

# Register Usage Summary

---

This appendix lists the general-purpose registers. It includes these sections:

- Overview  
Describes the type of register usage information included in this appendix.
- API Register Usage  
Describes the API register usage.

## Overview

This appendix shows how the general-purpose registers are set when control is returned to the next sequential instruction following the execution of an API macro instruction. The table in the following section indicates which registers are left unchanged by the macro instruction, and which ones may be modified between the time the macro instruction is executed and control is returned to the application program. The table also shows the disposition of registers when any exit routine receives control. For a detailed discussion of exit routines, read Cisco IOS for S/390 Assembler API Concepts.

Table C-1 lists the notes referred to in the following section when an integer is enclosed in square brackets.

**Table C-1 Table Notes Description**

[1]	Register 13 must contain the address of an 18-word save area when the macro instruction is executed. The API uses this save area to save and restore registers 2-12 before returning to the application program.	
[2]	Register 15 contains a general return code that indicates the overall success or failure of the macro instruction. Based on the contents of register 15, register 0 is set as this table indicates:	
	<b>Register 15</b>	<b>Register 0</b>
	0	X'00'
	4	X'04'
	8	X'08'
	12	X'0C'
	16	X'10'
	20	X'14'
	<p>If an error occurred and the SYNAD or LERAD exit routine has been invoked, registers 0 and 15 contain the values returned by the exit routine. If an error occurred and no SYNAD or LERAD exit routine exits, the API sets register 15 to 4 (X'04') and returns a recovery action code in register 0. Fatal errors resulting in a general return code greater than 4 never invoke the SYNAD or LERAD exit routines.</p> <p>If an error occurs during the execution of a TERROR or TSTATE macro instruction, the SYNAD and LERAD exit routines are bypassed, and control is returned directly to the application program.</p>	
[3]	When the SYNAD or LERAD exit routine is invoked, registers 2-12 are restored to their original contents at the time the macro instruction causing the error was executed.	

# API Register Usage

**Table C-2 API Register Usage**

Instance	Contents Of General-Purpose Registers					
	Register 0	Register 1	Register 2-12	Register 13	Register 14	Register 15
On return from AOPEN and ACLOSE macro instructions	Zero if successful, specific error code otherwise	Address of APCB	Unmodified	Unmodified [1]	Unpredictable	General return code
On return from TCHECK macro instruction	Conditional completion code if successful, otherwise see note [2]	Address of TPL	Unmodified	Unmodified [1]	Unpredictable	General return code [2]
On return from TERROR macro instruction	Address of TEM if successful, otherwise see note [2]	Address of TPL	Unmodified	Unmodified [1]	Unpredictable	General return code [2]
On return from TSTATE macro instruction	Endpoint state (TSW) if successful, otherwise see note [2]	Address of TPL	Unmodified	Unmodified [1]	Unpredictable	General return code [2]
On return from TEXTLST macro instruction	Unmodified	Address of TXL	Unmodified	Unmodified	Unmodified	General return code (always zero)
On return from all other TPL- based macro instructions OPTCD=SYNC	Conditional completion code if successful, otherwise see note [2]	Address of TPL	Unmodified	Unmodified [1]	Unpredictable	General return code [2]
On return from all other TPL- based macro instructions OPTCD=ASYNC	Zero if accepted, otherwise see note [2]	Address of TPL	Unmodified	Unmodified [1]	Unpredictable	General return code [2]
On invocation of a SYNAD or LERAD exit routine	Recover action code	Address of TPL	Unmodified [3]	Unmodified [1]	API return address	Address of exit routine
On invocation of a TPL completion exit routine	Unpredictable	Address of TPL	Unpredictable	Unpredictable	API return address	Address of exit routine
On invocation of all other asynchronous exit routines	Unpredictable	Address of TPL	Unpredictable	Unpredictable	API return address	Address of exit routine

