System Programming

09:10-12:00 Friday

TC-119

- Instructor
  - Quincy Wu (吳坤熹), solomon@ipv6.club.tw
- Textbook
  - Leland L. Beck, "System Software: An Introduction to Systems Programming" (3rd Edition), published by Addison Wesley.
- Requirements
  - Homework 20% (do not copy)
  - Quiz 30%
  - Mid-term exam 30%
  - Final exam 20%
- Course Webpage
  - http://solomon.ipv6.club.tw/Course/SP/
- Tas
  - 吳菖育 changyu@ms11.voip.edu.tw
  - 劉嘉翔 cht.liu@gmail.com

Goals of This Class

- Enforce your programming skill
- Get you acquainted with programming tools on Unix
- Make you prepared for graduate school entrance exams

Introduction

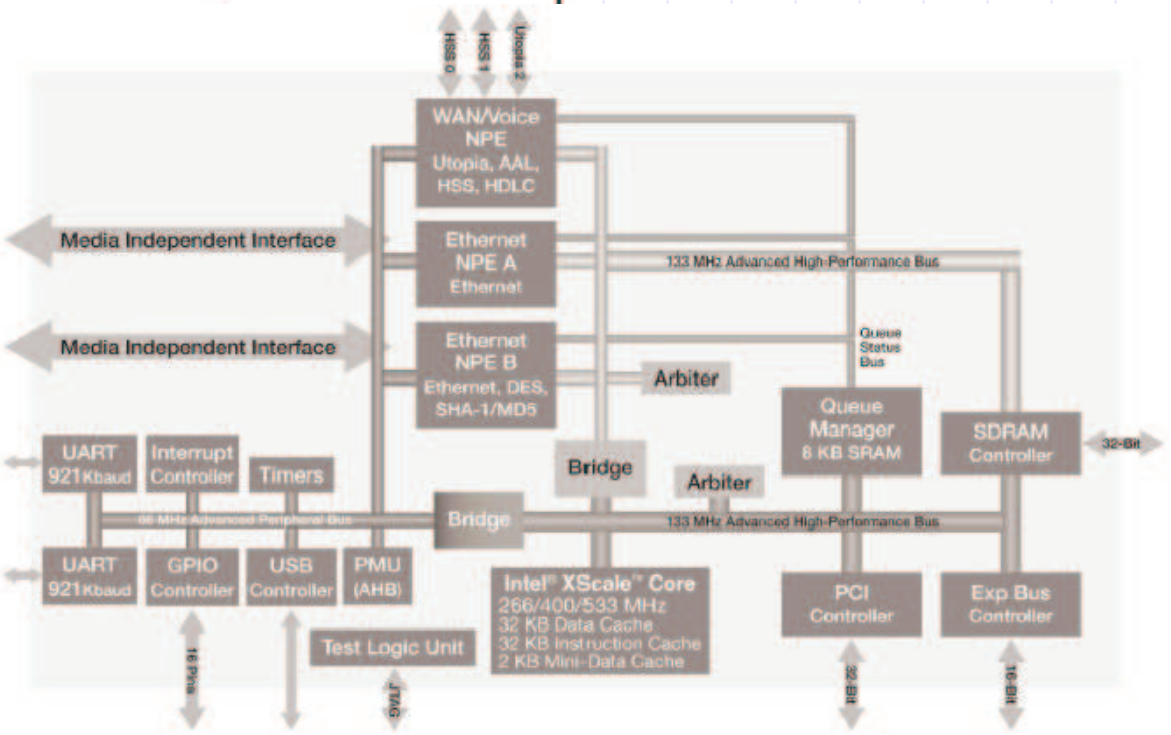
- Software
  - Application software usually used by end-user
    - It is concerned with the solution of some problem, using the computer as a tool, instead of how computers actually work.
  - System software
    - System software consists of a variety of programs that support the operation of a computer (ex: text editor, compiler, debugger)
    - One characteristic in which most system software differ from application software is machine dependency
    - A system software programmer must know the target machine structure

Target Machine Example

Interrupt Status Register Clarification

Section 14.5.1 — that describes the Interrupt Status Register — is missing the entry for Int11. The table is changed as shown below.

Register			INTR_ST (Sheet 1 of 2)
Interrupt Bit	Default Priority	Source	Description
Int0	0	WAN/HSS NPE	Debug/Ex
Int1	1	Ethernet NPE A	Debug/Ex
Int2	2	Ethernet NPE B	Debug/Ex
Int3	3	Queue Manager	Queue[1-
Int4	4	Queue Manager	Queue[33
Int5	5	Timers	General-P
Int6	6	GPIO	GPIO[0]
Int7	7	GPIO	GPIO[1]



Basic Features and Design Options

- Fundamental features
  - Basic functions and characteristics should remain essentially the same, regardless of what machine is being used.
- Major design options
  - There is no single “right” way of doing things; a software designer needs to be aware of the available options in order to make intelligent decisions

System Software and Machine Architecture

- Machine dependent system software
  - System programs are to support the operation and use of the target computer.
  - The difference between different machine
    - Machine code
    - Instruction formats
    - Addressing mode
    - Registers
- Machine independent system software
  - General design and logic is basically the same:
    - Code optimization
    - General design and logic of an assembler

System Software

- The system software includes
  - Assembler
  - Linker
  - Loader
  - Macro processor
  - Text editor
  - Compiler
  - Operating system
  - Debugging system
  - Source Code Control System
  - (optional) Database Management System