Internship Performance Review

Mayand Dangi
Embedded Intern

Contribution in Galaxeye Space

MBv2
Multithreading for sensor data read, process and write efficiently

Software requirements document

Custom linux image using petalinux tool

DAQ
RedPitaya's PS code optimization

BRAM data streaming

RXS1500 Testing
Integration of control box, and development of interrupt based switching code for ESP32

Involved in conduction system test

# Motherboard V2: Multithreading

- Objectives: To implement the sensor's data read, process and write operation using multithreading approach

- Three threads created:
    - T_1: Read
    - T_2: Process
    - T_3: Write

- Process will be pseudo parallel

Motherboard V2: Software Requirements Doc

- Software requirements for the new design of motherboard

Figure 1: Software requirements
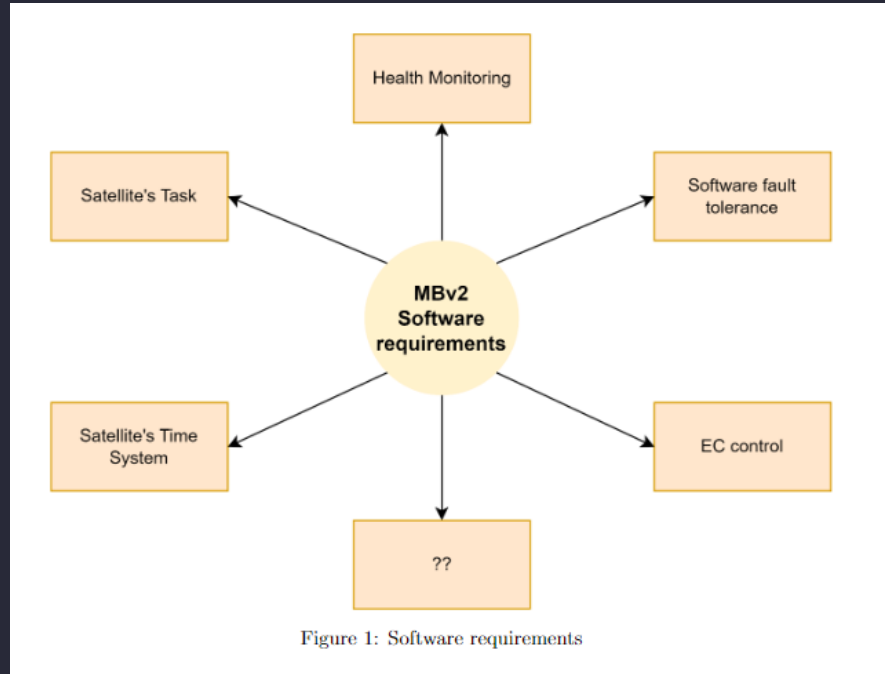
Motherboard V2: Software Requirements Doc
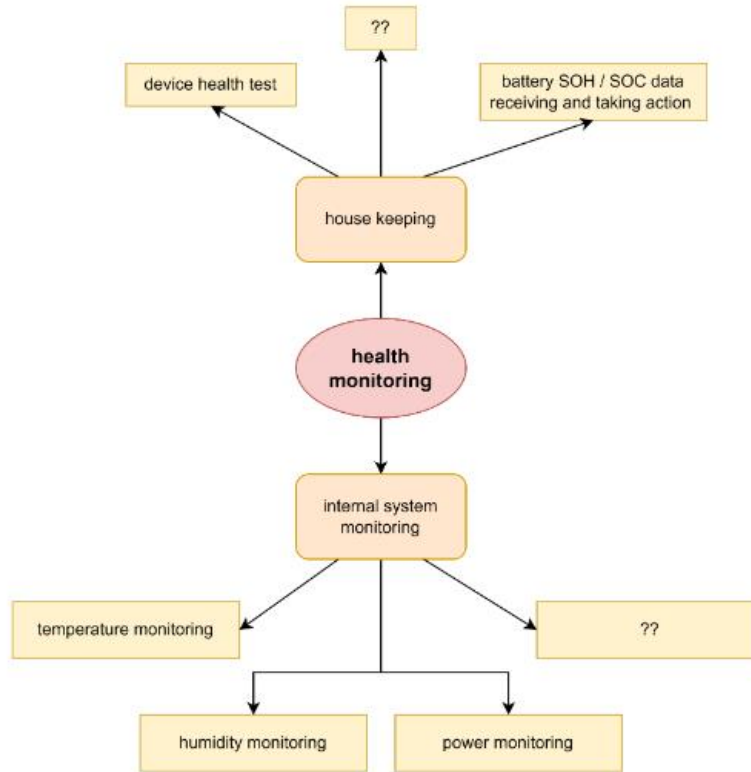
Motherboard V2: Software Requirements Doc

Motherboard V2: Custom OS - Petalinux

Why we need petalinux over ubuntu?

- **Purpose:** ubuntu is general purpose OS, petalinux specifically tailored for embedded systems

- **Integration with FPGA:** petalinux seamlessly integrates with FPGA dev tools

- Performance optimization and Kernel customization

Motherboard V2: Custom OS - Petalinux

- Did literature survey over petalinux, read and understand the Xilinx documentation

- Installation and setup of petalinux environment in ubuntu

    - Note: be careful while choosing version of Ubuntu, Vivado, Vitis and Petalinux all should be compatible with each other

- Explored the petalinux environment

Motherboard V2: Custom OS - Petalinux

Attempt 1

- Tried to built first image for Kria KV260 using pre available .bsp file.
    - Note: petalinux project can be start using .bsp file for .xsa file generated using vivado
- Project didn't get built up because of error in device tree of .bsp file
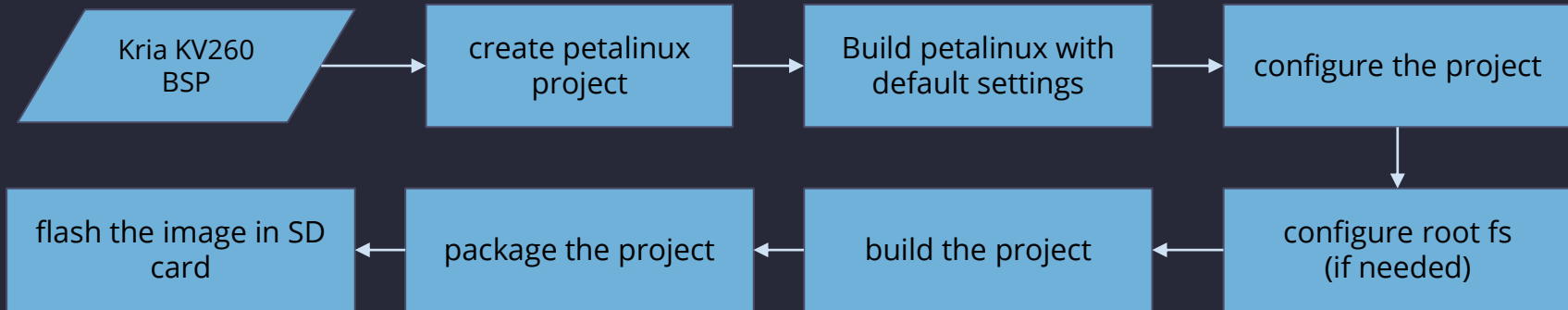
Motherboard V2: Custom OS - Petalinux

Attempt 1 and 2

- Tried to built first image for Kria KV260 using pre available .bsp file.
    - Note: petalinux project can be start using .bsp file for .xsa file generated using vivado
- Project didn't get built up because of error in device tree of .bsp file
- Tried second attempt using .xsa file but still there were some compilation error because .xsa file wasn't generated properly

Motherboard V2: Custom OS - Petalinux

Attempt 3

- Downloaded latest.bsp file of Kria KV260 AI starter kit.
  - Note: not use the downloads link given in documentation those direct to old files
- Use following flow to create the linux image

```
Kria KV260 BSP → create petalinux project → Build petalinux with default settings → configure the project
                                                                                              ↓
flash the image in SD card ← package the project ← build the project ← configure root fs (if needed)
```

Motherboard V2: Custom OS - Petalinux

Future works

- Increase the boot time

- Configure ethernet option properly

- Initialize the each peripherals and corresponding device drivers and perform test

- Use of .xsa file to build linux image

DAQ : Code optimization

Code Optimization

- Removed multiple print statements (reduces number of system calls -> reduce call of OS)
- Stopped data transmission during acquisition process
- Used optimization flag for compilation

Result:

Data Loss drops out significantly to <1% for 600 PRF and <15% for 700 PRF.

DAQ : Streaming

- Requirements for data streaming appeared because data transfer between RP and EC is significantly slow taking hours to transfer the data
- In streaming, RP acts as server and PC acts as client
- Reported data loss is 5-6 % for 600PRF.

Analog signal

data streaming

Server

Client
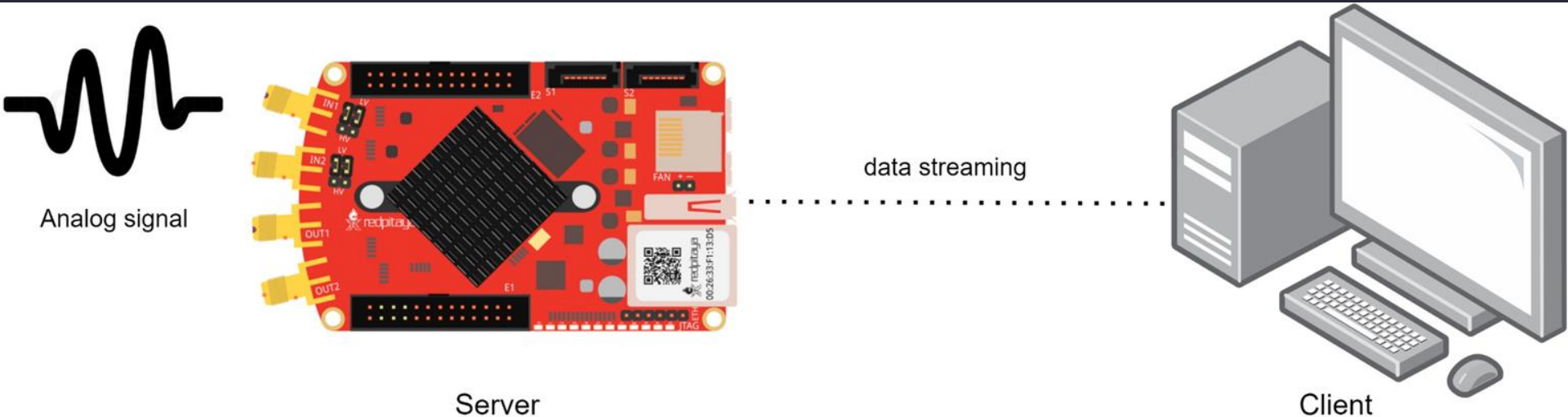
Skills gained

- Embedded C / C++

- Petalinux

- System Documentation

- Understanding ultrascale+ MPSOC

- System testing procedure