List of requirements:

Functional:

Player 1 (on game console):

* See the movement of the car on the display
* Move with the car to sides using joystick
* At the end of the game, see who won
* Timer controlled tasks DMS or RMS

Player 2 (on PC/Laptop):

* Start/Load the game
* Play by using simple inputs (arrows) instead of the controller
* Be able to see the score, game time, progress of the game
* Be able to see, the actual game on his screen
* End/terminate the game
* At the end of the game, see who won

Non-Functional:

* The system must be implemented with FreeRTOS
* Need to have our protocol with flow control and error detection
* Serial connection between game-console and PC
* Code should be unit tested
* We must use semaphores or mutexes
* At least 3 tasks
* At least 2 hard real-time tasks

Constraints

* We use only given hardware (ATmega 3247, DOT matrix, usb serial communication, joystick, R2R DA)
* PC app in C
* Use oscilloscope for computation time measurements
* Only 2 players

Design

Resources

Matrix

Display

Input queue

Tasks

Controller

Display updater – resources: Matrix

Constrain checker – serves the purpose of updating the Matrix correctly based on the input queue but insures that executed inputs make sense.

resources: Matrix, Input queue -> hard RT

Joystick sampler – resources: Input queue

Communication sender – resources: computer receiver -> hard RT

Communication receiver – resources: Input queue -> hard RT

Computer

Communication sender – resources

Communication receiver – resources

Input sampler

Questions for teachers

Deadlines for tasks.

3.3 ms period for display updater?

How to use RTOS on pc.

How to use oscilloscope and R2R.

Should input be sporadic?

Can we implement existing protocol or should design one?