What is a Programmable Controller?

What are programmable controllers and how do they work?

Programmable controllers are often defined as miniature industrial computers that contain hardware and software used to perform control functions. A controller consists of two basic sections: the central processing unit (CPU) and the input/output interface system. The CPU, which controls all system activity, can further be broken down into the processor and memory system. The input/output system is physically connected to field devices (e.g., switches, sensors, etc.) and provides the interface between the CPU and the information providers (inputs) and controllable devices (outputs).

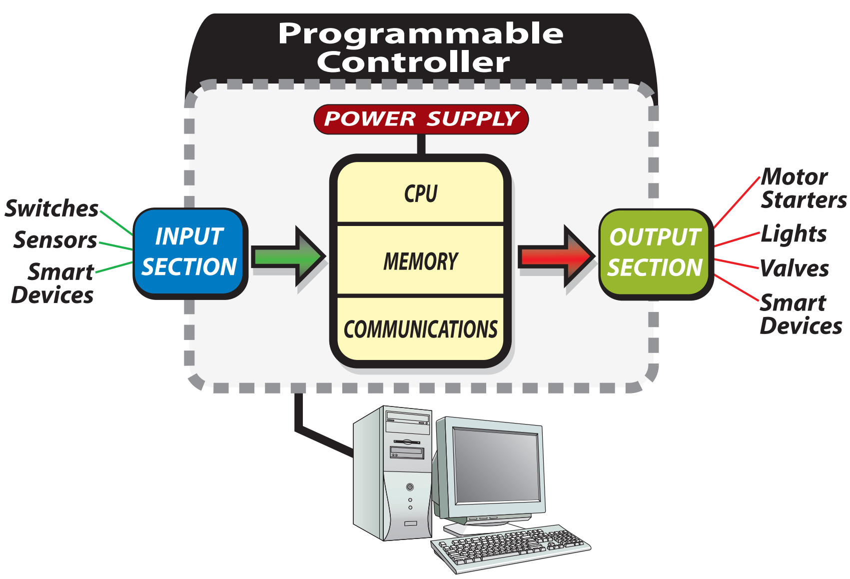
To operate, the CPU "reads" input data from connected field devices through the use of its input interfaces, and then "executes" or performs the control program that has been stored in its memory system. Programs are typically created in ladder logic, a language that closely resembles a relay-based wiring schematic, and are entered into the CPU's memory prior to operation. Finally, based on the program, the PLC "writes" or updates output devices via the output interfaces. This process, also known as scanning, typically continues in the same sequence without interruption, and changes only when a change is made to the control program.

Discrete applications

Programmable controllers are often used to control machines or processes that are sequential in nature, using "discrete" inputs and outputs that have defined states. For example, if a limit switch detects the presence of an object, it provides an "ON" signal to the PLC; if no object is detected, it provides an "OFF" signal. The machine or device typically performs actions based on time or events in a pre-defined order. The expected sequence is typically interrupted only when an abnormal condition occurs.

Process control applications

Programmable controllers can also control continuous processes that use analog I/O. For example, a temperature sensor may provide a variable signal, such as 0-10 volts, based on the measurement of an actual temperature. The controller program monitors the sensed values continuously and operates devices that may also be analog in nature. This could include setting the position of a valve between 0-100% open, or controlling the speed of a motor. Continuous applications are so called because they typically have no defined start or end once they are initiated; they maintain a process in a "steady" operating state.



Today's controllers

Initially, devices that exhibited the attributes discussed here were known as Programmable Logic Controllers (PLCs). This tended to emphasize that the main functionality of these systems was LOGIC operations. As technology has advanced, so have programming languages and communications capabilities, along with many other important features. These developments seemed to demand the definition of a new class of controller, the Programmable Automation Controller (PAC), which combines features of traditional PLCs with those of personal computers.

In the past, size was typically used to categorize controllers, and was often an indication of the features and types of applications it would accommodate. Small, non-modular PLCs (also known as fixed I/O PLCs) generally have less memory and accommodate a small number of inputs and outputs in fixed configurations. Modular PLCs have bases or racks that allow installation of multiple I/O modules, and will accommodate more complex applications. With the emergence of PACs, functionality is the determining factor in categorizing controllers.

Which programmable controller is right for you?

Choosing the most effective controller for your application depends on a number of factors. To begin the selection process, a drawing of the machine or process is a good start. This can help identify field devices and physical requirements for hardware locations. From the drawing, you can determine how many analog and/or discrete devices you will have.

Once the field device requirements and hardware locations are defined, you can review controllers that will meet your requirements. See the Controller Selection Worksheet in this section that will help you work through the considerations for determining the type of controller you will need, regardless of which manufacturers you are evaluating.

PLC vs. PAC vs. PC-based Control

The most common control systems today are the Programmable Logic Controller (PLC), PC-based control, and the most recent addition, the Programmable Automation Controller (PAC). While they each share a few attributes with the others, their differences lie mainly in form factor and functionality.

Programmable Logic Controller

The Farlex Dictionary defines a PLC as follows: "A programmable microprocessor-based device that is used in discrete manufacturing to control assembly lines and machinery on the shop floor as well as many other types of mechanical, electrical and electronic equipment in a plant. Typically RISC based and programmed in a specific-purpose programming language, a PLC is designed for realtime use in rugged, industrial environments. Connected to sensors and actuators, PLCs are categorized by the number and type of I/O ports they provide and by their I/O scan rate."

PLCs excel at sequential logic and basic analog control. Their modularity and ruggedness make them suitable for a wide variety of automation applications.

PC-based Control

With Personal Computer technology booming in the 1980s and 1990s, there was a natural progression to consider using the processing power in these units to solve more complicated applications that extended well beyond the realm of digital and analog I/O manipulation. These more advanced capabilites could be performed far more efficiently by hardware and software native to the commercial personal computer. Examples of these requirements include:

- The need for a Human Machine Interface (HMI) as well as control
- Advanced data manipulation and advanced math functions
- Data exchange with business applications (spreadsheets, ERP systems)
- One or more third-party PC cards, such as those for motion control or vision systems
- Communication with serial or networked field devices
- Storage or access to large amounts of data
- Large number of PID loops (64 or more)
- Open architecture for C/C++ or VisualBasic systems
- Online productivity tools to analyze and improve performance of the process

In a PC-based control system, a standard operating system such as Windows NT supports HMI and control software running on a PC platform, either a readily available commerical model or an industrially hardened unit. PC architecture allows the system to seamlessly support a variety of third-party I/O, specialty motion and vision systems, and field networks.

Programmable Automation Controller

A programmable automation controller is a compact controller that combines the features and capabilities of a PC-based control system with that of a typical programmable logic controller (PLC).

This hybrid arose not only to solve complex applications with the speed and processing power of a PC-based system, but to do it on a platform capable of withstanding the environmental pounding that PLCs have been subjected to for many years.

Ideally, a PAC encompasses the following features:

PLC Feel

- Modular footprint
- Industrial reliability
- Wide array of I/O modules and system configurations

PC Power

- Large memory and fast processing
- High-level data handling and enterprise connectivity
- Extensive communications capability, multiple protocols and field networks

PACs are most often used for advanced machine control, process control, data acquisition and equipment monitoring.

Although each PAC vendor uses their own development environment (IDE) and programming language, PAC networking is typically based on IP and Ethernet.

This class of controller provides more memory capacity and processing power which allows for better data processing capabilities, and connectivity to enterprise business systems from the plant floor.

Additionally, PACs offer the benefit of easy integration for multi-domain systems comprising Human Machine Interface (HMI), discrete control and process control.

Company Information
Control Systems Overview
CLICK PLC
Do-More PLCs Overview
Do-More H2 PLC
Do-More T1H PLC
DirectLOGIC PLCs Overview
DirectLOGIC DL05/06
DirectLOGIC DL105
DirectLOGIC DL205
DirectLOGIC DL305
DirectLOGIC DL405
Productivity Controller Overview
Productivity 3000
Universal Field I/O
Software
C-More HMI
C-More Micro HMI
ViewMarq Industrial Marquees
Other HMI
Communications
Appendix Book 1
Terms and Conditions

Considerations for Choosing a Controller

Use the worksheet on the following pages as a checklist of the things to consider when determining programmable controller requirements. It lists the most important areas to consider when choosing a system, and provides space for recording determinations of your system needs.

Consideration	Information to Record		Why this is important
1. Proposed System	____ New system	_____ Existing system	Determine whether your system is new or existing: Will your system be installed from scratch or are there existing products already installed? The rest of your system will need to be compatible with new components. *Why this is important:* Certain controller products may not be compatible with others. Making sure your existing products are compatible with any new products you are researching will save you time and money. Check appropriate entry.
2. Environmental Issues	____ Codes/environmental issues to consider	____ No codes or environmental issues to consider	Consider any environmental issues that will affect your application (temperature, dust, vibration, codes specific to your facility, etc.). *Why this is important:* Certain environments may affect the operation of a controller. For example, typical controllers have an operating temperature of 0-55 degrees Celsius (32-130 degrees F). If your application will include any extreme environmental conditions, or you have specific codes at your facility that must be met, you will need to either research products that meet those specifications or design the installation to meet requirements. Check appropriate entry.
3. Discrete Devices	_____ Total inputs: _____ AC _____ DC	_____ Total outputs: _____ AC _____ DC	Determine how many discrete devices your system will have. Which types (AC, DC, etc.) are needed? *Why this is important:* The number and type of devices your system will include is directly linked to the amount of I/O that will be necessary for your system. You will need to choose a controller that supports your I/O count requirements and has modules that support your signal types.Enter quantities and type based on corresponding field devices.
4. Analog Devices	_____ Total inputs: ____ Voltage ____ Current ____ Thermo ____ RTD	_____ Total outputs: ____ Voltage ____ Current	Determine how many analog devices your system will have. Which types (voltage, current, temperature, etc.) are needed? *Why this is important:* The number and type of devices your system will include is directly linked to the amount of I/O that will be necessary for your system. You will need to choose a controller that supports your I/O count requirements and has modules that support your signal types. Enter quantities and type based on corresponding field devices.
5. Specialty Modules or Features (application-specific)	_____ High speed counter _____ Positioning _____ Servo/stepper _____ BASIC programming _____ Real-time clock _____ Others (list)		Determine whether your system will require any specialty features: Will your application require high-speed counting or positioning? What about a real-time clock or other specialty feature? *Why this is important:* Specialty functions are not necessarily available in a controller CPU or in standard I/O modules. Understanding the special functions your system may perform will help you determine whether or not you will need to purchase additional specialty modules. Check all features required.

Table continued on the following page

Considerations for Choosing a Controller

Consideration	Information to Record	Why this is important
6. CPU Required	Hardware requirements: ________ K program memory required (estimated) ________ K data memory required (estimated) ______ Fast scan time required? ______ Battery backup required? Software/special function requirements: ____ PID ____ Floating Point Math Others (see Programming section below)	Determine the type of CPU you will need: How much memory will your system require? How many devices will your system have (determines data memory)? How large is your program, and what types of instructions will your program include (determines program memory)? How fast a scan time do you need? *Why this is important:* Data memory refers to the amount of memory needed for dynamic data manipulation and storage in the system. For example, counter and timer instructions typically use data memory to store setpoints, current values, and other internal flags. If the application requires historical data retention, such as measured device values over a long period of time, the size of the data tables required may determine the CPU model you choose. Program memory is the amount of memory needed to store the sequence of program instructions that have been selected to perform the application. Each type of instruction requires a specific amount of program memory, typically defined in a programming manual. Applications that are basically sequential in nature can rely on the I/O device rule of thumb to estimate program memory (five words of memory for each I/O device); complex applications will be more difficult to judge. If scan time is important in your application, consider the CPU processor speed as well as instruction execution speed. Some CPUs are faster at boolean logic but slower with data handling instructions. If special functions such as PID are required, the CPU you select may make those functions easier to perform. For program memory required, follow this rule of thumb: 5 words of program memory for each discrete device and 25 words for each analog device. Check or calculate all requirements that apply.
7. I/O Locations	______ Local . . . only _______ Remote Locations Specific remote I/O protocol required? Which one? ____________________	Determine where your I/O will be located: Will your system require only local I/O, or both local and remote I/O locations? *Why this is important:* If subsystems will be needed at long distances from the CPU, you will need a controller that supports remote I/O. You will also have to determine if the remote distances and speeds supported will be adequate for your application. Serial and Ethernet-based I/O hardware are two typical choices available for most systems. This I/O may also be referred to as distributed I/O, and may require a particular protocol, such as Modbus. Enter number of physical locations needed, and if/what specific protocol may be required.
8. Commuications	_____ Ethernet ____ PLC to PLC ____ Modbus RTU ____ ASCII (interface to serial devices) ____ Other	Determine your communication requirements: Will your system be communicating to other networks, systems or field devices? *Why this is important:* Communication ports (other than the programming port) are not always included with a controller. Knowing your system communication requirements will help you choose a CPU that supports your communication requirements, or additional communication modules if necessary. Check any/all communications functions required.
9. Programming	____ Floating point math ____ Drum sequencer ____ PID loops ____ number of loops needed ____ Subroutines ____ Direct interrupts ____ Others (list)	Determine your programming requirements: Does your application require only traditional programming instructions, or are special instructions necessary? *Why this is important:* Certain controllers may not support every type of instruction. You will need to choose a model that supports all instructions that you may need for a specific application. For example, built-in PID functions are much easier to use than writing your own code to perform closed-loop process control. Typical instructions such as timers, counters, etc. are available in most controllers; note any other special instructions required here. Check any/all programming functions required.

Programmable Controller Summary

Those making the buying decisions for Programmable Controller applications can have very different needs. We offer a selection of controller families that can fit a variety of applications. Regardless if you are a newcomer to programmable controllers or if you are a seasoned veteran; whether you need simple discrete control or if you need to calculate complex algorithms lightning fast, we have a controller family that is perfect for you.

CLICK: Our best value PLC

- Easy for new user
- Basic machine control
- Lowest cost
- Simple analog
- FREE Software

The CLICK PLC is becoming one of the industry's favorite control systems in the 142 I/O or less category.

Basic CPUs:
- 8 DC In / 6 DC Out (sinking)
- 8 DC In / 6 DC Out (sourcing)
- 8 DC In / 6 Relay Out
- 8 AC In / 6 Relay Out

Standard CPUs:
- 8 DC In / 6 DC Out (sinking)
- 8 DC In / 6 DC Out (sourcing)
- 8 DC In / 6 Relay Out
- 8 AC In / 6 Relay Out

Analog CPUs:
- 4 DC In / 4 DC Out (sinking), 2 Analog In, 2 Analog Out (current/voltage selectable)
- 4 DC In / 4 DC Out (sourcing), 2 Analog In, 2 Analog Out (current/voltage selectable)
- 4 DC In / 4 Relay Out, 2 Analog In, 2 Analog Out (current/voltage selectable)

- Built-in communication ports (two in Basic CPUs, three in Standard and Analog CPUs)
- Optional battery backup (Standard and Analog CPUs only)
- Real time clock/calendar (Standard and Analog CPUs only)
- Removable terminal blocks for easy wiring
- Stackable discrete and analog I/O option modules (DIN-rail or panel mountable)
- Program AND documentation stored in CPU
- Decimal memory addressing
- 21 easy-to-use instructions
- 8,000 steps of program memory

Do-more H2 and T1H Series CPUs: Spend Less, Do More!

Do-more

- Fast CPU
- CPU w/ built-in Ethernet
- H2 works with most DL205 I/O modules. T1H works with most Terminator I/O.
- Advanced discrete
- Process control
- Expandability
- FREE Software

Do-more is a micro-modular PLC that leverages the I/O to create an incredibly powerful PLC at a fraction of the cost of comparable controllers.

- Cost effective hardware
- Documentation can be stored on board
- Built-in communications include USB programming, serial, and (optional) Ethernet
- Practical counting/pulse
- Ethernet-connected expansion I/O
- High-performance processors
- FREE programming software (with built-in simulator)
- Powerful control over program execution
- Enhanced troubleshooting tools
- FREE online training with coupon
- Starter kits available

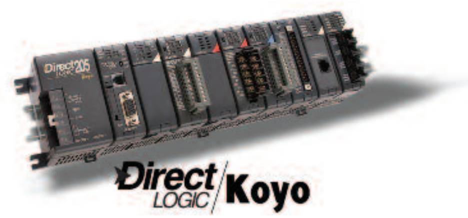
DirectLOGIC: Long-running PLCs

- Advanced discrete
- Basic process control
- Expandability
- Ethernet
- FREE Software*

*100-word program limitation, $395 for unlimited program sizes

DirectLOGIC PLCs (nano fixed I/O to modular units) are industry workhorses, time-tested in some of the toughest industrial settings.

All platforms use the same DirectSOFT programming software, so your investment is protected.



Six PLC platforms to choose from in the DirectLOGIC family:
- DL05 stand-alone brick PLC with one option slot (30 I/O max)
- DL06 stand-alone brick PLC with 4 option slots (100 I/O max)
- DL105 stand-alone brick PLC with high amp relays (18 I/O max)
- DL205 powerful modular PLC with the most available option modules (up to 16,384 I/O max)
- DL305 time tested, legacy control platform (up to 368 I/O max)
- DL405 time tested, legacy control platform (up to 16,384 I/O max)

Productivity3000: Premium features at a value price

- Advanced discrete and process
- Data collection
- Extensive communication
- Distributed I/O
- FREE Software

The Productivity3000 controller shatters the price per feature paradigm in every category, with prices that can't be beat and a two-year warranty on all modules.

- **Latest additions include high-speed input and output modules, and a lower-cost CPU and remote I/O slave module**

- Auto discovery of hardware, including remote I/O bases and GS drives when connected to the Ethernet remote I/O network
- Tag name database programming
- Task management
- Advanced "fill-in-the-blank" instructions
- Seamless corporate database connectivity
- Run-time editing and project transfer
- Project file, tag database and ladder documentation stored in the CPU
- FREE Productivity Suite software

Application Briefs

DL06 PLC puts heaters to the test

Pyromatics Automation Systems of Crystal Lake, Il. was contracted by a customer to develop a Life Cycle Test Station for its electric heating elements.

This test station needed a user-friendly graphical interface to give operators the ability to select multiple ramp/soak parameters, output voltages, temperature sensor types, amperage ratings and total cycle counts on tests for the cast-in electric heater platens. The system also needed to record temperature, volts, and current draw throughout the test for use in quality reports. Also, a failure of the heater required a safe shutdown of the test while alerting the quality department of the alarm condition.

Pyromatics selected the cost-effective DirectLOGIC® DL06 PLC as the heart of the system because of its ability to control up to eight PID loops and the multiple expansion slots available for thermocouple cards and analog input modules. It also controls two heaters, two chillers and an array of panel indicators, buttons, switches and relays.

A C-more 10-inch TFT touch-screen operator interface was used to provide operators with the necessary interface to operate and monitor the tests.

The completed system allows users to quickly connect the heater to be tested, enter test parameters, and run the test. Trend charts on the C-more panel track test parameters and quickly identify potential issues such as sudden drops in current or temperature.

Alarm reporting and history are also automatically recorded, allowing the operator to determine causes of failure. Data from the test can be easily uploaded to a USB thumb drive from the C-more panel. The data can then be imported into the user's choice of word processor or spreadsheet.

Semi cab sheeting production improved

ITS, a design build firm in Columbus, Ohio specializes in industrial automation. The company was contacted by a division of International Harvester responsible for the manufacturing of semi cabs. International Harvester uses automated machines to place aluminum rivets on sheeting that is attached to the frame of the semi cabs. The original CNC machines were becoming antiquated and needed to be upgraded.

ITS chose a DL205 PLC as the new controller for the machines, along with discrete I/O and an H2-CTRIO high-speed counter module that drives a dual axis servo. An H2-ECOM Ethernet Communications card links the machines back to an office for data acquisition. ITS also added a 15-inch touch screen for diagnostics.

In the new system, an operator stamps sheets of aluminum to welded framework with a handful of hand rivets and then places the product onto a dual axis servo table. After the operator selects one of five different parts programs, the machine will navigate the panel under the head assembly, which is responsible for the drilling and riveting, with a tolerance of 1/10 of a millimeter. The panel is drilled and a rivet is installed and squeezed to approximately 1200 PSI, producing a rivet consistency within .003 in. After completion of the panel (between 64 and 138 rivet locations), the machine will return to its home position and await the next product.

The solution increased productivity by approximately 30% and provides an easy way to run and maintain the machines.

Cost-effective I/O simplifies hydroelectric plant controls upgrade

Lockhart Power Company owns and operates a hydroelectric plant located on the Broad River in upstate South Carolina.

The plant includes an 8-gate dam feeding a canal that channels the water flow to the powerhouse. The powerhouse contains five turbine generators with a combined power capacity of over 17 MW. The dam and turbine control system receives data from power, flow, and level sensing devices to perform monitoring and control of the dam, generators, and associated equipment.

Lockhart Power contracted North Fork Electric in Crumpler, NC, to lend their expertise to a renovation of the conrtol system.

The system consists of seven DirectLOGIC DL205 micro-modular PLCs with built in PID functionality. Each of the five systems for generator control includes discrete and analog I/O, and an Ethernet communications module. The remaining two PLCs are configured in a master/slave arrangement and control the dam gates, located upriver from the powerhouse, via radio modems. Operator interfaces include two 6-inch color touch screen panels and a Windows NT-based PC running the LookoutDirect SCADA/HMI software package.

In the automatic mode, the PLC can start, stop, and operate the generator, and control startup and synchronization of the turbine. Changing the generator gate position varies the flow of water to the turbine.

The dam control system controls the eight canal gates located at the dam, which regulate the flow of water downstream to the turbines.

Programmable Controller Selection Guide

Selection Criteria		I/O Capacity				Basic Machine Control		Process Control		High-Speed and Motion	CPU Communications						
Controller Family	Series / CPU	Built-In I/O		Local I/O (with Expansion)	Total Possible I/O	Digital I/O & Simple Logic Requirements	Stage Programming (State Machine Control)	Analog I/O, Simple Math and data manipulation	16 PID loops, Complex Math formulas &/or Array Manipulation	High-speed I/O modules with Motion Control	Built-In Ethernet & USB	Built-In Local I/O Expansion Ports	Built-In Remote I/O	Built-In RS-232 Serial port	Built-In RS-422 Multi-drop port	Built-In RS-485 Multi-drop port	Modbus TCP Ethernet Protocol
		AC, DC, Relay I/O	Analog In & Out														
CLICK PLC	Basic CPU	8 In/6 Out		142	142	✔		✔						✔			
	Standard CPU	8 In/6 Out		142	142	✔		✔						✔		✔	
	Analog CPU	4 In/4 Out	2 In/2 Out	140	140	✔		✔						✔		✔	
Do-more	H2-DM1			256	65,536	✔	✔	✔	✔	✔				✔			
	H2-DM1E			256	65,536	✔	✔	✔	✔	✔	✔		✔⑥	✔			✔
	T1H-DM1			256	65,536	✔	✔	✔	✔	✔				✔			
	T1H-DM1E			256	65,536	✔	✔	✔	✔	✔	✔		✔⑥	✔			✔
Productivity	P3-550 CPU			3520	59,840	✔		✔	✔	✔	✔	✔	✔	✔		✔	✔
	P3-530 CPU			3520	3520	✔		✔	✔	✔	✔	✔		✔		✔	✔

① Scan time is based on type and amount of ladder logic instructions and total system I/O
② Scan times may vary during Run-Time Transfers
③ DeviceNet & ProfiBus Slave modules for the DL205 series are installed in place of the CPU in the CPU slot
④ High Speed inputs available on DC input models / Pulse output available on DC output models
⑤ RS-485 for Modbus protocol only
⑥ The Ethernet ports on the H2-DM1E and T1H-DM1E allow expansion beyond the local base.

Programmable Controller Selection Guide

Ports & Protocols							Communications and Specialty Modues					Programmability										
Modbus RTU Slave	Modbus RTU Master	K-Sequence Slave	DirectNet Slave	DirectNet Master	ASCII Out	ASCII IN	Ethernet (10/100 Mb)	Serial RS-232 & RS-485	Basic Coprocessor	Ethernet or Serial Remote I/O	DeviceNet/Profibus Slaves	Total Memory	Battery Backed Memory Clock / Calendar	Stage Programming	Run Mode Edits (Outputs pause during transfer)	Run-Time Transfer (Scan updates during transfer)	Built-in High Speed Counter & Pulse Output	Floating point Math	Freeform Expressions in Math	Drum Sequencer	Email Instruction	Sub-Divided Program Tasks
✔	✔				✔	✔						8k steps						✔	✔	✔		✔
✔	✔				✔	✔							✔					✔	✔	✔		✔
✔	✔				✔	✔							✔					✔	✔	✔		✔
✔	✔	✔			✔	✔	✔	✔		✔		262kb	✔	✔	✔	✔		✔	✔	✔	✔	✔
✔	✔	✔			✔	✔	✔	✔		✔		262kb	✔	✔	✔	✔		✔	✔	✔	✔	✔
✔	✔	✔			✔	✔	✔	✔		✔		262kb	✔	✔	✔	✔		✔	✔	✔	✔	✔
✔	✔	✔			✔	✔	✔	✔		✔		262kb	✔	✔	✔	✔		✔	✔	✔	✔	✔
✔	✔				✔	✔				✔		50Mb	✔			✔②		✔	✔	✔	✔	✔
✔	✔				✔	✔						25Mb	✔			✔②		✔	✔	✔	✔	✔

Programmable Controller Selection Guide

Selection Criteria			I/O Capacity				Basic Machine Control		Process Control		High-Speed and Motion	CPU Communications						
Controller Family	Series / CPU		Built-In I/O: AC, DC, Relay I/O	Built-In I/O: Analog In & Out	Local I/O (with Expansion)	Total Possible I/O	Digital I/O & Simple Logic Requirements	Stage Programming (State Machine Control)	Analog I/O, limited PID, Simple Math and data manipulation	>16 PID loops, Complex Math for-mulas &/or Array Manipulation	High-speed I/O modules with Motion Control	Built-In Ethernet & USB	Built-In Local I/O Expansion Ports	Built-In Remote I/O	Built-In RS-232 Serial port	Built-In RS-422 Multi-drop port	Built-In RS-485 Multi-drop port	Modbus TCP Ethernet Protocol
DirectLOGIC	DL05	All CPUs	8 In/6 Out		30	30	✔	✔	✔		✔				✔			
	DL06	All CPUs	20 In/ 16 Out		100	100	✔	✔	✔		✔				✔	✔	✔	
	DL105	All CPUs	10 In/ 8 Out		n/a	18	✔	✔							✔			
	DL205	D2-230			256	256	✔	✔							✔			
		D2-240			256	896	✔	✔			✔				✔			
		D2-250-1			768	16,384*	✔	✔	✔		✔			✔	✔	✔		
		D2-260			1280	16,384*	✔	✔	✔		✔			✔	✔	✔	✔⑤	
	DL305	D3-330			176	176	✔						✔					
		D3-340			184	184	✔						✔		✔			
		D3-350			368	368	✔	✔	✔				✔	✔	✔	✔		
	DL405	D4-430			640	1664	✔	✔					✔		✔			
		D4-440			640	2688	✔	✔					✔		✔	✔		
		D4-450			2048	16,384*	✔	✔	✔		✔		✔	✔	✔	✔		

* 16384 (fully expanded H4-EBC slave bases, using V-memory & bit of word instructions)

① Scan time is based on type and amount of ladder logic instructions and total system I/O

② Scan times may vary during Run-Time Transfers

③ DeviceNet & ProfiBus Slave modules for the DL205 series are installed in place of the CPU in the CPU slot

④ High Speed inputs available on DC input models / Pulse output available on DC output models

⑤ RS-485 for Modbus protocol only

⑥ The Ethernet ports on the H2-DM1E and T1H-DM1E allow expansion beyond the local base.

Programmable Controller Selection Guide

Ports & Protocols							Communications and Specialty Modues					Programmability										
Modbus RTU Slave	Modbus RTU Master	K-Sequence Slave	DirectNet Slave	DirectNet Master	ASCII Out	ASCII IN	Ethernet (10/100 Mb)	Serial RS-232 & RS-485	Basic Coprocessor	Ethernet or Serial Remote I/O	DeviceNet/Profibus Slaves	Total Memory	Battery Backed Memory Clock / Calendar	Stage Programming	Run Mode Edits (Outputs pause during transfer)	Run-Time Transfer (Scan updates during transfer)	Built-in High Speed Counter & Pulse Output	Floating point Math	Freeform Expressions in Math	Drum Sequencer	Email Instruction	Sub-Divided Program Tasks
✔	✔	✔	✔	✔	✔		✔	✔	✔		✔	6.0k		✔	✔		✔④		✔	✔	✔	
✔	✔	✔	✔	✔	✔	✔	✔	✔	✔		✔	14.8k	✔	✔	✔		✔④	✔	✔	✔	✔	
		✔										2.4k		✔	✔		✔④			✔		
		✔									✔③	2.4k		✔	✔							
		✔	✔				✔	✔	✔	✔	✔③	3.8k	✔	✔	✔							
✔	✔	✔	✔	✔	✔		✔	✔	✔	✔	✔③	14.8k	✔	✔	✔			✔	✔	✔	✔	
✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔③	30.4k	✔	✔	✔			✔	✔	✔	✔	
			✔	✔								3.8k										
✔			✔									3.9k										
✔	✔	✔	✔	✔	✔			✔				14.8k	✔	✔	✔			✔		✔		
		✔	✔				✔	✔	✔	✔		6.5k	✔	✔	✔							
		✔	✔				✔	✔	✔	✔		22.5k	✔	✔	✔					✔		
✔	✔	✔	✔	✔	✔		✔	✔	✔	✔		30.8k	✔	✔	✔			✔	✔	✔	✔	