Douglas College

Course Information

A: *Division* Instructional
B: *Department* Pure and Applied Science and Technology

Date Aug 4, 1997
New Course
Revision of Course X
Dated March 12, 1992

C: CMPT-150 *Course Number*
D: Intro. to Digital Circuits and Assembly Programming *Descriptive Title*
E: 3 *Credits*

F: *Calendar Description*

This course introduces the theory and practice of digital circuit design, computer architecture, and assembly language programming. Topics include number representation, Boolean algebra, expression minimization using mapping techniques, asynchronous and synchronous circuits, flip-flops, memories, arithmetic logic units, controllers, and interfacing to computers. Designs are implemented using a commercial software product. Assembly language for the 80x86 and/or 68HC11 family of microprocessors is introduced, including register transfer, branching, subroutines, and interfacing.

Summary of Revisions

Sections revised:
C (revision for CMPT-220)
D, G, K, M, N, O, P, Q, R
(please see attachment for summary and justification for C)

G: *Type of Instruction*

Lecture	2+1 hrs/week
Lab.	2 hrs/week
Seminar	
Clinical Experience	
Practicum	
Shop	
Studio	
Student Directed Learning	5 hrs/week (approx.)
Other	
Total	10 hrs/week

H: Course Prerequisites:
CMPT-110

I: *Course Corequisites*
None

J: *Course for which this course is a prerequisite*

K: *Maximum Class Size*
Class 25
Lab. 25

L: *College Credit*
Transfer X

M: *Transfer Credit*
Requested X
Granted

Course Equivalents

U.B.C.	ELEC256
S.F.U.	CMPT150

Course Designer

Vice-President - Instruction

Dean

Registrar

N: *Textbook and Materials to be Purchased by Students*

- Malvino, Brown, Digital Computer Electronics, Macmillan/McGraw-Hill
- Portfolio for logic design assignments
- Two 31/2" high density diskettes

O: *Course Objectives*

The student should be able to:

- demonstrate an understanding of the logic blocks composing a microprocessor
- appreciate, via comparisons, the architecture of microprocessors
- demonstrate, by design and implementation, using a software simulator, gate level logic of microprocessor components such as memory, ALU, and controller
- understand, via use, data representation including numbers of various bases and characters
- design and implement assembly language programs

P: *Course Content*

1 Data representation
- 1.1 Number systems
  - 1.1.1 Decimal, binary, octal, hexadecimal
  - 1.1.2 one and two's complement arithmetic
- 1.2 Character representation
  - 1.2.1 ASCII (others such as EBCDIC and Unicode may be introduced)

2 Gates and combinational circuits
- 2.1 Simplification techniques
  - 2.1.1 Boolean algebra
  - 2.1.2 Karnaugh maps
  - 2.1.3 Brief survey of other (software) techniques
- 2.2 Circuits
  - 2.2.1 Decoders and multiplexers
  - 2.2.2 ALU

3 Memory and sequential circuits
- 3.1 flip-flops
- 3.2 registers
- 3.3 memory
- 3.4 counters and synchronous circuits
- 3.5 sequential machines and controllers

4 Computer architecture
- 4.1 Machine cycles
  - 4.1.1 Fetch-decode-execute-increment PC
  - 4.1.2 instruction cycles and register transfer

5 Comparison of microprocessor families
- 5.1 architecture
- 5.2 instruction sets

6 Assembly language programming
- 6.1 the assembler
- 6.2 data transfer and addressing modes
- 6.3 CPU states, flags, and logical operations
- 6.4 branching and structured programming
- 6.5 subroutines and parameter passing

Q: *Method of Instruction*

There are three components to the course: lectures, labs., and assignments.

The lecture is used to introduce new material; usually via a sequence of theoretical concepts, examples, and practical considerations. The book is to be used as a close adjunct to the lecture notes and examples.

The two hour weekly lab. is used for the teaching and evaluation of circuit designs using the software product LogicWorks 3 and also the evaluation of assembly language programs implemented by the student.

Assignments include data representation, logic designs using LogicWorks 3, and assembly language programming.

R: *Evaluation*

The final grade will be calculated from a particular distribution from the range below. The exact distribution will be given to the student on the first day of classes along with the course outline and necessary policies.

Distribution Range:

class participation[1]	=	0% - 5%
labs. (7 - 14)	=	15% - 25%
2 tests @ 15% - 20% each	=	30% - 40%
1 exam	=	20% - 30%
assignments (4 - 8)	=	20% - 35%

Note #1: participation includes (but is not limited to) short pop quizzes and/or handing-in (part-of) a homework assignment

Example Distribution:

class participation	=	5%
8 labs.	=	20%
test #1	=	15%
test #2	=	15%
assignments	=	20%
exam	=	25%
Total	=	100%

Summary of Revisions

C: There are two major reasons for changing the course number:
1. S.F.U. has changed the number of the corresponding course because the course is now being offered in conjunction with engineering science;
2. CMPT-220 was being offered in conjunction with CIS (as CIS-450); this is no longer the case.

This is not a new course but rather a change in emphasis.

D: Changed to emphasize logic circuits.

G: Changed from 4 to 3 hrs./week with the addition of the lab. for pedagogical reasons.

K: Changed from 36 to 25 due to current software license agreement.

M: Because of the academic structural change at S.F.U. and also the new emphasis on logic circuits for U.B.C.

N, O, P: Because of a change in course emphasis.

Q, R: Because of change in G (type of instruction).