NIKHIL DIVEKAR

+1 (720) 688-6301 nikhil.divekar@colorado.edu linkedin.com/in/nikhildivekar https://github.com/NikhilDivekar7

Education:

University of Colorado, Boulder, M.S., Electrical Engineering (Embedded Systems): GPA: 3.7/4 Aug 2017 - May 2019

- Courses: Principles of Embedded Software, Advanced Practical Embedded Software Design, IoT Embedded Firmware, Machine Vision, Sensors and Actuators, Embedded Interface Design, Network Systems, Real-time Embedded Systems.

Vivekanand Education Society's Institute of Technology, B.Eng., Electronics Engineering June 2012 – Aug 2016

Work Experience:

Firmware Engineer at Samsung Semiconductors: July 2019 – Present

- Currently working as the part of NVMe Firmware Development team at Samsung to write the firmware for the SSD drives as per the configurations provide by different customers.
- Also, helping with finding root-cause and thereby solving the defects found in test phase or customer qualification phase.

Graduate Teaching Assistant at CU Boulder for 2 major courses: Aug 2018 – May 2019

- Helped Prof. Kevin Gross with redesigning of the Principles of Embedded Software course materials and projects.
- Worked with Prof. Richard Heidebrecht on a very important course about Embedded Linux and Real-time systems.
- Helped students with coding and debugging issues, conducted the demos for key concepts like Makefile, FreeRTOS, Linux.

Associate Software Engineer at Accenture: Nov 2016 - June 2017

- Performed web development for ATnT's website using C# on Visual Studio and MySQL query into database.
- Developed and deployed software in agile model with impetus to entire software lifecycle from analysis to product maintenance.

Project Lead at CU-Electric Self-Driving Car: July 2018 – May 2019

- Responsible for efficient video processing with OpenCV and TensorFlow developed in Python and giving feedback to system.
- Serving the leadership role of delegating work among members, conducting meetings, creating and supervising timeline.

Technical Skills:

- **Programming Languages**: C, C++, Python, VHDL, C#, Embedded C, Verilog, Android, Javascript.
- **Operating System and concepts:** System calls, Kernel module, Multi-threading, Real time operating systems (FreeRTOS), IPC mechanisms: sockets, pipes, message queue, shared memory, synchronization mechanisms: mutex, semaphores.
- **Protocols:** UART, I2C, SPI, TCP/IP, Bluetooth Low Energy, Stop-and-wait protocol, HTTP.
- **Software:** Git, GCC Toolchain, AWS Framework, Kinetis Design Studio, Simplicity Studio, OpenCV, Xilinx, Code Composer Studio, MySQL, Tortoise SVN, PSoC Creator, Visual Studio, MIT app inventor, MATLAB.

Projects:

AWS based Smart Warehouse using Image processing: [LINK] Oct 2018 – Dec 2018

- Developed an automated warehouse system using Image processing (AWS Rekognition) and robotic arm for precise pick-up and drop-off. The system can fetch objects from any position and discard the invalid requests along with confirmation feedback.
- Implemented the usage of AWS IoT frameworks along with MQTT, WebSocket, SQS and RabbitMQ queueing service and designed a user interface using PyQt and HTML (using node.js) which will help user get what he wants with just few clicks.

Self-Driving Car Prototype: [LINK] June 2018 - Aug 2018

- Designed a prototype for self-driving car image processing unit using NVIDIA Jetson TK1 and C++ OpenCV libraries which included lane, pedestrian, vehicle and road sign detection along with steering and speed control according to the feedback.
- Also ran the ROC analysis for determining the efficiency of various modules and result was 85% overall efficiency.

Beaglebone Multithreading and FreeRTOS Application: [LINK] Mar 2018 - Apr 2018

- Designed a fire and smoke detection system using multithreaded programming in Kernel and FreeRTOS.
- Interfaced gas, flame and temperature sensors to TIVA board for continuous monitoring and logged the data on beaglebone.
- Established a communication interface between TM4C1294XL board(Client) and Beaglebone Green(Server) via UART with the feature of graceful exit of all the participating threads in case of any failure.

GCC Build Systems and Firmware bare metal coding: [LINK] Oct 2017 - Dec 2017

- Designed an architecture independent and portable build system using Makefile along with the compile-time switches.
- Optimized memory usage by designing a circular buffer for UART data transmission, developed a binary logger and created CMOCKA unit tests to validate the incorporated functions. These functions were implemented for FRDM KL25Z and beaglebone.
- Developed the SPI driver to interface Freedom KL25Z board with Nordic transceiver and perform data transmission.
- Implemented DMA transfer to offload some data from CPU and profiled the execution time for various library and user-created functions. All these functions were implemented on Linux platform.

Networking projects: [LINK] Aug 2018 - Oct 2018

- Created a file transfer system using UDP Server-Client Model along with reliability check which handles the packet loss.
- Developed a web server using HTTP protocols and distributed systems server-client model.
- Implemented a proxy server capable of relaying HTTP requests to HTTP server and generating page cache for faster access later.

Bluetooth based Smart Street lighting: [LINK] Mar 2018 - Apr 2018

- Designed a prototype for Bluetooth based automated street lighting according to the traffic density and time of the day.
- Achieved a 45% improvement on energy consumption by implementing event-driven firmware with load power management and bare-metal coding, while performing innovation and automation.