|  |  |  |
| --- | --- | --- |
| cf |  |  |
| **Benha University** |  | **Faculty of Computers & Artificial Intelligence** |

**BBqM™**

**In**

Computer Architecture

**by**

Student’s Full Name (Student Section Number)

|  |
| --- |
| Manar Hamada Elsayed (Section 22) |
| Mona Mohamed Hamdy (Section 22) |
| Menna Muhammad Ibrahim (Section 22) |

**Supervised by**

Eng. Abdallah El Ghamry

Table of Contents

[1. Introduction 1](#_Toc39109705)

[2. Content 1](#_Toc39109706)

[*2.1* Deliverables](#_Toc39109707) 4

[*2.2* *Testbench result* 2](#_Toc39109708)

# Introduction

Implementing an embedded system inside a bank which monitors the clients queue in front of the tellers. Monitoring was divided into calculating number of people waiting in the queue, and displaying the expected waiting time for each client in the queue, and the maximum number of clients in the queue. The system was simulated using ModelSim, and coded using Verilog.

# Content

## *Deliverables*

In this project, you are going to model the operation of BBqM™ and verify it via simulation. Then, you will synthesize the model. Here is the list of deliverables:

1. In a table, identify *BBqM™* inputs and outputs and briefly describe their meaning and possible values.

Table ‎2.1 inputs and outputs

|  |  |  |
| --- | --- | --- |
| **Pin/s Name** | **Type** | **Description** |
| **Reset** | Input | Reset the system |
| **CountUp/Down** | Input | Count up when one client enter the queue or leave |
| **TCount** | Input | Number of tillers at the counter |
| **Pcount** | Output | Number of clients in the queue |
| **Wtime** | Output | To indicate Waiting Time |
| **EmptyFlag** | Output | To indicate that queue is empty |
| **FullFlag** | Output | To indicate that queue is full |
| **Alarm** | Output | To indicate that queue is getting full or empty |

b) Draw an icon for the BBqM™, clearly showing its input and output signals.

Reset

BBqM™

Alarm

Count UP/Down EmptyFlag

FullFlag

Tcount Pcount

wtime

c) Draw a block diagram showing the BBqM™ structure.

Icon

Description automatically generated

Reset

Flags

UpDownCounter

Pcount[0:3] Full

CLK

Empty

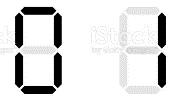
Up/Down Alarm

T0

Teller

Wtime

T1 Tcount[1:0]  **/**

T2 **%** 

d) Draw the necessary FSM diagram(s).

Reset

|  |  |
| --- | --- |
|  | Up |
|  | Down |
| S0 | Empty |
| S7 | Full |

e) In a table, propose a test strategy to verify the operation of the BBqM™ model. Carefully select an appropriate set of test cases that test various design aspects. For example, you should test that the two flags reflect the correct status of the queue. Additionally, you should ensure that the counter does not wrap around. That is, it does not display ‘0’ when it gets an increment signal while the queue is full or ‘7’ when it gets a decrement signal while the queue is empty

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Reset | Up/Down | T1 | T0 | PCOUNT |  | WTime |  |  | F | E |
| 1 | x | x | x | 0 | ~1111110 | 0 | ~1111110 | ~1111110 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | ~0110000 | 3 | ~1111110 | ~1111001 | 0 | 0 |
| 0 | 1 | 0 | 1 | 2 | ~1101101 | 6 | ~1111110 | ~1011111 | 0 | 0 |
| 0 | 1 | 1 | 0 | 2 | ~1101101 | 4 | ~1111110 | ~0110011 | 0 | 0 |
| 0 | 1 | 1 | 1 | 3 | ~1111001 | 5 | ~1111110 | ~1011011 | 0 | 0 |
| 0 | 1 | 1 | 1 | 4 | ~0110011 | 6 | ~1111110 | ~1011111 | 0 | 0 |
| 0 | 1 | 1 | 1 | 5 | ~1011011 | 7 | ~1111110 | ~1110000 | 0 | 0 |
| 0 | 1 | 0 | 1 | 5 | ~1011011 | 15 | ~0110000 | ~1011011 | 0 | 0 |
| 0 | 1 | 1 | 1 | 6 | ~1011111 | 8 | ~1111110 | ~1111111 | 0 | 0 |
| 0 | 1 | 1 | 1 | 7 | ~1110000 | 9 | ~1111110 | ~1111011 | 1 | 0 |
| 0 | 1 | 0 | 1 | 7 | ~1110000 | 21 | ~1101101 | ~0110000 | 1 | 0 |
| 0 | 1 | 1 | 0 | 7 | ~1110000 | 12 | ~0110000 | ~1101101 | 1 | 0 |
| 0 | 1 | 0 | 1 | 6 | ~1011111 | 18 | ~0110000 | ~1111111 | 0 | 0 |
| 0 | 1 | 1 | 0 | 5 | ~1011011 | 10 | ~0110000 | ~1111110 | 0 | 0 |
| 0 | 1 | 1 | 0 | 4 | ~0110011 | 9 | ~1111110 | ~1111011 | 0 | 0 |
| 0 | 1 | 1 | 0 | 4 | ~0110011 | 7 | ~1111110 | ~1110000 | 0 | 0 |
| 0 | 1 | 1 | 0 | 3 | ~1111001 | 6 | ~1111110 | ~1011111 | 0 | 0 |
| 0 | 1 | 1 | 1 | 2 | ~1101101 | 4 | ~1111110 | ~0110011 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | ~0110000 | 3 | ~1111110 | ~0110011 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | ~0110000 | 3 | ~1111110 | ~0110011 | 0 | 0 |
| 0 | 1 | 1 | 0 | 6 | ~1011011 | 10 | ~0110000 | ~1111110 | 0 | 0 |

## *Testbench result :*

1. Do wtime

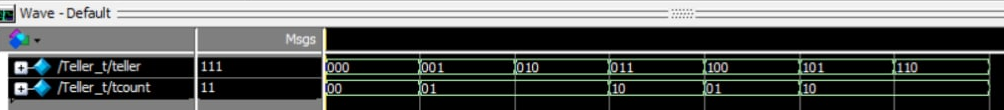
Graphical user interface

Description automatically generated

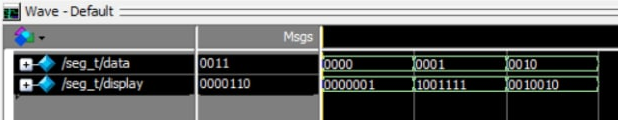
1. Do UpDown\_counter



1. Do Teller



1. Do 7Seg



1. Do unit

