Project report on
Alarm Clock using FPGA

Sreenivasulu N
EE18MTECH11003

Sreeshan S
EE19MTECH01007

Under the guidance of
Dr. GVV Sharma
Dept. of EE, IITH

April 2019

1 Abstract

Hardware implementation of Alarm Clock using Field Programmble Gate Array(FPGA) board is proposed. To make quick processing we are using FPGA board here. Using verilog programming we are implementing hardware. Allocation of input and output ports in FPGA board is necessary. The accuracy of alarm depends upon the how many digit input we are providing to the system. For better understanding we are showing the running time in display using 7-segment display. Finally our output will be some sort of audio signal from a buzzer like normal Alarm.

Keywords:-FPGA,Alarm Clock,7-segment display,Audio buzzer.

2 Introduction

Field Programmable Gate Array has wide variety of uses. Using FPGA makes processing of a program faster by parallel processing.

While FPGAs can be used to solve any computational problem (like computers), and offer more flexibility in terms of the number of things you can do in parallel.

Alarm clock using FPGA is our point of interest here. Using alarm we can remind ourselves and here using FPGA we can get large number of operations on single FPGA.

We will fetch alarm code into a FPGA board. Using some external hardware we will give input to our system. FPGA will give signal after some time which we have given as input. Signal of output will be audio.

3 Block Diagram

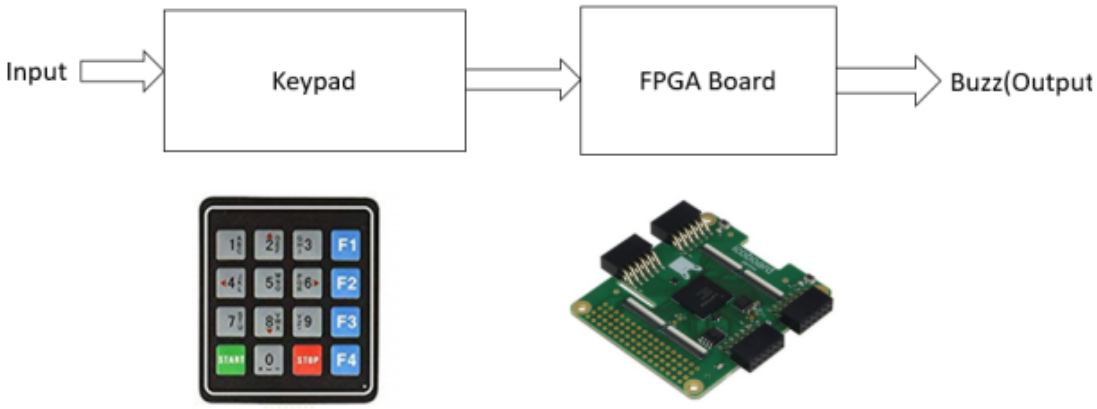

Figure 1: Block diagram

From this block diagram we can clearly understand what is happening here. Input we are taking from external hardware. We are making one keypad circuit which is having digits from 0-9. From this hardware we can give any input according to our requirement.In second we are having FPGA board which means that it will take some input and it will process the input and it will give the output signal after completion of the process. This output can be be a sound signal.

4 Flow Chart

Below Flow chart will explain clearly how this algorithm works on FPGA.

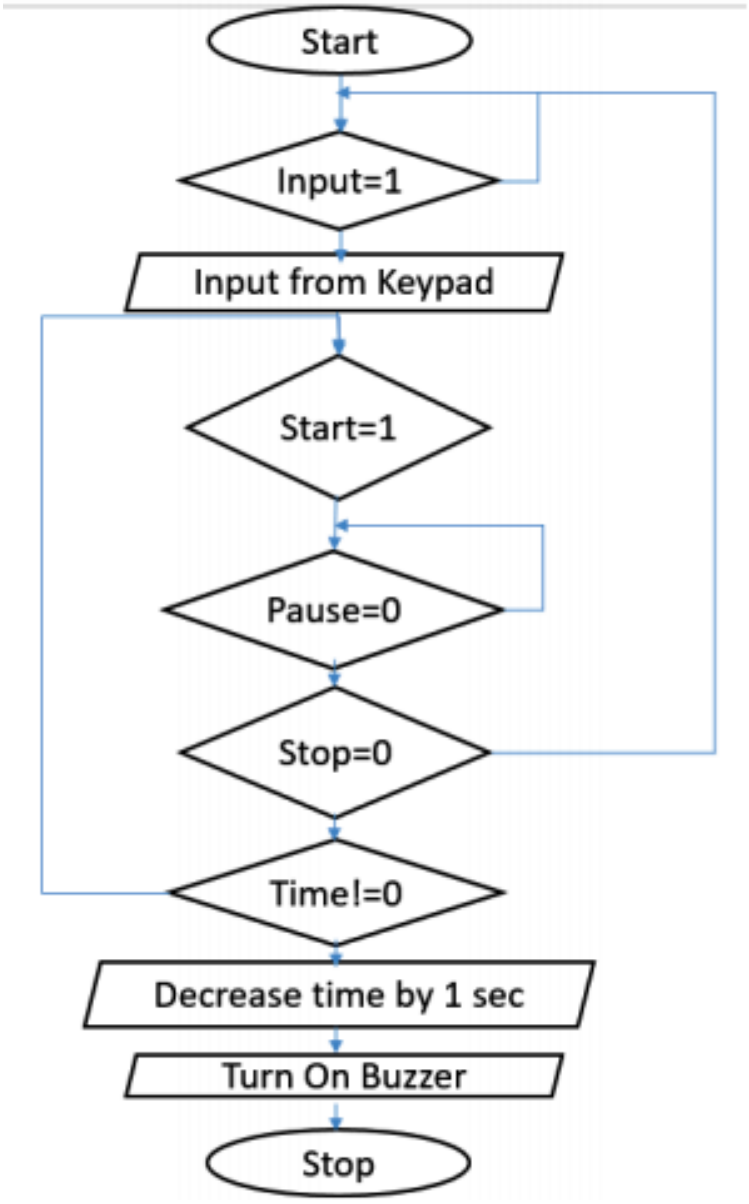

Figure 2: Flow chart

5 Working Process

Before starting we have to fetch the code into FPGA board. If the FPGA board is not in INPUT mode then it will go to starting again and will wait for INPUT mode to activate and on the other hand FPGA is in INPUT mode it will take 6 digit input (i.e. `hh mm ss`). Then FPGA will go into START mode and it will count down the time.

After the START mode if we give STOP mode it will again goes to starting if not after the prescribed time we will get output from the FPGA board to the output, after receiving a signal from the board buzzer will give some sort of audio signal to remind us that alarm time completed. After audio signal we stop or reset using STOP/RESET mode.

6 Result

On performing all these MODES of operations we will get output. Here output is audio signal(`buzzer`).