# Monish Nene

*Searching for job as Firmware or Software Engineer*

☎ +1 (720) 688-6309 ✉monish.nene@colorado.edu
https://www.linkedin.com/in/monishnene888a2412a/
Git Profile: https://github.com/monishnene

## Education:

| | | |
|---|---|---|
| Masters in Embedded Systems Engineering University of Colorado, Boulder. | GPA: 3.4/4.0 | May 2019 |
| Bachelor of Engineering in Electronics VESIT, Mumbai University, Mumbai, India. | GPA: 8.00/10.00 | May 2017 |

## Skills:

- **Scripting and Software Development Languages:** C, C++, Python, Shell, GNU Make, ARM and 8051 Assembly.
- **Software Platforms**: Linux, Code blocks, Jupyter, Simplicity Studio, Keil, Kinetis Design Studio, PSoC Creator, Matlab, Altium.
- **Debugging and Evaluation:** Git, Dhrystone Benchmarking, Logic Port Analyzer, Algorithm time Analysis, Digital Oscilloscope.
- **Coding Skills:** Sorting, String Manipulation, Tree, Linked Lists, Object Oriented Programming, pthreads, Regex, data logging.
- **Courses:** Principles of Embedded Software, IoT Embedded Firmware, Real Time Embedded Systems, Network Systems, Embedded System Design, Embedding Sensors and Actuators, Mastering Embedded Systems Architecture, Graduate Algorithms.

## Experience:

- ***Graduate Teaching Assistant for the course Embedding Sensors and Actuators*** *August 2018-Present*

The course gives hands-on experience on converting analog signals from various sensors to computable data and controlling the speed, position and torque of different types of motors. PID control, analog and digital filters, Component selection.

## Projects:

- ***Time-lapse*** *[ Raspberry Pi (Linux) using openCV library and webcam]{C++}* *August 2018*
  - *( video link - https://www.linkedin.com/feed/update/urn:li:activity:6436535986410319872 )*
  - *Frames were captured at 1Hz and saved as in ppm with embedded timestamp and jpeg format. 5 Real-Time threads were used for scheduler, capture image, save as ppm, edit ppm header and compress and save in JPEG format*
- ***Data Transmission via Optical Medium*** *[TI MSP432p401r (Transmitter) and AT89C51(Receiver)] {C}* *April 2018*
  - *( video link - https://www.linkedin.com/feed/update/urn:li:activity:6396109419922411520 )*
  - Audio signal converted to digital signal with ADC and sent over optical medium using Laser and Photodiode. Audio signal at 160kbps sent as data with a custom asynchronous communication with start bit and data byte. Bit-Banging done at the receiver.
- ***Accelerometer Controlled Robot - Bluetooth low energy*** *[Silicon Labs EFR32 Blue Gecko BGM 121] {C}* *April 2018*
  - Complete Bluetooth Protocol written in C. Client developed with software event scheduler to handle 2 Services at a time. Accelerometer was on the handheld client and according to its orientation, the command was sent to the server. The server moved a chassis according to command from client. The second service was a feedback from the robot to the client about the ambient light measured with a I2C interfaced LUX sensor. Encryption Decryption using AES32 and MITM protection done.
- ***System on chip Design for AT89C51*** *{C,8051 Assembly}* *March 2018*
  - Power, Reset, Clock, NVSRAM interfacing, SPLD for memory mapping, RS232 Connection with UART, Bootloader mode switch, Debug Latch, 16 x 4 LCD Interfacing, EEPROM with I2C interface, DAC controlled with SPI, user interface designing.
- ***Health Thermometer - Bluetooth Low Energy*** *[Silicon Labs EFR32 Blue Gecko BGM 121] {C}* *February 2018*
  - Sleep routine to keep the average current below 5uA.Tranmission power adjusted with feedback. Over the air update successful.
- ***Laser Detector and follower*** *[Cypress PSoC 5LP CY8C5868AXI-LP035] {C}* *December 2017*
  - *( video link - https://www.linkedin.com/feed/update/urn:li:activity:6359496404951449600/ )*
  - A LASER was attached to a stepper motor moving +10° to -10°. Accelerometer feedback for initial homing. 2 LDRs attached to a linear slider DC motor. The slider moved according to feedback from the LDRs and followed the LASER using a state machine. Feedback from a Potentiometer on slider used to avoid ramming at the ends. The system worked at max 3Hz frequency
- ***Interfacing SPI, Data Logging, DMA and Profiling*** *[Beagle Bone Black (Linux), FRDM KL25Z] {C}* *November 2017*
  - Task queue and interrupt handlers used for scheduling. DMA used for data memory movements. Binary data logger with RTC time stamp to log encoded data for tasks performed. The logs were later decoded using python and converted to text. Profiling, finding the execution time of functions done for various parameters. Interfaced a Nordic Wireless transceiver with SPI protocol.
- ***DC Motor Torque and Speed Control*** *[Cypress PSoC 5LP CY8C5868AXI-LP035] {C}* *October 2017*
  - Speed control using PWM and Torque Control by controlling armature current. Current was controlled by using Power MOSFET in linear range or by changing frequency of PWM. Feedbacks used were Encoder, Back EMF and Current Ripple separately.
- ***Interfacing UART*** *[ FRDM KL25Z (Arm Cortex M0+ Bare metal OS) ] {C}* *October 2017*
- ***Linux Compilation and Cross-Compilation*** *[Ubuntu(Linux), Beagle Bone Black, FRDM KL25Z ] {C}* *September 2017*
  - Wrote datatype conversion and memory management functions along with error checks. Compile time switch macro used to make the code compatible for all platforms. Created a Makefile to select the compiler according to platform, compile C files, link required object files and build an executable file.
- ***Voice Controlled Robot*** *[Raspberry Pi (Linux) , Arduino]{Python}* *March 2017*

p.s. For detailed information about me and my projects visit my linkedIn profile. Link at the top right of the page.