

**SÜPERİLETKEN ENTEGRE DEVRELER İÇİN ÇOK KANALLI, KARARLI,
DÜŞÜK MALİYETLİ, DÜŞÜK GÜRÜLTÜLÜ, YÜKSEK HASSASİYETLİ,
GENİŞ ARALIKLI AKIM KAYNAĞI GELİŞTİRİLMESİ**

UFUK YUMRUKAYA

YÜKSEK LİSANS TEZİ

ELEKTRİK VE ELEKTRONİK MÜHENDİSLİĞİ

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

NİSAN 2015

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Osman EROĞUL

Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Prof. Dr. Murat ALANYALI

Anabilim Dalı Başkanı

Ufuk YUMRUKAYA tarafından hazırlanan “SÜPERİLETKEN ENTEGRE DEVRELER İÇİN ÇOK KANALLI, KARARLI, DÜŞÜK MALİYETLİ, DÜŞÜK GÜRÜLTÜLÜ, YÜKSEK HASSASİYETLİ, GENİŞ ARALIKLI AKIM KAYNAĞI GELİŞTİRİLMESİ” adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Doç. Dr. Ali BOZBEY

Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Prof. Dr. Mehmet Önder EFE

Üye : Doç. Dr. Ali BOZBEY

Üye : Doç. Dr. Oğuz ERGİN

TEZ BİLDİRİMİ

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

Ufuk YUMRUKAYA

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Elektrik ve Elektronik Mühendisliği
Tez Danışmanı : Doç. Dr. Ali BOZBEY
Tez Türü ve Tarihi : Yüksek Lisans – Nisan 2015

Ufuk YUMRUKAYA

**SÜPERİLETKEN ENTEGRE DEVRELER İÇİN ÇOK KANALLI, KARARLI,
DÜŞÜK MALİYETLİ, DÜŞÜK GÜRÜLTÜLÜ, YÜKSEK HASSASİYETLİ,
GENİŞ ARALIKLI AKIM KAYNAĞI GELİŞTİRLMESİ**

ÖZET

Tek akı kuantum (SFQ) mantık devreleri, icadından bu yana, birçok uygulamada kullanılarak gelişmekte olan teknolojilerden biri haline gelmiştir. SFQ devrelerin çekirdeğini Josephson eklemi oluşturmaktadır. Bu nedenle SFQ devreler akım beslemeli devrelerdir. SFQ devreler, Josephson eklemelerini anahtarlama aracı olarak kullanırlar, bu sayede daha az güç ile daha hızlı işlemler gerçekleştirirler. Karmaşık SFQ devrelerde termal yüke bağlı olan Joule ısınma etkisinin düşük olması için devreye birçok noktadan akım beslemesi gerekmektedir. Bu akım beslemesi için gerekli olan seviye, kapı seviyesindeki birimler için μA mertebesindeyken, mikroişlemci devreler için birkaç A mertebelerinde olabilir. Oda sıcaklığında çalışan yarıiletken devreler voltaj kutuplamalı devrelerdir. Dolayısı ile termal yük problemleri ve çok noktadan akım beslemesine ihtiyaçları yoktur. Bu nedenle piyasadan, süperiletken devreler için, kompakt, düşük gürültülü, çok kanallı ve geniş aralıklı gibi özellikleri olan bir akım kaynağı bulmak çok mümkün değildir. Bu çalışmada, süperiletken entegre devrelerin akım beslemeleri için, düşük gürültülü, yüksek hassasiyetli ve geniş aralıklı akım kaynağı geliştirilmiştir. Burada akım kaynağından daha kararlı ve düşük gürültülü bir akım çıkışı elde etmek için geribesleme algoritması kullanılmıştır. Akım kaynağının çıkışları FPGA ile kontrol edilmektedir. Kullanıcı istediği akımı bilgisayar üstündeki bir arayüzden girebilmektedir. Bilgisayar kullanıcının istediği akım için gerekli parametreleri hesaplayıp FPGA'ye gönderir. Bu çalışmada, 1 mA akım çıkışı için, 1.5 mHz'den 10 Hz'e kadar ölçülen toplam gürültü akımı $13.9 \text{ nA}/\sqrt{\text{Hz}}$ 'dir.

Anahtar Kelimeler: Akım kaynağı, süperiletken entegre devreler, SFQ devreler.

University : TOBB Economics and Technology University
Institute : Institute of Natural and Applied Sciences
Science Program : Electrical and Electronics Engineering
Supervisor : Associate Professor Dr. Ali BOZBEY
Degree Awarded and Date : M.Sc. – Nisan 2015

Ufuk YUMRUKAYA

**DEVELOPMENT OF A MULTI-CHANNEL, STABLE, LOW-COST, LOW-
NOISE, HIGH PRECISION, WIDE RANGE CURRENT SOURCE FOR
SUPERCONDUCTING INTEGRATED CIRCUITS**

ABSTRACT

Since the discovery of single flux quantum (SFQ) logic circuits, SFQ logic became one of the emerging technologies, being used in numerous applications. As the core element of an SFQ circuit is Josephson junction, they are current biased circuits. SFQ logic circuits provide faster operations with lower power consumption, using Josephson junctions as the switching devices. Complicated circuits need to be biased from many points in order to decrease the Joule heating effect related thermal load (I^2R) as the current source is placed at the room temperature whereas the SFQ circuits are at the cryogenic temperatures. In addition, the requirements of the range of current bias may be in the order of μA as in the case of 1-bit comparator cell to a range of few Amperes as in the case of a microprocessor circuit. Semiconductor circuits operating at room temperatures do not have such requirements as they are voltage biased and they do not have a thermal load problem. Due to the small market associated to superconducting integrated circuits, it is not easy to find a compact, low noise, multi-channel and computer controller compact current sources sold off the shelf. So, in this study, we develop a low noise, high precision, wide range current source for the biasing of superconducting integrated circuits. Feedback algorithm is used to increase the stability and decrease the noise amount of the current source. Channels of the current source are controlled with an FPGA. Desired current can be set via the user interface which is running on the PC. The PC sends the required commands to FPGA, which controls the high speed feedback algorithm. For a 1 mA setting, the measured noise current is $13.9 \text{ nA}/\sqrt{\text{Hz}}$, in a bandwidth of 1.5 mHz to 10 Hz.

Keywords: Current source, superconducting integrated circuits, SFQ circuits.

TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren baőta danıőmanım Doç. Dr. Ali Bozbey'e, tez çalıőmalarımda teknik olarak bana yardımcı olan Mustafa İNCİ, Mustafa ÇALLI ve Sasan RAZMKHAH'a, sunduęu imkanlardan ve sağladıęı araőtırma bursundan ötürü TOBB Ekonomi ve Teknoloji Üniversitesi'ne ve her türlü desteęi ile hep yanımda olan eőim Radiye'ye teőekkürü bir borç bilirim.

Yapılan çalıőmalar, 111E191 nolu TÜBİTAK projesi tarafından desteklenmektedir.

İÇİNDEKİLER

ÖZET	iii
ABSTRACT	iv
TEŞEKKÜR	v
İÇİNDEKİLER.....	vi
ÇİZELGELERİN LİSTESİ	ix
ŞEKİLLERİN LİSTESİ	x
KISALTMALAR.....	xii
1. GİRİŞ.....	1
1.1 Süperiletkenlik	2
1.2 Josephson Etkisi.....	6
1.3 Tek Akı Kuantumu (Single Flux Quantum) Devreler.....	7
1.4 Akım Kaynağı Devreleri	9
1.4.1 Akım Aynası.....	9
1.4.2 Baz Akımı Kompanzasyonlu Akım Aynası	10
1.4.3 Widlar Akım Kaynağı	11
1.4.4 Wilson Akım Kaynağı.....	12
1.4.5 Gerilim Kontrollü Akım Kaynağı.....	12
2. SÜPERİLETKEN DEVRELER İÇİN AKIM KAYNAĞI TASARIMI.....	14
2.1 Akım Kaynağı İçin Gerekli Güç Beslemesi	15
2.2 Çıkış Akımı Aralığı.....	15
2.3 Çıkış Voltaj Aralığı	16
2.4 Çıkış Akımı Kararlılığı.....	16
2.5 Bilgisayar Arayüzünden Akım Çıkışı Programlanabilirliği	16
2.6 Gürültü.....	17
2.7 Kısa Süreli Tepki (Transient Response).....	17
2.8 Fiziksel Boyutlar	18

3.	ELEKTRONİK DEVRELERDE GÜRÜLTÜ VE HESAPLAMALARI	19
3.1	Saçma Gürültüsü (Shot Noise).....	19
3.2	Termal Gürültü (Thermal Noise)	19
3.3	Titreşim Gürültüsü (Flicker Noise, 1/f Noise).....	20
3.4	Patlak Gürültü (Burst Noise)	20
3.5	Gürültü Karakteristiği.....	20
3.6	Gürültü Kaynaklarının Toplanması.....	21
3.7	Gürültü Spektrumu	23
3.8	İstenen Frekansta Toplam Gürültü (Integrated Noise) Hesaplamaları.....	23
4.	TASARIM	24
4.1	Sisteme Genel Bakış.....	24
4.2	Akım Kaynağı Donanımı.....	25
4.2.1	Akım Kaynağı Kanalları Donanımı	26
4.2.1.1	Güç Besleme Devreleri	26
4.2.1.2	DAC Devresi	27
4.2.1.3	Akım Kontrol Devresi.....	29
4.2.1.4	ADC Devresi	29
4.2.1.5	İzolatör Devreleri.....	31
4.2.2	Ortak Devreler	32
4.2.2.1	Ortak Kısım Güç Besleme Devreleri	32
4.2.2.2	FPGA Devreleri	32
4.2.2.3	Osilatör Devresi	33
4.2.2.4	Seri Port Dönüşüm Devresi	34
4.3	Akım Kaynağı Yazılımı	34
4.3.1	FPGA Yazılımı	35
4.3.2	Arayüz Yazılımı.....	37
5.	GELİŞTİRİLEN AKIM KAYNAĞININ TEST EDİLMESİ.....	38

5.1	DC Akım Kaynađı için Grlt Hesapları	38
5.2	Test Sonuları	39
6.	SONU.....	44
	KAYNAKLAR.....	46
	EKLER	51
	ZGEMİŐ.....	65

ÇİZELGE LİSTESİ

Çizelge 1.1: Bazı bilinen elementlerin süperiletkenliğe geçiş sıcaklıkları	3
Çizelge 5.1: Farklı akım çıkışları için hesaplanan toplam gürültüler	43

ŞEKİL LİSTESİ

Şekil 1.1: Heike Kamerlingh Onnes'in çizimiyle cıvanın sıcaklık direnç ilişkisi.....	3
Şekil 1.2: Kritik akım, kritik sıcaklık ve kritik manyetik alan ile süperiletkenlik bölgesinin tanımlanması.....	4
Şekil 1.3: Meissner Etkisi – Mükemmel iletken ile süperiletken karşılaştırması.....	5
Şekil 1.4: Josephson Ekleme.....	7
Şekil 1.5: SFQ Darbesi.....	8
Şekil 1.6: Josephson ekleminin anahtarlanması durumunda a)akım-gerilim b)faz gerilim grafikleri.....	8
Şekil 1.7: npn BJT transistörlerle kurulan akım aynası devresi.....	9
Şekil 1.8: Çok çıkışlı akım aynası.....	10
Şekil 1.9: Baz akımı kompanzasyonlu akım aynası.....	10
Şekil 1.10: Çok çıkışlı baz akımı kompanzasyonlu akım aynası.....	11
Şekil 1.11: Widlar akım kaynağı.....	11
Şekil 1.12: Wilson akım kaynağı.....	12
Şekil 1.13: Voltaj kontrollü akım kaynağı.....	13
Şekil 3.1: Gürültü genliğinin Gaussian dağılımı.....	21
Şekil 3.2: R1 ve R2 dirençleri gürültü modelleri.....	22
Şekil 4.1: Sistem Akış Diyagramı.....	24
Şekil 4.2: Akım kontrol devresi şeması.....	25
Şekil 4.3: Akım kaynağı donanımı.....	26
Şekil 4.4: Besleme izolasyon devresi.....	26
Şekil 4.5: Birinci kanal için güç besleme devreleri.....	27
Şekil 4.6: Kanal 1 için DAC Devresi.....	28
Şekil 4.7: Kanal 1 için ADC Devresi.....	30
Şekil 4.8: Kanal 1 için izolasyon devreleri.....	31
Şekil 4.9: Ortak kısım güç besleme devreleri.....	32

Şekil 4.10: FPGA devreleri	33
Şekil 4.11: Osilatör devresi	34
Şekil 4.12: Seri port dönüşüm devresi	34
Şekil 4.13: Geribesleme algoritmasının akış şeması	36
Şekil 4.14: Akım kaynağı kullanıcı arayüzü	37
Şekil 5.1: 0,1 μ A akım çıkışının 100 Ω üzerindeki yaklaşık 1 saatlik verisi.....	39
Şekil 5.2: 1 μ A akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi.....	40
Şekil 5.3: 10 μ A akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi.....	40
Şekil 5.4: 100 μ A akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi.....	40
Şekil 5.5: 1000 μ A akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi....	41
Şekil 5.6: 0,1 μ A akım çıkış verisinin akım gürültü spektrumu.....	41
Şekil 5.7: 1 μ A akım çıkış verisinin akım gürültü spektrumu	41
Şekil 5.8: 10 μ A akım çıkış verisinin akım gürültü spektrumu.....	42
Şekil 5.9: 100 μ A akım çıkış verisinin akım gürültü spektrumu.....	42
Şekil 5.10: 1000 μ A akım çıkış verisinin akım gürültü spektrumu.....	42

KISALTMALAR

ADC	Analog dijital çevirici (Analog to digital converter)
BJT	Çift kutup yüzeyli transistor (Bipolar Junction Transistor)
CMOS	Tamamlayıcı metal oksit yarıiletkeni (Complementary Metal oxide semiconductor)
DAC	Dijital analog çevirici (Digital to analog converter)
HDL	Donanım tanımlama dili (Hardware description language)
RSFQ	Hızlı tek akı kuantumu (Rapid single flux quantum)
SFQ	Tek akı kuantumu (Single flux quantum)
SQUID	Süperiletken kuantum girişim cihazı (Superconducting quantum interference device)
VHDL	Çok hızlı donanım tanımlama dili (Very high speed hardware description language)
VLSI	Çok büyük çapta tümleşim (Very large scale integration)

1. GİRİŞ

Ana parçalarını Josephson eklemlerinin [1] oluşturduğu tek akı kuantumu mantık devreleri (Single Flux Quantum, SFQ), çok düşük güç tüketimleriyle [2] yüksek hızlarda, 100 GHz üstü, çalışabilme yeteneğine sahiptir. Bu yeni teknoloji analog-dijital çeviriciler, mikroişlemciler ve ağ anahtarları gibi dijital sistemlerin performanslarının geliştirilmesini sağlar. Şimdiye kadar bunlar gibi birçok farklı karmaşık devrelerin birçok farklı gösterimleri yapılmıştır [3]–[8]. Bunlara ek olarak, yakın zamanda, SFQ tabanlı algılayıcı okuma devre uygulamalarında da artış olmuştur [9]–[12].

CMOS ve SFQ entegre devreleri arasında, kutuplama şemaları açısından temel farklılıklar vardır [13]. CMOS devreleri voltaj kutuplamalı devreler iken SFQ devreler ve Josephson eklemleri akım kutuplamalı devrelerdir [14]. Büyük SFQ devreler için, akım beslemesi 1 A değerini geçebilir, ancak bu devreler, büyük kısımları süperiletken durumda ve kutuplamalı voltajları 2.5 mV seviyelerinde olduğundan, az güç harcarlar. Buna ek olarak, SFQ devreleri μT değerinden daha küçük manyetik alanlar altında çalışmaktadırlar. Bu sebeple test sırasında Dünyanın manyetik alanı yüksek μ materyallerden yapılan kalkanlar kullanılarak azaltılmalıdır, fakat şu da unutulmamalıdır ki manyetik alanlar akım beslemelerinden de kaynaklanır. Bu yüzden, devre için 2 A akım beslemesi gerekse bile bu akım tek kablodan sağlanmamalı, manyetik alan etkisini azaltmak için birçok paralel kablodan sağlanmalıdır. Genel olarak, tasarımcı her akım besleme hattını, negatif akım giderici ile birlikte 100 mA değerinin altında tutar. Dolayısı ile bir tek entegre devrede bile, birçok akım beslemesi gerekebilir. Bu konfigürasyon aynı zamanda, özellikle kapalı döngü soğutucu tabanlı sistemlerin kablolarındaki Joule ısınma etkisini düşürmeye de yardım eder. Büyük SFQ devreler için, klasik güç kaynakları kompaktlığından ve/veya maliyetinden feragat edilerek kullanılabilir. Ayrıca bu büyük SFQ devreler için geliştirilmiş halihazırda markette satılan ürünler bulunmaktadır [15]. ADC ve yakın zamanda gelişmekte olan detektör okuma devreleri [16], [17] gibi karışık sinyalli entegre devrelerin ön taraflarında, sadece besleme akımı olarak değil de tasarım sürecindeki test girişleri için de, düşük gürültülü, kararlı ve bilgisayar

kontrollü akım kaynakları gereklidir. Bu akım kaynağından istenen akım seviyeleri ~100 nA değerinden birkaç mA değerine kadar olmaktadır. Düşük gürültülü akım kaynakları önemli test ekipmanı firmaları tarafından üretilmektedir ancak aynı anda kullanılması gereken akım kaynağı sayısı göz önünde bulundurulduğunda ve bu ürünler SFQ test düzeneğinin bir parçası olarak düşünüldüğünde, ne fiyat olarak ne de kaplayacakları yer bakımından uygun olmaktadır. Diğer önemli akım kaynağı üreticileri lazer diyot sürücü şirketleridir ve yeteri kadar düşük akım ve/veya bilgisayar kontrolü sağlayamamaktadır. Bundan dolayı, SFQ tabanlı detektör okuyucu devrelerin akım beslemeleri ve test akımları için hala düşük seviyeli akım kaynağı geliştirilmesine ihtiyaç vardır.

SFQ devreler için kullanılacak akım kaynağı büyük bir aralığı sağlayabilir ve düşük gürültülü olmalıdır. Ve ayrıca, bu kaynağın akım çıkışının hassasiyeti ve gürültü miktarı zamana ve mekana göre farklılık göstermemelidir.

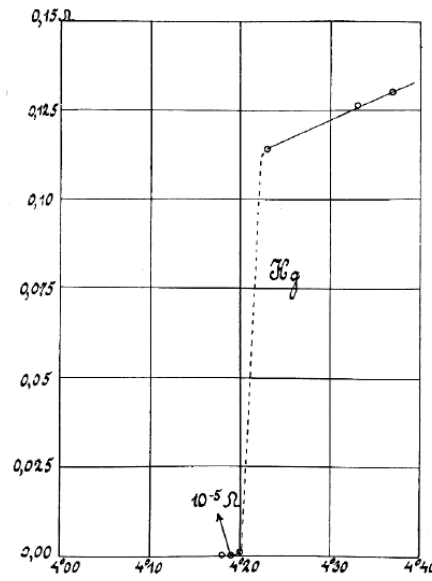
Çıkış akımındaki gürültü miktarının bir kısmı sistemin güç beslemesinden gelmektedir [18]. Bu yüzden devreye beslemeler genelde bataryalardan verilmektedir.

Başlangıçta akım kaynağını küçük ölçekli karışık devreler için kullanmayı hedeflememize rağmen, ana topolojiyi ve algoritmaları değiştirmeden direnç gibi bazı çevresel devre parametrelerini değiştirerek sistemin akım çıkışını yükseltmek mümkündür. Bu tezde, sağlanan çıkış akım aralığı 100 nA'den 1 mA'e kadardır ve bu aralıkta akım çıkışı 10 nA'lik adımlarla ayarlanabilir. Bütün kanallar test düzeneğimize entegre edilmeye uygun olan Labview programı üzerinden bağımsız olarak bilgisayardan kontrol edilebilmektedir.

1.1 Süperiletkenlik

1908 yılında Heike Kamerlingh Onnes, helyum gazını sıvılaştırmayı başardı ve sonrasında düşük sıcaklık ortamında testler gerçekleştirdi. 1911 yılında Cıva (Hg) üzerinde yaptığı deneylerde, cıva elementinin sıvı Helyum ile 4 Kelvin sıcaklığına

soğutulması esnasında, elektriksel direncinin 4.2 Kelvin dolaylarında aniden kaybolduğunu fark etmesi ile ilk süperiletkenlik fenomeni gözlemlendi [19]. Heike Kamerlingh Onnes süperiletkenliğe ulaşan malzemenin elektrik akımına sıfır direnç gösterdiğini buldu ve buluşu 1913 yılında kendisine Nobel Fizik ödülünü kazandırdı. Heike Kamerlingh Onnes'in cıvada süperiletkenlik geçişini gösteren çizimi Şekil 1.1'de sunulmuştur.



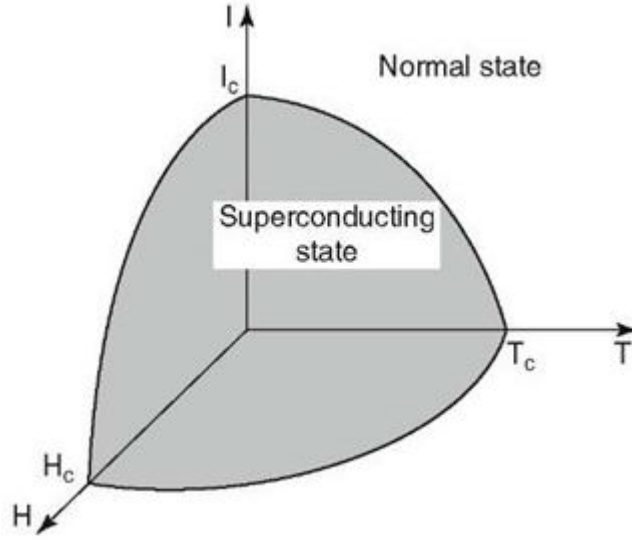
Şekil 1.1 Heike Kamerlingh Onnes'in çizimiyle cıvanın sıcaklık direnç ilişkisi [19]

Süperiletkenliğin 1911'deki keşfinden sonra, pek çok metalin direncinin, her metale özgü kritik bir T_c sıcaklığının altında, sıfıra gittiği gözlenmiştir. Bazı bilinen elementlerin kritik sıcaklıkları Çizelge 1.1'de verilmiştir.

Çizelge 1.1 Bazı bilinen elementlerin süperiletkenliğe geçiş sıcaklıkları

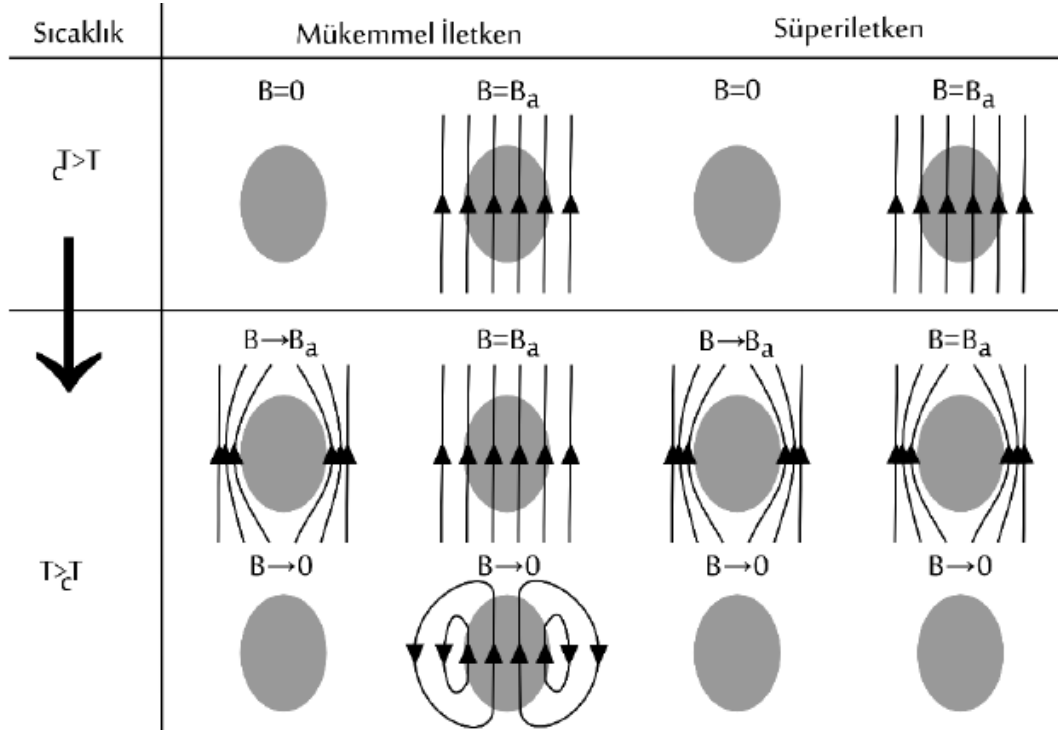
Element	T_c , °K	$H_c(0)$ (Tesla)
Al	1,2	0,0105
Ga	1,1	0,0058
In	3,4	0,0281
Pb	7,2	0,0803
Hg(α)	4,2	0,041
Nb	9,3	0,1991
Th	3,7	0,010
Sn	0,4	0,1023
W	5,3	0,0054
V	0,9	

Süperiletken malzemeler özelliklerini koruyabilmek için bazı kritik değerlere sahiptir. Bu değerler kritik sıcaklık (T_c), kritik akım yoğunluğu (J_c) ve kritik manyetik alan şiddeti (H_c) olarak ifade edilir. Kritik sıcaklık, kritik akım yoğunluğu ve kritik manyetik alan şiddeti değerlerinin birbirlerini etkilediği bulunmuştur. Mesela, süperiletken malzemenin kritik sıcaklık değeri, malzemenin üzerinden geçen akım yoğunluğuna ve üzerine uygulanan manyetik alan şiddetine bağlı olarak değişiklik gösterir. Bu üç kritik parametre arasındaki ilişki Şekil 1.2’de verilmiştir.



Şekil 1.2 Kritik akım, kritik sıcaklık ve kritik manyetik alan ile süperiletkenlik bölgesinin tanımlanması [20]

1933’de iki Alman fizikçi Meissner ve Ochsenfeld, süperiletkenlerin manyetik özelliklerini incelediler. Manyetik alanda soğutulan süperiletkenin manyetik akıyı dışarıladığını ve içine sokmadığını gözlemlediler (Şekil 1.4) [21]. Meissner Etkisi olarak adlandırılan bu mükemmel diamanyetizma özelliği sayesinde süperiletken malzemeler günümüzde; Maglev trenlerinde, yüksek verimli elektrik motorlarında ve diğer benzeri endüstriyel alanlarda kullanılmaktadır.



Şekil 1.3 Meissner Etkisi – Mükemmel iletken ile süperiletken karşılaştırması [22]

1957 yılında John Bardeen, Leon Neil Cooper ve John Robert Schrieffer tarafından geliştirilen ve 1972’de Nobel fizik ödülü verilen BCS teorisi, atomların fonon spektrumuyla elektronların etkileşimi sayesinde kısa süreli de olsa çiftler halinde hareket edebileceklerini ve atomlara çarpmadan madde içinde ilerleyebileceklerini gösterdi. Bu teori ile süperiletkenlik durumu teorik olarak ilk kez açıklanıyordu [23].

Brian David Josephson, 1962 yılında, aralarında ince yalıtkan bulunan iki süperiletken arasında Cooper Çiftlerinin iletimi gerçekleşebileceği teorisini ortaya sürmüştür [24]. Daha sonra ispatlanan bu durum Josephson Etkisi olarak adlandırılmış; oluşan yapıya Josephson Eklemleri (JJ) denilmiş ve bu öngörüsü ile Josephson, 1973 Nobel Fizik Ödülü’nü kazanmıştır. Bu tünelleme olgusunu deneysel olarak ispatlayan Iver Giaever ve Leo Esaki de, bu keşiften dolayı Nobel Fizik Ödülü’nü kazanmıştır. Josephson Eklemleri ile süperiletken elektronik uygulamalar gerçekleştirilebilir hale gelmiştir. SQUID ile 5×10^{-18} Tesla [25] hassasiyetinde manyetik ölçümler yapılabilir olmuş; sağlık sektörü, uzay araştırmaları ve yerküre incelemeleri başta olmak üzere endüstrinin birçok alanında ilerleme sağlamıştır. Ayrıca RSFQ

dijital elektronik teknolojisi ile süperiletkenler kullanılarak dijital sinyallerin işlenmesi mümkün olmuştur.

Süperiletkenlik konusunda bir Nobel Fizik ödülü de 1987 yılında seramik malzemelerde süperiletkenliği keşfederek o güne kadarki bilinen en yüksek kritik sıcaklığa, 30°K [26], ulaşan Georg Bednorz ve Alex Müller'e verilmiştir. Seramik malzemelerin iletken olmamasına rağmen süperiletken davranışı göstermesi, ilerleyen yıllarda diğer araştırmacıları daha yüksek sıcaklıklarda süperiletken davranışı sergileyen seramikleri sentezleme çalışmalarına yöneltmiştir.

CMOS teknolojili yüksek performanslı bilgisayar sistemlerinde enerji tüketimi ciddi bir sorun oluşturmaktadır [27]. Bu yüzden düşük enerji tüketimine sahip olan RSFQ teknolojili devreler, sonraki nesil VLSI teknolojisi olarak öngörülmektedir.

1.2 Josephson Etkisi

1961 yılında Brian Josephson, tek parçacık tünellemesine ek olarak, Cooper çiftlerinin de tünellenebileceği teorisini öne sürdü. Josephson etkisi, ince bir yalıtkan maddeyle ayrılmış iki süperiletken cisim arasında (Josephson Ekleme, Şekil 1.4), uygun koşullar sağlandığında gerçekleşen elektrik akımı oluşabilmesi durumudur. Çiftler hiçbir dirençle karşılaşmadan tünellenerek bir DC akım oluşturmaktadır. Bu akım hiçbir gerilim uygulanmazken de söz konusudur. Bu duruma DC Josephson etkisi denir. Burada geçen akım miktarı malzemelerdeki cooper çifti yoğunluğuna ve aralarındaki süperiletken olmayan malzemenin kalınlığına bağlıdır. Josephson'a göre de ekleme ayrıca bir gerilim uygulandığında bir AC akım ortaya çıkmaktadır. Bu duruma AC Josephson etkisi denir. Ekleme gerilim uygulanmasıyla Josephson akımı (1.1) de verilen frekans değeri ile salınır.

$$f = \frac{2eV_0}{\hbar} \quad (1.1)$$

Burada \hbar Planck sabiti, e de, bir elektronun negatif yüküdür. Bu iki değer de sabit olduğu için $2e / \hbar$ değeri de sabit bir sayı olarak hesaplanabilir. Bu değer 483,6

GHz/mV'a eşittir. AC Josephson Etkisi'nin bu özelliği, gerilim standardı belirlenmesi için kullanılabilir [28].



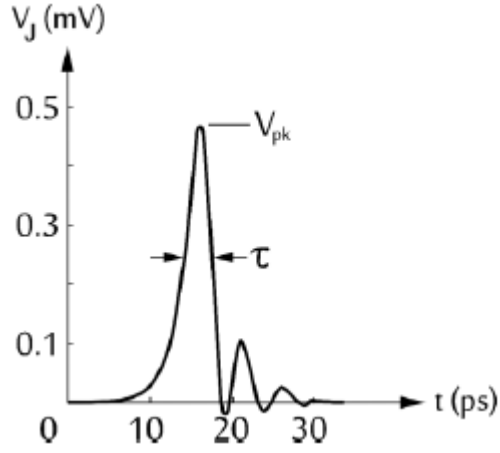
Şekil 1.4 Josephson Ekleme

1.3 Tek Akı Kuantumu (Single Flux Quantum) Devreler

SFQ, süperiletken devrelerdeki kuantum etkilerini kullanan sayısal elektronik teknolojisidir. Bu sayısal elektronik teknolojisi Josephson Sayısal Devreleri'ni kullanarak sayısal sinyalleri işler. Josephson Eklemleri, SFQ elektroniğinin aktif elemanlarıdır. Bu yarıiletken devrelerde transistörlere karşılık gelir. Buradaki sayısal bilgi, Josephson eklemlerinde üretilen manyetik akı kuantaları ile taşınır. Üretilen manyetik akı kuantaları pikosaniye süreli üretilen voltaj atımları ile süperiletken iletim hatları üzerinden taşınır. Pikosaniye kadar kısa süreli aktarımları nedeniyle SFQ devreler oldukça hızlı çalışır. 1985 yılında Josephson Dijital Devreleri'ndeki bilginin, yarıiletkenler teknolojilerdeki gibi sadece voltaj ile ifade edilmediği saptanmıştır [2]. SFQ devrelerdeki dijital veri olan pikosaniyelik manyetik akı kuantası $V(t)$ gerilimi olarak şu şekilde ifade edilebilir.

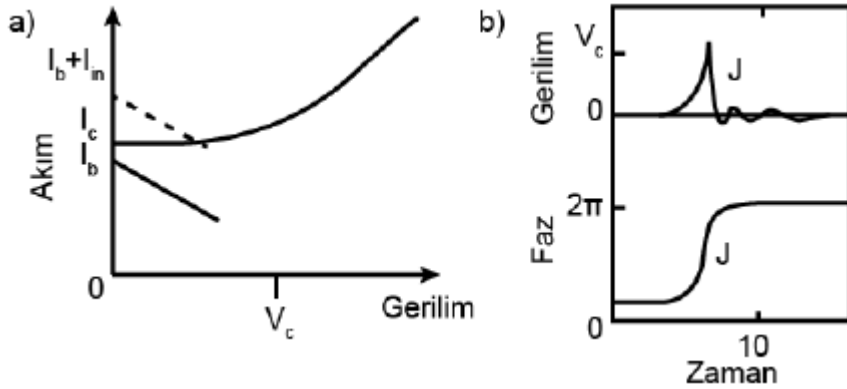
$$\int V(t)dt \equiv \Phi_0 \cong \frac{\hbar}{2e} \approx 2.07 \text{ mV} \times \text{ps}$$

Bu formülde de görüldüğü gibi bir manyetik akı kuantası taşıyan sabit alanlı voltaj darbelerinde (τ), darbe süresinin değişmesi ile gerilim değeri (V_{pk}) de değişir ancak Şekil 1.5'teki SFQ darbesinin şekil olarak yapısı değişse bile darbe sinyalinin enerjisi sabit olduğu için altında kalan alan değişmez.



Şekil 1.5 SFQ Darbesi [29]

Josephson Eklemleri taşıyabilecekleri kritik akım değerinin altında bir besleme akımı (I_b) ile beslenir. Bu tezde bu besleme akımı üretilip kullanılacaktır. Besleme akımının kritik değere yakın olması ile daha hızlı anahtarlama elde edilirken, daha düşük olması ile gürültü sinyali nedeniyle anahtarlama olması engellenir. Anahtarlama yapan bir Josephson ekleminin akım–gerilim ve gerilim–faz ilişkileri Şekil 1.6’da gösterilmiştir.



Şekil 1.6 Josephson ekleminin anahtarlama durumunda a) akım-gerilim ve b) faz gerilim grafikleri [2]

Süperiletkenler için anahtarlama sırasında oluşan gerilimin değeri $300 \mu\text{V}$ ile 1 mV arasındadır. Bu da bize bir Josephson ekleminin sürekli açık kalması durumunda bile çok az enerji harcamasını açıklar. Bu durumda harcanan enerji mikrowatt’tan daha azdır. Josephson eklemlerindeki tüketim de yalnızca anahtarlama sırasında harcanır, durağan halde enerji tüketimi sıfırdır. Durağan halde yalnızca SFQ devreler üzerindeki

kutuplama dirençleri üzerinde enerji tüketimi mevcuttur. Durağan halde tüketilen enerjinin azaltılması için de çalışmalar yapılmaktadır [30]–[32].

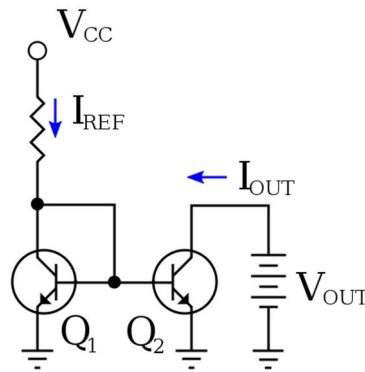
1.4 Akım Kaynağı Devreleri

Akım kaynağı, gördüğü voltajdan bağımsız olarak, elektrik akımını verebilen veya çekebilen elektrik devreleridir. Devre teorisinde, gördüğü voltajdan bağımsız olarak akım çıkışı veren devre elemanlarına da ideal akım kaynağı denir. Bir ideal akım kaynağının iç direnci sonsuzdur. Yani bağımsız bir akım kaynağı sıfır akım verdiğinde açık devre olarak kabul edilir.

Devre tiplerine göre çok yaygın olarak kullanılan belli başlı bazı akım kaynağı devre modelleri vardır. Burada geliştirilen akım kaynağına da başlamadan önce bu devreler de değerlendirilmiştir. Bu bölümde en yaygın olarak kullanılan akım kaynağı devre modelleri incelenmiştir.

1.4.1 Akım Aynası

İki transistörle kurulan bir devredir. Şekil 1.7’de 2 adet npn BJT ile kurulan akım aynası akım kaynağı devresi görülmektedir.



Şekil 1.7 npn BJT transistörlerle kurulan akım aynası devresi [33]

Burada Q1 diyot olarak bağlanmıştır ve Q2 transistörünün baz gerilimini sağlamaktadır. Bu devrede çıkış seviyesini I_{REF} akımı çıkış akımı seviyesinde şu şekilde ifade edilir;

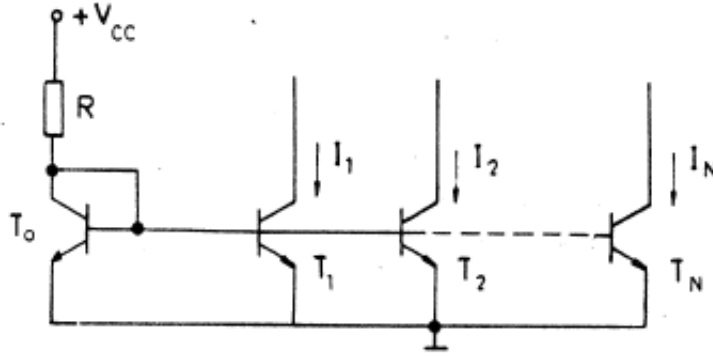
$$I_{ref} = I_C \left(1 + \frac{2}{\beta_0} \right)$$

Bu denklemde I_C çıkış akımını (I_{OUT}) ifade etmektedir. $\beta_0 \gg 1$ olduğu durumda,

$$I_{REF} = I_{OUT}$$

olmaktadır. Buradan hareketle, I_{REF} akımının yansımalarının I_{OUT} 'da aynen görülmesi nedeniyle bu devreye akım aynası devresi denir.

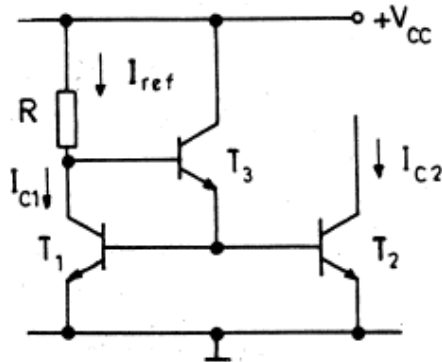
Akım aynasını çoğaltarak tek bir I_{REF} akımından sürmek de mümkündür (Şekil 1.8).



Şekil 1.8 Çok çıkışlı akım aynası [34]

1.4.2 Baz Akımı Kompanzasyonlu Akım Aynası

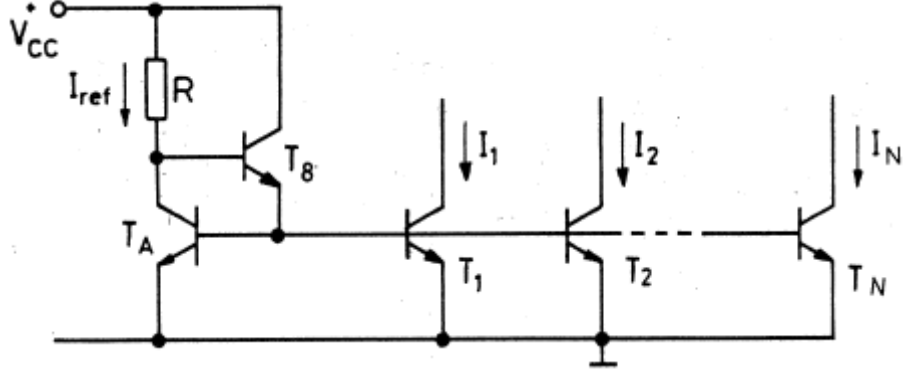
Temel akım aynası devresindeki en önemli hata kaynağı, baz akımlarının doğrudan doğruya I_{REF} akımından çekilmesidir. Bu hata devreye fazladan bir transistör ekleyerek giderilebilir. Bu düzeltilmiş durum Şekil 1.9'da gösterilmiştir.



Şekil 1.9 Baz akımı kompanzasyonlu akım aynası [34]

Yapılan düzeltme ile T_1 ve T_2 'nin baz akımları T_3 transistörü üzerinden çekilmektedir. I_{REF} akımından ise sadece T_3 'ün baz akımı çekilmektedir.

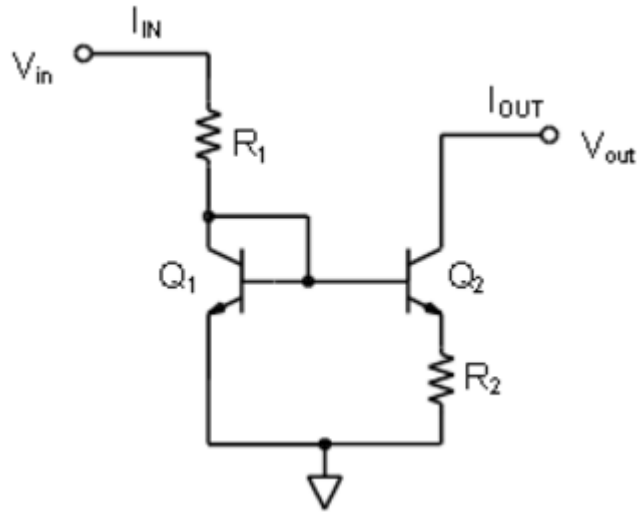
Bu devre genellikle tek bir referans akımından çok çıkışlı akım kaynağı oluşturmakta kullanılır (Şekil 1.10).



Şekil 1.10 Çok çıkışlı baz akımı kompanzasyonlu akım aynası [34]

1.4.3 Widlar Akım Kaynağı

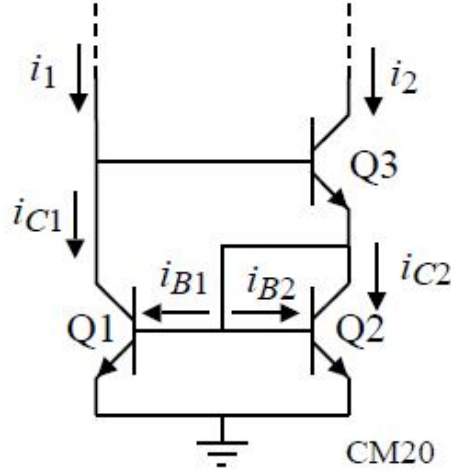
Çok küçük akımlara ihtiyaç olduğunda kullanılır. Normal akım aynasına fazladan bir direnç ekleyerek gerçekleştirilir (Şekil 1.11). Emetör alanlarının oranlandırılmasıyla I_{OUT} akımı ayarlanabilir.



Şekil 1.11 Widlar akım kaynağı [35]

1.4.4 Wilson Akım Kaynağı

Wilson akım kaynağı şekil 1.12’te gösterilmiştir. Çıkış direncinin yükseltilmesi için Wilson akım kaynağında geribesleme kullanılmıştır. Geribesleme Q2 transistörü üzerinden sağlanmaktadır. Wilson akım kaynağı bu haliyle çıkış direncini artırır ve baz akımlarının önemini de azaltmaktadır. Şekilde referans akım olan i_1 ile i_2 arasındaki fark, Q3 transistörünün bazından akmaktadır. Bu da diyot olarak bağlanmış Q2 transistöründen akar. Q2 transistörü Q1 transistöründen aynı akımın akmasını sağlar. Böylece i_2 akımını düzelter ve i_1 akımına yaklaşık olarak eşit olmasını sağlayan bir geribesleme çevrimi oluşur.

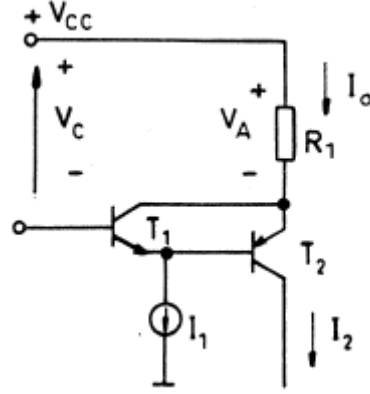


Şekil 1.12 Wilson akım kaynağı [36]

Wilson akım kaynağındaki en büyük sorun, baz akımı etkisini yok etmeden, aynı referans akımından çok sayıda çıkış elde edilmesinin kolayca mümkün olmamasıdır.

1.4.5 Gerilim Kontrollü Akım Kaynağı

Şimdiye kadar olan akım kaynaklarında, I_{REF} referans akımını belirleyen R direncine uygulanan gerilimin değiştirirsek, gerilim kontrollü akım kaynağı geliştirebiliriz. Şekil 1.13’te buna örnek olabilecek bir voltaj kontrollü akım kaynağı devresi görülmektedir.



Şekil 1.13 Voltaj kontrollü akım kaynağı [34]

Bu tezde geliştirilen akım kaynağı gerilim kontrollü bir akım kaynağı uygulamasıdır (Şekil 4.2). Bu şekilde de görüldüğü üzere, birinci işlemsel yükselteç devrenin ön kısmından gelen referans voltajı ikinci işlemsel yükselteç için ölçeklendirir ve ikinci işlemsel yükselteçte de çıkıştan alınan bir geribeslemeyle komparatöre sokulur. Bu geribesleme sayesinde gelen referans voltaja göre çıkış akımı değiştirilir.

Bilgisayar kontrollü bir akım kaynağı geliştirmek için akım çıkışını uzaktan kontrol edebilmek gereklidir. İncelediğimiz en çok kullanılan akım kaynağı devrelerinde akım çıkışları genel olarak direnç ile kontrol edilmektedir. Ancak son incelediğimiz akım kaynağında bu kontrol voltaj ile yapılmaktadır. Uzaktan kontrol bu ikisinden biriyle yapılabilir. Direnç ile akım çıkışını kontrol etmek istediğimizde, kontrol direnci için dijital bir potansiyometre kullanmak gerekmektedir. Ancak piyasadan alınabilecek potansiyometrelerin hiçbiri burada geliştirilen akım kaynağı kadar hassas, düşük ve düşük adımlarla bir akım çıkışını kontrol edebilmemizi sağlamayacaktır. Çünkü markette satılan dijital potansiyometreler 10 biti geçmemektedir. Ancak hassas bir akım kontrolü için ve daha düşük akımlara inebilmek için yüksek bitli (18 ve üstü) DAC ile kontrol etmek daha uygun olmaktadır. Bu yüzden bu çalışmada gerilim kontrollü bir akım çıkışı verebilen bir devre üzerinde çalışılmıştır.

2. SÜPERİLETKEN DEVRELER İÇİN AKIM KAYNAĞI TASARIMI

SFQ devrelerinin çekirdeğini Josephson eklemi oluşturmaktadır. Josephson eklemeleri taşıyabilecekleri kritik akım değerlerinin altında bir besleme akımı (I_b) ile beslenmesi gerekmektedir. Dolayısıyla SFQ devrelerin bu I_b akım beslemesini sağlayacak bir akım kaynağına ihtiyaç vardır.

SFQ mantık devreleri, Josephson eklemelerini anahtarlama aracı olarak kullanarak, daha az güç tüketerek daha hızlı anahtarlama gerçekleştirirler. Akım kaynağı normal oda şartlarında, SFQ devreler kriyojenik sıcaklıklarda çalışırken, karmaşık devrelerde, termal yüke (I^2R) bağlı olan Joule ısınma etkisinin düşük olması için, birçok noktadan akım beslemesi gerekmektedir. Ayrıca burada besleme akımının kritik değere yakın ve uzak olması da sistemin hızını, gürültü oranını ve hatta doğru çalışıp çalışmamasını etkiler. Akım beslemesi için istenen seviye, kapı seviyesindeki birimler için μA mertebesinde, mikroişlemci gibi karmaşık devreler içinse birkaç Amper mertebesinde olabilir.

Bu nedenlerden dolayı besleme akımlarını sağlayacak olan akım kaynağı, düşük gürültülü, çok kanallı, değişen koşullarda kararlı çıkış verebilen ve nA mertebelerinden mA mertebelerine kadar çıkış verebilen bir kaynak olmalıdır. Bu akım kaynağının çıkışı kolaylıkla kullanıcı tarafından değiştirilebilmeli, bu yüzden bilgisayar kontrollü olmalıdır. Piyasadan satın alınabilecek bu tarz ürünler tek kanallıdır ve maliyetleri çok yüksektir. Tek kanallı olduklarından dolayı, bu ürünlerden fazla sayıda alınması gerekmektedir. Bu da maliyeti çok yükseltmektedir. Ayrıca çok sayıdaki akım kaynağının kaplayacağı hacim çok büyük olacağından yer sıkıntısı da sorun oluşturmaktadır.

Akım kaynağı birçok farklı gereksinim için tasarlanabilir [37]. Süperiletken devreler için gerekli akım kaynağının tasarımını ve performansını etkileyen parametreleri aşağıdaki gibi listeleyebiliriz.

- Akım kaynağı için gerekli güç beslemesi
- Çıkış akım aralığı
- Çıkış voltaj aralığı
- Çıkış akımı kararlılığı
- Bilgisayar arayüzünden akım çıkışı programlanabilirliği

- Gürültü
- Kısa süreli tepki (Transient response)
- Fiziksel boyutlar

2.1 Akım Kaynağı İçin Gerekli Güç Beslemesi

Akım kaynağı için gerekli güç beslemesi, kullanılacak devre tasarım ve elemanlarının ihtiyaçları dikkate alınarak karar verilmelidir. Örneğin tasarlanan akım kaynağı 12V-18V voltaj aralığında ve en fazla 500mA akım gereksinimleriyle tanımlanabilir. Ayrıca beslemenin gürültüsü, akım kaynağının çıkış gürültüsüyle bağlantılıdır. Bu nedenle besleme, AC/DC dönüştürücü ya da masaüstü güç kaynakları ile değil de batarya ile olursa, akım çıkışının üstünde olan gürültü miktarı azalacaktır. AC/DC dönüştürücü ya da masaüstü güç kaynakları kullanıldığında, çıkış akımı üzerinde, şebeke frekansı olan 50 Hz'te, bir gürültü varlığı net bir şekilde gözlenmektedir. Bu gürültü, çıkış akımı düşükçe kendini daha da belirgin hale getirmektedir.

2.2 Çıkış Akımı Aralığı

Süperiletken devreler için besleme akım gerekliliği süperiletken devrelerdeki Josephson eklemelerinin sayısına bağlı olarak değişiklik göstermektedir. Tek bir eklem için μA 'ler seviyesinde olan bu besleme akımı gereksinimi, kapı seviyesindeki devreler için mA'ler mertebesinde, binlerce eklem içeren karmaşık devrelerde ise birkaç A mertebesinde olmaktadır. Bu tez kapsamında, Josephson eklemi ve kapı seviyesindeki devrelerin testlerine yönelik tasarımlarda yoğunlaşmış olup, 100 nA-1 mA mertebesinde, 10 nA hassaslığında bilgisayar ile kontrole izin veren bir devre geliştirilmiştir. Laboratuvar ortamında yapılacak testler ve denemeler için kullanılacak olan akım kaynağının, çok kanallı ve her bir kanalından 100nA - 1 mA aralığında ayarlanabilir akım çıkışı verebilecek şekilde olması, süperiletken devre tasarımı çalışmalarında faydalı olacaktır.

2.3 Çıkış Voltaj Aralığı

Süperiletken devreler için tasarlanacak olan akım kaynağının çıkış voltaj aralığı beslenecek devrelerin çıkış empedansı dikkate alınarak verilmelidir. Aslında bu değer akım kaynakları için çıkış empedansı değeri olarak da verilmektedir [15], [38]. Süperiletken devrelerde, giriş empedansı yaklaşık 5 Ω 'dan küçük olmaktadır. Ancak devrelerdeki kablolar, bakır hatlar, konektörler gibi ara elemanlar nedeniyle toplam giriş empedansının 100 Ω 'dan düşük olduğu kabulü ile tasarım yapılmıştır. Dolayısı ile devrenin giriş empedansı 100 Ω iken 1 mA akım değeri kaynaktan çıkışa verilirse, kaynağın çıkışında 0,1 V potansiyeli desteklemesi gereklidir.

2.4 Çıkış Akımı Kararlılığı

Süperiletken devrelerin akım beslemesindeki en büyük problemlerden biri de, süperiletken devreler kontrollü kriyojenik şartlarda çalışırken, dışarıda bunları besleyen akım kaynaklarının değişen oda şartlarında çalışmasıdır. Bu değişen oda şartlarında sıcaklığa, manyetik alana ve bunun gibi bazı başka dış etmenlerin etkilediği ortam koşullarına bağlı olarak, akım kaynakları her zaman aynı çıkışı vermeyebilir. Çıkan akım, zamanla bir miktar aşağı ya da yukarı yöne doğru sapabilir, ya da bu değişen koşullara bağlı olarak belli bir değerde aşağı ya da yukarı yönde ötelenebilir. Bu yüzden tasarım sırasında, akım kaynağının elektronik donanımında, ortama bağlı değişimi olabildiğince az olan komponentler kullanılmalıdır. Bütün donanımı bu malzemelerle tamamlamak maliyeti bir hayli yükseltebileceğinden, gerekli yerlerde bu malzemeleri kullanmak maliyet açısından tercih edilmiştir. Ayrıca kart üstünde, baskılı devre kartı tasarımında ve kutu tasarımında da bu konulara dikkat edilmeli ve gerekli önlemler alınmalıdır. Bu sayede tasarlanan akım kaynağının çıkış kararlılığı ve tekrarlanabilirliği olabildiğince iyileştirilmiş olacaktır.

2.5 Bilgisayar Arayüzünden Akım Çıkışı Programlanabilirliği

Akım kaynaklarında çıkışı kontrol edebilmek için kullanıcıya sunulan bir arayüzün olması gerekir. Nihayi bir ürün haline gelmiş bir akım kaynağında, baskılı devre

üzerinde malzeme deęiřtirerek ya da herhangi bir malzemeye müdahale ederek akım çıkıřını deęiřtirmek çok uygulanabilir ve kullanıřlı olmamaktadır. Bu tasarımda kullanılacak olan arayüz, bilgisayar üstünde kontrol edilebilir bir řekilde geliştirilmiřtir. Arayüz bu haliyle çok kanallı bir akım kaynaęı için daha kullanıřlı olmaktadır. Bu řekilde birçok kanal tek bir kullanıcı arayüzü ile kontrol edilebilmektedir.

2.6 Gürültü

Akım kaynaęından süperiletken devreye alınan akım, SFQ devreler için besleme akımı olarak kullanılmaktadır. Burada sistemin hızı, devreye besleme akımı olarak verilen akımın gürültü oranı ile doğrudan baęlantılıdır. Besleme akımının genlięinin kritik deęere yakın olması ile daha hızlı anahtarlama edilmektedir [2]. Bu genlięin daha düşük olmasıyla da, beslemenin üzerinde bulunan gürültü sinyali nedeniyle anahtarlama olması engellenir. Devreye verilen besleme akımı yüksek ama düşük gürültülü olursa, gürültüler anahtarlama oluřturmaz ve sistemin daha hızlı çalıřması saęlanır. SFQ devreler DC akım ile besleneceęinden, burada gürültü analizleri DC ve DC'ye yakın frekanstaki gürültüler için yapılmalıdır. Yani süperiletken devreler için kullanılacak akım kaynaklarında söz konusu olan ve dikkat edilmesi gereken gürültü bilgileri, 0 Hz – 10 Hz arasındaki gürültü bilgileridir. Burada geliştirilen akım kaynaęı için yapılan testlerde, 100 nA akım çıkıřı için 1.1 mHz ile 10 Hz arasındaki toplam gürültü 5.3 nA/ $\sqrt{\text{Hz}}$, 1 mA akım çıkıřı için 1.5 mHz ile 10 Hz arasındaki toplam gürültü 13.9 nA/ $\sqrt{\text{Hz}}$ olarak ölçülmüřtür.

2.7 Kısa Süreli Tepki (Transient Response)

Süperiletken devreler için kullanılacak akım kaynaęının kısa süreli tepkisi hızlı olmalıdır. Devreye baęlanan akım kaynaęının çıkıřı, kullanıcının istedięi akım çıkıřı deęerini girdięi anda, o deęere gelmelidir. Çıkıřtaki akımın istenen akımdan fazla veya az olup, zamanla istenen seviyeye gelmesi süperiletken devrelerde istenen bir

durum deęildir. Bunun gerekleřtięi durumlarda devrenin hasar gorme ihtimali vardır.

2.8 Fiziksel Boyutlar

Geliřtirilen akım kaynaęının boyutları, laboratuvar ortamında kullanıma uygun olmalıdır. Piyasadan alınabilecek akım kaynakları her ne kadar normal boyutlarda olsa da, tek kanallı olduklarından dolayı, fazla sayıda kullanılacaęından, kapladıkları hacim laboratuvar ortamında kullanılmaya uygun olmamaktadır. Burada geliřtirilen akım kaynaęı, bir PCB ve üzerindeki devre elemanlarından oluřmaktadır. 6 kanallı ve 174 x 138 mm boyutundadır. Yukseklieęi de 15.05 mm'dir. Bu bir kutu ierisinde veya var olan bir test sistemine monte edilerek laboratuvar ortamında kullanıma uygundur.

3. ELEKTRONİK DEVRELERDE GÜRÜLTÜ VE HESAPLAMALARI

Elektronik devrelerde en çok bilinen dört gürültü tipi vardır. Bunlar;

- Saçma gürültüsü (shot noise)
- Termal gürültü (thermal noise)
- Titreşim gürültüsü (Flicker noise, 1/f noise)
- Patlak gürültü (Burst noise)

3.1 Saçma Gürültüsü (Shot Noise)

Bir potansiyel farktan akan akım ile oluşur [39]. Yarıiletkenlerde akımın ortalama değerinden rastgele dalgalanmasıyla oluşur. Yarıiletkenlerde deliklerin ve elektronların p-n jonksiyonlar üzerinden rastgele difüzyonları ile ve delik-elektron çiftlerinin rastgele üretim ve birleşimleri ile üretilir. Δf frekans bandındaki rms saçma akım gürültüsü şu formülle hesaplanabilir;

$$I_{sh} = \sqrt{2qI\Delta f}$$

Burada q elektronik yükü ve I da cihaz üstünden akan DC akımı temsil etmektedir.

3.2 Termal Gürültü (Thermal Noise)

Termal gürültü, termal enerjinin, serbest elektronların rezistif materyal içinde rasgele hareket etmelerini sağlamasıyla oluşur [40]. Johnson gürültüsü olarak da anılır. Açık devre rms termal gürültüsü şu formül ile hesaplanabilir;

$$I_t = \sqrt{\frac{4kT\Delta f}{R}}$$

Burada k Boltzmann sabiti, T mutlak sıcaklık, R direnç ve Δf de sözkonusu olan frekans bandıdır.

3.3 Titreşim Gürültüsü (Flicker Noise, 1/f Noise)

İki iletken arasındaki düzgün olmayan bağlantı, dalgalanan iletkenliğe neden olur. Bu da üzerinden geçen DC akımın dalgalanmasına yol açar. Bu durumda oluşan gürültüye titreşim gürültüsü denir [41]. Bütün iki iletkenin birleştiği durumlarda bu gürültü oluşur. Metal film dirençlerde karbon film dirençlere göre daha az oluşur. Δf frekans bandındaki rms titreşim gürültüsü şu formülle ifade edilir;

$$I_f = \sqrt{\frac{K_f I^m \Delta f}{f^n}}$$

Burada I DC akımını, $n \approx 1$, K_f titreşim gürültüsü katsayısını ve m de titreşim gürültüsü kuvvetini ifade etmektedir. BJT'lerdeki baz akımı modellerinde m değeri $1 < m < 3$ aralığında olmaktadır [42]. Analizleri daha da basitleştirmek için hesaplamalarda genelde $n=m=1$ olarak kullanılır [43].

3.4 Patlak Gürültü (Burst Noise)

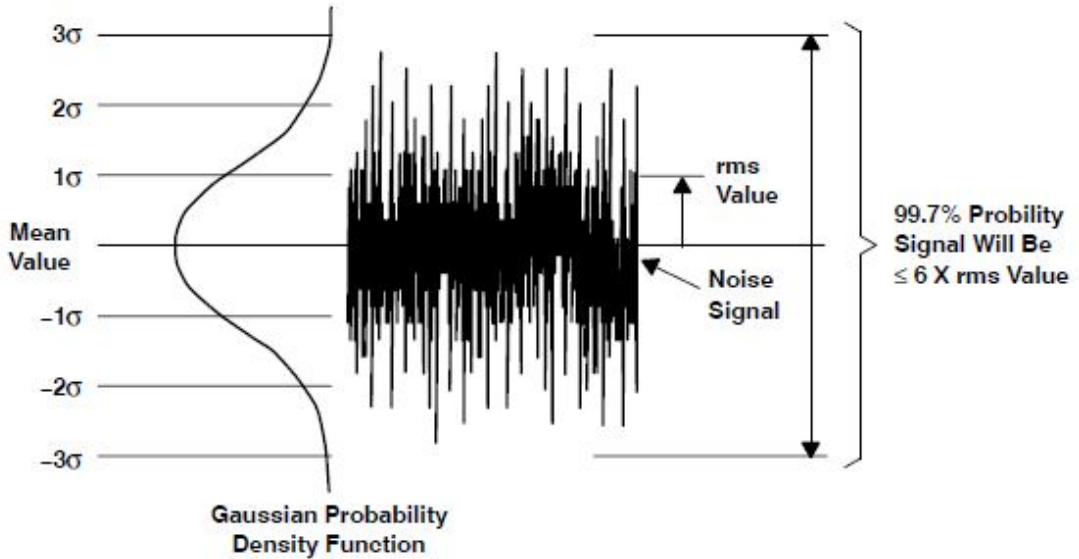
Patlak gürültü bir p-n eklemesindeki metalin saflığının bozukluğundan meydana gelir [44]. Genellikle üretim hatasından kaynaklanır, üretim prosesinin iyileştirilmesiyle bu gürültüden büyük oranda kurtulunabilir. Tipik olarak termal gürültünün 2 ile 100 katı arasında değişir [44].

3.5 Gürültü Karakteristiği

Gürültü kaynakları genlikleri zamana bağlı olarak değişkenlik göstermektedirler ve bunlar sadece olasılık yoğunluğu fonksiyonu (probability density function) ile açıklanabilirler. Termal gürültü ve saçma gürültüsü (shot noise) Gaussian olasılık yoğunluğu fonksiyonu ile açıklanabilirler. Eğer Gaussian dağılımının standart sapmasına δ dersek, gürültünün anlık değeri bu $\pm\delta$ 'lık (%68) ortalama değerler

arasında yer alacaktır. Tanıma göre, δ^2 (varyans), ortalama değere göre, karelerin ortalamasının değişimidir. Bu da demektir ki, genlikleri Gaussian dağılımına sahip gürültü sinyallerinde, ortalamaya göre karelerin ortalamasının değişimi, akımın veya enerjinin karelerinin ortalaması, varyansa (δ^2) eşittir ve bunların rms değerleri de standart sapmaya (δ) denk gelir.

Teorik olarak gürültü genlikleri sonsuza yaklaşabilir. Ancak, genlik arttıkça bunun olma ihtimali ciddi şekilde düşer. $\pm 3\delta$ limitleri gürültü sinyalinin olabilecek aralığının %99,7'sini kapsar. Şekil 3.1'de gürültü genliklerinin olabilme ihtimali olan bölgeler ile rms değerleri aynı grafikte gösterilmiştir [45].

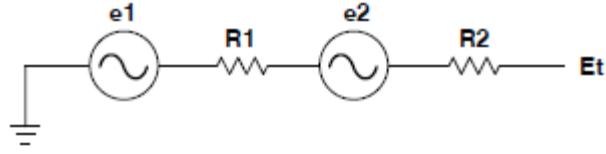


Şekil 3.1 Gürültü genliğinin Gaussian dağılımı [45]

3.6 Gürültü Kaynaklarının Toplanması

Bir devrede farklı gürültü kaynakları vardır ve ve birleşerek gürültünün tamamını oluşturular.

Örneğin birbirine seri olarak bağlanmış iki direnci düşünelim ve bu dirençler R1 ve R2 olsun. Bu iki direncin de gürültü üreteçleri vardır ve bu üreteçler dirençlerin kendisiyle iç içedir. Bu durum Şekil 3.2'de gösterilmiştir.



Şekil 3.2 R1 ve R2 dirençleri gürültü modelleri

Burada e_1 ve e_2 'yi aşağıdaki gibi ifade edebiliriz.

$$\overline{e_1^2} = \int 4kTR_1 df$$

$$\overline{e_2^2} = \int 4kTR_2 df$$

İki direncin voltaj karelerin ortalamasının hesaplanması için anlık değerlere,

$$E_t(t) = e_1(t) + e_2(t)$$

diyelim. Bu durumda iki direncin voltaj karelerinin ortalaması;

$$\overline{E_t(t)^2} = \overline{[e_1(t) + e_2(t)]^2} = \overline{e_1(t)^2} + \overline{e_2(t)^2} + \overline{2e_1(t)e_2(t)}$$

olur. Burada $e_1(t)$ ve $e_2(t)$, tamamen bağımsız iki dirençten gelmektedir. Bu yüzden bunların çarpımı sıfır olacaktır.

$$\overline{2e_1(t)e_2(t)} = 0$$

Sonuç olarak;

$$\overline{E_t^2} = \overline{e_1^2} + \overline{e_2^2}$$

olacaktır. Bu nedenle, gürültü kaynakları tamamen birbirinden bağımsız olduğu sürece, bağımsız gürültü kaynaklarının ortalama karelerinin değeri toplamı, ortalama karelerinin ayrı ayrı toplamına eşittir.

3.7 Gürültü Spektrumu

Düzgün bir sinüs dalgasında güç sadece bir frekansta bulunur. Gürültü gücü ise bütün frekans spektrumuna yayılmış durumdadır. Voltaj gürültü güç yoğunluğu ($\overline{e^2}/Hz$) ve akım gürültü güç yoğunluğu ($\overline{i^2}/Hz$) gürültü hesaplamalarında sıklıkla kullanılmaktadır. Ortalama kare değerini hesaplamak için, güç yoğunluğu, söz konusu olan frekans aralığında toplanır.

3.8 İstenen Frekansta Toplam Gürültü (Integrated Noise) Hesaplamaları

Verilen bir frekans bandındaki gürültüyü ifade etmek için başlangıç ve bitiş frekansları, hesaplanacak toplam gürültü için frekans limitleri olarak kullanılmaktadır. Gürültü toplama işlemi aşağıdaki hesaplamalarla yapılmaktadır. Bu hesaplamalarda voltaj ele alınmıştır ancak akım da olsa yine birebir aynı şekilde yapılacaktır.

Eğer elimizde beyaz veya sabit voltaja karşılık frekans varsa;

$$\overline{e^2} = \int_{f_L}^{f_H} C df = C(f_H - f_L)$$

yazılabilir. Burada $\overline{e^2}$ voltaj ortalama karesi, C Hertz başına düşen spektral güç yoğunluğu, f_L en düşük frekans, f_H en yüksek frekanstır.

Buna göre 1/f gürültüsü karşılık frekans şu şekilde hesaplanabilir;

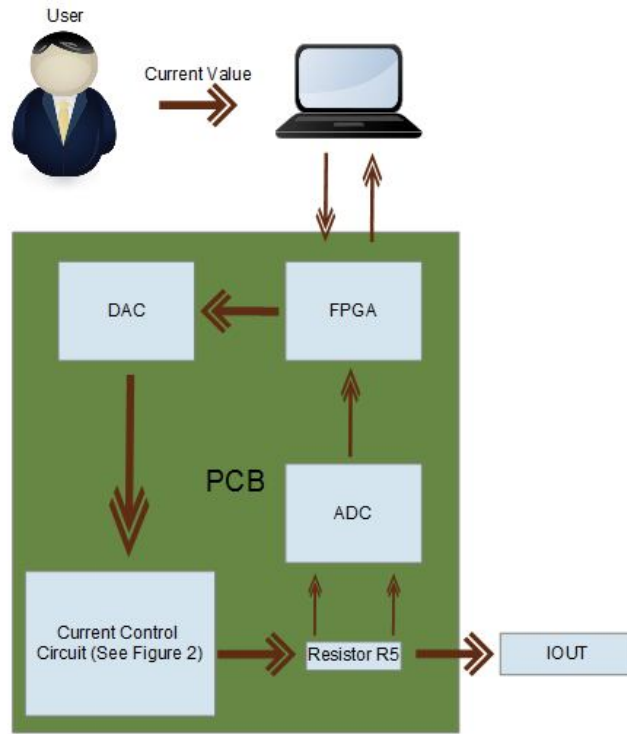
$$\overline{e^2} = \int_{f_L}^{f_H} \frac{K^2}{f} df = K^2 \ln \frac{f_H}{f_L}$$

Burada K cihazın volt cinsinden sabit katsayısıdır.

4. TASARIM

4.1 Sisteme Genel Bakış

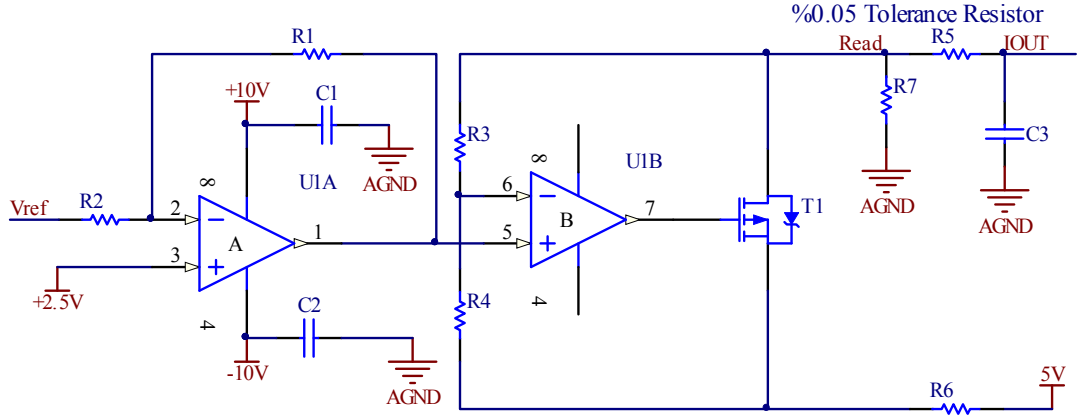
Bu çalışmada süperiletken devreler için geliştirilen akım kaynağının tasarımı iki ana parçadan oluşmaktadır. Bunlardan ilki akım kaynağı donanımının tasarlanması ve uygulanmasıdır. İkinci parçası da akım kaynağı yazılımının tasarlanması ve uygulanmasıdır. Yazılım kısmı da iki parçadan oluşmaktadır. Bunlardan ilki FPGA geri besleme algoritmasının ve HDL kodların tasarlanması ve uygulanmasıdır. Diğeri de akım kaynağının akım çıkışının kontrol edilebilmesi amacıyla geliştirilmiş olan kullanıcı arayüzüdür.



Şekil 4.1 Sistem Akış Diyagramı

Sistemin akış diyagramı Şekil 4.1'de gösterilmiştir. Şekilde de görüldüğü üzere, öncelikle kullanıcı akım değerini bilgisayardaki geliştirilmiş olan kullanıcı arayüzü üzerinden girer. Bu veriye göre bilgisayar, FPGA (XC3S200AN-4FTG256I [46]) tarafından, akım kaynağı cihazının elektronik kartı üstündeki DAC'a (DAC9881 [47]) yazılacak olan yazmaç değerini hesaplar ve bunu FPGA'ye gönderir. Bunun

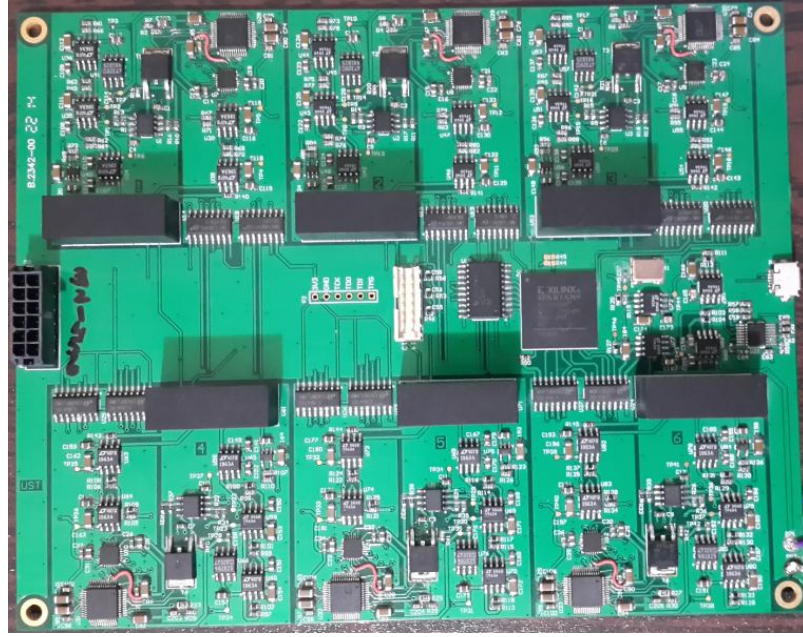
yanında bilgisayar aynı zamanda akım çıkışından geri besleme verilmesini sağlayan ve akım çıkışını okuyan ADC'nin (AD7660 [48]) okuması gereken değerleri de hesaplayarak FPGA'ye gönderir. FPGA devamlı bu hesaplanan değeri R5 direnci üzerinden okuduğu çıkış akımı verisiyle karşılaştırır ve geri besleme olarak gönderir. Bu karşılaştırma verisine bağlı olarak, FPGA çıkış akımını yükseltir, azaltır ya da aynı seviyede bırakır. R5 direnci ve akım devresi Şekil 4.2'de verilmiştir. Bu işlemlerin hepsi akım kaynağında bulunan 6 kanallı hepsi için ayrı ayrı ve birbirlerinden bağımsız olarak yapılır.



Şekil 4.2 Akım kontrol devresi şeması

4.2 Akım Kaynağı Donanımı

Akım kaynağı donanımı, 6 katlı, 174x138 mm boyutlarında bir elektronik baskı devre kartı ve üzerindeki elektronik devre elemanlarından oluşmaktadır (Şekil 4.3). Baskılı devre kartı üzerinde 6 adet akım kaynağı ve bunlar için gerekli ortak devreler bulunmaktadır. Her bir akım kaynağı kanalı 5 kısımdan oluşmaktadır. Bunlar, güç besleme devreleri, DAC devresi, akım kontrol devresi, ADC devresi ve izolatör devrelerinden ibarettir. Ortak devreler 4 kısımdan oluşmaktadır. Bunlar, ortak kısım güç besleme devreleri, FPGA devreleri, osilatör devresi ve seri port dönüşüm devresinden ibarettir.

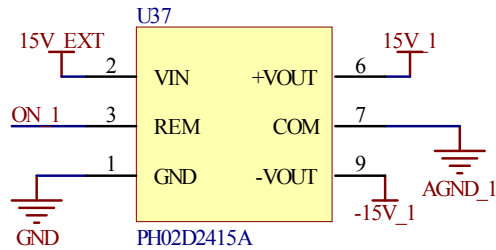


Şekil 4.3 Akım kaynağı donanımı

4.2.1 Akım Kaynağı Kanalları Donanımı

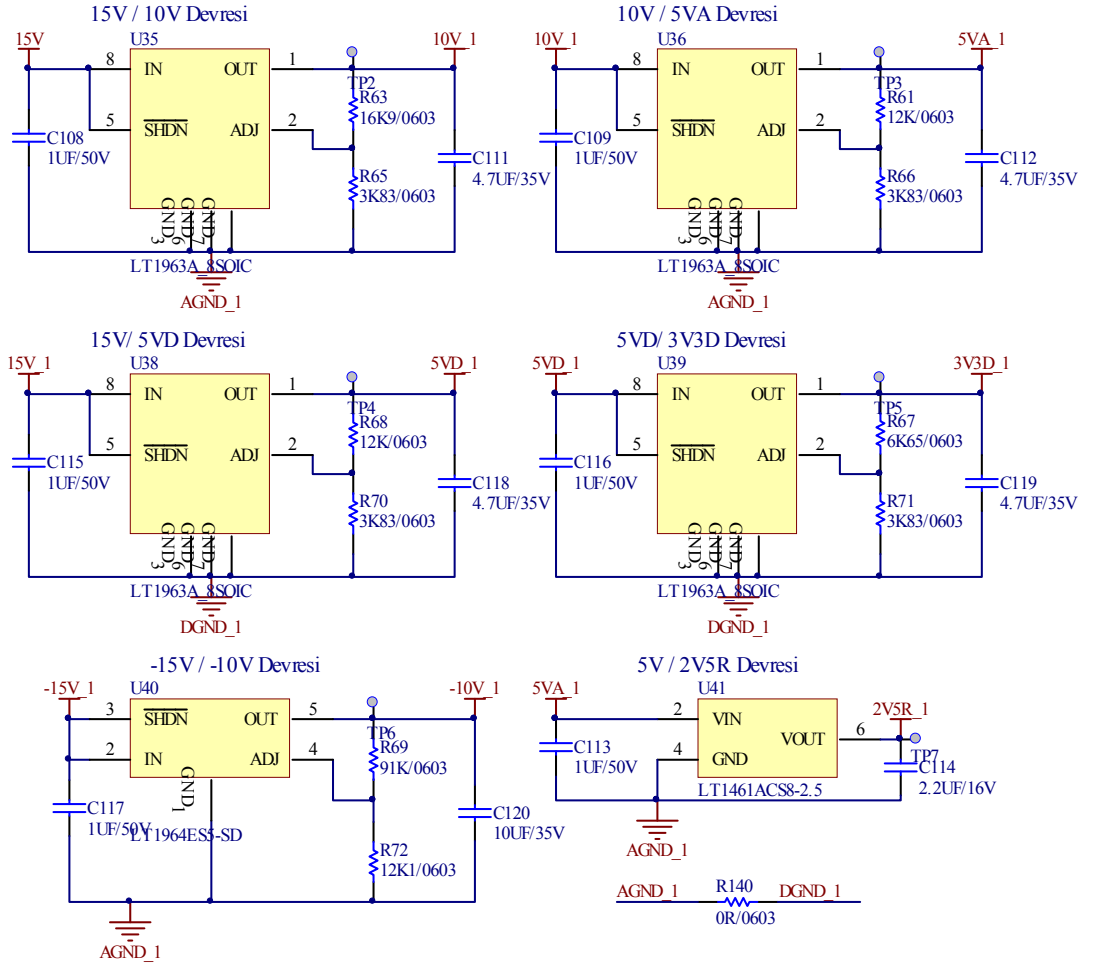
4.2.1.1 Güç Besleme Devreleri

Bu devrelerde her bir kanalın kendi içinde gereken güçler üretilmektedir. Geliştirdiğimiz 6 kanallı akım kaynağında, her bir kanal birbirinden tamamen bağımsız ve izole olduğundan dolayı, her kanal ayrı ayrı besleme kullanmak zorundadır. Ancak karta 12V – 18V arasında yalnızca bir adet besleme gelmektedir. Bu nedenle bu beslemeden her kanal için ayrı izole beslemeler üretilmelidir. Bunun için her kanalın ana besleme girişinde izole bir DC-DC çevirici (PH02D2415A [49]) kullanılmıştır. Bu çevirici kanalın içi ile dışını birbirinden yalıtır (1500 VDC izolasyon voltajı) ve 9V – 18V aralığındaki voltajı $\pm 15V$ 'a çevirerek içeri verir (Şekil 4.4).



Şekil 4.4 Besleme izolasyon devresi

Kanalı dış ortamdan yalıtıttıktan sonra iç ortamda artık istenen voltajlar üretilebilir. Kanalın içinde elektronik devrelerin çalışabilmesi için sırasıyla $\pm 10V$, analog 5V, sayısal 5V, sayısal 3.3V ve referans 2.5V beslemeleri gerekmektedir. Kanal 1 için oluşturulan bu devreler Şekil 4.5'te gösterilmektedir. Her kanal, birbirleri arasında izole olduklarından, bu devreleri bulundurmaktadır.



Şekil 4.5 Birinci kanal için güç besleme devreleri

4.2.1.2 DAC Devresi

Gürültü problemlerinden dolayı DAC seçiminde bazı kriterler göz önünde bulundurulmuştur. Bunlardan ilki, entegrenin analog ve sayısal besleme ve topraklama hatlarının ayrı olmasına dikkat edilmiştir. Bu şekilde, elektronik baskı devre kartında da dikkatli bir sayısal ve analog topraklama ile, yüksek frekansta çalışan (50 MHz) sayısal kısmın anahtarlama ve yüksek frekans gürültülerinin,

analog kısmı etkilemesinin önüne büyük ölçüde geçilmiş olunacaktır. Ayrıca DAC'ın 16 bit ve üstü bir DAC olmasına dikkat edilmiş ve DAC9881 [47] seçilmiştir. Bu DAC'ta analog, sayısal besleme ve topraklama hatları ayrı olarak verilmiştir ve bu DAC 18 bitlik bir DAC'tır. Birinci kanalın DAC devresi Şekil 4.6'da gösterilmiştir.

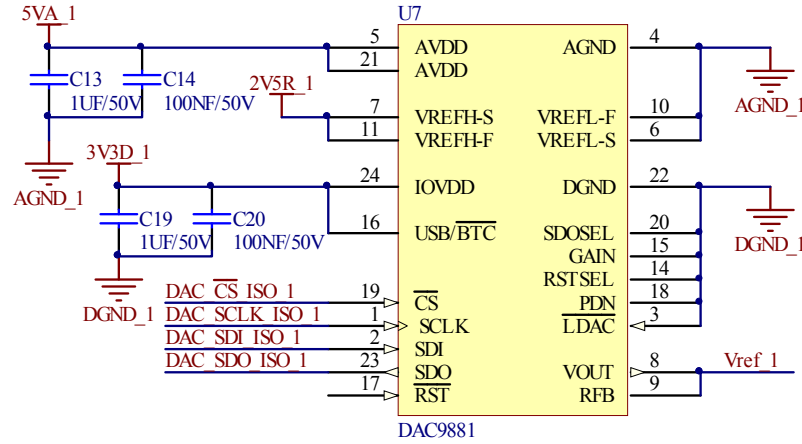
DAC devresinin beslemeleri şu şekildedir; analog voltaj olarak 5V, sayısal voltaj olarak da 3.3V kullanılmaktadır. Bir de referans sinyali için 2.5V kullanılmıştır.

DAC devresinin görevi, FPGA'den gelen veriye göre bir gerilim (Vref) çıkartmaktır. Bu gerilim akım kontrol devresinin referans gerilimi olarak kullanılmaktadır.

Kullanılan DAC 18 bittir ve referans aralık bu sayede 2^{18} (262144) basamağa bölünebilir. DAC devresi referans olarak 2.5V kullanılmaktadır. Yani DAC devresi akım kontrol devresine

$$2.5 \text{ V} / 262144 = 9,5 \mu\text{V}$$

adımlarla referans voltajı verebilir.



Şekil 4.6 Kanal 1 için DAC Devresi [47]

Kanallar arasındaki DAC'lar birbirine halka-zincirleme dizimi (daisy chain) ile bağlıdır. Bu sayede kodlamada, baskılı devre kartı tasarımında ve iletişimde kolaylık sağlanmıştır.

4.2.1.3 Akım Kontrol Devresi

Akım kontrol devresi Şekil 4.2’de gösterilmiştir. Akım kontrol devresinin görevi gelen referans voltaja göre bir akım çıkışı vermektir. Ayrıca bu devrede çıkış akımını okumak için çıkış akımı, hassas (%0.05) ve sıcaklıkla az değişen (5 ppm/°C) bir direnç üstünden geçirilmektedir. Bu direnç üzerinden voltaj, ADC devresi tarafından diferansiyel olarak okunur.

Geliştirilen akım kaynağında 2 farklı akım kontrol devresi kullanılmıştır. Kart üzerindeki kanallarda 3 tanesi 0,1 μA – 10 μA aralığında akım çıkışı verebilirken, 3 tanesi de 10 μA – 1000 μA aralığında akım çıkışı verebilmektedir. Devrelerde herşey aynıdır, yalnızca çıkışın istenen değerlerde olmasını sağlayan bazı pasif devre komponentlerinin değerleri, akım çıkış kabiliyetlerine göre değiştirilmiştir. Bu devre elemanları Şekil 4.2’de görülen R5, R7 ve R6 dirençleridir.

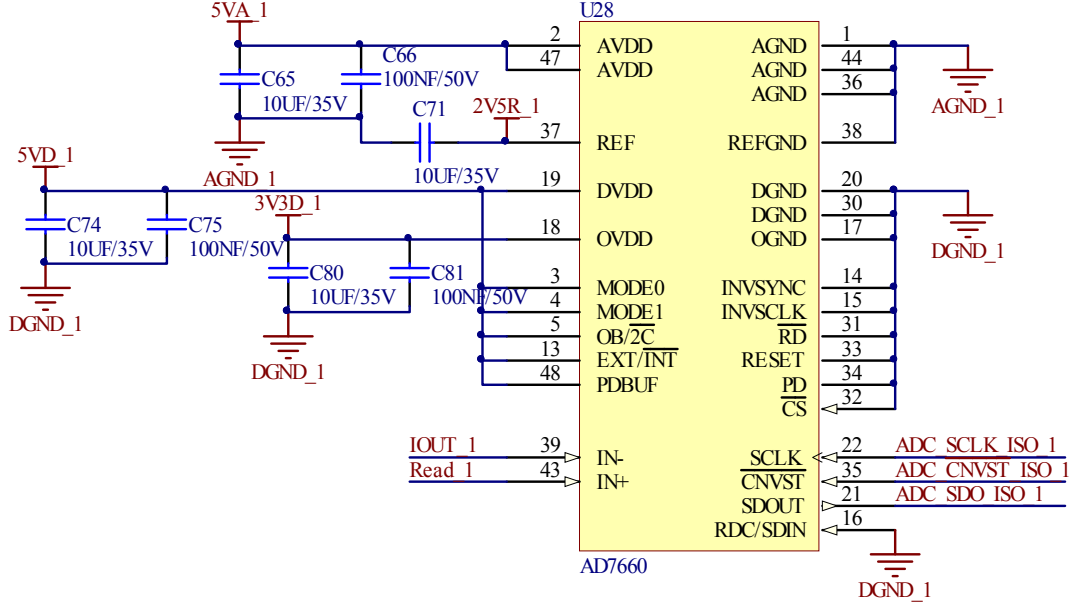
4.2.1.4 ADC Devresi

Gürültü problemlerinden dolayı ADC seçiminde de DAC seçiminde göz önünde bulundurulmuş kriterler burada da söz konusudur. Entegrenin analog ve sayısal besleme ve topraklama hatlarının ayrı olmasına burada da dikkat edilmiştir. Yine seçilen ADC’nin de 16 bit ve üstü bir ADC olmasına dikkat edilmiş ve AD7660 [48] seçilmiştir. Bu ADC’de analog, sayısal besleme ve topraklama hatları ayrı olarak verilmiştir ve bu ADC 16 bitlik bir ADC’dir. Birinci kanalın ADC devresi Şekil 4.7’de gösterilmiştir.

ADC devresinin beslemeleri şu şekildedir; analog voltaj olarak 5V, sayısal voltaj olarak da 5V ve 3.3V kullanılmaktadır. Bir de referans voltaj sinyali için 2.5V kullanılmıştır.

ADC devresinin görevi, akım kontrol devresinde bulunan hassas okuma direnci üzerindeki voltajı diferansiyel olarak okumaktır. Bu direnç çıkışı akımı yolu üzerinde bulunmaktadır ve çıkışa yönlendirilen akım bu direnç üzerinde bir voltaja dönüştükten sonra ADC tarafından okunacaktır. Burada okunan voltaj değeri,

FPGA'ya geri besleme olarak gönderilir. Bu bilgiye göre de FPGA, DAC'a gönderdiği yazmaç değerini artırır ya da azaltır.



Şekil 4.7 Kanal 1 için ADC Devresi [48]

Kullanılan ADC 16 bittir ve referans aralık bu sayede 2^{16} (65536) basamağa bölünebilir. ADC devresi referans olarak 2.5V kullanmaktadır. Yani ADC devresi hassas okuma direnci üzerinden

$$2.5 \text{ V} / 65536 = 38,1 \mu\text{V}$$

okuyabilir. $0,1 \mu\text{A} - 10 \mu\text{A}$ aralığında akım çıkışı verebilen akım kontrol devresinde hassas okuma direnci olarak $240 \text{ K}\Omega$ bulunmaktayken, $10 \mu\text{A} - 100 \mu\text{A}$ aralığında akım çıkışı verebilen hassas okuma direnci olarak $2,4 \text{ K}\Omega$ direnci kullanılmıştır. Bu nedenle kullanılan ADC, $0,1 \mu\text{A} - 10 \mu\text{A}$ akım devresinde okuyabileceği en küçük adım olarak,

$$38,1 \mu\text{V} / 240 \text{ K}\Omega = 158,9 \text{ pA}$$

değerine izin verirken, $10 \mu\text{A} - 100 \mu\text{A}$ akım devresinde okuyabileceği en küçük adım olarak,

$$38,1 \mu\text{V} / 2,4 \text{ K}\Omega = 15,9 \text{ nA}$$

değerine izin verir.

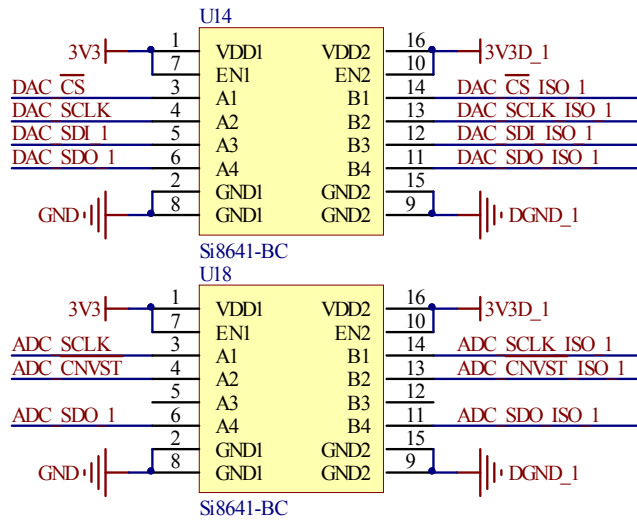
Kanallar arasındaki ADC'ler birbirine halka-zincirleme dizimi (daisy chain) ile bağlıdır. Bu sayede kodlamada, baskılı devre kartı tasarımında ve iletişimde kolaylık sağlanmıştır.

4.2.1.5 İzolatör Devreleri

Her bir akım kaynağı kanalının elektronik kartta bulunan FPGA ile bilgi alışverişi olmaktadır. 6 kanal için bir adet FPGA kullanılmaktadır. 6 kanal birbirinden izole olduğu için bu ortak alanda bulunan FPGA ile iletişimde bulunabilmesi için her bir kanalın kendi izolatör devreleri olmalıdır.

Burada bu izolatör devreleri için Si8641 [50] kullanılmıştır. Bu izolatör entegresi 3 giriş ve bir çıkış sayısal sinyali izole edebilmektedir. Her kanaldaki ADC ve DAC'lar halka zincirleme dizilimi ile birbirlerine bağlı olduğundan ADC ve DAC'ların birbirleri arasındaki sinyal iletişimi için, ikisi için de ayrı ayrı olmak üzere her kanalda 2 adet Si8641 kullanılmıştır.

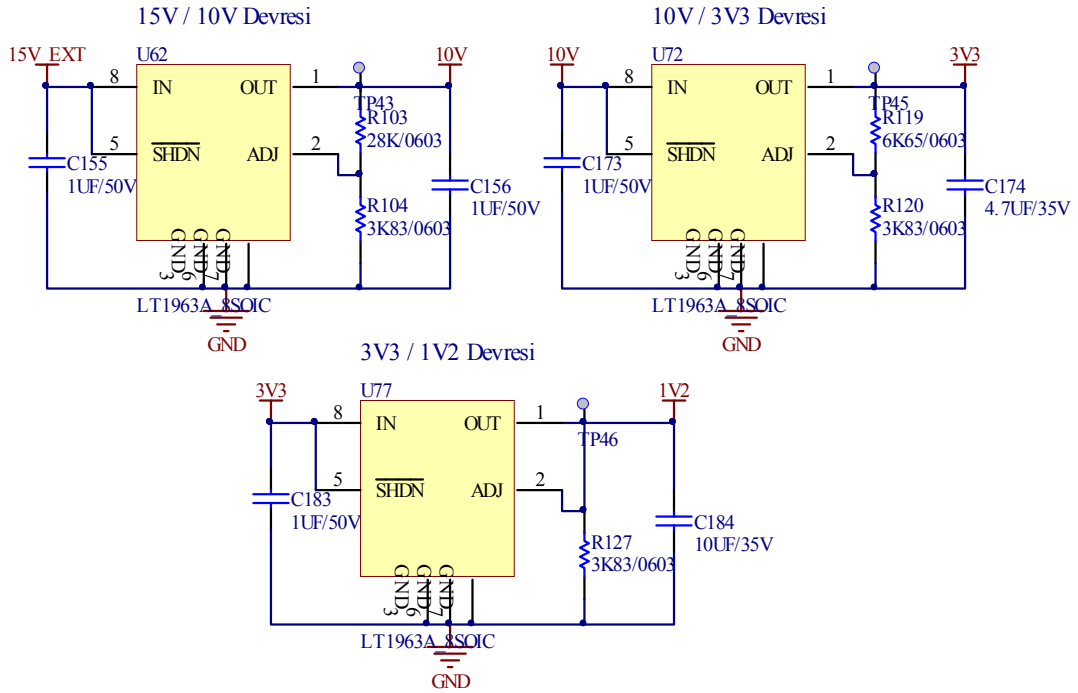
Bu devreler besleme olarak her kanaldan bir adet 3.3V beslemesine ve ortak kısımdan da bir adet 3.3V beslemesine ihtiyaç duyarlar. Kanal 1 için kullanılan izolasyon devreleri Şekil 4.8'de gösterilmiştir.



4.2.2 Ortak Devreler

4.2.2.1 Ortak Kısım Güç Besleme Devreleri

Bu devreler, 6 kanalın ortak kullandığı devreler için gerekli beslemeleri üretir. Kanallar izole olduğu için bu beslemelerin ayrıca üretilmesi gerekmektedir. Ortak kısımdaki devreler 3.3V ve 1.2V beslemelerini kullanmaktadır. Kart üstünde dışarıdan gelen yalnızca 12V – 18V beslemesi bulunmaktadır. Dönüşümler bu besleme üzerinden yapılacaktır. 12V – 18V seviyesinden 3.3V seviyesine düşüldüğünde lineer regülatör üstünde çok fazla enerji harcanacağından ve lineer regülatörü fazla ısıtacağından, ihtiyaç olan beslemeler haricinde bir ara 10V beslemesi de bu devrelerde üretilmiştir. Devre Şekil 4.9’da gösterilmiştir.

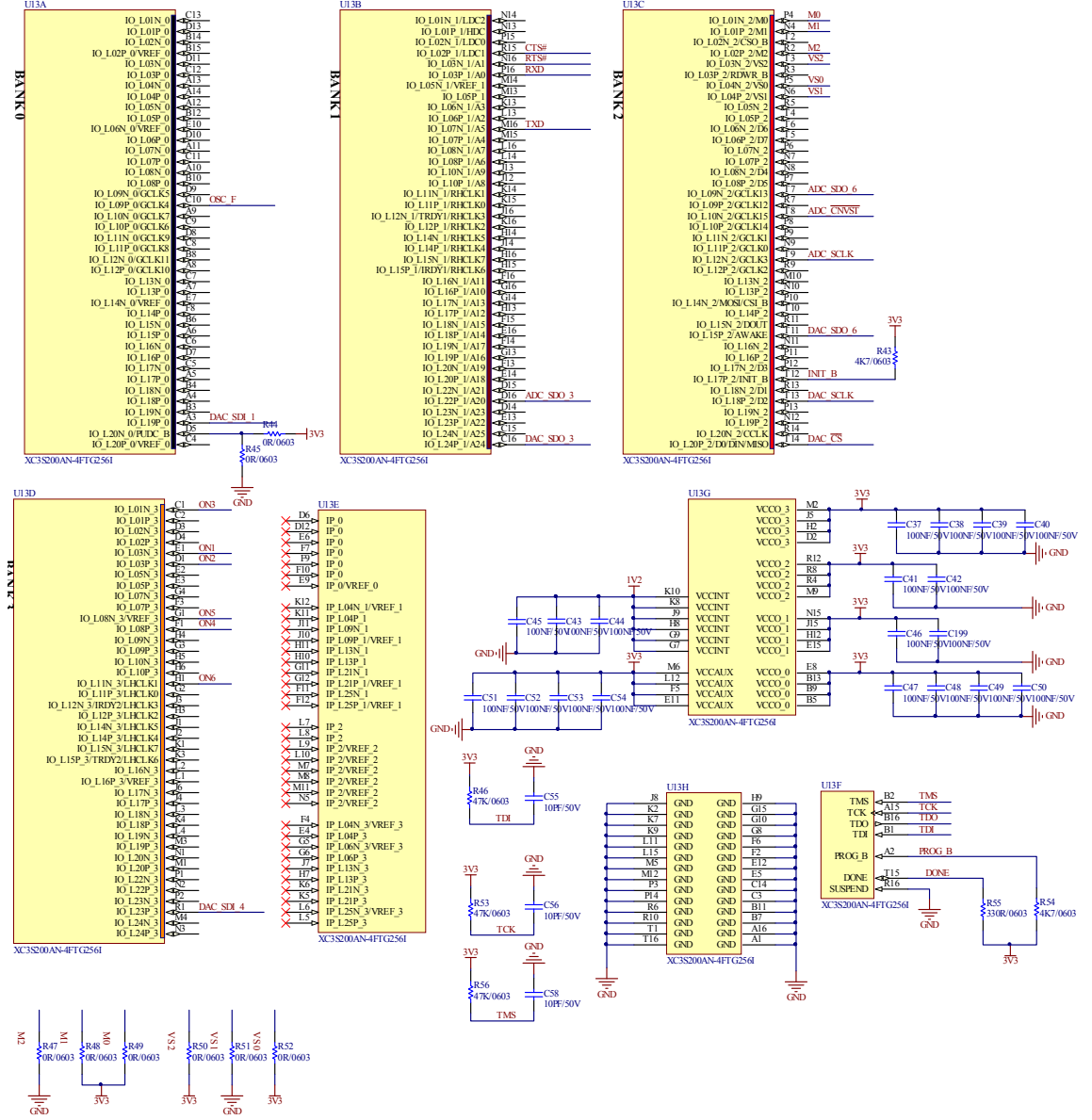


Şekil 4.9 Ortak kısım güç besleme devreleri

4.2.2.2 FPGA Devreleri

FPGA devreleri kart üstündeki FPGA'in (XC3S200AN-4FTG256I) sürülmesi için gerekli olan devrelerdir. Bu devreler XC3S200AN-4FTG256I'nın veri sayfalarında [46] anlatıldığı şekilde kurulmuştur. Şekil 4.10'da FPGA devreleri gösterilmiştir.

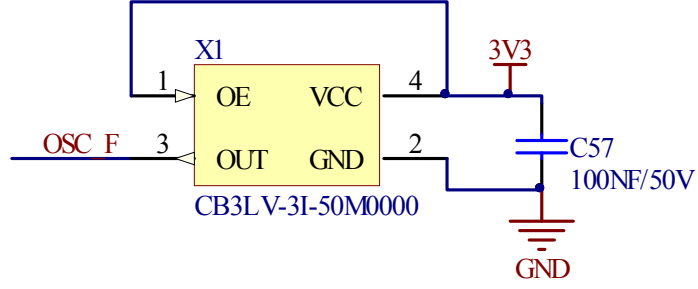
FPGA devreleri besleme olarak 3.3V ve 1.2V beslemelerini kullanmaktadır.



Şekil 4.10 FPGA devreleri [46]

4.2.2.3 Osilatör Devresi

FPGA'in saat frekansı için gerekli olan saat sinyali osilatör devresinde üretilir. Bu devrede 50 MHz'lik saat sinyali üretilir. Bu devrede osilatör olarak CB3LV-31-50M0000 [51] kullanılmıştır. Bu devre giriş besleme voltajı olarak 3.3V kullanır. Bu devre Şekil 4.11'de gösterilmiştir.

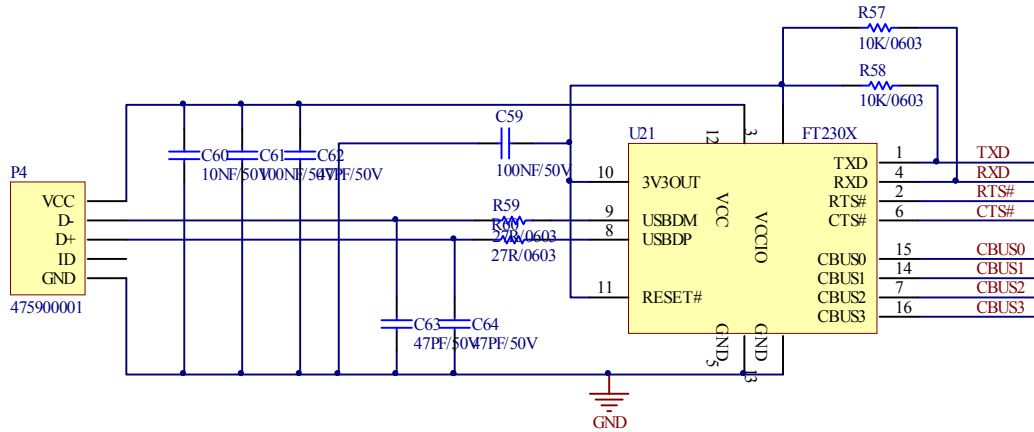


Şekil 4.11 Osilatör devresi

4.2.2.4 Seri Port Dönüşüm Devresi

Akım kaynağı bilgisayarla iletişimini USB üzerinden seri portla gerçekleştirmektedir. Bu nedenle FPGA'den çıkan bilginin bilgisayar tarafından alınabilmesi için seri porta dönüştürülmesi gerekmektedir. Bu iş için FT230X [52] entegresi kullanılmıştır. Bu devre Şekil 4.12'de gösterilmektedir.

Devrenin çalışması için kartta üretilen herhangi bir beslemeye gerek yoktur. Devre beslemesini bilgisayardan gelen USB portundaki 5V'u kullanmaktadır.



Şekil 4.12 Seri port dönüşüm devresi

4.3 Akım Kaynağı Yazılımı

Akım kaynağı yazılımı 2 kısımdan oluşmaktadır. Bunlardan ilki kart üzerindeki FPGA'in içine gömülü yazılımdır. Bu yazılım VHDL programlama dili ile

yazılmıştır. Diğer kısım da bilgisayar üstündeki kullanıcının gördüğü ve akım çıkış değerini ayarlayabildiği arayüz yazılımıdır. Bu yazılım da LabView ile oluşturulmuştur.

4.3.1 FPGA Yazılımı

FPGA üzerindeki yazılım 3 kısımdan oluşmaktadır. Bunlar, ADC ve DAC sürücüleri kısmı, çıkış akım kontrol kısmı ve bilgisayar veri arayüzü kısmıdır.

Bu projede FPGA yazılımı oluşturulurken öncelikle ADC ve DAC'ların FPGA ile kontrol edilebilmesine çalışılmıştır. Bunun için öncelikle ADC ve DAC'lar için sürücülerin yazılması gerekmektedir. Bu sürücülerin yazımı için tek referans kaynağı ADC ve DAC'ların veri sayfalarındaki zamanlama bilgileridir. Bunlar kullanılarak ADC ve DAC'lar için sürücü yazılımları oluşturulmuştur. Ek 1'de DAC için VHDL ile yazılan sürücü kodları, Ek 2'de de ADC için VHDL ile yazılan sürücü kodları verilmiştir.

FPGA yazılımının oluşturulmasının ikinci adımında akım çıkışlarının kontrolü sağlanmıştır. Bu aşamada bilgisayar arayüzü daha kullanımda olmadığından denemeler FPGA koduna akım çıkışını yazarak ve bunu kontrol etmeye çalışarak gerçekleştirilmiştir. Bu kısımda yapılan iş, ilk kullanıcı girişine göre DAC'a belli bir değer yazıldıktan sonra, ADC'den gelen akım çıkışının kontrol edilip, akım çıkışının düzeltilmesidir. FPGA sürekli olarak ADC'de okuduğu akım çıkışı değerine göre DAC'ı sürdüğü değeri düzeltir. Kontrol döngüsü 1 kHz'te gerçekleşmektedir. Bu geribesleme algoritmasının akış şeması Şekil 4.13'te gösterilmiştir. DAC kontrolcüsü için yazılan VHDL kodları Ek 3'te verilmiştir.

ADC ve DAC'lar başarılı bir şekilde sürülüp akım çıkışı da başarılı bir şekilde kontrol edildikten sonra bilgisayar arayüzünden gelen verinin tanımlanıp ayrıştırılması gereklidir. Üçüncü kısımda da bu işlem gerçekleştirilmektedir. FPGA ve bilgisayar arasındaki haberleşme USB üzerinden sanal seri port haberleşmesidir ve baud hızı 115200 bps'dir. FPGA'ye gelen bütün veri 47 bayt uzunluğundadır. FPGA'den de bilgisayara, bilgisayardaki arayüzün kullanıcı ekranına anlık akım çıkışını yansıtabilmesi için anlık akım bilgisi gitmektedir. FPGA'den bilgisayara

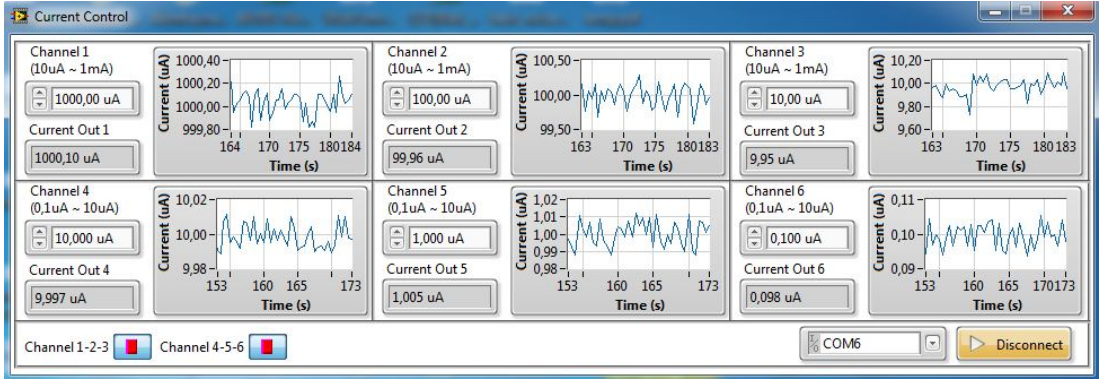
giden bu veri 35 bayt uzunluğundadır ve 2 Hz ile gönderilmektedir. Bu 35 baytlık verinin oluşturulması da bu kısımda yapılmaktadır. FPGA'in bilgisayardan gelen veriyi okuyabilmesi için yazılan VHDL kodu Ek 4'te verilmiştir.



Şekil 4.13 Geribesleme algoritmasının akış şeması

4.3.2 Arayüz Yazılımı

Kullanıcının akım çıkışı değerini girebilmesi için gerekli olan bilgisayar arayüzü yazılımıdır. LabView ile oluşturulmuştur. Bu arayüz Şekil 4.14'te gösterilmektedir.



Şekil 4.14 Akım kaynağı kullanıcı arayüzü

Arayüzde 6 kanal için ayrı ayrı kontroller bulunmaktadır. Akım kaynağından gelen akım çıkış bilgisine göre de her kanal için ayrı ayrı 20 saniyelik grafikler de ekranda gösterilir. Veriler 2 Hz ile bu arayüze gelmektedir. Akım kaynağından gelen bu 2 Hz'lik verilerde aynı zamanda ekranda sayısal olarak gösterilir.

Kullanıcı istediği kanala istediği akım çıkış değerini yazdığı zaman, bu arayüzde, DAC'a yazılması gereken değer ve kaynak bu akımı verdiğinde ADC tarafından R5 üzerinden okunması gereken değer hesaplanır. Bu hesaplanan değerler FPGA'ye gönderilir. Bu andan sonra, bilgisayardan FPGA'e, kullanıcı yeni bir akım çıkış değeri girene kadar, hiçbir veri gitmez, bu andan sonra artık bilgisayar hiçbirşeye karışmaz. Bu andan sonra bilgisayar arayüzünde yapılması gereken, akım kaynağından gelen verilen ekrana yansıtılması ve bu verilerden grafik oluşturulmasıdır. Kullanıcı bu arayüze her yeni çıkış akımı girdiğinde bu işlemler tekrarlanır.

Bu arayüzde ayrıca 1,2 ve 3 kanalları ile 4, 5 ve 6 kanalları grup olarak aynı anda açılabilir veya kapatılabilir. Bu şekilde bu kanalların durağan haldeyken ve kullanılmıyorken fazladan güç tüketmesinin önüne geçilir.

5. GELİŞTİRİLEN AKIM KAYNAĞININ TEST EDİLMESİ

Geliştirilen akım kaynağı ile 100 Ω 'luk yük sürülerek farklı akım çıkış değerleriyle testler yapılmıştır. Akım çıkışları 0.1 μ A, 1 μ A, 10 μ A, 100 μ A ve 1000 μ A olarak ayarlanıp gürültü miktarlarına bakıldı.

5.1 DC Akım Kaynağı için Gürültü Hesapları

SFQ devrelerinde gerekli olan besleme akımları için kullanılacak akım kaynaklarında, gürültünün önemli olduğu frekanslar DC ve DC'ye yakın frekanslardır. Çünkü bu kaynaklar DC akım çıkışlı kaynaklardır. Genellikle bu kaynaklardaki akım gürültü yoğunluğu, 0 Hz – 10 Hz frekans aralığı için verilmektedir [38], [53]. Akım gürültüsünün birimi, verilen frekans bandındaki karekök Hertz başına düşen pikoamper ($\text{pA}/\sqrt{\text{Hz}}$) veya nanoamper ($\text{nA}/\sqrt{\text{Hz}}$) olarak ifade edilir [54]. Akım gürültüsüne aynı zamanda, eşdeğer açık devre RMS gürültü akımı da denir.

Herhangi bir DC akım kaynağının herhangi bir frekans aralığındaki akım gürültüsünün hesaplanması için farklı yöntemler uygulanabilir. Bu çalışmada akım gürültüsü hesaplamaları için şu adımlar takip edilmiştir;

- Farklı akım çıkışları ile 100 Ω yük sürüldü.
- Yük üstünden yaklaşık 17 ve 60 dakikalık voltaj verileri 50 Hz ve 100 Hz frekanslarla alındı.
- Bu verilerin akım gürültü spektrumları Matlab ile hesaplandı. Öncelikle alınan voltaj verileri yükümüz olan 100 Ω ' bölünerek akıma dönüştürüldü ve bu verilerin Matlab'ta fft'si (Fast Fourier Transform) alındı.
- Sonrasında bu verilerden 10 Hz'e kadar olan kısmı toplanarak 0 Hz – 10 Hz aralığındaki toplam gürültü bulundu. Toplam gürültü, (3)'teki gibi 0 Hz – 10 Hz arasındaki frekanslardaki hesaplanan gürültülerin karelerinin toplamının karekökü olarak hesaplanmıştır [45], [55].

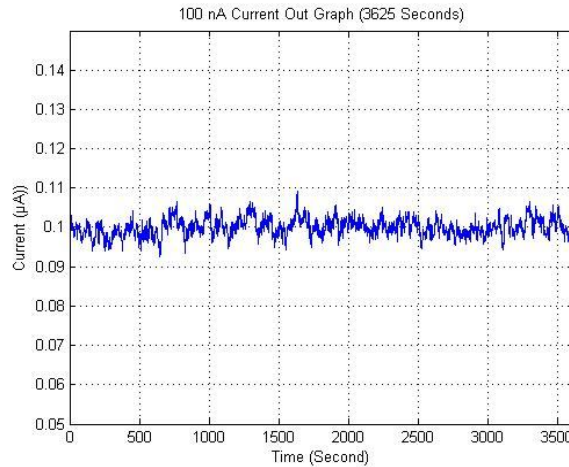
Bu testlerde ölçümler 26 bit 7½-Digit NI PXI-4071 PXI Digital Multimeter [56] ile yapılmıştır. Çünkü 100 Ω yükten 100nA akım geçtiğinde üzerinde yaklaşık 10^{-5} V oluşacağından ± 1 V ölçüm aralığı olan bir multimetrenin en az 20 bit olması gerekir ki, bu durumda bu multimetre ancak 1,91 μ V'u okuyabilir.

$$e_t = \sqrt{\sum_{i=f_1}^{f_{max}} e_{f_i}^2}$$

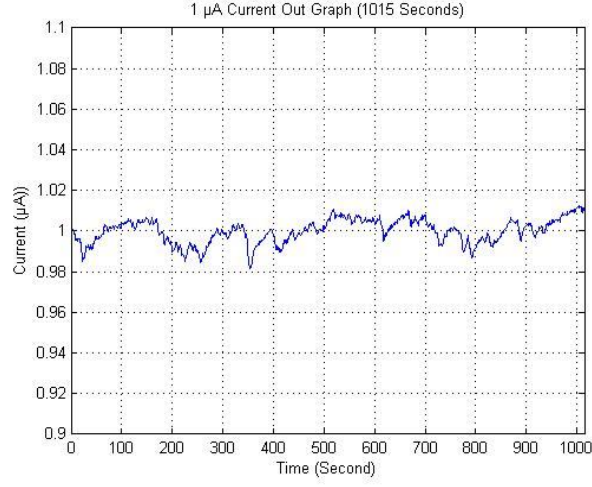
Burada e_t çıkış akımının toplam akım gürültüsü, f_{max} gürültüsü hesaplanan frekans aralığının en yüksek frekansı, e_{f_i} her bir frekans aralığı için olan akım gürültüsüdür. Bu formülde burada geliştirilen akım kaynağı için yapılan hesaplamalarda f_{max} değeri 10 Hz olarak alınmıştır.

5.2 Test Sonuçları

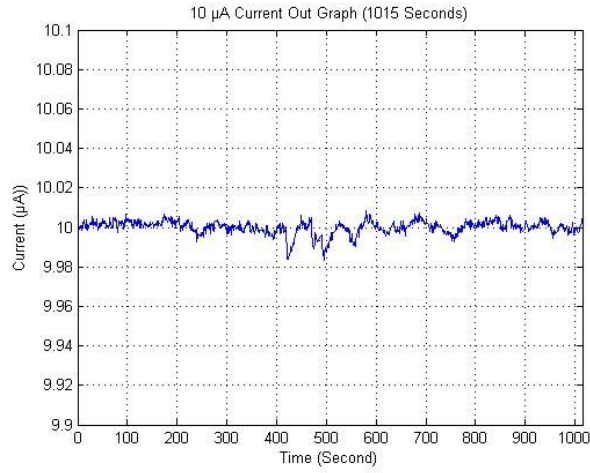
Yapılan testlerde yalnızca 0,1 μ A akım çıkışı testinde yaklaşık 1 saat 50 Hz ile, geri kalan diğer testlerin hepsinde, yaklaşık 17 dakika ve 200 Hz ile veri alınmıştır. Akım kaynağının en düşük değerinin daha uzun süreli ve detaylı incelenmesi istendiğinden 0,1 μ A akım çıkışından daha uzun süreli veri alınmıştır. Verilerin zamanla değişimini gösteren grafikler Şekil 5.1 - 5.5'te görülmektedir.



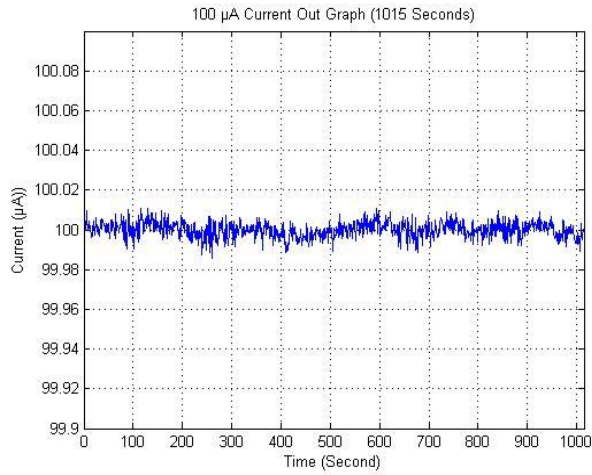
Şekil 5.1 0,1 μ A akım çıkışının 100 Ω üzerindeki yaklaşık 1 saatlik verisi



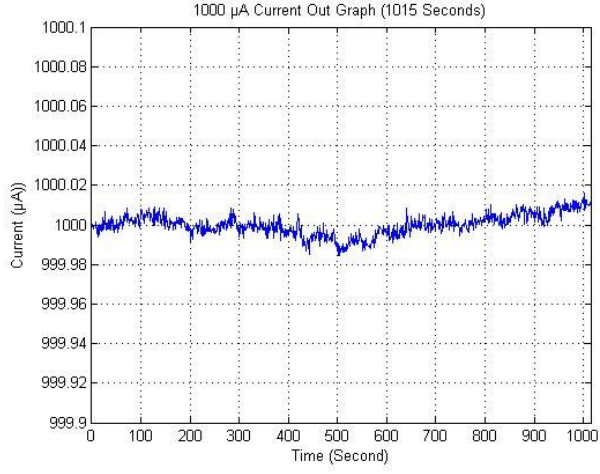
Şekil 5.2 1 μA akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi



Şekil 5.3 10 μA akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi

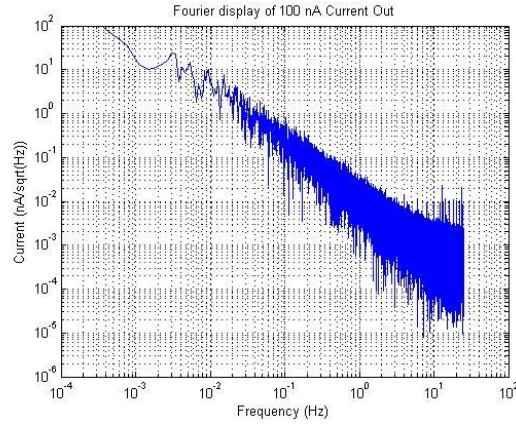


Şekil 5.4 100 μA akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi

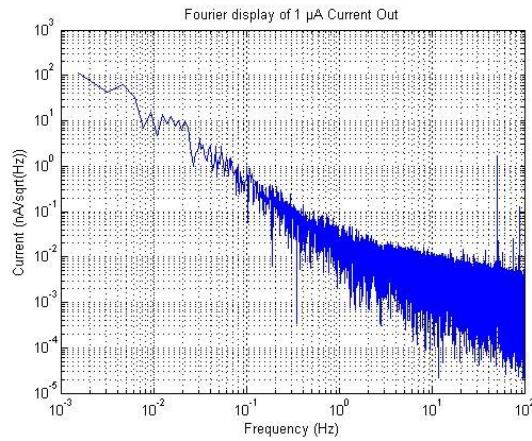


Şekil 5.5 1000 μA akım çıkışının 100 Ω üzerindeki yaklaşık 17 dakikalık verisi

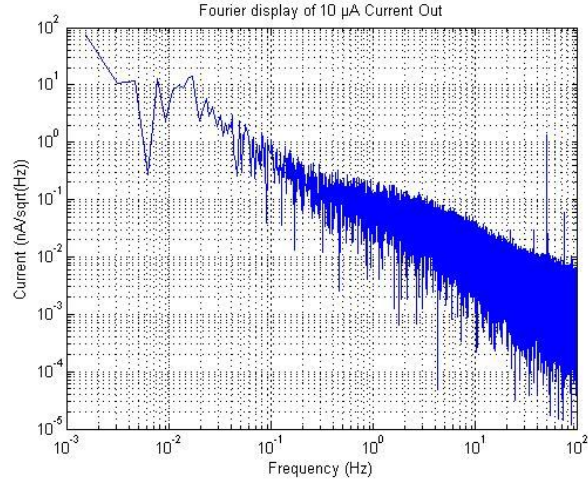
Bu verilerin Matlab'ta hesaplanan akım gürültü spektrumları da Şekil 5.6, 5.7, 5.8, 5.9 ve 5.10'da gösterilmektedir.



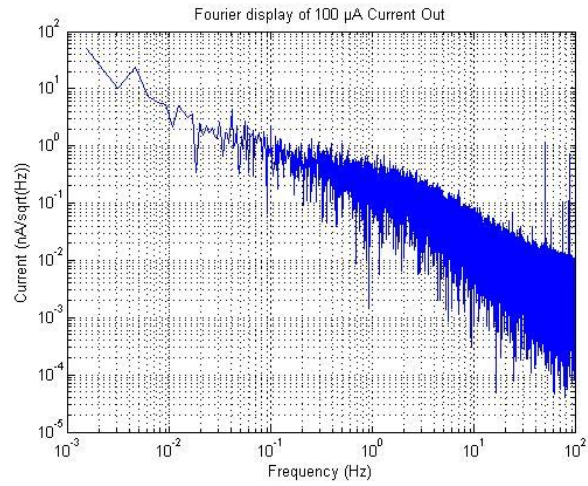
Şekil 5.6 0,1 μA akım çıkış verisinin akım gürültü spektrumu



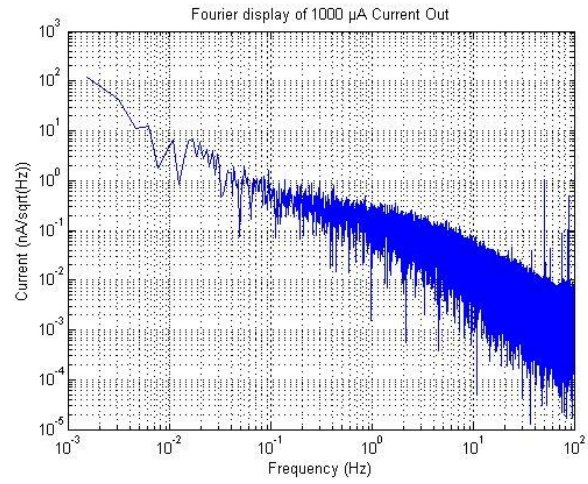
Şekil 5.7 1 μA akım çıkış verisinin akım gürültü spektrumu



Şekil 5.8 10 µA akım çıkış verisinin akım gürültü spektrumu



Şekil 5.9 100 µA akım çıkış verisinin akım gürültü spektrumu



Şekil 5.10 1000 µA akım çıkış verisinin akım gürültü spektrumu

Bu verilerden elde edilen sonuçlara göre akım kaynağının farklı akım çıkış değerlerinde görülen toplam gürültü (integrated noise) miktarları Çizelge 5.1’de verilmiştir.

Çizelge 5.1 Farklı akım çıkışları için hesaplanan toplam gürültüler

Akım Çıkışı (μA)	Frekans Aralığı	Toplam Gürültü Akımı ($\text{nA}/\sqrt{\text{Hz}}$)
0.1	1.1 mHz - 10 Hz	5.3
1	1.5 mHz - 10 Hz	8.5
10	1.5 mHz - 10 Hz	7.8
100	1.5 mHz - 10 Hz	11.1
1000	1.5 mHz - 10 Hz	13.9

6. SONUÇ

Bu çalışma, çok noktadan besleme akımı gerektiren süperiletken entegre devreler ve süperiletken algılayıcılar için kullanılacak olan çok kanallı, kararlı, düşük maliyetli, düşük gürültülü, yüksek hassasiyetli ve geniş aralıklı akım kaynağı geliştirilmesidir. Piyasadan satın alınabilecek olan bu tarz akım kaynaklarındaki en önemli sorun tek çıkışlarının olması ve bununla birlikte maliyetlerinin yüksek olmasıdır. Süperiletken devrelerde çok noktadan akım beslemesi gerektiğinden bu akım kaynaklarından fazla sayıda almak gerekmektedir. Bu da maliyetin yükselmesine ve akım kaynakları için laboratuvarında ayrılacak yerin artmasına neden olmaktadır.

Bu geliştirilen akım kaynağında 6 adet birbirinden tamamen bağımsız çıkış bulunmaktadır ve bu 6 kanalın hepsi de akım kontrolü için yalnızca bir FPGA entegresi kullanmaktadır. Ayrıca bu 6 kanallı akım kaynağını akım kaynağı kutusunun içine bir USB çoklayıcı koyarak çoklamak mümkündür. Yazılımda da gerekli ayarlamalar yapılarak, kanal sayısı 6'dan 12'ye, 18'e, 24'e vs. çıkartılabilir.

Bu cihaz zaman ve sıcaklık gibi çevresel koşullardan bağımsız olarak kararlı akım çıkışları vermektedir. Bu, devreye uygulanan geri besleme algoritması ile, devrenin çıkışının devamlı kontrol edilmesi ile sağlanmaktadır. Yani devre, çevresel koşullardan ve zamandan bağımsız olarak kurulan kapalı bir döngü içerisinde devamlı kendini kalibre etmektedir. Burada kurulan kapalı döngüde en önemli komponent R5 direncidir. Bu direncin hassasiyeti çok yüksektir (0.05% tolerans) ve sıcaklık değişimlerinden az etkilenmektedir (± 5 ppm/ $^{\circ}$ C) [57].

Akım kaynağının çıkışına bağlı olarak gürültü seviyesi çok etkilenmemektedir. Cihaz akım çıkışı 1 μ A'ken toplam gürültü miktarı 8.5 nA ölçülürken, cihaz çıkışı 1000 μ A olduğunda da toplam gürültü miktarı 13.9 nA olarak ölçülmüştür. Yani çıkış akımı yükseldikçe gürültünün akım çıkışındaki yüzdelik miktarı gittikçe düşmektedir. Bunun sebebinin ADC ve DAC'ın düşük çözünürlüklü olmasından dolayı kaynaklandığı düşünülmektedir. Burada ADC 16 bit olarak kullanılmakta ve DAC da 18 bit olarak kullanılmaktadır. Çok küçük akımlardaki gürültünün akıma oranının yüksek akımlardaki gürültünün akıma oranından yüksek olmasının sebebinin,

ADC'nin minimum adım genişliğinden kaynaklandığı düşünülmektedir. Küçük akımlardaki küçük akım değişikliklerini, ADC düşük bitli olmasından dolayı algılayamazken, yüksek akımlarda bu durum çok sorun oluşturmamaktadır. DAC'ta da yine aynı şekilde düşük akımlarda DAC'ta oluşturulan referans sinyali için voltaj aralıkları yüksek gelirken, yüksek akımlarda yine bu çok sorun oluşturmamaktadır. Bu nedenle bu ADC ve DAC, 24 bitlik olanlarla değiştirilirse, düşük akımlarda da gürültünün akıma oranının düşmesi beklenmektedir. Ayrıca ADC ve DAC'ın 16 ve 18 bit olmalarından dolayı düşük ve yüksek aralık akım kaynaklarında adım aralıkları sırasıyla 1 nA ve 10 nA olarak ayarlanmıştır. ADC ve DAC entegreleri daha yüksek bitli olanlarla değiştirildiğinde de bu adım aralıkları yine daha da düşecektir.

Özet olarak, burada süperiletken entegre devrelerin besleme akımlarını üretmek için kullanmaya uygun, LabView ile bilgisayardan kontrol edilebilen, düşük maliyetli, düşük gürültülü ve kararlı bir akım kaynağı geliştirildi ve test edildi.

KAYNAKLAR

- [1] V. K. Semenov ve D. V. Averin, “SFQ control circuits for Josephson junction qubits”, *IEEE Trans. Appl. Supercond.*, c. 13, sayı 2, ss. 960–965, Haz. 2003.
- [2] K. K. Likharev ve V. K. Semenov, “RSFQ logic/memory family: a new Josephson-junction technology for sub-terahertz-clock-frequency digital systems”, *IEEE Trans. Appl. Supercond.*, c. 1, sayı 1, ss. 3–28, Mar. 1991.
- [3] O. A. Mukhanov, V. K. Semenov, W. Li, T. V. Filippov, D. Gupta, A. M. Kadin, D. K. Brock, A. F. Kirichenko, Y. A. Polyakov, ve I. V. Vernik, “A superconductor high-resolution ADC”, *IEEE Trans. Appl. Supercond.*, c. 11, sayı 1, ss. 601–606, Mar. 2001.
- [4] N. B. Dubash, V. V. Borzenets, Y. M. Zhang, V. Kaplunenko, J. W. Spargo, A. D. Smith, ve T. Van Duzer, “System demonstration of a multigigabit network switch”, *IEEE Trans. Microw. Theory Tech.*, c. 48, sayı 7, ss. 1209–1215, Tem. 2000.
- [5] M. Ozer, M. Eren Çelik, Y. Tukul, ve A. Bozbey, “Design of RSFQ wave pipelined Kogge–Stone Adder and developing custom compound gates”, *Cryogenics*, c. 63, ss. 174–179, Eyl. 2014.
- [6] M. Dorojevets, P. Bunyk, ve D. Zinoviev, “FLUX chip: design of a 20-GHz 16-bit ultrapipelined RSFQ processor prototype based on 1.75- μm LTS technology”, *IEEE Trans. Appl. Supercond.*, c. 11, sayı 1, ss. 326–332, Mar. 2001.
- [7] P. Bunyk, M. Leung, J. Spargo, ve M. Dorojevets, “Flux-1 RSFQ microprocessor: physical design and test results”, *IEEE Trans. Appl. Supercond.*, c. 13, sayı 2, ss. 433–436, Haz. 2003.
- [8] M. Dorojevets, C. L. Ayala, N. Yoshikawa, ve A. Fujimaki, “8-Bit Asynchronous Sparse-Tree Superconductor RSFQ Arithmetic-Logic Unit With a Rich Set of Operations”, *IEEE Trans. Appl. Supercond.*, c. 23, sayı 3, ss. 1700104–1700104, Haz. 2013.
- [9] S. Nakamura, H. Numabe, A. Bozbey, ve A. Fujimaki, “Current Resolution of a Single-Flux-Quantum Readout Circuit Based on Current-to-Time Conversion Toward a Flux Qubit System”, *IEEE Trans. Appl. Supercond.*, c. 19, sayı 3, ss. 973–976, Haz. 2009.
- [10] S. Miyajima, T. Ortlepp, H. Toepfer, A. Bozbey, ve A. Fujimaki, “Experimental Demonstration and Numerical Analysis of Microampere Gray Zone Width with Enhanced Operating Margin in Shunted Quasi-One Junction Superconducting Quantum Interference Device Comparators”, *Jpn. J. Appl. Phys.*, c. 52, sayı 3R, s. 033101, Mar. 2013.
- [11] A. Bozbey, S. Miyajima, T. Ortlepp, ve A. Fujimaki, “Design and Circuit Analysis of Quasi-one Junction SQUID Comparators for Low Temperature Detector Array Read-out”, *J. Supercond. Nov. Magn.*, c. 24, sayı 1–2, ss. 1065–1069, Eyl. 2010.

- [12] A. Bozbey, S. Miyajima, H. Akaike, ve A. Fujimaki, “Single-Flux-Quantum Circuit Based Readout System for Detector Arrays by Using Time to Digital Conversion”, *IEEE Trans. Appl. Supercond.*, c. 19, sayı 3, ss. 509–513, Haz. 2009.
- [13] Y. Kameda, S. Yorozu, ve Y. Hashimoto, “Automatic Single-Flux-Quantum (SFQ) Logic Synthesis Method for Top-Down Circuit Design”, *J. Phys. Conf. Ser.*, c. 43, sayı 1, s. 1179, Haz. 2006.
- [14] H. Terai, Y. Kameda, S. Yorozu, A. Fujimaki, ve Z. Wang, “The effects of DC bias current in large-scale SFQ circuits”, *IEEE Trans. Appl. Supercond.*, c. 13, sayı 2, ss. 502–506, Haz. 2003.
- [15] “Current Source | Multi-Channel Programmable | Model CS-48-100 | Semiconductor | Digital Superconductor | Hypres Inc.” [Çevrimiçi]. Available at: <http://www.hypres.com/products/current-source/>. [Erişim: 07-Mar-2015].
- [16] S. Miki, H. Terai, T. Yamashita, K. Makise, M. Fujiwara, M. Sasaki, ve Z. Wang, “Superconducting single photon detectors integrated with single flux quantum readout circuits in a cryocooler”, *Appl. Phys. Lett.*, c. 99, sayı 11, s. 111108, Eyl. 2011.
- [17] C. M. Natarajan, M. G. Tanner, ve R. H. Hadfield, “Superconducting nanowire single-photon detectors: physics and applications”, *Supercond. Sci. Technol.*, c. 25, sayı 6, s. 063001, Haz. 2012.
- [18] D. Haddad, B. Waltrip, ve R. L. Steiner, “Low noise programmable current source for the NIST-3 and NIST-4 watt balance”, içinde *2012 Conference on Precision Electromagnetic Measurements (CPEM)*, 2012, ss. 336–337.
- [19] H. K. Onnes, “Further experiments with liquid helium. C. On the change of electric resistance of pure metals at very low temperatures etc. IV. The resistance of pure mercury at helium temperatures”, içinde *Through Measurement to Knowledge*, K. Gavroglu ve Y. Goudaroulis, Ed. Springer Netherlands, 1991, ss. 261–263.
- [20] “Electrical Conduction in Metals and Alloys (Electrical Properties of Materials) Part 2”. [Çevrimiçi]. Available at: <http://what-when-how.com/electronic-properties-of-materials/electrical-conduction-in-metals-and-alloys-electrical-properties-of-materials-part-2/>. [Erişim: 31-Mar-2015].
- [21] W. Meissner ve R. Ochsenfeld, “Ein neuer Effekt bei Eintritt der Supraleitfähigkeit”, *Naturwissenschaften*, c. 21, sayı 44, ss. 787–788, Kas. 1933.
- [22] “physicsfigures”. [Çevrimiçi]. Available at: <http://users-physics.au.dk/philip/pictures/physicsfigures/physicsfigures.html>. [Erişim: 31-Mar-2015].
- [23] H. Rogalla ve P. H. Kes, Ed., *100 Years of Superconductivity*, 1 edition. Boca Raton: CRC Press, 2011.
- [24] B. D. Josephson, “Possible new effects in superconductive tunnelling”, *Phys. Lett.*, c. 1, sayı 7, ss. 251–253, Tem. 1962.

- [25] I. Avci, R. Akram, A. Bozbey, M. Tepe, ve D. Abukay, “Selection of the Best Proper DC-SQUIDS in a Multi-SQUID Configuration”, *IEEE Trans. Appl. Supercond.*, c. 17, sayı 2, ss. 680–682, Haz. 2007.
- [26] J. G. Bednorz ve K. A. Müller, “Possible highT_c superconductivity in the Ba–La–Cu–O system”, *Z. Für Phys. B Condens. Matter*, c. 64, sayı 2, ss. 189–193, Haz. 1986.
- [27] ITRS, “International Technology Roadmap for Semiconductors 2007 - Emerging Research Devices”. 2007.
- [28] C. J. Burroughs, S. P. Bent, T. E. Harvey, ve C. A. Hamilton, “1 volt DC programmable Josephson voltage standard”, *IEEE Trans. Appl. Supercond.*, c. 9, sayı 2, ss. 4145–4149, Haz. 1999.
- [29] T. V. Duzer ve C. W. Turner, *Principles of superconductive devices and circuits*. Elsevier, 1981.
- [30] D. E. Kirichenko, S. Sarwana, ve A. F. Kirichenko, “Zero Static Power Dissipation Biasing of RSFQ Circuits”, *IEEE Trans. Appl. Supercond.*, c. 21, sayı 3, ss. 776–779, Haz. 2011.
- [31] O. A. Mukhanov, “Energy-Efficient Single Flux Quantum Technology”, *IEEE Trans. Appl. Supercond.*, c. 21, sayı 3, ss. 760–769, Haz. 2011.
- [32] M. H. Volkmann, A. Sahu, C. J. Fourie, ve O. A. Mukhanov, “Implementation of energy efficient single flux quantum digital circuits with sub-aJ/bit operation”, *Supercond. Sci. Technol.*, c. 26, sayı 1, s. 015002, Oca. 2013.
- [33] “Current mirror”, *Wikipedia, the free encyclopedia*. 12-Şub-2015.
- [34] H. H. Kuntman, *Analog tümdevre tasarımı*. Birsen yayınevi, 1998.
- [35] “Chapter 11: The Current Mirror [Analog Devices Wiki]”. [Çevrimiçi]. Available at: http://wiki.analog.com/university/courses/electronics/text/chapter-11#widlar_current_source. [Erişim: 30-Mar-2015].
- [36] “CHAPTER 4 - CMOS SUBCIRCUITS - Current_Mirrors.pdf”. [Çevrimiçi]. Available at: http://users.ece.gatech.edu/phasler/Courses/ECE4430/Unit2/Current_Mirrors.pdf. [Erişim: 30-Mar-2015].
- [37] “Current sources for fiber-optic lasers: a compendium of pleasant current events”, *EDN*. [Çevrimiçi]. Available at: <http://www.edn.com/design/analog/4346595/Current-sources-for-fiber-optic-lasers-a-compendium-of-pleasant-current-events>. [Erişim: 19-Mar-2015].
- [38] Keithley Instruments, “Model 6220 DC Current Source and Model 6221 AC and DC Current Source Datasheet”. [Çevrimiçi]. Available at: <http://www.keithley.com/data?asset=15911>.
- [39] A. van der Ziel, “Noise in solid-state devices and lasers”, *Proc. IEEE*, c. 58, sayı 8, ss. 1178–1206, Ağu. 1970.
- [40] B. M. Oliver, “Thermal and quantum noise”, *Proc. IEEE*, c. 53, sayı 5, ss. 436–454, May. 1965.

- [41] P. R. Gray ve R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 2nd baskı. New York, NY, USA: John Wiley & Sons, Inc., 1990.
- [42] M. Stoisiak ve D. Wolf, “Origin of 1/f noise in bipolar transistors”, *IEEE Trans. Electron Devices*, c. 27, sayı 9, ss. 1753–1757, Eyl. 1980.
- [43] J. W.M. Leach, “Fundamentals of low-noise analog circuit design”, *Proc. IEEE*, c. 82, sayı 10, ss. 1515–1538, Eki. 1994.
- [44] H. W. Ott, *Noise reduction techniques in electronic systems*. John Wiley & Sons Canada, Limited, 1976.
- [45] “Noise Analysis In Operational Amplifier Circuits (Rev. B - slva043b.pdf”. [Çevrimiçi]. Available at: <http://www.ti.com/lit/an/slva043b/slva043b.pdf>. [Erişim: 23-Mar-2015].
- [46] “Spartan-3AN FPGA Family Data Sheet (DS557) - ds557.pdf”. [Çevrimiçi]. Available at: http://www.xilinx.com/support/documentation/data_sheets/ds557.pdf. [Erişim: 25-Mar-2015].
- [47] “DAC9881 | Precision DAC (≤ 10 MSPS) | Digital to Analog Converter | Description & parametrics”. [Çevrimiçi]. Available at: <http://www.ti.com/product/dac9881>. [Erişim: 25-Mar-2015].
- [48] “AD7660 | datasheet and product info 16-Bit 100 kSPS CMOS Successive Approximation PulSAR® ADC with No Missing Codes | Analog Devices”. [Çevrimiçi]. Available at: <http://www.analog.com/en/products/analog-to-digital-converters/ad-converters/ad7660.html#product-overview>. [Erişim: 25-Mar-2015].
- [49] “PH02S/D Series.pdf”. [Çevrimiçi]. Available at: http://www.deltaww.com/filecenter/Products/download/01/0102/datasheet/DS_PH02S&D.pdf. [Erişim: 26-Mar-2015].
- [50] “Si864x Data Sheet - Si864x.pdf”. [Çevrimiçi]. Available at: <https://www.silabs.com/Support%20Documents/TechnicalDocs/Si864x.pdf>. [Erişim: 27-Mar-2015].
- [51] “Model CB3-CB3LV Clock Oscillators.pdf”. [Çevrimiçi]. Available at: <http://www.ctscorp.com/components/Datasheets/008-0256-0.pdf>. [Erişim: 26-Mar-2015].
- [52] “FT230X - DS_FT230X.pdf”. [Çevrimiçi]. Available at: http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT230X.pdf. [Erişim: 27-Mar-2015].
- [53] S. Linzen, T. L. Robertson, T. Hime, B. L. T. Plourde, P. A. Reichardt, ve J. Clarke, “Low-noise computer-controlled current source for quantum coherence experiments”, *Rev. Sci. Instrum.*, c. 75, sayı 8, ss. 2541–2544, Ağu. 2004.
- [54] “AN-104 Noise Specs Confusing (Rev. C) - snva515c.pdf”. [Çevrimiçi]. Available at: <http://www.ti.com/lit/an/snva515c/snva515c.pdf>. [Erişim: 21-Mar-2015].

- [55] “AN1560: Making Accurate Voltage Noise and Current Noise Measurements on Operational Amplifiers Down to 0.1Hz - an1560.pdf”. [Çevrimiçi]. Available at: <http://www.intersil.com/content/dam/Intersil/documents/an15/an1560.pdf>. [Erişim: 23-Mar-2015].
- [56] “NI PXI-4071 PXI Digital Multimeter (DMM) - National Instruments”. [Çevrimiçi]. Available at: <http://sine.ni.com/nips/cds/view/p/lang/en/nid/14857>. [Erişim: 22-Mar-2015].
- [57] “Susumu RG Series.pdf”. [Çevrimiçi]. Available at: http://www.susumu-usa.com/pdf/RG_RM_RGH_DATASHEET.pdf. [Erişim: 26-Mar-2015].

EKLER

Ek 1: DAC için VHDL ile yazılan ve FPGA'ye gömülen sürücü kodları

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity DAC9881_Driver is
    generic (
        clkFreq : integer := 50_000_000
    );
    Port (
        clk      : in std_logic;
        DAC9881_SCLK : out std_logic;
        DAC9881_SDI : out std_logic;
        DAC9881_CS : out std_logic;
        yeniVeri  : in std_logic;
        DACData1  : in std_logic_vector(17 downto 0);
        DACData2  : in std_logic_vector(17 downto 0);
        DACData3  : in std_logic_vector(17 downto 0);
        tamamlandi : out std_logic
    );
end DAC9881_Driver;

architecture Behavioral of DAC9881_Driver is

    constant sclkCntrLim : integer := clkFreq/1_000_000/2;

    signal sendReg : std_logic_vector(71 downto 0) := (others => '0');
    signal sclkReg : std_logic_vector(1 downto 0) := (others => '0');

    signal sclkCntr : integer range 0 to sclkCntrLim := 0;
    signal bitCntr  : integer range 0 to sendReg'length := 0;

    signal cs      : std_logic := '1';
    signal sclk    : std_logic := '0';
    signal sclkFall : std_logic := '0';
    signal veriYazildi : std_logic := '0';

    begin
        DAC9881_CS <= cs;
        DAC9881_SCLK <= sclk;
        tamamlandi <= '1' when veriYazildi = '1' and cs = '0' else '0';

    process(clk)
    begin
        if clk'event and clk = '1' then
            if yeniVeri = '1' then
                cs <= '0';
            elsif veriYazildi = '1' then
                cs <= '1';
            end if;
        end if;
    end process;
end Behavioral;
```



```

        end if;
    end if;
end process;

process(clk)
begin
if clk'event and clk = '1' then
    sclkReg <= sclkReg(0) & sclk;
    if cs = '1' then
        sclkCntr <= 0;
        sclk <= '0';
    elsif sclkCntr = sclkCntrLim - 1 then
        sclkCntr <= 0;
        sclk <= not sclk;
    else
        sclkCntr <= sclkCntr + 1;
    end if;
end if;
end process;
sclkFall <= '1' when sclkReg = "10" else '0';

process(clk)
begin
if clk'event and clk = '1' then
    if cs = '1' then
        sendReg <= "000000" & DACData1 & "000000" & DACData2 & "000000" &
DACData3;
        bitCntr <= 0;
        veriYazildi <= '0';
    elsif sclkFall = '1' then
        if bitCntr = sendReg'length - 1 then
            veriYazildi <= '1';
        else
            bitCntr <= bitCntr + 1;
            sendReg <= sendReg(sendReg'left-1 downto 0) & '0';
        end if;
    end if;
end if;
end process;
DAC9881_SDI <= sendReg(sendReg'left);

end Behavioral;

```

Ek 2: ADC için VHDL ile yazılan ve FPGA'ye gömülen sürücü kodları.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity AD7660_Driver is
    generic (
        clkFreq : integer := 50_000_000
    );
    Port (
        clk      : in std_logic;
        AD7678_sclk : out std_logic;
        AD7678_cnvst : out std_logic;
        AD7678_sdout : in std_logic;
        veriOkundu  : out std_logic;
        trig       : in std_logic;
        adcData1   : out std_logic_vector(15 downto 0);
        adcData2   : out std_logic_vector(15 downto 0);
        adcData3   : out std_logic_vector(15 downto 0)
    );
end AD7678_Driver;

architecture Behavioral of AD7678_Driver is

    constant sclkCntrLim : integer := clkFreq/1_000_000/2;
    constant delayCntrLim : integer := clkFreq/50_000/2;

    signal sclkReg : std_logic_vector(1 downto 0) := (others => '0');
    signal cnvstReg : std_logic_vector(1 downto 0) := (others => '0');
    signal tempReg : std_logic_vector(47 downto 0) := (others => '0');

    signal sclkCntr : integer range 0 to sclkCntrLim := 0;
    signal delayCntr : integer range 0 to delayCntrLim := 0;
    signal bitCntr : integer range 0 to tempReg'length := 0;
    signal cntr : integer range 0 to 65535 := 0;

    signal cnvst      : std_logic := '1';
    signal sclk       : std_logic := '0';
    signal sclkFall   : std_logic := '0';
    signal yeniVeri   : std_logic := '0';
    signal cntrDoldu  : std_logic := '0';
    signal cnvstFall  : std_logic := '0';
    signal cntrRst    : std_logic := '1';
    signal cnvstdelayed : std_logic := '1';

begin

    AD7678_sclk <= sclk;
    --AD7678_cnvst <= cnvst;
    AD7678_cnvst <= cntrRst;
```

```

process(clk)
begin
if clk'event and clk = '1' then
    if trig = '1' then
        cnvst <= '0';
    elsif yeniVeri = '1' then
        cnvst <= '1';
    end if;
end if;
end process;

process(clk)
begin
if clk'event and clk = '1' then
    sclkReg <= sclkReg(0) & sclk;
    if cnvstdelayed = '1' then
        sclkCntr <= 0;
        sclk <= '0';
    elsif sclkCntr = sclkCntrLim - 1 then
        sclkCntr <= 0;
        sclk <= not sclk;
    else
        sclkCntr <= sclkCntr + 1;
    end if;
end if;
end process;
sclkFall <= '1' when sclkReg = "10" else '0';

process(clk)
begin
if clk'event and clk = '1' then
    if cnvst = '1' then
        bitCntr <= 0;
        yeniVeri <= '0';
    elsif sclkFall = '1' then
        if bitCntr = tempReg'length-1 then
            yeniVeri <= '1';
            ADCData1 <= tempReg(46 downto 31);
            ADCData2 <= tempReg(30 downto 15);
            ADCData3 <= tempReg(14 downto 0) & AD7678_sdout;
        else
            bitCntr <= bitCntr + 1;
            tempReg <= tempReg(tempReg'left-1 downto 0) & AD7678_sdout;
        end if;
    end if;
end if;
end process;
veriOkundu <= yeniVeri and cnvst;

process(clk)
begin
if clk'event and clk = '1' then

```

```

        cnvstReg <= cnvstReg(0) & cnvst;
    end if;
end process;
cnvstFall <= '1' when cnvstReg = "10" else '0';

process(clk)
begin
    if clk'event and clk = '1' then
        if cnvstFall = '1' then
            cntrRst <= '0';
        elsif cntrDoldu = '1' then
            cntrRst <= '1';
        end if;
    end if;
end process;

process(clk)
begin
    if clk'event and clk = '1' then
        if cntrRst = '1' then
            cntr <= 0;
            cntrDoldu <= '0';
        elsif cntr = 25 then
            cntrDoldu <= '1';
        else
            cntr <= cntr + 1;
        end if;
    end if;
end process;

process(clk)
begin
    if clk'event and clk = '1' then
        if cnvst = '1' then
            delayCntr <= 0;
            cnvstdelayed <= '1';
        else
            if delayCntr = delayCntrLim-1 then
                cnvstdelayed <= '0';
            else
                delayCntr <= delayCntr + 1;
            end if;
        end if;
    end if;
end process;

end Behavioral;

```

Ek 3: Kontrolcünün VHDL kodları.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity DAC_Controller is
  Port ( clk      : in std_logic;
        yeniADCData : in std_logic;
        yeniPCData : in std_logic;
        yeniVeriOut : out std_logic;
        DACKomut   : in std_logic_vector (17 downto 0);
        ADCDataIn  : in std_logic_vector (15 downto 0);
        ADCLimHigh : in std_logic_vector (15 downto 0);
        ADCLimLow  : in std_logic_vector (15 downto 0);
        DACDataOut : out std_logic_vector (17 downto 0)
        );
end DAC_Controller;

architecture Behavioral of DAC_Controller is

  signal iDACDataOut : std_logic_vector(17 downto 0) := (others => '0');

begin

  process(clk)
  begin
    if clk'event and clk = '1' then
      yeniVeriOut <= '0';
      if yeniPCData = '1' then
        iDACDataOut <= DACKomut;
      elsif yeniADCData = '1' then
        yeniVeriOut <= '1';
        if ADCDataIn < ADCLimLow then
          if iDACDataOut /= ("11" & x"FFFF") then
            iDACDataOut <= iDACDataOut + 1;
          end if;
        elsif ADCDataIn > ADCLimHigh then
          if iDACDataOut /= ("00" & x"0000") then
            iDACDataOut <= iDACDataOut - 1;
          end if;
        end if;
      end if;
    end if;
  end process;
  DACDataOut <= iDACDataOut;

end Behavioral;
```

Ek 4: Bilgisayar arayüzü VHDL kodları.

Alınan veri için;

```
library ieee;
use ieee.std_logic_1164.all;

entity rxUnit is
  generic ( oscClkFreq : real := 25.0e6; -- Hz
           baudRate   : real := 460800.0; -- Hz
           sampleRate : integer := 6);
  port ( clk      : in std_logic; -- system clock signal
        reset    : in std_logic; -- uart reset
        RX       : in std_logic; -- uart data input
        yeniVeri : out std_logic; -- Byte available
        dataOut  : out std_logic_vector(7 downto 0)); -- Byte received
end rxUnit;

architecture Behaviour of rxUnit is

  constant clkCntrLim : integer := integer(oscClkFreq/(baudRate*real(sampleRate)));
  constant baudAyarlanan : integer := integer(oscClkFreq/(real(clkCntrLim)*real(sampleRate)));
  constant baudHata: integer := integer (abs((baudRate/real(baudAyarlanan))-1.0)*1000.0);

  -- integer
  signal clkCntr : integer range 0 to clkCntrLim := 0;

  signal receiveReg : std_logic_vector(7 downto 0); -- receive register

  signal clkEn      : std_logic := '0';
  signal veriHazir  : std_logic := '0';

  begin

  assert false report "FPGA Gercek BaudRate " & integer'image(baudAyarlanan) severity warning;
  assert false report "FPGA BaudRate Hatası Binde " & integer'image(baudHata) severity warning;

  process(clk)
  begin
  if clk'event and clk = '1' then
    if clkCntr = clkCntrLim - 1 then
      clkCntr <= 0;
    else
      clkCntr <= clkCntr + 1;
    end if;
  end if;
  end process;
  clkEn <= '1' when clkCntr = clkCntrLim - 1 else '0';

  process(clk)
  variable BitPos      : integer range 0 to 10 := 0;      -- Position of the bit in the frame
  variable sampleCntr  : integer range 0 to sampleRate-1 := 0; -- sample counter
```

```

variable sampleCntrEn : std_logic := '0'; -- Count from 0 to 3 in each bit
begin
if clk'event and clk = '1' then
    if reset = '1' then
        veriHazir <= '0';
        dataOut <= x"00";
        sampleCntr := 0;
    BitPos := 0;
        sampleCntrEn := '0';
    elsif clkEn = '1' then
        case BitPos is
            when 0 => -- idle
                veriHazir <= '0';
                sampleCntrEn := '0';
        if RX = '0' then -- Start Bit
            sampleCntrEn := '1';
            BitPos := 1;
        end if;
        when 10 => -- Stop Bit
            BitPos := 0; -- next is idle
            veriHazir <= '1'; -- Indicate byte received
            dataOut <= receiveReg; -- Store received byte
                sampleCntrEn := '0';
            when 1 =>
                if sampleCntr = sampleRate-1 then -- Increment BitPos
                    BitPos := BitPos + 1;
                end if;
        when others =>
            if (sampleCntr = (sampleRate-1)/2) then -- Sample RX
                receiveReg(BitPos-2) <= RX; -- Deserialisation
            end if;
            if sampleCntr = sampleRate-1 then -- Increment BitPos
                BitPos := BitPos + 1;
            end if;
        end case;
        if sampleCntr = sampleRate-1 or sampleCntrEn = '0' then
            sampleCntr := 0;
        else
            sampleCntr := sampleCntr + 1;
        end if;
    end if;
end if;
end process;

```

```

yeniVeri <= veriHazir and clkEn;
end Behaviour;

```

Giden veri için;

```

library ieee;
use ieee.std_logic_1164.all;

```

```

entity PCArayuzu is
  generic (
    clkFreq : integer := 50_000_000
  );
  Port (
    clk : in std_logic;
    -- DAC data read
    DACData1 : in std_logic_vector (17 downto 0);
    DACData2 : in std_logic_vector (17 downto 0);
    DACData3 : in std_logic_vector (17 downto 0);
    DACData4 : in std_logic_vector (17 downto 0);
    DACData5 : in std_logic_vector (17 downto 0);
    DACData6 : in std_logic_vector (17 downto 0);
    -- ADC data read
    ADCData1 : in std_logic_vector (15 downto 0);
    ADCData2 : in std_logic_vector (15 downto 0);
    ADCData3 : in std_logic_vector (15 downto 0);
    ADCData4 : in std_logic_vector (15 downto 0);
    ADCData5 : in std_logic_vector (15 downto 0);
    ADCData6 : in std_logic_vector (15 downto 0);

    -- DAC controller aktivatörleri
    yeniKomut1 : out std_logic;
    yeniKomut2 : out std_logic;
    yeniKomut3 : out std_logic;
    yeniKomut4 : out std_logic;
    yeniKomut5 : out std_logic;
    yeniKomut6 : out std_logic;

    -- yazılacak DAC verileri
    DAC1 : out std_logic_vector (17 downto 0);
    DAC2 : out std_logic_vector (17 downto 0);
    DAC3 : out std_logic_vector (17 downto 0);
    DAC4 : out std_logic_vector (17 downto 0);
    DAC5 : out std_logic_vector (17 downto 0);
    DAC6 : out std_logic_vector (17 downto 0);

    -- DAC controllere girecek limit değerler
    ADC1_high : out std_logic_vector (15 downto 0);
    ADC2_high : out std_logic_vector (15 downto 0);
    ADC3_high : out std_logic_vector (15 downto 0);
    ADC4_high : out std_logic_vector (15 downto 0);
    ADC5_high : out std_logic_vector (15 downto 0);
    ADC6_high : out std_logic_vector (15 downto 0);

    ADC1_low : out std_logic_vector (15 downto 0);
    ADC2_low : out std_logic_vector (15 downto 0);
    ADC3_low : out std_logic_vector (15 downto 0);
    ADC4_low : out std_logic_vector (15 downto 0);
    ADC5_low : out std_logic_vector (15 downto 0);
    ADC6_low : out std_logic_vector (15 downto 0);

    --

```



```

--          kanal : in std_logic_vector (7 downto 0);
            TX    : out std_logic ;
            RX    : in std_logic
--          timeOut : in std_logic
            );
end PCArayuzu;

architecture Behavioral of PCArayuzu is

constant baudRate    : integer := 115_200;
constant timeOutCntrLim : integer := clkFreq/2;
constant baudCntrLim  : integer := integer(real(clkFreq)/real(baudRate));

constant rzv  : std_logic_vector(15 downto 0) := x"aaaa";
constant z6   : std_logic_vector(5  downto 0) := "000000";
constant header : std_logic_vector(15 downto 0) := x"cdab";

signal sendReg : std_logic_vector(279 downto 0) := (others => '0');
signal crc     : std_logic_vector(15  downto 0) := (others => '0');
constant CRClnit : std_logic_vector(15  downto 0) := x"0000";

signal bitCntr : integer range 0 to 15 := 0;
signal byteCntr : integer range 0 to sendReg'length/8 := 0;
signal baudCntr : integer range 0 to baudCntrLim := 0;
signal timeOutCntr : integer range 0 to timeOutCntrLim := 0;

signal baudEn  : std_logic := '0';
signal timeOut : std_logic := '0';

-- receive signals
signal rxReg  : std_logic_vector( 3 downto 0) := (others => '1');
signal rxSync : std_logic := '1';
signal yeniByte : std_logic := '0';
signal rxData  : std_logic_vector( 7 downto 0) := (others => '0');
signal gelenCRC : std_logic := '0';
signal recReg  : std_logic_vector(375 downto 0) := (others => '0');

signal kanal : std_logic_vector (7 downto 0):= (others => '0');

type durumlar is (BOS, CRCKONTROL);
    signal durum : durumlar := BOS;

-- output signal definitions
signal yeniKomut1_t : std_logic := '0';
signal yeniKomut2_t : std_logic := '0';
signal yeniKomut3_t : std_logic := '0';
signal yeniKomut4_t : std_logic := '0';
signal yeniKomut5_t : std_logic := '0';
signal yeniKomut6_t : std_logic := '0';

COMPONENT rxUnit
    generic ( oscClkFreq : real := 25_000_000.0; -- Hz

```

```

    baudRate : real := 460_800.0; -- Hz
    sampleRate : integer := 6);
    PORT(
        clk : in std_logic;
        reset : in std_logic;
        RX : in std_logic;
        yeniVeri : out std_logic;
        dataOut : out std_logic_vector(7 downto 0)
    );
END COMPONENT;

-----
-- BEGIN -----
-----

begin

kanal <= "00" & yeniKomut6_t & yeniKomut5_t & yeniKomut4_t & yeniKomut3_t & yeniKomut2_t
& yeniKomut1_t;

U0: rxUnit
    generic map(
        oscClkFreq => real(clkFreq),
        baudRate => real(baudRate),
        sampleRate => 7)
    port map(
        clk => clk,
        reset => '0',
        RX => rxSync,
        yeniVeri => yeniByte,
        dataOut => rxData
    );

-----
-- 10 ms COUNTER --
-----
process(clk)
begin
if clk'event and clk = '1' then
    if timeOutCntr = timeOutCntrLim-1 then
        timeOutCntr <= 0;
    else
        timeOutCntr <= timeOutCntr + 1;
    end if;
end if;
end process;
timeOut <= '1' when timeOutCntr = 0 else '0';

-----
-- RX SYNCRONIZATION PROCESS --
-----
process(clk)
begin

```

```

if clk'event and clk = '1' then
    rxReg <= rxReg(rxReg'left-1 downto 0) & rx;
end if;
end process;
rxSync <= rxReg(rxReg'left);

-----
-- TRANSMIT PROCESS --
-----

process(clk)
begin
if clk'event and clk = '1' then
    if timeOut = '1' then
        bitCntr <= 0;
        byteCntr <= 0;
        sendReg <= x"BBCC" & ADCData6 & z6 & DACData6 & ADCData5 & z6 & DACData5
& ADCData4 & z6 & DACData4 &
        ADCData3 & z6 & DACData3 & ADCData2 & z6 & DACData2 & ADCData1 & z6 &
DACData1 & kanal & header;
        tx <= '1';
    elsif baudEn = '1' then
        if byteCntr = sendReg'length/8 then      -- bayt sayac kontrol
            tx <= '1';
        else
            if bitCntr = 0 then
                tx <= '0';
                bitCntr <= bitCntr + 1;
            elsif bitCntr = 9 then
                tx <= '1';
                bitCntr <= bitCntr + 1;
                byteCntr <= byteCntr + 1;
            elsif bitCntr = 10 then
                tx <= '1';
                bitCntr <= 0;
            else
                tx <= sendReg(0);
                sendReg <= '1' & sendReg(sendReg'left downto 1);
                bitCntr <= bitCntr + 1;
            end if;
        end if;
    end if;
end if;
end process;

-----
-- BAUD RATE PROCESS --
-----

process(clk)
begin
if clk'event and clk = '1' then
    if baudCntr = baudCntrLim-1 then
        baudCntr <= 0;
    else

```

```

        baudCntr <= baudCntr + 1;
    end if;
end if;
end process;
baudEn <= '1' when baudCntr = 0 else '0';

-- receive interface
process(clk)
begin
if clk'event and clk = '1' then
    if yeniByte = '1' then
        recReg <= recReg(recReg'left-8 downto 0) & rxData;
    end if;
end if;
end process;

process(clk)
begin
if clk'event and clk = '1' then
    case durum is
        when BOS =>
            yeniKomut1_t <= '0';
            yeniKomut2_t <= '0';
            yeniKomut3_t <= '0';
            yeniKomut4_t <= '0';
            yeniKomut5_t <= '0';
            yeniKomut6_t <= '0';

            if yeniByte = '1' then
                durum <= CRCKONTROL;
            end if;

        when CRCKONTROL =>
            durum <= BOS;
            if gelenCRC = '1' then
                if recReg(375 downto 360) = x"ABCD" then

                    yeniKomut6_t <= recReg(357);
                    yeniKomut5_t <= recReg(356);
                    yeniKomut4_t <= recReg(355);
                    yeniKomut3_t <= recReg(354);
                    yeniKomut2_t <= recReg(353);
                    yeniKomut1_t <= recReg(352);

                    DAC1 <= recReg(345 downto 328);
                    ADC1_High <= recReg(327 downto 312);
                    ADC1_Low <= recReg(311 downto 296);
                    DAC2 <= recReg(289 downto 272);
                    ADC2_High <= recReg(271 downto 256);
                    ADC2_Low <= recReg(255 downto 240);
                    DAC3 <= recReg(233 downto 216);
                    ADC3_High <= recReg(215 downto 200);
                    ADC3_Low <= recReg(199 downto 184);
                end if;
            end if;
        end case;
    end if;
end process;

```

```

DAC4          <= recReg(177 downto 160);
ADC4_High <= recReg(159 downto 144);
ADC4_Low  <= recReg(143 downto 128);
DAC5          <= recReg(121 downto 104);
ADC5_High <= recReg(103 downto 88);
ADC5_Low  <= recReg(87 downto 72);
DAC6          <= recReg(65  downto 48);
ADC6_High <= recReg(47  downto 32);
ADC6_Low  <= recReg(31  downto 16);

                                end if;
                        end case;
end if;
end process;

-- output signal assignments
yeniKomut1 <= yeniKomut1_t;
yeniKomut2 <= yeniKomut2_t;
yeniKomut3 <= yeniKomut3_t;
yeniKomut4 <= yeniKomut4_t;
yeniKomut5 <= yeniKomut5_t;
yeniKomut6 <= yeniKomut6_t;

end Behavioral;

```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : Ufuk Yumrukaya
Uyruğu : T.C.
Doğum Tarihi ve Yeri : 09.08.1983 – Manisa
Telefon : 0 (536) 604 13 30
E-posta : ufukyumrukaya@gmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Lisans	Hacettepe Üni. / Elektrik-Elektronik Müh.	2007
Yüksek Lisans	TOBB ETU / Elektrik-Elektronik Müh.	2015

İş Deneyimi

Yıl	Yer	Görev
2007-2010	Vestel Elektronik A.Ş.	Uzman Sistem Tasarım Müh.
2010-.....	Roketsan A.Ş.	Uzman Müh.

Yabancı Dil

İngilizce

Yayınlar

- O. Çakır, M. Engin, E. Z. Engin, U. Yumrukaya, “Investigation of Muscle Fatigue by Processing EMG Signals”, 14th National Biomedical Engineering Meeting (BIYOMUT 2009), 20-22 May 2009, İzmir, Türkiye
- U. Yumrukaya, A. Bozbey, “Development of a multi-channel, stable, low-cost, low-noise, high precision, wide range current source for superconducting integrated circuits” 4th International Conference on Superconductivity and Magnetism (ICSM 2014), 27 April – 02 May, Antalya, Turkey.

- U. Yumrukaya, A. Bozbey, “Development of a multi-channel, stable, low-cost, low-noise, high precision, wide range current source for superconducting integrated circuits” 9th International Conference on Magnetic and Superconducting Materials (MSM15), 30 April – 03 May, Antalya, Turkey.