

LISTEN.
THINK.
SOLVE.®

Advanced PLC Topics for Micro controllers

Rick Sykora – Product Manager

Leo Osma – Application Engineer

Mark Tabler – Application Engineer

Agenda

1. Connected Components Update

2. How to quickly develop machine

3. Hand-on lab

4. Wrap-up

Demo and Lab

- What will you know after completing the lab
 - Be able to quickly configure a drive for speed control
 - Be able to quickly configure the PLC
 - What is the best way to control the drive using the PLC
 - How you can handle interruption of machine operation

Innovation Micro800 & Connected Components Workbench

Just enough control

Part of Connected Components Bundle

Unified Software

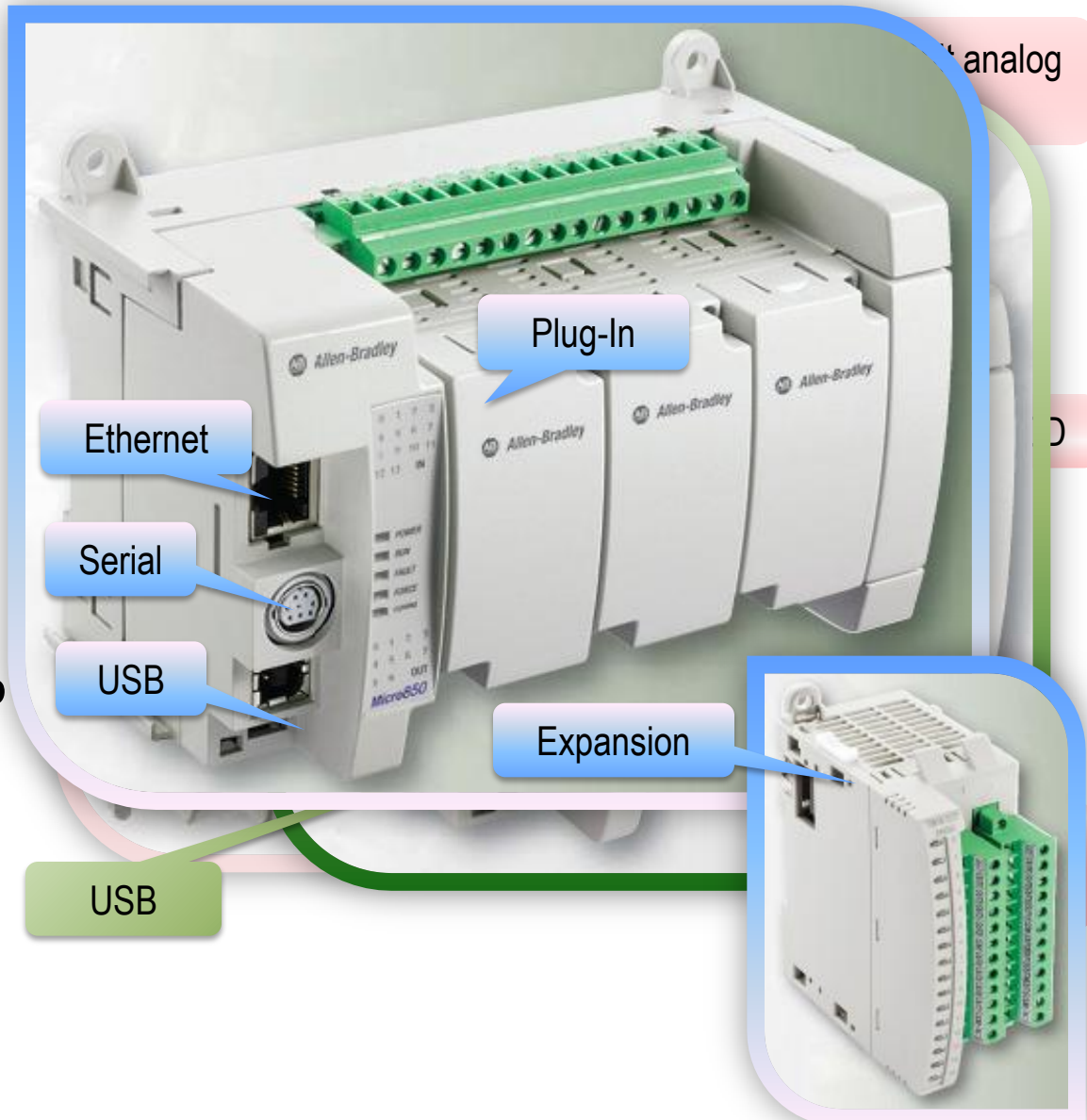
Convenience and connectivity

Easy to Install and Maintain



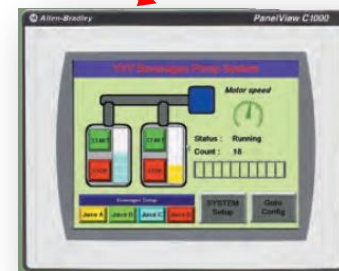
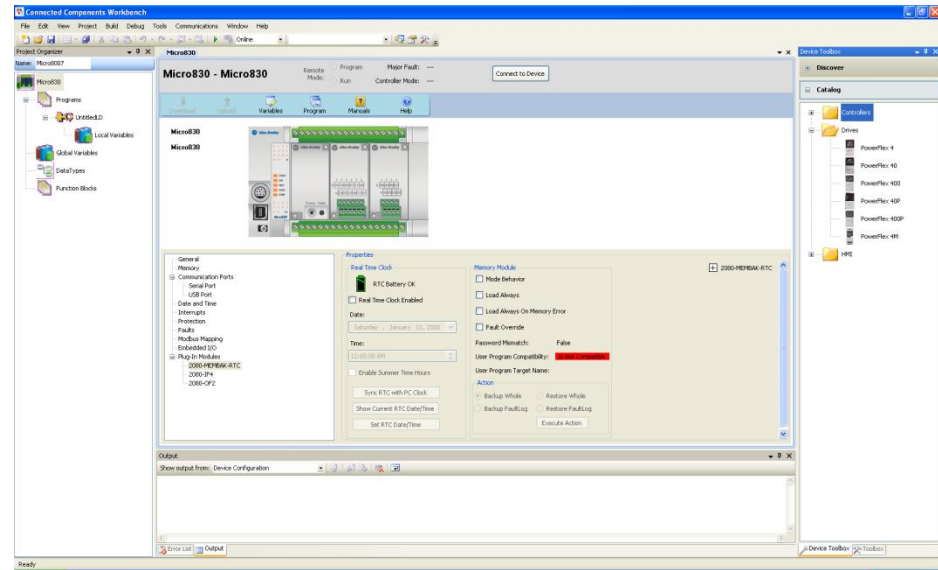
Innovation Micro800 PLC Family

- **Micro810**: Price of a smart relay with the functionality of a Micro PLC
- **Micro830**: Economical PLC with Plug-In customization and PLOpen motion
- **Micro850**: Adds EtherNet/IP and more I/O expansion to Micro830 24 And 48 point PLCs (4Q2012)



Connected Components Workbench Software

- **Easy to Acquire/Install**
 - Free Internet download
- **Easy to Configure**
 - Single software for component class products
 - Graphical Device Configuration
- **Easy to Program**
 - Extensive use of Microsoft and IEC-61131 standards
 - Symbolic Programming
 - Rockwell Automation and user-defined function blocks
- **Easy to Update**
 - Software updates available via Internet



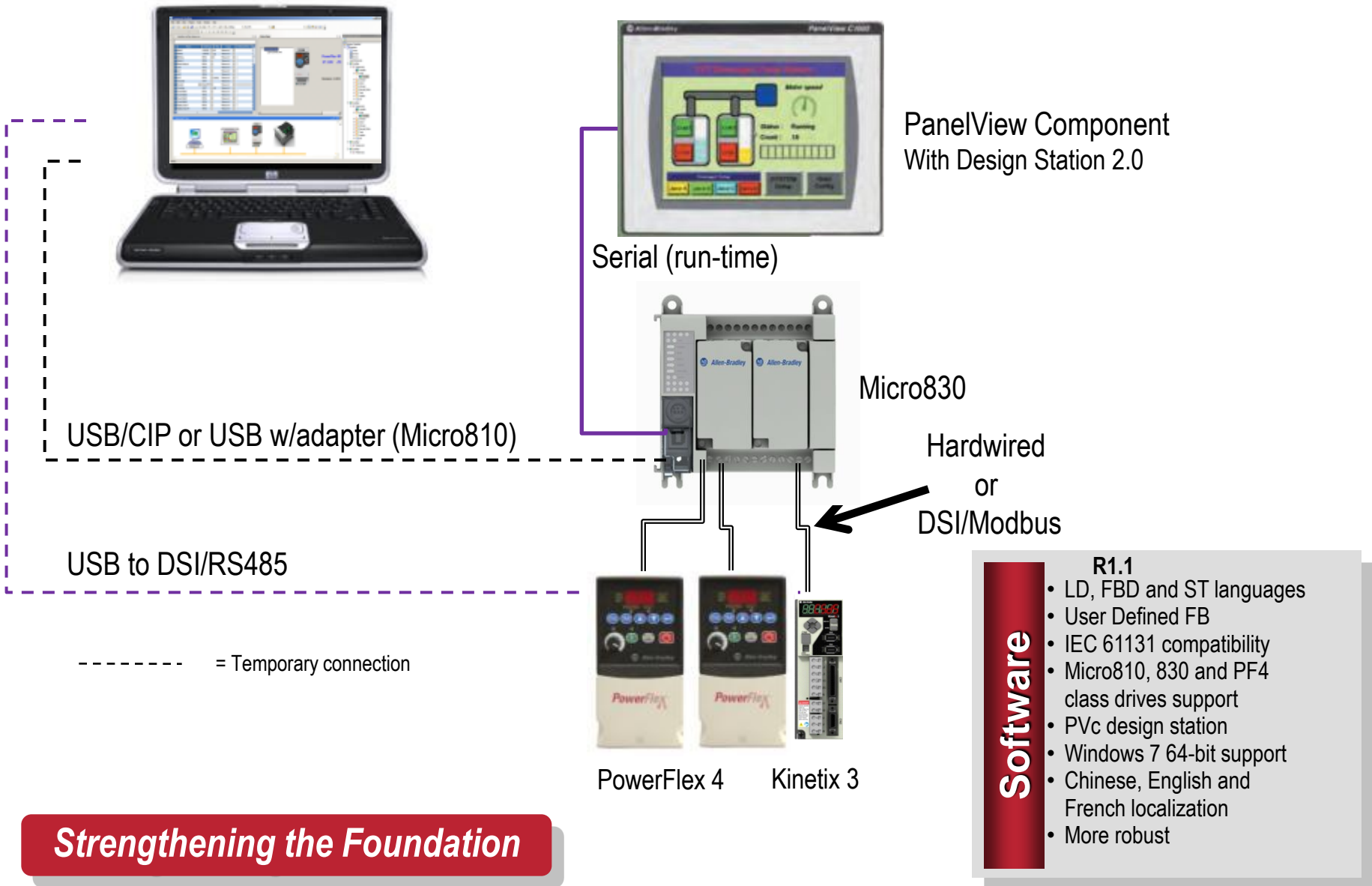
Micro800 Plug-In Modules

- Allows for highly customizable hardware configurations by extending the controller's I/O and communications
- Plugs into front of controller without increasing panel space
- Wide range of plug-Ins available such as analog I/O, serial ports, motion, DeviceNet scanner, GPRS modem, etc.
- Additional modules in future from Encompass partners



Changes the “Personality” of the Base Unit Controller

Connected Components Release 1.1

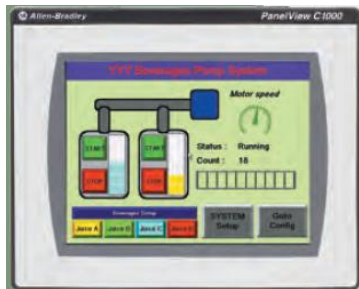
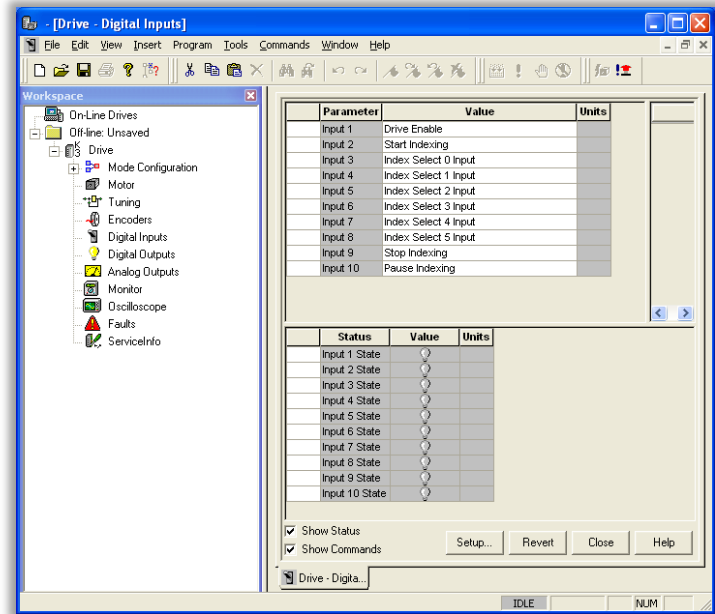


R1.1 Features (AFC December 2011)

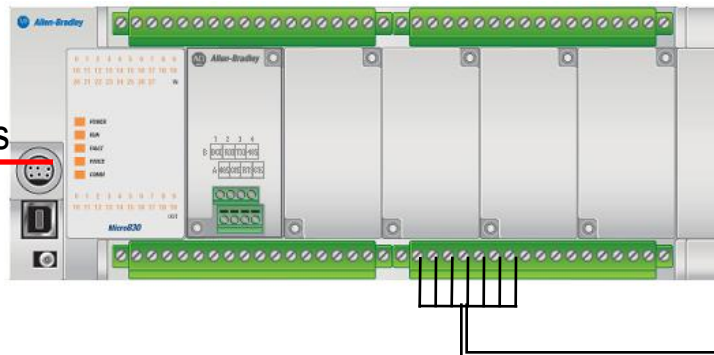
- Windows 7 64-bit support
- Chinese and French localization
 - English user manuals
- Smaller download footprint for R1 upgrade
- (Global) variable export
- User-defined function block debugging
- Improved download infrastructure (potentially improved speeds)
- Major R1 Anomalies Fixed
 - Ladder editor copy/paste
 - Debug view stays in frame
 - Fewer errors/steps to get into debug
 - Improved Save-As
 - Fewer exception faults

Current Micro800 Motion Support - Discrete I/O

- Discrete I/O interface to one or more Kinetix 3 servo drives
 - Micro800 user application sets digital I/O to start, stop and set index positions
 - Refer to [Kinetix 3 Component Servo Drives](#) manual
- Kinetix 3 configured via Ultraware
 - Map digital inputs to index positions
 - Define index positions



Modbus

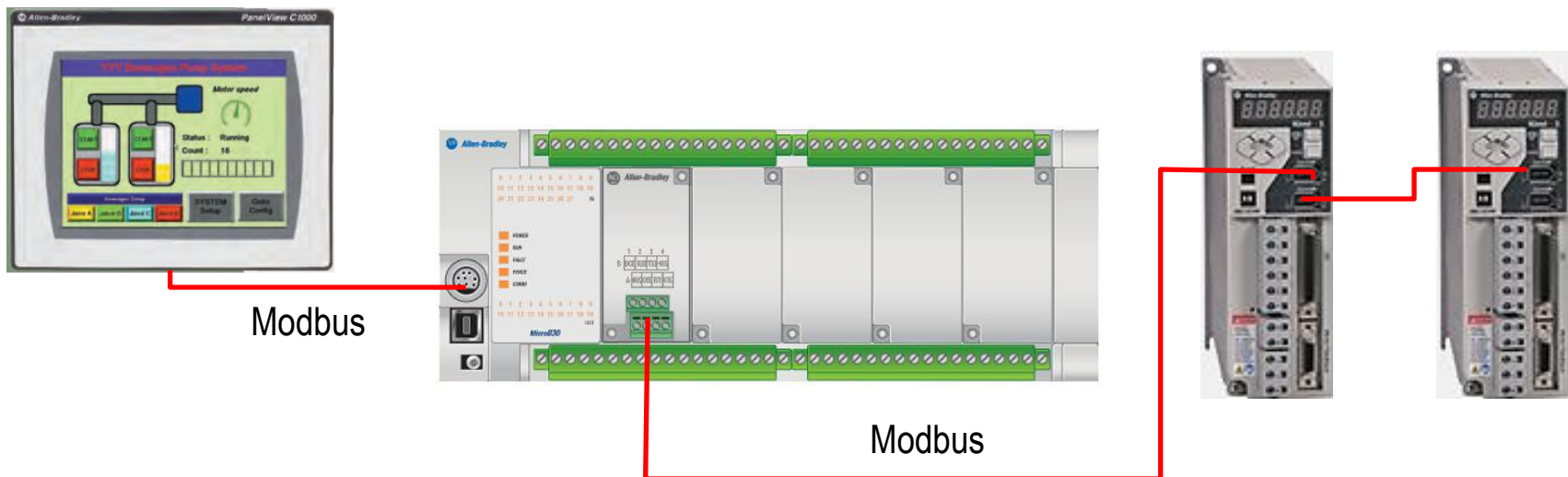


Digital I/O



Current Micro830 Motion Support - Modbus

- Modbus communications to one or more Kinetix 3 servo drives
 - Micro830 user application uses MSG_MODBUS instructions or K3-specific UDFBs (available from Sample Code website) to control servo drives in indexing mode.
 - Refer to [Kinetix 3 Host Commands for Serial Communication](#) manual for Modbus commands and parameters.
- Micro830 controller requires serial port plug-in configured as Modbus RTU Master
- Kinetix 3 drives configured via Ultraware using 2090-CCMPDS cable

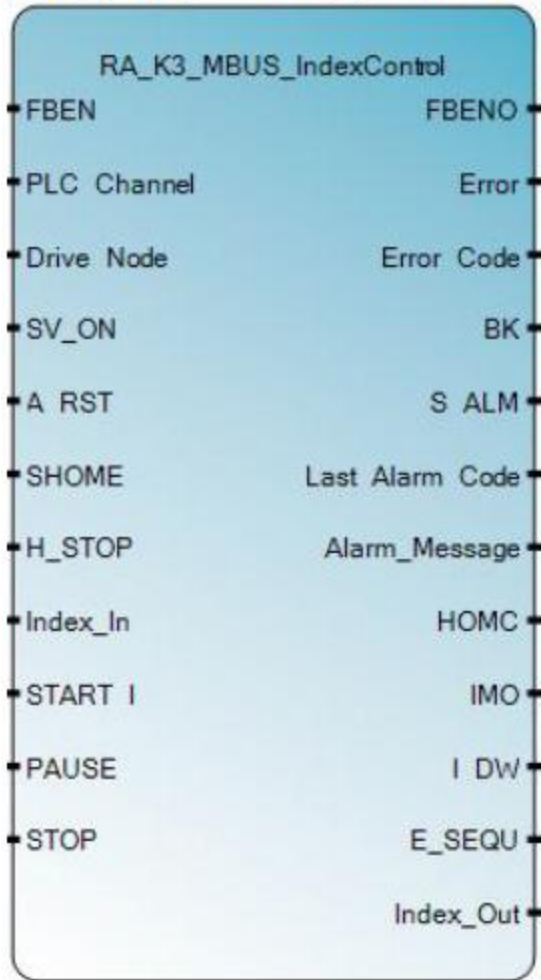


Available K3 UDFBs from Sample Code Website

- **RA_K3_MBUS_OperationMode** – Reads the operation mode of the drive and change it if the input is different. This function block continuously updates the output values on a 10 second cycle, but will only write the new mode to the drive when triggered.
- **RA_K3_MBUS_IndexControl** – Provides simple index control and status for the drive. The default operation runs continuously with a 100 millisecond cycle time.
- **RA_K3_MBUS_WriteIndex** – Provides simple index configuration for one index at a time in the drive. This function block operates on a trigger, but will take more than one scan to complete.
- **RA_K3_MBUS_ReadIndex** – Reads the current index configuration for one index at a time in the drive. This function block operates on a trigger, but will take more than one scan to complete.
- **RA_K3_MBUS_VelocityControl** – Provides simple velocity control and status for the drive. The default operation runs continuously with a 100 millisecond cycle time.
- **RA_K3_MBUS_ChangeVelocity** – Updates the Preset Velocity values for the drive. This function block continuously updates the output values on a 10 second cycle, but will only write the input values to the drive when triggered.
- **RA_K3_MBUS_Jog** – Assuming that the drive is ready but not enabled, this function block will allow the user to jog the drive forward and backward at the default jog speed, without changing the operation mode of the drive.

RA_K3_MBUS_IndexControl UDFB

RA_K3_MBUS_IndexControl

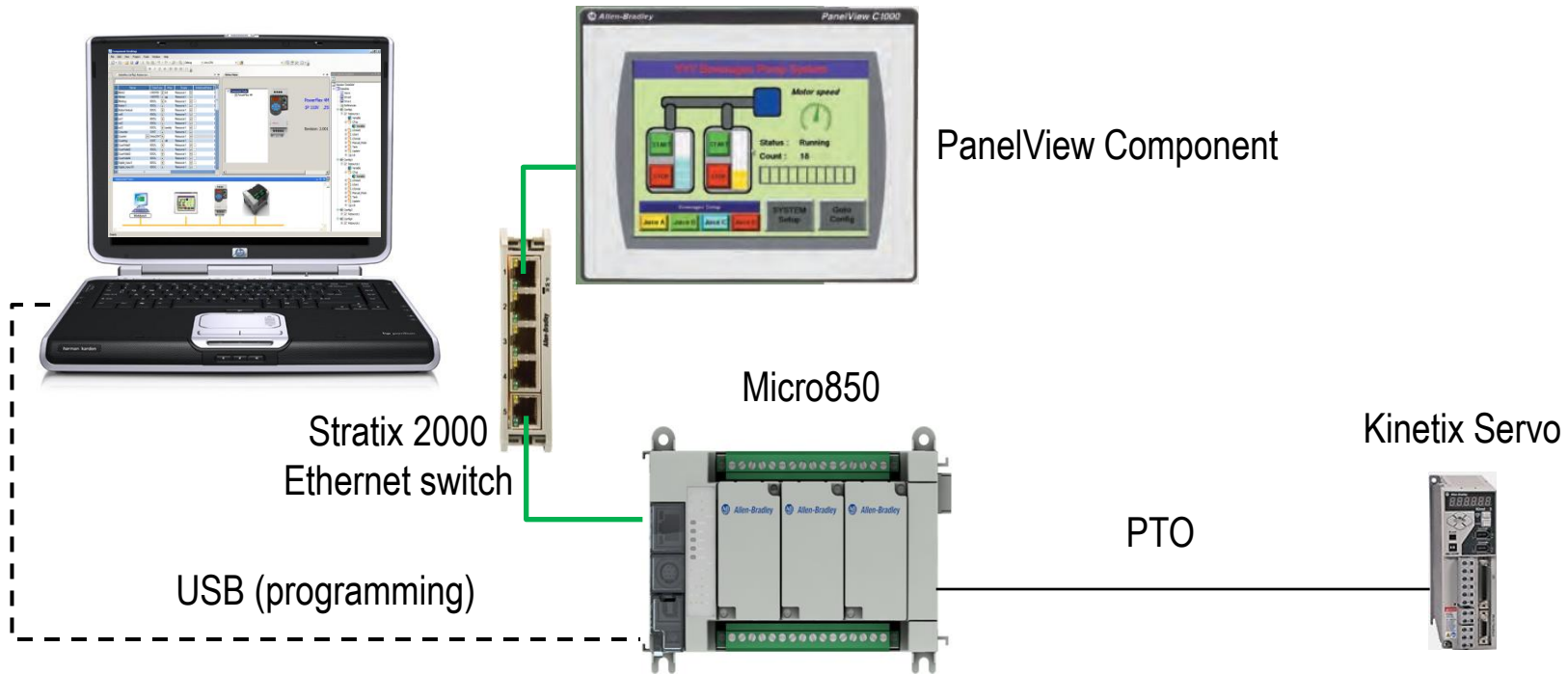


Description:

Every 100 milliseconds, this function block will update the command word in the drive based on the inputs, and update the outputs based on the drive status word.

<http://samplecode.rockwellautomation.com>

What's new in R2?



Software

- Micro850 support (24 or 48 point base with Ethernet)
- Controller password
- 2085 I/O support
- PLCopen motion (Micro830 and 850)
- PowerFlex 7 drives and support

Conveyor application example for the lab



Now let's take a look at a simple program for a conveyor application for loading freight. A motor drives the conveyor belt moves the item(s) until the item can be unloaded. Depending on the type of freight, the speed of the conveyor is adjusted

What controls do I need for this application?

- A motor
- A drive controller for the motor
- A small PLC to coordinate the start/stop/speed change of the conveyor
- An HMI – may be a simple switch(es) or touchscreen display.

LISTEN.
THINK.
SOLVE.®