T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GERÇEK ZAMANLI STEREO GÖRME SİSTEMİNİN FPGA'DE TASARIMI VE UYGULAMASI

NİLUFAR İSAKOVA

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI

DANIŞMAN
PROF. DR. A. COŞKUN SÖNMEZ

İSTANBUL, 2012

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

GERÇEK ZAMANLI STEREO GÖRME SİSTEMİNİN FPGA'DE TASARIMI VE UYGULAMASI

Nilufar İSAKOVA tarafından hazırlanan tez çalışması 25.06.2012 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Prof. Dr. A. Coşkun SÖNMEZ

Yıldız Teknik Üniversitesi

Jüri Üyeleri

Prof. Dr. A. Coşkun SÖNMEZ

Yıldız Teknik Üniversitesi _____

Prof. Dr. Eşref ADALI

İstanbul Teknik Üniversitesi _____

Prof. Dr. Nizamettin AYDIN

Yıldız Teknik Üniversitesi _____

Bu çalışma, Yıldız Teknik Üniversitesi Bilimsel Araştırma Projeleri Koordinatörlüğü' nün 2012–04–01-YL01 numaralı projesi ile desteklenmiştir.

ÖNSÖZ

FPGA’ler düşük maliyetli olması ve tasarım sırasında kullanıcıya esneklik sağlaması sebebiyle kullanımı gittikçe yaygınlaşmaktadır. Savunma, sayısal işaret işleme, uzay, tıbbi görüntüleme ve otomotiv gibi birçok alanlarda uygulanmaktadır. FPGA'ler özellikle paralel işlem gerektiren uygulamalarda yaygın olarak kullanılmaktadır.

Stereo Görme, 3 boyutlu nesne algılama ve robotik uygulamalarında örneğin, navigasyon, yol planlama, haritalama ve lokalizasyon, gibi birçok gerçek zamanlı uygulamalarda kullanım nedeniyle Bilgisayar Görme alanında en önemli konulardan biridir. Stereo kayıklık paralel işlem gerektiren bir uygulama olduğu için gerçek zamanlı ve donanımsal olarak FPGA’de uygulanmıştır.

Bu çalışmanın gerçeklenmesi konusundaki destekleri ve cesaretlendirici tavırlarından dolayı başta tez danışmanım Prof. Dr. A. Coşkun SÖNMEZ ve Selçuk BAŞAK ve tüm Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı öğretim üyelerine sonsuz teşekkür ediyorum.

Yıldız Teknik Üniversitesi Bilimsel Araştırma Projeleri Koordinatörlüğüne bu çalışmaya desteklerinden dolayı teşekkür ederim.

Haziran, 2012

Nilufar İSAKOVA

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ 5

KISALTMA LİSTESİ 5

ŞEKİL LİSTESİ 7

ÖZET 1

ABSTRACT 2

BÖLÜM 1

GİRİŞ 3

1.1 Literatür Özeti 3
1.2 Tezin Amacı 7
1.3 Bulgular 8

BÖLÜM 2

STEREO GÖRME 9

2.1 Stereo Kayıklık Problemi 10
2.3 Stereo Kayıklık Yöntemleri 12
2.4 Önerilen Algoritma 14

BÖLÜM 3

PROGRAMNALABİLİR MANTIK 16

3.1 FPGA(Field Programmable Gate Array-Alanda Programlanabilir Kapı Dizileri) .. 18

3.2 FPGA Uygulama Alanları 22
3.3 FPGA Tasarım Süreci 24

BÖLÜM 4

DONANIM TANIMLAMA DİLLERİ 26

4.1 Verilog Donanım Tanımlama Dili 27
4.2 VHDL Donanım Tanımlama Dili 29
4.3 Quartus II Yazılım Aracı 31

BÖLÜM 5

GERÇEK ZAMANLI UYGULAMA 32

5.1 DE4 Stratix IV GX cihazı 33
5.2 Yüksek Hızlı Mezzanine Connector (HSMC) DVI Kartı 37
5.3 5-Mega Benek Dijital Kamera 40
5.4 DE4 System Builder 41

BÖLÜM 6

UYGULAMA KURULUM DETAYLARI 44

BÖLÜM 7

UYGULAMA SONUÇLARI 49

7.1 MATLAB Uygulaması 49
7.2 FPGA Uygulaması 50

BÖLÜM 8

SONUÇ VE ÖNERİLER 55

KAYNAKLAR 56

ÖZGEÇMİŞ 59

SİMGE LİSTESİ

b	Kameralar arasındaki (baseline) çizgisi
d	Kayıklık aralığı
f_c	Odak uzaklığı (focal length)
i, j	Satır Sütün(rows, columns)
$I_L I_R$	Sağ ve Sol görüntü beneği, gri tonlama değerleri (grayscale value)
μ	Görüntü penceresinin gri tonlama değerinin (grayscale value) ortalaması
P_L	Sol kameranın iz düşümü
P_R	Sağ kameranın iz düşümü
w	Pencere boyutu
$X_L Y_L Z_L$	Sol kameranın koordinate sistemleri
$X_R Y_R Z_R$	Sağ kameranın koordinate sistemleri
x_L-x_R	Kayıklık değeri (disparity value)

KISALTMA LİSTESİ

ASIC	Application Specific Integrated Circuit
CLB	Configurable Logic Block
CMOS	Complementary Metal Oxide Semiconductor
CPLD	Complex programmable Logic Device
CPU	Central Processing Unit
CSELT	Centro Studi E Laboratori Telecomunicazioni
DDR	Double Data Rate
DLL	Delay Lock Loop
DMA	Direct Memory Access
DSP	Digital Sygnal Proscessing
DTD	Donanım Tanımlama Dili
DVI	Digital Visual Interface
DVI-D	Digital Visual Interface -Digital
EEPROM	Electrically Erasable Programmable Read Only Memory
EPROM	Erasable Programmable Read Only Memory
FPGA	Field Programmable Gate Arrays
FPP	Fast Passive Paralel
HDL	Hardware Descriptive Language
HSMC	High Speed Mezzanine Card
HSTC	High-Speed Terasic connector
HSYNC	Horizontal Synchronization
IDE	Integrated Drive Electronics
IEEE	Institute of Electrical and Electronics Engineering
ISP	Instruction Set Processor
JTAG	Joint Test Action Group
KARL	Kaiserslautern Üniversitesi
LE	Logic Element
LUT	Look Up Table
LVDS	Low-Voltage Differential Signaling
MMI	Monolithic Memories Inc
PAL	Programmable Array Logic
PAR	Place and Route
PCI	Peripheral Component Interconnect

PLA	Programmable Logic Array
PLL	Phase Lock Loop
PROM	Programmable Read Only Memory
RAM	Random Access Memory
ROM	Read Only Memory
RTL	Register Transfer Level
S/D	Serialize/Deserialize
SATA	Serial Attachment
SDRAM	Synchronous Dynamic Random Access Memory
SMA	Sub Miniature version A
SO-DIMM	Small Outline Dual in Line Memory Module
SPLD	Simple Programmable Logic Device
SRAM	Static Random Access Memory
SSRAM	Synchronous Static Random Access Memory
UXGA	Ultra Extended Graphics Array
VHDL	Very High Speed Integrated Circuit Hardware Descriptive Language
VLSI	Very Large Scale Integration
VSYNC	Vertical Synchronization
ZBT	Zero BusTurnaround

ŞEKİL LİSTESİ

Sayfa

Şekil 2. 1 Parallel Kamera Lokalizasyonu 10
Şekil 2. 2 3B Stereo Kamera Lokalizasyonu 11
Şekil 3. 1 PLA Mimari Yapısı 16
Şekil 3. 2 PAL Mimari Yapısı 17
Şekil 3. 3 CPLD Mimari Yapısı 18
Şekil 3. 4 FPGA Yapısı 19
Şekil 3. 5 Mantık Blokları 20
Şekil 3. 6 FPGA Tasarım Süreci 24
Şekil 4. 1 VHDL Tasarım Blok Şeması 28
Şekil 4. 2 Verilog ile 4-bit sayaç 29
Şekil 4. 3 VHDL ile 4-bit sayaç 30
Şekil 4. 4 Quartus II Yazılım Aracı Arayüzü 31
Şekil 5. 1 DE4 FPGA geliştirme bordu 32
Şekil 5. 2 DE4 Stratix IV GX Blok Diagramı 36
Şekil 5. 3 DVI kartı 37
Şekil 5. 4 HSMC DVI Kartı Blok Diagramı 39
Şekil 5. 5 Kameranın Performans Parametreleri 40
Şekil 5. 6 Align edilmiş 5 Mega Benek Kamera 41
Şekil 5. 7 System Builder arayüzü 42
Şekil 5. 8 Bir yapı tasarımının genel tasarım akışı 43
Şekil 6. 1 Stereo Kamera ve DVI kartlı Stratix IV FPGA geliştirme bordu 44
Şekil 6. 2 Sistemin donanımsal tasarımı 45
Şekil 6. 3 Hat ve arama bufferleri 46
Şekil 6. 4 SAD'en küçüğünü bulmak için ardışık düzenli çalışan ikili ağaç kullanımı. 46
Şekil 7. 1 Middlebury stereo çiftleri (a) Teddy (b) Koniler 49
Şekil 7. 2 Matlab 3x3 pencereli (a) Teddy (b) Koniler sonucu 50
Şekil 7. 3 Matlab 8x8 pencereli (a) Teddy (b) Koniler sonucu 50
Şekil 7. 4 Sağ ve Sol kamera görüntüleri. 51
Şekil 7. 5 Sağ ve Sol kamera görüntü farkları 51
Şekil 7. 6 Kameradan 3B görüntü 51
Şekil 7. 7 FPGA 3x3 pencereli RGB görüntü 52
Şekil 7. 8 FPGA 3x3 pencereli 3B görüntü 52
Şekil 7. 9 FPGA 8x8 pencereli RGB görüntü 53
Şekil 7. 10 FPGA 8x8 pencereli 3B görüntü 53

ÖZET

GERÇEK ZAMANLI STEREO GÖRME SİSTEMİNİN FPGA'DE TASARIMI VE UYGULAMASI

Nilufar İSAKOVA

Bilgisayar Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Tez Danışmanı: Prof. Dr. A. Coşkun SÖNMEZ

Stereo eşleme probleminde 3 boyutlu ortamda bir noktanın farklı görüntü düzlemlerinde karşılık geldiği noktaları bulmaya çalışılır. Eşleşen noktaların konumları arasındaki kayıklık stereo kayıklık (disparity) değerini verir. Stereo görme, insan gözünün temel mekanizmasına benzeyen bilinen derinlik ölçme yöntemidir.

Bu tezde, gerçek zamanlı stereo görme sisteminin FPGA'de tasarlanması ve uygulanması amaçlanmıştır. Stereo görme, doğrudan derinlik ölçme özelliği ile 3 boyutlu algılama için çok önemli çalışma alanı olmakla birlikte işlemlerin donanımsal olarak gerçeklenmesi hızlılık anlamında önemli olmaktadır.

Bu çalışmada, gerçek zamanlı olarak daha fazla alt-benek hassasiyeti ile yoğun bir kayıklık görüntü sağlayan bir ardışık stereo görme sistemi tam olarak önerilmektedir. Uygunluk, stereo eşleştirme gibi tüm stereo görme süreci, herhangi bir harici cihazlara gerek kalmadan tek bir sahada programlanabilir kapı dizisi (FPGA) kullanılarak gerçekleştirilebilmektedir.

Anahtar Kelimeler: Stereo Görme, Stereo Eşleme, Kayıklık, FPGA (Field Programmable Gate Array).

ABSTRACT

FPGA DESIGN AND IMPLEMENTATION OF A REAL-TIME STEREO VISION SYSTEM

Nilufar ISAKOVA

Department of Computer Engineering

MSc. Thesis

Advisor: Prof. Dr. A. Coşkun SÖNMEZ

Stereo correspondence is the problem of discovering the closest possible match between two images captured simultaneously from cameras in different spatial locations. The differences between the positions of the matching points are the stereo disparity value. Stereo vision is a well-known ranging method because it resembles the basic mechanism of the human eye.

This thesis describes the implementation and design of the real-time stereo vision system on FPGA. Stereo vision is one of the most important in the field of depth measuring in 3D feature detection and it is important to implement it in hardware to achieve high speed operations.

This work proposes a fully pipelined stereo vision system providing a dense disparity image with additional sub-pixel accuracy in real-time. The entire stereo vision process, such as correspondence, stereo matching is realized using a single field programmable gate array (FPGA) without the necessity of any external devices.

Keywords: Stereo Vision, Stereo Matching, Disparity, FPGA (Field Programmable Gate Array).

YILDIZ TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE

BÖLÜM 1

GİRİŞ

1.1 Literatür Özeti

Literatürde gerçek zamanlı ve donanımsal olarak engel tanıma ve algılama problemi için pek çok çalışmalar ele alınmıştır[1,2,3,4]. Stereo görüntü işleme sisteminin gerçek zamanlı uygulamasının yavaşlığı dezavantajlı olmasına rağmen akademik ve ticari olarak birçok çalışma bulunmaktadır. Bu çalışmada Stereo Görme sisteminin FPGA (Field Programmable Gate Arrays)'de Mutlak Farklar Toplamı (SAD) algoritması kullanarak donanımsal uygulanması ele alınmıştır.

Literatürde FPGA'de donanımsal olarak Mutlak Farklar Toplamı ve daha birçok algortimalar ile ilgili akademik ve ticari çalışmalar yapılmıştır.

S.Lee et al[18] notunda değiştirilmiş Mutlak Farklar Toplamı hesaplama sürümü kullanılmıştır. Mutlak Farklar Toplamı algoritmalarının çeşitli sürümleri kaynak gereksinimleri ve performansını belirlemek için sentezlenmiştir. Sütun ve satır Mutlak Farklar Toplamı hesaplaması bufferler kullanan bir Mutlak Farklar Toplamı korelatör ayrıştırarak, FPGA cihazı kaynak kullanımı açısından % 50 civarında bir tasarruf elde edilmiştir. Ayrıca, pencereleri dikdörtgen olanları yerine eşleşen farklı şekiller kullanarak, kaliteden taviz vermeden depolama gereksinimleri azaltılarak, saniyede 122 kare hızında kayıklık haritaları, 64 kayıklık seviyeleri ile 320 x 240 benek uzaysal çözünürlüğü (spatial resolution) bir görüntü çifti üretilmiştir.

Se. S et al[1] bildirisinde deniz altı ortamların 3B haritasını çıkartmakta mobil stereo kameradan bilinmeyen ortamın hızlı fotogerçekçi renkli 3B kalibre üretme, anlık sahne

geliştirme modeli kullanılmış. Robot gezerken ortamın fotogerçekçil 3B modelini ve ölçumünü yapabilmiştir.

Divyang et al[4] çalışmasında elde edilen kayıklık harita kalitesini artırmak için görüntü düzeltme (image rectification) ve sağ-sol tutarlılık denetimi gerçekleştirmek için dört FPGA kullanılmıştır. Gerçek zamanlı hızda bir görüntü çifti için 640 x 480 benek, saniyede 30 kareye ulaşan 128 kayıklık seviyeleri elde edilmiştir. Günümüzde yeniden yapılandırılabilir (reconfigurable) mantık kullanarak makine görmesi (machine vision) yöntemlerinin çoğunluğu ham performans (raw performance)'a odaklı, boyut ve güce (size & power) değil. Bu nokta da, mobil otonom uygulamalarında kritik bir durum söz konusudur. Özellikle, güç tüketimi büyük ölçüde görme araştırma alanlarında ihmal edilen bir noktadır.

Literatür çalışmalarında, gerçek zamanlı olarak daha fazla alt-benek hassasiyeti ile yoğun kayıklık bilgi verebilir sayım dönüşümü (census transform) kullanarak bir FPGA tabanlı stereo görme sistemi geliştirilmiştir. Önerilen sistemin bütün (pre ve post-processing) önce ve sonra işlemleri gibi doğrultma, Sağ-Sol kontrol (LR-check) gibi fonksiyonları ve teklik testi dahil olmak üzere tek bir FPGA üzerinde uygulanmıştır.

S.Jin et al[2] çalışmasında hedeflenen performans ve esneklik elde etmek için, önerilen stereo görme sistemi özel veri yolunu tasarımı sırasında ardışık ve birime yoğun kullanımı üzerine odaklanmıştır, geleneksel bilgisayar uygulamasına kıyasla 230 kat hızlı çalışmaktadır.

M.Albeto et al[3] bildirisinde sayım dönüşüme (census transform) dayalı yoğun bir stereo görme algoritmasının yeniden yapılandırılabilir (reconfigurability) mimari uygulaması ele alınmıştır. Sayım dönüşümü algoritmayı analiz ederek, boyutu ve erişim belleği azaltılarak, dolayısıyla da gecikme süresi azaltılmış olur. Bu en beklenen durumlardandır ki az hafıza gerektiren ve hızlı yapılabilen işlemlerdir yani işlemlerin paralelleşmesidir. Bu sistemde çok ölçekli (mullti-scale) ve çok-merkezli (multi-oriented) bir yaklaşım kullanarak kayıklık kestirimi Yerel Ağırlıklı Faz-Bağıntılı algoritmasına dayanmaktadır. Zaten FPGA de ilk amacımız işlemlerin paralel bir halde çalışmasıdır.

Faugeras et al [12] raporunda yeniden yapılandırabilir stereo makinelerinin çeşitleri bildirilmiştir. 4x4 küçük bir FPGA matrisi kullanarak, yaklaşık 7 saniyede bir kare hızında, 256x256 benek çözünürlüklü görüntüleri çapraz-ilişki (cross-correlation) gerçekleştirilmiştir.

Kanade et al[13] çalışmasında hibrid (hybrid) bir sistemi, programlanabilir mantık ile birlikte C40 dijital sinyal işlemcileri (PLDs, FPGA'e benzer) cihazları kullanarak VMEbus backplane'e monte edilerek açıklanmıştır. Yazarlar 200x200 benek uzaysal çözünürlükte (spatial resolution) görüntüler için saniyede 30 kare hızında kayıklık kestirimi (disparity estimation) yapan sistemi üretmişler.

Woodfill et al[14], PARTS güçlü, ölçeklenebilir, yineleyebilir bilgisayar (powerful, scalable, reconfigurable computer) maksimum 4x4, mesh-bağlantılı (mesh-connected) FPGA, yaklaşık 35.000 4- LUT girişi ile 320x240 benek uzaysal çözünürlük (spatial resolution) bir stereo sete saniyede 40 kareye ulaşabilmektedir.

Darabiha et al [15] 256x360 benek görüntü çözünürlüğü için saniyede 30 kare hızında çalışan, yoğun faz-bağıntı (dense phase-correlation) temelli stereo sistemi uygulamak için dört Xilinx Virtex 2000E cihazlara dayalı, bir geliştirme sistemi kullanılmıştır.

Diaz et al[16] çalışmasında ardışık (pipelined), optik akış (opticalflow) işleme sistemi ele alınmıştır. Sistem tek bir FPGA kullanarak saniyede 30 kare hızında 640x480 benek görüntü çözünürlüğü elde etmişlerdir.

Niitsuma et al[17] notunda FPGA tabanlı mimarisi ile bir off-the-shelf PCI board ile birlikte, verimli optik akışı hesaplamak için Mutlak Farklar Toplamı (SAD) tabanlı tekniği kullanılmıştır. Yoğun vektör haritaları (dense vector maps) önerilen sistem tarafından sırasıyle saniyede 840 kare hızında 320x240 ve saniyede 30 kare hızında 640x480 benek stereo görüntü çiftleri oluşturulmuştur. Alan-bazlı (Area-based) tekniği 121 seviyeleri aralığında maksimum eşitsizliği ile birlikte 7x7 benek eşleşen bir pencerelerdir kullanılmıştır.

Roh et al[19] bildirisinde uyarlanabilir (adaptive) pencere agregasyon (window-aggregation) yönteme dayalı bir teknikte Mutlak Farklar Toplamı (SAD) ile birlikte kullanılmıştır. Bu 32 kayıklık seviyeleri ile saniyede 47 kare hızında 1024x1024 benek boyutu görüntüleri işleyebilen, uygulanan pencere tabanlı (window-based)

algoritmaları düşük FPGA kaynak kullanımı ile birlikte kayıklık haritası kalitesinde gözle görülür bir performans sunmuştur.

S. Wong et al[32]'de bir 16 x 1 SAD işlemini gerçekleştiren SAD16 birimini önermişlerdir. Bu genel olarak 16 x 16 SAD işlemini hızlandırarak genel amaçlı bir işlemci çekirdeği artırmak için tasarlanmıştır. Ardışık (pipelined) durumunu 16 kata çoğaltılarak SAD16 biriminin tam 16x16 işlemini gerçekleştirilmesini göstermişlerdir. SAD16 uygulanması 19 saat döngüsünden sonra ilk sonucu üretilmiştir.

SAD16 biriminin çoğaltılmasını ve başka bir toplayıcı ağaç ekleyerek, ortaya çıkan tam paralelize olunmuş uygulanması 16 x16SAD sonucu elde etmek için 27 saat döngüsü gerektirmiştir. Ardışık (pipelined) SAD16 birimi yöntemi kullanarak başka bir alanda daha etkin bir 16 x 16 SAD işlemi yapabilmek için 42 saat döngüsü gerek olmaktadır.

M. Hariyama et al[33] çalışmalarında adaptive window-size SAD için yüksek hızlı ve güvenilir stereo eşleme işlemci mimarisi sunulmaktadır. Hesaplama karmaşıklığını azaltmak için, SAD örtüşmeyen bölge (non-overlapping region) ayrılmıştır görüntüleri kullanarak hesaplanır ve eşleşen sonuç iteratif bir pencere boyutunu azaltarak işlenmiş olur.

Algoritmanın tam paralelleşme gücünden faydalanmak için Pencere – paralel ve pixel–paralel (Window-parallel-and-pixel-parallel) mimarisini kullanma önerilmiştir. Bu mimari ayrıca, hafıza ve referans benek düzenliliğe dayalı işlevsel birimler arasında bir bağlantı ağı karmaşıklığını azaltmıştır.

Sonuç olarak, literatür'de FPGA tabanlı sistemler çoğu nispeten geniş kullanım ve ağır geliştirme bordları yüksek tüketim özellikleri ile yaygın kullanılmaktadır.

1.2 Tezin Amacı

Günümüz yazılım ve donanım dünyasının temel ihtiyaçlarından biri olan paralel işlem gerektiren bir uygulamanın donanımsal olarak uygulanması bu çalışmanın temel amacını oluşturmaktadır. Paralel işlem gerektiren Stereo görme, sistemi FPGA'de donanımsal olarak uygulanmıştır.

Stereo görme, doğrudan derinlik ölçebilme özelliği ile 3 boyutlu algılama için çok önemli bir çalışma alanıdır. Stereo eşleme problemi 3B bir noktanın farklı görüntü düzlemlerinde karşılık geldiği noktaları bulmaya çalışır. Eşleşen noktaların konumları arasındaki fark stereo kayıklık (disparity) değerini verir. Görüntü düzlemindeki her bir nokta için farkları temsil eden bir başka görüntü elde edersek bunada stereo kayıklık haritası (disparity map) elde etmiş oluruz. Stereo görme ile engel tespiti geliştirilen sistem iki aşamadan oluşmaktadır.

İlk aşamada öncelikle optik eksenlerde haraket olmayacak şekilde paralel olarak konumlandırılmış iki kameradan eş zamanlı olarak görüntü alımı yapılır, sonra stereo eşleme yöntemi ile sağ ve sol kameradan alınan eş zamanlı görüntüler kullanılarak stereo derinlik haritası çıkartılır.

İkinci aşamada stereo derinlik haritası üzerinden engellerin konumu kameralara olan uzaklığı ve kapladığı alan tespit edilerek görüntü elde edilir. Hesaplama karmaşıklığı ve büyük miktarda veri erişimi olan uygulamaları geleneksel bilgisayarda uygulamak oldukça karmaşık ve zaman alıcı bir uygulama olacaktır. Bu projenin amacı yukarıda belirtilen problemi tek FPGA kullanarak hızlı bir şekilde çözerek sonucu değerlendirmektir.

Görüntü alımında aralarındaki mesafe 6.5 cm olan iki kamera ile 1280 x 1024 çözünürlüğünde görüntü alımı yapılmıştır. Kameralardan görüntü alınırken kameralar bir birine paralel olması için düz bir plakaya vidalanarak sabitlenmiştir, ayrıca bazı hasas benek kayma noktalarında ise yazılım programı tarafından hizalanarak (align) stereo eşleme adımında arama uzayı tek bir boyuta indirilir.

Günümüzde Robotik uygulamada Stereo Görme sistemi yaygın olarak kullanılsa bile donanımsal (hardware) tabanlı uygulamalarda oldukça zor gerçekleştirilen konulardan biridir. Bu çalışmada Alteranın DE4 Geliştirme ve Uygulama bordunu kullanarak, Stereo eşleme problemlerinden Alan-Tabanlı yaklaşımıyla, Mutlak Farklar Toplamı algoritması ile kayıklık bulma işlemi gerçekleşmektedir.

1.3 Bulgular

Tez çalışmasında algoritma iki şekilde uygulanmıştır. İlk olarak MATLAB'ta sonra Stratix IV GX EP4SGX230 FPGA bordunda donanımsal olarak uygulandı. FPGA uygulamasında 3x3 penrece boyuntunda uygulandığında derleme süresi yaklaşık olarak 1 saat sürede, % 43 mantık kullanımı ve 72,545 / 182,400 (40 %) kombinasyonel aritmetik mantık birimi kullanarak sonuca ulaşmıştır. 8x8 pencere boyutunda uygulandığında ise derleme süresi yaklaşık olarak 5 saat sürede, % 85 mantık kullanımı ve 150,435 / 182,400 (82 %) kombinasyonel aritmetik mantık birimi kullanarak sonuca ulaşmıştır.

BÖLÜM 2

STEREO GÖRME

Stereo Görme doğrudan derinlik ölçebilme özelliği ile 3 boyutlu algılama alanıdır[20].

Uygulamada, stereo görüntüleme, iki kamera kullanırken dört adımdan oluşur.

1. Matematiksel olarak, radyal ve teğetsel mercek bozukluğu (lens distortion) kaldırmak bu da "undistortion" denilmektedir. Bu basamağın çıkışıda "undistorted images ".
2. Kameralar arasındaki açıları ve mesafeleri ayarlamak bu da düzeltme "rectification" denir, bu basamağın çıkışı da "row aligned"-satır hizalamak and "rectified"-doğrultulmuş denir.
3. Sağ ve sol kameradan alınan görüntülerin karşılıklarını bulmak (correspondence). Bu basamağın çıkışı da "disparity map" kayıklık haritası denir. Burada ki kayıklık te aynı cisim üzerinde görünen görüntü düzeyinin sağ ve sol kameralarının x koordinatlarının farkları $X_{(l)} - X_{(r)}$.
4. Eğer kameraların geometrik düzenlemeleri biliniyorsa, kayıklık haritasını "triangulation" aralığına koyabiliriz bu da "reprojection" denir. Bu basamağın çıkışıda "depth map" derinlik haritası denir.

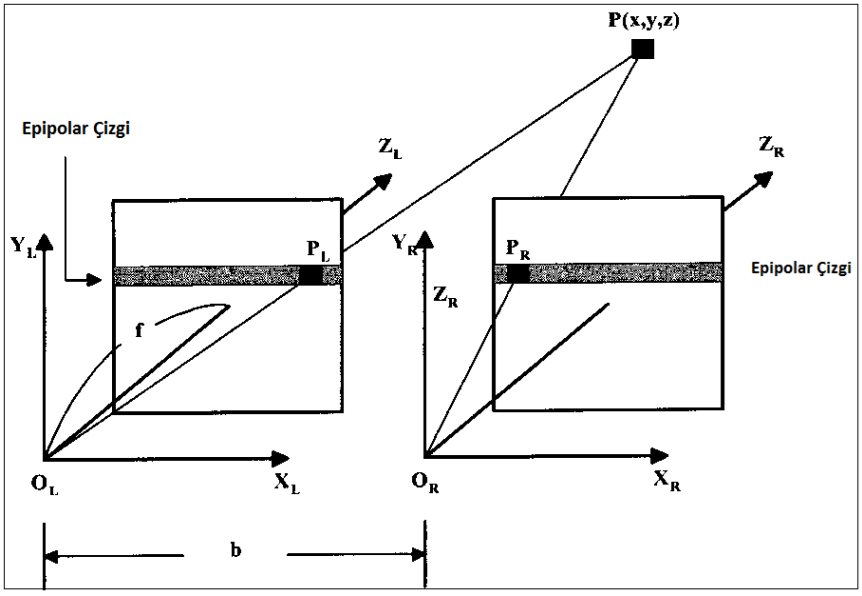
Stereo analizi 3B görüntü, 3B yeniden kurma (reconstruction), 3B nesne algılama ve robotik uygulamalarında örneğin, navigasyon, yol planlama, haritalama ve lokalizasyon, gibi birçok gerçek zamanlı uygulamalarda kullanım nedeniyle Bilgisayar Görme

alanında en önemli konulardan biridir. Stereo analizin ana problemide Stereo Eşleme, sağ ve sol görüntülerine karşılık gelen benek uygunluklarını bulmaktır.

2.1 Stereo Eşleme Problemi

Stereo Eşleme problemi 3B bir noktanın farklı görüntü düzlemlerinde karşılık geldiği noktaları bulmaya çalışır. Eşleşen noktaların konumları arasındaki fark stereo kayıklık değerini verir. Görüntü düzlemindeki her bir nokta için farkları temsil eden bir başka görüntü elde edersek bunada stereo kayıklık haritası (disparity map) denir.

Stereo eşleştirme problemi görüntü nesnelerin derinlik bilgisini bulma olarak tarif edilebilir. Çözüm sürecinde, özel olarak yerelleştirilmiş kameralar kullanarak sağ ve sol görüntüler yakalanır. Sonra bazı benek yoğunluğu benzerlikleri ve kısıtlamalara göre derinlikleri hesaplanır. Şekil 2.1'de gösterildiği gibi.

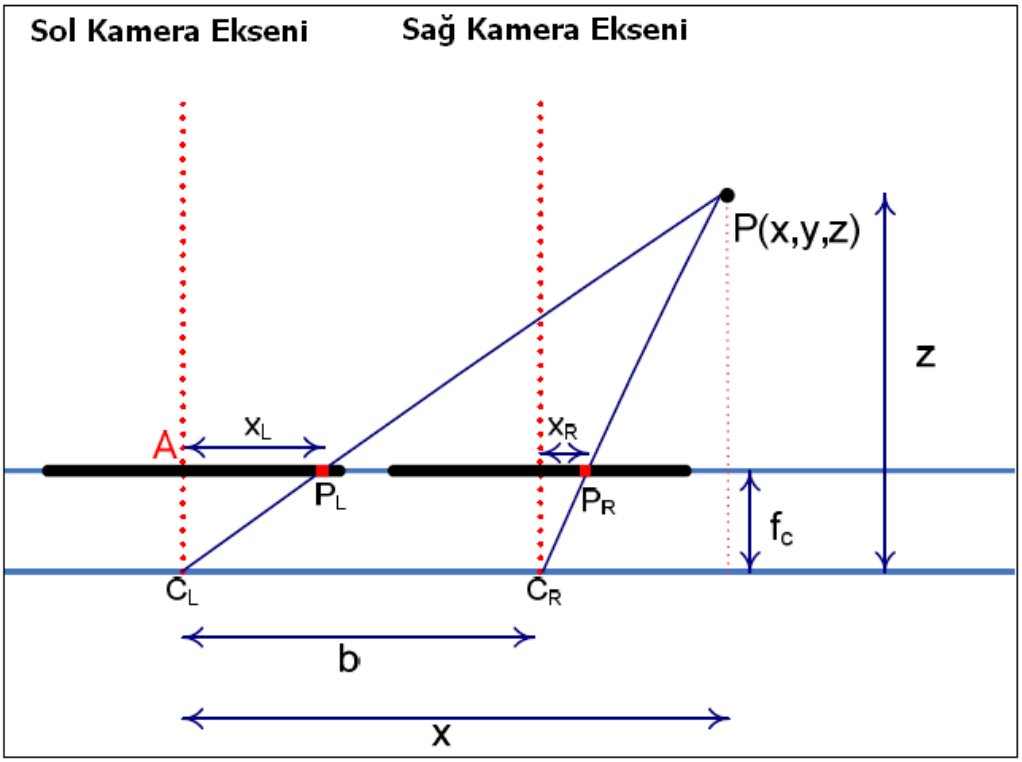


Şekil 2.1 Parallel Kamera Lokalizasyonu [5].

b Kameralar arasındaki "baseline" çizgisidir, odak uzaklığı (focal length)da f_c dir, $(X_L Y_L Z_L)$ ve $(X_R Y_R Z_R)$ sağ ve sol kameranın koordinat sistemleridir, P_L ise sol kameranın izdüşümüdür, P_R da sağ kameranın izdüşümüdür (projection). P_L ve P_R Sağ ve Sol görüntülerin farklı dikey koordinatları (vertical coordinate) olan epipolar çizgisi üstündedirler ancak kamera konumları epipolar çizgisinin her iki görüntününde aynı dikey koordinatlarda olması için ayarlanmıştır[5].

Kayıklık (disparity) mutlak yatay koordinat arasındaki sağ ve sol görüntülere karşılık gelen benekler (P_L, P_R). Nesne eşitsizliği $P(x, y, z)$, $(x_L - x_R)$ olarak tanımlanır. Stereo eşleşmenin, ana hedefini bir görüntüdeki benek kayıklık değerlerini hesaplamak.

Şekil 2.2'de Derinlik hesaplama formulünü elde etmek için Kamera Lokalizasyonu kullanılmıştır.



Şekil 2.2 3B Stereo Kamera Lokalizasyonu[11].

$\frac{f_c}{x_L} = \frac{z}{x}$ ve $\frac{f_c}{x_R} = \frac{z}{x-b}$, Derinlik(Depth), z, $\mathrm{P}(\mathrm{x},\mathrm{y},\mathrm{z})$ objesi böyledir $z = \frac{f_c * b}{|x_L - x_R|}$ (1)

Denklem (1), kayıklık $(x_L - x_R)$ nesnenin derinliği ile ters orantılıdır. f_c Odak uzaklığı ve b kameralar arasındaki mesafe verilmişken, kayıklık: derinlik hesaplama için sadece bilinmeyen bir değerdir. Görüntü üzerindeki benek eşitsizliği hesaplamak için farklı stereo eşleşen yöntemleri vardır. Bölüm 2.2'de bu yöntemlerin temel özelliklerinden bilgi verilir.

2.2 Stereo Eşleme Yöntemleri

Araştırmacılar Stereo eşleştirme performansını geliştirmek ve problemini çözmek için çeşitli teknikler önermişlerdir. Bu teknikler kullanım amaçlarına göre kategorilenmiştir.

Stereo eşleştirme algoritmaları genel olarak üç gruba ayrılır. 1) Yoğunluk-tabanlı (Intensity-based) 2) Özellik-Tabanlı (Feature-Baesd). 3) Faz-Tabanlı (Phase-Based) Bu algoritmaların kayıklık bilgisini (disparity information) hesaplamak için farklı yaklaşımları var[6].

Özellik-tabanlı (feature-baesd) stereo eşleşen yöntemler, görüntülerin kenarları ve köşeleri gibi bazı özel bilgileri kullanırlar. Öncelikle, bazı özellik algılama algoritmaları görüntülerin özel özelliklerini bulmak için kullanılır ve sonra da görüntülerde aynı özellikleri bulmak için eşleştirme işlemi uygulanır. Eşleştirme işlemi özel özellikleri olan beneklere kullanılır, sadece bu benekler kayıklık değerleri olarak atanır. Bu nedenle, özellik-tabanlı yöntemleri seyrek aykırılık haritaları (sparse disparity maps) üretiyor [9,10].

Faz-tabanlı (Phase-Based) Bölge tabanlı (region based) stereo eşleşen yöntemleri sağ ve sol görüntüler bölgeleri kullanarak eşitsizliği hesaplar. İlk önce, görüntüler bazı segmentasyon teknikleri kullanarak alt bölgelere ayrılır ve bu bölgeler karşılaştırılır. Karşılaştırılan sonuçlarına göre, aynı bölgedeki benek kayıklık değeri belirlenir. Bu nedenle, bölge tabanlı algoritmalar, benek tabanlı bir yöntem gibi yoğun kayıklık haritaları oluşturur[8].

Faz-Tabanlı yöntemler, stereo görüntüler arasında pürüzsüz aydınlatma çeşitleri bulunması halinde, sağlam olması gösterilmiştir[24].

Yoğunluk-Tabanlı (Intensity-Based) algoritmalar stereo görüntüdeki eşleşen görüntü beneklerini bulur. Evrensel ve Yerel olarak iki alt başlıklara ayrılır. Evrensel metodlar bazı eniyileme (optimization) tekniklerini kullanarak belirlenen enerji fonksiyonunu en küçük yapmayı amaçlar. Evrensel metoda kullanılan teknikler: Dinamik Programlama (Dynamic Programming) İnanç yayılma (Belief Propagation) ve Grafik Kesme (Graph Cut) [7].

Yerel metodlar ise belirlenen bir pencere içindeki görüntü elemanı bilgilerini kullanarak derinlik hesaplamaya çalışır. Karşılaştırıldığında toplam benek farkları, eşleşen maliyetini hesaplamak için farklı teknikler kullanılır: Fark Karelerinin Toplamı (Squared Intencity Difference), Mutlak Farkların Toplamı (Sum of Absolute Difference), Mutlak Yoğunluğu Farkları (Absolute Intensity Difference) [7].

Yoğunluk-tabanlı (Intensity-based), Bu yöntemde, en benzer beneği bulmak amacıyla bir resimdeki her beneğin yoğunluğu diğer görüntü içinde beneklerin bir aralık yoğunluğu ile karşılaştırılır. Performansını artırmak için bazı yoğunluk tabanlı (intansity-based) teknikler noktadan-noktaya karşılaştırma (point-to-point comparison) yerine pencere tabanlı karşılaştırma (window-based comparison) kullanır.

Bu tekniği kullanmada önemli nokta da pencere boyutunu (window size) doğru seçmektir. Küçük pencereler yeterli görüntü yapısına sahip olamaz ve dolayısıyla, yanlış eşlemeye yol açabilir. Büyük pencereler, pencere boyutundan küçük olan ince görüntü yapısını kaybedebilir. Pencere boyutu sorununu aşmak için bazı Adaptif Eşleşen Pencere Yaklaşımı (adaptive window matching approach)[13] ve Kabadan-İnceye-Yaklaşımı (coarse-to-fine approach) gibi teknikler ortaya atılmıştır.

Yoğunluk-Tabanlı tekniklerin temel zayıflığı bağımsız karşılaştırma yaklaşımı olarak binoküler görüntülerde parlaklık farklılıklarına duyarlı olmasıdır.

Stereo Eşlemede en zor problemlerden biri olan eşleme problemi ve buna ek olarakta donanımsal uygulaması da oldukça zor olan stereo eşleme probleminin bilgisayar görme alanında ve literatür'de halen aktif bir şekilde araştırılan konulardan biridir. Bu çalışmada Yukarıda bahsedildiği gibi Yoğunluk-Tabanlı (Intensity-based) yaklaşımda, her kameradan gelen benek yoğunlukları karşılaştırılarak, Mutlak Farkların Toplamının (SAD) minimum yoğunluğundan elde edilen verilerden bir kayıklık haritası oluşturulmuştur.

2.3 Önerilen algoritma

İlk olarak, görüntünün yerel farklılığı (local variation) çeşitli pencere boyutları (window sizes) için hesaplanır. İkincisi, farklılık sonuçlarına bağlı pencere boyutları için kayıklık haritası bir Mutlak Farkların Toplamının (SAD) pencere tabanlı tekniği kullanılarak hesaplanır. Bu adımlar donanımsal olarak bir FPGA cihazında uygulanmaktadır.

Değişken boyutlu pencerlerden farklılık (variation) tahmini (Variation estimation through variable sized windows), Sorunu çözmek için hızlı teknikleri sağlamak üzere, Yoğunluk-Tabanlı algoritmaları önerilmiştir[25]. Böyle algoritmaların çoğunda arama, bir benek merkezli bir pencere üzerinden yapılır. Küçük pencereler yeterli görüntü yapısına sahip olamaz ve dolayısıyla, yanlış eşlemeye yol açabilir. Büyük pencereler, pencere boyutundan küçük olan ince görüntü yapısını kaybedebilir. Böylece daha verimli kayıklık haritası değerlendirme elde etmek için, değişken boyutlu pencereler (variable sized windows) kullanarak resmin üzerinde benek açısından gri tonlama değeri (grayscale value) cinsinden yerel farklılık ölçüsünü tahmin etmek faydalıdır. Bir destek penceresinde benek yerel farklılık ölçüsü, pencerede komşu benekler arasındaki yoğunluk farkları basit bir istatistiktir. Birinci adım referans (sol) görüntü için pencerelerin yerel farklılık hesaplanmasından oluşmaktadır.

Lokal Farklılık (Local Variation)(VL) aşağıdaki formüle göre hesaplanmaktadır:

$$LV(p)=\sum_{i=1}^{N}\sum_{j=1}^{N}\left|I(i,j)-\mu\right| \qquad (2)$$

Burada μ = görüntü penceresinin gri tonlama değerinin (grayscale value) ortalaması. Burada verilen bir pencere merkez beneği p'nin yerel farklılığı komşu gri tonlama değerlerine (grayscale value) göre hesaplanır. N seçilen kare pencere boyutu, bizim durumumuzda, 2 veya 3 veya 7 olabilir. Bir 3x3 penceresinin durumuda, yerel farklılığı üst sol benek için hesaplanır.

Adaptif penceresi arama (Adaptive window search)

Kayıklık harita üretimi için Mutlak Farkların Toplamının (SAD) pencere tabanlı bir algoritma kullanılmıştır. Bu seçim hızı artırmak için yapılmıştır.

Burada, kare farkların toplamı, bağıntı (correlation) ya da görüntü önemi (image moments) gibi diğer ölçümler de kullanılabilirdi, ancak hız sorunları dikkate alındığında bunlar elenmiştir. Ayrıca donanımsal uygulamada Mutlak Farkların Toplamının (SAD) basitliği nedeniyle tercih edilmiştir. Görüntünün her noktası için pencere boyutu 3x3 benek sabit bir pencere boyutu ayarlandı. Bir stereo çifti görüntüye karşılık gelen noktaları bulmak için blok eşleme yöntemi Mutlak Farklar Toplamı (SAD) hesaplamaya dayalıdır. Kullanılan SAD denklemi aşağıdakidir:

$$SAD(i, j, d) = \sum_{\mu=-w}^{w} \sum_{v=-w}^{w} \left| I_l(i+\mu, j+v) - I_r(i+\mu, j-d+v) \right| \quad (3)$$

Burada $I_l \; ve \; I_r$ sağ ve sol görüntü benek, μ gri tonlama değerleridir (grayscale value), d kayıklık aralığıdır, w ise pencere boyutu, i, j satır sütün(rows, columns) ise SAD'dan hesaplanan pencere merkezi beneği kordinatlarıdır.

Bir kez SAD, bütün benekler ve tüm kayıklık değerleri için hesaplanır, bir benzerlik akümülatör her bir benek için düzenlenmiştir, buda büyük olasılıkla eşitsizliği göstermektedir. Kayıklık haritasını hesaplamak için tüm kayıklık değerler $(d_{\min} \; dan \; d_{\max} \; kadar)$ için SAD da bir arama her bir benek için yapılmaktadır. Kayıklık aralığında, $(d_{\min} \; dan \; d_{\max} \; kadar)$, bir benek için SAD minimum dur, bu değer kayıklık haritası için uygun gelen bir benek değeridir.

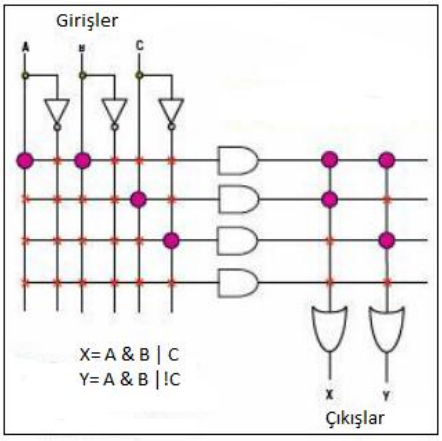
$$D(i, j) = \arg\min_{d \in [d_{\min}, d_{\max}]} SAD(i, j, d) \quad (4)$$

BÖLÜM 3

PROGRAMLANABİLİR MANTIK

Programlanabilir mantık cihazı (Programmable Logic Device PLD) yeniden yapılandırılabilir dijital devreleri oluşturmak için kullanılan elektronik bir bileşenidir. Aksine mantık kapıların sabit bir işlevi vardır, PLD imalatı sırasında tanımlanmış bir fonksiyonu yoktur. PLD bir devre de kullanılmadan önce programlanması gerekir, bu işleme yeniden yapılandırabilme (reconfigurable) denilmektedir.

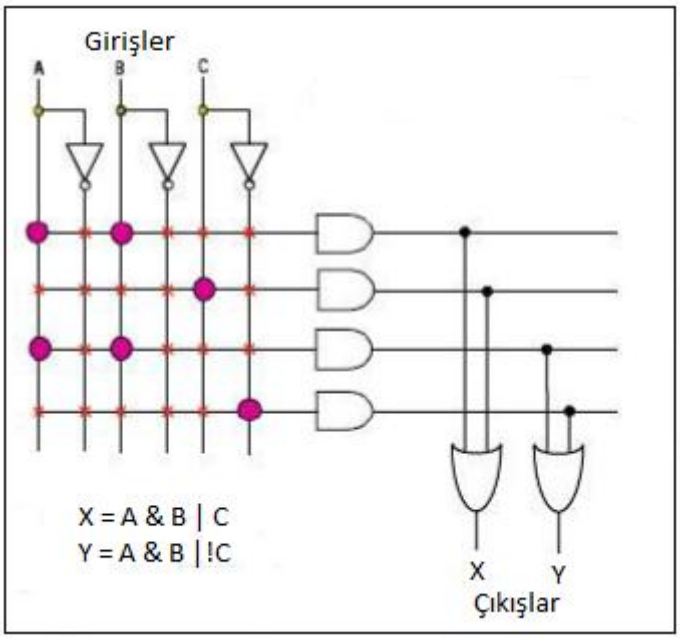
1970'lerin sonunda standart mantık birimlerinin kullanımı yaygınlaşarak dijital devre tasarımları standart mantık birimlerden oluşan elektronik kartlar ile oluşturulmaktaydı. Signetics firması tarafından dijital tasarımlara esneklik sağlamak için PLA (Programmable Logic Arrays-Programlanabilir Mantık Dizileri) yapısı geliştirildi. PLA dikey ve yatay hatlardan AND OR kapılarına bağlanan bir mimariye sahiptir (Şekil 3.1). Dikey ve yatay hatların kesişim noktalarında bulunan sigortalar yazılım tarafından "patlatılarak" (blowing) istenilen fonksiyon programlanabilmektedir. PLA mimarisi esnek yapıdadır, ama giriş-çıkış süresi büyük olduğundan yavaş kalmaktadır[28].



Şekil 3.1 PLA Mimari Yapısı[28]

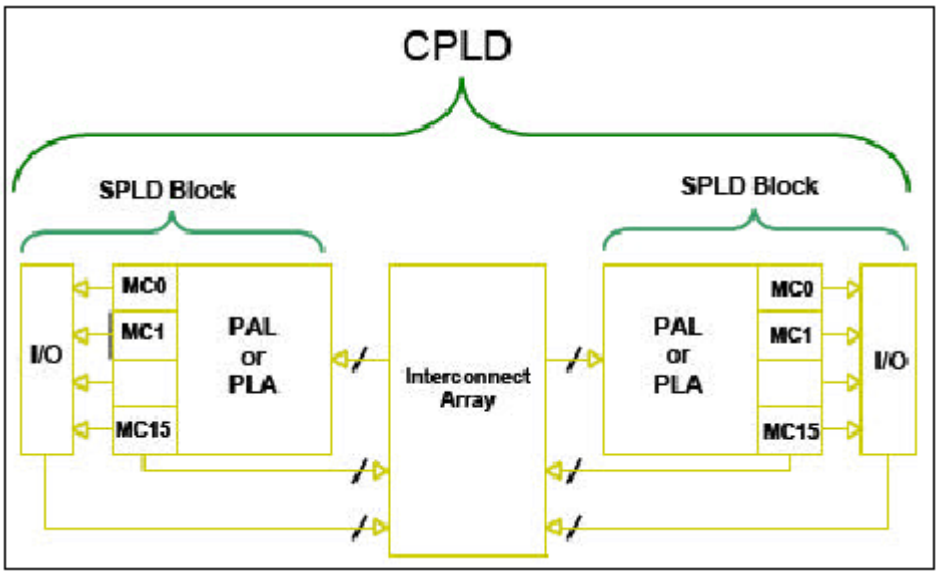
1978'lerin başında Monolithic Memories Inc (MMI) firması tarafından PLA mimarisinden farklı olarak OR dizisini sabitleştirerek PAL (Programmable Array Logic-Programlanabilir Dize Mantık) mimarisini geliştirmişitir (Şekil 3.2). PAL mimarisi PLA'ye kıyasla daha sade ve daha hızlıdır ama PLA kadar esnek değildir. PAL ve PLA tipindeki programlanabilir cihazlar genel olarak SPLD (Simple Programmable Logic Device-Basit Programlanabilir Mantık Cihaz) olarak bilinirler.

SPLD ile mantık tasarımın doğrulanması özel bir programlama cihazı ile istenmeyen bağlantıların açık devre yapılması ile sağlanır. SPLD'ler standart mantık entegrelerinin 50 katından daha fazla sayıda kapı kullanır ve daha tutarlı yapı sunmaktadır[28].



Şekil 3.2 PAL Mimari Yapısı[28]

CPLD (Complex Programmable Logic Device- Karmaşık Programlanabilir Mantık Cihaz)'nin mimari yapısı birçok sayıda macrocell olarak adlandırılan SPLD'ler ve bunların arasında genel amaçlı ara bağlantılardan oluşur (Şekil 3.3). Basit mantık fonksiyonlar tek macrocell içinde gerçekleştirilir, daha karmaşık fonksiyonlar birden fazla macrocellin genel amaçlı ara bağlantılari ile birlike elde edilir. CPLD' ler yüksek hızda çalışır ve tasarım kolaylığı maliyetin düşüklüğü gibi birçok avantaj sağlamaktadır, daha bir özelliğide yapılandırılmasını kendi üzerinde saklamasıdır.



Şekil 3.3 CPLD Mimari Yapısı[28]

3.1 FPGA (Field Programmable Gate Array-Alanda Programlanabilir Kapı Dizileri)

FPGA (Field Programmable Gate Array - Alanda Programlanabilir Kapı Dizileri), 1985 yılında Xilin'in kurucusu Ross Freeman tarafından icat edilmiştir. FPGA mimari yapısı kullanıcı tarafından belirlene bilen programlanabilir mantık blokları ve bu bloklar arasındaki ara bağlantılardan oluşan ve geniş uygulama alanlarına sahip olan sayısal tümleşik (integrated) devrelerden oluşur. Tasarımcının ihtiyaç duyduğu mantık işlevlerini gerçekleştirme amacına yönelik olarak üretilmiştir. Her bir mantık bloğunun işlevi kullanıcı tarafından düzenlenebilmektedir.

FPGA ile temel mantık kapılarının yapısı daha karmaşık olan devre elemanlarının işlevselliği artırılmaktadır. Mantık blokları ve ara bağlantıların imalat sürecinden sonra programlanabilmesidir. Bu nedenle Alanda programlana bilir ismi verilmiştir. FPGA'ler temel olarak CLB (Configurable Logic Block-Yapılandırabilir mantık bloklar) veya LE (Logic Element-Mantık Eleman) olarak mantık bloklar ve Giriş-Çıkış blokları içermektedir (Şekil 3.4).

Mantık Bloklar hem uyumlu (synchronous) hem birleşim (combinataion) mantık devreleri doğrulamak için temel mantık kaynakları içerir. Giriş-Çıkış blokları ise yonga'nın bacaklarına bağlı birimlerdir ve çeşitli mantık voltaj veya sinyal standartlarına göre programlanabilirler. Mantık bloklar içerisinde kombinasyonel mantık fonksiyonlar,

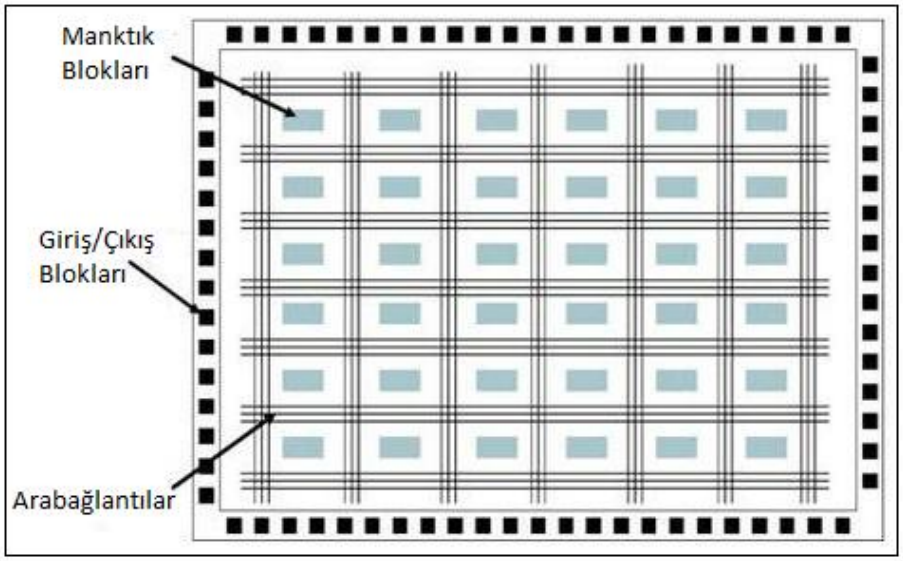
mantık kapıları yerine LUT (Look Up Table - Arama Tablosu) ile gerçekleştirir. Mantık bloklar ayrıca çoklayıcı flip-flop gibi yapılarıda içerir. FPGA'ler mantık blok ve giriş-çıkış blok bileşenleri yanında DPS birimi (module) (çarpma, çarpma ve toplama), RAM veya ROM hafıza blokları, fiziksel CPU gibi kaynaklara da sahip olabilirler.

FPGA programlanabilir mantık blokları, bu blokları dizisini çevreleyen giriş-çıkış blokları ve ara bağlantılar olmak üzere düzenlenebilir üç ana bölümden oluşur (Şekil 3.4) de gösterilmiştir. Programlanabilir mantık blokları, ara bağlantılar içerisine gömülü şekilde bulunur. Programlanabilir mantık bloklarının yapılandırılması ve bu bloklar arasındaki iletişim ara bağlantılar sayesinde gerçekleşir. Giriş çıkış blokları, ara bağlantılar ile bütünleşmiş devrenin paket bacakları arasındaki ilişkiyi sağlar.

FPGA'ler genellikle benzerleri ASIC'lere (Application Specific Integrated Circuit) göre daha yavaştır. Buların yanında FPGA'lerin pazara giriş zamanı düşük, daha kısa zamanda geliştirebilme, çıkan hataları saha da düzeltebilme gibi avantajı vardır.

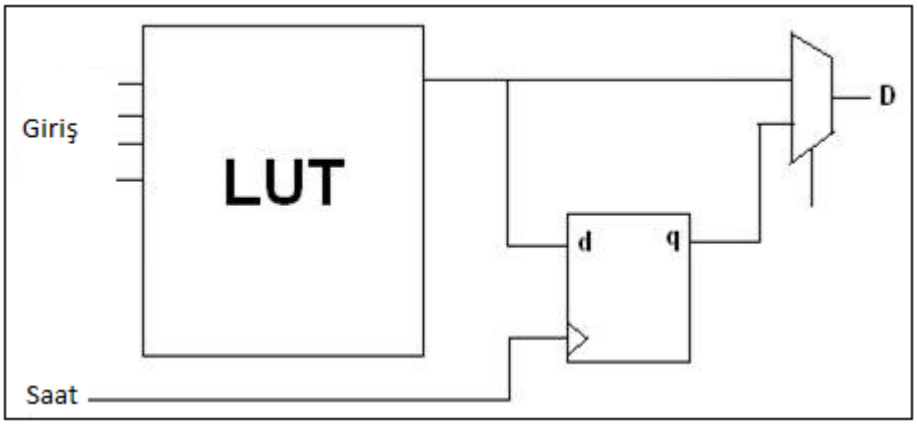
FPGA'ler düşük maliyetli olması ve tasarım sırasında kullanıcıya esneklik sağlaması sebebiyle kullanımı gittikçe yaygınlaşmaktadır. Savunma, sayısal işaret işleme, uzay, tıbbi görüntüleme ve otomotiv gibi birçok alanlarda uygulanmaktadır. FPGA'ler özellikle paralel işlem gerektiren uygulamalarda yaygın olarak kullanılmaktadır.

FPGA üreten firmalardan bazıları: Xilinx, Altera, Actel, Lattice Semiconductor ve QuickLogic. Bu çalışmada Alteranın ürettiği DE4 Stratix IV GX cihazını kullanılmaktadır.



Şekil 3.4 FPGA Yapısı[27].

FPGA’in Mantık Blokları bir Logic-Cell 1 adet Lookup Table (LUT), 1adet D Flip-Flop ve bir adet 2 to 1 Mux’ tan oluşur (Şekil 3.5).



Şekil 3.5 Mantık Blokları[27].

LUT 'lar aslında bir mantık işlemi yerine getiren küçük belleklerdir (RAM). N girişli bir LUT, 2 eN ’li bir belleğe işaret eder. Binlerce Mantık Bloklarının birleşimi sonucunca karmaşık ve büyük programlar oluşturulur. Mantık bloklarının ara bağlantıları matris şeklindeki veri yolları ve programlanabilir anahtarlarla (FPGA yüklenen programa göre) sağlanır. FPGA tasarımı, her bir mantık hücresinin uygulayacağı fonksiyonu ve programlanabilir anahtarların durumunu (açık/kapalı) belirleyerek bu mantık bloklar arasındaki bağlantıları tanımlar.

FPGA pinleri genel olarak 2 kategoriye ayrılır. Ayrılmış pinler (Dedicated pins). Bir FPGA'de tüm pinlerin %20 ila %30'u ayrılmış pindir. Bu pinler, FPGA'de gerçekleştirdikleri özel fonksiyonlara göre 3'e ayrılır. Güç Pinleri FPGA için gerekli olan güc (power) ve toprak (ground) sağlayan pinlerdir. Konfigürasyon Pinleri oluşturulan programın FPGA yüklenmesi (download) için kullanılan pinlerdir. Saat Pinleri, Saat sinyalleri için ayrılmış özel pinlerdir.

Kullanıcı pinleri (User pins) kullanıcı tarafından yapılandırabilen standart I/O pinleridir. Input, Output, Input/Output olarak üç kategoriye ayrılır. Her bir I/O pini FPGA'de bir IO hücresine bağlıdır. IO hücrelerinin güçleri VCCIO tarafından sağlanır. Eski FPGA' ler birden fazla VCCIO pinine sahip olmalarına rağmen, bütün pinler aynı voltajla beslenirdi. Yeni üretilen FPGA'lerde ise IO'lar gruplara ayrılabilir ve bu gruplar farklı

voltajlardan beslenebilirler. Böylelikle bir grup IO pinleri 3.3 V ile çalışırken, diğer grup IO pinleri de 2.5 volt ile çalışabilmektedir.

Saat ve Evrensel Çizgi FPGA genellikle uyumlu (synchronous) tasarlanır. Yani FPGA tasarımları saat tabanlıdır ve FPGA içerisindeki D Flip-Floplar, saat sinyalinin yardımıyla durum değiştirirler. Uyumlu tasarımlarda bir saat sinyalinin, bütün flip-flop'ları aynı anda tetiklemesi gerekir. Aksi takdirde FPGA'de elektriksel ve zamansal problemler oluşmaktadır. Bu probleme karşı, Evrensel Yol (Global Routing) veya Evrensel Çizgi (Global Line) olarak adlandırılan özel bir iç bağlantı geliştirilmiştir. Bu bağlantı sayesinde, Saat sinyalinin FPGA içerisindeki bütün Flip-Flop'lara aynı anda ulaşması sağlanır. Bundan dolayı saat beslemelerinin, FPGA'in saat için ayrılmış pinlerinden verilmesi gerekir.

RAM Blokları günümüzde FPGA'lerin hemen hepsinde ayrılmış RAM yani bellek üniteleri bulunmaktadır. Bunlar mantık devrelerinin işleyişi sırasında duyulan geçici depolama ihtiyacı için kullanılırlar. Bu RAM'ler tek veya çoklu erişimi destekleyebilirler. Çoklu erişimde birden fazla uygulama RAM üzerinde okuma/yazma yapabilmektedir. Çoklu erişim farklı saat'ta çalışan işlem blokları arasında veri aktarımı için iyi bir çözümdür. Örneğin 25 MHz saat ile çalışan bir veri toplama ünitesinden 50 Hz ile çalışan bir veri işleme ünitesine veri aktarmak için 2 port'lu bir RAM kullanabiliriz. 25 MHz saat ile çalışan veri toplama ünitesi veriyi RAM'e yazar ve 50 MHz de çalışan veri işleme ünitesi veriyi RAM'den okuyarak kullanır.

FPGA içerisinde büyük RAM ihtiyaçları için Block RAM'ler bulunurken küçük veriler için mantık blokları arasına serpiştirilmiş dağınık (distributed) küçük RAM'ler bulunmaktadır. Dağınık RAM için Xilinx ihtiyaca göre mantık bloklarının bazılarını RAM olarak kullanırken Altera blok RAM'leri FPGA içerisinde değişik boyutlarda paylaştırmaktadır.

FPGA düşük maliyetli talepleri ile yüksek yoğunluklu tasarımlar sunuyor. FPGA'ler hız gerektiren uygulamalar için yüksek işlem hızları sağlamada idealdir. ASIC cihazlara kıyasla FPGA'lerin önemli avantajı, NRE (nonrecurring-engineering) yokluğudur, maliyeti genellikle ASIC'lerin tasarımı ile ilgilidir.

Ayrıca, saha içi kullanım (in-field use) sırasında ''on-the-fly'' yeniden kablonabilir (rewired) ve ASIC cihazlara kıyasla FPGA'ler daha basit tasarım yöntembilimi (methodology) ile geniş yeniden yapılandıra bilirliği (reconfigurability) sağlamakta. Bir işlemci ile karşılaştırıldığında, FPGA'lerin ana avantajı yüksek işlem hızlı olmasıdır (higher processing rate). İşlemcilerin fonksiyonel birimleri sabitken (have fixed functional units) belirli uygulamaların gereksinimlerini karşılamak için FPGA kaynakların tahsisi özelleştirebilir (customize the resources allocation)[25].

Üretim teknikleri

SRAM: SRAM temelli yapılandırma hücreleri kullanılır. SRAM tekniğinin olumlu yanı, tasarım fikirlerinin hızlı bir şekilde geliştirilebilir ve sınanabilir olmasıdır. Olumsuz yanı ise sistemin her açılışında cihazın yapılandırılma zorunluluğudur.

Antifuse: Olumsuz yanı, bir kez programlanır olmasıdır. Bu özelliğinden dolayı uygulama geliştirme için tercih edilmez.

EPROM: Silinebilme ve yeniden programlanabilme özelliğine sahiptir. EPROM yongaları üzerine açılan pencere vasıtasıyla program belli bir zaman güneş ışığına ve morötesi (ultraviolet) ışınlara tutularak silinmektedir.

EEPROM: Elektrikle silinip programlanabilme özelliğine sahiptir.

Flash: Bazı çeşitleri devre içinde programlanabilirdir.

Fuse: Bir kez programlanır.

3.2 FPGA Uygulama Alanları

FPGA uygulama alanları dünyada birçok yerde yaygın olarak kullanılmaktadır[30].

Otomotiv alanında kullanıldığı uygulamalar şunları kapsamaktadır.

- Grafik/Görüntü İşleme
- Araç içi iletişim ağı
- Sürücü Yardımı
- Ses İşleme.

Yayın/Haberleşme alanında kullanıldığı uygulamalar şunları kapsamaktadır.

- Ses/Görüntü Çözümleri, Biçim Dönüştürme, Ses Örnekleme Hızı Dönüşümü.

Endüstriyel Kontrol alanında kullanıldığı uygulamalar şunları kapsamaktadır.

- Güvenlik, Endüstriyel Ağ
- Makine Görmesi
- Motor/Hareket Kontrol
- Video Kayıt,
- DSP.

Sağlık alanında kullanıldığı uygulamalar şunları kapsamaktadır.

- Tanısal Görüntüleme
- Tıbbi Cihazlar.

Askeri alanda kullanıldığı uygulamalar şunları kapsamaktadır.

- Güvenli İletişim
- Radar ve Algılayıcılar, Elektronik Harp.

Güdüm (Guidance) ve Kontrol alanında kullanıldığı uygulamalar şunları kapsamaktadır.

- Askeri DSP
- SWaP (reducing system size, weight, and power).

Bilgisayar/Depolama alanında kullanıldığı uygulamalar şunları kapsamaktadır.

- Depolama Çözümleri, Yüksek Performans Hesaplama.

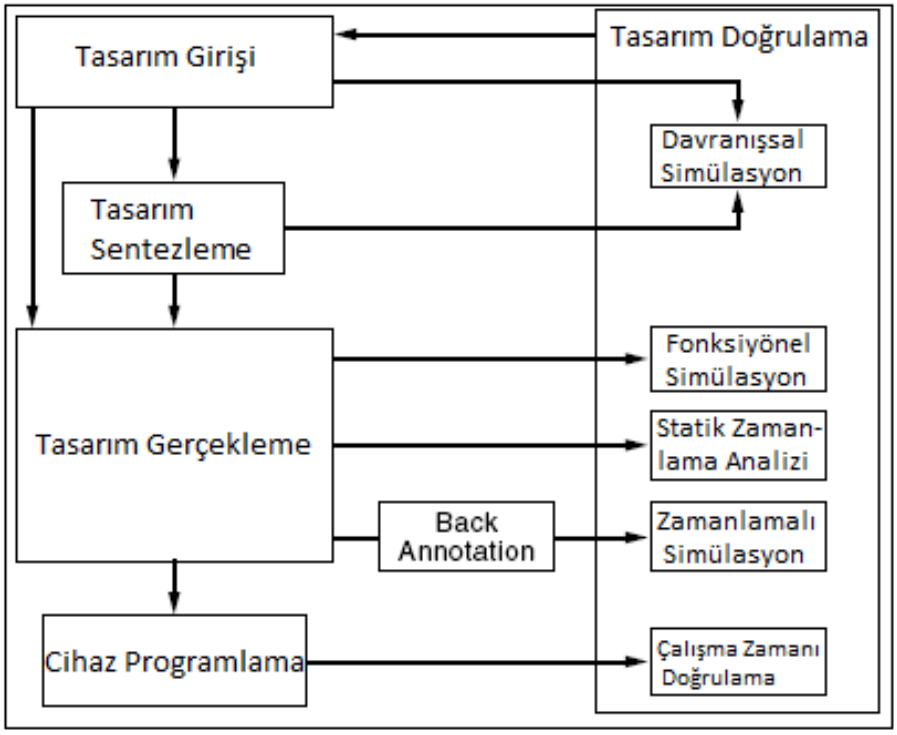
Tüketici Elektroniği alanında kullanıldığı uygulamalar şunları kapsamaktadır.

- Dijital Televizyon, DVD Çalar, Monitör/Projektör, Ev-içi ağ, Taşınabilir Oyun
- Konsolu, Dokunmatik Ekran, Ev Aletleri.

Bu alanlarda yaygın olarak kullanılmaktadır.

3.3 FPGA Tasarım Süreci

FPGA gerçeklenecek sistemler, FPGA'in yapısı ve sağladığı kaynaklar gözetilerek seçilen tasarım kalıpları kullanarak ifade edilmelidir. FPGA yapısı gereği bağlantı gecikmeleri ile "barrel shifter" tasarımı geniş bir çoklayıcı yerine donanımsal çarpma birimleri ile doğrulamak daha etkin çözüm sunmaktadır. Geniş bir yazmaç dosyasını flip-floplar üzerinde doğrulamak yerine hafıza blokları ile doğrulamak hem alan hem de hız yönünden avantaj sağlamaktadır. FPGA üzerinde dijital sistem tasarımı süreci çeşitli aşamalardan oluşmaktadır (Şekil 3.6).



Şekil 3.6 FPGA Tasarım Süreci

Tasarım Girişi / Design Entry

FPGA Platformunda tasarım süreci bir mantık tasarımın girişi ile başlar. Bu işlem HDL ve/veya Şematik olarak yapılabilir.

Davranışsal Simülasyon / Behavioral Simulation

Tasarım girildikten sonra sistemin davranışsal simulasyonu yapılır ve tasarımın doğrulaması yapılır. Bu aşamada giriş yapılan tasarımın istenen fonksiyonları gerçeklendiği doğrulanır.

Tasarım Sentezlemesi / Design Synthesis

Tasarım girişinde HDL kullanılması durumunda kurulum yapılabilmesi için HDL'den donanım sentezlemesi tapılır. Tasarım girişi şematik olarak yapılmış ise, şematik tasarım ile ifade edilen sistem hedeflenen cihazın kaynaklarına göre donanım sentezlemesi yapılır. Sentezleme sonucunda tasarlanan sistemi oluşturan mantık blokları ve bunların bağlantılarını içeren bir netlist oluşturulur.

Tasarım Kurulumu / Design Implementation

FPGA platformunda kurulumu, sentezleme sonucunda elde edilen netlist, giriş/çıkış bacak konfigurasyonu, zamanlama kısıtları ve diğer tarasım kısıtları kullanılarak işlem sırayla, Translate, Map ve Place&Route (PAR) olarak üç alt işlem olarak yapılır.

Translate sentezleme sonucunda elde edilen netlist ve tasarım kısıtlamalarını (örneğin giriş/çıkış pinlerini) birleştirir.

Map: işlemi tasarımı hedeflenen cihaz üzerinde bulunan kaynaklara eşleştirir.

PAR(Place&Route): Tasarımı zamanlama kısıtlarına göre tasarımı hedef cihaz üzerindeki yerleşimini ve yolları belirler.

Zamanlama Analizi / Timing Analysis

Zamanlama analizi, tasarımdaki mantık ve yol gecikmeleri inceleme ve kurulum imkânı sağlar.

Zamanlamalı Simülasyon / Timing Simulation

Zamanlamalı Simülasyon, PAR işlemi sonucunda belirlenen hedef cihaz yapılandırması, standart mantık ve yol gecikmeleri ile beraber yapılır.

Cihaz Programlama / Device Programming

Gerçeklenen tasarım programlama formatına dönüştürülerek hedef cihaz'ın yapılandırmasının saklandığı PROM'a yüklenir.

Çalışma Zamanı Doğrulaması / In-Circuit Verification

Hedef cihaza yapılandırma yüklendikten sonra, yazılım araçları kullanarak çalışma zamanında tasarım doğrulaması yapılabilir.

BÖLÜM 4

DONANIM TANIMLAMA DİLLERİ

Donanım tanımlama dili donanımın yapı ve davranışlarını metin olarak ifade edebilmesini sağlamakadır. İlk donanım tanımlama dilleri (HDL) 1977'li yıllarda Carnegie Mellon üniversitesi tarafından geliştirilen ISP (Instruction Set Processor) ve Kaiserslautern Üniversitesi tarafından geliştirilen KARL'dır. ISP daha çok programlama diline benziyordu ve giriş/çıkış'lar arasındaki bağlantıları tanımlıyordu. Bu sebeple tasarım benzetimi için kullanılabilirken, sentezlenemezlerdi.

KARL VLSI yonga yerleşim planı ve yapısal donanım tasarım desteği sağlamıştır. KARL'lı temel alan ABL 1980'lerin başında ABLED grafik VLSI tarasım editöründe kullanılmak üzere telekomünikasyon araştırma merkezi Centro Studi E Laboratori Telecomunicazioni CSELT (Torino, Italy) tarafından geliştirilmiştir. 1983'te Data-I/O ABEL dilini geliştirmiştir. ABEL programlana bilir mantık cihazları için sonlu durum makinelerini tasarlamak için kullanılmıştır.

Donanım tanımlama dili (DTD), elektronik devrelerin tanımlanması amacıyla kullanılan bir bilgisayar dilinin sınıfına verilen addır. Donanım tanımlama dili, devrenin yürüttüğü işlemi, tasarımını ve simülasyon yoluyla doğrulamasını gerçekleştirir.

Yazılım programlama dilinden farklı olarak, DTD'nin sözdizimi (syntax) ve semantiği donanımın birinci özelliği olan zaman ve uyumluluk açısından açık ifadelere yer vermektedir. Sadece devre bağlantılarını ifade eden ve bloklar arasındaki hiyerarşiyi tanımlamaya yarayan dillere netlist denmektedir.

DTD önceleri herhangi bir donanım için çalıştırılabilir özelliklerin benzetimini yapmak amacıyla kullanılıyordu. Zamana bağlı işlem yapan benzetim programları sayesinde bir

donanım parçası fiziksel olarak üretilmeden önce tasarımcı tarafından test edilebiliyordu. Bu çalıştırılabilirlik DTD'nin bir programlama dili olduğu hissini vermiştir.

Donanım anlamı C++ gibi geleneksel programlama dilleri ile kazanılabilir, fakat C++ dilinde zamanı açık bir şekilde ifade etmek mümkün olmadığından düzgün bir donanım tanımlama dili değildir.

Herhangi uygun bir donanım tanımlama dili ile yazılmış kod, sentezleyici adı verilen bir yazılım tarafından işlenir. Bu işlem koddan mantıksal işlemlerin çıkarılması ve buna eş değer bir netlist yaratılarak donanımın taslağının oluşturulması işlemidir. Bu işlem koddaki herhangi bir zaman yapısının ihmal edilmesi ile elde edilir. Bir dilin bir altkümesinin sentezlenebilir olması o dilin bir DTD olduğunu temin etmez.

Bir sistemi DTD kullanarak tasarlamak, bir programlama dili ile yapmaktan çoğu zaman daha zor ve daha çok zaman gerektiren bir işlemdir. Bu sebeple C programlama dili kodunu otomatik olarak DTD'ye çevrilmesi üzerinde birçok uğraş verilmiştir fakat 2004 yılı itibariyle bu işlemin yüksek bir ticari başarıya ve kullanıma ulaştığı söylenemez.

4.1 Verilog Donanım Progamlama Dili

İlk modern DTD, Verilog 1983/1984 yıllarında Automated Integrated Design Systems daha sonraları "Gateway Design Automation" olarak ismi değiştirildi, Phil Moorby ve Prabhu Goel tarafından icat edilmiştir. 1985 yılında ise donanım modelleme dili olarak değiştirilmiştir.

Daha sonraları Cadence Design Systems gelecek on senenin DTD benzetim standardı haline gelecek olan Verilog-XL'nin haklarını aldı. Cadence Design Systems tarafından lisanslanan Verilog-XL simulatörü de-facto standartı olmuştur. 1987 yılında Amerika Savunma Dairesi tarafından yapılan bir istek ile VHDL (Very High Speed Integrated Circuit Hardware Description Language) geliştirildi. Başlangıçta Verilog ve VHDL şematik devre tasarımlarının dokümantasyonu ve simülasyon için kullanılmıştır. DTD simülasyonu mühendislere şematik seviyesinden daha ileri bir seviyede çalışma imkânı sağlayarak tasarımı yüzlerce transistörden binlerceye yükseltmiştir[31].

Donanım tanımlama dillerinin mantık kapıları seviyesinde sentezlenebilmesi sayısal tasarım alanında DTD'leri ön plana çıkarmıştır. Sentezleme araçları DTD kaynak

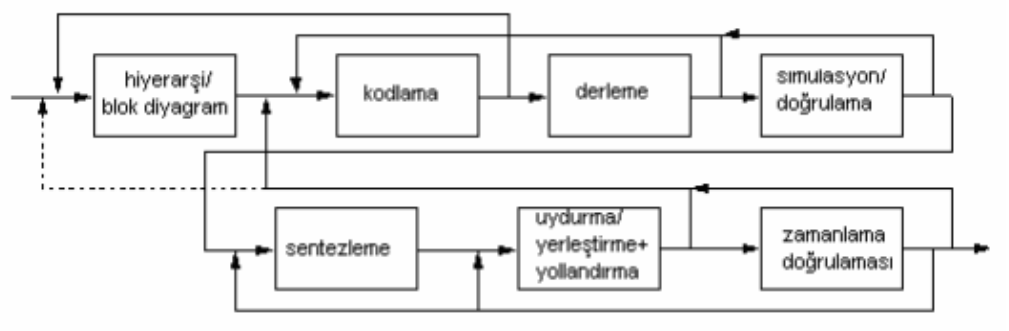
dosyalarını (yazmaç aktarımı seviyesi adı verilen bir biçimde yazılmış olan) derleyerek kapı/transistör seviyesinde üretime hazır netlist tanımlamaları yaratmaktadır.

Verilog ve VHDL senelerdir sayısal tasarımın en önde gelen iki donanım tanımlama dilidir fakat ikisi de hemen hemen aynı sınırlamalara sahiptir. Örneğin iki DTD analog veya karışık-sinyal devrelerinin benzetimleri için uygun değillerdir. Özyinelemeli (recursive) kurulan mantık yapıları için gerekli tanımlamalara sahip değillerdir. Bu amaçla ortaya çıkan özelleşmiş DTD'ler olsa dahi hiç biri VHDL/Verilog'un yerini tutamamıştır.

Verilog elektronik sistemleri modellemek için kullanılan bir donanım tanımlama dilidir Verify Logic ten gelmektedir. Verilog analog, sayısal ve karışık işaretli devrelerin tasarımını, doğrulanmasını ve yürütülmesini (execution) değişik düzeylerde desteklemektedir. Verilog C dilini anımsatır.

Senelerdir DTD'leri geliştirmek için büyük bir emek harcanmaktadır. Veriloğun 2005 yılında IEEE 1800-2005 standardıyla Systemverilog olarak bilenen son sürümü daha iyi rastgele hale getirilmiş test, tasarım hiyerarşisi ve tekrar kullanım adına birçok yenilik (sınıflar, rastgele değişkenler vs.) içermektedir. Böylece hem VHDL hem de Verilog sahip oldukları yeniliklerle gelecek senelerde de aktif olarak kullanılacaktır.

Verilog'da sistem hiyerarşik birimler ile tanımlanır. Bu birimler, Yapısal (Düşük seviyeli mantık olarak), Veri akışı ile (Çıkışları giriş sinyallerinin dönüşümü ile) Davranışsal olarak (Devreden beklenen davranışın ifadesi şeklinde) üç değişik yapıda ifade edilebilir. Üst seviyeli birimlerde soyutlamaya gidilebilmesini ve alt seviyeli birimlerinde de verimli olarak ifade etme imkanı sağlar.



Şekil 4.1 VHDL Tasarım Blok Şeması

```
module counter(CLOCK, DIRECTION, COUNT_OUT);
input CLOCK;
input DIRECTION;
output [3:0] COUNT_OUT;

reg [3:0] count_int = 0;

always @(posedge CLOCK)
      if (DIRECTION)
         count_int <= count_int + 1;
      else
         count_int <= count_int - 1;

assign COUNT_OUT = count_int;
endmodule
```

Şekil 4.2 Verilog ile 4-bit sayaç

Verilog ile 4-bit sayaç örneğinde (Şekil4.2), ilk modül oluşturulur, saat ve yönler belirlenir, sonra davranışsal olarak geliştirilmeye devap edilir.

4.2 VHDL Donanım Tanımlama Dili

VHDL (Very High Speed Integrated Circuit Hardware Description Language) Amerika Birleşik Devletleri Savunma Bakanlığı sponsorluğun'da elektronik cihazların modellenmesi ve simülasyonu için geliştirilmiştir. VHDL'in geliştirilmesindeki temel amaç modelleme ve simülasyon olmuştur. Ancak sonradan araştırmacılar tarafından tasarım aşamalarının otomasyonunu sağlayabilmek için sentezlemede de kullanılmaya başlamıştır. VHDL IEEE tarafından ilk olarak 1987 yılında standartlaşmıştır. VHDL Verilog'a göre daha esnek sistem tasarımlarına imkân vermekle beraber üst seviyeli tasarıma daha uygundur. Ancak yazılım biraz daha karmaşık ve öğrenilmesi biraz uzun süre gerektirir. Şekil 4.3'de yönü seçilebilen 4-bitlik sayaç sentezlene bilir olarak VHDL ile ifade edilmiştir.

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity counter is
Port ( CLOCK : in STD_LOGIC;
       DIRECTION : in STD_LOGIC;
       COUNT_OUT : out STD_LOGIC_VECTOR (3 downto 0));
end counter;
architecture Behavioral of counter is
signal count_int : std_logic_vector(3 downto 0) := "0000";
begin
       process (CLOCK)
       begin
          if CLOCK='1' and CLOCK'event then
             if DIRECTION='1' then
                count_int <= count_int + 1;
             else
                count_int <= count_int - 1;
             end if;
          end if;
       end process;
       COUNT_OUT <= count_int;
end Behavioral;
```

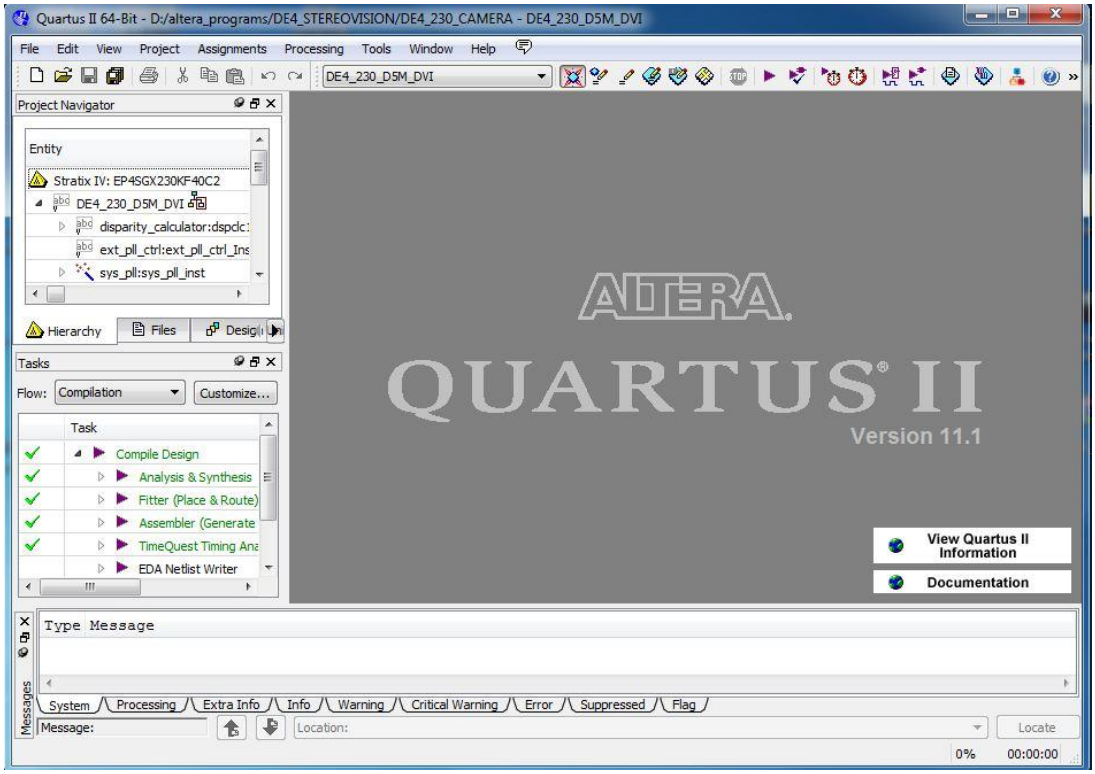
Şekil 4.3 VHDL ile 4-bit sayaç

4.3 Quartus II Yazılım Aracı

Altera Quartus II programlanabilir mantık cihaz tasarımı yazılım aracıdır. Quartus II yazılım aracı Altera tarafından DTD (hardware descriptive language) tasarımları analizleme ve sentezlemek için üretilmiştir. Geliştiricilerin (developer) kendi tasarımlarını derleme, zamanlama analizlerini gerçekleştirme, RTL şemalarını inceleme, programmer ile hedef cihazı (target device) yapılandırabilme gibi birçok özellikere sahip.

Quartus II'nin iki tane kullanım seçenekleri vardır, web baskılı ve tam lisanslı. Web baskılısında ise Alteranın web sitesinden indirilebilir ya da e-mail ile teslim alınabilir. Web baskılısı Alteranın sınırlı sayıdaki cihazları derleme ve programlamayı sağlar. FPGA'in düşük maliyetli Cyclone ailesi bu baskı tarafından desteklenir, aynı anda CPLD'nin MAX ailesi de, yani küçük geliştiriciler ve eğitim amaçlı kullanılabilir. Web baskılının lisans kaydı istenildiği sürece kullanıla bilir ve yenilene bilirler.

Tam lisanslı baskısında ise programın tam işlevselliğini kullanabilmek için ödemeli olarak indirilebilir.

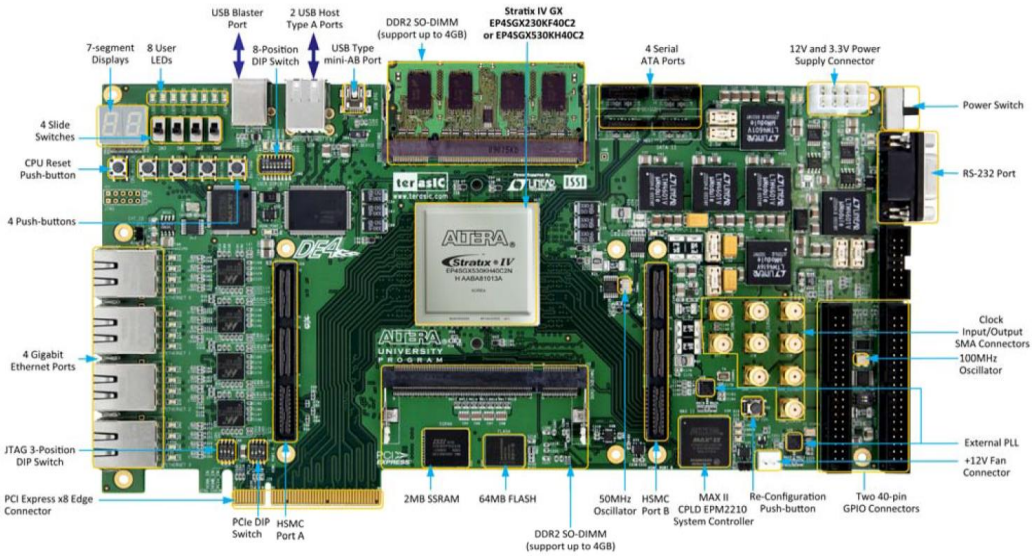


Şekil 4.4 Quartus II Yazılım Aracı Arayüzü

BÖLÜM 5

GERÇEK ZAMANLI UYGULAMA

Uygulamada Alteranın Stratix IV GS geliştirme bordu kullanılmıştır (Şekil 5.1).



Şekil 5.1 DE4 FPGA geliştirme bordu[26]

Genel birim (module) tasarım paralel ve ardışık (paralel & pipelined) mimariye dayanmaktadır. Altera cihazın mimarisi Stratix IV ailesinin tek bir FPGA cihaz üzerinde uygulanmıştır. Ana işleme kartı Stratix IV GX EP4SGX230 FPGA ve verileri saklamak için DDR2 SO-DIMM hafızasına sahip Altera DE4 Uygulama ve Geliştirme Stratix IV bordudur. 2 adet 2,592 H x 1.944 V aktif benek destekleyen ve saniyede tam çözünürlüklü 15 kare hıza kadar yüksele bilen, çıkış verisi RGB Bayer Modeli formatında 5 Mega Benek Dijital Kamera Paketi ve görüntüyü monitöre vermek için Yüksek Hızlı Ezanine Bağlayıcı (HSMC) kartı kullanılmıştır.

5.1 DE4 Stratix IV GX cihazı

Altera'nın DE4 Geliştirme bordu, yüksek performanslı, seri bağlantı ve gelişmiş bellek arayüzü ile sistem tasarımları için ideal bir donanım platformu sağlamaktadır. Daha büyük bant genişliği, geliştirilmiş jitter performansı ve düşük güç tüketimi gibi birçok son pazarlamada hızla gelişen gereksinimlerine yanıt bulmak için özel olarak geliştirilmiş. DE4 Stratix IV GX cihazı endüstri standardı çevre birimleri tarafından desteklenerek, bağlayıcı ve ara yüzleri bağlantıları geniş sahalı yoğun işlem uygulamalara (compute-intensive applications) uygun birçok özellikleri sunmaktadır.

Altera DE4 Stratix IV GX geliştirme ve eğitim bordunun teknik özellikleri aşağıdaki şekildedir.

Stratix IV GX EP4SGX230

•228.000 mantık öğeleri (LES)

•17.133 K toplam bellek Kbit

•1288 18x18 bit çarpan blokları

•2 adet PCI Express, sabit IP blokları

•744 kullanıcı I / O

•8 faz kilitlemeli döngüler (PLL'ler)

FPGA Yapılandırma (configuration)

•JTAG ve Hızlı Pasif Paralel (FPP) yapılandırma

•On-board USB Blaster

Bellek Sürücüsü (memory devices)

•16-bit veri yolu ile 64 MB bellek

•2 MB ZBT SSRAM

•I2C EEPROM

İki adet DDR2 SO-DIMM Yuva (sockets)

•400 MHz saat hızı

•102 üzerinden Gbps maksimum teorik bant genişliği

•Toplam kapasitesi 8 GB'te kadar

SD Kart Yuvası (socket)

•SPI ve SD Kart erişim için 4-bit SD modu sağlar.

Düğmeler, Anahtarlar ve LED'ler

•4 butonlar

•4 slayt anahtarları

•8 LED

•8-pozisyon DIP anahtarı

Yerleşik saateleri (**On-Board clocks)**

•FPGA üzerinden yapılandırabilen 3 programlanabilir PLL'ler

•HSMA, HSMB alıcı-verici saat kaynağı

•SATA referans saat

•FPGA LVDS saat giriş

•50MHz/100MHz osilatör

•SMA Bağlayıcılar

2 harici verici saat girişi için SMA bağlayıcı (SubMiniature version A)

•LVDS saat giriş / çıkış için 4 SMA bağlayıcı

•2 saat çıkışı için SMA bağlayıcıları

•Harici saat girişi için 1 SMA bağlayıcı

Dört Seri ATA Portu

•SATA 3.0 standart 6Gbps sinyal oranı

•İki ana ve iki cihaz bağlantı noktası (ports)

Dört adet Gigabit Ethernet Portları

•Entegre 1.25 GHz SERDES

11.PCI Express x8 Kenar Bağlayıcı

•5.0Gbps/lane az Gen2 için 2.5Gbps/lane Gen1 Destek bağlantı hızı

•Bağlantı x8 veya x16 PCI Express yuvasına sahip anakart ile kurulan

İki adet 172-pin Yüksek Hız Mezzanine kartı (HSMC)

•2 dişi HSMC bağlayıcılar

•I / O voltajı 2.5V

•8.5 Gbps Toplam 12 yüksek hızlı alıcılar

•Toplam 38 LVDS çiftinin 1.6 Gbps

İki adet 40-pin Genişleme Başlıkları

•72 FPGA I / O pini yanı sıra 4 güç ve topraklama hatları, iki adet 40-pin genişleme bağlantıları için çıkartılır

•40-pin IDE sabit diskler için kullanılan bir standart 40-pin şerit kablo başlığı kabulu için tasarlanmıştır

•I / O voltajı 3.0V

USB ana sistem (host) / bağlı sistem (slave) Denetleyiciler

•USB rev 2.0 teknik özellikleri ile tam uyumludur

•Tam hız, yüksek hız ve düşük hızda veri transferi sağlar

•Her iki USB ana sistemi ve cihaz desteği sağlar

•Üç adet USB bağlantı noktası (tek mini-AB tipi ana sistem / cihaz için ve iki A tipi ana sistem için

•I / O (PİO) ve Doğrudan bellek erişimi (DMA) için Programlı Destekli

Güç

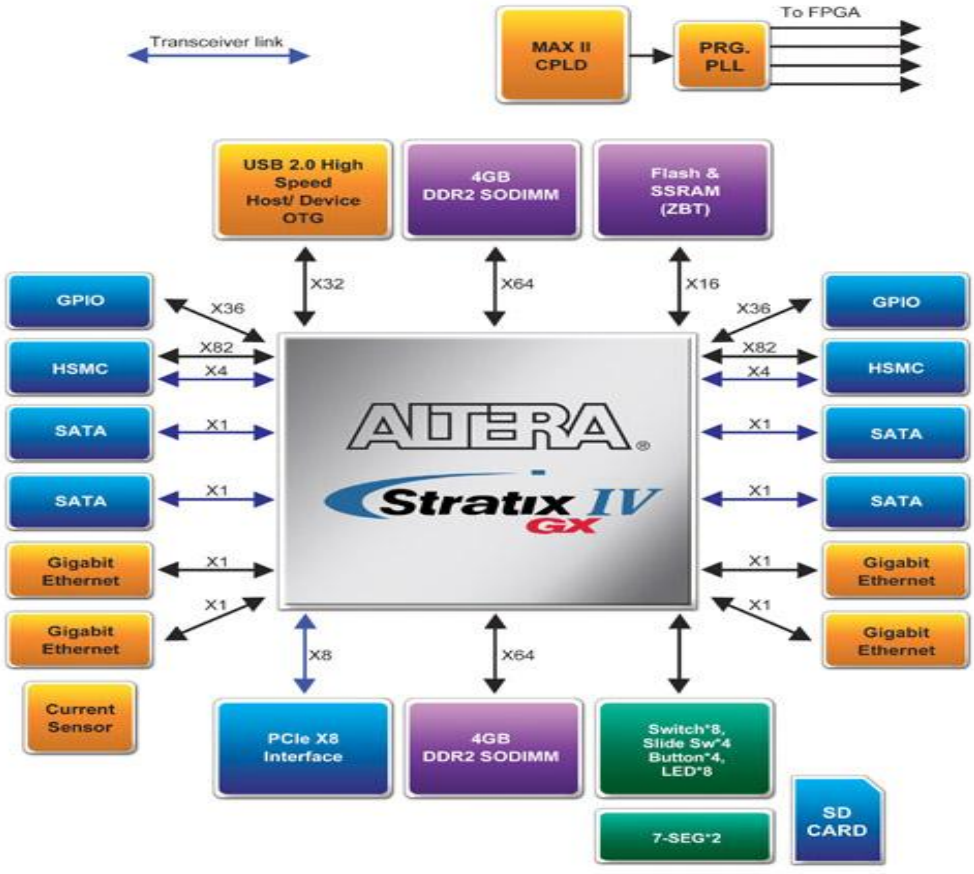
•DC giriş 12V ve 3.3V

•PCI Express kenar bağlayıcı gücü

•PCI Express dış standart güç kaynağı Destekli, •Yerleşik (On-Board) güç ölçüm devresi.

Stratix IV GX FPGA platformunun avantajı tümleşik (integrated) alıcı (transciever) ile PCI Express standardının 2.0 sürümü ile tam uyumlu olması. Bu durum müşterilerin yüksek hızlı bağlantılı, geniş kapsamlı uygulamaları için tasarımları sağlayan PCI Express tabanlı uygulamaların ana gelişimini hızlandırmaktadır.

DE4 seri ATA (SATA) arabirimi ile birleştiğinde, depolama uygulamaları geliştirmek için bir çözüm sunuyor. DE4, PCI Express built-in blokları için SATA alıcı doğrulama testleri ve Gigabit Ethernet protokolü testleri sunmaktadır. DE4 daughter kartları ile ölçekleme ve özelleştirmeye (scaling and customization) izin veren iki Yüksek Hızlı Mezzanine kartı (HSMC) bağlayıcıları desteklemektedir. Aşağıdaki blok diagramında daha detaylı olarak gösterilmektedir.



Şekil 5.2 Stratix IV GX Blok Diagramı[26]

5.2 Yüksek Hızlı Mezzanine Bağlayıcı (HSMC) DVI Kartı

Terasic HSMC-DVI kartı, Yüksek Hızlı Mezzanine Bağlayıcı (HSMC) arayüzü ile birlikte bir DVI verici / alıcı kartıdır. Geliştiricilerin (developers) FPGA de kaliteli ve yüksek çözünürlüklü (UXGA Resolution (Pixel Rates up to 165 MHz)) video sinyallerine erişilmesi için tasarlanmıştır. Aynı kart üzerinde DVI alıcı ve vericiyi birleştirerek yüksek çözünürlüklü görüntü işleme sistemlerinde gerekli esnekliği elde edilebilmekte. HSMC-DVI daughter kartı herhangi bir host board HSMC / HSTC arayüzü ile bağlanabilir. HSMC / HSTC arayüzü sayesinde tüm Terasic / Altera bordları ile uyumluluk sağlar.



Şekil 5.3 DVI kartı[26]

DVI-HSMC Kartının özellikleri şu şekildedir.

Dijital Verici

- Tek aktarma noktası ile bir DVI verici
- Dijital Görsel Arabirim, Digital Visual Interface (DVI) Uyumlu
- VGA UXGA çözünürlük (25 MHz - 165 MHz Benek değerleri) destekler

Evrensel Grafik Denetleyicisi Arabirimi (Universal Graphics Controller Interface)

- 12-Bit çift-kenar ve 24-Bit Tek-kenar girdi modları
- Ayarlanabilir 1,1 V - 1,8 V ve Standart 3,3 V CMOS Giriş Sinyal seviyeleri

- Tam Diferansiyel ve Tek uçlu Giriş saat modları
- Standart Intel ™ 81x Chipsete uyumlu olarak 12-bit Dijital Video portu

Geliştirilmiş PLL Gürültü bağışıklığı (Enhanced PLL Noise Immunity)

- On-Chip Regülatörleri ve sistem maliyetlerinin azaltılması için Bypass Kapasitörleri

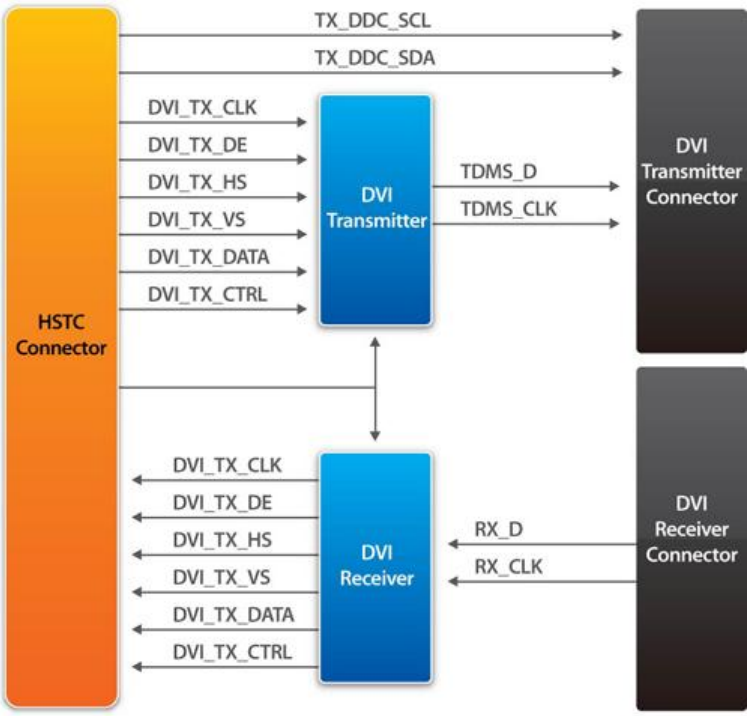
Geliştirilmiş jitter performansı

- HSYNC Jitter Anomali'siz, İhmal edilebilir veri bağımlı jitter
- I^2C Seri arabirim kullanarak programlanabilir
- Tek 3.3-V besleme operasyonu

Dijital Alıcısı

- Tek alıcı portu ile bir DVI alıcısı
- UXGA çözünürlüğü destekler (Çıkış Benek değerleri en fazla 165 MHz kadar)
- Dijital Görsel Arabirim, Digital Visual Interface (DVI) özelliklerine uyumlu
- True-Color, 24 Bit / benek, 16.7M renklerde saat başına 1 veya 2-benek
- Lazer Trimmed Optimum sabit eşleme için dahili sonlandırma dirençleri
- 4x Örneklem Üzerinde (Over-Sampling)
- Önerilen zamanlamanın benek çıktılarını kullanarak topraklama zıplamalarını azaltmak
- TI PowerPad ambalajı kullanarak iyi güç dağıtma ve en düşük gürültüler üretebilme.

Boyut: 41 x 78.11 mm



Şekil 5.4 HSMC DVI Kartı Blok Diagramı[26]

DVI Kartının uygulanan alanları aşağıdaki alanlardır

- Makine görmesi
- Video telefon
- Uzaktan görüntü algılama
- Video streaming
- Görüntü tanıma, filtreleme ve sıkıştırma
- Biyometri
- Gözetleme sistemleri

Bağlantısı, arayüz dönüşümü için bir HSMC bağlayıcı ile HSMC / HSTC arayüzü sayesinde tüm Terasic / Altera bordları ile uyumludur.



5.3 5-Mega Benek Dijital Kamera

Micron 5 Mega Benek CMOS sensör ile donatılmış, Destek 2592 Y x 1944 V aktif benek, RGB Bayer Pattern formatında çıkış verileri, Tam çözünürlük kare hızı, saniyede 15 fps. Kullanıcıların tüm referans tasarımını sağlayan (Frame Grabber, yüksek performanslı çok-portlu SDRAM çerçeve arabelleği, görüntü işleme IP'leri), desteklediği bordlar DE3 / DE2_70 / dE2 / DE1 /DE4 ve Altera Cyclone II Starter panoları, Boyutu 78 x 59,5 mm dir.

5 Mega benek dijital kamera geliştirme paketi Terasic FPGA sistem üzerinde bir 5 Mega benek dijital kamera geliştirmek için gereken her şeyi sağlıyor. Çekilen resimleri bilgisayara BMP/JPG olarak kayd etmek için Verilog'da donanımsal tasarımı ve yazılımı mevcuttur.

Parameter		Value
Active pixels		2,592H x 1,944V
Pixel size		2.2μ m x 2.2μ m
Color filter array		RGB Bayer pattern
Shutter type		Global reset release (GRR),
Maximum data rate/master clock		96 Mp/s at 96 MHz
Frame rate	Full resolution	Programmable up to 15 fps
	VGA (640 x 480)	Programmable up to 70 fps
ADC resolution		12-bit
Responsivity		1.4 V/lux-sec (550nm)
Pixel dynamic range		70.1dB
SNRMAX		38.1dB
Supply Voltage	Power	3.3V
	I/O	1.7V ~ 3.1V

Şekil 5.5 Kameranın Performans Parametreleri[26]

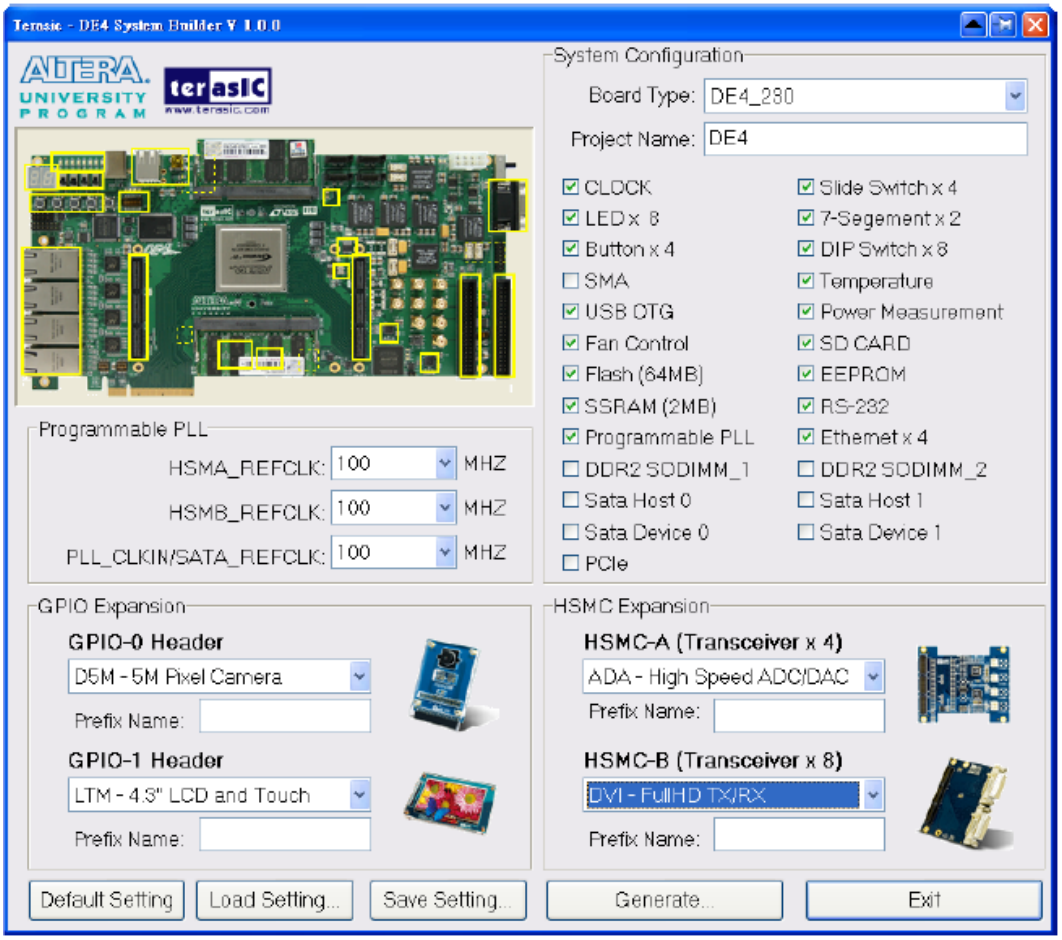


Şekil 5.6 Hizalanmış 5 Mega Benek Kamera

5.4 DE4 System Builder

Stratix IV GX Altera DE4 kartının en çok beğenilen bir özelliğide "System Builder"i olmasıdır. System Builder kullanıcıya Quartus II projesi oluşturmada kolaylık sağlayan Windows tabanlı yazılımdır (Şekil 5.7). Kullanıcı kolaylıkla Quartus II projelerini oluşturabilmektedir, oluşturulan projeler genellikle şu şekildedir:

- Quartus II Project File (Proje Dosyası)(.qpf)
- Quartus II Setting File (Ayar Dosyası)(.qsf)
- Top-Level Design File (Üst Düzey Tasarım Dosyası)(.v)
- External PLL Contorller (Dış PLL Contorller)(.v)
- Synopsis Design Constraints file (Özet Tasarım Koşulları dosyası)(.sdc)
- Pin Assignment Document (Pin Atama Belgesi)(.htm)



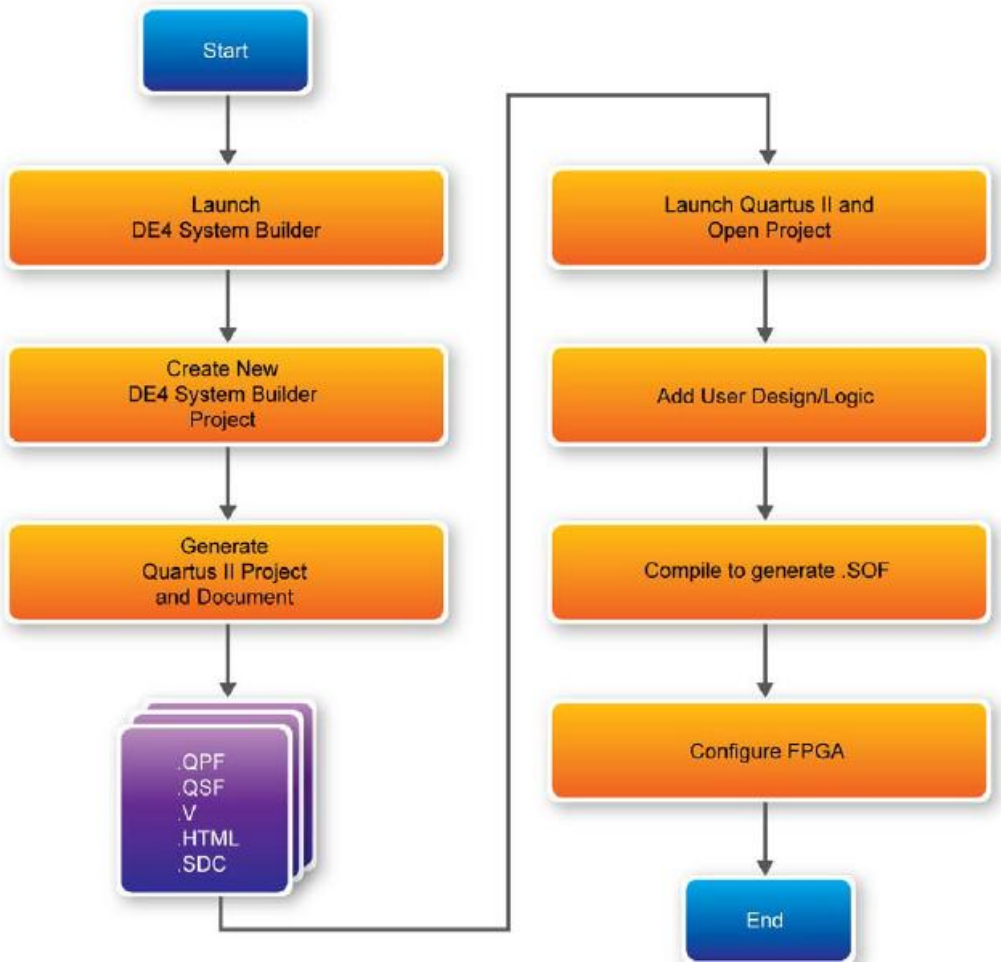
Şekil 5.7 System Builder arayüzü[26]

DE4 Sistem Builder ile DE4 bordun'da bir proje oluşturmak için genel tasarım akışı (Şekil 5.8)'de gösterilmektedir. Kullanıcılar DE4 Sistem Builderi başlatarak, tasarım ihtiyaçlarına göre yeni bir proje oluşturmaları gerekir. Kullanıcılar projeye yönelik ayarlarını tamamladığında, DE4 System Builder, üst düzey tasarım dosyası (.v) ve Quartus II ayar dosyası (.qsf) olarak iki büyük dosyaları oluşturur.

Üst düzey tasarım dosyası, kullanıcıların kendi tasarımı / mantığını eklemek için üst düzey Verilog sarmalayıcı (wrapper) içerir.

Quartus II ayar dosyası ise FPGA cihaz türü, üst düzey pin atamaları ve her kullanıcı tanımlı giriş-çıkış (I / O) pinleri için giriş-çıkış (I / O) standartı olan bilgiler içerir.

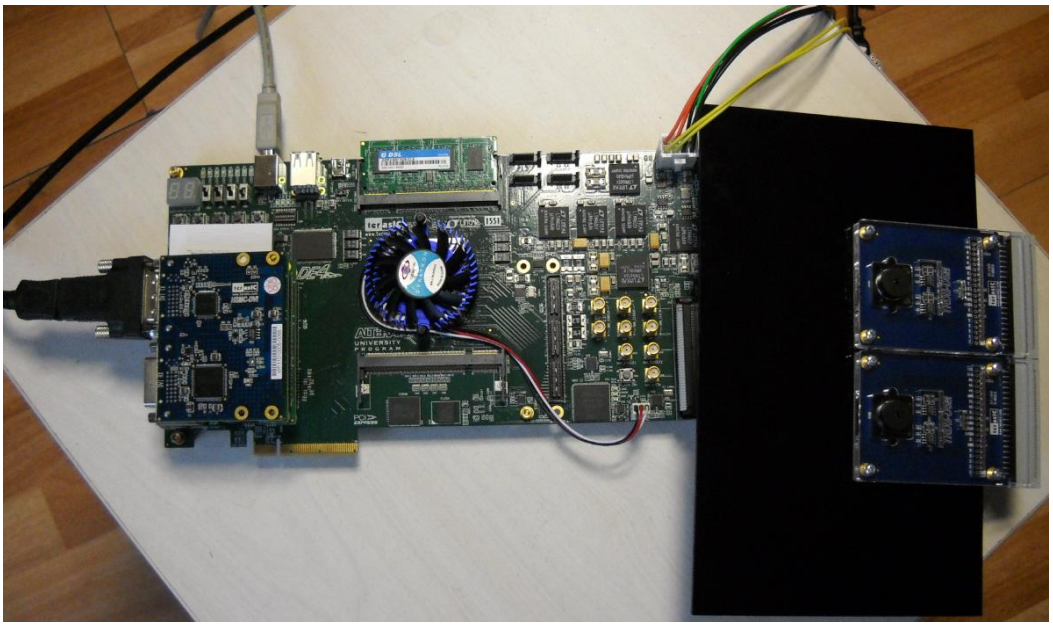
Son olarak, Quartus II programlayıcısı JTAG arabirimini kullanarak DE4 borduna SOF dosyasını indirerek programı programlamış olur.



Şekil 5.8 Bir yapı tasarımının genel tasarım akışı[26]

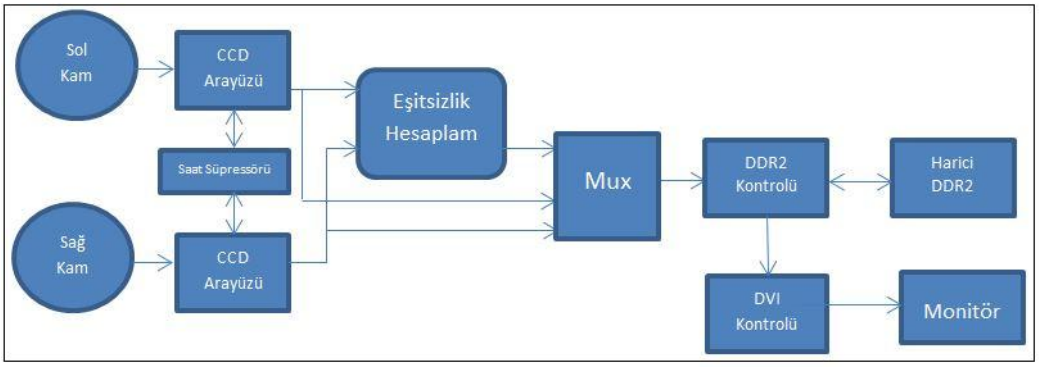
BÖLÜM 6

UYGULAMA KURULUM DETAYLARI



Şekil 6.1 Stereo Kamera ve DVI kartlı ile Stratix IV FPGA geliştirme bordu

Donanımsal uygulamanın kurulumunda Stratix IV GX EP4SGX230 FPGA'e 2 tane 5 Mega benek kameralar bir plakanın üzerine sabitlenerek vidalanmıştır. Yazılım tarafından benekleri kaydırılarak aynı hizaya (align) getirilmiştir.



Şekil 6.2 Sistemin donanımsal tasarımı

Kullanılan yöntem daha önce Georgulas et al [25] birimi ile önerilmiştir. Mutlak Farklar Toplamı (SAD) algoritması. Benek tabanlı kayıklık harita üretimi için SAD algoritma kullanılmıştır. SAD'a kıyasla daha fazla hesaplama karmaşıklığı ve gerekli işlem sürelerinin artırılması, bu yöntemler ile ilgilenmektedir. Toplam kare farkları, bağıntı görüntü anları (image moments) gibi ölçümlerde kullanılabilirdi ancak hız sorunları dikkate alındığında seçilmemişlerdir. Ayrıca donanımsal uygulamada Mutlak Farkların Toplamının (SAD) basitliği nedeniyle tercih edilmiştir.

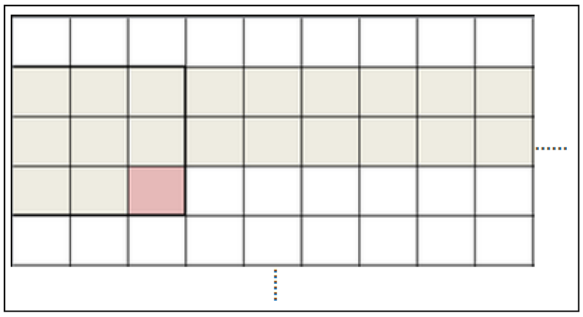
Bu çalışmada, donanımsal kurulum aşamasında, gerçek zamanlı olarak kayıklık haritası çıkartmak için stereo görme eşleşme problemi ele alınmıştır. Tüm işlemler gerçek zamanlı olarak, yüksek hız gerektiren uygulamadır. Yüksek hızlı birim tarafından, saniyede 15 kare hızda (fps) 1280x1024 benek çözünürlüklü stereo çifti ile kayıklık haritası hesaplanmıştır.

Donanımsal uygulamada tek bir Altera Stratix IV GX DE4 FPGA'de uygulanmıştır. FPGA yüksek işlem gücüne sahip tasarımları sağlamaktadır. ASIC cihazlara kıyasla FPGA'in önemli bir avantajı yapılandırabilirliğe (reconfigurability) sahip olmasıdır. Genel birimin tasarımı paralel, ardışık (pipelined) mimariye sahip olmasıdır.

Sistemin yapısı Şekil 6.2'de özetlenmiştir. Donanımsal tasarımda iki kameradan GPIO portu ile alınan girdiler, sağ ve sol kameralardandır. Girişler, CCD'nin işlenmemiş görüntü (raw image) çözümleme (decoding) ve kontrol birimden oluşan CCD arayüz birimi içinde beslenmektedir. Kameralardan 12-bitlik RGB olarak alınan görüntü 12-bit gri tonlu görüntüye dönüştürülerek stereo eşleme birimine aktarılır. Stereo eşleme

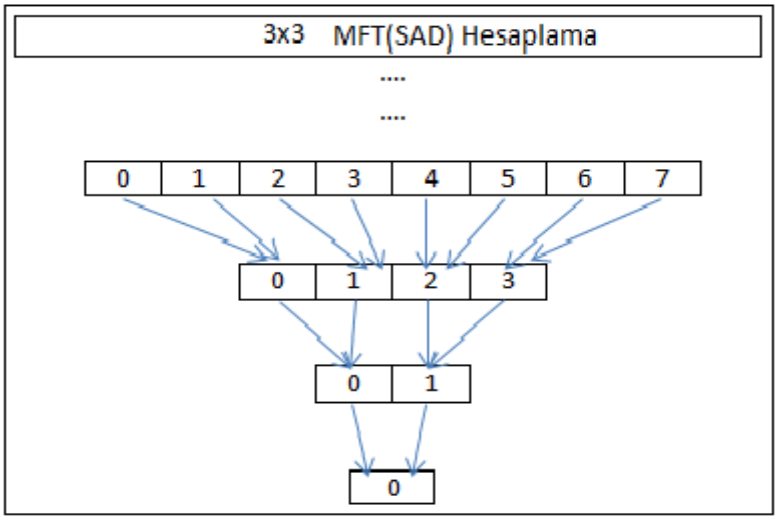
biriminde, iki kameranın son 3 aktif satırını saklayan hafıza blokları bulunur. Bu hatların kullanımını döndürmek için yardımcı olan bir durum makine kullanır.

En iyi blok eşleşmesi 3x3 pencere ile hesaplanan minimum mutlak fark toplamı (SAD) değerinin bulunması ile belirlenir. Stereo eşlemeyi paralel olarak yapabilmek için sağ kameradan 3x128 ve sol kameradan da 3x3 benek bilgisini saklayan bir fifo yapısı vardır (Şekil 6.3).



Şekil 6.3 Hat ve arama bufferleri

En iyi eşleşen blokları bulmak için minimum SAD'dı arayarak 3x3'lük bir pencere boyutu kullanılmıştır. Verimliliği arttırmak için, bir ardışık (pipelinned) ikili karşılaştırma ağacı donanımsal olarak tasarlanmıştır (Şekil 6.4). Sonuç olarak bulunan minimum SAD değeri belirli bir seviyenin üstünde ise, bunun kaynak endeksi o pencere için kayıklık değerini verir. Karşılaştırma ağacın sistem yapısı genel olarak (Şekil 6.4) de gösterilmiştir.



Şekil 6.4 SAD'ın en küçüğünü bulmak için ardışık düzenli çalışan ikili ağaç kullanımı.

Sol kamera ile çekilen orijinal renkli görüntü ve Sağ kamera ile çekilen orijinal renkli görüntüler ve bu iki kameradan alınan görüntülerden hesaplanmış kayıklık haritasının çıkış sinyalleridir. Resim akımının biri dış uyumlu (synchronous) DDR2 yongası için bir geçici bellek depolama olarak hareket eden bir FIFO (ilk-giren, ilk-çıkar) biriminde beslenir. DDR2 DVI çıkışı için meşgul olmadığı sürece, veri DDR2 FIFO'yu okur ve DDR2 içinde depolanır. DDR2 denetleyici (controller) bu işlemi üstlenir.

Kayıklık haritası hesaplanmasında birim, sol ve sağ kameralar için görüntü benek verisini stereo eşleme modülünden alır ve aşağıdaki formüle göre her bir kayıklık değeri için Mutlak Fark Toplamı (SAD) hesaplanır.

$$SAD(i, j, d) = \sum_{\mu=-w}^{w} \sum_{v=-w}^{w} \left| I_l(i+\mu, j+v) - I_r(i+\mu, j-d+v) \right|$$

Çalışmada bir 3x3 pencere kullanılıyor, buna göre SAD basitleştirilmiştir.

$$SAD(i, j, d) = \left| I_l(i, j) - I_r(i, j-d) \right|, \; 0 < d < 127$$

Stereo eşleme birimi iki kameranın 128 gri seviyeli beneklerini saklanır. Sol görüntüden her benek için en iyi eşleşme beneği sağ görüntü içinde aranır. Her aday benek için mutlak fark hesaplanır ve kayıt dizide (register array) depolanır. Donanımda ikili ağaç (binary tree) karşılaştırma algoritması kullanılarak uygulanır. En iyi eşleşme beneki ve kayıklık değeri 8 döngüde bulunur ve büyük ölçüde mantık akışı arttırır. Bu ardışık (pipelined) mimarisi gerçek zamanlı olarak çok hızlı eşleşen sonuç vermiştir.

Ayrıca en önemli bir kısım da kameraları senkronize etmektir. Donanımsal tasarımında kamera senkronize işlemi saat bastırıcı (saat suppressor) ile yapılmıştır. Saat baskılayıcı iki kameranın saat sinyallerini aynı hizaya ayarlar. Saat bastırıcının asıl amacı, iki kameradan birinin birine doğrulanmadığında işlemi senkronize etmektir. Eğer kameralar aynı anda (simultaneously) senkronize ise, aynı pozisyondaki kameraların benekleri karşılaştırılarak gerçek zamanlı olarak eşitsizliği hesaplayabiliriz. Bir kamera ileride ise, saat baskılayıcı diğer kamera ulaşıncaya kadar bu kamerayı besleyen saati durur. Kamera yeni bir frame işlemine başlıyorken, framemin pozitif kenarı geçerli sinyali gösterdiğinde bu işlem çerçevenin (frame) sonunda gerçekleşir. Eğer bir kamera

diğer kameradan önce yükselen frame geçerli bir sinyal gösterdiğinde onun saati durdurulmuş olur.

Kayıklık haritası ve sağ veya sol kameraların görüntüleri DVI-D veya uyumlu arayüzü olan bir LCD monitör üzerinde görüntülenir. DDR2 Bellek video görüntüsü bufferi olarak kullanılır. Bu tasarımda DDR2 bellek arayüzü olarak Alteranın DDR2 IP core Denetleyici kullanılmıştır. FIFO'ları yazmak ve okumak için DDR2'nin performansını kullanılmıştır.

FIFO'nun görevi DDR2'ye kaydedilmesi gereken veri akışı için bir buffer sağlamaktır. Kameralar tarafından çıkış verilerin sürekli akışı nedeniyle DDR2'ye kaydedilmesi gereken bir sürekli veri akışı olmasına rağmen buffer gerekli, DDR2 DVI'ye veri çıkışı sırasında zaman harcamaktadır. DDR2 DVI'e çıktı vermekle meşgülken buffer DDR2'ye veri yazmayi önler.

Ayrıca DDR2 durumunu kontrol etmek için DDR2 denetleyiciye ihtiyacımız vardır. DDR2 denetleyicinin görevi, FIFO taşma yapmasın diye DVI denetleyiciden düzenli okuma isteklerini tutmak ve DDR2 FIFO tarafından bufferin girdi akışını gözetlemektir. Kurulum sonuç görüntüleri bölüm 7'de gösterilmiştir.

BÖLÜM 7

UYGULAMA SONUÇLARI

7.1 MATLAB Uygulaması

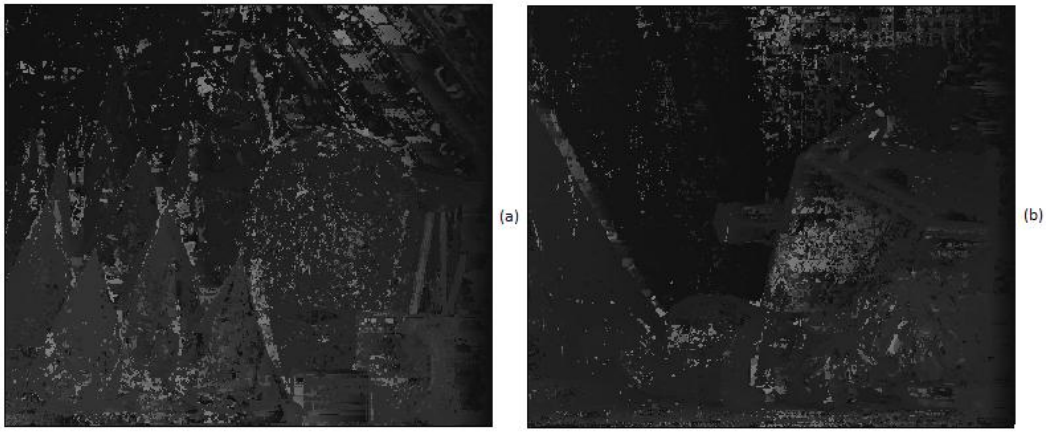
Bu bölüm önerilen algoritmanın FPGA ve MATLAB'ta uygulanarak kayıklık haritası elde edilmesi hakkındadır. MATLAB uygulaması sonuçları Middlebury Stereo çiftleri Teddy ve Koniler orijinal resimleriyle denenmiştir. Middlebury tarafından elde edilen stereo çiftleri sonuçları (Şekil 7.1)[35].



Şekil 7.1 Middlebury stereo çiftleri (a) Teddy (b) Koniler [35].

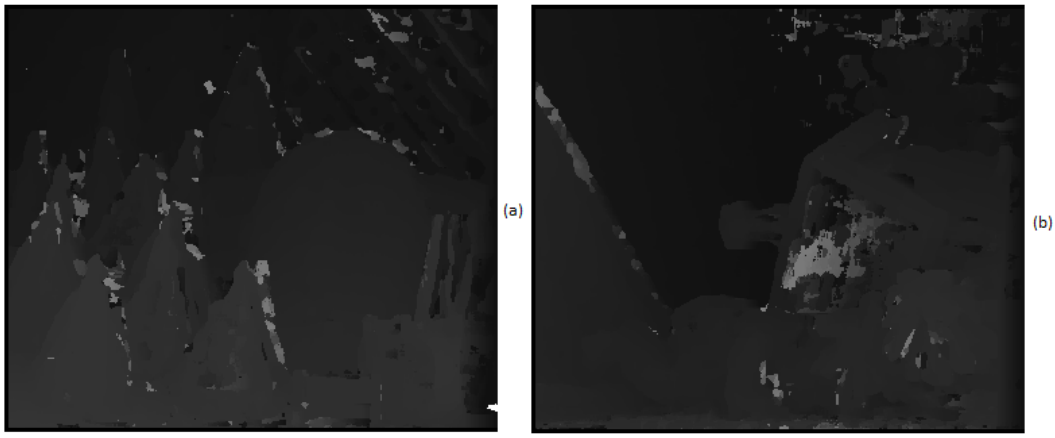
Uygulama kurulmunda, geliştirilen algoritma iki şekilde kullanılarak başarılı sonuç elde edilmiştir. Birincisi, algoritma geleneksel bilgisayarda MATLAB programında uygulanmıştır. Uygulamada middlebury veri tabanıda kullanılan orijinal resimler

kullanılarak ilk olarak 3x3'lük bir pencere boyutunda denendiğinde biraz gürültülü sonuç alınmıştır (Şekil 7.2).



Şekil 7.2 Matlab 3x3 pencereli (a) Teddy (b) Koniler sonucu

İkinci olarakta 8x8'lik bir pencere boyutunda denendiğinde daha belirgin bir sonuç elde edilmiştir (Şekil 7.3).



Şekil 7.3 Matlab 8x8 pencereli (a) Teddy (b) Koniler sonucu

MATLAB uygulamasında doğruluğu (accuracy) daha iyi hesaplamak için çift kayar nokta (double floating point) kullanır.

7.2 FPGA Uygulaması

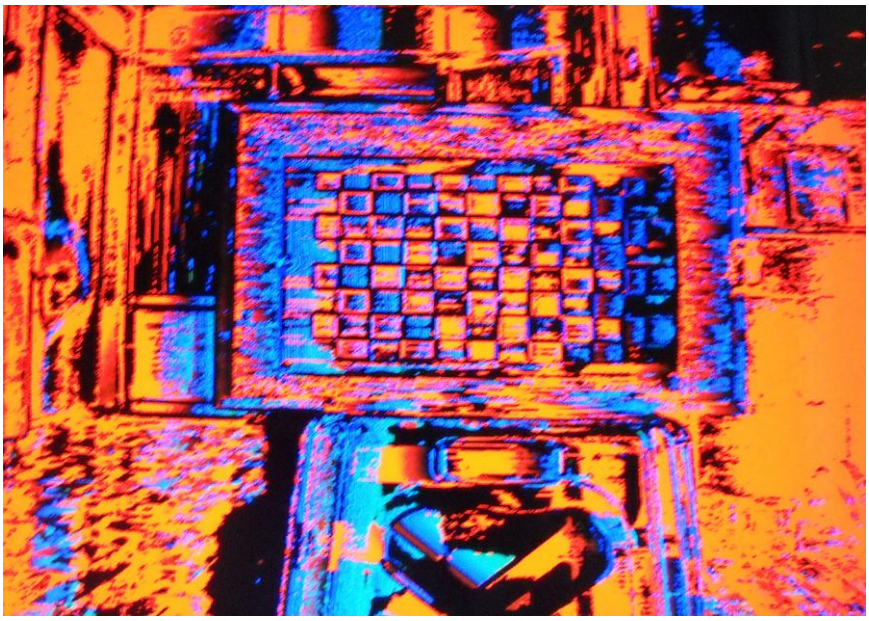
FPGA uygulamasında iki kameradan alınan görüntünün aynı hizada olmasını görebilmek için sağ kamera görüntüsünün çıkış verisini mavi renkte sol kameranın çıkış verisi ise kırmızı renkli olarak ekrana yazdırdığımızda iki kamera farkları gözükmekte.



Şekil 7.4 Sağ ve Sol kamera görüntüleri



Şekil 7.5 Sağ ve Sol kamera görüntü farkları

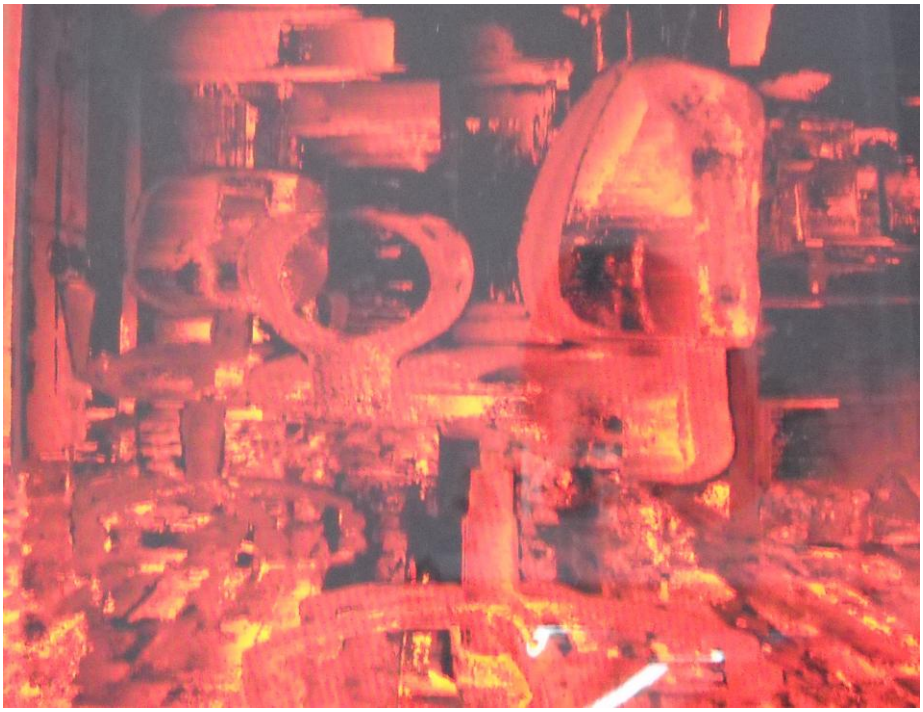


Şekil 7.6 Kameradan 3B görüntü

Donanımsal uygulamada da iki farklı pencere boyutları ile denenerek sonuçlar elde edilmiştir.



Şekil 7.7 FPGA 3x3 pencereli RGB görüntü



Şekil 7.8 FPGA 3x3 pencereli 3B görüntü

Donanımsal uygulamada stereo eşleştirme alanı 128 benektir. 128 beneklik görüntü kayması kameradan 70 cm uzaklıktaki nesneler ile oluşmaktadır. Bu nedenle sistem 70 cm'den daha yakın nesneler için doğru bir eşleme yapamaz. Kameradan nesne uzaklaştıkça renk değeri değişecektir. 3x3 pencere boyutlu aramada sonucu şekil 7.8'de gözüktüğü gibi elde edilmiştir.



Şekil 7.9 FPGA 8x8 pencereli RGB görüntü



Şekil 7.10 FPGA 8x8 pencereli 3B görüntü

8x8 pencere boyutlu aramada ise nesneleri daha belirgin bir şekilde tanımakta. Arama penceresi büyüdükçe hata algılaması azılmakta.

Sonuç olarak önerilen algoritma MATLAB programında ve FPGA'de uygulanarak sonuçlar elde edilmiştir. İlk olarak MATLAB'ta uygulandı, sonra Stratix IV GX EP4SGX230 FPGA bordunda donanımsal olarak uygulandı. FPGA uygulamasında 3x3 penrece boyuntunda uygulandığında derleme süresi yaklaşık olarak 1 saat sürede, % 43 mantık kullanımı ve 72,545 / 182,400 (40 %) kombinasyonel aritmetik mantık birimi kullanarak sonuca ulaşmıştır. 8x8 pencere boyutunda uygulandığında ise derleme süresi yaklaşık olarak 5 saat sürede, % 85 mantık kullanımı ve 150,435 / 182,400 (82 %) kombinasyonel aritmetik mantık birimi kullanarak sonuca ulaşmıştır.

BÖLÜM 2

SONUÇ VE ÖNERİLER

Bu tezde önerilen algoritma iki şekilde uygulanmıştır ilk olarak MATLAB programında denenmiştir. Sonrada gerçek zamanlı olarak FPGA'de denenmiştir.

Çalışma kapsamında Stereo Görme Sisteminin FPGA'de tasarımı ve uygulaması gerçek zamanlı olarak geliştirilmiştir. Stereo Görme yöntemleri günümüzde en yaygın olarak araştırılan konulardan biri olsada donanımsal tabanlı uygulamaları oldukça zordur.

Stereo görme, bilgisayar görme alanında üç boyutlu yapı bilgisini elde etmek için en sık kullanılan ve araştırılan yöntemlerden biridir. Stereo görme yöntemi temel olarak bir sahnenin aynı anda iki ya da daha fazla görüntüsünün alınmasına ve bu alınan görüntüler üzerindeki kaymaların hesaplanmasına dayanır. Sahne üzerindeki bir nokta iki farklı açıdan çekim yapan kameraların resim düzlemlerinde farklı konumlara düşecektir.

Robotik uygulamalarda insanın fiziksel olarak erişemeyeceği noktalarda kullanılmakla beraber 3B nesne algılama navigasyon, yol planlama, haritalama ve lokalizasyon, gibi birçok gerçek zamanlı uygulamalarda kullanım nedeniyle Bilgisayar Görme alanının en önemli konulardan biridir.

Bu Tez çalışmasında Altera'nın DE4 Stratix IV GX EP4SGX230 Geliştirme ve Uygulama bordunu ve 2 tane 5-Mega Benek Dijital Kamera kullanarak, Stereo eşleme problemlerinden Alan-Tabanlı yaklaşımıyla, Mutlak Farklar Toplamı (SAD) algoritması ile kayıklık bulma işlemi, sağ ve sol resimler arasındaki uygunluğu bulma gerçek zamanlı ve donanımsal olarak gerçekleştirilmiştir. İlerideki çalışmalarda, yerel görüntü özelliklerine göre adaptif pencere boyutu seçimi kullanmaya çalışılacaktır.

KAYNAKLAR

[1] Se, S. ve Jasiobedzki, P., (2008). "Stereo-Vision Based 3D Modeling and Localization for Unmanned Vehicles", International journal of intelligent control and systems, 47–58.

[2] Jin, S., Cho, J., Pham, X., Lee, K. M., Park, S.K., Kim, M. ve Jeon, J. W., (2010). "FPGA Design and Implementation of a Real-Time Stereo Vision System", Transactions on circuitsand systems for video technology,20-1.

[3] Mario, A., Manzano, I., Almanza, O., Michel, D., Jean-Louis. B. ve Jean-Yves., F., (2009). "Stereo vision algorithm implementation in FPGA using census transform for effective resource optimization", 12th Euromicro Conference on Digital System Design / Architectures, Methods and Tools.

[4] Divyang, K., Masrani, W. ve MacLean, J., (2006). "A Real-Time Large Disparity Range Stereo-System using FPGA", Proceedings of the Fourth IEEE International Conference on Computer Vision Systems.

[5] Wang, J., Hsieh., R. ve Chiu, H. C., (1999). "A Progressive Constraint Search Approach for Disparity Matching in Stereo Vision", Proceedings of the National Science Council, 789–798.

[6] Bilgin, A. ve Ulusoy, I., (2009). "Selection and Fusion of Multiple Stereo Algorithms for Accurate Disparity Segmentation", Sinyal İşleme ve İletişim Uygulamaları Konferansı.

[7] Scharstein, D. ve Szeliski, R., (2002) "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms", 7–42.

[8] Cohen, L., Vinet, L., Sander, P.T. ve Gagalowicz. A., (1986). "Hierarchical region based stereo matching", Computer Vision and Pattern Recognition, Proceedings CVPR, 416–421.

[9] Canny, J., (1986). "A Computational Approach To Edge Detection", IEEE Trans. Pattern Analysis and Machine Intelligence, 8.679–714.

[10] Lee, Y.S., Koo, H.S. ve Jeong, C.S., (2010). "A straight line detection using principal component analysis", Pattern Recognition Letters 27(14): 1744-1754.

[11] Özsaraç, H.S., (2010). FPGA Implementation of Graph Cut Method for Real Time Stereo Matching, METU.

[12] Faugeras, O., Hotz, B., Mathieu, H., Vieville, T., Zhang, Z., Fua, P., Theron, E., Moll, L., Berry, G., Vuillemin, J., Bertin, P., ve Proy, C., (1993). “Real time correlation-based stereo: algorithm, implementations and applications”, Technical Report Research Report 2013.

[13] Kanade, A., Yoshida, K., Oda, H., Kano, M. ve Tanaka., (1996). “A stereo machine for videorate dense depth mapping and its new applications”, Proceedings of the 15th IEEE Computer Vision & Pattern Recognition Conference, 196– 202.

[14] Woodfill, J. ve Von Herzen, B., (1997). “Real time stereo vision on the parts reconfigurable computer”, Proceedings of the 5th Annual IEEE Symposium on Field- Programmable Custom Computing Machines, 201–210.

[15] Darabiha, A., MacLean, W.J. ve Rose, J., (2006). “Reconfigurable hardware implementation of a phase-correlation stereo algorithm”, Journal of Machine Vision and Applications, 116–132.

[16] Diaz, J., Ros, E., Pelayo, F., Ortigosa, E.M. ve Mota, S., (2006). “FPGA based real-time opticalflow system”, IEEE Transactions on Circuits and Systems for Video Technology, 274–279.

[17] Niitsuma, H. ve Maruyama, T., (2005). “High-speed computation of the optical flow”, Lecture Notes in Computer Science: Image Analysis and Processing, 287–295.

[18] Lee, S., Yi, J. ve Kim, J., (2005). “Real-time stereo vision on a reconfigurable system”, Lecture Notes in Computer Science: Embedded Computer Systems, 299–307.

[19] Roh, C., Ha, T., Kim, S. ve Kim, J., (2004). “Symmetrical dense disparity estimation: algorithms and FPGAs implementation”, Proceedings of the IEEE International Symposium on Consumer Electronics, 452–456.

[20] Bradski, G. Ve Kaehler, A., (2008). Computer Vision with the OpenCV Library, OReily, 405–458.

[21] Hartley, R. ve Zisserman, A., (2003). Multiple Viev Geometry in Computer Vision, Cambridge University Pres.

[22] Başak, S., (2011). SelCPU2:Soft-Core Processor Design, YTU.

[23] Muhlmann, K., Maier, D., Hesser, J. ve Manner, R., (2002). “Calculating dense disparity maps from color stereo images, an efficient implementation”, Int. J. Comput, 79–88.

[24] Fleet, D.J., Jepson, A.D. ve Jepson, M., (1991). “Phase-based disparity measurement”, CVGIP: Image Understanding, 198–210.

[25] Georgoulas, C., Kotoulas, L., Ch, G., Sirakoulis, I., Andreadis, A. ve Gasteratos., (2008). "Real-time disparity map computation module." Microprocessors and Microsystems, 159–170.

[26] http://www.terasic.com.tw/en/. Mart 2012.

[27] http://en.wikipedia.org/wiki/Field-programmable_gate_array. Mart 2012.

[28] Parnell, K. Ve Mehta, N., (2002). "Programmable Logic Design Quick Start Hand Book", Xlinx.

[29] www.fpga.gen.tr. Mart 2012

[30] www.altera.com. Mart 2012

[31] http://en.wikipedia.org/wiki/Hardware_description_language. Mart 2012

[32] Wong, S., Vassiliadis, S. ve Cotofana, S., (2002). "A Sum of Absolute Differences Implementation in FPGA Hardware", Euromicro Conference, 183-188.

[33] Hariyama, M., Yokoyama, N. ve Kameyama, M., (2005). "FPGA Implementation of a Stereo Matching Processor Based on Window-Parallel-and-Pixel-Parallel Architecture", Circuits and Systems.

[34] Stefano, L. Di., Marchionni, M. Ve Mattoccia, S., (2004). “A fast area-based stereo matching algorithm”, Image Vis. Comput. 983–1005.

[35] http://vision.middlebury.edu/stereo/. Mart 2012.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : NİLUFAR İSAKOVA

Doğum Tarihi ve Yeri : Tacikistan 15.02.1987

Uyruğu : Kırgızistan

Yabancı Dili : Türkçe, İngilizce

E-posta : isakovanilu2011@gmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Lisans	Uygulamalı Matematik	Uluslararası Atatürk-Alatoo Universitesi, Bişkek, Kırgızistan	2009

YAYINLARI

1. "Gerçek Zamanlı Stereo Görme Sisteminin FPGA'de Tasarımı ve Uygulaması", International Symposium on INnovations in Intelligent SysTems and Applications, INISTA-2012.