

. i.T.Ü.

ELEKTRİK FAKÜLTESİ

MMLS DİPLOMA TEZİ

LOJİK ANALİZÖRLER

VE

8 KANALI BİR ANALİZÖRÜN

GELİŞTİRİLMESİ

Yöneten : Prof. Dr. Yılmaz Tokad

Murat Demirer

İstanbul 1982

Bu çalışmanın yürütülmesi sırasında laboratuvar, araç ve gereç gibi olanaklarından sınırsız yararlanabilmemi sağlayan Türkiye Bilimsel ve Teknik Araştırma Kurumu - TÜBİTAK Başkanlığına ve çok değerli zamanlarını esirgemeyen, bilgi ve tecrübelerinden geniş ölçüde yararlanabilmeme izin veren, başta TÜBİTAK Elektronik Bölümü Başkanı Sayın Prof.Dr. Yılmaz Tokad ve Sayın Yük.Müh. (BS,MS) Atok Karaali'ye sonsuz şükranlarımı arz eder, en derin saygılarımı sunarım.

ÖZET

Tezde ayrıntılı olarak incelenen ve geliştirilen lojik analizör cihazının, sayısal sistemlerdeki arıza aramaları sırasında, arıza belirtilerini tespit etmek açısından çok değerli bir araç olduğu açıklanmıştır. Tez'de ayrıca, gerçekleştirme aşamasında olan yeni sayısal cihazların, özellikle mikro işlemci tabanlı cihazların, tasarımlarında, lojik analizörlerinden yararlanılmasının artık belirgin bir düzeye erişmiş olduğu vurgulanmıştır. 1960'larda kullanılan ve o zamanlar için uygun görülen klasik teknikler, frekans domeni (frequency domain) ve zaman domeni (time domain) türünde yaklaşımlar olup Fourier ve Laplace dönüşümleri temeline dayandırılmışlardır. Çeşitli voltmetreler, osiloskoplar, sayıcılar ve spektrum analizörleri bu türden cihazlardır.

Tümdevrelerin ve minibilgisayarların ortaya çıkması, bilgisayar tabanlı sistemlerin tasarım imkânlarının, çok belirgin bir düzeyde gelişmesine neden olmuştur. Başlangıçta tasarımcılar ve arıza arayıcıları, yazılım (software) araştırmalarının, ani sıçramaların (glitch) tesbit edilmesi veya gürültü düzeyinin kontrolü için büyük gayret sarfetmişlerdir. İşte lojik analizörler bu zorlukları ortadan kaldırmak üzere yeni bir sınıf cihaz olarak meydana çıkmıştır.

Bir bilgisayarda hatalar, genellikle veri akışında (data flow) oluşan düzensizliklerden doğmaktadır. Bunlar, yanlış veri dizisi (incorrect data sequence) nedeni ile oluşan hatalar olup bu tür hataların analizi için en uygun tür cihazlar da veri domeni analizi (data domain analysis) cihazlarıdır. Başlangıçta lojik analizörler sayısal skoplar (digital scopes) olarak düşünülmüştür. Bu konudaki çalışmalar, kelime ve olayın bir

İÇİNDEKİLER

1. LOJİK ANALİZÖRLERİNDEKİ ZAMANIMIZDA VE GELECEKTEKİ EĞİLİMLER
 - 1.1. GİRİŞ
 - 1.2. GİRİŞTE "PROBE" UYGULAMASI
 - 1.3. ANALİZÖRÜN UZAKTAN ARIZA BULMA ARACI OLARAK KULLANILMASI
2. BİR ARIZA BULMA CİHAZI OLARAK LOJİK ANALİZÖR
 - 2.1. GENEL ÖZELLİKLER
 - 2.2. LOJİK ZAMANLAMA ANALİZÖRLERİ İLE LOJİK DURUM ANALİZÖRLERİNİN KARŞILAŞTIRILMASI
 - 2.3. TETİKLEME ÖNCESİ DURUMU
 - 2.4. M 6800 MİKROBİLGİSAYAR SİSTEMİNDE KARŞILAŞILAN BİR ARIZANIN LOJİK ANALİZÖR İLE BULUNMASI
 - 2.5. LOJİK ANALİZÖRLERİN DURUMU
 - 2.6. TETİKLEME ÖNCESİ KAYDETME YETENEĞİ
 - 2.7. KOMBİNEZONSAL TETİKLEME
3. SAYISAL SİSTEMLERDE ÖLÇMELER
 - 3.1. GİRİŞ
 - 3.2. İŞLEM YAPICI (PROCESSOR) TABANLI SİSTEMLER
 - 3.3. OSİLOSKOPLARIN SINIRLANDIĞI NOKTALAR
 - 3.4. GEREKLİ SAYISAL SİSTEM ÖLÇÜMLERİ
4. LOJİK DURUM ANALİZÖRLERİ
 - 4.1. GİRİŞ
 - 4.2. LOJİK DURUM ANALİZÖRLERİ
 - 4.3. EKРАНLAR
 - 4.4. SAAT KONTROLLARI
 - 4.5. TETİKLEME
 - 4.6. İZLENENİN SEÇİMİ
 - 4.7. BİRBİRİNİ ETKİLEYEN OSİLOSKOP / ANALİZÖR ÖLÇMELERİ
 - 4.8. BİLGİSAYAR SİSTEMLERİNİN GÖZLENMESİ
 - 4.9. SONUÇ

5. LOJİK ZAMAN ANALİZÖRLERİ

5.1. GİRİŞ

5.2. ZAMAN ANALİZÖRÜNÜN AYIRIM YETENEĞİ VE KANALLAR ARASINDAKİ
GECİKME FARKLILIKLARI

5.3. ZAMAN ANALİZÖRÜNÜN ASENKRON OLARAK TETİKLENMESİ

5.4. TEST İŞARETİNİN ÖZELLİKLERİ

5.5. SONUÇ

6. TEZDE GELİŞTİRİLEN LOJİK ANALİZÖR

6.1. LOJİK ANALİZÖRÜN TEMEL ÇALIŞMA PRENSİBİ

6.2. TEZDE GELİŞTİRİLEN LOJİK ANALİZÖRÜN ÇALIŞTIRILMASINDA
BAZI AYRINTILAR

6.3. SONUÇ

EK. A. KISA SÜRELİ SIĞRAMLARI (GLITCH) TESPİT EDİCİ VE GENİŞLETİCİ
DEVRE

EK. B. REFERANSLAR

EK. C. LOJİK ANALİZÖRÜN BASİTLEŞTİRİLMİŞ DİYAGRAMI

EK. D. TEZ'DE GELİŞTİRİLEN LOJİK ANALİZÖRÜN AYRINTILI DİYAGRAMI

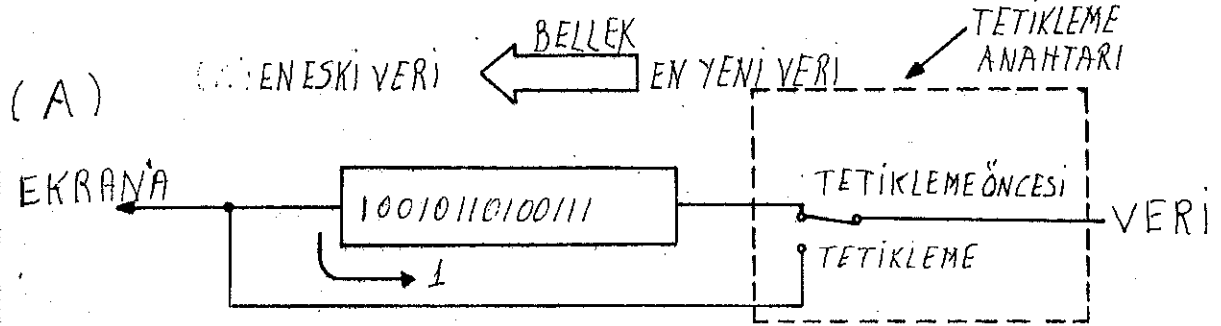
BİRİNCİ BÖLÜM

LOJİK ANALİZÖRLERİNDEKİ ZAMANIMIZDA VE GELECEKTEKİ BÜYÜMLER

1.1. GİRİŞ

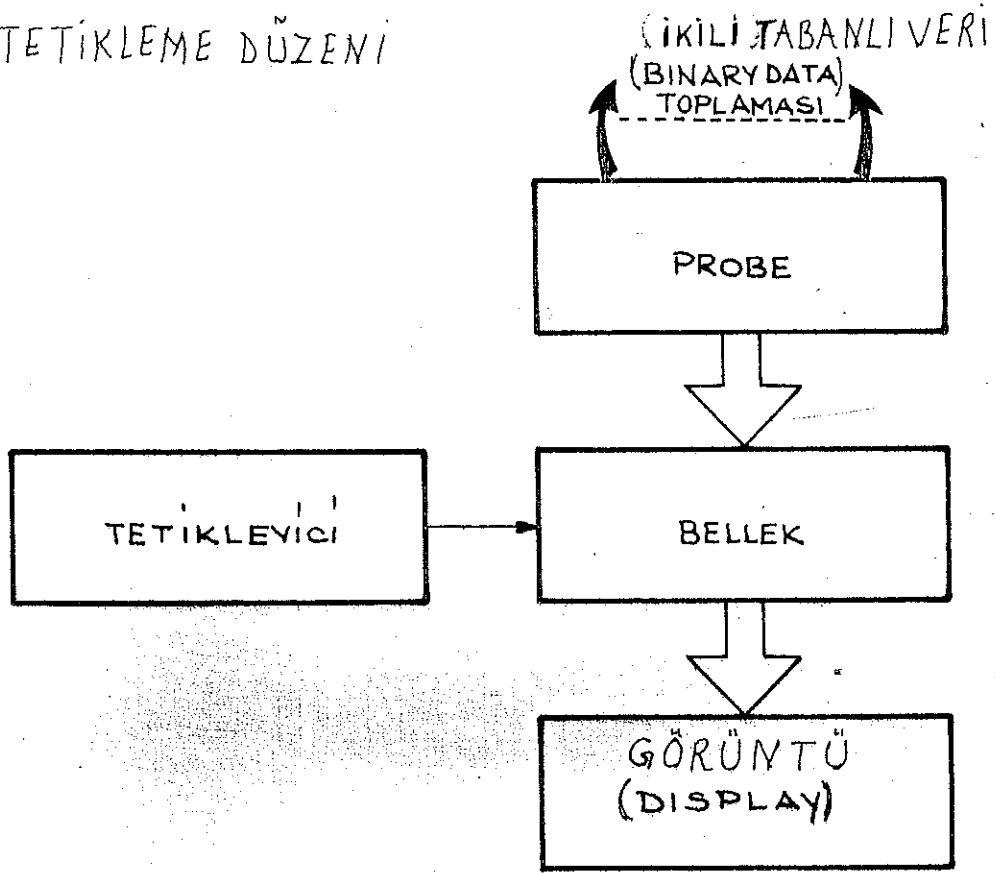
1970 yıllarının başlangıcında sayısal alandaki gelişmeler büyük hız kazanmıştır. Lojik analizörlerinden önce kullanılmakta olan osiloskoplar, depolanan verileri gözle izleyebilmek üzere skop çoklayıcı (scope multiplexer) ile birleştirilmiştir. Bu veriler osiloskopta "1" ler ve "0" lar olarak gösterilmek sureti ile, deney altındaki veri akışına eşdeğer kelimeler sağlamıştır. Deney altındaki sistem, bu verileri senkron olarak kaydetmek üzere, saat darbeleri üretmektedir. Ön panel anahtarınca uygulanan bir tetikleme kelimesi, hangi verilerin elde edileceğini belirler. Bu kelime tetiklendikten sonra, tetikleme kelimesinin her iki yanındaki veri kelimeleri gözlenebilir. Pek çok lojik analizörler, halen osiloskop/çoklayıcı birleşimi ve temel kelime tetikleme tekniğini kullanırlar. Şekil 1-1 de temel tetikleme kavramı görülmektedir.

Osiloskobun analog dünyasındaki anlamı ne ise, lojik analizörün de sayısal dünyadaki anlamı ona eşdeğerdir. Şu anda piyasada çeşitli özelliklerde lojik analizörler mevcuttur. Bazan, kullanıcının sistemlerindeki araştırmalarında, hangi lojik analizörün daha belirgin olarak, yardımcı olacak özelliklere sahip olduğunu anlamakta zorluk çekebilir. Fiyat ve üretici firmanın markası, analizörün kullanılabilirlik derecesini belirtmede, performansı kadar önemli değildir. Bu husus akılda tutularak, bu çalışmada lojik analizörün kuram ve uygulaması açıklanmıştır.



TEMEL TETİKLEME DÜZENİ

(B)



Şekil 1-1(A) İkili veri (binary data) elde edilmiştir. Tetikleme için tanımlanan kelime ve elde edilen veri birbirine eşit olduğu zaman tetikleme olayı oluşur. Tetikleme kelimesi bellekte saklanır.

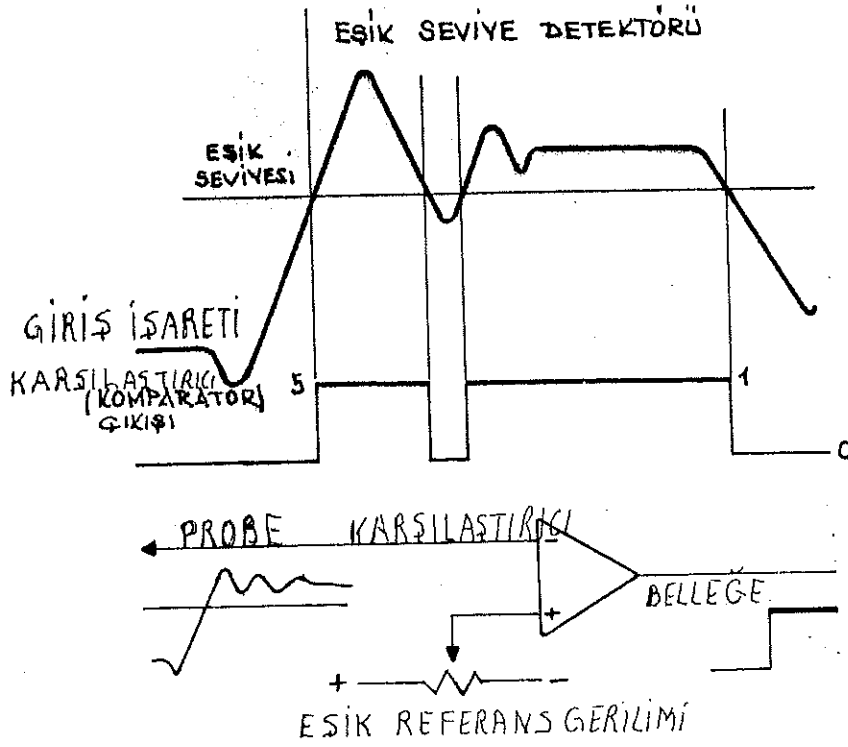
(B) Veri, probalar üzerinden elde edilir, tetikleme oluşur ve bellek, veriyi saklar. Bundan sonra, veri, görüntülenmek üzere şekillenir.

Lojik analizörlerinin tümünde bir bellek birimi bulunmaktadır. Bütün analizörler giriş verilerinden örnekler alarak ve bunları bellekte saklamak sureti ile bir dizi sayısal girişi arar, bulur, yakalar ve kaydeder. Değişik üretici firmalar tarafından, bellekler, izlenen her kanal için 64 bit'ten 2048 bit'e kadar değişik kapasitede olmak üzere üretilmektedir. Lojik analizör, örneklenmesi istenilen verileri depoladıktan sonra, bu verileri bir CRT (katot ışınlı tüp) li osiloskop aracılığı ile uygun bir şekilde görüntüler.

Aşağıda bir lojik analizörün çalışması biraz daha ayrıntılı olarak incelenmiştir.

Modern bir lojik analizör, çok kanallı verileri aynı anda yakalar. Bu verileri, bir ardışıl erişimli bellekte (linear) saklar. Bu veri saklama işlemini bir tetikleme ile durdurur. Bundan sonra, veriler şekillendirilerek değerlendirilme için görüntülendirilir (şekil 1-1-B). Bu suretle, verilerin tesbiti kullanıcıya bir hatayı veya tetikleme durumunu yakalama ve analiz edebilme olanağını sağlar. Bundan başka, bellek uzunluğuna bağlı olarak, hata veya tetikleme anına kadar oluşan veri akışı (pre-trigger data) veya tetiklemeden sonra birbiri ardına oluşan faaliyetler (post-trigger data) tesbit edilebilir. Modern analizörler, hem veri durumunu (data state) hemde donanım (hardware) ve yazılım (software) için zamanlama bilgisini (timing information) verirler.

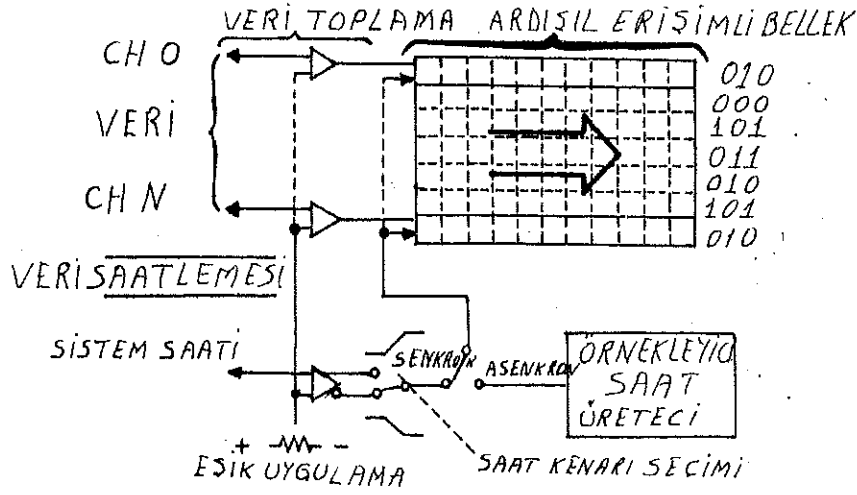
Verilerin elde edilmesi sırasında, hatalı okumaları önlemek için, giriş verileri daha fazla nitelenmelidir. Nitelemenin bir türü, veri elde edilmesinden önce, uygun bir eşik (threshold) seviyesinin seçimi biçiminde olur. Gelen verilerin "yüksek" veya "alçak" durumlarını ayırtmak için, her kanal bir gerilim karşılaştırıcısına (voltage comparator)



Şekil 1-2. Yukardaki diyagram, bir karşılaştırıcı (comparator) tekniği kullanmak sureti ile, eşik arayıcısını (threshold detector) göstermektedir.

sahiptir. Herhangi bir işaret gerilimi incelenirken, işaret gerilimi, karşılaştırıcının eşik geriliminden büyük ise, "yüksek" (lojik 1), işaret gerilimi, karşılaştırıcının eşik geriliminden düşük ise, "alçak" (lojik 0) olacaktır. Birçok üreticiler, farklı eşik gerilimi seviyesini seçebilmek için bir potansiyometre kullanmaktadır.

Lojik analizörler, veri örneklemelerini ayrık zamanlarda (discreet times) yaparlar. İkili durumlar (binary states), bellekte, bir saat işaretinin "yükselen" veya "düşen" kenarında (aktif olan kenarda) iken saklanırlar. Veriler senkron olarak örneklendiği zaman, test altındaki sistemde bulunan bir saat ile örneklenirler. (Şekil 1-3 ve Şekil 1-4)

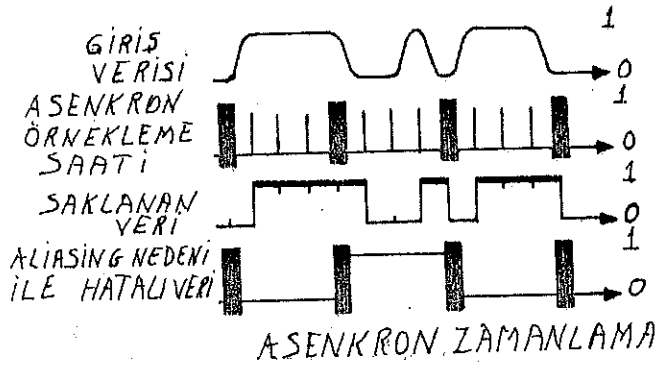
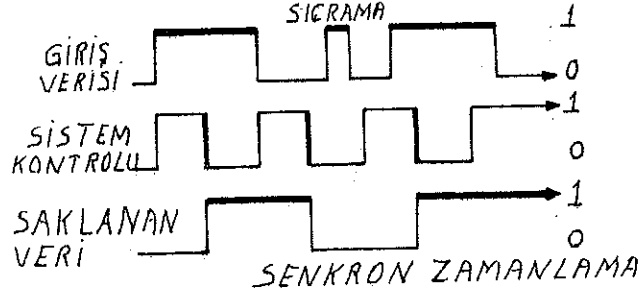


Şekil 1-3. Ardışıl giriş belleği (sequential access memory), basit bir ötelemeli yazıcı (shift registers) gurubu olarak gösterilmiştir. Pratikte, analizörlerin çoğu, bir sayıcı tarafından adreslenen yüksek hızlı RAMs kullanırlar. Bununla beraber çalışma aynıdır. Yeni veri saklanınca, eski veri kaybolur.

Buna senkron örnekleme (synchronous sampling) denilmektedir. Senkron örnekleme "glitch" olarak bilinen dar sıçrama darbelerini ihmal edip, sistemin ardışıl saat darbelerinde giriş kanalının durumlarını kaydeder. Lojik durum analizörleri, senkron olarak, lojik zaman analizörleri ise, asenkron olarak örnek alırlar.

Bazı mikrobilgisayar sistemlerinde, örneğin Z80'de, işlemci bir kapı (port) ile iletişim kurarken, sadece R/W (yaz/oku) verileri izlenmek istenebilir. Bundan başka, kullanıcı, Z80 belleği ile iletişim kurmak için, aynı anda oluşan R/W darbelerini kayıt etmek isteyebilir. Böyle bir durumda, dış bir tanımlayıcı kullanarak, veriler seçilip saklanabilir. Böyle ki; Bu tanımlayıcı lojik "1" durumuna seçildiği zaman belleğin R/W darbelerini ve lojik "0" durumuna seçildiği zaman kapının (port) R/W darbelerini yakalayabilir, "göz önüne almama" (don't care) durumunda

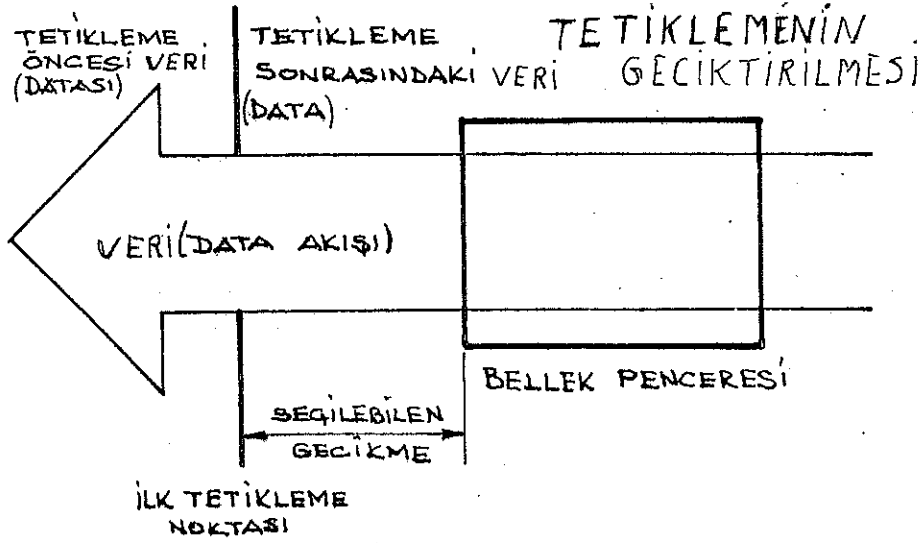
A SENKRON VERİ ÖRNEKLEMESİ



Şekil 1-4. Veri akışındaki durumlar, senkron ve asenkron olarak kaydedilmiş halde, gösterilmektedir. Asenkron olarak yakalanan sıçramalar (glitches), bir bit genişliğinde darbeler olarak belirmiştir.

İse hem bellek hemde kapının (port) R/W verilerinin yakalanmasına izin verecektir. Böyle bir tanımlayıcı (qualifier) kullanılarak, kullanıcının sadece seçilmiş bilgileri kayıt etmesi sağlanmış ve etkin olarak lojik analizörün, belleğini iki katına çıkarmasına izin vermiş olunur.

Lojik zamanlama bilgilerini toplamak için, lojik analizörün içinde bir saat (clock) kullanmak sureti ile, asenkron olarak veri alabiliriz. Bu saat ile iyi bir ayırım yeteneği (resolution) sağlamak için, deney altındaki sistemden 3-10 kat daha hızla veri örneklenebilmelidir. Eğer lojik zamanlama analizörünün saati, sistem saatinden daha yavaş ise



Şekil 1-5. Belirlenmiş bir tetikleme kelimesi kullanarak, verinin herhangi bir andaki durumu, saat sayımları (clock counts) vasıtası ile istenildiği kadar gecikme sağlanabilir. Bir dizi gecikme uygulayarak, veri akışı boyunca, veri bloklarının herhangi bir andaki durumları tesbit edilebilir.

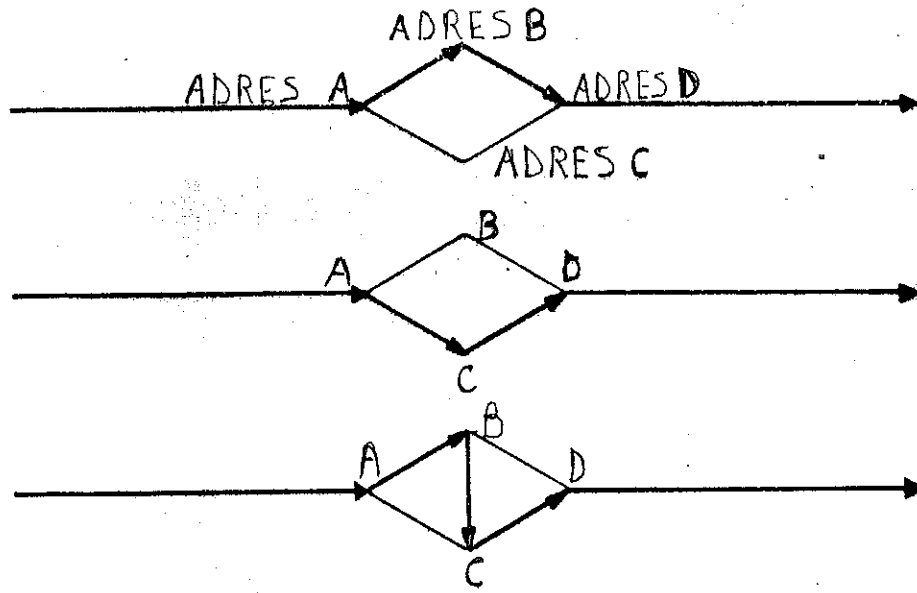
"aliasing" meydana gelecektir. Lojik zaman analizörlerinde, en az darbe genişliği kriterini sağlamak için oluşturulan çok dar darbeler, bir "glitch latch" yardımı ile tesbit edilebilirler. Örneklem darbeleri arasında meydana gelen durum değişimleri (transition), bir sonraki saat periodunda, bu periyod boyunca darbeler olarak görünürler (şekil 1-4).

Daha önce belirtildiği üzere, tetikleme kelimesi veri yakalama işlemini durdurur, ve analizöre bu verileri saklaması için bilgi verir. Çoğu analizörler, tetikleme kelimesinden önceki (pre-trigger) olayları veya tetikleme kelimesi sonrası (post trigger) olaylarını göstermeyi

sağlayan değişken bir tetikleme noktasına sahiptirler. Duruma bağlı olarak, kullanıcı, tetiklemeden hemen önceki veya tetikleme sonrası birkaç veya pekçok olayı görmeyi arzu edebilir. Böylece değişken bir tetikleme noktasına ihtiyaç belirir. Veri dizilerinin çok uzun ve bellek kapasitesinin sınırlı olabilmesi nedeni ile, bir bellek penceresi (window) yerleştirilmesi uygun olmaktadır. Bir geciktirme üretici (delay generator) ve bir sayıcı sistemin saat darbeleri (counting system clock pulses) kullanılarak, ilk tetikleme noktasından itibaren, seçilen bilgilerin belleğe depolanması, belirli bir görüntü sağlayacak olan pencere sınırları oluşuncaya kadar kolaylıkla geciktirilebilir (şekil 1-5). Aynı tetikleme kelimesi kullanılarak ve gecikme değiştirilerek, verilen program bloklara ayrılarak, tek tek ve tümü ile incelenebilir.

Kapalı çevrimleri (nested loops) veya alt programları (subroutines) olan birçok programlar, tekrarlanan kelimeler veya çevrim geçişlerini gösterirler. Böylece, bir alt programın başarı ile yürütülüp yürütülmediğini belirlemek için, bu geçişleri sayan bir geçiş sayıcısına (pass counter) ihtiyaç vardır. Geçişlerin sayımı, yalnızca verilen bir alt programın tamamlandığını kaydetmekle kalmaz, aynı zamanda, bir alt programın hangi noktada, zamanından önce sonuçlanabileceğini belirtir. Birçok analizör en azından bir düzeyde geçiş sayımına sahiptir. Örneğin, 4 düzeylik ardışıl kelime tetikleme yapan bir analizör; beher kelime için 48 bit'lik genişlik sağlar.

Ardışıl kelime tetikleme, görünüşte belirsiz birtakım yollar için, tek bir program yolu seçmek için etkin bir teknik olabilir. Şekil 1-6'daki analizör, yalnız A kelimesinden sonra gelmesi koşulu ile, B kelimesi



Şekil 1-6. Veri akışının muhtemel üç sırası şekilde gösterilmiştir. Ardışıl tetikleme ve geçiş sayımı (pass counting) aracılığı ile her durum gözetlenebilir.

üzerinde tetikleme yapacaktır. Donanım (hardware) koşullarına göre, A adres kelimesi, analizörü, tetikleme öncesi durumuna getirir (arms), B'de (veya B kelimesinin n'inci görünümünde) analizörü tetikler.

Yazılım (software) aracılığı ile, A'da başlayan ve D kelimesinde biten bir alt program oluşturulabilir. A ve D arasında, B ve C üzerinden olmak üzere, belirgin olarak üç yol vardır. Bunlar, ABD, ACD, ABCD dir. Bu üç yol ED, AC, BC sırası ile tetiklenerek belirginleştirilebilir. Böyle bir tetikleme, çok sayıda tetikleme seviyelerini gerektirir ve çok sayıda kapalı çevrim geçişleri olan alt programlarla beraber, oluşumların sayısını da izlemek gerekir.

Karşılaştırma sınırlamaları koyan, referans bellek yeteneği, daha fazla dikkat isteyen yeni bir özelliktir. Bir referans belleğin varlığı halinde, bilinen verilerle yüklenip, bu veriler deney altındaki bir sistemden alınan verilerle karşılaştırılabilir. Şüpheli nokta (bug) tesbit edildiği zaman, benzemeyen kelime üzerinde kolayca tetikleme yapılabilir.

Hatadan önce ve sonra oluşan olaylar meydana çıkarılabilir. Temel olarak, böyle bir lojik analizör x,Y,Z tetikleme kriterlerini kullanarak, bir programın şüpheli kısmından, normal bir veri bloğunu elde edebilir ve bunu kendi içindeki ikinci bir referans belleğe aktarabilir. Bundan sonra, karşılaştırma şeklini (comparison mode) kullanarak analizör x,Y,Z kriterlerine uygun düğün verileri bekler. Bu veriler gelince de analizör, yakalanan veriler ile , referans bellekteki normal verileri karşılaştırır. Eğer veriler benzemezse cihaz tetiklenir. Veriler benzer ise , cihaz bunu alır ve x,Y,Z kriterlerinin tekrar belirmesini bekler.

Bu işleme yapılacak böyle bir ilave ile hem kanal genişliği hemde karşılaştırma için kullanılan belleğin kelime derinliği arttırılır. Örneğin, A alt programında beklenmeyen bir veri birleşiminin (data combination) belirmediğini ve yanlış bir adrese atlamasından şüphe edildiğini kabul edelim. Eğer A alt programındaki veriler devamlı değişiyorsa, analizörün, veri ortak yolundaki bilgileri (bus information), aynı anda kayıt etmekte olsa bile, karşılaştırmayı, yalnız 16 adres kanalı ile sınırlandırması mümkündür.

Bundan sonra, A alt programı 103 adres uzunluğunda ise karşılaştırma, referans bellekte 103 kelimeye kadar sınırlandırılabilir. A alt programına farklı yerlerden erişildiği ve diğer alt programlara çıkış olduğu zaman bu özellik çok faydalı olur.

Geleceği düşünürsek, 1980 li yıllarda, mikrobilgisayar alanında daha verimli lojik analizörlerin kullanılmaları beklenmektedir.

1980 yılında üretilen yeni lojik analizörler, uygulama farklılıkları ile birlikte, özel bir konu olarak belirginleşmektedir. Bu çeşit -

lilik, deęişik üreticiler tarafından deęişik giriş probu ve yardımcı parçaları kullanmalarından meydana gelmektedir. Sonuç olarak, çeşitli uygulamalarda geniş çapta kullanılabilen genel amaçlı bir lojik analizör meydana gelmiştir.

Özel bir amaç için analizör üretiminde uzmanlaşma, alışlagelen analizöre göre daha fazla giriş kanallarına ihtiyaç gösterir. Bugün özellikle, en yüksek saat hızı yavaş olan, geniş giriş yetenekli, mikro-bilgisayar tabanlı sistemlerin analizi için 32, 48, 64, 96 giriş kanalına sahip lojik analizörler vardır. Çok sayıda giriş elemanları kullanıldığında, kanaldan kanala gecikme farklılıklarının (channel-to-channel skew) artması yüzünden tasarım zorlanmış olup bunlar genellikle 10 MHz'in üzerinde çalıştırılmazlar. Çoğunlukla, kanaldan kanala oluşan gecikme hataları, analizörün kendi belleğine en yüksek veri alabilme (clocking) hızını sınırlar.

Bununla beraber 1980 yılındaki birçok yeni lojik analizörlerin fiyatları 750 ilâ 15000 ₺. arasında deęişmekte idi. 1979 ve 1980 yıllarında üretilen bütün zaman analizörleri 5 pF'dan 15 pF' a kadar deęişik giriş empedansına sahip olmalarına rağmen, 5 ns'a kadar oluşan çok dar sıçramaları (glitches) yakalayabilirler. Bu sıçramaları yakalayabilme yeteneğinin dięer bir yönü, analizörün, verilen bir darbe genişliğini sezmesi için gerekli olan aşırı çalışmadır. Tüm problemin bir nedeni, kullanıcının deney noktasından itibaren lojik analizörün ilk aktif elemanına kadar uzanan kabloların uzunluğudur. Bu pasif kablolar ne kadar uzun olurlarsa, giren işarete o kadar büyük kapasite ve endüktans gös-

termiş olur ve işaret kullanıcının devresine doğru, geriye yansır.

Bu durum, düşük hızlı (ortalama 20 MHz) lojik analizörler için, genellikle bir problem değildir. Çok yüksek, örneğin 50 MHz. den büyük saat hızına sahip olan lojik analizörler için, giriş "probe" tasarımı, lojik analizör alma devresine işaret geçişinin bütünlüğünü (integrity) güvence altına almak ve işaretin bu devre empedansı nedeni ile geriye, kullanıcının sistemine yansıtılması, büyük önem taşır.

Mikrobilgisayar kontrolsüz olarak üretilen bazı lojik analizörler, başlıca düğme ve anahtarlardan oluşan bir ön kontrol tablosuna sahiptirler. Cihazdan sağlanan çalışma türü sayısı genellikle orta düzeydedir. Şimdi kullanılmakta olan ve bir anahtar tablosu ile durumların izlenebildiği, bir CRT göstergesine sahip olan, üçüncü nesil analizörde özellikleri ve yetenekleri artmıştır.

Yapımcılar, yetenek ve özellikleri geliştirilmiş olan analizörlerin kullanılmaya hazır olduklarını iddia etmektedirler. Kullanıcı kendisine gösterilen analizörün tetikleme olaylarını programlayabilme, geciktirme uygulayabilme kolaylıklarına v.b. özelliklere ve sonuç olarak cihazın insanlarca ne kadar kolaylıkla kullanılabilme yeteneğine sahip olduğunun çok dikkatle değerlendirilmesi önem taşımaktadır. Bu değerlendirme gelecekte özellikle önem kazanacaktır. Çünkü, lojik analizörler ardışıl tetikleme olayları v.b. gibi daha karmaşık özellikleri kazanmaktadırlar.

Cihazın kullanılabilirliğinin diğer bir görünüşü, lojik analizörün arasına kullanılan bir araç oluşudur. Bununla beraber, mühendis, mikrobilgisayar tabanlı bir sistemi geliştirirken, osiloskobunu hergün

kullanırken lojik analizörü geliştirme programının sadece çok özel bir kısmında kullanılmaktadır. Artık bir projeden bir sonraki projeye kadar bir daha lojik analizörünü kullanmayabilir. Aracı raftan alıp yeni yeni bir probleme uyguladığı zaman onun için önemli olan, cihazın kullanılabilirliğidir. Kullanıcı için, analizörü kurmak ve çok kısa bir süre için kullanmak değer taşır. Gelecekteki 1-2 yıl içinde yeni bir ürünün başarısında aranılacak en önemli etkenler, özelliklerin adresleme kolaylıkları, programlama ve kullanım alanı genişliği olacaktır.

1.2. GİRİŞTE "PROBE" UYGULAMASI (INPUT PROBING)

Son zamanlarda, değişik türde özellikler arzeden giriş "probe" uygulamaları olmaktadır. Bunlar, genellikle, ya özel bir mikrobilgisayar için kullanılan sert tel "probe" şeklinde, veya IEEE 488 veya RS-232 C özel bir ortak yol (bus-oriented) güdümlü sisteme uygulamayı sağlayıcı "probe" şeklinde oluşturmaktadırlar. "Probe"lar, özel bir işlemin uygulanması için yerleştirildiklerinden, çok yönlü kullanılmamalarına karşın, bütün "probe" uygulamaları rahat kullanım bakımından operatöre fazladan bir boyut kazandırır. "Probe" lar kullanıcıyı, "probe" ları bir mikrobilgisayara veya onun ortak yol güdümlü sistemine bağlama sınırlarından kurtarır. "Probe" kullanma eğiliminin devam etmesi beklenmektedir. Çünkü, özellikle adreslenebilen pek çok değişik ortak yol güdümlü sistemler vardır. Bu sistemleri kullanan kişiler için, "probe"lar, gerçekten çok uygun görünmektedir.

Kullanıcının gelecekte karşılaşacağı en önemli problem 'desteklenmedir!'

Yapımcı, donanım ve yazılım konularında, hangi mikrobilgisayar "chips" i desteklemeye istekli veya yetenekli olduğuna karar vermeli- dir. Yapımcı, destekleme için en çok bilinen ve kullanılan veya en i- yi uygulanabilen "chips" leri seçmelidir. "Bununla da çalışır" düşün- cesinden kesinlikle kaçınılmalıdır. Yukarıda tartışması yapıldığı gibi, yüksek performanslı lojik analizörler, amaçlanan sınırlar içindeki sür- ate uygun "probe" ayarlamasına ihtiyaç duyarlar. Eğer amaçlanan sınır, 10 ns.lik zaman domenine (time domain) kadar ayırım yeteneğine sahip ise giriş "probe" ları yükselme süresi 1-2 ns. sırasındaki veri geçiş- lerini sayabilmelidir.

1.3. ANALİZÖRÜN UZAKTAN ARIZA BULMA ARACI OLARAK KULLANILMASI

(REMOTE DIAGNOSIS)

1980 yıllarında ortaya çıkan diğer bir eğilim de lojik analizör - lerin uzaktan arıza teşhisine yönelmektedir. Bu durum Dolch, Paratro - nics ve BP cihazları ürünü olan lojik analizörlerde RS- 232 C veri bağlantı ünitesi aracılığı ile sağlanmıştır. Bu olanak, lojik analizö- rün uzakta bulunan bir sistemden elde edilen verileri, onarım kolaylık- ları bulunan merkezi bir yere gönderebilmesini sağlar. Burada bulunan uzmanlarda böylelikle arızayı tesbit edebilirler.

Biomation K-100-D ve Hewlett Pacard 1602 gibi diğer lojik anali - zörler, IEEE 488 I/O (giriş/çıkış) ara bağlantı elemanı (interface) kul- lanırlar. Budurum, lojik analizörün kontrol sınırlarını 488 bağlantı elemanından geçirerek uygun bir kontrol aracı ile genişletmiş olur, ancak IEEE 488 ve RC- 232 C arasında bazı değiştirme elemanlarına da ihtiyaç yaratır.

Lojik analizörlerin, uzaktan arıza bulma aracı olarak olarak kullanılmaları düşüncesi, son birkaç yılda büyük önem kazanmıştır. Pek çok ana bilgisayar yapımcıları geniş çapta uzaktan arıza bulma yeteneğini uygulamaktadır. Lojik analizörü bu tür bir sistem halinde genişletme düşüncesi büyük bir aşama olarak görünmekte ve böylece uzaktan arıza tesbiti için gerekli bağlantılar yapılarak sağlanmış olur.

Lojik analizör, pek çok değişik kanallarda, aynı anda beliren verileri, zamanlama (timing) analizi için kendi iç saatine göre, veya veri "domain" analizi için sistemin saatine göre toplar. Geniş ve karışık bir sistemde, lojik analizör aracılığı ile arıza araştırıp bulunabilmesi için, sistemi çok iyi bilen personele ihtiyaç vardır. Çoğu bilgisayar üreticileri, bu tür bir analizi yapacak yetenekte personel bulmayı veya düşük bilgi düzeyindeki personeli yetiştirmeyi son derece pahalı ve zor bir iş olarak görmektedir. Böylece, ancak bir lojik analizör ile en iyi olarak arızası tayin edilebilecek olan güç ve karmaşık sorunların, araştırılması yapılan sistemi çok iyi bilen, fabrika personeli seviyesindeki destekleyici personel ile çözümlenmesine devam edilecektir. Arıza belirtilerinin uzaktan teşhisi, onların zaman kazanmalarına öylesine yardımcı olur ki , arıza görülen her yere gitmelerine gerek kalmaz. Bilinen doğru verilerle önceden programlanan bir analizör için, bir durum oluşturulabilir. Sonra şüpheli veriyi yakalayabilir ve iki veriyi karşılaştırarak onarım personeline ipucu sağlanabilir. Bu basitleştirilmiş bir yaklaşımdır. Çünkü sistem çapındaki problemlere, kendi belirtilerine göre yaklaşım, daha akılcı bir yoldur. Deneyler göstermiştir ki, lojik analizörler, uzmanların elinde daha verimli kullanılırlar.

Önceden programlanmış bir lojik analizörün ikinci bir görünümü, programlanma işlemidir. Arızanın bulunmasını sağlayabilecek bir referans veri dağılımını geliştirmek oldukça güçtür. Çünkü, kullanım sırasında görülebilen özel bir arıza fabrikada henüz programlanmamış olabilir.

Servis bakım-tutum ve arıza aramasında, en uygun görülen yöntem, deney altındaki sistemin geniş kapsamlı olarak, kendi belirtilerine göre arızanın tesbit edilmesi yöntemidir. Yapılışı sırasında, cihaz veya sisteme eklenen, kapsamlı ve akıllıca düzenlenen bir arıza programı (diagnostic routine), problemlerin , bir basılı devre kartına veya bir alt sistem düzeyine kadar hızla indirilerek, bulunmasını sağlayacaktır. Çoğu olaylarda baskılı devre (PC) seviyesine kadar arıza araması ve bu baskılı devreyi değiştirerek, sistemi tekrar çalıştırmak halen en hızlı yoldur.

İKİNCİ BÖLÜM

BİR ARIZA BULMA CİHAZI OLARAK LOJİK ANALİZÖR

2.1. GENEL ÖZELLİKLER

Sayısal osiloskopların yeni bir şekli olan lojik analizör, analog osiloskopların yıllardır temelini oluşturduğu bilgisayar gelişimi ve onarımı alanında, önemli bir test cihazı çeşidi olarak büyük bir ilerleme kaydetmiştir. Lojik analizörler, diğer araçlara göre, çok sayıda işaretini aynı anda ya senkron ya da asenkron olarak, dalga şekillerini yakalamak yerine lojik durumları yakalayarak ve bu durumları tetikleme işaretinden hemen sonra göstermek yerine, tetikleme işaretinden önce göstermek sureti ile büyük kolaylık sağlamışlar ve ticari bir cihaz özelliğini kazanmışlardır.

Lojik analizörler, genel olarak altı özelliğe sahip bir aile olarak tanımlanırlar.

1. Lojik analizörler, diğer bellek çeşitlerinden (özel CRT cihazları gibi) ziyade sayısal "solid-state" bellekleri kullanırlar.

2. Bir çoğu 16 veya daha fazlasayıda kanala sahiptirler.

3. Tetiklemeden önce gelen verileri saklayarak, "tetikleme öncesi gösterimi" (pre-trigger viewing) olarak adlandırılan eşsiz bir yeteneğe sahiptir.

4. Lojik analizörler tüm kanallardaki verileri "bir kez oluşan olaylar" esasına dayanarak saklarlar ve bu verilerin eşidini çıkarmak için kaynağa (test altındaki sisteme) gerek duymadan tekrar görüntülerler.

(Diğer bir deyimle, görüntüleme için istenen tekrarlama işlemi, saklamadan sonra analizör içersinde üretilir.)

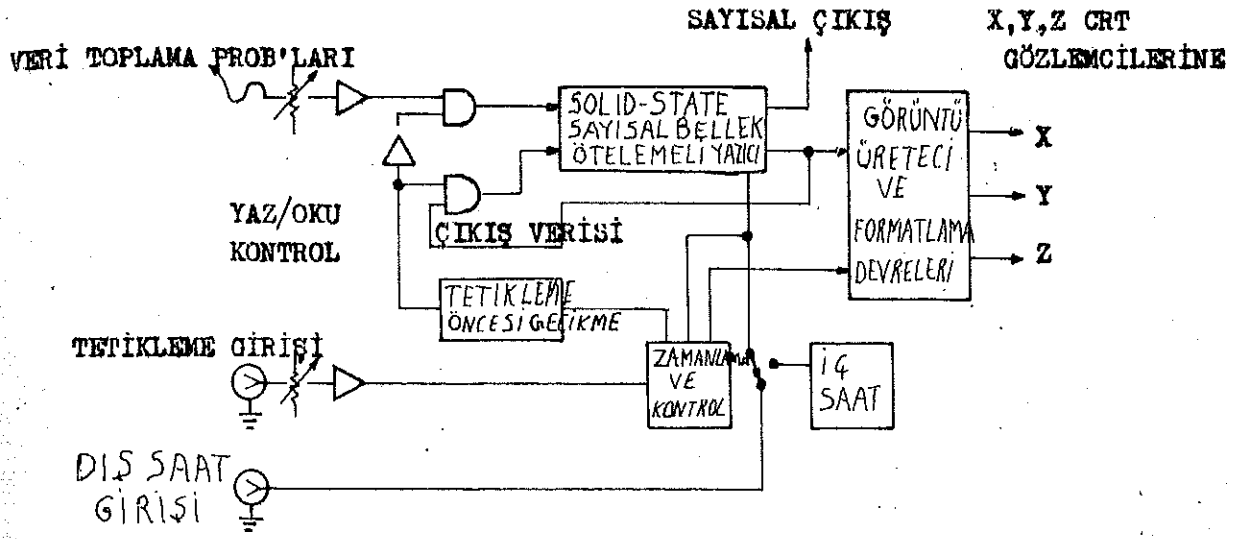
5.Çıkışlar daima "sayısal" şekildedir ve bu sebepten diğer sayısal araçlarla gerekli ara bağlantılar (inteface) kolaylıkla yapılabilir.

6.Lojik analizörlerin ekranları, yalnızca zamana bağlı olmayıp, tam tersine, bir kesme olduğu andaki lojik durumlar örneğinde olduğu gibi herhangi iki değişken görüntülenebilir.

Birçok firmalar tarafından üretilen lojik analizörler farklı yapı - larda görünmelerine rağmen gerçekte bu araçların tümü benzer genel bir yapıya sahiptirler(şekil 2-1).

Osiloskoplardaki gibi dalga şekillerini yakalamaktan çok, lojik durumları yakalamak için, bu araçlar "lojik 1" ve "lojik 0" için ayrı ayrı görüntü veren izler sağlarlar.(Bu durum, diğer gereçlerle birleştiril - dikleri zaman, özellikle bazı uygulamalarda, dalga şeklinin görüntülen - mesine göre belirgin bir üstünlük sağlar.) Bir bilgisayar sisteminde bu - lunan işaretler, aşırı boyutlarda olmadıkça, büyük ölçüde problemlere neden olmayan, düşük bir yükselme hızı veya sıçrama (overshoot) olayla - rı türünde pek hata göstermezler (şekil 2-2 a). Kritik faktör, genellikle bu noktadaki adres ve veri hatlarının yerleşmesi ile saat geçişleri arasındaki zaman ilişkisidir. Eğer bu işaretler, saat geçişleri olduğu anda, uygun "lojik 1 " veya "lojik 0" durumlarında kararlı bir konuma sa - hip değillerse, güvenilemeyen veriler elde edilebilir.

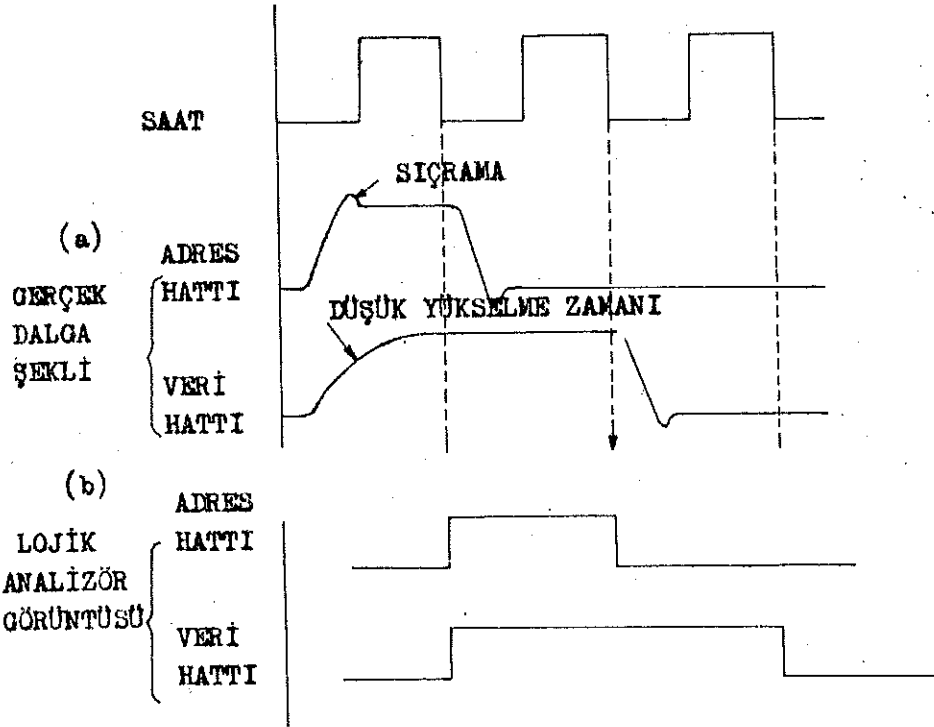
Bu tür lojik yapı, genellikle "saatlenmiş" veya "senkron" olarak ad - landırılır. Buraya kadar olan açıklamalar, bu gün kullanılan bilgisayar



Şekil 2-1. Genel bir lojik analizör. Şekilde yalnız bir kanal görünüyorsa da, çoğu üniteler en az 16 kanala sahiptir. Kendine ait ekranı olabileceği gibi, şekildekine benzer, ayrı olarak X, Y, Z, ekranları kullanılabilir. Bilgi yakalama bölümünde (diyagramın sol tarafı) önceden uygulanabilecek seçimler sağlanabilir.

sistemlerinde ortak olan özelliklerdir. Diğer tür "kendiliğinden saat-lemeli veya "asenكرون", özelliği olanlar, iletişim (communications) cihazlarında ve hızlı tasarımlarda kullanılır. Senkron çalışmada sistemin kendisi, lojik analizörün zamanlama kaynağı olduğu kadar izlenecek olan verilerin ve tetiklemenin de kaynağıdır. Temel olarak analizör, test altındaki sistemden, her saat geçişi anında, veriden bir örnek saklar. Bellek farklı yapılarda oluşturulur. Fakat en genel ve kullanışlı olanı 16x256 bit'lidir. Bu bellek mevcut bilgisayar sistemlerinde de geniş olarak kullanılan 16 bit'lik kelimeleri kullanabilir.

İşaretin kendisinden çok, yalnızca saat geçişleri anlarında ikili



(c)

```

1010 1111 1001 0100 0000 0000 0000 0001
1010 0110 1011 1111 0000 0000 0000 1010
1100 0101 1110 1100 0101 1001 0011 1000
1111 1000 0001 1001 1010 0001 1000 1100

1110 1001 1010 1101 0010 0110 0001 1111
1110 1010 0001 1100 1000 1010 1100 0011
1100 1000 1110 1100 1010 1110 1100 0001
1110 1010 0101 1010 0110 1001 1010 1110

1110 0001 1100 1010 0101 0011 0110 1001
1100 1010 0101 0011 1110 0001 0101 1111
1100 0001 0001 1010 1100 1010 0101 0001
0001 1110 1010 0101 0001 1110 1100 0011

0001 1110 1110 1111 1010 0001 1110 1110
0101 1010 1100 0011 1001 0110 0101 1011
0001 1010 1001 0110 1000 0001 1001 1110
1100 0001 1111 1010 0000 0001 0101 1001

```

Şekil 2-2. Bilgisayarlarda kullanılan genel işaretler.

(a). Verilerin sisteme girişi anında, saatin negatif geçişinde veri ve adres hatları karardır. Saatli bir lojik sistemde lojik seviyesi en önemli faktördür. Yükselme ve düşme zamanları daha az önem taşırlar.

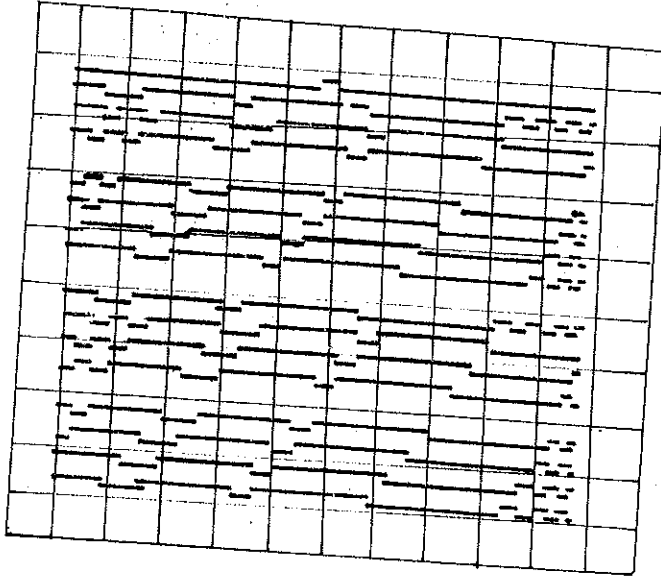
(b). Böylece, bir lojik analizörde yeterli bir görüntü yalnızca, seviyeleri belirtir, bir saat aralığı kadar gecikir ve saatin yalnızca negatif kenarında değişir. (c) seviyelerin ikili değerlerini gösterir, özellikle yazılım (software) ve donanım (hardware) işlemlerinin gözleniminde çok faydalıdır.

durumu saklama yeteneđi nedeni ile büyük miktarda sayısal verinin saklanabilmesini sağlar ve yalnız test altındaki sistemin tanımlandığı kısmı kaydedilir. Bu şekilde saklanan veri parametrik değil , fonksiyonel bir tanıtımdır. Bu, verilerin, bir işaret hattını "yüksek" veya "alçak" gerilim seviyesi olarak tanınması anlamına gelir. Çoğu uygulamalarda bu durum tercih edilirse de tam bir tanımlama değildir. Çoğunlukla, bir sistem yanlış çalışıyorsa, güçlükler, bir parametrik sınıfa giren, gürültü, uygunsuz işaret genlikleri veya diğer benzer durumlar olabilir. Lojik analizör buna benzer yanlış çalışmaları bulabilirse de, bir osiloskop, uygun olmayan durumları ölçmede ve istenilen düzeltmeleri tayin etmede kullanılabilen en iyi cihazdır.

Şekil 2-2(b) de gösterilen, donanım tasarımcılarının (hardware designers) çok iyi tanıdıkları dikdörtgen şeklindeki, dalga şeklini gösteren skop, bu gün pek çok lojik analizörde kullanılan bir gösteriş şeklidir. Özellikle çok sayıda kanalların varlığı halinde daha kolay bir okuma şekli sağlayabilen "0" ve "1" lerden oluşan bir diğer gösterim şekli de çok uygun olup şekil 2-2(c) de gösterilmiştir. Bilgisayar sisteminde arıza ararken ve arıza programının çalışmıyş nedenleri araştırılırken, 0/1 şeklindeki görüntüleme, özellikle cihazın kendi dilindeki programda, "0" ve "1" lerden oluştuđu için olayın çözümlenmesi yönünden büyük kolaylık sağlar.

2.2. LOJİK ZAMAN ANALİZÖRLERİ İLE LOJİK DURUM ANALİZÖRLERİNİN KARŞILAŞTIRILMASI

Bir iç saat (internal clock) kullanıldığı zaman, lojik analizör, bir zaman analizörüne dönüşür. Dikdörtgen şeklinde bir ekran ile birlik-



Şekil 2-3. Dikdörtgen görünümlü dalga şekli. Tipik bir 16 kanallı ekran yukarda gösterilmiştir.

te kullanıldığı zaman, osiloskop tipi , zaman tabanlı bi diyagram olarak görüntü elde edilmiş olur (şekil 2-2 (c)). Bu özellik çok geniş olanaklar sağlayıp gürültü sıçramaları gibi işaret bozuklukları veya uygun olmayan darbe genişlikleri kolayca gözlenebilir. Bununla beraber, bugün lojik analizörlerde en hızlı saat hızları yaklaşık, 200 MHz olup, 5 ns. ayırım yeteneği verir. Yükselme zamanları 10 ns. düzeyinde olan TTL sistemlerinde arıza aramak için bu ayırım yeteneği yeterlidir. Fakat, bu ayırım yeteneği, osiloskobun karakteristik özelliği olan nano saniyenin kesri kadar olan bir ayırım yeteneği ile karşılaştırılmaz. Diğer yünden, mekanik elemanlarla veya daha yavaş MOS lojik ailesi ile çalışıldığı zaman, 5-10 ns.lik ayırım yeteneği tam anlamı ile yeterlidir.

Hemen hemen, tüm lojik durum analizörleri, lojik zaman analizörleri kadar kolay çalışırlar. Bunlar, bir iç saat yerine, bir dış saat (sistem kendi saati olan veya olmayan) kullanırlar. Gerçek saat özelliği.

göstermeyen bir dış saat, arada kalan veriyi (adreslenmediği zamanki veriyi göstermeden, her adreslendiği andaki bir dış durum yazıcısının (status register) kapsamındakilerini göstermek için kullanılır. Bu tür gösterim için, dış "saat" adres kod çözücüsünden veya "chip - enable" işaretinden gelmelidir.

2.3. TETİKLEME ÖNCESİ DURUMU

Bir bilgisayar sistemindeki fonksiyonel hata (yazılım veya donanım) birçok makina çevrimlerinde hata meydana geldikten sonra belirgin olmaya bilir. Bu tür problemlerle karşılaşıldığı zaman, analizör, tesbit edilen hata koşullarını bir tetikleme olayı olarak kullandığı için, bellek yapısı nedeni ile, tetiklemeden önce nelerin oluştuğunu gösterir. Bu özellik, analizörü güçlü bir cihaz durumuna getirmektedir.

Lojik analizörün belleği (solid-state memory, FIFO-first in-first out. ilk giren ilk çıkar) saklama elemanı olarak çalışan, sürekli yenilenen ötelemeli yazıcı ile fonksiyonel açıdan eş özelliktedir. Saklama özelliği olan bir osiloskop (storage oscilloscope), yalnızca, bir tetikleme darbesinin yatay tarama devresini başlattığı andan sonraki verileri saklar. Böylece, sadece tetiklemeden sonraki verileri gösterir. Düşey saptırma devresindeki bir gecikme hattı (delay line), bu hattın meydana getirdiği geciktirme ile düşey kuvvetlendiricinin yayılma zamanının (propagation delay) toplamı, yatay tarama üretici devresinin ve tetikleyicinin yayılma zamanının toplamından büyük ise tetiklemeden hemen önceki bilgileri görüntüleyebilir. Fakat bu süre 50 ile 100ns. arasında sınırlıdır.

Lojik analizör sürekli olarak veriyi saklar veya yazar, tetikleme darbesi alınınca da saklama işlemini durdurur. Sonuçta, tetikleyici, bir osiloskopta olduğu gibi çalışmayı başlatmak yerine, çalışmayı durdurmuş olur. Tetikleyici, sistemi, "yaz durumundan" (write mode), "oku durumuna" (read mode) anahtarladığında, bellekteki veriler tümü ile geçmişe ait olur. Bu veriler, tetikleme olayından hemen öncesine kadar olan verilerdir. Tetikleme olayından sonra oluşan bazı veriler de saklanabilir, tetiklemeden önce ve sonraki verilerin toplamı, "yaz" durumundan, "oku" durumuna dönüşümünü geciktiren basit bir sayıcı ile kolayca kontrol edilebilir.

ÇOK IŞINLI EKРАНLAR (MULTI-BEAM DISPLAYS)

Birçok osiloskoplar, tek bir elektron tabancası ile değişik anahtarlama teknikleri kullanarak, skop'un birden fazla elektron tabancası varmış gibi görüntü vermesini sağlarlar. Genel olarak, kullanılan teknikler kesintili (chopped) ve değişken (alternate) düşey (vertical) çalışma şekilleridir. Böylece iki veya daha çok giriş kanalı arasında elektronik olarak anahtarlama oluşur ve bir CRT (katod ışınlı tüp) üzerinde "zaman paylaşımli çoklama" (time division multiplexing) şeklinde görüntülenirler. Herhangi bir belirli anda, yalnız bir kanal ekranda görüntülenebilir.

Bu durum, sayısal sistemleri kullananlar için problemlere neden olur. Çoğunlukla, bir bilgisayar sistemindeki işaretler, ya bir kez oluşan olaylardır (single-shot) veya öyle bir düşük tekrarlama hızına sahiptirler ki bütün pratik amaçlar için bunlar bir kez oluşan olaylar olarak gözönüne alınabilirler. Genellikle kontrol sırasında sak -

lama özelliği olmayan, anahtarlanmış ışınli (switched - beam) osiloskop kullanırken, bir bilgisayar tam bir kapalı çevrim şeklinde programlanmalıdır. Böylece problem, CRT ekranı üzerinde açık olarak görünebilir. Bu tekrar programlama şekli, yalnızca zaman yitirici olmayıp, çözümlenmesine çalışılan orijinal problemin gizlenmesine de neden olabilir. Çünkü çevrimin, tekrarlanması, koşulların tekrar oluşumunu sağlamaz. Örneğin, skop çoğunlukla yazılım (software) çevriminin başlaması ile tetiklenebilir. Bununla beraber, görülmesi istenilen problem üzerinde her an işlem yapılan verilere veya çevrim içinden birbiri ardına geçişler ile değişen diğer bazı etkenlere bağlı olabilir. "Değişken" (alternate mode) tarama şekli kullanıldığında, 1 no.lu kanal görüntülenirken, olay 2 no.lu hatta belirebilir. Bir sonraki geçişte, 2 no.lu kanal görüntülenecektir, fakat durumlar değiştiğinden olay kaybolmuştur. Osiloskop'un kesikli (chopped) tarama çalışma şekli kullanılarak, tarama sırasında her iki işaretten de parçalar alınacaktır. Fakat bugünkü sistemlerde gerek duyulan hızlı tarama işlemi için, mevcut osiloskoplardaki hızlar yeterli değildir.

Bellek kanalları paralel olduğundan, lojik analizör, gerçek bir çok ışınli osiloskoba benzer ve buna ayrıca "tetikleme öncesi saklama" (pre-trigger storage) yeteneği eklenmiştir.

Yazılım olarak tekrar programlamak (software reprogramming), lojik problemleri bulmada artık gerekli değildir. Bunun yerine, düzenli arıza programı sisteme yüklenir, lojik analizör uygun izleme noktalarına bağlanır, tetikleme düzenleyicileri (tetikleme kelimesi v.b.) seçilir ve analizör çalıştırılır. Programın ilk geçişinde yanlış bir çalışma, yaz/oku

(read/write) belleğinin kapsamındaki yok etse bile, analizör oluşan olayların sırasını göstermeye devam eder. Bu sıra, programın iyi çalışmasından beklenen sıra ile karşılaştırılmalıdır. Analizörde birçok paralel kanal kullanarak, bir geçişte tüm gerekli bilgilerin yakalanmasına izin verilir. Eğer gerekli ise, analiz bütün günü alabilir. Çünkü görüntü, hergün kullandığımız osiloskoplarda olduğu gibi, kaybolmaz. "solid-state" saklama yeteneği ile birleştirilmiş gerçek bir çok-ışınlı osiloskop kullanma fikri, sayısal elektronik alanında güçlü ve tek bir araç yaratmıştır.

SAYISAL ÇIKIŞ (DIGITAL OUTPUT)

Bazı analizörler, sakladıkları ikili bilgileri (binary information) bir ekranda göstermekle birlikte, bir çıkış kapısına (output port) da verebilmektedirler. Bu verileri analiz edebilmek için diğer bir sayısal bilgisayar sistemi kullanılacak ise bu özellik çok yararlı olabilir.

Önceki örnekte olduğu gibi, bir bilgisayar, alınan verilerin bit paternini arıza arama bilgisayarında saklamış ve iyi bilinen bir patern ile karşılaştırabilir. Bu tür çalışma, deney teknisyenine lojik hatanın bulunmasında önemli zaman kazandırır. Maalesef, bu ek çıkış kapısı herkez tarafından kullanılmaktadır, bununla beraber, özellikle uzaktan arıza aramalarına (remote diagnostics) önem verildiğinde, gelecekteki tasarımlar için büyük ümitler vermektedir.

BELLEK BOYUTLARI (MEMORY SIZE)

Bir bilgisayar tasarımı yaparken göz önüne alınması gereken önemli parametreler şunlardır. Veri kelimesinin uzunluğu, adres kelimesinin uzunluğu, saatleme için gerekli olan "hand shaking" işaretleri, seri ve

paralel formatlar, tekrarlama hızları(repetition rates), üç durumlu bağlantı elemanları (interfaces with 3-state) ve genel zamanlama (general timing relationships) ilişkileridir. Bunlar, kanal sayıları ile aynı olan belleğin kelime genişliğini ve bellek hızını etkiler.

Diger bir ana faktör, bellekteki toplam kelime sayısıdır(derinlik). Genel bir kural olarak, en derin bellek en iyi olanıdır. Derin bir bellek, tek bir geçişte , daha fazla veri yakalayabilir. Bir geçiş sırasında, hatanın programı bozması durumunda, bu faktör önem kazanır. Bellek büyüdükçe maliyet/bit oranı da düşmektedir. Büyük kapasiteli bellek kullanmak daha akla yakındır. Tez konusu analizörün belleği, 8 bit genişliğinde, 1024 bit uzunluğundadır.

2.4. M 6800 MİKROBİLGİSAYAR SİSTEMİNDE KARŞILAŞILAN BİR ARIZANIN

LOJİK ANALİZÖR İLE BULUNMASI

Bu araçların faydalarını daha iyi anlamak için, pratikte nasıl kullanıldıklarını tartışalım. Örneğin, MOTOROLA M 6800 mikrobilgisayar sisteminde karşılaşılan bir arıza durumunda, yazılım (software) ve donanımın (hardware) uygun çalıştığı biliniyor. 16 tabanlı 0000 dan 03FF'a kadar olan bellek yerleri, R/W (yaz-oku) bellek alanıdır. FA00 dan FFFF'a kadar olan yerler ise bir programlanabilen (p-ROM) salt-oku bellek alanıdır. Doğru çalışan bir sistemde, güç uygulandıktan sonra sistem, p/ ROM'da saklanmış olan başlangıç çalışma programına girer, bu anda belirli yaz/oku bellek yerleri silinir. Bu işlemin sonunda sistem durur ve daha fazla veri elde etmek için bir giriş terminalinden (teletype benzeri) bir kesme(interrupt) bekler. Sistemin çalışmamasının nedeni, p/ROM 'da kaybolmuş bir bit olmaktadır. Başlangıç çalışma programındaki doğru komut-

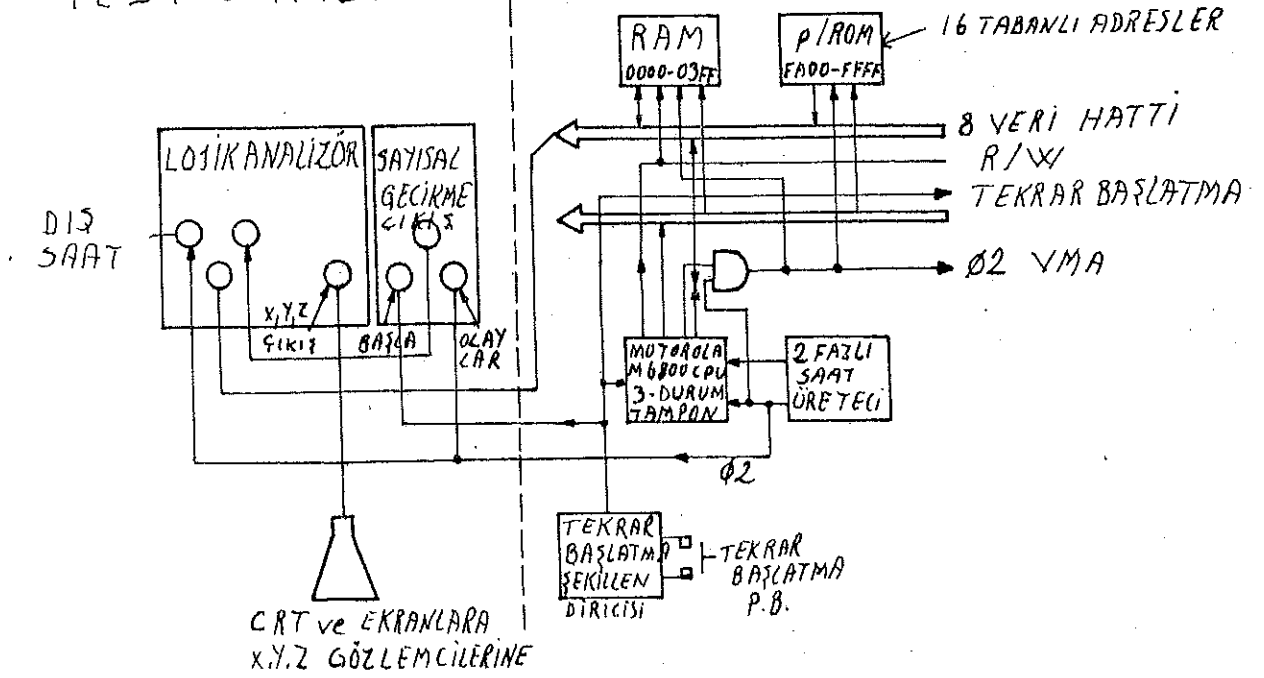
lardan biri "0010₁₆ bellek adresini temizle" (clear memory location) komutu idi. Bu komutun işlem kodu (OP CODE) 7F dir. Koşulsuz atlama komutu (unconditional jump) 7E dir. Bununla beraber, FPAC₁₆ adresindeki bu komutun en az anlamlı bit'i(LSB:least significant bit) düşerek 7F yerine 7E olmasına neden olur. 0010 adresinin kapsamındaki silineceğine, bu adres program sayıcıya yerleştirilmiş olur. Sistem R/W belleğe atlar ve sanki bu adrestekiler komutmuş gibi bu bölgede saklanan verileri yürütmeğe uğraşır. Bundan başka, meydana gelen bu hata , gözlemci için açıkça görünmez. Çünkü, merkezi işlem birimi (CPU) çok hızla çalışarak bir çıkış noktası bulur ve bir 'kesme için bekleme'(wait for interrupt) durumuna varır.

Bir lojik analizör, bu arızanın bulunmasında etkin olmuştur. Lojik analizörde bir tetikleme kaynağı olarak sayısal bir geciktirme ünitesi kullanılmıştır(şekil 2-4). Lojik analizör $\phi 2$ saatinden gelen belirli sayıda darbeleri sayar ve bir çıkış darbesi üretir. Bu darbe, $\phi 2$ işareti aracılığı ile saatlenen lojik analizörün tetikleme girişini sürer. Analizörün sekiz kanalı sistemin ortak yolunda bulunan verileri kaydeder. Eğer analizörün yeterli kanal kapasitesi var ise adres hatları da kaydedilir.

Bu tür düzenlemede analizör bir defa oluşan olayların ölçümünü yapar. Sayısal geciktirme ünitesi bir kez çalışmaya başlatılınca (initializing), lojik analizör, CPU' yu tekrar başlatmadan (restart:RESET) diğer bir çevrime (gözetleme) girmeyecektir. Analizör her komutun yürütülmesinden sonraki değil, her makina çevriminden sonraki ($\phi 2$ saat aralığı) veri hatlarının kapsamındaki gösterir. M 6800 , verilen bir

TEST CİHAZI

TEST ALTINDAKİ SİSTEM



Şekil 2-4 Mikrobilgisayar kontrolü. Sayısal gecikme önceden ayarlanan sayıda olayların oluşumundan sonra lojik analizörü tetikler. Bunlar düzenli veya düzensiz işaretler olabilir.

komutun başlangıcını ve sonunu gösteren bir işarete sahip değildir. Diğer bazı modeller bu bilgiyi saklayabilirler. Bu işaretin yokluğunda, ara bilgi, durum sırasını gösteren ekranda görünür. Bu veri programın akış diyagramı (flow-chart) ile bağlantılı değildir. Böyle bir durumda, lojik analizör, FFAC adresindeki p/ROM bellek yerindeki bozulmuş bir bit'i bularak, arıza arayıcı bir cihaz olmuştur. Böyle bir probleme neden olan özel hatayı tanımak bir olasılık problemidir. Fakat lojik analizör bu durumda, bilgisayarın çalıştırma programının faaliyetinin, belirli bir aralıkta gözlenmesini sağlamıştır.

KELİME TANIMLAYICILARI (WORD RECOGNİZERS)

Bir önceki örnekte olduğu gibi, analizörü tetiklemek için CPU'daki

tekrar başlatma işaretini geciktirerek kullanmak çoğunlukla pratik değildir. Ya da bu işlem uygulama için çok sınırlı kalır. Çok etkin bir tetikleme şekli , program sayıcısının verilen bir adresidir. Arıza bulmak için kullanılan pek çok yazılım (software) programları , çok sayıda adres kesme noktaları ile (breakpoint) birleştirilmiştir. Böylece değişik yazıcıların kapsamındakiler görüntülenir ve bu kontrol noktaları (check-points) , programda bu noktaya gelindiği zaman sistemin durumunu tanımlamaya yardım eder. Adres kesme noktası , yalnızca bir arıza arama programı ile kullanılmayıp , eğer yazılım olarak değil donanım (hardware) olarak uygulanmışsa , herhangi bir program ile de kullanılabilir. Bu durumda , programlarda ve çalıştırma sisteminde (operating system) değişiklikler yapmaktan kaçınmış olunur. Bu gibi sistem değişiklikleri , henüz çalışmayan sisteme karmaşıklıklar eklediklerinden , arıza yaratıcı yapıya sahiptirler.

Donanım , bir kelime tanımlayıcısıdır (word recognizer). Sürekli olarak program sayıcısının içeriğini , sabit bir adres ile karşılaştırır. Her ikisi , herhangi bir anda uyuşursa bir darbe üretir. Sabit olan adres , önceden anahtarlar ile ve el ile tatbik edilir. Kelime tanımlayıcısı , yalnız program sayıcısının olmayıp , herhangi bir yazıcının içerdiklerini izlemek için de kullanılabilir ve bir adres veya veriyi temsil eden herhangi bir bit paternine karşı tepki gösterebilir.

Bazen bir kelime tanımlayıcısı bir analizörün içindedir ve giriş kanalları ile paralel olarak bağlanmıştır. Bu durum her zaman istenmez. Çünkü özel bir adresin veya özel bir bit paterninin izlenmesi , kelime tanımlayıcısının esnekliğini sınırlamaya eğilimlidir. Genellikle daha

iyi bir yaklaşım , ana cihazın dışında bir kelime tanımlayıcısı kullanmaktadır.

SONUÇ:

Lojik analizör , bilgisayar arıza aramasında eski metodlara göre belirgin avantajlar sağlar. Kuşkusuz bir osiloskobun sağladıklarını da içermektedir. Kullanıcı, bu araçlara olan ihtiyacı anlamalı ve bilgisayar sistemlerinde kullanılmasını bilmelidir.

VERİLERİN GÖRÜNTÜLENME TÜRLERİ

Günümüzde üretilen lojik analizörler, çoğunlukla 16 tabanlı görüntü şeklini kullanmakla beraber 0/1 veya dikdörtgen şeklinde görünüm sağlayan skop kullanımları da vardır. 16 lı taban sistemi, kanal adedinin çokluğu halinde ve deneylerin yazılım (software) ve donanım (hardware) bağlı yapıldığı durumlarda daha avantajlı ise de elektriksel analizlerin yapılması durumlarında daha az faydalı olur. Örneğin, toplam 32 hatlık bilgisayar adresleri ve veri hatlarının izlenmesi yapılırken, bütün bu bilgileri tek bir ekranda gözlemek, yalnızca 8 adet 16 lı taban karakteri kullanmakla mümkün olur.

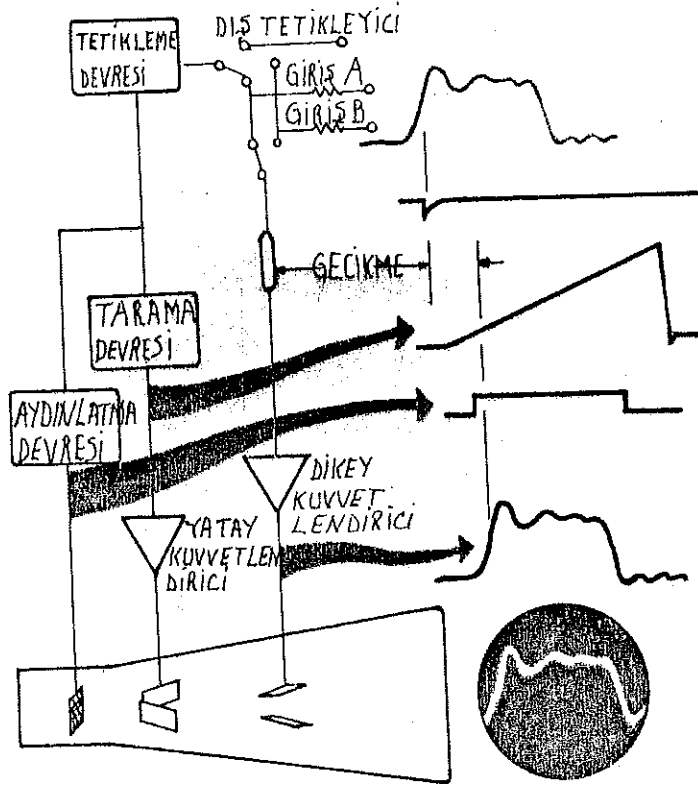
2.5. LOJİK ANALİZÖRLERİN DURUMU (THE STATE OF LOGIC ANALYZERS)

Günümüzde, aynı anda bir kaç kanalı birden kaydetme, tetikleme öncesi bellek ve kombinezonsal tetikleme teknikleri , zorlu sayısal ölçümlerde kullanılmaktadırlar.

Son zamanlara kadar , pek çok sayıda sayısal sistem gruplarında arasına meydana gelen ve çözümleri yaklaşık altı hafta kadar sürebilen problemler, bir tür olasılık oyunları oynamak suretiyle çözümlenebilmekteydi. Bu durum artık eğlendirici olmaktan çıkınca,

Lojik analizörlerin kullanılmasına başlanmış ve aynı problem karşısında, bir saat içinde, arıza belirlenebilmiştir. Bu örnek, lojik analizörlerin sayısal sistemlerde, bir osiloskobun, analog sistemlerdeki vazgeçilmezliğine eşdeğer düzeyde önem kazandığını göstermektedir. Gerçekten bu yeni sınıf cihazın, tetikleme öncesi kayıt, aynı anda çok kanal izleme, kombinezonsal tetikleme gibi karakteristik özellikleri, elektronik cihazlardaki zorlu gelişmeleri çözmeye, tüm kontrol ve bakım-tutum sorunlarını çözümlenmede oldukça ideal bir cihaz olduğunu kanıtlamıştır.

Lojik analizörler, sayısal uygulamalarda çok etkindirler. Sayısal sistemler, son yirmi yıl içinde gelişmiş ve daha karmaşık sistemler haline gelmişlerdir. Lojik analizörlerin kendiliğinden kontrol ve araştırma yeteneğine sahip oluşu nedeni ile, karmaşık sayısal sistemlerde çalışma alanları çok genişlemiştir. Bilindiği gibi osiloskoplarda veri toplama ve veri görüntülenmesi tek ve aynıdır. Lojik analizörlerde ise osiloskoplardan farklı olarak, birbirinden tamamen bağımsız bilgi elde edilmesi ve görüntülenmesi işlemleri oluşur ve sistem değişik bir biçimde çalışır. Bunun anlamı, hiç bir zaman lojik analizörlerin, osiloskopların yerini alamayacağı şeklinde olmayıp, lojik analizörlerin, sayısal sistemlere özgü sorunların çözümünde daha etkili olduklarının belirtilmesidir. Bir osiloskop (şekil 2-5) ile bir lojik analizör arasında en belirgin fark, tetikleme olayının, lojik analizörde veri toplama işlemini durdurmak için, osiloskopta ise veri toplama işlemini başlatmak için kullanılmasıdır. Bunun sonucu olarak, analizör veri toplama öteleme yazıcıları, tetikleme olayından sonraki değil, hemen önceki de-



Şekil 2-5. Veri toplama ve veri görüntüleme işlemlerinin birbirine bağlılığını gösteren tipik bir osiloskop diyagramı

ğişimlerin (conversions) sonuçlarını içerir. Çoğu zaman, tetiklemeden sonraki işaretlerin de gözlenmesi istendiğinden, lojik analizörler veri toplama periyodunun sonuçlanmasını, tetiklemeden sonra istenilen sayıda saat darbeleri kadar , geciktirmeyi sağlayabilirler. Bu nedenle, analizörler, hem negatif, hem pozitif olarak tetiklemeyi geciktirme kabiliyetindedirler. Örneğin, toplama yazıcıları 100 bit uzunluğunda ve durdurma olayı , tetiklemeden sonra 62 saat darbesi kadar geciktirilmiş ise, veri toplama olayı durdurulduktan sonra, yazıcı, tetiklemeden sonra 62 , tetiklemeden evvel 38 saat darbesi süresince oluşan olayları ihtiva edecektir. Geciktirme hatları (delay lines) yerine sayıcılar (counters) kullanıldıklarından, pozitif tetikleme gecikmesinin miktarı tamamen gelişigüzedir. Bunun sonucu olarak , bazı analizörler 10^6 saat darbesine ka-

dar çok geniş bir ayarlama kabiliyetine sahiptirler ve böylece tetikleme kelimesinden çok sonra başlayan zaman pencereleri (time window) içindeki işaretlerin gözlenmesinde yüksek ayırım (resolution) yeteneği sağlanmış olur.

Lojik analizörler tetikleme olayını veri toplamasını başlatmaktan ziyade, durdurmak için kullandıklarından, karmaşık tetikleme kriterlerini test için yeterince zaman vardır. Bazı analizörler, bu zamanı, tüm girişleri (bir paralel kelime olarak kabul edilen), "n" sayıda giriş için doğru (true), yanlış (false), göz önüne almama (don't care) birleşimlerinin mümkün olan 3^n durumundan herhangi birisiyle karşılaştırmak için kullanılırlar. Tetikleme ölçütü "giriş" kelimesiyle seçilen "birleşim" arasındaki uyumsuzdur. Böylece lojik analizörler, çok sayıda ikili işaretler (binary signals) arasındaki zamanlama ilişkilerini, bu işaretlerin içinde, tanımlanması önceden belirlenmiş olan bazı paternlerin oluşumundan hemen evvel veya hemen sonraki bir zaman periyodu süresince, gözlemeye yeterlidirler. Bu sayısal sistemlerde olduğu gibi, işaretlerin oluşum anında sistemin durumunu belirttiği yerlerde, aranılan kesin ve ince ölçüm tipidir.

DURUM DİZİSİNİN KAYDEDİLMESİ (STATE SEQUENCE RECORDING)

Bir sayısal sistemde, çok sayıda değişkenleri, aynı anda gözlemenin önemi üzerinde gereğinden fazla durulmayabilir. Herhangi bir sayısal sistemin çalışması, "n" boyutlu bir durum uzayında, durum değiştirmelerinin sırası olarak görüntülenebilir. Genel olarak, belirlenen herhangi bir durum için "doğru" olan "bir sonraki durum", verilen durum ile giriş değişkenlerinin o andaki değerlerinin bir fonksiyonudur. Bir sayısal sistemin

"uygun" çalışmasının anlamı, durum uzayında bir yol izleyerek, sistemi, o ana kadarki giriş sırası için doğru olan duruma getirmesidir. "Uygun olmayan çalışma"nın anlamı ise, sistemin durum uzayında kaybolmasıdır. Kaybolmuş bir sistem ya yanlış bir durumda kalacak (hang-up) veya çevrimde dolaşacaktır. (Hang-up in loop). Problem sistemin durum uzayında nerede olduğu ve buraya nasıl geldiğini belirlemektir.

Sistemin çalışması doğru olmayınca (yani sistem tasarlandığı gibi çalışmıyorsa), bir sistem modeline, hatayı arayıcı yargı yolu ile problemi belirlemek imkansızdır. Sistemin, doğru durum değiştirme sırasından nerede ayrıldığını oldukça kesin olarak ortaya çıkartmak amacı ile bu değişkenleri gözlerken, sistemi bilinen bir duruma getirmek (clear it) ve giriş sırasını tekrar etmek gereklidir.

Eğer bu analizi yapmak için kullanılan cihaz, değişkenleri aynı anda gözleyemezse, sonuçlar çok aldatıcı olabilir.

DURUM DEĞİŞTİRME SIRASININ KAYDEDİLMESİ (RECORDING TRANSIENT SEQUENCE)

Gerçekte, herhangi bir bilgisayarda aralıklarla oluşan arızaların sayısı, günde iki veya üç defayı geçmez. Bir analog sistemde aralıklı olarak meydana gelen arızaları izlemek için en uygun cihaz geçiş kaydedicisi (pass counter), bellekteki osiloskoptur. Fakat bir sayısal sistem hakkındaki kritik bilgi, bir veya iki kanallı cihazların kullanılmasını ciddi derecede sınırlayan, dalga şekillerinden ziyade, işaretler arasındaki zamanlama ilişkilerindedir.

Bununla beraber, lojik analizörlerde kullanılan bellek ve aynı anda veri toplama özelliği, bu cihazları lojik değişimlerin pratik kaydedicisi durumuna getirir. Gerçekten, sistemde aralıklarla meydana gelen her

arızanın oluşumunda, lojik analizörün çok sayıda bilgi kanalını kaydedebilme kabiliyeti, problemi anlamaya ve çözmeye yeterli veriyi toplamak için gerekli olan zamanı azaltır.

Aynı anda veri toplama zorunluluğunu kabul eden tasarımcılar, değişkenlerin aynı anda gözlenmesine izin vermek için, sistemlerine özel yetenekler eklemişlerdir. Sistemin özel bakımı-tutumu sırasında, zamanlamanın sağlanması için elle çalışan bir saat eklemek, çok kullanılan bir tekniktir. Bu saat, bir düğmeye basarak (push button) çalıştırılır ve gözleme altındaki her değişkenin statik durumu, her saat darbesinden sonra yazılır. Her saat darbesinden sonra değişkenlerin değerlerini belirlemek ve kaydetmek birkaç dakika alırsa da, gözlemler, saat işaretlerine göre yapıldığından, aynı anda oluşturulmuş kabul edilebilir. Can sıkıcı olmasının yanında oldukça etkin olan bu "durum değiştirme sırasını tek tek adımlama" tekniğinin dört problemi vardır.

1. Sistem yavaş çalıştırıldığı zaman, yayılma gecikmesi (propagation delay), kendi kendini bulma (recovery) zamanı v.b. olaylarla ilgili problemler meydana gelmeyecektir.

2. Ya veri veya kontrol işaretleri paternlerine bağlı olan, araştırmacının bilmediği problemler, kabul edilebilir bir uzunlukta olan, adımlama parçalarında meydana gelmeyebilir. Yani sistem 1 MHz. de çalıştığı zaman hemen hemen saatte bir kez oluşan rastgele (random) problemlerin, sistem, 0,02 Hz. de elle çalıştırıldığında, yaklaşık her 6000 yılda bir kez meydana gelmesi beklenebilir.

3. Sık karşılaşılan bir durumda şudur: Örneğin, bir disk gibi,

sistemin bazı parçaları tek tek adımlanamaz. Bu durumda sistemin geri kalan kısmının tek tek adımlanmasına izin vermek için, bu parçanın işlemsel olarak bir benzerini yapmak gerekli olur. Sistemlerin arıza sırasında yeterli derecede araştırılmalarına izin vermek için gerekli işlemler, bu iş simülatörler, bedellerinin yüksekliğine ve gerçek koşulları tümü ile oluşturamamalarına , genellikle kullanılmaktadırlar.

4. Bir sistem tek tek adımlanıyor iken beklenen görevi yürütemez.

Cihazın içinde tek tek adımlama donanımı kullanmak yerine lojik analizör kullanan bir araştırmacı, her ölçümde belirsiz sayıda değişkenleri kaydedemez. Bununla beraber, araştırmacı , sistem devrede iken normal çalışma halinde ve en yüksek hızında çalışırken dahi kullanabilir. Bunun sonucu olarak, arada bir oluşan problemleri, bir analizör ile farklı değişken gruplarında bir seri kayıtlar yaparak çözümlemenin bedeli, tek tek adımlama metodu kullanarak durum değiştirme sırasını (sequence of transitions) kaydetmeye göre daha az olacaktır.

2.6. TETİKLEME ÖNCESİ KAYDETME YETENEĞİ

Sayısal sistemlerin test edilmesinde tetikleme öncesi kaydetme (negatif tetikleme gecikmesi) yeteneği son derece değerlidir. Doğaldır ki, sayısal sistemlerde oluşan bazı basit problemlerde , tetikleme öncesi kaydetme yeteneği bir üstünlük sağlamaz.

Örneğin bir bilgi işlemci (processor), özel bir bellek bankasından okuduğu her kelime için "n" bit düşerse, bilgisayarı, bellek bankasından, kapalı bir çevrim içinde devamlı okumaya zorlamak basit bir iştir.

Bundan sonra bit'i düşüren devre bulunana kadar, bit'i yolu üzerinde için alışılmış bir osiloskop kullanılabilir. Hatalar arasındaki zaman kısâ olmağa ve hemen hemen sabitleştirmeye zorlanabildiğinden, çalışma kolay olur. Sonuç olarak, hatanın n'inci oluşumunda bir osiloskop'u tetiklemek, hatanın (n+1) inci düşüşünün uygun bir zaman skalasında izlenmesine izin verir. Maalesef meydana gelen pekçok zor sorunlar bu şekilde çözümlenemezler. Bundan başka, bir hatadan ayırık olan işaretlerin incelenmesi, bu işaretlerin bir milyarda 999999999 'uncuda bir kez oluşacağını meydana çıkaracaktır. Bir hatadan hemen önce gelen, milyarda bir oranındaki bu periyod, yararlı bilgiyi içerir. Buandaki işaretleri gözlemenin çaresi yoktur. Araştırma, probleme neyin sebep olduğunu tahmin etmeyi güçleştirir ve sonrada bu tahminlerin kanıtlanıp kanıtlanmadığının gözlenmesine başlanır.

Tetikleme öncesi kaydetme özelliğinin değeri, geçmişte, tasarımcının çoklamalı(multiplexer) ve kontrol edici gibi çok önemli sistem bileşenlerinde çok sayıda durum yazıcıları (stocked status registers) ile birlikte kullanmaya olan isteği ile kendini göstermiştir. Bu yazıcıların sistemin bütün değişkenlerini kaydedecek derecede genişlikte olmamakla beraber çok seyrek olarak üç kademedan (level) daha derin olamazlar ve herhangi bir yanlışlık olmadıkça kullanılmazlar. Bu yazıcılar karmaşık sistemlerde çok değerlidirler. Örneğin, son üç disk transferinin başlama adreslerini belirlemek sureti ile ve bu adreslerin disk kontrol edicilerce alınması ile, sistemin bozulmasının sebebinin bulunması için değeli bir ip ucu sağlamış olurlar. Lojik analizörler, yığın halinde yazıcılar-

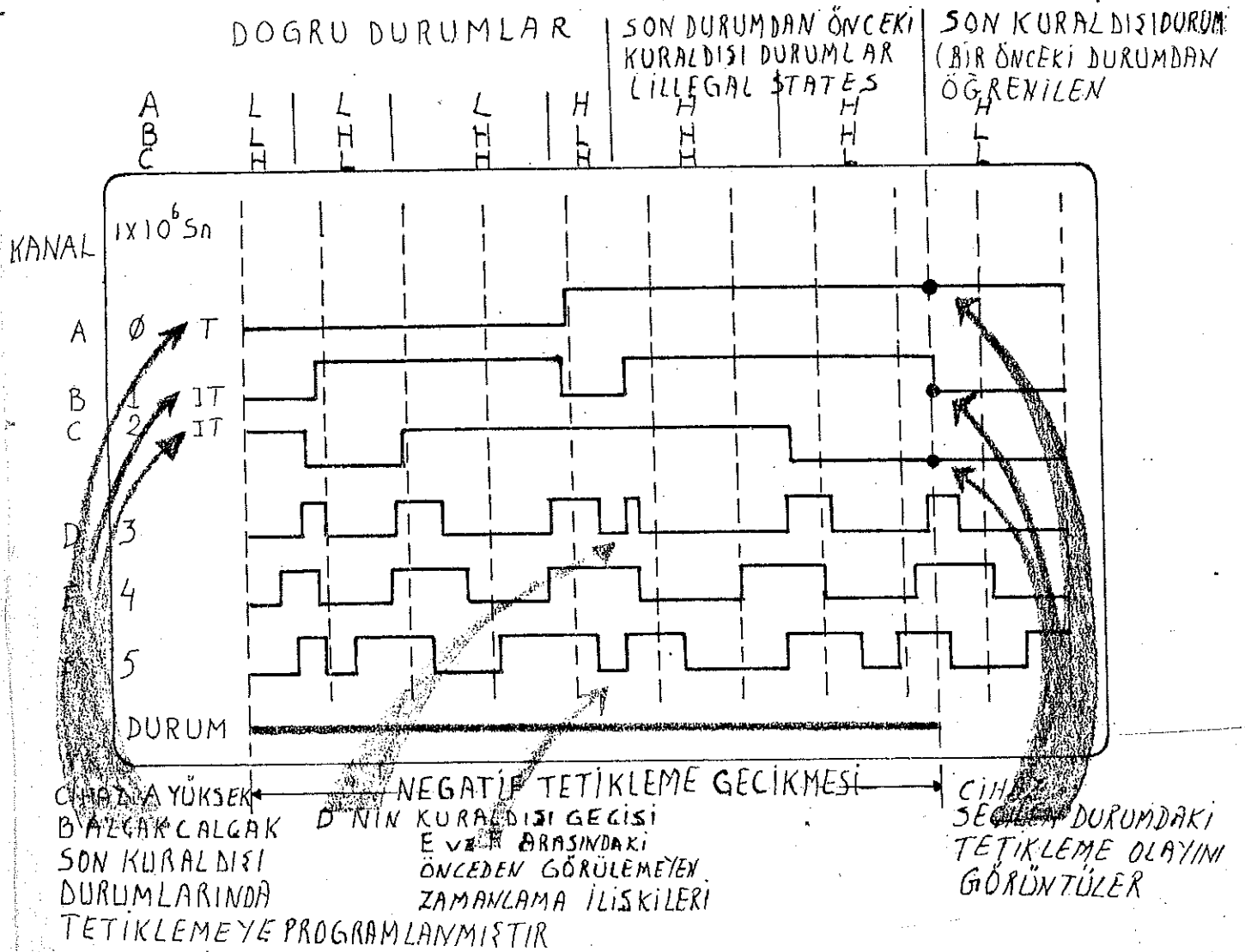
daki (stocked register) bilgilerin arşiv olarak saklanabilmelerini sağlayabilir ve yalnızca bir yazıcıya donanım olarak bağlanmış olanlar değil, aynı zamanda sistem değişkenlerinin herhangi birini kaydetmek için de kullanılabilirler.

2.7. KOMBINEZONSAL TETİKLEME

Kombinezonsal tetikleme, sayısal sistemlerin gözlenmesinde özellikle yararlı bir yetenektir. Bu sistemlerin doğal bir özelliği, bu sistemlerin durumlarının, herhangi bir değişkenin değeri veya durum değiştirmesinden pek çok sayıda değişkenin aynı andaki durumları ile, tanımlanmasıdır. Bunun sonucu olarak, lojik analizörün, özel bir sayısal sistem durumunu tanıma ve tetikleme yeteneği, bir osiloskobun belirli bir gerilim seviyesinde tetikleme yeteneği ile fonksiyonel olarak karşılaştırılabilir. Bir sayısal sistem, hata yapar ve kural dışı (illegal) bir durumda durursa veya bu durumu kapsayan bir kapalı çevrime (loop) girerse, değişkenlerin oldukça basit bir kombinezonsal birleşimi vasıtası ile bu kural dışı durumu tanımlamak genellikle mümkün olur.

Örneğin, ilerle (jump in progress) , bir sonraki alma (fetch next instruction) ve dolaylı adresleme (indirect address) değişkenlerinin hepsi "gerçek" (true) durumda iken bilgisayar durabilir (hang-up). BU üç değişkenin hiçbir vakit aynı anda oluşmadığını kabul edelim ,bu kombinezonsal birleşim hatalı durumu tam anlamı ile tanımlamamasına karşın,bu hatalı durumu diğer hatasız bir durumdan ayırt etmeğe yeterlidir.

Bunun sonucu olarak, bir lojik analizörün üç kanalı ,yukarda anılan üç değişkeni izlemek için kullanılabilir ve diğer kanallar, yukardakine benzer düşüncü yolu ile hatalı çalışmanın yapısını meydana çıkaracak olan ila-

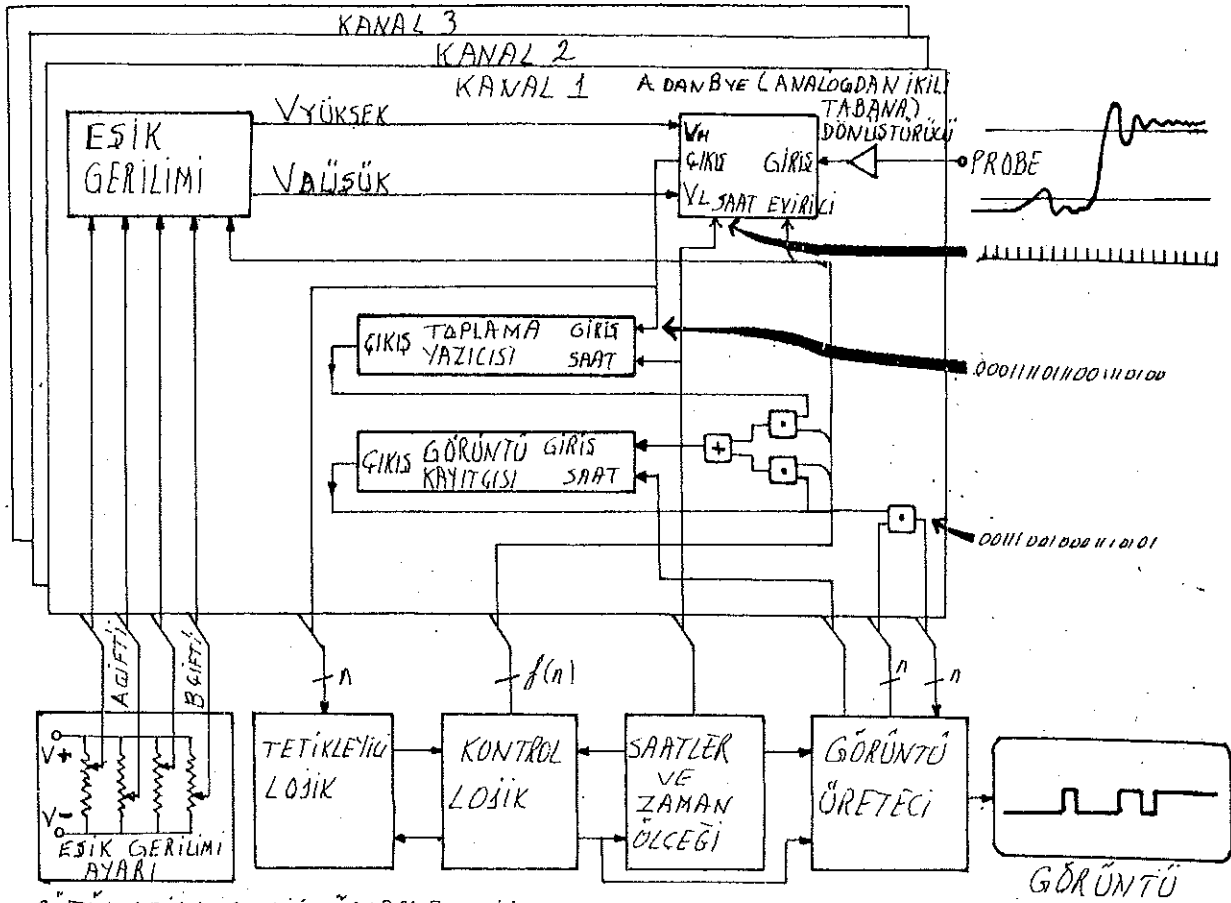


Şekil 2-6. Son kural dışı durumda tetikleme yapan bir lojik analizör, D işaretinin istenmeyen bir geçişi ile başlatılan kural dışı durumların sırasını ortaya koyar. Buna, E ve F arasındaki önceden görülmeyen bir zamanlama ilişkisi sebep olmuştur.

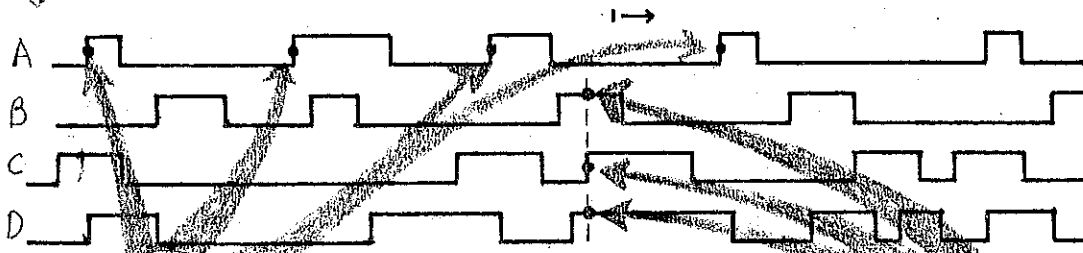
ve değişkenleri kaydetmek için kullanılabilir. Anormal durumu karakterize eden, bu üç değişkenin aynı anda oluşması (true) durumunda, analizör tetiklendiğinde, bir anda bir veya ikisinin "gerçek" (true) durumuna geldiğinde lojik analizör veri toplamayı durdurur ve bu kural dışı duruma geçişten hemen öncesine kadar, seçilmiş değişkenlerin kaydedilmesini sağlar.

Sık sık, bu kayıt etme, daha önceki bir kural dışı durumun varlığını

ortaya çıkaracaktır (şekil 2-6). Lojik analizör, hatanın başlangıcını elde etmek için, bu, aynı anda oluşan üçlü "gerçek" (true) durumundan hemen evvelki durumu tetiklemek için kullanılabilir. Anormal lojik durum değiştirme dizisinden geçerek geriye doğru çalışma tekniği kullanılarak, sistem arızasının gerçek yeri bulunabilir ve incelenebilir.



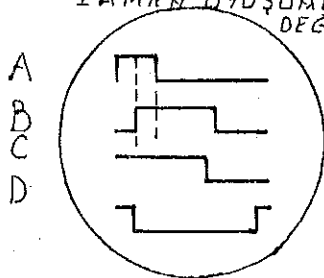
BÜTÜN LOJİK ANALİZÖRLERDE VERİ TOPLAMASI VE GÖRÜNTÜ KARAKTERİSTİKLERİ BAĞIMSIZDIR.



OSİLOSKOP BU ANLARDA TETİKLER, İZLEMELER ZAMAN UYUŞUMLU DEĞİLLERDİR.

TETİKLEME ANALİZÖRÜNE ABC NİN GATİSMASI ASLA OLUŞMAZ.

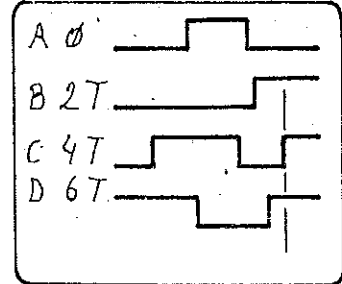
ANALİZÖR BCD NİN AYNI ANDAKARŞILASMASINDA TETİKLER



CRT EKRANI

A	0	T
B	2	T
C	4	T
D	6	

LOJİK ANALİZÖR



NEGATİF TETİKLEME GECİKMESİ

ÜÇÜNCÜ BÖLÜM

SAYISAL SİSTEMLERDE ÖLÇMELER

3.1. GİRİŞ

Modern sayısal sistemler çoğunlukla işlem yapıcı (processor) tabanlıdır. Dolayısıyla , modern IC (tümdevre) yapım tekniklerinden yararlanılarak, artan güvenilirlik avantajı elde edilmiş ve çok maksatlı devreler bir tek kırkık (chip) içerisinde geliştirilmişlerdir. Mikrobilgisayarlar , bu gelişimin doğal sonucu olmuştur ve ek bir avantaj sağlamışlardır. Mikrobilgisayarların olağanüstü gücü , pek çok görevlerin yapılabilmesine izin verir. Mikrobilgisayarlar yazılım (software) kontrollu olduklarından , sistemin çalışması, pahalı donanım (hardware) değişimlerine lüzum göstermeksizin, derhal değiştirilebilir. Bununla beraber, mikrobilgisayar sistemleri kendiliklerinden problemler yaratırlar. Bu problemlerden biri sistemin izlenmesi ve analizidir.

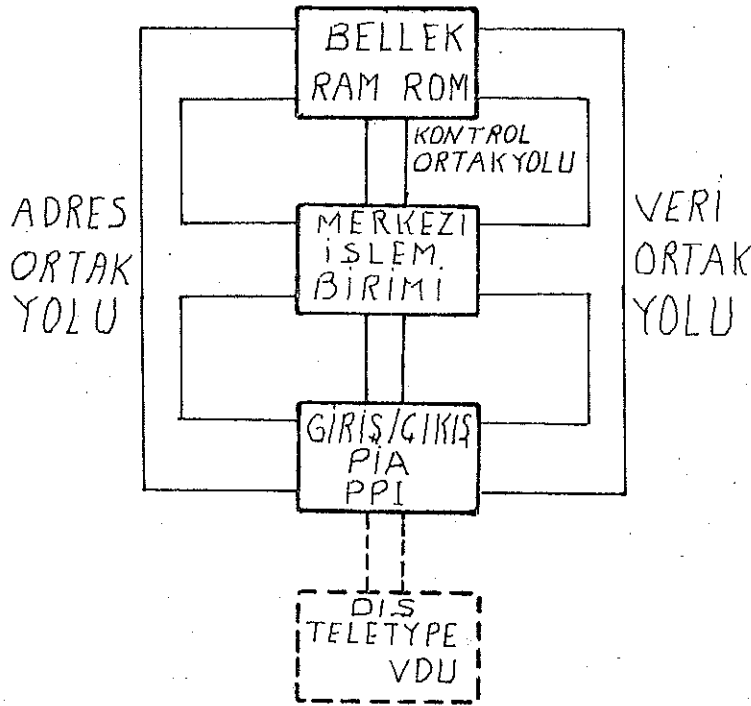
3.2. İŞLEM YAPICI (PROCESSOR) TABANLI SİSTEMLER

İşlemsel tabanlı modern sayısal sistemlerin genel görünümü şekil 3-1 deki gibidir. Bu basit sistem bile 28 hat (16 adres, 8 veri (data), 4 kontrol) içerebilir. Dolayısıyla, bu karmaşıklığıdaki sistemin çalışmasını tam olarak açıklamak üzere bu 28 hattın izlenmesi gerekir. I/O(giriş/çıkış) kapılarını izleme ihtiyacı, izleyici cihazdan istenen özelliklerin artmasına sebep olmuştur.

İşlemsel bir sistemin üç ortak yolu (bus) değişik özelliklere sahiptir ve izleyici cihazların çalışmasında buna karşı düşen özellikler

aranır. Adres ve veri ortak yolları (bus) veri taşırlar. Bu bilgiler, bir veya daha fazla kontrol devresi veya saati (clock) ile eş zamanlıdır. Bu ortak yollarda oluşan veri, yalnızca bu veriyi ilgilendiren saat darbesinin aktif kenarında meydana gelir. Kontrol hatları ve paralel giriş/çıkış hatları için, değişik hatların durum değiştirme anının sırası önemlidir. Seri bir bilgi kapısında (port) veya yolunda kelime veya kelime tanımlayıcılarının her ikisi ardışıldır.

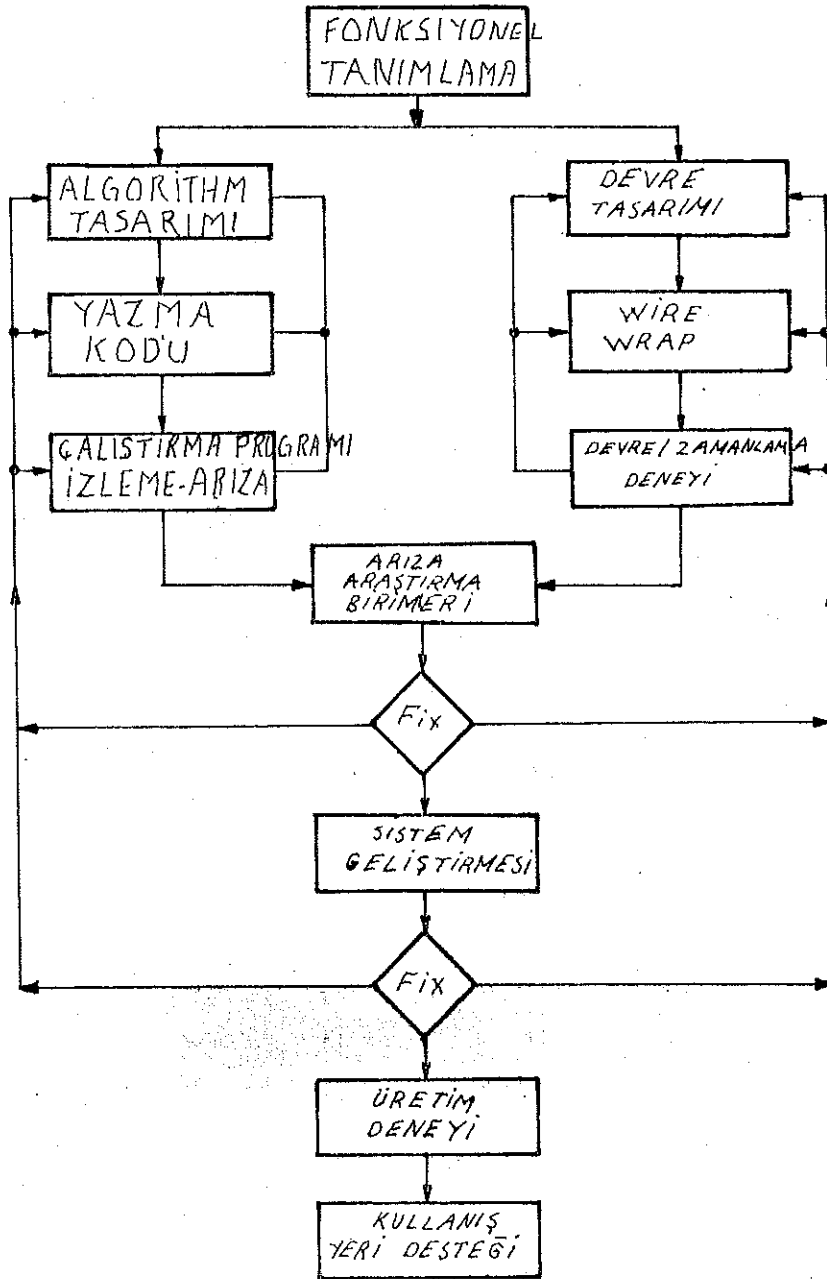
Sayısal sistemlerin çalışma çevriminin basitleştirilmiş biçimi şekil 3-1 ve şekil 3-2 de verilmiştir. Bir kere tasarım belirginleştik-



Şekil 3-1 Tipik mikrobilgisayar sistemi.

ten sonra iki yoldan gidilir. Yazılım ile uğraşanların, kod üretimine yönelik çalışmalarına karşın, donanım ile uğraşanlar, bellek ve ara bağlantı (interface) elemanlarının geliştirilmesine ağırlık verirler. Bu iki yol tasarımda bağdaştırılabilirdir. Tasarım başarı ile denendikten sonra, bunu üretim izler ve neticede hizmete sokulur.

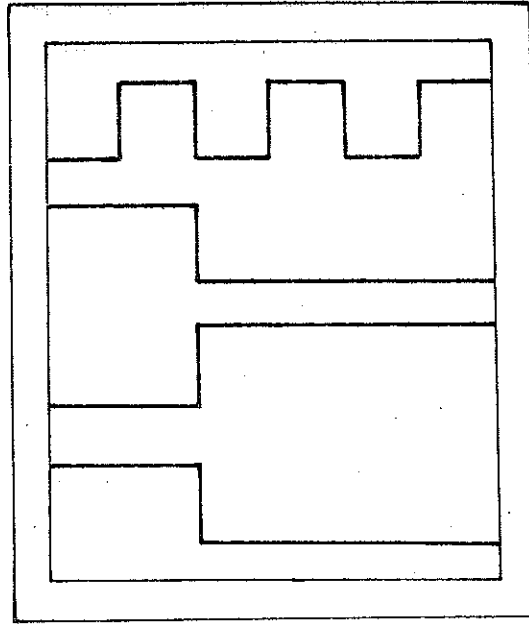
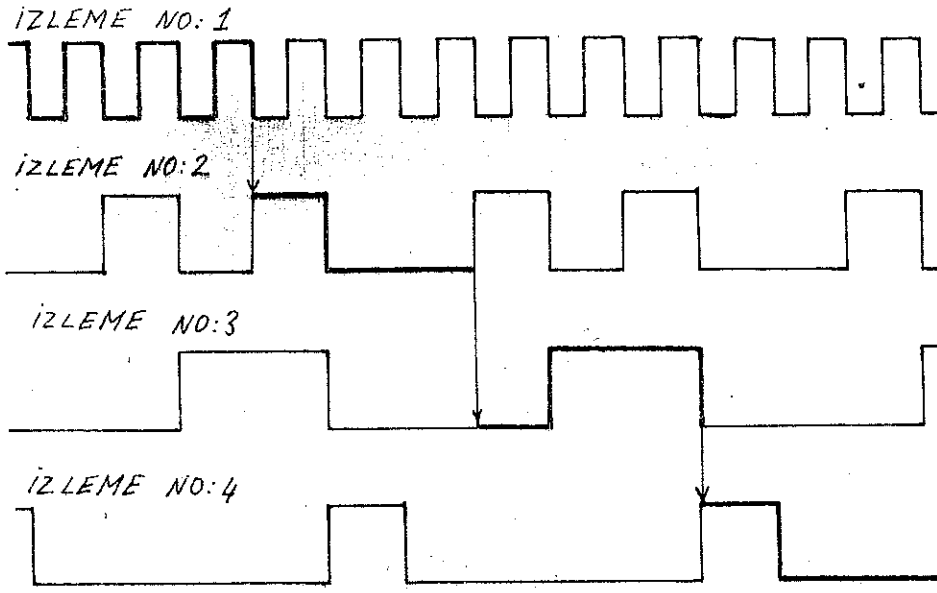
Sistemin gelişmesindeki değişik evrelerde, değişik izleme işlemlerine gerek duyulmuştur. Geliştirme aşamasında izleme cihazlarının kapasite-leri, oldukça memnuniyet verici düzeyde olmalıdır. Üretim aşamasında ya-pılacak olan bir çok yüksek hız denemeleri önemlidir. Kullanıcı bakımın-dan ise, taşınabilirlik ve kolay kullanım temel özellikler olmalıdır. Bu cihazlarda aranan kriterler ileride tartışılacaktır.



Şekil 3-2 Sayısal sistemde gelişme işlemi

3.3. OSİLOSKOPLARIN SINIRLANDIĞI NOKTALAR

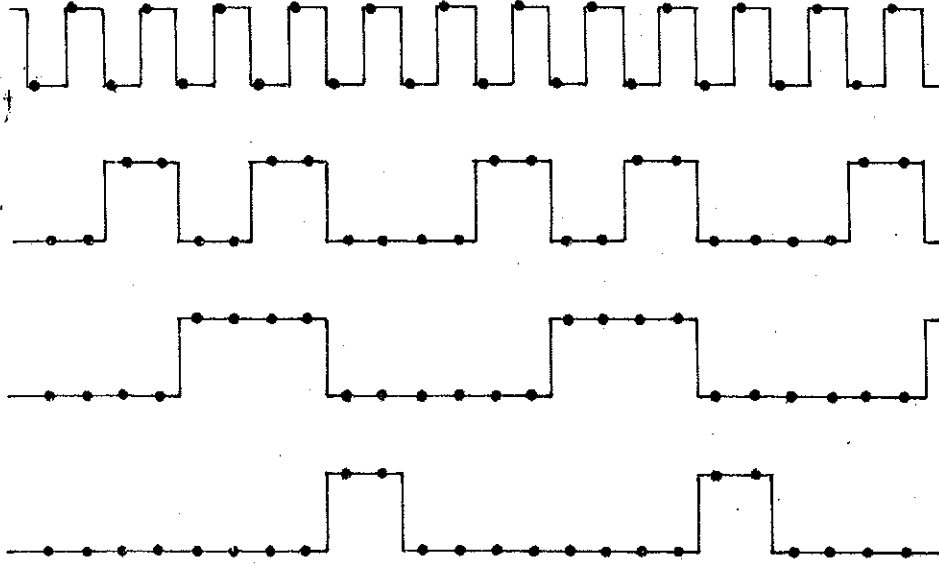
Çok yakın zamanlara kadar, sabit işaretleri ölçmek için test cihazları olarak multimetreler ve zamana bağlı işaretleri ölçmek için ise de osiloskoplar kullanılagelen cihazlar olmuşlardır. Osiloskoplar yükselme ve alçalma süreleri, dalgalanmalar, aşırı sıçramalar veya bir işaretin darbe (pulse) genişliği gibi gerilim bilgilerinin (parametrelerinin) görüntülenmesi için ideal cihazlardır. Bununla beraber, modern sayısal sistemlerde aynı anda izlenmesi gereken çok sayıda düğüm noktaları bulunmaktadır. Çok az sayıda osiloskop ikiden fazla bilgi kanalının izlenmesine müsaade eder. Ortak yolda (bus) meydana gelen gelişimleri izlemek için, bir operatör bir kanalı referans kabul eder ve diğer hatları sıra ile izler. Memnuniyet verici olmamasına karşın, bu teknik yıllardır kullanılan bir teknik olmuştur. Bu çözüm şekli, çok kanallı osiloskopların geliştirilmesi olarak görülmüştür. Bununla beraber bu tür osiloskoplar temelde bazı sınırlamalara uğramıştır. Alternate mod'unda çalışmada, test altındaki sistem, tekrarlamalı olmalı ve osiloskopun ekranında bütün dalga şekillerini senkron edici bir tetikleme işareti sağlamalıdır. Sayısal dünyaya cihaz pazarlayan firmalar ilk olarak sayısal arıza bulucu cihazlardaki bu tetikleme problemini gördüler. Şekil 3-3, doğru olan senkron edilmemiş bir işareti göstermektedir. Ekran, ondalık tabanında 0,1,2,3 sırasını göstermesi gerekirken onaltılı (hexadecimal) tabanında AB4545 sırasını gösterir. "Chop" mod'unda, bir osiloskop test altındaki tipik bir sistemin saat frekansına hemen hemen eş değer maksimum olarak yalnızca bir kaç MHz. lık "chop" frekansına sahiptir. Verim, "n" kanal sayısı olmak üzere, en azından "2n" lık bir düşünüm gösterir. Osiloskop



GÖRÜNTÜ

Şekil 3-3. Osiloskop değişken (alternate) modunda iken osiloskoptaki dalga şekilleri görülmektedir. Özel tetikleme devreleri kullanılmadıkça ve işaretler tekrarlanan türde olmadıkça, ekran üzerindeki dalga şekilleri senkron olmayacaktır.

"chop" modunda iken bilgi, yalnızca $f/2n$ hızından daha küçük olduğu zaman ekrandaki bilgi kabul edilebilir.



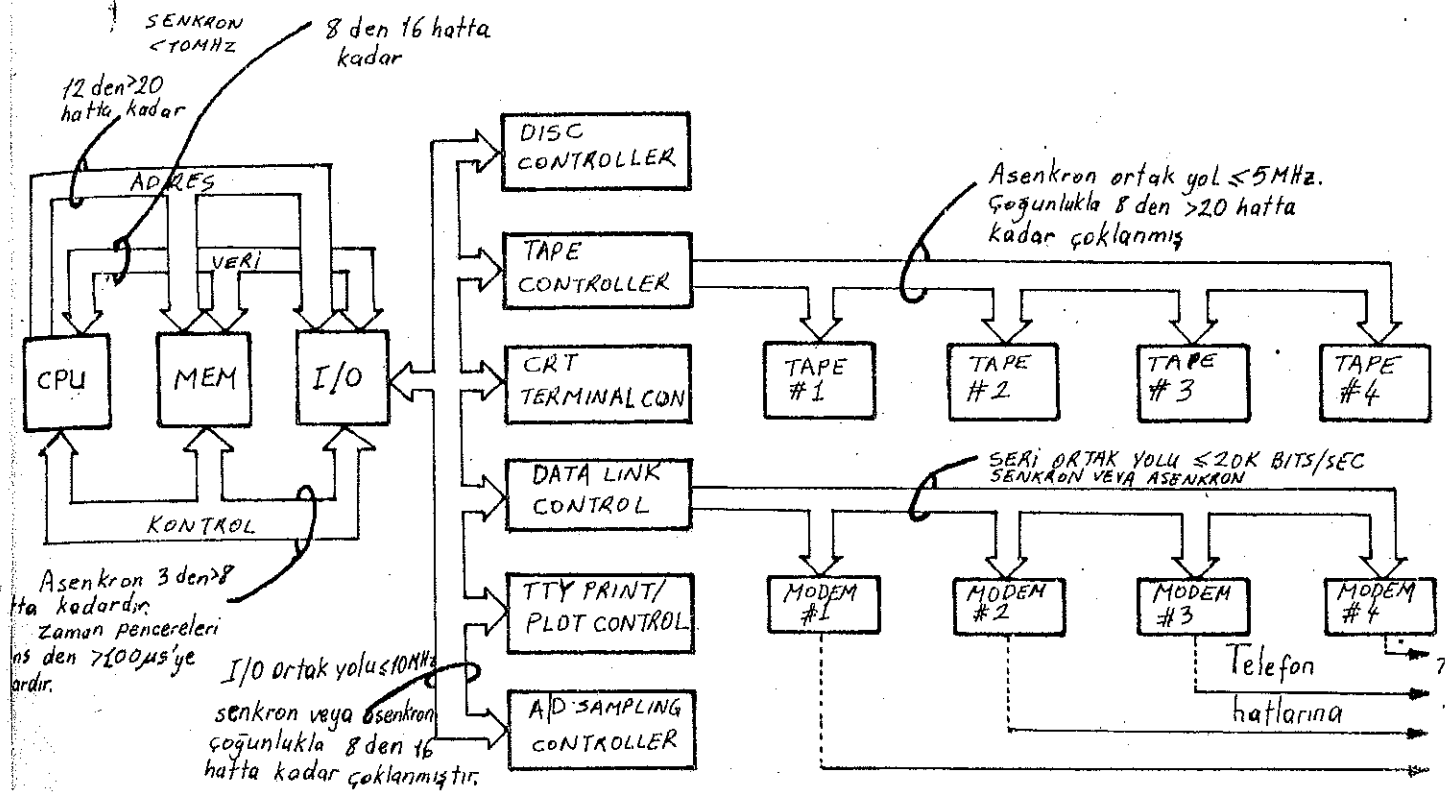
Şekil 3-4. "Chop mode" daki işaretler için osiloskoptaki dalga şekilleri görülmektedir. En yüksek veri oranının (maximum data rate), "chop" frekansının sekiz katı olduğu kabul edilmiştir.

3.4 GEREKLİ SAYISAL SİSTEM ÖLÇÜMLERİ

İŞLEMSEL ÖLÇÜMLER.

Bir mikrobilgisayar geliştirme düzeneği göz önüne alınsın. Bu düzeneklerin çoğu, sistemi adım adım yürütme (single step) yeteneğine sahiptir. İç devre yapısında adres ve bilgi ortak yollarındaki (bus) yalnız LED'ler, her saat çevriminde (clock cycle) bu hatlar üzerinden geçen veriyi gösterir. Bir mikrobilgisayar komutlarından oluşan program verildiğinde, operatör veriyi çözümler ve sistemde nelerin meydana geldiğini belirleyebilir ve sistemin beklenen şekilde çalışıp çalışmadığını anlayabilir.

Örneğin, bir 6502 mikrobilgisayar sisteminde adres, veri, oku/yaz (R/W) ve senkronizasyon hatlarının veri akışına izin verdiğini (latch)



Şekil 3-5. Tipik bir sayısal sistemde ortak yol düzeni görülmektedir.

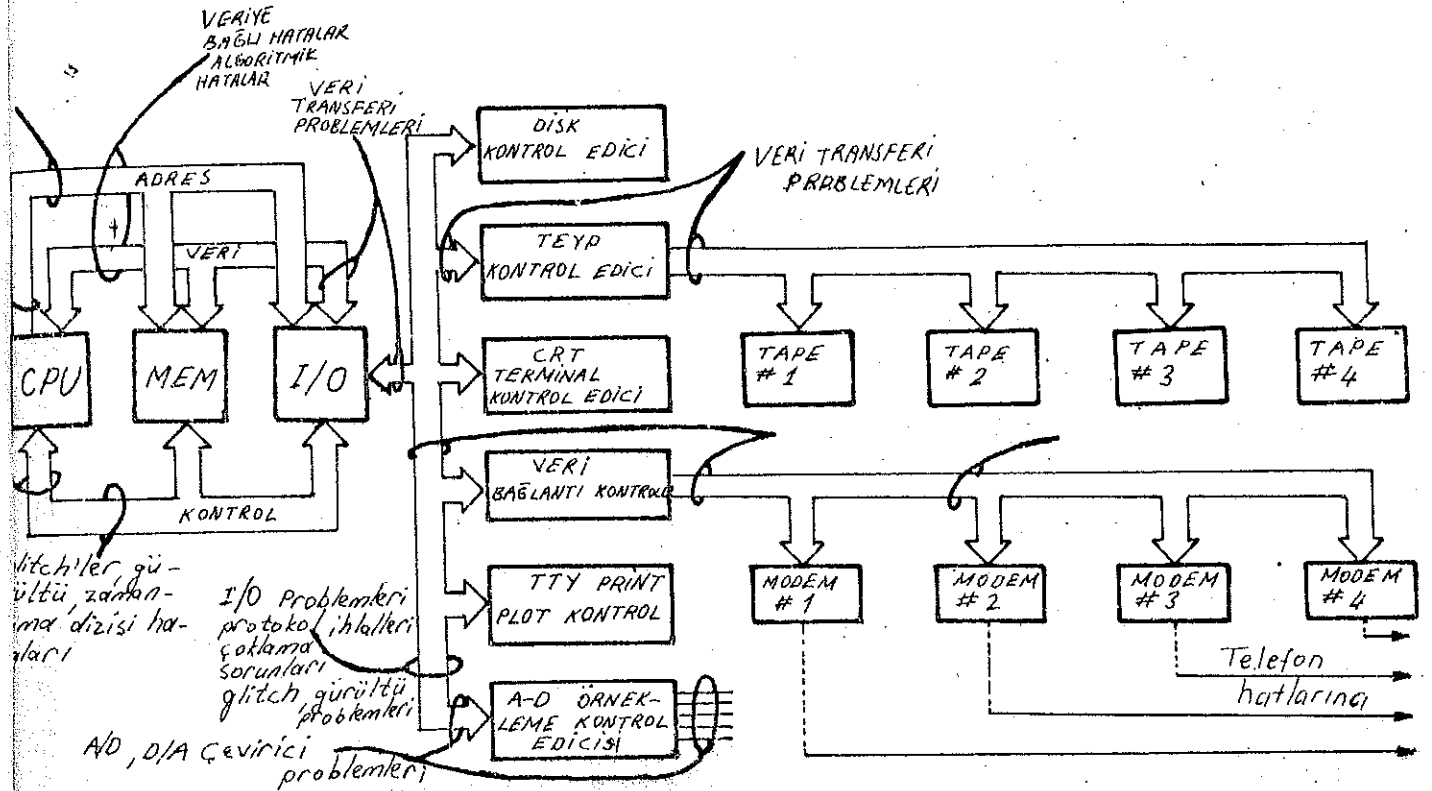
ve dört saat çevrimi boyunca adım adım yürütüldüğünü kabul edelim ve sonuçlar aşağıdaki şekilde çıkmış olsun:

R/W	Syn	Adres	Veri(Data)
1	1	0202	8D
1	0	0203	01
1	0	0204	17
0	0	1701	20

Burada adresler ve veri hatları onaltılı (hexadecimal) şekilde gösterilmiştir. Bulisteleme aşağıdaki biçimde açıklanabilir :

(a). 0202 adresinden, bir sonraki komut için "op code" u okuyunuz.

(8D= akümülatörde sakla)



Şekil 3-6. Tipik bir sayısal sistem ve problem sahaları.

Çoklu ortak yol ölçümleri.

(b). İki ardışıl yerden, akümülatörlerin depolandığı adresleri oku(1701)

(c). Gelen komutu yürüt(icra et). Yani 1701 adresindeki bellek gözüne

akümülatördeki bilgiyi sakla (20)

Bu bilgi parçacıklarını inceleyerek, operatör, hem "hardware" , hem

"software"nin beklendiği şekilde yürütüldüklerini kontrol edebilir. Bu şekilde

sistemin gerekleşmesi ařağıdaki bilgilerin saėlanması ile elde edilir.

(a). Her saat darbesindeki durum

(b). Durum dizisi (sequence of states)

Bir mikrobilgisayarın adres ve veri yolları gibi senkron sistemlerde, durum ve her aktif saat darbesi kenarındaki "durum dizisi" istenilen bilgilerdir. Operatörün, gerilim seviyesinin, test altındaki sistemin eşik (threshold) geriliminin üstünde veya altında olduğunu bilmesi yeterlidir. Bir mikrobilgisayar sisteminin kontrol hatları veya giriş/çıkış (I/O) hatları gibi asenkron yollar için, operatör, durum deėiştiren deėişik işaret hatlarının durum deėiştirme sırasını ve verilen bir durumda her işaretin ne kadar uzunlukta olduğunu bilmek zorunluluğundadır. Durum-zaman (state-time) ilişkisi önemlidir. Buna karşın gerilim-zaman ilişkisi önemli değildir. Bölçümler, işlemsel ölçümlerdir. Tam bir fonksiyonel analiz, verilen ayırık (discrete) zaman aralıklarında sistemin durumunu ve işaretin şekil deėiştirdiėi zamanı gösterir.

ARANAN ÖLÇÜM YETENEĐİ.

Fonksiyonel ölçümleri daha ayrıntılı olarak incelemek için şekil 3-5 verilmiştir. Şekil 3-6 da örnek bir sayısal sistem kuruluşunu ve gözlenmesi gereken olaėan problemler gösterilmiştir. İlk ölçüm türü, durum akışını izleyen bir sistemde yapılabilir ve deėişik lojik durum sıraları meydana gelebilir. Bu teknik yaklaşım, şekil 3-6 daki birinci grup problemleri açıklar ve çözümler. Aynı anda oluřan olayları izlemek, adres ve veri yolu olaylarında karşılıklı ilgiyi göstermek için durum akışı izlemesi, en azından adres yoluna ve tercihan, aynı derecede veri yoluna olan bağlantıyı meydana çıkarır. Bu yollar, 10 MHz. den ařağıda çalıřan, toplam 20

ilâ 40 hat ile aynı anda oluşturulur. Bu yollardaki bilgiler çoğaltılabilirliklerinden, bazı tanımlamalar aranacaktır. Grup 1. deki sorunlarla ilgili diğer ölçümler, bekleme çevrimlerinin (loop) zamanlamalarını gerçekleştir-
mek için seçilmiş durumlar arasındaki zaman aralıklı ölçümlerdir.

Adres ve veri ortak yollarını izleme yeteneği ile birlikte, herhangi bir izleme cihazı için ikinci önemli kriter, istenen kelime (desired word) veya kelime yahut olayların dizilerinin görüntülenmesini sağlayacak pencereleme (to window) yeteneğidir.

İkinci grup ölçümler kontrol ortak yolunda (bus) karşılaşılan problemler ve sistemin program çalışmasında bu sorunların etkisi ile ilgilidir. Temel ölçümler, zaman aralıkları ve olayların dizilimidir. İzleyici cihaz belirgin zamanlama sıralarının ve dar sıçramaları (glitch) göstermek ve tayin edilen zaman süresince oluşan durumları ve belirlenen dar sıçramaları görüntülemek ve bunlarla ilgi kurabilmek yeteneğine sahip olmalıdır. Bu ölçümler hem adres hem de veri ortak yolundaki hareketlerle bağlantılı olduklarında daha fazla önem kazanırlar.

Grup 3 de sıralanan problemler, bilgisayar veri ortak yolundan giriş/çıkış (I/O) yoluna veya kapısına (port) veya I/O ortak yolundan dıştaki diğer bir cihaza yapılan veri transferi ile ilişkilidir. Bir sayısal sistemin muhtelif I/O ortak yolları, elde bulunan dıştaki (harici) cihazların tiplerine bağlı olarak pek çok değişik çizelgeler kullanırlar. Zaman uyumlu (senkron) veya zaman uyumsuz (asenkron) seri iletişimler (communications) kullanılmaktadır. IEEE-488 cihaz bağlantı (interface) ortak yolunun, verilerin paralel fakat asenkron olarak değişen iç içe üç hat üzerinden gönderilmesi nedeniyle ek cihazlara ihtiyaç vardır. Basit izleme teknikleri bu

hatları gözlemeye yeterlidir. Fakat bilgi, veri gruplarına veya her ortak yoldaki program akışının diğer ortak yoldaki ile ilgisi karşılaştırıldığında daha çok anlamlı olur.

Durum (state) veya zaman (timing) olan bir lojik analizörün önemli bir niteliği, bir tetikleme durumunun görülmesinden önceki verinin yakalanması ve gözlenmesi yeteneğidir. Böylece probleme yönelik durumların gözlenmesi olanağı gelişmiş olur. "Eksi zaman" (negative time) yeteneği, bir osilaskop aracılığı ile elde edilemeyen bir özelliktir.

DÖRDÜNCÜ BÖLÜM

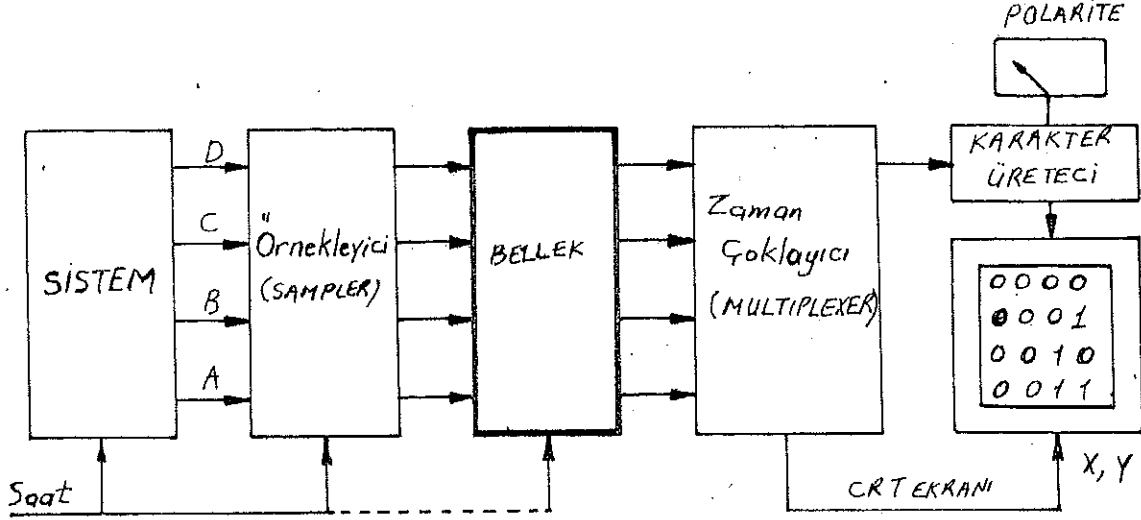
LOJİK DURUM ANALİZÖRÜ

4.1. GİRİŞ

Geleneksel arıza bulma aracı olan osiloskop, modern sayısal sistemlerin çok kanallı veya çok düğümlü izleme isteklerini karşılamaz. Bir sayısal sistemde durum ve durum dizilimini izlemek için osiloskop yerini "lojik durum analizörüne" bırakmıştır. aşağıda lojik durum analizörünün tasarımı ve bazı özellikleri ele alınmıştır. Öncedende belirtildiği gibi, göz önünde tutulması gereken en önemli kriter, lojik durum analizörünün çok kanallı olmasıdır. Basit bir mikrobilgisayar tabanlı sistemi izlemek için en azından kanal sayısının 24, adres hattının 16 ve veri hattı sayısının da 8 olması gereklidir. Daha verimli bir sistemi izlemek için 32 veya daha da büyük sayıda kanala ihtiyaç vardır. Tam bir bilgisayar 100 izleme kanalından daha fazla bir kapasiteyi kullanılabilir.

Lojik durum analizörü test altındaki sayısal sistemi izler, durum ve dizi akışını görüntüler. Kural olarak durum dizisi, "ötelemeli yazıcı" (shift register) bellek kullanarak yakalanır. Örnek alınması, test altındaki sistemin saati ile aynı anda (senkron) sağlandığında, veriler belleğin içine alınır. Saat kaldırılınca yazıcıda durum dizisi dondurulur ve analizörün ölçümü olarak görüntülenir.

Şekil 4.1. de lojik durum analizörünün, lojik durumu belirleyen giriş devresi, bir örnekleyici (sampler) olarak gösterilmektedir. Bu ne-



Şekil 4-1. Veri durumu ve sırası, ötelemeli yazıcıda belleğin içine kelime kelime yerleştirilir(latch). Çıkış devresi, belleklerin tümünün "bit-bit" olarak alır ve veri ve seçilen lojik polariteye bağlı olarak, buları "1" ve "0" lar olarak liste halinde görüntüler. Bu örnekte, durum yatay olarak (DCBA) , sıra ise, yukardan aşağıya dikey olarak gösterilmiştir.

denle örnekleyici, test altındaki sistemdeki aynı lojik özellikleri içerir. Değişik lojik aileleri izlendiklerinde, operatör test altındaki sistemin lojik eşik gerilim seviyesini ayar edebilme yeteneğine sahip olmalıdır.

4.2. LOJİK DURUM ANALİZÖRLERİ

Özet: Lojik durum analizörleri, bilgisayarlar, mikrobilgisayarlar gibi ardışıl işlemcilerde (sequential processor), test, araştırma ve arıza

arama aracı olarak, program sırasını ve veri akışını izlemek ve ölçmek için kullanılırlar. Bu ölçümlerin uygulanması, lojik durum analizörlerince gerçekleştirilen araştırma ve arıza bulma teknikleri, bu kısımda örnekler kullanılarak tartışılmıştır.

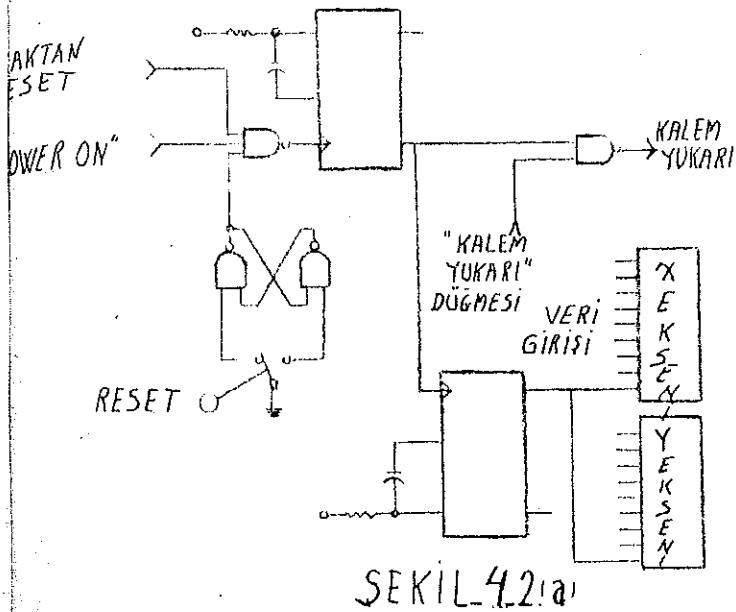
Yukarıda anılan ölçmelerin yapılabilmesi için analizörün sahip olması gerekli olan özellik ve yetenekleri ortaya konulmağa çalışılmıştır. Böylece, bu ölçme özellikleri ve yetenekleri, bir lojik analizörün fonksiyonel tanımını oluşturmaktadır.

Bu ölçme yetenek ve özelliklerinin uygulamadaki kullanım biçimlerini göstermek üzere bir blok diyagram verilmiştir. Bu blok diyagram, lojik veya veri dizisini ve bu ölçmelerin, arıza aramasında, araştırmalarda ve ardışıl sayısal makina ve sistemlerde uygulanma biçimini açıkça gösterir.

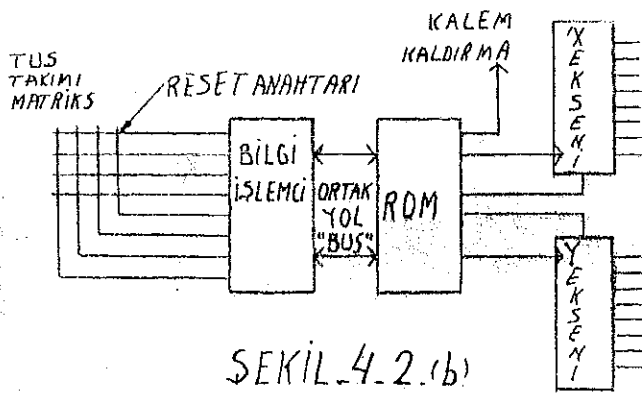
Lojik durum analizörleri, sayısal işlemcilerin performanslarının ölçülmesi için tasarımı yapılmış olan yeni cihazlardır. Cihaz, işlemcinin içindeki veri akışını izler ve bu verileri bir osiloskop izi gibi kolayca okunabilir biçimde ve analog işaretlerle gösterir. Spektrum analizörleri bu verileri çözümler ve görüntülerler.

Çok önemli görevler üstlenen bir işlemci (processor) veya bilgisayar tabanlı sistem veya cihazlarda test, araştırma ve arızalarının belirlenmelerinde özellikle kullanılacak araçlar lojik durum analizörleridir.

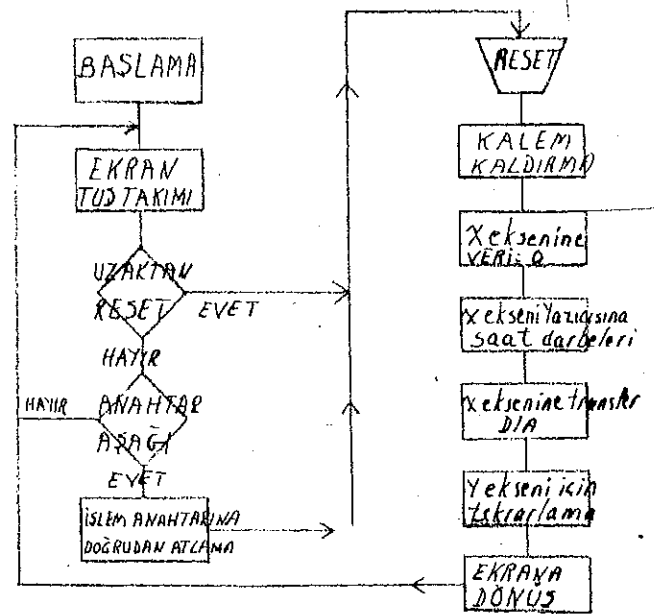
Şekil-4-2(a) kombinezonsal lojik kullanılarak meydana getirilen bir sayısal çizicinin (digital plotter) bir kısmına ait şematik görüntüdür. Şekilde, reset anahtarı ve kalemi kaldırıp çizelgenin sol alt köşesine döndürücü lojik devre görülmektedir. "Remote reset" ve "power on" anahtarları yanlış (false) durumunda (yüksek) (lojik 1) oldukları sürece, reset anahtarı tarafından



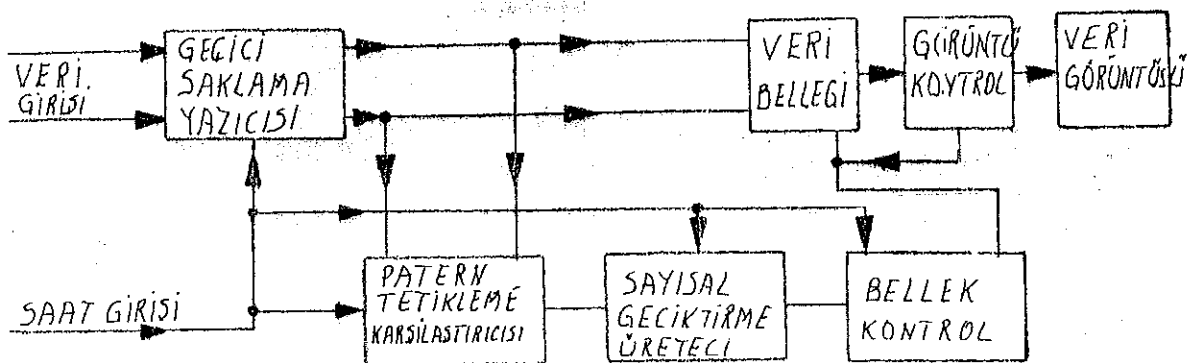
SEKIL 4.2(a)



SEKIL 4.2(b)



SEKIL 4.2(c)



ŞEKİL - 4-2 (d)

üretilen işaret, bir osiloskop veya lojik "probe" aracılığı ile oldukça kolay olarak bir düğümden bir sonraki düğüme kadar izlenebilir. Böylece, "reset" anahtarına basıldığı zaman, kalem, çizelgenin sol alt köşesine dönmez ise reset işareti izlenir, kaybolduğu yer görülünce problemin nedeni belli olur. Bu kombinezonsal lojikte arıza aramada basit bir problemin tipik bir örneğidir. Bu problemde iki gerçek unsur ilgi çekici olup, bunlardan birincisi, reset anahtarından gelen işaret, zaman ve aralık olarak izole edilmiştir. Yani, reset anahtarına basılmadıkça reset yolunda hiçbir işaret yoktur. Reset yolu diğer işaret yollarına nazaran değişik düğüm dizilerine sahiptir. Böylece yol üzerinde görünen herhangi bir işaret, bir reset işareti olmalıdır. İkinci nokta ise, sayısal lojik konusunu bilen bir kişinin şematik diyagramı incelemekle sistemin çalışmasını kolayca anlayabilmesi hususudur. Yani şema ve (IC: integrated circuits) tümdevreler kataloğu kombinezonsal lojik sistemin çalışmasını tanımlamaya yeterlidir. Şekil 4-2(b) fonksiyonel olarak benzer olan fakat mikrobilgisayar kullanılan bir çizicinin şematik diyagramıdır. İşareti reset anahtarından x ve Y eksen yazıcılarına kadar izlemek artık mümkün değildir. Aynı zamanda bu şema, çizici reset devresinin çalışması hakkında pratik olarak hiçbir bilgi vermez. Lojik ve mikrobilgisayarlarla ilgili olan birine reset anahtarına basmanın sonucunda nelerin olabileceği hakkında da hiçbir şey ifade etmemektedir. Reset anahtarına basıldığı zaman neleri meydana geleceğinin tayin edilebilmesi için gerekli bilgi şekil-4-2(c) de gösterilmiştir. Yani işlemsel tabanlı bir sistemin (processor based system) çalışmasının

açıklanabilmesi için böyle bir akış diyagramına ve program listelemesine ihtiyaç vardır. Örnekteki çizici kalemin, reset anahtarından gelen komuta cevap olarak , uygun olan başlangıç durumuna dönmeyişinin (reset olmalığının) nedenini tayin edebilmek için, tuş takımındaki (keyboard) ekran üzerinde , program akışını ve reset programlarındaki (reset routines) yanlışlık noktalarını izlemek gereklidir. Bu bir osiloskop veya lojik probe ile yapılabilir. Çünkü , programın hangi basamağının yürütülmüş olduğuna bakılmaksızın , şekil-2. deki ortak yolda , bir tür faaliyet vardır. Bildiğimiz bu arıza bulma cihazlarının çalışmayış nedeni , ortak yoldaki işaretin veri ile birlikte olması şartıdır. Yani , program yürütülmesinin izlenebilmesi için, ortak yoldaki her hat üzerindeki işaretler aynı anda kontrol edilmelidir. Özellikle, kalemin uygun olarak reset olmayışının nedenini öğrenmek için , program adreslerinin sırası , gözlenmelidir. Şekil-4-2(c) çizici (plotter) örneğinde , program adreslerini izleyerek tuş takımı tablosu tarayıcısı (keyboard scan) çalışıyorsa , reset anahtarının tanınıp tanınmadığı,verinin X ve Y yazıcılarına gönderilip gönderilmediği v.b silgileri elde edilebilir. Yanlış bir program yürümesi tesbit edildiği zaman, bir skop , lojik probe veya diğer bir test cihazı problemin oluşum nedenini anlamak için kullanılabilir. İlk problem , yanlış program yürümesinin nerede ve ne zaman oluştuğunun bulunması isede bundan sonra, deneme ve yanılma metodu yerine, usulüne uygun bir biçimde bazı elektriksel problemlerin araştırılmasına geçilebilir.

Açıkça belirtmek gerekirse, bir işlemcinin (processor) çalışması,

yanlızca onun iç yapısındaki veri etkileşimleri (internal data transactions) vasıtası ile tayin edilir. Bu veriler, kuramsal bir bakışla, elektriksel işaretlerin aksine, zaman domeni yapısında değıllerdir. Elektriksel işaretler, sadece veri taşırlar. Bu işlemciler veri domeni makineleridir. Veri domeninin zaman domeninden olan farklılığı , frekans domeninin zaman domeninden olan farkına benzemektedir. Maalesef, veri domeni ve zaman veya frekans domeni arasındaki ilişkiyi tanımlayacak, örneğın bir Fourier dönüşümü biçiminde benzer bir bağıntı yoktur. Veri domeni , bilginin birden fazla araç vasıtası ile gönderilebildiğı herhangi bir işlemi içerir. Burada , yalnızca ardışıl sayısal işlemciler (sequential digital processors) olarak üretilen , veri domeni makineleri ile ilgili durumlar söz konusudur.

Veri göndericisi veya veri alıcısı, işlemcinin elektriksel tasarımında tanımlanan veri işlem (data transaction) kurallarına uymazsa ardışıl işlemciler doğru bir veri dizisi (data squence) üretemezler. Yani bir kapı , bir yazıcı v.b. gibi devreler başarısız olurlar. Böylece zaman domeni bir soruna dönüşmüş olur ki bu da en iyi şekilde ancak bir zaman domeni cihazı ile çözümlenir.

Lojik durum analizörü, bir veri domeni cihazı olup, veri işlemlerini , açıkça çözümlenebilmeleri yönünden kolayca okunabilecek biçimde görüntüler. Lojik durum analizörleri , deney altındaki cihazın verileri gördüğü gibi , mümkün olduğu kadar en benzer biçimde program akışını veya veri işlemlerini görmeğe ve görüntülemeğe yetenekli olmalıdır. Tıpkı osiloskoplar gibi zaman domeni işaretlerini, en düşük düzeyde doğrusal olmayış (nonlinearity), en az sıçramalar (overshoot) veya en az bant genişliği sıçramaları (band width degradation) ile göstermeli ve spektrum analizörleri gibi , frekans domeni işaretlerini en düşük gürültü, en az harmonik ve genlik bozulmaları ile göstermelidir.

Verileri, test altındaki sistemin gördüğü gibi görmek üzere veriler, ikili veriler (binary data) olarak okunmalıdır. Eşik (threshold) gerilimi veya bir lojik "1" ile bir lojik "0" seviyesi arasındaki aralık

test altındaki makinada kullanılan lojik ailenin eşik gerilimine mümkün olduğu kadar yakın olmalıdır. Dar genlikli sıçramaların (glitches) varlığı veya önceden tayin edilemeyen lojik seviyelerin meydana çıkarılması için bazı çift-eşik (dual threshold) uygulamaları hakkında daha fazla bilgiler sağlayan açıklamalı cetveller (schemes) zaman domeni olmakla beraber, bir lojik analizör ile veri işlemlerindeki hataların bulunmasından sonra çok yararlı olurlar.

Tüm veri kelimelerinin derhal izlenebilmesini sağlamak için, lojik analizörler yeterli girişe sahip olmalıdırlar. Eğer kelimelerin yalnızca bir parçası izlenebilirse, veri tam olarak tanınmaz. Bu durum bir osiloskop izlemesinin veri domeni eş değeri olur ki ekrandan dikey olarak kayarak çıkar veya incelenilen frekansların ancak yarısını kapsayabilen bir spektrum analizörün eş değeri olur. Lojik durum analizörü ile test altındaki sistem, giren verileri aynı anda okumalıdırlar. Bunun anlamı: Bilgilerin, lojik durum analizörüne ve test altındaki sisteme, aynı saat darbesi tarafından gönderilmesinin sağlanmasıdır. Modern sayısal tasarım uygulamalarında, lojik durum analizörlerinin, sıfır veri tutma zamanını (zero hold time) ve saat girişine nazaran mümkün olduğu kadar kısa veri oluşma zamanına (data setup time) sahip olması istenilir. Çünkü, bir veri uygulandığı anda, aktif saat kenarından hemen sonra veri değişmeye başlar ve bir sonraki aktif saat kenarından çok kısa bir zaman önce kararlı hale gelir.

Lojik durum analizörü, test altındaki sistemi mümkün olduğu kadar az etkilemelidir. Test cihazının, test altındaki sistemin çalışma verimini değiştirmemesi vaz geçilmez bir gerçektir. Yine, her hangi

bir cihaz normal çalışma durumunda test edilmelidir. Bir işlemci, tek tek adımlama (single step) durumunda ve 5 MHz. saat darbesinde çalıştırılır. Bunun gibi programı izlemek için pek çok sayıda çıkış (yazıcı veya ekran gibi) ünitelere eklemek, bir işlemci programı için en başta gelen değişikliklerdir. Bunların her ikisi, yüksek hızlı doğrusal bir kuvvetlendiricinin (high speed amplifier), düşük hızda çalışmasının eş değeridir ki, sınırlı bir bant genişliği olan bir osiloskop ile test edilebilir veya bir spektrum analizörünün girişini tahrip etmemek için onda bir güçte çalışan bir verici (transmitter) kullanmaya benzer.

Girişteki bu isteklere ilave olarak, okunabilmesi ve yararlı olabilmesi için skopta aranan pek çok özellikler vardır. Yararlı bir veri dizi cetveli elde edebilmek için, lojik durum analizörü bir tür tetikleyiciye sahip olmalıdır, böylece verinin özel bazı bölümleri görüntülenmek için seçilebilir. Görüntülenen bilgi, ikili veri (binary data) şeklinde olduğundan, lojik tetikleme bazı veri biçiminin veya veri kelimesinin (data word) oluşumu şeklindedir. Yani veri kelimesi, önceden kararlaştırılan tetikleme kelimesinin tamamen aynı olduğu zaman, bilgi depolanması başlar veya durur. Program akışının izlenmesinde, tetikleme kelimesi bir alt program giriş noktasının adresi olabilir. Sayısal kontrollu bir yazıcıda arıza aramasında ise tetikleme kelimesi "A" harfi için kod olabilir.

Veri dizi cetvelini daha kolay okunabilir hale getirmek için, lojik durum analizörü daha pozitif gerilimi bir lojik "1" ile ve en negatif seviyedeki gerilimi bir lojik "0" ile veya bu sıralamayı tersine

yapmak suretiyle eşleştirebilir, böylece kullanıcı gerilimden veriye olan bu dönüşümü, bir kağıt ve bir kalem yardımıyla yapmaya mecbur kalmaz. Ek olarak veri akışı dizisinin yazılması için genellikle kullanılan bir yöntem, en belirgin bit solda, en az ağırlıklı bit sağda ve ilk kelime en yukarıda ve onun arkasından gelen her kelime, kendinden önce gelen kelimenin altında olacak biçimde dizilmelidir. Bu biçim, geniş ölçüde kabul edildiğinden ve çoğu insanlar için çok doğal görüldüğünden, lojik durum analizörünün, bir durum sırası cetveli sunmak için kullanması gereken çizelge bu şekilde kabul edilmiştir.

Butemel özellikleri içeren bir lojik durum analizörünün genel görüntüsü "4-2(d)" şeklinde görülmektedir.

Olağan çalışmada, veri girişleri, analiz edilmek üzere bazı veri ortak yoluna (data bus) veya işlemcideki yazıcıya bağlanır. Saat girişleri sistem saatine bağlanır. Bundan sonra, veri gerekli yerleşme zamanını azaltmak için mümkün olduğu kadar az karıştırma ile (interference) geçici depolama yazıcısına (temporary storage register) örneklenir. Saatte ve veri yolundaki nisbi gecikme o şekilde düzenlenir ki, durumları en kötü tutma zamanı (hold time), mümkün olan en uzun saat gecikmesi ve mümkün olan en kısa veri gecikmesi "0" olur. Saat gecikmesi mümkün olduğu kadar kısa ve veri gecikmesi mümkün olduğu kadar uzun olduğu takdirde en kötü durum yerleşme (set up) zamanı meydana gelir. Veri ve saat yolundaki lojik eşik gerilimini en düşük düzeye indirmekle her iki yoldaki en uzun ve en kısa gecikmeler arasındaki fark azaltılabilir, bu da lojik durum analizörünün yerleşme zamanına olan ihtiyacını en düşük düzeye indirir. Kuşkusuz, geçici

saklama yazıcısı tam olarak kenar tetiklemeli "flip-flop"lar (edge-triggered flip-flops) dizisi şeklinde oluşur. Biçim (pattern) tetikleme karşılaştırıcısı (pattern trigger comparator) geçici depolama yazıcısında saklanan veri ile bir anahtarda veya depolama yazıcısında önceden yerleştirilmiş olan "biçim" i karşılaştırır. Geçici depolayıcıda saklanan giriş verisinin her biri, halen var olan tetikleme kelimesinin her bitine eşit ise, karşılaştırıcının çıkışı doğru (true) (lojik 1) olur. Yine, tetikleme amaçları ile her hangi özel bir kanalın kapatılabilmesi de istenilir.

Bu "biçim" tetikleme karşılaştırıcısının çıkışı, bir sayısal geciktirme üreticinin (digital delay generator) çalışmaya başlatılması için kullanılır. Bu üretici, tetikleme önceden belirlenen giriş saat darbesi sayısı kadar geciktirir. Geciktirme üretici, uzun program dizilerinin analizine imkan sağlamak için, sabit bir tetikleme noktasından itibaren tetikleme 10000 veya 100000 sayı kadar geciktirebilme-ye muktedir olabilmelidir,

Geçici depolama yazıcısının çıkışı, daha sonra görüntülenmek amacıyla ile veri belleğinde de depolanır. Veri belleği, saat darbeleri hızında çalışabilecek kadar hızlı olmalıdır. Belleğin yaz komutu darbe genişliğinin (write pulse width), giriş saat darbesi periyodunun yaklaşık yarısını geçmemesi genel bir kuraldır. Bellek, kuşkusuz giriş verisi kadar uzunlukta ve en az 16 kelime derinliğinde olmalıdır.

Sayısal geciktirme üreticisinden çıkan geciktirilmiş tetikleme darbesi, bellek kontrolüne giden bir komut işaretidir. Bu tetikleme komutunun alınması üzerine, bellek kontrolü bir operatörün cihaz

kontrolüne (set control) bağılı olarak, veri belleği tarafından veri depolanmasını başlatabilmeli veya durdurabilmelidir. Geçici depolama yazıcısındaki veri akımının, veri belleğine devamlı olarak dolması bellek kontrolü tarafından sağlandığı zaman , en yüksek süratli sistem çalışması elde edilmiş olur. Bundan sonra bellek, o anda mevcut olan "yaz" komutu darbesinin sonu ile, bir sonraki darbenin başlangıcı arasındaki sürede, veri depolama işlemini durdurur. Bellek kontrolünün görüntüyü başlatacağı zaman geciktirme üreticisinden (delay generator) gelen bir tetikleme darbesi bir sayıcıyı tetikler. Bu sayıcı, veri belleğini doldurmak için tetikleyiciyi dahil onu takip eden yeterli sayıda veri kelimesini sayar. Bellek kontrolünün görüntülemeyi sona erdireceği zaman geciktirme üreticisinden gelen tetikleme darbeleri, bir veri depolama çevriminin (data storage cycle) başlangıcında red edilirler. Bu durum bir veri belleğinin birini takip eden verilerle dolmasına kadar devam eder. Bundan sonra, geciktirme üreticisinden gelen bir sonraki tetikleme darbesi, veri alınmasını derhal durdurur, böylece bellek, tetikleme kelimesini ve tetikleme kelimesinden hemen önce gelen verileri kapsar. Önceden belirlenen (preset) bazı koşullar meydana geldiği zaman verinin veri belleğine depolanmasını sağlamak için bellek kontrolünü ayarlamak mümkündür. Örneğin, zaman paylaşimli ortak bir yolda (time shared bus), yalnızca program adresleri depolanabilir, diğer bütün veriler kabul edilmezler. Depolama süresinin sonunda veri belleğinin kontrolü görüntü kontrolüne devredilir.

Görüntü kontrolü veri belleğini doğru olan sırada okur, bu verileri

bir çizelge üzerine sıralar ve verileri görüntüleyen veri skop'una gönderir. Tek ve en faydalı görüntü şekli durumların sırasını gösteren durum tablosu (state table) dir. Bu çizelge genellikle lojik '1' ve lojik '0' lardan oluşur ve zaman-gerilim veya durum-gerilim biçimindeki bir görüntüden daha kolay okuma olanağı sağlar. Görüntü kontrolü (display control) aynı zamanda belleğin kontrolünün bellek kontrolüne (memory control) dönmelerinden önceki periyotta verinin görüntülenme süresini belirler. Görüntü kontrolü operatör tarafından el ile "reset" edilinceye kadar, bilgiyi sınırsız olarak görüntüleme yeteneğine sahip olmalıdır. Yeni ve rilerin yakalanıp görüntülenmesinin sağlanabilmesi için, görüntü kontrolü önceden belirlenen ve ayarlanan bir sürenin sonunda, kontrolün bellek kontrolüne otomatik olarak dönmelerini de sağlayabilme yeteneğine sahip olmalıdır.

1. Tek örnekleme biçimi (single sample mode) tek veya sık tekrarlanmayan veri dizisini gözlemede faydalıdır.

2. Tekrarlamalı biçim (repetitive mode) özel bir sıralamayı veya bir değişimi yahut da kararsız bir diziyi (unstable sequence) araştırırken faydalıdır.

Görüntü skop'u , bilgileri görüntüleyebilen herhangi bir türde olabilir. Bir CRT görüntü skop'u durum dizi tablosunca (state sequence table) üretilen çok sayıda verilerin görüntülenmesinde faydalı olmuştur.

Lojik durum analizörleri, veri domeni cihazlarının performans analizleri için çok faydalı araçlardır. Veri domeni, ardışıl sayısal işlemciye veri gönderilişi gibi, bilginin , bilgi taşıyıcıdan bağımsız olarak işlem görmesi şeklinde tanımlanabilir. Temel bir lojik durum analizörü,

program adresleri gibi özel bir veri işlem sırasını yakalar ve görüntüler. Bu program adresleri, durum (state) veya veri (data) tanıyıcı bir tetikleyici tarafından seçilir. Bu veri işlemlerini analiz etme yeteneği, mikrobilgisayarlar ve bilgisayarlar gibi veri işlem makinalarında deney , araştırma ve arıza bulunması sırasında çok önemlidir.

4.3. EKРАНLAR

İkili (binary) liste veya tablo:

Bir basit lojik durum analizöründe, analizör belleğinin tüm içerikleri, bir LED dizisi kullanarak ya kelime kelime veya bir anda çok sayıda kelime olarak görüntülenebilir. Bir diğer şekilde ise, bellekteki bilgi, bir osiloskop üzerinde sahte durum +zaman görüntüsü meydana getirmek için kullanılabilir. Her iki görüntüleme şeklinde, sonuçlar fonksiyonel şekle konulabilir. Sayısal sistemlerde bir fonksiyonel görüntüleme, program listelemesine çok yakın eş değerde olan liste veya tablo şeklindedir. Liste veya tablo oluşturma devresi şekil 4-2 de gösterilmiştir. Her bellek biti veya yeri sıra ile zaman çoklama (multiplexer) ünitesince test edilir ve lojik '1' veya '0' dan uygun olanını ekrana gönderir.

Ek kontroller şunlardır:

1. Pozitif veya negatif lojik seçimine izin veren "lojik polarite kontrolü".
2. Ekran üzerinde, izin istenen kısmını görmek için ötelemeyi sağlamak veya bellek yer değiştirmesini uygulayabilmek için "öteleme kontrolü" (çünkü ekran boyutları sınırlıdır.)
3. Analizör görüntüleme şeklini seçerken kelime genişliğini belirtmek

için "kelime genişliği kontrolü".

Şekil 2-2(c) deki CRT ekranı, liste veya tablo olarak adlandırılır ve operatörün, durum ve sıra akışı yönünden çok ayrıntılı bilgiler elde etmesini sağlar. Her yatay hat, test altındaki sistemin kelime veya durumunu gösterir. Dikey sıra ise, ekran, zaman yönünden ayırık anlarındaki durum sırasını veya zaman olaylarını gösterir. (DIT:Discrete instances in time) bilgi, her saat darbesinde örneklenirse ve saat frekansı sabit ise, zaman aralıkları da sabit kalacaktır. Bununla beraber pratikte böyle olmayabilir.

Formatlanan Listeler

Sayısal cihazlarda, sonuçların ikili tabanda gösterilmesi, gereksiz yere uzun görünümlü olup, aynı zamanda okunması sırasında hata yapabilecek bir düzenlemedir. Okumayı kolaylaştırmak için iki rakamlı sonuçlar, üçlü veya dörtlü bit'lerden oluşan gruplar halinde kümelendirilir. Daha belirgin bir anlatımla, gruplar ya 8 tabanında (octal) veya 16 tabanında (hexadecimal) eşdeğeri olarak görüntülenebilir. Örneğin,

Sekiz tabanında:1010010111001100 : 1 010 010 111 001 100 : 122714₈

Onaltı tabanında:1010010111001100 : 1010 0101 1100 1100 : A 5 C 16

Bazı cihazlar, ondalık (decimal) olarak da görüntülerler. Yukardaki iki rakamlı sayıların ondalık eşiti şöyle olur:

$$2^{15} + 2^{13} + 2^{10} + 2^8 + 2^7 + 2^6 + 2^3 + 2^2 = 32768 + 8192 + 1024 + 256 + 128 + 64 + 8 + 4 \\ = 42444_{10}$$

Bazı uygulamalarda, giriş giriş gruplarına değişik şekillerde tanımlamak fayda sağlar. Örneğin bir cihazın bağlantı ünitesi (interface) ortak yolu gözlenirken, karışık bir okuma düzeni kullanmak faydalı olur. Verilerin

sekizli, onlu veya onaltılı olarak anlatımı daha uygun olduğu halde , işletme (management) hatları en iyi şekilde iki tabanında gözlenebilir.

Lojik analizör uygulamalarının çoğu mikrobilgisayar, adres, veri ve kontrol ortak yollarını izlemeyi içerir. Genel amaçlı bir analizöre sahip olan bir operatör, cihazını amacına uygu olarak kullanmak için mikrobilgisayar protokolunu incelemek suretiyle ve ilgili "mnemonic"leri, makina çevrimlerini ve bunun gibi ya da benzeri özelliklerini ize dönüştürerek cihaza uygulamalıdır. Özel bir amaca yönelik lojik analizör bu sorunları çözümler ve sonuçları bir bilgisayar programı listesi şeklinde gösterecek tarzda üretir. Analizör listelemesi ve orijinal program arasındaki bir temel fark, analizörün mikrobilgisayarın çalışmasının gerçek sırasını listelemesine karşın, programın bellek dizisindeki bütün çalışmalarını listelemesidir. Bunun anlamı, lojik analizör program listelemesindeki izleme hatlarına ilaveten mikrobilgisayar çevrimlerini izleyecektir. Örneğin bir yükleme (LOAD) komutu bir program listelemesinde kullanabilir.

ADRES	İŞLEM KODU(OP.CODE)/Mnemonic	
A ₀	LDA	Akümülatörü yükle
A ₀₊₁	12	Adresin yüksek anlamlı kısmı= 12
A ₀₊₂	34	" alçak " " = 34
A ₀₊₃	bir sonraki	Bir sonraki komut

Analizör listesinde bu üç hat dört olabilir. Gösterilen dördüncü çevrim komutun yürütüldüğü gerçek çevrim olacaktır.

ADRES	İŞLEM KODU/Mnemonic	
A ₀₊	LDA	
A ₀₊₁	12	Bir hat olarak birleştirilebilir.
A ₀₊₂	34	Yani LDA 1234 olur.
1234	veri	Gerçek olarak komutun yürütüldüğü çevrim

A₀₊₃ bir sonraki

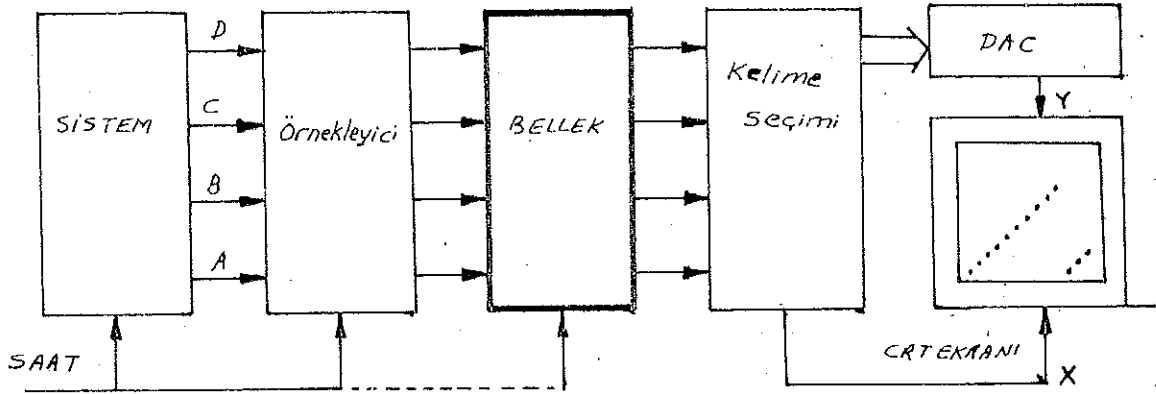
```
0100 0000 1111 1010 0011 0101 0011 1100
1110 0010 0001 1110 1111 0000 0011 1010
0101 0011 0100 1010 1100 1111 0001 0000
1110 0011 1100 0001 1001 0110 1011 0010
```

Durumların listelenmesinde değişik şekillerde görüntülemeler kullanılabilir. Şekilde HP 1600 S lojik durum analizörünün ikili taban kullanılarak oluşturduğu format görülmektedir.

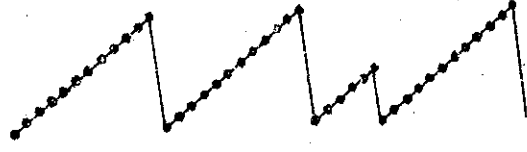
Analizörlerin çoğu daha önceki bir izleme veya mevcut saklı veri ile bir izlemenin karşılaştırılmasını sağlayan "izleme ve karşılaştırma mod'u"nu içerir. Ekran alışlagelmiş liste şeklinde olabildiği gibi, bir EX-OR (Exclusive-OR) tablosu şeklinde de oluşur ki, eşitlikler 0 ile, farklar 1 ile görüntülenir. Karşılaştırma mod'u, üretim testi ile arıza analizi için faydalıdır. Yeni bilgiler A tablosuna yerleştirilmelidir ve önceden saklanan B tablosundaki verilerle karşılaştırılmalıdır. EX-OR görüntüleme şekli yalnızca ayrıntıları vurgular. Arada bir oluşan hataları bulmak için, "halt A-B" tetikleme mod'u ilave edilmiştir. Bu mod hatanın A tablosundaki veri ile B tablosundaki veri arasında bir fark oluşturmasına kadar sistemin beklemeksizin devamlı olarak çalışmasını sağlar ve veri bu noktada daha sonra gözlenmek üzere dondurulmuştur.

Resim Ekranları

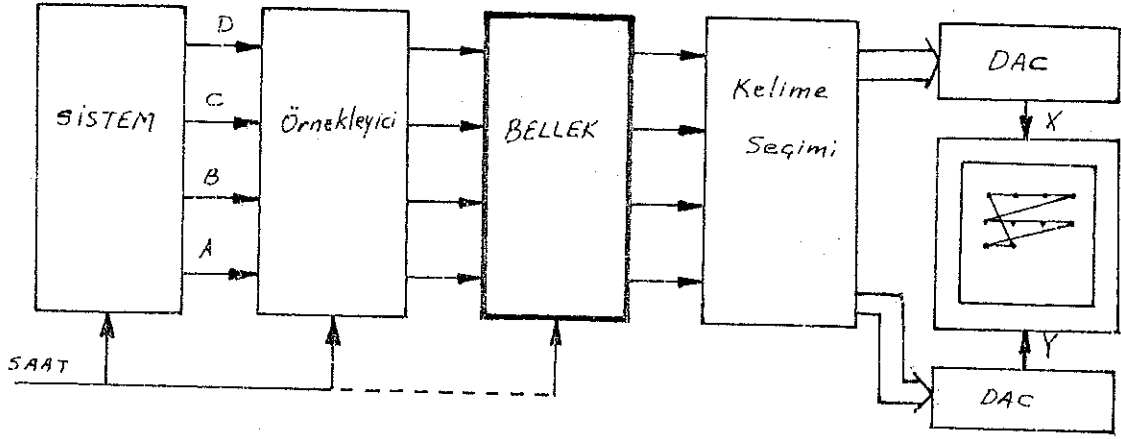
Listeleme şekli (list mode) ile uygulanan veri, hızlı değerlendirme için aşırı derecede ayrıntılı olabilir. Bu durumda "resim" şeklinde görüntüye ihtiyaç vardır. Şekil 4-3 teki devre ile bir grafik çizilebilir. Grafik, bir CRT'nin Y levhasını, her bir bellek kelimesinin analog eşdeğeri ile sürmek sureti ile program çalışmasının sıra ve akışını vurgular. X levhası, seçilen bellek kelimesini veya benzeri bir kelimeye uygun bir işaretlerle sürülür. Test altındaki sistem basit bir BCD sayıcısı ise, grafik bütün "0" durumlarında sol köşeden başlayacaktır. Her saat darbesi ile "bir" bayı yükseldikçe ekran üzerindeki nokta, sağa ve yukarı birer adım hareket edecektir. Bu durum "9" a kadar olan sayılarda devam eder ve bu noktada sayıcı tekrar başlangıç durumuna döner.



Şekil-4-3 Grafik, veri akışı dizisinin bir parçasının resmini sağlar. Bellekteki her kelimenin iki tabanındaki değeri, DAC (sayısal-analog çevirici) aracılığı ile üretilmiştir ve CRT nin Y levhasına beslenmiştir. X saptırma levhaları kelime dizilerini vurgular.



Şekil-4-4 Grafik mod'unu kullanarak, operatör beklenmeyen kesikli hatları araştırır ve bu kesikli hatlar, ardışıl durumlarda ayrılışları ve sıçramaları belirtebilirler.



Şekil-4-5 Bellekteki bir kelimenin her yarısı DAC'lar aracılığı ile CRT'nin X ve Y eksenlerine geçerler. Ekrandaki sonucun yeri veri kelimesine uygun olur. Kelime akışı sonucu bir "harita" oluşur.

Grafik, listenin bir "resmini" sunar ve veri akış sırasını vurgular. "Grafik" mod'unu kullanarak, operatör beklenmeyen kesintili hatları araştırır. Bu kesintili hatlar, ardışıl durumlardan ayrılışları ve sıçramaları belirtir(şekil-4-4). Bu kesintiler liste veya tabloda belirgin olmayabilir. Yeterli bir çözüm (ayırım) sağlayabilmek için, grafik dikey ekseninde kullanıcının belirttiği sınırlara sahiptir. Ayırım yeteneği (resolution) yükseltildikçe, aynı anda gösterilebilecek faal durumların boyutunu sınırlar ve bu nedenle de grafiğin sağladığı görüntülerin adedi sınırlanmış olur.

olur. Bir operatörle bütün faal lojik durumların görüntülerini sağlayan bir skop'la birlikte hâla son derece iyi bir ayırım yeteneği sağlayan bir "harita" şekil-4-5 te gösterilmiştir.

Bu "harita", analizörün veri probe'larındaki faaliyetlerin mümkün olan en geniş şekilde görüntülenmesini sağlar. Her bellek kelimesi ikiye bölünmüştür, her yarım kelime bir sayısal DAC'ı harekete geçirir ve bunların her biri CRT'nin ya X veya Y saptırma lavhalarını besler. "Harita" yalnızca ayrı bitlerin değil, bütün kelimelerin varlığını veya yokluğunu gösterir. Diğer ekranlardan ayrı olarak en çok kullanılan "harita" gösterilmesi şekli "devamlı mod" durumunda sağlanır. Noktalı çizgiler doğru faaliyetlerin oluşup oluşmadığını belirtir ve noktaların parlaklığı, karşılık olan kelime frekanslarının (sıklıklarının) nitelik göstergesi olur. Kelime dizisi, her noktayı kayan bir yıldız görünümünde vektörlerle birleştirerek gösterir. Noktaları birleştiren vektörler daha kalındırlar ve bir sonraki noktaya doğru daha parlaktır.

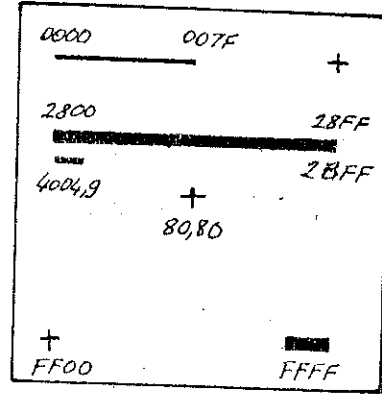
"Harita" görüntülenmesinde, veri durumu veya pozitif lojikteki bütün "0" bellek kelimeleri sol üst köşede görüntü noktası üretir. Aynı anda bütün "1"ler en sağ alt köşede bir nokta üretirler. Şekil-4-5 te "harita" dört hat veya 16 bitlik görüntü için ölçeklenmiştir. Bir BCD sayıcısı ile görüntü, bütün sıfır (0000) yerlerinden başlar ve sonra 0100'a ilerler ve böylece devam eder. Bir BCD sayıcısıyla devre 0110 durumundan hemen sonra reset durumuna getirilir. Ekranda bir BCD sayıcısının haritasını görüntüler. Sayıcı yanlış çalışırsa harita değişecektir. Bu değişim tecrübesiz bir operatör tarafından dahi görülebilecektir. Pratikte 16 değişik durumdan daha fazlasını gösterebilir ve sistem çalışmasının son de-

rece güzel bir dinamik görüntüsünü sağlar. Diğer devreler ve sistemler kendilerine özgü haritalarını çizeceklerdir. Tecrübeli bir operatör haritaya bakarak sistem faaliyetinin doğru yönde olup olmadığını söyleyebilecek ve her durumda buna uygun olarak sarf edilen zamanda bütün noktaların bu orantılı parlaklığını görebilecektir.

Tartışılan üç görüntüleme şeklinde elde edilen şekillerden bir veri domeni cihazı, neticeleri belirtebilir.

0000	BASE PAGE STACK
00FF	
2800	Anahat Programı (MAINLINE)
2BFF	
4004	Giriş / Çıkış
4009	
FFF8	Vektörler
FFFF	

PROGRAM BELLEĞİ



BEKLENİLEN SİSTEM FAALİYETİ

Şekil-4-6 Beklenen sistem çalışması

Şekilde gösterilen biçimde basit bir sayma işlemi ile, bütün ekranlar temelde aynı bilgiyi sağlar. Bununla beraber, daha karmaşık sistemlerde değişik gösteriş şekilleri, biri diğerine tamamlayıcı niteliktedir. "Harita", test altındaki sistemin çalışmalarına çok süratli bir bakış sağlar. Grafik veri akış dizisini belirlerken, listede seçilen faaliyet şeklinin daha ayrıntılı bir çalışmasını gösterir. Bu üç özelliği bulunduran bir cihaz hata arama işleminde kullanılabilirse "harita" önce sistemin çalışmasının tam anlamı ile tetkikini sağlar, sonra da benzer problemlerin çözümünde yardımcı olur. Grafiği kullanarak bu problem olan alanlardaki veri akışı daha yakından kontrol edilebilir. Sonuç olarak daha başka-

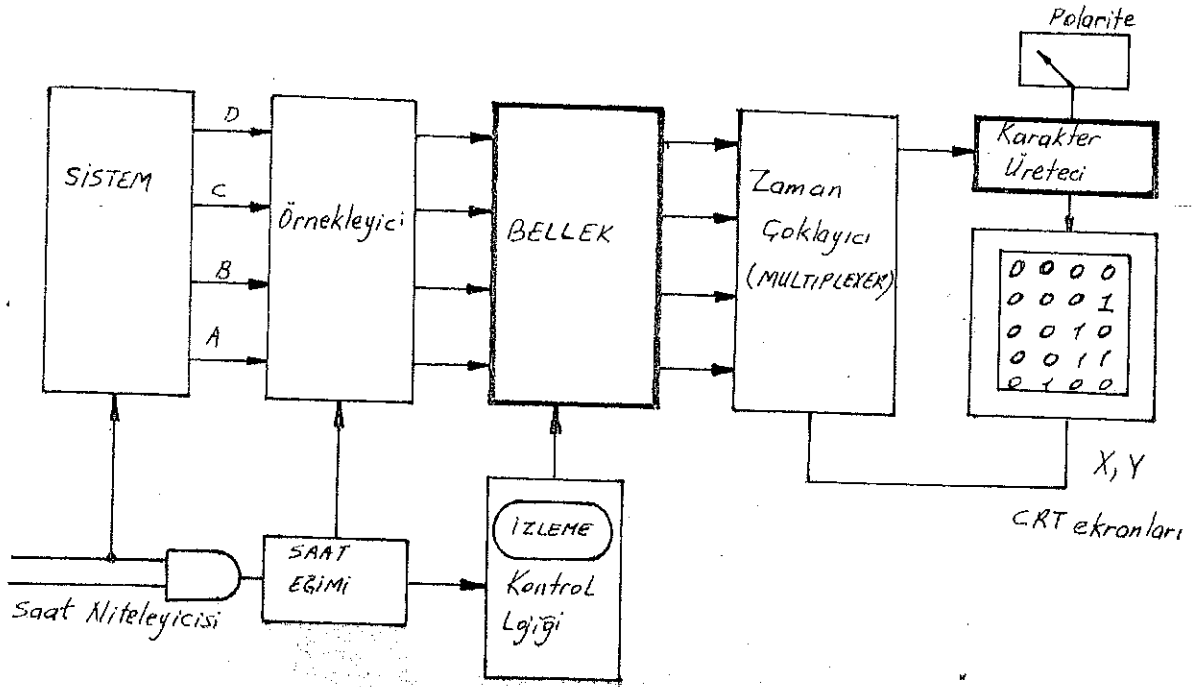
rılı bir kontrol için liste incelenmelidir.

4.4. SAAT KONTROLLARI

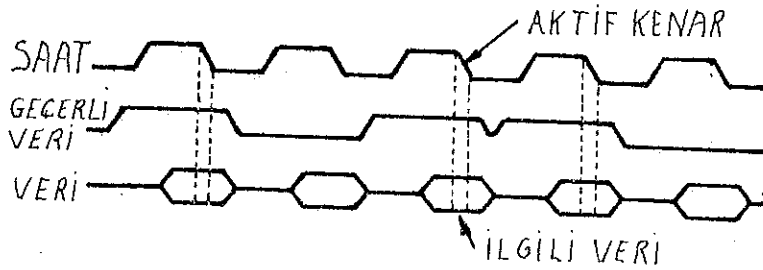
Bir lojik durum analizörünün veri durumu anlatımı ile gözlem altındaki sistemin anlatım durumları aynı şekildedir. Gerilim seviyelerinin gösterilmesi bölüm 4.1 de bahsedilmiştir. Zamanın tanımlanması, uygun saat kenarının seçimi ile elde edilir (şekil-4-7). Sayısal sistemler çoğunlukla senkron olduklarından ve veri durumlarını da saat kenarında oluşturduklarından durum analizörünün veri örneklerini aynı aktif kenarda örnekleme temel ilkedir. Bu şekilde, analizör, sistem tarafından ardışıl çalışmalarda kullanılan veriyi izler. Uygun saat geçişi analizörün ön kontrol tablosundan seçilir.

Pek çok sayısal sistemlerde, birinci saat darbesinde veri ortak yolu, birinci bileşenden veri bulundururken, ikinci saat darbesinde ortak yol, ikinci bileşenden veri bulunduracak şekilde, veri ortak yolu çoklanmıştır (multiplex). Diğer sistemlerde her saat darbesinde ortak yolda geçerli veya yeni veri olmayabilir. Her iki durumda da belleğe, özellikleri belirlenmiş ve seçilmiş veri gönderilmedikçe analizör belleği beklenmeyen örneklerle dolacaktır.

Ortak yol üzerindeki veriler, bir veya daha fazla kontrol hattının durumu ile tanımlanır. Bu kontrol hatları, belleğin içerisinde yalnızca istenen verinin alınması (latch) için, bu kontrol hatları lojik olarak sistemin saati ile "AND" lenmiştir. Bu niteleyici devre, sisteme ve analizöre dışarıdan eklenebilir. Bu durum bazı karmaşık sistemlerde tek çözüm olabilir. Doğal olarak, bir veya daha fazla saat niteleyicileri, lojik analizörün parçasıdır. Niteleyici lojik devre, genellikle aç-kapa (on-off)



Şekil-4-7 Operatör, sistemin saatinin pozitif veya negatif geçişlerinde örnek alabilme seçimini "saat eğimi kontrolü" aracılığıyla sağlar. Verinin, sistem saatinin tam faal kenerında örneklendiğini kanıtlamak için, analizör bir "sıfır yakalama zamanı" (zero hold time) ile tasarlanmıştır. Saat tanımlayıcı kontrolü "lojik 1, lojik 0" veya "göz önüne almama" durumlarına ayarlanabilir(konulabilir). Veri, analizörün belleğine, yalnızca "saat tanıyıcı kontrolü" nun koşulları oluştuğu zaman yerleşir.

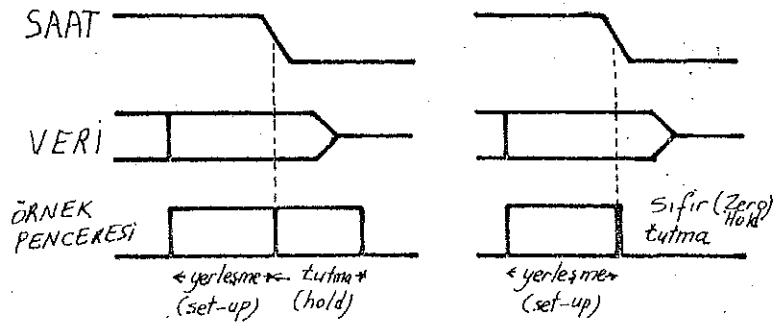


Şekil-4-8 Veri örneği alabilmek için, bir lojik analizörün, veri geçerli olduğu zaman, bir saat geçişine gereksinimi vardır. Veri geçerli

değilse veya her iki saat kenarında istenmiyorsa bu veri, "saat tanıyıcısı" tarafından değerlendirilebilir.

kontrolludur. Aç-kontrol (on-control), ya lojik"1" ya da lojik"0" için geçerlidir. Saat niteleme devresi, bazen ekran tanımlayıcısı olarak bilinir. Bir saatin aktif kenarında bir örnek alınınca, niteleyici koşullar gerçekleştiğinde görüntülenir.

Pratik örnekleme devreleri veriyi sonlu bir zaman penceresi içinde toplar. Olağan bir durum şekil-4-9 da gösterilmiştir. Örnekleme, aktif saat darbesinden biraz önce başlar, biraz sonra biter. Bu zamanlar, sırasıyla yerleşme (set-up) ve tutma (hold) zamanlarına uygundur ve veri bu zamanlar süresi boyunca kararlı olmalıdır. Pek çok sistemlerde veri, aktif saat kenarından hemen sonra durum değiştirir. Bu koşullarda pozitif tutma zamanlı lojik durum analizörleri, belirtilmemiş verileri izleyebilir. Yalnızca akan bilgilerin yakalanmasını sağlamak için, bir lojik durum analizörü, sıfır tutma zamanına sahip olmalıdır.



Şekil-4-9 Veriyi yakalamak için alışlagelmiş bir örnekleme devresi saat darbesinin faal kenarından hemen önce bir zaman süresince, girişin kararlı kalmasına ve bu kenarı takip eden zamanda bir süre "tutma"sına gereksinim duyacaktır. Pek çok sayısal sistemde faal saatin bir kaç nano saniyesi içinde veri değişir. Lojik durum analizörünün, yalnızca o andaki veriyi izlemesini sağlamak için, lojik analizör bir "sıfır tutma zamanına" sahip olmalıdır.

Analizörün şemasında bellek dolunca, bellek saatinin durması sağlanmışır. Bir izlemenin başlatılması için operatör, izle (trace) düğmesine basmalıdır, böylece belleği yeterli örneklerle doldurmuş olur. Özellikle "harita" şeklini elde etmek için yararlı olan sürekli görüntü mod'unda kontrol lojiği sürekli izleme ve durdurma komutlarını sağlayabilmelidir.

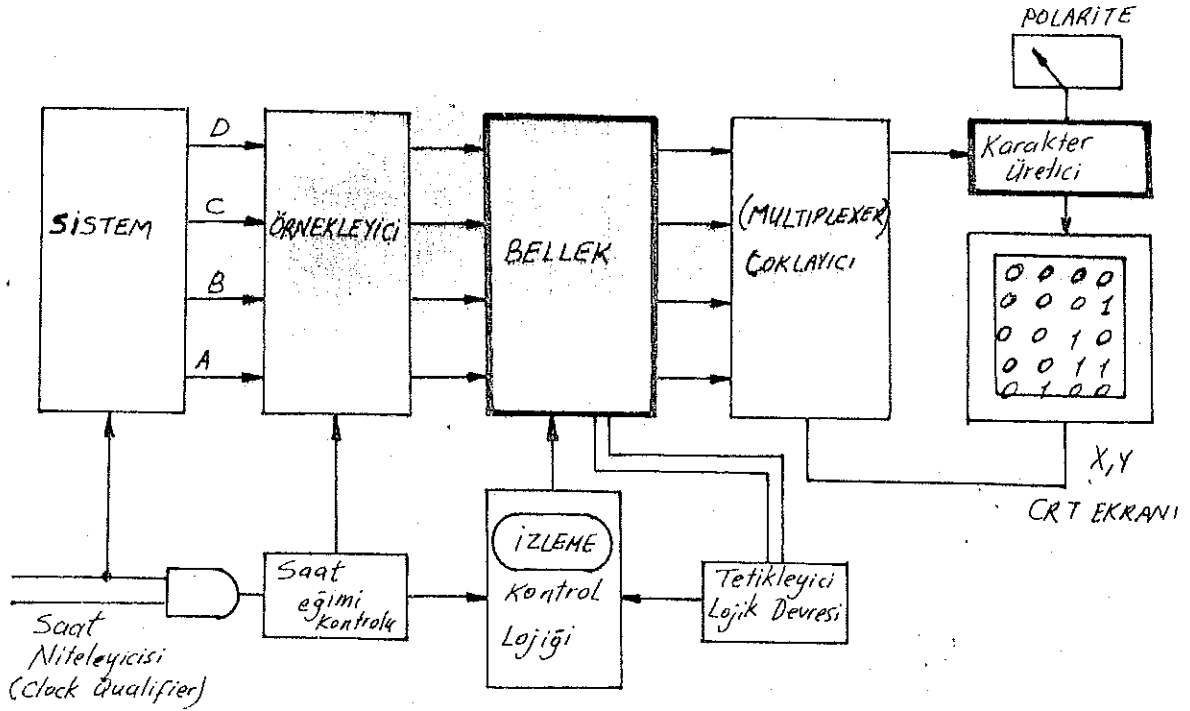
4.5. TETİKLEME

4.5.1 Temel Tetikleme

Şekil-4-7 de gösterilen devre enerjilendiğinde ilk veri grubunun yakalanacağı görülebilir. Eğer veri gelişigüzel olsaydı veya operatör verinin ne olabileceğini bilmeseydi, bu durum kabul edilebilir bir teknik olurdu. Bununla beraber, pratikte analizör bilinen bir veri kelimesini izleyen özel bir veri dizisini araştırmak için sık sık kullanılırdı. Örneğin izleme için, tek bir bit paterni veya tetikleme kelimesi referans alınabilirdi.

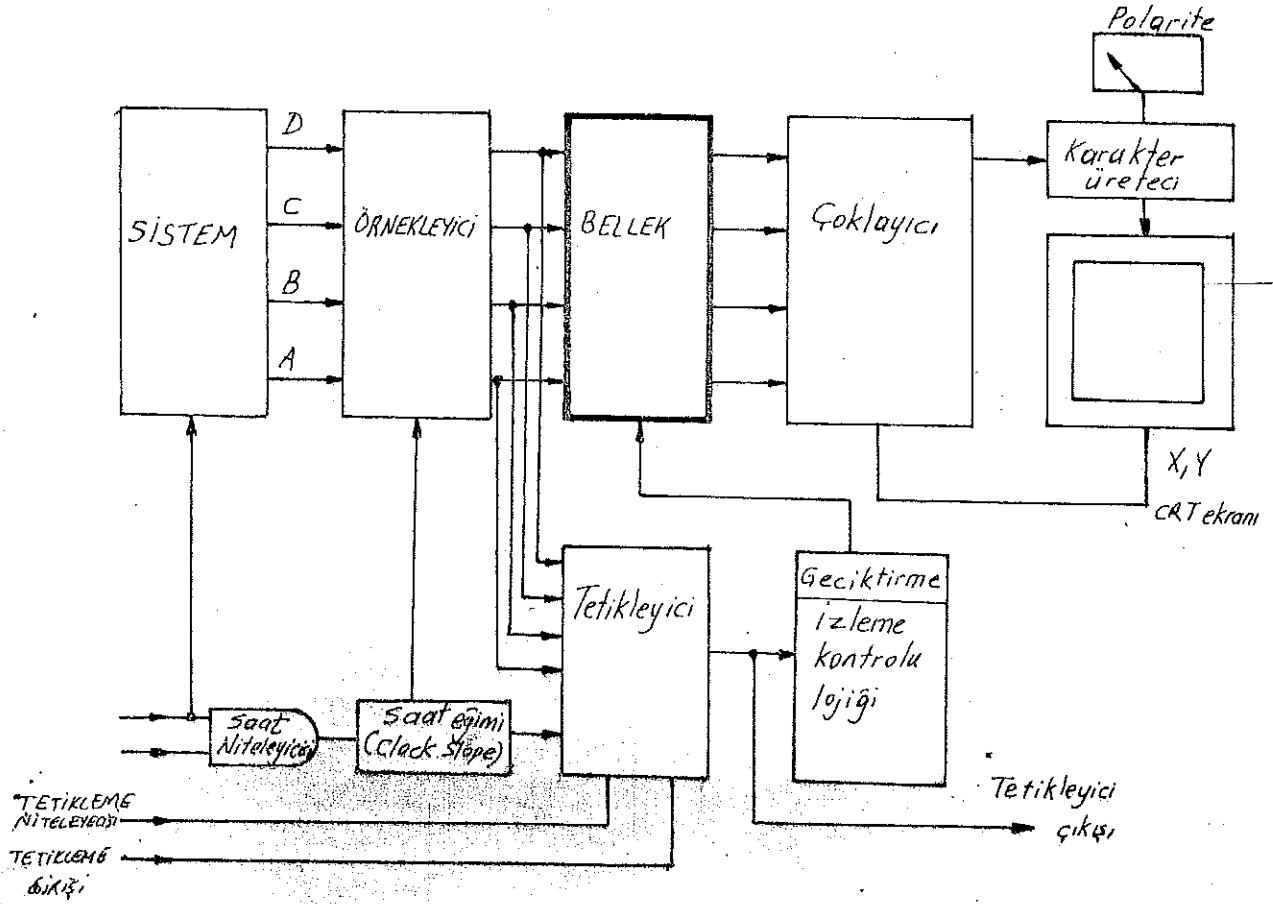
Bilinen bir tetikleme kelimesinin görüntüsünü referans kabul etme tekniği, bu kelimeyi bir tetikleyici lojik devresinde göstermek ve bu paterni analizör belleğindeki son kelimeye kadar karşılaştırmaktır. Patern örnekle uyuyorsa tetikleyici lojik devresi izlemeyi durdurarak tetikleme kelimesini ve bu tetikleme kelimesini izleyen veri sırasını bellekte tutmak için bir kontrol işareti sağlar. Bu sıra CRT ekranında görüntülenmiştir. Tetikleyicinin izleme mod'unu başlattığı nokta mod'un referans noktasıdır.

Şekil-4-10 da kabul edilen tetikleyici lojik devresi, bir gerçek zamanlı senkron işareti sağlamayacaktır. Böyle bir işaret, bir osiloskop gibi zaman domenindeki bir veri kanalının görüntülenmesinde veya ardışıl



Şekil-4-10 Son bellek yerindeki kelimeyi bir "preset" tetikleme kelimesiyle karşılaştırarak, izleme bir "preset" tetikleme kelimesinde durdurulabilir. Artık bellek, tetikleme kelimesini ve tetiklemeden sonraki verileri bulundurur. Bununla beraber zaman domeninde çalışan cihazları senkronize etmek için gerçek zamanlama işareti elde edilemez.

tipte tetiklemeyi sağlayan ikinci bir analizörü çalıştırmak için kullanılabilir. Gerçek zamanlı bir senkronizasyon işareti üretmek için, tetikleme koşulları oluşur oluşmaz tespit edilebilmelidir. Gerçek zamanlı bu senkronizasyon işaretini üretmek için de tetikleyici lojik devresi, bellek girişine yerleştirilmiştir. (Şekil-4-11). Hemen bunun ardından, tetikleyici lojik devresi, dış cihazlar için senkronlayıcı işareti sağlayan bir darbe gönderir. Aynı darbe bellekteki örnek sayısını izleyen içteki bir sayıcı geciktirme devresini çalıştırır ve bellek dolunca girişi keser (disable).



Şekil-4-11 Tetikleyici lojik devreye girişi, hemen örnekleme devresinden sonra koyarak, osilaskopları senkronize edici bir çıkış işareti üretilebilir. "Tetikleyicinin başlattığı izleme mod'unda" belleği dondurmak için bir geciktirme devresi bellek içinden geçen kelimeleri sayar ve bellek dolunca girişleri keser. "Tetikleyicinin sonlandırdığı izleme mod'unu" sağlamak için geciktirme sıfıra gelmelidir. Veriler tetikleme paterni tanımlanana kadar bellek içine alınacaklardır ve bu anda izleme duracaktır. Bellekte tetikleme kelimesi öncesi donmalıdır. Ek bir gecikme ilave ederek tetikleme kelimesinden oldukça uzaktaki verinin yakalanmasını sağlar. Tetikleme devresinin ilave görevleri arasında olan tetikleme niteleyicisi ve tetikleme girişi veya ardışıl tetikleme ileride anlatılmıştır.

Tetikleyici lojik devresi bellekle paralel yerleştirilerek çalışmasına

devam eder ve bellek dolarsa ve izleme tamamlansa bile bundan sonra dahi dış bir tetikleme darbesini sağlar. Ekran üzerinde, tetikleme kelimesi parlaklığı arttırmak suretiyle ve siyah beyaz zıtlaşmasından yararlanılarak belirgin bir şekle getirilir. Her tetikleme biti için lojik 1, lojik 0 veya "Don't care" (gözönüne almama) tetiklemesine izin vermek için üç kontrol durumu vardır. Tetikleme kelimesinin bazı bitleri bilinmiyor veya kullanıcıyı ilgilendirmiyorsa gözönüne almama (don't care) tetikleme durumu uygundur. Analizör bundan sonra pek çok durumlarda tetikleme yapabilecektir. Örneğin tetikleme kelimesinin bir kısmı OXX0 ise X bir "gözönüne almama" durumudur. Bir analizör, hangisi daha önce belirir ise 0,2,4,6 ondalık tabanındaki eşitlikleri, herhangi bir ikili durumlarında tetikleme sağlayacaktır.

"Gözönüne almama" tetikleme durumu hata bulmada faydalıdır. "Harita"ya bakarak problemlerin olduğu yerler bulunabilir. Bu bölgenin daha ayrıntılı incelenmesi, "harita"nın bu bölgesini tanımlayan değerler ile tetiklenerek sağlanabilir. Eğer bu teknik ile başlangıçta analizör tetiklenemez ise tetikleme "gözönüne almama" durumunun yerine diğer bir bit korularak dinlenme (relaxe) durumuna alınabilir. Bununla birlikte eğer izlenen veri ilgi çekici değilse, araştırma daha çok tetikleyici bitleri lojik 1 ve lojik 0 olarak belirginleştirmek suretiyle daha ilgi çekici bir bölgeye yönlenebilir. "Gözönüne almama" tetikleme durumları bazı uzlaşmalara gerek duyabilir. Örneğin tetikleme kelimesinin 16 lı tabanda 1234 veya 1235 formunda olduğunu kabul edelim. İki tabanında gösteriminde, tetikleme herhangi bir problem yaratmayacaktır. Böylece 0001 0010 0011 010X olacaktır ki burada X bir "gözönüne almama" durumudur. Fakat 16 tabanındaki

(hexadecimal) gösterimde $123X_{16}$ tetikleme kelimesi yalnızca 1234_{16} ve 1235_{16} değil, 1230_{16} ve $123F_{16}$ arasındaki bütün kelimeleri tetikleme kelimesi olarak kullanır.

Bölge tetiklemesi yapan bir lojik analizör (daha büyük ve daha küçük kelimelerin tetiklemesi) bu tür boşuna çabayı önler, fakat bu kolaylığı eklemekle, analizör tasarımında daha fazla karışıklıklara neden olur. Buna karşın bazı cihazlar operatörün, gösterme parametrelerini, değişik tabanlarda tanımlanmasına olanak sağlar. Buraya kadar anlatılanlarda tetikleme kelimeleri ikili tabanda tanımlanmış ve sonuçlar 16 lı tabanda olarak görüntülenmiştir. Tetikleme 1234_{16} veya 1235_{16} durumlarında oluşur. Fakat "hex" görüntüsü 123^* olabilir (*yani tetikleme koşulları "hexadecimal" tanımlanmakta fakat "hexadecimal" olarak yazılamamaktadır.) Şekil-4-11 deki cihaz tetikleme kelimesini takip eden verileri izler. Hata aramasında, gözlem altındaki sistemin belirli bir veri durumunun gitmemesi gereken yere gittiği öğrenilebilir. Burada problem verinin oraya gidiş nedenidir. Cihaz verilerin bellek içerisinden akmasına izin vererek ve tetikleme koşulları olduğu zaman, veri akışını durdurarak izleme mod'unu sonuçlandıran bir tetikleme işareti sağlar.

Şimdi bellek, tetikleme kelimesiyle birlikte analizörün negatif zaman verisine yakalaması gibi bu tetikleme kelimesi öncesi veriyi de birlikte bulundurur. Sistemin bozulması tetikleyicinin "izlemeyi durdurduğu mod'da" yakalanabilir. Sistemin saatini kaybettiği durumu düşünelim. "Tetikleyicinin izlemeyi durdurma mod'unda" analizörün izlemesi, saatin bozulmasından hemen önce olacaktır. Saat bozulunca, daha başka veriler izlenmeyecektir. Bununla beraber normal program belleğinin dışındaki yerler tetikleme keli-

mesi ile birlikte oluşabilir. Sistemin, bu bellek bölgesine her girişinde analizör durur ve istenmeyen bölgelere gidişine neden olan sıra daha sonraki inceleme için bellekte tutulur.

4.5.2 Geciktirilmiş veya tetikleme kelimesini takip eden ve izlemenin başlamasından önce örnek veya saat darbelerini "gecikme" kadar ileri öteleyerek yapılan tetikleme şekli (delayed or indexed triggering) :

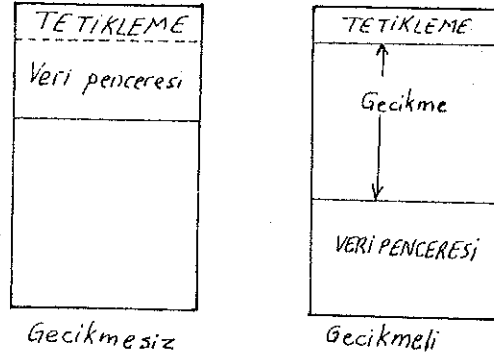
Oldukça geniş bir programın izlenmesi için, eğer bu izleme analizörün bellek derinliğinin pratik sınırları içinde yapılacaksa, daha ayrıntılı incelemeye gerek vardır. İlk teknik yöntem, bir saat niteleyicisi kullanarak daha az ayrıntılı olarak veriyi yakalamaktır. İstenilen izleme (selective tracing) yalnızca özellikleri belirtilmiş verilerin seçimi için kullanılabılır (bölüm 4.6). Diğer bir teknik, her geçişin son veri kelimesini bir sonraki geçiş için tetikleme kelimesi olarak kullanmak suretiyle, birden fazla geçişler yapmaktır. Sonuçta bu teknik aracılığıyla tüm ardışıl dizi meydana getirilebilir.

Program çalışmasını izlemek için birden fazla izlemeler kullanarak sazi sınırlamalar oluşturulur. Programın kapalı çevrim bulundurması durumunda bu çevrim içerisindeki hiç bir adres tek olmayacaktır. Bununla, bir sonraki geçiş içinde tek bir tetikleme işareti sağlayamayacaktır. Tek bir tetikleme kelimesinden oldukça ayrı program çalışmasına ulaşmak için kapalı çevrimleri atlamak veya geçmek için geciktirilmiş veya ötelenmiş (delayed or indexed) tetikleme şekli geçerli bir methodur. İkinci bir teknik ise ardışıl tetiklemedir.

Şekil-4-12 de görüldüğü gibi normal tetikleyici işaretin izlemeyi başlattığı mod'da tetiklemeyi hemen takip eden veri kelimeleri yakalanır.

Gecikme ile analizörün veri penceresi, izlenen tetikleme kelimesinden oldukça ayrı olacak şekilde indekslenir. Şekil-4-13 te sistemin yalnızca sol yoldan geçişinden sonra izleme (trace) yapılmalıdır. Bu durum, 2800₁₆ kelimesinde tetikleme yapılarak ve soldaki kapalı çevrimden çıkmak için gereken saat çevrimi sayısınca geciktirilerek elde edilir.

Çoğu analizörler (64 K dan 100 K ya kadar) nispeten geniş bir geciktirme yeteneğine sahiptirler. Sistem çalışmasını, tetikleme kelimesinden oldukça ayrı olarak örnekler.



Şekil-4-12 Gecikme olmadığı durumda analizör, tetikleme kelimesinden hemen sonraki verileri yakalar. Gecikme olduğu zaman, şekilde olduğu gibi veri penceresini tetiklemeye göre bir miktar ileriye atma sağlanmıştır.

Pratikte sayısal gecikmeleri kullanmak için sınırlamalar mevcut olabilir. Pek az programlar şekil-4-13 teki kadar basit kapalı çevrimler buldururlar ve bu kapalı çevrimlerinin içerisinden geçiş sayıları bazı ek koşullara bağlı olarak değişir. Budurumlarda izleme sırası 2800₁₆ tetikleme kelimesiyle ve verilen sabit bir sayısal gecikme ile tanımlanamaz. Ardışıl bir tetiklemeye gerek vardır. Tetikleyicinin izlemeyi durdurma mod'unda çalışan bir analizör kullanılması tetikleme kelimesi bir küçük geciktirme kullanarak görüntünün ortasına yerleştirilebilir.

4.5.3. Analizörün tetikleme öncesi koşulu ve ardışıl tetikleme (arming and sequential triggering).

Bir lojik analizör için önemli bir istek tetiklemenin bir kelimeler dizisine göre yapılma yeteneğidir. Örneğin şekil-4-13 ün programında sol taraftaki kapalı çevrimde yürütmelerin sayısı, yalnız sol taraftaki yoldan sonra izleme dizisinin yakalanmasına izin vermek için yeterince tanımlanmamışsa, analizör içindeki özel lojik devreleri (tetikleme) 2800_{16} durumu tetikleme öncesi koşulu olarak konulabilir. Sonra analizör 2815_{16} durumunda tetiklenmek için düzenlenebilir. Bu mod'da eğer sağ taraftaki yol izlenirse analizör 2815_{16} da tetikleme kabul etmez. Yani tetikleme öncesi veya çalıştırma koşulları 2815_{16} tetikleme kelimesinden önce kullanılmalıdır.

Bazı analizörler tetikleme öncesi veya çalıştırma mod'unu bulunduran tetikleme lojik devresiyle birlikte tasarımı yapılmışlardır. Çalıştırma koşulları tetikleme kelimesinden ve izlemenin başlamasından önce meydana getirilmelidir ya da tetikleme öncesi koşulu iki analizör kullanarak elde edilir. İlk tetikleme kelimesi olan patern uygulayıcı olan bitin alınması ile ilk analizör, kendisine ait patern tetikleme çıkış hattına (PTO: Patern Trigger Output) bir işaret çıkışı sağlar. PTO hattı ikinci analizörün "TRIG ARM" girişine bağlanmıştır. Tetikleme öncesi koşulunun oluşturulması işaretinin alınması üzerine ikinci analizör, tetikleme koşulları oluşunca bir izleme yürütür. Tetikleme öncesi koşullandırması yalnız durum analizörlerine has değildir. Çok önemli bir uygulama şekli bir durum analizörünün, bir zamanlama analizörünü veya bir zamanlama analizörünün durum analizörünü tetikleme öncesi koşullandırması için kullanılması

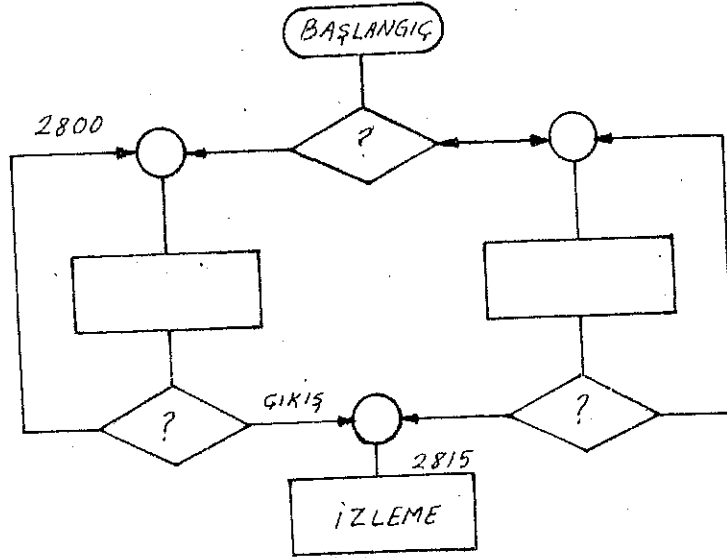
diđi zaman meydana gelir.

Karmaşık bir donanım ile, tetikleme öncesi koşullaması ile elde edilen iki kademeli tetikleme elverişli olmayacaktır. Örneđin şekil-4-14 de açıklanan, yalnızca sistemin iki nolu yolu (alt programı) takip etmesinden sonra bir izlemenin yürütülmesi için diđer tüm yollar seçilmemelidir.

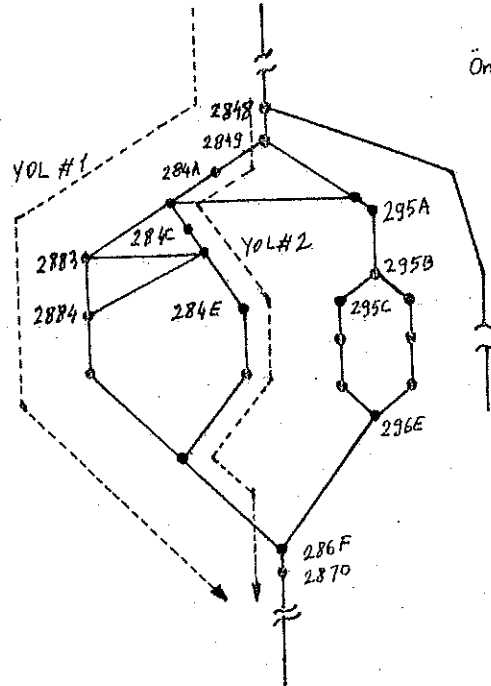
Analizör yalnızca bir dizi üzerinde bulunan 2849₁₆ , 284C₁₆ ve 284E₁₆ durumlarından sonra izlemeye gerek duyar.

Eđer analizör, tetikleme dizisinde verilen adresler ile birleştirilmiş verileri görüntülerse, bu mod daha sonraki çalışmalarda veriyi elde etmek için kullanılır. Örneđin şekil-4-15 teki parametreler 286A₁₆ , 286D₁₆ ve 2877₁₆ adımlarında yüklenmişlerdir. Problem olan program 29A7₁₆ adresinde başlar. Gösterilen tetikleme özelliklerini kullanarak analizör, bütün parametreleri ve izlemeyi listeler. Şekilde gösterilen örnekte bir durum sayıcısı çalıştırılmıştır. Bu, 286A₁₆ adresindeki ilk parametrenin yakalanması ile çođaltma programı başlaması arasındaki 1509 durumu gösterir. Bu parametreleri yakalamak için daha geniş bir tetikleme kapasitesi olmaksızın 1600 durumdan daha fazla geniş bir belleđe gerek vardır. Tetikleme öncesi parametrelerindeki daha fazla geriye gidiş bellek kapasitesindeki artışı gerektirir.

Pek çok karmaşık devrelerde ardışıl tetikleme isteđi birden çok sayıda girişler kullanılarak karşılanabilir. Şekil-4-16 da bu sorun, sıralamanın tekrar başlatılması veya tetikleme kelimesi ile çözümlenebilir. Sıralamanın tekrar başlatılması uygulanmadıđı anda 1 no'lu yol ilk iki geçişi sağlar. Arta kalan tetikleme kelimeleri 3 no'lu yol "alt geçit" üzerinden sağlanabilir. Eđer sıralamanın tekrar başlatılması (sequence



Şekil-4-13 "İZLEME" diye isimlendirilen izleme sırasını, sistemin sol tarafındaki devreyi geçmesinden hemen sonra yakalamak için, tetikleme kelimesi 2800₁₆'ya konulmalı ve geciktirme, 2800₁₆ ve 2815₁₆ durumları arasındaki toplam saat çevrimlerine ayarlanmalıdır.



Önce bulunacak sıra: 2849
 Sonra: 284A
 Sonra: 284C
 Sonra: 284E
 Sonra: 286F
 Başla: 286F

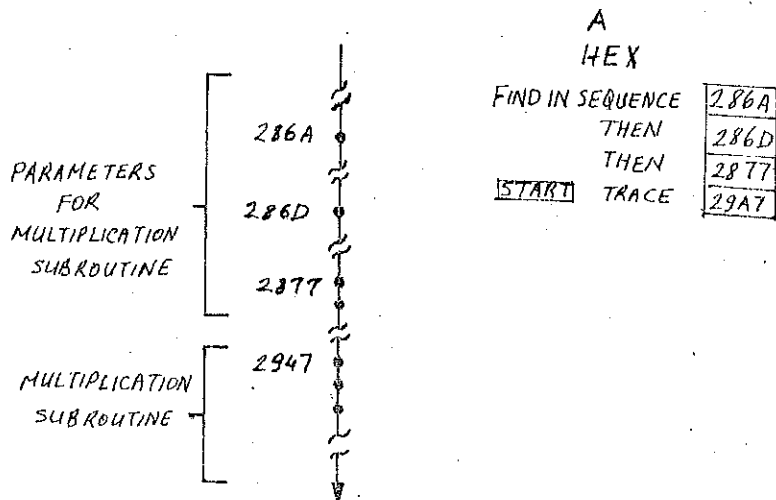
Şekil-4-14 Bir izleme, ardışıl tetikleme suretiyle şekilde gösterildiği dizide, karmaşık yollu bir programın tanımlanan bir yolundan

geçtikten sonra elde edilebilir.

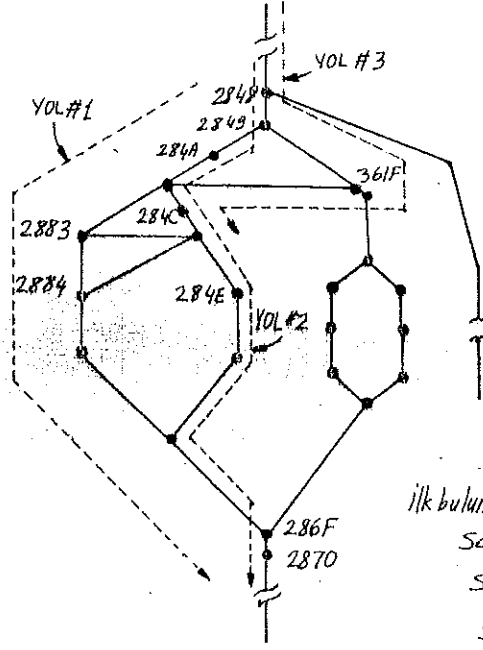
restart) 2870₁₆ durumuna yerleştirilmişse analizör bir izleme olayına bağlamadan önce program içinden bir geçişte tetikleme sırasını bulmalıdır. tetikleme bir defa kesildimi, tetikleme sırası tekrar başlamalıdır. Sıralamanın tekrar başlatılması, sıfır uzunluğunda olan bir yol içinde bir geçişin izlenmesi için kullanılabilir. Burada sırayı tanımlamak için tek noktalar yoktur. (Şekil-4-17) Bu programda 2849₁₆ şartları bir kez oluştu mu, her birbirini izleyen kelimedede analizör:

A. Başlama sırasını kontrol edecektir. Kelime 287C₁₆ ise analizör bir izlemeye başlayacaktır. Eğer 287C₁₆ değilse analizör:

B. Tekrar başlama koşullarını kontrol edecektir. Eğer bu durum olmazsa analizör 2849₁₆ kelimesini tekrar arar. Bu durumda sıralamanın tekrar başlatılması faaliyete geçmez. O halde (A) daki koşullar oluşmadıkça ("sıfır" uzunluktaki yol gibi) analizör daima tekrar başlayacaktır.

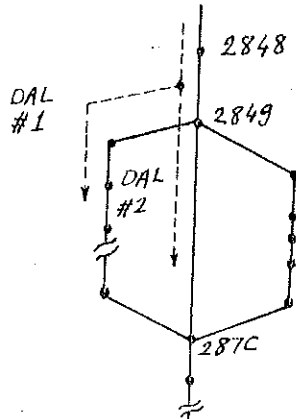


Şekil-4-15 İzleme özellikleri ve sonucu olan sıralama listesi, daha sonraki bir program yolunda kullanılan parametreleri toplamak için bir 1610 A lojik analizörü ile ardışıl tetiklemenin nasıl kullanılabileceğini göstermektedir.



ilk bulunulacak: 2848
 Sonra : 284A
 Sonra : 284C
 Sonra : 284E
 BAŞLA: 286F
 Sıralamanın tekrar
 başlatılması: 2870

Şekil-4-16 Çok kollu bir çevrimde, özel bir yolu izlemek için, izleme özellikleri ve çevrim içinden tek bir geçişten sonra özel yoldan geçildiğini göstermek için bir çıkış durumunu garanti etmek için sıralamanın tekrar başlatılmasının kullanılması.

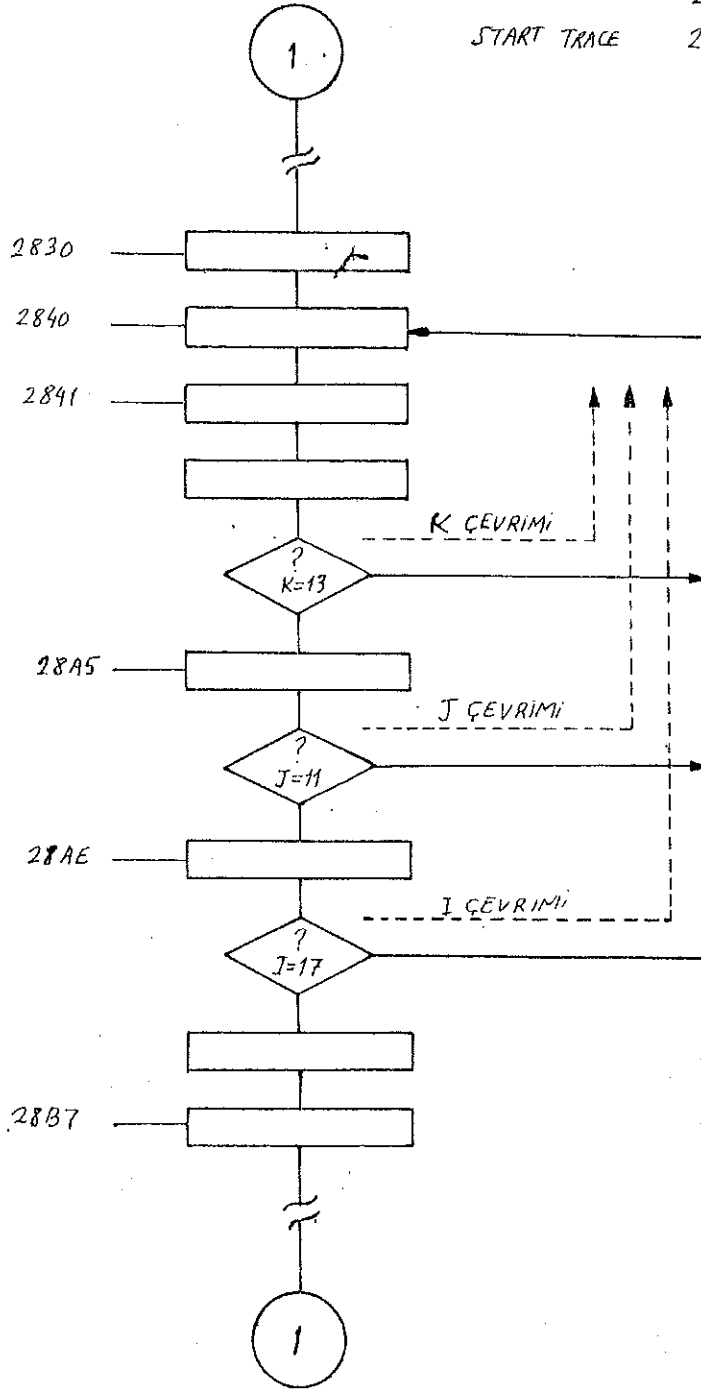


FIND IN SEQUENCE 2849
 START 287C
 SEQ RESTART ON XXXX

DAL #1: Değişken yol uzunluğu
 DAL #2: "Sıfır" yol uzunluğu
 DAL #3: Sabitleştirilmiş yol uzunluğu

Şekil-4-17 Sıralamanın tekrar başlatılmasını kullanarak(direk atlama) bir sıfır uzunluktaki dalı izlemek için izleme özellikleri

	OCUR	
	DEC	
FIND IN SEQUENCE	2830	1
THEN	28AE	9
THEN	28A5	8
START TRACE	2841	7



Şekil-4-18 Şekilde gösterilen izleme özelliklerine sahip bir lojik analizör tanımlanan her kapalı çevrim içinden geçerek veri toplayacaktır.

4.5.4. Sayma Tetiklemesi

Çoğunlukla karşılaşılan program şekil-4-18 de gösterilen bir alt programdır. Eğer bir analizör tetikleyici lojik devresinde bir geçiş

sayıcısına sahipse, program, çalışmasını kontrol etme olanağı vardır. Örneğin "I" kapalı çevriminin 9'uncu geçişi, "J" kapalı çevriminin 8 inci geçişi ve "K" kapalı çevriminin 7 nci geçişi gibi.

Şekil-4-18 de gösterilen izleme tanımlamaları, analizörü, yalnızca bu koşullar oluştuğunda 2841₁₆ durumunda kazanılan verilere yönlendirecektir. 2830₁₆ durumu, analizörün en geniş alt program çalışması sırasında tam ortadan programa girmemesini sağlamak üzere tanımlanmıştır. Tetiklemede, sayma fonksiyonunun olmaması durumunda, yaklaşık 28AE₁₆ kelimesine kadar tanımlanmış olan geçişteki veriyi bulundurmak için çok geniş bir belleğe ihtiyaç vardır. İstenilen veri için lüzumlu olan bu belleği araştırmak için sonsuz bir operatör sabrına gerek vardır.

Sayısal gecikme, bütün "gözönüne almama" durumlarının sırasını belirtmek suretiyle ve bir geçiş sayıcısıyla sağlanabilir. Örneğin 284B₁₆ kelimesinin ilk belirlemesini bulmak için ve izlemeyi XXXX₁₆ kelimesinin 837 nci belirmesinde başlatmak için bir analizörü düzenleme işlemi, 284B₁₆ tetikleme kelimesine ve 837₁₀ luk bir gecikmeye eşdeğerdedir.

4.5.5. Tetikleme Koşulları Niteleyicileri (Tetikleme koşullarının istenilen zamanda oluşumunu sağlayan dış işaret) (Trigger Qualifiers)

Çoğu analizörler, saat veya ekran için kullanılanlara benzer tetikleme düzenleyicilerine sahiptirler. Bir tetikleme niteleyicisine sahip olan analizör, yalnızca hem tetikleme hem de düzenleme koşulları bir arada oluştukları zaman bellekteki verileri donduracak veya tetikleyecektir. Tetikleme niteleyicileri tetikleme kelimesinin uzunluğunu, her niteleyici girişi için ek bir kanal aracılığıyla genişletir. Tetiklemede esnekliğin artmasıyla birlikte veri toplamadaki esneklik de artar. Tetikleneyi nite-

Boyan işaretler görüntülenmezler ve bellekte bir yere ihtiyaçları yoktur. Tetikleme tanımlanarak, görüntüyü değiştirmez. Bununla beraber, tetikleme, saat veya görüntü niteleyicileri tarafından düzenlenecektir.

Tetikleme niteleyicisi, mikrobilgisayar sistemlerini gözlemede yararlıdır. Örneğin, eğer iki alt program arka arkaya saklanmışsa, birinci alt programın alt program dönüşünün (RTS) yürütülmesi sırasında, modern bazı mikrobilgisayarlar ortak yol üzerine SUB 2 adresine koyacaktır. SUB 2 adresini tetikleyici olarak kullanan bir lojik durum analizörü, hem birinci alt programın tamamlanmasında ve hem de ikinci alt programın başlangıcında da bir izleme yürütecektir.

Bununla beraber, farklılıklar vardır. İki no'lu alt programın yürütülmesi sırasında adres ortak yolu üzerinde SUB 2 adresi olduğu zaman, bu adres bir işlem kodunun alma işlemidir. (Op. code fetch).

Bir çok mikrobilgisayarda yukarıdaki olayın özelliklerini tanımlayan bir kontrol hattına sahiptirler. Örneğin, 6502 mikrobilgisayarında SYNC hattı yükselecektir (lojik 1). Bu kontrol hattını tetikleme niteleyicisi olarak kullanırsak, bu hat analizörün yalnız ikinci alt programda bir izleme yapmasını sağlar. SUB 2 adresi, bir önceki RTS komutunun sonucu olarak meydana gelir, SYNC hattı lojik 0 durumuna gelir. Bu sebepten tetikleme niteleyen koşullar oluşmaz ve izleme görülmez.

SONUÇ:

Basit bir sistem, tek bir tetikleme kelimesi ile tetiklenerek izlenebilir. Bir tetikleme kelimesi izlemeyi başlatabilir veya sonuçlandırır.

İçerik işaretler görüntülenmezler ve bellekte bir yere ihtiyaçları yoktur. Tetikleme tanımlanarak, görüntüyü değiştirmez. Bununla beraber, tetikleme, saat veya görüntü nitelendiricileri tarafından düzenlenecektir.

Tetikleme nitelendiricisi, mikrobilgisayar sistemlerini gözlemede yararlıdır. Örneğin, eğer iki alt program arka arkaya saklanmışsa, birinci alt programın alt program dönüşünün (RTS) yürütülmesi sırasında, modern bazı mikrobilgisayarlar ortak yol üzerine SUB 2 adresine koyacaktır. SUB 2 adresini tetikleyici olarak kullanan bir lojik durum analizörü, hem birinci alt programın tamamlanmasında ve hem de ikinci alt programın başlangıcında bir izleme yürütecektir.

Bununla beraber, farklılıklar vardır. İki no'lu alt programın yürütülmesi sırasında adres ortak yolu üzerinde SUB 2 adresi olduğu zaman, bu adres bir işlem kodunun alma işlemidir. (Op. code fetch).

Bir çok mikrobilgisayarda yukarıdaki olayın özelliklerini tanımlayan bir kontrol hattına sahiptirler. Örneğin, 6502 mikrobilgisayarında SYNC hattı yükselecektir (lojik 1). Bu kontrol hattını tetikleme nitelendiricisi olarak kullanırsak, bu hat analizörün yalnız ikinci alt programda bir izleme yapmasını sağlar. SUB 2 adresi, bir önceki RTS komutunun sonucu olarak meydana gelir, SYNC hattı lojik 0 durumuna gelir. Bu sebepten tetikleme nitelendirici koşullar oluşmaz ve izleme görülmez.

SONUÇ:

Basit bir sistem, tek bir tetikleme kelimesi ile tetiklenerek izlenebilir. Bir tetikleme kelimesi izlemeyi başlatabilir veya sonuçlandırabilir.

bilir. Bir tetikleme kelimesi normal olarak, bir ekran üzerinde parlaklığın artırılması veya ters renklendirme suretiyle belirginleştirilir. Eğer tetikleme kelimesi üzerinde bazı sınırlandırılmalar isteniyorsa, tetikleme nitelendiricileri kullanarak daha ileri derecede sınırlandırmalar sağlanabilir. Tek bir tetikleme kelimesinden oldukça uzak bir program çalışmasına ulaşmak için geciktirme veya atlama (indexed) türü tetikleme çeşitleri kullanılır. Daha karmaşık program çalışmalarını izlemek için tetikleme öncesi koşullandırılmış analizör ardışıl tetikleme için, verilen bir durum oluşunca tetikleme sırasını tekrar başlatma, tetikleme tam anlamı ile oluşturmak için analizörün belirtilen sayıda tetikleme durumunu bulan tetikleme türlerine gerek vardır. Tetikleme yeteneği bir lojik analizörün en önemli özelliklerinden biridir. Yeterli tetikleme olmaksızın sistemin çalışmasının istenilen bölümünün gözlenmesi çok zordur.

4.6 İZLEMENİN SEÇİMİ

Sayısal sistemleri izlerken, bir lojik analizör çok sayıda veriyi yakalayacaktır. Operatör bu veriler arasında seçim yaparak düzenlemelidir. En iyi seçim şekli bu işlemi cihazın içinde yapmaktır. Buna "izlemenin seçimi" denir. Analizör içindeki lojik devreler yalnızca istenilen verileri seçer ve belleğe bu verileri saklar. Bellek, çok sayıda istenmeyen verileri saklamayıp daha az sayıda fakat tanımlanmış olan verileri saklar.

İzleme seçiminde en geniş olarak kullanılan ölçme türü, yalnız tetikleme kelimelerini izlemektir. Genelde tam anlamıyla tanımlanmış bir tetikleme kelimesi kullanarak bir programı incelemek, yetersiz sayıda örnek alınmasına neden olur. Yeterli sayıda durumların elde edilmedi, tetikleme kelimelerine bazı "gözönüne almama" komutlarını eklemekle sağlanabilir.

Örneğin, bir tetikleme kelimesi $12X4_{16}$ olarak tanımlanmış ise, yalnızca 1204_{16} , 1214 , 1224 , 1234 $12F4_{16}$ izlenecektir. Program ardışıl ise ve bütün durumlar izlenmiş ise görüntü 1200 , 1201 , 1202 den $12FF_{16}$ ya kadar 256 kelime araştırılmış ve çözümlenmiş olur ve 16 kelimelik izleme tetikleme kullanarak elde edilen sonuca eşdeğerde bir fonksiyonel bir neticeye ulaşılmış olur.

İzleme tetikleme kelimeleri ile sayısal geciktirmeyi birleştirerek bir "izleme olayları mod'u" oluşturulur. İzlenen olaylar, tetikleme kelimesini izleyen her "n" inci bitte olacaktır. Burada "n", geciktirme kontrolü tarafından belirlenecektir. Test altındaki sistem çok sayıda dallara ayrılacağından "tetikleme kelimesinin" ve "geciktirme"nin uygun olarak seçimi, program çalışmasının tam bir görünümünü verecektir.

Daha ayrıntılı lojik analizörler, yalnızca tanımlanan durumları izleyebilme yeteneğine sahiptirler. Bu analizörler, her "seçilmiş izleme" (trace-only) durumunda, çok sayıda oluşumları sayabilme yeteneğine sahiptirler. Böylece, tanımlanan durumlar sayıcılar tarafından uygulanan sayıda oluşunca, belleğe alınırlar. Bu tür izleme seçimi bir yazılım uygulaması olarak düşünülebilir. Veriler analizöre girerler ve "seçilmiş izleme" koşullarıyla karşılaştırılırlar. Koşullarla uygun bulunurlarsa, veriler belleğe girebilirler, koşullarla uygun bulunmazlarsa bu veriler gözönüne alınmazlar.

Saat darbesi niteliyicileri bir donanım uygulayıcısı olmasına karşın, "izleme seçimi" sağlayabilirler. Yalnızca tanımlanması yapılmış girişlere ayrılmış özel hatların durumlarına bağlı olarak veriler elde edilebilir.

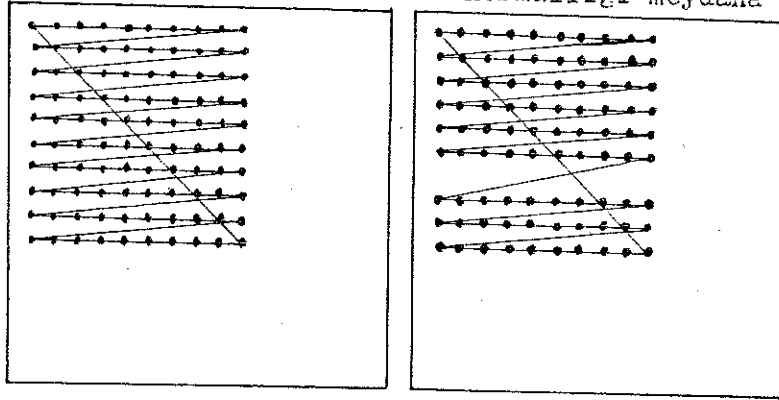
Bölüm 4.4 de verilen örnekte bir mikrobilgisayar kontrol hattı, bir nitelendirici olarak kullanılmaktadır. Eğer nitelendiriciler, mikrobilgisayarın bazı adres hatlarına bağlanmış ise, yalnızca seçilmiş uzunluktaki adres, analizör tarafından izlenebilir. Bu durumda "saat nitelendiricisi" (clock qualifier) ile "izlemenin seçimi" (selective trace) mod'u arasında hiç bir fark olmaz.

4.7. BİRBİRİNİ ETKİLEYEN OSİLOSKOP/ANALİZÖR ÖLÇMELERİ

Çoğu sayısal sistemlerde, durum akışının bir analizi, bir problem bölümünün ortaya çıkarılmasını sağlar. Eğer bu durum problemleri, zamana bağlı olan, gerilim düzeyi, darbe (pulse) genişliği, yükselme zamanı, sığırma veya bir veya birkaç işaret devresindeki gürültü gibi olaylardan meydana geldiği şüphesi oluşursa, bu devre ya bir osiloskop veya bir zaman (timing) analizörü kullanarak daha ayrıntılı olarak incelenebilir. Eğer işlem tekrar edilebilse ve her işaret hattı ayrı ayrı, birbirinden bağımsız olarak incelenebilse osiloskop en yararlı bir cihaz olur. Bununla beraber, işaretler arasındaki zamanlama önemli ise veya negatif zamana ihtiyaç varsa yahut da dar sığramalar (glitch) gibi hızlı gelişen olaylar araştırılacak ise "zaman analizörü" kullanılmalıdır. Durum akışı problemi olan yerlere, her iki cihazı senkronize etmek için, durum analizöründen alınan gerçek zamanlı (real-time) tetikleme çıkışı işareti kullanılır. Arıza aramasında, bir lojik analizörün osiloskop'u tetiklemede nasıl kullanılabileceğine bir örnek olarak, iki "onlu" BCD sayıcısının doğru olarak çalışmadığını düşünelim. Problemi sınırlandırmak için ilk adım, sistemin çalışmasının genel bir görüntüsünü elde etmektir. "Harita" mod'un da lojik durum analizörü, atlanılan durumları vurgulayarak belirtir.

Halbuki "grafik" mod'unda hatalı olan sayma dizileri belirtilir. Sayıcı doğru olarak çalışıyorsa, şekil-4-20(a) elde edilir. Sol-üst nokta "00" sayısını gösterir, sağ-alt nokta ise BCD'nin "99" sayısını gösterir. Noktalar arasındaki çizgiler de sıralamayı gösterir. Şekil-4-20(b) hatalı saymayı göstermektedir. Sayıcı "59" (0101 1001) durumundan "70" durumuna "60" dan "69" a kadar olanlardan geçmeksizin atlamıştır.

Hata "59" durumundadır ve bu noktada tetikleme kelimesi üretilir. BCD sayıcısının her "59" a gelişinde lojik analizör, osiloskop'u tetiklemek için bir çıkış tetikleme darbesi gönderir. Osiloskop probe'ları bütün sayıcı hatlarından, sıra ile geçirilir. Sayıcının veri veya kontrol hatlarından birinde osiloskop ile yapılacak mikroskobik bir inceleme, sayıcının kötü çalışmasına neden olan anormalliği meydana çıkarır.



(A) Şekil-4-20(a) İki adet onlu BCD sayıcısıyla elde edilmesi gereken "harita"da, ondalık olarak "0"dan "99" a kadar olan yerlere karşın her mevkie bir nokta görülmektedir. (b) Hatalı sayıcıda bu durumların bazıları atlanmıştır.

Lojik analizör problemin yerini vurgular. Sonra, analizörün patern tanımlaması ve tetikleme yeteneği, osiloskop'un, uygun olan sayısal zaman sınırlarını meydana getirmesini sağlar. Lojik analizör olmaksızın tek çözüm şekli, sayıcının saatini tek tek darbeleri olarak çalıştırmaktır.

Dondurulmuş bir izleme den sonra dahi tetikleme koşulları oluştukça, analizör, bir tetikleme darbesi üretmeve devam eder ve bir standart osiloskop'ta tekrarlanan işaretlerin görüntülenmesine izin verir.

4.8. BİLGİSAYAR SİSTEMLERİNİN GÖZLENMESİ

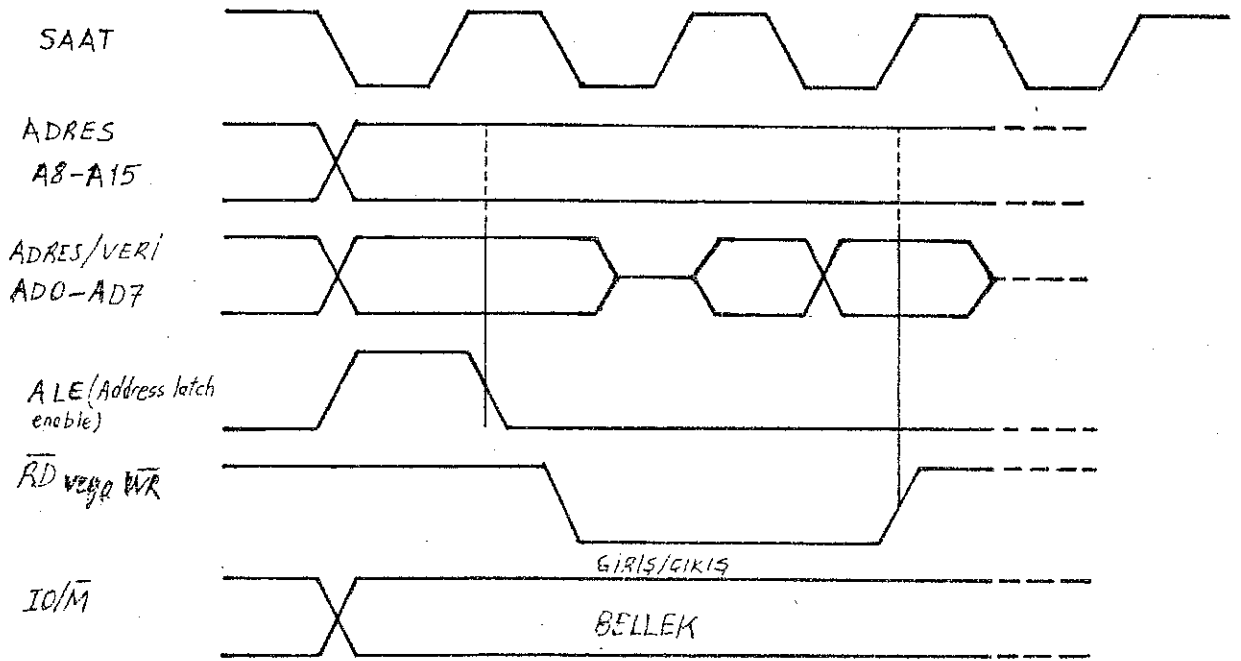
Modern sayısal sistemlerin çoğunda, örneğin bir mikrobilgisayar gibi, herhangi bir tür bilgisayar vardır. Böyle bir sistemin tam anlamıyla gelişmesi için geniş simülasyon, emülasyon ve analiz yeteneğine ihtiyacı vardır. Pek az sayıda gelişmiş sistemler bu problemlerle uğraşırsa da çoğu sistem bunları önemli ölçüde gözönüne almaz. Analiz etme yeteneği tüm sistem çalışmasının yalnızca genel bir görüntüsünü belirlemekten daha öte, program akışının ve verimin ayrıntılı bir incelemesini buldurmamalıdır. Planlama tamamlandıktan ve üretime geçildikten sonra, üretim deneyleri ve kullanıcı bakımı sırasında durum analizörüne ihtiyaç duyulacaktır. Bir mikrobilgisayar kullanılış ömrü boyunca izlenmek için bir lojik durum analizörüne ihtiyaç duyulacağından bu analizörü orijinal tasarıma ilave ederek sistemi üretmek daha verimli olacaktır.

ÖRNEK: INTEL 8085

Ek olarak devre elemanları gerektiren bir mikrobilgisayar Intel 8085 tir. Mikrobilgisayar, bir ortak yoldaki alçak anlamlı adres ve verileri çoklar (multiplex). Adres, ALE (adress latch enable) hattının negatif geçişlerinde yakalanır. Veri ise RD (oku) veya WR (yaz) hattının yükselen kenarında yakalanır. Eğer kesmeler gözleniyorsa, veri "kesme anlaşıldı" (INTA :Interrupt acknowledge) hattının pozitif geçişinde geçerli olacaktır.

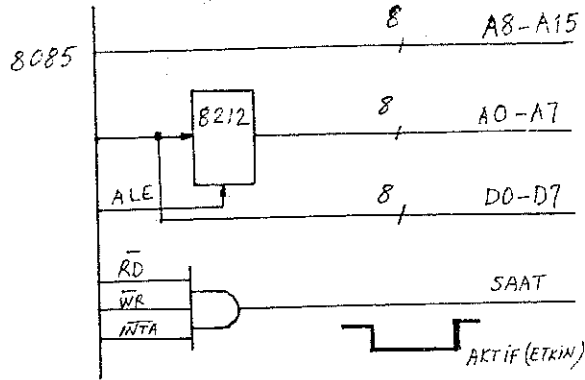
Bir lojik durum analizörünü, bir 8085 mikrobilgisayarına uygulama

devresi şekilde gösterilmiştir. Deney altındaki sistemde, 8085 mikrobilgisayarından başka birimler varsa, Intel 8212 ye 8-bitlik saklama yazıcısının eşdeğeri bir ünite sağlanmalıdır. Bu durumda, ara bağlantı birimi sağlayabilmek için, bir lojik analizöre yalnızca üç girişli bir AND kapısı eklenmelidir.

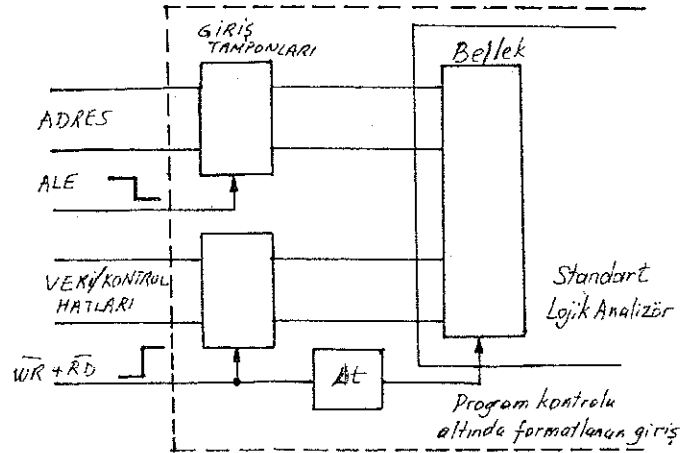


Şekil-4-21 8085 mikrobilgisayarının zamanlaması. Adres ve verinin alçak anlamlı "byte" ları, bir ortak yol üzerinde, zaman-paylaşımlı olarak çoklanmıştır. Adres, ALE hattının negatif geçişi sırasında örneklennelidir. Veri ise ya "okuma" veya "yazma" hatlarından uygun olanının yükselme kenarında örneklenir.

Oku (RD), yaz(WR) veya \overline{INTA} hatlarından uygun olanının, saat darbesi olarak kullanılmasına bağlı olarak, 8085 izlemesi seçilebilir. Diğer bir uygulama şekli de , okuma, yazma veya işlem kodunun alma işleminin nitelendirilmesi için, iki durum hattının kullanılmasıdır. Intel 8085 belleği I/O (giriş/çıkış) birimlerinden ayırır. İki çalışma şekli, $\overline{IO/M}$ ni kullanarak tanımlanabilir.



Şekil-4-22 Bir lojik durum analizörünü bir 8085 mikrobilgisayarına ara bağlantı sağlayan (interface) devre.



Şekil-4-23 Çok girişli (multiple) saat analizörü 8085 mikrobilgisayarını izlemek üzere yazılım olarak düzenlenebilir. Adres verisi ALE nin negatif geçişinde örneklenmiştir. Veri ise ya \overline{WR} veya \overline{RD} kontrol hatlarının pozitif geçişlerinde örneklenmiştir. Bundan sonra adres veya veri geciktirilmiş $\overline{WR} + \overline{RD}$ saat darbelerinin bileşimi ile belleğe gönderilir.

8085 mikrobilgisayarı, ara işlemcili bir devreye olan ihtiyacı ortadan kaldırmak suretiyle bir çok girişli (multiple) saat analizörü kullanılarak izlenebilir. Şekil-4-23 çok girişli saat darbeleri ile izlemeye,

adres verisi zamanlama için programlanarak analizöre gönderilir ve ALE nin negatif geçişlerinde örneklenirken, veri ve durum hatları ya \overline{RD} veya \overline{WR} kontrol hatlarının pozitif geçişlerinde örneklenirler. Veriler beraberindeki adreslerden geri kaldıkları için, adresleri sıraya koymak bakımından, bir saat darbesi "temel" olarak seçilmelidir. Ekranında düzenlemenin sağlanması için, verinin analizör giriş ünitesinden (input buffers) belleğe geçişini sağlamak bakımından \overline{RD} ve \overline{WR} saat darbeleri birleşimi temel kabul edilmelidir.

4.9. SONUÇ

Lojik durum analizörünün bir tartışmasını bulunduran bu bölümde özet olarak şu sonuçlara varılmıştır:

1-) Analizör çok düğümlü olup, mikrobilgisayarlar gibi sistemlerin veri ortak yollarındaki bilgileri izlemeyi sağlar. Veriler, değişik lojik ailelerinin kullanabileceği değişik eşik gerilimine sahip olabilen analizör giriş karşılaştırma (comparator) üniteleri tarafından değerlendirilirler.

2-) Bir tek tetikleme veya patern (görüntü) tanıyıcısıyla birlikte sayısal gecikme, problemlı program kesiminde veri penceresinin yerleşmesine izin verecektir. Daha gelişmiş olan analizörler bir izlemenin yürütülmesinden önceki tetikleme kelimelerinin veya patern sırasının tanınmasını sağlayabildikleri gibi belirlenmiş kelime paternleri arasındaki zaman olaylarını da sayabilirler.

3-) Lojik analizör bir defada oluşan olayların yakalanmasını sağlamak için, iç saklama yeteneğine sahiptir.

4-) Tetikleme kelimelerinden önce oluşan olayların yakalanabilmesini

sağlayan "negatif zaman yeteneği" uygulanabilmektedir.

5-) Uygun tetikleme ve etkili olarak bellek kullanımını sağlamak için tetikleme ve saat niteleyicileri (tetiklemenin ve veri örneklemesinin zamanlamalarını düzenleyen analizör dışı işaretler) kullanılmaktadır.

6-) Lojik durum analizörü, sonuçları ya "harita", grafik veya liste şeklinde görüntüler. Bütün fonksiyonel görüntüler program akışının izlenmesine izin verirken hataların belirgin olarak büyütülmesini sağlar.

7-) Test altındaki sistemde kullanıldığı gibi, verileri aynı aktif saat darbesi kenarında izlemek için, analizörün bir "saat darbesi kenarı seçme" kontrolü vardır. Belirsiz veya hatalı verileri izlemekten kaçınmak için analizörün "sıfır yakalama zamanı" vardır. (Aktif saat kenarından kısa bir süre önce kararlı duruma gelmesi gereken veriyi, bu kenarı izleyen kısa bir süre tutabilme özelliği)

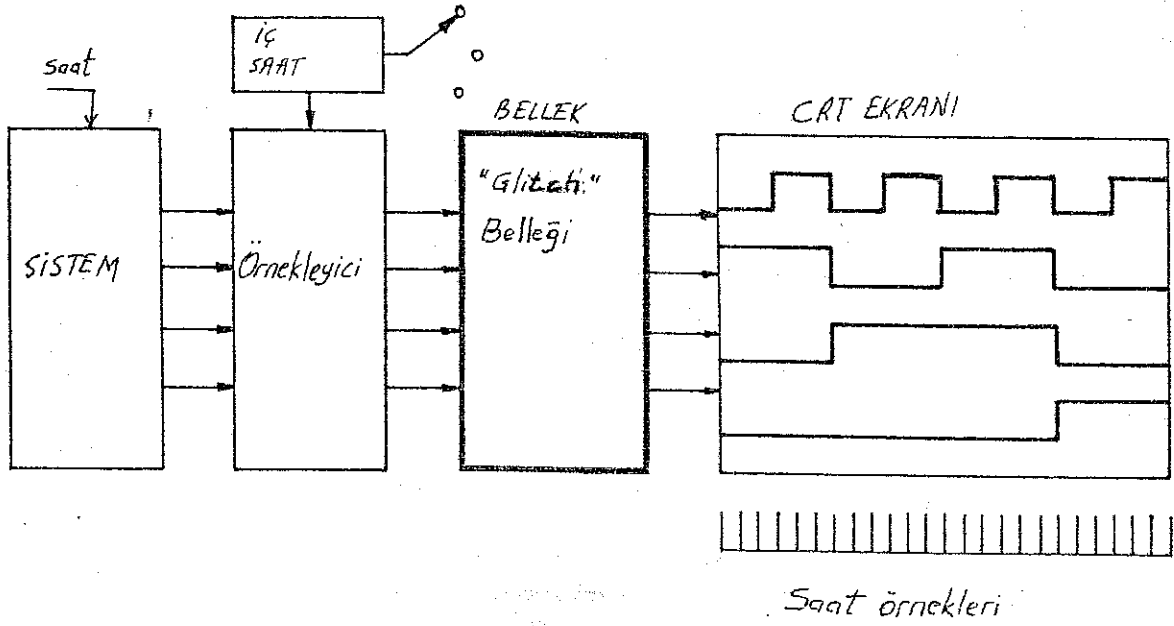
8-) Tek bir kanala zaman domeninde gözlemeye izin vermek için analizörün, zaman domeni cihazlarına, özellikleri belirlenmiş durum sırasına senkronize eden bir tetikleme çıkışı vardır.

BEŞİNCİ BÖLÜM

LOJİK ZAMAN ANALİZÖRLERİ

5.1 GİRİŞ

Lojik zaman analizörünün, lojik durum analizöründen en belirgin ayırımı, zaman analizörünün kendine ait bir iç saatinin bulunmasıdır. Veri bu saat darbesinin aktif kenarında örneklenir. Bu içte bulunan saat, test altındaki sisteme göre asenkron olarak çalıştığından, analizöre "asekron analizör" denir. Buna karşın durum analizörünü "senkron analizör" denir. Bu durum şekil-5-2 de gösterilmiştir.

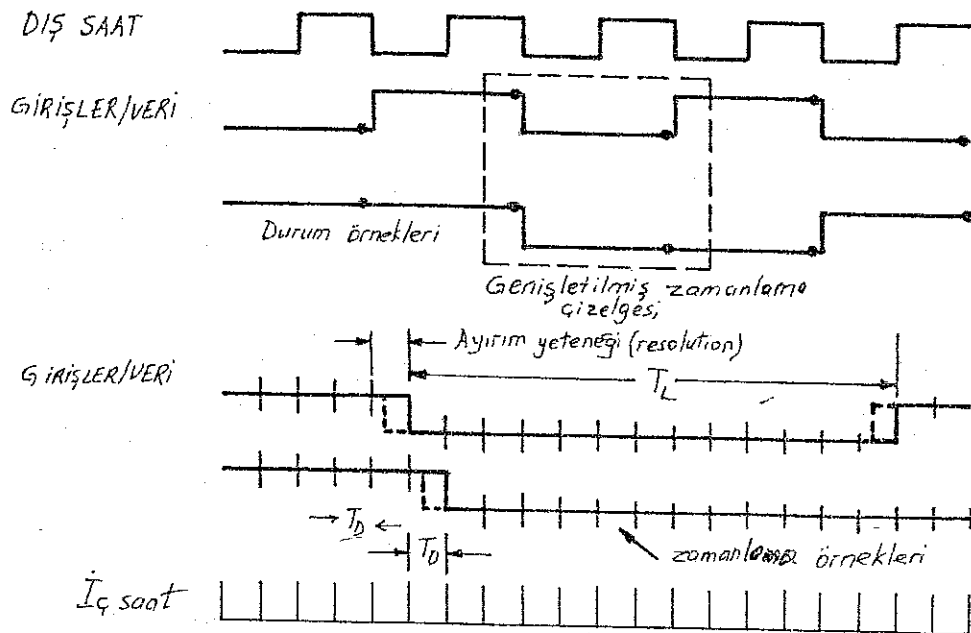


Şekil-5-1 Dış sistemi asenkron olarak örnekleme için lojik zaman analizörünün bir iç saati vardır. Sonuçlar, bir zamanlama çizelgesi olacak şekilde görüntülenmiştir ve lojik durumlar, zaman analizörünün iç saatine bağlı olarak gösterilmektedir.

Durum analizörü, sistem birinci saat darbesi durumunda (1.0), ikinci saat darbesinde (0.1) durumunda iken v.b. veri sağlar. Durum analizörü,

bir zamanlayıcı (timer) ilave edilmedikçe yalnız durum ve durum sırası verilerini sağlar, zaman verilerini sağlamaz. Diğer yünden, zaman analizörü , üstteki işaretin T_L saniye süresince lojik 0 olduğunu ve kanallar arasındaki zaman farkınının T_D saniye boyunca lojik 0 a gittiğini gösterir. Zaman analizörü durum sırasını çözmez. Örneğin, sistem bir dış saat darbesinin yalnız bir çevrimi için (0,0) durumunda olabilir veya bu durumda pek çok çevrim süresince kilitli (lock) kilitli olabilir. Bu veri doğrudan doğruya zaman analizörü tarafından sağlanmamıştır.

Zaman analizörü, I/O işaretleri veya kontrol hatları arasındaki zaman ilişkilerinin ve lojik kapılar arasındaki veya lojik kapılarda meydana gelen yayılma gecikmeleri (propagation delays) gözlenimlerine yöneliktir. Durum analizörü, sayısal sistemlerin ortak yollarındaki durum sırasının izlenmesinde daha yararlıdır.



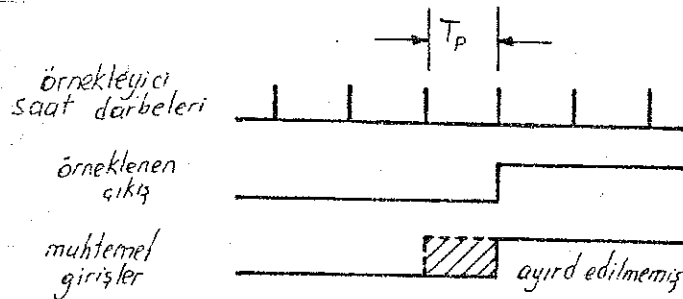
Şekil-5-2 Lojik durum analizörü, dış veya sistem saat darbelerinin aktif geçişlerinde lojik durumu belirginleştirir. Bir lojik zaman

analizörü içinde kullandığı saatin darbelerinde veriyi örnekler ve şekilde olduğu gibi çözümler. İki işaret arasındaki fark azalıp T_T (saniye) olurken ilk işaret T_L (saniye) boyunca lojik 0 olarak görüntülenir.

Zaman analizörü ekranı bir zamanlama çizelgesidir, osiloskop dalga şekli değildir. Analizör veriyi her iç saat darbesinde örnekler ve verinin eşik geriliminden daha büyük veya daha küçük olduğunu belirleyerek sonuçları lojik 1 veya lojik 0 olarak görüntüler. Bir sonraki örnek alınmasına kadar görüntü bu değerde kalır. Düşük yükselme zamanı veya gürültü gibi analog değerler yoktur. Zaman domeni ölçümleri ya bir "yüksek" veya "alçak" seviye şeklinde sonuçlanırlar.

5.2 ZAMAN ANALİZÖRÜNÜN AYIRIM YETENEĞİ VE KANAL GEÇİŞLERİNDEKİ GECİKME FARKI (RESOLUTION AND SKEW)

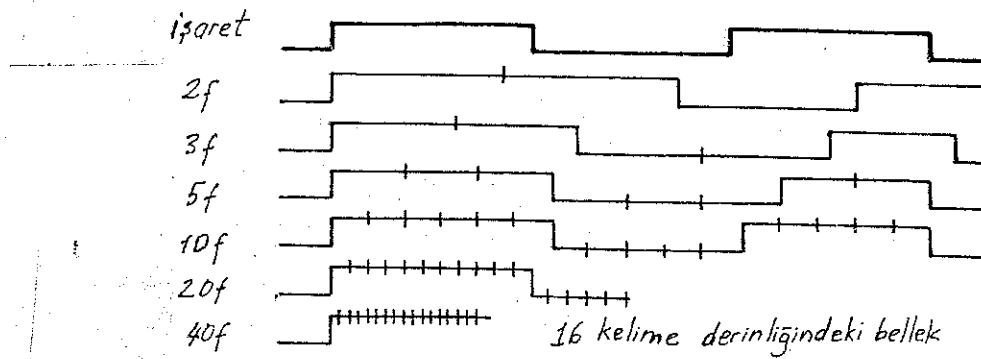
Asenkron bir cihaz kullanırken en önemli düşünce, ölçme yaparken ayırımdaki yeterlilik derecesidir. Temel olarak, ayırım yeteneği (resolution) örnekleyen saat darbesinin periyodundan daha iyi olamaz. Bu durum şekil-5-3(a) da gösterilmiştir.



Şekil-5-3(a) Bir dalga şekli örneklendiği zaman, işaret yalnızca saat darbesi periyodunda ayırım edilebilir.

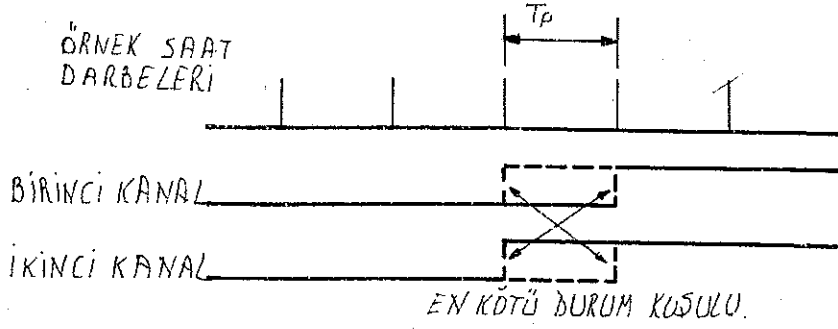
Belirsiz işaretler iki ardışıl örnekleme saat darbesi arasındaki herhangi bir zamanda lojik 1 seviyesine yükseltilirler. Ayırım yeteneği

T_P dir. ve örnekleyen saatin periyodudur. Eğer ayırım yeteneğinin $0,1 \mu s$ olması isteniyorsa, 10MHz. lik bir saat kullanılmalıdır. En yüksek derecede bir ayırım yeteneği sağlayabilmek için saat frekansı mümkün olduğu kadar yüksek olmalıdır. Verilen bir bellek kapasitesi için yüksek frekanslı bir saat yalnızca dar bir zaman sınırları içindeki verileri izleyecektir. Test altındaki sistemin veri hızının 5-10 katı örnekleme hızı normal olarak, ayırım yeteneğiyle yeterli bir sistem zamanlaması arasında akla yakın bir yaklaşım sağlar. Bu durum şekil-5-3(b) de gösterilmiştir.

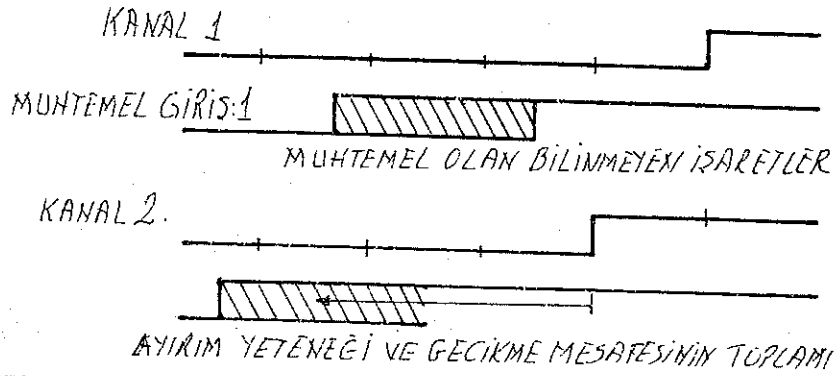


Şekil-5-3(b) Bir zaman analizörü ile örnekleme yapıldığı zaman bir kare dalganın ayırım yeteneği örnekleme frekansına bağlı olacaktır. Frekans yükseldikçe ayırım yeteneği daha iyi olur. Fakat analizörün zaman penceresinin sınırları (analyzer time window) daha küçük olur. Çizimde kolaylık sağlama amacıyla, bellek 16 bit genişliğinde kabul edilmiştir. İlk örnekleyici darbe, giriş işaretinin yükselen kenarını yakalamıştır. Şekilde gösterilen durum en kötü durumdur.

Sayısal sistemlerin çoğu, en çok bir kaç MHz. veri hızında çalıştırlarından , pek çok zamanlama problemlerinin çözümünde, 10-20 MHz. lik bir saat hızı yeterli olacaktır. Anahtarlama sırasında oluşan dar sığrılar (glitch) , "glitch" yakalama devreleri tarafından tespit edilirler.



Şekil-5-4(a) Kanallar arasındaki $\pm T_p$ ayırımını gösteren en kötü durum koşulları şekildedir.



Şekil-5-4(b) Bir örnekleme saat darbesi tarafından yerleştirilen iki kanalı gösteren bir zaman analizörü görüntüsü şeklindeki diyagramda belirtilmiştir. Ayırım yeteneği ile gecikmenin toplamı görüntüden çıkarıldığı zaman bilinmeyen girişler şeklindeki taranmış bölgelerde olabilirler. Kanaldan kanala olan geçişteki gecikme (skew) değişiklikleri sonucu, gerçek girişler farklı bir dizide meydana gelebilirler.

Bir tür "glitch" yakalama devresi, konu olan test çalışmasında, tasarıma eklenmiştir. Bu devre 70 ns. den küçük darbeleri tespit edebilmektedir.

Şekil-5-3(a) da bir işaretin tam geçiş anında örneklenebileceğini göstermektedir. Bu an kaçırılırsa bir örnekleyici saat periyodu geçene kadar işaret yakalanamayacaktır. İki veya daha fazla kanal izlenirken

her iki kanaldan biri diğere (veya diğelerine) göre kayabilir, ayırım yeteneği, artı veya eksi örnekleyici saat periyoduna ($\pm T_p$) eşit olabilir (şekil-5-4a).

Pratikte probe'ların fiziki boyutları nedeniyle, probe giriş noktaları ile, örnekleyici arasında aslında, elektriksel bir gecikme oluşur. Herhangi bir kanaldaki gecikme veya analizör tarafından örnekleme yapıldığı aktif saat kenarından hemen önceki yerleşme zamanı (set-up time) gecikmesi bir problem yaratmaz, fakat kanallar arasındaki veya kanal geçişlerindeki gecikme farkı (skew), problem yaratır. Pratikte kanal geçişlerindeki gecikme farkı 10 ns. olabilir. Şekil-5-4b de görüldüğü gibi kanal geçişindeki gecikme farkı, görüntülenmiş işaretlerin gerçekte olduğu gibi meydana gelmelerinden farklı bir sırada görüntülenmelerine neden olabilir.

5.3 ZAMAN ANALİZÖRÜNÜN ASENKRON OLARAK TETİKLENMESİ

Zaman analizörü asenkron bir cihazdır. Bir geçici durumda giriş verilerini örnekleme için iç saat kullanılır. Örnek alındıktan sonra, bu değer bir sonraki saat darbesine kadar kabul edilen durum olur. Bu durum, ayırım yeteneğinin sınırları içerisindeyken, sistemin gerçek bir temsili olmayabilir ve tetikleme koşullarını sağlamak için, diğer işaretlerle bir araya gelebilir. Şekil-5-6 da analizör (1,1) geçici durumunu yakalar. Bu durum bellekte saklanır ve durum analizöründe olduğu gibi tetikleyici lojik devresi, örnekleyiciden sonra yerleştirilmiş ise bu (1,1) durumu tetikleme koşullarını meydana getirir. Diğer bir çözüm şeklinde, tetikleyici lojik devresi içerisine bir zaman süzgeci eklenir. Tetikleme koşullarının süresi bir örnekleme periyodundan daha büyük olarak sağlan-

dığı zaman tetikleme devresi uyarılır. Bu düzenleme, analizörün bazı koşullarda tetiklemesini garantilemek için, garantileme periyodu kısa tutulmalıdır. Fakat , tetiklemeden sonraki faaliyetler yakalanacak ise, örnek-leme periyodu nisbeten uzun tutulmalıdır. Genelde, yüksek duyarlıklı parametrik ölçümler, bir zaman analizörlerinden beklenemez ve bu nedenle daha çok fonksiyonel ölçmelerde kullanılır. İki işaret arasındaki zamanlama önemli ise tetikleyici lojik devresi olan bir zaman analizörü, bu iki işareti ayırmak için kullanılabilir.

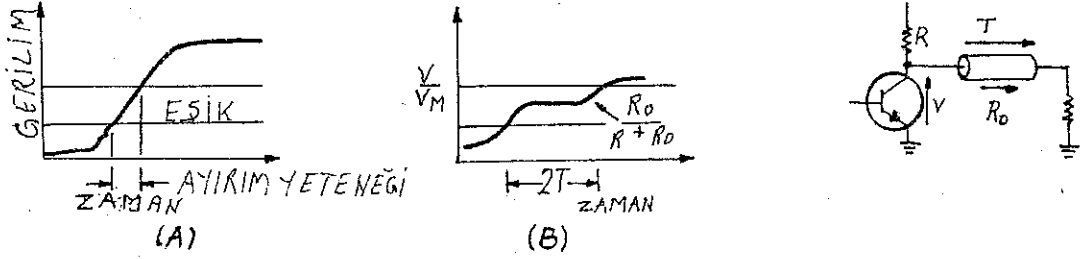
5.4 TEST İŞARETİNİN ÖZELLİKLERİ

Zaman analizörlerinin asenkron tetiklenmesinin tartışılması, gözlem altındaki işaretlerin keskin yükselme ve düşme zamanlarının olduğu kabul edilerek yapılmıştır. Bu koşullarda, deney altındaki sistem ve analizörlerin eşik gerilimleri arasındaki farktan meydana gelen zamanlama hataları olmayacaktır. Bununla beraber pratikte elemanlar sonlu yükselme ve düşme zamanlarına sahiptirler. Bunların her ikisi de analizörün yüklemesinden etkilenebilirler. Bir mikrobilgisayarın saati 50ns. ye kadar yükselme ve düşme zamanına sahip olabilir. Özellikle kapasitif yüklemeli mikrobilgisayar hatları için, bu zamanlar daha uzun olabilirler.

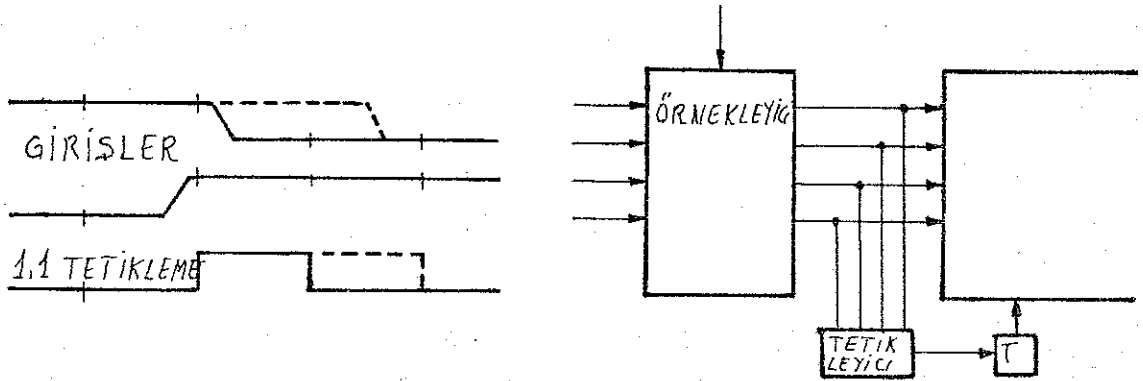
Bu sonlu zamanlar nedeni ile gözlem altındaki işaretler eşik gerilimi sınırları içerisinde uzun bir zaman süresince kalır ve bir zaman aralığında analizörünün zaman ayırma yeteneğini kısıtlarlar.

Elektriksel boyutların küçük olmadığı devrelerde veya yüksek frekanslı probe'lar analizörde kullanılmamışsa, transmisyon hatlarının etkileri önemli olur. Çıkış gerilimi, her biri kısmen transmisyon hattının karakteristik empedansının, devre empedansına oranı $R_o / (R + R_o)$ olarak bilinen

basamaklar serisi olarak son değerine ulaşır. Bu basamaklardan bazıları, analizörün doğruluğunu azaltarak, eşik gerilimi sınırlara içerisinde olabilir. Tipik bir kabloda, elektriksel işaretler 20 cm./ns. hızla akacaktır ve kablunun her 10cm. si için 1ns.lik basamaklar meydana gelecektir.



Şekil-5-5(a) Tipik bir devre, lojik durumlar arasındaki belirsiz bölgelerde belirli bir zaman harcayacaktır. Bu zaman, bir lojik analizör ile muhtemel olan zaman ayırma yeteneğini azaltır. (b) yüksek frekanslarda devreler transmisyon hatlarının karakterini gösterir. Çıkış gerilimi en son değerine, bir dizi adımlar halinde sıçrayacaktır. Bu adımlardan biri belirlenmemiş bölgede kalabilir ve bir analizörün, gözlem altındaki sistemin ayırma yeteneğini azaltır.



Şekil-5-6 Bir asenkron analizör, (1,1)in bir geçişi durumunda örnek almak suretiyle bu veri saat periyodu süresince sabitleştirecektir. Durum analizöründe kullanıldığı gibi, eğer bu veri tetikleme kelimesiyle uyuyorsa tetikleme devresi tetiklenecektir.

Dalga şekilleri nedeniyle zamanlama ölçümlerindeki belirsizlikler, her bir analizör probe'unun eşik seviyesini dikkatlice ayarlayarak

azaltılabilir. Sonlu yükselme zamanlı işaretler de yükselme kenarı için bir seviye, düşme kenarı için ikinci bir seviye olmak üzere, çift eşik gerilimi kullanılır. Yüksek ayırım yetenekli ölçmelere ihtiyaç olduğu zaman, çift ışınlı bir osiloskop kullanılmalıdır. Böylece iki işaretin aynı anda gözlemlenmesi sağlanmış olur. İstenilen zaman çerçevesi içerisinde, osiloskop'u ayrıntılı olarak gözlemek üzere, lojik analizör osiloskop'u tetiklemede kullanılmalıdır.

Neticede, yüksek frekanslı, asenkron analizörlerle birlikte kullanılan yüksek ayırım yeteneğine sahip ölçmeler, lojik eşik seviyesi yakınındaki tanımlanmamış bölgede, gözlem altındaki işaretler için kullanılan sonlu zaman nedeniyle, sistemin durumu gerçek olarak temsil edilmeyebilir. Analizör probe'larının yükleme etkisi, çok yüksek frekanslarda, işaretin özelliklerini bozar ve neticede ölçmenin doğruluğu azalır.

5.5 SONUÇ

Zaman analizörleri aşağıdaki tasarım kriterlerini sağlamalıdır.

- 1-) Cihaz, bir çok hat üzerinde aynı anda oluşan geçişleri izleyebilmek için çok düğümlü olmalıdır.
- 2-) Analizör, yeterli ayırım yeteneğine (resolution) sahip olmalıdır.
- 3-) Analizör kısa süreli sıçramaları (glitch) güvenilir derecede tespit edebilmeli ve belirgin olarak görüntüleyebilmelidir.
- 4-) Tetikleme noktasından uzakta, genişletilmiş bir görüntü penceresi elde edilmesine izin vermelidir. Bir zaman analizörü, durum ve zaman birimleri olarak geciktirme koşullarına sahip olmalıdır.

Bir sayısal sistemde karşılaşılan pek çok problemler, değişik sayı-

da cihaz ve teknik kullanarak ortaya çıkarılabilir. Çözüm için asenkron cihaz ve teknikler gerektiren problemler (zamanlamamanın çok önemli olduğu, özellikle olayların kolayca tekrarlanmadığı durumlarda veya tetiklemeden önceki negatif zamanda oluşan olaylarda), çok kanallı olayların gözlemlenmesini, kısa süreli sıçramalarda tetikleme yakalama ve görüntüleme olaylarını içerir.

Bu problemler, çoğunlukla bilgisayar sistemlerinin kontrol hatlarında ve bilgisayar ile dış elemanlar arasındaki bağlantı elemanlarında meydana gelirler.

ALTINCI BÖLÜM

TEZDE GELİŞTİRİLEN LOJİK ANALİZÖR

6.1 LOJİK ANALİZÖRÜN TEMEL ÇALIŞMA PRİNSİBİ

İlk olarak FF1 ve FF2 "flip-flap"ları Q çıkışları lojik 0 olacak biçimde "reset" edilir. Önceden değer verilebilen (presetable) bir bölücü ile birleştirilmiş bir saat osilatörü, çıkışları, 8×1024 bit'lik bir yaz/oku belleği (RAM) adreslemeyi sağlayan bir A sayıcısı için saat darbeleri üretir. DO...D7 kanallarından örneklenen sayısal işaretler 8 bit'lik "latch" yazıcı yardımı ile, saat frekansında belleğe yazılırlar.

1023 saat darbesinden sonra A sayıcısı "reset"lenir, çıkışlar lojik 0 durumuna gelir. Bellek bir kez daha gelen veri ile dolmaya başlar. Bir tetikleme darbesi üretildiği an, FF1 "flip-flap"ı B sayıcısını saymaya bağlatmasına neden olarak, durumunu değiştirir. Bu (B) sayıcısının ilk durumu, "tetikleme şeklini" seçen bir anahtar ile belirtilebilir. "Tetikleme sonrası durumunda, B sayıcısı ilk durumu 0_{10} 'a getirecektir. "Tetikleme ortası" ve "tetikleme öncesi" çalışma şekillerinde, B sayıcısı sırası ile 0,256, 512, 768 durumlarına getirecektir. B sayıcısına verilen ilk değere bağlı olarak B sayıcısını doldurmak için, belirli bir sayıda saat darbesi gerekecektir ve sonra sayıcı bir darbe (carry) üretecektir. Sayıcının bu noktada ürettiği darbe FF2 "flip-flap" ını "set" durumuna, yani Q çıkışını lojik 1 durumuna getirecektir. Böylelikle daha fazla yeni verinin belleğe okunması önlenmiş olacaktır. Örneğin, tetikleme anahtarı "tetikleme

sonrası" durumunda iken, gelen veriyi belleğe yazma işlemi , yazma çevrimi durdurulmadan önce 1024 saat çevrimi boyunca devam edecektir.

Diğer bir deyimle tetikleme darbesi üretildikten sonra giren 1024 "byte" lık veri bellekte saklanacaktır. "Tetikleme ortası" durumunda tetikleme darbesinden önceki 512 (byte:kelime) ve sonraki 512 kelime saklanacaktır."Tetikleme öncesi" durumunda ise tetikleme darbesinden önceki 256, 768 kelime saklanacaktır. Bu çok kullanışlı ve çok değerli bir özelliktir. "Tetikleme darbesini" üretebilmek için üç yöntem vardır.

Birincisi ve en basit olanı, "dış" bir tetikleme işareti kullanmaktır. Bu amaçla test altındaki devrede bazı düğümler, istenen anda böyle bir darbe vermeleri için kullanılabilir.

İkinci yöntemde, bir tetikleme darbesi gelen veriden çıkarılır.

Sonuncu yöntem, ilk iki yöntemde kullanılanların bir kombinasyonudur. Bu durum için , bir kelime tanıyıcısı gerekecektir. Adından da anlaşılacağı gibi, gelen veri yerleştiği zaman , devre 8 bit'lik bir kelimeyi tanımlar. Tüm giriş verisi, ilk olarak 8 bit'lik bir yazıcıya (latch)yazılır. Bu yazıcının kapsamındakiler kelime tanıyıcısında önceden belirtilmiş olan kelimeye eşit olduğu zaman bir tetikleme darbesi üretilir.

Şimdi, yaz/oku belleğinin (RAM) kapsamındakiler okunabilir ve osiloskopta görüntülenebilir.

FF 2 (flip-flap) , "set"edildiğinde, Q çıkışı lojik"1" durumuna gelir. Bu, flip-flop'u aynı zamanda tüm sistemi önceden verilmiş bir saat frekansında , sabit bir tarama (scan) frekansına geçiren bir S 2 anahtarının uyarılmasına neden olur. MMV (monostable multivibrator),

B sayıcısı tarafından üretilen her işaretle darbelenecektir (carry-out). Bu osiloskopun zaman tabanını, yeni bir tetikleme hattına hazırlamak için saat osilatörünün , MMV' nin zaman periodu boyunca önlenmesini sağlar. Bu her geçişten sonra, aynı anda C sayıcısının kapsamındakiler bir arttırır, bir tetikleme işareti osiloskopa aktarılır. Sonra , bir hat, C sayıcısının durumunun belirttiği bir durumla ekran üzerinde gösterilir. Bu üç bit'lik sayıcının (C sayıcısı) çıkışları, skobun Y girişine doğrudan doğruya bağlanan bir D/A (sayısal analog) dönüştürücüyü kontrol eder.

Tetikleme darbesinden sonra A sayıcısı saymaya devam ederek, yaz/oku bellekte mevcut saklanmış olan veri bir çoklayıcıya (multiplexer) geçirir. C sayıcısının kapsamındakilerin değişmemesi koşulu ile çoklayıcı , her saat darbesinde , gelen kelimelerin (byte) bir tek bit'ini D/A çeviricinin en az ağırlıklı bit'ine (LSB) geçirecektir. Bu şekilde, bir tek giriş hattındaki tüm 1024 bit'lik veri, D/A

(sayısal analog) çeviriciye aktarılır ve ekran üzerine yazılır. Bir lojik "1" durumunun oluşumunda, osiloskopun Y girişindeki gerilim seviyesi hafifçe arttırılacaktır. Lojik "0" durumunda ise C sayıcısının kapsamındakilere bağlı olan sabit bir seviyede kalmasına neden olacaktır. Bu sebepten, tüm veri hattındaki sayısal bilgilerin ekran üzerinde görünmesi sağlanmış olur.

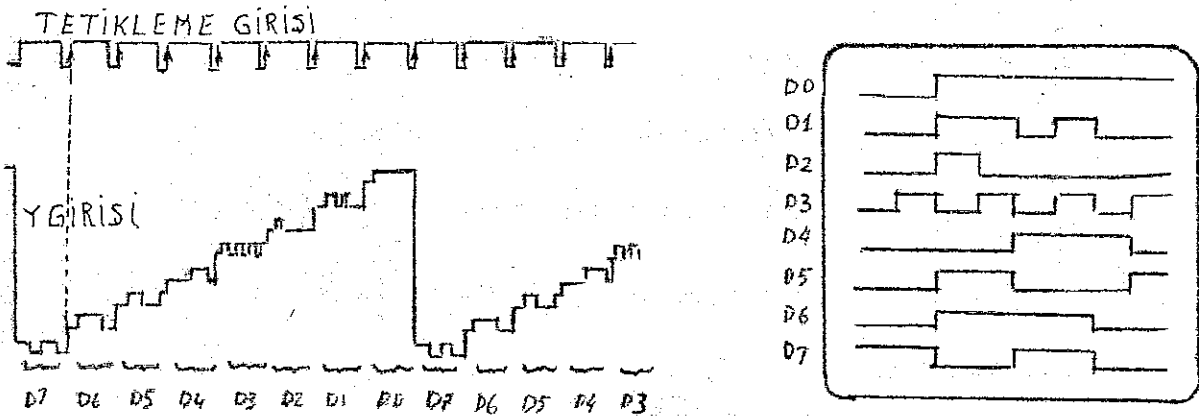
Eğer C sayıcısının kapsamındakiler ϕ/ϕ ise D/A çeviricinin çıkış gerilimi seviyesi 0 V. olacaktır ve hat ekranın en alt kenarı boyunca izlenecektir. Sonra çoklayıcı, yaz/oku belleğin (RAM) D 7 veri hattına, bellek hattının dışarı okunması için gereken zaman periyodu boyunca

D/A çevirici üzerinden anahtarlacaktır.

1024 saat darbesi sonra, D 7 hattındaki tüm bilgi dışarı yazılmıştır. Sonra B sayıcısı bir "carry" işareti üretecektir ve MMV tetiklenecektir. MMV'nin zaman periodunun sonunda, C sayıcısının kapsamında kiler bir arttırılacak ve aynı anda osiloskop tetiklenecektir. Şimdi ekrana yazılan hat bir önceki hattan hafifçe daha yüksek olacaktır. Önce olduğu gibi çoklayıcı, yaz/oku belleğin veri çıkış hattını D/A çeviriciye sadece bu zamanda bağlar. Artık veri hattı D 6 dır. Bu hattaki tüm bilgi şimdi yükselen ikinci hat üzerinde ekranda görünecektir.

Yukardaki çalışma, belleğin 8 veri hattına ekran üzerinde karşılık düşen 8 hat oluşuncaya kadar tekrarlanır. Bu durum her biri 1024 bit'e sahip 8 hat , belleğin tümünü temsil eder.

Bellekten okuma çevriminin tümü tamamlandıktan sonra, bu çevrim yeni baştan başlatılarak tekrarlanır



Şekil-6-2, RAM' in içerdiği bilgilerden çıkarılan 8 hattı temsil etmektedir.

Şekil-6-2. D/A çeviriciden ekrana beslenen gerilim seviyelerini göstermektedir. En üstte gösterilen işaret tetikleyici girişidir ve

yazılan her yeni hat için bir darbe sağlar. Y girişinin dalga biçimi, bellekten alınan her 1024 bit'lik veri hattı için bir basamak oluşturan basamaklı gerilim seviyelerini kapsar. Şekil-6-2, osiloskop ekranında her hat, Y girişinde bir basamağa karşılık düşmekte olduğunu göstermektedir. Bu örnekte az sayıda veri bit'leri gösterilmiştir, gerçekte ekran boyunca her hat 1024 bit'e kadar veri kapsayabilir.

6.2. TEZDE GELİŞTİRİLEN LOJİK ANALİZÖRÜN ÇALIŞTIRILMASINDA BAZI AYRINTILAR

Bu bölümde, tezde geliştirilen analizörü oluşturan tümdevrelerin çalışma biçimleri ayrıntılı olarak incelenmiş olup, fonksiyonel çalışma durumu gözden geçirilmiştir.

Analizörde, LSTTL (low power schottky TTL) ailesi kullanılmıştır. Bu ailenin elektriksel özellikleri ek'te belirtilmiştir. Lojik analizörün en önemli kısmı, bir 10 MHz'lik osilatör ile zaman tabanlı bir S l anahtarından meydana gelmiştir. Farklı örnekleme hızları, osilatör frekansından bölücü katlar (74LS390) yardımı ile elde edilmiştir. Ayrıca bir dış saat girişi bu anahtara eklenmiştir. Lojik analizör saatin yükselen kenarında örnekleme yapmaktadır.

S1 f 250ns 500ns 1µs 5µs 10µs 50µs 100µs 500µs 1ms

Saat kontrolü devresi, "NAND" ve "INVERTER" kapılarından meydana gelmiştir. Bu kapılar (74LS14 ve 74LS00) sistemi, seçilen örnekleme frekansından, sabit tarama frekansına (1µs) veya sabit tarama frekansından seçilen örnekleme frekansına anahtarlamayı sağlarlar. Ayrıca bu kontrol devresinde saat darbesini daraltan bir türev alıcı devre ile

bir geciktirme (alçak geçiren) devresi vardır.

8 veri girişi 74LS374 tümdevresine saklanır(latch). Bu tümdevre giriş verisini paralel olarak aynı anda, S1 anahtarı ile seçilen örnekleme hızında çıkışlarına transfer eder, veri transferi ile örnekleme darbesi arasında geçen gecikme zamanı, MMV1 (74LS123)-monostable multivibrator) ile önceden ayarlanabilir. Geciktirme anahtarının (S3) iki durumu vardır. Anahtar (a) durumunda olduğu zaman geciktirme zamanı 50 ns. olacaktır. (b) durumunda olduğu zaman, P1 potansiyometresi ile 150-500 ns. arasında ayarlanabilir. MMV1'in A girişi verinin 74LS374 tümdevresine yazılması sırasındaki örnekleme darbelerini üreten C3(NAND) kapısının çıkışına bağlanmıştır. Verilerin ekranda gösterilmesi sırasında, bu saklayıcı (74LS374), yeni verilerin lojik analizöre alınmasını önler. Bu durum saklayıcının \overline{OE} (output control) girişinin, FF3 flip-flop'u tarafından lojik 1 durumuna getirilerek sağlanır.

Lojik analizörde 1KX8 (1024X8) kapasitesinde bir yaz/oku bellek (MK-4118) bulunmaktadır. Bu belleğin erişme zamanı (access time) 250 ns.dir. Böylelikle lojik analizörün en yüksek örnekleme hızı 4MHz.'i aşamaz. Belleğin 10 bit'lik adresi bir adres sayıcısı tarafından sağlanır. Bu sayıcı üç adet 4 bit'lik senkron sayıcı tümdevresinden (74LS191) meydana gelmiştir. İlk anda, tetikleme işaretinin, tetikleme flip-flop'unu (FF1) tetitlemesi, adres sayıcısının tüm belleği (1024 kelime) tarayacağına kadar önlenmiştir. Bu nedenle ilk anda belleğin, geçerli veri ile doldurulması garantilenmiştir. Bu durumu sağlayan FF4 flip-flop'udur. FF4 , ilk 1024 kelimenin belleğe yazılması sırasında FF1 'in D girişini lojik 0 da tutarak, saat (CP) girişine gelen bir tetikleme darbe-

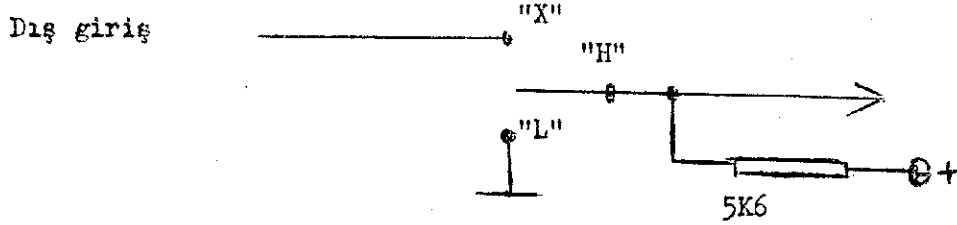
sinin Q çıkışını lojik 1 durumuna getirmesini önler. Adres sayıcısı bellekteki tüm adresleri sürekli olarak tarar. Bu sayıcının saat girişi, örnekleme darbelerini üreten C3 (NAND) kapısının çıkışıdır. Sayıcının en ağırlıklı çıkışları üzerindeki anahtarlar verilerin osiloskopta gösterilmesi sırasında belleğin sadece seçilen bir bölgesini göstermesini sağlar.

74LS374 saklama tümdevresini geçen 8 bit'lik veriler, aynı zamanda bir kelime tanıyıcısına (word recognizer) uygulanmıştır. Bu tanıyıcı, iki adet 4 bit'lik 74LS85 (sayısal karşılaştırıcı) tümdevresinden meydana gelmiştir. İki 74LS85 tümdevresi kaskod bağlanmıştır. (8 bit'lik sayısal karşılaştırıcı). Bu tüm devrelerin B0...B3 veri girişleri S5...S12 anahtarlarına, A0...A3 girişleri saklayıcının (latch) çıkışlarına bağlanmıştır. Tanıyıcı iki tabanında 8 bit'lik A ve B (önceden anahtarlarla belirtilmiş kelime) kelimelerinin büyüklüklerini karşılaştırır. Bu seçilen kelime saklayıcının (74LS374) çıkışlarında görüldüğü zaman (eşit olma durumu) FFI flip-flop'u kelime tanıyıcısı tarafından tetiklenecektir. Kaskot bağlanan iki 74LS85 sayısal karşılaştırıcı tümdevrelerinde azami yayılma gecikmesi 46ns.dir.

Sistem, gösterme (display) mod'unda iken bellekte olan tetikleme kelimesinin periodik olarak (1024 saat darbesinde bir) kelime tanıyıcısını çalıştırmasını önlemek için, tetikleme flip-flop'unun (FFI) \bar{Q} çıkışı, 74LS85'in $I_A = B$ girişine bağlanmıştır. Ters halde, kelime tanıyıcısının FFI saat girişinin birden fazla tetiklenmemesi sağlanmıştır.

Sayısal karşılaştırıcının B girişindeki anahtarlar üç durumlu olarak düzenlenmişlerdir:

Anahtarlar; "L" (lojik 0), "H" (lojik 1).ve "X" (dış giriş veya göz önüne almama) durumlarına sahiptirler.



FF1 tetikleme flip-flop'u dolayısıyla, lojik analizör üç şekilde tetiklenir.

1. Kelime tanıyıcısı (word recognizer)
2. Dışardan tetikleme (external trigger)
3. El ile tetikleme (manuel trigger)

FF1, tetiklendiği zaman ($Q=1$, $\bar{Q}=0$), lojik analizör, belleğe aldığı verileri osiloskopta görüntüler.

Reset devresi FF1 ile anahtarlardan(push-button) meydana gelmiştir.

Reset anahtarına basıldığı zaman:

- a. FF1, FF2, FF3, FF4 flip-flop'larının Q çıkışları asenkron olarak lojik 0 durumuna gelir. FF2 saklama /görüntüleme flip-flop'udur.
- b. C sayıcısı, adres sayıcısı reset'lenir.
- c. Saklama(74LS374-latch)tümdevresinin \bar{OE} ve 74LS245(buffer-tampon)

tümdevresinin \bar{CE} girişleri, FF3 tarafından sırası ile , lojik 0 ve lojik 1 durumlarına getirilirler. Saklayıcı 74LS245 tümdevresi test altındaki devreden 8 bit'lik kelimeleri aynı anda eş olarak örnekleme sureti ile belleğe yazar. 74LS245 tümdevresi ise çoklayıcıya veri geçişini önler. Bu sürücü (8 bit buffer) tümdevresi, sistem tetiklendiği zaman belleğin çıkışlarındaki verileri çoklayıcıya(multiplexer)geçirir. C sayıcısı, çoklayıcının hangi girişinin, çıkışa bağlanacağını belirtir.

D/A (sayısal -analog çevirici)devrede 4 adet dirençten meydana gelmiştir. Bu dirençler, osiloskopun Y girişi için gereken uygun orantılı bir analog gerilimi elde etmek için C sayıcısının çıkışlarını ve çoklayıcının veri çıkışını toplarlar.

S4 anahtarları ile önceden verilen değer (preset) ve FF1'den gelen tetikleme işareti yardımı ile tetikleme sayıcısı (E-F-C tümdevreleri), belleğe alınan verinin aynı şekilde bellekten okunmasını sağlar.

"Tetikleme şekli" S4 anahtarları, bu sayıcının girişlerine 0,256,512, 768 sayılarını yükler. Tetikleme sayıcısı önceden verilen bu değerlerden 1023'e kadar saymaya başlayarak, 1024 saat darbesinde G sayıcı tümdevresinin (74LS193) Q2 çıkışından bir darbe üreterek FF2 flip-flop'unu tetikler. Saklama/ gösterme (store/display) seçici flip-flop'u, lojik analizörü örnekleme durumundan, gösterme durumuna geçirir(anahtarlar).

6.3 SONUÇ

Bu bölümde lojik analizörün çalışması tüm alt sistemleri, tam bir "yükleme" ve "gösterme" çevriminde kısaca açıklanmıştır.

Başlangıçta, istenen örnekleme hızı S1 anahtarı ile seçilmektedir. Bir tetikleme kelimesi S5...S12 anahtarları ile lojik analizöre programlanmaktadır. S13 anahtarına basıldığı zaman, FF1, FF2, FF3, FF4 flip-flop'ları, adres sayıcısı reset'lenir. Tetikleme sayıcısına S4 anahtarları ile belirtilen sayı (örneğin S12) yüklenir. Şimdi, veriler 8 bit'lik kelimeler olarak saklayıcı (latch) tümdevresinin girişlerinden bellek girişlerine transfer edilirler. Bellek "yazma" durumundadır.

Adres sayıcısı, sürekli olarak belleğin tüm adres uzayını (1024 bellek hücrelerini) taramaktadır. Yani her gelen 8 bit'lik veri, bir üst adreste saklanmaktadır.

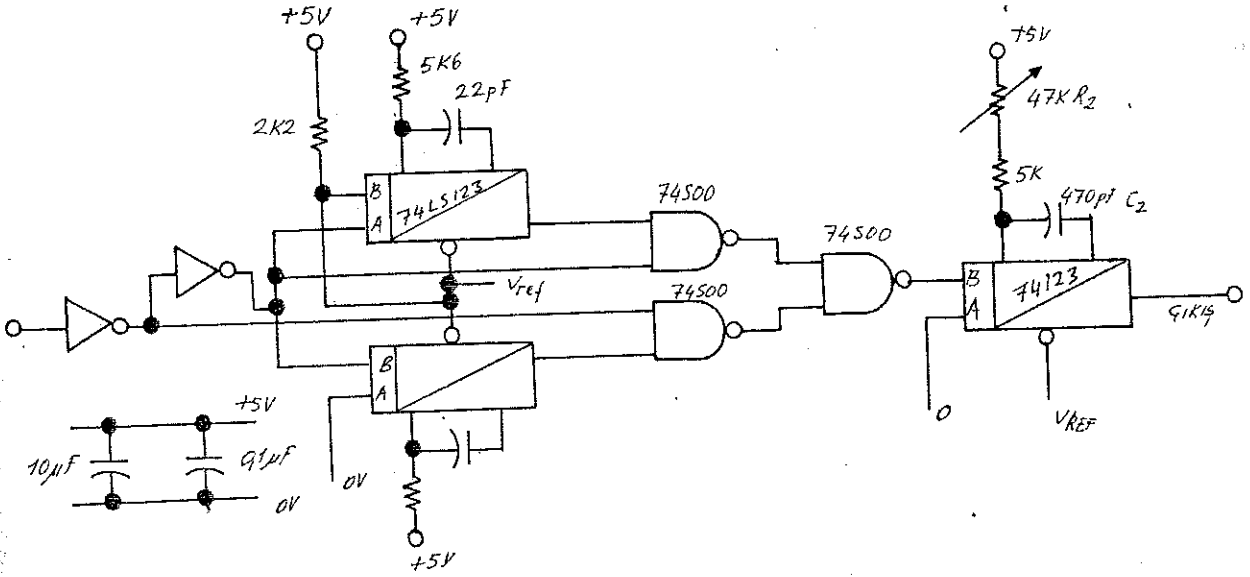
Gelen veri, S5...S12 anahtarları ile önceden verilmiş olan sayıya eşit olduğu zaman, kelime tanıyıcısı, FF1 tetikleme flip-flop'unu tetikleyecektir. D1 diod'u (LED) yanacaktır. Tetikleme sayıcısı,örnekleme darbelerini önceden verilmiş olan sayıdan itibaren saymaya başlayarak 1023'e kadar devam eder. Verilen ilk değere bağlı olarak, 1, 256, 512, 767 saat darbesi kadar geciktirme yapılabilir. Sayıcı sonra(1024.durum) bir darbe üreterek FF2 flip-flop'unu (saklama/gösterme) tetikler.Böylelikle lojik analizör,bellekteki verileri osiloskopta gösterme çalışma şekline girer. Şimdi bellek, sabit tarama frekansında (1MHz.) okuma durumundadır. İlk tarama sırasında veri hatlarından biri çoklayıcı tarafından seçilir ve ekran üzerinde görüntülenir. Bu taramanın sonunda (yani 1024 bit sonra) tetikleme sayıcısı tekrar bir darbe üretir. Önce olduğu gibi bu darbe, C sayıcısını (çoklayıcının hangi hattı seçeceğini belirten) bir artırır. Osiloskop, bir sonraki tetikleme darbesini alır.

Açıkça görüldüğü gibi 8 hat aynı anda ekran üzerinde görünemez. Fakat bu 8 hat yüksek bir hızda(tüm 8 hattın gösterilmesi 10ms'den az) çoklanırsa bu takdirde aynı anda "görünürler.

Sayısal olayların gelişimi sırasında oluşan kısa süreli sıçramaları(glitch) tesbit etmek için tasarlanan devre , tez'de geliştirilen lojik analizöre eklenmiş ve çalıştığı görülmüştür. Sıçramaları tesbit edici devrenin ayrıntıları ek:A da açıklanmıştır.

KISA SÜRELİ SİÇRAMALARI (GLITCH) TEBİT EDİCİ VE GENİŞLETİCİ DEVRE

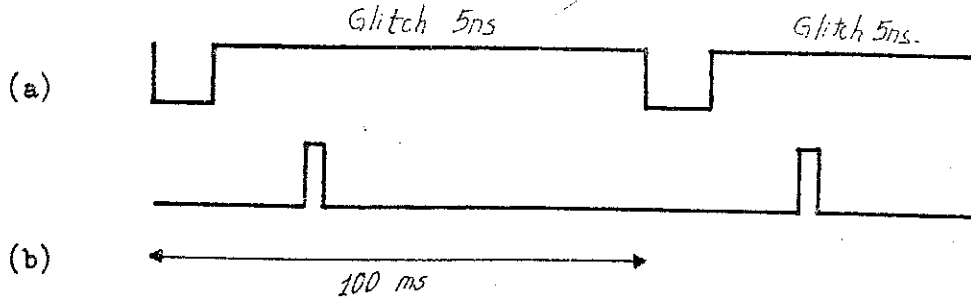
Karmaşık zamanlama işaretlerinde veya düşük tekrarlama hızı olan işaretlelerde bulunan ve karışıklık yaratabilecek özellikte olan kısa süreli sıçramaları (glitch) tesbit etmek bazan çok zor olur. Sıçramalar, adıgeçen işaretlere üreten kaynak tarafından meydana getirilirler.



Şekil-A-1. Sıçrama (glitch) arama devresi , labratuvar deney koşullarında ve gerçek koşullarda kullanılması uygun olan bir devredir. Bu devre, bir darbe dizisinde beliren sıçramaları tesbit eder ve uzunluğu ayar edilebilen bir çıkış sağlar.

Şekil-A-1.de görülen devre pozitif veya negatif gidişli olan bu sıçramaları (glitch) tesbit edip genişleterek çözümlerken, normal zamanlama darbelerine (timing pulses) karşı etkili olmaz. Sıçramalar, ikinci bir osiloskop kanalında kolayca görüntülenebilecek şekilde istenilen uzunluğa kadar genişletilmiş ve orijinal veri izi ile karşılaştırılmak için geçici olarak görüntülenmiş olurlar. Şekil-A-2. (a) da normal olarak skopta görüntülenemeyen sıçramaları (glitch) içeren tipik bir zamanlama işareti görülmektedir. Devrenin çıkışı, sıçramalar ileri kenarda (leading

edge) cımak üzere, şekil-A-2(b) de görülmektedir.



Şekil-A-2. Bir dalga şeklinde beliren sıçramalar, gözlenebilmek bakımından, aşırı kısalıkta olabilir. Tesbit edilen sıçramalar, bir tek kararlıyı (monostable) tetikler ve bu tek kararlıda kolayca tesbit edilebilen genişletilmiş çıkış sağlar.

Devre, her pozitif ve negatif gidişli kenarda birer adet tek kararlı (monostable) tetiklenme ile çalışır. Tek kararlı devrelerin çıkışları, tek kararlının periodu olan, yaklaşık 70ns.lik süreden büyük süreli olan normal zamanlama darbeleri tarafından men edilir. 70 ns.den kısa süreli darbeler, tek kararlı çıkışların kapılama devresinden (gating network) geçip, genişletici tek kararlı devreyi (expander monostable) tetiklemesini sağlar. Genişletici tek kararlı devre, kullanılan osiloskobun zaman bazı süresine (time base range) uygun olacak şekilde, istenilen oranda genişletmeye düzenlenebilir ve ayar edilebilir. Darbe genişliği ayırım yeteneği (puls width discrimination) R_1 ve C_1 ile kontrol edilen, iki, tek kararlı zaman sabitesi ile ayar edilmiştir ve görüntü darbe genişliği (display puls width) ise R_2 ve C_2 ile kontrol edilmiştir.

REFERANSIAR

1. Farnbach, W.A. "Logic State Analyzers--A New Instrument For Analyzing Sequential Digital Processes." IEEE Transactions On Instrumentation Vol. IM.24, No.4, December 1975.
2. Robin, Neil A. "The Logic Analyzer: A Computer Troubleshooting tool!" Computer design/ March 1976.
3. Hill, J. CARVER? "The State of Logic Analyzer." IEEE Spectrum December 1974.
4. Farley, BRUCE. "Logic Analyzers" Digital Design, Volume 10, No.1. January 1980.
5. Report, Staff . "Logic Analyzers" Digital Design, Volume 10, No.11 November, 1980.
6. Kneen, John. "Logic Analyzers For Microprocessors" Hayden Book Company, Inc, New Jersey 1980.
7. Hill J. Corver and Fiedler Cliff. "Logic Analyzers in System Debugging Make Time Run Backward" Computer Design, December 1975.