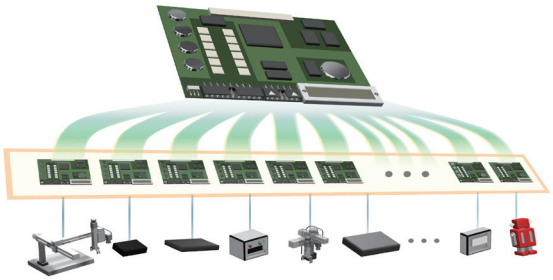
# Chapter 1: Introductory Outline

## 1-1 Hardware

The MPC-2000 series are board controllers. Each unit is a PCB board, and a system is constructed by inserting the boards into a rack (2~16 slots).



### CPU

As the CPU, Renesas SH2-70835 (Internal clock speed 80 MHz) is adopted to realize 4~5 times the processing power of H8/300 in spite of its small size. By this the execution speed of the loaded interpreter is improved by more than twice in comparison with comparable products of our company, and because the majority of the system memory is built in the CPU, reliability has been enhanced.

* SH2-70835 has a 512 K byte flash ROM and a 32 K byte RAM built on the CPU chip and accesses them with no wait and a 32 bit bandwidth. By this high-speed execution beyond the clock speed is made possible.
* MPC-2200 has a built-in SH2A (SH-7211)

### Peripheral boards

As peripheral boards, an I/O board, a pulse generation board, an RS-232C expansion board (compatible with RS-422/485), a CUnet (real-time LAN for FA), and A/D-D/A boards are prepared. For each of these, commands and functions are prepared so that they are described uniformly in one language.

## 1-2 Information Exchange

In present-day device control, information exchange with PCs is indispensable. In the MPC-2000 series, while serial communication capability such as RS-232C is provided as the default, it is also compatible with USB memory and CUnet.

### USB memory

Via a USB port provided on MPC-1000 and MPC-2000 direct data exchange is possible for text data only.
Device operation logs and measured values can be easily stored.
Saving/loading programs and point data are also supported with commands, facilitating migration to another model and maintenance.

### CUnet

Real-time memory sharing is supported on the network in CUnet. The USB-CUnet is prepared for PCs, which allows direct data storing/retrieval from a PC.
Because dedicated device driver and library DLLs are prepared for the USB-CUnet, PC programs can be created by the user using VB, VC, and the like.

* CUnet is a trademark of StepTechnica Co., Ltd. and is a real-time network for FA developed by the same company.

## 1-3 Programs

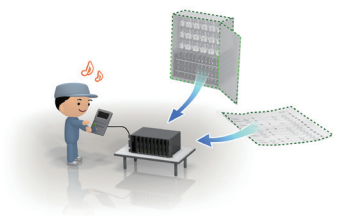
The interpreter of the MPC-2000 series (*Hereafter called MPC-2000) is a compact, BASIC-like, integer interpreter which functions on a one-chip CPU. Nevertheless, it provides 32 multitasks, a protocol for a touch panel, and RS-232C/RS-485 communication functions as commands. In addition, it can also handle floating-point arithmetic operations. Therefore, in spite of being a board controller, it has realized as wide a coverage as a PLC and as much flexibility as a PC. The MPC-2000 built-in language is named BL/1 after the computer language PL/1 developed by IBM for the first time.

A development tool FTMW dedicated for programming can be downloaded free from our company website and can be used.

### Advantages of BL/1

Before explaining the advantages of BL/1, a comparison with the industry standard PLC needs to be made.

PLC (sequencer) is a computer simulating device control with relay circuits which were the cutting-edge technology until around 1970. It first appeared as a product in the marketplace around 1980 when microcomputers were becoming popular, and was designed so that relay circuits could be easily described. Therefore, a control board (a collection of relays) accompanying each device was replaced with an existent PLC. This PLC culture developed mainly in the automobile industry dealing with metal processing. In fact, it is believed that the first PLC in the world was first developed and used by GM.

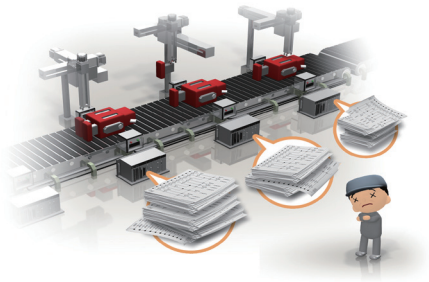


Through this history, LADDER language became the standard control language in the world of device control. The popularization of the PLC dramatically enhanced production efficiency in the factory.
The reason is that relays in the relay circuit control had a limited lifetime, and required periodic replacement. In addition, because changing a control sequence required rewiring which is a huge undertaking, factory needs could not be flexibly met.

Because a PLC using a microcomputer and an electronic circuit simultaneously solved these problems, it obtained market support and established the common sense that PLC is synonymous with device control. In the Japanese industrial community, PLC became synonymous with control, and afterwards in the electric and electronic industry which came to require automation, control with PLC was applied as it was.

PLC and its language which contributed to the production efficiency in Japanese factories in this way had the following shortcomings.

- Branch processing cannot be performed. (Unsuitable for grade classification such as good and defective products)
- Numerical values cannot be handled. (Unsuitable for numerical control)
- Programs cannot be modularized. (Reuse of a program is nearly impossible.)
- Exclusion of parasitic logic is difficult. (Bugs accompanied with malfunctions are essentially unavoidable.)



These shortcomings are caused by the fact that PLC is accomplished by relay circuits which are its origin and can be said to comprise its very characteristics. Recent PLCs are equipped with functions which compensate for these shortcomings, providing pseudo-solutions.

For example, as a measure against the inability to perform numerical control, there is a method wherein a separate unit called a positioner is installed and operated with a separate dedicated language.

This kind of work-around method is regularly practiced with a sequencer, which often creates a situation wherein rather than a program for the sequencer itself, a unit such as a positioner operated with a separate language carries a more important role.

BL/1 does not have such a shortcoming. It is a language which deals with numerical processing and can naturally describe conditional branching. Programs can be modularized in various ways, facilitating their reuse. In addition, because of its language structure, it can never perform unexpected behavior due to unexpected input as sequencer programs do.

## Advantages of MPC-2000

In the same period as when PLC was developed, proprietary control technology was developed in the semiconductor industry.

In the early semiconductor industry, laboratories and actual scenes were intermingled, and device control centering on minicomputers often used by researchers became the standard. Although control was full-fledged, using a high-level language, its cost was also high, and program development required high-level, professional knowledge. However, because the semiconductor industry at that time was a high value-added industry, the high cost for such facility was not a serious issue. Although PCs became inexpensive and came to be used often in place of minicomputers, the OS of PCs had the shortcoming of freezing from the beginning.

The cause of freezing was that PC hardware was a consumer product and thus its reliability was not given much importance, and so-called object-oriented programs which waste memory came to be used.

Although recent PCs also commonly have 1 G byte of built-in memory, if just 1 byte is unstable, the entire operation is affected.

Therefore, although a computer having 1 G byte memory built-in would require 100 thousand times the reliability of a minicomputer having 100 K byte memory to achieve the same level of reliability, the facts suggest otherwise. In addition, the programming languages themselves have become too large, decreasing reliability. Control utilizing a PC has the following shortcomings.

- Powering on takes time.
- Certain reliability cannot be secured.
  (Freezing is normal, and it is not clear whether it is due to bugs or natural death.)
- Exchange recovery also requires skill and time. (An equivalent product may not always be obtained.)
- Programming takes time to learn (even if the theme is simple).

The MPC-2000 (BL/1) has overcome these shortcomings. In spite of being a language built-in board, it boots up instantly. For its reliability, all possible measures such as using a one-chip microcomputer have been made for industrial use.
Even if it breaks down, board exchange and software installation are easy. In addition, programming can be learned within half a day if the theme is simple.