

***Facultatea*** Automatică și Calculatoare

***Field*** Computer Science

***Discipline*** Digital Systems Design

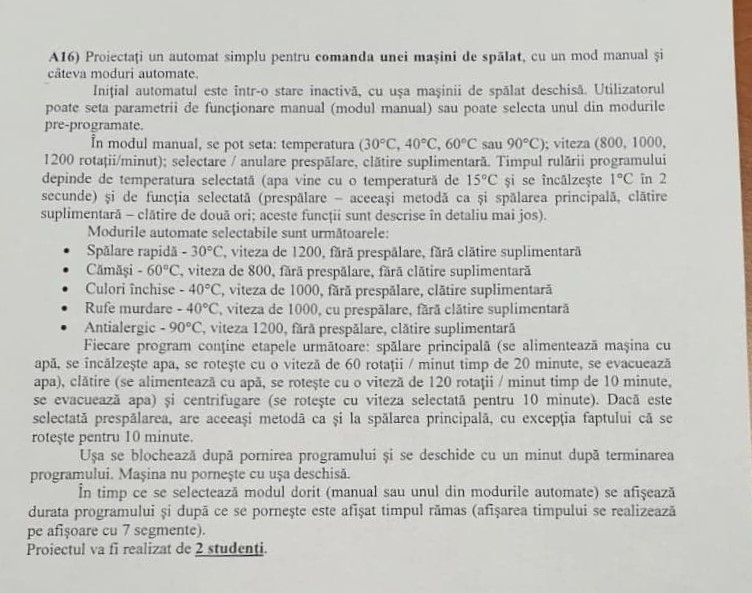
***Project*** Washing Machine Control

***Students*** Bodea Razvan-Marius

Haragas Alexandru

***Group 30412***

***1.Task***

******

-Select between manual/pre-programmed modes(5pre-programmed)

-Select water temperature

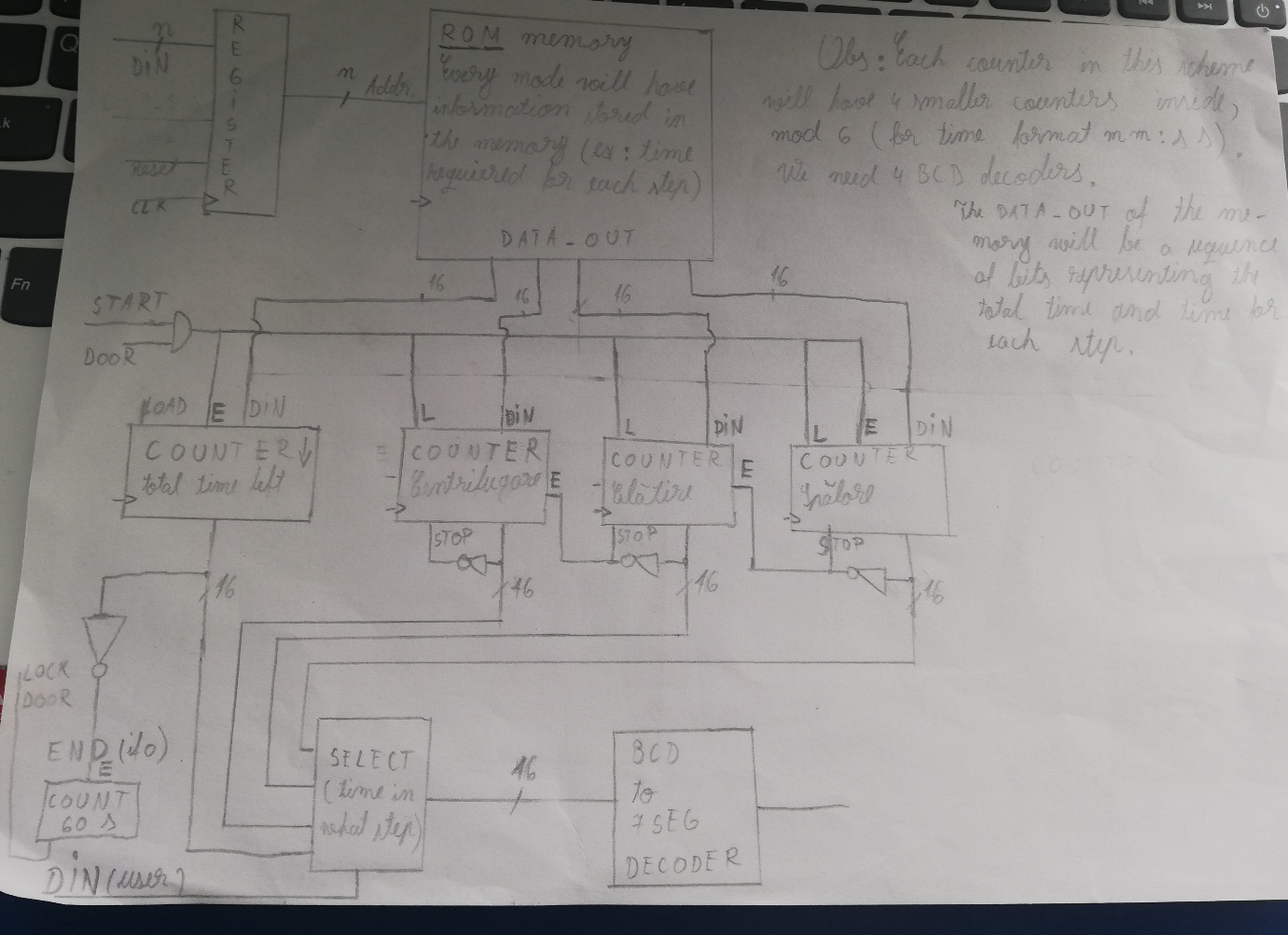
-Select rotation speed

-Select pre-wash & additional rinsing

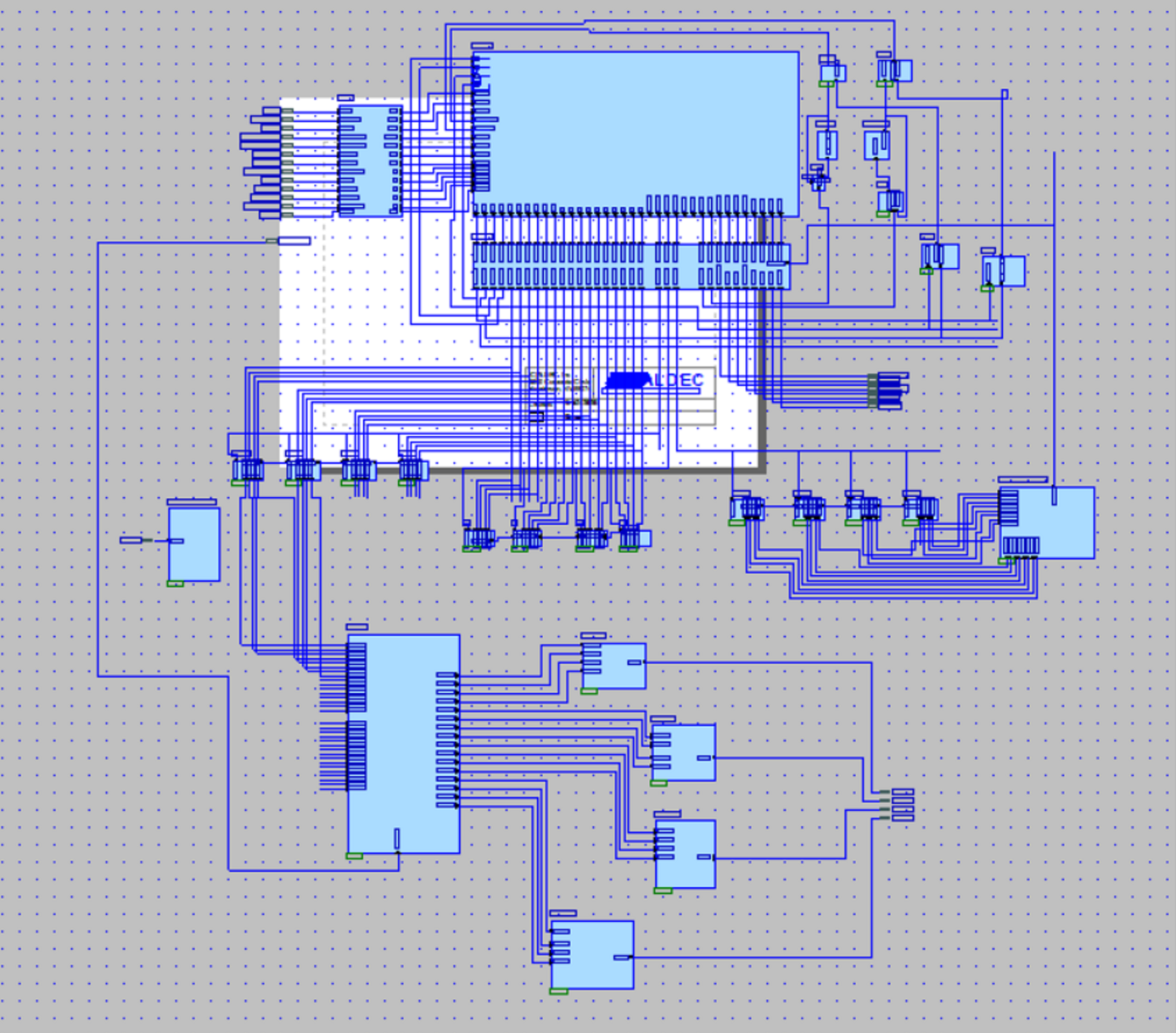
-Simulate the states

***2.Block Scheme***

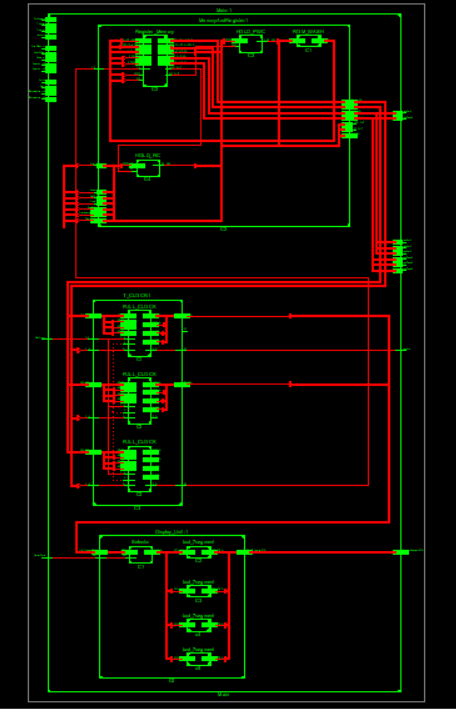
Initial design



Final Design (see Washing Machine Scheme document)



***Generated Schematic***

******

***3.Implementation***

State Diagram

Z14

Z8

Z7

Z6

Z5

A

J

E

Z0

0

1

00

I5

Z15

I0I1

11

Z10

Z9

K

Z1

L

0

I7

F

1

Z16

Z11

B

1

0

I1

G

00

10

I7I8

Z3

Z2

M

0

I6

01

1

C

Z19

Z18

Z17

Z12

Z4

H

D

Z13

Z16

00

00

11

I3I4

I

00

01

0->Default state ->stand-by

->Display total time for the whole selected process

1->Load total washing time

2->Supplying water (1min)

3->Heating water 30C -> 30s

40C -> 50s

60C -> 1m 30s

90C -> 2m 30s

4-> Rotation 10m -> if Pre-Wash selected

20m -> if Normal wash

5->Evacuating water (1min)

6->Check if Pre-Wash selected (if yes, go back to state 2 & signal completion)

->Else load total time for rinsing

7->Supplying water (1min)

8->Rotation (10min)

9->Evacuating water (1min)

10->Check if additional rinsing was selected (if yes, go back to state 7 & signal)

->Else load total time for

11->Centrifugation (10min) with selected rotation speed

12->Keep door locked (1min)

***4.Components***

1.ROM (see the Word Memory document)

->by far the most complicated component (the brain of the washing machine)

->it stores the total times for each step, the next states and the signals for temperature, rotation speed, pre-wash & additional rinse completion, loading for the counters)

->works together with a register in order to distribute information (micro-programming)

->depending on the inputs from the user and additional signals (pre-wash, additional rinse completed, next states), decides what happens next

2.Registers

->Hold information (inputs and the signals from the memory)

3.Counters

->reverse order counters

->6-0 counters for correct time display (format mm: ss)

->Keeping track of time elapsed

->Signaling the completion of a state

->Used to feed information to display

4.BCD to 7 segment decoders

->Used to display the information on the available segments on the FPGA board

->Transforming the clock information to visible numbers

5.Selector

->Used to switch the display between total time left & time left in the current state

6.Additional Logic (D flip-flops, NAND, AND, OR gates)

->Holding information about pre-wash, additional rinse completion

->Deciding when the state/process ended

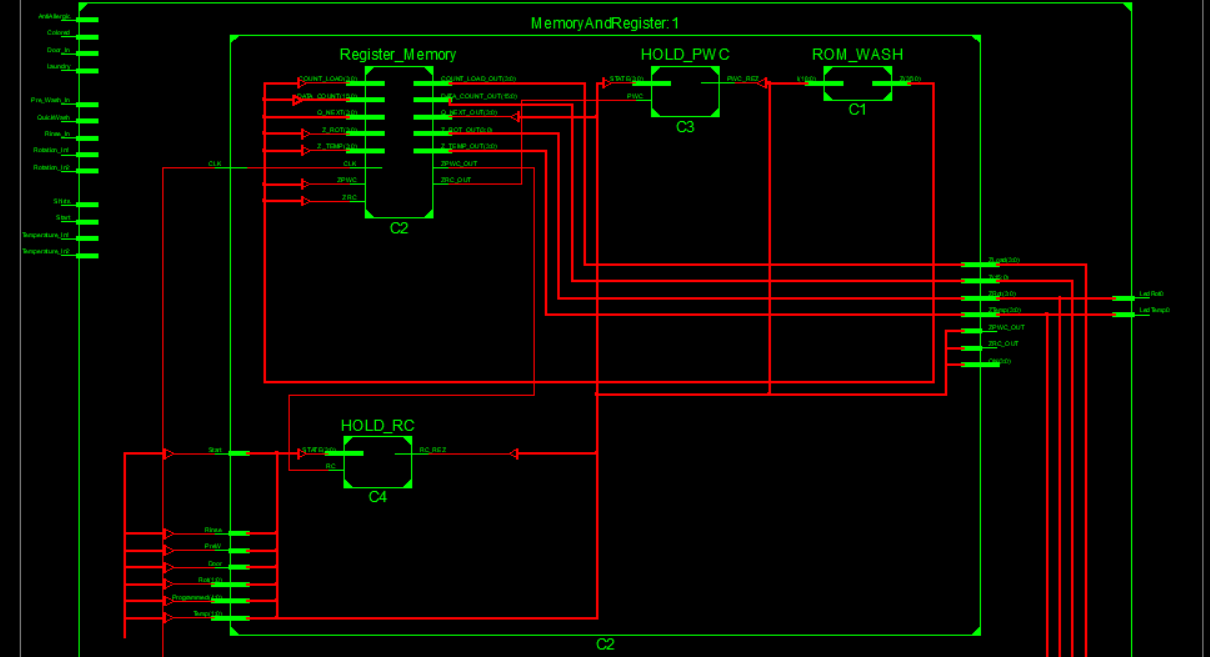
**7.!!!!Debugger**

->used for controlling the flow of the states

->a component we found out was necessary when we started simulating because the machine would get stuck in the 1st state because the information wouldn’t get loaded onto the clock.

->it checks if the state is 0 or 1 and sends a signal to the memory to execute the next step if this is true

**Command Unit**

****

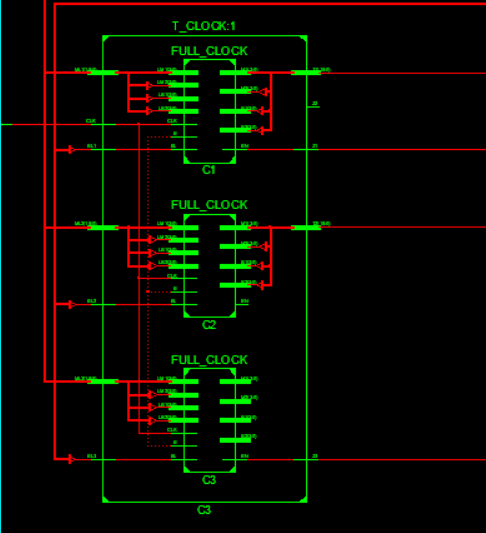
ROM Memory + Register + 2components for holding information (remembers if pre-wash or additional rinse have been completed).

The information is stored in the memory and is accessed by the register when it receives a clock signal from the execution unit (the counter that counts time left in the current state will send a signal to the register when it reaches 0).

The Hold components will signal ‘1’ if the specific action occurred and only when the memory need to have information about this (ex: state 10), else the output is’0’.

We have a total of 819 possible cases (divided into sub-groups for each specific set of inputs).

**Execution Unit**



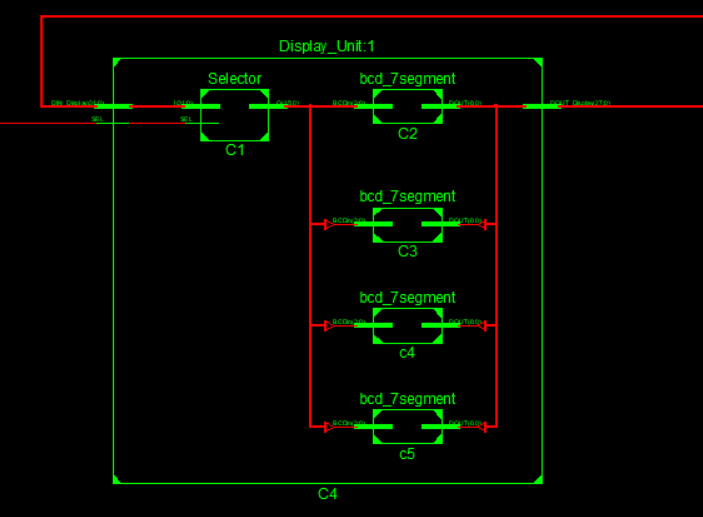
3 Main counters (built from 4 smaller counters) are used to keep track of time passed.

They receive the time as input bits from the command unit, along with signals that choose which counter is loaded and when.

1 Counter keeps track of total time left and when it reaches 0, it signals that the washing finished (by lighting a LED)

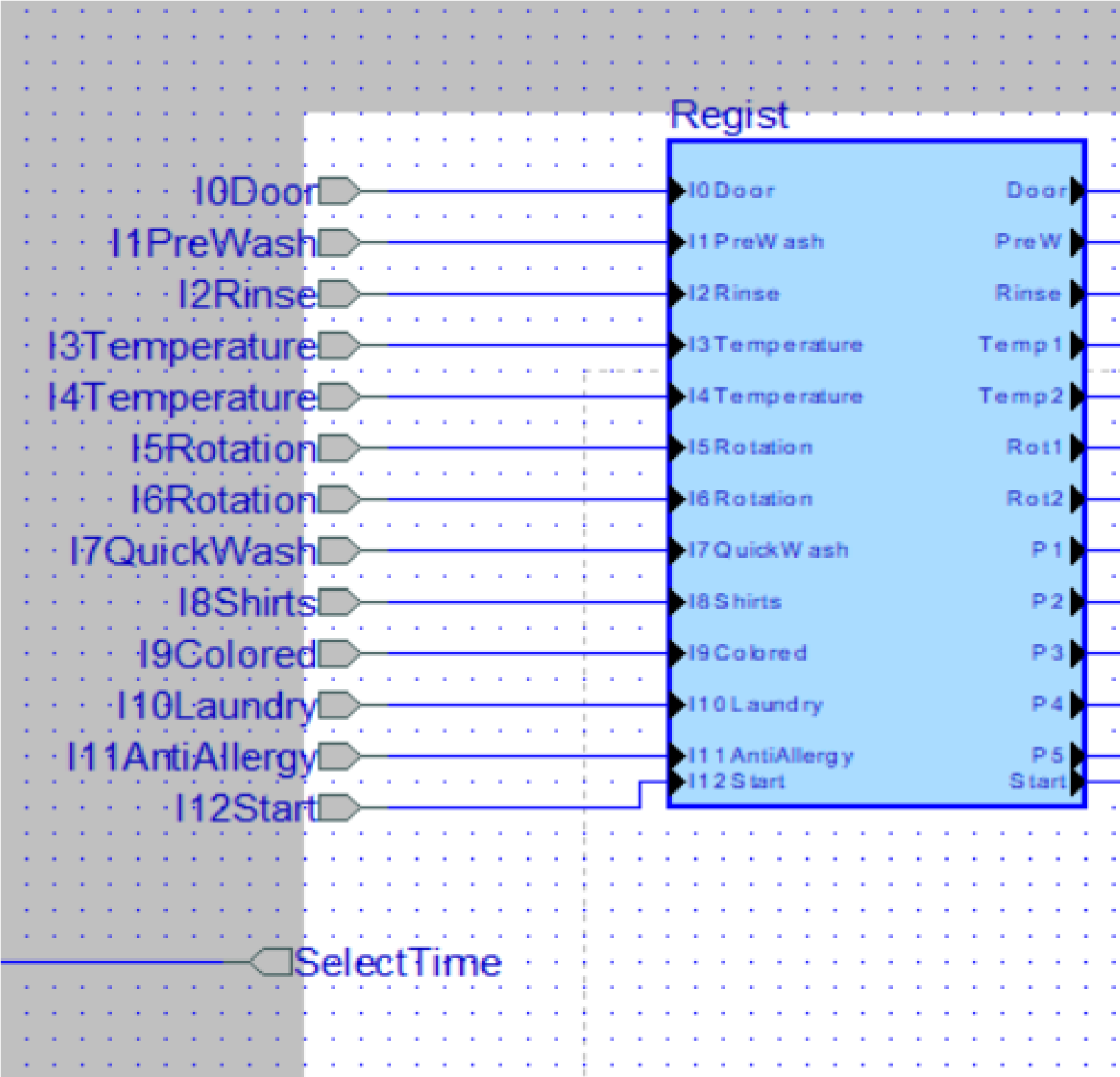
1 Counter keeps track of time in the time left in the current state (we have 3 main states, Washing, Rinsing and Centrifugation)

1 Counter keeps track of time left in the current sub-state of the main states (heating water, rotating with a specific speed, evacuating water etc.) This counter sends a signal when it reaches ‘0000’ that is used as a clock signal for the command unit to load the next state in.

**Display unit**

We use a selector with the Select input directly from the user, that lets the user choose between what time is displayed (total time left, or time left in current state)

4 segment decoders are used to transform the information from the counter into visible numbers on the FPGA board.

***5.Inputs ***

I0->Door Signal (the process only starts if the door is closed)

I1->I6 Inputs from the user used for manual programming

I7->I11 Inputs for the pre-programmed modes (choosing 1 is as easy as pressing the button for the mode and pressing start)

I12->Start Process

I13->Switch display between total time and time left in the current state

OBS: For the FPGA board, each input is represented by a switch and the order of the bits stays the same as presented from left to right.

***Outputs***

Led1->Led4 Water temperature (if it is heating)

1->30C

2->40C

3->60C

4->90C

Led5->Led8 Rotation speed

5->80 rot/min

6->800 rot/min

7->1000 rot/min

8->1200 rot/min

Led9 -> End Signal (Washing ended)

Segments 1->4: Display time left in the format mm: ss

OBS: For the FPGA board, the outputs are represented by LEDs (except the segment ones), and the order stays the same as presented, from left to right.

***Some Interior Signals***

QN in-out signals used for keeping track of the current and the next state of the process

Z0-15 time to load on the counters

Zload0-2 2->load clock structure 1 (total time)

1->load clock structure 2 (state time)

2->load clock structure 3 (process time)

ZPWC&ZRC signals if pre-wash and additional rinsing are complete

***6.Why this solution?***

We chose this solution because microprogramming seemed like and interesting way to approach this problem

We wanted an easy way to select the pre-programmed modes (just the press of a button) because they should be as obvious as possible for a user

Using a memory greatly reduced the number of other components required:

We used a total of 12 counters just for the display convenience (we could use only 8 if we remove the current state time display)

***7.Future Improvements***

1 problem of the design : the counters have a 1 second difference so the last state(locking the door for 1 min) will last only 58 seconds.

Memory is the main bottleneck of a system so improving the memory would make a faster, better and in some cases cheaper implementation

Finding compatible states and signals and grouping them together would reduce the number of bits/signals/wires used for the memory and would give a better speed.

Some parts of the machine could be removed if the user requested it

***8.User Manual***

You will have 14 possible switches to flip (presented from left to right)

1.Door (this signals that the door is closed; in a real case this would be automatically detected).

2.PreWash (if you want to pre-wash your clothes this will repeat the washing stage of the machine with a new rotation time of 10 minutes)

3.Rinse (if you want additional rinsing this will repeat the rinsing stage)

4&5. Temperature control: for manually selecting the temperatures the codes are:

4 off 5off ->30C

4 on 5 off ->40C

4 off 5 on ->60C

4 on 5 on ->90C

6&7. Rotation speed control

6 on 7 off ->800 rot/min

6 off 7 on ->1000 rot/min

6 on 7 on ->1200 rot/min

8.Pre-Programmed Quick wash

30C, 1200rot/min, no pre-wash, no additional rinsing

9.Pre-Programmed Shirts

60c, 800rot/min, no pre-wash, no additional rinsing

10.Pre-Programmed Colored

40C, 1000rot/min, no pre-wash, with additional rinsing

11.Pre-Programmed Laundry

40C, 1000rot/min, with pre-wash, no additional rinsing

12.Pre-Programmed Anti-Allergic

90C, 1200rot/min, no pre-wash, with additional rinsing

13.Start

14.Select What Time Is Displayed

->switch between total time left and time left in the current state

**Important:**

Only one switch for pre-programmed modes can be active at once.

If you pick a pre-programmed mode, you are not allowed to modify the other switches except start and door.

If you set the machine manually, you are not allowed to also pick a pre-programmed mode.

After you press start, do not change the switches!

Doing one of the above will result in a failure.

After you selected the desired program, press start.