır?
- 2.7.21) Kaç tür sıfırlama (reset) işlemi vardır? Devrelerini çiziniz?
- 2.7.22) Standart 8051 mikrodenetleyicisinde osilatör frekansı kaç Hz'dir?
- 2.7.23) 12 MHz'lik bir kristal frekansında kullanılan kondansatörler kaç Farad'dır?
- 2.7.24) Standart 8051 mikrodenetleyicisinde giriş/çıkış uç sayısı kaç adettir?
- 2.7.25) Port 0'ın diğer portlardan farkı nedir?
- 2.7.26) P0 portunda neden dahili olarak çekme dirençleri kullanılmaz?
- 2.7.27) Port 3'ün devre şemasını çiziniz?
- 2.7.28) Port 3'ün alternatif özelliklerini sayınız?
- 2.7.29) Harici bellek kullanımında ihtiyaç duyulan portları yazınız?

# 8051'İN BELLEK ORGANİZASYONU

## 3. BÖLÜM

### Amaçlar

- 8051 mikrodenetleyicisinin bellek türlerini öğrenmek
- Dahili veri belleği (Internal RAM) hakkında bilgi sahibi olmak
- Özel işlev saklayıcılarını tanımak
- Harici program bellek bağlantılarını kavramak
- Harici veri bellek bağlantılarını kavramak
- Bellek haritalama ve organizasyonu tasarımını kavramak

### Başlıklar

- 3.1. Giriş
- 3.2. Program (Kod) Belleği
- 3.3. Veri Belleği
- 3.4. Bellek Organizasyonu Tasarımı ve Adres Çözümleme
- 3.5. Özet
- 3.6. Sorular

### 3.5. Özet

8051 mikrodenetleyicisi çeşitli amaçlar için birçok bellek birimi içermesinin yanında aynı birimde farklı bellek alanları da içerir. Sistemin bu tarzda şekillenmesi, hem özgün mimariye sahip olunması hem de farklı amaçlara hizmet edilebilmesi büyük rol oynamıştır.

Her bellek bölgesine erişim, desteklenen çeşitli adresleme yöntemleri ile gerçekleştirilmektedir. Harici bellek birimlerine erişim de üretilen özel komutlar vasıtasıyla yapılmaktadır. Günümüzde üretilen bazı 8051 aileleri XRAM olarak adlandırılan bellek



birimlerini de içerebilmektedir. Böylece, harici bellek ihtiyacı dahili olarak karşılanabilmektedir. Teknolojik gelişmelerin eğilimleri göz önüne alındığında, harici RAM belleklerinin tamamını 8051 tümdevresi içerisinde bulunduran ailelerin üretimi uzak değildir.

Harici cihaz/birim bağlamaya son derece uyumlu olan 8051 mikrodenetleyicilerinde, harici birimlerin yetkilendirilmesi için daha çok bellek haritalı giriş/çıkış (Memory Mapped I/O) yöntemi kullanılır. Bu teknikte en önemli üç basamak:

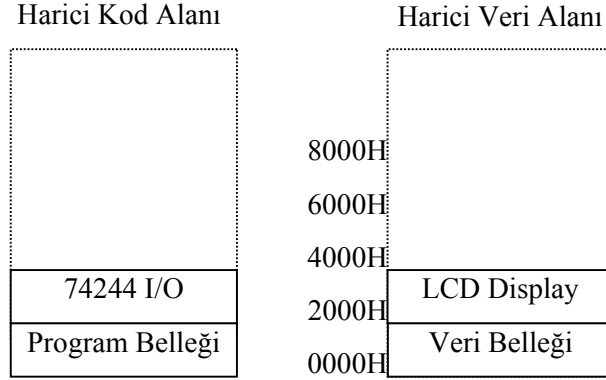
- Birimin temsil edileceği bellek türü (program, veri)
- Birimin adres sınırları
- Birimin etkinleştirileceği adresin çözümlenmesi

Sistemin bir bütünlük arz etmesi ve sorunsuz şekilde çalışması için bağlanacak birimlerin genel özelliklerinin çok iyi bilinmesi gerekir. Örneğin yazılabilme işlevine sahip bir harici birimin kod belleğine yerleştirilmesi büyük bir hatadır.

### 3.6. Sorular

- 3.6.1) Harici program belleğine ne zaman ihtiyaç duyulur?
- 3.6.2) Veri belleği kullanırken 8051'in hangi uçlarına ihtiyaç duyulur?
- 3.6.3) Harici program veya veri belleği kullanırken hangi tümdevreye ihtiyaç duyulur? Neden?
- 3.6.4) Standart 8051 mikrodenetleyicisi ile harici olarak en çok ne kadarlık bir alan adreslenebilir?
- 3.6.5) 16 KByte'lık bir harici kod bellek için kaç adet port ucuna ihtiyaç vardır?
- 3.6.6) Standart 8051 mikrodenetleyicisine ait dahili RAM kaç Byte'tır?
- 3.6.7) Standart 8051 mikrodenetleyicisinin dahili RAM bölgelerini adres sınırlarını belirterek çiziniz?
- 3.6.8) Alt RAM bölgesinin ayrıntılarını adres sınırları vererek gösteriniz?
- 3.6.9) 8051'de neden birden fazla saklayıcı deposu kullanılmaktadır?
- 3.6.10) Bit adreslenebilir alanın ayrıntılarını adres sınırları vererek gösteriniz?
- 3.6.11) Standart 8051 mikrodenetleyicisinde neden çok çeşitli dahili veri bellek türleri kullanılmaktadır?
- 3.6.12) Standart 8051'de toplam kaç adet bit değişkeni tanımlanabilir?
- 3.6.13) Alt ve üst RAM'e erişim yöntemlerini *assembly* dilinde gösteriniz?
- 3.6.14) SFR belleği nedir, ne amaçla kullanılmaktadır?
- 3.6.15) Bit adreslenebilir SFR saklayıcıları hangi adreslerde yer alırlar?
- 3.6.16) Maskeleyme nedir, ne amaçla kullanılır? Örneklerle açıklayınız.
- 3.6.17) Akümülatör hangi SFR adresinde yer alır?
- 3.6.18) PSW içerisindeki bitlerin işlevlerini açıklayınız?
- 3.6.19) Çarpma ve bölme komutlarında hangi saklayıcılar kullanılır?
- 3.6.20) Depo (Bank) değiştirmenin amacı nedir?
- 3.6.21) DPTR saklayıcısı neden 16 bit uzunluğundadır?

- 3.6.22) Yığın işaretçisi (SP) ne işe yarar, ne tür bir bellektir?  
3.6.23) Mikrodenetleyicinin düşük güç tüketim moduna getirilmesi için hangi saklayıcı ve hangi bit kullanılır?  
3.6.24) PD ve IDL modları arasındaki farklar nelerdir?  
3.6.25) Harici veri belleği kullanabilmek için hangi düzenlemeleri yapmak gerekir, blok şemasını çiziniz?  
3.6.26) Bellek haritasına neden ihtiyaç duyulur?  
3.6.27) Çevresel birimlere erişim işlevi hangi yöntemlerle yapılabilir?  
3.6.28) 3 adet 8 KB'lık veri belleğini 8051 mikrodenetleyicisine bağlayınız?  
3.6.29) Aşağıda bellek haritası verilen 8051 sistemini 74138 kod çözücü ile adres, veri ve kontrol yollarını düzenleyerek devre şeması şeklinde çiziniz?



# ADRESLEME MODLARI VE KOMUT KÜMESİ

## 4. BÖLÜM

### Amaçlar

- *Assembly* dilinin genel özelliklerini tanımak
- 8051 mikrodenetleyicisinin adresleme yöntemlerini kavramak
- 8051 komut kümesinin kullanımı hakkında bilgi sahibi olmak
- *Assembly* dilinde örnek uygulama yazabilmek

### Başlıklar

- 1.14.Giriş
- 1.15.*Assembly* Dili ve Özellikleri
- 1.16.Adresleme Yöntemleri
- 1.17.8051 Komut Türleri
- 1.18.Özet
- 1.19.Sorular

### 4.5. Özet

*Assembly* dili ile program geliştirme PC'ler için büyük oranda geride kaldıysa da, git-tikçe azalan oranda da olsa yerleşik (embedded) sistem tasarımlarında hala kullanılmaktadır. *Assembly* dilinin, yüksek düzeyli dillere oranla daha küçük boyutlu ve hızlı çalışan kod ürettiği bir gerçektir. Ancak profesyonel ve büyük ölçekli tasarımlarda zorunlu olmadıkça *assembly* dili tercih edilmemektedir. Hem dokümantasyon açısından hem de programcı açısından *assembly* dili büyük zorluklar meydana getirmektedir. *Assembly* dili makine dilinin (ikili kod) bir üst seviyesi olarak bilinir ve tüm komutların bire bir opkod karşılıkları vardır. Yüksek düzeyli dillerde ise derleyiciler aynı komutu, yerine göre farklı bir opkod dizisine dönüştürebilir.

Her mikroişlemci veya mikrodenetleyici adresleme yöntemlerine sahiptir. Temel olarak komutların daha kısa sürede işlenmesini hedefleyen adresleme yöntemleri çeşitlendikçe tümdevre üzerindeki donanım daha karmaşık hale gelmektedir. 8051 genel olarak: saklayıcı, doğrudan, dolaylı, ivedi, bağıl, mutlak, uzun ve sıralı adresleme yöntemlerine sahiptir.

*Assembly* komutları belli başlı birkaç başlık altında toplanabilir:

- Aritmetik komutlar
- Mantıksal komutlar
- Veri transfer komutları
- Bit işlem komutları
- Program dallanma komutları

Her komut grubunda çeşitli adresleme yöntemleri kullanılmaktadır. Bit işlem komutları mikrodenetleyicileri, mikroişlemcilerden ayıran komut grubudur.

#### 4.6. Sorular

- 4.6.1) Opkod nedir? Açıklayınız?
- 4.6.2) *Assembly* dilinin üstünlükleri nelerdir?
- 4.6.3) *Assembly* dili formatını bir komut üzerinde açıklayınız?
- 4.6.4) *Assembly* dilinde kullanılan sayı formatlarını birer örnekle gösteriniz?
- 4.6.5) ORG yönergesi ne amaçla kullanılır?
- 4.6.6) END yönergesi ne amaçla kullanılır?
- 4.6.7) DB yönergesi ne amaçla kullanılır?
- 4.6.8) EQU yönergesi ne amaçla kullanılır?
- 4.6.9) # ve \$ simgelerinin *assembly* dilindeki anlamlarını belirtiniz?
- 4.6.10) 8051 mikrodenetleyicisinde kullanılan adresleme yöntemlerini sayınız?
- 4.6.11) 8051'de neden çok sayıda adresleme yöntemi kullanılmaktadır?
- 4.6.12) Saklayıcı adresleme yönteminde hangi saklayıcılar kullanılabilir?
- 4.6.13) Saklayıcı adresleme yönteminde kullanılan komutlar kaç Byte ve kaç makine çevrimidir?
- 4.6.14) Doğrudan adreslemenin bellek sınırlarını belirtiniz?
- 4.6.15) Doğrudan adresleme yöntemi hangi bellek bölgelerine erişim için kullanılır?
- 4.6.16) Doğrudan adresleme yönteminde kullanılan komutlar kaç Byte ve kaç makine çevrimidir?
- 4.6.17) Dolaylı adresleme yöntemi ne amaçla kullanılır?
- 4.6.18) Dolaylı adresleme yönteminde kullanılan saklayıcılar nelerdir, bir örnekle açıklayınız?
- 4.6.19) Bir saklayıcıya doğrudan veri yükleme hangi adresleme yönteminde yapılır, bir örnekle açıklayınız?

4.6.20)

ADRES	VERİ
00h	80
01h	02
02h	XX
03h	XX
04h	80
05h	FA
06h	XX
07h	XX
08h	XX
09h	XX
0Ah	XX
0Bh	XX
0Ch	XX
0Dh	80
0Eh	F5
0Fh	

Yandaki ROM (program) belleğinde (80h) *SJMP* anlamına gelmektedir. Bir sonraki veriler de bağlı adresleri vermektedir. XX fark etmez anlamındadır. Buna göre 8051, RESET ile çalıştırılırsa program dallanmaları nasıl olur. Yandaki bellek haritasında oklar kullanarak gösteriniz. Her oku çalışma sırasına göre numaralandırınız?

4.6.21) Mutlak adresleme yöntemini kullanım amacını belirterek açıklayınız?

4.6.22) Uzun adresleme yönteminde kullanılan komutlar nelerdir?

4.6.23) Sıralı adreslemenin amacını belirtiniz?

4.6.24) 8051 mikrodenetleyicisindeki kaç farklı komut kümesi bulunmaktadır?

4.6.25) Aşağıdaki program ne gibi bir fonksiyon icra eder. Program BEKLE döngüsüne girdikten kaç komut sonra *SJMP BASLA* satırına gelir?

```

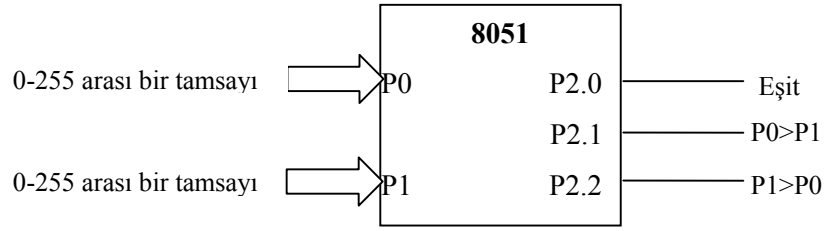
BASLA:  MOV     R0, #10h
        MOV     R1, #10h
BEKLE:  DJNZ   R0, BEKLE
        DJNZ   R1, BEKLE
        SJMP   BASLA

```

4.6.26) 8051'in Port-1 uçlarını  $F = (A+B.D) \cdot C$  fonksiyonunu gerçekleştirecek şekilde programlayınız?

C ile 8051 Mikrodenetleyici Uygulamaları

- 4.6.27) 8051'in Port-1 uçlarını kullanarak iki girişli bir XOR kapısı gerçekleştirmek için gerekli bağlantı ve programı gösteriniz. P1 uçları XOR kapısının girişleri ve çıkışı olacaktır?
- 4.6.28) 8051'in dahili belleğinde 60h'dan 90h'a kadar olan alan 00h bilgisi ile doldurulacaktır. Gerekli programı yazınız?



- 4.6.29) Yukarıdaki sistemde görüldüğü gibi P0 ve P1 portlarına 8-bitlik bir tamsayı bilgisi gelmektedir.  
Her iki sayı eşit ise P2.0=1 (HIGH), diğer çıkışlar '0' (LOW)  
P0'daki sayı P1'den büyükse, P2.1=1 diğerleri '0'  
P1'deki sayı P0'dan büyükse P2.2=1 diğerleri '0' olmaktadır. Bu koşulları yerine getiren *assembly* programını tasarlayınız?
- 4.6.30) 8051'in dahili belleğinde tamsayılardan (8-bit) oluşan bir dizi olduğunu varsayalım. Dizinin uzunluğu R5 saklayıcısında, dizinin başlangıç adresi de R1 saklayıcısında bulunuyor ise bu dizideki elemanların toplamını akümülatörde biriktiren alt programı yazınız ?

# ZAMANLAYICILAR / SAYICILAR

## 5. BÖLÜM

### Amaçlar

- Zamanlayıcı/Sayıcı tanımını kavramak
- 8051’de Zamanlayıcı/Sayıcı saklayıcılarını öğrenmek
- Zamanlayıcı/Sayıcı modlarının kullanımları hakkında bilgi sahibi olmak
- Uygulamalarda Zamanlayıcı/Sayıcı birimini kullanabilmek

### Başlıklar

- 1.20.Giriş
- 1.21.Zamanlayıcı/Sayıcı Saklayıcıları
- 1.22.Zamanlayıcı/Sayıcı Çalışma Modları
- 1.23.Zaman ya da Durum Sayma Seçimi
- 1.24.Zamanlayıcıların Kullanımı ve Örnekler
- 1.25.Zamanlayıcı/Sayıcı 2
- 1.26.Özet
- 1.27.Sorular

### 5.7. Özet

Özellikle endüstriyel uygulamalarda karşımıza çıkmakta olan periyodik görevler veya sayma işlevleri, mikrodenetleyicilerin dahili zamanlama ve sayma birimine ihtiyaç duymalarının en büyük nedenlerindedir. Zamanlama ve sayma kavramları, saat darbesinin hangi kaynaktan (dahili, harici) geldiği ile ilgilidir.

Zamanlayıcı/Sayıcı (Z/S) birimi mikroişlemcinin yazılım ile yapabileceği sayma ve zamanlama görevlerini donanımsal olarak gerçekleştirir. Böylece mikroişlemci, tüm işlem gücünü yazılan uygulamanın çalıştırılmasına harcar. Ayrıca gerçek-zamanlı sistemlerde mikroişlemcilerin iş yükü nedeniyle oluşacak bir gecikme, sistemin çalışma-

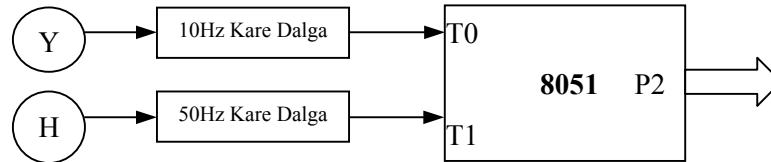
sını etkileyebilir. Bu gibi durumlarda en iyi çözüm yazılım yerine donanım kaynaklarının kullanılmasıdır.

8051 mikrodenetleyicisinde standart olarak T0 ve T1 olarak adlandırılan iki adet Z/S birimi vardır. 8051 ailesinin bir türevidir olan 8052 ailesinde, standart Z/S birimlerine ek olarak T2 Z/S birimi de bulunmaktadır. Sayma ve zamanlama yoğunluklu bir tasarımda çok sayıda Z/S birimlerine ihtiyaç duyulur.

Z/S birimleri, sahip oldukları saklayıcılar ile istenilen bir moda çalıştırılabilecek şekilde tasarlanmıştır. Böylece, Z/S birimlerinin çok yönlü ve farklı uygulamalara cevap verilebilecek şekilde ayarlanması mümkün olabilmektedir.

## 5.8. Sorular

- 5.8.1) Durum ve zaman sayma kavramlarını açıklayınız?
- 5.8.2) Harici ve dahili sayma işlemlerine neden gerek duyulmaktadır? Açıklayınız.
- 5.8.3) Harici sayma uçları hangi porttadır?
- 5.8.4) Zamanlayıcı saat frekansı nereden elde edilir ve nasıl hesaplanır?
- 5.8.5) Zamanlayıcı/sayıcı birimine ait olan SFR saklayıcılarını listeleyiniz?
- 5.8.6) TMOD saklayıcısı bitlerinin işlevlerini açıklayınız?
- 5.8.7) Zamanlayıcı/Sayıcı birimlerinde neden farklı modlara ihtiyaç duyulur?
- 5.8.8) TCON saklayıcısı bitlerinin işlevlerini açıklayınız?
- 5.8.9) Zamanlayıcı/sayıcının iç yapısını çiziniz?
- 5.8.10) Taşma olayı nedir? Oluşması için hangi koşulların oluşması gerekir?
- 5.8.11) Mod 0'ın çalışma yöntemini açıklayınız?
- 5.8.12) Mod 1'in çalışma yöntemini açıklayınız?
- 5.8.13) Mod 2'nin çalışma yöntemini açıklayınız?
- 5.8.14) Mod 3'ün çalışma yöntemini açıklayınız?
- 5.8.15) Mod 0 kullanarak zamanlayıcının her 2000 adımda taşması için sayma saklayıcılarına (TH0, TL0) kaç değerini yüklemek gerekir?
- 5.8.16) Port 1'in 0 nolu ucundan 500 Hz'lik bir kare dalga işareti elde etmek için gerekli olan assembly programı yazınız?
- 5.8.17) T0'in mod 1'de sayıcı olarak kullanılmasını şematik olarak gösteriniz?
- 5.8.18) Bir lastik üretim tesisinde, üretim bandından dakikada geçen lastik miktarı ölçülmesi istenmektedir. Bir algılayıcı her lastik sonrası bir kare darbeyi üretmektedir. Tasarlayacağınız sistem üretilen bu darbeyi alarak ve bir dakikada geçen lastik miktarını Port 1'in çıkışına verecektir. Port 1'deki sayısal değer ikili olacağına göre istenenleri karşılayacak *assembly* programını yazınız?





- 5.8.19) Yukarıdaki sistemde görüldüğü gibi T0 ve T1'e iki 10 Hz'lik ve 50 Hz'lik kare dalga üretici bağlıdır. Y ve H butonlarına basıldığı sürece üreteçler kare dalga üretmektedirler. Y yavaş sayma, H butonu hızlı saydırma için kullanılacaktır. Butona basılı kaldığı sürece sayma işlemi P2 de ikili olarak sürecektir; basılmadığı durumda ise sayma duracaktır. Sistem için gerekli *assembly* komutlarını yazınız?  
Not: Her ikisine beraber basıldığında P2 sıfırlanacaktır.
- 5.8.20) 8051'deki Port 1'i kullanarak 100 Hz. hızında 8 bitlik yürüyen ışık (LED'lerle) devresinin programını yazınız?

# SERİ PORT İŞLEMLERİ

## 6. BÖLÜM

### Amaçlar

- Seri haberleşmenin önemini kavramak
- 8051 seri port kontrol saklayıcılarını öğrenmek
- Seri port çalışma modları hakkında bilgi sahibi olmak
- Farklı hızlarda seri haberleşmeyi gerçekleştirebilmek
- Seri port kullanarak karakter gönderme ve almayı öğrenmek

### Başlıklar

- 1.28.Giriş
- 1.29.Seri Port Kontrol Saklayıcısı (SCON)
- 1.30.Seri Port Çalışma Modları
- 1.31.Seri Port Saklayıcılarının Kurulumu ve Seri Port Saklayıcılarına Erişim
- 1.32.Çok İşlemcili Haberleşme
- 1.33.Seri Port Baud Hızlarının Ayarlanması
- 1.34.Seri Port Uygulama Örnekleri
- 1.35.Özet
- 1.36.Sorular

### 6.8. Özet

Mikrodenetleyicilerin en önemli ilkelerinden birisi, az uçla çok işleve sahip olmasıdır. Herhangi bir şekilde harici bir sistemle haberleşmesi durumunda sahip olduğu uçların paralel haberleşme amacıyla kullanılmayacağı (belli koşullarda mümkün olabilir) açıktır. Bu durumda seri haberleşme yöntemleri mikrodenetleyiciler için vazgeçilmez bir seçenektir.

Seri port, bilgilerin seri olarak iletildiği bir giriş/çıkış birimidir. Standart 8051 mikrodenetleyicisinde en az bir adet seri port birimi bulunur ve Port-3'te bulunan TxD ve RxD uçları yardımıyla haberleşme çift yönlü olarak gerçekleştirilir. Seri porta ait bir zamanlama işareti bulunmadığından veri iletişimi eşzamansızdır (Mod 0 hariç) ve bu yüzden, senkronizasyon işlemi verinin kendisi ile sağlanmaya çalışılır.

Bir mikrodenetleyici için seri port, ilk zamanlarda sadece farklı mikrodenetleyiciler veya bir bilgisayar ile haberleşmek için kullanılıyordu. Teknolojinin gelişmesine paralel olarak seri port daha işlevsel olmaya başladı. Flash belleklerin yaygınlaşması ve ISP (In-System Programming) teknolojisi sayesinde, mikrodenetleyiciler oldukça pahalı olan paralel port programlayıcılardan da kurtulmuş oldular. Ancak yine de bazı özel durumlar için paralel programlayıcı cihaz veya kartlara ihtiyaç duyulmaktadır.

Günümüzde üretilen yeni nesil 8051 mikrodenetleyicileri endüstride oldukça yaygın olarak kullanılan *CAN*, *I<sup>2</sup>C* ve *SPI* protokollerinin yanında, PC endüstrisinin vazgeçilmez seri iletişim arayüzü olan *USB* protokolünü de içermeye başlamıştır.

## 6.9. Sorular

- 6.9.1) Paralel ve seri haberleşme tekniklerini karşılaştırınız?
- 6.9.2) Mikrodenetleyicilerde seri haberleşme protokollerinin kullanılma nedenleri nelerdir? Açıklayınız.
- 6.9.3) Seri portun fiziksel şemasını çiziniz?
- 6.9.4) Seri porta ait SFR saklayıcıları nelerdir? İşlevlerini açıklayınız?
- 6.9.5) RS-232 protokolü kullanımında yararlanılan port uçları hangileridir?
- 6.9.6) Seri port (RS-232) çalışma modlarını kısaca açıklayınız?
- 6.9.7) Ötelemeli saklayıcı kullanılarak port uçlarının sayısı nasıl artırılır?
- 6.9.8) Seri porttan bir karakter (7 bit+eşlik) göndermek için gerekli program parçasını yazınız?
- 6.9.9) Seri porttan bir karakter (8 bit+eşlik) göndermek için gerekli program parçasını yazınız?
- 6.9.10) Seri porttan bir karakter (7 bit+eşlik) almak için gerekli program parçasını yazınız?
- 6.9.11) Seri porttan bir karakter (8 bit+eşlik) almak için gerekli program parçasını yazınız?
- 6.9.12) Çok işlemcili haberleşme (RS-232) adımlarını açıklayınız?
- 6.9.13) Baud nedir ve baud hızı nasıl ayarlanır?
- 6.9.14) SMOD bitinin işlevini açıklayınız?
- 6.9.15) 11.0592 MHz kristal frekansında 9600 baud hızı elde edebilmek için TH1 hangi değere kurulmalıdır?
- 6.9.16) 11.0592 MHz kristal frekansında TH1=F4h ise hangi baud hızı elde edilir?
- 6.9.17) 11.0592 MHz kristal frekansında 9600 baud hızında “Merhaba” sözcüğünü seri porttan gönderen *assembly* programını yazınız?
- 6.9.18) 8051'e seri arayüzden bir PC tarafından gönderilen karakterler gelmektedir. SBUF'da bulunan herbir karakter, mikrodenetleyicimiz için bir fonksiyon icrası sayılmaktadır.

‘B’ : Basla,                      ‘Y’ : Yükle,                      ‘D’ : Dur,  
‘1’ : Birinci Program            ‘2’ : İkinci program

Böyle bir çalışma için gerekli program yapısını kurunuz?

# KESMELER

## (Interrupts)

## 7. BÖLÜM

### Amaçlar

- Kesme tanımını ve önemini kavramak
- 8051 mikrodenetleyicisinin kesme yapısını öğrenmek
- 8051’de kullanılan kesme türlerini öğrenmek
- Kesmeleri kullanarak uygulama geliştirmek

### Başlıklar

- 1.37.Giriş
- 1.38.8051 Kesme Düzenlemesi
- 1.39.Kesme Vektör Adresleri
- 1.40.Kesmelerin Çalışması
- 1.41.Kesme Örnekleri
- 1.42.Özet
- 1.43.Sorular

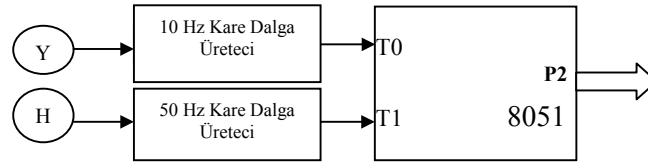
### 7.5. Özet

Mikrodenetleyicilerde, harici ve dahili gelişen olayları çözümlenin iki yolu vardır: yoklama döngüsü oluşturma veya kesme üretimi. Kesme birimleri, mikrodenetleyicilerin sürekli olarak bir veya birden fazla olay bayraklarını yoklama yöntemiyle sınavarak zaman kaybetmelerinin önüne geçerek, mikrodenetleyicinin yapması gereken diğer zaman-kritik görevlerine vakit ayırmalarını sağlar.

Standart 8051’de  $\overline{INT0}$ ,  $\overline{INT1}$ , T0, T1 ve RI-TI olarak toplam beş adet kesme kaynağı vardır. Gelen her kesme bildirim için program belleğinde dallerilecek standart bir vektör adresi tanımlanmıştır ve bu alana kesme vektör tablosu denilir. Böylece tüm 8051 türevlerinde, yazılan herhangi bir programın çalışma garantisi de sağlanmış olur.

## 7.6. Sorular

- 7.6.1) Kesme yönteminin getirdiği üstünlükler nelerdir?
- 7.6.2) Kesme önceliklerinin olması ne ile açıklayabiliriz. Yaşamdan bir örnek veriniz?
- 7.6.3) Yoklama kavramı ile kesme kavramı arasındaki farkları belirtiniz?
- 7.6.4) Kesme saklayıcılarını listeleyiniz?
- 7.6.5) Standart 8051 mikrodenetleyicisinde kaç adet kesme kaynağı vardır?
- 7.6.6) Standart 8051'deki kesme mekanizmasını şematik olarak çiziniz?
- 7.6.7) Sadece T1 kesmesinin yetkilendirilmesi için IE saklayıcısının değeri ne olmalıdır?
- 7.6.8) T0 kesmesinin  $\overline{INT0}$  kesmesinden daha öncelikli olarak çalışabilmesi için gerekli saklayıcı ayarlarını yapınız?
- 7.6.9) Bir lastik üretim tesisinde, üretim bandından dakikada geçen lastik miktarı ölçülmek istenmektedir. Bir algılayıcı her lastik sonrası bir kare darbesi üretmektedir. Tasarlayacağınız sistem üretilen bu darbeyi alarak ve bir dakikada geçen lastik miktarını Port-1 in çıkışına verecektir. Port-1 deki sayısal değer ikili olacaktır?



- 7.6.10) Yukarıdaki sistemde görüldüğü gibi T0'a 10 Hz'lik ve T1'e 50 Hz'lik kare dalga üretici bağlıdır. Y ve H butonlarına basıldığı sürece üreteçler kare dalga üretmektedirler. Y yavaş sayma, H butonu hızlı saydırma için kullanılacaktır. Butona basılı kaldığı sürece sayma işlemi P2 de ikili olarak sürecektir; basılmadığı durumda ise sayma duracaktır. Sistem için gerekli *assembly* programını kesme kullanarak yazınız?
- 7.6.11) 12 MHz ile çalışan bir 8051 de, P1.0 port ucundan 100 KHz'lik bir kare dalga elde etmek için gerekli programı Timer 0 ve ilgili kesme mekanizmalarını kullanarak *assembly* dilinde yazınız?

# 8051 YENİ TEKNOLOJİLERİ

## 8. BÖLÜM

### Amaçlar

- X2 çalışma modu ile X1 modu arasındaki farkları açıklamak
- Çift Veri İşaretçisi (Dual Data Pointer) kullanmak
- Gelişmiş zamanlayıcı/sayıcı, kesme ve bellek birimlerini kullanmak
- Tuş takımı arayüzünün sunduğu kolaylıkları öğrenmek
- Çeşitli popüler seri haberleşme protokollerini ve uygulamalarını öğrenmek
- Tümdevre üstü emülasyon teknolojilerini (ONCE) öğrenmek
- ISP (Sistem üzerinde programlama) teknolojisini uygulamak

### Başlıklar

- 8.1. Giriş
- 8.2. AT89C51RC2 Mikrodenetleyicisi
  - 8.2/a. Hızlı Çalışma Modu
- 8.3. Çift Veri İşaretçisi (Dual DPTR)
- 8.4. Genişletilmiş RAM (XRAM)
- 8.5. Gelişmiş Sayıcı Birimleri (PCA)
- 8.6. Tuş Takımı Arayüzü
- 8.7. Seri Port Arayüzü (SPI)
- 8.8. Gelişmiş Kesme Birimi
- 8.9. Donanımsal Kısır-Döngü Sayacı (Watchdog Timer)
- 8.10. Dahili Emülasyon (ONCE-On Chip Emulation)
- 8.11. EEPROM Veri Belleği
- 8.12. Flash Bellek ve ISP (Sistem Üzerinde Programlama)
- 8.13. Saklayıcılar
- 8.14. Özet
- 8.15. Sorular

### 8.14. Özet

Standart 8051 mikrodenetleyicisi sunmuş olduğu teknolojiler ve birimler mühendislerin daha ekonomik, hızlı, güvenilir ve işlevsel sistemleri tasarlayabilmelerine imkan tanımaktadır. Zenginleştirilmiş zamanlayıcı/sayıcılar, özel sayıcı birimleri (PCA) ve sistem güvenilirliğini artırıcı birimler, çeşitli bellek türleri ve yüksek kapasiteli bellek birimleri, hata bulma ve ayıklama işlemleri için özel çalışma modları (ONCE), gelişmiş kesme senaryoları ihtiyacını karşılayacak özel kesme birimleri ve çeşitli popüler seri haberleşme protokollerine sahip birimler bunlardan sadece bir kaçıdır.

Ayrıca tasarımların kısa sürede geliştirilmesini hedefleyen son derece esnek programlama (ISP) teknolojileri ve mobil uygulamalar için enerji sarfiyatını en aza düşüren çalışma modları öne çıkan özelliklerdendir. Elektronik endüstrisindeki ürün ve üretici çeşitliliği, tasarımcıların ihtiyaçlarını karşılayacak ürün ve üreticiyi seçme özgürlüklerini ve yeteneklerini artırmaktadır.

İleriki yıllarda mikrodenetleyicilerin, FPGA (Alan Programlanabilir Kapı Dizileri) ve CPLD (Karmaşık Programlanabilir Lojik Tümdevreler) tümdevreleriyle birlikte DSP (Sayısal İşaret İşleme) tümdevrelerinin görevlerini de kapsayacak ve böylece SoC (Entegre Sistem) teknolojisini barındıracak karma sistemlerden oluşacağı açıktır. Böyle bir sistemin geliştirilmesinde, *assembly* ve 8051-C dillerinin yanında donanım tanımlama dilleri olan VHDL, VERILOG veya *System-C* dilleri de popülerliklerini artıracaktır.

### 8.15. Sorular

- 8.14.1) AT89C51RC2 ailesinin standart 8051 ailesinden farklı olan özelliklerini sıralayınız.
- 8.14.2) X2 modunun olumlu yanlarını ve aktif yapılması için gerekenleri sıralayınız?
- 8.14.3) Çift Veri İşaretçisi ile amaçlanan iyileştirme nedir? Hangi saklayıcı çift veri işaretçisinden yararlanmak için kullanılır?
- 8.14.4) XRAM bellek birimi nasıl aktif yapılır?
- 8.14.5) XRAM bellek birimi kullanıldığında harici veri belleğinin tamamını neden kullanamayız?
- 8.14.6) XRAM bellek birimine ait kontrol saklayıcıları hangileridir? Görevlerini her bir saklayıcı için yazınız?
- 8.14.7) XRAM belleğinin yararları sıralayarak, neden 64KBlık bir veri belleğinin XRAM olarak 8051 tümdevresi içinde olmadığını tartışınız?
- 8.14.8) PCA biriminin kullanılabileceği potansiyel alanlardan örnekler veriniz?
- 8.14.9) PCA biriminin çalışma modlarının ve bunları kontrol eden saklayıcıları açıklayınız?
- 8.14.10) Tuş takımı arayüzü ne amaçla kullanılır? Kontrol saklayıcılarını sıralayınız?

- 8.14.11) SPI seri iletişimine alternatif seri iletişim arayüzü hangileridir? Hangi seri iletişim protokolü en iyidir, tartışınız?
- 8.14.12) SPI protokolünde ortaya çıkabilecek arıza/hata durumları nasıl önlenir?
- 8.14.13) Geliş kesme biriminin, standart kesme biriminden farkı nedir?
- 8.14.14) Gelişmiş kesme biriminde kaç adet öncelik belirlenebilir?
- 8.14.15) Kısır-döngü sayacı ne amaçla kullanılır?
- 8.14.16) Kısır-döngü sayacının aktif veya pasif yapılması için yapılacak işlemleri sıralayınız?
- 8.14.17) ONCE teknolojisi ile amaçlanan kolaylık veya üstünlük nedir?
- 8.14.18) EEPROM belleğin sunmuş olduğu avantajları sıralayınız. XRAM belleğiyle arasındaki ilişkiyi açıklayınız?
- 8.14.19) FLASH bellekler ve ISP teknolojisinin ilişkisini açıklayınız. ISP teknolojisinin üstünlükleri ve kullanılabilmesi için yapılacak işlemleri sıralayınız?
- 8.14.20) *Bootloader* programı nedir? Neden gereklidir? Bellekten nasıl kaldırılır veya tekrar yeniden nasıl yüklenir?



# MİKRODENETLEYİCİDE

## C PROGRAMLAMA

# 9.

## BÖLÜM

### Amaçlar

- Uygulamalar için gerekli 8051 geliştirme araçlarını tanımak/bilgilenmek
- Keil  $\mu$ Vision2 geliştirme yazılımının kullanımını öğrenmek
- ISIS/Proteus elektronik devre çizim ve simülasyon programını tanımak
- 8051 C derleyicisinin desteklediği veri türlerini, bellek türlerini öğrenmek
- 8051 C derleyicisindeki kesme ve C programlama yapısını öğrenmek
- 8051 C derleyicisindeki hazır makro ve fonksiyonları tanımak
- 8051 mikrodenetleyicisi için C tabanlı program yazabilmek

### Başlıklar

- 1.44.Giriş
- 1.45.8051 Geliştirme Araçları
- 1.46.8051'in C Diliyle Programlanması
- 1.47.Özet
- 1.48.Sorular

### 9.4. Özet

Mikrodenetleyicili sistemler uzun zaman *assembly* dili ve pahalı emülatör donanımları ile birlikte geliştirilmiştir. Yüksek düzeyli dillerin (C, Pascal, Basic) derleyicileri 1990'ların başlarında popüler olmaya başlamıştır. Ancak maliyetlerinin yüksek olması amatör tasarımcılar için erişimlerini imkansız hale getiriyordu. Ayrıca donanımsal hata bulma/ayıklama sistemleri olan emülatörlerin maliyetleri de küçük bütçeli tasarımcılar için oldukça fazlaydı.

Kitabımızda tanıtımı verilen, 8051 proje geliřtirmesi ve benzetiminde kullanılan *Keil  $\mu$ Vision* ile *Proteus* firmasının *ISIS* devre benzetim araçları tüm dünyada yaygın olarak kullanılmaktadır. Özellikle  *$\mu$ Vision* geliřtirme ve benzetim yazılımı tüm 8051 geliřtiricileri olarak standart bir yazılım olarak kabul edilmektedir.

*Keil  $\mu$ Vision*, C ve assembly dil desteđine sahiptir ve bir çok benzetim araçlarına sahip olan bir yazılımdır. Mikrodenetleyici, adım adım çalıştırılırken içerisindeki tüm saklayıcılar ve bellek alanları izlenebilir ve çalışmaya ait tüm süreler gerçek zamanlı olarak kaydedilir. Ayrıca mikrodenetleyicilerin programlanmasında kullanılan *INTEL* hex uzantılı dosyalar bu yazılım ile otomatik olarak üretilir.

8051 kodlarının çalışması *Keil  $\mu$ Vision*'da test edildikten sonra, tasarımın tamamlanabilmesi için ilgili kodun mikrodenetleyiciye yüklenmesi ve gerçek devre üzerinde denenmesi gerekir. Ancak bu aşama oldukça zaman aldığından, devrenin yazılımsal bir şekilde üretilip geliřtirilen kodun bu devre üzerinde denenmesi daha kısa sürede sistemin tasarımını gerçekleřtirmemize imkan tanır. *ISIS* bir ECAD (Electronic Computer Aided Design- Elektronik devre Bilgisayar Destekli Tasarımı) programıdır ve diđer bir çok ECAD programlarının başaramadığı bir özellik taşır: Mikroişlemci veya Mikrodenetleyicili sistem benzetimi. Aslında řu şekilde bir kanı çok da haksız sayılmaz: "Tek başına bu özellik, *ISIS* programının popülerliğini artırmıştır". *ISIS* programında benzetime tutulan sistem, kod yükleme aşamasından sonra adım adım çalıştırılarak sonuçlar izlenebilir, istenirse belli bir adreste sistem durdurularak saklayıcılardaki veya veri belleklerindeki deđerler incelenebilir.

8051-C dili aslında standart C dilinin bir türevidir, bununla 8051-C dili, C dilindeki bir çok yapıya sahip olmakla birlikte mikrodenetleyicilerin özel işlevlerine hizmete edecek fonksiyon, yapı ve kütüphanelere sahiptir. Örneđin bit tabanlı fonksiyonlar, bit tabanlı veri türleri standart C dilinde olmayan özelliklerdir. Aynı şekilde bir çok farklı bellek alanlarının kullanımı ve bunlara ait tanımlar da standart C dilinde bulunmaz. Bazı fonksiyonlarda mikrodenetleyicili sistemde aynı sonucu veremeyebilir. Örneđin, *printf* fonksiyonu bilgisayar ekranında istenen karakter veya sayıları yazarken, 8051-C de *printf* fonksiyonu (ekran olmadığı için) çıktıları seri haberleşme portuna yönlendirir.

## 9.5. Sorular

- 9.5.1) *Assembly* ve C dillerini yapısal olarak karşılaştırınız. C dilinin üstünlüklerini maddeler halinde yazınız?
- 9.5.2) Bir mikrodenetleyici geliřtirme yazılımından beklenen işlevler nelerdir?
- 9.5.3) C derleyicisinde bir proje geliřtirirken neden standart fonksiyonları kullanmaya çalışmalıyız?
- 9.5.4) ANSI C dilinden farklı olarak C51 derleyicisinde kullanılan veri türleri nelerdir?
- 9.5.5) C51 derleyicisinde bit deđerşkenler nasıl tanımlanır? Bir örnekle açıklayınız?

- 9.5.6) *signed char* ve *unsigned char* arasındaki fark nedir?
- 9.5.7) C51 derleyicisinde sbit değişkenler nasıl tanımlanır ve ne amaçla kullanılır? Bir örnekle açıklayınız?
- 9.5.8) C51 derleyicisinde bir değişkenin program belleğinde yer alması için hangi komut kullanılır?
- 9.5.9) C51 derleyicisinde dahili RAM belleğindeki alanlara erişim komutlarını birer örnekle gösteriniz?
- 9.5.10) C51 derleyicisinde harici RAM belleğindeki alanlara erişim komutlarını birer örnekle gösteriniz?
- 9.5.11) C51 derleyicisinde kullanılan bellek modellerini kısaca açıklayınız?
- 9.5.12) C51 derleyicisinde kesme fonksiyonları nasıl tanımlanır? Bir örnekle gösteriniz?

# C İLE 8051 UYGULAMALARI

## 10. BÖLÜM

### Amaçlar

- 8051 mikrodenetleyicisi için temel giriş/çıkış işlemlerini kullanabilmek
- 7-parçalı göstergenin sürülmesini kavramak
- Zamanlayıcı/sayıcı birimini kullanabilmek
- Seri haberleşme işlemini öğrenmek
- Kesme uygulamalarını kavramak
- Uygulamalarda gösterge elemanı olarak LCD kullanabilmek
- Bellek kontrolü ile ilgili işlemleri kavramak
- Tuş takımı uygulamalarını kavramak
- Analog/sayısal ve sayısal/analog çeviriciyi kavramak
- DC, servo ve adım motorları kontrol edebilmek

### Başlıklar

- 10.1. Giriş
- 10.2. Basit Giriş/Çıkış İşlemleri
- 10.3. 7-Parçalı Gösterge (Display) Uygulamaları
- 10.4. Zamanlayıcı/Sayıcı Uygulamaları
- 10.5. Seri Haberleşme Uygulamaları
- 10.6. Kesme Uygulamaları
- 10.7. LCD Uygulamaları
- 10.8. Bellek Uygulamaları
- 10.9. Tuş Takımı (Keyped) Kullanımı
- 10.10. Analog/Dijital Çevirici (ADC) ve Dijital/Analog Çevirici (DAC) Kullanımı
- 10.11. Motor Uygulamaları
- 10.12. Özet
- 10.13. Sorular

## 10.12. Özet

Bu bölüme kadar anlatılan tüm konular, bir mikrodenetleyicili sistem tasarımı yapılabilmesi için gerekli alt yapıyı oluşturmayı amaçlamıştır. Genel olarak bu aşamadan sonra okuyucuların öğrenmiş oldukları temel teorik bilgileri, uygulama alanına dökmeleleri istenir. Ancak çoğu okuyucu güzel örnekler görmeden uygulama geliştirme cesaretini kendinde bulamaz. İşte bu bölüm öğrenci veya tasarımcıların bu korkularını azaltmak veya tamamen yok etmek üzere hazırlanmıştır. Basitten karmaşığa doğru onlarca gerçek tasarım örnekleri ile zenginleştirilen bu bölüm, tasarımcıların kendilerine güven duymasını sağlayacak şekilde düzenlenmiştir.

Her örnek, endüstriyel bir tasarımın parçası veya bütünü olacak şekilde düşünülmüş ve devrenin çalışması adım adım açıklanmıştır. Ayrıca tasarıma ait kodlar açıklama satırları ile desteklenerek, yazılan kodların anlaşılabilirliği artırılmaya çalışılmıştır.

Bu bölümde anlatılan tüm örneklerin C kodları ve devre şemaları Kaynak CD’de ve kitabınızın Web sayfasında sunulmuştur.

## 10.13. Sorular

- 10.13.1) 8051’in port uçlarını giriş olarak ayarlamak için neler yapılmalıdır? Neden?
- 10.13.2) 8051’in P1 ve P2 portlarını kullanarak 74138 tümdevresinin fonksiyonun gerçekleştirecek programı 8051-C dilinde ifade ediniz.
- 10.13.3) 8051’in P1 ve P2 portlarını kullanarak 74244 tümdevresinin fonksiyonun gerçekleştirecek programı 8051-C dilinde ifade ediniz.
- 10.13.4) 8051’in P1 ve P2 portlarını kullanarak 3x8 çoklayıcı tümdevresinin fonksiyonun gerçekleştirecek programı 8051-C dilinde ifade ediniz.
- 10.13.5) P0 portunun 0-3 (LSB-MSB) numaralı uçlarından alınan ikilik sayının onaltılık sayı sistemindeki karşılığını 7-parçalı gösterge ile gösteren sistemi tasarlayınız.
- 10.13.6) T0 ve T1 Z/S birimini kesme meakinazması olmaksızın kullanarak, bir push-butona 5 saniyede kaç kez basıldığını iki adet 7-parçalı gösterge üzerinde gösteren sistemi tasarlayınız.
- 10.13.7) 19200 baud hızında ve 8-bit veri 1-bit tek eşlik biti içeren seri haberleşme altyapısını hazırlamak için yazılması gereken minimum C kodunu gösteriniz.
- 10.13.8) 10.13.5’inci soruyu kesme mekanizmalarını kullanarak çözünüz.
- 10.13.9) Kitapta anlatımı yapılan metin tabanlı LCD ekranı kullanarak adınızın sağdan sola ve soldan sağa kaydığı bir uygulama tasarlayınız.

C ile 8051 Mikrodenetleyici Uygulamaları

- 10.13.10) 8051'e harici veri ve kod bellekleri bağlayarak, ISIS programında "Logic Analyzer" cihazı ile erişim işaretlerini, adres ve veri değerlerini görmeye çalışınız.
- 10.13.11) Tuş takımı ve bir LED kullanarak şifreli kapı örneği tasarlayınız. Şifre '#' karakterleri arasına dört sayının girilmesi ile ayarlanır. Üç kere yanlış girildiğinde şifre girişi iptal edilir.
- 10.13.12) Bir 8-bitlik ADC biriminden gelen sayısal verileri bir saniyede bir LCD ekranda kayarak gösteriniz.
- 10.13.13) Step ve servo motor arasındaki farklar nelerdir?
- 10.13.14) Bir step motorun daha hızlı dönmesi hangi parametrelere bağlıdır?

**8051**

## **PROJE UYGULAMALARI**

**11.  
BÖLÜM**

### **Amaçlar**

- 8051 mikrodenetleyicisi kullanım alanlarını öğrenmek
- 8051 ile birlikte yeni teknolojileri kullanarak (I<sup>2</sup>C) uygulama geliştirmek
- 8051 mikrodenetleyicisinin kullanıldığı proje örneklerini öğrenmek

### **Başlıklar**

- 1.49. Giriş
- 1.50. LCD Kullanarak Tarih-Saat Uygulaması
- 1.51. I<sup>2</sup>C Protokolünü Kullanarak Termometre Uygulaması
- 1.52. 5x7 Matris LED Kullanarak Kayan Yazı Uygulaması
- 1.53. PS/2 Klavyesinden Metinsel Ekran LCD'ye Yazı Uygulaması
- 1.54. Programlanabilir Güç Kaynağı (0-5 V) Uygulaması
- 1.55. Özet
- 1.56. Sorular

### **11.7. Özet**

Bu bölümde kapsanan ve özellikle üniversite son sınıf öğrencilerinin bitirme projeleri düzeyinde üretilen sistem tasarım örnekleri, tasarımcı ve öğrenciler için gerçek hayata ait projelerin neler içerebildiğine dair belirli bir düzeyde fikir vermektedir. Projelerde üretilen C kodlarının mükemmel olmadığını bilincinde olarak, tasarımcı ve öğrencilerden alternatif algoritmalar veya kod iyileştirme önerilerini [www.8051turk.com](http://www.8051turk.com) sitesindeki forum kanallarında dile getirmelerini bekliyoruz.

Bu bölüm ile tasarımcıların, mikrodenetleyicili sistem tasarımlarının donanım yoğunluklu emekten yazılım yoğunluklu emeğe kaydığını görmeleri beklenir. Özellikle yeni teknolojiler, mikrodenetleyicili sistem tasarımı alanında donanımsal e-

međi minimuma çekerken yazılımsal emeđi ön plana çıkaracaktır. Ayrıca gelişmiş benzetim sistemlerinin hem donanımsal hem de yazılımsal tasarıma büyük kolaylık getirdiđi şüphesizdir. Ancak tabiatıyla donanımın benzetimi daha büyük avantajlar ve kolaylıklar getirmektedir.

8051-C dilinin mikrodenetleyicili sistem tasarımında ne kadar etkili ve verimli olduđu bu proje örneklerinden de anlaşılacaktır. Aksi halde bu projelerin gerçekleştirilmesi *assembly* dili ile haftalar hatta aylar alabilirdi.

### 11.8. Sorular

- 11.8.1) Tarih-saat uygulamasını grafik tabanlı LCD kullanarak gerçekleştirmek için yapılması gereken deđişiklikleri belirtiniz?
- 11.8.2) Kayan yazı uygulamalarında, yazı kaymanın daha düzgün olması hangi parametrelere bađlıdır? Çözüm önerileriniz nelerdir?
- 11.8.3) Fare kullanarak grafik LCD üzerinde çizim yapabilmek için gerekli olan fonksiyonlar nelerdir, tartışınız ?
- 11.8.4) Dört-işlem hesap makinesine trigonometrik fonksiyonları nasıl hesap ettirebilirsiniz?
- 11.8.5) Programlanabilir güç kaynađını rasgele işaret üretici olarak nasıl kullanabiliriz?
- 11.8.6) Kayan yazı sistemlerine, yeni verilerin girilmesi için kullanılacak kablosuz haberleşme protokollerinden (Bluetooth, IrDA, WLAN) hangisinin uygun olacağını tartışınız (maliyet, yapılabirlik,mesafe,güvenlik)?



## KAYNAKÇA

- Scott, M., “The 8051 Microcontroller”, Prentice-Hall.
- Matic, N., Verle, M., “Architecture And Programming Of 8051 Microcontroller”, Mikroelektronika.
- Ayala, K., “The 8051 Microcontroller Architecture, Programming and Applications”, West Publishing Company.
- “8051 Data Book”, Philips®.
- Yeralan, S., “Programming and Interfacing The 8051 Microcontroller”, Addison-Wesley.
- “AT89CRC2/RD2/ED2 Data-sheet”, *Atmel*®.
- Ekiz, H. “Mantık Devreleri”, Değişim Yayınevi, 2003.

**DİZİN**

$\mu$ Vision2	216	AT89C51RC2 geliştirme kartı	215
7-parçalı gösterge	267, 268, 272	AT89C51RC2 mikrodenetleyicisi	165
8051 geliştirme araçları	215		
genel yapısı	26	<b>B</b>	
iç yapısı	28	bağlı adresleme	79
kesme özellikleri	151	baud üretmeç modu	128
komut türleri	83	BCD sayıcı	256
özellikleri	27	bellek erişim modları	244
port yapısı	36	bellek haritası	49
program belleği	42	bellek organizasyonu	17, 57, 63231
uçları		bellek uygulamaları	323
veri belleği	44	bit işlem komutları	97
yeni teknolojileri	163	bit	226
8051'in C ile programlanması	225		
9-bit UART modu	137	<b>C-C</b>	
		C programlama	214
<b>A</b>		CISC	18
ADC 0808	337	CPU	16, 28
ADC	32, 334	çalışma ilkeleri	192
adım motor	355	çevresel birimler	30
adres hatlarının tespiti	65	çift veri işaretçisi	169
adres sınırlarını belirleme	64	çok işlemcili haberleşme	140
adresleme yöntemleri	75		
bağlı adresleme	79	<b>D</b>	
doğrudan adresleme	77	DAC	334, 343
dolaylı adresleme	77	dahili emülasyon	206
index adresleme	82	dahili RAM	44
ivedi adresleme	79	dahili veri belleği	231, 323
mutlak adresleme	81	dallanma komutları	98
saklayıcı adresleme	76	darbe genişlik üretici	183
sıralı adresleme	82	DB	74
uzun adresleme	82	DC motor	346
alt program çağırma	101	değişken baud hızı	136
alt programdan dönme	101	denetim yapıları	237
analog dijital dönüştürücü	337	directives	74
arayüz	185, 189	display	267, 272
aritmetik komutlar	85	doğrudan adresleme	77
assembly dili	72	dolaylı adresleme	77
aşağı/yukarı sayıcı	274		

## C ile 8051 Mikrodenetleyici Uygulamaları

do-while deyimi	241	kısır-döngü sayacı	184
döngü yapıları	240	kod belleği	231
<b><u>E-F</u></b>		kontrol uçları	32
EEPROM	29, 207	kontrol yapıları	237
END	74	LCD uygulaması	392
EPROM	29	LCD-grafiksel	312
EQU	75	LCD-metin	297
Flash bellek	29, 208	LED	255
float	228	loop	240
for deyimi	242	<b><u>M-N</u></b>	
<b><u>G-H</u></b>		makrolar	244
G/Ç işlemleri	254	mantıksal komutlar	89
G/Ç uçları	35	matris LED kullanımı	259
gecikme üretme	284	MİB	16
gelişmiş kesme birimi	199	mikrodenetleyici birimleri	15
gelişmiş sayıcı birimleri	174	mikrodenetleyici blok diyagramı	15
genel amaçlı saklayıcılar	47	mikrodenetleyici işlevleri	15
genişletilmiş RAM	171	mikrodenetleyici mimarisi	17
giriş/çıkış portları	31	Mod 0-13 bit zamanlayıcı/sayıcı	112
güç kaynağı uygulaması	399	Mod 1-16 bit zamanlayıcı/sayıcı	113
haberleşme hızı	191	Mod 2-8 bit otomatik yükleme	115
harici veri belleği	56, 325	Mod 3 - ayrık modu	116
hata durumları	195	motor uygulamaları	364
hazır fonksiyonlar	244	mutlak adresleme	81
hızlı çalışma modu	166	<b><u>O-Ö</u></b>	
<b><u>I-İ</u></b>		ORG	74
if deyimi	237	ortak anotlu/katodlu	267
index adresleme	82	osilatör girişleri	35
interrupts	150	otomatik yükleme	126
ivedi adresleme	79	OTP	29
<b><u>K-L</u></b>		öncelik	153
kara dalga üretme	282	ötelemeli saklayıcı modu	135
kayan yazı uygulaması	382	özel fonksiyon saklayıcıları	48
kesme	150	<b><u>P-R</u></b>	
kesme uygulamaları	290	PCA yakalama modu	181
kesme vektör adresleri	156	PLC	23
kesmede öncelik	153	program belleği	29, 42, 231
		program formatı	73

proje derlemesi 222  
Proteus benzetim yazılımı 224  
RISC 18-20

### S-Ş

saat darbesi 31  
saklayıcı adresleme 76  
saklayıcılar 207, 209  
sayıcılar 107  
sbit 229  
SCON 133  
seri haberleşme 286  
seri işlem portlar 132  
seri port arayüzü 189  
seri port çalışma modları 134  
seri port hız ayarları 141  
seri port saklayıcılarına erişim 136  
servo motor 353  
sfr/sfr16 230  
sıralı adresleme 82  
signed char 227  
signed int 228  
signed long 228  
signed short 227  
sistem belleği 29  
SPI saklayıcıları 197  
step motor 355  
switch deyimini 239

### T

T0/T1 zamanlayıcı/saklayıcı 111  
T2CON saklayıcısı 124  
tarih-saat uygulaması 364  
TCON saklayıcısı 109  
termometre uygulaması 374  
TMOD saklayıcısı 107  
tuş takımı arayüzü 185  
tuş takımı uygulaması 328

### U-Ü

unsigned char 227  
unsigned int 228

unsigned long 228  
unsigned short 227  
uygulama geliştirme yazılımı 216  
uzun adresleme 82  
üst RAM 47

### V-W

vektör adresleri 156  
veri belleği 30, 44, 207  
veri transfer komutları 93, 95  
veri türleri 226  
bit 226  
signed char 227  
unsigned char 227  
signed short 227  
unsigned short 227  
signed int 228  
unsigned int 228  
sbit 229  
float 228  
signed long 228  
unsigned long 228  
sfr 230  
sfr16 230  
virtual terminal 288  
watchdog timer 205  
while deyimini 241

### Y-Z

yakalama modu 126  
yönergeler 74  
yürüyen ışık 258  
zamanlayıcı/sayıcı 278  
zamanlayıcı 107  
zamanlayıcı/sayıcı 2 123