Chapter 10

PROGRAMMABLE LOGIC CONTROL FOR MODERN MATERIAL HANDLING SYSTEMS

Dr. Jean-Luc Cornet

Conveyor Dynamics, Inc.
1111 W. Holly St.
Bellingham, Washington 98225

As modern conveyors are increasing in size and complexity, their control requirements are becoming more and more critical. In addition to the safety and equipment protection requirements, control systems are now called to implement intelligent drive controls derived from dynamic analysis modelling, as well as to provide fast and accurate diagnostics while keeping operator controls simple. These requirements led to the development of a new and innovative approach to control system development. This paper describes the control philosophy used by Conveyor Dynamics, Inc. (CDI) and some of the key features of our approach to the architecture of control systems.

INTRODUCTION

Control systems for crushing and conveying applications have traditionally relied on hard-wired relay panels and simple ladder logic programming of Programmable Logic Controllers (PLC). In most applications, PLC's were used as relay replacement devices, and no effort was made to capitalize on their more powerful features. The increased requirements for intelligent controls and communication capabilities have made these well established methods unable to address the needs of modern conveyor technology.

The complex control requirements of modern drive and brake systems, the development of mine-wide PLC networks, and the ever increasing needs for fast and accurate diagnostics have created the need for well structured systems and new methods of development.

This paper describes the internal structure of the control systems developed by CDI over the last several years, and some of the key features responsible for their success.

CONTROL PHILOSOPHY

The general control philosophy aims at providing maximum safety, minimizing nuisance trips, simplifying operator controls and providing fast and simple diagnostics. This goal is achieved by a structured approach to the logic definition process and to the PLC programming task.

The logic definition is simplified by the use of structured programming. The control definition is developed using PC based tools, without a specific PLC hardware platform in mind. The complete control process is developed in flowchart mode with a Computer Aided Design (CAD) program. The project Input/Output (I/O) list and the internal register and flag allocation is developed with a data base program. The PLC memory is then allocated, and the flowcharts translated for a specific PLC system. In addition, special techniques are used throughout the control logic development to enhance the system availability, increase its safety, and minimize nuisance shutdowns.

This approach frees the control engineer from the limitation of any given PLC hardware, and allows him to think in much broader terms. The independence from a specific PLC platform offers the advantage of portability. A proven program can be ported to a different PLC and used in new applications, resulting in increased reliability, faster commissioning, and reduced development cost. This approach also allows the control engineer to expand his experience without the need for constant new learning curves.

STRUCTURED PROGRAMMING

The logic definition is simplified by the use of structured programming. This is achieved by separating the system into its elementary components and creating a hierarchical control structure within the system (control tree).

The control architecture is developed along equipment lines. The control task is divided into three or four levels of control and is expressed in block diagrams (Fig. 1). The hierarchical levels are defined starting at the mine level and extending down to single components like pumps and motors.

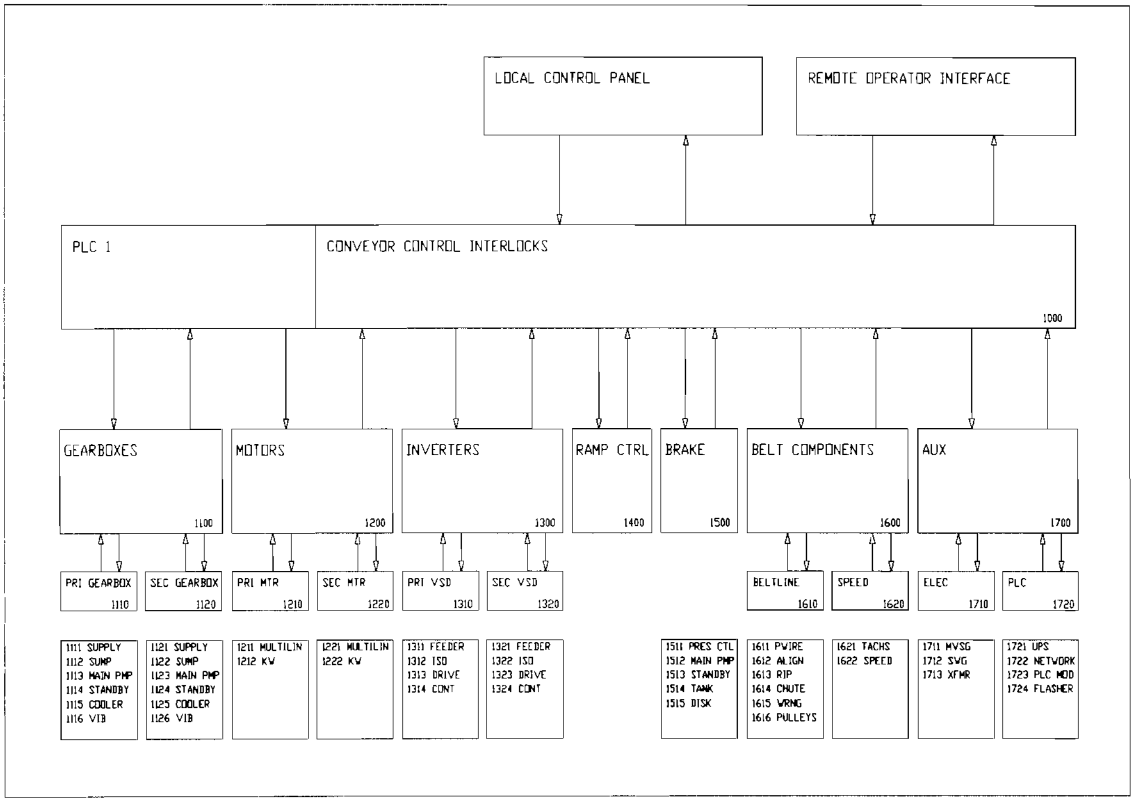

Figure 1 Typical conveyor block diagram showing the control tree structure

Each component is assigned a unique link number. The link number is used throughout the development effort to identify and separate each I/O sub-system. It is used to organize the PLC allocation and to identify each independent control task.

Each control task is composed of one independent logic block, responsible for the interlock, sequencing and monitoring of the elements at the next lower level. Each task is associated with a program block. This block is numbered along equipment lines whenever possible.

Special control, status and alarm flags are defined to communicate between associated control tasks.

As an example, a typical gearbox lubrication system will be defined as an independent unit with the following control and status flags:

- Lube on command (control)
- Lube Ready (status)
- Lube stopped (status)
- Lube Running (status)
- Lube malfunction (alarm)

The control logic for this gearbox lubrication system is programmed to simply follow the "lube on" command with a single interlock with the "lube malfunction" flag. The lube on command is manipulated from the next control level (conveyor control level) to turn the lubrication system on or off depending upon the conditions (starting, running, creep, etc..). The status flags are used at the next higher control level to interlock and sequence independent components.

The alarm flags are used in local interlocks to keep each sub-system independently protected. In addition, they are combined together in common alarm flags. These combined alarm flags are then used at the higher control levels to provide system wide interlocks. At the highest control level, a single conveyor malfunction flag is used to protect a conveyor from a malfunction in any of its components.

This control philosophy allows a simple approach to each control task by isolating each component of the system. It keeps each block small and independent, which allows the complete simulation of the final program prior to commissioning. This results in shorter commissioning time and a minimum number of software bugs. Furthermore, equipment modifications can be more easily performed in the field since only small and easily identifiable blocks of the complete program need to bemodified.

The hierarchical control structure can easily be expanded across a network. Each control block resides in its native PLC, and only the conveyor status and control flags need to be carried across the network. This feature is commonly used to provide automatic sequencing of conveyor routes across multiple PLC's. It also keeps each PLC independent, as only basic start/stop control information is required from the network. The required control information can be provided by local control panels in case of network failure, which allows the easy implementation of local controls in each PLC.

It is important to note that I/O information used for interlocking purposes should never be sent over a network. This would defeat the purpose of the structured approach and lower the system safety. As an example, consider the case of a PLC network controlling multiple conveyors on the same route. Each conveyor should be equipped with a low-speed switch at its tail station. This switch should be directly connected to the upstream conveyor PLC to detect downstream conveyor shutdowns. This is the only way to avoid material pile-up at the transfer chutes and insure system safety in case of network failure.

PROCESS FLOWCHARTS

A graphical description of the logic process is essential for good diagnostics and future program updates. It allows the operator to visualize the process with little technical knowledge of the PLC programming language. This is very important during simulation and commissioning, but is even more so for future system modifications when the original design engineer may no longer be available.

All the control processes are developed on flowcharts with a CAD program. A typical flowchart is shown in Fig. 2.

Each program block is represented by one or several flowcharts. If more than one flowchart is required for a given block, each separate flowchart is given a segment number. This segment number is also shown in the program printout to help in locating specific statements.

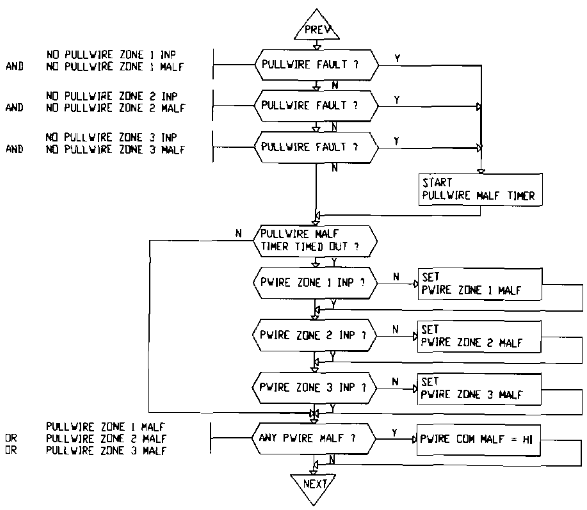

Figure 2 Typical logic flowchart

Flags are only manipulated through SET, RESET and EQUAL statements. All register operations (math, comparisons or transfer functions) are shown in the simplest possible way. If more than one flag is involved in a decision process, the control logic is shown on the left of the flowchart in boolean form using AND and OR statements.

I	CL	Digital input	- normally closed
I	OP	Digital input	- normally open
I	AN	Analog input	
Q	OF	Digital output	- maintained
Q	PLS	Digital output	- pulse
Q	AN	Analog output	
C	CD	Control flag	- command
C	ST	Control flag	- status
C	CM	Control flag	- common alarm/malfunction
F	CD	Internal flag	- command
F	ST	Internal flag	- status
F	AL	Internal flag	- alarm
A	A1	Alarm flag	- level 1, display only
A	A2	Alarm flag	- level 2, disable component
A	SP	Malfunction flag	- level 0, start permissive
A	M1	Malfunction flag	- level 1, empty motor stop
A	M2	Malfunction flag	- level 2, immediate motor stop
A	M3	Malfunction flag	- level 3, controlled brake stop
A	M4	Malfunction flag	- level 4, emergency brake stop
A	M5	Malfunction flag	- level 5, brake dump stop
T	CD	Timer	- control
T	AL	Timer	- alarm
R	VAR	Analog variable	
R	CST	Analog constant	
D	BUF	Input buffer	- floating-point/double integer
D	TMP	Temp register	- floating-point/double integer
D	CST	Analog constant	- floating-point/double integer

Table 1 Typical list of categories used in PLC allocation data base

PLC ALLOCATION DATA BASE

A fast and direct access to the PLC memory allocation table is also essential for good diagnostics and future program updates. It allows the operator to rapidly locate all the allocated registers and alarm flags without searching through the program. This is specially important when an operator interface is used.

The PLC allocation is developed with the help of a specially tailored data base program. The data base provides a complete description of the I/O's, flags, timers, registers and setpoints used in each PLC.

Each record is identified by a descriptive type field. This field is made of two parts. The first part identifies the record category (Input, Output, Flag, etc..). The second part identifies the record type within a category. The number of categories and record types used varies with the complexity of the project. As an example, the categories defined for a recent project are shown in Table 1. This project involved five levels of malfunction, each one with a corresponding stopping control algorithm. A typical data base printout is shown in Fig. 3.

Flags are divided into three categories: Control Flags, Internal Flags and Alarm Flags. Control Flags are used to communicate between control levels and are used in the interlocking logic. Internal Flags are only used within a given level. Alarm Flags are combined to generate the common alarm flags, and are used for diagnostics. Any flag may be used in a graphic screen on systems equipped with graphic stations.

CAT: 0=TITLE 1=DIG INPUT 2=DIG OUTPUT 3=ANA INPUT 4=ANA OUTPUT 5=RET FLAG 6=FLAG 7=TIMER 8=REG 9=PREG

	VARIABLE			ALLOCATION				VALUES		
LINK	DESCRIPTION	CAT	TYPE	PLC	RS B	REG	BIT	VALUE	UNITS	INT
1500 0	BRAKE SYSTEM									
1500 11	CONTROL STOP ENABLED	5	C CD	1			1190			
1500 20	READY	5	C ST	1			1193			
1500 21	STOP RAMP COMPLETED	5	C ST	1			1191			
1500 30	COMMON ALARM	5	C CM	1			1760			
1500 31	COMMON MALF	5	C CM	1			1761			
1511 0	BRAKE HYD PRESS CTL									
1511 1	HIGH PRESS SWITCH	1	I CL	1	07		95			
1511 1	CALIPER PRESSURE	3	I AN	1	15	7		4-20	MA	3000
1511 1	PROPORTIONAL VALVE	4	Q AN	1	17	2		+/-10	VDC	
1511 40	PRESS SIGNAL OK	5	F ST	1			1200			
1511 41	PRESS TARGET REACHED	5	F ST	1			1203			
1511 60	CAL PRESS INV	5	A A2	1			1785			
1511 71	CTL PRESS LO	5	A M2	1			1789			
1511 72	CTL PRESS HI	5	A M2	1			1790			
1511 50	BRK PRESS MALF TMR	7	T AL	1		454	454	3	SEC	30
1511 10	CALIPER PRESS REG	8	R VAR	1		1350				
1511 11	TARGET PRESS REG	8	R VAR	1		1351		0-1500	PSI	
1511 30	INI BRK PRESS REG	8	R CST	1		1120		250	PSI	250

Figure 3 Typical PLC allocation printout

The data base can be printed with two different sort orders. The first one is called "Allocation By Link" and includes all the categories. The second one separates I/O's, flags and registers into three printouts and is called "Allocation By PLC".

In the "Allocation By Link" printout, the data base is ordered according to the link number defined in the block diagrams. This presentation is the most useful during the development phase, or whenever working on a specific component. It provides a concise list of all the PLC I/O points, flags, timers and registers related to a specific piece of equipment.

In the "Allocation By PLC" printout, the data base is ordered by PLC addresses. This presentation is most useful when trying to identify an element from its address, or when looking for unused addresses (spares). This printout usually includes the Identification Tag Number assigned by the project. The same tag number is used on the construction and as-built drawings and can be used for field identification of I/O points.

SPECIAL TECHNIQUES

Nuisance shutdowns caused by electrical noise or poor connections are a common problem in a mining environment. Special noise prevention techniques can be used to minimize this problem and increase the system availability. The following paragraphs describe some of the most useful techniques.

Digital Input Filtering

Nuisance trips and false alarms are reduced by the systematic use of filtering techniques on all digital inputs. Digital inputs are filtered through variable timers to eliminate shutdowns induced by electrical transients and bouncing contacts. Actual inputs are never used in the interlocking logic.

All digital inputs are checked for both on and off conditions to detect faulty instruments, frozen switches and operator jumpers. When a valid fault is detected, an internal malfunction flag is latched. This flag is unique to the faulty input and remains latched until an operator reset is performed.

Analog Input Filtering

Analog inputs are first checked for out of range conditions and then filtered through a digital filter (moving average) to always insure noise free values.

All critical control input signals are checked against backup signals, when available, and against relevant digital inputs in order to detect inaccurate transducers. Any analog input found defective is disabled and a unique alarm flag is latched for diagnostic purposes.

Disabled analog inputs are ignored by the control logic, and alternate control options are used if required. Actual inputs are never used in any control process. Only valid filtered signals are used for level checks or process control.

As in the digital input processing, an internal malfunction flag is latched whenever a fault is detected. This flag is unique to the faulty condition and remains latched until an operator reset is performed.

The filtering and checks performed on analog inputs is critically important for any signal used in a control loop since a noisy signal can result in instabilities of the control loop and in possibly destructive action by the control system (Fig. 4).

Controlled PLC Scan

The PLC programs are divided into separate sections corresponding to each state of operation (i.e. starting, stopping, etc..), with only the required sections scanned at any given time. This approach greatly simplifies the logic definition process, since the logic required for each particular state can be defined independently.

When a large number of analog inputs are used, the inputs are scanned and processed on a priority based system, with the critical inputs processed more often.

This approach results in extremely fast scan time (typically 20 msec or less) even for very long and complex programs.

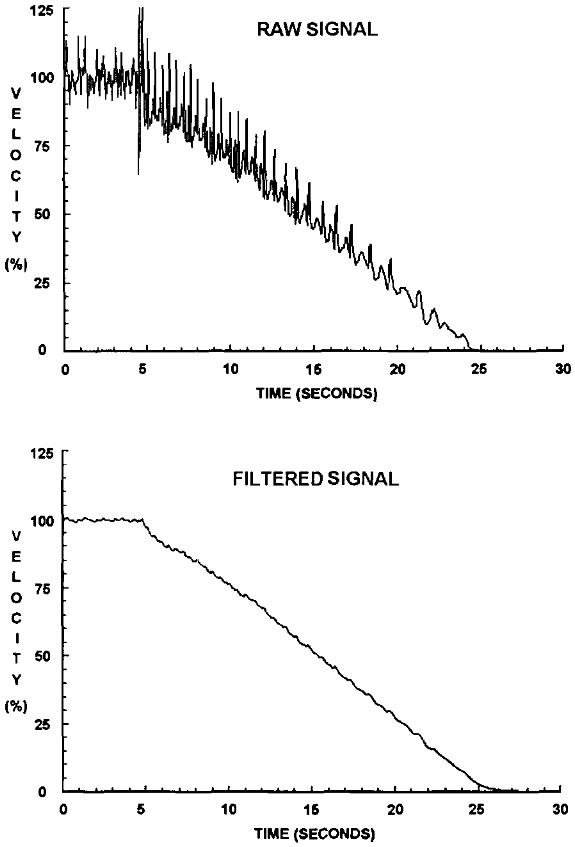

Figure 4 Effect of analog signal filtering on the signal from an improperly mounted tachometer. The first plot shows the raw signal, the second plot shows the same signal after digital filtering. The noise has been removed without altering the original signal shape. In this example, the noise was induced by improper alignment of the tachometer shaft with a pulley shaft. The noise frequency was about 1 Hz, well within the control system response time, and was inducing loop instabilities. The field recording immediately pinpointed the problem. The signal filtering was increased temporarily to stabilize the loop until the shafts could be realigned.

Analog Setpoints

In order to insure the setpoints integrity, their values are automatically set by the program. This task is performed in a special program block which is scanned on a fixed time basis. This prevents loss of data in case of PLC memory loss, and minimizes potential human errors as any modification of a setpoint requires modifying the program itself.

ALARM HANDLING

The proper handling of alarm conditions is essential to the safety of the system. It is also critical to the development of good diagnostics and the elimination of "ghost" shutdowns (a "ghost" shutdown is a shutdown without an apparent cause, it is usually extremely difficult to trace).

Every alarm and malfunction is latched at the time it is detected. An alarm flag remains latched until the alarm has been cleared and an operator reset is performed. The malfunction flags are combined into common alarm flags according to the control tree. Only common alarm flags are used in the interlocking logic. This approach prevents "ghost" shutdowns since only a latched flag can trigger a shutdown.

In addition, the hierarchical structure used for the common alarm flags provides an easy diagnostic path down to any faulty input. When combined with a properly designed graphical operator interface, this approach results in very fast and simple diagnostics.

MULTIPLE ALARM LEVELS

The control of modern conveyor drive and brake systems requires multiple control algorithms to account for various failure conditions [1]. This requires the alarms and malfunctions to be classified into several levels according to their severity. Three to eight levels may be required depending upon the complexity of the application. The various alarm levels are identified during the system design, and each level is associated with a backup control algorithm.

The following example shows the alarm levels used on a recent conveying project. This project involved the control of several downhill and overland conveyors on the same route. All the conveyors are driven by inverter drives and use mechanical brakes. Two levels of alarm and five levels of malfunction were required on the downhill conveyors to account for the various failure modes of the drive and brake systems. The following levels were used:

- Alarm level 1

This is the lowest level of alarm. It is for indication only. The PLC does not stop any equipment.

- Alarm level 2

This alarm level indicates a fault on non critical equipment or on an analog transducer. The PLC disables the faulty equipment and switches to back-up components if available. Disabled analog inputs are ignored by the control logic, and alternate control options are used if required.

- Start Permissive Alarm

This alarm level indicates a problem with components required during start-up. The PLC does not stop a running conveyor but will prevent a conveyor start.

- Malfunction level 1

This is the first level of malfunction. It indicates an impeding failure. The PLC stops the feed to the conveyor and starts a delay timer. When the conveyor is empty, the PLC initiates a motor controlled shutdown of the conveyor.

- Malfunction level 2

This malfunction level indicates a critical problem which requires an immediate shutdown of the conveyor. The PLC initiates a motor controlled shutdown of the conveyor.

- Malfunction level 3

This malfunction level indicates a critical problem which requires an immediate shutdown of the conveyor using the mechanical brake. It means that the motor is no longer available. The PLC initiates a brake controlled shutdown of the conveyor.

- Malfunction level 4

This malfunction level indicates a critical problem which requires an immediate shutdown of the conveyor using the emergency braking circuit. It means that normal brake control is no longer available. The PLC initiates an emergency brake shutdown of the conveyor.

- Malfunction level 5

This malfunction level indicates a critical problem which requires an immediate shutdown of the conveyor using a brake pressure dump. It means that the conveyor is out of control and requires maximum brake torque. The PLC initiates a pressure dump brake shutdown of the conveyor.

CONTROL MODES

Several modes of operator control are always defined for a system. The various modes are provided to insure maximum system availability and to help with sophisticated maintenance requirements.

The operating modes are selected with key selector switches located on each local PLC cabinet. Key switches are used to prevent unauthorized personnel from changing the selected mode of operation. The key switches are internally interlocked by each PLC. A mode change request is only accepted when it is safe to do so. The following modes are the most commonly used:

- Remote PLC mode
- Local PLC mode
- Creep mode
- Manual modes

A short description of the various control modes is presented below.

Remote Mode

This mode is required when a conveyor needs to be controlled from a remote location. This is usually the case when multiple conveyors are controlled from a central operator station. It is the normal mode of operation of most large systems.

In this mode, all the conveyors, crushers, feeders and auxiliary equipment in the system can be controlled by a single operator. Only fully interlocked equipment starts and stops can be initiated in this mode. This is the only mode which can provide centralized graphical status and alarm information to a single operator.

Local Mode

This mode provides complete local control of a conveyor from a control panel located on its PLC cabinet. This is the normal mode of operation in single conveyor systems.

In multiple conveyor systems, this mode is designed for conveyor maintenance. Each PLC is always configured and programmed as a completely independent unit and full conveyor interlocks are always maintained in this mode. A local mode is always implemented in networked systems, as it enables the operator to maintain production in the event of a network failure.

Creep Mode

This mode is designed for conveyor maintenance or whenever it is necessary to run a conveyor at low or variable speed. Full conveyor interlocks are always maintained in this mode.

Manual Modes (Drive and Brake)

These modes are required for conveyors equipped with complex brake or drive systems. They are designed to provide complete manual control of the equipment. Their main purpose is to simplify maintenance of sophisticated equipment.

Manual modes can only be selected when in local PLC mode with the conveyor stopped. Note that this mode can be dangerous since system interlocks have to be disabled to enable manual control of the equipment. For instance, a brake manual mode allows the release of the brake on a loaded downhill conveyor. This would result in uncontrolled acceleration of the conveyor, leading to potential damage to the system.

OPERATOR STATIONS

A good operator interface has to be simple in function to minimize training time and operator errors. It should also provide a positive feedback of operator action, and sufficient concise information for the operator to act properly in emergency situations. The requirements for a good graphical operator interface are simple controls, concise status information, and fast alarm diagnostics.

A PLC system can be interfaced to numerous control stations, from simple push button stations or mimic panels to full graphic systems. In our experience, a combination of simple operator control panels (push button or function keys), automated system sequencing and computer graphic diagnostics provides a good balance between the simple control needed for easy operator training and the sophisticated diagnostics required for efficient fault finding.

In single conveyor application, a local control panel with push buttons and a small operator interface screen is recommended. In large systems, an industrial computer with function key controls and graphic screens for status and diagnostic information is recommended. In most cases, a supervisory computer linked to the PLC and running a specialized information program is also recommended.

Diagnostics

The need for fast and comprehensive diagnostics cannot be overemphasized in light of the large cost of production downtime. The complete development of a good diagnostic system can often be paid back with just a few hours of avoided downtime. Good diagnostics also help organize preventive maintenance and proper mobilization, therefore increasing plant availability.

Whenever possible, we recommend the use of icon based graphical interfaces. The graphical approach is logical in nature, with multiple screens organized along equipment lines, and allows a complex system to be simply represented.

When developing a graphic system, the graphic screens are structured along the same hierarchical control structure developed for the PLC logic (control tree). The graphic screens should not be overcrowded, and should provide visual indication of the problem area, allowing to rapidly zoom in to the faulty component (Fig. 5).

Full diagnostics of any malfunction condition must be available to the operator in less than three or four keystrokes. The alarm structure used in the PLC programming allows the operator to quickly find any latched alarms or malfunctions. The common alarm or malfunction flags are indicated on the overview screen. the operator needs only to follow the path indicated on each successive screen to find the faulty component. In addition, the operator can check the status of each I/O point to see if the problem is corrected, and reset the PLC from the screen that shows the alarm.

A well designed graphical system should be simple to learn, resulting in a faster learning curve and shorter operator training time.

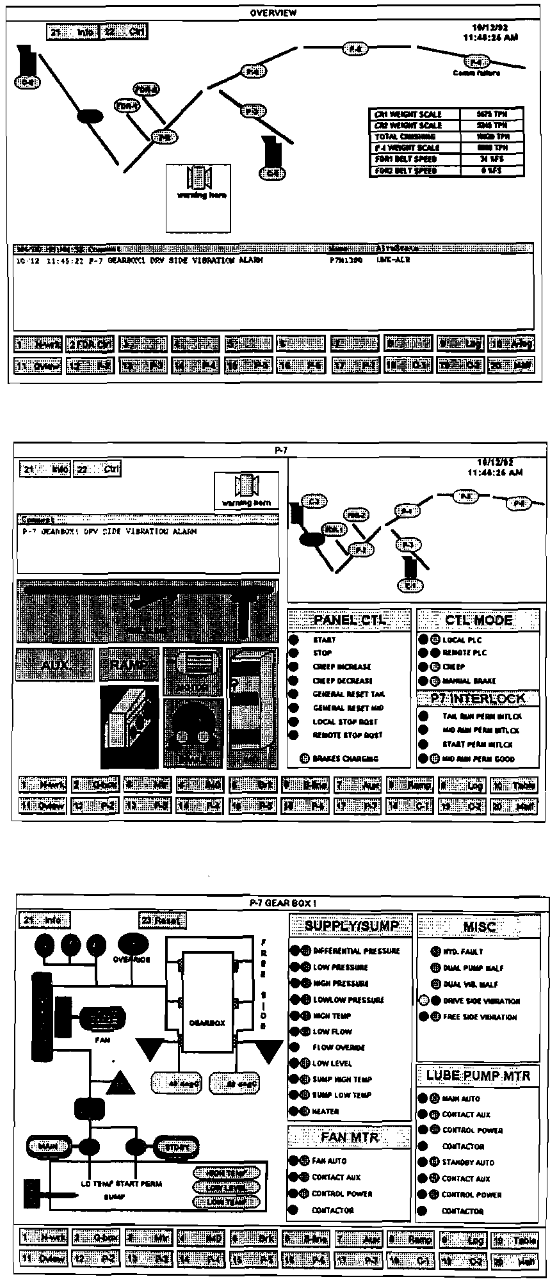

Figure 5 Example of graphic screen sequence. The first screen shows the complete mine overview, the second screen shows a single conveyor, and the third screen shows a single gearbox. The gearbox screen shows all the gearbox I/O status and malfunction. Each successive screen is accessed with a single keystroke.

When the project cannot justify the cost of a dedicated graphic station, small operator interface terminals should be used at each PLC station. Such terminals are inexpensive and typically offer alarm logging and time stamping for about 100 alarms. Diagnostic panels based on indicator lights are never recommended because of their inflexibility and because of the limited information they provide.

Supervisory Computer

The supervisory computer is used to provide additional status, alarm and historical information. It is never required for the operation of the system. A mine information program provides the basic building block and data base management from which better diagnostics can be extracted.

In its most basic form a Mine Information System (MIS) provides:

- on screen alarm messages

- alarm acknowledgement

- graphical status information

- printed alarm log

- daily report of activities

The data base generated by the MIS can further be used to generate historical and statistical data and forms the base needed to develop a preventive maintenance program based on past failure analysis, analog data trending, and future failure prediction.

The MIS is an invaluable help during the commissioning phase of the project by providing the start-up engineers with fast and accurate diagnostics of the system performance.

PROGRAM SIMULATION

A very important part of our approach is the use of simulation methods to check every logic control module. This ability to simulate every module individually is a direct result of the structured programming technique, and is an essential element in the success of this modular approach.

The simulation work includes full testing and simulation of all the logic functions and control algorithms of the PLC program prior to site delivery. It includes real time simulation of the starting and stopping algorithms using specially designed simulator panels to emulate the conveyor response. These simulator panels incorporate small controllers with analog capabilities, and are programmed to provide analog feedback during the simulation of complex control algorithms.

The program simulation enables the verification of both the conceptual control logic and the final written program in an easily controllable office environment. This minimizes commissioning time by several orders of magnitude by eliminating most software bugs and providing the start-up engineer with a large degree of confidence in the program. This confidence is invaluable during commissioning as it speeds up the diagnostics of field wiring and instrumentation problems. The simulation also minimize the potential risks to the equipment of using an untested program.

In a recent commissioning effort, a 92K program controlling several conveyors with soft-start and soft-stop controls was successfully completed in less than 8 hours. The short commissioning time obtained with such an approach can substantially decrease the cost of commissioning of a new system, or the down time of an operating plant during a change-over.

DOCUMENTATION

The documentation provided with a control system is extremely important. Good documentation is critical for future modifications of the equipment and is essential during the commissioning phase of a project.

The typical documentation provided by most PLC built-in software consists of a bulky printout of the program. This is inadequate because it does not describe the control process nor does it not provide any information about the PLC memory or flag allocation.

The technical documentation provided for each PLC station includes a description of the control architecture defined for each system (block diagram), a list of all the program blocks used, a data base description of the PLC allocation, a description of the control logic in flowchart form, and a compressed printout of the ladder logic programs. This results in a structured and very complete technical documentation manual.

A separate user manual is also developed to describe the process and the operator interface in non-technical terms.

The combination of these two manuals provides a complete documentation package addressing the needs of both operators and field technicians. These manuals also form the necessary basis for simple and effective training of the operator and maintenance personnel.

FIELD TESTING

Another essential useful technique is the use of data acquisition systems during commissioning to obtain instantaneous recordings of system performance (Fig. 6).

Field testing of the completed system is always performed. It serves to validate both the mechanical design and the performance of the control system. The tests are done according to a pre-established sequence, starting with empty tests and slowly increasing the load to maximum design tonnage. All the control algorithms implemented in the PLC, including the various back-up control algorithms, are fully tested under various conditions. During each test, all the important parameters of the system are recorded simultaneously, and the result of each test is carefully evaluated before proceeding to the next stage.

Such testing provides the only mean to insure that the implemented control algorithms are working exactly as designed. It can also pinpoint rapidly potential problems in the equipment (Fig. 4). Finally, it provides the operation personnel with a high level of confidence in the new system and maximizes safety, as potential problems can be discovered early.

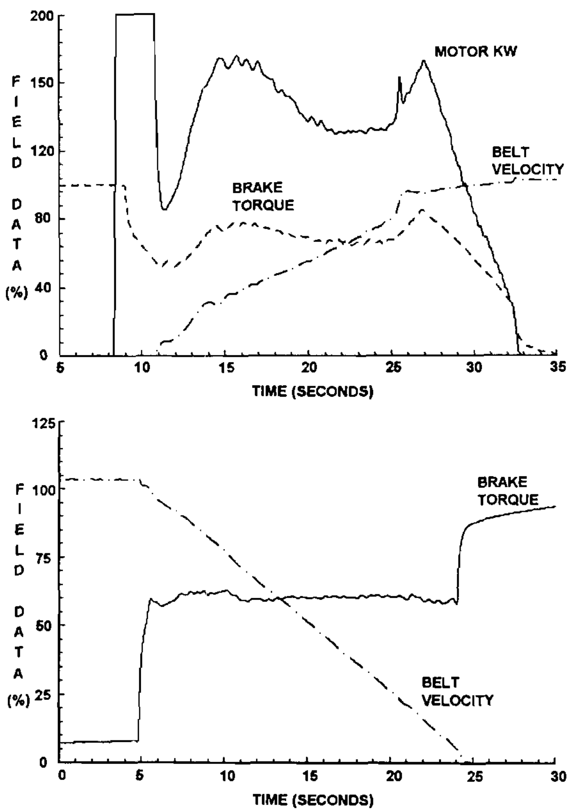

Figure 6 System performance recording using CDI data acquisition system. These field recordings show the starting and stopping sequences of a fully loaded downhill conveyor with fluid coupling drive and hydraulic disk brake. The brake is used to control the velocity during start-up and shutdown. Motor KW, conveyor velocity and brake torque are recorded.

The information provided by such tests is invaluable during the debugging phase of a start-up. It is also extremely useful for future system maintenance and trouble-shooting as it provides the initial signature of the new system.

CONCLUSION

Our modular approach to PLC programming was developed over several years of experience with the control of crushing and conveying systems. The principal motivation for its development was to simplify operator controls, minimize commissioning time, and increase system availability by providing fast and simple diagnostics. This approach requires an early involvement of the control engineer in the overall project development. Most of the techniques described in this paper have to be implemented in the logic definition from the very beginning of a project, and cannot be easily added later.

A structured approach may slightly increase the development time and the development cost of a control system. However, it is our experience that the higher initial investment is recovered several times over during commissioning and operation of the system, and that this modular approach is essential to a trouble free operation.

REFERENCES

[1] Cornet J. L., 1993, "Requirements for Stringent Specifications for Braking Systems on Regenerative Conveyors", Paper to be presented at SME Annual Meeting, February.