Mobile vehicle controlable via network(WiFi)

Mateusz Wasielewski

January 2016

Class: Intermediate Project
Instructor: dr Witold Paluszyński
Department: Electronic
University: Wroclaw Univesity of Technology

Assumptions/Goals:

- build mobile vehicle which can be controller via Network
- whole communication between user and vehicle will be acomplished by Network(Wifi)
- vehicle could be controlled by using keyboard
- user will have ability to control a robot over network by using XBOX/PS3 or other console controllers connected to the PC
- camera will be used to streaming the image from vehicle via network

Results:

- mobile vehicle could be controlled via network medium
- whole communication is accomplished by Network(Wifi)
- user is able to control vehicle either by using keyboard or popular XBOX/PS3 controller
- camera wasn't used to streaming the image from vehicle via network

1 Introduction

Main goal of the project is to build mobile vehicle which can be controlled via network. User will have ability to control a robot over network with control pads connected to the PC. Rapsberry Pi, camera and project evaulation board (FRDM-MKL25Z) will be fixed to mobile vehicle. All communication betweern user(PC) and vehicle(Rapsberry Pi) will be acomplished by Network(Wifi).

1.1 Project hardware(details)

- *Host:PC with Windows OS* - is responsible for connecting to raspberry Pi via SSH (could be used PuTTy program for that)
- *Mobile vehicle* - vehicle with servo and DC motors, mobile vehicle is a start kit for participants of Freescale Cup
- *Project evaulation board(FRDM-MKL25Z)* - FRDM-MKL25Z board is responsible for control the motors, servo and receiving the data from Raspberry-Pi via Uart Interface.
- *Rapsberry Pi* - Rapsberry Pi is resposible for processing the data from user and send it via UART interface to FRDM-Board.
- *Camera* - Camera which could be connected to the Rapsberry Pi, thanks that user will be able to control vehicle according to image from camera

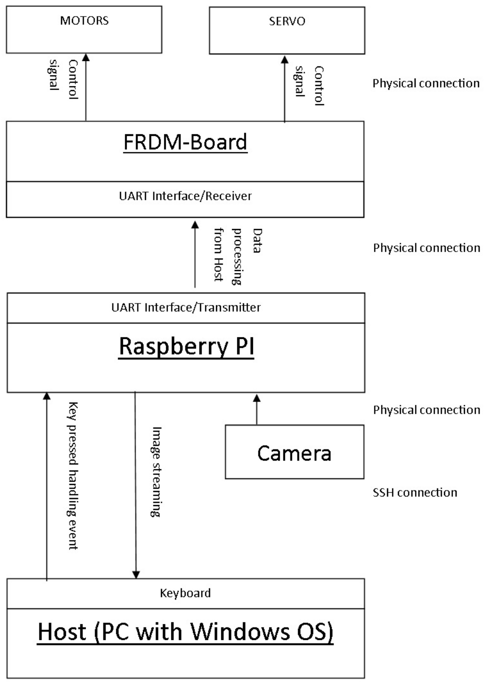

Figure 1: Project hardware relation graph

1.2 Software implementation

Main approach for this project was control vehicle via network. The software implementation was divided on two sections, Raspberry Pi and FRDM-Board. For Raspberry pi was written python script which was running via SSH connection from host side (Windows OS). For FRDM-Board the UART configuration and main control function was written.

1.2.1 Raspberry Pi application

Application on Raspberry Pi i responsible for processing the key pressed event and sending it via UART interface, this application was written in Python 2.7. Script is starting by using SSH connection between PC(Windows OS) and Raspberry Pi(Putty program can be used for that).

1.2.2 FRDM-Board application

Uart configuration function was written in C programming language, also main control function was written. Main control function is responsible for handling/receiving the data from UART interface, processing it and based on accomplished information, determining the speed of motors and servo position.

1.3 Open source application used

To accomplished the control vehicle by using popular XBOX/PS3 controller the open source application was used. This application is mapping the buttons, analog sticks from controller to the specified keyboard keys.This application is called Xpadder (see Reference section)

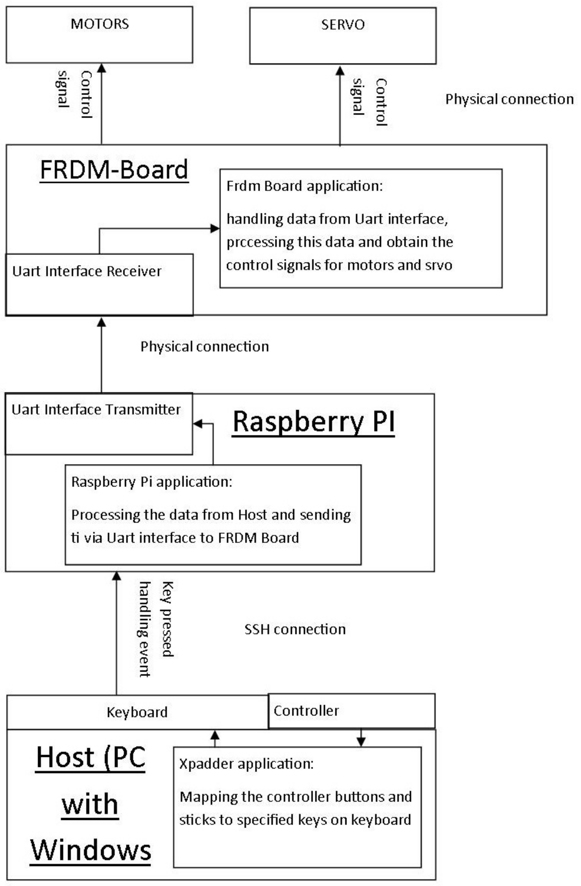
MOTORS
SERVO
Control signal
Control signal
Physical connection
FRDM-Board
Frdm Board application:
handling data from Uart interface, prccessing this data and obtain the control signals for motors and srvo
Uart Interface Receiver
Physical connection
Uart Interface Transmitter
Raspberry PI
Raspberry Pi application:
Processing the data from Host and sending ti via Uart interface to FRDM Board
Key pressed handling event
SSH connection
Keyboard
Controller
Host (PC with Windows
Xpadder application:
Mapping the controller buttons and sticks to specified keys on keyboard

Figure 2: Project software layer relation graph

2 Summary

Main goal of the project was accomplished. User is able to control mobile vehicle via network by using keyboard or popular XBOX/PS3 controller. User is able to increasing/decreasing speed, controlling the servo position and st oped the vehicle by pressing controller sticks or keyboard.
The image isn't streaming via network, camera dedicated to Raspberry Pi crushed during preparing the project. Because of that the image streaming goal wasn't accomplished.
Main problem which occurs is that there exists a delay between sending the first character from keyboard/controller from Host side to raspberry Pi. This problem occurs because the operation system on host side is Windows and this delay occurs because of the windows implemented features. Probably if the operating system for Host side will be changed from Windows to Linux such problems won't occur.

3 Reference

- Freescal Cup resources:
  https://community.freescale.com/docs/DOC-1284?tid=vanFREESCALECUP
- Rapsberry Pi documentation:
  https://www.raspberrypi.org/documentation/
- Linux w systemach embedded, Marcin Bis
- Xpadder application reference: http://sourceforge.net/projects/xpadder/