Erick Chacon
echacon@uncc.edu

Demonstration points:

Requirements:

Req. 1 The obstacle avoidance program from lab 1 will be used. _____

Req. 2 Two IR sensors will be used. _____

Req. 3 A LabVIEW program will be written that will use the IR sensors to detect when the robot is about to fall off of an edge, and reverse and change direction of the robot. _____

Req. 4 Robot will operate autonomously. _____

Req. 5 Requirements 1 and 3 will be implemented concurrently. _____
(Robot should adopt obstacle avoidance behavior, and only use edge detection code when the IR sensor detects an edge. After robot has reversed and changed direction, robot should continue with obstacle avoidance behavior.)

Req. 6 Robot should continue running obstacle avoidance behavior after it has detected an edge and changed course. _____

General Learning Objectives:

The general learning objectives of this lab was to provide the users' the experience of programming edge detection feature with concurrent obstacle avoidance behavior. To achieve this functionality the IR sensors & ultrasonic sensor were connected to the sbRIO and there outputs were used in a closed loop system to maneuver the robot.

General Steps Needed to Complete the Lab:

The general steps needed to complete this lab were to follow the tutorial provided to us. The IR sensors on the robot frame were connected to the sbRIO port pins. A sub VI called RoamingSubVI was created from the Roaming VI to handle roaming and edge detection. This sub VI was further called in the main VI called the RoamingWithEdgeDetection.

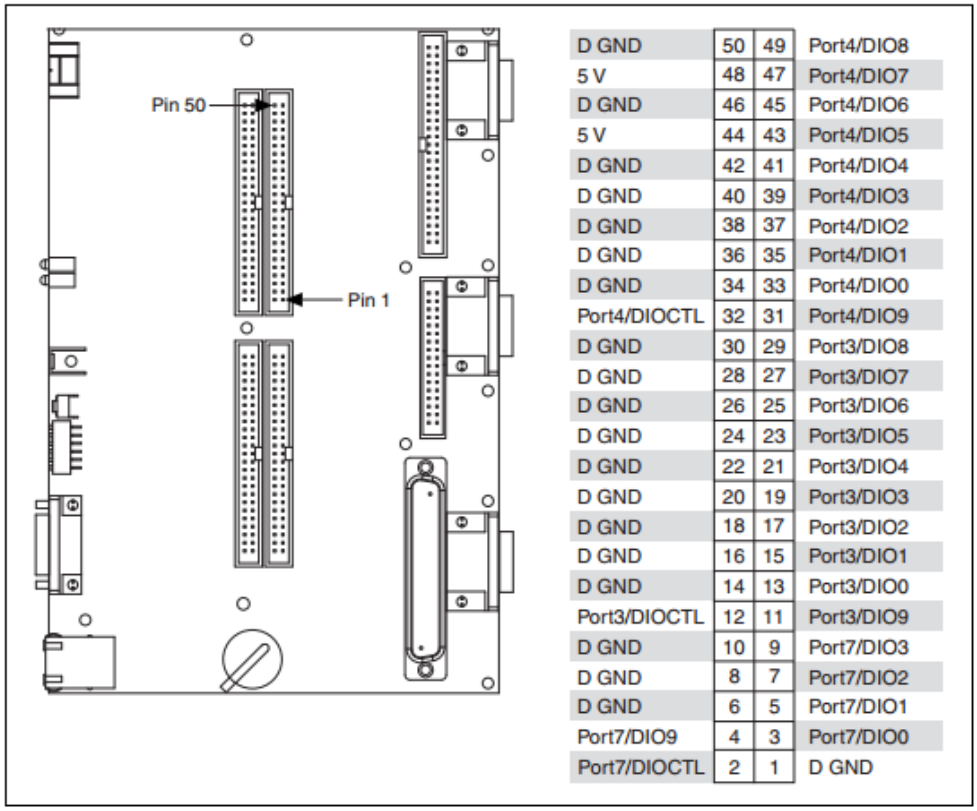
Procedure / Detailed Steps to Complete the Lab:

The lab experiment was broken down into 6 steps:

1. The DaNI 2.0 robot was connected to the computer using a LAN cable. It was ensured that the motors were switched off and only the main supply was on. The successful connection of the robot was indicated by the LAN icon on the windows task bar.

2. LabVIEW was then opened and then using the Robotics 2011 plugin, the hardware was detected. Furthermore, proper function of the motor, ping sensor and servos was confirmed using the Roaming VI. The Roaming VI was left open in the background.

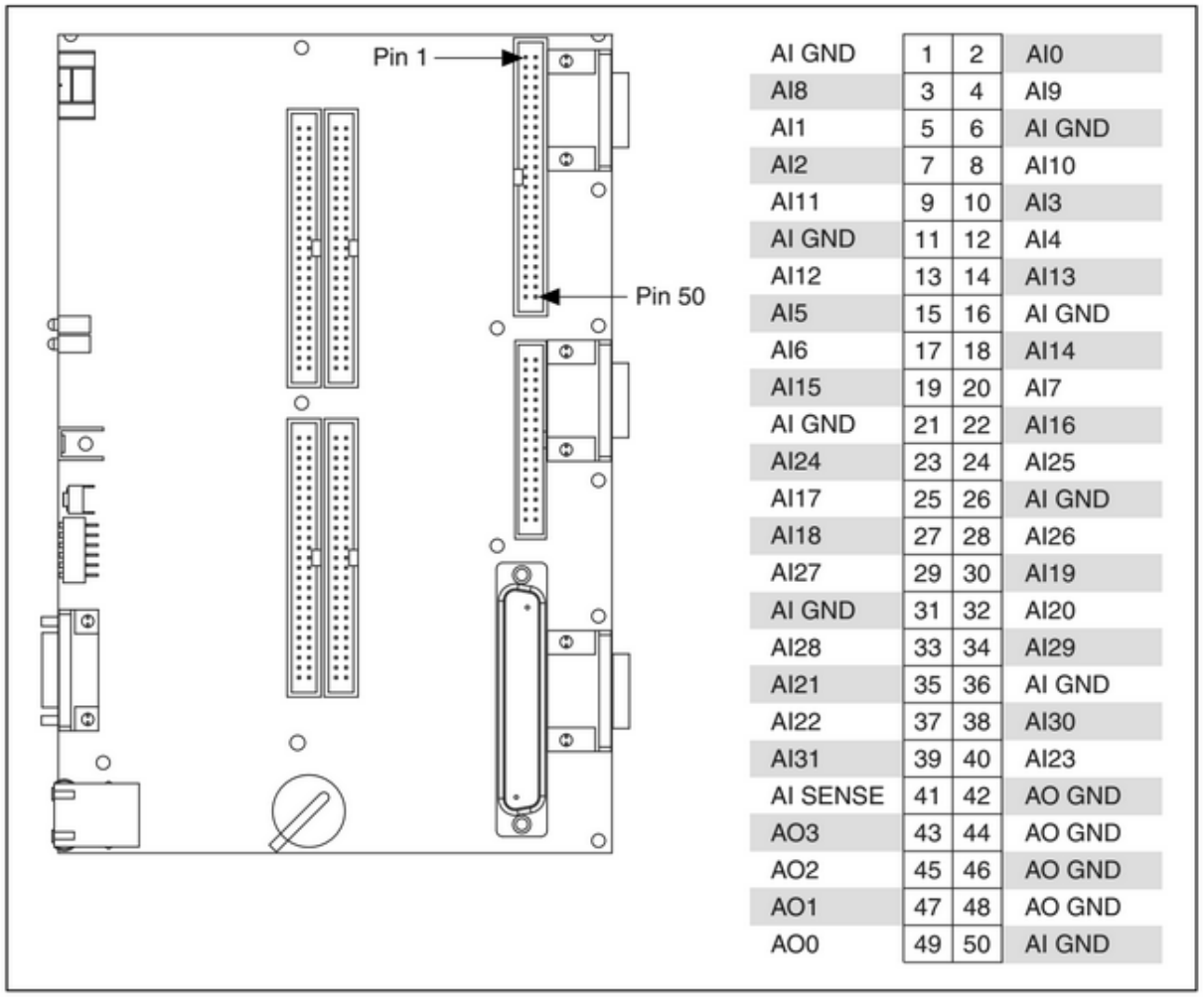
3. Vcc wires from the sensors were connected on connectors P5 on port pins 48 (VccR) & 44(VccL). And the ground from right sensor was connected to pin 50(DGndR) and that from left was connected to pin 46 (DGndL).The connections were made as illustrated by Figure1.



D GND	50	49	Port4/DIO8
5 V	48	47	Port4/DIO7
D GND	46	45	Port4/DIO6
5 V	44	43	Port4/DIO5
D GND	42	41	Port4/DIO4
D GND	40	39	Port4/DIO3
D GND	38	37	Port4/DIO2
D GND	36	35	Port4/DIO1
D GND	34	33	Port4/DIO0
Port4/DIOCTL	32	31	Port4/DIO9
D GND	30	29	Port3/DIO8
D GND	28	27	Port3/DIO7
D GND	26	25	Port3/DIO6
D GND	24	23	Port3/DIO5
D GND	22	21	Port3/DIO4
D GND	20	19	Port3/DIO3
D GND	18	17	Port3/DIO2
D GND	16	15	Port3/DIO1
D GND	14	13	Port3/DIO0
Port3/DIOCTL	12	11	Port3/DIO9
D GND	10	9	Port7/DIO3
D GND	8	7	Port7/DIO2
D GND	6	5	Port7/DIO1
Port7/DIO9	4	3	Port7/DIO0
Port7/DIOCTL	2	1	D GND

Figure 1: Pinout of I/O Connector P5, 3.3 V Digital I/O
Source: http://www.ni.com/pdf/manuals/375052c.pdf

The sensor output from the right sensor was connected to the analog I/O ports AI0 and that from the left was connected to AI9. The connections were made as illustrated by Figure2.



AI GND	1	2	AI0
AI8	3	4	AI9
AI1	5	6	AI GND
AI2	7	8	AI10
AI11	9	10	AI3
AI GND	11	12	AI4
AI12	13	14	AI13
AI5	15	16	AI GND
AI6	17	18	AI14
AI15	19	20	AI7
AI GND	21	22	AI16
AI24	23	24	AI25
AI17	25	26	AI GND
AI18	27	28	AI26
AI27	29	30	AI19
AI GND	31	32	AI20
AI28	33	34	AI29
AI21	35	36	AI GND
AI22	37	38	AI30
AI31	39	40	AI23
AI SENSE	41	42	AO GND
AO3	43	44	AO GND
AO2	45	46	AO GND
AO1	47	48	AO GND
AO0	49	50	AI GND

Figure 2: Pinout of I/O Connector J7, Analog I/O
Source: http://www.ni.com/pdf/manuals/375052c.pdf

A new FPGA file project file called s Lab3EdgeDetection was created. From the project tree a new VI as created and saved as RoamingWithEdgeDetection. This served as the main VI of the project. In the back panel of this VI a three frame flat sequence was created. The first frame was left vacant for calling the subVI as instructed by the tutorial. The second & third frame were inputted with the Write DC Motor Block .The block in the second frame was configured to make the robot move backwards while the block in the third frame was configured to take a 90 degree turn. To re iterate this behavior a while loop was placed around this sequence. The initialize & close blocks for the FPGA were added outside this while loop. The relevant connections were made. The connections were made as illustrated by Figure3.The connections on the sbRIO are further displayed in Figure 4.

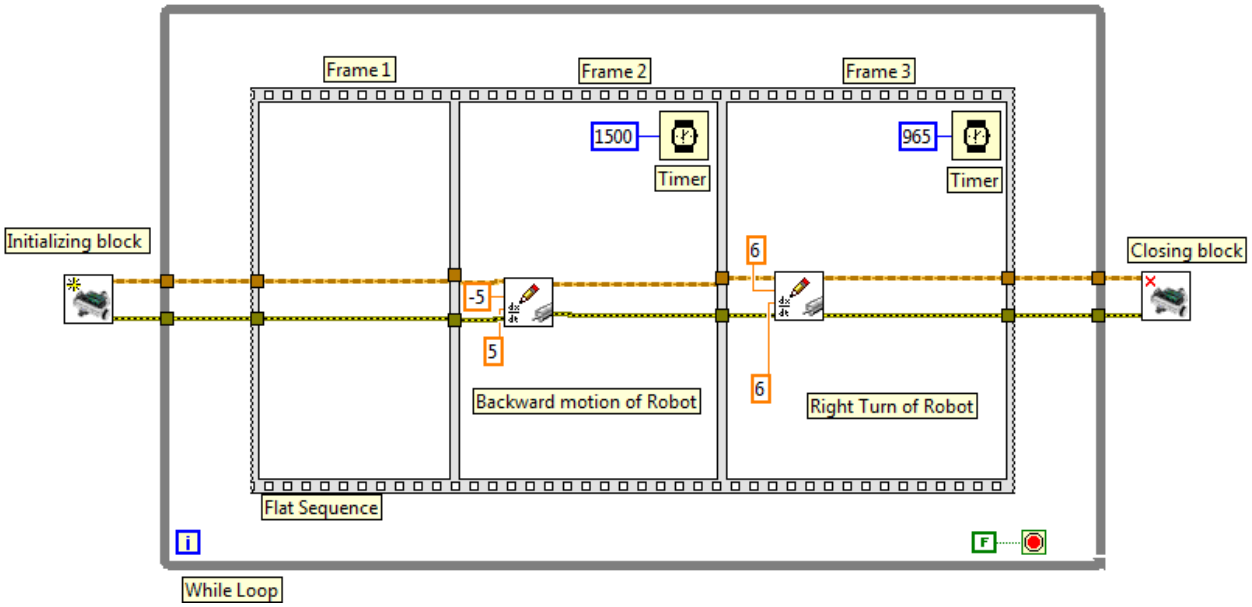


Figure 3: RoamingWithEdgeDetection without sub VI

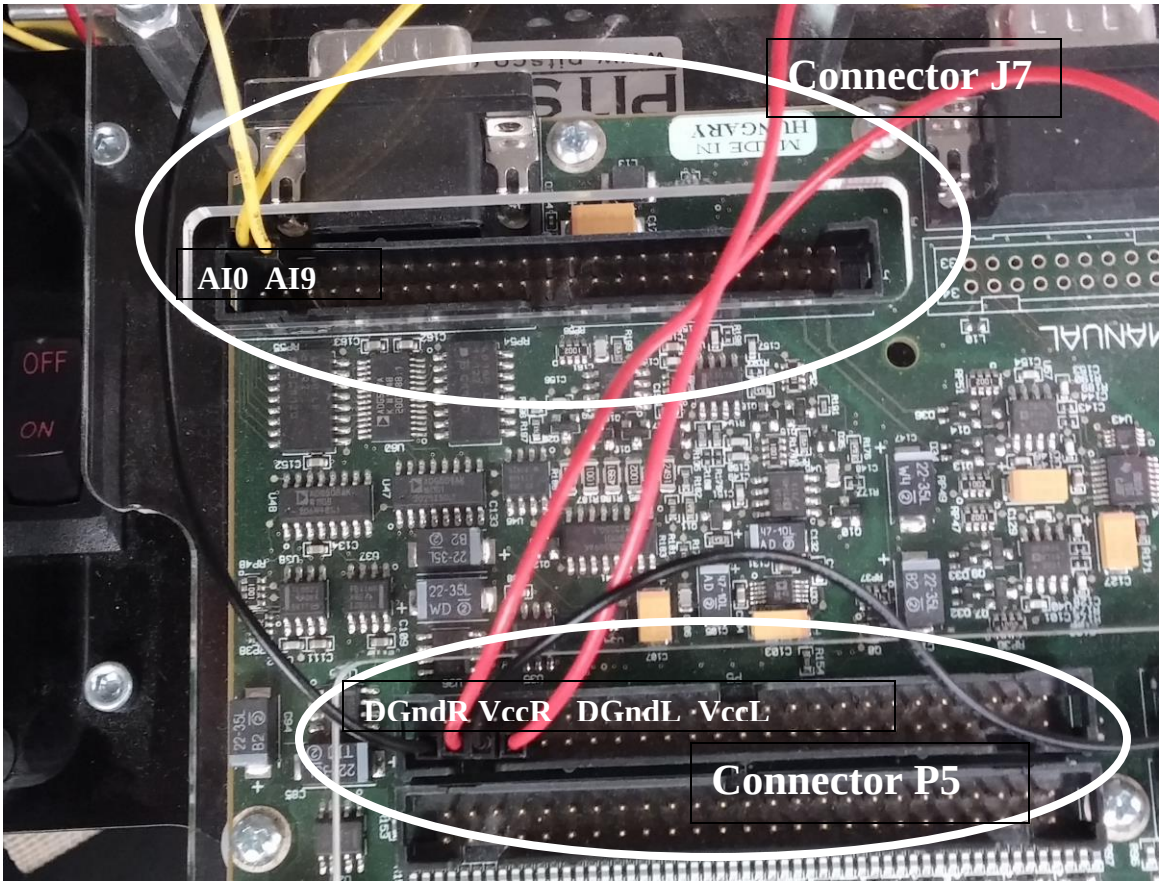


Figure 4 Sensor connections on the sbRIO

4. Later, the back panel from the Roaming VI was copied into a new VI named as RoamingSubVI. The initialize & close blocks for the FPGA were deleted from this copied VI. The host and error wires connected to the initialize block were connected with control block while those connected to the close block were connected to indicators. This loop would function till an edge is detected. Hence provision has to be made to incorporate for the sensor readings when an edge is detected. As instructed by the lab 4 assignment sheet it can be noted that the value outputted by the sensor drops below 1.2 volts when an edge is detected. This principle was used to exit the RoamingSubVI VI. Thus to read the analog values two Read AI blocks were added to this VI. By creating a constant for each of these blocks, the port pin to which each of the sensor output was physically connected was selected. Thus the left motor was connected to AI9 whereas the right pin was connected to AI0.The values from these two sensors were checked to see if they were less than 1.2 volts. If either of the sensor value fell below this threshold then an edge is detected. To implement this logic the sensor values were compared with a 1.2 constant. The output of both these Less than blocks was OR (OR Gate 1) and provided as one of the input to OR Gate 3. The inputs from the constant and status were fed to OR Gate 2.The output of this OR Gate 2 was given as the second input to OR gate 3. The OR gate 3 output was connected to a stop symbol. This algorithm can be stated as follows: If either of the sensor value goes below 1.2 V or the status pin is held high then the RoamingSubVI would stop execution and the control would be shifted to the next frame (Frame 2) of the flat sequence. Thus the status of both the IR sensors & ping sensor were taken into account to maneuver the robot. Thus the robot would back for a fixed duration of time and take a 90 degree right turn. The completed RoamingSubVI VI can be illustrated as per Figure 5.

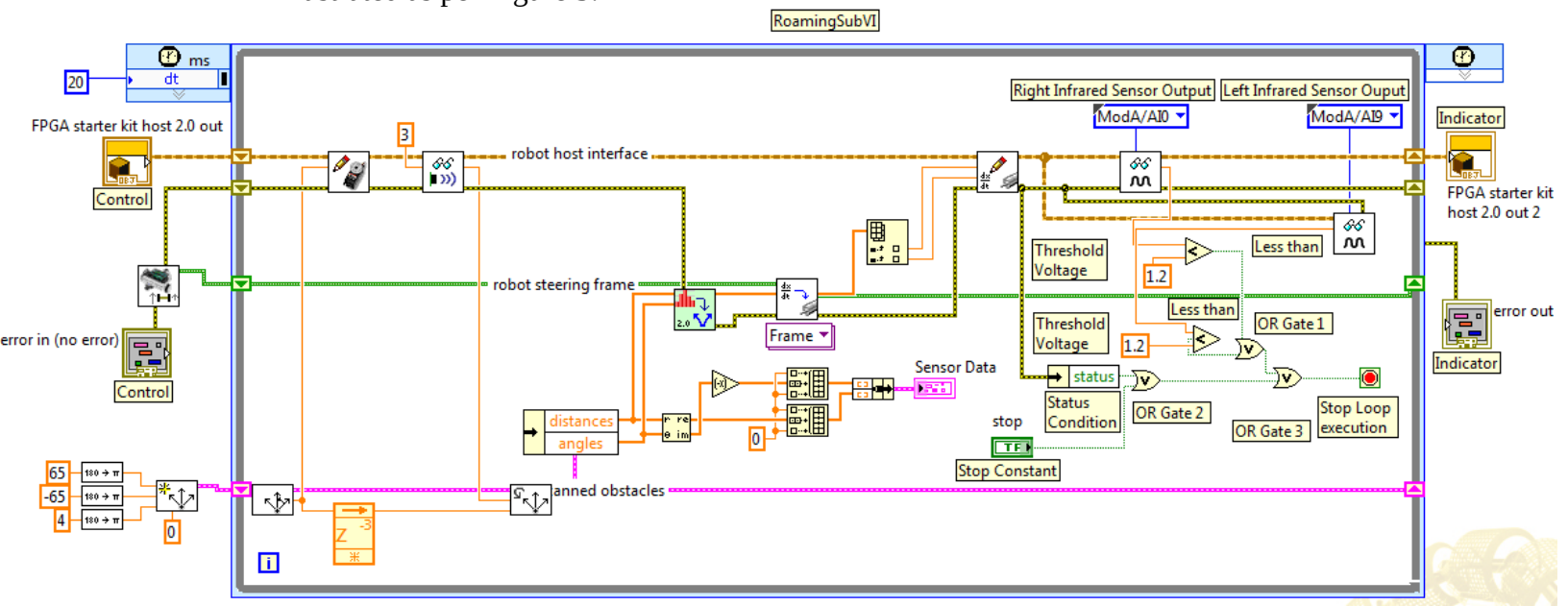


Figure 5: Complete RoamingSubVI program

5. On the front panel of the RoamingSubVI VI the top two blocks i.e. left and right of the FPGA starter kit host 2.0 out were connected to the top left & right connector blocks respectively. Also the bottom error in & error out lines were connected to the bottom left & right connector blocks respectively. The completed front panel VI was as illustrated by Figure 6.

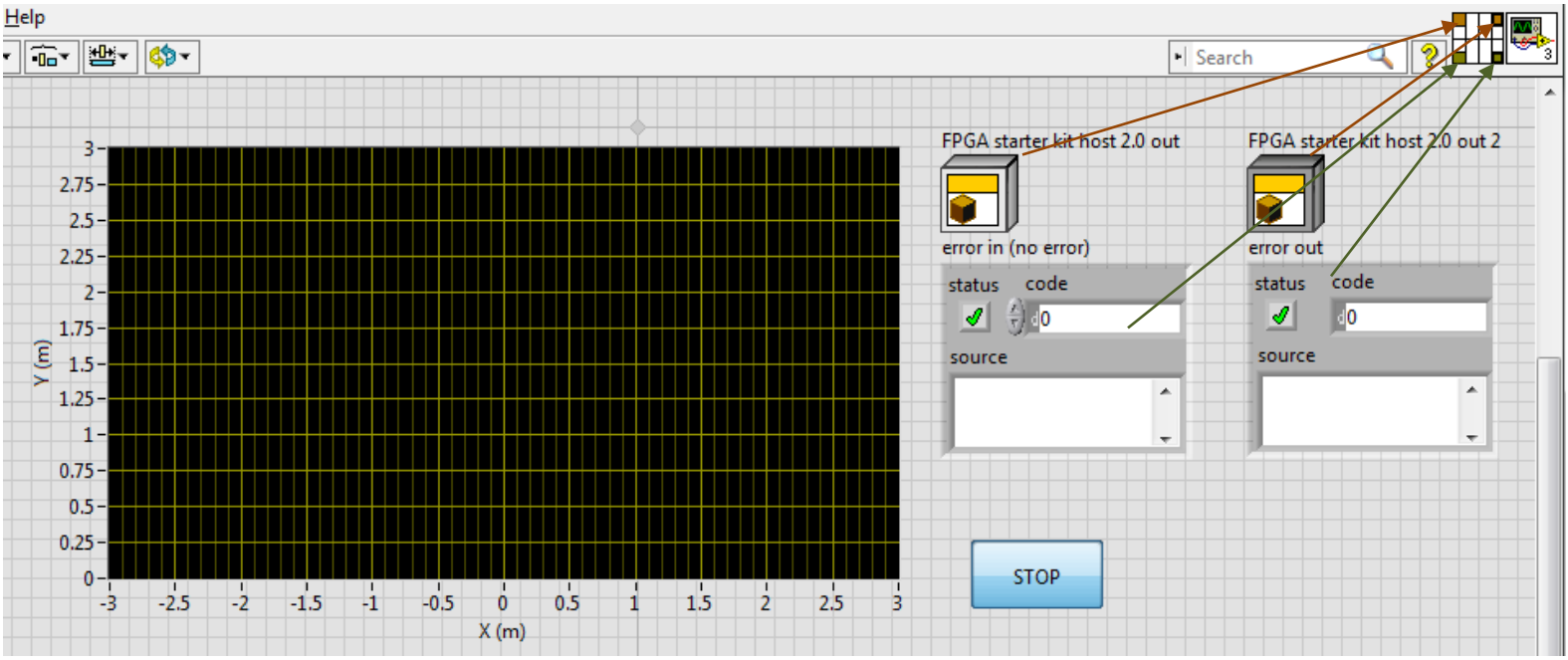


Figure 6: Front Panel of the Completed RoamingSubVI program

6. The completed RoamingSubVI VI was called into the RoamingWithEdgeDetection VI in Frame 1.This was done by selecting the RoamingSubVI VI and adding it to the RoamingWithEdgeDetection VI. The completed main VI was as illustrated by Figure 7.

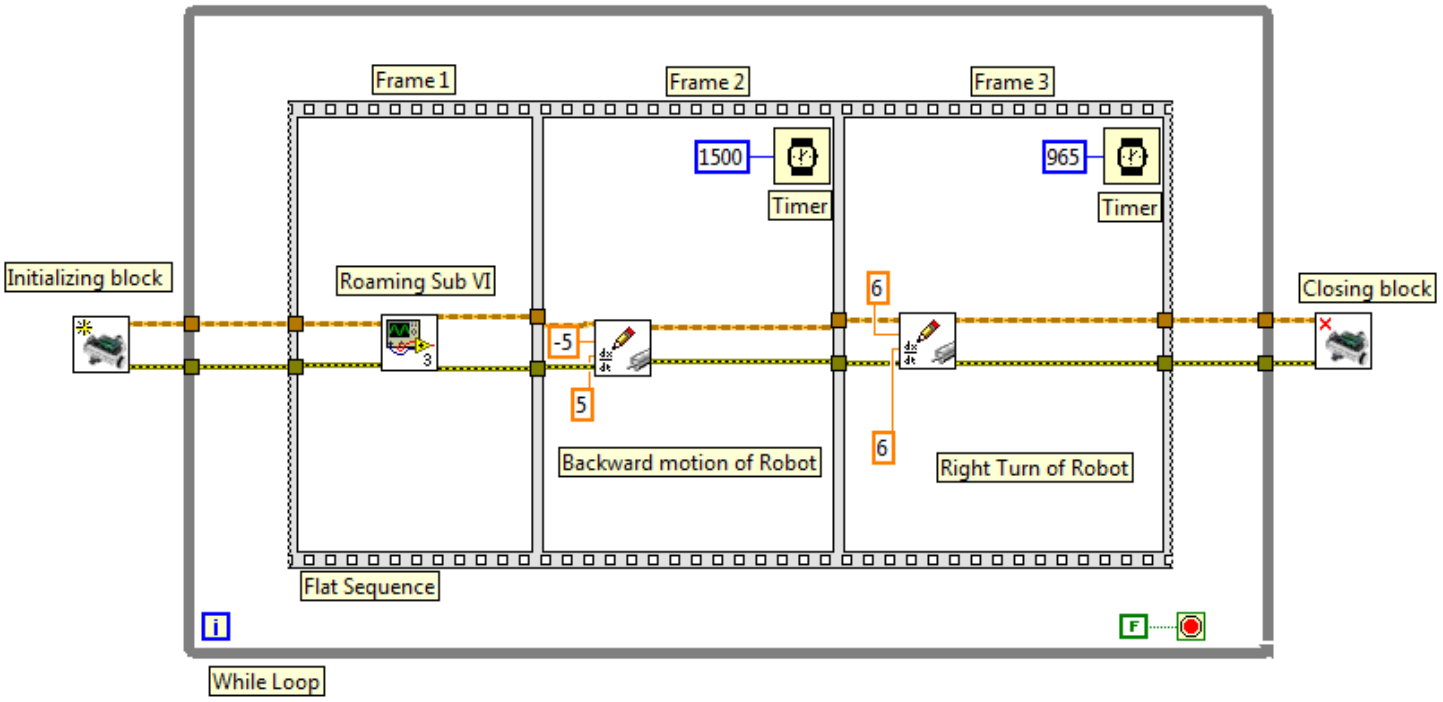


Figure 7: RoamingWithEdgeDetection completed program

Observations while completing/testing the Lab:

It was observed in the experiment that the infrared sensors & the ultrasonic sensor did behave as expected. The robot was able to successfully detect an edge, back up and turn in order to avoid falling over the edge. This behavior was also replicated when a black object (which absorbs infrared radiation and thus mimics large distance) was brought in close vicinity of either of the infrared sensors. Another observation which was made was that the robot was also able to evade walls and obstacles. Both the obstacle evading & edge detection features were found to be functional individually.

The robot continued to move forward until an edge was detected or an obstacle presence was reported by the ping sensor status. Thus the robot was able to evade obstacles as well as detect edges & respond to them concurrently.

Lessons Learned:
In this lab we learned the following concepts & implementations.
a. To interface Sharp IR sensors to the sbRIO.
b. Extreme caution needs to be taken to ensure that the robot does not fall of an edge during testing and demonstration as the algorithmic logic might be perfect but the sensors which govern the working of this algorithm aren't accurate and are subject to ambient reflections.
c. The usage of the Less Than & OR gates.

References:

NI sbRIO 9632 Manual, Source: http://www.ni.com/pdf/manuals/375052c.pdf
Lab 4 Assignment & Tutorial