Wesley Vo
ESE 356 Project 2 Phase 1

This project is incredibly hard to implement, but I believe I understand the idea of how the modules communicate with each other? The problems I have is

1. How do we know where the robot wants to go? (Does it default go in one direction?)
2. When the processing sends a STOP to the server, how does the server know if the robot should resume moving?
    a. It'd have to know the x and y positions of the robot, but how would it know if this information is supposedly stored only in the processing module?
3.

The map of the grids is provided in the attached Excel file. Each row contains the current grid number, its X and Y coordinates, and the grid number to its North, South, East, and West directions. The N/S/E/W grids are marked as -1 if the corresponding direction leads to the boundary of the map or if there is an obstacle.

The idea of the processing module is to have its prc_update() process constantly update the positions of the robots and the obstacles every clock cycle.

1. If at any point during the process the robot's coordinates are within 3 units of the obstacle's coordinates, the status in the main table of the respective robot index will be changed to a 0, the modified column will be set as 1, and the robot will stop moving.
    a. Tx_counter is incremented for every modified column that is set to 1
2. If the robot is close to the boundary of the grid, the status in the main table will be changed to a 3, the modified column will be set as 1, and the robot will stop moving
    a. Tx_counter is incremented for every modified column that is set to 1
3. At the end of prc_update(), the process prc_tx() is notified through the sc_event
    a. prc_tx() scans the update table for MODIFIED and then brings the status that is to the tx_table, and then outputs the data and a flag to the robot module into its prc_rx_p() process

The idea of the robot module is to forward the data sent from the Processing module's tx_table from prc_tx into the main table and then forward the modified data into the Server module's rx_table from prc_rx.

The idea of the server module is to determine if the robot should cross the boundary or resume after an obstacle clearing by checking the coordinates of the current grid and the next grid.

Rough flow of how data is forwarded

1. Processing main_table -> Processing tx_table -> Robot rx_p_table -> Robot main_table
2. Robot main_table -> Robot tx_s_table -> Server rx_table -> Server main_table

Wesley Vo
ESE 356 Project 2 Phase 1

3. Server main_table -> Server tx_table -> Robot rx_s_table -> Robot main_table
4. Robot main_table -> Robot tx_p_table -> Processing rx_table -> Processing main_table