SYSMAC
C200H-FZ001
Fuzzy Logic Unit

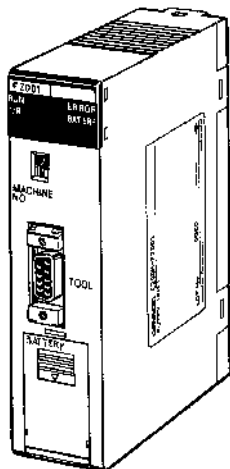# OPERATION MANUAL

OMRON

C200H-FZ001 Fuzzy Logic Unit

Operation Manual

Revised September 1999

Notice:

OMRON products are manufactured for use according to proper procedures by a qualified operator and only for the purposes described in this manual.

The following conventions are used to indicate and classify warnings in this manual. Always heed the information provided with them.

⚠ **DANGER**   Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

⚠ **WARNING**   Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

⚠ **Caution**   Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.

OMRON Product References

All OMRON products are capitalized in this manual. The word "Unit" is also capitalized when it refers to an OMRON product, regardless of whether or not it appears in the proper name of the product.

The abbreviation "Ch," which appears in some displays and on some OMRON products, often means "word" and is abbreviated "Wd" in documentation in this sense.

The abbreviation "PC" means Programmable Controller and is not used as an abbreviation for anything else.

Visual Aids

The following headings appear in the left column of the manual to help you locate different types of information.

Note Indicates information of particular interest for efficient and convenient operation of the product.

1, 2, 3... 1. Indicates lists of one sort or another, such as procedures, checklists, etc.

About this Manual:

This manual describes the operation of the C200H-FZ001 Fuzzy Logic Unit and includes the sections described below. The C200H-FZ001 is used in conjunction with Fuzzy Support Software. Refer to the *Fuzzy Support Software Operation Manual* for details on the use of the Fuzzy Support Software.

Please read this manual completely and be sure you understand the information provide before attempting to operate the C200H-FZ001.

Section 1 Introduction introduces the features of the Fuzzy Logic Unit and explains the fuzzy logic processing that takes place during operation.

Section 2 Components and Switch Settings describes the components of the C200H-FZ001 and their functions.

Section 3 Memory Allocation explains how words in the C200H PC are allocated to the C200H-FZ001 and their functions as flags, control bits, and I/O words.

Section 4 Program and Knowledge Base explains briefly how to prepare the sub-program used in the C200H and the knowledge base used in the C200H-FZ001.

Section 5 Example Application describes the steps in the development of an actual control system that uses a C200H-FZ001.

Section 6 Maintenance and Troubleshooting provides guidelines for routine maintenance and dealing with errors that might occur while using the C200H-FZ001.

⚠️ **WARNING**   Failure to read and understand the information provided in this manual may result in personal injury or death, damage to the product, or product failure. Please read each section in its entirety and be sure you understand the information provided in the section and related sections before attempting any of the procedures or operations given.

TABLE OF CONTENTS

PRECAUTIONS

This section provides general precautions for using the C200H-FZ001 Fuzzy Logic Unit and related devices.

The information contained in this section is important for the safe and reliable application of the C200H-FZ001 Fuzzy Logic Unit. You must read this section and understand the information contained before attempting to set up or operate the C200H-FZ001 Fuzzy Logic Unit.

1   Intended Audience

This manual is intended for the following personnel, who must also have knowledge of electrical systems (an electrical engineer or the equivalent).

- Personnel in charge of installing FA systems.
- Personnel in charge of designing FA systems.
- Personnel in charge of managing FA systems and facilities.

2   General Precautions

The user must operate the product according to the performance specifications described in the relevant manuals.

Before using the product under conditions which are not described in the manual or applying the product to nuclear control systems, railroad systems, aviation systems, vehicles, combustion systems, medical equipment, amusement machines, safety equipment, and other systems, machines, and equipment that may have a serious influence on lives and property if used improperly, consult your OMRON representative.

Make sure that the ratings and performance characteristics of the product are sufficient for the systems, machines, and equipment, and be sure to provide the systems, machines, and equipment with double safety mechanisms.

This manual provides information for programming and operating the Unit. Be sure to read this manual before attempting to use the Unit and keep this manual close at hand for reference during operation.

⚠ **WARNING**   It is extremely important that a PC and all PC Units be used for the specified purpose and under the specified conditions, especially in applications that can directly or indirectly affect human life. You must consult with your OMRON representative before applying a PC system to the above-mentioned applications.

3   Safety Precautions

⚠ **WARNING**   Do not attempt to take any Unit apart while the power is being supplied. Doing so may result in electric shock.

⚠ **WARNING**   Do not touch any of the terminals or terminal blocks while the power is being supplied. Doing so may result in electric shock.

⚠ **WARNING**   Do not attempt to disassemble, repair, or modify any Units. Any attempt to do so may result in malfunction, fire, or electric shock.

⚠ **WARNING**   Provide safety measures in external circuits (i.e., not in the Programmable Controller), including the following items, in order to ensure safety in the system if an abnormality occurs due to malfunction of the PC or another external factor affecting the PC operation. Not doing so may result in serious accidents.

- Emergency stop circuits, interlock circuits, limit circuits, and similar safety measures must be provided in external control circuits.
- The PC will turn OFF all outputs when its self-diagnosis function detects any error or when a severe failure alarm (FALS) instruction is executed. As a countermeasure for such errors, external safety measures must be provided to ensure safety in the system.

• The PC outputs may remain ON or OFF due to deposition or burning of the output relays or destruction of the output transistors. As a countermeasure for such problems, external safety measures must be provided to ensure safety in the system.

• When the 24-VDC output (service power supply to the PC) is overloaded or short-circuited, the voltage may drop and result in the outputs being turned OFF. As a countermeasure for such problems, external safety measures must be provided to ensure safety in the system.

4 Operating Environment Precautions

⚠ **Caution** Do not operate the control system in the following locations:

• Locations subject to direct sunlight.

• Locations subject to temperatures or humidity outside the range specified in the specifications.

• Locations subject to condensation as the result of severe changes in temperature.

• Locations subject to corrosive or flammable gases.

• Locations subject to dust (especially iron dust) or salts.

• Locations subject to exposure to water, oil, or chemicals.

• Locations subject to shock or vibration.

⚠ **Caution** Take appropriate and sufficient countermeasures when installing systems in the following locations:

• Locations subject to static electricity or other forms of noise.

• Locations subject to strong electromagnetic fields.

• Locations subject to possible exposure to radioactivity.

• Locations close to power supplies.

⚠ **Caution** The operating environment of the PC system can have a large effect on the longevity and reliability of the system. Improper operating environments can lead to malfunction, failure, and other unforeseeable problems with the PC system. Be sure that the operating environment is within the specified conditions at installation and remains within the specified conditions during the life of the system.

5 Application Precautions

Observe the following precautions when using the PC system.

⚠ **WARNING** Always heed these precautions. Failure to abide by the following precautions could lead to serious or possibly fatal injury.

• Always ground the system to 100 Ω or less when installing the Units. Not connecting to a ground of 100 Ω or less may result in electric shock.

• Always turn OFF the power supply to the PC before attempting any of the following. Not turning OFF the power supply may result in malfunction or electric shock.

• Mounting or dismounting I/O Units, CPU Units, Memory Units, or any other Units.

• Assembling the Units.

• Setting DIP switches or rotary switches.

• Connecting cables or wiring the system.

• Connecting or disconnecting the connectors.

⚠ **Caution** Failure to abide by the following precautions could lead to faulty operation of the PC or the system, or could damage the PC or PC Units. Always heed these precautions.

- Fail-safe measures must be taken by the customer to ensure safety in the event of incorrect, missing, or abnormal signals caused by broken signal lines, momentary power interruptions, or other causes.
- Always use the power supply voltages specified in this manual. An incorrect voltage may result in malfunction or burning.
- Take appropriate measures to ensure that the specified power with the rated voltage and frequency is supplied. Be particularly careful in places where the power supply is unstable. An incorrect power supply may result in malfunction.
- Install external breakers and take other safety measures against short-circuiting in external wiring. Insufficient safety measures against short-circuiting may result in burning.
- Do not apply voltages to the Input Units in excess of the rated input voltage. Excess voltages may result in burning.
- Do not apply voltages or connect loads to the Output Units in excess of the maximum switching capacity. Excess voltage or loads may result in burning.
- Disconnect the functional ground terminal when performing withstand voltage tests. Not disconnecting the functional ground terminal may result in burning.
- Be sure that all the mounting screws, terminal screws, and cable connector screws are tightened to the torque specified in this manual. Incorrect tightening torque may result in malfunction.
- Leave the label attached to the Unit when wiring. Removing the label may result in malfunction if foreign matter enters the Unit.
- Remove the label after the completion of wiring to ensure proper heat dissipation. Leaving the label attached may result in malfunction.
- Double-check all wiring and switch settings before turning ON the power supply. Incorrect wiring may result in burning.
- Wire correctly. Incorrect wiring may result in burning.
- Mount Units only after checking terminal blocks and connectors completely.
- Be sure that the terminal blocks, Memory Units, expansion cables, and other items with locking devices are properly locked into place. Improper locking may result in malfunction.
- Check the user program for proper execution before actually running it on the Unit. Not checking the program may result in an unexpected operation.
- Confirm that no adverse effect will occur in the system before attempting any of the following. Not doing so may result in an unexpected operation.
    - Changing the operating mode of the PC.
    - Force-setting/force-resetting any bit in memory.
    - Changing the present value of any word or any set value in memory.
- Resume operation only after transferring to the new CPU Unit the contents of the DM Area, HR Area, and other data required for resuming operation. Not doing so may result in an unexpected operation.
- Do not pull on the cables or bend the cables beyond their natural limit. Doing either of these may break the cables.
- Do not place objects on top of the cables or other wiring lines. Doing so may break the cables.
- Use crimp terminals for wiring. Do not connect bare stranded wires directly to terminals. Connection of bare stranded wires may result in burning.
- When replacing parts, be sure to confirm that the rating of a new part is correct. Not doing so may result in malfunction or burning.

- Before touching a Unit, be sure to first touch a grounded metallic object in order to discharge any static build-up. Not doing so may result in malfunction or damage.

SECTION 1
Introduction

This section introduces the features of the Fuzzy Logic Unit and explains the fuzzy logic processing that takes place during operation.

1-1 Features

The C200H-FZ001 Fuzzy Logic Controller brings advanced, high-speed fuzzy logic technology to the C200H PC. Control of some complicated systems that previously had to be supervised by a specialist can now be handled by the C200H.

Large I/O Capacity   The C200H-FZ001 provides 8 inputs and 4 outputs, and each rule can have up to 8 condition and 2 conclusion parts, allowing it to be used in a wide range of applications.

Wide Variety of I/O Units   The user can select from the wide variety of I/O Units and Special I/O Units that can be mounted on a C200H Backplane.

System Compatibility   The C200H-FZ001 Fuzzy Logic Controller is treated like other Special I/O Units and can be used with simple instructions like MOV(21), so a fuzzy control system can be created if there is space for just one Special I/O Unit in the existing control system.

IBM-compatible Software   The Fuzzy Support Software (FSS), used to create and transfer the knowledge base as well as to monitor operation, runs on an IBM PC/AT or compatible computer. All of the computer operations can be controlled with pull-down menus, making the software easy to use, even for beginners.

1-2 Configuration

In the figure below, the C200H-FZ001 is mounted in a CPU Rack, but it can be mounted in an Expansion I/O Rack or Slave Rack like other Special I/O Units. A total of 10 Special I/O Units can be controlled by a single C200H CPU and up to 4 Special I/O Units can be mounted on a Slave Rack.

C500-SU981-E
Fuzzy Support Software

IBM PC/AT
or compatible

Mouse

C200H-FZ001
Fuzzy Logic Unit

SYSMAC
C200H
C200HS
C200HX
C200HG
C200HE

C200H-AD001
Analog Input Unit

C200H-DA001
Analog Output Unit

RS-232C connecting cable
(9 pin to 9 pin or 9 pin to 25 pin)
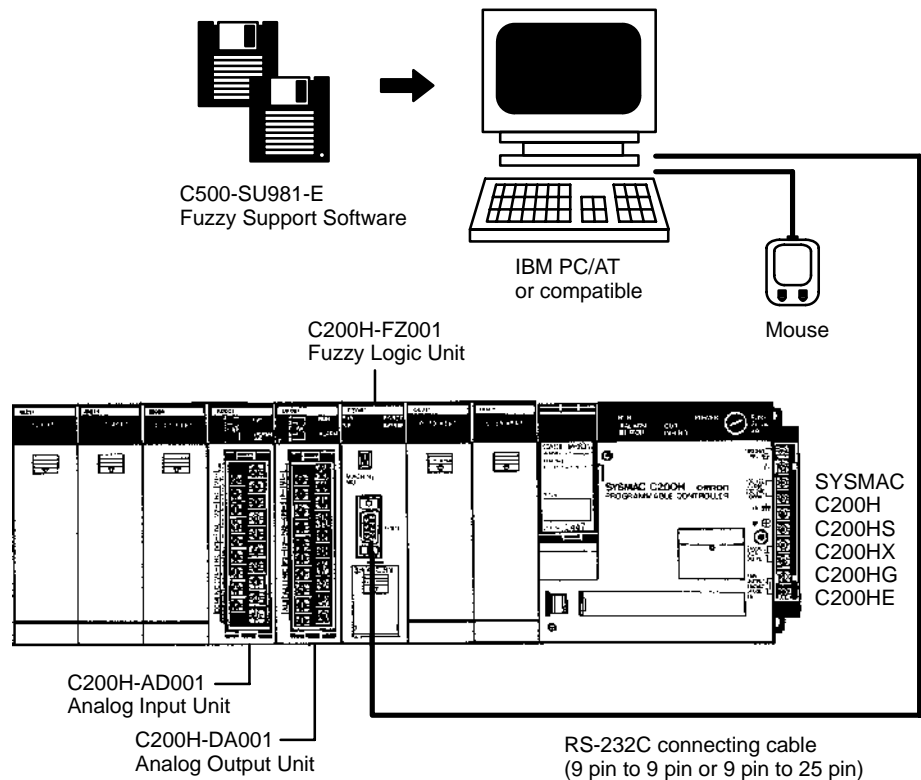
Note   Do not mount the C200H-FZ001 in the 2 slots next to the CPU on the CPU Rack.

1-3   Fuzzy Logic Theory

1-3-1 Introduction to Fuzzy Logic

Before fuzzy logic theory, modern control methods required a precise expression (either theoretical or numerical) of the relationship between inputs and outputs in a system. Instead of a precise mathematical formula, fuzzy logic expresses the relationship between inputs and outputs in a group of if/then statements based on human know-how and experience. In fuzzy logic theory the if/then statements are called **rules**. The "if" part of the rules describes possible system conditions, and the "then" part of the statements describes the appropriate action(s) for those conditions.

Creating a fuzzy control system with if/then statements is much, much simpler than expressing a control process using a set of mathematical equations, but the key to fuzzy logic theory isn't the use of rules, it's the assignment of a **grade** to those statements according to how well they fit the current system conditions. All of the rules are combined with their relative importance determined by their grade. Assigning grades to the statements allows computers to deal easily with the variations in the real world.

Let's consider the concept of "warmth" to demonstrate how fuzzy theory assigns grades to conditions. "Warm" is an inexact, or fuzzy, concept. Most people will agree that 75° is warm, but wouldn't say that a temperature of 70° or 80° isn't warm. The feeling of "warm," as opposed to "hot" or "cold," just isn't as strong at 70° or 80°.

Fuzzy theory assigns a grade to this feeling of warmth, and the function that relates the condition (feeling of warmth) and the grade is called the **membership function**. A set of values defined by a membership function is called a **fuzzy set**. The most accurate membership function is usually a bell curve, but triangular or trapezoidal shapes are used most often because they are easy to define and use. The following diagram shows a triangular membership function for "warmth."
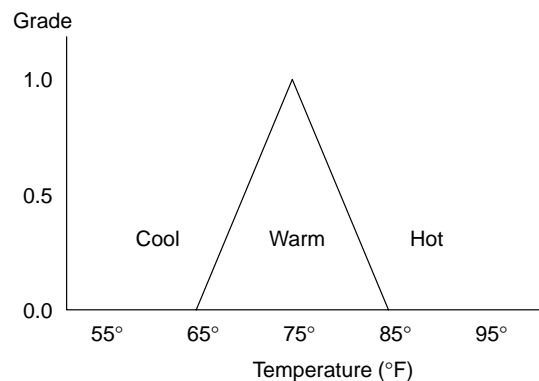
From this function, the computer can calculate the level of "warmth" for any temperature. A temperature of 75° has a grade of 1, while 70° and 80° have a grade of 0.5. In other words, a rule that uses "warmth" in its if statement (If it's warm, then ...) will be weighted by a grade of 1 at 75°, or 0.5 at 70° or 80°.

The fuzzy logic processor calculates the grade for every rule according to the current conditions and then combines these results to yield a single result. The final result is affected by all of the rules that the programmer included.

Making If/Then Statements The first step in creating a fuzzy control system is expressing the control process in if/then statements. The following statements express the know-how that people use unconsciously when they adjust the temperature of their bath.

> If the water is very hot, turn the faucet far to the left.

3

If the water is hot, turn the faucet a little to the left.

If the water is warm, don't turn the faucet.

If the water is cool, turn the faucet a little to the right.

If the water is cold, turn the faucet far to the right.

A pattern emerges in these statements that shows about how much to adjust the faucet depending on the temperature of the water. The know-how that one uses when he adjusts the bath temperature has been converted to a form that can be entered into a computer.

Making the Rules

The "if" clauses of a rule are called its **condition parts** and the "then" clauses are called its **conclusion parts**.

```
┌─────────────────┐              ┌──────────────────────────┐
│ Condition       │   ───────▶   │ Action                   │
│ (Very hot water)│              │ (Turn faucet far to the  │
│                 │              │  left)                   │
└─────────────────┘              └──────────────────────────┘
         ▲                                    ▲
   IF (condition) part              THEN (conclusion) part
```
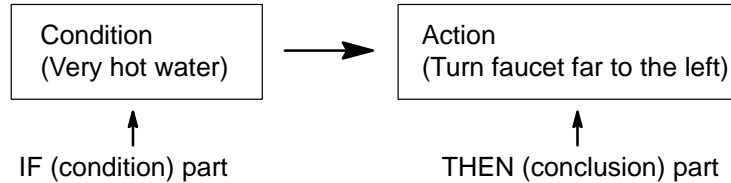
The C200H-FZ001 can have up to 128 rules with 8 condition and 2 conclusion parts each. More condition and conclusion parts allow the fuzzy control to handle more complicated control processes. For the bath example, a rule might have two condition parts such as "If the water is very hot and the air is cold," and two conclusion parts such as "turn the faucet far to the left and add a little water." The rules can be listed as shown below.

```
Condition parts:              Conclusion parts:

Aa1, Aa2, ...     ──────▶        Ca1, Ca2
Ab1, Ab2, ...     ──────▶        Cb1, Cb2
Ac1, Ac2, ...     ──────▶        Cc1, Cc2
         ┊                           ┊
         ┊                           ┊
```
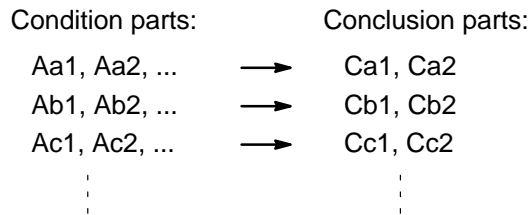
Condition parts within one rule are linked by logical ANDs, i.e., all of the parts must be true. Rules are linked by logical ORs, i.e., any rule can be satisfied regardless of the status of other rules. These relationships are illustrated below.

IF (Aa1 AND Aa2 AND ...) THEN (Ca1 AND Ca2)

OR

IF (Ab1 AND Ab2 AND ...) THEN (Cb1 AND Cb2)

OR

etc.
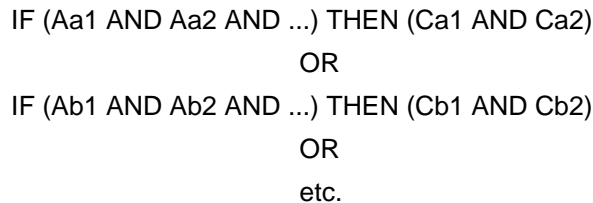
Membership Functions

The condition membership function assigns a numerical value to how well a specific value of a **fuzzy variable** (temperature, distance, speed, etc.) satisfies a condition (cool, warm, close, far, slow, fast, etc.) of a rule.

Condition membership functions can have four different shapes: S, Z, Λ, or Π. Refer to *1-3-3 Developing the Knowledge Base* for details on membership function shapes.

The C200H-FZ001 can have up to 7 membership functions for each fuzzy variable. The conditions (cool, warm, close, far, slow, fast, etc.) defined by the mem-

bership functions are called **labels**. The following diagram shows a standard arrangement of Λ-type membership functions.

Labels:
NL: Negative Large
NM: Negative Medium
NS: Negative Small
ZR: Zero Range
PS: Positive Small
PM: Positive Medium
PL: Positive Large
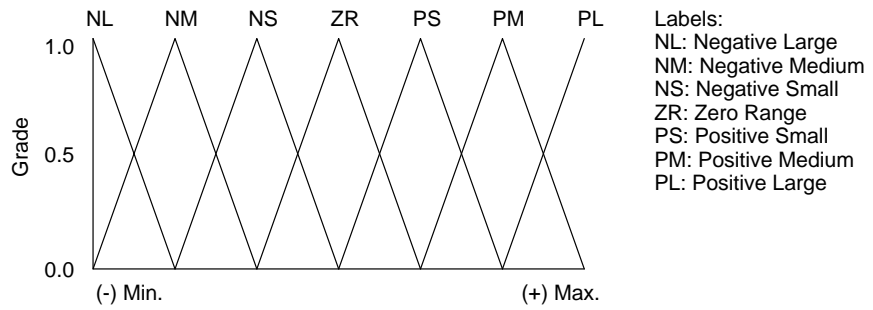
When creating membership functions for a particular system, modify this standard arrangement to suit the particular application. Refer to *1-3-3 Developing the Knowledge Base* for details on creating condition membership functions.

The conclusion membership function assigns a specific value to an action (turning a valve, raising voltage, etc.) indicated in the conclusion part of a rule. The shape of the conclusion membership function has almost no effect on the results, so the conclusion membership functions are approximated by vertical lines in the C200H-FZ001, as shown in the following diagram.

Labels:
NL: Negative Large
NM: Negative Medium
NS: Negative Small
ZR: Zero Range
PS: Positive Small
PM: Positive Medium
PL: Positive Large
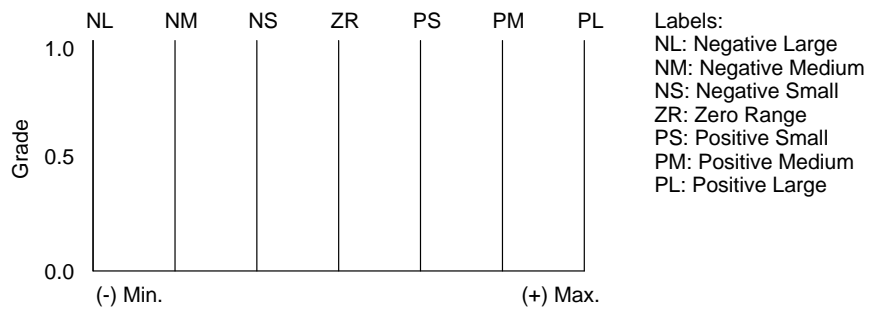
Fuzzy Logic Processing

This section briefly explains how the fuzzy logic processor calculates a result from the condition and conclusion parts of the rules that were expressed in terms of membership functions above. Refer to *1-3-2 Fuzzy Logic in the C200H-FZ001* for a more detailed explanation.

First of all, the grades for the condition parts of each rule are calculated from the inputs and condition membership functions for each rule. The minimum grade of the condition parts is the **rule grade**. The minimum value is taken because condition parts of a rule are linked by logical ANDs, so all of the parts must be satisfied simultaneously.

For a rule with two condition parts such as "If the water is very hot and the air is cold," the rule grade is the lesser of the two grades. In the case shown below, the rule grade is 0.7.
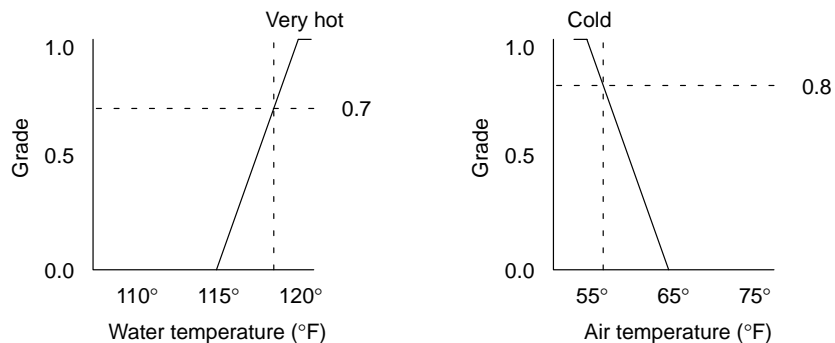
5

After all of the rule grades are calculated, the grade for each label is calculated. The grade for a label is called its **fuzzy output**. The fuzzy output is the maximum rule grade for that label. The maximum value is taken because the rules are linked by logical ORs. Taking the maximum value ensures that the results from all of the rules are taken into account fairly. The final result is calculated from the fuzzy outputs in an operation called defuzzification.

Defuzzification

There are two methods of defuzzification available in the C200H-FZ001, the center of gravity method and the maximum value method. The center of gravity method determines the center of gravity using the fuzzy outputs as the weight and the location of the label as the location. With this method, the location of the center of gravity is the final result.

The maximum value method uses the location of the label with the maximum fuzzy output as the final result. If two or more fuzzy outputs are equal to the maximum, either the leftmost (minimum) or rightmost (maximum) label location will be selected. The programmer determines in advance whether the leftmost or rightmost label location will be selected.

1-3-2 Fuzzy Logic in the C200H-FZ001

In this section, the fuzzy logic processing that was outlined in the previous section is examined in detail. Fuzzy logic processing in the C200H-FZ001 is divided into three steps: condition part processing, conclusion part processing, and defuzzification.

Calculating Rule Grades

Condition part processing involves calculating the rule grades. This is a two step process because one rule can have up to 8 condition parts. First, the grade for each condition part of a rule is calculated from its condition membership function and the inputs, then the minimum grade is taken as the rule grade.

The rule in the following example has 3 condition parts.
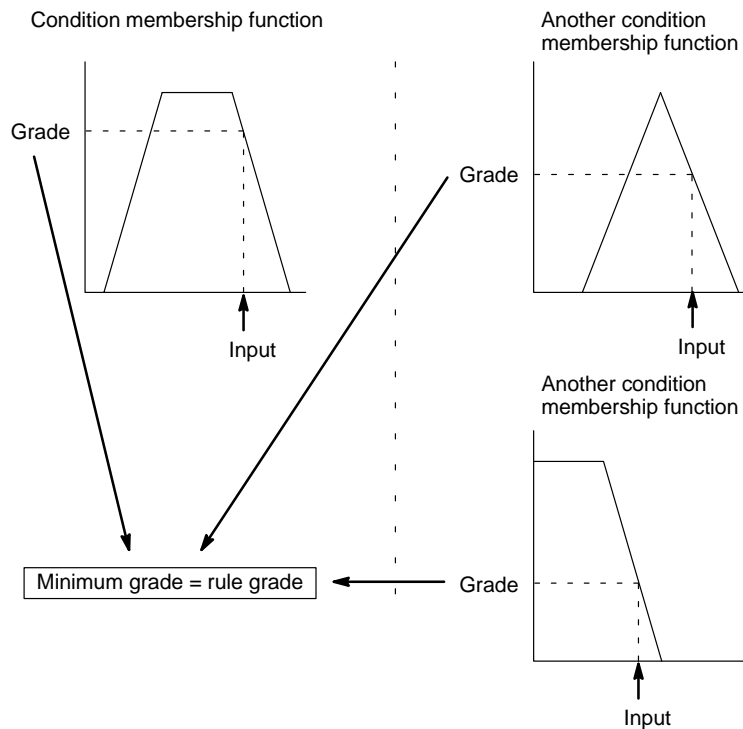
The rule grades, which are used in conclusion part processing, are stored in RAM and can be monitored with the FSS (Fuzzy Support Software).

Condition Functions

A membership function can be defined for each label of each input. Since there can be up to 7 labels for each of 8 inputs, a total of 56 condition membership functions can be defined.

Membership functions are made up of connecting line segments and are defined by the end points of the line segments. Up to four end points (defining three line segments) can be entered. The grade at the end points must be between 0 and 1 (between 000 and FFF) and an end point must have a grade of 0 or 1 (000 or FFF) at the edges of the input range.

The following diagram shows examples of <u>correct</u> membership functions.
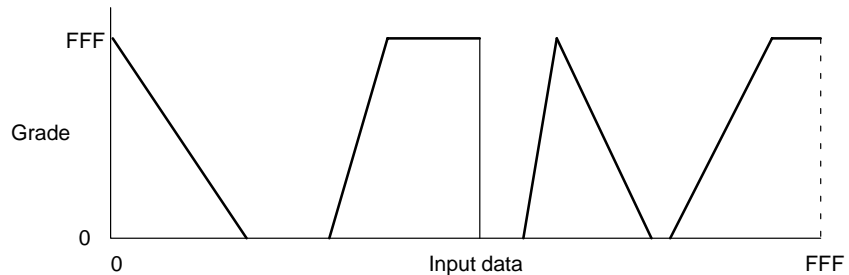
The following diagram shows examples of <u>incorrect</u> membership functions.
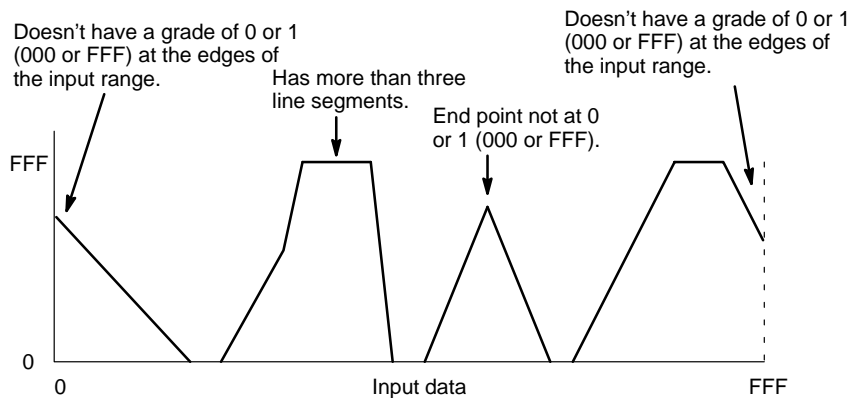
Conclusion Part Processing   Conclusion part processing involves calculating the fuzzy output for each label. The rule grades for all rules that have that label in their conclusion part are compared and the maximum is taken as the fuzzy output for that label. In the C200H-FZ001, the fuzzy output is equal to the maximum rule grade for that label because the conclusion membership function is simply a vertical line.

In the following example rules 0, 3, and 4 affect label A.

Rules with label A in the conclusion part:
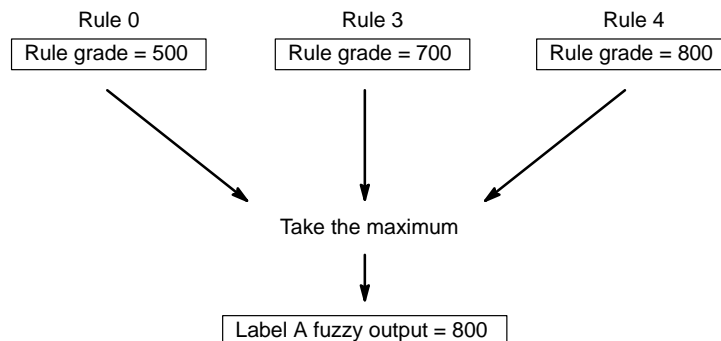
When the operation shown in the previous diagram is performed for all of the labels, each label will have a fuzzy output.
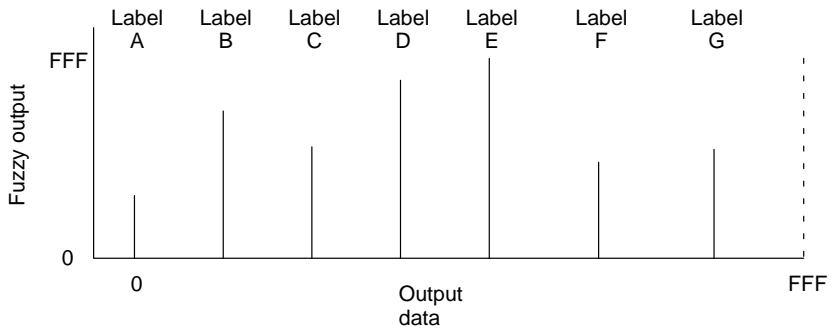
Defuzzification

Defuzzification calculates the final result of the fuzzy logic processing from the fuzzy outputs. There are two methods of defuzzification available in the C200H-FZ001, the center of gravity method and the maximum value method.

Center of Gravity Method
The center of gravity method determines the center of gravity using the fuzzy outputs as the weight and the location of the label as the location. With this method, the final result, R, is the location of the center of gravity calculated with the equation below.

$$R = \frac{\displaystyle\sum_{n=A}^{n=G}(S_n \quad F_n)}{\displaystyle\sum_{n=A}^{n=G}(F_n)}$$
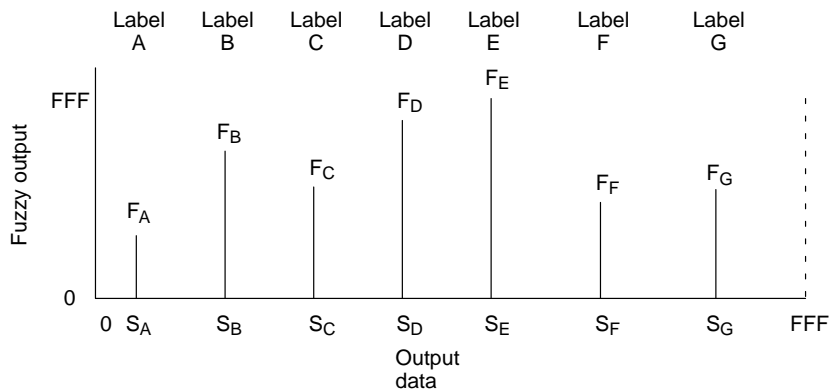
If all of the fuzzy outputs are zero, the divisor in the equation above will also be zero, the division will not be performed and the Error Flag (bit 15) and No Corresponding Output Flag (bit 04) for that output will be turned ON. Refer to *3-3-2 Output Data* for details on the output data format.

Maximum Value Method
The maximum value method uses the location of the label with the maximum fuzzy output as the final result. If two or more fuzzy outputs are equal to the maximum, either the leftmost (minimum) or rightmost (maximum) label location will be used. The programmer determines in advance whether the leftmost or rightmost label location will be used.

In the following diagram, F_B, F_E, and F_G are equal and the final result (R) would be equal to S_B if the leftmost label location were specified, or S_G if the rightmost label location were specified.
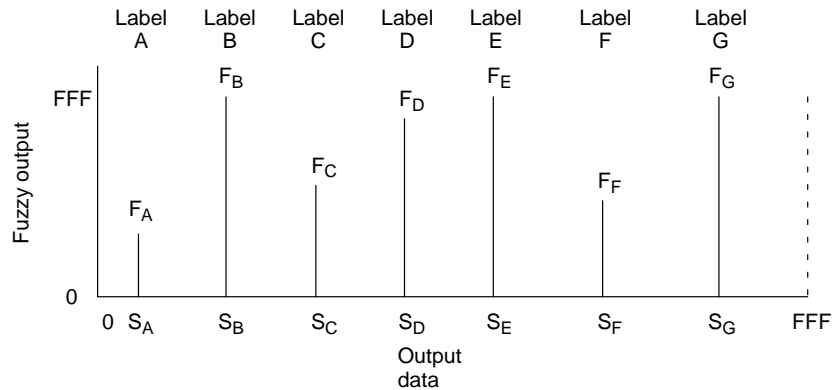
If the maximum fuzzy output is zero, the Error Flag (bit 15) and No Corresponding Output Flag (bit 04) for that output will be turned ON. Refer to *3-3-2 Output Data* for details on the output data format.

1-3-3 Developing the Knowledge Base

The rules and membership functions together are known as the **knowledge base**. This section explains how to organize the rules and membership functions used in the C200H-FZ001.

Determining Objectives

The first step in designing a fuzzy control system is determining the parameters that will be controlled.

1, 2, 3...
1. Decide what part of the controlled system will be controlled by the Unit.
2. One of the strengths of fuzzy logic is its capacity to operate with more than one objective. Decide what the objectives are for the fuzzy control system, the order of priority for those objectives, and the goal for each objective. As an example, the following table lists three objectives for a manufacturing process in order of importance: increasing manufacturing precision, speed, and energy efficiency.

Item	Objective	Goal
Manufacturing precision	Maximize precision	x_1 μm
Manufacturing speed	Maximize speed	x_2 m/s
Energy efficiency	Minimize power consumption	x_3 W

Determining I/O

1, 2, 3...
1. Organize the information about the controlled system and decide what information is necessary to achieve the objectives.
2. Divide the information into inputs and outputs.
3. Organize the information into groups by criteria such as those below:
   - Grouped according to objective.
   - Grouped according to output.
   - Grouped according to input.
   - Grouped according to situation.
   - Grouped according to steps (e.g., condition → goal → output value).
   These groups can be used as they are when creating rules.
4. Review the groups to verify that they satisfy the objectives.

System Configuration

The Fuzzy Logic Unit can be used alone or in a conventional-fuzzy hybrid control system. The best control system will depend on the characteristics of the controlled system.

1, 2, 3...   1. The following diagrams show two versions of conventional-fuzzy hybrid control systems. The first uses fuzzy logic in parallel with the existing control system and the second uses fuzzy logic to refine the output of the existing control system.

Parallel control:

Existing control → α
Fuzzy control → 1-α → Controlled system

Fuzzy logic-refined control:
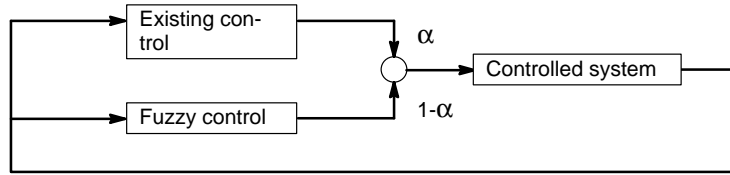
Existing control → ○ → Controlled system
Fuzzy control
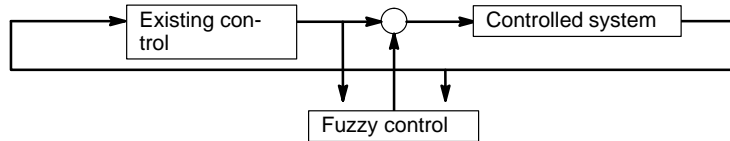
2. Fuzzy logic alone can be used to control the system, as shown below.

Fuzzy logic control:

Fuzzy control → Controlled system
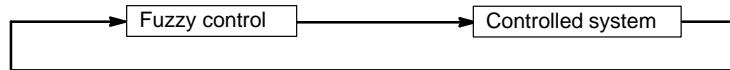
3. A Fuzzy Logic Unit can be installed to act as an interface between the human operator and existing control system in systems that require an operator to adjust settings.

Fuzzy logic interface:

Operator → Fuzzy control
Settings ↓
Existing control → Controlled system
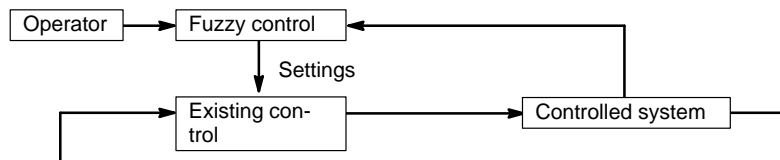
4. A Fuzzy Logic Unit can be used to tune the output of the existing control system. This configuration is similar to the fuzzy logic-refined control system in configuration 1.

Fuzzy logic-tuned control:

Fuzzy control
Tuning ↓
Existing control → Controlled system
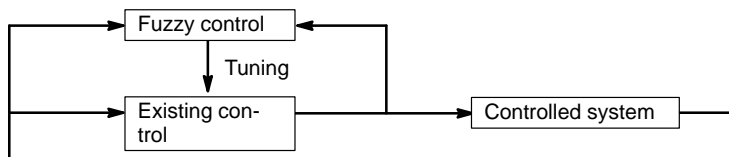
Making the Rules   Make enough rules so that there are outputs from at least two rules in input ranges where precision and smooth operation are required. However, having 3 or more rules with a grade of more than 50% at the same time is redundant.

There must be at least one output from a rule at all times. An error will occur if there isn't an output, so be sure not to allow gaps in input coverage such as the one shown below.
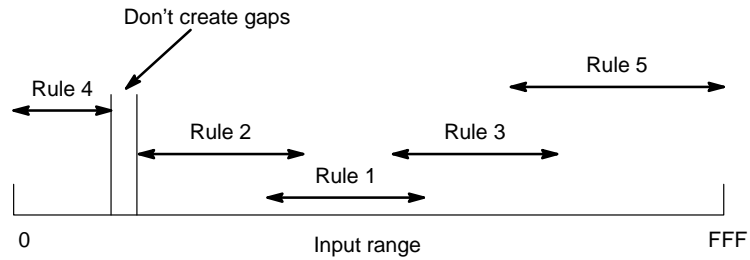
In cases where precise response is not required, make rules that cover a broad section of the input range, as shown below.
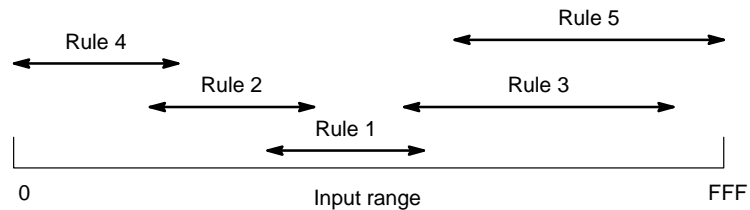
The time required to make and execute the program can be reduced by making a single rule with two conclusion parts instead of two separate rules with just one conclusion part where possible. If the response time is too slow, try grouping rules according to situation, reducing the number of rules as much as possible, and broadening the input range that each rule covers.

We recommend saving to the floppy disk frequently to avoid losing data, and keeping a journal of changes and the reasons for making the changes as an educational resource for the programmer himself and others.

Condition Functions   The four standard shapes for condition membership functions are shown below.
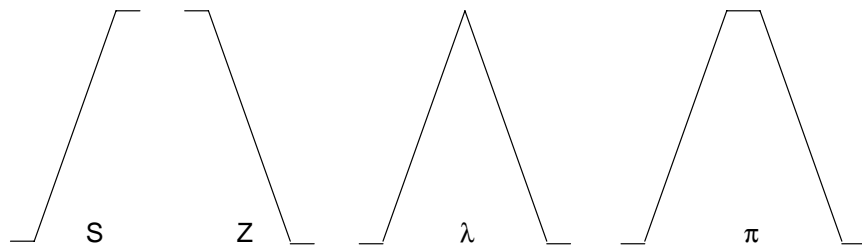
In a closed loop system, we recommend starting with triangular (λ) membership functions that are spaced one-half their width, as shown below. The input range has a large effect on control performance, so proper I/O signal conversion (gain adjustment) is very important.
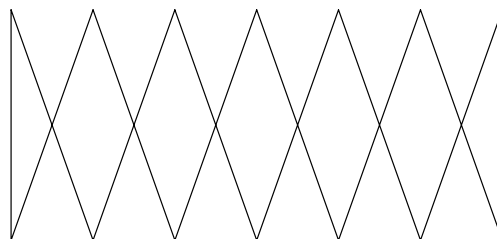
The following diagrams show two effects of modified membership functions. In the diagram on the left, the rules in one section of the input range cover a narrow-

er range for greater sensitivity. In the diagram on the right, the central section of each membership function has no sensitivity to changes in the input value.

Greater sensitivity in this region.

No influence from neighboring membership functions and no sensitivity to changes in input value in these regions.
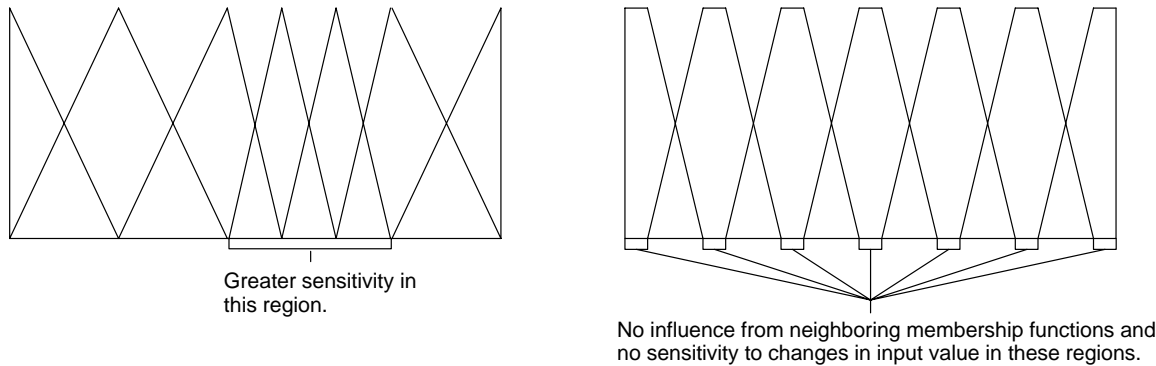
The construction of the membership functions will depend on the characteristics of the controlled system. The arrangement of membership functions can become very asymmetrical and complicated, particularly with open-reel control systems and recognition/identification systems.

Conclusion Function   The conclusion membership function is simply a vertical line. The only setting required is the position of the line.
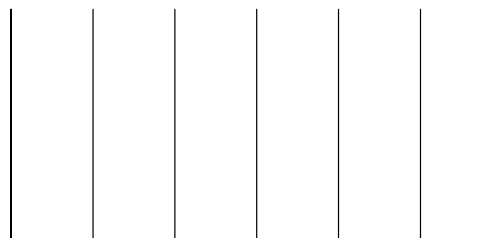
Integrated Development of the Rules and Membership Functions

Follow the procedure below when modifying the knowledge base for improved performance.

1, 2, 3...   1. The first step in designing the knowledge base is to list the rules. Use a standard shape for the membership functions.

After listing the rules, we'll modify the knowledge base based on the results of simulations.

2. Modify the rules, add new rules when needed.

3. Before modifying the membership functions, adjust the I/O gain.

4. Modify the membership functions. Begin by adjusting the width and position of the condition parts. For the conclusion parts, position is the only setting that can be adjusted.

After completing this procedure, test for unusual or unexpected outputs using a variety of input combinations.

1-3-4 Downloading the Knowledge Base

After the knowledge base has been written in the computer, it must be transferred (downloaded) to the Fuzzy Logic Unit. Refer to *4-2 Making and Downloading the Knowledge Base* for details.

SECTION 2
Components and Switch Settings

This section describes the components of the C200H-FZ001 and their functions.

2-1 Components

Front: **Back:**
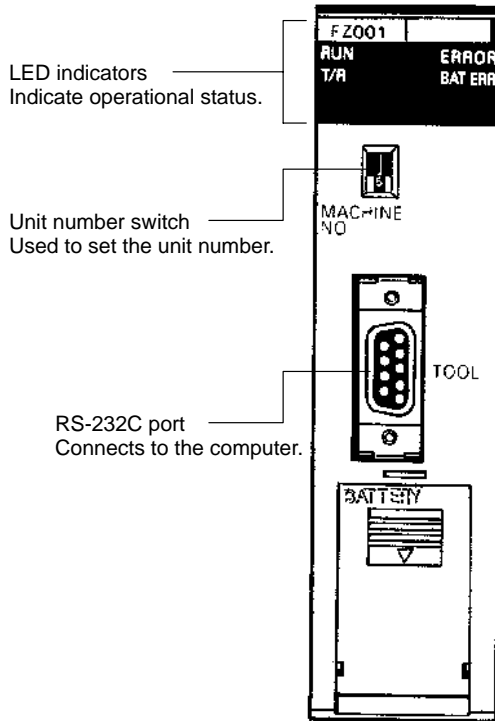
LED indicators
Indicate operational status.

Unit number switch
Used to set the unit number.

RS-232C port
Connects to the computer.

DIP switch 1 ――――   ―――― DIP switch 2
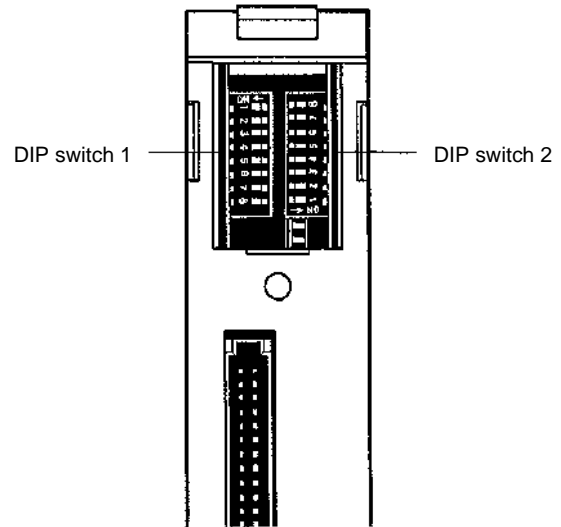
2-2 Indicators

The LED indicators on the front of the Fuzzy Logic Unit provide visual information on the general operation of the Unit. The function of the indicators is described in the following table.

Indicator	Function
RUN	Lights when the Unit is operating normally. (green)
T/R	Flashes when data is being transmitted to or from the Unit. (green)
ERR	Lights when an error is discovered in self-diagnosis operations. Flashes when the knowledge base hasn't been entered or has been damaged. (red)
BAT ERR	Lights when the battery voltage is below the minimum. (red)

2-3 Switch Settings

The setting of the unit number switch on the front of the Unit determines what words in the PC will be allocated to the Unit. The settings of the DIP switches on the back of the Unit determine how the Unit will communicate with the computer through the RS-232C port. Always turn the PC power off before setting the unit number switch or removing the Unit to set the DIP switches.

Unit Number Switch

As a Special I/O Unit, the Fuzzy Logic Unit will be allocated 10 words in the IR Area of the C200H. The 10 words begin at word $100+(10\times n)$, where n is the number set on the unit number switch. If two or more Special I/O Units are set to the same unit number, a non-fatal "I/O UNIT OVER" error will occur in the PC, so be sure not to set the Unit to a unit number that is already being used.

Set the unit number with a standard screwdriver after turning off the power at the CPU. The Unit will not operate properly if the switch is stopped between two numbers.

DIP Switch 1

Pins 5 to 8 of DIP switch 1 are used to set the data format for RS-232C port communications, as shown in the following table. Pins 1 to 4 are not used. Leave these pins in their factory settings.

Pin(s)	Factory setting	Function
1 to 4	OFF	Not used. Leave set to OFF.
5	ON	Sets the number of stop bits: OFF for 1 and ON for 2.
6	OFF	Sets the parity mode: OFF for even and ON for odd.
7	ON	Enables parity: OFF disables parity and ON enables parity.
8	ON	Sets the number of data bits: OFF for 8 and ON for 7.

DIP Switch 2

Pins 1 to 5 of DIP switch 1 are used to set the unit number of the C200H-FZ001 as seen from the computer. The unit number can be 0 to 31, and is set as a binary number. Pin values (when ON) are shown in the table below. (The unit number is set to 0 at the factory.)

Pin value				
Pin 1	**Pin 2**	**Pin 3**	**Pin 4**	**Pin 5**
2^0 (1)	2^1 (2)	2^2 (4)	2^3 (8)	2^4 (16)

Pins 6 to 8 set the baud rate of communications through the RS-232C port, as shown in the table below.

Pin settings			Baud rate
6	**7**	**8**	
OFF	OFF	OFF	Not used.
ON	OFF	OFF	19,200
OFF	ON	OFF	9,600
ON	ON	OFF	4,800
OFF	OFF	ON	2,400
ON	OFF	ON	1,200
OFF	ON	ON	600
ON	ON	ON	300

2-4   RS-232C Interface

The C200H-FZ001 communicates with the computer through the RS-232C port on the front of the Unit. The function of the pins in the RS-232C port are shown in the following table.

Connector Pin Numbers
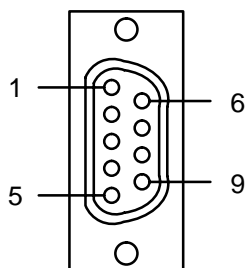
Pin No.	Abbr.	Name	I/O
1	FG	Frame ground	-
2	SD	Send data	Out
3	RD	Receive data	In
4	RS	Request to send	Out
5	CS	Clear to send	In
6	-	Not used	
7	-	Not used	-
8	ER	Enable receive	Out
9	SG	Signal ground	-

Connector: XM3B-0922 (OMRON) or equivalent
                          (set includes connector cover and plug)

Recommended cable: AWG28x5P IFVV-SB (Fujikura Densen)

Cable length: 15 m max.

C200H-FZ001/Computer Connection

Connect the cable to the Unit only after turning off the computer.
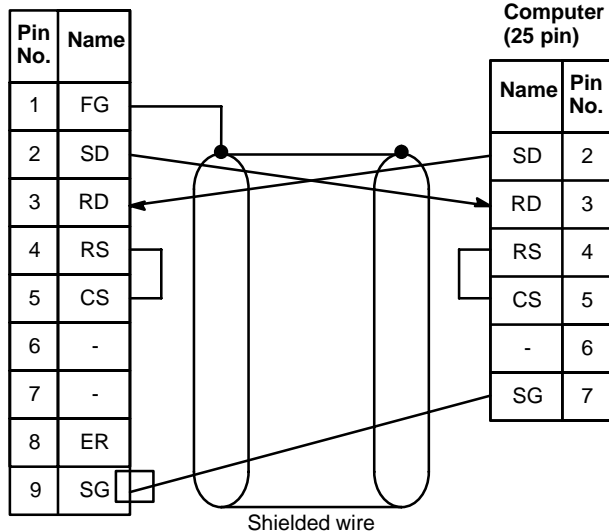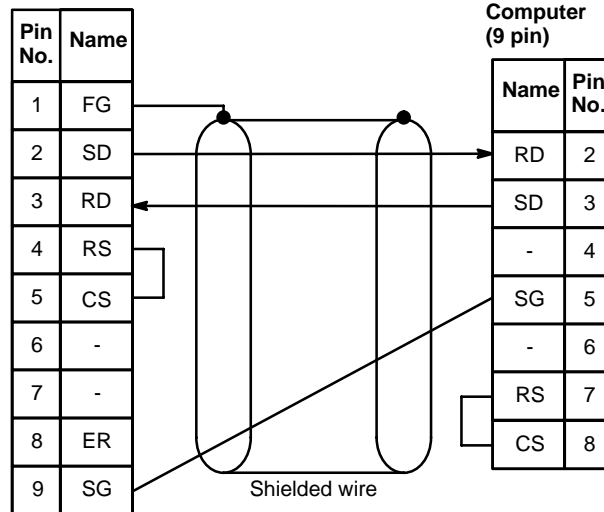
Connecting Cable

The maximum cable length between the C200H-FZ001 and the computer is 15 m. The connecting cable must be purchased separately.

2-5   Dimensions

Dimensions for the C200H-FZ001 are shown below. All dimensions are in millimeters.
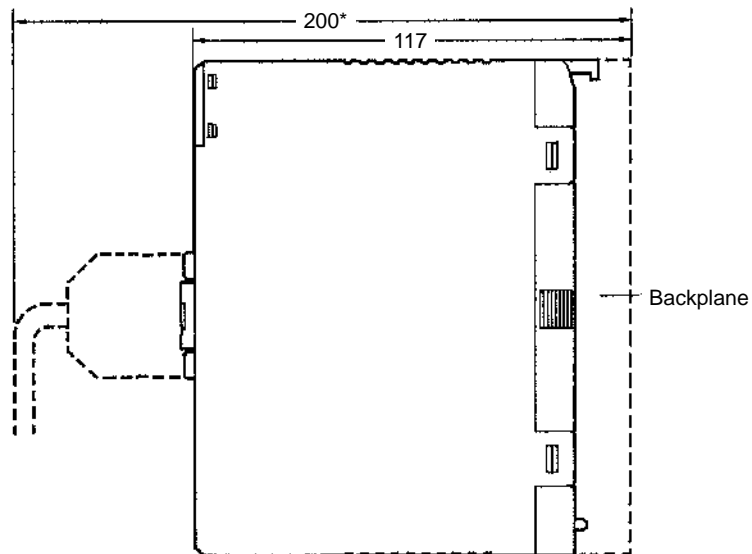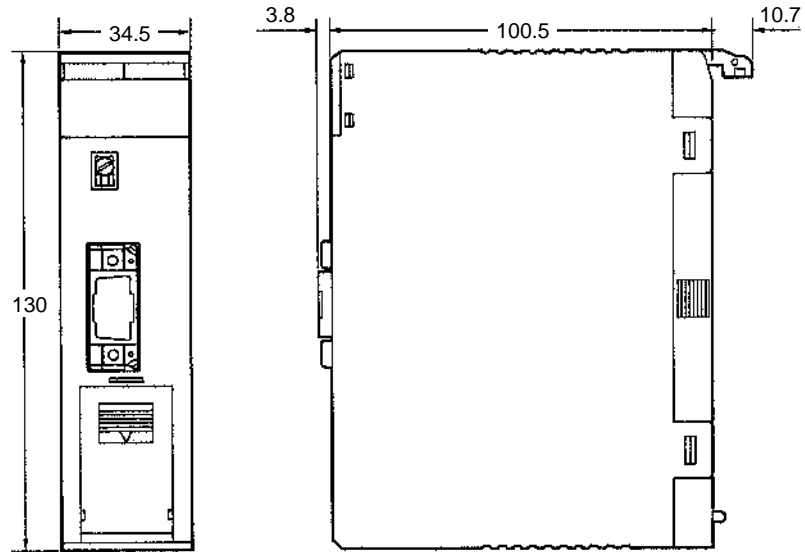
*Allow a total of approximately 200 mm for cable
clearance.

SECTION 3
Memory Allocation

This section explains how words in the C200H are allocated to the Fuzzy Logic Unit, and the function of those words as flags, control bits, and I/O words.

3-1   Memory Allocation in the C200H

As a Special I/O Unit, the Fuzzy Logic Unit will be allocated 10 words in the IR Area of the C200H according to the unit number set on the front of the Unit, as shown below.

Unit number (n)	IR words
0	100 to 109
1	110 to 119
:	:
9	190 to 199

The C200H-FZ001 uses only the first 5 of the 10 words allocated in the C200H. The first 4 words are written from the C200H CPU to the Unit and the fifth word is read from the Unit to the C200H CPU. The last 5 words can be used as work words in the C200H program.

IR words	Function
1n0 to 1n3	Fuzzy Logic Settings
1n4	Fuzzy Logic Unit Flags
1n5 to 1n9	Available as work words

Note If two or more Special I/O Units are set to the same unit number, a non-fatal "I/O UNIT OVER" error will occur in the PC, so be sure not to set the Unit to a unit number that is already being used.

3-2   IR Area Words

This section describes the functions of the IR words allocated to the Fuzzy Logic Unit in the C200H. An "n" in an IR word in this section indicates the unit number set on the front of the Unit.

Fuzzy Logic Settings

The IR words in the table below contain settings and control bits that are transferred from the C200H CPU to the C200H-FZ001. The settings and control bits listed below are described in more detail following the table.

IR word[1]	Bit(s)	Function
1n0	00 to 03	Number of Inputs (1 to 8, BCD)
	04 to 14	Not used.
	15	Processing Start Bit
1n1	00 to 15	First Input Word
1n2	00 to 03	Number of Outputs (1 to 8, BCD)
	04 to 15	Not used.
1n3	00 to 15	First Output Word

Note   1. The "n" in the IR word column indicates the unit number set on the unit number switch on the front of the Unit.

2. Bits not used in IR 1n0 to IR 1n3 cannot be used as work bits in the program.

Number of Inputs   Bits IR 1n000 to IR 1n003 specify the number of inputs that are written to the Unit for fuzzy logic processing. The number must be BCD, between 1 and 8. The location of the input data in the PC is indicated by address specified for the First Input Word, described below.

Processing Start Bit   Turning ON the Processing Start Bit (IR 1n015) initializes fuzzy logic processing in the Unit.

First Input Word   IR 1n1 specifies the address of the word that contains the data for the first input. The leftmost digit specifies the data area and the rightmost 3 digits specify the word in that area. The rightmost 3 digits must be BCD, but the leftmost digit can take any of the values shown in the table below.

Data area	Bit status			IR 1n1 leftmost digit value
	Bit 13	**Bit 14**	**Bit 15**	
DM	0	0	0	0
IR or SR	0	0	1	8
HR	0	1	0	4
AR	0	1	1	C
LR	1	0	0	2
TC	1	0	1	A

As an example, a value of C007 would specify AR 007 as the first input word. Input data for the second input would be contained in AR 008, and so on.

Number of Outputs   Bits IR 1n200 to IR 1n203 specify the number of outputs (results of fuzzy logic processing) that are read from the Unit. The number must be BCD, between 1 and 8. The location of the output data is indicated by address specified for the First Output Word, described below.

First Output Word   IR 1n3 specifies the address of the word that contains the first output. The leftmost digit specifies the data area and the rightmost 3 digits specify the word in that area. The rightmost 3 digits must be BCD, but the leftmost digit can take any of the values shown in the table under the First Input Word heading, above.

As an example, a value of 0100 would specify DM 0100 as the first output word. The first 4 words, beginning with the First Output Word, contain output data for the 4 outputs and the next 4 words contain error information for those 4 outputs. This format is always used regardless of the number of outputs specified in IR 1n2. Refer to *3-3-2 Output Data* for details.

Fuzzy Logic Unit Flags

Word IR 1n4 contains flags that are transferred from the C200H-FZ001 to the C200H CPU. These flags are turned ON and OFF to indicate the operating status of the Fuzzy Logic Unit.

IR 1n4 bit(s)[1]	Flag name	Function
00	Results Output Flag	Turned ON to indicate that the results of fuzzy logic processing have been output.
01 and 02	---	Not used.
03	Setting Error Flag	Turned ON to indicate that an error has been found in the output word (1n0 to 1n3) settings.
04	Fuzzy Error Flag	Turned ON to indicate that an error has occurred during the self-diagnostic check.
05	Memory Error Flag	Turned ON to indicate that an error has been found in the knowledge base.
06	Battery Error Flag	Turned ON to indicate that the battery voltage has fallen below the minimum, or that the battery isn't installed.
07	Processing Enabled Flag	Turned ON to indicate that the Unit is ready to begin fuzzy logic processing.
08 to 15	---	Not used.

Note   1. The "n" in the first column indicates the unit number set on the unit number switch on the front of the Unit.

2. Bits not used in IR 1n4 cannot be used as work words in the program.

3-3 I/O Data Format

The Fuzzy Logic Unit receives input data from the C200H CPU, calculates the results for up to four outputs, and outputs those results to the C200H CPU. This section describes the structure of that input and output data.

3-3-1 Input Data

The number and location of the inputs is specified by the user in the Fuzzy Logic Settings, as described in *3-2 IR Area Words*. Each input is allocated one word, and the input data is contained in bits 0 to 12 of that word in 3-digit hexadecimal. The input range is thus 000 to FFF (0 to 4095 decimal). Bits 12 to 15 are not used.

The data structure for inputs is shown in the following table beginning from the First Input Word, I. In this case there are 8 inputs, but anywhere from 1 to 8 inputs can be used.

Word	Bits 12 to 15	Bits 0 to 11
I	Not used. Set to zero.	Input 0 data
I+1	Not used. Set to zero.	Input 1 data
⋮	⋮	⋮
I+7	Not used. Set to zero.	Input 7 data

3-3-2 Output Data

The number and location of the outputs is specified by the user in the Fuzzy Logic Settings, as described in *3-2 IR Area Words*. Each output is allocated one word, and the output data is contained in bits 0 to 12 of that word in 3-digit hexadecimal. The output range is thus 000 to FFF (0 to 4095 decimal). Bits 12 to 15 are not used.

The output data structure is shown in the following table beginning from the First Output Word, O. Four words (from O to O+3) are allocated to output data regardless of the number of outputs specified in the Fuzzy Logic Settings.

Word	Bits 12 to 15	Bits 0 to 11
O	Not used. Set to zero.	Output 0 data
O+1	Not used. Set to zero.	Output 1 data
O+2	Not used. Set to zero.	Output 2 data
O+3	Not used. Set to zero.	Output 3 data

Error codes for outputs 0 to 4 are contained in words O+4 to O+7, as shown in the following table. These four words are allocated regardless of the number of outputs specified in the Fuzzy Logic Settings.

Word	Bit 15	Bits 5 to 14	Bits 0 to 4
O+4	ER	Not used. Set to zero.	Output 0 error code
O+5	ER	Not used. Set to zero.	Output 1 error code
O+6	ER	Not used. Set to zero.	Output 2 error code
O+7	ER	Not used. Set to zero.	Output 3 error code

The Error Flag (ER) will be turned ON when any of bits 0 to 4 for that output are turned ON. The following table shows the function of the error code bits.

Bit	Flag name	Function
00	Input Range Exceeded Flag	Turned ON to indicate that the input range has been exceeded.
01	Too Many Input Words Flag	Turned ON to indicate that too many inputs have been transferred to the Unit.
02	Too Few Input Words Flag	Turned ON to indicate that not enough inputs have been transferred to the Unit.
03	Undefined Output Flag	Turned ON to indicate that the output was not defined in the Fuzzy Logic Settings.
04	No Corresponding Output Flag	Turned ON to indicate that no grade has been output for that output.

It isn't absolutely necessary to read the error codes, but we recommend reading them to find any errors in fuzzy logic processing as soon as possible. If you determine that it isn't necessary to read the error codes, set the Number of Outputs setting in the Fuzzy Logic Settings to the required number of outputs.

SECTION 4
Program and Knowledge Base

This section explains how to prepare the two basic software requirements for Fuzzy Logic Unit operation: the sub-program used in the C200H and the knowledge base used in the Fuzzy Logic Unit itself.

4-1   C200H Sub-program

A sub-program must be prepared within the C200H program to control the transfer of data between the C200H and the Unit. This program can be written with standard instructions such as MOV(21). The four basic steps in this sub-program are listed below.

1, 2, 3...   1. Specify (in the Fuzzy Logic Settings) the words to which the input and output data will be written.

2. Transfer the input data to the words specified in 1.

3. Turn ON the Processing Start Bit to initialize fuzzy processing.

4. Transfer the output data from the words specified in 1 when the Results Output Flag is turned ON.

Example Sub-program

The following example shows a sub-program for a system with 2 Analog Input Units and 2 Analog Output Units. Each Analog Input Unit has 4 inputs and each Analog Output Unit has 2 outputs, for a total of 8 inputs and 4 outputs.

Fuzzy Logic Settings

In this example the unit number is set to 5 (n=5), so words IR 150 to IR 159 are automatically allocated to the Fuzzy Logic Unit. Refer to *3-2 IR Area Words* for details on the function of bits in words IR 150 to IR 154.

IR words	Function
150 to 153	Fuzzy Logic Settings
154	Fuzzy Logic Unit Flags
155 to 159	Available as work words

The 8 inputs will be contained in DM 0100 to DM 0107. While only 3 of the 4 outputs are needed, the structure of the outputs is fixed at 4 output and 4 error code words which will be contained in DM 0110 to DM 0117.

Word	Function	Value
IR 150	Number of Inputs (8)	0008
IR 151	First Input Address (DM 0100)	0100
IR 152	Number of Outputs (3)	0003
IR 153	First Output Address (DM 0110)	0110

The ladder diagram and mnemonics for inputting these settings are shown below.

```
00000
─┤├────────┬──────────────────────────────┤ MOV(21) │
           │                                │  #0008  │
           │                                │   150   │
           │
           ├──────────────────────────────┤ MOV(21) │
           │                                │  #0100  │
           │                                │   151   │
           │
           ├──────────────────────────────┤ MOV(21) │
           │                                │  #0003  │
           │                                │   152   │
           │
           └──────────────────────────────┤ MOV(21) │
                                            │  #0110  │
                                            │   153   │
```
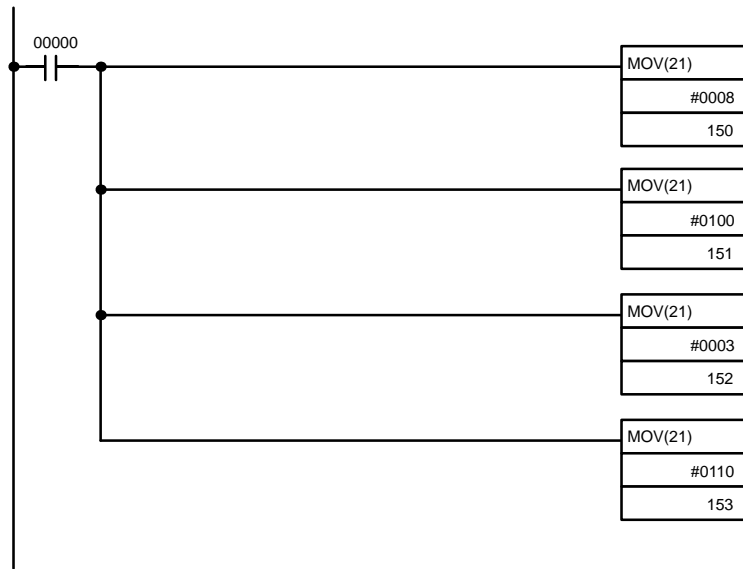
Address	Instruction	Operands	
00000	LD		00000
00001	MOV(21)		
		#	0008
			150
00002	MOV(21)		
		#	0100
			151

Address	Instruction	Operands	
00003	MOV(21)		
		#	0003
			152
00004	MOV(21)		
		#	0110
			153

Input Data Transfer and Fuzzy Processing Initialization

The unit number of the Analog Input Units (Special I/O Units) are set to 0 and 1, so the digital input data will be stored automatically in IR 100 to IR 104 and IR 110 to IR 114. The input data for these 8 inputs must be transferred to the words specified by the First Input Address (DM 0100 to DM 0107). After the input data is transferred, fuzzy logic processing is started by turning ON the Processing Start Bit (15015).

The ladder diagram and mnemonics for transferring the input data and initializing processing are shown below. (To save space, only the first two input words are shown; the other six follow the same pattern.)
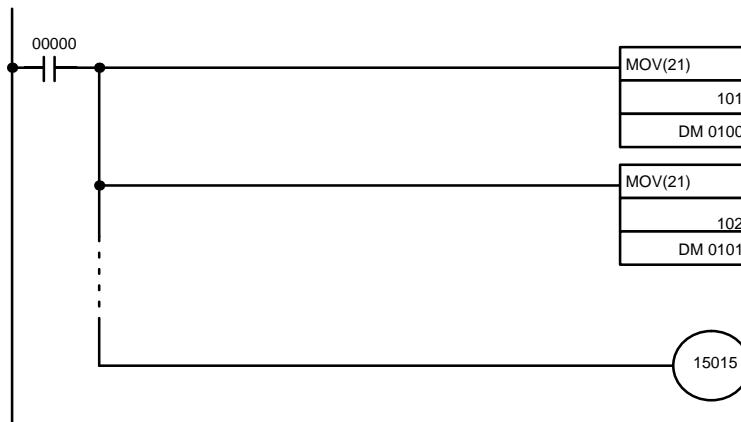
Address	Instruction	Operands	
00000	LD		00000
00001	MOV(21)		
			101
		DM	0100
00002	MOV(21)		
			102
		DM	0101
⋮	⋮	⋮	
00009	OUT		15015

Note The Fuzzy Logic Settings, Input Data Transfer, and Initialize Fuzzy Processing mnemonics can all be input after a single LD 00000 instruction to reduce the size of the program.

28

Output Data Transfer

The results of the fuzzy logic processing and error codes are output automatically to the words beginning at the First Output Address, so it isn't always necessary to transfer the output data, but in this case the data must be transferred to the words allocated to the Analog Output Units.

The unit number of the Analog Output Units (Special I/O Units) are set to 2 and 3, so the digital output data will be loaded from IR 120, IR 121, IR 130, and IR 131. The ladder diagram and mnemonics for transferring the output data are shown below.

```
        00000   15400
         ┤├──────┤├───────┬──────────────┌──────────────┐
                          │              │ MOV(21)       │
                          │              ├──────────────┤
                          │              │   DM 0110     │
                          │              ├──────────────┤
                          │              │       120     │
                          │              └──────────────┘
                          │              ┌──────────────┐
                          ├──────────────│ MOV(21)       │
                          │              ├──────────────┤
                          │              │   DM 0111     │
                          │              ├──────────────┤
                          │              │       121     │
                          │              └──────────────┘
                          │              ┌──────────────┐
                          ├──────────────│ MOV(21)       │
                          │              ├──────────────┤
                          │              │   DM 0112     │
                          │              ├──────────────┤
                          │              │       130     │
                          │              └──────────────┘
                          │              ┌──────────────┐
                          └──────────────│ MOV(21)       │
                                         ├──────────────┤
                                         │   DM 0113     │
                                         ├──────────────┤
                                         │       131     │
                                         └──────────────┘
```
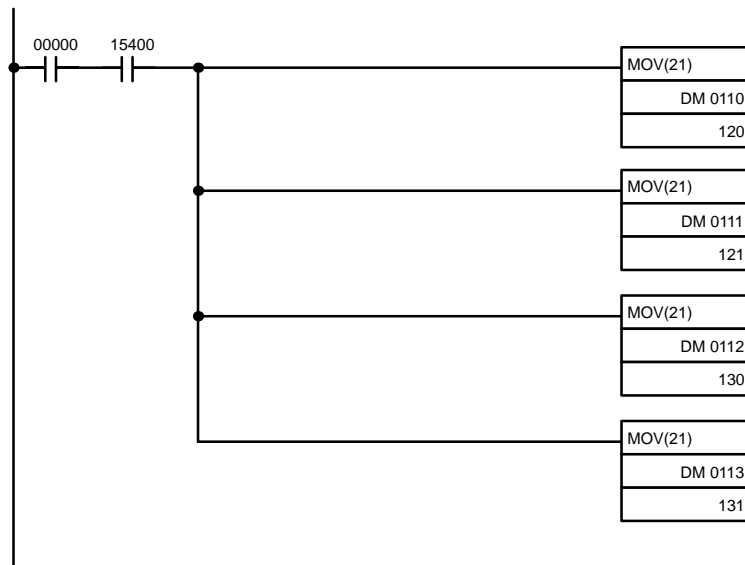
Address	Instruction	Operands	
00000	LD		00000
00001	AND		15400
00002	MOV(21)		
		DM	0110
			120
00003	MOV(21)		
		DM	0111
			121

Address	Instruction	Operands	
00004	MOV(21)		
		DM	0112
			130
00005	MOV(21)		
		DM	0113
			131

4-2   Making and Downloading the Knowledge Base

The knowledge base must be created with the FSS in the computer and then downloaded to the Fuzzy Logic Unit. The basic steps involved in making and downloading the knowledge base are listed below.

1, 2, 3...   1. Make the knowledge base using the FSS.

2. Verify that the knowledge base has been written correctly.

3. Download the knowledge base to the Unit.

The operations that can be performed with the FSS are described briefly in a table below. Refer to the *Fuzzy Support Software Operation Manual* for details on these FSS operations.

Operation		Function
File management	Making a new knowledge base file	Clears the knowledge base file being edited and starts making a new one.
	Loading a knowledge base file	Loads a knowledge base file from the floppy or hard disk and starts editing.
	Saving a knowledge base file	Saves the knowledge base file being edited to the floppy or hard disk.
	Recovering the Unit's knowledge base file	Recovers the knowledge base file that was set in the Unit for editing.
	Saving the Unit's processing data	Saves the Unit's file data as a file.
	Print	Prints the knowledge base file being edited and the Unit's operating status.
Making the knowledge base	Making membership functions	Edits the condition and conclusion membership functions.
	Making the rules	Edits the rules.
	Defuzzification method	Sets (for each output) the center of gravity, leftmost maximum value, or rightmost maximum value method. The default is center of gravity.
	No rule grade setting	Sets (for each output) the grade that will be output when all of the rule grades are zero. The default is 4095.
Verifying the knowledge base	General check	Checks whether the knowledge base can be used in fuzzy logic processing.
	Detail check	Checks for redundancies in the Knowledge base.
Online (link) operations	Affirming the Unit's knowledge base	Loads the knowledge base from the Unit and displays information such as the knowledge base name.
	Knowledge base comparison	Compares the knowledge base in the Unit to the one being edited.
	Inference	Starts or stops fuzzy logic processing.
	Knowledge base download	Downloads the knowledge base file being edited to the Unit.
	Processing status monitor	Monitors the processing status of the Unit in real-time.
	Unit initialization	Initializes the Unit.

SECTION 5
Example Application

This section describes the development of an actual control system that uses a C200H-FZ001 Fuzzy Logic Unit.

5-1 Conveyor Belt Control System Summary

In this example application, a Fuzzy Logic Unit is used to control two conveyor belts for product packaging. The products are carried on conveyor A at random intervals, but at a fixed speed. The boxes are carried at regular intervals on conveyor B, which runs parallel to conveyor A at a speed controlled by the Fuzzy Logic Unit.

The Fuzzy Logic Unit adjusts the speed of conveyor B so that the boxes arrive at the same time as the products. The required information is the offset between the product and the box and the rate that the offset is changing. A conventional control system eliminates the offset between the product and the box, but sometimes no speed adjustment is necessary even though there is a large offset, so the system controlled by the Fuzzy Logic Unit is much more efficient.
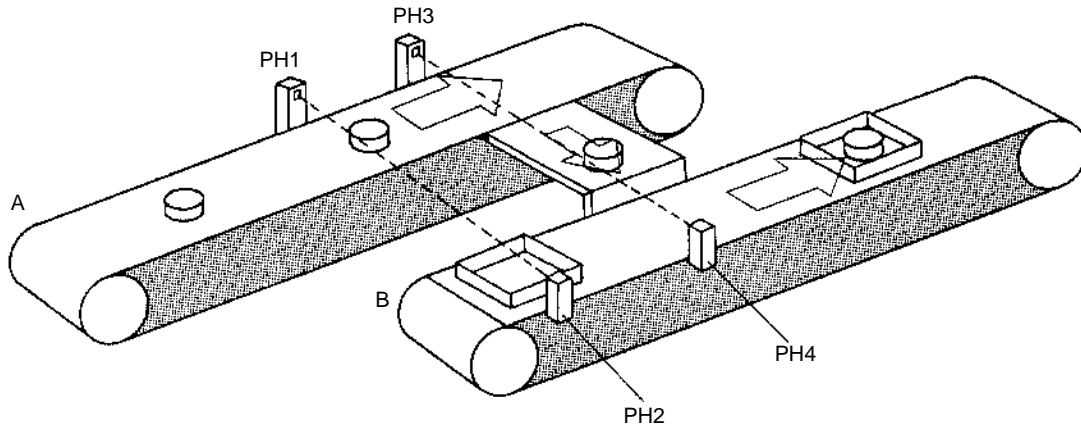
Conveyor A: Carries the products.
Conveyor B: Carries the boxes.
PH 1 to PH 4:   Photoelectric sensors

System Configuration
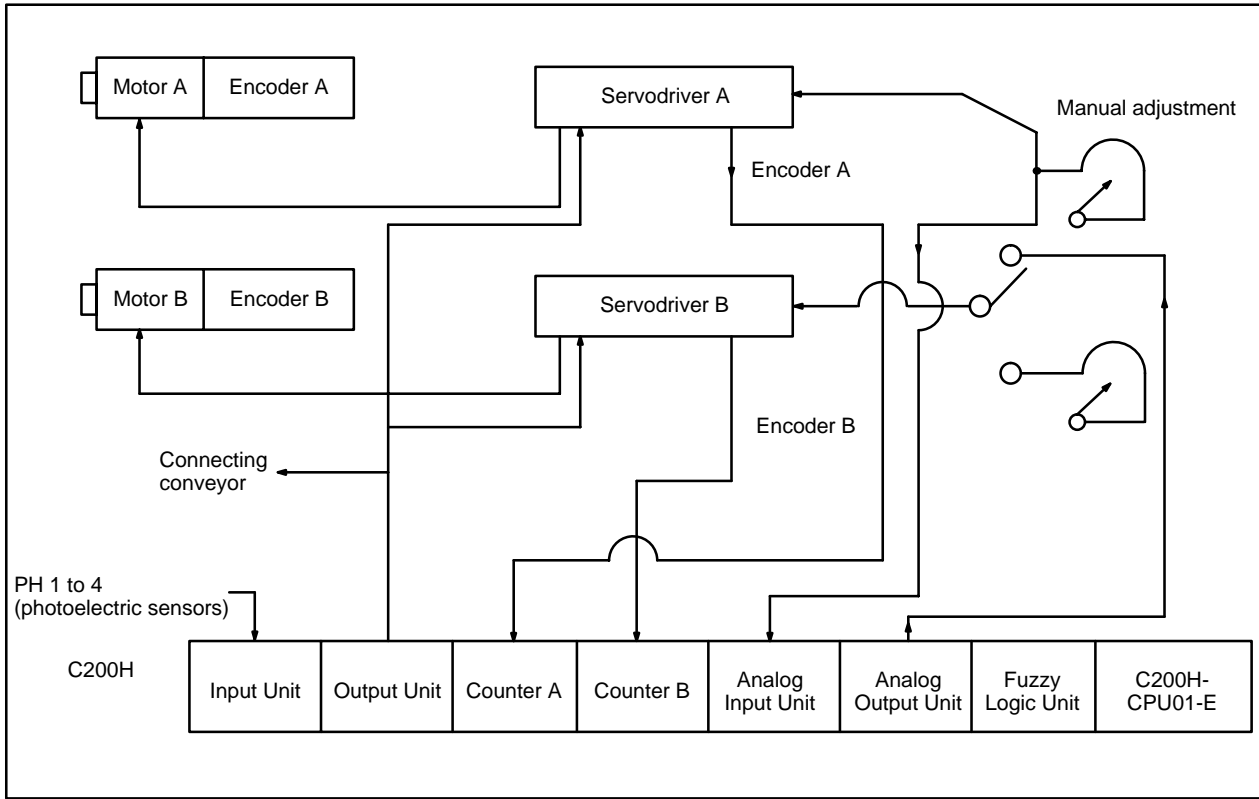
Parts List

Part	Function	Model number
Motor A (B)	Drives conveyor A (B).	---
Rotary encoder A (B)	Encoder for motor A (B)	---
Servodriver A (B)	Servodriver controlling motor A (B)	---
Photoelectric sensors (4)	Sense passing products and boxes.	---
Input Unit	Receives photoelectric sensor inputs.	C200H-ID215
Output Unit	Outputs to servodrivers and connecting conveyor.	C200H-OD215
High-speed Counter A (B)	Counts the output of rotary encoder A (B).	C200H-CT001
Analog Input Unit	Converts analog speed data from servodriver A to digital data.	C200H-AD001
Analog Output Unit	Converts the digital output from fuzzy logic processing to analog data and outputs it to servodriver B.	C200H-DA001
Fuzzy Logic Unit	Handles fuzzy logic processing.	C200H-FZ001

System Operation   The distance that the box travels from the time that it passes the first photoelectric sensor on conveyor B (PH 2) until the product passes the first photoelectric sensor on conveyor A (PH 1) can be calculated from the rotary encoders attached to the motors and High-speed Counter Units A and B. We will call this distance the product/box offset.

The first rules in the Fuzzy Logic Unit will be based on this offset:

"If the product/box offset is large, then slow down conveyor B."

"If the product/box offset is small, then don't change conveyor B's speed much."

Fuzzy logic processing using these rules and rules based on the rate of change of the product/box offset allow very smooth adjustment of conveyor B's speed.

Note We assume that only one product or box travels between the photoelectric sensors at a time.

5-2 Making the Knowledge Base

5-2-1 Input Data

The input data for the product/box offset is taken from the relative product/box position (E) calculated as shown below:

E = (Count from rotary encoder A) - (Count from rotary encoder B)

The input data for the rate of change of the offset (ΔE) is simply the difference between the most recent value of E (E_n) and the previous value of E (E_{n-1}) as shown below:

$\Delta E = (E_n) - (E_{n-1})$

5-2-2 Membership Functions

The diagrams in this section show the condition membership functions for the product/box offset and offset rate of change, as well as the conclusion membership function (conveyor B speed adjustment).

Product/Box Offset

Offset Rate of Change

```
 F: File | M: Make | C: Check | L: Link | H: Help       Model No: C200H

  File:   CONVEYOR     Blk Name:CONVEYOR    I/P No: 1      Signal: DE
  Label   [NL ][NM ][    ][ZR ][    ][PM ][PL ]   IF Range:       0-    4095
                                                  Ref Range:       -

 X:     0       1     1025    ----    ----    ----
 Y:   1.00    1.00    0.00    ----    ----    ----                      0

   1.00

   0.00
        0                                                        4095
                                                                   Edit
 1 Copy  2Delete 3 Range 4 Edit  5 Shift 6 Zoom 7Prv Sig8Nxt Sig9Confirm0 Prev
```

Conveyor B Speed Adjustment

```
 F: File | M: Make | C: Check | L: Link | H: Help       Model No: C200H

  File:   CONVEYOR     Blk Name:CONVEYOR  O/P No:    0      Signal: VB
  Label   [NL ][NM ][NS ][ZR ][PS ][PM ][PL ]   IF Range:       0-    4095
                                               Ref Range:       -

 X:      1
 Y:    1.00

   1.00

   0.00
        0                                                        4095
                                                                   Edit
 1 Copy  2Delete 3 Range 4 Edit  5        6 Zoom 7Prv Sig8Nxt Sig9Confirm0 Prev
```

5-2-3 Creating Rules

Expressing the Rules   Start creating the rules by organizing your "know-how" or "past experience" of the system in everyday expressions. A good way to organize these expressions is a table like the one below.

E \\ ΔE	Box is ahead		About even		Product is ahead
Box is faster	Slow box a lot	Slow box a lot	Slow box a little	Speed it up a little	Speed it up
	Slow box a lot	Slow the box	Slow box a little	Speed it up a little	Speed it up
About even	Slow the box	Slow box a little	Don't change	Speed it up a little	Speed it up
	Slow the box	Slow box a little	Speed it up a little	Speed it up	Speed it up a lot
Box is slower	Slow the box	Slow box a little	Speed it up a little	Speed it up	Speed it up a lot

Converting to Labels   Next, convert the input data, output data, and the expressions from the table above to labels, as shown below.

Input data:

Product/box offset: E

Offset rate of change: DE (ΔE)

Output data:

Conveyor B speed adjustment: VB

The expressions from the table above can be converted as follows:

E \\ DE	NL	NS	ZR	PS	PL
PL	NL	NL	NS	PS	PM
PS	NL	NM	NS	PS	PM
ZR	NM	NS	ZR	PS	PM
NS	NM	NS	PS	PM	PL
NL	NM	NS	PS	PM	PL

Converting to If/Then   Next, convert the labels from table form to if/then statement form, which can be entered into a computer.

IF E=NL AND DE=PL THEN VB=NL

IF E=NL AND DE=PS THEN VB=NL

IF E=NL AND DE=ZR THEN VB=NM

IF E=NL AND DE=NS THEN VB=NM

IF E=NL AND DE=NL THEN VB=NM

IF E=NS AND DE=PL THEN VB=NM

IF E=NS AND DE=PS THEN VB=NM

IF E=NS AND DE=ZR THEN VB=NS

IF E=NS AND DE=NS THEN VB=NS

IF E=NS AND DE=NL THEN VB=NS

IF E=ZR AND DE=PL THEN VB=NS

IF E=ZR AND DE=PS THEN VB=NS

IF E=ZR AND DE=ZR THEN VB=ZR

IF E=ZR AND DE=NS THEN VB=PS

IF E=ZR AND DE=NL THEN VB=PS

IF E=PS AND DE=PL THEN VB=PS

IF <u>E=PS</u> AND <u>DE=PS</u> THEN <u>VB=PS</u>

IF <u>E=PS</u> AND <u>DE=ZR</u> THEN <u>VB=PS</u>

IF <u>E=PS</u> AND <u>DE=NS</u> THEN <u>VB=PM</u>

IF <u>E=PS</u> AND <u>DE=NL</u> THEN <u>VB=PM</u>

IF <u>E=PL</u> AND <u>DE=PL</u> THEN <u>VB=PM</u>

IF <u>E=PL</u> AND <u>DE=PS</u> THEN <u>VB=PM</u>

IF <u>E=PL</u> AND <u>DE=ZR</u> THEN <u>VB=PM</u>

IF <u>E=PL</u> AND <u>DE=NS</u> THEN <u>VB=PL</u>

IF <u>E=PL</u> AND <u>DE=NL</u> THEN <u>VB=PL</u>

Entering If/Then Statements   Finally, enter the if/then statements into the computer for downloading to the Fuzzy Logic Unit.

5-3   I/O Allocation and Ladder Program

5-3-1   I/O Allocation

The following table shows the I/O words allocated to Units mounted on the CPU Rack. The unit number is set with a switch on the front of Special I/O Units.

Unit	Unit number	I/O word(s)
Input Unit	---	IR 000
Output Unit	---	IR 001
High-speed Counter Unit A	0	IR 100 to IR 109
High-speed Counter Unit B	1	IR 110 to IR 119
Analog Input Unit	2	IR 120 to IR 129
Analog Output Unit	3	IR 130 to IR 139
Fuzzy Logic Unit	4	IR 140 to IR 149

The following table shows the I/O bit allocation in the Input and Output Units.

Unit	I/O Bit	Function
Input Unit	00002	PH 1 (first photoelectric sensor on conveyor A)
	00003	PH 3 (second photoelectric sensor on conveyor A)
	00005	PH 1 (first photoelectric sensor on conveyor B)
	00007	PH 4 (second photoelectric sensor on conveyor B)
Output Unit	00102	Turns on the connecting conveyor between A and B.
	00104	Turns on servodriver A.
	00105	Turns on servodriver B.

5-3-2   DM Settings for Special I/O Units

Some of the Special I/O Units used in this system require control words. Here the control data is written in an area of DM that is read-only from the program. This data must be stored in DM with a Programming Console before operation.

High-speed Counter Units (Units Number 0 and 1)

Word	Content	Function
DM 1000	0400	Specifies Gate mode for Unit A (unit number 0).
DM 1001	0001	Specifies internal inputs for Unit A.
DM 1100	0400	Specifies Gate mode for Unit B (unit number 1).
DM 1101	0001	Specifies internal inputs for Unit B.

Analog Input Unit (Unit Number 2)

Word	Content	Function
DM 1200	0002	Scaling LOW data for input 1
DM 1201	4095	Scaling HIGH data for input 1
DM 1202	0002	Scaling LOW data for input 2
DM 1203	4095	Scaling HIGH data for input 2
DM 1204	0002	Scaling LOW data for input 3
DM 1205	4095	Scaling HIGH data for input 3
DM 1206	0002	Scaling LOW data for input 4
DM 1207	4095	Scaling HIGH data for input 4
DM 1208	9999	Mean function data for input 1
DM 1209	9999	Mean function data for input 2
DM 1210	9999	Mean function data for input 3
DM 1211	9999	Mean function data for input 4

Analog Output Unit (Unit Number 3)

Word	Content	Function
DM 1301	0FFF	Specifies the output upper limit.

5-3-3 Ladder Program

The ladder diagram sub-program that controls the conveyor belt system is shown below. Refer to *5-3-4 Adjusting I/O Data* for a description of special aspects of the following program.

25315
```
├─┤├──────────────────────────────────────┤ MOV(21)  │
                                          ├──────────┤
                                          │   #0000  │
                                          ├──────────┤
                                          │  DM0180  │
                                          └──────────┘
```

03000
```
├─┤├────●─────────────────────────────────┤ CLC(41)  │
        │                                 └──────────┘
        │
        ●─────────────────────────────────┤ SUB(31)  │     Calculates the present pro-
                                          ├──────────┤     duct/box offset.
                                          │  DM0110  │
                                          ├──────────┤
                                          │  DM0120  │
                                          ├──────────┤
                                          │  DM0130  │
                                          └──────────┘
```

   25504(CY)
```
        ├─┤├──────●──────────────────────( 03110 )        Turned ON if the box is
                  │                                       ahead of the product.
                  ●───────────────────────┤ CLC(41)  │
                  │                       └──────────┘
                  │
                  ●───────────────────────┤ SUB(31)  │
                                          ├──────────┤
                                          │  DM0180  │
                                          ├──────────┤
                                          │  DM0130  │
                                          ├──────────┤
                                          │  DM0130  │
                                          └──────────┘
```

03000
```
├─┤├────●─────────────────────────────────┤ CLC(41)  │
        │                                 └──────────┘
        │
        ●─────────────────────────────────┤ SUB(31)  │     Calculates the rate of
                                          ├──────────┤     change of the offset.
                                          │  DM0130  │
                                          ├──────────┤
                                          │  DM0132  │
                                          ├──────────┤
                                          │  DM0134  │
                                          └──────────┘
```

   25504(CY)
```
        ├─┤├──────●──────────────────────( 03111 )        Turned ON if the box and
                  │                                       product are getting closer.
                  ●───────────────────────┤ CLC(41)  │
                  │                       └──────────┘
                  │
                  ●───────────────────────┤ SUB(31)  │
                                          ├──────────┤
                                          │  DM0180  │
                                          ├──────────┤
                                          │  DM0134  │
                                          ├──────────┤
                                          │  DM0134  │
                                          └──────────┘
```

25315
```
├─┤├────●─────────────────────────────────┤ MOV(21)  │     Sets the factor by which
        │                                 ├──────────┤     the product/box offset
        │                                 │   #0003  │     will be adjusted.
        │                                 ├──────────┤
        │                                 │  DM0100  │
        │                                 └──────────┘
        │
        ●─────────────────────────────────┤ MOV(21)  │     Sets the factor by which
                                          ├──────────┤     the rate of change of the
                                          │   #0003  │     offset will be adjusted.
                                          ├──────────┤
                                          │  DM0102  │
                                          └──────────┘
```

03000
```
├─┤├────●─────────────────────────────────┤ MUL(32)  │
        │                                 ├──────────┤
        │                                 │  DM0130  │     Adjusts the product/box
        │                                 ├──────────┤     offset by factor set above.
        │                                 │  DM0100  │
        │                                 ├──────────┤
        │                                 │  DM0140  │
        │                                 └──────────┘
        │
        ●─────────────────────────────────┤ MUL(32)  │
                                          ├──────────┤
                                          │  DM0134  │     Adjusts the rate of change of
                                          ├──────────┤     the offset by factor set above.
                                          │  DM0102  │
                                          ├──────────┤
                                          │  DM0150  │
                                          └──────────┘
```

```
  03000        03110
───┤├──────●────┤/├─────────────────────────┌──────────────┐
            │                                │ ADD(30)      │
            │                                ├──────────────┤
            │                                │        #2048 │
            │                                ├──────────────┤
            │                                │       DM0140 │
            │                                ├──────────────┤
            │                                │       DM0145 │
            │                                └──────────────┘
            │    03110
            └────┤├──────●──────────────────┌──────────────┐
                        │                    │ SUB(31)      │
                        │                    ├──────────────┤
                        │                    │        #2048 │
                        │                    ├──────────────┤
                        │                    │       DM0140 │
                        │                    ├──────────────┤
                        │                    │       DM0145 │
                        │                    └──────────────┘
                        │  25504(CY)
                        └────┤├──────●──────┌──────────────┐
                                    │        │ MOV(21)      │
                                    │        ├──────────────┤
                                    │        │        #0000 │
                                    │        ├──────────────┤
                                    │        │       DM0145 │
                                    │        └──────────────┘
                                    │        ┌──────────────┐
                                    └────────│ CLC(41)      │
                                             └──────────────┘
```

Adjusts the product/box offset data to be centered about 2048.

```
  03000        03111
───┤├──────●────┤/├─────────────────────────┌──────────────┐
            │                                │ ADD(30)      │
            │                                ├──────────────┤
            │                                │        #2048 │
            │                                ├──────────────┤
            │                                │       DM0150 │
            │                                ├──────────────┤
            │                                │       DM0155 │
            │                                └──────────────┘
            │    03111
            └────┤├─────────────────────────┌──────────────┐
                                             │ SUB(31)      │
                                             ├──────────────┤
                                             │        #2048 │
                                             ├──────────────┤
                                             │       DM0150 │
                                             ├──────────────┤
                                             │       DM0155 │
                                             └──────────────┘
                 25504(CY)
            └────┤├─────────────────────────┌──────────────┐
                                             │ MOV(21)      │
                                             ├──────────────┤
                                             │        #0000 │
                                             ├──────────────┤
                                             │       DM0155 │
                                             └──────────────┘
```

Adjusts the rate of change of the offset data to be centered about 2048.

```
  03000
───┤├──────●─────────────────────────────────┌──────────────┐
            │                                  │ CMP(20)      │
            │                                  ├──────────────┤
            │                                  │        #4095 │
            │                                  ├──────────────┤
            │                                  │       DM0145 │
            │                                  └──────────────┘
            │   25507(<)
            ├────┤├───────────────────────────┌──────────────┐
            │                                  │ MOV(21)      │
            │                                  ├──────────────┤
            │                                  │        #4095 │
            │                                  ├──────────────┤
            │                                  │       DM0145 │
            │                                  └──────────────┘
            │                                  ┌──────────────┐
            ├──────────────────────────────────│ CMP(20)      │
            │                                  ├──────────────┤
            │                                  │        #4095 │
            │                                  ├──────────────┤
            │                                  │       DM0155 │
            │                                  └──────────────┘
            │   25507(<)
            └────┤├───────────────────────────┌──────────────┐
                                               │ MOV(21)      │
                                               ├──────────────┤
                                               │        #4095 │
                                               ├──────────────┤
                                               │       DM0155 │
                                               └──────────────┘
```

Sets the maximum value for input data at 4095.

25315

MOV(21)
#000C
120

Sets the Scaling ON Flags
for Inputs 1 and 2 of the
Analog Input Unit.

MOV(21)
#0002
140

Specifies the Number of Inputs.

MOV(21)
#0170
141

Specifies the First Input
Address (DM 0170).

MOV(21)
#0004
142

Specifies the Number of Outputs.

MOV(21)
#0190
143

Specifies the First Output
Address (DM 0190).

03000

BIN(23)
DM0145
DM0170

Moves the product/box offset
data to the first input address.

BIN(23)
DM0155
DM0171

Moves the rate of change of the offset
data to the second input address.

14403 14404 14405 14407   (14015)

Turns ON the Start Processing Bit
to initialize fuzzy logic processing.

03000

BCD(24)
DM0190
DM0160

Converts the Fuzzy Logic
Unit's output to BCD.

SUB(31)
DM0160
#2048
DM0108

Subtracts 2048 from the
Fuzzy Logic Unit's output to
calculate the control value.

25504(CY)   (03112)

Turned ON if the speed of
the box will be reduced.

CLC(41)

SUB(31)
#0000
DM0108
DM0108

@MOV(21)
#0000
DM0162

```
  03000
───┤├────────┬──────────────────────────────────────┌──────────────┐
             │                                        │ CMP(20)      │
             │                                        ├──────────────┤
             │                                        │ DM0106       │
             │                                        ├──────────────┤
             │                                        │ DM0108       │
             │                                        └──────────────┘
             │   25507(<)
             ├──────┤├─────────┬────────────────────┌──────────────┐
             │                 │                     │ ADD(30)      │
             │                 │                     ├──────────────┤
             │                 │                     │ DM0106       │
             │                 │                     ├──────────────┤
             │                 │                     │ DM0162       │
             │                 │                     ├──────────────┤
             │                 │                     │ DM0162       │
             │                 │                     └──────────────┘
             │                 │                     ┌──────────────┐
             │                 ├─────────────────────│ CMP(20)      │
             │                 │                     ├──────────────┤
             │                 │                     │ DM0162       │
             │                 │                     ├──────────────┤
             │                 │                     │ DM0108       │
             │                 │                     └──────────────┘
             │                 │   25505(>)
             │                 └──────┤├────────────┌──────────────┐
             │                                       │ MOV(21)      │
             │                                       ├──────────────┤
             │                                       │ DM0108       │
             │                                       ├──────────────┤
             │                                       │ DM0162       │
             │                                       └──────────────┘
             │   25507(<)
             └──────┤/├──────────────────────────────┌──────────────┐
                                                      │ MOV(21)      │
                                                      ├──────────────┤
                                                      │ DM0108       │
                                                      ├──────────────┤
                                                      │ DM0162       │
                                                      └──────────────┘
  03000       12508
───┤├────────┬──┤/├─────────┬────────────────────────┌──────────────┐
             │               │                        │ MOV(21)      │
             │               │                        ├──────────────┤
             │               │                        │ 121          │
             │               │                        ├──────────────┤
             │               │                        │ DM0210       │
             │               │                        └──────────────┘
             │   03201       │
             ├──────┤├────────┤                        ┌──────────────┐
             │               │                        │ MOV(21)      │
             │               │                        ├──────────────┤
             │               │                        │ 123          │
             │               │                        ├──────────────┤
             │               │                        │ DM0212       │
             │               │                        └──────────────┘
             │       25503    00001
             └────────┤/├──────┤├────────────────────( 03201 )
  03000  03201  12500  12501  12503  00001
───┤├────┤/├────┤/├────┤/├────┤/├────┤/├─────────────┌──────────────┐
                                                      │ MOV(21)      │
                                                      ├──────────────┤
                                                      │ DM0212       │
                                                      ├──────────────┤
                                                      │ 130          │
                                                      └──────────────┘
  25315
───┤├────────────────────────────────────────────────┌──────────────┐
                                                      │ MOV(21)      │
                                                      ├──────────────┤
                                                      │ #0003        │
                                                      ├──────────────┤
                                                      │ DM0104       │
                                                      └──────────────┘
```

Moves Servodriver A data (converted by the Analog Input Unit) to DM 0210.

Moves Servodriver B data (converted by the Analog Input Unit) to DM 0210.

Enables reading of Analog Input Unit data.

Moves Analog Input Unit data to the Analog Output Unit.

Sets the factor by which the Fuzzy Logic Unit final result will be adjusted.

```
        03000
          ┤├───────●───────────────────────────┌──────────────┐
                                                │ MUL(32)      │
                                                ├──────────────┤    Adjusts the Fuzzy Logic
                                                │      DM0162  │    Unit final result by factor
                                                ├──────────────┤    set above.
                                                │      DM0104  │
                                                ├──────────────┤
                                                │      DM0165  │
                                                └──────────────┘

                          ──────────────────────┌──────────────┐
                                                │ MOV(21)      │
                                                ├──────────────┤
                                                │      DM0210  │
                                                ├──────────────┤
                                                │      DM0185  │
                                                └──────────────┘

                          ──────────────────────┌──────────────┐
                                                │ SRD(75)      │
                                                ├──────────────┤
                                                │      DM0185  │
                                                ├──────────────┤
                                                │      DM0185  │
                                                └──────────────┘

                 03112                          ┌──────────────┐
                  ┤/├──────────────────────────│ ADD(30)      │    Calculates the output to con-
                                                ├──────────────┤    veyor B by adding the Fuzzy
                                                │      DM0210  │    Logic Unit final result to the
                                                ├──────────────┤    conveyor A value.
                                                │      DM0165  │
                                                ├──────────────┤
                                                │      DM0169  │
                                                └──────────────┘

                 03112                          ┌──────────────┐
                  ┤├───────────●───────────────│ SUB(31)      │    Calculates the output to con-
                                                ├──────────────┤    veyor B by subtracting the
                                                │      DM0210  │    Fuzzy Logic Unit final result
                                                ├──────────────┤    from the conveyor A value.
                                                │      DM0165  │
                                                ├──────────────┤
                                                │      DM0169  │
                                                └──────────────┘

                   25504(CY)                    ┌──────────────┐
                     ┤├─────────────────────────│ MOV(21)      │
                                                ├──────────────┤    Ensures that conveyor B
                                                │      DM0185  │    won't stop completely.
                                                ├──────────────┤
                                                │      DM0169  │
                                                └──────────────┘

        03000
          ┤├───────────────────────────────────┌──────────────┐
                                                │ CMP(20)      │
                                                ├──────────────┤
                                                │      DM0185  │
                                                ├──────────────┤
                                                │      DM0169  │
                                                └──────────────┘

                25505(>)                        ┌──────────────┐
                  ┤├──────────────────────────── │ MOV(21)      │
                                                ├──────────────┤
                                                │      DM0185  │
                                                ├──────────────┤
                                                │      DM0169  │
                                                └──────────────┘

                          ──────────────────────┌──────────────┐
                                                │ CMP(20)      │
                                                ├──────────────┤
                                                │       #4095  │
                                                ├──────────────┤
                                                │      DM0169  │
                                                └──────────────┘

                25507(<)                        ┌──────────────┐
                  ┤├──────────────────────────── │ MOV(21)      │
                                                ├──────────────┤    Sets the maximum data input
                                                │       #4095  │    to the Analog Output Unit at
                                                ├──────────────┤    4095.
                                                │      DM0169  │
                                                └──────────────┘

                00001                           ┌──────────────┐
                  ┤├──────────────────────────── │ BIN(23)      │
                                                ├──────────────┤    Transfers the output to con-
                                                │      DM0169  │    veyor B to the Analog Output
                                                ├──────────────┤    Unit.
                                                │         130  │
                                                └──────────────┘

          ●─────────────────────────────────────┌──────────────┐
                                                │ END(01)      │
                                                └──────────────┘
```
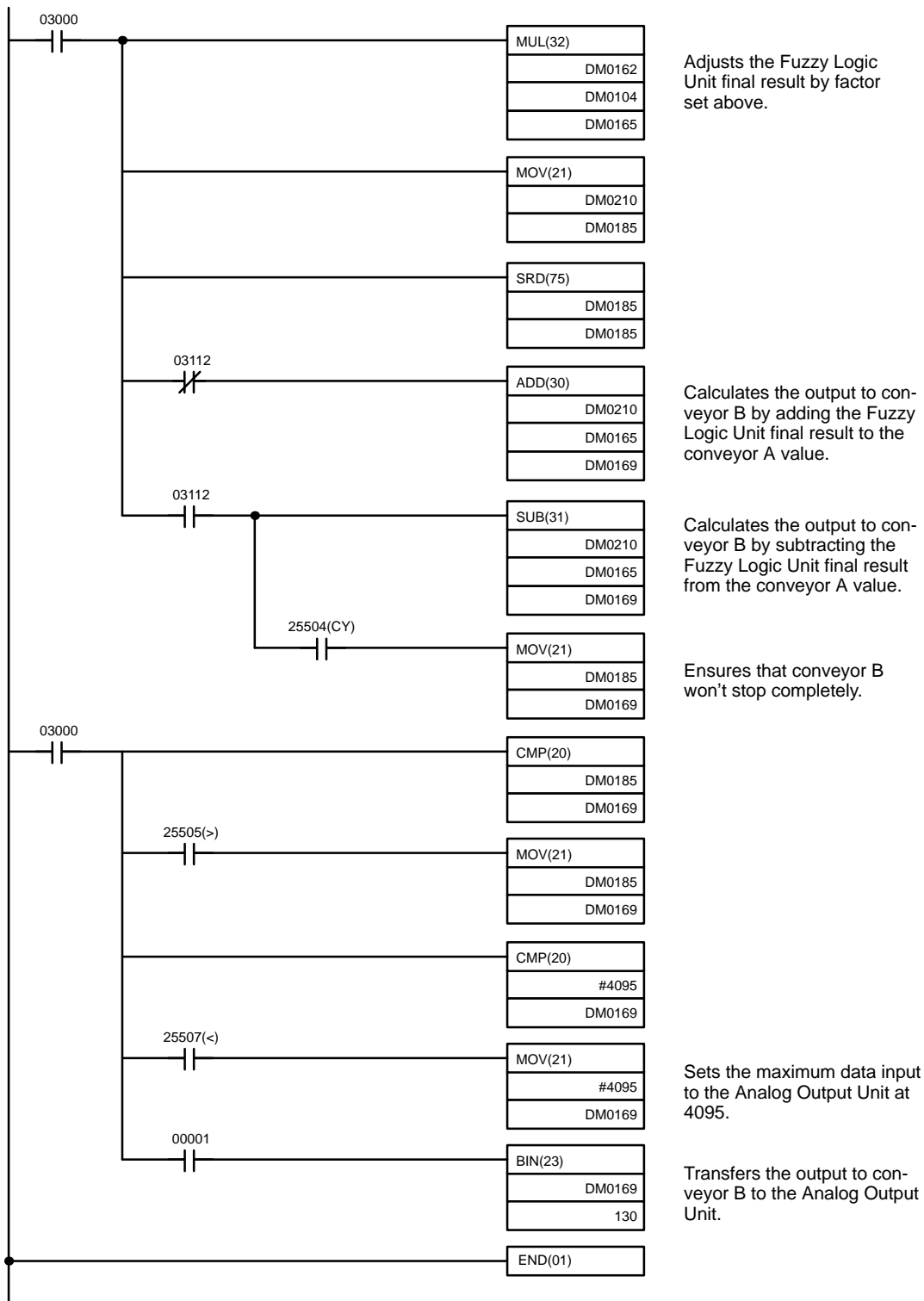
5-3-4 Adjusting I/O Data

If the range of the input data is too narrow, it can be expanded by multiplying the input data by an appropriate factor in the ladder program before it is transferred to the Fuzzy Logic Unit. In the ladder program for the conveyor belt control system, both the product/box offset and the rate of change of the offset were multiplied by scaling factors.

It is possible that the servodriver for the motor on conveyor B will be too sensitive to the output of the Fuzzy Logic Unit, even though the output is relatively smooth.

In such a case, the output can be subdivided before being sent to the servodriver. In the ladder program for the conveyor belt control system, the output was subdivided and then multiplied by a scaling factor. This process produces smooth operation when the output from the Fuzzy Logic Unit is large and good response when the output is small.

The following chart shows the motor reaching its target speed too quickly. To correct the problem, the output from the Fuzzy Logic Unit is reduced when it exceeds a given value, so the motor reaches its target speed more gradually. Reducing the output from the Fuzzy Logic Unit makes the motor operation smoother, but increases the time required for the motor to reach its target speed.
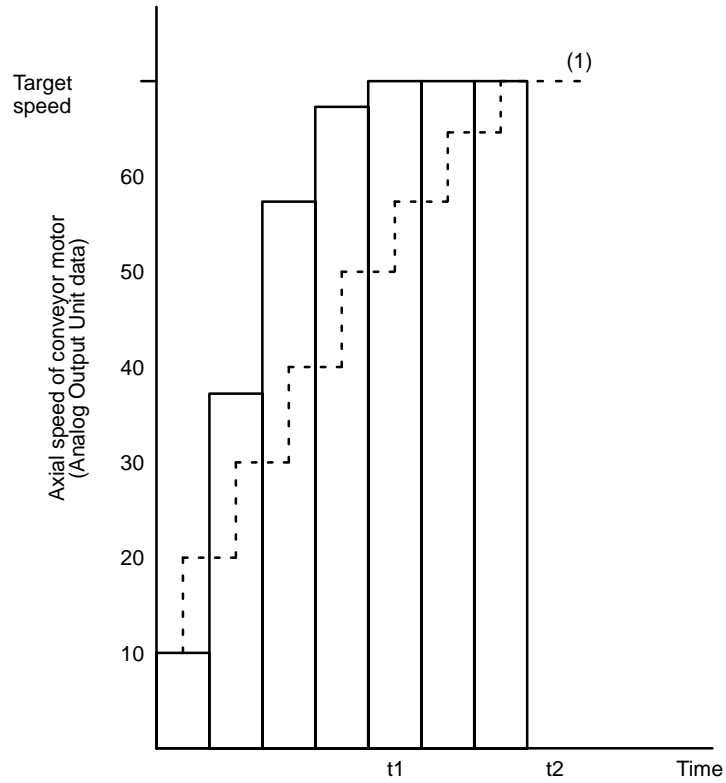
SECTION 6
Maintenance and Troubleshooting

This section provides guidelines for dealing with routine maintenance and errors that might occur while using the Fuzzy Logic Unit.

6-1 Maintenance

The items listed in the following table should be inspected regularly to ensure correct operation.

Area	Item	Correct condition
Ambient conditions	Temperature	0° to 55°C
	Humidity	35% to 85% (no condensation)
	Dust or debris	No dust or debris should be allowed to accumulate on the Unit.
Installation	Unit mountings	Check that the Unit is mounted securely.
	Cable connectors	Check that the cable connectors are tight.
	Connecting cable	Check the connecting cable for damage.

Replacing the Battery

The C200H–BAT09 battery in the Fuzzy Logic Unit has a life expectancy of 5 years at 25°C. The lifetime will be shorter at higher temperatures.

When the battery begins to fail, the BAT ERR indicator will light. Replace the battery as outlined below within 1 week after the BAT ERR indicator first lights.

1, 2, 3...
1. Turn off the power. If the power was off already, turn it on for at least one minute, then turn it off.

2. Remove the battery cover by pressing it and sliding it down.
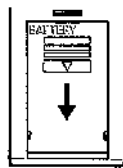
Press and slide the cover down.

Notch

3. Remove the old battery and its connector. Replace the old battery with a new one within 5 minutes.

C200H-BAT09 Battery

4. When installing the new battery, be sure that you press the connector into its socket firmly.

5. Replace the battery cover. Be sure that the catch on the top of the cover goes into the notch in the Unit.

Notch

Battery holder

Battery connector

Catch

Battery cover
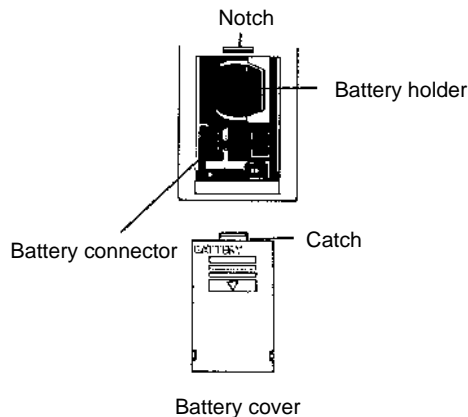
6-2 Troubleshooting

LED Error Indications The table below lists error conditions, their probable cause, and possible remedies.

Indicator(s)	Probable cause	Remedy
No indicators lit.	The PC power supply is OFF.	Connect the PC power supply or check connectors.
	Fuzzy Logic Unit mounting screws are loose.	Tighten the screws.
	The test switch on the setting switch is ON.	Turn OFF the "unused" DIP switch pins.
ERROR indicator is lit.	An error has occurred during the Fuzzy Logic Unit's self-diagnostics.	Replace the Unit.
BAT ERR indicator is lit.	The battery isn't connected.	Check the battery contacts and be sure that they are properly connected.
	Battery voltage has dropped below the minimum.	Replace the battery.
ERROR indicator is flashing.	The rules and membership functions haven't been input or they have been corrupted somehow.	Transfer the rules and membership functions to the Fuzzy Logic Unit.

Replacing the Unit We recommend that users have a backup Unit available to make repairs and minimize down-time if a problem occurs with a Fuzzy Logic Unit. Please observe the following precautions in the event of a problem:

• Always turn the power off when replacing a Fuzzy Logic Unit.

• If a Fuzzy Logic Unit fails, replace it with a new one and immediately verify that the new Unit is working properly.

• When returning a malfunctioning Fuzzy Logic Unit for repair, please attach a detailed description of the problem to the Unit and return it to the sales office nearest you (see listing at the end of this publication).

• If you suspect that a poor connection is the cause of a malfunction, clean the connectors using a clean, soft cloth and industrial-grade alcohol. Remove any lint or threads left from the cloth, and re-mount the Fuzzy Logic Unit.

6-3 Error Processing

Error Messages

The following table lists errors that can occur in the C200H CPU due to Special I/O Units such as the Fuzzy Logic Unit, as well as the probable causes of the errors, their affect on PC operation, and possible remedies.

Error and message	FAL no.	Probable cause	Possible remedy
Waiting for Special I/O Unit CPU WAIT'G	None	A Special I/O Unit has a hardware defect. The PC will not run.	Replace the defective Unit. The defective Unit will be indicated by a "$" displayed in the I/O Table Load operation.
Special I/O Error SPECIAL UNIT ERR	D0	Error has occurred in refresh between a Unit and the CPU. The affected Unit won't operate, but the rest of the system will continue operating.	Determine the affected Unit from the Special I/O Unit Error Flags (ar 0000 to AR 0009), correct the cause of the error, and toggle the Unit's Restart Bit to restart the Unit. Replace the Unit if it cannot be restarted.
Too many Units I/O UNIT OVER	E1	Two or more Special I/O Units have been assigned the same unit number. The PC will not run.	Check the unit numbers with the I/O Table Load operation, and eliminate any duplication.

Special I/O Unit Error Flags and Restart Bits

The following table lists Error Flags and Restart Bits for Special I/O Units. If a Special I/O or Too Many Units error occurs, the corresponding bits in AR 00 will be turned ON. Toggling (turning OFF, ON, and OFF again) a Restart Bit will restart the corresponding Unit.

Word	Bit(s)	Function
AR 00	00 to 09	Error Flags for Special I/O Units 0 to 9 (also function as Error Flags for PC Link Units)
AR 01	00 to 09	Restart Bits for Special I/O Units 0 to 9 (also function as Restart Bits for PC Link Units)
SR 254	15	Special Unit Error Flag. (also functions as the error flag for PC Link, Host Link, Remote I/O Master, SYSMAC NET Link, and SYSMAC LINK Units) This flag will be turned ON in the event of a Special I/O error or Too Many Units error, which were described in the preceding table.

Appendix A
Standard Models

Name	Model number	Notes
Fuzzy Logic Unit	C200H-FZ001	---
Battery	C200H-BAT09	One Battery is included with the Fuzzy Logic Unit.
Fuzzy Support Software (FSS)	C500-SU981-E	The FSS is available on either 3.5" or 5.25" floppy disks.

Appendix B
Specifications

Item		Specification
Fuzzy logic processor	I/O capacity	8 inputs and 4 outputs max.
	Rule format	8 condition and 2 conclusion parts max.
	Number of rules	128 rules max.
	Logic process	Forward logic
	Logic rule	MAX–MIN logical product
	Number of labels	7 max.
	Final calculation	Each output can be set independently for calculation by center of gravity, leftmost maximum, or rightmost maximum method.
	Treatment of rule with no grade	Can be set from the programming computer to retain the previous value or take a preset value.
Membership functions	Condition	The function is defined by up to 4 endpoints. Grade (height) at the endpoints must be either 0 or 1 (0 or 4095). Resolution is 4095 by 4095.
	Conclusion	Only the horizontal position is set. Height is fixed at 4095. Resolution is 4095 by 4095.
I/O words	Inputs	Each input is allocated one word. 12 bits of the word are used, so the range is 000 to FFF (0 to 4095 decimal).
	Outputs	Each output is allocated one word. 12 bits of the word are used, so the range is 000 to FFF (0 to 4095 decimal).
Peripheral device communications	Communications	Half duplex
	Synchronization	Start-stop synchronization
	Baud rate	300, 600, 1200, 2400, 4800, 9600, or 19200 (set on DIP switch 2)
	Transmission distance	15 m max.
	Interface	RS–232C port
	Communications protocol	Special procedure (1:N)
Processing time		6 ms max. for the Fuzzy Logic Unit, 3 to 4 times the scan time for the system.
Data retained in a power interruption		Rules and membership functions are retained.
Battery	Life expectancy	5 years at 25°C. Shorter at higher temperatures.
	Replacement	Replace within one week after the BAT ERR LED lights. Data is retained for 5 minutes max. without the battery.
Self-diagnostics	Program check	A "memory error" will be generated if an error occurs during the program check.
Power supply		5 VDC, 0.3 A max. (Supplied from the Backplane.)
Dimensions (mm)		$35 \times 130 \times 117$ (W x H x D)
Weight		400 g

Glossary

condition part
The "if" clauses of a rule. A single rule can have up to 8 condition parts in the form "IF X_1 AND X_2 ... AND X_8."

center of gravity method
A method of defuzzification. The center of gravity of the fuzzy outputs (along the x-axis) is calculated as the final result of fuzzy logic processing.

conclusion part
The "then" clauses of a rule. A single rule can have up to 2 conclusion parts in the form "THEN Y_1 AND Y_2."

defuzzification
Defuzzification is the process that combines the fuzzy outputs to a single result that is output from the Fuzzy Logic Unit. There are two methods of defuzzification available in the C200H-FZ001: the center of gravity method and maximum value method.

fuzzy output
After all of the rule grades are calculated, the grade for each label is calculated. The grade for a label is called its fuzzy output. The fuzzy output is the maximum rule grade for that label. The maximum value is taken because the rules are linked by logical ORs.

fuzzy variable
A variable such as distance, temperature, or pressure, that is measured by the PC's sensors. The measured values of the fuzzy variables are input to the Unit for fuzzy logic processing.

grade
A number between 0 and 1 that indicates how well the given value of the fuzzy variable satisfies a label. A grade of 0 means the label is not satisfied at all, and a grade of 1 means the label is perfectly satisfied at the given value of the fuzzy variable.

label
Labels are used to describe the state of a fuzzy variable. Labels for distance might be "close," "OK," and "far." Labels for temperature might be "cold," "cool," "warm," and "hot." Labels for pressure might be "light," "moderate," and "hard." The C200H-FZ001 can have up to 7 labels for each fuzzy variable.

knowledge base
The rules and membership functions.

maximum value method
A method of defuzzification. The position (along the x-axis) of the maximum fuzzy output is used as the final result of fuzzy logic processing. Either the leftmost or rightmost position will be used if there are two or more fuzzy outputs with the same maximum value.

membership function
Defines the relationship between the grade and the fuzzy variable for each label. The most accurate shape for a membership function is usually a bell curve, but triangles and trapezoids are used most often to simplify and speed up processing.

rule
An if/then statement that expresses a relationship between inputs and outputs. The C200H-FZ001 can have up to 128 rules.

rule grade
Membership functions assign grades to the labels of the condition parts of a rule according to the present conditions. The rule grade is the minimum of these grades.

Index

A

applications, precautions, ix

B

battery
  life expectancy, 48
  replacement, 48

C

center of gravity method. *See* defuzzification
components, location, 14
conclusion part
  definition, 4
  processing, 7
condition part
  combinations, 5
  definition, 4
  processing, 6
configuration, system, 2
connecting cable, 16
control system
  hybrid, 10
  pure fuzzy logic, 10

D

defuzzification
  center of gravity method, 8
  definition, 8
  maximum value method, 8
dimensions, 17
DIP switches, setting, 15

E

error
  CPU waiting, 50
  I/O unit over, 50
  special unit, 50
error codes, 22
Error Flag, output, 23
error indications, 49
error messages, 50
error processing, 50
examples, application, 31

F

First Input Word, setting, 21
First Output Word, setting, 21
flags, 21
  Battery Error Flag, 21

Fuzzy Error Flag, 21
Input Range Exceeded Flag, 23
Memory Error Flag, 21
No Corresponding Output Flag, 23
Processing Enabled Flag, 21
Results Output Flag, 21
Setting Error Flag, 21
Special I/O Unit error, 50
Too Few Input Words Flag, 23
Too Many Input Words Flag, 23
Undefined Output Flag, 23
fuzzy logic
  introduction, 3
  processing, 6
  settings, 20
    example, 26
  starting, 20
    example, 28
fuzzy output
  calculating, 7
  definition, 6
fuzzy set, definition, 3
fuzzy variable, definition, 4

G

grade
  definition, 3
  labels, 6
  rules, 5

I

I/O data, format, 22
indicators
  error indications, 49
  functions, 14
  location, 14
input data, transfer, example, 28
input range, of rules, 10
inputs
  data format, 22
  setting the number, 20
installation
  acceptable locations, 48
  PC mounting slots, 2
  precautions, ix

K

knowledge base
  definition, 9
  developing, 9 , 12

L

label, definition, 5