IBM System/370 Extended Architecture

Interpretive Execution

Publication Number
SA22-7095-0

File Number
S370-01

PREFACE

The reader of this publication should become familiar with the IBM 370-XA Principles of Operation, SA22-7085. Terms and concepts are referred to in this publication which are explained in the Principles of Operation and which, for the most part, are not covered again in this publication.

First Edition (January 1984)

Changes are made periodically to the information herein; before using this publication in connection with the operation of IBM equipment, refer to the latest IBM System/370 and 4300 Processors Bibliography, GC20-0001, for the editions that are applicable and current.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Product Publications, Department B98, PO Box 390, Poughkeepsie, NY, U.S.A. 12602. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

CHAPTER 1. INTRODUCTION

This publication describes the interpretive-execution facility of the System/370 extended architecture (370-XA). The interpretive-execution facility serves the special purpose of improving the efficiency with which a specialized component of a control program can oversee the execution of programs in a virtual-machine environment. Execution of these programs is achieved through a combination of capabilities provided by the real machine while in the interpretive-execution mode and of services supplied by supporting programs. This publication describes the capabilities provided by the machine, which include the complete handling of many aspects of the architecture of an interpreted machine, as well as the presentation of status in convenient forms for support-program processing where complete machine handling is not provided.

The interpreted machine is called the guest. Facilities appropriate to the mode of the guest machine, either System/370 mode or 370-XA mode, are provided when the real machine is operated in the interpretive-execution mode. The interpreted machine is said to constitute a virtual machine. The term host is used to refer to the real machine and to the control program which both manages real machine resources and provides services to the guest program or machine. The guest and host machines execute guest and host programs, respectively.

The instruction START INTERPRETIVE EXECUTION (SIE) is provided for invoking the interpretive-execution mode. Execution of SIE causes the machine to enter the interpretive-execution mode and to commence execution of the guest program. Certain operations encountered in the guest cannot be performed in the interpretive-execution mode, and some may have been optionally designated to cause interpretive execution to be discontinued. In these cases, the machine exits from the interpretive-execution mode, SIE execution is completed, and the instruction in the host program that follows SIE is designated as the next instruction to be executed. This process is called interception, and it includes saving the state of the guest in an area called the state description and providing information about the reason for the interception.

The machine may also exit from the interpretive-execution mode by interrupting SIE execution. This would occur, for example, for a host I/O interruption. Execution of SIE follows the rules for interruptible instructions; that is, the parameters in the state description for continuing the execution of the guest are updated in such a way that when the old host PSW is loaded, causing SIE to be reexecuted, execution resumes at the interrupted point in the guest.

The operand of the SIE instruction is called the state description. The state description specifies the type of system to be interpreted, the host storage to be used to represent guest main storage, the contents of some of the program-addressable guest registers, the addresses of related control tables, bits for controlling the operation of optional facilities, areas for displaying information concerning an interception, and information about other aspects of the operation.

Two methods of representing guest main storage are provided. In the pageable-storage mode, guest main storage consists of a portion of a host address space. That is, guest absolute addresses, to which an offset is added, are treated as host virtual addresses. In the preferred-storage mode, guest main storage consists of the first portion of host main storage. That is, guest absolute addresses are treated unmodified as host absolute addresses. With both methods, guest address-translation mechanisms, including dynamic address translation and prefixing, are provided by the machine.

The various forms of storage have been described in other publications in terms of levels. Real (host) main storage has been called level-1 storage. In the pageable-storage mode, host virtual storage used to represent guest main (real) storage has been called level-2 storage. And guest virtual storage has been called level-3 storage. This publication does not use those terms.

Guest storage-key protection is provided, with guest keys set as the real keys for the real storage assigned to the guest. All other protection mechanisms are provided as well for the guest. In addition, host page protection is applicable to guest references to guest main storage in both the System/370 mode and 370-XA mode in the in the pageable-storage interpretive-execution mode.

A separate control area, called the reference-and-change-preservation (RCP) area, is provided for retaining the values of the change and reference bits

that are separately applicable to the guest and to the host for each individual 4K bytes of guest main storage. The RCP area is used only when the pageable-storage mode is specified. A true indication of the change-and-reference status is obtained by ORing the bits from the RCP area and the change and reference bits in the storage key. Only the bits in the storage key for host real storage are set by references to storage.

Timing facilities are, for the most part, fully interpreted for the guest. A guest time-of-day-clock value is obtained by adding a constant found in the state description to the value of the host TOD clock. The guest CPU-timer and guest clock-comparator values are specified in the state description, with the function of these facilities provided by the machine while it is in the interpretive-execution mode. Most instructions associated with the timing facilities are executed in the same manner in the interpretive-execution mode as when the machine is not in that mode. When the System/370 mode is specified, the interval timer is also maintained by the machine and very nearly reflects the time spent in the interpretive-execution mode. Guest interruption processing associated with guest timing facilities is performed in part by the machine. The operation of host timing facilities is not affected by operation of the machine in the interpretive-execution mode.

Most instructions are executed in the same manner in the interpretive-execution mode as when the machine is not in that mode. However, a number of instructions are given special treatment in the interpretive-execution mode. This includes instructions that are:

- Not interpreted. Interception is mandatory for these instructions. This group includes the I/O instructions and other infrequently occurring privileged instructions.

- Handled differently in the interpretive-execution mode This is generally because of special characteristics of the interpretive-execution environment. This group includes the storage-key-handling instructions.

- Interpreted as invalid when the System/370 interpretive-execution mode is specified. This group includes those instructions which are available in the 370-XA mode but not available in the System/370 mode.

- Conditionally interpreted depending on the setting of control bits.

Later sections of this publication describe the instructions for which special handling is provided.

Provisions for handling guest I/O include the following:

- Most I/O instructions cause interception, with information about the instruction provided in the state description at interception.

- A means is provided for recording a pending guest I/O interruption, and causing an interception when the guest PSW is enabled for I/O interruptions. A similar mechanism is provided for guest external interruptions.

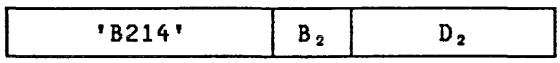
Interpretive execution can provide the effect of a guest shared-main-storage multiprocessing configuration. Multiple host processors can be simultaneously employed in this activity. Each guest CPU is defined by a separate state description, with each state description specifying the same guest main storage and RCP areas. An additional area, called the system-control area (SCA), is used for control purposes; the same SCA is designated by each state description of a guest multiprocessing configuration.

On entry to the interpretive-execution mode, the contents of the state description are subjected to a number of checks for errors and for consistency, and, during interpretive execution, checking is performed with respect to items fetched from the state description. Control is returned to the host program by means of a validity interception when an invalid condition is found.

The following sections give a detailed description of the operation of the machine in the interpretive-execution mode. These sections include the definition of the START INTERPRETIVE EXECUTION (SIE) instruction, a description of the format and contents of the state description, a discussion of guest storage, including the addressing mechanisms and the handling of storage keys, and a description of the special handling of some instructions.

START INTERPRETIVE EXECUTION

SIE $D_2(B_2)$ [S]



The CPU is placed in the interpretive-execution mode and performs the functions of the interpreted machine. The location designated by the second-operand address, called the state description, describes the interpreted machine. The interpreted machine is referred to as the guest. The CPU performing the interpretive execution by executing the START INTERPRETIVE EXECUTION instruction is referred to as the host.

Processing in the interpretive-execution mode continues until either an interception or host interruption occurs. Interception occurs when conditions are encountered in the guest which cannot be handled in the interpretive-execution mode or for which special assistance is supplied by a program in the host. Interception is accomplished by updating the state description to indicate the cause of the interception and to reflect the current state of the guest, by leaving the interpretive-execution mode, and by causing the execution of START INTERPRETIVE EXECUTION to be completed. Host interruptions are accomplished by updating the state description to reflect the current state of the guest, by leaving the interpretive-execution mode, and by setting the instruction address in the old host PSW to designate the interrupted START INTERPRETIVE EXECUTION instruction.

Host PSW bits 16 and 32 must be zero and one, respectively; otherwise, a special-operation exception is recognized, and instruction execution is suppressed.

The second-operand address is a real address, and accesses to the second-operand location are not subject to key-controlled protection. A specification exception is recognized, and instruction execution is suppressed, if the operand is not designated on a 256-byte boundary, or if bits 1-19 of the operand address are zeros or are equal to the host prefix. Host low-address protection is not applied to guest references to guest storage. (The location of guest storage within host storage is specified in the state description.)

Host PER for instruction fetching applies to the fetch of START INTERPRETIVE EXECUTION itself and is indicated, if applicable, as for other interruptible instructions. Host PER monitoring for general-register alteration is not applied, nor is host PER monitoring for storage alteration applied to the state description, the RCP area, the SCA, or the host storage area used for guest storage, while execution takes place in the interpretive-execution mode. Host PER does not apply to the execution of guest instructions.

Host tracing does not apply to guest operations.

A serializing function and a checkpoint-synchronization function are performed on entrance to and exit from the interpretive-execution mode.

Condition Code: The code remains unchanged.

Program Exceptions:

- Addressing (fetch and store operand 2)
- Operation (the instruction is not installed or provided on this CPU)
- Page translation (RCP area and guest storage area)
- PER (I-fetch)
- Privileged operation
- Segment translation (RCP area and guest storage area)
- Special operation (PSW bit 16 is not zero and PSW bit 32 is not one)
- Specification (operand-2 address bits 24-31 are not zeros, or bits 1-19 either are zeros or are equal to the prefix)
- Translation specification (RCP area and guest storage area)

Programming Notes

1. Addressing and page-protection exceptions for the RCP area and guest storage area are not reported to the host but instead are reported to the guest. Host translation exceptions for these areas are reported to the host.

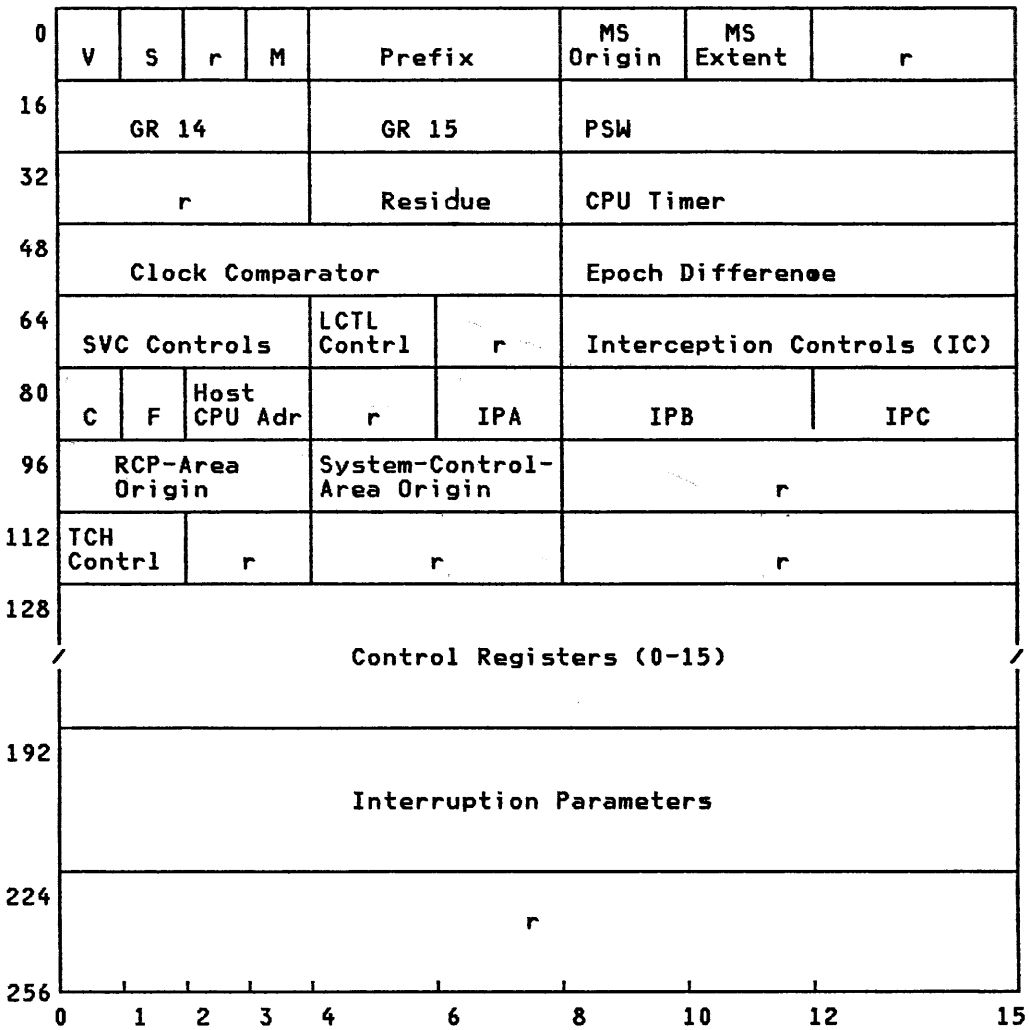
2. Low-address protection does not apply to the state description and RCP areas for SIE since these addresses are checked to ensure that they are not in the low-

address area. In the preferred-storage mode, the storage designated by the host prefix register is neither permitted to be within the guest storage area nor to match the address of the SCA. The RCP and SCA are designated by fields in the state description.

The START INTERPRETIVE EXECUTION (SIE) instruction specifies an operand in host real main storage called the state description. On entry to the interpretive-execution mode, the state description contains the initial state of the guest CPU. On exit from the interpretive-execution mode, the state of the guest CPU is stored in the state description. Various control and status fields concerned with the operation of interpretive execution are also provided in the state description.

After entry has been made to the interpretive-execution mode, changes to the state description by the channel or by another CPU do not necessarily affect the interpretive execution or the associated guest program. Also, fetch references to the state description by another CPU or by the channel do not necessarily obtain the current values of the associated guest registers and may result in unpredictable operation. An exception to this unpredictability rule occurs for the fields labeled "intervention requests" and "TCH control." The bits in these fields may be set to ones by an interlocked update by one CPU while the fields are concurrently being used to control interpretive execution in another CPU, with assurance that the change will be observed by the CPU in the interpretive-execution mode. The contents of these fields are said to be dynamically observed or recognized.

The figure "State Description" shows the format of the state description. A detailed specification for each of the fields follows the figure.



Explanation:

C	Interception code	M	Mode controls
F	Interception status	MS	Main storage
IPA	Instruction parameter A	r	Reserved
IPB	Instruction parameter B	S	State controls
IPC	Instruction parameter C	V	Intervention requests

State Description

CONTROL OF OPERATIONS

This section describes the controls for handling pending requests for action and the controls over the modes of interpretive execution.

INTERVENTION REQUESTS (V)

Intervention-request controls are provided to indicate that interception is requested so control can be returned to the host program for handling a guest I/O interruption, a guest external interruption, or an externally initiated request to stop.

When set to one, an intervention-request bit causes an interception under the conditions described below.

0 7

r: Reserved

P: 1 A stop interception is requested.
0 A stop interception is not requested.

I: 1 An I/O interception is requested. Interception occurs when a guest PSW I/O-mask bit is one. That is, in the BC mode at least one of the guest PSW bits 0-6 is one, and in the EC mode (System/370 and 370-XA) PSW bit 6 is one.
0 An I/O interception is not requested.

E: 1 An external interception is requested. Interception occurs when the guest PSW external mask (PSW bit 7) is one.
0 An external interception is not requested.

An intervention-request bit is inspected under these circumstances:

1. On entry to the interpretive-execution mode.

2. When the new PSW is loaded after completion of the execution of a guest LPSW, SSM, or STOSM instruction, and after a guest program or supervisor-call interruption.

   An intervention request is recognized after the validity of a new PSW is established and either before corresponding interruptions enabled by the new PSW are recognized or before an instruction is fetched.

3. Periodically when more than one CPU is in the configuration. Periodic inspection may or may not be provided when a configuration consists of only one CPU or when the host TOD clock is in the error, stopped, or not-operational state. The period between periodic inspections is referred to as the polling interval.

Inspection may also occur at other times.

The intervention-request bits may be changed to ones by one CPU while they are concurrently being used to control the interpretive-execution mode in another CPU, with the assurance that the change will be observed by the CPU in the interpretive-execution mode. However, changes to the contents of this field by I/O during interpretive execution may or may not be recognized and may be lost.

The intervention-request bits are not reset to zeros by the machine on the occurrence of the corresponding interception.

STATE CONTROLS (S)

The state-control byte provides a means of recording a requirement for a location-80 (interval-timer) external interruption, which nominally became due because the interval timer was decremented through zero.

The byte is set on exit from the interpretive-execution mode to indicate whether there is a pending guest interval-timer interruption. The byte is examined on entrance to the interpretive-execution mode and also when the guest becomes enabled for interval-timer interruptions.

0 7

T: 1 A guest external interruption for the interval timer is to be taken when the guest is enabled for interval-timer interruptions (PSW bit 7 and CR0 bit 24 are ones). This bit is examined only for a guest operation in the System/370 mode and is ignored in the 370-XA mode.

When the corresponding guest interruption, and associated mandatory interception which places the interruption parameters in the state description,

occurs, the T bit is reset to zero.

0 An interval-timer interruption is not requested.

r: Reserved

MODE CONTROLS (M)

For the System/370 mode, control is provided over whether the interval timer is active. A control is provided as well over whether the guest is executed in the System/370 or the 370-XA mode. Also, there is a control over whether guest main storage is represented as a portion of a host address space or is considered to be assigned to the corresponding portion of host absolute storage.

MM: 00 Invalid
01 The guest machine is executed in the System/370 mode.
10 The guest is executed in the 370-XA mode.
11 Invalid

G: 0 Pageable-storage mode
1 Preferred-storage mode

D: 0 Interval timer active
1 Interval timer inactive

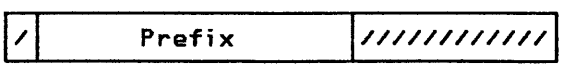
D is ignored when MM is not 01.

r: Reserved

STORAGE DEFINITION

This section describes how the location of guest storage within host storage is specified and how the amount of guest storage to be made available is specified. Provision of prefixing for the guest is also described.

PREFIX



This field contains the contents of the guest prefix register. Prefixing is considered to be installed for all guests. A validity interception is recognized if the prefix value does not designate a location within guest main storage. Bits 1 through 7 may or may not be ignored when the System/370 mode is specified; if they are not ignored, a nonzero value causes a validity interception.

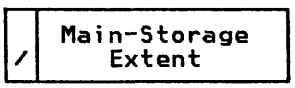
When both the System/370 mode and the pageable-storage mode are specified, prefixing is applied to guest dynamic-address-translation table references for both implicit translation and for LOAD REAL ADDRESS. When the preferred storage mode or the 370-XA mode is specified, prefixing may or may not be applied to dynamic-address-translation table references for implicit translation and for LOAD REAL ADDRESS.

MAIN-STORAGE ORIGIN

When the pageable-storage mode is specified, this field designates the virtual-storage location within the host address space specified by host control register 1 which represents guest absolute-main-storage address 0. The location corresponds to an address which is a multiple of 64K bytes. A nonzero value causes a validity interception when the preferred-storage mode is specified.

The same origin value within the same host address space may be designated in another state description for the purpose of interpreting a shared-main-storage multiprocessing configuration.

MAIN-STORAGE EXTENT



This field specifies the maximum number of contiguous bytes of main storage that are made available to the guest, in both the pageable-storage and preferred-storage modes. The extent is a multiple of 64K bytes, expressed as one less than the multiple to be allowed; that is, a value of n causes (n + 1) times 64K bytes to be made available for use by the guest.

A validity interception is presented when the preferred-storage mode is specified if either the state description or the host prefix would fall within the

storage available to the guest, that is, if the extent value is not less than the contents of bits positions 1-15 of the address of the state description and if it is not less than bits 1-15 of the host prefix value.

If the sum of the main-storage origin and the extent of guest main storage, without wrapping, exceeds the length specified for the segment table designated by host control register 1 when the pageable-storage mode is specified, a validity interception may or may not be presented.

In the preferred-storage mode, the storage allowed by the extent value may include locations which are not available in the real configuration. No check for this is performed on entry to the interpretive-execution mode.

GENERAL REGISTERS AND THE PSW

The contents of just two of the guest's general registers are obtained from the state description. Described below are the fields that contain the contents for these registers and the requirement to preserve the host contents for these registers. Also described is the field containing the guest PSW that is stored on exit from the interpretive-execution mode and that is used on entry to the interpretive-execution mode.

GR 14 AND GR 15

Contents for guest general registers 14 and 15 are obtained from these fields on entry to the interpretive-execution mode and are saved in these fields on exit. The contents of host general registers 14 and 15 are saved on entry to the interpretive-execution mode and are restored on exit. The contents of general registers 0-13 are not changed on entry to or exit from the interpretive-execution mode.

PSW

This field contains the contents of the guest PSW. The PSW specifies control and status information under which instruction execution for the interpreted machine starts on entry to the interpretive-execution mode. On exit from the interpretive-execution mode, the contents of the current guest PSW are stored in this field. When the exit from the interpretive-execution mode is due to interception for a guest interruption, the guest PSW stored in the state description is the PSW which would have been stored as the guest-interruption old PSW.

TIMING CONTROL

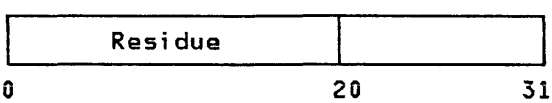
This section describes the two fields that contain the contents of the guest's CPU-timer and clock-comparator registers. Also described are the residue field, used in maintaining the guest interval timer, and the epoch field, used in providing a guest TOD-clock value.

Except for SET CLOCK, which is mandatorily intercepted, and for guest clock synchronization, which is not verified, when guest multiprocessing is invoked, all aspects of the timing facilities of a guest are provided for a guest under interpretive execution. Guest timing interruptions are handled as mandatory interceptions. Guest timing facilities are provided in addition to host timing facilities.

Between the instant that some condition causing exit from the interpretive-execution mode is initially recognized and exit from the interpretive-execution mode finally occurs, additional time may be reflected in the values placed in the state description for the guest CPU timer and interval timer. The condition of these values, and the enabled state of the guest, may indicate that an interruption is due. Interruptions and interceptions which become due during exit from the interpretive-execution mode are not necessarily recognized instead of the condition which initiated the exit, even if the new condition is ordinarily handled at a higher priority.

If a CPU is in the interpretive-execution mode when the host TOD clock accessed by that CPU enters the running state or is changed by another CPU sharing the same host TOD clock, the values of the guest interval timer, residue field, and guest CPU timer may be unpredictably changed. Also, guest interruptions due to the interval timer, CPU timer, and clock comparator may be lost, delayed, or erroneously generated.

RESIDUE COUNTER



The residue-counter field is used to maintain an accurate accumulation of CPU time spent in the interpretive-execution mode that has not yet been accounted for

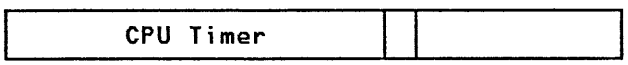
by decrementing the interval timer in guest storage. This field is ignored when the interval timer is not activated.

Bits 0-19 of the residue-counter field are considered to constitute an unsigned integer, with bit 19 representing microseconds. The use of bits 20-31 depends on the model; normally they should be set initially to zeros. The format of the residue field thus corresponds to bits 32-63 of the TOD clock.

On entry to the interpretive-execution mode, the interval timer in guest storage is adjusted by decrementing bit position 23 once for each integral multiple of 3,333 microseconds contained in the residue counter. The corresponding amount is deducted from the counter, with the remainder treated as an initial elapsed-time value. The adjustment may or may not occur when a preexisting interception-causing condition is recognized on entry to the interpretive-execution mode.

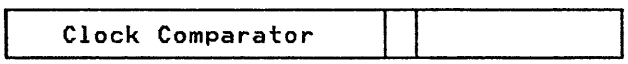
On exit from the interpretive-execution mode, the residue counter field contains the sum of the value of the residue counter at entry, plus the CPU time considered to have elapsed in the interpretive-execution mode since entry, minus 3,333 microseconds for each time the interval timer has been decremented in guest storage.

CPU TIMER



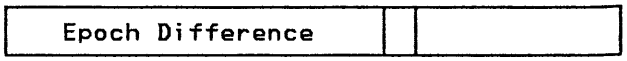
This field is provided for the guest CPU-timer value. The guest CPU timer is incremented only when the machine is in the interpretive-execution mode and at the same rate as the host CPU timer.

CLOCK COMPARATOR



This field contains the guest clock-comparator value.

EPOCH DIFFERENCE



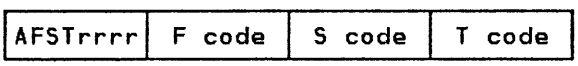
This field contains the difference between the guest and the host TOD-clock epochs, expressed as a 64-bit unsigned integer, with bit 51 representing microseconds, giving this field the same format as that of the TOD clock. A guest TOD-clock value is produced by the addition of the epoch difference to the value of the host TOD clock, with a carry out of bit 0 ignored. Bits representing higher resolution than provided by the host clock may or may not be lost.

INTERCEPTION CONTROL

The term "conditional interception" refers to functions which are executed for the guest unless a specified condition is encountered which causes control to be returned to the host by the process called interception. Described below are some of the controls which can cause interception when the related function is handled for the guest.

SVC CONTROLS

This field provides for optionally intercepting either every guest SUPERVISOR CALL (SVC) or just those for which the SVC code matches one of up to three given values.



A: 1 Instruction interception occurs for every guest SVC instruction.
0 Interception occurs only for guest SVC instructions for which the SVC code is equal to a control code enabled for matching by one or more of the next three bits.

F: 1 Instruction interception occurs for a guest SVC instruction for which the SVC code is equal to the contents of the F-code byte.
0 A match with F code is not recognized.

S: Same as for F but applies to S code.

T: Same as for F but applies to T code.

F code: A one-byte value to be compared with the contents of the I field

of SVC instructions if matching is allowed by a one in bit position F.

S code: Same as for F code but with respect to the S bit.

T code: Same as for F code but with respect to the T bit.

LCTL CONTROLS

The bits of this field are numbered, starting from the left, to correspond to control-register numbers. When a bit is one and a guest LOAD CONTROL (LCTL) designates the corresponding control register in the range of registers to be loaded, instruction execution is suppressed, and instruction interception is recognized.

INTERCEPTION CONTROLS (IC)

A bit value of one causes interception when the associated function is encountered in the guest.

Bit Position	Associated Function
0	Invalid operation
1	Program interruption (privileged-operation exception)
2	Program interruption (other than mandatory, bit-0, and bit-1 cases)
4	TEST AND SET, for CC1
5	COMPARE AND SWAP, for CC1
6	COMPARE DOUBLE AND SWAP, for CC1
7	INVALIDATE PAGE TABLE ENTRY
9	LOAD PSW
10	PURGE TLB
11	SET SYSTEM MASK
13	STORE CONTROL
14	STORE THEN AND SYSTEM MASK
15	STORE THEN OR SYSTEM MASK
16	STORE CLOCK
17	INSERT STORAGE KEY and INSERT STORAGE KEY EXTENDED
18	SET STORAGE KEY and SET STORAGE KEY EXTENDED
19	RESET REFERENCE BIT and RESET REFERENCE BIT EXTENDED
20	PROGRAM CALL
21	PROGRAM TRANSFER
22	TEST PROTECTION
23	LOAD ADDRESS SPACE PARAMETERS
25	SET CPU TIMER and STORE CPU TIMER
26	SET CLOCK COMPARATOR and STORE CLOCK COMPARATOR

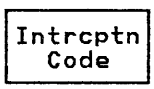
Unassigned bits are reserved.

INTERCEPTION PARAMETERS

Exit from the interpretive-execution mode takes two forms. One is exit by interruption, after which the causative condition can be handled by a host program. The second is exit by interception and causes execution of instructions to resume in the host with the instruction following the SIE instruction. Normally, interception results from a function or condition encountered in the guest which is to be treated by a host program. To allow the condition to be efficiently analyzed by the host program, various descriptive information about the condition is stored in the state description on exit from the interpretive-execution mode (that is, at interception). The information provided is described in this section.

INTERCEPTION CODE (C)

The code placed in this field during interception indicates the reason for the interception. The contents of this field are changed only at interception.



Code	Interception Condition Applicable to the Guest
04	Instruction
08	Program interruption
12	Instruction and program interruption
16	External request
20	External interruption
24	I/O request
28	Wait state
32	Validity
40	Stop request
44	Operation exception

Unassigned codes are reserved.

The definitions of the conditions under which each code is used are given below.

Code 04 (Instruction)

Instruction interception is indicated both for instructions for which interception is mandatory and for instructions for which the option to intercept is chosen. The conditions under which instruction interception are recognized apply to the target instruction of EXECUTE.

Some instructions are valid either in the 370-XA mode or in the System/370 mode, but not in both modes. When instruction interception is either mandatory or conditional in the mode for which the instruction is valid, depending on the model, instruction interception may be recognized unconditionally in the invalid mode rather than an operation exception.

Guest instruction execution is suppressed when code 04 is given.

Code 08 (Program Interruption)

Interception is mandatory, and the interception code is set to 08, for guest program interruptions caused by these exceptions:

- Protection
- Addressing
- Specification
- Special operation

Otherwise, program interruptions cause an interception only if they are of a type for which the corresponding interception control, bit 1 or bit 2, is set to one.

A protection exception may be due to a key mismatch, guest page or segment protection, guest low-address protection, or host page protection. (These conditions are also used to set the condition code for a guest TEST PROTECTION instruction.)

An addressing exception may be due to a host segment-table entry or page-table entry at an invalid address, or an invalid guest or host absolute address.

In all cases of interception, the parameters of the interruption are placed in the state description at interception. The PSW placed in the state description is the current guest PSW or, in the case of a guest program interruption, the guest old PSW.

Code 12 (Instruction and Program Interruption)

This code is indicated only for instructions that cause an interception after completing execution and for which a PER event has also occurred. The PER event results in the placement of the corresponding program-interruption parameters in the state description. Code 12 is indicated regardless of whether the interception control for the program interruption is also set to cause interception. These conditions apply to the instructions COMPARE AND SWAP, COMPARE DOUBLE AND SWAP, and TEST AND SET, for which conditional interception is recognized when condition code 1 is set.

Code 16 (External Request)

Interception is due to bit 7 of the intervention-request field being one when PSW bit 7 is one.

Code 20 (External Interruption)

Interception is mandatory for guest external interruptions. Guest external interruptions are generated for the following facilities:

- Guest clock comparator
- Guest CPU timer
- Guest interval timer

The parameters of the interruption are placed in the state description.

Code 24 (I/O Request)

Interception is due to bit 6 of the intervention-request field being one either when any of bits 0-6 of a BC-mode PSW is one or when bit 6 of an EC-mode PSW is one.

Code 28 (Wait State)

Interception is due to bit 14 of the guest PSW being one when no other intervention, interruption, or interception condition meets the conditions for recognition.

Code 32 (Validity)

Interception for validity is or may be indicated for the following conditions:

1. A mode is specified that is invalid or not installed on the model. A validity interception may also be presented when a model-dependent limit is exceeded.

2. For the pageable-storage mode, part of guest main storage maps to a host address-space location that appears to exceed $2^{31}-1$ (that is, guest main storage appears to wrap in the host address space).

   Optionally, a validity interception may also be recognized on entry to the interpretive-execution mode when guest main storage exceeds the length of the host virtual address space, as specified in the segment-table length in control register 1, or an invalid format is specified in host control register 0. Alternatively, these conditions may be recognized instead as host translation exceptions when a translation is performed.

3. For the pageable-storage mode, the origin of the RCP area is 0, or wrapping of the RCP area is implied by the amount of guest main storage specified, or the implied size of the RCP area exceeds the host address-space length as specified in the segment-table length in control register 1.

4. For the preferred-storage mode, guest main storage includes the state description or host real page 0, or the main-storage origin is not zero.

5. Guest real page 0, designated by the prefix value in the state description, is not within guest main storage. For the System/370 mode, bits 1-7 of the prefix value may or may not be examined. When they are examined, nonzero contents result in a validity interception.

6. An access exception is encountered when referencing guest real page 0 to perform a program or supervisor-call interruption, or to update the interval timer. In the pageable-storage mode, host translation exceptions are allowed, alternatively, to be recognized as such instead, rather than resulting in a validity interception, in which case the guest operation is nullified. The existence of an access exception may or may not be recognized on entry to the interpretive-execution mode and result in a validity interception.

7. Either during execution of INVALIDATE PAGE TABLE ENTRY or on entry to the interpretive-execution mode, bits 1-19 of the SCA match the host prefix value or are zero when bits 20-27 are not zero, indicate an invalid address, or, in the preferred-storage mode, designate storage within the guest extent.

State-description information for which validity checking is defined is checked each time it is fetched from the state description. This may occur at times other than on entry to the interpretive-execution mode.

A validity interception which is caused by an inability to access guest real page 0 may result in the loss of information for these items:

- Interruption or interval-timer information to be placed in guest real page 0
- The T bit in the state-control byte
- The residue field
- The guest PSW placed in the state description

Before any reference is made to guest main storage on entry to the interpretive-execution mode, a validity check is made of the mode control and, in preferred-storage mode, for inclusion of the state description and host real page 0 within guest main storage. Interception for invalid mode controls takes precedence over all other reasons for interception.

Code 40 (Stop Request)

Interception is due to bit 5 of the intervention-request field being one.

Code 44 (Operation Exception)

An operation code which is unassigned in both the 370-XA and System/370 modes is handled as follows:

1. A guest program interruption for an operation exception is taken unless interception-control bit 0 is one.

2. An interception is recognized when interception-control bit 0 is one and the following conditions occur:

   a. Interception code 44, operation exception, is given.

b. The first two bytes of the instruction are placed in instruction parameter A (IPA). Instruction parameter B (IPB) and instruction parameter C (IPC) are set to zero. Instruction execution is suppressed. The instruction-length code (ILC) in a guest BC-mode PSW is not necessarily valid. Interruption parameters are not generated, except that the interruption-code portion of a BC-mode PSW may or may not specify operation exception.

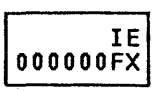
c. The EX bit (bit 7) of interception status is set to specify whether the unassigned operation code is the target of an EXECUTE instruction.

d. The IF bit (bit 6) of interception status is set to specify whether an I-fetch PER event is applicable.

An operation code which (1) is assigned in one mode but not in the other and (2) is subject to either conditional or mandatory interception in the mode for which it is defined, is handled in one of two ways in the mode for which it is invalid. It may be handled as an operation exception, as described above, or, depending on the model, as an instruction interception. These options also apply to operation codes that are assigned but which are not installed.

INTERCEPTION STATUS (F)

This field specifies, for interception for a guest instruction, whether an I-fetch PER event is applicable and whether the guest instruction causing the interception is the target of a guest EXECUTE instruction. Zero is stored in this field for interception codes other than 04, 12, and 44.



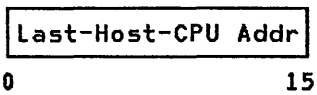
IF: 1 When the bit is one, an instruction-fetch PER event is applicable to the instruction for which interception has occurred. When the intercepted instruction is the target of an EXECUTE instruction, the instruction-fetch event is indicated when it is applicable to either the EXECUTE instruction or the target instruction or both. This condition is indicated only with interception codes 04 and 44.

0 When the bit is zero, an instruction-fetch PER event has not been recognized. The bit is always set to the appropriate value for interception codes 04 and 44; the bit may or may not be set for interception code 12.

EX: 1 Interception applies to an instruction that is the target of an EXECUTE instruction. The condition is indicated with interception codes 04, 12, and 44.

0 The instruction for which interception is indicated is not the target of an EXECUTE instruction.

LAST-HOST-CPU ADDRESS

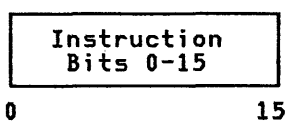


On models which retain guest TLB entries after exit from the interpretive-execution mode, this field contains the 16-bit host CPU address of the CPU that was most recently in the interpretive-execution mode for this state description, as would be obtained by a host STORE CPU ADDRESS instruction. On these models, on entry to the interpretive-execution mode, the contents of this field are compared with the host CPU address. If the field matches, then existing guest-type TLB entries can be used. If the field does not match, then existing guest-type TLB entries for this guest are purged. On models which purge guest TLB entries on exit from the interpretive-execution mode, the contents of this field are unpredictable.

The value FFFF hex can be used by the program to cause copies of information retained internally to be discarded.

This field may be updated by the machine on entry to the interpretive-execution mode.

INSTRUCTION PARAMETER A (IPA)



When the code is 04, 12, or 44, this field contains the first two bytes, bits 0-15, of the instruction causing the interception, as modified by EXECUTE if applicable. Zero is stored in this field

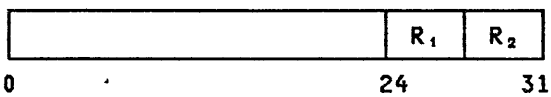
for interception codes other than 04, 12, and 44.

INSTRUCTION PARAMETER B (IPB)

Addressing information is placed in this field in conjunction with intercepting for some instructions. Zero is stored for all interception codes other than 04, and when the instruction for which the interception is indicated is not defined to provide an addressing parameter. The formats that are used are shown below.

In the first case, an address is formed according to the current guest addressing mode, as the sum D(B), and then is placed in this field as a 31-bit value. This format is used for instruction interception for RX, RS, SI, and S format instructions, and for the first operand address for SSE format instructions.

In the second case, the fourth byte of an RRE-format instruction, which specifies R_1 and R_2, is placed in the fourth byte of this field. The contents of bit positions 0-23 may or may not change.



INSTRUCTION PARAMETER C (IPC)

Addressing information is placed in this field in conjunction with interception for an SSE-format instruction. Zero is stored in this field for all interception codes other than 04, and when the instruction for which the interception is indicated is not defined to provide this addressing parameter. The guest second-operand address is formed according to the current guest addressing mode, as the sum D(B), and then is placed in this field as a 31-bit value.

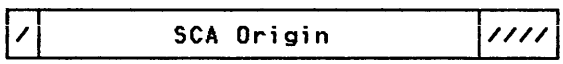
ORIGINS OF RELATED TABLES

Depending on the controls specified, additional tables are used to contain control information. The origins of these tables are obtained from the state description, from the fields described below.

RCP-AREA ORIGIN

The contents of this field, with 12 zero bits appended at the right, specify the host virtual address of an area used to retain change and reference information. This address is on a 4K-byte page boundary within the address space specified by host control register 1. This field is ignored when the preferred-storage mode is specified. The area contains one byte for each 4K bytes of guest main storage, with the index of a byte within the area equal to the index of the corresponding 4K-byte block within guest main storage. A validity interception is recognized if the origin field specifies zero, or if an addressing overflow (wraparound) would be encountered for any byte of the area. A validity interception may or may not be recognized if the area would exceed the length of the host address space.

SYSTEM-CONTROL-AREA ORIGIN



The origin of the SCA is contained in bits 1-27. With four zeros appended on the right, the origin is treated as a real address. Bit 0 and bits 28-31 are ignored.

A validity interception is recognized, either during execution of INVALIDATE PAGE TABLE ENTRY (IPTE) or on entry to the interpretive-execution mode, if bits 1-19 match the host prefix value or are zero when bits 20-27 are not zero, if they contain an invalid address, or, in the preferred-storage mode, if they designate storage within the guest extent.

IPTE is executed without broadcasting when bits 1-27 are zeros. When bits 1-27 are not zeros, broadcasting for guest IPTE instructions is performed, removing all guest TLB entries formed

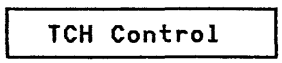
under a matching SCA-origin value on all CPUs in the configuration.

Bit 0 of the SCA, the IPTE-interlock-control (K) bit, is set to one at the beginning of IPTE execution, and to zero on completion of execution, by means of an interlocked update. When the initial value of the bit is one, instruction execution is suppressed, and instruction interception is recognized.

OTHER CONTROLS

This section describes a control field related to I/O operations for the guest.

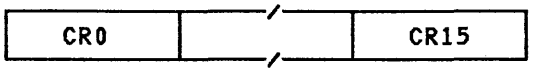
TCH CONTROL



The bits of this field are numbered, starting from the left, to correspond with channel addresses 0-15. TEST CHANNEL (TCH) is executed by setting condition code 0 unless (1) a privileged-operation exception is recognized, or (2) interception is recognized because the bit corresponding to the designated channel is one or because a channel number larger than 15 is designated. Instruction execution is suppressed for interception, and $D_2(B_2)$ is provided in instruction parameter B.

This field may be changed dynamically by an interlocked update by another CPU with the assurance that the change will be observed by a CPU in the interpretive-execution mode for the state description containing the changed bytes.

CONTROL REGISTERS



This area contains the 16 control registers for the guest on entry to and on exit from the interpretive-execution mode. The registers appear in ascending order of register numbers, starting with control register 0.

INTERRUPTION PARAMETERS

For all guest external interruptions, and for guest program interruptions associated with an interception for which the general definition specifies storing at locations 128 through 159, storing in the interpretive-execution mode occurs instead in the state description at locations 192 through 223. Locations 192 through 223 of the state description are formatted identical to locations 128 through 159 of the guest. Storing is appropriate to the mode of the guest PSW, which is either the 370-XA mode, System/370 BC mode, or System/370 EC mode. Storing occurs in conjunction with interception codes 08, 12, and 20.

ADDRESSING

Two mechanisms are provided for representing guest main storage. One mechanism represents guest main storage as a contiguous portion of the host virtual address space designated by host control register 1. The origin of guest main storage within the host address space is provided in the state description. This is called the pageable-storage mode.

The second mechanism represents guest absolute addresses as corresponding to host absolute addresses; host dynamic address translation (DAT) and host prefixing are not applied. This is called the preferred-storage mode.

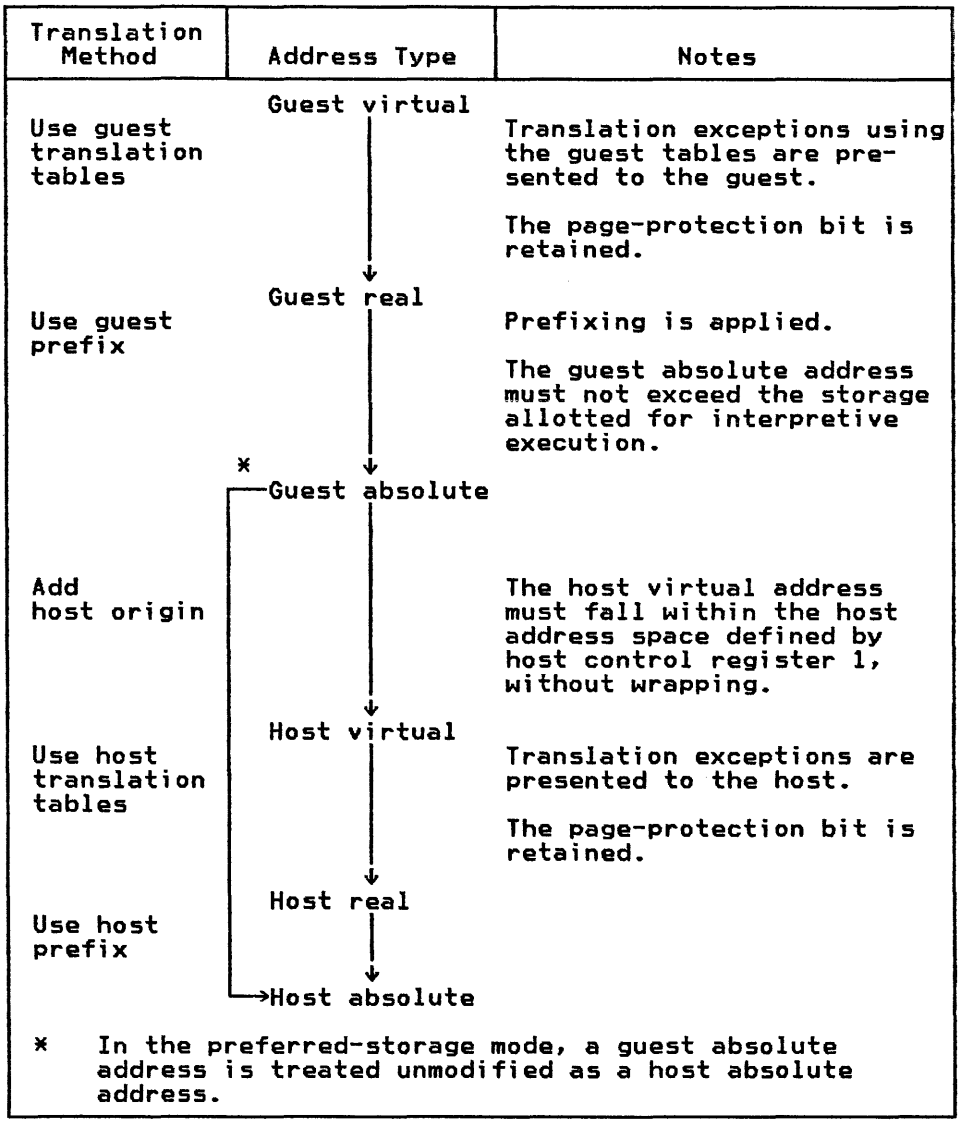
In both cases, the amount of guest main storage to be provided is specified in the state description.

The guest may also use DAT. A guest virtual address is first translated to a guest real address, using guest translation tables. Guest prefixing is then applied to produce a guest absolute-main-storage address. A guest absolute-main-storage address is treated as the corresponding host absolute-main-storage address when the preferred-storage mode is specified. When the pageable-storage mode is specified, a guest absolute address is converted to a host virtual address by adding to it the host virtual address at which guest storage begins in the host address space (from the main-storage-origin field in the state description). The host address is then translated by using host translation tables designated by host control register 1. Finally, host prefixing is applied. Host-DAT translation exceptions for host addresses result in host interruptions, while guest-DAT translation exceptions for guest addresses result in guest interruptions. All addressing and protection exceptions are considered guest exceptions. The address-translation mechanisms are illustrated in the figure "Translating Addresses for Interpretive Execution."

Guest DAT table entries are designated by guest real addresses. In the pageable-storage mode, these table-entry addresses are converted to host virtual addresses, which are then translated to host real addresses. When the System/370 mode is specified, prefixing is applied to DAT table-entry addresses in the pageable-storage mode, but may or may not be applied in the preferred-storage mode for references for implicit address translation and by the LOAD REAL ADDRESS instruction. In the 370-XA mode, prefixing may or may not be applied to DAT table-entry addresses in both the pageable-storage mode and preferred-storage mode. In all cases, however, prefixing applies to the operand address of the INVALIDATE PAGE TABLE ENTRY instruction in the same way as when the machine is in interpretive-execution mode.

Depending on the model, the support of DAT in the System/370 mode may or may not include support for the 1M-byte-segment size and for the 2K-byte page size. All models support the 64K-byte segment size and 4K-byte page size. When the page size is 4K bytes, the effective page-frame real address is always formed by appending page-table-entry bits 13-14, giving a 14-bit value which allows 26-bit real-storage addressing for a guest in the System/370 mode.

Guest-type TLB entries may be formed to provide a rapid means of translating guest storage addresses to host real storage addresses. These entries may or may not be retained when the CPU exits from the interpretive-execution mode. The entries thus formed are associated with both guest and host translation parameters, and with the host real address of the state description, the host real address of the system-control area, and the host CPU address on which the guest was last executed. Dispatching a guest on a different host CPU causes guest TLB entries to be purged automatically that were previously formed under a matching state-description address. A host PURGE TLB (or INVALIDATE PAGE TABLE ENTRY) must be issued when the host program changes any of the host or guest translation parameters, reassigns storage assigned to a state description or a system-control area, or changes between the preferred-storage and the pageable-storage modes.



Translating Addresses for Interpretive Execution

ACCESS CONTROLS

Guest accesses to guest storage, in all modes of interpretive execution, are subject to key-controlled protection in the usual way, by using the real storage keys. Guest accesses are also subject to guest low-address protection, to guest segment protection when the System/370 mode is specified, and to guest page protection and guest fetch-protection override when the 370-XA mode is specified.

In addition, host page protection applies to guest references in the pageable-storage mode. This checking applies to guests in both the System/370 and 370-XA modes and includes the checking of guest references to guest main storage and the checking of implied references to the RCP area. Disallowed storing causes a guest program interruption for a protection exception. However, disallowed storing into guest real locations 0-4095 to perform a guest program or supervisor-call interruption, unless the operation is nullified, or to update the interval timer, results in a validity interception.

INSERT STORAGE KEY, INSERT STORAGE KEY EXTENDED, SET STORAGE KEY, and SET STORAGE KEY EXTENDED are interpreted by fetching and storing, respectively, the access-control and fetch-protection bits in the real storage key. In the pageable-storage mode, additional information concerning the change and reference bits is kept in a byte in a table called the RCP area.

Obtaining the host main-storage address associated with the guest real storage address specified by the key-handling

instructions -- and thereby specifying the real storage key to be used -- requires translating a host address in the pageable-storage mode. The access to the RCP area also involves translating a host address. It is unpredictable whether access exceptions for the RCP area are recognized before access exceptions for the real key are recognized.

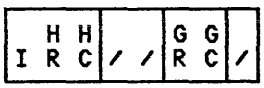
In the interpretive-execution mode, no checking occurs with respect to the host PSW key. No storage-key protection is applicable to references to the state description or to references to the RCP area.

CHANGE-AND-REFERENCE HANDLING

When the pageable-storage interpretive-execution mode is specified, a page frame of host real storage may be the object of a translation for guest purposes and for host purposes simultaneously. The setting or resetting of the real reference and change bits through the use by the guest of RESET REFERENCE BIT, RESET REFERENCE BIT EXTENDED, SET STORAGE KEY, AND SET STORAGE KEY EXTENDED in conjunction with managing guest storage would leave the bits in an incorrect state for managing host real storage. Conversely, changes to these bits by host actions result in the real indicators not correctly indicating the status of guest real storage. For this reason, three sets of bits are maintained. One set is the real-storage set. The bits in this set are the only indicators affected by storage accesses, including accesses by subchannels. The other two sets are maintained in the RCP-area table. One set, the host RCP set, is used to retain status for the host when the real set is changed by guest key-handling operations. The other set, the guest RCP set, is provided to retain status for the guest when the real set is changed by host operations. True status is obtained by ORing real status with the appropriate set in the RCP area.

The RCP area consists of one byte for each 4K bytes of guest storage. The RCP area is designated by an address in the state description and resides in the host address space designated by host control register 1. An RCP area is provided only when the pageable-storage mode is specified. The same area can be designated in more than one state description when a guest shared-main-storage multiprocessing configuration is interpreted. The byte associated with a particular 4K-byte block is identified by an index which equals the offset of the corresponding 4K-byte block in guest main storage, beginning with guest absolute zero. Only as many bytes are provided as there are guest 4K-byte main-storage blocks.

Five bits of each byte in the RCP area are used. The first bit is an interlock control which is set by each key-handling instruction in the pageable-storage mode to provide exclusive use of the byte. The next two bits are used to save the host reference and the host change indications, respectively. Similarly, the next two bits are provided for saving the guest reference and change indications. The format of a byte of the RCP area is illustrated below:



I RCP interlock control
HR RCP host reference indicator
HC RCP host change indicator
GR RCP guest reference indicator
GC RCP guest change indicator
/ Bits 3, 4, and 7 are ignored and unchanged

The following facilities, in addition to the basic computing functions defined in the IBM System/370 Principles of Operation, constitute the System/370 facilities provided as standard under interpretive execution:

- Universal instruction set (includes standard instruction set, floating point, and decimal)
- Extended-precision floating point
- Translation (with 1M-byte segments optional and 2K-byte pages optional)
- Extended (26-bit) real addressing
- CPU timer and clock-comparator facility
- Conditional swapping
- PSW-key handling
- Multiprocessing
- Storage-key 4K-byte block
- Extended facility: IPTE, low-address protection, and TEST PROTECTION
- TEST BLOCK
- Branch and save
- Dual address space
- Storage-key-instruction extensions
- Segment protection

Instructions defined for the 370-XA mode and not defined for the System/370 mode cause an operation exception to be recognized when the System/370 mode is specified for the guest.

The basic 370-XA architecture is provided as standard when the 370-XA mode is specified for the guest.

All control registers are considered to be installed in the guest machine.

Optional facilities may or may not also be offered in the interpretive-execution mode. Depending on the facility, full interpretive execution may be provided, or only portions may be provided, with the rest to be provided by programmed means. Several facilities are more fully described in the following sections.

GUEST INSTRUCTION PROCESSING

Most guest instructions are fully interpreted, with execution proceeding from one guest instruction to the next in a normal fashion. However, some guest instructions require treatment by the host program. Control is returned to the host program in these cases by a process called interception. Interception occurs for some instructions after they have been completed. However, most instructions for which interception occurs are suppressed prior to interception. The two general reasons for interception are (1) interception has been selected by a bit set to one in the state description associated with the instruction, or (2) the instruction is not interpreted under the existing conditions, in which case interception is mandatory. Useful parameters of the guest instruction are usually provided at interception.

The privileged-operation, specification, and special-operation exceptions can be recognized ahead of instruction interception. For the conditionally interceptible instructions, all other applicable exceptions can also be recognized ahead of interception.

Following is a list of instructions for which interception is mandatory in the modes in which they are valid:

CLEAR CHANNEL
CLEAR I/O
CONNECT CHANNEL SET
CLEAR SUBCHANNEL
DIAGNOSE
DISCONNECT CHANNEL SET
HALT DEVICE
HALT I/O
HALT SUBCHANNEL
MODIFY SUBCHANNEL
RESET CHANNEL PATH
RESUME I/O
RESUME SUBCHANNEL
SET ADDRESS LIMIT
SET CHANNEL MONITOR
SET CLOCK
START INTERPRETIVE EXECUTION
SIGNAL PROCESSOR
START I/O
START I/O FAST RELEASE
SET PREFIX
START SUBCHANNEL
STORE CPU ADDRESS
STORE CHANNEL PATH STATUS
STORE CHANNEL REPORT WORD
STORE CHANNEL ID
STORE CPU ID

STORE PREFIX
STORE SUBCHANNEL
TEST BLOCK
TEST I/O
TEST PENDING INTERRUPTION
TEST SUBCHANNEL

INTERACTIONS OF FACILITIES

Exit from the interpretive-execution mode is part of the unit of operation in which an interception or interruption condition is detected.

The machine exits from the interpretive-execution mode before entering the stopped state, with the host PSW address designating the SIE instruction. Thus, when the machine is in the interpretive-execution mode, the host stop function is handled like a host interruption, except that there is no exchange of host PSWs. Similarly, the machine exits from the interpretive-execution mode before completing a reset operation. If a CPU reset is initiated while the machine is in the interpretive-execution mode, the contents of the state description and the host registers, including the host PSW and other host registers (such as the prefix register, CPU timer, and clock comparator) are unpredictable. If the host rate control is in the instruction-step position when the host PSW address designates SIE, entry is made to the interpretive-execution mode to execute a guest unit of operation, a guest interruption, or an interception, or to update the guest interval timer. Then the machine exits from the interpretive-execution mode. On entry to the interpretive-execution mode, a guest interruption has higher priority than the execution of a guest unit of operation. The definition of a unit of operation applicable to the guest is honored in the interpretive-execution mode. Host asynchronous interruptions are recognized between guest units of operation. Machine-check interruptions are presented to the host after exit from the interpretive-execution mode is accomplished.

The START INTERPRETIVE EXECUTION (SIE) instruction is nullified if a machine check is reported while the checkpoint-synchronization action which is part of the entry to the interpretive-execution mode is being performed. If a machine check occurs during execution of a guest checkpoint-synchronization action in the interpretive-execution mode, the guest state and the values placed in the state description are unpredictable. The checkpoint-synchronization action on exit from the interpretive-execution mode is taken after the SIE instruction execution is completed.

Interruptions and interceptions which become due during exit from the interpretive-execution mode are not necessarily recognized instead of the condition which initiated the exit from the interpretive-execution mode.

A host I-fetch PER event which applies in the interpretive-execution mode is handled as for interruptible instructions. That is, a host program interruption for PER is taken (1) after SIE is completed because of interception, or (2) PER is indicated concurrently with another program interruption, or (3) exit from the interpretive-execution mode is accomplished by nullifying SIE -- as if a host I/O, external, or machine-check interruption were being taken -- except that a program interruption for PER is taken instead.

SYSTEM/370 EXTENDED FACILITY

The System/370 extended facility is provided when the System/370 mode is specified for the guest, subject to the adjustments in the methods of operation described below. The handling of the capabilities provided by the facility is summarized in the figure "Handling of System/370 Extended Facility."

Handling	Capability
F	Common-segment bit
F	Low-address protection
F(IC)	INVALIDATE PAGE TABLE ENTRY instruction
F(IC)	TEST PROTECTION instruction
F(SP)	Four lock-handling instructions
F(M)	Six tracing instructions
F(SP)	FIX PAGE instruction
F(SP)	SVC ASSIST instruction
F	Virtual-machine extended-facility assist*

Explanation:

- F — Fully provided in the interpretive-execution mode.
- F(IC) — Execution is provided in the interpretive-execution mode, with an option to intercept instead.
- F(SP) — Execution in the interpretive-execution mode in some or all cases may be according to the simplified-execution path definition.
- F(M) — Execution is provided in the interpretive-execution mode, subject to methods of operation described in this section.
- * — This is the only virtual-machine assist provided in the interpretive-execution mode.

Handling of System/370 Extended Facility

The following considerations apply in the interpretation of this facility:

- The four-bit number from bit positions 12-15 of the halfword at location 414 hex is placed in the channel-set-number field of the trace-table entry for TRACE I/O INTERRUPTION when the instruction is executed in the interpretive-execution mode.
- The extended-facility-trace instructions obtain the value used as the CPU address from logical address 31B hex in the interpretive-execution mode.

3033 EXTENSION FEATURE

SVC ASSIST and FIX PAGE: The extensions to these functions are optionally provided in full in the interpretive-execution mode. The full function may be provided only in certain circumstances, such as for the preferred-storage mode only. The simplified execution path is provided when the full function is not provided. It depends on the model whether, and under what circumstances, the full function, rather than the simplified execution path, is provided.

The VM/370 assists for the extensions are not provided in the interpretive-execution mode.

Page-Fault Assist: This function is optionally provided in the interpretive-execution mode, though possibly only in certain circumstances, such as only for the preferred-storage mode. Bit 13 of control register 0 is ignored when the function is not provided. It depends on the model whether, and under what circumstances, the function is provided.

The VM/370 assist for this function is not provided in the interpretive-execution mode.

ADD FRR: This function is provided in the interpretive-execution mode when the function is provided by the native machine, that is, by the machine when it is not in interpretive-execution mode. It depends on the model whether this function is provided in the interpretive-execution mode when the function is not provided by the native machine.

The VM/370 assist for this function is not provided in the interpretive-execution mode.

Dual-Address-Space (DAS) Facility: DAS is provided in the interpretive-execution mode. The VM/370 assist for this function is not provided in the interpretive-execution mode.

PREFERRED-MACHINE ASSIST

The preferred-machine-assist facility is not provided in the interpretive-execution mode. The operation code E616 hex causes an operation exception to be recognized.

OPTIONAL IMPLEMENTATIONS

It depends on the model whether optional functions, and assists for which simplified execution paths are available, are provided in full in the interpretive-execution mode. Functions may be provided in full, for example, only in the preferred-storage mode.

IBM System/370
Extended Architecture
Interpretive Execution

READER'S COMMENT FORM

Order No. SA22-7095-0

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name, company, mailing address, and date:

_____

_____

_____

_____

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the front cover or title page.)

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure-sensitive or other gummed tape to seal this form.

SA22-7095-0

Reader's Comment Form

Cut or Fold Along Line

Fold and tape | Please Do Not Staple | Fold and tape

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department B98
P.O. Box 390
Poughkeepsie, New York 12602

Fold and tape | Please Do Not Staple | Fold and tape

IBM

IBM System/370 Extended Architecture Interpretive Execution (File No. S370-01) Printed in U.S.A. SA22-7095-0