



**FUNDAMENTAL OF DIGITAL SYSTEM FINAL PROJECT REPORT
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITAS INDONESIA**

AUTOMATED DRIVE-THRU CAR WASH

GROUP A4

| | |
|-------------------------------|-------------------|
| M. AIDAN DAFFA | 1906300800 |
| BAGUS NURHUDA | 1906300681 |
| M. ILHAM MAULANA SIDIK | 1906300813 |
| TEDI SETIAWAN | 1906300864 |

PREFACE

Puji Syukur kehadiran Tuhan yang Maha Esa yang telah memberikan segala nikmat dan karunianya, sehingga kami masih diberi kesempatan untuk menyelesaikan proyek laporan ini pada Praktikum Perancangan Sistem Digital dengan judul “Automated Drive-Thru Car Wash” ini dengan tepat waktu.

Adapun tujuan dari penulisan laporan ini adalah untuk memenuhi tugas laporan proyek akhir untuk modul 10 pada Praktikum Perancangan Sistem Digital. Selain itu, laporan proyek akhir ini juga bertujuan untuk menambah wawasan dan pengetahuan tentang penerapan VHDL pada implementasi prototipe bagi para pembaca, penulis dan juga asisten laboratorium.

Kami ucapkan terima kasih sebanyak-banyaknya kepada setiap pihak yang telah mendukung serta membantu kami selama proses penyusunan dan penyelesaian laporan proyek akhir PSD ini. Kami memahami bahwa laporan ini masih dapat ditemukan banyak kekurangan. Oleh sebab itu, kami sangat menerima kritik dan saran yang diberikan untuk penyempurnaan laporan ini ataupun laporan di masa mendatang sehingga kualitas laporan kami dapat meningkat.

Terakhir, kami berharap laporan proyek yang telah kami buat dapat bermanfaat di kemudian hari serta dapat menjadi referensi & berguna bagi kemajuan teknologi di masa mendatang.

Depok, 2 Juni 2021

Group A4

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

- 1.1 Background
- 1.2 Project Description
- 1.3 Objectives
- 1.4 Roles and Responsibilities

CHAPTER 2: IMPLEMENTATION

- 2.1 Equipment
- 2.2 Implementation

CHAPTER 3: TESTING AND ANALYSIS

- 3.1 Testing
- 3.2 Result
- 3.3 Analysis

CHAPTER 4: CONCLUSION

REFERENCES

APPENDICES

- Appendix A: Project Schematic
- Appendix B: Documentation5

INTRODUCTION

1.1 BACKGROUND

Teknologi merupakan sebuah hal yang sering kita dengar belakangan ini. Perkembangan teknologi dapat dirasakan sangat cepat dan tidak bisa dihindarkan. berbagai sektor mulai dari perdagangan, perbankan, pertanian, transportasi, dan berbagai perusahaan jasa juga sudah menerapkan teknologi pada kehidupannya masing-masing. Perkembangan teknologi merupakan suatu hal positif dan harus terus dikembangkan dengan berbagai macam-macam inovasi dan kreativitas.

Umumnya, teknologi diciptakan untuk mempermudah pekerjaan manusia serta dapat mempersingkat waktu yang besar. Salah satu implementasinya yaitu pada penerapan bidang yang awalnya masih manual menggunakan tenaga kerja manusia, dapat digantikan oleh mesin secara automated, misalnya pada penerapan gerbang tol, ATM, Cashier drive thru, Vending Machine, auto pilot dan berbagai penerapan teknologi lainnya yang dirancang secara automated.

Oleh karena itu, dengan didukungnya dari modul-modul yang telah kami ikuti pada praktikum, kami memiliki inovasi untuk menciptakan automated car wash drive thru, dimana nanti pencucian dan pembayaran akan dilakukan secara otomatis dengan menggunakan mesin sejenis sensor detector dan vending machine.

1.2 PROJECT DESCRIPTION

Pada Proyek PSD kali ini, kami akan membuat Automated Drive Thru Car Wash dimana memiliki fitur otomatisasi sistem pada input jenis mobil, member dan uang. Sesuai fiturnya, mesin ini akan menginput ukuran mobil dengan menggunakan sensor, member card dengan menggunakan sensor infrared dan input koin dengan menggunakan 3 jenis uang kertas. mesin ini juga akan memiliki output berupa uang kembalian, penggerak mesin cuci mobil dan juga isi angin bagi yang memiliki member card. Pertama, kita akan mengambil enum variabel dari input sensor ukuran mobil dan juga member card. Ukuran mobil bertujuan untuk membedakan harga yang akan ditentukan dimana harga mobil kecil yaitu Rp30.000

dan mobil besar Rp40.000 , sedangkan member card untuk memberikan tambahan isi angin, jika tidak maka tidak mendapatkan bonus isi angin. setelah itu mobil akan berlanjut ke mesin input uang, dimana mesin ini mirip vending machine dan memiliki input 3 uang kertas yaitu Rp10.000, Rp20.000 dan Rp50.000 serta mesin juga akan mengeluarkan output uang kembalian apabila terdapat kelebihan uang. setelah uang telah memenuhi jumlah harga yang ditentukan dari ukuran mobil, makan mesing akan melakukan pencucian dan berlanjut melakukan isi ulang angin apabila user memiliki member card yang sudah di scan pada bagian input.

1.3 OBJECTIVES

1. Mengimplementasikan pemodelan Finite State Machine (FSM) kedalam sebuah perangkat keras
2. Memberikan contoh dan gambaran Finite State Machine (FSM) pada cara kerja perangkat Automated Drive-Thru Car Wash.
3. Menerapkan fitur otomatisasi pada konsep Drive-Thru dengan menggunakan sensor dan FSM sebagai input uang yang menghasilkan output sesuai dengan ketentuan input yang diberikan.
4. Menerapkan dan melakukan simulasi terhadap Automated Drive-Thru Car Wash dengan menggunakan testbench dan ModelSim.

1.4 ROLES AND RESPONSIBILITIES

| Roles | Responsibilities | Person |
|-----------|---|---|
| Laporan | Membuat laporan beserta komponen didalamnya | Bagus Nurhuda, Aidan Daffa, |
| Program | Membuat program VHDL murni dan memastikan program berjalan baik | Ilham Maulana, Tedi Setiawan, Bagus Nurhuda |
| TestBench | Membuat program testbench dari program utama dan memastikan | Tedi Setiawan, Ilham Maulana, Aidan Daffa |

| | | |
|--------------|--|---|
| | program test bench berjalan dengan baik | |
| Editor Video | mengumpulkan video, mengedit video dan mengupload video ke platform online. | Bagus Nurhuda, Aidan Daffa |
| Concept Idea | Menyusun ide rencana program dan konsep alatnya | Bagus Nurhuda, M Ilham, Tedi Setiawan, |

Table 1. Roles and Responsibilities

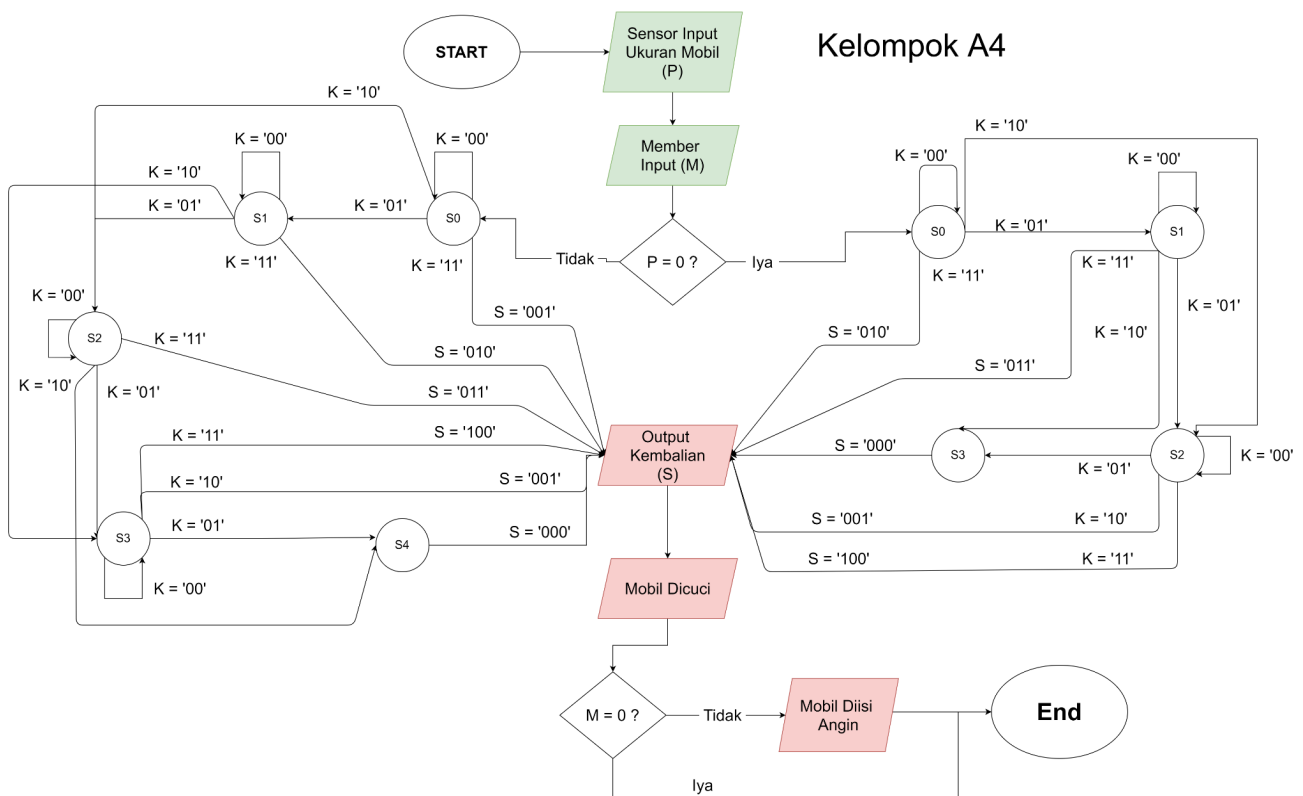
CHAPTER 2

IMPLEMENTATION

2.1 EQUIPMENT

- Source code VHDL (program + testbench)
- ModelSim (simulasi)
- Draw.io (Flowchart & State diagram maker)

2.2 IMPLEMENTATION



Gambar diatas merupakan flowchart dari program yang akan dibuat dalam VHDL. Program tersebut memiliki 3 input serta 3 output. 2 input diantaranya akan menentukan jumlah state yang akan dipakai dan menentukan 1 hasil outputnya. Pada input koin, terdapat maksimal jumlah state yaitu 5 jika saat ukuran mobil besar, sedangkan hanya 4 state ketika ukuran mobil kecil. Setiap state selalu memiliki 4 kemungkinan input namun tidak berlaku untuk state terakhir karena hanya meneruskan enum yang sudah dihasilkan. 4 kemungkinan input tersebut yaitu :

- K = '00' , yaitu ketika program tidak diinput sama sekali, sehingga state akan looping pada statenya
- K = '01' , yaitu ketika program diinput dengan menggunakan uang Rp10.000.
- K = '10' , yaitu ketika program diinput dengan menggunakan uang Rp20.000.
- K = '11' , yaitu ketika program diinput dengan menggunakan uang Rp50.000.

Dari setiap state yang diinput dengan menggunakan uang (K) akan menghasilkan kemungkinan output kembalian pada variabel S. berikut adalah variasinya :

- S = '000' , yaitu ketika output kembalian yang dikeluarkan sebesar nol/ tidak ada.
- S = '001' , yaitu ketika output kembalian yang dikeluarkan sebesar Rp 10.000.
- S = '010' , yaitu ketika output kembalian yang dikeluarkan sebesar Rp 20.000.
- S = '011' , yaitu ketika output kembalian yang dikeluarkan sebesar Rp 30.000.
- S = '100' , yaitu ketika output kembalian yang dikeluarkan sebesar Rp 40.000.

Setelah program berada pada state terakhir, maka program/mesin akan segera melakukan output berupa uang kembalian, pencucian mobil dan isi angin.

VHDL Code

```

1  -- This is a VHDL code by A4 Team
2  -- Automated Drive Thru Car Wash
3  -- A4
4  -- Bagus Nur Huda
5  -- Muh. Aidan Daffa
6  -- Muhammad Ilham Maulana
7  -- Tedi Setiawan
8
9  -- pada project Automated Drive Thru Car Wash akan diklasifikasikan menjadi 2 jenis mobil, yaitu besar dan kecil
10 -- harga untuk mobil besar adalah sebesar 40 koin
11 -- harga untuk kecil besar adalah sebesar 30 koin
12 -- jika memiliki member, maka akan terdapat pelayanan tambahan berupa isiBan
13
14
15 library ieee;
16 use ieee.std_logic_1164.all;
17
18 entity carwash_entity is
19     port(
20         --Input
21         member, intUkuran, CLR, CLK : IN STD_LOGIC;
22         koin : IN STD_LOGIC_VECTOR (1 downto 0);
23         --Output
24         MesinCuci, isiBan : OUT STD_LOGIC;
25         Kembalian : OUT STD_LOGIC_VECTOR (2 downto 0)
26     );
27 end entity;
28
29
30 architecture carwash_architecture of carwash_entity is -- Berikut adalah architecture dari entity carwash_entity
31
32     type ST is (S0, S1, S2, S3, S4);
33     type ukuran is (kecil, besar);
34     signal PS, NS : ST;
35     signal mobil : ukuran;
36
37     -- Deklarasi standart library IEEE
38
39     -- Deklarasi Port yang digunakan dalam proyek Automated Drive
40     -- Dengan nama Entity "carwash_entity"
41     -- Berikut adalah port input yang digunakan pada proyek Autor
42     -- Terdapat 4 input yang bertipe STD_LOGIC
43     -- Terdapat 1 input yang bertipe STD_LOGIC_VECTOR (0 = 00, 1 = 01, 2 = 10, 3 = 11)
44     -- Berikut adalah port output yang digunakan pada proyek Auto
45     -- Terdapat 2 output yang bertipe STD_LOGIC
46     -- untuk output MesinCuci dan isiBan 1 = TRUE (dilak
47     -- Terdapat 1 input yang bertipe STD_LOGIC_VECTOR
48     -- untuk output Kembalian (0 = 000, 10 = 001, 20 = 010, 30 = 011, 40 = 100)

```



```

36 L
37 begin
38
39 SyncProc: process (CLK, NS, CLR)
40 begin
41     if (CLR = '1') then PS <= S0;
42     elsif (rising_edge(CLK)) then PS <= NS;
43     end if;
44 end process SyncProc;
45
46 CombProc: process (PS, koin, mobil, member, intUkuran)
47 begin
48     MesinCuci <= '0';
49     Kembalian <= "000";
50     isiBan <= '0';
51
52     case intUkuran is
53     when '0' =>
54         mobil <= kecil;
55     when others =>
56         mobil <= besar;
57     end case;
58
59     case PS is
60     when S0 =>
61         MesinCuci <= '0';
62         if (koin = "11") then
63             if (mobil = kecil) then
64                 NS <= S3;
65                 Kembalian <= "010";
66             elsif (mobil = besar) then
67                 NS <= S4;
68                 Kembalian <= "001";
69             else Kembalian <= "000";
70             end if;
71         elsif (koin = "10") then NS <= S2;
72         elsif (koin = "01") then NS <= S1;
73         else NS <= S0;
74         end if;
75     when S1 =>
76         MesinCuci <= '0';
77         if (koin = "11") then
78             if (mobil = kecil) then
79                 NS <= S3;
80                 Kembalian <= "011";
81             elsif (mobil = besar) then
82                 NS <= S4;
83                 Kembalian <= "010";
84             else Kembalian <= "000";
85             end if;
86         elsif (koin = "10") then NS <= S3;
87         elsif (koin = "01") then NS <= S2;
88         else NS <= S1;
89         end if;
90     when S2 =>
91         MesinCuci <= '0';
92         if (koin = "11") then
93             if (mobil = kecil) then
94                 NS <= S3;

```

```

-- Synchronous Process dengan CLK, NS,
-- Saat CLR bernilai 1 maka akan PS (P:
-- Saat CLK High, PS akan bernilai NS
-- Synchronous Process selesai
-- Combinatorial Process dengan PS, ko:
-- Mengembalikan value MesinCuci, kemb:
-- Case statement untuk menentukan val:
-- Saat input intUkuran = 0, maka mobi:
-- Saat input intUkuran = 1, maka mobi:
-- Case statement untuk menentukan PS,
-- Saat PS = S0
-- maka MesinCuci tidak akan dilak:
-- pada state ini jika diberi inpu:
-- lalu jika kondisi signal m:
-- NS = S3
-- Kembalian = 010 (20)
-- lalu jika kondisi signal m:
-- NS = S4
-- Kembalian = 001 (10)
-- pada state ini jika diberi inpu:
-- pada state ini jika diberi inpu:
-- Saat PS = S1
-- maka MesinCuci tidak akan dilak:
-- pada state ini jika diberi inpu:
-- lalu jika kondisi signal m:
-- NS = S3
-- Kembalian = 011 (30)
-- lalu jika kondisi signal m:
-- NS = S4
-- Kembalian = 010 (20)
-- pada state ini jika diberi inpu:
-- pada state ini jika diberi inpu:
-- Saat PS = S2
-- maka MesinCuci tidak akan dilak:
-- pada state ini jika diberi inpu:
-- lalu jika kondisi signal m:
-- NS = S3

```

```

91      MesinCuci <= '0'; -- maka MesinCuci tidak akan
92      if (koin = "11") then -- pada state ini jika diberi
93          if (mobil = kecil) then -- lalu jika kondisi sigr
94              NS <= S3; -- NS = S3
95              Kembalian <= "100"; -- Kembalian = 100 (40)
96          elsif (mobil = besar) then -- lalu jika kondisi sigr
97              NS <= S4; -- NS = S4
98              Kembalian <= "011"; -- Kembalian = 011 (30)
99          else Kembalian <= "000";
100         end if;
101     elsif (koin = "10") then -- pada state ini jika diberi
102         if (mobil = kecil) then -- lalu jika kondisi sigr
103             NS <= S3; -- NS = S3
104             Kembalian <= "001"; -- Kembalian = 001 (10)
105         elsif (mobil = besar) then -- lalu jika kondisi sigr
106             NS <= S4; -- NS = S4
107             Kembalian <= "000"; -- Kembalian = 000 (00)
108         end if;
109     elsif (koin = "01") then NS <= S3; -- pada state ini jika diberi
110     else NS <= S2;
111     end if;
112 when S3 => -- Saat PS = S3
113     if (mobil = kecil) then -- lalu jika kondisi signal
114         MesinCuci <= '1'; -- maka MesinCuci akan di
115         if (member = '1') then -- jika member = 1
116             isiBan <= '1'; -- maka isiBan akan dilak
117         else isiBan <= '0'; -- selain 1, maka isiBan
118         end if;
119         NS <= S0; -- Kembalian = 001 (40)
120     elsif (mobil = besar) then -- lalu jika kondisi signal
121         MesinCuci <= '0'; -- maka MesinCuci tidak a
122         if (koin = "11") then -- pada state ini jika di
123             NS <= S4; -- NS = S4
124             Kembalian <= "100"; -- Kembalian = 100 (4
125         elsif (koin = "10") then -- pada state ini jika di
126             NS <= S4; -- NS = S4
127             Kembalian <= "001"; -- Kembalian = 001 (4
128         elsif (koin = "01") then -- pada state ini jika di
129             NS <= S4; -- NS = S4
130             Kembalian <= "000"; -- Kembalian = 000 (0
131         else NS <= S3;
132         end if;
133     else NS <= S3;
134     end if;
135 when S4 => -- Saat PS = S4
136     MesinCuci <= '1'; -- maka MesinCuci akan dilak
137     if (member = '1') then -- jika member = 1
138         isiBan <= '1'; -- maka isiBan akan dilak
139     else isiBan <= '0'; -- selain 1, maka isiBan
140     end if;
141     NS <= S0; -- NS = S0
142 when others => -- Saat PS = selain S0, S1, S2, S
143     MesinCuci <= '0'; -- maka MesinCuci tidak akan
144     NS <= S0; -- NS = S0
145 end case;
146
147 end process CombProc; -- Combinatorial Process selesai
148
149 end architecture;

```

Program code Automated Drive-Thru Car Wash yang telah dibuat merupakan hasil dari penggunaan bahasa VHDL dengan menggunakan konsep Finite State Machine dengan tipe Mealy. Program disusun dengan menggunakan input member, intUkuran, CLR, CLK dan

koin berupa vector serta outputnya yaitu MesinCuci, IsiBan, dan Kembalian (vector) yang terdapat pada portmap entity.

Pada bagian architecture, dilakukan deklarasi bagian state dari S0-S4 dan tipe mobilnya (besar dan kecil). Setelah itu juga dilakukan pendeklarasian signal present state dan next state dari state yang sudah dideklarasikan sebelumnya. Dimana PS dan NS ini akan bertujuan untuk memberikan gambaran output pada state yang sedang dijalankan dan berikutnya.

Program juga dibuat dengan menggunakan sinkronus proses dan kombinasional proses. Sinkronus proses bertujuan untuk mereset PS dan NS menjadi S0 menggunakan clock pada rising edge. Pada kombinasional proses, program dibuat secara berurutan dari S0 hingga S4 dengan didahului mereset output agar hasilnya sesuai. Pada setiap statenya, mulai dari S0 hingga S4 dilakukan percabangan dengan menggunakan if statement sesuai dengan ukuran mobilnya dan koinnya yang masing-masing pada setiap kondisi tersebut juga akan mempengaruhi output kembalian dan juga next statenya.

Misal pada S0 program akan diinput dengan menggunakan 3 koin ('01', '10', '11'). jika input = '11' pada mobil kecil, maka NS akan langsung ke S3 dan akan mengembalikan uang sebesar '010' jika pada mobil besar maka NS akan langsung ke S4 dan mengembalikan uang sebesar '001'. jika diinput dengan koin '10' maka NSnya S2. Jika diinput dengan koin '01' maka NSnya yaitu S1 dan selain itu semua, maka NS = PS. Contoh tersebut dapat diterapkan pada semua state kecuali state terakhir pada setiap ukuran mobil.

penggunaan program pada S3 di mobil kecil dan S4 di mobil besar juga berbeda dengan state yang lainnya dimana state tersebut digunakan untuk membuat output diisi dengan bit '1' sesuai inputnya seperti output kembalian, mesin cuci dan isi angin sehingga tidak lagi diperlukan input koin/uang pada state tersebut.

TechBench Code

```

9  --Library
10 library ieee;
11 use ieee.std_logic_1164.all;
12 use ieee.numeric_std.all;
13
14 --Entity
15 entity carwash_ent_tb is
16 end entity;
17
18 --Architecture
19 architecture carwash_arch_tb of carwash_ent_tb is
20     component carwash_entity is
21     port(
22         --Input pada
23         member, intUkuran, CLR, CLK : IN STD_LOGIC;
24         koin : IN STD_LOGIC_VECTOR (1 downto 0);
25         --Output
26         MesinCuci, isiBan : OUT STD_LOGIC;
27         Kembalian : OUT STD_LOGIC_VECTOR (2 downto 0)
28     );
29 end component;
30
31 signal member, intUkuran, CLR, CLK, MesinCuci, isiBan : STD_LOGIC;
32 signal koin : STD_LOGIC_VECTOR (1 downto 0);
33 signal Kembalian : STD_LOGIC_VECTOR (2 downto 0);
34 signal i : integer := 0;
35
36 begin
37     -- menghubungkan signal testbench dengan Automated Drive Thru Car Wash port (port mapping)
38     UUT : carwash_entity port map(member,intUkuran,CLR,CLK,koin,MesinCuci,isiBan,Kembalian);
39
40     tb1:process
41
42         --mengatur clock dan jumlah clock maksimum
43         constant T:time:= 20 ns;
44         constant max_clk : integer := 4;
45         --membuat suatu tipe data yang berjenis array yang berisikan array juga di dalamnya
46         type intkoin is array (0 to 3) of STD_LOGIC_VECTOR (1 downto 0);
47         type cek_kembalian is array (0 to 4) of STD_LOGIC_VECTOR (2 downto 0);
48
49         type intkoin is array (0 to 3) of STD_LOGIC_VECTOR (1 downto 0);
50         type cek_kembalian is array (0 to 4) of STD_LOGIC_VECTOR (2 downto 0);
51         --membuat look up table
52         constant stream_koin : intkoin := ( 0=>"01", 1=>"01", 2=>"10", 3=>"11");
53         constant stream_mobil : STD_LOGIC_VECTOR(0 to 3) := "0010";
54         constant stream_opt : STD_LOGIC_VECTOR(0 to 4) := "00011";
55         constant stream_kembalian : cek_kembalian := ( 0=>"000", 1=>"000", 2=>"000", 3=>"000", 4=>"010");
56         constant stream_isiBan : std_logic_vector(0 to 4) := ("00011");
57
58         begin
59             CLK <='0';
60             CLR <='0';
61
62             wait for T/2;
63             CLK <='1';
64             wait for T/2;
65
66             if(i < max_clk) then i <= i+1;
67             if (i > 1) then member <= '1';
68             end if;
69             intUkuran <= stream_mobil(i);
70             koin <= stream_koin(i);
71             else wait;
72             end if;
73
74             -- statement untuk menyesuaikan kondisi pada parameter yang dibandingkan
75             assert((MesinCuci = stream_opt(i)) and (Kembalian = stream_kembalian(i)) and (isiBan = stream_isiBan(i)))
76             -- jika ada yang tidak sesuai maka akan terdapat feedback atau report berupa :
77             report "testbench projek fail pada loop ke-" & integer'image(i) severity error;
78             end process;
79 end architecture;

```

Program diatas merupakan Testbench code VHDL dari program awal Automated Drive-Thru Car Wash. Testbench yang sudah dibuat yaitu dengan mengubah seluruh entity pada program utama termasuk portmap, signal, dll menjadi component didalam architectur testbenchnya. setelah itu semua komponen input portmap menjadi satu kedalam UUT.

Setelah semuanya di deklarasikan kembali pada architecture, selanjutnya memulai process statement tb yang diawali deklarasi waktu setiap periodenya sebesar 20ns dan melakukan perulangan sebanyak 4 kali. setelah itu juga diperlukan tipe data array untuk menyimpan data berjenis vektor pada intkoin dan kembaliannya. Hal ini bertujuan untuk memasukkan array tersebut kedalam array stream.

Selanjutnya pada test bench juga harus membuat stream dimana untuk mensimulasikan input secara otomatis pada setiap loop yang akan dijalankan serta 3 stream output untuk digunakan dalam assert and report. Pada bagian ini kita memasukkan uji coba input dengan menggunakan 2 koin Rp10K, 1 koin Rp20K dan 1 koin Rp50k dengan menggunakan ukuran mobil besar & kecil. selain itu kita juga menentukan stream untuk kembalian, cuci mesin dan isi angin sesuai dengan output yang benar.

Setelah pembuatan stream, selanjutnya mereset & mengatur clock dan clear agar dapat berjalan sesuai fungsinya. Kemudian baru membuat looping dengan sesuai banyak perulangan yang sudah dideklarasikan di awal. looping ini bertujuan untuk memasukkan input dari stream yang telah diset sebagai input yaitu stream koin dan stream ukuran mobil. Selanjutnya, bagian terakhir dari testbench yaitu membuat report dan assert untuk membuktikan apakah program yang telah diatur berhasil atau tidak.

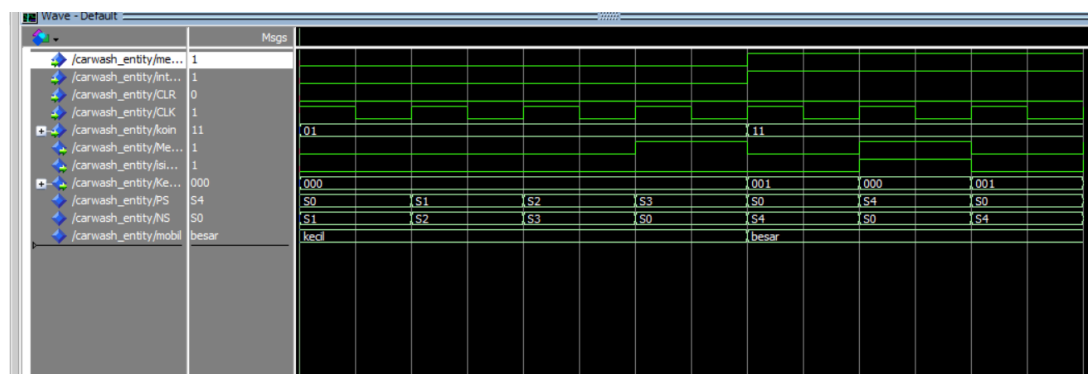
CHAPTER 3

TESTING AND ANALYSIS

3.1 TESTING

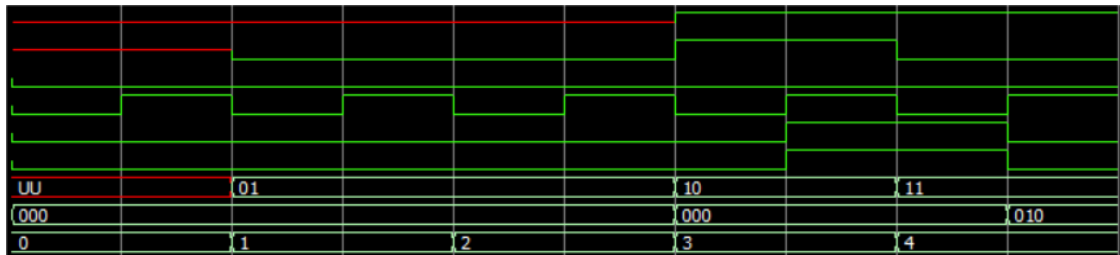
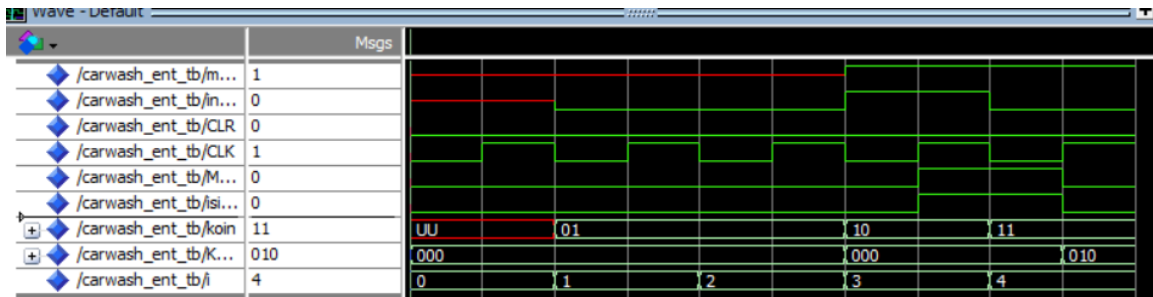
- Kode VHDL Automated Drive Thru Car Wash

percobaan kode dilakukan dengan menggunakan modelsim, setelah di simulate didapat hasil sebagai berikut.



- Testbench dari Automated Drive Thru Car Wash

percobaan kode dilakukan dengan menggunakan modelsim, setelah di simulate didapat hasil sebagai berikut.



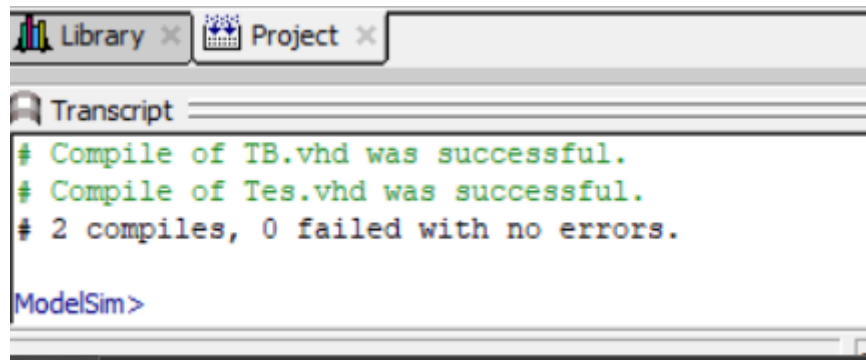
3.2 RESULT

pada testbench dilakukan 2x percobaan, yang pertama mobil kecil, pada mobil kecil output mesin cuci terlaksana pada looping ke3, dan pada percobaan mobil besar terlaksana pada looping ke4. begitu juga output isi ban nya.

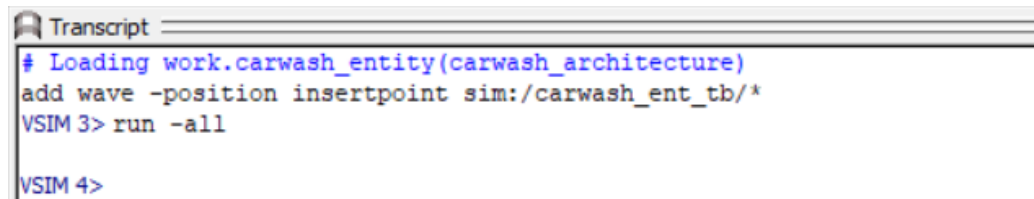
- Tampilan pada transcript
 - di kode vhdl Automated Drive Thru Car Wash

```
force -freeze sim:/carwash_entity/CLK 1 0, 0 {50 ps} -r 100
force -freeze sim:/carwash_entity/koin 01 0
VSIM 42> run
run
run
VSIM 43> run
force -freeze sim:/carwash_entity/koin 11 0
force -freeze sim:/carwash_entity/intUkuran 1 0
force -freeze sim:/carwash_entity/member 1 0
VSIM 46> run
run
VSIM 47> run
VSIM 47>
```

- Di testbench Automated Drive Thru Car Wash



```
Library x Project x
Transcript
# Compile of TB.vhd was successful.
# Compile of Tes.vhd was successful.
# 2 compiles, 0 failed with no errors.
ModelSim>
```



```
Transcript
# Loading work.carwash_entity(carwash_architecture)
add wave -position insertpoint sim:/carwash_ent_tb/*
VSIM 3> run -all
VSIM 4>
```

pada transcript menandakan bahwa tidak ada error yang terjadi pada program.

3.3 ANALYSIS

Berdasarkan percobaan yang telah dilakukan diatas, didapatkan hasil pada test bench Automated Drive Thru Car Wash sesuai dengan apa yang dituliskan pada kode. Hal tersebut dapat dilihat pada hasil di wave, dan dapat dibuktikan tidak ada feedback assert-report pada transcript. Dan hasil data pada wave di testbench juga sudah sesuai dengan kode VHDL Automated Drive Thru Car Wash. Pada testbench pada clock pertama terdapat value undefined. Hal ini dikarenakan mesin belum mendeteksi input. Oleh karena itu kita mengakali pada output di port test bench kita melebihkan satu bit percobaan. Misal jika kita memerlukan 4 bit untuk 4 looping. Maka akan dilebihkan 1 bit input yang diperuntukan untuk input undefined tersebut yang terjadi pada clock 1.

Pada test bench tersebut dilakukan 2x percobaan. pada percobaan 1, dilakukan pada mobil kecil, non member, dan memberikan input koin pertama 01 (Rp. 10.000). Hal ini sesuai karena setelah diinput koin 01 (Rp. 10.000) sebanyak 3 kali, terdapat output yang menyatakan mesin cuci menyala (1), kembali 000 (0), dan tidak ada isi ban. Pada percobaan kedua, dilakukan pada mobil besar, member, dan input koin pertama adalah 11 (Rp. 50.000). Hal ini sesuai karena terdapat output yang menyatakan mesin cuci menyala (1), kembali 001 (Rp. 10.000), dan terdapat pengisian ban. Kedua percobaan tersebut telah sesuai berdasarkan

kode VHDL Automated Drive Thru Car Wash dan telah sesuai juga berdasarkan state diagram yang ada.

CHAPTER 4

CONCLUSION

VHDL atau VHSIC HDL merupakan suatu bahasa pemrograman yang digunakan untuk mendeskripsikan suatu hardware. Dimana terdiri atas sejumlah gerbang- gerbang yang dapat diprogram. dalam pengerjaan proyek Automated Drive Thru Car Wash yang dikembangkan oleh kelompok A4, kelompok A4 dapat mengimplementasikan bahasa pemrograman VHDL ini terhadap proyek Automated Drive Thru Car Wash tersebut.

Pada proyek kelompok A4,yaitu Automated Drive Thru Car Wash kami menggunakan 5 input yaitu member, intUkuran, CLR, CLK, dan koin. dan menghasilkan 3 output, yaitu MesinCuci, isiBan, dan Kembalian. Terdapat 2 klasifikasi untuk intUkuran yaitu kecil dan besar. Untuk input koin, kelompok A4 menggunakan 3 jenis koin yaitu 01 (Rp. 10.000), 10 (Rp. 20.000), dan 11 (Rp. 50.000). Dan jika pengendara tersebut merupakan member maka akan terdapat bonus berupa isiBan.

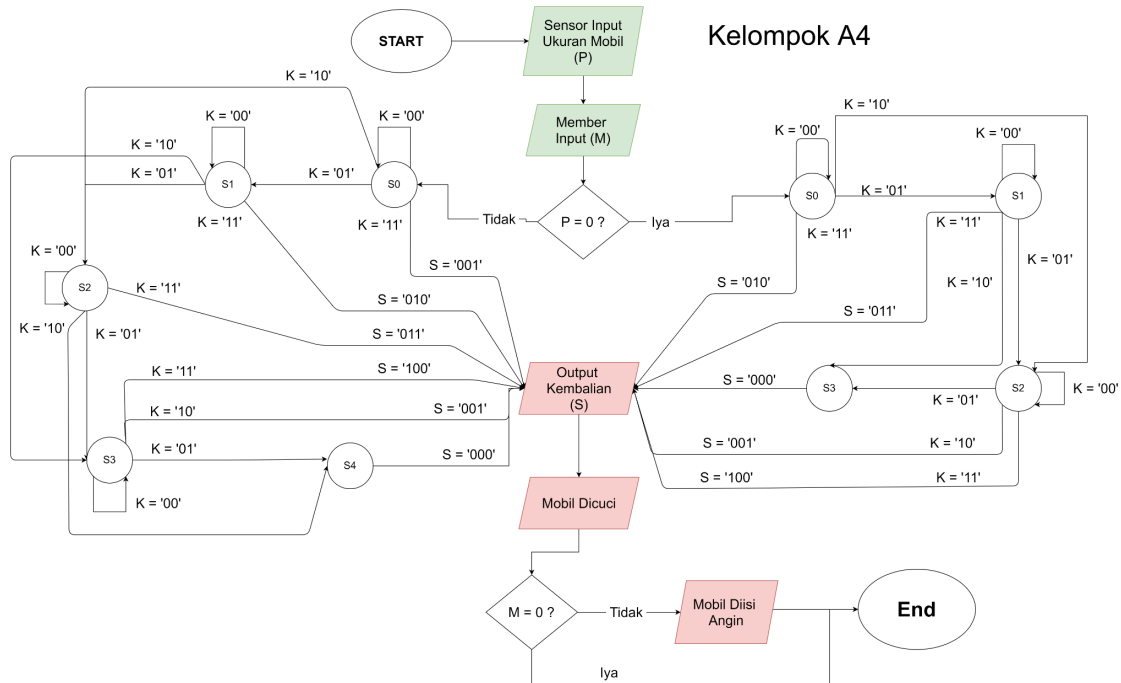
Berdasarkan hasil simulasi terbukti bahwa program VHDL dan test bench mengalami keberhasilan sesuai dengan konsep pemikiran diawal. Demikian kode VHDL Automated Drive Thru Car Wash dan laporan proyek akhir dari kelompok 4. Semoga proyek kami bisa berguna untuk kita sendiri maupun orang lain. Kami memohon maaf jika masih ada kekurangan baik dalam kode pemrograman ataupun dalam laporan.

REFERENCES

- [1] FREE RANGE VHDL 2021, FREE RANGE VHDL by Meally Bryan and T. Fabrizio , available at: freerangefactory.org [5 Juni 2021].
- [2] VHDLwhiz. 2021. *How to create a Finite-State Machine in VHDL - VHDLwhiz*. [online] Available at: <https://vhdlwhiz.com/finite-state-machine/> [Accessed 6 June 2021].
- [3] Implementing a Finite State Machine in VHDL. 2021. [online] Available at: <https://www.allaboutcircuits.com/technical-articles/implementing-a-finite-state-machine-in-vhdl/> [Accessed 6 June 2021].

APPENDICES

Appendix A: Project Schematic



Appendix B: Documentation

