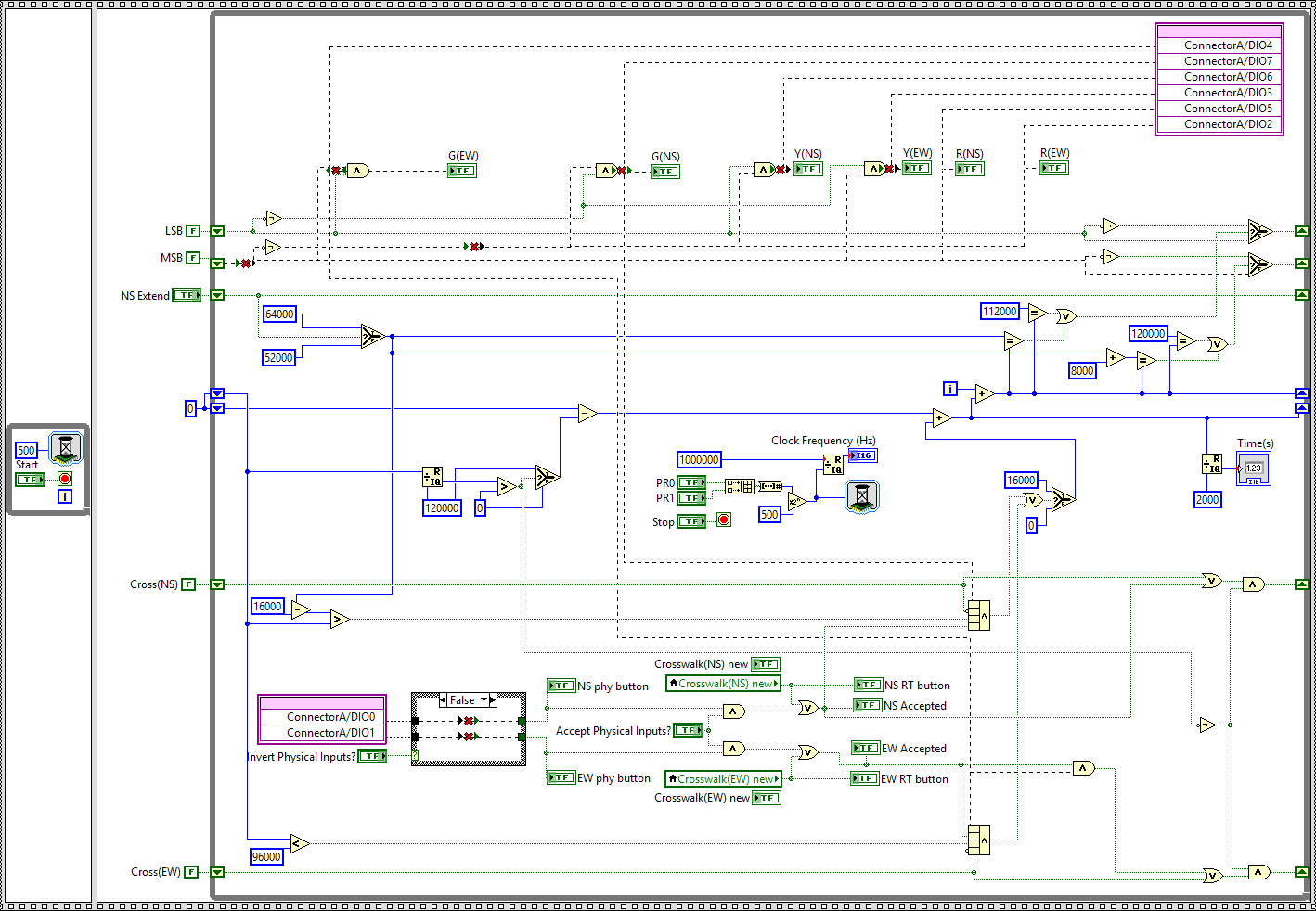
## main\_FPGA.vi

This VI runs on the FPGA and is responsible for controlling stoplight operation and interfacing physical and digital I/O necessary to alter stoplight functionality.

### Block Diagram

1 2

­

7

11

4

10

8

9

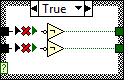
6

7.1

7.2

5

3

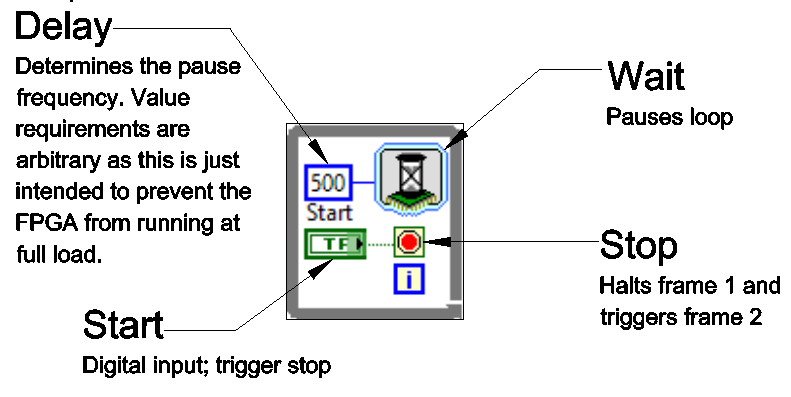


### Functional Subunits

1: Frame 1 – Prevents execution of frame 2 until stoplight operation is desired

2: Frame 2 – Contains the stoplight loop

3: Hungry Loop – The mechanism by which frame 1 is maintained and frame 2 is held off

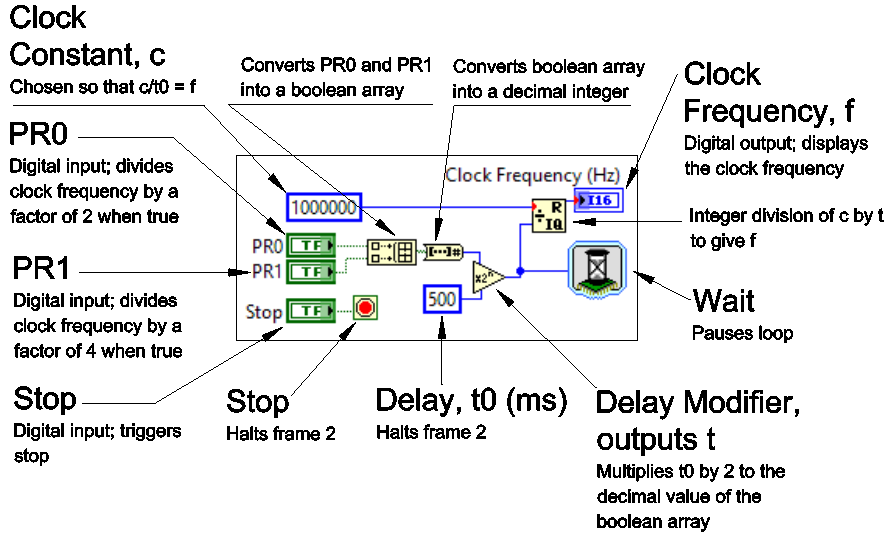


4: Stoplight Loop – Contains the logic for stoplight operation and handles I/O in a manner

corresponding to that specified in the lab instructions.

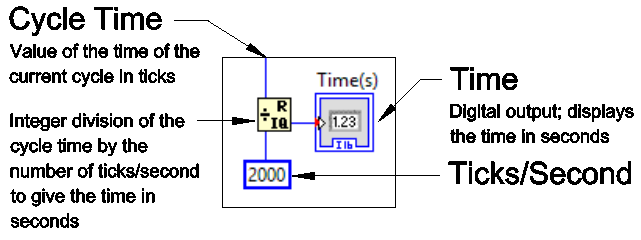
5: Clock – Determines the frequency of the loop clock in response to PR0 and PR1, with a

default frequency of 2kHz



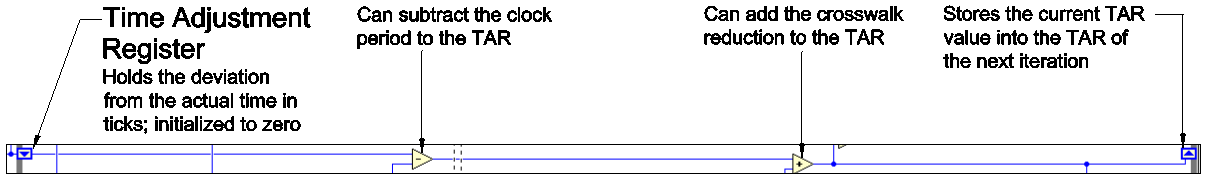
6: Time – Converts the number of iterations (here being the equivalent to clock ticks) into

seconds. 2000 ticks/second.

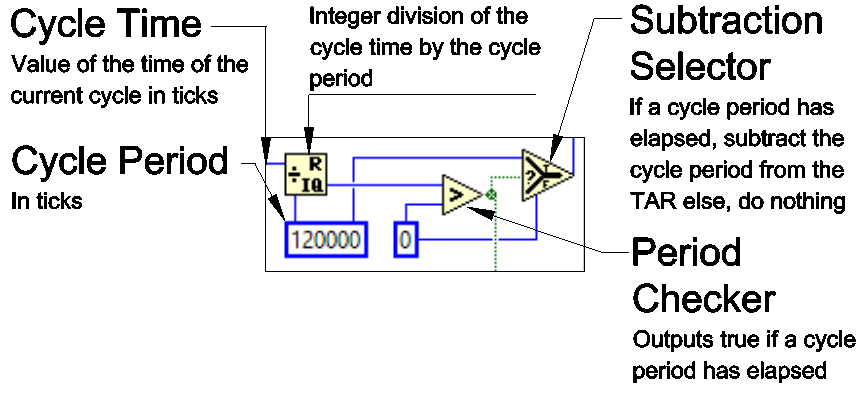


7: Time Adjustment Register (TAR) – Modifies the transmitted value of the clock register in

response to the completion of a cycle of a crosswalk request.

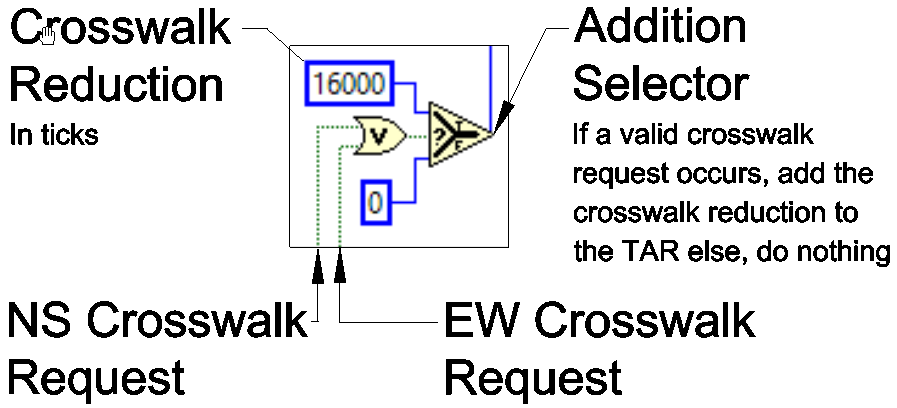


7.1: Cycle Adjust – Subtracts the cycle period (120s) from the TAR on the completion of a cycle



7.2: Crosswalk Adjust – Adds the crosswalk reduction (8s) to the TAR on a valid crosswalk

request.



8: Time Register – Adds the value in the TAR to the number of iterations (right) to get a cycle

specific virtual time (left).

9: Cross Request Validator – Approve the transmission of a crosswalk request in accord with

the constraints specified in the lab instructions. Accepts physical and digital crossing

requests.

10: Light Switcher – Determines the periods of light operation in response to the virtual time and

the value of the NS Extend Register. Operated via the toggling of the LSB and MSB

registers.

11: Light Logic – Determines the output configuration of the lights in response to the values of

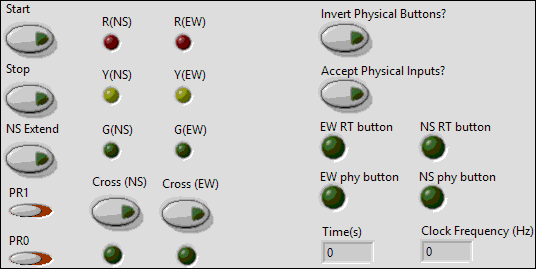
the LSB and MSB register in accord with the truth table specified in the lab instructions.

Outputs digitally and physically.

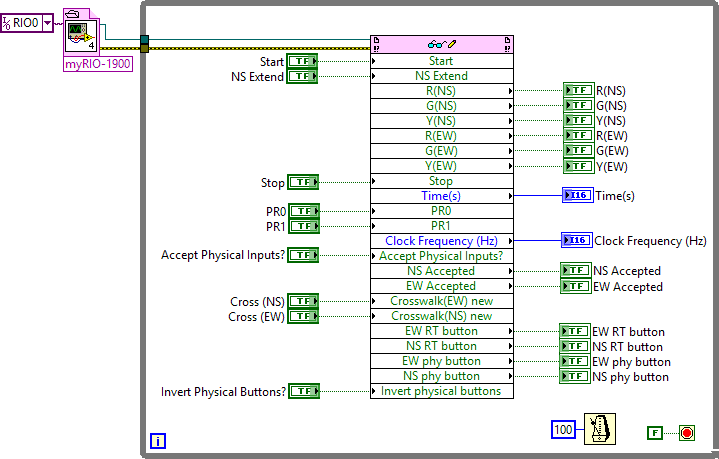
## main\_RT.vi

This VI runs on the RT module and is responsible for routing physical and digital I/O to the targeted FPGA bitfile.

### Front Panel



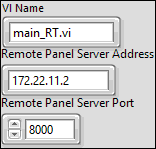
### Block Diagram



## Remote Panel Methods Client.vi

This VI illustrates how to programmatically connect to a remote front panel. A connection is established to the specified VI and control can also be requested. Once control is requested, the VI can only be run once since the server will close the connection automatically after one run.

### Front Panel



### Block Diagram

