TimModule Reference Manual

0.2

Generated by Doxygen 1.2.14

Sun Jun 27 19:57:31 2004

Contents

1 TimModule Documentation

Author:
John Lane <jbl@hep.ucl.ac.uk>

This is an experimental Main Page for Doxygen

NB define eg I_AM_LINUX_HOST for processor.h (eg typedef UINT32)

Here is somewhere to start:

- TimModule.h

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Reference: http://www.hep.ucl.ac.uk/~jbl/SCT/TIM_registers.html

TIM registers reference

2 TimModule Namespace Index

2.1 TimModule Namespace List

Here is a list of all namespaces with brief descriptions:

3 TimModule Hierarchical Index

3.1 TimModule Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

4 TimModule Compound Index

4.1 TimModule Compound List

Here are the classes, structs, unions and interfaces with brief descriptions:

5 TimModule File Index

5.1 TimModule File List

Here is a list of all files with brief descriptions:

6 TimModule Namespace Documentation

6.1 SctPixelRod Namespace Reference

Compounds

- class SctPixelRod::TimException

  TimException: a general exception class for TIM errors.

- class SctPixelRod::TimModule

  TimModule: A derived class for VME TIM modules.

- struct SctPixelRod::TimScanControl

  Structure definition for control of scanning Sequencer buffer.

- struct SctPixelRod::TimScanResults

  Structure definition for results of scanning Sequencer buffer.

- class SctPixelRod::TimSequin

  TimSequin: A class for TIM Sequencer information.

Enumerations

- enum TimTimingSCT { TIM_L1A_DEADTIME = 3, TIM_ECR_DEADTIME = 7, TIM_BCR_DEADTIME = 7, TIM_CAL_DEADTIME = 27, TIM_BCID_OFFSET = 6 }

  Define timing in clock cycles for SCT, Pixel is different.

- enum TimRegister { TIM_REG_ENABLES = 0x00, TIM_REG_COMMAND = 0x02, TIM_REG_BURST_COUNT = 0x04, TIM_REG_FREQUENCY = 0x06, TIM_REG_WINDOW = 0x08, TIM_REG_DELAY = 0x0A, TIM_REG_STATUS = 0x0C, TIM_REG_FIFO_STATUS = 0x0E, TIM_REG_TRIGGER_IDLO = 0x10, TIM_REG_TRIGGER_IDHI = 0x12, TIM_REG_TRIGGER_BCID = 0x14, TIM_REG_TRIGGER_TYPE = 0x16, TIM_REG_RUN_ENABLES = 0x18, TIM_REG_SEQ_CONTROL = 0x1A, TIM_REG_SEQ_END = 0x1C, TIM_REG_ROD_MASK = 0x1E, TIM_REG_ROD_BUSY = 0x20, TIM_REG_ROD_LATCH = 0x22, TIM_REG_ROD_MONITOR = 0x24, TIM_REG_TTC_DATA = 0x26, TIM_REG_TTC_SELECT = 0x28, TIM_REG_TTC_BCID = 0x2A, TIM_REG_TTC_RX = 0x2C, TIM_REG_TTC_STATUS = 0x2E, TIM_REG_OUTPUT = 0x30, TIM_REG_TIM_ID = 0x32 }

  Define register offsets in bytes.

- enum TimBitEnables { TIM_BIT_EN_INT_TRIG = 0x0002, TIM_BIT_EN_INT_ECR = 0x0004, TIM_BIT_EN_INT_BCR = 0x0008, TIM_BIT_EN_RANDOM = 0x0010, TIM_BIT_EN_INT_FER = 0x0020, TIM_BIT_EN_WINDOW = 0x0040, TIM_BIT_EN_INT_BUSY = 0x0080, TIM_BIT_EN_EXT_CLK = 0x0100, TIM_BIT_EN_EXT_TRIG = 0x0200, TIM_BIT_EN_EXT_ECR = 0x0400, TIM_BIT_EN_EXT_BCR = 0x0800, TIM_BIT_EN_EXT_CAL = 0x1000, TIM_BIT_EN_EXT_FER = 0x2000, TIM_BIT_EN_EXT_SEQ = 0x4000, TIM_BIT_EN_EXT_BUSY = 0x8000 }

  Define register bits as masks.

- enum TimMaskFrequency { TIM_MASK_TRIG_100_KHZ = 0x0006, TIM_MASK_TRIG_10_0KHZ = 0x000E, TIM_MASK_TRIG_1_00KHZ = 0x0016, TIM_MASK_TRIG_0_10KHZ = 0x001E, TIM_MASK_FER_10_00HZ = 0x0600, TIM_MASK_FER_1_000HZ = 0x0E00, TIM_MASK_FER_0_100HZ = 0x1600, TIM_MASK_FER_0_010HZ = 0x1E00 }
- enum TimBitBackplane { TIM_L1A = 0x01, TIM_ECR = 0x02, TIM_BCR = 0x04, TIM_CAL = 0x08, TIM_SID = 0x10, TIM_STT = 0x20, TIM_CMD = 0xCF, TIM_RES = 0xC0, TIM_FER = 0x40, TIM_SPA = 0x80, TIM_TRG = 0x31 }

  Applies to Sequencer and Output.

- enum TimBitCommand { TIM_VTRG = 0x02, TIM_VECR = 0x04, TIM_VBCR = 0x08, TIM_VCAL = 0x10, TIM_VFER = 0x20, TIM_VSPA = 0x40, TIM_BIT_VRESET = 0x8000 }

  Applies to Command register.

- enum TimBitRunEnables { TIM_BIT_EN_ID = 0x0200, TIM_BIT_EN_TYPE = 0x0400 }
- enum TimBitSeqControl { TIM_BIT_SEQ_EN_ALL = 0x00FF, TIM_BIT_SEQ_RESET = 0x0200, TIM_BIT_SEQ_GO = 0x0400, TIM_BIT_EN_CYCLIC = 0x0800 }
- enum Name { NONE = -999 }

Variables

- const INT32 TIM_L1ID_FIRST = 0

  triggers.

- const INT32 TIM_SEQ_SIZE = 0x4000

  bytes.

- const INT32 TIM_SEQ_ADDR = 0x8000

  bytes.

- const int s_masks [2] = { TIM_SID, TIM_STT }
- const int s_words [2] = { 2, 1 }
- const int s_bits [2][2] = {{ 24, 12 },{ 10, 0 }}

6.1.1 Enumeration Type Documentation

6.1.1.1 enum SctPixelRod::Name

Enumeration values:

NONE

Definition at line 27 of file TimSequin.cxx.

Referenced by timKeyword.

6.1.1.2 enum SctPixelRod::TimBitBackplane

Applies to Sequencer and Output.

Enumeration values:

TIM_L1A Level-1 Accept trigger.

TIM_ECR Event Counter Reset.

TIM_BCR Bunch Counter Reset.

TIM_CAL Calibrate strobe.

TIM_SID Serial event ID.

TIM_STT Serial Trigger Type.

TIM_CMD Commands available.

TIM_RES Commands reserved.

TIM_FER Front-End Reset - reserved.

TIM_SPA Spare command - reserved.

TIM_TRG Trigger and serial streams.

Definition at line 113 of file TimDefine.h.

Referenced by SctPixelRod::TimSequin::addByBunch, and SctPixelRod::TimSequin::addByIndex.

6.1.1.3 enum SctPixelRod::TimBitCommand

Applies to Command register.

Enumeration values:

TIM_VTRG Single VME Trigger.

TIM_VECR Single VME ECR.

TIM_VBCR Single VME BCR.

TIM_VCAL Single VME CAL.

TIM_VFER Single VME FER.

TIM_VSPA Single VME SPA.

TIM_BIT_VRESET

Definition at line 127 of file TimDefine.h.

Referenced by SctPixelRod::TimModule::issueCommand.

6.1.1.4 enum SctPixelRod::TimBitEnables

Define register bits as masks.

Enumeration values:

TIM_BIT_EN_INT_TRIG Enable internal repetitive Trigger.

TIM_BIT_EN_INT_ECR Enable internal repetitive ECReset.

TIM_BIT_EN_INT_BCR Enable internal repetitive BCReset.

TIM_BIT_EN_RANDOM Enable internal trigger randomizer.

TIM_BIT_EN_INT_FER Enable internal repetitive FEReset.

TIM_BIT_EN_WINDOW Enable trigger window.
TIM_BIT_EN_INT_BUSY Enable internal Busy.
TIM_BIT_EN_EXT_CLK Enable external inputs: clock.
TIM_BIT_EN_EXT_TRIG Enable external inputs: trigger.
TIM_BIT_EN_EXT_ECR Enable external inputs: ECReset.
TIM_BIT_EN_EXT_BCR Enable external inputs: BCReset.
TIM_BIT_EN_EXT_CAL Enable external inputs: Calibrate.
TIM_BIT_EN_EXT_FER Enable external inputs: FEReset.
TIM_BIT_EN_EXT_SEQ Enable external inputs: Sequencer Go.
TIM_BIT_EN_EXT_BUSY Enable external inputs: Busy.

Definition at line 82 of file TimDefine.h.

6.1.1.5 enum SctPixelRod::TimBitRunEnables

Enumeration values:
TIM_BIT_EN_ID
TIM_BIT_EN_TYPE

Definition at line 138 of file TimDefine.h.

6.1.1.6 enum SctPixelRod::TimBitSeqControl

Enumeration values:
TIM_BIT_SEQ_EN_ALL
TIM_BIT_SEQ_RESET
TIM_BIT_SEQ_GO
TIM_BIT_EN_CYCLIC

Definition at line 143 of file TimDefine.h.

6.1.1.7 enum SctPixelRod::TimMaskFrequency

Enumeration values:
TIM_MASK_TRIG_100_KHZ
TIM_MASK_TRIG_10_0KHZ
TIM_MASK_TRIG_1_00KHZ
TIM_MASK_TRIG_0_10KHZ
TIM_MASK_FER_10_00HZ
TIM_MASK_FER_1_000HZ
TIM_MASK_FER_0_100HZ
TIM_MASK_FER_0_010HZ

Definition at line 101 of file TimDefine.h.

Referenced by SctPixelRod::TimModule::intTrigStart.

6.1.1.8 enum SctPixelRod::TimRegister

Define register offsets in bytes.

Enumeration values:

TIM_REG_ENABLES
TIM_REG_COMMAND
TIM_REG_BURST_COUNT
TIM_REG_FREQUENCY
TIM_REG_WINDOW
TIM_REG_DELAY
TIM_REG_STATUS
TIM_REG_FIFO_STATUS
TIM_REG_TRIGGER_IDLO
TIM_REG_TRIGGER_IDHI
TIM_REG_TRIGGER_BCID
TIM_REG_TRIGGER_TYPE
TIM_REG_RUN_ENABLES
TIM_REG_SEQ_CONTROL
TIM_REG_SEQ_END
TIM_REG_ROD_MASK
TIM_REG_ROD_BUSY
TIM_REG_ROD_LATCH
TIM_REG_ROD_MONITOR
TIM_REG_TTC_DATA
TIM_REG_TTC_SELECT
TIM_REG_TTC_BCID
TIM_REG_TTC_RX
TIM_REG_TTC_STATUS
TIM_REG_OUTPUT
TIM_REG_TIM_ID

Definition at line 51 of file TimDefine.h.

Referenced by SctPixelRod::TimModule::loadBitClear, SctPixelRod::TimModule::loadBitSet, SctPixelRod::TimModule::loadByteHi, SctPixelRod::TimModule::loadByteLo, SctPixelRod::TimModule::regFetch, and SctPixelRod::TimModule::regLoad.

6.1.1.9 enum SctPixelRod::TimTimingSCT

Define timing in clock cycles for SCT, Pixel is different.

Enumeration values:

TIM_L1A_DEADTIME
TIM_ECR_DEADTIME
TIM_BCR_DEADTIME
TIM_CAL_DEADTIME
TIM_BCID_OFFSET

Definition at line 33 of file TimDefine.h.

6.1.2 Variable Documentation

6.1.2.1 const int SctPixelRod::s_bits[2][2] = {{ 24, 12 },{ 10, 0 }} [static]

Definition at line 31 of file TimSequin.cxx.

Referenced by SctPixelRod::TimSequin::addTrigger, and SctPixelRod::TimSequin::scan.

6.1.2.2 const int SctPixelRod::s_masks[2] = { TIM_SID, TIM_STT } [static]

Definition at line 29 of file TimSequin.cxx.

Referenced by SctPixelRod::TimSequin::addTrigger, and SctPixelRod::TimSequin::scan.

6.1.2.3 const int SctPixelRod::s_words[2] = { 2, 1 } [static]

Definition at line 30 of file TimSequin.cxx.

Referenced by SctPixelRod::TimSequin::addTrigger, and SctPixelRod::TimSequin::scan.

6.1.2.4 const INT32 SctPixelRod::TIM_L1ID_FIRST = 0

triggers.

Definition at line 41 of file TimDefine.h.

Referenced by SctPixelRod::TimSequin::addByIndex, SctPixelRod::TimSequin::reset, and SctPixelRod::TimSequin::scan.

6.1.2.5 const INT32 SctPixelRod::TIM_SEQ_ADDR = 0x8000

bytes.

Definition at line 47 of file TimDefine.h.

Referenced by SctPixelRod::TimModule::seqFetch, and SctPixelRod::TimModule::seqLoad.

6.1.2.6 const INT32 SctPixelRod::TIM_SEQ_SIZE = 0x4000

bytes.

Sequencer RAM is 16K bytes for both source and sink memory. They are accessed together as 16K 16-bit words.

Definition at line 46 of file TimDefine.h.

Referenced by SctPixelRod::TimSequin::addTrigger, damon, SctPixelRod::TimSequin::fill, SctPixelRod::TimSequin::reset, SctPixelRod::TimSequin::scanDefaults, SctPixelRod::TimSequin::setBuffer, test, and timKeyword.

6.2 std Namespace Reference

Overloaded operator to print TIM status.

Functions

- ostream & operator<< (ostream &os, TimModule &tim)

6.2.1 Detailed Description

Overloaded operator to print TIM status.

6.2.2 Function Documentation

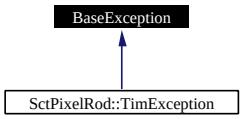
6.2.2.1 ostream & std::operator<< (ostream & *os*, SctPixelRod::TimModule & *tim*)

Definition at line 382 of file TimModule.cxx.

7 TimModule Class Documentation

7.1 BaseException Class Reference

Inheritance diagram for BaseException:



The documentation for this class was generated from the following file:

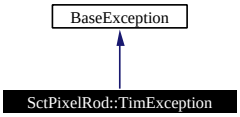
- TimModule.h

7.2 SctPixelRod::TimException Class Reference

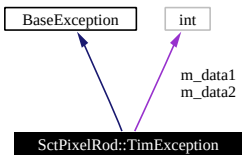
TimException: a general exception class for TIM errors.

`#include <TimModule.h>`

Inheritance diagram for SctPixelRod::TimException:



Collaboration diagram for SctPixelRod::TimException:



Public Methods

- TimException (std::string descriptor, int data1, int data2)
  Constructors. Use defaults for destructor, copy, and assignment.
- int getData1 ()
- int getData2 ()
- virtual void what (std::ostream &)

Private Attributes

- int m_data1
  First data value returned.
- int m_data2
  Second data value returned.

7.2.1 Detailed Description

TimException: a general exception class for TIM errors.

This class is thrown if an error in a TIM operation is detected.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Definition at line 51 of file TimModule.h.

7.2.2 Constructor & Destructor Documentation

7.2.2.1 SctPixelRod::TimException::TimException (std::string *descriptor*, int *data1*, int *data2*)

Constructors. Use defaults for destructor, copy, and assignment.

Definition at line 43 of file TimModule.cxx.

References m_data1, and m_data2.

7.2.3 Member Function Documentation

7.2.3.1 int SctPixelRod::TimException::getData1 () [inline]

Definition at line 56 of file TimModule.h.

References m_data1.

Referenced by what.

7.2.3.2 int SctPixelRod::TimException::getData2 () [inline]

Definition at line 57 of file TimModule.h.

References m_data2.

Referenced by what.

7.2.3.3 void SctPixelRod::TimException::what (std::ostream &) [virtual]

Definition at line 51 of file TimModule.cxx.

References getData1, and getData2.

7.2.4 Member Data Documentation

7.2.4.1 int SctPixelRod::TimException::m_data1 [private]

First data value returned.

Definition at line 62 of file TimModule.h.

Referenced by getData1, and TimException.

7.2.4.2 int SctPixelRod::TimException::m_data2 [private]

Second data value returned.

Definition at line 63 of file TimModule.h.

Referenced by getData2, and TimException.

The documentation for this class was generated from the following files:

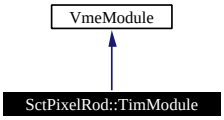
- TimModule.h
- TimModule.cxx

7.3 SctPixelRod::TimModule Class Reference

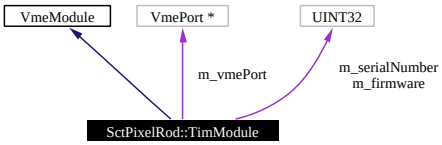
TimModule: A derived class for VME TIM modules.

`#include <TimModule.h>`

Inheritance diagram for SctPixelRod::TimModule:



Collaboration diagram for SctPixelRod::TimModule:



Public Methods

- TimModule (UINT32 baseAddr, UINT32 mapSize, VmeInterface &ourInterface)

This is the only constructor to use.

- virtual ~TimModule ()
- UINT32 getFirmware ()
- UINT32 getSerialNumber ()
- VmePort * getVmePort ()
- void initialize ()
- void reset ()
- void status ()
- UINT32 fetchL1ID ()

  Read trigger number.

- UINT16 fetchTimID ()

  Read TIM ID register.

- void intTrigStart (const TimMaskFrequency frequency)

  Internal Trigger.

- void intTrigStop ()

  Stop Internal Trigger.

- void issueCommand (const TimBitCommand mask)

  Issue TIM command.

- void issueVCAL (const UINT8 pipelineDelay)

  Issue VCAL + L1A command.

- void loadBitClear (const TimRegister addr, const UINT16 mask)

  Clear bit.

- void loadBitSet (const TimRegister addr, const UINT16 mask)

  Set bit.

- void loadByteHi (const TimRegister addr, const UINT8 byte)

  Upper byte.

- void loadByteLo (const TimRegister addr, const UINT8 byte)

  Lower byte.

- UINT16 regFetch (const TimRegister addr)

  Read from a 16-bit VME register.

- void regLoad (const TimRegister addr, const UINT16 data)

  Write to a 16-bit VME register.

- void seqFetch (const UINT16 size, UINT16 buffer[ ])

  Read sequencer.

- void seqLoad (const UINT16 size, const UINT16 buffer[ ])

  Load sequencer.

- void seqRun (const UINT16 size)
  Run sequencer.
- UINT16 vmeFetch (const UINT32 addr) throw (VmeException &)
  Read from a 16-bit VME register.
- void vmeLoad (const UINT32 addr, const UINT16 data) throw (VmeException &)
  Write to a 16-bit VME register.

Private Attributes

- UINT32 m_firmware
  Firmware version number.
- UINT32 m_serialNumber
  Board serial number.
- VmePort * m_vmePort
  VME Port handle.

7.3.1 Detailed Description

TimModule: A derived class for VME TIM modules.

This is the implementation of a TIM class derived from the VmeModule base class. It should be the sole interface for VME communication with TIM.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Definition at line 76 of file TimModule.h.

7.3.2 Constructor & Destructor Documentation

7.3.2.1 SctPixelRod::TimModule::TimModule (UINT32 *baseAddr*, UINT32 *mapSize*, VmeInterface & *vme*)

This is the only constructor to use.

Objects of this class cannot be copied or assigned. This could lead to VME conflicts.

Definition at line 69 of file TimModule.cxx.

References baseAddr, m_serialNumber, m_vmePort, mapSize, and vme.

7.3.2.2 SctPixelRod::TimModule::~TimModule () [virtual]

This deletes all its VME Ports.

Definition at line 95 of file TimModule.cxx.

References m_vmePort.

7.3.3 Member Function Documentation

7.3.3.1 UINT32 SctPixelRod::TimModule::fetchL1ID ()

Read trigger number.

This method reads the last TIM L1ID value.

Definition at line 154 of file TimModule.cxx.

References m_firmware, regFetch, SctPixelRod::TIM_REG_TRIGGER_IDHI, and SctPixelRod::TIM_REG_TRIGGER_IDLO.

Referenced by status.

7.3.3.2 UINT16 SctPixelRod::TimModule::fetchTimID ()

Read TIM ID register.

This method reads the TIM ID register.

Definition at line 170 of file TimModule.cxx.

References m_firmware, m_serialNumber, regFetch, and SctPixelRod::TIM_REG_TIM_ID.

Referenced by initialize.

7.3.3.3 UINT32 SctPixelRod::TimModule::getFirmware () [inline]

Definition at line 89 of file TimModule.h.

References m_firmware.

7.3.3.4 UINT32 SctPixelRod::TimModule::getSerialNumber () [inline]

Definition at line 90 of file TimModule.h.

References m_serialNumber.

7.3.3.5 VmePort* SctPixelRod::TimModule::getVmePort () [inline]

Definition at line 91 of file TimModule.h.

References m_vmePort.

7.3.3.6 void SctPixelRod::TimModule::initialize ()

This method configures the TIM into its initialized state.

Definition at line 110 of file TimModule.cxx.

References fetchTimID, regLoad, SctPixelRod::TIM_BCID_OFFSET, SctPixelRod::TIM_BIT_EN_ID, SctPixelRod::TIM_BIT_EN_TYPE, SctPixelRod::TIM_REG_COMMAND, SctPixelRod::TIM_REG_RUN_ENABLES, and SctPixelRod::TIM_REG_TRIGGER_BCID.

7.3.3.7 void SctPixelRod::TimModule::intTrigStart (const TimMaskFrequency *frequency*)

Internal Trigger.

This method Enables Internal Triggers with the given repeat frequency.

Definition at line 184 of file TimModule.cxx.

References loadBitSet, loadByteLo, SctPixelRod::TIM_BIT_EN_INT_TRIG, SctPixelRod::TIM_REG_ENABLES, SctPixelRod::TIM_REG_FREQUENCY, and SctPixelRod::TimMaskFrequency.

7.3.3.8 void SctPixelRod::TimModule::intTrigStop (void)

Stop Internal Trigger.

This method stops Internal Triggers.

Definition at line 195 of file TimModule.cxx.

References loadBitClear, SctPixelRod::TIM_BIT_EN_INT_TRIG, and SctPixelRod::TIM_REG_ENABLES.

7.3.3.9 void SctPixelRod::TimModule::issueCommand (const TimBitCommand *mask*)

Issue TIM command.

This method issues a TIM command edge-mode bit - experimental!

Definition at line 205 of file TimModule.cxx.

References loadBitClear, loadBitSet, SctPixelRod::TIM_REG_COMMAND, SctPixelRod::TIM_VSPA, and SctPixelRod::TimBitCommand.

Referenced by issueVCAL.

7.3.3.10 void SctPixelRod::TimModule::issueVCAL (const UINT8 *pipelineDelay*)

Issue VCAL + L1A command.

This method issues a VME CAL command followed by an L1A, after the given pipeline delay.

Definition at line 218 of file TimModule.cxx.

References issueCommand, loadByteLo, SctPixelRod::TIM_REG_DELAY, and SctPixelRod::TIM_VCAL.

7.3.3.11 void SctPixelRod::TimModule::loadBitClear (const TimRegister *addr*, const UINT16 *mask*)

Clear bit.

This method clears the bits set in a 16-bit bit mask into a VME register, leaving the other bits unchanged.

Definition at line 242 of file TimModule.cxx.

References SctPixelRod::TimRegister, vmeFetch, and vmeLoad.

Referenced by intTrigStop, and issueCommand.

7.3.3.12 void SctPixelRod::TimModule::loadBitSet (const TimRegister *addr*, const UINT16 *mask*)

Set bit.

This method writes the bits set in a 16-bit bit mask into a VME register, leaving the other bits unchanged.

Definition at line 230 of file TimModule.cxx.

References SctPixelRod::TimRegister, vmeFetch, and vmeLoad.

Referenced by intTrigStart, and issueCommand.

7.3.3.13 void SctPixelRod::TimModule::loadByteHi (const TimRegister *addr*, const UINT8 *byte*)

Upper byte.

This method writes a byte into the upper half of a 16-bit VME register, leaving the other byte unchanged.

Definition at line 254 of file TimModule.cxx.

References SctPixelRod::TimRegister, vmeFetch, and vmeLoad.

7.3.3.14 void SctPixelRod::TimModule::loadByteLo (const TimRegister *addr*, const UINT8 *byte*)

Lower byte.

This method writes a byte into the lower half of a 16-bit VME register, leaving the other byte unchanged.

Definition at line 266 of file TimModule.cxx.

References SctPixelRod::TimRegister, vmeFetch, and vmeLoad.

Referenced by intTrigStart, and issueVCAL.

7.3.3.15 UINT16 SctPixelRod::TimModule::regFetch (const TimRegister *addr*)

Read from a 16-bit VME register.

This method reads a 16-bit value from a VME register.

Definition at line 277 of file TimModule.cxx.

References SctPixelRod::TimRegister, and vmeFetch.

Referenced by fetchL1ID, fetchTimID, and status.

7.3.3.16 void SctPixelRod::TimModule::regLoad (const TimRegister *addr*, const UINT16 *data*)

Write to a 16-bit VME register.

This method writes a 16-bit value into a VME register.

Definition at line 288 of file TimModule.cxx.

References SctPixelRod::TimRegister, and vmeLoad.

Referenced by initialize, reset, and seqRun.

7.3.3.17 void SctPixelRod::TimModule::reset ()

This method issues a reset to the TIM.

Definition at line 124 of file TimModule.cxx.

References regLoad, SctPixelRod::TIM_BIT_VRESET, and SctPixelRod::TIM_REG_COMMAND.

7.3.3.18 void SctPixelRod::TimModule::seqFetch (const UINT16 *size*, UINT16 *buffer*[ ])

Read sequencer.

This method reads the Sequencer memory into a buffer.

Definition at line 312 of file TimModule.cxx.

References SctPixelRod::TIM_SEQ_ADDR, and vmeFetch.

7.3.3.19 void SctPixelRod::TimModule::seqLoad (const UINT16 *size*, const UINT16 *buffer*[ ])

Load sequencer.

This method writes a buffer into the Sequencer memory.

Definition at line 324 of file TimModule.cxx.

References SctPixelRod::TIM_SEQ_ADDR, and vmeLoad.

7.3.3.20 void SctPixelRod::TimModule::seqRun (const UINT16 *size*)

Run sequencer.

This method starts the Sequencer executing.

Definition at line 298 of file TimModule.cxx.

References regLoad, SctPixelRod::TIM_BIT_SEQ_EN_ALL, SctPixelRod::TIM_BIT_SEQ_GO, SctPixelRod::TIM_BIT_SEQ_RESET, SctPixelRod::TIM_REG_SEQ_CONTROL, and SctPixelRod::TIM_REG_SEQ_END.

7.3.3.21 void SctPixelRod::TimModule::status ()

This method reports the status of the TIM. For now, it simply prints to standard output.

Definition at line 135 of file TimModule.cxx.

References fetchL1ID, m_firmware, m_serialNumber, regFetch, SctPixelRod::TIM_REG_STATUS, and SctPixelRod::TIM_REG_TRIGGER_BCID.

7.3.3.22 UINT16 SctPixelRod::TimModule::vmeFetch (const UINT32 *addr*) throw (VmeException &)

Read from a 16-bit VME register.

This method reads a 16-bit value from a VME register.

Definition at line 338 of file TimModule.cxx.

Referenced by loadBitClear, loadBitSet, loadByteHi, loadByteLo, regFetch, and seqFetch.

7.3.3.23 void SctPixelRod::TimModule::vmeLoad (const UINT32 *addr*, const UINT16 *data*) throw (VmeException &)

Write to a 16-bit VME register.

This method writes a 16-bit value into a VME register.

Definition at line 357 of file TimModule.cxx.

Referenced by loadBitClear, loadBitSet, loadByteHi, loadByteLo, regLoad, and seqLoad.

7.3.4 Member Data Documentation

7.3.4.1 UINT32 SctPixelRod::TimModule::m_firmware [private]

Firmware version number.

Definition at line 131 of file TimModule.h.

Referenced by fetchL1ID, fetchTimID, getFirmware, and status.

7.3.4.2 UINT32 SctPixelRod::TimModule::m_serialNumber [private]

Board serial number.

Definition at line 132 of file TimModule.h.

Referenced by fetchTimID, getSerialNumber, status, and TimModule.

7.3.4.3 VmePort* SctPixelRod::TimModule::m_vmePort [private]

VME Port handle.

Definition at line 133 of file TimModule.h.

Referenced by getVmePort, TimModule, and ~TimModule.

The documentation for this class was generated from the following files:

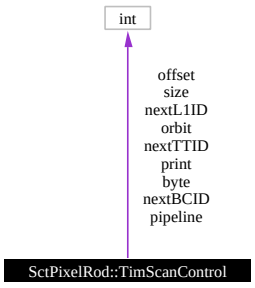
- TimModule.h
- TimModule.cxx

7.4 SctPixelRod::TimScanControl Struct Reference

Structure definition for control of scanning Sequencer buffer.

`#include <TimSequin.h>`

Collaboration diagram for SctPixelRod::TimScanControl:



Public Attributes

- int size

  Size to scan.

- int byte

- int print
- int offset
- int orbit
- int pipeline
- int nextL1ID
- int nextBCID
- int nextTTID

7.4.1 Detailed Description

Structure definition for control of scanning Sequencer buffer.

Definition at line 29 of file TimSequin.h.

7.4.2 Member Data Documentation

7.4.2.1 int SctPixelRod::TimScanControl::byte

Definition at line 31 of file TimSequin.h.

7.4.2.2 int SctPixelRod::TimScanControl::nextBCID

Definition at line 37 of file TimSequin.h.

7.4.2.3 int SctPixelRod::TimScanControl::nextL1ID

Definition at line 36 of file TimSequin.h.

7.4.2.4 int SctPixelRod::TimScanControl::nextTTID

Definition at line 38 of file TimSequin.h.

7.4.2.5 int SctPixelRod::TimScanControl::offset

Definition at line 33 of file TimSequin.h.

7.4.2.6 int SctPixelRod::TimScanControl::orbit

Definition at line 34 of file TimSequin.h.

7.4.2.7 int SctPixelRod::TimScanControl::pipeline

Definition at line 35 of file TimSequin.h.

7.4.2.8 int SctPixelRod::TimScanControl::print

Definition at line 32 of file TimSequin.h.

7.4.2.9 int SctPixelRod::TimScanControl::size

Size to scan.

Definition at line 30 of file TimSequin.h.

The documentation for this struct was generated from the following file:

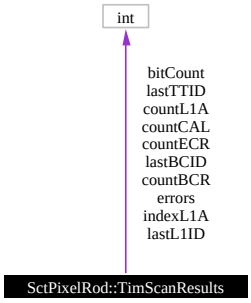
- TimSequin.h

7.5 SctPixelRod::TimScanResults Struct Reference

Structure definition for results of scanning Sequencer buffer.

`#include <TimSequin.h>`

Collaboration diagram for SctPixelRod::TimScanResults:



Public Attributes

- int lastL1ID
- int lastBCID
- int lastTTID
- int indexL1A
- int countL1A
- int countECR
- int countBCR
- int countCAL
- int bitCount
- int errors

7.5.1 Detailed Description

Structure definition for results of scanning Sequencer buffer.

Definition at line 45 of file TimSequin.h.

7.5.2 Member Data Documentation

7.5.2.1 int SctPixelRod::TimScanResults::bitCount

Definition at line 54 of file TimSequin.h.

7.5.2.2 int SctPixelRod::TimScanResults::countBCR

Definition at line 52 of file TimSequin.h.

7.5.2.3 int SctPixelRod::TimScanResults::countCAL

Definition at line 53 of file TimSequin.h.

7.5.2.4 int SctPixelRod::TimScanResults::countECR

Definition at line 51 of file TimSequin.h.

7.5.2.5 int SctPixelRod::TimScanResults::countL1A

Definition at line 50 of file TimSequin.h.

7.5.2.6 int SctPixelRod::TimScanResults::errors

Definition at line 55 of file TimSequin.h.

7.5.2.7 int SctPixelRod::TimScanResults::indexL1A

Definition at line 49 of file TimSequin.h.

7.5.2.8 int SctPixelRod::TimScanResults::lastBCID

Definition at line 47 of file TimSequin.h.

7.5.2.9 int SctPixelRod::TimScanResults::lastL1ID

Definition at line 46 of file TimSequin.h.

7.5.2.10 int SctPixelRod::TimScanResults::lastTTID

Definition at line 48 of file TimSequin.h.

The documentation for this struct was generated from the following file:

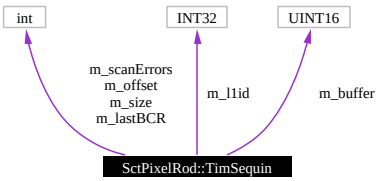
- TimSequin.h

7.6 SctPixelRod::TimSequin Class Reference

TimSequin: A class for TIM Sequencer information.

`#include <TimSequin.h>`

Collaboration diagram for SctPixelRod::TimSequin:



Public Methods

- TimSequin ()

  TimSequin: A class for TIM Sequencer.

- ~TimSequin ()
- int getBuffer (const UINT16 size, UINT16 buffer[ ])
- int setBuffer (const UINT16 size, const UINT16 buffer[ ])
- int getLength ()
- void setOffset (const int offset)
- void addByBunch (const TimBitBackplane mask, const int bcid)
- void addByIndex (const TimBitBackplane mask, const int index)
- void addTrigger (const int index, const UINT32 ids[3], const int delays[2])
- void fill (const UINT16 size, const UINT16 value)
- int getBCID (const int iL1A, const int iBCR, const int offset)
- int getIndex (const int bcid, const int iBCR, const int offset)
- void reset ()
- int scanRun ()
- TimScanResults scan (const TimScanControl control)
- TimScanControl scanDefaults ()

Private Methods

- void m_scanError (const char *s, const int x, const int y)
- void m_scanPrintCMD (const char *s, const int x, const int y, const char *c)
- void m_scanPrintL1A (const int index, const int start[2], const int stop[2], const int bcid, const int l1id, const int ttid)

Private Attributes

- UINT16 m_buffer [TIM_SEQ_SIZE]
- INT32 m_l1id
- int m_lastBCR
- int m_size
- int m_offset
- int m_scanErrors

7.6.1 Detailed Description

TimSequin: A class for TIM Sequencer information.

This is useful for building TIM Sequences.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Definition at line 67 of file TimSequin.h.

7.6.2 Constructor & Destructor Documentation

7.6.2.1 SctPixelRod::TimSequin::TimSequin ()

TimSequin: A class for TIM Sequencer.

This is useful for loading the TIM Sequencer.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Definition at line 46 of file TimSequin.cxx.

References reset.

7.6.2.2 SctPixelRod::TimSequin::~TimSequin ()

Definition at line 53 of file TimSequin.cxx.

7.6.3 Member Function Documentation

7.6.3.1 void SctPixelRod::TimSequin::addByBunch (const TimBitBackplane *mask*, const int *bcid*)

Definition at line 90 of file TimSequin.cxx.

References addByIndex, getIndex, m_lastBCR, m_offset, SctPixelRod::NONE, and SctPixelRod::TimBitBackplane.

7.6.3.2 void SctPixelRod::TimSequin::addByIndex (const TimBitBackplane *mask*, const int *index*)

Definition at line 101 of file TimSequin.cxx.

References addTrigger, getBCID, m_buffer, m_l1id, m_lastBCR, m_offset, m_size, SctPixelRod::NONE, SctPixelRod::TIM_BCR, SctPixelRod::TIM_CAL, SctPixelRod::TIM_ECR, SctPixelRod::TIM_L1A, SctPixelRod::TIM_L1ID_FIRST, and SctPixelRod::TimBitBackplane.

Referenced by addByBunch.

7.6.3.3 void SctPixelRod::TimSequin::addTrigger (const int *index*, const UINT32 *ids*[3], const int *delays*[2])

Parameters:

index Index of L1A in data buffer

Definition at line 133 of file TimSequin.cxx.

References getBCID, m_buffer, m_lastBCR, m_offset, m_size, SctPixelRod::NONE, SctPixelRod::s_bits, SctPixelRod::s_masks, SctPixelRod::s_words, SctPixelRod::TIM_L1A, and SctPixelRod::TIM_SEQ_SIZE.

Referenced by addByIndex.

7.6.3.4 void SctPixelRod::TimSequin::fill (const UINT16 *size*, const UINT16 *value*)

Definition at line 184 of file TimSequin.cxx.

References m_buffer, and SctPixelRod::TIM_SEQ_SIZE.

7.6.3.5 int SctPixelRod::TimSequin::getBCID (const int *iL1A*, const int *iBCR*, const int *offset*)

Definition at line 195 of file TimSequin.cxx.

Referenced by addByIndex, addTrigger, and scan.

7.6.3.6 int SctPixelRod::TimSequin::getBuffer (const UINT16 *size*, UINT16 *buffer*[ ])

Definition at line 62 of file TimSequin.cxx.

References m_buffer, and m_size.

7.6.3.7 int SctPixelRod::TimSequin::getIndex (const int *bcid*, const int *iBCR*, const int *offset*)

Definition at line 202 of file TimSequin.cxx.

Referenced by addByBunch.

7.6.3.8 int SctPixelRod::TimSequin::getLength () [inline]

Definition at line 82 of file TimSequin.h.

References m_size.

7.6.3.9 void SctPixelRod::TimSequin::m_scanError (const char * *s*, const int *x*, const int *y*) [private]

Definition at line 461 of file TimSequin.cxx.

References m_scanErrors.

Referenced by scan.

7.6.3.10 void SctPixelRod::TimSequin::m_scanPrintCMD (const char * *s*, const int *x*, const int *y*, const char * *c*) [private]

Definition at line 473 of file TimSequin.cxx.

Referenced by m_scanPrintL1A, and scan.

7.6.3.11 void SctPixelRod::TimSequin::m_scanPrintL1A (const int *index*, const int *start*[2], const int *stop*[2], const int *bcid*, const int *l1id*, const int *ttid*) [private]

Definition at line 483 of file TimSequin.cxx.

References m_scanPrintCMD.

Referenced by scan.

7.6.3.12 void SctPixelRod::TimSequin::reset ()

Definition at line 211 of file TimSequin.cxx.

References m_buffer, m_l1id, m_lastBCR, m_offset, m_size, SctPixelRod::NONE, SctPixelRod::TIM_BCID_OFFSET, SctPixelRod::TIM_L1ID_FIRST, and SctPixelRod::TIM_SEQ_SIZE.

Referenced by TimSequin.

7.6.3.13 TimScanResults SctPixelRod::TimSequin::scan (const TimScanControl *control*)

Definition at line 223 of file TimSequin.cxx.

References getBCID, m_buffer, m_scanError, m_scanErrors, m_scanPrintCMD, m_scanPrintL1A, SctPixelRod::NONE, SctPixelRod::s_bits, SctPixelRod::s_masks, SctPixelRod::s_words, SctPixelRod::TIM_BCR, SctPixelRod::TIM_BCR_DEADTIME, SctPixelRod::TIM_CAL, SctPixelRod::TIM_CAL_DEADTIME, SctPixelRod::TIM_CMD, SctPixelRod::TIM_ECR, SctPixelRod::TIM_ECR_DEADTIME, SctPixelRod::TIM_L1A, SctPixelRod::TIM_L1A_DEADTIME, SctPixelRod::TIM_L1ID_FIRST, SctPixelRod::TIM_RES, SctPixelRod::TIM_SID, and SctPixelRod::TIM_STT.

Referenced by scanRun.

7.6.3.14 TimScanControl SctPixelRod::TimSequin::scanDefaults ()

Definition at line 419 of file TimSequin.cxx.

References m_offset, SctPixelRod::NONE, SctPixelRod::TIM_CAL_DEADTIME, and SctPixelRod::TIM_SEQ_SIZE.

Referenced by scanRun.

7.6.3.15 int SctPixelRod::TimSequin::scanRun ()

Definition at line 438 of file TimSequin.cxx.

References m_lastBCR, m_offset, m_size, scan, scanDefaults, and SctPixelRod::TIM_CAL_DEADTIME.

7.6.3.16 int SctPixelRod::TimSequin::setBuffer (const UINT16 *size*, const UINT16 *buffer*[ ])

Definition at line 74 of file TimSequin.cxx.

References m_buffer, m_size, and SctPixelRod::TIM_SEQ_SIZE.

7.6.3.17 void SctPixelRod::TimSequin::setOffset (const int *offset*) [inline]

Definition at line 83 of file TimSequin.h.

References m_offset.

7.6.4 Member Data Documentation

7.6.4.1 UINT16 SctPixelRod::TimSequin::m_buffer[TIM_SEQ_SIZE] [private]

Definition at line 110 of file TimSequin.h.

Referenced by addByIndex, addTrigger, fill, getBuffer, reset, scan, and setBuffer.

7.6.4.2 INT32 SctPixelRod::TimSequin::m_l1id [private]

Definition at line 112 of file TimSequin.h.

Referenced by addByIndex, and reset.

7.6.4.3 int SctPixelRod::TimSequin::m_lastBCR [private]

Definition at line 113 of file TimSequin.h.

Referenced by addByBunch, addByIndex, addTrigger, reset, and scanRun.

7.6.4.4 int SctPixelRod::TimSequin::m_offset [private]

Definition at line 115 of file TimSequin.h.

Referenced by addByBunch, addByIndex, addTrigger, reset, scanDefaults, scanRun, and setOffset.

7.6.4.5 int SctPixelRod::TimSequin::m_scanErrors [private]

Definition at line 116 of file TimSequin.h.

Referenced by m_scanError, and scan.

7.6.4.6 int SctPixelRod::TimSequin::m_size [private]

Definition at line 114 of file TimSequin.h.

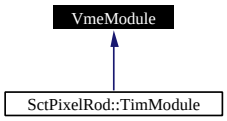
Referenced by addByIndex, addTrigger, getBuffer, getLength, reset, scanRun, and setBuffer.

The documentation for this class was generated from the following files:

- TimSequin.h
- TimSequin.cxx

7.7 VmeModule Class Reference

Inheritance diagram for VmeModule:



The documentation for this class was generated from the following file:

- TimModule.h

8 TimModule File Documentation

8.1 DoxyMain.h File Reference

Experimental Main Page for Doxygen.

8.1.1 Detailed Description

Experimental Main Page for Doxygen.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

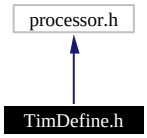
Definition in file DoxyMain.h.

8.2 TimDefine.h File Reference

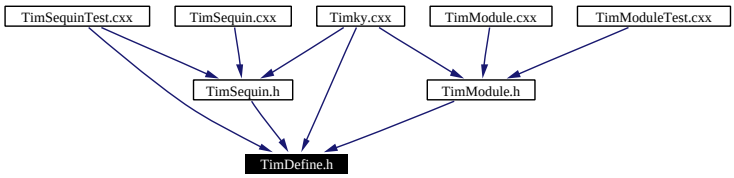
TimDefine.h: An incomplete prototype definition of a TIM.

```
#include "processor.h"
```

Include dependency graph for TimDefine.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace SctPixelRod

8.2.1 Detailed Description

TimDefine.h: An incomplete prototype definition of a TIM.

NB define eg I_AM_LINUX_HOST for processor.h (eg typedef UINT32)

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Id:
TimDefine.h,v 1.4 2003/06/04 15:04:32 tmeyer Exp

Log:
TimDefine.h,v

Revision 1.4 2003/06/04 15:04:32 tmeyer Removed explicit directory structure from includes

Revision 1.3 2002/12/11 21:30:49 jbl TimModule major update

Reference: http://www.hep.ucl.ac.uk/~jbl/SCT/TIM_registers.html

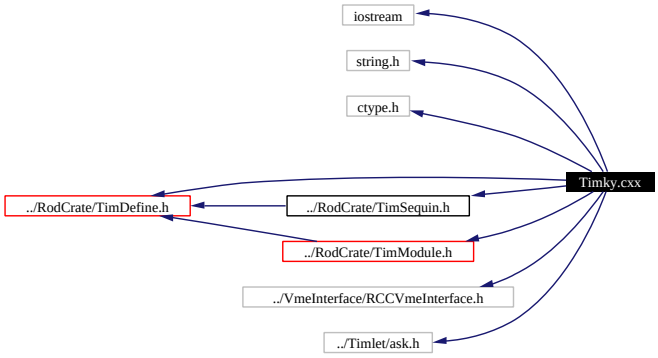
Definition in file TimDefine.h.

8.3 Timky.cxx File Reference

Execute TIM keywords.

```
#include <iostream>
#include <string.h>
#include <ctype.h>
#include "../RodCrate/TimDefine.h"
#include "../RodCrate/TimSequin.h"
#include "../RodCrate/TimModule.h"
#include "../VmeInterface/RCCVmeInterface.h"
#include "../Timlet/ask.h"
```

Include dependency graph for Timky.cxx:



Defines

- #define MAXLINE 128

Enumerations

- enum name { NONE = -999 }

Functions

- int damon (const char *FileName, unsigned short *buffer)
- int winky (const char *FileName)
- int open_window (const int Addr, const int Mode)
- FILE * open_file (const char *FileName, const char *FileMode)
- int close_file (FILE *fp, const char *FileName)
- void test (void)
- void timKeyword (const char *String)

- int main (int argc, char *argv[ ])
- unsigned short * vme_get_window (const int dummy0, const int dummy1, const int dummy2)

Variables

- const UINT32 baseAddr = 0x0D000000
- const UINT32 mapSize = 0x10000
- VmeInterface * vme = new RCCVmeInterface()
- TimModule * tim = new TimModule(baseAddr, mapSize, *vme)
- TimSequin * seq = new TimSequin()

8.3.1 Detailed Description

Execute TIM keywords.

This program executes the relevant method for TIM keywords, which may be read from a file or interactively.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

$Header$

Definition in file Timky.cxx.

8.3.2 Define Documentation

8.3.2.1 #define MAXLINE 128

Definition at line 39 of file Timky.cxx.

Referenced by test, timKeyword, and winky.

8.3.3 Enumeration Type Documentation

8.3.3.1 enum name

Enumeration values:
NONE

Definition at line 37 of file Timky.cxx.

8.3.4 Function Documentation

8.3.4.1 int close_file (FILE * *fp*, const char * *FileName*)

Definition at line 123 of file Timky.cxx.

Referenced by damon, and winky.

8.3.4.2 int damon (const char * *FileName*, unsigned short * *buffer*)

Definition at line 180 of file Timky.cxx.

References close_file, open_file, SctPixelRod::TIM_BCR, SctPixelRod::TIM_CAL, SctPixelRod::TIM_ECR, SctPixelRod::TIM_L1A, SctPixelRod::TIM_SEQ_SIZE, SctPixelRod::TIM_SID, SctPixelRod::TIM_STT, and timKeyword.

Referenced by timKeyword.

8.3.4.3 int main (int *argc*, char * *argv*[ ])

Definition at line 60 of file Timky.cxx.

References seq, test, tim, vme, and winky.

8.3.4.4 FILE * open_file (const char * *FileName*, const char * *FileMode*)

Definition at line 108 of file Timky.cxx.

Referenced by damon, and winky.

8.3.4.5 int open_window (const int *Addr*, const int *Mode*)

Definition at line 94 of file Timky.cxx.

References tim.

Referenced by timKeyword.

8.3.4.6 void test (void)

Definition at line 230 of file Timky.cxx.

References MAXLINE, tim, and timKeyword.

Referenced by main.

8.3.4.7 void timKeyword (const char * *String*)

Definition at line 247 of file Timky.cxx.

References damon, mapSize, MAXLINE, SctPixelRod::Name, NONE, open_window, seq, SIZE, tim, SctPixelRod::TIM_BCR, SctPixelRod::TIM_CAL, SctPixelRod::TIM_ECR, SctPixelRod::TIM_L1A, SctPixelRod::TIM_SEQ_SIZE, and winky.

Referenced by damon, test, and winky.

8.3.4.8 unsigned short* vme_get_window (const int *dummy0*, const int *dummy1*, const int *dummy2*)

Definition at line 89 of file Timky.cxx.

8.3.4.9 int winky (const char * *FileName*)

Definition at line 134 of file Timky.cxx.

References close_file, MAXLINE, open_file, and timKeyword.

Referenced by main, and timKeyword.

8.3.5 Variable Documentation

8.3.5.1 const UINT32 baseAddr = 0x0D000000

Definition at line 53 of file Timky.cxx.

Referenced by test, and SctPixelRod::TimModule::TimModule.

8.3.5.2 const UINT32 mapSize = 0x10000

Definition at line 54 of file Timky.cxx.

Referenced by test, timKeyword, and SctPixelRod::TimModule::TimModule.

8.3.5.3 TimSequin* seq = new TimSequin()

Definition at line 58 of file Timky.cxx.

Referenced by main, and timKeyword.

8.3.5.4 TimModule* tim = new TimModule(baseAddr, mapSize, *vme)

Definition at line 57 of file Timky.cxx.

Referenced by main, open_window, test, and timKeyword.

8.3.5.5 VmeInterface* vme = new RCCVmeInterface()

Definition at line 56 of file Timky.cxx.

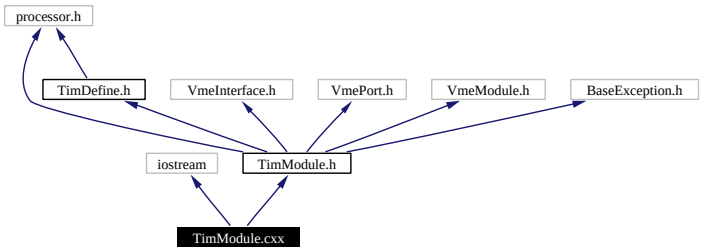
Referenced by main, test, and SctPixelRod::TimModule::TimModule.

8.4 TimModule.cxx File Reference

TimModule: A derived class for VME TIM modules.

```
#include <iostream>
#include "TimModule.h"
```

Include dependency graph for TimModule.cxx:



Namespaces

- namespace SctPixelRod
- namespace std

8.4.1 Detailed Description

TimModule: A derived class for VME TIM modules.

This is the implementation of a TIM class derived from the VmeModule base class. It should be the sole interface for VME communication with TIM.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Id:
TimModule.cxx,v 1.6 2004/04/01 13:26:04 gallop Exp

Log:
TimModule.cxx,v

Revision 1.6 2004/04/01 13:26:04 gallop BJG 1/4/04 Restore decimal mode after status

Revision 1.5 2003/12/04 19:10:49 jbl TimModule uses BaseException

Revision 1.4 2003/05/20 19:26:25 jbl TimModule UINT8 & UINT16

Revision 1.3 2002/12/11 21:30:50 jbl TimModule major update

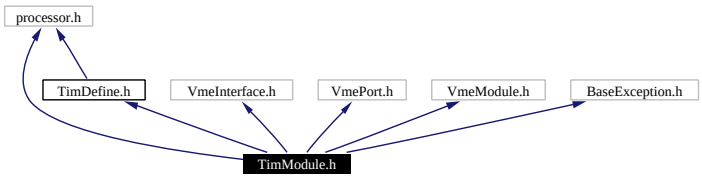
Definition in file TimModule.cxx.

8.5 TimModule.h File Reference

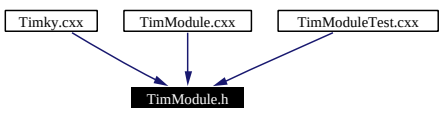
TimModule: A derived class for VME TIM modules.

```
#include "TimDefine.h"
#include "processor.h"
#include "VmeInterface.h"
#include "VmePort.h"
#include "VmeModule.h"
#include "BaseException.h"
```

Include dependency graph for TimModule.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace SctPixelRod
- namespace std

8.5.1 Detailed Description

TimModule: A derived class for VME TIM modules.

This file declares a TIM class derived from the VmeModule base class.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Id:
TimModule.h,v 1.7 2003/12/04 19:10:49 jbl Exp

Log:
TimModule.h,v

Revision 1.7 2003/12/04 19:10:49 jbl TimModule uses BaseException

Revision 1.6 2003/06/04 15:04:47 tmeyer Removed explicit directory structure from includes

Revision 1.5 2003/05/20 19:26:25 jbl TimModule UINT8 & UINT16

Revision 1.4 2002/12/11 21:30:50 jbl TimModule major update

NB define eg I_AM_LINUX_HOST for processor.h (eg typedef UINT32)

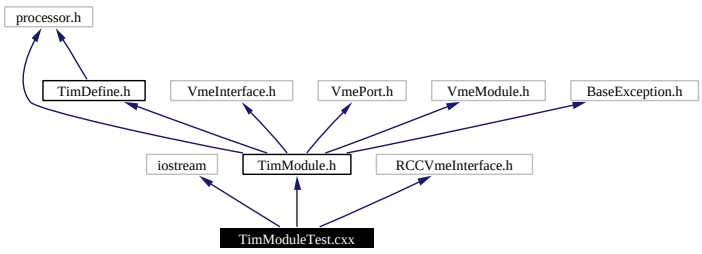
Definition in file TimModule.h.

8.6 TimModuleTest.cxx File Reference

Test TimModule.

```
#include <iostream>
#include "TimModule.h"
#include "RCCVmeInterface.h"
```

Include dependency graph for TimModuleTest.cxx:



Functions

- void test ()
- int main ()

8.6.1 Detailed Description

Test TimModule.

This is a prototype test program for TimModule.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Id:
TimModuleTest.cxx,v 1.4 2003/09/11 14:44:21 pixeldaq Exp

Log:
TimModuleTest.cxx,v

Revision 1.4 2003/09/11 14:44:21 pixeldaq Removed path info from includes

Revision 1.3 2002/12/11 21:30:50 jbl TimModule major update

Definition in file TimModuleTest.cxx.

8.6.2 Function Documentation

8.6.2.1 int main ()

Definition at line 30 of file TimModuleTest.cxx.

References test.

8.6.2.2 void test ()

Definition at line 66 of file TimModuleTest.cxx.

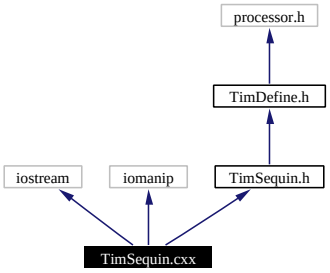
References baseAddr, mapSize, tim, SctPixelRod::TIM_BCR, SctPixelRod::TIM_MASK_TRIG_10_-0KHZ, SctPixelRod::TIM_REG_TIM_ID, SctPixelRod::TIM_SEQ_SIZE, SctPixelRod::TIM_VTRG, and vme.

8.7 TimSequin.cxx File Reference

A TIM Sequencer information class.

```
#include <iostream>
#include <iomanip>
#include "TimSequin.h"
```

Include dependency graph for TimSequin.cxx:



Namespaces

- namespace SctPixelRod

8.7.1 Detailed Description

A TIM Sequencer information class.

This is the implementation of a TIM Sequencer information class. It has no VME communication with TIM.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Id:
TimSequin.cxx,v 1.1 2004/06/07 19:12:41 jbl Exp

Log:
TimSequin.cxx,v

Revision 1.1 2004/06/07 19:12:41 jbl TimSequin first version

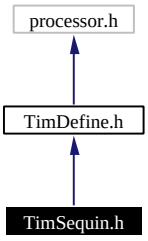
Definition in file TimSequin.cxx.

8.8 TimSequin.h File Reference

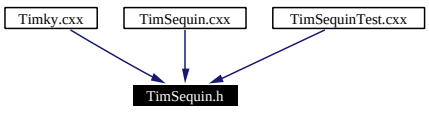
Declare a TIM Sequencer information class.

```
#include "TimDefine.h"
```

Include dependency graph for TimSequin.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace SctPixelRod

8.8.1 Detailed Description

Declare a TIM Sequencer information class.

This file declares a TIM class for Sequencer information. It has no VME communication with TIM.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Id:
TimSequin.h,v 1.1 2004/06/07 19:12:42 jbl Exp

Log:
TimSequin.h,v

Revision 1.1 2004/06/07 19:12:42 jbl TimSequin first version

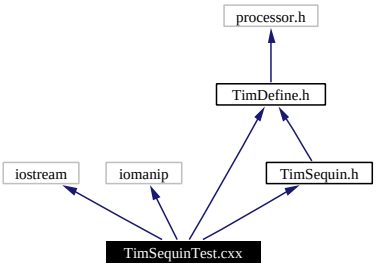
Definition in file TimSequin.h.

8.9 TimSequinTest.cxx File Reference

Test TimSequin class.

```
#include <iostream>
#include <iomanip>
#include "TimDefine.h"
#include "TimSequin.h"
```

Include dependency graph for TimSequinTest.cxx:



Functions

- void testscan ()
- void scanner (const int size, TimScanControl control)
- int main ()

Variables

- const int SIZE = 106
- TimSequin * seq = new TimSequin()

8.9.1 Detailed Description

Test TimSequin class.

This is a test program for TIM Sequencer information stuff. In particular it tests the scan code.

Contributors: John Lane <jbl@hep.ucl.ac.uk> - originator

Id:
TimSequinTest.cxx,v 1.1 2004/06/07 19:16:47 jbl Exp

Log:
TimSequinTest.cxx,v

Revision 1.1 2004/06/07 19:16:47 jbl TimSequin first version

Definition in file TimSequinTest.cxx.

8.9.2 Function Documentation

8.9.2.1 int main ()

Definition at line 34 of file TimSequinTest.cxx.

References seq, and testscan.

8.9.2.2 void scanner (const int *size*, TimScanControl *control*)

Definition at line 99 of file TimSequinTest.cxx.

References seq, and SIZE.

Referenced by testscan.

8.9.2.3 void testscan ()

Definition at line 42 of file TimSequinTest.cxx.

References scanner, seq, SIZE, SctPixelRod::TIM_BCID_OFFSET, SctPixelRod::TIM_BCR, SctPixelRod::TIM_BCR_DEADTIME, SctPixelRod::TIM_CAL, SctPixelRod::TIM_CAL_DEADTIME, SctPixelRod::TIM_FER, and SctPixelRod::TIM_L1A_DEADTIME.

Referenced by main.

8.9.3 Variable Documentation

8.9.3.1 TimSequin* seq = new TimSequin()

Definition at line 29 of file TimSequinTest.cxx.

Referenced by main, scanner, and testscan.

8.9.3.2 const int SIZE = 106

Definition at line 27 of file TimSequinTest.cxx.

Referenced by scanner, testscan, and timKeyword.

Index