Introduction

This week's laboratory consists of two parts. In part A, you will use the NUM PLC to practice exchanging information between the PLC and the CNC. In part B, you will practice programming of logic sequences on the Siemens PLC.

Part A (CNC-PLC Communication)

It is possible using the PLC program to read some data from CNC, control the functions of CNC and/or interface with external sensors and actuators. In this part, we will perform two mini-experiments to understand how these things can be done. Imagine that the CNC has been connected to a machine tool and the machine tool has multiple sensors attached to the spindle unit and worktable:

Today, we will perform two mini-experiments to understand how these things can be done.

1) The first experiment is to set up a program to read sensor input and display corresponding error messages. The input ports are connected to the fault detection sensors as follows:

   E1 Lubricant Fault (1)
   E2 Spindle Overheating (2)
   E3 Excessive Tool Wear (3)
   E4 Machining Chatter (4)
   E5 Spindle Overload (5)

   For display, the error code needs to be stored in Byte Variable AN. As AN can only store one error code at a time, the error with larger/largest number, which is more urgent to be addressed, should be written into AN in the presence of multiple errors.

   For example, if spindle overheating is detected, AN=2. Then if machining chatter (a kind of harmful self-excited vibration that many damage both the spindle and workpiece) occurs before the overheating is addressed, AN should be overwritten by 4. If the spindle is overloaded, AN should be 5 no matter what other errors exist. If no error is detected, AN=0.

   After the program is entered, simulate this on NUM machine.

2) The second experiment is to automatically control the cycle start after the part is loaded (E0). Write a simple part program (or use the one you created previously) and use it for this experiment. After checking whether a part is loaded (E0), a correct tool change is performed (E1), and switch E2 is on, begin the cycle by turn on A0. Stop the table if the speed exceeds 100% of the specified max. speed. The current speed is automatically loaded into the byte variable SP. The speed can be read from SP and the maximum speed corresponding to SP = 212. You can increase the speed by turning the speedometer.

Simulate the programs on the NUM PLC and get the inspection of the TA.

Note: the CNC is not really connected to a machine tool, so you still need to use the buttons and LEDs on the front panel to mimic the inputs (E) and the outputs (A, AN).

Note: The following operator is helpful for this lab assignment:

SEL: Binary Selection.

Example in IL:

```
LD  TRUE
SEL 3,4    (* IN0 = 3, IN1 =4 *)
ST  Var1   (* result is 4 *)
LD  FALSE
SEL 3,4
ST  Var1   (* result is 3 *)
```

LT: byte comparison, less than

```
LD    SP
LT    BYTE#212
```

Requirements
Simulate the programs on the NUM PLC and get the inspection of the TA. To save the simulation results, take pictures of the CNC front panel in representative cases. The report must include instruction lists, ladder diagrams and simulation results of both experiments.

Part B (Siemens PLC)

The second laboratory experiment involves programming of logic sequences on Siemens PLC. Siemens PLC's provide three different options for programming. STL is called S5 (see the handout). You can use either ladder logic or S5 language for programming and readily convert one to the other.

1. First, follow the example in the tutorial section until you become familiarized with operation of the Siemens PLC. You can work as a group.

2. Next, generate a ladder logic diagram of the conveyor example problem covered in the class, and then convert it into S5. Refer to the hand for the choice and use of timers.

The report must include printouts of the ladder diagram and S5 program.