EventMask Objects

Introduction

The **EventMask** determines which event types will be sent from the controller to the Notify object. By setting bits in the EventMask, specific events can be sent to the Notify object when they occur or by clearing specific bits, the events can be ignored. By default, no Events will occur until the EventMask is set.

The Event, EventMgr, and Notify objects are used with the EventMask to handle controller events.

Data Types

MPIEventMask	An array of unsigned longs

Macros

mpiEventMask**ALL** / meiEventMask**ALL**	Set all events within the event mask.
mpiEventMask**AND_ASSIGN**	Assign dst all events masked by both *src* and *dst*.
mpiEventMask**ASSIGN**	Assign the value of event mask *src* to the event mask *dst*.
mpiEventMask**AXIS** / meiEventMask**AXIS**	Set all MPIAxis events within the event mask.
mpiEventMask**BIT**	Set mask to only handle events of type *type*.
mpiEventMask**BitGET**	Reports if a mask is set to handle events of type type.
mpiEventMask**BitSET**	Sets mask to handle events of type *type*.
mpiEventMask**BIT_POSITION**	Returns the bit number that is associated with MPI/MEI event type.
mpiEventMask**BIT_POSITION_MASK**	Returns an element's bit-mask for the specified event type.
mpiEventMask**CLEAR**	Set mask to handle no events.
mpiEventMask**COMPLEMENT**	Change the value of every bit within the event mask.
mpiEventMask**EXTERNAL**	Set external events within the event mask.
mpiEventMask**GET**	Reports if a mask is set to handle events of type *type*.
mpiEventMask**IS_CLEAR**	
mpiEventMask**IS_EQUAL**	Tests the equality of two event masks.
mpiEventMask**MOTION** / meiEventMask**MOTION**	Set all MPIMotion events within the event mask.
mpiEventMask**MOTOR** / meiEventMask**MOTOR**	Set all MPIMotor events within the event mask.
mpiEventMask**OR_ASSIGN**	Add all events masked by src to the event mask dst.

mpiEventMask**RECORDER**	Sets all MPIRecorder events within the event mask.
mpiEventMask**SET**	Set mask to handle events of type *type*.
mpiEventMask**SET_ALL**	Set mask to handle all events whose type is enumerated less than type.
meiEventMask**SYNQNET**	Set all MEISynqNet events within the event mask.
meiEventMask**SQNODE**	Set all MEISqNode events within the event mask.
mpiEventMask**WORD**	

Constants

MEIEventMask**BITS_IN_ELEMENT**	Define the number of bits in each data element of MPIEventMask.
MPIEventMask**ELEMENTS**	Define the number of data elements in a MPIEventMask.
MPIEventMask**ELEMENT_TYPE**	Define what the data type MPIEventMask is comprised of.

MPIEventMask

MPIEventMask

```
#define MPIEventMaskELEMENTS    (2)
typedef unsigned long MPIEventMaskELEMENT_TYPE;
typedef MPIEventMaskELEMENT_TYPE MPIEventMask[MPIEventMaskELEMENTS];
```

Description

EventMask is an array of unsigned longs, with a length defined by MPIEventMaskELEMENTS. Each bit in the array represents a mask for a particular event. Be sure to always use the mpiEventMask…(…) macros to set or clear the event masks.

See Also

MPIEventType | MPI EventMask Objects

mpiEventMaskALL / meiEventMaskALL

mpiEventMaskALL

Declaration

```
mpiEventMaskALL(mask)
```

Required Header

stdmpi.h

Description

EventMaskALL is a macro that sets all the bits associated with MPI events in the event mask. The MPI event types are defined in the MPIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiXxxxxEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

meiEventMaskALL

Declaration

```
meiEventMaskALL(mask)
```

Required Header

stdmei.h

Description

EventMaskALL is a macro that sets all the bits associated with MEI events in the event mask. The MEI event types are defined in the MEIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventType | MPIEventMask

mpiEventMaskAND_ASSIGN

Declaration

```
mpiEventMaskAND_ASSIGN(dst,src)
```

Required Header

stdmpi.h

Description

EventMaskOR_ASSIGN is a macro that bitwise ANDs all the bits associated with MPI/MEI events in the event mask ***src*** with ***dst*** and assigns the result to ***dst***. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

dst	An array of unsigned longs. Use MPIEventMask to declare the dst. Each bit in the array represents a mask for a particular event.
src	An array of unsigned longs. Use MPIEventMask to declare the src. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MPIEventMaskASSIGN | MPIEventMaskOR_ASSIGN |

mpiEventMaskASSIGN

Declaration

```
mpiEventMaskASSIGN(dst,src)
```

Required Header

stdmpi.h

Description

EventMaskASSIGN is a macro that assigns all the bits associated with MPI/MEI events in the event mask ***src*** to the event mask ***dst***. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

dst	An array of unsigned longs. Use MPIEventMask to declare the dst. Each bit in the array represents a mask for a particular event.
src	An array of unsigned longs. Use MPIEventMask to declare the src. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | mpiEventMaskOR_ASSIGN | mpiEventMaskAND_ASSIGN

mpiEventMaskAXIS / meiEventMaskAXIS

mpiEventMaskAXIS

Declaration

```
mpiEventMaskAXIS(mask)
```

Required Header

stdmpi.h

Description

EventMaskAXIS is a macro that assigns all the bits associated with MPI Axis object events to the event ***mask***. The MPI event types are defined in the MPIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

meiEventMaskAXIS

Declaration

```
meiEventMaskAXIS(mask)
```

Required Header

stdmei.h

Description

EventMaskAXIS is a macro that assigns all the bits associated with MEI Axis object events to the event ***mask***. The MEI event types are defined in the MEIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MEIEventType

mpiEventMaskBIT

Declaration

```
mpiEventMaskBIT(mask,type)
```

Required Header

stdmpi.h

Description

EventMaskBIT is a macro that assigns a bit associated with MPI/MEI event type in the event mask. The event types are defined in the MPIEventType and MEIEventType enumerations. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.
type	An enumerated event type. Use an enumerated value from MPIEventType or MEIEventType.

See Also

MPIEventMask | MPIEventType | MEIEventType | mpiEventMaskBitGET | mpiEventMaskBitSET |

mpiEventMaskBitGET

Declaration

```
mpiEventMaskBitGET(mask,bit)
```

Required Header

stdmpi.h

Description

EventMaskBitGet is a macro that returns TRUE if the ***bit*** associated with a MPI/MEI event in the event ***mask*** is TRUE. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.
bit	A bit number associated with an event type in an event mask.

Returns

TRUE if the specified bit is TRUE in the event mask.
FALSE if the specified bit is FALSE in the event mask.

See Also

MPIEventMask | mpiEventMaskBIT | mpiEventMaskBitSET |

mpiEventMaskBitSET

Declaration

```
mpiEventMaskBitSET(mask,bit,value)
```

Required Header

stdmpi.h

Description

EventMaskBitSET is a macro that sets (value = TRUE) or clears (value = FALSE) the ***bit*** associated with a MPI/MEI event in the event ***mask***. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.
bit	A bit number associated with an event type in an event mask.
value	TRUE to set a bit, FALSE to clear a bit.

See Also

MPIEventMask | mpiEventMaskBitGET | mpiEventMaskBIT |

mpiEventMaskBIT_POSITION

Declaration

```
mpiEventMaskBIT_POSITION (type)
```

Required Header

stdmpi.h

Description

EventMaskBIT_POSITION is a macro that returns the bit number that is associated with MPI/MEI event ***type***. The event types are defined in the MPIEventType and MEIEventType enumerations.

type	An enumerated event type. Use an enumerated value from MPIEventType or MEIEventType.

Returns

A bit number associated with the event type.

See Also

MPIEventMask | MPIEventType | MEIEventType | mpiEventMaskBIT | mpiEventMaskBitGET | mpiEventMaskBitSET |

mpiEventMaskBIT_POSITION_MASK

Declaration

```
mpiEventMaskBIT_POSITION_MASK (type)
```

Required Header

stdmpi.h

Description

EventMaskBIT_POSITION_MASK is a macro that returns an event mask with a bit set that is associated with MPI/MEI event ***type***. The event types are defined in the MPIEventType and MEIEventType enumerations. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

type	An enumerated event type. Use an enumerated value from MPIEventType or MEIEventType.

Returns

An event mask. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MPIEventType | MEIEventType | mpiEventMaskBIT | mpiEventMaskBitGET | mpiEventMaskBitSET |

mpiEventMaskCLEAR

Declaration

```
mpiEventMaskCLEAR(mask)
```

Required Header

stdmpi.h

Description

EventMaskCLEAR is a macro that clears all the bits in an event ***mask***.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | mpiEventMaskIS_CLEAR | mpiEventMaskSET_ALL

mpiEventMaskCOMPLEMENT

Declaration

```
mpiEventMaskCOMPLEMENT(mask)
```

Required Header

stdmpi.h

Description

EventMaskCOMPLEMENT is a macro that inverts all the bits in an event ***mask***. If a bit associated with an event type is TRUE, EventMaskCOMPLEMENT will set the bit to FALSE. And likewise, if a bit associated with an event type is FALSE, EventMaskCOMPLEMENT will set the bit to TRUE.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | mpiEventMaskCLEAR | mpiEventMaskSET_ALL

mpiEventMaskEXTERNAL

Declaration

`mpiEventMaskEXTERNAL(mask)`

Required Header

stdmpi.h

Description

EventMaskEXTERNAL is a macro that assigns all the bits associated with MPI External events to the event ***mask***. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask

mpiEventMaskGET

Declaration

```
mpiEventMaskGET(mask,type)
```

Required Header

stdmpi.h

Description

EventMaskGET is a macro that returns TRUE if the bit associated with the event ***type*** in the event ***mask*** is TRUE. The event types are defined in the MPIEventType and MEIEventType enumerations. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.
type	An enumerated event type. Use an enumerated value from MPIEventType or MEIEventType.

Returns

TRUE if the bit associated with the event type is TRUE in the event mask.
FALSE if the bit associated with the event type is FALSE in the event mask.

See Also

MPIEventMask | MPIEventType | MEIEventType | mpiEventMaskSET | mpiEventMaskSET_ALL |

mpiEventMaskIS_CLEAR

Declaration

```
mpiEventMaskIS_CLEAR(mask)
```

Required Header

stdmpi.h

Description

EventMaskIS_CLEAR is a macro that returns TRUE if all the bits in an event ***mask*** are FALSE.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

Returns

TRUE is all the bits in an event mask are FALSE.
FALSE if any bit in an event mask is TRUE.

See Also

MPIEventMask | mpiEventMaskCLEAR | mpiEventMaskSET_ALL

mpiEventMaskIS_EQUAL

Declaration

```
mpiEventMaskIS_EQUAL(mask1,mask2)
```

Required Header

stdmpi.h

Description

EventMaskIS_EQUAL is a macro that returns TRUE if event ***mask1*** is the same as event ***mask2***. The MPI event types are defined in the MPIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask1	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.
mask2	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

Returns

TRUE if event mask1 is the same as event mask2.
FALSE if event mask1 is different from event mask2.

See Also

MPIEventMask | mpiEventMaskGET | mpiEventMaskSET

mpiEventMaskMOTION / meiEventMaskMOTION

mpiEventMaskMOTION

Declaration

```
mpiEventMaskMOTION(mask)
```

Required Header

stdmpi.h

Description

EventMaskMOTION is a macro that assigns all the bits associated with MPI Motion object events to the event ***mask***. The MPI event types are defined in the MPIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

meiEventMaskMOTION

Declaration

```
meiEventMaskMOTION(mask)
```

Required Header

stdmei.h

Description

EventMaskMOTION is a macro that assigns all the bits associated with MPI Motion object events to the event ***mask***. The MPI event types are defined in the MPIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MPIEventType

mpiEventMaskMOTOR / meiEventMaskMOTOR

mpiEventMaskMOTOR

Declaration

```
mpiEventMaskMOTOR(mask) mpiEventMaskBitSET((mask), MPIEventTypeAMP_FAULT, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeHOME, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeLIMIT_ERROR, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeLIMIT_HW_NEG, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeLIMIT_HW_POS, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeLIMIT_SW_NEG, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeLIMIT_SW_POS, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeENCODER_FAULT, 1), \
                        mpiEventMaskBitSET((mask), MPIEventTypeAMP_WARNING, 1)
```

Required Header

stdmpi.h

Description

EventMaskMOTOR is a macro that assigns all the bits associated with MPI Motor object events to the event ***mask***. The MPI event types are defined in the MPIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

meiEventMaskMOTOR

Declaration

```
meiEventMaskMOTOR(mask)
```

Required Header

stdmei.h

Description

EventMaskMOTOR is a macro that assigns all the bits associated with MEI Motor object events to the event ***mask***. The MEI event types are defined in the MEIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MEIEventType

mpiEventMaskOR_ASSIGN

Declaration

```
mpiEventMaskOR_ASSIGN(dst,src)
```

Required Header

stdmpi.h

Description

EventMaskIS_EQUAL is a macro that bitwise ORs all the bits associated with MPI/MEI events in the event mask ***src*** with ***dst*** and assigns the result to ***dst***. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

dst	An array of unsigned longs. Use MPIEventMask to declare the dst. Each bit in the array represents a mask for a particular event.
src	An array of unsigned longs. Use MPIEventMask to declare the src. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MPIEventMaskASSIGN | MPIEventMaskAND_ASSIGN

mpiEventMaskRECORDER

Declaration

```
mpiEventMaskRECORDER(mask)  mpiEventMaskBitSET((mask), MPIEventTypeRECORDER_HIGH, 1),
\
                            mpiEventMaskBitSET((mask), MPIEventTypeRECORDER_FULL, 1),
\
                            mpiEventMaskBitSET((mask), MPIEventTypeRECORDER_DONE, 1)
```

Required Header

stdmpi.h

Description

EventMaskRECORDER is a macro that assigns all the bits associated with MPI Recorder object events to the event ***mask***. The MPI event types are defined in the MPIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MPIEventType

mpiEventMaskSET

Declaration

mpiEventMaskSET(mask,type)

Required Header

stdmpi.h

Description

EventMaskSET is a macro that sets the ***bit*** associated with MPI event type in the event ***mask***. The event types are defined in the MPIEventType and MEIEventType enumerations. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.
type	An enumerated event type. Use an enumerated value from MPIEventType or MEIEventType.

See Also

MPIEventMask | mpiEventMaskGET | MPIEventType | MEIEventType | mpiEventMaskSET_ALL

mpiEventMaskSET_ALL

Declaration

```
mpiEventMaskSET_ALL(mask,type)
```

Required Header

stdmpi.h

Description

EventMaskSET_ALL is a macro that sets all the ***bits*** associated with MPI/MEI event types that are less than the specified event ***type***. The event types are defined in the MPIEventType and MEIEventType enumerations. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.
type	An enumerated event type. Use an enumerated value from MPIEventType or MEIEventType.

See Also

MPIEventMask | MPIEventType | MEIEventType | mpiEventMaskGET | mpiEventMaskSET

meiEventMaskSYNQNET

Declaration

```
meiEventMaskSYNQNET(mask)
```

Required Header

stdmei.h

Description

EventMaskSYNQNET is a macro that assigns all the bits associated with MEI SynqNet object events to the event mask. The MEI event types are defined in the MEIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MEIEventType

meiEventMaskSQNODE

Declaration

```
meiEventMaskSQNODE(mask)
```

Required Header

stdmei.h

Description

EventMaskSQNODE is a macro that assigns all the bits associated with MEI SqNode object events to the event ***mask***. The MEI event types are defined in the MEIEventType enumeration. After the event mask bits are initialized, the mask can be passed to a mpiObjectEventNotifySet(...) method, which configures the controller to generate events.

mask	An array of unsigned longs. Use MPIEventMask to declare the mask. Each bit in the array represents a mask for a particular event.

See Also

MPIEventMask | MEIEventType

mpiEventMaskWORD

Declaration

```
mpiEventMaskWORD(type)
```

Required Header

stdmpi.h

Description

EventMaskWORD is a macro that returns the word number that is associated with MPI/MEI event ***type***. The event types are defined in the MPIEventType and MEIEventType enumerations.

type	An enumerated event type. Use an enumerated value from MPIEventType or MEIEventType.

Returns

The word number associated with the event type.

See Also

MPIEventMask | | MPIEventType | MEIEventType | mpiEventMaskBIT_POSITION

MEIEventMaskBITS_IN_ELEMENT

MEIEventMaskBITS_IN_ELEMENT

```
#define MEIEventMaskBITS_IN_ELEMENT ((unsigned long)
                                     (sizeof(MPIEventMaskELEMENT_TYPE) * 8))
```

Description

EventMaskBITS_IN_ELEMENT defines the number of bits in each data element of MPIEventMask.

NOTE: MEIEventMaskBITS_IN_ELEMENT replaced mpiEventMaskBITS_IN_ELEMENT.

See Also

MPIEventMask | mpiEventMaskBIT | mpiEventMaskBitGET | mpiEventMaskBitSET | mpiEventMaskBIT_POSITION_MASK

MPIEventMaskELEMENTS

MPIEventMaskELEMENTS

```
#define MPIEventMaskELEMENTS    (2)
```

Description

EventMaskELEMENTS defines the number of data elements in a MPIEventMask.

See Also

MPIEventMask

MPIEventMaskELEMENT_TYPE

MPIEventMaskELEMENT_TYPE

```
typedef unsigned long MPIEventMaskELEMENT_TYPE
```

Description

EventMaskELEMENT_TYPE defines what the data type MPIEventMask is comprised of.

See Also

MPIEventMask