

Embedded System Programming with 8051	
<p>Total duration - 60-70 hours of lectures and practicals/hands-on. After this a mini-project will be undertaken by the trainee based on the learnings of this course. The trainee would be encouraged to design the project before implementing.</p> <p>This training will prepare you for designing your own projects using 8051.</p>	
<p>Course Instructor - Rupam Das, Freelance Trainer and Consultant. Total professional experience of over 13 years in Embedded Software and related domains.</p> <p>Email: info@rupamdas.com, mailrupam@gmail.com</p> <p>Phone: +91 98402 61709</p>	
<p>My Website - http://www.rupamdas.com</p>	
<p><u>Introduction</u></p> <p>This training program has been designed to provide a thorough understanding of Embedded Systems basics and principles using 8051 based 8-bit microcontrollers. It tackles all the basic components of Embedded Systems in isolation and also creates an understanding on how to integrate these components, to design, program and develop a complete system. The knowledge imparted in this training will serve as a solid foundation for Embedded Systems design and development principles, whether for 8-bit microcontrollers or a 64-bit microprocessor.</p> <p>This training contains coverage of technical knowledge that would provide a definite edge in core sector job interviews. The level of C programming taught in this course is at par with what fresh engineers work on, after joining their first job.</p> <p>All the modules have ample hands-on and lab sessions, with more tutorials provided for further study. Each topic in 8051 programming will be matched with real-time experiments on the hardware kits.</p>	
<p><u>Target audience for the program</u></p> <p>This program is targeted at Engineering Students in their pre-final and final years, but not limited to them. This program can also be attended by interested students of MCA and PG Engineering streams.</p>	
<p><u>Prerequisites/Knowledge required to attend this program</u></p> <ul style="list-style-type: none"> - Exposure to programming, in any language - C programming - basic level - Basics of Digital Electronics would be advisable 	
<p><u>Areas to be covered in this program</u></p> <ul style="list-style-type: none"> - C Foundations - Refresher - Introduction to Embedded Systems - 8051 Architecture and C51 Assembly Language Programming - NXP P89V51RD2 microcontroller and its advanced features w.r.t. normal 8051 microcontroller - Embedded C for 8051 - Design of Embedded Systems 	

Content	
1	C Foundations - Refresher
	Foundations of C programming - Header files, Libraries, Compilers
	Types, declarations and expressions
	Control Flow, Functions and program structure
	Arrays, pointers, structures and unions
	Input and output handling
2	Introduction to Embedded Systems
	What is Embedded System?
	Standard component of Embedded Systems - Processor, Memory, I/O, Peripherals, Software, Algorithms
	Processor types - Microcontrollers, Microprocessors, DSP, FPGA
	Memory types - RAM, SRAM, DRAM, DDRAM, EPROM, EEPROM, OTP
	Peripherals - Parallel and Serial Ports, UART, Timers, Real time clocks, I2C bus, DMA controllers
	Analogue Interfaces - A-D and D-A conversion, CODECs
	Interrupts and Interrupt handling
	Real-Time Operating Systems (RTOS) - Introduction and basics
3	8051 Architecture and C51 Assembly Language Programming
	8051 Architecture - Oscillator, Clock, Internal memory and RAM, SFRs, Internal ROM, I/O pins and ports
	External memory, Counters and Timers, Serial Data I/O, Interrupts
	Assembly language instructions
	Data operations, logical operations, arithmetic operations, Program control
	Configuration and programming of 8051 Ports, Timers and Interrupts
	Interfacing 8051 with LEDs and 7-Segment Display and programming
	Interfacing with LCD and programming
	Serial Data Communication programming
4	Embedded C
	Embedded Software development using C
	Porting 8051 Assembly code to C
	Cross compilation, Downloading, Testing, Debugging
5	Design of Embedded Systems
	Problem definition, requirements and specifications
	Software planning - Hardware design and software design
	Project ideas and discussion