Course Title: Assembly Language and Machine Organization: CSCI 2230
Prerequisite: CSCI 1210.

Course Description: Computer structure and system organization, instruction execution, memory addressing modes, hardware/software interface. Programming in assembly language.

SLOs:

1. Describe the architecture of a microcontroller, the interconnections between the components, and the major units inside the CPU
2. Use signed and unsigned numbers, bitwise operations, branching instructions, and the corresponding flags in the status register
3. Use immediate, direct, indirect addressing modes in assembly instructions
4. Map high-level programming language features to assembly instructions, such as loops, conditionals, procedure calls, value and reference parameter passing, return values, and recursion
5. Interface with input/output devices via instructions, memory addressing, or interrupts
6. Design and implement an assembly language program

4 credits

Comparitors:

NMT:

CSE 221, Computer System Organization, 3 cr, 3 cl hrs Prerequisite: CSE 101, 122 each with a grade of C or higher Usually offered in the Fall semester. The hardware/software interface. Basic organization of hardware and operating systems. Memories, buses, interrupts, input and output, and instruction set architecture. Programming in assembly language.

Learning Outcomes:

1. Understand the implications of bitwise operations, basic operating system concept and the IEEE 754 floating point standard.

2. Understand and write AT&T assembly including arithmetic, logic, conditional and unconditional jumps and calling conventions.

3. Understand basic computer organization and the memory hierarchy.

NMSU:

C S 273. Machine Programming and Organization
4 Credits (3+2P)
Computer structure, instruction execution, addressing techniques; programming in machine and assembly languages. Taught with **C S 464**. May be repeated up to 4 credits.
Prerequisite: At least a C- in **C S 172** or **ENGR 140**.
Learning Outcomes

7. Describe the architecture of a microcontroller, the interconnections between the components, and the basic units inside the CPU
8. Use signed and unsigned numbers, the associated branching instructions, and the corresponding flags in the status register

9. Explain immediate, direct, indirect addressing modes, their opcode and operands, and their utilities
10. Map high-level programming language features to assembly instructions, including loops, conditionals, procedure calls, value and reference parameter passing, return values, and recursion
11. Interface with I/O devices including LED and sensors via digital input and output, and analog-to-digital conversion
12. Program timers/counters and interrupts to control real-time applications
13. Design an assembly program