
# GE Fanuc Automation

*Programmable Control Products*

## *VersaPro™*
## *Programming Software*
### *User's Guide*
### *프로그램 사용자 매뉴얼*

*GFK-1670A* *July 2000*

# *Wanrning, Caution 및 Note*

## Warning

Warning 표시는 인명에 손상을 입힐 수 있는 위험한 전압, 전류, 온도 및 기타 조건들이 이 장비에 존재하거나 장비 사용과 관련될 수 있다는 것을 강조하기 위한 것입니다.

부주의로 인명 손상이나 장비 손상을 일으킬 수 있는 상황에 Warning 표시를 사용됩니다.

## Caution

Caution 표시는 조심하지 않을 때, 장비에 손상을 줄지도 모르는 곳에 사용됩니다.

## Note

Note 표시는 장비를 이해하고 작동하는데 매우 중요한 정보의 주의를 환기시키기 위한 것입니다.

이 도큐먼트는 이 매뉴얼이 발행된 시점에서 사용할 수 있는 정보를 기반으로 하고 있습니다. 정확성을 기하기 위해 많은 노력을 했지만 여기에 있는 정보는 하드웨어와 소프트웨어의 모든 세부사항이나 변화를 다루고 있다는 것은 아니며, 설치, 운용 및 유지보수와 관련된 모든 가능성 있는 우발적 사고에 대비하는 것을 의미하는 것은 아닙니다. 이 매뉴얼에 서술된 내용들이 모든 하드웨어 및 소프트웨어 시스템에 적용되는 것은 아닙니다. GE Fanuc Automation 은 이 매뉴얼이 만들어진 후의 변화에 대해 이 도큐먼트의 변경사항을 의무적으로 공지하지는 않습니다.

GE Fanuc Automation 은 이 글에 들어 있는 정보에 대해 설명, 보증, 표현, 의미 또는 법령과 정확성, 완전성, 충분성 및 효용성에 대한 책임이 없다는 것을 밝힙니다. 시장성에 대한 보증이나 목적에 대한 적합성은 적용되지 않습니다.

다음은 GE Fanuc Automation North America, Inc.의 트레이드 마크입니다.

| | | | |
|---|---|---|---|
| Alarm Master | Field Control | Modelmaster | Series 90 |
| CIMPLICITY | GEnet | Motion Mate | Series One |
| CIMPLICITY Control | Genius | PowerMotion | Series Six |
| CIMPLICITY PowerTRAC | Genius PowerTRAC | ProLoop | Series Three |
| CIMPLICITY 90-ADS | Helpmate | PROMACRO | VuMaster |
| CIMSTAR | Logicmaster | Series Five | Workmaster |

이 매뉴얼은 VersaPro™ programming software, Version 1.10 을 이용하여 Series 90™-30 PLC, VersaMax™ PLC, VersaMax Nano/Micro PLCs 및 Series 90 Micro PLC 를 프로그램하는데 사용되는 특성을 기술하였습니다.

# 개정된 내용

이 매뉴얼과 제품 온라인 도움말 기능은 버전 1.10 에 의해 제공되는 새로운 특성을 반영하기 위해 업데이트했습니다. 새로운 특성은 다음과 같습니다.:

- 새로 추가된 특성(Chapter 2)
- Information Window 은 새로운 탭 컨트롤 기능을 가지고 있으며, Find and Cross Reference information 를 디스플레이합니다.(Page 2-12)
- 프린트 리포트 유틸리티를 위한 새로운 디자인.( Chapter 2)
- 새로 추가된 Block 특성(Chapter 3)
- LD Editor 기능 향상 (Chapter 4 )
- IL Editor 기능 향상(Chapter 5)
- 다양한 Declaration Table 에서의 새로운 특성 (Chapter 6)
- 하드웨어 구성은 새로운 Parameter Editor 의 특성을 갖고 있으며, Series 90 Micro, VersaMax Nano/Micro, Motion Mate DSM314, Series 90-30 Release 10 특성들을 지원합니다( Chapter 7).
- 툴 추가 기능, Options Dialog(Chapter 4, 7, 8).
- Reference View Table 의 주요 재디자인. Variable View Table 향상(Chapter 8).
- Motion Mate DSM314 motion controller 용 Local Logic 과 Motion program blocks 생성 능력. Motion program editor, Local Logic program editor 와 Local Logic Variable View Table 을 포함하고 있습니다(Chapter 9) .
- 재설계된 PLC 상태 정보 다이얼로그 박스. 새로운 Motion/LL tab 포함(Chapter 10).
- Toolbars 에 새로운 옵션 추가: View Toolbar 에 Local Logic Variable Table button 추가. 알파벳 순서에 따라 Function Toolbar 분류(부록 B).

# 목 차

# 관련 매뉴얼

| | |
|---|---|
| GFK-1742 | ***Motion Mate™ DSM314 for Series 90™-30 PLCs User's Manual*** |
| GFK-1645 | ***VersaMax™ Micro PLCs User's Manual*** |
| GFK-1504 | ***VersaMax™ Modules, Power Supplies and Carriers*** |
| GFK-1503 | ***VersaMax™ PLC User's Manual*** |
| GFK-1411 | ***Series 90™-30 System Manual*** |
| GFK-1065 | ***Series 90™-30 Micro PLC User's Manual*** |
| GFK-0356 | ***Series 90™-30 Programmable Controller Installation Manual*** |

# *Contents*

# *Contents*

# *Contents*

# *Contents*

# *Contents*

# *Contents*

Chapter 1

# 소개

## VersaPro 에 오신 것을 환영합니다

Series 90™-30 과 VersaMax™ PLC 용 GE Fanuc 의 Windows®-기반 프로그래밍 소프트웨어인 VersaPro™에 오신 것을 환영합니다. VersaPro 는 Windows 95, Windows 98 과 Windows® NT 4.0 에서 설치 운용할 수 있도록 설계되었습니다.

VersaPro 의 역할:

- 폴더에서 PLC logic 과 관련된 정보
- PLC 하드웨어 구성
- 변수 생성 및 편집
- Ladder 또는 Instruction List Logic 실행의 생성 편집 및 모니터
- Motion 과 Local Logic program 생성

### 목차

이 장은 다음과 같은 정보를 제공합니다:

- VersaPro 프로그래밍 소프트웨어 콤포넌트 개요
- Workbench Window
- Workbench Option 구성(에디터 및 타깃 하드웨어 설정)

1

# VersaPro 이용하기 – 개요

VersaPro 는 PLC 하드웨어를 구성하고 PLC 로직을 생성 편집하며, PLC 프로그램의 실행을 모니터할 수 있도록 Single Programming Interface 를 제공합니다.

| 콤포넌트 | 설명 |
|---|---|
| Folder Browser | VersaPro 는 편리한 소프트웨어 패키지입니다. 기본 언어와 창과 에디터 표시기를 만들 수 있습니다. |
| Hardware Configuration | VersaPro 의 기본 하드웨어 플랫폼은 VersaMax PLC 입니다. Series 90-30 과 VersaMax PLC 에 대한 프로그램과 하드웨어 구성을 생성하기 위해 하드웨어 플랫폼을 변경시킬 수 있습니다 |
| Logic Editors | PLC 로직을 생성하기 위해 다음 두 에디터가 있습니다: Instruction List Editor 와 Ladder Diagram Editor. IL 에서 RLD 로 프로그램과 블록을 변환시킬 수 있습니다. |
| Motion Editors | Motion Editor 와 Local Logic Editor 는 Motion Mate DSM314 모션 컨트롤 모듈용 Local Logic 과 Motion Program 을 생성하는데 제공됩니다. |
| Syntax Checking | VersaPro 는 프로그램이 PLC 에 저장하기 전에 문맥적으로 올바른지 점검하는 툴을 제공합니다. |
| PLC Communications | VersaPro 은 Series 90-30 PLC 와 Serial/Ethernet Connection 을 제공하고 VersaMax PLC 와 Serial connection 을 제공합니다 |
| Online Monitoring | Online Monitoring 과 Control 은 Editor, Reference View Table 과 Variable View Table 에서 지원됩니다. |
| Fault System | Fault System 은 PLC 에서 PLC 와 I/O 오류 테이블을 볼 수 있도록 빠른 연결을 제공합니다. |
| Information Window | 정보 창은 VersaPro 에서 수행되는 동작 결과를 보여줍니다. |

다음은 VersaPro 에서 사용되는 약어입니다:

| 약어 | Component |
|---|---|
| IL | Instruction List |
| RLD | Relay Ladder Diagram |
| RVT | Reference View Table |
| VDT | Variable Declaration Table |
| VVT | Variable View Table |
| LL | Local Logic |
| LLVT | Local Logic Variable Table |

# Workbench Window

VersaPro 를 처음 시작할 때, Workbench 가 나타납니다. Workbench 에서는 다음과 같은 업무를 수행합니다.

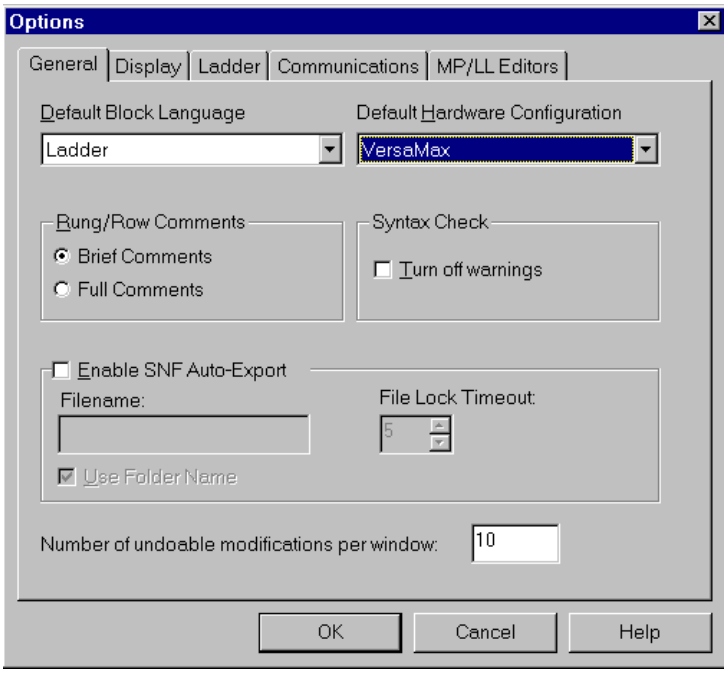
- 폴더 생성과 열기
- VersaPro 폴더 백업과 복구
- 통신 파라미터 설치, PLC 와의 연결 및 Fault Table 보기
- Workbench Option 만들기

## Workbench Option 설정

VersaPro 는 Editor 와 Window Option 을 만들어 줍니다. 새로운 폴더를 생성하기 전에 기본 언어와 하드웨어 구성을 확인하는 것이 중요합니다. 다른 옵션들은 폴더를 열거나 폴더를 연 후에 만들 수 있습니다.

### 일반 옵션

VersaPro 의 기본 하드웨어 구성은 VersaMax PLC 입니다. Default Block Language 는 Ladder Diagram 입니다. 타깃 디바이스에 대한 프로그램과 하드웨어 구성을 생성할 수 있으며 기본 구성을 변경하고, 디스플레이와 편집 옵션을 만들기 위해서는 도구 메뉴를 선택하여 옵션을 선정하십시오. 옵션 다이얼로그 박스가 나타날 것입니다.




- **Block Language 설정:** 기본 언어는 Ladder Diagram 입니다. Instruction List 에서 Default Block Language 를 변경시키려면, General Tab 을 선택하십시오. Default Block Language 를 클릭하여 Instruction List 를 선택하십시오.

- **Hardware Configuration Default 변경:** VersaPro 의 기본 하드웨어 플랫폼은 VersaMax PLC 입니다. 모든 Series 90-30, VersaMax, VersaMax Nano/Micro 와 Series 90 Micro PLC 용 프로그램과 하드웨어 구성을 생성할 수 있습니다. Default Hardware Configuration 을 변경하기 위해 General tab 을 선택하고 Default Hardware Configuration 부분을 설치하십시오.

- **코멘트 설정:** LD 와 IL Editor 에서, 코멘트는 완전(Full) 또는 요약(Brief) 모드를 볼 수 있도록 구성할 수 있습니다. 기본은 Brief Mode 에서 코멘트를 볼 수 있습니다. 기본값을 변경하기 위해 General Tab 을 선택하여 Full Comments 버튼을 체크하십시오.

- **Syntax Check (Turn off Warnings) 설정:** 기본값에 의해 VersaPro 는 Logic 상에서 Syntax Check 를 수행합니다. – 만약 Logic 이 문맥상으로 올바르지 않다면 Warning 이 정보 창에 나타날 것입니다. Warning 표시를 감추기 위해 General tab 을 선택하여 Turn Off Warnings box 를 체크하십시오.

- **Enable SNF Auto-Export.** 이 옵션은 변수를 편집하거나 생성하고 VDT 에 저장할 때, 변수가 SNF 포맷에서 자동적으로 추출됩니다. 단지 선택된 EXT Checkbox 와 함께 VDT 에 있는 변수들은 자동 추출될 것입니다.

  .SNF 파일은 자동적으로 폴더로 추출될 것이고, 동일한 이름을 가진 기존 .SNF 파일은 덮어쓰기가 될 것입니다.

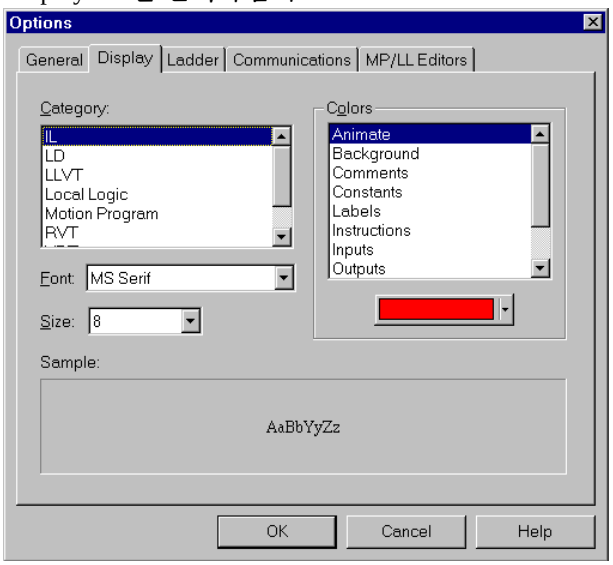
- **실행취소 수 설정:** VersaPro 는 편집하는 동안 사용할 수 있는 Undo/Redo 운용의 수를 구성하도록 해줍니다. 기본 값은 10 입니다. 유효 범위는 1 ~ 100 입니다.
- **파일 이름**– 폴더 이름을 선택하지 않고 사용하면 파일 이름을 요구할 것입니다. 이 이름은 VersaPro 폴더 이름처럼 동일한 규칙에 따라 이름이 지어집니다.
- **폴더 이름 사용**(디폴트) – 이 박스가 체크되면, 추출된 파일 이름은 *foldername.snf* 가 될 것입니다.
- **파일 잠금 해제** – *.lck (lock) 파일은 *.snf 와 동시에 발생됩니다. *.lck 파일은 사용 중에 *.snf 파일을 읽고, 쓰지 못하도록 하기 위해 사용됩니다. VersaPro 는 이 필드에 명시된 시간동안 *.lck 파일과 접근하려고 시도할 것입니다. 허용값은 1 ~ 60 초입니다.

## Display Option 설정

VersaPro 는 IL, LD, Local Logic 과 Motion Editor, VDT, Variable View, Reference View 및 Local Logic Variable Table 에서 별도로 텍스트 폰트, 크기, 색깔을 설정할 수 있습니다.

Display Option 을 설정하기 위해 다음과 같은 단계를 수행합니다:

1. 도구 메뉴나 LD 또는 IL Editor Context Sensitive 메뉴에서 옵션을 선정하십시오(CSM).
2. Options Dialog Box 가 나타납니다.
3. Display tab 을 선택하십시오.




4. Category List 에서 구성하기 원하는 옵션을 표시하는 에디터와 테이블을 선택하십시오.
5. Colors List 는 선택된 에디터나 테이블을 이용해 구성할 수 있는 항목을 보여줍니다.

6. 선택된 에디터나 테이블의 각각의 구성 영역에서 사용할 색깔을 선택하십시오.구성 영역의 리스트는 구성중인 창과는 다릅니다.

**Note:** 색과 백그라운드의 결합 중 어떤 경우에는 읽을 때와 다를 수 있습니다.

7. 선택된 에디터와 테이블에서 텍스트를 보기 위해 사용할 폰트 이름과 크기를 선택하시고 Ok 를 클릭하십시오.

**Note:** 컬럼과 행 앞에 사용된 폰트는 변경시킬 수 없습니다.

선택된 색깔은 선택된 에디터와 테이블의 선택된 영역에 적용됩니다.
선택된 폰트는 선택된 에디터와 테이블에 있는 모든 텍스트에 적용됩니다.
만약 선택된 에디터와 테이블을 볼 수 있다면 변경된 색깔, 폰트와 크기를 같이 볼 수 있습니다.

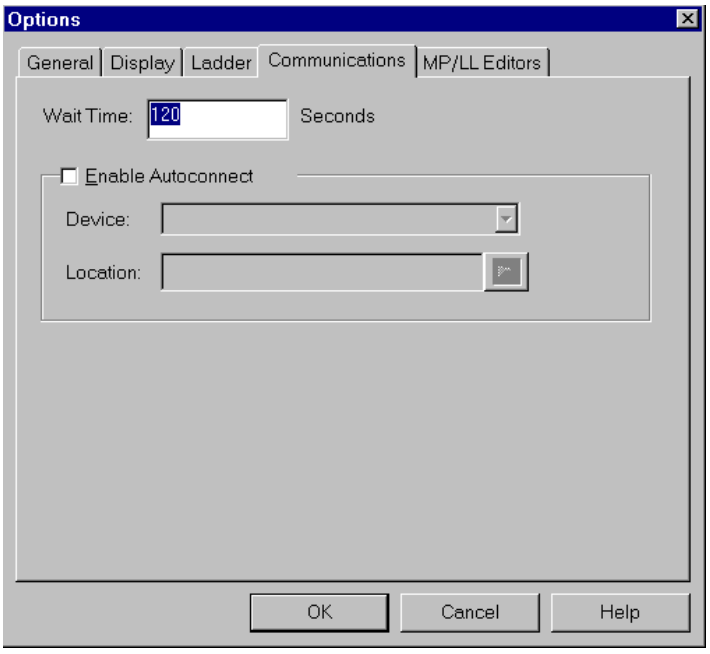
**Note:** 선택된 폰트는 프로그램을 인쇄한 후에 사용됩니다.

## Autoconnect Option 설정

가동하자마자 즉시 특수 디바이스와 연결하고 그 PLC 와 관련된 폴더를 열 수 있도록 VersaPro 를 구성할 수 있습니다

**Autoconnect 구성하기**

1. 도구 메뉴나 Editor CSM 에서 옵션을 선정하십시오.
2. Options dialog box 가 나타납니다.
3. Communications tab 을 선택하십시오.



4. Enable Autoconnect Checkbox를 클릭하십시오.
5. 디바이스 항목에서, 가동하자마자 자동적으로 연결하고자 하는 디바이스를 선택하십시오.
6. Location 항목에서 이 PLC 와 관련된 폴더의 위치를 입력하거나, Folder button  에 대한 브라우즈를 클릭하고, Folder dialog box에서 브라우즈로부터 위치를 선정하십시오.
7. OK 를 클릭하십시오.

**Note:** Autoconnect 기능을 없애려면, Options dialog 에서 Communications tab 을 선택하고 Enable Autoconnect button 에 체크를 하지 마십시오.

## Ladder Option 설정

VersaPro 는 LD Editor Symbol 과 LD Editor Grid Cell 의 폭에 대해 나타난 정보의 타입을 명시할 수 있습니다. 또한 셀에서 디스플레이 크기 정보를 적절하게 확대/축소하기 위해 ZoomIn 또는 Out 할 수 있습니다.

**LD Editor Symbol 에 대해 나타난 정보 타입 명시하기:**

1. 도구 메뉴나 LD Editor CSM 에서 옵션을 선정하십시오.Options dialog box 가 나타납니다.
2. Ladder tab 을 선택합니다.
3. Variable Name, Reference Address 와 Description 을 볼 것인지 감출 것인지 선정합니다.체크된 각 항목들은 LD Editor 에 나타납니다.
4. Variable Name 과 Description 들이 각각에 대해 Brief 또는 Full 을 선택하여 부분적으로나 아니면 가득 차게 디스플레이할 것인지 선정합니다.
5. 가변 설명과 참조 어드레스는 현 셀 폭에 따라 각 셀의 하단에 나타납니다. 4 줄을 초과한 텍스트는 "…"으로 대체됩니다

**LD Editor Grid Cell Width 설정하기:**

VersaPro 는 보다 많은 텍스트를 볼 수 있도록 LD Editor Grid Cell 들의 폭을 변경할 수 있습니다. 셀의 폭만 변경할 수 있습니다: 셀 높이와 폰트 크기는 변경되지 않습니다.

1. 도구 메뉴나 LD Editor CSM 에서 옵션을 선정하십시오.Options dialog box 가 나타납니다.
2. Ladder tab 을 선택합니다.
3. Grid Cell 폭 내에서 Grid Cell 크기를 증가 시키기 위해 슬라이더를 오른쪽으로 드래그하거나 줄이기 위해 왼쪽으로 드래그합니다.
4. OK 를 클릭하십시오. LD Editor Grid 크기가 변할 것입니다.

## Motion Program/Local Logic Editor Option 설정

1. 도구 메뉴나 Editor CSM 에서 옵션을 선정하십시오 Options dialog box 가 나타납니다.
2. SMP/LL Editors Tab 를 선택하십시오.

| | |
|---|---|
| Tab Size | 한 Tab 과 같은 스페이스 수를 설정하십시오. 유효값은 1 ~ 64 입니다. |
| Insert Spaces | Tab 들을 Tab Size 에 명시된 스페이스 수로 변환시키십시오. |
| Keep Tabs | Tab Character 와 같이 Tab 들을 저장하십시오. |
| None | 자동 코드 삽입은 안됩니다. |
| Default | Tab Size 를 이용해 코드를 삽입하십시오. 삽입은 Scope Word 에 기반합니다. |
| Use Previous Line | 같은 라인에 직접 현재 라인을 삽입시킵니다. |

3. Tab 과 Auto-indent Option 을 선택하십시오.
4. OK 를 클릭하십시오.

Chapter 2

# 폴더로 작업하기

폴더는 Series 90-30, VersaMax, VersaMax Nano/Micro 와 Series 90 Micro PLC 를 프로그램, 구성 및 모니터 하는데 필요한 모든 콤포넌트들을 포함하고 있습니다. 콤포넌트들은 Hardware Configuration, Variable Declaration Table, View Table 및 프로그램 블록들을 포함하고 있습니다. 이 폴더에 유저 도큐먼트와 스프레드시트와 같은 다른 윈도우 콤포넌트를 삽입시킬 수 있습니다.

VersaPro 는 한 번에 한 폴더를 열 수 있습니다. 그러나 PC 에서 VersaPro 를 동시에 여러 번 호출할 수 있습니다.

## 목차

이 장은 VersaPro 소프트웨어를 이용하여 폴더를 생성하고 관리하는 방법에 대해 서술하고 있습니다.

2

# *VersaPro 폴더 생성 및 열기*

## 새로운 폴더 생성

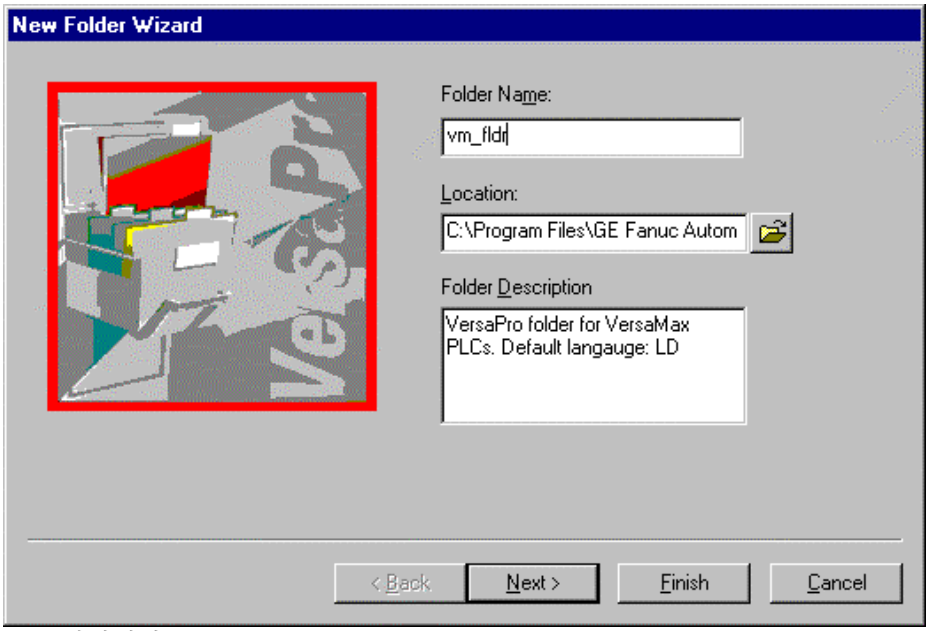
VersaPro 는 새로운 폴더를 생성하는데 여러 옵션을 제공합니다. 빈 폴더를 생성하거나 기존 VersaPro 폴더를 기반으로 해서 새로운 폴더를 생성할 수 있습니다. 또한 VersaPro 는 Logicmaster 90(Series 90-30 또는 Series 90 Micro 폴더) 나 Control(Series 90-30 또는 VersaMax 폴더)에서 컨텐츠를 불러냄으로써 폴더를 생성할 수 있는 옵션을 제공합니다.

### 새롭고 빈 폴더 생성하기:

새로운 폴더의 기본 언어는 Ladder Diagram 입니다. Default Hardware Configuration 은 VersaMax 입니다. 새로운 폴더를 위해 기본 언어나 Target Hardware 를 변경하기를 원한다면, 도구 메뉴를 선택하여 옵션을 선정하십시오. Options dialog box에서 설정을 변경하십시오

VersaPro workbench 에서 새롭고 빈 폴더 생성하기:

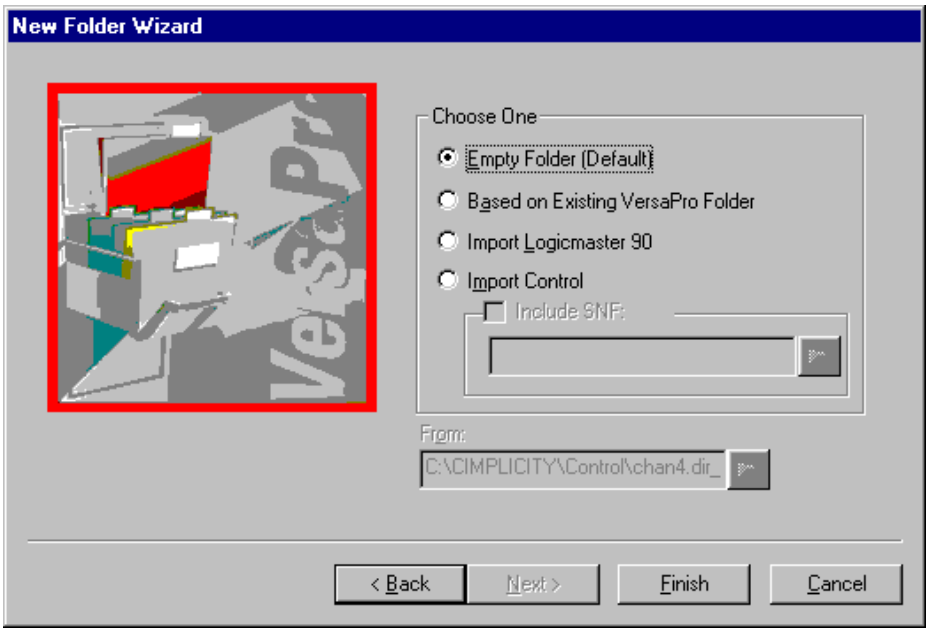
1. New Folder button  을 클릭하고 파일 메뉴에서 New Folder 를 선정하거나 Ctrl+N 을 누르십시오. New Folder Wizard 가 나타날




것입니다.

- Folder Name field 에 새로운 폴더 이름을 입력하십시오. 폴더 이름이 필요합니다(페이지 2-10 에 있는 “폴더 이름과 닉네임” 참조.)
- 기본값으로부터 다른 위치에 새로운 폴더를 저장하기 원한다면 Location field 맞은 편에 있는 브라우즈 버튼을 클릭하십시오.
- Folder Description field 에서 Folder Description 을 입력하십시오. 64 글자까지 입력할 수 있습니다. 이 부분은 옵션입니다.

2. Next button 을 클릭하십시오. New Folder Wizard 의 다음 화면이 나타납니다.




3. Empty Folder (기본) button 은 자동으로 선택됩니다. Finish 를 클릭하십시오.

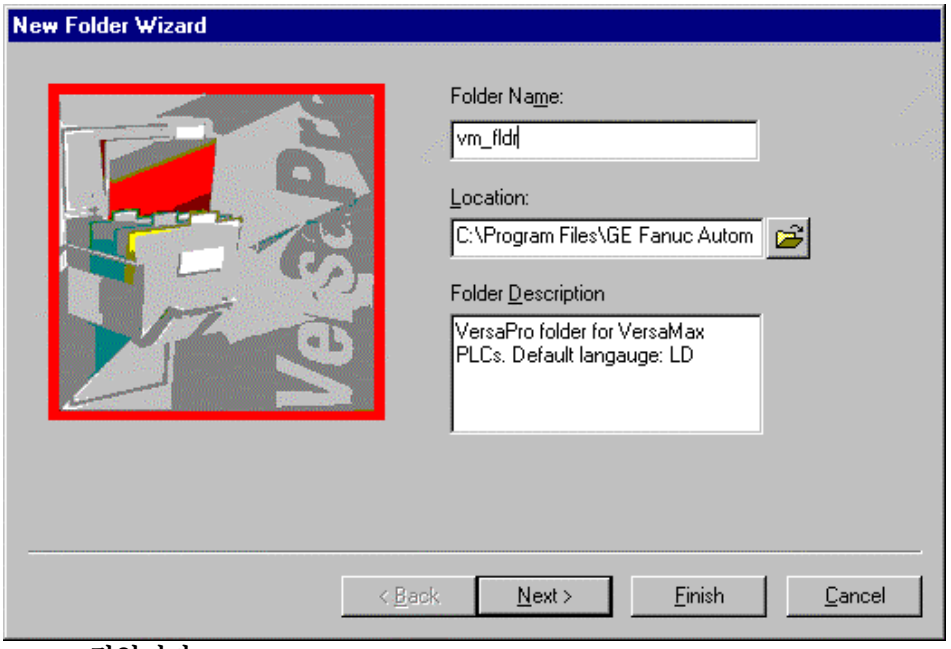
새로운 폴더가 디스플레이 되면서 Folder Browser 가 나타납니다. _MAIN block 은 Default Language 의 Editor 에 나타날 것입니다.

## 기존 VersaPro 폴더를 기반으로 새로운 폴더 생성하기:

새로운 폴더를 위한 기본 언어는 Ladder Diagram 입니다. Default Hardware Configuration 는 VersaMax 입니다. 만약 새로운 폴더에 대한 기본 언어나 Target Hardware 를 변경하기 원한다면, 도구 메뉴를 선택하고 옵션을 선정하십시오. Options dialog box에 있는 설정이 변경됩니다.

기존 VersaPro 폴더를 기반으로 새로운 폴더를 생성하기 위해서는 VersaPro 를 가동하고 다음 단계를 수행하십시오:

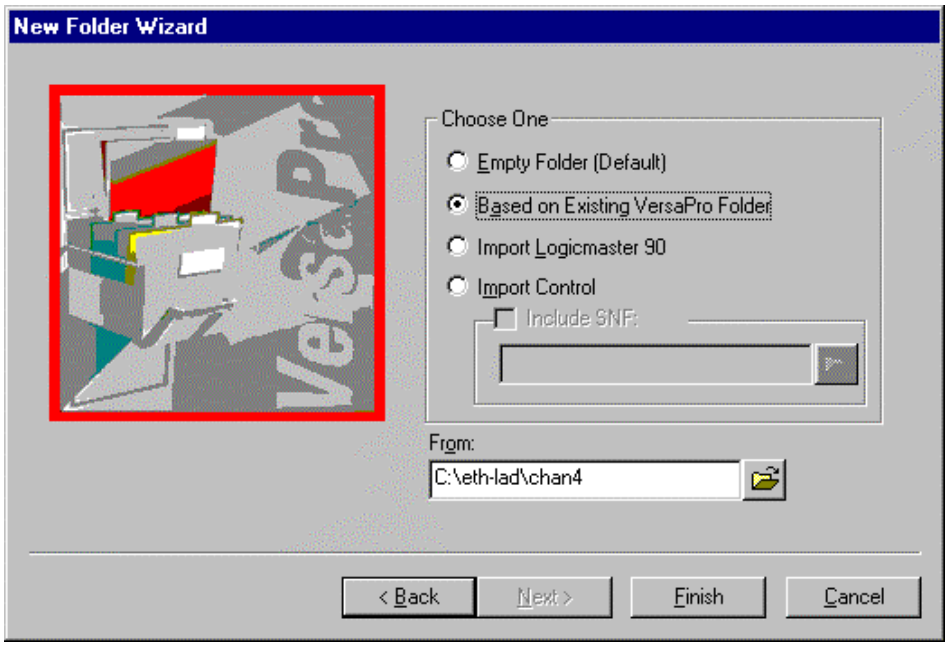
1. New Folder button  을 클릭하고, 파일 메뉴에서 New Folder 를 선정하거나 Ctrl+N 을 누르십시오. New Folder Wizard 가 나타날




것입니다.

- Folder Name field 에 새로운 폴더 이름을 입력하십시오. 폴더 이름이 필요합니다.
- 기본값으로부터 다른 위치에 새로운 폴더를 저장하기 원한다면 Location field 맞은 편에 있는 브라우즈 버튼을 클릭하십시오
- Folder Description field 에서 Folder Description 을 입력하십시오. 64 글자까지 입력할 수 있습니다. 이 부분은 옵션입니다.

2. Next button 을 클릭하십시오. New Folder Wizard 의 다음 화면이 나타납니다.




3. Based on Existing VersaPro Folder button 을 클릭한 다음 Browse button 을 클릭하십시오. 나타난 Browse dialog box 에서 새로운 폴더를 베이스로 하기 원하는 VersaPro 폴더에 위치하십시오.
4. Finish 에 클릭하십시오. 기존 폴더의 구조와 컨텐츠(블록, VDT, etc)는 새로운 폴더에 중복될 것입니다.

새로운 폴더가 디스플레이 되면서 Folder Browser 가 나타납니다. _MAIN 블록은 Default Language 의 Editor 에 나타날 것입니다.

## Logicmaster 90 Folder 를 불러들이면서 새로운 폴더 생성하기:

Series 90-30 또는 Series 90 Micro PLC 용으로 구성된 Logicmaster 폴더를 불러들일 수 있습니다.

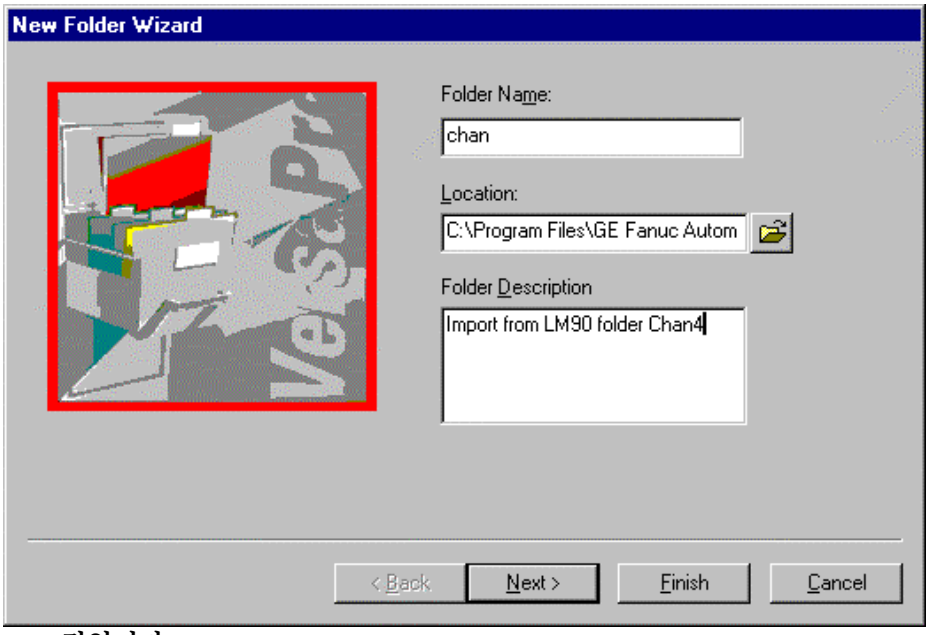
**Note:** VersaPro 는 로직을 불러들이기 전에 하드웨어 구성을 불러들일 수 있습니다. Version 1.02 이상에서는 폴더를 불러들이기 전에 Default Hardware Configuration 을 더 이상 설정할 필요가 없습니다. 또한 VersaPro 는 잠겨 있는 Control 이나 LM90 block 을 불러들일 수 있습니다.

**Note:** 본래 HHP 를 이용해 생성된 프로그램은 high end 90-30 나 VersaMax CPU 로 구성할 때, Logicmaster 에서 VersaPro 로 불러들일 수 없었습니다. VersaPro 에 low end 90-30

Logicmaster program 우선 불러들인 다음 VersaPro 에서 high end 90-30 CPU 로 변환해야 합니다.

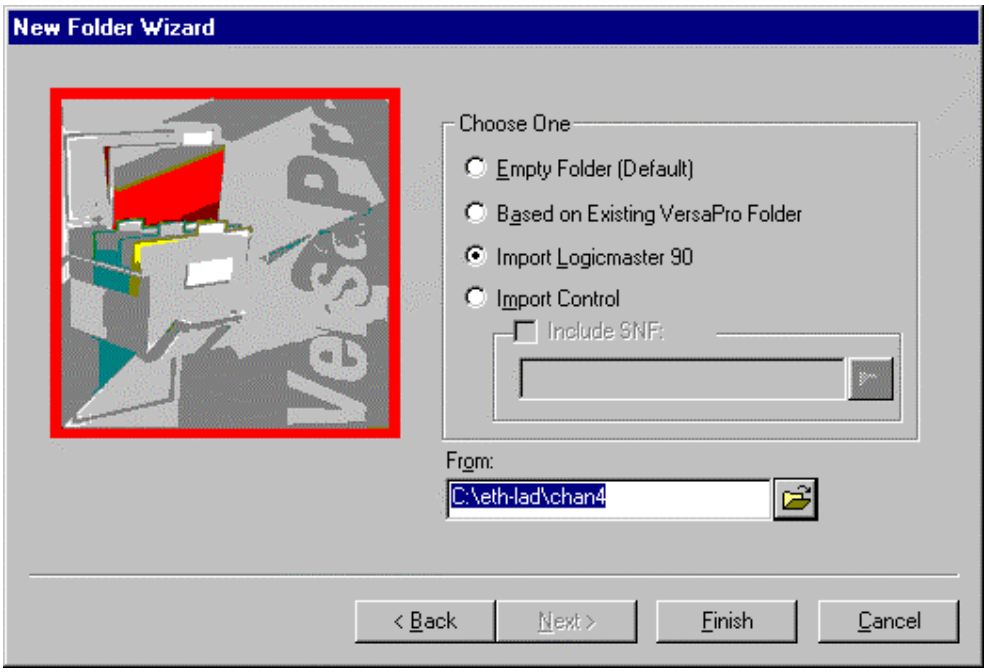
**기존 VersaPro 폴더를 기반으로 새로운 폴더를 생성하기 위해서는 VersaPro 를 가동하고 다음 단계를 수행하십시오:**

1. New Folder button  을 클릭하고, 파일 메뉴에서 New Folder 를 선정하거나 Ctrl+N 을 누르십시오. New Folder Wizard 가 나타날




것입니다.

- Folder Name field 에 새로운 폴더 이름을 입력하십시오. 폴더 이름이 필요합니다.
- 기본값으로부터 다른 위치에 새로운 폴더를 저장하기 원한다면 Location field 맞은 편에 있는 브라우즈 버튼을 클릭하십시오
- Folder Description field 에서 Folder Description 을 입력하십시오. 64 글자까지 입력할 수 있습니다. 이 부분은 옵션입니다.




2. Next button 을 클릭하십시오. New Folder Wizard 의 다음 화면이 나타납니다.
3. Import Logicmaster 90 button 을 클릭한 다음 Browse button 을 클릭합니다. 나타난 Browse dialog box 에서 새로운 폴더를 베이스로 하기 원하는 Logicmaster 90 폴더에 위치합니다.
4. Finish 를 클릭하십시오. Logicmaster 90 폴더의 컨텐츠는 새로운 폴더로 들어갑니다. 로직의 구성이 올바른지 체크합니다.
   - 만약 어떤 Syntax 에러가 발견되면 이 에러들은 정보 창에 리포트되고 프로세스는 멈추게 됩니다. 새 폴더는 생성되지 않습니다.
   - 만약 Syntax 에러가 없다면 폴더는 Logicmaster 90 폴더로부터 로직과 하드웨어 구성과 함께 상주하게 될 것입니다.

## 폴더가 잠긴 블록을 포함하고 있을 때

Logicmaster 90 폴더를 불러들이거나 잠긴(보호된) 블록을 포함하고 있는 PLC 로부터 폴더를 로딩해야 할 경우 다음 조건들이 적용됩니다:

- 잠긴 블록을 포함하고 있는 폴더가 Logicmaster 90 에서 들어올 때, 블록 이름, 타입 및 언어는 유지되지만 블록은 컨텐츠를 포함하지 않습니다.
- 보호된 블록이 PLC 에서 로드될 때, 빈 블록이 생성됩니다.

Folder Browser 블록용 아이콘을 디스플레이합니다.

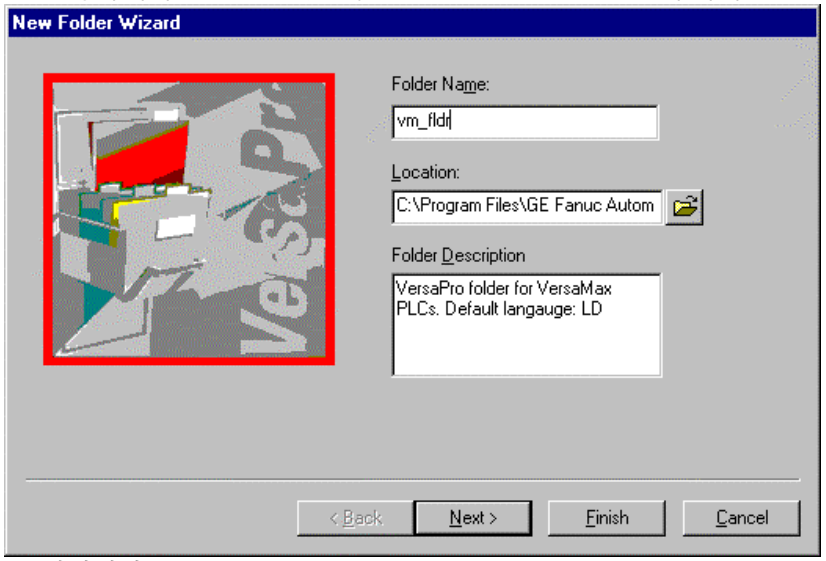
## Control 폴더에서 로직을 불러들이면서 새 폴더 생성하기:

Control 에서 VersaMax(.F2K)와 Series 90-30(.F3X) 폴더를 불러들일 수 있습니다.

**Note:** VersaPro 는 로직을 불러들이기 전에 하드웨어 구성을 불러들일 수 있습니다. Version 1.02 이상에서는 폴더를 불러들이기 전에 Default Hardware Configuration 을 더 이상 설정할 필요가 없습니다. 또한 VersaPro 는 잠겨 있는 Control 이나 LM90 block 을 불러들일 수 있습니다.

기존 VersaPro 폴더를 기반으로 새로운 폴더를 생성하기 위해서는 VersaPro 를 가동하고 다음 단계를 수행하십시오:

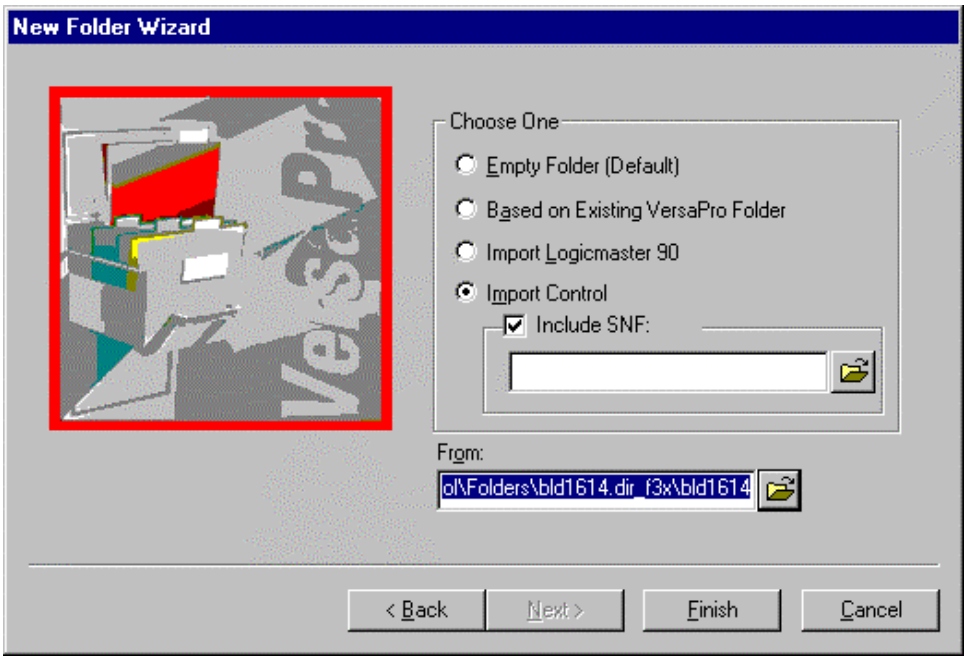
1. New Folder button  을 클릭하고, 파일 메뉴에서 New Folder 를 선정하거나 Ctrl+N 을 누르십시오. New Folder Wizard 가 나타날




것입니다.

- Folder Name field 에 새로운 폴더 이름을 입력하십시오. 폴더 이름이 필요합니다.
- 기본값으로부터 다른 위치에 새로운 폴더를 저장하기 원한다면 Location field 맞은 편에 있는 브라우즈 버튼을 클릭하십시오
- Folder Description field 에서 Folder Description 을 입력하십시오. 64 글자까지 입력할 수 있습니다. 이 부분은 옵션입니다.

2. Next button 을 클릭하십시오. New Folder Wizard 의 다음 화면이 나타납니다.




3. Import Control button 을 클릭한 다음 Browse button 을 클릭합니다. 나타난 Browse dialog box에서 새로운 폴더를 베이스로 하기 원하는 Control 폴더에 위치합니다.

**Note:** .f3x 와 .f2k extensions 을 지닌 Control 폴더만을 불러들일 수 있습니다.

**Note:** Control 폴더는 불러오기 전에 Control 에 성공적으로 내장되어 있어야 합니다.

4. Include SNF Checkbox를 선택한 후에 Browse button 을 클릭하십시오 나타난 Browse dialog box에서 불러 들릴 수 있는 Control 파일과 관련된 SNF 파일에 위치합니다(이 부분은 옵션입니다.)

**Note:** Control 폴더를 불러들일 때, 변수 닉네임을 추가하기 위해, 불러들일 수 있는 폴더와 관련된 SNF(shared name format) 파일은 Control 을 이용하는 Control 폴더로부터 우선 추출되어야 합니다. SNF 파일을 추출하는 항목에 대한 Control 온라인 도움말을 참조하십시오.

5. Finish 를 클릭하십시오. Control 폴더의 컨텐츠는 새로운 폴더에 들어갑니다. 로직은 올바른 Syntax를 구성하는지 체크합니다.
   - 만약 어떤 Syntax 에러가 발견되면 이 에러들은 정보 창에 리포트되고 프로세스는 멈추게 됩니다. 새 폴더는 생성되지 않습니다.

- 만약 Syntax 에러가 없다면 폴더는 Control 폴더로부터 로직과 하드웨어 구성과 함께 상주하게 될 것입니다.

# 폴더 이름과 닉네임에 대한 규칙

이 섹션은 새로운 폴더와 폴더 닉네임에 대한 변환과 규칙을 설명합니다.

## 폴더 이름에 대한 규칙

다음 규칙들은 새 VersaPro 폴더에 대한 폴더 이름을 적용하는 것입니다:

- 0A 폴더 이름은 130 자 이상을 넘을 수 없습니다.
- 다음 문자들은 폴더 이름으로 사용할 수 없습니다: . (period), \ (backslash), / (forward slash), : (colon), * (asterisk), ? (question mark), < (less than), > (greater than), | (vertical bar), " (quotation mark).

## 폴더 닉네임에 대한 규칙

폴더 닉네임(또는 프로그램 네임)은 PLC 에 폴더를 저장할 때 사용되는 이름입니다. Series 90-30 과 VersaMax PLC 모두 7 문자 이내의 닉네임을 인식할 것입니다.

- 닉네임은 7 문자 이하이어야 합니다.
- 닉네임의 첫 번째 캐릭터는 문자이어야 합니다.
- 다른 캐릭터는 문자, 숫자 또는 "_"를 사용할 수 있습니다.
- "_"는 연속해서 사용할 수 없습니다.
- 폴더가 생성될 때, 폴더 닉네임은 폴더 네임의 마지막 7 캐릭터까지 기본값이 됩니다.

# 새로운 폴더에 대한 디렉토리 구조와 파일

새로운 빈 폴더가 생성될 때, 다음 파일들은 폴더의 특수한 위치에서 생성됩니다. 예를 들어 폴더의 이름은 *newfolder* 입니다.

- _MAIN.blk (_MAIN block)
- *newfolder*.fld (폴더 컨텐츠 설명)
- *newfolder*.vdt (Variable Declaration Table)
- *newfolder*.hwcfg (Hardware Configurations)
- *newfolder*.ssx (Smart Store Extensions)

*LM90impt* 같이 LM90 import 이름을 기반으로 생성된 새로운 폴더에 있어서, 다음 파일들을 포함한 특수한 위치에서 생성됩니다:

- _MAIN.blk(_MAIN block)

- LM90 block 에 대한 .blk 파일
- *LM90impt*.fld(폴더 컨텐츠 설명)
- *LM90impt*.vdt(Variable Declaration Table)
- *newfolder*.hwcfg(Hardware Configurations)
- *newfolder*.ssx(Smart Store Extensions)

# 기존 폴더 열기

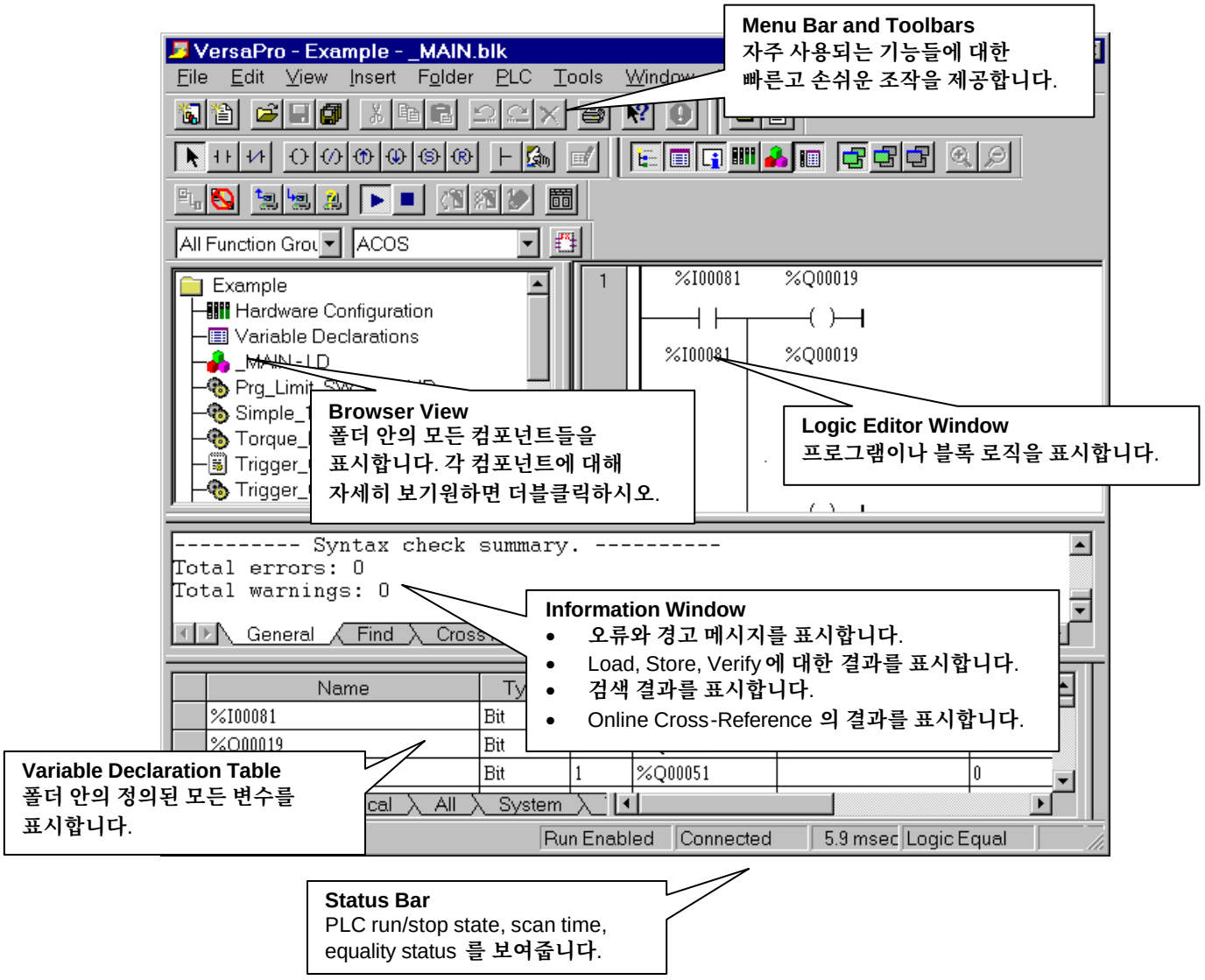
VersaPro 는 로컬 드라이브나 디스켓에서 폴더를 열 수 있습니다. 기존 VersaPro 폴더에서 열기 위해서는 다음 단계를 수행하십시오.

1. 파일 메뉴에서 Open Folder 를 선정하거나 툴바에서, Open Folder button 을 클릭하십시오. 
2. 다른 폴더가 이미 열려 있다면, "열려 있는 폴더를 닫기 원하십니까" 라는 질문과 함께 Dialog Box 가 나타납니다
3. 열려 있는 폴더를 닫기만 하면, Open Dialog Box가 나타납니다. 폴더들은 file type .fld 을 가지고 있기 때문에 "Files of type:" list 에서 "*.fld"를 볼 수 있을 것입니다.
4. 열기 위해 드라이브와 폴더를 선택하십시오
5. Open 버튼을 클릭하십시오.
6. 폴더가 열립니다. 이 콤포넌트들은 마지막 저장으로 남아 있는 속성에 따라 디스플레이 됩니다.

2

# 폴더 브라우저 사용하기

새로운 VersaPro 폴더가 생성되거나 기존 폴더가 열린 후에, 기본 창과 함께 Folder Browser 가 나타납니다(Logic Editor, Information Window, etc.).



Folder Browser 는 폴더와 관련된 모든 콤포넌트를 보여주면서 트리 구조로 디스플레이 합니다. 이 디스플레이는 폴더( _MAIN block 포함)에 있는 모든 불록 리스트뿐만 아니라 Hardware Configuration, Variable Declaration Table 과 View Tables 과의 액세스를 제공하는 아이콘을 포함하고 있습니다..

Folder Browser 는 폴더를 쉽게 운용할 수 있게 해주며, 블록과 블록 속성들을 생성, 변용, 수정 및 삭제를 할 수 있도록 해줍니다. 또한 폴더 내에서 블록의 Syntax 를 체크해 줍니다.

브라우저 창에서 항목을 열기위해 브라우저에 있는 아이템을 더블 클릭하십시오.

# Information Window

General tab 은 에러와 경고 메시지를 나타냅니다(예를 들어 Syntax Error 에 대한 로직 체킹 결과). 또한 한 언어에서 다른 언어로의 변환, 탐색, 대체 등의 세부 사항과 Find Unused Variables 명령어의 결과(폴더 메뉴에서 Find Unused Variables 를 선택)를 나타냅니다.

Find tab 은 Find in Blocks 운용의 결과를 나타냅니다(편집 메뉴나 Folder Browser CSM 에서 블록에 있는 Find 를 선택하십시오.) . 탐색하고자 하는 아이템을 이용하는 모든 블록, 렁, 명령을 찾아냅니다

Cross Reference tab 은 선택된 블록에서 사용되는 모든 변수를 열거하고 각 Variable 이 사용되는 렁과 명령들을 표시합니다. Cross Reference 를 나타내기 위해 도구 메뉴나 Folder Browser CSM 에서 View Online Cross Reference 를 선택합니다.

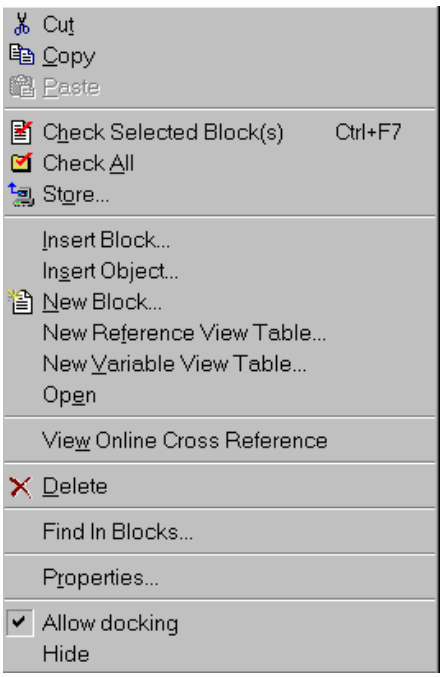
**Note:** 정보 창이 보다 큰 창으로 바뀔 때, 스크롤 박스가 너무 크기 때문에 스크롤 바로 이동시킬 수 없게 됩니다. 윈도우를 스크롤하려면 스크롤 바의 상단과 하단에 있는 방향 버튼을 사용하십시오.

# Context-sensitive Menu

Context-Sensitive Menu(CSM)는 메뉴 바에 있지 않습니다. 이것은 어떤 VersaPro 창이나 어떤 요소 상에 있는 보조 마우스 버튼을 클릭할 때 나타납니다. 메뉴는 주어진 Context 에서 어울리는 자주 사용되는 옵션 리스트를 포함하고 있습니다.

### CSM사용하기

1. 폴더 요소를 선택하십시오(폴더나 Sub-element 중 하나).
2. Element 상에 있는 보조 마우스 버튼을 클릭하십시오.
3. 아래 나타낸 그림과 유사한 CSM 가 나타납니다. 사용할 수 없는 옵션은 Gray Text 로 나타납니다.
4. 원하는 옵션을 클릭하십시오.



# VersaPro Workspace 에서 창으로 작업하기

VersaPro workspace 에 있는 모든 창은 drag & drop 으로 이동시킬 수 있고 크기를 조절할 수 있습니다. 윈도우를 움직이기 위해 타이틀 바나 상단 경계선을 클릭하십시오. 만약 타이틀 바가 없다면 타이틀 바를 드래그하십시오. 창의 크기를 조절하기 위해 가장자리나 코너를 클릭하여 드래그하십시오.

Folder Browser, Variable Declaration Table 과 Local Logic Variable Table(Motion program 에서 사용)들은 고정(Docking)시키거나 움직일(Undocking) 수 있습니다. Docking 을 선택하거나 해제하고자 할 경우 윈도우에 있는 보조 마우스 버튼을 Context-Sensitive Menu 에서 Allow Docking 을 선정하십시오.

**Note:** Folder Browser 가 Undock 상태 일 때, 열려 있는 다른 창들의 이동 및 크기 조절이나 닫을 수 없습니다.

창을 숨기는 것은 창을 일시적으로 사라지게 하는 것입니다. 이는 창의 데이터가 저장되는 것은 아닙니다. 창을 숨기기 위해 View toolbar 에 있는 윈도우 버튼을 클릭하거나 보기 메뉴에서 창의 이름을 선택하십시오 고정된 창을 숨기려면 CMS 에서 Hide 를 선택하십시오.

일단 숨겨진 창을 디스플레이하기 위해서는 창의 toolbar 버튼을 클릭하거나 보기 메뉴에서 창의 이름을 선택하십시오.

# 폴더 운용

이 섹션은 VersaPro 폴더에서 수행해야 할 추가 운용 및 기능을 기술합니다:

- 표준 파일 운용 수행하기(닫기, 저장, 삭제, 백업, 복구)
- VersaPro 폴더에 접근 제한하기(패스워드 보호 가능)
- 폴더 속성 편집하기

## VersaPro 폴더 닫기

하나의 폴더만이 한 창의 VersaPro 에서 열 수 있기 때문에, 열린 폴더는 새로운 폴더가 생성되거나 기존 폴더가 로딩되기 전에 닫아야 합니다. VersaPro 에서 폴더가 열려 있을 때 다른 폴더를 열려고 한다면, 새로운 폴더를 열기 전에 현재 열려 있는 폴더의 저장 여부를 묻는 질문을 합니다. 또한 다음 단계를 통해 폴더를 닫을 수 있습니다:

1. 파일 메뉴를 선택하고 폴더 닫기를 선택하십시오 만약 어떤 폴더의 콤포넌트를 변경했다면 각 콤포넌트에 대한 변경을 즉시 저장해야 합니다.
   - 변경 사항을 저장하기 위해서는 Yes 를 클릭하시고 변경사항을 무시하려면 No 를 클릭하십시오. 닫기 운용을 취소하려면 Cancel 을 클릭하십시오.
2. 폴더를 닫으십시오. 폴더를 닫는 절차에서, Folder Browser 는 자동으로 저장되고 마지막의 블록의 크기는 기억되는 블록 상태로 저장되며 모든 고정 가능한 창과 모든 컬럼의 개별적인 크기와 위치는 기억되어 저장됩니다.

## 폴더 저장하기

폴더의 현재 위치로 폴더와 폴더 컨텐츠를 저장합니다. 이 작업은 순수하게 폴더를 저장하지 각 파트를 개별적으로 저장하는 것은 아닙니다.

폴더를 저장하기 위해 파일 메뉴를 선택하시고 Save All 을 선택하십시오.

또한 툴바에 있는 Save folder button  을 클릭하십시오. 열린 폴더는 저장됩니다. 메뉴의 저장(SAVE) 항목은 단지 선택한 콤포넌트만을 저장합니다.

## 폴더 삭제하기

VersaPro 폴더를 삭제하기 위해 하드 드라이브에 폴더를 지정하고 삭제하기 위해 Windows 탐색기를 사용하십시오.

# 백업과 복구하기

VersaPro 는 폴더를 백업하고 백업으로부터 폴더를 복구하는 명령어를 제공합니다.

## 폴더 백업하기

백업은 데이터의 손실을 막아주는 역할을 합니다. 프로그램을 보호하기 위해 가끔 폴더를 백업하십시오.

1. 폴더 메뉴를 선택하고 백업을 선택하십시오. Backup Folder dialog box 가 나타날 것입니다.
2. Source Folder 에서 백업 받고자 하는 폴더의 경로와 이름을 입력하거나 Open Folder button 을 클릭하고 Browse dialog box 에서 폴더를 선택하여 위치를 지정하십시오.
3. Destination 에서 저장하기 원하는 곳에 백업과 위치의 이름을 입력하시거나 Open Folder button 을 클릭하고 Browse dialog box 에서 폴더를 선택하여 경로를 지정하십시오.
4. OK 를 클릭하십시오. 선택한 폴더의 백업은 지정된 위치에 지정된 이름으로 생성되고 저장됩니다.

# 백업에서 폴더 복구하기

VersaPro 는 필요에 따라 백업된 파일을 복구할 수 있도록 해줍니다. 압축된 백업 파일은 압축이 풀리고 복구 기능이 수행되면 애플리케이션으로 완전하게 복구됩니다.

1. 열린 폴더를 닫으십시오.
2. 폴더 메뉴를 선택하고 복구(Restore)를 선택하십시오. Restore Folder dialog box 가 나타날 것입니다.
3. Source Folder 에서 복구하기 원하는 폴더의 경로와 이름을 입력하거나 Open Folder button 을 클릭하고 Browse dialog box 에서 폴더를 선택하여 위치를 지정하십시오.
4. Destination 에서 복구하기 원하는 폴더의 위치를 지정하거나 Open Folder button 을 클릭하고 Browse dialog box 에서 폴더를 선택하여 경로를 지정하십시오.
5. OK 를 클릭하십시오. 선택된 폴더는 지정된 위치로 복구됩니다.

2

# 폴더 접근 제한하기

VersaPro 는 폴더를 잠금으로써 폴더 접근을 제한할 수 있습니다. 대부분의 경우에 잠긴 폴더의 컨텐츠를 볼 수는 있지만 변경시킬 수는 없습니다. 폴더는 패스워드 사용 여부에 관계없이 잠글 수 있습니다. 만약 폴더가 패스워드에 의해 잠겨 있다면 폴더를 해제하기 위해 패스워드를 입력해야 합니다. 잠긴 폴더는 "읽기전용(read-only)" 상태가 됩니다.

## 폴더 잠금

폴더를 잠그는데 다음과 같은 단계를 수행합니다:

1. 잠그고자 하는 폴더를 여십시오.
2. Folder menu 에서 Lock 을 선택합니다. Folder Security dialog box 가 나타날 것입니다.
   - 패스워드 없이 폴더를 잠기 위해서는 OK 를 클릭하십시오.
   - 패스워드 보호로 폴더를 잠기 위해서는 Password checkbox 로 Lock 를 체크하십시오. Enter Password 와 Verify Password 에 패스워드를 입력하십시오. Note: 패스워드는 대소문자를 구별합니다.
     - 패스워드는 8 문자 이하이어야 합니다.
     - 유효한 패스워드 문자는 숫자(0-9), 대문자(A-Z) 와 소문자(a-z) 그리고 언더바(_)입니다.
3. 일단 폴더가 성공적으로 잠기게 되면 Folder Title Bar 는 폴더가 보호되어 있다는 것을 가리키는 READ_ONLY 를 나타낼 것입니다. 기존 보기 테이블에 변수를 추가하는 것 이외에는 폴더를 편집할 수 없습니다.

**Note:** 폴더가 패스워드로 잠기게 되면 패스워드를 변경시킬 수 없습니다. 새로운 패스워드를 할당하기 위해서는 잠긴 폴더를 해제하고 새로운 패스워드로 폴더를 잠가야 합니다.

## 잠긴 폴더에서 제한적 운용

- 기본 보기 테이블을 편집하십시오. 새로운 보기 테이블을 생성할 수 없습니다. Variable Declaration Table 에 이미 규정된 변수는 모니터링할 목적으로 VVT 에 추가 시킬 수 있습니다. 하지만 변경사항을 저장시킬 수는 없습니다. 어드레스는 RVT 에 추가 시킬 수 있지만 저장할 수는 없습니다.
- 창이나 블록을 열고 닫습니다.
- 창을 정렬합니다
- 로직 실행을 모니터합니다.

- I/O 의 Toggle 과 Override
- Variable 값을 변경합니다.
- 폴더의 로직 블록 전체나 일부분을 복사합니다.
- 잠긴 폴더를 기반으로 한 새로운 폴더를 생성합니다. 또한 새로운 폴더는 과거 폴더의 패스워드로 잠길 것입니다.

### 폴더 잠김 해제

만약 폴더가 READ_ONLY 상태라면 폴더를 편집하기 위해 잠긴 폴더를 풀어야 합니다. 만약 폴더를 잠글 때, 패스워드를 사용했다면, 잠긴 폴더를 해제하기 위해 정확한 패스워드를 입력해야 합니다. 잠긴 폴더를 해제하기 위해 다음과 같은 단계를 수행해야 합니다:

1. 폴더 메뉴에서 Unlock 을 선택하십시오.
2. 만약 폴더에 패스워드가 있다면 Password 입력 창이 나타날 것입니다.
3. 패스워드를 입력하십시오.
4. 폴더의 타이틀 바에 있는 “READ ONLY” 표시가 사라질 것입니다.
5. 지난 패스워드는 없어집니다. 이제 폴더를 편집할 수 있습니다.

## 속성 변경하기

폴더와 블록 속성은 폴더가 생성될 때 설정됩니다. 하지만 폴더 설명, 닉네임 또는 블록 설명 등은 언제든지 변경시킬 수 있습니다. 폴더 속성을 변경하기 위해 다음과 같은 단계를 수행하십시오:

1. 변경하기 원하는 설명이나 닉네임이 있는 폴더를 여십시오
2. 편집 메뉴에서 속성을 선택하십시오. Properties dialog box가 나타납니다.
3. Folder tab 을 선택하시고 원하는 설명을 변경하십시오. 설명은 64 문자까지 입력할 수 있습니다.
4. 원하는 닉네임을 변경하십시오. 닉네임은 7 문자까지 입력할 수 있습니다.
5. OK 를 클릭하십시오. 폴더 속성은 저장되고 Properties dialog box 는 닫힙니다.

## *프로그램 인쇄하기*

Hardware Configuration 은 인쇄된 리포트에 포함되지 않습니다. 만약 Hardware Configuration 을 인쇄하기 원한다면 Stand Alone Hardware Configuration 유틸리티를 사용해야 합니다.

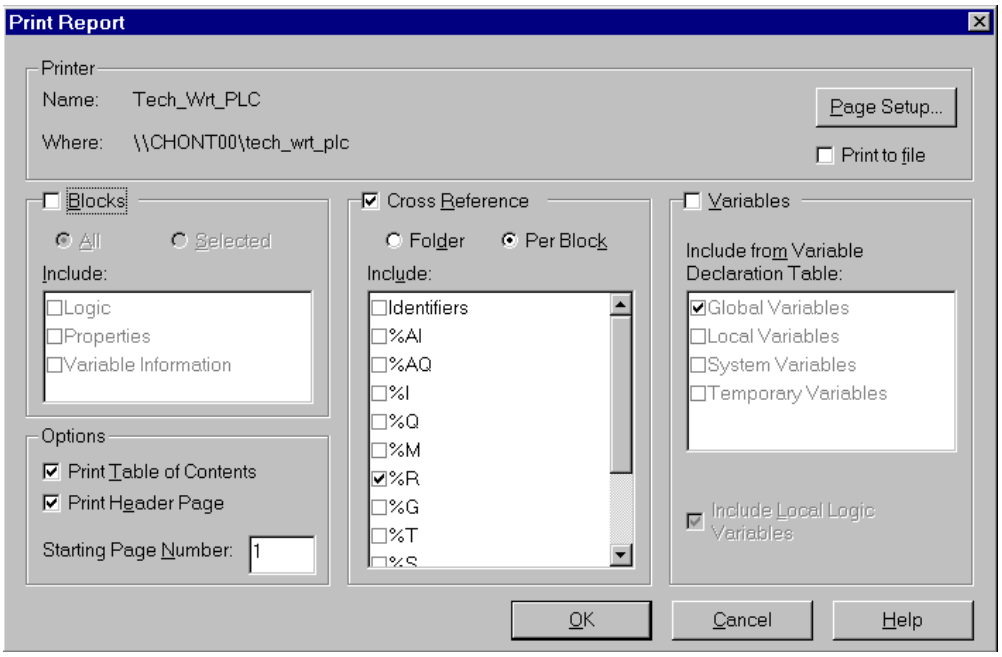
## 리포트 인쇄하기

Print Report dialog box 는 리포트가 발생할 때마다 인쇄해야 하는 콤포넌트를 규정할 수 있습니다. Block, Cross Reference, Table of Content, Header Page 및 변수를 인쇄할 수 있습니다. Print dialog box는 프린터, 프린터 범위 및 인쇄 매수를 선택할 수 있게 해줍니다. Page Setup dialog box는 용지크기, 소스, 방향 및 여백 등을 설정할 수 있게 해줍니다.

인쇄 콤포넌트, 블록 및 블록에 대한 세부사항, 목차 및 표지 선정 그리고 Cross Reference 와 관련 Reference Address 들에 대한 정보는 인쇄하기에 포함되어 있고, 처음 리포트가 발생한 후에도 남아 있습니다. 원하는 정보를 변경할 수 있지만 매번 인쇄할 때마다 리셋시킬 필요가 없습니다. 다른 모든 설정값은 리포트가 발생할 때마다 입력을 요구합니다.

프로그램을 만들고 인쇄하기 위해 다음 단계를 수행하십시오:

1. 열린 폴더에서 파일 메뉴를 선택하고, Print Report 를 선택하십시오. Report dialog box가 나타날 것입니다. 아래 나타낸 것처럼 리스트를 만들기 위해 이 dialog box 를 사용하십시오.




- 리포트에 블록을 포함시키기 위해 Blocks checkbox 를 클릭하십시오.

**Note:** 여러 블록을 인쇄할 때, 블록들은 Folder Browser 에 나타나 순서대로 인쇄됩니다.

블록이 포함되지 않은 리포트는 XREF table 나 VDT 로만 구성됩니다. Cross Reference 나 Variable 이 선택됩니다.

블록 그룹에서 리포트에 포함시키고자 하는 블록 요소들을 체크하십시오. 다음과 같은 내용을 선택할 수 있습니다:

☐ Logic

- ☐ Properties
- ☐ Variable Information (이 블록에서 사용되는 모든 Variable 과 모든 로컬 Variable, 로컬 Variable 은 블록에서 사용되지 않는다 하더라도 모두)

**Note:** 선택된 Radio button 을 활성화시키기 위해서는 먼저 폴더 브라우저로 가서 블록이나 인쇄하고자 하는 블록들을 선택하십시오.

- 리포트에 Cross Reference 리스트를 포함시키려면 Cross References Checkbox를 클릭하시고 속성 옵션 버튼을 클릭하여 모든 폴더 또는 블록마다 Cross Reference 를 포함시키기를 원하는 지를 선택하십시오.
  - ☐ 만약 블록마다 Cross Reference 를 인쇄하기를 선택하고 인식자(Jump, Label Names, Calls, MCR Names 와 End MCR Names)를 포함시키고자 한다면 Identifiers Checkbox를 클릭하십시오.

**Note:** Cross Reference 는 어드레스를 사용하는 모든 변수에 대한 함축된 어드레스를 포함하고 있습니다. 함축된 어드레스는 괄호[ ]에 의해 둘러싸여 있습니다.

- 리포트에 Variable Table 을 포함시키려면 Variables Checkbox 를 클릭하고 Cross References Checkbox를 클릭하시고 속성 옵션 버튼을 클릭하여 모든 폴더 또는 블록마다 Cross Reference 를 포함시키기를 원하는 지를 선택하십시오.
- Local Logic 변수들을 포함시키려면 Local Logic Variables checkbox를 선택하십시오.
- 용지 크기, 소스, 방향 및 여백을 선택하려면 파일 메뉴에서 Page Setup 을 선택하거나 Print Report dialog 에 있는 Page Setup button 을 클릭하십시오. 사양을 저장하려면 OK 버튼을 클릭하십시오.
- 리포트 목차를 인쇄하려면 Print Table of Contents 를 선택하십시오.
- 폴더 속성 정보를 인쇄하려면 Print Header Page 를 선택하십시오.
- 리포트의 시작 페이지를 변경하려면 Starting Page Number box에서 시작하려는 페이지를 입력하십시오. 최대 9999 까지이며, 기본값은 1 입니다.
- 리포트에 대한 용지 크기, 소스, 방향 및 여백을 선택하려면 Page Setup button 을 클릭하십시오.

2. OK button 을 클릭하거나 프린트에서 Enter 를 누르십시오.

# Logic, Variable Declaration Table 또는 Information Window Content 인쇄하기

창을 이용해 인쇄를 선택하기 원하면 파일 메뉴에서 Print 를 선택하시고 Print button  을 클릭하거나 Ctrl + P 를 누르십시오. Print dialog box가 나타날 것입니다.

1. Printer group 에서 리포트를 인쇄하는데 사용하기 원하는 프린터를 선정하려면 Drop-down List 를 사용하십시오. 또한 Properties button 을 클릭하여 Printer Properties 를 보고 선택할 수도 있고 파일 체크박스에서 Print 를 클릭하여 파일로 인쇄를 선정할 수 있습니다.
2. Print range group 에서 현재 설성, Row, Rung 또는 Tab 이나 리포트에 인쇄해야 할 모든 항목을 결정합니다.
3. Copies group 에서는 수동으로 입력하거나 필드 오른쪽에 있는 화살표 버튼을 이용하여 인쇄 매수를 결정합니다.
4. 인쇄하기 위해 OK 를 클릭하거나 인쇄하지 않고 Dialog Box 를 닫으려면 Cancel 을 클릭하십시오.

Chapter 3

# 블록으로 작업하기

Series 90-30 와 VersaMax PLC 는 폴더마다 여러 블록을 지원합니다(블록의 수는 구성된 PLC 에 따라 다양합니다. 모든 PLC 는 블록 중 하나는 _MAIN 이어야 한다는 것을 요구한다. 이 블록은 새로운 폴더가 생성될 때, 자동으로 생성됩니다.

이 장은 VersaPro Programming Software 를 이용해 블록으로 작업하는 방법에 대해 서술합니다.

## 목차

이 장은 다음과 같은 정보를 제공합니다:

- 블록과 객체를 생성하고 여는 방법
- 폴더 브라우저에서 블록과 관련 콤포넌트들을 관리하는 방법
- 폴더 콤포넌트와 관련된 변수로 작업하는 방법
- IL 과 LD 간의 블록을 변환하는 방법

3

# 블록 생성, 삽입 및 열기

VersaPro 는 폴더마다 여러 블록을 사용할 수 있게 해줍니다. 이 블록 중 하는 반드시 _MAIN block 이 존재해야 합니다. 블록은 Ladder Diagram (LD)이나 Instruction List (IL) 프로그래밍 언어에서 쓸 수 있습니다.

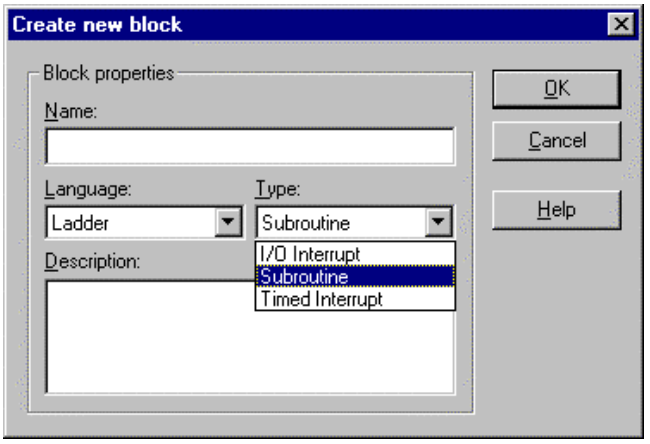
만약 프로그램에서 여러 블록을 사용하기 원한다면 _MAIN 이나 다른 블록에서 부를 수 있는 블록들을 생성할 수 있습니다. 또한 기존에 존재하지 않지만 생성할 예정인 블록을 불러 대체할 수 있습니다. 컴파일할 때, Syntax Error 는 정의되지 않은 새로운 블록이 있다는 것을 나타내는 것입니다.

여러 블록을 동시에 열 수 있습니다.

## 새로운 블록 생성하기

폴더 브라우저에서 새로운 블록을 빠르고 쉽게 생성하려면 다음 단계를 수행하십시오:

1. 폴더를 열어 파일 메뉴를 선택한 다음 New Block 을 선택하거나 툴바에 있는 New Block button 을 클릭하십시오.  New Block dialog box가 나타날 것입니다.




2. 블록 이름을 입력하십시오
   - 사용할 수 있는 문자는 A-Z 과 0-9 그리고 _ 입니다(“_” 중복 사용 못함). 이름은 7 문자까지 가능합니다.
   - 블록 이름은 문자나 _로 시작해야 합니다. Series 90-30 Timed Interrupt Block 들은 다른 규칙을 따르십시오.
   - 블록은 열려진 폴더에 존재하는 다른 블록과 동일한 이름을 가질 수 없습니다. 동일한 이름을 입력하면 에러 메시지가 나타납니다.

3. Drop-down List 를 이용해 새로운 블록에 대한 언어를 선택하십시오(Ladder 나 Instruction List).
4. 블록 타입 구성
   - **Subroutine:** 표준 블록을 생성하기 위해 Subroutine 을 선택하십시오.
   - **Timed Interrupt:** Timed Interrupt Basis 에서 작동하게 될 블록을 생성하려면 Timed Interrupt 를 선택하고 Timed Interrupt Properties 에 명시된 추가 파라미터를 입력하십시오.
   - **I/O Interrupt:** I/O 인터럽트 상황을 기반으로 실행될 블록을 생성하려면 I/O Interrupt 블록 타입을 선택하고 I/O 인터럽트 상황에 명시된 I/O Interrupt 속성 파라미터를 완성하십시오

**Note:** Timed 와 I/O Interrupt Block 들은 VersaPro Version 1.0 에서는 지원되지 않습니다.

5. 블록에 대한 설명을 입력하십시오. 64 문자까지 입력할 수 있습니다. 이 파일은 옵션입니다.
6. Dialog Box 에서 입력을 마치고자 할 때 OK 를 클릭하십시오. VersaPro 는 블록을 생성할 것입니다.

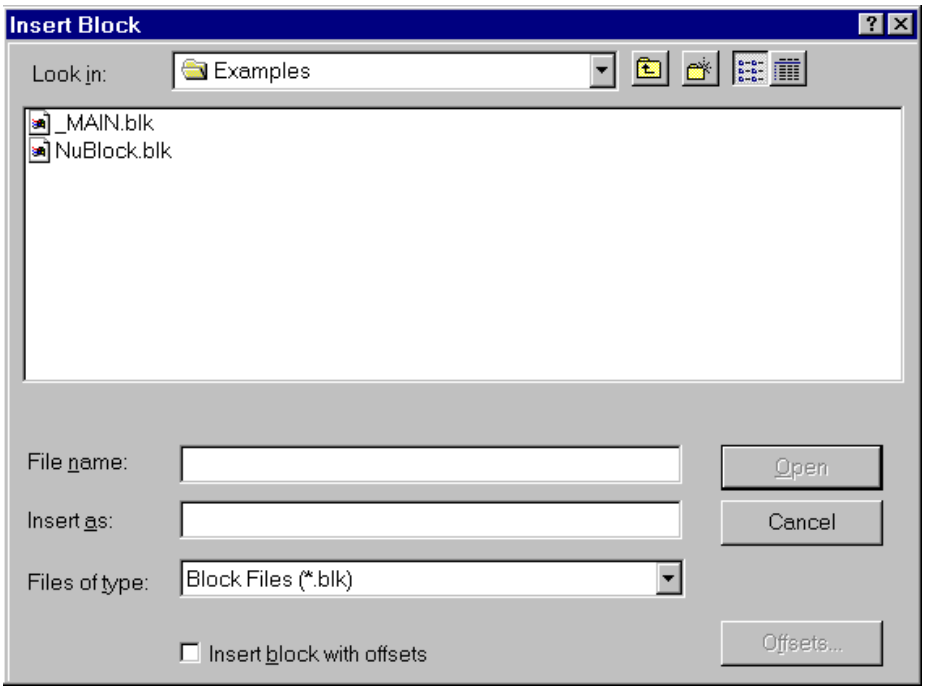
## 다른 폴더에 블록 삽입하기

Insert Block 옵션을 이용해 폴더 사이에 블록을 복사할 수 있습니다. 삽입하고자 하는 블록의 이름은 현재 폴더와 달라야 합니다. 만약 삽입하고자 하는 블록이 기존 블록과 같은 이름이면 에러 메시지가 나타날 것입니다.

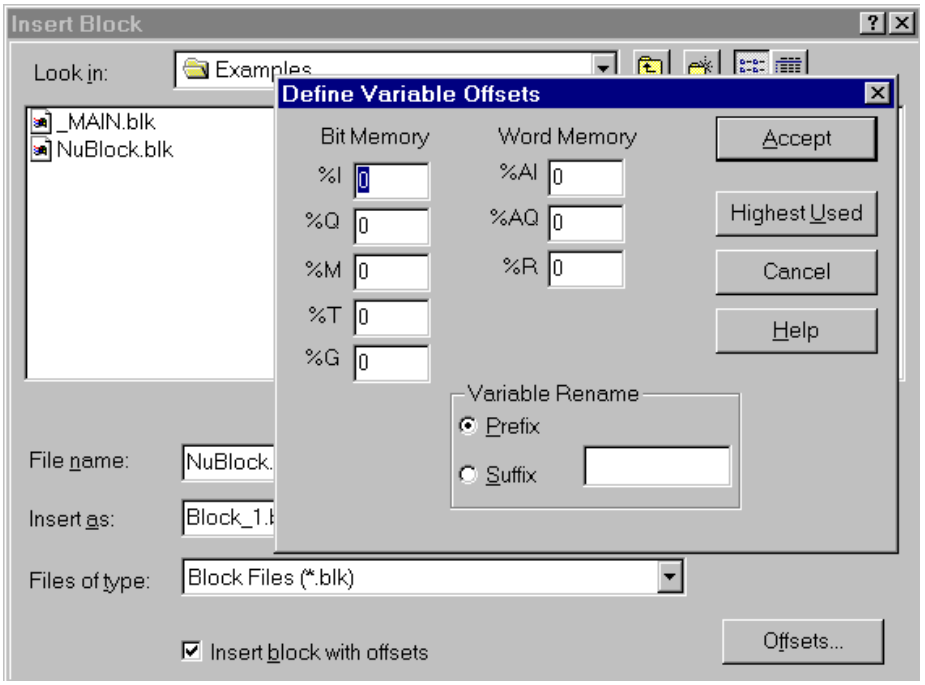
블록을 추가할 때, 블록 변수는 Variable Declaration Table 에 추가됩니다.

다른 폴더에 블록을 삽입하려면 다음 단계를 수행하십시오.

1. 블록을 삽입하려는 폴더를 여시고 Folder Browser 로 가십시오.
2. 삽입 메뉴를 선택하고 블록을 선정하든가 오른쪽 마우스 버튼을 클릭하여 Insert Block 을 선택하십시오. Insert Block dialog box 가 나타날 것입니다.
3. Look In box 에서 삽입하고자 하는 블록을 포함하고 있는 폴더를 찾으십시오.




4. **블록 타입을 선택하십시오**:
   * LD, IL **용** .blk, Local Logic **또는** Motion blocks
   * C blocks **용** .exe
5. **블록의 이름을 변경하려면** Insert As box **에서 이름을 편집하십시오**.
6. **블록에 대한 변수 빼기나 변수 이름에 접두사나 접미사 추가를 규정하려면** Offsets button **을 가진** Insert Block **을 선택하여** Offsets button **을 클릭하십시오**.




7. Define Va riable Offsets dialog box에서 변경하시고 Accept 를 클릭하십시오.

8. Insert Blocks dialog box 에서 Open 을 클릭하십시오.

블록은 현 폴더의 디렉토리에 복사됩니다.

- 만약 Folder, Hardware Configuration 이나 Variable Declaration table 이 Folder Browser tree 에 선택되면 블록은 Tree 밑에 삽입됩니다.
- 블록이나 보기 테이블이 선택되면 블록은 선택된 항목에 나타날 것입니다.
- 만약 삽입된 블록이 기존 블록과 같은 이름이라면 에러 메시지가 나타날 것입니다.

**Note**: 폴더 안에 있는 블록과 같은 이름을 가진 블록을 삽입하면 허용이 안되고 에러 메시지가 나타날 것입니다.

- 블록 추가는 Variable Declaration Table 에 블록 변수를 함께 추가 됩니다. 만약 삽입되는 블록이 폴더에 있는 블록과 충돌을 일으키는 변수를 가지고 있다면 "Variable name/scope already exists in the variable declaration table"라는 메시지가 나타날 것입니다. OK 를 클릭하고 Variable Resolution dialog box를 여십시오. 변수가 충돌하는 것을 해결해 줄 것입니다(6 장 "Variable Resolution"을 참조하십시오).

# 블록 열기

일단 블록이 Folder Browser 의 한 부분이 되면, 언제라도 블록을 열고 편집할 수 있습니다. 여러 블록들은 동시에 열 수도 있습니다. Folder Browser 에서 블록을 열려면, 열려고 하는 블록을 선택하고 블록 이름을 더블 클릭하십시오.

# 객체 삽입 및 편집

VersaPro 는 객체를 폴더와 연관짓도록 해줍니다. 예를 들면 폴더에 Excel spreadsheet, Word document 나 Access database 파일을 삽입할 수 있습니다. 객체가 폴더와 연관짓게 되면 Folder Browser 에서 더블 클릭하여 객체를 편집할 수 있습니다.

## 폴더에 객체 삽입하기:

1. 객체를 삽입하고자 하는 폴더를 열고 Folder Browser 로 가십시오.
2. 객체를 삽입하고자 하는 항목 아래를 선택하십시오.
3. 삽입 메뉴를 선택하고 객체를 선정하거나 오른쪽 마우스 버튼을 클릭하여 Insert Object 를 선택하십시오. Insert Object dialog box 가 나타날 것입니다.
4. 삽입하고자 하는 파일의 이름과 위치를 선택하십시오. VersaPro 콤포넌트 파일(.blk, .fld, .hwc, .wvr, .wrt 또는 .vdt)들은 삽입시킬 수 없습니다.
5. Open 을 클릭하고 객체를 삽입하십시오.

## 삽입된 객체 보내기:

1. 보내려는 객체를 포함하고 있는 폴더를 열고 Folder Browser 로 가십시오.
2. 객체를 열기 위해 삽입된 객체를 더블 클릭하거나 상하 방향키를 이용해 객체를 선택하고 Enter 키를 누르거나 객체를 선택하고 오른쪽 마우스 버튼을 클릭하여 Open 을 누르십시오.

만약 애플리케이션이 폴더 타입과 관련되어 있다면 선택된 객체를 열면서 애플리케이션은 가동할 것입니다. 객체와 관련된 애플리케이션이 없다면 표준 “Open With” Windows prompt 가 나타날 것입니다.

3

# *Folder Browser 에서 블록과 객체로 작업하기*

VersaPro 는 폴더와 관련된 블록, 보기 테이블 Variable Declaration Table 및 객체를 관리할 수 있는 유연한 브라우저 인터페이스를 제공합니다.

이 섹션은 Folder Browser 에서 콤포넌트와 관련된 파일과 정보를 관리하는 방법을 서술합니다.

## 블록 정렬

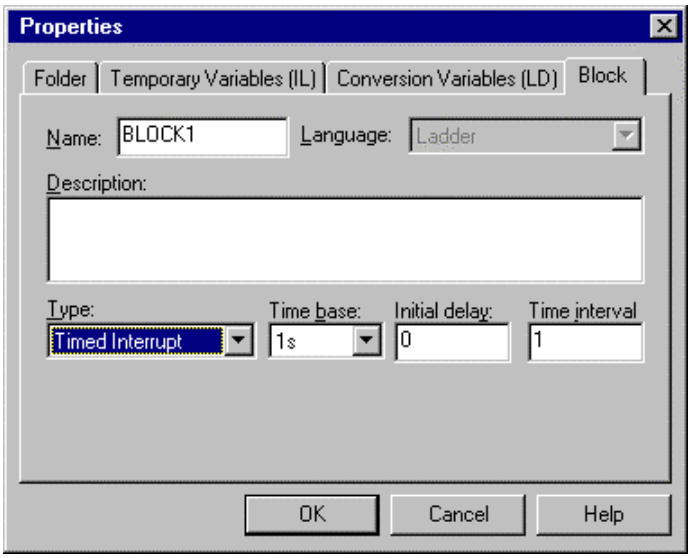
VersaPro 는 디스플레이되거나 인쇄되는 블록의 순서를 변경할 수 있도록 해줍니다(블록들은 Folder Browser 에 나타난 순서에 따라 인쇄됩니다.). 순서를 정렬하기 위해 이동하고자 하는 콤포넌트를 선택하고 새로운 위치로 Drag and Drop 합니다. 또한 표준 윈도우 자르기, 복사 및 붙여넣기 명령어로 블록을 이동시킬 수 있습니다.

## 블록 속성 보기와 수정하기

블록을 생성한 후에 블록 이름, 설명 및 타입을 보고 수정할 수 있습니다.

블록의 속성을 편집하기 위해 다음 단계를 수행하십시오.

1. Folder Browser 에서 보고자 하는 속성을 지닌 블록을 선택하십시오.
2. 편집 메뉴를 선택하여 속성을 선택하거나 오른쪽 마우스 버튼을 클릭하고 속성을 선택하십시오. Properties dialog box가 나타날 것입니다.




3. Block tab 을 선택하십시오. 블록은 블록 이름, 언어, 타입, 설명 및 다른 추가 파라미터가 나타날 것입니다(만약 블록이 I/O 나 Timed Interrupt type 일 경우).

4. 원하는 블록 파라미터를 편집하십시오. VersaPro 는 충돌을 체크할 것이고 변경할 수 없다면 에러 메시지가 나타날 것입니다.

일단 VersaPro 에 의해 변경 사항을 받아들인다면 새로운 정보는 Properties dialog box 의 Block tab 에 나타날 것입니다.

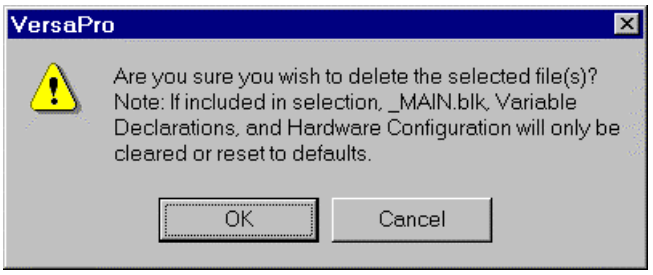
# 블록과 폴더 콤포넌트 수정하기

VersaPro 는 폴더 안이나 블록과 애플리케이션간에 블록을 복제하거나 이동시키기 위해 일반적인 자르기, 복사 및 붙여넣기 기능을 제공합니다.

이 섹션은 Folder Browser 에 있는 블록이나 다른 콤포넌트를 자르기, 복사 및 붙여넣기 방법을 기술합니다.

## 블록 자르기:

1. Folder Browser 을 열고, 자르려는 항목을 선택하십시오. *_MAIN, Variable Declarations 또는 Hardware Configuration 은 자를 수 없습니다.*
2. 편집 메뉴나 오른쪽 마우스 메뉴에서 Cut 를 선택하고 Cut button  을 누르거나 Ctrl + X 를 누르십시오. 다음과 같은 메시지가 나타날 것입니다.




3. OK 를 클릭하십시오.
4. 선택된 항목과 관련된 목차는 클립보드로 이동됩니다. 이제 다른 VersaPro 폴더나 다른 애플리케이션에서 붙여넣기를 할 수 있습니다.

## 블록 복사하기:

1. Folder Browser 를 열고 복사하고자 하는 항목을 선택하십시오. Variable Declarations 이나 Hardware Configuration 은 복사할 수 없습니다.
2. Folder Browser 에서 항목이나 항목 수를 선택하십시오. Folder node 는 복사할 수 없습니다.
3. 편집 메뉴나 오른쪽 마우스 메뉴에서 Copy 를 선택하고 Copy button  을 클릭하거나 Ctrl + C 를 누르십시오. 선택된 항목과 관련된 목차는 클립보드로 이동됩니다. 이제 다른 VersaPro 폴더나 다른 애플리케이션에 붙여넣기를 할 수 있습니다.

### 블록 붙여 넣기:

1. Folder Browser 에서 항목을 자르거나 복사하십시오.
2. 같은 폴더나 다른 폴더에서 자르거나 복사한 항목을 붙여넣기 하고자 하는 항목 위를 선택하십시오.
3. 편집 메뉴나 오른쪽 마우스 메뉴에서 Paste 를 선택하고 Paste button  ,을 클릭하거나 Ctrl + V 를 누르십시오. 붙여넣기한 항목이 같은 폴더에서 복사했다면 항목에 대한 새로운 폴더를 제공할 것인지를 묻는 대화창이 나타날 것입니다.
   - 항목에 대한 새로운 이름을 입력하고 OK 를 클릭하십시오.

Folder Browser 에서 선택한 항목 위에 붙여넣기를 하게 됩니다.

### Bitmap 이나 Text 와 같은 콤포넌트 복사하기:

Bitmap 과 같은 다른 애플리케이션에서 폴더 콤포넌트를 복사할 수 있습니다. 또한 텍스트와 같은 다른 애플리케이션에서 IL logic 을 복사할 수 있습니다.

Bitmap 과 같은 요소를 복사하기 위해서는 다음과 같은 단계를 수행하십시오:

1. Bitmap 이나 Text 로 복사하고자 하는 요소를 선택하십시오.
2. 편집 메뉴를 선택하여 Copy As -> Bitmap 를 선택하십시오(IL logic 에서 Copy As -> Text 를 선택할 수 있습니다.) . Bitmap 이나 Text 로 클립보드에 복사됩니다. Bitmap 과 같이 복사된 요소는 Bitmap 이미지를 받아들이는 다른 애플리케이션에 붙여넣기를 할 수 있습니다(Word-processing 이나 Graphics 편집 프로그램과 같은).

## Folder Browser 에서 drag and drop-Copy:

**Note:** 블록을 복사나 Drag 하기 위해 블록을 닫아야 합니다.

1. Folder Browser 를 열고 항목을 선택하십시오.
2. 선택한 항목에 마우스 키를 클릭한 상태를 유지하십시오. 선택된 항목이 밝아집니다.
3. Shift key 를 누른 상태에서 선택된 항목의 이미지를 원하는 폴더로 Drag 합니다.
4. 마우스 버튼을 놓으면 이미지가 Drop 됩니다. 선택된 항목과 컨텐츠는 잘리고 새로운 폴더로 이동합니다.
5. VersaPro 버전에서 다른 버전으로 Drag 할 때, 시도하고자 하는 Drop 동작은 변수들과 충돌하게 됩니다.(예를 들면 Target folder 의 VDT 에 이미 정의된 변수를 포함하고 있는 Reference View Table 을 자른다면), 충돌하는 변수를 변경할 것인지를 묻는 대화창이 나타날 것입니다.

- 변수 이름, 타입, 길이 또는 스코프를 변경한 다음 OK 를 클릭하거나 Drag and Drop 을 취소하려면 Cancel 을 클릭하십시오.
- Drag and Drop-Cut 기능은 _MAIN, VDT, Hardware 와 Folder 콤포넌트에는 적용되지 않습니다.
- Drag and Drop-Cut 기능은 VersaPro 내에서만 가능합니다. 항목은 클립보드로 잘리지 않고 다른 윈도우 애플리케이션에서 사용할 수 없습니다.

### Folder Browser 에서 drag and drop-Copy 하기:

1. Folder Browser 를 열고 항목을 선택하십시오.
2. Ctrl key 를 누르고 누른 상태를 유지하십시오. 선택한 항목을 클릭한 상태를 유지하십시오. 선택된 항목이 밝아집니다.
3. 선택된 항목의 이미지를 원하는 폴더로 Drag 합니다.
4. 마우스 버튼과 Ctrl key 를 놓으면 이미지가 Drop 됩니다. 선택된 항목과 컨텐츠는 복사되고 새로운 폴더로 이동합니다. VersaPro 버전에서 다른 버전으로 Drag 하거나 시도하고자 하는 Drop 동작은 변수들과 충돌하게 된다면(예를 들면 Target folder 의 VDT 에 이미 정의된 변수를 포함하고 있는 Reference View Table 을 복사한다면), 충돌을 없애기 위해 문제 변수를 변경할 것인지를 묻는 대화창이 나타날 것입니다.
   - 변수 이름, 타입, 길이 또는 스코프를 변경한 다음 OK 를 클릭하거나 Drag and Drop 을 취소하려면 Cancel 을 클릭하십시오.
   - Drag and Drop-Copy 기능은 Folder Node 나 VDT 에서는 적용되지 않습니다.
   - Drag and Drop-Copy 기능은 VersaPro 내에서만 가능합니다. 항목은 클립보드로 복사되지 않고 다른 윈도우 애플리케이션에서 사용할 수 없습니다.

# 블록 삭제하기

블록을 폴더 안에 저장해둘 필요가 없다고 생각되면 블록을 삭제할 수 있습니다. 만약 블록이 _MAIN 이라면 삭제할 수 없습니다. 그러나 동일한 절차로 _MAIN 에서 컨텐츠는 지울 수 있습니다. 블록 제거로 Variable Declaration Table 에서 글로벌 변수는 제거되지 않지만 로컬 변수는 지워집니다.

### 블록 또는 보기 테이블 삭제하기:

1. 블록을 삭제하기 원하는 곳에서 폴더를 엽니다. 폴더는 여러 블록을 가지고 있어야 합니다.
2. 삭제하려는 블록은 닫아야 합니다.
3. Folder Browser tree 에서 _MAIN 을 제외한 블록을 선택합니다.
4. 편집 메뉴나 Context-Sensitive menu 에서 Delete 를 선택하고 Delete button  을 클릭하거나 Delete key 를 누르십시오.
5. 이 운용을 수행하기 원하느냐고 묻는 경고 메시지가 나타납니다. 만약 OK 를 클릭하면 하드 드라이브에서 삭제되고 브라우저 트리에서 없어집니다.

삭제된 블록의 레퍼런스가 Syntax Check 에서 경고가 발생할 것입니다.

삭제된 블록에 포함된 글로벌 변수들은 Variable Declaration Table 에서 삭제되지 않지만 로컬 변수는 삭제됩니다.

### _MAIN block 의 컨텐츠 삭제하기:

1. _MAIN 의 컨텐츠를 삭제하기 원하는 곳에서 폴더를 여십시오. Folder Browser 에서 _MAIN 을 선택하십시오.
2. Delete key 를 누르거나 Delete toolbar button  을 클릭하십시오. _MAIN 의 컨텐트가 제거될 것이라는 것을 알게 될 것입니다.
3. OK 를 클릭하십시오. _MAIN 의 모든 로직이 제거되고 Block Description 도 제거됩니다. 폴더에 있는 다른 블록들은 손상되지 않고 다른 운용에 영향을 주지 않습니다. _MAIN 가까이 있는 변수는 삭제됩니다. 글로벌 변수들은 Variable Declaration Table 에서 삭제되지 않습니다.

# 블록 저장하기

VersaPro 는 현 위치에서 선택한 블록을 저장해 줍니다. 만약 VDT 나 Folder Browser 가 변경되더라도 블록들은 저장됩니다.

### 블록 저장하기

1. 저장하고자 하는 블록을 선택하십시오.
2. 파일 메뉴에서 Save 을 선택하십시오(이 옵션은 마지막으로 저장한 이후 선택된 블록이 변경되었을 때만 사용할 수 있습니다.). 사용중인 블록과 변경된 항목 (VDT, 폴더)은 저장됩니다.

# 블록 닫기

선택한 블록이나 열려있는 모든 블록을 닫을 수 있지만 다른 열려있는 창(예를 들면 VDT 와 Folder Browser)은 남겨둘 수 있습니다.

- **블록 닫기:** 닫고자 하는 블록을 선택하십시오. 파일 메뉴에서 Close 를 선택하십시오. 선택된 블록은 닫혀질 것입니다.
- **모든 블록 닫기:** Window 메뉴를 선택하고 Close All 을 선택하십시오. 만약 저장되지 않은 변경사항이 있는 블록이 있다면 블록을 저장하기 전에 변경사항을 저장하려는 것을 볼 수 있습니다.
  - 변경사항을 저장하고 블록을 닫으려면 Yes 를 클릭하십시오.
  - 저장하지 않고 블록을 닫으려면 No 를 클릭하십시오.
  - 블록을 열어둔 상태로 남겨 두려면 Cancel 을 클릭하십시오

3

# Drag and Drop 이용, 폴더 콤포넌트에서 자르기, 복사, 붙여넣기 및 실행취소/실행반복

VersaPro 는 자르기, 복사 및 붙여넣기 정보에 대한 표준 편집 명령을 사용합니다. 또한 VersaPro 는 폴더 콤포넌트를 만드는 변경사항을 설정하기 위한 실행취소와 실행반복을 지원합니다.

이 장은 VersaPro Folder Browser 에서 데이터와 변경사항을 관리하기 위해 윈도우-표준 명령어를 사용하는 방법을 기술합니다.

## Undo(실행 취소)와 Redo(실행반복) 사용하기

VersaPro 는 Hardware Configuration, Variable Declaration Table , Variable View Table 과 IL Editor 나 LD Editor 에서 어떤 동작(요소를 넣거나 행을 삭제하는 것과 같은)을 Undo 나 Redo 를 할 수 있습니다.

표준 메뉴 명령어, 키보드 단축 키 및 툴 바 버튼을 이용해 Undo 나 Redo 기능을 실행할 수 있습니다.

지원되는 Undo/Redo 작동 수는 1~100 사이의 값으로 설정할 수 있습니다(디폴트 값은 10 입니다). VersaPro 에 의해 지원되는 Undo/Redo 작동 수를 조절하기 위해 도구 메뉴를 선택하고 옵션을 선택합니다. 만약 한 동작을 수행하려 할 때, 동작이 복잡하거나 데이터 양이 혼란스럽기 때문에 , 실행 취소를 할 수 없고 이 사실을 알리기 위해 메시지 박스가 나타날 것입니다.

- **실행취소 하기:** Undo toolbar button.  을 클릭하고, 편집 메뉴를 선택하여 실행취소를 선택하거나 Ctrl+Z 을 누르십시오. 수행해야 할 이전 실행취소 동작이 없다면 Undo 메뉴 옵션은 엷은 색으로 변할 것입니다.
- **실행반복하기:** Redo button  을 클릭하고, 편집 메뉴를 선택하여 실행취소를 선택하거나 Ctrl+Y 를 누르십시오. Redo 해야 할 이전 실행반복 동작이 없다면 Redo 메뉴 옵션은 엷은 색으로 변할 것입니다

# 블록을 편집하면서 변수로 작업하기

VersaPro 는 로직을 작업하는 동안 변수를 생성할 수 있게 해줍니다. 이를 행하는 한 방법은 로직에 새로운 변수를 지닌 블록을 추가하는 것입니다. 추가된 블록에 포함된 새 변수는 유효성에 대해 점검을 받습니다. 만약 변수가 유효하다면 변수는 Variable Declaration Table 에 추가되고 사용할 수 있습니다.

블록에 대해 Instruction List 를 사용하고 있다면 임시 변수들은 신고해야 합니다.

## 블록 추가중 변수 생성하기

1. 폴더를 여십시오.
2. 폴더에 정의된 변수를 지닌 블록을 추가하십시오.
3. 행 검증은 기존 변수와 충돌하는 글로벌 스코프를 점검하기 위해 각 변수를 수행합니다.
4. 만약 변수가 행 검증에 실패하게 되면 문제를 설명하는 에러 메시지가 나타납니다.
5. OK 를 클릭하면 문제점을 정정할 수 있도록 하는 대화창이 나타납니다. 만약 문제를 해결하면 행 검증은 다른 변수들을 실행합니다. 만약 Cancel 을 클릭하면 변수는 VDT 에 추가되지 않고 모든 진행은 취소됩니다. 행 검증이 완성되면 충돌은 일어나지 않으며 Variable Declaration Table 은 원 데이터와 추가된 블록에서 나온 데이터가 함께 상주하게 됩니다

## 임시 변수 할당하기:

1. 폴더를 여십시오. 편집 메뉴나 Folder Browser 에 있는 Context-Sensitive menu 에서 Properties 을 선택하십시오.
2. Properties dialog box나 나타납니다.
3. Temporary Variables tab 을 선택하십시오.
4. Temporary Variables(IL) radio button 을 누르십시오.
5. Boolean 과 Integer 변수를 위한 시작 어드레스가 사용할 수 있게 됩니다.

**Note:** 만약 어떤 포인트에서 현 임시 변수를 삭제하고자 한다면 No Variables (IL) radio button 을 선택하여 삭제할 수 있습니다. 임시 변수를 위해 사용되는 메모리 영역은 변환 변수를 위해 사용되는 메모리 영역과 일치하지 않습니다.

6. Boolean 변수를 할당하기 위해 유효한 시작 bit 어드레스를 입력하십시오.

- Boolean Temporary 변수는 오직 %M, %Q 와 %T 메모리 영역에서만 할당될 수 있습니다.
- Boolean Temporary 변수는 byte-aligned address 에서 시작해야 하고 (예를 들면: 1, 9, 17, etc.), 8 bits 메모리를 사용합니다.

7. Integer 임시 변수를 할당하기 위해 유효한 시작 word 어드레스를 입력하십시오.
   - Integer Temporary 변수는 오직 %R, %AI 와 %AQ 메모리 영역에서만 할당될 수 있고 2 word 메모리를 사용됩니다.
   - 시작과 종료 어드레스 값은 1 - 99,999 에 있어야 합니다.
   - 시작 어드레스 필드는 빈 공간으로 있어서는 안됩니다. 각각의 시작 어드레스를 입력할 때 종료 어드레스는 read-only 필드에서 역동적으로 계산되고 디스플레이 됩니다.
8. OK 를 클릭하십시오. 모든 시작과 종료 값이 유효하다면 대화창은 닫힐 것이고 임시 변수들은 VDT 의 Temporary tab 에 추가됩니다. Syntax Check 는 임시변수의 특유한 생성과 정확한 수에 대해 점검합니다.

# *VersaPro 에서의 탐색 및 대체*

VersaPro 는 폴더 콤포넌트에서 데이터를 네비게이션하고, 지정 탐색 및 대체하기 위한 툴의 수를 제공합니다. 이 섹션은 Folder Browser 에서 정보를 지정하는 방법과 탐색하고 대체하는 방법을 기술합니다.

## VersaPro 에서 네비게이션

폴더 콤포넌트에서 어떤 포인트를 발견하고 이동하는 여러 방법이 있습니다:

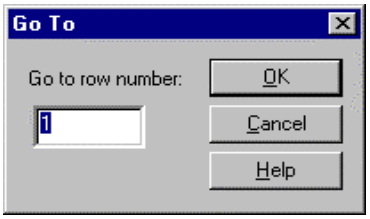
- **Go To:** 이 행위는 특수 행으로 가거나 **L**anguage Editors(IL or LD)에서 운용할 수 있도록 해줄 뿐만 아니라 특수 변수를 지정하거나 그 위치로 갈 수 있게 해줍니다.
- **Find:** 이 행위는 변수, 레퍼런스 어드레스를 지정하거나 블록 로직에서 레퍼런스를 부를 수 있도록 해줍니다.
- **Edit Variable:** IL 나 LD Editor 에서 이 명령어를 선택함으로써 Variable Declaration Table 에서 변수 정의를 이끌 수 있습니다.

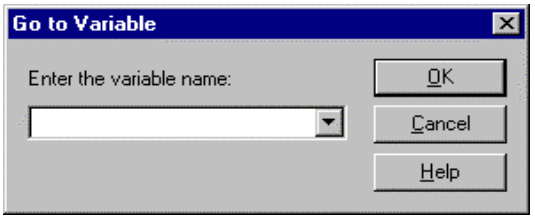
### Rung, Row 또는 Variable 지정 및 가기

특수 Rung(LD Editor), Row(IL Editor) 이나 Variable(Variable Declaration Table)에서 포커스를 변경하기 위해 GO TO 특성을 사용할 수 있습니다.

Go To 특성을 사용하기 위해 다음 단계들을 수행하십시오:

1. LD Editor, IL Editor 나 Variable Declaration Table 에 커서를 놓으십시오.
2. 편집 메뉴를 선택하고 Go To(LD 또는 IL Editor)를 선택하거나 Go To Variable(Variable Declaration Table)를 선택하십시오. Go To dialog box 가 나타날 것입니다.




3. LD Editor, IL Editor 나 Variable Declaration Table 에 대해 기술한 것처럼 dialog box 를 완성하십시오. Rung, Row 또는 Name 을 입력한 후에 OK 를 클릭하십시오.
    - **LD Editor:** 가고자 하는 Rung 의 수를 입력하십시오.
    - **IL Editor:** 가고자 하는 Row 의 수를 입력하십시오.

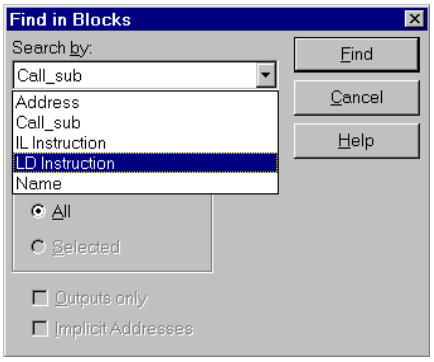
- **Variable Declaration Table:** 가고자 하는 변수의 수를 입력하십시오.

## 블록에서 이름, 어드레스 및 Call Statements 지정하기

IL 과 LD blocks 에서 변수 이름, 레퍼런스 어드레스나 Call subroutines 의 모든 발생을 발견하기 위해 Find In 특성을 사용할 수 있습니다.

Find In 기능을 사용하기 위해 다음 단계를 수행하십시오:

1. Folder Browser 나 폴더 콤포넌트에서 편집 메뉴를 선택하여 Find In Blocks 을 선택하십시오. Find In Blocks dialog box 가 나타날 것입니다.




2. 다음과 같이 대화창를 완성하십시오: 탐색을 시작하기 위해 Find button 을 클릭하십시오.

- **Search By:** 탐색하고자 하는 것을 명시하십시오.
  - **Address:** 레퍼런스 어드레스를 탐색하기 위해 어드레스를 선택
  - **Call Sub**: 서브루틴 블록에서 Call 을 찾기 위해 Call Sub 를 선택
  - **IL Instruction:** Instruction List 명령어 탐색
  - **LD Instruction:** Ladder Diagram 명령어 탐색
  - **Name:** 변수 이름을 탐색하기 위해 Name 선택
- **Search For:** 탐색 기능에서 사용하기 위해 어드레스, 블록 이름이나 변수 이름을 설정하십시오.
- **Blocks:** 탐색의 스코프를 확정하고 All blocks 나 선택된 블록을 선택하십시오(Folder Browser 에서 현재 선택된 모든 블록 탐색).
- **Outputs Only:** Outputs only 처럼 사용되는 레퍼런스에 대한 탐색 스코프 제한 (coils). Outputs only 로 탐색을 제한하기 위해 이 box 를 체크하십시오.
- **Implicit Addresses:** 어드레스가 사용되는 로직을 지정할 수 있지만 펑션 블록에서 확실한 참조를 하지 못합니다. 확실한 어드레스를 탐색하기 위해 이 박스를 클릭하십시오.

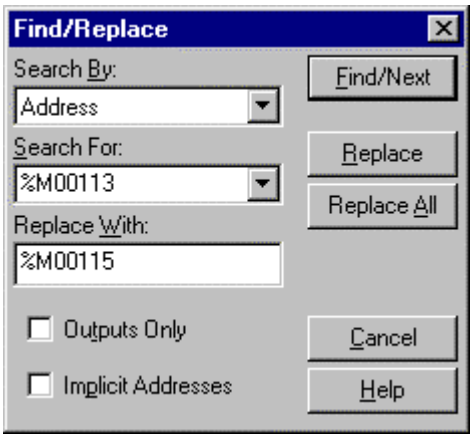
  탐색을 수행한 후 탐색 결과는 정보 창에 나타납니다. 발견된 항목은 블록, Rung, Row 및 명령어에 의해 리스트됩니다. 정보 창에서 항목을 더블 클릭하여 적절한 블록이 열리면 어울리는 요소를 선택하십시오.

# 탐색과 대체하기

VersaPro 는 블록 로직에서 요소를 찾고 대체시킬 수 있습니다. 블록 로직에서 변수 이름, 레퍼런스 어드레스와 다른 요소들을 탐색할 수 있고 새로운 이름과 어드레스를 가진 탐색 항목을 대체할 수 있습니다.

특성을 사용하기 위해 다음 단계를 수행하십시오.

1. 탐색과 대체 기능을 이용해 수정하고자 하는 블록을 여십시오. 편집 메뉴를 선택하고 Find/Replace 를 선택하십시오. Find/Replace dialog box 가 나타날 것입니다.




2. 다음과 같이 Dialog box를 완성시키십시오.
   - **Search By:** 탐색하고자 하는 것을 정의하기 위해 이 필드를 사용합니다.
   - **Address**: 레퍼런스 어드레스를 탐색하기 위해 어드레스를 선택합니다.
   - **Call Sub**: 서브루틴 블록으로 Call 하기 위해 Call Sub 를 선택합니다.
   - **Name:** 변수 이름을 선택하기 위해 이름을 선택합니다.
   - **Jump(n)/Label(n):** 프로그램에서 Jump 나 Label 을 탐색하기 위해 이를 선택합니다.
   - **MCR(N)/END_MCR(N):** 프로그램에서 MCR/ENDMCR 기능을 탐색하기 위해 선택합니다.
   - **Search For:** 탐색/대체에서 사용할 수 있는 어드레스, 블록 이름이나 변수 이름 또는 라벨을 설정하기 위해 이 필드를 사용합니다.
   - **Replace With:** 대체 운용에서 사용될 수 있는 문자열(String), 어드레스나 변수 이름을 입력하십시오.
   - **Outputs Only:** Outputs only 처럼 사용되는 레퍼런스에 대한 탐색 스코프를 제한하기 위해 이 박스를 체크 하십시오.

- **Implicit Addresses:** 펑션 블록에 명명되지는 않았지만 포함되어 있는 어드레스를 지정하기 위해 이 박스를 체크하십시오.

3. 수행하고자 하는 동작을 선택하십시오
   - **Find Next:** 프로그램에서 탐색 문자열의 다음 예를 찾으려면 이 버튼을 클릭하십시오.
   - **Replace:** 프로그램에서 탐색 문자열의 밝아진 예를 대체하려면 이 버튼을 클릭하십시오.
   - **Replace All Button:** 프로그램에서 발견된 탐색 문자열의 모든 부분을 대체하려면 이 버튼을 클릭하십시오.
4. Find/Replace dialog box를 닫으려면 Cancel 버튼을 누르십시오.

**Note:** Find and Replace 동작이 현 편집 세션에 있는 동안 명시되어 있다면, Find Next 명령어를 이용해 추가 발견된 문자열을 지정하기 위해 Find Next 기능을 사용할 수 있습니다. 다음 예를 찾기 위해 편집 메뉴에서 Find Next 를 선택하거나 F3 를 누르십시오.

# *Syntax Check 수행하기*

VersaPro 는 로직 블록을 위해 Syntax Check 기능을 제공합니다. 일단 블록 로직이 생성되면 Syntax 의 일부분이나 Syntax 에러에 대한 모든 블록을 점검할 수 있습니다.

Syntax check 에서 로직에 있는 명령어들은 구성된 PLC 에 대한 정보와 비교합니다. 지원되지 않는 명령어는 따로 표시됩니다.

Syntax check 기능은 PLC 에서 로직을 저장하거나 확인하자마자 자동적으로 수행됩니다.

수동으로 Syntax 를 점검하려면 다음과 같은 단계를 수행하십시오:

- **Check All Blocks** 을 위해 Folder Browser 로 가서, 오른쪽 마우스 버튼을 클릭하여 Check All  을 선택하십시오
- **Check Selected Blocks** 을 위해 Folder Browser 에서 체크 하려고 하는 블록을 밝게 하고, 오른쪽 마우스 버튼을 클릭하고 Check Selected Blocks  을 선택하십시오

블록이 체크되었습니다. 모든 경고와 에러는 Information Window 로 쓰여집니다.

**Note:** Information Window 에 있는 에러를 더블 클릭하여 Syntax 에러의 위치로 이동할 수 있습니다.

3

# 블록 로직 변환하기

VersaPro 는 IL logic 을 LD logic 으로 변환시켜줍니다. 한 언어에 있는 대부분의 구조는 다른 언어로 번역될 수 있습니다. 어떤 예외적인 명령어들은 변환을 시도하는 동안 에러가 발생할 수 있습니다. 만약 사용자의 로직을 변환하고자 할 경우 특수 명령어는 변환하지 않는 것이 에러 발생을 최소화할 수 있습니다. (페이지 3-23 에서 "성공적인 변환을 위한 규칙"을 참조하십시오.)

**Note:** 블록은 명령어 베이스에 의해 명령으로 변환되기 때문에 로직은 LD 언어에 대해 최적화시킬 수 없습니다.

## 변환변수 설정하기

임시 변수와 변환 변수는 IL 에서 LD 로 블록을 변환할 때 사용됩니다. 변환 변수를 설정하기 위해 다음 단계를 수행하십시오

1. 변환하고자 하는 블록을 포함하고있는 폴더를 여십시오. 편집 메뉴를 선택하고 Properties 을 선택하거나, 오른쪽 마우스 버튼을 클릭하여 Properties 을 선택하십시오. Properties dialog box 가 나타날 것입니다.
2. Conversion Variables tab 을 선택하십시오.
3. Conversion Variable Information 옵션 버튼을 클릭하십시오. Bool Address 와 Integer Address 의 시작주소가 활성화 됩니다.
4. Boolean 변환 변수에 대한 유효한 시작 bit 어드레스를 입력하십시오. 종료 어드레스는 자동으로 계산됩니다.
5. Integer 변환 변수에 대한 유효한 시작 word 어드레스를 입력하십시오. 종료 어드레스는 자동으로 계산됩니다.
6. 구성 범위를 받아들이기 위해 OK 를 클릭하십시오.

**Note**: 변환 변수용으로 사용되는 메모리 영역은 임시 변수용으로 사용되는 메모리 영역과 겹쳐질 수 없습니다.

**Note**: Conversion dialog box 는 어드레스 범위를 가지고 있기 때문에 나중에 블록 변환은 처음 변환 할 때 명시된 동일한 변수를 사용할 수 있습니다.

## 블록 로직 변환하기

일단 변환 변수가 구성되면 변환을 위한 블록을 선택할 수 있습니다. 블록을 변환하기 전에 블록의 문장을 체크 해야 합니다(폴더 메뉴에서 Check Block 이나 Check All 을 선택하십시오.).

1. Folder Browser 에서 블록을 선택하거나 활동중인 블록을 변환하려는 블록을 만드십시오.
2. 도구 메뉴를 선택하여 Convert Block 을 선정하십시오.

변환을 시도하고자 하는 블록에 있는 로직은 문장 구성상 적합한지를 점검합니다. 만약 로직이 문장 구성상 적합하다고 나타나면, 로직은 한 언어에서 다른 언어로 변환되고 변환된 사항은 Information Window 에 나타납니다.

만약 어떤 Syntax 에러가 발견되면 에러는 Information Window 에 로그되고 변환은 취소됩니다.

### 성공적인 변환을 위한 규칙

다음 가이드라인은 성공적인 블록 변환을 위해 제안한 것입니다:

- 다음 명령어를 포함하고 있는 IL 로직은 LD 로 변환할 수 없습니다.
  - XOR, XORN, XOR(
  - ANDN(, ORN(, XORN(
- 임시 변수의 사용을 정의하기 위해 Properties dialog box 를 사용하는 것은 IL 로직과 변환 변수를 위해 필요하고, IL logic 를 LD 언어로 변환할 때 필요합니다.

# Non-Nested Instructions 변환하기

Series 90 Micro 과 low-end 90-30 PLC 는 MCR, ENDMCR, JUMP 와 LABEL 명령어의 Non-nested 형태를 지원합니다. Non-nested 명령어를 포함하고 있는 Logicmaster 폴더를 불러오거나 Series 90 Micro 폴더를 VersaMax Nano/Micro 폴더를 변환하고자 할 때, 이들 명령어는 자동으로 nested 형태로 변환되지 않을 것입니다. 변환 후 프로그램을 컴파일하기 위해 이들 명령어를 nested 형태(MCRN, ENDMCRN, JUMPN 과 LABELN)로 바꿔줄 필요가 있습니다.

Non-nested 명령어를 변환하기 위해 도구 메뉴로 가서 Nested Conversion 에서 Non-Nested 를 선택하십시오. 변환하기를 원하는 지를 물어보는 대화창이 나타날 것입니다. OK 를 클릭하십시오.

**Note:** 모든 Non-nested 명령어는 Nested 형태로 변환될 것입니다 이변환은 컴파일된 로직의 크기를 증가시킬 수 있습니다. 또한 이 변환은 실행 취소를 시킬 수 없습니다.

Chapter 4

# LD Editor 에서 작업하기

The LD Editor window 는 래더 로직에서 생성된 프로그램을 생성, 편집, 디스플레이 및 모니터할 수 있다. 이 장은 LD Editor window 와 LD Editor window 를 볼 수 있도록 만드는 방법과 로직을 생성하고 편집하는 방법을 소개하고 있습니다.

## 목차

이 장은 다음과 같은 정보를 제공합니다:

- LD Editor 접근 방법
- LD Editor Window 개요
- LD Editor 의 창 설정과 Rung 편집하기와 코일 점검하기 옵션을 디스플레이하고 설정하기 위한 방법
- 로직을 입력하고 편집하는 방법

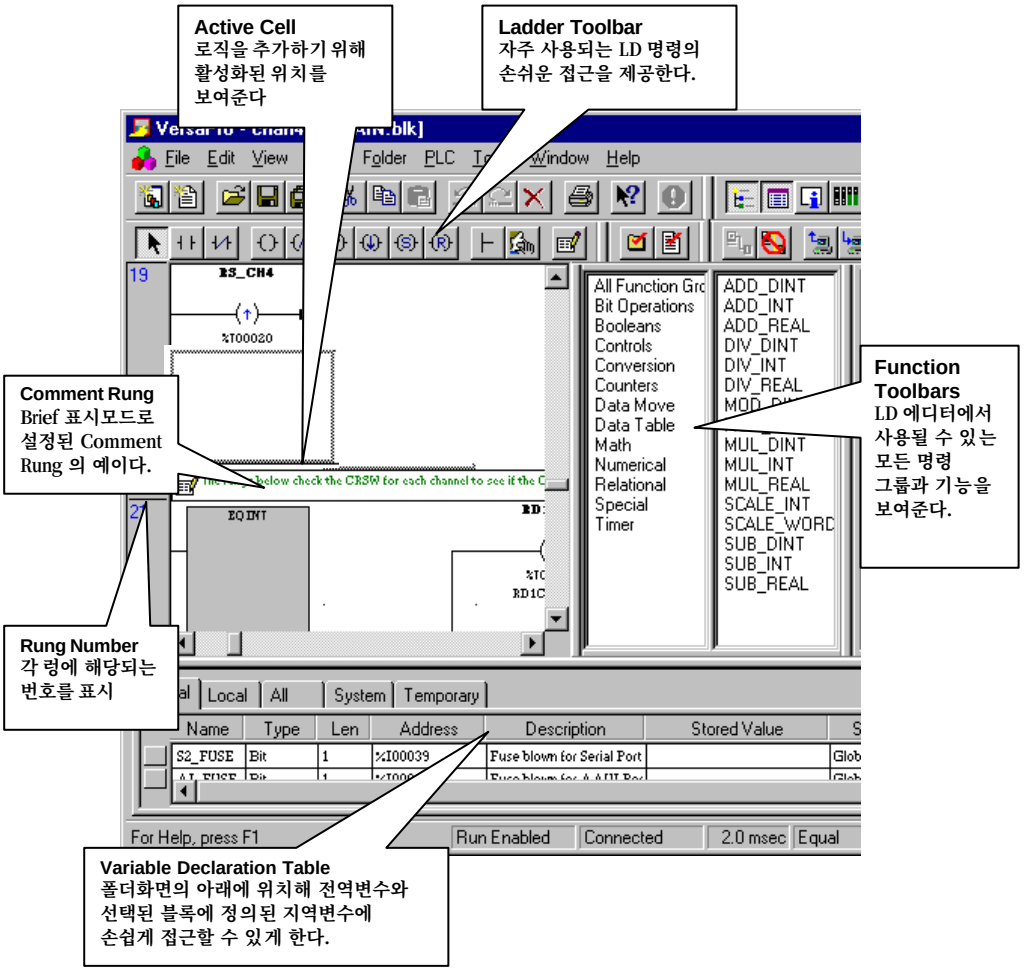
4

# LD Editor 개요

The Ladder Diagram Editor는 Ladder Diagram 프로그래밍 언어를 이용해 프로그램을 생성하는데 사용됩니다. Ladder Diagram은 PLC에 의해 수행된 동작을 그래픽으로 표현하는데 사용됩니다.

편집 창은 행으로 분할되어 있고(Rungs) 각 Rung 셀로 분할되어 있습니다. 각 셀은 로직 요소를 받아들일 수 있습니다(펑션 블록 또는 릴레이 블록). LD Editor는 키보드, 마우스 및 툴바를 통해 래더 로직 요소를 입력할 수 있습니다. 모든 요소는 선택된 셀의 컨텐츠를 오버라이트함으로써 삽입됩니다.

아래 그림은 LD Editor window와 브라우저 보기의 일반적으로 레퍼런스된 콤포넌트를 보여주고 있습니다.



# LD Editor Window 만들기

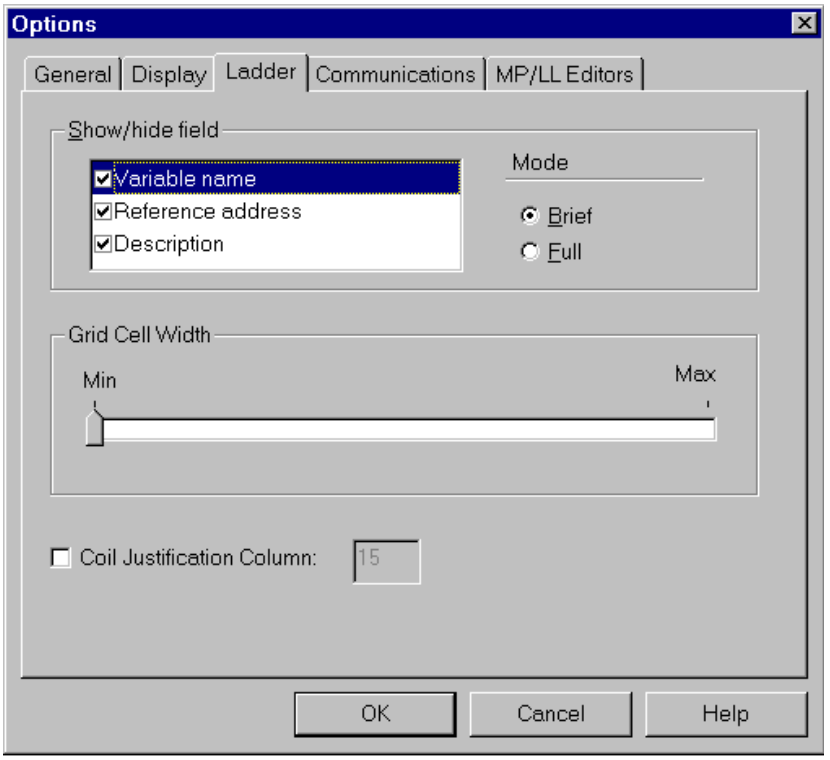
VersaPro 는 LD Editor window 에서 디스플레이되는 정보의 포맷을 선택하고 눈금과 디스플레이되는 크기를 조절 할 수 있게 해줍니다

이 섹션은 사용자가 조절할 수 있는 LD Editor window 에 대해 기술합니다.

## 래더 디스플레이 옵션 만들기

LD Editor 에서 작업할 때, 사용자는 온라인으로 필드를 만들기 원할 수 있습니다. 래더 디스플레이 옵션을 만들기 위해 다음 단계를 수행하십시오:

1. Tools 메뉴를 선택하여 Options 을 선택하거나 LD Editor 에서 오른쪽 마우스 버튼을 클릭하고 Options 을 선택하십시오. Options dialog box 가 나타날 것입니다. Ladder tab 을 선택하십시오.




2. 사용자가 필요로하는 만큼 디스플레이 옵션을 만드십시오: 변경을 허락하기 위해 OK button 을 클릭하십시오.

- **Show/Hide Field:** LD Editor window 에서 디스플레이할 필드를 설정해 줍니다:
  - **Variable Name:** 모드에서 Variable Name 을 선택하고 해당 버튼을 클릭함으로써 요약 또는 전체로 디스플레이 모드를 설정할 수 있습니다. 전용 버튼을 체크 하십시오. Brief mode 는 한 행으로 디스플레이 되고 Full mode 는 두 행까지 디스플레이 됩니다.

- **Reference Address:** 변수와 관련된 레퍼런스 어드레스를 보려면 이 박스를 체크 하십시오.
- **Description:** 변수 설명을 보려면 이 박스를 체크 하십시오. 만약 모든 설명을 보기 원한다면 Grid Cell Width 을 증가 시킬 수 있습니다. 또한 해당 버튼을 체크 함으로써 요약 또는 전체로 디스플레이 모드를 설정할 수 있습니다. Brief mode 는 한 행으로 디스플레이되며 Full mode 는 네 행까지 디스플레이 됩니다.

- **Grid Cell Width:** 셀 폭 조절을 위해 그리드를 만들 수 있습니다. 폭을 늘리고 줄이려면 Min 에서 Max 까지 조절하십시오.
- **Coil Justification Column:** 만약 이 박스가 체크되면 코일을 삽입할 때 코일은 명시된 컬럼에 놓입니다. 유효 값은 10 에서 20 까지 입니다. 만약 이 박스가 체크되지 않으면 코일은 다음 빈 컬럼에 삽입됩니다.

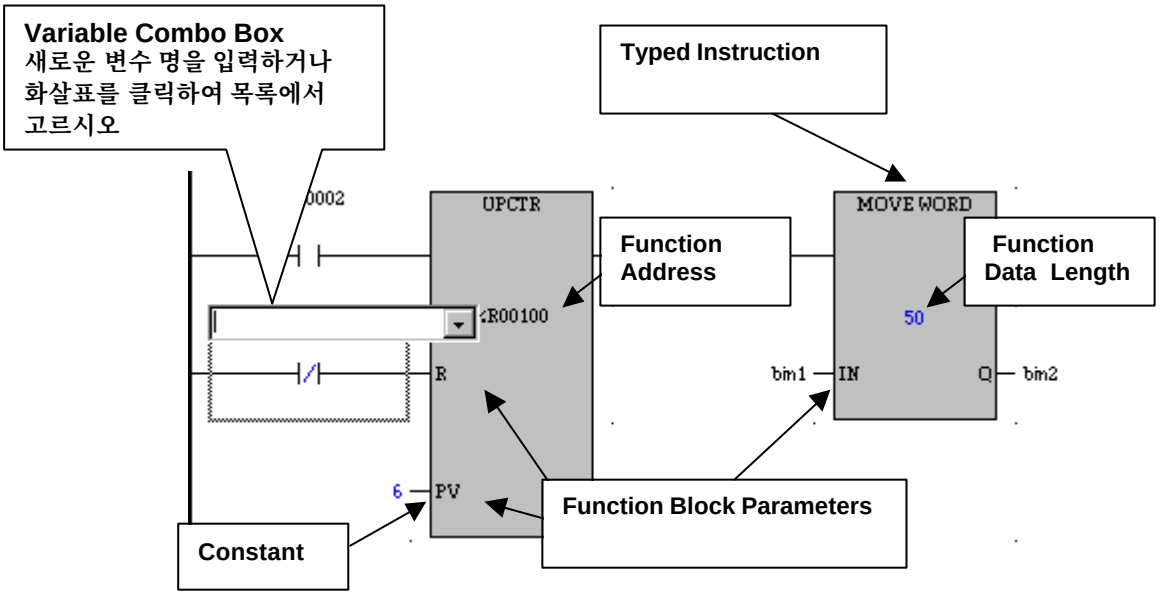
## Zoom Ratio 조절하기

Zoom in/Zoom out 버튼을 클릭함으로써 LD Editor 에 대한 Zoom ratio 을 늘리거나 줄일 수 있습니다. Zoom ratio 을 조절하려면 다음 단계를 수행하십시오:

1. LD block 을 만드십시오
2. 보기 메뉴를 선택하고 Zoom In  이나 Zoom Out  을 선택하십시오. 툴바 버튼을 클릭할 수도 있습니다.

# 명령어와 명령어 파라미터 입력하기

이 섹션은 래더 프로그램에서 명령어와 Rung 을 입력하고 수정하는 방법을 기술합니다. 프로그램 로직은 로직의 Rung 에 함께 배열된 릴레이, 타이머, 수학 기능 및 다음 기능과 같은 요소로 구성되어 있습니다. 변수와 상수는 기능에서 입력, 출력처럼 기능 파라미터에 할당됩니다.




위의 예는 LD Editor 에서 로직을 입력할 때 적용되는 개념과 세부사항을 설명한 것입니다.

- **Typed Instructions (Functions):** VersaPro 에서 많은 펑션 블록들은 "정형화"되어 있습니다. 이것은 타입(INT, DINT, REAL)이 변수가 아니라 펑션과 관련되어 있다는 것을 의미합니다.
- **Function Data Length:** 많은 펑션들은 데이터 길이를 요구합니다. 길이는 변수가 아니라 펑션과 관련되어 있습니다.
- **Function Address:** 펑션 블록에 의해 요청된다면 레퍼런스 어드레스는 디스플레이 됩니다.
- **Function Block Parameters:** 펑션 블록에서 입출력은 완성되어야 합니다. 많은 펑션 블록 파라미터들은 Add 기능과 같은 변수나 상수값을 요구합니다. 기타 펑션 블록들은 펑션의 실행을 가능하게 하고 조절하기 위해 Boolean 파라미터(상수와 같은)를 요구합니다.
- **Variable Combo Box:** 펑션 블록 파라미터에 변수를 할당할 때, Variable Combo Box는 기존 변수나 새로운 변수를 빠르게 선택할 수 있게 해줍니다.

4

# 명령어 입력하기

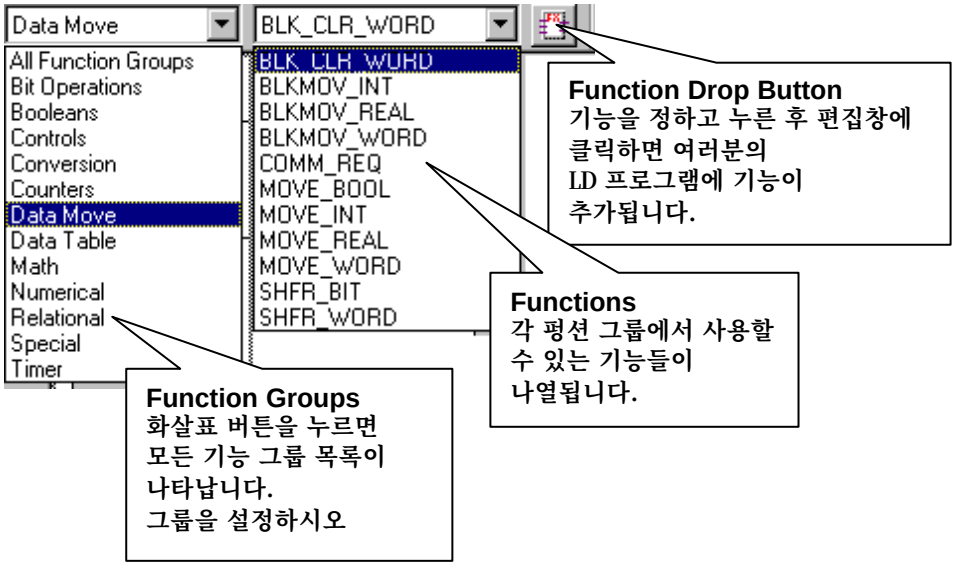
개별 레퍼런스와 편집 스타일에 맞게 LD Editor에 명령어를 입력하는 방법이 여러 가지 있습니다. VersaPro 를 이용해 Function Toolbar와 Ladder Toolbar를 이용하거나 직접 키보드 입력을 통해 명령어를 입력할 수 있습니다(마우스 명령어는 필요 없음). 이 섹션은 프로그램에 명령어를 입력할 수 있는 다양한 방법에 대해 기술합니다.

## Function Toolbar를 이용해 명령어 입력하기

Function toolbar LD 프로그램에 펑션 블록은 선택하고 추가하는데 사용됩니다. Function Toolbar는 두 가지 형태 중 한 형태로 디스플레이할 수 있습니다. 한 형태는 스크린에서 일부분을 차지하는 콤팩트형이며 다른 한 형태는 모든 그룹과 기능들을 디스플레이할 수 있는 확장형입니다.

### Function Toolbar를 이용해 펑션 선택하기 – Compact

콤팩트형으로 디스플레이 되는 Function Toolbar는 Group Combo Box, Instruction Combo Box와 Function Drop button으로 구성되어 있습니다. 콤팩트형을 선택하려면 보기 메뉴를 선택하여 Function Toolbars -> Compact 를 선택하십시오. Function Toolbar는 LD Editor window상에 나타날 것입니다.

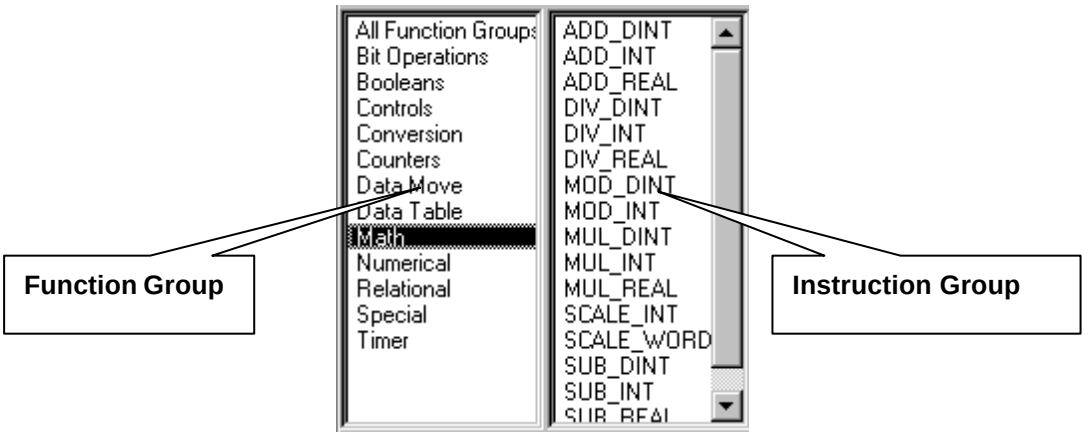



- 펑션 그룹을 선택하려면 Function Groups 박스에 있는 화살표를 클릭하고 펑션 그룹을 선택하십시오. 모든 명령어를 보려면 All Function Groups 을 선택하십시오.

- 명령어를 선택하려면 리스트에 위치한 펑션 이름의 첫 글자를 입력하거나 전체 리스트를 스캔하기 위해 마우스를 사용해 Functions 박스에 있는 화살표로 클릭하십시오.
- LD Editor window 에서 펑션을 드롭하려면 Function Drop Button 을 클릭하고 LD Editor window 에 있는 Insert Location 에 커서를 위치하고 프로그램으로 펑션을 드롭하기 위해 클릭하십시오.

## Function Toolbar 를 이용한 펑션 선택하기 – Expanded

확장형으로 디스플레이된 Function Toolbar 는 Group window 와 Instruction window 로 구성되어 있습니다. 확장형을 선택하려면 보기 메뉴를 선택하여 Function Toolbars -> Expanded 를 선택하십시오. Group window 와 Instruction window 는 LD Editor window 오른쪽에 나타날 것입니다.



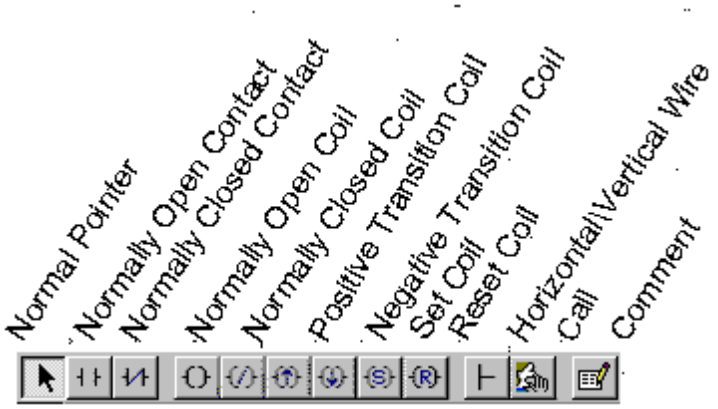

- 펑션 그룹을 선택하려면 Group window 에서 Function Group 이름을 밝게 하십시오. 모든 명령어를 보려면 All Function Groups 을 선택하십시오.
- 명령어를 선택하려면 프로그램에 추가하려는 명령어가 있는 명령어 리스트를 스캔하기 위해 Slide Control 을 사용하십시오.
- LD Editor window 에서 펑션을 드롭하려면, 명령어 이름을 더블 클릭하고(커서를 펑션 블록 아이콘으로 변경하는 것에 유의하십시오), LD Editor window 에 있는 Insert Location 에 커서를 위치하고 프로그램으로 펑션을 드롭하기 위해 클릭하십시오.

## Ladder Toolbar 를 이용해 명령어 입력하기

일반적으로 사용하는 Boolean 펑션을 선택하고 LD 프로그램에서 명령어를 코멘트하고 부르기 위해 래더 툴바를 사용할 수 있습니다.

툴바 명령어 중 하나를 선택하려면 툴바 버튼을 간단하게 클릭하고 삽입지점으로 커서를 이동시킨 후 명령어를 배치 위해 클릭하십시오.

코멘트 툴바 버튼은 현커서 위치에서 코멘트를 배치합니다.

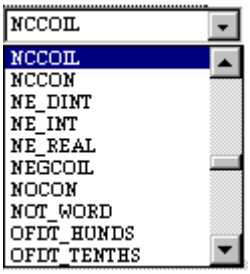



**Note:** 펑션이나 명령어를 설치할 때, 펑션의 예는 마우스를 클릭할 때마다 LD Program에 삽입될 것입니다. 활성화된 펑션을 취소하시려면 Normal Pointer 툴바 버튼을 클릭하십시오.

## 키보드로 명령어 입력하기

마우스를 사용하는 것이 아니라 키보드로 로직을 직접 입력할 수 있습니다. 키보드를 이용하여 명령어를 입력하려면 다음 단계를 수행하십시오:

1. 명령어를 삽입하고자 하는 셀에서 네비게이션 하십시오.
2. Shift + Enter 키를 누르십시오. 모든 명령어 리스트가 들어 있는 Combo Box가 나타날 것입니다(만약 셀이 비어 있다면 Enter 키를 누르십시오.).

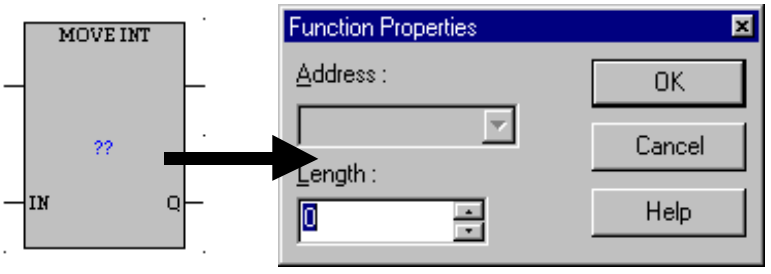



3. 펑션을 지정하려면 명령어 이름의 첫 글자를 입력하거나 리스트 전체를 스크롤하기 위해 방향 키를 사용하십시오.
4. 이름을 선택하고 프로그램에 명령어를 삽입하려면 Enter 키를 누르십시오.

## 어드레스와 길이 정보 완성하기

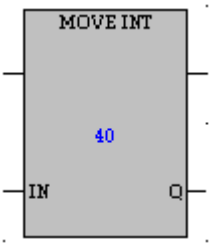
어떤 펑션 블록들은 정확하게 운용하기 위해 펑션에 대한 길이(예, 이동 명령어)나 어드레스(예, 타이머 명령어)를 제공할 것을 요구하기도 합니다. 이러한 정보를 요구하는 펑션들은 펑션 블록 안에 ??를 표시할 것입니다. 어드레스와 길이 정보를 완성하기 위해 다음 단계를 수행하십시오:

1. 어드레스와 길이 정보를 요구하는 펑션을 입력하십시오. 펑션 블록에서 ?? 필드를 확인하십시오.
2. ?? 영역을 더블 클릭하거나 펑션을 선택하고 Enter 키를 누르십시오. 길이와 어드레스를 입력할 수 있도록 Function Properties dialog box나 나타날 것입니다.




3. Function Properties dialog box에서 활동하는 파라미터에 따라 길이나 어드레스를 입력하십시오. 정보가 완성되면 OK 를 클릭하십시오.

4. 펑션은 Function Properties dialog box에서 공급하는 길이나 어드레스를 디스플레이할 것입니다.



# 펑션 블록 파라미터 완성하기

펑션 블록, 코일 또는 콘택트가 프로그램에 추가되면 각 파라미터에 변수나 상수를 할당할 필요가 있습니다. 이 섹션은 다음 방법으로 펑션 블록 파라미터를 완성하는 방법을 기술합니다:

- 파라미터에 기존 변수 할당하기
- 파라미터에 대한 새로운 변수 생성하기
- 파라미터에 상수값 할당하기

## 펑션 블록 파라미터에 변수 할당하기

래더 로직에 있는 요소와 파라미터에 변수를 할당할 수 있습니다. 변수는 전역이나 지역일 수 있습니다. 변수에 대한 기본적인 범위는 Global Scope(전역)입니다. 펑션 블록 파라미터에 기존 변수를 할당하거나 새로운 변수를 할당할 수 있습니다. 이 섹션은 마우스와 키보드를 이용해 파라미터에 변수를 할당하는 방법을 기술합니다.

**Note:** Variable Declaration Table 에서 펑션 블록 파라미터로 선택된 변수를 드래그&드롭할 수 있습니다.

**Note:** 오버랩된 Variable Declaration 은 PLC 에 저장 값을 저장하려고 할 때 충돌할 수 있습니다. 충돌을 해결하기 위해서는 중복을 피하도록 변수를 재지정하기 위해 VDT 를 사용해야 합니다(예를 들면 길이 3 을 지닌 %R8 에 있는 변수와 길이 2 의 %R10 에 있는 변수는 %R10 에서 오버랩을 가지고 있습니다.).

### 키보드를 이용해 기존 변수 할당하기:

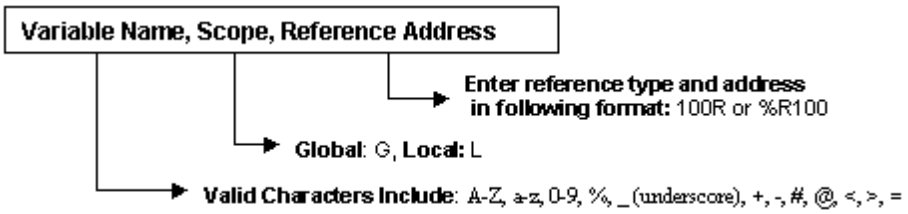
1. 변수를 할당하기 원하는 파라미터에 커서를 위치하도록 방향키를 사용하십시오.
2. 다음과 같은 변수를 입력하거나 선택하십시오:
   - 정의된 변수 리스트에서 변수를 선택하고 Enter 키를 누르십시오. Variable Combo-Box 가 나타날 것입니다. 변수를 디스플레이하기 위해 상하 키를 사용하십시오. 사용하려고 하는 변수를 찾으면 받아 들이 위해 Enter 키를 누르십시오.
   - 이름으로 변수를 선택하려면 변수 이름을 입력하십시오. 받아 들이기 위해 Enter 키를 누르십시오.

## 마우스를 이용해 기존 변수를 할당하기:

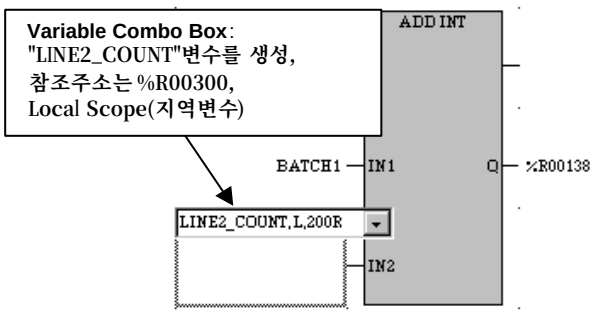
1. 변수를 할당하기 원하는 파라미터 셀을 더블 클릭하십시오. Variable Combo-Box가 나타날 것입니다.
2. 변수 리스트를 확장하려면 Combo-Box에서 화살표를 클릭하십시오. 리스트를 스크롤하기 위해 Slide Control을 사용하십시오.
3. 리스트에서 변수를 선택하십시오. Enter 키를 누르거나 받아들이기 위해 파라미터 셀 영역 바깥을 클릭하십시오.

## 펑션 블록 파라미터용 새로운 변수 생성하기

1. 키보드와 마우스를 이용하여 파라미터 셀을 선택하십시오.
   - **Keyboard Select:** 변수를 할당하고자 하는 파라미터에 커서를 갖다 놓기 위해 방향 키를 사용하십시오. Enter를 누르십시오. Variable Combo-Box가 나타날 것입니다.
   - **Mouse Select:** 파라미터 셀을 더블 클릭하십시오 Variable Combo-Box가 나타날 것입니다.
2. 다음 가이드 라인을 이용해 편집 박스에 변수 이름, 레퍼런스 어드레스와 스코프(옵션)를 입력하십시오. (필드는 “,”에 의해 분리해야 합니다.)




- 다음 예는 ADDINT Function의 파라미터 IN2에 대한 변수 “LINE2_COUNT”을 생성하는 것을 보여 주고 있습니다.




3. 변수를 받아 들이려면 Enter 키를 누르십시오. – 활동중인 셀을 클릭 오프하려면 마우스를 이용하십시오.

**Note:** 기존 변수에 대한 정보를 편집하려면 LD Editor 에서 변수를 선택하고, 오른쪽 마우스 버튼을 클릭하여 Edit Variable 을 선택하십시오. 변수는 Variable Declaration Table 에서 밝아질 것입니다.

LD 로직을 변경하고 기존 변수를 대체하게 됩니다. 변수를 대체할 때, 그 정보에 대한 정보는 여전히 Variable Declaration Table 에 남아 있게 됩니다. 만약 새로운 변수로 기존 변수를 대체한다면 새로운 변수 Variable Declaration Table 에 추가됩니다. 어떤 경우에는 상수가 변수에 의해 대체될 수 있습니다.

변수(또는 상수)는 기존 변수나 새로운 변수로 대체시킬 수 있습니다.

## LD 에서 기존 변수로 변수 대체하기:

1. 키보드와 마우스를 이용해 파라미터 셀을 선택하십시오.

   - **Keyboard Select:** 변수를 할당하기 원하는 파라미터에 커서가 위치할 수 있도록 방향키를 사용하십시오. Enter 키를 누르십시오. Variable Combo-Box 가 나타날 것입니다. 현 변수나 상수는 편집 필드에 디스플레이될 것입니다.

   - **Mouse Select:** 파라미터 셀을 더블 클릭하십시오. Variable Combo-Box 가 나타날 것입니다. 현 변수나 상수는 편집 필드에 디스플레이될 것입니다.

2. 대체 변수의 이름을 입력하거나 상수를 입력하십시오. 받아들이려면 Enter 키를 누르십시오.

# 펑션 블록 파라미터에서 상수 할당하기

상수는 래더 로직에서 펑션과 파라미터에 할당될 수 있습니다. 상수값은 상수값을 사용할 때마다 요소의 기본 포맷에서 모니터링되는 동안 디스플레이 됩니다.

## 키보드를 이용해 상수 할당하기:

1. 상수를 할당하기 원하는 파라미터에 커서가 위치할 수 있도록 방향키를 사용하십시오.
2. 상수값을 입력하고 Enter 키를 누르십시오. 만약 상수가 펑션 블록에 유효하지 않다면 에러 메시지가 나타날 것입니다.

## 마우스를 이용해 할당하기:

1. 활성화된 펑션을 취소하기 위해서는 방향 버튼  을 누르십시오. 변수를 할당하기 원하는 파라미터 셀을 더블 클릭하십시오. Variable Combo-Box가 나타날 것입니다.
2. 상수값을 입력하고 Enter 키를 누르십시오. 만약 상수가 펑션 블록에 유효하지 않다면 에러 메시지가 나타날 것입니다.

## 상수에 관한 규칙

상수값은 오퍼랜드 타입의 용량을 초과해서는 안되며 Function Block 타입을 지닌 일관된 포맷에 맞게 입력해야 합니다:

| Format | Prefix | Example |
|---|---|---|
| Decimal (default) | None | 43 |
| Real | None | 43.72 |
| Scientific | None | 6.93e4 |
| Binary | 2# | 2#11110000 |
| Octal | 8# | 8#730 |
| Hexadecimal | 16# | 16#dc4b |

# 래더 로직 편집하기

일단 로직이 LD 프로그램에 존재한다면, 로직 부분을 편집, 복사나 삭제할 필요가 있습니다. 이 섹션은 LD Editor 에서 로직을 선택하는 방법과 선택된 로직을 삭제, 복사 및 붙여넣기하는 방법을 기술합니다

## 로직 선택하기

LD 로직에서 편집을 수행하기 전에 작업하려고 하는 로직을 선택해야 합니다. 이 섹션은 마우스와 키보드를 이용해 로직을 선택하는 방법을 기술합니다.

### 마우스를 이용해 Ladder Editor 에서 선택 설정하기:

1. 대체 모드가 래더 툴바에 있는 방향 버튼(Normal Pointer) 을 클릭하여 정지되었는지 확인하십시오. 
2. 셀이나 Rung 다음에 있는 Power Rail 에 마우스 포인터를 놓으십시오
3. 첫 번째 마우스 버튼을 누르고 있으십시오.
4. 원하는 셀로 선택한 것을 이동하거나 원하는 Rung 으로 Power Rail 을 따라가도록 마우스를 드래그하십시오.
5. Non-contiguous rungs 들을 선택하려면, Rung 를 선택하는 동안 Ctrl key 를 누르고 있으십시오. 원하는 Rungs 이 선택됩니다.

### 키보드를 이용해 Ladder Editor 에서 선택 설정하기:

1. Esc 를 눌러 대체 모드를 정지되었는지 확인하십시오.
2. 셀을 선택하십시오:
   - 싱글 셀을 선택하려면 원하는 셀을 찾은 후 스페이스바를 누르십시오.
   - 셀 그룹을 선택하려면 셀을 찾은 다음 원하는 영역으로 선택물을 이동시키기 위해 화살표 키를 이용하는 동안 Shift key 를 누르고 있으십시오.
   - 싱글 셀을 선택하려면 원하는 Power Rail 을 네비게이션하고 Ctrl + 스페이스바를 누르십시오.
   - 인접한 Rung 들의 그룹을 선택하려면 Power Rail 을 네비게이션 하십시오. 모든 Rung 이 선택됩니다. 보다 많은 Rung 들을 선택하려면 인접한 Rung 으로 선택물을 이동시키기 위해 Shift key 를 누르고 상하 방향키를 사용하십시오.
   - 인접하지 않은 Rung 들을 선택하려면 Single Rung 을 선택하고 Ctrl key 를 누르면서 다른 Rung 을 네비게이션하고 스페이스바를 누르십시오. Rung 들이 선택됩니다.

4

# LD Editor에서 자르기, 복사, 붙여넣기 및 삭제하기

LD Editor에서 Variable Declaration Table에서 Rung이나 셀을 포함하고 있는 변수를 제거하지 않고 래더의 한 부분에서 다른 부분으로 그리고 한 블록에서 다른 블록으로 Rung이나 셀을 자르고 복사할 수 있습니다.

또한 선택한 Rung들이나 셀들을 삭제할 수 있습니다.

로직은 LD 블록에서 IL 블록으로 붙여넣기 할 수 있지만 LD 로직을 VersaPro가 아닌 애플리케이션으로 붙여넣기를 할 수 없습니다. 그러나 비트맵 이미지와 같은 래더 로직은 복사할 수 있습니다.

VDT에서 LD Editor로 변수를 붙여넣기 할 수 있습니다.

이 섹션은 LD Editor window에서 자르기, 복사, 붙여넣기 및 드래그&드롭을 수행할 수 있는 다양한 방법을 기술합니다.

## LD Editor에서 자르기와 복사하기:

1. 편집하고자 하는 LD 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. 편집 메뉴나 오른쪽 마우스 메뉴에서 Cut이나 Copy를 선택하고 표준 툴바에서 Cut  이나 Copy  button을 클릭하거나 Ctrl+C(Copy)나 Ctrl+X(Cut)를 누르십시오. 선택된 정보는 클립보드에 자르기나 복사됩니다.

## Drag and Drop을 이용해 LD Editor에서 자르기와 복사하기:

1. 편집하고자 하는 LD 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. **자르기 위해** (drag and drop 를 이용해), 마우스 화살표를 선택된 로직에 위치시켜놓고 Down 을 누르고 첫번째 마우스 버튼을 누르십시오. **복사하기 위해** (drag and drop 를 이용해) 마우스 화살표를 선택된 로직에 위치시켜놓고 CTRL key 를 계속해서 누른 후 첫번째 마우스 버튼을 누르십시오.
3. 적절한 지역으로 로직을 이동시키십시오. 선택물을 드롭하기 위해 마우스 버튼이나 Ctrl key 를 놓으십시오.

## LD Editor에서 bitmap로 복사하기:

1. 편집하고자 하는 LD 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. 편집메뉴를 선택하고 Copy As -> Bitmap 을 선택하십시오.
3. 선택된 정보는 Bitmap 이미지로 클립보드에 복사됩니다. 이 이미지는 비트맵 이미지를 인식하는 애플리케이션에 붙여넣기를 할 수 있습니다.

### LD Editor 에서 붙여넣기:

1. 편집하고자 하는 LD 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. 편집 메뉴를 선택하여 Paste  를 선택하시고 Paste 툴바 버튼을 클릭하거나 Ctrl+V를 누르십시오.
3. 요소들은 래더에 붙여넣을 수 있습니다. 붙여넣기는 붙여넣는 곳에서 이미 존재하는 요소를 겹쳐 쓸 수 있습니다. 기존 코멘트, 라벨 또는 MCR 들은 이 요소들을 수용하기 위해 시프트 다운 됩니다.

### 하나 또는 하나 이상 삭제하기:

1. LD Editor 에서 행이나 행 그룹, 셀이나 사각형 형태의 셀 그룹 및 하나 또는 하나 이상의 Rung 을 선택하십시오.
2. 편집 메뉴를 선택하여 Delete 를 선택하고 툴바에서 Delete button  을 클릭하거나 Delete key 를 누릅니다. 선택된 부분이 삭제됩니다. 삭제된 로직에서 변수는 VDT 에 남아 있습니다.

**Note:** 자르기/복사/붙여넣기/삭제는 편집 메뉴를 선택하여 Undo 를 선택하거나 Undo 툴바 버튼을 클릭하여 실행 취소할 수 있습니다.

## LD Editor에서 빈 행 삽입하기

LD Editor 에서 빈 행을 삽입하려면 다음 단계를 수행하십시오.

1. 편집하려는 블록을 선택하십시오.
2. 빈 행을 삽입하려는 행 위에 명령어를 선택하십시오.
3. 삽입 메뉴를 선택하여 Row 를 선택하고 Insert key 를 누르거나 오른쪽 마우스 버튼을 클릭하여 Insert Row 를 선택하십시오. 빈 행은 선택된 행 위에 삽입됩니다. 다음의 모든 행은 한 행 아래로 내려갑니다. 삽입 포인트 위에 새로운 행에 있는 셀은 선택됩니다. 첫 번째 Rung 의 첫 번째 행이 선택되면 새로운 행은 이전 Rung 에 추가됩니다. 첫 번째 Rung 의 첫 번째 행이 선택되면 새로운 Rung 은 첫 번째 Rung 에 생성됩니다.

## 다음코일로 가기:

ILD Editor 에서 Contact 이나 Coil 을 선택합니다.

편집 메뉴나 CSM(or press Alt + F3.)에서 Go to the Next Coil 를 선택하십시오

커서는 시작한 Contact 이나 Coil 과 같은 이름을 가진 다음 Coil 로 갈 것입니다.

Chapter 5

# Instruction List Editor 에서 작업하기

Instruction List Editor window 는 Instruction List 에서 생성된 프로그램과 블록을 생성, 편집과 디스플레이 및 모니터할 수 있게 해줍니다. 이 장은 IL Editor window 와 로직을 생성하고 편집하는 방법과 IL Editor window 를 볼 수 있도록 만드는 방법을 소개하고 있습니다.

## 목차

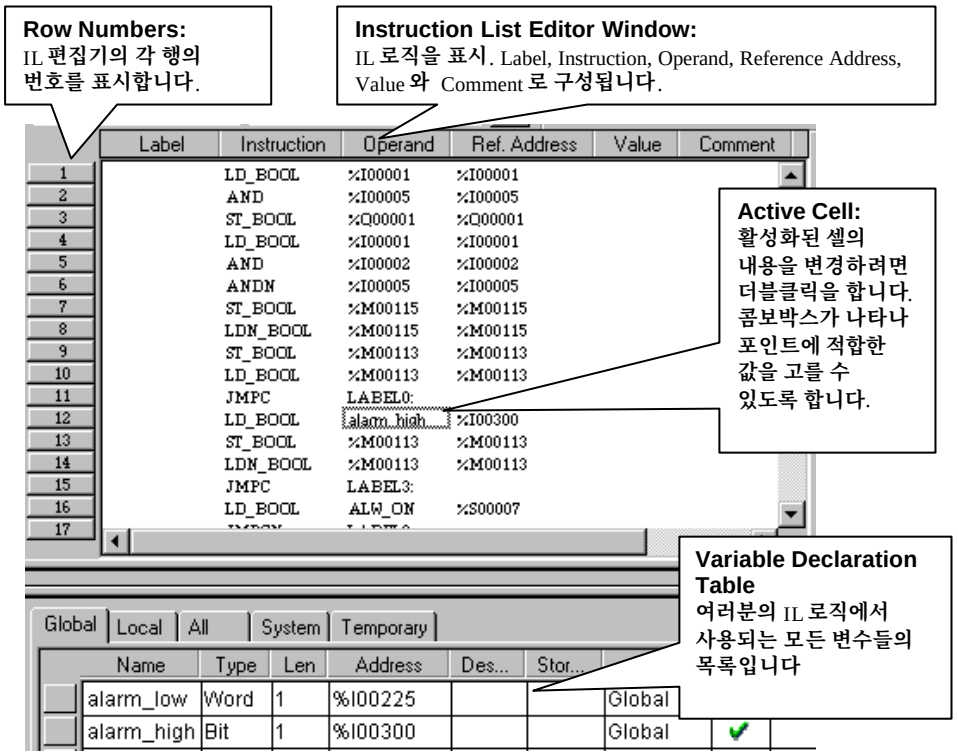
이 장은 다음과 같은 정보를 제공합니다:

- IL Editor 접근 방법
- IL Editor Window 개요
- IL Editor Window display 만드는 방법
- 로직을 입력하고 편집하는 방법
- Instruction List Operation 에 대한 세부사항

5

# Instruction List Editor 개요

Instruction List Editor는 Instruction List 프로그래밍 언어를 이용해 프로그램을 생성하는데 사용됩니다. Instruction List는 PLC에서 수행된 동작을 표현하는 일련의 명령어로 구성된 텍스트 기반 언어입니다.



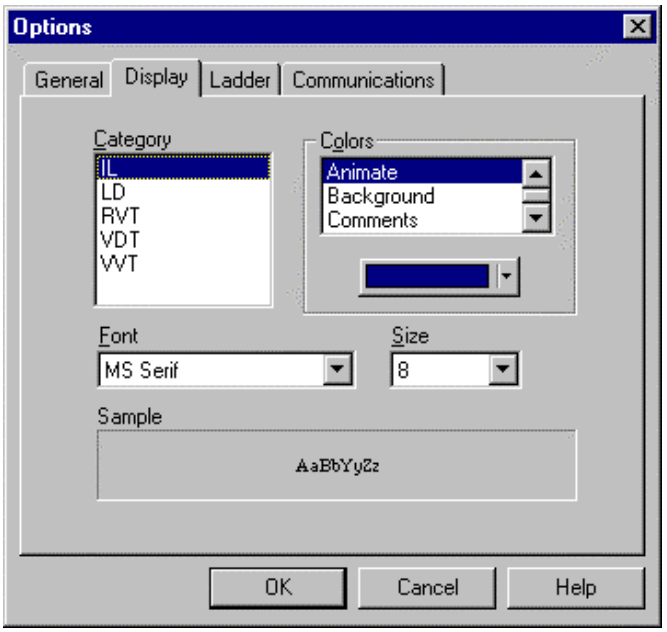

IL Editor는 다음 컬럼/필드를 포함하고 있습니다:

- **Label:** JUMP 명령어에 의해 참조될 수 있는 라벨을 디스플레이합니다.
- **Instruction:** 펑션 이름을 디스플레이합니다. 명령어는 명령어 필드를 클릭하고 Drop-down 된 리스트에서 입력하거나 선택함으로써 입력됩니다.
- **Operand:** 명령어에 의해 사용되는 변수와 상수를 디스플레이합니다. 오퍼랜드는 필드를 클릭하고 Drop-down 된 리스트에서 입력하거나 선택함으로써 입력됩니다. 이 리스트는 이 명령어 타입에 어울리는 변수만을 디스플레이합니다.
- **Reference Address**: 오퍼랜드의 메모리 위치를 디스플레이합니다.
- **Value:** PLC에서 오퍼랜드의 실제 값.
- **Comment:** 명령어에 대한 옵션 코멘트를 디스플레이합니다.

## IL Editor Display 만들기

IL Editor 를 위한 폰트 선택과 컬러를 만들 수 있습니다. 디스플레이를 만들기 위해 다음 단계를 수행하십시오:

1. Folder Browser 에서, 도구 메뉴를 선택하여 옵션을 선택하십시오. Options dialog box 가 나타날 것입니다. Display tab 을 선택하십시오.




2. 폰트를 변경할 수 있고 온라인 값의 디스플레이에 사용되는 색상, 백그라운드 디스플레이 색상 및 코멘트를 디스플레이하는데 사용되는 색상을 변경할 수 있습니다.
3. 원하는 변경사항을 작성하십시오. OK 를 클릭하십시오.

5

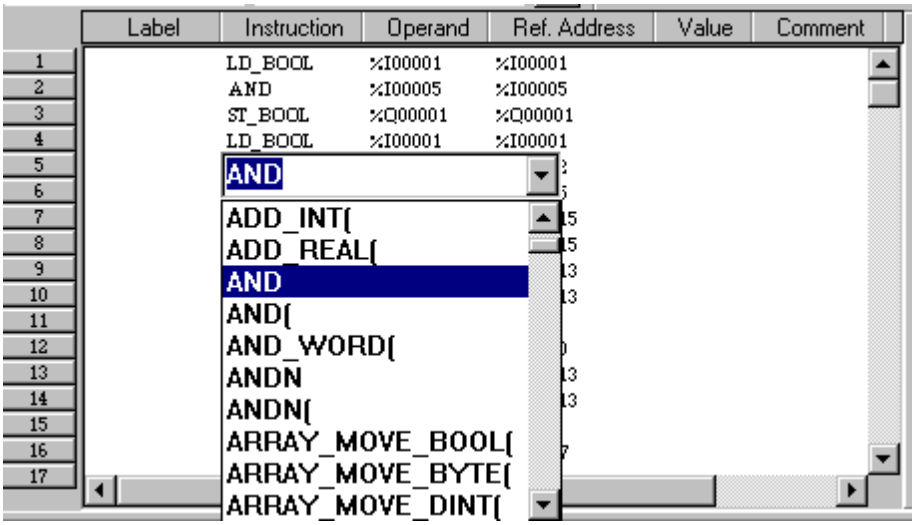
# 명령어 및 명령어 파라미터 입력하기

이 섹션은 IL 로직을 입력하고 수정하는 방법을 기술합니다. 명령어는 Boolean 로직을 수행하고 타이머, 수학 기능 및 기타 기능과 같은 기능을 실행하는 프로그램 요소로 구성되어 있습니다. 변수와 상수는 명령어를 위한 오퍼랜드로 할당됩니다.

## 명령어 선택하기

명령어는 IL Editor window 에서 직접 선택됩니다. 새로운 명령어를 추가하거나 기존 명령어를 변경하기 위해 다음 단계를 수행하십시오.

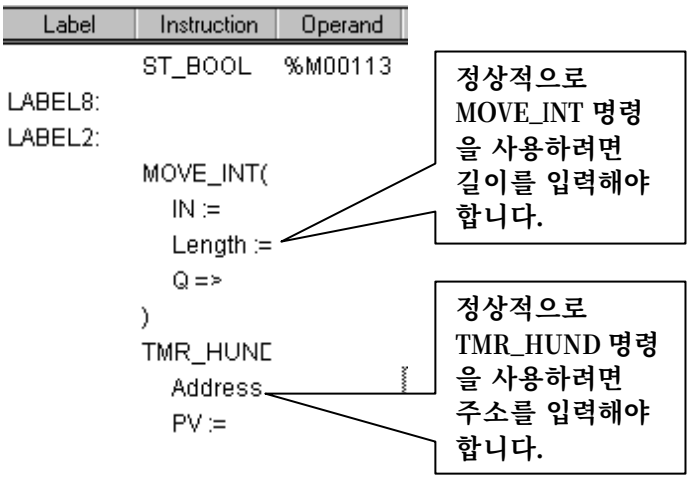
1. 새로운 명령어를 추가하거나 기존 명령어를 변경하고자 하는 위치를 선택하십시오.
2. 셀을 더블 클릭하십시오. 아래에서 보는 봐와 같이 Instruction Combo-Box가 나타날 것입니다.




3. 명령어를 선택하십시오. 펑션은 FUNCTION_NAME{으로 인식됩니다. 만약 펑션을 선택했다면 파라미터들은 펑션 블록을 프레임으로 하는 브라킷 사이에 나타날 것입니다.
4. 새로운 명령어는 IL 프로그램에 삽입됩니다.

## 어드레스와 길이 정보 완성하기

어떤 명령어는 운용하기 위해 명령어에 대한 길이(예, 이동 명령어)나 어드레스(예, 타이머 명령어)를 제공되기 원합니다. 이 정보를 요구하는 명령어는 아래 나타난 것처럼 명령어 컬럼에서 길이나 어드레스 필드를 보여줄 것입니다. 적절하게 길이나 어드레스를 입력하십시오.




## 명령어 파라미터 완성하기

명령어를 프로그램에 추가한 후 각각의 명령어 오퍼랜드에 변수나 상수를 할당할 필요가 있습니다. 이 섹션은 다음 방법 중 하나로 명령어 오퍼랜드를 완성하는 방법을 서술합니다:

- 오퍼랜드에 기존 변수 할당하기
- 오퍼랜드에 대한 새로운 변수 생성하기
- 오퍼랜드에 상수 값 할당하기

## 명령어 오퍼랜드에 변수 할당하기

변수는 명령어 오퍼랜드에 할당될 수 있습니다. 변수는 Local Scope 나 Global Scope 일 수 있습니다. 변수에 대한 기본 스코프는 Global Scope 입니다.

기존 변수를 할당하거나 명령어 오퍼랜드에 대한 새로운 변수를 생성할 수 있습니다. 이 섹션은 마우스와 키보드를 이용해 파라미터에 변수를 할당하는 방법을 서술합니다.

### 키보드를 이용해 기존 변수 할당하기:

1. 변수를 할당하기 원하는 파라미터에 커서를 위치하도록 방향키를 사용하십시오.

2. 다음과 같은 변수를 입력하거나 선택하십시오:

   - 정의된 변수 리스트에서 변수를 선택하고 Enter 키를 누르십시오. Variable Combo-Box 가 나타날 것입니다. 변수를 디스플레이하기 위해 상하 키를 사용하십시오. 사용하려고 하는 변수를 찾으면 받아 들이 위해 Enter 키를 누르십시오.
   - 이름으로 변수를 선택하려면 변수 이름을 입력하십시오. 받아 들이기 위해 Enter 키를 누르십시오.

## 마우스를 이용해 기존 변수를 할당하기:

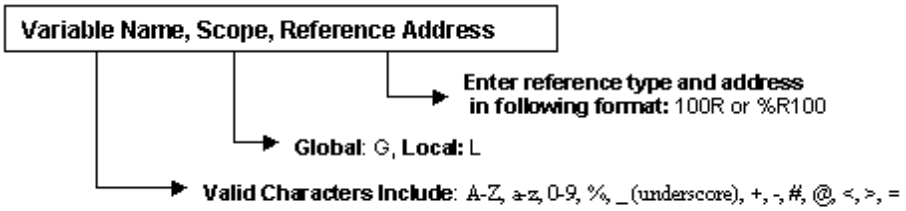
1. 변수를 할당하기 원하는 오퍼랜드를 더블 클릭하십시오. Variable Combo-Box 가 나타날 것입니다.
2. 변수 리스트를 확장하려면 Combo-Box 에서 화살표를 클릭하십시오. 리스트를 스크롤하기 위해 Slide Control 을 사용하십시오.
3. 리스트에서 변수를 선택하십시오. Enter 키를 누르거나 받아들이기 위해 오퍼랜드 영역 바깥을 클릭하십시오.

## 명령어 오퍼랜드에 대한 새로운 변수 생성하기

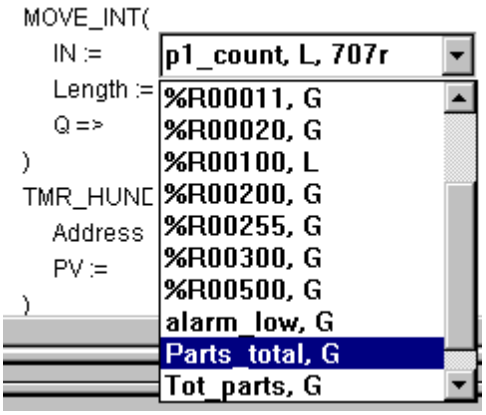
1. 키보드와 마우스를 이용하여 명령어 오퍼랜드를 선택하십시오.

   - **Keyboard Select:** 변수를 할당하고자 하는 파라미터에 커서를 갖다 놓기 위해 방향 키를 사용하십시오. Enter 를 누르십시오. Variable Combo-Box 가 나타날 것입니다.
   - **Mouse Select:** 파라미터 셀을 더블 클릭하십시오. Variable Combo-Box 가 나타날 것입니다.

2. 다음 가이드 라인을 이용해 편집 박스에 변수 이름, 레퍼런스 어드레스와 스코프(옵션)를 입력하십시오(필드는 “,”에 의해 분리해야 합니다.)




- 다음 예는 MOVE_INT Function 의 Parameter IN 에 대한 변수 “p1_count”의 생성을 보여주고 있습니다.




3. 변수를 받아들이기 위해 Enter key 를 누르십시오. – 또는 활동중인 셀을 클릭 오프하기 위해 마우스를 사용하십시오.

## 명령어 오퍼랜드에 상수 할당하기

상수는 명령어 오퍼랜드에 할당될 수 있습니다.

### 키보드 입력으로 상수 할당하기:

1. 상수를 할당하고자 하는 명령어 오퍼랜드에 위치하려면 방향키를 사용하십시오.
2. 상수값을 입력하고 Enter key 를 누르십시오.

### 상수에 대한 규칙

상수값은 오퍼랜드의 용량을 초과할 수 없으며, 명령어 타입을 지닌 일정한 포맷에 입력해야 합니다.

| Format | Prefix | Example |
|---|---|---|
| Decimal (default) | None | 43 |
| Real | None | 43.72 |
| Scientific | None | 6.93e4 |
| Binary | 2# | 2#11110000 |
| Octal | 8# | 8#730 |

## IL Editor에서 자르기, 복사, 붙여넣기 및 삭제하기

IL Editor 에서 Variable Declaration Table 에서 Rung 이나 셀을 포함하고 있는 변수를 제거하지 않고 명령어 리스트의 한 부분에서 다른 부분으로 그리고 한 블록에서 다른 블록으로 Rung 이나 셀을 자르고 복사할 수 있습니다.

또한 선택한 rungs 이나 셀들을 삭제할 수 있습니다.

이 섹션은 IL Editor window 에서 자르기, 복사, 붙여넣기 및 드래그&드롭을 수행할 수 있는 다양한 방법을 기술합니다.

## IL Editor 에서 자르기와 복사하기:

1. 편집하고자 하는 IL 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. 편집 메뉴나 오른쪽 마우스 메뉴에서 Cut 이나 Copy 를 선택하고 표준 툴바에서 Cut  이나 Copy  button 을 클릭하거나 Ctrl+C(Copy)나 Ctrl+X(Cut)를 누르십시오. 선택된 정보는 클립보드에 자르기나 복사됩니다.

**Note:** 텍스트로 Multi-row Instruction 의 한 부분을 복사할 수 있습니다. 다만 Multi-row Instruction 의 한 행을 복사한다면, VerasPro 로 붙여넣기를 하지 못할 것입니다.

## Drag and Drop 을 이용해 IL Editor 에서 자르기와 복사하기:

1. 편집하고자 하는 LD 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. **자르기 위해** (drag and drop 를 이용해), 마우스 화살표를 선택된 로직에 위치시켜놓고 Down 을 누르고 첫번째 마우스 버튼을 누르십시오. **복사하기 위해** (drag and drop 를 이용해) 마우스 화살표를 선택된 로직에 위치시켜놓고 CTRL key 를 계속해서 누른 후 첫번째 마우스 버튼을 누르십시오.

- 적절한 지역으로 로직을 이동시키십시오. 선택물을 드롭하기 위해 마우스 버튼이나 Ctrl key 를 놓으십시오.

### 다른 어플리케이션에 탭으로 구분된 문서로 복사하기:

1. 편집하고자 하는 IL 블록을 여십시오. 복사하고자 하는 행과 셀을 선택하십시오.
2. 편집 메뉴를 선택하여 Copy as Text 를 선택하십시오.
3. 선택된 정보는 탭으로 구분된 문서로 복사되고 , 탭으로 구분된 문서를 사용할 수 있는 모든 애플리케이션에서 붙여넣기 할 수 있습니다.

5

# 명령어 리스트 오퍼랜드

IL 프로그래밍 언어는 어큐뮬레이터를 기반으로 한 언어입니다. IL 프로그램에 의해 실행되는 명령어는 PLC 메모리에 위치하고 있는 어큐뮬레이터를 수정하거나 사용합니다. 어큐뮬레이터는 연산자나 Function call 에 의해 설정되어 있는 형식을 가지고 있습니다.

IL 프로그램에서 사용될 수 있는 연산자는 다음 표에 열거했습니다. IL 이 어큐뮬레이터를 기반으로 한 프로그래밍 언어이기 때문에 연산자는 어큐뮬레이터에서 직접 작업합니다. 비교 연산자는 동작 결과에 따라 Boolean (1 또는 0) 결과를 가지고 있습니다. IL 명령어에 대한 오퍼랜드는 변수, 레퍼런스 또는 상수의 형태로 될 수 있습니다. 연산자와 펑션은 IL 프로그램에서 충돌할 때 무조건 실행됩니다. 즉 명령어는 이전 명령어에 대한 결과에 관계없이 실행됩니다.

LD* (Load) 연산자는 오퍼랜드에 의해 명시된 변수와 상수값에 어큐뮬레이터의 컨텐츠를 설정하는데 사용됩니다. 또한 LD 연산자는 어큐뮬레이터의 타입을 설정합니다. ST* (Store) 연산자는 어큐뮬레이터의 컨텐츠에 오퍼랜드에 명시된 변수값을 설정하는데 사용됩니다. 연산자의 remainder 는 어큐뮬레이터의 컨텐츠를 조작하거나 어큐뮬레이터 값을 기반으로 한 동작을 수행합니다. 수식과 관계 연산에 있어서 어큐뮬레이터의 타입은 정수가 되어야 합니다

| Operator | Operand | Description / Affect upon Accumulator |
|---|---|---|
| LD_BOOL | Discrete Variable or Ref Address | Accumulator := Boolean Variable<br>어큐뮬레이터에서 Boolean 변수의 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Boolean 로 설정. |
| LDN_BOOL | Discrete Variable or Ref Address | Accumulator := NOT Variable<br>어큐뮬레이터에서 Boolean 변수의 네비게이션하는 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Boolean 로 설정. |
| LD_INT | Variable, Ref Address, or constant | Accumulator := Integer Variable<br>어큐뮬레이터에서 Integer 변수의 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Integer 로 설정. |
| LD_ENO | None | Accumulator := Boolean ENO<br>어큐뮬레이터에서 마지막 펑션이나 펑션 블록 Call 로부터 ENO(Enable Output)의 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Boolean 로 설정. |
| ST_BOOL | Discrete Variable or Ref Address | Variable := Boolean Accumulator<br>Boolean 변수에서 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| STN_BOOL | Discrete Variable or Ref Address | Variable := NOT Boolean Accumulator<br>Boolean variable 에 Boolean Accumulator 의 네비게이션하는 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| ST_INT | Variable or Ref Address | Variable := Integer Accumulator |

| | | |
|---|---|---|
| ST_DINT<br>ST_REAL<br>ST_WORD | | Variable := Double Integer Accumulator<br>Variable := Real Accumulator<br>Variable := Word Accumulator<br>변수에 Integer, Double Integer, Real, or Word Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨. |
| S | Discrete Variable or Ref Address | Set Coil Variable := Boolean Accumulator<br>Set Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| R | Discrete Variable or Ref Address | Reset Coil Variable := Boolean Accumulator<br>Reset Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| NT | Discrete Variable or Ref Address | Negative Transition Variable := Boolean Accumulator<br>Negative Transition Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기 .<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| PT | Discrete Variable or Ref Address | Positive Transition Variable := Boolean Accumulator<br>Positive Transition Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| NOT | None | Accumulator := NOT Accumulator<br>Invert the Accumulator 의 Boolean 상태 반전하기.<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| AND<br>ANDN<br>AND( | Discrete Variable or Ref Address | Accumulator := Accumulator AND Boolean Variable<br>Accumulator := Accumulator AND NOT Boolean Variable<br>Accumulator := Accumulator AND Boolean expression<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| OR<br>ORN<br>OR( | Discrete Variable or Ref Address | Accumulator := Accumulator OR Boolean Variable<br>Accumulator := Accumulator OR NOT Boolean Variable<br>Accumulator := Accumulator OR Boolean expression<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| XOR<br>XORN | Discrete Variable or Ref Address | Accumulator := Accumulator XOR Boolean Variable<br>Accumulator := Accumulator XOR NOT Boolean Variable<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| ADD | Variable, Ref Address, or Constant | Accumulator := Accumulator + operand<br>Accumulator Type unchanged (Integer). |
| SUB | Variable, Ref Address, or Constant | Accumulator := Accumulator – operand<br>Accumulator Type unchanged (Integer). |
| MUL | Variable, Ref Address, or Constant | Accumulator := Accumulator * operand<br>Accumulator Type unchanged (Integer). |
| DIV | Variable, Ref Address, or Constant | Accumulator := Accumulator / operand<br>Accumulator Type unchanged (Integer). |
| MOD | Variable, Ref Address, or Constant | Accumulator := Accumulator MOD operand<br>Accumulator Type unchanged (Integer). |
| GT | Variable, Ref Address, or Constant | Accumulator := (Accumulator > operand)<br>Accumulator Type set to Boolean. |
| GE | Variable, Ref Address, or | Accumulator := (Accumulator >= operand) |

| | | |
|---|---|---|
| | Constant | Accumulator Type set to Boolean. |
| EQ | Variable, Ref Address, or Constant | Accumulator := (Accumulator = operand)<br>Accumulator Type set to Boolean. |
| NE | Variable, Ref Address, or Constant | Accumulator := (Accumulator != operand)<br>Accumulator Type set to Boolean. |
| LE | Variable, Ref Address, or Constant | Accumulator := (Accumulator <= operand)<br>Accumulator Type set to Boolean. |
| LT | Variable, Ref Address, or Constant | Accumulator := (Accumulator < operand)<br>Accumulator Type set to Boolean. |
| RET<br>RETC<br>RETCN | None | 블록으로 무조건 돌아가기<br>Boolean Accumulator 이 1 의 값을 갖을 때 블록으로 돌아가기<br>Boolean Accumulator 이 0 의 값을 갖을 때 블록으로 돌아가기 |
| JMP<br>JMPC<br>JMPCN | Label | 라벨로 무조건 건너뛰기<br>Boolean Accumulator 이 1 의 값을 갖을 때 라벨로 점프하기<br>Boolean Accumulator 이 0 의 값을 갖을 때 라벨로 점프하기 |
| CAL<br>CALC<br>CALCN | Subroutine Name | 무조건 블록 부르기<br>Boolean Accumulator 이 1 의 값을 갖을 때 블록 부르기<br>Boolean Accumulator 이 0 의 값을 갖을 때 블록 부르기 |
| FUNC_NAME(<br>IN1:=<br>IN2:=<br>Q2=><br>) | Variable, Ref Address, or Constant, dependent upon type of the function parameters. | 내장된 PLC Function 부르기.<br>Accumulator 값과 타입은 출력 파라미터 Q1 의 값과 타입으로 설정됨. |
| ) | | End of Boolean nested operation.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |

# 상수

IL Editor 는 다음과 같은 오퍼랜드로 사용할 수 있는 상수를 지원합니다: Integer, Real, Real with exponents, Base 16(16#xxxx), Base 8(8#xxx)와 Binary (2#xxxxxxxx). Boolean 연산자로 사용하는 오퍼랜드를 위해 1 이나 0 의 상수를 사용하려면, ALW_ON(1 또는 TRUE)과 ALW_OFF(0 또는 FALSE) 시스템 변수를 사용해야 합니다.

# 펑션

펑션은 형식적인 입력 리스트를 사용하고자 합니다. 또한 펑션은 이전의 명령어 결과를 무시하고 무조건 실행합니다. 펑션은 보통 결과를 Accumulator 에 배열하고 Accumulator 의 타입은 펑션에 의해 결정됩니다.

```
                              "AddResultGt100 := ((SQRT_INT(Input1) > 100)
SQRT_INT(
   IN:=     Input1       "Input to Function call
)
GT          100
ST_BOOL     AddResultGt100 "Set to TRUE if result greater than 100
```

# 펑션 블록

펑션 블록 또한 형식적인 입력 리스트를 사용합니다. 또한 평션 블록은 이전의 명령어 결과를 무시하고 무조건 실행합니다. 펑션 블록은 보통 결과를 Accumulator 에 배열하고 Accumulator 의 타입은 펑션 블록에 의해 결정됩니다. 펑션 블록의 어드레스는 입력 파라미터입니다.

```
UPCTR(
     Address :=    %R00022
     R :=          %M00012
     PV :=         1
)
```

# PLC 내장형 펑션과 펑션 블록

다음 PLC 내장형 펑션과 펑션 블록은 IL 프로그램에 의해 부를 수 있습니다. Resulting Accumulator 의 값과 형태는 블록과 펑션 블록마다 명시됩니다. ENO 컬럼에서 YES 는 펑션이나 펑션 ENO(Enable Output) 변수를 설정한다는 것을 가리킵니다. 테이블 또한 펑션이나 펑션 블록이 어큐뮬레이터에게 펑션 블록 앞에 LD Bool 명령어를 삽입함으로써 초기화되기를 요구합니다.

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| 타이머와 카운터(펑션 블록) | | | |
| ONDTR_TENTHS | Yes | Yes | N/A |
| ONDTR_HUNDS | Yes | Yes | N/A |
| ONDTR_THOUS | Yes | Yes | N/A |
| TMR_TENTHS | Yes | Yes | N/A |
| TMR_HUNDS | Yes | Yes | N/A |
| TMR_THOUS | Yes | Yes | N/A |
| OFDT_TENTHS | Yes | Yes | N/A |
| OFDT_HUNDS | Yes | Yes | N/A |
| OFDT_THOUS | Yes | Yes | N/A |
| UPCTR | Yes | | N/A |
| DNCTR | Yes | | N/A |
| Standard Math (펑션) | | | |
| ADD_INT | Yes | | Output Parameter Q, Integer |
| ADD_DINT | Yes | | Output Parameter Q, Double Integer |
| ADD_REAL | Yes | | Output Parameter Q, Real |
| SUB_INT | Yes | | Output Parameter Q, Integer |
| SUB_DINT | Yes | | Output Parameter Q, Double Integer |
| SUB_REAL | Yes | | Output Parameter Q, Real |
| MUL_INT | Yes | | Output Parameter Q, Integer |
| MUL_DINT | Yes | | Output Parameter Q, Double Integer |
| MUL_REAL | Yes | | Output Parameter Q, Real |
| DIV_INT | Yes | | Output Parameter Q, Integer |
| DIV_DINT | Yes | | Output Parameter Q, Double Integer |
| DIV_REAL | Yes | | Output Parameter Q, Real |
| ADD_INT | Yes | | Output Parameter Q, Integer |
| ADD_DINT | Yes | | Output Parameter Q, Double Integer |
| ADD_REAL | Yes | | Output Parameter Q, Real |
| MOD_INT | Yes | | Output Parameter Q, Integer |
| MOD_DINT | Yes | | Output Parameter Q, Double Integer |
| SQRT_INT | Yes | | Output Parameter Q, Integer |
| SQRT_DINT | Yes | | Output Parameter Q, Double Integer |
| SQRT_REAL | Yes | | Output Parameter Q, Real |

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| Trig Functions (Functions) | | | |
| SIN | Yes | | Output Parameter Q, Real |
| COS | Yes | | Output Parameter Q, Real |
| TAN | Yes | | Output Parameter Q, Real |
| ASIN | Yes | | Output Parameter Q, Real |
| ACOS | Yes | | Output Parameter Q, Real |
| ATAN | Yes | | Output Parameter Q, Real |
| Logarithmic/Exponential Functions (Functions) | Yes | | Output Parameter Q, Real |
| LOG | Yes | | Output Parameter Q, Real |
| LN | Yes | | Output Parameter Q, Real |
| EXP | Yes | | Output Parameter Q, Real |
| EXPT | | | |
| Radian Conversion Functions (Functions) | | | |
| RAD_2_DEG | Yes | | Output Parameter Q, Real |
| DEG_2_RAD | Yes | | Output Parameter Q, Real |
| Relational Functions (Functions) | | | |
| EQ_INT | No | | Output Parameter Q, Boolean |
| EQ_DINT | No | | Output Parameter Q, Boolean |
| EQ_REAL | No | | Output Parameter Q, Boolean |
| NE_INT | No | | Output Parameter Q, Boolean |
| NE_DINT | No | | Output Parameter Q, Boolean |
| NE_REAL | No | | Output Parameter Q, Boolean |
| GT_INT | No | | Output Parameter Q, Boolean |
| GT_DINT | No | | Output Parameter Q, Boolean |
| GT_REAL | No | | Output Parameter Q, Boolean |
| GE_INT | No | | Output Parameter Q, Boolean |
| GE_DINT | No | | Output Parameter Q, Boolean |
| GE_REAL | No | | Output Parameter Q, Boolean |
| LT_INT | No | | Output Parameter Q, Boolean |
| LT_DINT | No | | Output Parameter Q, Boolean |
| LT_REAL | No | | Output Parameter Q, Boolean |
| LE_INT | No | | Output Parameter Q, Boolean |
| LE_DINT | No | | Output Parameter Q, Boolean |
| LE_REAL | No | | Output Parameter Q, Boolean |
| | No | | Output Parameter Q, Boolean |
| | No | | Output Parameter Q, Boolean |
| Relational Functions (Functions) | | | |
| RANGE_INT | No | | Output Parameter Q, Boolean |
| RANGE_DINT | No | | Output Parameter Q, Boolean |
| RANGE_WORD | No | | Output Parameter Q, Boolean |
| Bit Functions (Functions) | | | |
| AND_WORD | Yes | | Output Parameter Q, Word |
| OR_WORD | Yes | | Output Parameter Q, Word |
| XOR_WORD | Yes | | Output Parameter Q, Word |

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| NOT_WORD | Yes | | Output Parameter Q, Word |
| Bit Functions (Functions) | | | |
| SHL_WORD | No | | Output Parameter B2, Boolean |
| SHR_WORD | No | | Output Parameter B2, Boolean |
| ROL_WORD | Yes | | N/A |
| ROR_WORD | Yes | | N/A |
| BTST_WORD | No | | Output Parameter Q, Boolean |
| BSET_WORD | Yes | | N/A |
| BCLR_WORD | Yes | | N/A |
| BPOS_WORD | Yes | | Output Parameter POS, Integer |
| MSKCMP_WORD | No | | Output Parameter MC, Boolean |
| MSKCMP_DWORD | No | | Output Parameter MC, Boolean |
| Data Move Functions (Functions) | | | |
| MOVE_BOOL | Yes | | N/A |
| MOVE_INT | Yes | | N/A |
| MOVE_WORD | Yes | | N/A |
| MOVE_REAL | Yes | | N/A |
| Data Move Functions (Functions) | | | |
| BLKMOV_INT | Yes | | N/A |
| BLKMOV_WORD | Yes | | N/A |
| BLKMOV_REAL | Yes | | N/A |
| BLKCLR_WORD | Yes | | N/A |
| SHFR_BIT | Yes | Yes | N/A |
| SHFR_WORD | Yes | Yes | N/A |
| COMM_REQ | No | | Output Parameter FT, Boolean |
| Data Move Function Blocks (Functions) | | | |
| BIT_SEQ | Yes | Yes | N/A |
| Table Functions (Functions) | | | |
| ARRAY_MOVE_INT | Yes | | N/A |
| ARRAY_MOVE_DINT | Yes | | N/A |
| ARRAY_MOVE_BIT | Yes | | N/A |
| ARRAY_MOVE_BYTE | Yes | | N/A |
| ARRAY_MOVE_WORD | Yes | | N/A |
| SRCH_EQ_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_EQ_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_EQ_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_EQ_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_GE_INT | Yes | | Output Parameter FD, Boolean |

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| SRCH_GE_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_GE_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_GE_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_WORD | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| Conversion Functions (Functions) | | | |
| INT_TO_BCD4 | Yes | | Output Parameter Q, Word |
| BCD4_TO_INT | Yes | | Output Parameter Q, Integer |
| INT_TO_REAL | Yes | | Output Parameter Q, Integer |
| DINT_TO_REAL | Yes | | Output Parameter Q, Integer |
| BCD4_TO_REAL | Yes | | Output Parameter Q, Integer |
| WORD_TO_REAL | Yes | | Output Parameter Q, Integer |
| REAL_TO_WORD | Yes | | Output Parameter Q, Word |
| REAL_TO_INT | Yes | | Output Parameter Q, Integer |
| REAL_TO_DINT | Yes | | Output Parameter Q, Double Integer |
| Control Functions (Functions) | | | |
| DOIO | Yes | | N/A |
| SVC_REQ | Yes | | N/A |
| Control Function Blocks (Function Blocks) | | | |
| PID_IND | Yes | Yes | Output Parameter CV, Integer |
| PID_ISA | Yes | Yes | Output Parameter CV, Integer |
| New Series 90-30 (36x) Function Blocks (Function Blocks) | | | |
| SER | Yes | Yes | N/A |
| VersaMax Release 1.1 and later Function Blocks | | | |
| DRUM | Yes | Yes | Output Parameter Q, Word |
| SCALE_INT | Yes | | Output Parameter OUT, Integer |
| SCALE_WORD | Yes | | Output Parameter OUT, Word |

5

# 논리식 네스트 구성하기

사용자는 8 단위까지 **AND(** 와 **OR(** 연산자를 이용해 논리식을 네스트(내포)할 수 있습니다. 네스트된 논리식은 AND( 이나 OR ( 연산자로 시작되고 다음 명령어는 LD_BOOL 또는 LDN_BOOL 연산자가 되어야 합니다. 네스트된 논리식에서 사용할 수 있는 연산자는 다음과 같습니다. : S, R, ST_BOOL, STN_BOOL, NT, PT, NOT, AND*, OR*과 XOR*. 네스트된 논리식은 복잡한 Boolean 방정식의 값을 구하는데 사용됩니다. 다음은 네스트된 논리식의 예입니다:

```
LD_BOOL     %I00001     %I00001
AND(
LD_BOOL     %I00002     %I00002
OR          %T00010     %T00010
AND         %T00011     %T00011
OR(
LD_BOOL     %T00032     %T00032
OR          %T00033     %T00033
ST_BOOL     %M00077     %M00077
ANDN        %T00088     %T00088

)
AND         %T00088     %T00088
OR          %Q00067     %Q00067

)
ST_BOOL     %T00099     %T00099
```

컨트롤 명령어나 펑션 부르기는 네스트된 논리식에 넣을 수 없습니다.

# ENO 사용하기

펑션이나 펑션 블록이 실행될 때, ENO 출력은 펑션 블록의 성공적인 실행을 나타내는데 사용됩니다. 펑션을 실행할 때, 에러가 발생하면 ENO 출력을 설정하거나 재설정합니다. 사용자는 어큐뮬레이터에서 ENO 출력을 로드하기 위해 LD_ENO 연산자를 이용하여 ENO 의 상태를 점검한 다음 발생된 에러 값을 검사할 수 있습니다.

```
"Calculate the following expression
        %R00005 := (%R00001 + 100) * %R00077
        Set calcOK variable to 1 if calculation successful

        ADD_DINT(
          IN1 :=    %R00001     %R00001
          IN2 :=    100
        )
        ST_DINT     %R00003     %R00003
        LD_ENO                          "LD_ENO used to get ENO output of ADD_DINT
        JMPCN       error:
        MUL_DINT(
          IN1 :=    %R00003     %R00003
          IN2 :=    %R00077     %R00077
        )
        ST_DINT     %R00005     %R00005
        LD_ENO                          "LD_ENO used to get ENO output of MUL_DINT
        JMPCN       error:
        LD_BOOL     ALW_ON      %S00007
        ST_BOOL     calcOK      %Q00077 "calcOK := 1
        RET
error:
        LD_BOOL     ALW_OFF     %S00008
        ST_BOOL     calcOK      %Q00077 "calcOK := 0
        RET
```

5

# 임시 변수 정의하기

명령어 리스트를 프로그래밍할 때, 사용자는 IL Accumulator에 의해 사용될 수 있도록 메모리 위치를 정의해야 합니다. 사용자는 Boolean Accumulator로 사용할 수 있도록 %T, %M 또는 %Q 메모리에 8 개의 연속된 디지털 포인트를 정의해야합니다. 또한 the Word, Integer, Double Integer와 Real Accumulator로 사용할 수 있도록 %AI, %AQ 또는 %R 메모리에 2 워드의 레퍼런스 메모리를 정의해야 합니다. 사용자는 Folder Properties dialog 를 이용해 IL 임시 변수의 정의를 내립니다. 일단 임시 변수가 정의를 내리기 되면 이들 레퍼런스는 프로그램에서 직접 레퍼런스될 수 없습니다.

# 에러 점검하기

컴파일될 때, 다음과 같은 에러 점검은 IL 프로그램에서 수행됩니다.

1. Accumulator의 형태는 연산자 형태와 호환되어야합니다.
2. LD 명령어나 펑션은 부르기는 ST 명령어 전에 발생해야 합니다 (Accumulator는 값과 형태를 가지고 있습니다.)
3. 네스트된 식의 ()는 매치되어야 합니다.
4. 레퍼런스 형태는 연산자로 고쳐야 합니다. 예를 들어 데이터 타입은 bit 가 되어야 하기 때문에 “S %R00001”이 될 수 없습니다.
5. JMPs 와 MCRs 은 연결 라벨을 가지고 있어야 합니다.
6. 모든 변수는 정의되어야 합니다.
7. CALs 은 IL/LD Blocks 과 매치되어야합니다.
8. 레퍼런스 어드레스는 설치된 CPU 의 메모리 범위 내에 있어야합니다.
9. 펑션과 펑션 블록 파라미터는 정확한 메모리 형태와 값이 되어야 합니다.

Chapter 5

# Instruction List Editor 에서 작업하기

Instruction List Editor window 는 Instruction List 에서 생성된 프로그램과 블록을 생성, 편집과 디스플레이 및 모니터할 수 있게 해줍니다. 이 장은 IL Editor window 와 로직을 생성하고 편집하는 방법과 IL Editor window 를 볼 수 있도록 만드는 방법을 소개하고 있습니다.

## 목차

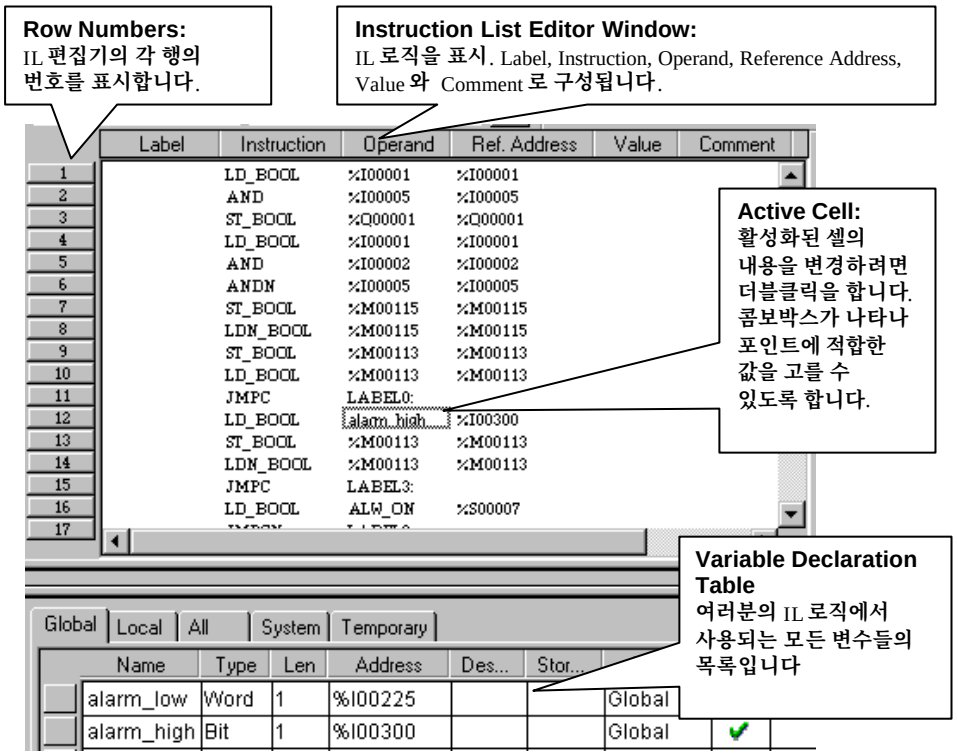
이 장은 다음과 같은 정보를 제공합니다:

- IL Editor 접근 방법
- IL Editor Window 개요
- IL Editor Window display 만드는 방법
- 로직을 입력하고 편집하는 방법
- Instruction List Operation 에 대한 세부사항

5

# Instruction List Editor 개요

Instruction List Editor는 Instruction List 프로그래밍 언어를 이용해 프로그램을 생성하는데 사용됩니다. Instruction List는 PLC에서 수행된 동작을 표현하는 일련의 명령어로 구성된 텍스트 기반 언어입니다.



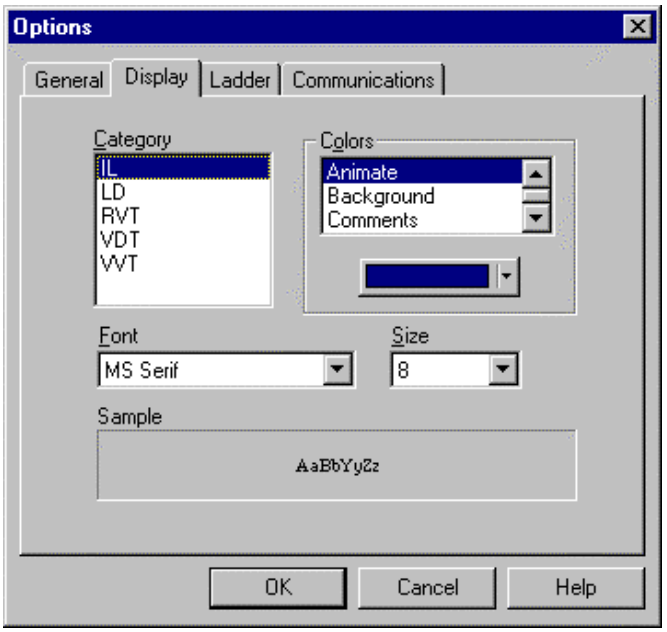

IL Editor는 다음 컬럼/필드를 포함하고 있습니다:

- **Label:** JUMP 명령어에 의해 참조될 수 있는 라벨을 디스플레이합니다.
- **Instruction:** 펑션 이름을 디스플레이합니다. 명령어는 명령어 필드를 클릭하고 Drop-down 된 리스트에서 입력하거나 선택함으로써 입력됩니다.
- **Operand:** 명령어에 의해 사용되는 변수와 상수를 디스플레이합니다. 오퍼랜드는 필드를 클릭하고 Drop-down 된 리스트에서 입력하거나 선택함으로써 입력됩니다. 이 리스트는 이 명령어 타입에 어울리는 변수만을 디스플레이합니다.
- **Reference Address**: 오퍼랜드의 메모리 위치를 디스플레이합니다.
- **Value:** PLC에서 오퍼랜드의 실제 값.
- **Comment:** 명령어에 대한 옵션 코멘트를 디스플레이합니다.

## IL Editor Display 만들기

IL Editor 를 위한 폰트 선택과 컬러를 만들 수 있습니다. 디스플레이를 만들기 위해 다음 단계를 수행하십시오:

1. Folder Browser 에서, 도구 메뉴를 선택하여 옵션을 선택하십시오. Options dialog box 가 나타날 것입니다. Display tab 을 선택하십시오.




2. 폰트를 변경할 수 있고 온라인 값의 디스플레이에 사용되는 색상, 백그라운드 디스플레이 색상 및 코멘트를 디스플레이하는데 사용되는 색상을 변경할 수 있습니다.
3. 원하는 변경사항을 작성하십시오. OK 를 클릭하십시오.

5

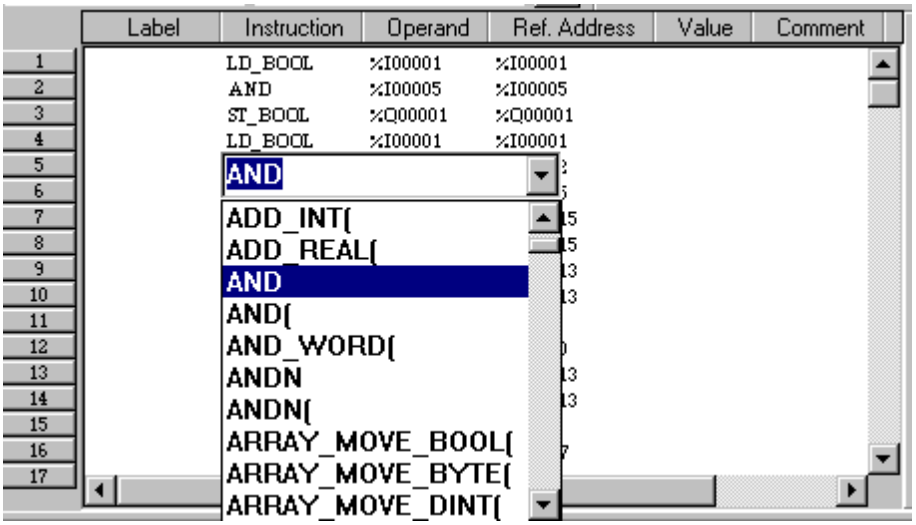
# 명령어 및 명령어 파라미터 입력하기

이 섹션은 IL 로직을 입력하고 수정하는 방법을 기술합니다. 명령어는 Boolean 로직을 수행하고 타이머, 수학 기능 및 기타 기능과 같은 기능을 실행하는 프로그램 요소로 구성되어 있습니다. 변수와 상수는 명령어를 위한 오퍼랜드로 할당됩니다.

## 명령어 선택하기

명령어는 IL Editor window 에서 직접 선택됩니다. 새로운 명령어를 추가하거나 기존 명령어를 변경하기 위해 다음 단계를 수행하십시오.

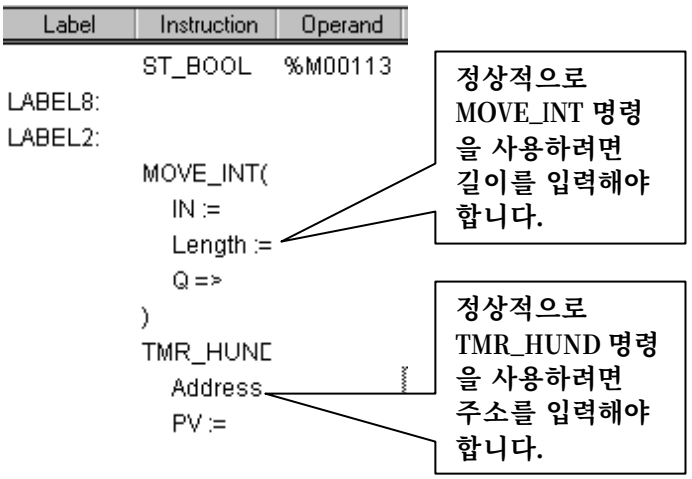
1. 새로운 명령어를 추가하거나 기존 명령어를 변경하고자 하는 위치를 선택하십시오.
2. 셀을 더블 클릭하십시오. 아래에서 보는 봐와 같이 Instruction Combo-Box가 나타날 것입니다.




3. 명령어를 선택하십시오. 펑션은 FUNCTION_NAME{으로 인식됩니다. 만약 펑션을 선택했다면 파라미터들은 펑션 블록을 프레임으로 하는 브라킷 사이에 나타날 것입니다.
4. 새로운 명령어는 IL 프로그램에 삽입됩니다.

## 어드레스와 길이 정보 완성하기

어떤 명령어는 운용하기 위해 명령어에 대한 길이(예, 이동 명령어)나 어드레스(예, 타이머 명령어)를 제공되기 원합니다. 이 정보를 요구하는 명령어는 아래 나타난 것처럼 명령어 컬럼에서 길이나 어드레스 필드를 보여줄 것입니다. 적절하게 길이나 어드레스를 입력하십시오.




## 명령어 파라미터 완성하기

명령어를 프로그램에 추가한 후 각각의 명령어 오퍼랜드에 변수나 상수를 할당할 필요가 있습니다. 이 섹션은 다음 방법 중 하나로 명령어 오퍼랜드를 완성하는 방법을 서술합니다:

- 오퍼랜드에 기존 변수 할당하기
- 오퍼랜드에 대한 새로운 변수 생성하기
- 오퍼랜드에 상수 값 할당하기

## 명령어 오퍼랜드에 변수 할당하기

변수는 명령어 오퍼랜드에 할당될 수 있습니다. 변수는 Local Scope 나 Global Scope 일 수 있습니다. 변수에 대한 기본 스코프는 Global Scope 입니다.

기존 변수를 할당하거나 명령어 오퍼랜드에 대한 새로운 변수를 생성할 수 있습니다. 이 섹션은 마우스와 키보드를 이용해 파라미터에 변수를 할당하는 방법을 서술합니다.

### 키보드를 이용해 기존 변수 할당하기:

1. 변수를 할당하기 원하는 파라미터에 커서를 위치하도록 방향키를 사용하십시오.

2. 다음과 같은 변수를 입력하거나 선택하십시오:

   - 정의된 변수 리스트에서 변수를 선택하고 Enter 키를 누르십시오. Variable Combo-Box 가 나타날 것입니다. 변수를 디스플레이하기 위해 상하 키를 사용하십시오. 사용하려고 하는 변수를 찾으면 받아 들이 위해 Enter 키를 누르십시오.
   - 이름으로 변수를 선택하려면 변수 이름을 입력하십시오. 받아 들이기 위해 Enter 키를 누르십시오.

## 마우스를 이용해 기존 변수를 할당하기:

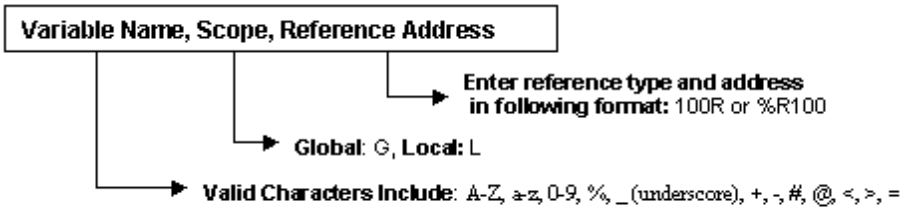
1. 변수를 할당하기 원하는 오퍼랜드를 더블 클릭하십시오. Variable Combo-Box 가 나타날 것입니다.
2. 변수 리스트를 확장하려면 Combo-Box 에서 화살표를 클릭하십시오. 리스트를 스크롤하기 위해 Slide Control 을 사용하십시오.
3. 리스트에서 변수를 선택하십시오. Enter 키를 누르거나 받아들이기 위해 오퍼랜드 영역 바깥을 클릭하십시오.

## 명령어 오퍼랜드에 대한 새로운 변수 생성하기

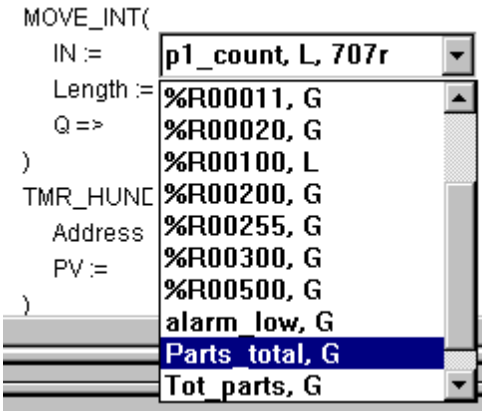
1. 키보드와 마우스를 이용하여 명령어 오퍼랜드를 선택하십시오.

   - **Keyboard Select:** 변수를 할당하고자 하는 파라미터에 커서를 갖다 놓기 위해 방향 키를 사용하십시오. Enter 를 누르십시오. Variable Combo-Box 가 나타날 것입니다.
   - **Mouse Select:** 파라미터 셀을 더블 클릭하십시오. Variable Combo-Box 가 나타날 것입니다.

2. 다음 가이드 라인을 이용해 편집 박스에 변수 이름, 레퍼런스 어드레스와 스코프(옵션)를 입력하십시오(필드는 “,”에 의해 분리해야 합니다.)




- 다음 예는 MOVE_INT Function 의 Parameter IN 에 대한 변수 “p1_count”의 생성을 보여주고 있습니다.




3. 변수를 받아들이기 위해 Enter key 를 누르십시오. – 또는 활동중인 셀을 클릭 오프하기 위해 마우스를 사용하십시오.

## 명령어 오퍼랜드에 상수 할당하기

상수는 명령어 오퍼랜드에 할당될 수 있습니다.

### 키보드 입력으로 상수 할당하기:

1. 상수를 할당하고자 하는 명령어 오퍼랜드에 위치하려면 방향키를 사용하십시오.
2. 상수값을 입력하고 Enter key 를 누르십시오.

### 상수에 대한 규칙

상수값은 오퍼랜드의 용량을 초과할 수 없으며, 명령어 타입을 지닌 일정한 포맷에 입력해야 합니다.

| Format | Prefix | Example |
|---|---|---|
| Decimal (default) | None | 43 |
| Real | None | 43.72 |
| Scientific | None | 6.93e4 |
| Binary | 2# | 2#11110000 |
| Octal | 8# | 8#730 |

## IL Editor에서 자르기, 복사, 붙여넣기 및 삭제하기

IL Editor 에서 Variable Declaration Table 에서 Rung 이나 셀을 포함하고 있는 변수를 제거하지 않고 명령어 리스트의 한 부분에서 다른 부분으로 그리고 한 블록에서 다른 블록으로 Rung 이나 셀을 자르고 복사할 수 있습니다.

또한 선택한 rungs 이나 셀들을 삭제할 수 있습니다.

이 섹션은 IL Editor window 에서 자르기, 복사, 붙여넣기 및 드래그&드롭을 수행할 수 있는 다양한 방법을 기술합니다.

## IL Editor 에서 자르기와 복사하기:

1. 편집하고자 하는 IL 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. 편집 메뉴나 오른쪽 마우스 메뉴에서 Cut 이나 Copy 를 선택하고 표준 툴바에서 Cut  이나 Copy  button 을 클릭하거나 Ctrl+C(Copy)나 Ctrl+X(Cut)를 누르십시오. 선택된 정보는 클립보드에 자르기나 복사됩니다.

**Note:** 텍스트로 Multi-row Instruction 의 한 부분을 복사할 수 있습니다. 다만 Multi-row Instruction 의 한 행을 복사한다면, VerasPro 로 붙여넣기를 하지 못할 것입니다.

## Drag and Drop 을 이용해 IL Editor 에서 자르기와 복사하기:

1. 편집하고자 하는 LD 블록을 여십시오. 복사나 자르기 하려는 Rung(s)이나 셀들을 선택하십시오.
2. **자르기 위해** (drag and drop 를 이용해), 마우스 화살표를 선택된 로직에 위치시켜놓고 Down 을 누르고 첫번째 마우스 버튼을 누르십시오. **복사하기 위해** (drag and drop 를 이용해) 마우스 화살표를 선택된 로직에 위치시켜놓고 CTRL key 를 계속해서 누른 후 첫번째 마우스 버튼을 누르십시오.

- 적절한 지역으로 로직을 이동시키십시오. 선택물을 드롭하기 위해 마우스 버튼이나 Ctrl key 를 놓으십시오.

### 다른 어플리케이션에 탭으로 구분된 문서로 복사하기:

1. 편집하고자 하는 IL 블록을 여십시오. 복사하고자 하는 행과 셀을 선택하십시오.
2. 편집 메뉴를 선택하여 Copy as Text 를 선택하십시오.
3. 선택된 정보는 탭으로 구분된 문서로 복사되고 , 탭으로 구분된 문서를 사용할 수 있는 모든 애플리케이션에서 붙여넣기 할 수 있습니다.

5

# 명령어 리스트 오퍼랜드

IL 프로그래밍 언어는 어큐뮬레이터를 기반으로 한 언어입니다. IL 프로그램에 의해 실행되는 명령어는 PLC 메모리에 위치하고 있는 어큐뮬레이터를 수정하거나 사용합니다. 어큐뮬레이터는 연산자나 Function call 에 의해 설정되어 있는 형식을 가지고 있습니다.

IL 프로그램에서 사용될 수 있는 연산자는 다음 표에 열거했습니다. IL 이 어큐뮬레이터를 기반으로 한 프로그래밍 언어이기 때문에 연산자는 어큐뮬레이터에서 직접 작업합니다. 비교 연산자는 동작 결과에 따라 Boolean (1 또는 0) 결과를 가지고 있습니다. IL 명령어에 대한 오퍼랜드는 변수, 레퍼런스 또는 상수의 형태로 될 수 있습니다. 연산자와 펑션은 IL 프로그램에서 충돌할 때 무조건 실행됩니다. 즉 명령어는 이전 명령어에 대한 결과에 관계없이 실행됩니다.

LD* (Load) 연산자는 오퍼랜드에 의해 명시된 변수와 상수값에 어큐뮬레이터의 컨텐츠를 설정하는데 사용됩니다. 또한 LD 연산자는 어큐뮬레이터의 타입을 설정합니다. ST* (Store) 연산자는 어큐뮬레이터의 컨텐츠에 오퍼랜드에 명시된 변수값을 설정하는데 사용됩니다. 연산자의 remainder 는 어큐뮬레이터의 컨텐츠를 조작하거나 어큐뮬레이터 값을 기반으로 한 동작을 수행합니다. 수식과 관계 연산에 있어서 어큐뮬레이터의 타입은 정수가 되어야 합니다

| Operator | Operand | Description / Affect upon Accumulator |
|---|---|---|
| LD_BOOL | Discrete Variable or Ref Address | Accumulator := Boolean Variable<br>어큐뮬레이터에서 Boolean 변수의 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Boolean 로 설정. |
| LDN_BOOL | Discrete Variable or Ref Address | Accumulator := NOT Variable<br>어큐뮬레이터에서 Boolean 변수의 네비게이션하는 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Boolean 로 설정. |
| LD_INT | Variable, Ref Address, or constant | Accumulator := Integer Variable<br>어큐뮬레이터에서 Integer 변수의 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Integer 로 설정. |
| LD_ENO | None | Accumulator := Boolean ENO<br>어큐뮬레이터에서 마지막 펑션이나 펑션 블록 Call 로부터 ENO(Enable Output)의 컨텐츠 로드하기.<br>어큐뮬레이터 타입은 Boolean 로 설정. |
| ST_BOOL | Discrete Variable or Ref Address | Variable := Boolean Accumulator<br>Boolean 변수에서 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| STN_BOOL | Discrete Variable or Ref Address | Variable := NOT Boolean Accumulator<br>Boolean variable 에 Boolean Accumulator 의 네비게이션하는 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| ST_INT | Variable or Ref Address | Variable := Integer Accumulator |

| | | |
|---|---|---|
| ST_DINT<br>ST_REAL<br>ST_WORD | | Variable := Double Integer Accumulator<br>Variable := Real Accumulator<br>Variable := Word Accumulator<br>변수에 Integer, Double Integer, Real, or Word Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨. |
| S | Discrete Variable or Ref Address | Set Coil Variable := Boolean Accumulator<br>Set Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| R | Discrete Variable or Ref Address | Reset Coil Variable := Boolean Accumulator<br>Reset Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| NT | Discrete Variable or Ref Address | Negative Transition Variable := Boolean Accumulator<br>Negative Transition Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기 .<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| PT | Discrete Variable or Ref Address | Positive Transition Variable := Boolean Accumulator<br>Positive Transition Coil Boolean 변수에 Boolean Accumulator 의 컨텐츠 저장하기.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |
| NOT | None | Accumulator := NOT Accumulator<br>Invert the Accumulator 의 Boolean 상태 반전하기.<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| AND<br>ANDN<br>AND( | Discrete Variable or Ref Address | Accumulator := Accumulator AND Boolean Variable<br>Accumulator := Accumulator AND NOT Boolean Variable<br>Accumulator := Accumulator AND Boolean expression<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| OR<br>ORN<br>OR( | Discrete Variable or Ref Address | Accumulator := Accumulator OR Boolean Variable<br>Accumulator := Accumulator OR NOT Boolean Variable<br>Accumulator := Accumulator OR Boolean expression<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| XOR<br>XORN | Discrete Variable or Ref Address | Accumulator := Accumulator XOR Boolean Variable<br>Accumulator := Accumulator XOR NOT Boolean Variable<br>어큐뮬레이터 타입은 변경 안됨(Boolean). |
| ADD | Variable, Ref Address, or Constant | Accumulator := Accumulator + operand<br>Accumulator Type unchanged (Integer). |
| SUB | Variable, Ref Address, or Constant | Accumulator := Accumulator – operand<br>Accumulator Type unchanged (Integer). |
| MUL | Variable, Ref Address, or Constant | Accumulator := Accumulator * operand<br>Accumulator Type unchanged (Integer). |
| DIV | Variable, Ref Address, or Constant | Accumulator := Accumulator / operand<br>Accumulator Type unchanged (Integer). |
| MOD | Variable, Ref Address, or Constant | Accumulator := Accumulator MOD operand<br>Accumulator Type unchanged (Integer). |
| GT | Variable, Ref Address, or Constant | Accumulator := (Accumulator > operand)<br>Accumulator Type set to Boolean. |
| GE | Variable, Ref Address, or | Accumulator := (Accumulator >= operand) |

| | | |
|---|---|---|
| | Constant | Accumulator Type set to Boolean. |
| EQ | Variable, Ref Address, or Constant | Accumulator := (Accumulator = operand)<br>Accumulator Type set to Boolean. |
| NE | Variable, Ref Address, or Constant | Accumulator := (Accumulator != operand)<br>Accumulator Type set to Boolean. |
| LE | Variable, Ref Address, or Constant | Accumulator := (Accumulator <= operand)<br>Accumulator Type set to Boolean. |
| LT | Variable, Ref Address, or Constant | Accumulator := (Accumulator < operand)<br>Accumulator Type set to Boolean. |
| RET<br>RETC<br>RETCN | None | 블록으로 무조건 돌아가기<br>Boolean Accumulator 이 1 의 값을 갖을 때 블록으로 돌아가기<br>Boolean Accumulator 이 0 의 값을 갖을 때 블록으로 돌아가기 |
| JMP<br>JMPC<br>JMPCN | Label | 라벨로 무조건 건너뛰기<br>Boolean Accumulator 이 1 의 값을 갖을 때 라벨로 점프하기<br>Boolean Accumulator 이 0 의 값을 갖을 때 라벨로 점프하기 |
| CAL<br>CALC<br>CALCN | Subroutine Name | 무조건 블록 부르기<br>Boolean Accumulator 이 1 의 값을 갖을 때 블록 부르기<br>Boolean Accumulator 이 0 의 값을 갖을 때 블록 부르기 |
| FUNC_NAME(<br>IN1:=<br>IN2:=<br>Q2=><br>) | Variable, Ref Address, or Constant, dependent upon type of the function parameters. | 내장된 PLC Function 부르기.<br>Accumulator 값과 타입은 출력 파라미터 Q1 의 값과 타입으로 설정됨. |
| ) | | End of Boolean nested operation.<br>어큐뮬레이터 값과 타입은 변경 안됨(Boolean). |

# 상수

IL Editor 는 다음과 같은 오퍼랜드로 사용할 수 있는 상수를 지원합니다: Integer, Real, Real with exponents, Base 16(16#xxxx), Base 8(8#xxx)와 Binary (2#xxxxxxxx). Boolean 연산자로 사용하는 오퍼랜드를 위해 1 이나 0 의 상수를 사용하려면, ALW_ON(1 또는 TRUE)과 ALW_OFF(0 또는 FALSE) 시스템 변수를 사용해야 합니다.

# 펑션

펑션은 형식적인 입력 리스트를 사용하고자 합니다. 또한 펑션은 이전의 명령어 결과를 무시하고 무조건 실행합니다. 펑션은 보통 결과를 Accumulator 에 배열하고 Accumulator 의 타입은 펑션에 의해 결정됩니다.

```
                              "AddResultGt100 := ((SQRT_INT(Input1) > 100)
SQRT_INT(
   IN:=     Input1       "Input to Function call
)
GT          100
ST_BOOL     AddResultGt100 "Set to TRUE if result greater than 100
```

# 펑션 블록

펑션 블록 또한 형식적인 입력 리스트를 사용합니다. 또한 평션 블록은 이전의 명령어 결과를 무시하고 무조건 실행합니다. 펑션 블록은 보통 결과를 Accumulator 에 배열하고 Accumulator 의 타입은 펑션 블록에 의해 결정됩니다. 펑션 블록의 어드레스는 입력 파라미터입니다.

```
UPCTR(
     Address :=    %R00022
     R :=          %M00012
     PV :=         1
)
```

# PLC 내장형 펑션과 펑션 블록

다음 PLC 내장형 펑션과 펑션 블록은 IL 프로그램에 의해 부를 수 있습니다. Resulting Accumulator 의 값과 형태는 블록과 펑션 블록마다 명시됩니다. ENO 컬럼에서 YES 는 펑션이나 펑션 ENO(Enable Output) 변수를 설정한다는 것을 가리킵니다. 테이블 또한 펑션이나 펑션 블록이 어큐뮬레이터에게 펑션 블록 앞에 LD Bool 명령어를 삽입함으로써 초기화되기를 요구합니다.

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| 타이머와 카운터(펑션 블록) | | | |
| ONDTR_TENTHS | Yes | Yes | N/A |
| ONDTR_HUNDS | Yes | Yes | N/A |
| ONDTR_THOUS | Yes | Yes | N/A |
| TMR_TENTHS | Yes | Yes | N/A |
| TMR_HUNDS | Yes | Yes | N/A |
| TMR_THOUS | Yes | Yes | N/A |
| OFDT_TENTHS | Yes | Yes | N/A |
| OFDT_HUNDS | Yes | Yes | N/A |
| OFDT_THOUS | Yes | Yes | N/A |
| UPCTR | Yes | | N/A |
| DNCTR | Yes | | N/A |
| Standard Math (펑션) | | | |
| ADD_INT | Yes | | Output Parameter Q, Integer |
| ADD_DINT | Yes | | Output Parameter Q, Double Integer |
| ADD_REAL | Yes | | Output Parameter Q, Real |
| SUB_INT | Yes | | Output Parameter Q, Integer |
| SUB_DINT | Yes | | Output Parameter Q, Double Integer |
| SUB_REAL | Yes | | Output Parameter Q, Real |
| MUL_INT | Yes | | Output Parameter Q, Integer |
| MUL_DINT | Yes | | Output Parameter Q, Double Integer |
| MUL_REAL | Yes | | Output Parameter Q, Real |
| DIV_INT | Yes | | Output Parameter Q, Integer |
| DIV_DINT | Yes | | Output Parameter Q, Double Integer |
| DIV_REAL | Yes | | Output Parameter Q, Real |
| ADD_INT | Yes | | Output Parameter Q, Integer |
| ADD_DINT | Yes | | Output Parameter Q, Double Integer |
| ADD_REAL | Yes | | Output Parameter Q, Real |
| MOD_INT | Yes | | Output Parameter Q, Integer |
| MOD_DINT | Yes | | Output Parameter Q, Double Integer |
| SQRT_INT | Yes | | Output Parameter Q, Integer |
| SQRT_DINT | Yes | | Output Parameter Q, Double Integer |
| SQRT_REAL | Yes | | Output Parameter Q, Real |

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| Trig Functions (Functions) | | | |
| SIN | Yes | | Output Parameter Q, Real |
| COS | Yes | | Output Parameter Q, Real |
| TAN | Yes | | Output Parameter Q, Real |
| ASIN | Yes | | Output Parameter Q, Real |
| ACOS | Yes | | Output Parameter Q, Real |
| ATAN | Yes | | Output Parameter Q, Real |
| Logarithmic/Exponential Functions (Functions) | Yes | | Output Parameter Q, Real |
| LOG | Yes | | Output Parameter Q, Real |
| LN | Yes | | Output Parameter Q, Real |
| EXP | Yes | | Output Parameter Q, Real |
| EXPT | | | |
| Radian Conversion Functions (Functions) | | | |
| RAD_2_DEG | Yes | | Output Parameter Q, Real |
| DEG_2_RAD | Yes | | Output Parameter Q, Real |
| Relational Functions (Functions) | | | |
| EQ_INT | No | | Output Parameter Q, Boolean |
| EQ_DINT | No | | Output Parameter Q, Boolean |
| EQ_REAL | No | | Output Parameter Q, Boolean |
| NE_INT | No | | Output Parameter Q, Boolean |
| NE_DINT | No | | Output Parameter Q, Boolean |
| NE_REAL | No | | Output Parameter Q, Boolean |
| GT_INT | No | | Output Parameter Q, Boolean |
| GT_DINT | No | | Output Parameter Q, Boolean |
| GT_REAL | No | | Output Parameter Q, Boolean |
| GE_INT | No | | Output Parameter Q, Boolean |
| GE_DINT | No | | Output Parameter Q, Boolean |
| GE_REAL | No | | Output Parameter Q, Boolean |
| LT_INT | No | | Output Parameter Q, Boolean |
| LT_DINT | No | | Output Parameter Q, Boolean |
| LT_REAL | No | | Output Parameter Q, Boolean |
| LE_INT | No | | Output Parameter Q, Boolean |
| LE_DINT | No | | Output Parameter Q, Boolean |
| LE_REAL | No | | Output Parameter Q, Boolean |
| | No | | Output Parameter Q, Boolean |
| | No | | Output Parameter Q, Boolean |
| Relational Functions (Functions) | | | |
| RANGE_INT | No | | Output Parameter Q, Boolean |
| RANGE_DINT | No | | Output Parameter Q, Boolean |
| RANGE_WORD | No | | Output Parameter Q, Boolean |
| Bit Functions (Functions) | | | |
| AND_WORD | Yes | | Output Parameter Q, Word |
| OR_WORD | Yes | | Output Parameter Q, Word |
| XOR_WORD | Yes | | Output Parameter Q, Word |

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| NOT_WORD | Yes | | Output Parameter Q, Word |
| Bit Functions (Functions) | | | |
| SHL_WORD | No | | Output Parameter B2, Boolean |
| SHR_WORD | No | | Output Parameter B2, Boolean |
| ROL_WORD | Yes | | N/A |
| ROR_WORD | Yes | | N/A |
| BTST_WORD | No | | Output Parameter Q, Boolean |
| BSET_WORD | Yes | | N/A |
| BCLR_WORD | Yes | | N/A |
| BPOS_WORD | Yes | | Output Parameter POS, Integer |
| MSKCMP_WORD | No | | Output Parameter MC, Boolean |
| MSKCMP_DWORD | No | | Output Parameter MC, Boolean |
| Data Move Functions (Functions) | | | |
| MOVE_BOOL | Yes | | N/A |
| MOVE_INT | Yes | | N/A |
| MOVE_WORD | Yes | | N/A |
| MOVE_REAL | Yes | | N/A |
| Data Move Functions (Functions) | | | |
| BLKMOV_INT | Yes | | N/A |
| BLKMOV_WORD | Yes | | N/A |
| BLKMOV_REAL | Yes | | N/A |
| BLKCLR_WORD | Yes | | N/A |
| SHFR_BIT | Yes | Yes | N/A |
| SHFR_WORD | Yes | Yes | N/A |
| COMM_REQ | No | | Output Parameter FT, Boolean |
| Data Move Function Blocks (Functions) | | | |
| BIT_SEQ | Yes | Yes | N/A |
| Table Functions (Functions) | | | |
| ARRAY_MOVE_INT | Yes | | N/A |
| ARRAY_MOVE_DINT | Yes | | N/A |
| ARRAY_MOVE_BIT | Yes | | N/A |
| ARRAY_MOVE_BYTE | Yes | | N/A |
| ARRAY_MOVE_WORD | Yes | | N/A |
| SRCH_EQ_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_EQ_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_EQ_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_EQ_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_NE_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_GT_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_GE_INT | Yes | | Output Parameter FD, Boolean |

| Function or Function Block | ENO | Requires LD Boolean Instruction | Resulting Accumulator Value and Type |
|---|---|---|---|
| SRCH_GE_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_GE_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_GE_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_LT_WORD | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_INT | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_DINT | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_BYTE | Yes | | Output Parameter FD, Boolean |
| SRCH_LE_WORD | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| | Yes | | Output Parameter FD, Boolean |
| Conversion Functions (Functions) | | | |
| INT_TO_BCD4 | Yes | | Output Parameter Q, Word |
| BCD4_TO_INT | Yes | | Output Parameter Q, Integer |
| INT_TO_REAL | Yes | | Output Parameter Q, Integer |
| DINT_TO_REAL | Yes | | Output Parameter Q, Integer |
| BCD4_TO_REAL | Yes | | Output Parameter Q, Integer |
| WORD_TO_REAL | Yes | | Output Parameter Q, Integer |
| REAL_TO_WORD | Yes | | Output Parameter Q, Word |
| REAL_TO_INT | Yes | | Output Parameter Q, Integer |
| REAL_TO_DINT | Yes | | Output Parameter Q, Double Integer |
| Control Functions (Functions) | | | |
| DOIO | Yes | | N/A |
| SVC_REQ | Yes | | N/A |
| Control Function Blocks (Function Blocks) | | | |
| PID_IND | Yes | Yes | Output Parameter CV, Integer |
| PID_ISA | Yes | Yes | Output Parameter CV, Integer |
| New Series 90-30 (36x) Function Blocks (Function Blocks) | | | |
| SER | Yes | Yes | N/A |
| VersaMax Release 1.1 and later Function Blocks | | | |
| DRUM | Yes | Yes | Output Parameter Q, Word |
| SCALE_INT | Yes | | Output Parameter OUT, Integer |
| SCALE_WORD | Yes | | Output Parameter OUT, Word |

5

# 논리식 네스트 구성하기

사용자는 8 단위까지 **AND(** 와 **OR(** 연산자를 이용해 논리식을 네스트(내포)할 수 있습니다. 네스트된 논리식은 AND( 이나 OR ( 연산자로 시작되고 다음 명령어는 LD_BOOL 또는 LDN_BOOL 연산자가 되어야 합니다. 네스트된 논리식에서 사용할 수 있는 연산자는 다음과 같습니다. : S, R, ST_BOOL, STN_BOOL, NT, PT, NOT, AND*, OR*과 XOR*. 네스트된 논리식은 복잡한 Boolean 방정식의 값을 구하는데 사용됩니다. 다음은 네스트된 논리식의 예입니다:

```
LD_BOOL     %I00001     %I00001
AND(
LD_BOOL     %I00002     %I00002
OR          %T00010     %T00010
AND         %T00011     %T00011
OR(
LD_BOOL     %T00032     %T00032
OR          %T00033     %T00033
ST_BOOL     %M00077     %M00077
ANDN        %T00088     %T00088

)
AND         %T00088     %T00088
OR          %Q00067     %Q00067

)
ST_BOOL     %T00099     %T00099
```

컨트롤 명령어나 펑션 부르기는 네스트된 논리식에 넣을 수 없습니다.

# ENO 사용하기

펑션이나 펑션 블록이 실행될 때, ENO 출력은 펑션 블록의 성공적인 실행을 나타내는데 사용됩니다. 펑션을 실행할 때, 에러가 발생하면 ENO 출력을 설정하거나 재설정합니다. 사용자는 어큐뮬레이터에서 ENO 출력을 로드하기 위해 LD_ENO 연산자를 이용하여 ENO 의 상태를 점검한 다음 발생된 에러 값을 검사할 수 있습니다.

```
"Calculate the following expression
        %R00005 := (%R00001 + 100) * %R00077
        Set calcOK variable to 1 if calculation successful

        ADD_DINT(
          IN1 :=    %R00001     %R00001
          IN2 :=    100
        )
        ST_DINT     %R00003     %R00003
        LD_ENO                          "LD_ENO used to get ENO output of ADD_DINT
        JMPCN       error:
        MUL_DINT(
          IN1 :=    %R00003     %R00003
          IN2 :=    %R00077     %R00077
        )
        ST_DINT     %R00005     %R00005
        LD_ENO                          "LD_ENO used to get ENO output of MUL_DINT
        JMPCN       error:
        LD_BOOL     ALW_ON      %S00007
        ST_BOOL     calcOK      %Q00077 "calcOK := 1
        RET
error:
        LD_BOOL     ALW_OFF     %S00008
        ST_BOOL     calcOK      %Q00077 "calcOK := 0
        RET
```

5

# 임시 변수 정의하기

명령어 리스트를 프로그래밍할 때, 사용자는 IL Accumulator에 의해 사용될 수 있도록 메모리 위치를 정의해야 합니다. 사용자는 Boolean Accumulator로 사용할 수 있도록 %T, %M 또는 %Q 메모리에 8 개의 연속된 디지털 포인트를 정의해야합니다. 또한 the Word, Integer, Double Integer와 Real Accumulator로 사용할 수 있도록 %AI, %AQ 또는 %R 메모리에 2 워드의 레퍼런스 메모리를 정의해야 합니다. 사용자는 Folder Properties dialog 를 이용해 IL 임시 변수의 정의를 내립니다. 일단 임시 변수가 정의를 내리기 되면 이들 레퍼런스는 프로그램에서 직접 레퍼런스될 수 없습니다.

# 에러 점검하기

컴파일될 때, 다음과 같은 에러 점검은 IL 프로그램에서 수행됩니다.

1. Accumulator의 형태는 연산자 형태와 호환되어야합니다.
2. LD 명령어나 펑션은 부르기는 ST 명령어 전에 발생해야 합니다 (Accumulator는 값과 형태를 가지고 있습니다.)
3. 네스트된 식의 ()는 매치되어야 합니다.
4. 레퍼런스 형태는 연산자로 고쳐야 합니다. 예를 들어 데이터 타입은 bit 가 되어야 하기 때문에 “S %R00001”이 될 수 없습니다.
5. JMPs 와 MCRs 은 연결 라벨을 가지고 있어야 합니다.
6. 모든 변수는 정의되어야 합니다.
7. CALs 은 IL/LD Blocks 과 매치되어야합니다.
8. 레퍼런스 어드레스는 설치된 CPU 의 메모리 범위 내에 있어야합니다.
9. 펑션과 펑션 블록 파라미터는 정확한 메모리 형태와 값이 되어야 합니다.

Chapter 6

# Variable Declaration Table로 작업하기

Variable Declaration Table(VDT)는 온라인으로 변수를 생성, 편집 및 보는데 사용하는 창입니다. VDT 는 보통 Folder Browser 가 열릴 때 자동으로 저장합니다. 만약 VDT 가 Folder Browser 에서 디스플레이가 되지 않으면 보기 메뉴를 선택하여 Variable Declaration Table 을 선택하거나 Variable Declaration Table toolbar button  을 클릭하십시오.

이 장은 변수를 소개하고 LD 와 IL 프로그램에서 사용할 수 있는 변수를 생성하는 방법을 기술합니다.

## 목차

이 장은 다음과 같은 정보를 제공합니다:

- Variable Declaration Table(VDT) 개요
- 변수 소개
- 변수를 생성하고 편집하는 방법
- 자르기, 복사와 붙여넣기 등 표준 명령어를 이용한 변수로 작업하는 방법
- 변수를 삭제하고 VDT 를 압축하여 저장하는 방법
- SNF 파일로 변수 보내기와 불러오는 방법

6

# *Variable Declaration Table 개요*

Variable Declaration Table 은 탭으로 이루어졌습니다. 각각의 탭은 아래 기술한 변수들을 디스플레이합니다:

**Global Tab:** 모든 Global Variables 디스플레이하기.

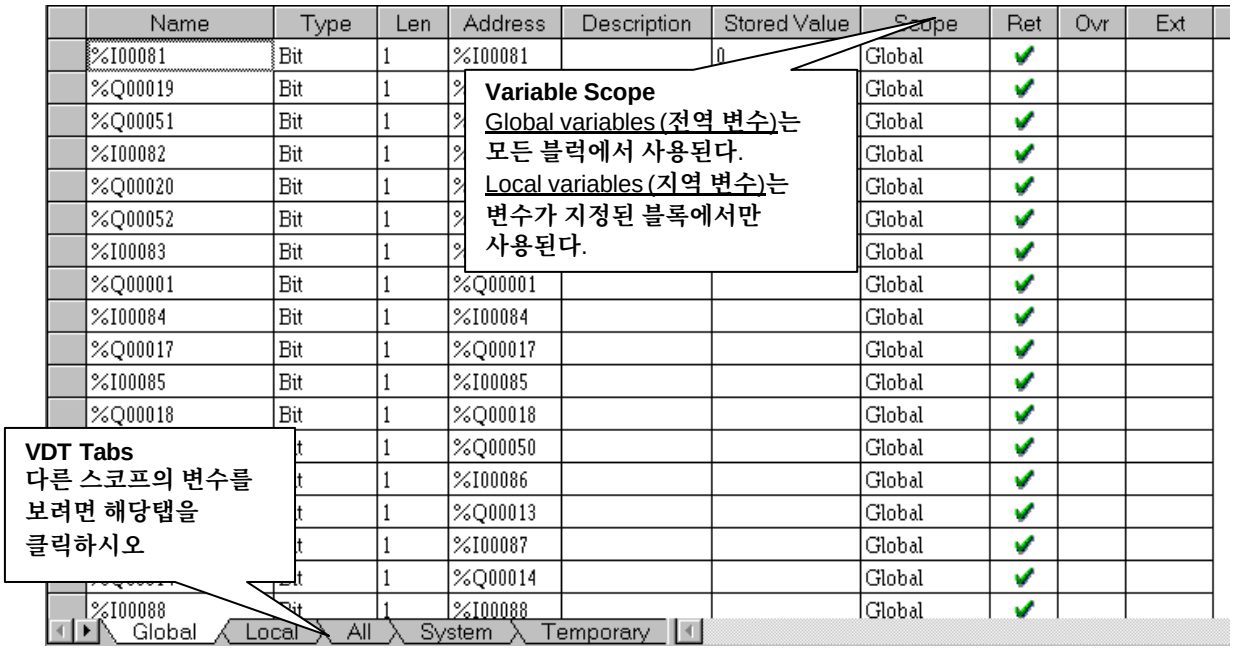
**Local Tab:** 활동중인 블록에서 Local Variables 디스플레이하기. 만약 로컬 탭이 열려있는 블록이 없다면 공백상태가 됩니다.

**All Tab:** 모든 변수 디스플레이하기.

**System Tab:** 시스템 변수 디스플레이하기. 이들 변수는 편집할 수 없지만 로직으로 사용할 수 있습니다.

**Temporary Tab:** IL Editor 에서 사용하는 임시 변수에 대한 Folder Properties dialog box 에서 생성되는 변수 디스플레이하기. 임시 변수는 편집할 수 없습니다.

아래 그림은 Variable Declaration Table 의 예를 보여주고 있습니다.



# Variable Table Field 설명

이 장은 변수를 정의할 때 포함되는 필드를 기술합니다.

## 이름

변수 프로그래밍에서 이름은 메모리 어드레스에 할당됩니다. 로컬 이름으로 할당하거나 레퍼런스 어드레스와 변수 이름과 같게 연결시킬 수 있습니다. 만약 변수에 대한 이름이 할당되지 않는다면 레퍼런스 어드레스는 이름처럼 자동으로 사용됩니다. 변수 이름은 31 문자를 넘을 수 없으며 오직 A - Z, a - z, 0 – 9, _, +, -, #, @, <, >, = 등의 문자로만 구성될 수 있습니다. 변수 이름은 글자나 _로 시작해야 합니다.

## 형태

각각의 변수는 변수가 프로그램에서 사용되는 방법을 묘사한 데이터 형태로 할당되어야 합니다. VersaPro 는 Bit, Byte 또는 Word 데이터 형태로 변수를 할당할 수 있도록 합니다.

## 길이

길이는 사용하고 있는 변수의 총 데이터를 열거한 것입니다. 워드 데이터 형태에서 길이는 워드 수를 열거합니다. Byte 데이터 형태에서 길이는 Byte 수를 열거합니다. Bit 데이터 형태에서 길이는 1 이 되어야 합니다. 데이터 길이는 변수에 대해 열거된 총 저장 값 데이터를 정의합니다.

## 어드레스

레퍼런스 어드레스는 PLC 의 물리적인 위치한 변수에 연결합니다.

## 설명

설명 필드는 변수에 대한 정보를 입력하는 곳입니다. 이 필드는 옵션이며 공백으로 남아 있을 수 있습니다. 설명은 64 문자를 넘어서는 안됩니다.

## 저장값

Variable Declaration Table 에서 변수에 대해 열거된 초기값은 변수가 프로그램에서 사용하지 않는다 하더라도 프로그램 전송에 따라 PLC 로 전송됩니다. 이러한 값들은 변수의 속성을 고려한 것입니다. Reference Table 을 보유하는 만큼 변수마다 많은 정보를 열거할 수 있습니다.

## Scope

Scope 는 폴더에서 사용될 수 있는 변수 영역을 정의합니다. Scope 에 대한 가능 변수는 “Global”와 (폴더에서 각 블록에 대한) “Local: <block name>” 입니다. 시스템 변수는 PLC CPU 에 의해 사용을 미리 정의합니다.

### 유지 (Retentive)

Retentive 필드는 정전이 된 후 선택된 변수의 값을 유지할 것인지 말 것인지를 명시합니다. VDT 에 나타난 Retentive 상태는 저장할 때, 변수를 초기화합니다. 로직이 운용될 때는 현재 Retentive 상태는 나타나지 않습니다. VersaPro 는 Retentive 상태가 %M, %Q 와 %T 을 제외한 모든 메모리 영역을 점검할 것을 요구합니다. Row validation 는 이들 영역을 제외한 잘못된 설정을 자동으로 고쳐줍니다.

### Overrides

Override 는 프로그램을 점검하고 메인터넌스하는 툴입니다. 입력을 시뮬레이트하기 위해 Overrides 를 이용함으로써 I/O 하드웨어와 연결되지 않은 PLC 에서 프로그램을 테스트할 수 있습니다. 또한 I/O 하드웨어가 코일 동작을 막기 위해 오버라이드를 이용해 연결할 때, 프로그램을 점검할 수 있습니다. 오버라이드는 디바이스가 수리될 때까지 기계 작동은 계속해야 하기 때문에 잘못된 입력 디바이스를 일시적으로 통과시킬 수 있습니다. 하지만 이러한 방법으로 오버라이드를 사용하면 정상적인 디바이스의 오버라이드될 수 있음을 명심하십시오. 저장값이나 오버라이드가 저장될 때 VDT 에서 보여지는 오버라이드 상태는 변수를 초기화합니다. 프로그램이 운용될 때는 현재의 오버라이드는 나타나지 않습니다. 오버라이드는 %G, %I, %M 이나 %Q 으로 할당된 Non-system bit 변수를 점검할 수 있습니다

### EXT

EXT 필드가 선택되면 변수는 다른 애플리케이션으로 분할 될 수 있습니다.

# 변수 생성 및 수정하기

VersaPro 는 프로그램 로직에서 사용하기 위해 변수를 생성하고 수정할 수 있도록 해줍니다. 변수는 다양한 데이터 형태, 스코프, 길이 및 어드레스를 가질 수 있습니다. 여러 변수는 동일한 어드레스를 레퍼런스할 수 있고 사용자는 설명과 변수를 지닌 초기 저장 값과 결합시킬 수 있습니다. 여러분은 VersaPro 로 프로그램 구조에 적합한 변수를 만들거나 수정할 수 있습니다.

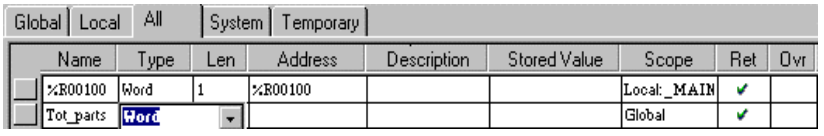
시스템 변수(%S, %SA, %SB, %SC)는 편집될 수 없습니다. 이들은 폴더가 생성될 때마다 변수 리스트에 나타냅니다.

Variable Declaration Table 에서 변수로 만드는 수정은 프로그램 로직을 포함하고 있는 모든 레퍼런스 포인트에 영향을 미칩니다.

**Note:** 오버랩된 변수의 Declaration(예를 들면 길이 3 을 지닌 %R8 에 있는 변수와 길이 2 의 %R10 에 있는 변수는 %R10 에서 오버랩을 가지고 있습니다.)은 PLC 로 저장값을 저장하려고 할 때 저장값에서 충돌하게 될 것입니다. 이러한 충돌을 해결하기 위해 변수를 재위치 시키기 위해 VDT 를 사용해야 합니다.

## VDT 에서 새로운 변수를 생성하거나 기본 변수 수정하기:

1. Folder Browser 에서 Variable Declarations icon 을 더블클릭 하십시오. 커서는 Variable Declaration Table 에서 활동하게 될 것입니다.
2. 생성이나 편집하기 위해 다음 단계를 수행하십시오:
   - 새로운 변수를 생성하기 위해 오른쪽 마우스 버튼을 클릭하여 Insert Row 를 선택하십시오. 새로운 행이 VDT 에 나타날 것입니다.
   - 기존 변수를 편집하기 위해 변수를 위치하고 편집하고자 하는 필드를 더블클릭 하십시오.




3. 새로운 변수에 대해 이름 형태 및 어드레스를 완성하고 스코프와 길이를 확인하십시오. 또한 설명과 저장값을 입력할 수 있습니다. 기존 변수를 수정하기 위해 애플리케이션에서 요구하는 만큼 파라미터를 변경하십시오:
   - VersaPro 는 변수 정의가 불완전하다면 메시지 박스를 나타내거나 수정된 변수를 인정하지 않을 것입니다.

6

# *VDT 에서 네비게이션하기*

VersaPro 는 마우스나 키보드를 이용하여 네비게이션할 수 있도록 해줍니다. 또한 Go to Variable dialog box 나 Edit Variable 메뉴항목을 이용하여 네비게이션할 수 있습니다.

## 마우스와 키보드를 이용해 네비게이션하기

### 마우스를 이용해 네비게이션하기:

편집하고자 하는 셀로 마우스를 이동한 다음 첫 번째 마우스 버튼을 클릭합니다. 키보드와 함께 네비게이션할 때, 셀을 편집한다면 선택물 이동은 이동하고자 하는 곳에서 셀 안으로 데이터를 받아들인다는 것을 의미합니다.

### 키보드를 이용해 네비게이션하기:

키보드를 이용해 VDT 를 통해 네비게이션하려면 다음 표를 참조하십시오.

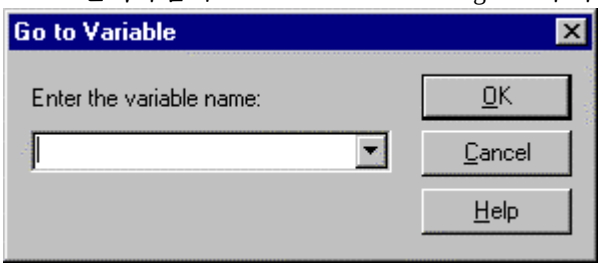
| Key | Action |
|---|---|
| Arrow Keys | 만약 현재 셀이 편집하기 위해 선택되지 않았다면(텍스트가 아니라 셀이 밝아졌다면), 방향키는 선택한 한 셀을 방향키 방향으로 이동시킵니다.<br>만약 현재 셀이 편집하기 위해 선택되었다면, 오른쪽과 왼쪽 방향키는 텍스트의 처음이나 끝에 도달할 때까지 셀에서 오른쪽이나 왼쪽으로 커서를 이동시킵니다.<br>방향키를 이용해 한 행에서 다른 행으로의 이동은 행 확인을 병행합니다. |
| Enter | 만약 현재 셀이 편집하기 위해 선택되지 않았다면(텍스트가 아니라 셀이 밝아졌다면), Enter key 는 편집하기 위한 셀을 선택합니다.<br>만약 현재 셀이 편집하기 위해 선택되었다면 Enter key 는 셀의 컨텐츠를 확인하고 현재 행 아래에 있는 셀을 선택합니다. |
| F2 | 편집하기 위해 셀을 선택여부를 결정합니다. |
| Shift + Arrow Keys | 만약 현재 셀이 편집하기 위해 선택되지 않았다면(텍스트가 아니라 셀이 밝아졌다면), Shift + 방향 키는 현재 셀을 선택하고 선택한 한 셀을 방향키 방향으로 이동시킵니다.<br>만약 현재 셀이 편집하기 위해 선택되었다면, 키는 현재 캐릭터를 선택하고 선택한 한 캐릭터를 방향키 방향으로 이동시킵니다. |
| Shift + Space Bar | 현재 행을 선택합니다. |
| Home | 만약 현재 셀이 편집하기 위해 선택되지 않았다면(텍스트가 아니라 셀이 밝아졌다면), Home 키는 현재 행에 있는 첫번째 셀을 선택합니다.<br>만약 현재 셀이 편집하기 위해 선택되었다면, Home 키는 셀에 있는 첫 번째 캐릭터의 왼쪽으로 커서를 이동시킵니다. |

| | |
|---|---|
| End | 만약 현재 셀이 편집하기 위해 선택되지 않았다면(텍스트가 아니라 셀이 밝아졌다면), End 키는 현재 행에 있는 마지막 셀을 선택합니다.<br>만약 현재 셀이 편집하기 위해 선택되었다면, End 키는 셀에 있는 마지막 캐릭터의 오른쪽으로 커서를 이동시킵니다. |
| Ctrl + Home | 첫 번째 행에서 첫 번째 셀 을 선택합니다. |
| Ctrl + End | 마지막 행에서 첫 번째 셀을 선택합니다. |
| Tab | 만약 현재 셀이 편집하기 위해 선택되지 않았다면(텍스트가 아니라 셀이 밝아졌다면), Tab 키는 오른쪽에서 다음 셀을 선택합니다.<br>만약 현재 셀이 편집하기 위해 선택되었다면, Tab 키는 셀의 컨텐츠를 확인하고 오른쪽으로 다음 셀을 선택합니다. |
| Shift + Tab | 만약 현 셀이 편집하기 위해 선택되지 않았다면(텍스트가 아니라 셀이 밝아졌다면), Shift + Tab 키는 왼쪽에서 다음 셀을 선택합니다.<br>만약 현 셀이 편집하기 위해 선택되었다면, Shift + Tab 키는 셀의 컨텐츠를 확인하고 왼쪽으로 다음 셀을 선택합니다.<br>Note: 가장 왼쪽에 있는 셀에서 오른 쪽으로 탭하거나 가장 오른쪽에서 오른쪽으로 탭하면서 한 행이 다른 행으로 이동하는 것은 항상 행 인증을 시킵니다. |
| Ctrl + Tab | 오른쪽으로 다음 VDT tab(Local, Global, All, System)을 디스플레이합니다. |
| Page Down/Up | 페이지를 수직으로 불 수 있도록 Page down/up 시키고 이동하기 전에 동일한 x, y 위치를 선택합니다. |

## Go To Variable Dialog Box 를 이용해 변수 위치 정하기

Go To Variable dialog box 를 이용해 변수의 위치를 빠르게 정할 수 있습니다.이름으로 변수의 위치를 정하려면 다음 단계를 수행하십시오

1. Variable Declaration Table 에서 오른쪽 마우스 버튼을 클릭하여 Go to Variable 선택하십시오. Go to Variable dialog box 가 나타날 것입니다.




2. 가고자 하는 변수 이름을 입력하거나 drop-down list 에서 변수 이름을 선택하십시오.
3. OK 를 클릭하십시오. 입력된 변수는 Variable Declaration Table 에서 선택됩니다.

6

# VDT 에서 자르기, 복사, 붙여넣기 및 Drag and Drop 사용하기

VersaPro 는 래더 로직에서 래더 요소를 제거하지 않거나 명령어 리스트 로직에서 명령어를 제거하지않고 자르기나 복사한 후에 Variable Declaration Table 에 있는 단일 또는 여러 변수에 변수나 셀을 붙여넣기 할 수 있습니다. 또한 Variable Declaration Table 에 있는 VersaPro 가 아니라 애플리케이션에 의해 클립보드에 있는 항목을 붙여넣기 할 수 있고 탭으로 분리된 문서를 받아들인 다른 애플리케이션으로 붙여넣기 하기 위해 클립보드로 항목을 자르거나 복사할 수 있습니다. 붙여넣기는 셀이나 변수에서 정보를 겹쳐쓰는 것입니다.

VersaPro 는 VDT 으로 또는 VDT 에서 편집하기 위해 drag and drop -cut 이나 –copy 할 수 있게 합니다. VDT 으로 드롭하는 것은 셀이나 변수에서 정보를 겹쳐쓰는 것입니다.

## Variable Declaration Table 에서 자르기, 복사 및 붙여넣기 :

1. Variable Declaration Table 를 열고 단일 혹은 다중 변수 중에서 변수나 셀을 선택하십시오.
2. 편집 메뉴에서 Cut 이나 Copy 를 선택하여 툴바에서 Cut button  이나 Copy button  을 클릭하거나 Ctrl + C(Copy)나 Ctrl + X(Cut)를 누르십시오. 변수를 선택하기 위해 변수에 있는 모든 필드는 선택되어야 합니다. 선택된 데이터는 클립보드로 자르거나 복사됩니다. 이제 탭으로 분리된 문서를 읽을 수 있는 다른 애플리케이션으로 붙여넣기 할 수 있습니다.
3. 자르거나 복사한 정보를 붙여넣기 하고자 하는 곳에 변수나 셀을 선택하십시오. 또다른 애플리케이션에서 시작된 클립보드에 있는 정보는 VersaPro 로 붙여넣기할 수 있습니다.
4. 편집 메뉴에서 Paste 를 선택하여 Paste button  을 클릭하거나 Ctrl + V 를 누르십시오. 행 인증은 VDT 로 붙여넣기 하기 전에 글로벌 스코프가 기존 변수와 충돌하는지를 점검하기 위해 각각의 변수를 수행하게 합니다. 만약 변수가 행 인증에 실패한다면 에러 메시지가 나타날 것이고 그 다음에 에러를 나타낸 대화창이 나타날 것입니다. 만약 문제를 해결하면 행 인증은 다른 변수에 대해 검증을 계속할 것입니다. 만약 클립보드에 있는 정보가 VersaPro 에 붙여넣기를 할 수 있고 변수나 셀의 수가 밝아진 영역과 일치한다면 정보는 붙여넣기를 할 수 있습니다.

## 자르기와 복사하기에 대한 세부사항

- 변수에 있는 모든 셀이 선택되고 자르기를 한다면 변수들은 폴더에서 삭제됩니다.
- 변수간의 수직 차는 변수를 클립보드로 복사될 때, 삭제됩니다.
- 클립보드에 있는 행은 CR/LF 에 분리되고 컬럼은 탭에 의해 분리됩니다. 이러한 분리는 외부 애플리케이션으로 붙여넣기할 때 명백해집니다.
- 자르기나 복사된 길이는 1 이상이면 저장된 값은 외부 애플리케이션으로 붙여넣기할 때, 콤마에 의해 분리됩니다(Microsoft Excel®과 같이).
- 클립보드로 자르기나 복사되었을 때, Retentive 값과 Override 값은 True=1 과 False=0 으로 저장됩니다.
- 키보드로는 연속되지 않은 행을 한꺼번에 선택할 수 없습니다.
- 다음 필드는 변수에서 자르기를 할 수 없습니다: Type, Length, Scope, Retentive, Override.

## 붙여넣기에 대한 세부사항

- 붙여넣기 위해 선택된 변수가 클립보드에 있는 변수보다 많으면 선택된 변수는 중복되고 남아있는 클립보드 변수는 테이블에 삽입됩니다.
- 붙여넣기 위해 선택된 변수가 클립보드에 있는 변수보다 적으면 가능한 한 선택된 변수 만큼 중복되고 나머지는 삭제됩니다.
- 아무 것이나 System tab 으로 붙여넣기 할 수 없습니다.
- 아무 것이나 Temporary tab 으로 붙여넣기 할 수 없습니다.
- 변수가 Global table 로 붙여넣기가 된다면, 원래의 스코프와 관계없이 Global Variable 이 됩니다.
- 변수가 Local table 로 붙여넣기가 된다면, 원래의 스코프와 관계없이 블록에서 Local 이 됩니다.
- Global variable 이 모든 테이블에 붙여넣기가 된다면, 이 변수는 Global 로 남아 있습니다.
- Local variable 이 모든 테이블에 붙여넣기가 된다면, 프로그램은 정의된 블록으로 할당하려 할 것입니다. 이 블록이 폴더에 존재하지 않는다면 Global variable 을 만들 것입니다.
- 보이는 테이블에서 붙여넣기 동작에 의해 변경된 어떤 변수 또한 변수가 나타나는 다른 테이블에서도 변경될 것입니다.
- VDT 에서 변경된 변수 또한 IL 과 LD program logic displays 에서 변경될 것입니다.
- 셀에 있어서 선택된 영역이 클립보드에 있는 영역보다 작을 경우 에러 메시지가 나타나고 붙여넣기 기능은 상실될 것입니다. 만약 선택된

영역이 클립보드에 있는 영역보다 클 경우 자르기나 복사된 정보는 선택된 영역의 왼쪽 코너 위에 붙여질 수 있습니다.

- 어떤 텍스트는 free floating 나 line comment 에 붙여질 수 있습니다.
- 행에 있어서 선택된 행의 수가 클립보드에 있는 수보다 작을 경우 에러 메시지가 나타나고 붙여넣기 기능은 상실될 것입니다. 선택된 행의 수가 클립보드에 있는 수보다 작을 경우 여분의 행은 삭제될 것입니다. 오직 셀만 선택된다면 행은 그 행 위에 붙여질 것입니다.

## VDT 에서 drag and drop-cut 하기:

1. Variable Declaration Table 을 여십시오.
2. 단일 또는 여러 변수 내에서 변수와 셀을 선택하십시오
3. 선택된 변수와 셀 위에 마우스를 고정시키십시오.
4. 첫 번째 마우스 버튼을 누르고 있으십시오.
5. 적당한 영역으로 변수와 셀을 이동시키십시오.
6. 선택물을 드롭시키기 위해 마우스 버튼을 떼십시오. 만약 이 선택물이 새로운 위치에 맞는다면 선택된 변수와 셀은 원 위치에서 없어지고 새로운 위치에 놓일 것입니다. 만약 그렇지 않다면 drag and drop 작동을 취소시키십시오.

## VDT 에서 drag and drop-copy 하기:

1. Variable Declaration Table 을 여십시오.
2. 단일 또는 여러 변수 내에서 변수와 셀을 선택하십시오
3. 선택된 변수와 셀 위에 마우스를 고정시키십시오.
4. Ctrl key 를 누르고 있으십시오.
5. 첫 번째 마우스 버튼을 누르고 있으십시오.
6. 적당한 영역으로 변수와 셀을 이동시키십시오.
7. 선택물을 드롭시키기 위해 마우스 버튼을 떼십시오. 만약 이 선택물이 새로운 위치에 맞는다면 선택된 변수와 셀은 원 위치에 복사되고 새로운 위치에 놓일 것입니다. 만약 그렇지 않다면 drag and drop 작동을 취소됩니다.

## VDT drag and drop-cut 하기에 대한 세부사항

- Dragging and Dropping 으로 발생한 값으로의 변경은 영향을 받는 테이블이나 에디터에도 나타날 것입니다.
- 변수의 Drag and drop 는 동일한 VDT tab 에서만 가능합니다.
- 선택물이 하나이상의 범위를 가지고 있다면, 행 인증은 수행됩니다. 만약 선택된 영역에 있는 셀이 새로운 위치에서 쓸모없어 진다면 모든 drag and drop 동작은 취소됩니다.
- 완전한 행이 선택되지 않으면 다음 셀들은 자르기를 할 수 없습니다: Type, Length, Scope, Retentive, Override
- 선택물이 VDT 의 마지막 행으로 드롭된다면, 새로운 행은 드롭을 수용하기 위해 생성됩니다.
- 선택물이 드롭되었을 때, 기존 셀은 겹쳐집니다.
- VersaPro 는 비연속 셀들은 drag and drop 을 할 수 없습니다.

## VDT 에서 LD 나 IL Editor 로 drag and drop 하기:

1. IL 나 LD Editor 에 있는 블록을 여십시오.
2. Variable Declaration Table 을 여십시오.
3. VDT 에 있는 변수를 선택하십시오.
4. 선택된 변수에 마우스를 고정시키십시오.
5. 첫 번째 마우스 버튼을 누르고 있으십시오.
6. IL Editor 에 있는 기존 명령어의 오퍼랜드나 LD Editor 에 있는 요소나 오퍼랜드로 변수를 이동시키십시오.
7. 변수를 드롭하기 위해 마우스 버튼을 떼십시오. 변수는 선택된 명령어에 대한 오퍼랜드가 될 것입니다.

## Editor 로 변수 Drop 하기에 대한 세부사항

- 선택물을 드롭할 때, 선택물은 기존 셀에 드롭됩니다.
- 변수는 기존 요소나 오퍼랜드에만 드롭될 수 있습니다.
- 만약 변수가 새로운 위치에서 쓸모없어 진다면 모든 드롭 작동은 취소됩니다.
- 여러 행은 VDT 에서 IL Editor 로 drag and drop 될 수 없습니다. VDT 에서 하나 이상의 행은 LD Editor 로 드롭될 수 없습니다.

6

# *Variable Declaration Table 유틸리티*

VersaPro 는 변수를 소트하고, 변수를 삭제할 뿐만 아니라 VDT 를 지우고 저장할 수 있는 여러 유틸리티를 제공합니다.

이 섹션은 VDT 에 의해 제공되는 유틸리티 사용 방법을 서술합니다.

## 변수 정렬하기

VersaPro 는 저장값 컬럼에 따라 오름차순이나 내림차순으로 VDT 에서 정렬할 수 있습니다.

### VDT 에서 변수 정렬하기

1. Variable Declaration Table 을 여십시오.
2. 빠른 선택 버튼을 클릭함으로써 VDT 에 있는 컬럼을 선택하십시오. VersaPro 는 저장값 컬럼을 정렬할 수 없습니다.
3. 보기 메뉴에서 소트를 선택하고 나타난 서브메뉴에서 오름차순이나 내림차순을 선택하거나 Context-Sensitive Menu 에서 Sort Ascending 이나 Sort Descending 선택하십시오.

## 변수와 VDT 의 컨텐츠 삭제하기

변수는 래더 로직에서 래더 요소를 지우지 않거나 명령어 리스트 로직에서 명령어를 지우지 않고 Variable Declaration Table 에서 지울 수 있습니다. 제거된 변수가 있는 곳의 요소들은 이름이 붙여지지 않습니다.

또한 VersaPro 는 Variable Declaration Table(VDT 를 지움으로써)의 컨텐츠를 삭제할 수 있습니다. 이러한 삭제 기능을 수행함으로써 Global, Local 과 모든 tab 은 물론 프로그램 로직에서 사용된 모든 VDT 의 변수는 지워집니다. 로직 자체는 영향을 받지 않습니다. VDT 는 폴더에 남아 있고 항상 Folder Browser 에 디스플레이 됩니다. 이것은 지워지기만 하지 삭제되지는 않습니다.

### VDT 에서 변수 삭제하기

1. Variable Declaration Table 를 여십시오.
2. 삭제하고자 하는 변수를 선택하십시오.
3. 편집 메뉴에서 Delete 를 선택하거나 Delete key 를 누르십시오. 변수는 VDT 에서 삭제됩니다.
4. 변수는 프로그램 로직에서 삭제됩니다.
   - Ladder Diagram block 에서, 삭제된 변수의 이름은 요소에서 제거되지만 변수가 할당되어 있는 요소는 제거되지 않습니다.

- Instruction List block 에서, 삭제된 변수의 이름은 오퍼랜드에서 제거되고 이들 어드레스는 Reference Address 에서 제거됩니다. 하지만, 변수가 할당되어 있는 릴레이와 명령어는 제거되지 않습니다.

**Note:** 단일 변수가 삭제됨에 따라 변수에 있는 모든 필드(모든 행)는 선택되어야 합니다. 만약 모든 필드가 선택되지 않았다면 삭제 기능을 수행할 때, 선택된 필드는 지워지지만 변수는 기존에 남아 있습니다.

### Variable Declaration Table 지우기

1. Folder Browser Variable Declaration Table 을 밝게 하십시오.
2. 편집 메뉴나 Context-Sensitive menu 에서 Delete 를 선택하거나 Delete key 를 누르십시오. 프로그램에서 모든 변수를 정말로 지우기 원하십니까 하는 확인 메시지가 나타날 것입니다.
3. 변수를 삭제하려면 OK 를 누르십시오. 그렇지 않으면 Cancel 을 클릭하십시오.
   - 모든 글로벌 변수와 로컬 변수는 VDT 에서 삭제됩니다. 변수는 Global, Local 이나 모든 테이블에 남아 있지 않습니다.
   - 프로그램 로직에서 모든 변수는 제거됩니다.
   - 시스템 변수는 시스템 테이블에 남아 있습니다. 시스템 변수 결코 지울 수 없습니다.

## VDT 에서 편집 저장하기

### Declaration Table 저장하기:

1. Variable Declaration Table 에서, 툴바에 있는 Save Folder button  을 클릭하고, File 이나 Context-Sensitive menu 에서 Save 를 선택하거나 Ctrl + S 를 누르십시오.
2. VDT 데이터는 저장됩니다.

**Note:** 폴더나 애플리케이션을 닫을 때 VDT 는 자동으로 저장됩니다.

# VDT 압축하기

자르기, 복사하기, 붙여넣기 및 삭제하기 기능으로 VDT 를 편집할 때, VDT 는 편집에 대한 정보를 가지고 있습니다. 이 편집 과정은 VDT 가 실제로 사이즈가 커지는 원인이 됩니다. 대용량의 편집이 특별한 VDT 에 배열이 된다면, 때때로 VDT 의 편집과정을 제거하는데 사용되는 사이즈를 줄일 수 있습니다. 폴더 메뉴에서 압축 옵션은 이를 가능하게 해줍니다.

## VDT 압축하기:

1. 폴더를 여십시오.
2. Variable Declaration Table 을 여십시오.
3. Variable Declaration Table 을 만드십시오.
4. Folder 메뉴에서 압축을 선택하십시오.

VDT 는 압축됩니다.

# Highest References Used 보기

모든 메모리 형태에 대해 사용된 레퍼런스 리스트를 보기 위해 VDT Context Sensitive menu 에서 Highest References Used 를 선택하십시오

# 사용하지 않는 변수 찾기

- Folder 메뉴에서, Find Unused Variables 를 선택하십시오. 사용하지 않은 변수는 Name, Scope 와 Address 로 Information window 에 리스트되어 있습니다.

# 변수의 Cross-Reference 리스트 생성하기

- 도구 메뉴나 Folder browser CSM 메뉴에서, View On line Cross References 를 선택하십시오. Information window 는 레퍼런스 타입에 의해 선택된 블록에서 사용된 모든 변수의 리스트를 디스플레이합니다. Coss Reference 또한 변수를 사용하는 Rung 과 Instruction 를 리스트합니다.

# 변수 불러오기와 불러내기

VersaPro 는 Shared Name File Format 에서 사용하는 외부 위치에서 변수를 불러오고 불러낼 수 있습니다. 이 섹션은 SNF Files 로 변수를 불러오거나 SNF Files 에서 불러내는 방법을 기술합니다.

## 변수 불러오기:

1. SNF 파일을 불러오고자 하는 폴더를 여십시오.
2. VDT CSM 나 Tools 메뉴에서 Import Variable 을 선택하십시오. 불러오고자 하는 파일의 위치를 묻는 대화창이 나타날 것입니다.
3. 불러오고자 하는 파일의 위치를 정하십시오.
4. Open button 을 누르십시오. 파일은 불러오고 변수는 VDT 에 디스플레이 됩니다.

## SNF File 불러오기에 대한 세부사항

- VersaPro 는 SNF files 의 Header 부분을 무시합니다.
- 불러온 SNF file 은 PT_ID field 가 포함되어야 하고 그 필드는 첫 번째 Non-comment field row 에서 첫 번째 필드가 되어야 합니다.
- 데이터 등록 수는 Field Names 의 수와 일치해야 합니다.

VersaPro 는 다음 필드 이름을 지원합니다:

- PT_ID (name)
- ADDR
- DESC
- PT_TYPE
- LENGTH
- RETENTIVE
- OVERRIDE
- STORED_VAL
- SHOW_EXT

다른 모든 필드 이름과 일치하는 데이터 등록은 무시됩니다.

- 빈 데이터는 VDT 와 동일한 초기값으로 등록됩니다.
- 행 인증에 위배되는 데이터 등록은 특수한 필드(PT_TYPE field 이외에)에 대한 부적절한 등록을 포함하고 있거나 반복 변수가 충돌 에러를 발생시킵니다.
- 불러온 모든 SNF 변수는 다른 블록이 어드레스 필드에 명시되지 않았다면 글로벌 테이블에 추가됩니다.

- 불러온 SNF 변수는 기존 변수 아래에 추가됩니다.
- Conflict dialog 에서 Cancel 을 선택하면, 모든 불러오기 프로세스는 취소되고 모든 데이터는 제거됩니다.
- 인용 마크가 포함된 설명은 변수에서 설명과 추가 텍스트사이에서 발생하는 분리점을 결정하기 위해 인용마크로 시작과 종료해야 합니다.

## 변수 보내기:

1. VDT 에서, 보내고자 하는 변수를 선택하십시오.
2. VDT CSM 나 툴 메뉴에서 Export Variable 을 선택하십시오. 선택된 변수가 없다면 보내기 위해 최소한 한 변수를 선택해야 한다는 에러 메시지가 나타날 것입니다. 보내는 파일을 저장하고자 하는 위치를 결정하라는 대화창이 나타납니다.
3. 위치를 선택하십시오. 기본 보내기 위치는 사용된 마지막 Import/Export 디렉토리입니다. OK 를 클릭하십시오.

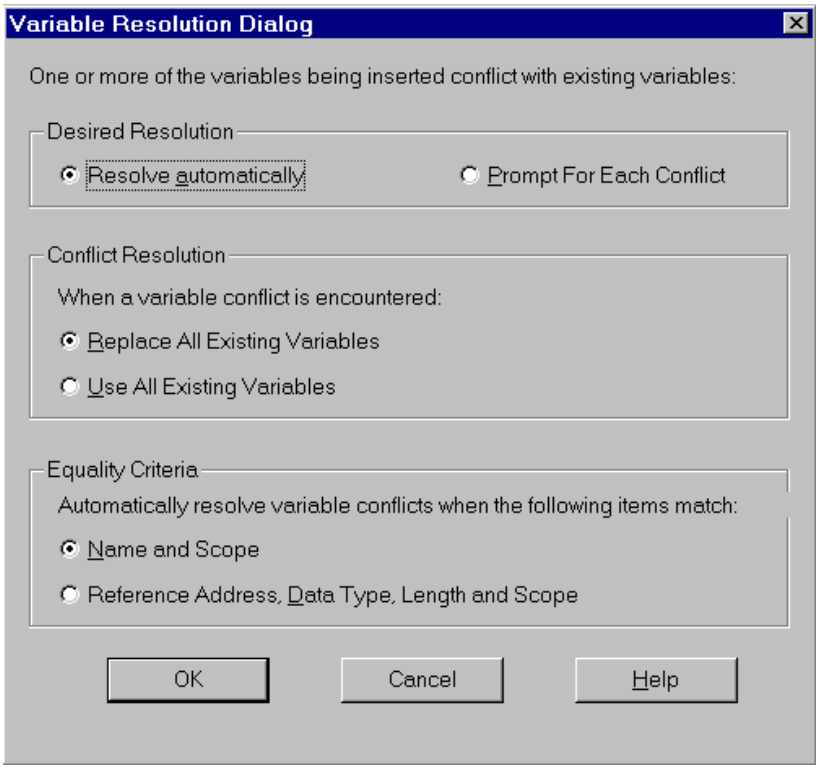
## SNF File 보내기에 대한 세부사항

- 보내는 파일은 다음과 같습니다.:
  - PT_ID (name) – 변수 이름
  - ADDR (로컬 변수에 대한 "어드레스, 블록 이름" 포맷) – 변수 레퍼런스 어드레스와 스코프
  - DESC – 변수 설명
  - PT_TYPE – 변수 형태
  - LENGTH – 변수 길이
  - RETENTIVE – 변수 보유 상태
  - OVERRIDE – 변수 오버라이드 상태
  - STORED_VAL - comma delimited format 저장값
  - SHOW_EXT – HMI 에 의해 볼 수 있는 변수
- SNF 헤더의 $$&&FileType 부분은 I/O 가 될 것입니다.
- VersaPro 는 SNF 헤더의 ##DelVarDEcl 부분은 보낼 수가 없습니다.
- SNF 파일에서 추가 코멘트는 외부 에디터를 사용하여 추가시킬 수 있습니다.
- 보내기 파일의 기본 확장자는 *.SNF 입니다.

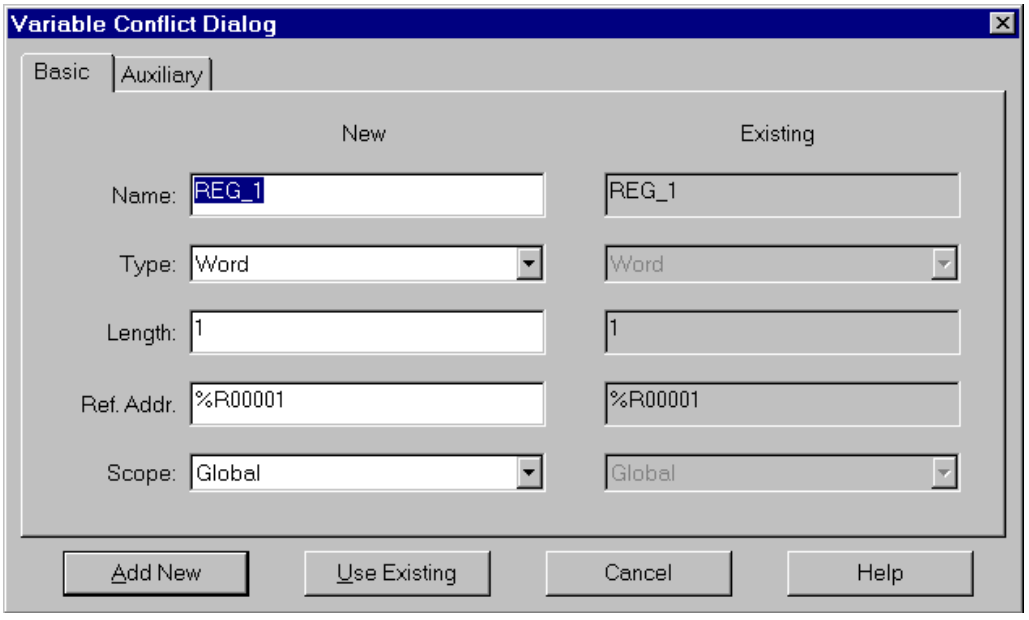
# *Variable Resolution*

대화 창은 변수 충돌을 쉽게 해결해 줍니다. 기존 변수를 가질 수 있거나 이 변수를 겹쳐 쓸 수 있습니다. 이 대화 창은 하나 이상의 변수가 기존 변수와

충돌하게 되는 Paste, Import SNF 또는 Insert Block 작동의 결과로서 나타납니다.




충돌에 대한 프롬프트를 선택하면 다음 대화 창이 나타날 것입니다.



변수에 대한 Description, Stored Value, Scope, Override, Retentive 와 형식을 입력하려면 Auxiliary tab 을 선택하십시오(이들 값은 VDT 에서도 편집할 수 있습니다.).

Chapter 7

# PLC Hardware 구성하기

Stand Alone Hardware Configuration(HWC)은 Series 90-30, VersaMax, VersaMax Nano/Micro 와 Series 90 Micro PLC 에 대한 모든 구성 정보뿐만 아니라 Ethernet Global Data 구성 정보를 포함하고 있습니다.

또한 시스템에 소비 전원과 추가된 모듈과 맵핑하는 I/O 를 모니터할 수 있는 툴들을 제공합니다. HWC 를 입력할 때, 디스플레이되는 기본 랙과 CPU 구성은 VersaPro 에서 Options dialog box 를 통해 설정됩니다. 구성 정보와 시스템의 물리적 하드웨어와 매치시키는 것은 중요합니다.

PLC 하드웨어를 구성하는데 전형적인 단계는 다음과 같은 내용을 포함하고 있습니다:

- 랙 타입과 파워 서플라이 구성하기
- CPU 구성하기
- 모듈 구성하기와 애플리케이션에 파라미터 만들기
- VersaPro 에서 PLC 로 저장시킬 수 있도록 HWC 저장하기

특수 모듈용 구성 파라미터에 대한 상세 정보는 온라인 도움말 기능을 통해 가능합니다. 이러한 정보에 액세스하기 위해 모듈용 Parameter Editor window(모듈 더블 클릭)을 연후, Help menu 에서 Module Help 를 선택합니다.

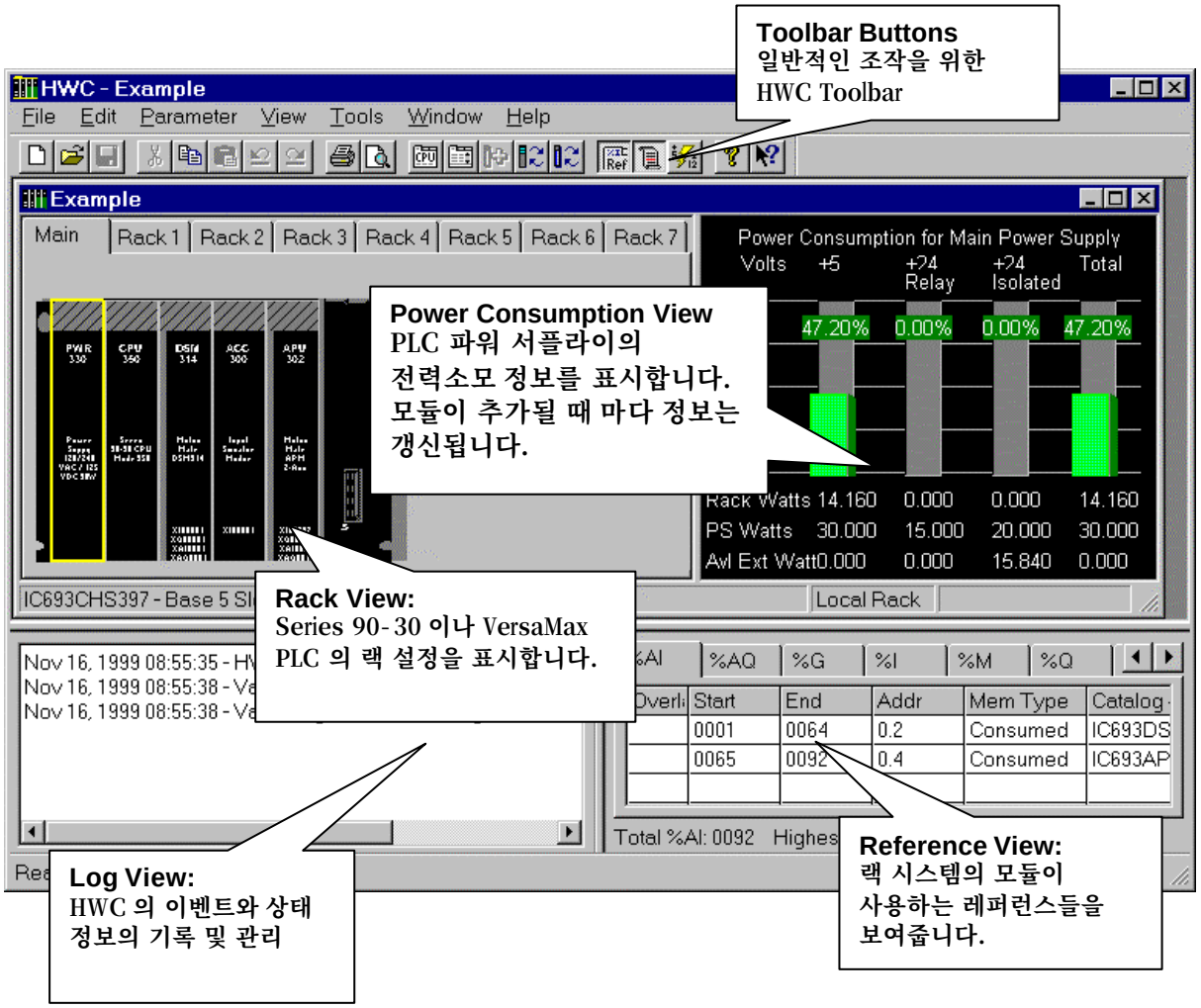
## 목차

이 장은 다음과 같은 정보를 제공합니다:

7

# 하드웨어 구성 액세스하기

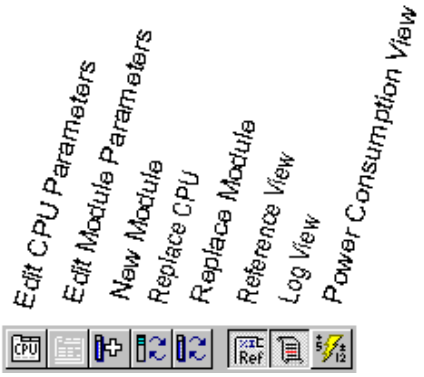
VersaPro 에서 HWC 에 액세스하기 위해, 폴더 브라우저에서 Hardware Configuration icon 을 더블 클릭합니다. 구성된 CPU hardware 용 랙 시스템은 모듈의 전력 소모 통계를 보여주는 그래픽, 하드웨어 구성에서 사용되는 레퍼런스를 보여주는 테이블과 HWC 이벤트의 기록을 유지하는 로그와 함께, 나타날 것입니다.



# 하드웨어 구성 툴바

HWC 을 시작하면 메인 창이 디스플레이됩니다. Hardware Configuration 에서 작업을 하는 동안 Hardware Configuration 의 공통적으로 사용되는 영역과의 액세스를 제공하는 Hardware Configuration toolbar 를 디스플레이할 수 있습니다:

Hardware Configuration toolbar 를 보기 위해 View 메뉴로 가서 Toolbar 다음에 있는 체크마크를 배치시키십시오.




# Parameter Editor 사용하기

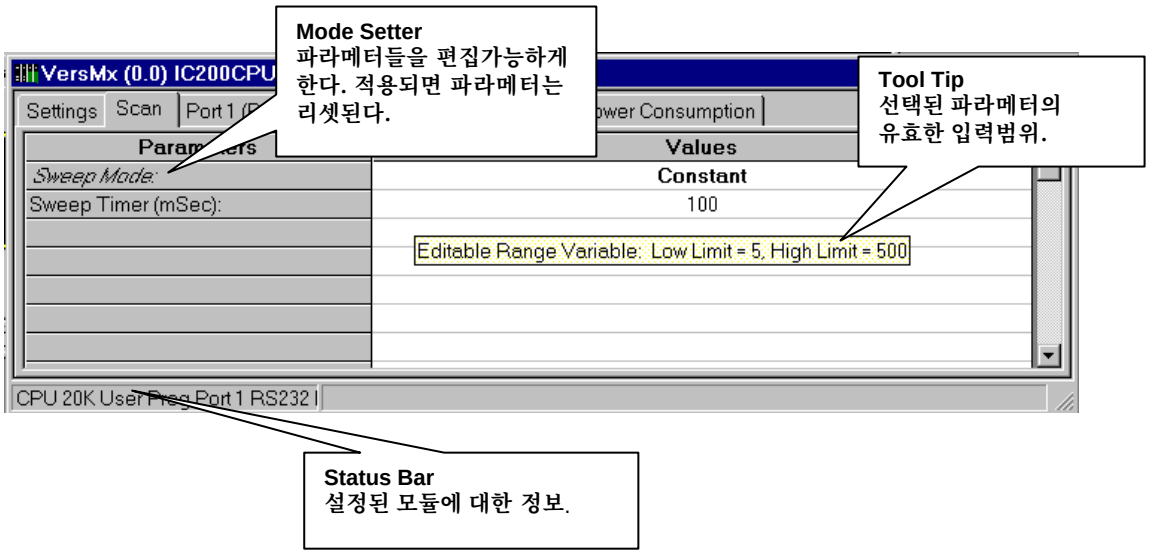
모듈 파라미터를 구성하기 위해 모듈을 삽입하고 대체시킬 때 자동으로 나타나는 Parameter Editor 를 사용할 수 있습니다. 모듈에 대한 파라미터를 변경시키기 위해 Context-sensitive menu 에서 Configure Parameters 를 선택하거나 Edit, Module Operations submenu 에서 Configure Parameters 를 선택하여 Module 을 더블 클릭합니다.

Parameter Editor 는 최소한의 인력으로 시스템을 구성할 수 있도록 윈도우 호환 편집 설비의 확장 세트를 제공합니다.

- 파라미터는 크기 재설정, 최대화와 최소화시킬 수 있는 편집 창에 디스플레이됩니다.
- 동시에 여러 변수 보기를 열 수 있습니다. 이러한 기본값은 다른 파라미터를 편집하는 동안 한 모듈의 파라미터를 볼 수 있게 해주고 모듈 보기간의 단일 파라미터 값을 쉽게 복사하고 붙일 수 있습니다.
- 편집 창은 Tabbed Mode(기본값)나 Spreadsheet Mode 에서 디스플레이시킬 수 있습니다(모든 파라미터 그룹이 단일 스크롤이 가능한 View 에서 디스플레이하고 편집될 수 있습니다.). 디스플레이를 변경시키기 위해 View menu 에서 Parameter Edit 를 선택하십시오.
- 윈도우의 Title Bar 는 랙 시스템 이름, 모듈 랙/슬롯 어드레스 및 모듈 카탈로그 넘버와 함께 관련 모듈을 독특하게 인식합니다.

**Note:** 모션 모듈과 같은 보다 복잡한 모듈은 *Mode Setters* 라 불리는 키 파라미터를 가지고 있습니다.

다른 파라미터에 영향을 미치는 모든 파라미터(Mode Setters 를 포함) 는 *italics* 체로 작성했습니다. 이들 파라미터에 대한 파라미터 관계의 설명을 구하려면 Editor window 에 있는 **F1** 을 누르고 Help index에서 파라미터 이름을 선택하거나 오른쪽 마우스를 클릭하여 What's This from the CSM 를 선택하십시오.




## Tips 편집하기

각 파라미터의 유효값은 파라미터 셀에 마우스 커서를 갖다 댈 때 , 나타나는 tool tip 에 디스플레이됩니다:

- Spreadsheet cell 에 입력하는 것과 유사한 방법으로 선택된 셀에 직접 입력하거나 Data Entry Tool 을 사용합니다. Data Entry Tool 을 열려면, 파라미터 셀에서 오른쪽을 클릭하여 Context-sensitive menu 에서 Data Entry Tool 를 선택하고 F2 key 를 누르거나 더블 클릭하십시오.
- 만약 유효값이 독특한 초기 문자를 가진 선택의 리스트로 구성되어 있다면 셀에 있는 첫 문자를 간단하게 입력함으로써 값을 설정합니다.

다른 파라미터 값 셀에서 한 파라미터를 **Copying and Pasting** 또는 **Dragging and Dropping** 하여 여러 파라미터(동일한 형태)에 대한 동일한 값을 입력하면 시간을 절약할 수 있습니다. Copy/Paste 와 drag/drop 은 동일한 모듈 보기 내에서 사용하거나 모듈 보기 사이에서 사용될 수 있습니다.

Drag/drop 은 다음과 같이 키보드를 이용해 작업합니다:

- 소스 셀에서 드롭 셀로 데이터를 복사하려면 아무 키를 누르지 말고 드래그하는 동안 **Ctrl** key 를 누르십시오.

- 소스 셀에서 드롭 셀로 데이터를 이동시키려면 드래그하는 동안 **Shift** key 를 누르십시오.
- 소스 셀에서 소스 셀과 드롭 셀 사이(드롭 셀 포함)에 있는 모든 셀로 데이터를 복사하려면 드래그하는 동안 **Alt** key 를 누르십시오. 이것은 drag and drop 을 떠맡고 있는 이것은 같은 편집 창에서 발생하고 소스 셀과 드롭 셀은 동일한 컬럼에 존재합니다. 이 특성은 동일한 범위를 지닌 셀의 모든 범위를 빠르게 채울 수 있게 해줍니다.

**Note:** 편집을 시작한 이후 변경된 파라미터는 굵게 나타납니다.

## 변경 실행취소하기

- Start of Edit Session value 이나 Factory Original (기본값) setting 에서 단일 파라미터를 리셋하려면, 파라미터 상에서 마우스 오른쪽버튼을 클릭하여 Context-sensitive menu 에서 Reset Parameter 를 선택하십시오.
- tab 상에서 모든 파라미터를 리셋하려면, tab 의 어디서든지 마우스 오른쪽 버튼을 클릭하여 Context-sensitive menu 에서 Reset Parameter 를 선택하십시오.
- 변경에 대해 실행취소하려면, 편집 메뉴에서 Undo 를 선택하거나 Editor toolbar 에서 Undo button 을 클릭하십시오.

  Editor 는 1~32 레벨의 Undo/Redo(기본값은 10 입니다.)를 제공합니다. Tools menu 에서 Options 을 선택함으로써 Undo/Redo 레벨을 설정할 수 있습니다.

**Note:** Mode Setter parameter 에서 나타난 변경을 Undo 시킬 수 있습니다.

## 에러 통지

잘못된 값은 Error Color(기본값에 의해 빨간색)로 디스플레이되고 ToolTip 도움말이 에러에 대비해 제공됩니다.

- Error Color를 변경하려면 Tools menu 에서 Options 을 누르십시오.
- 잘못된 설정에 대해 ToolTip 을 보려면 파라미터 셀에 마우스 커서를 고정시키십시오.

## 에러 수정하기

잘못된 값을 자동으로 수정하려면, 그 값에서 오른쪽을 클릭하여 Context-sensitive menu 에서 Auto Correct 를 설정하십시오. 만약 에러가 포함된 구성을 닫으려 한다면, Parameter Error List dialog box가 나타날 것입니다. 모든 에러를 자동으로 수정하려면 이 대화창에 기입하고 Auto Correct button 을 클릭하십시오.

# HWC 용 단축 키

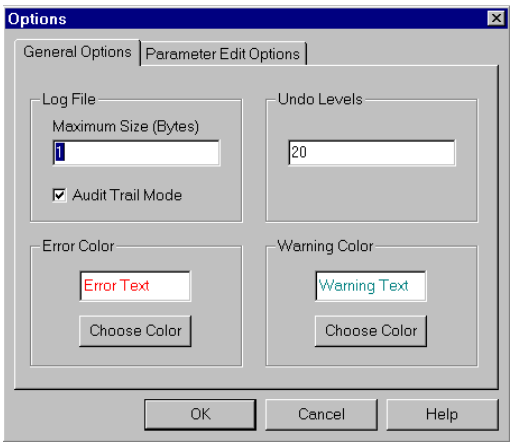
메뉴 바 선택, Context-sensitive menu 또는 단축키를 이용해 Hardware Configuration 에서 정보를 편집할 수 있습니다.

- 메뉴 바를 이용해 모듈을 편집하려면 구성하려고 하는 모듈과 슬롯을 클릭하고 Edit 메뉴로 가서 적절한 기능을 선택하십시오.
- 오른쪽 마우스 버튼을 이용해 모듈을 편집하려면 구성하려고 하는 모듈과 슬롯을 클릭하고 오른쪽 마우스 버튼을 클릭하여 적절한 기능을 선택하십시오.
- 표준 자르기, 복사 및 붙여넣기 기능을 이용해 모듈을 편집하려면 모듈을 밝게 하고 Edit menu options 이나 CTRL key 를 이용하십시오.
- 모듈을 복사하려면 복사하고자 하는 모듈을 클릭하고 CTRL+C 를 누르십시오. 모듈을 붙여넣기 하려면 새로운 슬롯을 클릭하고 CTRL + V를 누르십시오.
- 모듈을 자르기 하려면(다른 슬롯에 붙여넣기 할 수 있음), 모듈을 클릭하고 CTRL+X 를 누르십시오. 모듈을 삭제하려면 모듈을 클릭하고 Edit Menu 에서 Delete 를 선택하거나 Delete key 를 누르십시오.
- 이전 실행을 취소하려면, CTRL + Z 을 누르십시오.
- 슬롯간 모듈을 이동하기 위해 drag and drop 기능을 사용할 수 있습니다. drag and drop 기능을 사용하려면, 이동시키려는 모듈을 밝게 하고 왼쪽 마우스 버튼을 클릭한 상태에서 새로운 슬롯으로 모듈을 드래그 하십시오.

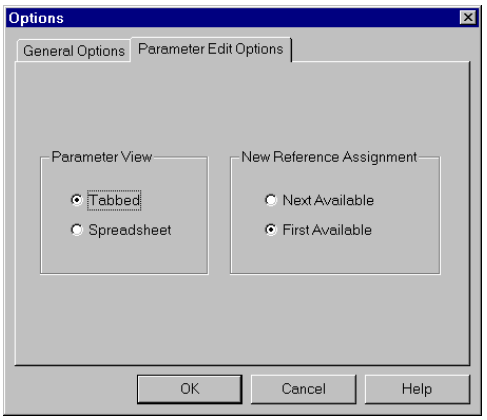
- 창을 닫으려면, Ctrl + F4를 누르십시오.
- HWC를 닫으려면, Alt + F4를 누르십시오.

# Hardware Configuration 에서 옵션 설정하기

HWC options 을 설정하려면, Tools 메뉴로 가서 Options 을 선택합니다. HWC System Log 의 크기와 컨텐츠를 조절하고, Audit Trail Mode 를 하려면, Undo Levels 의 수를 설정하고 Error and Warning 색깔과 General Options tab 을 선택합니다.



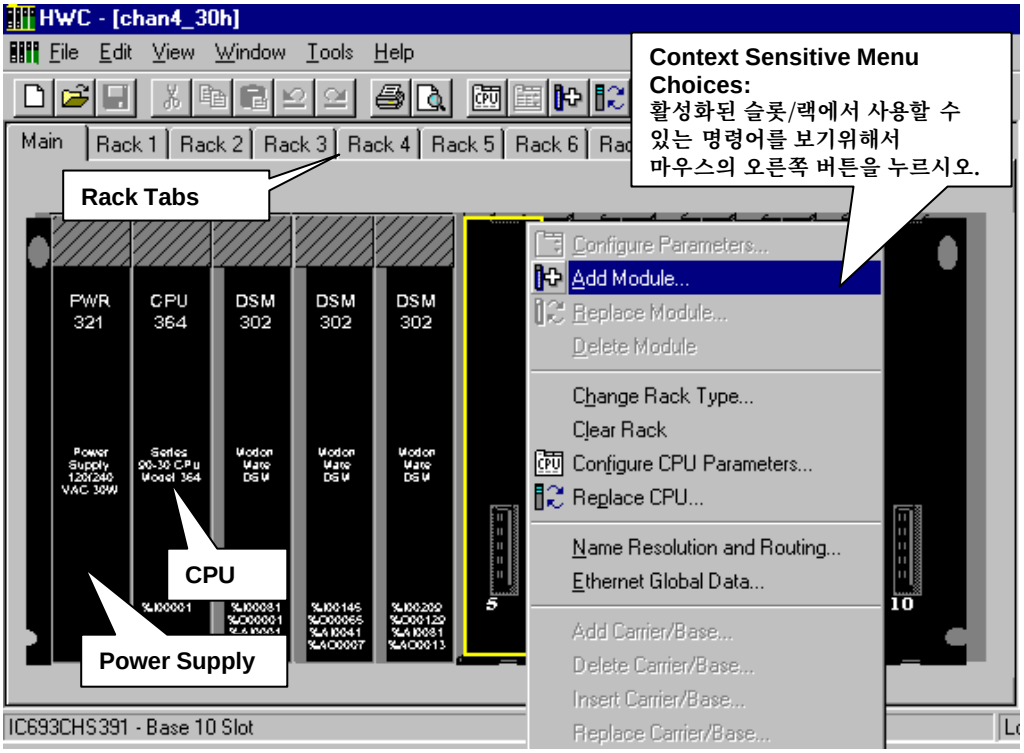

Parameter Editor window 의 포맷과 다음 레퍼런스 할당하는 모드를 변경하려면, Parameter Edit Options tab 을 선택하십시오.



7

# *Series 90-30 Rack 구성하기*

Series 90-30 에 대한 Hardware Configuration 을 입력할 때, 기본 보기는 Rack (Main)입니다. 기본 CPU 모델은 CPU351 입니다.




## Rack Type 변경하기

기본 랙 타입은 Series 90-30 용 10-slot base 입니다. 랙 타입을 변경하려면 다음과 같은 단계를 수행하십시오:

1. Rack window 아무 곳이나 커서를 고정시킨 후 오른쪽 마우스 버튼을 누르십시오.
2. Change Rack Type 을 선택하십시오.
3. 시스템에 맞는 타입의 설명과 카탈로그 번호를 클릭한 후 OK button 을 클릭하십시오.

**Note:** Change Rack Type 은 비 내장형 CPU 에서만 작동됩니다(CPU331 과 그 이상 ).

# Power Supply 와 CPU 변경/구성하기

## Power Supply 구성하기

Series 90-30 에 대한 기본 파워 서플라이는 PWR321 입니다. 기본 파워 서플라이를 변경하려면 다음 단계를 수행하십시오:

1. Rack window 에서 파워 서플라이를 클릭하십시오.
2. 오른쪽 마우스 버튼을 클릭하여 Replace Module 을 선택하십시오.
3. 시스템에 맞는 타입의 설명과 카탈로그 번호를 클릭한 후 OK button 을 클릭하십시오.
4. 무엇인가 나타나면 응답하십시오.
5. 기본 파워 서플라이를 대체할 때, 상세한 전력 소모 정보를 볼 수 있도록 Parameter Editor 가 나타납니다. 만약 기본 파워 서플라이를 사용한다면 세부 사항을 보기위해 Rack window 에서 파워 서플라이 슬롯을 더블 클릭할 수 있습니다.
6. 모든 단계를 마치려면 OK button 을 클릭하십시오.

## CPU 구성하기

Series 90-30 의 기본 CPU 는 CPU351 입니다. 기본 CPU 를 변경하려면 다음 단계를 수행하십시오:

1. Rack view 에서 오른쪽 마우스 버튼을 클릭하고 Replace CPU 를 선택합니다.
2. 시스템에 맞는 CPU 의 설명과 카탈로그 번호를 클릭한 후 OK button 을 클릭하십시오.

## CPU Parameter 구성하기

CPU 파라미터를 구성하려면 다음 단계를 수행하십시오:

1. CPU 를 대체할 때, Parameter Editor window 이 나타납니다. 만약 기본 CPU 를 사용한다면 CPU 슬롯을 더블 클릭하십시오.
2. Parameter Editor window 내에서 보거나 수정하려 하는 파라미터에 속하는 tab 을 클릭하십시오.
3. 편집하고자 하는 Values fields 를 클릭하십시오(페이지 7-4 “Editing Tips” 참조).
4. CPU 파라미터 편집을 마치려면 window close button.  을 클릭하십시오

7

# Rack Module 추가하기/구성하기

랙에 모듈을 추가하려면 다음 단계를 수행하십시오:

1. 대체하고자 하는 모듈이 있는 슬롯을 선택하십시오.
2. 빈 슬롯을 더블 클릭하거나 오른쪽 마우스 버튼을 클릭하여 Add Module 을 선택하십시오(만약 예전에 구성한 모듈로 대체하려면 오른쪽 마우스 버튼을 클릭하여 Replace Module 을 선택하십시오.).
3. 필요한 I/O 모듈 타입을 설명한 tab 을 클릭하십시오(Discrete Input, Discrete Output, Analog Input, Analog Output, Communications, ….).
4. 시스템에 맞는 모듈의 설명과 카탈로그 번호를 클릭한 후 OK button 을 클릭하십시오.

## Module Parameter 구성하기

모듈 파라미터를 구성하려면 다음 단계를 수행하십시오:

1. 모듈을 추가하거나 대체할 때, Parameter Editor 가 나타납니다. 또한 Rack window 에서 모듈 슬롯을 더블 클릭함으로써 이 dialog box 와 액세스할 수 있습니다.
2. Parameter Editor 내에서, 보거나 수정하려 하는 파라미터에 속해 있는 tab 을 클릭하십시오 .
3. 편집하고자 하는 Values fields 를 클릭하십시오(페이지 7-4 “Editing Tips” 참조).
4. 모듈 파라미터 편집을 마치려면 window close button  을 클릭하십시오

# 확장 랙 구성하기

확장 랙은 메인 랙과 동일한 방법으로 구성됩니다. 확장 랙에 모듈을 추가하려면 Rack tab 을 클릭하여 하드웨어 일치하는 모듈 구성을 추가하십시오.

# 랙 모듈 삭제하기

모듈을 삭제하기는 랙에서 모듈이 제거하는 것입니다. 만약 모듈을 삭제한 후 Hardware Configuration 을 저장하려면 삭제된 모듈에 대한 구성 정보는 없어질 것입니다. 모듈을 삭제하려면 다음 단계를 수행하십시오:

1. 삭제하려고 하는 모듈을 선택하십시오.
2. Delete key 를 누르거나 오른쪽 마우스 버튼을 누른 후 Delete Module 을 선택하십시오.

# 랙 지우기

랙을 지우기는 파워 서플라이와 CPU 를 제외한 모든 모듈을 제거하는 것입니다. 랙을 지운 후 Hardware Configuration 을 저장하려면 삭제된 모듈에 대한 구성 정보는 없어질 것입니다. 모듈을 삭제하려면 다음 단계를 수행하십시오:

1. 랙 아무 곳이나 클릭하십시오.
2. 오른쪽 마우스 버튼을 누르고 Clear Rack 을 선택합니다.
3. Confirmation dialog box 가 나타날 것입니다. 랙 지우는 기능을 확인하려면 Yes button 을 누르십시오.

# Ethernet Global Data 구성

Ethernet Global Data 는 하나의 디바이스(Producer)가 하나 또는 그 이상의 다른 디바이스(Consumers)에 규칙적으로 정해진 전송 속도로 데이터를 전송(exchange)할 수 있도록 해줍니다. Ethernet Global Data 는 Series 90-30 CPU364 로 구성할 수 있습니다.

**Caution**

**Ethernet Global Data 교환은 디바이스 간에 샘플 데이터를 간단하고 효율적인 통신할 수 있도록 설계되었습니다. 중요한 데이터의 샘플이 손실이 생긴 경우 경고하는 이벤트를 발생하지 않습니다. 경고 이벤트를 하려면 SRTP 연결을 사용할 것을 권합니다.**

데이터 교환은 Producer ID(Identification Number)와 Exchange ID 를 통해 독특하게 정의되어 있습니다:

- 네트워크 상에서 각각의 디바이스를 교환하는 Ethernet Global data 는 독특한 고유 Producer ID 를 할당받아야 합니다. Producer ID 는 네트워크 상에서 Ethernet Global data 의 소스 디바이스를 독특하게 인식합니다. Producer ID 번호는 Dotted-decimal format(IP Address 와 같은)에 있습니다. 예를 들면 3.0.0.1 는 유효한 Producer ID 입니다.
- Exchange ID 는 전송되어지는 특수 변수나 변수의 그룹을 인식하는 독특한 번호입니다.

디바이스를 생산하고 소비하는 것은 각각에 대해 완전하게 비동기적으로 운용됩니다. 각 디바이스에 있어서, 데이터와 디바이스가 생산하고 소비하는 것은 불리하여 정의해야 합니다.

## Ethernet Interface Adapter 이름 구성하기

CPU364 는 Ethernet Global Data exchanges 에서 사용하려면, Adapter Name 을 명시해야 합니다. Ethernet Global Data exchange 을 정의할 때,

네트워크 어댑터가 교환을 산출하도록 명시하기 위해 Adapter Name 을 사용할 것입니다.

Ethernet Interface 에 대한 어댑터 이름을 명시하기위해 다음을 수행하십시오:

- Hardware Configuration 에서 CPU364 가 구성되어 있는지 확인하고 Ethernet 모듈에 대한 IP 어드레스가 정확한지 확인하십시오. 만약 그렇다면 충족된 슬롯을 더블 클릭하십시오.
- Parameter Editor 가 나타날 것입니다. Ethernet tab 을 선택하십시오.
- Adapter Name 이외에 Values field 를 클릭하여 어댑터 이름을 입력하십시오. 마치려면 OK button 을 클릭하십시오.

## Aliases Remote Ethernet Interfaces 용 Aliases 설정하기

원격 Ethernet Controller 는 Ethernet Global Data exchanges 에서 사용하려면, 카드의 IP 어드레스를 지닌 Alias 와 관련되어야 합니다. Ethernet Global Data exchange 를 정의할 때, 네트워크 어댑터가 교환을 산출할 것을 명시하기 위해 Alias 를 사용할 것입니다.

원격 Ethernet Interface 에 대한 Alias 를 명시하려면 다음 단계를 수행하십시오:

- Hardware Configuration 에서 Edit 메뉴로 간 다음 Rack Operations 를 선택하고 Name Resolution Routing(또는 오른쪽 마우스 버튼을 클릭하여 pop-up menu 에서 Name Resolution Routing 을 클릭하십시오)을 선택하십시오.
- Name Resolution and Routing Table dialog box 에서 Aliases tab 을 클릭하십시오.
- Add Entry button 을 클릭하십시오.
- Alias field 에서 원격 Ethernet Controller 를 조회하기 위해 사용하려는 이름을 입력하십시오.
- IP Address field 를 클릭하여 Ethernet Controller 의 IP 어드레스를 입력한 다음 OK button 을 클릭하십시오.

## Name Resolution 과 Routing Table 구성

Routing Table 은 PLC 간의 통신을 구성하기 위한 Advanced tool 입니다. Routing Table 은 Network Administrator 에서 지원이 없으면 구성할 수 없습니다. Routing Table 은 하나 이상의 게이트웨이가 구성될 때 구성해야 합니다.

### Routing Table 에 관련된 용어 정의 :

Destination Subnet ID - 송신된 데이터의 destination subnet 은 이 Subnet ID 에 의해 작성된다. 동일한 Subnet 은 Next Hop IP 주소로 전송되도록 합니다.

Destination Subnet Mask – Mask 는 destination subnet 의 환경과 관계 있습니다.

Next Hop IP Address - 데이터가 최종 목적지로 전달되도록 하는 IP 주소

Cost – 이 필드는 예약되어 있습니다. 기본값은 1 입니다.

## Routing Table Configuration 접근하기

Hardware Configuration 에서 편집 메뉴로 간 다음 Rack Operations 을 선택하여 Name Resolution and Routing 을 선택하십시오(또는 오른쪽 마우스 버튼을 클릭하여 Shortcut 메뉴에서 Name Resolution and Routing 을 선택하십시오). Name Resolution and Routing Tables dialog box 가 나타날 것입니다.

Name Resolution and Routing Tables dialog box 에서 Routing tab 를 클릭하십시오.

Add Entry button 을 클릭하십시오.

Destination Subnet ID field 에서 데이터를 루트하고자 하는 곳에 원격 서브넷의 subnet ID(in dotted decimal format)을 입력시키십시오(예, 4.12.20.0).

Destination Subnet Mask field 에서, 원격 서브넷과 관련된 Subnet Mask 를 입력하십시오(예, 255.255.255.0).

Next Hop IP Address field 에서, 데이터를 보내고자 하는 곳에 로컬 게이트웨이의 IP 어드레스를 입력한 후(예, 3.16.32.1), OK button 을 클릭하십시오.

## Subnet, Subnet ID, Subnet Mask

Subnet Mask 는 네트워크 상에 있는 노드에 IP 어드레스의 로컬 할당에 따라 보다 작은 서브넷에 대형 네트워크를 논리적으로 분할하는 메커니즘입니다. 서브넷에 명시된 Bit 와 같은 IP 어드레스를 가지고 있는 네트워크 상에 있는 노드들은 서로 *직접* 대화를 할 수 있습니다; 동일한 bit 에서 같지 않은 IP address 를 지닌 노드들은 게이트웨이나 라우터를 통해 *간접적으로* 대화를 해야 합니다.

Ethernet module user manual 은 부록으로 기술하였기 때문에 32bit 의 IP 어드레스 *net id* 부분과 *host id* 부분으로 나뉘어 집니다(IP 어드레스의 *class* 는 *net id* 에 얼마나 많은 bit 가 있어야 하며 *host id* 에 얼마나 많은 bit 가 있어야 하는지를 결정합니다.) 일반적으로 IP 어드레스의 *net id* 부분은 인터넷 인증에 의해 할당되고 *host id*(오른쪽에 있는)는 로컬 네트워크는 Local Network Administrator 에 의해 할당됩니다. *Subnetting* 은 부분적으로 옵션이며, 확장된 *net id* 나 *subnet id* 처럼 *host id* bits 의 지정된 수로 구성되어 있습니다. 추가된 *subnet id* bit 는 *net id* 에 근접해 있는 *host id* bits 에서 정상적으로 획득할 수 있습니다. Ethernet module

configuration 에서, 이들 비트는 1 로 명시되고 남아있는 *host id* bit 는 0 으로 명시됩니다. 예를 들면 다음과 같습니다:

| | | | |
|---|---|---|---|
| IP Address: | 11.22.33.44 | | |
| *Net Id* bits: | 255.0.0.0 | *Net Id*: | 11.0.0.0 |
| Added *Subnet Id* bits: | 0.255.0.0 | *Subnet Id*: | 0.22.0.0 |
| *Subnet Mask*: | 255.255.0.0 | "Extended" *Net Id*: | 11.22.0.0 |

Subnet id 와 subnet mask 는 Multiple Gateway 를 구성할 때 Routing Table 에 입력됩니다. *extended net id*(예를 들면, 위의 예에서 11.22.0.0 )는 subnet id 에 입력되어야 합니다.

"subnetted" network 에 있는 모든 노드는 mask 를 위해 동일한 값을 사용해야 합니다. 서브넷은 다양한 *subnet id*'s 에 의해 구별되고, IP 어드레스의 로컬 할당에 의해 생성됩니다.

## Ethernet Global Data Exchange 정의하기

생산되고 소비되는 PLC 를 교환할 때마다 교환되기 위해 운용 특성과 데이터를 명시해야 합니다. 다음 교환 한계를 주의하십시오:

- PLC system 에 대해 정의될 수 있는 익스체인지의 최대 수는 255 입니다. 이 숫자는 생산 및 소비된 익스체인지의 결합으로 나누어 집니다.
- 단일 익스체인지의 총 크기는 1400 byte 를 넘어설 수 없습니다. 총 크기는 리스트에 있는 모든 데이터 요소의 데이터 타입 길이의 합계가 되도록 정의합니다. 익스체인지 크기는 Ethernet Global Data dialog box 에 디스플레이됩니다.

Ethernet Global Data 교환의 정의하려면 다음 단계를 수행하십시오:

1. Hardware Configuration 에서 편집 메뉴로 간 다음, Rack Operations 을 선택하고 Ethernet Global Data 를 선택하십시오(또는 오른쪽 마우스를 클릭하여 팝업 메뉴에서 Ethernet Global Data 를 클릭하십시오).
2. Ethernet Global Data dialog box 에서 PLC 에 의해 생산되는 교환을 정의하기 위해 Produced Exchanges tab 을 클릭하거나 PLC 에 의해 소비되는 교환을 정의하기 위해 Consumed Exchanges tab 을 클릭하십시오.

## PLC Timing 고려사항

PLC 스캔 주기와 교환 주기 설정(생산된 교환에 대한 교환 주기와 소비된 교환에 대한 스캔 주기)이 비동기성일 때 다음 사항을 주의해야 합니다:

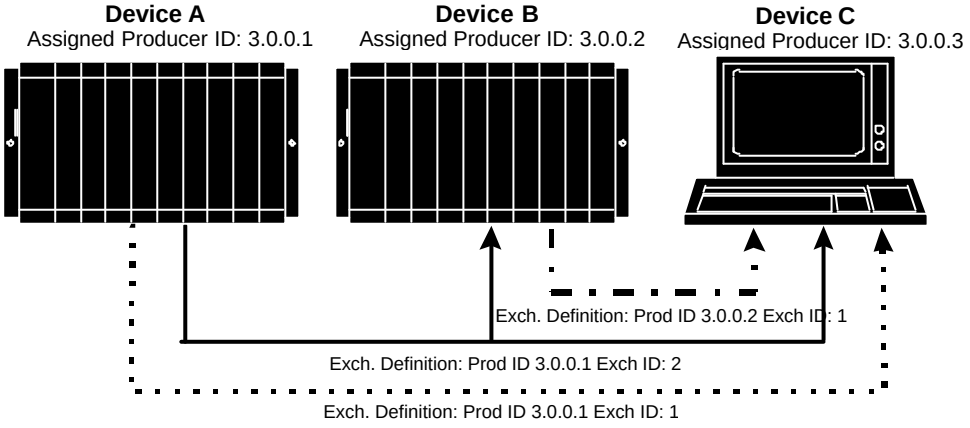
- 네트워크 상에 있는 데이터 생산의 샘플은 명시된 교환주기에서 발생하며 CPU 에서 업데이트되는 데이터 샘플은 시간과 무관합니다. CPU 는 교환 주기 간격에서 데이터를 업데이트하려 할 것이지만 PLC sweep time 에 의해 지연되기 때문에 명시된 시간에서 전달을 보증할 수 없습니다. 예를 들어 만약 교환주기가 만료될 때 CPU 가 로직을 실행한다면 업데이트는 로직 실행이 완료될 때까지 지연될 것입니다.

그러므로 네트워크 상에서 생산된 데이터 샘플은 이전에 전송된 샘플과 중복될 수 있습니다. 데이터가 새로운 것인지를 확인하기 위해 간단한 Network Time Protocol (SNTP) 타임스탬프 특성을 사용하십시오.

- CPU 에서 네트워크 어댑터까지 데이터 샘플의 첫 업데이트는 첫 스캔의 종료시 발생할 것입니다. 네트워크 상에 있는 첫 데이터 생산은 CPU 가 I/O 가능 상태로 전이후 한 Production Period 를 발생시킬 것입니다.
- 소비자에서 CPU 는 데이터가 네트워크로부터 업데이트될 때와는 별개로 Scan Period 에서 데이터를 스캔할 것입니다(네트워크 어댑터는 네트워크로부터 데이터 샘플이 도착할 때까지 업데이트 될 것입니다.). 데이터는 네트워크로부터 수신되지 않았다면 CPU 메모리에서 업데이트 되지 않을 것입니다.
- 교환 리스트에서 레퍼런스 메모리의 범위는 길이가 256 bytes 이하로 된다면 지속적으로 스캔하여 볼 수 있습니다. 만약 인터럽트가 가능하다면 일련의 다양한 범위는 동일한 스캔을 하는 동안 모든 것을 소비자측에 맡길 수는 없을 것입니다.

## Ethernet Global Data: 사례 1

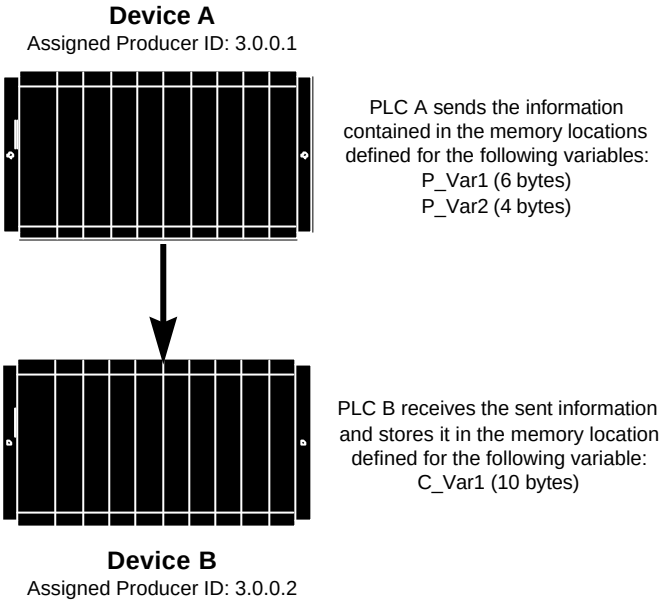
Ethernet Global Data 가 어떻게 작업할 것인가를 보여주는 사례입니다. 다음 그림은 Ethernet Global Data 교환을 하고 있는 세 디바이스(두 PLC 와 CIMPLICITY HMI 을 운용하는 PC) 을 보여주고 있습니다.




이 사례에서 Device A 는 두 교환을 생산하고 소비하는 것은 Device B 는 하나의 교환을 생산하고 하나의 교환을 소비하며 Device C 는 생산하는 교환은 없지만 세 교환을 소비합니다. Device A 에 의해 생산되는 교환과 Device B 와 Device C 에 의해 소비되는 교환은 여러 디바이스가 하나의 생산하는 디바이스에 의해 생산되는 데이터를 소비하는 그룹 사용법의 예입니다.

## Ethernet Global Data: 사례 2

이 사례는 생산하는 디바이스 A 에서 소비하는 디바이스 B 까지 간단한 데이터 교환의 세부사항을 보여주고 있습니다.




PLC A 는 다음 레퍼런스 메모리 위치에 포함된 정보를 전송합니다:

%I0001 to %I0005 (5 Bytes)

%Q0001 to %Q0005 (5 Bytes)

PLC B 는 다음과 같이 %G 메모리에서 10 연속 바이트에서 정보를 수신하고 저장합니다:

%G0001 to %G0010

## Producing PLC A 의 세부사항

PLC A 가 생산하는 교환의 세부사항은 Ethernet Global Data dialog box 의 Produced Exchanges tab 에서 정의됩니다:

- **Local Producer:** 기본값으로 PLC A 의 CPU 에 근접해 있는 네트워크 인터페이스의 IP 어드레스를 사용합니다. Local Producer 사용자에 의해 변경시킬 수 있습니다.
- **Exchange ID:** 전송될 데이터 패킷을 인식합니다.
- **Net Address Name:** 교환을 생산하게 될 PLC system 안에서 네트워크 어댑터로 할당된 이름.
- **Consumer Type:** 어떻게 소비하는 디바이스가 인식될 것인가를 명시합니다(이름, IP 어드레스 또는 group ID 에 의해). 이 사례에서 소비하는 PLC 는 Local Producer IP 어드레스에 의해 인식됩니다.
- **Consumer Address:** PLC B 에 할당된 Local Producer IP 어드레스.

- **Send Type:** 교환을 전송할 때 정의됩니다. 이 필드는 Always 로 고정되어 있습니다.
- **Producer Period:** PLC A 는 50 millisecond 마다 데이터를 생산할 것입니다.
- **Reply Rate:** 사용 불가.
- **Status Word:** 상태 워드가 있는 상태 메모리 위치에 대한 레퍼런스 어드레스는 저장됩니다. 상태 워드는 에러가 발생할 때 나타날 수 있습니다.

## 소비하는 PLC B 의 세부사항

PLC B 가 소비하는 교환의 세부사항은 Ethernet Global Data dialog box 의 Produced Exchanges tab 에서 정의됩니다:

- **Local Producer:** PLC B 의 CPU 에 근접해 있는 네트워크 인터페이스의 IP 어드레스를 사용합니다. Local Producer 사용자에 의해 변경시킬 수 있습니다
- **Exchange ID:** 수신될 데이터 패킷을 인식합니다.
- **Net Address Name:** 이 PLC 시스템에 있는 네트워크 어댑터에 할당된 이름은 교환을 소비하게 될 것입니다.
- **Producer ID:** PLC A 에 Local Producer IP Address 할당.
- **Group ID:** 하나 이상의 디바이스가 한 생산자로부터 동일한 교환을 소비할 때 사용합니다. 값 0 은 오직 한 디바이스만이 정보를 소비한다는 것을 의미합니다.
- **Consumed Period:** PLC B 의 CPU 는 25 millisecond 마다 새로운 데이터나 업데이트되는 상태를 위해 로컬 네트워크 어댑터를 스캔할 것입니다.
- **Update Timeout:** 만약데이터가 1000 millisecond 이내에서 수신되지 않는다면 네트워크 어댑터는 리프레쉬된 에러를 알리기 위해 상태 에러를 설정할 것입니다.
- **Status Word:** 상태 워드가 있는 메모리 로케이션에 대한 레퍼런스 어드레스는 저장됩니다. 이 상태 워드는 에러가 발생할 때, 나타날 수 있습니다.
- **Time Stamp:** 타임 스탬프가 있는 메모리 로케이션에 대한 레퍼런스 어드레스는 저장됩니다. 수신된 데이터가 이전에 수신된 샘플과 중복되는 것이 아니라 새로운 샘플이라면 타임 스탬프는 확인하기 위해 점검될 수 있습니다.

7

# *VersaMax Modular Rack 구성하기*

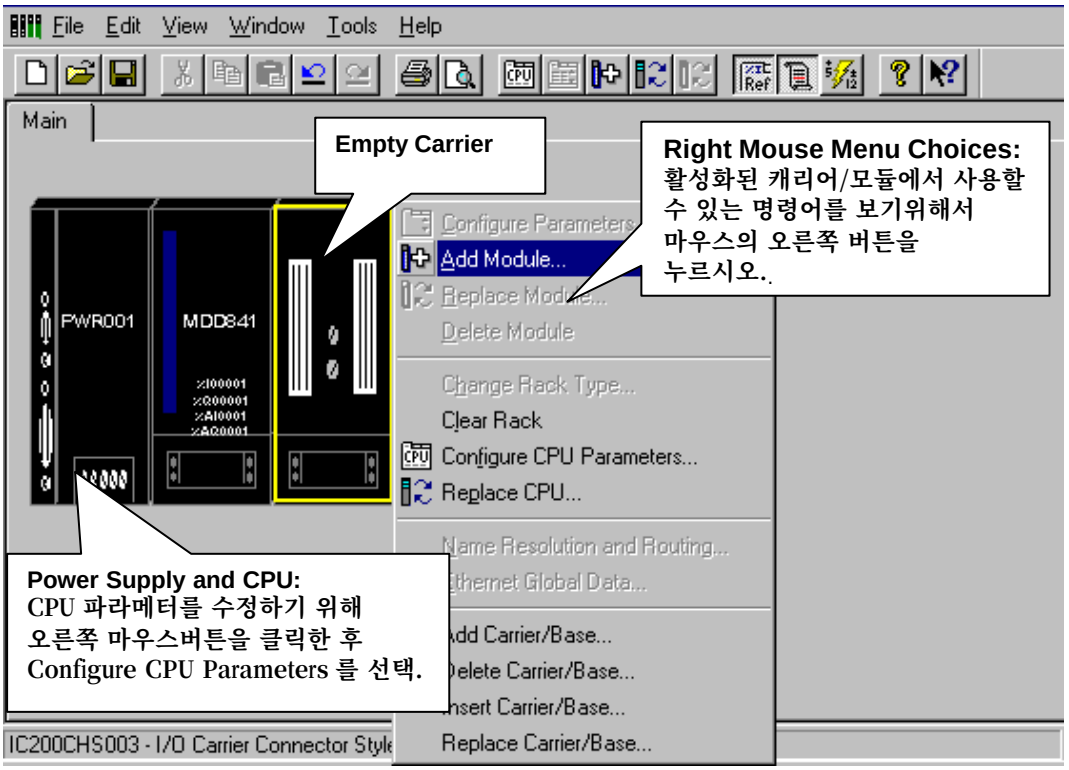
VersaMax 장비 폴더에 Hardware Configuration 을 입력할 때, 기본 보기는 Rack (Main)입니다. 파워 서플라이(PWR001)와 CPU(CPU001)는 기본 구성 안에 있습니다.

VersaMax 랙 시스템은 추가되는 I/O, 통신과 파워 서플라이 모듈에서 캐리어를 선택함으로써 구축됩니다.

기본 VersaMax 랙 시스템은 메인 랙을 가지고 있습니다. 다양한 파워 서플라이는 랙 시스템에서 추가해야 하는 모듈에 의해 요구하는 것처럼 메인 랙으로 지원됩니다.

다음 두 VersaMax 확장 네트워크는 사용될 수 있습니다: Local Single Rack과 Multiple Remote Rack. Multiple Remote Rack 구성은 7 개 확장 랙까지 가능합니다. Multiple Power Supply 는 확장 랙에서 사용될 수 있습니다.

Physical Hardware 에서, 필드 배선은 모듈 대신에 모듈 캐리어와 연결됩니다. HWC 는 Module Parameters 에서 모듈 배선을 디스플레이합니다. 만약 캐리어 간에 모듈을 이동시킨다면 그에 따른 배선 정보를 확실하게 업데이트시킵니다.



# Power Supply 와 CPU 변경하기/구성하기

## CPU Power Supply 구성하기

기본 파워 서플라이는 IC200PWR001 입니다. 기본 파워 서플라이를 변경시키기 위해서는 다음 단계를 수행하십시오:

1. 파워 서플라이 슬롯을 클릭하거나 두 번째 마우스 버튼을 클릭하여, Replace Module 을 선택합니다. Module Catalog 에서 새로운 파워 서플라이를 선택합니다.
2. 파워 서플라이에 대한 Parameter Editor window 가 나타날 것입니다.
3. 파워 서플라이 파라미터를 다시 보는 것을 마치려면 Parameter Editor window 를 닫기 위해 Window Close button  을 클릭하십시오.

기본 CPU 는 IC200CPU001 입니다.

- CPU 를 대체하려면 다음 단계를 수행하십시오:
  1. 두 번째 마우스 버튼을 클릭하여, CSM 에서 Replace Module 을 선택합니다.
  2. Module Catalog 에서 CPU 를 선택합니다. 선택된 CPU 에 대한 Parameter Editor window 가 나타날 것입니다.
  3. Parameter Editor 에서 CPU 파라미터를 다시 보고 수정합니다.
  4. CPU parameters 를 편집하는 것을 마치려면, Parameter Editor window 를 닫기 위해 Window Close button  을 클릭하십시오.
- CPU parameters 를 보거나 변경하기:
  1. 두 번째 마우스 버튼을 클릭하여 CSM 에서 Configure CPU Parameters 를 선택합니다. CPU 에 대한 Parameter Editor window 가 나타날 것입니다.
  2. Parameter Editor 에서 CPU 파라미터를 다시 보고 수정합니다.
  3. CPU parameters 를 편집하는 것을 마치려면, Parameter Editor window 를 닫기 위해 Window Close button  을 클릭하십시오.

# Booster Base 용 Power Supply 구성하기

파워 서플라이는 VersaMax system 을 통해 추가 시틸 수 있습니다. Booster Base 에서 구성된 파워 서플라이를 추가하거나 대체하려면 다음 단계를 수행하십시오:

1. 빈 booster base 를 더블 클릭하십시오. 구성된 파워 서플라이를 지닌 booster bases 에 있어서 오른쪽 마우스 버튼을 클릭하여 Replace Module 을 선택하십시오. Power Supply Module Catalog dialog box 가 나타날 것입니다.
2. 구성하고자 하는 파워 서플라이의 카탈로그 번호를 클릭한 후 OK button 을 클릭하십시오. Parameter Editor 가 디스플레이될 것입니다.
3. 파워 서플라이에 대한 정보를 볼 수 있습니다. 파워 서플라이 구성을 완성하기 위해 OK 를 클릭하십시오.

# Module Carrier 추가하기

캐리어는 VersaMax PLC 에서 모듈을 수용하기 위해 구성해야 합니다.

1. 구성 모듈 오른쪽에 캐리어를 추가하려면 오른쪽 마우스 버튼을 클릭하여 Add Base/Carrier 를 선택하십시오. Module Catalog for Carriers 가 디스플레이될 것입니다.

   두 구성 모듈 사이에 캐리어를 삽입하려면 새로운 캐리어 오른쪽에 있는 모듈을 밝게 하고 오른쪽 마우스 버튼을 클릭하여 Insert Base/Carrier 을 선택하십시오

**Note:** 오른쪽 마우스에서 구성 선택을 보기 위해 랙 구성에 있는 모듈이나 캐리어를 오른쪽 마우스를 클릭해야 합니다.

2. 추가하려는 캐리어의 카탈로그 번호와 설명을 클릭한 후 OK button 을 누르십시오.

# Power Supply Booster Bases 추가하기

Power Supply booster 는 부스터에서 파워 모듈을 다운스트림하도록 시스템 구성에 추가할 수 있습니다. Power Supply Booster Base 를 추가하기 위해 다음 단계를 수행하십시오:

1. 구성 모듈의 오른쪽에 Power Supply Booster Base 를 추가하려면 오른쪽 마우스 버튼을 클릭하고 Add Base/Carrier 을 선택하십시오. Module Catalog 로 디스플레이 될 것입니다.

   두 구성 모듈에 Power Supply Booster base 를 삽입하려면 새로운 베이스 오른쪽에 위치한 모듈을 밝게 하고 오른쪽 마우스 버튼을 클릭하여 Insert Base/Carrier 를 선택하십시오.

2. Power Supply Booster Bases tab 을 클릭하십시오. 베이스를 선택하기 위해 카탈로그 번호를 클릭하고 OK button 을 누르십시오.
3. 파워 서플라이를 구성하려면 오른쪽 마우스 버튼을 클릭하여 Add Module 을 선택합니다. 리스트에서 Power Supply 를 선택하고 OK 를 클릭하십시오.

# Module 추가하기/구성하기

캐리어가 구성되면 모듈은 시스템에 추가될 수 있습니다:

1. 모듈을 수용할 캐리어를 밝게 하십시오.
2. 빔 캐리어를 더블 클릭하거나 오른쪽 마우스 버튼을 클릭하여 Add Module 을 클릭하십시오(만약 이전에 구성한 모듈을 대체하려면 오른쪽 마우스 버튼을 클릭하여 Replace Module 을 클릭하십시오.).
3. 필요한 I/O 모듈을 설명하고 있는 tab 을 클릭하십시오(Discrete Input, Discrete Output, Analog Input, Analog Output, …..).
4. 시스템에 모듈의 카탈로그 번호와 설명을 클릭한 후 OK button 을 누르십시오.

# Module Parameter 구성하기

모듈의 파라미터를 구성하려면 다음 단계를 수행하십시오:

1. 모듈을 추가하거나 대체할 때, Parameter Editor 가 나타납니다. 또한 Rack window 에서 모듈 슬롯을 더블 클릭하여 이 창에 액세스할 수 있습니다.
2. Parameter Editor 에서 보거나 수정하려는 파라미터에 적합한 tab 을 클릭하십시오(어디서든 적용가능한).
3. 편집하고자 하는 Values fields 를 클릭하십시오(페이지 7-4 에서 “Editing Tips” 참조)
4. 모듈 파리미터 구성하기를 마치려면 Window Close button  을 클릭하십시오.

# Rack Modules 삭제하기

모듈을 제거하는 것은 랙에서 모듈을 제거하는 것입니다. 만약 모듈을 삭제한 후 Hardware Configuration 을 저장하면 삭제된 모듈에 대한 구성 정보는 잃어버릴 것입니다.

모듈을 삭제하려면 다음 단계를 수행하십시오:

1. 삭제하고자 하는 모듈을 선택합니다.
2. Delete key 를 누르거나 오른쪽 마우스를 클릭하고 Delete Module 을 선택하십시오.

# Carriers 삭제하기

캐리어를 삭제하기 위해 메뉴 바나 오른쪽 마우스를 사용해야 합니다.

캐리어를 삭제하려면 다음 단계를 수행하십시오.:

1. 삭제하고자 하는 빈 캐리어를 밝게 하십시오.
2. 오른쪽 마우스를 클릭하고 Delete Carrier 를 선택하십시오.

# Rack 지우기

랙을 지우는 것은 모든 모듈을 제거하는 것입니다(파워 서플라이와 CPU 는 제외). 만약 랙을 지운 후 Hardware Configuration 를 저장하면 삭제된 모듈에 대한 구성 정보는 잃어 버릴 것입니다.

랙 구성을 지우기 위해 다음 단계를 수행하십시오:

1. 랙에서 아무 곳이나 클릭하십시오.
2. 오른쪽 마우스 버튼을 클릭하고 Clear Rack 을 선택하십시오.
3. Confirmation dialog box 가 나타날 것입니다. 랙 지우기 동작을 확인하려면 Yes button 을 누르십시오.

# VersaMax Expansion Network 구성하기

VersaMax 랙 시스템에는 세 가지 타입의 구성 방법이 있습니다. 랙 구성을 변경하려면 Context-Sensitive Menu 나 Edit, Expansion Rack System 서브메뉴에서 원하는 타입을 선택하십시오.

**None**

단지 Main VersaMax 랙만을 포함하고 있습니다(확장형 없음). VersaPro 1.0 을 이용해 생성된 시스템은 이러한 구성 타입을 가지고 있습니다. 이것이 기본 VersaMax System 이며, 파일 메뉴에서 New 를 선택하거나 Series 90-30 과 같은 또다른 랙 시스템으로 변경할 때 생성됩니다.

**Local Single Rack**

Main VersaMax I/O station 과 하나의 Expansion rack 을 포함하고 있습니다. Main VersaMax rack 은 트랜스미터를 포함하고 있지 않습니다. Expansion rack 은 비절연 파워 서플라이와 Receiver Integrated Unit(IC200ERM002)을 포함합니다. 파워 서플라이는 Receiver 가 활동하지 않으면 구성되지 않습니다. 확장 랙에 있는 기본 파워 서플라이는 메인 랙의 파워 서플라이와 동일할 것입니다. 메인 랙의 주요 파워 서플라이를 대체하더라도 확장 랙의 주요 파워 서플라이에는 영향을 주지 않습니다.

**Multiple Remote Rack**

7 개 확장 랙까지 구성할 수 있습니다(7 개 Rack tab 은 항상 볼 수 있습니다.). Main VersaMax 랙은 트랜스미터(IC200ETM001)를 포함하고 있습니다. 기본 값으로 구성된 Non Isolated Receiver (IC200ERM002)은 Isolated Receiver (IC200ERM001)로 대체시킬 수 있습니다. 리시버 타입을 변경하려면 Non Isolated Receiver Unit 은 활성화되어 있어야 합니다.

**Notes:** Transmitter 와 Receiver 모듈은 모듈 카탈로그에서 추가시킬 수 없습니다. 이들 모듈을 랙 구성에 추가할 수 있는 한 방법은 적절한 구성 타입을 선택하는 것입니다. 또한 랙 구성 타입을 변경시키는 것 외에 랙 구성에서 모듈을 삭제할 수 없습니다.

Single-ended Receiver 가 구성되고 확장 트랜스미터가 나타나면, 치명적인 "expansion transmitter mismatch" 오류가 발생할 것입니다. 또한 differential(transmitter and receivers)이 구성되고 확장 트랜스미터가 물리적으로 나타나지 않는다면 이 오류는 발생할 것입니다. 이러한 경우에 오류는 확장 랙에 있는 개별 모듈에 대해서는 발생하지 않습니다. CPU 는 확장 버스를 통해 연결된 리시버 모듈의 존재 유무를 검출하고 물리적 구성을 프로그램된 구성과 비교합니다. 없는 리시버에 대해서는 "Loss of rack" 알람을 발생시키고 엑스트라 리시버에 대해서는 "Addition of or extra rack" 알람을 발생시킵니다.

## VersaMax Rack System 연결하기

한의 VersaMax 랙 시스템과 다른 VersaMax 랙 시스템을 연결할 수 있습니다. 이는 VersaPro 1.10 이전 버전에서 생성된 싱글 랙 시스템들을 Local Single Rack 과 Multiple Remote Rack 시스템으로 업그레이드 시킬 수 있습니다. 랙 시스템들을 연결시킴으로써 다음과 같은 효과를 가져올 수 있습니다:

구성들을 연결할 때 메시지는 기존 CPU/모듈의 파라미터 변경을 실행하는 옵션을 새로운 랙 구성으로 제공하는 것을 디스플레이합니다.

랙 시스템 타입을 변경하려면 CSM 나 편집 메뉴에서 Expansion Rack System 을 선택하십시오. 서브 메뉴에서 Expansion Rack 타입을 선택하십시오.

| From | To | Results |
|---|---|---|
| None | Local Single Rack | 새로운 확장 랙이 나타납니다. |
| None | Multiple Remote Rack | 새로운 7 개 확장 랙은 랙 시스템에 나타납니다. |
| Multiple Remote Rack 또는 Local Single Rack | None | 랙과 모듈의 손실과 작동을 실행취소시킬 수 없다는 경고 메시지가 디스플레이됩니다. 만약 OK 를 클릭하면 표준 랙이 디스플레이 될 것입니다. |
| Multiple Remote Rack | Local Single Rack | 랙과 모듈의 손실과 작동을 실행취소시킬 수 없다는 경고 메시지가 디스플레이됩니다. 만약 OK 를 클릭하면 Main and Rack 1 tab 을 포함하고 있는 랙 시스템이 디스플레이될 것입니다. 만약 첫 확장 랙에 절연 리시버가 있다면 기본 비절연 리시버로 대체됩니다. |

## Multiple Remote Rack Systems 용 파라미터

CPU 파라미터에 대한 Settings tab 상에서, 확장 랙 중 한 랙에 Multiple Remote Rack 이 선택되거나 절연 리시버가 나타난다면 Expansion Bus Speed 파라미터가 나타날 것입니다. 값은 확장 거리(기본 값)이고 정상입니다. Normal distance 에 있어서, 최대 허용 케이블 길이는 15m(250ft)이고 전송률은 250KHz, 1MHz, 3MHz 또는 5MHz 입니다. Extended Distance 에서, 최대 허용 케이블 길이는 250m(820ft )이고 전송률은 250KHz 입니다.

각각의 확장 랙은 하나의 파워 서플라이와 Non Isolated Receiver Unit 을 가지고 있습니다. 절연 리시버는 비절연 리시버와 Vice Versa 로 대체될 것입니다(“Expansion Rack Operations” 참조).

**Note:** 모듈이 자동 구성되고 Expansion Bus Speed 설정값이 변경되면 “Expansion bus speed change” 오류가 발생할 것입니다. Expansion Bus Speed 설정값을 정확하게 해야 합니다.

## Expansion Rack 운용

Power Supply And Receiver-Integrated 유닛에 대한 카탈로그 ID 와 설명은 Status bar 상에서 볼 수 있습니다. 통합 유닛을 선택할 때, 리시버 정보는 Status bar 의 한 부분으로 보여집니다.

### Expansion Receiver 선택하기

기본값으로 Expansion Receiver 타입은 없습니다. 파워 서플라이는 Expansion Receiver 가 구성될 때까지 Expansion Rack 에 구성되지 않습니다. Expansion Receiver 를 구성하는데 다음 세 가지 방법이 있습니다:

- 확장 랙에 베이스 유닛 추가. Non-Isolated Receiver Unit 은 기본값으로 구성될 것입니다.
- 랙 상에서 오른쪽 마우스를 클릭하고 Expansion Receiver 서브메뉴에서 Receiver Unit 선택.
- 편집 메뉴에서 Receiver Unit 선택.

유닛을 비활성화하기 위해 Expansion Receiver 서브 메뉴에서 None 을 선택하십시오(이 옵션은 랙 구성이 하나 또는 하나 이상의 Carrier/Base 유닛을 포함하고 있다면 사용할 수 없습니다.).

### Multiple Rack Systems 에서 Expansion Racks 용 Receiver Type 변경하기

Power Supply 와 Receiver-Integrated 유닛은 기본값으로 IC200PWR001 와 IC200ERM002(non-isolated receiver)으로 구성되어 있습니다. Multiple Remote Rack 시스템에 있는 확장 랙에 있어서 절연 리시버(IC200ERM001)는 선택될 수 있습니다. 리시버를 변경하려면 편집 메뉴나 CSM 에서 Expansion Receiver 를 선택하십시오.

## 전력 소모 값 보기

파워 서플라이의 파라미터를 보려면 Power Supply 와 Receiver-Integrated 유닛을 더블 클릭하십시오. 리시버에 대한 전원 값은 리시버의 파라미터 대화창에 나타날 것이고 값은 편집할 수 없습니다. 리시버 전원은 메인 파워 서플라이의 총 전력 소모의 한 부분으로 포함됩니다.

리시버에 대한 파라미터를 보려면 편집 메뉴에서 Rack Operations 메뉴나 CSM 에서 Configure Receiver Parameters 를 선택하십시오.

### Multiple Remote Rack Systems 용 랙 지우기

메인 랙에 초점을 맞춰 이 옵션을 선택할 때, 트랜스미터는 지워지지 않을 것입니다. 만약 확장 랙에 초점을 맞춰 이 옵션을 선택한다면 Power Supply 과 Receiver-Integrated 유닛을 제외한 모든 모듈들은 삭제됩니다.

### Receiver Parameter 구성하기

이 옵션은 선택된 Expansion rack 에 대한 Receiver Parameter 를 디스플레이합니다(리시버는 구성 파라미터를 가지고 있지 않습니다.).

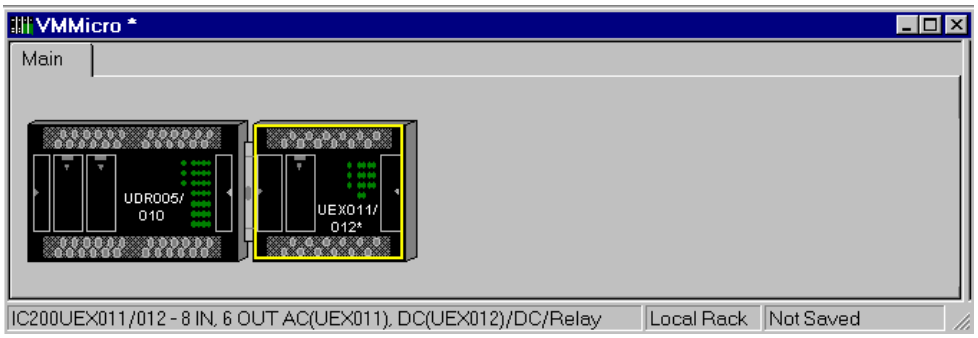
7

# *VersaMax Nano 와 Micro PLC 구성하기*

VersaMax Micro PLC 시스템은 메인 모듈과 4 개까지의 옵션 확장 랙으로 구성되어 있습니다(VersaMax Nano PLC 는 확장 랙을 지원하지 않습니다.).

메인 모듈은 CPU, I/O 와 HSC/PWM/PTO 기능을 포함하고 있습니다. Expansion Unit 은 오직 I/O 기능 만을 제공합니다. 콤팩트하고 가벼운 유닛은 35mm DIN rail 이나 판넬 취부형으로 설계되었습니다.

아래 나타난 샘플 시스템은 메인 모듈(요청시)과 하나의 확장 유닛을 가지고 있습니다.




## Rack/Module 운용

메인 모듈(IC200UDR005/010 이 기본값입니다.)을 대체하려면, CSM 에서 Replace Module 이나 Replace CPU 를 선택하거나 편집에 있는 Rack Operations 서브 메뉴에서 Replace CPU 를 선택하십시오.

시스템에 Expansion Unit 을 추가하려면 편집 메뉴에서 Module Operations 서브 메뉴나 CSM 에서 Add Module 을 선택하십시오. Expansion Unit 은 체인 끝에 추가됩니다.

Expansion Unit 을 대체하기 위해, Expansion Unit 을 선택하고 CSM 이나 편집 메뉴에 있는 Module Operations 서브 메뉴에서 Replace Module 을 선택합니다.

시스템에서 Expansion Unit 을 제거하려면 CSM 이나 편집 메뉴에 있는 Module Operations 서브 메뉴에서 Delete Module 을 선택하십시오 체인에서 마지막 확장 유닛만 삭제될 것입니다.

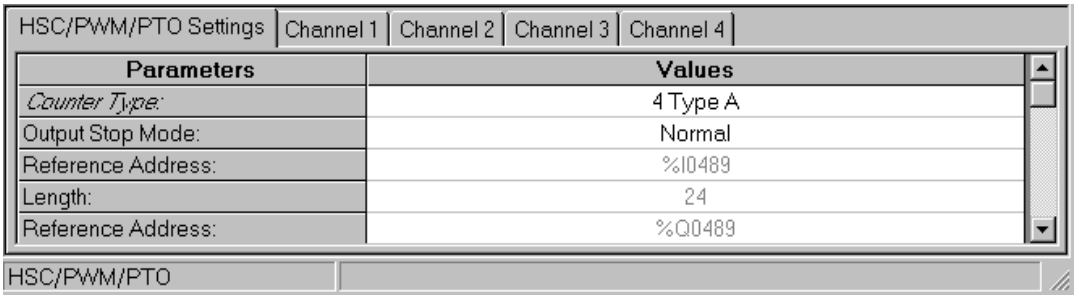
**Note:** 메인 모듈은 시스템에서 삭제시킬 수 없습니다.

시스템에서 모든 Expansion Unit 을 제거하려면 CSM 이나 편집 메뉴에 있는 Module Operations 서브 메뉴에서 Clear Rack 를 선택하십시오.

모듈을 추가하거나 대체할 때, Parameter Editor window 이 나타날 것입니다.

# Counter, Pulse Width Modulation 와 Pulse Train Output 구성하기

High Speed Counter, Pulse Width Modulation 와 Pulse Train Output 기능을 구성하려면 편집 메뉴로 가서 Module Operations, Configure Parameters, HSC/PWM/PTO 를 선택하십시오. 또한 CSM 에서 Configure Parameters, HSC/PWM/PTO 를 선택할 수 있습니다. 아래 그림과 같이 유사한 Parameter Editor window 가 나타날 것입니다(Nano PLC 는 오직 세 카운터 탭을 가지고 있습니다.). 이들 기능을 구성하는 세부사항은 Help menu 에서 Module Help 를 선택하십시오.

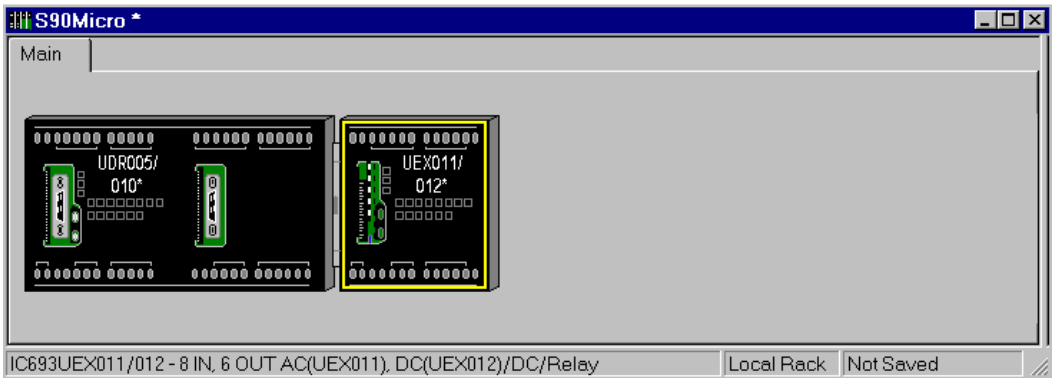


7

# *Series 90 Micro PLC 구성하기*

Series 90 Micro PLC 는 CPU, I/O 와 HSC/PWM/PTO 기능을 제공하는 Self-contained Module 입니다. 23 과 28 점 Micro PLC 는 I/O 기능만을 제공하는 확장 유닛을 네 대까지 지원합니다. 콤팩트하고 가벼운 이 유닛은 35mm DIN rail 이나 판넬 취부형으로 설계되어 있습니다.

아래 나타낸 샘플 시스템은 메인 모듈(요청시)과 하나의 확장 유닛을 가지고 있습니다.




## Rack/Module 운용

메인 모듈(IC693UDR005/010 은 기본값입니다.)을 대체하려면, CSM 에서 Replace Module 이나 Replace CPU 를 선택하거나 편집 메뉴에 있는 Rack Operations 서브 메뉴에서 Replace CPU 을 선택하십시오.

시스템에 Expansion Unit 을 추가하려면 CSM 이나 편집 메뉴에 있는 Rack Operations 서브 메뉴에서 Add Module 을 선택하십시오. Expansion Unit 은 체인 끝에 추가됩니다.

Expansion Unit 을 대체하기 위해, Expansion Unit 을 선택하고 CSM 이나 편집 메뉴에 있는 Module Operations 서브 메뉴에서 Replace Module 을 선택합니다.

시스템에서 Expansion Unit 을 제거하려면 CSM 이나 편집 메뉴에 있는 Module Operations 서브 메뉴에서 Delete Module 을 선택하십시오. 체인에서 마지막 확장 유닛만 삭제될 것입니다.

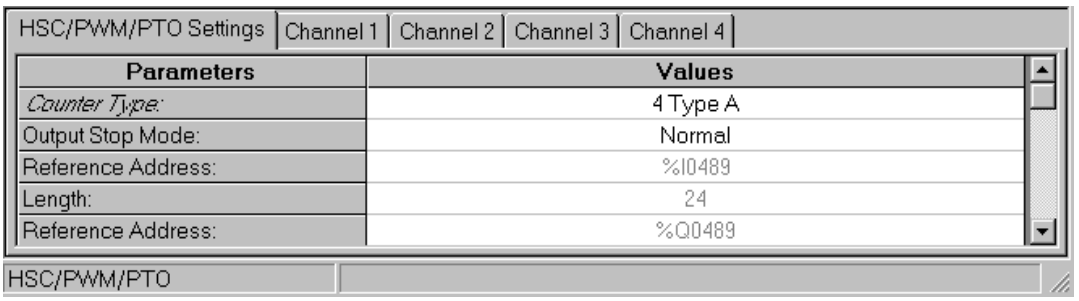
**Note:** 메인 모듈은 시스템에서 삭제시킬 수 없습니다.

시스템에서 모든 Expansion Unit 을 제거하려면 CSM 이나 편집 메뉴에 있는 Module Operations 서브 메뉴에서 Clear Rack 를 선택하십시오.

모듈을 추가하거나 대체할 때, Parameter Editor window 가 나타날 것입니다.

# Counter, Pulse Width Modulation 와 Pulse Train Output 구성하기

High Speed Counter, Pulse Width Modulation 와 Pulse Train Output 기능을 구성하려면 편집 메뉴로 가서 Module Operations, Configure Parameters, HSC/PWM/PTO 를 선택하십시오. 또한 CSM 에서 Configure Parameters, HSC/PWM/PTO 를 선택할 수 있습니다. 아래 그림과 같이 유사한 Parameter Editor window 가 나타날 것입니다(Nano PLC 는 오직 세 카운터 탭을 가지고 있습니다.). 이들 기능을 구성하는 세부사항은 Help menu 에서 Module Help 를 선택하십시오.

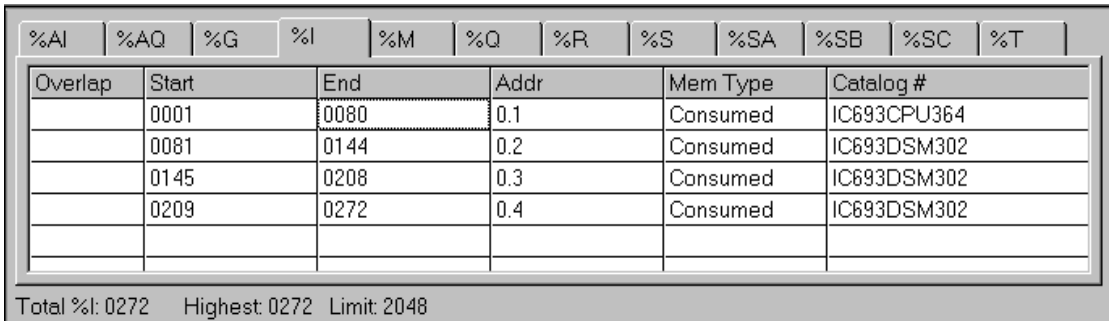



## *하드웨어 구성 레퍼런스 보기*

HWC Reference View는 모듈이 랙 시스템에 추가시킬 때 배치해야 할 레퍼런스에 대한 정보를 제공합니다. 이 기능은 I/O나 특수 모듈이 시스템에서 추가되거나 삭제될 때마다 업데이트 됩니다

Reference View는 삭제할 수 있는 보기 기능으로 마우스로 이를 선택할 수 있고 HWC 메인 창으로 이동시킬 수 있습니다. Reference View의 디스플레이를 조절할 수 있습니다. Reference View를 디스플레이 하거나 숨기려면 View 메뉴를 선택하여 Reference View를 클릭하십시오.

Reference View는 HWC(%I, %Q, %AI, %AQ, %G, %M, %R, …)에서 사용되는 각각의 레퍼런스 타입에 대해 개별 탭을 유지합니다. 특수 레퍼런스 타입에 대한 레퍼런스를 보려면 레퍼런스 타입에 일치하는 탭을 클릭하십시오.

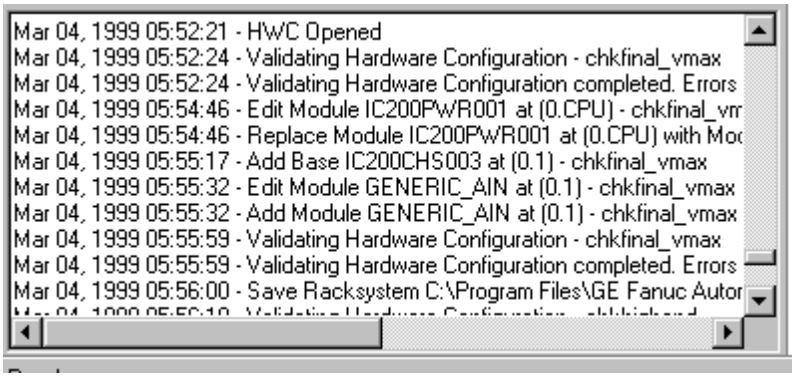



각각의 행은 특수 모듈에 대한 레퍼런스 포인트 할당을 제공합니다. 컬럼은 다음과 같이 구성됩니다:

- **Overlap:** 기본 값으로 Non-fatal Overlap 은 파란색 텍스트로 확인되며 Fatal overlap 은 붉은 색 텍스트로 인식됩니다.
- **Start:** 모듈에 대한 시작 레퍼런스 포인트 할당을 디스플레이합니다.
- **End:** 모듈에 대한 종료 레퍼런스 포인트 할당을 디스플레이합니다.
- **Addr:** Rack/Slot 에서 모듈의 Rack/Slot address 를 디스플레이합니다:
  - R 은 상대적인 랙 번호입니다(0 은 처음 또는 메인, 랙을 가리킵니다.).
  - S 는 할당된 랙에 있는 상대적인 랙 번호입니다(0 은 가장 왼쪽에 있는 슬롯을 가리킵니다)
- **Mem Type:** 할당된 메모리 영역에서 모듈이 데이터를 생산하고 소비하는지를 확인합니다.
- **Catalog #:** 모듈의 카탈로그 번호 표식.

# *Hardware Configuration Log View*

HWC 는 HWC 이벤트와 상태 정보의 이력을 기록하고 유지하는 로깅 기능을 제공합니다. 데이터는 PC 하드디스크에서 개별 로그 파일로 저장되고 다양한 편집 세션을 통해 액세스할 수 있습니다. 로그 파일의 기본 크기는 1MB 입니다. 만약 로그 데이터가 이 크기를 초과하면 가장 오래된 로그 정보는 새로운 로그 데이터로 겹쳐집니다.




HWC 는 로그 데이터를 기록하기 위해 두 가지 모드를 제공합니다:

- **Exception Only Mode**: Warning 과 예외 상황은 로그에 기록됩니다.
- **Audit Trail Mode (default mode)**: 정보 메시지(모듈 삭제, 추가 등과 같은)는 Warning 과 예외 상황과 함께 로그에 기록됩니다.

로그 파일의 설정치를 변경하려면 도구 메뉴를 선택하여 옵션을 선택하십시오. Audit Trail Mode 를 설정하고 기본 로그 파일을 조절할 수 있도록 Options dialog box 가 디스플레이될 것입니다.

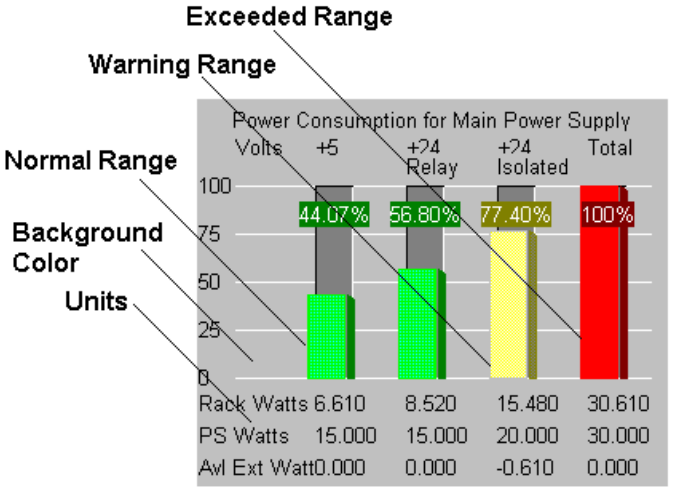
Log View 는 삭제할 수 있는 보기 기능으로 마우스로 이를 선택할 수 있고 HWC 메인 창으로 이동시킬 수 있습니다. Log View 의 디스플레이를 조절할 수 있습니다. Log View 를 디스플레이 하거나 숨기려면 View 메뉴를 선택하여 Log View 를 클릭하십시오.

# *Hardware Configuration Power Consumption View*

모든 VersaMax와 Series 90-30 Rack System Edit window는 이동할 수 있는 Splitter Bar에 의해 분할된 두 섹션으로 구성되어 있습니다. 왼쪽에 있는 섹션은 Rack System의 Tabbed View를 나타낸 것입니다. 오른쪽에 있는 섹션은 현재 선택된 모듈에 전원을 공급하는 파워 서플라이에 대한 현재 전력 소모 통계를 디스플레이하는 Power Consumption View를 나타낸 것입니다. 이들 통계는 개별 랙에서 구성된 I/O 모듈의 전원 요구사항을 모니터하는 것을 지원하도록 제공됩니다.

다음은 전력 소모 통계의 요약을 디스플레이 한 것입니다:

- 바 그래프는 각각의 전압에 사용된 전원과 총 시스템 사용량을 %로 나타낸 것입니다.
- Rack Watt 나 Amp 는 모듈이 사용한 총 전원을 디스플레이한 것입니다.
- PS Watt 나 Amp 는 파워 서플라이에 의해 공급된 총 전원을 디스플레이한 것입니다. 사용 가능한 총 시스템 전원이 개별 전압의 합계는 아니라는 것을 주의하십시오. 그러므로 개별 전압 용량을 초과하지 않고 시스템 용량을 초과하는 것이 가능합니다.
- AVL Ext Watt 와 Amp 는 각각의 전압에서 사용할 수 있는 외부 전원과 사용할 수 있는 총 외부 전원을 디스플레이합니다(Series 90-30 systems).



# 전력 소모 콤포넌트 보기

## 통계를 가진 파워 서플라이 디스플레이하기

이것은 선택된 모듈에서 가장 왼쪽에 있는 파워 서플라이입니다. 90-30 System 에서 랙마다 단일 Power Supply 이기 때문에 Power Supply 는 항상 슬롯 0 에 있을 것입니다. 다양한 파워 서플라이를 지원하는 VersaMax System 은 0 번 슬롯은 물론 어떤 슬롯도 설치할 수 있습니다.

### Normal Range

75% 이하로 나타나는 소비 전원 % 바는 녹색으로 디스플레이됩니다.

### Warning Range

75%~100%로 나타나는 소비 전원 % 바는 노란색으로 디스플레이됩니다.

### Exceeded Range

100%이상으로 나타나는 전력 소모 % 바는 빨간색으로 디스플레이됩니다. 소비가 초과했다면 그래프 바닥에 있는 라벨은 실제로 사용된 총계를 디스플레이하지만 %바는 100%보다 높게 나타나지는 않습니다.

## 컬러 구조

삼색 구조는 Power Consumption View를 디스플레이하는데 사용할 수 있습니다. 위의 예는 어두운 배경을 보여줍니다. 컬러 구조는 보기 메뉴의 Consumption 서브 메뉴에서 변경시킬 수 있습니다.

### Units

The Power Consumption View는 Watt 나 Amp 로 디스플레이시킬 수 있습니다. 디스플레이된 유닛을 변경하려면 View menu 로 가서 Power Consumption 을 선택한 다음 Power Display Units 서브메뉴를 선택하십시오.

**Note:** The 90-30 랙 시스템은 Amps Unit 을 선택하지 않으면 *Total* graph bar 로 디스플레이되지 않을 것이고 VersaMax 랙 시스템은 Amps Unit 을 선택하지 않으면 *Total* graph bar 로 디스플레이되지 않을 것입니다.

## 전력 소모 한계 계산

Rack System Edit Window 의 오른쪽에 있는 Power Consumption View 과 Power Consumption Dialog(편집 메뉴의 Power Consumption sub-menu 에 위치)는 파워 서플라이에 의해 산출된 다양한 전압에 대한 총 소비를 나타내는 그래프를 디스플레이합니다. 뿐만 아니라 그래프는 시스템 *총* 그래프 바를 나타낼 수 있고 널리 사용되고 있는 총 파워 서플라이 용량의 %를 나타냅니다. 전형적으로 시스템 총 합계가 파워 서플라이에 의해 산출되는 모든 전압의 합계는 아니라는 것을 주의하십시오.

90-30 파워 서플라이는 *제한된 전원*이므로 *총* 그래프 바는 디스플레이 단위가 Watt 일 때 계산할 수 있습니다. 반대로 VersaMax 파워 서플라이는 *제한된 전원*이므로 *총* 그래프 바는 디스플레이 단위가 Amp 일 때 계산할 수 있습니다.

## *Rack System 변환하기*

기본 하드웨어 구성은 VersaMax 입니다. 만약 이 구성이 작업하고자 하는 타입이 아니라면 타입은 다음 단계를 이용해 변경시킬 수 있습니다:

1. HWC 에서 File Menu 를 선택한 다음 Convert To 를 선택하십시오.
2. 리스트에서 사용할 수 있는 PLC 타입을 선택하십시오.
3. I/O 모듈들이 새로운 랙 시스템으로 변환되지 않을 것이라는 Warning 의 메시지 박스가 나타날 것입니다. 이 동작은 실행취소를 시킬 수 없다는 것에 주의하십시오.
4. 전환을 계속하려면 Yes 를 클릭하십시오. 변환하지 않고 기존 하드웨어 구성을 유지하려면 No 를 클릭하십시오.

**Note:** 기본 하드웨어 구성을 변경하려면 VersaPro window 에 있는 Tools 메뉴에서 Options 을 선택하십시오. General tab 에서, Default Hardware Configuration 을 사용할 수 있는 타입으로 변경시키십시오.

## *Hardware Configuration 인쇄하기*

파일 메뉴로 간 다음 Print 를 선택함으로써 Rack 구성을 인쇄할 수 있습니다. 인쇄 옵션을 설정하려면, Print dialog box 에 있는 Range button 을 클릭하십시오 다음과 같은 인쇄옵션을 사용할 수 있습니다:

- 모든 랙이나 (rack:slot 에 선택된)모듈 범위에 대한 하드웨어 구성 정보를 인쇄하기 위해 선택하십시오.
- 구성에 대한 개요와 세부사항을 인쇄하기 위해 선택할 수 있습니다:

- **Overview:** 스크린에 나타난 인쇄하기
- **Detail:** 랙에 있는 개별 모듈에 대한 파라미터를 인쇄하기
- **Ethernet Global Data:** 만약 Series 90-30 CPU364 가 Ethernet Global Data exchange 에 관여하기 위해 구성된다면 CPU 에 대한 EGD 구성을 인쇄하기 위해 선택할 수 있습니다.
- **Reference Details:** 메모리 시작과 종료, 랙 슬롯 어드레스 및 기타 레퍼런스 데이터를 인쇄합니다.
- **Name Resolution:** Name Resolution 을 이용한 Series 90-30 CPU364 구성에 있어서, Name Resolution 구성 정보를 인쇄하기 위해 이 박스를 체크합니다.

만약 프린터 종류, 용지 크기나 방향을 변경하려면 Print dialog box 에서 Setup button 을 클릭하십시오.

Chapter 8

# Reference 와 Variable View Table 사용하기

VersaPro 는 PLC 에 있는 요소의 실시간 값과 상태를 모니터링하기 위한 두 View Table 을 제공합니다:

- Variable View Table 은 온라인으로 볼 수 있도록 윈도우에서 중요한 변수들을 그룹화해 줍니다.
- Reference View Table 은 온라인으로 볼 수 있도록 윈도우에서 중요한 레퍼런스들을 그룹화해 줍니다.

Multiple View Tables 은 곧바로 열 수 있습니다.

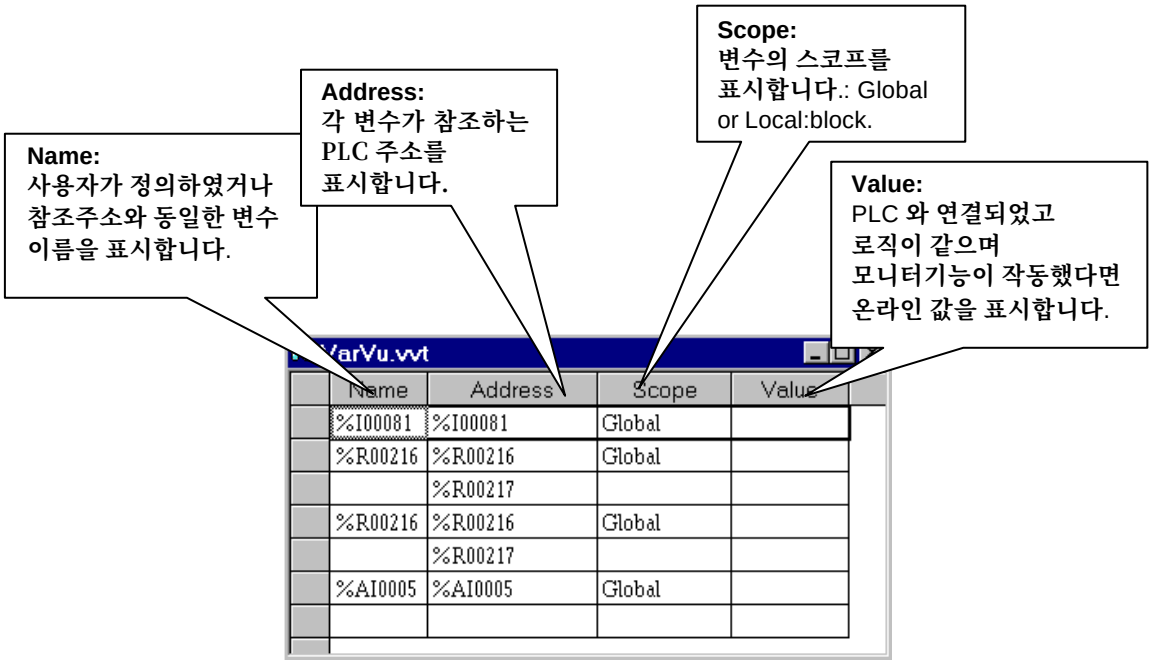
## 목차

이 장은 다음 주제에 대한 정보를 제공합니다:

# *Variable View Tables –개요*

Variable View Table(VVT)은 변수 상태를 모니터할 수 있게 해줍니다. 폴더는 0 또는 그 이상의 혼합형 VVT 을 가지고 있습니다. VVT 와 VVT 등록은 폴더간에 자르기와 복사 그리고 drag and drop 할 수 있습니다.

VVT 의 예를 아래에 나타냈습니다.




Name 컬럼만 편집할 수 있습니다. 다른 모든 컬럼은 읽기 전용입니다.

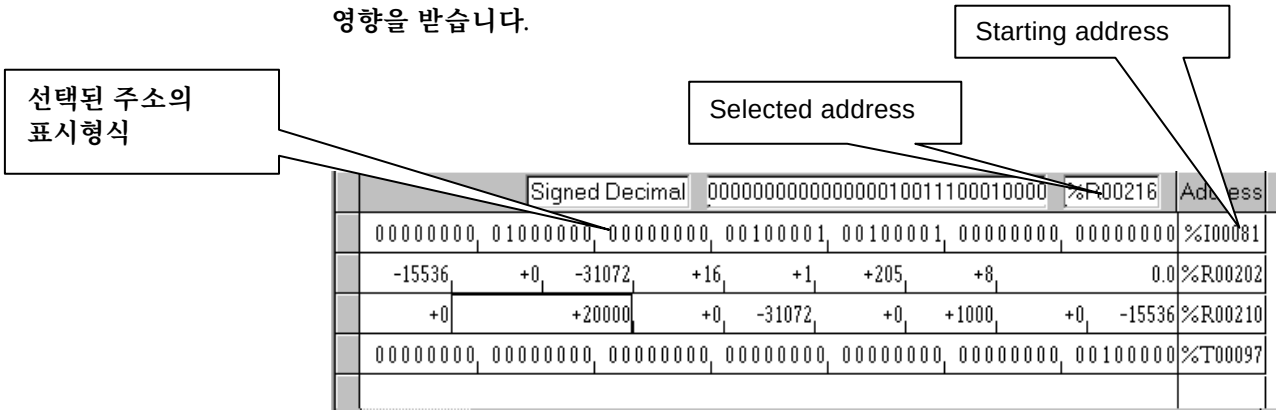
Value 컬럼은 폴더가 온라인일 때까지 공백입니다. Value Cell 은 레퍼런스 어드레스를 할당되지 않은 변수에 대해서는 "****"로 디스플레이합니다.

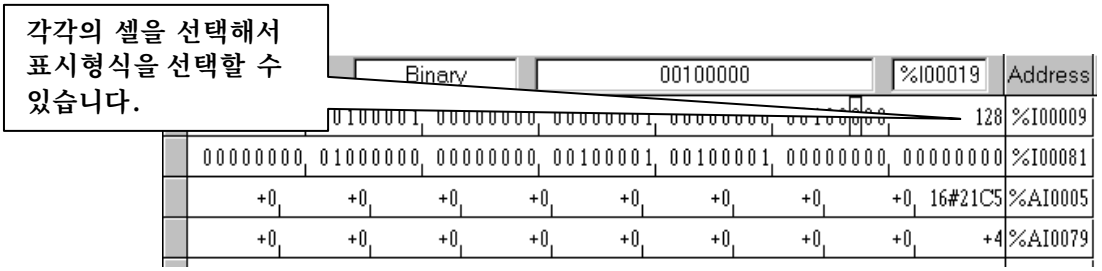
만약 셀이 온라인 값을 디스플레이하기에 너무 좁다면 이 셀은 ####로 디스플레이 될 것입니다.

# *Reference View Table – 개요*

Reference View Table(RVT)는 실시간으로 모니터하고 업데이트시킬 수 있는 레퍼런스의 리스트를 포함하고 있습니다. 폴더는 0 또는 그 이상의 레퍼런스 테이블을 가지고 있습니다. RVT 는 폴더 간에 자르기, 복사와 drag and drop 할 수 있습니다. 하나 이상의 Reference Area 는 단일 RVT 에 입력시킬 수 있습니다.

RVT 에 포함된 등록 수는 RVT 의 수행에 영향을 주지 않습니다. 수행은 View 에서 디스플레이되고 업데이트시켜야 하는 등록 수에 의해 영향을 받습니다.



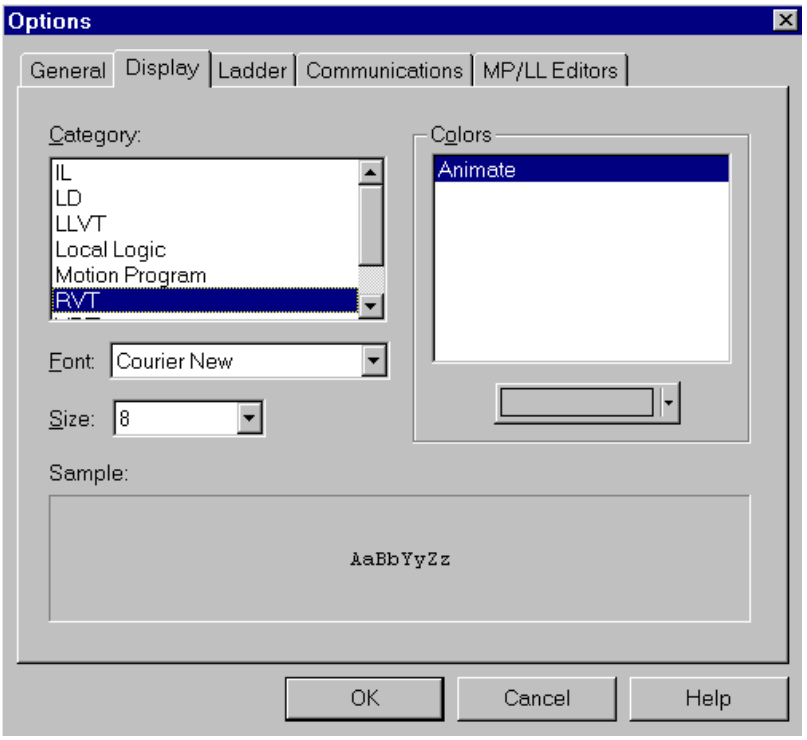




모든 레퍼런스 어드레스는 여러분이 입력한 주소에서 시작해 가장 적은 오프셋을 가지고 설정되며, 8 컬럼 뒤에 끝납니다(데이터의 10 컬럼은 %R, %AI, %AQ reference address 에 대해 디스플레이 됩니다.). 컬럼에서 디스플레이되는 데이터의 양은 포맷에 달려 있습니다.

## View Table Display 만들기

VVT 와 RVT 에 대한 폰트 선택과 색상을 최적으로 만들 수 있습니다. 디스플레이를 최적으로 만들려면 다음 단계를 수행하십시오:

1. Folder Browser 에서, Tools 메뉴를 클릭하고 Options 을 선택하십시오. Options dialog box 나 나타날 것입니다. Display tab 을 선택하십시오.



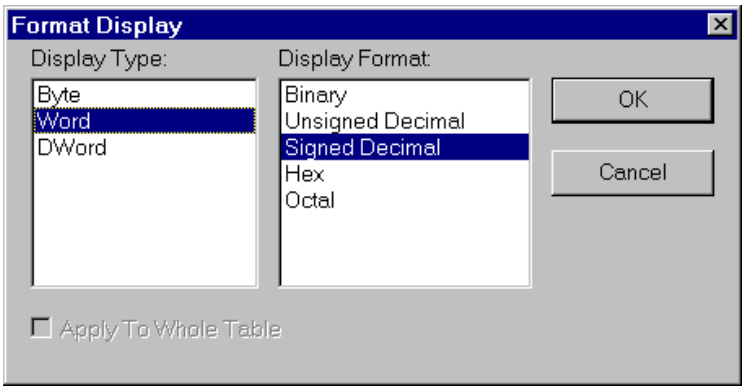

2. 온라인 값의 디스플레이에 대해 폰트 선택을 변경하고 사용된 색상을 변경할 수 있습니다.

   - RVT 에 대한 설정치를 수정하려면, RVT 에 Category 를 설정하고 원하는 대로 변경을 하십시오. 수용하기 위해 OK 를 누르십시오.
   - VVT 에 대한 설정치를 수정하려면, VVT 에 Category 를 설정하고 원하는 대로 변경을 하십시오. 수용하기 위해 OK 를 누르십시오.

## Reference View Table Display Format 변경하기

Byte, Word 또는 Dword 의 선택은 각 셀에서 디스플레이되는 데이터가 얼마나 되는지 관리합니다. RVT 을 포맷하는 것은 테이블에서 발생합니다. RVT 에 대한 디스플레이 포맷을 변경하려면 다음 단계를 수행하십시오:

1. RVT 를 여십시오.
2. View 메뉴를 선택하고 Format View Table 을 선택하십시오. Format Display dialog box 가 나타날 것입니다.




3. 애플리케이션 요구에 적합한 Grouping 과 Display Format 을 선택하십시오:
   - **Grouping Field**
     - Byte(각각의 셀은 8 bits 를 포함하고 있음)
     - Word(각각의 셀은 16 bits 를 포함하고 있음)
     - Dword(각각의 셀은 32bits 를 포함하고 있음)
   - **Display Format**
     - Binary
     - Hex
     - Octal
     - Unsigned Decimal
     - Signed Decimal
     - Real(Dword 를 선택할 때만 사용)
     - Scientific Notation(Dword 를 선택할 때만 사용).

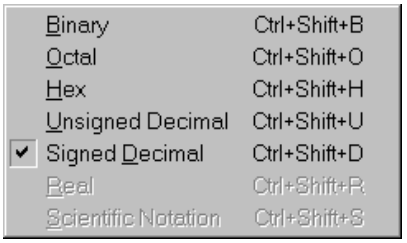
CSM 에서 셀을 선택하고 Display Format 을 선택함으로써 개별 셀의 디스플레이 포맷을 변경할 수 있습니다.

## Variable View Table Display 만들기

Variable View Tables 에 대해 Value Display Format 을 만들 수 있습니다. 온라인 값에 대한 디스플레이 포맷을 변경하려면 다음 단계를 수행하십시오:

1. VVT 를 여십시오. 디스플레이 포맷을 변경하고자 하는 변수를 포함하고 있는 행을 선택하십시오.

2. 오른쪽 마우스 버튼을 클릭하고 CSM 에서 Display Format 을 선택하십시오.




3. 애플리케이션 요구사항에 적합하도록 디스플레이를 변경시키십시오. 포맷은 선택된 변수로 변경될 것입니다.

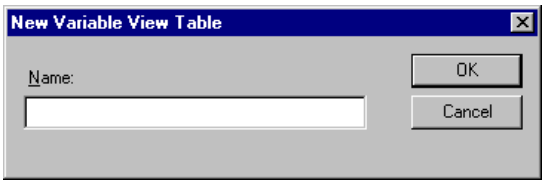
**Note:** 만약이 값이 ####로 디스플레이 된다면, 온라인 값을 보기 위해 값 컬럼의 폭을 증가시키십시오.

# 새 Reference 와 Variable View Tables 생성하기

새 Variable View Table 과 Reference View Table 은 Folder Browser 에서 빠르게 생성됩니다. 폴더에서 여러 View Table 을 가질 수 있습니다. 이 섹션은 VVT 와 RVT 을 생성하는 방법을 기술합니다.

## Variable View Table 생성하기:

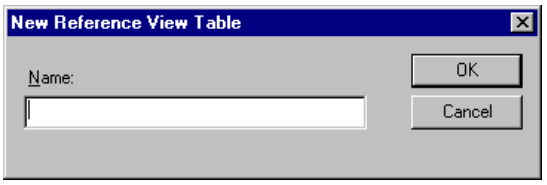
1. Folder Browser 에서 오른쪽 마우스 버튼을 클릭하고 New Variable View Table(또는 File 메뉴를 선택하여 New Variable View Table 를 선택하십시오.)을 선택하십시오. New Variable View Table dialog box 가 나타날 것입니다:




2. VVT 에 대한 이름을 입력하십시오. 수용하면 OK 를 클릭하십시오.
3. 새 VVT 는 폴더 창에서 생성되고 디스플레이 됩니다. 새 VVT 아이콘은 VVT 이름으로 Folder Browser 로 추가됩니다.

## 새 Reference View Table 생성하기:

1. Folder Browser 에서 오른쪽 마우스 버튼을 클릭하고 New Reference View Table(또는 File 메뉴를 선택하여 New Reference View Table 를 선택하십시오.)을 선택하십시오. New Reference View Table dialog box 가 나타날 것입니다:




4. RVT 에 대한 이름을 입력하십시오. 수용하면 OK 를 클릭하십시오.
2. 새 RVT 는 폴더 창에서 생성되고 디스플레이 됩니다. 새 RVT 아이콘은 RVT 이름으로 Folder Browser 로 추가됩니다.

**Note:** 옵셋 컬럼은 PLC 에 연결되고 모니터링이 가능할 때까지 거주하지는 않습니다.

# View Table 에 요소 추가하기

온라인으로 모니터링하기 위해 View Table 로직 요소(레퍼런스 또는 변수)를 추가할 수 있습니다.

여러분이 View Table 에 데이터를 추가하고자 할 때는 off-line 혹은 on-line 일 것이다. 만약 온라인 상태이고 PLC 와 동기가 되어있다면, 새로 추가된 변수의 데이터는 자동으로 업데이트됩니다.

## VVT 에 직접 등록하여 요소 추가하기:

1. Folder Browser 에서 VVT 를 여십시오.
2. VVT 에서 빈 라인에 Name field 를 선택하십시오.
3. Name field 의 drop-down list 를 이용해 변수를 선택하거나 정확한 변수 이름을 입력하십시오.
4. Enter 를 누르거나 다른 필드를 선택하십시오. 등록된 변수와 일치하는 컬럼(어드레스와 스코프)은 VVT 에 디스플레이 됩니다.

## 정확한 VVT 변수 등록

정확한 변수 등록은 다음과 같은 요소로 구성되어 있습니다:

- name
- address(어드레스만이 Name field 에 입력된다면, 그 어드레스는 Name and Address fields 에 나타날 것입니다.)
- name, address(스코프는 기본 적으로 전역(Global)입니다.)
- name, g(스코프는 Global 로 할당될 것입니다.)
- address, g
- name, address, g
- name, local: scope(스코프는 변수가 부분적으로 할당되도록 하는 블록입니다.예를 들면, name “A”를 가진 변수를 block named “B”에 부분적으로 할당하려면, Name field 에 “A,local:B”라고 입력하십시오.)
- address, local: scope
- name, address, local: scope

View Table 에서 변수의 길이는 Variable Declaration Table 로 입력되는 길이를 반영합니다. 길이 변경은 VDT 를 통해서만 가능합니다. 정보는 Address 나 Scope fields 에 직접 입력할 수 없습니다.

## Reference View Table 에 어드레스 추가하기:

1. Folder Browser 에서 RVT 를 여십시오.
2. 빈 라인에서 Reference Address 컬럼을 선택하십시오. 다음 포맷중 하나에 레퍼런스 어드레스를 입력하십시오 (%I00001, 1I).
3. Enter 를 누르거나 다른 필드를 선택하십시오. 등록된 변수와 일치하는 컬럼들은 RVT 에 디스플레이 됩니다.

## 정확한 RVT 등록:

- Word Memory 영역(R, AI, AQ)은 등록한 어드레스로 시작합니다.
- 동일한 레퍼런스 어드레스는 한 번 이상 등록되고 디스플레이 될 수 있습니다.
- 값이 너무 커서 현재 셀 폭에 디스플레이할 수 없다면, "#####" 표시는 모든 값이 디스플레이하기 위해 컬럼 폭을 넓힐 때까지 나타납니다.
- 오프셋의 유효범위는 0-99999 입니다. 만약 입력한 오프셋이 최대값을 초과한다면 최대값과 범위를 초과했다는 사실을 알리기 위해 에러 메시지가 나타납니다. RVT 에서 모니터링은 모든 오프셋이 PLC 의 한계 내에 있도록 결정할 때까지 할 수 없을 것입니다.
- 만약 오프셋이 유효하다면 데이터는 사용할 수 없고 0 이 디스플레이 됩니다. 만약 오프셋이 유효하지 않다면 셀은 빈공간을 남아 있습니다.

## Reference View Table 에 범위 추가하기:

1. Folder Browser 에서 RVT 를 더블 클릭하여 RVT 를 여십시오.
2. 빈 라인에서 Reference Address 컬럼을 선택하십시오.
3. 시작 주소를 넣고 콤마를 넣고 마지막 주소를 입력합니다. 여러 레퍼런스 어드레스들은 콤마에 의해 분리시켜야 합니다. 잘못된 범위의 참조 주소를 입력하면 에러가 발생할 것입니다.
   - 같은 레퍼런스 어드레스는 한 번 이상 입력되고 디스플레이 됩니다.
   - 오프셋의 유효범위는 0-99999 입니다. 만약 입력한 오프셋이 최대값을 초과한다면 최대값과 범위를 초과했다는 사실을 알리기 위해 에러 메시지가 나타납니다. RVT 에서 모니터링은 모든 오프셋이 PLC 의 한계 내에 있도록 결정할 때까지 할 수 없을 것입니다.
   - 만약 오프셋이 유효하다면 데이터는 사용할 수 없고 0 이 디스플레이 됩니다. 만약 오프셋이 유효하지 않다면 셀은 빈공간으로 남아 있습니다.

8

# *View Table 에서 정보 수정하기*

VersaPro 는 변수 이름을 변경함으로써 View Table 에 포함된 변수를 편집할 수 있게 합니다. View Table 에서 이름을 변경하려면 VDT 에서 다른 변수를 선택하고, View Table 에 있는 기존 변수를 대체시키십시오. 이름 변경은 VDT 에서만 가능합니다.

레퍼런스에 값을 쓰기 위해, VVT 나 RVT 에서 레퍼런스와 일치하는 셀을 선택하고 새 값을 입력하십시오(또한 PLC 메뉴에서 F10 을 누르거나 Write 를 선택하여 셀을 더블 클릭함으로써 View Table 에 넣을 수 있습니다.).

# *View Tables 에서 자르기, 복사, 붙여넣기 및 Drag and Drop 사용하기*

## VVT 에서 자르기, 복사 붙여넣기 기능 사용하기

1. 변수 입력이나 등록을 자르기 하고자 하는 VVT 를 여십시오.
2. 자르기 하려는 각각의 행에서 최소 한 행을 선택하십시오.
3. Edit 메뉴에서 Cut 이나 Copy 를 선택하고 Cut  이나 Copy button  을 클릭하거나 Ctrl + X(자르기)나 Ctrl + C(복사)를 누르십시오.
4. 모든 행은 VVT 에서 완전히 선택되어 제거되고 VersaPro 리스트 포맷(텍스트가 아닌)에 있는 클립보드에 놓이게 됩니다. 자르기 한 공간을 채우기 위해 다른 행을 이동시키십시오.
5. VVT 에 있는 기존 변수 사이에 변수를 삽입시키려면 기존 변수에 겹치기 하는 것이 아니라 삽입하고자 하는 지점 밑의 행에 있는 셀을 선택하십시오.
6. 기존 변수와 겹치려면 겹치려는 행을 선택하십시오.
7. Context-Sensitive 나 Edit 메뉴에서 Paste 를 선택하고 Paste button  을 클릭하거나 Ctrl + V 를 누르십시오. 변수를 삽입하면 변수는 선택한 셀에 있는 행 위의 새 행에 나타나고 다른 영향을 받는 테이블에 덧붙여집니다. 변수가 기존 변수와 겹쳐진다면 변수들은 기존 변수위치에 나타납니다.

### 자르기와 복사하기에 대한 세부사항

- VVT 에서 변수 자르기와 변수 복사는 동일하거나 다른 VVT 에서만 붙여넣기를 할 수 있습니다. 또한 다른 폴더에 있는 VVT 사이에 붙여넣기를 할 수 있습니다.
- VVT 에서 변수를 자르거나 복사하는 것은 VDT 에 영향을 주지 않습니다.
- 변수는 모니터링하는 동안에도 자르거나 복사할 수 있습니다.
- 텍스트는 편집 모드에 있을 때 Name 사이에 붙여넣기를 할 수 있습니다.

**Note:** 모든 행은 자르기를 할 수 있습니다. 행에 있는 셀을 선택한 후 자르기를 하면 모든 행은 자르기가 됩니다.

## 붙여넣기에 대한 세부사항

- 겹쳐쓰기 하려고 선택된 변수보다 클립보드에 있는 변수가 많다면 선택된 변수는 겹쳐 쓰게 되고 남아 있는 클립보드 변수는 테이블에 삽입됩니다.
- 겹쳐쓰기 하려고 선택된 변수보다 클립보드에 있는 변수가 적다면 가능한 한 선택한 변수만큼 겹쳐 쓰게 되고 나머지는 삭제됩니다.
- VVT 의 두 번째 인스턴스에 붙여넣기 할 때, VDT 에 새로운 변수가 생성됩니다.
- VDT 에 변수를 생성할 때 충돌이 발생하면, 문제를 설명한 에러 메시지가 나타납니다.
- OK 를 클릭하십시오. 문제를 해결할 수 있도록 대화창이 나타납니다.
- 문제를 해결하십시오. Row validation 은 다른 변수에 대해 계속 합니다. 만약 취소한다면 모든 붙여넣기는 취소됩니다.

## VVT 에서 drag and drop-cut 하기:

1. 변수 입력이나 등록을 자르기 하고자 하는 VVT 를 여십시오.
2. 자르기 하려는 각각의 행에서 최소 한 행을 선택하십시오.
3. 마우스를 선택한 행에 위치하십시오.
4. 첫 번째 마우스 버튼을 누르고 있으십시오.
5. 적당한 지역으로 변수나 셀을 이동시키십시오.
6. 드래그 해야 할 대상을 가리키기 위해 선택한 영역을 그래픽으로 표시하여 디스플레이합니다. 이 표시는 다른 셀 규격을 가지고 있는 VVT 의 두 인스턴스 사이에 극적으로 변경됩니다.
7. 대상을 드롭 하려면 마우스 버튼을 떼십시오. 만약 대상이 선택한 위치에 맞는다면 선택된 변수나 셀은 원위치에서 제거되고 새 위치에 놓이게 됩니다. 대상이 맞지 않는다면 drag and drop 동작은 취소됩니다.

## VVT 에서 drag and drop-copy 하기:

1. 변수 입력이나 등록을 자르기 하고자 하는 VVT 를 여십시오.
2. 자르기 하려는 각각의 행에서 최소 한 행을 선택하십시오.
3. 마우스를 선택한 행에 위치하십시오.
4. Ctrl key 를 누르고 있으십시오.
5. 첫 번째 마우스 버튼을 누르고 계십시오 .
6. 적당한 지역으로 변수나 셀을 이동시키십시오.
7. 드래그 해야 할 대상을 가리키기 위해 선택한 영역을 그래픽으로 표시하여 디스플레이합니다. 이 표시는 다른 셀 규격을 가지고 있는 VVT 의 두 인스턴스 사이에 극적으로 변경됩니다

8. 대상을 드롭 하려면 마우스 버튼을 떼십시오. 만약 대상이 선택한 위치에 맞는다면 선택된 변수나 셀은 원위치에서 제거되고 새 위치에 놓이게 됩니다. 대상이 맞지 않는다면 drag and drop 동작은 취소됩니다.

### VVT 에서 drag and drop 하기에 대한 세부사항

- 대상은 행의 아무 부분으로나 드롭 시킬 수 있습니다.
- 대상을 테이블 마지막 행 다음에 있는 VVT 의 마지막 행에 드롭시킨다면 이를 수용하기 위해 새로운 행을 생성시킵니다.
- 변수는 VVT 에 한 번 이상 입력시킬 수 있습니다.
- 대상을 VVT 의 마지막 행 다음에 드롭 시킨다면 이를 수용하기 위해 새로운 행을 생성시킵니다.
- 대상을 드롭시킬 때 대상은 기존 셀에 겹쳐집니다.
- VersaPro 는 비연속된 셀은 drag and drop 할 수 없습니다.
- VVT 의 두 번째 인스턴스에 드롭시킬 때, 새로운 변수는 VDT 에 생성됩니다.
- VDT 에 변수를 생성할 때 충돌이 발생하면, 문제를 설명한 에러 메시지가 나타납니다.
- OK 를 클릭하십시오. 문제를 해결할 수 있도록 대화창이 나타납니다.
- 문제를 해결하십시오. Row validation 은 다른 변수에 대해 계속 합니다. 만약 취소한다면 모든 붙여넣기는 취소됩니다.

## View Tables 저장하기

편집할 때 RVT 와 VVT 에 변경사항을 저장하는 것은 중요합니다. 이 섹션은 View Table 을 저장하는 방법을 기술합니다.

온라인 값은 View Table 에 저장되지 않습니다. 온라인 값은 PLC 에 유지되고 View Table 에서 디스플레이 됩니다.

View Table 을 편집할 때 변경사항을 저장하려면, 파일 메뉴를 선택하고 Save 를 선택하거나 Save toolbar button  을 클릭하십시오.

## Online 으로 보기

View Table 에서 온라인 값을 보려면, 폴더는 열려 있어야 하고 PLC 와 연결시켜야 합니다. 또한 View Table window 은 PLC 데이터를 모니터링해야 합니다. 모니터링하기 위해 View Table 을 설정하려면 Monitor All 이나 Monitor Active toolbar button 을 클릭하십시오.

이 장은 온라인으로 작업할 때, 윈도우 작동에 대한 세부사항을 제공하고 온라인으로 값을 변경하는 방법을 기술합니다.

## Online 으로 보기에 대한 세부사항

- 값이 너무 커서 현 셀 폭에 디스플레이할 수 없다면, "#####" 표시는 모든 값이 디스플레이하기 위해 컬럼 폭을 넓힐 때까지 나타납니다.
- 오버라이드 상태는 밑줄 친 값으로 디스플레이 됩니다.
- Overriding 과 Toggling 변수는 오른쪽 마우스를 통해 행해집니다.
- 배열 변수(1 보다 큰 길이를 지닌 변수)는 첫 입력 라인이 이어지는 곳에 주소, 스코프, 값을 보여줍니다.

## 레퍼런스 토글 또는 오버라이드하기:

RVT 에서 온라인으로 작업하려면, PLC 와 연결하고 레퍼런스를 Toggle 이나 Override 하기 위해 다음 단계를 수행하십시오.

1. RVT 를 여십시오. View 메뉴를 선택하고 Monitoring -> All 또는 Monitoring -> Active Window 을 선택하십시오.
2. Toggle 혹은 Override 하려는 레퍼런스의 오프셋에 커서를 놓으십시오.
3. PLC 메뉴를 선택하고 Toggle 을 선택하십시오 F12 key 를 누르거나 Toggle button  을 클릭하십시오. RVT Override/Toggle Bit dialog box 가 나타날 것입니다.
4. 오버라이드하기 위해 F11 key 를 누르거나 Override button 을 클릭하십시오.

해당 레퍼런스가 Toggle 되거나 Override 됩니다.

## 레퍼런스 값 쓰기:

5. RVT 를 여십시오. View 메뉴를 선택하고 Monitoring -> All 또는 Monitoring -> Active Window 을 선택하십시오.
1. 변경하려는 레퍼런스의 오프셋에 커서를 놓으십시오.
2. PLC 메뉴를 선택하고 Write Reference Value 을 선택하거나 Write Reference Value button  을 클릭하십시오.
3. 값을 입력하십시오. 수용하려면 OK button 을 클릭하십시오.

이 값은 PLC 에 있는 레퍼런스 어드레스에 쓰입니다.

Chapter
9

# 모션 프로그래밍

Motion Mate DSM314 는 Series 90-30 PLC logic solving 과 통신 기능을 고도로 통합한 사용하기 쉬우며 고성능의 다축 모션 컨트롤이 가능한 모듈입니다. VersaPro 소프트웨어 패키지는 다음과 같은 특성을 제공합니다:

- **DSM314 Configuration.** 모듈 설정치와 기본 옵션 파라미터를 선택할 수 있습니다. 구성은 SAHWC 에 의해 수행됩니다.
- **Motion program creation**. 10 개의 모션 프로그램과 40 개의 서브루틴까지 가능합니다.
- **Local Logic program creation**. Local Logic 프로그램은 모션 프로그램과 동시적으로 운용되지만 PLC 의 CPU 스캔과는 독립되어 있습니다. 이것은 DSM314 이 모션 I/O 시그널용 로직이 PLC 에서 운용되는 메인 래더 프로그램에서 운용되는 것보다 페이스 플레이트 상에서 모션 I/O 시그널을 보다 빠르게 상호작용할 수 있도록 해줍니다.

Motion Editor 와 Local Logic Editor 는 VersaPro 의 Folder Browser 에서 액세스됩니다. 하지만 일단 생성되고 저장되면 모션 프로그램과 Local Logic 프로그램들은 Hardware Configuration 의 한 부분이 되고 다른 구성정보와 함께 PLC 에 저장됩니다.

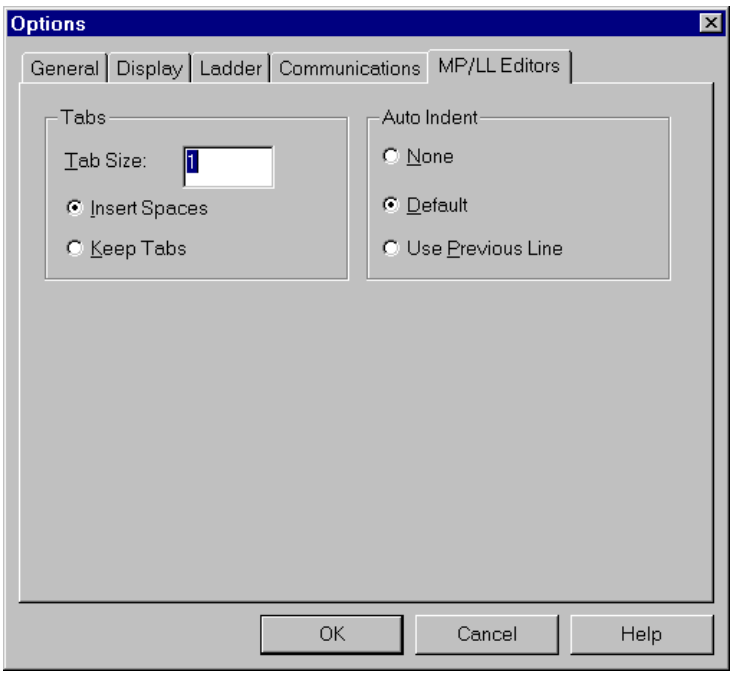
## 목차



For additional information about configuring and programming the DSM314 를 구성과 프로그래밍에 대한 추가 정보는 VersaPro 의 온라인 도움말 기능과 *Motion Mate DSM314 for Series 90-30 PLCs User's Manual* 를 참조하십시오, GFK-1742.

# 모션 프로그램/Local Logic Editor Options 설정하기

1. Tools 메뉴나 Editor CSM 에서 Options 을 클릭하십시오.
2. Options dialog box가 나타납니다.
3. MP/LL Editors Tab 을 선택하십시오.

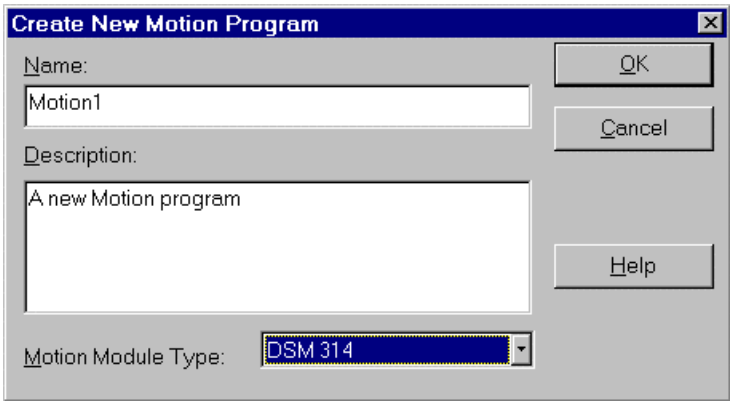



4. 원하는 Tab 과 Auto-indent options 을 선택하고 OK 를 클릭하십시오.

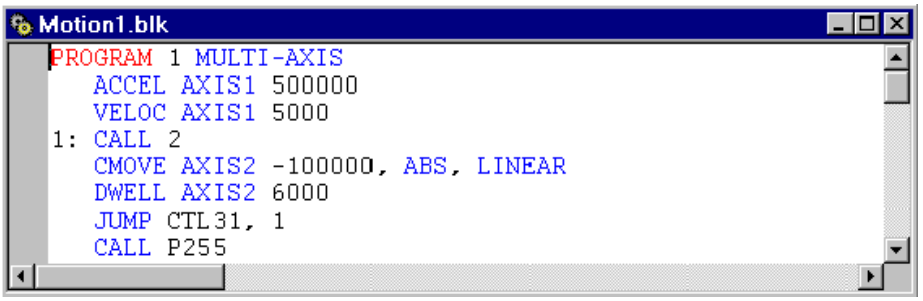
# *Motion Program Editor 사용하기*

## Motion Program Editor 열기

- 메뉴 바에서 파일을 열고 New Motion 을 선택한 후 Motion Program을 선택합니다.
- Create New Motion Program dialog box가 나타날 것입니다.




- 모션 프로그램 이름과 설명을 입력하십시오. 만약 하나 이상의 모션 모듈 타입이 drop-down listbox에서 사용 가능하다면, 이 프로그램에서 운용되는 모듈 타입을 나타내는 모션 모듈 타입을 선택하십시오. 그 다음 OK button 을 클릭하십시오. Name 을 요구할 것입니다. 설명은 옵션입니다. 새 모션 프로그램 블록을 위한 Motion Editor window가 열릴 것입니다. 모션 프로그램 이름은 Motion Editor window의 타이틀과 Folder Browser window에서 나타나는 새 모션 프로그램에 대한 아이콘으로 나타납니다.
- 텍스트 기반 모션 프로그램과 서브루틴은 다음 그림처럼 Motion Editor window에서 생성됩니다. 헤더(“PROGRAM 1 MULTI-AXIS”와 같은)를 인식함으로써 분리되는 10 개의모션 프로그램과 40 개의 서브루틴은 동일한 편집 창에서 프로그램시킬 수 있고 동일한 모션 프로그램 블록에 저장됩니다.



9

# Motion Program Logic 생성하기

**모션 프로그램 명령어**

DSM314 는 10 개의 모션 프로그램과 40 개의 서브루틴과 총 1000 개의 모션 프로그램 상태를 지원합니다. 코멘트와 White space 는 모션 프로그램 상태로 고려되지 않습니다.

## 기본 요구사항

### Program 의 수, Subroutine 과 Statements

### 포맷

- 모션 프로그램과 서브루틴은 ASCII text 를 이용해 쓰여집니다.
- 하나의 모션 언어 상태는 라인마다 허락을 받고 한 라인 이상은 연결시킬 수 없습니다. 정상 코멘트는 여러 라인을 연결시킬 수 있습니다.
- White space 와 빈 라인은 일기 능력을 향상시키고 항목을 분리하는데 상용될 수 있습니다.
- Motion Editor 는 민감한 사항은 아닙니다.
- 모든 모션 프로그램과 서브루틴은 단일프로그램 블록에 포함되어야 합니다.

### 단축과 다축 프로그램과 서브루틴

제공되는 단축 프로그램은 Program definition statement 에 명시된 1 축에서 운용할 수 있도록 해야 합니다. 그러므로 단축 프로그램과 서브루틴에 있는 모션 언어 명령어는 축을 명시하지 않습니다. 그러나 PROGRAM statement 에 명시된 축은 프로그램에서 부르는 모든 모션 명령어와 서브루틴에 사용됩니다. 다축 프로그램과 서브루틴은 다축 서브루틴만을 부를 수 있습니다. 마찬가지로 단축 프로그램과 서브루틴은 단축 서브루틴만을 부를 수 있습니다. 다축 모션 프로그램은 Axis 1 과 Axis 2 에만 유효합니다. Axis 3 과 4 단축 프로그램만을 프로그램을 해야 하는 구성에서 가능합니다.

### 프로그램과 서브루틴 정의문 스테이트먼트

Motion Editor 는 프로그램/서브루틴 번호와 축 구성이 명시된 “Program” 과 “Subroutine” 문장을 요구합니다(PROGRAM 1 MULTIAXIS 또는 SUBROUTINE 2 MULTIAXIS). 이들 Statement 는 프로그램과 서브루틴의 첫 라인에 놓입니다. 프로그램은 ENDPROG 문장으로 종결되고 서브루틴은 ENDSUB 문장으로 종결됩니다. 이들 문장은 프로그램과

서브루틴 간의 분리자의 역할을 하고 프로그램과 서브루틴의 번호를 인식하고 프로그램의 타입을 나타냅니다(단축이나 다축).

## 블록 번호와 Sync block

블록 번호는 콜론과 함께 끝에 붙을 것입니다(10: for example).블록 번호는 라인 상에 단독으로 나타나거나 동일한 라인에서 모션 명령어 앞에 나타납니다.

Sync block(SYNC 명령을 포함하고 있는 블록)은 SYNC 명령어에 의해 따라오는 블록 번호를 지닌 라인에 의해 인식됩니다(20: SYNC for example).

The SYNC (동기 블록) 명령어는 지정된 지점에서 두 축을 동기화하기 위해 다축 프로그래밍에서 사용됩니다. 추가 정보는 *Motion Mate DSM314 for Series 90-30 PLCs User's Manual*, GFK-1742 에 언급했습니다.

9

# *Local Logic Editor 사용하기*

DSM314만이 Local Logic 프로그래밍을 지원합니다. 이 모듈에 대한 추가 정보는 *Motion Mate DSM314 for Series 90-30 PLCs User's Manual*, GFK-1742 에 언급했습니다.

## Local Logic Editor 열기

- 메뉴바에서 File 을 클릭하고 New Motion 을 선택한 다음 Local Logic Program을 선택하십시오.
- Create New Local Logic Program dialog box가 나타날 것입니다.
- Local Logic 프로그램 이름과 설명을 입력하고, 모듈 타입을 선택한 다음 OK button 을 클릭하십시오. 이름이 필요하고 설명은 옵션입니다. 새 프로그램 블록을 위한 Local Logic Editor window가 열릴 것입니다. 프로그램 이름은 Local Logic Editor window의 타이틀과 Folder Browser window에서 나타나는 새 모션 프로그램에 대한 아이콘으로 나타납니다.

## Local Logic 명령어

로컬 로직 명령어는 IEC 1131 Structured Text 표준에서 파생된 free-format text 기반 구성을 사용합니다.

Local Logic Syntax는 사용하기 쉽고 프로그래밍 업무를 완성시킬 수 있는 다양한 특성을 제공합니다. Local Logic Programming 에 대한 세부사항은, *Motion Mate DSM314 for Series 90-30 PLCs User's Manual*, GFK-1742 에 언급되어 있습니다.

# *Local Logic Variable Table 사용하기*

Local Logic 은 사전에 정의된 변수 이름을 이용해 DSM314 모션 컨트롤러 변수와 파라미터 레지스터를 액세스합니다. Local Logic Variable Table (LLVT)는 6 탭으로 구성된 세 가지 변수를 디스플레이합니다: Axis 1 – 4, Global, CTL Bits 와 Parameter Registers.

각각의 축 변수는 제어 변수, 상태 변수와 Faceplate I/0 를 포함하고 있습니다.

### LLVT 에서 작업하기 위한 트립

- 컬럼 폭을 변경하려면 컬럼 헤드 오른쪽에 있는 디바이더를 선택하여 드래그합니다.
- LLVT 에서 선택된 영역의 컨텐츠를 클립보드에 복사하려면 Ctrl + C 를 누르거나 CSM 에서 Copy 를 선택하십시오. 단일 셀을 선택하려면, LLVT 에서 클릭합니다. 인접한 셀의 블록을 선택하려면 클릭하고 드래그 또는 클릭하고 Shift key 를 누른 상태에서 다시 클릭합니다. 테이블의 상단과 좌측에 있는 보더를 클릭하거나 클릭과 드래그하여 줄과 컬럼을 선택합니다.
- LLVT 를 소트하기 위해, 소트하고자 하는 컬럼을 클릭한 다음 CSM 이나 View 메뉴에서 Sort 를 선택하십시오. 서브메뉴에서 Ascending 이나 Descending 을 선택하라고 나타납니다.
- LLVT 을 인쇄하려면, LLVT 을 선택하고 File 메뉴에서 Print 를 선택하거나 Ctrl + P 를 누르십시오.

## Local Logic Variable Table 열기

LLVT 을 열려면, View 메뉴에서 Local Logic Variable Table 을 선택하고 Alt + 6 을 누르거나 툴바에 있는 Toggle Local Logic Variable Table 버튼을 누르십시오. 

# Local Logic Variable Table 에 의해 제공되는 정보

LLVT 는 다음 탭을 포함하고 있습니다: Axis 1, Axis 2, Axis 3, Axis 4, Global, CTL Bits 와 Parameter Registers

각각의 탭은 다음 정보를 제공합니다.:

### Name

이 이름은 변수 이름을 포함하고 있으며 Local Logic 프로그램에서 사용될 수 있습니다.

### Type

변수에 대한 데이터 타입.

주어진 변수 크기보다 큰 값을 쓰려고 하면 불완전한 상태가 될 것입니다. 예를 들어, 수식 운용의 결과가 32bit 의 길이인데 16bit 로 할당된다면 16bit 형태로 저장될 것입니다. Non-Boolean Math 동작에서 목적 변수로 사용된다면 Parser 는 Warning 을 리포트합니다(결과의 최소한의 중요한 bit 만이 저장됩니다).

32bit 이하인 Local Logic 변수는 **Signed** 이나 **Unsigned** 입니다(항상 Unsigned 인 Bit Operands 는 제외). Logic Engine 에 있는 모든 Math/Logic 동작은 signed 32 bit 동작입니다(64 bit signed Divide 와 Modulus 동작 제외). 32bit 이하인 Signed 변수는 Logic Engine 에 로드될 때 자동적으로 32bit 로 확장됩니다. Unsigned 변수는 Sign 으로 확장되지 않습니다. 그러므로 Logic Engine 은 자동적으로 모든 데이터의 변환과 제한적인 체크를 합니다.

**Note**: 즉각적인 명령어 (%AQ) 변수는(Torque Limit, Velocity Loop Gain, Follower Ratio, Position Increment 와 Position Loop Time Constant) Local Logic 변수 크기보다 작은 허용 범위를 가질 수 있습니다. 만약 프로그램이 허용 범위 이외의 값을 쓰면 모듈은 Warning 에러 코드를 리포트하고 잘못된 값을 거절합니다. 각각의 immediate command 에 대한 허용 범위의 설명은 *Motion Mate DSM314 for Series 90-30 PLCs User's Manual*, GFK-1742 에 언급되어 있습니다.

### Group

축 지향 변수는 컨트롤 변수, 상태 변수와 Faceplate I/0 을 포함하고 있습니다 (Global tab 은 Global variable 만을 포함합니다.)

### Description

이 컬럼은 변수 설명을 포함하고 있습니다. 셀에 커서를 올려 놓고 완전한 설명을 보려면 *Motion Mate DSM314 for Series 90-30 PLCs User's Manual,* GFK-1742 를 참조하십시오.

### R

이 컬럼이 체크되면 변수는 Local Logic 프로그램에 의해 읽혀질 수 있습니다.

### W

이 컬럼이 체크되면 변수는 Local Logic 프로그램에 의해 쓰일 수 있습니다.

**Note:** 변수는 **Read-Only**, **Write-Only** 또는 **Read-Write** 될 수 있습니다. Read-Only 변수로 쓰거나 Write-Only 변수로 읽으려 한다면 Parser 는 에러를 리포트합니다.

## *Motion/Local Logic Program 저장하기*

컴퓨터 하드 디스크에 모션 프로그램/서브루틴을 저장하려면, 툴바에서 Save icon 을 클릭하거나 메뉴 바에서 File 을 클릭하고 Save 를 클릭하십시오.

## *Motion/Local Logic Programs 인쇄하기*

모션 프로그램을 인쇄하는데 Print 기능이나 Print Report 기능 등 두 가지 방법을 사용할 수 있습니다. 이들 항목들은 File 메뉴에서 찾을 수 있습니다.

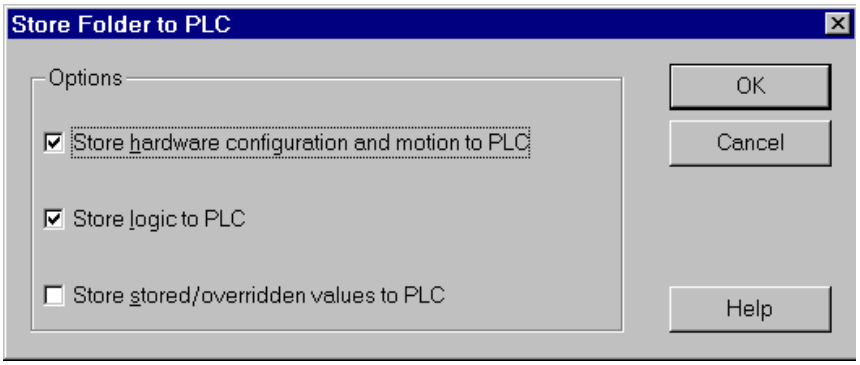
Print 기능을 사용하려면, 메뉴 바에 있는 File 을 클릭하고 Print 를 선택합니다. Printer dialog box 가 나타날 것입니다. 원하는 프린터 설정을 변경한 다음 OK button 을 클릭하십시오.

폴더에 있는 다른 정보를 지닌 리포트의 한 부분으로 모든 모션 프로그램을 인쇄하려면 메뉴 바에서 File 을 클릭하고 Print Report 를 선택합니다. Print report dialog box가 나타날 것입니다. Print Report dialog box에서 Blocks checkbox를 클릭하십시오. 모션 프로그램과 로컬 로직 프로그램은 이 리포트의 한 부분으로 인쇄됩니다.

만약 선택된 모션 프로그램 블록만을 인쇄하고자 한다면 브라우저에 있는 이들 블록을 밝게 하고 File 을 클릭하여 Print Report 를 선택하십시오. Blocks checkbox 를 체크하고 Selected radio button 을 선택하십시오. 이것은 선택된 블록만을 리포트 하도록 제한하고 있습니다.

# PLC 에서의 Motion Program 과 Subroutine 저장하기

모션 프로그램과 로컬 로직 프로그램은 하드웨어 구성의 한 부분으로 저장됩니다. Motion/Local Logic 프로그램을 저장하려면, STOP 모드에 PLC 를 두고 PLC 메뉴에서 Store 을 클릭하십시오. 대화창에서 Store Hardware Configuration and Motion to PLC 를 선택하십시오.




# Motion 과 Local Logic Blocks 복구하기

열려 있는 VersaPro 폴더에서 모션과 로컬 로직 블록의 백업 복사는 동일한 블록의 이름 포함하고 있는 PLC 에서 모든 모션과 로컬 로직 프로그램을 포함하여 PLC 구성을 로드할 때 자동으로 만들어집니다. VersaPro 는 폴더에 저장되어 있는 모션과 로컬 로직 블록과 PLC 에 저장된 블록 사이에 차이점이 있다면 단지 백업만을 생성할 것입니다. 유저 프롬프트는 백업을 생성하기 원하는지를 질문할 것입니다. 한 블록에 대한 여러 백업을 자동으로 만들어지지 않습니다. 이 복구 메커니즘은 VersaPro 폴더에 저장되어 있는 코멘트에 손실을 주지 않고 수정된 모션 프로그램 블록을 복구하는 방법을 제공합니다. 코멘트를 포함하고 있는 VersaPro 폴더를 사용한다면 코멘트는 PLC 에 저장시키지 않고 사용할 수 있습니다.

Motion (*.mbk)이나 Local Logic (*.lbk) block 을 저장하려면 다음 단계를 수행하십시오:

1. Folder 메뉴에서 Restore Motion Blocks 을 선택하십시오.
2. Restore Blocks dialog box가 나타날 것입니다. Available Blocks 리스트는 폴더에서 발견되는 모든 모션과 로컬 로직 블록의 백업 리스트를 포함하고 있습니다. 복구하기 위해 하나 또는 그 이상의 블록을 선택할 수 있습니다. 만약 PLC 와 VersaPro 폴더 모션 블록들이 PLC process 에서 로드하는 동안 동일하다면 모션 백업은 만들어지지 않고 리스트는 빈 상태가 될 것입니다. 하지만 처음 로드하는 것이 아니고 PLC 와 VersaPro 폴더 사이에 모션 블록 차이점이 있다면, 이들 백업은 리스트에 디스플레이 될 것입니다. 다시 말해 백업 모션 블록을

복구할 때, 백업은 폴더에 남아 있게 되고 백업을 물리적으로 제거할 수 없다면 리스트에 디스플레이 할 것입니다.

3. 단일 모션 블록을 복구하려면, 블록을 밝게 하고 Add button 을 클릭하십시오.
4. 여러 블록을 복구하려면 Add All button 을 클릭하거나, Ctrl 이나 Shift key 를 누르고 마우스를 이용해 리스트의 블록이나 섹션을 선택한 다음 Add button 을 클릭하십시오.
5. 블록에서 저장된 리스트까지 블록을 제거하려면 블록을 선택하고 Remove Button 을 클릭하십시오.
6. 리스트에서 모든 블록을 제거하려면 Remove All button 을 클릭하십시오.
7. 선택된 블록을 제거하려면 Ctrl 나 Shift key 를 누르고 블록을 클릭한 다음 Remove button 을 클릭하십시오.
8. 리스트에 만족한다면 OK button 을 클릭하십시오. 폴더에 있는 블록(PLC 에서 로드된)은 선택한 백업(VersaPro 폴더에서)으로 대체될 것입니다.
9. 대체 동작을 취소하려면 Cancel button 을 누르십시오.

**Note:** 블록을 복구시킬 때, 백업은 삭제되지 않습니다. 디스크 공간이 부족하거나 백업을 남겨 두지 않으려면 폴더를 네비게이션하여 백업을 수동으로 삭제하기 위해 MS-Explorer 나 Computer 를 사용해야 합니다. 백업 파일은 .mbk (motion)나 .lbk (local logic) 확장자를 갖습니다.

Chapter
10

# PLC 와 통신하기

PLC 와 통신하려면 프로그램을 저장하고 로딩하거나 레퍼런스나 PLC 상태 정보 보기와 같은 운용을 수행하는데 필요합니다. 이 장은 PLC 를 연결하는 방법과 기본적인 온라인 운용을 수행하는 방법을 기술합니다.

## 목차

이 장은 다음과 같은 정보를 제공합니다:

- PLC 와 컴퓨터 연결하는 방법
- PLC 로 또는 PLC 에서 전송하는 방법
- PLC 에서 폴더 데이터를 지우는 방법과 정보를 확인하는 방법
- 플래쉬 메모리를 읽고/쓰고/확인하는 방법
- PLC 상태 정보를 보는 방법
- Run 이나 Stop 모드에서 PLC 배치하는 방법
- 로직 실행 모니터링하기

10

# *PLC 와 연결하기*

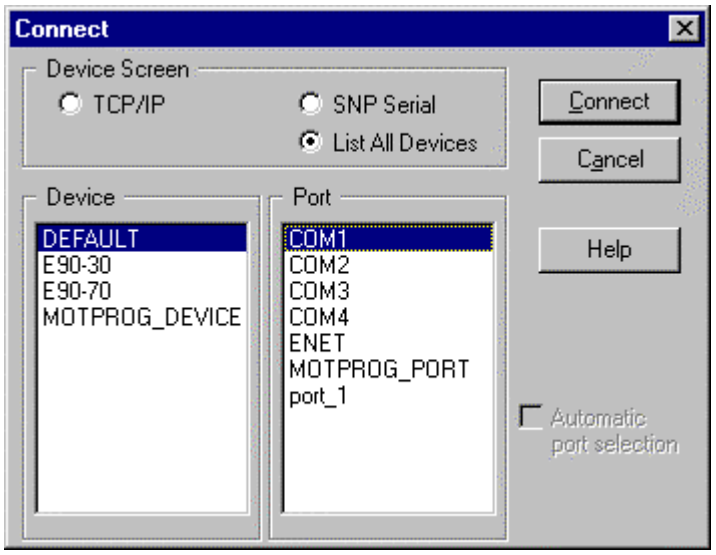
PLC 용량에 따라 시리얼이나 이더넷 연결에 의해 VersaPro 에서 PLC 를 연결할 수 있으며, 디바이스를 확인하고 포트를 선택함으로써 연결시킬 수 있습니다. 디바이스는 기본 PC 통신 포트, 타깃 PLC 타입, PLC SNP 어드레스와 PLC IP 어드레스를 포함하여 성공적으로 연결하는데 필요한 정보를 명시하고 있습니다. 포트는 PLC 에서 통신을 초기화하는데 사용되는 PC 포트를 인식합니다.

이 장은 PLC 를 연결하는 다양한 방법을 기술하고 있습니다.

## Connect Dialog Box 사용하기

PLC 메뉴를 선택하고 Connect 를 선택하면, Connect dialog box 가 나타납니다. 이 dialog box 는 PLC 와 함께 통신을 초기화하기 위해 디바이스와 포트를 선택하는데 사용됩니다.

PLC 와 통신하는데 가장 널리 사용되는 방법은 SNP Direct 입니다. SNP Direct 연결을 쉽게 하기위한 특수 디바이스 이름인 DEFAULT 는 SNP 기본설정에 맞춰 COM1 에 구성되어 있습니다. 이 디바이스는 기본 프로토콜 설정치(19200, odd parity, 1 stop bit)를 이용해 SNP Direct 연결할 수 있도록 작업할 것입니다.

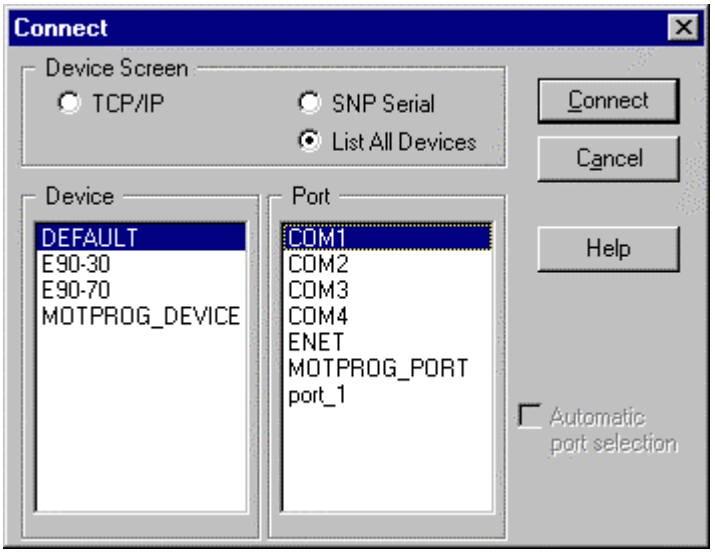


# Direct Serial Port Connection

Connect dialog box에 있는 디바이스 리스트는 DEFAULT 를 포함하고 있습니다. PC 에 있는 COM1 에서 PLC 에 있는 내장형 SNP 포트로 연결할 수 있어, 기본값은 PC 와 PLC 를 직접 연결하는데 사용하도록 설계되었습니다. 이 방법은 PLC 에 연결하는 가장 간편한 방법입니다.

COM1 에 직접 연결하려면 다름 단계를 따르십시오:

1. PC 에 있는 COM1 에서 PLC 에 있는 RS-232/RS-485 컨버터까지 시리얼 케이블로 연결하십시오.
2. PLC 메뉴를 선택하고 Connect 를 선택하십시오. Connect dialog box가 나타날 것입니다.




3. 디바이스 “DEFAULT”와 포트 “COM1”을 선택하십시오. 통신 설정을 마치려면 Connect button 을 클릭하십시오.

# Multidrop 과 Modem Connection

Multidrop 이나 Modem 구성은 결코 디바이스에 대해 정의되지 않거나, PLC 구성은 이러한 연결을 지원하기 위해 설정되어 있지 않았다면 아래 명령어를 따르십시오.

PC 에서 PLC 까지의 Physical Connection 은 애플리케이션에 달려 있습니다; 케이블과 연결에 대한 세부사항은 부록 B, “Communications Cabling”을 참조하십시오.

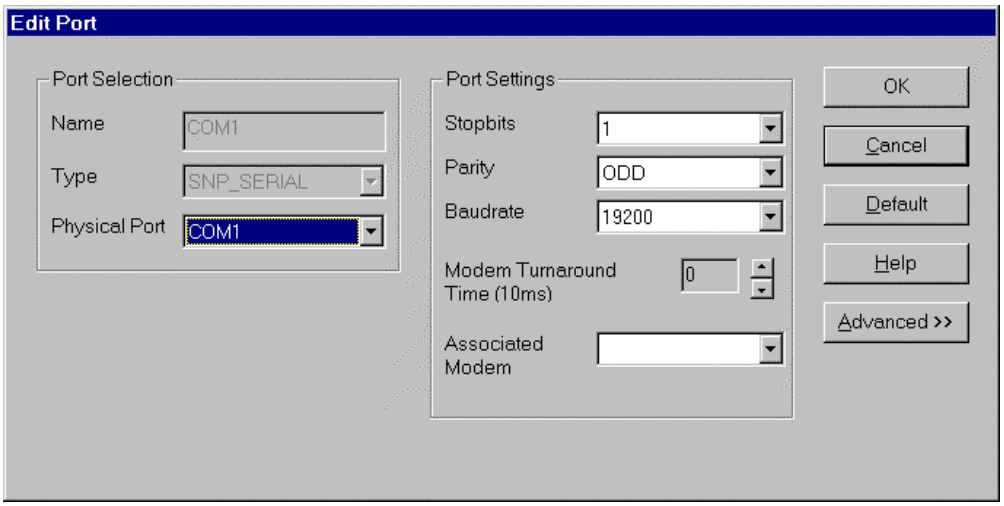
## 디바이스와 PLC CPU 구성하기

애플리케이션이 SNP default(19200, ODD, 8 Data Bits, 1 Stop Bit)에서 기본 시리얼 포트 설정치 변경이나 SNP ID 설정을 필요로 한다면, 다음과 같이 실행시키십시오:

1. Hardware Configuration(HWC)을 시작하고 PLC CPU 를 더블 클릭하십시오.
2. Parameters dialog box 에서 Settings tab 을 클릭하십시오:
3. 시리얼 포트 설정치나 SNP ID 를 변경하고 OK button 을 클릭하십시오.
4. PLC 에 하드웨어 구성 변경을 저장하십시오.

일단 PLC CPU 에 대한 SNP 기본 설정치를 변경을 했다면, PC 포트에 대한 설정치를 변경해야 합니다. PC 에 대한 포트 구성을 수정하려면 다음 단계를 수행하십시오.

1. Tools 메뉴에서 Communications Setup 을 선택하십시오. password dialog box 가 나타날 것입니다. 작업을 계속하려면 패스워드를 입력하고 OK 를 누르십시오(기본 패스워드는 **netutil** 입니다.). Communication Configuration Utility 의 메인 화면이 나타날 것입니다.
2. 새로운 디바이스를 추가하기 위해 New button 을 클릭하십시오. Add New Device dialog box 가 나타날 것입니다:
3. PC 포트와 연결되는 기본 포트를 설정하고(COM1 ~ COM4), PLC 에 대한 SNP ID 를 입력하십시오. 디바이스를 추가하려면 OK 를 누르십시오
4. 기본 시리얼 포트 설정치를 변경하려면 Ports tab 를 클릭하십시오. 그 다음에 수정하고자 하는 PC 포트를 밝게 하고 Edit 를 클릭하십시오. Edit Port dialog box 가 나타날 것입니다. 애플리케이션에 알맞게 포트 구성을 하십시오(SNP 타이머는 Advanced button 을 클릭함으로써 액세스 시킬 수 있습니다). 포트에 대한 변경을 저장하려면 OK 를 클릭하십시오.




5. PC 포트(COM1 ~ COM4)를 설정하고 PLC SNP 포트 중 하나가 존재한다면 SNP ID 를 입력하십시오. SNP ID 가 없다면 NULL 을 입력하십시오. 디바이스를 추가하려면 OK 를 클릭하십시오.

## PLC 와 연결하기

PLC 와 연결하려면 다음 단계를 수행하십시오:

1. Folder browser 에서 PLC 메뉴를 선택하여 Connect(또는 PLC toolbar 에 있는 Connect button  을 클릭하십시오)를 선택하십시오.
2. Device 이름을 선택하고 Port 이름을 Communications Utility dialog box 에서 구성하십시오. 통신을 수행하려면 Connect button 을 누르십시오.
3. 연결이 완성되면 스크린 밑부분에 있는 상태 정보는 Connected 로 바뀔 것입니다.

# TCP/IP Ethernet Connection

TCP/IP Ethernet 연결 구성은 특수한 폴더와 관련되어 있지 않고 VersaPro 에서 셋업의 한 부분이 됩니다.

만약 TCP/IP Ethernet 구성이 디바이스에 대해 정의되어 있지 않거나 PLC Ethernet Interface 가 없다면 “Configuring Your PLC and PLC Hardware”에 있는 명령어를 따르십시오

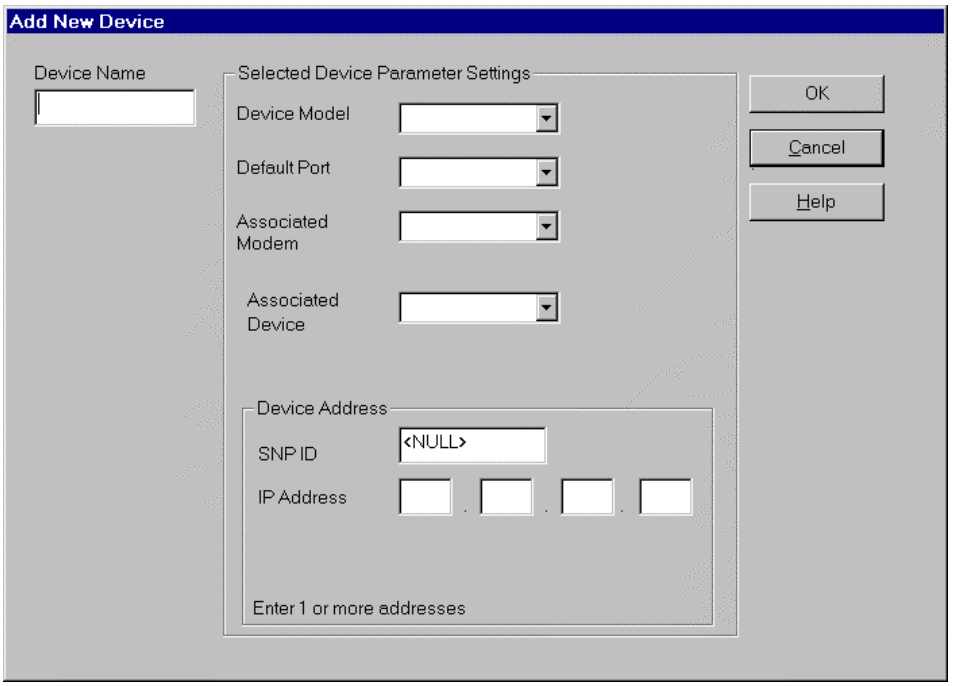
# PC 와 PLC 구성하기

TCP/IP Ethernet 을 통해 연결하기 전에 PLC Ethernet Interface 에서 IP 어드레스를 설정해야 합니다. PLC IP 어드레스를 구성하려면 다음 단계를 수행하십시오:

1. Hardware Configuration 에서, Series 90-30 rack 에 Ethernet 모듈이나 CPU364 를 추가하십시오.
2. 모듈에 대한 IP 어드레스를 설정하십시오. PLC CPU 에 하드웨어 구성을 저장하십시오.

일단 PLC CPU 에 대한 IP 어드레스를 구성하면 PC 포트에 대한 설정치를 변경해야 합니다. PC 에 대한 포트 구성을 수정하려면 다음 단계를 수행하십시오.

1. Ethernet 네트워크에 PLC 와 PC 를 연결하십시오.
2. Tools 메뉴에서 Communication Setup 을 선택하십시오. password dialog box 가 나타날 것입니다. 작업을 계속하려면 패스워드를 입력하고 OK 를 누르십시오(기본 패스워드는 “netutil”입니다.). Communication Configuration Utility 의 메인 화면이 나타날 것입니다.
3. 새로운 디바이스를 추가하려면 New button 을 클릭하십시오. Add New Device dialog box 가 나타날 것입니다:




4. ENET 에 Default Port 를 설정하고 Ethernet Interface 에서 구성된 IP 어드레스와 연결되는 IP 어드레스를 입력하십시오.

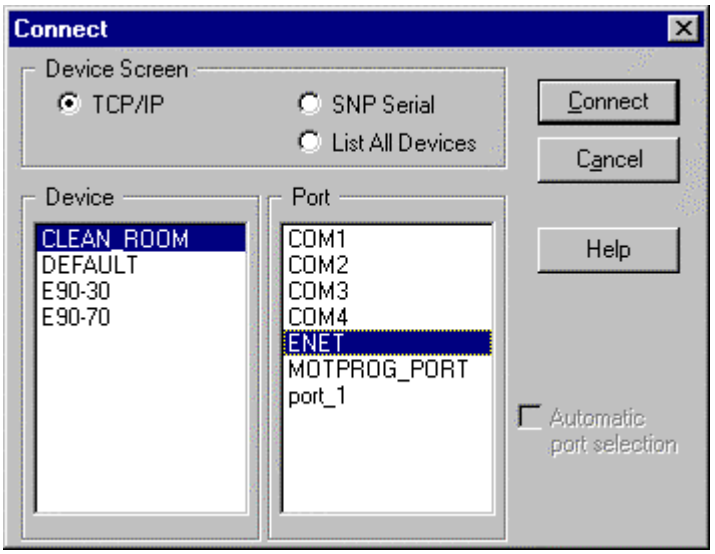
5. 글로벌 타임아웃을 만들려면 Ports tab 을 클릭하여 ENET 을 선택하고 Edit 을 클릭하십시오. 클로벌 타임아웃을 액세스하려면 Advanced 를 클릭하십시오.

**Note:** CCU 통해 변경된 Modem Properties 는 운용 시스템 모뎀 설정에 저장되지 않습니다. 이들 변경사항을 유지하려면, Windows Control Panel 에 있는 Modem Properties 를 편집해야 합니다(Control Panel 를 선택한 다음 설정하십시오.).

## PLC 로 연결하기

PLC 로 연결하려면 다음 단계를 수행하십시오:

1. PLC 메뉴를 선택하고 Connect 를 선택하십시오(또는 PLC toolbar 에 있는 Connect button  을 클릭하십시오). Connect dialog box 가 나타날 것입니다. Ethernet 통신에 대한 Device Name 과 Port(ENET)를 설정하십시오. 통신을 수행하려면 Connect button 을 클릭하십시오.




2. 연결이 완성되면 화면 밑부분에 있는 상태 정보는 Connected 로 바뀔 것입니다.

# 폴더 전송하기

PLC 로의 연결을 마친 후에 컴퓨터와 PLC 간의 폴더를 전송할 수 있습니다. 이 섹션은 다음과 같은 내용을 포함하고 있습니다:

- PC 에서 PLC 로 폴더 저장하기
- PLC 에서 PC 로 폴더 로딩하기
- 폴더가 현재 PLC 에서 운용중인 폴더와 부합되는지 확인하기
- PLC 에서 항목 지우기
- PLC 플래쉬 메모리에서 읽고/쓰기 수행하기

## PLC 에 폴더 저장하기

PLC 에 하드웨어 구성, 로직 또는 저장/오버라이드 값을 저장할 수 있습니다.

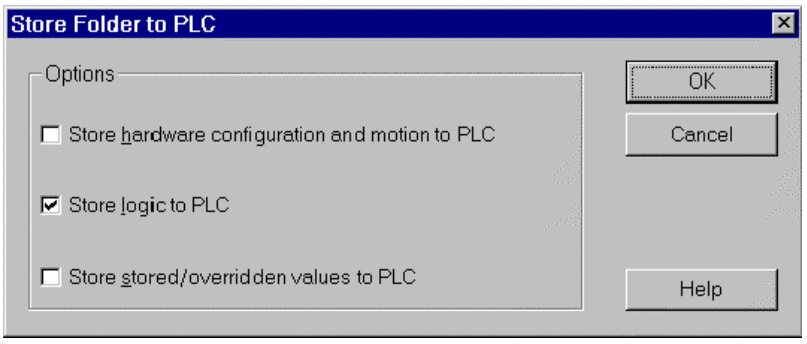
대부분의 경우, PLC 에 저장하기 전에 Stop 모드에 PLC 를 설치해야 합니다. 프로그램 변경이나 PLC 닉네임과 일치하는 폴더 닉네임을 저장할 때, Run 모드 저장을 수행할 수 있습니다.

Stop 모드에 PLC 를 설치하려면 Stop toolbar button을 클릭하거나 PLC 메뉴로 가서 Stop PLC 를 선택하십시오.

## 폴더 저장하기

컴퓨터에서 연결된 PLC 로 현 폴더를 저장하려면 다음 단계를 수행하십시오:

1. PLC 를 연결했는지 확인하십시오(VersaPro 의 상태 바는 "Connected."를 나타내야 합니다. 만약 상태 바를 볼 수 없다면 Ctrl + 0 을 누르십시오.).
2. Store Button 을 클릭하십시오(또는 PLC 메뉴를 선택하고 Store 을 선택하십시오). Store Folder to PLC dialog box가 나타날 것입니다.



3. PLC 에 저장하고자 하는 폴더 콤포넌트를 선택하십시오: 하드웨어 구성과 모션, 로직과 저장/오버라이드 값(Motion Mate DSM314 모듈에 대한 모션 프로그램은 하드웨어 구성의 한 부분으로 저장됩니다.)
4. VersaPro 는 PLC 에 선택된 콤포넌트를 저장할 것입니다. 저장을 성공적으로 수행했다면 Message box 가 나타날 것입니다. Information Window 는 저장 방법에 대한 부수적인 세부사항을 제공할 것입니다.

## Word for Word Changes in LD

Word for word changes 는 Series 90-30, Series 90 Micro 와 VersaMax modular PLC 에서 수행할 수 있습니다(VersaMax Nano/Micro PLC 는 word for word changes 기능을 지원하지 않습니다.)

프로그램을 수정하지 않고 프로그램을 변경하려면 word-for-word changes 기능을 고려하십시오. 콘택트나 코일타입을 변경하거나 기존 펑션 블록을 이용해 레퍼런스 어드레스를 변경하는 것도 포함됩니다.

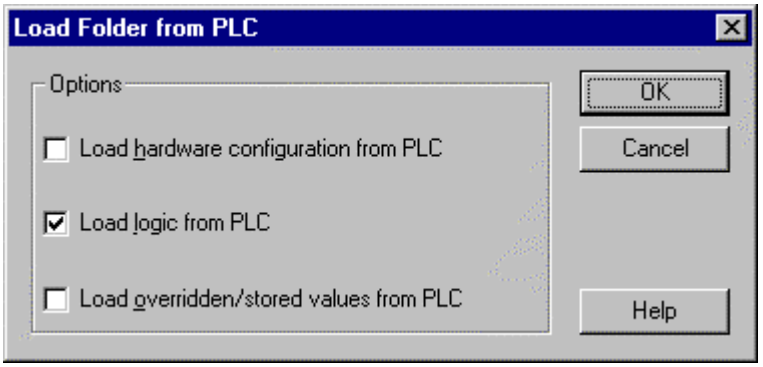
# 컴퓨터로 폴더 로딩하기

PLC 에서 PC 로 하드웨어 구성과 모션, 로직과 저장/오버라이드 값을 로드할 수 있습니다.

### 폴더 로딩하기

PLC 에서 컴퓨터로 폴더를 로딩하려면 다음 단계를 수행하십시오:

1. PLC 로 연결되었는지 확인하십시오(VersaPro 의 상태 바는 “Connected”를 나타내야 합니다. 만약 상태 바를 볼 수 없다면 Ctrl + 0 을 누르십시오.)
2. Load Button 을 클릭하십시오(또는 PLC 메뉴를 선택하고 Load 을 선택하십시오). Load Folder to PLC dialog box 가 나타날 것입니다:




3. 로드하려고 하는 콤포넌트를 선택하십시오. 하드웨어 구성과 모션, 로직과 저장/오버라이드 값에서 선택하십시오 .
4. 로드하려면 Load button 을 클릭하십시오.

5. 선택된 정보는 PLC 에서 로드될 것입니다. 정보 창은 로드 운용 결과를 디스플레이할 것입니다. 로드 성공적이며 메시지 박스는 “Load from PLC Completed”를 디스플레이할 것입니다.

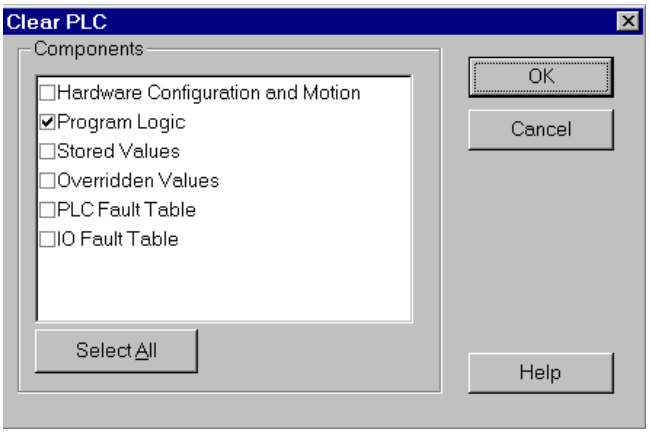
**Note:** PLC 에서 Ethernet Global Data 나 Name Resolution 은 로드할 수 없습니다.

# PLC 지우기

하드웨어 구성, 로직, 저장/오버라이드 값, PLC 와 PLC 에 있는 I/O Fault Table 을 지울 수 있습니다.

PLC 에서 선택한 프로그램 항목을 지우려면, 다음 단계를 수행하십시오:

1. PLC 를 연결했는지 확인하십시오(VersaPro 의 상태 바는 “Connected”를 나타내야 합니다. 만약 상태 바를 볼 수 없다면 Ctrl + 0 을 누르십시오.)
2. PLC 메뉴를 선택하여 Clear 를 선택하십시오. Clear PLC dialog box 가 나타날 것입니다.

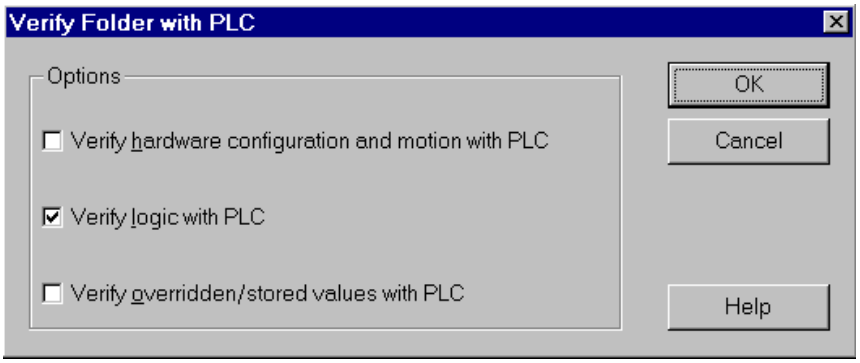



3. PLC 에서 지우려는 항목을 선택하십시오. 항목 옆에 표시된 콤포넌트는 지워질 것입니다. 항목을 클릭하여 개별 콤포넌트를 선택하거나 선택 취소를 할 수 있습니다.
4. 항목을 지우려면 OK button 을 누르십시오.
5. 지우기가 완성되면 메시지 박스는 “Clear Completed Successfully”를 디스플레이할 것입니다.

# *Folder Data 확인하기*

PLC 에서 현재 운용되고 있는 것과 일치하는 폴더를 확인하려면 다음 단계를 수행하십시오. PLC 와 함께 하드웨어 구성, 로직, 저장/오버라이드 값을 확인할 수 있습니다.

3. PLC 를 연결했는지 확인하십시오(VersaPro 의 상태 바는 “Connected”를 나타내야 합니다. 만약 상태 바를 볼 수 없다면 Ctrl + 0 을 누르십시오.)
1. PLC 메뉴를 선택하여 Verify 를 선택하거나 Verify Folder with PLC 을 클릭하십시오. 
2. Verify Folder with PLC dialog box가 나타날 것입니다.




6. 항목 옆에 표시된 콤포넌트가 확인될 것입니다. 항목을 클릭하여 개별 콤포넌트를 선택하거나 선택 취소를 할 수 있습니다.
4. 항목을 확인하려면 OK button 을 누르십시오.
5. 메시지 박스가 확인이 완성되었다는 것을 확인할 것입니다. 정보 창은 확인 운용에 대한 세부사항을 포함할 것입니다.

# *Flash Memory 읽기/쓰기/확인하기*

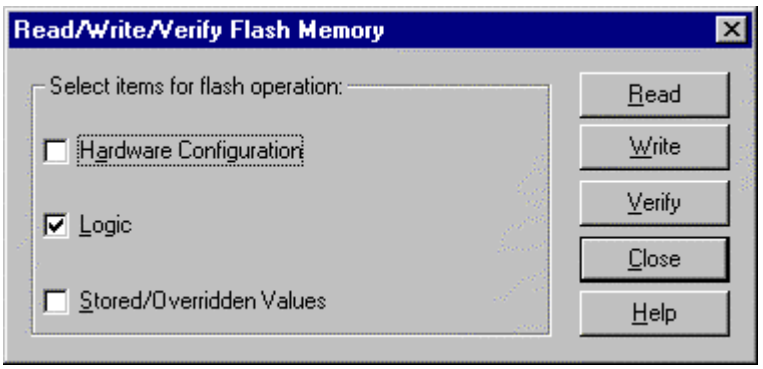
어떤 CPU 모듈은 PLC에 위치한 플래쉬 메모리 디바이스에 폴더 데이터를 저장할 수 있도록 하기 위해 플래쉬 메모리 옵션을 지원합니다.

다음 플래쉬 메모리 특성들은 사용할 수 있습니다.:

- **Read:** PLC 메모리에 있는 선택된 플래쉬 메모리의 컨텐츠를 읽음
- **Write:** 플래쉬 메모리에 있는 선택된 PLC 메모리의 컨텐츠를 씀
- **Verify:** CPU와 연결된 플래쉬 메모리의 컨텐츠를 확임함

PLC의 플래쉬 메모리에서 읽기, 쓰기 및 확인 기능을 수행하려면 다음 단계를 수행하십시오 :

1. PLC에 연결되었는지 확인하십시오.
2. PLC 메뉴를 선택하고 Flash/EEPROM 을 선택하십시오. Read/Write/Verify Flash Memory dialog box가 나타날 것입니다.

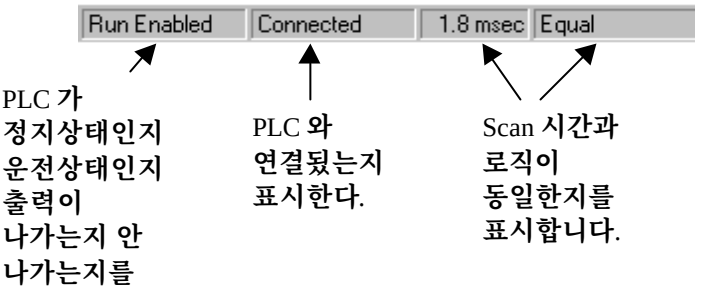



3. 플래쉬 운용을 하려는 폴더 콤포넌트를 선택하십시오. 콤포넌트를 클릭하여 개별 콤포넌트를 선택할 수 있습니다.
4. 정확한 버튼을 클릭함으로써 플래쉬 기능을 선택하십시오 :
   - CPU에서 플래쉬 메모리를 읽으려면, Read button 을 클릭하십시오.
   - CPU에서 플래쉬 메모리로 쓰려면, Write button 을 클릭하십시오.
   - CPU의 프로그램과 플래쉬 메모리의 매치되는지 확인을 하려면 Verify button 을 클릭하십시오.
5. 기능이 완성되면 메시지 박스는 성공이나 에러를 나타낼 것입니다.

# *PLC 상태 정보 보기*

VersaPro 가 PLC 와 연결되면 윈도우 밑부분에 있는 Status line 이나 Status Window dialog box 를 통해 PLC 상태 정보를 볼 수 있습니다.

PLC 를 연결했을 때, 윈도우 밑부분에 있는 Status line 은 상태 정보를 요약한 디스플레이를 제공합니다.




보다 상세한 정보는 Status information dialog box 를 통해 볼 수 있습니다. 이 정보를 보려면 PLC 메뉴를 선택하여 Status Info 를 선택하거나 F4 를 누르십시오.

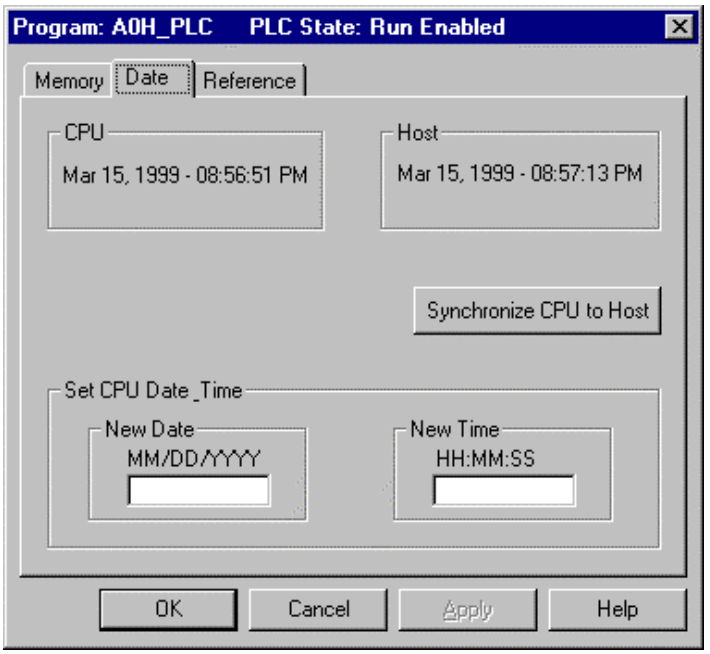
Target Communications Window 의 Tab 는 다음과 같은 정보를 포함하고 있습니다:

- **Memory:** CPU 디바이스 정보, 현재 저장된 프로그램 이름과 CPU 메모리 사용법 제공
- **Date:** 날짜와 시간 보기, 변경이나 동시화 해줌
- **Reference:** PLC CPU 뿐만 아니라 모든 포인트 오류 레퍼런스에 있는 레퍼런스 테이블에 대한 크기 디스플레이
- **Protection:** 액세스 레벨과 OEM 보호 상태 디스플레이. 액세스 패스워드나 OEM key 를 설정할 수 있게 해줍니다.
- **Sweep:** Sweep 모드와 시간 디스플레이. VersaPro 가 연결 되면 PLC 가 RUN 모드에 있는 동안 Sweep Mode 와 Constant Sweep Time 을 설정할 수 있습니다. 이러한 변경은 STOP 모드로 넘어 갈 때까지 PLC 에 영향을 주지 않습니다. PLC 가 RUN 모드로 되돌아 갈 때, 기본 Sweep 모드가 실행됩니다.

**Note:** 이 탭의 설정치는 PLC 에 저장되고 프로그램 폴더에는 저장되지 않습니다.

## 날짜와 시간 보기와 변경하기

현재 PLC 시간과 날짜를 보려면, Date tab 을 선택하십시오.




만약 CPU 의 날짜와 시간이 호스트 컴퓨터의 날짜와 시간이 맞지 않는다면 CPU 클록을 동기화하든가 CPU 의 날짜와 시간을 설정할 수 있습니다.

- **Host 에서 CPU 동기화 :** Host button 에서 Synchronize CPU 를 클릭하십시오.
- **CPU Date & Time 설정 :** New Date field 에 연월일(옵션)을 입력하고 New Time field 에 시분초(옵션)를 입력하십시오. 변경사항을 받아들이려면 Apply button 을 클릭하십시오.

# Run 또는 Stop Mode 에서 PLC 설치하기

Run 이나 Stop 모드에서 PLC 를 수동으로 설치할 수 있습니다. Run 모드에서, 출력을 가능하게 하거나 불가능하게 할 수 있도록 선택할 수 있으며, Stop 모드에서, I/O Scan 을 가능하게 하거나 불가능하게 할 수 있도록 선택할 수 있습니다.

## Run Mode 에서 PLC 설치하기

1. PLC 가 연결되었는지 확인하십시오.
2. Run PLC button  을 클릭하십시오(또는 PLC 메뉴에서 Run 을 선택하십시오).

## Stop Mode 에서 PLC 설치하기

1. PLC 가 연결되었는지 확인하십시오.
2. Stop button  을 클릭하십시오(또는 PLC 메뉴에서 Stop 을 선택하십시오).
3. Stop Execution dialog box에서, 가능한 출력과 불가능한 출력을 선택한 다음, OK button 을 클릭하십시오.

10

# 로직 실행 모니터링하기

VersaPro 는 로직을 실행하는 동안 Reference View Table, Variable View Table, Ladder Editor 나 Instruction List Editor 에서 요소나 오퍼랜드의 값이나 상태를 볼 수 있게 해줍니다. 이것은 로직 실행의 테스트, 실행 폴더에서 이벤트 트리거 또는 로직 실행에서 요소를 제거할 수 있는 기회를 제공합니다. 또한 여러 블록을 동시에 모니터할 수 있습니다. 온라인으로 값을 모니터하기 위해서는 로직이 PLC 에서 운용되어야 하고 VersaPro 는 논리적으로 PLC 를 연결해야 합니다.

## Monitoring On/Off 조절하기

1. 모니터하고자 하는 실행 로직을 가진 PLC 를 연결하십시오.
2. PLC 상에 있는 폴더와 같은 현재 폴더에서 로직을 확인하십시오.
3. View 메뉴에서 Monitor 를 선택하십시오.
4. 나타난 서브메뉴에서 활동중인 윈도우 또는 모든 윈도우에 있는 요소들이 실시간으로 업데이트 되는지 결정하기 위해 Active Window 이나 ALL 을 선택하십시오.

   만약 Active Window 를 선택했다면, 활동중인 윈도우는 로직 실행에서 데이터를 업데이트 합니다.

   만약 ALL 를 선택했다면, 모든 활동중인 윈도우는 로직 실행에서 데이터를 업데이트 합니다.

   - IL, LD, VVT 와 RVT 들은 모니터할 수 있는 유일한 활성화된 윈도우입니다.
   - Single Active Window 는 TCP/IP 연결 상에서 초당 10 회정도 업데이트합니다.

- "All"이 선택되고 하나 이상의 윈도우가 열리면 활성화된 윈도우는 항상 다른 윈도우보다 자주 업데이트 됩니다. 하지만 나중에 열린 윈도우는 초당 1 번 정도로 활성화된 윈도우의 업데이트 되는 속도보다 느립니다.
- 일단 모니터링이 되면 확실하게 꺼질 때까지 계속됩니다.

5. 모니터링을 끝내려면, View 메뉴에서 모니터링을 선택하십시오.
6. 나타난 서브메뉴에서 Off 를 선택하십시오.

**Note:** 실시간 업데이트 되는 상황을 보기 위해 로직은 실행되어야 합니다. 만약 실행중인 로직이 멈춘다면 마지막 값은 모니터링이 멈출 때까지 디스플레이 됩니다.

# 모니터된 로직의 Display Format 변경하기

1. 모니터하고자 하는 실행 로직을 지닌 PLC 를 연결하십시오.
2. PLC 상에 있는 폴더와 같은 현재 폴더에서 로직을 확인하십시오.
3. IL 이나 LD Editor 에서 변수를 선택하십시오.
4. Context-Sensitive 나 View 메뉴에서 Display Format 을 선택하십시오.
5. 나타난 서브메뉴에서 데이터를 디스플레이하려는 포맷을 선택하십시오. 다음 사항을 선택할 수 있습니다:

   Octal
   Binary
   Hex
   Real (Only available for the Real data type)
   Scientific Notation (Only available for the Real data type)
   Signed Decimal
   Unsigned Decimal

   선택한 디스플레이 포맷은 선택된 변수에 적용됩니다.

# Update in progress or unavailable - Press F1 for Help

"* Update in progress or unavailable – Press F1 for Help *"메시지는 VersaPro 가 PLC 로부터 실시간 데이터를 기다리는 동안 Window Title Bar 에 설치됩니다. 이 메시지는 Reference View Table, Logic 과 Variable View Table Windows 에 나타날 수 있습니다. 이 메시지는 RTU(Real Time Updates)가 활동하는 동안 정상 운영하에서는 순간적으로 나타날 것입니다. 만약 이 메시지가 Window Title Bar 에 장시간 남아 있다면, 이는 PLC 에서 지원하는 수보다 더 많은 창을 띄웠다는 것이다. 이러한 상황에서 View 메뉴나 툴 바를 이용해 "Monitor Active" 모드에 VersaPro 를 설치하든지 아니면 열려 있는 윈도우의 수를 감소시켜야 합니다. 모니터링할 수 있는 오픈된 윈도우의 최대 수는 다음과 같이 PLC 모델에 따라 다양합니다:

**지원되는 윈도우와 애플리케이션의 최대 수**

| Target | Windows | Applications* |
|---|---|---|
| VersaMax Nano | 2 | 2 |
| VersaMax Micro | 2 | 4 |
| VersaMax modular | 16 | N/A |
| Series 90-30 (CPU350-364) | 16 | N/A |
| Series 90-30 (CPU311-341) | 32 | N/A |
| Series 90 Micro | 16 | N/A |

* Example: VersaPro 와 Fault Table 은 두 애플리케이션으로 카운트됩니다.

# Override 탐색하기

VersaPro 폴더가 PLC 에 연결되면, 오버라이드를 지닌 변수에 대한 PLC 로직을 탐색할 수 있습니다. 이 특성을 이용하려면 PLC 메뉴에서 Search for Overrides 를 선택하십시오. 오버라이드된 변수는 Information Window 에 등록됩니다.

Chapter 11

# Fault Table로 작업하기

PLC 와 I/O Fault Table 은 CPU 나 PLC 에 있는 모듈에 의해 로그되는 오류 정보를 디스플레이합니다. 이 정보는 PLC CPU 에서 운용되는 PLC 하드웨어나 소프트웨어와 함께 문제가 있는지를 결정하는데 사용됩니다. Fault system 을 이용해 오류를 보고, 소트하고, 인쇄하고 지울 있습니다.

**Note:** Fault Table 시스템은 메뉴와 툴바를 지닌 VersaPro 와는 다른 애플리케이션입니다. Fault Table 이 최소화되면 Windows taskbar 에 있는 버튼은 줄어들고 Equipment Workbench Window 에는 나타나지 않습니다.

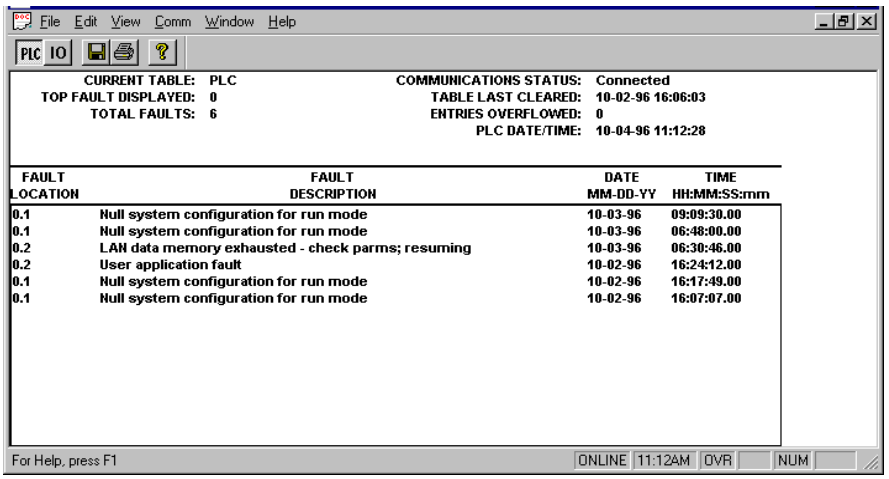
## 목차

# PLC 와 I/O Fault Table 로 작업하기

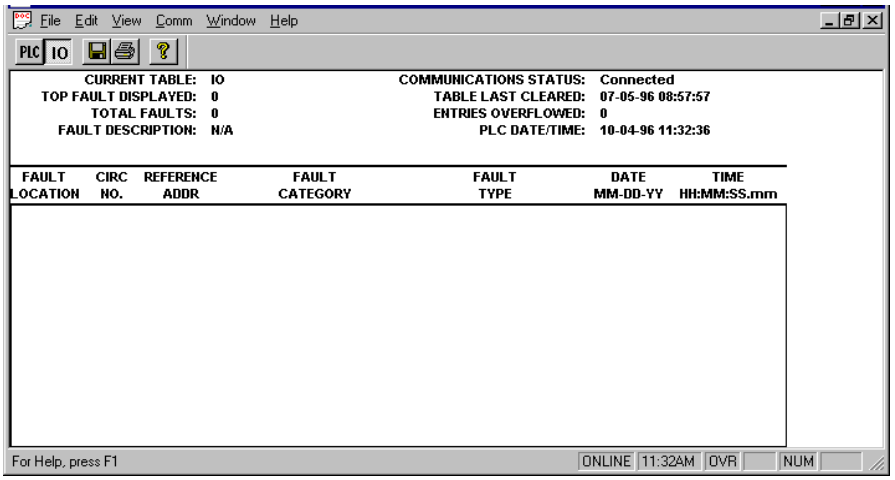
PLC 와 I/O Fault Table 을 보려면, PC 가 PLC 와 연결해야 하거나 온라인이 되어야 합니다. Fault Table 을 열기 전에 PLC 와 연결할 수 있습니다. 만약 VersaPro 가 PLC 와 연결을 안하고 Fault Table Window 을 열려면 PLC 나 IO button 을 클릭하면 Connect dialog box 가 나타날 것입니다(세부사항은 Chapter 10 " PLC 와 연결하기"에 있습니다.).

# PLC 와 I/O Fault Tables 보기

VersaPro 에서 Fault Table 을 보려면 Tools 메뉴로 가서 Fault Table 을 선택하십시오. 만약 사용자가 PLC Fault Table 에 있다면, I/O Fault Table 을 액세스 하기 위해 IO toolbar button 을 클릭하십시오:



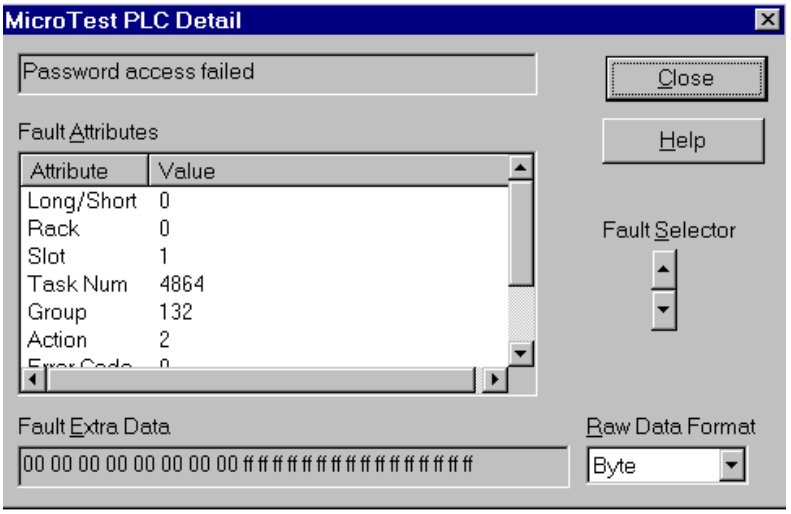
VersaPro 에서 I/O Fault Table 을 보기 위해 IO toolbar button 을 클릭하십시오:




# Expanded Fault Information 얻기

Fault Table 은 각 오류에 대한 요약된 설명을 제공합니다. 오류에 대한 추가 정보는 다음 방법 중 하나와 액세스 할 수 있습니다:

- 명시된 오류에 대한 추가 정보를 얻으려면 Fault 를 더블 클릭하십시오. Fault Detail dialog box 가 나타날 것입니다.




- 모든 오류와 확장된 오류 데이터를 리스트하려면,File 메뉴를 선택하고 Save(또는 Save As)를 선택하십시오. 일단 이 정보를 저장하면, 다음에는 저장하여 생성된 텍스트 파일을 보거나 인쇄하기 위해 텍스트 에디터를 사용할 수 있습니다.

# 오류 소트하기

오류는 로그된(날짜와 시간) 순서에 따라 디스플레이됩니다. 그러나 오류 위치에 따라 순서를 정할 수 있고 오류 내용에 따라 그룹화하도록 오류를 선택할 수 있습니다.

디스플레이의 순서를 변경시키려면, View 메뉴를 선택하여 Sort 를 선택하고 소트된 디스플레이에 대한 키 필드를 선택하십시오.

PLC Fault Table 에 있어서, 소팅 선택은 다음과 같습니다:

- 위치
- 설명
- 날짜/시간

I/O Fault Table 에 있어서, 소팅 선택은 다음과 같습니다:

- 위치
- 회로 번호
- 레퍼런스 어드레스
- 카테고리
- 타입
- 날짜/시간

**Note:** 원 데이터 순서에 리스트를 저장하려면 View 메뉴를 선택하고 Sort Ascending 을 선택하십시오.

# 오류 정보 저장하기

문제를 도큐먼트하고 트러블슈트하기 위해 텍스트 파일로 오류 정보를 저장할 수 있습니다. 파일로 저장할 때, 오류 테이블 등록과 오류의 세부 사항은 파일로 저장됩니다.

오류 데이터를 저장하려면 다음 단계를 수행하십시오:

1. PLC 나 I/O Fault Table 을 선택하십시오.
2. File 메뉴를 선택하고 Save 를 선택하거나 Fault Table Window 에 있는 Save button 을 클릭하십시오.
3. Save As dialog box 에 오류 데이터의 파일 이름을 입력하십시오(만약 오류 정보가 이미 저장되어 있다면 새로운 데이터는 동일한 이름으로 저장될 것입니다.)

# Fault Table 지우기

F9 을 누르거나 Edit 메뉴에서 Clear Table 을 선택하여 모든 오류 테이블을 지울 수 있습니다. 또한 연결된 로그인된 모든 Genius blocks 에서 관련된 오류 표시를 지웁니다. 만약 오류 테이블을 지우기 위해 Clear Table 을 선택했다면 PLC 에 있는 오류 비트는 리셋되고 RUN 는 다시 작동됩니다.

**Caution**

**오류 테이블을 지우는 것은 오류를 제거하는 것입니다. 오류가 발생되는 중요한 문제는 고정되어 있지 않습니다. 오류 조건이 존재한다 하더라도 오류 테이블을 지우면 PLC 가 Run 모드에 설치할 수 있도록 할 것입니다. 만약 오류를 발생시키는 상황이 계속 존재한다면 오류는 구성을 저장하고, PLC 로 전원이 순환한 후에 Stop 모드에서 Run 모드로 전환하는 동안 다시 리포트될 수 있습니다.**

**Note:** 오류 테이블을 지우면 블록에 있는 오류 조건을 지움과 동시에 I/O 오류 테이블이 지워집니다. 오류 테이블이 지워진 후에도 오류가 지속된다면 Genius Circuit Fault 와는 다른 I/O 오류는 로그되지 않습니다. 오류 테이블을 지운 후에 치명적으로 구성된 활성화된 오류 조건과 함께 PLC 는 RUN 모드에 설치될 수 있습니다. 이러한 조건을 피하기 위해 프로그램을 저장하고 PLC 를 구성하기 전에 항상 오류 테이블을 지우십시오.

# 다른 오류 테이블 열기

통신 메뉴에서 Communications Setup Option 을 이용해 포트/디바이스 할당이 완성된다면 다른 PLC 에서 오류 테이블을 볼 수 있습니다.

다른 오류 테이블을 열려면 다음 단계를 수행하십시오:

1. File 메뉴에서 New I/O Table 이나 New PLC Table 을 선택하십시오. Connect dialog box 가 나타날 것입니다.
2. 열려고 하는 오류 테이블을 포함하고 있는 PLC 와 일치하는 Device Name 을 선택하십시오
3. PLC 에 링크되는 통신과 일치하는 포트를 선택하십시오.
4. OK button 을 누르십시오.

# 폰트 변경하기

Edit 메뉴를 선택하고 Set Font 를 선택함으로써 Fault Table 의 폰트를 변경할 수 있습니다. 폰트 타입, 스타일 및 크기를 변경할 수 있는 Font dialog box 가 나타날 것입니다.

**Note:** 폰트 변경은 오류 테이블에만 영향을 주고 VersaPro 의 다른 영역에는 영향을 주지 않습니다.

# Fault Table 인쇄하기

활성화된 Fault Table 을 인쇄하려면, File 메뉴를 선택하고 Print 를 선택하거나 툴바에서 Print button 을 클릭하십시오.

부록
A

# VersaPro 메뉴와 툴바

이 부록은 VersaPro 프로그래밍 소프트웨어와 Stand Alone Hardware Configuration Tool 에서 사용되는 메뉴 바와 툴바를 나타낸 것입니다.

## VersaPro 메뉴

## 파일 메뉴

파일 메뉴는 VersaPro 콤포넌트용 표준 파일과 인쇄 기능을 포함하고 있습니다.

| Option | Button | Action |
|---|---|---|
| New Block (Ctrl + B) |  | 새로운 블록 생성하기. |
| New Motion | | 새로운 로컬 로직이나 모션 프로그램 생성하기. |
| New Reference View Table | | 새로운 Reference View Table 생성하기. |
| New Variable View Table | | 새로운 Variable View Table 생성하기. |
| Save (Ctrl + S) |  | 선택된 항목 저장하기. |
| Save All (Shift + Ctrl +A) |  | 모든 폴더 저장하기. |
| Close (Ctrl + F4) | | 선택된 윈도우 닫기 |
| New Folder (Ctrl + N) |  | 새로운 폴더 생성하거나 기존 폴더 불러오기. |
| Open Folder (Ctrl + O) |  | 기존 폴더 열기. |
| Close Folder | | 열린 폴더 닫기. |
| Print |  | 선택된 항목 인쇄하기. |
| Print Report | | 콤포넌트와 블록은 리포트에 포함되어야 하고 다음에 리포트를 인쇄해야 한다는 것을 정의 내릴 수 있도록 함 |
| Page Setup | | 페이지 파라미터를 명시할 수 있도록 함. |
| <MRU list> | | 메뉴 상에 나타나는 Most Recently Used folders 리스트(MRU list)에서 최근 사용된 폴더를 열 수 있도록 함. |
| Exit (Alt + F4) | | VersaPro 소프트웨어 닫기 |

# 편집 메뉴

| Option | Button | Action |
|---|---|---|
| Undo <action> (Ctrl + Z ) |  | 이전동작으로 돌아가기(예를 들면 Drop 이나 Delete). "<action>"대신에 나타남. |
| Redo <action> (Ctrl + Y) |  | 실행취소 이전으로 돌아가기(예를 들면 Drop 이나 Delete). "<action>"대신에 나타남. |
| Cut (Ctrl + X) |  | 선택된 항목을 제거하고 클립보드에 넣기. |
| Copy (Ctrl + C) |  | 선택된 항목을 클립보드에 복사하기. |
| Copy As | | 비트맵이나 텍스트같은 요소를 클립보드에 복사하기. |
| Paste (Ctrl + V) |  | 선택된 영역으로 클립보드 컨텐츠를 붙여넣기. |
| Delete (Del) |  | 선택된 항목 삭제하기. |
| Select All (Ctrl + A) | | 모든 항목 선택하기 |
| Find/Replace (Ctrl + F3) | | 변수 이름이나 어드레스, 서브루틴으로 부르기나 로직에 있는 Jump/Label 이나 MCR/END_MCR 를 찾아 바꾸기 |
| Find Next (F3) | | 정의된 탐색기준으로 다음 항목 찾기. |
| Find In Blocks (Ctrl + F10) | | 변수 이름이나 어드레스, 또는 서브루틴으로 부르기로 특수 글로벌이나 시스템 변수 찾기. |
| Go to (Ctrl + G) | | IL 로직에 있는 줄이나 LD 로직에 있는 Rung으로 가기. |
| Go to Variable (Ctrl + F11) | | VDT 에 있는 변수로 가기. |
| Go to Next Coil (Alt + F3) | | LD Editor 에서 선택된 레퍼런스와 같은 이름으로 다음 코일로 커서를 이동하기. |
| Properties (Alt + Enter) | | 현 블록에서 설명 추가하기. 현 폴더로 설명을 추가하기 또는 임시변수나 변환변수의 속성을 설정하기. |

# 보기 메뉴

| Option | Button | Action |
|---|---|---|
| Toolbars (Ctrl + T) | | 툴바 보기와 숨기기. |
| Function Toolbars | | 기능 툴바를 확장시키거나 압축하기. |
| Status Bar (Alt + O) | | 상태바를 보거나 숨기기. |
| Folder Browser (Alt + 1) |  | Folder Browser window 를 열거나 닫기. |
| Information Window (Alt + 2) |  | Information window 를 열거나 닫기. |
| Variable Declaration Table (Alt + 3) |  | Variable Declaration Table Window 를 열거나 닫기. |
| Hardware Configuration (Alt + F4) |  | Hardware Configuration window 열기. |
| MAIN block (Alt + F5) |  | _MAIN block 열기. |
| Local Logic Variable Table (Alt + 6) |  | 폴더가 Local Logic 프로그램을 포함하고 있다면 Local Logic Variable Table 을 열거나 닫기. |
| Sort | | 내림차순이나 오름차순으로 VDT 에 명시된 컬럼 소트하기. |
| Monitor |  | 모든 윈도우나 활성화된 윈도우는 PLC 에 대해 실시간 로직 실행 가능 여부 허용. |
| Display Format | | 모니터된 로직의 디스플레이 포맷 변경. |
| Format View Table | | 선택된 Variable View Table 열, Reference View Table 열 또는 모든 Reference View Table 에서 데이터가 어떻게 디스플레이 되는지 결정. |
| Zoom In (Ctrl + "=") |  | 보기 확대하기. |
| Zoom Out (Ctrl + "-") |  | 보기 축소하기. |

# 삽입 메뉴

| Option | Button | Action |
|---|---|---|
| Object | | Folder Browser 에 객체 삽입하기. |
| Block | | Folder Browser 에 객체 삽입하기. |
| Row (Ins) | | IL editor, LD editor, VDT 또는 View Table 에 선택된 줄 위에 삽입하기. |
| Comment |  | 코멘트 삽입하기. |

# 폴더 메뉴

| Option | Button | Action |
|---|---|---|
| Check Selected Block(s) (Ctrl + F7) |  | Syntactic 의 정확성을 위해 선택된 블록 점검하기. 만약 블록이 선정되지 않았다면 이 메뉴는 “Check Block_Main”을 읽고 Syntactic 의 정확성을 위해 _Main block 을 점검한 결과를 선택. |
| Check All (F7) |  | Syntactic 의 정확성을 위해 모든 블록 점검하기. |
| Lock/Unlock | | 폴더 열기를 잠거나 풀기. |
| Backup (F8) | | 폴더 백업하기. |
| Restore (Ctrl + F8) | | 폴더 복구하기. |
| Restore Motion Blocks | | Load 동작이 이루어지는 동안 백업 된 개별 모션 블록 복구. |
| Find Unused Variables | | 사용하지 않은 변수 찾기. 사용하는 변수는 Name, Scope 과 Address 에 의해 Information window 에 등록됨. |
| Compact | | VDT 에서 편집 이력 정보를 제거함으로써 폴더 크기 줄이기. |

# PLC 메뉴

| Option | Button | Action |
|---|---|---|
| Connect (F9) |  | PLC 에 연결. |
| Disconnect (Ctrl + F9) |  | PLC 로부터 분리. |
| Store (Alt + F2) |  | PLC 에 폴더의 컨텐츠 저장하기. |
| Load (Ctrl + F2) |  | PLC 로부터 폴더의 컨텐츠 로드하기. |
| Verify (F4) |  | 연결된 PLC 와 현 폴더가 동일한지 확인. |
| Clear | | PLC 메모리 지우기. |
| Flash/EEPROM | | Flash 메모리 읽기/쓰기/확인하기. |
| Search for Overrides | | VersaPro 가 PLC 에 연결될 때, 변수가 오버라이드 되는지 PLC 로직 찾기. |
| Run (F5) |  | PLC 에 있는 로직 작동하기. |
| Stop (Ctrl + F5) |  | PLC 에 있는 로직 작동 멈춤. |
| Toggle (F12) |  | 레퍼런스 toggle 하기. |
| Override (F11) |  | 레퍼런스 오버라이드하기. |
| Write Reference Value (F10) |  | LD editor, IL editor, VVT 또는 RVT 에 있는 선택된 레퍼런스로 값 쓰기. |
| Tuning Parameters | | PID 명령어 조절하기. |
| Status Info (Shift + F4) |  | PLC 상태 보기. |
| Abort! (Ctrl + Break) |  | 통신 멈추기. |

# 도구 메뉴

| Option | Action |
|---|---|
| Fault table | 오류 테이블 열기. |
| Communications Setup | 통신 파라미터 정의하기. |
| Import Variables | 변수 불러오기. |
| Export Variables | 변수 내보내기. |
| Convert Block | 한 언어에서 다른 언어로 로직 변환하기. |
| View Online Cross Reference | 정보창에서 레퍼런스 타입에 의해 선택된 블록에서 사용된 모든 변수 목록 디스플레이하기. |
| Non-Nested to Nested Conversion | Non-nested MCR, ENDMCR, JUMP 와 LABEL 명령어를 Nested Version 으로 변환하기. |
| Options (Ctrl + E) | General(관련 언어), Display(컬러, 폰트), Ladder(필드 셀 폭 보기)와 자동 연결 옵션 설정하기. |

# 창 메뉴

| Option | Action |
|---|---|
| Cascade (Alt + 6) | 모든 타이틀 바를 보고 활성화된 윈도우를 앞에 놓이기 하기 위해 열려있는 모든 블록 정렬하기. |
| Tile Horizontally (Alt + F7) | 열려 있는 모든 윈도우를 보기 위해 열려 있는 모든 윈도우의 크기를 조절하고 수평으로 정렬. |
| Tile Vertically (Alt + F8) | 열려 있는 모든 윈도우를 보기 위해 열려 있는 모든 윈도우의 크기를 조절하고 수직으로 정렬. |
| Arrange Icons (Alt + F9) | 최소화된 모든 블록 윈도우 정렬시키기. |
| Close All (Alt + F10) | 열려 있는 모든 블록 윈도우 닫기. |
| Next Window (F6) | 활성화된 편집 윈도우와 열려 있는 dockable windows 가운데 포커스 이동하기(편집 윈도우에서 선택된 열과 셀로 가기). |
| Previous Window (Shift + F6) | 활성화된 편집 윈도우와 열려 있는 dockable windows 가운데 포커스 이동하기(편집 윈도우에서 선택된 열과 셀로 가기). |
| [Current Window] | 활성화된 윈도우 이름 디스플레이하기. |

# 도움말 메뉴

| Option | Action |
|---|---|
| Contents and Index | VersaPro Help 디스플레이하기. |
| About VersaPro™ | VersaPro 버전 사용 가능한 메모리 및 디스크 공간 디스플레이하기. |

# *VersaPro 툴바*

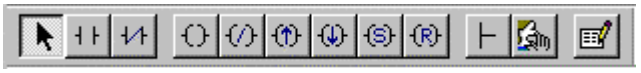
VersaPro 는 다음과 같이 6 개의 툴바를 가지고 있습니다: 표준 툴바, 래더 툴바, 보기 툴바, 폴더 툴바, PLC 툴바 및 기능 툴바. 작업 공간을 넓히기 위해 특정 또는 모든 툴바를 숨길 수 있습니다. 사용하지 않는 툴바 버튼은 희미한 회색으로 보입니다.

## 표준 툴바

표준 툴바를 숨긴다 하더라도 이 툴바는 기본 값으로 디스플레이 됩니다. 표준 툴바는 폴더 열기, 폴더 및 블록 저장하기, 자르기, 복사하기, 붙여넣기 및 실행 반복 등과 같은 일반 기능을 제공합니다.

| Button | Function |
|---|---|
|  | 새로운 폴더 생성하기. |
|  | 새로우 블록 생성하기. |
|  | 기존 폴더 열기. |
|  | 선택된 블록 저장하기. |
|  | 모든 블록 저장하기. |
|  | 선택된 항목 제거하고 클립보드에 넣기. |
|  | 선택된 항목 복하고 클립보드에 넣기. |
|  | 선택된 영역으로 클립보드 컨텐츠를 붙여넣기. |
|  | 이전 실행으로 돌아가기. |
|  | 실행취소 이전으로 돌아가기. |
|  | 선택된 항목 삭제하기. |
|  | 프린터로 선택된 항목 보내기. |
|  | Context-Sensitive Help 기능 디스플레이하기. |
|  | 통신 중지. |

# 래더 툴바



"click and drop"을 이용해 래더 로직을 생성하기 위해 래더 툴바를 사용하십시오

| Button | Function |
|---|---|
| | Normal Pointer 선택; "click and drop" 기능 이용 안함. |
| | NO 접점 드롭하기. |
| | NC 접점 드롭하기. |
| | NO 접점 코일 드롭하기. |
| | NC 접점 코일 드롭하기. |
| | Positive Transition Coil 드롭하기 |
| | Negative Transition Coil 드롭하기 |
| | 세트 코일 드롭하기. |
| | 리셋 코일 드롭하기. |
| | 수직선이나 수평선 드롭하기. |
| | CALL 명령어 드롭하기. |
| | 코멘트 열 삽입하기. |

**Note:** 드롭 모드를 사용할 수 없으면 ESC 키를 누르고 포인터를 만든 다음 다시 활성화하십시오

# 보기 툴바

보기 툴바는 윈도우를 보거나 숨길 때, _MAIN block 을 열 때나 윈도우의 컨텐츠를 확대하거나 축소할 때 사용하십시오. 현재 작업하고 있는 업무와 관련되지 않은 버튼은 사용할 수 없습니다.

| Button | Function |
|---|---|
|  | Folder Browser 를 보거나 숨기기. |
|  | Variable Declaration Table 을 보거나 숨기기. |
|  | Information Window 를 보거나 숨기기. |
|  | Hardware Configuration window 열기 |
|  | _MAIN block 열기 |
|  | Local Logic Variable Table 열기 |
|  | 모든 윈도우를 모니터하기. |
|  | 활성화된 윈도우를 모니터하기. |
|  | 모니터링 끄기. |
|  | 활성화된 윈도우 컨텐츠 확대하기. |

# 폴더 툴바

Syntax를 점검하기 위해 폴더 툴바를 사용하십시오.

| Button | Function |
|---|---|
|  | Syntax가 정확한지 모든 블록 점검. |
|  | Syntax가 정확한지 선택된 블록 점검. |

# PLC 툴바

PLC를 연결하거나 로직 저장하기, 로딩하기 또는 실행하기와 멈추기와 같은 연결된 PLC를 수반하는 동작을 수행하기 위해 PLC 툴바를 사용하십시오.

| Button | Function |
|---|---|
|  | PLC 연결하기. |
|  | PLC 분리하기. |
|  | PLC에 폴더 저장하기. |
|  | PLC로부터 폴더 로드하기. |
|  | PLC 상에 나타난 요소와 현재 폴더간의 동일한지 확인하기. |
|  | PLC 실행하기. |
|  | PLC 정지하기. |
|  | 레퍼런스 toggle하기. |
|  | 레퍼런스 오버라이드하기. |
|  | 레퍼런스로 값 쓰기. |
|  | PLC 상태 보기. |

# 기능 툴바

LD Editor 에서 프로그래밍하는 동안 펑션 블록 타입을 선택하기 위해 펑션 툴바를 사용하십시오. 펑션 그룹은 알파벳순으로 열거되어 있습니다.

펑션 블록은 확대 모드나 축소 모드에서 디스플레이할 수 있도록 구성시킬 수 있습니다. 모드를 변경하려면 View 메뉴를 선택하고 펑션 툴바를 선택한 다음 Expanded 이나 Compact 를 선택하십시오.

A

# *HWC 메뉴*

## 파일 메뉴

이 파일 메뉴는 다음과 같은 명령어를 제공합니다:

| Option | Action |
|---|---|
| New (Ctrl + N) | 새로운 Hardware Configuration Folder 생성하기. |
| Open (Ctrl + O) | 기존 Hardware Configuration Folder 열기. |
| Close | 열려 있는 Hardware Configuration Folder 닫기. |
| Import Stand-Alone HWC File | Rack System Edit window 로 HWC Hardware Configuration Folder(.hwcfg) 불러오기. |
| Import Hardware Configuration Folder | HWC Hardware Folder 로 LM90 이나 CC90 에 의해 만들어진 IOCFG.CFG 파일(그리고 옵션 CPUCFG.CFG 파일) 불러오기. |
| Save (Ctrl + S) | 같은 파일 이름을 사용해 열려 있는 도큐먼트 저장하기. |
| Save As | 특정 파일 이름으로 열려 있는 도큐먼트 저장하기. |
| Convert To | 현재 Rack System 구성을 선택된 타입의 기본 Rack System 구성으로 변환하기:<br>Series 9030<br>VersaMax<br>VersaMax Nano/Micro<br>Series 90 Micro<br>(다른 타입의 Rack System 으로만 Rack System 구성을 변환시킬 수 있습니다. 만약 현재 Rack System 과 같은 Rack 구성을 생성하려면 Save As 를 사용하십시오.) |
| Print (Ctrl + P) | Rack System Hardware Configuration 의 보든 부분이나 일부분 인쇄하기. |
| Print Preview | 인쇄하고자 하는 Rack System Hardware Configuration 부분을 스크린에 디스플레이하기. |
| Print Setup | 프린터나 프린터 연결 선택하기. |
| Exit | HWC 나오기. |

# 편집 메뉴

| Option | Action |
|---|---|
| Undo (Ctrl + Z) | 이전 모듈 실행으로 돌아가기. 16 레벨까지 Undo 지원됨. |
| Redo (Ctrl + Y) | 샐행취소(Undo) 전으로 돌아가기. 16 레벨까지 Redo 지원됨. |
| Cut (Ctrl + X) | Rack System 을 선택된 모듈을 삭제하고 클립보드로 이동하기. |
| Delete (Del) | Rack System 을 선택된 모듈을 삭제하고 빈 슬롯으로 되돌아가기. |
| Copy (Ctrl + C) | Rack System 을 선택된 모듈을 클립보드로 복사하기. |
| Paste (Ctrl + V) | 클립보드에서 이전에 자르기 했거나 복사했던 모듈을 선택된 Rack System 에 붙여넣기. |
| Expansion Rack System | VersaMax Rack System 에 명시하는 편집 실행의 서브메뉴. |
| Expansion Receiver | VersaMax Expansion Rack System 에 명시하는 편집 실행의 서브메뉴. |
| Rack Operations | Rack System 에 명시하는 편집 실행의 서브메뉴. |
| Module Operations | 선택된 Module 에 명시하는 편집 실행의 서브메뉴. |

## 편집 메뉴, 확장 랙 시스템 서브 메뉴

편집메뉴의 Expansion Rack System sub-menu 는 VersaMax 시스템 타입을 선택할 수 있게 해줍니다. 하나의 VersaMax 시스템 타입에서 다른 VersaMax 시스템 타입까지 연결시킬 수 있지만, 구성시 나타난 변경사항은 실행취소를 할 수 없습니다.

| Option | Action |
|---|---|
| None | 이 옵션은 파일 메뉴에서 Nwe 를 선택했을 때 생성되는 기본 VersaMax 시스템 입니다. 이는 Main VersaMax I/O 스테이션만을 포함합니다(Expansion 안됨). |
| Local Single Rack | Main VersaMax I/O 스테이션과 하나의 Expansion I/O Station(Rack)을 포함하지만 Transmitter 를 포함하고 있지 않습니다. Expansion I/O Station 은 Non Isolated Receiver 와 Power Supply Integrated Unit(IC200ERM002)을 포함하고 있습니다. |
| Multiple Remote Rack | 7 개까지의 Expansion I/O Station 을 구성할 수 있습니다(7 개 Rack tab 은 항상 나타나 있음). Main VersaMax 랙은 Transmitter(IC200ETM001)를 포함하고 있습니다. 각각의 Expansion I/O Station 은 Power Supply 와 Receiver Integrated Unit 을 포함하고 있습니다. 기본 값으로 구성된 Non Isolated Receiver (IC200ERM002)는, Isolated Receiver(IC200ERM001)으로 대체시킬 수 있습니다. |

**Note:** 트랜스미터 모듈은 모듈 카탈로그에서 선택할 수 없습니다. 랙 구성에 이 모듈을 선택할 수 있는 유일한 방법은 적절한 구성을 선택하는 것입니다. 또한 랙 구성 타입을 변경하는 것 이외에 랙 구성에서 트랜스미터 모듈을 삭제시킬 수 없습니다.

## 편집 메뉴, 확장 리시버 서브 메뉴

편집 메뉴의 확장 리시버 서브 메뉴는 VersaMax Local Single Rack 나 Multiple Remote Rack 시스템용 리시버를 선택할 수 있게 해줍니다.

| Option | Action |
|---|---|
| None | Receiver 를 비활성화시킵니다. 이것은 Carrier/Base 가 구성되어 있지 않더라도 기본 선택입니다. 이 옵션은 확장 랙에 최소 하나의 Carrier/Base 가 구성되어 있다면 사용할 수 없습니다. |
| Non-Isolated Receiver Unit | Non Isolated Receiver 와 Power Supply Integrated Unit(IC200ERM002)을 선택합니다. 이는 Local Single Rack 시스템에 허용되는 유일한 리시버 타입입니다. |
| Isolated Receiver Unit | Non Isolated Receiver(IC200ERM002)를 선택합니다. 이 옵션은 Multiple Remote Rack 시스템에서만 사용할 수 있습니다. |

## 편집 메뉴, 랙 운용 서브 메뉴

| Option | Action |
|---|---|
| Change Rack Type | 90-30 Change Rack Type Dialog 를 디스플레이합니다. 이 메뉴는 VersaMax 시스템용으로 사용할 수 없습니다. |
| Clear Rack | 선택된 Rack 에서 Power Supply 와 CPU 를 포함해 모든 모듈을 삭제합니다. |
| Configure CPU Parameters | Rack System CPU 용 구성 파라미터를 편집할 수 있는 Module Parameter Dialog 를 디스플레이합니다. |
| Replace CPU | Rack System CPU 을 대체할 수 있도록 Module Selection Dialog 를 디스플레이 합니다. |
| Name Resolution and Routing | Ethernet Adapter Naming and Routing Tables 의 사양과 관계된 90-30 Ethernet Adapter Name Resolution Dialog 를 디스플레이합니다. 이 메뉴는 VersaMax 시스템용으로 사용할 수 없습니다. |
| Ethernet Global Data | Ethernet Exchange Id 를 고려하는 90-30 Ethernet Global Data Dialog 를 디스플레이합니다. 이 메뉴는 VersaMax 시스템용으로 사용할 수 없습니다. |
| Add Base/Carrier | VersaMax Add Base/Carrier Dialog 를 디스플레이하고 선택된 Base/Carrier 를 지닌 선택된 랙을 확장합니다. 이 메뉴는 90-30 시스템용으로 사용할 수 없습니다. |
| Delete Base/Carrier | VersaMax rack 에 있는 선택된 Base/Carrier 를 삭제합니다. 이 90-30 시스템용으로 사용할 수 없습니다. |
| Insert Base/Carrier | VersaMax Insert Base/Carrier Dialog 를 디스플레이하고, VersaMax 랙의 선택된 슬롯 앞에 선택된 Base/Carrier 를 삽입합니다 이 메뉴는 90-30 시스템용으로 사용할 수 없습니다. |
| Replace Base/Carrier | VersaMax Replace Base/Carrier Dialog 를 디스플레이하고, VersaMax 랙의 선택된 슬롯에 선택된 Base/Carrier 를 대체합니다. 이 메뉴는 90-30 시스템용으로 사용할 수 없습니다. |
| Configure Receiver Parameters | 구성 파라미터가 없는 VersaMax 확장 랙 리시버용 전력소모 탭을 디스플레이합니다. 이 메뉴는 리시버가 선택된 확장 랙의 첫 슬롯에 위치할 때에만 사용할 수 있습니다. |

## 편집 메뉴, 모듈 운용 서브 메뉴

| Option | Action |
|---|---|
| Configure Parameters (Ctrl + E) | 선택된 모듈용 구성 파라미터를 편집할 수 있는 Module Parameter Dialog를 디스플레이합니다. |
| Add Module | 선택된 빈 랙 슬롯에 새 모듈의 선택과 구성을 할 수 있는 Module Selection Dialog을 디스플레이합니다(또는 VersaMax 모듈러 시스템에 있는 Carrier/Base). |
| Replace Module | 선택된 빈 랙 슬롯에 대체 모듈의 선택과 구성을 할 수 있는 Module Selection Dialog을 디스플레이합니다. |
| Delete Module | Rack 시스템에서 선택된 Module 을 삭제합니다. |

# 파라미터 메뉴

| Option | Action |
|---|---|
| Data Entry Tool (F2) | 파라미터를 선택하여 편집할 수 있도록 Data Entry Tool 을 엽니다. |
| Reset Parameter | Start of Edit Session 값이나 Factory Original Value에 단일 파라미터를 리셋합니다. |
| Reset Tab | Start of Edit Session 값이나 Factory Original Value 을 위해 탭에 모든 파라미터를 리셋합니다. |
| Cancel Edit Session | 변경사항을 저장하지않고 활성화된 파라미터 편집 윈도우를 닫습니다. |
| Insert | Profibus Slave Information tab 과 같은 어떤 탭에 선택된 열 위에 기본 값을 지닌 열을 삽입합니다. |
| Delete | Profibus Slave Information tab 과 같은 어떤 탭에 선택된 열을 지우고 매트릭스 끝에 기본 열을 추가합니다. |

# 보기 메뉴

| Option | Action |
|---|---|
| Toolbar | 툴바를 보거나 숨기기. |
| Status Bar | 상태 바를 보거나 숨기기 . |
| Log View | 온라인 HWC Log View 를 보거나 숨기기. |
| Reference View | 온라인 Rack System Reference View 를 보거나 숨기기. |
| Power Consumption | Rack System Power Consumption View 에 관련된 옵션을 포함하는 서브메뉴를 디스플레이합니다. 이 항목은 Rack System Edit Window 가 열려 있을 동안만 볼 수 있다는 것에 주의하십시오. |
| Parameter Edit | Parameter Editor 디스플레이 포맷을 변경할 수 있는 서브메뉴를 디스플레이합니다. 이들 옵션은 모듈을 위한 Parameter Editor window 가 열릴 때에만 사용할 수 있습니다. |

## 보기 메뉴, 전력 소모 서브 메뉴

| Option | Action |
|---|---|
| Power Display Units | Power Consumption View 의 단위를 Amp 나 Watt 로 변경시킬 수 있는 서브메뉴. |
| Power Display Colors | Power Consumption View 의 백그라운드 색을 어둠, 중간 또는 밝음으로 변경할 수 있는 서브 메뉴. 일단 색상을 선택하면 Power Consumption View 와 Dialog는 다른 색상을 선택할 때까지 백그라운드에 디스플레이될 것입니다. |
| Power Consumption View | 온라인 Rack System Power View 를 보거나 숨기기. |
| Power Consumption Dialog | Power Consumption Dialog. 보기 |

## 보기 메뉴, 파라미터 편집 서브 메뉴

**Note:** 이 설정치는 Parameter Editor 를 닫을 때 저장되지 않습니다. Parameter Editor 의 기본 디스플레이 모드를 설정하려면, Tools 메뉴로 가서 Options 을 선택하여 Options dialog box 에 있는 Parameter Edit tab 을 선택하십시오.

| Option | Action |
|---|---|
| Singlerow Tab | 탭 포맷의 Parameter Editor 를 표시합니다. 모든 탭을 표시하기에 창이 좁아 크기 변경을 원한다면, 안 보이는 탭을 사용하기위해 스크롤 바를 사용하시오 |
| Multirow Tab | 탭된 포맷에서 Parameter Editor window 를 디스플레이합니다. 만약 윈도우가 너무 좁아 단일 행에서 볼 수 없어 윈도우 크기를 조정하면 탭은 늘어날 것입니다. |
| Spreadsheet | 스프레드시트 포맷에서 Parameter Editor window 를 디스플레이합니다. Tab 의 표제는 볼드체와 밑줄 친 텍스트로 디스플레이됩니다. |
| Error Status | 파라미터 에러들을 인식하고 윈도우를 닫기 전에 이들이 고쳐졌는지를 확인하는 Parameter Editor Error List 가 디스플레이됩니다. 이 옵션은 현재 Parameter Editor window 가 에러를 포함하고 있을 때, 사용할 수 있습니다. |

# 도구 메뉴

| Option | Action |
|---|---|
| Hardware Configuration Data View | 텍스트와 바이너리가 포함된 정보의 프레젠테이션을 포맷하도록 하는 탭된 다이얼로그에서 PLC Configuration Data 를 디스플레이합니다. |
| Options | Log 사이즈와 Parameter Editor 보기와 같은 어떤 애플리케이션 행위를 조절할 수 있는 Options dialog 를 디스플레이합니다. |
| CPU Information | CPU Model Number 를 기입하고 현재 정의된 Reference Memory Limits 를 열거하는 CPU Information dialog 를 디스플레이합니다. 이 항목은 Rack System Edit Window 가 열려 있을 때에만 볼 수 있다는 것에 주의하십시오. |

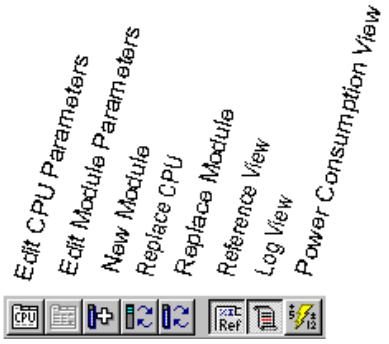
# 창 메뉴

| Option | Action |
|---|---|
| Cascade | 오버랩된 모양에서 윈도우 정렬하기. |
| Tile | 오버랩되지 않은 타일에서 윈도우 정렬하기 |
| Arrange Icons | 닫힌 윈도우의 아이콘 정렬하기. |
| Window 1, 2, . | 명시된 윈도우로 가기. |

# 도움말 메뉴

| Option | Action |
|---|---|
| Contents | 구조화된 Help Table of Contents 를 디스플레이합니다. |
| Help Index | 주제와 관련된 인덱스를 제공합니다. |
| Module Help | 선택한 Parameter Editor window 에 대한 파라미터 지향 Help 기능을 제공합니다. |
| How to Use Help | HWC 에서 사용할 수 있는 다양한 형태의 Help 를 열거하고, 프로그램을 배우고 사용하는 것을 돕기 위해 각각의 타입을 적용시킬 수 있는 방법을 제공합니다. |
| About HWC | 이 애플리케이션의 버전을 디스플레이합니다. |

# HWC 툴바




| Button | Function |
|---|---|
| | CPU 를 위한 Parameter Editor window 열기. |
| | 선택된 모듈을 위한 Parameter Editor window 열기. |
| | 구성시키고자 하는 모듈을 선택할 수 있는 Module Catalog 열기. |
| | 구성시키고자 하는 CPU 을 선택할 수 있는 Module Catalog 열기. |
| | 선택된 모듈의 위치에서 구성시키고자 하는 모듈을 선택할 수 있는 Module Catalog 열기. |
| | 랙 시스템을 구성하는 레퍼런스가 열거된 Reference View 를 디스플레이하거나 숨기기. |
| | 하드웨어 구성 편집과 관련된 이벤트와 상태 이력 정보를 디스플레이하는 Log View 를 디스플레이하거나 숨기기. |
| | 시스템에 있는 모듈에 의해 소비되는 전원을 볼 수 있는 Rack System Power Consumption dialog 디스플레이. |

Appendix
B

# 시리얼 연결

이 부록은 Series 90 과 Series 90 Protocol(SNP)용 VersaMax PLC 뿐만 아니라 Miniconverter Kit 를 연결하는데 사용되는 시리얼 포트, 컨버터 및 케이블을 기술하였습니다. 이 정보는 레퍼런스와 다양한 케이블 들을 요구하는 애플리케이션을 가진 사용자를 위한 것입니다.

B

# 시리얼 포트와 케이블

## RS-422 Interface

Series 90-30, Series 90 Micro 와 VersaMax PLC 는 EIA RS-422 사양과 호환됩니다. RS-422 드라이버와 리시버는 five twisted pairs 를 지닌 단일 케이블 상에서 여러 드라이버/리시버 결합을 이용한 여러 시스템 콤포넌트 사이에서 통신을 성취하기 위해 사용됩니다. 총 케이블 길이는 4000feet 를 초과할 수 없습니다.

8 대의 드라이브와 리시버의 멀티 드롭 시스템이 구성될 수 있습니다. 추가 드롭간의 최대 커먼 모드 전압은 –7 ~ + 7V 의 RS-422 standard 입니다. 드라이버 출력은 100ohm 에서 최소 ± 2 V 의 성능을 가지고 있어야 합니다. 드라이버 출력 임피던스는 높은 임피던스 상태에서 최소 120Kohm 이 되어야 합니다. 리시버 입력 저항은 12Kohm 이상이어야 합니다. Receiver 감도는 ± 200 mV 입니다.

**Caution**

**Common mode 전압 사양에 부합되도록 해야 합니다. 이 사양을 넘어선 Common mode 조건은 전송시 에러를 발생시키거나 콤포넌트에 손상을 입힐 수 있습니다.**

# 케이블과 커넥터 사양

케이블 어셈블리는 통신 장애 중 가장 일반적인 원인 중에 하나를 제공합니다. 최상의 퍼포먼스를 구성하기 위해 추천하는 부품과 사양에 따라 케이블을 조립하십시오.

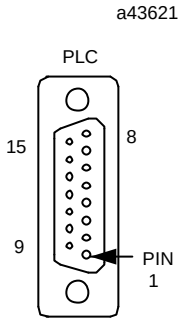
| Item | Description |
|---|---|
| Mating Connector: | **PLC: 매트릭스 하드웨어를 지닌** Serial (RS-422) port<br>Connector: 15-pin male, D-Subminiature Type, Cannon DA15S(solder pot) Hood: AMP 207470-1 connector shell<br>Hardware Kit: AMP 207871-1 Kit 2 metric screw**와** 2 screw clip **을 포함하고 있습니다**. |
| | **IBM-AT/XT:** standard RS-232 connector**를 지닌** Serial(RS-232) port<br>Connector: 9-pin female, D-Subminiature Type, DE110963-31 hood**나 동일한 성능** (standard RS-232 connector)**을 지닌** Cannon DE9S(solder pot) |
| | **RS-232/RS-485 Converter:** 15-pin male **과** 25-pin male connector<br>15-pin male connector**는 매트릭스 하드웨어**(Series 90 PLC **에 있어서는 동일한** connector, hood hardware)**를 요구합니다**.<br>25-pin male D-Subminiature Type, DE110963-3 hood**나 동일한 성능**(standard RS-232 connector)**을 지닌** Cannon DA25S (solder pot) |
| Cable: | Computer grade, 24 AWG (.22mm$^2$), **모두 차폐된 것**<br>**카탈로그 번호**: Belden 9505, Belden 9306, Belden 9832<br>**이들 케이블은 다음과 같이** 19.2 Kbps **까지의 데이터 전송률을 수용할 수 있는 운용을 제공합니다**:<br>RS-232: 50 feet(15m) **최대 길이**<br>RS-422/RS-422: 4000 feet(1200m) **최대 길이**. **최대** +7V ~ -7V**의** RS-422 Common Mode **사양을 초과해서는 안됨**. **원격용에서 절연은** Common Mode **전압을 줄이거나 제거하여 사용할 있습니다**.<br><br>50feet(15m)**이하의 짧은 거리에서**, Twisted Pair**나** Shielded Twisted Pair Cable **은 선이 직접 연결될 수 있는 만큼 사용될 것입니다**.<br><br>RS-422/RS-422 **을 사용할 때**, Twisted Pair**는 전송 신호는 한** Twisted Pair**로 구성되어 있고 수신 신호도 다른 한** Twisted Pair**로 구성되어 있기 때문에 일치해야 합니다. 만약 이를 무시할 경우에 통신 시스템에 나쁜 영향을 끼칠 것입니다**.<br><br>**통신 케이블이 옥외로 연결될 때**, **절연장치**(**어레스터**)**는 번개나 방전으로 인한 손상을 줄여 줍니다**.<br>***연결된 모든 디바이스들이 공통된 지점으로 접지되도록 해야 합니다. 이로 인한 장애는 장비에 손상을 초래합니다.*** |

# PLC Serial Port

Series 90 과 VersaMax PLC 시리얼 포트는 RS-422 와 호환됩니다. RS-232 to RS-422 컨버터는 RS-232 호환 인터페이스를 제공하는 시스템과 인터페이스해야 합니다.

PLC RS-422 시리얼 포트는 SNP 통신용 물리적 연결을 지원합니다.

아래 그림은 PLC 포트용 시리얼 방향과 커넥터 구성도를 보여 주고 있습니다 .




아래 테이블은 핀 번호와 신호 할당을 나타낸 것입니다:

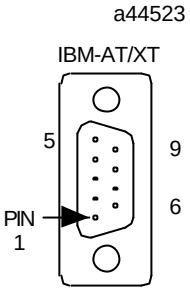
| Pin Number | Signal Name | Description |
|---|---|---|
| 1 | Shield | |
| 2 | | No Connection |
| 3 | | No Connection |
| 4 | ATCH * | Hand-Held Programmer attach signal |
| 5 | +5V * | +5V Power for: HHP and RS-232/485 Converter |
| 6 | RTS (A) | Request To Send |
| 7 | Signal Ground | Signal Ground, OV |
| 8 | CTS (B') | Clear To Send |
| 9 | RT * | Terminating Resistor for RD ** |
| 10 | RD (A') | Receive Data |
| 11 | RD (B') | Receive Data |
| 12 | SD (A) | Send Data |
| 13 | SD (B) | Send Data |
| 14 | RTS (B) | Request To Send |
| 15 | CTS (A') | Clear To Send |

* 신호는 커넥터에서 사용할 수 있지만 RS-422 사양에는 포함되어 있지 않습니다.
SD (Send Data)와 RD (Receive Data)는 TXD 와 RXD(Series Six PLC에서 사용되는)와 동일합니다.
(A)와 (B)는 –와 +로 동일합니다. A와 B는 출력을 나타낸 것이고, A' 와 B'는 입력을 나타낸 것입니다.

.

# IBM-AT/XT Serial Port

The IBM-AT, IBM-XT 또는 호환 컴퓨터의 RS-232 시리얼 포트는 아래 그림에서 나타낸 것처럼 9-pin D-type male connector입니다.




다음 테이블은 Serial Port Pins-out 을 보여주고 있습니다:

| IBM-AT Pin No. | Signal | Description |
|---|---|---|
| 1 | DCD | Data Carrier Detect |
| 2 | RD | Receive Data |
| 3 | TD | Transmit Data |
| 4 | DTR | Data Terminal Ready |
| 5 | GND | Signal Ground |
| 6 | | NC |
| 7 | RTS | Request to Send |
| 8 | CTS | Clear to Send |
| 9 | | NC |

| IBM-XT Pin No. | Signal | Description |
|---|---|---|
| 1 | | NC |
| 2 | TD | Transmit Data |
| 3 | RD | Receive Data |
| 4 | RTS | Request to Send |
| 5 | CTS | Clear to Send |
| 6 | | NC |
| 7 | GND | Signal Ground |
| 8 | DCD | Data Carrier Detect |
| 9 | DTR | Data Terminal Ready |

NC = Not Connected

# RS-232/RS-485 Converter

The RS-232/RS-485 Converter(IC690ACC900)는 RS-232 에서 RS-422/RS-485 통신으로 변환하는데 사용될 수 있습니다. 컨버터는 15-pin female D-type port, 와 25-pin female D-type port 를 가지고 있습니다.

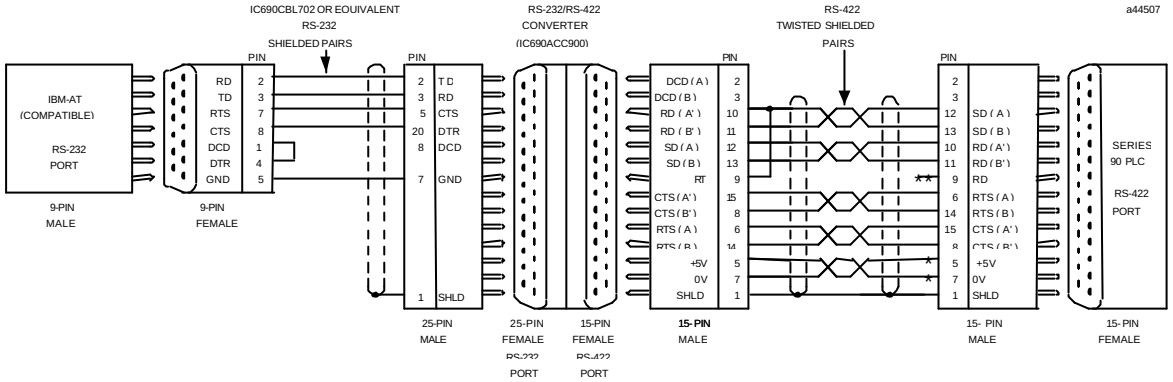
이 컨버터는 GE Fanuc Automation 에서 구입하실 수 있습니다. GE Fanuc Automation 사무소나 대리점으로 연락 주십시오.

# Serial Cable Diagram

이 섹션은 Series 90 과 VersaMax PLC 용 두 시리얼 포트 통신을 보여 주고 있습니다. 케이블링은 애플리케이션에 사용되는 디바이스와 컨버터에 따라 매우 다양합니다.

Point- to-Point 구성에서 두 디바이스만이 동일한 통신 라인에 연결될 수 있습니다. 통신 라인은 RS- 232(50 feet, 15m 최대)나 RS- 485(4000feet, 1200 m 최대)를 이용해 직접 연결 시킬 수 있습니다. 모뎀은 원거리에 사용할 수 있습니다.

## RS-232 Point-to-Point Connection

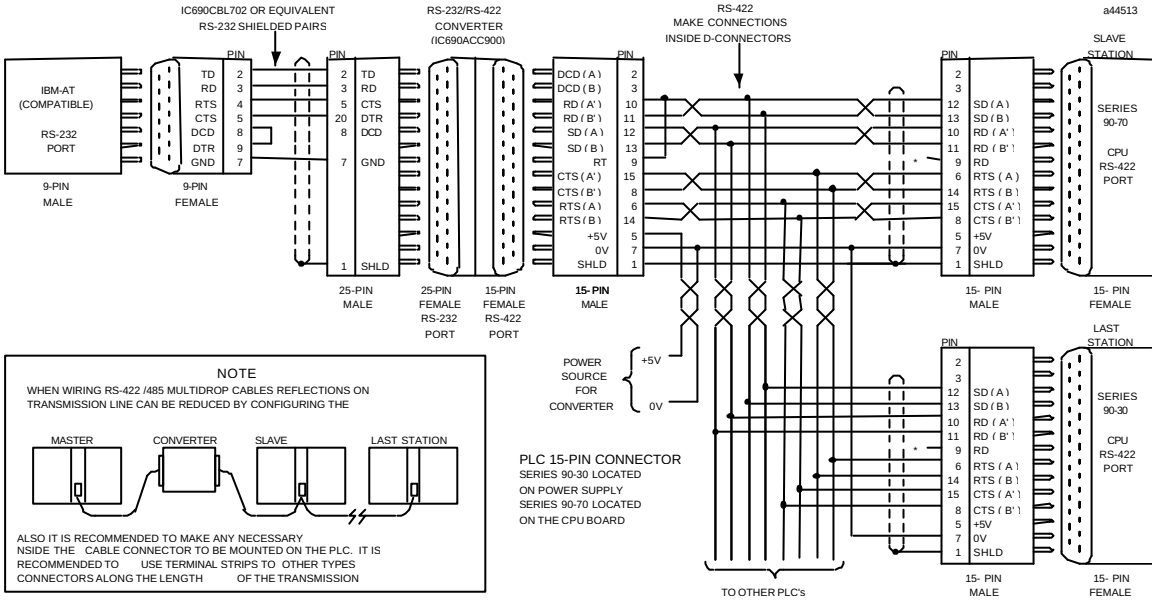



* POWER SOURCE FOR POINT-TO-POINT CONNECTION 10 FEET (3 METERS) ONLY. CONVERTER POWER SOURCE BEYOND 10 FEET (3 METERS) AND FOR CONNECTION MUST BE EXTERNAL

** TERMINATION RESISTANCE FOR THE RECEIVE DATA (RD) SIGNAL NEEDS TO BE CONNECTED ONLY ON UNITS AT THE END OF THE LINES. THIS TERMINATION IS SERIES 90 PLC PRODUCTS BY CONNECTING A JUMPER BETWEEN PIN 9 AND PIN 10 INSIDE THE 15-PIN D-SHELL WITH THE FOLLOWING EXCEPTION. FOR SERIES CATALOG NUMBERS IC697CPU731 AND IC697CPU771. THE TERMINATION FOR RD AT THE PLC IS IMPLEMENTED BY A JUMPER BETWEEN PIN 9

## Multidrop Connection

멀티드롭 구성에서, 호스트 디바이스는 마스터로 구성되고 하나 이상의 PLC 는 슬레이브로 구성됩니다. 이 방법은 마스터와 슬레이브간의 최대 거리가 4000feet(1200m)를 넘지 않을 때 사용될 수 있습니다. 이 그림은 양질의 케이블과 원만한 *Noisy* 환경 조건을 가정한 것입니다. 최대 8 대의 슬레이브는 *Daisy Chain* 이나 멀티드롭 구성에서 RS-422 를 이용해 연결할 수 있습니다. RS-422 라인은 핸드 handshaking 을 포함해야 하고 *Cable and Connector Specification* 에 명시된 와이어 타입을 사용해야 합니다.




* TERMINATION RESISTANCE FOR THE RECEIVE DATA (RD) SIGNAL NEEDS TO BE CONNECTED ONLY ON UNITS AT THE END OF THE LINES. THIS TERMINATION IS SERIES 90 PLC PRODUCTS BY CONNECTING A JUMPER BETWEEN PIN 9 AND PIN 10 INSIDE THE 15-PIN D-SHELL WITH THE FOLLOWING EXCEPTION. FOR SERIES CATALOG NUMBERS IC697CPU731 AND IC697CPU771. THE TERMINATION FOR RD AT THE PLC IS IMPLEMENTED BY A JUMPER BETWEEN PIN 9

GROUND POTENTIAL: MULTIPLE UNITS. NOT CONNECTED TO THE SAME POWER SOURCE. MUST HAVE COMMON GROUND POTENTIALS OR GROUND PROPER OPERATION OF THIS

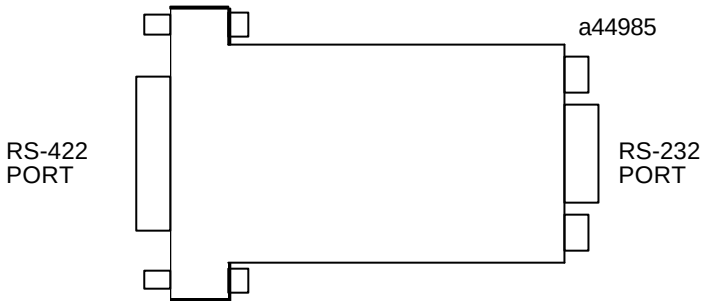
**B**

# *Miniconverter Kit*

이 장은 Series 90 과 VersaMax PLC 와 사용되는 Miniconverter Kit 에 대해 기술하고 있습니다.

## Miniconverter 개요

Miniconverter Kit(IC690ACC901)는 RS-422 (SNP) to RS-232 Miniconverter, 6foot (2m) 시리얼 확장 케이블과 9-pin ~ 25-pin Converter Plug Assembly 로 구성되어 있습니다. Miniconverter 에 있는 15-pin SNP Port Connector 는 Series 90-30 과 VersaMax PLC 에 있는 시리얼 포트에 직접 접속됩니다. Miniconverter 에 있는 9-pin RS-232 Port Connector 는 RS-232 호환 디바이스와 연결됩니다.




IBM PC-AT 나 호환 컴퓨터를 사용할 때, 확장 케이블 한쪽 끝은 Miniconverter 의 9-pin Serial Port Connector 와 접속되고, 다른 한쪽 끝은 컴퓨터의 9-pin Serial Port 에 접속됩니다. The Converter 플러그(키트에서 공급되는)는 Miniconverter 의 9-pin Serial Port Connector 를 IBM PC-XT 나 PS/2 Personal Computer 의 25-pin Serial Port Connector 로 변환할 필요가 있습니다 .

## Pin 할당, RS-232 Port

Miniconverter 의 Pinout 은 다음 두 테이블에 나타내었습니다. 첫 번째 테이블은 RS-232 포트용 Pinout 을 나타낸 것입니다. 신호 흐름의 방향은 Miniconverter 에 관한 것입니다.

| Pin | Signal Name | Direction |
|---|---|---|
| 2 | SD - Send Data | Output |
| 3 | RD - Receive Data | Input |
| 5 | GND - Ground | n/a |
| 7 | CTS - Clear To Send | Input |
| 8 | RTS - Request To Send | Output |

Pinout 들은 IBM PC-AT 에 직접 연결하거나 일 대 일 케이블(키트에서 제공되는)로 연결할 수 있습니다. 대부분의 RS-232 포트를 장착한 IBM 호환 컴퓨터들은 위에 나타낸 Pinout 중 하나를 제공할 것입니다.

## Pin 할당, RS-422 Port

다음 테이블은 Miniconverter 의 RS-422 시리얼 포트 용 Pinout 을 나타낸 것입니다. 신호 흐름의 방향 또한 Miniconverter 에 관한 것입니다.

| Pin | Signal Name | Direction |
|---|---|---|
| 1 | SHLD - Shield | n/a |
| 5 | +5 VDC - Power | Input |
| 6 | CTS(A') - Clear To Send | Input |
| 7 | GND - Ground | n/a |
| 8 | RTS(B) - Request To Send | Output |
| 9 | RT - Receive Termination | Output |
| 10 | SD(A) - Send Data | Output |
| 11 | SD(B) - Send Data | Output |
| 12 | RD(A') - Receive Data | Input |
| 13 | RD(B') - Receive Data | Input |
| 14 | CTS(B') Clear To Send | Input |
| 15 | RTS(A) - Request To Send | Output |

# 시스템 구성

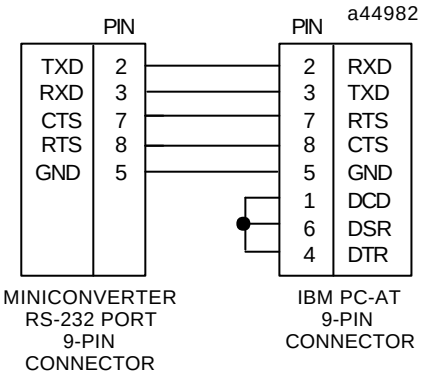
Miniconverter 는 위에 서술한 것처럼 Point-to-Point 구성이나 마스터로 구성된 호스트 디바이스와 슬레이브로 구성된 하나 이상의 PLC 로 구성된 멀티드롭 구성에서 사용될 수 있습니다.

멀티드롭 구성은 Miniconverter 의 RS-422 포트에서 첫 슬레이브 PLC 의 SNP 포트까지 곧은 (1 to 1) 케이블을 필요로 합니다. 다른 슬레이브들은 슬레이브간의 Daisy Chain Connection 을 필요로 할 것입니다. 최대 8 대의 디바이스는 RS-422 멀티드롭 구성으로 연결시킬 수 있습니다. 모든 디바이스들은 접지시켜야 합니다. 만약 절연이 필요하다면 Miniconverter 대신에 GE Fanuc Isolated Repeater/Converter(IC655CCM590)를 사용할 수 있습니다.

모뎀 연결로 Miniconverter 을 사용할 때 RTS 와 CTS 로 점퍼할 필요가 있습니다(모뎀에 대한 유저 매뉴얼 참조)

## Cable Diagrams (Point-To-Point)

Miniconverter 를 Hardware Handshaking 을 지닌 IBM PC 호환 컴퓨터에 연결할 때, 다음 케이블 연결을 사용해야 합니다:



다음 테이블은 Miniconverter 의 사양입니다

| | |
|---|---|
| **Mechanical:** | |
| **RS-422** | 15-pin D shell male(Series 90 나 VersaMax 시리얼 포트에 직접 취부용). |
| **RS-232** | 9-pin D shell male(Workmaster II computer 와 Personal Computer 의 RS-232 시리얼 포트 연결용). |
| **Electrical and General:** | |
| **Voltage Supply** | +5 VDC(PLC 파워 서플라이에 의해 공급) |
| **Typical Current** | Version A(IC690ACC901A) - 150mA<br>Version B(IC690ACC901B) - 100mA |
| **Operating Temperature** | 0 to 70° C (32 to 158° F) |
| **Baud Rate** | 최대 38.4K |
| **Conformance** | EIA-422(Balanced Line) 또는 EIA-423(Unbalanced Line) |
| **Ground Isolation** | 제공 안됨 |

## 마



## 바

# 사

# *Index*



## 아

# 자



# 차



# 카

# 타



# 파

# 하