# Series 90™ to PACSystems™

## APPLICATION CONVERSION GUIDE

EMERSON™

# Contents

## Warning, Caution Notes as Used in this Publication

**Warning**

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury to exist in this equipment or may be associated with its use.
In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

**Caution**

Caution notices are used where equipment might be damaged if care is not taken.

***Notes:*** *Notes merely call attention to information that is especially significant to understanding and operating the equipment.*

These instructions do not purport to cover all details or variations in equipment, nor to provide for every possible contingency to be met during installation, operation, and maintenance. The information is supplied for informational purposes only, and Emerson makes no warranty as to the accuracy of the information included herein. Changes, modifications, and/or improvements to equipment and specifications are made periodically and these changes may or may not be reflected herein. It is understood that Emerson may make changes, modifications, or improvements to the equipment referenced herein or to the document itself at any time. This document is intended for trained personnel familiar with the Emerson products referenced herein.

Emerson may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not provide any license whatsoever to any of these patents.

Emerson provides the following document and the information included therein as-is and without warranty of any kind, expressed or implied, including but not limited to any implied statutory warranty of merchantability or fitness for particular purpose.

# Chapter 1: Introduction

PACSystems controllers incorporate the functional features of the Series 90 PLC family with the Ethernet Global Data (EGD) capabilities of the Series 90-30 PLC family and improved Ethernet communications. PACSystems provides many enhancements compared to the Series 90 PLCs, although some Series 90 functionality is not supported in the current version.

## Additional Information

### PACSystems Manuals

PACSystems CPU Reference Manual, GFK-2222

TCP/IP Ethernet Communications for PACSystems, GFK-2224

Station Manager for PACSystems, GFK-2225

PACSystems C Toolkit User's Guide, GFK-2259

PACSystems Hot Standby CPU Redundancy User's Manual, GFK-2308

PAC Machine Edition Logic Developer Getting Started, GFK-1918

PACSystems RX3i Hardware and Installation Manual, GFK-2314

PACSystems RX7i User's Guide to Integration of VME Modules, GFK-2235

### Series 90 Manuals

Series 90 Programmable Coprocessor Module and Support Software, GFK-0255

Series 90 PLC Serial Communications Driver User's Manual, GFK-0582

C Programmer's Toolkit for Series 90 PLCs User's Manual, GFK-0646

Installation Requirements for Conformance to Standards, GFK-1179

TCP/IP Ethernet Communications for the Series 90 PLC Station Manager Manual, GFK 1186

Series 90-70 CPU Instruction Set Reference Manual, GFK-0265

Series 90-30 Genius Bus Controller, GFK-1034

Series 90-30 System Manual, GFK-1411

Genius I/O System User's Manual, GEK-90486-1

Genius I/O Analog and Discrete Blocks User's Manual, GEK-90486-2

In addition to these manuals, the Important Product Information (IPI) documents provided with individual modules describe supported features and product revisions. The most recent PACSystems documentation is available on the Support website.

# Chapter 2: PACSystems – Series 90 Comparison

This section summarizes differences in features and operation between the two control systems.

## 2.1 CPU Operation

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| CPUs allowed in any rack 0 slot. | Not supported. | Not supported. | Supported by RX3i (requires two slots). Not supported by RX7i. |
| Ladder Logic Execution See “Row- vs. Column-major LD Execution” on page 3 | Column major | Row major | Row major |
| Fault clearing | Clear individual faults. | Clear individual faults. | Clear the controller fault table or I/O fault table. |
| Reset of IO Module fault | Generated on store or clear of HWC. | Not generated on store or clear of HWC. | Generated on store or clear of HWC. |
| %S0020 status bit | Not supported. | Set ON when a relational function using REAL data executes successfully. Cleared when either input is NaN (Not a Number). | Not supported. In LD logic, the OK output can be used to indicate the function has executed successfully. |
| Access Privileges | Levels 0 through 4. | Levels 1 through 4 | Levels 1 through 4 |
| Stack overflow | Transitions to Stop/Halt mode when it detects a stack overflow. | Transitions to Stop/Fault mode when it detects a stack overflow. | If there is not enough stack space left to support a given block call, an “Application Stack Overflow” fault is logged. In these circumstances, the CPU cannot execute the block. Instead, it sets the block’s Boolean outputs to FALSE, and resumes execution at the point after the block call instruction. |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Program name | Not used | Not used | Read-only LD program name, LDPROG1, used by legacy drivers in Plant Edition software. |
| Low battery indication | Supported via the %SA0011 system status bit and the PLC fault table. | Supported via the %SA0011 system status bit and the PLC fault table. | Supported for some combinations of CPU model and battery. For details, refer to the battery documentation. |

# 2.2 Logic Operation

## 2.2.1 Row- vs. Column-major LD Execution

Series 90-70 PLCs use column major execution: they execute a rung of LD logic by going from top to bottom, left to right, through each column in the rung. Series 90-30, VersaMax, and PACSystems use row major execution: they execute an LD rung by tracing paths from left to right and top to bottom. Differences in execution order result from branching, or the divergence and/or convergence of power flow within a rung, which is not allowed in Series 90-30. Therefore, the following examples do not apply to Series 90-30.

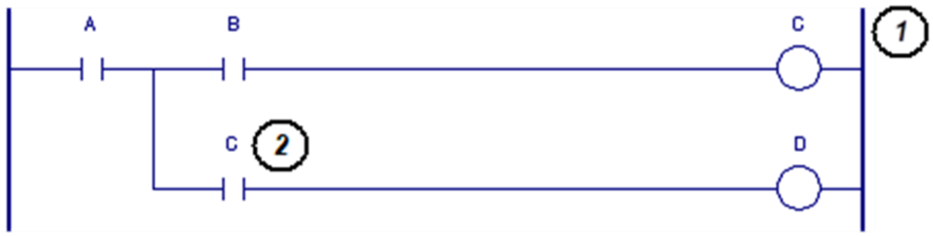
**Note:** *The conversion of a Series 90-70 target to a PACSystems target does not rewrite the logic from column major execution to row major execution. Rungs that may execute differently because of the column/row major difference are reported in the target conversion report, but it is not guaranteed that every execution difference will be detected and reported.*

### Example 1

In this example, the order in which the contact and coil that reference the variable C are executed differs between column-major and row-major execution.

- In row-major, the coil (1) sets the value of C before the contact (2) is evaluated.
- In column-major, the C contact (2) is evaluated before the coil (1) is executed.

**Figure 1**

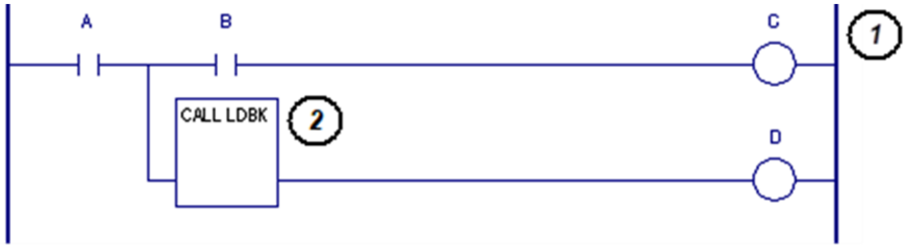


## Example 2

In this example, the problem is less obvious. If the variable C is used as an input and/or output inside LDBK or any block that LDBK calls, the difference in row-major versus column-major execution within the called blocks may affect the execution of coil C.

- In row-major execution, the C coil (1) is executed before the Call (2) is executed.
- In column-major execution, the Call (2) is executed before the C coil (1) is executed.

**Figure 2**

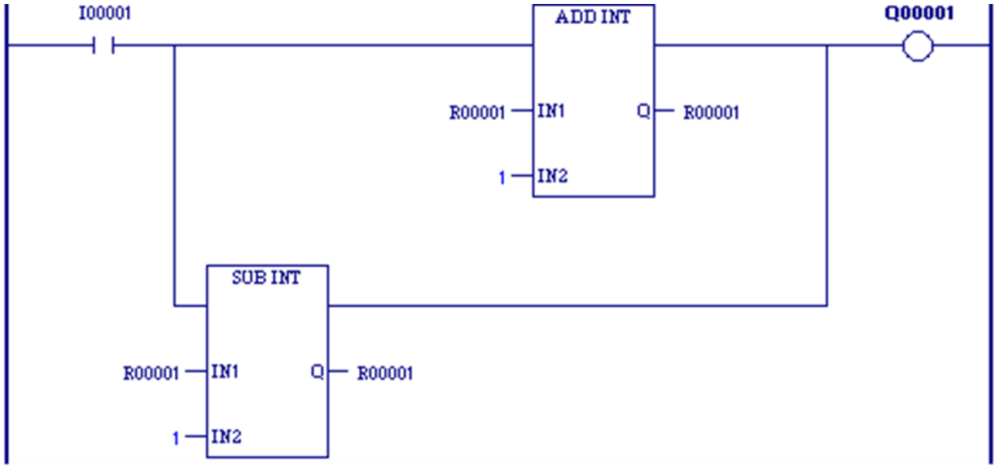


## Example 3

In the following example:

- Using column-major execution (Series 90-70), the SUB_INT always executes before the ADD_INT.
- Using row-major execution (PACSystems ), the ADD_INT executes before the SUB_INT.

**Figure 3**



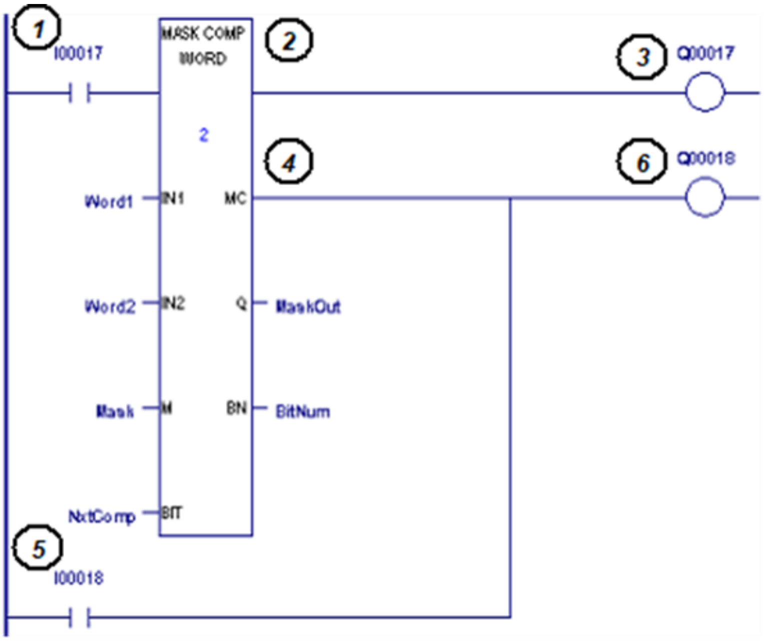
## Example 4

In this example, even though the order of execution is different, the outcome is the same for both types of execution.

## Row Major Execution (PACSystems)

The external input I00017 (1) is evaluated first. If I00017 is set to 1, the Masked Compare function (2) is executed, passing power to Q00017 (3). If there is a mis compare, the output MC (4) is set to 1. I00018 (5) is then evaluated. The state of Q00018 (6) is set to the OR of the MC output and I00018.

**Figure 4**

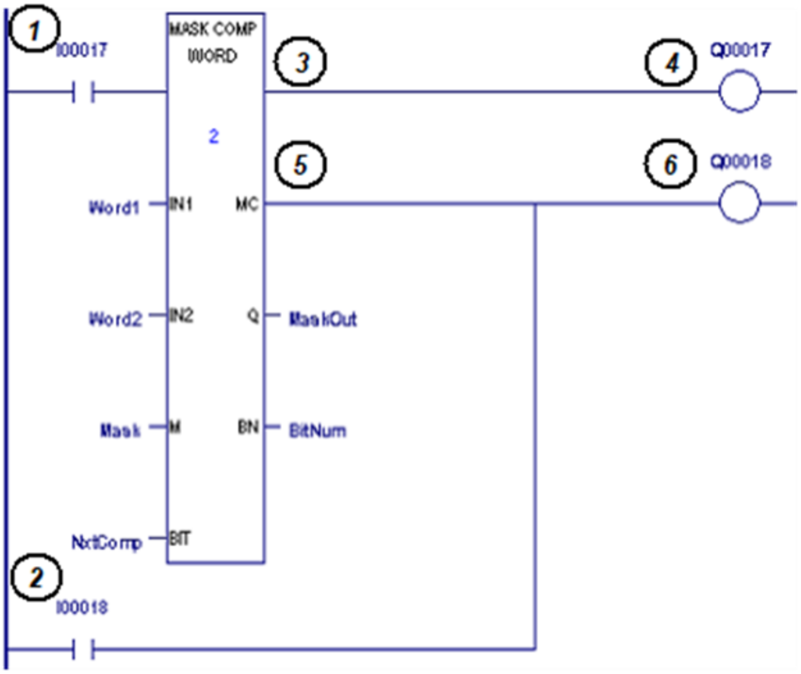


## Column Major Execution (Series 90-70)

The external input I00017 (1) is evaluated first, followed by external input I00018 (2). If I00017, is set to 1, the Masked Compare function (3) is executed, passing power to Q00017 (4). If there is a mis compare, the output MC (5) is set to 1.

The state of Q00018 (6) is set to the OR of the MC output and I00018.

Figure 5



## 2.2.2 Maximum Number of Blocks

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Maximum number of blocks | 256, including _MAIN | 65, including _MAIN | 512, including _MAIN |

## 2.2.3 User Programs

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| C Standalone Programs | Supported | Not supported | Not supported |
| LD Program | Supported | Supported | Supported |
| Structured Text | Not supported. | Not supported. | For availability on a given CPU version, refer to the IPI document provided with that CPU. |
| Program Scheduling | Five modes are supported. | Only the Ordered mode is supported. | Only the Ordered mode is supported. |
| Interrupt programs | Supported | Not supported | Not supported |
| Function Blocks | Supported | Not supported. | For availability on a given CPU version, refer to the IPI document provided with that CPU. |
| Sequential Function Chart programming | Supported | Supported | Not supported |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Synchronous Scan Sets | Supported | Not supported | Not supported |
| FIP, Micro cycle mode and periodic programs | Supported | Not supported | Not supported |
| Multiple programs per folder | Supported (up to 16) | Not supported | Not supported |
| C Debugger | Supported | Not supported | Not supported |
| State Logic | Supported | Supported | Not supported |

## 2.2.4 Stack Size

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Stack Size | Valid range: 1 through 64 KB<br>Default: 20 KB (LD Program) | Not configurable. | Valid range: 8 through 320 KB, in increments of 8 KB<br>Default: 64 KB |

## 2.2.5 C Blocks

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| C toolkit used to develop | GFK-0646 | GFK-0646 | GFK-2259 |
| Bits | 16-bit blocks | 16-bit blocks | 32-bit blocks |
| Compiled block extension | .exe or .sta | .exe | .gefelf |
| Maximum size | 64,000 bytes | 81,920 bytes | Limited only by available user memory.<br>**Note:** *By default, for any C block, if its size is greater than 256 KB, a validation error takes place. This may be prevented by right-clicking the C block, choosing Properties and, in the Inspector, setting the Check Size Limits property to False. This, however, introduces the risk of memory fragmentation.* |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Importing | Cannot import blocks developed for PACSystems or Series 90-30 | Cannot import blocks developed for PACSystems or Series 90-70 | Cannot import blocks developed for Series 90-70 or Series 90-30. They must be recompiled with a .gefelf extension. |
| Name property | Up to 7 characters long | Up to 7 characters long | Up to 31 characters long |
| Check Size Limits | Not supported. | Not supported. | Supported. |
| Third-party VME interrupts | Supported. | Not supported. | Replaced with Module interrupts |
| Module interrupts | Not supported | Not supported. | Selected from the list of Triggers when you schedule the block. First, configure the module in the Hardware Configuration; then set a module's interrupt ID in the Interrupts tab of the Parameter editor. |
| Number of defined parameters | 0 through 7 input/output pairs. Must have as many inputs as outputs. | Not supported. | Input: 0 through 63.<br>Output: 1 through 64.<br>Does not need to have as many inputs as outputs. |
| Optional parameters optional in the CALL to the C block | Not supported. You must supply a value for every defined parameter. | Not supported. | Machine Edition enables you to leave the value of any parameter blank. It is your responsibility to ensure the C block uses an acceptable default value and no run time error occurs. |
| BYTE data type | Supported | Not supported. | Refer to the *IPI* document for a specific CPU firmware version. |
| NWORD data type | Supported. | Not supported. | Not required. Can use WORD instead. |
| Data flow | Not supported. | Not supported. | Supported |
| Indirect references | Not supported. | Not supported. | Supported |
| Bit references in non-discrete memory | Not supported | Not supported. | Supported. Must be byte-aligned. |

# 2.3 Blocks

A block can be of type Block, Parameterized Block, or Function Block.

## 2.3.1 Blocks

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Maximum size | 32 KB | 16 KB | 128 KB |
| Name property | Up to 7 characters long | Up to 7 characters long | Up to 31 characters long |
| Extra Local Words allocated for %P or %L memory per program block | Not configurable | Not supported | Configurable. |
| Third-party VME interrupts | Supported | Not supported | Replaced with Module interrupts |
| Module interrupts | Not supported | Not supported | Selected from a list when scheduling the block. First you need to configure the module and the interrupt in the Hardware Configuration. |
| Bit references in non-discrete memory | Not supported | Not supported | Supported |

## 2.3.2 Parameterized Blocks

Referred to as parameterized subroutine blocks (PSBs) in Series 90-70. Not supported in Series 90-30.

| Category | Series 90-70 | PACSystems |
|---|---|---|
| BYTE data type | Supported | Refer to the *IPI* document for a specific CPU firmware version. |
| NWORD data type | Supported | Not required. Can use WORD instead. |
| Y0 parameter | Cannot be used in the logic of a zero-parameter parameterized block. Y0 is assumed to be set to True. | Can be used in the logic of any parameterized block, including one with no parameters. Can also be used in a program block, except for _MAIN. |
| BOOL parameters | Flow not supported into a BOOL input parameter whose length is greater than 1, or out of a BOOL output parameter whose length is greater than 1. Only power flow supported; not data flow. | Data and power flow supported. |

| Category | Series 90-70 | PACSystems |
|---|---|---|
| | Constants are not supported. | Constants are supported for input parameters. |
| 32-bit parameters (DINT, DWORD, and REAL) | Constants are not supported. | Constants are supported for input parameters. |
| Bit references in non-discrete memory | Not supported | Supported |
| Block name | Maximum of 7 characters | Up to 31 characters long. |
| Parameters | Maximum of 7 inputs and 7 outputs. (8 including Y0 output) | Maximum of 63 inputs and 63 outputs (64 including Y0 output). Refer to the *IPI* document for a specific CPU firmware version. |

### 2.3.3 Function Blocks

PACSystems allows the use of Function Blocks, which are user-defined logic blocks that have parameters and instance data. Series 90 PLCs do not support this feature.

## 2.4 Functions Introduced with PACSystems

The following functions were introduced with PACSystems and may not be supported in all Series 90 and VersaMax PLC product families.

### 2.4.1 BUS instructions

- BUS_RD_BYTE. Replaces the Series 90-70 VME_RD_BYTE function.
- BUS_RD_DWORD
- BUS_RD_WORD. Replaces the Series 90-70 VME_RD_WORD function.

*Note: The BUS_RD_ instructions replace the Series 90-70 VME_CFG_READ function.*

- BUS_RMW_BYTE. Replaces the Series 90-70 VME_RMW_BYTE function.
- BUS_RMW_DWORD
- BUS_RMW_WORD. Replaces the Series 90-70 VME_RMW_WORD function.
- BUS_TS_BYTE. Replaces the Series 90-70 VME_TS_BYTE function.
- BUS_TS_WORD. Replaces the Series 90-70 VME_TS_WORD function.
- BUS_WRT_BYTE. Replaces the Series 90-70 VME_WRT_BYTE function.
- BUS_WRT_DWORD
- BUS_WRT_WORD. Replaces the Series 90-70 VME_WRT_WORD function.

*Note: The BUS_WRT_ instructions replace the Series 90-70 VME_CFG_WRITE function.*

## 2.4.2 New Transitional Coils and Contacts

- NTCOIL
- NTCON
- PTCOIL
- PTCON

The status of the new transitional contacts PTCON and NTCON is determined by the value the associated BOOL variable had the last time the contact was executed. The status of the existing POSCON and NEGCON transitional contacts is determined by the last write to the BOOL variable associated with the contact.

## 2.4.3 Service Requests

The PACSystems SVC_REQ function supports the following services:

- #50: Read Elapsed Time Clock (Two DWORDs)
- #51: Read Sweep Time from Beginning of Sweep (DWORD)

# 2.5 Function Differences Resulting from Features Introduced with PACSystems

Most function differences between PACSystems and other PLC families, such as Series 90 and VersaMax, are due to the PACSystems CPUs' support of the following new features:

- Symbolic variables
- Bit addressing in non-discrete memory
- BOOL arrays of enough length can replace operands of other data types.

The following table lists other differences between Series 90-70 and PACSystems with regards to individual built-in functions.

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| ARRAY_MOVE_ (all mnemonics) | The SR parameter does not support data flow. | Same as Series 90-70. | The SR parameter supports data flow. |
| ARRAY_RANGE_ (all mnemonics) | The LL, UL, IN, and Q parameters do not support data flow. | Function not supported. | The LL, UL, IN, and Q parameters support data flow.<br>When an invalid (reference out of range) operand occurs and length is equal to one the result is set to false |
| BIT_SEQ | The N (STEP) parameter does not support data flow. | Same as Series 90-70. | The N parameter supports data flow. |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Bit Operation Functions | Overlapping input and output reference address ranges in multi-word functions may produce unexpected results. | Same as Series 90-70. | When using overlapping inputs and outputs, the PACSystems performs the operation on the data when the function is invoked, not the data output from earlier executions. |
| COMM_REQ | Supports WAIT mode COMM_REQs. | Same as 90-70. | Does not support WAIT mode COMM_REQs. |
| | A smart module can update the status word if you have level 3 or level 4 security access. | Same as 90-70. | Level 2 is enough, except when a Series 90-70 GBC is involved. In this case, level 3 or 4 is required. |
| | Can use COMM_REQs with serial and Ethernet ports. | Same as 90-70. | Using COMM_REQs for communication with serial and Ethernet ports supported. Refer to the IPI document for a specific CPU firmware version. |
| | System IDs address modules located in double-width slots. | Same as 90-70. | System IDs address modules located in single-width slots (RX7i). |
| DATA_INIT_ASCII<br>DATA_INIT_COMM<br>DATA_INIT_DINT<br>DATA_INIT_DWORD<br>DATA_INIT_INT<br>DATA_INIT_REAL<br>DATA_INIT_UINT<br>DATA_INIT_WORD<br>DATA_INIT_DLAN | The Q output parameter does not support indirect references or data flow. | Function not supported. | The Q output parameter supports indirect references and data flow. |
| END | Not supported | Supported | Not supported |
| Enhanced DO_IO | Not supported | Supported | Not supported |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| PID | Elapsed time computed in 10ms units. | Elapsed time computed in 10ms units. | The PID algorithm has been modified to improve some error cases; therefore, PID functions differently on PACSystems.<br>Elapsed time is computed in 100µs units instead of 10ms units. This smooth the output characteristic, eliminating periodic adjustments that occurred when the remainder accumulated to 10ms.<br>See the PACSystems CPU Reference Manual, GFK-2222 for additional details. |
| RANGE_ (all mnemonics) | The L1, L2, and IN parameters do not support data flow. | Same as Series 90-70. | The L1, L2, and IN parameters support data flow. |
| SVC_REQ 5<br>Change Background Task Window Mode and Timer Value | Supported. | Not supported | Supported |
| SVC_REQ 6<br>Change/Read Number of Words to Checksum | Number of words to be check summed is rounded to a multiple of 8. (Same as PACSystems.) | Number of words to be check summed must be in the range 0 to 32. | Number of words to be check summed is rounded to a multiple of 8. |
| SVC_REQ 13<br>Shut down (stop CPU) | Parameter block is ignored. (You must specify a dummy parameter, which SVC_REQ 13 does not use.) Stops at the end of the next scan.<br>To mimic this behavior in PACSystems version 2.00 or later, use a value of –1 in the SVC_REQ parameter block and set Number of Last Scans in HWC to 0. | Parameter block is ignored. Always runs one scan before stopping CPU.<br>To mimic this behavior in PACSystems version 2.00 or later, use a value of –1 in the SVC_REQ parameter block and set Number of Last Scans in HWC to 1. | For CPUs with firmware version 2.00 or later, 0 to 5 scans are allowed. A value of –1 causes the Number of Last Scans specified in HWC to be used.<br>For CPUs with firmware versions earlier than 2.00, the value in the parameter block must be 0. |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| SVC_REQ 15 Read Last-Logged Fault Table Entry | Supports Read Extended Controller Fault Table (80h) and Read Extended I/O Fault Table (81h). | Does not support Read Extended Controller Fault Table (80h) or Read Extended I/O Fault Table (81h). | Supports Read Extended Controller Fault Table (80h) or Read Extended I/O Fault Table (81h). |
| SVC_REQ 23 Read Master Checksum | Returns a 15-word output block. (Same as PACSystems.) | Returns an 11-word output block. Operation is different from Series 90-70 and PACSystems. Refer to the Series 90-30 CPU Reference Manual, GFK-0467. | Returns a 15-word output block. |
| SVC_REQ 24 Reset Smart Module | Not supported. | Supported. (Same as PACSystems.) | Supported. |
| SVC_REQ 26/30 Interrogate I/O | SVC_REQ 26 is Role Switch (redundancy) service request. Interrogate I/O function is supported via fault locating references. (Same as PACSystems.) | Service Requests 26 and 30 are identical. Refer to the Series 90-30 CPU Reference Manual, GFK-0467. | SVC_REQ 26 is Role Switch (redundancy) service request.<br>Interrogate I/O function is supported via fault locating references. |
| SVC_REQ 36 Read from/Write to Bulk Memory Area | Supported. | Not supported. | Not supported. This service request was the way to access BMA on a 90-70 CPU. Since BMA can be accessed as %W memory in a PACSystems RX7i, the service request is no longer needed. |
| SVC_REQ 39 ESCM Port Status | Supported. | Not supported. | Not supported. Specific to the Series 90-70, which has an ESCM to manage its serial communications. |
| SVC_REQ 44 Logic Driven Dynamic Ethernet Global Data. | Supported | Not supported. | Not supported. |
| SVC_REQ 45 Skip Next I/O Scan | Not supported | Supported | Refer to the IPI document for a specific CPU firmware version. |
| SVC_REQ 46 Fast Backplane Status Access | Not supported. (Same as PACSystems.) | Supported. Refer to the Series 90-30 CPU Reference | Not supported. |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| SVC_REQ 48 Reboot After Fatal Fault Auto Reset |  | Manual, GFK-0467 for details. |  |
| SVC_REQ 49 Auto Reset Statistics |  |  |  |
| Timed Contacts | Timed contact %S references are reset on a Stop to Run transition.<br>Forces a transition if sweep is longer than half the timed contact clock cycle.<br>Not updated in Stop mode. | Timed contact %S references are reset on a Stop to Run transition.<br>Forces a transition if sweep is longer than half the timed contact clock cycle.<br>Not updated in Stop mode. | Determines the state of each timed contact reference based on a free running timer that has no relationship to the start of each sweep. If the sweep time remains in phase with the timed contact clock, the contact will always appear to be in the same state. For example, if the CPU is in constant sweep mode with a sweep time setting of 100ms, the T_10MS and the T_100MS bits will never toggle. (Same as Series 90-30.)<br>Updated in Stop mode. |
| Timer functions | A timer function starts accumulating time when the block is called. | Operates differently from timers in PACSystems and Series 90-70. | Same as Series 90-70. A timer function starts accumulating time on a Stop to Run transition.<br>To prevent a timer from accumulating time, use the block fst_exe contact to reset the timers in the first sweep the block is called. |
| OFDT function blocks | If the current value (CV) is greater than the preset value (PV) for one sweep, the CV is clamped at the PV until the function block is reset. | If the current value (CV) is greater than the preset value (PV) for one sweep, the CV remains at the higher value. It is not clamped at the PV. | Same as Series 90-70. |
| VME_ Functions | Supported. | Not supported. | Not supported. Replaced by BUS_ functions. |

## 2.5.1 Floating Point Functions

PACSystems CPUs may return slightly different values for Not A Number (NaN) as compared to Series 90-70, Series 90-30, and VersaMax CPUs. For details on NaN values returned by floating point functions, refer to the PACSystems CPU Reference Manual, GFK-2222.

Floating point functions handle the NaN propagation and cases differently. PACSystems CPUs allow the floating-point hardware to handle the NaN cases, instead of treating these instructions as special cases. Impacts: ADD, SUB, MUL, DIV, SIN, COS, TAN, ASIN, ACOS, ATAN, LOG, LN, EXP, EXPT, DEG_TO_RAD, RAD_TO_DEG, ABS_REAL, SQRT_REAL functions.

PACSystems calculates TRIG functions for a larger range of values than the other PLCs. As a result, some values that previously returned NAN return a correct value.

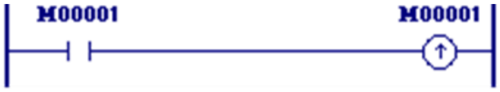
PACSystems returns values for some unusual cases that were previously returned as NaN. Specifically, 00 returns 1 instead of NaN.

## 2.5.2 Legacy Transitional Coils and Contacts

When implementing a transition (one-shot) coil, where there is a contact and a transition coil in the same rung, do not use the same reference for the contact and the coil.

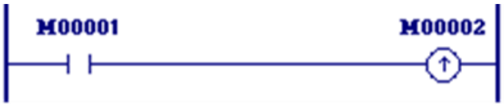
### Do not use:

**Figure 6**



### OK to use

**Figure 7**



## 2.5.3 Set, Reset Coil

**⚠ WARNING**

SET / RESET coils write an undefined result to the transition bit for the given reference. This result differs from that written by Series 90-70 CPUs and could change for future PACSystems CPU models.

Because they write an undefined result to transition bits, do not use SET or RESET coils with references used on POSCON or NEGCON transition contacts.

When a SET coil receives power flow, it sets its discrete reference ON. When a SET coil does not receive power flow, it does not change the value of its discrete reference. Therefore, whether the coil itself continues to receive power flow, the reference stays ON until the reference is reset by other logic, such as a RESET coil.

When a RESET coil receives power flow, it resets a discrete reference to OFF. When a RESET coil does not receive power flow, it does not change the value of its discrete reference. Therefore, its reference remains OFF until it is set ON by other logic, such as a SET coil.

The last solved SET coil or RESET coil of a pair takes precedence.

## 2.6 Online Editing Mode

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Online editing and online testing features | Not supported | Not supported | Allow editing and testing of logic changes that are permitted for a Run Mode Store. For programming software online help and PACSystems Logic Developer – PLC Getting Started, GFK-1918.details on using this feature, refer to the |

## 2.7 Variables

| Category | Series 90-70/90-30 | PACSystems |
|---|---|---|
| Multi-bit variables mapped to discrete memory | 8-bit and 16-bit variables | 8-bit, 16-bit, and 32-bit variables |
| | Mappable to %I, %Q, %M, %T, and %G.<br>WORD variables can sometimes be mapped to %S, depending on the instructions. | Mappable to %I, %Q, %M, %T, and %G.<br>WORD variables can sometimes be mapped to %S, while DWORD variables can sometimes be mapped to %SA, %SB, and %SC, depending on the instructions. |
| Bit addressing in non-discrete memory | Not supported | You can address individual bits in BYTE, WORD, INT, UINT, DINT, and DWORD variables in non-discrete memory (%R, %AI, %AQ, %L, %P, and %W). |
| BOOL arrays used to replace other data types | Not supported | Supported |
| Index of indirect references | 16 bits | 32 bits when referring to %W memory.<br>16 bits when referring to other memory areas. |

| Category | Series 90-70/90-30 | PACSystems |
|---|---|---|
| Symbolic variables | Not supported | Supported. A symbolic variable is a variable in logic that does not have an assigned reference address. Machine Edition handles all the mapping in a special portion of PACSystems user memory outside %R, %AI, %AQ, %P, %L, %W, %I, %Q, %M, %T, %S, and %G memory. |
| Publishing variables | Not supported | Supported |

## 2.7.1 System Variables

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| CPU Overtemperature Status (#OVR_TMP) | Not supported | Not supported | Supported |
| Fault Locating System Variables | Eight characters long | Not supported. | 10 characters long |
|  | Can locate 10 slots, from #0 through #9 |  | Can locate 32 slots, from #0 through #31 |
|  | Can locate 32 modules, from #0 through #31 |  | Can locate 256 modules, from #0 through #255 |

# 2.8 Communications

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Communicating with Machine Edition | Modem, serial port, and Ethernet. | Same as Series 90-70. | Ethernet supported on all versions. Serial port supported on later versions. refer to the IPI document for a specific CPU firmware version. |
| Ethernet adapters | Ethernet Interface Module IC697CMM742, in any rack. | CPUs 364 and 374 have an embedded Ethernet interface.<br>Ethernet Interface Module IC693CMM321, in any rack. | RX7i CPU modules have an embedded Ethernet daughterboard.<br>Ethernet module IC698ETM001 in RX7i main rack only. Maximum number: 3.<br>Ethernet module IC695ETM001 in RX3i main rack only. Maximum number: 4. |
| Configuring Ethernet | Involves temporarily connecting your computer to the PLC by serial cable. | Same as Series 90-70. | Can use the Set Temporary IP Address utility for a temporary connection, during which the permanent IP address can be set. Later versions support serial connection. Refer to the IPI document for a specific CPU firmware version. |
| Web-based data monitoring | Not supported. | Not supported. | Up to 16 web server and FTP connections (combined) |
| Network routing | Supported through CMM742 Ethernet Interface configuration. | Not supported. | Not supported. |
| Serial ports | Can be used to communicate with Machine Edition.<br>Provide SNP, Disabled, and Custom modes.<br>Refer to the Serial Communications User's Manual, GFK-0582. | Same as Series 90-70. | Ports 1 and 2 provide serial interfaces to external devices. Port 1 is also used for firmware upgrades. The third on-board serial port is used as the Ethernet station manager port. .<br>Provide RTU Slave, Message, and SNP Slave, Serial I/O, and Available modes. |
| Serial port default protocol | SNP | SNP | Modbus RTU. |
| Serial port communications from C applications | Scanf and Printf. | Same as Series 90-70. | ANSI-style read/write. |

## 2.9 Ethernet Global Data

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| EGD variables | Limit of 1,200 variables per exchange | Limit of 1,200 variables per exchange | No limit, if the total size of the EGD configuration does not exceed 64KB. |
| EGD upload | Does not support upload of EGD Configuration from the PLC to the Programmer | Same as Series 90-70. | Supported |
| Broadcast IP | Not supported | Not Supported | Broadcast option provides support for the production and consumption of EGD pages using the broadcast address of the local subnetwork. |
| Name Resolution | Supported. | Supported. Same restrictions as Series 90-70. | Not Supported. Adapter name in an EGD page configuration defaults to the rack slot location. |
| Consumed Period | Configurable. | Configurable for CPU364.<br>CPU374 same as PACSystems.; | Has a constant, read-only value of 200 ms. |
| Selective consumption | Not supported. | Not supported. | Supported. Ranges in a consumed EGD page can be set to Ignore. |
| %W | N/A | N/A | Supported in EGD page configuration |

## 2.10 Flash Operations

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Clearing Flash | Not supported. | Same as Series 90-70. | Supported. Clear affects all items (HWC, logic, and initial/forced values). |
| Reading/Verifying logic | Can read and verify logic and HWC separately. | Same as Series 90-70. | Cannot read and verify logic and HWC separately. |
| If EGD is present | Cannot read, write, or verify Flash memory. | Can read, write, verify, or clear Flash memory. (Same as PACSystems.) | Can read, write, verify, or clear Flash memory. |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Restoring from Flash | CPX models same as 90-30. NA for other 90-70 models. | When powering up from flash, the 90-30 CPU powers up in the mode that it was in when it powered down.<br>If the battery-backed memory is corrupt (no battery, super capacitor discharged) the 90-30 CPU can be configured to power up from flash. | OEMs can configure PACSystems to automatically restore user programs/data from Flash to user memory RAM. After such a restoration takes place, the PLC boots from RAM and not Flash, if the RAM's contents are valid (as determined during the power-on tests).<br>For a detailed explanation, see "Logic/Configuration Source and CPU Operating Mode at Power-up" in the PACSystems CPU Reference Manual, GFK-2222. |
| Writing to Flash | Does not write the transitions. | Does not write the transitions. | Takes a snapshot of the transitions that are currently set and writes them to Flash |
| Verifying Flash | Does not verify transitions. | Does not verify transitions. | Verifies the transitions. If a transitional changed value from 0 to 1 and back to 0, the value would be equal, but the transition could be unequal. |
| Store/Restore transition bits from flash | When reference tables are restored from flash, transition bits are set according to the differences between what was in RAM versus what was in flash at the time of the restore. |  | Stores/restores the state of the transition bits to/from flash along with the status values and overrides of each reference table. |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Logic/Configuration source and CPU Mode when Flash contains no configuration and Powerup Source in RAM is Always Flash | | Uses default logic/configuration and goes to Stop Disabled mode. | Uses logic/configuration from RAM. CPU mode determined as described in “Effect of Logic/Configuration Power-up Source on CPU Behavior” in the PACSystems CPU Reference Manual, GFK-2222. |
| Interruptible Flash Read/Write | The entire transfer process must complete before it can be canceled. | The entire transfer process must complete before it can be canceled. | The contents of flash memory or RAM are copied as individual files. This allows you to cancel a flash read or write operation during the copy process instead of waiting for the entire transfer process to complete. |

# 2.11 Memory

## 2.11.1 Differences in the Memory Areas Supported

| Memory Area | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| %GA through %GE | Supported | Uses %G (same as PACSystems). | Uses %G.<br>Target conversion from Series 90-70 PLC to PACSystems automatically converts %GA - %GE memory mappings to %G mappings. For details, see "Changes Made During the Conversion," |
| Bulk Memory Area | Accessed by means of SVC_REQ 36 | N/A | Accessed by mapping variables to %W memory |
| %W | In LD programs, the BMA is accessed by the SVCREQ 36 function. In C programs, the BMA is accessed through the PLCC_buil_mem() function. | N/A | Supported. Represents the Bulk Memory Area. |
| Symbolic | Not supported | Same as Series 90-70. | Supported. One memory area is used for discrete symbolic variables and another for non-discrete symbolic variables. |

## 2.11.2 Maximum Memory Sizes

The maximum memory size of some memory areas in PACSystems targets is larger than in Series 90 PLCs.

| Memory Area | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| %AI, %AQ | 8,192 words (16,384 bytes) | 32640 | 32,640 words (65,280 bytes) each |
| %I, %M, %Q | 12,288 points (1,536 bytes) | 2048 | 32,768 points (4,096 bytes) each |
| %T | 256 points (32 bytes) | 256 | 1,024 points (128 bytes) |
| %R | 16,384 words (32,768 bytes) | 32640 | 32,640 words (65280 bytes) |
| %W | Not supported | Not supported. | Maximum available user RAM |

| Memory Area | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Total user space | | | 10 megabytes |

For details on items that count against user memory, refer to the *PACSystems CPU Reference Manual*, GFK-2222.

### 2.11.3 Other Differences in Memory Support

| Memory Area | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| %P and %L | Buffer size not configurable. | Not supported. | Buffer size can be configured through the LD block's Extra Local Words property |

## 2.12 Redundancy

| Category | Series 90-70 | Series 90-30/RX3i | PACSystems Rx7i |
|---|---|---|---|
| CPU redundancy | Supported | Not supported. | Supported in redundancy CPUs |
| Genius redundancy | Supported | | Supported.<br>CPU over Genius supported in redundancy CPUs. |
| IP Address redundancy | Supported | | Supported in redundancy CPUs. |

## 2.13 Genius Communications

| Category | Series 90-70 | Series 90-30 | RX3i | PACSystems RX7i |
|---|---|---|---|---|
| Handling loss of GBC | Sets input data for devices associated with a failed GBC to Hold Last State, regardless of how the Input Default parameter is configured for each device. | Sets input data for devices associated with a failed GBC to zero. | Sets input data for devices associated with a failed GBC to the state specified (Default/Hold Last State) by the Input Default parameter for the GBC. | Sets input data for devices associated with a failed GBC to the state specified by the Input Default parameter for each device. |

| Category | Series 90-70 | Series 90-30 | RX3i | PACSystems RX7i |
|---|---|---|---|---|
| Handling losses and additions of Genius device | The Series 90-70 GBC logs a Loss of Device or an Addition of Device fault and the Series 90-70 PLC updates the appropriate fault locating reference. | The Series 90-30 GBC does not log a fault. The Series 90-30 PLC does not support fault locating references. | The Series 90-30 GBC does not log a fault, therefore the RX3i does not update fault locating references. | The Series 90-70 GBC logs a Loss of Device or an Addition of Device fault and the RX7i updates the appropriate fault locating reference. |
| Handling of data for lost redundant Genius blocks | Immediately applies the default input data to the input reference tables and updates the associated diagnostic tables. | N/A | NA | Updates input data and input diagnostic tables with the default data during the input scan immediately following the loss of device fault. Updates output diagnostic data tables during the output scan immediately after the loss of device fault. |
| Setting of Force Present status bit, FRC_PRE %S12. | Limited to Genius blocks. | NA | Not limited to Genius blocks. If any input module reports that it has a force present, this bit is set. | Not limited to Genius blocks. If any input module reports that it has a force present, this bit is set. |
| Point fault references | Supported by Series 90-70 CPUs, Series 90-70 GBC, and other legacy Series 90-70 modules. | Not supported by Series 90-30 CPU or legacy Series 90-30 modules. | RX3i CPU supports point fault references. Series 90-30/RX3i GBC does not. | Supported |

# 2.14 I/O and Intelligent Modules

Table 1

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| Discrete output modules | Discrete output modules controlled by a Series 90-70 CPU drive their outputs only after the second sweep. DO I/O have no effect in the first sweep because outputs are not enabled until they are part of an output scan. | Same as PACSystems | Discrete output modules controlled by a PACSystems CPU drive their outputs to the values set by user logic the first time they are part of an output scan.<br>A DO I/O function causes a module's outputs to be enabled at the point of the DO I/O execution (if that module's outputs have not already been enabled). |
| Default conditions for input modules on STOP to RUN transition. | When the CPU transitions from STOP to RUN mode, the modules' input values are retained. | | When the CPU transitions from STOP to RUN mode, the modules' input values are set to 0. |
| Do I/O to High Speed Counter | If %AI is specified, returns only AI input data. If %I is specified, returns all input data (I and AI). | Same as PACSystems | Returns all input data (I and AI). |
| I/O interrupt blocks | Prioritizes timed interrupts before I/O interrupts. | NA (does not support I/O interrupts) | Normally scheduled relative to each other on a first-come first-served basis.<br>CPUs with later firmware versions allow preemptive block scheduling. For availability on a given CPU version, refer to the *IPI* document provided with that CPU. |
| Setting up an I/O Interrupt Block for an I/O Address that is not configured and/or enabled for interrupts. | No faults - CPU can go to Run mode. | NA (does not support I/O interrupts) | CPU goes to Stop-Fault mode after download to CPU. Controller Fault Table must be cleared to place CPU in Run mode. Module interrupts should be configured/enabled in hardware configuration. |

| Category | Series 90-70 | Series 90-30 | PACSystems |
|---|---|---|---|
| I/O interrupts on High Speed Counter and IC697MDL641 module | The legacy transitions (POSCON/NEGCON) are not set. | N/A | The legacy transitions are always set for every one of the discrete input references at the time the interrupt occurs. (If the new state of the bit is on, the up-transition contact will be true. If the new state is OFF, the down transition contact will be true). |
| Analog Input Interrupts | HIALR and LOALR fault contacts are not set at the time the interrupt logic is run. | Point faults not supported. | If point faults are enabled, the HIALM or LOALM fault contact will be set to TRUE before running the interrupt logic. |
| Analog Expansion modules | No limit on number of expanders. Analog base module and expansion modules can be assigned to different scan sets. | N/A | (RX7i only) Maximum of three expanders allowed in an RX7i rack. Expansion modules are assigned to the same scan set as the analog base module. |

# Chapter 3: Converting an Application from Series 90-70 to PACSystems RX7i

**⚠ WARNING**

There may be execution differences when converting an application from a Series 90-70 target to a PACSystems target. It is the application developer's responsibility to validate and test the application execution prior to deployment into a production environment.

**⚠CAUTION**

PLC target conversions are irreversible. When logic blocks are deleted during a conversion, they cannot be restored. It is recommended that you make a backup of your project before converting a target in it.

## 3.1 Preparing for the Conversion

### 3.1.1 Analog Expander Modules

If you want to convert a Series 90-70 rack configuration that contains IC697ALG440 or IC697ALG441 analog input expander modules, you must ensure that the IC697ALG230 base module and the expander modules are in certain slots as detailed below; otherwise, the modules will not be converted. (If you do not use expander modules, the base module can be configured in any slot.) If you use expander modules, you must put the base module in slot 3, 4, or 5 of the Series 90-70 rack before conversion.

- If you put the base module in slot 3 of the Series 90-70 rack, you must put the IC697ALG440 or IC697ALG441 expander modules in slots 4, 5, or 6.
- If you put the base module in slot 4, the expander modules can go in slots 5 or 6.
- If you put the base module in slot 5, you can have only one expander module, in slot 6.

***Note:*** *Another convertible configuration is a base module in slot 3 and its expander module in slot 4, and another base module in slot 5 and its expander module in slot 6.*

### 3.1.2 PCM, CMM, and DLAN Modules

If you want to convert a Series 90-70 rack configuration that contains an IC697PCM711, IC697CMM711, or IC697BEM761 you must ensure that these modules are in slots less than or equal to 9 in an RX7i rack. If any of these modules occupies a slot greater than 5 in the 90-70 rack configuration, it will not be converted because the corresponding destination slot in the RX7i rack configuration would be greater than 9. For example, an IC697PCM711 module in slot 6 of a Series 90-70 rack configuration would not be converted because the destination in the RX7i rack would be slot 11.

**Note:** *If your 90-70 application uses a PCM, CMM, or DLAN module, please refer to the PACSystems CPU Reference Manual, GFK-2222 for information about using these modules with PACSystems RX7i.*

## 3.1.3 VME_ Instructions

The PACSystems Rx7i provides Bus_Read and Bus_Write functions like the Series 90-70 VME_Read and VME_Write functions. When converting a Series 90 70 application, some modifications to hardware configuration and logic are required to use the new instructions. These changes must be made manually after converting the application. Before converting your application, review the descriptions of VME addressing and information needed to complete the conversion.

**Note:** *For details on selecting, configuring, and programming VME modules in a PACSystems control system, refer to PACSystems RX7i User's Guide to Integration of VME Modules, GFK-2235.*

## 3.1.4 PACSystems vs. Series 90-70 VME Addressing Schemes

## 3.1.5 VME Modules

A typical VME board is configured to respond to certain VME memory addresses. Some boards are configured using one or more jumpers to set an Address Modifier (AM code) and a base address that the board will respond to. Typically, a board has one or two specific memory areas that can be accessed. PACSystems refers to these areas as Regions. A Region is comprised of a base address, AM code, and size, as well as a “width”, termed Interface Type, (byte, word, dword, etc) that can be accessed.

In PACSystems, each VME Region on a board is specified in the Hardware Configuration. PACSystems supports a maximum of eight regions per module. A Region specification includes a region number, AM code, Base Address, Size, and Interface Type. Specifics of these settings are covered in the programming software Help Information. The PACSystems BUS_ functions include a region (RGN) parameter that refers to the regions configured in the Hardware Configuration information.

In Series 90-70, the AM code and VME address are specified as parameters to the VME_ function itself. The “width” specification is determined by the instruction used (for example, READ_BYTE versus READ_WORD).

### Information Needed to Complete the Conversion

To complete the conversion process, you will need to identify the memory area(s) being used on each VME module, including AM code, and VME base address. This information should be provided with the VME board being used. If not available, this information can be obtained by reviewing the logic in the Series 90-70 application. As a rule, a separate region will be needed for each AM code used.

### 3.1.6 PCM Applications

The VME address assignments for VME modules in an RX7i system differ from the assignments in a 90-70 system. If an application program running on the IC697PCM711 accesses the VME bus (that is, it uses Set_vme_ctl, Vme_read, Vme_test_and_set, or Vme_write), the VME addresses used by that program will need to be updated to be in agreement with the PACSystems RX7i VME addressing assignments. To determine the correct VME addresses to use on the RX7i, please refer to the following sections in the PACSystems RX7i User's Guide to Integration of VME Modules, GFK-2235:

"VME Addresses for Emerson Modules in the Main Rack"

"VME Addresses for Emerson Modules in Expansion Racks"

Also please note that the S9070_xxxx macros, listed below, that are provided by the PCM C toolkit in the file Vme.h cannot be used to calculate VME addresses in an RX7i system.

```
S9070_RACKSLOT_VALID(r,s) (r>=0&&r<=7&&s>=2&&s<=9)

S9070_VME_HI_ADDR(r,s) ((r)?(0xF0-(0x10*(r))+2*((s)-2)):(2*((s)-2)))

S9070_VME_SHORT_ADDR(s) (0x800*s)
```

## 3.2 Converting the Series 90-70 Target

To convert a Series 90-70 target to a PACSystems RX7i target in the programming software:

1. In the Project tab of the Navigator, right click the target you want to convert and select Properties. The Inspector displays the target properties.
2. In the Properties, select the PACSystems RX7i family. A Target Conversion Warning message appears. If you want to continue with the conversion, click OK.
3. The target is converted to the PACSystems RX7i family. The target conversion report is displayed in the InfoViewer when the conversion is complete.
4. Review the target conversion report, correct any problems identified, and validate the application. The application should be thoroughly tested to detect problems that may be caused by execution differences before deploying it in a production environment.

   For additional information, please refer to "Changes Made During the Conversion" and "Finishing the Conversion"

**⚠ WARNING**

There may be execution differences when converting an application from a Series 90-70 target to a PACSystems target. It is the application developer's responsibility to validate and test the application execution prior to deployment into a production environment.

# 3.3 Changes Made During the 90-70 to PACSystems RX7i Conversion

Each Series 90-70 module supported by a PACSystems RX7i target is remapped from its Series 90-70 rack and double-width slot to the corresponding PACSystems RX7i rack and single-width slots.

**Note:** *Because a PACSystems RX7i power supply uses only one slot, the resulting PACSystems RX7i slot number is calculated as follows:*

*PACSystems RX7i slot = (2 * [Series 90-70 slot number]) – 1*

- The parameter values for each converted module are preserved whenever possible. Parameters unique to RX7i are set to their default settings.
- Ethernet Global Data (EGD) is converted. For each rack, adapter 1 is converted to the RX7i's CPU embedded Ethernet daughterboard, while adapters 2, 3, and 4 are converted to RX7i Ethernet modules (IC698ETM001), and adapters 5 and beyond are ignored.

**Note:** *The Series 90-70 slot in which adapter 1 resides is left empty in RX7i, as per the formula empty RX7i slot = (2 * [Series 90-70 slot number of adapter 1]) - 1*

- References to %GA - %GE memory areas are converted to %G addresses with new offsets as follows:

**Table 2**

| Pre conversion memory type | Post conversion memory type and offset |
|---|---|
| %G | %G+0 |
| %GA | %G+1280 |
| %GB | %G+2560 |
| %GC | %G+3840 |
| %GD | %G+5120 |
| %GE | %G+6400 |

- C blocks are retained and flagged in the report. You may need to edit them. You will also need to recompile them with the PACSystems C Toolkit and update them in the PACSystems target.
- All C programs are deleted.
- LD blocks are converted and scanned for instructions that require updating.

**Note:** *The first validation of a converted target flags the LD instructions that are not supported in the RX7i.*

- The original Series 90-70 system variables are deleted except when they are used in logic. PACSystems RX7i system variables are added to the target. A warning is reported for each system variable found to be used in logic. It is up to you to ensure that these system variables are still valid for the new target type.

- When you convert a Series 90-70 target to an RX7i target, all Series 90-70 fault locating system variables are converted to PACSystems RX7i versions by inserting a 0 before the slot number. For example, if #BUS_121 was used on a Series 90-70 target, the variable is renamed to #BUS_1021 when the target is converted to RX7i. Also, the #RACK_0r variables are converted to #RACK_00r.

# 3.4 Finishing the Conversion

## 3.4.1 Review the Target Conversion Report

The target conversion report, displayed when the conversion is completed, summarizes the results of hardware configuration conversion and logic conversion. Items that were not converted are identified.

**Note:** *The conversion report does not warn about all possible differences in logic execution. A validity check after conversion may report problems that could not be detected during conversion. Execution differences may exist when converting from Series 90-70 to PACSystems RX7i, even for rungs that were not mentioned in the report or reported during validity checks.*

The report provides an analysis of each LD block that warns of unsupported instructions, unsupported service requests, fault locating reference usage, instructions that were converted, and instructions that could not be converted and why.

Lines displayed in red characters warn of the most important potential differences in logic execution (for example, due to the different execution order of a Series 90-70 PLC or PACSystems RX7i when faced with branches in logic). For each potential difference reported, you should examine the logic. Look for variables used as both inputs and outputs on different instructions where the instruction with the input is not on the same row as the instruction with the output.

The report is saved in the Documentation files folder in the Supplemental Files folder in the resulting PACSystems RX7i target. You can print the report directly from the InfoViewer or print it from the copy saved in the Documentation Files folder.

## 3.4.2 Replace VME_ Instructions with BUS_ Instructions

RX7i provides a set of BUS_ instructions that can be used to access data in a module on a bus: BUS_RD, BUS_WRT, BUS_RMW, and BUS_TS. These instructions are similar to the VME_ instructions supported by Series 90-70. Series 90-70 VME_ instructions are not automatically converted to RX7i BUS_ instructions.

The following is a summary of the changes that must be made after converting the application from Series 90-70 to PACSystems.

- Determine the number of regions needed for a given board, and the AM code and base address for each region. Note that each AM code requires a separate region.

- Using Machine Edition hardware configuration, configure each non-Emerson board, adding the appropriate number of regions, and specifying the necessary AM Code, Base Address, Size, and Interface type. Each region will be referred to by number within the LD program.
- Modify the LD function calls.
  - Replace each VME_ instruction with the corresponding BUS_ instruction. (For example, VME_READ_WORD should be replaced by BUS_READ_WORD; VME_CFG_READ should be replaced by a BUS_READ_ instruction, etc.) Unsupported instructions are identified in the Target Conversion Report.
  - Add the Rack and Slot parameters to refer to the appropriate non VME module.
  - Specify the appropriate Region number used in the hardware configuration.
  - Compute the “address” parameter for the function. Note that the value needed is now an offset relative to the specified region as opposed to an absolute VME address. Therefore, you must subtract the base address specified by the region you are referring to. For example, if the Series 90-70 instruction used an address of 0x400100 and you have specified 0x400000 as the base address of the region, you would enter 0x100 as the offset.
  - If your RX7i application program needs to access the dual port memory of a PCM, CMM, or DLAN, use the BUS READ and WRITE functions. When accessing one of these modules, set the function’s Region parameter to 1. (For the PCM, CMM, and DLAN modules, region 1 is predefined to be the module's entire dual port memory. Configure these modules according to their catalog numbers; do not configure these modules as “VME modules.”)

Note that other (optional) parameters have been added. Specifics on these parameters are provided in the Machine Edition online help.

### 3.4.3 Increasing Stack Allocation for Programs Converted from Series 90-70

Series 90-70 programs are converted to RX7i with the same stack allocation. RX7i uses more stack space than the Series 90-70, so some user programs may not run after conversion. To increase the stack space, right click the _MAIN block and select Properties. Stack Size is listed at the bottom of the Properties page. The default stack size in RX7i programs is 64KB. Programs with many nested calls may need more stack space. Generally, the stack for the converted RX7i program should be set to approximately three times the stack size of the Series 90-70 version of the program. A diagnostic fault is displayed if the program runs out of stack space.

# Chapter 4: Converting an Application from Series 90-30 to PACSystems RX3i

## 4.1 Preparing for the Conversion

### 4.1.1 End Instruction

PACSystems does not support the END instruction, which unconditionally terminates program execution and transfers control to the beginning of the program (the first rung of the _MAIN block) for the next scan. Following are suggestions for reorganizing programs that use the END instruction:

- Adding JUMP(s) to the end of the block(s) in the CALL chain
- Separating the logic following the END instruction into a separate block that is called only if the END would not have been executed.
- Implementing other means of debugging the LD program, such as saving copies of temporary register contents when the circumstances being debugged occur.

### 4.1.2 Updated 24V Analog Modules (IC693ALG220, IC693ALG221, and IC693ALG222C)

If you are using an old version (ALG220G, ALG221G, ALG222C or earlier) of these modules, you must provide an external 24V power source to the backplane terminals, or replace the old version with the latest version.

### 4.1.3 SRTP Communication with Older Clients

The RX3i CPU can be placed in any slot in rack 0 (except the last slot). PACSystems provides an SRTP Destinations service that allows an external client using SRTP to find the CPU's rack and slot location in order to communicate with it. However, legacy clients, such as Series 90 PLC applications using SRTP channels and HCT host applications do not use this service. They assume that the server CPU is always in rack 0, slot 1 as is required in the Series 90 systems.

To support communications with legacy SRTP clients such as Series 90 PLCs using SRTP Channels, the RX3i redirects service requests arriving on an SRTP server connection destined for rack 0 slot 1 to rack 0 slot 2 if:

- There is only one CPU in the system.
- The CPU is in rack 0 slot 2.
- The remote client has not issued an SRTP Destination service on the connection to discover the rack and slot of the CPU.

Redirecting the services from rack 0 slot 1 to rack 0 slot 2 consists only of changing the rack and slot portion of the destination address within the service request mailbox message. The content (payload) of the service request is not examined or modified, nor is the service request response mailbox message from the CPU.

All services used by SRTP channels clients in the Series 90 and all HCT services can be successfully redirected.

## 4.1.4 CPU Slot Location

The RX3i CPU is a double-width module whose connector is right justified as viewed when installed in a rack. It is referenced for configuration and by user logic applications by the leftmost slot that it occupies. For example, if the RX3i CPU has its physical connector inserted into slot 4, which means it occupies slots 3 and 4, the CPU is referenced as being in slot 3. The referenced location of the CPU is not determined by what slot the physical connector is in, but by the leftmost slot occupied by the entire module.

The RX3i CPU may be in any slot in the main rack except slot 11 of a 12-slot rack or slot 15 of a 16-slot rack, because these slots would require the physical connector to be in the slot reserved for an expansion module.

When migrating a Series 90-30 CPU system to a PACSystems RX3i CPU, be aware that to maintain the Slot 1 location of the CPU, only a single-width power-supply may be used in slot 0. Therefore, if the application using an existing Series 90-30 system must maintain a slot 1 CPU and uses a double-width power-supply, the power supply must be in a slot to the right of the RX3i CPU in Slot 1.

In deciding to place the CPU in a slot other than Slot 1, you should be aware of the possible application migration issues that could arise. The following table lists the areas that could be affected when migrating an application from one CPU slot to another.

### CPU Slot Placement Issues

**Table 3**

| Item Affected | | How Affected |
|---|---|---|
| User Logic | Service Request #15 (Read Last-Logged Fault Table Entry) | Location of CPU faults will not be the standard 0.1 location, but will reflect the slot the CPU is located in. User logic that decodes fault table entries retrieved by these service requests may need updating. |
| | Service Request #20 (Read Fault Tables) | |
| | Communications Request (COMM_REQ) | COMM_REQs directed to the CPU (e.g. those directed to the serial ports of the CPU) will need to be updated with the correct CPU slot reference. |
| H/W Configuration | CPU Slot location | Slot location of the CPU must be updated in the HW Configuration to reflect the CPU's true location. |
| Fault Tables | Faults logged for the CPU | The location of faults logged for the CPU in the fault table will not be the standard 0.1 (rack.slot) location, but will reflect the CPU's actual slot. |
| External Devices | Series 90 PLCs | |

| Item Affected | How Affected |
|---|---|
| | Remote Series 90 PLCs that use SRTP Channels COMM_REQs expect the CPU to be in slot 1. To support communications with Series 90 SRTP clients such as Series 90 PLCs using SRTP Channels, the RX3i internally redirects incoming SRTP requests destined for {rack 0, slot 1} to {rack 0, slot 2}, provided that the CPU is located in rack 0 slot 2 (and the remote client has not issued an SRTP Destination service on the connection to discover the rack and slot of the CPU). This special redirection permits Series 90-30 applications that expect the power supply to be located leftmost and the CPU to be located to the right of the power supply to function. Attempts to establish channels with CPUs in slots other than 1 or 2 will fail if initiated from Series 90 PLCs. |
| | HMI and External Communication Devices |
| | All external communication devices that interact with the CPU should be checked for compatibility with CPU slot locations other than slot 1. Problems may arise with, but are not limited to, initial connection sequences and fault reporting. View -Machine Edition customers should select "GE SRTP" as their communications driver – it can communicate with a CPU in any slot. |
| | Host Communications Toolkit (HCT) |
| | Applications that utilize the Host Communications Toolkit may require updated drivers. |

## 4.2 Converting the Series 90-30 Target

To convert a Series 90-30 target to a PACSystems RX3i target in the programming software:

1. In the Project tab of the Navigator, right click the target you want to convert and select Properties. The Inspector displays the target properties.
2. In the Properties, select the PACSystems RX3i family. A Target Conversion Warning message appears. If you want to continue with the conversion, click OK.
3. The target is converted to the PACSystems RX3i family. The target conversion report is displayed in the InfoViewer when the conversion is complete.
4. Review the target conversion report, correct any problems identified, and validate the application. The application should be thoroughly tested to detect problems that may be caused by execution differences before deploying it in a production environment.

   For additional information, please refer to "Changes Made During the Conversion," and "Finishing the Conversion."

**⚠ WARNING**

There may be execution differences when converting an application from a Series 90-30 target to a PACSystems target. It is the application developer's responsibility to validate and test the application execution prior to deployment into a production environment.

# 4.3 Changes Made During the 90-30 to PACSystems RX3i Conversion

## 4.3.1 Hardware Configuration

- Each Series 90-30 module supported by a PACSystems RX3i target is remapped from its Series 90-30 rack slot to the corresponding PACSystems RX3i rack and slots.
- The RX3i CPU and default power supply require two slots. Slot locations of other modules are adjusted as needed.
- Series 90-30 CPUs with embedded Ethernet interface are converted to an RX3i CPU and an IC695ETM001 peripheral Ethernet module. Slot locations are adjusted as needed.
- The parameter values for each converted module are preserved whenever possible. Parameters unique to RX3i are set to their default settings.
- Power consumption requirements are converted from Watts to Amps.

## 4.3.2 Logic

- C blocks are retained and flagged in the report. You may need to edit them. You will also need to recompile them with the PACSystems C Toolkit and update them in the PACSystems target.
- All C programs are deleted.
- IL (Instruction List) and SFC (Sequential Function Chart) programs are not translated. IL and SFC programming are not supported.
- LD blocks are converted and scanned for instructions that require updating.

The following instructions are flagged as not supported:

- SVC_REQ #41 (PEEK), SVC_REQ #42 (Daughterboard Info). These are not translated. Other means of debugging the operation of the system and of determining daughterboard revision information are provided.
- SVC_REQ 46 Fast Backplane Status Access
- SCV_REQ #48 & #49 (auto-restart parameters). These are not translated, since the auto-restart feature is not implemented. The program is translated successfully without them, but you are notified that they have been omitted.
- END. Not supported.

The following instructions are flagged for manual translation:

- SVC_REQ 6, Change/Read Number of Words to Checksum
- SVC_REQ 15, Read Last-Logged Fault Table Entry
- SVC_REQ 23, Read Master Checksum
- SVC_REQ 26/30, Interrogate I/O. Note that the Series 90-30 Interrogate I/O functionality is supported in PACSystems by fault locating references.

The following instructions are changed:

- WORD_TO_REAL instruction translated to UINT_TO_REAL.
- REAL_TO_WORD instruction translated to REAL_TO_UINT.
- Enhanced DO_IO translated to standard DO_IO (The constant ALT parameter is discarded and ignored.)
- Non-nested JUMP, LABEL, MCR, & ENDMCR. These are translated to the corresponding nested JUMPs, LABELs, MCRs, & ENDMCRs.

## 4.4 Finishing the Conversion – Reviewing the Target Conversion Report

The target conversion report, displayed when the conversion is completed, summarizes the results of hardware configuration conversion and logic conversion. Items that were not converted are identified.

**Note:** *The conversion report does not warn about all possible differences in logic execution. A validity check after conversion may report problems that could not be detected during conversion. Execution differences may exist when converting from Series 90-30 to PACSystems, even for rungs that were not mentioned in the report or reported during validity checks.*

The report provides an analysis of each LD block that warns of unsupported instructions, unsupported service requests, fault locating reference usage, instructions that were converted, and instructions that could not be converted and why.

Lines displayed in red characters warn of the most important potential differences in logic execution. For each potential difference reported, you should examine the logic.

The report is saved in the Documentation files folder in the Supplemental Files folder in the resulting PACSystems target. You can print the report directly from the InfoViewer or print it from the copy saved in the Documentation Files folder.

# Chapter 5: Hot Standby CPU Redundancy System Conversions

Converting a redundancy target from one platform to another is very similar to converting a non-redundant target. This section describes conversion issues that apply only to redundancy systems.

The following types of conversions are described:

- Converting a PACSystems RX7i Redundancy System to RX3i
- Converting a PACSystems RX3i Redundancy System to RX7i
- Converting a PACSystems RX3i Redundancy System to Series 90-70
- Converting a Series 90-70 Redundancy System to PACSystems RX7i
- Converting a Series 90-70 Redundancy System to PACSystems RX3i

## General

- Main racks, power supply modules, redundancy CPUs and redundancy modules are converted to the corresponding default selection in the new target. The parameter values of all converted modules are preserved when possible. Parameters that are unique to the target system are set to their default values.
- All other modules are removed from the hardware configuration.
- Expansion racks and the modules they contain are removed.
- EGD exchanges are retained with their names intact (except when converting RX3i to Series 90-70). The Adapter Name of each exchange is recalculated to match the location of the corresponding converted Ethernet interface. (When converting RX3i to Series 90-70, EGD is deleted.)

**⚠ WARNING**

There may be execution differences when converting an application from one target to another. It is the application developer's responsibility to validate and test the application execution prior to deployment into a production environment.

# 5.1 Converting a PACSystems RX7i Redundancy System to RX3i

## Hardware Configuration

- The PACSystems Rx7i main rack converts to the default PACSystems Rx3i main rack (IC695CHS012)
- RX7i AC power supply (PSA100 or PSA350) converts to RX3i (PSA140) in slots 0 and 1.
- RX7i DC power supply (PSD300) converts to RX3i (PSD140) in slots 0.
- RX7i Redundant CPU (CRE0x0) converts to CRU320 in slots 2 and 3 and ETM in slot 4 when PACS Machine Edition selects PSA140 as power supply. RX7i Redundant CPU (CRE0x0) convert to CRU320 in slots 1 and 2 and ETM in slot 3 when PAC Machine Edition selects PSD140 as power supply.
- In order from lowest slot to highest slot convert each of the following modules and assign them to the next highest available slot on the RX3i:
  - RX7i RMX converts to RX3i RMX128
  - RX7i CMX converts to RX3i CMX128
  - RX7i ETM001 convert to the RX3i ETM module

## Conversion of other elements of the target

- All EGD exchanges are retained with their names intact. The value of the Adapter Name property of each exchange is recalculated to match the location of the corresponding converted Ethernet module.
- The Genius bus configuration is deleted.
- C blocks are retained and flagged in the target conversion report.
- FBD blocks and ST blocks are retained.
- PACSystems system variables, including fault locating system variables, are retained.
- LD blocks are retained. PACSystems Rx7i and Rx3i support the same instruction set.
- Transfer list properties of user defined function blocks, IEC timers, FTrig and RTRIG instances are preserved.
- Transfer list properties of variables (Symbolic, Structured Variables and all mapped reference variables) are preserved.

# 5.2 Converting a PACSystems RX3i Redundancy System to RX7i

## Hardware Configuration

- The PACSystems Rx3i main rack is converted to the default PACSystems RX7i main rack (IC698CHS017)
- All RX3i power supplies are converted to one RX7i PSA350 module in slot 0.
- The CRU320 module is converted to an RX7i CRE020 module in Slots 1 and 2.
- Configuration data from one ETM at the lowest slot number is applied to the CRE020 Ethernet daughter board.
- Other RX3i ETMs are converted to RX7i ETMs starting at the next lowest available slot.
- RX3i CMX/RMX modules are converted to their corresponding RX7i CMX/RMX modules at the next lowest available slot.
- All other modules are removed from the hardware configuration.
- The parameter values of all converted modules are preserved.
- Expansion racks and all the modules they contain are removed.

## Conversion of other elements of the target

- All EGD exchanges are retained with their names intact. The value of the Adapter Name property of each exchange is recalculated to match the location of the corresponding converted Ethernet module.
- C blocks are retained
- FBD blocks and ST blocks are retained.
- The PACSystems system variables, including fault locating system variables, are retained.
- LD blocks are retained. PACSystems RX7i and Rx3i support the same instruction set.
- Transfer list properties of user defined function blocks, IEC timers, FTrig and RTRIG instances are preserved.
- Transfer list properties of variables (Symbolic, Structured Variables and all mapped reference variables) are preserved.

# 5.3 Converting a PACSystems Rx3i Redundancy System to Series 90-70

## Hardware Configuration

- The PACSystems RX3i main rack is replaced with the default Series 90-70 main rack (IC697CHS790).
- The PACSystems RX3i main rack power supply/supplies are replaced with the Series 90-70 power supply (IC697PWR710) in slot 0.
- The PACSystems RX3i CRU320 is replaced with the Series 90-70 CPU (IC697CPX782) in slots 1.
- All the other Rx3i modules are removed.
- The parameter values for the Series 90-70 CPU IC697CPX782 are set to their default values.
- The Series 90-70 expansion racks are removed with all the modules they contain.

## Conversion of the fault locating system variables

- Only some PACSystems fault locating system variables are converted to Series 90 70 fault locating system variables. The conversion occurs when removing two zeroes from the variable name is possible:
  - #RACK_000r variables are converted to #RACK_0r.
  - #SLOT_0rss variables are respectively converted to #SLOT_rs when the slot number ranges from 01 through 09: the zero is removed after the underline character and the leading zero is removed from the slot number. When the slot number ranges from 10 through 31, however, the variables are not converted. When the Series 90-70 target is validated, errors are issued for the unconverted variables.

## Conversion of other elements of the target

- Reference addresses in the %G memory area are converted to %G through %GE reference addresses with new offsets described in the PACSystems CPU Reference Manual, GFK 2222.
- The EGD component is deleted.
- C blocks, FBD blocks, and ST blocks are deleted.
- C programs are deleted.
- LD blocks are retained.

# 5.4 Converting a Series 90-70 Redundancy System to PACSystems RX7i

Before converting a Series 90 redundancy application to PACSystems, you should review the general information that applies to Series 90-70 to RX7i conversions, presented earlier in this manual.

The programming software automatically makes the following changes to the configuration during the conversion:

- The redundancy CPUs from the Series 90-70 rack system are converted to CRE020 CPUs. The Redundancy tab parameters are copied without any changes.
- The Series 90-70 RCM is replaced with two RMX modules.
- The Control Strategy setting in the CPU (GHS or GDB) is changed to HSB.

## 5.4.1 Control Strategy Conversion

PACSystems supports only the HSB control strategy, which is equivalent to the GDB control strategy in the Series 90-70 PLC. The GHS configuration option is not supported by PACSystems.

With the HSB control strategy, all redundant Genius outputs must be included in the Output Transfer List. For details, refer to “Transfer List” in the PACSystems Hot Standby CPU Redundancy User’s Manual, GFK-2308.

If your 90-70 target used the GHS control strategy, you may need to adjust the %Q and %AQ ranges in the Output Transfer list so that they include all redundant Genius outputs. In addition, if preferred master is desired, Ladder Logic application programming is required (see “Logic for Implementing Preferred Master” in the PACSystems Hot Standby CPU Redundancy User’s Manual, GFK-2308).

Check your program logic for the use fault locating references that correspond to the remote RCM (rack 7). Adjust them to refer to the local RMX modules.

## 5.4.2 Applications with a Programmable Coprocessor Module

If your 90-70 application includes a Programmable Coprocessor Module (IC697PCM711), you may need to adjust the application running in the PCM to make it compatible with the PACSystems version of CPU redundancy. Specifically, when power is applied to the Rx7i rack, the Rx7i Redundancy CPU can take a longer time to respond to the first communication request made by the PCM. To account for this delay, you could adjust your PCM application so that it retries the first communication requests for approximately 60 seconds before declaring an error.

### 5.4.3 Converting the Target

To convert a Series 90-70 target to a PACSystems RX7i target in the programming software:

1. In the Project tab of the Navigator, right click the target you want to convert and select Properties. The Inspector displays the target properties.
2. In the Properties, select the PACSystems RX7i family. A Target Conversion Warning message appears. If you want to continue with the conversion, click OK.
3. The target is converted to the PACSystems RX7i family. The target conversion report is displayed in the InfoViewer when the conversion is complete.
4. Review the target conversion report, correct any problems identified, and validate the application. The application should be thoroughly tested to detect problems that may be caused by execution differences before deploying it in a production environment.

   For additional information on changes made during conversion, refer to "Changes Made During the 90-70 to PACSystems RX7i Conversion" on page 31.

**⚠ WARNING**

There may be execution differences when converting an application from a Series 90-70 target to a PACSystems target. It is the application developer's responsibility to validate and test the application execution prior to deployment into a production environment.

## 5.5 Converting a Series 90-70 Redundancy System to PACSystems RX3i

Before converting a Series 90 application to PACSystems, you should review the general information that applies to Series 90 to PACSystems conversions, presented earlier in this manual.

### Hardware Configuration

- The Series 90-70 main rack is replaced with the default PACSystems RX3i main rack (IC695CHS012).
- The Series 90-70 power module and CPU are replaced with the default PACSystems RX3i power module (IC695PSA140) in slot 0 and 1.
- The Series 90-70 redundancy CPU is replaced with the PACSystems RX3i CPU (CRU320) in slots 1 and 2.
- In order from lowest slot to highest slot convert each of the following modules and assign them to the next highest available slot on the RX3i:
- 90-70 Redundant Communication Module (IC697RCM711) converts to two RX3i RMX128 board.
- 90-70 Ethernet Controller Modules (IC697CMM741/IC697CMM742) convert to Rx3i ETM module

- All the other Series 90-70 modules are removed.
- The parameter values for the CPU are preserved whenever possible. When parameters are unique to PACSystems RX3i, they are set to their default settings.
- Expansion racks and all the modules they contain are removed.

## Conversion of the fault locating system variables

- All Series 90-70 fault locating system variables are converted to PACSystems RX3i fault locating system variables by inserting two zeroes to each variable name as follows:
- #RACK_0r variables are converted to #RACK_000r.
- #SLOT_rs variables are converted by inserting a zero after the underline character and inserting a zero before the slot number.
- #M_rsbmm and #BUS_rsb variables are removed.

## Conversion of other elements of the target

- Ethernet Global Data (EGD) is converted. The first eight adapters encountered in the original hardware configuration are converted to PACSystems RX3i Ethernet modules (IC695ETM001), and any other adapters are ignored. Every Adapter Name is converted from a user-defined string to a rack.slot string. On produced exchanges, a Destination Type set to Name is converted to Unicast with an IP Address set to 0.0.0.0, which you must replace with a valid IP address.
- Reference addresses in the %GA - %GE memory areas are converted to %G reference as described in the PACSystems CPU Reference Manual, GFK 2222.
- The Genius bus configuration is deleted.
- C blocks are retained and flagged in the target conversion report.
- C programs are deleted.
- The original Series 90-70 system variables are deleted except when they are used in logic. PACSystems RX3i system variables are added to the target.
- LD blocks are retained

## Technical Support & Contact Information

Home link: http://www.Emerson.com/Industrial-Automation-Controls

Knowledge Base: https://www.emerson.com/Industrial-Automation-Controls/support

**Note:** If the product is purchased through an Authorized Channel Partner, please contact the seller directly for any support.

Emerson reserves the right to modify or improve the designs or specifications of the products mentioned in this manual at any time without notice. Emerson does not assume responsibility for the selection, use or maintenance of any product. Responsibility for proper selection, use and maintenance of any Emerson product remains solely with the purchaser.