

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**FPGA ÜZERİNDE GÖRÜNTÜ İYİLEŞTİRME
UYGULAMASI**

Elektronik Müh. Tayfun ÇAYIR

FBE Elektronik ve Haberleşme Müh. Anabilim Dalı
Elektronik Programında Hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı: Prof. Dr. Vedat TAVŞANOĞLU (YTÜ)

Prof. Dr. Sabri Arık
J. Arık

Prof. Dr. Tülay YILDIRIM
T. Yildirim

Prof. Dr. Vedat Tavşanoğlu

İSTANBUL, 2010

Tavşanoğlu

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iv
KISALTMA LİSTESİ	v
ŞEKİL LİSTESİ	vi
ÖNSÖZ	x
ÖZET	xi
ABSTRACT	xii
1. GİRİŞ	1
2. SAYISAL GÖRÜNTÜ VE ÖZELLİKLERİ	3
2.1 Sayısal Görüntünün Tanımı	3
2.2 Sayısal Görüntünün Matrisel Gösterimi	7
2.3 Uzaysal Çözünürlük ve Pikselin Gri Seviyesi	11
2.4 Pikseller Arasındaki Temel İlişkiler	16
2.4.1 Piksel Komşuluğu	16
3. GERÇEK ZAMANLI GÖRÜNTÜ TEMİZLEME	19
3.1 Medyan Filtre	19
3.1.1 Medyan Filtrenin Özellikleri	19
3.1.2 Medyan Filtre Yapısı ve Kullanılan Şablonlar	22
3.1.3 Medyan Filtrenin Uygulanışı	26
4. MEDYAN FİLTRENİN FPGA İLE GERÇEKLENMESİ	33
4.1 Paralel Girişli Medyan Filtre Bloğu	34
4.2 İş Hattı Mantığıyla Oluşturulmuş Medyan Filtre Bloğu	37
4.3 Medyan Blokların Karşılaştırılması	40
4.4 İkili Karşılaştırmacı Yapıları	40
4.4.1 Lojik Karşılaştırmacılar	41
4.4.1.1 74LS85 4-bit Karşılaştırmacı	41
4.4.1.2 Optimize Edilmiş 74LS85 4-bit Karşılaştırmacı	45
4.4.1.3 74LS682 8-bit Karşılaştırmacı	49
4.4.2 Davranışsal Karşılaştırmacılar	51
4.4.3 Karşılaştırmacı Yapıların Birbirleriyle Kıyaslanması	52
4.5 Tezde Kullanılan 9-bit Karşılaştırmacı Yapısı	54
4.6 Piksel Değerlerinin Filtre Şablonundan Okunması	57
4.6.1 RAM Yapıları	57
4.6.2 FIFO (First In First Out) Buffer	58
4.6.3 İş Hattı Mantığıyla Oluşturulmuş Medyan Filtre Yapısı İçin Piksel Değerlerinin Şablondan Okunması	59
4.7 Medyan Filtre Bloğu	61
4.7.1 VGA Video Sinyalleri	63
4.7.2 Video Sinyalleri Senkronizasyon Bloğu	66

4.7.3	Medyan Filtre Bloğunun Gürültü Bastırmadaki Performansı	68
5.	SONUÇLAR.....	73
KAYNAKLAR.....		77
EKLER		79
Ek 1 İş hattı mantığıyla oluşturulmuş 3x3 kare medyan filtre bloğunun detaylı yapısı.....		80
Ek 2 9-bit karşılaştırıcı düğüm noktasının içyapı şeması		81
Ek 3 Fifo buffer bloğu modelsim simülasyonu		82
Ek 4 Medyan filtre bloğu gecikmesi (720x1280 çözünürlük).....		83
ÖZGEÇMİŞ.....		84

SİMGE LİSTESİ

1-D	Bir boyutlu işaret
2-D	İki boyutlu işaret, görüntü işareti
$C(i, j)$	i. satır ve j. sütunda yer alan hücre.
F_0	Temel açısal frekans bileşeni
f_x	x düzlemindeki uzaysal frekans
f_y	y düzlemindeki uzaysal frekans
$I(x, y)$	İki boyutlu görüntü işareti
i, j	İki boyutlu görüntünün satır ve sütun indisleri.
L	Pikselin gri seviyesi – pikselin değeri – pikselin büyüklüğü
M	Sayısal görüntünün satır sayısı – bir sütunda bulunan piksel sayısı
N	Sayısal görüntünün sütun sayısı – bir satırda bulunan piksel sayısı
$S_r(i, j)$	$C(i, j)$ hücresinin etki alanı (sphere of influence) - (i, j) koordinatındaki piksel büyüklüğünün $(2r+1)$ komşuluğu
x	Dikey uzaysal mesafe
X_0	Temel uzaysal frekans bileşeni
y	Yatay uzaysal mesafe

KISALTMA LİSTESİ

BCD	Binary Coded Decimal (İkili Kodlanmış Ondalık)
DSP	Digital Signal Processor (Sayısal Sinyal İşlemcisi)
DVI	Digital Visual Array (Sayısal Görüntü Dizisi)
FPGA	Field Programmable Gate Array (Alanda Programlanabilen Kapı Dizisi)
FSM	Finite State Machine (Sonlu Durum Makinesi)
H (HIGH)	Lojik üst seviye – ‘1’ olarak tanımlanır
HDL	Description Language (Donanım Tanımlama Dili)
L (LOW)	Lojik alt seviye – ‘0’ olarak tanımlanır
LSB	Least Significant Bit
MSB	Most Significant Bit
MSE	Mean Squared Error (Ortalama Karesel Hata)
PROM	Programmable Read-Only Memory (Programlanabilen Yalnızca Okunabilir Bellek)
PSNR	Peak Signal Noise Ratio (Darbe İşaret Gürültü Oranı)
RAM	Random Access Memory (Rastgele Erişimli Bellek)
ROM	Read-Only Memory (Yalnızca Okunabilen Bellek)
VGA	Video Graphics Array (Video Grafikleri Dizisi)
VHDL	Very High Speed Integrated Circuit Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
VLSI	Very Large Scale Integration (Çok Büyük Ölçekli Tümleştirme)

ŞEKİL LİSTESİ

Şekil 1.1 248x248 darbe gürültülü görüntü.....	1
Şekil 2.1 Sayısal görüntünün gri seviyelerini gösteren $I(x,y)$ fonksiyonu (Toprak, 2006).....	3
Şekil 2.2 Sayısal görüntüde piksellerin numaralandırılışı.....	4
Şekil 2.3a Sürekli görüntü, Görüntünün AB kesiti (Toprak, 2006).....	4
Şekil 2.3b Örnekleme ve nicelendirme, Sayısal tarama çizgisi (Toprak, 2006).....	5
Şekil 2.4 1-boyutlu ve 2-boyutlu dizilerin koordinat sisteminde gösterimi.....	5
Şekil 2.5 128x128 bir piksel genişlikli siyah beyaz çizgiler.....	6
Şekil 2.6 128x128 dört piksel genişlikli siyah beyaz çizgiler.....	6
Şekil 2.7 Sayısal görüntünün matris olarak ifade edilmesi.....	7
Şekil 2.8 190x110 çözünürlüklü gri seviyeli görüntü.....	9
Şekil 2.9 250x170 çözünürlüklü gri seviyeli görüntü.....	9
Şekil 2.10 320x240 çözünürlüklü gri seviyeli görüntü.....	9
Şekil 2.11 380x270 çözünürlüklü gri seviyeli görüntü.....	10
Şekil 2.12 640x480 çözünürlüklü gri seviyeli görüntü.....	10
Şekil 2.14 128x128 çözünürlüklü siyah-beyaz dikey çizgi.....	12
Şekil 2.15 Yatay yönde değişen siyah-beyaz renk değişimi.....	12
Şekil 2.16 1280x720 8-bit gri seviyeli görüntü(*).....	14
Şekil 2.17 640x480 8-bit gri seviyeli görüntü.....	14
Şekil 2.18 320x240 8-bit gri seviyeli görüntü.....	15
Şekil 2.19 100x60 8-bit gri seviyeli görüntü.....	15
Şekil 2.20 1280x720 8-bit gri seviyeli görüntü(*).....	15
Şekil 2.22 Piksel komşuluğu 3x3 (Leon ve Thomas, 2002).....	17
Şekil 2.23 Piksel komşuluğu 5x5 (Leon ve Thomas, 2002).....	17
Şekil 2.24 Piksel komşuluğu 7x7 (Leon ve Thomas, 2002).....	17
Şekil 2.25 3x3 piksel komşuluğu ve komşu piksellerle ilişkileri.....	18
Şekil 2.26 3x3 piksel komşuluğu ve komşu pikseller (Vijaykumar vd., 2009).....	18
Şekil 3.1 240x150 çözünürlükte %10 oranında gürültülü resim.....	20
Şekil 3.2 320x240 çözünürlükte %5 oranında gürültülü resim.....	20
Şekil 3.3 960x540 çözünürlükte %15 oranında gürültülü resim(*).....	21
Şekil 3.4 1280x720 çözünürlükte %25 oranında gürültülü resim(**).....	21
Şekil 3.5 3x3, 5x5, 7x7 medyan filtre şablonları.....	22
Şekil 3.6 Farklı şekillerde şablon tipleri.....	23
Şekil 3.7 460x278 çözünürlüğünde orijinal görüntü.....	24
Şekil 3.8 20% oranında gürültülü görüntü işareti.....	24
Şekil 3.9 3x3 medyan filtre şablonu ile filtrelenmiş görüntü.....	25
Şekil 3.10 5x5 medyan filtre şablonu ile filtrelenmiş görüntü.....	25
Şekil 3.11 7x7 medyan filtre şablonu ile filtrelenmiş görüntü.....	26
Şekil 3.12a Basamak şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001).....	27
Şekil 3.12b Rampa şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001).....	27
Şekil 3.12c Darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001).....	27
Şekil 3.12d Çift darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001).....	27
Şekil 3.12e Üçlü darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001).....	28
Şekil 3.12f Üçgen darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001).....	28
Şekil 3.13 Kare şeklindeki şablonun medyan filtrelemedeki etkisi (Pratt, 2001).....	28
Şekil 3.14 Artı şeklindeki şablonun medyan filtrelemedeki etkisi (Pratt, 2001).....	29
Şekil 3.15a 3x3 kare medyan filtre şablonlarıyla medyan değerinin bulunması.....	29
Şekil 3.15b 3x3 kare medyan filtre şablonlarıyla medyan değerinin bulunması.....	30
Şekil 3.16a Şablonun görüntü üzerindeki hareketi.....	30
Şekil 3.16b Şablonun görüntü üzerindeki hareketi.....	31
Şekil 3.16c Şablonun görüntü üzerindeki hareketi (Dhanasekaran ve Boopathy, 2009).	31

Şekil 4.1 Xilinx virtex-5 serisinden bir fpga	33
Şekil 4.2 3x3 kare medyan filtre şablonu (Smith, 1996)	34
Şekil 4.3 Paralel girişli 3x3 kare medyan filtre blok yapısı (Smith, 1996)	35
Şekil 4.4 Paralel girişli medyan filtre bloğu içyapısı.....	36
Şekil 4.5 Paralel girişli medyan filtre bloğu gecikmesi.....	36
Şekil 4.6 İkili karşılaştırıcı alt blok gösterimi (Vega-Rodriguez vd., 2002)	37
Şekil 4.7 İş hattı mantığıyla oluşturulmuş medyan filtre blok yapısı (Smith, 1996).....	38
Şekil 4.8 İş hattı mantığıyla oluşturulmuş medyan filtre bloğunun gecikmesi	38
Şekil 4.9 İkili karşılaştırıcı alt blok gösterimi	41
Şekil 4.10 74LS85'in lojik sembolü(*)	41
Şekil 4.11 n-bitlik iki sayının karşılaştırılması.....	42
Şekil 4.12 74LS85'in lojik yapısı(*)	42
Şekil 4.13 74LS85 ile 8-bitlik karşılaştırıcı bloğu(*)	43
Şekil 4.14 İki giriş-iki çıkışlı karşılaştırıcı bloğu	44
Şekil 4.15 İki giriş-iki çıkışlı karşılaştırıcı bloğunun içyapısı.....	44
Şekil 4.16 2x2 çoğullayıcı bloğunun içyapısı.....	45
Şekil 4.17 8-bit optimize edilmiş 74LS85 karşılaştırıcısının LSB'si	46
Şekil 4.18 8-bit optimize edilmiş 74LS85 karşılaştırıcısının MSB'si	46
Şekil 4.19 Optimize edilmiş 74LS85 ile elde edilen 8-bit karşılaştırıcı lojik yapısı	47
Şekil 4.20 Optimize edilmiş 74LS85 ile elde edilen 8-bit karşılaştırıcı blok yapısı	47
Şekil 4.21 8-bit optimize edilmiş yapının bloklar halinde gösterimi	48
Şekil 4.22 İki giriş-iki çıkış 8-bit optimize edilmiş karşılaştırıcı bloğu	48
Şekil 4.23 İki giriş-iki çıkış 8-bit optimize edilmiş karşılaştırıcının blok halinde gösterimi ...	49
Şekil 4.24 74LS682'nin lojik diyagramı(*).....	49
Şekil 4.25 74LS682'nin blok gösterimi.....	50
Şekil 4.26 İki giriş-iki çıkış 74LS682'li karşılaştırıcının blok gösterimi.....	50
Şekil 4.27 İki giriş-iki çıkış 74LS682'li karşılaştırıcının iç blok gösterimi	51
Şekil 4.28 9-bit karşılaştırıcının lojik devre yapısı.....	54
Şekil 4.29 9-bit karşılaştırıcının blok gösterimi	55
Şekil 4.30 İki giriş-iki çıkış 9-bit karşılaştırıcının blok gösterimi.....	56
Şekil 4.31 İki giriş-iki çıkış 9-bit karşılaştırıcının iç blok gösterimi.....	56
Şekil 4.32 İki giriş-iki çıkış çoğullayıcı iç blok gösterimi	56
Şekil 4.33 Medyan filtre şablondaki piksel değerlerinin okunması	57
Şekil 4.34 Fifo buffer (Pong, 1959).....	58
Şekil 4.35 Block RAM blok gösterimi	58
Şekil 4.36 Fifo buffer blok gösterimi	59
Şekil 4.37 Fifo buffer blok içyapısı	60
Şekil 4.38 15x15 boyutunda bir görüntü matrisi	61
Şekil 4.39 Medyan filtre test blok diyagramı	62
Şekil 4.40 Medyan filtre bloğu	62
Şekil 4.41 Horizontal sync ve vertical sync sinyalleri (Sanchez, 2010)	63
Şekil 4.42 640x480 çözünürlükte VGA sinyalleri (Sanchez, 2010)	63
Şekil 4.43 Medyan filtre bloğunda kullanılan h_frame ve v_frame sinyalleri.....	64
Şekil 4.44 Görüntü üzerinde h_frame, v_frame sinyallerinin durumları	64
Şekil 4.45 Medyan filtre bloğu	66
Şekil 4.46 Video sinyalleri senkronizasyon bloğu içyapısı.....	66
Şekil 4.47 H_frame_synch sinyalinin oluşumu	67
Şekil 4.48 V_frame_synch sinyalinin oluşumu-1	67
Şekil 4.49 V_frame_synch sinyalinin oluşumu-2	68
Şekil 4.50 Enigma, 720x1280 çözünürlükte görüntü işareti(*).....	70
Şekil 4.51 %10 oranında gürültülü görüntü işareti(**)	70
Şekil 4.52 %10 oranında gürültülü görüntü işaretinin medyan filtreden geçirilmiş hali(*).....	71

Şekil 5.1 Altera firmasına ait stratix IV fpga	74
Şekil 5.2 Stratix IVgx fpga development kiti ve hsmc_dvi video giriş/çıkış kartı.....	75
Şekil 5.3 Gerçeklenecek medyan filtre bloğu test blok diyagramı.....	75

ÇİZELGE LİSTESİ

Çizelge 4.1 1.durum (e-bergi, 2009)	39
Çizelge 4.2 2.durum (e-bergi, 2009)	40
Çizelge 4.3 74LS85'in doğruluk tablosu	43
Çizelge 4.4 8-bit optimize edilmiş karşılaştırıcının doğruluk tablosu.....	48
Çizelge 4.5 74LS682'nin doğruluk tablosu	50
Çizelge 4.6 Davranışsal 8-bit karşılaştırıcı bloğu.....	51
Çizelge 4.7 Karşılaştırıcı lojik kapı sayıları	53
Çizelge 4.8 Çeşitli parametrelerle karşılaştırıcı blokları	53
Çizelge 4.9 9-bit karşılaştırıcının doğruluk tablosu	55
Çizelge 4.10 Orijinal, gürültülü ve filtrelenmiş görüntüler arasında MSE ve PSNR oranları .	72

ÖNSÖZ

Bu tez çalışmasının gerçekleşmesinde emeği geçen, özellikle tez danışmanımın belirlenmesi sırasında karşılaştığım zorluklarda desteğini ve yardımını esirgemeyen, içindeki bitmek bilmeyen enerjisini engin deneyim ve bilgilerini bizlere kazandırmak için uğraşan eşsiz insan, değerli hocam Prof. Dr. Vedat Tavşanoğlu'na şükranlarımı sunmak isterim.

Her yardıma ihtiyaç duyduğumda yanımda olan araştırma görevlisi arkadaşlarım Evren Cesur ve Nerhun Yıldız'a teşekkür ederim.

Ayrıca, tüm eğitim hayatım boyunca maddi, manevi destekleriyle daima yanımda olan annem ve babama, bana verdiği destek ve gösterdiği anlayışıyla sevgili eşime teşekkürü bir borç bilirim.

ÖZET

Görüntü işaretleri, çeşitli kaynaklardan etkilenecek gürültü işaretlerine maruz kalabilir. Bunlar: depolama birimlerinden kaynaklanan gürültüler, görüntü oluşumunu sağlayan donanımlardan (kamera, webcam, dijitalizer, vb...) kaynaklanan gürültüler, alıcı-verici ve kayıt donanımlarından kaynaklanan gürültüler, iletim kanalları (televizyon veya radyo yayınları, uydu hatları, kablolu ağlar, statik çarpıcılar, vb...) gibi altyapı ortamlarından kaynaklanan gürültüler olarak tanımlanabilir. Gürültülü görüntü işaretlerinin kullanıcıya sunulması, farklı amaçlar doğrultusunda değerlendirilebilmesi için içerisinde bulunan gürültü işaretlerinin ortadan kaldırılması gerekmektedir. Örnek olarak tıbbi uygulamalarda doktorun hastasıyla ilgili bir MR görüntüsünde değerlendirme yapabilmesi, otomasyon sistemiyle üretim yapan fabrikada banttan çıkan ürünün örnek ürünle karşılaştırmasını görüntü işleme ile gerçekleştiren sistemin elde ettiği kamera görüntüsünün değerlendirilebilmesi için, görüntünün varsa gürültülü işaretlerinin elemine edilmesi oldukça önemlidir.

İncelenecek ve üzerinde çalışma yapılacak olan gürültü çeşidi, genellikle hava ortamındaki iletim kanallarından, güvenli olmayan sayısal hatlarındaki veri iletimi sırasında paket kayıplarından ya da piksel kayıplarından meydana gelir. Bu gürültü çeşidi literatürde darbe gürültüsü, tuz-biber gürültüsü olarak adlandırılır. Bu tarz gürültü işaretlerini temizlemede kullanılan en etkili filtrelerden bir tanesi standart medyan filtredir.

Medyan filtre, görüntü işlemede, darbe gürültülü işaretlerde kullanılan güçlü bir araçtır. Standart medyan filtre herhangi bir modifikasyona gerek kalmadan filtrelemede oldukça iyi sonuçlar verir. Genellikle tuz-biber gürültülerini bastırmada kullanılır. Filtrelemeyi gerçekleştirirken görüntüde bulunan kenar geçişlerini korur. Gaussian filtreleri gibi keskin kenar geçişlerini düzeltirme gibi görüntü işareti üzerinde herhangi bir etki bırakmaz. Fakat görüntü üzerinde bulunan küçük detayları ortadan kaldırabilir. Bu detayların oranı, kullanılan medyan filtre şablonu büyüklüğüyle de doğru orantılıdır. Medyan filtre şablonu büyürse detaylar da o oranda kaybolur. Bunun önüne geçmek için farklı yapılarıdaki medyan filtreleme teknikleri kullanılır. Üzerinde duracağımız ve çalışmalarımızı gerçekleştireceğimiz filtre çeşidi ise standart medyan filtrelerdir.

İki boyutlu medyan filtrelerde $N \times N$ kare medyan filtre şablonunu kullanmak, görüntüde bulunan her bir piksel değeri için şablondaki $N \times N$ tane piksel değerini sıralamak ve bunlardan ortanca değerini bulup görüntüdeki ilgili piksel değeriyle yer değiştirmek oldukça önemli hesaplama gücü gerektirir. Gerçek zamanlı görüntülerde medyan filtreyi uygulamanın tek yolu, algoritmayı donanımsal olarak gerçekleştirmektir. Programlanabilir lojik, kullanıcının ihtiyaç duyduğu tüm uygulamaları gerçekleştirmeye imkân sağladığından, paralel işlem yapabilme yeteneğiyle, gerçek zamanlı görüntü üzerinde medyan filtre uygulamasını FPGA'ler ile mümkün kılar (Dhanasekaran ve Boopathy, 2009).

Çalışmada medyan filtre uygulaması üzerinde durularak, filtrenin yapısı, uygulanış şekli, görüntü üzerindeki etkileri, avantaj ve dezavantajları gibi birçok noktaya değinilmiş, yapının FPGA üzerinde temel bir donanım algoritması kullanılarak yüksek çözünürlüklü gerçek zamanlı görüntüler üzerinde uygulaması yapılmıştır.

Anahtar Kelimeler: Medyan filtre, darbe gürültüsü, tuz & biber gürültüsü, görüntü işleme, fpga, lineer olmayan, yüksek çözünürlük, gerçek zamanlı görüntü.

ABSTRACT

Image signals may be affected from various noise sources. These noises may be caused by the storage unit, the capture hardware that forms the image (camera, webcam, digitizer, etc.), receiver/transmitter or recording equipment, or the transmission channels (television or radio broadcasts, satellite lines, cable networks, static multipliers, etc.). Consequently, the noise on the image should be eliminated at the user side before processing. For example, in a medical application such as an MR image of a patient, it is important to eliminate the noise as much as possible before the evaluation process.

Scope of this thesis is the elimination of the salt & paper noise that is mostly emerged by packet or pixel losses on unreliable transmission channels. These types of noise are also referred as pulse noise in the literature. Median filters are one of the most effective, efficient and non-destructive filters used to retrieve the original image.

Median filters are powerful instruments used in image processing. The traditional median filtering algorithm, without any modifications gives good results. Median filters are used mainly to remove salt-and pepper noise. They preserve edges in the image, preserve their location and do not affect their steepness, unlike Gaussian filters, but unfortunately may destroy small features in the image. A workaround is to apply center-weighted median filtering instead of a plain median, in exchange of the filter's ability to suppress impulse noise. The required high computational power is also a drawback of various kinds of the median filters (Dhanasekaran ve Boopathy, 2009).

Computing a two-dimensional median for a $N \times N$ window, requires sorting of $N \times N$ elements around each image pixel. After the sorting, each queue element is assigned a value called a rank, specifying its position in the queue as a result of sorting. Therefore, using median filtering in any real-time vision system requires a significant computational power. One way to speed up the computations is to implement the algorithm in hardware, e.g. FPGAs. Moreover, programmable logic creates the possibility to tailor the implementation to the user's needs. The result is the significant speedup over the software implementations by using sequential processors.

In this thesis, a median filter structure is realized on an FPGA. Different structures, advantages, disadvantages and applications of the median filters are also discussed. The primary goal was to realize a high-resolution real-time median filter block on FPGA.

Keywords: Median filter, impulse noise, salt & pepper noise, image processing, fpga, non-linear, high definition, real time video.

1. GİRİŞ

Görüntü işaretleri, çeşitli kaynaklardan etkilenerek gürültü işaretlerine maruz kalabilir. Görüntü işaretleri için olası gürültü kaynakları: Depolama birimleri, görüntü oluşumunu sağlayan donanımlar, iletim kanalları (televizyon veya radyo yayınları, uydu hatları, kablolu ağlar, statik çarpıcılar, vb...), alıcı ve kayıt donanımlarıdır. Farklı gürültü kaynaklardan etkilenen görüntü işaretlerinin kullanıcı tarafından farklı amaçlar doğrultusunda değerlendirilebilmesi için görüntü içerisinde bulunan gürültü işaretlerinin ortadan kaldırılması gerekmektedir. Örneğin tıbbi uygulamalarda doktorun hastasıyla ilgili bir MR görüntüsünde değerlendirme yapabilmesi için o görüntünün varsa gürültülü işaretlerinin elemine edilmesi oldukça önemlidir. Araç tanıma sistemlerinde ve plaka tanıma sistemlerinde de kameradan alınan görüntü işaretlerinde görüntünün temizlenmesi oldukça önemlidir (Dhanasekaran ve Boopathy, 2009).

İnceleyeceğimiz ve üzerinde çalışacağımız gürültü çeşidi genellikle hava ortamındaki iletim kanallarından, güvenli olmayan sayısal iletim hatlarındaki veri iletimi sırasında paket kayıplarından ya da piksel kayıplarından meydana gelir. Bu gürültü çeşidine çeşitli kaynaklarda (kitap, dergi, makale, bildiri ve tez) farklı isimler altında rastlanmaktadır. Bunlar: Tuz-biber gürültüsü (salt & pepper noise), benek gürültüsü (speckle noise), darbe gürültüsüdür (impulse or shot noise). Tuz beyaz, biber ise siyah gürültü piksellerini temsil etmektedir.



Şekil 1.1 248x248 darbe gürültülü görüntü.

Görüntü üzerinde bu tür gürültü işaretlerini temizlemede en çok kullanılan filtre medyan filtredir. Doğrusal olmayan bu filtre, darbe gürültülü işaretlerde doğrusal filtrelere göre daha iyi sonuç vermektedir. Medyan filtrenin farklı çeşitleri bulunmaktadır (Brownrigg, 1984).

Bunlardan bazıları:

- Standart medyan filtre,
- Adaptif medyan filtre (Adaptive median filter),
- Ağırlıklı medyan filtredir (Weighted median filter).

Uygulamada gerçek zamanlı yüksek çözünürlüklü (720p) görüntüler üzerinde standart medyan filtre uygulaması ele alınmıştır. Farklı tipte standart medyan filtre blokları üzerinde çalışmalar yapılmış, filtrelerin yapısı, uygulanış şekli, görüntü üzerindeki etkileri, artıları ve eksileri gibi birçok noktaya değinilmiştir Uygun olan filtre blokları FPGA platformu üzerinde gerçekleştirilerek monitör üzerinden gözlenmiştir.

2. SAYISAL GÖRÜNTÜ VE ÖZELLİKLERİ

Bu bölümde sayısal görüntünün tanımı, sayısal görüntünün matrissel tanımı, uzaysal çözünürlük ve pikselin gri seviyesi ve pikseller arasındaki temel ilişkiler konu olarak ele alınıp incelenmiştir.

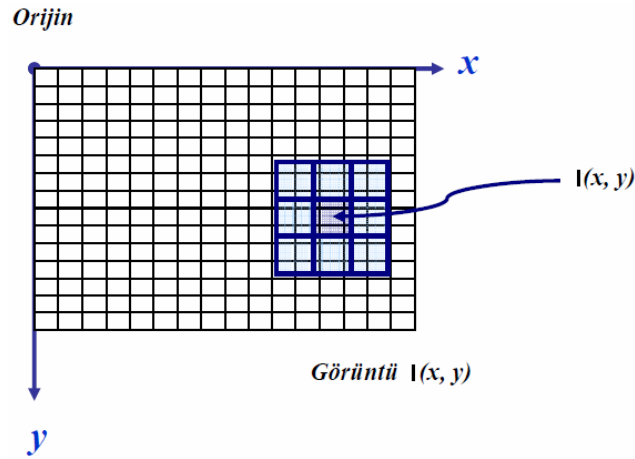
2.1 Sayısal Görüntünün Tanımı

Görüntü iki boyutlu $I(x,y)$ işareti olarak tanımlanır. Burada x ve y koordinatlar, I ise o koordinattaki noktanın gri seviyesini gösteren bir fonksiyondur.

I fonksiyonunun iki değişkeni vardır:

$$I(x,y) \quad (2.1)$$

Burada x ve y sırasıyla dikey ve yatay uzaysal mesafelerdir (vertical and horizontal spatial distances).

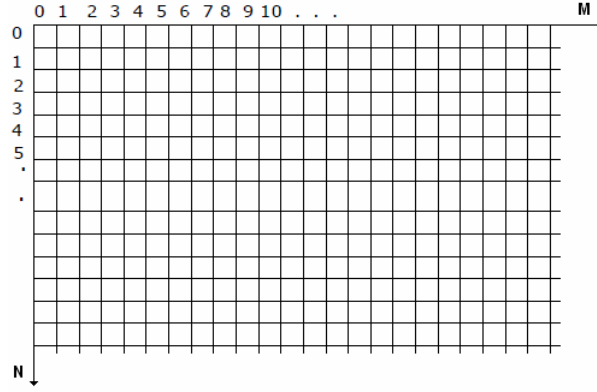


Şekil 2.1 Sayısal görüntünün gri seviyelerini gösteren $I(x,y)$ fonksiyonu (Toprak, 2006)

Bir siyah-beyaz görüntü, siyah ve beyaz renk tonları arasında gri rengi renk seviyelerinden oluşur. Buna sürekli tonlu resim (continuous tone image) adı verilir.

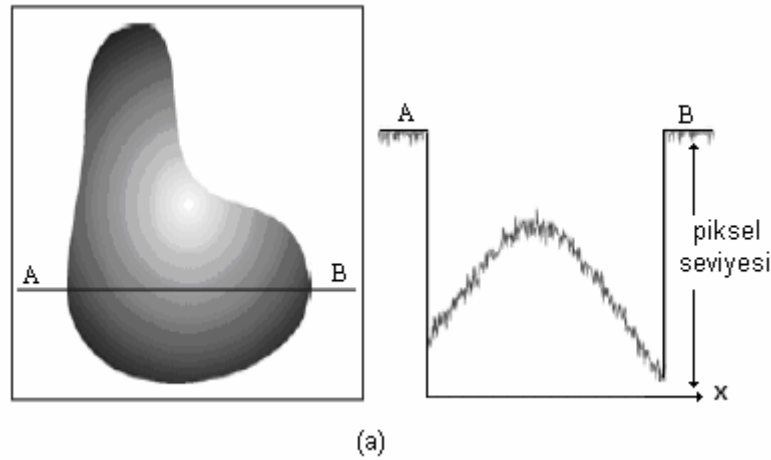
Sayısal görüntü işlemede görüntü ayrık olmalıdır. Bu sebeple, sürekli tonlu görüntünün parçalanarak ayrık veriler haline getirilmesi gerekir. Bu işleme örnekleme (sampling) adı verilir. Burada örneklenecek olan boyutlar, uzayın sürekli bir fonksiyonu olan parlaklık

seviyeleridir. Örneklemenin sonucunda görüntü kare ızgaralara ayrıştırılır ve bu ayrıştırma sonucunda oluşan her bir ızgaradaki elemana görüntü elemanı adı verilir. Ayrıştırılmış bu görüntü elemanları da bulundukları yeri belirten bir çift sayıyla etiketlenir. Bu koordinatın ilk elemanı görüntünün satır numarasını, ikincisi ise sütun numarasını gösterir.

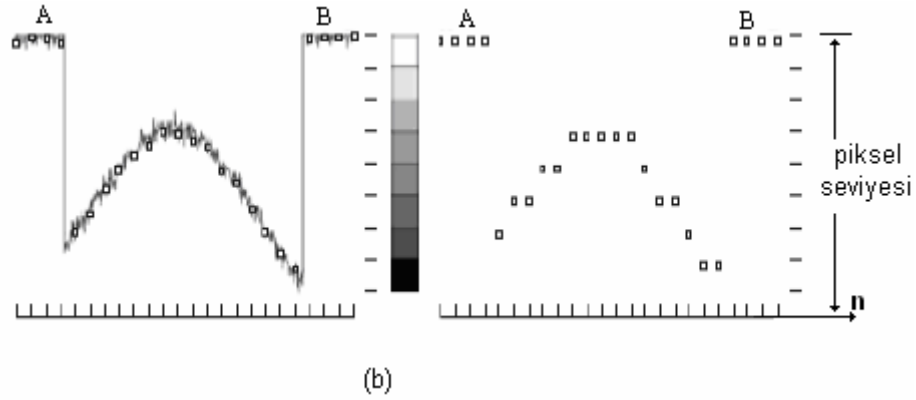


Şekil 2.2 Sayısal görüntüde piksellerin numaralandırılışı

Sözü edilen görüntü elemanı piksel olarak tanımlanır. Tezin ilerleyen bölümlerinde görüntü elemanı yerine piksel kullanılmıştır. Ayrıştırma işlemi görüntünün parlaklık seviyeleri üzerinde gerçekleştirilir. Siyah-beyaz görüntüde sürekli tonlu gri seviyeler ayrıştırılır. Genlikteki ayrıştırma işlemine nicelendirme (quantization), koordinat değerlerinin ayrıştırma işlemine ise örnekleme denir. 8-bitlik sistemde, belirtilen gri renk tonu yelpazesi $2^8 = 256$ farklı gri renk seviyesine sahiptir.



Şekil 2.3a Sürekli görüntü, Görüntünün AB kesiti (Toprak, 2006)

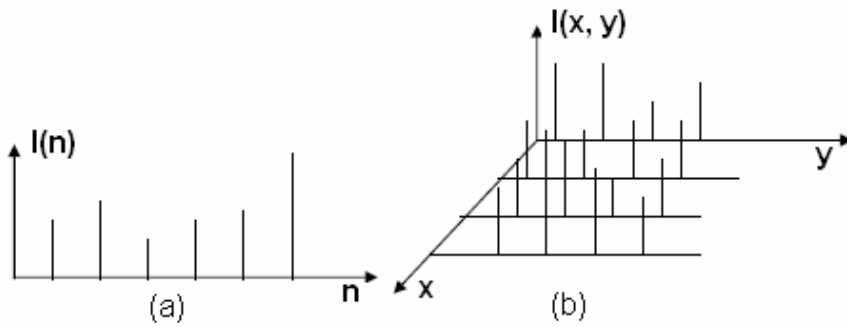


Şekil 2.3b Örnekleme ve nicelendirme, Sayısal tarama çizgisi (Toprak, 2006)

Şekil 2.3'te sürekli 2-boyutlu görüntü ve onun bir kesitinden elde edilen 1-boyutlu işaret, örneklenmiş ve nicelendirilerek gri seviyeleri verilmiştir.

Tanımlanmış olan I fonksiyonu 0-255 arasındaki gri değerleri alabilen ayrık bir fonksiyondur. Bir ayrık eleman dizisi tek boyutlu bir matris şeklinde ifade edilirken, ayrık görüntü işareti ise iki boyutlu ve (x, y) koordinatında belli bir gri seviyesi olan fonksiyonun matrisi şeklinde ifade edilebilir. Bu durum, Şekil 2.4'teki grafiklerde gösterilmiştir (Toprak, 2006).

Fiziksel işaretin frekans içeriği gözlem uzunluğu boyunca görünen hızlı değişimlere bağlıdır. Sayısal işaret işlemede bu uzunluk kayıt uzunluğu (record length) olarak tanımlanır. İki boyutlu görüntüde tanımlanan uzaysal frekans bileşenleri iki adettir. Bunlar f_x , f_y olarak adlandırılır. Sayısal görüntüde, frekans bileşenlerine uzaysal frekans (spatial frequencies) adı verilir.

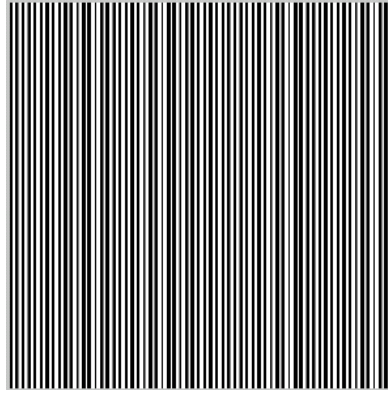


Şekil 2.4 1-boyutlu ve 2-boyutlu dizilerin koordinat sisteminde gösterimi

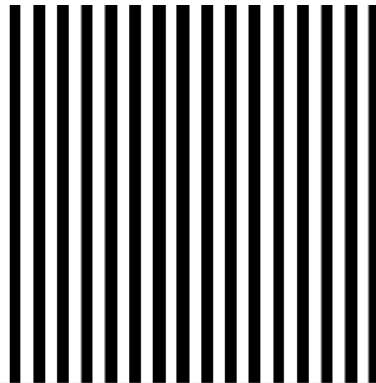
Bir görüntünün sayısal ortama aktarılmasıyla elde edilen sayısal görüntünün orijinal görüntüyü ne kadar benzerlikle temsil ettiği her zaman sorgulanır. Örneklenerek ayrıklaştırılan sayısal görüntü, orijinal görüntüden ne kadar farklıdır, ya da ne kadar oranda orijinal görüntünün birebir aynısıdır.

Bu noktada sayısal işlemin en başından sınırları belirlemek gerekmektedir. Bu sınırlar ise çözünürlük (resolution) -birim alandaki piksel sayısı- olarak adlandırılır. Görüntünün birim alanındaki piksel sayısı fazla ise detaylar o oranda fazla, birim alandaki piksel sayısı az ise detaylar o oranda az olur. Bu noktadan hareketle “görüntü kalitesi çözünürlükle doğru orantılıdır” önermesi doğrudur.

Şekil 2.5’deki 128x128 resimde her bir dikey siyah ve beyaz çizgiler birer piksel genişliğindedir. Şekil 2.6’daki 128x128 resimdeki her bir siyah ve beyaz çizgiler ise 4 piksel ile temsil edilmektedir.



Şekil 2.5 128x128 bir piksel genişlikli siyah beyaz çizgiler



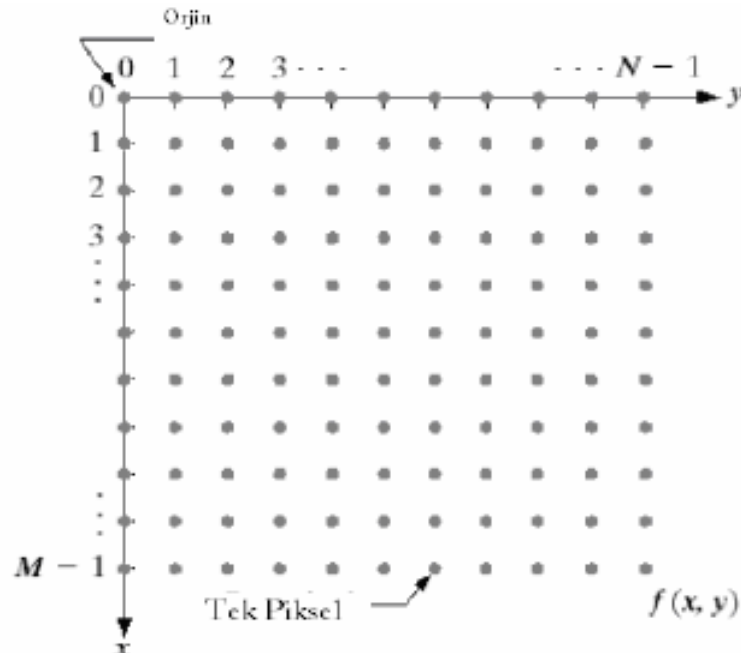
Şekil 2.6 128x128 dört piksel genişlikli siyah beyaz çizgiler

Her bir siyah beyaz çizgi çifti yatay yönde bir periyodu temsil eder. Şekil 2.5'deki görüntüde 64 adet periyot, Şekil 2.6'daki görüntüde ise 16 tane periyot bulunmaktadır. Bunun anlamı gri seviyelerindeki uzaysal frekans değişimi Şekil 2.5'deki görüntünün Şekil 2.6'daki görüntüye göre 4 kez daha fazladır.

Özetlemek gerekirse, görüntü işaretimizin temel frekans bileşenleri alçaktan yükseğe doğru sıralanır. Bir sayısal görüntüde eğer hızlı şekilde parlaklık geçişleri varsa uzaysal frekans yüksek, yavaş parlaklık geçişleri söz konusu ise bu durumda uzaysal frekansımızın değeri düşük olur. Eğer görüntüde bir noktada keskin bir siyah-beyaz geçişi bir piksel ile sağlanıyorsa o noktada yüksek uzaysal frekans vardır denir.

2.2 Sayısal Görüntünün Matrisel Gösterimi

Örnekleme ve nicelendirmenin sonucu elde edilen sayısal görüntü bir matris ile temsil edilebilir. Görüntü üzerinde gerçekleştirilecek işlemlerin kolaylıkla yapılması için görüntünün matris gösteriminden faydalanılır. Görüntü işaretinin örneklenmesiyle elde edilen $I(x,y)$ matrisinde satır ve sütun sayıları sırasıyla M , N olarak adlandırılır. Elde edilen bu görüntü matrisinde her bir (x, y) koordinat değeri ayrıktır. Sayısal görüntü üzerinde (x, y) koordinatının gösterdiği her bir nokta ayrıklaştırılmış piksel seviyelerine karşılık gelir. İşlemlerde kolaylık sağlaması ve çalışılan görüntü işleme programlarındaki notasyondan dolayı bu çalışmada görüntü matrisinin orijin noktası $(x, y) = (0, 0)$ kabul edilir (Şekil 2.7).



Şekil 2.7 Sayısal görüntünün matris olarak ifade edilmesi

Bir önceki paragrafta yapılan tanımlama gereği, $M \times N$ şeklinde sayısal görüntü (2.2) eşitliğindeki gibidir.

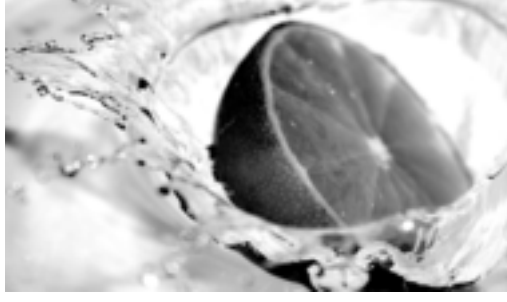
$$A = I(x, y) = \begin{bmatrix} I(0,0) & I(0,1) & \dots & \dots & \dots & I(0,N-1) \\ I(1,0) & I(1,1) & \dots & \dots & \dots & I(1,N-1) \\ & & & & & \\ & & & & & \\ & & & & & \\ I(M-1,0) & I(M-1,1) & \dots & \dots & \dots & I(M-1,N-1) \end{bmatrix} \quad (2.2)$$

(2.2) eşitliğinde sağ taraf tanım itibari ile sayısallaştırılmış görüntüdür. Bu matris dizisinin her bir elemanına sayısal görüntünün elemanı, resim elemanı veya piksel denir. Bazı kaynaklarda (2.2)'deki eşitliği (2.3) eşitliğindeki gösterildiği gibi bir ifadeyle gösterilebilir.

$$A = \begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} & \dots & \dots & \dots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & a_{1,2} & \dots & \dots & \dots & a_{1,N-1} \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ a_{M-1,0} & a_{M-1,1} & a_{M-1,2} & \dots & \dots & \dots & a_{M-1,N-1} \end{bmatrix} \quad (2.3)$$

Görüntü ile ilgili yapacağımız tüm işlemlerde, simülasyonlar ve sentezlemelerde görüntünün matrissel formundan yararlanılır.

M ve N değerleri, sayısal hale getirilen matrisin satır ve sütun sayılarını gösterir. Bu değerler ile sayısal görüntünün kaç tane pikselden oluştuğu hesaplanır. Her bir (x, y) değeri ilgili pikselin matris içinde konumunu belirler. Konum bilgisinin yanında bir de renk seviyesinin bilinmesi gerekmektedir.



Şekil 2.8 190x110 çözünürlüklü gri seviyeli görüntü



Şekil 2.9 250x170 çözünürlüklü gri seviyeli görüntü



Şekil 2.10 320x240 çözünürlüklü gri seviyeli görüntü



Şekil 2.11 380x270 çözünürlüklü gri seviyeli görüntü



Şekil 2.12 640x480 çözünürlüklü gri seviyeli görüntü



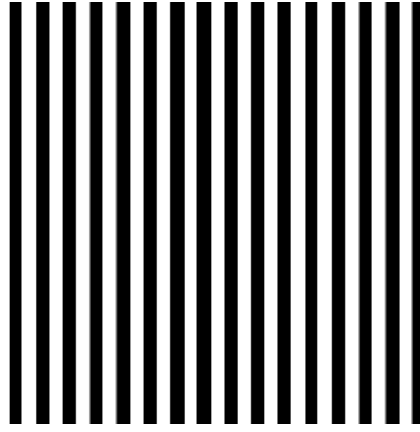
Şekil 2.13 1280x720 çözünürlüklü gri seviyeli görüntü(*)

2.3 Uzaysal Çözünürlük ve Pikselin Gri Seviyesi

Uzaysal çözünürlük bir görüntüdeki fark edilebilen en küçük detay olarak adlandırılır. Bir imajın uzaysal çözünürlüğünü belirlemedeki temel faktör örneklemedir. Örnekleme ile elde edilen en küçük görüntü elemanı (piksel) sayısal görüntünün yapı taşı olarak adlandırılır. Sayısal görüntüyü yan yana dizilen pikseller oluşturur. Piksellerin en boy değerleri yoktur. Kendi başlarına ölçüleri belli olmayan, ait oldukları görüntü içinde boyutlarını kazanan pikseller sahip oldukları bu özellikle görüntünün çözünürlük kalitesinde etkilidir.

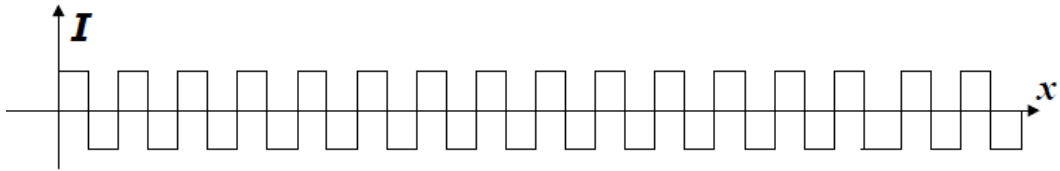
Şekil 2.14'teki dikey siyah ve beyaz çizgilerin genişliği W 'dir. Bu durumda siyah ve beyaz her bir çizgi çifti genişliği $2W$ 'dir. Bunun anlamı bir birim alana $1/2W$ çizgi çifti düşmektedir. Çözünürlüğün yaygın olarak kullanılan bir tanımı ise basitçe şöyledir. Mesafe birimi başına fark edilebilen çizgi çiftidir. Örneğin milimetre başına 100 çizgi gibi.

* 1280x720 çözünürlüğünde olan Şekil 2.13 gri seviyeli görüntü A4 sayfa boyutundan büyük olduğundan boyutu 70% oranında küçültülmüştür.



Şekil 2.14 128x128 çözünürlüklü siyah-beyaz dikey çizgi

Şekil 2.14'deki her bir siyah veya beyaz çizgi genişliği 4 pikseldir. Görüntünün yatay yöndeki siyah beyaz değişimini Şekil 2.15'deki gibi tek boyutta incelenebilir.



Şekil 2.15 Yatay yönde değişen siyah-beyaz renk değişimi

$$X_0 \frac{\text{units}}{\text{cycle}} = \frac{\text{birim}}{\text{devir}} : \text{Temel uzaysal frekans bileşeni} \quad (2.4)$$

$$F_0 = \frac{1}{X_0} \frac{\text{devir}}{\text{birim}} \quad \Omega_0 = \frac{2\pi \text{ radyan/devir}}{X_0 \text{ birim/devir}} = \frac{2\pi \text{ radyan}}{X_0 \text{ birim}} \quad (2.5)$$

Görüntü boyutunun 0.128m x 0.128m olduğu kabul edilsin.

$$X_0 = \frac{0.128}{256} = 0.0005 \text{ m/devir} \quad (2.6)$$

$$F_0 = \frac{1}{X_0} = 2048 \text{ devir/m} \quad (2.7)$$

Özetlersek, y yönünde herhangi bir değişim olmadığından $f_y = 0$ 'dır. Sadece x yönünde değişim söz konusudur.

$$F_x = F_0 \quad (2.8)$$

Sayısal hale getirme işlemi sonucunda görüntü piksel büyüklükleri $[0 - L-1]$ aralığında tam sayılara dönüştürülür. Piksel büyüklüklerinin yayıldığı değer aralığına sayısal görüntünün dinamik aralığı (kontrast) denir. Görüntüde siyah renk tonu ile beyaz renk tonu arasındaki piksel değerlerinin değer aralığı olarak ta adlandırılır. Siyah renk ile beyaz renk arasındaki gri seviye sayısı ne kadar fazla ise görüntünün kontrastı yüksektir, denir. Bunun aksine siyah renk ile beyaz renk arasındaki gri seviye sayısı ne kadar az ise, görüntü düşük kontrastlı olarak adlandırılır.

Görüntü işleme süreçlerinde işlem yapabilmek ve işlenmiş görüntüleri saklamak için görüntülerin gri seviyelerini temsil edecek olan sayılar genellikle iki ve ikinin katları 2^k şeklinde ifade edilir.

$$L = 2^k. \quad (2.9)$$

Yüksek dinamik aralığındaki görüntülerin k değeri artacağından bu değerle işlem yapan sistemlerde bit sayısı da artar. Çalışmada dinamik aralığı $[0 - 255]$ olan gri seviyeli (grayscale) görüntüler üzerinde işlem yapılmıştır.

Benzer şekilde gri seviye çözünürlüğü (kontrast) ise gri seviyedeki fark edilebilen en küçük değişikliktir. Literatürde kontrast değerleri farklı oldukça fazla görüntü formatı yer almaktadır. Bu tez çalışmasında $2^8=256$ farklı gri seviye çözünürlüğünü kullanılacaktır.

Aşağıda aynı görüntü birbirinden farklı çözünürlüklerde görülmektedir. Görüntülerin çözünürlükleri değiştikçe üzerlerindeki detayların durumları da değişmektedir.



Şekil 2.16 1280x720 8-bit gri seviyeli görüntü(*)



Şekil 2.17 640x480 8-bit gri seviyeli görüntü

* 1280x720 çözünürlüğünde olan Şekil 2.16 gri seviyeli görüntü A4 sayfa boyutundan büyük olduğundan boyutu 75% oranında küçültülmüştür.



Şekil 2.18 320x240 8-bit gri seviyeli görüntü



Şekil 2.19 100x60 8-bit gri seviyeli görüntü



Şekil 2.20 1280x720 8-bit gri seviyeli görüntü(*)

* 1280x720 çözünürlüğünde olan Şekil 2.21 gri seviyeli görüntü boyutu 75% oranında küçültülmüştür.

Şekil 2.16, Şekil 2.17, Şekil 2.18, Şekil 2.19 aynı görüntü farklı çözünürlüklerde yer almaktadır. Görüldüğü gibi piksel sayısı azaldıkça resmin boyutu küçülmekte ve ayrıntılar giderek kaybolmaktadır. Benzer şekilde elde edilen düşük çözünürlükteki görüntünün (Şekil 2.19) büyütülmesi durumunda da ilk görüntünün aynısı elde edilmez (Şekil 2.20). Boyut olarak aynı olan bu iki görüntünün birbirinden farkı sahip oldukları piksel sayılarıdır. Piksel sayısı fazla olan Şekil 2.16, Şekil 2.20'ye göre daha fazla detaya sahiptir.

Bu durum görüntünün analog formdan sayısal forma dönüştürülmesi sırasında örneklemeğin görüntü kalitesi oluşumunda önemini göstermektedir. Oluşturulacak sayısal görüntü örneklemeğin artmasıyla orijinal analog görüntüye o oranda benzeyecektir.

2.4 Pikseller Arasındaki Temel İlişkiler

Bu bölümde sayısal görüntü içerisinde bulunan pikseller arasındaki ilişkiler ele alınıp açıklanmıştır. Önceden belirtildiği gibi bir sayısal görüntü $I(x,y)$ ile ifade edilirken üzerinde işlem yapılmış olan piksel p veya q gibi harflerle temsil edilir.

2.4.1 Piksel Komşuluğu

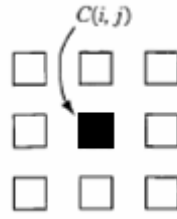
(x,y) koordinatlarında bulunan bir pikselin $(x+1, y)$, $(x-1, y)$, $(x, y+1)$, $(x, y-1)$ koordinatları ile belirlenen dört adet yatay ve dikey komşusu bulunmaktadır. Bu komşu piksellerin yanında (x,y) pikseline komşu koordinatlarıyla verilen $(x+1, y+1)$, $(x+1, y-1)$, $(x-1, y+1)$, $(x-1, y-1)$ dört tane de köşegen pikselleri yer alır. Bu yapılan tanım $r = 1$ için özel bir tanımdır. Daha genel bir ifadeyle piksel komşuluğu:

$$S_r(i,j) = \{C(k,l) \mid 1 \leq k \leq M, 1 \leq l \leq N \text{ max } \{|k-i|, |l-j|\} \leq r\} \quad (2.10)$$

$$S_r(i,j) : (i,j) \text{ koordinatındaki pikselin } (2r+1) \text{ komşuluğu} \quad (2.11)$$

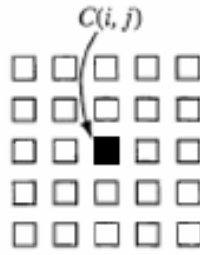
şeklinde tanımlanır (Leon ve Thomas, 2002).

$r=1$ için pikselin (3x3) komşuluğundan söz edilebilir. Yukarıda yapılan tanım da bu koşulda geçerliliğini korumaktadır.

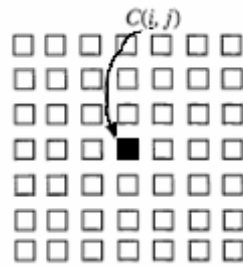


Şekil 2.22 Piksel komşuluğu 3x3 (Leon ve Thomas, 2002)

$r=2$ için, pikselin 5x5 komşuluğundan ve $r=3$ için, pikselin (7x7) komşuluğundan söz edilebilir.



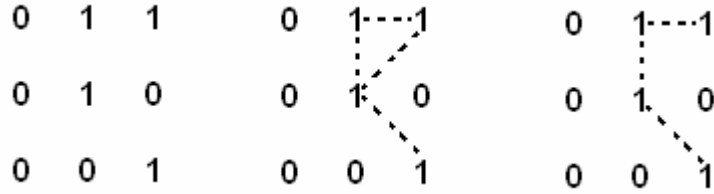
Şekil 2.23 Piksel komşuluğu 5x5 (Leon ve Thomas, 2002)



Şekil 2.24 Piksel komşuluğu 7x7 (Leon ve Thomas, 2002)

Tez boyunca üzerinde durulacak olan komşuluk 3x3 piksel komşuluğu olacağından lineer olmayan filtreleme teknikleri incelenirken kullanılan şablonlar ve sentezlenen devreler hep

3x3'lük piksel komşuluğu temel alınarak gerçekleştirilmiştir. 5x5 ve 7x7 piksel komşuluklarıyla da işlem yapılarak birbirlerine göre avantajlı veya dezavantajlı olduğu noktalar belirtilecektir.



Şekil 2.25 3x3 piksel komşuluğu ve komşu piksellerle ilişkileri

$X(i-1,j-1)$	$X(i-1,j)$	$X(i-1,j+1)$
$X(i,j-1)$	$X(i,j)$	$X(i,j+1)$
$X(i+1,j-1)$	$X(i+1,j)$	$X(i+1,j+1)$

Şekil 2.26 3x3 piksel komşuluğu ve komşu pikseller (Vijaykumar vd., 2009)

Piksel komşulukları aynı zamanda gürültünün bulaşmış olduğu piksellerin tespitinde de önemli rol oynar. Çoğu zaman kenar pikselleri ile gürültü bulaşmış pikseller birbirlerine karışabilir. Bunun önüne geçmek için sürekliliği olan piksel farklılıkları daha çok kenarları belirlerken, süreksiz ve rastgele dağılmış farklı pikseller de gürültü olarak kabul edilir (Vijaykumar vd., 2009).

Hangi pikselin gerçekten kenar pikseli, hangisinin gürültü sonucu oluşmuş pikseller olduğunu çeşitli algoritmalar kullanarak anlamak mümkündür.

Eğer merkez piksel ile komşu pikseller arasında anlamlı bir gri seviye farklılığı yoksa ele alınan 3x3'lük matristeki merkez pikselin gürültü olma olasılığı düşer. Ancak anlamlı bir fark olması durumunda gürültü olma olasılığı üzerinde durularak çeşitli gürültü temizleme yöntemleri üzerinde durulabilir.

3. GERÇEK ZAMANLI GÖRÜNTÜ TEMİZLEME

Görüntü iyileştirme, görüntü üzerinde istenen bölgelerin ön plana çıkarılması, istenmeyen alanların ise yok edilmesi için gerçekleştirilen uygulamalar bütünüdür. Bu uygulamalar bazen şeklin bütününe daha çok bozarken, ilgilenilen kısmı daha görünür hale getirebilir. Görüntü iyileştirme tekniklerinin hiçbiri görüntüde olmayan yeni bir bilgiyi elde etmez, sadece mevcut görüntü bilgilerinin değişik işlemlerle daha kullanışlı hale getirilmesini sağlar. Görüntü iyileştirme, eldeki bir görüntünün belli bir amaca uygun şekilde daha iyi anlaşılır hale getirilmesidir.

Ele alınmış ve incelenmiş olan görüntü çeşitleri daha çok gerçek zamanlı gürültülü görüntü işaretleridir. Çalışmada ağırlıklı olarak tuz-biber gürültülü gerçek zamanlı görüntüler incelenecektir. Bu tip gürültülü işaretler üzerinde birçok lineer ve lineer olmayan temizleme teknikleri uygulanabilmektedir. Bu uygulamalardan etkili olan, görüntüye uygulandığında iyi sonuç veren filtre çeşitleri mevcuttur. Her bir tekniğin uygulamada birbirlerine göre artıları ve eksileri vardır. Yukarıdaki paragrafta da bahsedildiği gibi amaca uygun teknik seçilerek görüntü iyileştirilebilir.

Çalışmada medyan filtre uygulaması üzerinde durularak, filtrenin yapısı, uygulanış şekli, görüntü üzerindeki etkileri, avantaj ve dezavantajları gibi noktalara değinilip, yapının FPGA üzerinde uygun bir algoritma kullanılarak yüksek çözünürlüklü gerçek zamanlı görüntüler üzerinde uygulaması yapılmıştır.

3.1 Medyan Filtre

Medyan isminden de anlaşılacağı gibi ortanca veya ortadaki anlamına gelmektedir. Medyan Filtre, doğrusal olmayan uzaysal bir filtredir. Genel olarak görüntü temizlemede kullanılır. En etkili olduğu gürültü çeşidi tuz-biber gürültüsü (*salt & pepper noise*)'dür. Tuz-biber gürültülü görüntüler üzerinde medyan filtrenin yaptığı iş kadar etkili olan, onun kadar basit uygulaması olan başka bir filtreleme tekniği yoktur.

3.1.1 Medyan Filtrenin Özellikleri

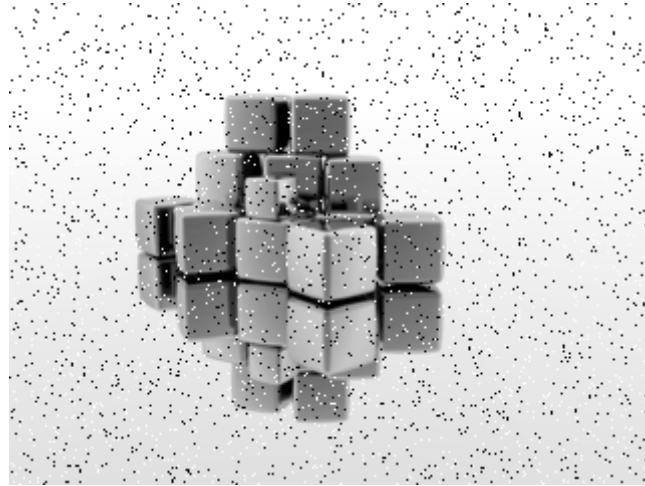
- Lineer olmayan bir filtredir.
- Darbe gürültülü (shot noise) görüntülerde tercih edilir.

- Bir ve iki boyutlu işaretlere uygulanabilir.
- Filtreleme işleminde farklı büyüklük ve şekillerde şablonlar kullanılabilir.
- Filtre, gürültü yoğunluğu görüntüyü oluşturan piksel sayısının %20'sini geçmeyen durumlarda oldukça etkilidir. Diğer bir deyişle kullanılan filtre şablonunda bulunan gürültülü piksel sayısı, şablonun eleman sayısının yarısından fazla ise kullanılacak olan bu medyan filtre çoğunlukla iyi sonuç vermez (Acharya ve Ray, 2005).
- Detayları fazla olan görüntülerde ayrıntıların kaybolmasına neden olabilir.*

Aşağıda çeşitli çözünürlükte, farklı oranlarda gürültülü görüntü işaretleri yer almaktadır.



Şekil 3.1 240x150 çözünürlükte %10 oranında gürültülü resim

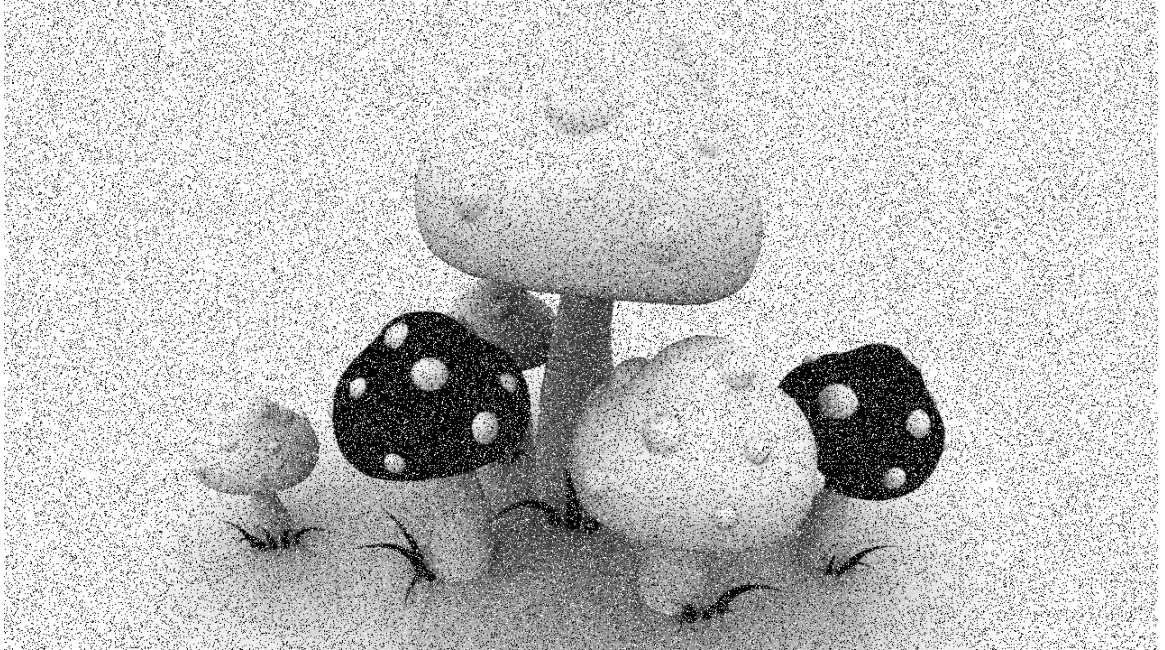


Şekil 3.2 320x240 çözünürlükte %5 oranında gürültülü resim

* Filtrelemede kullanılacak olan filtre şablonunun şekline ve büyüklüğüne bağlıdır. Bir görüntüde birden fazla aynı filtreleme yapılırsa yine detaylar yok olur.



Şekil 3.3 960x540 çözünürlükte %15 oranında gürültülü resim(*)



Şekil 3.4 1280x720 çözünürlükte %25 oranında gürültülü resim(**)

* 960x540 çözünürlüğünde olan Şekil 3.3 gri seviyeli görüntü A4 sayfa boyutundan büyük olduğundan boyutu 45% oranında küçültülmüştür.

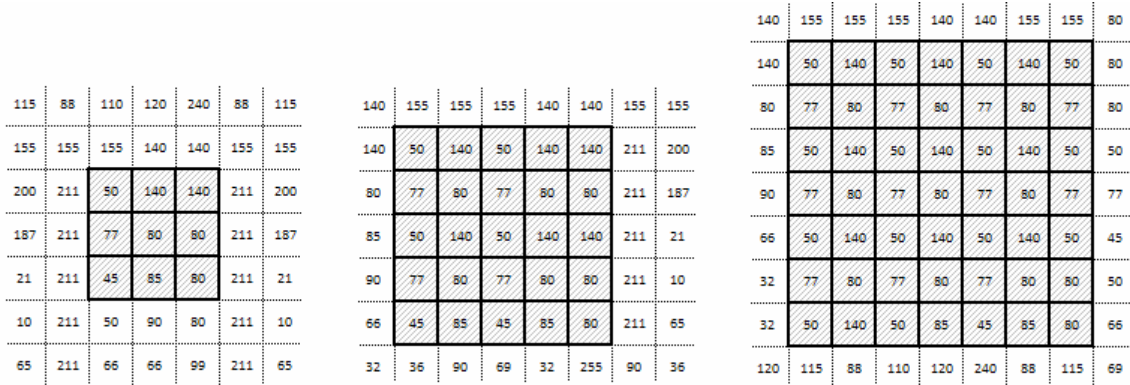
** 1280x720 çözünürlüğünde olan Şekil 3.4 gri seviyeli görüntü A4 sayfa boyutundan büyük olduğundan boyutu 55% oranında küçültülmüştür.

3.1.2 Medyan Filtre Yapısı ve Kullanılan Şablonlar

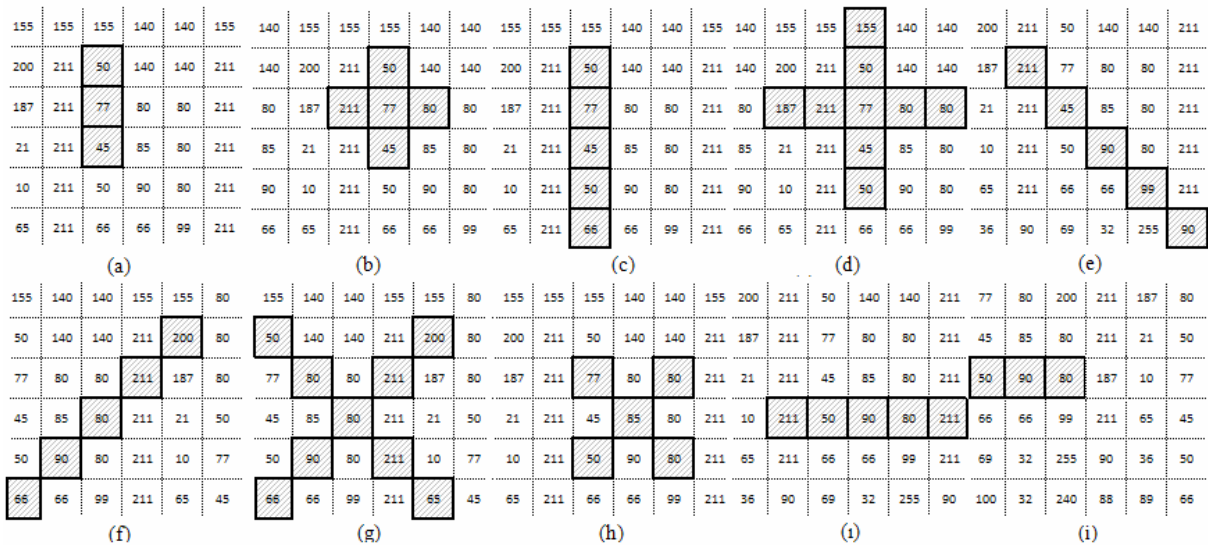
Görüntü işaretleri üzerine uygulanacak medyan filtre şablonunun seçimi oldukça önemlidir. Temizlenecek görüntüde ön plana çıkarılacak olan özelliğe göre medyan şablonu seçilmelidir. Literatürde bu alanda kullanılan çok farklı medyan filtre şablonları bulunmaktadır. Bu çalışmada kullanılmış medyan filtre şablonu 3x3 boyutunda karedir. Aşağıda medyan filtrelemede kullanılan bazı şablon türlerine örnekler verilmiştir.

Sık kullanılan medyan filtre şablonları:

- 3x3 kare şablonu
- 5x5 kare şablonu
- 7x7 kare şablonu
- Çeşitli geometrik şekillerde şablonlar



Şekil 3.5 3x3, 5x5, 7x7 medyan filtre şablonları



Şekil 3.6 Farklı şekillerde şablon tipleri

Şekil 3.6’da çeşitli boyut ve şekillerde medyan filtre şablonları görülmektedir. Bu şablonlardan her biri darbe gürültüsünü bastırmada kullanılır. Her biri filtrelemede kullanıldığında uygulanan görüntü işareti üzerinde farklı etkiler bırakabilir.

(1) ve (i) şekillerinde bulunan şablonlar, görüntü üzerinde yatay eksenle etkilidir. Dikey eksenle görüntüde var olan dik çizgi veya sınırı gürültü olarak kabul edeceğinden bu şablonların uygulandığı görüntü işaretlerinde var ise dikey çizgiler veya sınırlar kaybolur. Bu tip şablonlar medyan filtrelemede özel uygulamalar için kullanılabilir. Benzer şekilde (a) ve (c) şablonları da yatay doğrultuda bulunan siyah veya beyaz sınırları, doğruları gürültü sanacağından yok eder. (b), (d), (g), (e) şablonları diğer şablonlara göre daha iyi sonuçlar verebilir. Bu şablonların hepsinin bir arada kullanılacağı gürültü temizleme uygulamaları da olabilir.

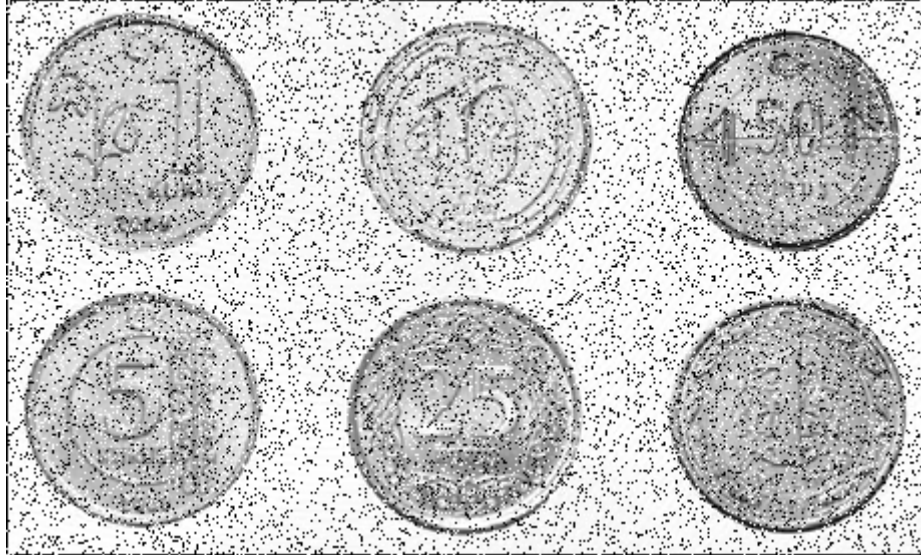
Bu noktada dikkat edilmesi gereken önemli nokta, gürültü temizleme uygulamalarında amaca yönelik uygun filtre şablonunun seçilmesidir.(*)

* Bir boyutlu işaretlerde kullanılan medyan filtre şalonu beş uzunlukludur. Filtrelemede kullanılacak olan filtre şablonunun şekline ve büyüklüğüne bağlıdır. Bir görüntüde birden fazla aynı filtreleme yapılırsa yine detaylar yok olur.

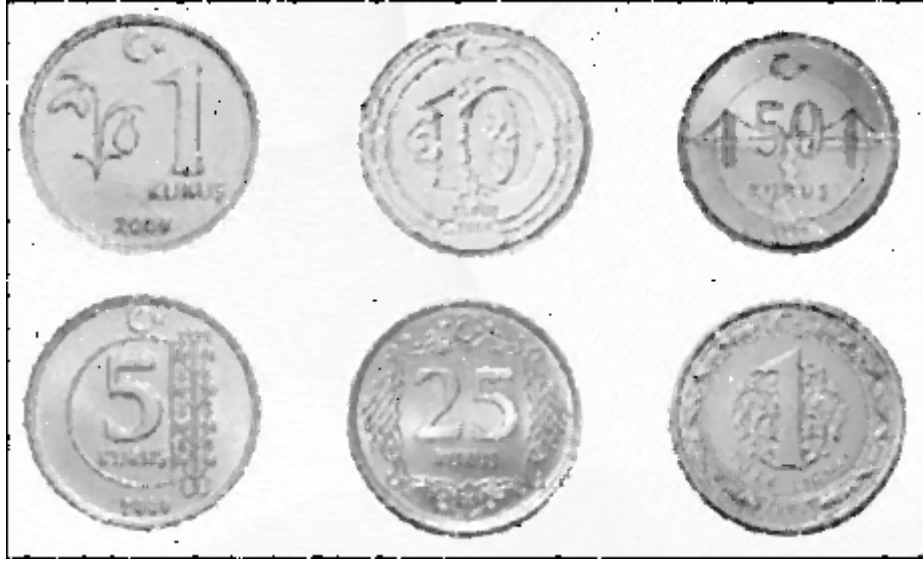


Şekil 3.7 460x278 çözünürlüğünde orijinal görüntü

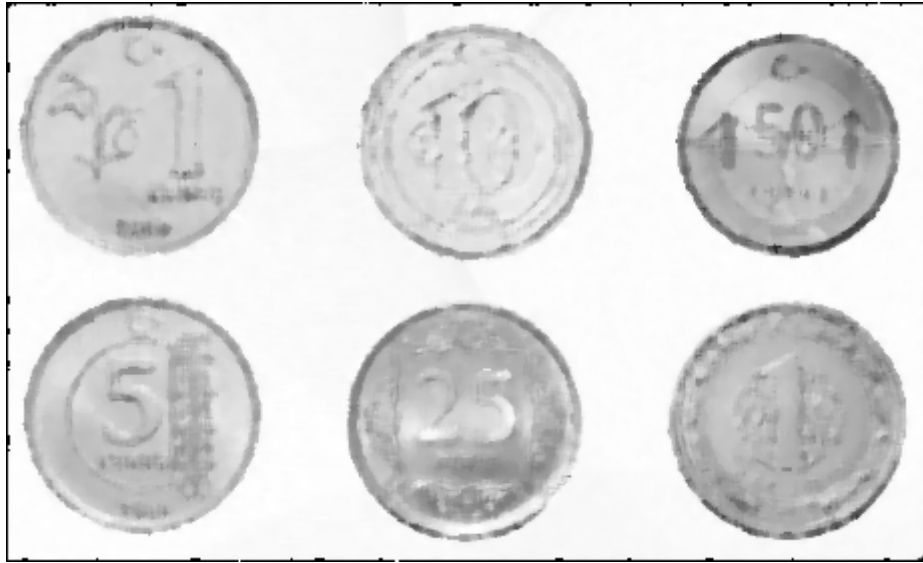
Şekil 3.7’ deki görüntü işaretine %20 oranında gürültü işareti eklenmiş sonra bu gürültülü işaret sırasıyla, 3x3, 5x5 ve 7x7 medyan filtre şablonları kullanılarak medyan filtreden geçirilmiştir.



Şekil 3.8 20% oranında gürültülü görüntü işareti



Şekil 3.9 3x3 medyan filtre şablonu ile filtrelenmiş görüntü



Şekil 3.10 5x5 medyan filtre şablonu ile filtrelenmiş görüntü

Sonuç olarak orijinal görüntüye en yakın çıkış veren medyan filtre şablonu 3x3 kare şablondur. Görüldüğü gibi 5x5 ve 7x7 kare medyan filtre şablonları kullanıldığında resim üzerindeki detaylar giderek kaybolmaktadır. Sonuç olarak aynı çözünürlükteki bir resme uygulanacak olan kare medyan şablonların boyutu arttıkça resim üzerindeki detaylar kaybolur.



Şekil 3.11 7x7 medyan filtre şablonu ile filtrelenmiş görüntü

3.1.3 Medyan Filtrenin Uygulanışı

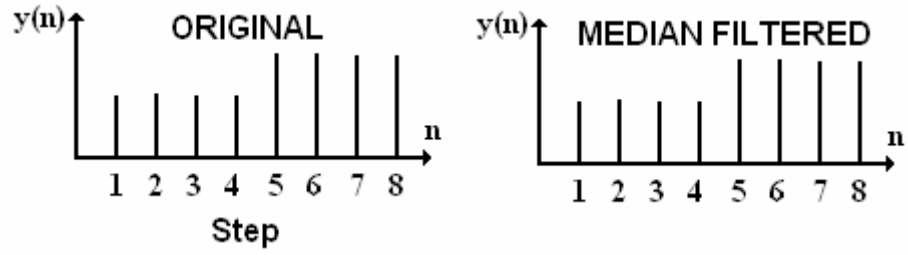
Medyan filtreler tek ve iki boyutlu işaretler üzerinde uygulanabilir.

$$Y(n) = \text{med}(x_{n-k}, \dots, x_n, \dots, x_{n+k}) \quad (3.1)$$

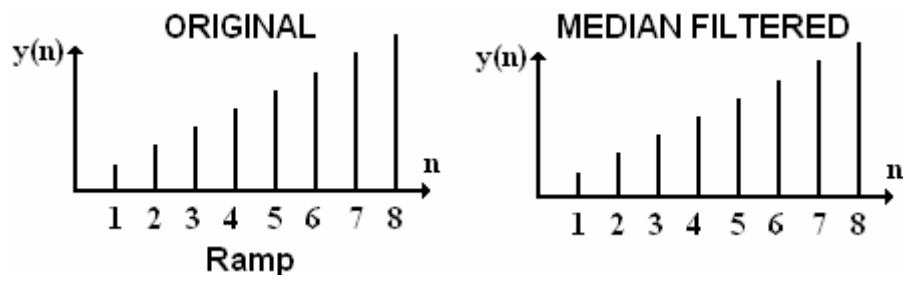
Gürültülü işaretleri bastırmada oldukça yaygın olarak kullanılan gürültü temizleme yöntemidir. Tek boyutlu işaretlerde alınan pencereye göre, pencere içerisinde bulunan elemanlar sıraya dizilir, ortadaki eleman medyan olarak bulunan eleman ile yer değiştirilir. Genellikle kullanılacak olan pencere tek elemanlı seçilir. Çift elemanlı seçilmesi durumunda ortada bulunan iki piksel değerinin aritmetik ortalaması alınır.

Şekil 3.12’de çeşitli türden dizi işaretlerine (genlik olarak) medyan filtresi uygulanmış ve sonuçları Şekil 3.12’de verilmiştir(*) (Acharya ve Ray, 2005).

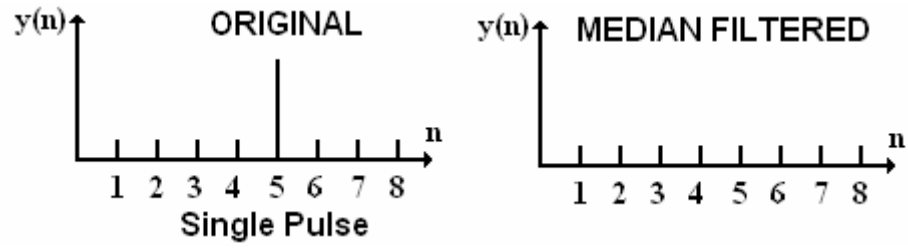
* Şekil 3.12’deki tek boyutlu işaretlere uygulanan medyan filtre şablonu beş uzunlukludur.



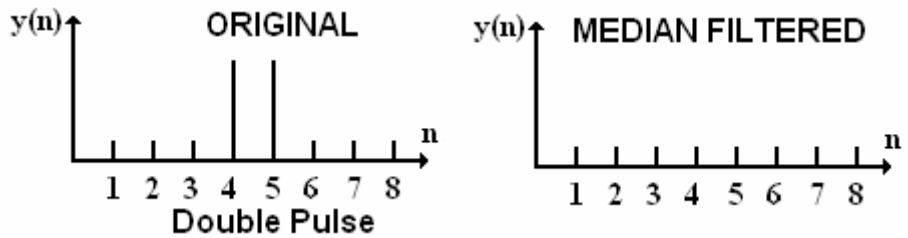
Şekil 3.12a Basamak şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001)



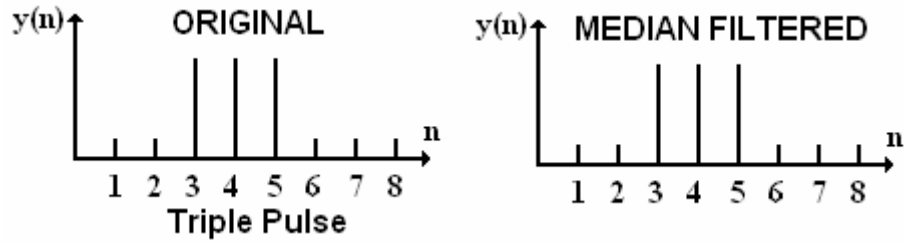
Şekil 3.12b Rampa şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001)



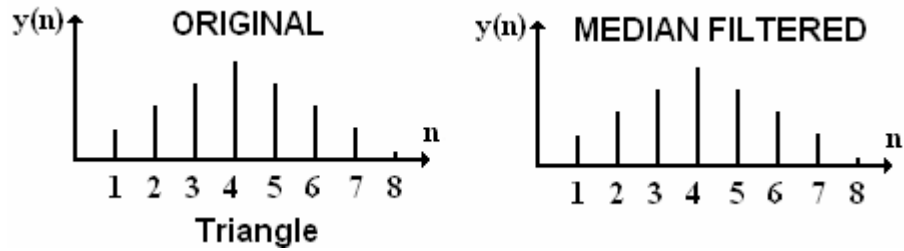
Şekil 3.12c Darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001)



Şekil 3.12d Çift darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001)



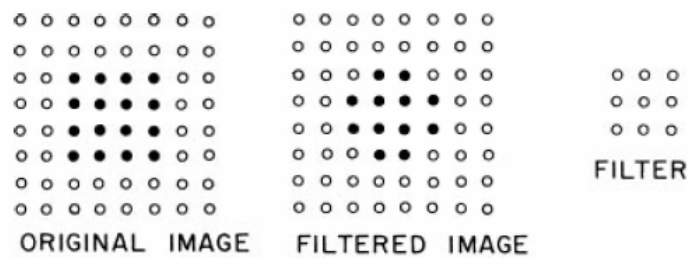
Şekil 3.12e Üçlü darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001)



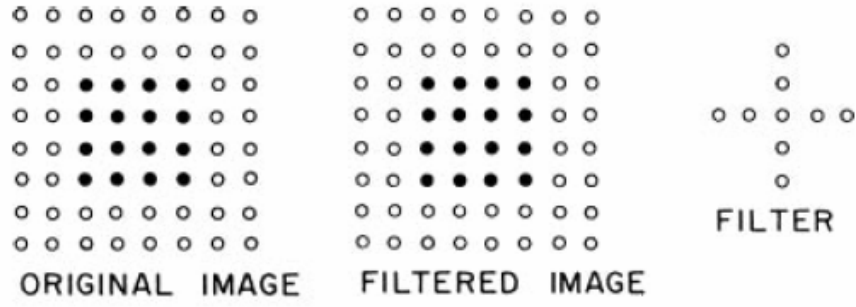
Şekil 3.12f Üçgen darbe şeklindeki gürültüde medyan filtrelemesi (Pratt, 2001)

Şekil 3.12’de görüldüğü gibi medyan filtreleme en iyi sonucu darbe şeklinde olan gürültülü işaretlere uygulandığında vermektedir. Tek darbe ve ikili darbe gürültülü işaretler medyan filtreden geçirildikten sonra gürültü işaretleri yok olmuştur. Zaten medyan filtrelerin kullanım yerleri darbe gürültülü işaretlerdir. Bu tür işaretlerde çok iyi sonuç verirler. Tek boyutlu işaretlerde kullanıldığı gibi iki boyutlu işaretlerde de gürültü bastırmada kullanılırlar.

Medyan filtreleme iki boyutlu işaretlerde, seçilen şablonun görüntü üzerinde hareket ettirilmesi ve her bir hareketi sırasında şablon içerisine gelen piksel değerlerinin büyüklük olarak sıralanıp bu değerlerden ortanca değerin bulunması daha sonra şablonun merkezinde bulunan piksel değeriyle yer değiştirilmesinden ibarettir (Pratt, 2001).



Şekil 3.13 Kare şeklindeki şablonun medyan filtrelemedeki etkisi (Pratt, 2001)



Şekil 3.14 Artı şeklindeki şablonun medyan filtrelemedeki etkisi (Pratt, 2001)

Medyan filtrelemede uygulanacak olan şablonunun şekli önemlidir. 3x3 lük bir şablonun görüntüye uygulanışı ile artı şeklinde şablonun aynı görüntüye uygulanması farklı sonuçlar vermektedir. 3x3 lük kare medyan filtresi şablonu gürültü bastırmada daha etkili sonuç verir.

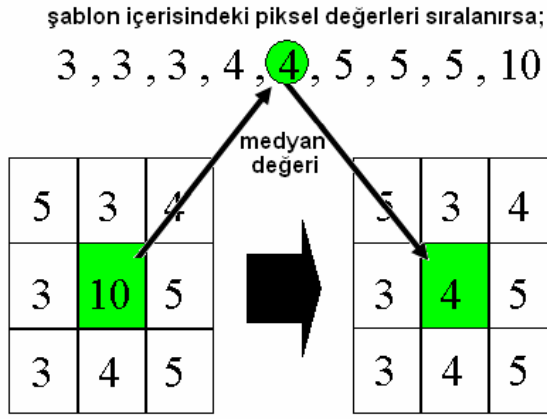
Görüntüde bulunan ve gürültü olarak nitelendirilen kare şekil, medyan filtreleme sonrasında köşeleri kaybolmuş fakat artı şeklindeki medyan filtrenin gürültülü görüntü işareti üzerinde böyle bir etkisi olmadığı görülmüştür.

	123	125	126	130	140
	122	124	126	127	135
	118	120	150	125	134
	119	115	119	123	133
	111	116	110	120	130

şablon içerisindeki piksel değerleri:
115, 119, 120, 123, 124, 125, 126, 127, 150
medyan değeri: 124

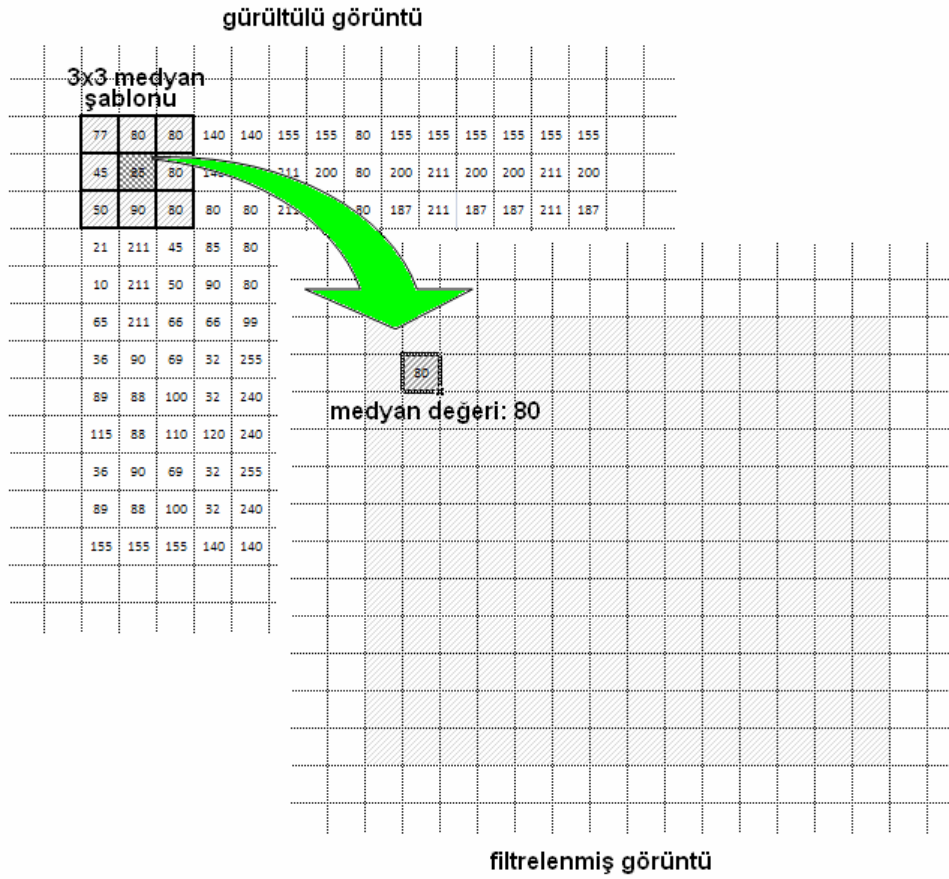
(a)

Şekil 3.15a 3x3 kare medyan filtre şablonlarıyla medyan değerinin bulunması

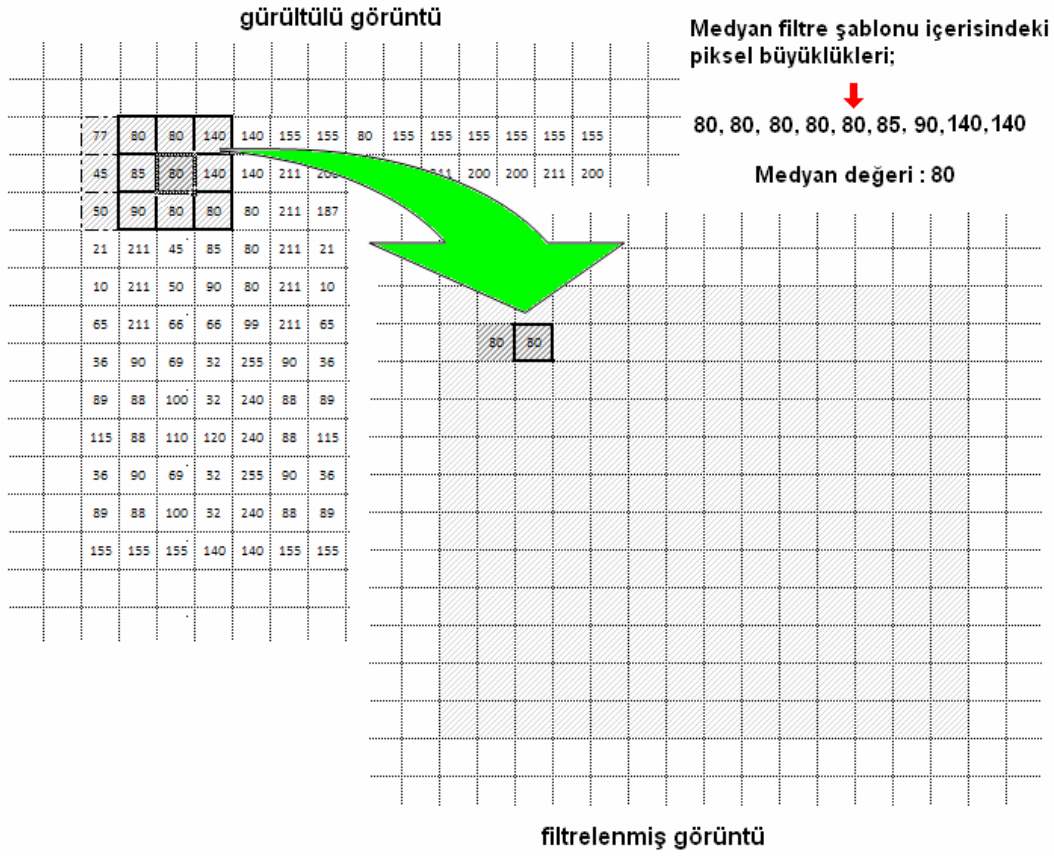


(b)

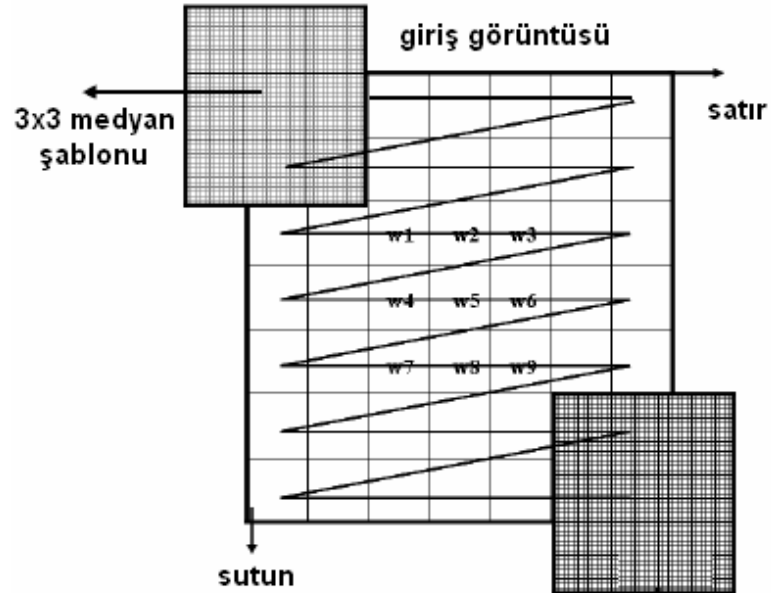
Şekil 3.15b 3x3 kare medyan filtre şablonlarıyla medyan değerinin bulunması



Şekil 3.16a Şablonun görüntü üzerindeki hareketi



Şekil 3.16b Şablonun görüntü üzerindeki hareketi



Şekil 3.16c Şablonun görüntü üzerindeki hareketi (Dhanasekaran ve Boopathy, 2009).

Şekil 3.16’larda görüldüğü gibi medyan şablonunun görüntü matrisi üzerine uygulanması ve medyan değerinin bulunup filtrelenmiş görüntü matrisinde uygun olan piksel yerine konulmasını göstermektedir. Standart lineer olmayan medyan filtrenin çalışma prensibi bu şekilde açıklanabilir (Dhanasekaran ve Boopathy, 2009).

4. MEDYAN FİLTRENİN FPGA İLE GERÇEKLENMESİ

Bir önceki bölümlerde medyan filtrelerin nasıl çalıştığını, tek ve iki boyutlu matrislere nasıl uygulandığı incelenmiştir. Bu bölümde ise gerçek zamanlı iki boyutlu görüntü üzerinde medyan filtre uygulamaları gerçekleştirilmiştir. Bu çalışmada, yüksek çözünürlüklü 720p (1280x720) gerçek zamanlı gürültülü işaretler üzerinde 3x3 kare medyan filtre şablonu kullanarak medyan filtreleme gerçekleştirilmiştir.

Görüntü çözünürlüğü veya görüntü boyutu arttığında söz konusu gerçek zamanlı olarak medyan filtrelemeyi gerçekleştirmek için oldukça hızlı işlem yapabilen işlemcilere, mikrodenetleyicilere ya da FPGA'lere (Field Programmable Gate Array) ihtiyaç duyulur.

Uygulamamızda görüntü işaretlerimiz, günümüz elektroniğinde oldukça yaygın ve popüler olan FPGA'ler yardımıyla temizlenmiştir. Paralel işlem yapabilme kapasitesine sahip olan programlanabilir lojik yapılar olan FPGA'lerin sağladığı avantajlar aşağıdaki gibi sıralanır.



Şekil 4.1 Xilinx virtex-5 serisinden bir fpga

- Birim zamanda çok hızlı çalışabilmeleri,
- Sıralı işlem yapabilmenin yanında paralel işlem yapabilme kapasiteleri,
- Yüksek çözünürlüklü görüntü işleme uygulamalarında sık tercih edilmeleri,
- Lojik kapılar seviyesinde kod yazabilme, belli bir iş için özelleştirilen blokların tasarlanmasına imkân sağlamak,

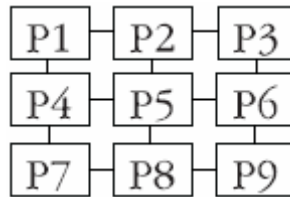
- Bir işi için özel olarak tasarlanmış bloklar, kombinezonsal ya da ardışık elemanlarla iş yapabilme,
- Oldukça esnek yazım diline sahip olması.
- Teknolojinin FPGA'lere doğru kayması, tasarımcılar arasında oldukça popüler olması olarak sıralanabilir.

Medyan filtrelemede yapılan iş, medyan filtre şablonu olarak kullanılan 3x3'lük filtre şablonun görüntü işareti üzerinde gezdirmek ve her bir adımda şablonun içerisinde bulunan dokuz tane piksel değerini büyüklük olarak sıralayıp bulunan medyan değerini şablonun görüntü matrisi üzerinde durduğu yerin merkezindeki piksel değeriyle yer değiştirmektir. Bu işlem görüntü işareti ya da görüntü matrisinin her bir piksel değeri için tekrarlanır.

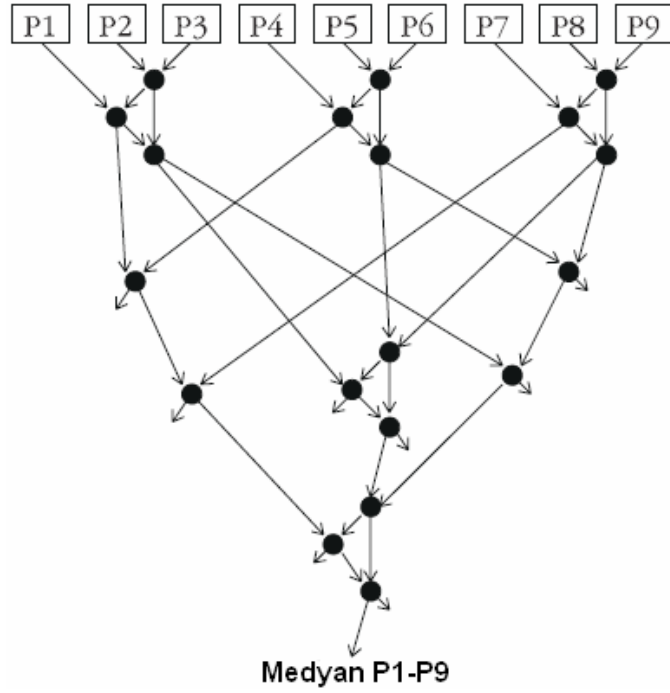
Aşağıdaki alt bölümlerde, 3x3 kare medyan şablonunun gerçekleştirdiği işi yapan iki farklı blok yapısı incelenmiştir.

4.1 Paralel Girişli Medyan Filtre Bloğu

Adından da anlaşılacağı üzere dokuz adet piksel değeri medyan filtre bloğunun girişine aynı anda verilir. Bloğun çalışma prensibine göre çıkışında ise bu dokuz piksel değerinin küçükten büyüğe sıralandığında beşinci elamanı (medyan elaman) çıkışa aktarılmış olur.



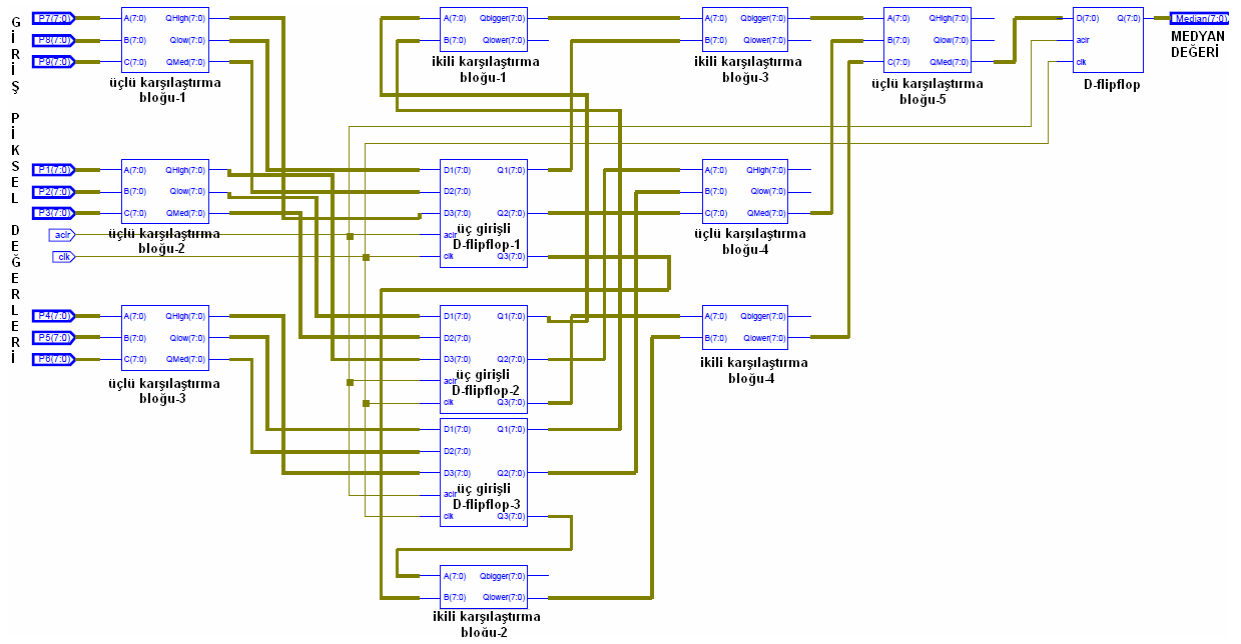
Şekil 4.2 3x3 kare medyan filtre şablonu (Smith, 1996)



Şekil 4.3 Paralel girişli 3x3 kare medyan filtre blok yapısı (Smith, 1996)

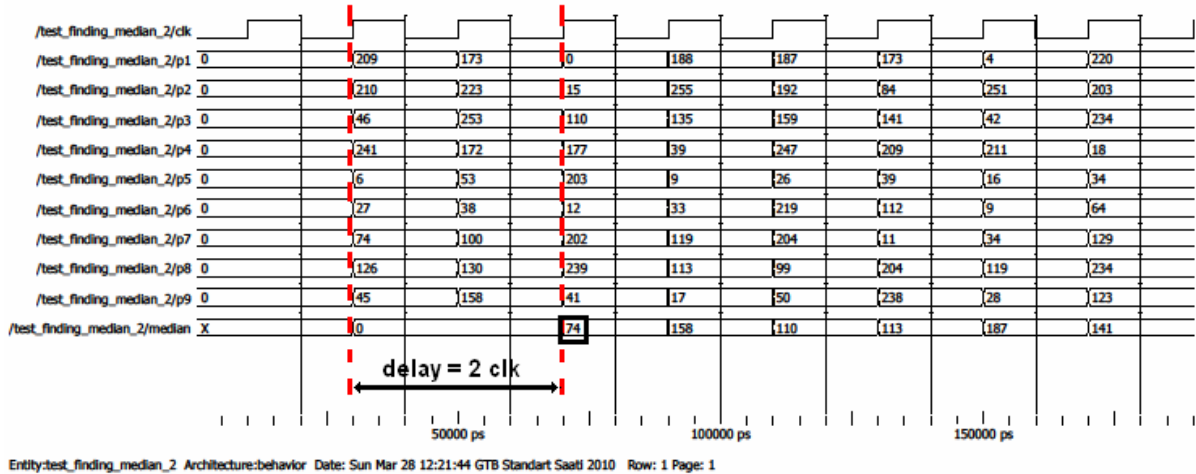
Şekil 4.3'te görüldüğü gibi görüntü üzerinde, 3x3 kare medyan şablonun üzerine geldiği dokuz adet piksel değeri, paralel girişli 3x3 kare medyan filtre blok yapısının girişleri olan P1, P2, P3, P4, P5, P6, P7, P8, P9 uçlarının her birine atanır. Bu dokuz adet piksel değeri blok içerisinde birbirleriyle karşılaştırılarak sıraya konur ve bulunan medyan piksel değeri bloğun çıkışına iletilir (Smith, 1996).

Bloğun yapısına kısaca değinecek olursak devre kombinezon sal karşılaştırmacı blokları ve bu blokların aralarına belli seviyelerde yerleştirilen ardışıl D-bellek elemanlarından oluşur. Clk ve aclr sinyal giriş uçlarıyla kontrol edilen paralel girişli medyan blok yapısı aşağıdaki şekildeki gibidir.



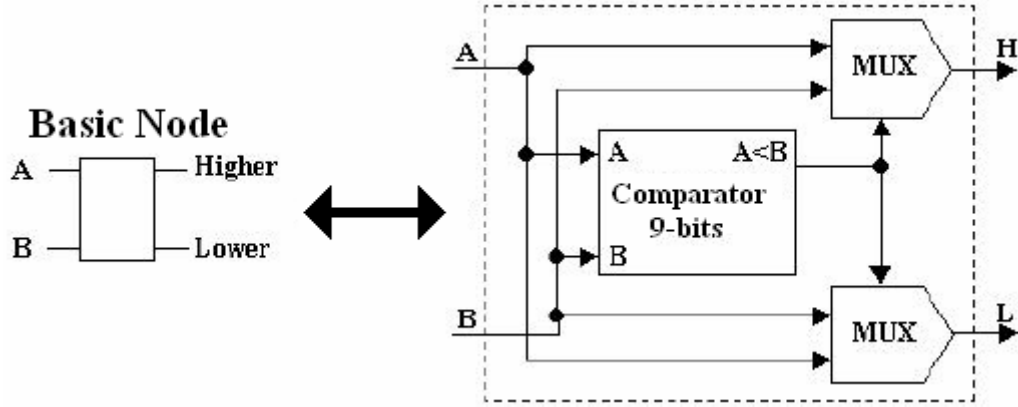
Şekil 4.4 Paralel girişli medyan filtre bloğu içyapısı

Bir “clk” darbesiyle bloğun girişlerine gelen piksel değerleri, tasarlanan bu blokta değerlendirilir ve birbirini izleyen iki “clk” sonrasında çıkışta medyan piksel değeri oluşturulur.



Şekil 4.5 Paralel girişli medyan filtre bloğu gecikmesi

Şekil 4.3’de belirtilen her bir siyah düğüm noktası bir karşılaştırma bloğunu temsil eder. Bu karşılaştırma blokları kendi içlerinde birer alt bloğu oluştururlar. İki girişli iki çıkışlı olan bu alt bloklarda piksel değerleri ikili olarak karşılaştırılır.



Şekil 4.6 İkili karşılaştırıcı alt blok gösterimi (Vega-Rodriguez vd., 2002)

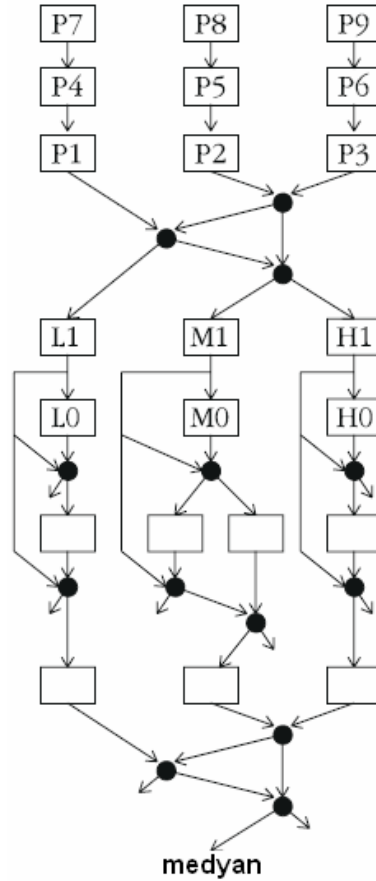
Karşılaştırılan piksel değerleri bloğun çıkışında soldan başlayarak sırayla küçük ve büyük olarak sıralanır. Bu dokuz adet piksel değerleri bu mantıkla sıralanıp, medyan piksel bulunmuş olur.

4.2 İş Hattı Mantığıyla Oluşturulmuş Medyan Filtre Bloğu

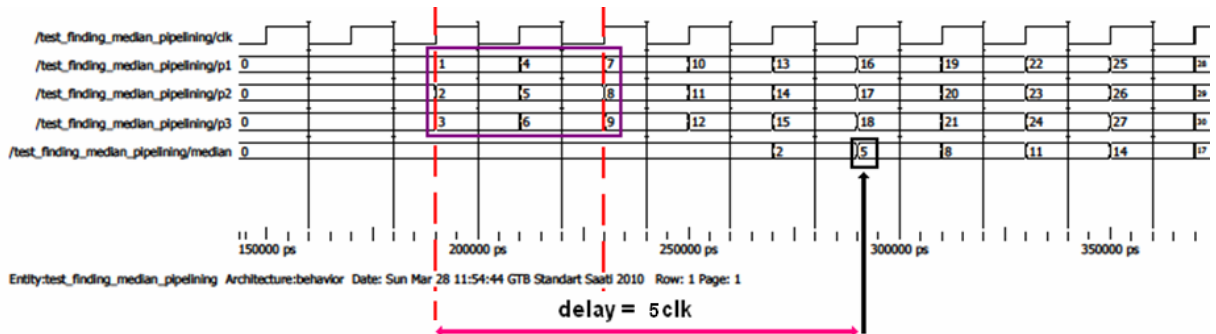
Bu yapı paralel girişli medyan yapısından farklı olup, iş hattı mantığıyla seri olarak çalışmaktadır. Bloğun çalışma mantığı paralel blok gibi medyan piksel değerini bulmak üzerinedir. Paralel bloktan farkı, dokuz adet piksel değerleri aynı anda değil üçerli olarak seri şekilde girişe uygulanmasıdır. Bloğun her bir kısmı medyan piksel değerini bulma anlamında farklı bir iş için özelleşmiş bu nedenle çalışması iş hattı mantığına benzetilmiştir.

Her bir saat darbesiyle yapının girişine 3x3 kare matrisin dokuz adet piksel değerlerinden birbiri ardına her bir saat darbesiyle birlikte üç piksel girişe aktarılır. Böylece üç saat darbesi sonunda şablonun içeriği olan dokuz adet piksel değeri yapının girişlerine uygulanmış olur. Yapının gövdesinde bu girişler değerlendirilerek bulunan medyan piksel değeri çıkışa aktarılır (Smith, 1996).

İş hattı mantığıyla oluşturulan medyan blok yapısı Şekil 4.7’de olduğu gibidir. FPGA yapısında uygun bir düzen oluşturulduğunda bu işlem sıralı olarak her bir saat darbesi ile tekrarlanıp, görüntü işaretini oluşturan her bir piksel değeri için medyan piksel hesaplanmış olur. Ek 1’de medyan değerini bulan iş hattı (pipeline) bloğuna ait yazmaç, ikili ve üçlü karşılaştırmacı bloklardan oluşan bir alt seviye içyapısını inceleyebilirsiniz. Ek 1, Şekil 4.7’nin daha detaylı bir blok gösterimidir.



Şekil 4.7 İş hattı mantığıyla oluşturulmuş medyan filtre blok yapısı (Smith, 1996)



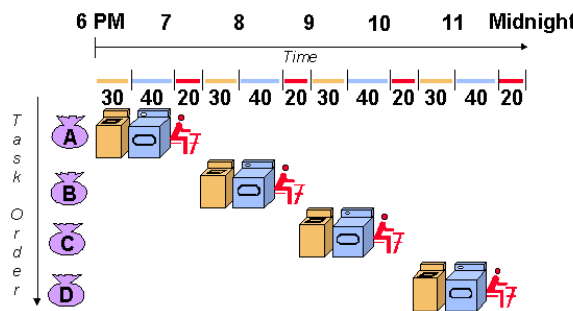
Şekil 4.8 İş hattı mantığıyla oluşturulmuş medyan filtre bloğunun gecikmesi

Blok gecikmesi 3 sistem saati kadardır. Daha açık bir şekilde bloğun girişine gönderilen dokuz adet piksel büyüklüğünün tamamlanmasından sonra medyan değerinin belirlenip bloğun dışına çıkış olarak aktarılması ancak 3 sistem saati sonrasında gerçekleşir. Bloğun gecikmesini şu şekilde değerlendirmekte mümkündür. Bloğun girişine uygulanan ilk üç piksel değerinden sonra sadece beş sistem saati sonunda medyan piksel çıkışa aktarılır. Gecikmeyi (latency) bu şekilde değerlendirmek daha doğru olacaktır.

İş hattının bir örnekle açıklanması bu mantıkla oluşturulan medyan bloğunun çalışma şeklini daha iyi anlatacaktır. İş hattında en önemli nokta, bir yapının (instruction) işleyişini evrelere bölmektir. Bu sayede işlemci farklı yapıların farklı evreleri üzerinde aynı anda çalışabilir ve az zamanda daha çok iş yapılabilir. Bu durum bir örnekle açıklanabilir (Smith, 1996).

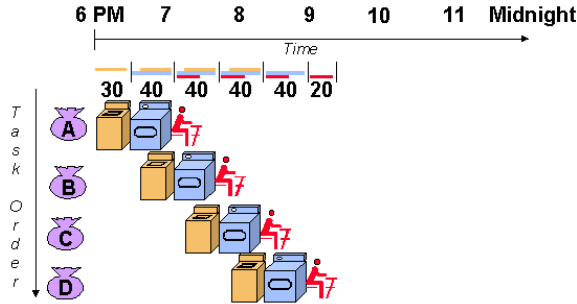
1.durum: Çamaşırhanede tek bir çamaşır makinesi, tek bir kurutma makinesi ve katlama yapılabilecek tek bir yer var. Yıkama işlemi 30 dk., kurutma 40 dk, katlama da 20 dk sürmektedir. Sırada 4 kişi beklemekte, birinin işi tamamen bitmeden diğer işler yapılamamaktadır. Öğleden sonra 6'da başlanıldığı ve bir müşteriden diğerine geçerken süre kaybedilmediği farz edilirse ancak gece yarısında 4 müşterinin tüm işleri bitmiş olur (e-bergi, 2009).

Çizelge 4.1 1.durum (e-bergi, 2009)



2.durum: Ancak birinci müşterinin çamaşır makinesiyle işi bitip kurutma makinesini kullanmaya başladığında, 2. müşterinin daha fazla beklemesine ihtiyaç yoktur. Çamaşır makinesini kullanmaya artık başlayabilir çünkü yıkama, kurutma ve katlama işlemleri de birbirinden bağımsız işlemlerdir. Bu şekilde 4 kişinin işi akşam saat 9.30 civarı bitecektir. Bu şekilde gerçekleştirilen bir sisteme iş hattı (pipeline) yapısı denmektedir (e-bergi, 2009).

Çizelge 4.2 2.durum (e-bergi, 2009)



4.3 Medyan Blokların Karşılaştırılması

Bu bölümde, paralel medyan bloğu ile iş hattı mantığıyla oluşturulmuş medyan bloğun karşılaştırılması yapılmıştır. İki blok yapısı da, görev itibarıyla benzer işi yerine getirir. İkisi de dokuz adet piksel değerinin içerisinde medyan piksel değerini bulup çıkışa aktarır. Farkları, yapıları ve çalışma şekilleridir.

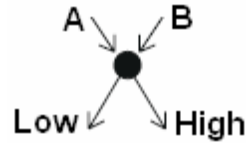
Paralel girişli medyan bloğu, medyan filtre şablonundaki dokuz adet piksel büyüklüğünü tek seferde bloğun içine alır ve bulduğu medyan pikseli iki sistem saati sonunda çıkışa aktarır.

İş hattı mantığıyla oluşturulmuş medyan bloğu ise piksel büyüklüklerini üçerli alarak işleme başlar. Böylelikle şablon içindeki dokuz adet piksel değeri üç sistem saati ile blok içine alınmış olur. Çıkış olan medyan değeri bulabilmek için, şablonun içerisindeki tüm piksel büyüklüklerinin giriş uçlarına uygulanması gerekir. Bulduğu medyan pikselini, girişine uygulanan ilk üç piksel değerinin sistem saatinden 5 sistem saati sonra çıkışa aktarır.

Bu çalışmamızda kullanılacak bloğun iş hattı (pipeline) veya paralel yapıda olmasının çok bir farkı olmayacaktır. Fakat büyük bir sistem düşünüldüğünde iş hattı yapısının avantajları daha fazla olduğundan tercih edilmelidir. Bu çalışmada da iş hattı mantığıyla çalışan blok tercih edilmiş ve uygulamalar o yapı temel alınarak gerçekleştirilmiştir.

4.4 İkili Karşılaştırmalı Yapıları

Sistemimizin en temel yapı birimi karşılaştırmalı blokları, lojik karşılaştırmacılar ve davranışsal karşılaştırmacılar olmak üzere ikiye ayrılır.



Şekil 4.9 İkili karşılaştırıcı alt blok gösterimi

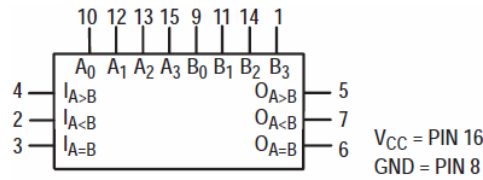
Bir grup piksel büyüklüğü arasından medyan değerin bulunabilmesi için bu piksel büyüklüklerinin hepsi birbirleriyle büyüklük açısından karşılaştırılmalıdır. Karşılaştırma işlemlerini gerçekleştirmede her iki blokta da Şekil 4.9'daki karşılaştırıcı blok kullanılmıştır. Aşağıdaki bölümlerde bu karşılaştırıcı blokların çeşitleri üzerinde durulmuştur.

4.4.1 Lojik Karşılaştırıcılar

Bu bölümde lojik karşılaştırıcı yapılarından bazıları üzerinde durulmuş, incelenmiş ve tez kapsamında kullanılabilecek karşılaştırıcı bloklara yer verilmiştir.

4.4.1.1 74LS85 4-bit Karşılaştırıcı

4-Bit iki sayıyı karşılaştıran 74LS85 tümleşik devresi ikilik sistemdeki iki sayıyı ya da ikili kodlanmış onluk (BCD) iki sayıyı karşılaştırır. Sonucu $Q_{A>B}$, $Q_{A<B}$ veya $Q_{A=B}$ gibi birer bitlik üç farklı çıkış ucuna aktarır.

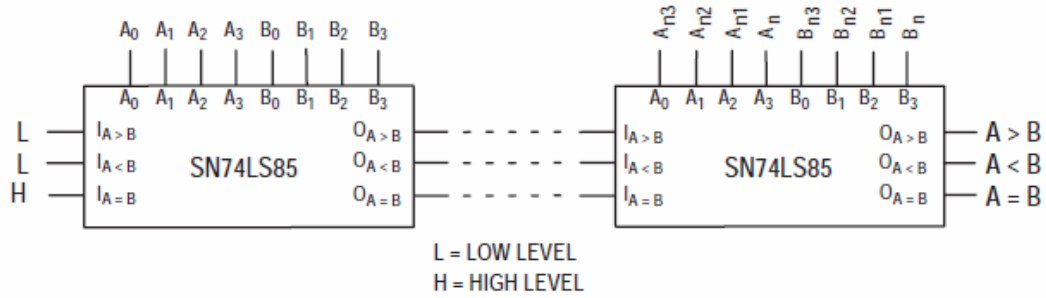


Şekil 4.10 74LS85'in lojik sembolü(*)

* 74LS85 onsemi marka tümleşik devresine ait bilgiler alldatasheet internet sitesinden alınmıştır.

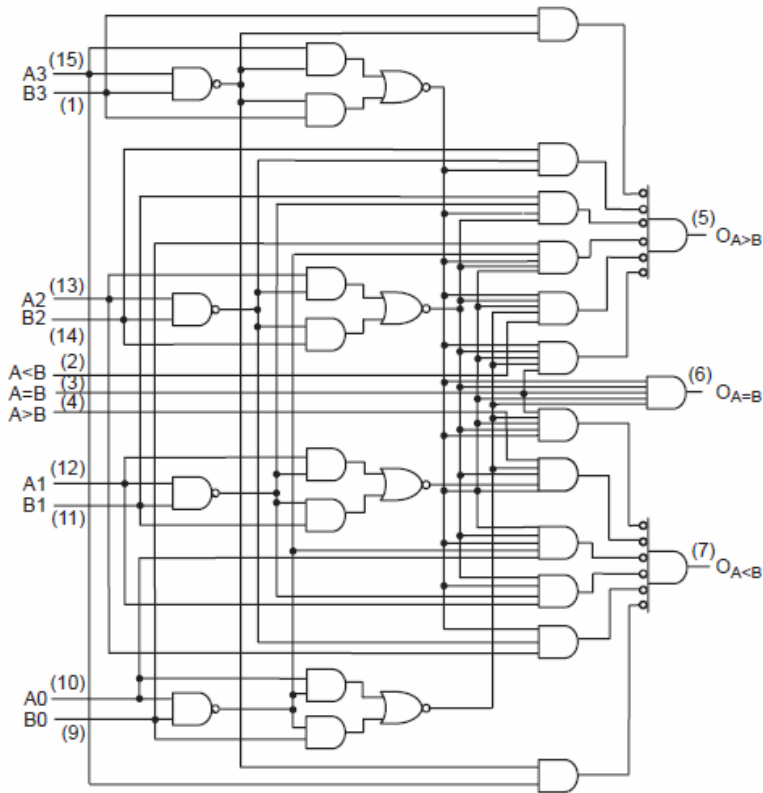
<http://pdf1.alldatasheet.com/datasheet-pdf/view/12664/ONSEMI/74LS85.html>

4-bitten daha büyük sayılar karşılaştırılmak istendiğinde, bu ikili karşılaştırıcı eleman birbirine kaskad (ard arda) bağlanır.



Şekil 4.11 n-bitlik iki sayının karşılaştırılması

VHDL kodunda bu elemanı gerçekleyebilmek için karşılaştırıcının lojik yapısı doğruluk tablosuyla birlikte incelendiğinde (Şekil 4.12, Çizelge 4.3),



Şekil 4.12 74LS85'in lojik yapısı(*)

* 74LS85 onsemi marka tümleşik devresine ait bilgiler alldatasheet internet sitesinden alınmıştır.

<http://pdf1.alldatasheet.com/datasheet-pdf/view/12664/ONSEMI/74LS85.html>

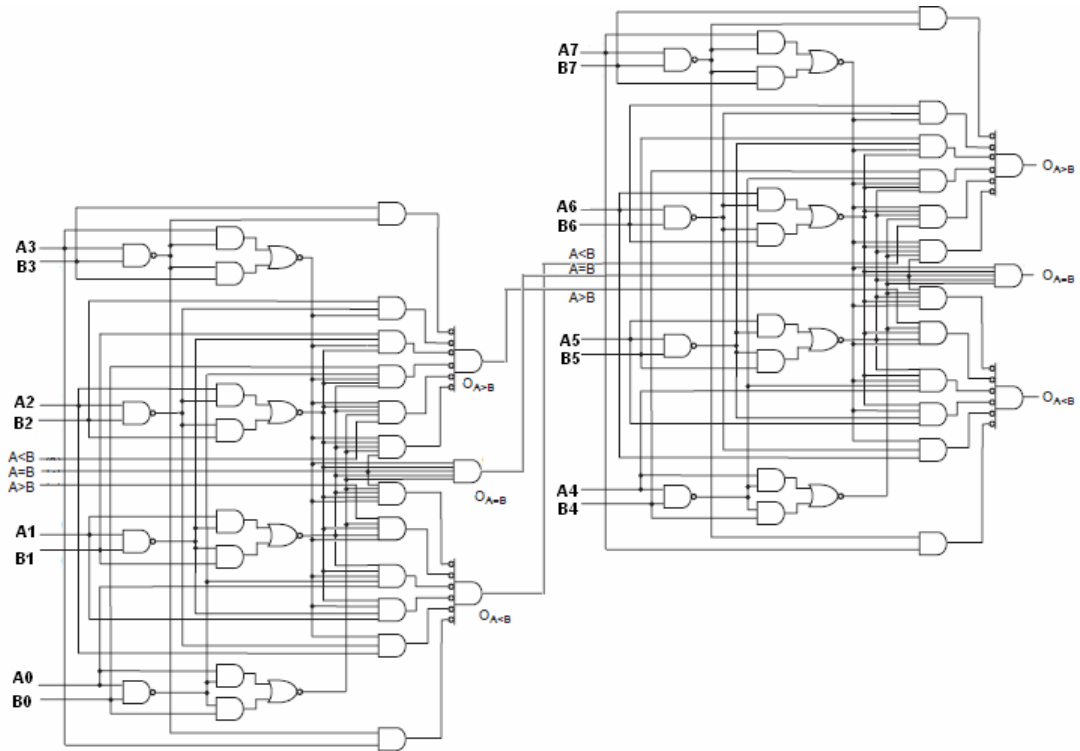
8-bit olarak VHDL kodunu yazmamız daha kolay olacaktır.

Çizelge 4.3 74LS85'in doğruluk tablosu

COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A ₃ B ₃	A ₂ B ₂	A ₁ B ₁	A ₀ B ₀	I _{A>B}	I _{A<B}	I _{A=B}	O _{A>B}	O _{A<B}	O _{A=B}
A ₃ >B ₃	X	X	X	X	X	X	H	L	L
A ₃ <B ₃	X	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ >B ₂	X	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ <B ₂	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ >B ₁	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ <B ₁	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ >B ₀	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ <B ₀	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	L	L	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	H	L	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	X	X	H	L	L	H
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	H	L	L	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	L	L	H	H	L

H = HIGH Level
L = LOW Level
X = IMMATERIAL

8-bitlik iki sayı kaskad bağlandığında elde edilecek olan karşılaştırıcı bloğun lojik yapısı Şekil 4.13'te gösterilmiştir.

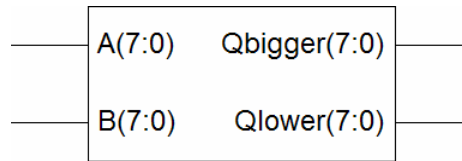


Şekil 4.13 74LS85 ile 8-bitlik karşılaştırıcı bloğu(*)

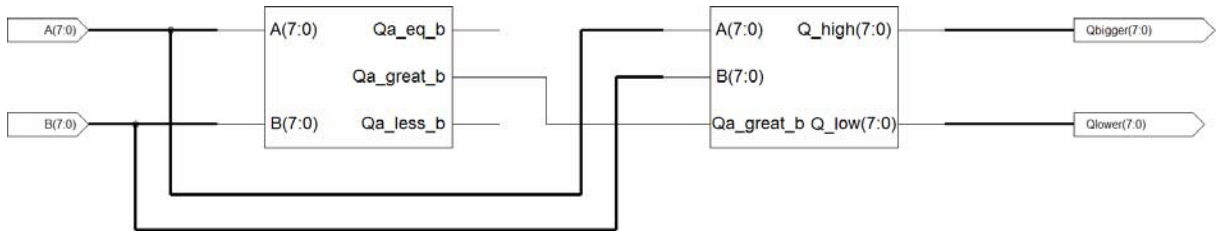
* 74LS85 onsemi marka tümleşik devresine ait bilgiler alldatasheet internet sitesinden alınmıştır.

<http://pdf1.alldatasheet.com/datasheet-pdf/view/12664/ONSEMI/74LS85.html>

Elde edilen 8-bitlik iki sayıyı karşılaştıran lojik yapının çıkışları Şekil 4.13'te gösterilmiştir. Bloğun $I_{A>B}$, $I_{A=B}$, $I_{A<B}$ girişleri lojik 'L' a çekilmiştir. Bloğun çıkışları $Q_{A>B}$, $Q_{A<B}$ veya $Q_{A=B}$ 'L' ya da 'H' olarak gösterilmektedir. Tasarımda gerekecek karşılaştırmacı yapısı 8 bit iki sayıyı karşılaştırmakla kalmayıp, küçük ve büyük sayıları göstermelidir. Böyle bir yapıyı elde edebilmek için ise lojik yapının çıkışına iki adet iki girişli tek çıkışlı çoğullayıcılar (multiplexer) eklemek gerekecektir. Çoğullayıcının kontrol ucunu ise, 74LS85 kullanarak gerçekleştirdiğimiz lojik devremizin çıkışı (Bu tasarımda kontrol ucu lojik devrenin $Q_{A>B}$ çıkışı alınmıştır) oluşturur.



Şekil 4.14 İki giriş-iki çıkışlı karşılaştırmacı bloğu



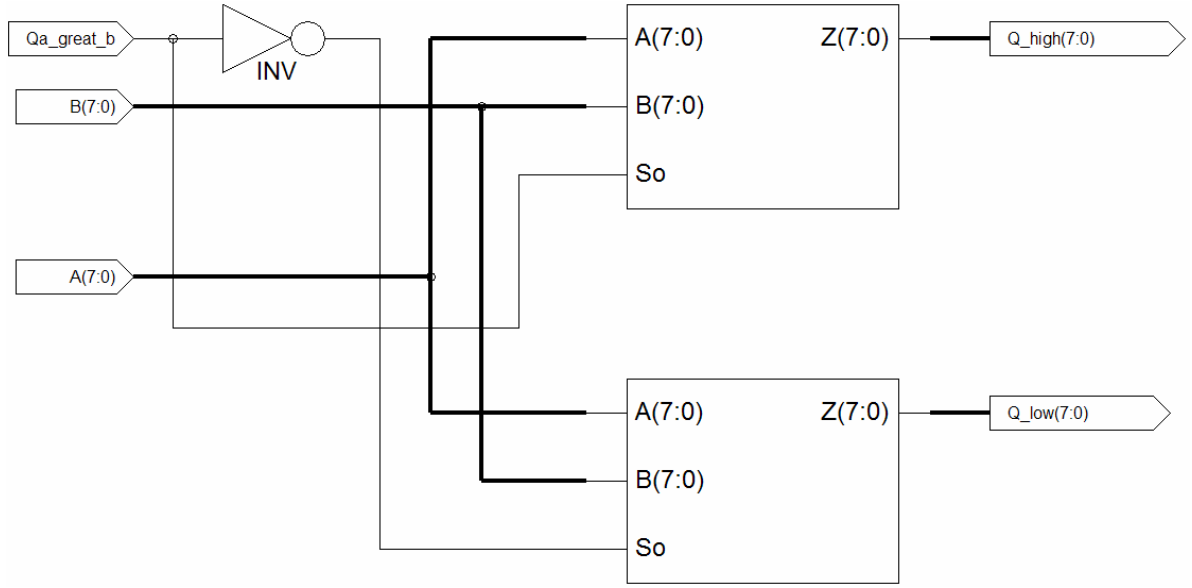
Şekil 4.15 İki giriş-iki çıkışlı karşılaştırmacı bloğunun içyapısı

2x2 çoğullayıcı içyapısına da kısaca değinilecek olursa karşılaştırılan 8-bitlik A ve B sayıları karşılaştırmacı bloğun girişleri, çoğullayıcının kontrol ucu ise karşılaştırmacı bloğun $Q_{A>B}$ çıkışıdır. $Q_{A>B}$ çıkışı 'H' olduğunda Şekil 4.16'daki üst kısımda bulunan 2x1 çoğullayıcının kontrol ucu 'H' olduğundan çoğullayıcının çıkışı 8-bit A sayısını büyük sayı olarak çıkışa aktarır.

Alt kısımda bulunan 2x1 çoğullayıcının kontrol ucu da eviriciden dolayı 'L' olacağından çoğullayıcının çıkışı 8-bit B sayısını da küçük sayı olarak çıkışa aktarır.

Bloğun $I_{A>B}$, $I_{A=B}$, $I_{A<B}$ girişleri lojik 'L' değerine çekildiği için A ve B sayılarının eşit olması durumunda da $Q_{A>B}$ çıkışı 'H' olacağından 2x2 çoğullayıcının büyük sayı çıkışı A sayısını,

küçük sayı çıkışı da B sayısını verir.



Şekil 4.16 2x2 çoğullayıcı bloğunun içyapısı

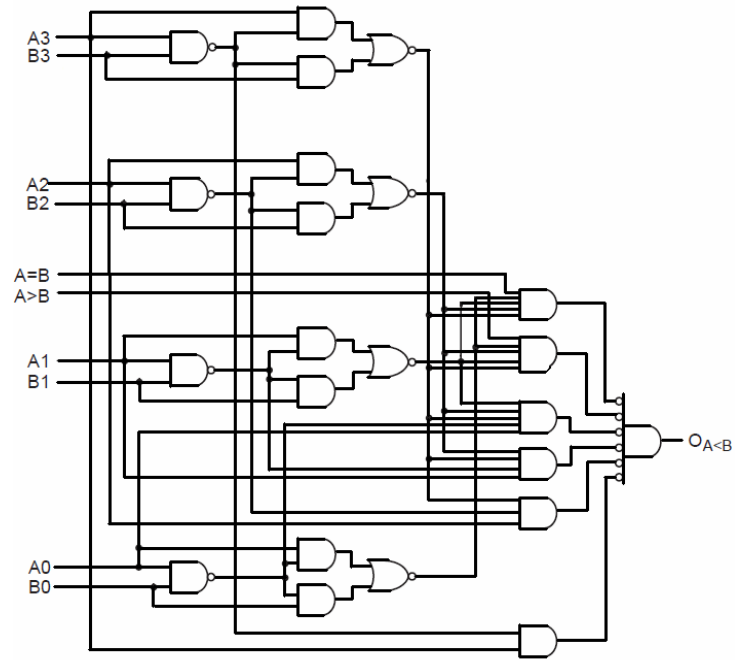
Bu tez çalışması büyük bir görüntü işleme projesinin bir parçasını oluşturmaktadır. Gerçeklenecek olan hdl tasarımı FPGA üzerinde olabildiğince az yer kaplamalıdır. Yüksek çözünürlüklü gerçek zamanlı video görüntüsünün işleneceği de düşünüldüğünde (saniyede milyonlar mertebesinde işlem yükü), kullanılacak olan karşılaştırıcı bloğun minimum sayıda lojik kapılarla gerçekleştirilmesi, aynı zamanda minimum zamanda çözüme gitmesi modülün hızı ve performansı açısından önemlidir. Bu nedenle tasarlanan kombinezonsal devre yapısında optimizasyona gidilerek karşılaştırıcı yapının daha az sayıda eleman ile gerçekleştirilmesi üzerinde durulmuştur.

4.4.1.2 Optimize Edilmiş 74LS85 4-bit Karşılaştırıcı

Bölüm 4.4.1.1’de gerçekleştirilen karşılaştırıcı yapısında kullanılmayan lojik kapılar kaldırılmış yapının minimum adette kapı elemanı ile gerçekleştirilmesi sağlanmıştır. Elde edilen optimizasyon sonucunda aşağıdaki lojik yapılar elde edilmiştir.

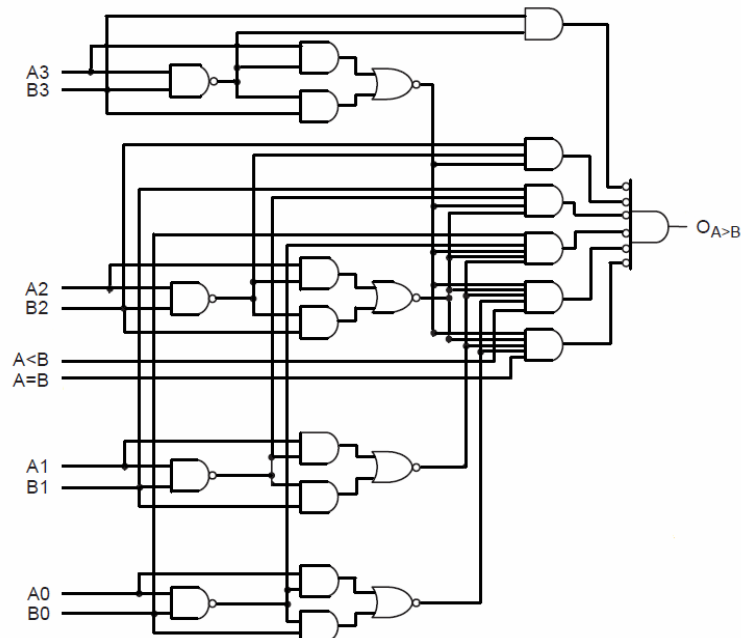
SN74LS85 4-BIT COMPARATOR (MINIMIZE) - LSB

LOGIC DIAGRAM

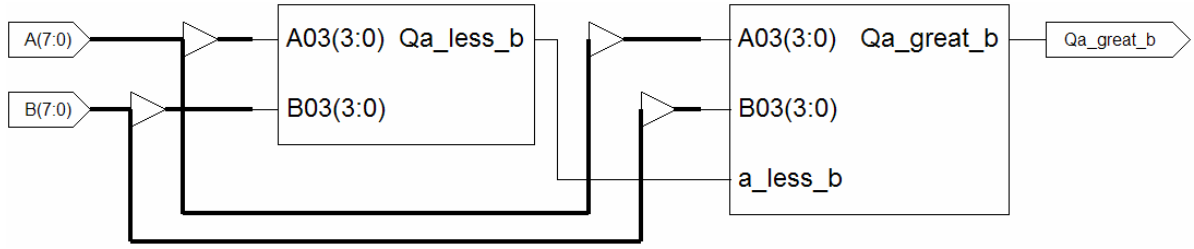


Şekil 4.17 8-bit optimize edilmiş 74LS85 karşılaştırıcısının LSB'si

LOGIC DIAGRAM



Şekil 4.18 8-bit optimize edilmiş 74LS85 karşılaştırıcısının MSB'si



Şekil 4.21 8-bit optimize edilmiş yapının bloklar halinde gösterimi

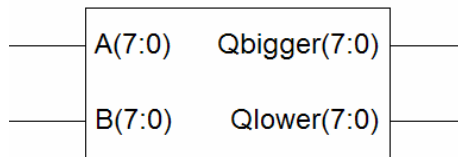
Çizelge 4.4 8-bit optimize edilmiş karşılaştırıcının doğruluk tablosu

<i>Comparator Inputs</i>	<i>Cascading Inputs</i>			<i>Comparator Output</i>
	Ia>b	Ia<b	Ia=b	Qa>b
A>B	X	X	X	H
A<B	X	X	X	L
A=B	L	L	L	H
A=B	H	H	H	L

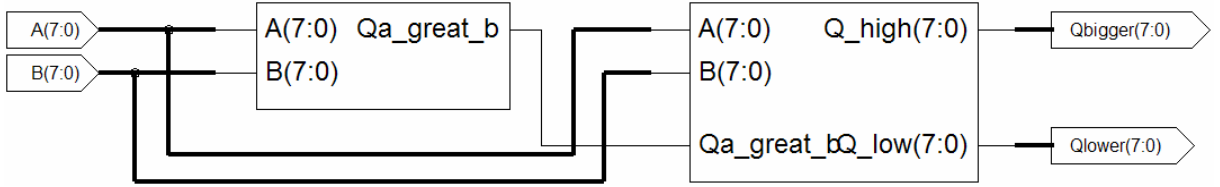
X=IMMETERIAL, L=LOW LEVEL, H=HIGH LEVEL

Çizelge 4.4'ten anlaşılacağı üzere karşılaştırıcı bloğun kaskad girişleri ne olursa olsun A sayısı B' den büyükse karşılaştırıcı çıkışı lojik 'H', A sayısı B' den küçükse karşılaştırıcı çıkışı lojik 'L' olur. Girişlerin eşit olması durumunda karşılaştırıcı bloğun çıkışının pek önemi yoktur.

Şekil 4.20'deki optimize edilmiş karşılaştırıcı bloğun çıkışına yine 2x2 çoğullayıcı bloğu eklenerek aşağıdaki iki girişli, iki çıkışlı karşılaştırıcı yapısı elde edilir.



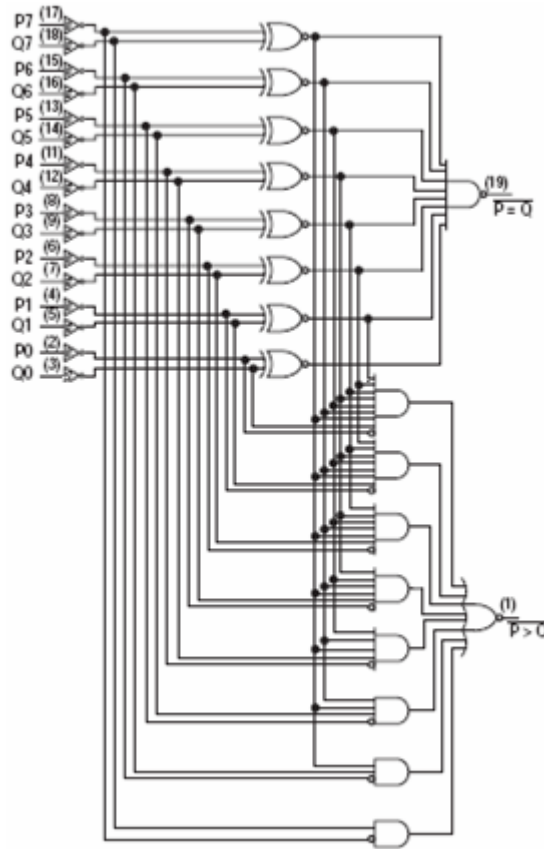
Şekil 4.22 İki giriş-iki çıkış 8-bit optimize edilmiş karşılaştırıcı bloğu



Şekil 4.23 İki giriş-iki çıkış 8-bit optimize edilmiş karşılaştırıcının blok halinde gösterimi

4.4.1.3 74LS682 8-bit Karşılaştırıcı

İncelenen diğer lojik karşılaştırıcı yapısı da 74LS682 tümleşik devresidir. Bu tümleşik devrenin girişleri 74LS85'ten farkı iki tane 8-bit sayıyı karşılaştırabilen bir yapıdadır. Böylelikle tek bir blok yapı ile karşılaştırıcı modül gerçekleştirilmiş olur.



Şekil 4.24 74LS682'nin lojik diyagramı(*)

* 74LS682 onsemi marka tümleşik devresine ait bilgiler alldatasheet internet sitesinden alınmıştır.

http://www.datasheetcatalog.org/datasheets/150/491878_DS.pdf

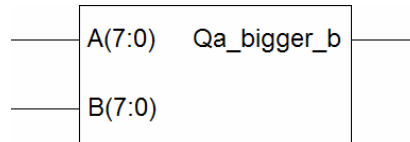
Çizelge 4.5 74LS682'nin doğruluk tablosu

INPUTS			OUTPUTS	
DATA	ENABLES		$\overline{P=Q}$	$\overline{P>Q}$
P, Q	$\overline{G}, \overline{GT}$	$\overline{GZ}$		
P = Q	L	L	L	H
P > Q	L	L	H	L
P < Q	L	L	H	H
X	H	H	H	H

H = HIGH Level, L = LOW Level, X = Irrelevant

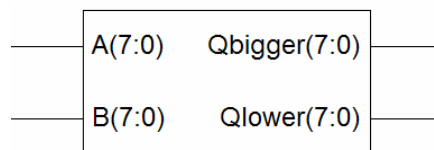
Bu yapı da yukarıda bahsedilen yapılara benzer bir bloktan oluşur. Bloğun girişi iki adet 8-bit sayı, çıkışı ise bir tane bir bitlik bir sinyaldir. Çizelge 4.5'teki doğruluk tablosundan da yararlanarak iki giriş iki çıkıştan oluşan karşılaştırıcı bloğu gerçekleştirir. Bu yapıda da yine 2x2 çoğullayıcı kullanılmıştır.

Kombinezonsal yapının çıkışlarından biri olan $\overline{P=Q}$ kullanılmadığı için karşılaştırıcı yapısından çıkarılmıştır. Diğer çıkış olan $\overline{P>Q}$ ucu kullanılmıştır. Doğruluk tablosundan da görüldüğü gibi P sayısı Q sayısından büyük olduğunda çıkış lojik '0', küçük olduğunda ise karşılaştırıcı yapının çıkışı lojik '1' olur. Çıkış sinyali evrik olduğundan 2x2 çoğullayıcı bloğunun seçim ucuna bağlanmadan önce bir evirici lojik kapısından geçirilmiştir.

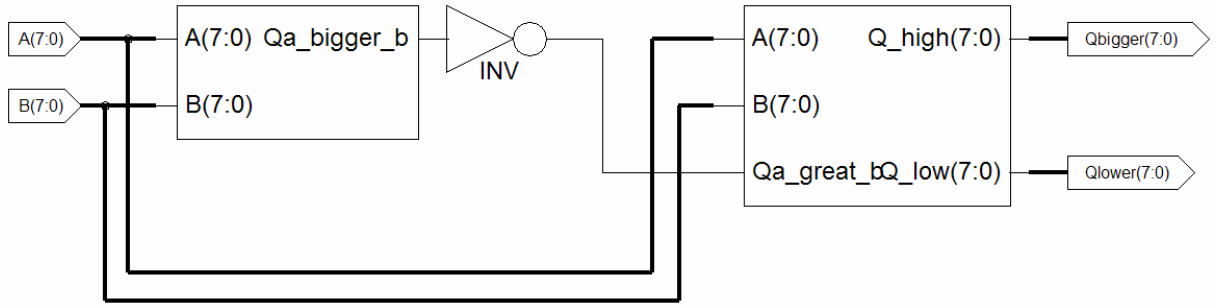


Şekil 4.25 74LS682'nin blok gösterimi

74LS682 tümleşik devresinin çıkışına çoğullayıcı eklenerek elde edilen iki 8-bit giriş, iki 8-bit çıkışlı karşılaştırıcı bloğu Şekil 4.26'daki gibi tanımlanabilir.



Şekil 4.26 İki giriş-iki çıkış 74LS682'li karşılaştırıcının blok gösterimi

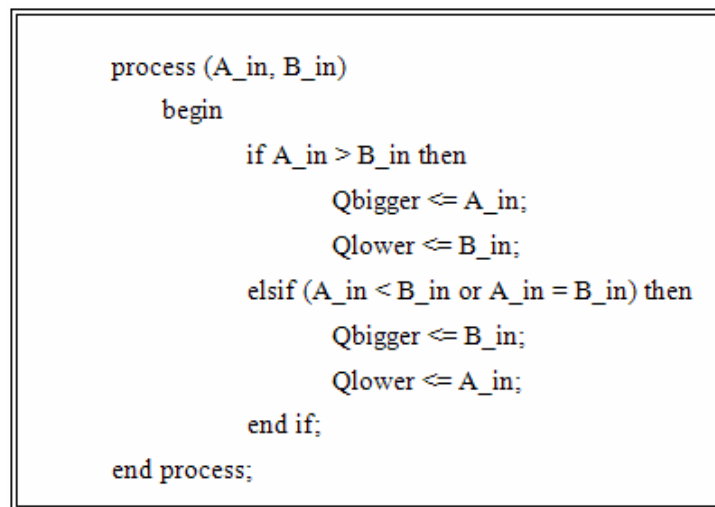


Şekil 4.27 İki giriş-iki çıkış 74LS682’li karşılaştırıcının iç blok gösterimi

4.4.2 Davranışsal Karşılaştırıcılar

Şimdiye kadar incelenen 8-bit iki sayıyı karşılaştıran karşılaştırıcılar lojik kapılardan oluşmaktadır. Bu yapılar birbiri ardına bağlanmış kombinezon sal devreler bütünüdür. Bu bölümde vhdl dilinde davranışsal olarak yazılan bir karşılaştırıcıdan bahsedilmiştir. Davranışsal (behavioural) olarak 8-bit iki sayıyı karşılaştıran yapı Çizelge 4.6’daki gibi tanımlanabilir.

Çizelge 4.6 Davranışsal 8-bit karşılaştırıcı bloğu



4.4.3 Karşılaştırmacı Yapıların Birbirleriyle Kıyaslanması

Şimdiye kadar incelenen tüm karşılaştırmacı yapılarını birbirlerine göre hız ve FPGA içerisinde kapladıkları alan bakımından kıyaslanacak olursa, birbirlerine göre avantaj ve dezavantajları bulunmaktadır. Yüksek çözünürlüklü gerçek zamanlı görüntünün işlenmesinde hızın çok büyük bir rolü olduğu gerçektir. Tasarımda FPGA kullanılmasının en önemli özelliği diğer yapılara (mikroişlemci, DSP vb...) göre hızlı olmasıdır.

Öyle ki, yüksek çözünürlükteki gerçek zamanlı bir görüntünün işlenebilmesi için FPGA de yapılan işlemlerin HD Ready çözünürlüğü için 74,25MHz olduğu dikkate alınırsa yapının ne kadar hızlı çalışması gerektiği anlaşılmaktadır. Dolayısıyla yapı içerisinde bulunan her bir lojik kapı gecikmelerinin önemi bu noktada ortaya çıkmaktadır. Bu nedenledir ki tez çalışması boyunca farklı karşılaştırmacı yapıları üzerinde durulmuş, hatta bazılarında ise yapıyı minimum sayıda lojik elemanla gerçekleştirmek adına karşılaştırmacı modüllerde optimizasyona gidilmiştir.

Çizelge 4.7’de her bir karşılaştırmacı bloğa ait bazı veriler yer almaktadır. Bu veriler sayesinde, karşılaştırmacıların FPGA üzerinde kapladıkları alandan kaç adet lojik birimden oluştuklarına, kodu yazan kişinin bloğun her bir noktasındaki sinyal hareketlerinin durumuna hâkim olup olmamasına göre değerlendirme yapılabilir.

Buradaki değerlendirme ölçütleri, istenen işlemi minimum sayıda lojik elemanlarla gerçeklemek ve işlemi en kısa zamanda sonuca götürmektir. Bu parametrelerin yanında tasarımcının kodun her noktasına hâkim olması, herhangi bir iç sinyaldeki değişimlere yetkin olması, kodun (yapının) ilettilmesi, güncellenmesi ve geliştirilmesi de diğer önemli parametrelerdir.

Çizelge 4.7’ye göre en az kapı ile gerçekleştirilen karşılaştırmacı 74LS682 tümleşik devresi ile gerçekleştirilen karşılaştırmacı yapıdır. Çizelge 4.7’deki karşılaştırma gecikme zamanları açısından ele alınacak olursa aşağıdaki veriler elde edilir.

Çizelge 4.7 Karşılaştırmalı lojik kapı sayıları

	74LS85	74LS85_MINİMİZE	74LS682	BEHAVIOURAL COMP_(IF)
AND KAPISI	46	30	8	148
OR KAPISI	0	0	0	45
NAND KAPISI	8	8	1	0
NOR KAPISI	8	8	1	0
XOR KAPISI	0	0	0	0
XNOR KAPISI	0	0	8	0
NOT KAPISI	24	12	16	129
MUX_2X1	0	0	0	16
LDCP	0	0	0	16
TOPLAM LOJİK ELEMEN SAYISI	86	58	34	354

Çizelge 4.8 Çeşitli parametrelerle karşılaştırmalı blokları

	74LS85	74LS85_MINİMİZE	74LS682	BEHAVIOURAL COMP_(IF)
Worst Net by delay (ns)	2,009	2,323	2,221	3,306
Number of slices	15	16	17	39
BELS (LUT2+LUT3+LUT4)	26	27	30	87
MUX	0	0	0	16
LDCP	0	0	0	16
4 Inputs LUTs	30	27	30	70
max.Pad To Pad Delay (ns)	14,756	15,513	14,592	14,383

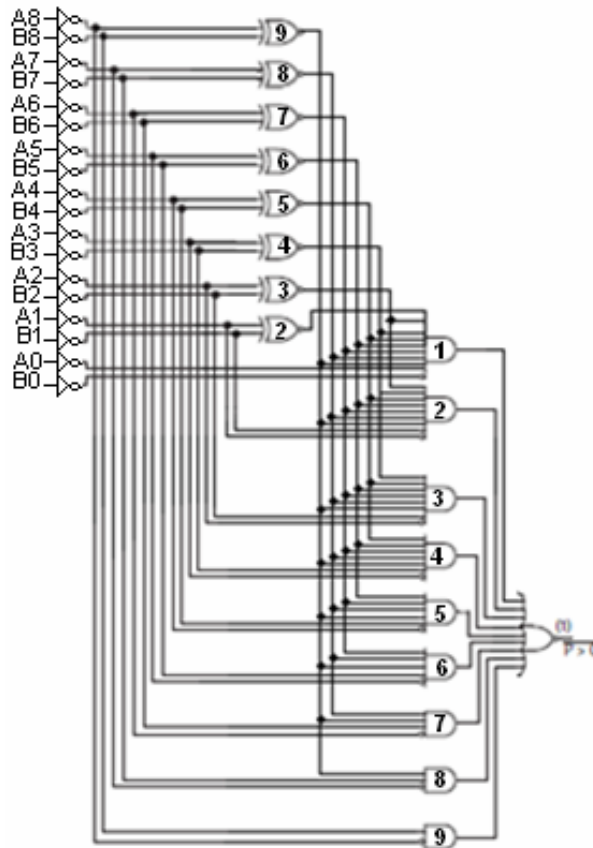
Görüldüğü gibi lojik kapı sayıları açısından birinci sırada olan 74LS682’li karşılaştırmalı yapısı zaman diyagramı açısından da diğer iki lojik bloğa göre daha hızlı, davranışsal karşılaştırmalı bloğu ile beraber en hızlı yapılardır. Bu noktada 74LS682’li karşılaştırmalı bloğunun davranışsal karşılaştırmalının yerine tercih edilmesinin nedeni ise daha çok donanımsal dilde yapıyı oluşturmak ve mümkün olduğunca davranışsal bloklardan uzak durmaktır.

4.5 Tezde Kullanılan 9-bit Karşılaştırıcı Yapısı

Tercih edilen ve tezde kullanılan karşılaştırıcı yapısının temeli 74LS682 tümleşik devresi oluşturur. Fakat ihtiyaç duyulan karşılaştırıcı bloğun 9-bit iki sayıyı karşılaştırması gerekmektedir. Ana projede veri yapısı 9-bit ile temsil edildiğinden ve gerçekleştirilen bu tez çalışması da o proje kapsamında bir modülü oluşturacağından yapının diğer modüllerle uyumlu olması, sistem ile bütünleşebilmesi için gerçekleştirilen medyan filtre bloğunun veri yolu genişliği 8-bit yerine 9-bit olacaktır.

Bu nedenle 74LS682 8-bit lojik karşılaştırma bloğu amaca uygun olarak modifiye edilmiş, gereksiz (kullanılmayacak olan uçlarla ilgili lojik elemanlar) kapı elemanları kombinezonsal devreden çıkarılmış, 9-bit karşılaştırıcı bloğu için ilave olarak lojik kapılar eklenmiştir.

Şekil 4.28’de bahsedilen değişiklikler yapılmış olan 9-bit iki sayıyı büyüklük bakımından karşılaştıran devrenin lojik yapısını görülmektedir.



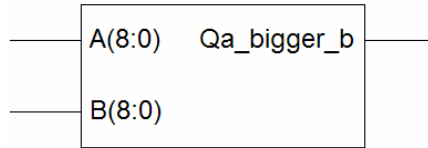
Şekil 4.28 9-bit karşılaştırıcının lojik devre yapısı

Çizelge 4.9 9-bit karşılaştırıcının doğruluk tablosu

INPUTS			OUTPUTS	
DATA	ENABLES			
P, Q	$\overline{G}$, $\overline{GT}$	$\overline{GZ}$	$P=Q$	$P>Q$
P = Q	L	L	L	H
P > Q	L	L	H	L
P < Q	L	L	H	H
X	H	H	H	H

H = HIGH Level, L = LOW Level, X = Irrelevant

Bloğun girişi iki adet 9-bit sayı, çıkışı ise bir tane bir bitlik bir sinyaldir. Çizelge 4.9'daki doğruluk tablosundan da yararlanarak, iki giriş iki çıkıştan oluşan karşılaştırıcı bloğu gerçekleştirir.

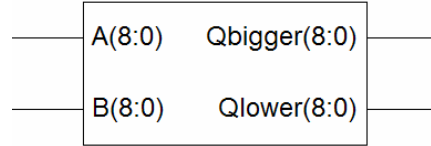


Şekil 4.29 9-bit karşılaştırıcının blok gösterimi

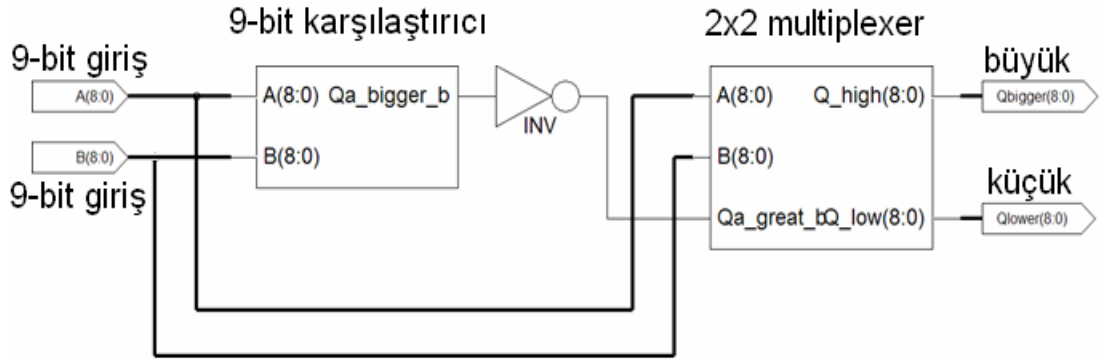
Bu yapıya ilave olarak bazı lojik devre elemanları ilave edilmiş, iki giriş tek çıkışlı karşılaştırıcı yapı, iki giriş iki çıkışlı karşılaştırma bloğuna dönüştürülmüştür.

Amaç, girişte bulunan 9-bit sayıları büyük veya küçük olma durumlarına göre sıraya dizmektir. Şekil 4.29'daki blok sayıları karşılaştırmakta ve sonucu çıkışa bir bit olarak vermektedir. Çizelge 4.9'de gösterildiği gibi girişteki sayılardan $P>Q$ ise çıkış "L", $P<Q$ ise çıkış "H" olmaktadır.

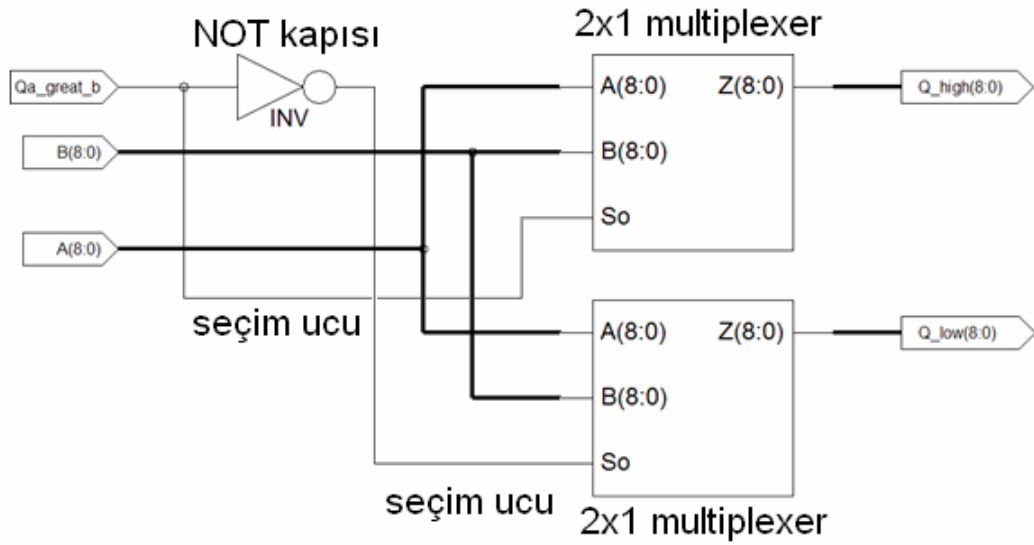
Şekil 4.29'daki bloğun çıkışına ilave elemanlar eklenerek elde edilen iki 9-bit giriş, iki 9-bit çıkışlı karşılaştırıcı blok Şekil 4.30'daki gibi tanımlanabilir.



Şekil 4.30 İki giriş-iki çıkış 9-bit karşılaştırıcının blok gösterimi



Şekil 4.31 İki giriş-iki çıkış 9-bit karşılaştırıcının iç blok gösterimi



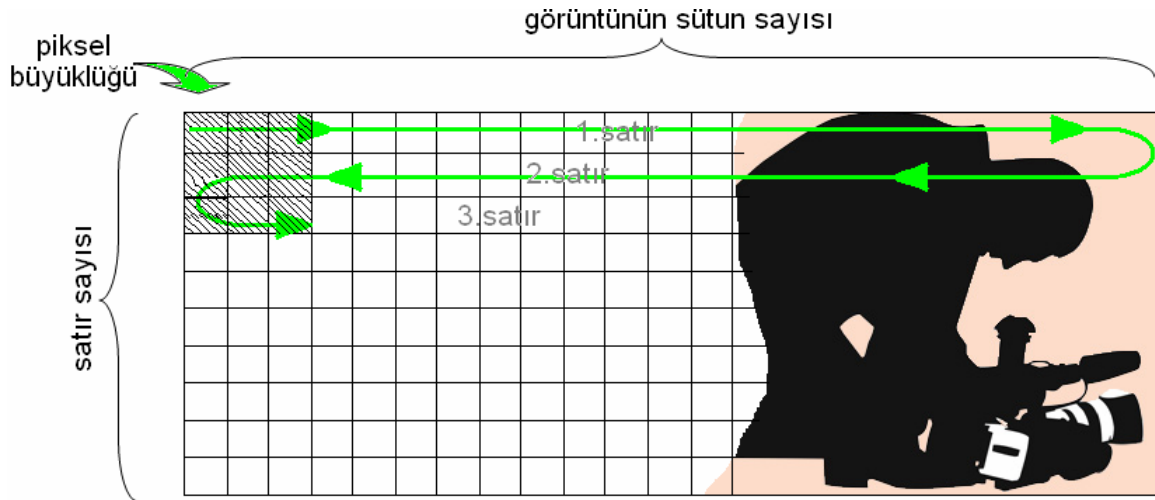
Şekil 4.32 İki giriş-iki çıkış çoğullayıcı iç blok gösterimi

9-bit iki girişli karşılaştırıcı düğüm noktasının detaylı şeması Ek 2’de verilmiştir.

4.6 Piksel Değerlerinin Filtre Şablonundan Okunması

Görüntüyü oluşturan piksel değerlerinin medyan şablonundan okunabilmesi için RAM bölgelerine ihtiyaç duyulur. Kullanılan medyan filtre şablonu 3x3 kare olduğundan şablonun işlem yapabilmesi için birbiri ardına üç satırda bulunan piksel değerlerine ihtiyaç duyar. İlk iki satırda bulunan tüm piksel değerlerinin ve üçüncü satırın ilk üç piksel değerinin bir RAM bölgesinde saklanması gerekmektedir.

Giriş görüntüsünün çözünürlüğüne bağlı olarak kullanılan FPGA’de bulunan RAM bölgelerinin büyüklüğü değişir. Uygulamada gerçekleştirilen yapı 1280x720 yüksek çözünürlükteki (720p HD Ready) gerçek zamanlı gri seviyeli görüntüyü filtrelemektedir. 3x3’lük filtre şablonunun görüntü üzerindeki piksel değerlerini alabilmesi için iki adet 1280x9 bit büyüklüğünde ram bölgelerine ihtiyaç duyulur. Elde edilen bu RAM yapısı sayesinde 3x3 şablonun piksel değerleri elde edilip bu değerler medyan filtre yapısının girişine uygulanır (Ahmad ve Sundararajan 1988).



Şekil 4.33 Medyan filtre şablonundaki piksel değerlerinin okunması

4.6.1 RAM Yapıları

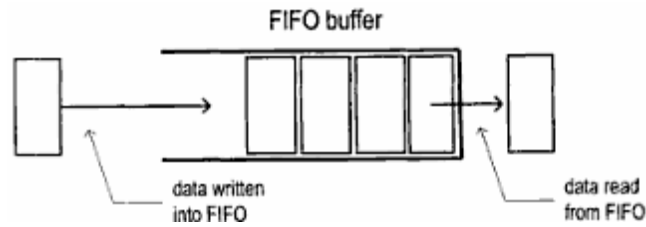
FPGA’lerde ayrı olarak tasarlanmış RAM bölgeleri yer almaktadır. Bu RAM bölgeleri Block RAM olarak adlandırılır ve her model FPGA de sayıları ve büyüklükleri farklıdır. Bir de dağılmış (Distributed) RAM adı verilen FPGA yapısının içerisinde derleyici ve sentezleyici

tarafından, yazılan vhdl koduna bağlı olarak RAM bölgeleri de oluşturulabilir.

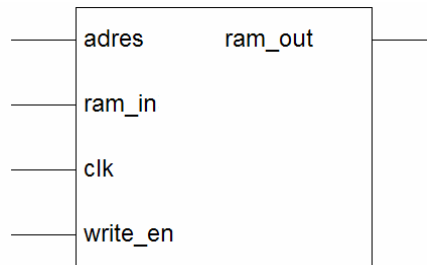
Block RAM'den farkları FPGA içerisinde adından da anlaşılacağı üzere dağınık bir şekilde sentezlenmesidir. Tanımlanan RAM bölgeleri derleme ve sentezleme esnasında FPGA içerisinde yerleşim açısından neresi uygunsa oraya yerleştirilir. Hatta birbirlerinden dağınık bir şekilde de bulunabilir. Bu da FPGA'yi ekonomik kullanmak isteyen biri için kontrolü zor bir parametredir. Uygulama bir proje olduğundan ve üzerinde çalışılan yapı da bu projenin bir modül bloğu olduğundan mümkün olduğunca yazılan kodlarda FPGA yapısını en ekonomik kullanmak amaç edinilmiştir. Bu nedenle kullanılan RAM yapıları Block RAM olarak seçilmiştir.

4.6.2 FIFO (First In First Out) Buffer

Şekil 4.34'deki buffer yapısı Block RAM'ler ile gerçekleştirilmiştir. Yapıda bir Ram_in, clk, write_en, adres girişleri, ram_out çıkışı yer almaktadır. Yapı, her bir clk (saat darbesiyle) darbesiyle ram_in girişinde bulunan veriyi adres giriş uçları ile bildirilen ram bölgesinin içine, o ram bölgesi içinde bulunan veriyi ise ram_out çıkışına aktarır (Pong, 1959).



Şekil 4.34 Fifo buffer (Pong, 1959)

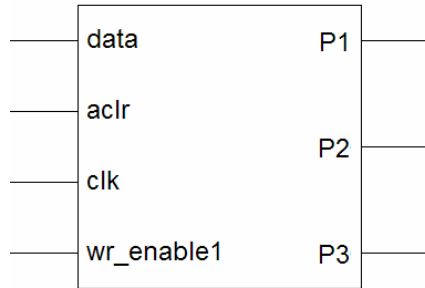


Şekil 4.35 Block RAM blok gösterimi

Genel amaçlı olarak kullanılan bir yapıdır. Uygulamalar özelleştikçe yapılar da uygulamaya özgü hale getirilir. Bu uygulamada görüntü pikselleri medyan filtre bloğuna üçerli olarak üç saat darbesiyle alınır. Medyan filtrelemede kullanılan ana blok iş hattı mantığıyla oluşturulmuş medyan filtre bloğu olduğundan, gerçekleştirilen FIFO Buffer da bu bloğu sürebilecek yapıda tasarlanmıştır (Pong, 1959).

4.6.3 İş Hattı Mantığıyla Oluşturulmuş Medyan Filtre Yapısı İçin Piksel Değerlerinin Şablondan Okunması

Görüntüyü oluşturan piksel değerleri, tasarlanan medyan bloğun data girişi (9-bitlik bir port) üzerinden seri bir şekilde bloğun içine alınır. Bloğun girişine gelen bu seri datalar, 3x3 kare medyan filtre şablonunun değerleri haline gelebilmesi için blok içerisinde Block RAM bölgelerine ihtiyaç duyulur. Yukarıdaki bölümlerde bahsedildiği üzere tasarlanan yapı içerisinde her bir satır için, piksel değerlerini içerisinde barındıran Block RAM bölgelerinin yer alması gerekir. Görüntünün birinci satırı için bir, ikinci satırı için ise ikinci bir Block RAM yapısı kullanılmıştır.



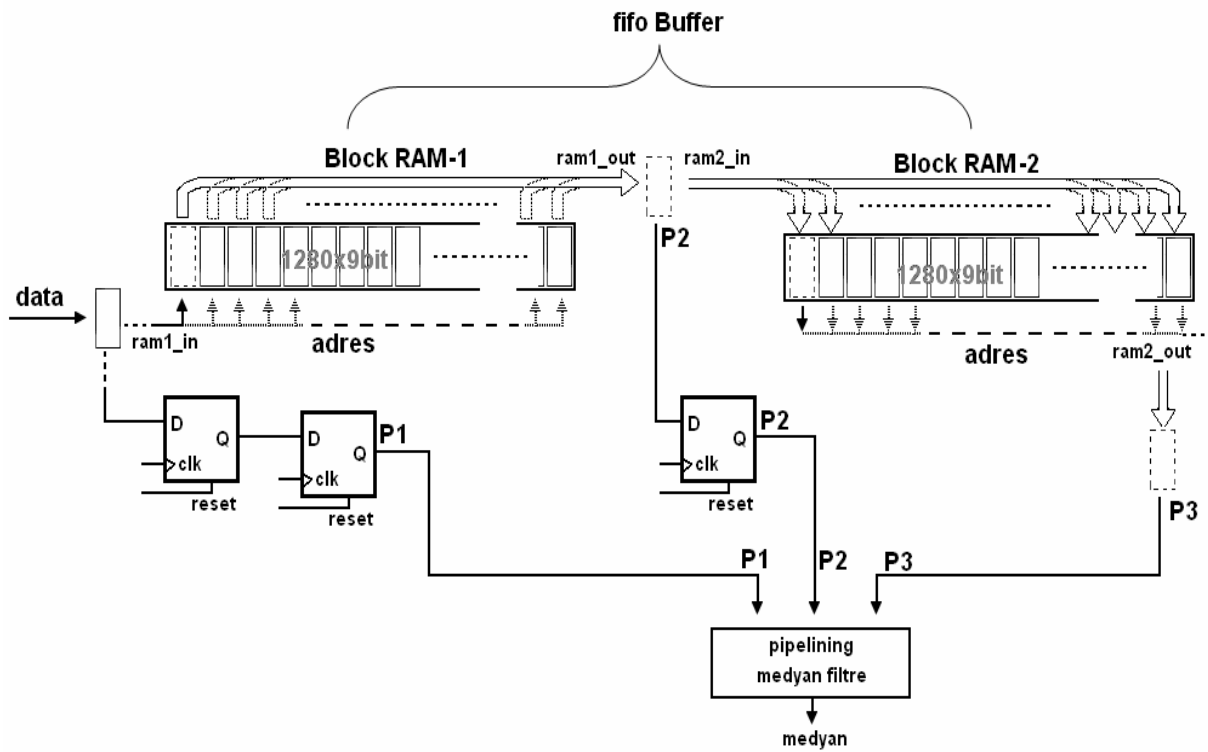
Şekil 4.36 Fifo buffer blok gösterimi

Bloğun çıkışları P1, P2 ve P3, iş hattı mantığıyla oluşturulmuş medyan bloğu girişlerine göre ayarlanmıştır. P1, P2 ve P3 çıkışları, Gürültülü giriş görüntüsünün filtrelemede kullanılacak olan 3x3 kare medyan filtre şablonunun piksel büyüklükleri iş hattı mantığıyla oluşturulmuş medyan filtre bloğunun girişlerine uygulanır.

Şekil 4.37'deki yapının çalışma prensibi kısaca şu şekildedir. Read First modunda çalışan RAM yapısında, data girişine her bir saat darbesiyle gelen görüntünün piksel büyüklükleri raml_in sinyaline atanır. Block RAM-1'in adres yazmacının göstermiş olduğu RAM

bölgesinde bulunan piksel değeri saat darbesiyle ram1_out sinyaline atanarak ilgili RAM bölgesi yeni data yazılması için boşaltılmış olur. Böylelikle ram1_in sinyaline gelen yeni piksel değeri o adres yazmacı ile gösterilen RAM bölgesine yazılır. ram1_out sinyaline atanan bir önceki piksel değeri de Block RAM-2 de ram2_in sinyaline iletilmiş olur.

Sonraki saat darbesiyle beraber Block RAM2'nin adres yazmacının göstermiş olduğu yerde bulunan piksel büyüklüğü de ram2_out sinyaline atanarak ilgili ram bölgesi boşaltılır, takip eden saat darbesiyle de ram2_in deki data ilgili ram bölgesine yazılmış olur. Bu işlem her bir saat darbesiyle adres yazmaçları bir arttırılarak devam ettirilir.



Şekil 4.37 Fifo buffer blok içyapısı

Kontrol devreleriyle RAM bölgelerinin adres yazmaçları ram blokların girişlerine sürekli olarak gelen dataları ilgili adreslerdeki ram bölgelerine yazar, çıkışları da P1, P2 ve P3 uçlarına atayarak bloğun düzenli şekilde çalışmasını sağlar.

Örneğin, görüntünün ikinci satırında bulunan 15. piksel değeri Block RAM-1 bölgesinin 15. bölgesine yazılırken, o bölgede bulunan birinci satırın 15. piksel değeri de Block RAM-2 bölgesinin 15. bölgesine yazılır. Bu bölgede bulunan bir önceki satıra ait piksel büyüklüğü de P3 çıkışına aktarılır. Böylelikle amaçlanan, görüntünün satır genişliği büyüklüğünde iki adet

RAM bölgesini birbiri ardına bağlayarak, saat darbesiyle çalışan bir shift register (kayan yazmaç) elde etmektir.

Yukarıda anlatılanlar aşağıdaki örnekle açıklanırsa Şekil 4.38'deki gibi 15x15 çözünürlüğünde bir giriş görüntüsü, fifo buffer yapısının data girişine her bir saat darbesiyle bir piksel büyüklüğü gelecek şekilde uygulanır. Sonrasında bloğun çıkış uçları P1, P2, P3 ün durumlarını ve bunlarla birlikte Blok RAM bölgelerinin saat darbesiyle nasıl değiştikleri incelenir.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100	101	102	103	104	105
106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
121	122	123	124	125	126	127	128	129	130	131	132	133	134	135
136	137	138	139	140	141	142	143	144	145	146	147	148	149	150
151	152	153	154	155	156	157	158	159	160	161	162	163	164	165
166	167	168	169	170	171	172	173	174	175	176	177	178	179	180
181	182	183	184	185	186	187	188	189	190	191	192	193	194	195
196	197	198	199	200	201	202	203	204	205	206	207	208	209	210
211	212	213	214	215	216	217	218	219	220	221	222	223	224	225

Şekil 4.38 15x15 boyutunda bir görüntü matrisi

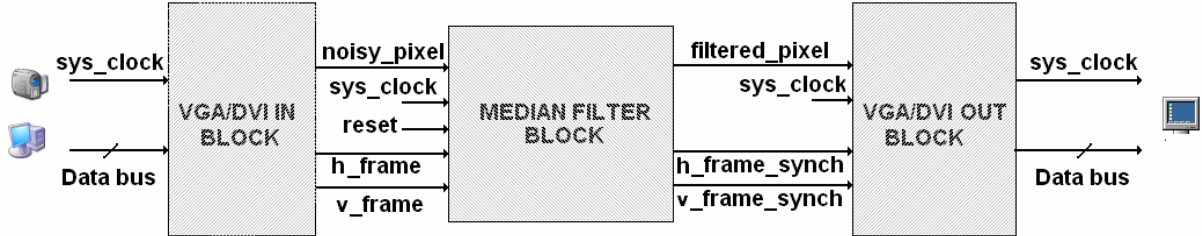
Bu görüntü modelsim simülasyonunda bloğun girişine uygulanmış, çıkışların ve RAM yapıların durumu incelenmiş sonuçlar Ek 3'te grafik olarak verilmiştir.

4.7 Medyan Filtre Bloğu

Ana proje kapsamında olan medyan filtre modülü, görüntü işleme üzerine birçok farklı görevi gerçekleştiren sistemin elemanlarından yalnızca bir tanesidir. Dış dünyaya açılan, direkt olarak giriş/çıkış birimlerine doğrudan bağlı (webcam, kamera, bilgisayar, vb.) bir yapı değildir. Bu yapı, sistemin içerisinde kullanılmış, ihtiyaç duyduğu girişleri kendisinden bir önceki modülden alıp, aynı şekilde işleyip çıkışa aktardığı bilgileri de kendinden sonraki

farklı bir görevi yerine getiren bloğa iletecek bir iç bloktur. Görevi itibariyle, dış dünyadan alınan görüntülerin ilk uğrak noktasıdır.

Tasarlanan modül görüntü işaretlerinin çeşitli açılardan ele alınıp işlenebilmesi için görüntü üzerindeki medyan filtrelemeyi gerçekleştirir.

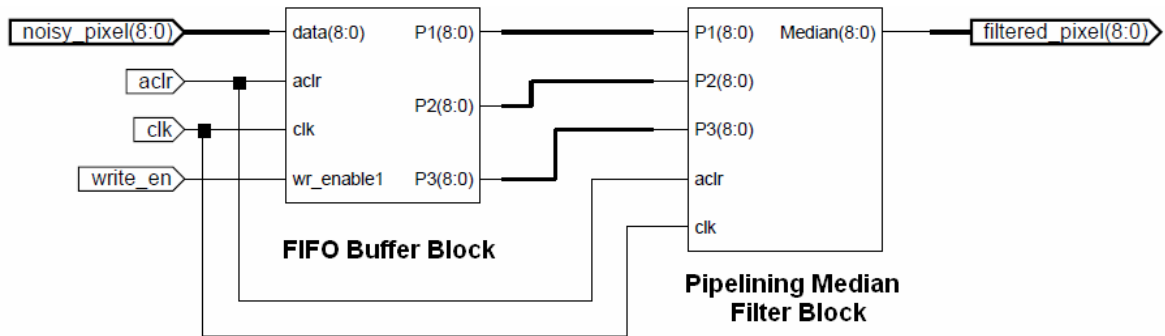


Şekil 4.39 Medyan filtre test blok diyagramı

Şekil 4.39’da tasarlanan medyan bloğunun test edileceği yapı blok şeması halinde verilmiştir.

Bu bloğu test etmenin en basit yolu dış dünyadan alınan verilerin VGA/DVI_in gibi bir blokta medyan filtre bloğunun anlayacağı giriş sinyallerine çevrilmesi, aynı şekilde medyan filtre bloğunun ürettiği çıkış sinyallerinin ise bir VGA/DVI_out gibi bir bloktan geçirilerek sinyallerin VGA/DVI protokolüne çevrilmesidir. Sonrasında da sinyaller çıkıştaki monitöre iletilir.

Medyan filtre bloğu esas itibariyle Şekil 4.40’da gösterildiği gibi iki ana bloktan meydana gelir.



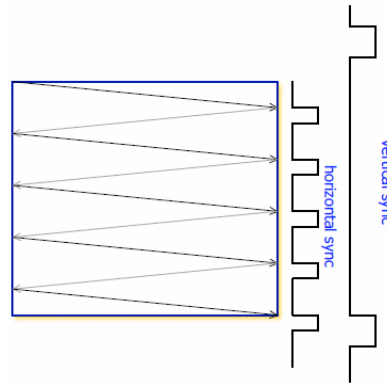
Şekil 4.40 Medyan filtre bloğu

Bunlardan bir tanesi FIFO Buffer bloğu, diğeri ise iş hattı mantığıyla oluşturulmuş medyan filtre bloğudur. Bu iki blok yukarıdaki bölümlerde detaylı bir şekilde anlatılmıştır.

4.7.1 VGA Video Sinyalleri

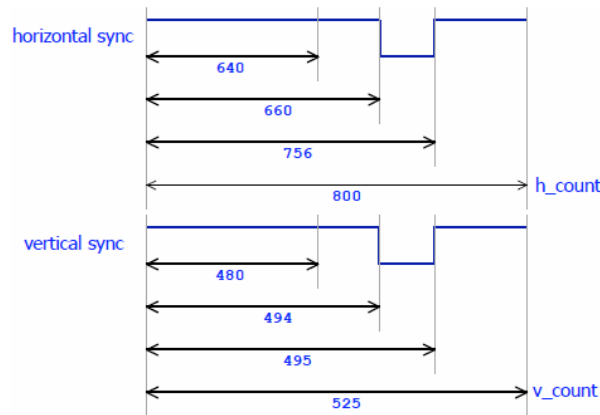
VGA video sinyalleri 5 adet sinyalden oluşur. Bunlar, horizontal sync ve vertical sync video senkronizasyonu için kullanılan sayısal sinyaller, Red (R), Green (G), Blue (B) Renk bilgilerini taşıyan analog sinyaller (0-0.7V) olarak tanımlanır.

RGB sinyallerinde, analog gerilim seviyelerindeki değişim diğer renkleri oluşturur. Bu tez çalışmasında, sistemin içerisinde RGB sinyallerinin yerine bunların belli katsayılarla çarpılıp toplanmasından oluşan gri seviyeli görüntüler kullanılacaktır (Sanchez, 2010).



Şekil 4.41 Horizontal sync ve vertical sync sinyalleri (Sanchez, 2010)

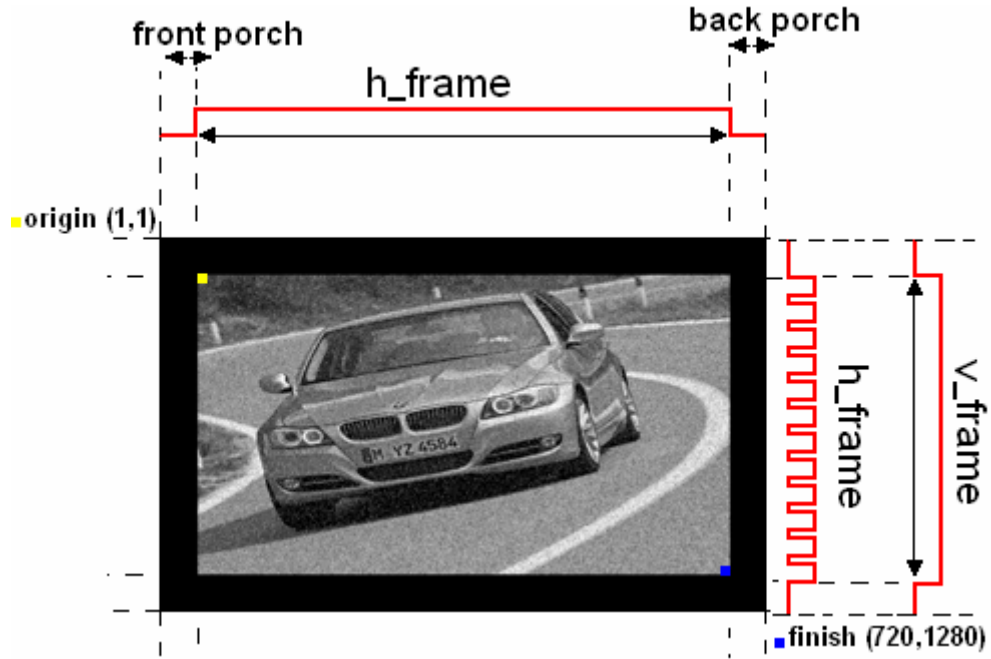
Şekil 4.41’de görüntü üzerinde, horizontal sync ve vertical sync sinyallerinin değişimi görülmektedir.



Şekil 4.42 640x480 çözünürlükte VGA sinyalleri (Sanchez, 2010)

Uygulamada böyle karışık bir sinyal dizisi kullanmak yerine işlemleri kolaylaştırmak, proje

genelinde bakıldığında sistemi oluşturan her bir blok arasında uyum sağlamak açısından karmaşık sinyal takımını basite indirmek için aşağıda açıklandığı gibi sinyal yapısı kullanmak projenin bütünlüğünü sağlamak açısından daha uygun olacaktır (Şekil 4.43).



Şekil 4.43 Medyan filtre bloğunda kullanılan h_frame ve v_frame sinyalleri

Medyan filtre bloğunda standart VGA sinyalleri horizontal sync ve vertical sync yerine Şekil 4.43'deki h_frame ve v_frame sinyalleri kullanılmaktadır.

h_frame=0 v_frame=0	h_frame=1 v_frame=0	h_frame=0 v_frame=0
h_frame=0 v_frame=1	Görünür Bölge h_frame =1 v_frame=1	h_frame=0 v_frame=1
h_frame=0 v_frame=0	h_frame=1 v_frame=0	h_frame=0 v_frame=0

Şekil 4.44 Görüntü üzerinde h_frame, v_frame sinyallerinin durumları

Bu sinyallerin özellikleri, görüntü işareti üzerinde görünür bölgelerde veya görünür olmayan

bölgelerdeki durumları Şekil 4.44'te gösterildiği gibidir.

Özetle h_frame ve v_frame sinyallerinin ikisinin “1” olduğu bölge, ekrana görünür bölge olarak gönderilmektedir.

Medyan filtre bloğu çalışması için h_frame ve v_frame sinyallerine ihtiyaç duyulduğundan, bloğun girişinde VGA sinyallerini bu iki sinyale dönüştürmek, bloğun çıkışında ise yine h_frame ve v_frame sinyallerini VGA sinyallerine dönüştürmek için ilave bloklar kullanılır.

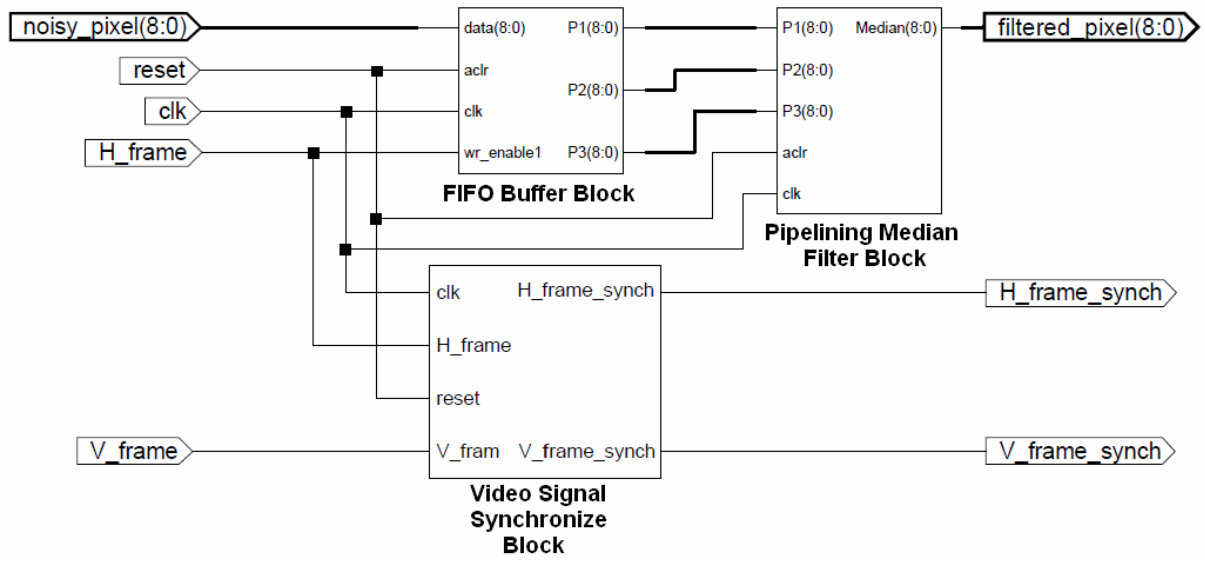
Şekil 4.39'daki düzeneğe yardımıyla tasarlanan Medyan filtre bloğu test edilecek olursa, düzeneğin çıkışında bulunan monitör ekranında girişteki monitördekinden farklı olarak gecikmeli bir görüntü oluşacaktır. Monitör ekranları arasında oluşan piksellerdeki bu gecikmeler ekranların asenkron olmasına sebep olur. Gecikmenin temel nedeni, tasarlanan devrenin çalışmadaki gecikmesinden kaynaklanır. Yapı içerisindeki RAM bölgeleri ve ardışıl elamanların yer alması söz konusu gecikmeyi kaçınılmaz kılar.

Piksellerdeki bu gecikmeler düzeneğin arasına yerleştirilen medyan filtre bloğundan kaynaklanır. Medyan filtre bloğunun içerisinde yer alan iki adet birbirlerine kaskad bağlı Block RAM'ler gecikmenin temel nedenidir. Buna ilave olarak bir de iş hattı mantığıyla oluşturulmuş medyan filtrenin kendi gecikmesini de hesaba katmak gerekir.

Ek 4'te yapının girişine gelen gürültülü görüntü işaretleri ve bu işaretlere karşı devrenin ürettiği medyan piksel değerleri yer almaktadır. Bu şemada gecikme miktarı(*) çok rahat görülmektedir.

Medyan filtre bloğunun çıkışında oluşan bu gecikmeyi ortadan kaldırmak için bloğun girişine gelen v_frame ve h_frame video sinyalleri blok içerisinde oluşan gecikmeyi ortadan kaldıracak şekilde geciktirilerek çıkışa v_frame_synch ve h_frame_synch sinyalleri altında aktarılır. Böylece medyan filtre bloğuna birde bu işi gerçekleştirmek üzere video sinyal senkronizasyon bloğu eklenmiştir.

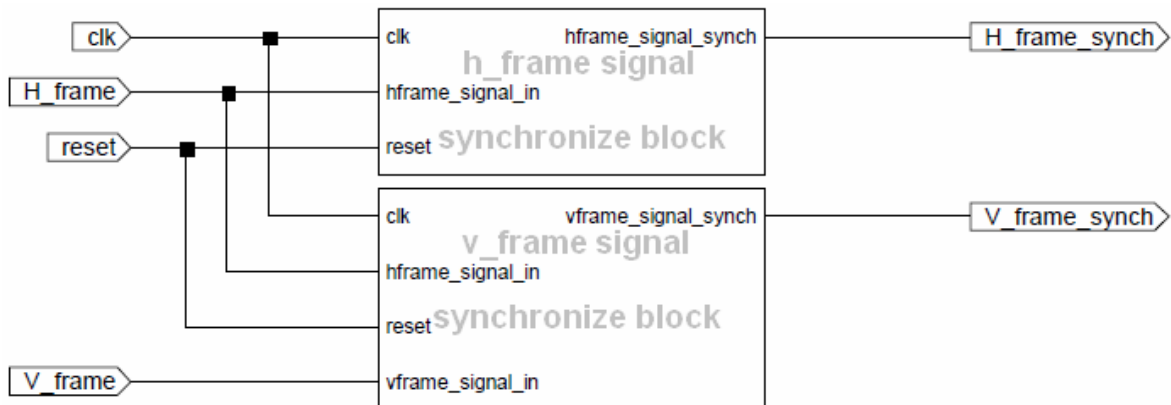
* Gecikme miktarı, giriş görüntüsünün genişliğiyle ya da sütun sayısı ile değişir. Ek 4'te HD Ready 720x1280 çözünürlüğe uygun olarak tasarlanmış Medyan filtre bloğu için analiz yapılmıştır.



Şekil 4.45 Medyan filtre bloğu

4.7.2 Video Sinyalleri Senkronizasyon Bloğu

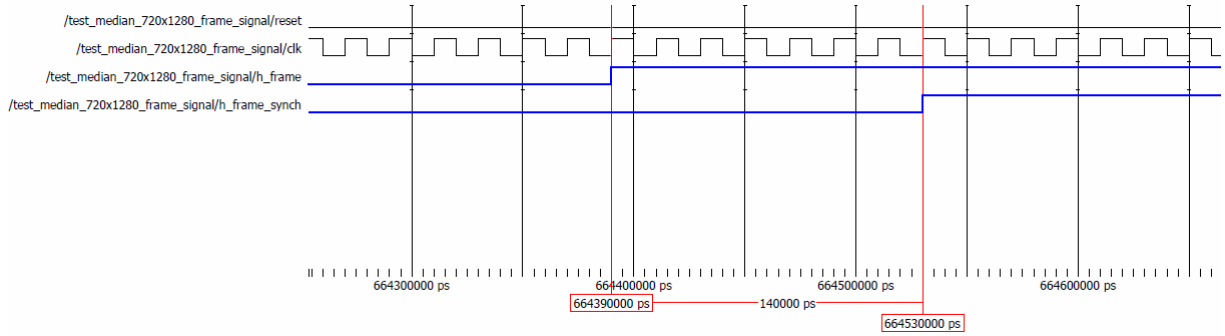
Video sinyalleri senkronizasyon bloğu, `h_frame` ve `v_frame` video sinyallerinin Medyan filtre bloğunun gecikmesine bağlı olarak yeniden ayarlanıp sırasıyla, `h_frame_synch` ve `v_frame_synch` isimlerini aldığı bloktur. Medyan filtre bloğundan bağımsız işlem yapan blok sadece bu iki sinyali bloğun gecikmesi kadar geciktirip çıkışa aktarır.



Şekil 4.46 Video sinyalleri senkronizasyon bloğu içyapısı

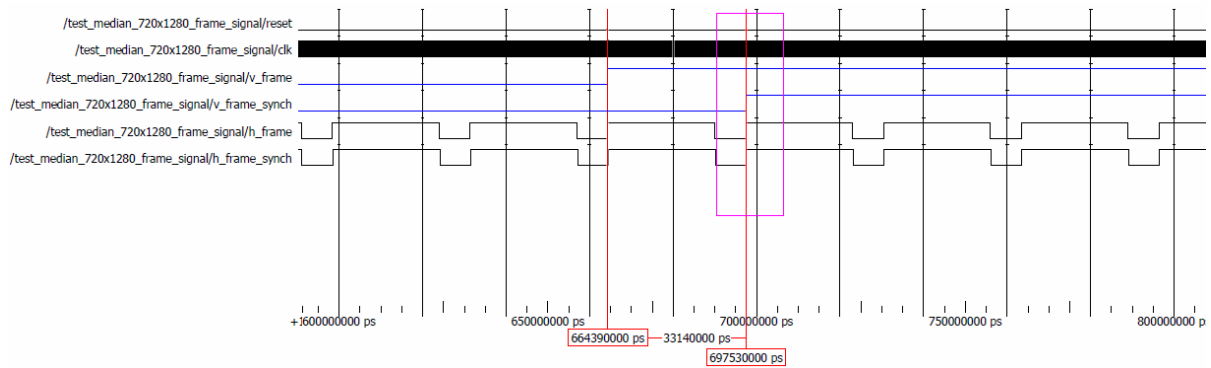
Kendi içerisinde iki bloktan oluşur. Bunlardan bir tanesi h_frame sinyal senkronize bloğu diğeri de v_frame sinyal senkronize bloğudur.

h_frame sinyal senkronize bloğu, h_frame sinyalini 7 piksel clock geciktirir. Bunun için sonlu durum makineleri (FSM-Final State Machine) yapısından yararlanır.

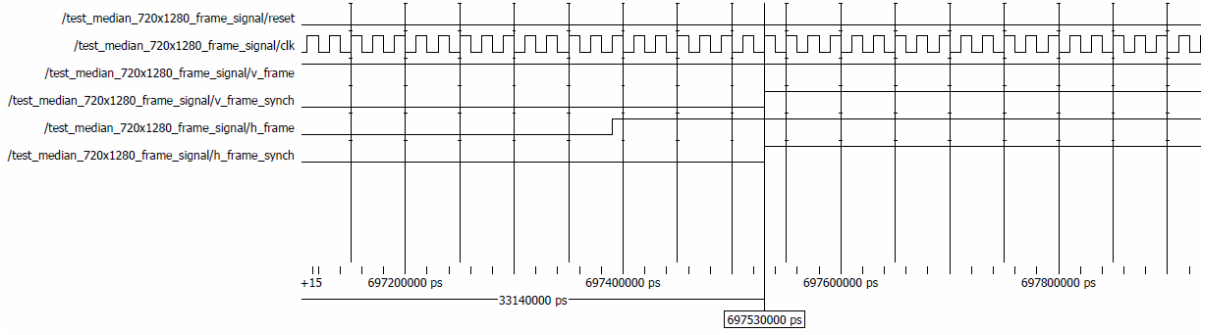


Şekil 4.47 H_frame_synch sinyalinin oluşumu

V_frame sinyal senkronize bloğu ise v_frame sinyalini bir görüntü satır genişliği ve h_frame gecikmesi toplamı kadar geciktirir. Buradaki yapıda da sonlu durum makinelerinden yararlanılmıştır. V_frame sinyalini lojik “1” olarak yakaladıktan sonra blok, bu sefer h_frame sinyalinin lojik “1” olmasını beklemektedir. H_frame sinyalini lojik “1” olarak yakaladıktan sonra da blok bu sefer h_frame gecikmesi kadar bir gecikme uygulayarak v_frame_synch sinyalini üretir (Şekil 4.49).



Şekil 4.48 V_frame_synch sinyalinin oluşumu-1



Şekil 4.49 V_frame_synch sinyalinin oluşumu-2

4.7.3 Medyan Filtre Bloğunun Gürültü Bastırmadaki Performansı

Tezin en başından beri gürültülü görüntü işaretlerinin temizlenmesinde medyan filtrenin kullanıldığından bahsedilmiştir. Bu bölümde ise tasarlanan 3x3 kare medyan filtre şablonlu medyan filtre bloğunun gürültü bastırmadaki performansı ele alınmıştır (Vijaykumar vd., 2009). Tasarlanan bu medyan filtre modülü hangi oranda gürültülü görüntü işaretlerinde maksimum performansı vermektedir. Görüntü işaretleri üzerinde bu tür kıyaslama yapabilmek için aşağıdaki iki terim hakkında bilgi sahibi olmak gerekir. Bunlardan biri ortalama karesel hata (MSE-Mean Squared Error) diğeri ise darbe sinyal gürültü oranıdır (PSNR-Peak Signal Noise Ratio) (Vijaykumar vd., 2009).

MSE (Mean Squared Error):

Ortalama Karesel Hata: Genellikle iki büyüklük, iki vektör arasındaki farkı, benzerliği ortaya çıkarmak, göstermek için kullanılan hesaplamalardan biridir. Görüntü işlemede oldukça sık kullanılan popüler bir terimdir. Görüntü işareti de iki boyutlu bir vektör olduğundan MSE değerinin hesaplanması da oldukça kolaydır.

İki görüntü işareti için hesaplanan MSE değeri ne kadar küçük ise iki vektör arasındaki benzerlikte o kadar fazladır. Hesaplamalarda kullanılacak iki görüntüden biri gürültülü görüntü işareti, diğeri de medyan filtre bloğu çıkışından elde edilen filtrelenmiş görüntü işaretidir. m ve n görüntü işaretinin boyutlarını göstermektedir.

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I_1(i,j) - I_2(i,j)]^2 \quad (4.1)$$

PSNR (Peak Signal Noise Ratio):

Darbe İşaret Gürültü Oranı: İki görüntü işareti arasındaki sinyal gürültü oranını hesaplamada kullanılır. İşaret işlemede sık kullanılan bir terimdir. Hesaplanması (4.2) denklemiyle(*) gösterildiği gibidir.

Hesaplanan PSNR değeri ne kadar büyük ise karşılaştırılan görüntü işaretleri o oranda benzer demektir.

$$PSNR = 10 \log_{10} \frac{R^2}{MSE} \quad (4.2)$$

Şekil 4.50'deki görüntü işaretine çeşitli oranlarda tuz-biber gürültüsü eklenmiş ve her bir durumdaki gürültülü görüntü işaretleri medyan filtre bloğundan geçirilmiş, elde edilen filtrelenmiş görüntü işaretleri üzerinde MSE ve PSNR oranları incelenmiştir (Çizelge 4.10). Çizelge 4.10'dan hareketle tasarlanan medyan filtre bloğunun gürültü bastırmadaki performansı incelenmiştir (Vijaykumar vd., 2009).

Çizelgede 4.10'da görüldüğü üzere görüntü işareti üzerinde gürültü oranı arttıkça tasarlanan Medyan filtre bloğu istenildiği oranda temizleme yapamamaktadır. Bunu PSNR oranlarının gürültü oranı arttıkça azaldığından anlamak mümkündür. Tipik olarak 3x3 kare medyan şablonlu Medyan filtre bloğu, şablonun içerisine yerleşen dokuz adet piksel değerinin yarıdan fazlası gürültü işareti ise, iş görmez.

Çünkü elde edilen medyan değeri de yine bir gürültülü piksel olduğundan filtreleme sonuç vermemektedir. Medyan şablonunun boyutu arttırıldığında ise görüntü bulurlaşmaya başlayarak görüntü üzerindeki detaylar kaybolur. Bu gibi durumda amaca uygun farklı tipte çalışan filtre yapıları kullanmak daha etkili sonuç vermektedir (Vijaykumar vd., 2009).

* 3.7 denkleminde yer alan R değeri incelenen görüntü gri seviyeli ise 255 sabit değerini almaktadır. Çalışmada yapılan hesaplamalarda bu değer kullanılmıştır..



Şekil 4.50 Enigma, 720x1280 çözünürlükte görüntü işareti(*)



Şekil 4.51 %10 oranında gürültülü görüntü işareti(**)

* 1280x720 çözünürlüğünde olan Şekil 4.50 gri seviyeli görüntü A4 sayfa boyutundan büyük olduğundan boyutu 60% oranında küçültülmüştür.

** 1280x720 çözünürlüğünde olan Şekil 4.51 gri seviyeli görüntü A4 sayfa boyutundan büyük olduğundan boyutu 60% oranında küçültülmüştür.



Şekil 4.52 %10 oranında gürültülü görüntü işaretinin medyan filtreden geçirilmiş hali(*)

(4.1)'daki eşitlikte belirtilen MSE değeri hesaplandıktan sonra (4.2)'deki PSNR değerinin hesaplanabilmesi için R değerinin bilinmesi gerekmektedir. Çalışmamızda incelediğimiz gri seviyeli görüntü [0-255] aralığında olduğundan hesaplamalarda R değeri 255 alınmalıdır.

Orijinal görüntü-gürültülü görüntü ve orijinal görüntü-filtrelenmiş görüntü işaretleri için MSE değeri Matlab programı kullanılarak hesaplanmış ve bu hesaplanan değerler kullanılarak her bir görüntü çifti için PSNR değerleri aşağıdaki gibi hesaplanmıştır.

Orijinal görüntü ile %10 gürültülü görüntü işareti için PSNR değeri için;

$$PSNR = 10 \log_{10} \frac{255^2}{2574} \quad (4.3)$$

$$= 10 \log_{10} \frac{65025}{2574} = 10 \log_{10} 25,262 = 10 \times 1,4024 = 14,024 dB \quad (4.4)$$

* 1280x720 çözünürlüğünde olan Şekil 4.52 gri seviyeli görüntü A4 sayfa boyutundan büyük olduğundan boyutu 55% oranında küçültülmüştür.

Orijinal görüntü ile %20 gürültülü görüntü işareti için PSNR değeri için;

$$PSNR = 10 \log_{10} \frac{255^2}{4812} \quad (4.5)$$

$$= 10 \log_{10} \frac{65025}{4812} = 10 \log_{10} 13,513 = 10 \times 1,1308 = 11,308 \text{ dB} \quad (4.6)$$

Orijinal görüntü ile %50 gürültülü görüntü işareti için PSNR değeri için;

$$PSNR = 10 \log_{10} \frac{255^2}{9810} \quad (4.7)$$

$$= 10 \log_{10} \frac{65025}{9810} = 10 \log_{10} 6,628 = 10 \times 0,8214 = 8,214 \text{ dB} \quad (4.8)$$

Orijinal görüntü ile %80 gürültülü görüntü işareti için PSNR değeri için;

$$PSNR = 10 \log_{10} \frac{255^2}{12870} \quad (4.9)$$

$$= 10 \log_{10} \frac{65025}{12870} = 10 \log_{10} 5,052 = 10 \times 0,7034 = 7,034 \text{ dB} \quad (4.10)$$

şeklinde hesaplanır.

Diğer görüntü çiftleri içinde benzer hesaplama yöntemi kullanılarak Çizelge 4.10 elde edilmiştir.

Çizelge 4.10 Orijinal, gürültülü ve filtrelenmiş görüntüler arasında MSE ve PSNR oranları

Orijinal Görüntü & Gürültülü Görüntü				
	%10 Gürültülü	%20 Gürültülü	%50 Gürültülü	%80 Gürültülü
MSE	2574	4812	9810	12870
PSNR	14,024 dB	11,308 dB	8,214 dB	7,034 dB

Orijinal Görüntü & Filtrelenmiş Görüntü				
	%10 Gürültülü	%20 Gürültülü	%50 Gürültülü	%80 Gürültülü
MSE	42,042	95,92	855,01	2543
PSNR	31,893 dB	28,312 dB	18,811 dB	14,078 dB

5. SONUÇLAR

Bu tezde görüntü temizleme uygulamalarında oldukça sık kullanılan, etkili sonuç veren lineer olmayan medyan filtre gerçekleştirilmesi yapılmıştır. Bu filtreleme tekniğinin hangi tip gürültülü işaretlerde uygulanacağı, nasıl bir sonuç vereceği, ne şekilde bir filtre şablonuyla kullanılacağı anlatılmış, literatürde kullanılan medyan filtreleme şablonları hakkında bilgiler verilmiş, bu çalışmada kullanılan 3x3 kare medyan şablonuyla karşılaştırmaları yapılmıştır. Medyan filtreleme tekniğini görüntü üzerinde uygularken, FPGA platformundan yararlanılmıştır. Çalışmalar öncelikle sabit bir görüntü üzerinde Matlab ve Modelsim programları kullanılarak çalışmanın yol haritası belirlenmiş, buradan elde edilen sonuçlarla gerçekleştirilecek donanım temel oluşturulmuştur.

Üzerinde çalışılan görüntü tipi, gerçek zamanlı 720p HD Ready diye adlandırılan 1280x720 çözünürlüklü görüntü işaretleridir(*).

FPGA üzerinde gerçekleştirilecek olan medyan filtre bloğunda asıl iş, kullanılacak olan medyan filtre şablonunu belirlemektir. Kullanılacak filtre şablonu seçilirken temizlenecek olan görüntü işaretinin hangi amaçla kullanılacağı dikkat edilecek en önemli husustur. Kullanım amacına uygun seçilen filtre şablonu gürültülü görüntü işareti üzerinde gezdirilir. Soldan sağa, en üst satırdan başlanarak görüntünün sonu olarak adlandırılan en alt satır en son piksel elemanına kadar yapılan gezdirme işleminde, filtre şablonu üzerine denk gelen piksel değerleri sahip oldukları gri seviyelerine göre küçükten büyüğe sıralanır ve bu piksel dizisinin orta elemanı bulunur. Medyan değer olarak ta adlandırılan bu piksel büyüklüğü, gürültülü görüntü işaretinde filtre şablonunun merkezine denk gelen piksel büyüklüğüyle yer değiştirilir. Bu işlem görüntü üzerinde bulunan her bir piksel için tekrarlanır. Filtre bloğunun hedef giriş işaretinin de gerçek zamanlı yüksek çözünürlüklü bir görüntü olduğu düşünülürse bu filtre bloğunun hızının saniyede minimum 60 resim çerçevesi işleyeceği (her bir resim çerçevesinde 720p HD ready görüntü işareti için 1280x720 adet piksel) düşünülürse, saniyede milyonlar mertebesinde işlem yapabilmesi gerekmektedir. Bu nedenle söz konusu işi sıradan yazılım algoritmalarıyla sıralı işlem olarak aynı performansı almak imkânsızdır. Bu nedenle günümüz teknolojisinde paralel işlem yapabilen, programlanabilen lojik kapılardan oluşan FPGA'ler ile donanım gerçekleştirme yoluna gidilmiştir.

* Unix notasyonu: modeline "720p" 74,25 1280 1390 1430 1650 720 725 730 750 (1280 yataydaki görünür bölge uzunluğu, 720 düşeydeki görünür bölge uzunluğu, 1390-1430 arası hsync aktif, 725-730 arası vsync aktif).

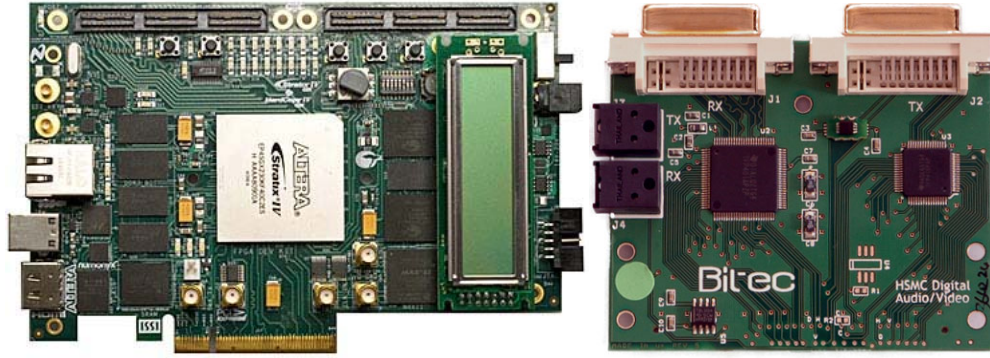
Asıl yapılan iş şablon içerisindeki piksel büyüklüklerini sıraya koyup orta elemanını bulmak olduğundan, piksel büyüklüklerini karşılaştıracak olan karşılaştırmacı bloğunu belirlemek ve belirlenen modülü FPGA platformunda tasarlamak, problemi yarı yarıya çözmek demektir. Çalışmanın ilk başlarında, dört farklı yapıda karşılaştırmacı blok üzerinde çalışılmıştır. Bunlardan biri vhdl yazım dili ile yazılan behavioural karşılaştırma, diğerleri ise donanım ile karşılaştırma yapan kombinezonsal yapılar olmuştur. Her biri detaylı bir şekilde Bölüm 4.4'te anlatılmıştır. 74LS682 8-bit iki sayıyı karşılaştıran ve sonucu bir bit ile çıkışa aktaran lojik tümleşik devre, medyan filtrelemenin gerçekleştirilmesinde karşılaştırmacı modül olarak kullanılmıştır. Projede kullanılan veri yolu genişliği 9-bit olarak düşünüldüğü için bu yapıya ilave lojik kapılar eklenmiş, karşılaştırmacı bloğu 9-bite göre yeniden düzenlenmiştir. Bu karşılaştırmacı modül yapısını kullanarak iki farklı medyan filtre bloğu üzerinde çalışma yapılmıştır. Bunlardan biri, kullanılan 3x3 kare medyan şablonun içerisindeki 9 adet piksel büyüklüğünü tek seferde bloğun içine alarak işlem yapan ve ortanca elemanı bulan paralel girişli medyan bloğudur. İkincisi ise 9 adet piksel büyüklüğünün 3'erli olarak 3 saat darbesiyle alıp işleyen ve medyan değerini hesaplayan iş hattı mantığıyla oluşturulmuş medyan bloğudur. Literatür taramasında iş hatlarından oluşmuş medyan filtrelerin yüksek çözünürlükte çalışan uygulamalarına rastlanmaması nedeniyle iş hattı yapısının kullanılması tercih edilmiştir.

İş hattı mantığıyla oluşturulmuş medyan filtre bloğunun dışında bu tez çalışmasında kullanılan Block RAM ve sinyal senkronizasyon modülleri, tasarlanan yapının Modelsim üzerinde denenip test edilebilmesi için kullanılmıştır.

Tasarlanan medyan bloğu Altera firmasının üretmiş olduğu gelişmiş bir FPGA olan Altera Stratix IV GX EP4SGX230KF40C2N'ün bulunduğu Stratix IV GX FPGA Development Kit üzerinde denenmiş ve iş hattı mantığıyla oluşturulmuş medyan modülün 720p çözünürlükte başarıyla çalıştığı monitör üzerinden gözlenmiştir. Sentezleme için Altera FPGA'leriyle uyumlu olan Quartus II 9.1 SP2 yazılımı kullanılmıştır.

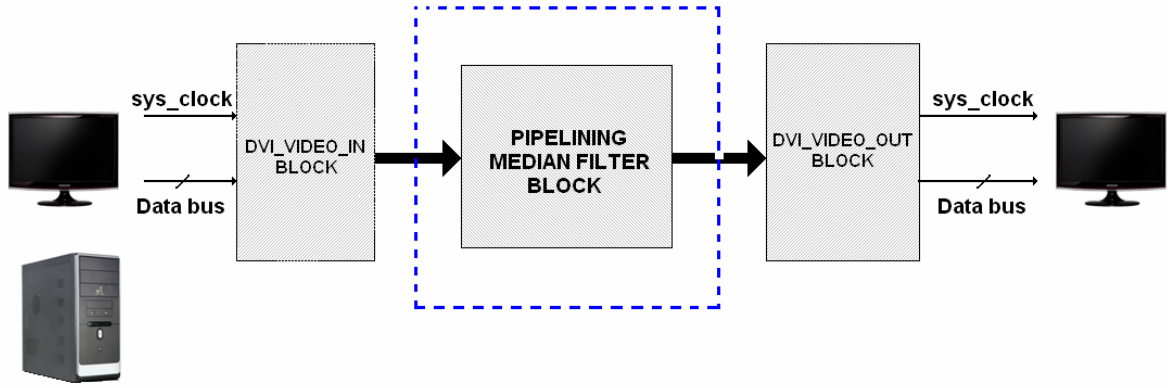


Şekil 5.1 Altera firmasına ait stratix IV fpga



Şekil 5.2 Stratix IVgx fpga development kiti ve hsmc_dvi video giriş/çıkış kartı

Bu tezde 108E023 numaralı TÜBİTAK projesi kapsamında yürütülen, proje modüllerinden biri olan ve filtreleme modülü adı altında medyan filtre modülü gerçekleştirilmiştir. Bu geniş proje kapsamında tasarlanmış olan diğer modüllerle birlikte kullanılabilirliği sağlamak amacıyla medyan filtre bloğu standart giriş ve çıkışları oluşturan DVI giriş çıkış arabirim modüllerinin ortasına yerleştirilerek test edilmiştir.



Şekil 5.3 Gerçeklenecek medyan filtre bloğu test blok diyagramı

Yapıda, 243 kombinezonsal ALUT, 313 özel amaçlı lojik yazmaç ve 4 x 9Kbit Block RAM kullanılmıştır. Projede tasarlanacak diğer modüller de düşünüldüğünde medyan filtre modülünün donanım olarak FPGA üzerinde fazla bir alan kullanmadığı görülmektedir. 1280x720 çözünürlükteki 720p gerçek zamanlı görüntü için iş hattı mantığıyla oluşturulmuş medyan filtre sorunsuz bir şekilde çalışmaktadır. Toplam piksel sayısı yatayda 1650, düşeyde 750 pikseldir. Ekrandan saniyede geçen görüntü sayısı ise 60'tır. Bu da sistemin çalışması

için gereken minimum çalışma frekansını belirler. Bu çözünürlükteki gerçek zamanlı görüntü işleyen modülün çalışma hızı 74,25MHz ($1650 \times 750 \times 60 = 74,25$ Mpiksel/saniye)'dir. Tasarladığımız bu blok, bu çalışma frekansında sorunsuz bir şekilde çalışabilmektedir.

Tasarlanan blok, gürültü bastırma performansı açısından incelenecek olursa görüntü işareti üzerinde gürültü oranı arttıkça medyan filtre bloğu istenildiği oranda temizleme yapamamaktadır. Bu sonuç PSNR oranlarının gürültü oranı arttıkça azalmasından da gözlenebilir. Genel olarak 3x3 kare medyan şablonlu içerisine yerleşen dokuz adet piksel değerinin yarısından fazlası gürültü işareti ise iş görmez. Elde edilen medyan değeri yine bir gürültülü piksel değeri olmaktadır*. Medyan şablonunun boyutu arttırıldığında ise bu sefer de görüntü karikatürize olup resim üzerindeki detaylar kaybolmaya başlar. Bu gibi durumda amaca uygun farklı filtre yapıları kullanmak daha etkili sonuçlar verebilir.

* Eğer 3x3 kare medyan şablonun 9 adet pikselinden 4'ü tuz gürültüsü, 4'üde biber gürültüsü ise bu durumda medyan değeri 9. piksel olacaktır. Dokuzuncu piksel gürültülü piksel değildir. Bu yapılan genellemenin dışında bir durum olarak düşünülmelidir.

KAYNAKLAR

- Abreu E., Lightstone M., Mitra S. K. ve Arakawa K. (1996), "A New Efficient Approach For The Removal of Impulse Noise From Highly Corrupted Images," IEEE Trans. Image Processing, vol. 5, pp. 1012–1025.
- Acharya T. ve Ray A.K. (2005), Image Processing Principles and Applications, A John Wiley & Sons, ISBN-13 978-0-471-71998-4
- Ahmad M. O. ve Sundararajan D. (1988), "Parallel Implementation of a Median Filtering Algorithm", Int. Symp. on Signals and Systems.
- Astola J. ve Kuosmanen P. (1997), "Fundamentals of Nonlinear Digital Filtering", Boca Raton, FL: CRC.
- Bates GL. ve Nooshabadi S. (1997), "FPGA Implementation of A Median Filter", IEEE Region 10 Annual International Conference, Proceedings/TENCON, Vol.2, pp.437-440.
- Brownrigg D. R. K. (1984), "The Weighted Median Filter," Commun. Ass. Comput. Mach., vol. 27, no. 8, pp. 807–818.
- Castleman K. R. (1995), Digital Image Processing, Prentice Hall, ISBN: 0132114674.
- Dhanasekaran, D. ve Boopathy B.K. (2009), "High Speed Pipelined Architecture for Adaptive Median Filter", European Journal of Scientific Research, pp. 454-460
- e-bergi, ODTÜ Bilgisayar Topluluğu (2009), Pipelining Nedir?, İnternet: <http://e-bergi.com/2009/Ekim/pipelining>
- Gonzalez R.C. ve Woods R.E. (2002), Digital Image Processing, Addison-Wesley Publishing Company.
- Leon O. Chua ve Thomas R. (2002), Cellular Neural Networks and Visual Computing, Cambridge University Press, ISBN 0 521 65247 2
- Lin H. ve Willson A. N. (1988), "Median Filter With Adaptive Length," IEEE Trans. Circuits Syst., vol. 35, pp. 675–690.
- Olafzar K. (1983), "Design and Implementation of A Single Chip 1-D Median Filter", IEEE Trans. Acoustic Speech and Signal Processing, ASSP-31, pp. 1164-1168.
- Pratt W.K. (2001), Digital Image Processing, John Wiley & Sons, ISBN: 0-471-37407-5
- Pitas I. ve Venetsanopoulos A. N. (1990), "Nonlinear Digital Filters, Principles and Applications". Boston, MA: Kluwer.
- Pong, P. Chu. (1959), FPGA Prototyping by VHDL Examples, John Wiley & Sons, Inc., Hoboken, New Jersey.
- Rao V.V.B. ve Rao S.K. (1986), "A New Algorithm for Real-Time Median Filtering", IEEE Transactions on Acoustics, Speech, Processing VOL ASSP-34. NO. 6.
- Sanchez E. (2010), A VGA Display Controller, İnternet: http://lslwww.epfl.ch/pages/teaching/cours_lsl/ca_es/VGA.pdf
- Smith, John L. (1996), "Implementing Median Filters in XC4000E FPGAs", Univision Technologies Inc. Billerica, MA, the Quarterly Journal for Xilinx Programmable Logic Users, issue 23, p.16
- Sun T. ve Neuvo Y. (1994), "Detail-Preserving Median Based Filters in Image Processing," Pattern Recognit. Lett., vol. 15, pp. 341–347.
- Tavşanoğlu V. "General Image Processing Concepts & Image Sampling", Ders Notları

Toprak, A. (2006) “Bulanık Adaptif Medyan Filtresi Kullanarak Tıbbi Görüntülerdeki Darbe Gürültüsünün Bastırılması”, Gazi Üniversitesi, Ankara

Vainio O., Neuvo Y. ve Butner S. E. (1989), “A Signal Processor for Median-Based Algorithms”, IEEE Transactions on Acoustics, Speech, Processing VOL 37. NO. 9.

Vasicek Z. ve Sekanina L.(2008), “Novel Hardware Implementation of Adaptive Median Filters” 978-1-4244-2277-7/08/ IEEE, Grant Agency of the Czech Republic

Vega-Rodriguez M.A., Sanchez-Perez J.M. ve Gomez-Pulido J.A. (2002), “An FPGA-Based Implementation For Median Filter Meeting The Real-Time Requirements Of Automated Visual Inspection Systems”, Proceedings of the 10th Mediterranean Conference on Control and Automation, Lisbon, Portugal

Vijaykumar, V.R., Vanathi, P.T., Kanagasabapathy, P. ve Ebenezer, D. (2009), “Robust Statistics Based Algorithm to Remove Salt and Pepper Noise in Images” International Journal of Signal Processing 5;3

Wang Z. ve Zhang D. (1997), “Impulse Noise Detection and Removal Using Fuzzy Techniques,” Electron. Lett., vol. 33, pp. 378–379.

Wang Z. ve Zhang D. (1998), “Restoration of Impulse Noise Corrupted Image Using Long-Range Correlation,” IEEE Trans. Signal Processing Lett., vol. 5, pp. 4–8.

Xilinx Foundation Series Quick Start Guide, 1991-1997. Xilinx. Inc.

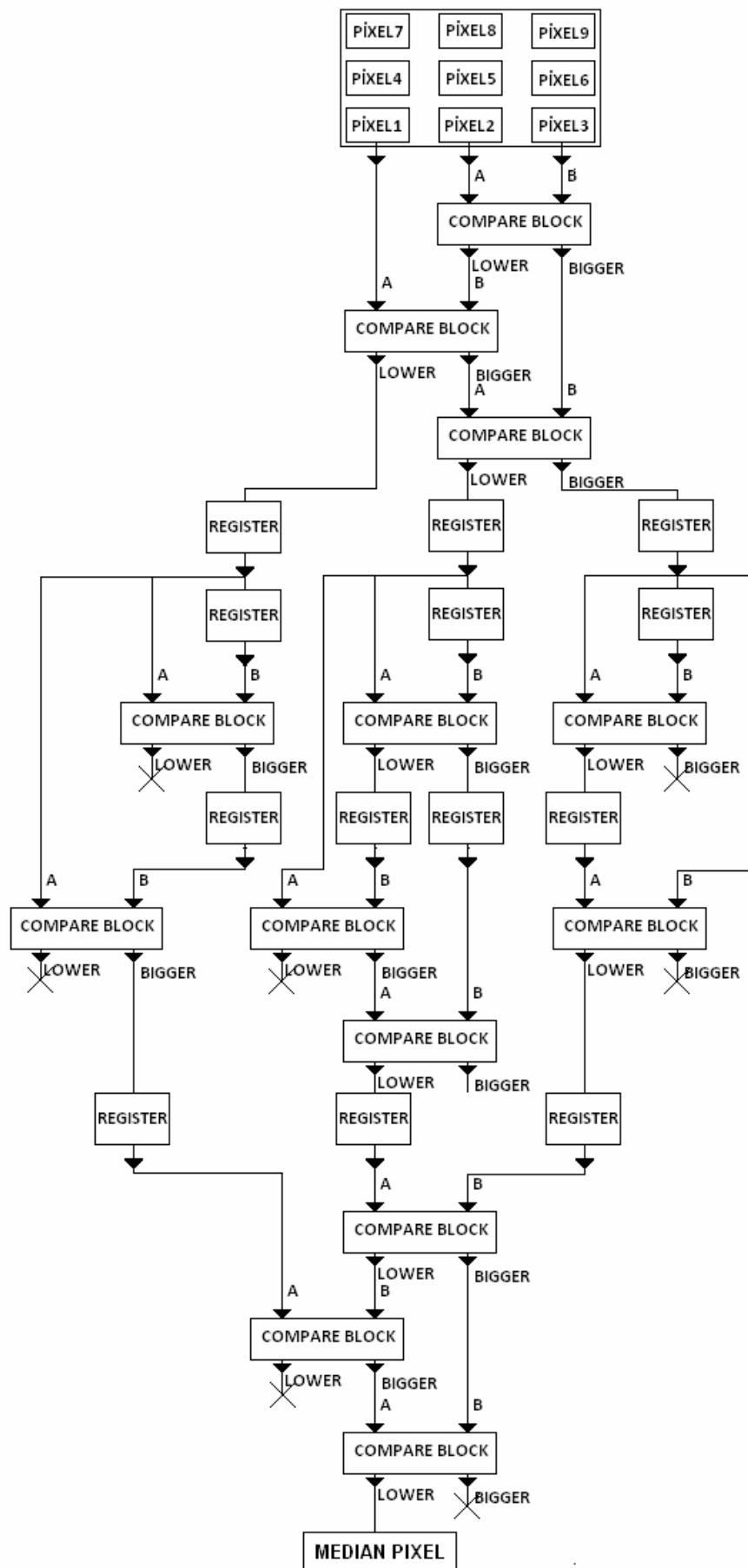
74LS85 Onsemiconductor, Alldatasheet (2010), İnternet: <http://pdf1.alldatasheet.com/datasheet-pdf/view/12664/ONSEMI/74LS85.html>

74LS682 Onsemiconductor, Alldatasheet (2010), İnternet: http://www.datasheetcatalog.org/datasheets/150/491878_DS.pdf

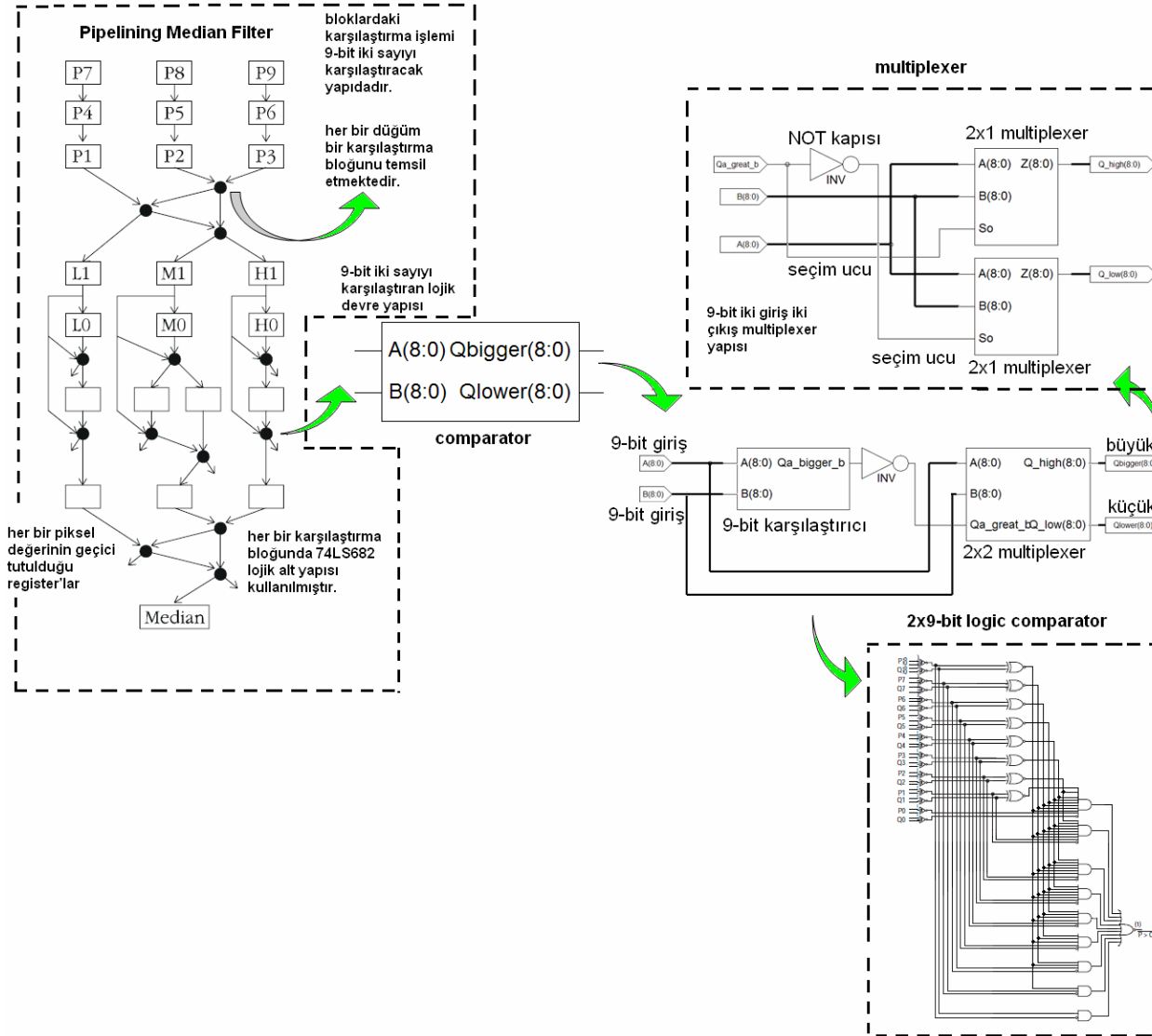
EKLER

- Ek 1 İş hattı mantığıyla oluşturulmuş 3x3 kare medyan filtre bloğunun detaylı yapısı
- Ek 2 9-bit karşılaştırıcı düğüm noktasının içyapı şeması
- Ek 3 Fifo buffer bloğu modelsim simülasyonu
- Ek 4 Medyan filtre bloğu gecikmesi (720x1280 çözünürlük)

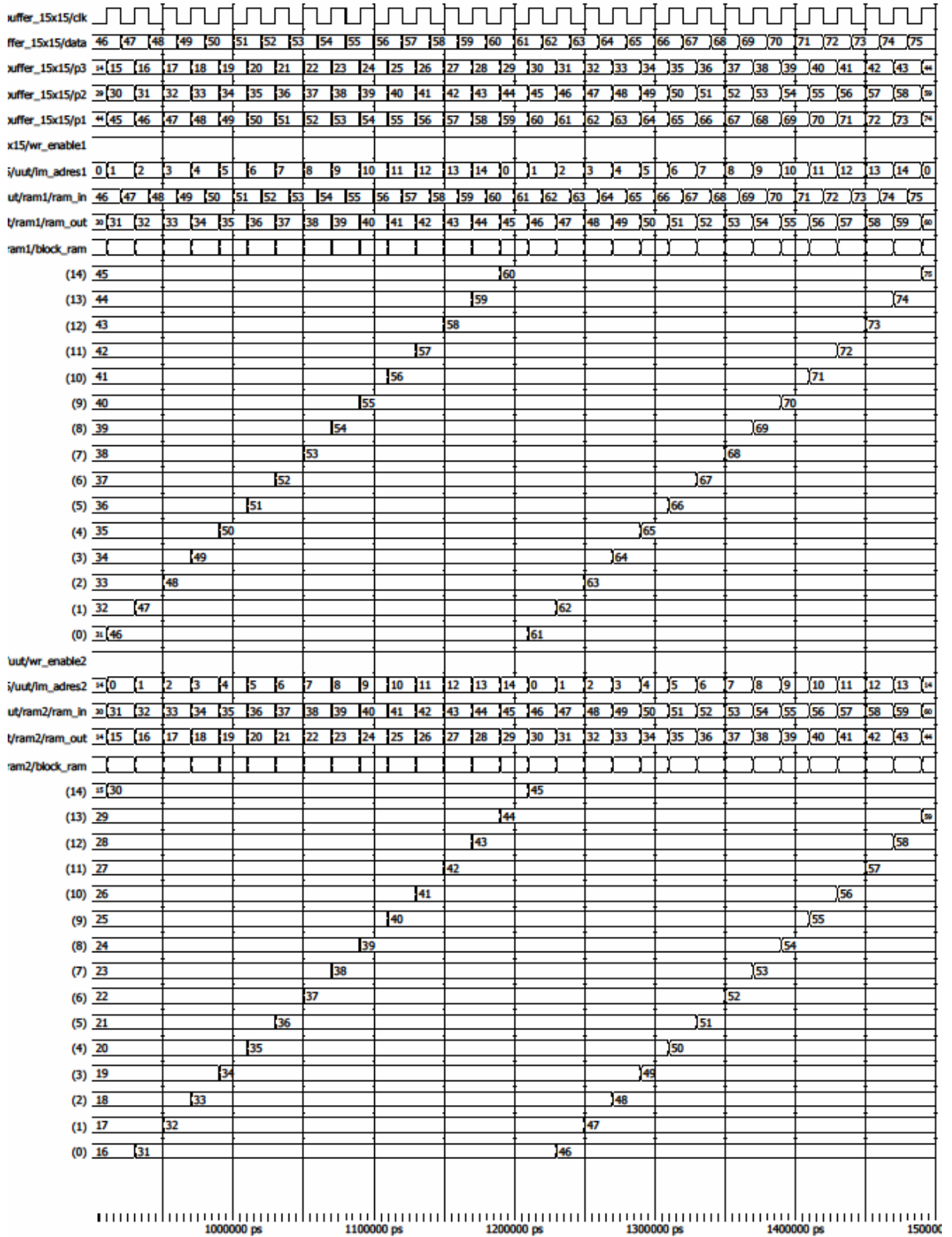
Ek 1 İş hattı mantığıyla oluşturulmuş 3x3 kare medyan filtre bloğunun detaylı yapısı



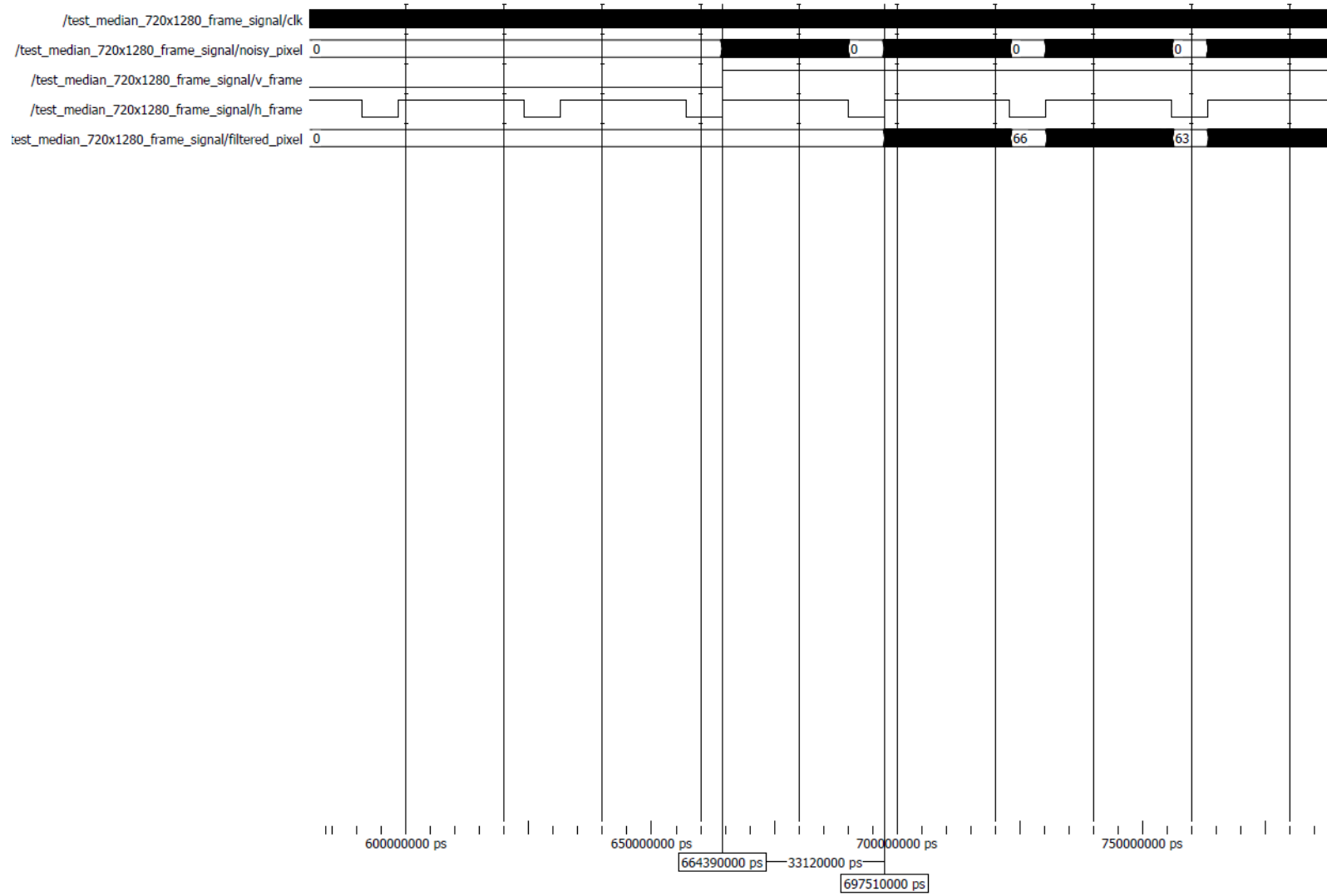
Ek 2 9-bit karşılaştırıcı düğüm noktasının içyapı şeması



Ek 3 Fifo buffer bloğu modelsim simülasyonu



Ek 4 Medyan filtre bloğu gecikmesi (720x1280 çözünürlük)



ÖZGEÇMİŞ

Doğum tarihi	20.10.1981	
Doğum yeri	Gümüşhane - Şiran	
Lise	1995-1999	Pendik Süper Lisesi
Lisans	1999-2003	Yıldız Üniversitesi Elektrik-Elektronik Fak. Elektronik-Haberleşme Mühendisliği Bölümü

Çalıştığı kurum(lar)

2004-2006	Intrade Bilgisayar ve Elektronik San. Tic. A.Ş. Elektronik Komponent Satın Alma Bölümü
2006-Devam ediyor	TUBİTAK-Ulusal Elektronik Kriptoloji Araştırma Enstitüsü, Araştırmacı