

İŞLEMCİLERDE DÜŞÜK GÜÇ TÜKETİMİNE YÖNELİK YÖNTEMLER

Y. Sinan HANAY

YÜKSEK LİSANS TEZİ

Elektrik ve Elektronik Mühendisliği

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Haziran 2007

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Yücel ERCAN

Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Prof. Dr. Ünver KAYNAK

Anabilim Dalı Başkanı

Yusuf Sinan HANAY tarafından hazırlanan İŞLEMCİLERDE DÜŞÜK GÜÇ TÜKETİMİNE YÖNELİK YÖNTEMLER adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Doç. Dr. M. Önder EFE

Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Doç. Dr. Veysel GAZİ

Üye : Yrd. Doç. Dr. Oğuz ERGİN

Üye : Doç. Dr. M. Önder EFE

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Y. Sinan HANAY

Üniversitesi	:TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü	:Fen Bilimleri
Anabilim Dalı	:Elektrik ve Elektronik Mühendisliği
Tez Danışmanı	:Doç. Dr. M. Önder EFE
Tez Türü ve Tarihi	:Yüksek Lisans - Haziran 2007

Y. Sinan HANAY

İŞLEMCİLERDE DÜŞÜK GÜÇ TÜKETİMİNE YÖNELİK YÖNTEMLER

ÖZET

Bu tez mikroişlemcilerde düşük güç tüketimine yönelik bazı yöntemler önerip, onları ayrıntılı bir şekilde inceleyip ele almaktadır. Günümüzde daha yüksek işlem gücüne talep sürerken aynı zamanda daha düşük güç tüketimi de istenmektedir. Moore yasasına göre 18 ayda bir, birim tümleşik devreye sığdırılabilecek transistör sayısı 2 katına çıkmaktadır. Bu işlemcilerin üstel olarak hızlanmasına olanak verirken, diğer taraftan işlemcilerin daha yüksek frekanslarda çalışması tükettiği gücü de artırmaktadır. Dolayısıyla işlemcilerde güç tüketimine yönelik yeni yöntemlere ihtiyaç duyulmaktadır. Genel olarak iki teknik öne sürülecektir; mikroişlemcilerde dar değerlerin açığa çıkarılıp kullanılması ve hızlı karşılaştırıcı devre yapılarının kullanılması. Mikroişlemcilerde, dar değerlerin varlığından faydalanarak güç tüketiminin %36 oranında azaltıldığı gösterilecektir. Bu yöntem işlemcilerde üretilen ve tüketilen değerlerin yazmaç genişliğinden daha dar bir şekilde ifade edilebilmesine dayanır. Dar değerler sayesinde işlemci yazmaçlarının üst kısımları kapatılarak, bu kısımların tükettiği durağan güç azaltılmış olur. Çeşitli denek programlarla ideal yazmaç genişliği saptanıp bunun kullanılmasıyla elde edilecek güçten kazanım ortaya konacaktır. Buna ek olarak, dar değerlerden faydalanılarak yazmaçlarda hatalara karşı önerilen hızlı bir şekilde değer eşitliği üretme yönteminden bahsedilecektir. Son olarak da, güç kesintisi uygulayan uyku transistörlü devrelerin uyanma zamanını belirleme süresinin kısaltılması için devrelerin polinomlarla modellenmesi önerilecek, ve sonuçlar sunulacaktır.

Anahtar Kelimeler: Mikroişlemciler, Düşük Güç Tüketimi

University :TOBB Economics and Technology University
Institute :Institute of Natural and Applied Sciences
Science Programme :Electrical and Electronics Engineering
Supervisor :Associate Professor Onder Efe
Degree Awarded and Date :M.S. - June 2007

Y. Sinan HANAY

LOW POWER TECHNIQUES FOR SUPERSCALAR MICROPROCESSORS

ABSTRACT

This thesis proposes and analyzes some techniques aiming at reducing power dissipation of superscalar microprocessors. Nowadays, the drive toward faster microprocessors continues while requiring low power dissipation at the same time. The Moore's law states that the number of transistors that can be packed per integrated circuit doubles every 18 months. Although this leads to the speed up of processors exponentially, it also causes to the increase of power dissipation. Thus there is a need for new techniques to reduce power dissipation. Two techniques are presented in this thesis: exploiting narrow values and use of high performance comparators. With the use of narrow values a 36% reduction in power dissipation is achieved, and is presented in detail. This technique relies on the idea that some values inside the registers can be represented with fewer bits than the datapath width. With the help of narrow values most of the significant bits of registers are disconnected from power supply and thus power dissipation of these bits are reduced. We use spec2000 benchmark programs in order to identify optimum register width, and present power saving with using this approach. Additionally with narrow values we propose a fast parity generation in registers. Finally, we propose the modelling of circuits with polynomials in order to reduce the wake up time of circuits which uses sleep transistors.

Key Words: Microprocessors, Low Power Consumption

TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren hocam Yrd. Doç. Dr. Oęuz Ergin'e, Doç. Dr. Önder Efe'ye yine kıymetli tecrübelerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Elektrik Elektronik Mühendislięi Bölümü öğretim üyelerine teşekkürü bir borç bilirim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ	ix
ŞEKİLLERİN LİSTESİ	x
KISALTMALAR	xi
SEMBOL LİSTESİ	xii
1. GİRİŞ	1
1.1. CMOS Devrelerde Güç Tüketimi	2
1.1.1. Devingen Güç Tüketimi	2
1.1.2. Durağan Güç Tüketimi	4
1.1.3. CMOS Devrelerin Toplam Güç Tüketimi	4
1.2. Dar Değerlerin Açığa Çıkarılması	5
1.3. Güç Kesintisi Uygulayan Devrelerin Uyanma Gecikmelerinin Belir- lenme Zamanının Azaltılması	5
2. DAR DEĞERLERİN KULLANILMASIYLA GÜÇ TÜKETİMİNİN AZALTILMASI	7
2.1. Yazmaç Kütüğünün Yapısı	8
2.2. Dar Değerler	9
2.2.1. Dar Değerlerin Algılanması	10
2.3. Dar Değerlerin Açığa Çıkarılması	12

2.4. Benzetim Yöntemi	16
2.5. Tartışmalar	17
2.6. İlgili Çalışmalar	20
2.7. Sonuçlar	22
3. HIZLI HATA DÜZELTME KODLARI ÜRETİLMESİ	24
3.1. Hata Düzeltme Kodları ve Eşlik	25
3.2. Değişken Gecikmeli Eşlik Biti Üretim Devresi	26
3.3. Benzetim Yöntemi	28
3.4. Benzetim Sonuçları	28
3.5. Sonuçlar	31
4. DEVRELERİN UYANMA GECİKMELERİNİN BELİRLENME SÜRESİNİN KISALTILMASI	33
4.1. Davranış Modellemesi ile Benzetim Zamanının Azaltılması	34
4.2. Sonuçlar ve Açıklamalar	36
4.3. Sonuç	40
KAYNAKLAR	40
ÖZGEÇMİŞ	46

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 1.1. RISC Mimarisine Sahip İşlemciler	3
Çizelge 1.2. CISC Mimarisine Sahip İşlemciler	3
Çizelge 2.1. Dar Değerler	10
Çizelge 2.2. Kullanılan Mimari	16
Çizelge 3.1. Eşlik Üretimi	24
Çizelge 3.2. HDK İle Eşlik Üretimi	26
Çizelge 3.3. ÖZEL VEYA Eşlik Devresinin Özellikleri	29
Çizelge 3.4. Sıfır Algılayıcı 0.18 μ TSMC	30
Çizelge 4.1. Tablo ve Üretilen Polinomlar Arasındaki İlintiler	36
Çizelge 4.2. Tablo ve Üretilen Polinomların Karşılaştırılması	38
Çizelge 4.3. Sanal Toprağın Kararlı Hale Gelmesi İçin Gereken Zaman	39

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. İki kapılı SRAM bit Hücresi Yapısı	8
Şekil 2.2. Yazmaç Kütüğü Yapısı	9
Şekil 2.3. Sıfır Algılayıcı	10
Şekil 2.4. Farklı Sayıda Bitler İçin Sıfır Algılayıcısının Güç Tüketimi ve Gecik- meleri	11
Şekil 2.5. Dar Değer Algılama Devresi	12
Şekil 2.6. Dar Değer Algılama Yazmaç Kütüğü Yapısı	14
Şekil 2.7. Aktarım Kapıları Kullanılarak Oluşturulmuş 1- bit Çoğullayıcı	15
Şekil 2.8. Dar Değerlerin Çeşitli SPEC 2000 Denektaşlarında Yüzdesi	17
Şekil 2.9. SPEC2000 Denektaşları İçin Yazmaç Kütüğündeki Enerji Tasarrufu	19
Şekil 3.1. 8 bitlik Eşlik Üretme Devresi	25
Şekil 3.2. Önerilen Eşlik Üretme Devresi	27
Şekil 3.3. Spec2000 Denektaşlarında Dar Değerlerin Yüzdesi	29
Şekil 3.4. 8 bitlik Dar Değerler İçin Hızlı Eşlik Üretici	32
Şekil 4.1. Uyku Transistörlü CMOS Evirici	35
Şekil 4.2. Davranış Modeli	36
Şekil 4.3. Tek Evirici İçin İşlemci Yürütümü Zamanı	37
Şekil 4.4. 10 Evirici İçin İşlemci Yürütümü Zamanı	38
Şekil 4.5. Polinom Modellerde Doğruluk ve Hızlandırma	40

KISALTMALAR

Kısaltmalar	Açıklama
CISC	Complex Instruction Set Computer (Karmaşık Komut Kümeli Bilgisayar)
CMOS	Complementary Metal Oxide Semiconductor (Bütünleyici Metal-Oksit Yarıiletken)
DHA	Dal Hedefi Arabelleği
DYYZ	Düşükten Yüksek Yayılım Zamanı
HDK	Hata Düzeltme Kodları
NMOS	N-Channel MOSFET (N kanallı MOSFET)
PMOS	P-Channel MOSFET (P kanallı MOSFET)
RISC	Reduced Instruction Set Computer (İndirgenmiş Komut Kümeli Bilgisayar)
TSMC	Taiwan Semiconductor Manufacturing Company
YDYZ	Yüksekten Düşüğe Yayılım Zamanı
YSA	Yeniden Sıralama Arabelleği

SEMBOL LİSTESİ

Simgeler	Açıklama
Å	Angström
I	Akım (Amper)
E	Enerji (Joule)
f	Devrenin Çalışma Frekansı (Hertz)
P	Güç (Watt)
SiO_2	Silikon Di Oksit
V	Gerilim (Volt)

BÖLÜM 1

1. GİRİŞ

Güç tüketimi günümüzde tasarlanan sayısal sistemler için önemli bir kısıtlamadır. Yüksek güç tüketimi, yüksek soğutma masraflarına, güvenilirlik zaafına, yüksek enerji masraflarına ve taşınabilir cihazlar için düşük pil ömrüne neden olur. CMOS devrelerin güç tüketimi durağan ve devingen güç tüketimi olmak üzere iki ayrı türe ayrılır. Devrelerin anlamlı bir iş yapmadığı anlarda bile oluşan durağan güç tüketimi, kaynak gerilimi ile toprak arasına yerleştirilen transistör yolu sayısına bağlı olarak artar. CMOS devrelerin işlemesi sırasında ara düğümlerdeki sığaların dolup boşalması ve giriş çıkış işaretlerinin değişmesi sırasında anlık olarak NMOS ve PMOS türü transistörlerin aynı anda iletim durumunda olması nedeniyle devingen güç tüketimi oluşur. Dvingen güç tüketimi devrelerin işlem yaptığı sırada oluşması nedeniyle daha anlamlıdır ve günümüze kadar geçen sürede üretilmiş pek çok devrede baskın güç tüketimi türü olagelmiştir. Ancak günümüzde küçülen transistör boyutları, yükselen işletim sıcaklıkları ve artan transistör sayıları durağan güç tüketimini toplam güç tüketiminin ihmal edilemez bir parçası haline getirmiştir.

1965’de Moore’un öngördüğü gibi [1], birim yongaya sığdırılabilecek transistör sayısı zamanla üstel olarak artmaya devam etmektedir. Günden güne üretililecek transistörlerin kanal uzunlukları küçülmektedir. Bunun sonucu olarak da günden güne sayısal devrelerin çalışma frekansı ve güç tüketimleri artmaktadır. Son 10 yılda, mikroşlemcilerin çalışma frekansı her üç yılda bir iki katına çıkmıştır ve gözlemler bu eğilimin azalmayacağı yönündedir [2]. Güç tüketimi sadece taşınabilir cihazlar için önemli bir sorun gibi gözükse de, Intel genel müdürü Pat Gelsinger’in 2003 yılındaki öngörüsü [3] durumun vehametini ortaya koymaktadır. Gelsinger, devrelerin çalışma frekansı ve transistör sayısının aynı hızda artmaya devam etmesi durumunda, birkaç yıl içinde işlemcilerin tükettiği güçlerin kabul edilebilir sınırları aşacağını belirtmiştir. Şüphesiz bu açıklamadan günümüze geliştirilen birçok yöntem bugün işlemcilerimizin güç tüketimini makul seviyelerde tutmaktadır. Ancak işlemcilerin hızlanması sürekli devam etmektedir, ve daha az güç tüketmeye yönelik yöntemlere ihtiyaç duyulmaktadır.

Bu tezin amacı mikroişlemcilerde güç tüketimini düşürecek bazı yöntemler sunmak ve bunların sağladıkları kazançları irdelemektir. Öncelikle yazmaçlarda dar değerlerin ortaya çıkarılması ve bunlardan faydalanılarak, güç tüketiminin azaltılması hedeflenmektedir. Dar değerler sayesinde işlemci yazmaçlarının üst kısımları kapatılarak, bu kısımların tükettiği durağan güç azaltılmış olur. Çeşitli denek programlarla ideal yazmaç genişliği saptanıp bunun kullanılmasıyla elde edilecek güçten kazanım ortaya konacaktır. Buna ek olarak, dar değerlerden faydalanılarak yazmaçlarda hatalara karşı önerilen hızlı bir şekilde değer eşitliği üretme yönteminden bahsedilecektir. Son olarak da, güç kesintisi uygulayan uyku transistörlü devrelerin uyanma zamanını belirleme süresinin kısaltılması için devrelerin polinomlarla modellenmesi önerilecek, ve sonuçlar sunulacaktır.

1.1. CMOS Devrelerde Güç Tüketimi

CMOS düşük güç tüketimi, üretim maliyeti ve yüksek gürbüzlüğü nedeniyle son yıllarda sayısal devrelerde en çok tercih edilen teknoloji olmuştur. CMOS devrelerin güç tüketimi iki sınıfa ayrılabilir: durağan ve devingen. Devre işlem yaptığı zamanlar devingen güç tüketir. Devre işlem yapmadığı, durağan durumlarda, sızıntı akımlarından kaynaklanan durağan güç tüketir. Bir devrenin devingen güç tüketiminin durağan güç tüketimine göre çok fazla olmasına rağmen son yıllarda bu fark kapanmaya başlamıştır [4]. Üretilebilecek en küçük boyuttaki CMOS transistörlerin kanal uzunlukları her geçen gün kısaltmakta, çalışma frekansları artmakta ve çalışma gerilimleri düşmektedir. Çizelge 1.1.'de basit komut kümesine sahip, RISC, işlemciler Çizelge 1.2.'de ise karmaşık komut kümesine sahip, CISC, işlemcilerin yıllara göre güç tüketimi, çalışma frekansı ve üretilebilecek en küçük CMOS'un kanal uzunluğu, üretim teknolojisi verilmiştir. Bir devrenin devingen güç tüketimi $P \propto f V^2$ şeklinde ifade edilebilir. Her geçen gün devrelerin çalışma frekansı artmaktadır, bu da devingen güç tüketimini artırır, bu artışı dengelemek için her yeni teknolojiyle beraber besleme gerilimi V_{DD} de azaltılır.

1.1.1. Devingen Güç Tüketimi

CMOS devrelerde tüketilen devingen enerjinin iki kaynağı vardır. Birincisi devredeki sığaların (yük, parazitik vs) dolup boşalmasından, ikincisi ise NMOS ve PMOS

Çizelge 1.1. RISC Mimarisine Sahip İşlemciler

İşlemci	Transistör	Teknoloji(μm)	Frekans (MHz)	Güç (W)	Yıl
microSPARC I	8×10^5	1.3	40	2.5	1992
SuperSPARC II	$3,1 \times 10^6$	0.8	90	16	1994
UltraSPARC IIs	$5,4 \times 10^6$	0.35	250	25	1997
UltraSPARC T1	3×10^8	0.09	1000	90	2005

Çizelge 1.2. CISC Mimarisine Sahip İşlemciler

İşlemci	Transistör	Teknoloji (μm)	Frekans (MHz)	Güç (W)	Yıl
Intel 4004	$2,3 \times 10^3$	10	1	1	1971
Intel 486	$1,4 \times 10^6$	0.8	66	4.9	1994
Pentium III	$2,8 \times 10^7$	0.18	1000	29	1999
Pentium 4	$1,7 \times 10^8$	0.09	3600	103	2000

transistörlerinin aynı anda iletimde olduğu ve kısa devre durumu oluşturmasında kaynaklanır. Sığaların dolup boşalması için güç kaynağında harcanan enerji aşağıdaki gibidir:

$$E_{V_{DD}} = \int_0^{\infty} i_{V_{DD}}(t) V_{DD} dt = V_{DD} \int_0^{\infty} C_Y \frac{dV_c}{dt} dt = C_Y V_{DD} \int_0^{V_{DD}} dV_c = C_Y V_{DD}^2 \quad (1.1)$$

burada C_Y yük sığası, V_c ise çıkış gerilimidir. Aynı şekilde, E_Y ile gösterilen yük sığasını doldurmak için harcanan enerji hesaplanırsa,

$$E_Y = \int_0^{\infty} i_{V_{DD}}(t) V_c dt = \int_0^{\infty} C_Y \frac{dV_c}{dt} V_c dt = C_Y \int_0^{V_{DD}} V_c dV_c = \frac{C_Y V_{DD}^2}{2} \quad (1.2)$$

bulunur, buradan üretilen enerjinin yarısı transistör üzerinde ısıya dönüşür, ve yüksek enerji tüketimi aynı zamanda devrenin sıcaklığını da artırır. Sığaların yüklenmesine ek olarak, CMOS devrelerde aşağı ve yukarı çekme devreleri, PMOS ve NMOS ağları, devre girdisi sınırlı bir şekilde arttığı için, kısa bir süre için aynı anda açık duruma geçerler [5]. Bu durumda beslemeden toprağa bir kısa devre akımı akar. Bu kısa devre güç tüketimi girdinin değişim hızı ile ters orantılıdır.

1.1.2. Durađan Güç Tüketimi

İdeal olarak CMOS devrelerin durađan halde kapalıyken akım geçirmemesi beklenir. Ancak CMOS devrelerinde eşik-altı sızıntısı, kapı-oksit tünel sızıntısı ve savak-taban ters kutuplanması ile kaynak-taban kavşağında oluşan banttandır. Bu bileşenlerin arasında toplam durađan güç tüketimi içinde en önemli payı alan eşik-altı sızıntısıdır. Eşik-altı sızıntısından kaynaklanan güç tüketimi transistörün eşik gerilimine ve sıcaklığa üstel bağıntıyla bağlıdır [7]. Her yeni nesil üretim teknolojisi ile CMOS devrelerin besleme gerilimi düşmesiyle birlikte sürme gerilimi ($V_{GS} - V_T$) de azalır ve bu da iletkenliğini düşürür, transistörün gecikmesini kabul edilebilir düzeyde tutabilmek için eşik gerilimleri, V_T , de düşürülür. Bu durum küçülen transistör boyutlarıyla birlikte eşik-altı sızıntı akımlarında önemli bir yükselmeye neden olmuştur. Özellikle devre yüksek sıcaklıklarda iken eşik-altı güç tüketimi devrenin toplam güç tüketiminin yarısına kadar çıkabilir. Sızıntı akımının her yeni mikroişlemci neslinde 7,5 kat artması beklenmektedir [8].

Eşik altı akımı $I_{EA} \propto e^{(V_{GS}-V_T)}$ şeklinde ifade edilebilir [9], buradan V_T 'nin azaltılması, I_{EA} 'nin üstel olarak artmasına neden olur. Sızıntı akımı I_S 'nin sabit olduğu düşünülürse, böylece ortalama ve azami I_S birbirine eşit olur ve durađan güç tüketimi:

$$P_{duragan} = I_S V_{DD} \text{ olmuş olur.}$$

Kapı yalıtkanı SiO_2 iyi bir yalıtkan olmasına rağmen, yeni CMOS teknolojilerinde kalınlığının iyice azalması sonucu bazı elektronlar yalıtkan üzerinden geçebilmektedir, elektronların yalıtkan üzerinden geçebilme ihtimali SiO_2 'in kalınlığının azalması ile üstel artar. Bu tünel sızıntı akımları, 20 Å kalınlığında veya daha ince okside sahip 130 nm CMOS teknolojisinde dikkate almaya deđecek kadar büyümüştür [10].

1.1.3. CMOS Devrelerin Toplam Güç Tüketimi

CMOS devrelerin toplam güç tüketimi durađan ve devingen güç tüketimlerinin toplamıdır. Birkaç yıl öncesine kadar durađan güç tüketimi, devingen güç tüketimine kıyasla çok az olmasından dolayı göz ardı edilmiştir. Fakat yukarıda da bahsedildiđi

gibi sızıntı akımlarının artması durağan güç tüketimini artırmaktadır, ve durağan güç tüketiminin artış hızı devingen güç tüketiminin artış hızından fazladır. Bir işlemcinin tüketebileceği azami güç o işlemcinin sahip olduğu soğutma biriminin vasıflarıyla sınırlıdır ve makul düzeylerde bir soğutma düzeneği ile bu rakam iş bilgisayarları ve sunucuları için 150 Watt civarındadır [10].

1.2. Dar Değerlerin Açığa Çıkarılması

İşlemcilerde üretilen ve tüketilen bir çok yazmaç değeri veri yolunun genişliğini tamamen kullanmaz. Bu tür değerler yazmaçların yüksek değerli bitlerini gereksiz yere kullanırlar, ve gereksiz yere güç tüketirler. Bu tür değerlere dar değerler denir. Kısa gösterimde işaret biti dar değer önüne getirilir. 64 bitlik veri yoluna sahip mimarilerde, dar değerler de 64 bitle gösterilmektedir; ancak daha az sayıda bit kullanarak bu değerleri saklamak güç tüketimini azaltacaktır. Dar değerleri anlayabilmek için, bir değer üst bitlerinin ardışık birer veya sıfırlar içerip içermediği bilinmelidir. Ardışık birleri ve sıfırları algılayan iki devre sunup bunların güç tüketimi ve hızlarını inceleyeceğiz.

1.3. Güç Kesintisi Uygulayan Devrelerin Uyanma Gecikmelerinin Belirlenme Zamanının Azaltılması

Sızıntı akımının yarattığı güç tüketimi yüksek ölçekli devrelerin üretilmesi için kullanılan küçük boyutlu teknolojilerde yonganın tükettiği toplam gücün büyük bir kısmını oluşturur. Yonganın işe yarar bir iş yapmadığı halde ortaya çıkan bu durağan durum güç tüketimi yüksek sıcaklıklarda tüketilen toplam gücün yarısına kadar çıkabilir. Durağan durumdaki güç tüketimi devrelerin etkin olmadıkları anlarda kaynak geriliminin kesilmesi yolu ile 100 kata kadar düşürülebilir. Kaynak geriliminden uyku transistörü adı verilen transistörlerle ayrılmış bu devrelere yeniden gereksinim duyulduğunda kesilen güç yeniden devreye sokulmalıdır.

Devre tasarımcıları için uyku transistörleri açıldıktan sonra, devrenin tamamen işler duruma geçmesi için gereken zamanı bilmek önemlidir. Şu anki tasarım araçları kullanılarak devrelerin uyku transistörlerinin üzerindeki düğümlerin kararlı duruma geçme zamanının hesaplanması benzetimlerde uzun zaman almaktadır. Bu bölümde

devrelerin uyku transistörleri olmadan gösterdikleri davranışı modelleyerek kararlı duruma ulaşma zamanının çözümlenmesi işlemini hızlandıracak bir yöntem önerilmektedir. Önerilen yöntem kullanılarak yapılan benzetimler olağan SPICE benzetimlerine göre 2 kat daha az zamanda sonuçlanmıştır.

BÖLÜM 2

2. DAR DEĞERLERİN KULLANILMASIYLA GÜÇ TÜKETİMİNİN AZALTILMASI

Çokyollu çağdaş mikroişlemcilerde yazmaçlar güç tüketiminde büyük paya sahiptir. Komut penceresi her yeni nesil işlemci ile büyüdüğünden, yazmaçlar da büyük ve daha çok güç tüketmeye başlar. Dolayısıyla yazmaçlarda düşük güç tüketimine yönelik yöntemlere ihtiyaç duyulmaktadır. Veri saklama birimleri yazmaçlara yazılan ve yazmaçlardan okunan birçok değer mevcut veri yazma alanının daha küçük bir kısmında saklanabilir. Bu tür verilerin üst bitlerini saklamaya gerek yoktur.

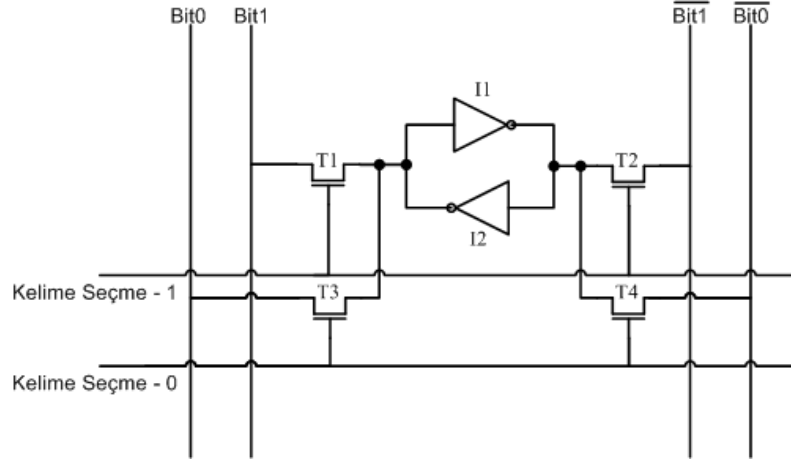
Çokyollu işlemciler başarımlarını artırmak için sırasız işleme, çok izleme gibi yöntemler kullanırlar. İşlemcinin komut sayısı günden güne arttığından, yazmaçların yeniden adlandırılması yöntemi daha çok yazmaca ihtiyaç duyar. Daha büyük yazmaçlar daha çok güç tüketir, ve güç tüketimlerinin azaltılması günümüz mikroişlemcilerinde önemli bir konudur.

Çokyollu işlemciler, komut düzeyinde koşutluğu artırmak ve yanlış veri bağımlılıklarını ortadan kaldırmak için yazmaçların yeniden adlandırılması yöntemini kullanırlar. Yazmaçların yeniden adlandırılmasında, sonuç üreten her komut için yeni bir fiziksel yazmaç atanır. Fiziksel yazmaç atandıktan sonra, komut tarafından hedef olarak kullanılan mimari yazmacı bu fiziksel yazmaca atanır ve bu mimari yazmacı tekrar başka bir komut tarafından değiştirilene kadar, geçerli fiziksel yazmaç - mimari yazmaç eşlemesi tutulur.

Önerilecek yöntem, veri tutan bileşenlerin, yazmaçların, basit bir şekilde bölümlenmesine dayanır. Bu bölümlenme işlemi, örnek denektaşlarının yazmaçlarda saklanan ortalama veri uzunlukları gözlemlerine göre yapılmıştır. Eğer veri alanında saklanacak değer gerekenden daha az bit ile yazılabilir durumdaysa, değerün üst bitleri güç tüketimini azaltmak için yazılmayacaktır.

2.1. Yazmaç Kütüğünün Yapısı

Yazmaçlar verileri tutmak için Şekil 2.1. de gösterilen SRAM bit hücrelerini kullanmaktadır.

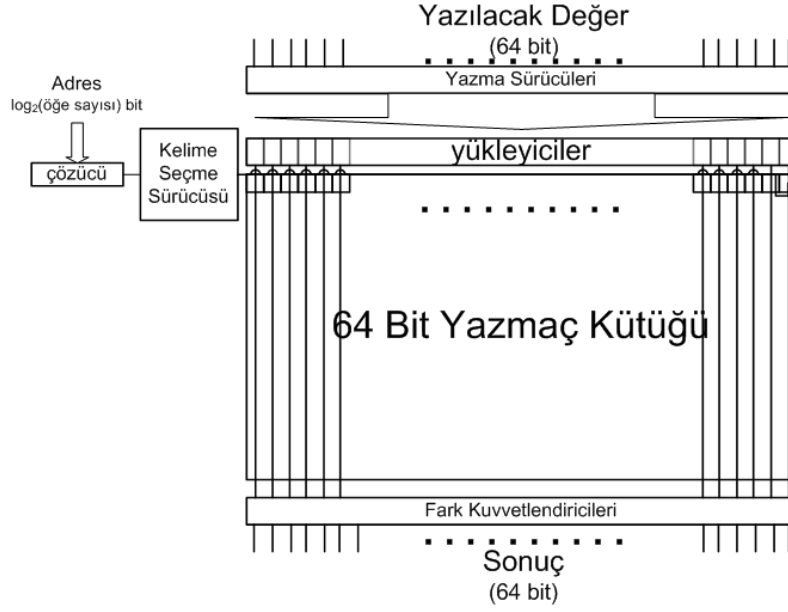


Şekil 2.1. İki kapılı SRAM bit Hücresi Yapısı

32 bitlik bir işlemcide eğer 128 yazmaç var ise, yazmaç kütüğü bu SRAM hücrelerinin 32×128 lik dizisinden oluşur.

P4 mimarisine sahip bir çevrimde 4 komut yayımlayan bir işlemcinin yazmaç kütüğü 4'ü yazma, 8'i okuma olmak üzere 12 kapılık SRAM hücrelerinden oluşur. Yüksek sayıda kapının olması güç tüketimini de artırır.

Yazmaç kütükleri, SRAM bit hücrelerine ek olarak önceden yükleme birimleri de içerir. Bu önceden yükleme birimleri, yazmaçlar erişilmeden önce bit hatlarını V_{DD} 'ye çekmek için kullanılır. Yazmaç erişimi, yazmaç belirtecinin çözücüler tarafından çözülmesiyle başlar. Çözücü bir uyum sinyali ürettiğinde, kelime seçme sürücüsü gerekli öğeyi seçer. Bit hücresinin bit ve ~bit hatlarında yeteri kadar gerilim farkı yaratmasından sonra, fark kuvvetlendiricileri hızlı bir şekilde değeri algılar ve mandal devresine aktarır. Yazma işlemi için, önceden yükleyiciler ve fark kuvvetlendiricileri yerine, yazma sürücülerini kullanarak bit ve ~bit hatlarının güçlü işaretlerle yüklenmesi sağlanır ve hızlı bir şekilde bit hücresinin içeriğinin yeni girdilere değişmesi sağlanır. Şekil 2.2. tarif edilen yazmaç kütüğü yapısını göstermektedir.



Şekil 2.2. Yazmaç Kütüğü Yapısı

Bir yazmaç kütüğünde her devre bileşenine erişim enerji tüketir. Ancak, 128 NMOS transistörün kapı sığasını süren kelime seçme sürücülerinin, öge sayısı kadar transistörün difüzyon sığasını süren yazma sürücülerinin ve önceden yükleyicilerin güç tüketimi yazmaç kütüğünün güç tüketiminde ana paya sahiptir.

Yazmaç kütüğünün de güç tüketimi devingen ve durağan olarak sınıflandırılabilir, ve yazmaç kütüğünün boyutu arttıkça güç tüketiminin her iki bileşeni de artar. Yazmaç kütüğünün durağan ve devingen güç tüketimi dar değerlerden faydalanılarak azaltılabilir. Bu bölümde dar değerlerin açığa çıkarılıp, kullanılmasından faydalanılarak devingen güç tüketiminin azaltılması konusuna değinilecektir.

2.2. Dar Değerler

İşlecimin içerisinde üretilen birçok değer veri yolunun genişliğinden daha az bitle ifade edilebilir. Çizelge 2.1. bazı 32-bit dar değer örnekleri göstermekte ve onların daha kısa bir şekilde gösterimini sunmaktadır.

64 bitlik veri yoluna sahip mimarilerde, dar değerler daha az bitle ifade edilebilecekken

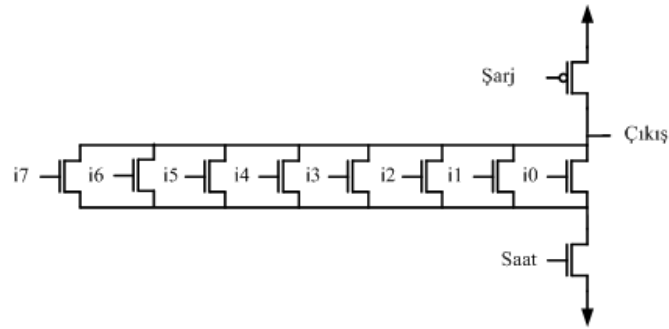
Çizelge 2.1. Dar Değerler

Dar Değer	Kısa Gösterimi
11111111111111111111111111111111	1 (1 bit)
00000000000000000000000000000001	01 (2 bit)
11111111111111111111111111110001010	10001010 (8 bit)
000000000000000000000000000010001001	010001001 (9 bit)

tüm 64 bitlik alanı kullanıp gereksiz yer israfına yol açarlar.

2.2.1. Dar Değerlerin Algılanması

Dar değerler işlemcinin içerisinde kolayca algılanabilirler. Dar değerleri anlayabilmek için, baş bitlerin ardışık 0 veya 1 olduğunu anlayan devreler kullanılabilir. Bu devreler kolayca durağan VE DEĞİL kapıları kullanılarak yapılabilir, ancak daha hızlı olması nedeniyle devingen mantık kapıları kullanılacaktır.



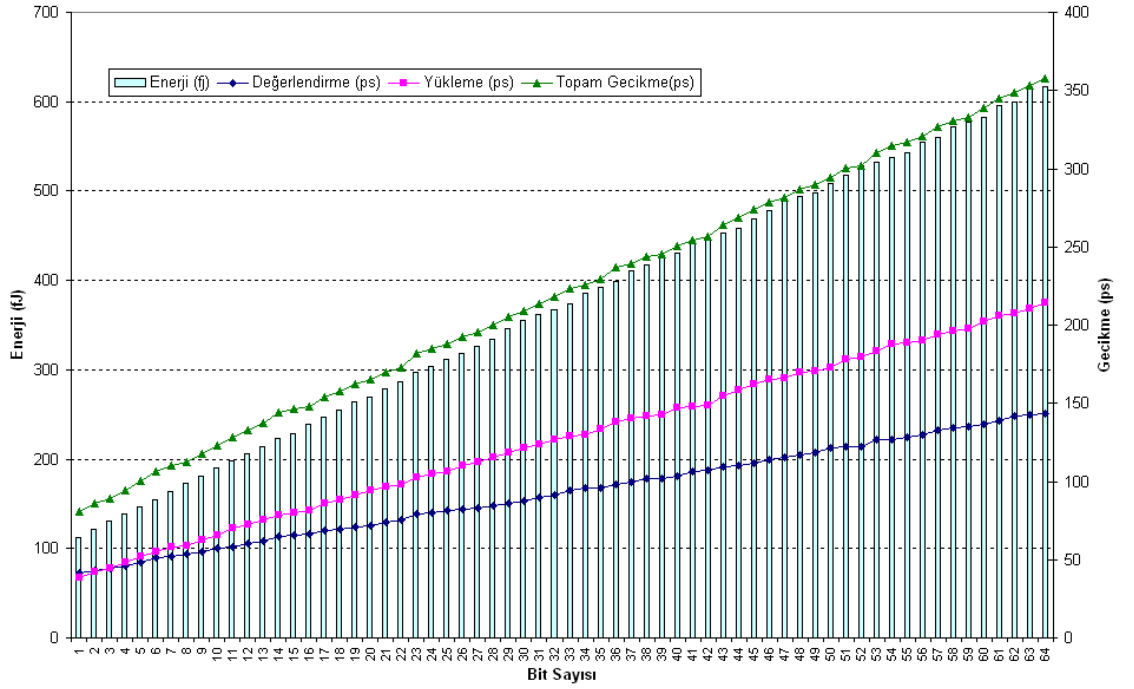
Şekil 2.3. Sıfır Algılayıcı

Şekil 2.3. devingen mantık ile oluşturulmuş sıfır ve bir algılama devrelerini göstermektedir. Bu devreler esasen devingen VEYA DEĞİL ve VE DEĞİL kapılarıdır. Devrenin çalışması önceden yükleme evresi ile başlar, eğer girişlerden birisi yüksek değere çıkarsa çıkış saat işaretinin yüksek değeriyle boşalır ve devre girişlerde ardışık sıfır olmadığını belirtmiş olur. Eğer tüm girişler sıfır ise, çıkış yükü boşaltılmaz ve bu devrenin okuduğu değer ardışık sıfırlardan oluştuğunu belirtir.

Burada dikkat edilmesi gereken nokta iki devrede de aşağı çekme devresi olarak NMOS'ların kullanılmasıdır. Bunun nedeni NMOS'ları 0'ı aktarmada PMOS'lardan daha iyi olmasıdır. Dolayısıyla, ardışık 1 devresi için, PMOS'ların kullanılması ye-

rine, girişleri evirilir. Ardışık 0 ve 1 algılama devreleri kullanıldığında, dar değer in boyu küçüldükçe devrenin boyu büyür. Bunun nedeni daha dar bir değer in daha çok gereksiz yüksek değerli bitleri olmasıdır.

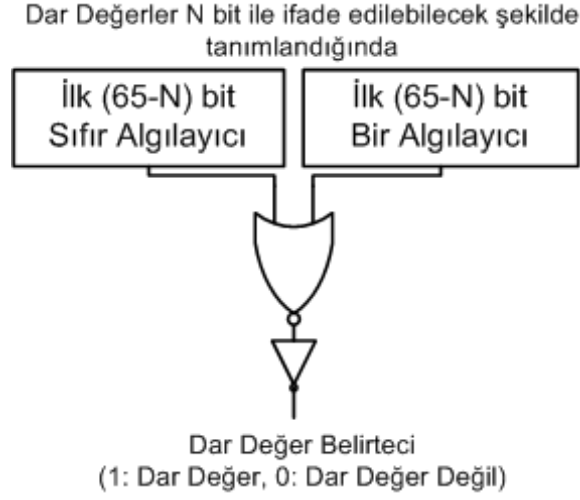
Şekil 2.4. ardışık 0 algılayıcısının enerji ve hız özelliklerini göstermektedir. Al- gılayıcı devreler, en yüksek hız için tasarlanmış ve algılayıcı devrelerin gecikmeleri çıkış düğümünün tek bir transistörden boşaldığı en kötü durum için ölçülmüştür. Hem önyükleme hem de değerlendirme gecikmeleri artan giriş sayısı ile artar. Bunun nedeni, çıkış düğümünün yüklenmesi ve boşaltılması gereken toplam sığasının art- masıdır. Şekilde 2 tane düşey eksen bulunmaktadır; soldaki eksen enerji tüketimi ver-



Şekil 2.4. Farklı Sayıda Bitler İçin Sıfır Algılayıcısının Güç Tüketimi ve Gecikmeleri

lerini gösterirken, sağdaki eksen ise gecikme değerlerini göstermektedir. Şekilden de görüleceği gibi artan girdi sayısı ile sistemin gecikmesi ve enerji tüketimi artmaktadır.

Bir algılama devresi için de gecikme ve enerji değerleri yaklaşık aynıdır, yalnız girdilere eklenen eviriciler gecikme ve güç tüketimini biraz daha artırmaktadır. Bir algılayıcı devrede eviricilerin kullanılması, önyükleme gecikmesini etkilemezken, değerlendirme zamanını 35ps civarında ve enerji tüketimini 60 fJ civarında artırmak-



Şekil 2.5. Dar Değer Algılama Devresi

tadır.

Bu iki algılama devresi Şekil 2.5.'te gösterildiği gibi birbirine paralel olarak kullanılmalıdır. Dar değer N bit olarak tanımlandığı bir veri yolunda, değer üst 65-N bitlik kısmına 0 ve 1 algılayıcı devreleri bağladıktan sonra, bu iki algılayıcının çıkışı bir VEYA kapısı kullanılarak değer dar olup olmadığı anlaşılır. Eğer bu gelen işaret 1 ise değer dardır, ama 0 ise gelen değer N bittenden daha geniştir.

Bu çalışmada önerilen yöntem tasarım zamanı açısından ele alınmıştır. Dolayısıyla [27]'de önerilen ve değer genişliğini tam olarak belirleyen karmaşık 0 ve 1 algılayıcıları kullanmak gereksizdir. Burada önerilen yöntem denektaşlarının ortak davranışları göz önüne alınarak oluşturulmuştur. Sistem sadece üretilen değer, ön tanımlı dar değer genişliğinin altında veya üstünde olup olmadığına bakmaktadır.

2.3. Dar Değerlerin Açığa Çıkarılması

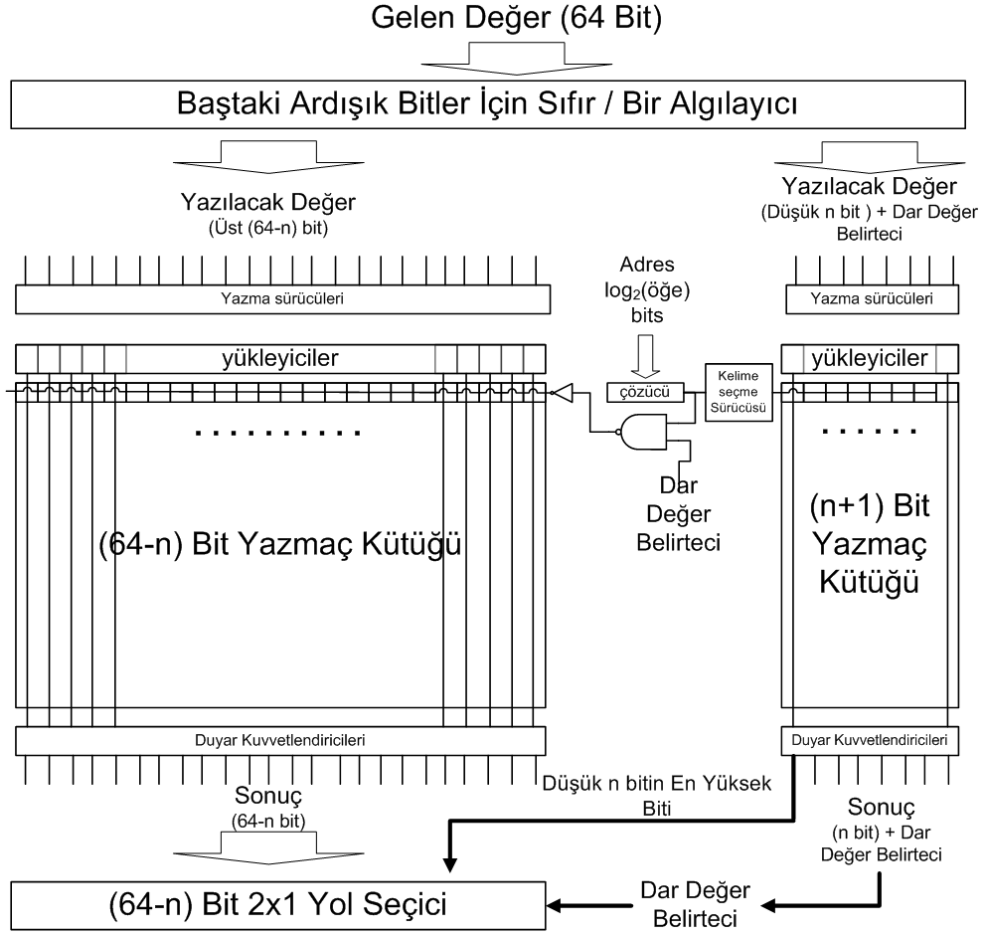
Dar değerler kullanılarak yazmaç kütüğünün enerji tüketimi azaltılabilir. Bu tür dar değerler, çok fazla tekrarlayan yüksek değerli işaret bitleri tuttuklarından, yazmaç kütüğüne yazılıp okunurken verimsizliğe neden olur. Dar değerleri belirleyerek ve yazmaç kütüklerinin üst kısmına yazmayarak enerji tüketiminin önemli ölçüde azaltılması mümkündür.

Bu kısımda, yazmaç kütüğünün iki parçaya bölünmesini ve sadece dar değerlerin önemli kısmının küçük bölüte yazılmasını öneriyoruz. Bu amaca yönelik, yazmaç kütüğündeki her öge için fazladan bir darlık belirteç biti ekleyeceğiz. Bu darlık belirteç biti, tutulan değer dar olup olmadığını ifade edecektir. Eğer yazılması gereken değer dar olduğu anlaşılırsa, darlık belirteç biti bu öge için geçerli kılınır ve değer üst kısımları büyük bölüte yazılmayarak güç tasarrufu sağlanır. Bir okuma işleminde ise, darlık belirteç biti denetlenir ve eğer okunacak değer dar ise, değer üst kısmı okunmaz. Bunun yerine bir Çoğullayıcı aracılığıyla dar değer önemli kısmının işaret biti uzatılır. Eğer değer dar olmadığı anlaşılırsa, dar değer belirteç biti sıfırlanır ve herhangi bir okuma durumunda, saklanan değer yüksek değerli bitleri de okunur.

Şekil 2.6. önerilen yazmaç kütüğü yapısını göstermektedir. Yazmaç kütüğü, alt ve üst bitleri tutan iki ayrı bölüte bölünmüştür. Bu çalışmada günümüzdeki çoğu işlemcinin veri yolunun 64 bit genişliğinde olması nedeniyle, veri yolunun genişliği 64-bit olarak ele alınmıştır. Eğer dar değer n bit olarak tanımlanırsa, alt bölüt $n+1$ bitlik darlık belirteç biti ile n bitlik dar değeri, üst kısım ise $64-n$ bitlik üst kısmı tutar. Değerin n bitlik alt kısmını tutan bölüt, yazma ve okumada her zaman kullanılır. $64-n$ bitli tutan üst bölüt ise yalnızca gelen değer dar olmadığı zaman kullanılır. Böylece dar değerlerin üst kısımlarını okumama ve üst kısımlarına yazmama yoluyla, güç tasarrufu sağlanmış olur. Yüksek güç tasarrufu ve dar değerleri algılayan devrenin güç tüketimini dengelemek için, dar değerlerin yüksek oranda olması gerekmektedir.

Şekil 2.6.'da gösterilen devre iki kısma bölünmesine rağmen, her yazmaç ögesi için sadece bir çözücü vardır. Çözücü devresi eşleme olduğunu belirttiğinde, sadece yazmacın üst kısmı gelen değer dar değerse kullanılır. Bu darlık belirteç işareti, erişim türüne bağlı olarak iki kaynaktan üretilebilir. Eğer erişim bir yazma erişimi ise, darlık belirteç işareti ardışık 0 ve 1 algılama devrelerinden gelir. Eğer erişim bir okuma işlemi ise, dar değer işareti okunacak ögenin darlık belirteç bitinden gelir. VE DEĞİL kapısının darlık belirteç girişinin çoğullamaya ihtiyacı yoktur, çünkü genelde yazmaç kütüğünün okuma ve yazma bağlantı noktaları ayrıktır. Okuma işlemi için adanmış kelime seçme hatları için, VE DEĞİL kapısının girişi saklanmış bittin ve yazma için ayrılmış kelime seçme hatları için darlık belirteç işareti dar değer algılayıcısı tarafından sağlanır.

Herhangi bir yazma işleminde enerji tasarrufu, saklanmış değer üst kısımlarını

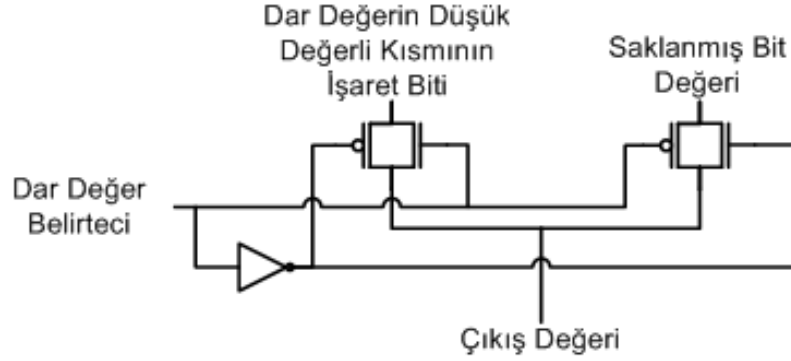


Şekil 2.6. Dar Değer Algılamalı Yazmaç Kütüğü Yapısı

sürmesi gereken kelime sürücü devrelerinin kapatılmasıyla, ve yüksek değerli bitleri tutan bölütün yerel kelime seçme hattının kapatılmasıyla sağlanır. Yazmaç kütüğünde bir değer okunacağı zaman, enerji tasarrufu; yüksek bölütü tutan bölütün yerel kelime seçme hattının açılmamasıyla ve bu bit hatlarına karşılık gelen fark kuvvetlendiricilerinin kapatılmasıyla sağlanır.

Bir dar değere erişim yazmaç kütüğü üzerinde bazı değişiklikleri zorunlu kılar. Saklanmış değerın üst değerli kısmı ve düşük değerli kısmın işareti biti arasında seçim yapmak için bir adet (64-n) bitlik 2x1'lik Çoğullayıcı gerekir. Eğer saklanmış değer dar ise, değer düşük değerli kısmın en yüksek bit olan işaret bitinin uzatılmasıyla tekrar oluşturulur.

Şekil 2.7. dar değerleri yeniden üretmek için ya da geniş bir değerın yüksek değerli



Şekil 2.7. Aktarım Kapıları Kullanılarak Oluşturulmuş 1- bit Çoğullayıcı

bitlerini okumak için kullanılan 1-bit çoğullayıcı devresini göstermektedir. Bu 1-bit çoğullayıcı devrelerinden (64-n) tane kullanılarak 64 bitlik son değer oluşturulması sağlanır. CMOS aktarım kapılarının kullanılma nedeni çoğullama işleminin hızlı olmasıdır.

Dar değer genişliğini tanımlarken önümüzde bazı açık ödünleşimler vardır. Enerji tüketiminde azami tasarruf için değer genişlikleri daha dar olmalıdır, böylece değer büyük parçalarında dar değerler ile okuma ve yazma işlemlerinden sakınılabilir. Diğer taraftan, çok az bit sayısına sahip bir dar değer genişliği tanımlamak çalışan programda düşük oranda dar değer elde edilmesine neden olacak ve daha az enerji tasarrufu sağlayacaktır. Eğer çoğu değer geniş olarak belirlenirse, yazmaç kütüğünün bölümlenmesi enerji tasarrufu sağlamayacak, hatta enerji tüketiminin artmasına bile neden olabilecek düzeyde zararlı olacaktır. Dolayısıyla, bir program içerisinde belirlenen dar değerlerinin yüzdesinin yüksek olması ve dar kısmın tuttuğu değerlerin çok geniş olmaması arasında bir noktada en iyi bir değer seçilebilir.

Diğer taraftan dar değer genişliğinin devre düzeyinde de bazı ödünleşmeleri vardır. 0 ve 1 algılama devrelerinin gecikme değerleri Şekil 2.4.'te de görüldüğü gibi dar değer genişliğine bağlıdır. Dar değer genişliği azaldıkça, ardışık 0 ve 1 algılama devrelerinin giriş sayıları artar. Dolayısıyla dar değerlerin genişliği darlaştıkça bir ve sıfır algılayıcı devrelerin gecikme ve güç tüketimleri de artar. Bu devre düzeyindeki özelliği ise, çok dar tanımlanan bir dar değer olası zararlı sonuçlarındandır.

Çizelge 2.2. Kullanılan Mimari

Parametre	Düzenleşim
makine genişliği pencere genişliği	4 - alma , 4 - yayımlama, 4 - onaylama 64 öğeli yayım kuyruğu, 64 öğeli yükleme-saklama kuyruğu, 128 öğeli YSA
işlev birimleri ve gecikmeleri (toplam/yayım) L1 Komut önbelleği	4 Int Add (1/1), 1 Int Mult (3/1) / Div (20/19), 2 Load/Store (2/1), 2 FP Add (2), 1FP Mult (4/1) /Div (12/12) / Sqrt (24/24) 32 KB, 2 yollu küme ilişkilendirmeli, 64 bayt hattı, 1 çevrim vuruş zamanı
L1 Veri önbelleği	64 KB, 4 yollu küme ilişkilendirmeli, 64 bayt hattı, 2 çevrim vuruş zamanı
L2 önbelleği	2 MB , 8 yollu küme ilişkilendirmeli, 128 bayt hattı, 6 çevrim vuruş zamanı
DHA Dal Öngörücüsü	4K Öğe- 2 yollu küme ilişkilendirmeli 1K öğeli, 8 bit evrensel tarih, 4K öge ikili modlu, 1K öge seçici
Hafıza	256 bit genişlik, 300 çevrim ilk kısım, 1 çevrim ara

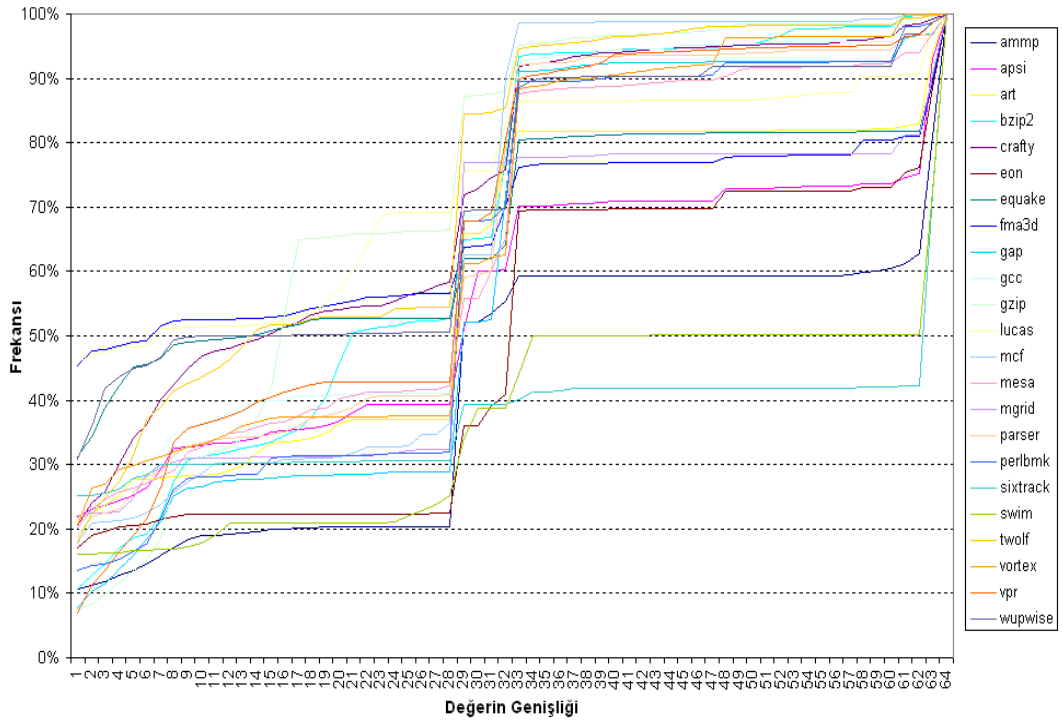
2.4. Benzetim Yöntemi

Günümüz x86 tabanlı işlemcilerinin içindeki dar değerlerin yüzdesini tam olarak anlayabilmek için 64-bit komutları da işleme yeteneğine sahip PTLsim işlemci benzeştiricisini [28] kullandık. SPEC 2000 denektaşlarını benzeştiricide çalıştırarak değişik programlar, değişik çalışma yükleri için işlemcinin içindeki dar değerlerin yüzdesini elde ettik. Kullanılan tüm SPEC 2000 denektaşları x86-64 tabanlı komut kümelerini hedef alarak en iyi düzeyde başarımlar için gcc 3.4.3 kullanılarak derlenmiştir. Denektaşlarının benzetiminde ilk bir milyar komutun sonuçları atlanmıştır, ondan sonraki bir milyar komutun sonuçları kullanılmıştır. Çizelge 2.2. benzetimde kullanılan mimariyi göstermektedir.

Yazmaç kütüğü içerisindeki enerji tüketimini hesaplamak için; benzeştiriciden olay sayıları, yazmaç kütüğünün enerji tüketimleri değerleriyle birleştirilerek kullanılmıştır. SPICE benzetimleri için BSIM 4.5.0 modeli kullanılmıştır [29].

2.5. Tartışmalar

Yazmaç kütüğünü işlenen genişliğine göre bölütlerken işlemci içerisindeki dar değerlerin yüzdesi enerji tasarrufu açısından önemlidir. Eğer dar değerlerin yüzdesi çok düşük ise, dar değerleri algılamak ve kullanmak için fazladan konulan devrenin enerji tüketimi dolayısıyla yazmaç kütüğünün enerji tüketimi yükselebilir. Fakat, birçok araştırmacı tarafından ortak işlenen çalışma yüklerinde işlenen değerlerin genişliği genellikle dar olarak gözlemlenmiş ve belirtilmiştir [30, 31].



Şekil 2.8. Dar Değerlerin Çeşitli SPEC 2000 Denektaşlarında Yüzdesi

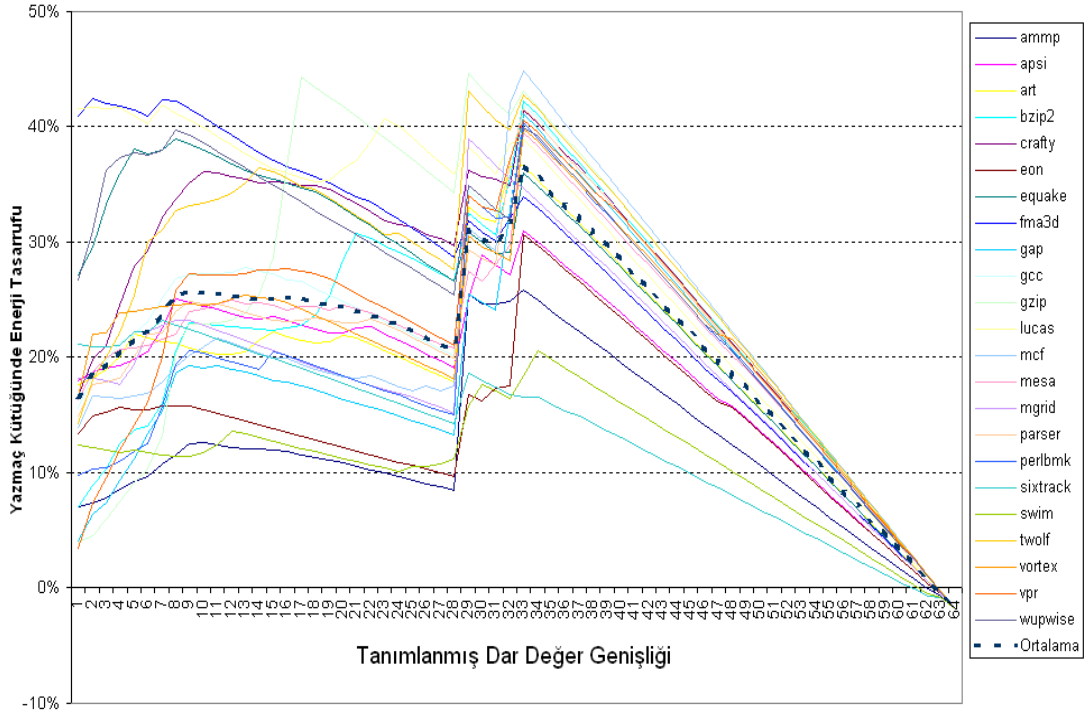
Şekil 2.8. her spec2000 denektaşını programı içerisindeki üretilmiş işlenenlerin genişliğinin frekansını göstermektedir. Her bir çizgi farklı bir denektaşını göstermektedir olup, her bir değer genişliği için dikey ekseninde bu değerden küçük veya bu değere eşit üretilen değerlerin yüzdesini göstermektedir. Şekilden de görülebileceği gibi, dar değerlerin yüzdesi bu çalışma yüklerinde oldukça fazladır. Çoğu denektaşını için, dar değeri 34 bit seçmek üretilmiş değerlerin çoğunu içermektedir. Önerdiğimiz işlenen değerlerin genişliğinin farkında olan yazmaç kütüğü yapısının enerji tüketimini hesaplamak için SPICE benzetimlerinden elde edilen devrelerin enerji tüketimi değerlerini, işlemci benzeştiricisinden aldığımız olayları da göz önüne alarak kullanırız.

Elde edilen enerji tasarruflarını hesaplarken yazmaç kütüğünün ve eklediğimiz ek devrenin enerji tüketimleri kullanılmıştır.

Şekil 2.9. bölümlenmiş yazmaç kütüğünün toplam enerji tüketimini göstermektedir. Her bir çizgi farklı bir denektaşının enerji tasarrufunu göstermektedir, ve yatay eksen saklanmış değer düşük kısmını tutan bölümünün genişliğini, yani dar değer genişliğini belirtmektedir. Görüldüğü gibi her denek taşının farklı genişlikte dar değerleri vardır ve değişik dar değer genişlikleri için değişik enerji tasarrufları elde edilmiştir. Fma3d gibi çok fazla dar değere sahip olan denek taşları dar değerlerin dar tanımlanarak en fazla enerjiden tasarrufu sağlamıştır. Öte yandan, orta boyutlu değerlerin çokça kullanıldığı diğer bazı denek taşları için dar değeri daha geniş seçmek enerji tüketimini düşürmek açısından daha verimlidir. Önerilen yazmaç kütüğünü bölümleme yöntemi ile en yüksek enerji tasarrufları tüm spec2000 denektaşları göz önüne alındığında ortalama olarak 33 bit çıkmıştır. Dar değer genişliğini 33 bit olarak tanımlamak, yazmaç kütüğünün enerji tüketiminde %36.6 lık bir düşüş sağlamıştır.

Her denektaşı için enerji tasarrufları farklılıklar göstermektedir. Şekil 2.9.'da çok sayıda tepeler ve sıçramalar görülmektedir. Bunun nedeni program içerisindeki dar değerlerin sıklığının, artan dar değer genişliğiyle birikimli olarak artmasıdır. Örneğin, eğer 33 bit değerlerin sıklığı fazla ise, dar değerlerin genişliğini 32 bit tanımlamak, işlemci içerisindeki 33 bitlik değerlerin 1 bit farkla kullanma şansını kaybettirir. Eğriler şeklin sağ tarafında ve sol tarafının ortasında doğrusal olarak azalmaktadır. Bunun nedenini anlamak Şekil 2.8.'i incelemek ile mümkündür. Şekil 2.8.'den de görülebileceği gibi bazı değer genişliklerinde, dar değer genişliğinin tanımını değiştirmek dar değerlerin frekansında bir değişikliğe yol açmamıştır. Çoğu denektaşı program için, genişlikleri 34 ile 62 arasında tanımlanabilecek çok fazla dar değer yoktur, ve enerji tasarrufunun azami seviyede tutulması için dar değer genişliği 33 bit olarak tanımlanmalıdır.

Şekil 2.9.'da gösterildiği gibi her denektaşının farklı enerji tasarrufları vardır. Bazı denektaşları birkaç noktada tepe noktası vardır ve azami enerji indirimi bu noktalarda en azdır. Örneğin, gzip denektaşının şekilde iki tepe değeri vardır. Dar değer genişliği 17 bit olarak tanımlandığında gzip denektaşı için yazmaç kütüğünün enerji tüketimi %44.3 oranında azalmıştır. Gzip için aynı oranda bir enerji tasarrufu gösteren diğer dar değer genişliği ise 29 bittir. Böyle durumlarda dar değer genişliğini



Şekil 2.9. SPEC2000 Denektaşları İçin Yazmaç Kütüğündeki Enerji Tasarrufu

daha büyük tutmak daha faydalıdır; böylece dar değerleri algılama devreleri daha hızlı çalışabilir.

Spec2000 denek taşı programları sonucu, ortaya dar değerlerin 33 bit tanımlanması durumunda azami enerji tasarrufunun sağlanabileceği sonucu çıkmıştır. Bunun nedeni, büyük değerlerin genellikle 29 veya 33 bit ile temsil edilebilmesidir. Şekildeki sıçramaların nedeni işlemciler içerisindeki yığıt ve çerçeve göstericileridir. Her denektaşının kendine has bir davranışı ve kendi en iyi enerji tasarrufunu sağlayan dar değer genişliği farklıdır. Tasarımcıların dar değer genişliğini belirlerken işlemcinin tipik özelliklerini göz önünde bulundurması ve çalışma yükleri üzerinde çözümler yapması gerekmektedir.

Dar değer genişliği 62 bitten daha yüksek bir değer seçildiğinde, her denektaşısı için enerji tasarrufu negatif değerlere gitmektedir. Bunun nedeni dar değerleri algılamaya yarayan ve yazmaç kütüğünü bölümleyen ek devrenin tükettiği enerjidir. 64 bitlik dar değer esasen hiç sıkıştırılmamış geniş bir değerden fazladan bir darlık biti eklenerek

yazılması anlamına gelmektedir.

2.6. İlgili Çalışmalar

Küçük değerlerle çalışmanın başarımlarını faydaları Intel'in Pentium 4 işlemcisinde çift pompalı işlev birimlerinde gerçekleştirilmiştir [32]. Bu boru hattına sahip işlev birimleri her çevrimde değerlerin 16 bitini işler ve her işlem 3 çevrim sürer. 32 bitlik veri yolunun işlev birimleri ilk çevrimde düşük değerli 16 bit üzerinde, ikinci çevrimde ise diğer kısmında çalışır. Üçüncü çevrim ise bir işlemde taşma, elde oluşması durumlarının üstesinden gelebilmek için ayrılmıştır. Bu yolla işlev birimlerinin çalışma frekanslarını artırmak mümkündür. Sistem düzeyinde enerji tasarrufu için benzer bir yöntem önerilmiştir [33].

Çoğu araştırmacı dar değerlerin varlığını bazı yollarla gözlemlemişlerdir, 0 baytlarının yazılması ve okunması buna bir örnektir [34, 35]. 0 içeren baytların kodlanması daha önce enerji tasarrufu için önerilmiştir. Ancak değerlerin genişliği büyük olduğunda, her bayt için fazladan bir bit eklemek çok verimli bir sonuç değildir.

Enerji verimi için, donanım ve derleyici yöntemleri kullanarak anlam sıkıştırması yöntemi de önerilmiştir [36, 37]. Anlam sıkıştırmasında, değerlerin boyutu boru hattından bayt öge boyutunda aktarılır. Aksine, bizim bölümlenme yaklaşımımız her veri tutan bileşen için dar değerlerin boyutunun tasarım zamanında, tüm veri genişliği öge boyutunda, ayrı olarak tanımlanmasına dayanır.

İşlemciler içerisindeki değerlerin darlığı başarımlarını iyileştirmede de kullanılmıştır. [38]'de çoklu dar değer işlenenleri geniş işlev birimleri [30]'da ise geniş yazmaçlar içerisine paketlenmiştir. [39]'ta ise, yeniden adlandırma tablosu içine sığacak kadar dar değerler, yazmaç belirteci yerine yeniden adlandırma tablolarına koyulmuştur. [40]'de ise değerlerin darlığı tahmin edilmiş ve bu tahmin baz alınarak tek bir işlev biriminde çok sayıda komut işlenmiştir. Benzer bir yaklaşım [41]'de görülmektedir. Değer tahmin etme tabloları [42] ve [43]'te karmaşıklığı azaltmak için işlenene göre bölümlenmiştir. Literatürde görünen bir çok çalışma değerlerin genişliğinin yüksek oranda tahmin edilebilir olduğunu göstermektedir [30, 40, 42].

Yazmaç kütüğünün enerji tüketimini azaltmak çokça araştırma yapılan alanlardandır. Veri saklama birimlerinin enerji tüketimini azaltmak için çok değişik yöntemler vardır: yazmaç kütüğünü bölümlenmek, yazmaçların önbelleklerle kullanılması ve kapıların kaldırılması gibi. Aggarwal ve Franklin yazmaçların yüksek değerli bitlerin genellikle gerekmediğini öne sürerek bu bitlerdeki kapıların azaltılmasını önermişlerdir [44]. Yazmaçların önbelleklerle kullanılması, genellikle ulaşılan değerlerin daha küçük bir veri saklama biriminde saklanmasına dayanır [45] ve yazmaç kütüğünü birçok öbeğe bölerek hem daha küçük bir yapıya sahip olması hem de enerji tüketimini azaltılmasına yardımcı olmaktadır [30, 46, 47]. Değerlerin yazılmasının geciktirilerek veya okuma kapılarının çoğullayıcıdan geçirilerek kapıların azaltılması da yazmaç kütüğünün enerji tasarrufu açısından önerilmiştir [48, 49]. Enerji tüketimi ve erişim gecikmelerini düşürmek için çok öbekli iki seviyeli yazmaç kütüğü [50]'de önerilmiştir. Önerdiğimiz daha fazla enerji tasarrufu için bu iki yöntemle beraber uygulanabilir.

Kondo değerlerin darlığını kullanmış, ve 64 bitlik yazmaçları 32 bitlik parçalara bölüp, verimli bir şekilde ayırarak enerjinin verimli kullanılmasını sağlamıştır [51]. Her komut 32 bit parça ayırır ancak bağımlıları bir değeri ararken her parçayı kullanmayabilir. Bu tezdeki çalışma bazı yönleriyle [51]'ye benzemektedir, fakat esasen bu yöntem işleminin yayım kuyruğu gibi bazı diğer bileşenlerinde de düzenlemeler gerektirir ve bu durum bu yöntemi çok karmaşık kılar. Bağımlılar, bir yerine iki yazmaç etiketi hatırlamak zorundadır, ve bu da yazmaç kütüğünün dışındaki bileşenlerin enerji tüketimini artırmaktadır.

Gonzalez, yazmacın içeriğinden haberdar bir yazmaç kütüğü mimarisi önermektedir [53]. Yazmaç kütüğü 3 farklı yazmaç kütüğü içeren çoklu öbeklere bölünmüştür. Bu öbeklerden birisi dar değerleri saklayan yazmaç kütüğüdür. Eğer yazmaç kütüğü geniş bir değer tutması gerekirse, göstericiler farklı bir öbekte tutulur ve değerler düşük değerli bitlerini işaret eder. Bu mimari dar değere erişim olacağı zaman enerjiden tasarruf sağlar, ancak bir geniş değere erişim olacağı zaman başarımda düşüş meydana gelir.

Başarımı artırmak ve enerji tasarrufunu azaltmak için veri yolunu işlenen genişliğini temel alarak iki asimetrik topağa ayırmak [54, 55]'de önerilmiştir. Dar genişlikli işlenenlere sahip komutlar sadece dar işlev birimlerinin ve dar yazmaçların mevcut olduğu

topaklara atılmıştır. Çoğu komut işleneni dar olmadığına bu bölümlerle bir yüklemle dengesizliğine neden olur. Üretilen çoğu değerin dar olmasına rağmen dar topağa atılabilecek komutların yüzdesi sınırlıdır ve bu da başarımların artırımı ve enerji tasarrufu açısından çok verimli değildir.

Dar değerler işlemciler içerisindeki geçici hataları azaltmak için de kullanılmıştır [53, 56]. Bu çalışmada olduğu gibi dar değerleri ortaya çıkarıp, daha sonra dar kısımda işaret uzatması yapılarak saklanan verinin örselenebilirliği azaltılmıştır. Dar değer içerisindeki düşük değerli bitleri korumak için, değerlerin önemli kısmı veri saklama kısmında hata algılama ve korunma için kopyalanmıştır.

2.7. Sonuçlar

İşlemciler içerisinde saklanan değerlerin çoğu sağlanan veri yolu genişliğinden daha dar bir alanda saklanabilir. Bu bitlerin üst kısımları gereksizdir, ve işlemcinin veri saklama birimlerine gereksiz yere yazılıp, okunur. Bu çalışmada yazmaç kütüğünü iki kısma ayırılması önerilmiş, ve en iyi enerji tasarrufu için olabilecek en iyi dar değer genişliğini belirlenmiştir. Düşük değerli bitleri tutan bölüme her dar değer yazılmış ve daha sonra bu düşük değerli kısmın işaret biti uzatılarak gerçek değer oluşturulmuştur.

İşlenen genişliği dikkate alınarak yazmaç kütüğünün bölümlenmesi, dar değer genişliği 33 bit olarak belirlendiğinde enerji tüketiminde x86 bir makinede, spec2000 denektaşlarında ortalama olarak %36 civarında bir düşüş sağlamıştır. Her denektaşının enerji tasarrufu farklılıklar göstermektedir, bazı denektaşlarında %44 gibi yüksek enerji tasarrufları sağlamak mümkündür.

Dar değerleri belirlerken mimari düzeyinde ödünleşimlerden başka ayrıca devre düzeyinde de bazı ödünleşimler olduğu gösterilmiştir. İki farklı dar değer genişliği tanımı aynı enerji tasarrufuna neden olduğunda, her zaman daha geniş değeri tanım olarak almanın algılayıcı devrelerin gecikmesini düşürmesi açısından daha faydalı olacaktır gösterilmiştir.

Bu kısımda anlatılan işlenen genişliğinden haberdar bölümlerle yöntem, işlemci içerisinde başka veri tutan birimlere de uygulanabilir. Örneğin veri önbelleğinde ve

yayım kuyruğunun anında erişim alanında uygulanabilir. Dar değerlerin açığa çıkarılması uyku transistörleri kullanarak durağan enerji tüketimini azaltabilir. Saklanan değer dar olduğunda, yazmaç kütüğünün yüksek değerlikli kısımları kapatılabilir.

BÖLÜM 3

3. HIZLI HATA DÜZELTME KODLARI ÜRETİLMESİ

Çağdaş işlemcilerde, geçici hatalar önemi artan bir sorun haline gelmiştir. Saklanmış verilerdeki tek bitlik hataları algılamak için eşlik üreten devreler yaygın bir şekilde kullanılır. Saklanan değerlerin herhangi bir bitinin değişip değişmediği eşlik biti üretilerek anlaşılabilir. Basitçe, bir değer içerdiği 1'lerin tek veya çift olması kıstas alınır. Ham veriye eşlik biti eklenerek yeni korumalı değer oluşturulmuş olur. Çizelge 3.1.'de örnek ham değerler ve tercih edilen eşlik kıstasına göre üretilmiş, korumalı değerler görülmektedir. Bu tür bir hata koruma yönteminin zayıf tarafı sadece tek sayıda bit değiştiğinde hatanın farkedilebilmesidir.

Çizelge 3.1. Eşlik Üretimi

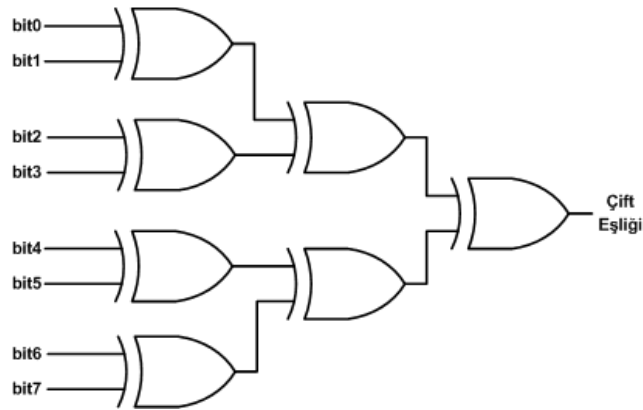
Ham Veri	Çift Eşliği	Tek Eşliği
0100	10100	00100
1111	01111	11111
1000	11000	01000
1100	01100	11100

Eşlik üreten devreler Özel VEYA ağaçlarından oluşur ve yüksek gecikme değerlerine sahiptir. Bu bölümde, eşlik üreten devre sunulup, gelen değer dar olduğunda daha hızlı eşlik üretmesi sağlanacaktır.

Her yeni nesil işlemcilerle asgari CMOS kanal uzunluğunun azalması, besleme gerilimlerinin düşürülmesiyle yüksek enerjili evrensel parçacıkların ve paketlemede ısı-malar nedeniyle geçici hataların artması beklenmektedir [25]. Geçici hataların devreler üzerinde kalıcı bir etkisi yoktur, ancak çalışan düzeneğin hata yapmasına ve çökmesine neden olabilir ki bu kabul edilebilir bir durum değildir. Hata düzeltme kodları ve eşlik bitleri mantıksal devrelerde bu tür hatalardan korunma için kullanılan yüksek gecikmesi olan devrelerdir. Bu bölümde, bu eşlik üretimindeki gecikme miktarını azaltabilecek dar değerlerin açığa çıkarılması üzerine bir yöntem sunulacaktır.

3.1. Hata Düzeltme Kodları ve Eşlik

Hata Düzeltme Kodları (HDK) ve eşlik günümüz işlemcilerinde hafıza birimlerini geçici hatalara karşı korumada sıkça kullanılan yöntemlerdendir. Eşlik ile sadece tek sayıda hatalar algılanabilirken, HDK kullanılarak tek bit hatalar düzeltilebilir ve iki bitlik hatalar algılanabilir. Bu duruma Tekli Hata Düzeltme, Çiftli Hata Algılama (THDÇHA) da denir. Eşlik koruması sayesinde bir değer içindeki 1'lerin sayılması ile tek sayıda hatalar kolaylıkla farkedilebilir. Bir değer içerisinde 1'lerin sayılması ise Şekil 3.1.'de gösterildiği gibi tüm bitlerin Özel VEYA'lanması ile olur. Özel VEYA kapıları diğer temel mantık kapılarına göre yavaş olduğundan ve devingen mantık kapılarının her bağımsız bit çiftini hesaplaması gerektiğinden devingen mantık kapıları da basit VE DEĞİL ve VEYA DEĞİL kapılarına kıyasla daha yavaştır. Eşlik bitleri kullanarak hatayı algılamak da aynı mantık devresini gerektirir. Hata algılama devresi eşlik bitini de içeren tüm bitleri Özel VEYA işlemine tabi tutar. Eğer bu işlemin sonucu eşlik bitinden farklı bir bit ise, bu değer bozulduğu anlamına gelir.



Şekil 3.1. 8 bitlik Eşlik Üretme Devresi

Hatalara karşı dayanıklılık gereksinimleri saklanmış veri üzerindeki hataların düzeltilmesini gerektirdiğinde veya bir eşlik biti kullanılıp daha fazla sayıda hata algılanacağı zaman HDK kullanılır. En çok kullanılan HDK'lerden birisi Hamming kodlarıdır. Hamming kodları hata algılama ve düzeltme için saklanan değerler için çoklu eşlik biti kullanır. Her eşlik biti değer içindeki bitlerin farklı kümeleriyle üretilir. Dolayısıyla, hamming kodları gerçekleştirilirken, çoklu eşlik biti üretici devreleri paralel olarak kullanılmalıdır. 8 bitlik bir $v_1v_2v_3v_4v_5v_6v_7v_8$ değeri için ikinin kuvvetleriyle belirtilen konumlara yerleştirilmiş dört adet parite biti, $p_1p_2p_3p_4$, kullanılır. Çizelge 3.2.'de buna bir örnek sunulmuştur.

Çizelge 3.2. HDK İle Eşlik Üretimi

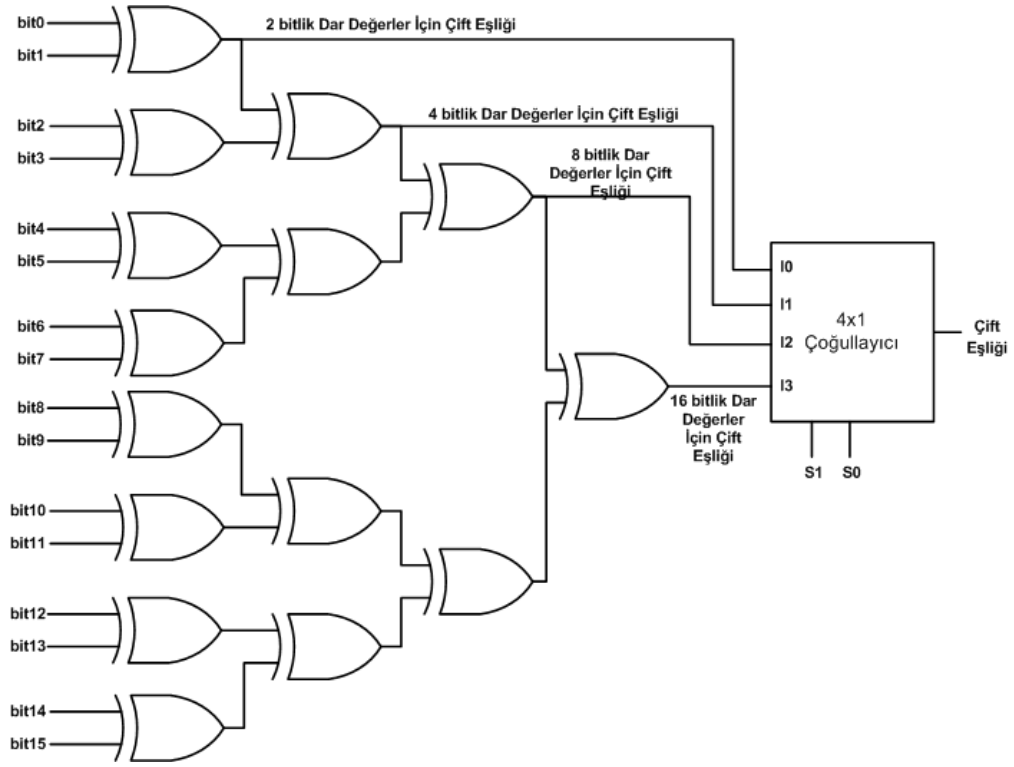
Ham Değer $v_1v_2v_3v_4v_5v_6v_7v_8$	HDK ile Korunmuş Hali $p_1p_2v_1p_3v_2v_3v_4p_4v_5v_6v_7v_8$
01010101	000110100101

Veri saklama alanından veri okunacağı zaman tüm eşlik bitleri tekrar hesaplanır ve doğrulanır. Eğer saklanmış veride bir hata olursa, hesaplanan eşlik bitlerinden bazılarının farklı değerleri olacaktır ve eşlik bitleri doğrulanmayacaktır. Hatanın meydana geldiği biti belirlemek için, 1 olan eşlik bitlerinin yerlerinin toplamı bulunur. Daha sonra düzeltme mantığı bu toplama sonucu elde edilen yanlış bitin değerini ters çevirerek, hatanın düzeltilmesini sağlamış olur. Hem HDK hem de eşlik bitleri birkaç ÖZEL VEYA kapısından oluşan şifreleme ve çözme mantık devrelerinden oluşurlar. Korunmalı hata saklama alanlarına yazılacak her bir yeni değer için bu şifreleme işlemi gereklidir, ve bu işlemin gecikmesi kaçınılmazdır. Dolayısıyla HDK ve eşlik bitleri sistemin çalışma hızını etkileyecek kritik yollarda kullanılmazlar, çünkü bu yollarda kullanılması tüm sistemin çalışma hızını düşürecektir. HDK genellikle sadece belleklerde kullanılır ve büyük şifreleme/çözme gecikmelerinden dolayı yazmaç kütüğü gibi veri yolu bileşenlerinde kullanılmaz.

3.2. Değişken Gecikmeli Eşlik Biti Üretim Devresi

İşlemcilerde üretilen ve tüketilen birçok değer dar değer olduğundan bir önceki bölümde bahsedilmişti. Bir ÖZEL VEYA işleminin tamamen 0'lardan oluşan veya çift sayıda 1'lerden oluşan bir değer üzerinde sonucu daima 0 olacağından, dar değerlerin üst kısmının çift sayıda olduğu sürece, üst kısımlarında ÖZEL VEYA işleminin uygulanmasına gerek yoktur. $0 \oplus X = X$ olduğundan, eşlik bitini tespit ederken dar değer sadece alt kısmının ÖZEL VEYA işlemi yeterli olacaktır. Dolayısıyla dar değerler için eşlik bitinin tespiti hızlandırılabilir. Bu yöntem yüksek gecikme zamanlarından dolayı eşlik bitleri üretilmeyen işlemci bileşenlerindeki değerlerden dar olanları için eşlik bitinin üretilmesini mümkün kılar. Ayrıca değer genişliğine bağlı olarak bir çevrimden daha fazla çevrimde eşlik biti üreten işlenen genişliğine bağlı gecikmeli eşlik üretme devreleri tasarlanabilir.

Gelen değer genişliğine göre değişken gecikmeli örnek bir eşlik üretme devresi Şekil 3.2.'de gösterilmiştir. ÖZEL VEYA ağacı değer değişik kısımları için eşlik bitini



Şekil 3.2. Önerilen Eşlik Üretme Devresi

hesaplar. Eşlik üretme devresinin iç düğümleri bir çoğullayıcıya bağlanmıştır. Çoğullayıcının seçme girişleri dar değer algılayıcılarının çıkışlarına bağlıdır. Eğer dar değer algılayıcıları gelen değer dar olduğunu belirtirse, çoğullayıcı ara düğümlerden gelen girdileri seçmektedir; aksi halde, değer geniş olması durumunda, eşlik üretim devresi ÖZEL VEYA ağacının çıkışını vermektedir. Görüldüğü gibi gelen değer dar olduğunda eşlik biti dar değer genişliğine göre 1 veya daha fazla seviye ÖZEL VEYA gecikmesine maruz kalmaktan kurtulmakta ve ÖZEL VEYA'dan daha hızlı bir çoğullayıcı kullanılması şartıyla eşlik bitinin üretilmesi hızlandırmaktadır. Eğer gelen değer geniş ise, bu durumda eşlik bitinin üretilmesi eski hale göre daha yavaşlayacaktır.

Şekil 3.2.'de görülen çoğullayıcı, aktarım kapıları kullanılarak gerçekleştirilmiştir ve Şekil 2.7.'de gösterilmektedir. Mantık 0 ve 1 algılayıcıları için önceki bölümde gösterilen devingen mantık kullanan devreler kullanılmıştır.

3.3. Benzetim Yöntemi

x86 işlemciler içerisindeki dar değerlerin kesin bir yüzdesini almak için, 32-bitlik x86 komutlarını işleme özelliğine sahip PTLsim işlemci benzeştiricisi kullanılmıştır [28]. Spec2000 denektaşı programlarını benzeştirici üzerinde kullanarak farklı iş yükleri için dar değerlerin yüzdesi elde edilmiştir. Tüm denektaşları gcc 3.4.3 kullanılarak ve 32 bitlik x86-64 komut kümesini hedef alacak şekilde azami en iyileştirme ile derlenmiştir. Sonuçlardan her denektaşında ilk bir milyar komut atlanmış, ondan sonraki iki yüz milyon komutun sonuçları göz önüne alınmıştır. Çizelge 2.2. benzetimde kullanılan işlemci mimarisini göstermektedir.

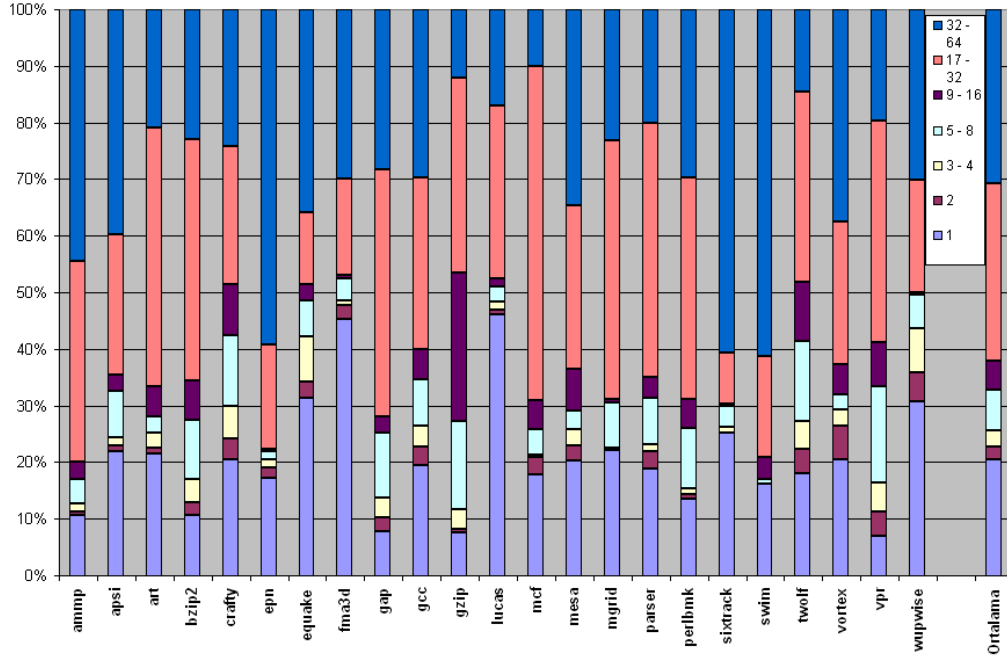
PTLsim ayrıca 64-bitlik x86 komutlarını çalıştırma özelliğine de sahiptir. Denektaşları 32-bitlik işlemciler için derlenmiştir. Ancak sonuçların 64 bitlik komut kümesi kullanıldığında da değişeceği düşünülmektedir.

Mentor Graphics'in tümleşik devre tasarım araçları, TSMC'nin 0.18 μ CMOS teknoloji kütüphaneleriyle kullanılıp eşlik üretme devrelerinin enerji tüketimi ve gecikmeleri hakkında tam sonuç almak için kullanılmıştır.

3.4. Benzetim Sonuçları

Önerilen yöntemin başarısı, ortak iş yüklerinde dar değerlerin varlığına dayanır. Günümüz iş yüklerinde üretilen ve tüketilen değerlerin çoğunun dar değer olduğu gözlemlenmiştir [31, 53]. Şekil 3.3. spec2000 denektaşlarında yazmaç kütüğüne yazılan değerlerin yüzdesini göstermektedir. ÖZEL VEYA ağaçlarının girdi sayılarına denk gelecek şekilde değerlerin genişlikleri gruplandırılmıştır. Şekilden de görülebileceği gibi, tüm denektaşlarında ortalama alındığında yazmaçlara yazılan değerlerin hemen hemen %50'si 8 bit veya daha az bit ile temsil edilebilir. Bu sayı bazı denektaşları için %75'e kadar çıkabilmektedir.

Eşlik üretim devresinin gecikmesi 2 kapılı tek bir ÖZEL VEYA kapısının gecikmesine bağlıdır. ÖZEL VEYA kapısı gerçekleşmesi kolay olmadığından, gecikmesi diğer kapılardan fazladır. Eşlik üretim devresi için değişik girdi sayılarına göre gecikme ve



Şekil 3.3. Spec2000 Denektaşlarında Dar Değerlerin Yüzdesi

enerji tüketimler Çizelge 3.3.'de verilmiştir.

Çizelge 3.3. ÖZEL VEYA Eşlik Devresinin Özellikleri

Bit	DYYZ (ps)	YDYZ (ps)	Enerji (fJ)
2	103,87	126,06	107,13
4	221,93	254,63	204,40
8	327,73	369,97	283,13
16	452,03	536,34	966,78
32	563,86	673,16	1748,05

Çizelgede eşlik üretim devresinin 2, 4, 8, 16 ve 32 girdili olması halindeki enerji tüketimini ve gecikmeleri göstermektedir. DYYZ çıktının, düşükten yükseğe yayılım zamanını; YDYZ ise yüksekten düşüğe yayılım zamanını gösteren gecikme ölçütleridir. Çizelgeden de görüldüğü gibi eşlik devresinin gecikmesi artan girdi sayısı ile artmaktadır. Şekil 3.2.'te gösterilen çoğullayıcının seçme girişlerinin 8 bit eşlik üretildiğinde hazır olacağı anlamına gelir. Bu yöntemin tek zayıf yönü, gelen değer dar olmadığı zaman değerlerin aktarımında fazladan bir çoğullayıcı gecikmesinin olmasıdır.

Ardışık sıfır algılama devresinin gecikme ve enerji tüketim değerleri Çizelge 3.4.'te

verilmiştir. 0 algılayıcısının gecikme değerleri eşlik devresine kıyasla çok daha yavaştır. Çizelge’de önyükleme değeri, çıkışın önyükleme evresinde mantıksal 0’dan 1’e çıkış zamanını temsil etmektedir. Bu zaman ölçümlerde, besleme geriliminin %10’undan %90’ına kadar çıkması için gerekli olan zaman gecikme olarak kabul edilmiştir.

Çizelge 3.4. Sıfır Algılayıcı 0.18 μ TSMC

Bit	Enerji (fJ)	YDYZ (ps)	Önyükleme (ps)
16	114,95	84,09	62,78
24	135,05	96,86	75,15
28	144,98	103,45	80,73
30	149,89	106,56	82,95

Ardışık 0 algılama devresinin iki türlü gecikme zamanı vardır: önyükleme gecikmesi ve değerlendirme gecikmesi. Ardışık 0 algılama devresi devingen mantık kullanılarak yapıldığı için devre saatinin her 0’a vurduğu zaman yukarı çekme transistörleri çıkış düğümünü mantık 1 değerine çeker. Çıkış düğümünün yüklenmesi için önyükleme zamanı kadar beklenmesi gerekmektedir. Önyükleme zamanı birinci bileşenidir, ikinci bileşeni ise değerlendirme zamanıdır. Değerlendirme zamanı saatin 1’e vurduğu zaman, girdilerdeki değişikliğe devrenin cevap verme süresidir. Örneğin, önyükleme çevriminde çıkış mantık 1 değerine çekildikten sonra, eğer devrenin değerlendirme çevriminde üreteceği değerler de mantık 1 değeri ise, bu durumda çıkış mantık 1 değerinde kalır ve gecikmesi 0 olmuş olur. Eğer bir önceki çevrimdeki değer de mantık 0 değerinde ise bu durumda düşükten yükseğe yayılım zamanı DYYZ 0 olur. Diğer taraftan eğer bir değerlendirme çevriminde girdiler çıkışı mantık 0 yapacak şekilde değişirse bu durumda girdideki değişikliklerin çıktıya yansımaları için bir zaman geçecektir, bu da YDYZ’dir. Dolayısıyla devrenin en kötü gecikme zamanı YDYZ ve önyükleme gecikmelerinin toplamına eşittir. Algılayıcının 30 girdiyle bile toplam gecikmesi 190 ps’den daha düşüktür ki bu değer de 8 bitlik eşlik üreten ÖZEL VEYA ağacının 369 ps olan gecikmesinden çok daha düşüktür.

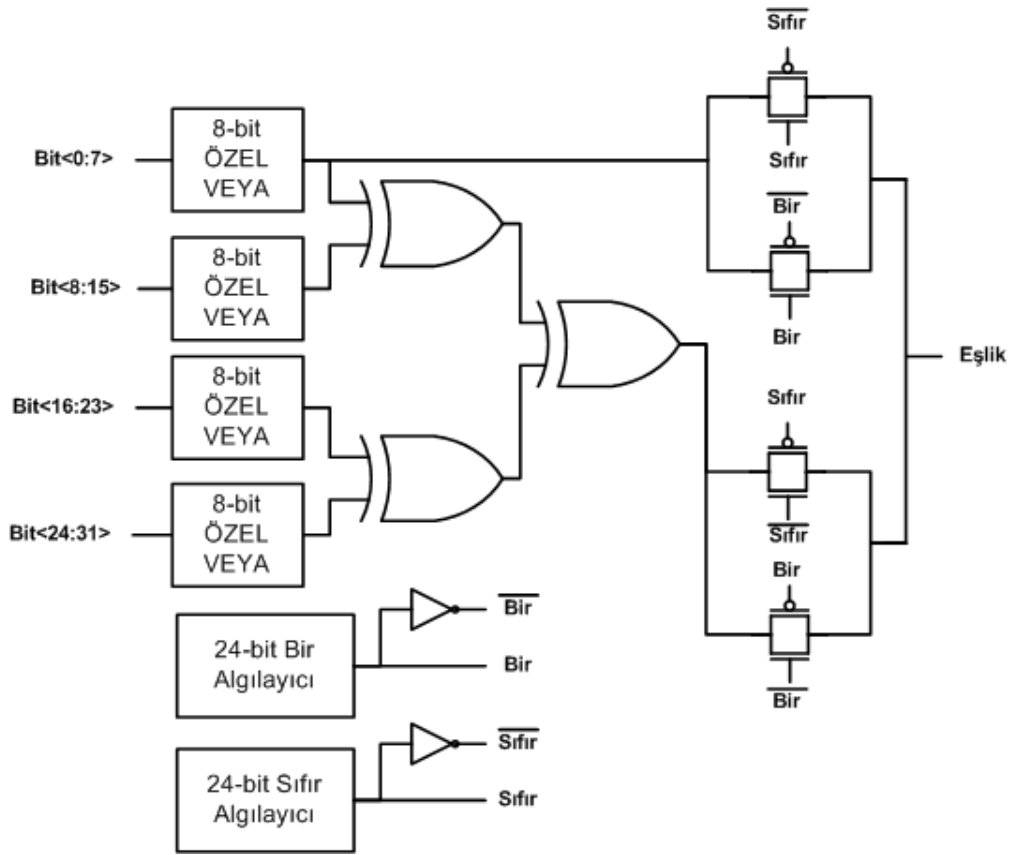
Önerilen devreyi denemek ve Şekil 3.2.’deki çoğullayıcının karmaşıklığını azaltmak için, dar değer genişliği 8 bit olarak tanımlanmıştır ve böylece sistem 8 bit ile temsil edilebilecek dar değerler için hızlı bir şekilde eşlik üretebilecektir. Bu durumda, Şekil 3.2.’deki devre yerine Şekil 3.4.’teki devre kullanılmıştır. Değerin üst kısımlarının aynı olup olmadığını anlamak için 24 bitlik 0 ve bir algılayıcıları paralel olarak konmuştur. Bu algılayıcıların çıkışları ve evrikleri Şekil 3.2.’deki çoğullayıcılarının seçme girişler-

ine bağlanmıştır. Girdi değeri 8 bitlik dar değer olduğunda, eşlik sonucu düşük değerli 8 bitin eşliğinin hesaplanması ile görece düşük bir aktarım kapasitesi gecikme farkıyla hemen hemen aynıdır. Dikkat edilmesi gereken nokta, dar değerlerin belirlenmesi ve 8 bitlik eşlik üretme işlemlerinin paralel olmasıdır ve 8 bitlik eşlik üretildiği zaman aktarım kapasitesinin girdileri çoktan kararlı hale gelmiş olacaktır.

Benzetim sonuçları eşlik bitinin 8 bitlik değerler için 452 ps, 32 bitlik değerler için 734 ps gecikme değerleri vermiştir. Bu 8 bitlik değerlerin eşliği için %32.9 bir hızlanma, diğer değerler için ise %9 bir yavaşlama anlamına gelmektedir.

3.5. Sonuçlar

Bu bölümde gelen değer dar olduğu zaman hızlı eşlik üreten bir eşlik üretme devresi önerildi. Önerilen devrenin çalışma esası işlemciler içerisinde üretilen ve tüketilen değerlerin çoğunun dar olması gözlemine dayanmaktadır. Yüksek gecikmeye yol açacağından tam koruma göz ardı edildiğinde önerilen yöntem kullanılabilir. Eğer işlemcinin iş yükü oransal olarak yüksek sayıda dar değer içeriyor ve veri saklama bileşenleri için kısmi koruma tasarımcıyı tatmin ediyorsa, önerilen çözüm sadece dar değerleri korumak için kullanılabilir. Ayrıca bu devre dar değer eşlikleri bir saat çevriminde, genişleri ise iki saat çevriminde üretebilir.



Şekil 3.4. 8 bitlik Dar Değerler İçin Hızlı Eşlik Üretici

BÖLÜM 4

4. DEVRELERİN UYANMA GECİKMELERİNİN BELİRLENME SÜRESİNİN KISALTILMASI

Sızıntıdan kaynaklanan güç tüketimini azaltmak için bugüne kadar pek çok teknik önerilmiştir. Bunların arasında en yaygın olarak kullanılan devrenin değişik yerlerinde farklı eşik gerilimleri kullanmaktır. Burada ana fikir hız açısından önemli olan devre yollarında düşük eşik gerilimi olan (yüksek hızlı ve yüksek güç tüketimli) transistörler kullanılması, genel devre hızı açısından önemsiz olan yollarda yüksek eşik gerilimli (düşük hızlı ve düşük güç tüketimli) transistörler kullanılmasıdır. Bu teknik devrenin genel gecikmesini değiştirmezken sızıntı akımından kaynaklanan güç tüketimini önemli oranda azaltır. Bu teknik seçilen eşik gerilimlerine bağlı olarak, yalnızca düşük eşik gerilimi kullanan devrelere göre güç tüketimini 2-10 kat arası azaltabilir [16, 19, 22, 23]. Transistörlerin besleme gerilimi ile toprak arasına yan yana seri olarak bağlanması da sızıntı akımını etkileyen bir başka etkidir. Seri olarak bağlanmış iki kapalı transistörden akan sızıntı akımının yarattığı güç tüketimi, tek bir kapalı transistörden akan sızıntı akımının yarattığı güç tüketiminden çok daha azdır. Bu gözlem pek çok devrenin durağan güç tüketimini azaltmakta kullanılmıştır. Literatürde devre girişleri transistörlerin en çok sayıda seri olacağı biçimde seçilmiştir [11, 24]. [13, 16, 18, 21]'de sızıntı akımının azaltılması için devre ile seri olarak bir uyku transistörü bağlanmıştır. Bu transistör devre çalışırken iletim durumunda devre anlamlı bir işlem yapmadığıdaysa kapalı durumdadır. Bu teknik devrenin çalışmadığı durumda gerçekleşen durağan güç tüketimini büyük ölçüde azaltır. Ancak bu tekniğin uyku transistörü üzerine düşen gerilim nedeniyle devre işlem yaptığı sırada işlem hızını yavaşlatıcı etkisi vardır. Aynı zamanda uyku transistörünün açılıp kapatılması sırasında da belirli bir zaman geçmekte ve devre uyku transistörü açıldıktan hemen sonra kullanılamamaktadır. Bu nedenle devrenin yeniden uyanması için geçen zamanının hesaplanması devrenin en az sürede işler duruma geçirilmesi ve uyku transistörünün yan etkilerini azaltmak için büyük önem taşımaktadır.

Durağan güç tüketiminin azaltılması için uyku transistörü kullanılan devrelerde devre uyku transistörü ile gerilim kaynağından ayrılmış durumda iken yeniden uyandırılmak

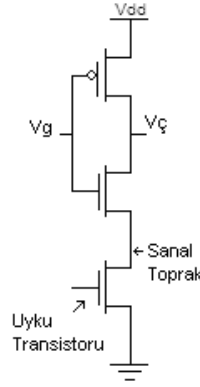
istendiğinde, devrenin tüm düğümlerindeki gerilimlerin ne kadar zamanda kararlı duruma geçeceğini bilinmesi önemlidir. Devrenin ara düğümlerindeki gerilimler kararlı duruma geçmeden devre yeniden kullanılamaz. Bu nedenle tasarım anında devrelerin ne kadar zamanda uyanacağını hesaplanması önemlidir.

Uyku transistörünün bağlı olduğu düğüm, transistör iletmediği durumda açıkta olduğu için, bu düğümün üzerine düşen gerilim kararlı duruma ulaşmaya kadar geçen zamanının benzetim yoluyla bulunması diğer sıradan CMOS devrelerinin gecikmelerinin benzetim yoluyla bulunmasına göre çok daha uzun zamanda gerçekleşmektedir. Çok yüksek sayıda transistöre sahip devreler sızıntı akımının azaltılması için uyku transistörleri kullanılarak tasarlandığında benzetim zamanının çok uzun olması nedeniyle uyanma süresinin doğru olarak hesaplanması mümkün olamamaktadır. Bu nedenle uyku transistörünün kullanıldığı durumlarda CMOS devrelerin benzetim zamanının azaltılması önem kazanmaktadır. Bu çalışmada CMOS devrelerin uyku transistörü dışında kalan kısmının davranışının modellenerek benzetim zamanının azaltılması önerilmektedir.

4.1. Davranış Modellemesi ile Benzetim Zamanının Azaltılması

Sızıntı akımını azaltmak amacıyla uyku transistörü kullanan devrelerin benzetim zamanını düşürebilmek için bu çalışmada uyku transistörü dışında kalan CMOS bileşenlerinin aktarım fonksiyonları polinomlarla modellenmiştir. Bu modelleme yapılırken bileşene uygulanan gerilim, V_G , ve uyku transistörünün yerleştirildiği düğümün (sanal toprak) gerilimi, V_{ST} , bağımsız değişkenler olarak alınmıştır. Uyku transistörü açıldığında, sanal toprak düğümü topraklanır ve devrenin gerçek topağına çok yakın bir değere erişir. Bu değer kararlı duruma gelmesinin benzetimi uyku transistörü bağlıyken çok uzun sürmekte, devre uyku transistörü olmadan benzetildiğinde ise benzetim kısa zamanda sonuçlanmaktadır. Bu nedenle devrenin uyku transistörlerinin dışında kalan bileşenlerinin bir polinomla modellenip bu polinomun uyanma zamanı benzetime bir kara kutu olarak katılması benzetim süresinden kazanç sağlamaktadır.

Bu yaklaşımın denenmesi için Şekil 4.1. 'de gösterilen uyku transistörlü bir eviricinin benzetimi seçilmiştir. Devre öbeklerinin davranışını uyanma zamanını bulmak için yapılan benzetime aktarmak için öbeklerin yerine polinom modelleri kullanılmıştır. Bu yöntemin Şekil 4.1. 'de görülen evirici devresine uygulanmış hali Şekil 4.2. 'de



Şekil 4.1. Uyku Transistörlü CMOS Evirici

gösterilmiştir.

Aktarım fonksiyonunun çıkarımında ilk adım, değişen V_G ve V_{ST} gerilimlerine göre oluşan çıkış gerilimi değerlerini, V_C , belirlemektir. CMOS evirici V_G ve V_{ST} 'yi $0V$ 'tan V_{DD} 'ye kadar $0.1V$ ve $0.05V$ 'lık artışlarla benzetilmiştir. Bu V_G ve V_{ST} 'ye karşılık gelen tüm çıkış gerilimi değerleri bir tabloya aktarılmıştır. Devre benzetim programı SPICE'in, polinom olarak modellenmiş aygıtları benzetme yeteneği kullanılarak Şekil 4.2.'de gösterilen evirici yerine benzetimlerden elde edilen tablonun iki değişkenli polinomlara çevrilmiş hali kullanılmıştır. Kullanılan bu tablo değişik girişler için devrenin sağladığı çıkış gerilimlerini göstermektedir. Tablolardaki satır ve sütunlarda kullanılan giriş gerilimlerinin arasında küçük artışlar olması tablodaki nokta sayısını ve dolayısıyla sonuçta ortaya çıkacak olan polinomun doğruluk oranını artırmaktadır. Bu tablolardaki gerilim değerleri MathCad yazılımı kullanılarak çeşitli dereceden polinomlar için polinom katsayıları hesaplanmış ve bu polinom katsayıları kullanılarak SPICE modelleri benzetim için hazırlanmıştır. Üretilen POLY isimli fonksiyon için yazılmış SPICE kodu aşağıdaki gibidir:

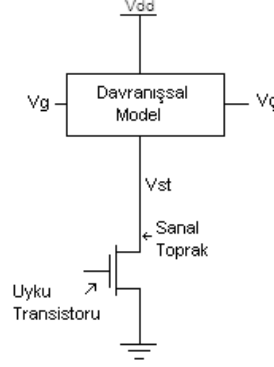
```

EXXX NODE+ NODE- POLY(degree) (x1, x2, ...) c0 c1 c2 .
. .

```

Bu ifadede yer alan EXXX, modelin gerilim ile denetlenen gerilim kaynağı olduğunu, NODE+ ve NODE- modelin çıkış düğümlerinin pozitif ve negatif düğümlerini, ve c_i 'ler de polinomun katsayılarını gösterir. x_i 'ler ise kullanılan polinom modelinin girişleridir. Örnek olarak kullanılan evirici için bu girişler V_{ST} ve V_G 'dir. Daha sonra değişik derecelerde polinomlarla modellenmiş eviriciler SPICE kullanılarak ben-

zetilmiş ve benzetim zamanları kaydedilmiştir. Elde edilen benzetim zamanları ve benzetim sonuçlarında sağlanan doğruluk oranları "sonuçlar ve açıklamalar" bölümünde ayrıntılı biçimde sunulmuştur.



Şekil 4.2. Davranış Modeli

4.2. Sonuçlar ve Açıklamalar

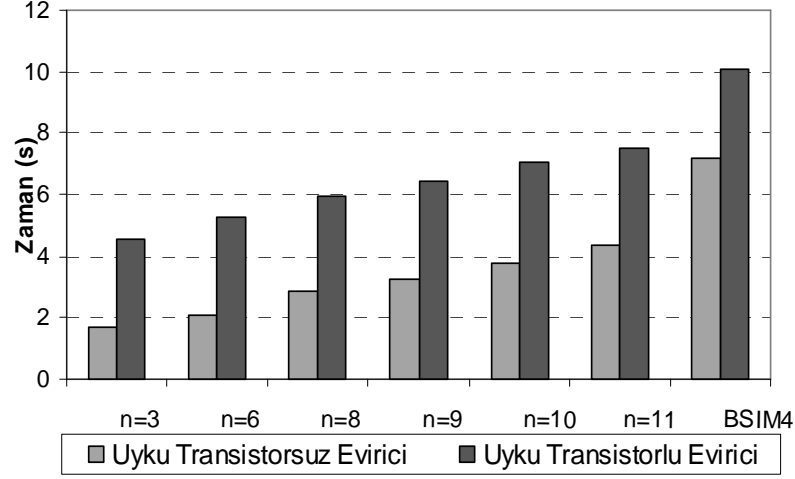
Tablo Çizelge 4.1. yapılacak benzetimde kullanılmak üzere çıkarılan tablo ile polinom model arasındaki Pearson ilinti katsayılarını göstermektedir. Derecesi n olan bir polinom için $(n+1)(n+2)/2$ katsayı hesaplanması gerekir ve bu durum aynı sayıda çarpım yapılmasına yol açar.

Çizelge 4.1. Tablo ve Üretilen Polinomlar Arasındaki İlintiler

Polinomun Derecesi	0,1 Artırımlı	0,05 Artırımlı
3	0,932	0,930
5	0,966	0,965
7	0,980	0,979
8	0,981	0,979
9	0,984	0,984
10	0,984	0,986
11	0,988	0,987

Şekil 4.2.'de gösterilen evirici hem polinomlarla modellenmiş hem de fiziksel BSIM4 [17] modeli kullanılarak benzetilmiştir. Yapılan iki ayrı benzetimin tamamlanma süreleri Şekil 4.3. 'te gösterilmiştir. Benzetim sonuçları BSIM4 ile karşılaştırıldığında önerilen 11. dereceden polinom modelinin benzetim zamanını 1,69 kat düşürdüğü gözlemlenmiştir. Alınan sonuçlar daha düşük dereceli polinom modellerinin kullanılmasının

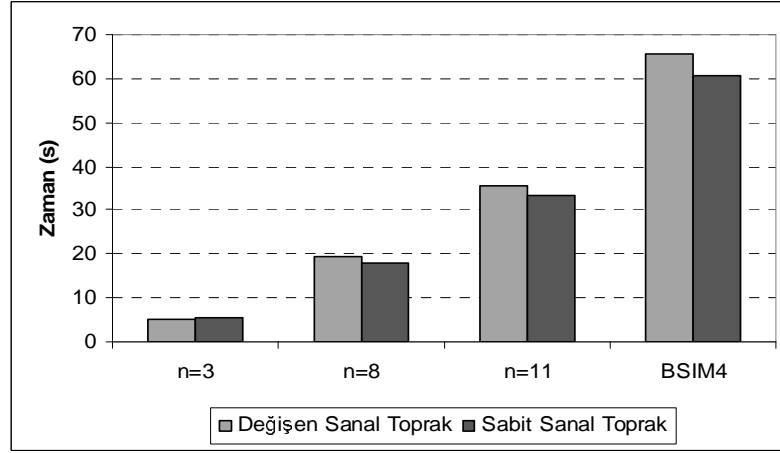
benzetim zamanını daha fazla düşürdüğünü, ancak benzetim sonuçlarının doğruluğunun düşük dereceli polinomlar kullanıldığında daha az olduğunu göstermiştir.



Şekil 4.3. Tek Evirici İçin İşlemci Yürütümü Zamanı

Şekil 4.3.'te sol tarafta gösterilen sonuçlarda her polinom derecesi için sol taraftaki çubuk, uyku transistorsuz eviriciler için benzetim zamanlarını, sağ taraftaki çubuk ise uyku transistörü bağlanmış eviricilerdeki benzetim zamanlarını göstermektedir. Şekilde görüldüğü gibi 11. dereceden karmaşık bir polinom bile benzetim zamanında çok önemli bir düşüş sağlamıştır. Benzetilen örnek devrelerde yalnızca işlevi gören devrelerin davranışı modellenmiş uyku transistörünün davranışı ise modellenmemiştir. Eğer uyku transistörünün davranışı da devrenin geri kalanı gibi ayrıca modellenirse, benzetim zamanı daha da düşecek ancak benzetim sonunda bulunan sonuçların doğruluğu da azalan benzetim zamanıyla birlikte azalacaktır.

Önerilen modelin daha büyük devrelerde etkisinin gözlenmesi için, art arda bağlanmış 10 eviricinin benzetimi yapılmıştır. Şekil 4.4. bu devrelerin benzetimi sırasında elde edilen koşma zamanlarını göstermektedir. Şekilde gösterilen her bir polinom derecesi için sağ taraftaki çubuklar sanal toprak geriliminin sabit tutulduğunda elde edilen benzetim zamanlarını, sol taraftaki çubuklar ise sanal toprağın aldığı gerilim değeri serbest bırakıldığında ölçülen benzetim zamanını göstermektedir. Sanal toprağın gerilim değeri sabitlendiğinde gözlemlenen benzetim zamanı 11. dereceden bir polinom için 1,81 katlık bir hızlanma göstermektedir. Ancak genelde uyku transistörü kullanan devrelerde V_{ST} geriliminin aldığı değer devingen olarak değiştiğinden Şekil 4.4.'te her bir polinom derecesi için sol yanda gösterilen çubuklar daha büyük önem taşır.



Şekil 4.4. 10 Evirici İçin İşlemci Yürütümü Zamanı

Değişken gerilimli sanal toprak kullanılarak yapılan benzetimlerde 11. derece polinom modeli kullanıldığında benzetim zamanında 1,84 kat, 8. derece polinom modeli kullanıldığında ise benzetim zamanında 3,64 katlık bir azalma sağlanmıştır.

Çizelge 4.2. Tablo ve Üretilen Polinomların Karşılaştırılması

Model	N=8	N=11	BSIM4
Yürütme Zamanı (sn)	19,27	35,52	65,70
Yineleme	141945	141330	141261
Devre Denklemleri	25	25	59

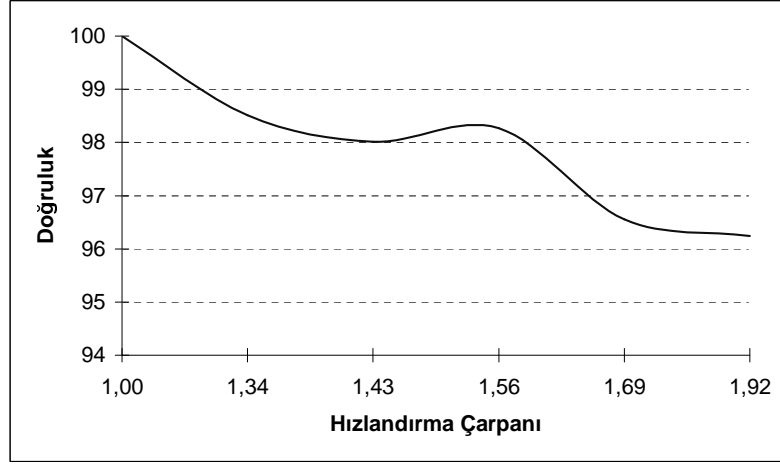
Tablo Çizelge 4.2., fiziksel model ile önerilen modelleme tekniğinin devre denklemleri sayılarını, benzetim zamanlarını ve sonuca yaklaşmak için yapılan yineleme sayılarını göstermektedir. Sonuçlar, yineleme sayılarının tüm modeller için yaklaşık aynı olmasına rağmen, fiziksel modelin polinom modelden daha fazla denklem sayısı olduğunu göstermektedir. Benzetim zamanı yineleme sayısı, (N_Y), ve devre denklem sayısı, (N_D), çarpımıyla doğru orantılıdır. Bundan dolayı 8. dereceden polinom modelinin yineleme sayısı fiziksel modelden fazla olmasına rağmen, polinom modeli kullanıldığında daha az devre denklemleri çözüldüğü için yapılan benzetim daha kısa sürmüştür. Fiziksel modelin $N_Y \times N_D$ çarpımının, 8. dereceden polinom modelinkine oranı 2,34 iken benzetim zamanı oranı 3,43'tür. Bu sonuç BSIM4 fiziksel CMOS modelinin denklemlerinin, iki değişkenli 8. derece polinoma göre neden daha fazla zaman aldığını göstermektedir.

Çizelge 4.3. Sanal Toprağın Kararlı Hale Gelmesi İçin Gereken Zaman

Model	t_k (ps)	İşlemci Zamanı(s)	Hata(%)
BSIM4	107,60	10,080	—
N=11	109,20	7,508	1,484
N=10	109,746	7,064	1,985
N=9	109,474	6,476	1,733
N=8	111,316	5,964	3,444
N=6	111,650	5,256	3,755
LEVEL 1	0,600	2,028	99,434
LEVEL 2	55,200	2,528	48,703
LEVEL 3	76,078	2,256	29,301
LEVEL 52	127,660	3,932	18,633

Dikkat edilmesi gereken diğer bir konu, benzetim sonuçlarının fiziksel model kullanılarak yapılan benzetim sonuçları ile olan tutarlılığıdır. Tutarlılığı değerlendirmek için, benzetilen eviriciye bir uyku transistörü bağlanmış, hem BSIM4 fiziksel modeli ile hem de üretilen polinom modelleri ile benzetimleri yapılmıştır. Tablo 3 uyku transistörü açıldıktan sonra sanal toprağın 0V'ta kararlı hale gelebilmesi için gereken zamanı, t_k , mikroişlemcide benzetim programının koşma zamanını ve elde edilen sonucun fiziksel model kullanılarak elde edilen sonuca göre bağlı hatasını göstermektedir. Bir eviriciyi polinomlarla ifade etmek çok fazla basitleştirme olarak görülse de, davranış modelleri kullanılarak elde edilen benzetim sonuçlarının beklenen değerlere çok yakın olduğu Çizelge 4.3. 'te görülmektedir. Bulunan sonuçlar daha önce geliştirilmiş diğer SPICE MOSFET modelleri kullanılarak yapılan benzetimlerin sonuçlarıyla da karşılaştırılmıştır. Bu modellerden LEVEL 1, 2 ve 3 genellikle kanal uzunluğu 1-10 μm olan devreler için geçerlidir; LEVEL 52 ise kısa kanal etkilerini de göz önüne alan BSIM3V3'tür.

Şekil 4.5. polinom modellerinin hızlandırma ve doğruluk ödünleşim eğrisini göstermektedir. Bu şekilde doğruluk BSIM4 modelinin sonuçlarına göre tanımlanmıştır. Şekilde de görüldüğü gibi doğruluktan bir miktar fedakarlık ederek benzetim zamanında önemli bir düşüş elde edilebilir.



Şekil 4.5. Polinom Modellerde Doğruluk ve Hızlandırma

4.3. Sonuç

Bu çalışmada uyku transistörü kullanan devrelerin benzetim zamanının düşürülmesi için bir yöntem sunulmuştur. Benzetimi yapılan kapıların, davranışlarını temsil eden bir eğri ile değiştirilmesiyle benzetim sonuçlarından çok az oranda kayıp vererek, benzetim zamanlarından yüksek kazanımlar sağlanmıştır. Örnek olarak seçilmiş uyku transistörlü eviricilerin benzetimleri, altıncı dereceden polinom için doğruluktan yalnızca %3,75 ödün vererek yaklaşık iki kat hızlandırma ve onbirinci dereceden polinom için benzetim sonucu doğruluğundan %1,484 oranında kayıp ile 1,342 kat hızlandırma elde edilebileceğini göstermiştir.

KAYNAKLAR

- [1] Moore, G., Cramming More Components into Integrated Circuits, *Electronics*, Vol. 38, Nr 8, 1965.
- [2] Rabaey, J. M., Chandrakasan, A., Nikolic, B., *Digital Integrated Circuits: A Design Perspective*, Prentice-Hall, Upper Saddle River, 2004.
- [3] Gelsinger, P., *Microprocessors for the New Millennium Challenges, Opportunities and New Frontiers*, ISSCC 2001.
- [4] De, V., and Borkar, S., *Technology and Design Challenges for Low Power and High-Performance*, International Symposium on Low Power Electronics and Design, 163-168, San Diego, U.S.A., Ağustos 1999.
- [5] Veendrick, H., Short-circuit dissipation of static CMOS circuitry and its impact on the design of buffer circuits. *IEEE Journal of Solid-State Circuits*, 19, 468-473, 1984.
- [6] Roy, K., Mukhopadhyay, S., Meimand, H., *Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicron CMOS Circuit*, Proceedings of the IEEE, 91, 302-327, 2003.
- [7] Allen, P.E., Holberg, D.R., *CMOS Analog Circuit Design*, New York: Oxford University Press, 2002.
- [8] Borkar, S., *Design Challenges of Technology Scaling*, *IEEE Micro*, pp.23-29, Ağustos, 1999.
- [9] Hodges, D.A., Jackson, H.G., Saleh, R.A. *Analysis and Design of Digital Integrated Circuits: In Deep Submicron Technology*, McGraw-Hill 3rd Edition, 2003.
- [10] Weste, N.H.E., Harris, D., *CMOS VLSI Design: A Circuits and Systems Perspective*, Addison-Wesley, 2005.
- [11] Halter, J.P., Najm, F., *A Gate-Level Leakage Power Reduction Method for Ultra-Low-Power CMOS Circuits Custom Integrated Circuits Conference*, 475-478, 1997.
- [12] Angelov, G., Hristov, G., *SPICE Modeling of MOSFETs in Deep Submicron*” 27th International Spring Seminar on Electr. Technology (ISSE), Bankya, Bulgaria, Book 2, 257-262, Mayıs 2004.
- [13] Inukai, T., *Boosted-gate MOS (BGMOS): Device/Circuit Cooperation Scheme*

- to Achieve Leakage-Free Giga-Scale Integration, Custom Integrated Circuits Conference, 409-412, 2000.
- [14] Kang, S. M., Leblebici, Y., CMOS Digital Integrated Circuits: Analysis and Design, New York: McGraw-Hill, 1999.
- [15] Kao, J. T., Chandrakasan, A. P., Dual-Threshold Voltage Techniques for Low-Power Digital Circuits, IEEE JSSC, 35(7):1009-1018, Temmuz 2000.
- [16] Ketkar, M., Sapatnekar, S. S., Standby Power Optimization via Transistor Sizing and Dual Threshold Voltage Assignment, International Conference on Computer Aided Design (ICCAD), 375-378, 2002.
- [17] Liu, W., Jin, X., Cao, K. M., Hu, C., BSIM4.0.0 MOSFET Model Users' Manual Berkeley, CA, Univ. California, 2000.
- [18] Mutoh, S., Douseki, T., Matsuya, Y., Aoki, T., Shigematsu, S., Yamada, J., 1-V Power Supply High-Speed Digital Circuit Technology with Multithreshold-Voltage CMOS, IEEE Journal of Solid-State Circuits, 30, No. 8, 847-854, Ağustos 1995.
- [19] Nguyen, D., Davare, A., Orshansky, M., Chinnery, D., Thompson B., Keutzer, K., Minimization of Dynamic and Static Power Through Joint Assignment of Threshold Voltages and Sizing Optimization, International Symposium on Low Power Electronics and Design (ISLPED), 158-163, 2003.
- [20] Schichman, H., Hodges, D. A., Modeling and Simulation of Insulated-Gate Field-Effect Transistors, IEEE Journal of Solid-State Circuits, vol. SC-3, no. 5, 285-289, 1968.
- [21] Shigematsu, S., Mutoh, S., Matsuya, Y., Tanabe, Y., Yamada, J., A 1-V High-Speed MTCMOS Circuit Scheme for Power-Down Application Circuits IEEE JSSC, 32(6):861-869, Haziran 1997.
- [22] Wang, Q., Vrudhula, S. B. K., Static Power Optimization of Deep Submicron CMOS Circuits for Dual VT Technology, International Conference on Computer Aided Design (ICCAD), 490-496, 1998.
- [23] Wei, L., Chen, Z., Roy, K., Ye, Y., De, V., 'Mixed-Vth (MVT) CMOS Circuit Design Methodology for Low Power Applications, Design Automation Conference (DAC), 430-435, 1999.
- [24] Ye, Y., Borkar, S., De, V., A new technique for standby leakage reduction in high performance circuits using transistor stack effects, International Symposium on VLSI Circuits, 40-41, 1998.

- [25] Semiconductors Industry Association: International Technology Roadmap for Semiconductors 2005, <http://www.itrs.net/Links/2005ITRS/Home2005.htm>.
- [26] Ünsal, O. S., Tschanz, J. W., Bowman, K., De, V., Vera, X., Gonzalez, A., and Ergin, O., Impact of Parameter Variations on Circuits and Microarchitecture, IEEE Micro Magazine, Vol. 26, No. 6, , 30-39, Kasım-Aralık 2006
- [27] Schmookler, M.S., Nowka, K.J., Leading Zero Anticipation and Detection: A Comparison of Methods, 15th IEEE Symposium on Computer Arithmetic (ARITH-15 '01), 2001.
- [28] Yourst, M. T., PTLsim User's Guide and Reference: The Anatomy of an x86-64 Out of Order Microprocessor, Technical report, www.ptlsim.org.
- [29] BSIM 4.5.0 Manual, available at <http://wwwdevice.eecs.berkeley.edu/~bsim3/bsim4.html>
- [30] Ergin, O., Balkan, D., Ghose, K., and Ponomarev, D., Register Packing: Exploiting Narrow-Width Operands for Reducing Register File Pressure, International Symposium on Microarchitecture, 2004.
- [31] Loh, G., Exploiting Data-Width Locality to Increase Superscalar Execution Bandwidth, in Proc. of the International Symposium on Microarchitecture, 2002.
- [32] Hinton, G., Sager, D., Upton, M., Boggs, D., The Microarchitecture of the Pentium 4 Processor, Intel Technology Journal, Q1, 2001.
- [33] Kumar, S., Pujara, P. and Aggarwal, A., Bit-Sliced Datapath for Energy-Efficient High Performance Microprocessors, in Proceedings of 4th Workshop on Power Aware Computer Systems (PACS 04) held in conjunction with MICRO-37, Portland, Oregon, USA, Aralık 2004.
- [34] Ponomarev, D., Küçük, G., Ergin, O., Ghose, K. and Kogge, P. M., Energy Efficient Issue Queue Design, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 11, No.5, October 2003, pp.789-800.
- [35] Villa, L., Zhang, M. and Asanovic, K., Dynamic Zero Compression for Cache Energy Reduction, in Micro-33, Dec. 2000.
- [36] Canal R., Gonzales A., and Smith J., Very Low Power Pipelines using Significance Compression, Proc. of the International Symposium on Microarchitecture, 2000.
- [37] Canal R., Gonzales A., and Smith J., Very Low Power Pipelines using Significance Compression, Proc. of the International Symposium on Microarchitecture,

2000.

- [38] Brooks, D. and Martonosi, M., Dynamically Exploiting Narrow Width Operands to Improve Processor Power and Performance, Proc. HPCA, 1999.
- [39] Lipasti, M., Mestan, B. R., and Gunadi, E., Physical Register Inlining, ISCA, 2004
- [40] Loh, G., Exploiting Data-Width Locality to Increase Superscalar Execution Bandwidth, in Proc. of the International Symposium on Microarchitecture, 2002.
- [41] Nakra, T., et.al., Width Sensitive Scheduling for Resource Constrained VLIW Processors, Workshop on Feedback Directed and Dynamic Optimizations, 2001.
- [42] Loh, G., Width Prediction for Reducing Value Predictor Size and Power, in First Value Prediction Workshop, ISCA, 2003.
- [43] Sato, T., Arita, I., Table Size Reduction for Data Value Predictors by Exploiting Narrow Width Values, in Proc. of the International Conference on Supercomputing, 2000.
- [44] Aggarwal, A., Franklin, M., Energy Efficient Asymmetrically Ported Register Files, in Proceedings of International Conference on Computer Design (ICCD), 2003.
- [45] Butts, A., Sohi, G., Use-Based Register Caching with Decoupled Indexing, in Proc. of the International Symposium on Computer Architecture, 2004.
- [46] Borch, E., Tune, E., Manne, S., Emer, J., Loose Loops Sink Chips, in Proc. of International Conference on High Performance Computer Architecture (HPCA-8), 2002.
- [47] Wallase, S., Bagherzadeh, N., A Scalable Register File Architecture for Dynamically Scheduled Processors, in Proceedings. of International Conference on Parallel Architectures and Compilation Techniques (PACT-5), 1996.
- [48] Park, I., Powell, M., Vijaykumar, T., Reducing Register Ports for Higher Speed and Lower Energy, in Proc. of Intl. Symposium on Microarchitecture (MICRO-35), 2002.
- [49] Kim, N., Mudge, T., "Reducing Register Ports Using Delayed Write- Back Queues and Operand Pre-Fetch", in Proc. of Int.l Conference on Supercomputing (ICS-17), 2003.
- [50] Balasubramonian, R., Dwarkadas, S., Albonesi, D., Reducing the Complexity of the Register File in Dynamic Superscalar Processor, in Proceedings of the International Symposium on Microarchitecture (MICRO-34), 2001.

- [51] Kondo, M., Nakamura, H., A Small, Fast and Low-Power Register File by Bit-Partitioning, HPCA, 2005.
- [52] Ergin, O., Ünsal, O., Vera, X., and González, A., Exploiting Narrow Values for Soft Error Tolerance, IEEE Computer Architecture Letters (CAL), Vol. 5, 2006.
- [53] Ergin, O., Ünsal, O., Vera, X., and González, A., Exploiting Narrow Values for Soft Error Tolerance, IEEE Computer Architecture Letters (CAL), Vol. 5, 2006.
- [54] Gonzalez, R., Cristal, A., Pericas, M., Veidenbaum, A., Valero, M., An Asymmetric Clustered Processor based on Value Content, ICS, 2005.
- [55] Ünsal, O., Ergin, O., Vera, X., Gonzalez, A., Empowering a Helper Cluster through Data Width Aware Instruction Steering Policies, in Proceedings of 20th International Parallel and Distributed Processing Symposium (IPDPS-20), Rhodes, Greece, April 2006.
- [56] Hu, J., Wang, S., Ziavras, S., In-Register Duplication: Exploiting Narrow-Width Value for Improving Register File Reliability, in DSN 2006.

Kişisel Bilgiler

Soyadı, adı : HANAY, Y. Sinan
Uyruđu : T.C.
Doğum tarihi ve yeri : 14.04.1983 Erzurum
Medeni hali : Bekar
Telefon : 0 (312) 221 10 48
Faks : 0 (312) 292 40 91
e-mail : hanay@etu.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Lisans	Sabancı Üniversitesi / Mikroelektronik	2005

İş Deneyimi

Yıl	Yer	Görev
2004-2006	TOBB ETÜ	Araştırma Görevlisi

Yabancı Dil

İngilizce

Yayınlar

Y. Sinan Hanay, et al. "Formation Control with Potential Functions and Newton Iteration" ,European Control Conference 2007, Greece