**Mahendra K. Dalavi**

**Email :** [**mahendradalavi@gmail.com**](mailto:mahendradalavi@gmail.com)

**Mobile : 09972543171/09986651110**

**SUMMARY:**

* Three years of experience in C, Micro controllers 8 bit, 32 bit and embedded application development and test automation.
* Experience in protocols like I2C, SPI, and UART.
* Wireless Communication protocols like IEEE 802.15.4, ZigBee Pro, Application profiles, 6LoWPAN
* Manual testing, Test automation and Python scripting using Python 2.5, Python 2.7 and Eclipse IDE for Python.
* Bachelor of Engineering in Electronics & communication engineering under Vishveshwarayya Technological University.
* Currently working for Mindteck India ltd, Bangalore

**TECHNICAL EXPERIENCE**

**Programming Languages** **:** Embedded C, C, Python scripting.

**Operating System :** Windows XP, Linux.

**Controllers/Processors** **:** ARM7, ARM Cortex M3, PIC16F, PIC18F, Intel 8051

**Interface Protocols :** SPI, UART, I2C

**Wireless Protocols :** IEEE802.15.4, Zigbee, Zigbee IP, SE 1.1, HA 1.x, ZLL 1.0,

6LoWPAN

**Debugging Tools** **and IDE** **:** Kiel µVision 3 IDE, MPLAB 8.30, IAR EWARM 6.3,

Orcad Schematics, Proteus schematic simulation, H-JTAG 1.0, Segger JLink, ST Micro JLink, Kiel µVision 4 IDE, Eclipse IDE (Helios, Kepler)

|  |  |
| --- | --- |
| **Project Name** | **MAC and PHY test scripts development** |
| **Environment** | Embedded c, Python2.7, Eclipse IDE(Helios) |
| **IDEs and Hardware** | Diantree Sensor Network Analyzer, Sniffer, Python win 2.7, Eclipse IDE |
| **Project Description** | This project involved complete MAC, Zigbee stack development and test scripts for MAC features test and Zigbee features test. As well as ZigbeeIP test scripts development. Written scripts for MAC and PHY in Eclipse. Configured Eclipse for Python 2.7. Tested Mac scripts (Automation Testing). Results will be logged in xml, http format. |
| **Responsibilities** | * + Written parser code for interfacing module.   + Automation testing   + Written test scripts for MAC and PHY layers for testing various features of MAC and PHY by referring ZigbeeIP test spec. |

|  |  |
| --- | --- |
| **Project Name** | **MAC 2006 interfacing** |
| **Environment** | Embedded c, Python2.7, Eclipse IDE (Helios) |
| **IDEs and Hardware** | IAR 6.5, Diantree Sensor Network Analyzer, Sniffer, CortexM3, Python win 2.7, Eclipse IDE |
| **Project Description** | This project involved the interfacing client provided mac firmware with MAC test suite. Developed parsing code for interfacing MAC firmware with MAC test suite. Written scripts for MAC in Eclipse. Configured Eclipse for Python 2.7. Tested mac scripts(Automation Testing). Results will be logged in xml, http format. |
| **Responsibilities** | * + Involved in parser development for Interfascing module.   + Automation testing   + Written over 100 test scripts for MAC layer testing and various features of MAC. |

|  |  |
| --- | --- |
| **Project Name** | **ZigBee Development** |
| **Environment** | Embedded c, Python2.5 |
| **IDEs and Hardware** | IAR 6.3, Diantree Sensor Network Analyzer, Sniffer,STM3210E, STM108, Python win 2.5, Python IDLE |
| **Project Description** | This project involved the development of IEEE802.15.4 MAC, ZigBee Pro, various application profiles and Over-The-Air upgrading support. This also involved developing a serial protocol used by a host application processor to interact with the ZigBee Pro stack running on a Network Co-Processor (NCP). Messages are sent between the host and the NCP over either a SPI or a UART interface. On host device an application will be running which controls the NCP device on which ZigBee Pro stack is running. STM32F based board acts as host device and STM32W based board which has cortex M3 ARM processor is used for NCP. |
| **Responsibilities** | * + Involved in driver development for SPI   + Involved in Automation testing   + Involved in Automated testing of OTA(Over The Air)   + Written over 100 test scripts for application profiles of Zigbee Pro such as MAC, ZigBee Pro, HA1.1, HA1.2, SE1.1 and ZLL (ZigBee Light Link). |

|  |  |
| --- | --- |
| **Project Name** | **6LoWPAN Development** |
| **Environment** | Embedded c, Python 2.5 |
| **IDEs and Hardware** | Kiel Uvision 4 IDE, Wireshark Sensor Network Analyzer, IDLE IDE, Segger JLink, IAR 6.5, Python win |
| **Project Description** | This project involved developing an ipv6 stack for a Cortex M3 based controller for 802.15.4 based Wireless Sensor networks.. 6LowPAN is the adaptation layer which provides header compression and fragmentation support to take care of small payload size of 802.15.4 based MACs. This is low power low data rate radio. 802.15.4 MAC has max transmission unit (127) size less as compared to the IPv6 MTU (1284)size. |
| **Responsibilities** | * Involved in Automation of testing * Written around 100 test Scripts for Automation of test cases in Python 2.5, to test various features of 6LoWPAN such as Header Compression(HC), Neighbor Discovery(ND), Routing Protocol for LLNs (RPL) |

|  |  |
| --- | --- |
| **Project Name** | **GPS Tracking system** |
| **Environment** | C , embedded c. |
| **IDEs and Hardware** | Kiel Uvision 3, inbuilt compiler,AT89C51RD2 |
| **Project Description** | The idea behind this project is to monitor and communicate with the vehicles that are belonging to an organization. This is implemented by using GPS (IWAVE). Using GSM (SIM300) module the module itself communicate with the authorized person without human interface. |

|  |  |
| --- | --- |
| **Project Name** | **Remote Monitoring of engines** |
| **Environment** | Embedded c. |
| **IDEs and Hardware** | MPLAB 8.30, Microchip CAN bus analyzer, PIC18F458 |
| **Project Description** | This project targets in the development of a system where we can monitor an automobile engine continuously and transmit the monitored parameters to a remote place through a wireless Xbee transceiver. And the received values are stored in fixed intervals of time. For the future analysis.  At the remote system we will have a Zigbee transceiver which is connected to a microcontroller. The Zigbee transceiver will receive the values transmitted by the transmission block. The values received by the remote block will be stored according to fixed intervals of time. The basic parameters to monitor are Temp, CO (Carbon Monoxide), Vibration in engine block. |
| **Responsibilities** | Developed Code for Application Which Interacts with the Device for Transmitting and receiving the data. |