Digital

Human Machine Interface

Pro-face

PL-6920/PL-7920 Series
Panel Computer
User Manual

When printing out this PDF manual, since the printer setting can effect the quality of the printout, be sure your printer's quality setting is set to "High".

Digital

Digital Electronics Corporation

Introduction

The PL-6920/PL-7920 series (PL) of Panel Computers are multipurpose factory automation (FA) computers, which embody Pro-face's latest, cost-effective architecture. Before using the PL, read this manual thoroughly to familiarize yourself with the PL's operation procedures and functions.

NOTE:

1. It is forbidden to copy the contents of this manual in whole, or in part, without the permission of the Digital Electronics Corporation.
2. The information in this manual is subject to change without notice.
3. This manual was written with care; however, if you should find any error or omissions, please contact Digital Electronics Corporation and inform them of your findings.
4. Regardless of the above clause, Digital Electronics Corporation shall not be held responsible for any damages or third-party claims for damages or losses resulting from the use of this product.

Product names used in this manual are the trademarks of their respective manufacturers.

© Copyright 2004, Digital Electronics Corporation

Windows® 95 OSR2 or later, Windows® 98 SE, Windows NT® 4.0, and Windows® 2000 are registered trademarks of the Microsoft Corporation.

Essential Safety Precautions

This manual includes the following cautions concerning procedures that must be followed to operate the PL correctly and safely. Prior to operating the PL, be sure to read this manual and any related materials thoroughly to understand the correct operation and functions of this unit.

Safety Icons

To allow you to use the PL correctly, throughout this manual, the following icons are provided next to operations requiring special attention. These icons are used to describe the following situations:

⚠ Warning	Indicates situations where severe bodily injury, death or major equipment damage may occur.
⚠ Caution	Indicates situations where slight bodily injury or machine damage can occur.

⚠ WARNINGS

- **To avoid the possibility of an electric shock, be sure to disconnect the power cord to the PL before connecting it to the main power supply.**
- **A fire or electrical shock may occur if voltages used with the PL are beyond the specified range. Be sure to use only the specified voltage.**
- **Before opening the PL's protective cover, be sure to turn the unit's power OFF. This is because the PL's internal parts carry high voltages.**
- **To avoid fires or electrical hazards, do not modify the PL in any way.**
- **Do not create touch panel switches that are used to either control or to ensure the safety of equipment and personnel. Mechanical switches, such as an emergency stop switch, a deadman (two-handed) start switch, etc., must be installed and operated via a separate control system.**

⚠WARNINGS

- Do not create touch panel switches which could possibly endanger the safety of humans and equipment. This is due to the possibility of a malfunction in the PL or its cable(s), causing the output of a signal that could result in a major accident. All of a system's major, safety-related switches should be designed to be operated separately from the PL.

- After the PL's backlight burns out, unlike the PL's "Standby Mode", the touch panel is still active. If the operator fails to notice that the backlight is burned out and touches the panel, a potentially dangerous machine miss-operation can occur.

  If your PL's backlight suddenly turns OFF, use the following steps to determine if the backlight is actually burned out.

  1) If your PL is not set to "Standby Mode" and the screen has gone blank, your backlight is burned out.
  2) Or, if your PL is set to Standby Mode, but touching the screen does not cause the display to reappear, your backlight is burned out.

- If metal particles, water or other types of liquids contact any of the PL's internal parts, immediately turn the unit's power OFF, unplug the power cord, and contact either your PL distributor or the Digital Electronics Corporation.
- Read and understand Chapter 4 "Installation and Wiring" thoroughly in order to select an appropriate installation location for the PL.
- Before either plugging in or unplugging a board or interface connector, be sure to turn the PL's power OFF.

⚠ WARNINGS

- To prevent a possible explosion, do not install the PL in areas containing flammable gases.
- The PL is not appropriate for use with aircraft control devices, aerospace equipment, central trunk data transmission (communication) devices, nuclear power control devices, or medical life support equipment, due to these devices' inherent requirements of extremely high levels of safety and reliability.
- When using the PL with transportation vehicles (trains, cars and ships), disaster and crime prevention devices, various types of safety equipment, non-life support related medical devices, etc. redundant and/or fail-safe system designs should be used to ensure the proper degree of reliability and safety.

⚠ CAUTIONS

- Do not push on the PL's screen too strongly, with either your finger or with a hard object. Excessive pressure can scratch, crack or damage the screen. Also, do not use a pointed object, such as a mechanical pencil or screwdriver, to press any of the touch panel's switches, since they can damage the display.
- If the screen becomes dirty or smudged, moisten a soft cloth with diluted neutral detergent, wring the cloth well, and wipe the display. Do not use thinner or organic solvents.
- Avoid exposing the PL to, or operating the PL in direct sunlight, high temperatures and humidity, and in areas where excessive dust and vibration will occur.

CAUTIONS

- Avoid using the PL in areas where sudden, extreme changes in temperature can occur. This may cause condensation to form inside the unit, possibly leading to an accident.
- To prevent the PL from overheating, be sure its air circulation vents are clear and clean, and keep the unit's operation area well-ventilated.
- Avoid operating or storing the PL near chemicals, or where chemicals can come into contact with the unit.

When PL Hard Disk (HDD) data is lost:

- The Digital Electronics Corporation cannot be held responsible or provide any compensation for damage(s) caused by the loss of data stored in the PL's hard disk drive (HDD). It is therefore strongly suggested that all important data and software be backed up regularly to an external data backup device.
- Please be aware that the Digital Electronics Corporation bears no responsibility for any damages resulting from the customer's application of this unit's hardware or software.
- Since the PL unit's hard disk drive (HDD) is a consumable item, i.e. it has a limited lifetime, be sure to back up its data regularly and prepare a spare HDD unit.
- To prevent file data damage, be sure to shut down the PL's OS before turning OFF the main power.
- After turning OFF the PL's power, wait until the internal HDD stops spinning before turning on the power again (approx. 5 seconds).
- When the PL unit's standard display is connected to the PL using the USB connection, wait three (3) seconds before turning the power ON again.

LCD Usage and Handling

- The PL's LCD contains a strong irritant. If the panel is ever cracked and the LCD's liquid contacts your skin, be sure to wash it with running water for at least 15 minutes. If any of this liquid should enter your eye, be sure to flush your eye with running water for more than 15 minutes and see a doctor as soon as possible.
- The brightness of the LCD screen will depend on the screen's current display and the LCD's contrast adjustment. Any brightness variations that result are normal for LCD displays.
- There are minute grid-points (Dark or Light points) on the LCD surface. These points are not defects and are a part of the PL unit's design.
- The displayed color will look different when viewed from an angle outside the specified view angle. This is also normal.
- When installing this unit, be sure that the screen is viewable from within the designated viewing angles. The screen image being difficult to see from outside its recommended viewing angle is normal.
- Displaying a single screen image for long periods of time can cause an afterimage to remain on the screen. To correct this, turn the unit OFF for 5 to 10 minutes, then ON again. This phenomenon is a common attribute of LCD displays, and is not a defect. To prevent this effect, you can:

1) Use the Display OFF feature; if the same image is to be displayed for a long period of time.

2) Change the screen display periodically to prevent the displaying of a single image for a long period of time.

Table of Contens

Chapter 1 Overview

Chapter 2 Specifications

Chapter 3 Installing Optional Units and Expansion Boards

Chapter 4 Installation and Wiring

Chapter 5 System Setup

Chapter 6 Setting Up Your PL OS

Chapter 7 Maintenance and Inspection

Appendices

Information Symbols

This manual uses the following icons:

CAUTION!	Indicates a warning or a product limitation. Be sure to follow the instructions given with this icon to insure the safe operation of the PL.
NOTE	Contains additional or useful information.
*	Indicates terms or items that require further explanation. See the footnote on that page.
☞	Indicates pages containing related information.
1), 2)	Indicates steps used to accomplish a given task. Be sure to follow these steps in the order they are written.
PL-X920 Series	Refers collectively to the PL-6920, PL-7920 and PL-B920 Series units.

PL Series Panel Types

Model Number:

PL*92 * - T 4 *

A B C D E

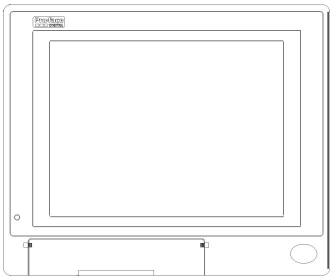
A	6	PL-6920 Series Unit
	7	PL-7920 Series Unit
B	0	4-slot type
	1	2-slot type
C	T	TFT Color LCD display
D	4	CE Marking, UL/c-UL (CSA) Approval
E	1	CPU:700MHz
	2	CPU:1GHz

Package Contents

The PL package should include the following items:

PL Unit

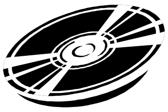
(PL-6920/PL-6921/PL-7920/PL-7921)



CAUTION! **Be careful when installing the PL not to damage the built-in HDD**

CD-ROM (1)

(PL-X920 Series User Manual & Driver CD)

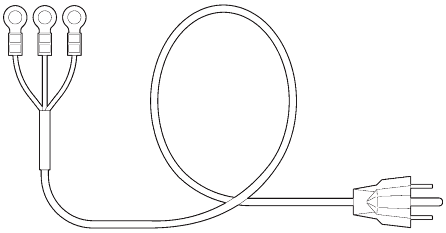


NOTE For details, refer to Chapter 6 - PL Setup

Installation Guide

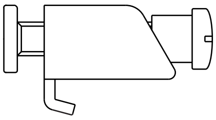
(English1/Japanese1)

Power Cord



CAUTION! **This cord is designed only for AC100V use. Any other voltage will require a different cord.**

Installation Brackets



NOTE PL-6920Series (8)

PL-7920Series (12)

NOTE If your PL unit contains a built-in accessory, that accessory's Installation Guide will also be included in the PL's packing box. Please check that all items normally included with that accessory are also included in this box.

PL Series Features

The PL-6920/PL-7920 series displays are equipped with the following features:

■ The Latest, High-Performance Architecture

Designed around the Pentium® III 700MHz CPU or 1GHz, the PL utilizes the type of high performance architecture that offers you superior compatibility. Add to this unrivalled support of the Windows® XP and other widely used operating systems.

■ Bright LCD with a Wide Viewing Angle

The PL's large TFT LCD display offers excellent visibility and brightness.

NOTE

Pro-face's top of the line TFT color LCD allows you to create detailed and powerful visual images, with excellent brightness, a wide viewing angle, and a display capable of 64K colors.

■ Easy Front Panel Installation

The PL is designed to be installed easily into the front of any panel or device. It is also rugged enough for use in harsh, industrial environments, such as those found in the factory automation industries and provides protection equivalent to the IP65f standard.

■ High Resolution, Analog-Resistance-Film Touch Panel

Standard equipment with the PL is a high resolution 1024 x 1024 touch panel, and the mouse emulation utility provides mouse-like functionality and pointer control.

■ Highly Expandable

Choose from the PL-6921/PL-7921 series unit - with 2 ISA slots, or the PL-6920/PL-7920 series unit - with 4 ISA slots. In addition, the 6921/PL-7921 unit provides a single PCI bus (slot 2), while the PL-6920/PL-7920 unit provides two PCI buses (slots 2 and 3). Also, these slots can accommodate both Digital's own optional boards as well as other commercially available expansion boards. Digital also offers a wide variety of optional products, such as a -5/-12V DC power unit, DIM memory modules, etc.

UL/c-UL (CSA) Application Notes

The PL6920-T4* / PL6921-T4* /PL7920-T4* /PL7921-T4* Series units are (c)UL 1950 recognized components. (UL File No. E171486). Please pay special attention to the following instructions when applying for UL/c-UL (CSA) approval for machinery which includes any of these PL units.

The PL conforms as a component to the following standards:

UL 1950, Third Edition, dated March 1,1998 (Standard for Safety of Information Technology Equipment, including Electrical Business Equipment)
CSA-C22.2 No. 950-95 (Standard for Safety of Information Technology Equipment, including Electrical Business Equipment)

PL6920-T4* (UL Registration Model No.: 2780054-04)
PL6921-T4* (UL Registration Model No.: 2780054-03)
PL7920-T4* (UL Registration Model No.: 2780054-02)
PL7921-T4* (UL Registration Model No.: 2780054-01)

- Equipment with a PL mounted in it requires UL/c-UL(CSA) evaluation for the combination of the PL and equipment.

- The PL must be used as a built-in component of an end-use product.
- Use the PL indoors only.
- When connecting the PL's power cord, be sure to use a cord that is appropriate for the current and voltage used and that has conductive wires that are 0.75 mm^2 or larger.
- With an end-use product which includes the PL, be sure to place the PL's Power cut-off switch where the unit's operator can easily reach it.
- Danger of explosion if backup battery is incorrectly replaced. Replaced only with same or equivalent type recommended by the manufacturer. Dispose of used batteries according to the manufacturer's instructions.
- Be sure the unit the PL is built into uses a UL1950 compatible equipment structure.

CE Marking

The PL6920-T4* / PL6921-T4* /PL7920-T4* /PL7921-T4*units are CE marked, EMC compliant products.

<Complies with the following Standards>

■ **Safety**

EN60950

■ **EMI**

EN55011 (Group1 Class A), EN61000-3-2, EN61000-3-3

■ **EMS (EN61000-6-2)**

EN61000-4-2, EN61000-4-3, EN61000-4-4, EN61000-4-5
EN61000-4-6, EN61000-4-8, EN61000-4-11

If following requirements are not met, the PL may fail to meet EN60950 standard requirements.

- The PL must be used as a built-in component of an end-use product.
- Use the PL indoors only.
- When connecting the PL's power cord, be sure to use a cord that is appropriate for the current and voltage used and that has conductive wires that are 0.75 mm^2 or larger.
- When installing the PL in a metal panel or cabinet, be sure to place the PL's Power disconnect device (cut-off switch) where the unit's operator can easily reach it.
- There is a danger of explosion if the backup battery is incorrectly replaced. This battery should be replaced only with same or equivalent type recommended by the manufacturer. Dispose of used batteries according to the manufacturer's instructions.
- Be sure the cabinet/enclosure the PL is built into uses an EN60950 approved sheet steel structure.

MEMO

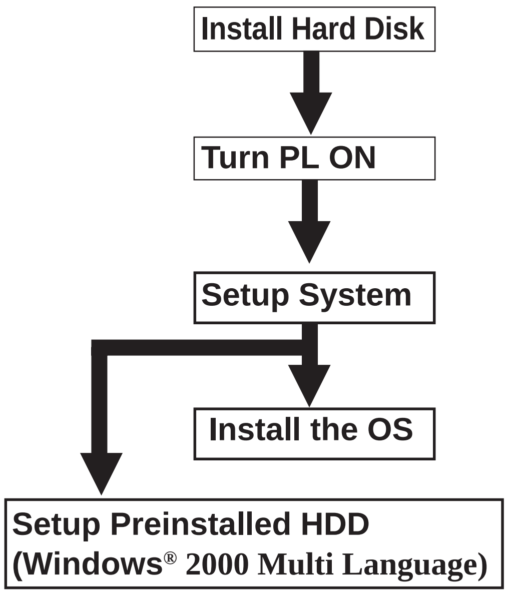
Chapter

1 Overview

1-1 Prior To Using the PL

Prior to actual use, be sure to set up your PL as follows.

■ Using the PL with a Hard Disk Unit Installed



☞ Refer to 3-1-5 Removing/Installing HDD Unit

☞ Refer to 4-2 Wiring the PL

☞ Refer to Chapter 5 System Setup

☞ Refer to 6-2-1 Setting UP HDD with no installed OS

☞ Refer to 6-2-2 Setting UP OS preinstalled HDD

- **After completing the hardware setup, before any data or applications can be placed on the drive, the OS (Windows® or MS-DOS®, etc.) must be used to initialize the HDD and create partitions. For details concerning these procedures, refer to the OS maker's installation manual.**
- **The PL's hard disk is designed for use with the Windows® 95, Windows® 98 Second Edition, Windows NT®4.0, Windows® 2000, Windows® XP or later OS. The Mirror Disk unit will operate only with the Windows NT®4.0 operating system. Other operating systems do not support this driver software, etc.**
- **After turning the PL OFF, be sure to wait at least 5 seconds before turning ON again. If the unit is stated within 5 seconds, it may not start up correctly.**

1 Setting Up the Touch Panel Connection

The connection method used can be via either a serial (RS-232C) or USB interface. Depending on the type of Touch Panel connection used, the OS types that can be used will vary.

Touch Panel Connection	Compatible OS Types
USB	Windows® 98 Second Edition Windows® 2000 Windows® XP
Serial (RS-232C)	Windows® 95 Windows® 98 Second Edition Windows NT®4.0 Windows® 2000 Windows® XP

The PL unit's factory setting is "Serial". When the touch panel connection method is changed to USB, the following BIOS level System settings must also be changed.

■ System Data Settings

For System Data Setting details, refer to **Chapter 5 - System Setup**

System Setting Menu	Setting Item	USB Setting
Integrated Peripherals	USB Controller	Enabaled
	Onboard Serial Port 4	Disabled
PnP/PCI Configuration	Assign IRQ For USB	Enabled

■ Touch Panel Interface Selector Switch (T-MODE)

Set the Touch Panel I/F Selector Switch (T-MODE) to "U". For information about the Touch Panel Interface (I/F) Selector Switch (T-MODE), refer to **2-4 Part Names and Features**

■ Mouse Emulation Software(UPDD)

When installing the Mouse Emulation Software, be sure to select USB.

2 Using the USB Interface

The PL unit's USB interface cannot be used without changing the factory settings. When connecting peripheral devices to the USB port, change the System Settings as shown below.

System Setup Menu	Menu Item	USB Setting
Integrated Peripherals	USB Controller	Enabled
PnP/PCI Configuration	Assign IRQ For USB	Enabled

CAUTION!

- *When using a commercial-type USB hub, additional hubs cannot be attached to the first hub. Only a single "level" USB hub can be used when connecting USB devices.*

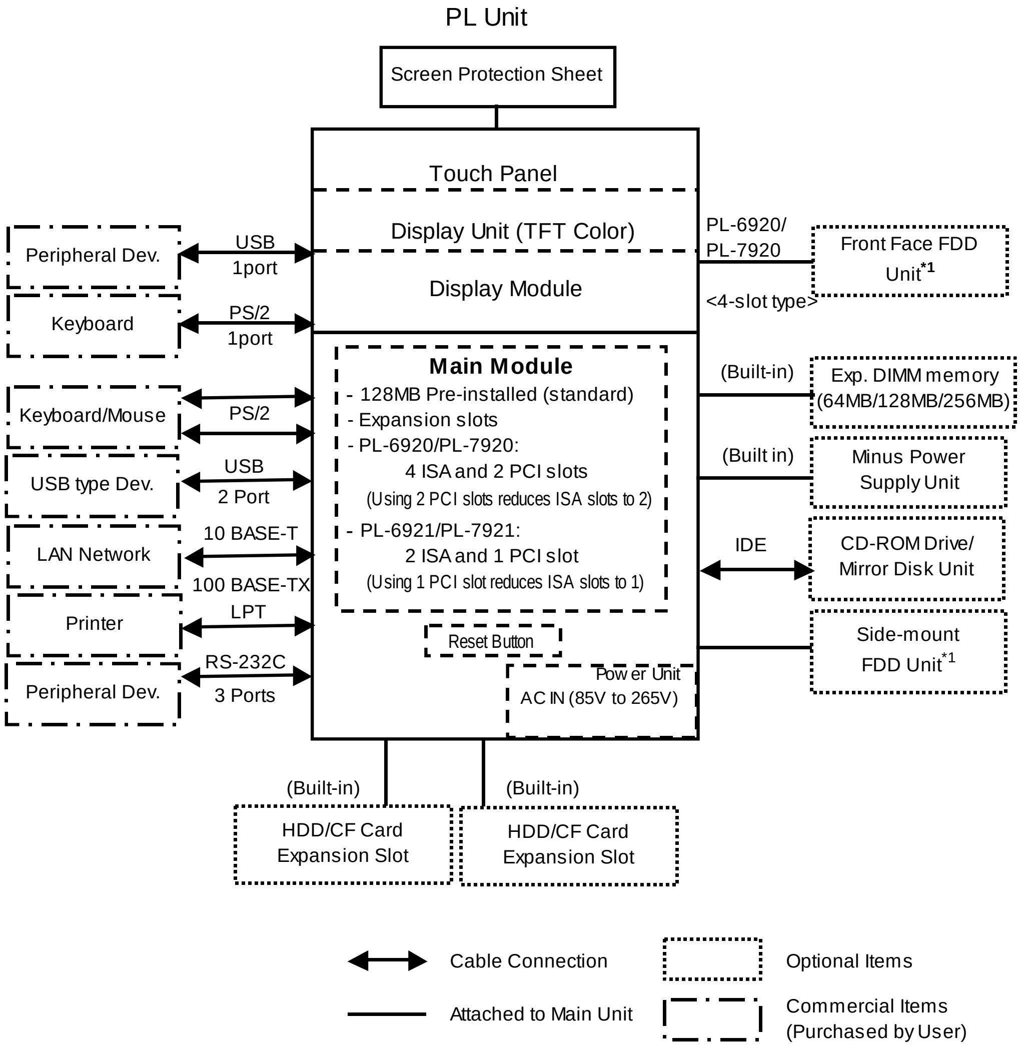
3 Using the LAN Interface

The PL unit's LAN interface cannot be used without changing the factory settings. When using the LAN port, change the System Settings as shown below.

System Setup Menu	Menu Item	LAN Setting
Integrated Peripherals	Onboard LAN	Enabled

1-2 System Configuration

The following chart shows the range of peripheral items connected to the PL.



CAUTION!

- ***This diagram shows only the PL's internal layout and connectable devices. The user's actual design may differ.***

**1 Only one FDD unit can be used at one time, i.e. either the front panel's FDD, or the main unit's FDD.*

1-3 Options

Expansion Options

Item	Model number	Description
DIM Module	PL-EM500	SDRAM (DIMM) Provides 64MB of memory
	PL-EM128	SDRAM (DIMM) Provides 128MB of memory
	PL-EM256	SDRAM (DIMM) Provides 256MB of memory
HDD Unit*1	PL-HD220	20GB 2.5" HDD Unit (OS not included)
Windows 2000 Multi Language Preinstalled HDD	PL-HDX920-W2K/ML	HDD is a 2.5 inch type unit. Capacity is 20GB. (PL unit's C: drive is already designated as 20GB.) Preinstalled OS is Windows 2000 Professional <Service Pack 2>. (Sold only in combination with the PL main unit - not separately.)
FDD Unit	PL-FD200	Windows compatible 3.5" FDD unit (Side slot)
	PL-FD210	Windows compatible 3.5" FDD unit (Front slot - PL-6920/PL-7920 only)
-5V/-12V Power Supply	PL-PW100	Provides -5V and -12V power to PL expansion slots. Total for two slots is 200mA.
CD-ROM Unit	PL-DK200	IDE (ATAPI) compatible CD-ROM drive unit (Connection cable is included with CD-ROM unit)
CF Card Unit	PL-CF200	Designed exclusively for 5V type cards.
Software Mirroring Utility	PL-SM900	Provides RAID Level 1 protection, without Mirror Disk unit.
Full-Sized Board Cover	PL-FC200	Used when ISA full-sized expansion board is used. (Used only with PL-6921/PL-7921<2-slot type>.)
	PL-FC210	Used when ISA full-sized expansion board is used. (Used only with PL-6920/PL-7920<4-slot type>.)
Mouse Emulation Software *2	UPDD	This software adds mouse and keyboard-like functionality to the Touch Panel.
RS-232C/RS-485 Conversion Unit	PL-RC500	Converts an RS-232C interface to a RS-485 interface. Attached to either COM2 or COM3.
Screen Protection Sheet	PL-CS100	Disposable overlay sheets for display face protection and stain resistance. Touch panel senses User's touch through sheet.(5sheets / set)
CF Card	GP077-CF20	CF Card(16MB) CF Card Unit (PL-CF200) is required.
	GP077-CF30	CF Card(32MB) CF Card Unit (PL-CF200) is required.

**1 The PL's HDD has a fixed usage lifetime. Be sure to back up data regularly and prepare a spare drive unit.*

**2 Visit our website (http://www.pro-face.com/otasuke/) and download the mouse emulation software(UPDD).*

Maintenance Options

Item	Model number	Description
Mirror Disk Unit Replacement HDD	PL-MD200-MD01	Mirror Disk Unit's replacement HDD (1).
Installation Fasteners	GP070-AT01	Used to install the PL into a panel or cabinet. Same as original equipment fasteners. (4 fasteners/set)
Installation Gasket	PL6900-WP00	Used to prevent moisture from entering into the PL's case from the front face. Same as original equipment gasket.
	PL7900-WP00	
Backlight	PL6920-BL00	Spare backlight for maintenance.
	PL7900-BL00-MS	Spare backlight for maintenance.

**1 Both the PL-FD200 and the PL-FD210 cannot be used at the same time.*

- **When using the FDD unit with PL-6920/PL-7920 (4-slot type), either the side mount FDD unit (PL-FD200) or the front mount FDD unit (PL-FD210) is available, not simultaneously.**
- **The PL is equipped with three IDE interfaces, two (2) of which can be used by the HDD unit, and one (1) which can be used by either the CD-ROM drive unit. Physically, even though up to three (3) IDE drive units can be connected at the same time, IDE interface specifications require that a controller's simultaneous operation be limited to a single master and slave unit, for a total of two devices.**

The following chart shows the combinations available when using two IDE units.

HDD Unit	MS	M	M	S		
CD-ROM Drive Unit		S			S	
CF Card Unit			S	M	M	MS

MS: Combination of 2 units - Master or Slave
M: Used only for Master.
S: Used only for Slave.

CAUTION!

- ***Since the PL unit's hard disk drive (HDD) is a consumable item, i.e. it has a limited lifetime, be sure to back up its data regularly and prepare a spare HDD unit.***
- ***The Hard Disk lifetime given here may be reduced due to unforeseen environmental factors, however, generally speaking, at an operating temperature of 20oC the disk should last for 20,000 hours (of operation) or approximately 5 years, whichever comes first.***

■ Commercially Available Items

The PL-6920 and PL-7920 Series units can all use commercially available expansion boards (PCI/ISA compatible) as well as a standard keyboard, mouse,printer, etc. When using a USB cable, the PL-6920 and PL-7920 Series units can also use USB compatible devices. However, among the commercially available USB devices, not all will be compatible with the PL unit.

NOTE

- **Since the PL expansion slots do not supply DC-5V and DC-12V current. To use expansion boards requiring DC-5V or DC-12 V, the optional PL-PW100 power supply should be installed.**
- **Be sure to use only DIM modules manufactured by Digital. Installing other DIM modules may result in either damage to or failure of the PL, and will void your warranty.**
- **When using USB type devices, be sure they are USB compatible, and be sure to read that device's installation guide prior to connecting it to the PL.**

Chapter 2

Specifications

2-1 General Specifications

1 Electrical Specifications

	PL-6920/PL-7920 (4-slot type)	PL-6921/PL-7921 (2-slot type)
Rated Voltage	AC 100V to AC 240V	
Rated Voltage Range	AC 85V to AC 265V	
Rated Frequency	50/60Hz	
Allowable Pause Duration	shorter than 1 cycle (however, pause occurrences must be more than 1 second apart)	
Power Consumption	150VA or less	120VA or less
Voltage Endurance	AC 1500V at 20mA for 1 minute (between the live wire and the grounding (FG) terminal)	
Insulation Resistance	Greater than 10MΩ at DC 500V (between the live wire and the grounding (FG) terminal)	

2 Environment Specifications

Ambient Operating Temperature	PL692*-T41 (CPU:700MHz)	Using Fan		5°C to 50°C (With HDD attached)
		Not using fan *1		5°C to 40°C (With HDD attached)
	PL792*-T41 (CPU:700MHz)	Cabinet Interior	Using Fan	5°C to 50°C (With HDD attached)
			Not using fan*1	5°C to 40°C (With HDD attached)
		Panel face	5°C to 40°C	
	PL692*-T42 (CPU:1GHz)	Using Fan		5°C to 45°C (With HDD attached)
		Not using fan *1		Must use fan
	PL792*-T42 (CPU:1GHz)	Cabinet Interior	Using Fan	5°C to 45°C (With HDD attached)
			Not using fan*1	Must use fan
		Panel face	5°C to 40°C	
Ambient Storage Temperature	-10°C to +60°C			
Ambient Humidity (Operating and Storage)	10% RH to 85% RH (no condensation)			
Dust Level	Free of dust			
Pollution Level	Pollution Level 2			
Maximum wet bulb temperature	29°C			
Operating Atmosphere	Free of corrosive gas			
Vibration Endurance	19.6m/s^2: 10 to 25Hz applied in X, Y, and Z directions for 30 minutes each (4.9m/s^2 when using HDD unit, 9.8m/s^2 when using FDD)			
Noise Endurance (Impulse Noise)	Noise Voltage: 1500V(via noise simulator) Pulse Duration: 50ns, 500ns, 1µs Rise Time: 1ns			
Noise Immunity (First transient burst noise)	Power Line: 2kV IEC 61000-4-4 COM Port: 1kV IEC 61000-4-4			
Electrostatic Discharge Immunity	4kV IEC 61000-4-2			

**1 When the PL's internal electric fan is removed.*

CAUTION!

- When using any of the PL's optional devices, be sure to check that device's specifications for any special conditions or cautions that may apply to its use.
- When using a full sized expansion board, be sure to check its dimensions and shape, since they will affect the board's environment specifications, such for vibration, etc.
- Be aware that not only does the Hard Disk have a fixed lifetime, but that accidents can always occur. Therefore, be sure to back up your Hard Disk's data regularly, or prepare another Hard Disk unit that can be used for backup.
- The Hard Disk lifetime given here may be reduced due to unforeseen environmental factors, however, generally speaking, at an operating temperature of 20°C the disk should last for 20,000 hours (of operation) or approximately 5 years, whichever comes first.
- Using the Hard Disk in an environment that is excessively hot and/or humid will shorten the disk's usage lifetime. A maximum wet bulb temperature of 29°C is recommended. This is equivalent to the following data.

Temperature	Humidity
at 35°C	no higher than 64%RH
at 40°C	no higher than 44%RH

3 Dimensions

		PL-6920 Series		PL-7920 Series	
		PL-6920	PL-6921	PL-7920	PL-7921
Grounding*1		Exclusive grounding only. Less than 100Ω, or your country's applicable			
Rating*2		Equivalent to IP65f (JEM1030)			
Cooling Method	700MHz CPU Model	Via heat convection tubes and electric fan			
	1GHz CPU Model	Via both heat convection tubes built in to CPU fan and electric fan			
Weight (with HDD and		Less than 9.5 kg	Less than 8.5 kg	Less than 10.5 kg	Less than 9.5 kg
External Dimensions (excluding projections)		W346×H287×D170mm	W346×H287×D123mm	W374×H325×D180mm	W374×H325×D134mm
Dimensions Including Full-sized Cover (excluding projections)		W393×H287×D170mm	W393×H287×D123mm	W422×H325×D180mm	W422×H325×D134mm
Dimensions Including RS-232C/RS-485 Conversion Unit (excluding cable & projections)		W346×H287×D170mm	W346×H287×D145mm	W374×H325×D180mm	W374×H325×D156mm

1* ☞ ***4-2-3 Grounding Cautions.*

**2 The front face of the PL unit, installed in a solid panel, has been tested using conditions equivalent to the standard shown in the specification . Even though the PL unit's level of resistance is equivalent to the standard, oils that should have no effect on the PL can possibly harm the unit. This can occur in areas where either vaporized oils are present, or where low viscosity cutting oils are allowed to adhere to the unit for long periods of time. If the PL's front face protection sheet becomes peeled off, these conditions can lead to the ingress of oil into the PL and separate protection measures are suggested. Also, if non-approved oils are present, it may cause deformation or corrosion of the front panel's plastic cover. Therefore, prior to installing the PL be sure to confirm the type of conditions that will be present in the PL's operating environment.*

If the installation gasket is used for a long period of time, or if the unit and its gasket are removed from the panel, the original level of the protection cannot be guaranteed. To maintain the original protection level, you need to replace the installation gasket regularly.

2-2 Performance Specifications

1 Performance Specifications

CPU		Pentium III (700MHz / 1GHz)		
DRAM(SDRAM DIMM)		Equipped with 128MB (2 DIMM sockets – max. of 512MB)		
BIOS		AWARD PC/AT Compatible		
Secondary Cache Memory		256KB (built-in)		
Graphics		PL-6920 Series	SVGA (800 x 600 dots)	
		PL-7920 Series	XGA (1024×768 dots)	
		VESA 16 colors/256 colors/64K colors		
Video Memory		UMA type		
Touch Panel	Type	Resistive Film (Analog)		
	Resolution	1024×1024		
	Interface	COM4	When the Mouse Emulation Software (UPDD) is installed, either the Serial (COM4) interface or the USB interface must be selected. [*1]	
		USB		
Inter-faces	Serial	RS-232C (w/FIFO)	COM1	D-Sub 9 pin male
			COM2	D-Sub 9 pin male (RI/+5V switch poss.)
			COM3	D-Sub 9 pin male (RI/+5V switch poss.)
	Printer	Complies with Centronics Standards (ECP/SPP/EPP equivalent) D-Sub 25 pin, female		
	Keyboard	PS/2 Interface (mini DIN 6 pin, female) front 1port / side 1port		
	Mouse	PS/2 Interface (mini DIN 6 pin, female) side 1port		
	RAS	RAS Interface (D-sub 25 pin, male)		
	Disk I/F	FDD Unit	Side Access/ 2 modes/ 3.5” FD	
			Front Access/ 2 modes/ 3.5” FD (Available for only PL-6920/PL-7920<4-slot type>)	
		IDE I/F	2.5” HDD I/F CD-ROM drive unit (PL-DK200)	
	USB[*2]	USB 1.1 compatible front 1port / side 2ports		
	LAN[*2]	Satisfies IEEE802.3 standard (auto change to 10BASE-T/100BASE-TX)		

**1 If the PL unit’s OS is Windows® 95 or WindowsNT® 4.0, only the Serial interface can be used.*

**2 BIOS setting must be changed.* Reference *5.2.5 Integrated Peripherals*

2 Display Functions

	PL-6920 Series	PL-7920 Series
Display Type	TFT Color LCD(12 inchi)	TFT Color LCD(15 inchi)
Pixel Density	800 x 600 pixels	1024 x 768 pixels
Dot Pitch	0.3075×0.3075mm	0.297×0.297mm
Effective DisplayArea	W246.0 x H184.5 mm	W304.1 x H228.1mm
Display Colors	64K colors	
Backlight’s Life span	More than 50,000 hours at an ambient temperature of 25°C. (Until the	
Backlight	CFL (Replaceable)	
Contrast Control	Not Possible	

The PL’s backlight should be replaced by only an authorized repairman. For information about this service, please contact your nearest authorized distributor.

3 Expansion Slots

		PL-6920/7920 (4-slot type)	PL-6921/7921 (2-slot type)	Board Size			
				Without the Full-sized cover	With the Full-sized cover	Slot Pitch	Actual Thickness of Expansion Board
1st slot		ISA	ISA	163 x 122 mm	163 x 122 mm	20 mm	Less than 13 mm
2nd slot		PCI	PCI	250 x 122 mm	250 x 122 mm	25 mm	Less than 18 mm
		ISA	ISA		338 x 122 mm		
3rd slot		PCI/ISA	None	250 x 122 mm	338 x 122 mm	25 mm	Less than 18 mm
4th slot		ISA	None	250 x 122 mm	338 x 122 mm	20 mm	Less than 13 mm
Power Supply	CPU 700MHz Model	5V:4A, 12V:1A (total for 4 slots)	5V:2A, 12V:0.5A (total for 2 slots)				
	CPU 1GHz Model	5V:3.5A, 12V:1A (total for 4 slots)	5V:1A, 12V:0.5A (total for 2 slots)				

NOTE

- For the 2nd and 3rd slots on the PL-6920/PL-7920<4-slot type>, and the 2nd slot on the PL-6921/PL-7921<2-slot type>, either the PCI or the ISA type can be used. Pins 1, 4, 16 and 17 will become O.D when the SPP mode specification is used. If the mode changes to ECP or EPP, these pins will become T.S

4 Clock(RTC) Accuracy

The PL's built-in clock (RTC) may have a slight error. With the ambient temperature mentioned in the specification with no power flow, the allowance is +180 seconds per month, however, the allowance may vary and could be up to +300 seconds per month depending on the ambient temperature difference or how old the unit is. If the clock accuracy is essential for the system, you need to adjust the clock regularly.

Clock(RTC) accuracy	±180 seconds per month

2-3 Interface Specifications

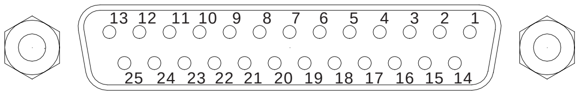
If the optional board (that connects to the extended slot (ISA/PCI) has a current consumption value equal to the specified upper limit value (described in "2.2.3 Expansion Slots", ensure that the I/O device's total 5V current consumption value conforms to the following standard:

	PLX920T-41 (CPU:700MHz) 4-slot type	PLX921T-41 (CPU:700MHz) 2-slot type	PLX920T-42 (CPU:1GHz) 4-slot type	PLX921T-42 (CPU:1GHz) 2-slot type
External I/O Device's Total Current consumption*1	1.5A	1.0A	0.5A	0.5A

***1** *Indicates the total current value being supplied from RS-232C I/F (COM2/COM3), RAS I/F, USB I/F and Mouse I/F's +5V terminal.*

1 Printer Interface (LPT1)

D-sub 25 Pin (Female)



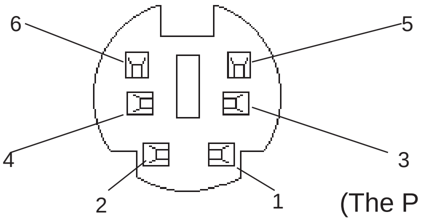
Screw Size: (4-40): Inch Type

Pin No.	SPP/ECP Mode Signal Name	EPP Mode Signal Name	Direction	Electrical Specif.	Pin No.	SPP/ECP Mode Signal Name	EPP Mode Signal Name	Direction	Electrical Specif.
1[*1]	$\overline{\text{STRB}}$	$\overline{\text{WRITE}}$	In/Output	O.D/T.S	14[*1]	$\overline{\text{AUTOFD}}$	$\overline{\text{DSTRB}}$	In/Output	O.D/T.S
2	DATA0	DATA0	In/Output	T.S	15	$\overline{\text{ERROR}}$	$\overline{\text{ERROR}}$	Input	TTL
3	DATA1	DATA1	In/Output	T.S	16[*1]	$\overline{\text{INIT}}$	$\overline{\text{INIT}}$	In/Output	O.D/T.S
4	DATA2	DATA2	In/Output	T.S	17[*1]	$\overline{\text{SLCTIN}}$	$\overline{\text{ADSTRB}}$	In/Output	O.D/T.S
5	DATA3	DATA3	In/Output	T.S	18	GND	GND		
6	DATA4	DATA4	In/Output	T.S	19	GND	GND		
7	DATA5	DATA5	In/Output	T.S	20	GND	GND		
8	DATA6	DATA6	In/Output	T.S	21	GND	GND		
9	DATA7	DATA7	In/Output	T.S	22	GND	GND		
10	$\overline{\text{ACKNLG}}$	$\overline{\text{ACKNLG}}$	Input	TTL	23	GND	GND		
11	BUSY	$\overline{\text{WAIT}}$	Input	TTL	24	GND	GND		
12	PE	PE	Input	TTL	25	GND	GND		
13	SLCT	SLCT	Input	TTL					

- Pins 1, 4, 16 and 17 will become O.D when the SPP mode specification is used. If the mode changes to ECP or EPP, these pins will become T.S

2 Keyboard Interface

Mini - DIN 6 pin (Female)

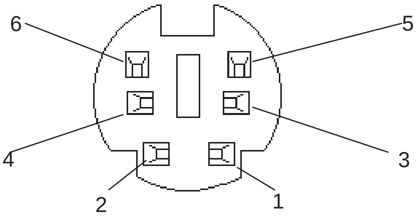


(The PL's front and side connectors are the same)

Pin No.	Signal Name
1	KEY DATA
2	NC
3	GND
4	+5V
5	KEY CLK
6	NC
SHIELD	GND

3 Mouse Interface

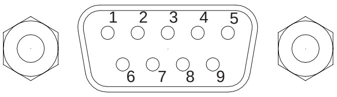
Mini - DIN 6 pin (Female)



Pin No.	Signal Name
1	Mouse DATA
2	NC
3	GND
4	+5V
5	Mouse CLK
6	NC
SHIELD	GND

4 RS-232C Interface (COM1/COM2/COM3)

Dsub 9 pin (Male)



Screw Size: (4-40): Inch Type

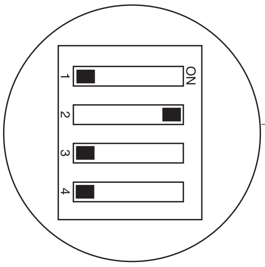
Pin No.	Signal Name	Pin No.	Signal Name
1	CD	6	DSR
2	RXD	7	RTS
3	TXD	8	CTS
4	DTR	9	RI/+5V
5	GND		

**1 COM2 and COM3 can perform RI/+5V changeover*

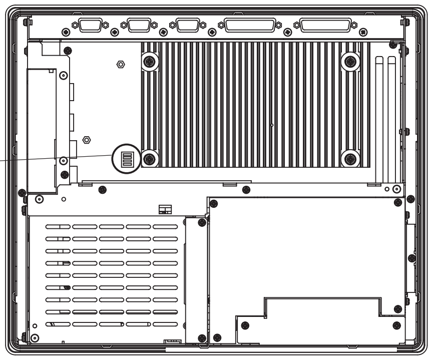
The No. 5 pin is the signal ground. Be sure to connect it with the other unit's SG (Signal Ground) pin.

Number 9 pin's [RI/+5V] changeover is COM2 and COM3. COM1 becomes [RI]. To set the changeover for COM2 and COM3, remove the PL's rear maintenance cover and use the main function switches, next to the PL's circuit board. To change COM2, set main function switch SW2 to ON, and COM2 will change to +5V. The factory setting is OFF and [RI]. To change COM3, simply turn SW3 to ON.

3-1-1 Removing the Rear Maintenance Cover



Main Function Switch



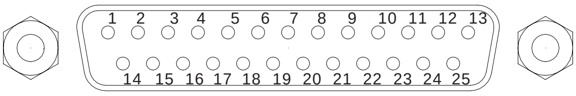
PL-6920/PL-7920<4-slot type>

(rear maintenance cover removed)

- **SW1 and SW4 are reserved. Do not change the factory settings.**
- **Be sure to confirm the interface specifications of the device you are connecting to prior to changing these settings. An incorrect setting could cause a unit malfunction or accident.**
- **Be sure to change these settings only after confirming that the PL's power supply is turned OFF. Failure to do so could cause a unit malfunction.**

5 RAS Interface

D-Sub 25 pin (Male)



Screw Size: (4-40): Inch Type

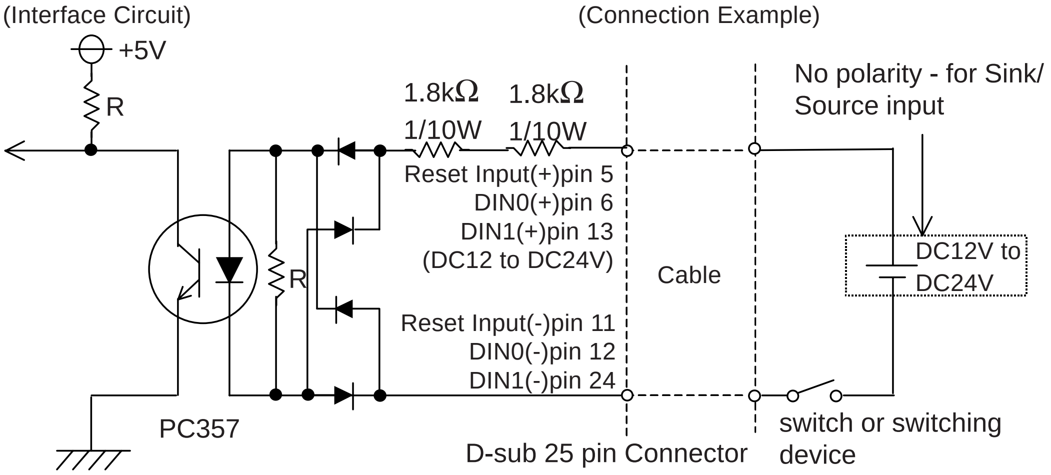
Pin No.	Signal Name	Pin No.	Signal Name
1	GND	14	GND
2	+5V (max. 100mA)	15	+5V
3	+12V (max. 100mA)	16	NC
4	NC	17	NC
5	RESET INPUT (+)	18	NC
6	DIN 0 (+)	19	NC
7	DOUT (-)	20	NC
8	DOUT (+)	21	LAMP OUT (-)
9	ALARM OUT (-)	22	LAMP OUT (+)
10	ALARM OUT (+)	23	NC
11	RESET INPUT (-)	24	DIN1 (-)
12	DIN 0 (-)	25	NC
13	DIN 1 (+)		

Be sure to use only the rated voltage level when using the No. 2 [+5V] and No. 3[12V] for external power output. Failure to do so can lead to a unit malfunction or accident.

*For RAS feature details, see **Appendix 2 RAS Feature***

■ External Input Signal (Dual use of DIN, Remote Set Input Port)

Input Voltage	DC12V to DC24V
Input Current	7mA
Operating Voltage	ON voltage: 9V (min), OFF voltage:3V (max)
Isolation Method	Via photocoupler



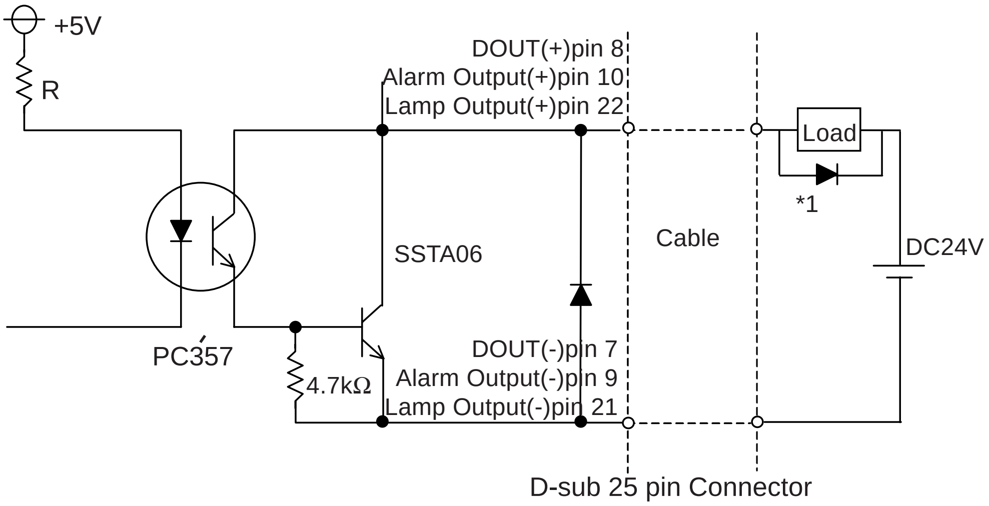
- **General Purpose Input (DIN) level must be 1.5S or longer to be detected.**
- **Be sure the voltage value between terminals is controlled via the input voltage, so that the PL is operated within its recommended range. If the input voltage exceeds this range, a malfunction or PL damage may occur.**

- **With Sink/Source input, even if the D(-), and RESET(-) are positive, and D(+), RESET(+) are negative, no problems are created. Be sure to operate the unit within the recommended voltage range.**

■ External Output Signal (DOUT, Alarm Output, Lamp Output Port)

Rated Load Voltage	DC12V to DC24V
Maximum Load Current	100mA/point
Maximum Voltage Drop between Terminals	1.5V (at 100mA load current)
Isolation Method	Via photocoupler

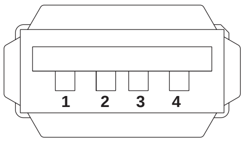
(Interface Circuit) (Connection Example)



- **Be sure to operate the unit within its maximum load current. If the maximum load current exceeds this range, a malfunction or PL damage may occur.**
- **Design your electrical system by adding the load current and voltage values to the terminal voltage. If load current value used is large, a maximum voltage of 1.5V will exist between the terminals.**
- **When connecting an induction load, be sure to connect the above drawing's protection diode(*1).**

6 USB Interface

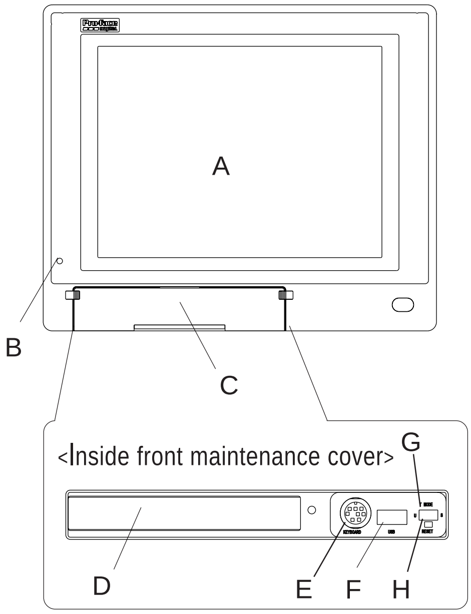
Receptacle

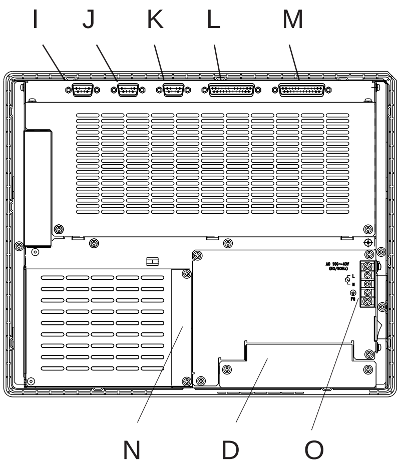


Pin No.	**Name**
1	Vcc
2	- Data
3	+ Data
4	GND

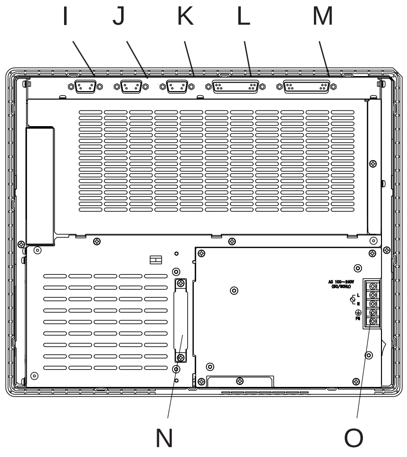
2-4 Part Names and Features

The following explanation uses the PL-6920 Series unit.





PL-6920(4-slot type)



PL-6921(2-slot type)

A:Display Area/Touch Panel

Display output area. The built-in SVGA or XGA controller supports PC compatible architecture.

B:Power Lamp LED/RAS Status Lamp

This LED indicates both the RAS monitor feature's status and the PL's power status. The status of the lamp changes according to the alarm type detected by the RAS feature.

A-2 RAS Feature

C:Front Maintenance Cover

Open this cover to connect the optional FDD unit.

D:Front Mount FDD Slot

Slot for installing the FDD unit (PL-FD200).
(only PL-6920/PL-7920<4-slot type>)

E:Keyboard Connector (KEYBOARD)

A PS/2 compatible keyboard is connected here.

F:USB Connector (USB)

USB 1.1 compatible devices can be connected here.

G:Touch Panel Interface Selector Switch (T-MODE)

U-Touch data uses the USB I/F(USB)

S-Touch data uses the serial I/F(COM4)

H:Hardware Reset Switch (RESET)

I:RS-232C Connector (COM1)

J:RS-232C Connector (COM2 - RI/+5V changeover**)**

K:RS-232C Connector (COM3 - RI/+5V changeover**)**

These RS-232C interfaces (D-sub 9 pin male connectors), allow communication with other computers and connection to peripheral devices.

L:Printer Connector (LPT1)

Centronics standard interface (D-sub 25 pin female connector), which connects a parallel device, such as a printer (supports ECP/SPP/EPP).

M:RAS Connector (RAS)

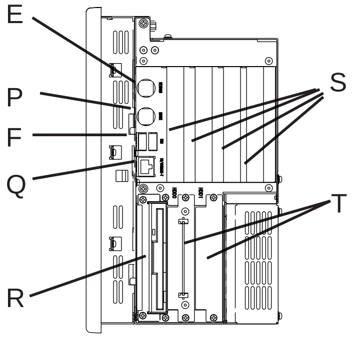
Interface for DIN, DOUT, Watchdog, and Remote Reset. (D-sub 25 pin male connector)

N:IDE I/F Cover

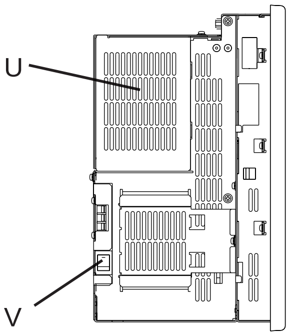
To connect the optional CD-ROM drive unit (PL-DK200), remove this cover and use this connector.

O:Power Terminals

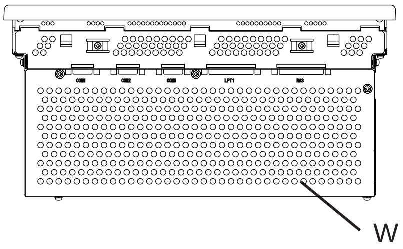
Connect the AC100V/AC240V power terminals here.



PL-6920(4-slot type)



PL-6920(4-slot type)



PL-6920(4-slot type)

P:Mouse Connector

A PS/2 compatible mouse is connected here.

Q:LAN Connector (10/100BASE-T)

Network Interface (meets IEEE802.3 standard / 10BASE-T/100BASE-TX autochangeover).

R:Side Mount FDD Slot

Houses the FDD unit.

S:Expansion Slots

T:HDD/CF Card Expansion Unit Slots

Houses an additional HDD unit, or CF Card Expansion unit.

U:Half Cover

When an optional DIM module or expansion board is used here, this cover is removed.

V:Power Switch

Use this switch to turn the PL's power ON or OFF.

W:Rear Maintenance Cover

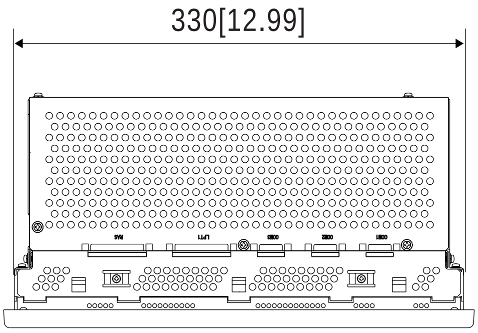
Remove this cover to install the optional DIM module, or an expansion board.

2-5 PL Dimensions

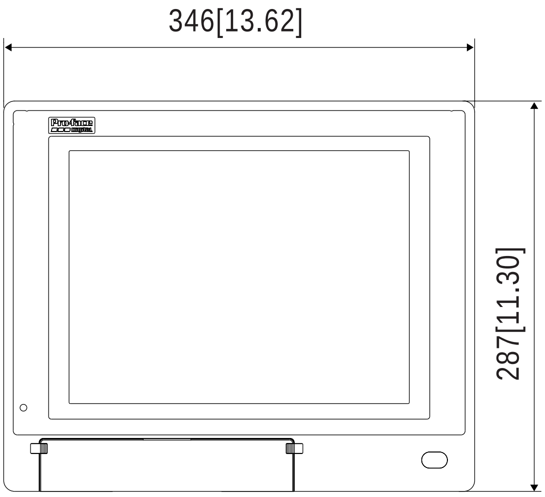
1 General Dimensions

■ PL-6920

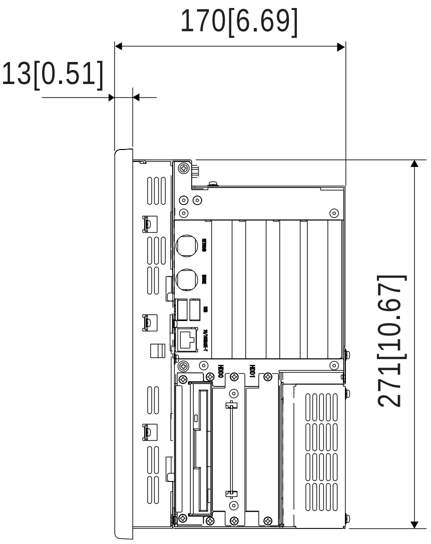
(Unit: mm/in. - excluding projections)



Top View

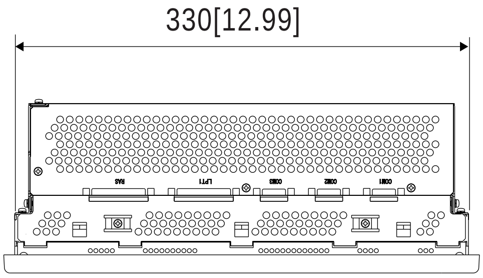


Front View

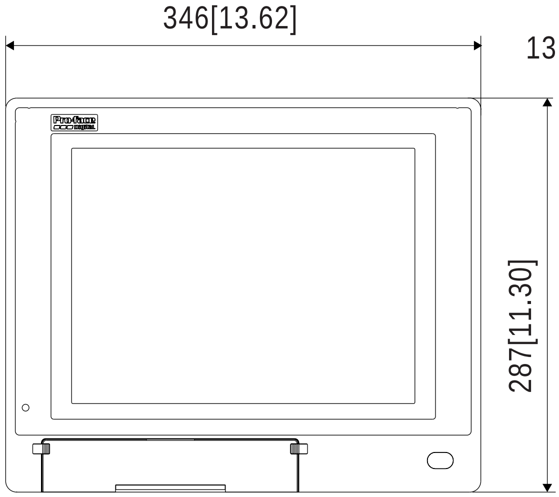


Side View

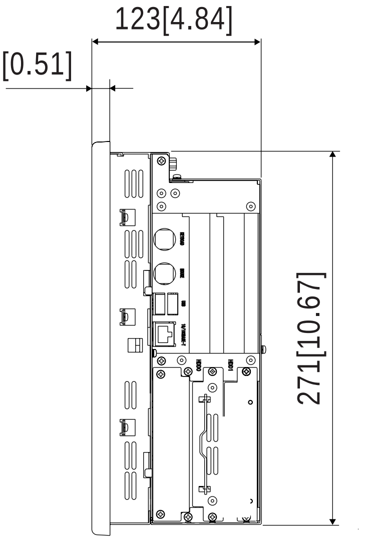
■ PL-6921



Top View



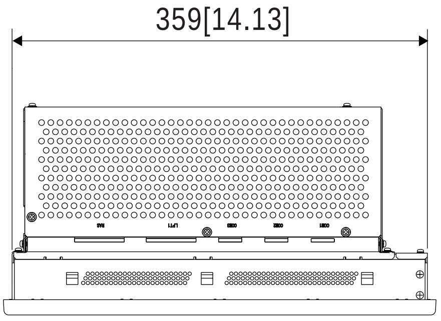
Front View



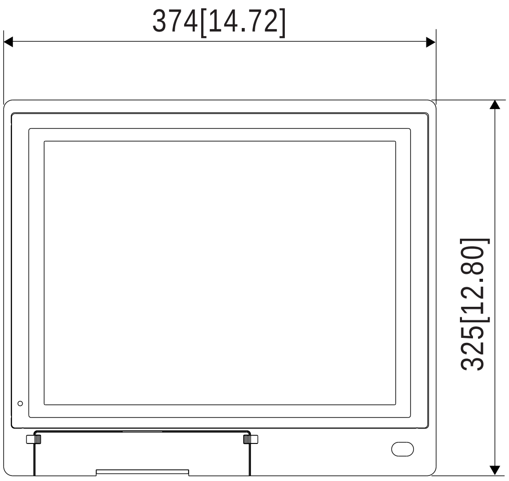
Side View

■ PL-7920

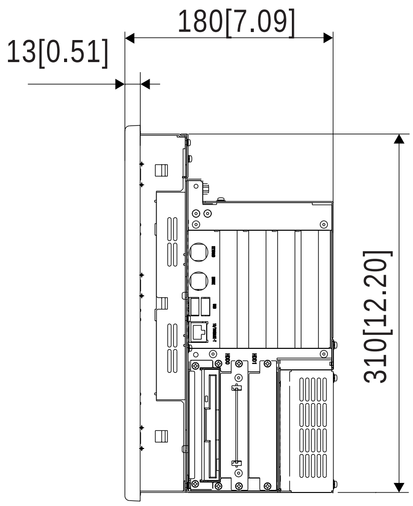
(Unit: mm/in. - excluding projections)



Top View



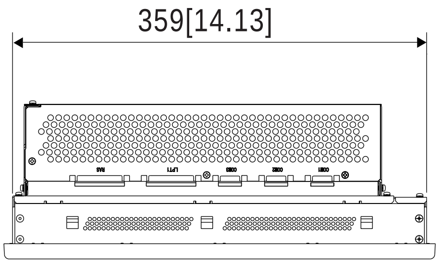
Front View



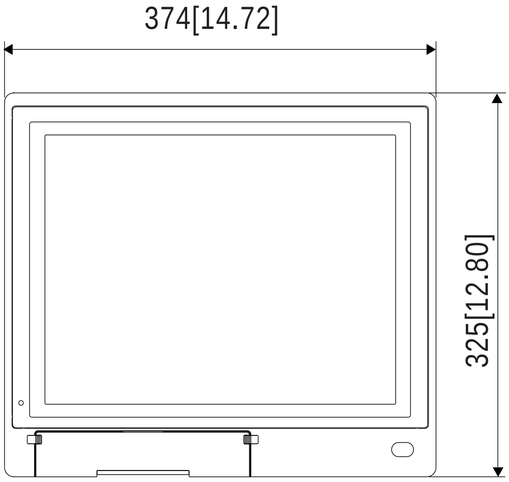
Side View

■ PL-7921

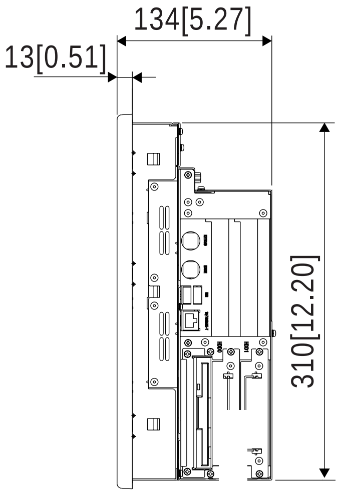
(Unit: mm/in. - excluding projections)



Top View



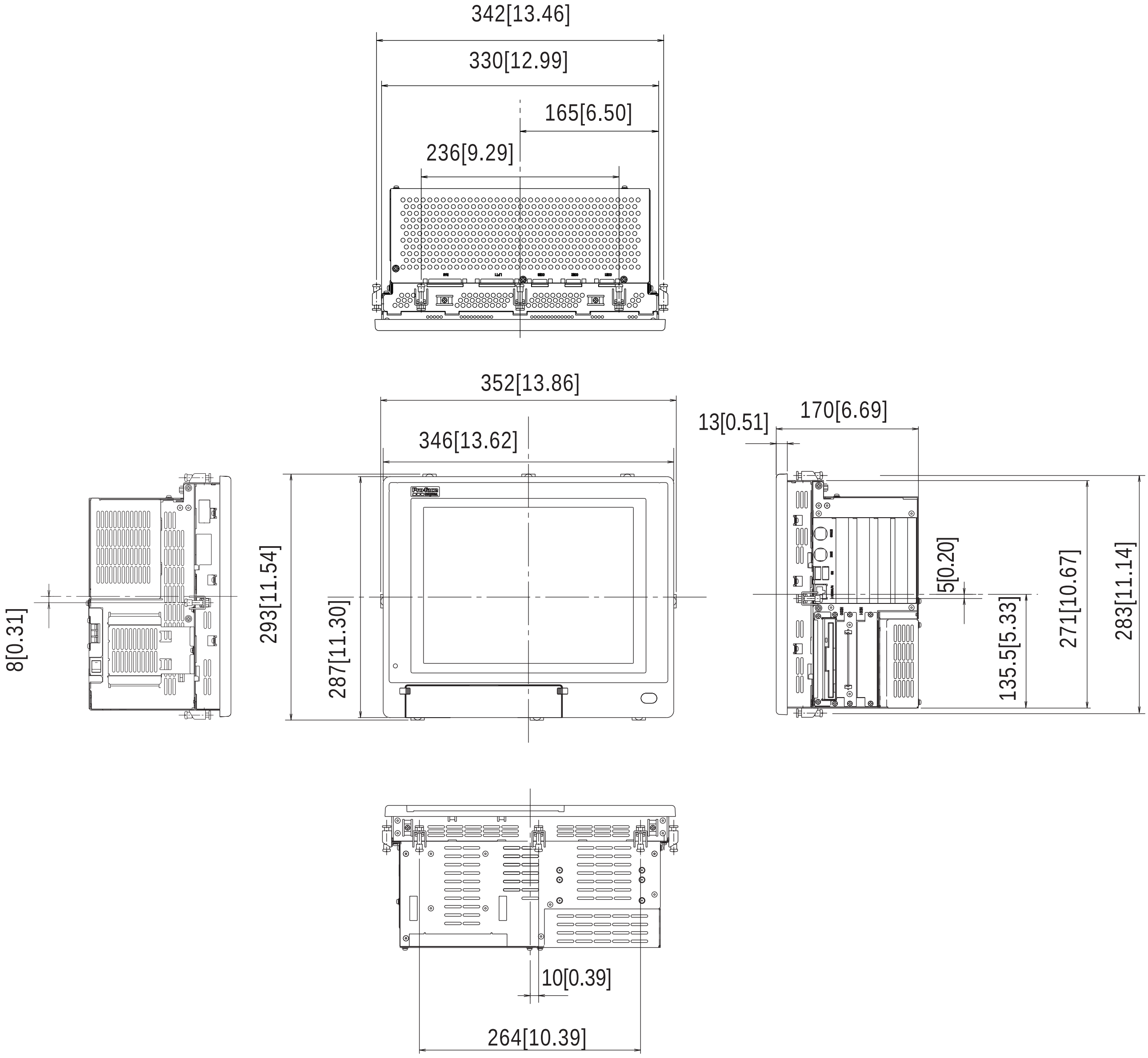
Front View



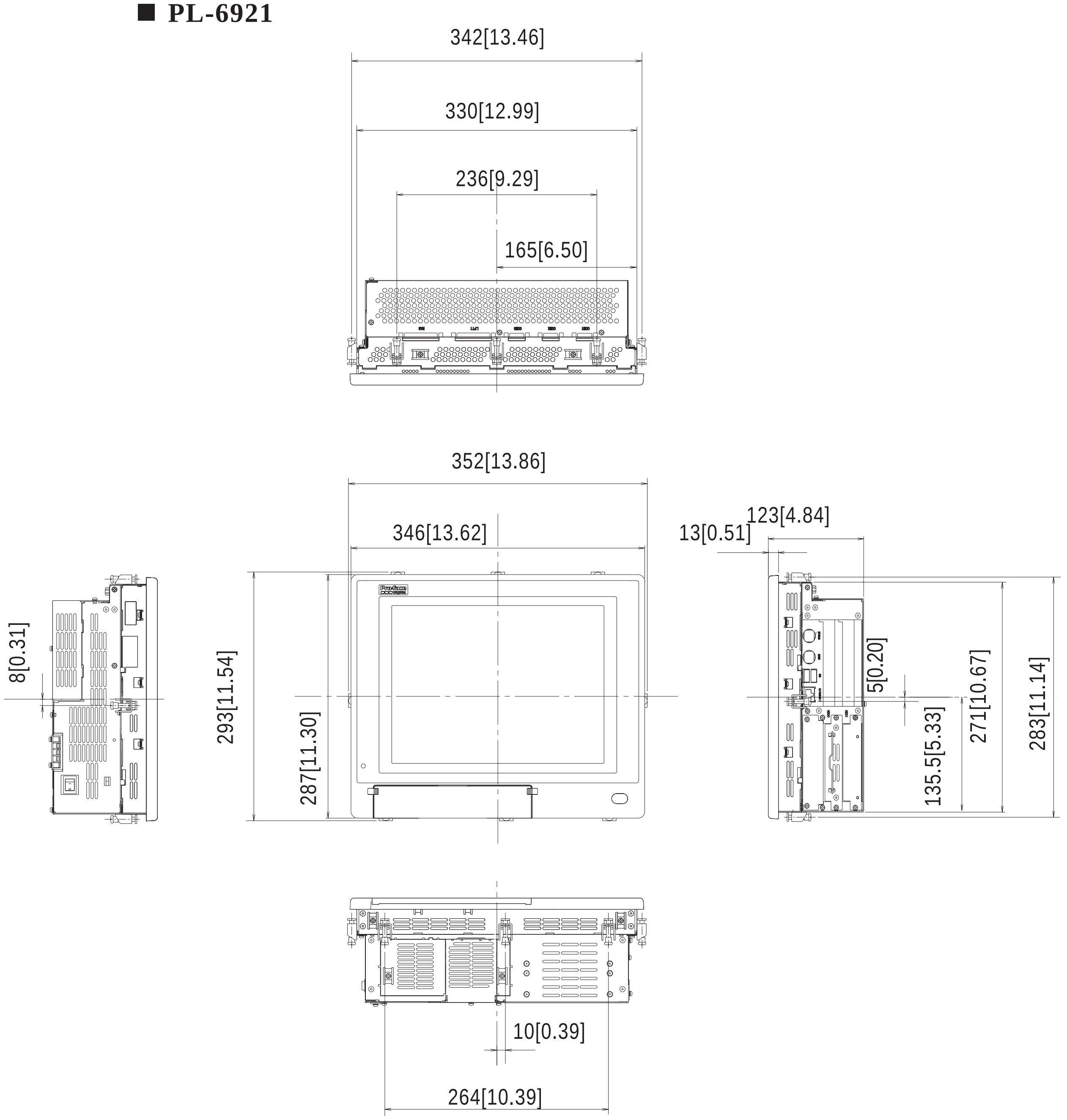
Side View

2 External Dimensions (with Installation Fasteners installed)

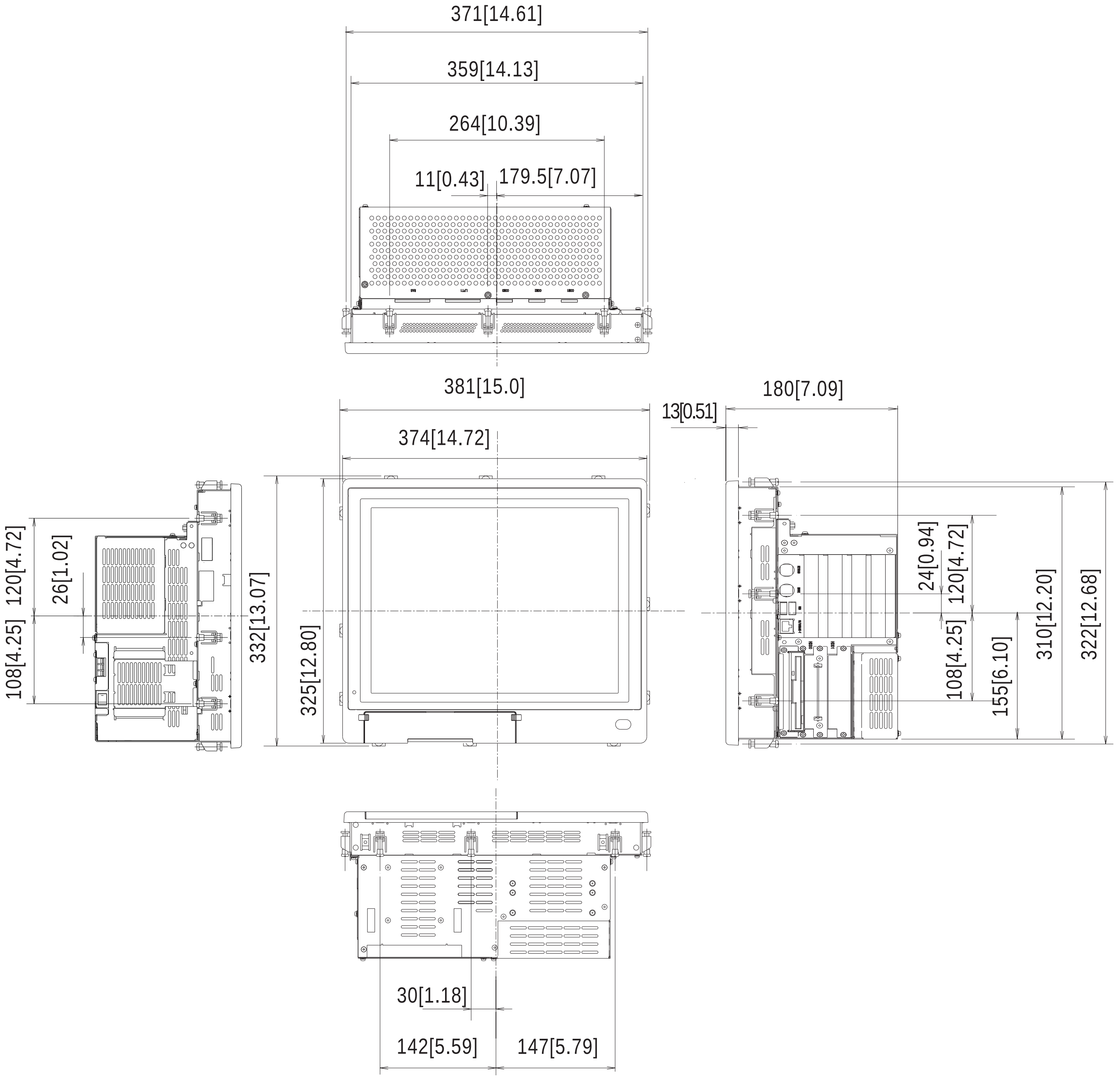
■ PL-6920



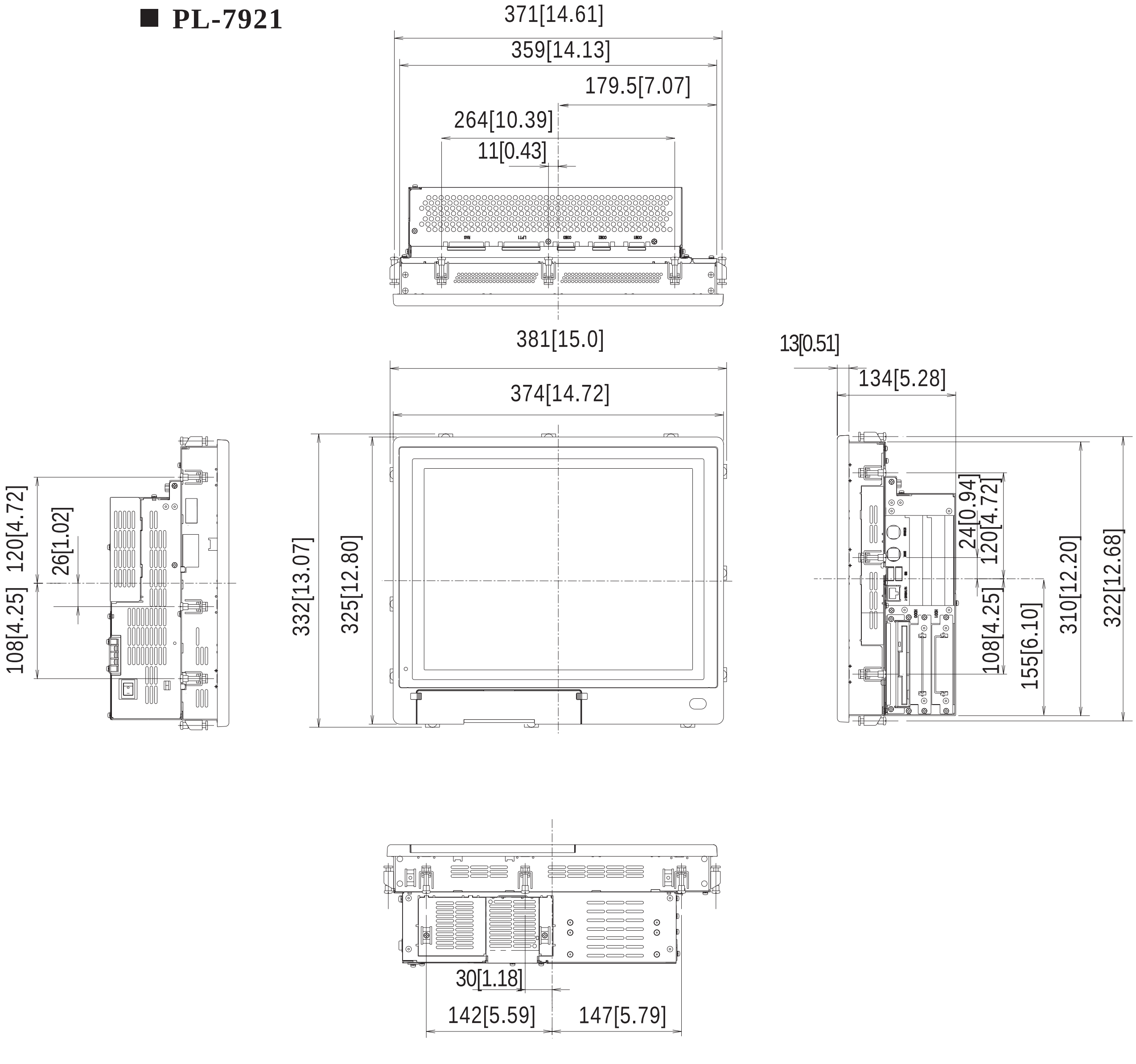
■ PL-6921


342[13.46]
330[12.99]
236[9.29]
165[6.50]
352[13.86]
346[13.62]
13[0.51]
123[4.84]
8[0.31]
293[11.54]
287[11.30]
5[0.20]
135.5[5.33]
271[10.67]
283[11.14]
10[0.39]
264[10.39]

■ PL-7920


371[14.61]
359[14.13]
264[10.39]
11[0.43]
179.5[7.07]
381[15.0]
374[14.72]
13[0.51]
180[7.09]
108[4.25]
120[4.72]
26[1.02]
332[13.07]
325[12.80]
24[0.94]
120[4.72]
108[4.25]
155[6.10]
310[12.20]
322[12.68]
30[1.18]
142[5.59]
147[5.79]

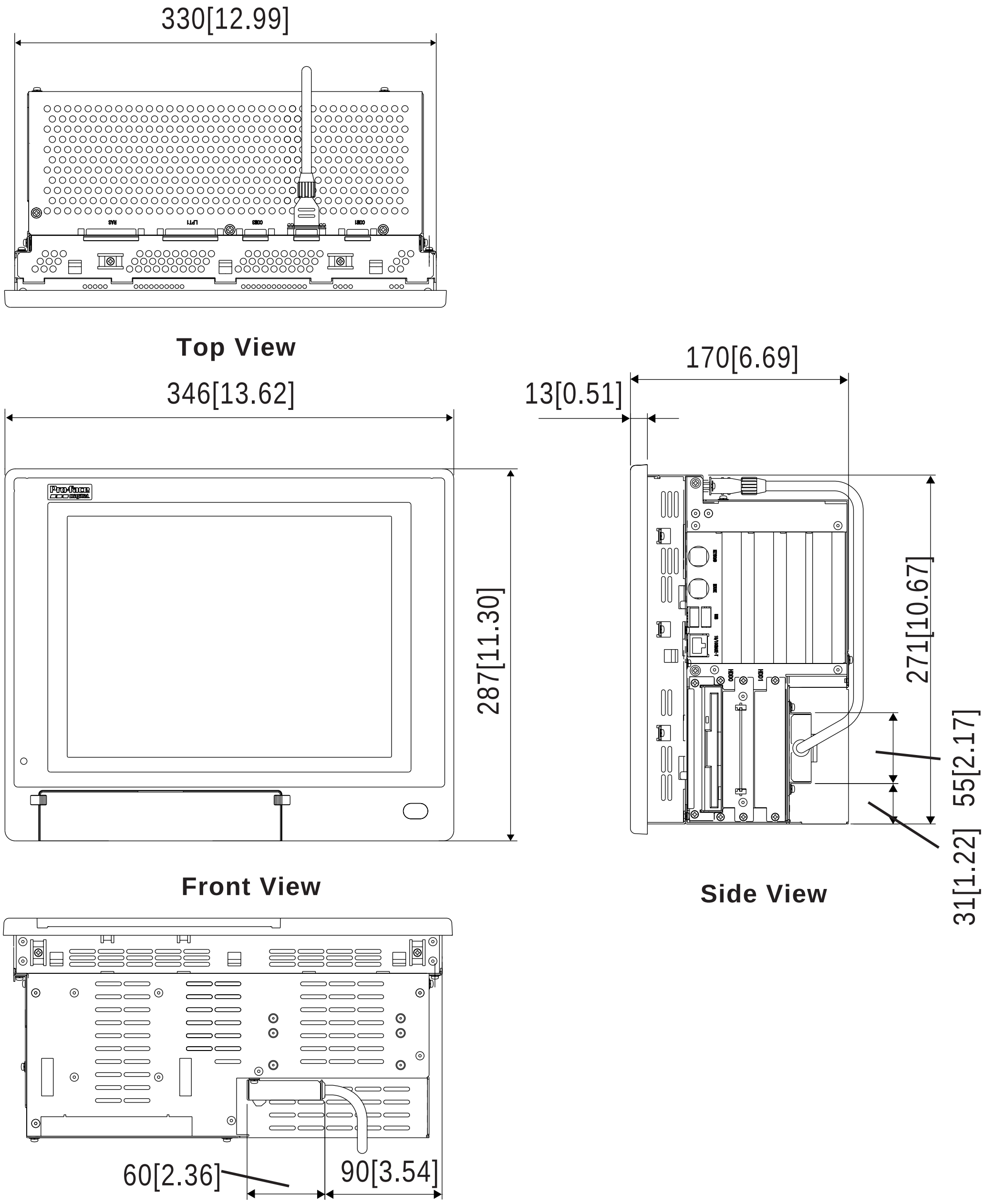
■ PL-7921


371[14.61]
359[14.13]
179.5[7.07]
264[10.39]
11[0.43]
381[15.0]
374[14.72]
13[0.51]
134[5.28]
120[4.72]
26[1.02]
108[4.25]
332[13.07]
325[12.80]
24[0.94]
120[4.72]
108[4.25]
155[6.10]
310[12.20]
322[12.68]
30[1.18]
142[5.59]
147[5.79]

3 PL and RS-232C/RS-485 Conversion Unit (PL-RC500) Dimensions

(Unit: mm/in. - excluding projections)

■ PL-6920



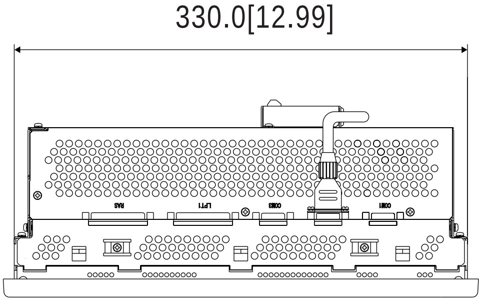
Top View

Front View

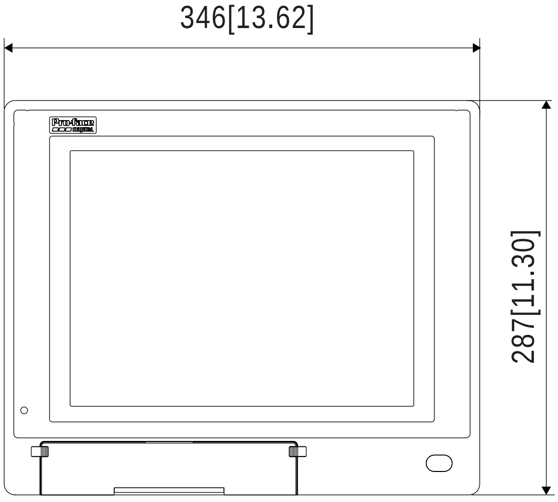
Side View

■ PL-6921

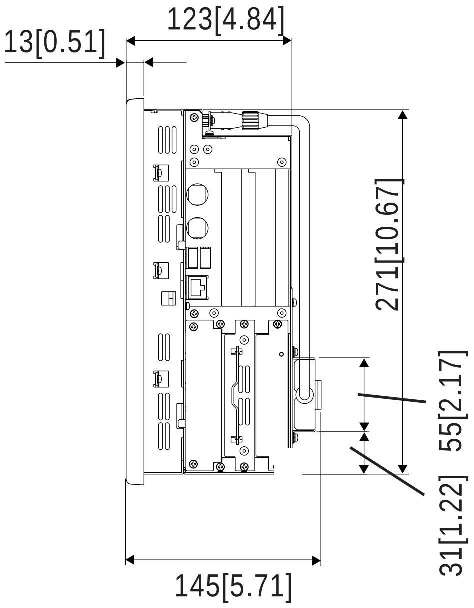
(Unit: mm/in. - excluding projections)



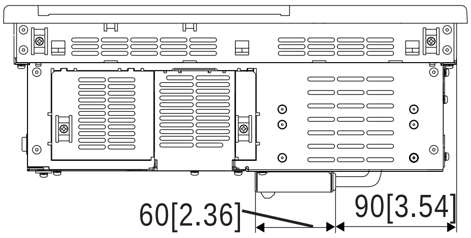
Top View



Front View

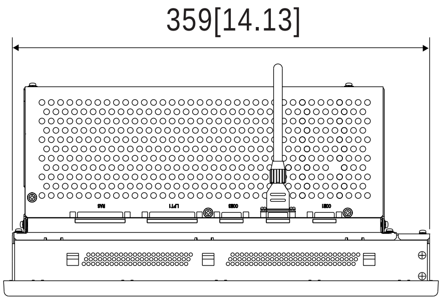


Side View

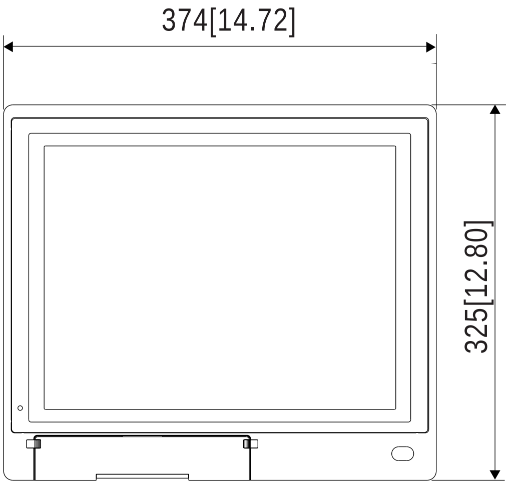


■ PL-7920

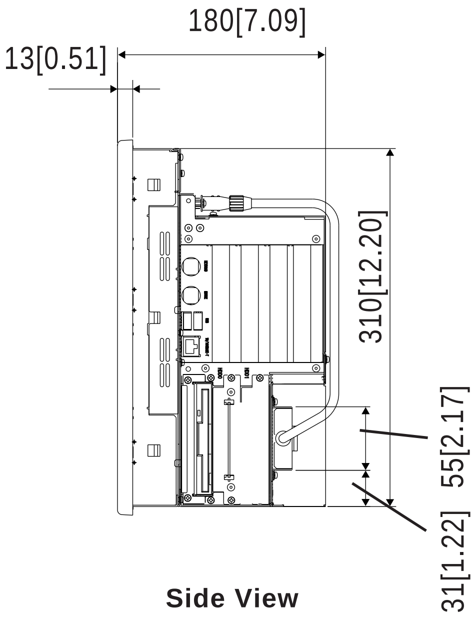
(Unit: mm/in. - excluding projections)



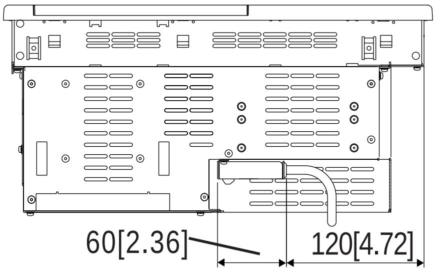
Top View



Front View

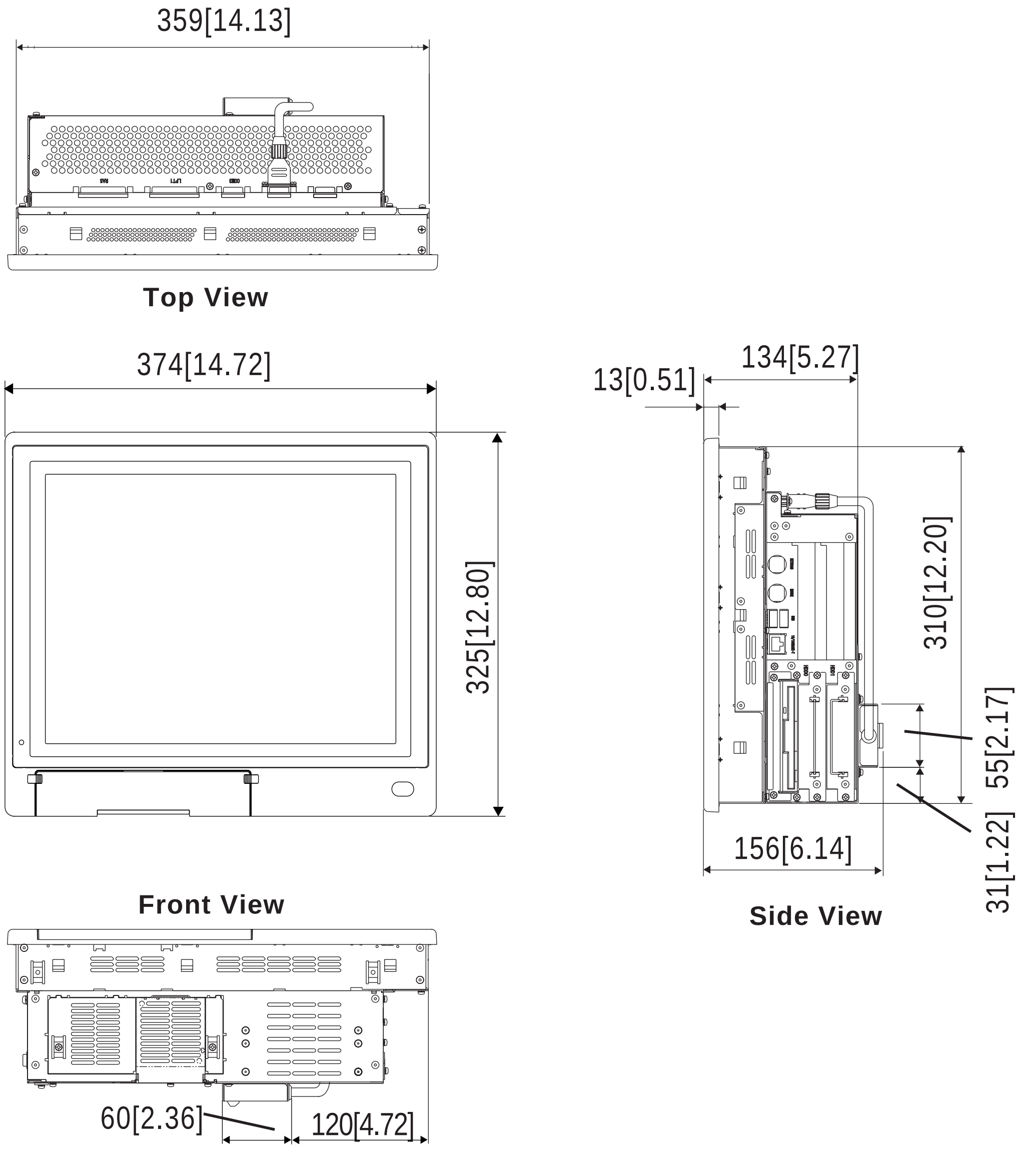


Side View



■ PL-7921

(Unit: mm/in. - excluding projections)

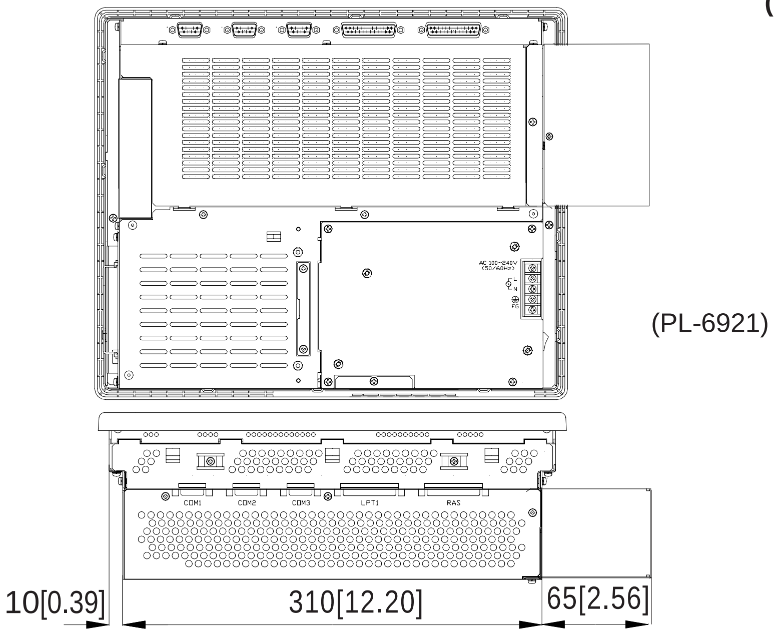


4 Full Sized Cover Attachment Dimensions

■ PL-6920 Series

The following explanation uses the PL-6921 unit.

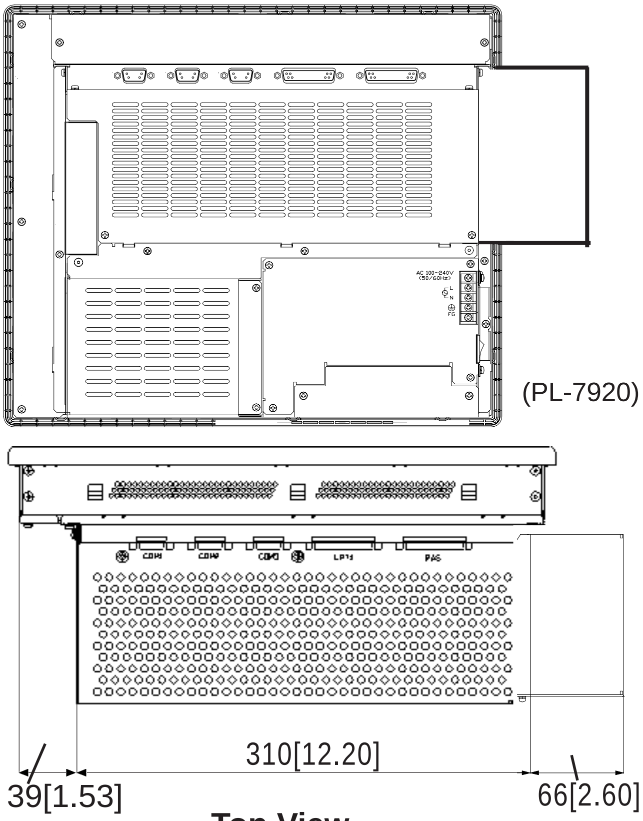
(Unit: mm/in.)



Top View

■ PL-7920 Series

The following explanation uses the PL-7920 unit.



Top View

- **Prior to installing a full-sized board and the PL's full-sized cover (PL-FC200/ PL-FC210), be sure that the PL is installed in its attachment panel/cabinet. Due to dimension differences, a full sized expansion board and PL's full-sized cover can not be attached prior to installing the PL into a panel.**
- **When using a full sized expansion board, be sure to check its dimensions and shape, since they will affect the board's environment specifications, such as for vibration, etc.**

- There are two types of full-sized covers - one for 4-slot units (PL-6920/7920) and one for 2-slot units (PL-6921/7921).

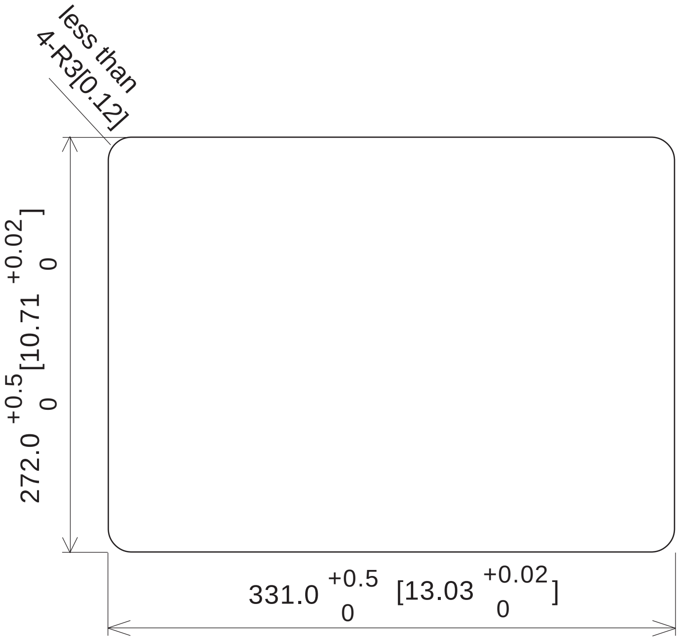
  2-slot type PL-FC200

  4-slot type PL-FC210

☞ *1.3 Optional Items*

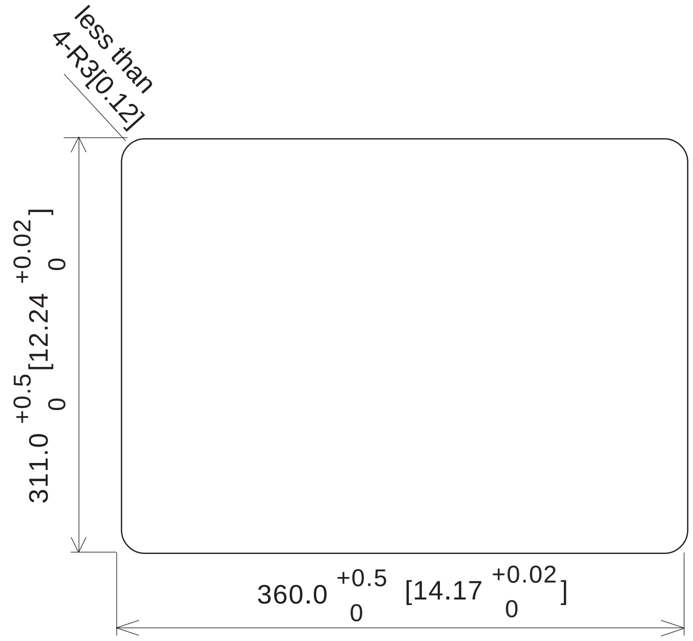
5 Panel Cut Dimensions

■ PL-6920 Series



■ PL-7920 Series

(Unit: mm/in.)

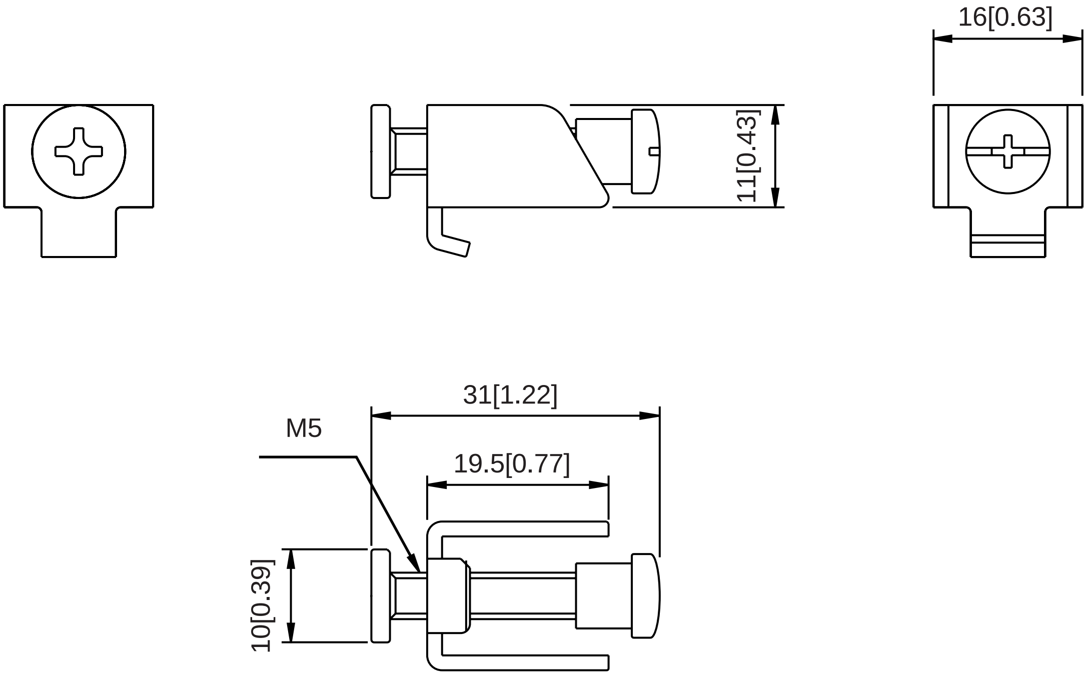


CAUTION!

- **Be sure the thickness of the installation panel is from 1.6 to 10 mm.**
- **All panel surfaces used should be strengthened. Especially, if high levels of vibration are expected and the PL's installation surface (i.e. an operation panel's door, etc.) can move (i.e.open or close) due consideration should be given to the PL's weight.**
- **To insure that the PL's water resistance is maintained, be sure to install the PL into a panel that is flat and free of scratches or dents.**
- **Be sure all installation tolerances are maintained to prevent the unit from falling out of its installation panel.**

6 Installation Fasteners

(Unit: mm/in.)



Chapter 3

Installing Optional Units and Expansion Boards

The User can install a variety of optional units and expansion boards made by Digital in the PL, as well as a number of commercially available ISA-bus compatible boards. This chapter describes both the products that can be installed in the PL and how to install them.

3-1 Installing Options and Expansion Boards

The following explanation pages describe the installation and removal procedures for the PL's DIM module (PL-EM500/PL-EM128), FDD unit (PL-FD200/PL-FD210), HDD unit (PL-HD220/PL-HDX920-W2K/ML), expansion boards, and the CD-ROM drive unit (PL-DK200) and the electric FAN Unit.

For information about the installation of other option units, please refer to those unit's individual [Installation Guide].

⚠ **WARNING**

To avoid electric hazards, be sure to turn the PL's power OFF before installing any optional units or expansion boards.

CAUTION!

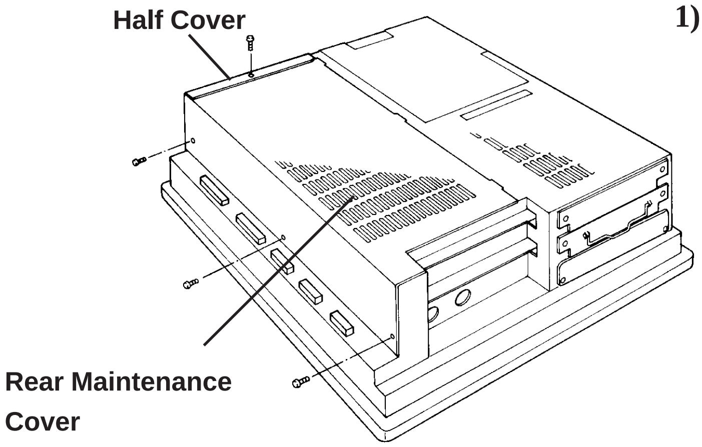
Use a screw driver to loosen or tighten the screws. Be careful not to over-tighten any screws, since it may damage the equipment.

Be careful when removing or inserting any screws inside the body of the PL.

1 Removing the Rear Maintenance Cover

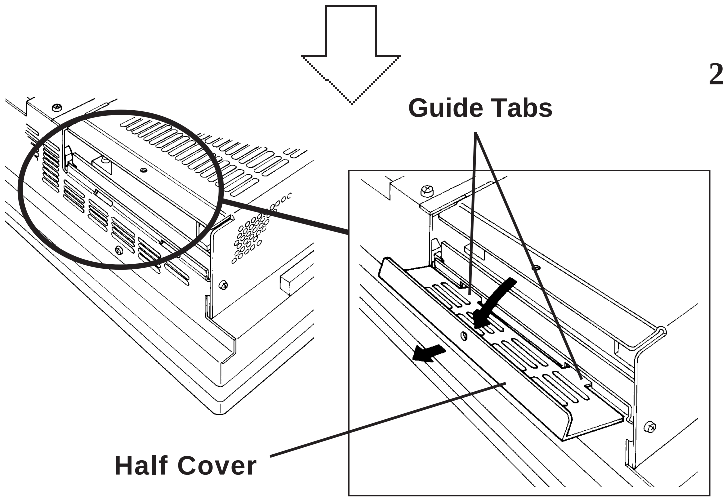
CAUTION! **Handle the rear maintenance cover with care, since it is made of aluminum and is easily bent.**

■ PL-6921/PL-7921 (2 slot type)

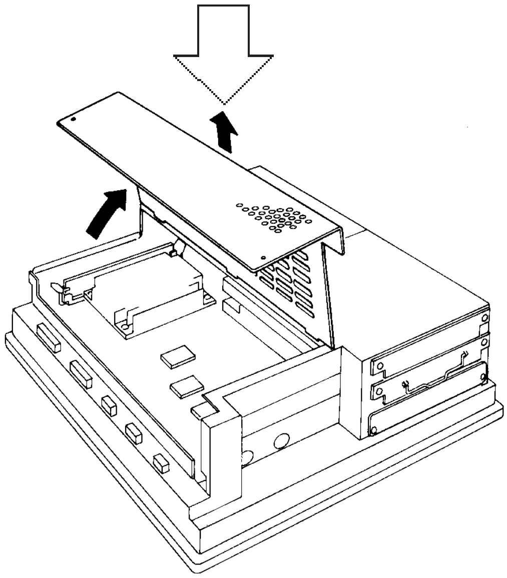


1) Unscrew the four attachment screws used to hold the rear maintenance cover and half-cover in place.

CAUTION! **Remove the half cover, and then the rear maintenance cover.**

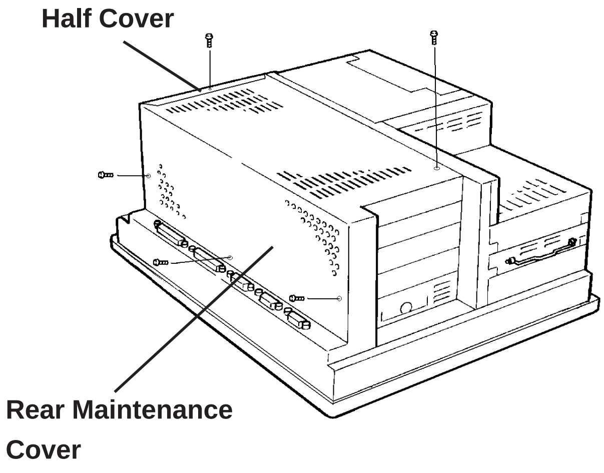


2) Pivot the half cover open and lift up slightly to free the guide tabs. Next, remove the half-cover.



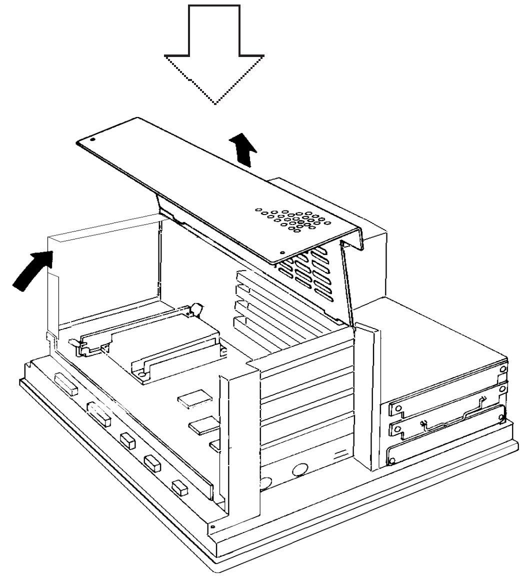
3) Remove the rear maintenance cover.

■ PL-6920/PL-7920 (4 slot type)

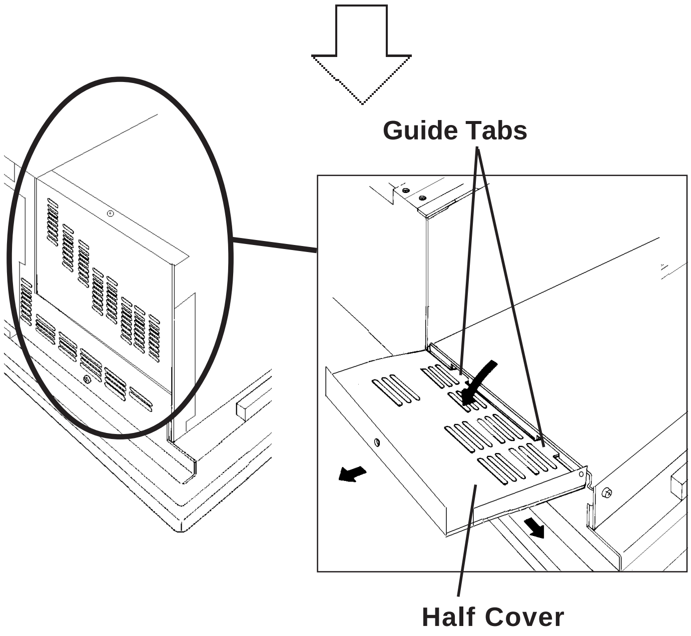


1) Unscrew the attachment screws used to hold the rear maintenance cover(4) and half cover(1) in place.

CAUTION! Remove the rear maintenance cover, and then the half cover.



2) Remove the rear maintenance cover.

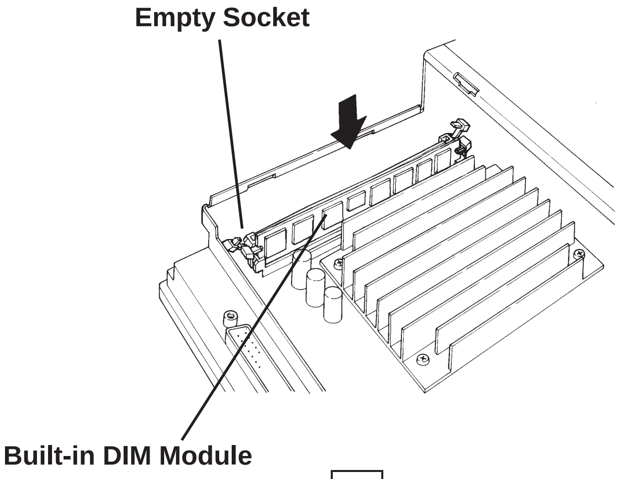


3) Pivot the half cover open and lift up slightly to free the guide tabs. Next, remove the half cover.

2 Installing the DIM Module (PL-EM500 / PL-EM128/ PL-EM256)

- **Since DIM module sockets are easy to break, be sure to install the DIM module very carefully.**
- **Do not change the factory installed DIM module's socket position.**

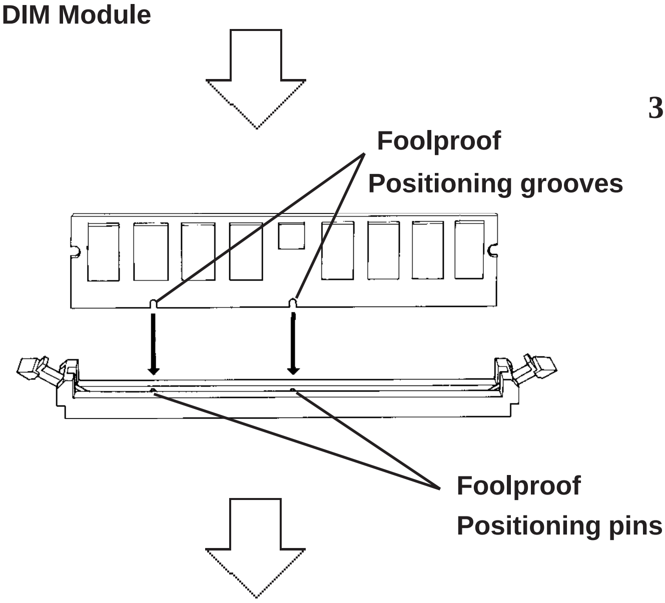
The PL comes with a single, 128MB DIM module pre-installed. There is one more empty socket that can be used and the procedures that follow describe how to install a DIM module in that empty socket.



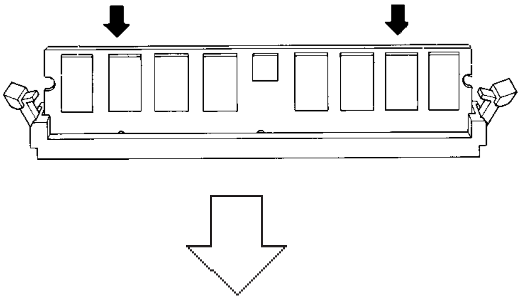
1) **Replace the rear maintenance cover and the half cover.**

3-1-1 Removing the rear maintenance cover

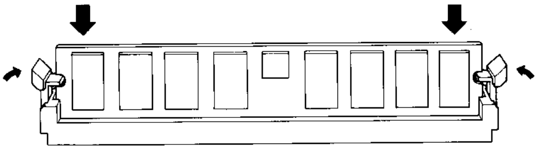
2) **Install the DIM module for expansion on the empty socket.**



3) **Adjust the foolproof positioning grooves so that they align with the foolproof positioning pins.**



4) **Insert the DIM module into the DIM module socket.**

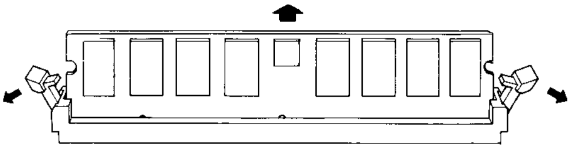


5) Push the DIM module down until the side stoppers lock.

6) Replace the rear maintenance cover and the half cover and secure them in place with the attachment screws.

◆ To Remove the DIM Module

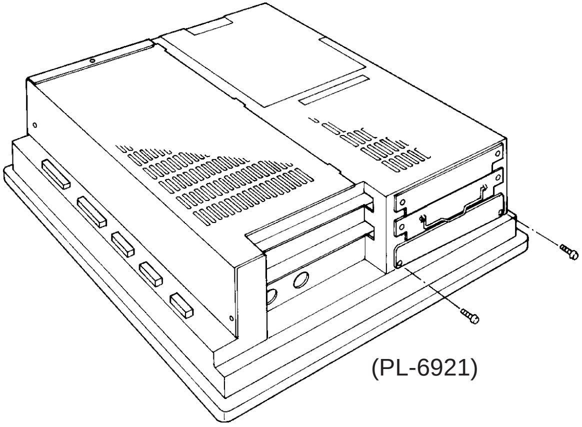
When removing the module from the socket, press down on the socket's ejector tabs to release the module.



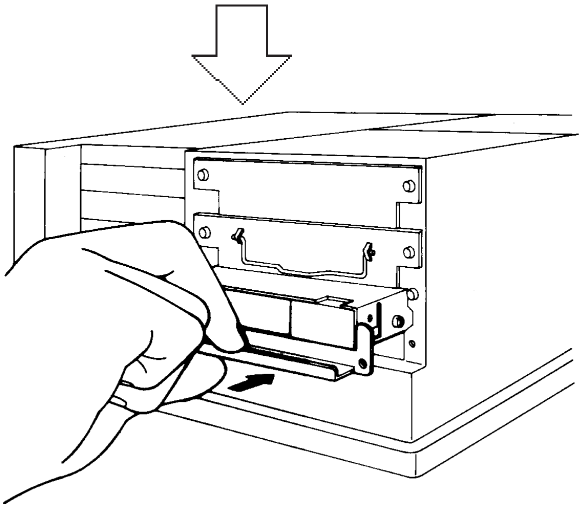
3 Installing the FDD Unit (PL-FD200)

NOTE

- The PL-FD200 and the PL-FD210 cannot be used at the same time.



1) Remove the two(2) attachment screws from the lowermost Expansion Slot Cover, and remove the cover.

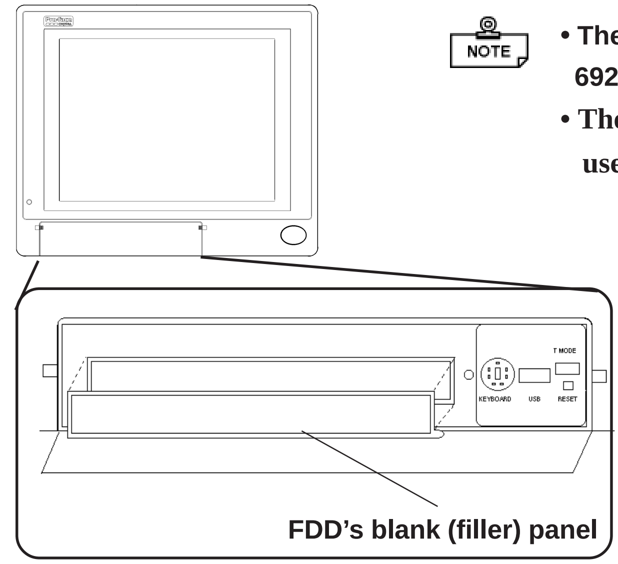


2) Insert the FDD unit so that its guide grooves fit the chassis guide ways. Push the unit in until its rear connector is connected securely.

3) Fix the unit in place with its two(2) attachment screws.

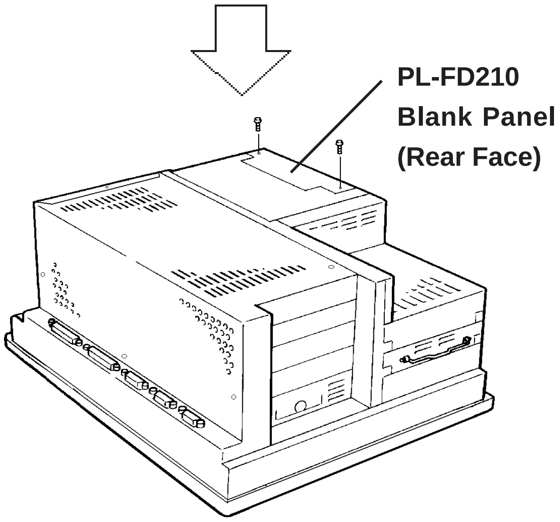
4 Installing the FDD Unit (PL-FD210)

- **The PL-FD210 can only be installed in the PL-6920/PL-7920 (4-slot type) unit.**
- **The PL-FD200 and the PL-FD210 cannot be used at the same time.**



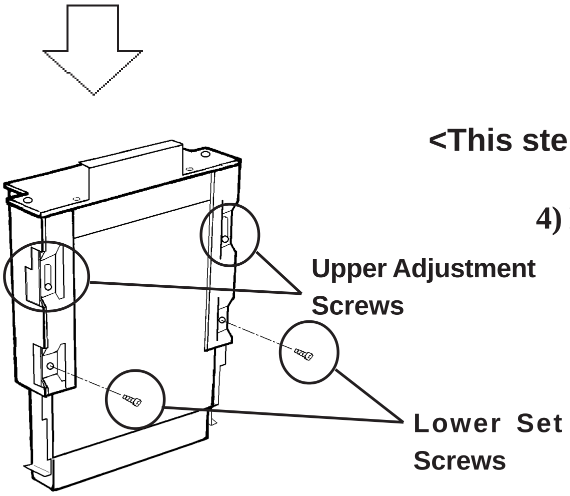
1) Open the front maintenance cover and remove the FDD’s blank (filler) panel.

2) Close the front maintenance cover.

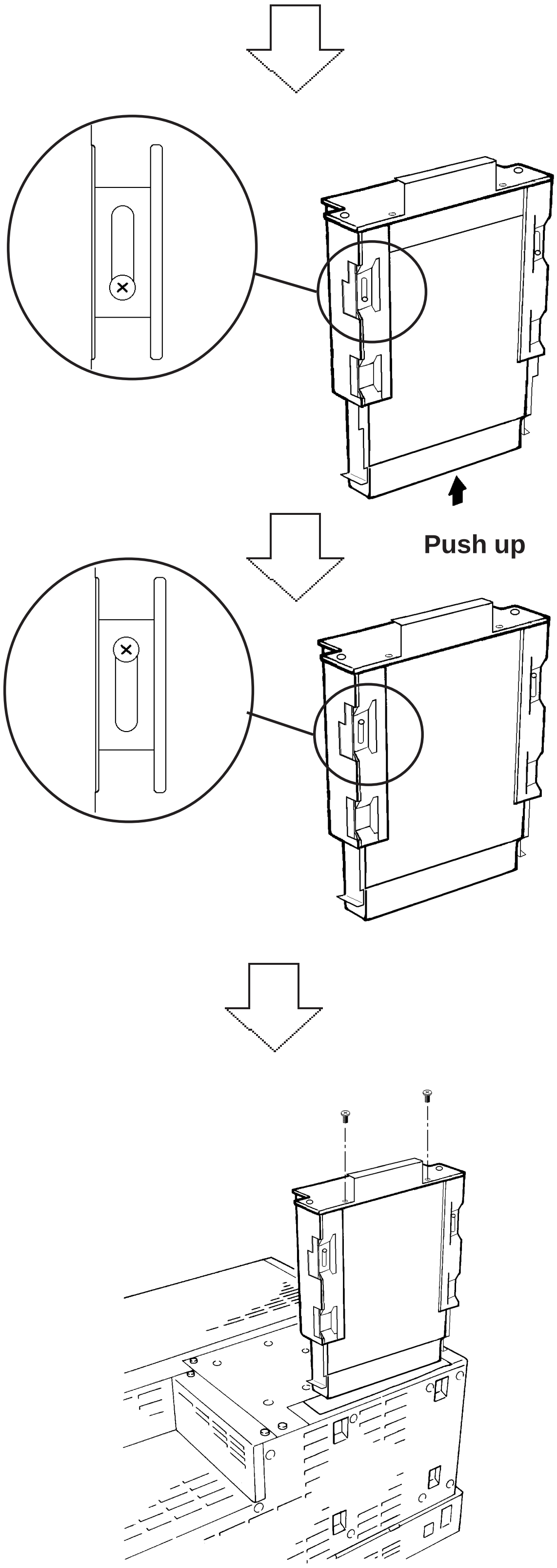


3) Unscrew the two(2) attachment screws from the PL’s Blank Panel, and remove the blank panel.

<This step is only for the PL-6920 Series>



4) Loosen the FDD unit chassis two (2) upper adjustment screws, and remove the unit’s two (2) lower set screws to allow the FDD unit to slide forward.



<This step is only for the PL-6920 Series>

5) Push on either end of the FDD unit to compress the unit to its shortest length. Then, re-tighten the two attachment screws to secure the unit in place.

6) Insert the FDD unit so that its guide grooves align with the chassis holder guideways. Push the unit into the PL holder until its connector (middle of unit) is connected securely.

CAUTION!

When installing the PL-FD210, insert it slowly into the PL's installation opening and be sure it is securely attached.

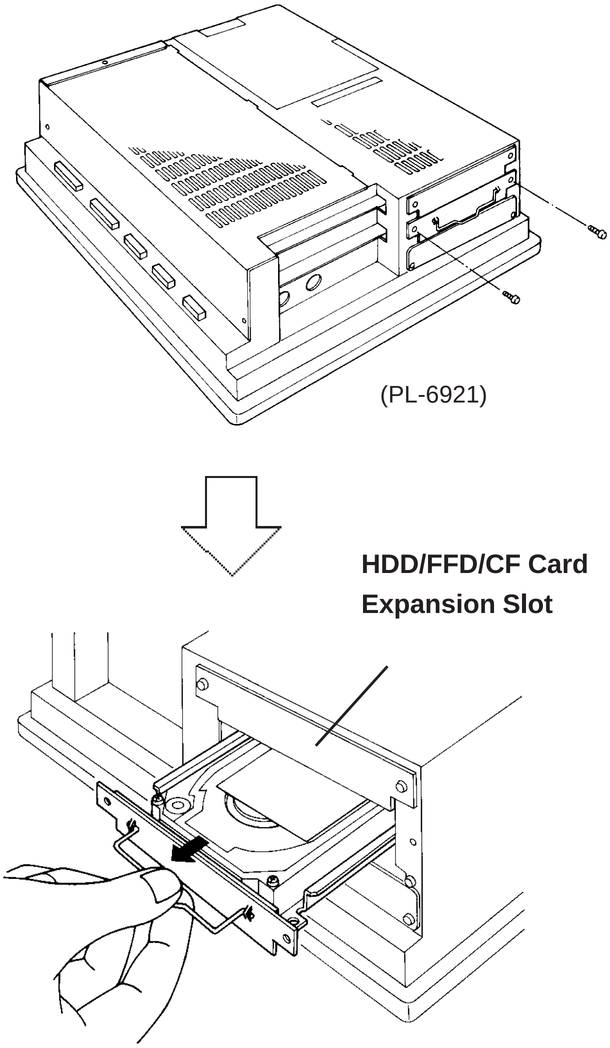
7) Fix the unit in place with its two(2) attachment screws.

8) Last, replace the Blank Panel (removed in step 3).

5 Removing / Installing the HDD Unit

(PL-HD220/PL-HDX920-W2k/ML)

NOTE The following procedure is also applicable to the FFD unit(PL-FF210) and the PL-CF200 CF Card Unit.



CAUTION! Since the HDD units are precision instruments, be sure not to jar or shake them unnecessarily.

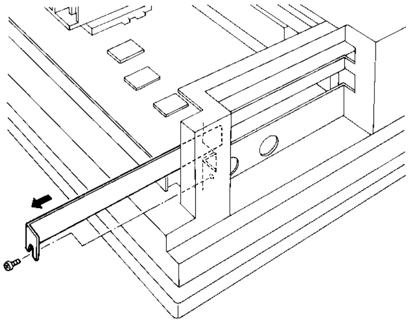
1) Remove the two(2) attachment screws from the middle Expansion Slot Cover.

2) Grasp the HDD unit's handle and pull the unit slowly out of the PL. Be sure you do not damage the unit.

3) Insert the HDD unit you selected inside the guideways of the PL and push it in until its rear connector is securely connected.

4) Fix the unit in place with its two(2) attachment screws.

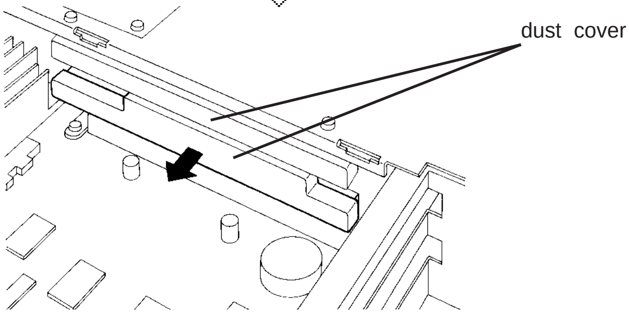
6 Installing an Expansion Board



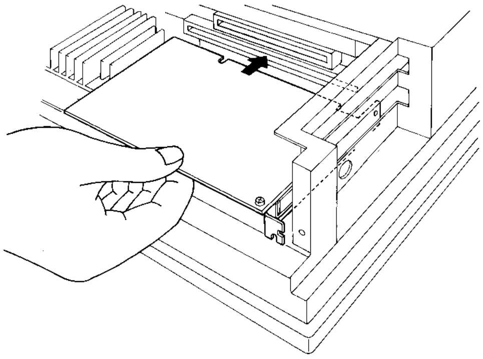
1) Remove the PL's rear maintenance cover and the half cover.

3-1-1 Removing the rear maintenance cover

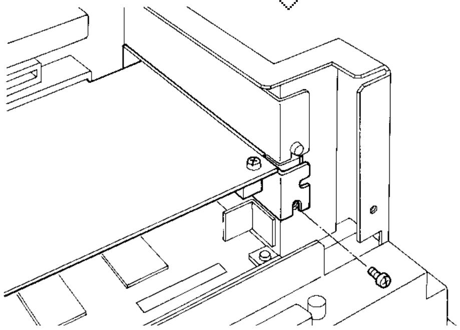
2) Unscrew the attachment screw from the expansion slot cover, and remove the cover.



3) Remove the dust cover.



4) Insert the expansion board into the expansion slot.

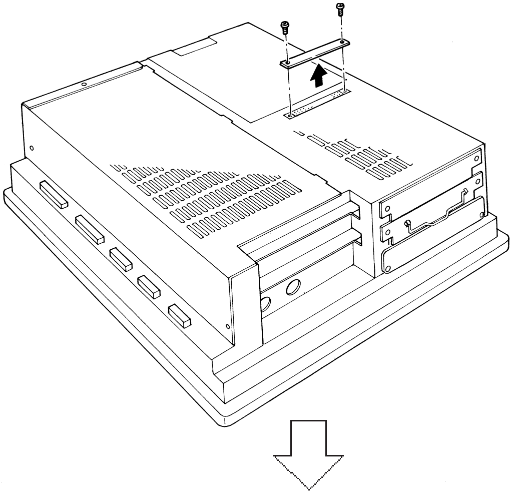


5) Fix the expansion board's metal positioning strip in place with its attachment screw.

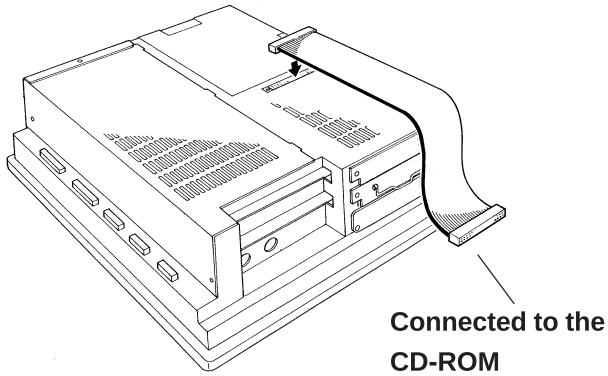
6) Last, secure the rear maintenance and half covers in place with their attachment screws.

7 Connecting the CD-ROM Unit (PL-DK200)

■ PL-6921/PL-7921(2-slot type)



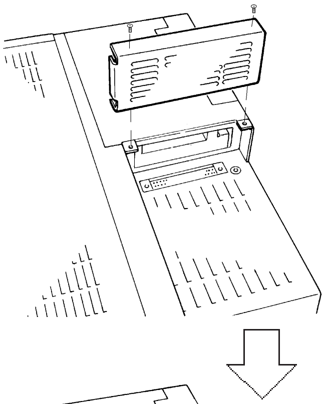
1) Unscrew the two(2) attachment screws on the IDE I/F cover, then remove the cover.



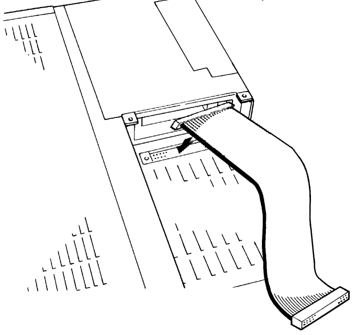
2) Connect the CD-ROM unit's cable (PL-X920 units) to the IDE I/F.

CAUTION! Be sure that the cable is securely connected before turning ON the power switch.

■ PL-6920/PL-7920 (4-slot type)



1) Unscrew the two(2) attachment screws on the IDE I/F cover, and remove the cover.



2) Connect the CD-ROM unit's cable (PL-X920 units) to the IDE I/F.

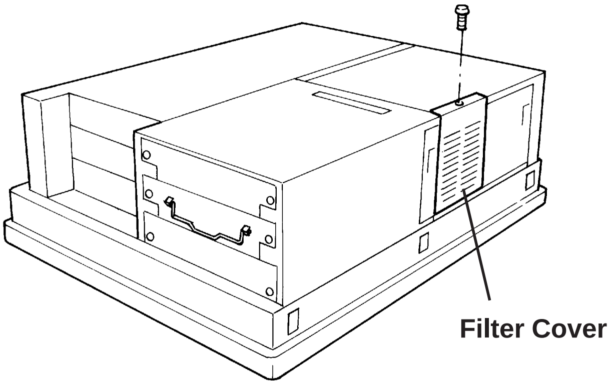
CAUTION! Be sure that the cable is securely connected before turning ON the power switch.

8 Removing the Electric Fan Unit

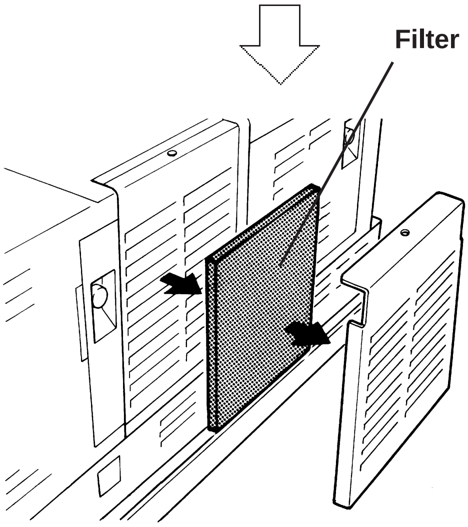
The PL can be operated without the bottom face electric fan unit. The user should, however, be aware that doing so (i.e. removing the fan unit) will cause the PL's ambient temperature to rise. See "*2-1-2 Environment Specifications*"

The electric Fan cannot be removed from the PL units (PL692*-T42 / PL792*-T42) which CPU is 1GHz.

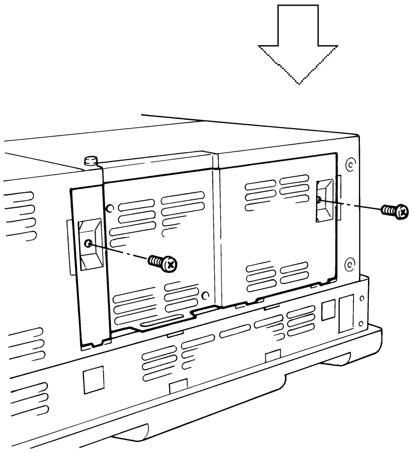
■ PL-6921/PL-7921 (2-slot type)



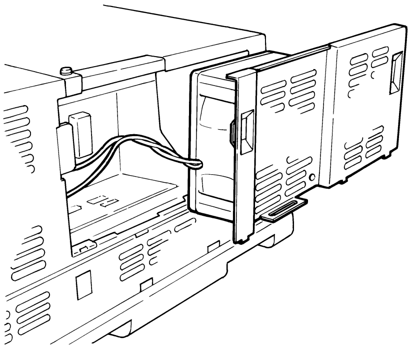
1) Remove the attachment screw on the filter cover, and remove the cover.

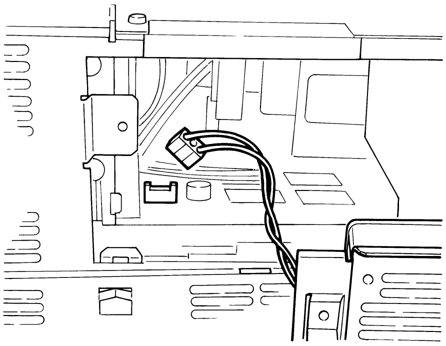


2) Remove the filter.

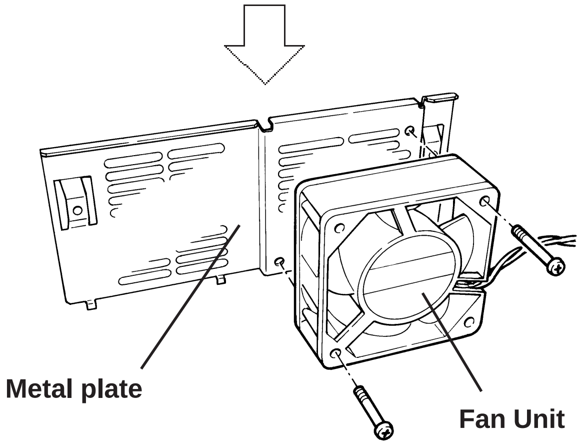


3) Remove the two(2) attachment screws on the fan unit, and take out the fan unit.





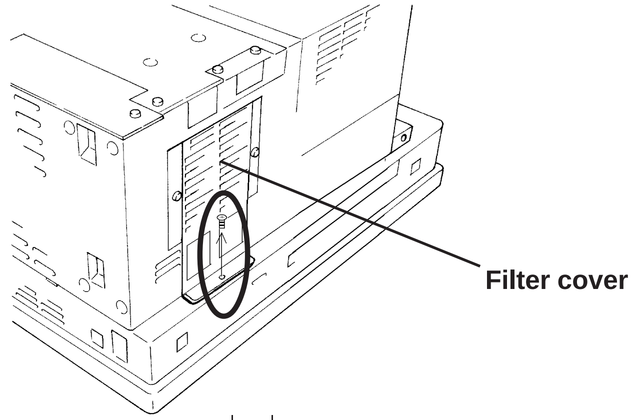
4) Unplug the power cable connector from the electric fan unit.



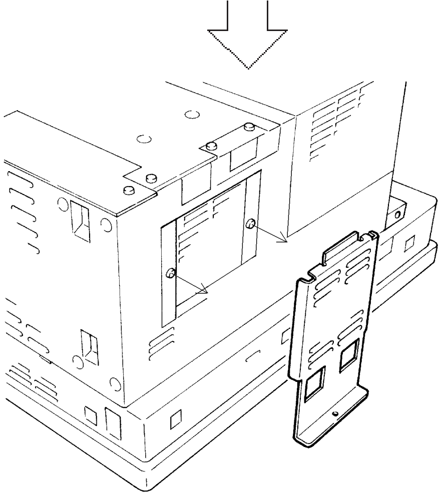
5) Unscrew the fan’s two(2) attachment screws, then remove the fan unit from its metal attachment plate.

6) Reattach the metal plate and the filter cover to the PL.

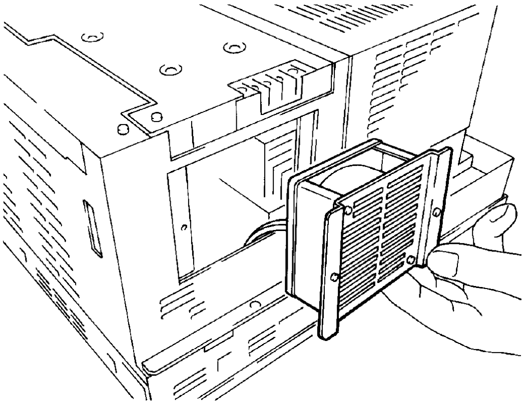
■ PL-6920/PL-7920 (4-slot type)



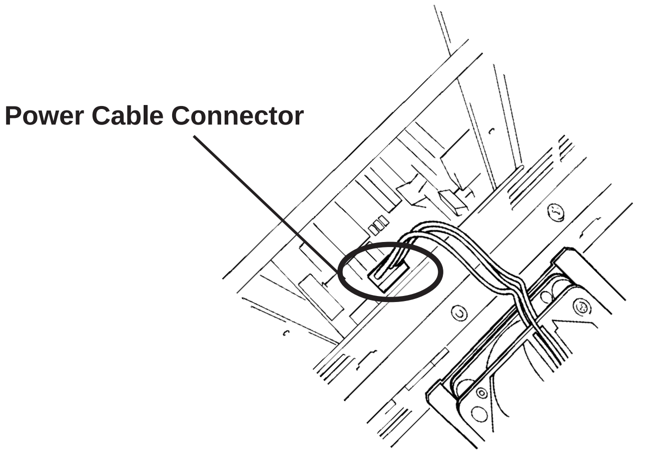
1) Remove the filter cover’s attachment screw.



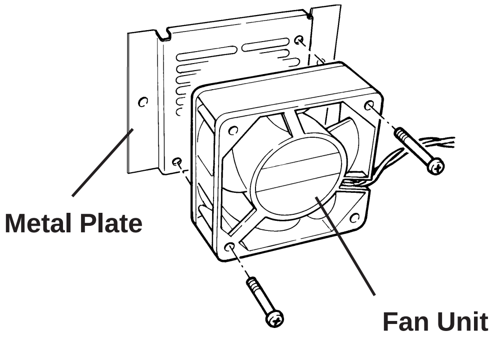
2) Remove the filter cover from the fan unit, and then remove the fan unit’s two(2) attachment screws.



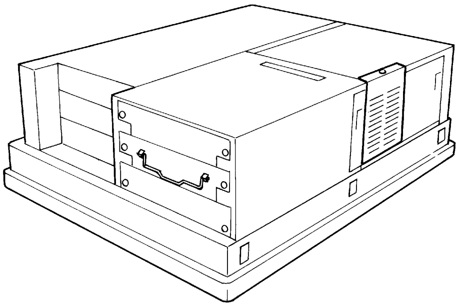
3) Remove the fan unit from the PL.



4) Disconnect the fan unit’s power cable connector from the PL.



5) Unscrew the two(2) fan unit’s attachment screws and then remove the fan unit from the metal plate holding the fan unit.



6) Re-attach the metal plate and the filter cover to the PL.

MEMO

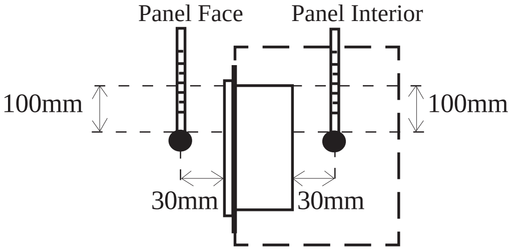
Chapter

4 Installation and Wiring

This chapter explains how to install and wire the PL Series, as well as the cautions required both before and during installation.

4-1 Installing the PL

1 Installation Caution

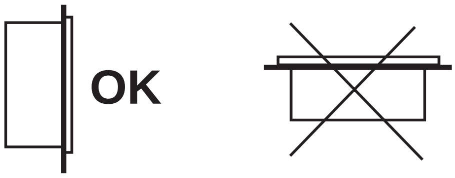


1) Temperature Related Cautions

- The PL should be installed in a vertical position, and forced air cooling should be used, instead of natural air circulation.
- To prevent a machine breakdown, be sure to use the PL within the allowable temperature range as below-listed. Please check “Ambient Operating Temperature” in the position drawn on the left. (“Ambient Operating Temperature” indicates both the panel interior and panel face temperature.)

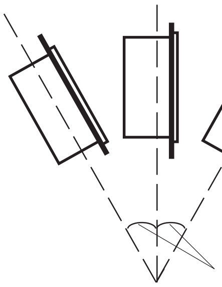
Ambient Operating Temperature	PL692*-T41 (CPU:700MHz)	Using Fan		5 °C to 50 °C (With HDD attached)
		Not using fan [*1]		5 °C to 40 °C (With HDD attached)
	PL792*-T41 (CPU:700MHz)	Cabinet Interior	Using Fan	5 °C to 50 °C (With HDD attached)
			Not using fan*1	5 °C to 40 °C (With HDD attached)
		Panel face	5 °C to 40 °C	
	PL692*-T42 (CPU:1GHz)	Using Fan		5 °C to 45 °C (With HDD attached)
		Not using fan [*1]		Must use fan
	PL792*-T42 (CPU:1GHz)	Cabinet Interior	Using Fan	5 °C to 45 °C (With HDD attached)
			Not using fan*1	Must use fan
		Panel face	5 °C to 40 °C	

***1** *When the PL’s internal electric fan is removed.*



2) Installation Positioning Cautions

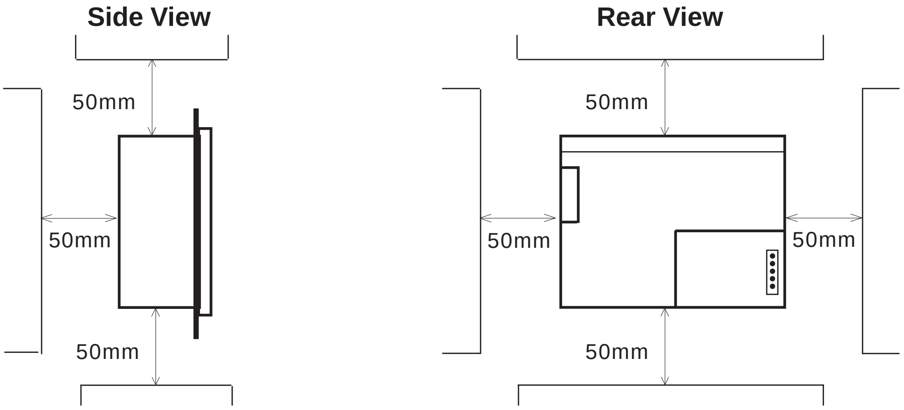
Be sure to install the panel in an upright (vertical) position.



Also, be sure that the panel's viewing angle is tilted no more than 30 degrees from parallel to the operator (i.e. directly in front).

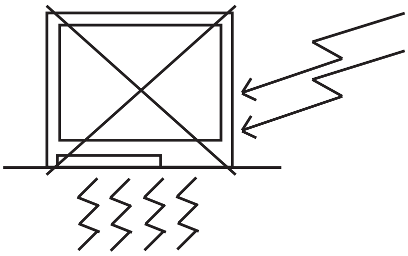
Installation Location

- Avoid placing the PL next to other devices that might cause overheating.
- Keep the PL away from arc-generating devices such as magnetic switches and non-fuse breakers.
- Avoid using the PL in environments where corrosive gases are present.
- To ensure the reliability, operability and ventilation of the PL, be sure to install it in locations that are more than 50mm away from adjacent structures or equipment. Also, consider the need for installing or removing expansion boards, or connectors when designing and installing your PL.



Vibration and Shocks

If the PL is moved when its enclosure doors are open, or while it is installed in a rack equipped with caster wheels, the hard disk can receive excessive vibration or jolting. Be especially careful at this time.



PL Unit	Shock Resistance
When using the HDD	Up to 4.9m/s^2
When using the FDD	Up to 9.8m/s^2
When using no drives	Up to 19.6m/s^2

• Be sure not to move the PL unit while the HDD is starting up. This can lead to a machine breakdown (Even a slight movement of the PL should not be performed).

• When using a fan to cool the PL unit, be sure that the fan does not point directly at any of the PL's disk drive units.

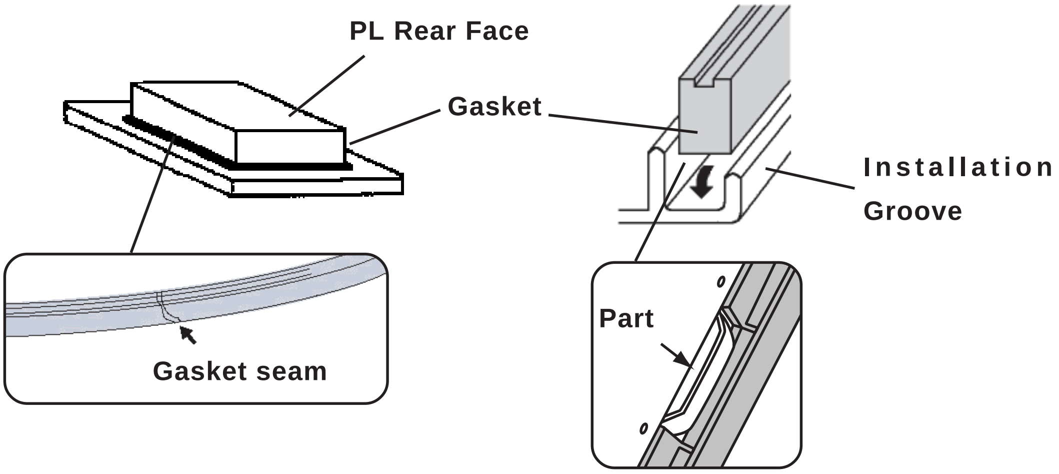
2 Installation Procedures

Follow the steps given below when installing the PL.

Attaching the Installation Gasket

Even if the your PL's Installation Gasket is not needed to prevent water from entering the unit, the gasket also acts as a vibration absorber and should always be attached. To install it, place the PL face down on a soft surface and attach the gasket to the rear side of the display face, in the plastic bezel's groove (see picture below).

- **Before mounting the PL into a cabinet or panel, check that the installation gasket is attached to the unit.**
- **A gasket which has been used for a long period of time may be scratched or dirty, and may have lost much of its water resistance. Be sure to change the gasket at least once a year, or when scratches or dirt become visible.**
- **Since the gasket is flexible but not elastic, be careful not to stretch it uneccessarily, as doing so could tear the gasket.**
- **Be sure, when pushing the gasket into the installation groove and around the corners of the PL, that the gasket's seam is not placed in a corner. Placing the seam here could eventually cause the gasket to tear.**



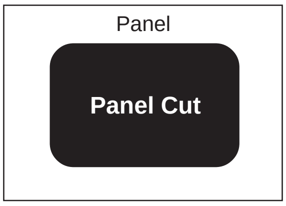
- Be sure to place the gasket's flat, non-grooved side facing down.
- The PL's bezel has a part attached to it. To prevent the installation gasket from contacting this part, be sure to press the installation gasket completely into its groove.

- **If the installation gasket is not properly inserted into the groove, the gasket's moisture resistance may not be equivalent to IP65f.**
- **When the installation gasket is properly inserted, approximately 2.0 mm of it will extend outside the groove. Before installing a PL into a panel, be sure to confirm that the gasket is correctly in place.**

Create a Panel Cut

Create a panel cut for the PL, like that pictured here. Two additional items, the installation gasket and the installation fasteners are required when installing the PL.

refer to 2-5-5 Panel Cut Dimensions

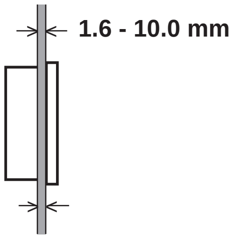


NOTE

- To obtain the maximum degree of moisture resistance, be sure to install the PL on a smooth, flat surface.
- The panel itself can be from 1.6 to 10.0 mm thick.

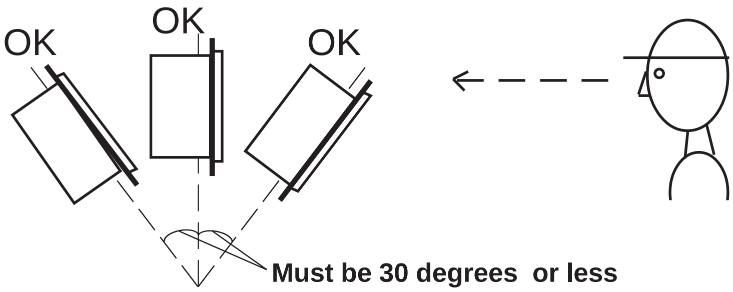
CAUTION!

Strengthening may be required for the panel. Be sure to consider the weight of the PL when designing the panel.



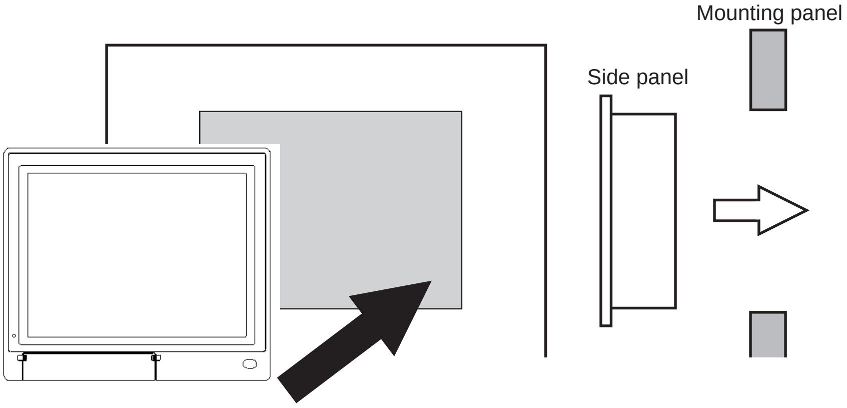
PL Viewing Angle

- Be sure that the panel's viewing angle is tilted no more than 30 degrees from parallel to the operator (i.e. operator is directly in front).



- Avoid placing the PL next to other devices that might cause overheating.
- Avoid using the PL where the ambient temperature will be higher than the allowed specification.
- Keep the PL away from arc-generating devices such as magnetic switches and non-fuse breakers.
- Avoid using the PL in environments where corrosive gases are present.

Insert the PL into the installation slot



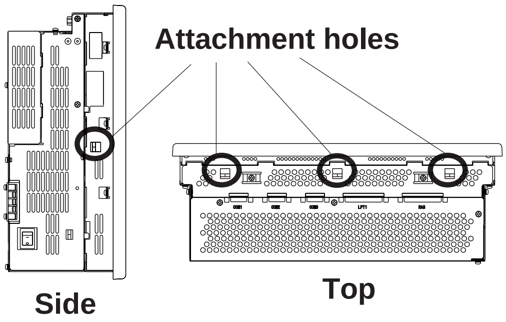
CAUTION! **Be sure the panel cut's actual measurements are the same as those given here, otherwise the PL may slip or fall out of the panel.**

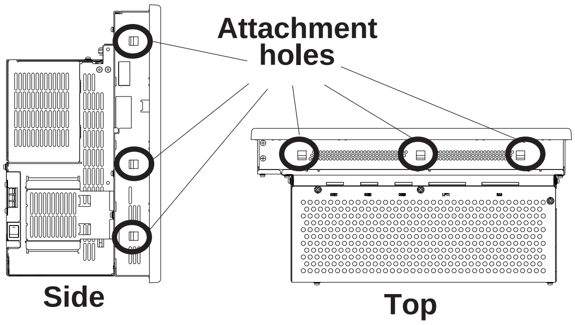
refer to 2-5-6 Panel Cut Dimensions

Attach and Secure the Rear Installation Fasteners

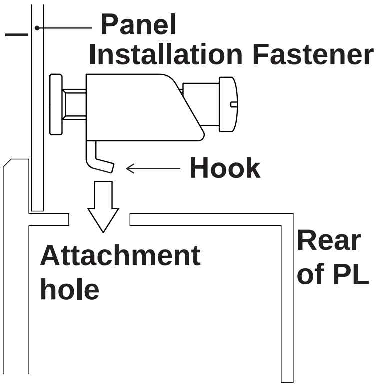
1) Locate the PL's attachment holes, located on the top, bottom, and sides of the PL.

■ **PL-6920 Series (8 attachment holes)** ■**PL-7920 Series (12 attachment holes)**

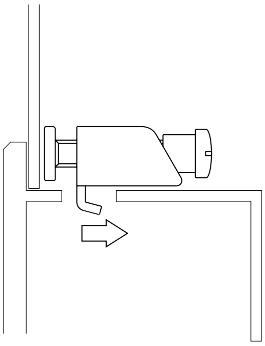




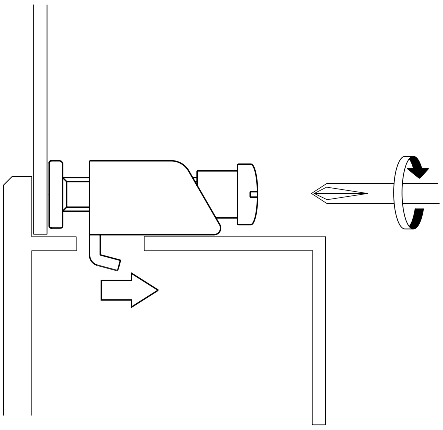
2) Insert each fastener into its attachment hole.



3) Slide the fastener backwards until it stops.



4) Use a screwdriver to tighten the fastener's adjustment screw. To ensure a high degree of moisture resistance, the torque should be 0.5N•m.

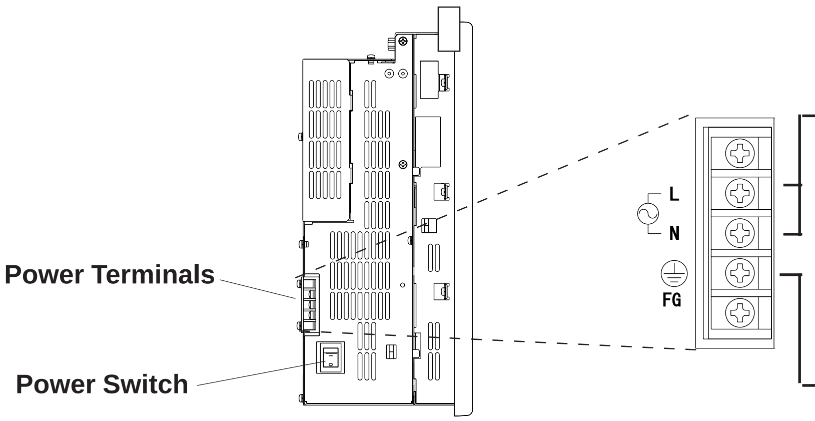


CAUTION! **Excessive torque may damage the panel or bracket.**

4-2 Wiring the PL

1 Connecting the Power Cord

(Rear of PL)



Power Input Terminals

Connecting these two terminals provides 100V to 240V AC power.

L= live and N= Neutral

(screw size: M3)

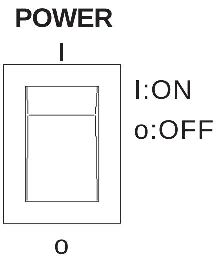
This terminal is connected to the PL's frame (chassis).

PL-6921/PL-7921 (2-Slot type)

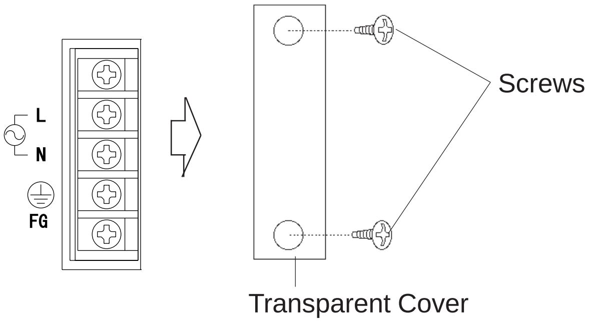
WARNINGS

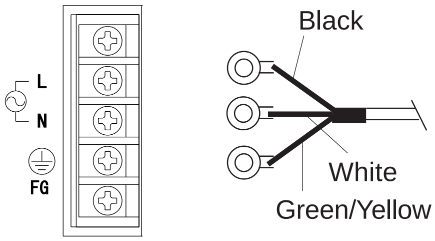
- **To prevent electric shocks, be sure to turn the PL OFF before connecting the power cord.**
- **To avoid the dangers of fire, electric hazards and equipment damage, be sure to use only the specified power supply voltage when operating the PL.**

Use the following steps when connecting the power cord to the PL's power terminals.



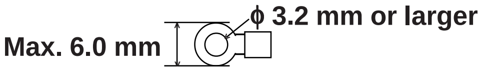
1) Confirm that the power switch is turned OFF. Then, remove the power terminal's transparent plastic cover.



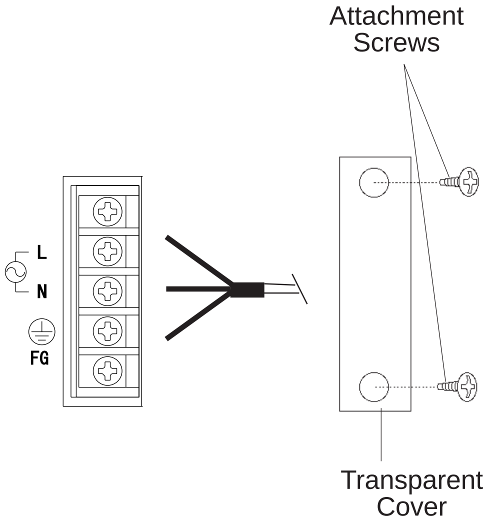


2) **Loosen and remove the middle three(3) screws from the terminal strip. Align the crimp terminals with each screw hole, and tighten the screws.**

- Crimp Terminal Types : V1.25-3, by J.S.T. or equivalent (JIS standard part number : **RAV1.25-3**)
- Crimp terminals must be the same as shown below.



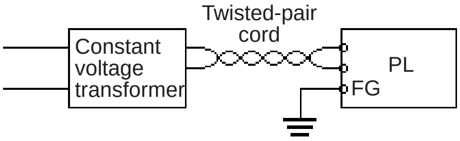
- **The colors in the figure above are for the cord which came with the PL.**
- **This power cord is designed only for AC100/115V use. Be sure to use a different cord when using other than AC100/115V power.**



3) **Reattach the terminal strip's transparent cover with the attachment screws.**

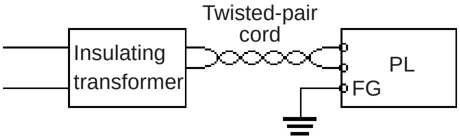
2 Power Supply Cautions

When connecting the PL unit's AC power terminals, please be aware of the following:



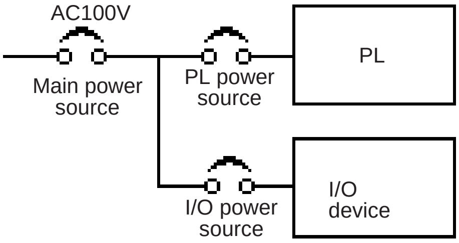
- If voltage fluctuations are expected to vary beyond the specified range, connect a constant voltage transformer.

For information about the specified voltage, refer to "**2-1 General Specifications**"

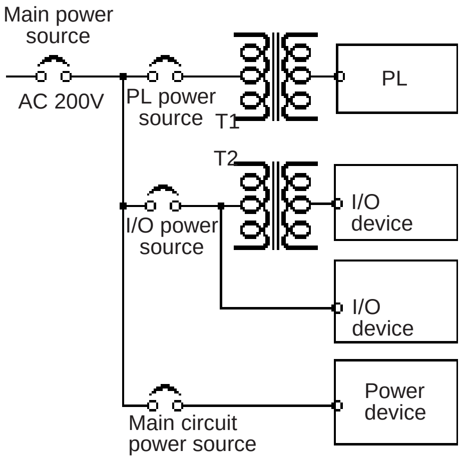


- Use a low-noise power supply both between the lines and between the PL and its ground. If there is still excess noise, connect an insulating transformer (noise-prevention type) .

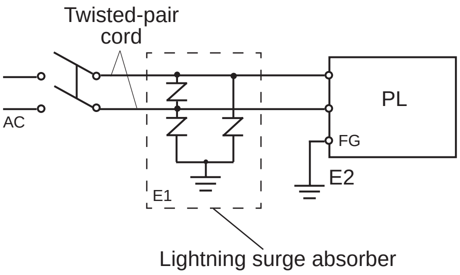
NOTE Be sure any constant or insulating transformer used has a capacity of 200VA or more.



- Wire the power cords of the PL, I/O devices, and power supply devices separately.



- To improve noise immunity, it is recommended to attach a ferrite core to the power cord.
- Isolate the main circuit (high voltage, large current) line, I/O signal lines, and power cord, and do not bind or group them together.



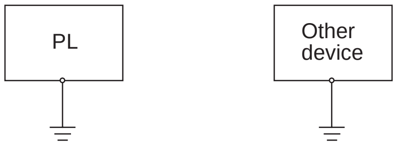
- To prevent damage from lightning, connect a lightning surge absorber.

CAUTION!

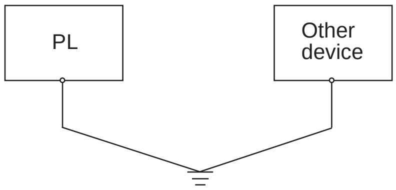
- **Ground the lightning surge absorber (E1) and the PL (E2) separately.**
- **Select a lightning surge absorber which will not exceed the allowable circuit voltage, even when the voltage rises to the maximum.**

3 Grounding Cautions

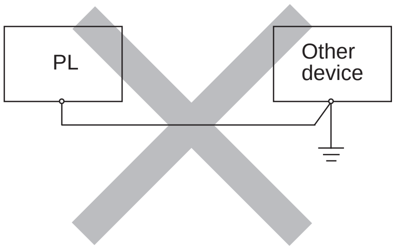
(a) Dedicated Ground *1



(b) Shared Ground - allowed *1



(c) Shared ground - not allowed



- **Set up a dedicated ground when using the rear panel's FG terminal.**
- **If a dedicated ground is not possible, use a shared ground, as shown in figure (b).**
- **Inside the PL unit, the SG (Signal Ground) and FG (Frame Ground) terminals are connected to each other.**
- **When connecting an external device to PL with the SG terminal, ensure that no short-circuit loop is created when you set up the system.**
- **The grounding point must be as close to the PL as possible, and the grounding wires must be as short as possible. If the wires must be long, use thick, insulated wires and run them through conduits.**

4 Cautions When Connecting I/O Signal Lines

- I/O signal lines must be wired separately from the power circuit cable. If the power circuit cable needs to be wired together with the input/output (I/O) signal lines for any reason, use shielded cables and ground one end of the shield to the PL's FG terminal.
- To improve noise immunity, it is recommended to attach a ferrite core to the power cord.

***1** Use a grounding resistance of 100Ω or less, and a 2mm^2 or thicker wire, or your country's applicable standard. For details, contact your local PL distributor.

Chapter 5

System Setup

This chapter explains how to enter a PL-6920/PL-7920 Series unit's system settings, as well as the cautions required both before and during set up.

5-1 Setup Procedures

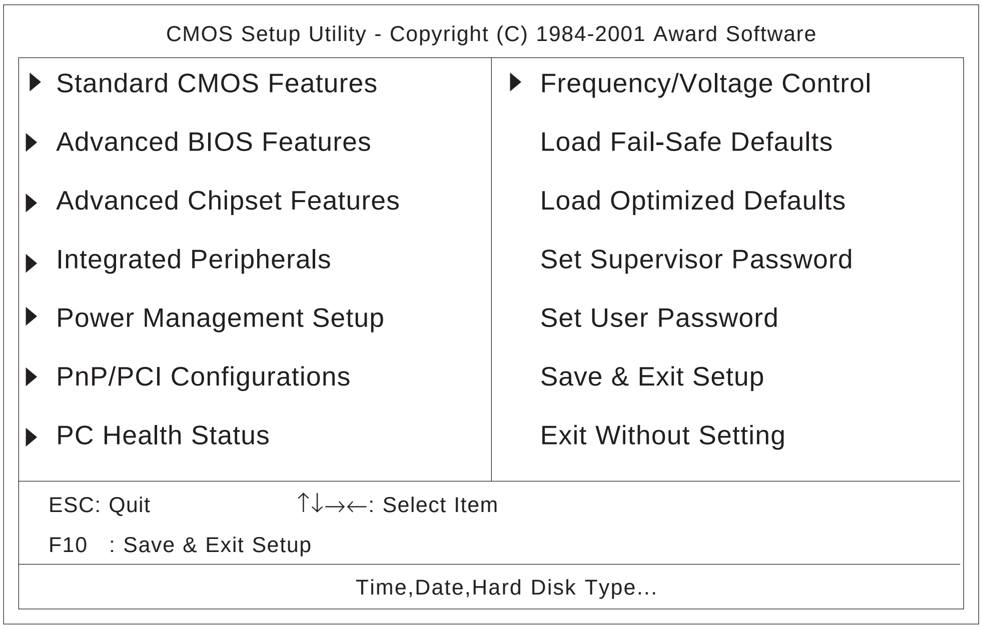
CAUTION! ***Normally, use only the factory (default) settings.***

NOTE **The following settings are those pre-set at the factory.**

1) Connect a keyboard to the PL.

2) Turn the PL's power ON.

3) After the message "Press <DEL> to Enter SETUP" appears, press the [DEL] key until the following screen appears.



```
              CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
 ▶ Standard CMOS Features          ▶ Frequency/Voltage Control
 ▶ Advanced BIOS Features            Load Fail-Safe Defaults
 ▶ Advanced Chipset Features         Load Optimized Defaults
 ▶ Integrated Peripherals            Set Supervisor Password
 ▶ Power Management Setup            Set User Password
 ▶ PnP/PCI Configurations            Save & Exit Setup
 ▶ PC Health Status                  Exit Without Setting
   ESC: Quit              ↑↓→←: Select Item
   F10  : Save & Exit Setup
                         Time,Date,Hard Disk Type...
```

KEYBOARD ACTION KEYS
Provides a summary of the keyboard keys used to carry out the set up.

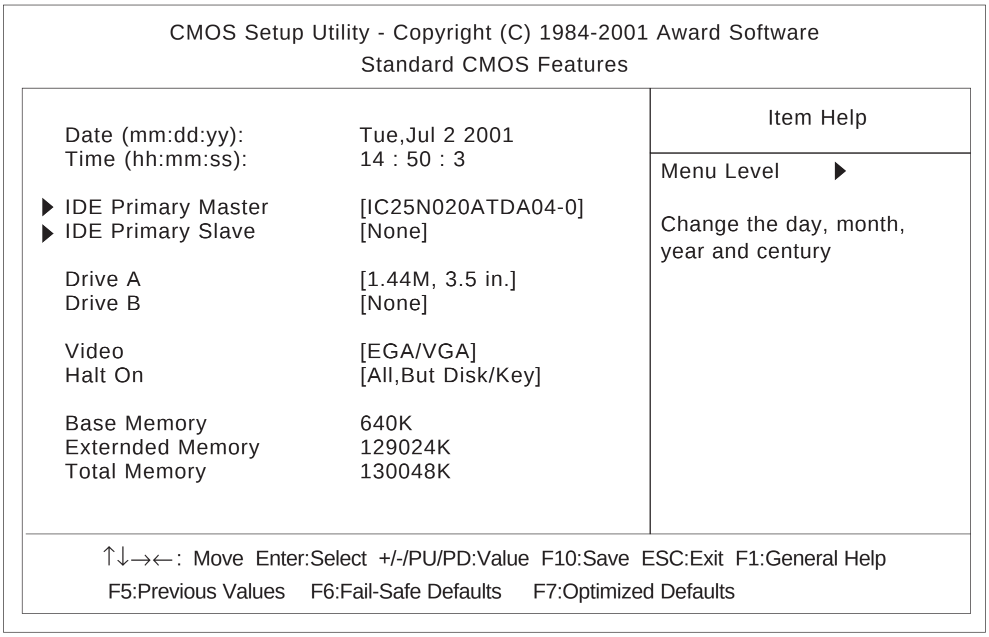
SYSTEM SETTING SELECTION AREA
Each of the titles (areas) listed here contains system setting items.

4) Use the arrow keys to move the cursor to the desired selection.

5-2 System Parameters

1 STANDARD CMOS FEATURES

Selecting the STANDARD CMOS FEATURES menu item produces the following screen.



```
CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                 Standard CMOS Features

                                                      Item Help
  Date (mm:dd:yy):        Tue,Jul 2 2001
  Time (hh:mm:ss):        14 : 50 : 3          Menu Level    ▶

▶ IDE Primary Master      [IC25N020ATDA04-0]   Change the day, month,
▶ IDE Primary Slave       [None]               year and century

  Drive A                 [1.44M, 3.5 in.]
  Drive B                 [None]

  Video                   [EGA/VGA]
  Halt On                 [All,But Disk/Key]

  Base Memory             640K
  Externded Memory        129024K
  Total Memory            130048K

↑↓→←: Move  Enter:Select  +/-/PU/PD:Value  F10:Save  ESC:Exit  F1:General Help
   F5:Previous Values   F6:Fail-Safe Defaults   F7:Optimized Defaults
```

■ Date/Time

This data sets the PL's internal time and date.

Hours	:00 - 23
Minutes	:00 - 59
Seconds	:00 - 59

■ IDE Primary Master (Slave)

Displays the size of the Hard Disk installed in the PL. Pressing the [Enter] key will call up the Parameter settings menu.

For details, refer to ***5-2-2 IDE HDD Auto Detection***

■ Drive A (B)

This setting determines the format used by the PL's internal floppy disk drive. The available settings are [None], [720K - 3.5in], [1.44M - 3.5in], or [2.88M, 3.5in]. The factory settings are Drive A [1.44M - 3.5in] and Drive B [None] and recommended for most users.

■ Video

The selections for the screen (video) mode. The available settings are [EGA/VGA], [CGA40], [CGA80] and [MONO]. The [EGA/VGA] selection is factory set and recommended for most users.

■ Halt On

Designates the type of processing that will be performed when an error occurs during the Initial Start-Up's Self Test. The [All But Disk /Key] selection is factory set and recommended for most users.

[All Errors] : Displays all errors and stops the unit.

[No Errors] : Displays all errors and does not stop the unit.

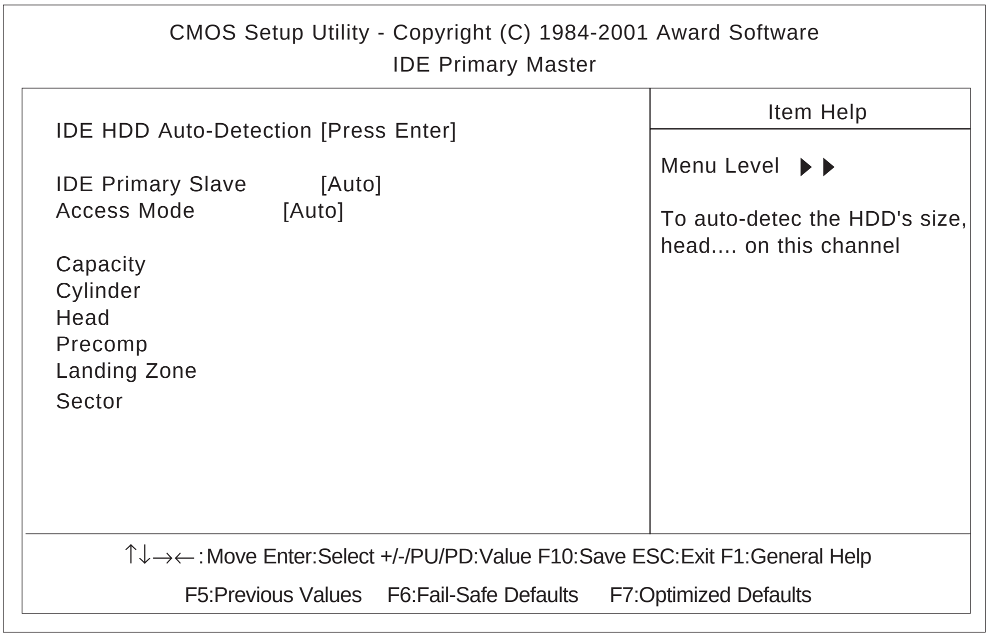
[All,But Keyboard] : Displays all errors, except for those related to the keyboard, and stops the unit. If the User has no keyboard connected, please use this setting.

[All,But Diskette] : Displays all errors, except for those related to the disk drive (FDD), and stops the unit.

[All,But Disk/Key] : Displays all errors, except for those related to the disk drive (FDD) and keyboard, and then stops the unit.

2 IDE HDD AUTO DETECTION

The Standard CMOS Features menu is used to select the IDE Primary Master and the IDE Primary Slave.



```
              CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                                  IDE Primary Master

  IDE HDD Auto-Detection [Press Enter]            |          Item Help
                                                  | Menu Level  ▶▶
  IDE Primary Slave          [Auto]               |
  Access Mode             [Auto]                  | To auto-detec the HDD's size,
                                                  | head.... on this channel
  Capacity                                        |
  Cylinder                                        |
  Head                                            |
  Precomp                                         |
  Landing Zone                                    |
  Sector                                          |

     ↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
          F5:Previous Values    F6:Fail-Safe Defaults    F7:Optimized Defaults
```

■ **IDE HDD Auto-Detection**

This setting detects the hard disk connected to the IDE interface.

■ **IDE Primary Master (Slave)**

This setting designates the IDE type Hard Disk's parameter setting method. The available settings are [None], [Auto], or [Manual]. The factory default setting is [Auto] and is recommended for most users.

■ **Access Mode**

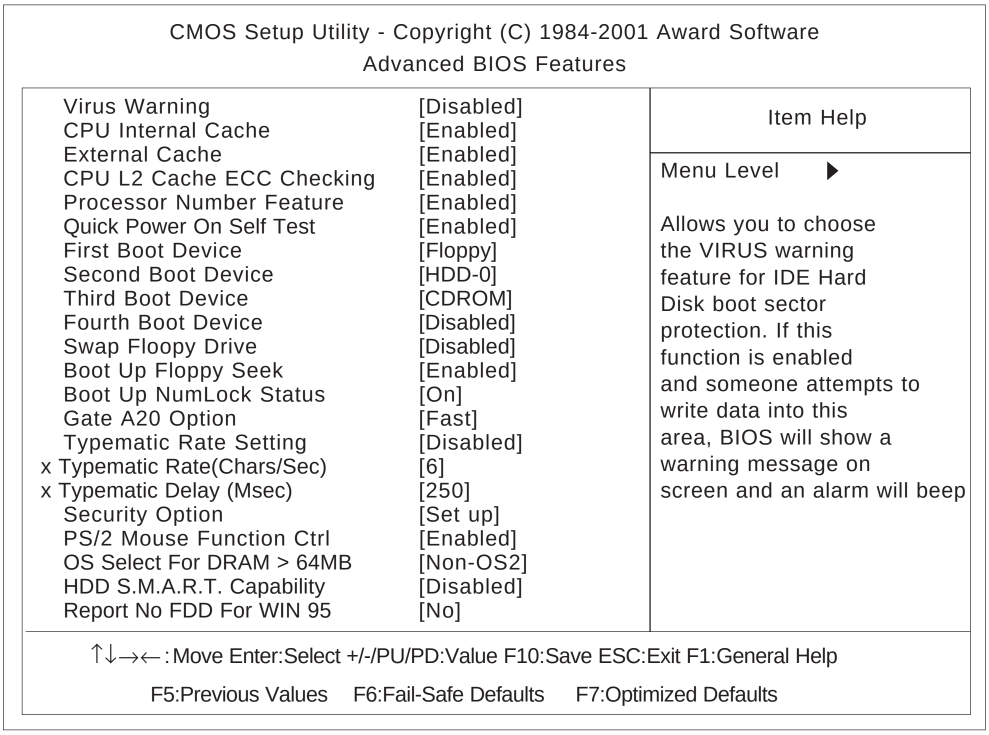
This setting designates the IDE type Hard Disk's access mode. The available settings are [CHS], [LBA], [Large], or [Auto]. The factory default setting is [Auto] and is recommended for most users.

■ **Capacity/Cylinder/Head/Precomp/Landing Zone/ Sector**

These settings designate individual IDE type Hard Disk parameter settings. These can only be set when the [IDE Primary Master (Slave)] setting is set to [Manual]. When the [IDE Primary Master (Slave)] setting is set to [Auto], these values are automaticaly detected. Capacity is set automaticaly.

3 ADVANCED BIOS FEATURES

Selecting the ADVANCED BIOS FEATURES menu item calls up the following screen.



```
        CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                          Advanced BIOS Features

   Virus Warning                   [Disabled]  |        Item Help
   CPU Internal Cache              [Enabled]   |
   External Cache                  [Enabled]   |----------------------------
   CPU L2 Cache ECC Checking       [Enabled]   | Menu Level    ▶
   Processor Number Feature        [Enabled]   |
   Quick Power On Self Test        [Enabled]   | Allows you to choose
   First Boot Device               [Floppy]    | the VIRUS warning
   Second Boot Device              [HDD-0]     | feature for IDE Hard
   Third Boot Device               [CDROM]     | Disk boot sector
   Fourth Boot Device              [Disabled]  | protection. If this
   Swap Floopy Drive               [Disabled]  | function is enabled
   Boot Up Floppy Seek             [Enabled]   | and someone attempts to
   Boot Up NumLock Status          [On]        | write data into this
   Gate A20 Option                 [Fast]      | area, BIOS will show a
   Typematic Rate Setting          [Disabled]  | warning message on
 x Typematic Rate(Chars/Sec)       [6]         | screen and an alarm will beep
 x Typematic Delay (Msec)          [250]       |
   Security Option                 [Set up]    |
   PS/2 Mouse Function Ctrl        [Enabled]   |
   OS Select For DRAM > 64MB       [Non-OS2]   |
   HDD S.M.A.R.T. Capability       [Disabled]  |
   Report No FDD For WIN 95        [No]        |

   ↑↓→←: Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
        F5:Previous Values    F6:Fail-Safe Defaults    F7:Optimized Defaults
```

■ **Virus Warning**

This setting determines whether to display a warning when a write to the boot sector is attempted. The available settings are [Enabled] or [Disabled]. The factory default setting is [Disabled] and is recommended for most users.

■ **CPU Internal Cache**

This setting determines the usage of the CPU's internal cache memory. The available settings are [Disabled] or [Enabled]. The factory default setting is [Enabled] and is recommended for most users.

■ **External Cache**

This setting determines the usage of the external cache memory (L2). The available settings are [Disabled] or [Enabled]. The factory default setting is [Enabled] and is recommended for most users.

■ **CPU L2 Cache ECC Checking**

This setting turns ON or OFF external(L2) Cache Memory's ECC (Error Check Correction). The available settings are [Disabled] and [Enabled]. The factory default setting is [Enabled] and is recommended for most users.

■ **Processor Number Feature**

This option is for the Pentium® III processor. If this setting is set to Enabled, it will check the CPU's serial number. If you do not need to know the serial number, set this option to [Disabled]. If this is turned ON, use the "Serial Number Control Program" found on Intel's web site to perform the check.

■ Quick Power On Self Test

This setting determines whether the quick self test is performed when the power is turned on. The available settings are [Disabled] or [Enabled]. The factory default setting is [Enabled] and is recommended for most users.

■ First/ Second/ Third/ Fourth Boot Device

The selections for the search drive sequence of the operating system. The available settings are [Floppy], [HDD-0]*1, [LS120], [HD-0], [SCSI], [CDROM], [HDD-1]*1, [ZIP100], [LAN], [ISA-FDD], and [Disabled].

■ Swap Floppy Drive

This setting swaps Drives A and B. The available settings are [Disabled] or [Enabled]. The factory default setting is [Disabled] and is recommended for most users.

■ Boot Up Floppy Seek

The setting checks whether the floppy disk drive is installed during the system boot-up process. The available settings are [Disabled] or [Enabled]. The factory default setting is [Enabled] and is recommended for most users.

■ Boot Up Numlock Status

This setting specifies the Numlock key status upon the startup. The available settings are [On] and [Off]. The factory default setting is [On] and is recommended for most users.

■ Gate A20 Option

The available settings are [Normal] and [Fast]. When [Normal] is selected, the Keyboard control will be employed to control Gate A20. When [Fast] is selected, the Chipset will be employed. The factory default setting is [Fast] and is recommended for most users.

■ Typematic Rate Setting

The setting specifies the keyboard speed used when repeating characters.
The available settings are [Enabled] and [Disabled]. The factory default setting is [Disabled] and is recommended for most users.

---

*1 *This has a different meaning from the expansion slot position codes "HDD0" and "HDD1" stamped into the side of the PL unit's metal chassis.*

HDD-0:

-*When only one HDD unit is installed.*

The HDD setting is recognized as "HDD-0" and is not related to the Master/Slave items.

-*When 2 HDD units are installed.*

An HDD unit designated as the Master HDD is recognized as "HDD-0".

HDD-1:

-*When only one HDD unit is installed.*

OS cannot be started. Be sure the BIOS setting is "HDD-0".

-*When 2 HDD units are installed.*

An HDD unit designated as the Slave HDD is recognized as "HDD-1".

- **Typematic Rate (Chars/ Sec)**

  This setting specifies the actual typematic rate (repeated character input per second) when the [Typematic Rate Setting] option is set to [Enabled]. The settings are used to determine how many repeats are performed in one second. The factory default setting is [6] and is recommended for most users.

- **Typematic Delay (Msec)**

  When [Typematic Rate Setting] is set to [Enabled], this setting determines the delay period until the initial repetition is started. The [250] selection is factory set and is recommended for most users.

- **Security Option**

  This setting designates the area to request a password. If during BIOS setup you are requested to enter a password, select [Setup]. If during System Startup you are requested to enter a password, select [System]. This setting is NOT available if the password is not set in the [SET SUPERVISOR PASSWORD] or [SET USER PASSWORD] areas. The factory default setting is [Setup] and is recommended for most users.

  [SET SUPERVISOR PASSWORD] Refer to ***5-2-14 Set Supervisor Password***
  [SET USER PASSWORD] Refer to ***5-2-15 Set User Password***

- **PS/2 Mouse Function Ctrl**

  The available selections are [Enabled] and [Disabled], [Enabled] is factory set and recommended for most users.

- **OS Select For DRAM > 64MB**

  The available settings are [Non-OS2] and [OS2]. The factory default settting is [Non-OS2] and is recommended for most users.

- **HDD S.M.A.R.T Capability**

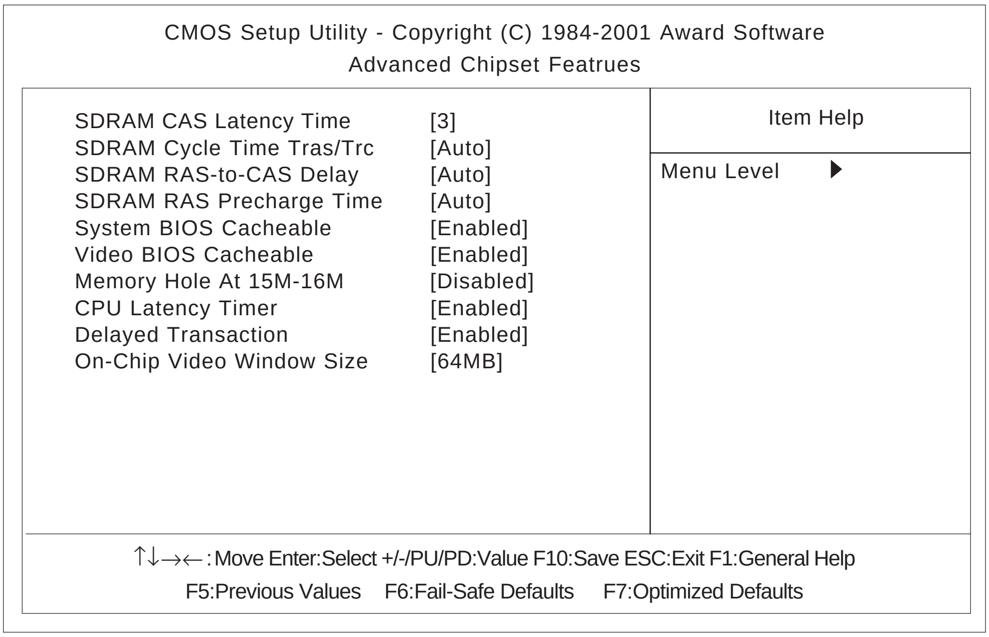
  This feature sets the HDD unit's SMART (Self-Monitoring Analysis and Reporting Technology). The available selections are [Enabled] and [Disabled], [Disabled] is factory set and recommended for most users.

- **Report No FDD For WIN 95**

  This setting determines if the FDD node is reported by BIOS to WIN95. The available settings are [No] and [Yes]. The factory default setting is [No] and is recommended for most users.

4 ADVANCED CHIPSET FEATURES

Selecting the ADVANCED CHIPSET FEATURES menu item calls up the following screen.



```
CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                Advanced Chipset Featrues

    SDRAM CAS Latency Time          [3]              Item Help
    SDRAM Cycle Time Tras/Trc       [Auto]
    SDRAM RAS-to-CAS Delay          [Auto]       Menu Level   ▶
    SDRAM RAS Precharge Time        [Auto]
    System BIOS Cacheable           [Enabled]
    Video BIOS Cacheable            [Enabled]
    Memory Hole At 15M-16M          [Disabled]
    CPU Latency Timer               [Enabled]
    Delayed Transaction             [Enabled]
    On-Chip Video Window Size       [64MB]

  ↑↓→← : Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
      F5:Previous Values    F6:Fail-Safe Defaults    F7:Optimized Defaults
```

- **SDRAM CAS Latency Time**
  Designates the clock counts used, from the enabling of CAS to the start of the burst transmission. Can be set to either [3] or [2]. Factory default setting is [3] and strongly recommended for most users.

- **SDRAM Cycle Time Trans/Trc**
  Designates the number of SLCK's for an access cycle, i.e. the minumum required time from when a bank is activated to the activation of an identical bank. Settings are [7/9], [5/7] or [Auto]. Factory default setting is [Auto] and strongly recommended for most users.

- **SDRAM RAS-to-CAS Delay**
  Designates the timing delay used between RAS and CAS strobe signals. Settings are [2], [3], or [Auto]. Factory default setting is [Auto] and strongly recommended for most users.

- **SDRAM RAS Precharge Time**
  Designates the pre-charge time value used, to allow RAS to accumulte its charge before DRAM refresh. Settings are [2], [3], or [Auto]. Factory default setting is [Auto] and strongly recommended for most users.

- **System BIOS Cacheable**
  Sets whether the System BIOS' Cache is used or not. When the OS is set to use this cache, the PL unit's processing speed will increase. Settings available are [Enabled] and [Disabled]. The factory setting is [Enabled] and is recommended for most users.

- **Video BIOS Cacheable**

Sets whether the Video BIOS' Cache is used or not. Settings available are [Enabled] and [Disabled]. The factory setting is [Enabled]. When this feature is [Enabled], the OS' BIOS ROM range available for caching is from C0000h - F7FFFh, which will improve the video performance. However, if another program tries to write to this area of memory, a system error may occur.

- **Memory Hole At 15M-16M**

This setting determines whether to designate the memory space from 15MB to 16MB as the buffer area for the ISA bus card. The available settings are [Disabled] and [Enabled]. The factory default setting is [Disabled] and is recommended for most users.

- **CPU Latency Timer**

If Enabled, a deferrable CPU cycle will only be Deferred after it has been in a Snoop Stall for 31 scans and another ADS# has arrived. If Disabled, changeable CPU cycles will be changed when an ADS number is received.

- **Delayed Transaction**

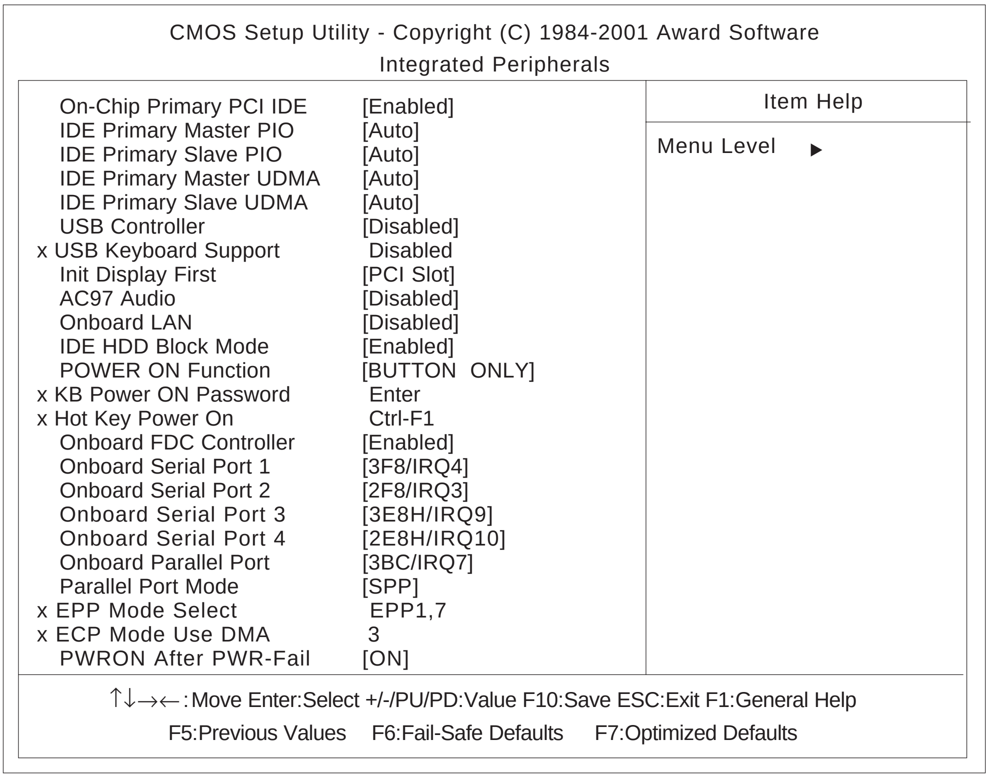
Designates the length of the pre-charge time. The available settings are [Enabled] and [Disabled]. The factory default setting is [Enabled] and is recommended.

- **On-Chip Video Window Size**

Designates the on-chip video window size used by the VGA driver. The available settings are [Disabled] and [64MB]. The factory default setting is [64MB] and is recommended for most users.

5 INTEGRATED PERIPHERALS

Selecting INTEGRATED PERIPHERALS SETUP menu item displays the following screen.



```
CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                  Integrated Peripherals

   On-Chip Primary PCI IDE      [Enabled]          Item Help
   IDE Primary Master PIO       [Auto]
   IDE Primary Slave PIO        [Auto]           Menu Level  ▶
   IDE Primary Master UDMA      [Auto]
   IDE Primary Slave UDMA       [Auto]
   USB Controller               [Disabled]
 x USB Keyboard Support          Disabled
   Init Display First           [PCI Slot]
   AC97 Audio                   [Disabled]
   Onboard LAN                  [Disabled]
   IDE HDD Block Mode           [Enabled]
   POWER ON Function            [BUTTON  ONLY]
 x KB Power ON Password          Enter
 x Hot Key Power On              Ctrl-F1
   Onboard FDC Controller       [Enabled]
   Onboard Serial Port 1        [3F8/IRQ4]
   Onboard Serial Port 2        [2F8/IRQ3]
   Onboard Serial Port 3        [3E8H/IRQ9]
   Onboard Serial Port 4        [2E8H/IRQ10]
   Onboard Parallel Port        [3BC/IRQ7]
   Parallel Port Mode           [SPP]
 x EPP Mode Select               EPP1,7
 x ECP Mode Use DMA              3
   PWRON After PWR-Fail         [ON]

   ↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
        F5:Previous Values   F6:Fail-Safe Defaults   F7:Optimized Defaults
```

- **On-Chip Primary PCI IDE**
  Designates the internal IDE port's setting. The two selections available are [Disabled] and [Enabled]. The [Enabled] selection is factory set and recommended for most users.

- **IDE Primary Master (Slave) PIO**
  Designates the Master Drive's Operation Mode. Only after the "Internal PCI/IDE" has been set to [Primary] can these IDE Primary Master PIO settings be entered. The two selections available are [Auto], [Mode0], [Mode1], [Mode2], [Mode3], and [Mode4]. The [Auto] selection is factory set and recommended for most users. When using the CD-ROM drive, select [Mode 2].

- **IDE Primary Master (Slave) UDMA**
  Designates the Master Drive's UDMA Operation Mode. The two selections available are [Auto] or [Disabled]. The [Auto] selection is factory set and recommended for most users.When using the CD-ROM drive, select [Mode 2].

USB Controller

Select [Enabled] or [Disable] when attaching a USB device. The [Disabled] selection is factory set and recommended for most users. When using the USB I/F or the USB connector for connecting the touch panel, set this item to “Enabled”.

CAUTION! ***If a USB device is connected, change the [USB Controller] and [PnP/PCI Configration] menu's [Assign IRQ For USB] settings to [Enabled].***

USB Keyboard Support

Select [Enabled] or [Disable] if your system contains a USB controller and you wish to use a USB keyboard. The available settings are [Disabled] and [Enabled]. The factory default setting is [Disabled]. Either [Disabled] or [Enabled] can be selected. If, however, the USB Controller is set to [Disabled], this setting cannot be changed.

CAUTION! ***If a USB device is connected, change the [USB Controller] and [PnP/PCI Configration] menu's [Assign IRQ For USB] settings to [Enabled].***

Init Display First

When both a PCI and an AGP device are installed, this setting designates which will output first. Settings are [PCI slot] or [Onboard/AGP]. The factory default setting is [PCI slot].

AC97 Audio

Enables or disables the 815 chipset family to support AC97 Audio. The available settings are [Disabled] and [Auto]. The factory default setting is [Disabled].

Onboard LAN

The available settings are [Disabled] and [Enabled]. The factory default setting is [Disabled].

IDE HDD Block Mode

This setting determines whether to enable the Block Mode on the HDD supporting the Block Mode. The available settings are [Disabled] and [Enabled]. The factory default setting is [Enabled] and is recommended for most users.

Power ON Function

This feature is not supported.

KB Power ON Password

This feature is not supported.

Hot Key Power ON

This feature is not supported.

■ Onboard FDC Controller

Designates whether the PL's FDD(Floppy Disk Drive) controller is used or not. Settings available are [Disabled] or [Enabled]. The [Enabled] selection is factory set.

■ Onboard Serial Port 1

Designates the PL's Serial Port 1 I/O address. The selections include [Disabled], [Auto], [3F8/IRQ4], [2F8/IRQ3], [3E8/IRQ4] and [2E8/IRQ3]. The [3F8/IRQ4] selection is factory set and recommended for most users.

■ Onboard Serial Port 2

Designates the PL's Serial Port2 I/O address. The selections include [Disabled], [Auto], [3F8/IRQ4], [2F8/IRQ3], [3E8/IRQ4] and [2E8/IRQ3]. The [2F8/IRQ3] selection is factory set and recommended for most users.

■ Onboard Serial Port 3

Designates the PL's Serial Port3 address setting. The selections include [Disabled], [3F8H/IRQ9], [2F8H/IRQ10], [3E8H/IRQ9] or [2E8H/IRQ10]. The [3E8H/IRQ9] selection is factory set and recommended for most users.

■ Onboard Serial Port 4

Designates the PL's Serial Port3 address setting. The selections include [Disabled], [3F8H/IRQ9], [2F8H/IRQ10], [3E8H/IRQ9] or [2E8H/IRQ10]. The [2E8H/IRQ10] selection is factory set and recommended for most users.When using the USB connector for connecting the touch panel, set this item to “Disabled”.

■ Onboard Parallel Port

Selects the PL Parallel Port 1's I/O Address. The selections include [Disabled], [3BC/IRQ7], [378/IRQ7] or [278/IRQ5]. The [3BC/IRQ7] selection is factory set and recommended for most users.

■ Parallel Port Mode

Here, the parallel port's operation mode can be selected ([SPP], [EPP1.7], [EPP1.9], [ECP], or [ECP + EPP]). Normally, [SPP] is used and recommended for most users. When either [ECP] or [ECP+EPP] is selected, the [ECP Mode Use DMA] setting can be used.

■ EPP Mode Select

When [EPP] or [ECP + EPP] mode is selected, this feature allows you to select the EPP mode protocol. [EPP 1.7] or [EPP 1.9] can be selected.

■ ECP Mode Use DMA

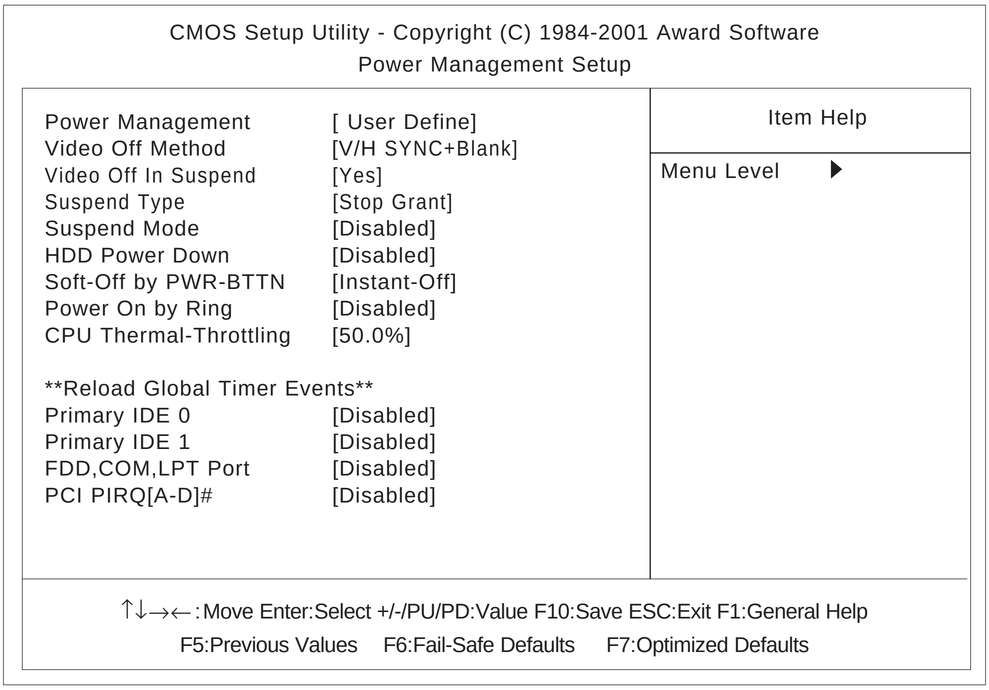
Designates the ECP mode's DMA channel. [1] and [3] are available selections, however, this item can be set only when the Parallel Port Mode is set to either [ECP] or [ECP+EPP].

■ PWRON After PWR-Fail

This feature is not supported.

6 POWER MANAGEMENT SETUP

Selecting the POWER MANAGEMENT SETUP menu item calls up the following screen.



```
        CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                          Power Management Setup

Power Management          [ User Define]            Item Help
Video Off Method          [V/H SYNC+Blank]
Video Off In Suspend      [Yes]                 Menu Level   ▶
Suspend Type              [Stop Grant]
Suspend Mode              [Disabled]
HDD Power Down            [Disabled]
Soft-Off by PWR-BTTN      [Instant-Off]
Power On by Ring          [Disabled]
CPU Thermal-Throttling    [50.0%]

**Reload Global Timer Events**
Primary IDE 0             [Disabled]
Primary IDE 1             [Disabled]
FDD,COM,LPT Port          [Disabled]
PCI PIRQ[A-D]#            [Disabled]

 ↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
     F5:Previous Values    F6:Fail-Safe Defaults    F7:Optimized Defaults
```

■ **Power Management**

You can choose from three power management options. These are [User Define], [Min Saving] or [Max Saving]. The [User Define] selection is factory set and recommended for most users.

■ **Video Off Method**

This setting determines the method to blank the display screen. The available settings are [Blank Screen], [V/H SYNC+Blank], and [DPMS Supported]. The [Blank Screen] selection blanks the display. The [V/H SYNC+Blank] blanks the display and also suspends the Vertical/Horizontal synchronization signal of the display. The [DPMS Supported] selection controls the operation when a CRT that supports DPMS is used. The factory default setting is [V/H SYNC+Blank] and is recommended for most users.

■ **Video Off In Suspend**

Designates how the monitor is blanked. The available settings are [Yes] and [No]. The factory default setting is [Yes].

■ **Suspend Type**

Designates the type of suspend method used. The available settings are [Stop Grant] and [PWRON Suspend]. The factory default setting is [Stop Grant].

■ **Suspend Mode**

When enabled, designates the period of time before all devices except the CPU are shut down. The available settings are [1Min], [2Min], [4Min], [8Min], [12Min], [20Min], [30Min], [40Min], [1Hour] and [Disabled]. The factory default setting is [Disabled].

- **HDD Power Down**

  Designates the length of time until the motor of the hard disk stops. The available selections are[1Min]→[15Min] and [Disabled]. [Disabled] is factory set and recommended for most users.

- **Soft-Off by PWR-BTTN**

  Set up the power buttons used for software control. Available selections are [Delay 4 sec] and [Instant-off], however, since the PL unit has no power button for software control, be sure to select [Instant-off]. Factory setting is [Instant-off].

- **Power On by Ring**

  When enabled, the system will boot up when the modem receives a call. When disabled, the sytstem will ignore all calls received by the modem. The available settings are [Enabled] and [Disabled]. The factory default setting is [Disabled].

- **CPU Thermal-Throttling**

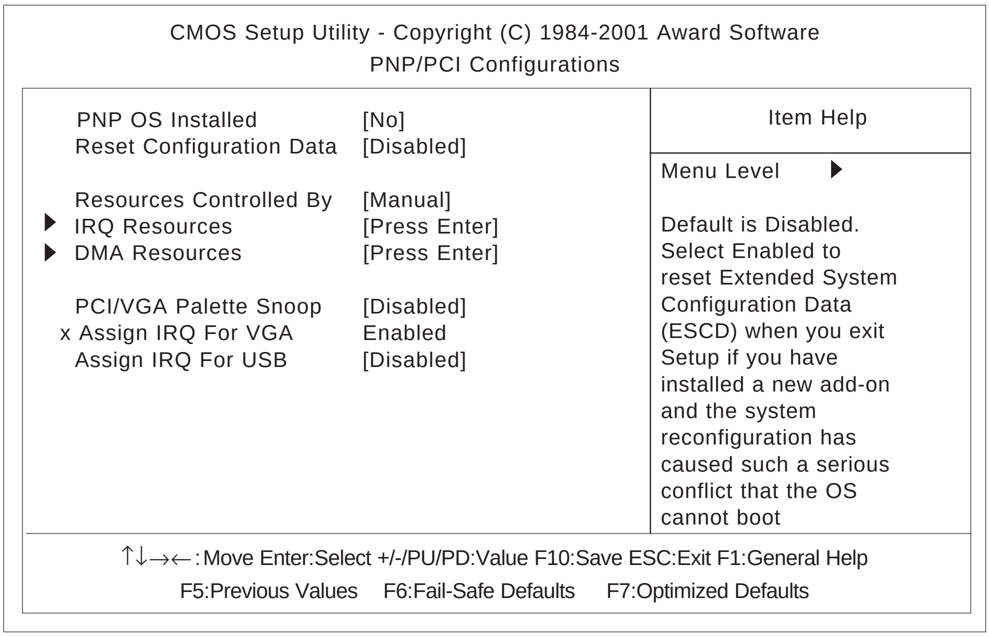
  Designates the CPU THRM-Throttling rate. The available settings are [12.5%], [25.0%], [37.5%], [50.0%], [62.5%], [75%] and [87.5%]. The factory default setting is [50.0%].

- ****Reload Global Timer Events****

  This feature sets the event that reloads the timer used to designate the amount (period) of PL unit idle time prior to changing to energy-saving mode. Thus, when an event occurs for any of the items in this area that are set to [Enabled], the system will reload the Timer.

7 PNP/PCI CONFIGURATION

Selecting the PnP/PCI CONFIGURATION menu item displays the following screen.



```
             CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                              PNP/PCI Configurations

     PNP OS Installed            [No]             |          Item Help
     Reset Configuration Data    [Disabled]       |--------------------------
                                                  | Menu Level      ▶
     Resources Controlled By     [Manual]         |
  ▶  IRQ Resources               [Press Enter]    | Default is Disabled.
  ▶  DMA Resources               [Press Enter]    | Select Enabled to
                                                  | reset Extended System
     PCI/VGA Palette Snoop       [Disabled]       | Configuration Data
   x Assign IRQ For VGA          Enabled          | (ESCD) when you exit
     Assign IRQ For USB          [Disabled]       | Setup if you have
                                                  | installed a new add-on
                                                  | and the system
                                                  | reconfiguration has
                                                  | caused such a serious
                                                  | conflict that the OS
                                                  | cannot boot

   ↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
          F5:Previous Values    F6:Fail-Safe Defaults    F7:Optimized Defaults
```

■ PNP OS Installed

Setting used when the user's OS complies with Plug-And-Play standards. The selection options are either [Yes] or [No]. The [No] setting is factory set and recommended for most users.

■ Reset Configuration Data

Designates whether ESCD (Extended System Configuration Data) data should be erased or not. The two selections available are [Disabled] and [Enabled]. The [Disabled] selection is factory set and recommended for most users.

■ Resources Controlled By

The Plug and Play feature allows you to designate whether the allocation of I/O Port, IRQ and DMA resources is performed automatically or manually. The two selections available are [Manual] or [Auto(ESCD)]. If [Auto(ESCD)] is selected, the IRQ Resources and DMA Resources selection will be disabled. The [Manual] selection is factory set and recommended for most users.

■ IRQ Resources

When resources are controlled manually, assign each system interrupt used a type, that reflects the type of device using the interrupt.

☞ refer to ***5-2-8 IRQ Resources***

■ DMA Resources

This menu provides control over the DMA resources used.

☞ refer to ***5-2-9 DMA Resources***

■ PCI/VGA Pallet Snoop

However, depending on the VGA or MPEG board used, select [Enabled]. For setup details, refer to that board's installation guide.

- **Assign IRQ for VGA**

  Designates whether the VGA interrupt is allocated or not. Set to [Enabled].

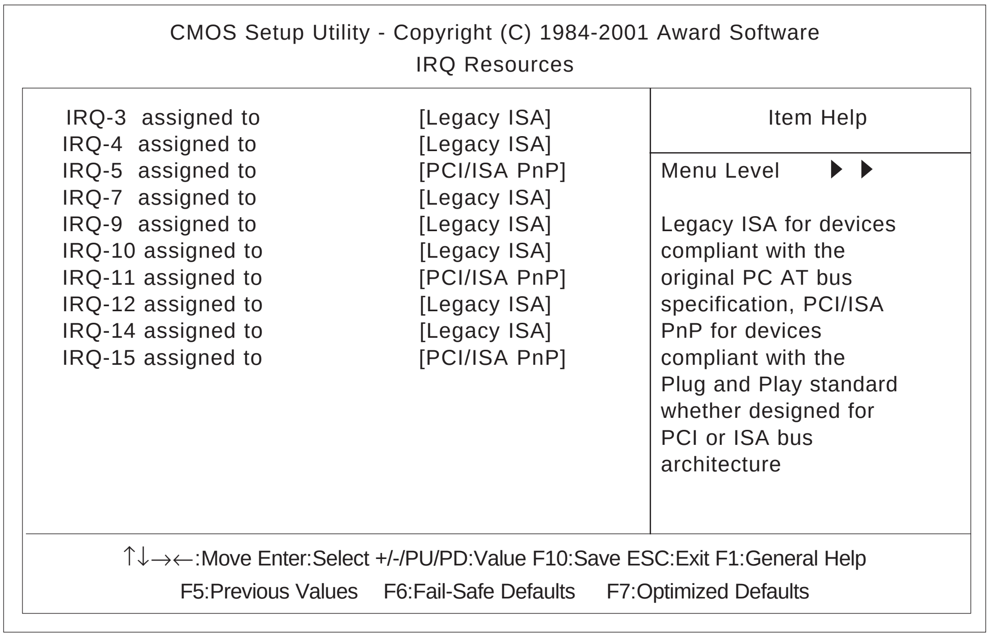
- **Assign IRQ for USB**

  The two selections available are [Disabled] and [Enabled]. The [Disabled] selection is factory set and recommended for most users.When using the USB I/F or the USB connector for connecting the touch panel, set this item to “Enabled”.

CAUTION! ***If a USB device is connected, change the [INTEGRATED PERIPHERALS] menu's [USB Controller] and [Assign IRQ For USB] settings to [Enabled].***

8 IRQ Resources

Select IRQ Resources from the PnP/ PCI Configurations menu and the following screen will appear.



```
          CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                               IRQ Resources

  IRQ-3  assigned to            [Legacy ISA]          Item Help
  IRQ-4  assigned to            [Legacy ISA]
  IRQ-5  assigned to            [PCI/ISA PnP]      Menu Level   ►►
  IRQ-7  assigned to            [Legacy ISA]
  IRQ-9  assigned to            [Legacy ISA]       Legacy ISA for devices
  IRQ-10 assigned to            [Legacy ISA]       compliant with the
  IRQ-11 assigned to            [PCI/ISA PnP]      original PC AT bus
  IRQ-12 assigned to            [Legacy ISA]       specification, PCI/ISA
  IRQ-14 assigned to            [Legacy ISA]       PnP for devices
  IRQ-15 assigned to            [PCI/ISA PnP]      compliant with the
                                                   Plug and Play standard
                                                   whether designed for
                                                   PCI or ISA bus
                                                   architecture

     ↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
          F5:Previous Values    F6:Fail-Safe Defaults    F7:Optimized Defaults
```

- **IRQ-3 assigned to ~ IRQ-15 assigned to**

This setting determines the type of device assigned to the IRQ. This function is available when the [Resource Control By] option under the [PnP/ PCI Configurations] menu is set to [Manual].

[PCI/ISA PnP] .. Select to use a PnP-ready PCI or ISA card.

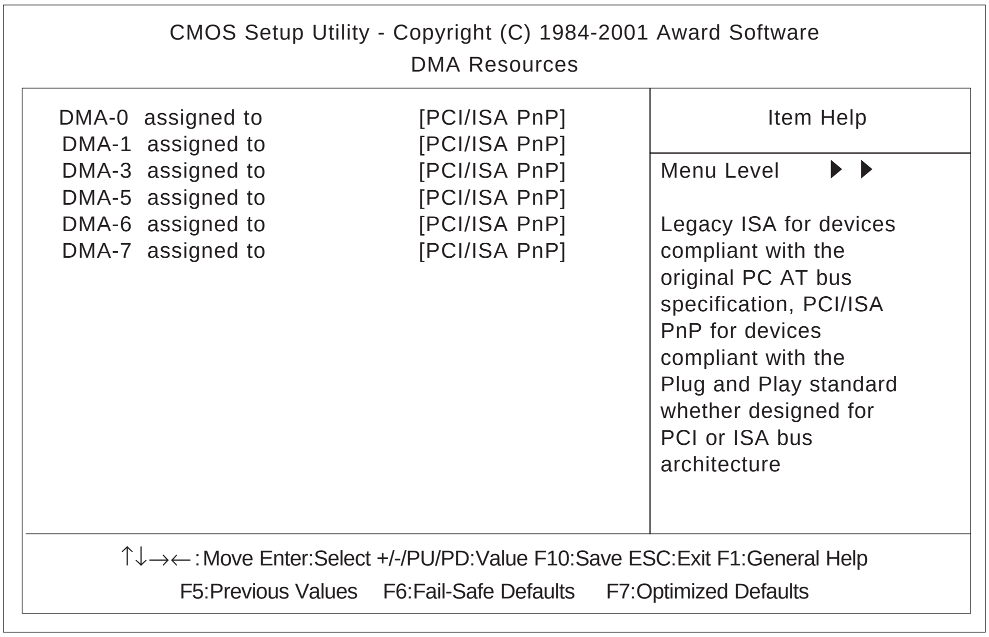
[Legacy ISA] .... Select to use a non-PnP ISA card.

The initial settings are as shown below.

	Initial Value		Initial Value
IRQ-3 assigned to	Legacy ISA	IRQ-10 assigned to	Legacy ISA
IRQ-4 assigned to	Legacy ISA	IRQ-11 assigned to	PCI/ISA PnP
IRQ-5 assigned to	PCI/ISA PnP	IRQ-12 assigned to	Legacy ISA
IRQ-7 assigned to	Legacy ISA	IRQ-14 assigned to	Legacy ISA
IRQ-9 assigned to	Legacy ISA	IRQ-15 assigned to	PCI/ISA PnP

9 DMA Resources

Selecting DMA Resources from the PnP/PCI Configuration menu and the following screen will appear.



```
CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                     DMA Resources

 DMA-0  assigned to          [PCI/ISA PnP]  |        Item Help
 DMA-1  assigned to          [PCI/ISA PnP]  |
 DMA-3  assigned to          [PCI/ISA PnP]  | Menu Level    ▶ ▶
 DMA-5  assigned to          [PCI/ISA PnP]  |
 DMA-6  assigned to          [PCI/ISA PnP]  | Legacy ISA for devices
 DMA-7  assigned to          [PCI/ISA PnP]  | compliant with the
                                            | original PC AT bus
                                            | specification, PCI/ISA
                                            | PnP for devices
                                            | compliant with the
                                            | Plug and Play standard
                                            | whether designed for
                                            | PCI or ISA bus
                                            | architecture

 ↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
    F5:Previous Values    F6:Fail-Safe Defaults    F7:Optimized Defaults
```

■ DMA-0 assigned to ~ DMA-7 assigned to

This setting determines the type of device assigned to the port address. This function is available when the [Resource Control By] option under the [PnP/PCI Configurations] menu is set to [Manual].

[PCI/ISA PnP]...Select to use a PnP PCI or ISA card.

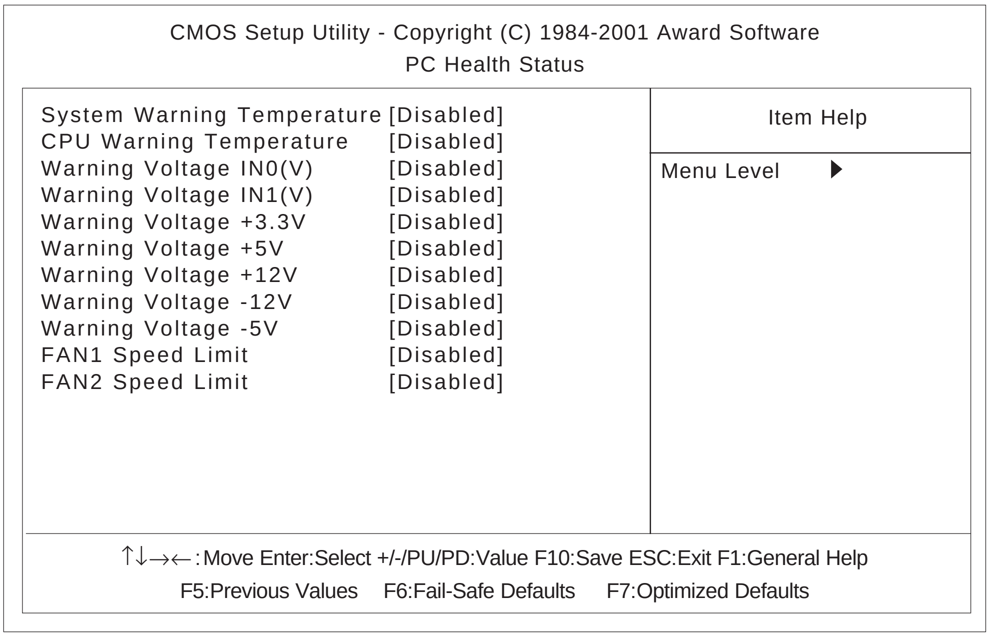
[Legacy ISA]...Select to use a non-PnP ISA card.

The initial settings are as shown below.

	Initial Value		Initial Value
DMA-0 assigned to	PCI/ISA PnP	DMA-5 assigned to	PCI/ISA PnP
DMA-1 assigned to	PCI/ISA PnP	DMA-6 assigned to	PCI/ISA PnP
DMA-3 assigned to	PCI/ISA PnP	DMA-7 assigned to	PCI/ISA PnP

10 PC Health Status

Select PC Health Status from the Main Menu and the following screen will appear.



```
              CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
                                  PC Health Status

 System Warning Temperature [Disabled]                  Item Help
 CPU Warning Temperature    [Disabled]
 Warning Voltage IN0(V)     [Disabled]       Menu Level   ▶
 Warning Voltage IN1(V)     [Disabled]
 Warning Voltage +3.3V      [Disabled]
 Warning Voltage +5V        [Disabled]
 Warning Voltage +12V       [Disabled]
 Warning Voltage -12V       [Disabled]
 Warning Voltage -5V        [Disabled]
 FAN1 Speed Limit           [Disabled]
 FAN2 Speed Limit           [Disabled]

   ↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
       F5:Previous Values   F6:Fail-Safe Defaults   F7:Optimized Defaults
```

- **System Warning Temp**

This setting designates the system temperature at which a warning will be issued. The available settings are [40°C/104°F], [45°C/113°F], [50°C/122°F], [55°C/131°F], [60°C/140°F], [65°C/149°F], [70°C/158°F], [75°C/167°F], [80°C/176°F], [85°C/185°F] and [Disabled]. The factory default setting is [Disabled].

- **CPU Warning Temperature**

This setting designates the CPU temperature at which a warning will be issued. The available settings are [40°C/104°F], [45°C/113°F], [50°C/122°F], [55°C/131°F], [60°C/140°F], [65°C/149°F], [70°C/158°F], [75°C/167°F], [80°C/176°F], [85°C/185°F] and [Disabled]. The factory default setting is [Disabled].

NOTE

- **When using a PentiumIII 1GHz CPU, select the [75°C/167°F] setting. When using a PentiumIII 700MHz CPU, select the [85°C/185°F] setting.**

- **Warning Voltage IN0 (V)**

When enabled, this setting determines the allowable range of the voltage of IN0 (Vcore) and designates that a warning will be issued if the temeperature exceeds that level. The available settings are [+/-6%], [+/-8%] and [Disabled]. The factory default setting is [Disabled].

- **Warning Voltage IN1 (V)**

When enabled, this setting determines IN1's allowable voltage range and designates that a warning will be issued if the temeperature exceeds that level. The available settings are [+/-6%], [+/-8%] and [Disabled]. The factory default setting is [Disabled].

- **Warning Voltage +3.3V**

  When enabled, this setting designates that a warning will be issued if the temeperature exceeds +3.3V's designated range. The available settings are [+/-6%], [+/-8%] and [Disabled]. The factory default setting is [Disabled].

- **Warning Voltage +5V**

  When enabled, this setting designates that a warning will be issued if the temeperature exceeds +5V's designated range. The available settings are [+/-6%], [+/-8%] and [Disabled]. The factory default setting is [Disabled].

- **Warning Voltage +12V**

  When enabled, this setting designates that a warning will be issued if the temeperature exceeds +12V's designated range. The available settings are [+/-6%], [+/-8%] and [Disabled]. The factory default setting is [Disabled].

- **Warning Voltage -12V**

  When enabled, this setting designates that a warning will be issued if the temeperature exceeds -12V's designated range. The available settings are [+/-6%], [+/-8%] and [Disabled]. The factory default setting is [Disabled].

- **Warning Voltage -5V**

  When enabled, this setting designates that a warning will be issued if the temeperature exceeds -5V's designated range. The available settings are [+/-6%], [+/-8%] and [Disabled]. The factory default setting is [Disabled].

- **FAN1 Speed Limit**

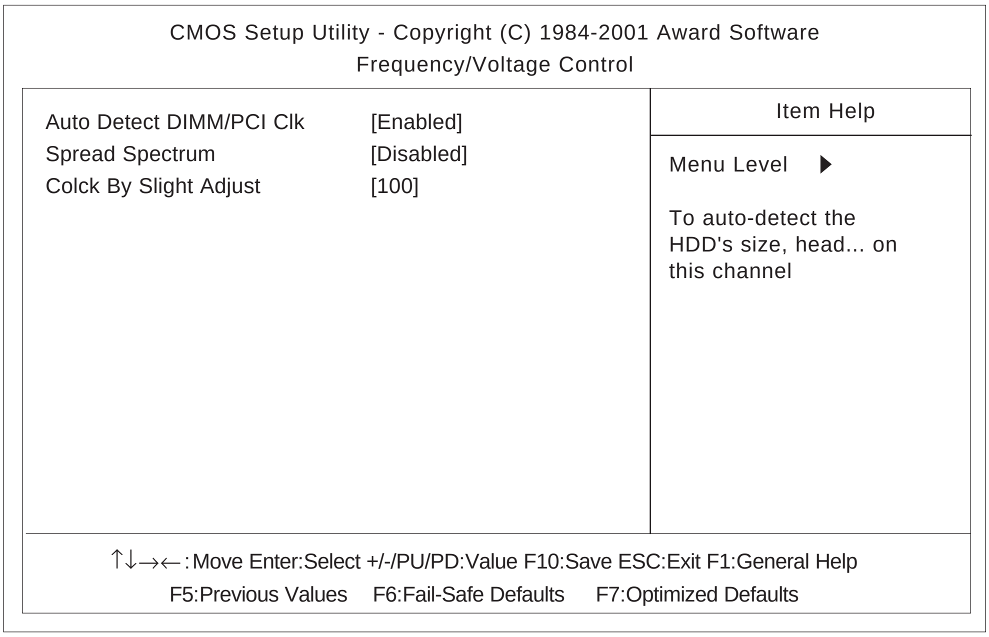
  When enabled, this setting designates the allowed speed range for FAN1. The available settings are [-30%], [-50%] and [Disabled]. The factory default setting is [Disabled]. FAN1 is used for the CPU fan.

- **FAN2 Speed Limit**

  When enabled, this setting designates the allowed speed range for FAN2. The available settings are [-30%], [-50%] and [Disabled]. The factory default setting is [Disabled]. FAN2 is used for the power supply fan.

11 Frequency/Voltage Control

Selecting the Frequency/Voltage Control menu item produces the following screen.



CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
Frequency/Voltage Control

```
Auto Detect DIMM/PCI Clk        [Enabled]
Spread Spectrum                 [Disabled]
Colck By Slight Adjust          [100]
```

Item Help

Menu Level ▶

To auto-detect the HDD's size, head... on this channel

↑↓→←:Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
F5:Previous Values F6:Fail-Safe Defaults F7:Optimized Defaults

- **Auto Detect DIMM/PCI CLK**

  This setting designates the auto detect of the DIMM/PCI clock. The available settings are [Enabled] and [Disabled]. The factory default setting is [Enabled] and strongly recommended for users.

- **Spread Spectrum**

  This setting allows you to set the CPU Clock/Spread Spectrum. The available settings are [Enabled] and [Disabled]. The factory default setting is [Disabled] and strongly recommended for users.

- **Clock By Slight Adjust**

  This setting sets the CPU's clock speed in either of three ranges - 133MHz to 166MHz, 100MHz to 132MHz, or 66MHz to 100MHz, depending on the CPU's host clock. The available settings are [100] to [132]. The factory default setting is [100] and strongly recommended for users.

12 Load Fail-Safe Defaults

When the Menu screen's [Load Fail-Safe Defaults] is selected, you are able to designated if the minimum number of System Settings is used or not. The selections are [Y] and [N].

13 Load Optimized Defaults

Selecting [Load Optimized Defaults" designates whether or not you will set up the PL unit's revert to the PL unit’s factory settings. The selections are [Y] and [N].

NOTE **When the PL unit uses its factory settings, the USB interface cannot be used. For information about connecting the Touch Panel via the USB connector, refer to *1.1.1 Setting Up the Touch Panel Connection*. For information about connecting the USB I/F, refer to *1.1.2 Using the USB Interface*.**

14 Set Supervisor Password

This password is used to change system information settings. It is designed to prevent unapproved users from changing the system information settings. Entering up to 8 characters here will overwrite the current password.

When you wish to have no password, click on the [Enter] key. Next, the words "PASSWORD DISABLE" will appear, providing confirmation that the Password is no longer set.

When password input is required, use the [Advanced BIOS Features] area's [Security Option] feature to enter the password. See 5.2.3 ADVANCED BIOS FEATURES

15 Set User Password

This password is used to view system information settings. It is designed to prevent unapproved users from viewing the system information settings. Entering up to 8 characters here will overwrite the current password.

When you wish to have no password, click on the [Enter] key. Next, the words "PASSWORD DISABLE" will appear, providing confirmation that the Password is no longer set.

When password input is required, use the [Advanced BIOS Features] area's [Security Option] feature to enter the password. See 5.2.3 ADVANCED BIOS FEATURES

- **Selecting either "Set Supervisor Password" or "Set User Password" allows you to view and change System Settings.**
- **When you enter a setting screen, "Set Supervisor Password" allows you to view or modify System Settings, however, "Set User Password" allows you to only view System Settings.**

16 Save & Exit Setup

This feature saves the settings entered in the Setup Utility and restarts the PL unit.

17 Exit Without Setting

This feature quits the Setup Utility program without saving any settings entered.

Chapter

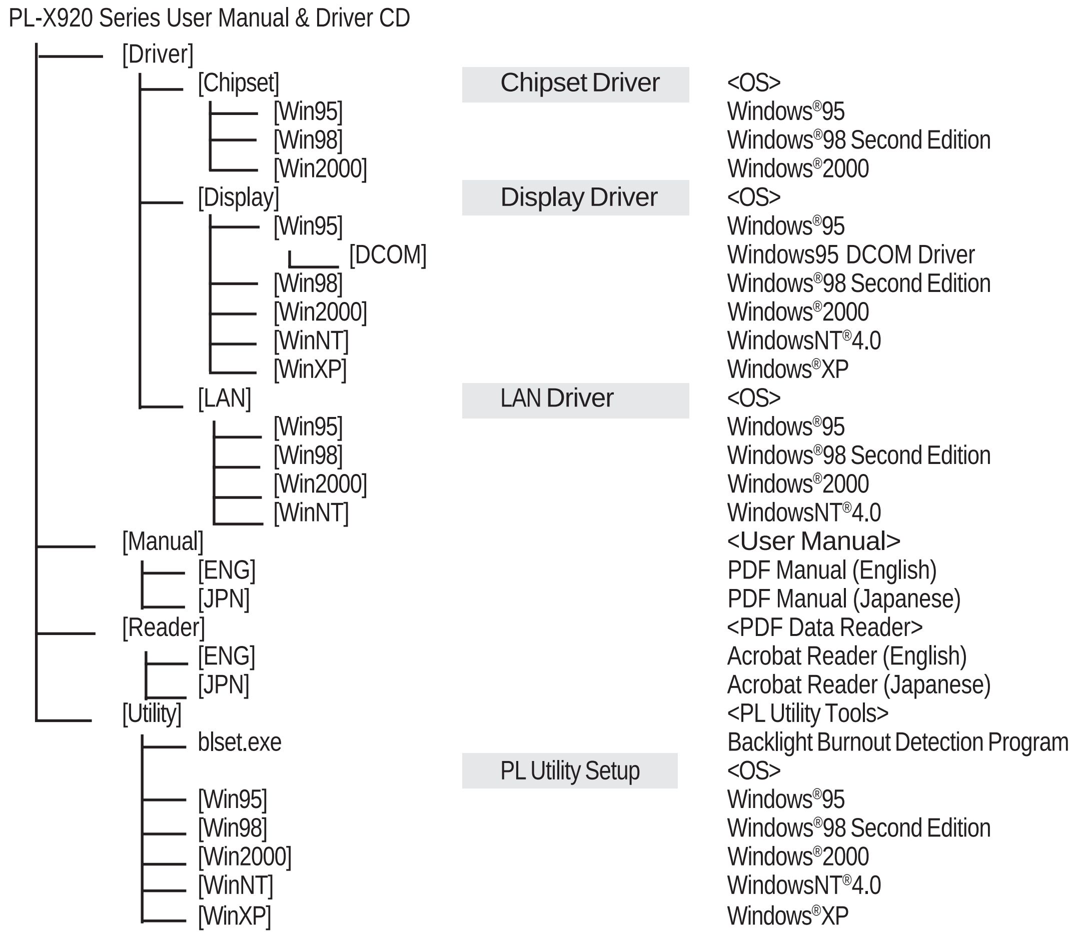
6 Setting Up Your PL OS

Pro-face has prepared the following additional program files which are not supported by the standard versions of the Windows® 95 OSR2 or higher, Windows NT® 4.0, Windows® 98 Second Edition, Windows® 2000 and Windows® XP operating systems. These files are located on the PL unit's additional CD-ROM.

6-1 CD-ROM Contents

1 Diagram

The following tree-structure diagram shows the contents of the CD-ROM disk.



```
PL-X920 Series User Manual & Driver CD
 ├── [Driver]
 │    ├── [Chipset]                 Chipset Driver     <OS>
 │    │    ├── [Win95]                                Windows®95
 │    │    ├── [Win98]                                Windows®98 Second Edition
 │    │    └── [Win2000]                              Windows®2000
 │    ├── [Display]                 Display Driver     <OS>
 │    │    ├── [Win95]                                Windows®95
 │    │    │    └── [DCOM]                            Windows95 DCOM Driver
 │    │    ├── [Win98]                                Windows®98 Second Edition
 │    │    ├── [Win2000]                              Windows®2000
 │    │    ├── [WinNT]                                WindowsNT®4.0
 │    │    └── [WinXP]                                Windows®XP
 │    └── [LAN]                     LAN Driver         <OS>
 │         ├── [Win95]                                Windows®95
 │         ├── [Win98]                                Windows®98 Second Edition
 │         ├── [Win2000]                              Windows®2000
 │         └── [WinNT]                                WindowsNT®4.0
 ├── [Manual]                                          <User Manual>
 │    ├── [ENG]                                        PDF Manual (English)
 │    └── [JPN]                                        PDF Manual (Japanese)
 ├── [Reader]                                          <PDF Data Reader>
 │    ├── [ENG]                                        Acrobat Reader (English)
 │    └── [JPN]                                        Acrobat Reader (Japanese)
 └── [Utility]                                         <PL Utility Tools>
      ├── blset.exe                                    Backlight Burnout Detection Program
      │                             PL Utility Setup   <OS>
      ├── [Win95]                                      Windows®95
      ├── [Win98]                                      Windows®98 Second Edition
      ├── [Win2000]                                    Windows®2000
      ├── [WinNT]                                      WindowsNT®4.0
      └── [WinXP]                                      Windows®XP
```

6-2 Setting Up Your PL OS

Two types of HDD units are available for your PL unit. One type is with no preinstalled OS, the other has a preinstalled OS. Each type of HDD unit will have its own setup procedures. This manual's explanation uses an HDD unit with no proinstalled OS. For instructions of how to set up an HDD unit with a preinstalled OS, refer to that HDD unit's Installation Guide.

1 Setting Up HDD with no installed OS

Prior to using the PL unit with the Windows® 95/Windows® 98 Second Edition/WindowsNT® 4.0/Windows® 2000/Windows® XP operating system, certain utility software must be installed.

■ Installing the HDD Unit

A hard disk unit must be installed in the PL unit. Refer to **PL-HD220 Installation Guide.**

■ System Parameters Setup

System Parameters Setup must also be entered into the PL unit. After these setting are entered, check that the HD drive is correctly recognized by the PL.

Refer to **PL-HD220 Installation Guide.**

■ OS Setup

The PL unit is designed to operate using the following OS types.

CAUTION!

The PL unit is designed to operate under the following standard Windows OS versions. PL operation with any other maker's OS is not guranteed.

Windows® 95 OSR2 or higher

Windows® 98 Second Edition

Windows NT® 4.0 (Windows Service Pack 3 or higher)

Windows® 2000

Windows® XP Professional

■ PL Utility Setup

Use the “PL-X920 Series User Manual & Driver Disk CD” to install the necessary drivers and utility software.

- **To set up the PL, a PS/2 type (Mini DIN) keyboard is required.**
- **To use the PL unit’s touch panel, the Mouse Emulation Software (UPDD) is required. When installing this software, be sure to designate the COM port as COM4. Visit our website (http://www.pro-face.com/otasuke/) and download the Mouse Emulation Software (UPDD).**

◆ Installing Software from the CD-ROM

To install the software on the PL, Digital’s CD-ROM drive (PL-DK200) is required.

To set up the correct software for your PL unit’s OS, be sure to use the “Disk1” folder’s “Setup.exe” file.

Ex. When running Windows® 98 Second Edition, and the CD-ROM drive is “D”:

D:Utility\Win98[*1]\Disk1\Setup.exe

[*1] *Windows® 95 OSR2 or higher:*	*“Win95”*
Windows® 98 Second Edition:	*“Win98”*
Windows NT® 4.0 (Windows Service Pack 3 or higher):	*“WinNT”*
Windows® 2000 :	*“Win2000”*
Windows® XP:	*“WinXP”*

◆ Proface Folder Contents (on PL hard disk)

When you set up the PL unit’s utility software, the folder [Proface] will automatically be created on the C: drive. Inside that folder are the following programs.

[Proface]

[69api]	API-DLL
[Blsaver]	Backlight control screen saver
[Chipset]	Chipset driver (not included in Windows NT® 4.0 or Windows® XP)
[Disp]	Display ON/OFF utility
[Display]	Graphic Accelerator Driver
[Keyclick]	On-screen Keyboard Emulator
[Lan]	LAN driver (not included in Windows® XP)
[Sysmon]	System monitor/RAS application

NOTE

- **The contents of the hard disk (shown above) may change, depending on the type of OS installed.**
- **In your PL hard disk's system folder you will find the following control-related drivers.**

PLSYSMON.VXD Hardware control driver
(Windwos® 95, Windwos® 98 Second Edition)
PLSYSMON.SYS Hardware control driver
(WindowsNT® 4.0, Windows® 2000, Windows® XP)
BLCTRL.VXD Backlight control driver
(Windwos® 95, Windwos® 98 Second Edition)
BLCTRL.SYS Backlight control driver
(WindowsNT® 4.0, Windows® 2000, Windows® XP)

2 Setting Up OS preinstalled HDD

HDD units with preinstalled Windows 2000 OS have had "Service Pack 4" plus all the required drivers and utility programs installed prior to shipping.

HDD units with preinstalled Windows XP OS have had "Service Pack 2" plus all the required drivers and utility programs installed prior to shipping.

■ Installing the HDD Unit

A hard disk unit must be installed in the PL unit. Refer to **OS Preinstalled HDD Installation Guide.**

■ System Parameters Setup

System Parameters Setup must also be entered into the PL unit. After these setting are entered, check that the HD drive is correctly recognized by the PL.

Refer to **OS Preinstalled HDD Installation Guide.**

■ OS preinstalled HDD Setup

You must first set up your HDD unit's preinstalled OS. For setup instructions, refer to your HDD unit's Installation Guide.

◆ Hard Disk files

You will find a folder named "Proface" on your HDD unit (C: drive). The files found on that drive are as follows:

[Proface]

[69api]	API-DLL
[Blsaver]	Backlight control screen saver
[Chipset]	Chipset driver (not included in Windows NT® 4.0 or Windows® XP)
[Disp]	Display ON/OFF utility
[Display]	Graphic Accelerator Driver
[Keyclick]	On-screen Keyboard Emulator
[Lan]	LAN driver (not included in Windows® XP)
[Setup]	Setup (not included in Windows® 95)
[Sysmon]	System monitor/RAS application
[Updd]	Mouse Emulator (not included in PL-B920 Series)

NOTE

- **The contents of the hard disk (shown above) may change, depending on the type of OS installed.**
- **In your PL hard disk's system folder you will find the following control-related drivers.**

PLSYSMON.VXD Hardware control driver (Windwos® 95)
PLSYSMON.SYS Hardware control driver
(WindowsNT® 4.0, Windows® 2000, Windows® XP)
BLCTRL.VXD Backlight control driver (Windwos® 95)
BLCTRL.SYS Backlight control driver
(WindowsNT® 4.0, Windows® 2000, Windows® XP)

6-3 Installing Drivers

In order to use the PL unit’s special features, 3 types of drivers have been created. ([Chipset], [Graphic Accelerator], [LAN].)

If your PL has no pre-installed OS, or has had its OS recovered, please install the following drivers as required.

The following explanation assumes the utility progams have been previously installed on your PL unit’s hard disk in the [Proface] folder.

Drivers set up manually cannot be uninstalled.

■ Installing the Chipset Driver

Use the following explanation to install the PL-X920 series Chipset driver in your PL unit. Installing this driver will cause your OS to recognize your hard disk. This driver, however, cannot be used with Windows NT® 4.0 or Windows® XP.

1) Depending on you PL OS, click on the following file in your Proface folder’s “Chipset” folder. Follow the instructions given to complete the installation.

C:\Proface\Chipset\infinst_enu.exe (Windows® 95)

C:\Proface\Chipset\infinst_autol.exe (Windows® 98, Windows® 2000)

■ Installing the Graphic Accellerator Driver

Use the following explanation to install the PL-X920 series Graphic Accellerator driver in your PL unit. Installing this driver will speed up your PL unit’s display, using special hardware features.

◆ With Windows® 95

1) Prior to installing this driver, start up the following file.

C:\Proface\Display\Dcom\Dcom95.exe

2) Start up C:\Proface\Display\win9xm66.exe. Follow the instructions given to complete the installation.

◆ With Windows® 98 Second Edition, Windows NT® 4.0, Windows® 2000, Windows® XP

1) Depending on your PL OS Start up the following file and follow the instructions given.

C:\Proface\Display\win9xm67.exe (Windows® 98)

C:\Proface\Display\winnt4m67.exe (WindwosNT® 4.0)

C:\Proface\Display\win2k_xpm67.exe (Windwos® 2000, Windows® XP)

NOTE **When your PL OS is WindowsNT®, be sure to use Service Pack 3 or higher.**

◆ With Windows® 98 Second Edition, Windows NT® 4.0, Windows® 2000, Windows® XP

1) Depending on your PL OS Start up the following file and follow the instructions given.

C:\Proface\Display\win9xm67.exe (Windows® 98)

C:\Proface\Display\winnt4m67.exe (WindwosNT® 4.0)

C:\Proface\Display\win2k_xpm67.exe (Windwos® 2000, Windows® XP)

NOTE

When your PL OS is WindowsNT®, be sure to use Service Pack 3 or higher.

■ Installing the LAN Driver

Use the following explanation to install the PL-X920 series LAN driver in your PL unit. Installing this driver allows you to access a LAN.

◆ Installing the LAN Driver

1) Change the BIOS [Integrated Peripherals] menu’s [Onboard LAN] setting to [Enabled]. Refer to **5-2-5 Integrated Peripherals**

2) Start up the PL unit's OS. With Windows® 95 and Windows® 98, the Installation Wizard will appear.

NOTE

When your PL OS is Windows®XP, the LAN Driver is automatically installed after changed the BIOS setting.

NOTE

◆ With Windows® 95

Be sure the PL unit's optional CD-ROM drive (PL-DK200) is connected and operating correctly prior to inserting the Windows® 95 OS CD-ROM into the CD-ROM drive.

1) Click on [Next].

2) Click on [Other Locations].

The location designation wizard will appear.

3) Enter “C:\Proface\lan “ and click [OK].

4) Enter “C:\Proface\lan “ and click [OK] again.

The device driver wizard will appear.

5) Click on [Finish].

6) Enter “D:\Win95“ and click on [OK].

Files will be copied from the Windows95 CD-ROM to the PL unit.

7) Click on [Yes] and restart the PL unit to complete the installation.

◆ With Windows® 98 Second Edition

NOTE

Be sure the PL unit's optional CD-ROM drive (PL-DK200) is connected and operating correctly prior to inserting the Windows® 98 OS CD-ROM into the CD-ROM drive.

1) Click on [Next].

2) Select [Search for a better driver than the one your device is using now. (Recommended)] and click on [Next].

3) Click on the [Specify a location] check box, enter "C:\Proface\Lan" in the location window, and click on [Next].

4) Click on [Next].

The file copy dialog box will appear and files will be copied from the Windows98 CD-ROM.

5) Click on [Finish]

The system settings dialog box will appear.

6) Click on [Yes] to restart your PL.

◆ With Windows NT® 4.0

Click the [Start] button, point to [Settings] and click on [Control Panel (C)].

1) Double click on the [Control Panel]’s [Network] icon.

The [Network Configuration] dialog box will appear.

2) Click on [Yes].

The Network Setup Wizard will appear.

3)Select [Wired to the network:], and click on [Next].

4) Click on [Select from list].

The Network Adaptor selection dialog box will appear.

5) Click on [Have disk].

The "Insert floppy disk" dialog box will appear.

6) Enter “C:\Proface\lan” and click [OK].

The "Select OEM Option" dialog box will appear.

7) Click on [OK].

The Network Setup wizard will appear.

8) Click on [Next].

9) Select the desired network protocol and click on [Next].

10) Select the desired service to install and click on [Next].

11) Click on [Next].

The WindowsNT setup dialog box will appear.

12) Enter “D:\I386” and click [Continue].

13) Enter “C:\Proface\lan” and click [Continue].

The "Speed /Duplex mode" dialog box will appear.

14) Click on [Continue].

The "Input Network Address" dialog box will appear.

15) Click on [OK].

The "Input Tx Early Threshold" dialog box will appear.

16) Click on [OK].

The TCP/IP dialog box will appear.

17) Enter the settings to connect with your network.

The Network Setup Wizard will appear.

18) Click on [Next].

19) Click on [Next].

20) Enter the Computer and Workgroup names, and click on [Next].

21) Click on [Finish].

The Network Settings Change dialog box will appear.

22) Click on [Yes(Y)] to restart your PL.

The Service Control Manager dialog box will appear.

23) Restarting the PL will cause an error message to appear, which requires the PL's Service Pack to be reinstalled.

24) After the Service Pack is reinstalled, restart the PL.

◆ With Windows® 2000

Click the [Start] button, point to [Settings] and click on [Control Panel (C)].

1) Double click on the [Control Panel]'s [Network and Dial-Up Connections] icon.

The Network and Dialup Connection window will appear.

2) Right click on the [Local Area Connection] icon, and select [Properties].

The Local Area Connection properties will appear.

3) Click on [Configure].

The Realtek RTL8139(A) PCI Fast Ethernet Adapter properties will appear.

4) Click on [Update Driver].

The Upgrade Device Driver Wizard will appear.

5) Click on [Next].

6) Select [Search for a suitable driver for my device (recommended)] and click [Next].

7) Select [Specify a location] and click [Next].

8) Enter “C:\Proface\lan “ and click [OK].

The search for the driver will start.

9) Click on [Next].

10) Click on [Finish] to restart your PL.

6-4 Application Features

The PL unit is equipped with the following special features. The following files have been copied to the PL unit's hard disk and are contained in the [Proface] folder.

File Name	Windows®95/Windows®98 Second Edition	Windows NT®4.0/ Windows®2000	Windows®XP
PL_BLIOC.DLL	C:\Windows\System	C:\Winnt\System32	C:\Windows\System32
PL_DLL.DLL			
PL_IOC.DLL			
Backlight Control.scr			
Disp.exe	C:\Proface\Disp		
Keyclick.exe	C:\Proface\Keyclick		
PL_Smon.exe	C:\Proface\Sysmon		
PL_Wps.exe	C:\Proface\Sysmon		
BLSET.EXE	Found inside the CD-ROM [PL-X920 Series User Manual & Driver CD], in the Utility\Blset.exe		

■ API-DLL

This is a dynamic library designed to provide access to the System BIOS' RAS feature for User applications. API-DLL consists of three types, which are explained below.

◆ Backlight Control API-DLL file (PL_BLIOC.dll)

This API-DLL file provides a dynamic library that allows User-created applications to utilize the PL-X920 series' backlight control feature. This file must be installed into the

same directory as the User's application.

For details, refer to the ***Appendix 4 - Backlight Control Feature API-DLL***

◆ System Monitor API-DLL (PL_DLL.DLL)

This API-DLL file provides a dynamic library that allows User-created applications to utilize the PL-X920 series' System Monitor feature. This file must be installed into the same directory as the User's application.

◆ RAS Feature API-DLL (PL_IOC.DLL)

This API-DLL file provides a dynamic library that allows User-created applications to utilize the PL-X920 series' System BIOS' RAS feature.

For details, refer to the ***Appendix 3 - System Monitor/RAS Feature API-DLL***

■ Backlight OFF Screen Saver (Backlight control.scr)

This software is used to turn OFF the PL's backlight after a specified period of inactivity. The use of this feature will help to extend the life of the PL's backlight.

NOTE

Certain application programs may not allow the PL's backlight to turn OFF. Please test each program individually to check if the screen saver will operate correctly.

■ Screen Display ON/OFF Utility (DISP.EXE)

This command line utility is used to turn OFF both the PL’s backlight and display.

Settings Used	DISP [ON/OFF]
Option Switch	ON: Displayed / OFF: Not Displayed
Return Value	0: Completed Normally / -1: Option Switch Error

■ Keyboard Emulator (Keyclick32)

This program allows the User’s mouse operation to perform keyboard-like data input.

It is required to agree to "IN-fINITY soft Keyclick32 License Agreement".

Refer to ***Appendix 6 Consent Agreement***

- **Certain application programs do not support this keyboard emulator. Please test each application individually to check if the keyboard emulator will operate correctly.**
- **This application cannot be used to enter Windows® startup screen User Name and Password information.**
- **To change the Keyclick program’s font size a keyboard is required.**

- **For details concerning the Keyclick program’s operation, simply click on the HELP button to call up the program’s online help data.**

■ System Monitor/RAS Application PL_SMON.EXE / PL_WPS.EXE

This utility provides monitoring of the PL's temperature, voltage level, and fan's operation, via the system BIOS' RAS and system monitoring functions.

◆ System Monitor Program Pl_Smon.exe

For details, refer to ***Appendix 3.3 System Monitor Operation***

◆ Monitor Parameter Setting Program Pl_Wps.exe

For details, refer to ***Appendix 3.2 System Monitor Property Settings***

■ Backlight Burnout Detection Program (BLSET.EXE)

PL-6920 Series unit’s only

This feature designates if the PL unit's touch panel is disabled or enabled when a backlight burnout occurs.The following program runs under MS-DOS®.

BLSET ON ↵ or BLSET OFF ↵

ON designates that the Touch Panel is disabled when a backlight burnout occurs.

OFF designates that the Touch Panel remains enabled when a backlight burnout occurs.

The factory setting is OFF.

For backlight burnout detection details, Refer to ***Appendix 2 RAS Feature***

1 Uninstalling PL-X920 Driver and Utility

1) Click the Windows main screen's bottom left corner [Start] button.

2) Click on [Settings] and then on [Control Panel].

3) Click the Control Panel's [Add/Remove Programs] icon.

4) Select the "PL-X920 Driver and Utility" and click on the Add/Remove button to remove the software.

CAUTION! ***All drivers installed with this program cannot be uninstalled.***

6-5 Windows NT® 4.0 / Windows ®2000 /Windows ®XP Cautions

Perform the following settings as required by your OS.

1 Automatic System Log-On Setup

■ **When using Windows NT® 4.0**

1) Click on the main screen's "Start" button, and select the "Enter Filename" item. Enter the text "C:\WINNT\REGEDIT.EXE" and press [Enter] to start the program.

2) When the REGEDIT Registry Tree appears, select the "Winlogon" subkey via the following text:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\ Current Version\Winlogon.

3) In the "DefaultUserName" field, enter the User name to be used for the Automatic Log-on.

4) Select the Edit menu's [New/String Value] feature.

5) To the Data Items present, add "AutoAdminLogon" to the Name column's data, and then enter "1" in that entry's Data field.

6) Add "Default Password" to the Name column's data, and enter the password used previously for the DefaultUserName in the Data field.

CAUTION! ***A user with no password cannot automatically log on.***

7) REGEDIT data entry is now finished.

NOTE

- **If a User attempting to automatically log on is not attached to an "Administrators" group, i.e. no Default Password string is specified, Windows NT automatically changes the value of the AutoAdminLogon key from 1(true) to 0(false), thereby disabling the AutoAdminLogon feature. In that case, if the Shift key is held down during Logoff, the "Login Data" dialog box will appear and the User can use the name of another, registered User to log-on successfully.**
- **If Auto LogOn Setting are not entered, when logging on, a PS/2 type keyboard is required.**

■ **When using Windows® 2000**

1) Start the Control Panel’s [Users and Passwords].

2) Select an automatic log-on user and deselect the [Users must enter a user name and password to use this computer] checkbox.

3) Click on the [Advancedl] tab and deselect the [Require users to press Ctrl-Alt-Delete before logging on] checkbox.

4) Click the [Apply] button and when the automatic login dialog box appears, enter your password.

■ **When using Windows® XP**

(1) Select [Run] from the [Start] menu.

(2) Enter “Control userpasswords2”, and click [OK].

(3) Deselect the [User must enter the user name and password to use this computer]

checkbox and click [Apply].

(4) Enter the user name and password for automatic logon and click [OK].

2 Using an Uninterrupted Power Supply

Prior to turning OFF the PL's power, be sure to shut down the NT correctly via the NT OS' "Shutdown" feature. It is recommended that an Uninterrupted Power Supply Device is used to prevent the accidental loss of User data, due to an unexpected power outage.

When using an Uninterrupted Power Supply (UPS), the unit can be set to switch to backup power, which will provide enough time to safely shut down your PL, or it can even shut down your PL for you.

For details, please consult your local dealer of UPS units.

3 When Changing the System Design

When the PL unit is connected to a printer or to a LAN network, the Windows system settings must be changed.

■ **When using Windows NT® 4.0**

- **Changing the System Design**

  When the Windows NT® 4.0 system design is changed, the following messages will appear.

> **Windows NT Setup**
>
> **Setup needs to copy some Windows NT files**
>
> **Setup will look for the files in the location specified below. If you want Setup to look in a different place, type the new location. When the location is corrent, click Continue.**

> **Files Needed**
> **Some files on WindowsNT Workstation CD-ROM are needed. Insert WindowsNT Workstation CD-ROM into the drive selected below, and then click OK.**

Even when one of these messages appears, designate a new location for the system design change folder (Windows NT® 4.0 CD-ROM's [I386]) and click on [Next].

D:\I386 (CD-ROM drive is "D")

- **Reinstalling Service Pack Data**

  When changing the Windows NT system settings, the system files are written over your existing Service Pack 1 files. Be sure to set up your Service Pack data again.

■ **When using Windows® 2000/Windows® XP**

- **Changing the System Design**

When the Windows® system configuration settings are changed, the following message appears. Enter the name of the folder where you will save the configuration changes and click [OK].

> **• Insert the CD labeled "Windows® xx*1 Professional CD-ROM" in the CD-ROM drive (D:), and click [OK].**
>
> **• When copying the file from other locations such as a floppy disk or network server, also click the [OK] button.**

- **When Your PL has No Preinstalled OS**

  Double-click the [I386] folder in the Windows® CD-ROM.

  D:\I386 (When the CD-ROM drive is "D")

- **When Your PL has a Preinstalled OS**

  C:\Proface\Setup\I386

4 Changing to the NTFS File System

■ **With a hard disk using WindowsNT® 4.0, Windows® 2000 and Windows® XP**

If your hard disk was formatted using the Windows DOS compatible FAT32 system, you can use Command Prompt to convert the hard disk to an NTFS system. Use the following command.

convert x:/fs:ntfs, where "x" is the drive name of your hard disk.

NOTE

- **After converting data to the NTFS file system, it cannot be converted back to the FAT32 (DOS compatible) file system.**

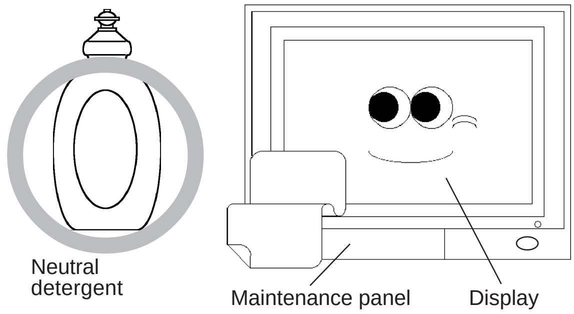
*1"xx" is your OS. Example) When using Windows®XP: "Windows®XP Professional CD-ROM"

Chapter 7

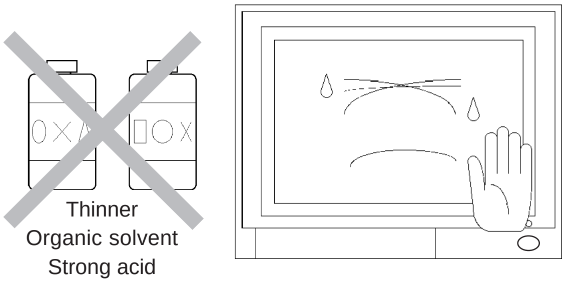
Maintenance and Inspection

7-1 Regular Cleaning

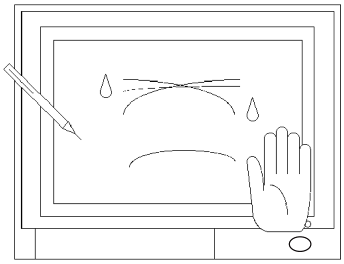
1 Cleaning the Display



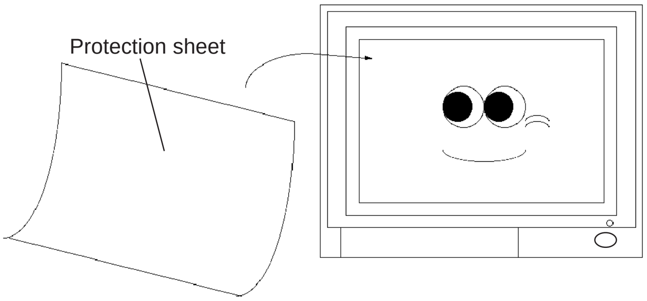
When the display surface or frame become dirty, use a soft cloth moistened with neutral detergent to wipe away any dust or stains.



Do not clean the unit with thinner, organic solvents, or strong acids.



Do not use sharp or hard objects, such as a mechanical pencil or screwdriver, to push on the display. This could damage the unit.



Use the screen protection sheet when using the PL in extremely dirty or dusty areas.

2 Installation Gasket Replacement

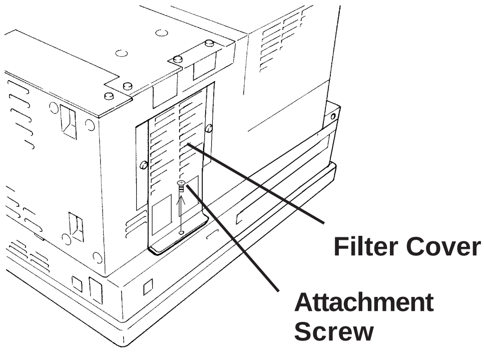
The moisture resistant gasket protects the PL and improves its water resistance. For instructions on installing the PL's gasket. ☞ ***4-1-2 Installation Procedures***

A gasket which has been used for a long period of time may have scratches or dirt on it, and could have lost much of its water resistance. Be sure to change the gasket periodically (or when scratches or dirt become visible).

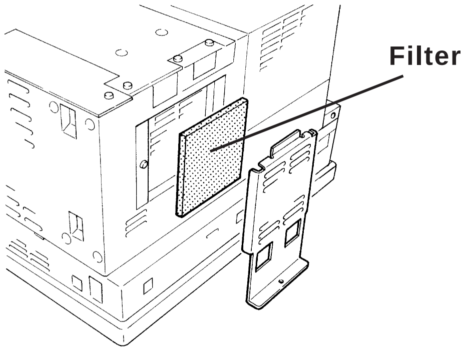
7-2 Cleaning the Filter

Since excessive dirt and dust in the filter of the PL's electric fan can potentially affect the performance of the unit, regular inspection and cleaning of the filter is strongly recommended.

■ PL6920/PL-7920 (4-slot type)



1) Unscrew the electric fan filter cover's attachment screw and remove the filter cover.

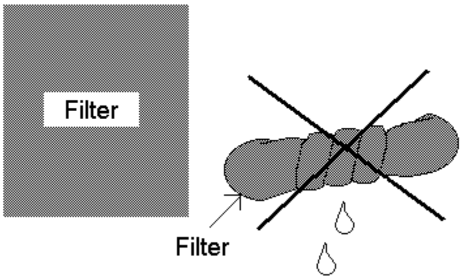


2) Remove the filter from the fan cover.

3) Clean the filter completely.

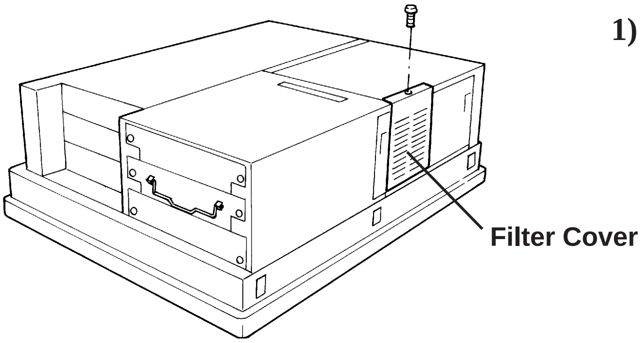
If stubborn dirt or stains are present, wash the filter with a neutral detergent.

- **Do not wring the filter to dry it.**
- **When drying the filter, dry it without exposing it to direct sunlight.**
- **Allow it to air dry completely before reattaching to the PL.**

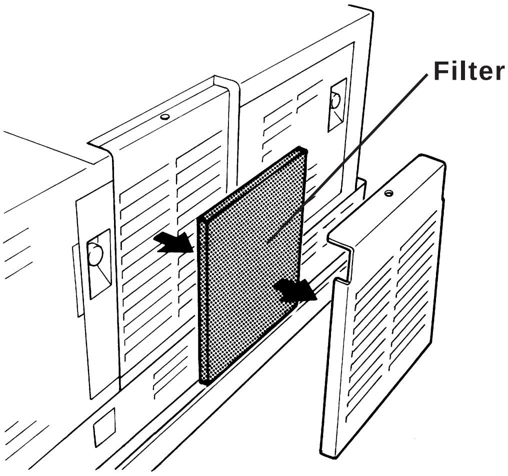


4) After reattaching the filter to the filter cover, reattach the cover with its cover attachment screw.

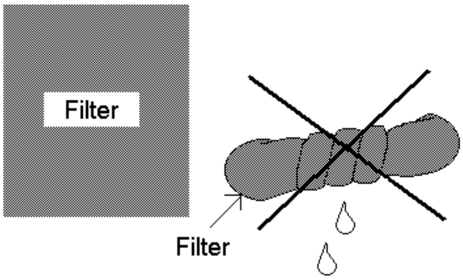
■ PL6921/PL-7920 (2-slot type)



1) Remove the electric fan filter cover's attachment screw.



2) Remove the filter from the fan cover.



3) Clean the filter completely.

If stubborn dirt or stains are present, wash the filter with a neutral detergent.

CAUTION!

- **Do not wring the filter to dry it.**
- **When drying the filter, dry it without exposing it to direct sunlight.**
- **Allow it to air dry completely before reattaching to the PL.**

4) After reattaching the filter to the filter cover, reattach the cover with its cover attachment screw.

7-3 Changing the PL Backlight

The PL's backlight can be changed after it wears out. The steps involved are outlined below.

☞ ***The steps for removing the Rear Maintenance Cover and the Half Cover are the same for each PL unit.***
Please use the following table to identify which backlight model number to use when ordering your backlight.

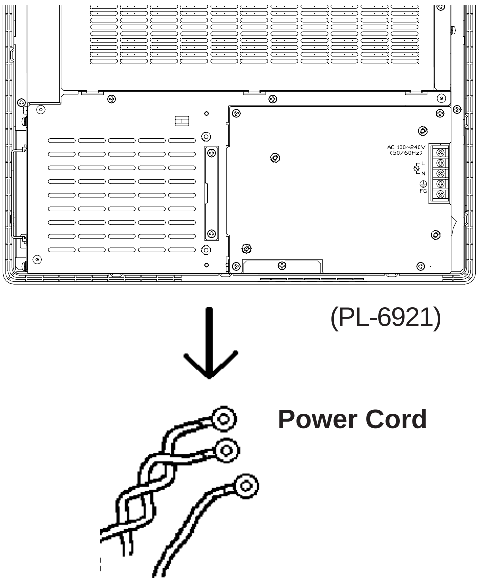
PL Type	Backlight Type
PL- 6920	PL6920-BL00
PL- 6921	
PL- 7920	PL790-BL00-MS
PL- 7921	

⚠ WARNINGS

- Whenever changing the backlight, be sure the PL's power cord has been disconnected and that the unit is cooled down.
- When the PL's power cord is connected and the PL is ON, high voltage runs through the wires in the backlight area—*do not touch them!*
- When the PL's power has just been turned OFF, the backlight area is still very hot! Be sure to wear gloves to prevent being burned.
- Do not try to replace the backlight while the PL is installed in a cabinet or panel. Remove the PL first, then begin the backlight replacement procedures.

Follow the steps given below to change the PL's backlight. Be sure to wear cotton gloves when performing this work to prevent burns. Also, remove the PL from its installation panel, and place it face down (as shown below) when performing this work.

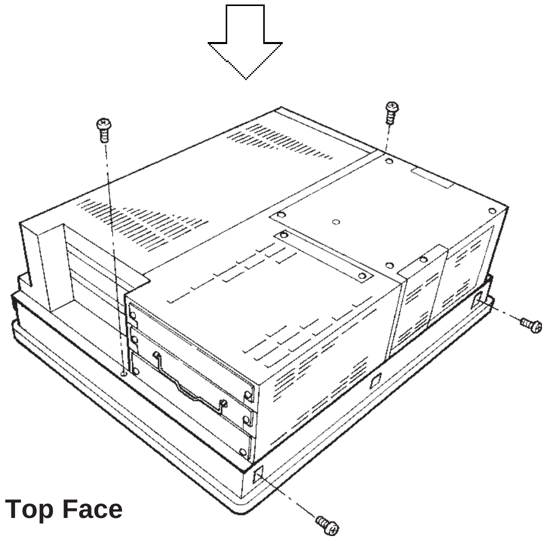
■ PL-6920 Series



1) Unplug the PL's power cord from the main power supply and then disconnect the PL power cord terminals from the PL's power terminal block.

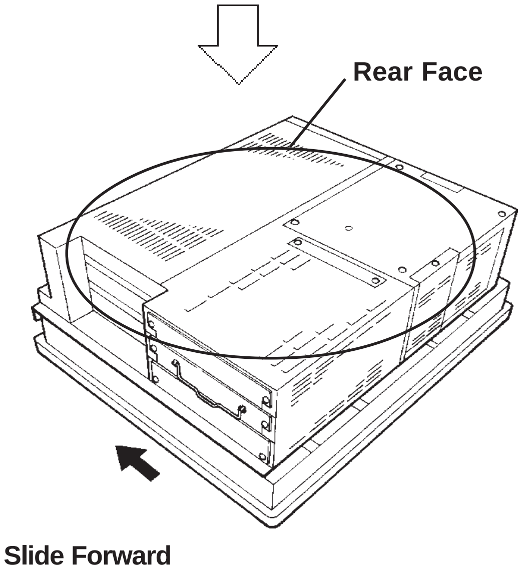
CAUTION!

- **Be sure to perform the backlight changeover on a flat, level surface. This will prevent damage to the PL unit and the accidental cutting of any of its power cord terminals.**

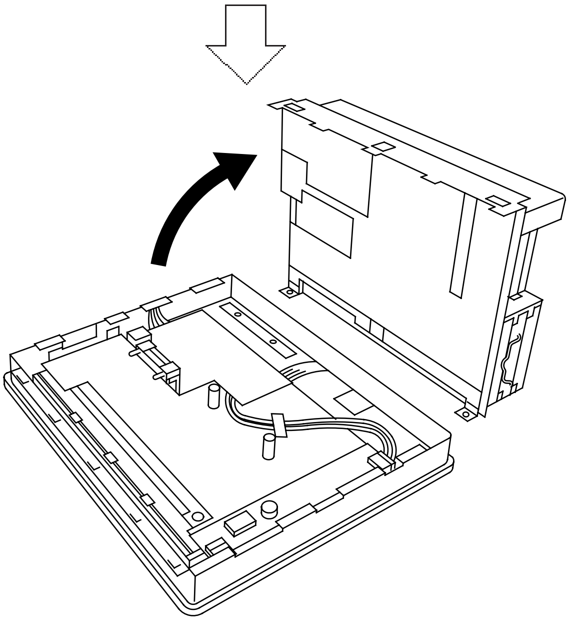


2) Remove the PL’s four chassis attachment screws.

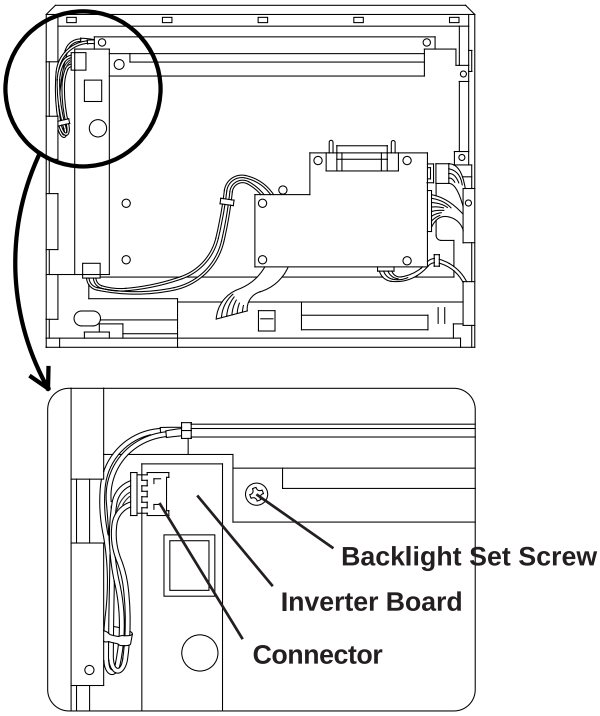
CAUTION! If the PL-FD210 is installed, be sure to remove it



3) Hold both sides of the front panel section and use your thumbs to slide (push) the chassis forward (see figure).



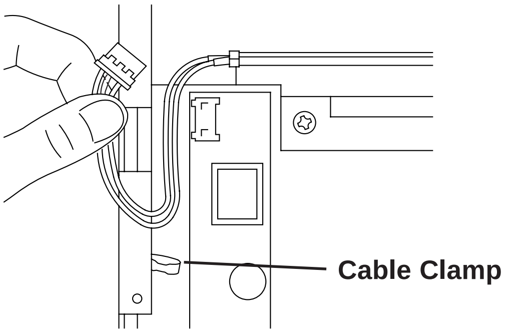
4) The chassis can then be pivoted back and open.



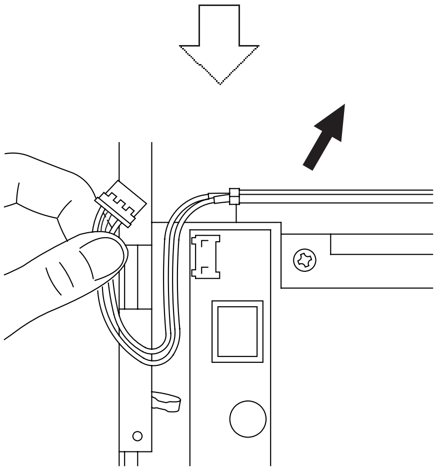
5) Disconnect the power connectors from the inverter board.

6) Loosen the backlight's set screw.

CAUTION! • Use a "1" sized Phillips screwdriver to remove the backlight attachment screws.

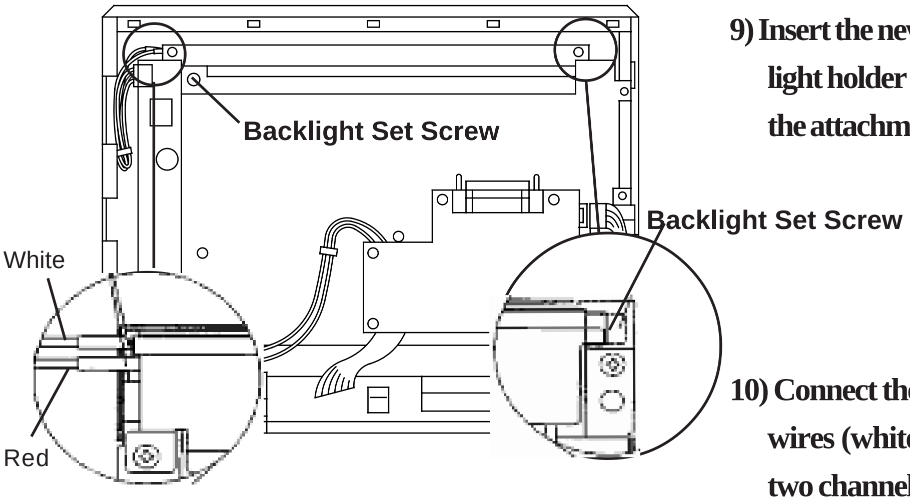


7) Open the cable clamp and release the cable.



8) Gently lift up on the power connector end of the backlight to free it and remove it from the PL.

CAUTION! • The entire backlight unit should be changed, not just the backlight.



9) Insert the new backlight into the backlight holder frame and push it in until the attachment clip clicks into place.

10) Connect the backlight's power cable wires (white and red) to the frame's two channels. Insert the one (1) white wire into the upper channel, and the two (2) red wires into the lower channel.

11) Tighten the backlight's set screw. (Use no more than 0.19N•m of force.)

12) Connect the backlight power cord's connector to the PL's inverter board.

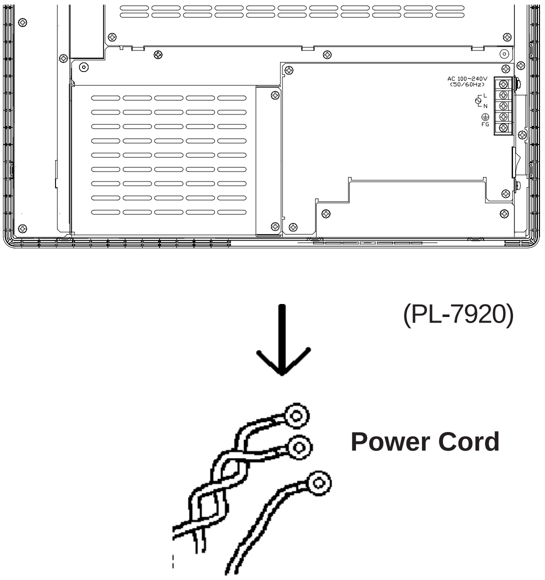
CAUTION!
- Be sure the connector is inserted completely into the inverter board. An incompletely connected connector could cause an accident.

13) Replace the PL's rear cover, being careful not to pinch or cut any of the PL's internal wires.

CAUTION!
- Be sure the backlight cables are completely secured by the cable wrap prior to connecting the PL's front and rear sections.

14) Secure all four (4) or the PL's attachment screws.

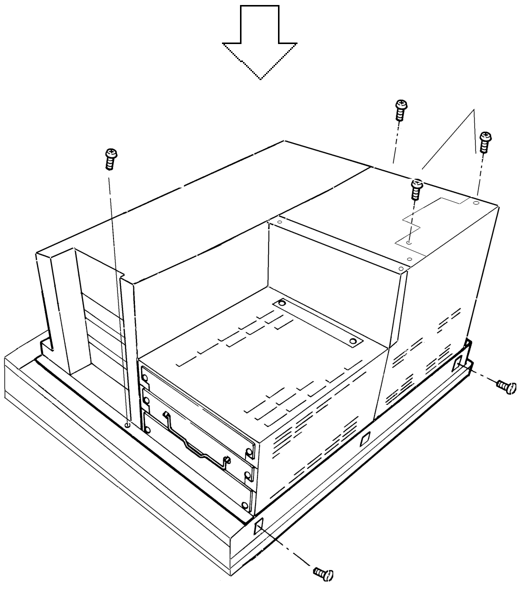
■ PL-7920 Series



1) Unplug the PL's power cord from the main power supply and then disconnect the PL power cord terminals from the PL's power terminal block.

CAUTION!

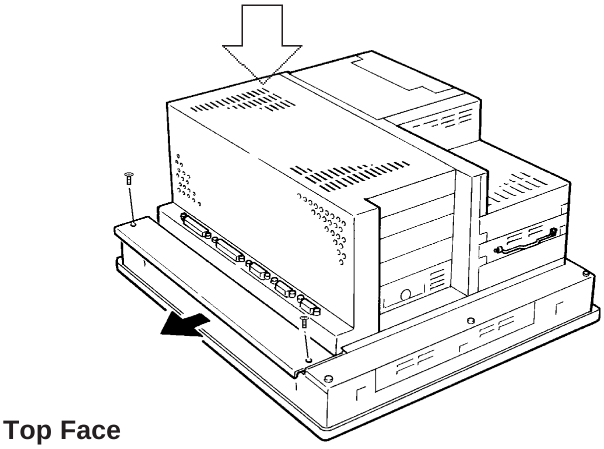
- Be sure to perform the backlight changeover on a flat, level surface. This will prevent damage to the PL unit and the accidental cutting of any of its power cord terminals.



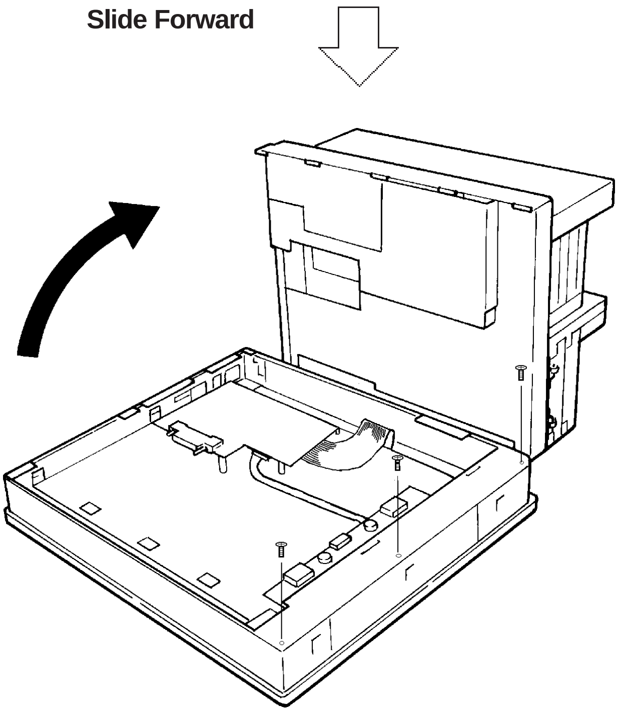
2) Remove the PL’s four (4) chassis attachment screws.

CAUTION!

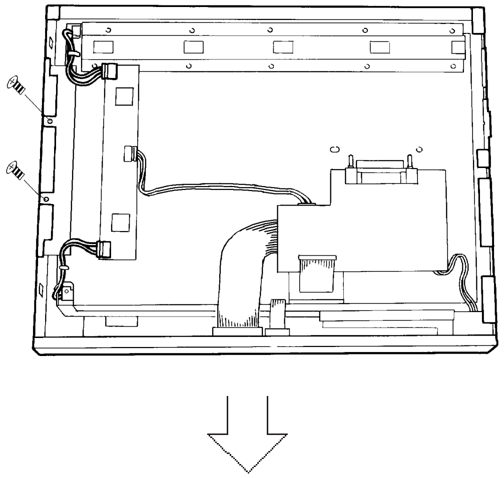
If the PL-FD210 is installed, be sure to remove it also. (Only with PL-7920)



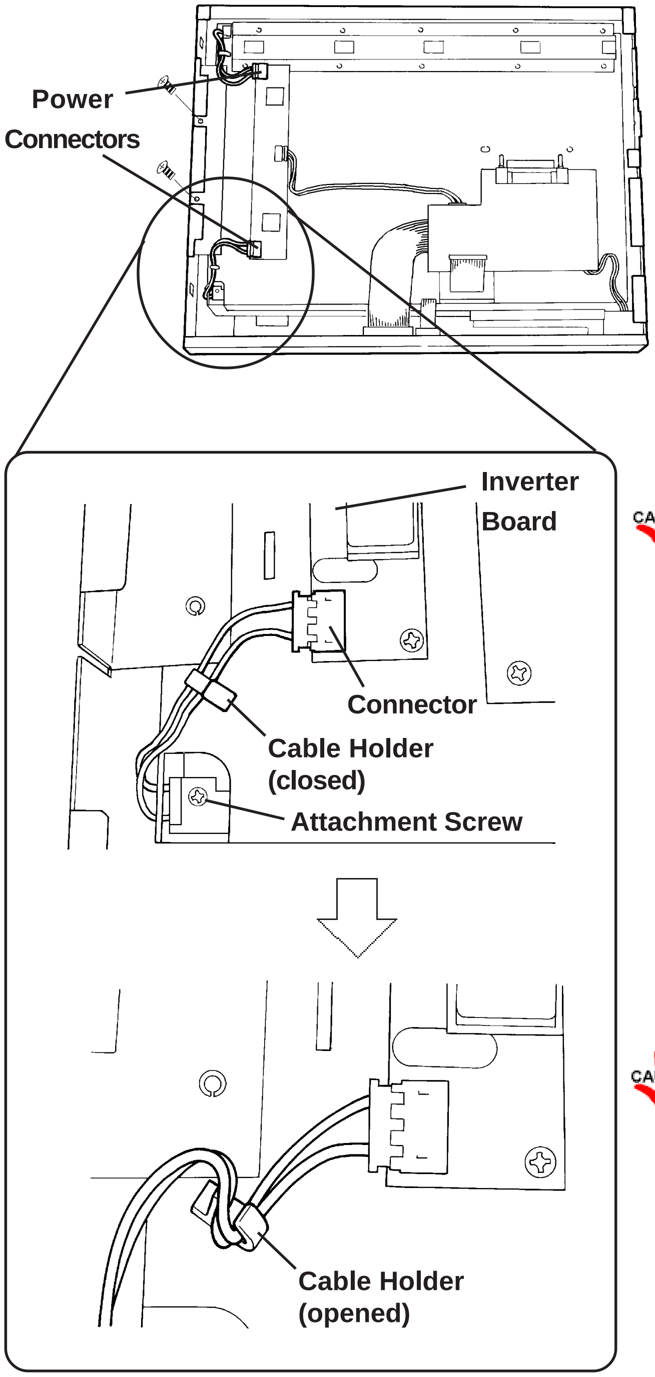
3) Remove the two (2) attachment screws, hold both sides of the front panel section and use your thumbs to slide (push) the chassis forward (see figure).



4) The chassis can then be pivoted back and open.
Remove the front face cover’s three (3) attachment screws.



6) The PL’s LCD display unit is fixed in place with two attachment screws.
Remove these screws.



7) Disconnect both backlight power connectors from the inverter board.

8) Each backlight (upper and lower) is fixed in place with an attachment screw. Remove each of these screws.

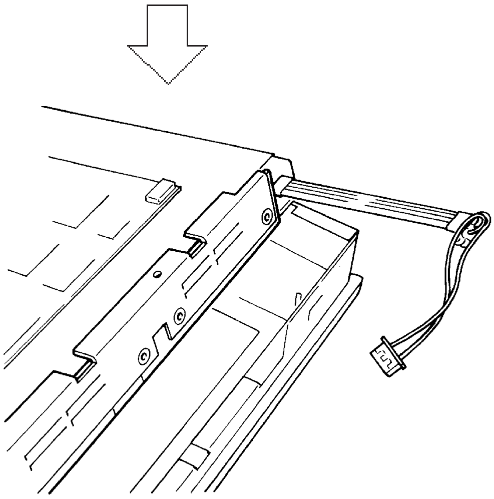
CAUTION!

- ***Use an"0" sized Phillips screwdriver to remove the backlight attachment screws.***
- ***Be careful not to lose any of the small attachment screws.***
- ***Do not let any of attachment screws fall inside the PL unit's chassis or front panel area, since it could cause an electrical short.***

9) Open/release each plastic cable holder.

CAUTION!

Each backlight uses a cable holder to prevent the cable from becoming caught or pinched when the PL's rear cover is opened or closed. Be sure both cables are secured by their holder before closing the rear cover.



10) Remove each backlight unit.

11) Insert each new backlight.

12) After installing both new backlights, replace and tighten all backlight and LCD panel attachment screws (total of 4 screws). Then, reconnect both inverter board power connectors.

CAUTION!

- *Be sure to always change both of the PL backlights at the same time.*
- *Be sure that the backlight's power cord connector is inserted completely into the Inverter Board's receptacle. If not, the PL may be damaged.*
- *The PL-FD210 can only be used in the PL-7920.*

13) Reassemble the rear cover, being careful not to pinch any internal electrical wires.

14) Replace the PL-FD210's two attachment screws (only PL-7900).

15) Replace and tighten the PL's four chassis attachment screws.

7-4 Periodic Inspection Items

Be sure to inspect the PL periodically to ensure it is in good working condition.

■ Ambient environment check points

- Is the ambient temperature within the specified range?

Ambient Operating Temperature	PL692*-T41 (CPU:700MHz)	Using Fan		5 °C to 50 °C (With HDD attached)
		Not using fan [*1]		5 °C to 40 °C (With HDD attached)
	PL792*-T41 (CPU:700MHz)	Cabinet Interior	Using Fan	5 °C to 50 °C (With HDD attached)
			Not using fan*1	5 °C to 40 °C (With HDD attached)
		Panel face	5 °C to 40 °C	
	PL692*-T42 (CPU:1GHz)	Using Fan		5 °C to 45 °C (With HDD attached)
		Not using fan [*1]		Must use fan
	PL792*-T42 (CPU:1GHz)	Cabinet Interior	Using Fan	5 °C to 45 °C (With HDD attached)
			Not using fan*1	Must use fan
		Panel face	5 °C to 40 °C	

***1** *When the PL's internal electric fan is removed.*

- Is the ambient humidity within the specified range (10%RH to 85%RH) ?
- Is the atmosphere free of corrosive gas ?

■ Electrical specifications check

- Is the voltage adequate (AC85V to AC265V, 50/60 Hz) ?

■ Installation check points

- Are all connection cables firmly connected ?
- Are any bolts or screws loose ?
- Are there any flaws or tears in the moisture resistant gasket ?

■ Display check

- Is the display bright enough ?

NOTE When the PL's backlight needs to be replaced, please contact your nearest authorized service center for prompt service.

Appendices

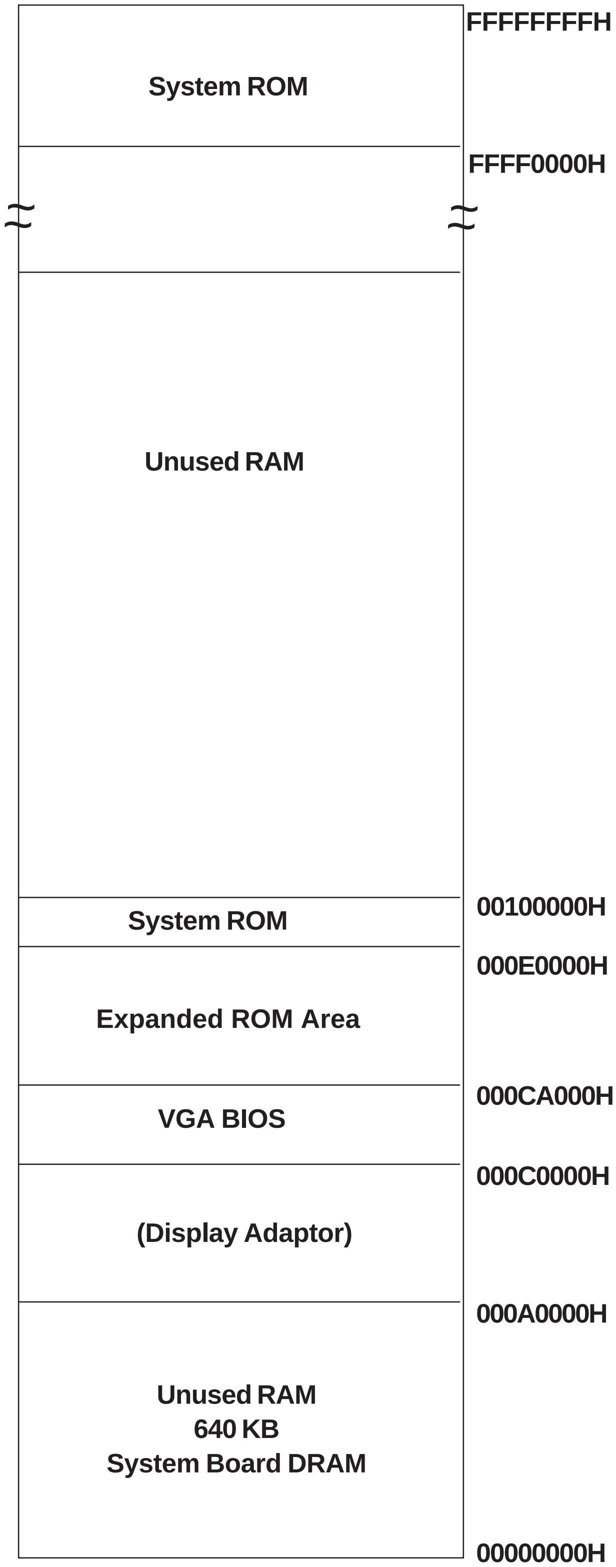
A-1 Hardware Configuration

The following data explains the design of the I/O Map, Memory Map and Interrupt Map, as well as additional hardware design items including the RAS feature.

1 I/O Map

Address	AT System Device	System Device
0000H - 001FH	DMA controller (8237)	
0020H - 003FH	Interrupt controller (8259A)	
0040H - 005FH	System timer (8254)	
0060H - 006FH	Keyboard Controller	
0070H - 007FH	Real-time clock, NMI mask	
0080H - 009FH	DMA page register	
00A0H - 00BFH	Interrupt controller 2 (8259A)	
00C0H - 00DFH	DMA controller 2 (8237)	
00F0H - 00FFH	Numeric data processor	
01F0H - 01FFH	Hard disk (IDE)	
0200H - 0207H	Game I/O	
0290H - 029FH	Reserved	
02E8H - 02EFH	Reserved	Touch Panel Serial Port 4 (COM4)
02F8H - 02FFH	Serial port 2 (COM2) : General Use	
03B0H - 03BBH	Video controller (VGA)	
03BCH - 03BFH	Parallel port 1 (LPT1)	
03C0H - 03DFH	Video controller (VGA)	
03E8H - 03EFH	Reserved	Serial port 3 (COM3)
03F0H - 03F7H	Floppy disk controller	
03F8H - 03FFH	Serial port 1 (COM1) :General Use	

2 Memory Map


FFFFFFFFH
System ROM
FFFF0000H
Unused RAM
00100000H
System ROM
000E0000H
Expanded ROM Area
000CA000H
VGA BIOS
000C0000H
(Display Adaptor)
000A0000H
Unused RAM
640 KB
System Board DRAM
00000000H

3 Interrupt Map

■ Hardware Interrupt List

	Description
NMI 0	Parity Error or I/O Channel Check
IRQ 0	Timer (in the Chipset)
IRQ 1	Keyboard
IRQ 2	Cascade from Controller 2
IRQ 3	Serial Port 2 (COM2): General Use Port
IRQ 4	Serial Port 1 (COM2): General Use Port
IRQ 5	Available for users *
IRQ 6	Floppy Disk Controller
IRQ 7	Parallel Port 1 (LPT1) : Printer Port
IRQ 8	Real Time Clock
IRQ 9	Serial Port 3 (COM3): General Use Port
IRQ 10	Serial Port 4 (COM4): Touch Panel (for Standard monitor)
IRQ 11	Available for users *
IRQ 12	PS/2 Mouse
IRQ 13	Numeric Data Processor
IRQ 14	Hard Disk (IDE)
IRQ 15	Available for users *

CAUTION!

The "*" mark indicates that, depending on the customer's settings, the following devices are automatically allocated as Plug and Play devices.

Display Controller

SMBus Controller

Multimedia Device

Network Controller

USB Controller

■ DMA Channel List

	Description
DMA 0	
DMA 1	
DMA 2	Floppy disk controller
DMA 3	
DMA 4	Cascade to controller 1
DMA 5	
DMA 6	
DMA 7	

For 8-bit transmission

For 16-bit transmission

A-2 RAS Feature

1 PL's RAS Features

RAS, which stands for Reliability, Availability and Serviceability, is a device-level monitoring function that provides a variety of features to improve the reliability of your PL system.

Though the standard set of RAS features used will vary depending on the devices used, the following features are used to provide Alarm Monitoring and External Input Signal support.

Alarm Monitoring	Power Voltage Alarm Cooling Fan Alarm Internal Temperature Alarm Watchdog Timer Time Up Software Mirroring Disk Alarm *3 Hardware Mirror Disk Alarm *1, *3 Touch Panel Alarm *4 Backlight Burnout Detection Feature *4 SMART Alarm
External Input Signal	General Purpose Input (DIN 2 bit) Remote Reset Input *2

CAUTION! ***General Purpose Input (DIN) level must be 1.5S or longer to be detected.***

Also, when either the one of the above mentioned alarms occurs, or an external signal input is received, the following types of alarm processing output signals and features are supported.

External Output Signal	General Purpose Output (DOUT 1 bit) Alarm Output (1 point) Lamp Output (1 point)
Types of Processing (all units)	LED Indicator (3-state display – 1 point) Pop-up Message Output Buzzer Output System Shutdown

Furthermore, using the PL's System Monitor feature (included in the PL's software utility disk), allows the easy setting and control (Enable/Disable) of the aforementioned Alarm Monitor and External Input Signals.

Reference *A-3 - System Monitor*

Last, the system monitor feature's use of an Application Link Library (API-DLL) allows it to also be used with other applications.

**1 When a Hardware Mirror Disk Alarm occurs and the standard RAS feature settings are used, the alarm output is limited to the Mirror Disk unit's LED indicator. (Alternately flashing orange ,green, and red)*

**2 The remote reset feature's input can be either enabled or disabled, the alarm output setting cannot be set to trigger a forced system reset.*

**3 The Hardware Mirroring Disk and Software Mirroring Disk must be purchased separately.*

**4 PL-6920 Series unit's only .*

2 RAS Feature Details

■ Alarm Monitoring

◆ Power Voltage Alarm

Monitors the condition of the PL's internal and CPU power.

◆ Cooling Fan Alarm

Monitors the condition of the PL's internal power and CPU cooling fans.

◆ Internal Temperature Alarm

Monitors the PL's internal and CPU vicinity temperatures.

The enabling or disabling of the above three items is performed via the System Setup Area's settings.

For detailed information about the monitoring settings,

Reference *5-2-10 PC Health Status*

This utility can also be used to enable or disable the above mentioned features, as well as designate what type of processing is to be performed.

◆ Watchdog Timer Time Up

This feature monitors the performance of the CPU by writing the CPU's time-up count value from the application to the RAS feature's built-in programmable timer. Errors are detected when the writing of the count value from the application stops, which will cause a timer overflow.

Use the System Monitor Property area to set the Watchdog Timer Timeup ON or OFF, and enter detailed error processing settings.

◆ Software Mirroring Disk Alarm

This optional product monitors the performance of the PL unit's hard disk, via the optional Software Mirroring disk.

◆ Hardware Mirroring Disk Alarm

Whenever a disk crash, or other alarm event occurs to the optional Hardware Mirroring Disk unit, the PL unit's LED indicator will flash (either orange or green) to indicate there is a problem.

◆ Touch Panel Alarm

Detects Touch Panel alarms. When an alarm occurs, the LED will flash orange.

◆ Backlight Burnout Detection Feature (PL-6920 Series unit's only)

Detects a backlight burnout.

When a backlight burnout is detected, the touch panel is disabled and the LED will indicate that an alarm has occurred. The use of this feature is recommended to prevent accidental touch panel operation. The enabling/disabling of this feature is performed via the BLSET.EXE program. (This is an MS-DOS utility)

Reference *6-4 Application Features*

◆ SMART Alarm

Monitors the status of the hard disk. Detects a warning of trouble of the hard disk.

CAUTION!

- ***Administrator Authentication is required for executing SMART Monitoring.***
- ***In order to execute a SMART monitoring, the OS needs to be Windows NT®, Windows® 2000, or Windows® XP.***
- ***A CF card doesn't support SMART and therefore the status of the CF card cannot be monitored.***
- ***When a hard disk except options made by Pro-face is used, operation of SMART Monitoring cannot be guaranteed.***

  The hard disks as shown in the table below are supported.

Hard Disk Model Number	Rev.
PL-HD220	Rev. C or later
PL-HDX920-W2K/ML	Hardware Rev. A or later

- ***The SMART Monitor can be performed for the hard disk connected to the IDE only. The state of the hard disk connected via a USB or a SCSI cannot be monitored.***
- ***When using a Software Mirroring Utility with Rev.C or before, the SMART Monitor cannot perform.***

■ External Input Signal

The PL's RAS interface connector uses the following input signals.

General Purpose Input (DIN)

This standard digital input is used for alarm detection in external devices. The input signal uses two bits.

The System Monitor utility can be used to enable or disable this feature, as well as designate what type of processing is to be performed once a signal is received.

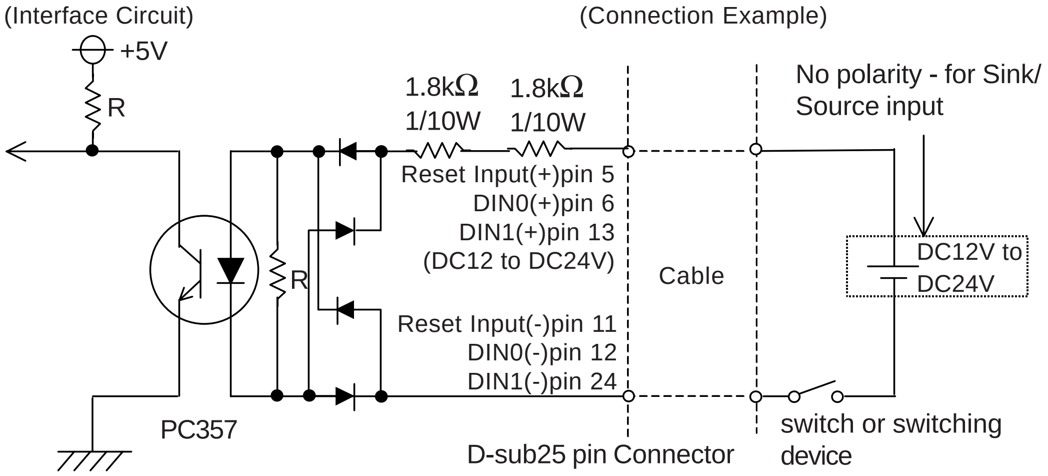
Remote Reset Input

This is the reset signal sent from an external device to the PL. When this signal is enabled, a forced reset of the PL is performed.

The System Monitor utility can be used to enable or disable this feature

External Input Signal (for both DIN and Remote Reset Input)

Input Voltage	DC12V to DC24V
Input Current	7mA
Operating Voltage	ON voltage: 9V (min), OFF voltage:3V (max)
Isolation Method	Via photocoupler



- ***General Purpose Input (DIN) level must be 1.5S or longer to be detected.***
- ***Be sure the voltage value between terminals is controlled via the input voltage, so that the PL is operated within its recommended range. If the input voltage exceeds this range, a malfunction or PL damage may occur.***
- ***With Sink/Source input, even if the D(-), and RESET(-) are positive, and D(+), RESET(+) are negative, no problems are created. Be sure to operate the unit within the recommended voltage range.***

For connection pin location details,

Reference ***2-3-5 RAS Interface***

■ External Output Signal

The PL's RAS interface connector uses the following output signals.

General Purpose Output (DOUT)

This general purpose digital output signal provides system condition information to external devices.

The System Monitor's API-DLL are used by applications to control this signal.

Alarm Output (1 point)

Lamp Output (1 point)

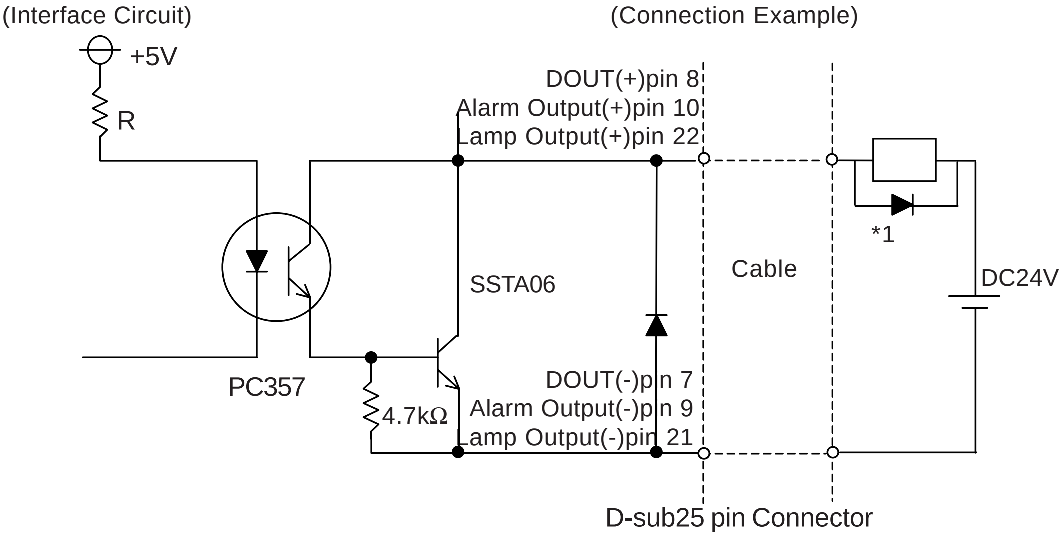
The above mentioned general purpose digital output signals provide system condition information to external devices.

The System Monitor utility can be used to enable or disable any of these output signals.

Also, when alarm output is enabled, the orange colored LED indicator will also blink.

External Output Signal (used for DOUT, Alarm Output, Lamp Output)

Rated Load Voltage	DC12V to DC24V
Maximum Load Current	100mA/point
Maximum Voltage Drop between Terminals	1.5V (at 100mA load current)
Isolation Method	Via photocoupler



CAUTION!

- ***Be sure to operate the unit within its maximum load current. If the maximum load current exceeds this range, a malfunction or PL damage may occur.***
- ***Design your electrical system by adding the load current and voltage values to the terminal voltage. If load current value used is large, a maximum voltage of 1.5V will exist between the terminals.***
- ***When connecting an induction load, be sure to connect the above drawing's protection diode (*1) .***

For connection pin location details,

Reference ***2-3-5 RAS Interface***

■ Types of Processing (all units)

The PL provides system condition information via the following methods.

LED Indicator (2-state display – 1 point)

In addition to indicating if the unit's power is ON or OFF, the 2-state LED indicator (power lamp) provides the following system condition information. PL-B920 LED indicator is on the left side of PL unit's. When connecting to a DU, the DU's LED indiator is on the left bottom of the DU screen.

Color	System Condition	Output Created
Orange	RAS has detected a system alarm.	The alarm output set in the system monitor is enabled.
Green	Normal Operation (Power ON)	None
Flashing Orange/Green	Soft Mirror and Mirror Disk Alarm has occurred	None
Flashing Orange/Red	Backlight Burnout Detection Feature (PL-6920 Series unit's only)	None
Flashing Red/ Green	Mirror Disk and Backlight Alarm has occurred (PL-6920 Series unit's only)	None

Pop-up Message Output

This feature uses the Windows® system's pop-up message feature to indicate that an alarm has occurred.

Buzzer Output

This feature uses the PL's internal speaker to indicate the system's condition.

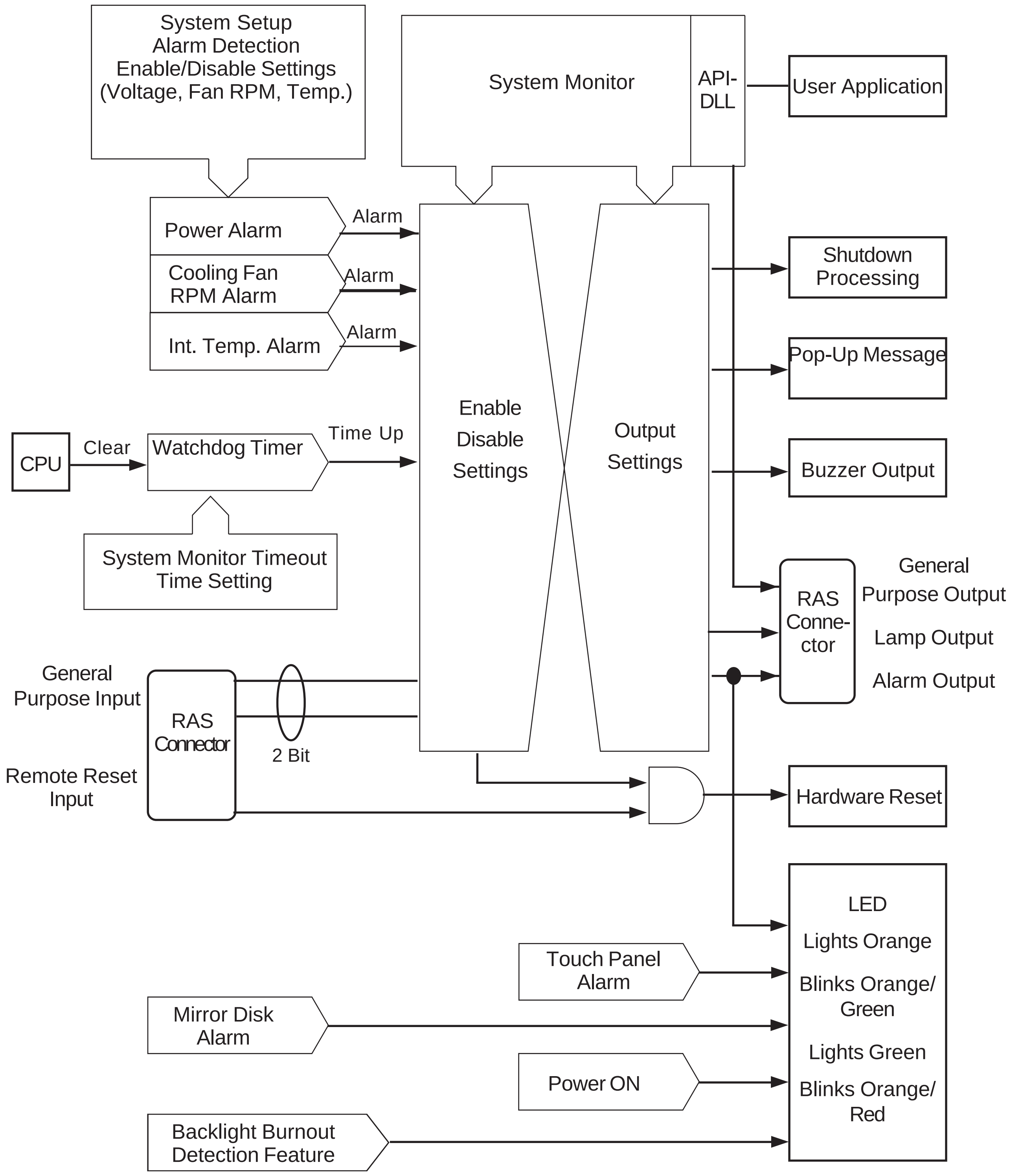
System Shutdown

This feature shuts down the PL's OS. The System Monitor utility can be used to enable or disable this feature.

System Reset

Resets the PL unit's system after the Watchdog Timer reaches "Time Up".

3 RAS Feature Overview

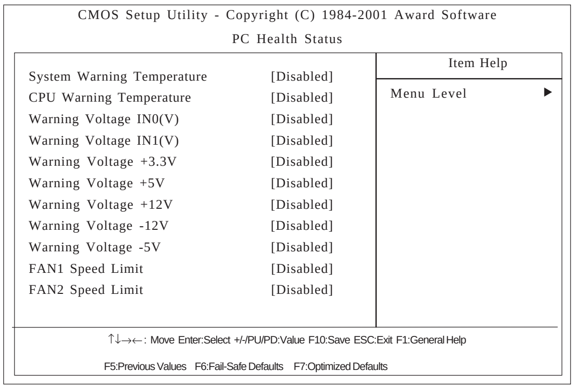


A-3 System Monitor

1 Setup Procedure

Follow the steps shown here to complete the System Monitor/RAS setup.

■ System Setup Screen Settings



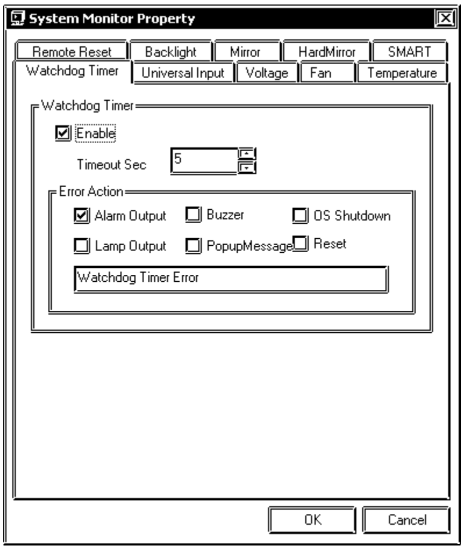
```
CMOS Setup Utility - Copyright (C) 1984-2001 Award Software
PC Health Status

System Warning Temperature     [Disabled]    Item Help
CPU Warning Temperature        [Disabled]
Warning Voltage IN0(V)         [Disabled]    Menu Level      ▶
Warning Voltage IN1(V)         [Disabled]
Warning Voltage +3.3V          [Disabled]
Warning Voltage +5V            [Disabled]
Warning Voltage +12V           [Disabled]
Warning Voltage -12V           [Disabled]
Warning Voltage -5V            [Disabled]
FAN1 Speed Limit               [Disabled]
FAN2 Speed Limit               [Disabled]

↑↓→←: Move Enter:Select +/-/PU/PD:Value F10:Save ESC:Exit F1:General Help
F5:Previous Values   F6:Fail-Safe Defaults   F7:Optimized Defaults
```

In the System Setup area's PC Health Status menu, enable or disable each feature according to your system needs.

■ System Monitor Property Settings



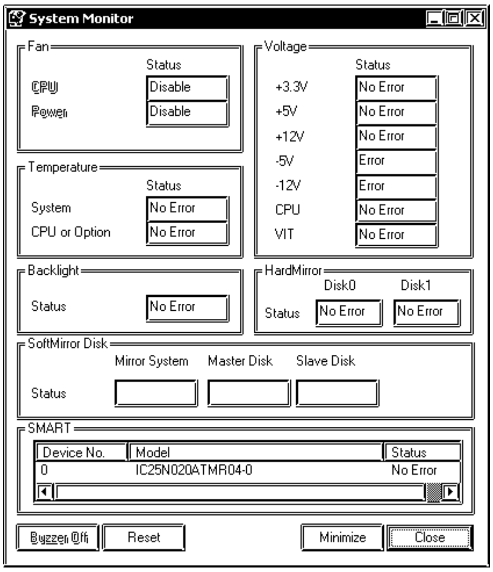
System Monitor Property

Tabs: Remote Reset | Backlight | Mirror | HardMirror | SMART | Watchdog Timer | Universal Input | Voltage | Fan | Temperature

Watchdog Timer
☑ Enable
Timeout Sec: 5

Error Action
☑ Alarm Output ☐ Buzzer ☐ OS Shutdown
☐ Lamp Output ☐ PopupMessage ☐ Reset
Watchdog Timer Error

1) Start the PL unit's OS and click on the [Start] -> [Program] ->[System Monitor]->[System Monitor Property] screen. In this screen enter the System Monitor/RAS Event settings for each feature/tab. Click on the [OK] button and the program will automatically close.



System Monitor

Fan
	Status
CPU	Disable
Power	Disable

Temperature
	Status
System	No Error
CPU or Option	No Error

Voltage
	Status
+3.3V	No Error
+5V	No Error
+12V	No Error
-5V	Error
-12V	Error
CPU	No Error
VIT	No Error

Backlight
Status: No Error

HardMirror
	Disk0	Disk1
Status	No Error	No Error

SoftMirror Disk
	Mirror System	Master Disk	Slave Disk
Status			

SMART
Device No.	Model	Status
0	IC25N020ATMR04-0	No Error

2) Restart the [System Monitor] utility, click on the [System Monitor] button and you can begin to monitor PL activity.

Note:

- **Administrator Authentication is required for executing SMART Monitoring. When a user who does not have the administrator authentication logs in, nothing is displayed in the item of SMART.**
- **In order to execute a SMART monitoring, the OS needs to be Windows NT®, Windows® 2000, or Windows® XP.**

  In case of Windows® 95 or Windows® 98, nothing is displayed in the item of the SMART.

2 System Monitor Property Settings (PL_Wps.exe)

The following chart shows the features available when any of the monitoring value ranges set in the [PC Health Status] menu is exceeded.

O: Can be set X: Cannot be set

	Alarm Output	Lamp Output	Buzzer	Popup Message	OS Shutdown	Reset
Watchdog Timer	O	O	O	O	O	O
Universal Input	O	O	O	O	O	X
Voltage	O	O	O	O	O	X
Fan	O	O	O	O	O	X
Temperature	O	O	O	O	O	X
Remote Reset *1	X	X	X	X	X	O
Backlight *2	O	O	O	O	X	X
Mirror	O	O	O	O	X	X
HardMirror	O	O	O	O	X	X
SMART	O	O	O	O	X	X

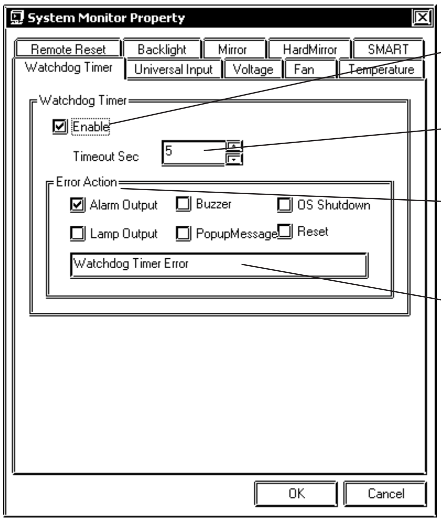
**1 When setting Enable on Remote Reset, the same action as the Reset occurs.*

**2 This feature can be used only with PL-6920 Series units.*

Each of the above items performs the following operation.

Item	Operation
Alarm Output	RAS Interface Alarm Output (#9 to #10) signal is output.
Lamp Output	RAS Interface Alarm Output (#21 to #22) signal is output.
Buzzer	Buzzer sound is output as an alarm notification. (except for when the OS Shutdown feature is checked)
Popup Message	Error message appears as a Pop-Up Message Screen (on the PL unit's screen)
OS Shutdown	Shuts down the PL unit's OS. This can be set to either display a shutdown confirmation message, or perform a forced OS shutdown.Default is display a shutdown confirmation message.
Reset	Resets the PL unit by performing a forced shutdown.
Enable	Allows setting of monitoring items.

The System Monitor Property Screen details is as shown below.



☑ Enable

Timeout Sec 5

☑ Alarm Output ☐ Buzzer ☐ OS Shutdown

☐ Lamp Output ☐ PopupMessage ☐ Reset

Watchdog Timer Error

Set Enable/Disable of the function

Set the timer

Set the Error action when over the Enabled range

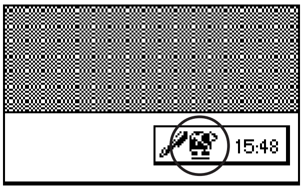
Input the message for the Pop-up Message Screen

3 System Monitor Operation (PL_Smon.exe)

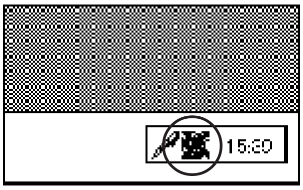
As soon as the PL unit's OS starts up, instead of showing the System Monitor Dialog box, the Alarm Monitoring icon appears in the System Tray. (see below)

Usually, when a System Monitor dialog box appears, the user checks the current condition and then left-clicks the mouse on the system tray icon to call up the System Monitor screen.

When an alarm is detected, the actions set in the System Monitor Property screen are performed, and an "X" appears over the System Tray icon. When this occurs, double-click on the icon to view the alarm contents.

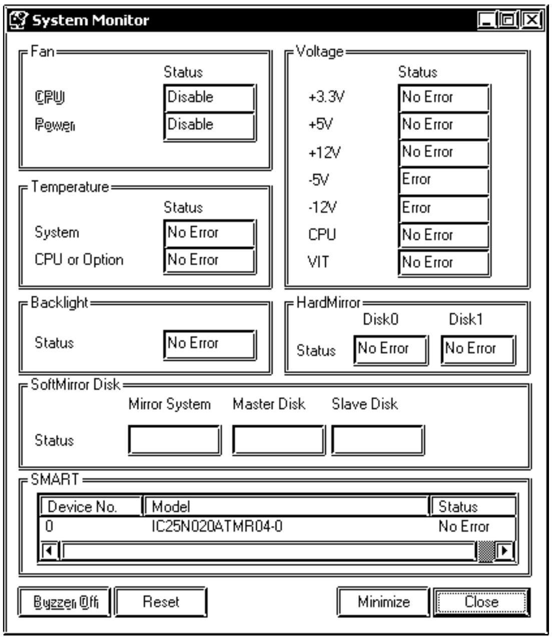


Start-up Icon



Alarm condition Icon

The System Monitor screen is as shown below.



System Monitor

Fan

	Status
CPU	Disable
Power	Disable

Temperature

	Status
System	No Error
CPU or Option	No Error

Voltage

	Status
+3.3V	No Error
+5V	No Error
+12V	No Error
-5V	Error
-12V	Error
CPU	No Error
VIT	No Error

Backlight

Status: No Error

HardMirror

	Disk0	Disk1
Status	No Error	No Error

SoftMirror Disk

	Mirror System	Master Disk	Slave Disk
Status			

SMART

Device No.	Model	Status
0	IC25N020ATMR04-0	No Error

System Monitor Screen

Note: • **As for Device No. of SMART, [0] means Master and [1] means Slave.**

In addition to the "Buzzer Off", "Reset", "Minimize", and "Close" buttons, the System Monitor screen contains the following features.

Button Name	Meaning
Buzzer Off	Stops the Alarm buzzer.
Reset	Resets alarm operation or System Monitor's internal alarm hold condition.
Minimize	Minimizes the System Monitor icon.
Close	Quits the System Monitor software.

Within the System Monitor screen is are the three words "No Error", "Error", and "Disable". They show the current status of each of the monitoring items, such as the Fan, Voltage, Temperature, Backlight, Software Mirroring Disk, Hardware Mirroring Disk and SMART.

Status Name	Meaning
No Error	Normal operation
Error	Alarm
Disable	Not monitored
Not Support	Not Support

Note:

- **[Not Support] is displayed when a device like a CF card, which does not support SMART, is detected.**

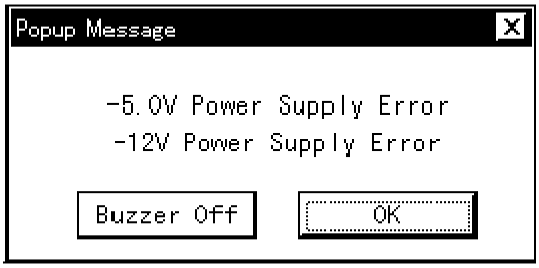
Whenever an alarm occurs for one of the monitored items, or when input (Universal Input) is detected from an external source, The operation designated in the System Monitor Property screen (Error Action) is performed.

Reference *3-2 System Monitor Property Settings*

Each error action, once an error or input is detected, is performed only once.

If +3.3V and +5.0V are monitored and a pop-up message is designated for the error action, when the +3.3V alarm is detected, a pop-up message will appear. Click on [OK] and the box is closed. Then, when a +5.0V alarm occurs, the +5.0V pop-up message will appear.

The pop-up message provides information about the type of error and the error contents. When this message box's [Buzzer Off] button is clicked, the buzzer sound will stop. Clicking on [OK] will close the box.



-5.0V Power Supply Error
-12V Power Supply Error

Pop-up Message Box

Once an alarm has occurred, the System Monitor will continue to remain in the "Alarm" state. (i.e. the alarm detected "X" will continue to be displayed on the System Tray icon. To release this condition, click on the System Monitor dialog box's [Reset] button. Or, turn the PL unit's power OFF, find and solve the problem, and turn the PL unit's power ON again.

4 Error Messages

The following error messages occur via the settings in the System Monitor, and the System Monitor Property dialog boxes.

■ System Monitor

◆ Alarm Pop-up Messages

When the Pop-up messages related to the Error Action are enabled, the following error messages will appear.

Error Type	Message
CPU Voltage	"CPU Power Supply Error"
+3.3V	"+3.3V Power Supply Error"
+5.0V	"+5V Power Supply Error"
+12V	"+12V Power Supply Error"
-12V	"-12V Power Supply Error"
-5V	"-5V Power Supply Error"
CPU Voltage 2	"VIT Power Supply Error"
Power FAN	"Power FAN Error"
CPU FAN	"CPU or OPTION FAN Error"
System Temperature	"System Temperature Error"
CPU or Option Temperature	"CPU Temperature Error"
Universal Input 0	"Universal Input 0"
Universal Input 1	"Universal Input 1"
Watchdog	"Watch Dog Timer Error"
Software Mirroring	"A Mirror disk error occurred"
Hardware Mirroring	"A Mirror disk error occurred"
Backlight	"Back Light Blowout Error"
SMART	"SMART Error"

Driver Error

"The system monitor driver not found."

"Install the latest driver."

Driver Version Error

"The old system monitor driver version."

"Update the driver."

Overlapping Startup message

"System monitor has started. "

"Terminate the system monitor in starting."

Shutdown Confirmation

"The system monitor is terminated."

"Are you sure?"

System Monitor Property Screen

Overlapping startup message

"System Monitor Property has started."

"Terminate the system monitor property in starting."

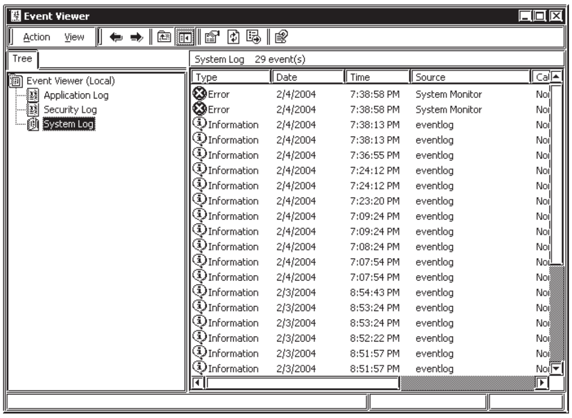
Shutdown Confirmation message

"Save Changes to the registry?"

5 Error Displays When Using Event Viewer

Error type/location and error actions are recorded as error events in the System Log. Error event information can be checked using the Event Viewer.

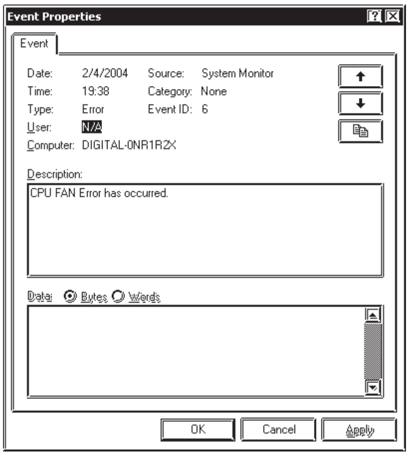
Error Message Display



System Log 29 event(s)

Type	Date	Time	Source	Ca
Error	2/4/2004	7:38:58 PM	System Monitor	No
Error	2/4/2004	7:38:58 PM	System Monitor	No
Information	2/4/2004	7:38:13 PM	eventlog	No
Information	2/4/2004	7:38:13 PM	eventlog	No
Information	2/4/2004	7:36:55 PM	eventlog	No
Information	2/4/2004	7:24:12 PM	eventlog	No
Information	2/4/2004	7:24:12 PM	eventlog	No
Information	2/4/2004	7:23:20 PM	eventlog	No
Information	2/4/2004	7:09:24 PM	eventlog	No
Information	2/4/2004	7:09:24 PM	eventlog	No
Information	2/4/2004	7:08:24 PM	eventlog	No
Information	2/4/2004	7:07:54 PM	eventlog	No
Information	2/4/2004	7:07:54 PM	eventlog	No
Information	2/3/2004	8:54:43 PM	eventlog	No
Information	2/3/2004	8:53:24 PM	eventlog	No
Information	2/3/2004	8:53:24 PM	eventlog	No
Information	2/3/2004	8:52:22 PM	eventlog	No
Information	2/3/2004	8:51:57 PM	eventlog	No
Information	2/3/2004	8:51:57 PM	eventlog	No

1) Start the [Control Panel] -> [Administrative Tools] -> [Event Viewer], and select [System Log].



Date: 2/4/2004 Source: System Monitor
Time: 19:38 Category: None
Type: Error Event ID: 6
User: N/A
Computer: DIGITAL-0NR1R2X

Description:
CPU FAN Error has occurred.

Data: ● Bytes ○ Words

2) Select the System Monitor's error and click the [Properties] icon. When the [Event Properties] dialog box appears, the error message will appear in the [Description] area.

◆ Error Type/Location

The error types/locations shown by the Event Viewer are as follows.

Error Type/Location	Error Message
+3.3V	+3.3V Error has occurred.
+5.0V	+5.0V Error has occurred.
+12V	+12V Error has occurred.
-12V	-12V Error has occurred.
-5.0V	-5.0V Error has occurred.
CPU valtage	CPU voltage Error has occurred.
Vit valtage	Vit valtage Error has occurred.
CPU FAN	CPU FAN Error has occurred.
Power FAN	Power FAN Error has occurred.
CPU Temp	CPU Temperature Error has occurred.
System Temp	System Temperature Error has occurred.
Universal Input 0	Universal Input 0 Error has occurred.
Universal Input 1	Universal Input 1 Error has occurred.
Watch Dog Timer	Watch Dog Timer Error has occurred.
Backlight	Backlight Error has occurred.
Soft Mirror	Mirror Disk Error has occurred.
Hard Mirror	Mirror Disk Error has occurred.
SMART	SMART Error has occurred. Attribute (No.) (Attribute Name) Device (No.) (HD Model). The descriptions in the parentheses vary depending on details of occurring errors and the device having errors occur (0:Master,1:Slave).

◆ Error Action

Using the Event Viewer the following actions can be taken when an error occurs.

Note:

- **The data shown in the table’s “□” indicate the error type/location.**
- **The actions to take after an error occurs are set via the System Monitor Property screen.**

Error Action	Error Message
Buzzer	Buzzer has sounded because of □□ error.
Popup Message	Popup message has been shown because of □□ error.
OS Shutdown	Windows has been shut down because of □□ error.
ALARM	ALARM has output because of □□ error.
LAMP	LAMP has output because of □□ error.

When a “+3.3V” error occurs and the buzzer sounds, two errors will be displayed, “+3.3V Error has occurred.” and “Buzzer has sounded because of +3.3V error”.

A-4 System Monitor/RAS Feature API-DLL

1 Operation Environment

The following information explains the Dynamic Link Libraries used by the System Monitor/RAS feature on a PL-X920 Series unit.

API-DLLs provide the interface for applications to access the System Monitor/RAS feature (System Monitor/RAS Device Driver). Applications can use DLLs to access the following types of features.

1. Driver Version information
2. System Monitor feature status
3. Read out (Get) various monitoring parameters (voltage, fan, temperature)
4. System Monitor current data (voltage, fan, temperature)
5. Watchdog parameters
6. Alarm processing
7. General input processing
8. Reset (of PL unit)
9. Software data mirroring
10. Event handling

■ Compatible Operating Systems

The API-DLLs contained on the PL unit's CD-ROM are compatible with the following OS types.

- Mircrosoft Windows®95
- Microsoft Windows®98
- Microsoft WindowsNT®4.0
- Microsoft Windows®2000
- Microsoft Windows®XP

Each OS must use its corresponding System Monitor/RAS Device Driver.

■ Compatible Languages

- Microsoft Visual C
- Microsoft Visual C++
- Microsoft Visual Basic

◆ Required Files

The following files are required when using DLLs. Each language requires its own set of files.

• Visual C

File Name	Description
PL_Iocif.h	Driver interface definition "include" file
PL_Ioc.LIB	Library definition file
PL_Ioc.dll	Dynamic link library file

• Visual C++

File Name	Description
PL_Iocif.h	Driver interface definition "include" file
PL_Iocall.h	CPL_Iocall class definition "include" file
PL_Ioctl.h	CPL_Ioctl class definition "include" file
PL_Ioc.LIB	Library definition file
PL_Ioc.dll	Dynamic Link library file
Sm.h	Soft Mirror difinition file (Only when using Soft Mirror)
PL_SmiIoctl.h	CPL_SmiIocrlclass definition "include" file (used only with software mirroring feature)

* "#include header files should be "included" in the following order.

#include PL_Iocif.h

#include PL_Ioctl.h

PL_Iocall.h is automatically included, and does not need to be directly designated.

• Visual Basic

File Name	Description
PL_Ioc.bas	Driver interface definition file
PL_Ioc.LIB	Library definition file
PL_Ioc.dll	Dynamic link library file

■ Dynamic Link Library (DLL)

In order for an application to use PL_Ioc.dll, it should be copied to the following folder.

OS	Location
Windows 95/Windows 98	C:\Windows\System
Windows NT4.0/Windows 2000	C:\Winnt\System32
Windows XP	C:\Windows\System32

2 Class Contents

■ CPL_Ioctl Class

This class is used to set the parameters for device driver access using CPL_Ioctl class.

Key Word	Type	Variable Name	Description
public	HANDLE	m_Drvhandle	Device driver handle

■ CPL_Iocall Class

This uses the parameters set in CPL_Ioctl, and calls up DeviceIoControl (Driver Access function).

However, since this class succeeds CPL_Ioctl, it cannot be used directly.

Key Word	Type	Variable Name	Description
public	HANDLE	m_h	Device driver handle
public	LONG	m_long	Control code for action to perform
public	void *	m_ibp	Input data buffer address
public	ULONG	m_ibsize	Input data buffer size
public	void *	m_obp	Output data buffer address
public	ULONG	m_obsize	Output data buffer size
public	DWORD	m_retsize	Address for actual no. of output bytes
public	LPOVERLAPPED	m_ovlp	Address of overlap design

■ CPL_SmiIoctl Class

This class is used to set the parameters for device driver access using CPL_SmiIoctl class.

This class is only used when using the Software Mirroring driver.

Key Word	Type	Variable Name	Description
public	HANDLE	m_Drvhandle	Device driver handle

3 Visual C Functions

Function Name	Description
InitIoctl	Creates the CPL_Ioctl object
EndIoctl	Destroys the CPL_Ioctl object
GetDrvHandle	Gets the driver handle
CloseDrvHandle	Destroys the driver handle
GetDrvVersion	Gets the driver version
GetMonitorSetup	Gets the monitoring enabled/disabled setting
GetVoltParam	Gets the voltage monitoring parameter
GetCurrentVolt	Gets the current voltage value
GetFanParam	Gets the fan monitoring parameter
GetCurrentFan	Gets the current fan value
GetTempParam	Gets the temperature monitoring parameter
GetCurrentTemp	Gets the current temperature value
SetWdtCounter	Sets the watchdog timer counter
GetWdtCounter	Gets the watchdog timer counter
SetWdtMask	Sets warning masking in case of watchdog timer timeout
GetWdtMask	Gets warning masking in case of watchdog timer timeout
StartWdt	Starts the watchdog timer
StopWdt	Stops the watchdog timer
RestartWdt	Restarts the watchdog timer
RunningWdt	Gets the watchdog timer operation status
SetWarningOut	Sets warning output
GetWarningOut	Gets warning output
GetUniversalIn	Gets universal input
ClearUniversalIn	Clears the universal input latched status
SetUniversalInMask	Sets universal input masking
GetUniversalInMask	Gets universal input masking
SetResetMask	Sets reset-masking
GetResetMask	Gets reset-masking
SetIdeErr	Sets data mirroring (software) error
GetIdeErrHard	Gets data mirroring (hardware) error
GetLightblowErr *1	Gets backlight burnout status
GetEvent	Gets the error event
ClearEvent	Clears the error event
StartInsideBuzzer	Starts PL internal buzzer
StopInsideBuzzer	Stops PL internal buzzer
ChkInsideBuzzer	Checks PL internal buzzer status
GetWdtTimeout	Gets the timeout status of the watchdog timer
ClearWdtTimeout	Clears the timeout status of the watchdog timer
SetWarningDOUT	Sets the warning output DOUT
GetWarningDOUT	Gets the warning output DOUT
GetSmiDrvHandle	Gets Software Mirroring driver handle
CloseSmiDrvHandle	Destroys Software Mirroring driver handle
GetSmiAryStatus	Gets status of Software Mirroring Array
GetSmiDevStatus	Gets status of Software Mirroring Device
SetWdtResetMask	Sets the Reset Mask of the watchdog timer
GetWdtResetMask	Gets the Reset Mask of the watchdog timer

*1 This feature can be used only with PL-6920 Series units.

4 Visual C Function Specifications (Details)

InitIoctl

Call Format	void WINAPI InitIoctl(void)
Return Value	None
Arguments	None
Processing	Creates a CPL_Ioctl object. The object is not destroyed until the EndIoctl function is called.
Example	InitIoctl();

EndIoctl

Call Format	void WINAPI EndIoctl(void)
Return Value	None
Arguments	None
Processing	Destroys the object created using the InitIoctl function.
Example	EndIoctl();

GetDrvHandle

Call Format	int WINAPI GetDrvHandle(HANDLE * pHndl)
Return Value	0: Normal 1: Error
Arguments	(I/O) HANDLE *pHndl Pointer to the device driver handle
Processing	Gets the device driver handle to communicate with the device driver.

Example	int ret;
	HANDLE hndl;
	ret = GetDrvHandle(&hndl);

Note: An error occurs if the System Monitor/RAS Device Driver is not running.

CloseDrvHandle

Call Format	BOOL WINAPI CloseDrvHandle(void)
Return Value	TRUE: Normal FALSE: Error
Arguments	None
Processing	Destroys the device driver handle created using the GetDrvHandle function.

Example	BOOL ret;
	//Destroys the handle
	ret = CloseDrvHandle();

GetDrvVersion

Call Format BOOL WINAPI GetDrvVersion(int *pMajor, int *pMinor)

Return Value TRUE: Normal
FALSE: Error

Arguments (I/O) int *pMajor Pointer to version information (Major, 0 to 99).
(I/O) int *pMinor Pointer to version information (Minor, 0 to 99).

Processing Gets the driver's version information.

Example
```
BOOL      ret;
int       Major, Minor;
ret = GetDrvVersion( &Major, &Minor );
```

Note: **If the version is 1.10, then you will get**
Major: 1 (decimal)
Minor: 10 (decimal).

GetMonitorSetup

Call Format BOOL WINAPI GetMonitorSetup(int Selector, int *pSetup)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_VOLT_CPU	CPU core voltage
MONITOR_VOLT_P33	+3.3 V
MONITOR_VOLT_P50	+5.0 V
MONITOR_VOLT_P12	+12 V
MONITOR_VOLT_M12	-12 V
MONITOR_VOLT_M50	-5.0 V
MONITOR_VOLT_VIT	CPU core voltage 2
MONITOR_TEMP_SYSTEM	System temperature
MONITOR_TEMP_CPU	CPU temperature
MONITOR_TEMP_OPT	Option temperature
MONITOR_FAN_CPU	CPU fan
MONITOR_FAN_POWER	Power fan
MONITOR_FAN_OPT	Option fan

(I/O) int *pSetup Pointer to Data
0: Disabled
1: Enabled

Processing Gets the current monitoring status (enabled/disabled).

Example
```
BOOL      ret;
int       Setup;
// Gets the CPU core voltage setup status.
ret = GetMonitorSetup( MONITOR_VOLT_CPU, &Setup );
```

GetVoltParam

Call Format BOOL WINAPI GetVoltParam
(int Selector, int *pULimit, int *pLLimit)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_VOLT_CPU	CPU core voltage
MONITOR_VOLT_P33	+3.3 V
MONITOR_VOLT_P50	+5.0 V
MONITOR_VOLT_P12	+12 V
MONITOR_VOLT_M12	-12 V
MONITOR_VOLT_M50	-5.0 V
MONITOR_VOLT_VIT	CPU core voltage 2

(I/O) int *pULimit Pointer to upper-limit voltage value (Unit: mV)

(I/O) int *pLLimit Pointer to lower-limit voltage value (Unit: mV)

Processing Gets the voltage monitoring parameter.

Example

```
BOOL     ret;
int      ULimit, LLimit;
// Get the upper and lower-limit values of the CPU core
// voltage.
ret = GetVoltParam( MONITOR_VOLT_CPU, &ULimit, &LLimit );
```

Note: Since the data taken from this function is shown in mV units, the following conversion is needed for use in (Volt) units:

Data in Volt unit = Data in mV unit/1000

GetCurrentVolt

Call Format BOOL WINAPI GetCurrentVolt(int Selector, int *pData)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_VOLT_CPU	CPU core voltage
MONITOR_VOLT_P33	+3.3 V
MONITOR_VOLT_P50	+5.0 V
MONITOR_VOLT_P12	+12 V
MONITOR_VOLT_M12	-12 V
MONITOR_VOLT_M50	-5.0 V
MONITOR_VOLT_VIT	CPU core voltage 2

(I/O) int *pData Pointer to the voltage value (Unit: mV)

Processing Gets the current voltage value.

Example

```
BOOL    ret;
int     Data;
// Gets the CPU core voltage value.
ret = GetCurrentVolt( MONITOR_VOLT_CPU, &Data );
```

Note: Since the data taken from this function is in mV units, the following conversion is needed for use in (Volt) units:

Data in Volt unit = Data in mV unit/1000

GetFanParam

Call Format BOOL WINAPI GetFanParam (int Selector, int *pLLimit)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_FAN_CPU	CPU fan
MONITOR_FAN_POWER	Power fan
MONITOR_FAN_OPT	Option fan

(I/O) int *pLLimit Pointer to the lower-limit fan rotation speed (Unit: RPM)
(RPM: Revolutions Per Minute)

Processing Gets the fan monitoring parameter.

Example

```
BOOL    ret;
int     LLimit;
// Gets the lower-limit CPU fan rotation speed.
ret = GetFanParam( MONITOR_FAN_CPU, &LLimit );
```

GetCurrentFan

Call Format BOOL WINAPI GetCurrentFan(int Selector, int *pData)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_FAN_CPU	CPU fan
MONITOR_FAN_POWER	Power fan
MONITOR_FAN_OPT	Option fan

(I/O) int *pData Pointer to the fan rotation speed
(Unit: RPM)
(RPM: Revolutions Per Minute)

Processing Gets the current fan rotational speed.

Example

```
BOOL      ret;
int       Data;
// Gets the CPU fan rotational speed.
ret = GetCurrentFan( MONITOR_FAN_CPU, &Data );
```

GetTempParam

Call Format BOOL WINAPI GetTempParam(int Selector, int *pULimit)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_TEMP_SYSTEM	System temperature
MONITOR_TEMP_CPU	CPU temperature
MONITOR_TEMP_OPT	Option temperature

(I/O) int *pULimit Pointer to the upper-limit temperature
(Unit: Degrees Celsius)

Processing Gets the temperature monitoring parameter.

Example

```
BOOL      ret;
int       ULimit;
// Gets the system temperature upper-limit value.
ret = GetTempParam( MONITOR_TEMP_SYSTEM, &ULimit );
```

GetCurrentTemp

Call Format BOOL WINAPI GetCurrentTemp(int Selector, int *pData)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_TEMP_SYSTEM	System temperature
MONITOR_TEMP_CPU	CPU temperature
MONITOR_TEMP_OPT	Option temperature

(I/O) int *pData Pointer to the temperature
(Unit: Degrees Celsius)

Processing Gets the current temperature value.

Example
```
BOOL    ret;
int     Data;
// Gets the system temperature value.
ret = GetCurrentTemp( MONITOR_TEMP_SYSTEM, &Data );
```

SetWdtCounter

Call Format BOOL WINAPI SetWdtCounter(int Counter)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Counter Sets to the watchdog timer's initial counter value (5 to 255) (Unit: Seconds)

Processing Sets the current watchdog timer's initial counter value.

Example
```
BOOL   ret;
// Sets the watchdog timer's initial counter value to 10 sec.
ret = SetWdtCounter( 10 );
```

GetWdtCounter

Call Format BOOL WINAPI GetWdtCounter(int *pCounter)

Return Value TRUE: Normal
FALSE: Error

Arguments (I/O) int *pCounter Pointer to the watchdog timer's initial counter value (Unit: Seconds)

Processing Gets the current watchdog timer's initial counter value.

Example
```
BOOL    ret;
int     Counter;
 ret = GetWdtCounter( &Counter );
```

SetWdtMask

Call Format: BOOL WINAPI SetWdtMask(int Selector, int Mask)

Return Value: TRUE: Normal
FALSE: Error

Arguments: (I) int Selector Setting Item

WARNING_LAMP	LAMP
WARNING_ALARM	ALARM

(I) int Mask Masking Information

MASK_OFF	Masking disabled
MASK_ON	Masking enabled

Processing: Sets masking for the warning output used when watchdog timer time-out occurs.

Example:

```
BOOL      ret;
// Enables masking for the lamp output.
ret = SetWdtMask( WARNING_LAMP, MASK_ON );
// Disables masking for the alarm output.
ret = SetWdtMask( WARNING_ALARM, MASK_OFF );
```

GetWdtMask

Call Format: BOOL WINAPI GetWdtMask(int Selector, int *pMask)

Return Value: TRUE: Normal
FALSE: Error

Arguments: (I) int Selector Setting Item

WARNING_LAMP	LAMP
WARNING_ALARM	ALARM

(I/O) int *pMask Pointer to Masking Information

MASK_OFF	Masking disabled
MASK_ON	Masking enabled

Processing: Gets the masking information used for warning output when watchdog timer time-out occurs.

Example:

```
BOOL        ret;
int         Mask;
// Gets the masking information for the LAMP.
ret = GetWdtMask( WARNING_LAMP, &Mask );
// Gets the masking information for the alarm.
ret = GetWdtMask( WARNING_ALARM, &Mask );
```

StartWdt

Call Format	BOOL WINAPI StartWdt(void)
Return Value	TRUE: Normal FALSE: Error
Arguments	None
Processing	Starts watchdog timer countdown.

Example	BOOL ret;
	ret = StartWdt();

StopWdt

Call Format	BOOL WINAPI StopWdt(void)
Return Value	TRUE: Normal FALSE: Error
Arguments	None
Processing	Stops watchdog timer countdown.

Example	BOOL ret;
	ret = StopWdt();

RestartWdt

Call Format	BOOL WINAPI RestartWdt(void)
Return Value	TRUE: Normal FALSE: Error
Arguments	None
Processing	Restarts watchdog timer countdown after resetting to the initial value.

Example	BOOL ret;
	ret = RestartWdt();

Note: **Processing does not occur if the watchdog timer is stopped.**

RunningWdt

Call Format	BOOL WINAPI RunningWdt(int *pRunFlag)		
Return Value	TRUE: Normal FALSE: Error		
Arguments	(I/O) int *pRunFlag	Pointer to Watchdog Timer Operation Status	
		WATCHDOG_STOP	Stopped
		WATCHDOG_COUNTDOWN	Countdown in progress
Processing	Gets the watchdog timer's operation status.		

Example	BOOL ret;
	int RunFlag;
	ret = RunningWdt(&RunFlag);		

SetWarningOut

Call Format BOOL WINAPI SetWarningOut(int Selector, int WarnOut)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Setting Item

WARNING_LAMP	LAMP
WARNING_ALARM	ALARM

(I) int WarnOut Output Status

OUTPUT_OFF	Output OFF
OUTPUT_ON	Output ON

Processing Sets setting item warning information (LAMP or ALARM).

Example

```
BOOL    ret;
// Sets the LAMP output status to ON.
ret = SetWarningOut( WARNING_LAMP, OUTPUT_ON );
// Sets the ALARM output status to OFF.
ret = SetWarningOut( WARNING_ALARM, OUTPUT_OFF );
```

GetWarningOut

Call Format BOOL WINAPI GetWarningOut(int Selector, int *pWarnOut)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Setting Item

WARNING_LAMP	LAMP
WARNING_ALARM	ALARM

(I/O) int *pWarnOut Pointer to Output Status

OUTPUT_OFF	Output OFF
OUTPUT_ON	Output ON

Processing Gets currently set item's warning status (LAMP or ALARM).

Example

```
BOOL      ret;
 int      WarnOut;
// Gets the LAMP output status.
ret = GetWarningOut( WARNING_LAMP, &WarnOut );
// Gets the ALARM output status.
ret = GetWarningOut( WARNING_ALARM, &WarnOut );
```

GetUniversalIn

Call Format BOOL WINAPI GetUniversalIn(int Selector, int *pUniIn)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Designated Port
PORT_UNI0 Universal Input 0
PORT_UNI1 Universal Input 1
(I/O) int *pUniIn Pointer to Input Status
INPUT_OFF Input OFF
INPUT_ON Input ON

Processing Gets the input status of the designated port (Universal Input 0, Universal Input 1).

Example
```
BOOL     ret;
int      UniIn;
// Get the input status of Universal Input 0.
ret = GetUniversalIn( PORT_UNI0, &UniIn );
// Get the input status of Universal Input 1.
ret = GetUniversalIn( PORT_UNI1, &UniIn );
```

ClearUniversalIn

Call Format BOOL WINAPI ClearUniversalIn(int Selector)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Designated Port
PORT_UNI0 Universal Input 0
PORT_UNI1 Universal Input 1

Processing Cancels the input status of the designated port (Universal Input 0, Universal Input 1).

Example
```
BOOL ret;
// Cancels the input status of Universal Input 0.
ret = ClearUniversalIn( PORT_UNI0 );
// Cancels the input status of Universal Input 1.
ret = ClearUniversalIn( PORT_UNI1 );
```

SetUniversalInMask

Call Format	BOOL WINAPI SetUniversalInMask(int Selector, int Mask)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I) int Selector Designated Port PORT_UNI0 Universal Input 0 PORT_UNI1 Universal Input 1 (I) int Mask Masking Information MASK_OFF Masking disabled MASK_ON Masking enabled
Processing	Sets the masking information for the designated port (Universal Input 0, Universal Input 1).

Example

```
BOOL      ret;
// Disable masking for Universal Input 0.
ret = SetUniversalInMask( PORT_UNI0, MASK_OFF );
// Enable masking for Universal Input 1.
ret = SetUniversalInMask( PORT_UNI1, MASK_ON );
```

GetUniversalInMask

Call Format	BOOL WINAPI GetUniversalInMask(int Selector, int *pMask)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I) int Selector Designated Port PORT_UNI0 Universal Input 0 PORT_UNI1 Universal Input 1 (I/O) int *pMask Pointer to Masking Information MASK_OFF Masking disabled MASK_ON Masking enabled
Processing	Gets the masking information for the designatd port (Universal Input 0, Universal Input 1).

Example

```
BOOL        ret;
int         Mask;
// Gets the masking information for Universal input 0.
ret = GetUniversalInMask( PORT_UNI0, &Mask );
// Gets the masking information for Universal input 1.
ret = GetUniversalInMask( PORT_UNI1, &Mask );
```

SetResetMask

Call Format BOOL WINAPI SetResetMask(int Mask)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Mask Masking Information

MASK_OFF Masking disabled

MASK_ON Masking enabled

Processing Sets reset-masking.

Example BOOL ret;

// Disable reset-masking.

ret = SetResetMask(MASK_OFF);

GetResetMask

Call Format BOOL WINAPI GetResetMask(int *pMask)

Return Value TRUE: Normal

FALSE: Error

Arguments (I/O) int *pMask Pointer to Masking Information

MASK_OFF Masking disabled

MASK_ON Masking enabled

Processing Gets the current reset-masking information.

Example BOOL ret;

int Mask;

ret = GetResetMask(&Mask);

SetIdeErr

Call Format BOOL WINAPI SetIdeErr(int IdeErr)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int IdeErr Error Output Status

IDE_ERROR_OFF Error Output OFF

IDE_ERROR_ON Error Output ON

Processing Sets the software control used to create IDE eror output.

Example BOOL ret;

// Sets IDE error output to OFF.

ret = SetIdeErr(IDE_ERROR_OFF);

GetIdeErrHard

Call Format BOOL WINAPI GetIdeErrHard(int Selector, int *pIdeErr)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

```
IDE_ERROR_1    IDE_ERR1
IDE_ERROR_2    IDE_ERR2
```

(I/O) int *pIdeErr Pointer to error signal

```
IDE_ERROR_OFF  Normal
IDE_ERROR_ON   Error
```

Processing Gets the current IDE error signal output by the hardware.

Example

```
BOOL     ret;
int      IdeErr;
// Gets the IDE_ERR1 signal
ret = GetIdeErrHard( IDE_ERROR_1, &IdeErr );
```

GetLightblowErr

Call Format BOOL GetLightblowErr(int *pLightErr)
TRUE: Normal
FALSE: Error

Arguments (I/O) int *pLightErr Error Information

```
BACKLIGHT_OK      OK
BACKLIGHT_ERR     NG
```

Processing Gets Backlight's current burnout error output.

Example

```
BOOL          ret;
int           LightErr;
// Gets backlight's burnout condition.
ret = GetLightblowErr( &LightErr );
```

Note: This feature can be used only with PL-6920 Series units.

GetEvent

Call Format BOOL WINAPI GetEvent(int Selector, int *pEvent)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Selector Parameters

EVENT_VOLT_CPU	CPU core voltage
EVENT_VOLT_P33	+3.3 V
EVENT_VOLT_P50	+5.0 V
EVENT_VOLT_P12	+12 V
EVENT_VOLT_M12	-12 V
EVENT_VOLT_M50	-5.0 V
EVENT_VOLT_VIT	CPU core voltage 2
EVENT_FAN_CPU	CPU FAN
EVENT_FAN_POWER	POWER FAN
EVENT_FAN_OPT	OPTION FAN
EVENT_TEMP_SYSTEM	SYSTEM temperature
EVENT_TEMP_CPU_OPT	CPU or option temperature
EVENT_UNI_IN0	Universal Input 0
EVENT_UNI_IN1	Universal Input 1
EVENT_WDT_TIMEOUT	Watchdog Timeout

(I/O) int *pEvent Pointer to Error Event Information

ERROR_EVENT_OFF	Without error event
ERROR_EVENT_ON	With error event

Processing Checks the machine for voltage, fan, and temperature errors, and the Universal Input information (event) and Watchdog Timeout information.

Example

```
BOOL      ret;
int       Event;
// Gets the error event information for the CPU core voltage.
ret = GetEvent( EVENT_VOLT_CPU, &Event );
```

ClearEvent

Call Format BOOL WINAPI ClearEvent(int Selector)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Selector Parameters used for cancelling error events

EVENT_VOLT_CPU	CPU core voltage
EVENT_VOLT_P33	+3.3 V
EVENT_VOLT_P50	+5.0 V
EVENT_VOLT_P12	+12 V
EVENT_VOLT_M12	-12 V
EVENT_VOLT_M50	-5.0 V
EVENT_VOLT_VIT	CPU core voltage 2
EVENT_FAN_CPU	CPU FAN
EVENT_FAN_POWER	POWER FAN
EVENT_FAN_OPT	OPTION FAN
EVENT_TEMP_SYSTEM	SYSTEM temperature
EVENT_TEMP_CPU_OPT	CPU or option temperature
EVENT_UNI_IN0	Universal input 0
EVENT_UNI_IN1	Universal input 1
EVENT_WDT_TIMEOUT	Watchdog Timeout

Processing Cancels the error event.

Example

```
BOOL     ret;
// Cancels the CPU core voltage error event.
ret = ClearEvent( EVENT_VOLT_CPU );
```

StartInsideBuzzer

Call Format BOOL WINAPI StartInsideBuzzer (int hz, int ms)

Return Value BOOL TRUE: Normal

FALSE: Error

Arguments (I) int hz Buzzer frequency (Hz)

(I) int ms Buzzer length (ms)

Processing Starts the PL unit's internal buzzer, based on the designated frequency and length.

Example

```
BOOL     ret;
int      hz = 600;
int      ms = 1000;
// PL internal buzzer will sound at 600MHz for 1 second.
ret = StartInsideBuzzer ( hz, ms );
```

Note: **This feature cannot be used with a PL running WindowsNT 4.0, Windows 2000 or Windows XP, due to the use of Windows 95 and Windows 98 functions.**

StopInsideBuzzer

Call Format	BOOL WINAPI StopInsideBuzzer (void)		
Return Value	BOOL	TRUE: Normal	
		FALSE: Error	
Arguments	None.		
Processing	Stops the PL unit's internal buzzer.		
Example	BOOL	ret;	
	// Stops PL internal buzzer.		
	ret = StopInsideBuzzer ( );		

Note: **This feature cannot be used with a PL running Windows NT, Windows 2000 or Windows XP, due to the use of Windows 95 and Windows 98 functions.**

ChkInsideBuzzer

Call Format	BOOL WINAPI ChkInsideBuzzer (int *BuzzerParam)		
Return Value	BOOL	TRUE: Normal	
		FALSE: Error	
Arguments	(I/O) int *BuzzerParam	Pointer to Buzzer Condition	
		BUZZER_ON	Buzzer is ON
		BUZZER_OFF	Buzzer is OFF
Processing	Checks the buzzer's operation status.		
Example	BOOL	ret;	
	int	BuzzerParam;	
	// Checks buzzer status.		
	ret = ChkInsideBuzzer (&BuzzerParam);		

Note: **This feature cannot be used with a PL running WindowsNT 4.0, Windows 2000 or Windows XP, due to the use of Windows 95 and Windows 98 functions.**

GetWdtTimeout

Call Format	BOOL WINAPI GetWdtTimeout(int *pTimebuf)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I/O) int *pTimebuf	Pointer to Watchdog Timeout Status	
		TIMEOUT_OK	Not timeout
		TIMEOUT_ERROR	Timeout
Processing	Gets watchdog timeout status.		

Example

```
BOOL     ret;
int      Timebuf;
// Gets watchdog timeout status.
ret = GetWdtTimeout( &Timebuf );
```

ClearWdtTimeout

Call Format	BOOL WINAPI ClearWdtTimeout(void)
Return Value	TRUE: Normal
	FALSE: Error
Arguments	None
Processing	Clears the watchdog timeout status.

Example

```
BOOL     ret;
// Clears the watchdog timeout status.
ret = GetWdtTimeout();
```

SetWarningDOUT

Call Format	BOOL WINAPI SetWarningDOUT(int WarningOut)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I) int WarningOut	Output status	
		OUTPUT_OFF	Output OFF
		OUTPUT_ON	Output ON
Processing	Sets DOUT warning status of current setting item.		

Example

```
BOOL     ret;
// Sets warning DOUT output status to OFF.
ret = SetWarningDOUT( OUTPUT_OFF );
```

GetWarningDOUT

Call Format	BOOL WINAPI GetWarningDOUT(int *pWarningOut)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I/O) int *pWarningOut	Pointer to Output Status	
		OUTPUT_OFF	Output OFF
		OUTPUT_ON	Output ON
Processing	Gets DOUT warning status of current setting item.		
Example	BOOL ret;		
	int WarningOut;		
	// Gets DOUT Output status.		
	ret = GetWarningDOUT(&WarningOut);		

GetSmiDrvHandle

Call Format	int WINAPI GetSmiDrvHandle(void)
Return Value	0: Normal
	1: Error
Arguments	None
Processing	Gets Software Mirroring Device Driver Handle.
Example	int ret;
	ret = GetSmiDrvHandle();

Note: **When the Software Mirroring Device Driver is not operating, an error occurs.**

CloseSmiDrvHandle

Call Format	BOOL WINAPI CloseSmiDrvHandle(void)
Return Value	TRUE: Normal
	FALSE: Error
Arguments	None
Processing	Destroys the device driver handle created using the GetSmiDrvHandle function.
Example	BOOL ret;
	// Destroys the device driver handle created using the GetSmiDrvHandle function.
	ret = ClosetSmiDrvHandle();

GetSmiAryStatus

Call Format	BOOL WINAPI GetSmiAryStatus(int *pStatus)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I/O) int *pStatus	Pointor to Software Mirroring Disk Status	
		ARYSTAT_GOOD	Good
		ARYSTAT_NOTEXIST	No output
		ARYSTAT_UNCONFIG	Unconfigured
		ARYSTAT_REBUILD	Rebuilding
		ARYSTAT_REDUCE	Reduced
		ARYSTAT_DEAD	Dead
Processing	Gets SoftMirror Status		
Example	BOOL ret;		
	int Status;		
	// Gets Software Mirroring Status.		
	ret = GetSmiAryStatus(&Status);		

GetSmiDevStatus

Call Format	BOOL WINAPI GetSmiDevStatus(int Id ,int *pType ,int *pStatus)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I) int Id	Device ID	
		0 : Master HDD	
		1 : Slave HDD	
	(I/O) int* pType	Device Type	
		ATADEVICE	ATA DEVICE
		ATAPIDEVICE	CD-ROM
		UNKNOWNDEVICE	Unknown DEVICE
		NODEVICE	No DEVICE
	(I/O) int* pStatus	Device Status	
		DEVSTAT_GOOD	Good
		DEVSTAT_NOTEXIST	No DEVICE
		DEVSTAT_BROKEN	BROKEN
Processing	Gets Software Mirroring Device Status		
Example	BOOL ret;		
	int Id, Type, Status;		
	// Gets the device status		
	Id = 0;		
	ret = GetSmiDevStatus(Id ,&Type ,&Status);		

SetWdtResetMask

Call Format	BOOL WINAPI SetWdtResetMask(int Mask)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I/O) int Mask	Masking Information	
		MASK_OFF	Masking disabled
		MASK_ON	Masking enabled
Processing	Sets the H/W reset mask used when WDT timeout occurs.		
Example	BOOL ret;		
	// Destroys the mask used for reset at WDT timeout.		
	ret = SetWdtResetMask(MASK_OFF);		

GetWdtResetMask

Call Format	BOOL WINAPI GetWdtResetMask(int *pMask)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I/O) int *pMask	Pointer to Masking Information	
		MASK_OFF	Masking disabled
		MASK_ON	Masking enabled
Processing	Gets the H/W reset mask data used at WDT timeout		
Example	BOOL ret;		
	int Mask;		
	// Gets the reset mask data used at WDT timeout		
	ret = GetWdtResetMask(&Mask);		

5 Visual C++ Functions

Function Name	Description
GetDrvHandle	Gets the driver handle
CloseDrvHandle	Destroys the driver handle
GetDrvVersion	Gets the driver version
GetMonitorSetup	Gets the monitoring enabled/disabled setting
GetVoltParam	Gets the voltage monitoring parameter
GetCurrentVolt	Gets the current voltage value
GetFanParam	Gets the fan monitoring parameter
GetCurrentFan	Gets the current fan value
GetTempParam	Gets the temperature monitoring parameter
GetCurrentTemp	Gets the current temperature value
SetWdtCounter	Sets the watchdog timer counter
GetWdtCounter	Gets the watchdog timer counter
SetWdtMask	Sets warning masking in case of watchdog timer time-out
GetWdtMask	Gets warning masking in case of watchdog timer time-out
StartWdt	Starts the watchdog timer
StopWdt	Stops the watchdog timer
RestartWdt	Restarts the watchdog timer
RunningWdt	Gets the watchdog timer operation status
SetWarningOut	Sets warning output
GetWarningOut	Gets warning output
GetUniversalIn	Gets universal input
ClearUniversalIn	Clears the universal input latched status
SetUniversalInMask	Sets universal input masking
GetUniversalInMask	Gets universal input masking
SetResetMask	Sets reset-masking
GetResetMask	Gets reset-masking
SetIdeErr	Sets software mirroring error
GetIdeErrHard	Gets hardware mirroring error
GetLightblowErr *1	Get BackLight Eror status
GetEvent	Gets the error event
ClearEvent	Clears the error event
StartInsideBuzzer	Starts PL unit's internal buzzer
StopInsideBuzzer	Stops PL unit's internal buzzer
ChkInsideBuzzer	Checks PL unit's internal buzzer
GetWdtTimeout	Gets watchdog timeout status
ClearWdtTimeout	Clears the watchdog timeout status
SetWarningDOUT	Sets warning DOUT
GetWarningDOUT	Gets warning DOUT
GetSmiDrvHandle	Gets Software Mirroring driver handle
CloseSmiDrvHandle	Gets Software Mirroring Status
GetSmiAryStatus	Gets Software Mirroring feature Status
GetSmiDevStatus	Gets Software Mirroring Device Status
SetWdtResetMask	Sets Watchdog Timeout reset masking
GetWdtResetMask	Gets Watchdog Timeout reset masking

*1 This feature can be used only with PL-6920 Series units.

6 Visual C++ Function Specifications (Details)

GetDrvHandle

Call Format	int GetDrvHandle(void) or int GetDrvHandle(HANDLE *pHndl)
Return Value	0: Normal 1: Error
Arguments	None
Processing	Gets the device driver handle to communicate with the device driver. The handle Getsed is stored into the member variable m_handle.

Example 1	CPL_Ioctl m_Ioc;
	m_Ioc.GetDrvHandle();

Example 2	int ret;
	HANDLE hndl;
	ret = ::GetDrvHandle(&hndl);

Note: **An error occurs if the System Monitor/RAS Device Driver is not running.**

CloseDrvHandle

Call Format	BOOL CloseDrvHandle(void)
Return Value	TRUE: Normal FALSE: Error
Arguments	None
Processing	Destroys the device driver handle created using the GetDrvHandle function.

Example 1	CPL_Ioctl m_Ioc;
	BOOL ret;
	// Destroys the device driver handle.
	ret = m_Ioc.CloseDrvHandle();

Example 2	BOOL ret;
	// Destroys the device driver handle.
	ret = ::CloseDrvHandle();

GetDrvVersion

Call Format BOOL GetDrvVersion(int *pMajor, int *pMinor)

Return Value TRUE: Normal
FALSE: Error

Arguments

(I/O) int *pMajor	Pointer to version information (Major, 0 to 99).
(I/O) int *pMinor	Pointer to version information (Minor, 0 to 99).

Processing Gets the driver's version information.

Example 1

```
CPL_Ioctl  m_Ioc;
BOOL       ret;
int        Major, Minor;
ret = m_Ioc.GetDrvVersion( &Major, &Minor );
```

Example 2

```
BOOL       ret;
int        Major, Minor;
ret = ::GetDrvVersion( &Major, &Minor );
```

Note:

If the version is 1.10, then you will get
Major: 1 (decimal)
Minor: 10 (decimal).

GetMonitorSetup

Call Format BOOL GetMonitorSetup(int Selector, int *pSetup)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_VOLT_CPU	CPU core voltage
MONITOR_VOLT_P33	+3.3 V
MONITOR_VOLT_P50	+5.0 V
MONITOR_VOLT_P12	+12 V
MONITOR_VOLT_M12	-12 V
MONITOR_VOLT_M50	-5.0 V
MONITOR_TEMP_SYSTEM	System temperature
MONITOR_TEMP_CPU	CPU temperature
MONITOR_TEMP_OPT	Option temperature
MONITOR_FAN_CPU	CPU fan
MONITOR_FAN_POWER	Power fan
MONITOR_FAN_OPT	Option fan
MONITOR_VOLT_VIT	CPU core voltage 2

(I/O) int *pSetup Pointer to Getsed Data
0: Disabled
1: Enabled

Processing Gets the current monitoring enabled/disabled status.

Example 1

```
CPL_Ioctl        m_Ioc;
BOOL             ret;
int              Setup;
// Gets the CPU core voltage setup status.
ret = m_Ioc.GetMonitorSetup( MONITOR_VOLT_CPU, &Setup );
```

Example 2

```
BOOL             ret;
int              Setup;
// Get the CPU core voltage setup status.
ret = ::GetMonitorSetup( MONITOR_VOLT_CPU, &Setup );
```

GetVoltParam

Call Format BOOL GetVoltParam (int Selector, int *pULimit, int *pLLimit)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_VOLT_CPU	CPU core voltage
MONITOR_VOLT_P33	+3.3 V
MONITOR_VOLT_P50	+5.0 V
MONITOR_VOLT_P12	+12 V
MONITOR_VOLT_M12	-12 V
MONITOR_VOLT_M50	-5.0 V
MONITOR_VOLT_VIT	CPU core voltage 2

(I/O) int *pULimit Pointer to upper-limit voltage value (Unit: mV)
(I/O) int *pLLimit Pointer to lower-limit voltage value (Unit: mV)

Processing Gets the voltage monitoring parameter.

Example 1

```
CPL_Ioctl    m_Ioc;
BOOL         ret;
int          ULimit, LLimit;
// Get the upper and lower-limit values of the CPU core voltage.
ret = m_Ioc.GetVoltParam( MONITOR_VOLT_CPU,
 &ULimit, &LLimit );
```

Example 2

```
BOOL         ret;
int          ULimit, LLimit;
// Get the upper and lower-limit values of the CPU core voltage.
ret = ::GetVoltParam( MONITOR_VOLT_CPU, &ULimit, &LLimit );
```

Note:

Since the data taken from this function is shown in mV units, the following conversion is needed for use in (Volt) units:

Data in Volt unit = Data in mV unit/1000

GetCurrentVolt

Call Format BOOL GetCurrentVolt(int Selector, int *pData)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_VOLT_CPU	CPU core voltage
MONITOR_VOLT_P33	+3.3 V
MONITOR_VOLT_P50	+5.0 V
MONITOR_VOLT_P12	+12 V
MONITOR_VOLT_M12	-12 V
MONITOR_VOLT_M50	-5.0 V
MONITOR_VOLT_VIT	CPU core voltage 2

(I/O) int *pData Pointer to the voltage value (Unit: mV)

Processing Gets the current voltage value.

Example 1

```
CPL_Ioctl        m_Ioc;
BOOL             ret;
int              Data;
// Get the CPU core voltage value.
ret = m_Ioc.GetCurrentVolt( MONITOR_VOLT_CPU, &Data );
```

Example 2

```
BOOL             ret;
int              Data;
// Get the CPU core voltage value.
ret = ::GetCurrentVolt( MONITOR_VOLT_CPU, &Data );
```

Note:

Since the data taken from this function is shown in mV units, the following conversion is needed for use in (Volt) units:

Data in Volt unit = Data in mV unit/1000

GetFanParam

Call Format BOOL GetFanParam (int Selector, int *pLLimit)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_FAN_CPU	CPU fan
MONITOR_FAN_POWER	Power fan
MONITOR_FAN_OPT	Option fan

(I/O) int *pLLimit Pointer to the lower-limit fan rotation speed (Unit: RPM)

(RPM: Revolutions Per Minute)

Processing Gets the fan monitoring parameter.

Example 1

```
CPL_Ioctl       m_Ioc;
BOOL            ret;
int             LLimit;
// Get the lower-limit CPU fan rotational speed.
ret = m_Ioc.GetFanParam( MONITOR_FAN_CPU, &LLimit );
```

Example 2

```
BOOL            ret;
int             LLimit;
// Get the lower-limit CPU fan rotation speed.
ret = ::GetFanParam( MONITOR_FAN_CPU, &LLimit );
```

GetCurrentFan

Call Format	BOOL GetCurrentFan(int Selector, int *pData)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I) int Selector	Parameters	
		MONITOR_FAN_CPU	CPU fan
		MONITOR_FAN_POWER	Power fan
		MONITOR_FAN_OPT	Option fan
	(I/O) int *pData	Pointer to the fan rotation speed (Unit: RPM)	
		(RPM: Revolutions Per Minute)	
Processing	Gets the current fan rotation speed.		

Example 1

```
CPL_Ioctl        m_Ioc;
BOOL             ret;
int              Data;
// Get the CPU fan rotational speed.
ret = m_Ioc.GetCurrentFan( MONITOR_FAN_CPU, &Data );
```

Example 2

```
BOOL             ret;
int              Data;
// Get the CPU fan rotational speed.
ret = ::GetCurrentFan( MONITOR_FAN_CPU, &Data );
```

GetTempParam

Call Format BOOL GetTempParam(int Selector, int *pULimit)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_TEMP_SYSTEM	System temperature
MONITOR_TEMP_CPU	CPU temperature
MONITOR_TEMP_OPT	Option temperature

(I/O) int *pULimit Pointer to the upper-limit temperature
(Unit: Degrees Celsius)

Processing Gets the temperature monitoring parameter.

Example 1

```
CPL_Ioctl    m_Ioc;
BOOL         ret;
int          ULimit;
// Get the system temperature upper-limit value.
ret = m_Ioc.GetTempParam( MONITOR_TEMP_SYSTEM,
&ULimit );
```

Example 2

```
BOOL         ret;
int          ULimit;
ret = ::GetTempParam( MONITOR_TEMP_SYSTEM, &ULimit );
```

GetCurrentTemp

Call Format BOOL GetCurrentTemp(int Selector, int *pData)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Parameters

MONITOR_TEMP_SYSTEM	System temperature
MONITOR_TEMP_CPU	CPU temperature
MONITOR_TEMP_OPT	Option temperature

(I/O) int *pData Pointer to the temperature
(Unit: Degrees Celsius)

Processing Gets the current temperature value.

Example 1

```
CPL_Ioctl    m_Ioc;
BOOL         ret;
int          Data;
// Gets the system temperature value.
ret = m_Ioc.GetCurrentTemp( MONITOR_TEMP_SYSTEM, &Data );
```

Example 2

```
BOOL         ret;
int          Data;
// Gets the system temperature value.
ret = ::GetCurrentTemp( MONITOR_TEMP_SYSTEM, &Data );
```

SetWdtCounter

Call Format BOOL SetWdtCounter(int Counter)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Counter Sets to the watchdog timer's initial (5 to 255) counter value (Unit: Seconds)

Processing Sets watchdog timer's initial counter value.

Example 1

```
CPL_Ioctl          m_Ioc;
BOOL               ret;
// Sets the Watchdog Timer's initial count value to 10.
ret = m_Ioc.SetWdtCounter( 10 );
```

Example 2

```
BOOL               ret;
// Sets the Watchdog Timer's initial count value to 10.
ret = ::SetWdtCounter( 10 );
```

GetWdtCounter

Call Format BOOL GetWdtCounter(int *pCounter)

Return Value TRUE: Normal
FALSE: Error

Arguments (I/O) int *pCounter Pointer to the watchdog timer's initial counter value (Unit: Seconds)

Processing Gets the current watchdog timer's initial counter value.

Example 1

```
CPL_Ioctl          m_Ioc;
BOOL               ret;
int                Counter;
ret = m_Ioc.GetWdtCounter( &Counter );
```

Example 2

```
BOOL               ret;
int                Counter;
ret = ::GetWdtCounter( &Counter );
```

SetWdtMask

Call Format	BOOL SetWdtMask(int Selector, int Mask)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I) int Selector	Setting Item	
		WARNING_LAMP	LAMP
		WARNING_ALARM	ALARM
	(I) int Mask	Masking Information	
		MASK_OFF	Masking disabled
		MASK_ON	Masking enabled
Processing	Sets masking for the warning that is output when watchdog timer time-out occurs.		
Example 1	CPL_Ioctl	m_Ioc;	
	BOOL	ret;	
	// Enable masking for LAMP output.		
	ret = m_Ioc.SetWdtMask(WARNING_LAMP, MASK_ON);		
	// Disable masking for ALARM output.		
	ret = m_Ioc.SetWdtMask(WARNING_ALARM, MASK_OFF);		
Example 2	BOOL	ret;	
	// Enable masking for LAMP output.		
	ret = ::SetWdtMask(WARNING_LAMP, MASK_ON);		
	// Disable masking for ALARM output.		
	ret = ::SetWdtMask(WARNING_ALARM, MASK_OFF);		

GetWdtMask

Call Format BOOL GetWdtMask(int Selector, int *pMask)

Return Value TRUE: Normal
FALSE: Error

Arguments

(I) int Selector	Setting Item		
	WARNING_LAMP	LAMP	
	WARNING_ALARM	ALARM	
(I/O) int *pMask	Pointer to Masking Information		
	MASK_OFF	Masking disabled	
	MASK_ON	Masking enabled	

Processing Gets the masking information for warning output that is created when a watchdog timer time-out occurs.

Example 1

```
CPL_Ioctl      m_Ioc;
BOOL           ret;
int            Mask;
// Gets the LAMP masking information.
ret = m_Ioc.GetWdtMask( WARNING_LAMP, &Mask );
// Get the ALARM masking information.
ret = m_Ioc.GetWdtMask( WARNING_ALARM, &Mask );
```

Example 2

```
BOOL           ret;
int            Mask;
// Gets the LAMP0 masking information.
ret = ::GetWdtMask( WARNING_LAMP, &Mask );
//  Get the ALARM masking information.
ret = ::GetWdtMask( WARNING_ALARM, &Mask );
```

StartWdt

Call Format BOOL StartWdt(void)

Return Value TRUE: Normal
FALSE: Error

Arguments None

Processing Starts watchdog timer countdown.

Example 1

```
CPL_Ioctl      m_Ioc;
BOOL           ret;
ret = m_Ioc.StartWdt();
```

Example 2

```
BOOL           ret;
ret = ::StartWdt();
```

StopWdt

Call Format	BOOL StopWdt(void)	
Return Value	TRUE: Normal	
	FALSE: Error	
Arguments	None	
Processing	Stops watchdog timer countdown.	
Example 1	CPL_Ioctl	m_Ioc;
	BOOL	ret;
	ret = m_Ioc.StopWdt();	
Example 2	BOOL	ret;
	ret = ::StopWdt();	

RestartWdt

Call Format	BOOL RestartWdt(void)	
Return Value	TRUE: Normal	
	FALSE: Error	
Arguments	None	
Processing	Restarts watchdog timer countdown after resetting to the initial value.	
Example 1	CPL_Ioctl	m_Ioc;
	BOOL	ret;
	m_Ioc.RestartWdt();	
Example 2	BOOL	ret;
	ret = ::RestartWdt();	

Note: **Processing cannot be performed if the watchdog timer is stopped.**

RunningWdt

Call Format BOOL RunningWdt(int *pRunFlag)

Return Value TRUE: Normal
FALSE: Error

Arguments (I/O) int *pRunFlag Pointer to Watchdog Timer Operation Status

WATCHDOG_STOP	Stopped
WATCHDOG_COUNTDOWN	Countdown in progress

Processing Gets the watchdog timer's operation status.

Example 1

```
CPL_Ioctl      m_Ioc;
BOOL           ret;
int            RunFlag;
ret = m_Ioc.RunningWdt( &RunFlag );
```

Example 2

```
BOOL           ret;
int            RunFlag;
ret = ::RunningWdt( &RunFlag );
```

SetWarningOut

Call Format BOOL SetWarningOut(int Selector, int WarnOut)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Setting Item

WARNING_LAMP	LAMP
WARNING_ALARM	ALARM

(I) int WarnOut Output Status

OUTPUT_OFF	Output OFF
OUTPUT_ON	Output ON

Processing Sets the warning information for the set item (lamp or alarm).

Example 1

```
CPL_Ioctl      m_Ioc;
BOOL           ret;
// Sets the LAMP output status to ON.
ret = m_Ioc.SetWarningOut( WARNING_LAMP, OUTPUT_ON );
// Sets the ALARMoutput status to OFF.
ret = m_Ioc.SetWarningOut( WARNING_ALARM, OUTPUT_OFF );
```

Example 2

```
BOOL           ret;
// Sets the LAMP output status to ON.
ret = ::SetWarningOut( WARNING_LAMP, OUTPUT_ON );
// Sets the ALARM output status to OFF.
ret = ::SetWarningOut( WARNING_ALARM, OUTPUT_OFF );
```

GetWarningOut

Call Format	BOOL GetWarningOut(int Selector, int *pWarnOut)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I) int Selector Settting Item WARNING_LAMP LAMP WARNING_ALARM ALARM (I/O) int *pWarnOut Pointer to Output Status OUTPUT_OFF Output OFF OUTPUT_ON Output ON
Processing	Gets the warning status of the current set item (LAMP or ALARM).

Example 1

```
CPL_Ioctl         m_Ioc;
BOOL              ret;
int               WarnOut;
// Gets the LAMP output status.
ret = m_Ioc.GetWarningOut( WARNING_LAMP, &WarnOut );
// Gets the ALARM output status.
ret = m_Ioc.GetWarningOut( WARNING_ALARM, &WarnOut );
```

Example 2

```
BOOL       ret;
int        WarnOut;
// Gets the LAMP output status.
ret = ::GetWarningOut( WARNING_LAMP, &WarnOut );
// Gets the ALARM output status.
ret = ::GetWarningOut( WARNING_ALARM, &WarnOut );
```

GetUniversalIn

Call Format BOOL GetUniversalIn(int Selector, int *pUniIn)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int Selector Designated Port

PORT_UNI0	Universal Input 0
PORT_UNI1	Universal Input 1

(I/O) int *pUniIn Pointer to Input Status

INPUT_OFF	Input OFF
INPUT_ON	Input ON

Processing Gets the input status of the designated port (Universal Input 0, Universal Input 1).

Example1

```
CPL_Ioctl       m_Ioc;
BOOL            ret;
int             UniIn;
// Gets the input status of Universal Input 0.
ret = m_Ioc.GetUniversalIn( PORT_UNI0, &UniIn );
// Gets the input status of Universal Input 1.
ret = m_Ioc.GetUniversalIn( PORT_UNI1, &UniIn );
```

Example 2

```
BOOL            ret;
int             UniIn;
// Gets the input status of Universal Input 0.
ret = ::GetUniversalIn( PORT_UNI0, &UniIn );
// Gets the input status of Universal Input 1.
ret = ::GetUniversalIn( PORT_UNI1, &UniIn );
```

ClearUniversalIn

Call Format BOOL ClearUniversalIn(int Selector)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Selector Designated Port

PORT_UNI0	Universal Input 0
PORT_UNI1	Universal Input 1

Processing Clears the input status of the designated port (Universal Input 0, Universal Input 1).

Example 1

```
CPL_Ioctl        m_Ioc;
BOOL             ret;
// Cancels the output of Universal Input 0.
ret = m_Ioc.ClearUniversalIn( PORT_UNI0 );
// Cancels the output of Universal Input 1.
ret = m_Ioc.ClearUniversalIn( PORT_UNI1 );
```

Example 2

```
BOOL             ret;
// Cancels the output of Universal Input 0.
ret = ::ClearUniversalIn( PORT_UNI0 );
// Cancels the output of Universal Input 1.
ret = ::ClearUniversalIn( PORT_UNI1 );
```

SetUniversalInMask

Call Format	BOOL SetUniversalInMask(int Selector, int Mask)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I) int Selector	Designated Port	
		PORT_UNI0	Universal Input 0
		PORT_UNI1	Universal Input 1
	(I/O) int Mask	Masking Information	
		MASK_OFF	Masking disabled
		MASK_ON	Masking enabled
Processing	Sets the masking information for the designated port (Universal Input 0, Universal Input 1).		

Example 1

```
CPL_Ioctl        m_Ioc;
BOOL             ret;
// Disable masking for Universal Input 0.
ret = m_Ioc.SetUniversalInMask( PORT_UNI0, MASK_OFF );
// Enable masking for Universal Input 1.
ret = m_Ioc.SetUniversalInMask( PORT_UNI1, MASK_ON );
```

Example 2

```
BOOL              ret;
// Disable masking for Universal Input 0.
ret = ::SetUniversalInMask( PORT_UNI0, MASK_OFF );
// Enable masking for Universal Input 1.
ret = ::SetUniversalInMask( PORT_UNI1, MASK_ON );
```

GetUniversalInMask

Call Format BOOL GetUniversalInMask(int Selector, int *pMask)

Return Value TRUE: Normal
FALSE: Error

Arguments
(I) int Selector Designated Port
PORT_UNI0 Universal Input 0
PORT_UNI1 Universal Input 1
(I/O) int *pMask Pointer to Masking Information
MASK_OFF Masking disabled
MASK_ON Masking enabled

Processing Gets the masking information for the designated port (Universal Input 0, Universal Input 1).

Example 1

```
CPL_Ioctl      m_Ioc;
BOOL           ret;
int            Mask;
// Gets the masking information for Universal input 0.
ret = m_Ioc.GetUniversalInMask( PORT_UNI0, &Mask );
// Gets the masking information for Universal input 1.
ret = m_Ioc.GetUniversalInMask( PORT_UNI1, &Mask );
```

Example 2

```
BOOL           ret;
int            Mask;
// Gets the masking information for Universal input 0.
ret = ::GetUniversalInMask( PORT_UNI0, &Mask );
// Gets the masking information for Universal input 1.
ret = ::GetUniversalInMask( PORT_UNI1, &Mask );
```

SetResetMask

Call Format BOOL SetResetMask(int Mask)

Return Value TRUE: Normal
FALSE: Error

Arguments
(I) int Mask Masking Information
MASK_OFF Masking disabled
MASK_ON Masking enabled

Processing Sets reset-masking.

Example 1

```
CPL_Ioctl      m_Ioc;
BOOL           ret;
// Disable reset-masking.
ret = m_Ioc.SetResetMask( MASK_OFF );
```

Example 2

```
BOOL           ret;
// Disable reset-masking.
ret = ::SetResetMask( MASK_OFF );
```

GetResetMask

Call Format	BOOL GetResetMask(int *pMask)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I/O) int *pMask Pointer to Masking Information MASK_OFF Masking disabled MASK_ON Masking enabled
Processing	Gets the current reset-masking information.

Example 1	CPL_Ioctl m_Ioc;
	BOOL ret;
	int Mask;
	ret = m_Ioc.GetResetMask(&Mask);

Example 2	BOOL ret;
	int Mask;
	ret = ::GetResetMask(&Mask);

SetIdeErr

Call Format	BOOL SetIdeErr(int IdeErr)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I) int IdeErr Error Output Information IDE_ERROR_OFF Error Output OFF IDE_ERROR_ON Error Output ON
Processing	Uses software control to set IDE error output.

Example 1	CPL_Ioctl m_Ioc;
	BOOL ret;
	ret = m_Ioc.SetIdeErr(IDE_ERROR_OFF);

Example 2	BOOL ret;
	ret = ::SetIdeErr(IDE_ERROR_OFF);

GetIdeErrHard

Call Format	BOOL GetIdeErrHard(int Selector, int *pIdeErr)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I) int Selector	Parameters	
		IDE_ERROR_1	IDE_ERR1
		IDE_ERROR_2	IDE_ERR2
	(I/O) int *pIdeErr	Pointer to Output Status	
		IDE_ERROR_OFF	Normal
		IDE_ERROR_ON	Error
Processing	Gets hardware's current IDE error signal.		
Example 1	CPL_Ioctl	m_Ioc;	
	BOOL	ret;	
	int	IdeErr;	
	// Gets the IDE_ERR1 signal status		
	ret = m_Ioc.GetIdeErrHard(IDE_ERROR_1, &IdeErr);		
Example 2	BOOL	ret;	
	int	IdeErr;	
	// Gets the IDE_ERR1 signal status		
	ret = ::GetIdeErrHard(IDE_ERROR_1, &IdeErr);		

GetLightblowErr

Call Format	BOOL GetLightblowErr(int *pLightErr)		
	TRUE: Normal		
	FALSE: Error		
Arguments	(I/O) int *pLightErr	Error Information	
		BACKLIGHT_OK	OK
		BACKLIGH_ERR	NG
Processing	Gets Backlight's current burnout error output.		
Example 1	CPL_Ioctl	m_Ioc;	
	BOOL	ret;	
	int	LightErr;	
	// Gets backlight's burnout condition.		
	ret = m_Ioc.GetLightblowErr(&LightErr);		
Example 2	BOOL	ret;	
	int	LightErr;	
	// Gets backlight's burnout condition.		
	ret = ::GetLightblowErr(&LightErr);		

Note: **This feature can be used only with PL-6920 Series units.**

GetEvent

Call Format BOOL GetEvent(int Selector, int *pEvent)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Selector Parameter s

EVENT_VOLT_CPU	CPU core voltage
EVENT_VOLT_P33	+3.3 V
EVENT_VOLT_P50	+5.0 V
EVENT_VOLT_P12	+12 V
EVENT_VOLT_M12	-12 V
EVENT_VOLT_M50	-5.0 V
EVENT_VOLT_VIT	CPU core voltage 2
EVENT_FAN_CPU	CPU fan
EVENT_FAN_POWER	Power fan
EVENT_FAN_OPT	Option fan
EVENT_TEMP_SYSTEM	System temperature
EVENT_TEMP_CPU_OPT	CPU or option temperature
EVENT_UNI_IN0	Universal input 0
EVENT_UNI_IN1	Universal input 1
EVENT_WDT_TIMEOUT	Watchdog Timeout

(I/O) int *pEvent Pointer to Error Event Information

ERROR_EVENT_OFF	Without error event
ERROR_EVENT_ON	With error event

Processing Checks the machine for voltage, fan, and temperature errors, and the Universal Input information (event) and Watchdog Timeout error.

Example 1

```
CPL_Ioctl       m_Ioc;
BOOL            ret;
int             Event;
// Gets the error event information for the CPU core voltage.
ret = m_Ioc.GetEvent( EVENT_VOLT_CPU, &Event );
```

Example 2

```
BOOL    ret;
int     Event;
// Gets the error event information for the CPU core voltage.
ret = ::GetEvent( EVENT_VOLT_CPU, &Event );
```

ClearEvent

Call Format BOOL ClearEvent(int Selector)

Return Value TRUE: Normal

FALSE: Error

Arguments (I) int Selector Designated Parameters for ClearEvent

EVENT_VOLT_CPU	CPU core voltage
EVENT_VOLT_P33	+3.3 V
EVENT_VOLT_P50	+5.0 V
EVENT_VOLT_P12	+12 V
EVENT_VOLT_M12	-12 V
EVENT_VOLT_M50	-5.0 V
EVENT_VOLT_VIT	CPU core voltage 2
EVENT_FAN_CPU	CPU fan
EVENT_FAN_POWER	Power fan
EVENT_FAN_OPT	Option fan
EVENT_TEMP_SYSTEM	System temperature
EVENT_TEMP_CPU_OPT	CPU or option temperature
EVENT_UNI_IN0	Universal input 0
EVENT_UNI_IN1	Universal input 1
EVENT_WDT_TIMEOUT	Watchdog Timeout

Processing Cancels the error event.

Example 1

```
CPL_Ioctl       m_Ioc;
BOOL            ret;
// Cancels the error event for the CPU core voltage.
ret = m_Ioc.ClearEvent( EVENT_VOLT_CPU );
```

Example 2

```
BOOL            ret;
// Cancels the error event for the CPU core voltage.
ret = ::ClearEvent( EVENT_VOLT_CPU );
```

StartInsideBuzzer

Call Format	BOOL WINAPI StartInsideBuzzer (int hz, int ms)	
Return Value	BOOL	TRUE: Normal FALSE: Error
Arguments	(I) int hz	Buzzer frequency (Hz)
	(I) int ms	Buzzer length (ms)
Processing	Starts the PL unit's internal buzzer, based on the designated frequency and length.	

Example 1

```
BOOL    ret;
int     hz = 600;
int     ms = 1000;
// PL internal buzzer will sound at 600MHz for 1 second.
ret = m_Ioc.StartInsideBuzzer ( hz, ms );
```

Example 2

```
BOOL    ret;
int     hz = 600;
int     ms = 1000;
// PL internal buzzer will sound at 600MHz for 1 second.
ret = ::StartInsideBuzzer ( hz, ms );
```

This feature cannot be used with a PL running WindowsNT 4.0, Windows 2000 or Windows XP due to the use of Windows 95 and Windows 98 functions.

StopInsideBuzzer

Call Format	BOOL WINAPI StopInsideBuzzer (void)	
Return Value	BOOL	TRUE: Normal FALSE: Error
Arguments	None.	
Processing	Stops the PL unit's internal buzzer.	

Example 1

```
CPL_Ioctl       m_loc;
BOOL            ret;
// Stops PL internal buzzer.
ret = m_Ioc.StopInsideBuzzer ( );
```

Example 2

```
BOOL            ret;
// Stops PL internal buzzer.
ret = ::StopInsideBuzzer ( );
```

This feature cannot be used with a PL running WindowsNT 4.0, Windows 2000 Windows XP due to the use of Windows 95 and Windows 98 functions.

ChkInsideBuzzer

Call Format BOOL WINAPI ChkInsideBuzzer (int *BuzzerParam)

Return Value BOOL TRUE: Normal
FALSE: Error

Arguments (I/O) int *BuzzerParam Buzzer Status Pointer
BUZZER_ON Buzzer is ON
BUZZER_OFF Buzzer is OFF

Processing Checks the buzzer's operation status.

Example 1
```
CPL_Ioctl      m_loc;
BOOL           ret;
int            BuzzerParam;
// Checks buzzer status.
ret = m_Ioc.ChkInsideBuzzer ( &BuzzerParam );
```

Example 2
```
BOOL    ret;
// Checks buzzer status.
ret = ::ChkInsideBuzzer ( &BuzzerParam );
```

Note: **This feature cannot be used with a PL unit running WindowsNT 4.0, Windows 2000 or Windows XP, due to the use of Windows 95 and Windows 98 functions.**

GetWdtTimeout

Call Format BOOL GetWdtTimeout(int *pTimebuf)

Return Value TRUE: Normal
FALSE: Error

Arguments (I/O) int *pTimebuf Pointer to Watchdog Status
TIMEOUT_OK Not timeout
TIMEOUT_ERROR Now timeout

Processing Gets watchdog timeout status.

Example 1
```
CPL_Ioctl      m_Ioc;
BOOL           ret;
int            Timebuf;
// Gets watchdog timeout status.
ret = Gm_Ioc.GetWdtTimeout( &Timebuf );
```

Example 2
```
BOOL    ret;
int     Timebuf;
// Gets watchdog timeout status.
ret = ::GetWdtTimeout( &Timebuf );
```

ClearWdtTimeout

Call Format	BOOL ClearWdtTimeout(void)
Return Value	TRUE: Normal FALSE: Error
Arguments	None
Processing	Clears the watchdog timeout status.

Example 1	CPL_Ioctl m_Ioc;
	BOOL ret;
	// Clears the watchdog timeout status.
	ret = m_Ioc.GetWdtTimeout();

Example 2	BOOL ret;
	// Clears the watchdog timeout status.
	ret = ::GetWdtTimeout();

SetWarningDOUT

Call Format	BOOL SetWarningDOUT(int WarningOut)		
Return Value	TRUE: Normal FALSE: Error		
Arguments	(I) int WarningOut	Output status	
		OUTPUT_OFF	Output OFF
		OUTPUT_ON	Output ON
Processing	Sets alarm status of DOUT.		

Example 1	CPL_Ioctl m_Ioc;
	BOOL ret;
	// Sets DOUT output status to OFF.
	ret = m_Ioc.SetWarningDOUT(OUTPUT_OFF);		

Example 2	BOOL ret;
	// Sets DOUT output status to OFF.
	ret = ::SetWarningDOUT(OUTPUT_OFF);		

GetWarningDOUT

Call Format	BOOL GetWarningDOUT(int *pWarningOUT)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I/O) int *pWarningOut Pointer to Output Status OUTPUT_OFF Output OFF OUTPUT_ON Output ON
Processing	Gets alarm status of DOUT.

Example 1	CPL_Ioctl m_Ioc;
	BOOL ret;
	int WarningOut;
	// Gets alarm status of DOUT.
	ret = m_Ioc.GetWarningDOUT(&WarningOut);

Example 2	BOOL ret;
	int WarningOut;
	// Gets alarm status of DOUT.
	ret = ::GetWarningDOUT(&WarningOut);

GetSmiDrvHandle

Call Format	int GetSmiDrvHandle(void)
Return Value	0: Normal 1 : Error
Arguments	NONE
Processing	Gets device driver handle for communication with Software Mirroring device driver.

Example 1	CPL_SmiIoctl m_SmiIoc;
	BOOL ret;
	// Gets Software Mirroring driver handle.
	ret = m_SmiIoc.GetSmiDrvHandle();

Example 2	BOOL ret;
	// Gets Software Mirroring driver handle.
	ret = ::GetSmiDrvHandle();

Note: **When the Software Mirroring Driver is not loaded, an error is returned.**

CloseSmiDrvHandle

Call Format	BOOL CloseSmiDrvHandle(void)	
Return Value	True: Normal	
	False: Error	
Arguments	NONE	
Processing	Destroys handle created in GetSmiDrvHandle.	
Example 1	CPL_SmiIoctl	m_SmiIoc;
	BOOL	ret;
	// Destroys Software Mirroring driver handle.	
	ret = m_SmiIoc.CloseSmiDrvHandle();	
Example 2	BOOL	ret;
	// Destroys Software Mirroring driver handle.	
	ret = ::CloseSmiDrvHandle();	

GetSmiAryStatus

Call Format	BOOL GetSmiAryStatus(int *pStatus)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I/O) int *pStatus	Pointer to Mirroring Status	
		ARYSTAT_GOOD	Good
		ARYSTAT_NOTEXIST	Not output
		ARYSTAT_UNCONFIG	Unconfigured
		ARYSTAT_REBUILD	Rebuilding
		ARYSTAT_REDUCE	Reduced
		ARYSTAT_DEAD	Dead
Processing	Gets Software Mirroring status.		
Example 1	CPL_SmiIoctl	m_Smiloc;	
	BOOL	ret;	
	int	Status;	
	// Gets Software Mirroring status.		
	ret = m_Smiloc.GetSmiAryStatus(&Status);		
Example 2	BOOL	ret;	
	int	Status;	
	// Gets Software Mirroring status.		
	ret = ::GetSmiAryStatus(&Status);		

GetSmiDevStatus

Call Format	BOOL GetSmiDevStatus(int Id ,int *pType ,int *pStatus)		
Return Value	TRUE: Normal		
	FALSE: Error		
Arguments	(I) int Id	Device ID	
		0 : Master HDD	
		1 : Slave HDD	
	(I/O int* pType	Device Type	
		ATADEVICE	ATA DEVICE
		ATAPIDEVICE	CD-ROM
		UNKNOWNDEVICE	Unknown DEVICE
		NODEVICE	No DEVICE
	(I/O) int* pStatus	Device Status	
		DEVSTAT_GOOD	Good
		DEVSTAT_NOTEXIST	No DEVICE
		DEVSTAT_BROKEN	BROKEN
Processing	Gets Device Status of software mirroring.		
Example 1	CPL_SmiIoctl	m_SmiIoc;	
	BOOL	ret;	
	int	Id, Type, Status;	
	// Gets device status.		
	Id = 0;		
	ret = m_SmiIoc.GetSmiDevStatus(ID ,&Type ,&Status);		
Example 2	BOOL	ret;	
	int	Id, Type, Status;	
	// Gets device status.		
	Id = 0;		
	ret = ::GetSmiDevStatus(ID ,&Type ,&Status);		

SetWdtResetMask

Call Format	BOOL SetWdtResetMask(int Mask)	
Return Value	TRUE: Normal	
	FALSE: Error	
Arguments	(I/O) int Mask	Masking Information
		MASK_OFF Masking disabled
		MASK_ON Masking enabled
Processing	Sets WDT Timeout H/W reset-masking.	

Example 1

```
CPL_Ioctl       m_Ioc;
BOOL            ret;
// Disables mask set during WDT timeout.
ret = m_Ioc.SetWdtResetMask( MASK_OFF );
```

Example 2

```
BOOL            ret;
// Disables mask set during WDT timeout.
ret = ::SetWdtResetMask( MASK_OFF );
```

GetWdtResetMask

Call Format	BOOL GetWdtResetMask(int *pMask)	
Return Value	TRUE: Normal	
	FALSE: Error	
Arguments	(I/O) int *pMask	Pointer to Masking Information
		MASK_OFF Masking disabled
		MASK_ON Masking enabled
Processing	Gets the current WDT timeout H/W reset-masking information.	

Example 1

```
CPL_Ioctl       m_Ioc;
BOOL            ret;
int             Mask;
// Gets data of WDT timeout reset mask.
ret = m_Ioc.GetWdtResetMask( &Mask );
```

Example 2

```
BOOL            ret;
int             Mask;
// Gets data of WDT timeout reset mask.
ret = ::GetWdtResetMask( &Mask );
```

7 Visual Basic Functions

Function Name	Description
InitIoctl	Creates a CPL_Ioctl object
EndIoctl	Destroys a CPL_Ioctl object
GetDrvHandle	Gets the driver handle
CloseDrvHandle	Destroys the driver handle
GetDrvVersion	Gets the driver version
GetMonitorSetup	Gets the enabled/disabled monitor settings
GetVoltParam	Gets the voltage monitoring parameters
GetCurrentVolt	Gets the current value of the voltage
GetFanParam	Gets the parameters for monitoring the FAN
GetCurrentFan	Gets the current value of the FAN
GetTempParam	Gets the parameters for monitoring the temperature
GetCurrentTemp	Gets the current value of the temperature
SetWdtCounter	Sets the value for the watchdog timer counter
GetWdtCounter	Gets the watchdog timer counter
SetWdtMask	Sets the watchdog timer counter time-out status warning mask
GetWdtMask	Gets the watchdog timer counter time-out status warning mask
StartWdt	Starts the watchdog timer
StopWdt	Stops the watchdog timer
RestartWdt	Restarts the watchdog timer
RunningWdt	Gets the watchdog status
SetWarningOut	Sets the warning output
GetWarningOut	Gets the warning output
GetUniversalIn	Gets the universal input
ClearUniversalIn	Clears the universal input latch
SetUniversalInMask	Sets the universal input mask
GetUniversalInMask	Gets the universal input mask
SetResetMask	Sets the reset mask
GetResetMask	Gets the reset mask
SetIdeErr	Sets the mirroring error (software error)
GetIdeErrHard	Gets the mirroring error (hardware error)
GetEvent	Gets an error event
ClearEvent	Clears an error event
StartInsideBuzzer	Starts PL internal buzzer
StopInsideBuzzer	Stops PL internal buzzer
ChkInsideBuzzer	Checks PL internal buzzer
GetWdtTimeout	Gets the time-out status of the watchdog timer
ClearWdtTimeout	Clear the time-out status of the watchdog timer
SetWarningDOUT	Sets the warning output DOUT
GetWarningDOUT	Gets the warning output DOUT
GetSemiDrvHandle	Gets Software Mirroring driver handle
CloseSmiDrvHandle	Destroys Software Mirroring driver handle
GetSmiAryStatus	Gets status of Software Mirroring Array
GetSmiDevStatus	Gets status of Software Mirroring Device
SetWdtResetMask	Sets the Reset Mask of the watchdog timer
GetWdtResetMask	Gets the Reset Mask of the watchdog timer

8 Visual Basic Function Specifications (Details)

InitIoctl

Call format	Declare Sub InitIoctl Lib "PL_Ioc.dll" ()
Return value	None
Argument	None
Processing	Creates a CPL_Ioctl object. The created object will not be released until the "EndIoctl" function is called.
Example	InitIoctl()

EndIoctl

Call format	Declare Sub EndIoctl Lib "PL_Ioc.dll" ()
Return value	None
Argument	None
Processing	Destroys the object created with the "InitIoctl" function.
Example	EndIoctl()

GetDrvHandle

Call format	Declare Function GetDrvHandle Lib "PL_Ioc.dll" (ByRef hndl As Long) As Long	
Return value	0: Normal 1: Error	
Argument	hndl As Long	Device driver handle (pass by reference)
Processing	Gets the device driver handle to exchange information with the device driver.	
Example	Dim ret As Long Dim hndl As Long ret = GetDrvHandle(hndl)	

Note: **An error will result if the system monitor/RAS device driver is not operating.**

CloseDrvHandle

Call format	Declare Function CloseDrvHandle Lib "PL_Ioc.dll"() As Long
Return value	Other than 0: Normal 0: Error
Argument	None
Processing	Destroys the handle acquired with the "GetDrvHandle" function.

Example	Dim ret As Long
	// Destroy handle
	ret = CloseDrvHandle()

GetDrvVersion

Call format Declare Function GetDrvVersion Lib "PL_Ioc.dll"
(ByRef Major As Long, ByRef Minor As Long) As Long

Return value Other than 0: Normal
0: Error

Argument
Major As Long Version data (Major, 0 to 99) (pass by reference)
Minor As Long Version data (Major, 0 to 99) (pass by reference)

Processing Gets the driver version.

Example
```
Dim ret As Long
Dim Major As Long
Dim Minor As Long
ret = GetDrvVersion(Major, Minor)
```

Note:

When the version is 1.10,

Major:1	**(Decimal)**
Minor:10	**(Decimal)**

GetMonitorSetup

Call format Declare Function GetMonitorSetup Lib "PL_Ioc.dll"
(ByVal Selector As Long, ByRef Setup As Long) As Long

Return value Other than 0: Normal
0: Error

Argument Selector As Long Parameters (pass by value)

MONITOR_VOLT_CPU	CPU Core voltage
MONITOR_VOLT_P33	+3.3V
MONITOR_VOLT_P50	+5.0V
MONITOR_VOLT_P12	+12V
MONITOR_VOLT_M12	-12V
MONITOR_VOLT_M50	-5.0V
MONITOR_VOLT_VIT	CPU Core voltage 2
MONITOR_TEMP_SYSTEM	SYSTEM Temp.
MONITOR_TEMP_CPU	CPU Temp.
MONITOR_TEMP_OPT	OPTION Temp.
MONITOR_FAN_CPU	CPU FAN
MONITOR_FAN_POWER	POWER FAN
MONITOR_FAN_OPT	OPTION FAN

Setup As Long Get data (pass by reference)
0:Disable
1:Enable

Processing Gets the current enabled/disabled monitor status.

Example
```
Dim ret As Long
Dim Setup As Long
// Get the setup status of the CPU core voltage
ret = GetMonitorSetup(MONITOR_VOLT_CPU, Setup )
```

GetVoltParam

Call format Declare Function GetVoltParam Lib "PL_Ioc.dll"

(ByVal Selector As Long, ByRef ULimit As Long, ByRef LLimit As Long) As Long

Return value Other than 0: Normal

0: Error

Argument Selector As Long Parameters (pass by value)

MONITOR_VOLT_CPU	CPU Core voltage
MONITOR_VOLT_P33	+3.3V
MONITOR_VOLT_P50	+5.0V
MONITOR_VOLT_P12	+12V
MONITOR_VOLT_M12	-12V
MONITOR_VOLT_M50	-5.0V
MONITOR_VOLT_VIT	CPU Core voltage 2

Ulimit As Long Voltage value upper limit (unit: mV) (pass by reference)

LLimit As Long Voltage value lower limit (unit: mV) (pass by reference)

Processing Gets the voltag monitoring parameter.

Example

```
Dim ret As Long
Dim ULimit  As Long
Dim LLimit As Long
// Get the upper/lower limit of the CPU core voltage value
ret = GetVoltParam( MONITOR_VOLT_CPU, ULimit, LLimit )
```

Note: **Since the data received from this function is in mV units, the following conversion is needed for use in (Volt) units:**

Data in Volt unit = Data in mV unit/1000

GetCurrentVolt

Call format Declare Function GetCurrentVolt Lib "PL_Ioc.dll"
(ByVal Selector As Long, ByRef Data As Long) As Long

Return value Other than 0: Normal
0: Error

Argument Selector As Long Parameters (pass by value)

MONITOR_VOLT_CPU	CPU Core voltage
MONITOR_VOLT_P33	+3.3V
MONITOR_VOLT_P50	+5.0V
MONITOR_VOLT_P12	+12V
MONITOR_VOLT_M12	-12V
MONITOR_VOLT_M50	-5.0V
MONITOR_VOLT_VIT	CPU Core voltage 2

Data As Long Voltage value (unit: mV) (pass by reference)

Processing Gets the current voltage value.

Example Dim ret As Long
Dim Data As Long
// Get the CPU core voltage value.
ret = GetCurrentVolt(MONITOR_VOLT_CPU, Data)

Note:

Since the data received from this function is in mV units, the following conversion is needed for use in (Volt) units:

Data in Volt unit = Data in mV unit/1000

GetFanParam

Call format Declare Function GetFanParam Lib "PL_Ioc.dll"
(ByVal Selector As Long, ByRef LLimit As Long) As Long

Return value Other than 0: Normal
0: Error

Argument Selector As Long Parameters (pass by value)

MONITOR_FAN_CPU	CPU FAN
MONITOR_FAN_POWER	POWER FAN
MONITOR_FAN_OPT	OPTION FAN

LLimit As Long CPU FAN revolution lower limit value (unit: RPM) (pass by value)
(RPM: revolutions per minute)

Processing Gets the parameter for monitoring the FAN.

Example Dim ret As Long
Dim LLimit As Long
// Get the CPU FAN lower limit rpm value
ret = GetFanParam(MONITOR_FAN_CPU, LLimit)

GetCurrentFan

Call format	Declare Function GetCurrentFan Lib "PL_Ioc.dll" (ByVal Selector As Long, ByRef Data As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	Selector As Long	Parameters (pass by value)	
		MONITOR_FAN_CPU	CPU FAN
		MONITOR_FAN_POWER	POWER FAN
		MONITOR_FAN_OPT	OPTION FAN
	Data As Long	CPU FAN revolution lower limit value (unit: RPM) (pass by reference) (RPM: revolutions per minute)	
Processing	Gets the current FAN rpm.		

Example	Dim ret As Long
	Dim Data As Long
	// Get the number of revolutions of the CPU FAN
	ret = GetCurrentFan(MONITOR_FAN_CPU, Data)		

GetTempParam

Call format	Declare Function GetTempParam Lib "PL_Ioc.dll" (ByVal Selector As Long, ByRef ULimit As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	Selector As Long	Get parameter (pass by value)	
		MONITOR_TEMP_SYSTEM	SYSTEM temp.
		MONITOR_TEMP_CPU	CPU temp.
		MONITOR_TEMP_OPT	OPTION temp.
	Ulimit As Long	Temperature upper limit (unit: °C) (pass by reference)	
Processing	Gets the parameter for monitoring the temperature.		

Example	Dim ret As Long
	Dim ULimit As Long
	// Gets the upper limit of SYSTEM temperature
	ret = GetTempParam(MONITOR_TEMP_SYSTEM, ULimit)		

GetCurrentTemp

Call format Declare Function GetCurrentTemp Lib "PL_Ioc.dll"
(ByVal Selector As Long, ByRef Data As Long) As Long

Return value Other than 0: Normal
0: Error

Argument Selector As Long Parameters (pass by value)

MONITOR_TEMP_SYSTEM	SYSTEM temp.
MONITOR_TEMP_CPU	CPU temp.
MONITOR_TEMP_OPT	OPTION temp.

Data As Long Temperature value (unit: °C)
(pass by reference)

Processing Gets the current temperature value.

Example
```
Dim ret As Long
Dim Data As Long
// Get the current value of SYSTEM temperature
ret = GetCurrentTemp( MONITOR_TEMP_SYSTEM, Data )
```

SetWdtCounter

Call format Declare Function SetWdtCounter Lib "PL_Ioc.dll"
(ByVal Counter As Long) As Long

Return value Other than 0: Normal
0: Error

Argument Counter As Long The initial counter value of the watchdog timer
(pass by value) (5 to 255) (unit: second)

Processing Sets the initial counter value for the watchdog timer.

Example
```
Dim ret As Long
// Sets the initial counter value for the watchdog timer to 10 seconds
ret = SetWdtCounter( 10 )
```

GetWdtCounter

Call format Declare Function GetWdtCounter Lib "PL_Ioc.dll"
(ByVal Counter As Long) As Long

Return value Other than 0: Normal
0: Error

Argument Counter As Long The initial counter value of the watchdog timer
(pass by value) (unit: second)

Processing Gets the initial counter value of the current watchdog timer.

Example
```
Dim ret As Long
Dim Counter As Long
ret = GetWdtCounter(Counter)
```

SetWdtMask

Call format

```
Declare Function SetWdtMask Lib "PL_Ioc.dll"
(ByVal Selector As Long, ByVal Mask As Long) As Long
```

Return value

Other than 0: Normal
0: Error

Argument

Selector As Long	Setup items (pass by value)		
	WARNING_LAMP	LAMP	
	WARNING_ALARM	ALARM	
Mask As Long	Mask data (pass by value)		
	MASK_OFF	Release mask	
	MASK_ON	Mask	

Processing

Sets the warning mask to be output when a watchdog timer time-out occurs.

Example

```
Dim ret As Long
// Mask the LAMP output
ret = SetWdtMask( WARNING_LAMP, MASK_ON )
// Release the mask for the ALARM output
ret = SetWdtMask( WARNING_ALARM, MASK_OFF )
```

GetWdtMask

Call format

```
Declare Function GetWdtMask Lib "PL_Ioc.dll"
(ByVal Selector As Long, ByRef Mask As Long) As Long
```

Return value

Other than 0: Normal
0: Error

Argument

Selector As Long	Setup items (pass by reference)		
	WARNING_LAMP	LAMP	
	WARNING_ALARM	ALARM	
Mask As Long	(pass by reference)		
	MASK_OFF	Release the mask	
	MASK_ON	Mask	

Processing

Gets the WDT timeout warning output mask data.

Example

```
Dim ret As Long
Dim Mask As Long
// Gets LAMP mask data
ret = GetWdtMask( WARNING_LAMP, Mask )
// Gets ALARM mask data
ret = GetWdtMask( WARNING_ALARM, Mask )
```

StartWdt

Call format	Declare Function StartWdt Lib "PL_Ioc.dll" () As Long
Return value	Other than 0: Normal
	0: Error
Argument	None
Processing	Stops the WDT countdown.
Example	Dim ret As Long
	ret = StartWdt()

StopWdt

Call format	Declare Function StopWdt Lib "PL_Ioc.dll" () As Long
Return value	Other than 0: Normal
	0: Error
Argument	None
Processing	Stops the WDT countdown.
Example	Dim ret As Long
	ret = StopWdt()

RestartWdt

Call format	Declare Function RestartWdt Lib "PL_Ioc.dll" () As Long
Return value	Other than 0: Normal
	0: Error
Argument	None
Processing	Resets the initial value of the watchdog timer to the default value, and restarts the countdown.
Example	Dim ret As Long
	ret = RestartWdt()

Note:

Processing cannot be performed if the watchdog timer is stopped.

RunningWdt

Call format	Declare Function RunningWdt Lib "PL_Ioc.dll" (ByRef RunFlag As Long) As Long	
Return value	Other than 0: Normal	
	0: Error	
Argument	RunFlag As Long	Operating status of the watchdog timer (pass by reference)
	WATCHDOG_STOP	Stopped
	WATCHDOG_COUNTDOWN	Counting down
Processing	Gets the operating status of the watchdog timer.	
Example	Dim ret As Long	
	Dim RunFlag As Long	
	ret = RunningWdt(RunFlag)	

SetWarningOut

Call format	Declare Function SetWarningOut Lib "PL_Ioc.dll" (ByVal Selector As Long, ByVal WarnOut As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	Selector As Long	Setting items (pass by value)	
		WARNING_LAMP	LAMP
		WARNING_ALARM	ALARM
	WarnOut As Long	Output condition (pass by value)	
		OUTPUT_OFF	Output OFF
		OUTPUT_ON	Output ON
Processing	Sets warning data for the setup items (LAMP and ALARM).		

Example	Dim ret As Long
	// Set the output status of the LAMP to ON
	ret = SetWarningOut(WARNING_LAMP, OUTPUT_ON)
	// Set the output status of the ALARM to OFF
	ret = SetWarningOut(WARNING_ALARM, OUTPUT_OFF)		

GetWarningOut

Call format	Declare Function GetWarningOut Lib "PL_Ioc.dll" (ByVal Selector As Long, ByRef WarnOut As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	Selector As Long	Setting items (pass by value)	
		WARNING_LAMP	LAMP
		WARNING_ALARM	ALARM
	WarnOut As Long	Output condition (pass by reference)	
		OUTPUT_OFF	Output OFF
		OUTPUT_ON	Output ON
Processing	Gets the current warning status of the setup items (LAMP and ALARM).		

Example	Dim ret As Long
	Dim WarnOut As Long
	// Gets the output status of the LAMP
	ret = GetWarningOut(WARNING_LAMP, WarnOut)
	// Get the output status of the ALARM
	ret = GetWarningOut(WARNING_ALARM, WarnOut)		

GetUniversalIn

Call format	Declare Function GetUniversalIn Lib "PL_Ioc.dll"		
	(ByVal Selector As Long, ByRef UniIn As Long) As Long		
Return value	Other than 0: Normal		
	0: Error		
Argument	Selector As Long	Designated port (pass by value)	
		PORT_UNI0	Universal Input 0
		PORT_UNI1	Universal Input 1
	UniIn As Long	Input status (pass by reference)	
		INPUT_OFF	No input
		INPUT_ON	Input
Processing	Gets the input status of the designated port (Universal Input 0 and Universal Input 1).		
Example	Dim ret As Long		
	Dim UniIn As Long		
	// Get the input status of the Universal Input 0		
	ret = GetUniversalIn(PORT_UNI0, UniIn)		
	// Get the input status of the Universal Input 1		
	ret = GetUniversalIn(PORT_UNI1, UniIn)		

ClearUniversalIn

Call format	Declare Function ClearUniversalIn Lib "PL_Ioc.dll"		
	(ByVal Selector As Long) As Long		
Return value	Other than 0: Normal		
	0: Error		
Argument	Selector As Long	Designated port (pass by value)	
		PORT_UNI0	Universal Input 0
		PORT_UNI1	Universal Input 1
Processing	Clears the input status of the designated port (Universal Input 0 and Universal Input 1).		
Example	Dim ret As Long		
	// Clear the input status of Universal Input 0		
	ret = ClearUniversalIn(PORT_UNI0)		
	// Clear the input status of Universal Input 1		
	ret = ClearUniversalIn(PORT_UNI1)		

SetUniversalInMask

Call format	Declare Function SetUniversalInMask Lib "PL_Ioc.dll"		
	(ByVal Selector As Long, ByVal Mask As Long) As Long		
Return value	Other than 0: Normal		
	0: Error		
Argument	Selector As Long	Designated port (pass by value)	
		PORT_UNI0	Universal Input 0
		PORT_UNI1	Universal Input 1
	Mask As Long	Mask data (pass by value)	
		MASK_OFF	Clear mask
		MASK_ON	Mask
Processing	Sets the masking information of the designated ports (Universal Input 0 and Universal Input 1).		
Example	Dim ret As Long		
	// Release the masking for Universal Input 0		
	ret = SetUniversalInMask(PORT_UNI0, MASK_OFF)		
	// Mask Universal Input 1		
	ret = SetUniversalInMask(PORT_UNI1, MASK_ON)		

GetUniversalInMask

Call format	Declare Function GetUniversalInMask Lib "PL_Ioc.dll"		
	(ByVal Selector As Long, ByRef Mask As Long) As Long		
Return value	Other than 0: Normal		
	0: Error		
Argument	Selector As Long	Designated port (pass by value)	
		PORT_UNI0	Universal Input 0
		PORT_UNI1	Universal Input 1
	Mask As Long	Mask data (pass by reference)	
		MASK_OFF	Release mask
		MASK_ON	Mask
Processing	Gets the masking information of the subject ports (Universal Input 0 and Universal Input 1).		
Example	Dim ret As Long		
	Dim Mask As Long		
	// Get the masking information for Universal Input 0		
	ret = GetUniversalInMask(PORT_UNI0, Mask)		
	// Get the masking information for Universal Input 1		
	ret = GetUniversalInMask(PORT_UNI1, Mask)		

SetResetMask

Call format	Declare Function SetResetMask Lib "PL_Ioc.dll" (ByVal Mask As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	Mask As Long	Mask data (pass by value)	
		MASK_OFF	Release mask
		MASK_ON	Mask
Processing	Sets the reset mask.		

Example	Dim ret As Long
	// Releases the reset mask
	ret = SetResetMask(MASK_OFF)		

GetResetMask

Call format	Declare Function GetResetMask Lib "PL_Ioc.dll" (ByRef Mask As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	Mask As Long	Mask data (pass by reference)	
		MASK_OFF	Release mask
		MASK_ON	Mask
Processing	Gets the current reset mask information.		
Example	Dim ret As Long Dim Mask As Long ret = GetResetMask(Mask)		

SetIdeErr

Call format	Declare Function SetIdeErr Lib "PL_Ioc.dll" (ByVal IdeErr As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	IdeErr As Long	Error output data (pass by reference)	
		IDE_ERROR_OFF	Disables error output
		IDE_ERROR_ON	Enables error output
Processing	Uses software control to set the IDE error output.		

Example	Dim ret As Long
	// Set the system to disable the IDE error output
	ret = SetIdeErr(IDE_ERROR_OFF)		

GetIdeErrHard

Call format	Declare Function GetIdeErrHard Lib "PL_Ioc.dll" (ByVal Selector As Long, ByRef IdeErr As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	Selector As Long	Get parameter (pass by value)	
		IDE_ERROR_1	IDE_ERR1
		IDE_ERROR_2	IDE_ERR2
	IdeErr As Long	Error signal (pass by reference)	
		IDE_ERROR_OFF	Normal
		IDE_ERROR_ON	Error
Processing	Gets the current IDE error signal output by the hardware.		

Example	Dim ret As Long
	Dim IdeErr As Long
	// Gets the IDE ERR1 signal
	ret = GetIdeErrHard(IDE_ERROR_1, IdeErr)		

GetLightblowErr

Call format	Declare Function GetLightblowErr Lib "PL_Ioc.dll" (ByRef LightblowErr As Long) As Long		
Return value	Other than 0: Normal 0: Error		
Argument	LightblowErr As Long	Error data (pass by reference)	
		BACKLIGHT_OK	Normal
		BACKLIGHT_ERR	Error
Processing	Gets the current backlight error information.		

Example	Dim ret As Long
	Dim LightblowErr As Long
	// Gets the backlight error information.
	ret = GetLightblowErr(LightblowErr)		

Note: **This feature can be used only with PL-6920 Series units.**

GetEvent

Call format Declare Function GetEvent Lib "PL_Ioc.dll"
(ByVal Selector As Long, ByRef Event As Long) As Long

Return value Other than 0: Normal
0: Error

Argument Selector As Long Parameters (pass by value)

EVENT_VOLT_CPU	CPU core voltage
EVENT_VOLT_P33	+3.3V
EVENT_VOLT_P50	+5.0V
EVENT_VOLT_P12	+12V
EVENT_VOLT_M12	-12V
EVENT_VOLT_M50	-5.0V
MONITOR_VOLT_VIT	CPU Core voltage 2
EVENT_FAN_CPU	CPU FAN
EVENT_FAN_POWER	POWER FAN
EVENT_FAN_OPT	OPTION FAN
EVENT_TEMP_SYSTEM	SYSTEM temp.
EVENT_TEMP_CPU_OPT	CPU or OPTION temp.
EVENT_UNI_IN0	Universal Input 0
EVENT_UNI_IN1	Universal Input 1
EVENT_WDT_TIMEOUT	Watchdog Timeout

Event As Long Error event data (pass by reference)

ERROR_EVENT_OFF	No error event
ERROR_EVENT_ON	Error event

Processing Checks for the irregularities in the machine voltage, FAN, and temperature, Universal Input function (event) data, and WatchDog Timeout data.

Example

```
Dim ret As Long
Dim Event As Long
// Gets the error event data of the CPU core voltage
ret = GetEvent( EVENT_VOLT_CPU, Event )
```

ClearEvent

Call format	Declare Function ClearEvent Lib "PL_Ioc.dll" (ByVal Selector As Long) As Long
Return value	Other than 0: Normal 0: Error
Argument	Selector As Long Designated error event cancel parameters (pass by value)

EVENT_VOLT_CPU	CPU core voltage
EVENT_VOLT_P33	+3.3V
EVENT_VOLT_P50	+5.0V
EVENT_VOLT_P12	+12V
EVENT_VOLT_M12	-12V
EVENT_VOLT_M50	-5.0V
MONITOR_VOLT_VIT	CPU Core voltage 2
EVENT_FAN_CPU	CPU FAN
EVENT_FAN_POWER	POWER FAN
EVENT_FAN_OPT	OPTION FAN
EVENT_TEMP_SYSTEM	System temp.
EVENT_TEMP_CPU_OPT	CPU or OPTION temp.
EVENT_UNI_IN0	Universal Input0
EVENT_UNI_IN1	Universal Input1
EVENT_WDT_TIMEOUT	Watchdog Timeout

Processing Cancels the error event.

Example

```
Dim ret As Long
// Cancels the error event.
ret = ClearEvent( EVENT_VOLT_CPU )
```

StartInsideBuzzer

Call format	Declare Function StartInsideBuzzer Lib "PL_Ioc.dll" (ByVal hz As Long, ByVal ms As Long) As Long	
Return value	Other than 0: Normal 0: Error	
Argument	hz As Long	Buzzer frequency (pass by value)
	ms As Long	Buzzer sound period (pass by value)
Processing	Triggers the internal buzzer to sound at the specified frequency and for the specified period.	

Example1	Dim ret As Long
	Dim hz As Long
	Dim ms As Long
	// Sound the buzzer for 1 second at 600 Hz
	hz = 600
	ms = 1000
	ret = StartInsideBuzzer(hz, ms)	

This feature cannot be used with a PL running WindowsNT 4.0, Windows 2000 or Windows XP due to the use of Windows 95 and Windows 98 functions.

StopInsideBuzzer

Call format	Declare Function StopInsideBuzzer Lib "PL_Ioc.dll" () As Long
Return value	Other than 0: Normal 0: Error
Argument	None
Processing	Stops the internal buzzer.

Example	Dim ret As Long
	// Stops the internal Buzzer.
	ret = StopInsideBuzzer()

This feature cannot be used with a PL running WindowsNT 4.0, Windows 2000 or Windows XP due to the use of Windows 95 and Windows 98 functions.

ChkInsideBuzzer

Call format	Declare Function ChkInsideBuzzer Lib "PL_Ioc.dll"	
	(ByRef buff As Long) As Long	
Return value	Other than 0: Normal	
	0: Error	
Argument	BuzzerParam As Long	Buzzer status (pass by reference)
		BUZZER_ON Buzzer is ON
		BUZZER_OFF Buzzer is OFF
Processing	Checks for the ON/OFF status of the internal buzzer.	
Example	Dim ret As Long	
	Dim BuzzerParam As Long	
	// Checks the buzzer status	
	ret = ChkInsideBuzzer(BuzzerParam)	

Note: **This feature cannot be used with a PL running WindowsNT 4.0, Windows 2000 or Windows XP due to the use of Windows 95 and Windows 98 functions.**

GetWdtTimeout

Call format	Declare Function GetWdtTimeout Lib "L_Ioc.dll"	
	(ByRef Timebuf As Long) As Long	
Return value	Other than 0: Normal	
	0: Error	
Argument	Timebuf As Long	WDT status (pass by reference)
Processing	Gets the watchdog timeout status.	
Example	Dim ret As Long	
	Dim Timebuf As Long	
	// Gets the timeout status of the watchdog.	
	ret = GetWdtTimeout(Timebuf)	

ClearWdtTimeout

Call format	Declare Function ClearWdtTimeout Lib "PL_Ioc.dll () As Long
Return value	Other than 0: Normal
	0: Error
Argument	None
Processing	Clears the timeout status of the watchdog.
Example	Dim ret As Long
	// Clear the timeout status of the watchdog.
	ret = ClearWdtTimeout()

SetWarningDOUT

Call format	Declare Function SetWarningDOUT Lib "PL_Ioc.dll"		
	(ByVal WarningOut As Long) As Long		
Return value	Other than 0: Normal		
	0: Error		
Argument	WarningOut As Long	Output status (pass by value)	
		OUTPUT_OFF	Output OFF
		OUTPUT_ON	Output ON
Processing	Sets the warning status of the current setup item (DOUT).		
Example	Dim ret As Long		
	// Set the output status of DOUT to OFF.		
	ret = SetWarningDOUT(OUTPUT_OFF)		

GetWarningDOUT

Call format	Declare Function GetWarningDOUT Lib "PL_Ioc.dll"		
	(ByRef WarningOut As Long) As Long		
Return value	Other than 0: Normal		
	0: Error		
Argument	WarningOut As Long	Output status (pass by reference)	
		OUTPUT_OFF	Output OFF
		OUTPUT_ON	Output ON
Processing	Gets the warning status of the current setup item (DOUT).		
Example1	Dim ret As Long		
	Dim WarningOut As Long		
	ret = GetWarningDOUT(WarningOut)		

GetSmiDrvHandle

Call format	Declare Function GetSmiDrvHandle Lib "PL_Ioc.dll" () As Long
Return value	0: Normal
	1: Error
Argument	None
Processing	Gets the device driver handle to exchange information with the software mirroring device driver.
Example1	Dim ret As Long
	ret = GetSmiDrvHandle()

Note: **An error will occur if the software mirroring device driver is not running.**

CloseSmiDrvHandle

Call format Declare Function CloseSmiDrvHandle Lib "PL_Ioc.dll" () As Long

Return value Other than 0: Normal

0: Error

Argument None

Processing Destroys the handle acquired with the "GetSmiDrvHandle" function.

Example Dim ret As Long

// Destroys the handle.

ret = CloseSmiDrvHandle()

GetSmiAryStatus

Call format Declare Function GetSmiAryStatus Lib "PL_Ioc.dll"

(ByRef Status As Long) As Long

Return value Other than 0: Normal

0: Error

Argument Status As Long Software mirroring status (pass by reference)

ARYSTAT_GOOD	Normal
ARYSTAT_NOTEXIST	No output
ARYSTAT_UNCONFIG	Not configured
ARYSTAT_REBUILD	Being rebuilt
ARYSTAT_REDUCE	Being reduced
ARYSTAT_DEAD	Mirror status destroyed

Processing Gets the status of the software mirroring feature.

Example Dim ret As Long

Dim Status As Long

// Get the status of the software mirroring feature.

ret = GetSmiAryStatus(Status)

GetSmiDevStatus

Call format: Declare Function GetSmiDevStatus Lib "PL_Ioc.dll"
(ByVal Id As Long, ByRef Type As Long, ByRef Status As Long) As Long

Return value: Other than 0: Normal
0: Error

Argument:

Id As Long — Device ID (pass by value)
0 : Master HDD
1 : Slave HDD

Type As Long — Device type (pass by reference)

ATADEVICE	ATA type device
ATAPIDEVICE	CD-ROM
UNKNOWNDEVICE	Unknown device.
NODEVICE	No device

Status As Long — Device status (pass by reference)

DEVSTAT_GOOD	Normal
DEVSTAT_NOTEXIST	Not connected
DEVSTAT_BROKEN	Device failure

Processing: Gets the device status of the software mirroring feature.

Example:

```
Dim ret As Long
Dim Id As Long
Dim Type As Long
Dim Status As Long
// Gets the device status of the software mirroring feature.
Id = 0
ret = GetSmiDevStatus( Id, Type, Status )
```

SetWdtResetMask

Call format: Declare Function SetWdtResetMask Lib "PL_Ioc.dll"
(ByVal Mask As Long) As Long

Return value: Other than 0: Normal
0: Error

Argument:

Mask As Long — Mask data (pass by value)

MASK_OFF	Release mask
MASK_ON	Mask

Processing: Sets the H/W reset mask for the WDT timeout.

Example:

```
Dim ret As Long
// Releases the reset masking used when WDT timeout occurs.
ret = SetWdtResetMask( MASK_OFF )
```

GetWdtResetMask

Call format	Declare Function GetWdtResetMask Lib "PL_Ioc.dll"		
	(ByRef Mask As Long) As Long		
Return value	Other than 0: Normal		
	0: Error		
Argument	Mask As Long	Mask data (pass by reference)	
		MASK_OFF	Releases mask
		MASK_ON	Mask
Processing	Gets the WDT timeout's H/W reset mask data.		
Example	Dim ret As Long		
	Dim Mask As Long		
	// Get the H/W reset mask data for the WDT timeout.		
	ret = GetWdtResetMask(Mask)		

A-5 Backlight Control API-DLL

1 Operation Environment

The following information explains the Dynamic Link Libraries used by the backlight control feature on a PL-X920 Series unit.

API-DLLs provide the interface for applications to access the System Monitor/ RAS feature (System Monitor/RAS Device Driver). Applications can use DLLs to access the following feature.

1. Backlight Control ON/OFF

■ Compatible Operating Systems

The API-DLLs contained on the PL unit's CD-ROM are compatible with the following OS types.

- Mircrosoft Windows®95
- Microsoft Windows®98
- Microsoft WindowsNT®4.0
- Microsoft Windows®2000
- Microsoft Windows®XP

Each OS must use its corresponding Backlight Control Device.

■ Compatible Languages

- Microsoft Visual C
- Microsoft Visual C++
- Microsoft Visual Basic

◆ Required Files

The following files are required when using DLLs. Each language requires its own set of files.

• Visual C

File Name	Description
PL_BLIocif.h	Driver interface definition "include" file
PL_BLIoc.LIB	Library definition file
PL_BLIoc.dll	Dynamic link library file

• Visual C++

File Name	Description
PL_BLIocif.h	Driver interface definition "include" file
PL_BLIocall.h	CPL_BLIocall class definition "include" file
PL_BLIoctl.h	CPL_BLIoctl class definition "include" file
PL_BLIoc.LIB	Library definition file
PL_BLIoc.dll	Dynamic Link library file

* "#include header files should be "included" in the following order.

```
#include PL_BLIocif.h
#include PL_BLIoctl.h
```

PL_BLIocall.h is automatically included, and does not need to be directly designated.

• Visual Basic

File Name	Description
PL_BLIoc.bas	Driver interface definition file
PL_BLIoc.LIB	Library definition file
PL_BLIoc.dll	Dynamic link library file

■ Dynamic Link Library (DLL)

In order for an application to use PL_BLIoc.dll, it should be copied to the following folder.

OS	Location
Windows95/Windows98	C:\Windows\System
WindowsNT4.0/Windows2000	C:\Winnt\System32
WindowsXP	C:\Windows\System32

2 Class Contents

■ CPL_BLIoctl Class

This class is used to set the parameters for device driver access using CPL_BLIoctl class.

Key Word	Type	Variable Name	Description
public	HANDLE	m_Drvhandle	Device driver handle

■ CPL_BLIocal Class

This uses the parameters set in CPL_BLIoctl, and calls up DeviceIoControl (Driver Access function).

However, since this class succeeds CPL_BLIoctl, it cannot be used directly.

Key Word	Type	Variable Name	Description
public	HANDLE	m_h	Device driver handle
public	LONG	m_long	Control code for action to perform
public	void *	m_ibp	Input data buffer address
public	ULONG	m_ibsize	Input data buffer size
public	void *	m_obp	Output data buffer address
public	ULONG	m_obsize	Output data buffer size
public	DWORD	m_retsize	Address for actual no. of output bytes
public	LPOVERLAPPED	m_ovlp	Address of overlap design

3 Visual C Functions

Function Name	Description
InitBLIoctl	Creates the CPL_BLIoctl object
EndBLIoctl	Destroys the CPL_BLIoctl object
GetBLDrvHandle	Gets the driver handle
GetBLDrvVersion	Gets the driver version
SetBLControl	Sets the backlight control values
GetBLControl	Gets the backlight control settings

4 Visual C Function Specifications (Details)

InitBLIoctl

Call Format void WINAPI InitBLIoctl(void)

Return Value None

Arguments None

Processing Creates a CPL_BLIoctl object. The object once created is not destroyed until the EndBLIoctl function is called.

Example InitBLIoctl();

EndBLIoctl

Call Format void WINAPI EndBLIoctl(void)

Return Value None

Arguments None

Processing Destroys the object created using the InitBLIoctl function.

Example EndBLIoctl();

GetBLDrvHandle

Call Format int WINAPI GetBLDrvHandle(HANDLE * pHndl)

Return Value 0: Normal
1: Error

Arguments (I/O) HANDLE *pHndl Pointer to the device driver handle

Processing Gets the device driver handle to communicate with the device driver.

Example

```
int       ret;
HANDLE    hndl;
ret = GetBLDrvHandle( &hndl );
```

Note: **An error will occur if the Backlight Control Device Driver is not running.**

GetBLDrvVersion

Call Format	BOOL WINAPI GetBLDrvVersion (int *pMajor, int *pMinor)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I/O) int *pMajor Pointer to version information (Major, 0 to 99). (I/O)int *pMinor Pointer to version information (Minor, 0 to 99).
Processing	Gets the driver's version information.

Example	BOOL ret;
	int Major, Minor;
	ret = GetBLDrvVersion(&Major, &Minor);

Note:

If the version is 1.10, then you will get
Major: 1 (decimal)
Minor: 10 (decimal).

SetBLControl

Call Format	BOOL WINAPI SetBLControl (int BLFlag)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I) int BLF flag Setting Parameters BACKLIGHT_OFF Backlight OFF BACKLIGHT_ON Backlight ON
Processing	Sets the backlight ON/OFF.

Example	BOOL ret;
	// Turns the backlight control ON.
	ret = SetBLControl(BACKLIGHT_ON);

GetBLControl

Call Format	BOOL WINAPI GetBLControl (int pBLFlag)
Return Value	TRUE: Normal FALSE: Error
Arguments	(I/O) int *pBLFlag Pointer to backlight condition BACKLIGHT_OFF Backlight OFF BACKLIGHT_ON Backlight ON
Processing	Gets the backlight control (settings) condition.

Example	BOOL ret;
	int BLFlag;
	// Gets the backlight control (settings) condition.
	ret = GetBLControl(&BLFlag);

5 Visual C++ Functions

Function Name	Description
GetBLDrvHandle	Gets the driver handle
GetBLDrvVersion	Gets the driver version
SetBLControl	Sets the backlight control values
GetBLControl	Gets the backlight control settings

6 Visual C++ Function Specifications (Details)

GetBLDrvHandle

Call Format int GetBLDrvHandle(void) or int GetBLDrvHandle(HANDLE *pHndl)

Return Value 0: Normal
1: Error

Arguments None

Processing Gets the device driver handle to communicate with the device driver. The handle obtained is stored in the member variable m_handle.

Example 1
```
CPL_BLIoctl      m_BLIoc;
m_BLIoc.GetBLDrvHandle();
```

Example 2
```
int      ret;
HANDLE   hndl;
ret = ::GetBLDrvHandle( &hndl );
```

Note: **An error will occur if the Backlight Control Device Driver is not running.**

GetBLDrvVersion

Call Format BOOL GetBLDrvVersion(int *pMajor, int *pMinor)

Return Value TRUE: Normal
FALSE: Error

Arguments (I/O) int *pMajor Pointer to version information (Major, 0 to 99).
(I/O)int *pMinor Pointer to version information (Minor, 0 to 99).

Processing Gets the driver's version information.

Example 1
```
CPL_BLIoctl m_BLIoc;
BOOL      ret;
int       Major, Minor;
ret = m_BLIoc.GetBLDrvVersion( &Major, &Minor );
```

Example 2
```
BOOL      ret;
int       Major, Minor;
ret = ::GetBLDrvVersion( &Major, &Minor );
```

Note:

If the version is 1.10, then you will get
Major: 1 (decimal)
Minor: 10 (decimal).

SetBLControl

Call Format BOOL SetBLControl (int BLFlag)

Return Value TRUE: Normal
FALSE: Error

Arguments (I) int BLFlag Setting Parameters

BACKLIGHT_OFF	Backlight OFF
BACKLIGHT_ON	Backlight ON

Processing Sets the backlight ON/OFF.

Example 1
```
CPL_BLIoc m_BLIoc;
BOOL      ret;
// Turns the backlight control ON.
ret = m_BLIoc.SetBLControl( BACKLIGHT_ON )
```

Example 2
```
BOOL      ret;
// Turns the backlight control ON.
ret = ::SetBLControl( BACKLIGHT_ON );
```

GetBLControl

Call Format BOOL GetBLControl (int *pBLFlag)

Return Value TRUE: Normal
FALSE: Error

Arguments (I/O) int *pBLFlag Pointer to backlight condition
BACKLIGHT_OFF Backlight OFF
BACKLIGHT_ON Backlight ON

Processing Gets the backlight control (settings) status.

Example 1

```
CPL_BLIoc m_BLIoc;
BOOL      ret;
int       BLFlag;
// Gets the backlight control condition.
ret = m_BLIoc.GetBLControl( &BLFlag );
```

Example 2

```
BOOL      ret;
int       BLFlag;
// Turns the backlight control ON.
ret = ::GetBLControl( &BLFlag );
```

7 Visual Basic Functions

Function Name	Description
InitBLIoctl	Creates the CPL_Ioctl object
EndBLIoctl	Destroys the CPL_Ioctl object
GetBLDrvHandle	Gets the driver handle
GetBLDrvVersion	Gets the driver version
SetBLControl	Sets the backlight control values
GetBLControl	Gets the backlight control settings

8 Visual Basic Function Specifications (Details)

InitBLIoctl

Call Format Declare Sub InitBLIoctl Lib "PL_BLIoc.dll" ()

Return Value None

Arguments None

Processing Creates a CPL_BLIoctl object. The object once created is not destroyed until the EndBLIoctl function is called.

Example Call InitBLIoctl

EndBLIoctl

Call Format Declare Sub EndBLIoctl Lib "PL_BLIoc.dll" ()

Return Value None

Arguments None

Processing Destroys the object created using the InitBLIoctl function.

Example Call EndBLIoctl

GetBLDrvHandle

Call Format Declare Function GetBLDrvHandle Lib "PL_BLIoc.dll" (ByRef hndl As Long) As Long

Return Value 0: Normal
1: Error

Arguments hndl As Long Pointer to device driver handle (pass by reference)

Processing Gets the device driver handle to communicate with the device driver.

Example

```
Dim ret  As Long
Dim hndl As Long
ret = GetBLDrvHandle( hndl )
```

Note: **An error will occur if the Backlight Control Device Driver is not running.**

GetBLDrvVersion

Call Format　Declare Function GetBLDrvVersion Lib "PL_BLIoc.dll"
(ByRef Major As Long, ByRef Minor As Long) As Long

Return Value　Other than 0: Normal
0: Error

Arguments　Major As Long　Pointer to version information (Major, 0 to 99) (pass by reference)
Minor As Long　Pointer to version information (Minor, 0 to 99) (pass by reference)

Processing　Gets the driver's version information.

Example

```
Dim ret As Long
Dim Major As Long
Dim Minor As Long
ret = GetBLDrvVersion(Major, Minor)
```

Note:

If the version is 1.10, then you will get
Major: 1 (decimal)
Minor: 10 (decimal).

SetBLControl

Call Format　Declare Function SetBLControl Lib "PL_BLIoc.dll"
(ByVal BLFlag As Long) As Long

Return Value　Other than 0: Normal
0: Error

Arguments　BLFlag As Long　Setting Parameters (pass by value)

BACKLIGHT_OFF	Backlight OFF
BACKLIGHT_ON	Backlight ON

Processing　Sets the backlight ON/OFF.

Example

```
Dim ret As Long;
// Turns the backlight control ON.
ret = SetBLControl( BACKLIGHT_ON )
```

GetBLControl

Call Format Declare Function GetBLControl Lib "PL_BLIoc.dll"
(ByRef BLFlag As Long) As Long

Return Value Other than 0: Normal
0: Error

Arguments BLFlag As Long Pointer to backlight status (pass by reference)
BACKLIGHT_OFF Backlight OFF
BACKLIGHT_ON Backlight ON

Processing Gets the backlight control (settings) status.

Example
```
Dim ret As Long
Dim BLFlag As Long
// Gets the backlight control (settings) condition.
ret = GetBLControl( BLFlag );
```

A-6 Consent Agreement

■ IN-fINITY soft Keyclick32

Digital Electronics Corporation shall be in no case liable for any loss or damages to you or any third party whatsoever arising from any faulty performance of IN-fINITY soft, nor shall be responsible to correct such error of the software program in question.