NEW ZEALAND INSTITUTES OF TECHNOLOGY AND POLYTECHNIC QUALIFICATIONS IN INFORMATION & COMMUNICATIONS TECHNOLOGY

PRESCRIPTION: PR530 PROGRAMMING (SYSTEMS)

AIM OF MODULE:	To provide students with a basic introduction to the way in which processors work and can be programmed.
CREDITS:	7
STUDENT LEARNING HOURS:	70
CONTENT REVISED:	1996
PRESCRIPTION EXPIRY DATE:	Nov 2011

Level and Assessment Schedule

TOPICS	Highest Skill Level R	C	A	P	Suggested Assessment Percentage
1. Fetch/Execute Cycle		*			20
2. Program Creation			*		10
3. Instruction Sets		*			15
4. Simple Program Loops			*		15
5. Memory	*				25
6. Understanding Bus Mechanisms	*				15
					100

The student will:

1 FETCH/EXECUTE CYCLE

➢ The student must understand the function of the main components of the CPU.

R 1.1 Define the properties and uses of a register.

C 1.2 Explain the two parts of an instruction:
- the operator (op code)
- the operand (data)

R 1.3 Use a diagram to show the relationship between the following CPU units:
- accumulator
- arithmetic and logic unit
- program counter
- instruction register
- stack pointer
- control and timing

C 1.4 Explain the purpose of each unit in 1.3 above.

R 1.5 Describe a basic Fetch/Execute Cycle.

C 1.6 Demonstrate a fetch/execute cycle with a "real" instruction.

2 PROGRAM CREATION

A 2.1 Define the algorithm to add two bytes from memory and put the result into memory. Draw the logic flow diagram. Code in machine language instructions. Debug.

3 INSTRUCTION SETS

C 3.1 State (and be able to recognise) the following instruction groups:
- arithmetic and logic
- data transfer
- test and branch

R 3.2 Describe the following types of addressing modes:
- direct
- indirect
- immediate
- offset

4 SIMPLE PROGRAM LOOPS

➢ The student is required to write an assembler language program containing loops.

R 4.1 Explain the concept of a loop as being a section of code which is obeyed repetitively.

C 4.2 Show how the number of times a loop is executed is controlled by a parameter, including the way such a parameter is tested on each pass.

C 4.3 Explain how a JUMP or BRANCH instruction is used to position the Program Counter at the beginning of the loop.

A 4.4 Write an algorithm to add five numbers (stored in sequential locations) together and put the sum in the sixth location. Design and write the program.

5 MEMORY

➢ The student should understand the relationship between the CPU and memory.

R 5.1 Explain the concept of memory as a series of uniquely addressable locations.

R 5.2 State the typical way in which the size of memory locations can be described (bytes, words).

R 5.3 Explain the relationship between bytes, words and the ALU.

R 5.4 Explain the operation of a simple memory chip.

R 5.5 Describe the process of address decoding.

R 5.6 Explain the relationship between the number of address lines and the available memory locations that can be addressed).

R 5.7 Describe the possible contents of a memory location (data, instruction, garbage).

R 5.8 Explain the functions of the following:
- chip enable
- read signals
- write signals
- data buses (lines)
- address buses (lines)

6 UNDERSTANDING BUS MECHANISMS

R 6.1 Describe the signals on a
- data bus
- address bus
- control bus

R 6.2 Explain how a bus is used for one piece of information at a time.

R 6.3 Explain how only one sending and one receiving device are active at any one time on a bus.

R 6.4 Describe the tri-state device and how it makes bus systems viable.

R 6.5 Describe how a device can be read from or written to (sequence of operations).