



KLİMA SİSTEMLERİNİN SIRALI VE YEDEKLİ BİR ŞEKİLDE ÇALIŞMASININ MODELSİM PROGRAMI KULLANARAK GELİŞTİRİLMESİ

Yazılım Mühendisliği Ana Bilim Dalı

Tezsiz Yüksek Lisans Bitirme Projesi

Ergin ÇALIŞKAN

Proje Danışmanı: Doç. Dr. Aytuğ ONAN

Haziran 2023

İzmir Kâtip Çelebi Üniversitesi Fen Bilimleri Enstitüsü Yazılım Mühendisliği A.B.D. öğrencisi **Ergin ÇALIŞKAN** tarafından hazırlanan **Klima Sistemlerinin Sıralı ve Yedekli Bir Şekilde Çalışmasının Modelsim Programı Kullanarak Geliştirilmesi** başlıklı bu çalışma tarafımda okunmuş olup, yapılan inceleme sonucunda kapsam ve nitelik açısından başarılı bulunarak tarafımdan **YÜKSEK LİSANS BİTİRME PROJESİ** olarak kabul edilmiştir.

ONAYLAYAN:

Proje Danışmanı: Doç. Dr. Aytuğ ONAN
İzmir Kâtip Çelebi Üniversitesi

Proje alıřmasına katkılarından dolayı danıřmanım Sayın Doent Doktor
Aytuđ ONAN'a teřekkürlerimi sunarım.

KLİMA SİSTEMLERİNİN SIRALI VE YEDEKLİ BİR ŞEKİLDE ÇALIŞMASININ MODELSİM PROGRAMI KULLANARAK GELİŞTİRİLMESİ

ÖZ

Soğuk Hava Deposu olarak kullanılan alanları soğutmak için çalıştıracağımız klima sistemlerinin sıralı ve yedekli bir şekilde çalıştırmayı, arıza durumunda kısa süre içerisinde kullanıcıya arıza hakkında detay vermesi amaçlanan sistemdir. Sıralı ve yedekli olarak çalıştırılacak olan klima sistemi tek klimaya bağımlılığı ortadan kaldıracak, malzeme yorgunluğundan kaynaklanan arızaları ortadan kaldıracak ve tek klimanın arızalanması durumunda depoda tuttuğumuz malzemelerin bozulma, hasar görme vs. durumların önüne geçilebilecektir.

Tasarımda 3 adet klimadan 1 ve 2 no.lu klimalar belirlenen süreler içerisinde sıralı bir şekilde çalışacaktır. 1 no.lu klima aktif olarak ON konumundayken 2 no.lu klima belirlenen süre sonunda devreye girmek üzere STANBY durumunda bekleyecektir. 3.no.lu klimada 1 veya 2 no.lu klima ON konumundayken yaşanabilecek bir FAULT ve 1 veya 2 no.lu klimanın STANBY konumundayken FAULT durumu düşmesi durumunda devreye girmek üzere STANBY konumunda bekleyecektir. 3 adet klimadan 2 adetinin arızalanması durumunda 1 adet klimayla da sistem FAULT alana kadar çalışmaya devam edecektir. 1, 2 ve 3 no.lu klimaların ELECTRIC, GAS, OUTER UNIT bölümlerinden gelecek bilgilere göre hızlı bir şekilde arızaya müdahale edilecek ve sistemin faaliyeti sağlanacaktır.

Keywords: ModelSim, Klima Sistemleri, Simülasyon Görseli, Tasarım, Arıza

İçindekiler

Teşekkür	ii
Öz	iii
İçindekiler	iv
1 VHDL - Donanım Tanımlama Dili	1
1.1 Giriş.....	6
1.2 Donanım Tanımlama Dili-HDL.....	7
1.2.1. Davranışsal Modelleme	7
1.2.2. Yapısal Modelleme	7
1.2.3. Register Transfer Level (RTL)	8
1.3. VHDL Tasarım Akışı	9
1.4. VHDL Yazım Kuralları	10
1.5. VHDL Temel Bölümleri	11
1.5.1. Entity.....	11
1.5.2. Mimari (Architecture).....	11
1.5.3. Paket (Package).....	11
1.5.4. Bileşen (Komponent).....	12
1.5.5. İşlem (Process).....	12
1.6. Veri Nesneleri.....	12
1.6.1. Sinyal (Signal)	12
1.6.2. Değişken (Variable).....	13
1.6.3. Sabit (Constant)	13
2 Klima Sistemleri.....	14
2.1. Giriş	14
2.2. Klima Kontrol Otomasyonu ve Eş Yaşlandırma	14
2.3. Klima Seçiminde Dikkat Edilmesi Gerekenler	15

2.3.1. Sađlıklı Voltaj	15
2.3.2. Dođru Montaj	15
2.3.3. Bilinçli Kullanım	16
2.3.4. Zamanında Dođru Bakım.....	16
2.4. Klima Kontrol Sisteminin Tasarımı	16
3 Tasarım Kodları	18
4 Simülasyon Görseli.....	31
Kaynakça	35
Özgeçmiş	36

1.VHDL - DONANIM TANIMLAMA DİLİ

1.1 Giriş

VHDL' in ingilizce kökeni Very High Speed Integrated Circuit Hardware Description Language' den gelir. Bu dil aynı zamanda çok yüksek hızlı tümleşik devre donanım tanımlama dili olarak da tabir edilebilir. HDL dillerinden en çok kullanılanlardan biridir. Bu programlama dili 1980' lerden bu yana süregelmiş ve devamlı geliştirilmiştir. IEEE topluluğu tarafından da bir standart olarak kabul edilmiştir.

HDL donanım tanımlama dillerinden birisi olan VHDL başlıca sentezleme ve simülasyon için kullanılır.

- **Sentezleme:** Kodun FPGA'e yüklenmeden önceki halini kurgular,
- **Simülasyon:** FPGA'e yüklenecek kodun simülasyonunu gerçekleştirme, library IEEE;

```
use IEEE.STD_LOGIC_1164.ALL;
```

```
entity Ve_kapisi is
```

```
Port ( x : in bit;
```

```
y : in bit;
```

```
z : out bit);
```

```
end Ve_kapisi ;
```

```
architecture Behavioral of Ve_kapisi is
```

```
begin
```

```
z <= x AND y;
```

```
end Behavioral;
```

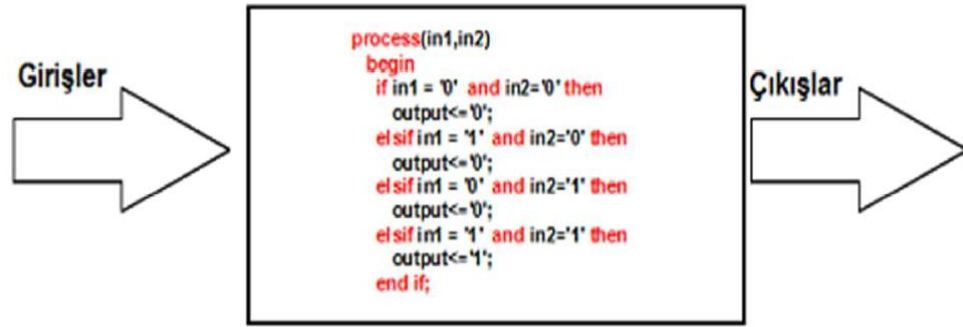
Yukarıdaki örnekteki yazılımda 2 girişli bir çıkışlı ve kapısı tanımlanır.

1.2. Donanım Tanımlama Dili-HDL

HDL, herhangi bir donanım kısmını yapılandırmada tercih edilen bir programlama dilidir. HDL, yazılım kullanılarak donanımları modelleme ve onların davranışlarını belirleme durumu sağlamaktadır. VHDL ve Verilog dilleri, FPGA donanımının programlanmasında en çok kullanılan donanım tanımlama dillerindedir.

1.2.1. Davranışsal Modelleme

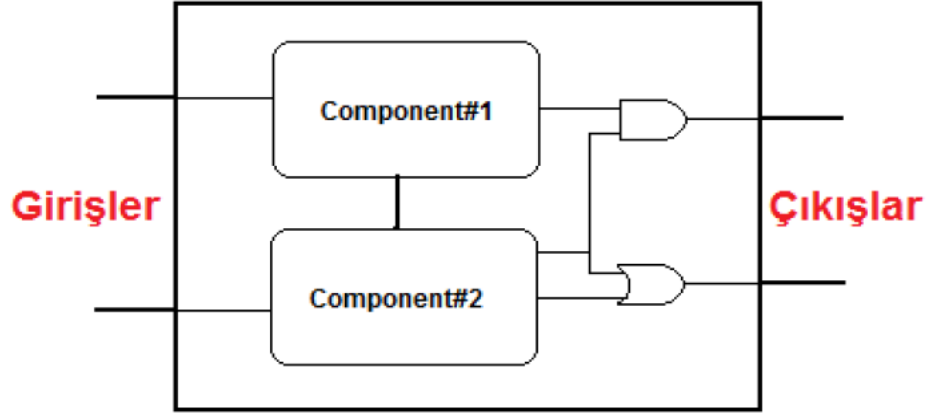
Davranışsal modelleme sistemin dış dünya ile alakalı davranışını belirleyerek Şekil 1.1.' deki gibi giriş ve çıkışa müdahale eder. Modelin giriş ve çıkış tepkimeleri davranışsal olarak tanımlanır. Modelin iç yapısına ve durumuna bakılmaz. Burada modelin işlevi ve fonksiyonu irdelenmektedir.



Şekil 1.1. Davranışsal Model

1.2.2. Yapısal Modelleme

Alt seviyelerdeki bileşenler ile arasındaki alakaları göstermektedir. Yapısal modelleme Şekil 1.2.' deki gibi davranışsal modelin gövdesinin yapılandırılmasıdır. VHDL programcıya yapısal modelleme için olanak sağlamaktadır. Bu modelleme çok karmaşık ve büyük donanım parçalarında kolaylık sağlamaktadır. VHDL tasarımlarında genellikle tercih edilen teknik, birbirinden farklı modellenen alt kısımdaki modüllerin structural modelleme ile oluşturulup; bu modüllerin üst kısımdaki yapısal modelleme ile birbirine bağlanması tekniğidir.

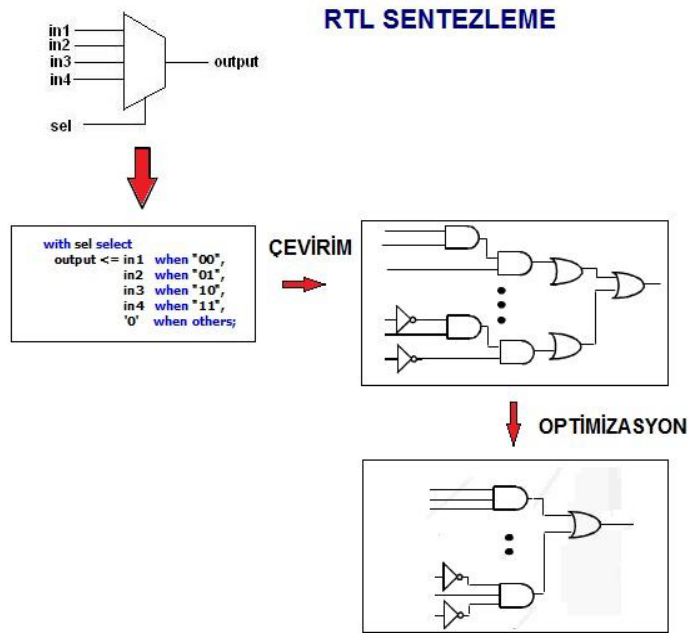


Şekil 1.2. Yapısal Model

1.2.3. Register Transfer Level (RTL)

RTL yöntemi, bir soyutlanmadır. Soyutlanma yazılımının analiz amacı gözeterek tercih edilir. Bu yöntem, tasarım için kodun register cinsinden tabir edilmesidir. VHDL koduna karşılık düşen yapıların lojik elementlerinden oluşan kısımdır.

Şekil 1.3.' te de görüldüğü gibi RTL modeli için çevrim ve optimizasyon adımları işlenir. RTL analizde VHDL kodu ilk başta çevrim işlemi ile dijital bir bloğa dönüştürülür. Ardından da iyileştirme işlemiyle VHDL kodunun karşılığı olan mantık kapıları kısmı iyileştirme edilerek, FPGA donanımının efektif bir biçimde kullanılması sağlanır.



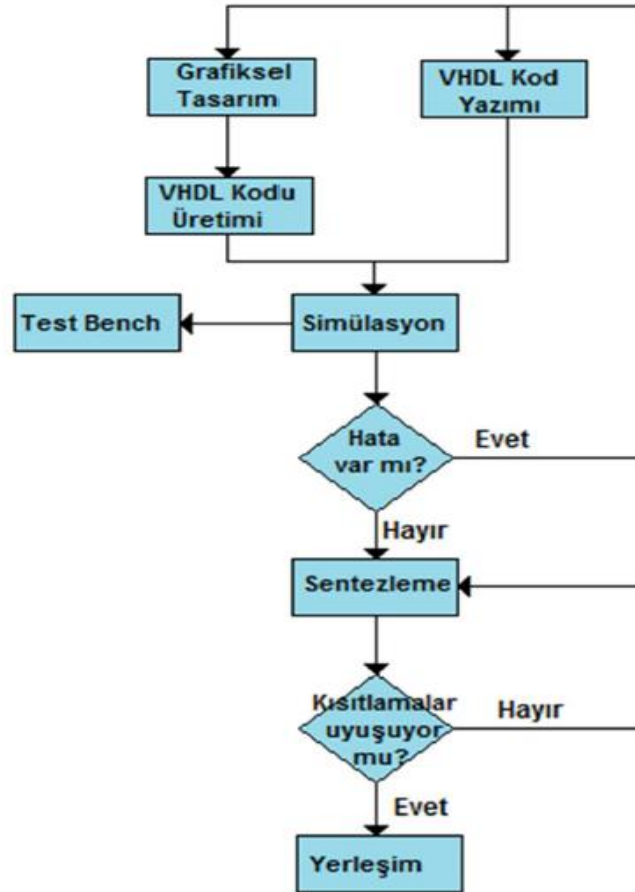
Şekil 1.3. RTL Modeli

Yukarıdaki şekilde 4 girişli tek çıkışlı bir MUX' un (veri seçici) RTL modeli görülmektedir. Tasarım VHDL olarak yazıldıktan sonra, compiler ile sentezlenir. Sentezlenen bu kısmın karşılığı olan mantık kapılarına dönüştürülür. Optimizeden sonra RTL akışı tamamlanır.

1.3. VHDL Tasarım Akışı

VHDL'de tasarım akışı sırasıyla aşağıdaki üç maddeden oluşmaktadır. Algoritması Şekil 1.4.' te görülen gibi tasarım akışının temel üç maddesi;

- Kodlama
- Simülasyon
- Sentezleme



Şekil 1.4. Tasarım Akışı Gösterimi

Kodlama: Bu kısım, programın VHDL kodunun yazıldığı kısımdır.

Simülasyon: Bu kısımda oluşturulan VHDL kodunun simülasyonu yapılır, programın doğruluğu gözlemlenir. Simülasyon sırasında ortaya çıkan hatalar, program FPGA' e yüklenmeden önce doğrulanmaktadır.

Sentezleme: Bu kısımda yazılan VHDL kod, donanım diline çevrilip RTL modeli çıkartılır. Sentezlemeden sonra kod derleyici tarafından FPGA'e yüklenmeye hazırlanacak konfigürasyon dosyasına çevrilir. Sentezleme işlemi genel olarak derleyici tarafından yürütülmektedir. Tasarımcı tarafından da sentezlenebilir fakat kullanışlı bir yöntem asla değildir.

1.4. VHDL Yazım Kuralları

- VHDL kod grubu iki kısımdan oluşur.

Sentezlenebilen Kodlar; FPGA' e yüklenebilir kodlardır. Fakat VHDL kodların tamamı FPGA' e yüklenebilir değildirler.

Sentezlenemeyen Kodlar: Yalnızca test amaçlıdır. Sentezlenemez ve dolayısıyla FPGA'e yüklenemezler.

- VHDL' de kendine özgü olan kodları vardır. Örneğin case, while, if, with, and, vb... VHDL' de ifadeler eş zamanlı(paralel) ve ardışık(sıralı) olarak ikiye ayrılır.

Eş zamanlı ifadeler; Aynı anda işlenirler.

Ardışık ifadeler; diğer dillerde de olduğu gibi sıraya göre işlenirler.

- VHDL komut satırların sonuna noktalı virgül (;) konur.

`C<=5 D<=6;`

” <=” sembolü küçük eşittir anlamından farklı olarak sinyal atama işine yarar.

Sağdaki değer soldakine atanmasına karşılık gelmektedir.

- VHDL kodlarında harf duyarlılığı yoktur.

`Type= type=TypeE`

• VHDL kodunda boşlukların anlamı yoktur ve düzeni etkilemez fakat koddaki boşlukların düzenli bırakılması karmaşık kodları anlaşılır ve okunur hale getirir.

`K<= M and N;` ile `K <= M and N;` aynı anlamdadır.

- Kod kısmında herhangi bir açıklamanın yazılmak istenirse başına "--" ile başlanarak yazılmaya müsait hale getirilir. Başına tire gelen ifadeler koda dahil olmazlar.

$K \leq L + M$; -- K çıkışına L ve M girişlerinin toplamını at

1.5. VHDL Temel Bölümleri

Bir VHDL tasarımı başlıca 5 ana bölümden oluşmaktadır.

- Entity (Entity)
- Mimari (Architecture)
- Paket (Package)
- Bileşen (Component)
- İşlem (Process)

1.5.1. Entity

Bir tasarımın en temel bloğudur. Verilen bir mantık fonksiyon için bütün giriş ve çıkışları yani mantık fonksiyonun dış dünya ile bağlantısını tanımlar. Her VHDL tasarım mutlaka en az bir entity içerir. Port tanımlamaları farklı biçimlerde olabilir. Burada port tanımlaması ile işaretin giriş/çıkış olduğu ve veri tipi belirtilir. Entity, tasarım ile onun dışındaki ara yüzü tanımlamaya yardımcı olmaktadır. Bu bölümde her türlü I/O portlarının tanımlanması yapılır.

1.5.2. Mimari (Architecture)

Tasarımın ne iş yaptığının belirlendiği kısımdır. Mimari entity' nin davranışını tanımlar. Bir entity birden fazla mimariye sahip olabilir. Bir mimari davranışsal (behavioral) tanımlama, yapısal (structural) tanımlama, veri akışı (data flow) olmak üzere üç farklı biçimde kullanılabilir.

1.5.3. Paket (Package)

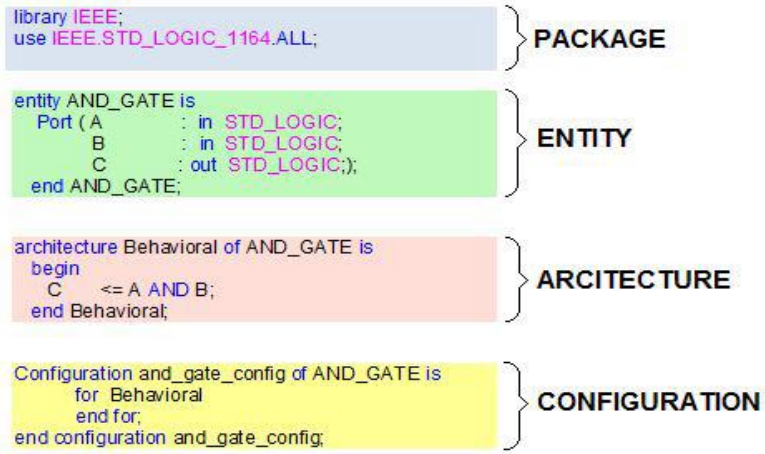
Paket; entity üniteleri tarafından kullanılan tanımlamaları bir grup haline getirir ve paylaşır. Package ayrıca istenen yapıları farklı tasarımlarda hazır tutmak üzere tasnifleyen kısımdır.

1.5.4. Bileşen (Komponent)

Bileşen yapısı; devre tanımlamasında bir alt devre gibi kullanılan bileşenin adını ve ara yüzünü tanımlar.

1.5.5. İşlem (Process)

İşlem bloğu sıralı şekilde gerçekleştirilecek durumları içerir. Bir mimaride birden fazla işlem bloğu anlık gerçekleştirilir. Yani işlem blokları aynı anda başlar ve her bir işlem bloğu kendi içinde satır satır sıralı olarak gerçekleştirilir.



1.6. Veri Nesneleri

VHDL nesne özelliklere sahiptir. Nesnelerin davranış ve işlevlerine göre fonksiyonellik kazanır.

1.6.1. Sinyal (Signal)

Sinyaller entity, mimari ve paket içinde kullanılabilir. Sinyallerin paket içinde kullanımı çok önemlidir. Çünkü paket genel bir ifade olduğundan dolayı işaretler burada kullanıldığı zaman diğer entity veya mimariler tarafından çağrılarak kullanılabilir. Sinyallere başlangıç değeri atanabilir ve işlem gerçekleşirken bu ifade yenilenebilir.

signal sinyalin_ismi: sinyal_tipi :=değeri;

1.6.2. Değişken (Variable)

Genel olarak geçici değerleri kullanmak için tercih edilmektedir. Böylece programın daha hızlı çalışması sağlanabilir. Ortak değişken (shared variable) türü de bulunmaktadır. Eğer bir değişken birden fazla process' de ve alt programda çağrılacak ise ortak değişken olarak tanımlanma yapılmalıdır.

variable değişkenin_adı: değişkenin tipi := değeri;

1.6.3. Sabit (Constant)

Değeri bilinen ve değişmeyen verilere ihtiyaç duyulursa sabitler kullanılır. Bu durum, tasarımın daha iyi gözlemlenebilmesi ve anlaşılabilmesi için önemli bir faktördür. Özellikle tasarım içinde sık sık kullanılan değeri aynı olan ifadeler için vazgeçilmez bir veri tipidir.

constant sabitin_adı: sabitin tipi := değeri;

2.KLİMA SİSTEMLERİ

2.1. Giriş

İklimlendirme en genel halde, kapalı bir ortamın konfor amacıyla insan, hayvan ve bitkilerin veya endüstriyel bir mamulün üretilmesi ve korunması sırasında atmosferik şartların otomatik olarak kontrol altında tutulması işlemi olarak ifade edilir. Pratikte ise iklimlendirme, havanın ısıtılması, soğutulması, nemlendirilmesi veya neminin alınması bir başka deyişle ortamdaki havanın şartlandırılması işlemine denir.

2.2. Klima Kontrol Otomasyonu ve Eş Yaşlandırma

Sistem odaları, Soğuk Hava Deposu gibi değerli ortamların iklimlendirilmesi (soğutulması) amacıyla kullanılan klimaların kapasite seçimleri doğru yapılmalıdır. Yani, ısı kaynakları doğru hesaplanıp, odanın hacmi ile beraber uygun bir soğutma kapasite değeri seçilmelidir. Oda uygun kapasiteli bir adet klima ile rahat bir şekilde soğutulabilmelidir. Ayrıca herhangi bir acil eylem durumunda ortamda asıl klima ile beraber aynı kapasitede yedek klimalarda konumlandırılmalıdır ki, oda içindeki değerli cihazlarınız ürünleriniz güvence altına alınsın.

Klimalar isteğe bağlı olarak 1, 2, 3...12 saat ara ile dinlenmeli çalıştırabilir, kapasite yetersizliğinde ya da asıl klimanın arızalanması durumunda ya da bir nedenle ortam sıcaklığı ayarlanmış bir değerin üzerine çıktığında çalışmayan klimayı otomatik olarak devreye alınabilsin.

Klimaların sıcaklık ve zaman bağımlı olarak otomatik olarak sırayla devreye alınabilmesi sayesinde klimaların bakım maliyetlerini azaltabilir ve önemli elektrik tasarrufu sağlanabilir.

2.3. Klima Seçiminde Dikkat Edilmesi Gerekenler

Daha doğru ve gerçekçi bir kapasite tayini için, ısı kazancını etkileyen tüm fonksiyonların göz önünde bulundurulduğu bir hesap yapmak zorunludur. İklimlendirilen ortama uygun cihaz kapasitesi belirlendikten sonra hava sirkülasyonunun iyi bir şekilde sağlanabilmesi amacıyla cihaz modeli doğru seçilmelidir.

Doğru kapasitede cihaz seçimi yapılmayan yerlerde sorunlar yaşanmaktadır. Kapasite yüksek seçilirse, cihaz sık aralıklarla termostata girerek ısıtma ve soğutmada problem yaşanır. Kapasite düşük seçilirse, cihaz sürekli çalışarak kompresörün zorlanmasına neden olur. Bu da kompresörün verimini ve çalışma ömrünü etkiler.

2.3.1. Sağlıklı Voltaj

Cihazın sağlıklı olarak çalışabilmesi için şebeke voltajı, teknik kataloglarda belirtilen limitler dahilinde olmalı, aksi durumda şebeke voltajını düzelten aletler kullanılmalıdır. Yüksek ve düşük voltaj cihaz için sakıncalıdır. Düşük voltajda çalıştırılan bir klima cihazının kompresörü arızalanabilir veya cihaz devreye giremez. Yüksek voltajda ise cihazda bulunan yüksek voltaj koruma elemanı ve transformatör arızalanabilir.

2.3.2. Doğru Montaj

Cihazın sağlıklı olarak çalışabilmesi için montaj kurallarına ve teknik kataloglarda belirtilen limitlere bağlı kalarak klima cihazı monte edilmelidir. Hatalı yapılan montajlar klima cihazının çalışma verimi ve ömrü ile direk ilgilidir.

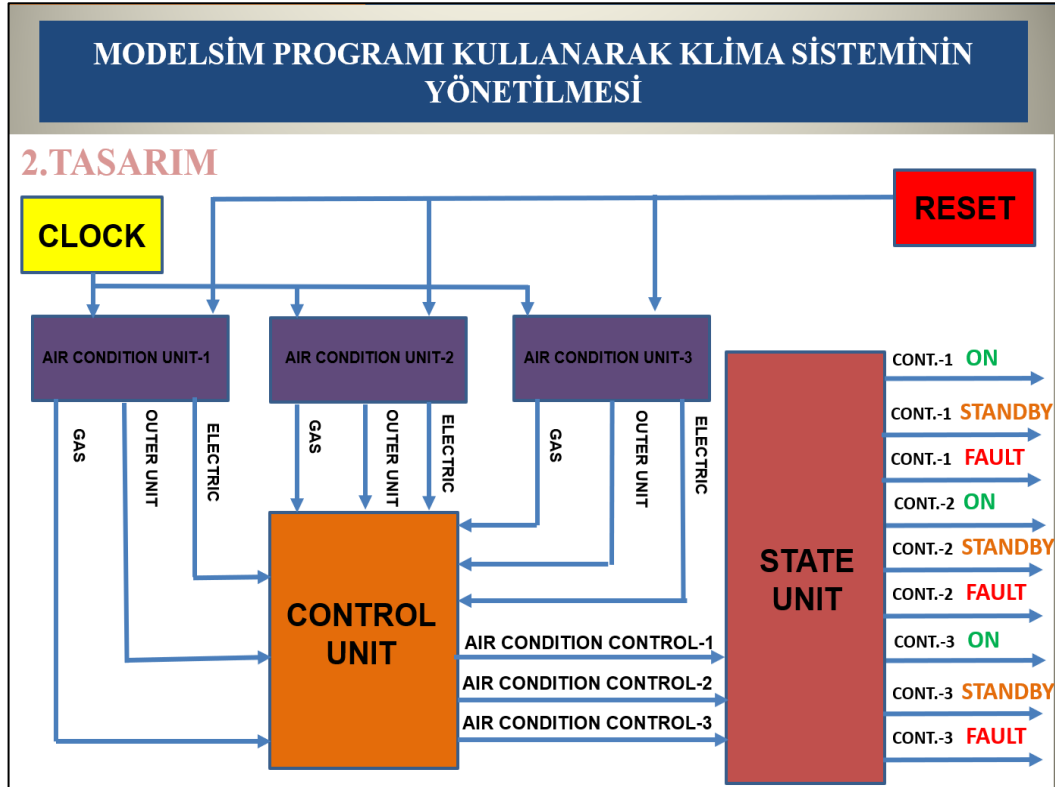
2.3.3. Bilinçli Kullanım

Klima cihazının montajının bitmesi demek işim bittiği anlamına gelmez. Montaj, kurallara uygun olarak yapıldıktan sonra cihazla ilgili tüm teknik ve kullanıma ait bilgilerle birlikte, kullanma kılavuzunun kullanıcıya verilmesi yetkili servisin sorumluluğudur. Anlatılan bu bilgileri anlamak ve uygulamak da tüketicinin sorumluluğudur.

2.3.4. Zamanında Doğru Bakım

Kullanıcıdan talep gelmesi halinde, periyodik bakım hizmetleri verilmelidir. Bakım klima cihazının çalışma verimi ve ömrü ile direkt ilgilidir. Bu yüzden bakımın gerekliliği kullanıcıya anlatılmalı ve bakım ile ilgili bilgiler verilmelidir.

2.4. Klima Kontrol Sisteminin Tasarımı



Tasarımda gösterilen 3 adet klimadan 1 ve 2 no.lu klimalar belirlenen süreler içerisinde sıralı bir şekilde çalışacaktır. 1 no.lu klima aktif olarak ON konumundayken 2 no.lu klima belirlenen süre sonunda devreye girmek üzere STANBY durumunda bekleyecektir.

3.no.lu klimada 1 veya 2 no.lu klima ON konumundayken yaşanabilecek bir FAULT ve 1 veya 2 no.lu klimanın STANBY konumundayken FAULT durumu düşmesi durumunda devreye girmek üzere STANBY konumunda bekleyecektir.

3 adet klimadan 2 adetinin arızalanması durumunda 1 adet klimayla da sistem FAULT alana kadar çalışmaya devam edecektir.

1, 2 ve 3 no.lu klimaların ELECTRIC, GAS, OUTER UNIT bölümlerinden gelecek bilgilere göre hızlı bir şekilde arızaya müdahale edilecek ve sistemin faaliyeti sağlanacaktır.

3.TASARIM KODLARI

```
library ieee;

use ieee.std_logic_1164.all;

use ieee.numeric_std.all;

use std.STANDARD.BOOLEAN;

entity air_condition is

port(

    Clk      : in std_logic;

    Rst      : in std_logic; -- Negative reset

    air_condition_1_devrede, air_condition_1_arizali, air_condition_1_beklemede : out
std_logic;

    air_condition_2_devrede, air_condition_2_arizali, air_condition_2_beklemede : out
std_logic;

    air_condition_3_devrede, air_condition_3_arizali, air_condition_3_beklemede : out
std_logic

);

end entity;

architecture sim_air_condition of air_condition is

    type t_State is (Durum1, Durum2, Durum3, durum4, durum5, durum6, durum7,
durum8, durum9, durum10, durum11, durum12);

    signal  Gaz1, Elektrik1, Dis_unite1,Gaz2, Elektrik2, Dis_unite2, Gaz3,
Elektrik3, Dis_unite3:std_logic:='1';
```

```

signal air_cont_1, air_cont_2, air_cont_3:std_logic;

signal State : t_State;

begin

air_cont_1<='1' when (gaz1='1' and elektrik1='1' and dis_unite1='1') else '0';

air_cont_2<='1' when (gaz2='1' and elektrik2='1' and dis_unite2='1') else '0';

air_cont_3<='1' when (gaz3='1' and elektrik3='1' and dis_unite3='1' and gaz1='0') else
    '1' when (gaz3='1' and elektrik3='1' and dis_unite3='1' ) else
    '1' when (gaz3='1' and elektrik3='1' and dis_unite3='1' and elektrik1='0') else
    '1' when (gaz3='1' and elektrik3='1' and dis_unite3='1' and dis_unite1='0') else
    '1' when (gaz3='1' and elektrik3='1' and dis_unite3='1' and gaz2='0') else
    '1' when (gaz3='1' and elektrik3='1' and dis_unite3='1' and elektrik2='0') else
    '1' when (gaz3='1' and elektrik3='1' and dis_unite3='1' and dis_unite2='0') else
'0';

process (clk)

variable count:integer:=0;

begin

if (rising_edge(clk)) then

IF (count >=20) THEN

    count := 0;

end if;

--RESET VALUES

if Rst = '0' then

```

```
air_condition_1_devrede <= '1';  
  
air_condition_1_arizali<='0';  
  
air_condition_1_beklemede<='0';  
  
air_condition_2_devrede <= '0';  
  
air_condition_2_arizali<='0';  
  
air_condition_2_beklemede<='1';  
  
air_condition_3_devrede <= '0';  
  
air_condition_3_arizali<='0';  
  
air_condition_3_beklemede<='1';
```

```
State <= durum1;
```

```
Count := 0;
```

```
else
```

```
-- DEFAULT VALUES
```

```
air_condition_1_devrede <= '0';  
  
air_condition_1_arizali<='0';  
  
air_condition_1_beklemede<='0';  
  
air_condition_2_devrede <= '0';  
  
air_condition_2_arizali<='0';  
  
air_condition_2_beklemede<='0';  
  
air_condition_3_devrede <= '0';  
  
air_condition_3_arizali<='0';
```

```

        air_condition_3_beklemede<='0';

    end if;

--1 VE 2 NOLU KLİMA SIRALI 3 DEVREYE GİRMEK ÜZERE BEKLEMEDE

    IF (count>=0) and (air_cont_1='1' and air_cont_2='1') and air_cont_3='1' THEN

        State <= durum1;

--2 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 1 VE 3 SIRALI
        ÇALIŞACAK

        elsif (count>=0) and (air_cont_1='1') and (air_cont_2='0') and
air_cont_3='1'THEN

            State <= durum4;

--1 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 2 VE 3 SIRALI
        ÇALIŞACAK

        elsif (count>=0) and (air_cont_1='0') and (air_cont_2='1') and
(air_cont_3='1')THEN

            State <= durum3;

--3 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 1 VE 2 SIRALI
        ÇALIŞACAK

        elsif (count>=0) and (air_cont_3='0') and (air_cont_1='1') and (air_cont_2='1')
THEN

            State <= durum6;

--1, 2 VE 3 NOLU KLİMALARIN ARIZALI OLMASI DURUMUNDA

        elsif (count>=0) and (air_cont_1='0') and (air_cont_2='0') and (air_cont_3='0')
THEN

            State <= durum12;

```

--1 VE 3 NOLU KLİMALAR ARIZALI OLMASI DURUMUNDA 2
NUMARA CALISACAK

elsiF (count>=0) and (air_cont_1='0') and (air_cont_2='1') and (air_cont_3='0')
THEN

State <= durum8;

--1 VE 2 NOLU KLİMALAR ARIZALI OLMASI DURUMUNDA 3 NUMARA
ÇALIŞACAK

elsIF (count>=0) and (air_cont_1='0') and (air_cont_2='0') and (air_cont_3='1')
THEN

State <= durum5;

--2 VE 3 NOLU KLİMALAR ARIZALI OLMASI DURUMUNDA 1 NUMARA
ÇALIŞACAK

elsIF (count>=0) and (air_cont_1='1') and (air_cont_3='0') and (air_cont_2='0')
THEN

State <= durum7;

end if;

--1 VE 2 NOLU KLİMA SIRALI 3 DEVREYE GİRMEK ÜZERE BEKLEMEDE

IF (count>=11) and (air_cont_1='1' and air_cont_2='1') and air_cont_3='1'
THEN

State <= durum2;

--2 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 1 VE 3 SIRALI
ÇALIŞACAK

elsIF (count>=11) and (air_cont_1='1') and (air_cont_2='0') and
air_cont_3='1'THEN

State <= durum9;

--3 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 1 VE 2 SIRALI
ÇALIŞACAK

```
elsIF (count>=11 ) and (air_cont_3='0') and (air_cont_1='1') and  
(air_cont_2='1') THEN
```

```
State <= durum11;
```

--1 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 2 VE 3 SIRALI
ÇALIŞACAK

```
elsiF (count>=11) and (air_cont_1='0')and (air_cont_2='1') and  
(air_cont_3='1')THEN
```

```
State <= durum10;
```

```
end if;
```

```
case State is
```

--1 VE 2 NOLU KLİMA SIRALI 3 DEVREYE GİRMEK ÜZERE BEKLEMEDE

```
when durum1 =>
```

```
air_condition_1_devrede <= '1';
```

```
air_condition_1_arizali<='0';
```

```
air_condition_1_beklemede<='0';
```

```
air_condition_2_devrede <= '0';
```

```
air_condition_2_arizali<='0';
```

```
air_condition_2_beklemede<='1';
```

```
air_condition_3_devrede <= '0';
```

```
air_condition_3_arizali<='0';
```

```
air_condition_3_beklemede<='1';
```


--1 VE 2 NOLU KLİMA SIRALI 3 DEVREYE GİRMEK ÜZERE BEKLEMEDE

when durum2 =>

air_condition_1_devrede <= '0';

air_condition_1_arizali<='0';

air_condition_1_beklemede<='1';

air_condition_2_devrede <= '1';

air_condition_2_arizali<='0';

air_condition_2_beklemede<='0';

air_condition_3_devrede <= '0';

air_condition_3_arizali<='0';

air_condition_3_beklemede<='1';

--1 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 2 VE 3 SIRALI CALISACAK

when durum3 =>

air_condition_1_devrede <= '0';

air_condition_1_arizali<='1';

air_condition_1_beklemede<='0';

air_condition_2_devrede <= '0';

air_condition_2_arizali<='0';

air_condition_2_beklemede<='1';

air_condition_3_devrede <= '1';

air_condition_3_arizali<='0';

air_condition_3_beklemede<='0';

--2 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 1 VE 3 SIRALI
ÇALIŞACAK

when durum4 =>

air_condition_1_devrede <= '1';

air_condition_1_arizali<='0';

air_condition_1_beklemede<='0';

air_condition_2_devrede <= '0';

air_condition_2_arizali<='1';

air_condition_2_beklemede<='0';

air_condition_3_devrede <= '0';

air_condition_3_arizali<='0';

air_condition_3_beklemede<='1';

--1 VE 2 NOLU KLİMALAR ARIZALI OLMASI DURUMUNDA 3 NUMARA
ÇALIŞACAK

when durum5 =>

air_condition_1_devrede <= '0';

air_condition_1_arizali<='1';

air_condition_1_beklemede<='0';

air_condition_2_devrede <= '0';

air_condition_2_arizali<='1';

air_condition_2_beklemede<='0';

air_condition_3_devrede <= '1';

air_condition_3_arizali<='0';

air_condition_3_beklemede<='0';

--3 NOLU DA ARIZA OLMASI DURUMUNDA 1 VE 2 SIRALI ÇALIŞACAK

when durum6 =>

air_condition_1_devrede <= '1';

air_condition_1_arizali<='0';

air_condition_1_beklemede<='0';

air_condition_2_devrede <= '0';

air_condition_2_arizali<='0';

air_condition_2_beklemede<='1';

air_condition_3_devrede <= '0';

air_condition_3_arizali<='1';

air_condition_3_beklemede<='0';

--2 VE 3 NOLU KLIMALAR ARIZALI OLMASI DURUMUNDA 1 NUMARA

ÇALIŞACAK

when durum7 =>

air_condition_1_devrede <= '1';

air_condition_1_arizali<='0';

air_condition_1_beklemede<='0';

air_condition_2_devrede <= '0';

air_condition_2_arizali<='1';

air_condition_2_beklemede<='0';

air_condition_3_devrede <= '0';

air_condition_3_arizali<='1';

air_condition_3_beklemede<='0';

--1 VE 3 NOLU KLİMALAR ARIZALI OLMASI DURUMUNDA 2 NUMARA
ÇALIŞACAK

when durum8 =>

air_condition_1_devrede <= '0';

air_condition_1_arizali<='1';

air_condition_1_beklemede<='0';

air_condition_2_devrede <= '1';

air_condition_2_arizali<='0';

air_condition_2_beklemede<='0';

air_condition_3_devrede <= '0';

air_condition_3_arizali<='1';

air_condition_3_beklemede<='0';

--2 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 1 VE 3 SIRALI
ÇALIŞACAK

when durum9 =>

air_condition_1_devrede <= '0';

air_condition_1_arizali<='0';

air_condition_1_beklemede<='1';

air_condition_2_devrede <= '0';

air_condition_2_arizali<='1';

air_condition_2_beklemede<='0';

air_condition_3_devrede <= '1';

air_condition_3_arizali<='0';

air_condition_3_beklemede<='0';

--1 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 2 VE 3 SIRALI
ÇALIŞACAK

when durum10 =>

air_condition_1_devrede <= '0';

air_condition_1_arizali<='1';

air_condition_1_beklemede<='0';

air_condition_2_devrede <= '1';

air_condition_2_arizali<='0';

air_condition_2_beklemede<='0';

air_condition_3_devrede <= '0';

air_condition_3_arizali<='0';

air_condition_3_beklemede<='1';

--3 NOLU KLİMADA ARIZA OLMASI DURUMUNDA 1 VE 2 SIRALI
ÇALIŞACAK

when durum11 =>

air_condition_1_devrede <= '0';

```
air_condition_1_arizali<='0';  
  
air_condition_1_beklemede<='1';  
  
air_condition_2_devrede <= '1';  
  
air_condition_2_arizali<='0';  
  
air_condition_2_beklemede<='0';  
  
air_condition_3_devrede <= '0';  
  
air_condition_3_arizali<='1';  
  
air_condition_3_beklemede<='0';
```

--1, 2 VE 3 NOLU KLİMALARIN ARIZALI OLMASI DURUMUNDA

when durum12 =>

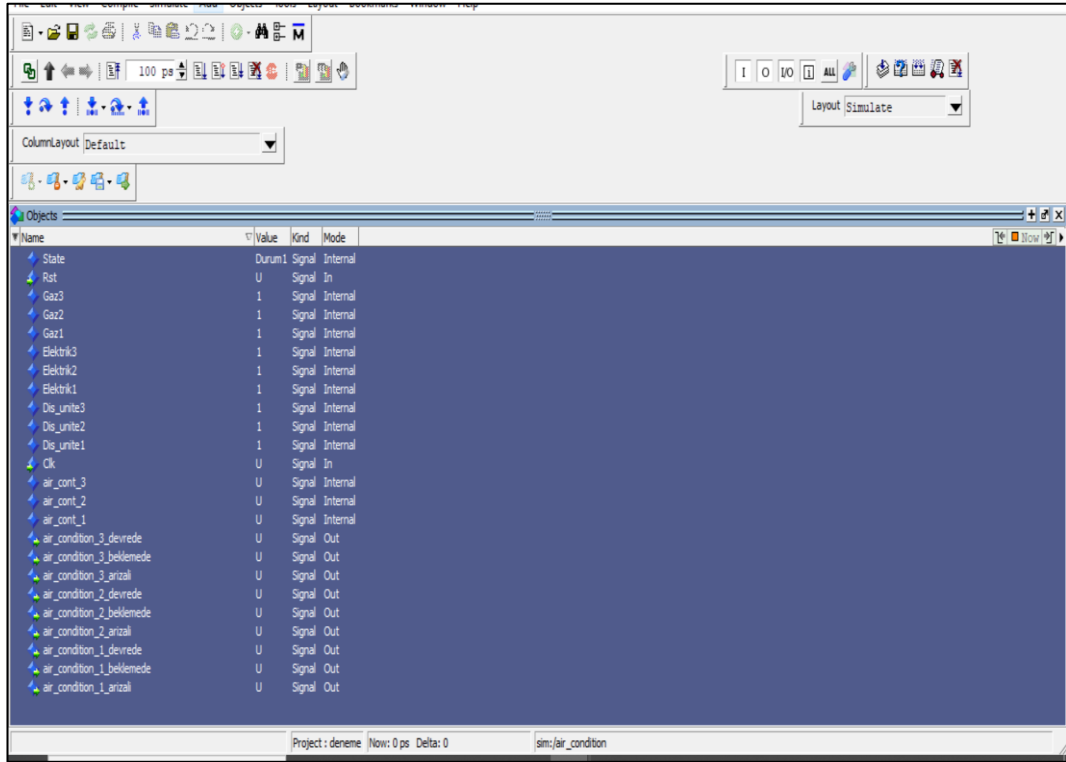
```
air_condition_1_devrede <= '0';  
  
air_condition_1_arizali<='1';  
  
air_condition_1_beklemede<='0';  
  
air_condition_2_devrede <= '0';  
  
air_condition_2_arizali<='1';  
  
air_condition_2_beklemede<='0';  
  
air_condition_3_devrede <= '0';  
  
air_condition_3_arizali<='1';  
  
air_condition_3_beklemede<='0';
```

end case;

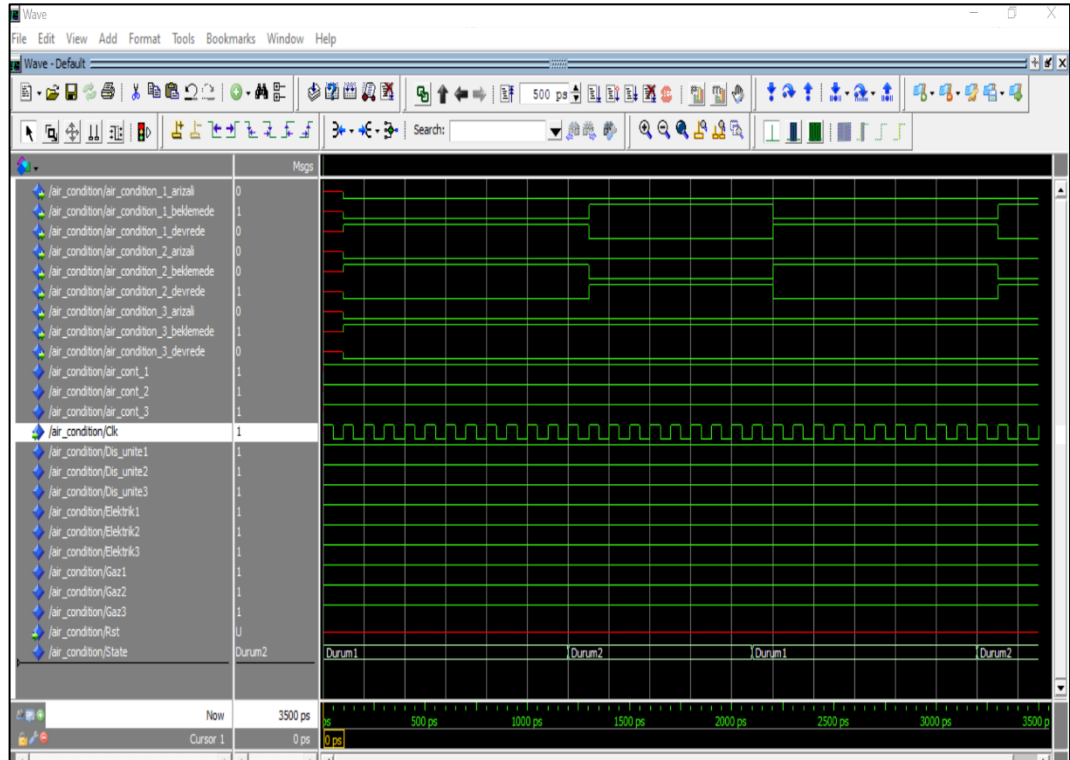
```
Count:=Count + 1;
```

```
end if;  
  
end process;  
  
end sim_air_condition;
```

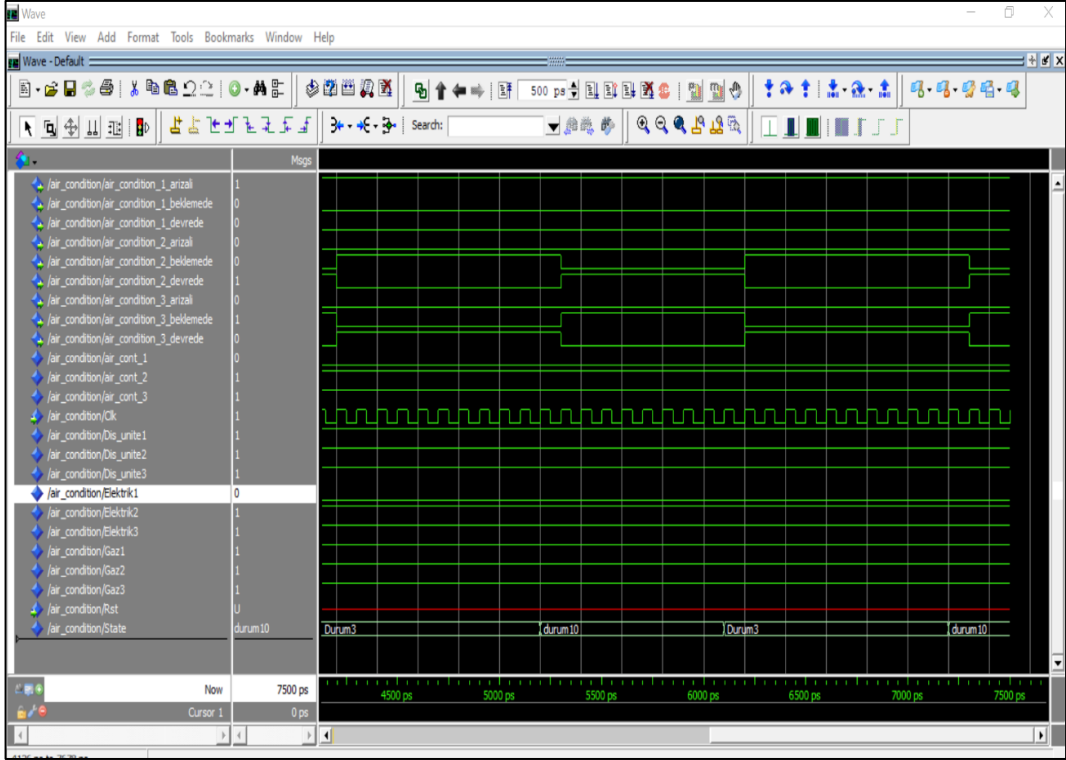
4.SİMÜLASYON GÖRSELİ



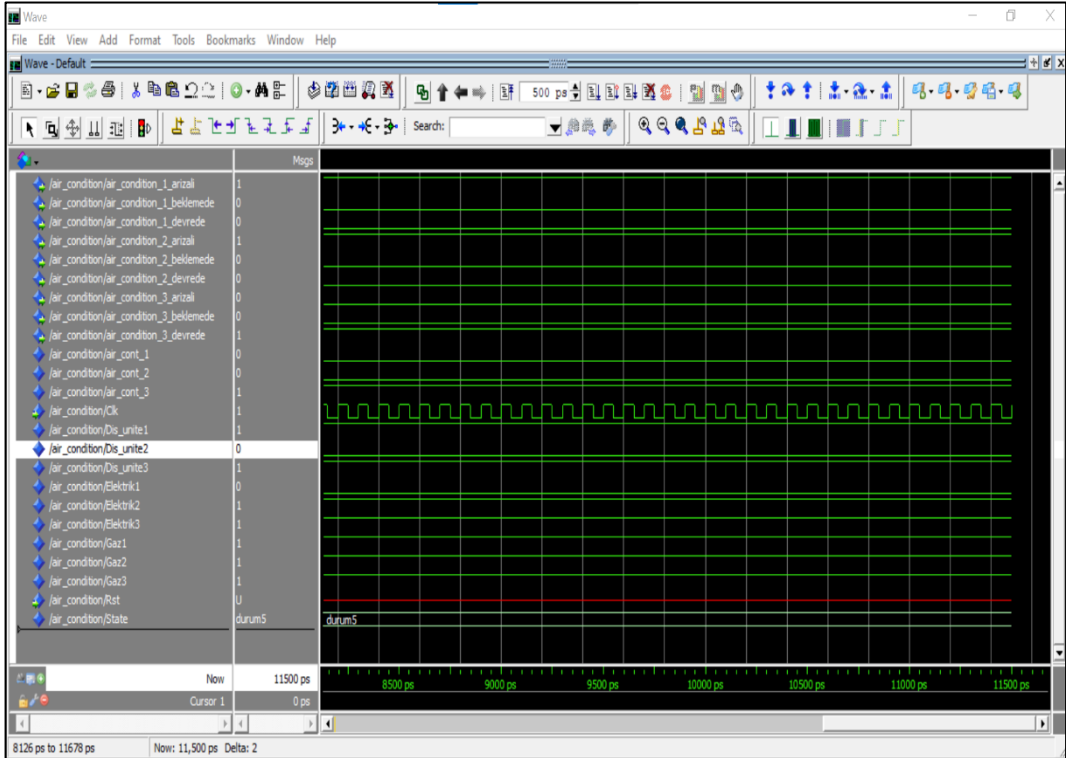
Şekil-4.1 Simülasyonda Yer alan Giriş ve Çıkış Nesnelerimiz



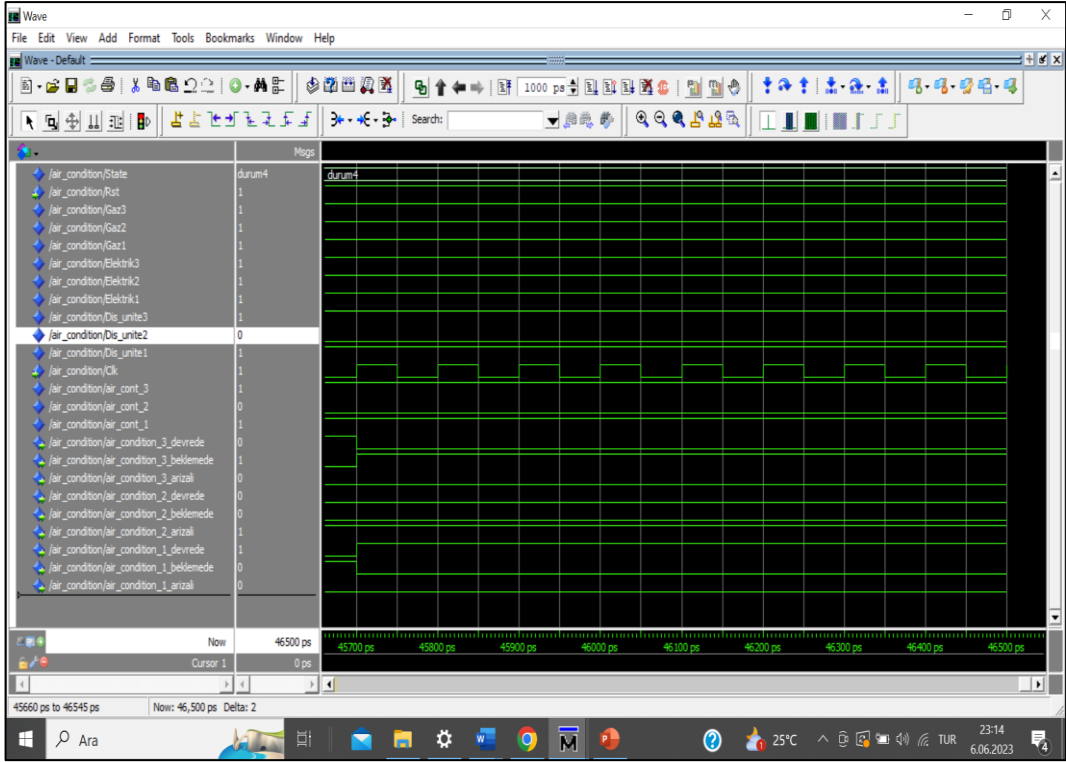
Şekil-4.2 1 ve 2 No.lu klimanın sıralı 3 no.lu klimanın Yedekli bir şekilde çalışması



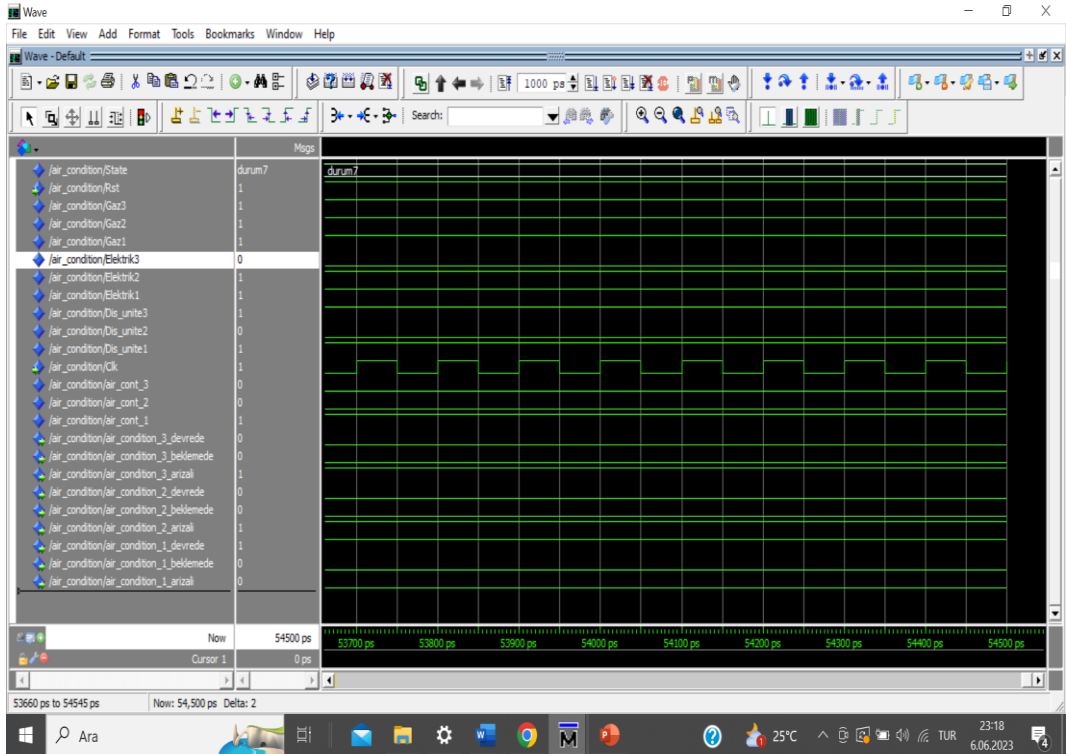
Şekil-4.3 1 no.lu klimada elektrik arızası olması sebebiyle 2 ve 3 no.lu klimanın sıralı bir şekilde çalışması



Şekil-4.4 1 no.lu klimada elektrik arızası, 2 no.lu klimada dış unite arızası olması sebebiyle 3 no.lu klimanın çalışması



Şekil-4.5 2 no.lu klimada dış ünite arızası olması sebebiyle 1 ve 3 no.lu klimanın sıralı bir şekilde çalışması



Şekil-4.6 2 no.lu klimada dış ünite, 3 no.lu klimada elektrik arızası olması sebebiyle 1 no.lu klimanın çalışması

KAYNAKÇA

- [1]. Zeidman, B., 2002. Designing with FPGAs and CPLDs, CMP Books, Kansas.
- [2]. Smith, G. R., 2010. FPGAs 101. Oxford: Elsevier.
- [3]. Matai, J., Irturk, A., and Kastner, R., 2011. Design and Implementation of an FPGA-based Real Time Face Recognition System, IEEE International Symposium on Field-Programmable Custom Computing Machines, Dept. Of Computer Science and Engineering, University of California, San Diego, USA, 97-100.
- [4]. Husin, M., Osman, F., Sabri, M. F. M., Abidin, W. A. W. Z., Othman, K., Marzuki, A.S.W., 2010. Development of Shape Pattern Recognition for FPGA-Based Object Tracking System, Faculty of Engineering Universiti Malaysia Sarawak, 2010 International Conference on Computer Applications and Industrial Electronics (ICCAIE 2010), December 5-7, 2010, Kuala Lumpur, Malaysia, 80 – 84.
- [5]. Tüzmir Katip Çelebi Üniversitesi VHDL Donanım Tanımlama Diline Giriş Ders Dökümanları
- [6]. Orhan Gazi - A Tutorial Introduction to VHDL Programming
- [7]. Alçın, Ö. F., 2011. Alan Programlanabilir Kapı Dizisi ile Sigma-Delta Modülatörlerin Gerçekleştirilmesi, Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ.
- [8]. <https://www.https://hassasklima.com/bilgi-bankasi/>
- [9]. <https://www.tproje.com/klima-kontrol/>

Özgeçmiş

Adı Soyadı: Ergin ÇALIŞKAN
E-mail (1): Y210234096@ogr.ikc.edu.tr
E-mail (2): ergincaliskan52@gmail.com

Eğitim:

2005–2007 Hava Astsubay Meslek Yüksekokulu Komutanlığı/Hava Astsubay
Meslek Yüksekokulu/Muhabere ve Elektronik Bilgi Sistemleri
2008–2011 Anadolu Üniversitesi/İşletme Fakültesi/İşletme
2017–2018 MEBS Subaylığı Temel Eğitimi

İş Deneyimi:

2007 – Halen Hava Kuvvetleri Komutanlığında Muhabere ve Elektronik Bilgi
Sistemleri