Sensing and Interpreting hardware I/O lines from the CNC

This document is a commentary with examples on the automatic collection of machine status data by actually connecting wires to available CNC signals in order to glean the information required. Refer to SuiteFactory Technote entitled **"Collecting Data on the Factory Floor with a DNC System"** which is an overview of several methods of factory floor machine status data collection.

The referenced document lists the following as good reasons for collecting data this way:

- Fully automatic requiring no operator training or intervention.
- Can supply almost everything about the machine's status and outputs the data in real time as events occur. Does not have to be queried.
- Does not require computers, terminals or barcode readers at or near each machine.

The referenced document lists the following as being the difficulties for collecting data this way:

- Requires I/O signals being available – these signals can be "mined" by an engineer using the machine builders wiring diagrams and/or finding them empirically at the machine or specifying the requirements on a new machine or having them added by the machine tool builder. In some cases it might be necessary to add sensors to get some piece of information where a built-in signal cannot be found. In either case there is the expense of finding the signals, adding sensors or paying the machine tool builder to add them.
- Requires interface hardware to sense these signals and transform them it a format that the data collection software can use.
- Requires interpretation of these signals. For example, on one CNC being 'In Cycle' turns an I/O line to 5 volts. On another CNC the signal might go to 24 volts or 0 volts. On one CNC the mode might be determined by a single signal for each mode. On another CNC the mode might be encoded into 3 lines.

A Real Example

Customer required the following CNC status information to be automatically collected from the machine: CNC ON/OFF, Mode (Auto, Single Step, MDI or Manual), Cycle Start, Cycle Stop (is the stop a normal program end or was it aborted?), Hold by pushbutton, Hold by programmed hold, M00 or M01, in edit, Feedrate override not at 100%, spindle speed override not at 100% and Emergency Stop.

This information is used to automatically count the number of parts make and also report on machine usage/efficiency. Requires the operator to login on a computer and enter what job he is about to run and what aspect of the job is happening; that is: Setting up, Running, Waiting for something or breaking down.

Some Specifics

A. To accomplish this on an older GE1050 CNC it required 12 signals soldered, spliced and/or attached to terminal strips.

- 1 to sense a -12volt control on signal
- 1 to sense the cycle start +5v to ground lamp driver signal
- 1 to sense the end of program stop +24 volts going to ground pulse
- 3 to sense and encode the 3 binary coded decimal signals for the M-codes

- 1 to sense the +12 volt to ground Feed Hold signal
- 1 to sense an emergency stop switch contact -12 to ground
- 3 to sense and encode the position of the mode switch +12 volts
- 1 to sense whether edit mode was entered +12 volts to ground

B. To accomplish this on an older E2000 CNC it required only 8 signals soldered, spliced and attached to terminal strips.

- 1 to sense a +5 volt control on signal
- 1 to sense the +5 volt cycle on lamp signal
- 1 to sense a part count +5 volt pulse indicating end of program
- 1 to sense a +5 volt feed hold signal
- 1 to sense a +5 volt Auto mode signal
- 1 to sense an emergency stop ground signal
- 1 to sense and edit mode ground signal
- 1 to sense a +5v Single mode signal

C. Brand new CNC's are usually built with printed circuit board front panels and CRT screens to display status. It is often not possible to "mine" signals from them because the information we need is buried in a printed circuit board or is simple "soft" , i.e., just displayed on a CRT, LCD or plasma screen. So in order to get the signal needed you need cooperation of the machine tool builder or distributor. To accomplish this on a brand new Haas CNC we specified our requirements to Haas and they supplied an output relay board with a terminal strip with the proper signals. Since Haas's relay board had only 8 outputs available and we required more, some had to be encoded by them to be decoded by us. It required 10 signals, 8 from the relay board, 1 from a power supply and 1 from a switch contact

- 1 to sense the +12 volt CNC on signal from a power supply
- 1 to sense the +12 volt cycle on signal
- 2 to determine mode- Auto, Single or Manual
- 2 to determine Hold, M00/01 or M02/M30
- 1 to sense feedrate override not at 100%
- 1 to sense spindle speed override not at 100%
- 1 to sense edit mode
- 1 to sense emergency stop from a switch contact

In addition you have to use some manual data input method to get data as to what program is running and who is running it for what job as well as information such as Machine Setup, Breakdown, Idle, Waiting for Material, Under Maintenance etc. This can easily be done at the machine tool controller by using SuiteDNC's Remote Request feature.

Hardware interface

In order not to interfere with the CNC's operation when attaching wires to it, optically coupled isolators must be used. These isolators must be modular so as to be able to choose different ones to match the various voltages and polarities involved. The data lines that I have been describing have been digital signals, that is, either on or off. It is also possible that analog signals might need to be sensed and measured.

Software interface

In order to interpret what the various signals mean, in light of the fact that each CNC on a shop floor could present a different set of signals for the same information requirements, a means must be provided for programming this in the field by technicians installing the system. By this means, the data presented to the DNC software (SuiteDNC) is "normalized". There does not have to be different version of the DNC software depending on what brand machines are on the factory floor. The DNC software merely time stamps and saves the data in a database in a format compatible for display and report creation.

Available off-the-shelf hardware/software

There are many ways to meet the above requirements including Programmable Logic Controllers (PLC's), I/O module racks, custom built hardware, ...

Our solution today is to use OPTO22's line of SNAP products. You can mix reasonably priced off the shelf I/O racks, modules (both digital and analog for a wide variety of voltages) and smart Ethernet rack controller and field programmable controllers using both flow chart and script techniques. One Ethernet connected programmable controller can be used to input and interpret the data from many I/O racks on the Ethernet LAN. This can be configured all or in part to be wireless by using off-the-shelf Ethernet to Wireless bridges.

In addition to OPTO22's flow chart/script programming software, HMI display software is available enabling you to create a graphical display of the machine's current status.