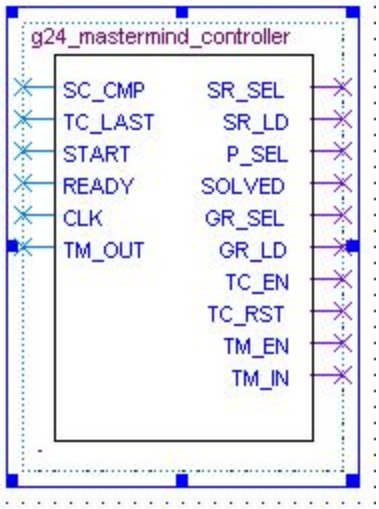
Lab 3: Group 24
Michael Curtis - 260475694
Yusaira Khan - 260526007

g24_mastermind_controller

Overview

Our mastermind controller deals with updating and switching signals within our datapath based on the outputs of that same datapath. Specifically it iterates through the possibility table for each guess and updates the table to signify whether or not that value is possible given the output of the hidden patterns score. The controller controls datapath which involves the guess register, score register, 3 selectors, the possibility table, and the score comparator. Once we find a matching score the controller has done its job.



Inputs and Outputs

Port Name	Port Type	Signal Type	Description
SC_CMP	In	std logic	Tells the controller whether the current scores being compared are the same or not.
TC_LAST	In	std logic	Indicates the last value in the possibility table has been reached for the current guess iteration.
START	In	std logic	Starts the mastermind game when set to ‘1’ and asynchronously resets the controller when set to 0.

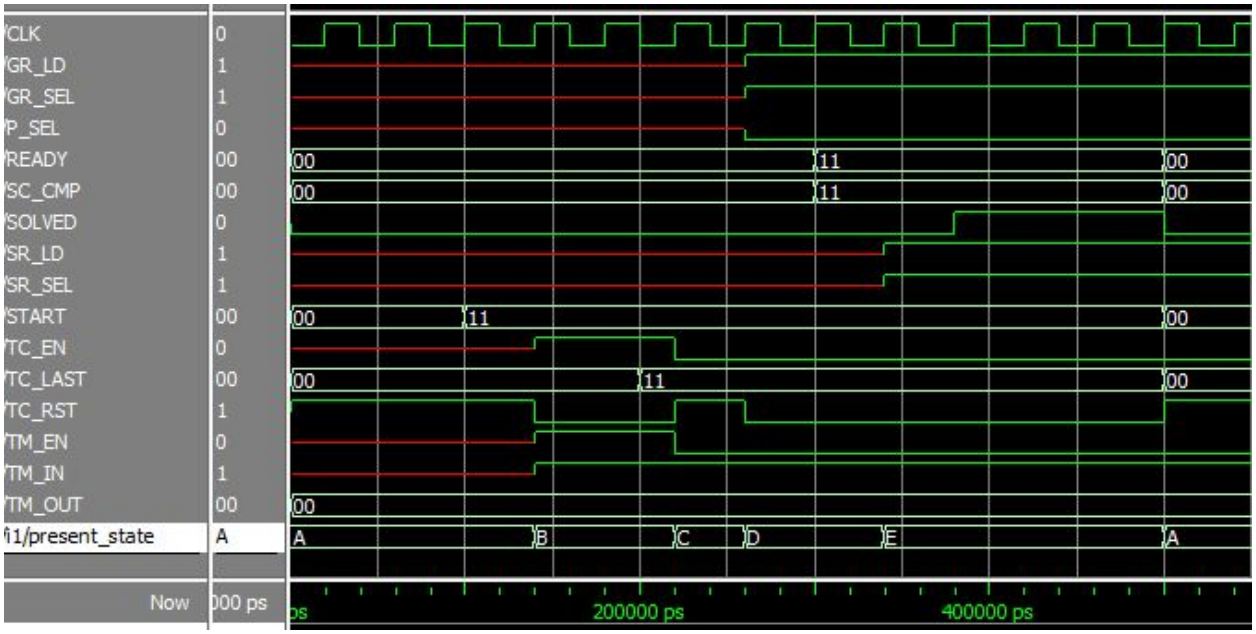
READY	In	std logic	Indicates the hidden pattern has been set and that player is ready.
TM_OUT	In	std logic	Checks to see if the current possibility table output is valid.
SR_SEL	Out	std logic	Switches the score comparison value B between (4,0) and whatever the current the current score module output is.
SR_LD	Out	std logic	Load the current score module output to the score register.
P_SELECT	Out	std logic	Decides whether we feed the hidden pattern or TM_ADDR value to the scoring module.
SOLVED	Out	std logic	Is triggered on high when we have solved the mastermind problem and guessed the correct colour pattern.
GR_SEL	Out	std logic	Selects whether or not we want the initial 0011 value or the TM_ADDR output will be sent to the guess register.
GR_LD	Out	std logic	If high this signals enables the guess register to store the value being sent to it.
TC_EN	Out	std logic	Allows the table to continue incrementing on each clock cycle.
TC_RST	Out	std logic	Restarts the possibility table back to its initial value of “000000000000”.
TM_EN	Out	std logic	Allows writing to the possibility table.
TM_IN	Out	std logic	If the TM_EN is set to high we write the TM_IN value to the current possibility table index.

Function:

The controller has a finite state machine. Initial state is state A. State A waits for start to be 1 and sets Solved as 0. When start is 1, state is B and instructions for table memory setting all bits to 1 are given. When this is finished, that is TC_LAST is ‘1’, Table memory counter is reset and state is D where the guess is compared against the pattern given, waiting for ‘Ready’ to be 1. When Ready is 1 the score is compared to 4,0 in state E by setting SR_LD and SR_SEL. If the score matches then the machine remains in that state till start is set to 0. If the score doesn’t match 4,0 then the state if F. In states F and G, the circuit removes all patterns that don’t match

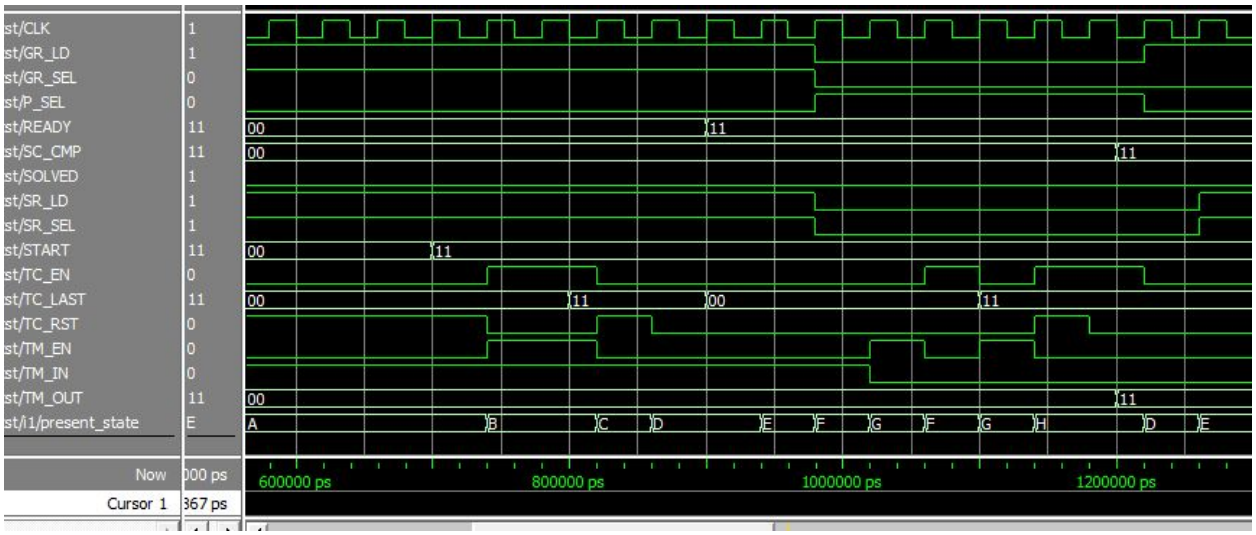
the score, once that is done, in State H, a valid pattern is searched for. The controller then returns to state D to check if the score matches

Testing Discussion



The circuit was tested by checking if the finite state machine works. the first diagram shows the controller working when the initial guess is correct. All the inputs have 2 bits since a forloop wasn't used in the testing. The circuit responds correctly to all the inputs. In state A, everything is reset with TC_RST and TC_EN. In state B, the memory is written to with TC_EN and TM_EN and TM_IN. In state C, the counters are reset again. The controller then tests the pattern and sees

that it's right in states D and E and remains in state E till the circuit is reset to state A with start set to 1.



This diagram shows the controller work when the initial guess is not 1. The circuit removes all guesses that don’t correspond to the score of the initial pattern.

Timing Performance

Slow model Fmax:355.87 MHz
Slow model setup time: -1.810

FPGA Resource Utilization
Total logic elements 34 (<1%)