

T.C.  
ONDOKUZ MAYIS ÜNİVERSİTESİ  
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ  
FİZİK ANA BİLİM DALI



KUANTUM FOURİER DÖNÜŞÜMÜ TABANLI BAZI  
KUANTUM ARİTMETİK İŞLEMLERİN  
GELİŞTİRİLMESİ VE UYGULAMALARI

Doktora Tezi

**Murat KURT**

Danışman  
**Prof. Dr. Azmi GENÇTEN**

II.Danışman  
**Doç. Dr. Selçuk ÇAKMAK**

Bu çalışma 122F298 numaralı TÜBİTAK projesi ile desteklenmiştir.

SAMSUN

2024

# TEZ KABUL VE ONAYI

Murat KURT tarafından, Prof. Dr. Azmi GENÇTEN ve Doç. Dr. Selçuk ÇAKMAK danışmanlığında hazırlanan "KUANTUM FOURIER DÖNÜŞÜMÜ TABANLI BAZI KUANTUM ARİTMETİK İŞLEMLERİN UYGULAMALARI VE GELİŞTİRİLMESİ" başlıklı bu çalışma, jürimiz tarafından 12/02/2024 tarihinde yapılan sınav sonucunda oy birliği ile başarılı bulunarak Doktora Tezi olarak kabul edilmiştir.

	Ünvanı Adı Soyadı	
	Üniversitesi	
	Ana Bilim / Ana Sanat Dalı	Sonuç
Başkan	Prof. Dr. Emin ÖZTEKİN	<input type="checkbox"/> Kabul
	Ondokuz Mayıs Üniversitesi	
	Fizik Ana Bilim Dalı	<input type="checkbox"/> Ret
Üye	Prof. Dr. Azmi GENÇTEN	<input type="checkbox"/> Kabul
	Ondokuz Mayıs Üniversitesi	
	Fizik Ana Bilim Dalı	<input type="checkbox"/> Ret
Üye	Prof. Dr. Sedat AKLEYLEK	<input type="checkbox"/> Kabul
	Ondokuz Mayıs Üniversitesi	
	Bilgisayar Müh. Ana Bilim Dalı	<input type="checkbox"/> Ret
Üye	Prof. Dr. Ferdi ALTINTAŞ	<input type="checkbox"/> Kabul
	Bolu Abant İzzet Baysal Üni.	
	Fizik Ana Bilim Dalı	<input type="checkbox"/> Ret
Üye	Doç. Dr. Abdulkadir KARACI	<input type="checkbox"/> Kabul
	Samsun Üniversitesi	
	Yazılım Müh. Ana Bilim Dalı	<input type="checkbox"/> Ret

Bu tez, Enstitü Yönetim Kurulunca belirlenen ve yukarıda adları yazılı jüri üyeleri tarafından uygun görülmüştür.

Prof. Dr. Ahmet TABAK  
Enstitü Müdürü

# BİLİMSEL ETİĞE UYGUNLUK BEYANI

Hazırladığım Doktora tezinin bütün aşamalarında bilimsel etiğe ve akademik kurallara riayet ettiğimi, çalışmada doğrudan veya dolaylı olarak kullandığım her alıntıya kaynak gösterdiğimi ve yararlandığım eserlerin Kaynaklar'da gösterilenlerden oluştuğunu, her unsurun enstitü yazım kılavuzuna uygun yazıldığını ve TÜBİTAK Araştırma ve Yayın Etiği Kurulu Yönetmeliği'nin 3. bölüm 9. maddesinde belirtilen durumlara aykırı davranılmadığını taahhüt ve beyan ederim.

Etik Kurul Gerekli mi ?

Evet

Hayır

12/02/2024

Murat KURT

## TEZ ÇALIŞMASI ÖZGÜNLÜK RAPORU BEYANI

**Tez Başlığı:** KUANTUM FOURIER DÖNÜŞÜMÜ TABANLI BAZI KUANTUM ARİTMETİK İŞLEMLERİN GELİŞTİRİLMESİ VE UYGULAMALARI

Yukarıda başlığı belirtilen tez çalışması için şahsım tarafından 05/01/2024 tarihinde intihal tespit programından alınmış olan özgünlük raporu sonucunda;

Benzerlik oranı : % 9

Tek kaynak oranı : % 1 çıkmıştır.

12/02/2024

Prof. Dr. Azmi GENÇTEN

# ÖZET

## KUANTUM FOURIER DÖNÜŞÜMÜ TABANLI BAZI KUANTUM ARİTMETİK İŞLEMLERİN GELİŞTİRİLMESİ VE UYGULAMALARI

Murat KURT

Ondokuz Mayıs Üniversitesi

Lisansüstü Eğitim Enstitüsü

Fizik Anabilim Dalı

Doktora, Şubat 2024

Danışman: Prof. Dr. Azmi GENÇTEN

II. Danışman: Doç. Dr. Selçuk ÇAKMAK

Kuantum bilgi işleme bilginin paralel işlenebilmesine olanak sağladığı için klasik bilgi işlemeden üstündür. Kuantum bilgi işlemede kullanılacak olan farklı kuantum algoritmalar geliştirilmiştir. Bunlardan bazıları QFT tabanlı kuantum algoritmalar olarak bilinir. Klasik bitlere kodlanmış bilginin süperpozisyon durumuna getirilerek kuantum hesaplama alanına taşınması QFT algoritması sayesinde olur. Bu çalışma QFT tabanlı kuantum aritmetik işlemler yapan kuantum devrelerinin geliştirilmesi ve uygulamalarıyla ilgilidir. İlk kez bu çalışmada  $n$  bitlik ikiden fazla sayının toplamını gerçekleştiren QFT tabanlı toplama için bir kuantum devre önerilmiştir. Bu öneride, her toplama işleminde tekrarlanan QFT-IQFT kullanılmadığı için gerekli kapı sayısında büyük oranda azalma olduğu gösterilmiştir. İşlemlerin kudit sistemlerle yapılması durumunda kapı sayısındaki azalma oranının çok daha fazla olduğu gösterilmiştir. Sonucu pozitif ve negatif çıkabilen QFT tabanlı çıkarma işlemlerini gerçekleştiren kuantum devre önerisi yapılarak örnekler üzerinde çalıştığı gösterilmiştir. Negatif sonuçlu çıkarma işlemleri için gerekli olan karşılaştırma işlemi yapan kuantum devre tasarımı yapılmıştır. QFT tabanlı toplama işlemini gerçekleştiren kuantum devre yapısı kullanılarak çarpma işlemi yapan devre geliştirilmiştir. Tek kübit ve tek kukuartlık sayılar arasındaki olası tüm çarpımları gerçekleştiren devreler weyl operatörleri kullanılarak oluşturulmuştur. Bölme işlemi için ise kendini tekrarlayan QFT tabanlı çıkarma işlemi barındıran kuantum devresi geliştirilmiştir. Geliştirilen bu devrede kaç kez çıkarma işlemi yapıldığı bilgisi gerekli olacağından bir kuantum sayaç tasarımı yapılmıştır. İlave olarak bu sayacın geri sayım yapısını da önerilerek  $d$  seviyeli herhangi bir kuantum sistemi için genelleştirilmiştir. Klasik bilgi işlemedeki NAND mantık kapısı, QFT tabanlı olarak tasarlanmıştır. Son olarak toplama ve NAND işlemlerini seçici girişler kullanarak tek bir devre üzerinde gerçekleştiren QFT tabanlı basit bir qALU tasarımı ilk kez bu çalışmada ortaya konulmuştur. Bu çalışmamızda geliştirilen tüm kuantum devrelerin kübitler için olanlarının doğruluğu IBM composer kullanılarak ve kuditler için olanlar da manuel olarak test edilmiştir.

**Anahtar Sözcükler:** Kuantum Bilgi İşleme, Kübit, Kudit, Kuantum Fourier Transform, Kuantum Aritmetik İşlem, Kuantum Aritmetik Mantık Birimi

# ABSTRACT

## DEVELOPMENT AND APPLICATIONS OF SOME QUANTUM ARITHMETIC OPERATIONS BASED ON QUANTUM FOURIER TRANSFORM

Murat KURT

Ondokuz Mayıs University  
Institute of Graduate Studies

Department of Physics

Ph.D., February 2024

Supervisor: Prof. Dr. Azmi GENÇTEN

Second Supervisor : Assoc. Prof. Dr. Selçuk ÇAKMAK

Quantum information processing is superior to classical information processing because it allows information to be processed in parallel. Various quantum algorithms have been developed for quantum information processing. Some of them are known as QFT-based quantum algorithms. The transfer of information encoded in classical bits into a superposition state in quantum computation is achieved through the QFT algorithm. This study is related to the development and applications of quantum circuits performing QFT-based quantum arithmetic operations. For the first time in this study, a quantum circuit for QFT-based addition of  $n$ -bit binary numbers greater than two is proposed. In this suggestion, the reduction in the number of required gates is demonstrated by not using repeating QFT-IQFT in each addition operation. It is shown that the reduction in the number of gates is even more significant when operations are performed with qudit systems. A quantum circuit design is proposed for QFT-based subtraction operations that can result in both positive and negative outcomes, and its functionality is demonstrated through examples. For negative results, a quantum circuit that performs the comparison required for subtraction operations has been designed. A quantum circuit performing multiplication operations is developed by using the QFT-based addition circuit structure. The circuits that realize all possible products between single qutrit and single ququart states are constructed by using Weyl operators. For division operations, a quantum circuit incorporating a self-repeating QFT-based subtraction operation is designed. Since information about the number of subtraction operations is essential in this circuit, a quantum counter design is proposed. Additionally, a countdown version of this counter is suggested and generalized for any  $d$ -level quantum system. The NAND logic gate in classical information processing is designed based on QFT. Finally, a QFT-based simple quantum Arithmetic Logic Unit (qALU) design performing addition and NAND operations on a single circuit by using selective inputs is presented for the first time in this study. The accuracy of all quantum circuits developed in this study for qubits was tested using IBM Composer, and those for qudits were tested manually.

**Keywords:** Quantum Information Processing, Qubit, Qudit, Quantum Fourier Transform, Quantum Arithmetic Operation, Quantum Arithmetic Logic Unit

# ÖNSÖZ VE TEŞEKKÜR

Bu çalışmanın fikren ortaya çıkması, oluşması ve tamamlanması sürecinde, mesai kavramının dışına çıkarak, akademik ve pedagojik desteğini hiçbir zaman esirgemeyen tez danışmanım Prof. Dr. Azmi GENÇTEN'e, yine akademik ve teknik destekleri ile çalışmaya canlılık katan, bu çalışmadaki ana fikrin doğru değerlendirmesini yaparak, bilimsel bir proje tarafından desteklenmesinin yolunu açan ikinci danışmanım Doç. Dr. Selçuk ÇAKMAK'a sonsuz teşekkürlerimi sunarım.

Tez İzleme Komitesinde bulunan ve tezin ilerlemesinde değerli öneri ve yorumları ile katkıda bulunan kıymetli hocalarım Prof. Dr. Emin ÖZTEKİN ve Prof. Dr. Sedat AKLEYLEK'e gönülden teşekkürlerimi sunarım.

Doktora süreci boyunca hem maddi hem de manevi desteğini esirgemeyen, sabırla süreci takip eden, en önemlisi bu uzun yolculuk sırasında hayatımıza isimleri İsmail ve Kerem olan, aile galaksimizin en parlak iki yıldızı olan oğullarımızı dahil ettiği için çok kıymetli eşime sonsuz teşekkürlerimi sunarım.

Son olarak akademik anlamda zihnimde ve kalbimde ilk akademik kıvılcımın oluşmasını sağlayan Prof. Dr. Ali HAVARE'ye sonsuz teşekkürlerimi sunarım.

Murat KURT

# İÇİNDEKİLER

TEZ KABUL VE ONAYI	i
BİLİMSEL ETİĞE UYGUNLUK BEYANI	ii
ÖZET	iii
ABSTRACT	iv
ÖNSÖZ ve TEŞEKKÜR	v
ŞEKİLLER DİZİNİ	viii
TABLolar DİZİNİ	xi
<b>1 GİRİŞ</b>	<b>1</b>
1.1 Giriş . . . . .	1
1.2 İkinci Kuantum Devrimi . . . . .	3
1.3 Literatür Özeti ve Tezin Amacı . . . . .	6
<b>2 KURAMSAL TEMELLER</b>	<b>9</b>
2.1 Kuantum Mekaniği Postülatları . . . . .	9
2.2 Kuantum Bilgi İşleme . . . . .	11
2.2.1 Kübitler, Bloch Küresi ve Tek Kübitlik Kapılar . . . . .	11
2.2.2 İki ve Daha Fazla Kübit İçin Kuantum Kapılar . . . . .	17
2.2.3 Küditler ve Bazı Genelleştirilmiş Kuantum Kapılar . . . . .	23
2.3 Kuantum Devreler . . . . .	31
2.3.1 Dolanıklık Devresi . . . . .	31
2.3.2 Tam ve Yarı Toplayıcı Devreler . . . . .	34
2.4 Kuantum Algoritmalar . . . . .	36
2.4.1 Kuantum Fourier Transform . . . . .	37
2.4.2 Kuantum Faz Tahmini . . . . .	39
2.5 QFT Tabanlı Aritmetik İşlemler . . . . .	40
<b>3 MATERYAL VE YÖNTEM</b>	<b>42</b>
3.1 Dört Kübitlik QFT Uygulaması . . . . .	42
3.2 İki Kukuart Girişli QFT Uygulaması . . . . .	43
3.3 Kuantum Hesaplama Simulatörleri . . . . .	44
3.4 IBM Kuantum Bilgisayarı . . . . .	46
3.5 IBM Simulator-mps Kullanılarak Dört Kübit Girişli Bir QFT Uygulaması . . . . .	48
3.6 Python İle İki Kukuart Girişli QFT Uygulaması . . . . .	49
3.7 Özel Durumlara Bağlı Bazı Kuantum Kapılar . . . . .	50
3.8 Bazı Özdeşlikler . . . . .	51
<b>4 BULGULAR VE TARTIŞMA</b>	<b>53</b>
4.1 QFT Tabanlı Seri Toplama Devresi ve Kübit-Küdit Kıyaslaması . . . . .	53

4.1.1	Kübit Kullanılarak Dört Adet İki Bitlik Sayının QFT Tabanlı Toplama İşlemi . . . . .	57
4.1.2	Kukuart ( $d = 4$ ) Kullanılarak Dört adet bir Bitlik Sayının QFT Tabanlı Toplama İşlemi . . . . .	59
4.2	QFT Tabanlı Çıkarma Devresi . . . . .	61
4.2.1	$a > b$ için $a - b$ İşlemini Gerçekleştiren QFT Tabanlı Çıkarma İşlemi . . . . .	62
4.2.2	$N$ Tane Sayı için Tek Bir Devrede Çıkarma İşlemi . . . . .	63
4.2.3	$a < b$ için $a - b$ İşlemini Gerçekleştiren QFT Tabanlı Çıkarma İşlemi . . . . .	65
4.2.4	QFT Tabanlı Toplama ve Çıkarma İşleminin Tek bir Devre Üzerinde Gerçekleştirilmesi . . . . .	70
4.3	KFD Tabanlı Seri Toplamaya Dayalı Çarpma Devresi . . . . .	70
4.4	Kürit ve Kukuart Sistemlerinin Çarpımları İçin Weyl Operatörlerinin Kullanılması . . . . .	73
4.5	Kuantum Sayaç . . . . .	76
4.6	KFD Tabanlı Çıkarma İşlemine Dayalı Bölme Devresi . . . . .	78
4.7	QFT Tabanlı NAND Kapısı . . . . .	80
4.8	QFT Tabanlı Aritmetik Mantık Birimi Tasarımı . . . . .	86
4.8.1	Bir Bitlik İki Sayının QFT Tabanlı Toplama ve NAND İşlemi	87
4.8.2	Bir Bitlik Dört Sayının QFT Tabanlı Toplama ve NAND İşlemi . . . . .	88
4.8.3	İki Bitlik İki Sayının QFT Tabanlı Toplama ve NAND İşlemi	90
<b>5</b>	<b>SONUÇ VE ÖNERİLER</b>	<b>93</b>
5.1	Sonuç . . . . .	93
5.2	Öneriler . . . . .	95
	<b>KAYNAKLAR</b>	<b>99</b>
	<b>EKLER</b>	<b>100</b>
	<b>ÖZGEÇMİŞ</b>	<b>114</b>

# ŞEKİLLER DİZİNİ

Şekil 2.1	Bloch küresi . . . . .	12
Şekil 2.2	Özdeşlik kapısı ve matris temsili . . . . .	13
Şekil 2.3	X kapısı ve durumlar üzerine etkisi . . . . .	14
Şekil 2.4	Y kapısı ve durumlar üzerine etkisi . . . . .	14
Şekil 2.5	Z kapısı ve durumlar üzerine etkisi . . . . .	14
Şekil 2.6	Hadamard kapısı ve durumlar üzerine etkisi . . . . .	15
Şekil 2.7	Faz kayması kapısı ve durumlar üzerine etkisi . . . . .	15
Şekil 2.8	$CNOT_a$ kapısının devre gösterimi . . . . .	19
Şekil 2.9	$CNOT_b$ kapısının devre gösterimi . . . . .	19
Şekil 2.10	Kontrol girişi $ 0\rangle$ olan CNOT kapısı . . . . .	20
Şekil 2.11	SWAP kapısı ve CNOT kapıları ile oluşturulan özdeşliği . . . . .	20
Şekil 2.12	Toffoli kapısı . . . . .	21
Şekil 2.13	Fredkin Kapısı . . . . .	21
Şekil 2.14	(a) Kontrol biti 1 (b) kontrol biti 0 olan $CP(\theta)$ kapıları . . . . .	22
Şekil 2.15	Çoklu kontrollü U kapısı . . . . .	23
Şekil 2.16	$H_3$ kapısının saf durumlara etkisi . . . . .	24
Şekil 2.17	$X_3$ kapısının kütritlerin saf durumlarına etkisi . . . . .	25
Şekil 2.18	$P_3(\theta_k)$ kapısının kütrit saf durumlara etkisi . . . . .	26
Şekil 2.19	Kütrit sistem için kontrollü $X_3$ kapıları . . . . .	26
Şekil 2.20	Kukuart sistem için Hadamard kapısı ve işlevi . . . . .	29
Şekil 2.21	Kukuart sistem için $X_4$ kapısının durumlar üzerine etkisi . . . . .	29
Şekil 2.22	Kukuart sistem için $P_4(\theta_k)$ kapısının durumlar üzerine etkisi . . . . .	30
Şekil 2.23	Kukuart sistem için kontrollü $X_4$ kapıları . . . . .	30
Şekil 2.24	İki girişli dolanıklık devresi . . . . .	32
Şekil 2.25	Yarı toplayıcı kuantum devre . . . . .	34
Şekil 2.26	Tam toplayıcı kuantum Devre . . . . .	35
Şekil 2.27	Genelleştirilmiş QFT devresi . . . . .	38
Şekil 2.28	d seviyeli ve n girişli QFT devresinin girdi ve çıktıları . . . . .	38
Şekil 2.29	d seviyeli ve t girişli QPE devresi . . . . .	40
Şekil 2.30	d seviyeli iki sayı için QFT tabanlı aritmetik işlem yapan genel devre . . . . .	40
Şekil 3.1	Dört kübit girişli QFT devresi . . . . .	42

Şekil 3.2	İki kukuart girişli QFT devresi . . . . .	43
Şekil 3.3	Bilgi işleme türleri . . . . .	46
Şekil 3.4	a) Transmon kübit b) Enerji seviyeleri (Alves, 2020) . . . . .	47
Şekil 3.5	IBM simulator-mps'de çizilen dört girişli QFT devresi . . . . .	48
Şekil 3.6	4 kubit girişli QFT devresinden elde edilen çıktılar . . . . .	48
Şekil 3.7	İki kukuart girişli QFT devresinden elde edilen çıktılar . . . . .	49
Şekil 3.8	Birinci kontrolü $ 1\rangle$ ikinci kontrolü $ 0\rangle$ olan CCNOT Kapısı .	50
Şekil 3.9	Kontrolleri sırasıyla $ 0\rangle$ ikinci kontrolü $ 1\rangle$ ve $ 0\rangle$ olan CCC- NOT Kapısı . . . . .	51
Şekil 3.10	CNOT kapısının Hadamard ve kontrollü faz kayması kapıları ile oluşturulması . . . . .	51
Şekil 3.11	$ 0\rangle$ kontrollü CNOT ve toffoli kapılarının özdeşlikleri . . . . .	51
Şekil 4.1	N adet n bitlik sayının QFT tabanlı seri toplayıcı devresi . . . . .	55
Şekil 4.2	Toplayıcı kısım . . . . .	56
Şekil 4.3	Dört adet iki bitlik sayının, $d=2$ için QFT tabanlı toplama devresi . . . . .	58
Şekil 4.4	Dört adet iki bitlik sayının $d = 2$ için QFT tabanlı toplama işleminin sonucunun <i>ibm - mps</i> 'den elde edilen sütun grafiği	59
Şekil 4.5	Dört adet bir bitlik sayının $d = 4$ için QFT tabanlı toplama devresi . . . . .	60
Şekil 4.6	Şekil 4.5'deki devrenin işleyişi . . . . .	61
Şekil 4.7	Kübit ve kukuartlar için N ye bağlı kapı sayısı değişimi . . . . .	62
Şekil 4.8	QFT tabanlı çıkarıcı devre . . . . .	63
Şekil 4.9	Çıkarıcı parça . . . . .	63
Şekil 4.10	QFT tabanlı seri çıkarıcı devre . . . . .	64
Şekil 4.11	$t+n$ bitlik genel çıkarıcı . . . . .	64
Şekil 4.12	Bir bitlik karşılaştırıcı devre . . . . .	65
Şekil 4.13	$ a\rangle =  1\rangle$ ve $ b\rangle =  0\rangle$ karşılaştırılması için IBM composer'de çizilmiş kuantum devre . . . . .	66
Şekil 4.14	$ a\rangle =  1\rangle$ ve $ b\rangle =  0\rangle$ karşılaştırılma devresinin <i>ibm- nairobi</i> 'den elde edilen sonuc grafiği . . . . .	66
Şekil 4.15	İki bitlik karşılaştırıcı devre . . . . .	68
Şekil 4.16	Genelleştirilmiş karşılaştırmalı çıkarıcı kuantum devresi . . . . .	69
Şekil 4.17	Ardışık toplayıcı ve çıkarıcı kuantum devresi . . . . .	70
Şekil 4.18	Üç bitlik iki sayının manuel çarpım yöntemi . . . . .	71
Şekil 4.19	3 Bitlik iki sayının çarpımını gerçekleştiren kuantum devre .	72

Şekil 4.20	Genelleştirilmiş çarpım devresi . . . . .	73
Şekil 4.21	Kütrit sistemlerde olası tüm çarpımları gerçekleştiren temel kuantum devresi . . . . .	74
Şekil 4.22	Kukuart sistemlerde olası tüm çarpımları gerçekleştiren temel kuantum devre . . . . .	75
Şekil 4.23	Genelleştirilmiş $n$ girişli ileri sayaç devresi . . . . .	76
Şekil 4.24	Genelleştirilmiş $n$ girişli geri sayaç devresi . . . . .	76
Şekil 4.25	3 kübitlik (a) ileri ve (b) geri sayım yapan kuantum devreleri ve işlevsel diyagramları. . . . .	77
Şekil 4.26	3 kütritlik (a) ileri ve (b) geri sayım yapan kuantum devreleri ve işlevsel diyagramları. . . . .	77
Şekil 4.27	3 kukuartlık (a) ileri ve (b) geri sayım yapan kuantum devreleri ve işlevsel diyagramları. . . . .	77
Şekil 4.28	Girişi $ 101\rangle$ olarak ayarlanmış 3 girişli kubit tabanlı IBM'in gerçek kuantum bilgisayarı için arayüzde çizilmiş devre. . . .	78
Şekil 4.29	$ 101\rangle$ girişli ileri sayaç devresinin çıktılarını gösteren sütun grafiği . . . . .	78
Şekil 4.30	QFT tabanlı kuantum bölme devresi . . . . .	79
Şekil 4.31	QFT tabanlı $n$ girişli NAND kapısı . . . . .	81
Şekil 4.32	QFT tabanlı 2 girişli NAND kapısı devresi . . . . .	81
Şekil 4.33	QFT tabanlı 2 girişli NAND kapısı devresinin çıktısı . . . . .	82
Şekil 4.34	QFT tabanlı 3 girişli NAND kapısı devresi . . . . .	82
Şekil 4.35	QFT tabanlı 3 girişli NAND kapısı devresinin çıktısı . . . . .	83
Şekil 4.36	Belirli bir girdiye uygulanan genelleştirilmiş QFT tabanlı NAND kapısı devresi . . . . .	84
Şekil 4.37	2 giriş için QFT tabanlı NAND devresi . . . . .	85
Şekil 4.38	3 giriş için QFT tabanlı NAND devresi . . . . .	85
Şekil 4.39	NAND(10) için simülasyon sonucu . . . . .	86
Şekil 4.40	NAND(110) için simülasyon sonucu . . . . .	87
Şekil 4.41	Bir bitlik 2 giriş için qALU . . . . .	88
Şekil 4.42	Bir bitlik 4 girişli qALU . . . . .	89
Şekil 4.43	2 bitlik 2 girişli qALU . . . . .	92

# TABLolar DİZİNİ

Tablo 2.1	Kütritler ( $d = 3$ ) için hesaplama bazları ve matris temsilleri	24
Tablo 2.2	Kukuartlar ( $d = 4$ ) için hesaplama bazları ve matris temsilleri	28
Tablo 2.3	Yarı toplayıcı devre için doğruluk tablosu . . . . .	35
Tablo 3.1	$ 1\rangle$ ve $ 0\rangle$ Kontrollü NOT kapısı . . . . .	50
Tablo 4.1	Bir bitlik karşılaştırıcı devre için olası girdiler ve çıktılar . .	65
Tablo 4.2	Kübitlerin olası tüm çarpımları ve sonuçları . . . . .	71
Tablo 4.3	Küritlerin olası tüm çarpımları ve sonuçları . . . . .	74
Tablo 4.4	Kukuartların olası tüm çarpımları ve sonuçları . . . . .	75
Tablo 4.5	Qiskit kullanılarak test edilen bölme işlemleri . . . . .	80
Tablo 4.6	IBM simulator-mps ile elde edilen 2 girişli QFT tabanlı NAND kapısı sonuçları . . . . .	82
Tablo 4.7	IBM simulator-mps ile elde edilen 3 girişli QFT tabanlı NAND kapısı sonuçları . . . . .	83
Tablo 4.8	Bir bitlik iki sayı için qALU . . . . .	88
Tablo 4.9	Bir bitlik dört sayı için qALU . . . . .	90
Tablo 4.10	2 bitlik 2 sayı için bazı durumlar qALU . . . . .	91

# 1. GİRİŞ

## 1.1. Giriş

İçinde bulunduğumuz çağ bilginin dijitalleşme çağı olarak adlandırılabilir. Bilginin üretilmesi ve üretilen bilginin sanayi, teknoloji, sağlık, ekonomi, askeri ve eğitim gibi alanlarda sisteme dâhil edilmesi, işlenmesi, depolanması ve güvenli bir şekilde aktarılması çağımızda gelişmişlik seviyesi için en önemli ölçüt olarak düşünülebilir. Bunu başarabilen ülke veya ülkelerin dünya üzerinde güçlü, söz sahibi ve yönlendirici olacağını öngörmek çok zor değildir. Bilginin üretiminden, işlenip kullanımına ve dağıtımına kadar olan tüm süreçler temel bilimlerin gelişimi ile paralellik göstermektedir. Temelleri bir asır öncesinde atılan kuantum fiziği bizlere yeni bir madde ve enerji anlayışı sunmuştur. Burada ilginç ve güzel olan şudur ki; yaşadığımız dünya ile ilgili bilgi edinme çabası, aynı zamanda var olan bilginin işlenmesi, depolanması ve iletilmesi gibi fiziksel süreçlerin kabuk değiştirmesine de sebep olmuştur. Bugün hayatımızın her alanında bizimle birlikte olan bilgisayarlar ve cep telefonları gibi içinde mikro işlemci barındıran her teknolojik ürün eskinin yerini alan bu yeni kabuğun ta kendisidir.

Bilginin dijitalleşmesinden yola çıkarak günümüz gelişmelerini anlayabilmek için yolculuğumuza 1854 yılından başlayabiliriz. George Boole yayınladığı Düşünce Yasaları adlı kitabında mantık ile cebiri birleştirerek; doğru kanıt içeren cümleleri 1, yanlış kanıt içeren cümleleri 0 rakamı ile temsil etti. Bu temsilleri cebirsel bir zemine oturtarak, bugün Boole Cebri olarak bildiğimiz mantığın cebirini ortaya koydu. 1937'de Hesaplanabilir Sayılar Üzerine, Karar Probleminin bir Uygulaması başlıklı çalışması ile Alan Mathison Turing, Boole cebirinin kullanarak programlama temelli bir makine tasarlamış ve algoritma kavramının temelini atmıştır (Turing, 1937). Hem bir matematikçi hem de bir elektrik mühendisi olan Claude Elwood Shannon Röle ve Anahtarlama Devrelerinin Sembolik Analizi isimli yüksek lisans tezi ile günümüz bilgisayarlarının yapı taşı olan elektrik anahtarlarının (açma-kapama anahtarları) kullanımının temelini atmıştır. Bu çalışma ile 1854 yılında Boole'un önerisi olan 1 ve 0 temsilleri fiziksel olarak gerçekleştirilebilmiştir. Aynı yıl Harward Üniversitesinde doktora öğrencisi olan

Howard Aiken, doktora tezi için gerekli olan zorlu ve uzun süren hesaplamaları bir toplama makinesi kullanarak yapmaya çalışıyordu. Hesaplamalar uzun sürdüğü için Harward'daki üslerine ve IBM yöneticilerine, 1822 yılında Babbage tarafından yapılan dijital makinenin modern versiyonunun yapılmasının gerektiğine dair etkili bir yazı yazdı. Bunun üzerine, 7 yıl süren zorlu bir süreçten sonra IBM 1944 yılında Mark-1 isimli makineyi tamamladı. Makine dijitaldi ama Boole'un önerdiği gibi 0 ve 1'ler ile işlem yapmıyordu. 15 metrelik mili üzerinde 23 basamağa kadar sayıların kaydedildiği 72 sayaç vardı. 5 tonluk makine 24 metre uzunluğunda 15 metre genişliğindeydi. Elektromanyetik röleler yerine elektrik motoru ile açılıp kapanan mekanik röleler vardı. Bir çarpma işlemini 6 saniyede yapıyor, program ve veriler kâğıt şeritle giriliyordu. Mark 1, enerji ihtiyacı karşılandığı sürece, insan müdahalesi olmadan günlerce çalışabiliyordu.

Yüksek lisans tezini yazdığı 1937 yılından 11 yıl sonra Claude Elwood Shannon İletişimin Matematiksel Teorisi başlıklı makalesinde 0 ve 1'ler kullanılarak temsil edilecek bilginin kaç bit olacağını yani bilginin boyutunu entropi yasaları ile açıkladı (Shannon, 1948).  $P$ , gerçekleşecek argüman ya da olayın gerçekleşme olasılığı olmak üzere, bilginin bit sayısını  $\log_2(1/P)$  ifadesi ile bulunabileceğini söyledi. Örneğin bir yazı-tura oyununda yazı veya tura gelme olasılığı  $\frac{1}{2}$  dir.  $P$  yerine  $\frac{1}{2}$  yazıldığında sonuç 1'dir. Bu bilginin 1 bitlik bir boyuta sahip olduğu anlamına gelir. 1941 yılında Amerika'nın 2. Dünya Savaşına girmesi ile Mauchly ve Eckert tasarladıkları hesaplama makinesi için fon bulma fırsatı yakaladı. Bu makine açma-kapama işlevini elektrik motorları yerine vakum tüpleri ile gerçekleştiriyordu. Alman şifrelerini çözmek ve balistik hesap yapmak gibi amaçları vardı. Bu makinenin en büyük avantajı istenildiğinde ulaşılacak bir belleğe sahip olmasıydı. Saniyede yaklaşık 5000 adet toplama çıkarma işlemi yapması bunun Mark 1'den daha hızlı olduğunu gösteriyordu. Mauchly ve Eckert tarafından tasarlanan bu makine ENIAC (Electronic Numerical Integrator and Computer) olarak adlandırıldı. 17500 vakum tüpüne sahip, 30 ton ağırlığında ve 170  $m^2$  alan kaplıyordu.

1947 yılına gelindiğinde günümüz bilgisayarlarının inşasında kullanılan çok önemli bir keşif gerçekleşti. Bell laboratuvarında Shockley, Bardeen ve Brattain vakum tüpleri yerine kullanılacak transistörü keşfettiler. Keşiflerinin altında yatan

dinamik ise yarı iletken fiziğindeki bilgi birikimi oldu. İlk transistörün keşfinden sonra yapılan çalışmalar yoğunluk kazandı. Yıldan yıla bilgisayarlardaki mikroişlemcilerdeki transistör sayısı arttı. 1971 yılında bir mikroişlemcideki transistör sayısı 2300 iken, 2012 yılına gelindiğinde bu sayı beş milyar düzeyine ulaştı. Bu artış transistörlerin gittikçe küçülmesinin bir sonucudur. Intel'in kurucularından Gordon Moore, 1965 yılında yayınladığı makalesinde mikroişlemcilerdeki transistör sayısının ortalama 18 ayda bir ikiye katlanacağını öngördü (Moore, 1965). Bu öngörü transistörlerin giderek atomik boyutlara ulaşacağını ve bu durumda da bilginin işleme sürecinde kuantum fiziği yasalarının devreye gireceğinin habercisi oldu. Moore'a göre 2020 yılına gelindiğinde transistör sayısındaki artış sonlanacaktı. Günümüzde, bilgisayar şirketleri mikroişlemciye daha fazla transistör sığdıramamanın zorluğunu, çok çekirdekli işlemciler alternatifi ile aşmaya çalışıyor.

## 1.2. İkinci Kuantum Devrimi

Birinci kuantum devriminin, 1900'lü yıllarda Max Planck tarafından ortaya konan enerjinin kuantumlu yapıda olması kabulü ile başladığı kabul edilir. Planck bu önerisi ile atomik yapıdaki ışın enerjisinin elektronların belirli enerji seviyelerine sahip olmasından kaynaklandığı söyler. Bu süreç Heisenberg'in belirsizlik ilkesini ortaya atışını ve Schrödinger'in dalga denklemini çözmesini kapsar. 1926 yılına gelindiğinde Kuantum fiziğinin matematiksel bir açıklaması yapılmış olur. Kuantum kuramının matematiksel temellerinden elde edilen bilgilerle maddenin atomik yapısına yeni bir bakış getirilmiş ve bu sayede kuramın ürünü olan teknolojilere zemin hazırlanmıştır. Örneğin, transistörlerin icadı, NMR gibi spektroskopik yöntemlerin temelleri ve elektron mikroskobu gibi ürünler ve yöntemler bu matematiksel sonuçların teknolojik ürünleri olarak kendilerini göstermektedir.

İkinci kuantum devrimi ise daha çok bu matematiksel sonuçların nasıl yorumlandığı ile ilgilidir ve kesin sınır olmamakla beraber Einstein, Rosen ve Podolsky tarafından 1935 yılında kaleme alınan EPR makalesi ile başlar. Aslında bu makale kuantum mekaniğinin henüz tamamlanmadığını ve eksik bir teori olduğu ile ilgilidir. Bu çalışma her ne kadar kuantum mekaniğinin eksiklerini ortaya çıkarmak amacını taşısa da, sonrasında elde edilen her sonuç kuantum me-

kaniğinin barındırdığı kavramların daha güçlü olmasına sebep olmuştur.

Çalışmalarda 1. kuantum devriminde ortaya konan belirsizlik ve yerel olmama gibi kabulü zor ve sorun çıkarıcı kavramlar, ikinci kuantum devrimindeki ürünlerin gelişmesine neden olmuştur. Örneğin belirsizlik ilkesi sayesinde son derece güvenli bir bilgi iletiminin(kuantum kriptografi), süperpozisyon özelliği sayesinde bilginin daha küçük birimlere daha çok sayıda kodlanarak işlenmesinin ve yerel olmama sayesinde parçacıkların dolanıklık özellikleri kullanılarak bilginin evrensel bir özellikte, mesafe gözetmeksizin anında iletilebileceği gösterilmiştir.

1959 yılında Altta Daha Çok Yer Var başlıklı çalışmasıyla bilgiyi atomik düzeyde işleme ve depolama konusuna değinen Richard Feynman, 1982 yılına gelindiğinde, doğadaki fenomenlerin kuantum fiziği yasaları ile açıklanabilmesi nedeniyle doğayı net bir şekilde simüle edebilmenin en iyi yolunun kuantum simülasyonları olduğunu söylemiştir (Feynman, 1960; Feynman, 1982). 1980 yılında Paul Benioff Turing makinelerinin kuantum versiyonunu önermiştir (Benioff, 1980). Benioff, Boole'nin bilgi temsili için önerdiği 0 ve 1'i atom altı parçacıkların spin durumları ile temsil edebileceğimize dikkat çekti. Benioff'un çalışması kuantum bilgi işleme fikri açısından çok önemlidir. Bu çalışma hak ettiği ilgiyi göremese de, işaret ettiği kapıdan geçişler gerçekleşmeye başladı. Fikri temelleri atılmış olan kuantum bilgi işleme süreçlerinin, gerçekleştirilebileceği yönündeki en önemli adımın atılması için 1985 ve 1988 yılını beklemek gerekecekti. Deutsch 1985 yılında Kuantum Teorisi, Church-Turing Prensipleri ve Evrensel Kuantum Bilgisayarı başlıklı çalışmasından 4 yıl sonra Kuantum Hesaplama Ağları başlıklı makalesi ile kuantum bilgi işleme süreçleri için geçerli olan kuantum mantık kapılarının teorisini ortaya attı (Deutsch, 1985; Deutsch, 1989). Kapılar teorik olarak önerilmesine karşın bu gelişme kuantum bilgisayarlarının gerçekleşme olasılığını arttırdı. Bernstein ve Vazirani, 1993 yılında kuantum devrelerindeki karmaşıklık teorisine katkı sağladılar (Bernstein ve Vazirani, 1993). 1994 yılında Peter Shor, çok uluslu şirketlerin güvenliğini sağlayan RSA kod sisteminin güvenliğini tehdit eden bir algoritma ile sahneye çıktı (Shor, 1994). Büyük sayıları asal çarpanlarına ayırma algoritması olarak bilinen bu algoritma ile kuantum bilgisayarlarının potansiyel gücü ortaya çıktı. Çünkü bu algoritma parçacıkların süperpozisyon ve dolanıklık özelliği kullanılarak gerçekleştiriyordu. Bugün bu durum kuantum üstünlüğü olarak

adlandırılmaktadır. Parçacıkların süperpozisyon durumunda bulunabilmeleri klasik bilgisayarda olmayan paralel işlemeyi beraberinde getiriyordu. Klasik bilgisayar kullanarak yıllarca sürebilecek çarpanlara ayırma işlemi Shor'un algoritması ile dakikalar mertebesine iniyordu. Bu sonuç bilgisayar bilimcilerinin bu yönde ilgisini çekmeye başladı. 1997 yılında Grover, veri tabanlarında arama yapmak için bir kuantum arama algoritması önerdi (Grover, 1997). Önerilen bu algoritma ile  $N$  elemanlı bir veri listesinden, ortalama  $\sqrt{N}$  tekrarda istenilen bilgi bulunabilecekti. Bundan sonra farklı amaçlar için değişik algoritmalar geliştirilmiştir (Nielsen ve Chuang, 2010). Toparlayacak olursak 1980-1994 arasında kuantum bilgisayarları fikri ortaya çıkmıştır. 1994'den sonrada farklı kuantum algoritmalar bulunmuştur. 2000 yılında sonra kuantum bilgisayarların geliştirilmesi yarışı başlamıştır. Bu yarışta farklı fiziksel prensiplerle çalışan kuantum bilgisayarlar geliştirilmiştir.

Kuantum bilgi işlemede temel algoritmalar olarak kabul edilen Shor ve Grover algoritmaları bugün hala bu alan için önemlerini korumaktadırlar. 1990'ların sonlarına yaklaşırken kuantum bilgi işlemeyi gerçekleştirecek yazılımlar ile ilgili bir hayli yol alındı. Bu aşamadan sonra yaklaşmakta olan problem, bu yazılımın gerçekleştirilebileceği donanımın nasıl oluşturulacağıydı. 100 yaşına yaklaşmakta olan kuantum fiziği atom ve atomaltı parçacıklar ile ilgili aşılamayan bir kesinlik bilgisi ortaya koymuştu. Bilginin kodlanacağı kuantum durumlarının diğer parçacıklar ve çevresel şartlardan etkilendiği biliniyordu. Dolayısıyla bir atom altı parçacığın müdahale edilebilir olması ve müdahale sonrası durumunu koruyamaması donanım kısmının oluşturulabilmesinin önünde büyük bir engel teşkil ediyordu. Bu engel günümüzde dahi tam olarak aşılamadı. Diğer parçacıklarla etkileşmeyen, ortam sıcaklığı ve basıncı gibi çevresel şartlardan etkilenmeyen, ölçüm sırasında problem çıkarmayan yalıtık sistemler elde edememe sorunu hala devam etmektedir. Bu şartları sağlayabilecek durumları oluşturmak her ne kadar meşakkatli ve maliyetli olsa da günümüzde bununla ilgili hem teorik hem de deneysel çalışmalar tüm hızıyla devam etmektedir.

1997 yılında kuantum bilgi işleme süreçlerinde NMR prensiplerinin kullanılabilceği Gershenfeld ve Chuang tarafından gösterilmiştir ve iki kübit üzerine etki edecek kuantum mantık kapılarının oluşturulmasında NMR prensipleri Price

ve arkadaşları tarafından çalışılmıştır (Gershenfeld ve Chuang, 1997; Price, vd., 1999). 2000 yılında David P. DiVincenzo “ Kuantum Hesaplamanın Fiziksel Uygulanması” başlıklı makalesinde bunun gerçekleşmesi için bazı kriterler ortaya koydu (DiVincenzo, 2000). DiVincenzo’ya göre bu kriterler aşağıdaki gibidir.

1. Kuantum hesaplamannın net olarak yapılabilmesi için kubitlerin iyi tanımlanmış olması gerekir.
2. Hesaplamadan önce, sistemin ilk durumu belirlenirken, kubitlerin  $|00\dots0\rangle$  gibi bir taban durumdan başlaması gerekir.
3. Sistemdeki etkileşimlerin gözlenebilmesi için en azından kubitlerin, çalışma süresinden daha uzun olan uzun durulma zamanına sahip olması gerekir.
4. Evrensel kuantum kapılarına sahip olması gerekir.
5. Kubitler üzerinde ölçme yeteneği olmalıdır ve istenildiğinde her bir kubit için ayrı ayrı sonuç bulunmalıdır.

Bu kriterlerler bilginin kodlanacağı kubitin kendisinden sistemin uyum ve izolasyonuna kadar olması gerekenleri söyler. 2001 yılında IBM, 7 kubitlik bir kuantum bilgisayar geliştirerek, Peter Shor’un algoritmasını uyguladı. Günümüzde IBM, Google gibi şirketler kuantum bilgisayarlarının mimarisi için bütçelerinden büyük paylar ayırmaktadırlar. IBM Q, D-Wave, Google gibi şirketler projelerle çalışmalarını kamuoyuna duyurmaktadırlar. Tüm bu gelişmeler gösteriyor ki, 1940’lı yıllarda klasik bilgisayarların geçtiği süreçleri bugün kuantum bilgisayarları yaşıyor. Kuantum süreçlerinin hassasiyetleri bir sorun olarak sürecin hızlanmasını yavaşlatıyorsa da çalışmalar hızla devam ediyor.

### **1.3. Literatür Özeti ve Tezin Amacı**

Bu tezin konusu QFT tabanlı toplama, çıkarma, çarpma ve bölme işlemlerini gerçekleştiren kuantum devrelerinin yeniden tasarlanması ve herhangi bir d seviyeli kuantum sistemi için genellemesinin yapılmasıdır. Bu konu ile ilgili yapılan çalışmalar 1996 yılına kadar uzanmaktadır. Vedral ve arkadaşları tarafından yapılan Quantum Network for Elementary Arithmetic Operation başlıklı

çalışmada kubit tabanlı ve QFT kullanılmadan toplama ve çarpma işlemi gerçekleştiren devre tasarımları yapılmıştır (Vedral, 1996). Ardından 1998 yılında Gosset ve 2004 yılında da Cuccaro ve arkadaşları tarafından yine kubit tabanlı ve QFT kullanılmadan toplama ve çarpma gibi aritmetik işlemleri gerçekleştiren kuantum devrelerinin tasarımları yapılmıştır (Gosset, 1998; Cuccaro, 2004).

2000 yılında Thomas Draper tarafından yayınlanan Addition on a Quantum Computer başlıklı makalede ilk kez QFT tabanlı toplama işlemi gerçekleştiren kuantum devresinin tasarımı yapılmıştır (Draper, 2000). Draper'in bu çalışması QFT tabanlı olması sebebiyle kendinden önceki çalışmalardan ayrıdır. 2004 yılında Floria ve Picca tarafından QFT tabanlı ve kubitler için çalışabilen toplama ve çarpma gibi aritmetik işlemleri gerçekleştiren çalışma yayınlanmıştır (Floria and Picca, 2004). Yaklaşık 13 yıllık bir sessizliğin ardından Perez ve Escartin tarafından 2017 yılında yine QFT tabanlı ve kubit temelli bazı aritmetik işlemleri gerçekleştiren kuantum devrelerinin önerisi yapılmıştır (Perez ve Escartin, 2017). Pavlidis yayınladığı çalışmada, aritmetik işlemleri gerçekleştiren devrelerde kullanılan kuantum kapılarının kuditler için matris temsillerini veren matematiksel eşitlikleri kullanarak hem QFT tabanlı hemde kuditler için çalışan ve toplama, çarpma ve kare alma gibi işlemleri gerçekleştiren devrelerin tasarımlarını yapmıştır (Pavlidis, 2020). 2020 yılında Engin Şahin tarafından QFT tabanlı temel aritmetik işlemler olmak üzere, sayının mutlak değeri ve iki sayının karşılaştırmasını yapan alt devrelerin tasarımları da literatüre kazandırılmıştır (Şahin, 2020). Ayrıca bu çalışmanın ürünlerinden biri olan qALU tasarımının fikri ilk kez Florio ve Picca'nın çalışmalarında geçmektedir.

Bu çalışmanın 2. bölümünde Kuantum Bilgi İşlemenin kuramsal temelleri verilecektir. Bunun için öncelikle kuantum mekaniği postülatlarına değinilecek ardından kubitlerin tanımı yapılacak ve tek kubit üzerinde işlem yapan kuantum kapıları tanıtılacaktır. Daha sonra 2 ve daha fazla kubit üzerinde işlem yapabilen kuantum kapılarına değinilecektir. Yine bu bölümde kuditlerin tanımı yapıldıktan sonra kuditler üzerinde işlem yapabilen bazı kuantum kapılarının matematiksel eşitlikleri, matris temsilleri ve etki ettiği kuditleri dönüştürdüğü yeni durumlar üzerinde durulacaktır. Bu bölümün devamında kuantum devrelerinin nasıl işlediği üzerinde durulacak, dolanıklık, yarı ve tam toplama işlemlerini gerçekleştiren ku-

antum devreleri tanıtılacaktır. Ardından kuantum algoritmalara geçilecek ve kuantum bilgi işlemenin temel algoritmaları olarak kabul edilen QFT ve kuantum faz tahmini algoritmalarına değinilecektir.

3. Bölümde bu tez çalışmasında kullanılan materyal ve yöntemler hakkında bilgi verilecektir. Düşük sayıda girişe sahip durumların QFT devreleri çizilecek ve sonuçları değerlendirilecektir. Çalışmalar sırasında kullandığımız Python ve alt kütüphanesi Qiskit hakkında bilgi verilecek, önerilen devreleri test ettiğimiz IBM kuantum bilgisayarı tanıtılacaktır. Bölümün ilerleyen kısımlarında ihtiyaç duyduğumuz yeni kapılar ve bazı özdeşlikler tanıtılacaktır.

4. Bölüm olan Bulgular ve Tartışma kısmında, ilk defa bu çalışmada QFT tabanlı toplama için 2'den fazla sayı üzerinde işlem yapabilen ve herhangi bir  $d$  seviyesi için çalışabilen kuantum devresi önerilmiştir. Bu devrenin sadece 2 sayının toplamını yapan devrelere göre avantajı tartışılacaktır. Benzer yöntemler kullanılarak QFT tabanlı çıkarma işlemi yapan kuantum devre önerilecektir. Bu bölümün ilk kısmında önerilen 2'den fazla sayının toplama işlemini tek bir devrede gerçekleştiren devre kullanılarak çarpma işlemini gerçekleştiren kuantum devresi verilecektir. Kuantum karşılaştırıcı ve kuantum sayaç devrelerinin tanımı yapılacak ve simülasyonları hem qiskit kullanılarak hemde IBM composer kullanılarak gerçekleştirilecektir. QFT tabanlı çıkarma işlemine dayalı, karşılaştırıcı ve sayaç devreleri kullanarak tekrarlama adımları içeren bölme işlemini gerçekleştiren kuantum devre sunulacaktır. Son olarak klasik hesaplamada kullanılan mantık kapılarından NAND kapısının QFT tabanlı tasarımı yapılacak ve önemi üzerinde durulacaktır. Ardından temel bir Q-ALU tasarımı üzerinde durulacaktır. Girişler değişmeden, toplama ve NAND gibi işlemleri tek bir devre üzerinde gerçekleştiren qALU tasarımı verilecek ve Qiskit yardımı ile çalışır olduğu gösterilecektir.

## 2. KURAMSAL TEMELLER

### 2.1. Kuantum Mekaniği Postülatları

Kuantum bilgi işleme, kuantum mekaniği prensipleri çerçevesinde bilgiyi işler, depolar ve iletir. Bu nedenle kuantum mekaniğinin postulatlarını anlamak önemlidir. Kuantum mekaniğinin postülatları 5 madde ile açıklanabilir.

1. Kuantum mekaniksel sistemlerin durumu, konuma ve zamana bağlı  $\psi(r, t)$  fonksiyonu ile temsil edilir. Bu fonksiyona dalga veya durum fonksiyonu denir. Kuantum mekaniksel sistemin durumunu tanımlayan dalga fonksiyonunun, parçacığın uzayın herhangi bir yerinde bulunma olasılığını belirleyen şartı sağlamalıdır. Bu normalizasyon şartıdır.

$$\int_{-\infty}^{+\infty} \psi(r, t)^* \psi(r, t) d\tau = 1 \quad (2.1)$$

Ayrıca  $\psi(r, t)$  dalga fonksiyonu  $|\psi\rangle$  ile de temsil edilir.  $|\psi_1\rangle$  ve  $|\psi_2\rangle$  bir kuantum mekaniksel sistemin olası iki durumu olmak üzere; Bu iki durumun lineer kombinasyonu da parçacığın bir durumunu temsil eder.

$$|\psi\rangle = a |\psi_1\rangle + b |\psi_2\rangle \quad (2.2)$$

Buna süperpozisyon durumu denir.

2. Konum, momentum, kinetik enerji, potansiyel enerji ve açısal momentum gibi gözlenebilir her özellik, kuantum mekaniğinde sistemi tanımlayan  $|\psi\rangle$ 'ler üzerine etki eden matematiksel operatörler ile tanımlanır.  $\hat{H}$  herhangi bir kuantum mekaniksel operatörü temsil etmek üzere, bu operatörün durum üzerine etkisinin matematiksel temsili  $\hat{H} |\psi\rangle$  biçiminde yazılır. Kuantum mekaniğindeki bu operatörler matrisler ile de temsil edilirler. Bu matrisler hermityen olmalıdır. Hermityen bir matrisin kompleks eşleşliğinin transpozu o matrisin tersine eşittir.

$$(H^*)^T = H^\dagger = H^{-1} \quad (2.3)$$

Bu eşitlik bize  $(H^*)^T H = I$  olduğunu söyler. Burada  $I$  birim matristir.

3. Her operatörün herhangi bir kuantum durumu için değişmeyen sonuçları vardır.

$$\hat{H} |\psi_a\rangle = a |\psi_a\rangle \quad (2.4)$$

Burada  $\hat{H}$  operatörü  $|\psi_a\rangle$  öz durumu için  $a$  özdeğerine sahiptir. Örneğin spin sayısı  $1/2$  olan elektronun  $+1/2$  ve  $-1/2$  olmak üzere iki öz durumu vardır.  $\hat{S}_z$  z yönündeki spin operatörü olmak üzere;

$$\hat{S}_z |S, m_s\rangle = \hbar m_s |S, m_s\rangle \quad (2.5)$$

eşitliği tanımlanmıştır.  $\hat{S}_z$  operatörü  $+1/2$  ve  $-1/2$  öz durumlarına uygulanığında,

$$\hat{S}_z |1/2, +1/2\rangle = \hbar/2 |1/2, +1/2\rangle \quad (2.6)$$

ve

$$\hat{S}_z |1/2, -1/2\rangle = -\hbar/2 |1/2, -1/2\rangle \quad (2.7)$$

sonuçları elde edilir. Bunun anlamı  $1/2$  spin için iki öz durum vardır. Bu sonuç ayrıca kuantum durumlarının kuantize olduğunu da gösterir.

4. Herhangi bir  $\hat{H}$  gözlenebilirin beklenen değeri;

$$\langle \hat{H} \rangle = \int_{-\infty}^{+\infty} \psi(r, t)^* \hat{H} \psi(r, t) d\tau \quad (2.8)$$

eşitliği ile bulunur. Bu eşitlik ket notasyonu ile  $\langle \hat{H} \rangle = \langle \psi_i | \hat{H} | \psi_j \rangle$  biçiminde de yazılabilir.

5. Bir kuantum mekaniksel sistemin kinetik ve potansiyel enerji operatörleri ile varsa çevresel etkileşimlerden kaynaklanan enerjisini temsil eden operatörlerin toplamı Hamiltonyen olarak adlandırılır. Hamiltonyen operatörünün dalga fonksiyonuna etkisi ile dalga fonksiyonunun zamanla değişimi arasında bir ilişki vardır. Bu ilişki Schrödinger denklemi ile ifade edilir.

$$\hat{H} \psi(r, t) = i\hbar \frac{\partial \psi(r, t)}{\partial t} \quad (2.9)$$

## 2.2. Kuantum Bilgi İşleme

Kuantum bilgi işlemede, bilgi elektron, çekirdek gibi kuantum mekaniksel parçacıkların spin durumlarına ve fotonun sahip olduğu yatay ve düşey polarizasyon durumlarına kodlanır. Genel olarak bilginin kodlandığı spin durumlarına kuantum sistem denir. Kuantum sistemleri Hilbert uzayında bir durum vektörü olarak tanımlanır. Bir durum vektörü, söz konusu kuantum sisteminin fiziksel durumunu temsil eder ve  $|\psi\rangle$  notasyonu ile gösterilir. İlk kez iki seviyeli spin sistemi için kübit tanımı Benjamin Schumacher tarafından yapılmıştır (Schumacher, 1995). Sonraki çalışmalarda 3 seviyeli spin sistemleri için kütrit ve 4 seviyeli spin sistemleri için kukuart tanımlamaları yapılmıştır (Gedik, vd., 2015 ; Karakaş ve Gençten, 2018). Fakat genel olarak 3 ve daha fazla seviyeli spin sistemlere kudit denir (Wang, vd., 2020). Klasik bilgi işlemede bilginin en küçük birimi ya 0 ya da 1 sayıları ile temsil edilir ve bunlara bit (Binary Digit) denir. Kuantum bilgi işlemede bilginin en temel birimi olan kübit veya kudit, olası tüm durumların bir toplamı (süperpozisyon) olarak ifade edilir. Bilgi bu kübit ve kudit durumlarına kodlanarak işlenir.

### 2.2.1. Kübitler, Bloch Küresi ve Tek Kübitlik Kapılar

İki seviyeli spin sistemlerinde spin durumu  $+1/2$  ve  $-1/2$  değerlerini alırken fotonlarda ise yatay ve düşey polarizasyon durumlarını alır. Bu durumlar sırası ile  $|0\rangle$  ve  $|1\rangle$  notasyonu ile gösterilir. Bir kübit ise olası durumların bir süperpozisyonu olarak ifade edilir.

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad (2.10)$$

Burada  $\alpha$  ve  $\beta$  normalizasyon katsayılarıdır ve  $|\alpha|^2 + |\beta|^2 = 1$  dir. Eğer  $\alpha$  veya  $\beta$  sıfır ise bu saf durum olarak adlandırılır. Bununla birlikte bir kübit  $2 \times 1$  boyutlu bir sütun matrisi ile temsil edilir.

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad (2.11)$$

$$|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (2.12)$$

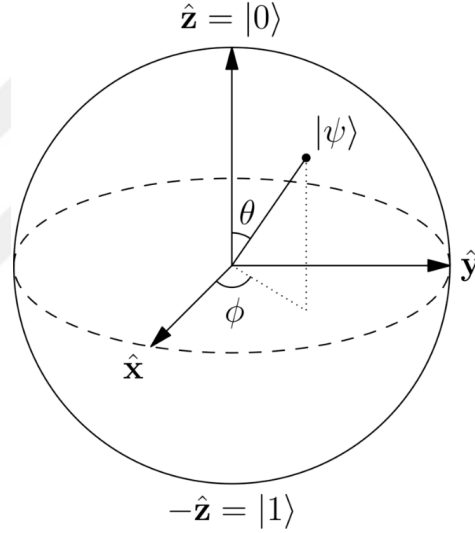
Eşitlik 2.11 ve 2.12, eşitlik 2.10'da yerine yazılırsa,

$$|\psi\rangle = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (2.13)$$

eşitliğine ulaşırız. Bir kubit aynı zamanda Şekil 2.1' de gösterilen Bloch Küresi üzerinde bir nokta ile de temsil edilebilir.  $0 \leq \theta \leq \pi$  ve  $0 \leq \phi \leq 2\pi$  olmak üzere, Bloch küresi üzerindeki bir nokta,

$$|\psi(\theta, \phi)\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle \quad (2.14)$$

denklemleri ile bulunabilir.



Şekil 2.1. Bloch küresi

Bloch küresi üzerinde tanımlanmış bir kubitte, kuantum mantık kapıları uygulanarak küre yüzeyinde tanımlı diğer kubitlere ulaşılabilir. Kuantum bilgi işlemede kullanılan kuantum mantık kapılarını temsil eden matrislerin üniter olmaları gerekir. Üniter olmayan bir kapı, bir kubit üzerine etkideğinde kubitte kodlanmış bilginin yok olmasına sebep olur. Kubitin taşıdığı bilginin kaybolmaması için ölçme işlemcisinin dışındaki kuantum kapıları üniter olmak zorundadır. Üniter matris, kompleks eşleniğinin transpozunu(devriği), söz konusu matrisin tersine eşit olan bir matristir.  $U$  bir matris olmak üzere,  $(U^*)^T = U^{-1}$  ise  $U$  matrisi üniter bir matristir. Ayrıca  $(U^*)^T = U^\dagger$  olduğundan,  $U^\dagger = U^{-1}$  eşitliği yazılabilir.

Kuantum kapılarını temsil eden bu matrislerin uniter özellikte olması aynı zamanda kuantum kapılarının tersinir olduğu anlamına gelir. Bir kubitte  $U$  kapısı uygulandıktan sonra oluşan yeni duruma  $U^\dagger$  kapısı uygulanırsa ilk duruma geri dönülür. Böylece kubitte kodlanmış bilgi kaybolmamış olur. Bu özellik klasik bilgi işlemede kullanılan mantık kapılarında bulunmamaktadır.

$$U^\dagger U = I \quad (2.15)$$

Burada  $I$  birim matrisi temsil eder. Eşitlik 2.15'ten yola çıkarak,

$$U^\dagger(U |\psi\rangle) = I \quad (2.16)$$

eşitliğini de yazabiliriz. Aşağıda tek kubit üzerine etkiyen kuantum kapıları, bu kapıların matris temsilleri ve işlevsellikleri verilmiştir.

Bu kapılardan ilki özdeşlik kapısıdır. Özdeşlik kapısı etki ettiği kubit üzerinde herhangi bir değişiklik meydana getirmez. Bu kapının devre gösterimi ve matris temsili Şekil 2.2'de görülmektedir.

$$|\psi\rangle \text{ --- } \boxed{I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}} \text{ --- } |\psi\rangle$$

Şekil 2.2. Özdeşlik kapısı ve matris temsili

Tek kubitlik kapılardan bir diğeri X (NOT) kapısıdır. Bu kapı Pauli X kapısı olarak da adlandırılır. Bu kapı,  $|0\rangle$  durumunu  $|1\rangle$  durumuna,  $|1\rangle$  durumunu da  $|0\rangle$  durumuna dönüştürür. Eğer girdi bir süperpozisyon durumu ise durumların katsayılarını değiştirir. Şekil 2.3'de X kapısının matris temsili ve işlevselliği görülmektedir. Benzer şekilde Y kapısının saf durumlara ve süperpozisyon durumuna etkisi aşağıdaki Şekil 2.4'de görülmektedir. Y kapısının saf durumlara ve süperpozisyon durumuna etkisi aşağıdaki Şekil 2.4'de görülmektedir.

Bir diğer kapı Z kapısıdır. Bu kapı  $|0\rangle$  durumu üzerinde bir değişiklik yapmazken  $|1\rangle$  durumuna bir faz ekler. Şekil 2.5'de Z kapısının durumlar üzerine etkisi görülmektedir. Bu kapılara Pauli-X, Pauli-Y ve Pauli-Z kapıları da denir.

$$\begin{array}{ccc}
|0\rangle & \xrightarrow{X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}} & |1\rangle \\
|1\rangle & \xrightarrow{X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}} & |0\rangle \\
a|0\rangle + b|1\rangle & \xrightarrow{X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}} & b|0\rangle + a|1\rangle
\end{array}$$

Şekil 2.3. X kapısı ve durumlar üzerine etkisi

$$\begin{array}{ccc}
|0\rangle & \xrightarrow{Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}} & i|1\rangle \\
|1\rangle & \xrightarrow{Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}} & -i|0\rangle \\
a|0\rangle + b|1\rangle & \xrightarrow{Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}} & -ib|0\rangle + ia|1\rangle
\end{array}$$

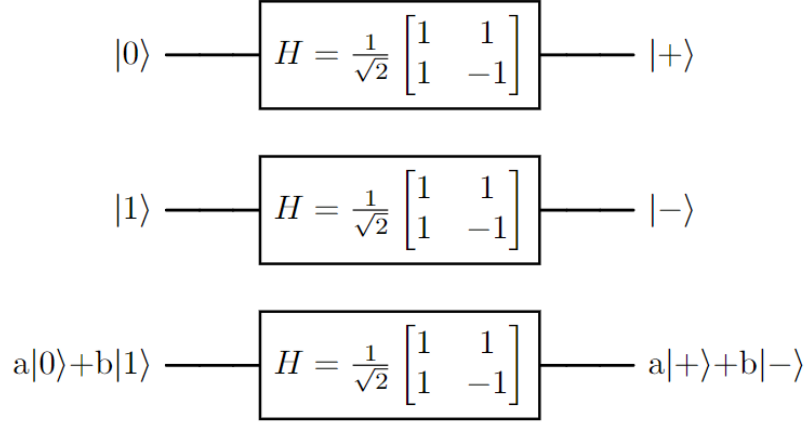
Şekil 2.4. Y kapısı ve durumlar üzerine etkisi

$$\begin{array}{ccc}
|0\rangle & \xrightarrow{Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}} & |0\rangle \\
|1\rangle & \xrightarrow{Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}} & -|1\rangle \\
a|0\rangle + b|1\rangle & \xrightarrow{Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}} & a|0\rangle - b|1\rangle
\end{array}$$

Şekil 2.5. Z kapısı ve durumlar üzerine etkisi

Hadamard kapısı tek kübite etki edip süperpozisyon durumu ortaya çıkaran bir kapıdır. Burada  $1/\sqrt{2}(|0\rangle + |1\rangle) = |+\rangle$  ve  $1/\sqrt{2}(|0\rangle - |1\rangle) = |-\rangle$  olmak üzere, Ha-

damard kapısının durumlar üzerine etkisi ve matris temsili Şekil 2.6'da görülmektedir.



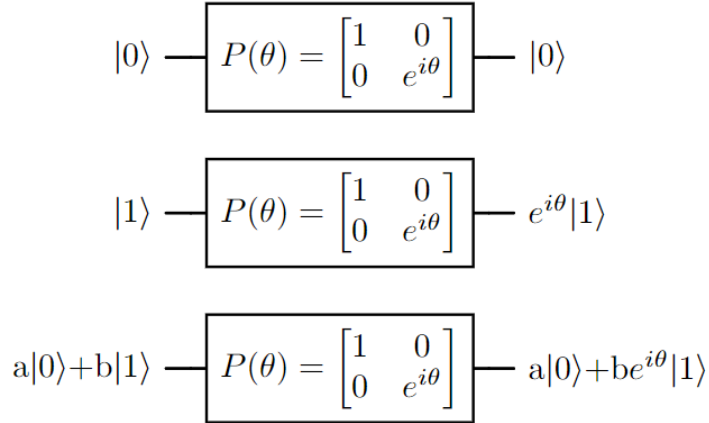
Şekil 2.6. Hadamard kapısı ve durumlar üzerine etkisi

Kubitler için Hadamard kapısının matematiksel formu,

$$H|a\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{2i\pi(0,a)}|1\rangle) \quad (2.17)$$

eşitliği ile verilir (Karimipour, 2001). Burada  $0, a = ax2^{-1}$ 'dir. Bu ikili sayı sisteminde ondalıklı sayının yazılma biçimidir.

Kubitlerde faz kayması kapısı  $P(\theta)$ ,  $|0\rangle$  durumu üzerine herhangi bir etki yapmazken  $|1\rangle$  durumuna bir faz ekler.



Şekil 2.7. Faz kayması kapısı ve durumlar üzerine etkisi

Burada  $\theta = 2\pi/2^k$  dır ve k değeri 1,2,3,...,n değerlerini alabilir. Kubitler için tanımlanan kapılar incelendiğinde  $k = 2$  için S kapısı,  $k = 3$  için T kapısı olur. Şekil 2.7'de faz kayması kapısının matris temsili ve durumlar üzerine etkisi gösterilmiştir.

Kuantum kapıları tersinir özelliğe sahip olmalıdır.  $U$  bir kuantum kapısını temsil eden matris olmak üzere,  $U^{-1} = U^\dagger$  şartının sağlanması gerekir. Bu şartı sağlayan matrislere üniter matris denir. Ayrıca  $U = U^\dagger$  şartını sağlayan matrislere de hermityen matris denir. Yazılan bu iki eşitlikten yola çıkarak  $U^{-1} = U$  olduğunu söyleyebiliriz. O halde  $U^{-1}U = I$  ise  $UU = I$  veya  $U^2 = I$  yazılabilir. Bu şartı sağlayan herhangi bir  $U$  matrisi hem üniter hem de hermityen bir matristir.

Yukarıdaki tanımlardan yola çıkarak  $e^{(-i\theta U)}$  biçimindeki bir kapının matris temsilinin nasıl bulunacağına bakalım. Bunun için  $e^x$  ifadesinin taylor açılımını kullanalım.

$$e^{(-i\theta U)} = I - iU\theta + (-i)^2 \frac{\theta^2}{2!} U^2 + (-i)^3 \frac{\theta^3}{3!} U^3 + (-i)^4 \frac{\theta^4}{4!} U^4 + (-i)^5 \frac{\theta^5}{5!} U^5 + \dots \quad (2.18)$$

$U^2 = I$  eşitliğini kullanarak yukarıdaki denklemde bir takım düzenlemeler yapalım.

$$e^{(-i\theta U)} = I - iU\theta - \frac{\theta^2}{2!} I + i \frac{\theta^3}{3!} U + \frac{\theta^4}{4!} I - i \frac{\theta^5}{5!} U + \dots \quad (2.19)$$

eşitliğin sağ tarafında bir gruptandırma yaptığımızda,

$$e^{(-i\theta U)} = I \left(1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} + \dots\right) - i \left(\theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} + \dots\right) U \quad (2.20)$$

ulaştığımız bu son eşitliğin sağ tarafında ilk parantez  $\cos \theta$ 'nın ikinci parantez ise  $\sin \theta$ 'nin açılımıdır. Böylece,

$$e^{(-i\theta U)} = \cos(\theta)I - i \sin(\theta)U \quad (2.21)$$

sonucuna ulaşırız. Bu sonucu ve daha önce tanıttığımız X,Y ve Z kapılarının matris temsillerini kullanarak, Bloch küresi üzerinde  $\theta$  kadarlık dönme işlemini

gerçekleştiren kapıların matris temsillerini oluşturabiliriz.  $R_x(\theta)$ , x ekseninde  $\theta$  kadarlık dönmeyi gerçekleştiren kapı olmak üzere, bu kapının matris temsili,

$$R_x(\theta) = e^{(-i\frac{\theta}{2}X)} = \cos\left(\frac{\theta}{2}\right)I - i \sin\left(\frac{\theta}{2}\right)X \quad (2.22)$$

eşitliği ile bulunur. Burada X ve I kapıların temsil eden matrisler yazıldığında,

$$R_x(\theta) = \cos\left(\frac{\theta}{2}\right) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - i \sin\left(\frac{\theta}{2}\right) \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (2.23)$$

sonucuna ulaşılır. Matrisin katsayısı olarak görülen  $\cos(\theta)$  ve  $\sin(\theta)$  içeriye dağıtılıp, çıkarma işlemi yapılırsa

$$R_x(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -i \sin\frac{\theta}{2} \\ -i \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \quad (2.24)$$

şeklinde matris temsili elde ederiz. Benzer yöntemlerle Y ve Z eksenleri etrafındaki  $\theta$  kadarlık dönüşleri gerçekleştiren  $R_y(\theta)$  ve  $R_z(\theta)$  kapıların matris temsillerini aşağıda gösterildiği gibi yazabiliriz.

$$R_y(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \quad (2.25)$$

$$R_z(\theta) = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix} \quad (2.26)$$

### 2.2.2. İki ve Daha Fazla Kübit İçin Kuantum Kapılar

Kuantum bilgi işlemede iki ve daha fazla kuantum durumu ve kuantum mantık kapısı tensör çarpımı ile bir araya getirilir. Örneğin A ve B iki kuantum kapısının matris temsilleri olmak üzere,

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \quad (2.27)$$

A ve B matrislerinin tensör çarpımı,

$$A \otimes B = \begin{pmatrix} A_{11}B & A_{12}B \\ A_{21}B & A_{22}B \end{pmatrix} \quad (2.28)$$

eşitliği ile hesaplanır. A ve B 2x2'lik matrisler olmasına karşın  $A \otimes B$  4x4 boyutlu bir matristir. Eşitlik 2.11 ve 2.12 de gösterildiği gibi bir kübitin saf durumları 2x1 boyutlu sütun matrisi ile temsil edilir. Eşitlik 2.28'deki tensör çarpımı kullanılarak, iki kübit bir sütun matrisi ile Eşitlik 2.29'daki gibi gösterilebilir.

$$|0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = |01\rangle \quad (2.29)$$

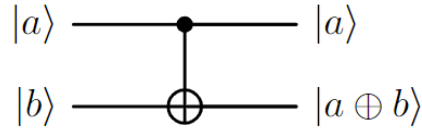
Yardımcı kübit gerektiren genelleştirilmiş bazı kuantum algoritmaların devre tasarımlarında n adet  $|0\rangle$  girişi bulunur. Bu durumda tensör çarpımı n-1 defa yapılır. Bu işlem Eşitlik 2.30'daki gibi yazılır.

$$|0\rangle^{\otimes n} = |0_1\rangle \otimes |0_2\rangle \otimes \cdots \otimes |0_n\rangle \quad (2.30)$$

Bu yöntem kuantum devrelerinin işleyişini matris tabanında anlamak için önemlidir ve Bölüm 2.3.1'de dolanıklık devresinin işleyişini açıklamak için kullanılmıştır.

İki veya daha fazla kubit üzerine etkiyen kuantum kapıları da mevcuttur. Bu kapılardan bazıları tek veya çoklu kontrollü NOT kapıları, kontrollü faz kayması kapısı, SWAP ve Fredkin kapısıdır. Kontrollü veya şartlı NOT kapısı girişlerden birinin kontrol diğerinin hedef olduğu kapıdır. Genel olarak kuantum bilgi işlemede kontrol girişinin  $|1\rangle$  olduğu durumda hedef girişe NOT uygulanır ve buna CNOT kapısı denir. Ancak kontrol girişinin  $|0\rangle$  olduğu durumlar için hedef girişe NOT işleminin uygulandığı kapılarda kullanılabilir. Şekil 2.8'de  $|a\rangle$  girişinin kontrol,  $|b\rangle$  girişinin hedef olduğu  $CNOT_a$  kapısının devre gösterimi,

işlevi Eşitlik 2.31’de işlevselliği ve Eşitlik 2.32’de de matris temsili görülmektedir.

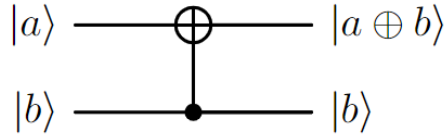


Şekil 2.8.  $CNOT_a$  kapısının devre gösterimi

$$CNOT_a |a, b\rangle = |a, a \oplus b\rangle \quad (2.31)$$

$$CNOT_a = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (2.32)$$

Kontrol girişinin  $|b\rangle$ , hedef girişin  $|a\rangle$  olduğu kapı  $CNOT_b$  kapısı olarak tanımlanabilir. Bu kapının devre gösterimi Şekil 9’da, işlevi ve matris temsili de sırasıyla Eşitlik 2.33 ve 2.34’de görülmektedir.

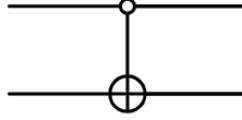


Şekil 2.9.  $CNOT_b$  kapısının devre gösterimi

$$CNOT_b |a, b\rangle = |a \oplus b, b\rangle \quad (2.33)$$

$$CNOT_b = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad (2.34)$$

Genelde kontrol girişinin  $|1\rangle$  olması durumunda NOT kapısının uygulanması yaygındır. Ancak Kontrol girişinin  $|0\rangle$  olması durumunda da hedef giriş NOT kapısının uygulanmasını gerektiren işlemler de olabilir. Kontrol girişinin  $|0\rangle$ , olduğu durumda uygulanan *CNOT* kapısının devre gösterimi Şekil 2.10'da, matris temsili de Eşitlik 2.35'de görülmektedir.

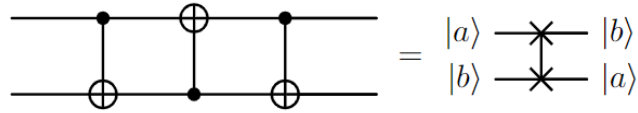


Şekil 2.10. Kontrol girişi  $|0\rangle$  olan CNOT kapısı

$$CNOT_0 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.35)$$

Benzer şekilde kontrol girişinin ikinci giriş, hedefin de ilk giriş olduğu  $|0\rangle$  kontrollü CNOT kapısı da oluşturulabilir.

SWAP kapısı iki giriş üzerine etki eden ve girişlerin yerlerini değiştiren bir kapıdır. Şekil 2.11'de görüldüğü gibi üç tane CNOT kapısı ile oluşturulabilir. İşlevi  $SWAP |a, b\rangle = |b, a\rangle$  eşitliği ile gösterilebilir.

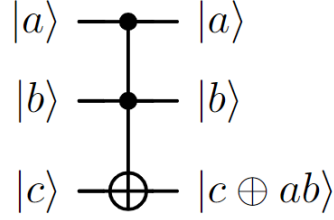


Şekil 2.11. SWAP kapısı ve CNOT kapıları ile oluşturulan özdeşliği

Bu kapının matris temsili aşağıda görülmektedir.

$$SWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.36)$$

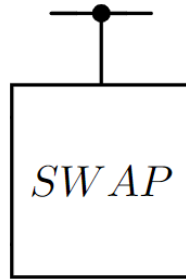
Kuantum devrelerde birden fazla kontrol girişine sahip kapılar da mevcuttur. Bilinen en yaygın kapı iki kontrollü Toffoli kapısıdır. Bu kapı kontrol girişlerinin her ikisinin de  $|1\rangle$  olması durumunda üçüncü giriş NOT operatörünü uygular. Toffoli kapısının devre gösterimi Şekil 2.12’de, matris temsili de Eşitlik 2.37’de görülmektedir.



Şekil 2.12. Toffoli kapısı

$$Tofoli = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \quad (2.37)$$

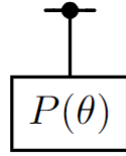
Fredkin kapısı olarak bilinen kontrollü SWAP kapısı, kontrol girişi  $|1\rangle$  ise diğer iki giriş SWAP işlemi uygulayan, üç girişli bir kapıdır. Bu kapının devre şeması Şekil 2.13’de matris temsili de Eşitlik 2.38’de görülmektedir.



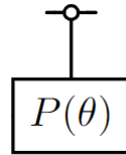
Şekil 2.13. Fredkin Kapısı

$$Fredkin = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.38)$$

Bu çalışmada sıklıkla kullanacağımız bir diğer kapı bölüm 2.2.1’de tanıttığımız faz kayması kapısının kontrollü olanıdır. Kontrol girişinin  $|0\rangle$  ya da  $|1\rangle$  olması durumuna göre hedef girişe faz kayması işlemi uygulayan bu kapı kontrollü faz kayması kapısı olarak isimlendirilir. Kontrol girişi  $|1\rangle$  ve  $|0\rangle$  olan bu kapının devre-deki temsili Şekil 2.14’de görülmektedir. Kontrollü faz kayması kapılarının matris temsilleri 2.2.3 bölümünde genelleştirilmiş eşitlikten yola çıkılarak verilecektir.



(a)

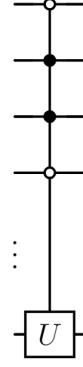


(b)

Şekil 2.14. (a) Kontrol biti 1 (b) kontrol biti 0 olan  $CP(\theta)$  kapıları

İki’den daha fazla kontrollü girişi olan kapılar da mevcuttur. Kontrollerin tamamı  $|1\rangle$  olabildiği gibi tamamı  $|0\rangle$  da olabilir. Ya da daha özel şartlar için kontrol girişlerinin bazılarının  $|1\rangle$  bazılarının da  $|0\rangle$  olduğu çoklu kapılar da tasarlanabilir. Şekil 2.15’de n giriş üzerine etki eden, kontrol girişlerinin bazılarının  $|1\rangle$  bazılarının da  $|0\rangle$  olduğu çoklu kontrollü bir kapının devre temsili görülmektedir. Burada U herhangi bir kuantum kapısını temsil etmektedir. Kontrollü Hadamard Kapısı veya kontrollü Y kapısı gibi benzer kapı tasarımları 2.2.1 bölümümde tanıttığımız

tüm kapılar için yapılabilir.



Şekil 2.15. Çoklu kontrollü U kapısı

### 2.2.3. Kuditler ve Bazı Genelleştirilmiş Kuantum Kapılar

Bilginin kodlanacağı kuantum sisteminin olası durum sayısı ikiden fazla ise bu sistemler kudit olarak isimlendirilir. Örneğin spin kuantum sayısı 1 olan bir sistemin olası durumları 1, 0 ve -1'dir. Bu üç seviyeli bir kuantum sistemidir ve kutrit olarak adlandırılır. Spin kuantum sayısı 3/2 olan bir kuantum sisteminin olası durumları 3/2, 1/2, -1/2 ve -3/2'dir. Bu dört seviyeli bir kuantum sistemidir ve kukuart olarak adlandırılır. Bu iki örnekten yola çıkarak, d kuantum sisteminin olası spin durumları sayısı olmak üzere  $d \geq 3$  olan sistemler kuditleri temsil eder(Gedik, 2015). Herhangi bir d seviyeli kudit sistemi için hesaplama baz seti,  $|0\rangle, |1\rangle, |2\rangle, \dots, |d-1\rangle$  elemanlarından oluşur. Genel olarak bir kudit,

$$|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle + \alpha_2 |2\rangle + \dots + \alpha_{d-1} |d-1\rangle = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{d-1} \end{pmatrix} \quad (2.39)$$

eşitliği ile temsil edilir. Bu eşitliğe göre d seviyeli bir kudit aynı zamanda dx1 boyutlu sütun matrisi ile temsil edilir. Tablo 2.1'de  $d = 3$  olduğunda yani kutritler için Hesaplama bazları ve onlara karşılık gelen sütun matrisler görülmektedir.

Tablo 2.1. Kütritler ( $d = 3$ ) için hesaplama bazları ve matris temsilleri

Kuantum Durumu	Hesaplama Bazı	Matris Temsili
$m_s = +1$	$ 0\rangle$	$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$
$m_s = 0$	$ 1\rangle$	$\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$
$m_s = -1$	$ 2\rangle$	$\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$
Süperpozisyon	$\alpha_0  0\rangle + \alpha_1  1\rangle + \alpha_2  2\rangle$	$\begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{pmatrix}$

Bir önceki kısımda kubitler için bazı kuantum kapılarını vermiştik. Bu kısımda bu kapıların bazılarının herhangi bir  $d$  seviyeli kuantum sistemi için matris temsilleri verilecektir. Herhangi bir  $d$  seviyeli sistem için Hadamard kapısının matris temsili;

$$H_d = \frac{1}{\sqrt{d}} \sum_{i,j=0}^{d-1} w^{ij} |i\rangle \langle j| \quad (2.40)$$

eşitliği ile bulunur (Karimipour, 2001). Burada  $w = e^{\frac{2i\pi}{d}}$  dir. Kütrit sistemler ( $d = 3$ ) için Hadamard kapısının matris temsili Eşitlik 2.41'de durumlar üzerine etkisi de Şekil 2.16'da gösterilmiştir.

$$H_3 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 1 & 1 \\ 1 & w & w^2 \\ 1 & w^2 & w \end{pmatrix} \quad (2.41)$$

$$\begin{aligned} |0\rangle &\text{---} \boxed{H_3} \text{---} \frac{1}{\sqrt{3}}(|0\rangle + |1\rangle + |2\rangle) \\ |1\rangle &\text{---} \boxed{H_3} \text{---} \frac{1}{\sqrt{3}}(|0\rangle + w|1\rangle + w^2|2\rangle) \\ |2\rangle &\text{---} \boxed{H_3} \text{---} \frac{1}{\sqrt{3}}(|0\rangle + w^2|1\rangle + w|2\rangle) \end{aligned}$$

Şekil 2.16.  $H_3$  kapısının saf durumlara etkisi

Kubitler için tanımlanan X (NOT) kapısının herhangi bir d seviyeli sistem için matris temsili,

$$X_d = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \vdots & 1 & 0 \end{pmatrix} \quad (2.42)$$

ile verilir (Wang, 2020).  $X_d$  kapısı bazlar arasında ardışık kaydırma işlemini gerçekleştirir. Eşitlik 2.42 kullanılarak kutrit sistemler için  $X_3$  kapısının matris temsili Eşitlik 2.43'deki gibi elde edilir.

$$X_3 = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \quad (2.43)$$

Bu kapının kutrit sistemlerdeki baz setlerine uygulanması sonucu Şekil 2.17'de görülen dönüşümler gerçekleşir.

$$\begin{aligned} |0\rangle & \xrightarrow{X_3} |1\rangle \\ |1\rangle & \xrightarrow{X_3} |2\rangle \\ |2\rangle & \xrightarrow{X_3} |0\rangle \end{aligned}$$

Şekil 2.17.  $X_3$  kapısının kütritlerin saf durumlarına etkisi

Ayrıca  $X_3^\dagger X_3 = I$  olduğundan,  $X_3^\dagger$  kapısı  $X_3$  kapısının yaptığı işlevi geri alır. Bu kuantum kapılarının tersinir özelliğidir.

Kuditler için faz kayması kapısının matris temsillerinin elemanları,

$$P_d(\theta_k) = \sum_{j=0}^{d-1} e^{i\theta_k j} |j\rangle \langle j| \quad (2.44)$$

eşitliği ile bulunur. Burada  $\theta_k = \frac{2\pi}{d^k}$  dir. Kütrit sistemler için  $P_3(\theta_k)$ 'nin matris

temsili,

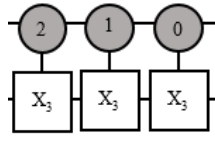
$$P_3(\theta_k) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & e^{i\theta_k} & 0 \\ 0 & 0 & e^{2i\theta_k} \end{pmatrix} \quad (2.45)$$

biçiminde yazılır. Burada ise  $\theta_k = \frac{2\pi}{3^k}$ 'dir. Temsilden de görüldüğü gibi yalnızca  $|1\rangle$  ve  $|2\rangle$  durumları üzerinde bir faz değişikliğine neden olur. Bu değişiklikler Şekil 2.18'de görülmektedir.

$$\begin{aligned} |0\rangle &\xrightarrow{P_3(\theta_k)} |0\rangle \\ |1\rangle &\xrightarrow{P_3(\theta_k)} e^{i\theta_k} |1\rangle \\ |2\rangle &\xrightarrow{P_3(\theta_k)} e^{2i\theta_k} |2\rangle \end{aligned}$$

Şekil 2.18.  $P_3(\theta_k)$  kapısının kütrit saf durumlara etkisi

Kubitlerde olduğu gibi kuditler içinde çoklu giriş üzerine etkiyen kontrollü kapılar mevcuttur. Eşitlik 2.42'de tanımlanan  $X_d$  kapısı kubitler için CNOT kapısının işlevine benzer şekilde kullanılabilir. Ancak kuditler için  $d - 1$  tane hesaplama bazı olduğu için kontrol girdisinin ne olduğunu bilmek gerekir. Örneğin kuditler için 3 tane kontrol durumu vardır. Bu durumlar Şekil 2.19'da görülmektedir.



Şekil 2.19. Kudit sistem için kontrollü  $X_3$  kapıları

Şekil 2.19'da kontrol girişinin  $|2\rangle$ ,  $|1\rangle$  ve  $|0\rangle$  olduğu durumlarda  $X_d$  kapısının uygulayan kapıların devre temsilleri görülmektedir. Bu kapılar özel kapılardır ve her biri  $9 \times 9$ 'luk kare matrisler ile temsil edilirler. Benzer şekilde Eşitlik 2.45'de

matris temsili verilen faz kayması kapısının kontrollü uygulanmasının da gerekli olduğu durumlar vardır. Kontrollü faz kayması kapısının matris temsilini veren genel form aşağıda görülmektedir (Pavlidis, 2017).

$$CP_d(\theta_k) = \sum_{j=0}^{d-1} \sum_{m=0}^{d-1} e^{i\theta_j m} |j\rangle \langle j| \otimes |m\rangle \langle m| \quad (2.46)$$

Bu eşitlikte,  $\theta_k$ 'nin genel tanımı  $\theta_k = \frac{2\pi}{d^k}$  olduğundan  $\theta_k = \frac{2\pi}{3^k}$ 'dir. İfadedeki tensör çarpımı bir sonraki bölümde tartışılacaktır. Kontrollü  $X_d$  kapısında olduğu gibi kontrol girdisini özel olarak belirtmeye gerek yoktur. Çünkü Eşitlik 2.46 kullanılarak oluşturulacak matris temsili olası tüm kontrol durumları için faz kapısının uygulanmasını içerir. Bu kapıda iki girişli kutrit sistemlerde olduğu gibi 9x9'luk matris ile temsil edilir. Herhangi bir k değeri için  $CP_3(\theta_k)$  kapısının matris temsili aşağıdaki gibi yazılır.

$$CP_3(\theta_k) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e^{\frac{i2\pi}{3^k}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & e^{\frac{i4\pi}{3^k}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e^{\frac{i4\pi}{3^k}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & e^{\frac{i8\pi}{3^k}} \end{pmatrix} \quad (2.47)$$

Son olarak, kutrit sistemlerdeki SWAP kapısı kubitlerde olduğu gibi iki girdi üzerine etki ederek onların yerlerinin değişmesini sağlar.

Dört olası duruma sahip kuantum sistemler kukuart olarak tanımlanır. Kukuartlar için hesaplama baz seti ve onlara karşılık gelen matris temsilleri Tablo 2.2'de görülmektedir. Tabloda da görüldüğü gibi kukuart sisteminin hesaplama bazlarının her biri 4x1 lik sütun matrisleri ile temsil edilir. Tek bir kukuart üzerine etkiyen kuantum mantık kapısı 4x4'lük kare matrisken, kontrollü kapılar 16x16'lık

matrislerle temsil edilirler.

Tablo 2.2. Kukuartlar ( $d = 4$ ) için hesaplama bazları ve matris temsilleri

Kuantum Durumu	Hesaplama Bazı	Matris Temsili
$m_s = +\frac{3}{2}$	$ 0\rangle$	$\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$
$m_s = +\frac{1}{2}$	$ 1\rangle$	$\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$
$m_s = -\frac{1}{2}$	$ 2\rangle$	$\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$
$m_s = -\frac{3}{2}$	$ 3\rangle$	$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$
Süperpozisyon	$\alpha_0  0\rangle + \alpha_1  1\rangle + \alpha_2  2\rangle + \alpha_3  3\rangle$	$\begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix}$

Eşitlik 2.40 kullanılarak kukuart sistemler için Hadamard kapısının matris temsilini oluşturabiliriz. Bu temsil 4x4 boyutlu bir kare matristir Eşitlik 2.48'de görülmektedir. Hadamard kapısı saf durumlara uygulandığında süperpozisyon durumu oluşturan bir kapıdır. Sırasıyla  $|0\rangle$ ,  $|1\rangle$ ,  $|2\rangle$ ,  $|3\rangle$  durumlarına etkiğinde oluşturduğu süperpozisyon durumları Şekil 2.20'de görülmektedir.

$$H_4 = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \quad (2.48)$$

$$\begin{aligned}
|0\rangle &\longrightarrow \boxed{H_4} \left[ \frac{1}{2}(|0\rangle+|1\rangle+|2\rangle+|3\rangle) \right] \\
|1\rangle &\longrightarrow \boxed{H_4} \left[ \frac{1}{2}(|0\rangle+i|1\rangle-|2\rangle-i|3\rangle) \right] \\
|2\rangle &\longrightarrow \boxed{H_4} \left[ \frac{1}{2}(|0\rangle-|1\rangle+|2\rangle-|3\rangle) \right] \\
|3\rangle &\longrightarrow \boxed{H_4} \left[ \frac{1}{2}(|0\rangle-i|1\rangle-|2\rangle+i|3\rangle) \right]
\end{aligned}$$

Şekil 2.20. Kukuart sistem için Hadamard kapısı ve işlevi

Eşitlik 2.42'deki matris temsili kullanılarak ( $d = 4$ ) kukuart sistemler için  $X$  kapısı aşağıdaki Eşitlik 2.49'daki gibi yazılabilir. Bu kapının işlevi Şekil 2.21'de görülmektedir.

$$X_4 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (2.49)$$

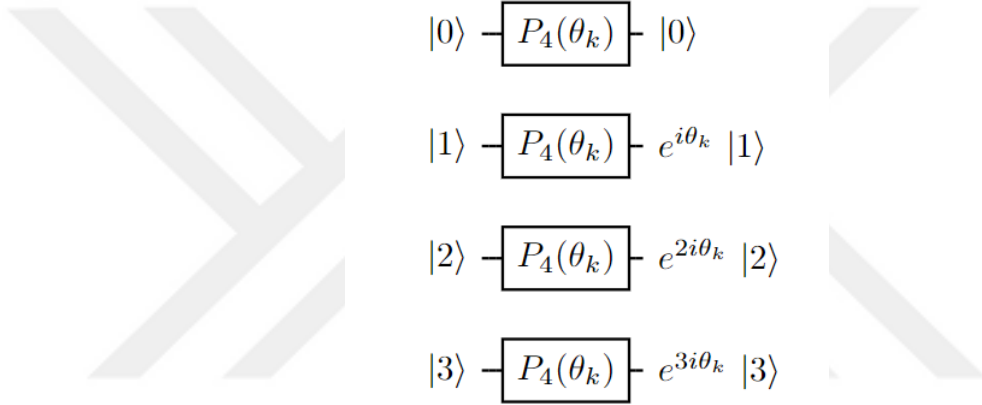
$$\begin{aligned}
|0\rangle &\longrightarrow \boxed{X_4} \left[ |1\rangle \right] \\
|1\rangle &\longrightarrow \boxed{X_4} \left[ |2\rangle \right] \\
|2\rangle &\longrightarrow \boxed{X_4} \left[ |3\rangle \right] \\
|3\rangle &\longrightarrow \boxed{X_4} \left[ |0\rangle \right]
\end{aligned}$$

Şekil 2.21. Kukuart sistem için  $X_4$  kapısının durumlar üzerine etkisi

Eşitlik 2.44 kullanılarak kukuart sistemleri için faz kayması kapısı  $P_4(\theta_k)$ 'nin matris temsilini yazabiliriz.

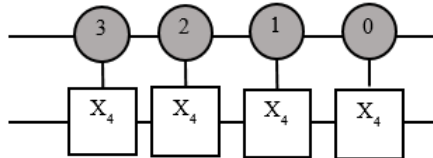
$$P_4(\theta_k) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{i\theta_k} & 0 & 0 \\ 0 & 0 & e^{2i\theta_k} & 0 \\ 0 & 0 & 0 & e^{3i\theta_k} \end{pmatrix} \quad (2.50)$$

Burada  $\theta = \frac{2\pi}{4^k}$ 'dir. Matris temsilinden anlaşılacağı üzere  $|0\rangle$  durumu hariç diğer durumlar üzerinde bir faz değişikliğine neden olur. Bu değişiklikler Şekil 2.22'de görülmektedir.



Şekil 2.22. Kukuart sistem için  $P_4(\theta_k)$  kapısının durumlar üzerine etkisi

Kukuart sistemlerde de kubit ve kutrit sistemlere benzer olarak kontrol durumlarının farklı olduğu kapılar kullanılabilir. Örneğin  $X_4$  kapısının kontrollü olarak uygulanması 4 farklı çeşitde olabilir. Kontrollü  $X_4$  kapıları Şekil 2.23'de görülmektedir. Bu kapıların her biri  $16 \times 16$ 'lık kare matrisler ile temsil edilir.



Şekil 2.23. Kukuart sistem için kontrollü  $X_4$  kapıları

Faz kayması kapısının kontrollü hali kukuart sistemler içinde uygulanır.

Herhangi bir  $k$  değeri için  $CP_4(\theta)$  kapısının matris temsilinin köşegen elemanları,

$$CP_4(\theta) = (1, 1, 1, 1, 1, e^{\frac{2i\pi}{4^k}}, e^{\frac{4i\pi}{4^k}}, e^{\frac{6i\pi}{4^k}}, 1, e^{\frac{4i\pi}{4^k}}, e^{\frac{8i\pi}{4^k}}, e^{\frac{12i\pi}{4^k}}, 1, e^{\frac{6i\pi}{4^k}}, e^{\frac{12i\pi}{4^k}}, e^{\frac{18i\pi}{4^k}}) \quad (2.51)$$

eşitliği ile yazılır. Bu 16x16'lık bir matristir ve köşegen elemanları dışındaki tüm elemanlar sıfır'dır.

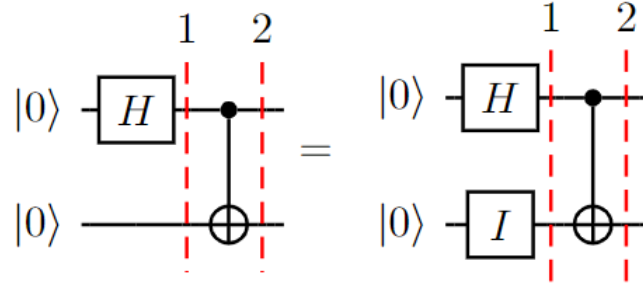
### 2.3. Kuantum Devreler

Kuantum devreler, bir algoritmanın gerçekleştirilmesi için kuantum kapılar kullanılarak tasarlanır. Devrelerin işleyişini girişlerin ve kapıların matris temsillerini kullanarak anlamamızın bazı zorlukları vardır. Herhangi bir  $d$  seviyeli kuantum sistemi için  $n$  girişli bir devrenin girişlerini temsil eden sütun matrisinin boyutu  $d^n \times 1$ 'dir. Bu girişin üzerine etkiyen kapıların matris temsilleri de  $d^n \times d^n$  boyutludur. Bu değerler düşük sayıda girişler için anlaşılması kolay olsa da giriş sayısının artması durumunda devasa boyutlu matrislerle işlem yapmak zorunda kalır. Örneğin 5 girişli kuart tabanlı bir kuantum devresinde, girişi temsil eden sütun matrisinin boyutu  $1024 \times 1$ 'dir. Bu girişe etkiyen kapıların matris temsili boyutları ise  $1024 \times 1024$  boyutludur. Böyle bir devreyi el yordamı ile hesaplamak neredeyse imkansızdır. Kuantum devrelerinin işleyişini anlamak için 2.3.1 bölümünde iki kubit girişli bir dolanıklık devresinin analizi yapılacaktır.

#### 2.3.1. Dolanıklık Devresi

Kuantum dolanıklık, kuantum bilgi işlemeyi geleneksel bilgi işlemeye üstün kılan fenomenlerden biridir (Bennet, 2000). Dolanık hale getirilmiş iki kuantum sistemi kuantum ışınlama gibi süreçlerde kullanılır. Dolanık hale getirilmiş iki kuantum sisteminde sistemlerin biri hakkında sahip olduğumuz bilgi ile ölçüm yapmadığımız diğer sistem hakkında bilgi sahibi oluruz. Dolanık durumlar da sistemin bir süperpozisyonu olarak kabul edilebilir. Şekil 2.24'de Hadamard ve CNOT kapıları kullanılarak oluşturulmuş, iki saf duruma kodlanmış bilgiyi dolanık hale getiren bir kuantum dolanıklık devresi görülmektedir. Bu şekilde birbirine eşit olan iki devre görülmektedir.

Soldaki dolanıklık devresinde girişlerin her ikisi de  $|0\rangle$  olarak ayarlanmıştır. Girişler diğer baz durumları da olabilir. Girişler tensör çarpımı ile ifade edilir ve



Şekil 2.24. İki girişli dolanıklık devresi

Eşitlik 2.52’de gösterildiği gibi daha sade biçimde de yazılabilir.

$$|\psi_{giris}\rangle = |0\rangle \otimes |0\rangle = |00\rangle \quad (2.52)$$

Birinci adımda ilk girişe Hadamard kapısı uygulanmıştır. Hadamard kapısının  $|0\rangle$  üzerindeki etkisini daha önce tanımlamıştık. Böylece Hadamard kapısından sonra gelinen durum,

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle \quad (2.53)$$

eşitliği ile yazılır. Bu eşitliği düzenlersek,

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) \quad (2.54)$$

sonucuna ulaşırız. İkinci adımda, birinci giriş kontrol, ikinci giriş hedef olmak üzere CNOT kapısı uygulanmıştır. Bu kapının sistemi getirdiği durum  $|\psi_2\rangle$  olmak üzere,  $|\psi_2\rangle = CNOT |\psi_1\rangle$  yazılabilir.

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}(CNOT |00\rangle + CNOT |10\rangle) \quad (2.55)$$

eşitliğin sağındaki ilk CNOT kapısı kontrol girişi  $|0\rangle$  olduğu için ikinci giriş üzerinde bir değişikliğe sebep olmaz. Ancak ikinci CNOT kontrol girişi  $|1\rangle$  olduğu

için ikinci girişe NOT işlemi uygular. Böylece sonuç,

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \quad (2.56)$$

olur. Bu sonucu doğru okumak çok önemlidir. Birinci çıkışı ölçtüğümüzde sonuç  $|0\rangle$  ise diğer çıktı da kesinlikle  $|0\rangle$ 'dır. Bu ihtimal yüzde ellidir. Diğer yüzde elli ihtimal ise birinci çıktının  $|1\rangle$  olarak ölçülmesi sonucu diğerinin de  $|1\rangle$  olmasıdır. Dolanıklık devresi bu iki olası sonucun bir süperpozisyon çıktısını verir. Şimdi şekil 2.24'de bulunan sağdaki devrenin matris temsili ile analizini yapalım. Devrenin girişi için,

$$|\psi_{giris}\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.57)$$

hazırlığını yapabiliriz. Görüldüğü gibi giriş  $4 \times 1$  boyutlu bir sütun matrisi ile temsil edilir. İlk uygulanan kapı Hadamard kapısıdır. Ancak Hadamard kapısının matris temsili  $2 \times 2$  lik bir kare matristir. Bu nedenle Hadamard kapısının bulunduğu yerde diğer girişe de sonucu değiştirmeyecek olan bir özdeşlik kapısı yerleştirilir. Bu daha çok girişe sahip devreler içinde geçerlidir. Boş olan yerlere özdeşlik kapıları yerleştirilerek matris işlemleri üzerinden devam edilebilir. Bu sayede birinci işlemde girişe uygulanacak  $4 \times 4$ 'lük bir matris oluşturulur. Bu Hadamard kapısının matris temsil ile özdeşlik kapısının matris temsilinin tensör çarpımı ile gerçekleştirilir. Devrenin tamamı için Eşitlik 2.58 yazılabilir. Bu eşitlikte kapıların matris temsillerini yerlerine yazalım.

$$|\psi_2\rangle = CNOT(H \otimes I) |\psi_{giris}\rangle \quad (2.58)$$

$$|\psi_2\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad (2.59)$$

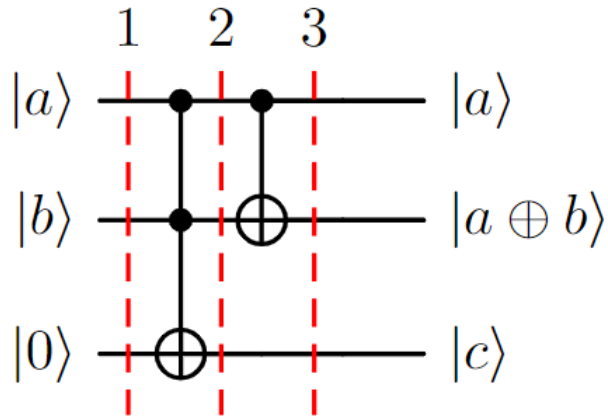
Bu eşitlikteki sonucu açarsak,

$$|\psi_2\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \left( \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right) = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) \quad (2.60)$$

elde edilir. Bu da diğer incelememizdeki ulaştığımız sonuçla aynıdır.

### 2.3.2. Tam ve Yarı Toplayıcı Devreler

Bu çalışmanın konusu QFT'ye dayalı aritmetik işlemler olduğu için, kuantum bilgi işlemede var olan yarı ve tam toplayıcı devreleri tanımak faydalı olacaktır. Şekil 2.25'de yarı toplayıcı kuantum devresi görülmektedir.



Şekil 2.25. Yarı toplayıcı kuantum devre

Bu devrede 1. adımda giriş durumu hazırlanır. Giriş durumu  $|\psi_{giris}\rangle = |a \otimes b \otimes 0\rangle$ 'dir. 2. adımda girişe Toffoli kapısı, 3. adımda da ilk giriş kontrol ikinci

giriş hedef olmak üzere CNOT kapısı uygulanır.

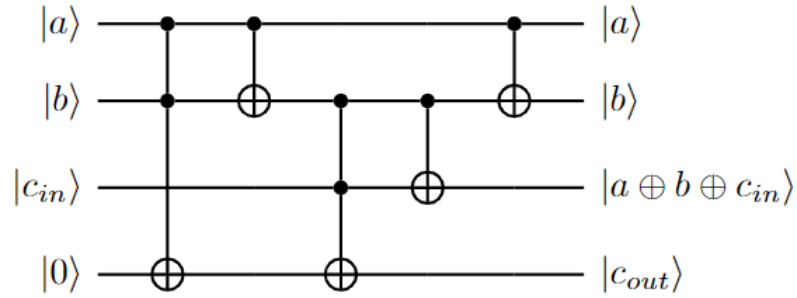
$$(CNOT)(Tofoli) |a \otimes b \otimes 0\rangle = |a \otimes (a \oplus b) \otimes c\rangle \quad (2.61)$$

Devrenin çıktısında girişlerin modüler toplamı ikinci hat üzerinde verilirken, elde sayı var ise bu sayı üçüncü hat üzerinden okunur. Tüm olası girişlere ve bunlara karşılık gelen çıkış değerleri aşağıdaki Tablo 2.3'de verilmiştir.

Tablo 2.3. Yarı toplayıcı devre için doğruluk tablosu

$ a\rangle$	$ b\rangle$	$ a\rangle$	$ a \oplus b\rangle$	$ c\rangle$
0	0	0	0	0
0	1	0	1	0
1	0	1	1	0
1	1	1	0	1

Tam toplayıcı kuantum devresi yarı toplayıcıya benzer bir yapıya sahiptir. Ancak bir önceki işlemde elde edilen elde değerinin bir sonraki bit- bit toplamına gönderilmesi için ekstra girişler kullanılarak tasarlanmıştır. Tam toplayıcı kuantum devresi Şekil 2.26'da görülmektedir.



Şekil 2.26. Tam toplayıcı kuantum Devre

Bu devreden çok sayıda birbirine entegre edilerek yani birinin  $c_{out}$  değerinin diğerinin  $c_{in}$  değerine aktarılması ile çok haneli sayılar arasında toplama işlemi gerçekleştiren devreler tasarlanabilir.

## 2.4. Kuantum Algoritmalar

Bu kısımda, konu ile ilgili olması açısından QFT algoritması, kuantum faz tahmini algoritmasını gerçekleştiren devre ve  $a$  ve  $b$  gibi iki sayının QFT tabanlı toplamını gerçekleştiren kuantum devresinin tanıtımı yapılacaktır. Kuantum algoritmalar geleneksel bilgi işleme yöntemleri ile yapılan işlemleri, süperpozisyon, dolanıklık ve kopyalanamama gibi kuantum mekaniksel özellikler kullanarak daha hızlı ve daha güvenli olarak gerçekleştirir. Bu özelliklere kuantum üstünlüğü denir. Ancak kuantum algoritmaları gerçekleştirecek kuantum devrelerin tasarlanması ve gerçekleştirilmesi bazı zorluklar içerir. Kuantum mekaniksel sistemlerin çevrelerinden izole ve manipüle edilmesi zor olduğundan bazı fiziksel sistemler kullanılarak taklit edilirler.

Eldeyi diğer bit-bit toplamına taşıyarak ilerleyen toplama algoritması (Quantum Ripple Adder) bilinen ilk kuantum algoritmalarındandır. Bu çalışmada konusu olan QFT, girdiler ne olursa olsun sistemi süperpozisyon durumuna getirerek, paralel işlemin önünü açmış olur. QFT her ne kadar tek başına kullanılmasa da paralel işlemeye hazırlık aşaması kuantum bilgi işleme için temel kabul edilen algoritma ve devrelerde alt algoritma kullanılır (Zhou, 2017). Bir diğer algoritma Deutsch-Jozsa algoritmasıdır. Klasik bilgi işlemede çok sayıda sorgu ile yapılacak olan işlemi az sorgu ile gerçekleştirir. Daha çok, sorgulanan fonksiyonun dengeli olup olmadığını ortaya koyar. Benzer şekilde Bernstein-Vazirani algoritması bit dizilerini tek sorguda bulmaya yarayan bir algoritmadır. Simon's algoritması özellikle bir türdeki özel fonksiyonların yapısını bulmak için kullanılır. Klasik bir bilgisayarla bu tür bir problem genellikle zorlu olabilir ve çok sayıda sorgu gerektirebilirken, Simon's algoritması bu süreci büyük ölçüde hızlandırabilir. Bu da kuantum bilgisayarlarının bazı matematiksel problemleri daha hızlı ve etkin bir şekilde çözebileceğini gösterir.  $N$  elemanlı bir veri tabanından istenilen bilgiyi  $\sqrt{N}$  adımda bulduran Grover algoritması ve Çok basamaklı sayıları asal çarpanlarına ayıran Shor algoritması diğer önemli kuantum algoritmalarıdır. Hamiltonyenlerin özdeğerlerini bulmak için de kuantum faz tahmini algoritması bilinen kuantum algoritmaların başını çeker (Kasirajan, 2021 ; Coles, 2018; Jozsa, 1997). Ayrıca son zamanlarda klasik bilgi işleme ile gerçekleştirilen makine öğrenmesi ve yapay

zeka alanlarındaki Ortalama K ve lineer denklem sistemlerini çözmek için önerilen HHL algoritmaları da kuantum algoritmalar sınıfına girmiştir (Harrow vd. 2009).

#### 2.4.1. Kuantum Fourier Transform

Kuantum Fourier Transform (QFT) kuantum bilgi işlemede önemli bir yere sahiptir (Nielsen ve Chuang, 2010). Kuantum bilgi işlemede temel kabul edilen algoritmalarda yer alır. QFT, Hızlı Fourier Transform'un (FFT) kuantum karşılığı olarak bilinir (Cooley, 1965). Çoklu girişe sahip kuantum devrelerinde girişlere uygulanarak olası tüm durumların bir süperpozisyonunu oluşturur. Bu işlevselliği ile tek kübit üzerine etki eden Hadamard kapısına benzer. Böylece bilgi işleme süreci paralel işlemeye hazır hale getirilir.  $|x\rangle = |x_1\rangle \otimes |x_2\rangle \otimes \dots \otimes |x_n\rangle$  ve  $|y\rangle = |y_1\rangle \otimes |y_2\rangle \otimes \dots \otimes |y_n\rangle$  olmak üzere,

$$QFT |x\rangle = \frac{1}{\sqrt{d^n}} \sum_{k=0}^{d^n-1} w^{xy} |y\rangle \quad (2.62)$$

eşitliği ile yazılır. Burada  $w = e^{\frac{2\pi ixy}{d^n}}$ 'dir. Eşitlik 2.62'de katsayı olan  $\frac{1}{\sqrt{d^n}}$  ve  $w^{xy}$ 'nin çarpımı oluşan süperpozisyon durumundaki her bir elemanın olasılık genliğine karşılık gelir. Bu eşitlik kullanılarak herhangi bir d seviyeli kuantum sistemi için QFT operatörünün matris temsili aşağıdaki gibi elde edilir.

$$QFT = \frac{1}{\sqrt{d^n}} \begin{pmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & w & w^2 & w^3 & \dots & w^{(d^n-1)} \\ 1 & w^2 & w^4 & w^6 & \dots & w^{2(d^n-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & w^{(d^n-1)} & w^{2(d^n-1)} & w^{3(d^n-1)} & \dots & w^{(d^n-1)(d^n-1)} \end{pmatrix} \quad (2.63)$$

Genelleştirilmiş matris temsilinden yola çıkarak  $d = 2$  ve  $n = 2$  için (iki kübit) QFT operatörünün matris temsili aşağıdaki gibi bulunur.

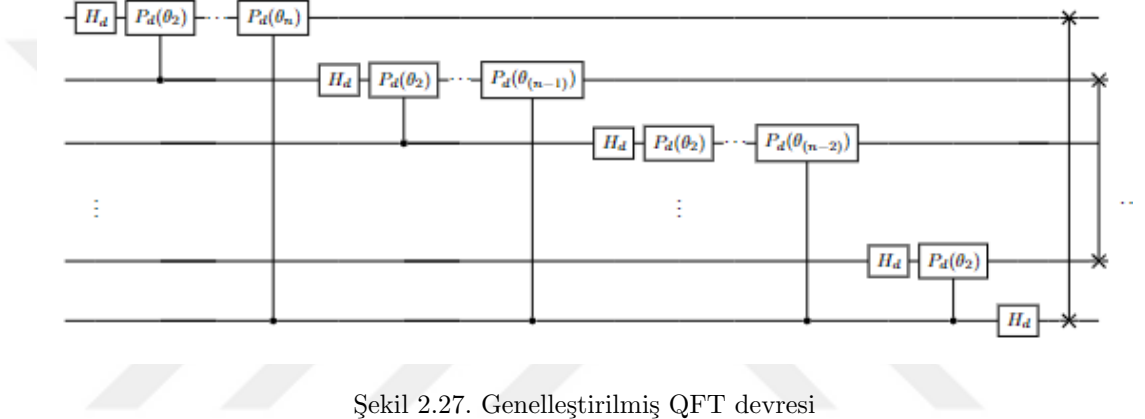
$$QFT = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \quad (2.64)$$

Benzer şekilde IQFT operatörünün matematiksel formu

$$IQFT |y\rangle = \frac{1}{\sqrt{d^n}} \sum_{x=0}^{d^n-1} w^{-xy} |x\rangle \quad (2.65)$$

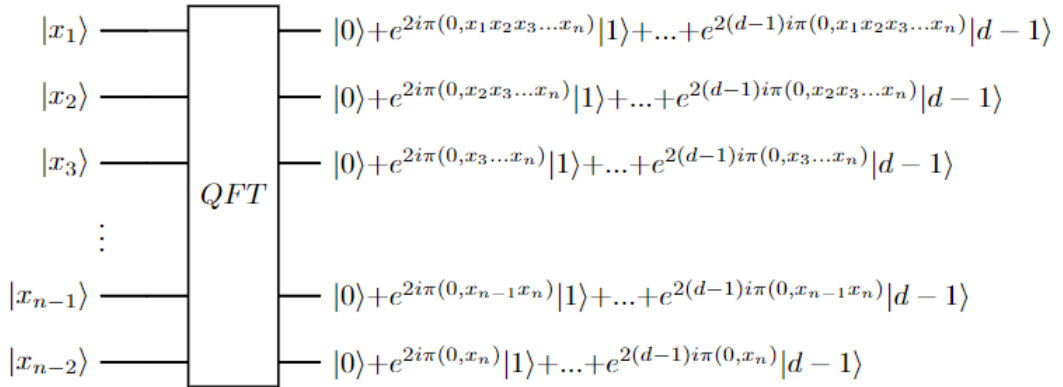
olarak yazılır.

Genelleştirilmiş QFT devresi Şekil 2.27’de görülmektedir. Görüldüğü gibi QFT devresi Hadamard, şartlı faz kayması kapıları ve SWAP kapılarından oluşmaktadır.



Şekil 2.27. Genelleştirilmiş QFT devresi

Şekil 2.28’de  $|x_1 x_2 \dots x_n\rangle$  girişli bir devreye QFT operatörü uygulandıktan sonra elde edilen çıktının matematiksel sonuçları görülmektedir. Her çıktıya tensör çarpımı uygulanarak sütun matrisi elde edilir. Burada SWAP kapıları ayrı tutularak yazılmıştır. Ayrıca çıktının katsayısı olan  $\frac{1}{\sqrt{d^n}}$  yazılmamıştır.



Şekil 2.28. d seviyeli ve n girişli QFT devresinin girdi ve çıktıları

Kuantum bilgi işlemede kullanılan tüm kapılar tersinir özelliğe sahip-

tir. QFT devresi her ne kadar bir çok kapıdan oluşsa da bir birim olarak düşünüldüğünde kendisi de tersinir özelliğe sahiptir. Bu nedenle  $QFT^\dagger = QFT^{-1}$  ve  $QFT^{-1}QFT = I$  durumunu sağlar.

QFT devresinde birinci girişe bir Hadamard ve  $n - 1$  tane şartlı faz kayması kapısı uygulanır. İkinci girişe bir Hadamard ve  $n - 2$  tane şartlı faz kayması kapısı uygulanır. Diğer girişlere de bir hadamard ve her defasında bir azalan sayıda şartlı faz kayması kapısı uygulanır. Eğer devrenin giriş sayısı çift ise  $n/2$ , tek ise  $(n-1)/2$  adet SWAP kapısı gereklidir.

QFT kuantum bilgi işlemede Kuantum Faz tahmini algoritması, Shor'un çarpanlarına ayırma algoritması gibi bir takım algoritmalarda yer alır. Görevi bilgi işleme sürecini paralel işlemeye hazır hale getirmektir. Bazı durumlarda bu bir dizi Hadamard kapıları kullanılarak da yapılır. Devre girişine QFT operatörünün uygulanması ile bir dizi Hadamard kapısının uygulanması arasında başlangıç olarak bir fark yoktur.

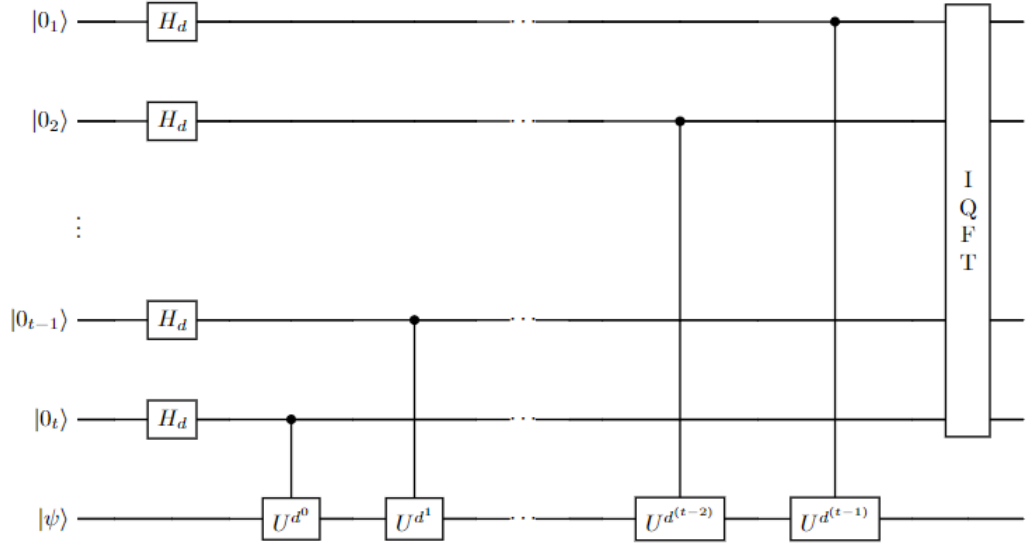
#### 2.4.2. Kuantum Faz Tahmini

Kuantum Faz Tahmini Algoritması QFT kullanımının iyi bir örneğidir.  $U$  uniter bir operatör olmak üzere,  $|\psi\rangle$  durumuna etki ederse  $U$  operatörünün fazını bularak özdeğerini bulmamızı sağlar. Bu yöntemle karmaşık hamiltonyenlerin özdeğeri tespit edilebilir.

$$U |\psi\rangle = e^{i\phi} |\psi\rangle \quad (2.66)$$

Şekil 2.29'da  $d$  seviyeli  $t$  girişli bir QPE (Kuantum Faz Tahmini) devresi görülmektedir. Birinci adımda Hadamard kapıları ile süperpozisyon durumu oluşturulup paralel işlemin önü açılır. Ardından şartlı kapılar ile işleme devam edilir ve son adımda IQFT operatörü kullanılarak sonuç elde edilir.  $|0\rangle$ 'a ayarlanmış  $t$  adet yardımcı girişin her birine Hadamard kapısının uygulanması QFT operatörünü tüm girişleri  $|0\rangle$  olan bir duruma uygulanması ile eşdeğerdir. Bu durum  $QFT |00\dots 0\rangle = H^{\otimes n} |0\rangle^{\otimes n}$  eşitliği ile gösterilebilir.

Bununla birlikte sıralanmamış  $N$  elemanlı bir veri tabanından istenen veriyi



Şekil 2.29. d seviyeli ve t girişli QPE devresi

$\sqrt{N}$  adımda bulmayı sağlayan arama algoritmasında da yer alır. Bahsi geçen bu algoritmalar kuantum bilgi işlemenin klasik bilgi işlemekten çok daha avantajlı olduğunu açık bir şekilde ortaya koyar.

## 2.5. QFT Tabanlı Aritmetik İşlemler

İki adet  $n$  bitden oluşan  $a$  ve  $b$  gibi iki sayımız olsun. Bu sayıların QFT tabanlı toplama işlemini gerçekleştiren devre Şekil 2.30'da görülmektedir. Herhangi bir  $d$  seviyeli kuantum sistemi kullanarak QFT tabanlı aritmetik işlem yapılacağıında bu sayılar önce mod- $d$ 'ye göre yazılır. Bu işlemle birlikte  $a = a_0d^{n-1} + a_1d^{n-2} + \dots + a_{n-1}d^0$  ve  $b = b_0d^{n-1} + b_1d^{n-2} + \dots + b_{n-1}d^0$  eşitliklerinden sonra  $|a\rangle = |a_0a_1\dots a_{n-1}\rangle$  ve  $|b\rangle = |b_0b_1\dots b_{n-1}\rangle$  gösterimleri ile  $n$  bitlik girişlere kodlanır. Özellikle toplama ve çarpma işlemlerinde elde olan sayıyı tutmak için  $|0\rangle$  ek giriş kullanılır. Bu giriş genelde yardımcı giriş olarak isimlendirilir.



Şekil 2.30. d seviyeli iki sayı için QFT tabanlı aritmetik işlem yapan genel devre

Birinci adımda yardımcı giriş ile birlikte  $a$  sayısını temsil eden  $n$  girişe QFT operatörü uygulanır. Bu sayede  $n+1$  giriş süperpozisyon durumuna sokularak paralel işlemeye hazır hale getirilir. Ardından şartlı faz kayması kapılarından oluşan toplayıcı parça kullanılarak  $b$  sayısı  $a$  sayısının da içinde olduğu süperpozisyon durumlarına eklenir. Son adımda IQFT uygulanarak hesaplama uzayından çıkılır ve ölçüm yapılır. Ölçümde okunan değer  $a$  ile  $b$  arasındaki aritmetik işlemin sonucudur (Draper, 2000; Yuan vd., 2023).

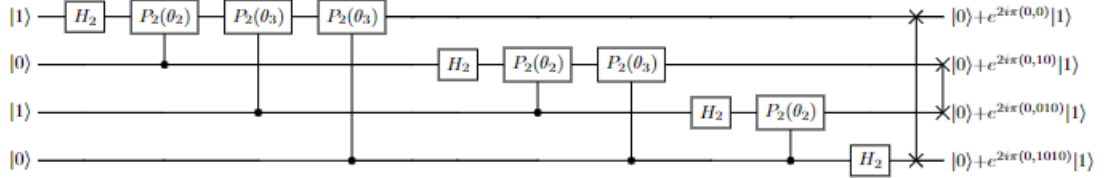


### 3. MATERYAL VE YÖNTEM

Bu bölümde, tez çalışmasında kullanılacak olan bazı yöntemler, kullanılacak olan paket programlar ve özel tanımlanmış kapılar tanıtılacaktır. Bölüm 2.4.1’de verilen genel QFT devresinin özel durumları hem kübit hem de kukuartlar için simüle edilecektir. Tez çalışmasında devre çizimleri için Latex’de tanımlı quantikz paketi kullanılmıştır (Kay, 2023). Bu paket kübit sistemler için önerilen devrelerin çizimleri için çeşitli araçlarla doludur. Kübitler için özel kapıların tanımlanmasına olanak sağlayan yapıdadır . Bu sayede ihtiyaç duyulan özel kapılar quantikz paketi kullanılarak da tasarlanabilir.

#### 3.1. Dört Kübitlik QFT Uygulaması

Bu kısımda örnek olarak dört kübitlik  $|1010\rangle$  durumuna QFT operatörünün uygulanması sonucu elde edilen çıktı tartışılacaktır. Şekil 2.28’de verilen genel devre  $|1010\rangle$  durumu için çıktıları ile beraber Şekil 3.1’de görülmektedir. Bu devrede 12 adet kapı kullanılmıştır.



Şekil 3.1. Dört kübit girişli QFT devresi

Bu devrenin çıktıları,

$$\begin{aligned}
 QFT |1010\rangle &= \frac{1}{2^4} (|0\rangle e^{2i\pi(0,0)} |1\rangle) \otimes (|0\rangle e^{2i\pi(0,10)} |1\rangle) \\
 &\otimes (|0\rangle e^{2i\pi(0,010)} |1\rangle) \otimes (|0\rangle e^{2i\pi(0,1010)} |1\rangle)
 \end{aligned} \tag{3.1}$$

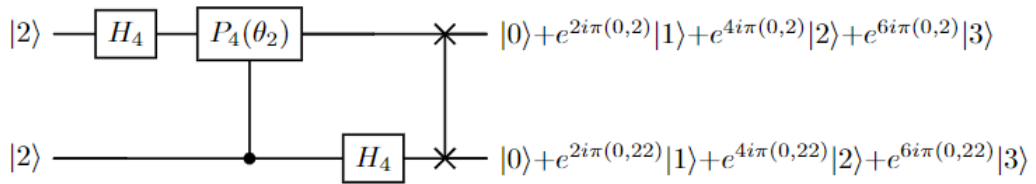
eşitliği ile yazılabilir. Bu eşitlikteki ara işlemler yapılırsa,

$$\begin{aligned}
QFT |1010\rangle = \frac{1}{2^4} (&|0000\rangle - e^{i\pi/4} |0001\rangle + i |0010\rangle - e^{i\pi/4} |0011\rangle \\
&- |0100\rangle + e^{i\pi/4} |0101\rangle - i |0010\rangle + e^{i\pi/4} |0111\rangle \\
&+ |1000\rangle - e^{i\pi/4} |1001\rangle + i |1010\rangle - e^{i\pi/4} |1011\rangle \\
&- |1100\rangle + e^{i\pi/4} |1101\rangle - i |1110\rangle + e^{i\pi/4} |1111\rangle) \quad (3.2)
\end{aligned}$$

sonucuna ulaşılır. Sonuçtan da anlaşılacağı üzere, dört girişli bir duruma QFT uygulandıktan sonra 16 olası durumun süperpozisyonu oluşur. Kuantum algoritmalarında bu adımdan sonra paralel işleme devreye girerek, klasik bilgi işlemede 16 adımda yapılan görev burada 1 adımda yapılır.

### 3.2. İki Kukuart Girişli QFT Uygulaması

3.1 bölümünde dört kubitlik bir durum üzerine QFT operatörünü uyguladığımızda 16 durumun süperpozisyonunun oluştuğunu gördük. Bu kısımda iki kukuart girişli bir duruma QFT uygulandı. Bu durum tercihen  $|22\rangle$  kabul edildi. Kubit tabanında temsil edilen  $|1010\rangle$  durumu ile kukuart tabanında temsil edilen  $|22\rangle$  durumu ondalık sistemde aynı sayıyı temsil etmektedir. Yani dört adet kubit ile temsil edilen bir bilgiyi iki adet kukuart ile temsil edebiliriz. Bu ise devre maliyetinde bir kazançta yol açar. Çünkü dört giriş üzerine uygulanan QFT için gerekli kapı sayısı ile iki giriş üzerine uygulanan QFT için gerekli olan kapı sayısı arasında kayda değer bir fark vardır. Şekil 3.2’de  $|22\rangle$  durumuna uygulanan QFT devresi ve çıktıları görülmektedir.



Şekil 3.2. İki kukuart girişli QFT devresi

Görüldüğü gibi bu devre sadece 4 adet kapı kullanılarak oluşturulmuştur. Bu devrenin çıktıları,

$$QFT |22\rangle = \frac{1}{4^2} (|0\rangle - |1\rangle + |2\rangle - |3\rangle) \otimes (|0\rangle - e^{i\pi/4} |1\rangle + i |2\rangle - e^{3i\pi/4} |3\rangle) \quad (3.3)$$

eşitliği ile yazılabilir. Ara işlemler yapılırsa,

$$\begin{aligned}
QFT |22\rangle = \frac{1}{4^2} (&|00\rangle - e^{i\pi/4} |01\rangle + i |02\rangle - e^{3i\pi/4} |03\rangle \\
&- |10\rangle + e^{i\pi/4} |11\rangle + i |12\rangle + e^{3i\pi/4} |13\rangle \\
&+ |20\rangle - e^{i\pi/4} |21\rangle + i |22\rangle - e^{3i\pi/4} |23\rangle \\
&- |30\rangle + e^{i\pi/4} |31\rangle - i |32\rangle + e^{3i\pi/4} |33\rangle) \quad (3.4)
\end{aligned}$$

sonucuna ulaşılır. Sonuçtan da anlaşılacağı üzere, iki kukuart girişli bir duruma QFT uygulandıktan sonra 16 olası durumun süperpozisyonu oluşur. Kuantum algoritmalarında bu adımdan sonra paralel işlem devreye girerek, klasik bilgi işlemede 16 adımda yapılan görev burada 1 adımda yapılır. Kübit yerine kukuartların kullanılması aynı görev için hem daha az bilgi birimine hem de daha az kapı gereksinimine ihtiyaç duyulduğunu gösterir.

### 3.3. Kuantum Hesaplama Simulatörleri

Kuantum devrelerinde girişler sütun matrisleri ile temsil edilirler. Örneğin beş girişli kubit tabanlı bir devrede girişi temsil eden matris  $2^5 \times 1$  boyutlu bir sütun matrisidir. Ardından bu giriş durumuna etki edecek kuantum kapıları da matrisler ile temsil edilir. Beş girişli bu devrede girişlerden sadece birine bile bir kuantum kapısı uygulansa bu kapı tensör çarpımı ile  $32 \times 32$  lik bir kare matrisine dönüştürülerek giriş üzerine etki eder. Çünkü bilinir ki matris çarpımı gerçekleşmesi için çarpılacak iki matrisin sütun ve satır sayılarının birbirine eşit olması gerekir. Bu örnek üç olası duruma sahip kutrit sistemler için beş giriş  $3^5 \times 1$  boyutlu bir sütun matrisi ile bir işlemcisi ise  $243 \times 243$  lük bir matris ile temsil edilir. Yine aynı durum 4 seviyeli kukuart sistemler için  $4^5 \times 1$  lik bir sütun matrisi ile bunun üzerine etki edecek bir kapı matrisi de  $1024 \times 1024$  lük bir kare matris ile temsil edilir.

Sadece beş girişli bir kuantum devre için bile bu boyutlardaki matrislerle uğraşmak çok zor iken giriş sayısının artması ile bu değerler üstel olarak artmaktadır. 30 kubit girişli bir kuantum devresinin girişi  $1073741824 \times 1$  boyutlu bir sütun matrisi ile temsil edilir. Kapı matrislerinin boyutları da bu değerlerin kare-

sine karşılık gelir. Sadece ardı ardına uygulanacak birkaç kapı olduğunda bile bu işlemi el yordamı ile yapmak neredeyse imkânsıza yakındır. Hal böyleyken çoklu girişe sahip devreleri kurup çalıştırmayı nasıl başarıyoruz. Bu konuda imdadımıza Mathematica, Python, Matlap gibi programlama dilleri yetişiyor. Bu programlar kullanılarak kuantum kapılarını temsil eden matrisleri tanımlanıp devre işleyişi ile birleştirildiğinde devrenin çıktıları elde edilir.

Gerçek bir kuantum bilgisayarı elbette bu matris temsillerini kullanarak kuantum algoritmaları gerçekleştiren devreleri sonuca ulaştırmıyor. Onlar gerçek kuantum sistemlerine elektromanyetik dalgalar göndererek kuantum hesaplama işlemini baştan sona kuantum mekaniksel gerçekleştiriyor. Tabii gerçek bir kuantum bilgisayara ulaşmak çok uzak olmasa da pek de yakın görünmüyor. Kuantum bilgi işleme süreçleri ve kuantum bilgi işleminin gerçekleşmesi multidisipliner bir konudur ve dünya genelinde büyük araştırma fonları bu konuya ayrılmaktadır.

Kuantum bilgi işlemede devreler tasarlayıp bunları test etmek amaçlı birkaç paket program bulunmaktadır. Bu programlar sayesinde yüksek boyutlu matris işlemlerinin zorluğundan kurtularak, yine kendi kullandığımız kişisel bilgisayarların işlem yapabilme kapasitesinin izin verdiği ölçüde devre tasarımları ve analizleri yapıp kuantum devrelerini test edebiliriz. Aşağıda bazı kuantum hesaplama simülatörleri listelenmiştir.

- Qiskit: IBM tarafından geliştirilen açık kaynaklı bir kuantum hesaplama platformudur. Python tabanlıdır ve kuantum devrelerini tasarlamak, simule etmek ve çalıştırmak için kullanılır.
- Cirq: Google tarafından geliştirilen bir kuantum hesaplama platformudur. Python dilini temel alır ve kuantum devrelerini oluşturmak ve simule etmek için kullanılır.
- QuTiP: Quantum Toolbox in Python (QuTiP), açık kaynaklı bir Python kütüphanesidir. Kuantum mekaniği ve kuantum bilgisayar simülasyonları için kullanılır. Kuantum işlemcilerin simülasyonunu yapabilir.
- Microsoft Quantum Development Kit: Microsoft'un kuantum hesaplama geliştirme kitidir. Q plus adında özel bir programlama dilini kullanır ve

kuantum simülasyonları için araçlar sunar.

- PennyLane: Kuantum bilgisayarlar ve kuantum hesaplama ile ilgili bir açık kaynaklı yazılım platformudur. PennyLane, kuantum devreleri tasarlamaya, optimize etmeye ve simüle etmeye olanak tanır. Python tabanlı bir kütüphanedir ve kuantum makineleri ile etkileşim kurmaya ve çeşitli kuantum algoritmalarını çalıştırmaya yardımcı olur. PennyLane, kuantum hesaplama alanında araştırma yapmak ve geliştirmek isteyenler için güçlü bir araçtır. Bu platform, açık kaynaklıdır ve sürekli olarak geliştirilmektedir, bu nedenle kuantum bilgisayarlarla ilgili çalışmalarınız için kullanılabilir.

Bu çalışmada kubit sistemler için önerilen bazı tasarımlarda Qiskit kütüphanesi kullanılmıştır. Kürit ve kuart tabanlı sistemler için henüz bir kütüphane mevcut değildir. Bunlar için de yine python'un sahip olduğu numpy kütüphanesinden faydalanarak devrelerin doğruluğu test edilebilir.

### 3.4. IBM Kuantum Bilgisayarı

Gerçek bir kuantum bilgisayar veri tipinin de kuantum algoritmalarının da kuantum mekaniksel olduğu bir durumdur. Bu durum Şekil 3.3'de hem veri tipinin hem de algoritma tipinin kuantum olduğu satır ve sütunun kesiştiği bölümde görülmektedir.

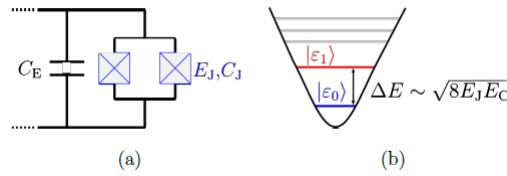
		Algoritma Tipi	
		Klasik	Kuantum
Veri Tipi	Klasik	Hem veri hem algoritma Klasik	Veri Klasik Algoritma Kuantum
	Kuantum	Veri Kuantum Algoritma Klasik	Hem veri hem algoritma Kuantum

Şekil 3.3. Bilgi işleme türleri

Ancak günümüzde gelinen nokta verinin klasik algoritmanın kuantum olduğu

durumudur. Bunu bölüm 3.3’de adı geçen simulatörler ile gerçekleştirmekteyiz. Fakat IBM’in IBM-QE adında ( IBM Quantum Experience) kullanıma sunduğu, yapay atom olarak adlandırılan ve bilginin transmon kubitlere kodlanarak işlendiği bir kuantum bilgisayarı vardır.

IBM-QE süperiletken bir kubit oluşturmak için iki adet josephson bağlantısı ve harici bir kapasitans kullanır. Bu sistem bir kuantum mekaniksel sistemdir ve kesikli enerji seviyelerine sahiptir. İlk iki enerji seviyesi taban ve birinci uyarılmış durum olarak dikkate alındığında bunlar sırasıyla  $|0\rangle$  ve  $|1\rangle$  durumlarına karşılık gelir (Alves, 2020) . Böyle iki seviyeli bir kuantum sistemi kullanılarak bir kubit oluşturulmuş olur.



Şekil 3.4. a) Transmon kubit b) Enerji seviyeleri (Alves, 2020)

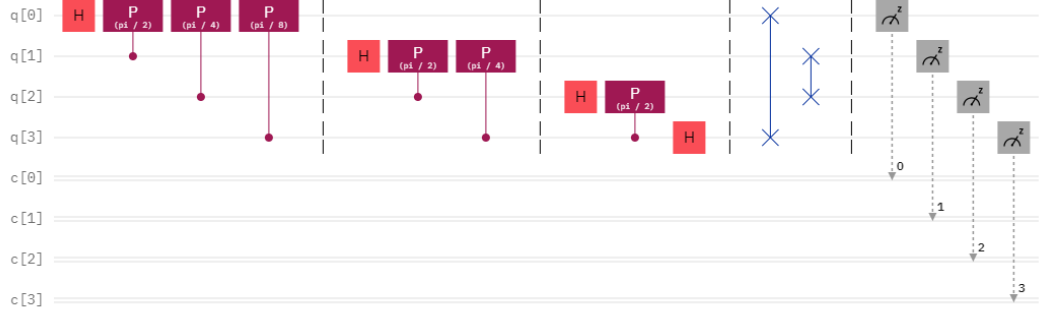
IBM-QE bu alt yapıyı kullanarak ücretli ve ücretsiz olmak üzere kuantum hesaplama sistemlerini kullanıma açmıştır. Ücretsiz olan hesaplayıcılar aşağıda listelenmiştir.

- ibm-brisbane , 127 kubit
- ibm-perth , 7 kubit
- ibm-lagos , 7 kubit
- ibm-nairobi , 7 kubit

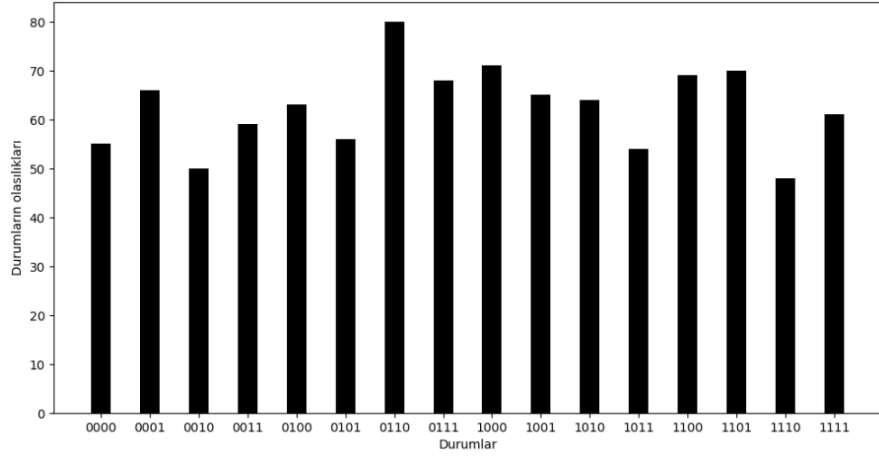
Bu kuantum hesaplayıcılarda çok sınırlı sayıda kuantum kapıları tanımlanmıştır. Bu nedenle tanımlı olmayan kapıları içeren ya da birbirinden uzak haritalanmış kubitler arasında kontrollü kapılar uygulanırken yüksek oranlı hatalar verebilmektedir. Bununla birlikte qiskit kütüphanesi kullanarak çalışan simulatörlerde bulunmaktadır. Bunların arasında simulatör-stabilizer isimli olan 5000 kubitli bir hesaplayıcı da bulunmaktadır.

### 3.5. IBM Simulator-mps Kullanılarak Dört Kubit Girişli Bir QFT Uygulaması

Şekil 3.5’de  $|0000\rangle$  girdisine QFT’nin uygulandığı kuantum devre görülmektedir. Bu çizim ve hesaplaması IBM’in 100 adet girişe izin veren simulator-mps isimli simülasyon paketine yaptırılmıştır. QASM kodları EK 1’de verilmiştir.



Şekil 3.5. IBM simulator-mps’de çizilen dört girişli QFT devresi



Şekil 3.6. 4 kubit girişli QFT devresinden elde edilen çıktılar

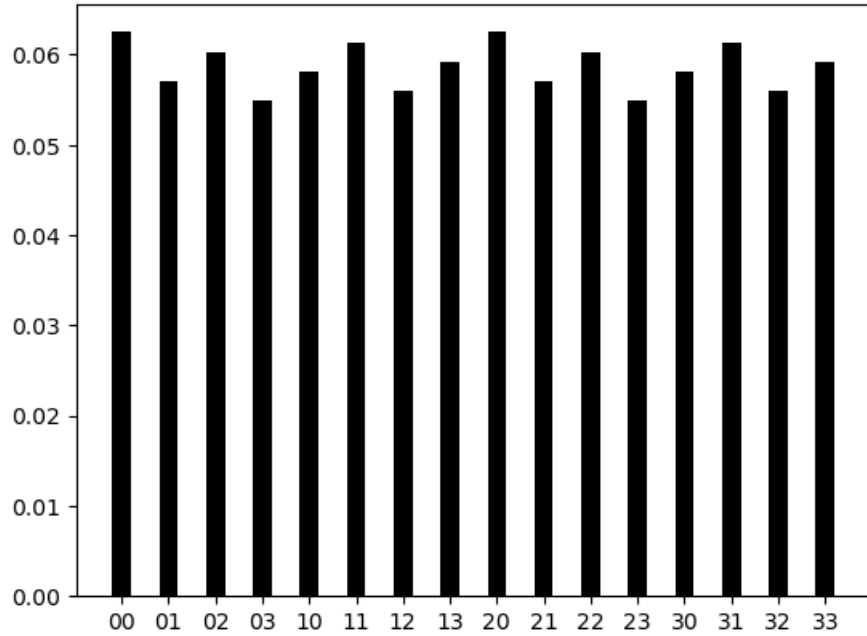
Şekil 3.6’da bu devrenin çıktılarının sütun grafiği görülmektedir. Bu sütun grafiğinden anlaşılacağı üzere,  $QFT|0000\rangle$  işleminin sonucu olası 16 durumun bir süperpozisyonudur. Bu adımdan sonra yapılacak işlem için paralel işlemeye hazır hale getirilmiştir.

### 3.6. Python İle İki Kukuart Girişli QFT Uygulaması

Kudit sistemler için herhangi bir kuantum hesaplama simülasyonu bulunmamaktadır. Ancak Python programının numpy kütüphanesi kullanılarak herhangi bir d seviyeli kuantum sistemlerinin sütun matrisleri tanımlanabilir. QFT için verilen matematiksel eşitlik kullanılarak da QFT operatörünün matris temsili elde edilebilir. QFT için matematiksel eşitlik, eşitlik 3.5'de görülmektedir (Nielsen 2011).

$$QFT |x\rangle = \frac{1}{d^{n/2}} \sum_{y=0}^{d^n-1} e^{2i\pi xy/d^n} |y\rangle \quad (3.5)$$

Bu eşitlikten yola çıkarak QFT operatörünün matris temsilinin elemanları bulunur. Herhangi bir d seviyeli ve n girişli bir kuantum sistemi için yazılan bu ilkel simülasyonun kodları Ek 2'de verilmiştir. 3.2 bölümünde yapılan  $QFT |22\rangle$  işlemi bu sistemde çalıştırıldığında 16 adet durumunun süperpozisyonu elde edilmiş ve sütun grafiği çizdirilmiştir. Bu şekil 3.7'de görülmektedir.



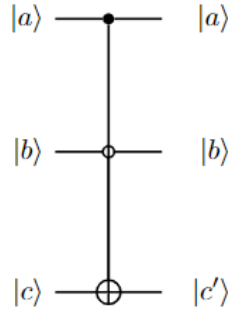
Şekil 3.7. İki kukuart girişli QFT devresinden elde edilen çıktılar

Bu simülasyon için yazılan kodlardan anlaşılacağı gibi işlem kuantum kapılarının matris temsilleri üzerinden ilerlememektedir. Eşitlik 3.5 kullanılarak QFT'nin matris temsili oluşturulmuştur. Genelleştirilmiş QFT devresinin matris tem-

sili elde edildikten sonra, girişi temsil eden sütun matrisi ile matris çarpımı gerçekleştirilir. Oradan elde edilen katsayılar kullanılarak da durumların genliklerini veren grafik çizilir.

### 3.7. Özel Durumlara Bağlı Bazı Kuantum Kapılar

İki giriş üzerine etkiyen kontrollü kapılardan CNOT kapısını ve üç giriş üzerine etki eden Toffoli (CCNOT) kapılarını tanımlamıştık. Ancak bilgi işleme sürecinde özel olarak kontrol girişleri aynı şartı gerektirmeyen ve üçten fazla kontrollü kapılar içeren kuantum devreler de tasarlanabilir. Örneğin kontrol bitlerinden biri  $|0\rangle$  diğeri  $|1\rangle$  olan iki kontrollü bir NOT kapısı Şekil 3.8’de ve bu kapının doğruluk tablosu Tablo 3.1’de görülmektedir. Bu tablodan da görüldüğü gibi 1 ve 0 kontrollü CCNOT kapısı sadece iki durum üzerinde değişiklik yapmaktadır. Bu durumlardan biri  $CCNOT |100\rangle = |101\rangle$ , diğeri  $CCNOT |101\rangle = |100\rangle$ ’dır.



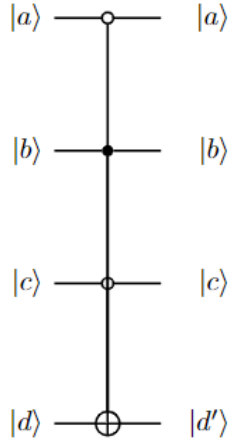
Şekil 3.8. Birinci kontrolü  $|1\rangle$  ikinci kontrolü  $|0\rangle$  olan CCNOT Kapısı

Tablo 3.1.  $|1\rangle$  ve  $|0\rangle$  Kontrollü NOT kapısı

$ a\rangle$	$ b\rangle$	$ c\rangle$	$ a\rangle$	$ b\rangle$	$ c'\rangle$
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	1
1	1	0	1	1	0
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	0
1	1	1	1	1	1

Benzer şekilde kontrol bitleri sırasıyla  $|0\rangle$ ,  $|1\rangle$  ve  $|0\rangle$  olan üç kontrollü CCCNOT kapısı Şekil 3.9’da görülmektedir. Bu kapı  $CCCNOT |0100\rangle = |0101\rangle$  ve  $CCCNOT |0101\rangle = |0100\rangle$  dönüşümlerini gerçekleştirir. Çoklu ve isteğe bağlı

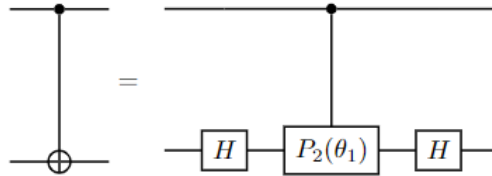
kontrollere sahip kuantum kapıları çoğaltılabilir. Buradaki esas nokta tüm kontroller sağlandıktan sonra hedef girişe söz konusu kapıyı uygulamaktır.



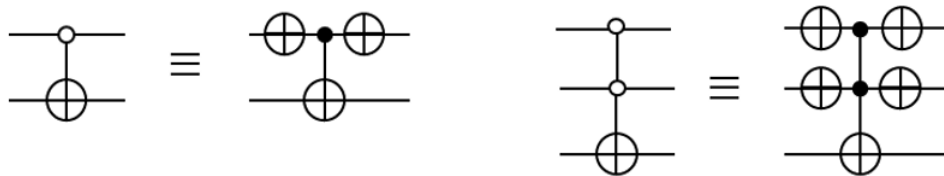
Şekil 3.9. Kontrolleri sırasıyla  $|0\rangle$  ikinci kontrolü  $|1\rangle$  ve  $|0\rangle$  olan CCCNOT Kapısı

### 3.8. Bazı Özdeşlikler

Bu kısımda kuantum kapılarından özellikle CNOT kapısına özdeş durumlardan bahsedilecektir. Şekil 3.10'da CNOT kapısının Hadamard ve kontrollü faz kayması kapıları ile oluşturulmuş bir özdeşliği görülmektedir.



Şekil 3.10. CNOT kapısının Hadamard ve kontrollü faz kayması kapıları ile oluşturulması



Şekil 3.11.  $|0\rangle$  kontrollü CNOT ve toffoli kapılarının özdeşlikleri

Bu çalışmada tasarlanan devrelerin test edilmesi için IBM'in gerçek kuantum bilgisayarı ve Qiskit kütüphanesi kullanıldı. Bu ortamlardan Qiskit özel bir tanımlama yöntemi ile kontrol girişinin  $|0\rangle$  olması durumunda NOT uygulayan

bir kontrollü kapı tanımlamasına izin verirken, IBM'in gerçek kuantum bilgisayarı kontrol bitinin yalnızca  $|1\rangle$  olduğu duruma NOT uygulayan kapıya izin vermektedir. Bu nedenle ilave NOT kapıları ve CNOT kapısı kullanarak kontrol biti  $|0\rangle$  olan NOT ve toffoli kapıları oluşturuldu. Bu özdeşlikler Şekil 3.11'de görülmektedir.



## 4. BULGULAR VE TARTIŞMA

Bu tez çalışmasının konusu QFT tabanlı aritmetik işlemlerin geliştirilmesi ve uygulamalarıdır. Bu bölümde ilk kez aynı anda ikiden fazla sayı arasında QFT tabanlı toplama işlemi gerçekleştiren kuantum devre tasarımı sunulacaktır. Benzer devre mimarisi kullanılarak QFT tabanlı çıkarma işlemi yapan devre önerisi de yapılacaktır. Ardından ikiden fazla sayı arasında toplama işlemi yapan QFT tabanlı toplama devresi kullanılarak çarpma işlemi gerçekleştiren QFT tabanlı devre tasarımı verilecektir. Bölme devresi için de ardı ardına QFT tabanlı çıkarma işlemi yapan devre ile birlikte bu devrede kullanılacak olan kuantum sayaç ve karşılaştırıcı devreler tanıtılacaktır. Son olarak klasik bilgi işlemede bulunan NAND kapısının QFT tabanlı devre mimarisi ve temel düzeyde, sadece QFT tabanlı toplama devresine dayalı, seçilen işleme göre istenildiğinde toplama, istenildiğinde de girdilerin NAND sonucunu veren QFT tabanlı bir qALU tasarımı tanıtılacaktır. Tasarımı verilen tüm bu devrelerin simülasyonları gerçekleştirilecektir.

### 4.1. QFT Tabanlı Seri Toplama Devresi ve Kübit-Küdit Kıyaslaması

Bölüm 2.5’de a ve b gibi iki sayının QFT tabanlı toplama işlemi gerçekleştiren kuantum devre tanıtılmıştı. Burada ilk kez 2’den fazla sayının QFT tabanlı toplama işlemi tek bir devre üzerinde, her defasında QFT ve IQFT kullanmadan gerçekleştiren kuantum devre sunulmuştur. QFT tabanlı toplama işlemi gerçekleştiren devrelerde toplama işlemi gerçekleştiren kısmı şartlı faz kayması kapılarından oluşur. Ayrıca sadece iki sayı arasında toplama işlemi gerçekleştiği için toplama sırasında elde olan sayıyı tutmak için bir tane yardımcı giriş kullanılmıştır(Draper, 2000; Perez ve Escartin, 2014). Ancak burada önerilen devre 2’den fazla sayının toplama işlemi tek bir devrede gerçekleştireceği için toplama işleminde elde olan sayı adedi artmaktadır. Örneğin  $(11)_2 + (10)_2 = (101)_2$  işleminde 1 adet yardımcı girişe gerek duyulurken  $(11)_2 + (10)_2 + (01)_2 + (11)_2 = (1001)_2$  işleminde 2 adet yardımcı girişe ihtiyaç duyulur. Sayı adedi arttığında yardımcı giriş sayısı da artmaktadır.

Öncelikle,  $d$  seviyeli bir kuantum sistemine kodlanmış  $N$  adet sayının toplama işlemini gerçekleştirecek devrede yardımcı giriş sayısını veren bağıntı ilk defa bu çalışmada türetilmiştir. Bu bağıntı aşağıda görülmektedir.

$$t = \log_d N \quad (4.1)$$

Her biri  $n$  bitten oluşan  $N$  adet sayının toplama işlemini gerçekleştiren devre Şekil 4.1'de verilmiştir. Devreye dikkatlice bakıldığında ilk adımda  $t$  tane  $|0\rangle$ 'a ayarlanmış yardımcı giriş ile  $a_1$  sayısının kodlandığı  $n$  adet bit olmak üzere toplamda  $n+t$  adet girişe QFT uygulanır. Ardından  $a_2$ 'den  $a_N$ 'ye kadar  $N-1$  defa toplayıcı kısım uygulanır. Son adımda QFT uyguladığımız hat üzerinden gelen durumlara IQFT uygulanarak sonuç elde edilir. Şekil 4.1'de Toplayıcı olarak isimlendirilen devre parçalarının iç yapısı Şekil 4.2'de görülmektedir.

Şekil 4.1'deki devre mimarisi  $d$  seviyeli herhangi bir kuantum sistemi için geçerlidir. Şekil 4.2'deki girişlerde  $|0\rangle_1$ 'den  $|a_{n-1}\rangle$ 'e kadar olan kısım Şekil 4.1'deki ilk 2 hattı temsil etmektedir.  $|b_0\rangle$ 'dan  $|b_{n-1}\rangle$ 'e kadar olan kısım da toplanacak sayıyı temsil etmektedir.  $|b_0\rangle$ 'dan  $|b_{n-1}\rangle$ 'e kadar olan girişler hem ikinci sayıyı hem de kontrol girişlerini temsil ederken faz kayması kapıları QFT uygulandıktan sonra ortaya çıkan süperpozisyon durumları üzerine etki eder.

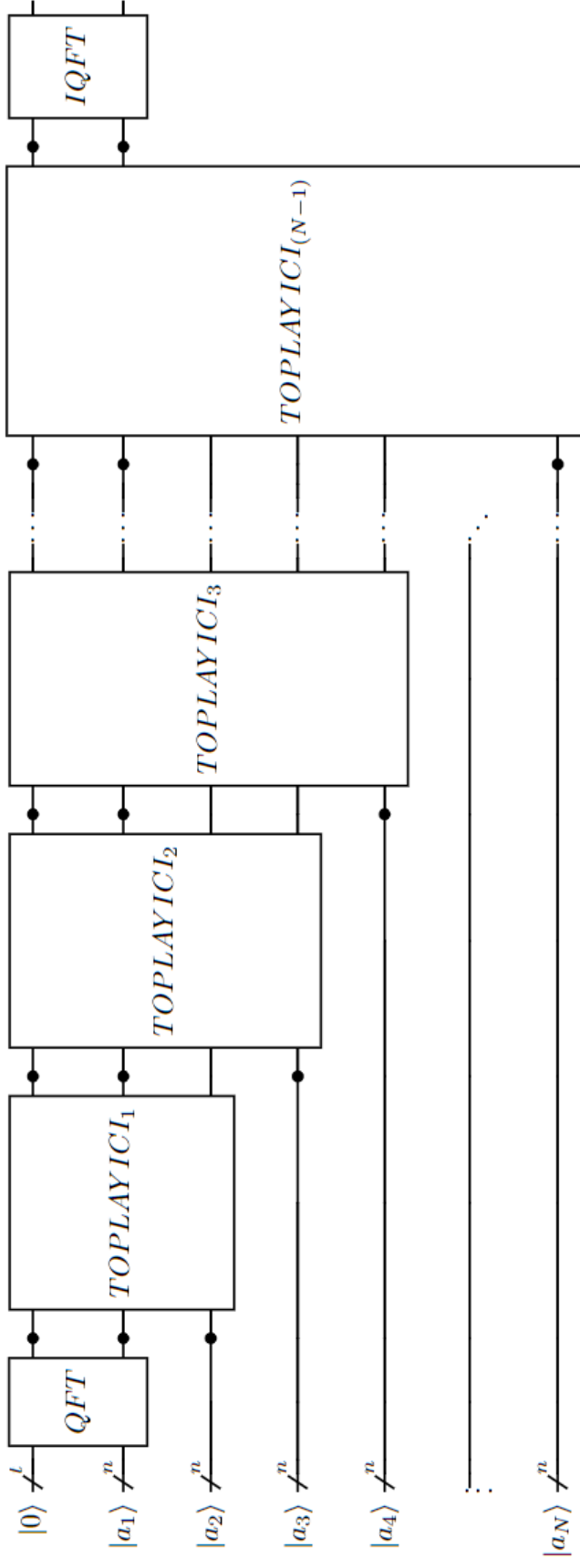
Genelleştirilmiş toplama devresinde  $N$  adet  $n$  bitlik sayı arasında toplama işleminde bir defa QFT ve IQFT uygulanır. Dolayısıyla  $N-2$  adet QFT ve IQFT'nin maliyetinden kurtulmuş olunur. Eşitlik 4.1 kullanılarak  $t$  belirlendikten sonra, tasarlanacak devre için kapı sayısı, eğer  $t+n$  çift ise,

$$KapıSayısı = (1+N)n\left[\frac{1+n}{2} + t\right] + t^2 + 2t + n \quad (4.2)$$

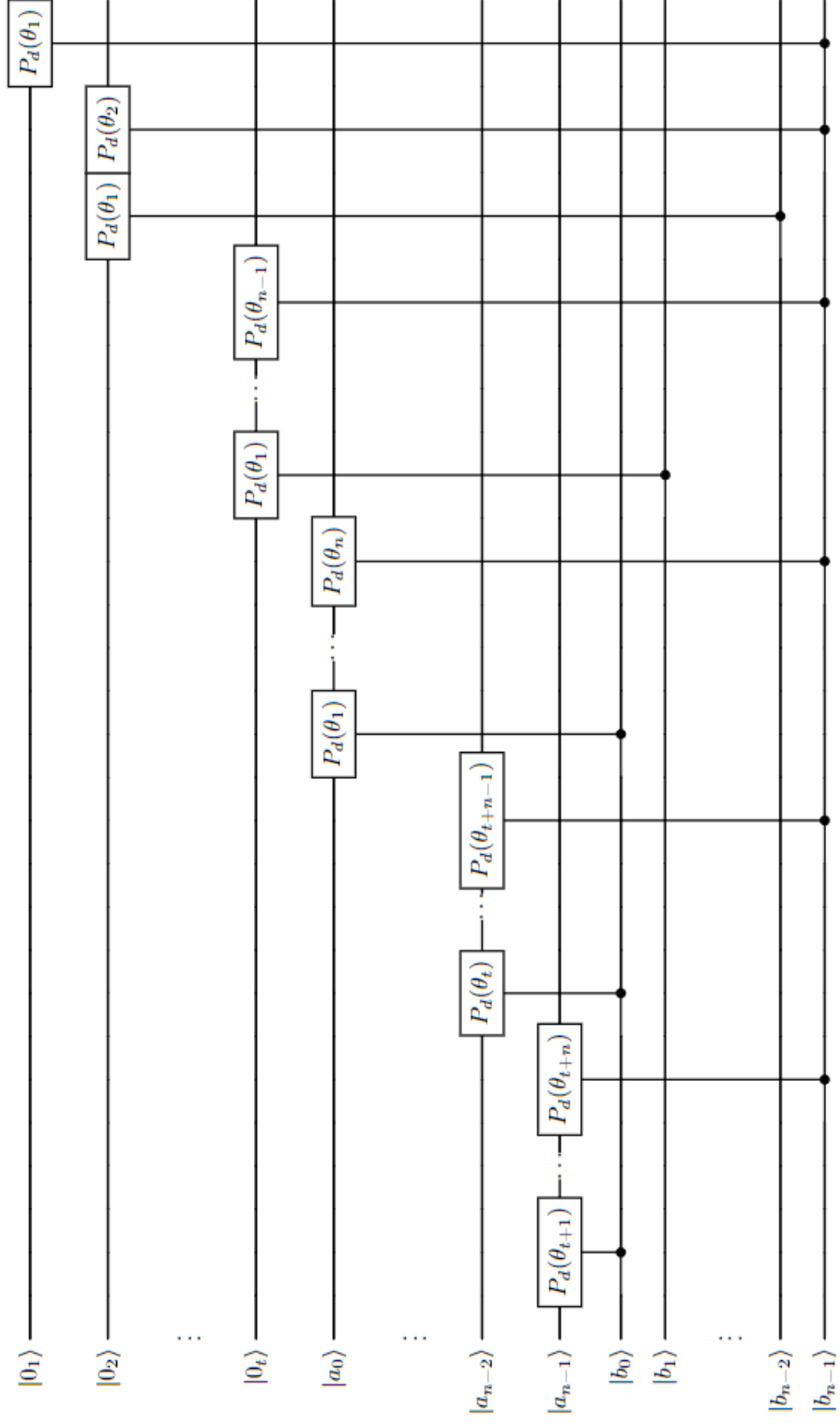
Eğer  $t+n$  tek ise,

$$KapıSayısı = (1+N)n\left[\frac{1+n}{2} + t\right] + t^2 + 2t + n - 1 \quad (4.3)$$

eşitlikleri ile bulunur. Bu iki eşitlik QFT ve IQFT operatörlerinde bulunan SWAP kapılarının farklı olmasından kaynaklanır.



Şekil 4.1. N adet n bitlik sayının QFT tabanlı seri toplayıcı devresi



Şekil 4.2. Toplayıcı kısım

#### 4.1.1. Kübit Kullanılarak Dört Adet İki Bitlik Sayının QFT Tabanlı Toplama İşlemi

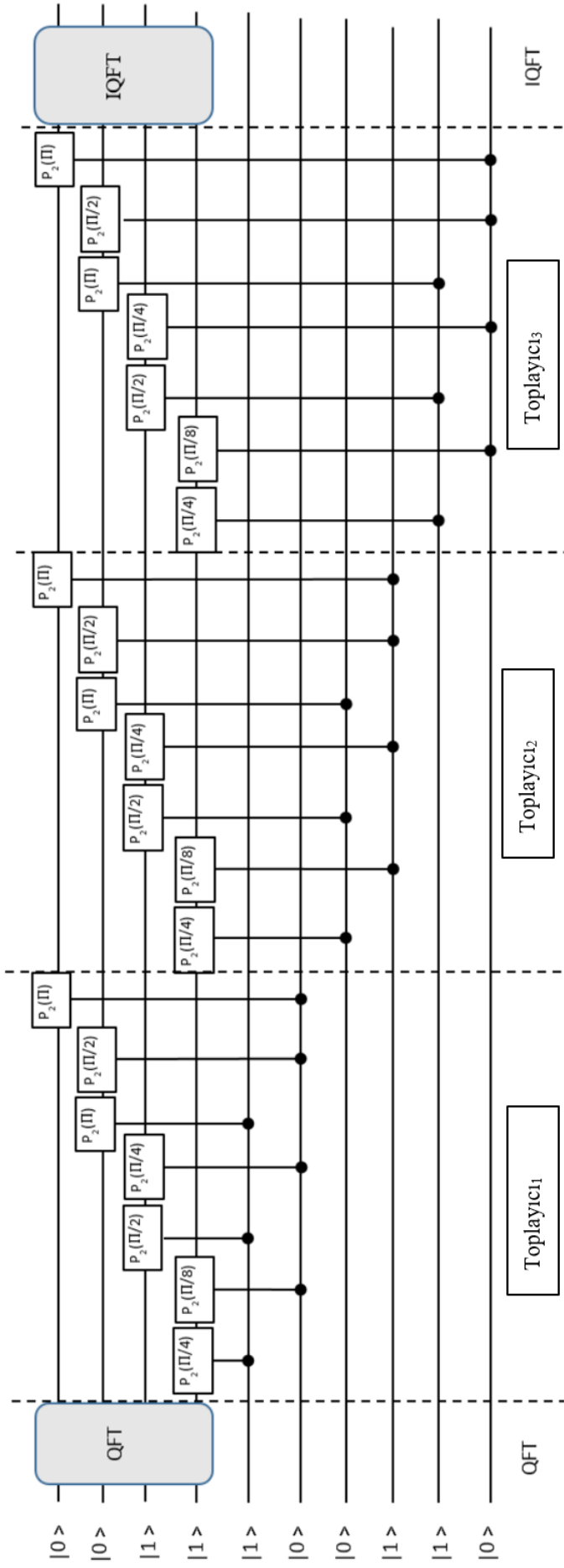
Ondalık sayı sisteminde  $3+2+1+2$  işleminin sonucu 8'dir. Bu işlemi kubitler kullanarak yapmak için öncelikle her bir sayı ikili sayı sisteminde yazılmalıdır. Bu durumda  $3 = (11)_2$ ,  $2 = (10)_2$  ve  $1 = (01)_2$  halinde her bir sayı iki basamaklı sayı ile temsil edilir. Yani her bir sayı iki bitlik girişe sahiptir. Tüm işlemi, sonucu ile beraber ikili sayı sisteminde yazarsak aşağıdaki gibi bir durum elde ederiz.

$$(11)_2 + (10)_2 + (01)_2 + (10)_2 = (1000)_2 \quad (4.4)$$

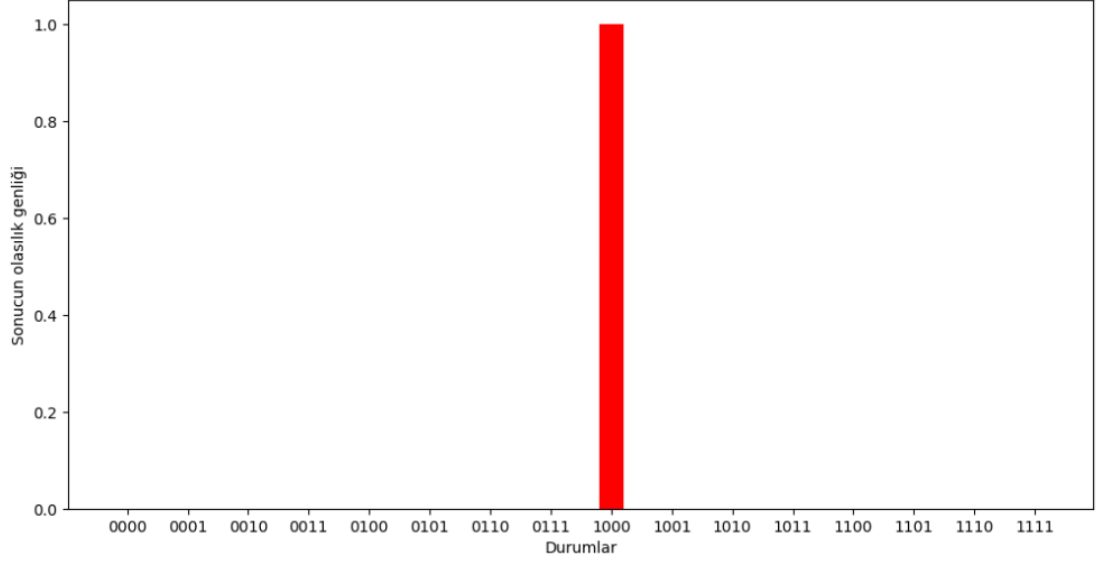
Her bir sayı iki bit ile temsil edildiği için ve  $d = 2$  için 4 adet sayının toplama devresinin tasarımında Eşitlik 4.1 kullanılarak 2 adet yardımcı girişe gerek duyulduğu görülür. Ayrıca Eşitlik 4.4'de işlemin sonucunun 4 basamaklı olması ilk sayıyı temsil eden iki girişe ilave olarak 2 giriş daha kullanılması gerektiğini doğrular.

Şekil 4.3'de görüldüğü gibi ilk sayı olan 3'ü temsil eden  $(11)_2$  sayısı 2 tane yardımcı giriş ile birlikte kubitlere kodlandığında  $|0011\rangle$  durumu ortaya çıkar ve buna QFT operatörü uygulanır. Bu işlem sonucunda 16 durumun süperpozisyonu ortaya çıkar. Diğer üç sayı da ikişer kubitte kodlanarak devre on girişli hale gelir. İkinci sayı olan 2'yi temsil eden  $|10\rangle$  girişleri kontrol bitleri olmak üzere şartlı faz kayması kapıları QFT işlemi sonucu elde edilen süperpozisyon durumları üzerine etki eder. Bu işlem  $T_{\text{oplayıcı}_1}$  olarak adlandırılmıştır. Benzer şekilde  $|01\rangle$  ve  $|10\rangle$  girişleri de kontrol girişleri olmak üzere ilk 4 hat üzerinde bulunan çıktılar üzerine uygulanır. Bu işlemler sırasıyla  $T_{\text{oplayıcı}_2}$  ve  $T_{\text{oplayıcı}_3}$  olarak isimlendirilmiştir. Son adımda ilk 4 giriş üzerine IQFT işlemi uygulanarak sonuç elde edilir.

Bu devrenin test edilmesi el yordamıyla çok zordur. Her adımda  $1024 \times 1024$  boyutlu kare matrislerle işlem yapmak gerekir. Bu nedenle devrenin test aşaması IBM'in ücretsiz kullanıma sunduğu simülörlerden 100 kübitlik *simulator – mps* kullanılarak gerçekleştirilmiştir. Bu işlemin QASM kodları Ek 3'te verilmiştir. Bu işlem için simülörden elde edilen sonucun sütun grafiği Şekil 4.4'de görülmektedir.



Şekil 4.3. Dört adet iki bitlik sayımm,  $d=2$  için QFT tabanlı toplama devresi



Şekil 4.4. Dört adet iki bitlik sayının  $d = 2$  için QFT tabanlı toplama işleminin sonucunun *ibm – mps*'den elde edilen sütun grafiği

Grafikte net bir şekilde sonucun doğruluğu görülmektedir. Eğer bu işlem gerçek bir kuantum bilgisayarda yapılsaydı diğer değerler üzerinde de gürültü seviyesinde sonuçlar görülebilirdi. Çünkü IBM'in kullandığı kubit düzeninde engellenemeyen bazı etkileşimler buna sebep olurdu. Bu sonuç matris tabanlı işlem yapan bir simulatorde gerçekleştirildiği için sonuç gürültü olmaksızın hesaplanmıştır. Ayrıca IBM'in kullanıma sunduğu gerçek kuantum bilgisayarında henüz şartlı faz kayması gibi kapılar gerçek bir fiziksel sistem ile temsil edilemediği için bu hesaplama gerçek bir kuantum bilgisayar kullanılarak hesaplanamamıştır.

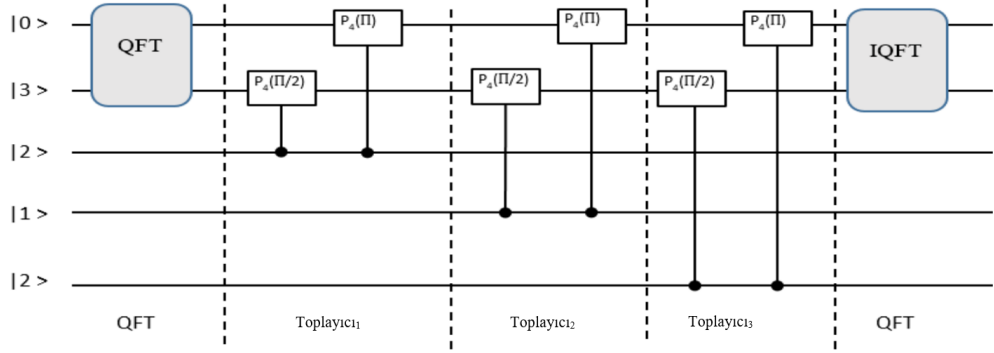
Bu devrede her sayı iki bit ile temsil edildiği için  $n = 2$  ve iki adet yardımcı giriş kullanıldığı için  $t = 2$ 'dir ve  $t + n = 4$  olduğundan Eşitlik 4.2 kullanılarak tüm devrede toplamda 45 adet kuantum kapısının kullanıldığı görülmektedir.

#### 4.1.2. Kukuart ( $d = 4$ ) Kullanılarak Dört adet bir Bitlik Sayının QFT Tabanlı Toplama İşlemi

Bu bölümde, 4.1.1. bölümünde kullanılan sayıları tekrar kullanarak aynı toplama işlemi kukuartlar kullanılarak gerçekleştirilmiştir. 3,2 ve 1 sayıları dörtlük sayı sisteminde yazıldığında yine aynı değerleri alır. Onluk sistemde  $3+2+1+2$  işlemi, sonucu ile birlikte dörtlük sayı sisteminde aşağıdaki gibi yazılır.

$$(3)_4 + (2)_4 + (1)_4 + (2)_2 = (20)_4 \quad (4.5)$$

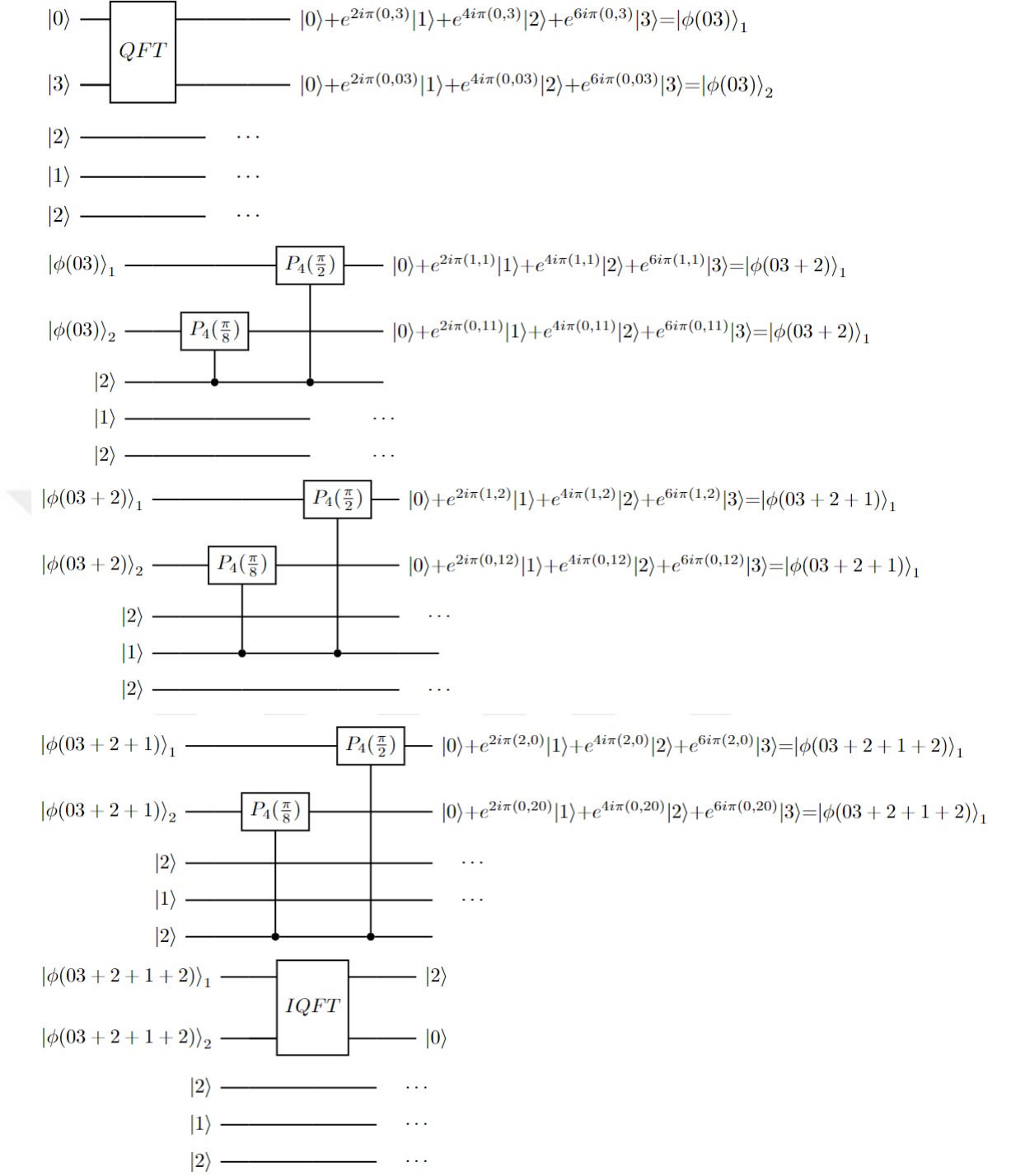
Eşitlik 4.5’den görüldüğü gibi her sayı bir giriş ile temsil edilir. Ayrıca 4 adet sayının toplamı  $d = 4$  sistemi kullanılarak hesaplanacaksa, bu işlemi yapacak devre için Eşitlik 4.1 kullanılarak 1 tane yardımcı girişe ihtiyaç duyulduğu görülür. Bu işlemi gerçekleştiren QFT tabanlı toplama devresi şekil 4.5’de görülmektedir.



Şekil 4.5. Dört adet bir bitlik sayının  $d = 4$  için QFT tabanlı toplama devresi

İlk adım olarak  $|03\rangle$  girişine QFT operatörü uygulanır. QFT operatörü uygulandıktan sonra elde edilen çıktılar Şekil 4.6’da görülmektedir. Ardından 2., 3. ve 4. sayılarını temsil eden girişler toplayıcı kısımlar kullanılarak eklenir. Son adımda IQFT uygulanarak sonuç elde edilir. Bu devrede toplam 14 kapı kullanılmıştır. Kullanılan kapı sayısı Eşitlik 4.2 ile de hesaplanabilir. Bu işlem kubitler kullanılarak 45, kuartlar kullanılarak 14 kapı ile gerçekleştirilmiştir.

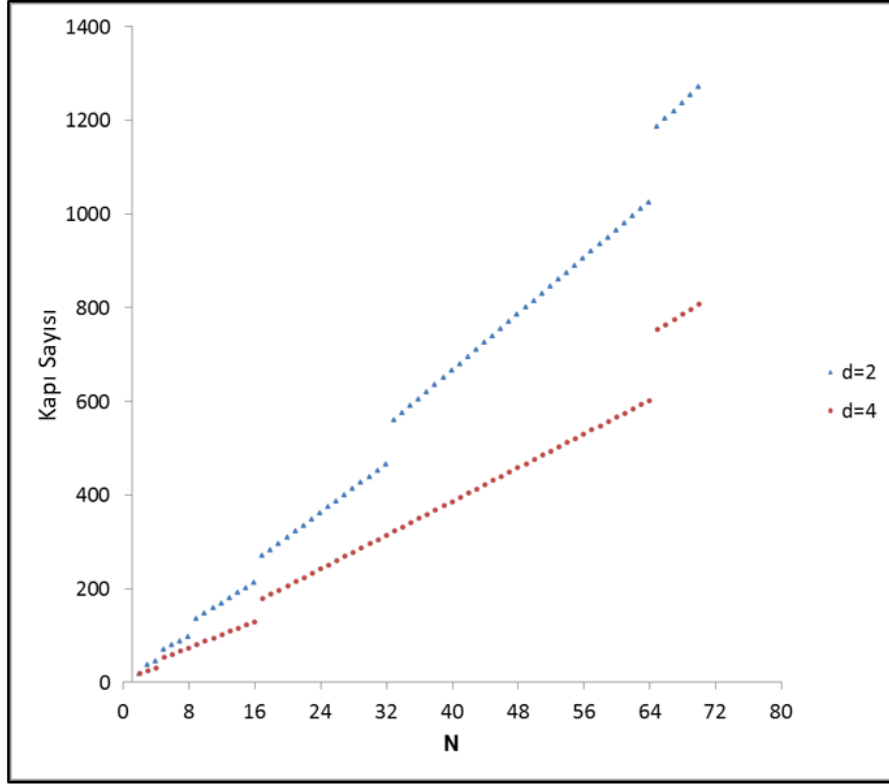
Bölüm 4.1.1’de ve bu bölümde ele alınan iki örnekte de doğru sonuçlar elde edilmiştir. Sayı adeti aynı olmak koşulu ile kubitlerin yerine kuartların kullanılması kapı gereksiniminde kayda değer bir avantaj sağlamaktadır. Bu maliyet azalmasını bu kısımda önerdiğimiz gibi sadece devrenin başında ve sonunda bir defa QFT ve IQFT kullanarak gerçekleştirdiğimizde maliyetin çok daha fazla düşmesi avantajına yol açar. Şekil 4.7’de  $N$ ’ye bağlı kapı sayısı gereksinimini veren grafik görülmektedir. Grafikte yatay eksen Sayı adetini, dikey eksen de o toplama işlemi gerçekleştirmek için gerekli olan kapı sayısını vermektedir.



Şekil 4.6. Şekil 4.5'deki devrenin işleyişi

## 4.2. QFT Tabanlı Çıkarma Devresi

QFT tabanlı çıkarma işlemi yapan kuantum devreler yapısal olarak QFT tabanlı toplama işlemleri yapan devrelere benzetilerek tasarlanabilir. Bu bölümde ilk olarak  $a > b$  olmak üzere  $a - b$  işlemini gerçekleştiren QFT tabanlı kuan-



Şekil 4.7. Kübit ve kukuartlar için N ye bağlı kapı sayısı değişimi

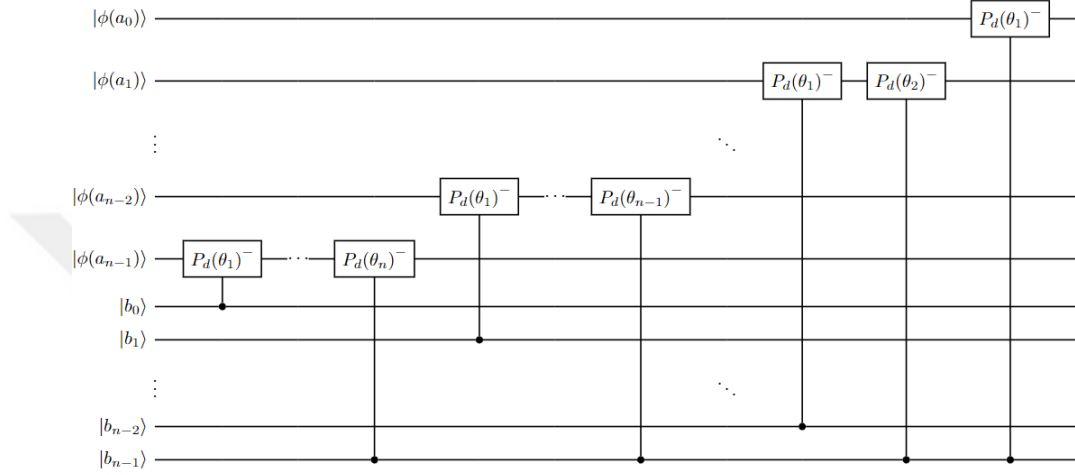
tum devre tasarısı verilecektir. Ardından sonuç negatif olmamak şartı ile birinci sayıdan N tane sayının çıkarılmasını gerçekleştiren devre tasarımı verilecektir. Sonrasında  $a < b$  olmak üzere negatif sonuçlu  $a - b$  devresinde negatif sonucun nasıl işleneceğinin üzerinde durulacak ve devre tasarımı verilecektir. Son olarak da toplama ve çıkarma işlemlerini tek bir devre üzerinde gerçekleştiren devre tasarımı sunulacaktır.

#### 4.2.1. $a > b$ için $a - b$ İşlemini Gerçekleştiren QFT Tabanlı Çıkarma İşlemi

$a > b$  olmak üzere  $a - b$  işlemini gerçekleştirmek için  $a$  sayısının kodlandığı bit dizisinden oluşan girişe QFT operatörü uygulandıktan sonra çıkarıcı olarak tanımlanan ve şartlı faz kayması kapılarının terslerinden oluşan kapı seti ile  $b$  sayısı devrenin işleyişine kontrol kubitleri olarak dahil edilir. Son adımda QFT uygulanan kısma IQFT operatörü uygulanarak sonuç elde edilir. Şekil 4.8'de çıkarıcı olarak isimlendirilen kısım Şekil 4.9'da görülmektedir.



Şekil 4.8. QFT tabanlı çıkarıcı devre



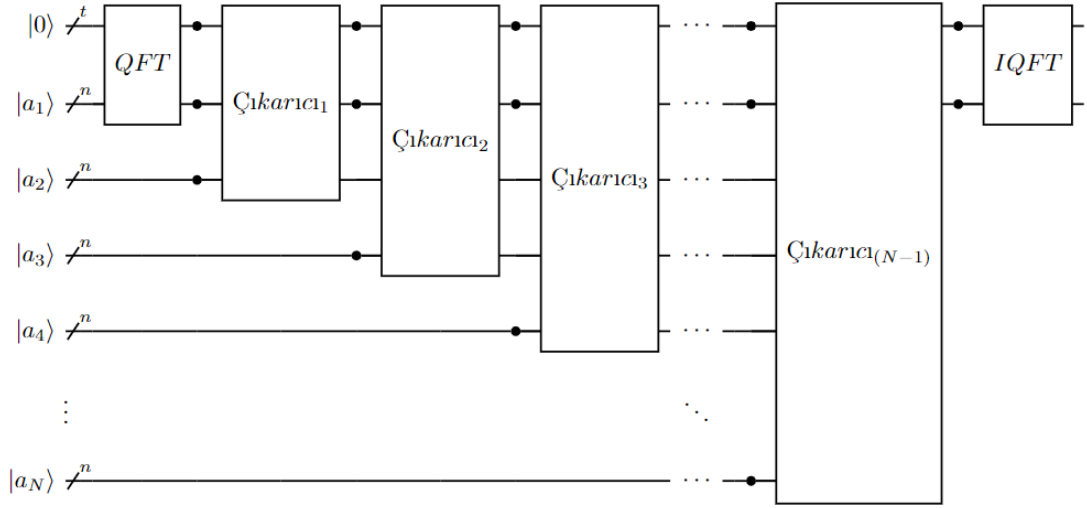
Şekil 4.9. Çıkarıcı parça

Şekil 4.9'daki bu parça iki sayının eşit bitlerle temsil edildiği ve  $a$  sayısının  $b$  sayısından büyük olduğu durumlar için geçerlidir. Böyle bir işlemde toplama-  
daki gibi elde olacak bir sayı olmadığı için elde olan biti tutmak için herhangi  
bir yardımcı girişe ihtiyaç duyulmaz. Ayrıca Kuantum hesaplama da kuantum  
kapılarını temsil eden matrisler hermityen oldukları için  $P_d(\theta_k)^-$  yerine  $P_d(\theta_k)^\dagger$   
temsilleri de kullanılabilir.

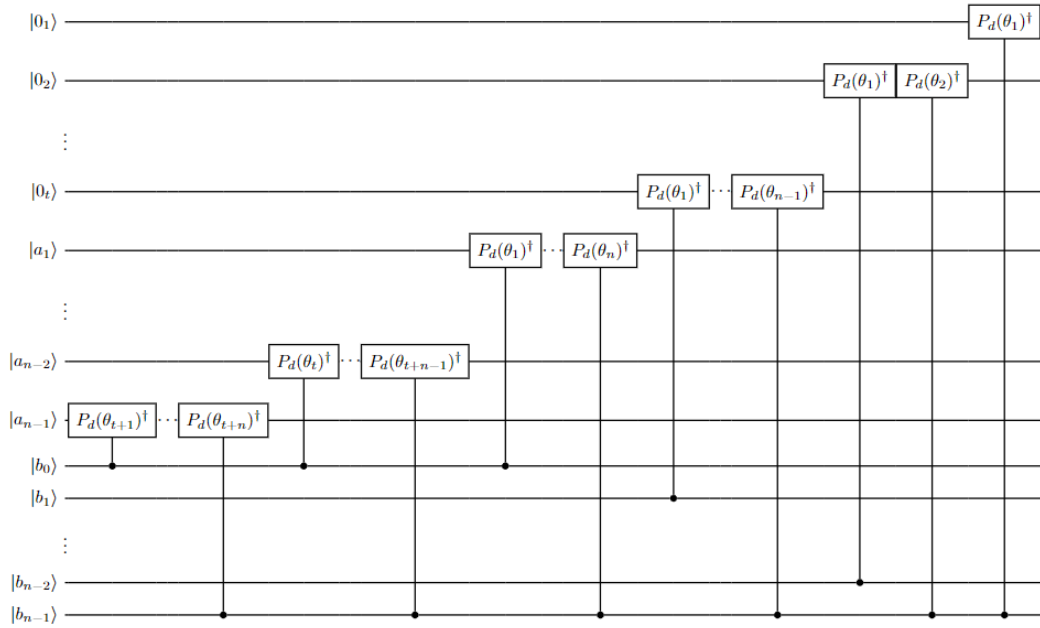
#### 4.2.2. N Tane Sayı için Tek Bir Devrede Çıkarma İşlemi

4.1 Bölümünde verilen  $N$  tane  $n$  bitlik sayının QFT Tabanlı toplama işlemini  
gerçekleştiren devreye benzer olarak, sonuç negatif çıkmamak şartıyla  $n$  bitlik  
sayıdan istenilen adette sayı yine tek bir devre kullanılarak çıkarılabilir. Bu  
işlemi gerçekleştiren QFT tabanlı kuantum devre Şekil 4.10'da görülmektedir.  
Bu devre, sadece başlangıçta ve son adımda QFT ve IQFT kullanıldığı için  
her işlem adımında QFT ve IQFT kullanılarak gerçekleşen devrelere göre daha  
avantajlıdır. Ayrıca kubitlerin yerine kuditlerin kullanılması da devrenin avan-

tajını artırmaktadır. Bu devrede dikkat edilmesi gereken nokta çıkarma işlemini gerçekleştiren kısmın  $t$  adet yardımcı girişli bir devrede kullanılmasıdır. Bu durumda Şekil 4.9'daki genelleştirme çalışmaz durumdadır. Bu kusuru giderebilmek için Şekil 4.2'de  $n+t$  bitlik genel toplayıcı devresinde kullanılan tüm kapıların hermityenlerini kullanarak genel bir çıkarıcı elde edilir. Bu genel çıkarıcı şekil 4.11'de görülmektedir.



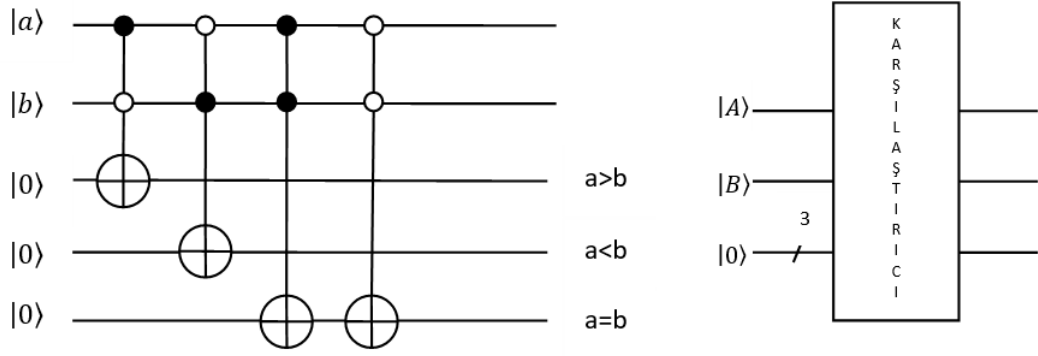
Şekil 4.10. QFT tabanlı seri çıkarıcı devre



Şekil 4.11.  $t+n$  bitlik genel çıkarıcı

### 4.2.3. $a < b$ için $a - b$ İşlemini Gerçekleştiren QFT Tabanlı Çıkarma İşlemi

Bazı durumlarda sayı değeri küçük olan bir sayıdan, sayı değeri büyük olan bir sayı çıkarılabilir. Böyle bir durumda sonucun negatif bir sayı olacağı bilinir. Bu durumda ilk sayıya QFT uygulanmadan önce sayıların karşılaştırmasını yapan bir ek devre olmalıdır. Karşılaştırma sonucu eğer küçük sayıdan büyük sayı çıkarılmak istenirse sonucun negatif olduğunu temsil eden ilave bir çıkışa daha sahip olmalıdır. Bir bitlik iki sayının karşılaştırmasını yapan devre şekil 4.12'de görülmektedir. Literatürde benzer çalışmalar mevcuttur (Oliveira,2007; Shahrokh, 2017). Burada sunulan karşılaştırmacı devre sayıların olası tüm karşılaştırmalarını verebilen 3 çıkışlı bir devredir.



Şekil 4.12. Bir bitlik karşılaştırmacı devre

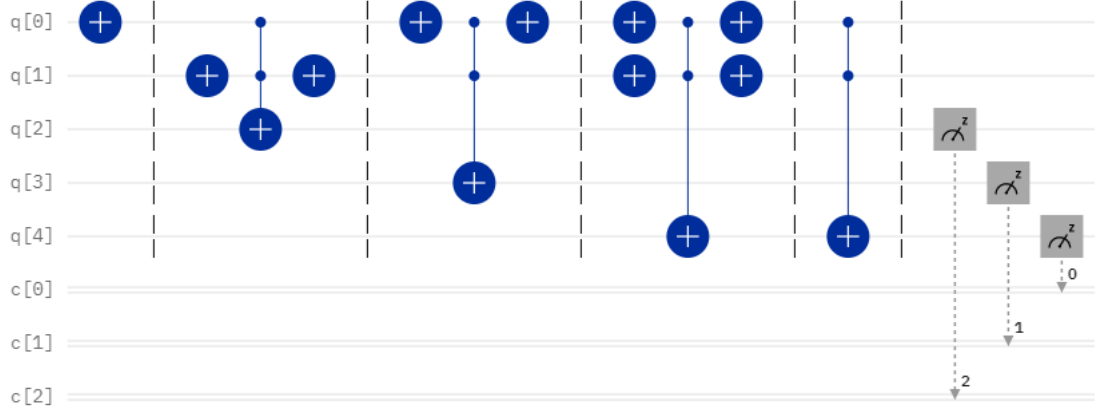
Görüldüğü gibi devre, ilk 2 giriş sayıları 3 giriş de çıktıyı tutmak için kullanılan yardımcı girişleri temsil etmek üzere 5 girişten oluşmaktadır. Ayrıca her iki girişi de kontrol eden iki kontrollü NOT kapılarından oluşmaktadır. Bu devrede gerçekleşecek olası tüm işlemler aşağıdaki Tablo 4.1'de görülmektedir.

Tablo 4.1. Bir bitlik karşılaştırmacı devre için olası girdiler ve çıktılar

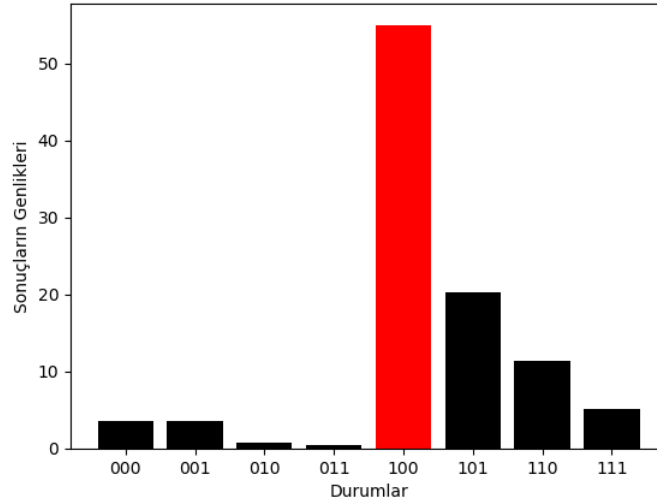
$a$	$b$	$a > b$	$a < b$	$a = b$
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

Şekil 4.12'deki karşılaştırmacı devre 7 kubit girişli *ibm - nairobi* gerçek kuantum bilgisayarında  $|a\rangle = |1\rangle$  ve  $|b\rangle = |0\rangle$  için hesaplanmıştır. İlk girişe NOT

kapısının uygulanmasının sebebi o girişi  $|1\rangle$ 'e ayarlamaktır. Arayüzde çizilen devre ve sonuçlar şekil 4.13 ve 4.14'de görülmektedir. Bu devrenin QASM kodları Ek 4'te verilmiştir.



Şekil 4.13.  $|a\rangle = |1\rangle$  ve  $|b\rangle = |0\rangle$  karşılaştırılması için IBM composer'de çizilmiş kuantum devre



Şekil 4.14.  $|a\rangle = |1\rangle$  ve  $|b\rangle = |0\rangle$  karşılaştırılma devresinin ibm-nairobi'den elde edilen sonuc grafiği

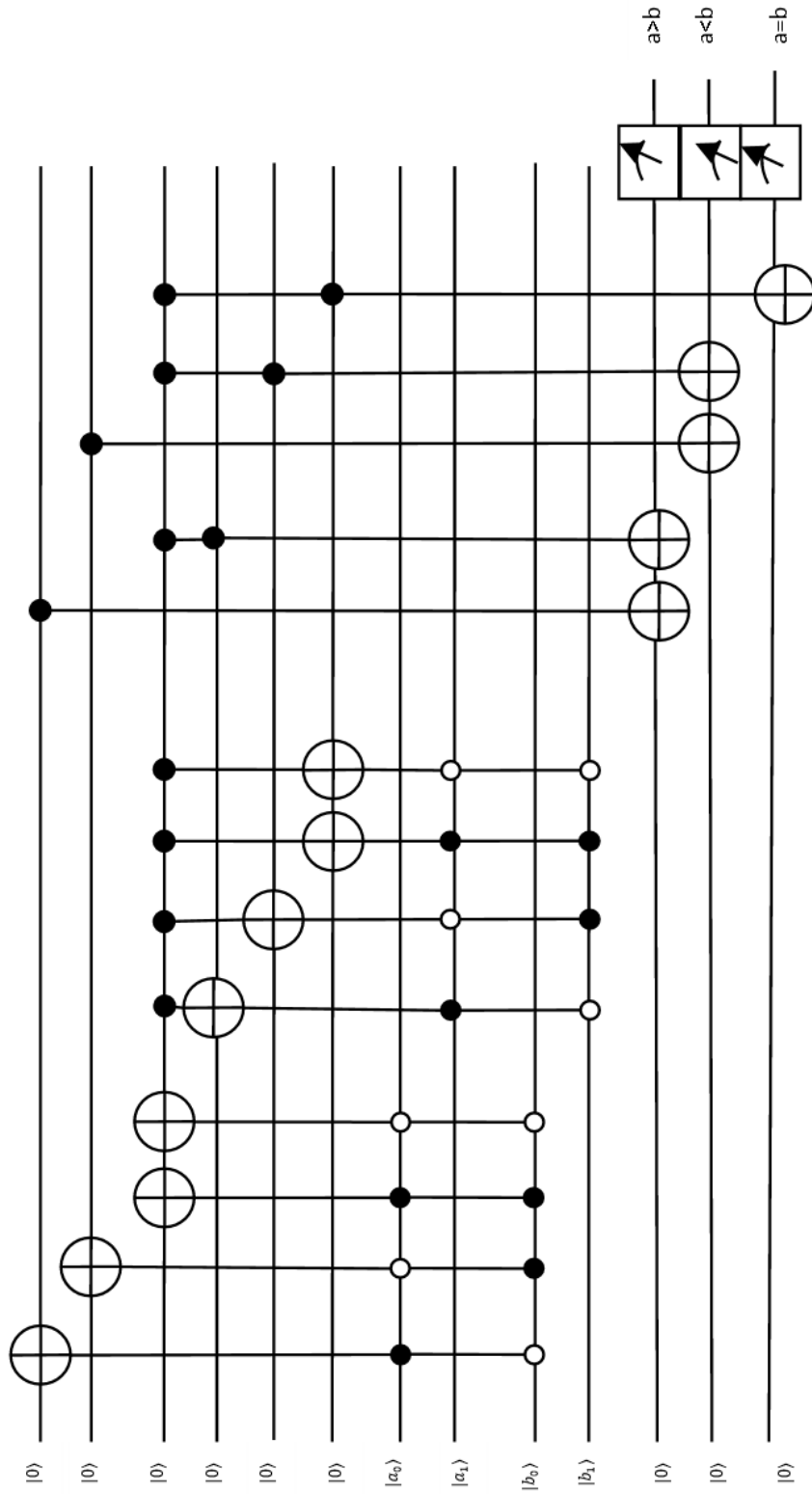
Şekil 4.14'deki sütun grafiğinde beklenen sonucun en yüksek genliğe sahip olduğu görülmektedir. Diğer değerlerdeki genlikler kubitler arası etkileşimlerden kaynaklanan gürültülerdir.

Karşılaştırılacak olan sayıların her biri iki bit ile temsil edildiğinde, karşılaştırma işlemini yapacak kuantum devre biraz daha karmaşıklaşır. Örneğin  $a_0a_1$  ve  $b_0b_1$  biçimindeki iki sayı için önce  $a_0$  ile  $b_0$  karşılaştırılması yapılır. Eğer  $a_0 > b_0$  ise  $a_0a_1$  sayısının  $b_0b_1$  sayısından büyük olduğu söylenir. Eğer  $a_0 < b_0$  ise  $a_0a_1$  sayısının  $b_0b_1$  sayısından küçük olduğu söylenir. Son olarak  $a_0 = b_0$  ise

karşılaştırma  $a_1$  ile  $b_1$  arasında yapılır. Bu işlemi gerçekleştiren kuantum devre Şekil 4.15’de görülmektedir.

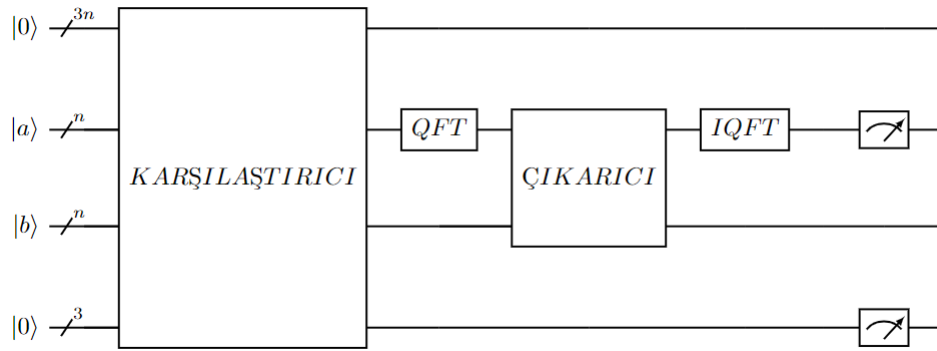
Bu devrede 9 adet yardımcı giriş kullanılmıştır. İlk 3 yardımcı giriş  $a_0$  ile  $b_0$  arasındaki karşılaştırma bilgisini tutar. Eğer bu bitler birbirine eşit ise buradan alınan çıktı ile  $a_1$  ile  $b_1$  sayıları karşılaştırılır. 4.5. ve 6. yardımcı girişler de buradan elde edilen karşılaştırmayı yani ilk sayıların eşitliği durumunda ikinci sayıların karşılaştırma bilgisini saklar. Son 3 yardımcı giriş ise nihai karşılaştırma sonucunu verir.

İki bitlik sayıların karşılaştırılmasını gerçekleştiren devre önerisi  $|01\rangle$  ve  $|10\rangle$  sayıları için IBM’in kullanıma sunduğu *simulator – mps* kullanılarak gerçekleştirilmiştir. Elde edilen sonucun beklenen sonuç ile uyumlu olduğu görülmüştür.



Şekil 4.15. İki bitlik karşılaştırıcı devre

Yukarıda tanımlanan 1 ve 2 bitlik sayıların karşılaştırmasını yapan devreler QFT tabanlı çıkarma işlemi yapan devreye entegre edilerek negatif sonucun da okunduğu bir çıkarma devresi tasarlanabilir. Bu devre sayılar arasında bir karşılaştırma yapar. Karşılaştırmadan sonra iki sayı arasında çıkarma işlemini gerçekleştirir. Eğer 1 bitlik iki sayı karşılaştırılacaksa Şekil 4.12’de görüldüğü gibi 3 tane yardımcı girişe ihtiyaç vardır. Eğer 2 bitlik iki sayı karşılaştırılacaksa Şekil 4.15’de görüldüğü gibi 9 tane yardımcı girişe ihtiyaç vardır. 3 bitlik iki sayı karşılaştırılacaksa 9 tane bit-bit karşılaştırması için 3 tane de sayıların kendilerini karşılaştırmak için toplamda 12 tane yardımcı girişe ihtiyaç vardır. Sayılar 4 bit ise 15, 5 bit ise 18 yardımcı girişe ihtiyaç duyulur. Buradan bir genelleme yapacak olursak bit sayısı 2 veya daha fazla olan sayılar için karşılaştırma devresinde  $3(n + 1)$  tane yardımcı girişe ihtiyaç duyulur. Bu bilgiler ışığında  $n$  bitlik iki sayı arasında önce karşılaştırma yapıp ardından QFT tabanlı çıkarma işlemi yapan kuantum devrenin genel şeması Şekil 4.16’de görülmektedir. Devredeki ilk ölçüm çıkarma işleminin sonucunu, ikinci ölçüm ise sonucun işaretinin okunduğu çıktıdır.

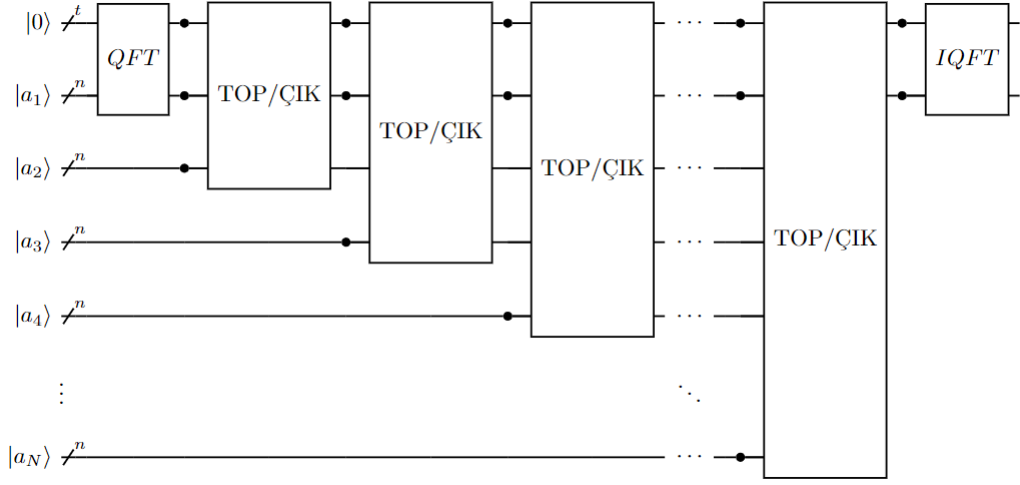


Şekil 4.16. Genelleştirilmiş karşılaştırmalı çıkarıcı kuantum devresi

Şekil 4.15’deki genel devre şeması kullanılarak,  $|a\rangle = |010\rangle$  ve  $|b\rangle = |110\rangle$  olmak üzere  $a - b$  işlemini gerçekleştiren devre Qiskit kullanılarak oluşturulmuş ve simüle edilmiştir. İşlemden anlaşılacağı üzere sonuç  $-100$  olmalıdır. Devre sonucunda  $|100010\rangle$  çıktısı elde edilmiştir. Burada ilk üç bit işlemin sonucunu, diğer üç bit sırası ile  $a > b, a < b$  ve  $a = b$  yi temsil etmektedir. Son üç bite bakıldığında  $a < b$ ’ye karşılık gelen çıktı 1 olduğundan, sonucun negatif olduğu anlamını taşır.

#### 4.2.4. QFT Tabanlı Toplama ve Çıkarma İşleminin Tek bir Devre Üzerinde Gerçekleştirilmesi

4.2.2 bölümünde QFT tabanlı seri çıkarıcı devre tanımlanmıştır. Bu şema kullanarak bir devre üzerinde ardışık şekilde toplama ve çıkarma işlemleri de yapılabilir. Örneğin  $a_1 - a_2 + a_3 - a_4 + \dots - a_{N-1}$  gibi bir işlem tanımlanabilir. Ardı ardına farklı kombinasyonlarda toplama ve çıkarma işlemi yapan kuantum devrenin genel şeması Şekil 4.17'de görülmektedir.



Şekil 4.17. Ardışık toplayıcı ve çıkarıcı kuantum devresi

#### 4.3. KFD Tabanlı Seri Toplamaya Dayalı Çarpma Devresi

Geleneksel bilgisayarlar çarpma işleminin ardı ardına toplama işlemini yaparak gerçekleştirir. Bu çalışmada kübit ( $d = 2$ ) sistemler için toffoli kapısını kullanarak sayılar bit-bit çarpıldıktan sonra çıkan sonuçlar 4.1 bölümünde tanımlanan QFT'ye dayalı toplama işlemi ile toplanıp sonuç elde edilmiştir.  $x$  ve  $y$  üçer bitlik iki sayı olmak üzere her biri  $|y_1y_2y_3\rangle$  ve  $|x_1x_2x_3\rangle$  olarak tanımlansın. Bu iki sayının çarpımı şekil 4.18'deki gibi gerçekleştirilir. Burada  $x_3$  sırasıyla  $y_3, y_2$  ve  $y_1$  ile çarpılıp sonuçlar satır olarak yazılır. Ardından  $x_2$  sırasıyla  $y_3, y_2$  ve  $y_1$  çarpımları yapıp 2. satıra bir adım sola kaydırarak yazılır. Son adımda da  $x_1$  sırasıyla  $y_3, y_2$  ve  $y_1$  ile çarpım sonuçları 3. satıra bir adım sola kaydırılarak yazılır. Bu sayıları toplamadan önce eşit bitli yapabilmek için sayıların sol ve sağına sıfırlar yazılır. Böylece bu işleme özgü olmak üzere 3 tane 5 bitlik toplanacak sayı elde ederiz. Çarpılacak sayıları temsil eden bit sayısı arttıkça

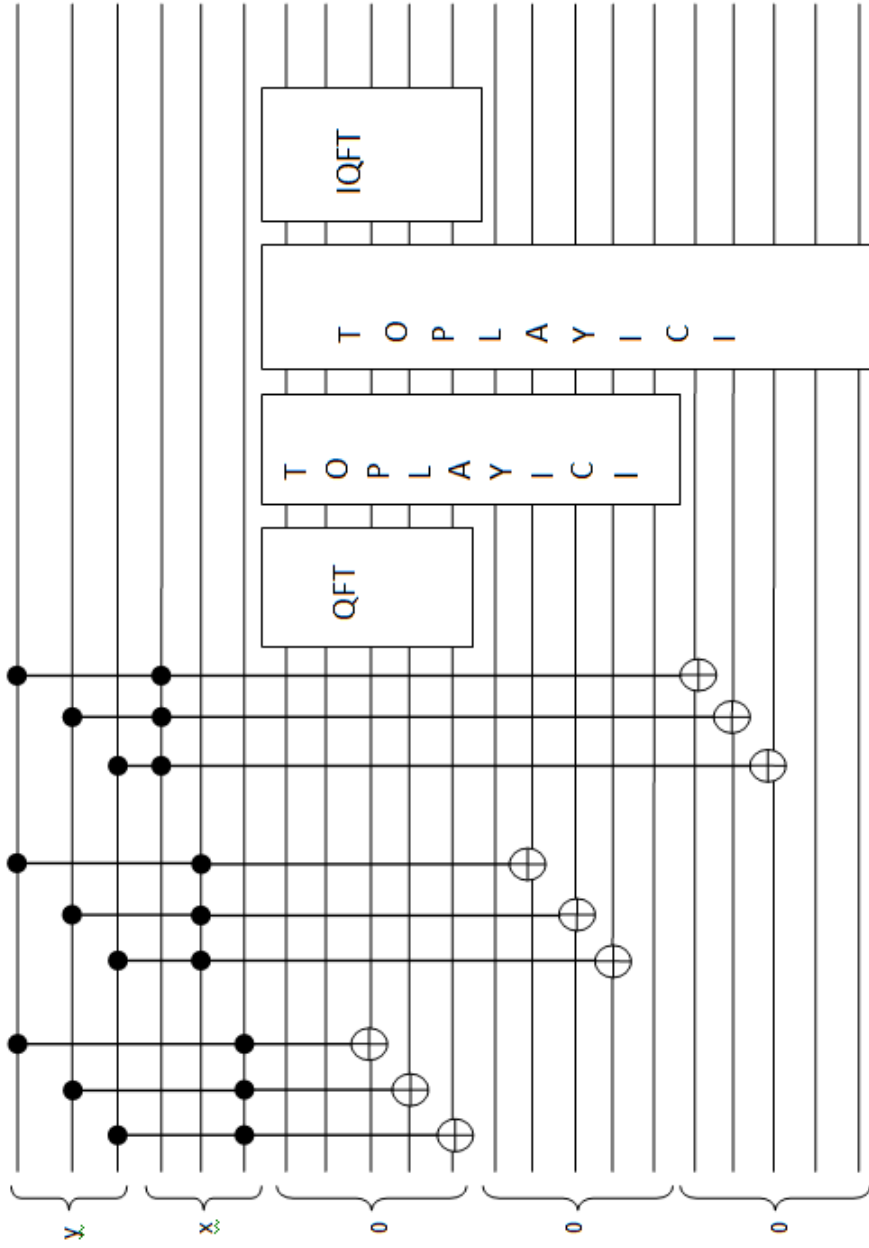
	y <sub>1</sub>	y <sub>2</sub>	y <sub>3</sub>	
	x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	
	X			
0	0	y <sub>1</sub> x <sub>3</sub>	y <sub>2</sub> x <sub>3</sub>	y <sub>3</sub> x <sub>3</sub>
0	y <sub>1</sub> x <sub>2</sub>	y <sub>2</sub> x <sub>2</sub>	y <sub>3</sub> x <sub>2</sub>	0
y <sub>1</sub> x <sub>1</sub>	y <sub>2</sub> x <sub>1</sub>	y <sub>3</sub> x <sub>1</sub>	0	0
	+			
	S O N U Ç			

Şekil 4.18. Üç bitlik iki sayının manuel çarpım yöntemi

toplanacak sayılarda artmaktadır. Bu aşamadan sonra QFT'ye dayalı seri toplama işlemi yapılarak çarpımın sonucu elde edilir. Tablo 4.2'de kubitler için olası tüm çarpımlar ve sonuçları görülmektedir. Bu tablo aynı zamanda toffoli kapısının da doğruluk tablosudur. Şekil 4.19'da 3 bitlik iki sayının çarpımını gerçekleştiren kuantum devre görülmektedir.

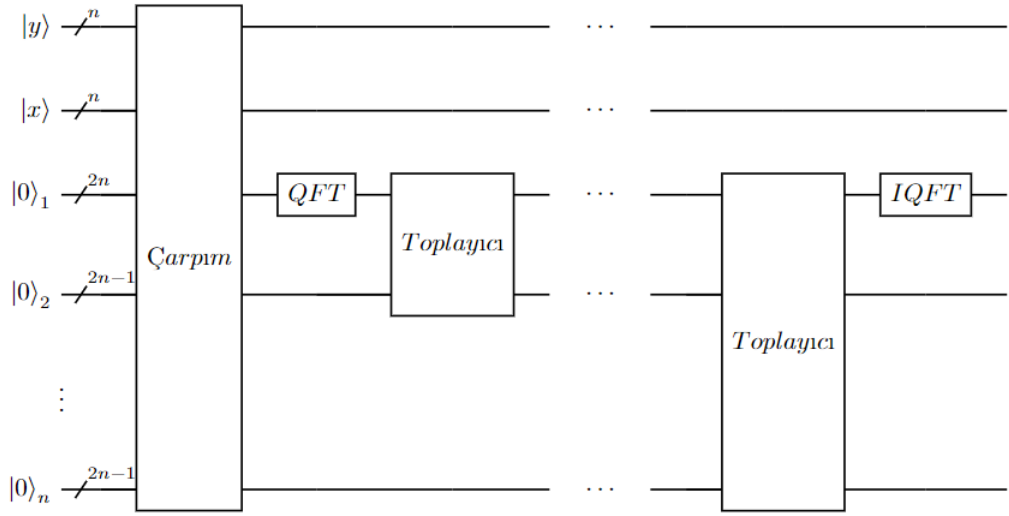
Tablo 4.2. Kubitlerin olası tüm çarpımları ve sonuçları

<i>x</i>	<i>y</i>	<i>x.y</i>
0	0	0
0	1	0
1	0	0
1	1	1



Şekil 4.19. 3 Bitlik iki sayının çarpımını gerçekleştiren kuantum devre

Burada 3'er bite kodlanan x ve y sayılarının çarpımı toffoli kapıları kullanılarak gerçekleştirilir. Bu çarpımlardan elde edilen sonuçları tutmak için 15 tane  $|0\rangle$ 'a ayarlanmış yardımcı giriş kullanılmıştır. Ardından şekil 4.18'deki her çarpım satırı QFT'ye dayalı seri toplama yöntemi ile toplanıp sonuç elde edilir. Bu devrede iki sayının alabileceği maksimum değer için elde olan sayıyı tutmak için 1 tane ilave yardımcı giriş kullanılmalıdır. Şekil 4.20'de n bitden oluşan iki sayının çarpım işlemini gerçekleştiren kuantum devre görülmektedir.



Şekil 4.20. Genelleştirilmiş çarpım devresi

#### 4.4. Kütrit ve Kukuart Sistemlerinin Çarpımları İçin Weyl Operatörlerinin Kullanılması

Kübitler arasında çarpım işlemi toffoli kapıları ile gerçekleştirilebilirken d seviyesi daha yüksek kuantum sistemler için toffoli kapısı geçersizdir. Daha yüksek seviyeli kuantum sistemler için kontrollü weyl operatörlerini kullanabilir. Genelleştirilmiş Weyl operatörünün matris elemanlarını veren matematiksel eşitlik aşağıda verilmiştir (Giraldo, 2020)

$$U_{nm} = \sum_{k=0}^{d-1} e^{\frac{2i\pi}{d} k.n} |k\rangle \langle (k+m) \bmod d| \quad (4.6)$$

Örneğin  $d = 2$ ,  $n = 0$  ve  $m = 1$  için bu  $U_{01} = X$ 'dir. Dolayısıyla  $U_{01} |0\rangle = |1\rangle$  ve  $U_{01} |1\rangle = |0\rangle$  işlemlerini gerçekleştirir.

Eşitlik 4.6,  $d = 3$ ,  $n = 0$  ve  $m = 2$  için çözümlerse  $U_{02}$  elde edilir. Bu operatör saf kütrit durumlarına uygulanırsa,

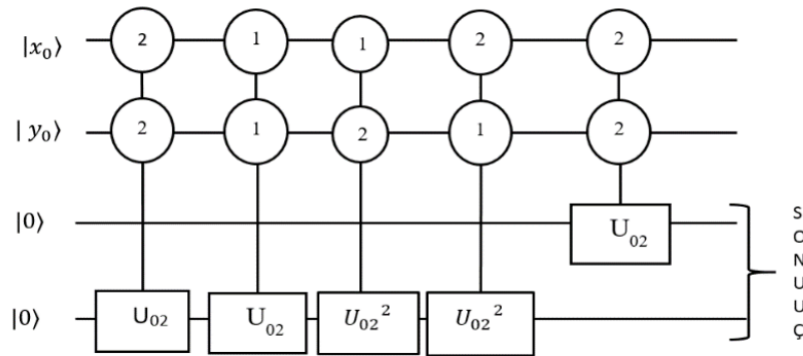
$$U_{02} |0\rangle = |1\rangle, U_{02} |1\rangle = |2\rangle, U_{02} |2\rangle = |0\rangle \quad (4.7)$$

dönüşümleri gerçekleşir. Tablo 4.3'de kütrit sistemlerdeki olası tüm çarpımlar ve sonuçları görülmektedir. Bu çarpımların sonuçlarını elde etmek için eşitlik 4.7'deki durumlar kullanılır.

Tablo 4.3. Kütritlerin olası tüm çarpımları ve sonuçları

$x$	$y$	$x.y$
0	0	00
0	1	00
0	2	00
1	0	00
1	1	01
1	2	02
2	0	00
2	1	02
2	2	11

Bu çarpımları gerçekleştiren kontrollü Weyl operatörlerinden oluşan kuantum devre Şekil 4.21'de görülmektedir.



Şekil 4.21. Kütrit sistemlerde olası tüm çarpımları gerçekleştiren temel kuantum devresi

Benzer şekilde Eşitlik 4.6,  $d = 4$ ,  $n = 0$  ve  $m = 3$  için çözümlerse  $U_{03}$  elde edilir. Bu operatör saf kukuart durumlarına uygulanırsa,

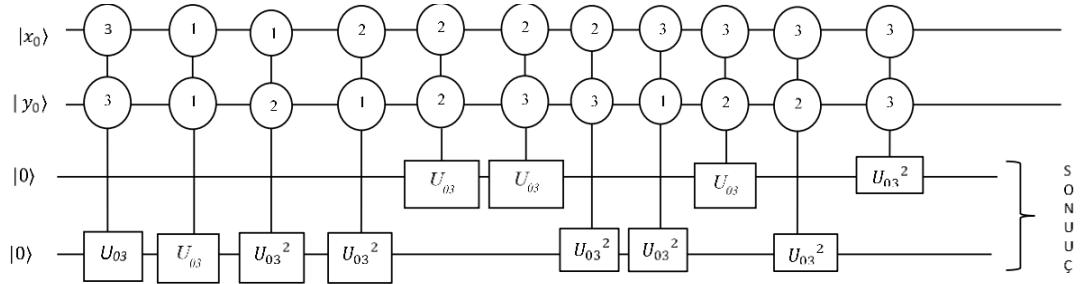
$$U_{03} |0\rangle = |1\rangle, U_{03} |1\rangle = |2\rangle, U_{03} |2\rangle = |3\rangle, U_{03} |3\rangle = |0\rangle \quad (4.8)$$

dönüşümleri gerçekleşir. Tablo 4.4'de kukuart sistemlerdeki olası tüm çarpımlar ve sonuçları görülmektedir. Bu çarpımların sonuçlarını elde etmek için eşitlik 4.8'deki durumlar kullanılır.

Tablo 4.4. Kukuartların olası tüm çarpımları ve sonuçları

$x$	$y$	$x.y$
0	0	00
0	1	00
0	2	00
0	3	00
1	0	00
1	1	01
1	2	02
1	3	03
2	0	00
2	1	02
2	2	11
2	3	12
3	0	00
3	1	03
3	2	12
3	3	21

Bu çarpımları gerçekleştiren kontrollü Weyl operatörlerinden oluşan kuantum devre Şekil 4.22'de görülmektedir.

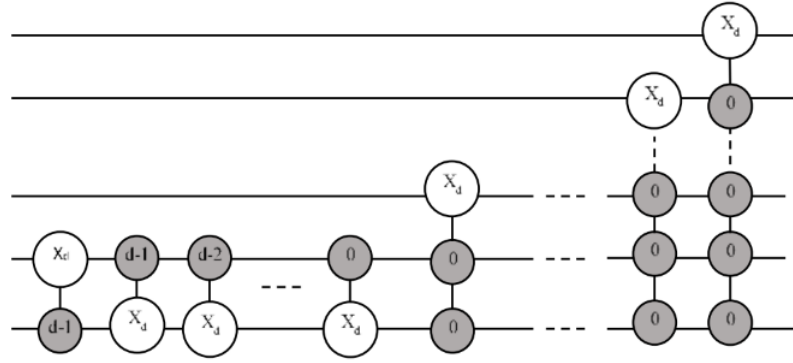


Şekil 4.22. Kukuart sistemlerde olası tüm çarpımları gerçekleştiren temel kuantum devre

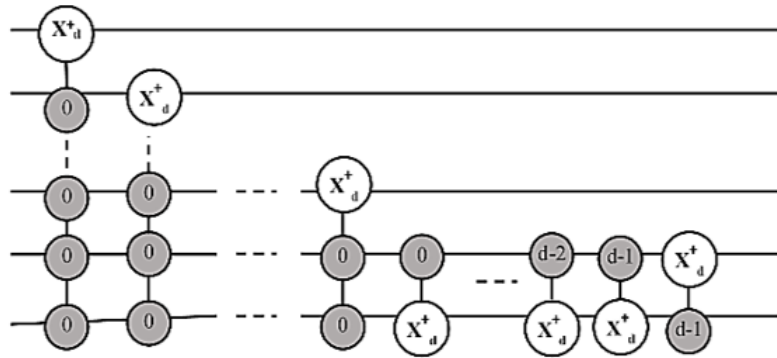
Görüldüğü gibi d seviyesi arttıkça olası tüm çarpımları gerçekleştiren kuantum devrelerinde hacmi büyümektedir. Şekil 4.19'deki toffoli kapılarından oluşan kısım, Şekil 4.21 ve 4.22'de verilen kuantum devreler Şekil 4.20'da Çarpım olarak isimlendirilen kısmı oluşturmaktadır. Bu işlemlerden sonra QFT'ye dayalı seri toplama uygulanarak çarpımların sonucu elde edilir.

## 4.5. Kuantum Sayaç

Geleneksel bilgisayarlar çarpma işlemi nasıl ki ardı ardına toplama işlemi yaparak gerçekleştiriyor, benzer şekilde bölme işlemi de ardı ardına çıkarma işlemi yaparak gerçekleştirirler. Ancak bölme işleminde çıkarma işleminin kaç kez tekrarlandığının önemli bir yeri vardır. Çünkü kaç kez çıkarıldığı bilgisine sahip olmak bölümün ne olduğu hakkında bilgi verir. Ayrıca Grover'in arama algoritması gibi algoritmalar kendini tekrarlayan adımlardan oluşur. Bu algoritmalarda tekrarın kaç kez yapıldığı da kuantum kapılarından oluşan bir alt devre ile ana devreye entegre edilebilir. Şekil 4.23 ve 4.24'de kontrollü  $X_d$  kapıları ile oluşturulmuş ve geliştirilmiş  $n$  girişli ileri ve geri sayım yapabilen kuantum devreler görülmektedir.



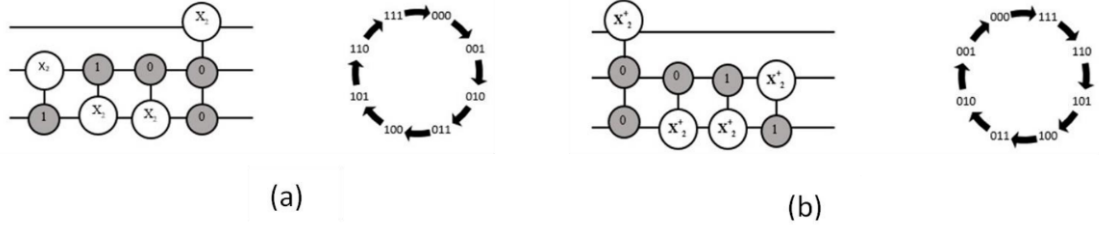
Şekil 4.23. Genelleştirilmiş  $n$  girişli ileri sayaç devresi



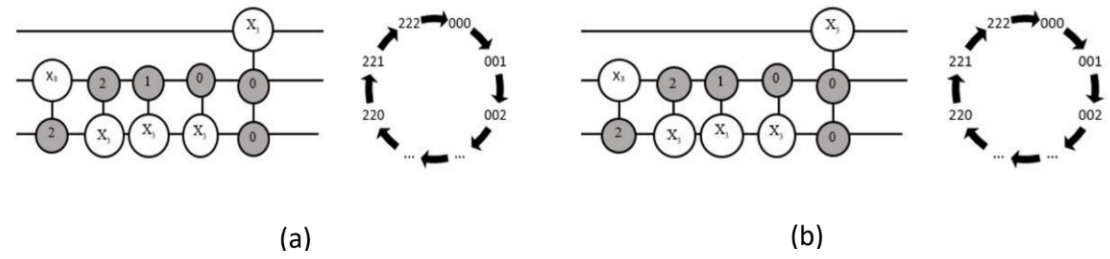
Şekil 4.24. Genelleştirilmiş  $n$  girişli geri sayaç devresi

Geri sayım yapan sayaç devresi kuantum kapılarının hermityen özelliği sayesinde kontrollü  $X_d^\dagger$  kapılarından oluşur. Şekil 4.23 ve 4.24'deki genel devrelerden

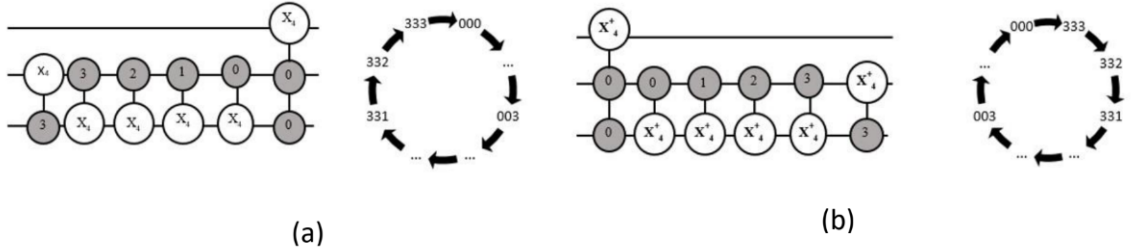
yola çıkarak 3 kubit girişli, 3 küttrit girişli ve 3 kukuart girişli ileri ve geri sayım yapan sayaç devreleri sırasıyla şekil 4.25, 4.26 ve 4.27’de çizilmiştir. Bu devrelerin yaptığı işlemlerin işlevsel diyagramları her şeklin yanında görülmektedir.



Şekil 4.25. 3 kubitlik (a) ileri ve (b) geri sayım yapan kuantum devreleri ve işlevsel diyagramları.



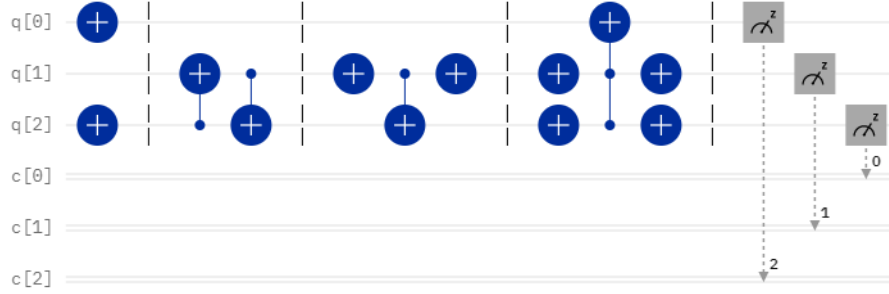
Şekil 4.26. 3 küttritlik (a) ileri ve (b) geri sayım yapan kuantum devreleri ve işlevsel diyagramları.



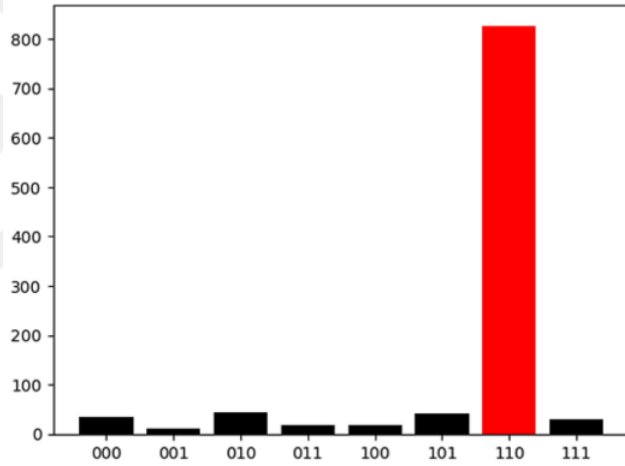
Şekil 4.27. 3 kukuartlık (a) ileri ve (b) geri sayım yapan kuantum devreleri ve işlevsel diyagramları.

Şekil 4.25(a)’da verilen 3 girişli sayaç devresi IBM’in gerçek kuantum bilgisayarı ( 5 kubitlik *ibm – quito*) test edilmiştir. Bu devrede giriş  $|101\rangle$  olarak ayarlandı. Bekleneceği üzere çıkışın  $|110\rangle$  olması gerekir. Elde edilen sonuçlar devrenin doğru tasarlandığını göstermektedir. Şekil 4.28’de IBM’in gerçek kuantum bilgisayarı için ara yüzde oluşturulmuş devre, Şekil 4.29’da ise bu devreden elde edilen sonuçlar sütun grafiği olarak gösterilmektedir. Bu devrenin QASM kodları da Ek 5’de verilmiştir. Beklenenin dışındaki değerlerde görülen büyüklükler

gerçek kuantum bilgisayarının maruz kaldığı gürültü v.b. nedenlerden kaynaklanmaktadır. Bunlar hedeflenen sonuç değerinin genliğine oranla ihmal edilebilir ölçekte dir.



Şekil 4.28. Girişi  $|101\rangle$  olarak ayarlanmış 3 girişli kubit tabanlı IBM'in gerçek kuantum bilgisayarı için arayüzde çizilmiş devre.



Şekil 4.29.  $|101\rangle$  girişli ileri sayaç devresinin çıktılarını gösteren sütun grafiği

#### 4.6. KFD Tabanlı Çıkarma İşlemine Dayalı Bölme Devresi

Geleneksel bilgisayarlarda bölme işlemi ardı ardına çıkarma işlemi yapılarak gerçekleştirilir. Aşağıda bölme işlemi gerçekleştirilmenin adımları anlatılmaktadır.

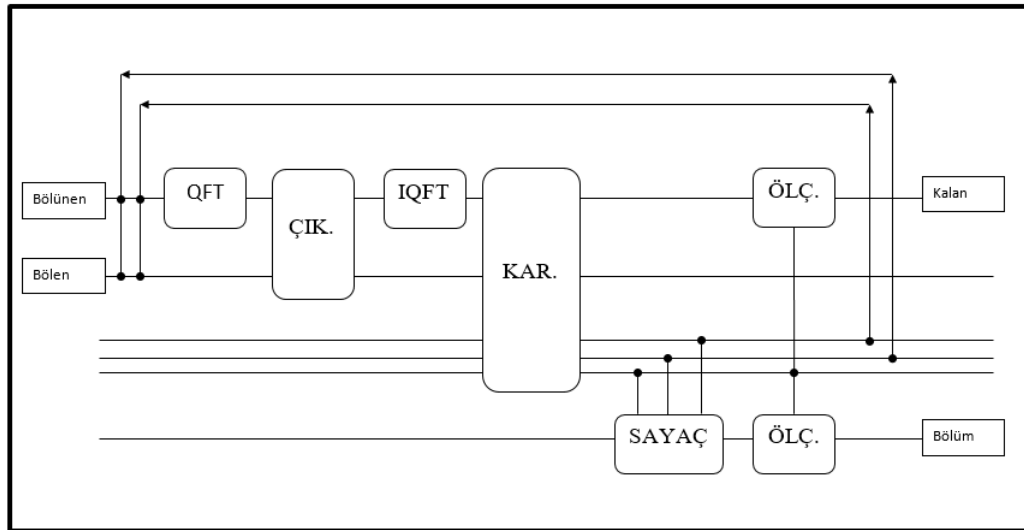
- $a$  bölünecek sayı  $b$  de bölen sayı olmak üzere;  $a - b$  işlemi gerçekleştirilir.
- $a - b = c$  olmak üzere  $c \equiv b$  ise bu döngü  $c - b < b$  olana kadar devam eder.
- $c - b < b$  olduğunda işlem durur. Bu tekrar kaç kez yapıldıysa bölüm ona eşittir. Kalan ise son çıkarma işleminde elimizde kalan sayıdır.

Bu adımlar tam sayılar arasında gerçekleşmektedir. Örneğin 5 sayısını 2'ye bölmek istediğimizde, önce  $5 - 2$  işlemi yapılır.  $5 - 2 = 3$ 'dür ve  $3 \equiv 2$  olduğundan çıkarma işlemine devam edilir. İlk çıkarma işleminden elde edilen sayıdan tekrar 2 çıkarılır.  $3 - 2 = 1$ 'dir. Görüldüğü gibi  $1 < 2$  olduğundan kalan 1'dir. Çıkarma işlemi de iki kez yapıldığı için bölüm 2'dir. Benzer yapıyı QFT'ye dayalı çıkarma işlemi yapan kuantum devre kullanarak da gerçekleştirebiliriz.

Kuantum bilgi işlemede bölme işlemini benzer adımlarla gerçekleştirebiliriz. Yine  $a$  bölünecek sayı  $b$  de bölen sayı olmak üzere,

- QFT Tabanlı  $a - b$  işlemi gerçekleştirilir. Sonuç  $c$  olmak üzere,
- 4.2.3 bölümünde tanıtilan karşılaştırma işlemi yapılır. Eğer  $c > b$  ise QFT tabanlı  $c - b$  işlemi tekrar yapılır. 4.4 bölümünde tanıtilan kuantum sayacı ile her çıkarma işlemi sayacın değerini 1 artırır.
- Tekrarlanan çıkarma işlemleri sonucunda  $c - b < b$  olduğunda işlem durur.
- Kontrollü ölçümler yapılarak sayaç devresi ve tekrarlanan çıkarma işleminin sonucu okunur. Sayaç devresinde okunan sonuç bölümü, çıkarma işleminden okunan sonuç da kalanı verir.

Şekil 4.30'de bu işlemi yapan devrenin genel şeması görülmektedir.



Şekil 4.30. QFT tabanlı kuantum bölme devresi

IBM'in kullanıma sunduğu gerçek kuantum bilgisayarında kontrollü ölçüm operatörü tanımlanmamıştır. Ancak Qiskit kütüphanesi kontrollü ölçüm yapmaya izin vermektedir. Bu sayede devrenin doğruluğu Qiskit kullanılarak test edilmiştir. Tablo 4.5'de verilen iki işlem Qiskit kütüphanesi kullanılarak simüle edilmiştir. Bu işlemlerden  $5/2$  işleminin Qiskit kodları da Ek 6'da verilmiştir.

Tablo 4.5. Qiskit kullanılarak test edilen bölme işlemleri

Bölünen	Bölen	Bölüm	Kalan	Devre Ölçüm (kalan-bölüm)	Sonuçları
3	2	1	1	01-01	
5	2	2	1	01-10	

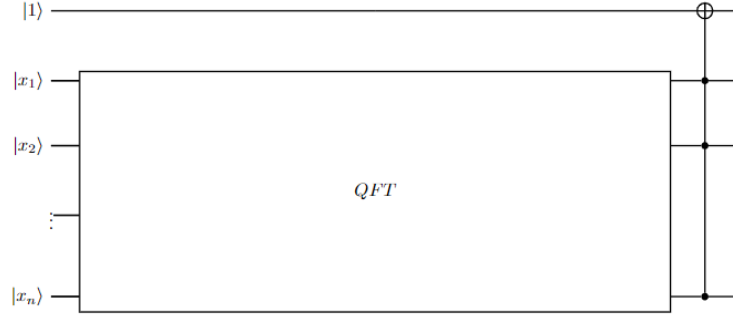
Tablo 4.5'de 1. işlem  $3 - 2$ 'dir. Bölüm 1 ve kalan 1'dir. Ölçüm değerlerine bakıldığında Kalan 01, bölüm 01 olarak okunmuştur. Bu devre QFT Tabanlı çıkarma işlemini yalnızca bir kez yaparak sonuca ulaşmıştır. İkinci işlem  $5 - 2$  işlemidir. Burada bölüm iki, kalan ise 1'dir. Ölçüm sonucu okunan değerler kalan için 01, bölüm için 10'dır. Bu işlemin gerçekleşmesi için QFT Tabanlı çıkarma işlemi 2 kez tekrarlanmıştır.

#### 4.7. QFT Tabanlı NAND Kapısı

Klasik bilgi işleme NAND, OR, AND gibi klasik mantık kapılarını kullanır. Bunların arasında en temel kapı NAND kapısıdır ve bu kapı kullanılarak diğer kapılar elde edilebilir. Bu kısımda QFT tabanlı NAND kapısı oluşturup sonuçları üzerinde tartışılacaktır.

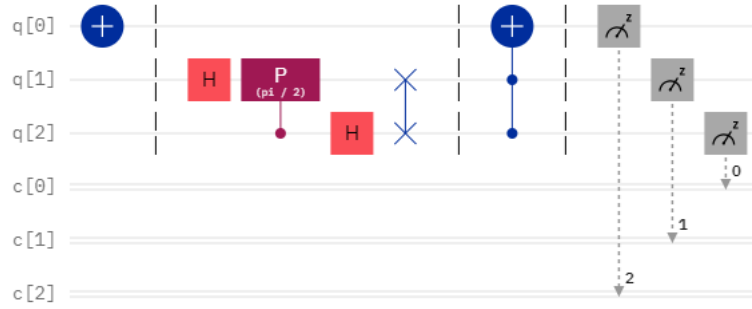
NAND mantık kapısı tüm girdilerin 1 olması durumunda 0, diğer tüm varyasyonlarda 1 çıktısını veren bir kapıdır. Bu kapının işlevi kuantum bilgi işlemede kontrollü NOT kapısı kullanılarak gerçekleştirilir. Örneğin iki giriş için bir tane de  $|1\rangle$ 'e ayarlanmış yardımcı giriş kullanılarak bu üç giriş üzerine toffoli kapısı uygulanarak gerçekleştirilir. Ancak kuantum bilgi işlemeyi klasik bilgi işlemekten ayıran en önemli özellik uygulanan kapıların süperpozisyon durumları üzerine etki ederek aynı anda tüm durumları dönüştürmesidir. Bu nedenle NAND kapısının işlevini kuantum bilgi işlemede girişlere QFT uyguladıktan sonra gerçekleştirilmiştir. QFT uygulandıktan sonra tüm olası durumlar süperpozisyon durumuna geçer ve uygulanan çoklu kontrollü NOT kapısı NAND kapısının işlevini yerine getirir.

Şekil 4.31’de  $n$  girişli bir durum için QFT tabanlı çoklu kontrollü not kapısından oluşan kuantum devre görülmektedir. Devrede  $|x_1x_2\dots x_n\rangle$  girişleri temsil etmektedir.



Şekil 4.31. QFT tabanlı  $n$  girişli NAND kapısı

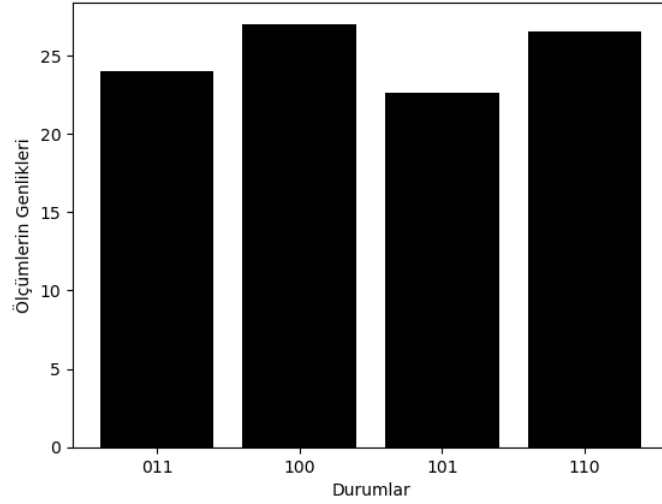
Örnek olarak 2 girişli bir durum için Şekil 4.31’deki devreyi uygulayalım. IBM simülatör-mps kullanılarak oluşturulan devre ve ölçüm sonuçları sırasıyla şekil 4.32 ve 4.33’de görülmektedir. Birinci giriş  $|1\rangle$ ’e ayarlanmış diğer iki giriş  $|00\rangle$ ’da tutulmuştur.  $|00\rangle$  durumuna QFT uygulandıktan sonra  $|00\rangle, |01\rangle, |10\rangle, |11\rangle$  durumları oluşturuldu. Ardından bu durumlar kontrol girişleri olmak üzere 1. girişteki  $|1\rangle$  durumuna NOT kapısı uygulandı. Şekil 4.33’de elde edilecek değerler ile Tablo 4.6’deki değerlerin uyum içinde olduğu görülmektedir.



Şekil 4.32. QFT tabanlı 2 girişli NAND kapısı devresi

Yatay ekseninde bulunan değerler görüldüğü gibi 3 bitlik değerlerdir. Birinci bit NAND işleminin sonucunu, 2. ve 3. bit ise girişleri göstermektedir.

Şekil 4.31’deki devreyi 3 giriş için uyguladığımızda NAND kapısı tüm girdilerin  $|1\rangle$  olması durumunda  $|0\rangle$ , diğer durumlarda  $|1\rangle$  çıktısını verir. IBM simülatör-mps

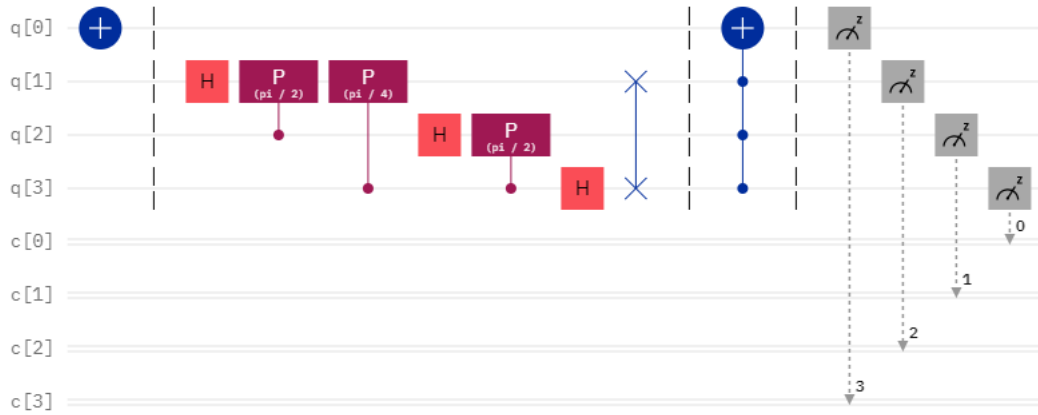


Şekil 4.33. QFT tabanlı 2 girişli NAND kapısı devresinin çıktısı

Tablo 4.6. IBM simlator-mps ile elde edilen 2 girişli QFT tabanlı NAND kapısı sonuçları

$q_1$	$q_2$	$\text{NAND}(q_1, q_2)$
0	0	1
0	1	1
1	0	1
1	1	0

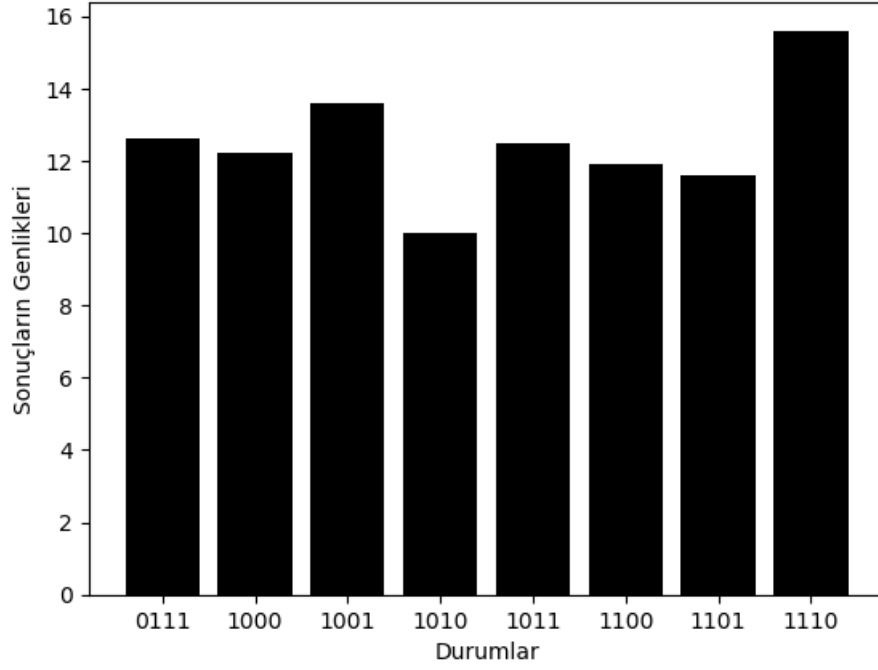
kullanılarak çizilen devre ve simülasyon sonuçları şekil 4.34 ve 4.35’de görülmektedir. Tablo 4.7’de 3 giriş üzerine etki eden NAND kapısının sonuçları ile simülasyon sonuçlarının uyum içinde olduğu da görülmektedir. QFT tabanlı NAND kapısı



Şekil 4.34. QFT tabanlı 3 girişli NAND kapısı devresi

için önerilen bu devrelerde girişlerin tamamı  $|0\rangle$ 'dır. Çıktılara bakıldığında ise olabilecek tüm girişler ve girişlere karşılık gelen tüm çıkışlar görülmektedir.

Ancak girişin özel olarak seçildiği bir durumda beklenen çıktıyı diğerlerinden



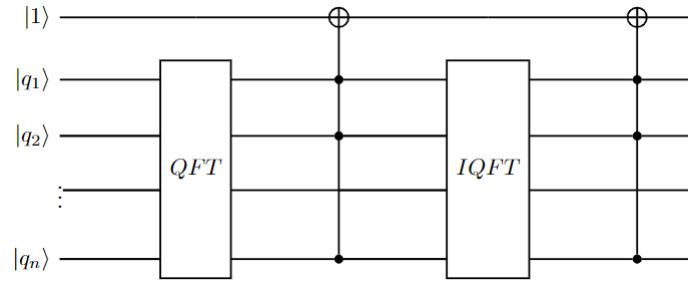
Şekil 4.35. QFT tabanlı 3 girişli NAND kapısı devresinin çıktısı

Tablo 4.7. IBM simlator-mps ile elde edilen 3 girişli QFT tabanlı NAND kapısı sonuçları

$q_1$	$q_2$	$q_3$	$NAND(q_1, q_2, q_3)$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

ayırmak için ölçüm genliğini nasıl artırılabilir. Bunu yapabilmek için yardımcı giriş yine  $|1\rangle$  olmak üzere diğer girişler özel olarak seçilir. Ardından girişlere QFT ve çoklu kontrollü NOT kapısı uygulanır. Sonraki adımda IQFT uygulandıktan sonra tekrar çoklu kontrollü NOT kapısı uygulanarak beklenen çıktı elde edilir. Bu işlemi gerçekleştiren geliştirilmiş kuantum devresi Şekil 4.36'da görülmektedir.

Örneğin  $NAND(1, 0) = 1$ 'dir. Şekil 4.36'daki geliştirilmiş devreyi bu örnek için uygularsak, Yardımcı girişle birlikte 3 girişli bir devre oluşturulur. Bu Şekil Qiskit kullanılarak çizilmiştir ve Şekil 4.37'de görülmektedir. İlk girdi  $|1\rangle$ , ikinci ve üçüncü girişler sırasıyla  $|1\rangle$  ve  $|0\rangle$ 'a ayarlanır. Burada  $q_0, q_1$  ve  $q_2$  sırasıyla 1., 2. ve 3. girişi temsil eder.

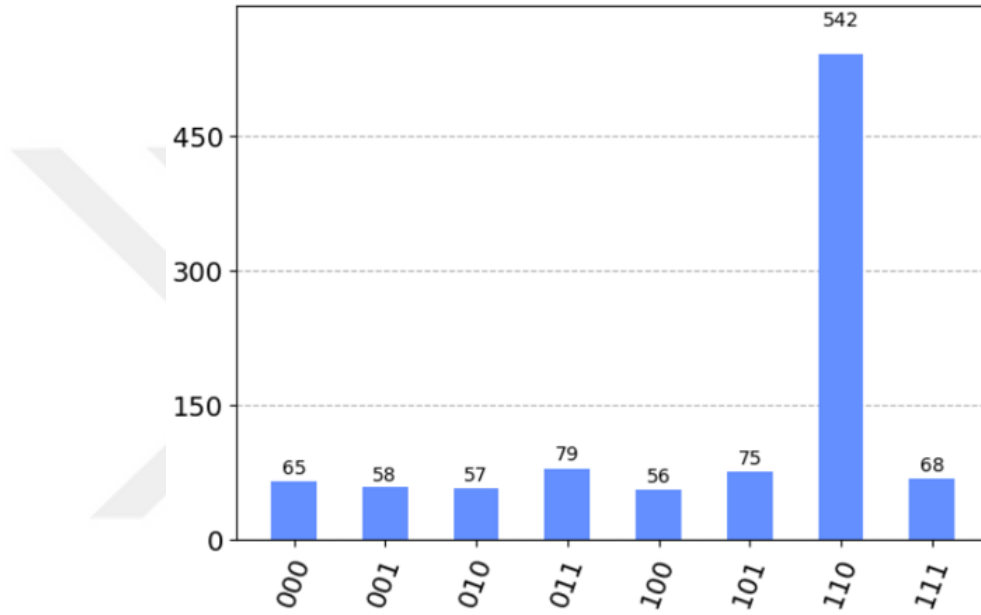


Şekil 4.36. Belirli bir girdiye uygulanan genelleştirilmiş QFT tabanlı NAND kapısı devresi



Bu çizim qiskit kullanılarak gerçekleştirilmiş ve simüle edilmiştir. Simülasyon sonucunun sütun grafiği de şekil 4.39’de görülmektedir. Görüldüğü gibi sonuç, beklenen ile uyum içindedir.

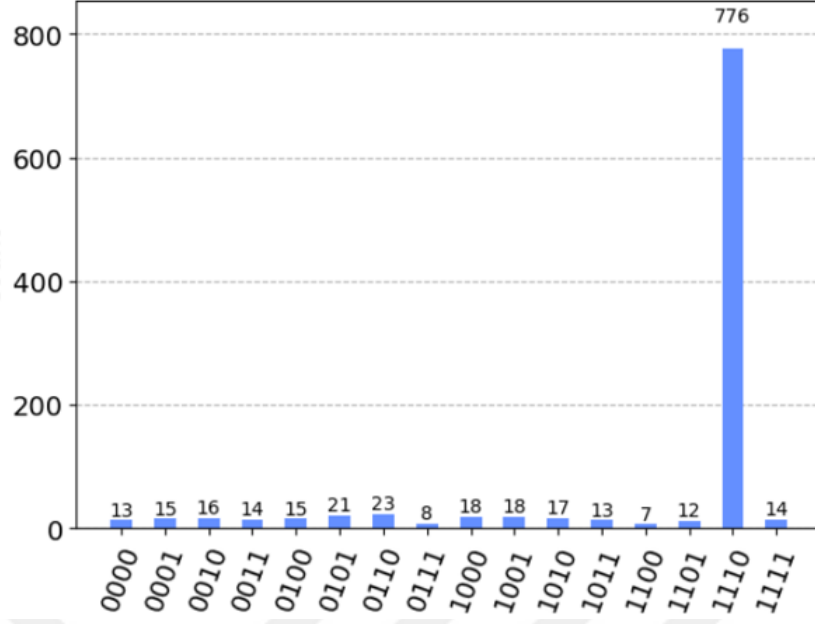
Benzer şekilde giriş sayımızı 1 artırarak  $NAND(110) = 1$  işlemini gerçekleştiren kuantum devreyi çizelim. Bu devre Şekil 4.38’deki gibidir. Devrenin simülasyon sonucu da Şekil 4.40’de görülmektedir. Sonucun beklenen ile uyumlu olduğu görülmektedir.



Şekil 4.39. NAND(10) için simülasyon sonucu

#### 4.8. QFT Tabanlı Aritmetik Mantık Birimi Tasarımı

Birden fazla aritmetik işlem yapabilen devrelere aritmetik mantık birimi (ALU) denir. Bu devreler birden fazla aritmetik işlemi yapmak üzere tasarlanmıştır. Ancak yapılmak istenen işlem seçildikten sonra devrenin işleyişi seçilen işlemi gerçekleştirmek üzere işler. QFT tabanlı olmayan çok sayıda aritmetik işlemi yapabilme kapasitesine sahip ALU tasarımları mevcuttur (Kamaraj,2019; Bolhassani,2016; Micheal, 2016). Bu çalışma da ilk defa QFT tabanlı bir kuantum aritmetik mantık birimi (qALU) tasarımı gerçekleştirilmiştir. Bu tasarı QFT Tabanlı NAND ve QFT tabanlı toplama işlemlerini gerçekleştirmektedir. Bu nedenle ilkel bir qALU adı verilmiştir. Bölüm 4.7.1., 4.7.2 ve 4.7.3’de sırasıyla bir bitlik iki sayı, bir bitlik dört sayı ve 2 bitlik 2 sayı için qALU devrelerinin ta-

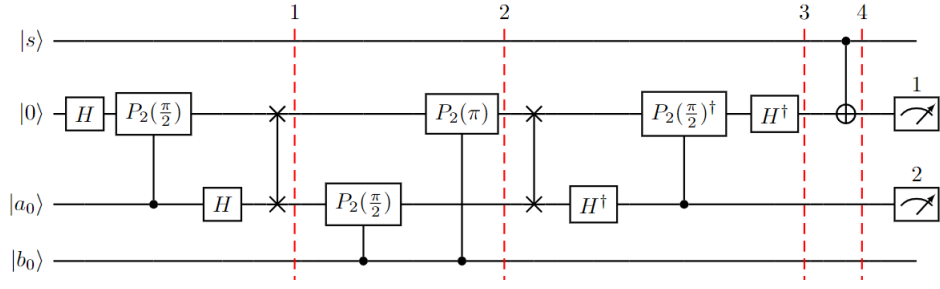


Şekil 4.40. NAND(110) için simülasyon sonucu

sarımları ve işleyişleri verilmektedir. Ayrıca bu devrelerin simülasyonu IBM'in simülatör-mps isimli simülatörü kullanılarak gerçekleştirilmiş ve sonuçların beklenen ile uyumlu olduğu görülmüştür.

#### 4.8.1. Bir Bitlik İki Sayının QFT Tabanlı Toplama ve NAND İşlemi

Şekil 4.41'de bir bitlik iki sayının QFT tabanlı NAND ve QFT tabanlı toplama işlemini gerçekleştiren dört girişli bir qALU devre tasarımı görülmektedir. Bu devrede girişler  $|s\rangle$ ,  $|0\rangle$ ,  $|a_0\rangle$  ve  $|b_0\rangle$ 'dır. Girişlerden  $|a_0\rangle$  ve  $|b_0\rangle$  aralarında işlem yapılacak sayıların kodlandığı girişlerdir.  $|0\rangle$  toplama işleminde gerekli olan yardımcı girişi  $|s\rangle$  ise yapılmak istenen işlemin seçileceği girişi temsil etmektedir.  $|s\rangle = 0$  olduğunda bu devre QFT tabanlı toplama işlemini gerçekleştirerek işlem sonucunu ölçüm operatörlerinin bulunduğu hatlarda vermektedir. 1. ölçüm işlem sonucunun birinci bitini, 2. ölçüm işlem sonucunun 2. bitini vermektedir. Eğer  $|s\rangle = 1$  seçilirse devre QFT tabanlı toplama işlemini kullanarak  $NAND(|a_0\rangle, |b_0\rangle)$  sonucunu vermektedir. NAND işleminin sonucu tek bit ile ifade edilebildiği için 2. ölçümden okunan değer bir önemi yoktur. Tablo 4.8'de bu devrenin gerçekleştirebileceği tüm işlemler sonuçları ile birlikte gösterilmiştir.



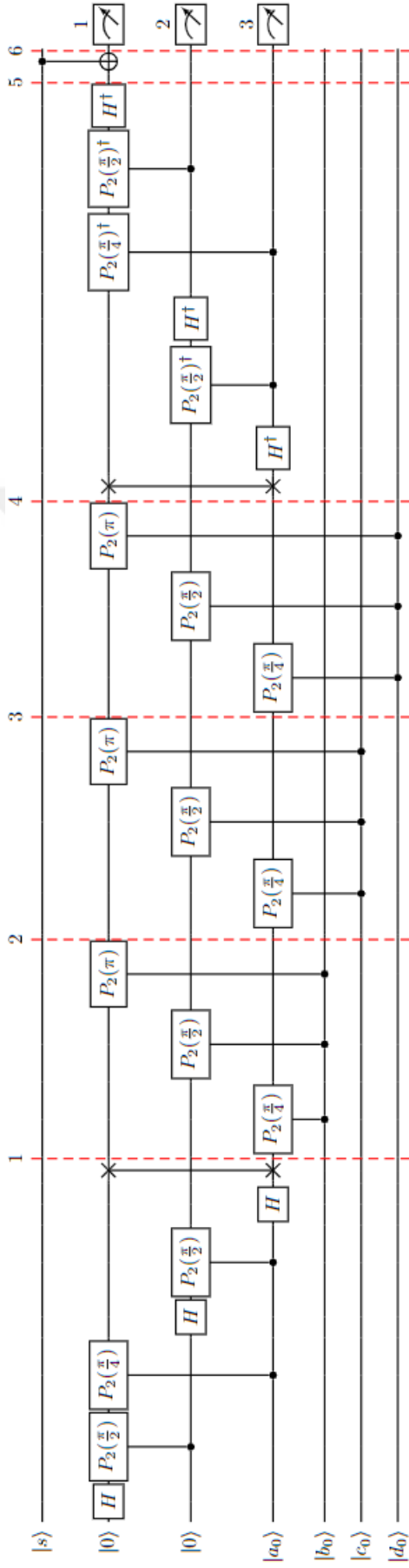
Şekil 4.41. Bir bitlik 2 giriş için qALU

Tablo 4.8. Bir bitlik iki sayı için qALU

$ s\rangle$	$ a_0\rangle$	$ a_1\rangle$	Ölçüm- 1	Ölçüm- 2
1	0	0	1	X
1	0	1	1	X
1	1	0	1	X
1	1	1	0	X
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0

#### 4.8.2. Bir Bitlik Dört Sayının QFT Tabanlı Toplama ve NAND İşlemi

Şekil 4.42'de  $|a_0\rangle, |b_0\rangle, |c_0\rangle$  ve  $|d_0\rangle$  gibi bir bitlik 4 sayının QFT tabanlı NAND ve toplama işlemini gerçekleştiren qALU devresi görülmektedir. Bu devre 7 girişli olarak tasarlanmıştır. İlk giriş olan  $|s\rangle$  yapılmak istenen işlemin NAND mı yoksa toplama mı olduğunun belirlendiği giriştir.  $|s\rangle = |0\rangle$  olduğunda devre QFT tabanlı toplama işlemini gerçekleştirmektedir.  $|0\rangle$ 'a ayarlanmış iki yardımcı giriş bu toplamadan elde edilecek sonucun tutulması için gerekli olan girişlerdir. 1. ölçümden sonucun birinci biti, ikinci ölçümden sonucun ikinci biti ve üçüncü ölçümden de sonucun 3. biti okunur.  $|s\rangle = |1\rangle$  olması durumunda devre bu sayıların NAND işlemini QFT tabanlı toplama işlemi yaparak gerçekleştirir.  $NAND(|a_0\rangle, |b_0\rangle, |c_0\rangle, |d_0\rangle)$  sonucu bir bitlik sayı olduğundan ölçüm 2 ve ölçüm 3 sonuçlarının okunmasına gerek yoktur.



Şekil 4.42. Bir bitlik 4 girişli qALU

Bu devrenin olası girdileri ve işlem hacmi Tablo 4.9'da görülmektedir.

Tablo 4.9. Bir bitlik dört sayı için qALU

$ s\rangle$	$ a_0\rangle$	$ b_0\rangle$	$ c_0\rangle$	$ d_0\rangle$	Ölçüm- 1	Ölçüm- 2	Ölçüm- 3
0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1
0	0	0	1	0	0	0	1
0	0	0	1	1	0	1	0
0	0	1	0	0	0	0	1
0	0	1	0	1	0	1	0
0	0	1	1	0	0	1	0
0	0	1	1	1	0	1	1
0	1	0	0	0	0	0	1
0	1	0	0	1	0	1	0
0	1	0	1	0	0	1	0
0	1	0	1	1	0	1	1
0	1	1	0	0	0	1	0
0	1	1	0	1	0	1	1
0	1	1	1	0	0	1	1
0	1	1	1	1	1	0	0
1	0	0	0	0	1	X	X
1	0	0	0	1	1	X	X
1	0	0	1	0	1	X	X
1	0	0	1	1	1	X	X
1	0	1	0	0	1	X	X
1	0	1	0	1	1	X	X
1	0	1	1	0	1	X	X
1	0	1	1	1	1	X	X
1	1	0	0	0	1	X	X
1	1	0	0	1	1	X	X
1	1	0	1	0	1	X	X
1	1	0	1	1	1	X	X
1	1	1	0	0	1	X	X
1	1	1	0	1	1	X	X
1	1	1	1	0	1	X	X
1	1	1	1	1	1	X	X
1	1	1	1	1	0	X	X

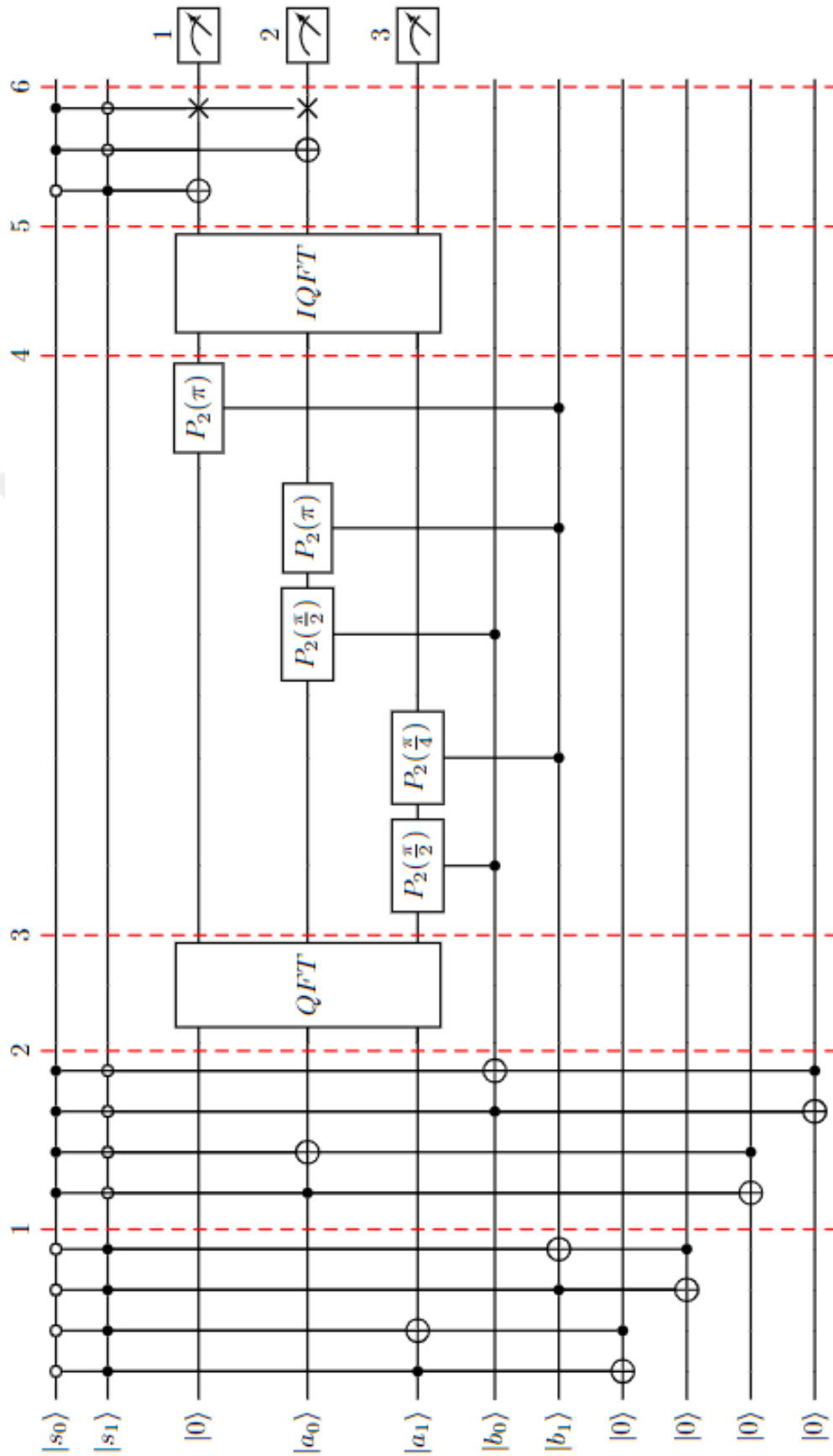
### 4.8.3. İki Bitlik İki Sayının QFT Tabanlı Toplama ve NAND İşlemi

Bu kısımda iki bitlik iki sayının QFT tabanlı toplama ve NAND işlemini gerçekleştiren kuantum devre tasarımı verilecektir. Şekil 4.43'de  $|a_0a_1\rangle$  ve  $|b_0b_1\rangle$  gibi kodlanmış iki sayının QFT tabanlı NAND ve toplama işlemini gerçekleştiren qALU tasarımı görülmektedir. Bu devre 2 adet NAND ve 1 adet toplama olmak üzere 3 işlemi gerçekleştirme kapasitesine sahiptir. İşlem seçimi için gerekli

olan  $|s_0s_1\rangle = |00\rangle$  ise devre QFT tabanlı toplama işlemini gerçekleştirmektedir. Eğer  $|s_0s_1\rangle = |01\rangle$  ise devredeki ilk 4 çoklu kontrollü not kapısı  $|a_1\rangle$  ve  $|b_1\rangle$  girişlerini sıfırlayarak QFT tabanlı  $|a_00\rangle + |b_00\rangle$  işlemini yapar. Elde ettiği sonucun ilk biti  $NAND(a_0, b_0)$ 'ın sonucudur. Bu değer 1. ölçümden okunur. Eğer  $|s_0s_1\rangle = |10\rangle$  ise devredeki ikinci 4 çoklu kontrollü not kapıları  $|a_0\rangle$  ve  $|b_0\rangle$  girişlerini sıfırlayarak QFT tabanlı  $|0a_1\rangle + |0b_1\rangle$  işlemini yapar. Elde ettiği sonucun ilk biti  $NAND(a_1, b_1)$ 'ın sonucudur. Bu değer kontrollü SWAP kapısı kullanılarak 1. ölçümden okunur. Örnek olarak, bu devrede gerçekleştirilebilecek örnek 2 işlem Tablo 4.10'da verilmiştir.

Tablo 4.10. 2 bitlik 2 sayı için bazı durumlar qALU

$ s_0\rangle$	$ s_1\rangle$	$ a_0\rangle$	$ a_1\rangle$	$ b_0\rangle$	$ b_1\rangle$	Ölçüm- 1	Ölçüm- 2	Ölçüm- 3
0	0	1	1	1	1	1	1	0
0	1	1	1	1	1	0	X	X
1	0	1	1	1	1	0	X	X
0	0	1	0	0	1	0	1	1
0	1	1	0	0	1	1	X	X
1	0	1	0	0	1	1	X	X



Şekil 4.43. 2 bitlik 2 girişli qALU

## 5. SONUÇ VE ÖNERİLER

### 5.1. Sonuç

QFT, kuantum bilgi işlemede önemli bir yer tutar. Girişlerin süperpozisyon durumuna getirilerek paralel işlemeye hazır hale getirilmesi, klasik bilgi işlemede olmayan bir özelliktir. Bu özellik kuantum bilgi işlemeyi klasik bilgi işlemeden ayıran en önemli özelliklerden biridir ve bilginin işlenmesinde kuantum üstünlüğüne sebep olur. Aritmetik işlemler QFT ön işlemi olmaksızın kuantum kapılar kullanılarak gerçekleştirilmiştir. Ancak bu yöntem klasik devrelerin yaptığı işlevi kuantum kapılar kullanarak gerçekleştirmekten öte geçmemiştir. QFT'nin önemi ve devre karşılığı literatüre kazandırıldıktan sonra a ve b gibi iki sayı arasında QFT tabanlı toplama, çıkarma, çarpma, bölme ve bir takım aritmetik işlemler gerçekleştiren kuantum devreler de literatüre kazandırılmıştır. Bu çalışma QFT tabanlı aritmetik işlemleri gerçekleştirmekle beraber yeni düzenlemeler ve beraberinde yeni avantajlar sağlayan devre önerilerinden oluşmaktadır.

Bu çalışmada önerilen yeni düzenlemeler ve buna bağlı tasarlanan devreler 4. bölümde bulunmaktadır. 4.1'de ikiden fazla sayı arasında gerçekleştirilecek toplama işleme için QFT tabanlı bir devre tasarısı önerilmiştir. Literatürde bulunan çalışmalara bakıldığında a ve b gibi iki sayı arasında toplama işlemi yapıldıktan sonra üçüncü bir c sayısının eklenmesi için ilk işlemde elde edilen sonuca tekrar QFT uygulayıp işlemde sonra IQFT uygulanarak sonucun elde edileceği çıkarımı yapılabilir. Ancak bu çalışmada  $N \geq 2$  olmak üzere N adet sayının toplama işlemi için devrenin başında ve sonunda olmak üzere tek bir defa QFT ve IQFT kullanılması önerilmiştir. Bu sayede her bir sayı için uygulanması gereken QFT ve IQFT'nin maliyetinden kurtulmuş olunur. 4.1.1'de 4 adet 2 bitlik sayının toplama işlemini yapan devre oluşturulduktan sonra IBM composer kullanılarak test edilmiş ve sonucun doğruluğu görülmüştür. 4.1.2'de aynı sayılar kukuartlara kodlanarak 4 adet bir bitlik giriş elde edilmiştir. Sadece devrenin başında ve sonunda bir kez QFT ve IQFT kullanılarak devre oluşturulmuş ve beklenen sonuç manuel olarak ile test edilmiş ve doğruluğu gösterilmiştir. Literatürde QFT tabanlı toplama işlemi yapan devre önerilerine bakıldığında şartlı faz kayması kapılarından

oluşan toplayıcı kısım yalnızca bir adet yardımcı girişe göre tasarlanmıştır.  $N$  adet sayının toplamı olduğunda bir adet yardımcı elde olan sayıyı tutmak için yetersiz kalmaktadır. Bunun için öncelikle ihtiyaç duyulan yardımcı giriş sayısını veren bağıntı herhangi bir  $d$  seviyeli kuantum sistemi için bulunmuş ve toplayıcı kısım bu yardımcı giriş sayısına göre yeniden düzenlenmiştir. 4.1 bölümünden elde edilen sonuçlara göre iki avantaj öne çıkmaktadır. Bunlardan ilki her sayı için tekrar QFT ve IQFT kullanımının önüne geçilerek kapı sayısı açısından ciddi bir maliyet düşüşü sağlanmıştır. İkinci olarak bu işlemlerin daha yüksek kuantum seviyeli sistemlerin kullanılması ile kapı sayısındaki azalma daha yüksek seviyede gerçekleştirilmiştir. 4.2 bölümünde kuantum kapılarının tersinir özelliği kullanılarak 4.1 bölümünde yeniden düzenlenen toplayıcı kısım ile benzer yapıda olan çıkarıcı kısım oluşturulmuştur. 4.2.1'de  $a - b$  işlemini gerçekleştiren QFT tabanlı kuantum devre önerisi yapılmıştır. Bu işlemde  $a > b$  kabul edildiği için çıkan sonuç pozitif olacaktır. 4.2.2'de ise yine sonuç pozitif olmak üzere  $a$  sayısından  $N$  tane sayının tek bir QFT ve IQFT kullanılarak çıkarılmasının gerçekleştirildiği kuantum devre önerilmiştir. Bu devrelerin işlerliği test edilip doğrulanmıştır. 4.2.3'de sonucu negatif çıkan işlemler için devre tasarımı yapılmıştır. Burada sonucun negatif olduğu bilgisini tutacak olan ilave girişlere ve çıkarma işlemi gerçekleşmeden önce sayıların büyüklük karşılaştırmasını yapan ek bir devre önerisi yapılmıştır. Karşılaştırma işlemini yapan devrenin 3 adet okuma çıktısı bulunmaktadır. Bu çıktılar sırasıyla  $a > b$ ,  $a = b$  ve  $a < b$  bilgilerini tutmaktadır. Bu sıraya göre eğer 3. çıktı 1 ise sonucun negatif olduğu anlaşılmaktadır. 4.2.4'de 4.1'de önerilen seri toplayıcı kalıbı kullanılarak hem toplama hem de çıkarma işlemi yapan devre tasarımı sunulmuştur. Toplayıcı ve çıkarıcı kısımlardaki kapı düzenleri aynı olmakla beraber çıkarıcı kısımdaki şartlı faz kayması kapıları toplayıcı kısımda kullanılanların kompleks eşleniklerinin transpozlarıdır.

4.3'de 4.1'de tanımlanan seri toplama işlemini gerçekleştiren devre kullanılarak QFT tabanlı çarpma işlemini gerçekleştiren devre önerisi yapılmıştır. Bir dizi bit ile temsil edilen sayılar arasında gerçekleşen çarpma işlemi sonucunda çıkan sonuçların toplanması gerekir. Bu adım seri toplayıcı kullanılarak gerçekleştirilmiştir. Ayrıca bu işlem kütrit ve kukuart işlemleri için de düzenlenmiştir. Kubitler arasında çarpım toffoli kapısı kullanılarak

gerçekleştirilebiliyorken kütrit ve kukuartlar arasında bu işlemi yapan ve Weyl operatörlerinden oluşan tasarımlar oluşturulmuştur.

4.4'de QFT tabanlı bölme işlemi için gerekli olan bir sayaç devresi tasarımı yapılmıştır ve bu devre herhangi bir  $d$  seviyeli kuantum sistemleri için genelleştirilmiştir. Ayrıca yine kuantum kapılarının tersinirlik özelliğinden faydalanılarak geri sayımda yapabilen devre önerisinde bulunulmuştur. Bu devreler IBM'in gerçek kuantum bilgisayarında test edilerek doğru sonuçlar verdiği görülmüştür.

4.5 bölümümde ardı ardına çıkarma işlemi yaparak bölme işlemi gerçekleştiren QFT tabanlı bölme işlemi gerçekleştiren devre önerisi yapılmıştır. Bu devrede 4.2'de önerilen karşılaştırıcı ve 4.4'deki sayıcı devreler alt devreler olarak kullanılmıştır. Birden fazla çıkarma işlemi yapılarak sonuca ulaşılan işlemlerde tekrar sayısını hesaplamak için sayıcı, ve tekrar çıkarma işleminin yapılacağına karar veren devre olarak da karşılaştırma devresi kullanılmıştır. Örnek olarak  $3/2$  ve iki tekrar içeren  $5/2$  işlemleri Qiskit kullanılarak test edilmiştir.

4.6'da kuantum bilgi işlemede karşılığı toffoli kapısı olan NAND kapısının, QFT tabanlı olanı tasarlanmıştır. Önce girişe QFT ile birlikte toffoli uygulanarak tüm durumları ayna anda veren sonuçlar elde edilmiştir. Ardından özel olarak girilen durumun NAND sonucu veren kuantum devre tasarımı yapılmıştır. Her iki aşama da IBM composer kullanılarak test edildi ve sonuçların sütun grafiği elde edilmiştir.

Son kısımda QFT tabanlı toplama işlemi merkez alarak istenildiğinde toplama, istenildiğinde de girişlerin NAND sonucunu veren basit bir qALU tasarımı yapılmıştır. Bölümün devamında bir bitlik iki sayı, bir bitlik dört sayı ve iki bitlik iki sayı arasında gerçekleşecek olan toplama ve NAND işlemi barındıran qALU devresi önerileri özel olarak oluşturulmuş ve test edilmiştir.

## 5.2. Öneriler

QFT tabanlı toplama veya seri toplama devresi merkez alınarak önerilen qALU devresi çıkarma, çarpma ve bölme gibi aritmetik işlemlerle beraber AND, OR, XOR, NOR gibi mantık kapılarını da içerek şekilde genişletilebilir. Bu çalışma da

önerilen devrelerin QFT tabanlı olması evrensel bir kuantum bilgisayarın çalışma ilkesine uymaktadır. Bu nedenle önerilecek tüm devrelerin QFT tabanlı olması kuantum bilgi işlemenin klasik bilgi işlemeden üstün olduğunu göstermenin en doğru yoludur.



## KAYNAKLAR

- Alves, É.M. vd. (2019). "Simulating single-spin dynamics on an IBM five qubit chip". *Revista Brasileira de Ensino de Física*, 42.
- Archimedes, P. ve Floratos E. (2017) "Arithmetic Circuits for Multilevel Qudits Based on Quantum Fourier Transform." *ArXiv* abs/1707.08834
- Benioff, P. (1980) "The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines". *J Stat Phys.* 22. 563-591,
- Bennett, C. ve DiVincenzo, D. (2000) "Quantum information and computation". *Nature*, 404, 247-255.
- Bernstein, E. ve Vazirani, U. (1997) "Quantum Complexity Theory". *SIAM Journal on Computing*, 26(5). 1411-1473.
- Bolhassani, A. ve Haghparast, M. (2017) "Optimized designs of reversible arithmetic logic unit". *Turkish Journal of Electrical Engineering and Computer Sciences*, 25(2). 1137-1146.
- Coles, P.J. vd. (2018). "Quantum Algorithm Implementations for Beginners". *ACM Transactions on Quantum Computing*, 3. 1-92.
- Cooley, J.V. ve Tukey J.W. (1965) "An Algorithm for the Machine Calculation of Complex Fourier Series". *American Mathematical Society*, 19(90). 297-301
- Cuccaro, S.A. vd. (2004) "A new quantum ripple-carry addition circuit". *arXiv: Quantum Physics*.
- Deutsch, D. (1985) "Quantum Theory, The Church-Turing Principle and The Universal Quantum Computer". *Proceedings of the Royal Society of London, Series A. Mathematical and Physical Sciences*, 400(1818). 97-117
- Deutsch, D. (1989) "Quantum Computational Networks". *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 425(1868). 73-90.
- Divincenzo, P.D ve IBM. (2000) "The Physical Implementation of Quantum Computation". *Fortschritte der Physik*, 48. 9-11.
- Draper, T.G. (2000) "Addition on a Quantum Computer". *arXiv: Quantum Physics*.
- Einstein, A. vd. (1935) "Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?". *Physical Review*, 47. 777-780.
- Feynman, R. (1960) "There's Plenty of Room at the Bottom". *Engineering and Science*, 23. 22-36.
- Feynman, R. (1982) "Simulating physics with computers". *Int. J. Theor. Phys*, 21. 467-488.
- Florio, G. ve Picca, D. (2004) "Quantum implementation of elementary arith-

- metic operations". *arXiv: Quantum Physics*.
- Gedik, Z. vd. (2014) "Computational speed-up with a single qudit". *Scientific Reports*, 5. 14671.
- Gershenfeld, N.A. ve Chuang I.L. (1997) "Bulk Spin-Resonance Quantum Computation". *Science*, 275(5298). 350-6.
- Giraldo, C. vd. (2020) "QuantumSkynet: A High Dimensional Quantum Computing Simulator" *Laser Science*, JW6A.25
- Gossett, P. (1998) "Quantum Carry-Save Arithmetic". *arXiv: Quantum Physics*.
- Grover, L.K. (1997) "Quantum Mechanics helps in Searching for a needle in a haystack". *Physical Review Letters*, 79. 325-328.
- Harrow, A.W. vd. (2009) "Quantum Algorithm for Linear Systems of Equations". *Physical Review Letters*, 103. 150502.
- Heidari, S. ve Farzadnia, E. (2017) "A novel quantum LSB-based steganography method using the Gray code for colored quantum images". *Quantum Information Processing*, 16. 242.
- Jozsa, R. (1997). "Quantum algorithms and the Fourier transform". *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454. 323-337.
- Kamaraj, A. ve Marichamy, P. (2019). "Design of integrated reversible fault-tolerant arithmetic and logic unit". *Microprocessors and Microsystem*, 69. 16-23.
- Karakaş, M.D ve Gençten, A. (2018). "Construction of Two-Ququart Quantum Entanglement by Using Magnetic Resonance Selective Pulse Sequences". *Zeitschrift für Naturforschung, A*. 73(10). 911-918
- Karimipour, V. vd. (2001) "Quantum key distribution for d -level systems with generalized Bell states". *Physical Review A*, 65. 052331.
- Kasirajan, V. (2021). "Fundamentals of Quantum Computing". *Springer Cham*.
- Kay, A. (2018) "Tutorial on the Quantikz Package". *arXiv: Quantum Physics*.
- Moore, G.E. (1965) "Cramming more components onto integrated circuits". *Reprinted from Electronics*, 38(8). 114
- Nielsen, M., ve Chuang, I. (2010). "Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge: Cambridge University Press.
- Oliveira, D.S. ve Ramos, R.M. (2007) "Quantum Bit String Comparator: Circuits and Applications" *Quantum Computers and computing*, 7(1). 17-26.
- Price, M.D. (1999) "Construction and Implementation of NMR Quantum Logic Gates for Two Spin Systems". *Journal of Magnetic Resonance*, 140. 371-378.
- Ruiz-Perez, L. ve García-Escartín, J.C. (2014) "Quantum arithmetic with the quantum Fourier transform". *Quantum Information Processing*, 16. 1-14.
- Sahin, E. (2020) "Quantum arithmetic operations based on quantum fourier transform on signed integers". *International Journal of Quantum Informa-*

tion, 2(13).

- Schumacher, B. (1995) "Quantum coding". *Physical review. A, Atomic, molecular, and optical physics*, 51(4). 2738-2747.
- Shannon, C.E. (1948) "A Mathematical theory of communication". *The Bell System Technical Journal*, 27(3). 379-423
- Shor, P.W. (1997) "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". *SIAM journal on computing*, 26(5). 1494-1509.
- Thomsen, M. K. vd. (2010) "Reversible arithmetic logic unit for quantum arithmetic". *Journal of Physics A: Mathematical and Theoretical*, 43(38).
- Turing, A.M. (1937) "On Computable Numbers With an Application to the Entscheidungsproblem". *Proc. London Math. Soc.*, 42(2). 230-265.
- Vedral, V. vd. (1995) "Quantum Networks for Elementary Arithmetic Operations". *Physical Review A.*, 54.147.
- Wang, Y.vd. (2020) "Qudits and High-Dimensional Quantum Computing". *Frontiers of Physics*, 8.
- Yuan, Y. vd. (2023) "An Improved QFT-Based Quantum Comparator and Extended Modular Arithmetic Using One Ancilla Qubit". *New Journal of Physics*, 25.
- Zhou, S.S. vd. (2017) "Quantum Fourier transform in computational basis". *Quantum Inf Process*, 16(82).

# EKLER

## Ek.1: Dört Girişli Bir QFT Devresinin QASM Kodları

```
1 OPENQASM 2.0;
2 include "qelib1.inc";
3 qreg q[4];
4 creg c[4];
5 h q[0];
6 cp(pi/2) q[1], q[0];
7 cp(pi/4) q[2], q[0];
8 cp(pi/8) q[3], q[0];
9 barrier q[0], q[1], q[2], q[3];
10 h q[1];
11 cp(pi/2) q[2], q[1];
12 cp(pi/4) q[3], q[1];
13 barrier q[0], q[1], q[2], q[3];
14 h q[2];
15 cp(pi/2) q[3], q[2];
16 h q[3];
17 barrier q[0], q[1], q[2], q[3];
18 swap q[0], q[3];
19 swap q[1], q[2];
20 barrier q[0], q[1], q[2], q[3];
21 measure q[3] -> c[3];
22 measure q[0] -> c[0];
23 measure q[1] -> c[1];
24 measure q[2] -> c[2];
```

## Ek.2: İki Girişli Bir Kukuart Sistem için QFT Devresinin Python Kodları

```
1 import numpy as np
2 from math import pi
3 import math
4 from math import sqrt
5 import matplotlib.pyplot as plt
6
7 d = int(input(" d seviyesini giriniz="))
8 n = int(input(" n, giris sayinizi giriniz="))
9 listem=[int(input("girisler")) for i in range(n)]
10 print(".....")
11 print("girisleriniz",listem)
12 print(".....")
13 b=[listem[i]*pow(d,(n-1-i)) for i in range(n)]
14 c=sum(b)
15 print("girisinizin onluk sistemdeki karsiligi=",c)
16
17 stt = np.identity(d**n)
18 input_state = [stt[x,c]for x in range(d**n)]
19 input_state = np.reshape(input_state,(d**n,1))
20 print(input_state)
21
22 k = round(math.cos(2*pi/(d**n)),2)
23 l = round(math.sin(2*pi/(d**n)),2)
24 a = k + complex(1j)*l
25
26 m=[[a**((x*y)%(d**n))]for x in range(d**n)]for y in
27 range(d**n)]
28
29 qft = np.reshape(m,(d**n,d**n))
30
31 result = m[c]
32 print(result)
```

```

33 result_matrix = np.reshape(result,(d**n,1))
34 print(result_matrix)
35
36 resultkare = [result[i]*result[i].conjugate()/(sqrt(d**n)**2)
37 for i in range(d**n)]
38 print(resultkare)
39
40 states = []
41
42 for a in range(d**n):
43     f=""
44     while a>0:
45         b=a/d
46         b=int(b)
47         c=d*b
48         h=a-c
49         e=str(h)
50         a=b
51         f=e+f
52     g=str(f).zfill(n)
53     states.append(g)
54 print(states)
55
56 qft_s = np.dot(qft,input_state)
57 print(qft_s)
58
59 plt.bar(states,resultkare,color='black',width=0.4)

```

**Ek.3: Dört Adet İki Bitlik Sayının Toplama İşlemini Gerçekleştiren  
QFT Tabanlı Toplama Devresinin QASM Kodları**

```
1
2 OPENQASM 2.0;
3 include "qelib1.inc";
4
5 qreg q[10];
6 creg c[4];
7 x q[2];
8 x q[3];
9 x q[4];
10 x q[7];
11 x q[8];
12 barrier q[0],q[1],q[2],q[3],q[4],q[5],q[6],q[7],q[8],q[9];
13 h q[0];
14 cp(pi / 2) q[1], q[0];
15 cp(pi / 4) q[2], q[0];
16 cp(pi / 8) q[3], q[0];
17 barrier q[0], q[1], q[2], q[3];
18 h q[1];
19 cp(pi / 2) q[2], q[1];
20 cp(pi / 4) q[3], q[1];
21 barrier q[0], q[1], q[2], q[3];
22 h q[2];
23 cp(pi / 2) q[3], q[2];
24 barrier q[0], q[1], q[2], q[3];
25 h q[3];
26 swap q[1], q[2];
27 swap q[0], q[3];
28 barrier q[0],q[1],q[2],q[3],q[4],q[5],q[6],q[7],q[8],q[9];
29 cp(pi / 4) q[4], q[3];
30 cp(pi / 8) q[5], q[3];
31 barrier q[3], q[4], q[5], q[6];
32 cp(pi / 2) q[4], q[2];
```

```

33 cp(pi / 4) q[5], q[2];
34 barrier q[2], q[3], q[4], q[5];
35 cp(pi) q[4], q[1];
36 cp(pi / 2) q[5], q[1];
37 cp(pi) q[5], q[0];
38 barrier q[0], q[1], q[2], q[3], q[4], q[5], q[6], q[7], q[8], q[9];
39 cp(pi / 4) q[6], q[3];
40 cp(pi / 8) q[7], q[3];
41 barrier q[3], q[4], q[5], q[6];
42 cp(pi / 2) q[6], q[2];
43 cp(pi / 4) q[7], q[2];
44 barrier q[1], q[2], q[3], q[4], q[5];
45 cp(pi) q[6], q[1];
46 cp(pi / 2) q[7], q[1];
47 cp(pi) q[7], q[0];
48 barrier q[0], q[1], q[2], q[3], q[4], q[5], q[6], q[7], q[8], q[9];
49 barrier q[0], q[1], q[2], q[3], q[4], q[5], q[6], q[7], q[8], q[9];
50 cp(pi / 4) q[8], q[3];
51 cp(pi / 8) q[9], q[3];
52 barrier q[1], q[2], q[3], q[4], q[5], q[6], q[7], q[8];
53 cp(pi / 2) q[8], q[2];
54 cp(pi / 4) q[9], q[2];
55 barrier q[1], q[2], q[3], q[4], q[5], q[6], q[7], q[8];
56 cp(pi) q[8], q[1];
57 cp(pi / 2) q[9], q[1];
58 cp(pi) q[9], q[0];
59 barrier q[0], q[1], q[2], q[3], q[4], q[5], q[6], q[7], q[8], q[9];
60 swap q[0], q[3];
61 swap q[1], q[2];
62 h q[3];
63 cp(pi / 2) q[3], q[2];
64 h q[2];
65 barrier q[0], q[1], q[2], q[3];
66 cp(pi / 4) q[3], q[1];

```

```
67 cp(pi / 2) q[2], q[1];
68 h q[1];
69 barrier q[0], q[1], q[2], q[3];
70 cp(pi / 8) q[3], q[0];
71 cp(pi / 4) q[2], q[0];
72 cp(pi / 2) q[1], q[0];
73 h q[0];
74 barrier q[0], q[1], q[2], q[3], q[4], q[5], q[6], q[7], q[8], q[9];
75 measure q[0] -> c[3];
76 measure q[1] -> c[2];
77 measure q[2] -> c[1];
78 measure q[3] -> c[0];
```

**Ek.4: İki Adet Bir Bitlik Sayının Karşılaştırılmasını Gerçekleştiren Devrenin QASM Kodları**

```
1
2 OPENQASM 2.0;
3 include "qelib1.inc";
4
5 qreg q[5];
6 creg c[3];
7 x q[0];
8 barrier q[0], q[1], q[2], q[3], q[4];
9 x q[1];
10 ccx q[0], q[1], q[2];
11 x q[1];
12 barrier q[0], q[1], q[2], q[3], q[4];
13 x q[0];
14 ccx q[0], q[1], q[3];
15 x q[0];
16 barrier q[0], q[1], q[2], q[3], q[4];
17 x q[0];
18 x q[1];
19 ccx q[0], q[1], q[4];
20 x q[0];
21 x q[1];
22 barrier q[0], q[1], q[2], q[3], q[4];
23 ccx q[0], q[1], q[4];
24 barrier q[0], q[1], q[2], q[3], q[4];
25 measure q[2] -> c[2];
26 measure q[3] -> c[1];
27 measure q[4] -> c[0];
```

## Ek.5: Üç Bitlik Sayaç Devresinin QASM Kodları

```
1
2 OPENQASM 2.0;
3 include "qelib1.inc";
4
5 qreg q[3];
6 creg c[3];
7 x q[0];
8 x q[2];
9 barrier q[0], q[1], q[2];
10 cx q[2], q[1];
11 cx q[1], q[2];
12 barrier q[0], q[1], q[2];
13 x q[1];
14 cx q[1], q[2];
15 x q[1];
16 barrier q[0], q[1], q[2];
17 x q[1];
18 x q[2];
19 ccx q[2], q[1], q[0];
20 x q[1];
21 x q[2];
22 barrier q[0], q[1], q[2];
23 measure q[0] -> c[2];
24 measure q[1] -> c[1];
25 measure q[2] -> c[0];
```

## Ek.6: 5/2 İşleminin Qiskit Kodları

```
1
2 # gerekli kutuphaneleri cagiriyoruz.
3
4 from qiskit import QuantumCircuit,BasicAer,execute,
5 ClassicalRegister,Aer
6 from qiskit.visualization import plot_histogram
7 from math import pi
8 from qiskit.circuit.library.standard_gates.x import XGate
9
10 qc = QuantumCircuit(16,8)
11 ccx10 = XGate().control(2,None,'10')
12 ccx00 = XGate().control(2,None,'00')
13 cx0 = XGate().control(1,None,'0')
14
15 qc.x(0)
16 qc.x(2)
17 qc.x(3)
18 qc.barrier()
19 qc.draw('mpl')
20
21 # 101 durumuna QFT uygulayal m
22
23 qc.h(0)
24 qc.cp(pi/2,1,0)
25 qc.cp(pi/4,2,0)
26 qc.h(1)
27 qc.cp(pi/2,2,1)
28 qc.h(2)
29 qc.swap(0,2)
30 qc.barrier()
31 qc.draw('mpl')
32
33 #cikarici parca
```

```

34
35 qc.cp(-pi/2,3,2)
36 qc.cp(-pi/4,4,2)
37 qc.cp(-pi,3,1)
38 qc.cp(-pi/2,4,1)
39 qc.cp(-pi,4,0)
40 qc.barrier()
41 qc.draw('mpl')
42
43 #Simdi IQFT'yi uygulayalim
44
45 qc.swap(0,2)
46 qc.h(2)
47 qc.cp(-pi/2,2,1)
48 qc.h(1)
49 qc.cp(-pi/4,2,0)
50 qc.cp(-pi/2,1,0)
51 qc.h(0)
52 qc.barrier()
53 qc.draw('mpl')
54
55 #karsilastirici kismi olustural m
56 # 101 - 10 = 011'dir. Burada 11 ile 10 karsilastirilacak.
57
58 qc.append(ccx10,[3,1,5])
59 qc.append(ccx10,[1,3,6])
60 qc.ccx(1,3,7)
61 qc.append(ccx00,[1,3,7])
62 qc.barrier()
63
64 qc.append(ccx10,[4,2,8])
65 qc.append(ccx10,[2,4,9])
66 qc.ccx(2,4,10)
67 qc.append(ccx00,[2,4,10])

```

```
68 qc.barrier()
69
70 qc.cx(5,11)
71 qc.ccx(7,8,11)
72 qc.cx(6,12)
73 qc.ccx(7,9,12)
74 qc.ccx(7,10,13)
75 qc.barrier()
76 qc.draw('mpl')
77
78 olcme islemi yapalim
79
80 qc.measure(0,7)
81 qc.measure(1,6)
82 qc.measure(2,5)
83
84 qc.measure(11,4)
85 qc.measure(12,3)
86 qc.measure(13,2)
87 qc.draw('mpl')
88
89 #kalanin bolenden kucuk oldugu durum icin devam edelim
90 asagida sayma islemini gerceklestiren kodlar bulunmaktadir.
91
92 qc.cx(15,14). c_if(4,1)
93 qc.cx(14,15). c_if(4,1)
94 qc.x(14). c_if(4,1)
95 qc.cx(14,15). c_if(4,1)
96 qc.x(14). c_if(4,1)
97
98 qc.measure(14,1)
99 qc.measure(15,0)
100 qc.draw('mpl')
101
```

```

102 simulator = Aer.get_backend('qasm_simulator')
103 job = execute(qc, simulator, shots=1000)
104 result = job.result()
105 counts = result.get_counts(qc)
106 print(counts)
107
108 #islem bitmedigi icin karsilastirici sonucu veren kısmi resetliyoruz.
109
110 qc.reset(11)
111 qc.reset(12)
112 qc.reset(13)
113
114 #ikinci cikarma islemini yapalim.
115
116 qc.h(0). c_if(4,1)
117 qc.cp(pi/2,1,0). c_if(4,1)
118 qc.cp(pi/4,2,0). c_if(4,1)
119 qc.h(1). c_if(4,1)
120 qc.cp(pi/2,2,1). c_if(4,1)
121 qc.h(2). c_if(4,1)
122 qc.swap(0,2). c_if(4,1)
123 qc.barrier()
124
125 qc.cp(-pi/2,3,2). c_if(4,1)
126 qc.cp(-pi/4,4,2). c_if(4,1)
127 qc.cp(-pi,3,1). c_if(4,1)
128 qc.cp(-pi/2,4,1). c_if(4,1)
129 qc.cp(-pi,4,0). c_if(4,1)
130 qc.barrier()
131
132 qc.swap(0,2). c_if(4,1)
133 qc.h(2)
134 qc.cp(-pi/2,2,1). c_if(4,1)
135 qc.h(1). c_if(4,1)

```

```
136 qc.cp(-pi/4,2,0). c_if(4,1)
137 qc.cp(-pi/2,1,0). c_if(4,1)
138 qc.h(0). c_if(4,1)
139 qc.barrier()
140 qc.draw('mpl')
141
142 # tekrar olcme islemi yapalim
143
144 qc.measure(0,7)
145 qc.measure(1,6)
146 qc.measure(2,5)
147
148 qc.measure(11,4)
149 qc.measure(12,3)
150 qc.measure(13,2)
151 qc.draw('mpl')
152
153 # tekrar karsilastiralim
154
155 qc.append(ccx10,[3,1,5])
156 qc.append(ccx10,[1,3,6])
157 qc.ccx(1,3,7)
158 qc.append(ccx00,[1,3,7])
159 qc.barrier()
160
161 qc.append(ccx10,[4,2,8])
162 qc.append(ccx10,[2,4,9])
163 qc.ccx(2,4,10)
164 qc.append(ccx00,[2,4,10])
165 qc.barrier()
166
167 qc.cx(5,11)
168 qc.ccx(7,8,11)
169 qc.cx(6,12)
```

```

170 qc.ccx(7,9,12)
171 qc.ccx(7,10,13)
172 qc.barrier()
173 qc.draw('mpl')
174
175 qc.measure(0,7)
176 qc.measure(1,6)
177 qc.measure(2,5)
178
179 qc.measure(11,4)
180 qc.measure(12,3)
181 qc.measure(13,2)
182 qc.draw('mpl')
183
184 # kalan bolenden küçük olduğu için son kez sayalım.
185
186 qc.cx(15,14). c_if(3,1)
187 qc.cx(14,15). c_if(3,1)
188 qc.x(14). c_if(3,1)
189 qc.cx(14,15). c_if(3,1)
190 qc.x(14). c_if(3,1)
191
192 qc.measure(14,1)
193 qc.measure(15,0)
194 qc.draw('mpl')
195
196 simulator = Aer.get_backend('qasm_simulator')
197 job = execute(qc, simulator, shots=1000)
198 result = job.result()
199 counts = result.get_counts(qc)
200 print(counts)
201
202 output:
203 {'00101010': 1000}

```

# ÖZGEÇMİŞ

Murat KURT, Samsun'un Bafra ilçesinde Yabancı Dil Ağırlıklı Bafra Lisesi'ni bitirdikten sonra Ondokuz Mayıs Üniversitesi Fen Edebiyat Fakültesi, Fizik bölümünden 2010 yılında mezun oldu. 2017 yılında OMÜ LEE Fizik Ana Bilim Dalında Yüksek Lisansını tamamladı. 2019 yılında aynı üniversitede doktora programına başladı. Mezuniyetinden bu yana bazı özel kurumlarda fizik eğitmeni olarak görev yapan Murat KURT, orta derecede İngilizce bilmektedir. Makine öğrenmesi ve yapay zeka uygulama alanlarına ilgilidir.

## İletişim Bilgileri

ORCID ID : 0000-0003-1091-5952

## Yayımlar:

1. Çakmak, S., Kurt, M., ve Gençten, A. (2023), "Quantum Fourier Transform-Based Arithmetic Logic Unit on a Quantum Processor". *Ann. Phys.* Early Access
2. Kurt, M. ve Gençten, A. (2021) "A Simulation of Quantum Fourier Transform for Qudit Systems", 4<sup>th</sup> International Conference on Physical Chemistry and Functional Materials (PCFM'21) 63-66
3. Kurt, M. ve Gençten, A. (2021) "A Generalized Quantum Addition Circuit Based on Quantum Fourier Transform", 37<sup>th</sup> International Physics Congress TPS37, Vol.03, No.02, pp.13-17.
4. Kurt, M., Kaltehei, A., Çakmak, S., ve Gençten, A. (2023) "Construction of the NAND Gate in Quantum Computing System Using Quantum Fourier Transform", 3<sup>rd</sup> International Black Sea Modern Scientific Research Congress
5. Kurt, M., Kaltehei, A., Çakmak, S., ve Gençten, A. (2023) "The Constructions of Up and Down Quantum Counter Circuits for Qudit Systems", 39<sup>th</sup> International Physics Congress TPS37, Vol.05, No.02, 37-41.

## Kazanılan Ödüller, Teşvikler ve Burslar

Bu çalışma 122F298 numaralı TÜBİTAK projesi ile desteklenmiştir.