Advanced Embedded System Design

Chapter 0: Course Introduction
Course Information

• Instructor
  – Truong Quang Vinh, Ph.D.
  – Department of Electronics
    http://www.dee.hcmut.edu.vn/vn/bomon/bmdientu
  – Email: tqvinh@hcmut.edu.vn
  – Homepage: http://www4.hcmut.edu.vn/~tqvinh
  – Office: 116B1, IC Design Lab, Monday 9-11am

• Related undergraduate courses:
  – Micro-processor (Vi xử lý)
  – Embedded system design (Thiết kế hệ thống nhúng)
  – Embedded programming (Lập trình nhúng)
Textbooks


Course Description

• This course provides students with advanced knowledge of embedded system design process.

• Students will have ability to
  – **design** hardware part of an embedded system using ARM microcontroller with peripherals including GPIO, ADC, UART, SPI, USB, and Ethernet.
  – **program** software part of an embedded system with and without operating system using C programming language.
  – **develop** an embedded system project using Proteus, IAR, and KeilC development tools.
<table>
<thead>
<tr>
<th>Week</th>
<th>Content</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Chapter 0: Course introduction</strong>&lt;br&gt;0.1. Course information&lt;br&gt;0.2. Syllabus and schedule&lt;br&gt;0.3. Course preparation <strong>Require students to prepare textbooks, tools, and course materials</strong></td>
<td>Students select class project’s topics</td>
</tr>
<tr>
<td>2</td>
<td><strong>Chapter 1: Embedded System Design Process</strong>&lt;br&gt;1.1. Embedded system features and issues&lt;br&gt;1.2. Embedded system design process&lt;br&gt;1.3. Embedded system analysis <strong>Require self-studying for 3 hours</strong></td>
<td>Quiz</td>
</tr>
<tr>
<td>3,4</td>
<td><strong>Chapter 2: Microcontroller Series</strong>&lt;br&gt;2.1. ARM Cortex-M3&lt;br&gt;2.2. ARM Cortex-M4 <strong>Require self-studying for 6 hours</strong></td>
<td>Assignment 1</td>
</tr>
</tbody>
</table>
## Syllabus

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Sections</th>
<th>Self-Studying Time</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5,6</td>
<td>Chapter 3: C Programming for Embedded Systems</td>
<td>3.1. C Program Basics, 3.2. ARM Cortex-M C Compiler, 3.3. ARM software library, 3.4. FreeRTOS</td>
<td>6 hours</td>
<td>Assignment 2</td>
</tr>
<tr>
<td>7</td>
<td>Chapter 4: Development tools</td>
<td>4.1. Advanced simulation with Proteus, 4.2. Programming tools: IAR and Keil</td>
<td>3 hours</td>
<td>Assignment 3</td>
</tr>
<tr>
<td>8,9</td>
<td>Chapter 5: Using Peripherals and Interrupts</td>
<td>5.1. Parallel IO ports, 5.2. Timers, 5.3. Interrupts, 5.4. Analog IO, 5.5 Serial communication</td>
<td>6 hours</td>
<td>Assignment 4</td>
</tr>
</tbody>
</table>
### Syllabus

| 10 | **Chapter 6: Designing an embedded system project**  
|    | 6.1. Project description  
|    | 6.2. Hardware design  
|    | 6.3. Software design  
|    | 6.4. Design simulation  
|    | 6.5. Design verification  
|    | **Require self-studying for 3 hours** |
| 11-15 | **Experiment**  
|    | 1. ARM Cortex M3 with Stellaris LM3S9B96 kit  
|    | 2. ARM Cortex M4 with Stellaris EK-LM4F120XL kit  
|    | **Require self-studying for 10 hours**  
| 16-19 | **Class project**  
|    | Each group of students do class project at lab  
| 20 | **Present class project**  
|    | Each group of students presents and reports the class project  
|    | **Students do class projects**  
|    | **Students report class projects** |
Grading

• Midterm exam: 20%
• Final exam: 50%
• Lab: 10%
• Project: 20%
  – 2-3 students for one group
  – Select project’s topic at week 3
  – Submit project at week 16
Course Preparation

• Textbooks:
  – download 3 required textbooks

• Software tools:
  – IAR
  – KeilC

• Programming knowledge:
  – C/C++ programming
Project’s requirements

• **Report** in MS Word (follow embedded system design process)
• **Simulate** the design
• **Make** prototype by bread board or PCB board.
• **Present** the design in class
Note: Friendly ARM, LM4F120 LaunchPad, BeagleBoard-xM are available at the Lab 116B1
Recommended class project topics

Using **STM32F3-Discovery Kit / STM32F4-Discovery Kit/ LM4F120 LaunchPad**
1. Hand motion detection
2. Remote Control through Ethernet
3. Temperature & humidity measurement
4. Solar control system
5. Motor control system

Using **Friendly ARM kit / Beagle Board / Raspberry Pi**
1. Image capturing system
2. Data acquisition system
3. Object detection & recognition
4. Remote Control through Ethernet
5. MP3 system
<table>
<thead>
<tr>
<th><strong>Simple project’s Topics</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. 20-Chasing LEDs (at least 10 modes)</td>
</tr>
<tr>
<td>2. LED Message Board (8x32)</td>
</tr>
<tr>
<td>3. 3D-LED cube (3x3x3)</td>
</tr>
<tr>
<td>4. LED fan display</td>
</tr>
<tr>
<td>5. Two-LED Dice</td>
</tr>
<tr>
<td>6. Two-digit 7-Segment LED counter up/down</td>
</tr>
<tr>
<td>7. Digital clock with LCD display</td>
</tr>
<tr>
<td>8. Voltmeter with LCD display</td>
</tr>
<tr>
<td>9. Calculator with keypad and LCD</td>
</tr>
<tr>
<td>10. Serial communication-based calculator</td>
</tr>
<tr>
<td>11. Step motor controller</td>
</tr>
<tr>
<td>12. DC motor controller using PWM</td>
</tr>
<tr>
<td>13. I2C data communication</td>
</tr>
<tr>
<td>14. Battery charger (1A)</td>
</tr>
<tr>
<td>15. Temperature controller</td>
</tr>
<tr>
<td>16. Alarm controller using IR LED</td>
</tr>
<tr>
<td>17. Automatic light controller</td>
</tr>
<tr>
<td>18. Simple music keyboard</td>
</tr>
<tr>
<td>19. Digital door lock</td>
</tr>
<tr>
<td>20. SD card project</td>
</tr>
</tbody>
</table>
Course Overview

1. What is an embedded system?
2. What are differences between embedded system and general computer system?
3. What are applications for embedded systems?
4. What is the most important part in an embedded system?
5. Which kind of embedded system development boards have you practiced on?
6. Which kind of micro-processors do you have experience on?