

ARTTIRILABİLİR HAFIZALI MP3 ÇALAR

Projeyi Yapan : Alen BARDİZBANYAN

Proje Danışmanı : Prof. Dr. Herman SEDEF

GİRİŞ

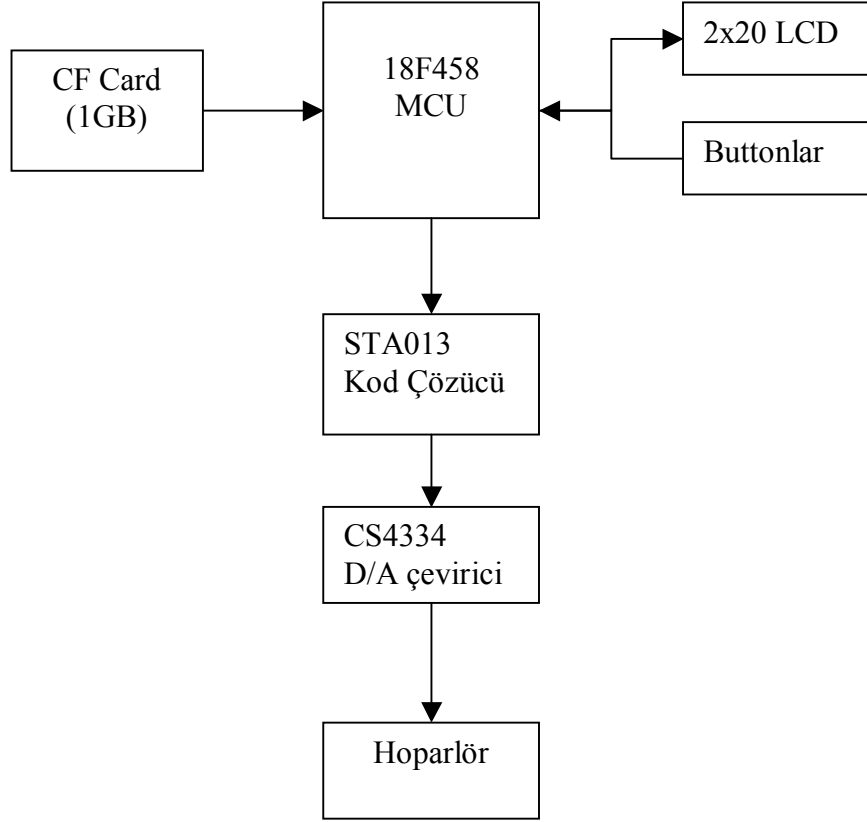
Günümüzde herkesin bilgisayarında yüzlerce hatta binlerce MP3 formatında şarkı bulunmaktadır, çünkü bu şarkıların boyutu çok küçük olduğu için kolaylıkla internet gibi yollardan temin edilebilmektedir. Bunun sonucunda son yıllarda CD çalarlar yerlerini, MP3 çalara devretmiştir. MP3 çalarların kullanımı da çok pratiktir, MP3 çaların hafızasına istediğiniz şarkıyı yükleyip silebilirsiniz ve bu işlem saniyeler sürer. Piyasada bulunan MP3'lerin hafızası genellikle sabit olmaktadır, fakat bu projede MP3 çaların hafızası dışarıdan takılan Compact Flash kart olduğu için sistemin hafızası istendiği zaman arttırılabilir, bugün 16GB boyutuna kadar Compact Flash kartlar üretilmektedir, yani şu an MP3 çaların hafızası 16GB'a kadar arttırılabilmektedir.

Bu projede arttırılabilir hafızalı bir MP3 çalar tasarlanıp gerçekleştirilmiştir, ve ayrıca projede kullanılan büyük sayılabilecek parçalar (mikrodenetleyici gibi) çok küçük boyutlarda da üretildiği için bu projede tasarlanmış olan MP3 çalar çok rahat bir şekilde portatif hale getirilebilir.

MP3 çaların hafıza birimi olarak Compact Flash kart seçilmiştir, çünkü Compact Flash kartlardan oldukça hızlı bir şekilde bilgi okunabilmektedir, bilginin hızlı okunması bitrate'i yüksek olan MP3'lerin sorunsuz bir şekilde çalınabilmesi için oldukça önemlidir. İstenilen şarkılar Compact Flash karta yüklenmektedir, daha sonra bu kart MP3 çalara takıldığında MP3 çalar bu karttaki şarkıları okuyup çalabilmektedir. MP3 çaların şarkıları okuyabilmesi için Compact Flash kart FAT16 dosya sistemine göre formatlanmış olmalıdır.

MP3 kod çözme işlemini STA013 entegresi gerçekleştirmektedir. Ona bağlı olan CS4334 sayısal-analog dönüştürücü sayesinde de ses çıkışı sağlanmaktadır. Compact Flash'taki MP3 bilgisini kod çözücüye gönderme, butonlardan gerekli bilgileri alma ve LCD ekranı sürme işlemleri ise Microchip firmasının oldukça gelişmiş bir mikrodenetleyicisi olan 18F458 mikrodenetleyicisi ile gerçekleştirilmektedir. Yazılım hızlı çalışması için ve sisteme tamamen hakim olabilmek için Microchip firmasının kendi assembly dilinde yazılmıştır.

SİSTEMİN BLOK DİYAGRAMI



Şekil 1 Sistemin Blok Diyagramı

Sistemin en temel hali yukarıda gösterildiği gibidir. Mikrodenetleyicinin çıkışları sınırlı olduğu için LCD ve Compact Flash aynı Data(bilgi) yolunu kullanmaktadır. Bu çok kolay bir şekilde sağlanabilmektedir çünkü LCD aktif 1 ile Compact Flash ise aktif 0 ile aktif hale gelmektedir bu yüzden LCD ve Compact Flashın aktif etme uçlarını aynı yere bağlandığı zaman %100 güvenli bir çalışma elde edilmektedir.

SİSTEMİN ÇALIŞMA ŞEKLİ

MP3 çalar açıldığı zaman, Compact Flash karttan şarkıların başlangıç adreslerini bulmaktadır, daha sonra ekranda ilk şarkının ve artistin ismi yazmaktadır, kullanıcı buttonları kullanarak istediği şarkıyı seçip çalabilmektedir.

FAT16 DOSYA SİSTEMİ

FAT16 dosya sistemi 1987 yılında ortaya çıkarılmış bir dosya sistemidir. O zamanlar hafıza birimlerinin boyutu küçük olduğu için bu dosya sistemi yeterliydi fakat hafıza birimlerinin boyutları büyüdükçe bu dosya sistemi yetersiz kaldı ve performansı arttırabilmek ve büyük boyutlu hafıza birimlerini daha verimli kullanabilmek için FAT32 ve NTFS dosya sistemleri geliştirildi. Bu projede FAT16 dosya sisteminin seçilmesinin sebebi 16bit adresleme yaptığı için yazılımda kolaylık sağlamaktadır. Eğer projeyi hazırlamak için daha uzun bir vakit olsaydı yazılım FAT32 ve NTFS için de geliştirilebilirdi.

FAT16 dosya sisteminde hafıza birimi Cluster'lara bölünmüştür. Bu hafıza birimindeki cluster'ların boyutu Hafıza biriminin toplam kapasitesine bağlıdır.

Volume Size	FAT16 Cluster Size	FAT32 Cluster Size	NTFS Cluster Size
7MB - 16MB	2KB	Not supported	512 bytes
17MB - 32MB	512 bytes	Not supported	512 bytes
33MB - 64MB	1KB	512 bytes	512 bytes
65MB - 128MB	2KB	1KB	512 bytes
129MB - 256MB	4KB	2KB	512 bytes
257MB - 512MB	8KB	4KB	512 bytes
513MB - 1GB	16KB	4KB	1KB
1GB - 2GB	32KB	4KB	2KB
2GB - 4GB	64KB	4KB	4KB
4GB - 8GB	Not supported	4KB	4KB
8GB - 16GB	Not supported	8KB	4KB
16GB - 32GB	Not supported	16KB	4KB
32GB - 2TB	Not supported	Not supported	4KB

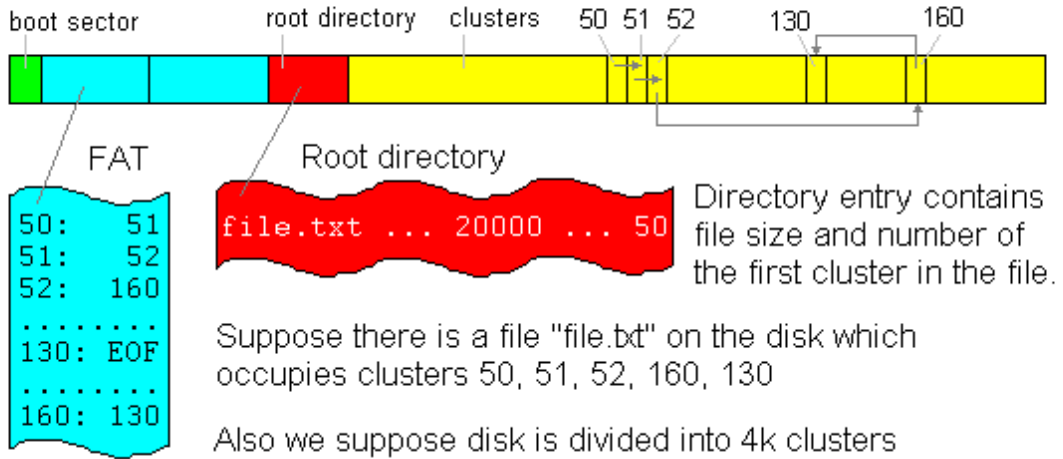
Tablo 1 Dosya Sistemlerine Göre Cluster Büyüklükleri

Görüldüğü gibi FAT16 dosya sistemi yaratıldığı zamanlar çok büyük boyutta hafıza birimleri bulunmadığı için 4GB'dan sonrasına FAT16 dosya sistemi destek vermemektedir.

FAT16 dosya sisteminde dosyalar clusterlara yüklenmektedir. Örneğin bu projede kullanılan Compact Flash kart 1GB'dır bu yüzden 1 cluster'ın boyutu 16KB'tır.Yani dosya 16KB'lık parçalara ayrılarak yüklenmektedir. Örneğin 2KB'lık bir dosya saklanmak istensin bir Cluster 16KB'dır bu 2KB'lık dosya yüklendikten sonra kalan 14KB artık kullanılamamaktadır FAT16 dosya sisteminin en önemli dezavantajı budur. Bu örnekte de görüldüğü gibi 2KB'lık bir dosya için 16KB'lık yer harcanmaktadır. Sadece 2KB da değil 1byte'lık bir bilgi için de 16KB'lık yer kaplanmaktadır. Örneğin 34KB'lık bir bilgi için 3 tane cluster kullanılmaktadır. İki tanesi tamamen dolarken bir tanesinin 34-32=2KB'lık bir kısmı dolmaktadır, bu durumda yine 14KB'lık hafıza boşa gitmektedir.

Eğer hafıza biriminde fragmantasyon yapılmışsa dosyaların yerleştiği Clusterlar lineer şekilde artmaktadır örneğin 34 KB'lık dosya 10., 11. ve 12. Clusterlara yerleşmektedir.Fakat hafıza birimi formatlanmadan içinden dosya silip ekledikçe bu clusterlar lineer şekilde artmayabilir , bu durumda bir dosyanın sırasıyla hangi clusterlarda bulunduğu FAT bölgesindeki pointerlarla belirtilmektedir.Bu projede Hafıza birimi Fragmantasyonlu şekilde kullanıldığı için bu pointerlara bakma ihtiyacı yoktur.

FAT-16



Şekil 2 FAT16 Dosya sisteminin yapısı

Yukarıdaki şekilde FAT16 dosya sisteminin tam yapısı görülmektedir. İlk sektörde boot sektörü bulunmaktadır. Bu sektör içerisinde bilgisayarı başlatmak için gerekli olan bazı kodlar ve partitionlar hakkında bilgi bulunmaktadır.

Offset	Description	Size
000h	Executable Code (Boots Computer)	446 Bytes
1BEh	1st Partition Entry (See Next Table)	16 Bytes
1CEh	2nd Partition Entry	16 Bytes
1DEh	3rd Partition Entry	16 Bytes
1EEh	4th Partition Entry	16 Bytes
1FEh	Executable Marker (55h AAh)	2 Bytes

Tablo 2 Boot Sectorunun içeriği

Offset	Description	Size
00h	Current State of Partition (00h=Inactive, 80h=Active)	1 Byte
01h	Beginning of Partition - Head	1 Byte
02h	Beginning of Partition - Cylinder/Sector (See Below)	1 Word
04h	Type of Partition (See List Below)	1 Byte
05h	End of Partition - Head	1 Byte
06h	End of Partition - Cylinder/Sector	1 Word
08h	Number of Sectors Between the MBR and the First Sector in the Partition	1 Double Word
0Ch	Number of Sectors in the Partition	1 Double Word

Tablo 3 Partition Girişinin İçeriği

Bu partition entry'si okunarak Partitionun tam olarak nerden başladığı öğrenilebilir.

02H adresindeki 16 bitlik Beginning of partition Cylinder/Sector bilgisi,

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Cylinder Bits 7 to 0								Cylinder Bits 9+8		Sector Bits 5 to 0					

Şekil 3 Partition Başlangıcı Adresi Bilgisi

Yukarıdaki şekle göre değerlendirilmelidir. Dikkat edilmesi gereken durum Cylinder adresi karışık bir şekilde yazılmıştır. MSB olan iki bit 7 ve 6 da bulunmaktadır yazılımda bu doğru bir şekilde okunmalıdır.

Örneğin bu projede bu bilgi

0000000000000001 şeklindeydi yani partition'un başlangıcı 1.sector 0.Cylinder ve Beginning of partition – Head yazmacında ise 0000001 değeri vardı yani Partition tam olarak: 1.Head – 0. Cylinder – 1.Sectorden başlamaktadır.

ÖNEMLİ NOT : Örneğin Compact Flash kartta toplam 16 tane head , 1986 tane Cylinder ve 63 tane de sector olsun , her bir 63 sectorden sonra Head değeri bir artmaktadır ve her bi 16 head değerinden sonra Cylinder değeri 1 artmaktadır yani bunları hiyerarşik olarak sıralamamız gerekirse sıralama aşağıdaki gibidir

Cylinder – Head – Sector

Partition başlangıcı bulunduktan sonra bu başlangıç okunduğunda FAT16 Boot Record'a ulaşılır.

Bu bilginin içerisinde neler olduğu bir sonraki sayfada gösterilmektedir.

Offset	Description	Size
00h	Jump Code + NOP	3 Bytes
03h	OEM Name	8 Bytes
0Bh	Bytes Per Sector	1 Word
0Dh	Sectors Per Cluster	1 Byte
0Eh	Reserved Sectors	1 Word
10h	Number of Copies of FAT	1 Byte
11h	Maximum Root Directory Entries	1 Word
13h	Number of Sectors in Partition Smaller than 32MB	1 Word
15h	Media Descriptor (F8h for Hard Disks)	1 Byte
16h	Sectors Per FAT	1 Word
18h	Sectors Per Track	1 Word
1Ah	Number of Heads	1 Word
1Ch	Number of Hidden Sectors in Partition	1 Double Word
20h	Number of Sectors in Partition	1 Double Word
24h	Logical Drive Number of Partition	1 Word
26h	Extended Signature (29h)	1 Byte
27h	Serial Number of Partition	1 Double Word
2Bh	Volume Name of Partition	11 Bytes
36h	FAT Name (FAT16)	8 Bytes
3Eh	Executable Code	448 Bytes
1FEh	Executable Marker (55h AAh)	2 Bytes

Tablo 4 FAT16 Boot Record

FAT16 boot recorddan önemli bilgiler elde edilebilir. Örneğin Sectors Per Cluster değerine bakarak bir Cluster'ın kaç KB olduğunu anlaşılabilir. Bu projede kullanılan Compact Flash Kart 1Gb olduğu için bu değer 32dir bir sector 512Byte olduğuna göre, $32 * 512 \text{ byte} = 16 \text{ KB}$

yani bir Cluster 16KB'dır. Ayrıca Root Directory'nin yani dosyalar hakkında bilgi barındıran bölümün başlangıç adresini bulmak için buradaki Sector Per Fat bilgisi ve Reserved Sectors bilgisi okunmalıdır. Çünkü Root Directory'nin başlangıç adresini bulmak için Fat16Boot Record adresine , reserved sectors sayısını ve Sectors Per Fat * 2 sayısını eklemeliyiz Sonucunda bulduğumuz adres Root Directory'nin başlangıç adresidir.

Root Directory'nin içerisinde klasörler, dosyaların ismi uzantıları bazı özellikleri ve başlangıç adresleri bulunur. Bunlar aşağıdaki tabloda gösterildiği gibi yerleşmişlerdir.

Offset	Length	Value
0	8 bytes	Name
8	3 bytes	Extension
11	byte	Attribute (00ARSHDV) 0: unused bit A: archive bit, R: read-only bit S: system bit D: directory bit V: volume bit
22	word	Time
24	word	Date
26	word	Cluster (desc. below)
28	dword	File Size

Tablo 5 Directory Tablosu

Tabloda dosya ismi için 8byte'lık bir yer ayrıldığı gözükmektedir. Eğer dosya ismi 8byte'tan büyükse bu durumda bir dosya birden fazla directory tablosu kullanmaktadır bu durumda da Hafıza biriminin alabileceği maksimum dosya sayısı azalmaktadır çünkü Root Directory 16KB'tır bir directory entry'si 32bytedir yani Root Directory'ye toplam 512 entry yapılabilir. Eğer dosya ismi 8 harften fazlaysa bu dosya birden fazla entry tutacaktır. Fakat bütün dosyaları ilk klasöre atmak yerine klasörler yaratılarak onların içine atılırsa otomatik olarak entry sayısı artırılmış olur yeni klasörler sayesinde. Bu projede MP3 çalar dosyaları direk olarak ilk yani ana klasörden okumaktadır, klasör okuma desteği yoktur fakat yaratılabilir. Ayrıca MP3 çalar bir dosyanın mp3 olup olmadığını anlayabilmektedir bu de entrynin 3 byte'lık Extension kısmından anlaşılmaktadır. MP3 dosyaları için bu kısım MP3 şeklindedir, MP3 çalar extension'da MP3 ü gördüğü zaman bunun bir MP3 dosyası olduğunu anlayıp o dosyanın başlangıç adresini kaydederek istendiği zaman ulaşılmasını sağlamaktadır. Başlangıç adresi cluster olarak verilmektedir bu yüzden yazılımla bu clusterın hangi Sector/Head ve Cylinder değerine karşılık geldiği bulunmalıdır.

Örneğin Dosyanın başlangıcı 1.ci clusterda gözükyor olsun. Bu projedeki 1GB'lık compact Flash kart için Root directory 8.ci Head 0.cı Cylinder ve 48.ci Sekötrde bulunmaktadır. Bu başlangıç 0.cı clusterdır. 1.ci cluster için 1 cluster boyutu kadar ileri gitmek gerekmektedir yani 16KB ileri gitmek gerekmektedir.16KB 32 sectordur bu durumda

0.Cylinder,8.Head,48.Sector + 32 Sector işlemi yapıldığında 0.Cylinder, 9.Head, 17. Sektör bulunmaktadır yani dosyanın başlangıcı buradadır. İşlemde 32 sektör ileri gidilmek istenmektedir başlangıç 48.ci sektördür buna 32 sektör eklersek sektör 80 olur 80 tane sektör olmadığı için head sayısı bir attırılmalıdır bu durumda Head sayısı 9 olur ve Sektör sayısı da $80-63=17$ olur.

Dikkat edilmesi gereken önemli bir nokta ise Cylinder ve Head değerleri 0'dan başlarken Sector değeri 1 den başlar , 63 sector olduğu için 1 den 63 e kadardır ,0 dan 62ye kadar değildir.

COMPACT FLASH KART'TAN BİLGİ OKUNMASI

Compact Flash kartlar taşınabilir elektronik cihazlarda kullanılan bir tür flash hafıza birimidir. Özelliği içindeki hafıza yapısı Hard Disklere çok benzerdir. Hard disklerdeki mekanik birimler Compact Flashta elektronik olarak bulunmaktadır. Bunlar Head,Cylinder ve Sector'dür.Bir Compact Flash'ın hafızası bu birimlerden meydana gelmiştir.Bir sector genellikle 512byte büyüklüğündedir.Bir Compact Flash kartta kaç tane Sector,Cylinder veya Head bulunduğu compact Flash kartın hafızasına bağlıdır.Örneğin bu projede kullanılan Compact Flash kart 1Gb'tır. 63 tane sectoru 16 tane Headi ve 1986 tane de cylinderi bulunmaktadır. Elimizdeki Compact Flash'ın içerisindeki Head , Sector ve Cylinder değerlerini bilmiyorsak bu değerleri Compact Flasha Indetify Drive komutunu göndererek de öğrenebiliriz.

Buradan kapasitesini hesaplamak istersek

$63*16*1986*512\text{byte}=1024966656\text{byte}=1000944\text{kbyte}=977\text{mbyte}$ etmektedir

Aslında 1GB olarak satılan Compact Flash kartlar pratikte 977mb'tır, fakat 1GB olarak satılmaktadır.

CompactFlash kartlar'da 11 tane adres pini ve 16 tane de data pini bulunmaktadır. Bilgi okumak için çeşitli adresleme şekilleri kullanılabilir.Bilgi okumak için bu pinlerin hepsini kullanmaya gerek yoktur.Örneğin bu projede Compact Flash kart PC Card Memory modunda çalıştırılacaktır.Bu modda bilgi okuyabilmek için 3 tane adres pinini (A0,A1,A2) ve 8 tane Data pinini(D0,D1,D2,D3,D4,D5,D6,D7) kullanmak yeterli olacaktır.Bunların haricinde çeşitli kontrol uçları da kullanılmaktadır.(Chip Enable gibi)

CompactFlash kartların yapısı HDD'lere çok benzemektedir. Hatta CompactFlash kartlar True Ide modunda çalıştırıldıklarında standart bir ata HDD ile tamamen aynı şekilde çalışmaktadırlar.

CompactFlash kart sektörler bölünmüş durumdadır. Genellikle bir sektör 512 byte büyüklüğündedir. Bu sektörler LBA(Logical Block Adressing) veya Cylinder/Head/Sector adreslemesi kullanılarak ulaşılabilir.

Aşağıdaki tabloda ise CompactFlash kartın içerisinde bulunan yazmaçların nasıl adreslendiği gösterilmektedir.

-REG	A10	A9-A4	A3	A2	A1	A0	-OE = 0	-WE = 0
1	0	X	0	0	0	0	Even RD data	Even WR data
1	0	X	0	0	0	1	Error	Features
1	0	X	0	0	1	0	Sector count	Sector count
1	0	X	0	0	1	1	Sector no.	Sector no.
1	0	X	0	1	0	0	Cylinder low	Cylinder low
1	0	X	0	1	0	1	Cylinder high	Cylinder high
1	0	X	0	1	1	0	Select card/head	Select card/head
1	0	X	0	1	1	1	Status	Command
1	0	X	1	0	0	0	Dup. even RD data	Dup. even WR data
1	0	X	1	0	0	1	Dup. odd RD data	Dup. odd WR data
1	0	X	1	1	0	1	Dup. error	Dup. features
1	0	X	1	1	1	0	Alt status	Device Ctl
1	0	X	1	1	1	1	Drive address	Reserved
1	1	X	X	X	X	0	Even RD data	Even WR data
1	1	X	X	X	X	1	Odd RD data	Odd WR data

Tablo 6 Compact Flash Kartın yazmaç içerikleri

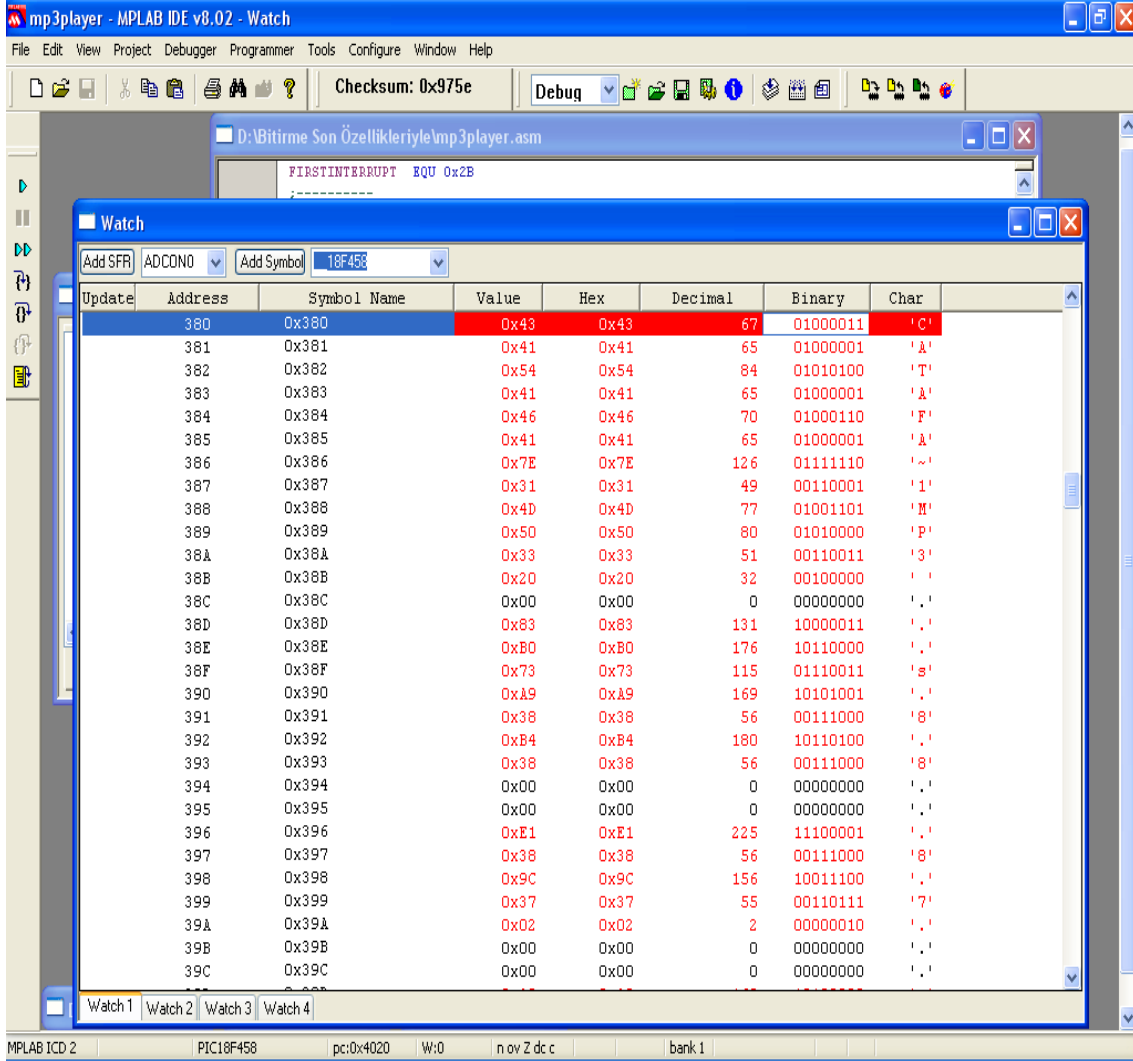
Veri transferi için ilk 8 tane yazmaç yeterlidir bu yüzden sadece 3 adres ucu kullanarak gerekli işlemleri yerine getirebilmekteyiz.

Örneğin bir okuma işlemi yapmak istiyoruz, bunun için WE=0 sütununda yazan yazmaçların içeriğini gerekli bilgilerle dolduruyoruz, bunun için yapılması gereken adres uçlarının gerekli yazmaç seçiciler şeklinde ayarlanması gerekmektedir, daha sonra o yazmaca yüklenecek değer data pinlerine girildikten sonra WE ucu , önce low sonra high konumuna getirildiğinde o yazmaca istenen değer yüklenecektir.Örneğin belirli bir sektörü okumak istiyoruz bunun için önce sektörün adresini yazmaçlara bu şekilde yükleriz daha sonra sektör oku komutunu Command yazmacına girdikten sonra CompactFlash'ın bufferi bu sektörün bilgileriyle dolar ve 000 adresindeki yazmacı , OE ucunu önce low sonra high konumuna getirmeye başlayarak sırayla okumaya başlarız.Örneğin ilk konum değiştirmede 1. byte'ı daha sonra 2.byte'ı ve bu bu şekilde devam etmektedir.

ŞARKI BAŞLANGIÇLARININ BULUNMASI

Daha önceki bölümlerde belirtildiği gibi 1GB'lık Compact Flash kartta, Root bilgisi 8.head, 0.cluster, 48. sektörden itibaren bulunmaktaydı. Yazılımda buradan itibaren 32kb'lık bir bölüm okunarak, ana klasördeki dosyalara bakılır. Eğer uzantısı MP3 olan bir dosyaya rastlanırsa bu şarkının başlangıç adresi 18F458'in kendi hafızasındaki 200H adresinden başlayarak kaydedilir ve her bir şarkı bulunduğu bir yazmaçta bulunan şarkı sayısı bir artırılır. Örneğin 3 tane şarkı varsa bu şarkıların başlangıç adresleri 200H ile 206H arasında kaydedilmiş olur, ve toplam şarkı sayısını saklayan yazmaç 3 olur. Her bir şarkının başlangıç

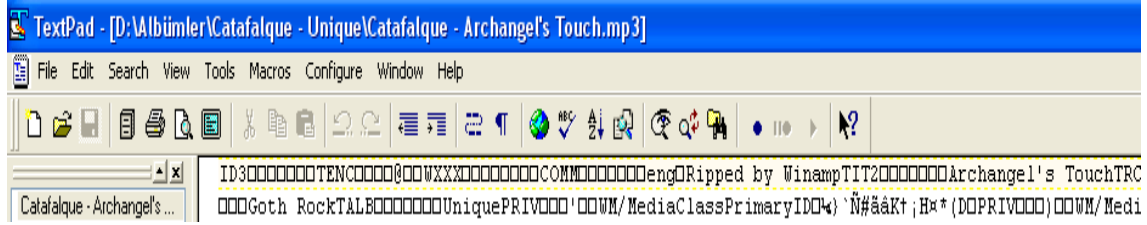
adresi 2 yazmaç kaplamaktadır, yani $2*8 = 16$ bit yer tutmaktadır. FAT16 sistemine adını veren 16 sayısı bu 16bitlik adresten gelmektedir.



Şekil 4 Root Directory'nin belirli bir bölümü

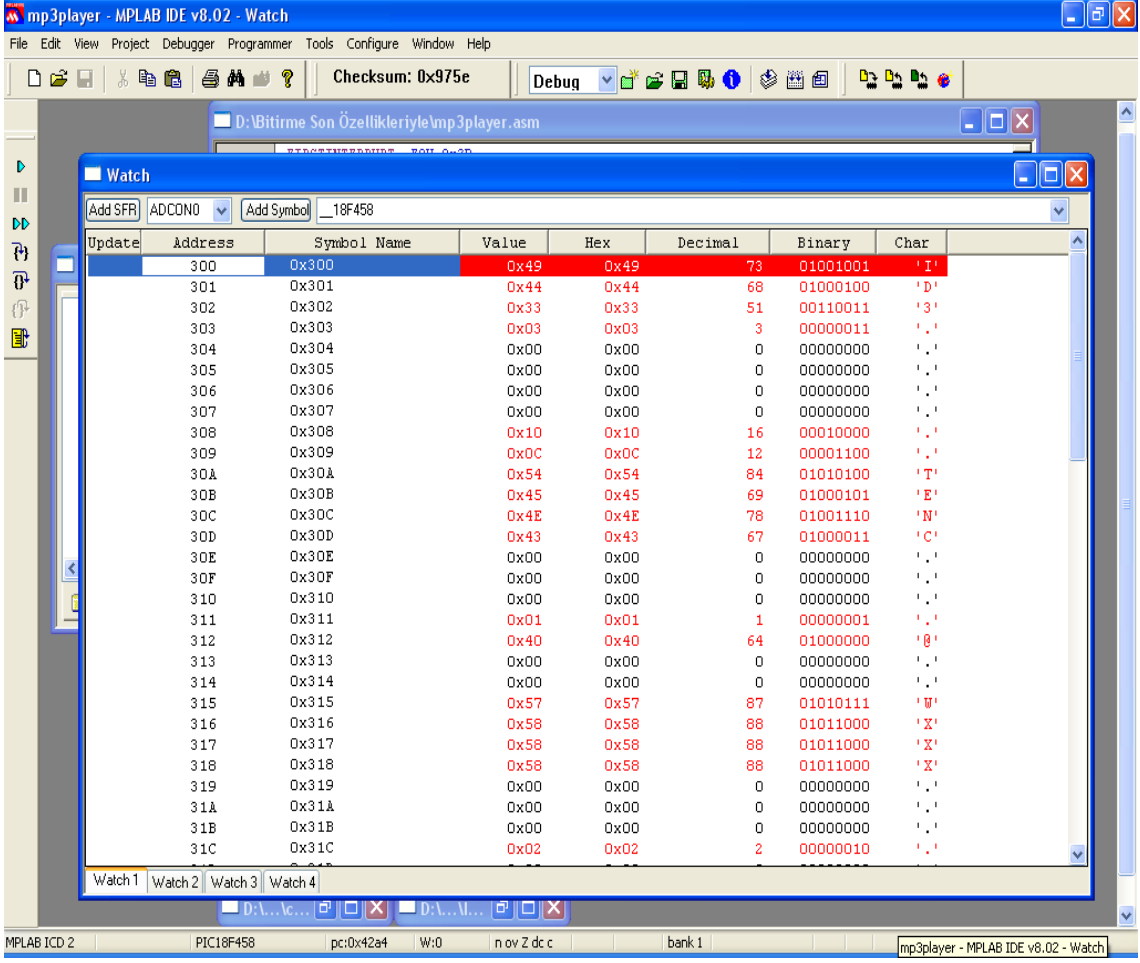
Yukarıdaki şekilde, Microchip'in ICD modülü sayesinde, Mikrodenetleyici çalışırken içerisindeki RAM'den belirli bir bölüm gözükmektedir. Bu bölüm Compact Flash kart'tan okunan Root Directory'nin belirli bir bölümüdür. Bu karttaki ilk şarkıdır, şarkının ismi 8 harften büyük olduğu için CATAFA~1 olarak gözükmektedir, normalde şarkının ismi CATAFALQUE – ARCHANGEL'S TOUCH şeklindedir, şarkı isminin geri kalanı 3 tane entry yazılmıştır, yani bu şarkı tek başına 4 tane entry kaplamaktadır. Şekilden de görüldüğü üzere şarkı isminden hemen sonra MP3 uzantısı yazmaktadır, yazılımda bu MP3 yakalandığı an bir MP3 dosyası bulunduğu anlaşılır. Tablo 5.5 'de bir dosya entry'sinin içeriğinin nasıl olduğu gözükmektedir. Tabloya bakılırsa 26. ve 27. byte'larda şarkının başlangıç adresi tutulmaktadır. Şekilde de görüldüğü üzere bu şarkının başlangıç adresi 2. Cluster olarak

gözükmektedir. Adresin MSB byte'ı 0 LSB byte'ı 2 dir. Yani dosya 2. Cluster'dan başlamaktadır. Her bir Cluster 16KB yani 32 sector kaplamaktadır. 1. Cluster 8.Head 0.Cylinder 48.sector'den başlamaktadır o halde 2. Cluster ise 32 sektör ilerden başlamaktadır yani 9.Head 0.Cylinder ve 17. Sector olmaktadır.



Şekil 5 MP3 dosyasının içeriğinin Microsoft Windows'ta görüntülenmesi

Yukardaki şekilde Mp3 dosyasının başlangıç kısmından bir bölüm gösterilmiştir. Bu içerik Microsoft Windows işletim sisteminde TextPad programı kullanarak görüntülenmiştir. Bir sonraki sayfada ise Microchipin ICD'ını kullanarak Compact Flash kartın 9.Head 0.Cylinder ve 17.Sector'ü okunarak Compact Flash kartın bu adresteki içeriği gösterilmiştir. İçerikle karşılaştırılırsa aynı olduğu gözükcektir , yani bu durumda MP3 'ün başlangıç adresi MP3 player donanımı ve yazılımıyla doğru bir şekilde bulunmuştur ve çalınmaya hazırdır.



Şekil 6 MP3 dosyasının içeriğinin Micorchip ICD yardımı ile çalışan devre üzerinden elde edilmesi

Yukardaki şekil MP3 çalar çalışırken ICD yardımıyla elde edilmiştir, içerik windowstaki ile karşılaştırılırsa ile aynı olduğu görülecektir, yani MP3'ün başlangıç adresi doğru olarak bulunmuştur ve çalınmaya hazırdır.

TAG YARDIMI İLE ŞARKININ, SANATÇININ İSMİ GİBİ BİLGİLERE ULAŞILMASI

Tag bilgisi MP3 dosyasının sonunda bulunan 128 byte'lık standart bir bilgidir. Bu 128 byte'ın içinde MP3'teki şarkıya ait, şarkı adı, sanatçı adı, şarkı yılı gibi bilgiler bulunmaktadır.

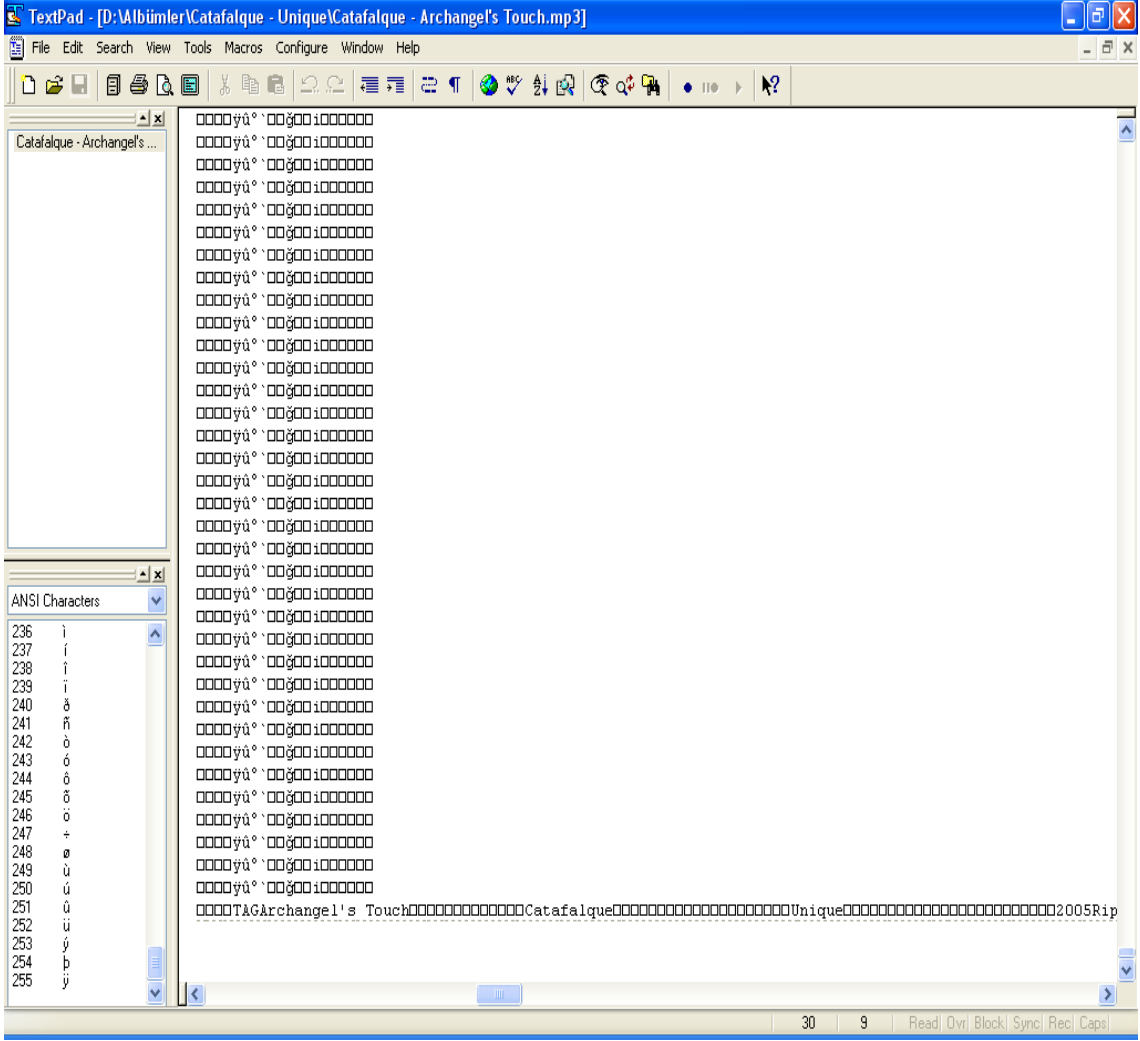
AAABBBBB BBBB BBBB BBBB
BCCCCCCC CCCCCCCC CCCCCCCC CCCCCC
DDDDDDDD DDDDDDD DDDDDDD DDDDEEE
EEEEEEEE FFFFFFFF FFFFFFFF FFFFFFFG

İşaret	Büyükölük	Yerleşimi (byte)	Tanımı
A	3	(0-2)	Tag belirteci."TAG" yazısı kesinlikle olmalıdır bu yazıdan sonra gelenler doğru demektir.
B	30	(3-32)	Başlık
C	30	(33-62)	Artist
D	30	(63-92)	Albüm
E	4	(93-96)	Yıl
F	30	(97-126)	Yorum
G	1	(127)	Tür

Tablo 7 Tag bilgisinin ayrıntıları

Yukarıdaki tabloda bu 128 byte'lik bilginin hangi bitlerinde hangi bilginin bulunduđu belirtilmektedir. Bu bilgiler ASCII olarak kodlanmıştır. MP3 çalarlar bu TAG bilgisini okuyarak ekrana şarkının adı, şarkıcının adı gibi bilgileri yazdırmaktadır.

Bir MP3'ün TAG bilgisi dosyanın sonunda bulunur. MP3 çalar ilk başta bütün dosyaların başlangıç adreslerini kaydetmektedir. Dosyalar lineer bir şekilde sıralandıđı için bir MP3'ten bir sonra gelen MP3 'ün başlangıç adresi aslında o MP3'ün bitiş adresini de vermektedir. Bu yüzden bir MP3 'ün sonuna ulaşmak için kendisinden bir sonra gelen MP3'ün başlangıç adresinden 2 cluster geriye gidilmektedir ve TAG yazısı aranmaktadır. Bulduğunda ise LCD'ye yazdırılmaktadır. Zaten bu bilgi ASCII olarak kodlandıđı için direk olarak LCD'ye gönderilebilmektedir.



Şekil 7 MP3 Tag bilgisinin Microsoft Windows'ta görüntülenmesi

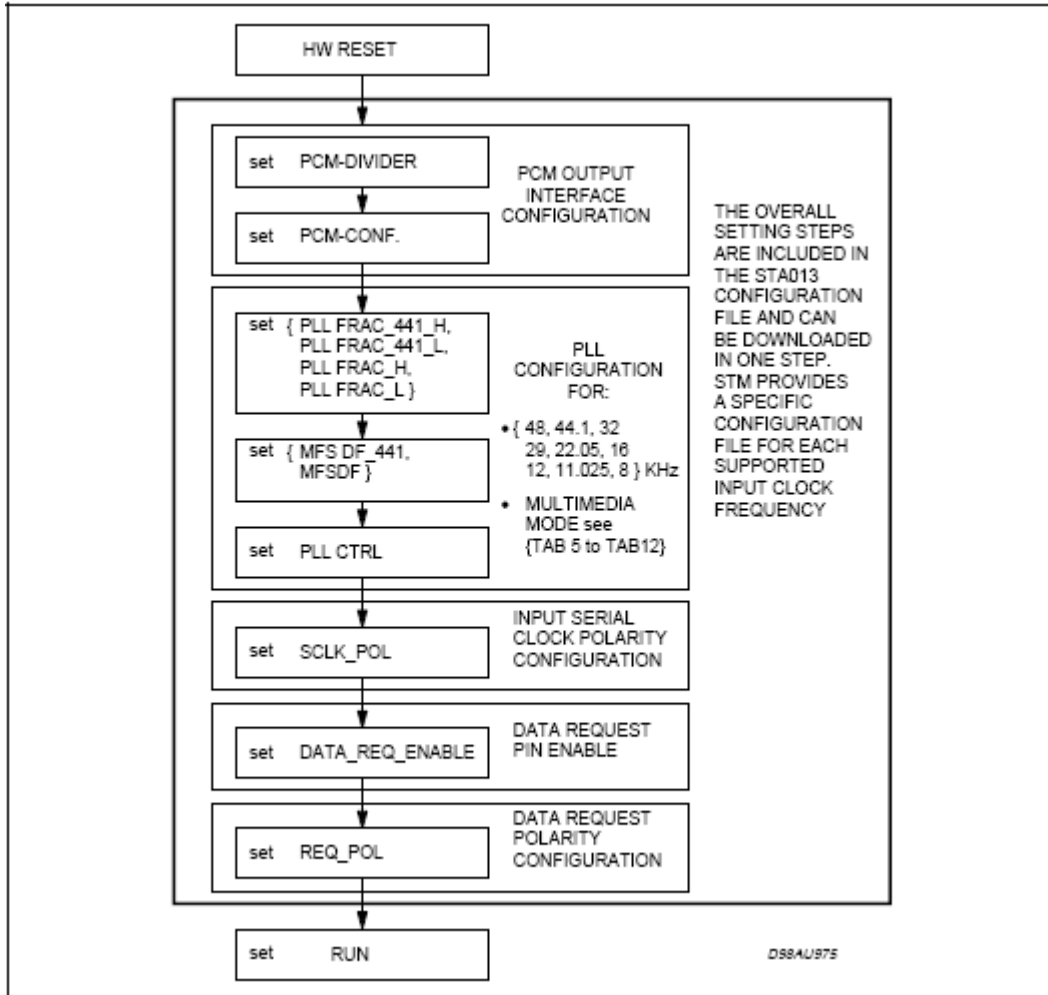
STA013 MP3 KOD ÇÖZÜCÜ ENTEGRESİ

STA013'ün içerisindeki kontrol yazmaçlarına okuma veya yazma yapabilmek için I2C haberleşmesi kullanılmaktadır, MP3 dataları ise SPI haberleşmesi üzerinden gönderilmektedir. I2C haberleşmesi SDA ve SCL pinleri üzerinden, SPI haberleşmesi SDI, SCKR pinleri üzerinden yapılmaktadır.

STA013 kod çözücüsünün iki ayrı modu vardır. Bunlar Broadcast mode ve Multimedia mode'dur, hangi modda çalışacağını SCR_INT pinine 0 veya 1 verilmesi belirler. SCR_INT pinine lojik "1" verilirse Multimedia Modunda, Lojik "0" verilirse Broadcast modunda çalışır. Broadcast modunda çalışırken MP3'ün kbps'a bağlı olarak mikroişlemci MP3 datalarını tam kbps'a bağlı olarak o zamanla göndermelidir. Örneğin MP3 128kbps ise saniyede $128/8=16KB$ veri gönderilmelidir ve bu veri doğru zamanla gönderilmelidir yani $1/16*1024=0,000061$ saniyede 1 byte gönderilmiş olmalıdır. Bu modun kullanımı oldukça zordur çünkü her mp3 için kaç kbps olduğuna bakılıp hesaplamalar vs.. yapılmalıdır.

Multimedia modunda ise STA013 kod çözücüsünün kendi iç buffer'ı devreye girmektedir, STA013'ün bufferı boşken yani veriye ihtiyacı olduğu zaman DATA_REQ pinini kullanıcının isteğine göre 0 veya 1 yapmaktadır. Default olarak bu değer veriye ihtiyacı olduğunda 1 olmaktadır. DATA_REQ ucu 1 olduktan sonra SPI arayüzünden MP3 dataları gönderilmeye başlanmaktadır. Kod çözücü entegre geçerli bir MP3 frame'i yakaladıktan sonra buffer'ını doldurduğunda DATA_REQ ucunu 0 yapmaktadır, DATA_REQ ucu 0 olduğunda artık daha fazla MP3 datası gönderilmemelidir. MP3 dataları gönderilirken eğer kod çözücü entegre play modundaysa bir yandan da gönderilen MP3 datasını çözmeye ve ses çıkışı vermeye başlayacaktır. Bufferı boşalmaya yakın olduğu zaman tekrar DATA_REQ ucu 1 olacaktır ve bu durumda tekrar DATA_REQ ucu 0 olana kadar mp3 datası gönderilmelidir. Bu sayede STA013'e bilgi gönderen mikroişlemcinin MP3'ün kbps'ı ile ilgilenmesine gerek yoktur zamanlamayı Kod çözücü entegre düzenlemektedir. Bu projede de Multimedia modu kullanılmıştır.

STA013 KOD ÇÖZÜCÜ ENTEGRESİNİN KONFIGÜRE EDİLMESİ



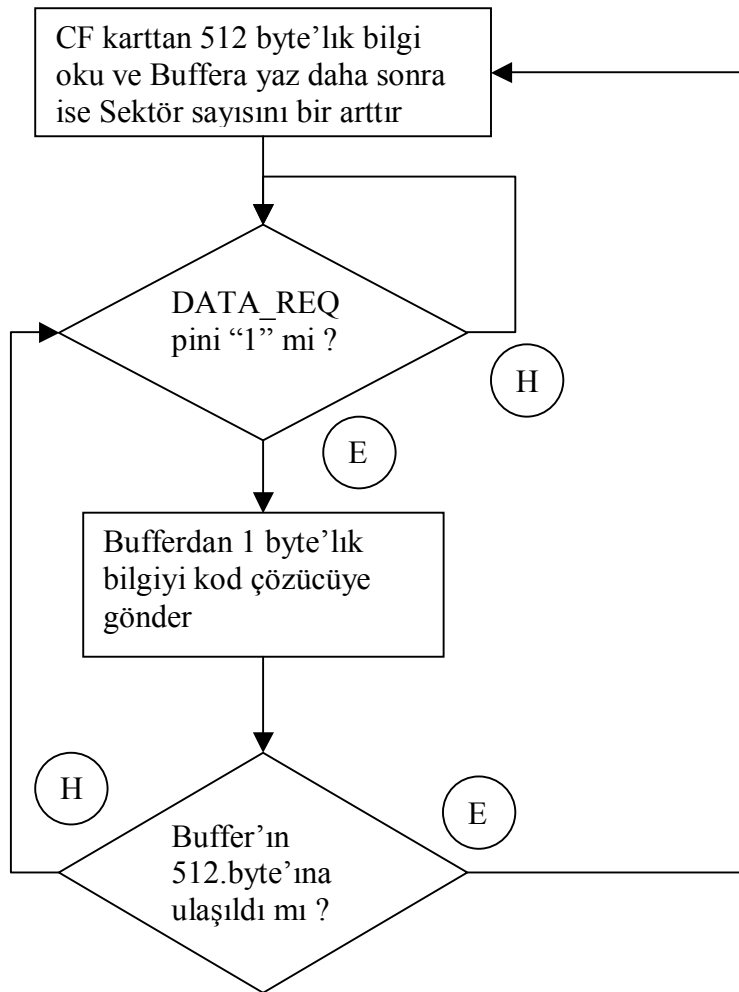
Şekil 8 STA013 kod çözücü entegresinin konfigüre edilmesi.

STA013 entegresinin doğru bir şekilde çalışmaya başlayabilmesi için yaklaşık 2007 tane yazmaca, belirli değerler yüklemelidir. Bunlar ST Microelectronics'in web sitesinde yer almaktadır.

MP3 ÇALDIĞI SIRADA GERÇEKLEŞEN İŞMELER

Bir önceki sayfada belirtilen konfigürasyon işlemi yapıldıktan sonra MP3 kod çözücüsü çalmaya hazır hale gelmiştir. MP3 kod çözücüsü çalmaya hazır hale getirildikten sonra 0x72H (RUN) yazmacına 1 yüklendiği zaman artık MP3 kod çözücüsü MP3 bilgilerini istemeye başlar ve gerekli büyüklükteki bilgiyi alana kadar DATA_REQ pinini lojik "1" de tutar , RUN yazmacına 1 yükledikten sonra 0x13H (PLAY) yazmacına da 1 yüklenirse artık , çözdüğü MP3 bilgilerini DAC'a gönderir ve DAC'tan ses çıkışı alınmaya başlanır.

RUN ve PLAY yazmaçlarına 1 yükledikten sonra, MP3 çalmaya başlayacaktır, MP3 çalarken sürekli olarak yeni MP3 bilgileri Compact Flash karttan okunup Mikrodenetelyicinin kendi iç RAM'inde belirli bir bölüme yazılmaktadır(buffer) daha sonra ise MP3 kod çözücüsüne gönderilmektedir, bunun algoritması aşağıda basit bir şekilde gösterilmektedir.



Şekil 9 MP3 çalma algoritması

ŞARKININ KAÇINCI SANİYEDE OLDUĞUNUN BULUNMASI

MP3 kod çözücüsüne MP3'ün ilk bilgilerini gönderdikten sonra kod çözücü entegre MP3'ün kaç kbps olduğu bilgisini bulup içindeki belirli bir yazmaçta saklamaktadır, bu yazmaç okunarak MP3'ün kaç kbps olduğu öğrenilmektedir.

Örneğin şarkı 128kbps olsun. 128 kilo bit per second = 16 kilo byte per second demektir yani Compact Flashtan her bir 16kb'lık bilgi okunduğunda 1 saniye geçmiş demektir bu sayede şarkının o an kaçınıcı saniyede olduğu çok rahat bir şekilde hesaplanabilmektedir.

SONUÇ

Sonuç olarak MP3 kod çözücüsünün, çözebileceği maksimum bit oranı olan 320Kbps'ye kadar olan MP3'lerin sorunsuz bir şekilde çalınabilmesi sağlanmıştır, yani oldukça performanslı bir sistem yaratılmıştır. Kullanıcı şarkı isimlerini LCD ekranda görüp seçebilmektedir, bunun haricinde dijital olarak ses seviyesini ayarlayabilmektedir. Kullanılan DAC 24-bit olduğu için ses çıkışı oldukça kalitelidir.

MP3 çaların Compact Flash kartı okuyabilmesi için Compact Flash kart FAT16 dosya sistemine göre formatlanmış olmalıdır. MP3 çaların FAT32 ve NTFS sistemlerini de okuması sağlanabilir fakat proje için kısıtlı bir zaman olduğu için ve gerekli parçaların yurtdışından gelmesinden dolayı yaşanan gecikmelerden dolayı MP3 çalar şimdilik sadece FAT16 dosya sistemini okuyabilmektedir.

Projede kullanılan MP3 kod çözücüsü ve DAC, yüzey montajlı olduğu için oldukça küçüktür, eğer 18F458 mikrodenetleyicisinin de yüzey montajlı modeli alınır ve küçük bir LCD alınır MP3 player oldukça portatif bir hale getirilebilir.



Şekil 10 MP3 Çalar çalıştığı sırada LCD ekrandan bir görüntü