

YILDIZ TEKNİK ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

**Musical Instrument Digital Interface
(Midi) Arabirimi**

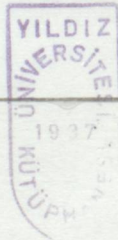
Üzeyir Kızılcırmut

Yüksek Lisans Tezi

YILDIZ TEKNİK ÜNİVERSİTESİ
KÜTÜPHANE DOKÜMANTASYON
DAİRE BAŞKANLIĞI

Kot	R 152 140
Alındığı Yer	FEN BİL. ENS.
Tarih	20.04.1992
Fatura	- - - - -
Fiyatı	25.000.TL.
Ayniyat No	1/2
Kayıt No	48351
UDC	621.3 378.242
Ek	

A



YILDIZ UNIVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ



Türkiye de henüz bu alanda çok az sayıda çalışmaya olanak
bulunmakta ve güncel konuya değinilen kitaplar da azdır. Bu nedenle
başından esirgeyemeyen kıyaslı olarak bu alanda çalışmış olan
SİREL ve Prof. Dr. Sergin AKDAĞ'a teşekkür ederim.
Ayrıca bu tezin hazırlanmasında değerli katkıları bana
yardımlarını esirgeyemeyen değerli dostları ve hocaları KIVANÇ KILIÇ,
YERMAN SEDEF, BORA ENGÜRER'e ve aileleri için teşekkür ederim.

YÜKSEK LİSANS TEZİ

MÜH. ÜZEYİR KIZILARMUT



İSTANBUL - 1990

TESEKKUR

Türkiye’de henüz hakkında pek çalışma yapılmamış olan bu ilginç ve güncel konuyu bana veren ve değerli bilgilerini benden esirgemeyen kıymetli hocalarım sayın Y.Doç.Dr. Osman SİREL ve Prof.Dr. Sezgin ALSAN’a teşekkürlerimi sunarım. Ayrıca bu tezin hazırlanmasında çeşitli konularda bana yardımlarını esirgemeyen değerli dostlarım Mehmet HORSANALI, Herman SEDEF, Bora DNGÜRER’e ve Yıldız Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü mensuplarına buradan teşekkürlerimi sunarım.

MIDI thru	7
2.2. MESAJLARINI YOLLAMA	7
2.3. MIDI CİHAZINI BAĞLAMA	8
2.4. ÇİCİR ÇALIŞMA SİSTEMİ	8
2.5. MIDI DEKODERİ	9
2.6. YILDIZ ÇALIŞMA SİSTEMİ	9
2.7. ÇİCİR ÇALIŞMA SİSTEMİ	10
MIDI MESAJLARI	13
3.1. MIDI CİHAZLARI NASIL BİLEŞİ ALIR VE YOLLAR	13
3.2. MIDI MESAJLARI	16
3.3. MIDI KANALLARI	17
3.4. MIDI KANAL KABİL MESAJI	19
3.5. Mod 1	20
3.6. Mod 2	21
3.7. Mod 3	21
3.8. Mod 4	21

2-5. KANAL MESAJLARI .. İÇİNDEKİLER	22
1- MIDI'Li ENSTRÜMANLAR VE MIDI BAĞLANTILARI	21
1-1. MIDI NEDİR?	22
1-2. MIDI'Li ENSTRÜMANLAR	3
1-3. MIDI BAĞLANTILARI	3
1-4. MIDI KABLOLARI	4
1-5. RADYO FREKANS GİRİŞİMİ	5
1-6. MIDI PORTLARI	6
1-6.1. MIDI in	6
1-6.2. MIDI out	6
1-6.3. MIDI thru	7
1-7. MIDI MESAJLARINI YOLLAMA	7
1-7.1. İki MIDI CİHAZINI BAĞLAMA	7
1-7.2. ZİNCİR ÇALIŞMA SİSTEMİ	8
1-7.3. MIDI GECİKMESİ	9
1-7.4. YILDIZ ÇALIŞMA SİSTEMİ	9
1-7.5. ÇEMBER ÇALIŞMA SİSTEMİ	10
2- MIDI MESAJLARI	15
2-1. MIDI CİHAZLARI NASIL BİLGİ ALIR VE YOLLAR?	15
2-2. MIDI MESAJLARI	16
2-3. MIDI KANALLARI	17
2-4. MIDI KANAL KABUL MODLARI	19
2-4.1. Mod 1	20
2-4.2. Mod 2	21
2-4.3. Mod 3	21
2-4.4. Mod 4	21

III

2-5. KANAL MESAJLARI	22
2-5.1. Kanal Ses Mesajları	23
2-5.2. Kanal Mod Mesajları	29
2-6. SİSTEM MESAJLARI	31
2-6.1. Sistem Gerçek Zaman Mesajları	32
2-6.2. Sistem Ortak Mesajları	36
2-6.3. Sistem Birime Özel Mesajları	37
2-7. KULLANIMDA MIDI	39
2-7.1. Aynı Zamanda Olayları İletmek	39
2-7.2. Çalışma Durumu	39
2-7.3. İşlem Dışı Mesajlar	40
2-7.4. Açığındaki MIDI Mesajları	41
2-7.5. Bir Şarkıyı Çalmaya Hazırlanmak	42
2-8. MIDI MESAJLARININ TEKNİK ÖZELLİKLERİ	43
2-8.1. İkili Aritmetik Düzeni	44
2-8.2. Byte'lar ve Bit'ler	45
2-8.3. Seri İletim	46
2-8.3.1. Baud Hızı	47
2-8.3.2. Başlangıç ve Bitiş Bitleri	47
2-8.4. MIDI Mesajlarının Formatları	49
2-8.4.1. Kanal Ses Mesajları	52
2-8.4.2. Kanal Mod Mesajları	59
2-8.4.3. Sistem Ortak Mesajları	62
2-8.4.4. Sistem Gerçek Zaman Mesajı	64
2-8.4.5. Sistem Birime Özel Mesajları	65
3- BİLGİSAYARLAR VE MIDI	68

3-1. MIDI ARABIRIMI	68
3-1.1. SMART VE DUMB MIDI ARABIRIMLERI	69
3-2. MIDI YAZILIMI	70
3-2.1. Müzik Kayıt ve Performans	71
3-2.2. Müzikal Notasyon ve Basım	78
3-2.3. Sentezleyici Ekleme Editörleri ve Kaynak Kütüphaneleri	80
4- IBM PC İÇİN MIDI ARABIRIMI	83
4-1. IBM PC DONANIMINA GENEL BİR BAKIS	83
4-1.1. 8088 DONANIM KESİNTİ İSTEĞİ	84
4-1.3. I/O KANALI VE I/O HARİTASI	85
4-1.4. I/O KANALININ TANIMLANMASI	88
4-2. PC MIDI ARABIRIMI DONANIMI	90
4-3. MIDI ARABIRIMI İÇİN GEREKLİ YAZILIM YAPISI	96
EK1- KULLANILAN TUMLEŞİK DEVRELERİN TEKNİK ÖZELLİKLERİ. 99	
UM 8250 A	100
7406	116
74LS30	117
74LS125A	118
74LS240	119
74LS245	122

SÖZETRY

Mikroişlemcili elektronik müzik enstrümanlarının etkili olarak kullanılabilmesi için bir bilgisayar veya bir ana klavye tarafından kumanda edilmeleri gereklidir. Bu sayede etkili bir elektronik müzik sistemi kurulabilir. Bu sistemin kurulması, yani bilgisayar veya ana klavye ve diğer elektronik müzik enstrümanlarının birbirleri ile bağlantıları ve muhtemel haberleşmelerin diğer MIDI (assisi) standardı) by ile gerçekleştirilebilmektedir.

Böyle bir müzik sistemi ile bir müzik stüdyosu gibi kayıt yapılabilir, bilgisayar ekranından yazılan notasyonlar çalınabilir, parçalar sayısal bilgi olarak disk gibi bir ortamda saklanabilir. Yine bu kayıtlar üzerinde, normal stüdyo kayıtlarından farklı olarak, yeniden kayıt yapmaya gerek duymadan düzeltmeler, eklemeler, çıkartmalar ve tekrarlar yapılabilir. Ayrıca bilgisayar kumandasının avantajı ile aynı anda birçok müzik enstrümanı insansız olarak veya tek kişi tarafından çalınabilir.

İyi sonuçları elde edebilmek için MIDI standardının gerektirdiklerini iyi bilmek ve etkili kullanmak gereklidir. Birçok bilgisayar direk veya aracı birim bağlantısı ve bir program ile MIDI sisteminde rahatlıkla kullanılabilir. MIDI standardında haberleşme seri olarak yapıldığından, kullanılan bilgisayarın seri giriş çıkış yeteneklerinin iyi olması büyük avantajlar sağlayacaktır. *advantages*

SUMMARY

The electronic musical instruments with microprocessor must be commanded by a computer or a main keyboard for an effective work. By this way it is possible to set up an effective electronic music system. To set up this system which means to connect the computer or main keyboard with the other electronic musical instruments and to provide the communication with each other is possible (through) by the MIDI standard.

It is possible to release the record as a music studio, to play the notation which are on the screen, songs can be kept as a numerical information within a disc by this system. As a difference from the normal studio records on this records it can be made corrections, additions, wiping out some pieces repetitions without any necessity for the rerecord. In addition, by the advantage of the command of computer many of musical instruments, at the same time, might be played without any person or by one person.

For the useful results and for an effective method of using the needs of MIDI standard must be known (very well) in detail many of the computers can be easily used as directly or as a connection of interface of a program in the MIDI system. Because of the communication is serial in the MIDI system the quality of in-out abilities of the computer which will be used will provide many advantages.

1-1. MIDI NEDİR ?

Elektronik teknolojinin gelişmesiyle birlikte mikro işlemci ve mikro kontrolörler üretilmiş ve çok çeşitli alanlarda kullanılmaya başlanmıştır. Bu sayede bir çok konuda klasik yapılar değişmiş, yerini elektronin getirdiği esnek ve daha gelişmiş yapılar almıştır. Bu alanlardan biride müzik ve dolayısı ile müzik enstrümanlarıdır.

Mikro işlemci ve mikro kontrolörler kullanılarak tamamen elektronik olarak bir çok müzik enstrümanı yapılmıştır. Bu enstrümanlar mikro işlemcili yapılarının getirdiği programlanabilme yeteneğinden ötürü klasik enstrümanlara göre daha kullanışlı ve esnektir. Yine bu enstrümanlar yapıları BİRİNCİ BÖLÜM bir bilgisayar veya birbirleri MIDI'LI ENSTRÜMANLAR VE MIDI BAĞLANTILARI ile bu tür cihazların birbirleri ile haberleşip, anlayabilecekleri bir program yapıları ve bağlantı şekli hazırlanmıştır. Bir çok üretici firma tarafından kabul edilerek kullanılan bu sistem bir standart haline gelmiştir. Bu standarda MIDI (Musical Instrument Digital Interface = Müzik Enstrümanı Sayısal Arabirimi) adı verilmiştir.

Bu çalışmada da bu konu incelenmiş ve IBM PC için bir MIDI arabirimi yapılmıştır.

1-1. MIDI NEDİR ? ENSTRÜMANLAR

Elektronik teknolojisinin gelişmesiyle birlikte mikro işlemci ve mikro kontrolörler üretilmiş ve çok çeşitli alanlarda kullanılmaya başlanmıştır. Bu sayede bir çok konuda klasik yapılar değişmiş, yerini elektronigin getirdigi esnek ve daha gelişik yapılar almıştır. Bu alanlardan biride; müzik ve dolayısı ile müzik enstrümanlarıdır.

Mikro işlemci ve mikro kontrolörler kullanılarak tamamen elektronik olarak bir çok müzik enstrümanı yapılmıştır. Bu enstrümanlar mikro işlemcili yapılarının getirdigi programlanabilme yeteneginden ötürü klasik enstrümanlara göre daha kullanışlı ve esnektir. Yine bu enstrümanlar yapılarından ötürü bir bilgisayar veya birbirleri tarafından yönetilebilirler. Bu amaçla bu tür cihazların birbirleri ile haberleşip, anlaşabilecekleri bir program yapısı ve bağlantı şekli hazırlanmıştır. Bir çok üretici firma tarafından kabul edilerek kullanılan bu sistem bir standart haline gelmiştir. Bu standarda MIDI (Musical Instrument Digital Interface = Müzik Enstrümanı Sayısal Arabirimi) adı verilir.

Bu çalışmada da bu konu incelenmiş ve IBM PC için bir MIDI arabirimi yapılmıştır. Bağlantısı MIDI kullanıcısının en basit kısmıdır. Standart ara kablolar ve prizleri bir MIDI cihazını bir diğerine bağlamayı çok kolay kılar. Buna rağmen bir MIDI bağlantısı için iki kabloyu birbirine takmaktan daha fazla yapacak şey vardır. MIDI mesajlarının portlardan ve

1-2. MIDI'LI ENSTRÜMANLAR

Günümüzde MIDI arabirim bağlantısı modern elektronik profesyonel müzik enstrümanlarının tümünde kullanılmaktadır. Amatör cihazların ise bazılarında MIDI bağlantısı mevcuttur. Genelde MIDI bağlantısı bulunan bazı elektronik enstrümanlar; elektronik piano (electric piano), elektronik bateri (drum machine), sentezleyici (synthsizer), elektronik org (electric organ) gibi müzik enstrümanlarıdır. Ayrıca APPLE MACHINTOSH II, ATARI 520ST, ATARI 1040ST, AMIGA 1000, AMIGA 2000, AMIGA 500 (uyarlayıcı arabirim ile) gibi bilgisayarlarda da MIDI arabirim bağlantısı mevcuttur.

MIDI bağlantısı bulunan müzik enstrümanları kullanılarak adeta tek kişilik bir orkestra oluşturulabilir. Zira bir MIDI sisteminde tüm enstrümanlar bir ana klavye ile tek kişi tarafından veya bir bilgisayar ile bir program tarafından yönetilebilir. Ayrıca çalınan veya çalınacak müzik parçaları disk gibi bir birikişim ünitesinde sayısal bilgi olarak korunabilir. MIDI sistemlerinin daha da geniş olmasını sağlayan çeşitli MIDI adaptörleri mevcuttur. Bu cihazlardan yeri geldikçe söz edilecektir.

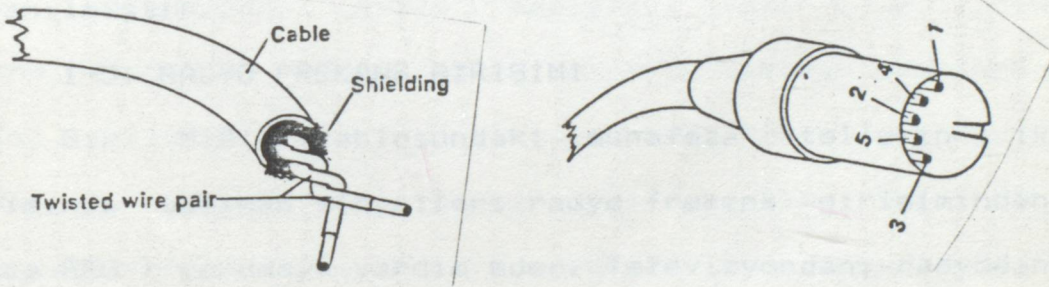
1-3. MIDI BAĞLANTILARI

MIDI cihazlarının bağlantısı MIDI kullanımının en basit kısmıdır. Standart ara kablolar ve prizler, bir MIDI cihazını bir diğerine bağlamayı çok kolay kılar. Buna rağmen bir MIDI bağlantısı için iki kabloyu birbirine takmaktan daha fazla yapacak şey vardır. MIDI mesajlarının portlardan ve

kablolardan nasıl geçtiğini ve mesajların istediğimiz şekilde akması için, hangi portu nasıl bağlamamız gerektiğini bilmeliyiz. Bu bölüm MIDI bağlantılarının nasıl yapılacağını anlatacaktır. İlk olarak MIDI kabloları anlatılacak ve bunların nasıl yerleştirilecekleri gösterilecektir. İkinci kısımda farklı MIDI portlarının tanımlanması ve son bölümde de bir MIDI sisteminde kullanılabilecek bağlantı cihazlarının farklı konfigürasyonlarını ve her bir konfigürasyondaki cihazların arasında mesajların işleme yönü gösterilecektir.

1-4. MIDI KABLOLARI

Bir MIDI kablosu; muhafazalanmış, kıvrılmış ve her ucunda beş ayaklı erkek DIN (Alman standartı) fişi olan ikili bir kablodur. (Şekil 1)



Şekil 1. MIDI Bağlantı kablosu ve fişi

Muhafazalı, kıvrılmış ikili kablo, iki telin biribiri çevresinde bükülmesiyle elde edilen bir kablodur. Muhafazadan anlatılmak istenen, yumuşak bir tel ağın tüm kablo boyunca gitmesi ve kıvrılmış ikili teli çevrelemesidir. Diğer bir deyişle, iki canlı blendajlı kablodur. MIDI'nin mesajları yollamakta kullandığı sinyalleri taşıyan bu yapıdaki kablodur.

de gösterildiği gibidir.

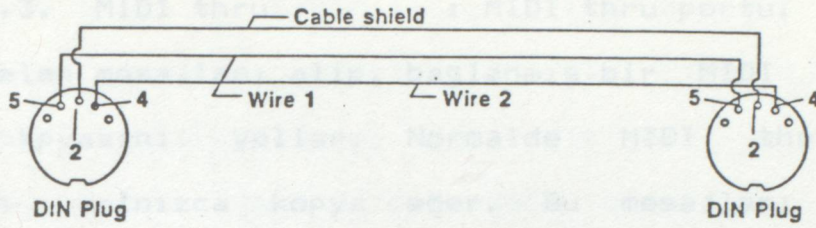
Beş ayaklı DIN fişi Avrupa'da sürekli kullanılan ses donanımlarının standart fişidir. A5-PIN DIN Fiş adı verilir. İsmi anlamı beş küçük ayak içermesinden gelir. Her biri numaralanıp, bir kablodaki tellere bağlanabilir. Şekil 1'de bu ayakların numaraları görülmektedir.

Şekil 2. DIN Fişlerin kablo ile bağlantısı.

Bir MIDI kablosu, DIN fişinin ayaklarının yalnızca 1-6. MIDI PORTLARI uçunu kullanır. 4 ve 5 nolu ayaklar MIDI sinyalini kablodaki MIDI portları bir MIDI cihazının giriş ve çıkış tellerin ikili kıvrımına (yani canlı uçlarına) taşır. 2 nolu ayak ise kablodaki muhafazaya topraklanmıştır. 1 ve 3 nolu ayaklar MIDI için kullanılmaz, ama olabilecek değişimlere karşıfazladan sinyalleri taşıyabilecek şekilde bırakılmıştır. Örneğin ATARI 520ST'de, bu ayaklar kullanılmıştır. Kendi gelen mesajları alır ve cihazın

1-5. RAYDO FREKANS GİRİŞİMİ portu kabloya herhangi bir mesaj Bir MIDI kablodaki muhafaza tellerin ikilinin kıvrımında taşınan sinyalleri radyo frekans girişiminden (kısaca RFG) korumaya yardım eder. Televizyondan, radyodan ya da diğer yayınlardan ve cihazlardan gelen sinyalleri engeller. Uzun bir kabloyu bir cihaza bağladığınızda kablo bir anten gibi davranıp, RFG'yi alır. MIDI kabloları da bu konuda bir istisna değildir. RFG, kablo üzerindeki mesajları değiştirerek oya da pistenmeyen mesajlar ekleyerek, MIDI sinyallerine engel olabilir. Blendaj muhafaza RFG'den gelen bozuklukları önlerdir. Ender durumlarda ise MIDI out portu olmayabilir. Böyle bir kabloyu, kullanıcının kendisinin bile yapması mümkündür. İki DIN fiş arasındaki bağlantı şekil 2.

de gösterildiği gibidir.



Şekil 2. DIN Fişlerin kablo ile bağlantısı.

1-6. MIDI PORTLARI

MIDI portları bir MIDI cihazının giriş ve çıkış bilgilerinin geçtiği ve beş ayaklı DIN soketinin girdiği yerdir. Üç çeşit MIDI portu vardır. MIDI in (MIDI girişi), MIDI out (MIDI çıkışı), MIDI thru (yönlendirilmiş MIDI).

1-6.1. MIDI in : MIDI in portu, bağlandığı bir MIDI kablosundan gelen mesajları alır ve cihazın mikroişlemcisine iletir. MIDI in portu kabloya herhangi bir mesaj yollayamaz. Bunun anlamı yalnızca MIDI in portu olan cihazlar diğerlerine mesaj yollayamazlar. Çoğu cihazda yalnızca tek bir MIDI in portu vardır. Bazı cihazlarda ise (bunlar geriye bilgi almaksızın diğer cihazları kontrol eder) MIDI in portu hiç yoktur.

1-6.2. MIDI out : MIDI out portu, cihazın mikro işlemcisinden çıkan mesajları bağlı olduğu MIDI kablosuna yollar. Bu port kablodan hiçbir mesaj alamaz. Çoğu cihazda yalnızca MIDI out portu vardır. Bazılarında ise birden fazla vardır. Ender durumlarda ise MIDI out portu olmayabilir. Bunun amacı cihazın alabileceği mesajları kısıtlamaktır.

yanar. 1-6.3. MIDI thru kablo ile: MIDI thru portu, MIDI'din portuna gelen mesajları alıp, bağlanmış bir MIDI kablo suna bunların kopyasını yollar. Normalde MIDI thru, bir MIDI mesajlarını, yalnızca kopya eder. Bu mesajları cihazın mikroişlemcisine yollamaz, veya kablolar üzerindeki mesajlara karışacak şekilde mikro işlemciden hiçbir mesaj almaz. MIDI thru, zincir çalışma MIDI sistemlerinin oluşturulmasında kullanılır. (emiler iki bölümde anlatılacaktır.) MIDI thru portu asli olarak bir alıcı-verici port değildir. Çogu cihazda bu denli çok yönlü olmalarına karşın bulunmazlar. MIDI thru

portuna 1-7. MIDI MESAJLARINI YOLLAMA sistemi: Birbirine bağlaya Biraz önce Zı anladiğimiz gibi MIDI cihazlarına takabileceğimiz MIDI portları ve kabloları bilgi alışverişi içindir. Bunların bağladığınızda iki önemli noktayı unutmamalısınız: İlki; MIDI mesajları MIDI kabloları üzerinde yalnızca bir yönde giderler, ve ikincisi; tek bir MIDI portu ya mesaj yollar ya da alır. İkisini birden yapamaz. MIDI cihazlarının kendi aralarında bilgi alışverişini serbestçe yapabilmeleri için kabloları doğru portlara bağlamalısınız. Bir sonraki bölümde bazı standart konfigürasyonların çizimlerini göreceksiniz. olarak tanımlıyor. 1) Bunun nedeni

1-7.1. İKİ MIDI CİHAZINI BAĞLAMA : En basit MIDI bağlantısı yalnızca bir MIDI cihazından diğerine mesaj taşır. Bir kablo bir cihazın MIDI out portundan diğer cihazın MIDI in portuna takılır. Bu bağlantı ilk cihazın MIDI out portundan ikinci cihazın MIDI in portuna mesaj yollamaya

yarar. Böyle basit bir kablo ile yalnızca birinci cihazdan ikincisine bilgi akışı yapılabilir. İlk cihazın MIDI in portuna bir kablo ekleyip, diğerinin MIDI out portuna takılılırsa, her iki cihazda birbirine bilgiyi geri yollayabilecek duruma gelmiş olur. Bu konfigürasyonda cihazlar birbirlerine bilgi yollayıp, alabilirler.

1-7.2. ZİNCİR ÇALIŞMA SİSTEMİ (DAISY CHAINING) :

Çogu MIDI sisteminin ikiden fazla cihaz kullanmasının nedeni; bir mesajın, bir cihazdan geri kalan diğerlerine rahatça gidebilmesi içindir. Eğer her bir cihaz, bir MIDI thru portuna sahipse bunları zincir çalışma sisteminde birbirine bağlayabilirsiniz. Zincir çalışmada ilk MIDI cihazı MIDI out portundan, ikinci cihazın MIDI in portuna mesaj yollar. İkinci cihaz gelen mesajın tam bir kopyasını thru portundan üçüncü cihazın MIDI in portuna yollar. Bu sistem zincirleme olarak pekçok cihazın thru portlarını, diğerlerinin MIDI in portlarına bağlayarak uzatılabilir. Diğer cihazlar araya mesaj sokamazlar. (Şekil 3) Bu, zincir çalışma sistemini bir ana cihazdan diğer cihazlara seri olarak bağlanması durumunda çok etkili kılar. (Sentezleyici kullanıcıları genellikle bu alt cihazları esir cihaz olarak tanımlarlar.) Bunun nedeni ana cihazın bilgi alamaması ve diğer cihazların bilgi yalayamamasıdır. (Ya da gerek duymamaları!) Esir cihazların kendi bilgilerini yollayamamaları önemli değildir. Ancak elbette ki ana cihazın MIDI in portuna diğer cihazlardan birisinin MIDI out portunu bağlayıp, ondan mesaj alması

mümkündür. Çünkü bazı cihazların thru portu yoktur. Bunlardan 1-7.3.1 MIDI GEÇİKMESİ (MIDI LAG) ay: Bu teoride tutabildiğiniz kadar uzun bir zincir çalışma sistemini her cihazda thru port olduğu sürece mümkün olduğunu gösterir. Pratikte ise bu mümkün değildir. Bazı MIDI thru portları MIDI mesajını geçirirken bu ciddi bir şekilde geç kalırlar. Bu gecikmelerin çoğu, bir MIDI cihazı mikro işlemciden gelen mesajı direkt MIDI out portundan MIDI thru portuna yollayamadığı için olur. Mikro işlemci tüm mesajı okuduktan sonra onu MIDI thru portuna yollar. Bazen kendi mesajlarını bile katar. Bu şekilde mesaj iletimi çok yavaşlar. MIDI mesajı üç ya da dört thru portundan küçük gecikmelerle geçtikten sonra mesaj temiz bir ses gecikmesi yaratacak kadar yavaşlar. Bu gecikmeye MIDI Gecikmesi adı verilir. **WORK**

Herhangi bir gecikmesi ya müzik performansında bir zamanlama sorunlarına yol açar. Eğer bir ana sentezleyicinin başında klavyeden notaları çalışıyorsanız, MIDI zincirinin uzak bir noktasında bir MIDI gecikmesi yüzünden notalar ana sentezleyicinin içaldığından sonra gelir. Her bir notanın başlangıcındaki atakları duyarsınız. Buna karşın günümüzde pek çok MIDI cihazının thru çıkışı bu gecikmeyi oluşturmayacak şekilde yapılmıştır. Sonuç olarak MIDI gecikmesi bir sorun oluşturmaz ve istenildiği kadar uzun zincir çalışma sistemleri yapılabilir. **Özetle, ayrıca yirmi**

1-7.4. **YILDIZ ÇALIŞMA SİSTEMİ**: Her zaman bir MIDI sistemi oluşturmak için yıldız çalışma sistemini

kullanamazsınız. Çünkü bazı cihazların thru portu yoktur. Bunlarda önemli bir MIDI gecikmesi sorunu ortaya çıkartırlar. Esir cihazları ana cihaza bağlamanın bir diğer yöntemi YILDIZ ÇALIŞMA SİSTEMİ (STAR NETWORK) oluşturmaktadır. (Şekil 4)

Yıldız çalışma sistemi MIDI thru portunu kullanmak yerine, sisteme MIDI thru kutusu (MIDI thru box) olarak adlandırılan bir cihaz eklenmesiyle çalışır. Bazı cihazlar (Başta bilgisayar) Bir MIDI thru kutusuna en az bir MIDI in portuna ve pek çok thru portuna sahiptir. MIDI in portundan giren mesajlar kopya edilerek thru kutusundaki her bir MIDI thru portuna tek tek iletilir. Ana cihazın MIDI out portunu, thru kutusunun in portuna ve thru kutusunun thru portlarını esir cihazlara bağlayarak tüm mesajlar tek tek iletilebilir. Her tarafından aynı

1-7.5. ÇEMBER ÇALIŞMA SİSTEMİ (RING NETWORK) :

Herhangi bir zincir veya yıldız çalışması belli bir limite sahiptir. MIDI mesajları çoğunlukla bir yönde, ana cihazdan esir cihazlara gider. Yıldız ya da zincir sistemlerinde her cihazın diğerleri ile haberleşme imkanı yoktur. Örnek olarak şekil 3'deki zincir çalışma sisteminde görüldüğü gibi esir cihazlar yalnızca ana cihazdan mesaj alır, ama yollayamaz. MIDI out portundan başka bir cihaza bağlasak bile, yalnızca bağladığımız cihaza mesaj yollayabilir. Beş cihazı bağlamak için ve her bir cihazın mesaj alıp yollayabilmesi için hepsinde MIDI out ve MIDI in portları olmalı, ayrıca yirmi tane kablo kullanılmalıdır. Bu da pahalı ve sağlıksız bir çözümdür. Şekil 5'de gösterilen bir çember çalışması, kablo

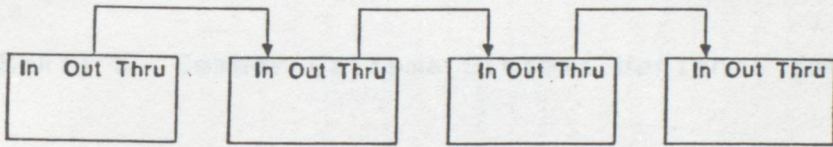
rahatça programlanmaya elverişli mikroişlemci olmayabilir. ve port gerektirmeden her cihazın hem mesaj alabilmesini, hem de yollayabilmesini sağlar. Çemberdeki her bir cihazın MIDI out portu, bir sonrakinin MIDI in portuna bağlıdır.

Çember çalışması çoğu MIDI cihazında kullanılmaz. Çünkü MIDI out portu kendi mikroişlemcisine mesaj yollar, MIDI in portundan aldığı mesajı geçirmez ve bunu da doğrudan mikroişlemcisine yollar. Ama bazı cihazlar (başta bilgisayarlar!) MIDI in portundan aldığı mesajı programla kullanabilecek kadar güçlü ve hızlıdırlar. Gelen mesajı kendi mesajları ile üst üste bindirip, bu karışımı MIDI out portuna yollayabilirler. Eğer çember çalışmasındaki her bir cihaz kendi MIDI out portunda bu tür geçiş sistemine sahipse, bir cihazdan yollanan her mesaj çemberdeki cihazlar tarafından aynen alınacaktır.

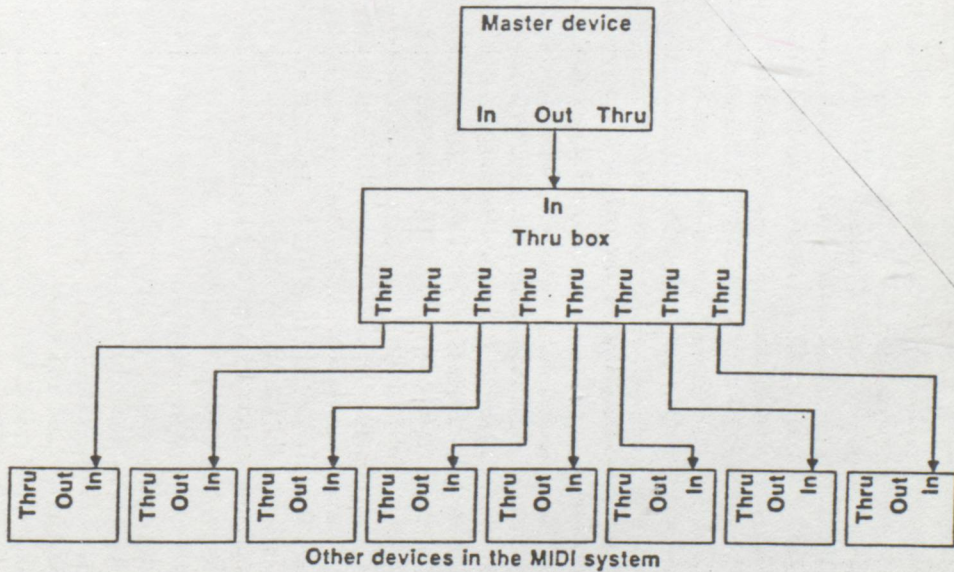
Çember çalışmasında üst üste gelme problemleri vardır. Örneğin mesaj gönderen bir cihaz, kendi mesajını, çember tamamlandığında kendi MIDI in portundan alabilir. Orjinal cihaz kendi mesajını tanıyacak kadar yenekli değilse, mesajı tekrar sonraki alete geçirecekve mesaj hiç durmaksızın sonsuz bir döngü içine girmiş olacaktır. Bu döngü Mesaj Geribeslemesi olayına neden olur ve bazı işe yarayan mesajların da alışverişini önler. Sistemdeki her cihaz kendi mesajını tanıyıp, geri göndermeyecek şekilde tanımlanmalıdır. Bu istemde MIDI gecikmesi de sorunlar yaratabilir.

Bu yazılanlardan sonra çember çalışmasının her cihazda kullanışlı olmadığı ortaya çıkar. Çünkü her cihazda

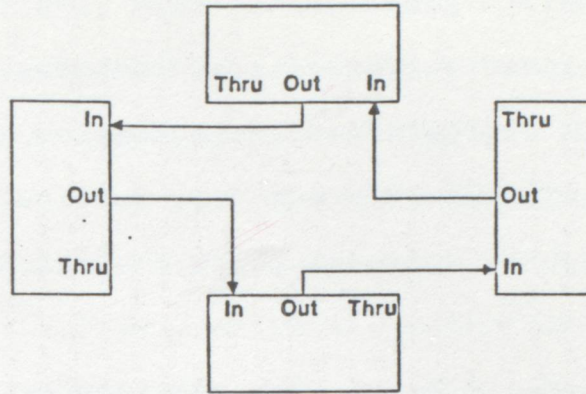
rahatça programlanmaya elverişli mikroişlemci olmayabilir. Uygun bir yazılımla pek çok kişisel bilgisayar çember çalışmasında kullanılabilir. Ancak pek az insan yalnızca bilgisayarlardan oluşan bir MIDI sistemini kullanmayı yeğler. Yeni üretilen sentezleyicilerde daha güçlü mikroişlemciler ve daha üstün programlama yeteneği bulunmaktadır. Teknolojik gelişmelerin ışığında belki bir gün çember çalışmasını taşınabilir bir MIDI sisteminde bile kullanmak mümkün olacaktır.



Şekil 3. Zincir Çalışma Sistemi bağlantı şekli.



Şekil 4. Yıldız Çalışma Sistemi bağlantı şekli



Sekil 5. Çember Çalışma Sistemi bağlantı şekli.

2-MIDI MESAJLARI

MIDI mesajları, MIDI'nin en önemli kısmıdır. Bunlar MIDI cihazları arasındaki haberleşmeye ve hangi müziğin bir cihazdan diğerine yollanacağını belirlerler. MIDI'yi tam ve doğru olarak anlayabilmek için tüm MIDI mesajlarını bilmek ve MIDI cihazlarının bunlara nasıl karşılık verdiğini öğrenmek gerekir.

Bu bölüm size MIDI'nin özü olarak nitelendirilebilecek MIDI cihazının kablolar üzerinden nasıl mesaj yollayacağını ve bu mesajları yollamak için hangi düzenlemeleri kullandığını öğretecektir.

İKİNCİ BÖLÜM

2-1. MIDI CİHAZLARI NASIL BİLGİ ALIR VE YOLLAR ?

MIDI cihazları MIDI MESAJLARI vermek için telgraf sistemine benzer bir yol kullanırlar. Hattın ucundaki telgrafçı, Morse alfabesini kullanarak, bir takım nokta ve çizgilerle her bir karakteri diğer tarafa yollar. Diğer uçtaki ise bu nokta ve çizgileri belli gruplara bölerek alır. Bunların tersine çevrilmesiyle tercüme gerçekleşir. Bir telgrafçı gibi, MIDI cihazındaki mikroislemci bir MIDI kablosuna kısa bir sinyal serisi yollar. Nokta ve çubukların yerine 1'ler ve 0'lar vardır. Bu 1 ve 0'lara Bit denir ve MIDI haberleşmesinde sekizlik gruplar halinde kullanılır. Her bir sekizlik grup bir Byte olarak adlandırılır. Ancak haberleşme sırasında karışmaları önlemek için bu sekizlik grubun başına başlangıç biti (Start Bit) adı verilen 1 ve sonuna bitiş biti (Stop Bit) adı verilen 0 eklenir. Böylece

sekiz 2-MIDI MESAJLARI bitlik paket halinde gönderilir. Bu yöntem MIDI mesajları MIDI'nin en önemli kısmıdır. Bunlar MIDI cihazları arasındaki haberleşmeye ve hangi müziğin bir cihazdan diğerine yollanacağını belirlerler. MIDI'yi tam ve doğru olarak anlayabilmek için tüm MIDI mesajlarını bilmek ve MIDI cihazlarının bunlara nasıl karşılık verdiğini öğrenmek gerekir. MIDI'de saniyede gönderilen bit sayısı 31250. Bu bölüm size MIDI'nin özü olarak nitelendirilebilecek MIDI cihazının kablolar üzerinden nasıl mesaj yollayacağını ve bu mesajları yollamak için hangi düzenlemeleri kullandığını öğretecektir. Saniyede 3125 byte bilgi gönderilir.

2-1. MIDI CİHAZLARI NASIL BİLGİ ALIR VE YOLLAR ?

MIDI cihazları bilgiyi alıp vermek için telgraf sistemine benzer bir yol kullanırlar. Hattın ucundaki telgrafçı Mors yalfabesini kullanarak, bir takım nokta ve çizgilerle her bir karakteri diğer tarafa yollar. Diğer uçtaki ise bu nokta ve çizgileri belli gruplara bölerek alır. Bunların tersine çevrilmesiyle tercüme gerçekleşir. Bir telgrafçı gibi, MIDI cihazındaki mikroişlemcin bir MIDI kablosuna kısa bir sinyal serisi yollar. Nokta ve çizgülerin yerine 1'ler ve 0'lar vardır. Bu 1 ve 0'lara Bit denir ve MIDI haberleşmesinde sekizlik gruplar halinde kullanılır. Her bir sekizlik grup bir Byte olarak adlandırılır. Ancak haberleşme sırasında karışmaları önlemek için bu sekizlik grubun başına başlangıç biti (Start Bit) adı verilen 1 ve sonuna bitiş biti (Stop Bit) adı verilen 0 eklenir. Böylece

yüksekliğini söyler. MIDI cihazının onu yeterince yüksek sekiz bitlik data, on bitlik paket halinde gönderilir. Bu yöntem bilgisayarlarda seri haberleşme standardı RS-232 olarak benzer biçimde kullanılmaktadır. (MIDI aslında RS-232'nin geliştirilmiş ve modifiye edilmiş bir şeklidir.) ve Data Byte adı verilen her sekizli grubun 256 olasılığı olduğu düşünülürse her bir sayıyı 0'dan 255'e kadar sunma olanakları vardır. MIDI'de saniyede gönderilen bit sayısı 31250'dir. Yani diğer bir deyişle MIDI kablosundan bir saniyede 31250 bit gönderilebilir. Bir datanın sekiz data biti, bir başlangıç ve bir bitiş biti olmak üzere on bitten oluştuğunu düşünürsek, saniyede 3125 byte bilgi gönderilir.

2-2. MIDI MESAJLARI

Eğer bir MIDI mesajlarında on bitten oluşan (aslında sekiz bit olan) her bir grup bir karaktere karşılıktır. Genellikle bir MIDI mesajını yollamak için iki yada üç byte'a ihtiyaç vardır.

2-3. MIDI KANALLARI

Herhangi bir MIDI mesajındaki ilk byte'a STATUS BYTE denir. Bu, mesajın cinsini söyler. Status Byte bir Note On mesajı olarak tanımlanır. (Örneğin; Notanın o anda başladığını gösteren bir mesaj.) Bir Pitch Bend Change (Gam Değişimi - bir sentezleyicideki ses yüksekliği değişimi kontrolörü.), ya da diğer muhtemel tiplerdeki mesajlar olarak okunur.

Herhangi bir MIDI cihazına ilk byte'a DATA BYTE'1 denir. Her Data Byte'1 Status Byte tarafından verilen bilgiyi ayrıntılı bir şekilde hazırlar. Örneğin; ilk Data Byte'1 notanın ses

yüksekliğini söyler. MIDI cihazının onu yeterince yüksek sesle çalabilmesi için gerekli atak değerini ise ikinci Data Byte'ı gösterir. Belli tanımlamaları taşımak için iki çeşit Data Byte'ı kullanılır. Bazı sistemler tek Data Byte'ı ile yetinirken, bazıları ise hiç istemez. Şekil 6 tipik Status ve Data Byte'larıyla mesajların dağılımını gösterir. Çalıştığı analom Data ve Status Byte'larının birbirinden ayrılmasında, MIDI 0'dan 127'ye Status Byte, 128'den 255'e ise Data Byte olarak kabul eder. Bir MIDI cihazı, MIDI kablosundan bir mesaj aldığında, bunun ne tip bir mesaj olduğunu, nasıl kullanacağını ve bunu neyin izleyeceğini bilir. Örneğin; bir cihaz bir Note On'da Status Byte aldı ise, sonraki iki Data Byte'ını ses yüksekliği ve şiddet verileri olarak yorumlar. Eğer bir cihaz ses yüksekliği değişimi Status Byte'ı alırsa, sonraki iki Data Byte'ını ses yüksekliği değişimi verisi olarak yorumlar. (İki dinleyen, doğru kişilerdir.)

2-3. MIDI KANALLARI

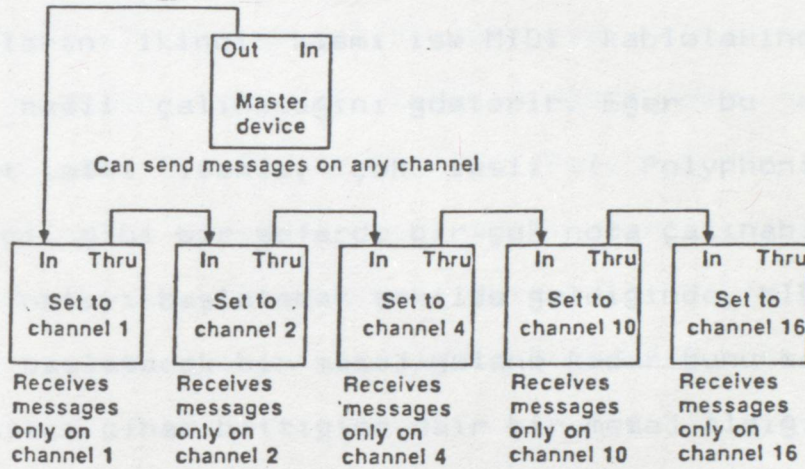
Önce Mors kod sistemini düşünün. Muhtemelen aynı hatta pek çok telgrafçı vardır. Biri bir telgraf yolladığında, geri kalan tümü bu mesajı alır. Bu pratik değildir. Örneğin; B, C ve D telgrafçıları görmeden sırf A'ya bir mesaj yollamak istiyoruz. Anlayacağınız gibi bu olanaksızdır.

Aynı durum MIDI sistemleri için de geçerlidir. Bir zincir çalışmasında bir ana cihazın pek çok alt cihaza bağlı olduğunu varsayın. Ana cihaz bir mesaj yolladığında, diğer tüm cihazlarda bu mesajı alır. Ana cihaz, diğer cihazları çalıştırmadan bir tanesine ulaşamaz. Bu durum; diğer tüm

Bu teknik sıkıcı görünse de, çok yüksek bir hızla cihazları da anal cihazdan çalmak istiyorsanız çok uygundur. Ancak, ana cihazdan her birisine ayrı bir bilgi yollayacaksanız işe yaramaz. 16 kanallı herhangi birisinden yolla MIDI, sistemdeki farklı cihazlarla haberleşebilmek için MIDI kanallarını kullanır. Bunların nasıl çalıştığını anlamak telgraf hatlarındaki ile aynıdır. Telgraf hattındaki kullanıcı bazı kurallar koyar. İki çeşit telgraf olacaktır. Genel telgraflar herkesin dinleyebileceği, özel telgraflar ise kime gönderildiyse, onun dinleyebileceği mesajlar olacaktır. Telgraf şirketi her telgrafçıya bir numara verir ve özel telgrafları yalnızca kendi numarası ile başlıyorsa okuyabilir. Özel bir telgraf numarası başka olan birine ulaştığında bu kişi hiç aldırmadan mesajın gerisini dinlemez. (Elbetteki tüm bu hipotetik telgrafçılar tümüyle vicdanlarının sesini dinleyen, dürüst kişilerdir.)

MIDI'de aynı tekniği kullanır. Tüm mesajları iki farklı tipe ayırır: Sistem mesajları; sistemdeki tüm cihazlara gider. Ve Kanal mesajları; yalnızca özellikle tanımlanmış cihazlara gider. Her kanal mesajı Status Byte'da yollanan 1'den 16'ya kadar bir kanal numarası içerir. Bir MIDI cihazı kanal mesajı içeren bir Status Byte aldığında kanal numarasına bakar. Eğer kanal numarası bu mesajı almasına izin veriyorsa, mesajdaki gibi davranır. Eğer değil ise hiç önemsemez. Sistem mesajlarında, Status Byte'da kanal numarası olmamasının nedeni tüm cihazların bunu alması gerektiğindendir.

Bu teknik sıkıcı görünse de, çok yüksek bir hızla uygulanır. MIDI kanallarının davranışının pratik görünümü telgraf hattından çok TV kanalları gibidir. (Şekil 6) Yollayıcı cihaz mesajları 16 kanalın herhangi birisinden yollayabilir. Yalnızca alıcı cihazlar kanal mesajına göre buna uyar ya da uymazlar. Cihazlar kanal mesajlarını önemseyip önemsemeyeceklerini kendileri ayarlayamazlar.



Şekil 6. MIDI cihazları ayarlandıkları kanalı alırlar.

2-4. MIDI KANAL KABUL MODLARI

Bir MIDI cihazını bir ya da daha fazla kanala aarlamak için, önce kontrol paneli kullanılarak MIDI kabul modlarından biri set edilir. Seçeceğiniz kabul modu MIDI cihazının farklı kanalları nasıl dinleyeceğini belirler.

Kullanılan dört mod şunlardır:

- * Mod 1: Omni On/Poly
- * Mod 2: Omni On/Mono
- * Mod 3: Omni Off/Poly
- * Mod 4: Omni Off/Mono

MIDI Modların ilk yarısı gelen MIDI kanallarındaki MIDI cihazın görünümünü anlatır. Eğer Omni açık ise tüm MIDI kanalları görülür ve tüm kanallara karşılık verilebilir. Hangi kanalın ne yolladığı hiç önemli değildir. Ancak eğer Omni kapalı ise MIDI cihazı yalnızca almaya set edildiği belli kanal yada kanallardan gelen mesajlara karşılık verir.

Modların ikinci kısmı ise MIDI kablolarından gelen notaların nasıl çalınacağını gösterir. Eğer bu modu Poly olarak set etti iseniz, çok sesli (Poly Polyphonic) bir sentezleyici gibi bir seferde bir çok nota çalınabilir. Bir mesaj bir notayı başlatacak şekilde geldiğinde, MIDI ikinci bir notayı başlatacak bir mesaj gelene kadar bunu saklar. İlk nota, yalnızca cihaz bittigine dair bir mesaj aldığında yada aynı zamanda gelen notaların sayısı, sentezleyicinin ses sayıcısında diğerini geçtiğinde çalmayı keser. Eğer mod mono ya ayarlanmışsa, cihaz notaları tek sesli (Monophonic) dir, sentezleyici gibi bir seferde bir nota çalar. MIDI mesajı yeni bir notayı başlattığında cihaz bir önceki notayı bitirmesine dair bir mesaj olsa da olmasa da keser.

Polytimbral sentez Mümkün olan her yolda bir modun iki farklı durumunun kombinasyonu ile izleyen bölümde anlatılan dört modu sonlandırabilirsiniz. Her birisi kendi yöntemleri ile MIDI kanal mesajlarını yanıtlayabilirler.

Ayarlayın: Sentezleyici

2-4.1. Mod 1: Mod 1 Omni On/Poly dir. Bu moda ayarlanan bir cihaz, hangi kanaldan gelirse gelsin bütün mesajları alır ve notaları çoksesli olarak çalar. Bu pek çok

MIDI cihazının ilk açıldığında ayarlandığı moddur. Bu bir güvenlik modudur. Çünkü mesaj herhangi bir kanaldan gelen mesajları alır. Anaya boş duran bir makina sokamazsınız, çünkü mesajları doğru kanaldan almaz. **2-4.2.**

Mod 2 : Omni On/Mono modudur. Bu moda ayarladığınızda, cihaz herhangi bir kanaldan gelen mesajı alır ve tek sesli olarak çalar. Çoğu kişi cihazın notaları çok sesli çalmasını ister. Bu nedenle Mod 2 ender kullanılır.

2-4.3. Mod 3 : Bu mod Omni Off/Poly modudur. Bir cihaz farklı kanallardan mesaj alıp çok sesli olarak çalacağı zaman bu moda ayarlanır. Alıcı kanalımızı 16 MIDI kanalından herhangi biri olarak seçebilirsiniz. Çoğu MIDI cihazı bu moda ayarlandığında, mesajları tek bir kanaldan alır. Bazi çok sesli ritim sentezliyecileri (polytimbral synth.) birkaç kanaldan gelen mesajları alıp aynı anda çalabilirler. Bunlar her kanal için farklı bir sentezleyici panosu kullanırlar. Bu mod etkin bir şekilde sentezleyiciyi bir kaç küçük sentezleyiciye parçalar. Her yanıt farklı bir MIDI kanalından yapılır. Bir örnek olarak, sekiz sesli bir polytimbral sentezleyici düşünün. Mod 3'e ayarlayın, seslerin üçünü bir elektronik piyano panelini MIDI kanal 11 yanıtlayacak şekilde, seslerin ikisini kanal ikide flüt çalacak şekilde ve kalan üçünü obua çalacak şekilde kanal üçe ayarlayın. Sentezleyici şimdi üç ayrı sentezleyici gibi davranır. Her bir paneldeki her mesaj kendi MIDI kanalındadır. **2-4.4.**

Mod 4 : Bu mod Omni Off/Mono'dur. Cihaz

mesajları farklı kanallardan alıp, tek sesli olarak çalabilir. Bu mod yalnız polytimbral sentezleyicilerde kullanılır. Bu moddaki bir sentezleyici seslerin her bir farklı kontrolü için ayarlanır ve her sese farklı bir MIDI kanalı atar. Kanallar ardışıktır. İlk sentezleyici sesini herhangi bir kanala atayabilirsiniz. Ancak diğer tüm sesler ilkinizi izleyen kanallara ardışık olarak yerleşecektir. Sentezleyiciyi pek çok tek sesli sentezleyiciye ayırmanın etkili sonucu, her çalınan mesajın kendi MIDI kanalından olmasıdır.

2-5. KANAL SES MESAJLARI

Örnek kan olarak mesadörtbelselings bir polytimbral sentezleyiciyi Mod 4'e ayarlayalım. Dört sesin her birisini panelin bir parçasına ayarlayalım. İlk ses için bir MIDI kanal numarası atayalım. Diğer sesler için kalan kanallar otomatik olarak belirlenecektir. Eğer ilk ses için beşinci kanalı seçtiyseniz, ikinci ses altıncı, üçüncü ses yedinci ve dördüncü ses sekizinci kanalı kullanır. Mod 2 gibi, Mod 4'te genellikle çok sesli cihazlara göre tercih edilmez. Ancak bazı teknik sınırlamalarla zorunlu olabilir.

2-5. KANAL MESAJLARI

Artık MIDI cihazlarının nasıl, seri data iletişimi yaptığını Byte'ları ve ileri geri mesajlar yollayan MIDI kanallarını gördük. Artık bu mesajlarla, cihazların bir diğerine ne dediklerini tam olarak göreceksiniz. Bu bölüm kanal mesajlarını, ardından sistem mesajlarının listesini, özelliklerini, hangi mesajın ne tip bilgi taşıdığını ve

bölümün sonunda sa teknik yapıları anlatılacaktır.

İki tip kanal mesajı vardır: Kanal ses mesajı ve kanal mod mesajı. Kanal ses mesajları aktif durumdaki datayı MIDI cihazları arasında yollar. Klavye hareketlerini tanımlar, kontrolör davranımlarını ve kontrol panelinde basılan tuşları tanımlar. Kanal mod mesajları MIDI cihazının yanıtladığı kanal ses mesajlarının alım şeklini belirler. Tüm bu mesajlar, yalnızca ulaşması istenen kanala gitmesi için özel kanal numaraları ile yollanır.

2-5.1. Kanal Ses Mesajları

Çoğu kanal ses mesajı, belirlenmiş ses yüksekliğindeki, genlikteki, renkteki, süreklilikteki ve diğer ses kalitelerindeki müzik ile tanımlanır. Her mesaj en az bir ve genellikle iki Data Byte ve Status Byte'ı bu ses kalitesine erişebilmek için içerir.

Note On (Nota Başlangıcı): Note On mesaj sinyali bir notanın başlangıcındadır. Genellikle MIDI destekli bir klavyeden, kullanıcının bir tuşa ilk basışında yollanır. Note On bir alıcı cihazın bir notayı çalmaya başladığını gösterir. Bir Note On mesajı bilginin üç parçası olarak iletilir. Mesajın yollandığı kanal, ses yüksekliği düzeyi (0'dan 127'ye kadar) ve notanın atak şiddeti (0 - 127) şeklindedir. Gam değeri (0 - 127) C#'in beş oktav altında orta C vardır (Piyano klavyesinde C#'in iki oktav altı). En düşük ses yüksekliği değeri olan 0, ince C#'i simgeler. En yüksek değer olan 127 G#'dir (Şekil 7).

Şekil 11 Ses yüksekliği değerleri, Note On mesajından gelen notanın aynısıdır. Hız değeri de aynı şekilde çalışır. 0 değeri 0 notanın hiç olmadığını ve 127 çok meçabuk olduğunu gösterir. Hız sürekli kontrol edilerek notanın sönme durumunda olup olmadığına bakılır. Küçük bir hız değeri genellikle bir notanın yavaşça söndüğünü gösterir. MIDI cihazları hız değeri 64 iken Note Off yollayamazlar. **Kanal Basıncı** Çok Sesli Tuş Basıncı (Polyphonic Key Pressure): Bu mesaj herhangi bir başka notanın çalındığını rapor eder. Genellikle başka tuşlara basıldığını saptama yeteneği olan bir klavyeden gönderilir. Bir cihaz bu mesajı aldığı anda, fazladan vibrato gibi basıncı arttırıcı cihazlar eklenmediği sürece, herhangi bir yoldaki numaradaymış gibi davranım gösterir. PKP üç çeşit bilgi içerir; mesajın yollandığı kanal, notanın basınç etkisinin ses yüksekliği değeri (0 - 127) ve notanın basıncı (0 - 127).

Ses yüksekliği değeri Note On'da olduğu gibi geçerlidir. Basınç değerleri, 0, basınç olmama hali, 127, mümkün olan en yüksek basınç hali olarak sınırlanır. Basınç vibratoyu belirler, düşük basınç değeri dar vibrato (küçük miktarda) nota için etki ve yüksek basınç genellikle geniş (büyük miktarda) vibratoyu ayarlar.

Kanal Basıncı (Channel Pressure): Bu mesaj her bir notanın baştan başa olan basıncını rapor eder. Genellikle klavyeden başka tuşlara basılmayıp, kendi tuşlarına basarak yollanır. Bir kanal basıncı mesajını alan cihaz farklı

şekillerde karşılık verebilir, ama çoğu cihazlar, çaldığı tüm notalarını vibratoya ayarlama bilgisi olarak kullanırlar. Bu mesaj iki parça bilgi taşır; Yollanan kanal mesajı ve baştan başa basınç değeri (0-127). Buyla yollanmaz. Çünkü hafif MIDI mesajı PKP'de olduğu gibi mesajın basınç değeri minimum 0'dan maksimum 127'ye kadar sıralıdır ve genellikle notaları çalmakta kullanılan vibrato miktarının etkisidir. Kanal basıncı mesajının etkisindeki temel fark, tüm notaların basit bir MIDI kanalından çalınmasıdır. Kanal kontrol değeri olan kontrol Program Değişimi (Program Change): Program değiştirme mesajında sentezleyicinin üzerinde bir nokta kısmı değişikliğini sağlar. Kullanıcı, sentezleyicinin kısmı klavyesindeki bir tuşa bastığında devreye girer. Bu mesajı alan cihaz kullandığı kısmı değiştirir. Program değiştirme mesajı iki bilgi taşır; Yollandığı kanal ve yeni seçilen kısmın numarası (0-127). Olup olmadığını kontrol eder. Yeni Program değiştirme mesajının nedeni, alıcı MIDI cihazında yeni bir nokta kısmı seçmek içindir. Yollanan kısmın numarası, çalınan notanın karakteristik seslerini etkiler. Kısmın numarası tüm sentezleyicilerde standart sonuçlar vermezler. Zira sentezleyiciden sentezleyiciye kısımlar farklı olabilir. Kontrol Değişimi (Control Change): Kontrol değişimi mesajı bir MIDI cihazındaki yeni ayarlamaların kontrol bilgisini yollar. Bir kontrolör ile oynadığınızda ya da sentezleyicide veya diğer bir MIDI cihazında bir anahtarı

açtığından bu mesaj yollar. Çoğu sentezleyiciler kontrol değiştirme mesajının yollandığını anlayacak kontrollara sahip değildir. Bazı sentezleyicilerde ses genliği (Volume) kontrolü, kontrol değişimi mesajıyla yollanmaz. O anki halini MIDI mesajı ile iletmez. Her hatta iletilir. Eğer sürekli kontrol değiştirme mesajı alan cihaz, yeni kontrol ayarlamasına uyacak şekilde işlemini değiştirir. Kontrol değiştirme mesajı üç farklı mesaj parçası taşır; Yollanan kanalın numarası, 0'dan 127'ye kadar kontrol değeri anılan kontrolün çekildiğini ya da konduğunu gösterir. Ayrıca ayar değeri olarak adlandırılan, 0'dan 127'ye kadar bir numara, kontrolün ya da ilerki belirlemelerin ayarlamasını yapar.

Kontrol değişimi mesajının üç çeşit çalışma ve kontrol şekli vardır: Sürekli kontrol edici (Continues controllers), ayarlamamanın tüm sınırlarında kontrol yapabilir. Anahtarların anı olarak açık ya da kapalı olup olmadığını kontrol eder. Veri kontrolörleri (Data Controller), tuşun her basılışında adım adım girilen nümerik data doğrultusunu kontrol eder. Kontrol edicinin her tipinin kontrol numarası için, MIDI özelliklerinde ayrı bir sınır tahsis edilmiştir. 0'dan 63'e kadar olan numaralar sürekli kontrol edici, 64'den 95'e anahtarlar (bazı tiplerde sürekli kontrol edici de olabilir) ve 96'dan 101'e veri kontrol edicileri içindir. Bazı kontrol numaraları özel kontrol edicilere ayrılmıştır. Çoğu numaralar bu şekilde tanınlanmadan bırakılmış ve MIDI özelliklerindeki beklenmeyen gelişmeler böylece kontrol

edilebilir hale getirilmiştir. digünizde devreye girer. Bu mesaj 0'dan 127'ye, kadara olan değerleri yollanmış bir kontrolün yeni ayarlamasını taşır. Sürekli bir kontrol edici için, 128 değeri farklı ayarlamaları verebilir. 0'dan (kontrol edici sınırın altı her hattı ile çekilmiş). Eger sürekli kontrol edici 128 mümkün değerden fazlasına gereksiniyor ise, bu kesinlikle ayarlanabilir. Bu sefer MIDI cihazı iki kontrol değiştirme mesajı yollayabilir. Birinci mesajın ikincisi ile karışması 16384 farklı ayarlamaya izin verir. Anahtarlar için ayarlama değeri daha basittir. 0'dan 63'e herhangi bir değer anahtarın kapalı olduğunu, 64'den 127'ye ise açık olduğunu söyler. Üç kontrol değiştirme mesajına kadar data kontrol edici yeni değerleri taşıyabilir. İlk iki mesaj değişen değerlerin niteliğini belirler. Kontrol değerlerinin her ikisi de data kontrol edici mesajlar olarak alınırlar. Ayarlanan değerler sentezleyicinin çalışma niteliğini etkiler. Alıcı cihaz, 0'dan 16383'e kadar bir numara olan nitelik değerini, ayarlama değerine kombine eder. Üçüncü kontrol değişim mesajı ayarlamaadaki değişiklikleri belirler. Alıcı cihaz kullandığı değer azaldığını ya da arttığını veya tümüyle değiştiğini bildirir. Ayarlama değeri, alıcı cihazın gerçek değerinin ne kadar aşağı ya da yukarı değişeceğini ya da tümüyle yeni bir değere gidişini sağlar.

Gam Bağlantı Değişimi (Pitch Bend Change): Bu mesaj bir ses yüksekliği bağlantı kontrolörü için yeni ayarlama bilgilerini yollar. Genellikle ses yüksekliği bağlantı kont-

rol ediciyi açık duruma getirdiğinizde devreye girer. Bu mesajı alan bir cihaz, o anda çaldığı notaları gelen bağlantı ve ses yüksekliğine göre değiştirir. Bu ses yüksekliği bağlantı değişimi mesajı iki tür bilgi parçası taşır; Yollandığı kanal numarası ve 0'dan 16383'e kadar olan, yeni ayarlamayı gösteren ses yüksekliği bağlantı kontrol edicisi değeri.

2-5.2. Kanal Mod Mesajları yüksekliğini bulmak yerine, Kanal mod mesajları farklı MIDI cihazları için gelen mesajları ayarlar. Çalınan notayı keser ve bir cihazın lokal kontrolünü etkiler. Bunlar kanal ses mesajlarından daha basittir ve her mesajda o kadar fazla veri taşımazlar.

Yerel Kontrol (Lokal Control): Bu mesaj bir MIDI cihazının kendi ses üreticisi ile bağlantısını keser ya da sağlar. Genellikle bir Sequencer (sıralayıcı) ya da bilgisayar kontrollu sentezleyiciye mesajlar yollar. Bir klavyede bir sentezleyici cihazının alt durumunda yollanan çok kullanışlı mesajlar olabilir. Yalnızca gelen MIDI mesajlarıyla çalmak isteyebilirsiniz. Sentezleyicinin yerel (Lokal) kontrolünü kapatırsak, hiç bir etki göstermez.

Eğer sentezleyicinin klavyesini bir ana klavye olarak kullanacaksanız bu durum oldukça kullanışlıdır. Herhangi bir anda tuşlara ya da kontrollara basarsanız, sonucunda mesaj sadece iletilir ve sentezleyicinin kendi ses üreticisinde hiç bir ses üretilmez.

Bir yerel kontrol mesajı iki tür bilgi parçası taşır: Yerel kontrolün statusu ve yollandığı kanal. Status açık ya

da kapalı olabilir. notaları MIDI kabul modunda yeniden

Tüm Notalar Kapalı (All Notes Off) : Bir sentezleyici Note On mesajları yolluyor ve bunları çalıyor ve her nasılsa notaları kapatmak için kullanacağı tüm Note Off mesajlarını alamıyorsa, bazı notalar seçilerek çalınır. Tüm Notalar Kapalı mesajları bu durumu kolaylaştırmak içindir. Çalınan notaların ses yüksekliğini bulmak yerine, mümkün olan 128 ses yüksekliğinin her birisine basit bir Tüm Notalar Kapalı mesajı yollanarak alıcı sentezleyicinin tüm çalan notaları kesmesini sağlar. Bu mesaj yalnızca yollandığı kanal numarasını taşır. isten mesajı vardır. Sistem Ortak

Omni Mod Kapalı (Omni Mode Off) : Bu mesaj geldiğinde cihazın tüm alıcı kanallarındaki mesajları almak yerine, farklı MIDI kanallarındaki mesajları alabilmesi için MIDI kanal kabul modunu (MIDI Channel Reception Mode) değiştirir. Eger cihaz zaten Omni Mod Kapalı modunda ise bu mesajın bir etkisi olmaz. Bu mesaj genellikle bir sequencer ya da bilgisayar kontrollü farklı MIDI cihazı tarafından yollanır. Omni Mod Kapalı mesajı yalnızca mesajın yollandığı kanal numarasının bilgisini taşır.

Omni Mod Açık (Omni Mode On) : Bir MIDI cihazı, Omni Mod Açık mesajını, hangi kanaldan gelirse gelsin tüm kanal mesajlarını alıp MIDI kanal kabulünü (Eger zaten Omni Mod Açık modda değilse) değiştirsin diye alır. Omni Mod Açık mesajı da yalnızca yollandığı kanal numarasını taşır.

Mono Mod Açık (Mono Mode On) : Bir MIDI cihazı bu

mesajı aldığında tüm notaları MIDI kabul modunda yeniden ayarlayarak tek sesli olarak çalar. Bu mesajı yalnızca yollandığı kanal numarasını taşır. izn verir. Playback için

Poly Mod Açık (Poly Mode On) : Bir MIDI cihazı bu mesajı aldığında tüm notaları MIDI kabul modunda yeniden ayarlayarak çok sesli olarak çalar. Yalnızca kullandığı kanal numarasını taşır. belli bir isim ya da numara vererek

2-6. SİSTEM MESAJLARI (SYSTEM MESSAGES) için belli tuşlar MIDI cihazları sistem mesajlarını, tüm alıcılar aynı mesajdaki şekilde davranışın diye kanal numarası kullanmadan yollaralar. Üç tip sistem mesajı vardır: **Sistem Ortak Mesajları (System Common Messages)**, **Sistem Gerçek Zaman Mesajları (System Real Time Messages)**, **Sistem Birime Özel Mesajları (System Exclusive Messages)**. Sistem ortak mesajı, şarkı koordinat seçiminde ve MIDI cihazındaki ses ayarlamalarında yardımcı olur. Sistem Gerçek Zaman mesajları, performans ve kayıt boyunca farklı sequencerler üzerindeki zaman senkronizasyonunu yapar. Sistem Özel mesajları ise diğer MIDI mesajlarının yollanamayacağı özel cihazlar arasında yollarılır.

Sistem mesajlarının nasıl çalıştığını anlamak sequencerlere bağlıdır. Bir şarkının MIDI haberleşme sisteminde ne olduğunu anlamak önemlidir. Sequencerla bir cihaz (Elektronik bir bateri, bilgisayar veya belirlenmiş bir sequencer.) gelen mesajları kaydettiğinde, mesajlar arasında kalan zaman parçasını da alır ve kaydeder. Bu mesaj

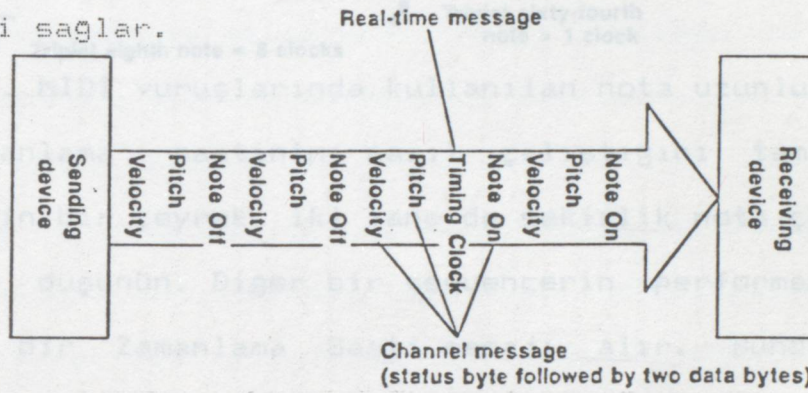
ayrısından önemlidir. Yani daha uzun diğer MIDI mesajlarını kaydınan " Sequence " denir. Çogu sequencerlar mesajları istediğimiz sequeunce'de düzenlememiz ve depolanmış farklı sequence'lerde hafızaya yollamamıza izin verir. Playback için hazır bitiş sequence'na şarkı denir.

Çogu sequencerler hafızalarında farklı şarkılar depolarlar. (Müzik dolapları bu yolla kayıt yaparlar.) Farklı şarkıları belli bir isim ya da numara vererek çağırabilirsiniz. Farklı şarkıları çağırmak için belli tuşlarını ve kısımlarını ayarladığınız bir elektronik batari buna iyi bir örnek oluşturur. Bu tuşlar sentezleyicideki kısım tuşları gibi davranabilirler. Bastığınız her tuş değişik bir şarkıyı çağırır. Çalmaya hazır olduğu zaman bir kaset banttan çalıyormuş gibi rahatça şarkıyı dinleyebilirsiniz. Tuşlardan biri çalmayı başlatır, bir diğeri ise keser.

Kasetteki ileri ve geri sarma gibi, şarkıyı duymak istediğiniz herhangi bir bölümden başlatabilirsiniz. Ayrı bir özellik olarak sequencer hızı arttırıp azaltabilir ve geriye çalabilir. (Tabiiki bunu yapmak için bazı kontrollarla oynamamız gerekir.)

2-6.1. Sistem Gerçek Zaman (Real Time) Mesajı :
Sistem Gerçek Zaman mesajlarının tümü de çok kısa ve basittirler. Yalnızca bir byte uzunluğundadırlar ve fazladan hiç bir veri taşımazlar. (Kanal numarası, ses yüksekliği, hız gibi.) Bu mesajlar MIDI cihazlarının zaman senkronizasyonunu da sağlar. Bu, bilginin tam gerektiği anda çağrılabilmesi

açısından önemlidir. Yani daha uzun diğer MIDI mesajlarını beklemek zorunda değildirler. Geçikme korkusuyla, sistem gerçek zaman mesajları - eğer gerekliyse - diğer mesajların tam ortasında yollanır. Örneği bir Byte'lık bir gerçek zaman mesajı bir Status Byte'ı ile bir kanal mesajının arasına yerleştirilir. (Şekil 8) Alıcı bir cihaz bir gerçek zaman mesajı aldığında hemen mesajdaki gibi davranır ve gerekli düzenlemeyi sağlar.



Şekil 8. Bir gerçek zaman mesajının yollanışı

Zamanlama Saati (Timing Clock) : Zamanlama Saati mesajı (MIDI vuruş olarak da bilinir.) en önemli ve en sık yollanan sistem gerçek zaman mesajıdır. Farklı sequencerlerin aynı MIDI sisteminde aynı tempo ile çalışmalarını sağlar.

Ancak, sequencer bir şarkıyı çaldığında diğer sequencerlere de bunu çaldığı şarkının temposunu taşıyan bir Zamanlama Saati mesajı olarak iletir. Zamanlama saatinin daha hızlı gelmesi alıcı cihazın şarkıyı daha hızlı çalmasına yol açar.

Standart bir zamanlama referansı için MIDI özellikleri her biri bir çeyrek notaya eşit olan 24 MIDI vuruşundan oluşur. Bunun anlamı diğer standart nota uzunluklarının şekil

9'daki gibi zaman eşdeğeri olduğudur.

o Whole note = 96 clocks	 Sixteenth note = 6 clocks
 Half note = 48 clocks	 Triplet sixteenth note = 4 clocks
 Triplet half note = 32 clocks	 Thirty-second note = 3 clocks
 Quarter note = 24 clocks	 Triplet thirty-second note = 2 clocks
 Triplet quarter note = 16 clocks	 Triplet sixty-fourth note = 1 clock
 Eighth note = 12 clocks	
 Triplet eighth note = 8 clocks	

Şekil 9. MIDI vuruşlarında kullanılan nota uzunlukları.

Zamanlama saatinin nasıl çalıştığını tam olarak anlamak için bir çeyrek, iki tane de sekizlik nota çalan bir sequencerı düşünün. Diğer bir sequencerin performans tempo ayarından bir Zamanlama Saati mesajı alır. Bundan sonra gelen çeyrek notayı çalmaya başlar ve bunu zamanlama saati ile sayar. 24 vuruş saydığı anda çeyrek notayı bitirir ve bir sekizlik notaya başlar. 12 vuruş daha saydıktan sonra sekizlik notayı da bitirir ve ikincisine başlar. Gelen MIDI vuruşunun hızı sequencerin çaldığı şarkının temposu belirler.

Başla (Start) : Başla mesajı basit bir mesaj olup, sequencerin aldığı şarkıyı çalmaya başlamasını söyler.

Dur (Stop) : Sequencerin çaldığı şarkıyı kesmesini söyler.

Devam (Continue) : Bu da şarkının son kesildiği noktadan tekrar çalmaya başlamasını söyler. Başla komutundan farklılık göstererek, parçayı en başından çalmaz, son dur

mesajın 3 dat kaldığı yerden devam eder. (2) : Sistem GİRİŞİLEME
 mesajı : Aktif Algılama (Active Sensing) : Mükemmel bir
 dünyadan MIDI kablolarının asla fişten çekilmeyecekmiş gibi
 olduğuna sanılır. Ancak bizim kusurlu dünyamızda sürekli fişte
 duramazlar. Ana klavyenizde boşa (iken) hiçbir yanıt
 alamazsınız. Hatta daha kötüsü bağlanmamış bir alet cihazda
 çalınan notalar, ana cihaz Note On ve Note Off mesajlarını
 almadığı için çalmaya devam eder. Aktif algılama mesajı bu
 hoş olmayan durumu gidermeye yardımcı olur. (3) : Yanıt
 ayarları Bir cihaz diğer bir cihaza Aktif Algılama mesajı
 yolladığında alıcı cihaz, iki cihaz arasında iyi bir bağlantı
 olduğunu saptar. Alıcı cihazın ilk Aktif Algılama mesajını
 almasından sonra, aynı mesajı her 300 milisaniyede bir almayı
 bekler. (4) : (Saniyede yaklaşık üç mesaj.) Eğer düzenli olarak
 Aktif Algılama mesajını alamazsa bağlantının kesilmesine izin
 verir ve tüm notaları çalmayı keser. Aktif Algılama düzenli
 aralıklarla gelmesi nedeniyle bağlantı testi amacıyla da
 kullanılır (Yollayıcı cihaz başka herhangi bir şekilde bağlı
 değilse) ve mesajı aldığı anda alıcı cihazda herhangi bir
 reaksiyon oluşmaz. Sequencer bu numara ile tanımlanmış şarkıyı
 bulur. Birkaç cihaz Aktif Algılama mesajını almaya ya da
 yollamaya keşlenebilir. Ancak bu uyum sorunu doğurmaz. Eğer
 bir cihaza Aktif Algılama mesajı alması söylenmiş ve hiçbir
 şey alamamıyorsa, bunu düzenli aralıklarla beklemekten
 vazgeçer. Ya da böyle bir şey söylenmemiş ve cihaz Aktif
 Algılama mesajı alıyorsa bu sefer de bunu hiç önemsemez. (5) :

Sistem Sıfırlama (System Reset) : Sistem Sıfırlama mesajı cihazın ilk açıldığı andaki durumuda yanlış ayarlama yüzünden kendini sıfırlayıp sıfırlamadığını sorar. Yanlış ayarlamadan anlatılmak istenen çoğunlukla MIDI kanal kabul modu 1'de olmasıdır. Bu durumda yerel (Local) kontrolde tüm sesler kapatılmıştır ve şarkı en başından çalmaya ayarlanmış olarak sequencer'dadır. (Aynı zamanda şarkının playback durumu da kapalıdır.) Sistem Sıfırlama mesajının her cihazda farklı bir sonucu olması nedeniyle cihazlar arasında yanlış ayarlama söz konusudur. **Tune Request** : Akord İsteği mesajı

2-6.2. Sistem Ortak (System Common) Mesajları : Sistem Ortak mesajları bir sequencer ya da sentezleyiciyi bir şarkıyı çalmaya hazırlayan mesajlardır. Farklı mesajlar bir şarkıyı seçmenizi sağlar. Şarkıda genel bir başlangıç noktası bulmak ve tüm sentezleyicileri, eğer çalma durumundaysalar, çaldırmak içindir.

• **Şarkı Seçimi (Song Select)** : Şarkı Seçimi mesajı bir sequencer'ın hafızasından özel bir şarkıyı yükleyip yüklemeyeceğini sorar. Şarkı Seçimi 0'dan 127'ye bir numara ile tanımlanır. Sequencer bu numara ile tanımlanmış şarkıyı bulur ve yükler. *(İleriki bölümlerde anlatılacaktır.)*

Şarkı Pozisyon İşaretleyicisi (Song Position Pointer) : Bu mesaj bir sequencer'daki şarkıya herhangi bir yerinden başlamasını söyler. 0'dan 16383'e kadar bir değer taşır ve bununla her yeni şarkıyı ayarlar. Bu numara şarkının pozisyonunu altı vuruşluk artışlarla başından ne kadar uzakta

127'e bir sayı ile tanımlanır. Bu üreticinin cihazın yolladığı mesajı tanımlamasını sağlar. (Örneğin 67, olduğunu söyler. Altı MIDI vuruşunun nedeni 16'lık bir notaya eşit olmasındandır. Aynı zamanda işaretleyici değeri şarkının başlangıcından 16'lık nota ölçüsüyle ne uzaklıkta olduğunu düşünebilirsiniz. İsa, mesajın geri kalanını da alır. Şarkı Pozisyonel İşaretleyicisi, bir sequencer bir şarkıyı çalarken kullanılabılır anlamına gelmez. Eğer bunu birkaç sequencer'da bir şarkıyı hep birlikte çalmak için kullanıyorsanız, senkronizasyon dışı sequencerı yöneltme mesajlarında zamanlama sorunlarıyla karşılaşabilirsiniz.

Akord İstemi (Tune Request) : Akord İstemi mesajı tüm balıcı MIDI cihazlarına kendilerinin akordlu olup olmadıklarını sorar. Bu mesaj pek ender kullanılır. Sayısal sentezleyicilerde (Digital Synthesizers) akord kayması görülmez. Örneksek (Analog) sentezleyiciler ise kendikendilerini akordlayamamanın eksikliğini sık sık duyarlar. Yukarıda da bu yöntem uygulanır. Diğer

Üretici Birime Özel Son (EOX - End Of Exclusive) : Birime özel mesajların sona erdiği bir sistemde MIDI cihazları EOX mesajını alır. EOX'dan bu bölümde söz edilmesinin nedeni; bir Sistem Ortak Mesajı gibi davranabilmesidir. Bu mesaj daha ayrıntılı olarak ileriki bölümlerde anlatılacaktır.

2-6.3. Sistem Birime Özel (System Exclusive) Mesajları : SEM (System Exclusive Messages) 'nin formatı üreticilere, her müşteriye ayrı bir şarkı içeren ürünler verme şansını yaratır. Her SEM bir status byte ile başlar. (Bu sonun SEM olduğunu tanımlar.) İlk data byte'ı ise 0'dan

127'e bir sayı ile tanımlanır. Bu üreticinin cihazın yolladığı mesajı tanımlamasını sağlar. (Örneğin 67, Yamaha'nın numarasıdır.)

127 Bir cihaz bir SEM aldığı anda mesajın yollandığı cihazın aynı üreticiden olup olmadığını kontrol etmek için tanımlama numarasına bakar. Eğer uygunsa, mesajın geri kalanını da alır. Aksi takdirde, alıcı cihaz mesajı hiç umursamaz. Bazı cihazlar, özellikle bilgisayarlar, program yardımıyla diğer üreticilerin cihazlarının yolladığı SEM mesajlarını alabilirler. Böyle cihazlar, farklı üreticilerin cihazlarının SEM'lerini de kullanarak, mesaj da yollayabilirler.

Bir SEM, yollayıcı cihazın istediği kadar data byte'ı içerebilirler. Bu data byte'ların (0-127) imalatçıya pek çok şeyi yapmasında kolaylık tanır. Bazı imalatçılar SEM data byte'larını sentezleyiciler arasındaki kısımlar için ileri geri bilgi göndermek amacıyla kullanırlar. Kısımları bir sentezleyiciden diğerine kontrol panelinden girmeksizin, basitçe yüklenebilsin diye bu yöntem uygulanır. Diğer üreticilerin de müşterilerine ürünleriyle ilgili bilgi göndermek amacıyla kullanırlar. Örneğin bazı Yamaha sentezleyicileri SEM'i özel Note On ve Note Off mesajlarını yollamak için kullanırlar. Bir cihaz SEM yollamayı bitirdiğinde, bir EDX mesajı yollayarak, sinyalin sonu olduğunu söyler. EDX mesajının alınmasıyla SEM'i önemsemeyen cihaz kabloya uyarı mesajı yollar. Tabii ki zamanlama saati ve diğer sistemlerin gerçek zaman mesajı gibi bazı mesajları SEM'in data byte'ları arasında sıkıştırabilirler. Bu mesajların basit bir status byte'ı (128-255) ile atanmasının

nedeni; alıcı cihazın önemsemediği SEM data byte'larının (0-127'de) uygulanabilmesi ve ona göre davranılabilmesi, önemsemediklerini ise geri yollaması içindir. Her nota iki data byte'ı vardır.

2-7. KULLANIMDA MIDI'nin bir status byte'ı içerir. Data byte'ları Şimdi de MIDI cihazlarının kendi aralarında ileri ve geri yollayabilecekleri mesajları göreceğiz. Cihazlar tümüyle bu mesajları kullandığında, bunların istenildiği şekilde çalışması için bilmemiz gereken birkaç nokta vardır.

2-7.1. Aynı Zamanda Olayları İletmek : MIDI seri bir arabirimdir. Kabloya arka arkaya basit mesajlar yollar. Bir kullanıcı, sentezleyicide aynı anda üç tuşa basarak bir akor yolladığında neler olur? MIDI her nasılsa birden mesajları kullanarak bu üç notayı aynı zamanda iletir. Cihaz başka tür bir mesaj MIDI'nin bu konudaki problemi; insan kulağından ve elinden daha hızlı olmasıdır. MIDI en hızlı mesajını (SEM hariç) saniyede 1040 mesajlık bir hızda yollar. Bir MIDI cihazı bunu daha hızlı şekilde okuyabilir. Kullanıcı bir akor bastığında, sesler duyulmayacak şekilde milisaniyelere bölünür. Aslında basılan nota serisi aynı zamanda gelecek kadar birbirlerine yakıdırlar. MIDI klavyesi bunları öyle bir hızda yollar ki biz onu aynı zamanda çalıyormuş gibi duyarız.

2-7.2. Çalışma Durumu : MIDI cihazları daha hızlı data iletimini yapmak için Çalışma Durumu denen bir teknik kullanırlar. Çalışma Durumu aynı tip mesajları her birisi için status byte kullanmadan yollama olanağı verirler. Alıcı cihaz bir status byte aldığında yenisi gelene kadar o status

byte'in söylediği modda kalır. Herhangi bir data byte'ını da o moda uygun kabul eder.

Örneğin bir seri nota mesajını düşünün. Her nota iki data byte'ı tarafından izlenen bir status byte'ı içerir. Data byte'larından biri ses yüksekliği, diğeri ise hız içindir. Çalışma durumu kullanılarak kullanılan bir Note On mesajında, cihaz her nota için bir Note On status byte'ı ve onu izleyen iki data byte'ı yollar. Alıcı cihaz mesajdaki ilk notayı aldığında Note On moduna geçer. İzleyen data byte'ındaki her değeri bu nota için hız ve ses yüksekliği değeri olarak yorumlar. Çünkü hızı 0 olan bir nota ses düzeyine sahip değildir. Data byte'ı 0 olan bir nota Note Off mesajı yollanmadan da kapanabilir. Yollayıcı cihaz başka tür bir mesaj yollamak istediğinde, mesajını kesip basitçe yenisini yollamaya başlar. Alıcı cihaz yeni bir status byte'ını ne kadar çabuk alırsa, muhtemel status byte'ını o kadar çabuk alıp buna bağlı olan data byte'ını yorumlar. Sürekli tekrarlanan data byte'ının eliminasyonundan sonra çalışma durumu mesaj byte'larını üretip, nota iletimini hızlandırır.

2-7.3. İşlem Dışı Mesajlar : MIDI mesajları MIDI cihazının özelliklerinin farklı bilgilerini taşır. Ne yazık ki MIDI cihazlarının tümü özelliklerinin tamamını kullanamazlar. Yeni bir cihaz arada sırada yapamayacağı bir şeyin mesajını alabilir. Ama bunları hiç önemsemezler.

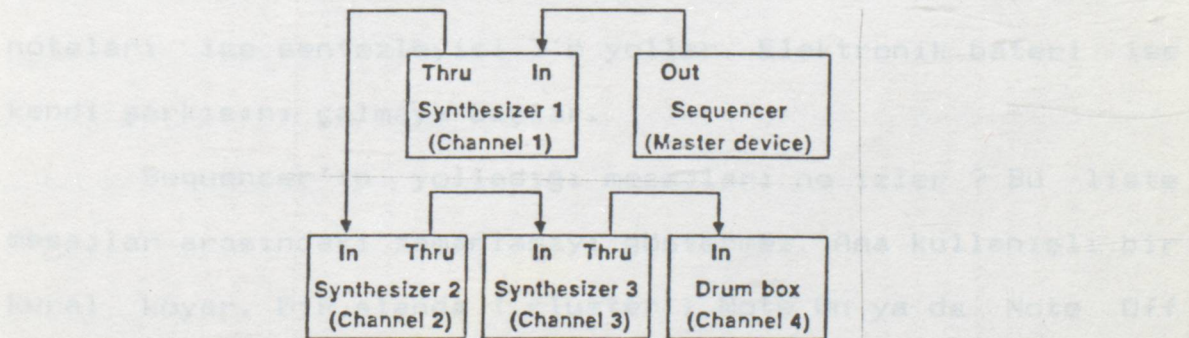
İşlem Dışı mesajlarının bir önemi şu kuraldır : Gerçekte bir cihaza yolladığımız bir mesaj o cihazdan yanıt

gelecek şekilde olmayabilir. Bir sentezleyiciye hız algılama klavyesi eklediğimizde, bu onun daha yüksek ve daha alçak gelen seslere yanıt verebileceği anlamına gelmez. Cihaz yapımından bu özelliğe sahip olmalıdır. Çünkü bir MIDI cihazı ne kadar çok özellik içerirse içersin, tüm MIDI mesajlarına yanıt verebilecek donanıma sahip olamaz.

2-7.4. Açıktaki MIDI Mesajları : MIDI mesajlarının nasıl çalıştığını görmenin en iyi yolu bir örneğe bakmaktır. Varsayılan MIDI sistemindeki mesajların nasıl olduğunu ve nasıl çalıştığını görmek, hangi farklı türde mesajların bir müzik yapımı için kullanılabileceğini gösterir.

Şekil 10 örnekte kullanılan MIDI sistemini gösterir. Bu sistem bir sequencer'a sahiptir. (Bir elektronik bateri ve üç sentezleyici ile elde edilmiştir.) Şekildeki oklar mesajın izlediği yolu gösterir.

Sequencer ana cihazdır. Sistemdeki diğer tüm cihazlara mesaj yollar. Diğer üç cihaz Mode 3'de (Omni Off/Poly) mesaj almaya ayarlanmıştır. Sentezleyici 1 MIDI kanal 1'den mesaj alır, sentezleyici 2 kanal 2'den, sentezleyici 3 kanal 3'den ve elektronik bateri kanal 4'den mesaj alır.



Şekil 10. Bir MIDI Sistem Örneği.

2-7.5. Bir Şarkıyı Çalmaya Hazırlanmak : Sequencer'ın sistemi çaldırması için yolladığı ilk mesajdır. Önce sequencer'ın kendi belleğinde sakladığı bir kısım transfer edilir. (Sentezleyici 2'ye) Ardından akordları yapılmış sentezleyicilerde ve elektronik bateride bir şarkı yüklenir. Aşağıdaki tablonun sol sütununda görülen mesajlar sequencer'ın yolladıklarıdır. Bunları mesajın status byte'ındaki herhangi bir data byte'ı ile yollarlar. Listelenmiş bu mesajların sonuçları ise tabloda sağ sütundadır.

<i>Message</i>	<i>Result</i>
System exclusive	All the devices wait to see if this affects them.
Manufacturer #61	Only synthesizer 2, made by manufacturer 61, tunes in.
A stream of data bytes	Synthesizer 2 gets a new patch.
EOX (End of Exclusive)	All devices tune in again.
Song Select	Devices that lack sequencers ignore this message.
Song #5	The drum machine loads song #5.
Tune Request	Synthesizer 3, an analog synthesizer, tunes up. The other devices ignore this message.

Sequencer'ın MIDI sistemini çalmaya hazırlamasından sonra müziğin küçük bir kısmını çalmaya başlar. Tiz notaları sentezleyici 1'e, orta notaları sentezleyici 2'ye, bas notaları ise sentezleyici 3'e yollar. Elektronik bateri ise kendi şarkısını çalmaya başlar.

Sequencer'ın yolladığı mesajları ne izler ? Bu liste mesajlar arasındaki zamanlamayı göstermez. Ama kullanışlı bir kural koyar. Bir alanda (cluster) Note On ya da Note Off mesajı görürseniz, sequencer bunları mümkün olduğu kadar

birlikte veya yakın olarak dışarı yollar. Zamanlama saati mesajı arasında sequencer'ın çaldığı notalara senkronize olmuş elektronik bateri de aynı zaman diliminde düzgün bir şekilde çalar. Byte'ların MIDI kablosu üzerinde nasıl yollandığı Tablolarda oluşturulmuş tüm veriler yalnızca birkaç saniyelik bir müziktir. Sequencer tumba için Note On ve Note Off mesajları yollamaz. Çünkü elektronik bateri bunu kendü sequencer'ından yapar. Atina'na geçebilirsiniz.

2-8. MIDI MESAJLARI'NIN TEKNİK ÖZELLİKLERİ

Eğer MIDI'yi iyice kavramak istiyorsanız ve Sistem Birime Özel Mesajları'nı kullanarak veya sentezleyicinin ne yaptığını tam olarak anlayarak kendi MIDI programınızı yazıp ellerinizi kirletiyorsanız; keene azından belli başlı MIDI çalışma düzenlerini bilmeniz gerekir. Bitler ve byte'ları elektrik akımı kullanarak bir MIDI kablosu üzerinden nasıl yollandığını da bilmelisiniz. 9'a değeri alır ve tam dönüşünde bir. İkinci ilk kısımda MIDI mesajlarının basit bir açıklamasını okudunuz. Genel düzenlemeler gösterildi ve mesaj grupları tanımlandı. Her bir grupla birlikte kullanabileceğiniz mesajların özel tiplerini ve her mesajın taşıdığı bilgiyi gördünüz. Şimdi de ikinci kısımda başladığımız MIDI mesajlarının geri kalan ayrıntılarını öğreneceğiz. Her büyük mesaj grubunun çalışma düzenini analiz edecek, özel byte ve bitlerdeki bilgiyi nasıl anladığını göreceksiniz. (Decimal) sistem denir. Çünkü bu her bir hane Bu bölümdeki mesaj betimlerini anlayabilmeniz için,

ikilik (binary) sayıları yakından tanımalısınız. Bunun için bu bölümün ilk kısmını okumanı yeterli olacaktır. Burada bitleri ve byte'ları da yakından tanıyacaksınız. Eğer bu bitlerin ve byte'ların MIDI kablosu üzerinde nasıl yol aldıklarını mesak ediyorsanız, sonraki bölümü okumalısınız. Burada basit seri haberleşmeler açıklanmıştır. Yok eğer, zaten gerçek bir bilgisayar tutkunuysanız, tüm bunları atlayıp mesajların anlatımına geçebilirsiniz.

2-8.1. İkili (Binary) Aritmetik Düzeni

İkili sayıları anlamak için bir arabadaki kilometre sayacını anlamakla işe başlayabiliriz: Kilometre sayacının her bir hanesi on farklı değer gösterebilen bir çemberdir; 0'dan 9'a. Siz arabayı sürerken en sağdaki hane bu değerleri 0'dan 9'a kateder ve 9'a vardığında tekrar 0 olur. Tam dönüş bittigi zaman aynı anda onun solundaki hanenin numarası bir artar. Bu komşu hane de 0'dan 9'a değer alır ve tam dönüşünde bir solundakini bir arttırır. İşlem en soldaki haneye kadar tekrarlanır durur.

Çünkü kilometre sayacının her bir hanesi on farklı degere sahiptir. En sağdakinin 10 tam dönüşü onun yanındaki haneyi bir kez döndürür. 100 dönüşü yanındakinin bir solundakini, 1000 dönüşü bir sonrakini, 10000 dönüşü daha sonrakini ve böyle sürer gider. Bu da kilometre sayacını en sağdakinden 10 çevrim gücünde sağdan sola arttırır. Bu sayı sistemine onluk (Decimal) sistem denir. Çünkü bu herbir hane için 10 mümkün değerden oluşmaktadır.

Şimdi aynı kilometre sayacını tekrar gözümüzün önüne getirelim, ancak bu sefer 0'dan 9'a kadar olan sayıların yerine, her sayaç çemberi için iki değer içersin: 0 ve 1. En sağdaki hane tam tur yaptığında solundaki bir artar. Bu komşu hane de tam bir tur yaptığında onun bir solundaki ve böylece sürer. Bu odometredeki her hane yalnızca iki muhtemel değere sahiptir. En sağdaki hanenin, komşu haneyi bir arttırması için yalnızca iki dönüş yapması yeterlidir. 4 dönüş, onun daha solundakini, 8 dönüş daha soldakini, 16 daha sonrakini v.s.

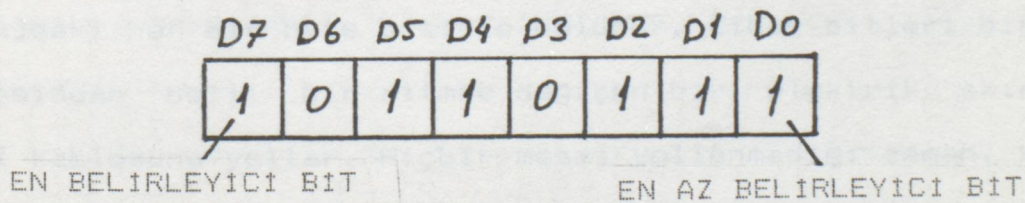
İkili sistendeki bir odometrede, belirleyici değer ikinin katlarıdır: 2, 4, 8, 16, 32, 64, 128, 256, ve böyle...

Bu sayma sistemine ikilik (Binary) Sistem denir.

2-8.2. Byte'lar ve Bit'ler

Tek bir byte 8 haneli ikilik bir sayıdır. Her bir haneye de bit denir.

Bir byte'daki bitler genellikle 0'dan başlayarak sağdan sola numaralanırlar. Çünkü en sağdaki hane 2'nin en küçük katını verir, ve sola doğru gittikçe hane değerleri artar. Bu sonuca göre en sağdaki haneye " En az belirleyici bit " (Least significant bit) denir. BKZ. Şekil 11.



Şekil 11. Bir byte'da bitlerin sıralanışı.

Bu bölümde tanımlanmış bitler, başına bir D harfi (Digit-Hane) gelerek bit numarasıyla anılacaktır.

Tek bir byte, 0'dan 255'e kadar değişen değerler alabilir ve 8 bitten oluşur. Byte'a fazladan bir bit katarak (9 bit), menzilini iki katına çıkarabiliriz. Yani her byte 0'dan 511'e kadar numaralarla belirlenebilir. Byte'dan bir bit çıkarırsak (7 bit), mümkün değerler yarıya iner: 0'dan 127'e. Eklenen her bit ikilini sayının değerini ikiye katlar, çıkarttığınız her bit'de yarıya böler.

Her byte'ın 8 bit sınırlamasının nedeni; tek bir byte'ın 255'den büyük değer alamıyacağıdır. Bununla birlikte iki byte'ı birleştirirseniz; 16 bitlik bir toplama sahip olursunuz. Bu da 0'dan 65535'e değer alabilir. 16 haneli ikili bir sayı oluşturduğunuz zaman, iki farklı byte oluşturmak üzere bunu ortadan bölebilirsiniz. Sol byte'a En Belirleyici Byte (Most Significant Byte) - MSB -, sağ byte'a da En Az Belirleyici Byte (Least Significant Byte) - LSB - denir. Bu iki byte'ı bir bağlantı kablosu üzerinden yolladığımızda, alıcı cihaz ikisini birlikte alır ve 16 bitlik bilgiyi okur.

2-8.3. Seri İletim

Bir MIDI cihazı, bir MIDI mesajı yolladığında, mesajdaki her bir byte bitlere bölünür. Cihaz bitleri birbiri arkasından belli bir ritimde değişen bir elektrik akımıyla MIDI kablosuna yollar. Hiçbir mesaj yollanmadığı zaman, küçük bir elektrik akımı kabloda bir cihazdan diğerine akar. Cihaz

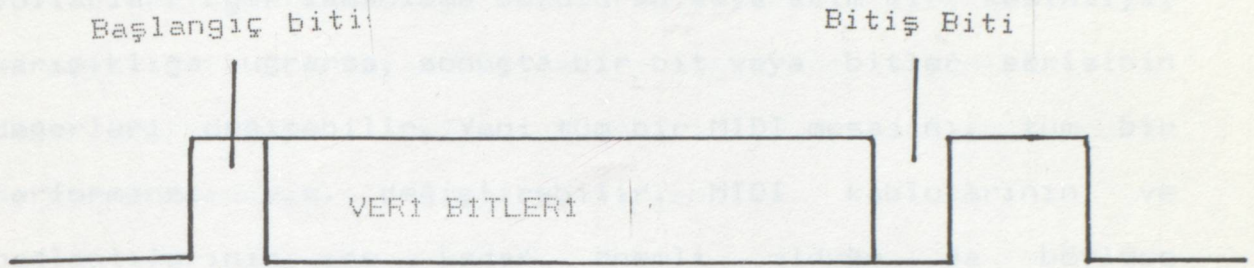
-1- biti yolladığı zaman akım çok kısa bir anda durur. Cihaz -0- biti yolladığında, akım tekrar geri döner. -0- bitidir.

Bu işlem çok basit gözükse de aslında daha karmaşıktır. Eger cihaz bir dizide üç tane -1- biti yollarsa, alıcı cihaz bunu akımı kesecek tek bir bit olarak yorumlamaz mı ? Hayır, öyle yapmaz. Alıcı ve verici cihazların ikisinde kontrol bitlerini zaman sinyaliyle kesecek şekilde yollar.

2-8.3.1. Baud Hızı (Baud Rate) : Tüm MIDI cihazları, hem alıcı hem verici olarak, 31250 baud hızla veri iletimine ayarlanabilirler. Bu, dayı saniyede 31250 bit demektir. Yollayıcı cihaz, her bir biti ya bir non, ya da off, akım durumuyla saniyenin $1/31250$ 'sinde yollar. Alıcı cihaz da gelen bilgiyi saniyenin $1/31250$ 'sinde alır ve her bir parçayı ya -1- (akım yok), ya da -0- (akım var) olarak okur. Bu zamanlama standartı birbiri ardınca cihazdan yollanan -1- / ler ve -0- / ları veri kaybına neden olmadan veya karışıklığa yol açmadan işleme koyar. Byte'ın 8 bitini takiben bitiş bitini

2-8.3.2. Başlangıç ve Bitiş Bitleri : MIDI cihazları için bilinen bir baud hızı ayarını her biri diğerinden ayrılmış bitler problemi çözer. Ancak eğer bitler kablo üzerinden sabit bir akımla geliyorlarsa, alıcı cihaz byte'ları nasıl ayırır ? Bir byte'ın başlayıp bittiğini nasıl söyleyebiliriz ? Ve eğer bir mesajdaki ilk bit -0- bitini ise, alıcı cihaz bunun yollandığını nasıl bilebilir ? Şu gibi yorumlar. Bitiş bitini ayırır MIDI her byte'da fazladan iki bit kullanır, byte'ın başlangıcında ve bitişinde. İlk bite Başlangıç Biti, ikinci

bite ise Bitiş Biti denir. İlki byte'ın başında gelen bir -1- bitidir. İkincisi ise sonunda gelir ve bir -0- bitidir.



Şekil 12. Bir byte MIDI sinyalinin şekli

Başlangıç ve bitiş bitlerinin nasıl çalıştığını görmek için bir örneği inceleyebiliriz: İki MIDI cihazı birbirlerine bağlanmışlar ve mesaj iletmektedirler, alıcı cihaz MIDI'nin portuna bakar ve sabit bir akımın yollayıcı cihazdan geldiğini görür. Ancak yollanan hiç bir byte yoktur. Bu sefer yollayıcı cihaz bir byte yollar. Bu -1- ile başlayan bir başlangıç bitidir. Byte'ın 8 bitini takiben bitiş biti olarak bir de -0- gelir. Alıcı cihaz akımdaki değişiklikleri izler ve başlangıç biti ile akımda ani bir kesilme belirler. Bunu başlangıç biti olarak ayırır ve sonraki 8 biti sayar. Ardından 31250 baud'luk hızıyla uygulamaya geçer. Bitiş bitini 8 bitten sonra aldığında, bu byte'daki bitleri almayı keser. 8 bitin hepsini tek bir byte olarak görür ve bunu bir MIDI mesajının bir parçasıymış gibi yorumlar. Bitiş bitini ayırır, çünkü o bir -0-'dır. Sabit akıma geri dönerek yani bir byte'a başlayacak sinyali getiren başka bir başlangıç

bitini gözler. Eger ki her bir byte 0'dan 255'e deger alabilir. Gördüğünüz gibi MIDI cihazları, bitleri ileri ve geri çok yüksek bir hızda kesin bir zamanlama gereksimi ile yollarlar. Eger zamanlama bozulursa veya akım bir kesintiye, karışıklığa uğrarsa, sonuçta bir bit veya bitler serisinin degerleri değişebilir. Yani tüm bir MIDI mesajını, tüm bir performansın v.s.b. değiştirebilir. MIDI kablolarının ve bağlantılarının ne kadar önemli olduğu da böylece görülebilir.

İkinci Bit - D7 - daima 1 - degerindedir. Bir MIDI 2-8.4. MIDI Mesajlarının Formatları da bunu bir durum Byte. Bölüm 2'de farklı cins MIDI mesajlarını tanıdınız ve bunların 0'dan 255'e kadar degerler alabileceğini gördünüz. Eger bu degerlere ikilik sayılar olarak bakarsanız MIDI cihazlarının onları yolladığı ve aldığı şekilde - bu sefer yalnızca - şaşmaz mantıksallıktan çok belli sınırlar ve kurallar içinde yönetiliyormuş gibi görünür.

İkinci grubun bitleri ikili sayılar olarak MIDI degerlerine baktığınızda özel birşey vardır: MIDI mesajdaki byte'ları 8 bitlik basit ikili sayılar (0-255) olarak kullanmaz. Bunun yerine her byte'ı daha küçük bit gruplarına böler, her grup kendi sayısıyla yalnız durur. Örneğin bir durum byte'i üç bit grubuna ayrılabilir (Şekil 13.) : Bir grup iki degerli basit bir biti içerir: 0 ve 1. Diğer bir grup 0'dan 7'e deger alan 3 alt grup içerir. Ve üçüncü grup ise 0'dan 15'e deger alabilen geri kalan 4 biti içerir. Bu durumda tek bir byte gerçekte üç farklı sayı taşır, bu yeterince açık bir olay

gibi görünmez, değer 0'dan 255'e değeri alabilen basit bir onluk sayı gibidir bakarsanız.

Sonraki ~~iki~~ kısım MIDI byte'larının nasıl iki tipe ayrıldığını gösterecektir. Durum byte'ları ve Veri byte'ları genellikle tek bir byte ile farklı değerlerde taşınırlar. ~~olarak yorumlanır.~~ Durum Byte'ı (Status Byte) : Bir Durum Byte'ı daima üç kısma bölünmüş bir formatta kullanılır.

~~Çoğu~~ 1xxxxxxx'ları düşük rezolüsyonlu veri taşır. Herhangi En Belirleyici Bit -D7- daima -1- değerindedir. Bir MIDI cihazı D7 olarak bir -1- değeri alırsa bunu bir Durum Byte'ı olarak yorumlar, Veri Byte'ı olarak değil! MIDI yüksek ~~rezolü~~ Bitlerin ikinci grubu D6'dan D4'e kadar olanlar (1xxx) bu Durum Byte'ını ileten cinsten mesajlar olacak özelliği taşır. 8 farklı cins mesaj sunabilecek üç bit vardır. En Belirli Bitlerin üçüncü grubu D3'den D0'a (xxxx) kadar olanlar, üçeşitli tipte bilgileri verirler. İkinci grubun bitlerinin özelliklerine göre bunların işlevleri belirlenir. Eğer mesaj bir Kanal mesajı ise (Note On veya Note Off), o zaman 4 bit MIDI kanal numarasını taşır. (Yani 0'dan 15'e bir sayı.). MIDI kanalları pratikte 1'den 16'ya kadar numaralanır, bir MIDI cihazı genellikle ikili değere kanal numarasını gösterdiğinde 1'ekler. Eğer mesaj bir Sistem Ortak Mesajı ise, veya Sistem Gerçek Zaman mesajı ise, son 4 bit daha ziyade mesajın cinsini belirler. (Bu kısımdaki alt başlıklar bu mesajları daha ayrıntılı açıklayacaktır.) ~~Veri~~ Veri Byte'ları (Data Bytes) : Veri Byte'ları Durum

Byte'larından daha basit bir formata sahiptirler. Bunlar yalnızca iki bit grubuna ayrılırlar:

Bu 0xxxxxxx mesajlar beş gruba ayrılmışlardır:

İlkan bit D7 daima 0 değerindedir. Bir MIDI cihazı bir D7 olarak bir 0-0 bitisi aldığı anda bunu bir veri byte'ı olarak yorumlar. Son 7 bit ise (D6'dan D0'a) 0'dan 127'e bir veri değeri taşırlar. Gerçek zaman mesajları,

Çogu MIDI mesajları düşük rezolusyonlu veri taşırlar. Herhangi bir mesaj 0'dan 127'e değeri balansa basit bir veri byte'ıdır. Diğer tip MIDI mesajları 0-127'den daha geniş bir aralığa gereksinim duyarlar. Bu nedenle, MIDI yüksek rezolusyonda veri taşımak için iki veri byte'ı kullanabilir. Bir cihaz aldığı iki yüksek rezolusyonlu verinin byte'ından genellikle ilkinin En Az Belirleyici Byte, anikincisinin En Belirleyici Byte olarak yorumlar. Kontrolde Değiştirme mesajlarında bunun bir istisnası olacaktır. Her veri byte'ının D7'si ayrıldıktan sonra, kalan 14 bit alınırsa, hepsiyle birlikte 0'dan 16383'e bir değeri taşıyabilir.

Birimsel (INDIVIDUAL) MIDI Mesaj Formatları

Bu kısım MIDI mesajlarının tümünü bir listesi sunmayı amaçlamaktadır. Herbiri, yollandığı emre göre özel byte bileşiklerine ayrılmıştır. Her bir byte'ın formatı özel mesajlara göre herhangi bazı bitleri içermektedir.

Bitler küçük harflerle belirlenen bazı farklı değeri alırlar. Bir mesajdaki tekrarlanan harfler aynı grup bitlere aittir. (Ve her bir bit için belirleyici bir anlam

taşımazlar. 8) İzleyen her mesajın formatı 8 bit değerlerinin kısa bir açıklamasıdır. Yan klavyeler için bu değer 64'dür.

But kisimdaki mesajlar beş gruba ayrılmışlardır: 60
 * Kanal ses mesajları, 60
 * Kanal mod mesajları, 60
 * Sistem ortak mesajları, 60
 * Sistem gerçek zaman mesajları, 60
 * Sistem birime özel mesajları. 60

Her bir gruptaki mesajları bilinen bir formatla tanınırlar. 60

2-8.4.1. Kanal Ses Mesajları : Kanal ses mesajları tümüyle bir durum mesajı olarak gelen, durum byte'ındaki D6'dan D4'e kadar olan bitleri kullanırlar. D3'den D0'a kadar olanları yollamaya devam ettiği belirli kanal numaraları olarak kullanır. Kanal ses mesajı daima bir ya da iki veri byte'ından sonra gelir. Mesajın yollandığı kanalı belirler. 60

Note On : Note On mesajı bir notayla başlar, notanın ses yüksekliğini ve ortak şiddetini söyler. 60
 Formatı: 1001nnnn 0kkkkkkk 0vvvvvvv

nnnn (0-15) : Mesajın yollandığı kanalı belirler. 60
 kkkkkkkk (0-127) : Yarım adımdaki ses yüksekliğinin en yüksek değerini, ses yüksekliğinin en yüksek değeri olarak belirler. 60
 orta ton -Do- sesini simgeler. 60
 vvvvvvvv (0-127) : Tuş atak şiddetini belirler. En yüksek değer, en yüksek şiddettir. 0 değeri hiç atak 60
 nnnn (0-15) : Mesajın yollandığı kanalı belirler. 60

olmamasını. **psimgeler** : ve bu **Note Off** mesajına eşdeğerdir. **Basing** duyarlılığı olmayan klavyeler için bu değer 64'dür.

Note Off : **Note Off** mesajı bir notanın sonunda gelir ve ses yüksekliği ile notanın belirli şiddetini söyler. **Verir**.
Bir dav **Format** : 1000nnnn0kkkkkkk0vvvvvvv diğer cihazlardaki **geçit** nnnn (0-15) : Mesajın yollandığı kanalı belirler.

kkkkkkk (0-127) : Yarım adımdaki ses yüksekliğinin en yüksek değerini en yüksek ses yüksekliği olarak belirler. 60 değeri orta -Do- 'dur. **Yeni bir yolun veya ritmin sayıları** vvvvvvvv (0-127) : Tuş kaybolma şiddetini belirler. En yüksek değer en yüksek şiddettir. **Change** : Kontrol değişim mesajı. **Çok Sesli Tuş Basıncı (Polyphonic Key Pressure)** : Bu mesaj bir klavyedeki sıradan bir tuşun üzerindeki basıncı rapor eder. **Format** : 1010nnnn0ccccccc0vvvvvvv

Format : 1010nnnn0kkkkkkkk0pppppppp kanalı belirler. nnnn (0-15) : Mesajın yollandığı kanalı belirler. **bir MIDI** c kkkkkkkk (0-127) : Tuşun yarım adımdaki vuruşunun en yüksek değerinin en yüksek ses yüksekliği olarak belirler. 60 değeri -Do- olarak görev yapar. **bu ve bu parametre tiplerini** alar. ppppppp (0-127) : Tuş basıncını belirler. En yüksek değer, en yüksek basınçtır.

Kanal Basıncı (Channel Pressure) : Tüm klavyedeki basıncı rapor eder. **Çoğu kez dokunma sonrası mesajı (Aftertouch messages)** olarak anılır.

Format : 1101nnnn0pppppppp **edici**.

nnnn (0-15) : Mesajın yollandığı kanalı belirler.

PPPPPPP (0-127) : Tümüleşik basıncı belirler. En yüksek değer en yüksek basınçtır.

Program Değişimi (Program Change) : Program Değişimi mesajı bir sentezleyicideki yolun değişim sinyalini verir. Bir davul makinasındaki ritm değişimi veya diğer cihazlardaki çeşitli değişimler gibi, *session* kontrol edici.

Format : 1100nnnn0ppppppp

nnnn (0-15) : Mesajın yollandığı kanalı belirler.

ppppppp (0-127) : Yeni bir yolun veya ritmin sayısını belirler. 31 değerleri için LSB (Least Significant Bit). Kontrol Değişimi (Control Change) : Kontrol değişim mesajı 64 farklı tiplerdeki sentezleyici kontrollerinin ayarlanması bilgisini taşır.

Format : 1011nnnn 0ccccccc 0vvvvvvv

nnnn (0-15) : Mesajın yollandığı kanalı belirler.

ccccccc (0-127) : Bir sayıyla simgelenen özel bir MIDI cihaz parametresinin değişimini veya kontrol edicilerin sayısının taşındığını gösterir. MIDI özellikleri farklı sayıları, 64 farklı özelliklere atar ve şu parametre tiplerini alır:

- 0 -- Belirlenmemiş.
- 1 -- Modülasyon çemberi veya "lever".
- 2 -- Soluk kontrol edicisi.
- 3 -- Belirlenmemiş.
- 4 -- Ayak (Foot) kontrol edicisi.
- 5 -- Portamento zamanı.

- 67-- Veri girişi MSB (Most Significant Byte).
- 78-- Ana ses (Volume).
- 89-- Balans.
- 90-- Belirlenmemiş.
- 101-- Pan. istenli parçaları.
- 112-- İfade (Expression) kontrol edici.
- 12-15 -- Belirlenmemiş.
- 16-19 -- Genel amaçlı kontrol ediciler.
- 20-31 -- Belirlenmemiş.
- 32-63 -- 0-31 değerleri için LSB (Least Significant Bit).
- 64 -- Damper pedalı (sustain).
- 65 -- Portamento.
- 66 -- Sostenuto.
- 67 -- Yumuşak (Soft) pedal.
- 68 -- Belirlenmemiş.
- 69 -- Tutma 2 (Hold 2).
- 70-79 -- Belirlenmemiş.
- 80-83 -- Genel amaçlı kontrol ediciler. (5'den 8'e numaralanırlar.)
- 84-91 -- Belirlenmemiş.
- 92 -- Tremelo derinliği.
- 93 -- Akor derinliği.
- 94 -- Celeste (Detune) derinliği.
- 95 -- Phaser derinliği.
- 96 -- Veri artışı.

97 -- Veri azalışı.

98 -- Registersiz parametre numara LSB'si.

99 -- Registersiz parametre numara MSB'si.

100 -- Registerli parametre numara LSB'si.

101 -- Registerli parametre numara MSB'si.

102-121 -- Belirlenmemiş.

122 -- Kanal Mod mesajları için ayrılmıştır.

vvvvvvvv (0-127) : Yeni kontrol ayarlamasının değerini belirler. Sürekli kontroledici durumunda, değer 0'dan 127'e değişebilir. Eğer kontrol bir anahtar ise ve yalnızca açık ya da kapalı ayarına sahipse, 0'dan 64'e kadar olan değerler kapalı 64'den 127'e olanlar açık olduğunu gösterir.

Veri değerleri listesinde daha ayrıntılı açıklama isteyen bazı fonksiyonlar vardır. 32'den 63'e kadar olan değerler, 0-31 arasındaki değerlere karşıt cevap için LSB taşırırlar. MIDI iki kontrol değişim mesajını birlikte kullanır. Her bir değer menziline bir yüksek rezolusyon değerini taşır. Örneğin özel bir cihazda foot kontrol edicisini hareket ettirdiniz, birinci kontrol değişim mesajını cihaz ilk veri byte'ındaki 4 sayısı ile yollayarak, belirler ve yeni ayar değerinin en belirleyici byte'ını ikinci veri byte'ında yollar. İkinci kontrol değişim mesajında, cihaz kontrol edici numarasını 36 olarak belirler (Kontrol edici sayılarının LSB menziline 5. değeridir.) ve

en az belirleyici veri byte'ını içerir. Alıcı MIDI cihazı ikinci kontrol değişim mesajından LSB veri byte'ı ile 0'dan 16383'e veri değerini yüksek rezolusyonlu üreten birinci mesajdan MSB'yini içerir. Alıcı cihaz belirli bir parametreyi kontrol eder. Bir kontrol edici yalnızca düşük rezolusyon veriye gereksiniyorsa, o zaman yollayıcı cihaz bir MSB kontrol değişim mesajı yollar ve bunu bir LSB kontrol değişim mesajı izlemez. Alıcı cihaz bunu 0'dan 127'e bir düşük rezolusyon veri byte'ı olarak yorumlar. (97) mesajı alırsa, bu bir Veri kontrol ediciler tarafından ayarlanan yeni sentezleyici parametrelerinin iletiminde, listedeki diğerlerdeki farklı kontrol edici değeri vardır. Bu değerler 6 ve 38 (MSB veri girişi ve LSB veri girişi.), 96 ve 97 (veri artışı ve veri azalışı.), 98 ve 99 (Registersiz parametre sayılı LSB ve registersiz parametre sayılı MSB.) ve 100 ve 101 (Registerli parametre sayılı LSB ve registerli parametre sayılı MSB.). Yeni bir parametre yollamak için bir cihaz registersiz veya registerli parametre sayılı kontrol değişim mesajını önce yollar ve ikinci veri byte'ındaki parametre sayısının LSB ve MSB'sini taşır. Alıcı cihaz 0'dan 16383'e bir parametreye yaratacak değerleri içerir. Bu parametreye sayıları bir sentezleyici parametresine karşılık gelir. Böyle ayarlamalar bir gösterge zarfının parçası veya bir örnekleme hızının ayarlamasıdır. Bilinen parametreler MIDI özelliklerinde ki numaralara atanır. Bu parametreler kontrol değişim mesajının registerli parametre numaralarına yollanmasında

kullanılır. MIDI özelliklerinin registersiz parametreli kontrol değişim mesajlarına yollanabilmesi vasıtasıyla parametreli sentezleyiciler bir değere atanmazlar. Bir byte birliğe Bir zamanlar bir alıcı cihaz belirli bir parametredeki kontrol değişim mesajı sayılarının parametresinde bir parçasını alırdı. Bu da veri artışı, veri azalışı veya kontrol değişim mesajının parametre için yeni bir değere ayarlayacak olan veri girişini bekler. Eğer cihaz bir veri artışı alırsa (96) veya veri azalışı (97) mesajı alırsa, bu durumda özel birimler tarafından ayarlanmış artan veya azalan akım parametresindeki mesajın değerinin ayarlanmasında kullanılır. Eğer cihaz veri girişi MSB mesajı (6) alırsa, bu durumda, mesajın değerinin ayarlanmasıyla parametre 0'dan 127'e yeni bir değere ayarlanır. Eğer bir veri girişi LSB mesajı (38)'nın arkasından gelirse, bu da yeni parametre değerini 0'dan 16384'e yeni bir değere ayarlar.

Bir zamanlar bir parametere sayısı, kontrol değişim mesajının parametre sayısının bir parçası tarafından ayarlanırdı. Herhangi bir altbirim veri artışı, veri azalışı veya kontrol değişim mesajının parametre sayısının yeni parçası yeni parametre sayısını belirleyene kadar kontrol değişim mesajının veri girişi bu özel parametreye etki eder.

veri 6 Ses Yüksekliği Değişimi (Pitch Bend Change) : Bu mesaj ses yüksekliği bağlantı çemberinin ayarındaki bir değişimi rapor eder.

Format : 1110nnnn 0bbbbbbb 0bbbbbbb 00000000 bir

sentez)nnnn1 (0-15) : Mesajın taşındığı kanalı belirler. veya
 kapalı)bbbbbbb dbbbbbbb. (0-16384) : Yeni ses yüksekliği
 değişimi yayarının değeri belirlenir. İkinci veri byte'ı
 birlikte taşınarak bir yüksek rezolüsyon değeri elde edilir;
 ilk byte LSB, ikinci byte'da MSB'dir. 8192'nin bir değeri
 merkez kayarlaması ve ses yüksekliği değişimi olmaması
 halidir. 8192'nin üstündeki değerler ses yüksekliğindeki bir
 damlayla belirlenir. 8192'den daha farklı bir değer, daha
 büyük bir ses yüksekliği değişimidir. el kontrol durumu
 belirler. 2-8.4.2. Kanal Mod Mesajları kontrolü kapalı 127 değeri
 yerel Altı kanal mod mesajı vardır. Tümü de bir kontrol
 değişim mesajı gibi aynı durum byte'ını kullanırlar. Bir
 kontrol değişim mesajının durum byte'ını ikinci veri byte'ı
 izler. İlk veri byte'ı işleme konan kontrol edicinin tipini
 belirler. Eger veri byte'ı 122'den 127'e altı değerden
 biriye, kontrol değişim mesajı, kanal mod mesajı olur.

Çünkü kanal mod mesajları gerçekte bir tip kontrol
 değişim mesajlarıdır. Bir kontrol değişim mesajıyla aynı
 formata sahiptirler. Bir durum byte'ı, onu izleyen ikinci veri
 byte'ı. Durum byte'ının D7'den D4'e kadar olan bitleri 1011
 ikilik sayısını belirler ve D3'den D0'a kadar olan bitler
 mesajın yollandığı kanalın açık olduğunu söyler. Birinci
 veri byte'ı kanal mod mesajı tipini belirler. (122'den 127'e
 bir sayıdır.) Ve ikinci byte da, eğer gerekliyse, ek bilgi
 taşır. Bir notayı çalarken bunu kapatmasını söyleyebilir.

Yerel : Kontrol : Yerel Kontrol mesajı bir

sentezleyicinin yerel Mes klavye aç kontrolünü; açıklar veya kapalılığını düzenler. Yerel aç kontrol Maçık olduğunda sentezleyici kendi klavyesinde çalınan notaya ve kendi kontrol panelinden girilen notaya karşılık verir. İlan kontrol değişim

Format : 1011nnnn01111010e0ccccccci01r
 nnnn (0-15) : Mesajın taşındığı kanalı belirler. ara, bu cih01111010e(01 Onluk 122'nine eşiti.)ye: Mesajın yerel kontrol mesajı olduğunu simgeler. un açık olmasını sorar. Ek olarak ccccccc cih(0-127an) : Yerel ay kontrolünün durumunu belirler. Bir -0- değeri yerel kontrolü kapar; 127 değeri yerel kontrolü açar. 1'den 126'ya kadar olan ana değerler önemsenmez. (0-15) : Mesajın taşındığı kanalı belirler.

Tüm Notalar Kapalı : Bu mesaj bir alıcının cihazda çalınan herhangi bir notanın kapanmasını sağlar.

Format : 1011nnnn01111011a00000000fan kontrol değişim mesajı nnnn (0-15) : Mesajın taşındığı kanalı belirler.

01111011d(onluk 123) : Mesajı bir Tüm Notalar Kapalı mesajı olarak belirler. asını sağlar. (Ve sonuç olarak çok

sesli 00000000a: sun iki kverime byte'ına sahip olan kontrol değişim mesajının tekil doluluk gereksinimidir. Bİ kanalı

aldığı Omni Mode Off : Omni Mode Off mesajı bir alıcı cihaza Omnipalıcının modunun kapalılığını sorar. la Omni modunun

kapalılığı ile alıcı cihaz yalnızca ayarlanmış kanallardaki mesajları alır. Omni mode off aynı zamanda bir alıcı cihaz

herhangi bir notayı çalarken bunu kapatmasını söyleyebilir. Format : 1011nnnn 01111100 00000000

nnnn (0-15) : Mesajın taşındığı kanalı belirler.
 01111100 (124) : Mesajın Omni Mode Off mesajı olduğunu belirler. Eğer cihaz MIDI kanal 3'e ayarlanmışsa ve 00000000 : iki veri byte'ına sahip olan kontrol değişim mesajının tekil doluluk gereksinimidir. Kanallarındaki mesajla Omni Mode On : Omni Mode On mesajı bir alıcı cihaza, bu cihaz, herhangi bir MIDI kanalından yollanan mesajları alabilse diye Omni alıcı modunun açık olmasını sorar. Ek olarak alıcı cihaza herhangi bir notayı çalarken kapanma durumunu sorar. alından aldığı notaları çok sesli olarak çalabilir. Format : 1011nnnn 01111101 00000000 hangi bir notanın kapanma nnnn (0-15) : Mesajın taşındığı kanalı belirler.

01111101 (125) : Mesajın Omni Mode On mesajı olduğunu gösterir. : Mesajın taşındığı kanalı belirler.

00000000 : İki veri byte'ına sahip olan kontrol değişim mesajının tekil doluluk gereksinimidir.

Mono Mode On : Mono Mode On mesajı bir alıcı cihazın mono alışı moduna geçmesini sağlar. (Veri sonuç olarak çok sesli -poly-3 modunun kapanmasını sağlar.) Mono Moduna açılması ile alıcı cihaz herhangi tek bir MIDI kanalından aldığı notaları tek sesli olarak çalar. Mesaj aynı zamanda çalınan herhangi bir notanın kapanmasını sağlar. n bitinden

Ve D3 Format : 1011nnnn 01111110 0000mmmm MIDI kanalını gösterir nnnn (0-15) : Mesajın taşındığı kanalı belirler. leri Sistem 01111110 (126) : Mesajın Mono Mode On mesajı olduğunu belirler. Durum Byte'inin ayarlandığı 1111 D7 den

D4'e $k = mmmmm(0-15)$: Bir alıcı cihazdaki MIDI kanallarının sayısını belirler. Bunlar ayarlanmış cihazın kanalları ile mesajı başlatır. Örneğin cihaz MIDI kanal 3'e ayarlanmışsa $over mmmm = 5$ ise, cihaz kanal 3'den 7'ye doğru mesajları alır. Eğer $over mmmm$ değeri 0 ise cihaz tüm MIDI kanallarındaki mesajları alır. Ayarlandığı kanaldan 16. kanala kadar.

Çok Sesli (Poly) Mode On: Bu mesaj bir alıcı cihazın çok sesli alıcı moduna geçmesine (Mono alıcı modun kapanmasına) yol açar. Çok sesli modun açılmasıyla cihaz tek bir MIDI kanalından aldığı notaları çok sesli olarak çalabilir. Aynı zamanda çalmakta olan herhangi bir notanın kapanmasını sağlayabilir.

Format: 1011nnnn 01111111 00000000 $back$ başlangıç $nnnn(0-15)$: Mesajın taşındığı kanalı belirler. $back$ için 01111111 $n(1-127)$: Mesajın Poly Mode On mesajı olduğunu belirler. $back$ için 00000000 yo : İki veri byte'ına sahip olan kontrol değişim mesajının tekil doluluk gereksinimidir.

2-8.4.3. Sistem Ortak Mesajları (System Common Messages)
 Önceki iki bölüm kanal mesajlarını anlattı. Tümünün kullandığı bitler, kanal mesajının bir cinsi olarak belirlenen Durum Byte'ının D6'dan D4'e kadar olan bitleridir. Ve D3'den D0'a olan bitler yollandıkları MIDI kanalını gösterirler. Tüm sistem mesajları için Sistem Ortak Mesajları Sistem Gerçek Zaman ve Sistem Birime Özel mesajları kadar iyi tanımlanmışlardır. Durum Byte'ının ayarlandığı 1111 D7'den

D4'e kadar olan bitlerdir. Herhangi bir cihaz dört tane 1-1-1-1 başlayan bir Durum Byte'ı aldığında bunu bir kanal mesajı olarak değil, bir Sistem mesajı olarak yorumlar. Buna bağlı olarak D3'den D0'a kadar olan bitler yollanmış Sistem Mesajı tipleridir. **Format : 11110110 (Veri Byte'ı yoktur.)**

Sistem Ortak Mesajına eşlik eden veri byte'ları hiç bir kesintiye uğramazlar. Bir veri byte'ı, tiki, ya da hiç olamayabilir. olarak bu bir kontrol cihazında herhangi bir zamanda **Şarkı Pozisyon İşaretleyicisi** : Bu bir mesaj bir playback'ın başlangıç noktasını kaydolmuş parçaları belirler. Ara eşlik eden hiç bir veri byte'ının olmadığı tek bir

Format : 01111001010000000000000000000000 aralarına yerleşir **00000000000000000000000000000000** (0-16383) : Kaydolu sırasının başlangıcından vuruşların sayısının belirlenmesi ile playback için başlangıç noktasını belirler. Her bir vuruş altı MIDI darbesine (1 Clock) eşittir. Tiki veri byte'ı birleşerek bir yüksek rezolüsyon değerini taşırlar. İlk byte LSB, ikinci ise MSB'dir. **Format : 11110000 (Veri Byte'ı yoktur.)**

Şarkı Seçimi : Sıralayıcıdaki şarkı bankasından bir şarkı seçer. Alıcı akımın başlangıcında çalmaya başlanmalarını söyler. **Format : 11110011 0sssssss**

sssssss : 1 (0-127) : Seçmek istediğiniz şarkının numarasını belirler. **Alıcı tüm alıcı sıralayıcılara çalarken Akord İstemi (Tune Request) :** Bu mesaj tüm alıcı cihazların kendilerini akord etmelerini söyler.

Format : 0111101100 (Veri Byte'ı yoktur.) Alıcı çalmanın

son du EOX (End of Exclusive) : EOX mesajı sistem birime özel mesajlarının sonunu işaret eder. Alıcı cihaz sistem birime özel veri başlangıcını tüm byte'lara önememesini söyler. 2-8.4.5. Sistem Birime Özel (Exclusive) Mesajları

Format : 11110110 (Veri Byte'ı yoktur.) ine has bir grupta 2-8.4.4. Sistem Gerçek Zaman Mesajı girerler. Her sistem Sisteme gerçek zaman mesajı MIDI sistemindeki zamanı korur. Sonuç olarak bu bir kontrol cihazında herhangi bir zamanda bilgi yollama yeteneği açısından önemlidir. Mesaj ile işaretli MIDI, sistem gerçek zaman mesajlarının diğer mesajlar ve bunlara eşlik eden hiç bir veri byte'ının olmadığı tek bir 011 durum byte'ı ile diğer mesajların aralarına yerleştirilmelerine izin verir. D7'den D4'e kadar olan bitler daima 1111 olarak ayarlanır ve D3'den D0'a olan bitler özel cins mesajları için ayrılır. 2-8.4.5.1. Zamanlama Darbesi (Timing Clock) : Bir zamanlama darbesi mesajı akıntısı alıcı sıralayıcı için tempoyu ayarlar. Format : 11110000 (Veri Byte'ı yoktur.) 2-8.4.5.2. Başla : Başla mesajı tüm alıcı sıralayıcılara yüklenmiş sıralı akımın başlangıcında çalmaya başlamamalarını söyler. Mesaj ile bitmesini ifade eder.

Format : 11111010 (Veri Byte'ı yoktur.) 2-8.4.5.3. Dur : Dur mesajı tüm alıcı sıralayıcılara çalmakta

olan sıranın durmasını söyler.

Format : 11111100 (Veri Byte'ı yoktur.)

Devam : Devam mesajı tüm alıcı sıralayıcılara çalmanın

son durduğu andaki noktalarından yüklenmiş sıranın çalmaya başlamasını söyler.

Format : 11111111 (256) (Veri Byte'ı yoktur.)

2-8.4.5. Sistem Birime Özel (Exclusive) Mesajları

Sistem birime özel mesajları kendilerine has bir gruptaki mesajlardır. Diğer hiç bir gruba girmezler. Her sistem birime özel mesajı iki eşlik edici veri byte'ı ve gerektiği kadar fazladan veri byte'ının minimumuna sahiptir. Her sistem birime özel mesajının sonu bir EOX mesajı ile işaretlenir.

Format : 11110000 Oiiiiiii Oxxxxxxx (Oxxxxxxx...)

11110111

Oiiiiiii (0-127) : Bir sistem birime özel mesajı yollamak istediğiniz cihazın yapımcısına karşılık gelen bir tanımlama numarasıdır. Bu numaralar MIDI Yapımcıları Birliği ve MIDI'nin ayrıntılı özellikleri tarafından verilir. Bazı büyük üreticilerin tanımlayıcı numaraları aşağıdadır.

Oxxxxxxx (0-127) : Üreticinin istediği herhangi birşeyi tanımlayabilir.

11110111 : Herhangi bir sistem birime özel mesajının EOX mesajı ile bitmesini ifade eder.

Bazı büyük MIDI cihazı üreticilerinin numaraları :

- 1 Sequential Circuits
- 7 Kurzweil
- 15 Ensoniq
- 16 Oberheim

64 Kawai

65 Roland

66 Korg

67 Yamaha

68 Casio

69 Akai

UCUNCU BÖLÜM

BİLGİSAYARLAR VE MIDI

3- BİLGİSAYARLAR VE MIDI

Bilgisayarlar MIDI mesajlarıyla çalışmada çok üstün bir performansa sahiptirler. Hatta MIDI mesajlarını kullanma konusunda kendilerine has dizaynları sayesinde bir kablo üzerinden mesajları seri olarak yollama yeteneğine sahiptirler. Bu da MIDI sisteminizin gücünü arttırır. Bilgisayarlar kullandıkları yazılımlara bağlı olarak fonksiyonlarını değiştirebilir ve genişletebilirler. Aynı şekilde pek çok MIDI sistemine bağlanabilme konusunda da yeteneklerini arttırabilirler.

Bu bölümde kişisel bilgisayarların bir MIDI sisteminde nasıl kullanılabileceği anlatılacaktır. Kişisel

ÜÇÜNCÜ BÖLÜM

Bilgisayarların MIDI cihazlarına nasıl bağlanacağı, ve neler yapabileceği gösterilmeye çalışılacaktır.

BİLGİSAYARLAR VE MIDI

3-1. MIDI ARABİRİMİ

Doğru bir şekilde bilgisayar kullanımı, ancak doğru bir program ile ve bilgisayarınızı MIDI sisteminize doğru olarak taktığınızda işe yarar. Yalnız bir sorun vardır : Doğru bilgisayar bir MIDI sisteminin sahip olduğu donanımın sahip olmadığı için, eksiklerini bir MIDI portu ile kapatmalıdır. Oyse, zaten çoğu bilgisayarda seri port vardır ve bir MIDI arabirimi ekleyerek bunu MIDI portlarına çevirebilirsiniz.

MIDI arabirimi bilgisayarın arkasındaki seri porta takılan küçük bir parçadır. Bu en azından bir MIDI in bir ve bir MIDI out portu içerir. Elektronik devreler seri port yoluyla gelen sinyali alıp uygun gerilimlerle MIDI kablolarına

veririz. 3- BİLGİSAYARLAR VE MIDI MIDI kablosundan bilgisayara (seri) Bilgisayarlar MIDI mesajlarıyla çalışmada çok üstün bir performansa sahiptirler. Hatta MIDI mesajlarını kullanma konusunda kendilerine has dizaynları sayesinde bir kablo üzerinden Mesajları seri olarak yollama yeteneğine sahiptirler. Bu da MIDI sisteminizin gücünü artırır. Bilgisayarları kullandıkları yazılımlara bağlı olarak fonksiyonlarını değiştirebilir ve genişletebilirler. Aynı şekilde pek çok MIDI sistemine bağlanabilme konusunda da yeteneklerini arttırabilirler.

Bu bölümde kişisel bilgisayarların bir MIDI sisteminde nasıl kullanılabileceği anlatılacaktır. Kişisel bilgisayarların MIDI cihazlarına nasıl bağlanacağı, ve neler yapabileceği gösterilmeye çalışılacaktır.

3-1. MIDI ARABİRİMİ Doğru bir şekilde bilgisayar kullanımı, ancak doğru bir program ile ve bilgisayarınızı MIDI sisteminize doğru olarak taktığınızda işe yarar. Yalnız bir sorun vardır : Çoğu bilgisayar bir MIDI sisteminin sahip olduğu donanımına sahip olmadığı için, eksiklerini bir MIDI portu ile kapatmalıdır. Oysa, zaten çoğu bilgisayarda seri port vardır ve bir MIDI arabirimi ekleyerek bunu MIDI portlarına çevirebiliriz.

MIDI arabirimi bilgisayarın arkasındaki seri porta takılan küçük bir parçadır. Bu en azından bir MIDI in bir de MIDI out portu içerir. Elektronik devreler seri port yoluyla gelen sinyali alıp uygun gerilimlerle MIDI kablosuna

verirler. Aynı şekilde bu işlemi MIDI kablосundan bilgisayara (seri porta) olacak şekilde de yapabilirler. **İlayabilir.** **Geri kalan** Çok çeşitli MIDI arabirimleri vardır. Her biri özel bir tip bilgisayar için dizayn edilmiş olup, diğеrinde çalışmaz. Çoğudа bilgisayarda tek bir MIDI portu varken, bazılarında iki MIDI'yi, ve aynı zamanda altı sekiz arası MIDI out portu vardır. Bazıları ise bir MIDI thru portu bile içerebilirler. MIDI arabiriminin en önemli gereksinimi ; MIDI portlarının bilgisayarınıza ve MIDI sisteminizdeki diğеr cihazlara uyabilmesidir. **halıdır.** **Ayrıca Smart arabirimi**

İşlemci 3-1.1. SMART VE DUMB MIDI ARABİRLERİ: Arabirimleri iki büyük grupta toplayabiliriz: Smart ve Dumb arabirimleri. Smart MIDI arabirimi bilgisayarın kendisi tarafından düzenlenir. Kendi ROM katında kendi belleği ve yazılımı vardır ve bilgisayarın seri portundan aldığı komutlarla çalışır. Smart arabirimi MIDI portundan aldığı mesajları ayırıp, bunlar hakkındaki bilgileri bilgisayara yollayabilir.

gelme Dumb MIDI arabirimi mikroişlemci içermez. Kendi kendine **3-2. MIDI Mesajı** alamaz, yollayamaz. Yalnızca bilgisayardan gelen sinyali MIDI kablосuna uyumlu ve kablodan gelen mesajı da bilgisayara uyumlu hale çevirir. Kişisel bilgisayarlarda bu mesajları anlayacak mikroişlemciler mevcuttur. **kayıt ve performans** **uygulama** **MIDI**

cihaz Smart MIDI arabiriminin asıl avantajı bilgisayarın kontrolüne fazla girmeden çalışabilmesidir. Örneğin bir Smart MIDI arabirimi bilgisayarın bağlı olduğu bir elektronik

bateri veya sıralayıcıdan MIDI out portuna gelen metronom sinyalinin düzgün MIDI vuruşları olarak yollayabilir. Geri kalan işi bilgisayarın mesajını alana kadar arabirim yapar. Aynı metronomu Dumb arabirimi ile elde etmek için bilgisayarın kendi MIDI vuruşunu, seri porttaki arabirime yollamalıdır. Bilgisayarın kendi kaynaklarıyla MIDI vuruşları arasındaki gerekli tempoyu hesaplayarak ayarlamalıdır.

Smart arabirimi çok üstün olmasına karşın kendi işlemcisini ve elektronik devrelerini taşıdığı için Dumb arabiriminin göre çok pahalıdır. Ayrıca Smart arabirimdeki işlemci ve program yetrince güçlü bir bilgisayar kullandıktan sonra çok gereksizdir. Ancak bazı bilgisayarlar (örneğin IBM PC) smart arabirim olmadan MIDI'ye takılamaz. Bu için gereklidir. Ayrıca bilgisayarın uygulamaları yazılımı smart arabiriminle uyumlayabilir. Çünkü her Smart arabirimi kendine has değişik tipte komutlara sahiptir. Bu dezavantajlar sonucunda Dumb arabirimi tercih edilebilir durumda gelmektedir. Mesajların Kaydetme ve Çoğaltma MIDI

3-2. MIDI YAZILIMI

Her ne kadar MIDI uygulama programları tamamen farklı ve sınıflandırılması çok çeşitli olsa da dört ana grupta incelenebilir: seri notayı bir adım zamanda kaydediyorsa,

1- Müzikal kayıt ve performans uygulamaları; bunlar MIDI cihazlarından bilgisayara gelen mesajları çağırmanızı ve çalmanızı sağlar. Zaman adımı girilen notaların birbirinin

2- Müzikal notasyon ve basım uygulamaları; notaları

geleneksel yöntemle yazmanızı sağlar. Bir stüdyodaki teyp
Sentezleyici kısım editörleri ve kütüphaneleri; farklı
sentezleyici kısımlarının bilgilerini bilgisayar belleğinde
veri olarak saklayabilir. Her ise farklı bir enstrümanın
Müzik eğitimi uygulamaları; bilgisayar ekranında ve
klavyesinde ve MIDI portuna bağlı diğer kısımlarda farklı
etkiler elde etmeye yarar. di yeniden kaydetmeksiniz, geri

3-2.1. Müzik Kayıt ve Performans ile ilgili Kaydetmek,
düzeltmek ve geri çalmak bilgisayarlar için kullanılan MIDI
yazılımlarının en bilindik kısımlarıdır. Bilgisayarlar bu
programlarla bir stüdyodaki kayıt ünitelerinden çok
dahakullanışlı olabilirler. Kayıt programları tam bir kayıt
stüdyosu gibidir ve bu işleme MIDI kayıt adı verilir. Sık sık
ekran gösteriminde play (başla), rewind (geri sarma), record
(kayıt), stop (dur) ve MIDI vuruş saatinin iz sayıcıları
bulunur. Ancak tam bir dek teyp gibi olmadığının için bunlarda
klavyeden belli şeyler girmedikçe MIDI mesajları kaydolamaz.

MIDI Mesajlarını Kaydetme : Çoğu MIDI
kaydedicileri müziği çalındığı anda (zaman adımında)
kaydederler. Kaydedici bir adım zamanında MIDI mesajını
aldığında onu mesajlar arasındaki zamana sokmadan kayıt eder.
Örneğin bir seri notayı bir adım zamanda kaydediyorsa, ilk
notayı çalar ve bilgisayar klavyesinden notanın uzunluğunu
girer. Bu durumda yeni bir nota ve bunun uzunluğu
girilmelidir. Bir zaman adımında girilen notaların herbirinin
uzunluğu tanımlanır. u mesajların her biri bir perdenen kayda

Başlatma Kaydedici de izlerir işi, herhangi : Bir stüdyodaki di teyp genellikle çok kanallı bir kaydedicidir. Basit bir banttaki çoğu izi kaydetme imkanı vardır. Her iz farklı bir ses sinyali depolayabilir. Her iz farklı bir enstrümanın kaydedilmesiyle mühendis diğer izlerde çalan enstrümanlara el sürmeden istediğini sesini, ritmini ve diğer değerlerini değiştirebilir. Tüm kaydı yeniden kaydetmek sizin, geri dinlemede farklı izlere değişik sesler verilebilir. Kaydedici özelliği MIDI kayıtları aynı zamanda kayıtlar için yeni izler de sunar. Her iz, bilgisayarın belleğindeki farklı bir kısımdaki MIDI mesajlarının bir akımı olarak kaydedilir. MIDI kaydedici kaydı geriye aldığı anda, tümü aynı anda çalınır diye bütün izler başlatılır. MIDI kaydedicideki her izi kaydettiğimiz ve ayarladığımız kanallar, MIDI sistemimizdeki her enstrümana ayrı bir mesaj yollar. Böylece her izde farklı bir enstrüman çalınabilir. düzeltmeleri tamamlanabilir. Bir kayıt Eger tüm enstrümanların kullanıldığı bir veya birkaç kaydedilecekse, bu bir seferde yapılır. Üstüste kayıt yöntemi kullanılır. Önce bir iz bir enstrüman kaydedilir, kayıt bittiğinde başka bir enstrüman başka bir izi, ancak bu sefer ilkiyle senkronize edilerek işlem yapılır. Daha sonrakilerde de aynı yöntem uygulanır. Olaneksizliği doğurur.

Vuru Girişi ve Vuru Çıkışı (punch in-punch out) : Bir izin bir kısmını kolayca yeniden kaydetmeyi yarayan diğer bir MIDI kayıt özelliğidir. Önceden MIDI mesajlarıyla doldurulmuş bir izi bu mesajların herhangi bir yerinden kayda

başlatmak için delme girişi, herhangi bir noktada durdurmak için delme çıkışı kullanılır. Bu özellik gerçek zaman performansındaki hataları düzeltmek için çok yararlıdır. Çoğu MIDI kaydedicileri geri çalışmadan önce izlerdeki delme girişi ve delme çıkışı noktalarını ayarlamamıza izin verir. Kaydedici delme girişi ve delme çıkışı mesajlarını doğru zamanda sizin hatalarınıza karşı otomatik olarak yolları dönüştürebilir ve **Eliminasyon Mesajı Filtresi** : Diğer bir kullanışlı MIDI kaydedici özelliği de mesaj filtresidir. Mesajı filtresine bir izde kaydettiğiniz belli tip mesajları atlaması için ayarlayabilirsiniz. **Quantization** kaydettiğiniz mesajların zamanı Mesajı filtreleri süreye kaydettiğiniz kadarını yuvarlatmak açısından yararlıdır. bir düzeltme özelliğidir. **Quantization** cihazı MIDI Kayıtlarını Düzeltme : MIDI kaydedicisinin bir teyp kaydedicisine yansıyan büyük üstünlüğü, kaydettiğiniz mesajlardaki düzeltmeleri tamamiyle yapabilemenizdir. Bir kayıt mühendisi teyp kaydedicisinde görece genliğe ve bir kayıttaki her bir dinamiğin tonunu ayarlamak için kayıttaki sesleri değiştirir. Mühendis iyi performans elde etmek için kötü kısımları kesip, diğerlerine ekleyerek çalışır. Fakat bu da teyp kaydındaki bir enstrümanın tek bir notasının kalitesini değiştirememesi olanaksızlığını doğurur. **Birime Özel MIDI Mesajlarının Değiştirilmesi** : Çoğu MIDI kaydedicisi belli bir izdeki tüm kaydedilmiş mesajları gösterir. Herhangi bir mesajı seçebilir ve taşıdığı veriyi değiştirebilirsiniz. geniş alanlarda çalışmanız için verir.

Kesme MIDI Mesajlarını Silme ve Dönüştürme : İyi bir MIDI kaydedicisi sadece yalnızca mesajları değiştirmenize değil, aynı zamanda bir MIDI esaltı mesajları eklemenize de izin verir. İstediginiz, yerlerdeki efektleri bir ve eklenotaları değiştirebilirsiniz. Ayrıca kayıt sırasında hatalı çaldığınız notalarınızı da değiştirebilirsiniz. Aslında tüm bir şarkıyı oluşturmaya başlarken birime özel notaları dönüştürebilir ve silebilirsiniz. (Çünkü Tek bir nota bile yazmadan.) Bu işlem zaman alıcı ve can sıkıcıdır ama şarkının her kısmında dakika dakika kontrolü sağlar. **nokta seçer ve MIDI kaydedicisine**

Quantization : Quantization kaydettiğiniz mesajların zamanının doğruluğu konusunda yardımcı olan ve çoğu MIDI kaydedicide bulunan bir düzeltim özelliğidir. Quantization çizgileri zaman boşlukları, aradaki mesafelerdeki noktaları üstüne yerleştirir. Bu özelliği performans kaydındaki pürüzlü ritimleri düzeltmekte kullanabiliriz. **Parça Quantization düzensiz ritimler içinde mucize çözüm**

degildir. Bir MIDI kaydedicisi men quantization değerinin çokluğuna, notaların başlangıcından giderse, eğer ritm çok düzensiz çalışıyorsa notaları Nyanlış düzelttuda Mida yollayabilir. Quantization aynı zamanda her vuruştan önce ya da sonra keskin vuruşların yarattığı ritmik farklılıkları da silebilir. **siniz. Çogu kaydediciler size standart bir dok**

Kesme ve Ekleme : Tüm düzeltmelerin küçük detayları yoktur. Kesme ve Ekleme gibi bazıları bir kaydın yapısını değiştirmeden çok geniş alanlarda çalışmanıza izin verir.

Kesme, basit bir işlemdir. Bir izin, bir kısmını kesmek istediğinizde, kesmenin başlangıcındaki MIDI mesajını ve sonundaki MIDI mesajını tanımlarsınız. Kaydedici parçayı izden keser, ve kalan iki kısmı birbirine ekler. MIDI kaydedicisi bir izin bir parçasını kestiginizde; bu parçayı daha sonra tekrar çağırabilmeniz için, bunu bilgisayarın belleğinde tampon adı verilen bölüme depolar. Tampon yalnızca son kestiginiz parçayı depolar. Ötadan bağlamamızı sağlayan sayıcı Tamponda bir parçanın üstüne ekleme yapmak için izdeki iki mesaj arasında bir nokta seçer ve MIDI kaydedicisine üstüne ekleme yapmasını söyler. Kaydedici, tampondaki parçanın bir kopyasını çıkartır ve izdeki iki mesaj arasında yerleştirir. Birine Özel İz kontrolü basitçe düzeltilmiş bir şarkı Kesme, kendi kendine, e izin dışındaki edit etmek istediğimiz tüm müzik blogunu içinden atabilir. Kesme ve üstüne ekleme izdeki bir kısımdan diğer kısma, kesilmiş müzik parçalarını taşınmasına izin verir. Aynı zamanda kesme üstüne yerleştirmeyi, san müziginin bölümlerini çoğaltmakta da kullanabilirsiniz.

Kayıtları Yeniden Çalma : Nihayet bir parçayı MIDI kaydedicisine kaydettiniz ve düzelttiniz. Bunu yeniden çalmak istediğinizde, istediğinizlikte ses MIDI değerlerini değiştirebilirsiniz. Çogu kaydediciler size standart bir dek teypteki yeniden çalma kontrollerini aynen sağlar. Play, kaydı çalmaya başlatmak için; fast forward, belli bir hızla çalınan parçayı daha hızlı çaldırmak için; rewind, aynı şeyi

geriye doğru yaptırmak ve stop, yeniden çalınan parçayı durdurmak içindir. Çalmanın temposunu değiştirme özelliği

çok Standartdır. Bir MIDI kaydedicisinde işleme başladığımızdan bu yana ne kadar kısım geçtiğimizi söyleyen bir sayıcı (counter) bulunur. Ya MIDI vuruşlarının sayısını

ya da parçanın başlangıcından bu yana eşit çeyrek notalara kaydedicisi tempoyu değiştirdiğimizde orijinal ses yüksekliği korunur. ayarladığımız vuruları gösterir. Bazı kaydedicilerde parçanın yeniden çalınmasına özel bir noktadan başlamamızı sağlayan kaydı seri bir tempoyla yeniden çalabilirsiniz. Tempo sayıcı ayarlamaları mevcuttur.

değişiklikleri belli bir amaç için yapılan kayıtlarda son Birime Özel İz Kontrolü : Eger pekçok farklı izde

derede kullanılırdır. müzik kaydettiyseniz, MIDI kaydedicisi tüm izleri aynı anda

Nota Çevrimi (Transposition) : Çogu MIDI çalabilir. Tek tek ya da istediğimiz formasyonda da

kaydedicileri bu özelliklerle parçayı yeniden çaldığınızda çalabilir. Birime Özel İz kontrolü basitçe düzeltilmiş bir

kaydedilmiş müziğin anahtarını değiştirebilirler. Bir parçayı şarkı parçasını dinlememize veya bir izdeki kısmını

Re anahtarından kaydedip, kaydediciden Do anahtarında dinlememize izin verir. Böylece izin üçlükleri ya da

çalmasını isteyebilirsiniz. Kaydedici bunu üzerine parçadaki her ikilikleri arasındaki senkronizasyonu kontrol edebilirsiniz.

notayı yeniden akord edilebilir. Bu parçadaki yeni bir enstrümanın üzerine yeni bir iz kayıt

kaydır. etmek istiyorsanız, diğerini yeniden çalarken sonrakinde

MIDI Eşliği : Bir MIDI kaydını canlı bir soliste eşlik kayıt etmeyi seçebilirsiniz.

ederek eşlikte çaldırarak çok kolaydır. Zor olan solistin

Tempo Değişiklikleri : MIDI kaydedicisindeki bir kaydı kaydı eşliğine oynasındır. Bir eşlik edici solistin

yeniden çaldığınızda kaydedici genellikle bağlandığı MIDI söyledigini veya çaldığını dinleyecek ve solist hızlandığında

cihazına yolladığı mesajlarla birlikte bir MIDI vuruş mesajı veya yavaşladığında Etkileme mesajıyla buna uygun olarak

da yollar. Bu MIDI vuruşu diğer MIDI cihazlarına hızını artırıp, azaltır.

sıralayıcılar ile (Örneğin elektronik bateri.) çaldıkları Standart MIDI kaydedicisi solist tarafından yapılan

MIDI kaydının ne kadar hızlı olması gerektiğini söyler. Eger tempo farklılıklarını sorgusuz çalar. Birkaç MIDI kaydedicisi

MIDI kaydındaki tempoyu değiştirmek isterseniz, diğer ve diğer özel MIDI programları canlı eşlik ediciyi

cihazlar da buna bağılı olarak tempolarını deęiřtirirler.

Yeniden geri almanın temposunu deęiřtirme zelligi temposunu deęiřtirebilir. Eřlik edicinin farklı ızları çok kullanıřlıdır. Bir MIDI kaydedicisinde tempoyu solo paraları, tüm eřlik edicilerin ilk ızalarında hızlandırdığımızda ya da yavaşlattığımızda kaydedilmiş gerekebilir ve kaydedilir. Eđer solist bir vokalistse veya müziğin ses yükseklięinin daha tiz ve daha bas ıkması gerekmez. M akustik enstrüman kullanan birisi ise performansı MIDI

kaydedicisi tempoyu deęiřtirdiğimizde orjinal ses yükseklięi korur.

track Küçük bir kısmı yavaş bir tempoyla kaydedilmiş bir kaydı seri bir tempoyla yeniden alabilirsiniz. Tempo deęiřiklikleri belli bir amaç için yapılan kayıtlarda son derece kullanıřlıdır ve eřlik ediciye doęru notalar gelene

kadar Nota. Çevrimi (Transposition) : Çoęu MIDI kaydedicileri bu zellikle parayı yeniden aldığımızda kaydedilmiş müziğin anahtarını deęiřtirebilirler. Bir parayı Rep anahtarından kaydedip, kaydediciden Do anahtarında almasını isteyebiliriz. Kaydedici bunu üzerine paradaki her notayı yeniden akord edip iki yarım adım bas yönünde kaydırır. geri alma MIDI müziğinin tesine taşıyan

program MIDI Eřligi : Bir MIDI kaydını canlı bir soliste eřlik edecek şekilde aldırmak çok kolaydır. Zor olan solistin kaydın eřligine uymasıdır. Bir eřlik edici solistin söylediğini veya aldığını dinleyecek ve solist hızlandığında veya yavaşladığında (Etkileme amacıyla) buna uygun olarak hızını arttırıp, azaltır. mařık bir matematik ile kanuoyunda

daha Standart MIDI kaydedicisi solist tarafından yapılan tempo farklılıklarını sorgusuz alar. Birkaç MIDI kaydedicisi ve dięer özel MIDI programları canlı eřlik ediciyi

dinleyebilir ve onun performansına uyacak şekilde çalma
 temposunu değiştirebilir. Eşlik edicinin farklı izlerdeki
 solo parçaları tüm eşlik edicilerin ilk kısımlarında
 gerekebilir ve kaydedilir. Eger solist bir vokalistse veya
 akustik enstrüman kullanan birisi ise performansı MIDI
 mesajına çevirmek için ses yüksekliği izleyici (pitch
 tracker) kullanılır. Basit değildir ve bazı sorunlar
 vardır. MIDI programı solo izdeki performansa göre solisti
 izler. Eger solist solo izindeki notaları çalmıyor ise
 program çalmayı keser ve eşlik ediciye doğru notalar gelene
 kadar bekler. Müziği göstermek için bilgisayar ortamında
 kullanılır. Tüm bu özelliklere rağmen bir bilgisayar bir insan
 eşlik ediciye yaklaşamaz. Hatalar için gerekli düzeltmeleri
 yapamaz ve solistin kaydedilmiş solo çizgisinin dışına
 çıkmasına izin vermez. (Eğer özellikle belirtilmediyse.) MIDI
 sistem Bilgisayarla üretilmiş Müzik programları Kaydedicinin
 başarısını, geri çalmak MIDI müziginin ötesine taşıyan
 programlar vardır. Aslında bunlar müzik yazmak ve onları
 çalmak üzere bağlı bulundukları MIDI cihazlarına yollayan
 aletlerdir. Programlarının çoğu süreklilik, genlik, ses rengi
 ve ses yüksekliği üretecek matematiksel formüllere bağlı
 olarak kullanılır. Müzik üreten programlar çok yorumcul bir
 müzik yapamazlar. Ama karmaşık bir matematik ile kamuoyunda
 daha kabul edilebilir müzikler üretebilirler.

3-2.2. Müzikal Notasyon ve Basım : Müzikal notasyon
 Destek programları müzikal notasyon ve basımını üç

alanda çalıştıracak şekilde dizayn edilirler.

--- Müzikal notasyonu MIDI mesajlarına çevirmek, program-- MIDI kaydedicisindeki mesajların sıralılarını yazılmış müzige çevirmek, -- Note Off mesajı ile her bir -- Bir yazıcıda müziği basmak. -- Müzikal MIDI notasyon programları -- MIDI kaydedici programları. kadar basit değildir ve özelliklerinin ayarlanması konusunda bir standartları yoktur. Her biri müzikal notasyon ve bakımında kendi yöntemlerini kullanırlar.

uzunluk Müziği Girme : Çoğu müzikal notasyon programı girdiğiniz müziği göstermek için bilgisayar monitörünü kullanır. Program ekranda nota çizgilerini çizer ve buradan notalar girilir. küçük değişiklikler yapılabilir. Bunların

çoğuna Bir Performansı Yazılmış Müzige Çevirmek : Bir müzikal notasyon programında bir parçayı girdinizmi, artık bunu MIDI sisteminizde çaldirabilirsiniz. Bazı programlar bilgisayarın kendi ses yeteneğini kullanırlar ve girdiğimiz müziği kendileri çalabilirler. Diğerleri ise müziği MIDI mesajları haline çevirip harici MIDI cihazlarına yollarlar. ya da onları

Yazılmış Müziği Bir Performansa Çevirmek : Müzisyenlerin sürekli düşüneyazdıkları bir parçanın performansa çevrilmesidir. Bu düş bilgisayarın parçayı alıp anlayacağı dile çevirmesiyle gerçekleşir. Çoğu bilgisayar kullanıcısı bir performansı kaydedip bunu diske atmak için bir MIDI kaydedicisi programı kullanır. Müzikal notasyon programlarında yazılmış müziği kaydedilmiş MIDI mesajları

olarak yorumların kendi eklenmesi. MIDI Mesajlarının Yorumlama : Müzikal notasyon programlarında kaydedilmiş MIDI notasının Note On mesajının numarasını alır ve Note On ile Note Off mesajları arasındaki herbir notanın MIDI vuruş numarasını, uzunluğunu ve kaydedilmiş MIDI notalarının ses yüksekliğini okuyarak söyleyebilir. Bunlar müzik portresindeki dikey pozisyonda ses yüksekliği değerini yorumlayabilir ve MIDI vuruşunun, vuruş-nota eşliğinin doğru nota genişliği değerini notanın uzunluğuna çevirebilir. Ya da yeni bir vuruşunkine de çevirebilir. Performansın notaları doğru ses yüksekliğinde ve doğru büyüklükte; portrede bunlar üzerinde anahtar ve zaman belirticisi gibi küçük değişiklikler yapılabilir. Bunların çoğunu bilgisayarlar yapar.

3-2.3. Sentezleyici Ekleme Editörleri ve Kaynak Kütüphanelerinde gösterimini sağlayan bilgisayar ekranı gibi bir Çoğu sentezleyici olan yapmakta olduğunuz eklemeleri gösteren bir ekrana sahiptir. Aynı zamanda eklemeleri depolamak için kısıtlı bir bellekleri vardır. On ya da onaltı yeni ekleme yaptıktan sonra bu alan tükenir. Yine de daha büyük ayarlamaları disk ile yapabiliriz. Ekleme Kaynak Kütüphaneleri : Ekleme Kaynak Kütüphaneleri bilgisayarınızda kullanabileceğiniz en basit ekleme programıdır. Mesajları bağlı olduğu sentezleyiciden özel sistem mesajı olarak alır ve onu hafıza ve diskte saklar. Buna gereksindığınız zaman tekrar çağırabilirsiniz.

Her sentezleyicinin kendi ekleme ve yaratma yönteminin olmasının nedeni; sentezleyicide eklenmiş bir kaynak kütüphanesinde çalışıp çalışmadığını anlamak içindir. Eğer ikiside uyumsuz ise program sentezleyici eklemeleri alamayacaktır.

Bir ekleme bilgi kütüphanesi; eklemeleri tek başına veya toplu olarak alabilir. Bir eklemeler topluluğu sentezleyicideki ekleme butonunun devreye sokulmasıyla ayarlanır.

Ekleme Editörleri : Ekleme kaynak kütüphaneleri eklemeleri yalnızca alır, gönderir veya depo eder; üzerinde değişiklik yapamaz. Bu şans ekleme, bilgisayarın belleğine alındığında elde edilir. Her sentezleyici bunu kendi yöntemi ile yapar.

Büyük farklılıklara karşın çoğu editörler eklemenin doğru pozisyonda gösterimini sağlayan bilgisayar ekranı gibi bir özelliğe sahiptirler. İyi bir ekleme editörü eklemeleri doğrudan sentezleyiciye yollayabilir ve birkaç notayı çaldığınızda eklemelerin etkisini görmenizi sağlar. Hemen tüm ekleme editör programları ekleme kütüphanelerini içerir. Yarattığınız ve ayarladığınız yeni eklemeler diskte depolandıktan sonra istendiği zaman sentezleyiciye yollanabilir.

IBM PC İÇİN MIDI ARABİRİMİ

Özellik 4- IBM PC İÇİN MIDI ARABİRİMİ

Bu bölümde IBM PC için bir MIDI arabirimi yapılmaya çalışılmıştır. Tabii burada arabirimin elektronik yapısı ile ilgilenilmiş ve uygun bir program için gerekli malzeme çıkarılmıştır. (Register yapıları ve adresleri gibi.) Bu arabirim ancak bu malzemeye uygun bir program yazılması ile aktif olarak kullanılabilir. (Avrupa ve Amerika'da nadiren de olsa bu tür programlar yapılmıştır.) Bu arabirim bazı kısımları açısından Smart türü arabirime benzese de Dumb türü bir arabirimdir. MIDI konusu önceki bölümlerde yeterince açıklanmış olduğundan bu bölümde sadece IBM PC'nin yapısından ve bazı özelliklerinden bahsedilecektir. **Boşluk**

Parity 4-1. IBM PC DONANIMINA GENEL BİR BAKIŞ

IBM PC sistem kartı beş birimden meydana gelir. Bunlar merkezi işlem birimi, ROM (Yalnız okunur bellek), RAM (Rastgele okunur yazılır bellek), Birleşik I/O (Giriş / çıkış) adaptörleri ve I/O kanalıdır. **Parity**

Sistem kartının merkezi işlem birimi 8088 mikro işlemcisidir. Bu mikroişlemci 16 bitlik 8086 mikro işlemcisinin 8 bit veri yolu bulunan bir düzenlemesi olup 8086 ile yazılım uyumludur. 8088 çarpma ve bölmeyi içeren 16 bitlik işlem ve 20 bitlik adresleme (1 Mbyte) yeteneğine sahiptir. Merkezi işlem biriminin çalışma frekansı 4.77 MHz dir.

İşlem birimi 20 bitlik doğrudan hafıza erişimi (DMA) olan 4 kanal, 16 bitlik sayma zamanlama kanalları ve 8 adet

öncelikli kesinti seviyesi içerir.

Dört adet DMA kanalının üç tanesi I/O yolları üzerinde olup, işlem ünitesinin araya girmesine gerek kalmadan I/O cihazları ile hafıza ünitesi arasında yüksek hızda veri iletimini sağlar. Dördüncü DMA kanalı ise dinamik bellekleri tazelemek içindir.

4-1.1. 8088 DONANIM KESİNTİ İSTEĞİ

PC de kullanılan sekiz adet öncelikli seviyesinin altısı, dış fonksiyon kartlarının bağlandığı sistem genişletme yolları üzerinde kullanılır. Diğer ikisi ise sistem kartı üzerinde kullanılır.

Maskelenemeyen kesinti (NMI) ise bellekteki boşluk (Parity) hatalarını bilgirmede kullanılır.

8088 donanım kesinti listesi aşağıdaki gibidir:

NUMARA	KULLANIM
NMI	Boşluk (parity)
0	Zamanlama
1	Tuş takımı
2	Sisteme ait
3	Asenkron haberleşme (ikincil - com2) SDLC haberleşme ikili tabanda senkron haberleşme (ikincil)
4	Asenkron haberleşme (birincil - com1) SDLC haberleşme ikili tabanda senkron haberleşme (birincil)

5	Sabit disk ünitesi
6	Disket ünitesi
7	Yazıcı

4-1.3. I/O KANALI VE I/O HARİTASI

I/O kanalı 8088 Mikroişlemci veri yolunun (DATA BUS) genişleme şeklidir. Güç kaynağı destekli, DMA ve çaşıtlı kesinti fonksiyonları ile geliştirilmiş bir yapıdır. I/O cihazları, I/O bellek haritasındaki adres alanları kullanılarak adreslenir.

I/O Kanal kartları için 512 adet I/O cihaz adresi vardır.

I/O Adres haritası aşağıda olduğu gibidir:

16'lı Adres alanı	Kullanım yeri
000 - 00F	8237a-5 DMA devresi için
020 - 021	8239a Kesinti devresi için
040 - 043	8253-5 Zamanlayıcı için
060 - 063	8255 Çevre arabirimi
080 - 083	DMA sayfa kaydediciler
0AX	NMI Maskeleye kaydedicisi
0CX	Sistem için ayrılmıştır
0EX	Sistem için ayrılmıştır
100 - 1FF	Kullanılamaz
200 - 20F	Oyun kontrol adaptörü
210 - 217	Sistem genişletme
220 - 24F	Sistem için ayrılmıştır

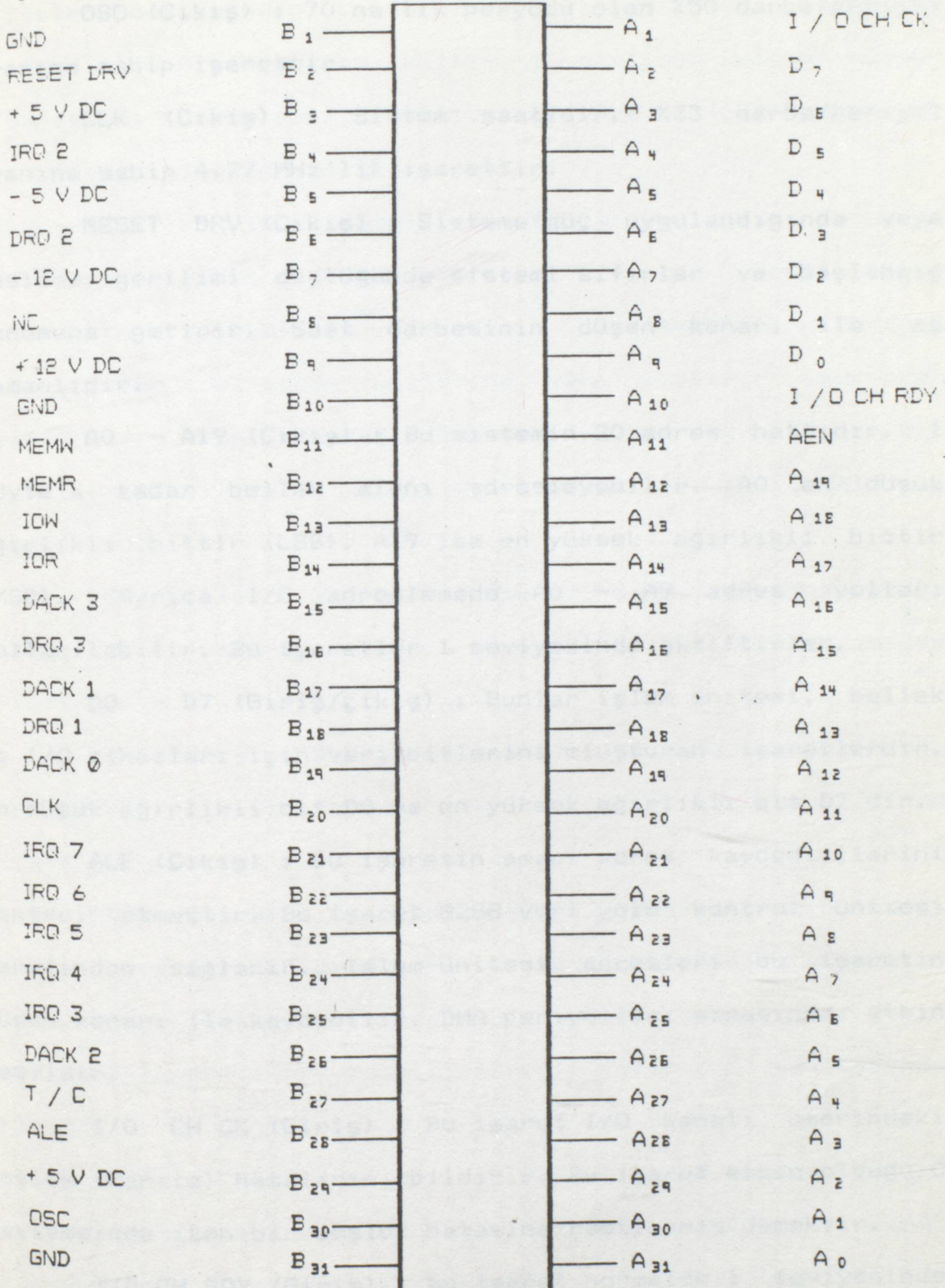
278 - 27F	Sistem için ayrılmıştır
2F0 - 2F7	Sistem için ayrılmıştır
2F8 - 2FF	Asenkron haberleşme (ikincil)
300 - 31F	Prototip kartı
320 - 32F	Sabit disk ünitesi
378 - 37F	Yazıcı
380 - 38C	SDLC Haberleşme
38D - 398	İkili tabanda haberleşme
3A0 - 3A9	İkili tabanda senkron haberleşme
3B0 - 3BF	Tek renkli ekran
3C0 - 3CF	Sistem için ayrılmıştır
3D0 - 3DF	Renkli ekran / grafik
3E0 - 3E7	Sistem için ayrılmıştır
3F0 - 3F7	Disket
3F8 - 3FF	Asenkron haberleşme (birincil)

I/O KANALININ ÖZELLİKLERİ

- 8 bitlik çift yönlü veri yolu
- 20 adres hattı
- 6 kesinti seviyesi
- I/O okuma/yazma hızı ve bellek için kontrol hatları
- Bellek tazelemesi için zamanlama kontrol hatları
- Adaptörler için güç ve toprak bağlantıları

I/O kanal bağlantısı 64 hatlı konektör aracılığı ile yapılmaktadır. Bu konektörün A yüzü Adres ve veri yollarını,

B yüzü ise kontrol işaretlerini kartlara taşır.



Şekil 14. I/O Kanalının şeması

4-1.4. I/O KANALININ TANIMLANMASI

Unitel OSC (Çıkış) : 70 ns'lik periyodu olan %50 darbe/periyot oranına sahip işarettir.

I/O CLK (Çıkış) : Sistem saatidir. %33 darbe/periyot oranına sahip 4.77 MHz'lik işarettir.

RESET DRV1 (Çıkış) : Sisteme güç uygulandığında veya besleme gerilimi düştüğünde sistemi sıfırlar ve başlangıç konumuna getirir. Saat darbesinin düşen kenarı ile eş zamanlıdır.

A0 - A19 (Çıkış) : Bu sistemin 20 adres hattıdır. 1 MByte'a kadar bellek alanı adresleyebilir. A0 en düşük ağırlıklı bittir (LSB). A19 ise en yüksek ağırlıklı bittir (MSB). Ayrıca I/O adreslemede A0 - A9 adres yolları kullanılabilir. Bu işaretler 1 seviyesinde aktiftirler.

D0 - D7 (Giriş/Çıkış) : Bunlar işlem ünitesi, bellek ve I/O cihazları için veri bitlerini oluşturan işaretlerdir. En düşük ağırlıklı bit D0 ve en yüksek ağırlıklı bit D7'dir.

ALE (Çıkış) : Bu işaretin amacı adres kaydedicilerini kontrol etmektir. Bu işaret 8288 veri yolu kontrol ünitesi tarafından sağlanır. İşlem ünitesi adresleri bu işaretin düşen kenarı ile kaydedilir. DMA periyotları sırasında etkin değildir. Veriyi kaydetmesi emrini verir. 0 seviyesinde

I/O CH CK (Giriş) : Bu işaret I/O kanalı üzerindeki boşluk (Parite) hatalarını bildirir. Bu işaret etkin olduğu 0 seviyesinde iken bir boşluk hatasına rastlanmış demektir.

I/O CH RDY (Giriş) : Bu işaret normalde 1 seviyesinde

olup, I/O kanalı üzerindeki yavaş belleğe bağlı I/O ünitelerinden herhangi bir okuma yapılırken işlemin süresini uzatmak için 0 seviyesine çekilir. Bu seviyede kaldığı sürece I/O kanalı hazır değil demektir. Bu süre hiç bir zaman 10 saat darbesini geçmemelidir. Aktif seviyeleri 0'dır. Dinamik bellek IRQ2'den IRQ7'ye (Giriş) : Bunlar kesinti isteğini oluşturan 6 adet giriştir. Herhangi bir I/O cihazı işlem ünitesinin dikkatini çekmeye çalıştığında, meydana getirilirler ve işlem ünitesine veri yolları üzerinden iletilirler. Öncelik sırası IRQ2'den IRQ7'ye doğru düşer. Bir kesinti isteği ilgili IRQ hattının 0'dan 1'e çekilmesiyle oluşturulur. Kontrolü DMA kontrol ünitesinden geçer.

IOR (Çıkış) : Bu işaret, bir I/O cihazına göndereceği veriyi veri yoluna sürmesi emrini verir. 0 seviyesinde aktiftir. Onu belirtir.

IOW (Çıkış) : Bu işaret ise bir I/O cihazına veri yolu üzerindeki bilgiyi okuması emrini verir. 0 seviyesinde aktiftir. yapısına uygun olarak tasarlanmıştır. PC yapısı ile uyumlu.

MEMR (Çıkış) : Bu işaret bellek elemanına, üzerindeki bilgiyi veri yoluna sürme emri verir. 0 seviyesinde aktiftir.

MEMW (Çıkış) : Bu işaret bellek elemanına, veri yolu üzerindeki veriyi kaydetmesi emrini verir. 0 seviyesinde aktiftir. İsmi tarafından verilen teknik özelliklere ait

DRQ1 - DRQ3 (Giriş) : Bu işaretler doğrudan bellek erişimi isteğini bildiren ve eşzamanlı olmayan kanal işaretleridir. İsmi Çevre ve elemanları 74LS tarafından DMA

gereksinimlerini bildirme amacı ile kullanılırlar. ilgili DRQ hattının 1'e çekilmesi ile oluşturulurlar.

DACK0 - DACK3 (Çıkış) : Bu işaretler DMA isteginin adrese yerleştirilebilecek yapıda tasarlanmıştır. alındığını göstermek için ve sistem dinamik belleğinin kart üzerinde A3'den A8'e kadar olan adresleri temizlenmesi için kullanılır. Aktif seviyeleri 0'dir. Dinamik bellek temizleme işleminde DACK0 kullanılır.

AEN (Çıkış) : Bu işaret DMA için gerekli veri seçilebilir olarak tasarlanmıştır. Genel olarak iletişimlerine olanak sağlamak amacıyla, işlem ünitesini ve adresi olan 2F8 - 2FF adreslerine veri göndermek için diğer ilgili elemanları I/O kanalına bağlamak amacıyla kullanılır. Bu işaret etkin olduğu 1 seviyesinde iken adres üzerinde A0, A1 ve A2 adres yollarına bağlı olarak veri yolları, veri yolları ile bellek ve I/O okuma/yazma işlemlerinin kontrolü DMA kontrol ünitesinden geçer.

T/C (Çıkış) : Bu işaret bir DMA kanalı önceden tanımlanmış sayıdaki priyodu tamamladığında 1 seviyesine arabiriminde RS 232'ye göre farklı olan bir üniteye çıkarak bunu belirtir.

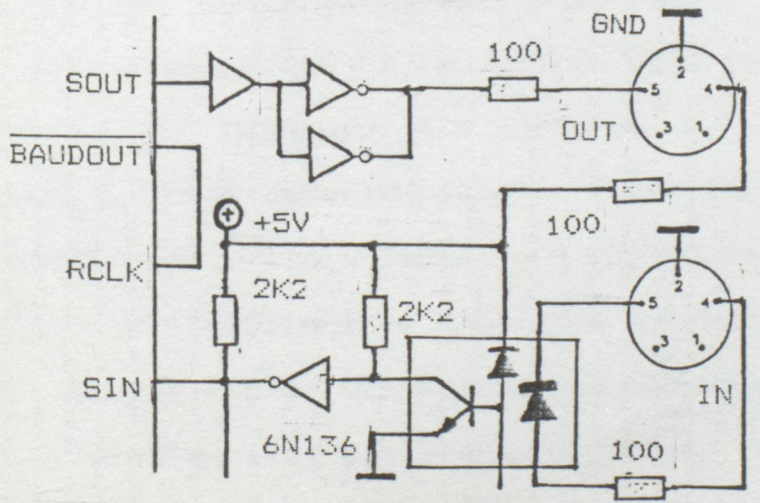
4-2. PC MIDI ARABİRİMİ DONANIMI

PC MIDI arabirimi I/O kanallarının önceki kısımda anlatılan yapısına uygun olarak tasarlanmıştır. PC yapısı ile uyumu yüksek tutabilmek amacı ile PC Asenkron haberleşme kanalı ile yazılım uyumlu olan UM8250A entegresi kullanılmıştır. Bu entegre 56Kbaud'a kadar asenkron haberleşme yapabilen bir entegredir. Entegre ile ilgili üretici firma tarafından verilen teknik özelliklere ait bilgiler EK1 de bulunmaktadır.

Devre şekil olarak I/O kanallarına takılabilecek boyutta tasarlanmıştır. Veri yolları 74LS245 entegresi ile

tamponlanmıştır. Kolaylık sağlamak amacı ile kart I/O haritası üzerinde 200-207 ile 3F8-3FF arasında herhangi bir adrese yerleştirilebilecek yapıda tasarlanmıştır. Bu amaçla kart üzerinde A3'den A8'e kadar olan adreslerin evrikleride üretilmiş ve bir köprü gurubu ile seçilebilir hale getirilmiştir. Ayrıca IRQ3 ve IRQ4 kesinti seviyeleride seçilebilir olarak tasarlanmıştır. Genel olarak kartın COM2 adresi olan 2F8 - 2FF adreslerine yerleşmesi hedeflenmiş ve köprüler kart üzerinde bu şekilde koyulmuştur. (8250 üzerinde A0, A1 ve A2 adres yollarına bağlanan 3 adet adres hattı olduğundan 8 Byte'lık bir I/O alanı kaplamaktadır.)

Buraya kadar olan kısım, yapı olarak bir PC Asenkron haberleşme Ünitesi RS 232 den farklı değildir. MIDI arabiriminde RS 232'ye göre farklı olan kısım Ünitenin giriş çıkış kısmıdır. Bu kısım şekil 15'te de görüldüğü gibi optik bağlayıcılar (Optocupler) üzerinden bağlantı kurulacak şekilde tasarlanmıştır. Bu MIDI standardının bir özelliğidir.



Şekil 15. MIDI'de bağlantı optik bağlayıcılar üzerindendir.

Yani çıkış bir başka cihazda bulunan girişi (Bu bir optik bağlayıcı olacaktır) sürebilecek şekilde ve girişte bir başka cihaz tarafından sürülebilecek bir optik bağlayıcıdan oluşmaktadır. Oluşturulan yapıdan da kolayca anlaşılacağı gibi çıkış bir tek optik bağlayıcıyı sürebilecek şekildedir. Bu yüzden burada olduğu gibi hemen tüm cihazlarda MIDI in girişine gelen sinyalin bir kopyasını taşıyan ve MIDI out gibi sürme yeteneğine sahip bir MIDI thru çıkışı vardır. Bu sayede birden fazla MIDI cihazının bir MIDI çıkışına bağlanabilmesi mümkün olabilmektedir.

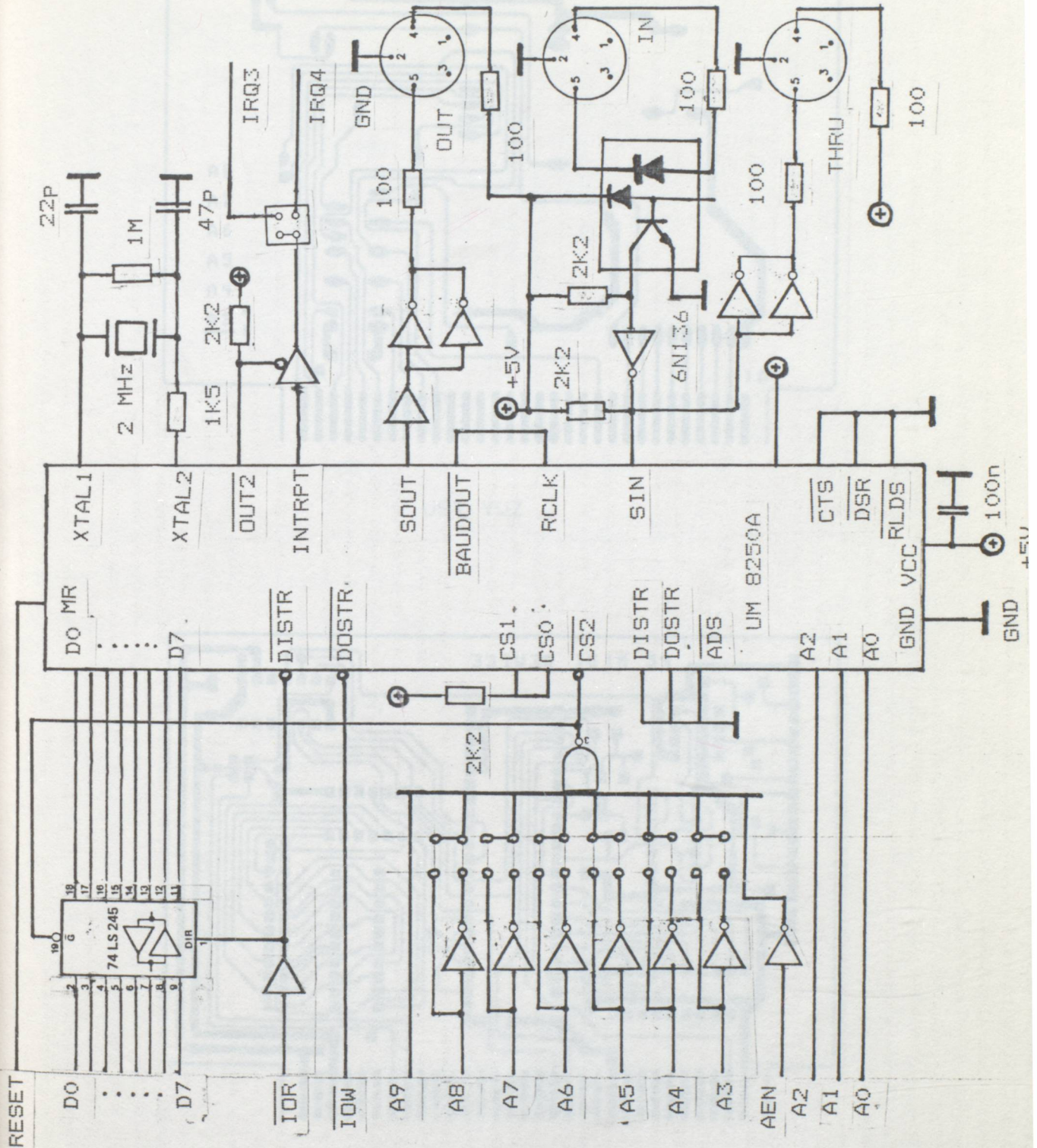
Şekil 16 MIDI arabiriminin elektronik şemasını göstermektedir. Şekil 17'de MIDI arabiriminin kartının baskılı devresinin üst ve alt yüzü görülmektedir. Şekil 18'de ise yine bu kartın yerleşim planı gösterilmiştir.

Devrede üretici firma değerlerinden farklı olarak 1.8432 MHz'lik kristal yerine 31250 Baudluk işareti tam olarak elde edebilmek amacı ile 2 MHz'lik bir kristal kullanılmıştır. 8250A entegresinin bölücüsü kullanılarak 2 MHz'lik kristalden 31250 Hz tam olarak elde edilebilmektedir. Bunun için bu bölücüyü 4'e bölmeye ayarlamak yeterli olmaktadır. (Bölme değeri=Kristal frekansı/16 x baud hızı)

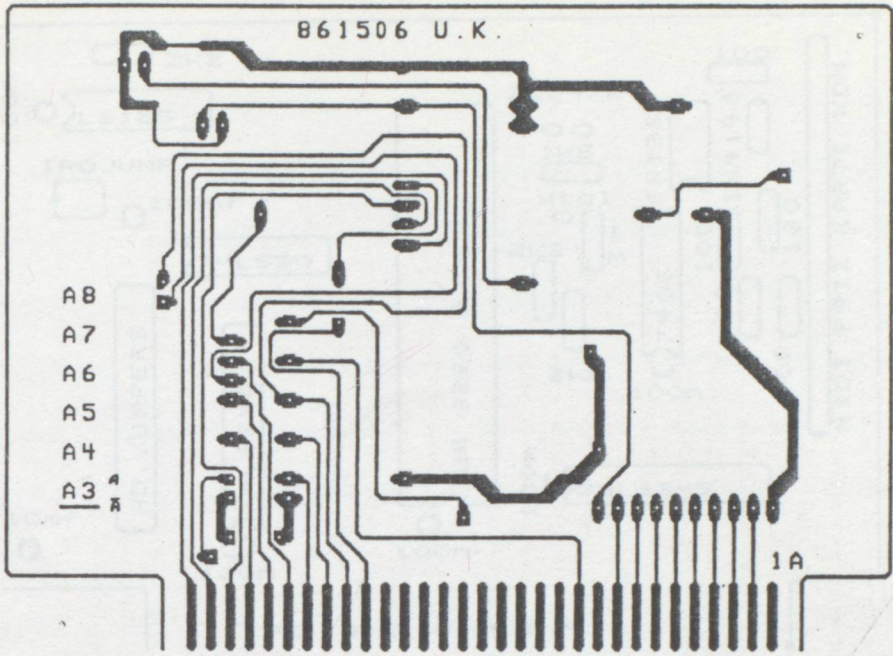
$$BD=2000000 / (16 \times 31250) = 4 \text{ olarak hesaplanır.}$$

MIDI in, MIDI out ve MIDI thru çıkışlarına standarta uyularak 5 ayaklı dişi DIN prizler koyulmuştur. Bu prizler PC'lerin dar olan slot çıkışlarından geçebilmesi için Şekil 19'da görülen karta yerleştirilmiş ve bir konnektör ile asıl

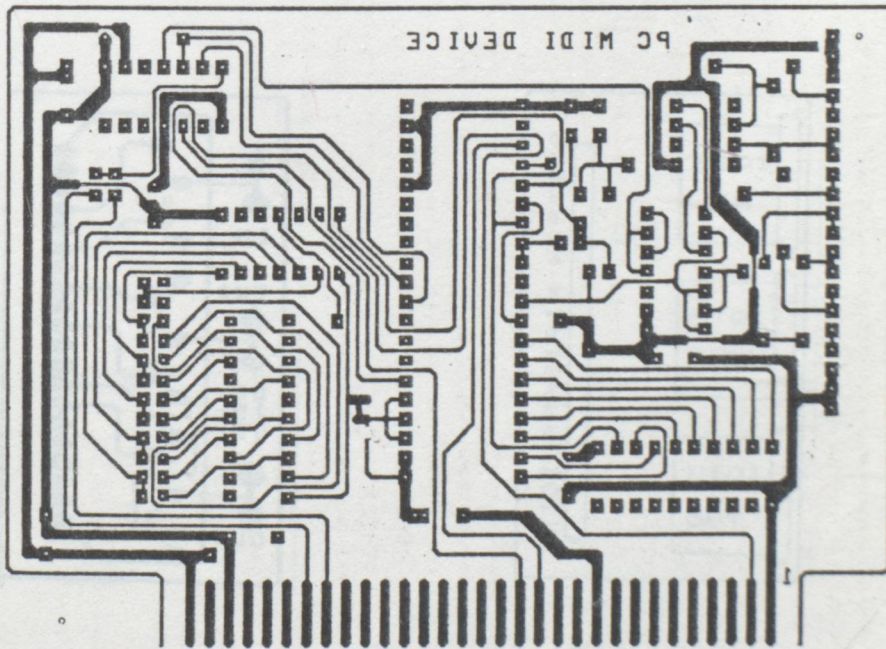
Şekil 17. MIDI arabiriminin elektronik şeması



Sekil 17. MIDI arabiriminin elektronik şeması



UST YUZ



ALT YUZ

Şekil 18. MIDI arabiriminin baskılı devresi.

4-3. MIDI ARABİRİMİ İÇİN GEREKLİ YAZILIM YAPISI

Bu çalışmada amaç MIDI konusunu açıklamak ve bir MIDI arabiriminin donanımının gerçekleştirilmesi olarak belirlenmişti. PC için bir MIDI programının yazılması tamamen bir programcılık işi olması ve kulfetli olması sebebiyle konu dışı bırakılmıştı. Ancak bu arabirimin kullanılması için gerekli programın oluşturulmasında donanımın, yazılıma neler sunduğunun bilinmesi gereklidir. Bu amaçla aşağıda maddeler halinde MIDI arabirim kartının yazılıma sunduğu olanaklar belirtilmiştir.

* MIDI arabirim kartı I/O alanında 200-207 ile 3F7-3FF adresleri arasında herhangi bir adrese yerleştirilebilir. Bu seçim yapılırken sistemin diğer I/O ünitelerinin durumu da göz ardı edilmemelidir ve uygun adresler seçilmelidir. COM2 adresi olan 2F8-2FF adresinin bu arabirim için uygun olacağı düşünülmüştür.

* Kart üzerinde IRQ3 ve IRQ4 kesinti istegi bağlantılarına herhangi biri seçilebilecek şekilde yer verilmiştir. Birincil ve ikincil asenkron haberleşmede kullanılan kesinti istegi bağlantıları olmaları sebebi ile bu uçlara yer verilmiştir.

* 8 byte'lik alana yerleşmiş 8250A kütük (Register) sıralaması tablo 1'de olduğu gibidir.

* UM 8250 A entegresinin baud hızı üretici bölücüsü (Programmable baud rate generator) 31250 baud elde edilebilmesi için 4'e ayarlanmalıdır.

DLAB	A ₂	A ₁	A ₀	Register
0	0	0	0	Receiver Buffer (read), Transmitter Holding Register (write).
0	0	0	1	Interrupt Enable.
X	0	1	0	Interrupt Identification (read only).
X	0	1	1	Line Control.
X	1	0	0	MODEM Control.
X	1	0	1	Line Status.
X	1	1	0	MODEM Status.
X	1	1	1	None.
1	0	0	0	Divisor Latch (least significant byte).
1	0	0	1	Divisor Latch (most significant byte).

Tablo 1. 8250 A kütüklerinin sıralanışı.

* Tablo 2'de UM 8250 A entegresinin kütüklerinin her bitinin ne anlama geldiği verilmiştir.

Bit No.	Register Address									
	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	3	4	5	6	0 DLAB=1	1 DLAB=1
	Receiver Buffer Register (Read Only)	Transmitter Holding Register (Write Only)	Interrupt Enable Register	Interrupt Ident. Register (Read Only)	Line Control Register	MODEM Control Register	Line Status Register	MODEM Status Register	Divisor Latch (LS)	Divisor Latch (MS)
	RBR	THR	IER		LCR	MCR	LSR	MSR	DLL	DLM
0	Data Bit 0*	Data Bit 0	Enable Received Data Available Interrupt (ERBFI)	"0" if Interrupt Pending	Word Length Select Bit 0 (WLS0)	Data Terminal Ready (DTR)	Data Ready (DR)	Delta Clear to Send (DCTS)	Bit 0	Bit 8
1	Data Bit 1	Data Bit 1	Transmitter Holding Register Empty Interrupt (ETBEI)	Interrupt ID Bit (0)	Word Length Select Bit 1 (WLS1)	Request to Send (RTS)	Overrun Error (OR)	Delta Data Set Ready (DDSR)	Bit 1	Bit 9
2	Data Bit 2	Data Bit 2	Enable Receiver Line Status Interrupt (ELSI)	Interrupt ID Bit (1)	Number of Stop Bits (STB)	Out 1	Parity Error (PE)	Trailing Edge Ring Indicator (TERI)	Bit 2	Bit 10
3	Data Bit 3	Data Bit 3	Enable MODEM Status Interrupt (EDSSI)	0	Parity Enable (PEN)	Out 2	Framing Error (FE)	Delta Receive Line Signal Detect (DRLSD)	Bit 3	Bit 11
4	Data Bit 4	Data Bit 4	0	0	Even Parity Select (EPS)	Loop	Break Interrupt (BI)	Clear to Send (CTS)	Bit 4	Bit 12
5	Data Bit 5	Data Bit 5	0	0	Stick Parity	0	Transmitter Holding Register Empty (THRE)	Data Set Ready (DSR)	Bit 5	Bit 13
6	Data Bit 6	Data Bit 6	0	0	Set Break	0	Transmitter Shift Register Empty (TSRE)	Ring Indicator (RI)	Bit 6	Bit 14
7	Data Bit 7	Data Bit 7	0	0	Divisor Latch Access Bit (DLAB)	0	0	Received Line Signal Detect (RLSD)	Bit 7	Bit 15

* Bit 0 is the least significant bit. It is the first bit serially transmitted or received.

Tablo 2. 8250 A kütüklerinin ve bitlerinin anlamları.



UM8250A

Asynchronous Communication Element (ACE)

Features

- Adds or deletes standard asynchronous communication bits (start, stop and parity) to or from serial data stream.
- Full double buffering eliminates need for precise synchronization.
- Independently-controlled transmit, receive, line status, and data set interrupts.
- Programmable baud rate generator allows division of any input clock by 1 to $(2^{16}-1)$ and generates the internal 16K clock.
- Independent receiver clock input.
- Modem control functions (CTS, RTS, DSR, DTR, RI, and carrier detect).
- Single +5 volt power supply.
- Fully programmable serial-interface characteristics.
 - 5-, 6-, 7-, or 8-bit characters.
 - Even, odd, or no-parity bit generation and detection.
 - 1-, 1½-, or 2-stop bit generation.
 - Baud rate generation (DC to 56K baud).
 - False start bit detection.
 - Complete status reporting capabilities.
 - Easily interfaces to most popular microprocessors.
 - Line break generation and detection.
 - Internal diagnostic capabilities.
 - Loopback controls for communications link fault isolation.
 - Break, parity, overrun, framing error simulation.
 - Full prioritized interrupt system controls.

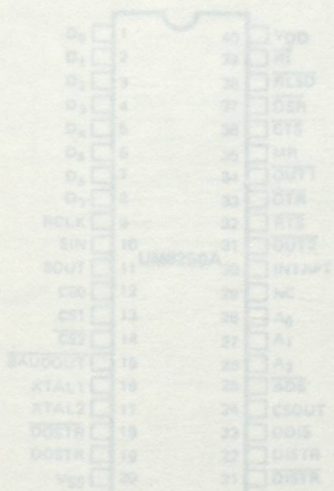
General Description

The UM8250A is a programmable Asynchronous Communication Element (ACE) that interfaces a microprocessor to a data character received from a peripheral device or a modem, and parallel-to-serial conversion on data characters received from the CPU. The CPU can read the complete

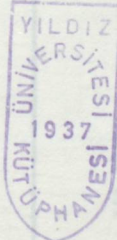
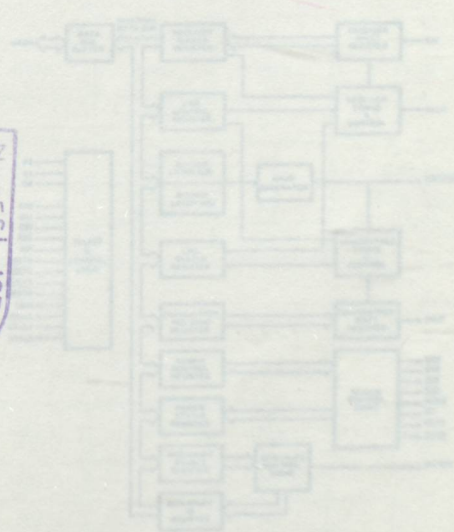
EK 1

status of the ACE at any time during operation. It also provides a loopback control for communications link fault isolation. The UM8250A is capable of dividing an input clock by divisors of 1 to $(2^{16}-1)$, and producing a 16K clock for driving the internal transmitter logic.

Pin Configuration



Block Diagram





UM8250A

Asynchronous Communication Element (ACE)

Features

- Adds or deletes standard asynchronous communication bits (start, stop and parity) to or from serial data stream.
- Full double buffering eliminates need for precise synchronization.
- Independently-controlled transmit, receive, line status, and data set interrupts.
- Programmable baud rate generator allows division of any input clock by 1 to $(2^{16}-1)$ and generates the internal 16X clock.
- Independent receiver clock input.
- Modem control functions (CTS, RTS, DSR, DTR, RI, and carrier detect).
- Single +5 volt power supply.
- Fully programmable serial-interface characteristics.

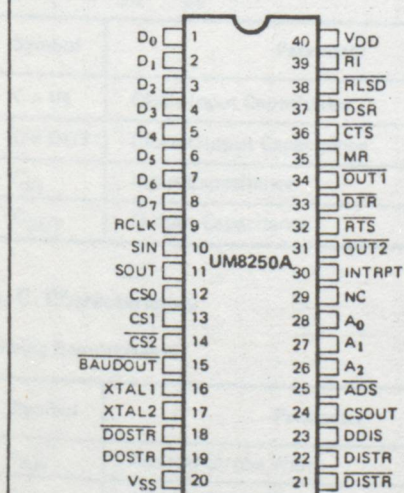
- 5-, 6-, 7-, or 8-bit characters.
- Even, odd, or no-parity bit generation and detection.
- 1-, 1½-, or 2-stop bit generation.
- Baud rate generation (DC to 56K baud).
- False start bit detection.
- Complete status reporting capabilities.
- Easily interfaces to most popular microprocessors.
- Line break generation and detection.
- Internal diagnostic capabilities.
 - Loopback controls for communications link fault isolation.
 - Break, parity, overrun, framing error simulation.
- Full prioritized interrupt system controls.

General Description

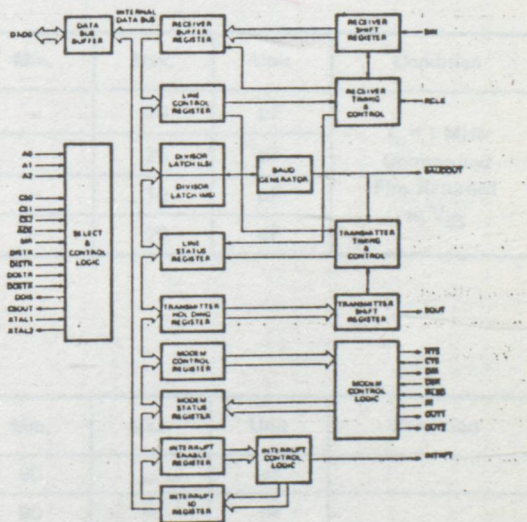
The UM8250A is a programmable Asynchronous Communication Element (ACE) chip fabricated with Si-Gate NMOS process. This product performs serial-to-parallel conversion on data characters received from a peripheral device or a modem, and parallel-to-serial conversion on data characters received from the CPU. The CPU can read the complete

status of the ACE at any time during operation. It also includes a programmable baud rate generator that is capable of dividing the timing reference clock input by divisors of 1 to $(2^{16}-1)$, and producing a 16X clock for driving the internal transmitter logic.

Pin Configuration



Block Diagram





UM8250A

Absolute Maximum Rating*

D.C. Supply Voltage V_{DD} -0.5V to 7V
 (With respect to V_{SS})
 Operating Temperature. 0°C – 70°C
 Storage Temperature -65°C – 150°C

Comments*

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only. Functional operation of this device at these or any other conditions above those indicated in the operational sections of this specification is not implied and exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D. C. Characteristics

($T_A = 25^\circ\text{C}$, $V_{DD} = 5V \pm 5\%$ unless otherwise specified).

Symbol	Parameter	Min.	Typ.	Max.	Unit	Condition
V_{IHx}	Clock Input High-Level Voltage	2.0	—	V_{DD}	V	
V_{ILx}	Clock Input Low-Level Voltage	-0.5	—	0.8	V	
V_{IH}	Input High-Level Voltage	2.0	—	V_{DD}	V	
V_{IL}	Input Low-Level Voltage	-0.5	—	0.8	V	
V_{OH}	Output High-Level Voltage	2.4	—	—	V	$I_{OL} = 1.6\text{ mA}$ on all outputs
V_{OL}	Output Low-Level Voltage	—	—	0.4	V	$I_{OH} = -100\text{ uA}$ on all outputs
I_{IN}	Input Leakage	—	—	± 10	μA	$V_{IN} = 5V - 0V$
I_{CIN}	Input Leakage	—	—	± 10	μA	Clock input
$I_{CC} \text{ (AVG)}$	AVG Power Supply Current	—	—	90	mA	$V_{IN} = V_{DD}$ or 0V, No load
I_{OZ}	OFF-State Output Current	—	—	20	μA	$V_O = V_{DD} - 0V$

Capacitance

($T_A = 25^\circ\text{C}$, $V_{DD} = V_{SS} = 0V$)

Symbol	Parameter	Min.	Max.	Unit	Condition
$C_{x\text{ IN}}$	Clock Input Capacitance	—	20	pF	$f_c = 1\text{ MHz}$ Unmeasured Pins Returned to V_{SS}
$C_{x\text{ OUT}}$	Clock Output Capacitance	—	25	pF	
C_{IN}	Input Capacitance	—	10	pF	
C_{OUT}	Output Capacitance	—	20	pF	

A. C. Characteristics**Timing Requirements**

Symbol	Parameter	Min.	Max.	Unit	Condition
t_{AW}	Address Strobe Width	90	—	ns	
t_{AS}	Address Setup Time	90	—	ns	

(Cont.)



Symbol	Parameter	Min.	Max.	Unit	Condition
t_{AH}	Address Hold Time	0	—	ns	
t_{CS}	Chip Select Setup Time	90	—	ns	
t_{CH}	Chip Select Hold Time	0	—	ns	
t_{DIW}	$\overline{DISTR}/DISTR$ Strobe Width	175	—	ns	
t_{RC}	Read Cycle Delay	500	—	ns	
RC	Read Cycle = $t_{AW} + t_{DID} + t_{DIW} + t_{RC}$	755	—	ns	
t_{DOW}	$\overline{DOSTR}/DOSTR$ Strobe Width	175	—	ns	
t_{WC}	Write Cycle Delay	500	—	ns	
WC	Write Cycle = $t_{AW} + t_{DOD} + t_{DOW} + t_{WC}$	755	—	ns	
t_{DS}	Data Setup Time	90	—	ns	
t_{DH}	Data Hold Time	60	—	ns	
t_{RA}^*	Address Hold Time from $\overline{DISTR}/DISTR$	20	—	ns	
t_{RCS}^*	Chip Select Hold Time from $\overline{DISTR}/DISTR$	20	—	ns	
t_{AR}^*	$\overline{DISTR}/DISTR$ Delay from Address	80	—	ns	
t_{CSR}^*	$\overline{DISTR}/DISTR$ Delay from Chip Select	80	—	ns	
t_{WA}^*	Address Hold Time from $\overline{DOSTR}/DOSTR$	20	—	ns	
t_{WCS}^*	Chip Select Hold Time from $\overline{DOSTR}/DOSTR$	20	—	ns	
t_{AW}^*	$\overline{DOSTR}/DOSTR$ Delay from Address	80	—	ns	
t_{CSW}^*	$\overline{DOSTR}/DOSTR$ Delay from Chip Select	80	—	ns	
t_{MRM}	Master Reset Pulse Width	10	—	ns	

Timing Response

Symbol	Parameter	Min.	Max.	Unit	Condition
t_{CSS}	CSOUT Delay from Latch	—	90	ns	
t_{DD}	$\overline{DISTR}/DISTR$ to Drive Disable Time	—	75	ns	All Outputs Loading 40 pF
t_{DDD}	Delay from $\overline{DISTR}/DISTR$ to DATA	—	175	ns	
t_{HZ}	$\overline{DISTR}/DISTR$ to Floating Data Delay	100	300	ns	
t_{CSC}^*	CSOUT Delay from Select	—	125	ns	

* Only applies when $\overline{ADS}$ is tied Low.



Other Timing

Baud Generator

Symbol	Parameter	Min.	Max.	Unit	Condition
N	Baud Rate Divisor	1	$2^{16} - 1$		
t_{BLD}	Baud Output Negative Edge Delay	—	250	ns	40 pF loading
t_{BHD}	Baud Output Positive Edge Delay	—	250	ns	40 pF loading
t_{LW}	Baud Output Down Time	425	—	ns	40 pF loading
t_{HW}	Baud Output Up Time	330	—	ns	40 pF loading

Receiver

Symbol	Parameter	Min.	Max.	Unit	Condition
t_{SCD}	Delay from RCLK to Sample Time	—	2	us	
t_{SINT}	Delay from Stop Bit to Set Interrupt	—	3	RCLK Cycle	40 pF loading
t_{RINT}	Delay from $\overline{DISTR}/DISTR$ (RD RBR/RDLSR) to Reset Interrupt	—	890	ns	40 pF loading

Transmitter

Symbol	Parameter	Min.	Max.	Unit	Condition
t_{HR}	Delay from $\overline{DOSTR}/DOSTR$ (WR INT) to Reset Interrupt	—	890	ns	40 pF loading
t_{IRS}	Delay from Initial INTR Reset to Transmit Start	8	24	$\overline{BAUDOUT}$ Cycle	
t_{SI}	Delay from Initial Write to Interrupt	16	32	$\overline{BAUDOUT}$ Cycle	
t_{SS}	Delay from Stop to Next Start	—	1	us	40 pF loading
t_{STI}	Delay from Stop to Interrupt (THRE)	—	8	$\overline{BAUDOUT}$ Cycle	
t_{IR}	Delay from $\overline{DISTR}/DISTR$ (RD IIR) to Reset Interrupt (THRE)	—	890	ns	40 pF loading



Modem Control

Symbol	Parameter	Min.	Max.	Unit	Condition
t_{MDO}	Delay from $\overline{DOSTR}/\overline{DOSTR}$ (WR MCR) to Output	—	1,000	ns	40 pF loading
t_{SIM}	Delay to Set Interrupt from MODEM Input	—	1,000	ns	40 pF loading
t_{RIM}	Delay to Reset Interrupt from $\overline{DISTR}/\overline{DISTR}$ (RD MSR)	—	1,000	ns	40 pF loading

Pin Description

Input Signal

Chip Select ($\overline{CS0}$, $\overline{CS1}$, $\overline{CS2}$), Pins 12–14: When $\overline{CS0}$ and $\overline{CS1}$ are high and $\overline{CS2}$ is low, the chip is selected. Chip selection is complete when the decoded chip select signal is latched with an active (low) Address Strobe ($\overline{ADS}$) input.

Data Input Strobe ($\overline{DISTR}$, $\overline{DISTR}$), Pins 21 and 22: When $\overline{DISTR}$ is high or when $\overline{DISTR}$ is low, the chip is selected; allows the CPU to read status information or data from a selected register of the UM8250A.

Data Output Strobe ($\overline{DOSTR}$, $\overline{DOSTR}$), Pins 18 and 19: The chip is selected when $\overline{DOSTR}$ is high or $\overline{DOSTR}$ is low and allows the CPU to write data or control words into a selected register of the UM8250A.

Note that only one of these two inputs, $\overline{DISTR}$ and $\overline{DISTR}$ ($\overline{DOSTR}$ and $\overline{DOSTR}$) is needed to activate the CPU read (Write). Tie either $\overline{DISTR}$ ($\overline{DOSTR}$) low or $\overline{DISTR}$ ($\overline{DOSTR}$) high if not needed.

Address Strobe ($\overline{ADS}$), Pin 25: When low, provides latching for the Register Select ($\overline{A0}$, $\overline{A1}$, $\overline{A2}$) and Chip Select ($\overline{CS0}$, $\overline{CS1}$, $\overline{CS2}$) signals. An active $\overline{ADS}$ input is required when the Register Select ($\overline{A0}$, $\overline{A1}$, $\overline{A2}$) signals are not stable for the duration of a read or write operation. If not required, tie the $\overline{ADS}$ input permanently low.

Register Select ($\overline{A0}$, $\overline{A1}$, $\overline{A2}$), Pins 26–28: These three inputs are used during a read or write operation to select a UM8250A register to read from or write into as indicated in Table 1. Note that the state of the Divisor Latch Access Bit (DLAB), which is the most significant bit of the Line Control Register, affects the selection of certain UM8250A registers. The DLAB must be set high by the system software to access the Baud Generator Divisor Latches.

Receiver Clock (RCLK), Pin 9: This input is the 16X baud rate clock for the receiver section of the chip.

Serial Input (SIN), Pin 10: Serial data input from the communications link (peripheral device, MODEM, or data set).

Clear to Send ($\overline{CTS}$), Pin 36: The $\overline{CTS}$ signal is a MODEM control function input whose conditions can be tested by the CPU by reading bit 4 (CTS) of the MODEM Status Register. Bit 0 (DCTS) of the MODEM Status Register indicates whether the $\overline{CTS}$ input has changed state since the previous reading of the MODEM Status Register. CTS has no effect on the Transmitter. Whenever the CTS bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Data Set Ready ($\overline{DSR}$), Pin 37: When low, this indicates that the MODEM or data set is ready to establish the communications link and transfer data with the ACE. The DSR signal is a MODEM-control function input whose condition can be tested by the CPU by reading bit 5 (DSR) of the MODEM Status Register. Bit 1 (DDSR) of the MODEM Status Register indicates whether the $\overline{DSR}$ input has changed state since the previous reading of the MODEM Status Register. Whenever the DSR bit of the MODEM Status Register changes state, an interrupt is generated if the MODEM Status Interrupt is enabled.

Received Line Signal Detect Pin 38: When low, indicates that the data carrier has been detected by MODEM or data set.

The $\overline{RLSD}$ Signal is a MODEM Control function input and can be tested by reading bit 7 (RLSD) of the MSR. Bit 3 (DRLSD) of MSR indicates whether the $\overline{RLSD}$ input has changed state since the previous reading of MSR. Whenever the RLSD bit of MSR changes state, an interrupt is generated if the MODEM Status Interrupt is enable.



Table 1.

DLAB	A ₂	A ₁	A ₀	Register
0	0	0	0	Receiver Buffer (read), Transmitter Holding Register (write).
0	0	0	1	Interrupt Enable.
X	0	1	0	Interrupt Identification (read only).
X	0	1	1	Line Control.
X	1	0	0	MODEM Control.
X	1	0	1	Line Status.
X	1	1	0	MODEM Status.
X	1	1	1	None.
1	0	0	0	Divisor Latch (least significant byte).
1	0	0	1	Divisor Latch (most significant byte).

Ring Indicator ($\overline{RI}$), Pin 39: When low, indicates that a telephone ringing signal has been received by the MODEM or data set. The $\overline{RI}$ signal is a MODEM-control function input whose condition can be tested by the CPU by reading bit 6 (RI) of the MODEM Status Register. Bit 2 (TERI) of the MODEM Status Register indicates whether the RI input has changed from a low to a high state since the previous reading of the MODEM Status Register. Whenever the $\overline{RI}$ bit of the MODEM Status Register changes from a high to a low state, an interrupt is generated if the MODEM

Status Register is enabled.

Master Reset (MR), Pin 35: This input is buffered with a TTL-compatible Schmitt Trigger with 0.5V typical hysteresis. When high, it clears all the registers (except the Receiver Buffer, Transmitter Holding, and Divisor Latches) and the control logic of the UM8250A. Also, the state of various output signals ($\overline{SOUT}$, $\overline{INTRPT}$, $\overline{OUT 1}$, $\overline{OUT 2}$, $\overline{RTS}$, $\overline{DTR}$) are affected by an active MR input. (Refer to Table 2).

Table 2. Reset Function

Register/Signal	Reset Control	Reset State
Interrupt Enable Register	Master Reset	All bits Low (0-3 forced and 4-7 permanent).
Interrupt Identification Register	Master Reset	Bit 0 is High, Bits 1 and 2 Low, Bits 3-7 are Permanently Low.
Line Control Register	Master Reset	All Bits Low.
MODEM Control Register	Master Reset	All Bits Low.
Line Status Register	Master Reset	All Bits Low, Except Bits 5 and 6 which are High.
MODEM Status Register	Master Reset	Bits 0-3 Low, Bits 4-7 - Input Signal.
$\overline{SOUT}$	Master Reset	High
$\overline{INTRPT}$ (RCVR Errs)	Read LSR/MR	Low
$\overline{INTRPT}$ (RCVR Data Ready)	Read RBR/MR	Low
$\overline{INTRPT}$ (THRE)	Read IIR/Write THR/MR	Low
$\overline{INTRPT}$ (Modem Status Changes)	Read MSR/MR	Low
$\overline{OUT 2}$	Master Reset	High
$\overline{RTS}$	Master Reset	High
$\overline{DTR}$	Master Reset	High
$\overline{OUT 1}$	Master Reset	High



UM8250A

Output Signals

Request to Send ($\overline{\text{RTS}}$), Pin 32: When low, informs the MODEM or data set that the UM8250A is ready to transmit data. The $\overline{\text{RTS}}$ output signal can be set to an active low by programming bit 1 (RTS) of the MODEM Control Register. The RTS signal is set high in a Master Reset operation.

Data Terminal Ready ($\overline{\text{DTR}}$), Pin 33: When low, informs the MODEM or data set that the UM8250A is ready to communicate. The DTR output signal can be set to an active low by programming bit 0 (DTR) of the MODEM Control Register to a high level. The DTR signal is set high in a Master Reset operation.

Output 1 ($\overline{\text{OUT 1}}$), Pin 34: User-designated output can be set to an active low by programming bit 2 ($\overline{\text{OUT 1}}$) of the MODEM Control Register to a high level. The $\overline{\text{OUT 1}}$ signal is set high in a Master Reset Operation.

Output 2 ($\overline{\text{OUT 2}}$), Pin 31: User-designated output can be set to an active low by programming bit 3 ($\overline{\text{OUT 2}}$) of the MODEM Control Register to a high level. The $\overline{\text{OUT 2}}$ signal is set high in a Master Reset Operation.

*Note that the $\overline{\text{RTS}}$, $\overline{\text{DTR}}$, $\overline{\text{OUT 1}}$ and $\overline{\text{OUT 2}}$ are forced to inactive state (high) during loop mode operation.

Chip Select Out (CSOUT), Pin 24: When high, indicates that the chip has been selected by active, CS0, CS1, and CS2 inputs. No data transfer can be initiated until the CSOUT signal is a logic 1. When chip is not selected, CSOUT remains low.

Driver Disable (DDIS), Pin 23: This pin remains low whenever the CPU is reading data from UM8250A. A high-level output can be used to disable an external transceiver if the CPU is not reading.

Baud Out ($\overline{\text{BAUDOUT}}$), Pin 15: 16X clock signal for the transmitter section of the UM8250A. The clock rate is equal to the main reference oscillator frequency divided by the specified divisor in the Baud Generator Divisor Latches.

The $\overline{\text{BAUDOUT}}$ may also be used for the receiver section by tying this output to the RCLK input of the chip.

Interrupt (INTRPT), Pin 30: Goes high whenever any of the following interrupt types has an active high con-

dition and is enabled via the IER: Receiver Error Flag; Received Data Available; Transmitter Holding Register Empty; and MODEM Status. The INTRPT signal is reset to low at the appropriate interrupt service or in a Master Reset operation.

Serial Output (SOUT), Pin 11: Composite serial data output to the communications link (peripheral, MODEM or data set). The SOUT signal is set to the Marking (logic 1) state in a Master Reset operation.

Input/Output Signals

Data (D_7-D_0) Bus, Pins 1-8: This bus comprises eight TRI-STATE input/output lines. The bus provides bidirectional communications between the UM8250A and the CPU. Data, control words, and status information are transferred via the D_7-D_0 Data Bus.

External Clock Input/Output (XTAL 1, XTAL 2), Pins 16 and 17: These two pins connect the main timing reference (crystal or clock signal) to the UM8250A.

Accessible Registers

There are 10 registers, as shown in Table 3, which may be accessed or controlled by the programmer via the CPU. These registers are used to control operations and to transmit and receive data.

Line Control Register

The system programmer specifies the format of the asynchronous data communications exchange via the Line Control Register. In addition to controlling the format, the programmer may retrieve the contents of the Line Control Register for inspection. This feature simplifies system programming and eliminates the need for separate storage in system memory of the line characteristics. The contents of the Line Control Register are indicated in Table 3 and are described below:

Bits 0 and 1: These two bits specify the number of bits in each transmitted or received serial character. The encoding of bits 0 and 1 is as follows:



Table 3. Summary of UM8250A Programmable Registers

Bit No.	Register Address									
	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	3	4	5	6	0 DLAB=1	1 DLAB=1
	Receiver Buffer Register (Read Only)	Transmitter Holding Register (Write Only)	Interrupt Enable Register	Interrupt Ident. Register (Read Only)	Line Control Register	MODEM Control Register	Line Status Register	MODEM Status Register	Divisor Latch (LS)	Divisor Latch (MS)
	RBR	THR	IER		LCR	MCR	LSR	MSR	DLL	DLM
0	Data Bit 0*	Data Bit 0	Enable Received Data Available Interrupt (ERBFI)	"0" if Interrupt Pending	Word Length Select Bit 0 (WLS0)	Data Terminal Ready (DTR)	Data Ready (DR)	Delta Clear to Send (DCTS)	Bit 0	Bit 8
1	Data Bit 1	Data Bit 1	Transmitter Holding Register Empty Interrupt (ETBEI)	Interrupt ID Bit (0)	Word Length Select Bit 1 (WLS1)	Request to Send (RTS)	Overrun Error (OR)	Delta Data Set Ready (DDSR)	Bit 1	Bit 9
2	Data Bit 2	Data Bit 2	Enable Receiver Line Status Interrupt (ELSI)	Interrupt ID Bit (1)	Number of Stop Bits (STB)	Out 1	Parity Error (PE)	Trailing Edge Ring Indicator (TERI)	Bit 2	Bit 10
3	Data Bit 3	Data Bit 3	Enable MODEM Status Interrupt (EDSSI)	0	Parity Enable (PEN)	Out 2	Framing Error (FE)	Delta Receive Line Signal Detect (DRLSD)	Bit 3	Bit 11
4	Data Bit 4	Data Bit 4	0	0	Even Parity Select (EPS)	Loop	Break Interrupt (BI)	Clear to Send (CTS)	Bit 4	Bit 12
5	Data Bit 5	Data Bit 5	0	0	Stick Parity	0	Transmitter Holding Register Empty (THRE)	Data Set Ready (DSR)	Bit 5	Bit 13
6	Data Bit 6	Data Bit 6	0	0	Set Break	0	Transmitter Shift Register Empty (TSRE)	Ring Indicator (RI)	Bit 6	Bit 14
7	Data Bit 7	Data Bit 7	0	0	Divisor Latch Access Bit (DLAB)	0	0	Received Line Signal Detect (RLSD)	Bit 7	Bit 15

* Bit 0 is the least significant bit. It is the first bit serially transmitted or received.



UM8250A

Bit 1	Bit 0	Word Length
0	0	5 Bits
0	1	6 Bits
1	0	7 Bits
1	1	8 Bits

Bit 2: This bit specifies the number of Stop bits in each transmitted or received serial character. If bit 2 is a logic 0, 1 Stop bit is generated or checked in the transmit or receive data, respectively. If bit 2 is a logic 1 and a 5-bit word length is selected via bits 0 and 1, 1½ Stop bits are generated or checked. If bit 2 is a logic 1 and a 6-, 7-, or 8-bit word length is selected, 2 Stop bits are generated or checked.

Bit 3: This bit is the Parity Enable bit. When bit 3 is a logic 1, a Parity bit is generated (transmit data) or checked (receive data) between the last data word bit and Stop bit of the serial data. (The Parity bit is used to produce an even or odd number of 1s when the data words bits and the Parity bit are added.)

Bit 4: This bit is the Even Parity Select bit. When bit 3 is a logic 1 and bit 4 is a logic 0, an odd number of logic 1s are transmitted or checked in the data word bits and Parity bit. When bit 3 is a logic 1 and bit 4 is a logic 1, an even number of logic 1s are transmitted or checked.

Bit 5: This bit is the Stick Parity bit. When bit 3 is a logic 1 and bit 5 is a logic 1, the Parity bit is transmitted and then detected by the receiver as a logic 0 if bit 4 is a logic 1 or as a logic 1 if bit 4 is a logic 0.

Bit 6: This bit is the Break Control bit. When bit 6 is a logic 1, the serial output (SOUT) is forced to the Spacing (logic 0) state and remains there regardless of other transmitter activities. The set break is disabled by setting bit 6 to a logic 0. This feature enables the CPU to alert a terminal in a computer communications system.

Bit 7: This bit is the Divisor Latch Access Bit (DLAB). It must be set high (logic 1) to access the Divisor Latches of the Baud Generator during a Read or Write operation. It must be set low (logic 0) to access the Receiver Buffer, the Transmitter Holding Register, or the Interrupt Enable Register.

Programmable Baud Rate Generator

The UM8250A contains a programmable Baud Rate Generator that is capable of taking any clock input (DC to 3.1 MHz) and dividing it by any divisor from 1 to ($2^{16}-1$). The output frequency of the Baud Generator is 16X the Baud rate [divisor # = (frequency input) ÷ (baud rate x 16)]. Two 8-bit latches store the divisor in a 16-bit binary format. These Divisor Latches must be loaded during initialization in order to ensure desired operation of the Baud Rate Generator. Upon loading either of the Divisor Latches, a 16-bit Baud counter is immediately loaded. This prevents long counts on initial load.

Table 4 and 5 illustrate the use of the Baud Generator with crystal frequencies of 1.8432 MHz and 3.072 MHz respectively. For baud rates of 38400 and below, the error obtained is minimal. The accuracy of the desired baud rate is dependent on the crystal frequency chosen. The maximum operating frequency of the Baud Generator is 3.1 MHz. However, when using divisors of 3 and below, the maximum frequency is equal to the divisor in MHz. For example, if the divisor is 1, then the maximum frequency is 1 MHz. In no case should the data rate be greater than 56K Baud.

Table 4. Baud Rates Using 1.8432 MHz Crystal

Desired Baud Rate	Divisor Used to Generate 16 x Clock	Percent Error Difference Between Desired and Actual
50	2304	—
75	1536	—
110	1047	0.026
133.5	857	0.0258
150	768	—
300	384	—
600	192	—
1200	96	—
1800	64	—
2000	58	0.69
2400	48	—
3600	32	—
4800	24	—
7200	16	—
9600	12	—
19200	6	—
38400	3	—
56000	2	2.86

Note: 1.8432 MHz is the standard 8080 frequency divided by 10.



UM8250A

Table 5. Baud Rates Using 3.072 MHz Crystal

Desired Baud Rate	Divisor Used to Generate 16 x Clock	Percent Error Difference Between Desired and Actual
50	3840	—
75	2560	—
110	1745	0.026
134.5	1428	0.034
150	1280	—
300	640	—
600	320	—
1200	160	—
1800	107	0.312
2000	96	—
2400	80	—
3600	53	0.628
4800	40	—
7200	27	1.23
9600	20	—
19200	10	—
38400	5	—

Fig 1 and 2 show the typical application circuits of the UM8250A's clock generation.

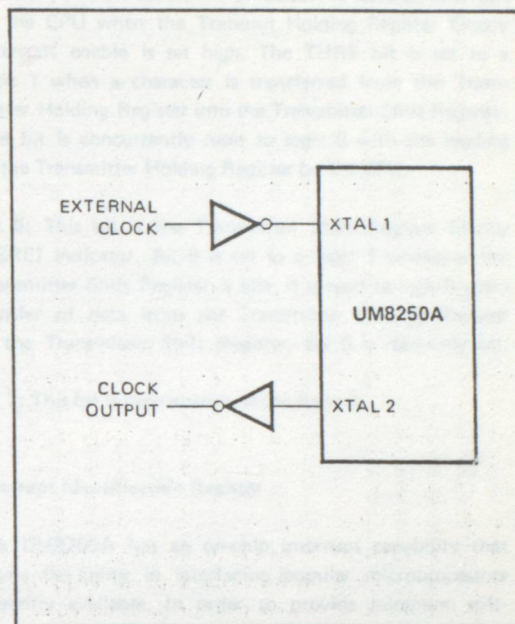


Figure 1. Clock Circuit with External Clock Signal

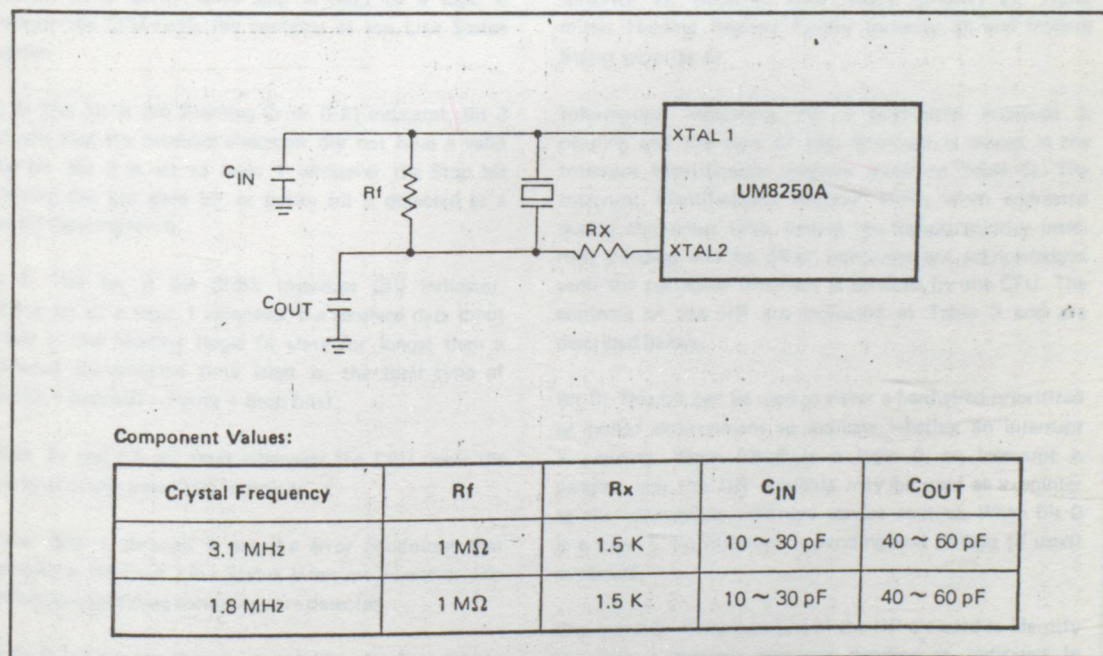


Figure 2. Clock Circuit with Crystal Oscillator



Line Status Register

This 8-bit register provides status information to the CPU concerning the data transfer. The contents of the Line Status Register are indicated in Table 3 and are described below.

Bit 0: This bit is the receiver Data Ready (DR) indicator. Bit 0 is set to a logic 1 whenever a complete incoming character has been received and transferred into the Receiver Buffer Register. Bit 0 may be reset to a logic 0 either by the CPU reading the data in the Receiver Buffer Register or by writing a logic 0 into it from the CPU.

Bit 1: This bit is the Overrun Error (OE) indicator. Bit 1 indicates that data in the Receiver Buffer Register was not read by the CPU before the next character was transferred into the Receiver Buffer Register, thereby destroying the previous character. The OE indicator is reset whenever the CPU reads the contents of the Line Status Register.

Bit 2: This bit is the Parity Error (PE) indicator. Bit 2 indicates that the received data character does not have the correct even or odd parity, as selected by the even-parity-select bit. The PE bit is set to a logic 1 upon detection of a parity error and is reset to a logic 0 whenever the CPU reads the contents of the Line Status Register.

Bit 3: This bit is the Framing Error (FE) indicator. Bit 3 indicates that the received character did not have a valid Stop bit. Bit 3 is set to logic 1 whenever the Stop bit following the last data bit or parity bit is detected as a zero bit (Spacing level).

Bit 4: This bit is the Break Interrupt (BI) indicator. Bit 4 is set to a logic 1 whenever the received data input is held in the Spacing (logic 0) state for longer than a full word transmission time (that is, the total time of Start bit + data bits + Parity + Stop bits).

*Note: BI and FE are reset whenever the CPU reads the contents of the Line Status indicator.

*Note: Bits 1 through 4 are the error conditions that produce a Receiver Line Status interrupt whenever any of the corresponding conditions are detected.

Bit 5: This bit is the Transmitter Holding Register Empty (THRE) indicator. Bit 5 indicates that the UM8250A is

ready to accept a new character for transmission. In addition, this bit causes the UM8250A to issue an interrupt to the CPU when the Transmitter Holding Register Empty Interrupt enable is set high. The THRE bit is set to a logic 1 when a character is transferred from the Transmitter Holding Register into the Transmitter Shift Register. The bit is concurrently reset to logic 0 with the loading of the Transmitter Holding Register by the CPU.

Bit 6: This bit is the Transmitter Shift Register Empty (TSRE) indicator. Bit 6 is set to a logic 1 whenever the Transmitter Shift Register is idle. It is reset to logic 0 upon transfer of data from the Transmitter Holding Register to the Transmitter Shift Register. Bit 6 is read-only bit.

Bit 7: This bit is permanently set to logic 0.

Interrupt Identification Register

The UM8250A has an on-chip interrupt capability that allows flexibility in interfacing popular microprocessors presently available. In order to provide minimum software overhead during data character transfers, the UM8250A prioritizes interrupts into four levels. The four levels of interrupt conditions are as follows: Receiver Line Status (priority 1); Received Data Ready (priority 2); Transmitter Holding Register Empty (priority 3); and Modem Status (priority 4).

Information indicating that a prioritized interrupt is pending and the type of that interrupt is stored in the Interrupt Identification Register (refer to Table 6). The Interrupt Identification Register (IIR), when addressed during chip-select time, freezes the highest priority interrupt pending and no other interrupts are acknowledged until the particular interrupt is serviced by the CPU. The contents of the IIR are indicated in Table 3 and are described below.

Bit 0: This bit can be used in either a hardwired prioritized or polled environment to indicate whether an interrupt is pending. When bit 0 is a logic 0, an interrupt is pending and the IIR contents may be used as a pointer to the appropriate interrupt service routine. When bit 0 is a logic 1, no interrupt is pending and polling (if used) continues.

Bits 1 and 2. These two bits of the IIR are used to identify the highest priority interrupt pending as indicated in Table 5.



Table 6. Interrupt Control Functions

Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 2	Bit 1	Bit 0	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Control
0	0	1	—	None	None	—
1	1	0	Highest	Receiver Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Reading the Line Status Register
1	0	0	Second	Received Data Available	Receiver Data Available	Reading the Receiver Buffer Register
0	1	0	Third	Transmitter Holding Register Empty	Transmitter Holding Register Empty	Reading the IIR Register (if source of interrupt) or Writing into the Transmitter Holding Register
0	0	0	Fourth	MODEM Status	Clear to Send or Data Set Ready or Ring Indicator or Received Line Signal Detect	Reading the MODEM Status Register

Bits 3 through 7: These five bits of the IIR are always set at logic 0.

Interrupt Enable Register

This 8-bit register enables the four types of interrupts of the UM8250A to separately activate the chip interrupt (INTRPT) output signal. It is possible to totally disable the interrupt system by resetting bits 0 through 3 of the interrupt Enable Register. Similarly, by setting the appropriate bits of this register to a logic 1, selected interrupts can be enabled. Disabling the interrupt system inhibits the Interrupt Identification Register and the active (high) INTRPT output from the chip. All other system functions operate in their normal manner, including the setting of the Line Status and MODEM Status Registers. The contents of the Interrupt Enable Register are indicated in Table 3 and are described below.

Bit 0: This bit enables the Received Data Available Interrupt when set to logic 1.

Bit 1: This bit enables the Transmitter Holding Register Empty Interrupt when set to logic 1.

Bit 2: This bit enables the Receiver Line Status Interrupt when set to logic 1.

Bit 3: This bit enables the MODEM Status Interrupt when set to logic 1.

Bits 4 through 7: These four bits are always set at logic 0.

MODEM Control Register

This 8-bit register controls the interface with the MODEM or data set (or a peripheral device emulating a MODEM). The contents of the MODEM Control Register are indicated in Table 3 and are described below.

Bit 0: This bit controls the Data Terminal Ready (DTR) output. When bit 0 is set to logic 1, the DTR output



is forced to a logic 0. When bit 0 is reset to a logic 0, the $\overline{\text{DTR}}$ output is forced to a logic 1. The $\overline{\text{DTR}}$ output of the UM8250A may be applied to an EIA inverting line driver (such as the DS1488) to obtain the proper polarity input at the succeeding MODEM or data set.

Bit 1: This bit controls the Request to Send ($\overline{\text{RTS}}$) output. Bit 1 affects the $\overline{\text{RTS}}$ output in a manner identical to that described above for bit 0.

Bit 2: This bit controls the output 1 ($\overline{\text{OUT 1}}$) signal, which is an auxiliary user-designated output. Bit 2 affects the $\overline{\text{OUT 1}}$ output in a manner identical to that described above for bit 0.

Bit 3: This bit controls the Output 2 ($\overline{\text{OUT 2}}$) signal, which is an auxiliary user-designated output. Bit 3 affects the $\overline{\text{OUT 2}}$ output in a manner identical to that described above for bit 0.

Bit 4: This bit provides a local loopback feature for diagnostic testing of the UM8250A. When bit 4 is set to logic 1, the following occur: the transmitter Serial Output (SO_{OUT}) is set to the Marking (logic 1) state; the receiver Serial Input (SIN) is disconnected; the output of the Transmitter Shift Register is "looped back" into the Receiver Shift Register input; the four MODEM Control inputs ($\overline{\text{CTS}}$, $\overline{\text{DSR}}$, $\overline{\text{DCD}}$, and $\overline{\text{RI}}$) are disconnected; and the four MODEM Control outputs ($\overline{\text{DTR}}$, $\overline{\text{RTS}}$, $\overline{\text{OUT 1}}$, and $\overline{\text{OUT 2}}$) are internally connected to the four MODEM Control inputs, and the MODEM Control output pins are forced to their inactive state (high). In the diagnostic mode, data that is transmitted is immediately received. This feature allows the processor to verify the transmit and received data paths of the UM8250A.

In the diagnostic mode, the receiver and transmitter interrupts are fully operational. The MODEM Control Interrupts are also operational, but the interrupts' sources are now the lower four bits of the MODEM Control Register instead of the four MODEM Control inputs. The interrupts are still controlled by the Interrupt Enable Register.

The UM8250A interrupt system can be tested by writing into the lower six bits of the Line Status Register and the lower four bits of the MODEM Status Register. Setting any of these bits to a logic 1 generates the appropriate interrupt (if enabled). The resetting of these interrupts is the same as in normal UM8250A operation. To return to normal operation, the registers must be reprogrammed for normal operation and then bit 4 of the MODEM control Register must be reset to logic 0.

Bits 5 through 7: These bits are permanently set to logic 0.

MODEM Status Register

This 8-bit register provides the current state of the control lines from the MODEM (or peripheral device) to the CPU. In addition to this current-state information, four bits of the MODEM Status Register provide change information. These bits are set to a logic 1 whenever a control input from the MODEM changes state. They are reset to logic 0 whenever the CPU reads the MODEM Status Register.

The contents of the MODEM Status Register are indicated in Table 3 and are described below.

Bit 0: This bit is the Delta Clear to Send (DCTS) indicator. Bit 0 indicates that the $\overline{\text{CTS}}$ input to the chip has changed state since the last time it was read by the CPU.

Bit 1: This bit is the Delta Data Set Ready (DDSR) indicator. Bit 1 indicates that the $\overline{\text{DSR}}$ input to the Chip has changed states since the last time it was read by the CPU.

Bit 2: This bit is the Trailing Edge of Ring Indicator (TERI) detector. Bit 2 indicates that the $\overline{\text{RI}}$ input to the chip has changed from a low state to a high state.

Bit 3: This bit is the Delta Received Line Signal Detector (DRLSD) indicator. Bit 3 indicates the $\overline{\text{RLSD}}$ input to the chip has changed states.

*Note that whenever bit 0, 1, 2 or 3 is set at logic 1, a MODEM Status interrupt is generated.

Bit 4: This bit is the complement of the Clear to Send ($\overline{\text{CTS}}$) input. If bit 4 (loop) of the MCR is set to a 1, this bit is equivalent to $\overline{\text{RTS}}$ in the MCR.

Bit 5: This bit is the complement of the Data Set Ready (DSR) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to $\overline{\text{DTR}}$ in the MCR.

Bit 6: This bit is the complement of the Ring Indicator ($\overline{\text{RI}}$) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to $\overline{\text{OUT 1}}$ in the MCR.

Bit 7: This bit is the complement of the Received Line Signal Detect ($\overline{\text{RLSD}}$) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to $\overline{\text{OUT 2}}$ of the MCR.



UM8250A

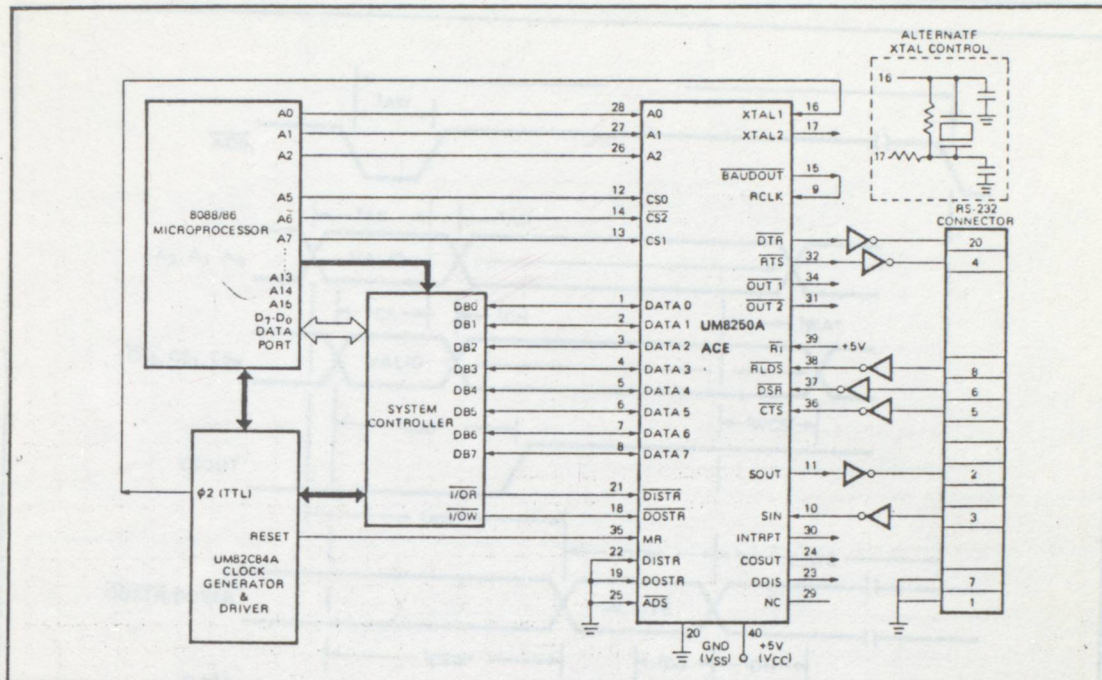
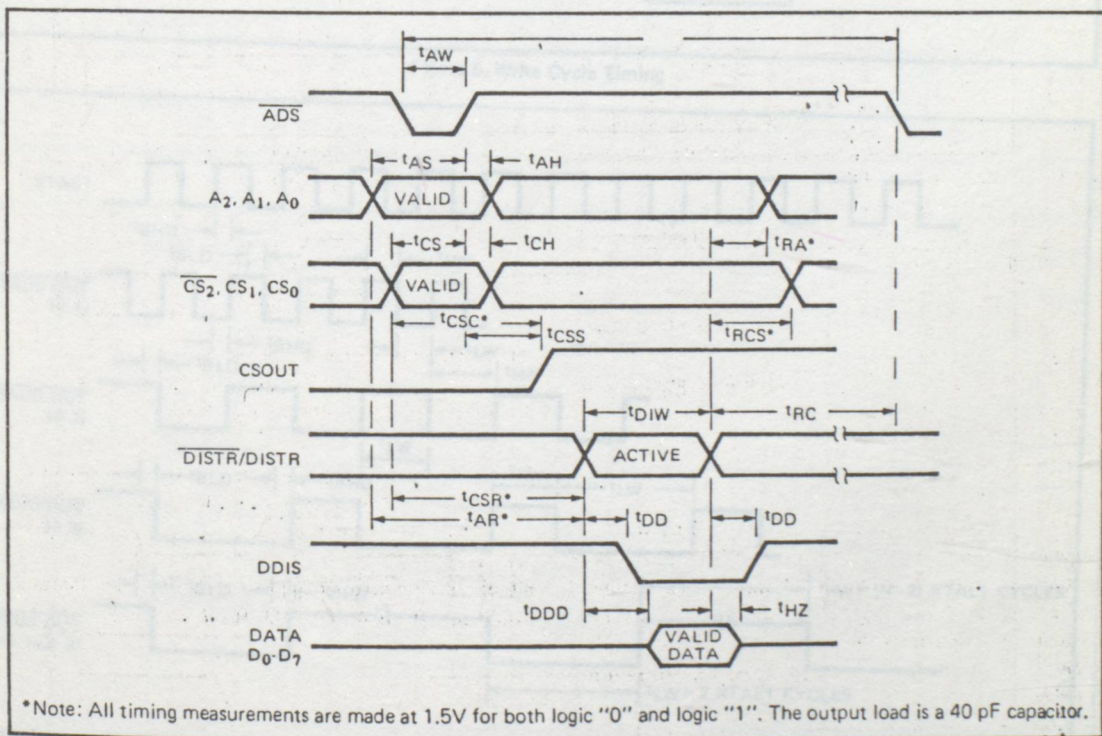


Figure 3.



*Note: All timing measurements are made at 1.5V for both logic "0" and logic "1". The output load is a 40 pF capacitor.

Figure 4. Read Cycle Timing



UM8250A

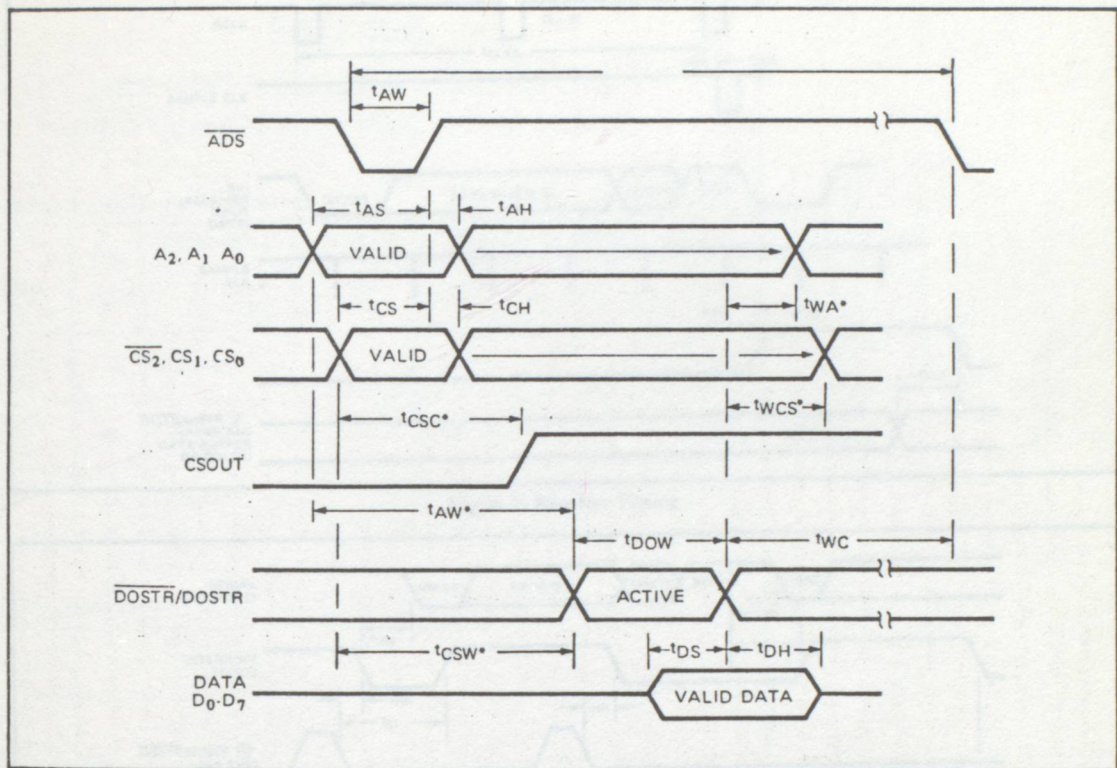


Figure 5. Write Cycle Timing

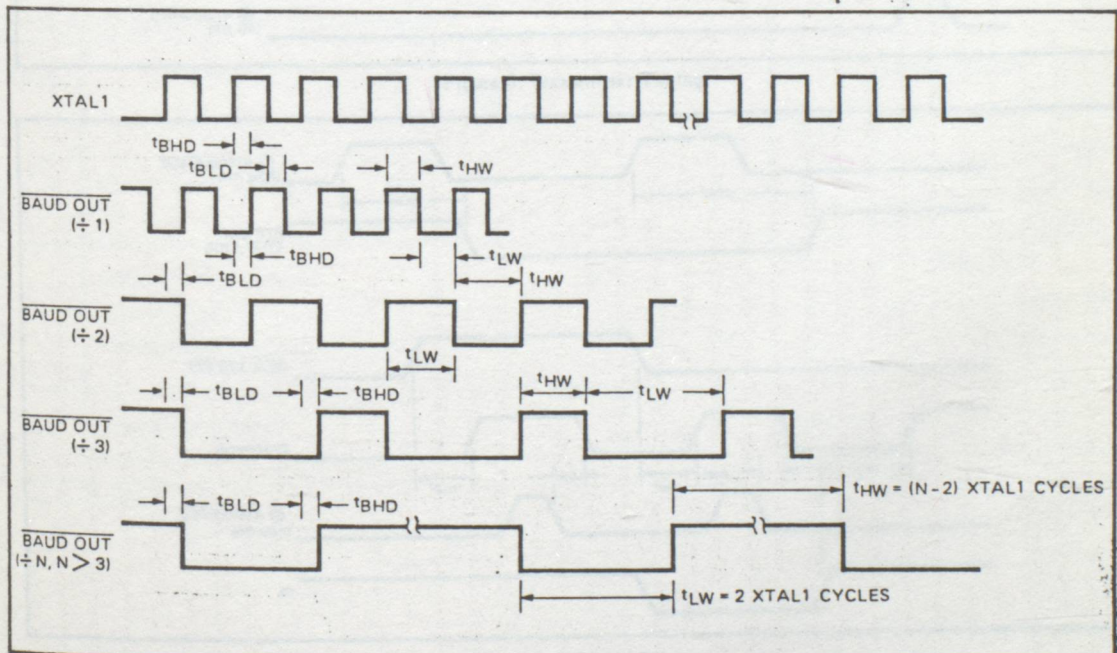


Figure 6. BAUDOUT Timing



UM8250A

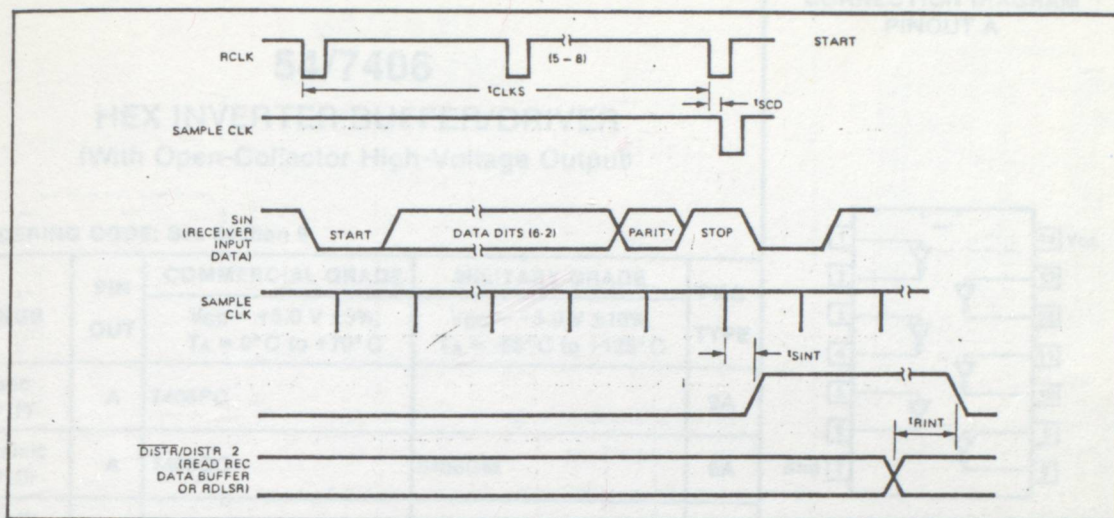


Figure 7. Receiver Timing

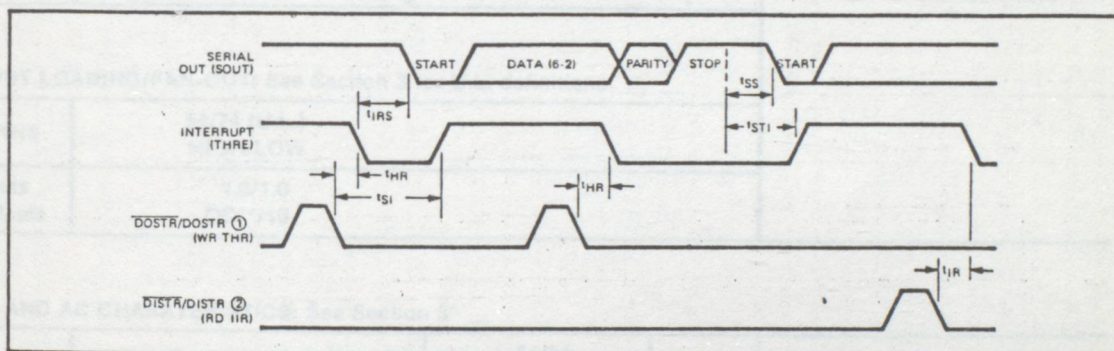


Figure 8. Transmitter Timing

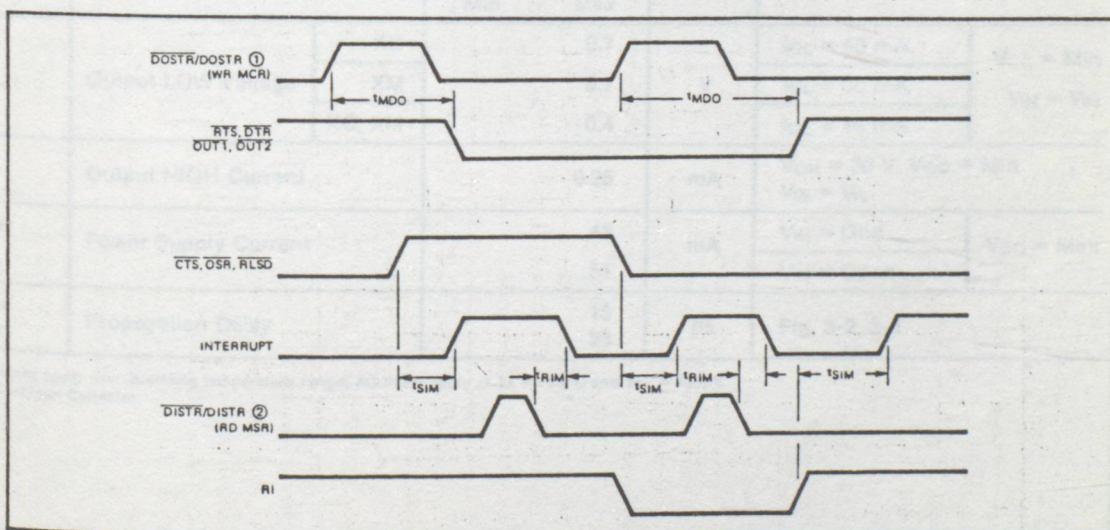


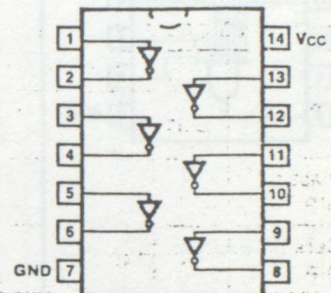
Figure 9. MODEM Timing

54/7406

HEX INVERTER BUFFER/DRIVER

(With Open-Collector High-Voltage Output)

ORDERING CODE: See Section 9

CONNECTION DIAGRAM
PINOUT A

PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		$V_{CC} = +5.0 \text{ V} \pm 5\%$ $T_A = 0^\circ \text{C to } +70^\circ \text{C}$	$V_{CC} = +5.0 \text{ V} \pm 10\%$ $T_A = -55^\circ \text{C to } +125^\circ \text{C}$	
Plastic DIP (P)	A	7406PC		9A
Ceramic DIP (D)	A	7406DC	5406DM	6A
Flatpak (F)	A	7406FC	5406FM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW
Inputs	1.0/1.0
Outputs	OC**/10

DC AND AC CHARACTERISTICS: See Section 3*

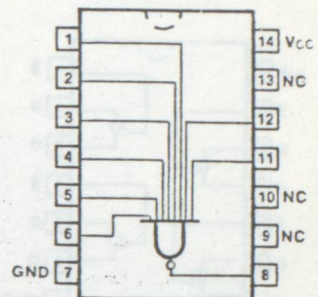
SYMBOL	PARAMETER		54/74		UNITS	CONDITIONS	
			Min	Max			
VOL	Output LOW Voltage	XC	0.7		V.	IOL = 40 mA	VCC = Min VIN = VIH
		XM	0.7			IOL = 30 mA	
		XC, XM	0.4			IOL = 16 mA	
IOH	Output HIGH Current		0.25		mA	VOH = 30 V, VCC = Min VIN = VIL	
ICCH	Power Supply Current		48		mA	VIN = Gnd	VCC = Max
ICCL			51			VIN = Open	
tPLH	Propagation Delay		15		ns	Fig. 3-2, 3-4	
tPHL			23				

*DC limits apply over operating temperature range; AC limits apply at $T_A = +25^\circ \text{C}$ and $V_{CC} = +5.0 \text{ V}$.

**OC—Open Collector

54/7430
54H/74H30
54S/74S30
54LS/74LS30
 8-INPUT NAND GATE

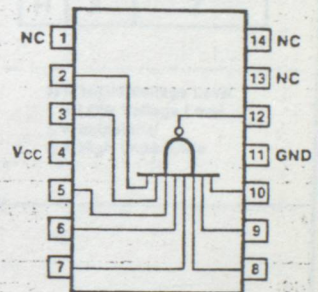
CONNECTION DIAGRAMS
PINOUT A



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		$V_{CC} = +5.0 \text{ V} \pm 5\%$, $T_A = 0^\circ\text{C to } +70^\circ\text{C}$	$V_{CC} = +5.0 \text{ V} \pm 10\%$, $T_A = -55^\circ\text{C to } +125^\circ\text{C}$	
Plastic DIP (P)	A	7430PC, 74H30PC 74S30PC, 74LS30PC		9A
Ceramic DIP (D)	A	7430DC, 74H30DC 74S30DC, 74LS30DC	5430DM, 54H30DM 54S30DM, 54LS30DM	6A
Pinpak (P)	A	74S30FC, 74LS30FC	54S30FM, 54LS30FM	3I
	B	7430FC, 74H30FC	5430FM, 54H30FM	

PINOUT B



INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74H (U.L.) HIGH/LOW	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	1.25/1.25	1.25/1.25	0.5/0.25
Outputs	20/10	12.5/12.5	25/12.5	10/5.0 (2.5)

DC AND AC CHARACTERISTICS: See Section 3*

SYMBOL	PARAMETER	54/74		54/74H		54/74S		54/74LS		UNITS	CONDITIONS	
		Min	Max	Min	Max	Min	Max	Min	Max			
I_{CC}	Power Supply Current	2.0	6.0	4.2	10	5.0	10	0.5	1.1	mA	$V_{IN} = \text{Gnd}$	$V_{CC} = \text{Max}$
											$V_{IN} = \text{Open}$	
t_{PLH} t_{PHL}	Propagation Delay	22 15		10 12		2.0 2.0	6.0 7.0	12 20		ns	Figs. 3-1, 3-4	

* Limits apply over operating temperature range; AC limits apply at $T_A = +25^\circ\text{C}$ and $V_{CC} = +5.0 \text{ V}$.

54/74125
54LS/74LS125A
 QUAD BUS BUFFER GATE
 (With 3-State Outputs)

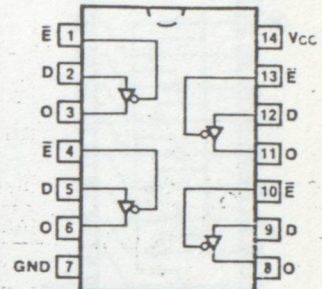
ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		$V_{CC} = +5.0 \text{ V} \pm 5\%$, $T_A = 0^\circ \text{C to } +70^\circ \text{C}$	$V_{CC} = +5.0 \text{ V} \pm 10\%$, $T_A = -55^\circ \text{C to } +125^\circ \text{C}$	
Plastic DIP (P)	A	74125PC, 74LS125APC		9A
Ceramic DIP (D)	A	74125DC, 74LS125ADC	54125DM, 54LS125ADM	6A
Flatpak (F)	A	74125FC, 74LS125AFC	54125FM, 54LS125AFM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	0.5/0.25
Outputs	130/10 (50)	65/15 (25)/(7.5)

CONNECTION DIAGRAM
PINOUT A*



TRUTH TABLE

INPUTS		OUTPUT
$\bar{E}$	D	
L	L	L
L	H	H
H	X	Z

H = HIGH Voltage Level
 L = LOW Voltage Level
 X = Immaterial
 Z = High Impedance

DC AND AC CHARACTERISTICS: See Section 3*

SYMBOL	PARAMETER		54/74		54/74LS		UNITS	CONDITIONS
			Min	Max	Min	Max		
V_{OH}	Output HIGH Voltage	XM	2.4				V	$I_{OH} = -2.0 \text{ mA}$
		XC	2.4					$I_{OH} = -5.2 \text{ mA}$
		XM			2.4			$I_{OH} = -1.0 \text{ mA}$
		XC			2.4			$I_{OH} = -2.6 \text{ mA}$
I_{OS}	Output Short Circuit Current	XM	-30	-70	-30	-130	mA	$V_{CC} = \text{Max}$
		XC	-28	-70	-30	-130		
I_{CC}	Power Supply Current		54		20		mA	Outputs OFF, $V_{IN} = \text{Gnd}$ $V_E = 4.5 \text{ V}$, $V_{CC} = \text{Max}$
t_{PLH} t_{PHL}	Propagation Delay Data to Output		13 18		15 18		ns	Figs. 3-3, 3-5
t_{PZH} t_{PZL}	Output Enable Time		17 25		16 25			
t_{PLZ} t_{PHZ}	Output Disable Time		8.0 12		25 25		ns	Figs. 3-3, 3-11, 3-12

*DC limits apply over operating temperature range; AC limits apply at $T_A = +25^\circ \text{C}$ and $V_{CC} = +5.0 \text{ V}$.

240 • 241 • 244

240 • 241 • 244

54S/74S240 • 54LS/74LS240 54S/74S241 • 54LS/74LS241 54LS/74LS244

OCTAL BUFFER/LINE DRIVER
(With 3-State Outputs)

DESCRIPTION — The '240, '241 and '244 are octal buffers and line drivers designed to be employed as memory address drivers, clock drivers and bus oriented transmitters/receivers which provide improved PC board density.

- HYSTERESIS AT INPUTS TO IMPROVE NOISE MARGINS
- 3-STATE OUTPUTS DRIVE BUS LINES OR BUFFER MEMORY ADDRESS REGISTERS
- OUTPUTS SINK 24 mA (74LS) OR 40 mA (74S)
- 15 mA SOURCE CURRENT
- INPUT CLAMP DIODES LIMIT HIGH SPEED TERMINATION EFFECTS
- FULLY TTL AND CMOS COMPATIBLE

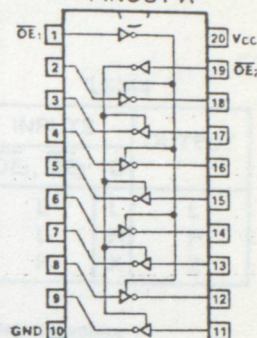
ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		$V_{CC} \equiv +5.0 \text{ V}, \pm 5\%$ $T_A = 0^\circ \text{C to } +70^\circ \text{C}$	$V_{CC} = +5.0 \text{ V}, \pm 10\%$ $T_A = -55^\circ \text{C to } +125^\circ \text{C}$	
Plastic DIP (P)	A	74S240PC, 74LS240PC		9Z
	B	74S241PC, 74LS241PC		
	C	74LS244PC		
Ceramic DIP (D)	A	74S240DC, 74LS240DC	54S240DM, 54LS240DM	4E
	B	74S241DC, 74LS241DC	54S241DM, 54LS241DM	
	C	74LS244DC	54LS244DM	
Flatpak (F)	A	74S240FC, 74LS240FC	54S240FM, 54LS240FM	4F
	B	74S241FC, 74LS241FC	54S241FM, 54LS241FM	
	C	74LS244FC	54LS244FM	

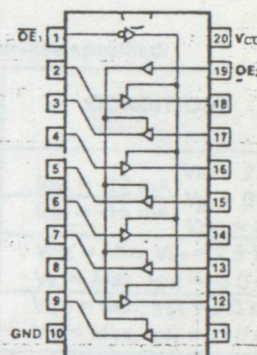
INPUT LOADING/FAN-OUT: See Section 9

PIN NAMES	DESCRIPTION	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
$\overline{OE}_1, \overline{OE}_2$	3-State Output Enable (Active LOW)	1.25/1.25	0.5/0.25
OE_2	3-State Output Enable (Active HIGH)	1.25/1.25	0.5/0.25
	Inputs	1.25/0.25	0.5/0.125
	Outputs	75/40 (30)	75/15 (7.5)

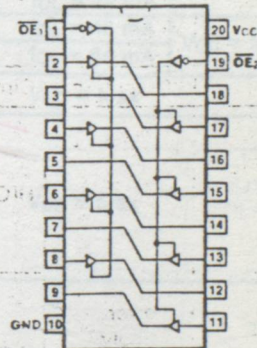
CONNECTION DIAGRAMS
PINOUT A



PINOUT B



PINOUT C



240 • 241 • 244

240 • 241 • 244

TRUTH TABLES

'S240, 'LS240

INPUTS		OUTPUT
$\overline{OE}_1, \overline{OE}_2$	D	
L	L	H
L	H	L
H	X	Z

'S241, 'LS241

INPUTS			OUTPUT
$\overline{OE}_1$	$\overline{OE}_2$	D	
L	H	L	L
L	H	H	H
H	L	X	Z

'LS244

INPUTS		OUTPUT
$\overline{OE}_1, \overline{OE}_2$	D	
L	L	L
L	H	H
H	X	Z

H = HIGH Voltage Level

L = LOW Voltage Level

X = Immaterial

Z = High Impedance

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

SYMBOL	DESCRIPTION		54/74S		54/74LS		UNITS	CONDITIONS	
			Min	Max	Min	Max			
V _{OH}	Output HIGH Voltage	XM	2.0		2.0		V	I _{OH} = -12 mA	V _{IH} = 2.0 V
		XC	2.0		2.0			I _{OH} = -15 mA	V _{IL} = 0.5 V
									V _{CC} = Min
		XM	2.4		2.4		V	V _{CC} = Min, V _{IH} = 2.0 V	
		XC	2.4		2.4			V _{IL} = Max, I _{OH} = -3.0 mA	
V _{OL}	Output LOW Voltage	XM				0.4	V	I _{OL} = 12 mA	V _{CC} = Min
		XC				0.4		I _{OL} = 12 mA	
		XC				0.5		I _{OL} = 24 mA	
		XM		0.55			V	I _{OL} = 48 mA	V _{CC} = Min
		XC		0.55				I _{OL} = 64 mA	
I _{OS}	Output Short Circuit Current		-50	-225	-40	-225	mA	V _{CC} = Max	
I _{CC}	Power Supply Current	HIGH	('240) XM	123		23	mA	V _{CC} = Max	
			XC	135		23			
			('241) XM	147		23			
			XC	160		23			
		LOW	('244) XM			23	mA	V _{CC} = Max	
			XC			23			
			('240) XM	145		44			
			XC	150		44			
		OFF	('241) XM	170		46	mA	V _{CC} = Max	
			XC	180		46			
			XM			46			
			XC			46			
			('240) XM	145		50	mA	V _{CC} = Max	
			XC	150		50			
			('241) XM	170		54			
			XC	180		54			
			XM			54			
			XC			54			

240 • 241 • 244

AC CHARACTERISTICS: $V_{CC} = +5.0\text{ V}$, $T_A = +25^\circ\text{C}$ (See Section 3 for waveforms and load configurations)

SYMBOL	PARAMETER	54/74S		54/74LS		UNITS	CONDITIONS
		$C_L = 50\text{ pF}$ $R_L = 90\ \Omega$		$C_L = 50\text{ pF}$			
		Min	Max	Min	Max		
t_{PLH} t_{PHL}	Propagation Delay Data to Output ('240)	7.0	7.0	14	18	ns	Figs. 3-1, 3-4
t_{PLH} t_{PHL}	Propagation Delay Data to Output ('241)	9.0	9.0	18	18	ns	Figs. 3-1, 3-5
t_{PLH} t_{PHL}	Propagation Delay Data to Output ('244)			18	18	ns	
t_{PZH} t_{PZL}	Output Enable Time ('S240)	10	15			ns	Figs. 3-3, 3-11, 3-12
t_{PZH} t_{PZL}	Output Enable Time ('LS240, 'LS241, 'S241)	12	15	23	30	ns	Figs. 3-3, 3-11, 3-12 $R_L = 667\ \Omega$ ('LS)
t_{PLZ} t_{PHZ}	Output Disable Time	15	9.0	25	18	ns	Figs. 3-3, 3-11, 3-12 $R_L = 667\ \Omega$, $C_L = 5\text{ pF}$ ('LS)

54LS/74LS245

OCTAL BUS TRANSCEIVER

(With 3-State Outputs)

DESCRIPTION — The 'LS245 is an octal bus transmitter/receiver designed for 8-line asynchronous 2-way data communication between data busses. Direction input (DR) controls transmission of data from bus A to bus B or bus B to bus A depending upon its logic level. The Enable input ($\bar{E}$) can be used to isolate the busses.

- HYSTERESIS INPUTS TO IMPROVE NOISE IMMUNITY
- 2-WAY ASYNCHRONOUS DATA BUS COMMUNICATION
- INPUT DIODES LIMIT HIGH SPEED TERMINATION EFFECTS
- FULLY TTL AND CMOS COMPATIBLE

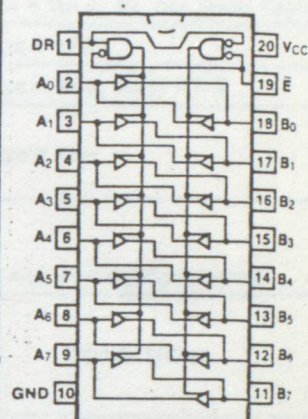
ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE	MILITARY GRADE	PKG TYPE
		$V_{CC} = +5.0 \text{ V} \pm 5\%$, $T_A = 0^\circ \text{C to } +70^\circ \text{C}$	$V_{CC} = +5.0 \text{ V} \pm 10\%$, $T_A = -55^\circ \text{C to } +125^\circ \text{C}$	
Plastic DIP (P)	A	74LS245PC		9Z
Ceramic DIP (D)	A	74LS245DC	54LS245DM	4E
Flatpak (F)	A	74LS245FC	54LS245FM	4F

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74LS (U.L.) HIGH/LOW
Inputs	0.5/0.125
Outputs	75/15 (7.5)

CONNECTION DIAGRAM
PINOUT A



TRUTH TABLE

INPUTS		OUTPUT
$\bar{E}$	DR	
L	L	Bus B Data to Bus A
L	H	Bus A Data to Bus B
H	X	Isolation

H = HIGH Voltage Level
L = LOW Voltage Level
X = Immaterial

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

SYMBOL	PARAMETER		54/74LS		UNITS	CONDITIONS
			Min	Max		
V_{OH}	Output HIGH Voltage	XM	2.0		V	$V_{CC} = \text{Min}$ $V_{IN} = V_{IH}$ or V_{IL} per Truth Table
		XC	2.0			
V_{OH}	Output HIGH Voltage		2.4		V	$I_{OH} = -3.0 \text{ mA}$, $V_{CC} = \text{Min}$ $V_{IN} = V_{IH}$ or V_{IL} per Truth Table
$V_T - V_T$	Hysteresis Voltage		0.2		V	$V_{CC} = \text{Min}$
I_{OS}	Output Short Circuit Current		-40	-225	mA	$V_{CC} = \text{Max}$, $V_{OUT} = 0 \text{ V}$
I_{CC}	Power Supply Current	HIGH		70	mA	$V_{CC} = \text{Max}$
		LOW		90		
		OFF		95		

AC CHARACTERISTICS: $V_{CC} = +5.0 \text{ V}$, $T_A = +25^\circ \text{C}$ (See Section 3 for waveforms and load configurations)

SYMBOL	PARAMETER	54/74LS		UNITS	CONDITIONS
		$C_L = 45 \text{ pF}$			
		Min	Max		
t_{PLH} t_{PHL}	Propagation Delay Data to Output		18 18	ns	Figs. 3-1, 3-5
t_{PZH} t_{PZL}	Output Enable Time		25 30	ns	Figs. 3-3, 3-11, 3-12 $R_L = 667 \Omega$
t_{PLZ} t_{PHZ}	Output Disable Time		25 18	ns	Figs. 3-3, 3-11, 3-12 $R_L = 667 \Omega, C_L = 5 \text{ pF}$

DZGEÇMİŞİM

1962 yılında İSTANBUL ili BEYOĞLU ilçesinde dünyaya geldim. İlk, orta ve lise tahsilimi İSTANBUL'daki çeşitli okullarda tamamladım. 1980 yılında o zamanki adıyla İSTANBUL DEVLER MİMARLIK VE MÜHENDİSLİK AKADEMİSİ ELEKTRİK FAKÜLTESİ'ne girdim. 1982 yılında YILDIZ ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ adını alan bu fakültenin ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ bölümünden 1985 yılında mezun oldum. 1986 yılında YILDIZ ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜNÜN Sayısal Elektronik konusunda açtığı yüksek lisans giriş sınavını kazanarak bu bölüme girdim. Halen aynı bölümde öğrenimime devam etmekteyim.

Müh. Üzeyir KIZILARMUT



