

# TimModule Reference Manual

## 0.2

Generated by Doxygen 1.2.14

Sun Jun 27 19:57:31 2004

## Contents

<b>1</b>	<b><a href="#">TimModule Documentation</a></b>	<b>1</b>
<b>2</b>	<b><a href="#">TimModule Namespace Index</a></b>	<b>1</b>
<b>3</b>	<b><a href="#">TimModule Hierarchical Index</a></b>	<b>2</b>
<b>4</b>	<b><a href="#">TimModule Compound Index</a></b>	<b>2</b>
<b>5</b>	<b><a href="#">TimModule File Index</a></b>	<b>2</b>
<b>6</b>	<b><a href="#">TimModule Namespace Documentation</a></b>	<b>3</b>
<b>7</b>	<b><a href="#">TimModule Class Documentation</a></b>	<b>9</b>
<b>8</b>	<b><a href="#">TimModule File Documentation</a></b>	<b>26</b>

## 1 TimModule Documentation

### Author:

John Lane <[jbl@hep.ucl.ac.uk](mailto:jbl@hep.ucl.ac.uk)>

This is an experimental Main Page for Doxygen

NB define eg LAM\_LINUX\_HOST for processor.h (eg typedef UINT32)

Here is somewhere to start:

- [TimModule.h](#)

Contributors: John Lane <[jbl@hep.ucl.ac.uk](mailto:jbl@hep.ucl.ac.uk)> - originator

Reference: [http://www.hep.ucl.ac.uk/~jbl/SCT/TIM\\_registers.html](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:

<b><a href="#">SctPixelRod</a></b>	<b>3</b>
<b><a href="#">std (Overloaded operator to print TIM status)</a></b>	<b>8</b>

## 3 TimModule Hierarchical Index

### 3.1 TimModule Class Hierarchy

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

<b>BaseException</b>	<b>9</b>
<b>SctPixelRod::TimException</b>	<b>9</b>
<b>SctPixelRod::TimScanControl</b>	<b>18</b>
<b>SctPixelRod::TimScanResults</b>	<b>20</b>
<b>SctPixelRod::TimSequin</b>	<b>21</b>
<b>VmeModule</b>	<b>26</b>
<b>SctPixelRod::TimModule</b>	<b>11</b>

## 4 TimModule Compound Index

### 4.1 TimModule Compound List

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

<b>BaseException</b>	<b>9</b>
<b>SctPixelRod::TimException</b> ( <b>TimException</b> : a general exception class for TIM errors)	<b>9</b>
<b>SctPixelRod::TimModule</b> ( <b>TimModule</b> : A derived class for VME TIM modules)	<b>11</b>
<b>SctPixelRod::TimScanControl</b> (Structure definition for control of scanning Sequencer buffer)	<b>18</b>
<b>SctPixelRod::TimScanResults</b> (Structure definition for results of scanning Sequencer buffer)	<b>20</b>
<b>SctPixelRod::TimSequin</b> ( <b>TimSequin</b> : A class for TIM Sequencer information)	<b>21</b>
<b>VmeModule</b>	<b>26</b>

## 5 TimModule File Index

### 5.1 TimModule File List

Here is a list of all files with brief descriptions:

<b>DoxyMain.h</b> (Experimental Main Page for Doxygen)	<b>26</b>
<b>TimDefine.h</b> ( <b>TimDefine.h</b> : An incomplete prototype definition of a TIM)	<b>27</b>
<b>Timky.cxx</b> (Execute TIM keywords)	<b>28</b>

<a href="#">TimModule.cxx</a> (TimModule: A derived class for VME TIM modules)	31
<a href="#">TimModule.h</a> (TimModule: A derived class for VME TIM modules)	32
<a href="#">TimModuleTest.cxx</a> (Test TimModule)	33
<a href="#">TimSequin.cxx</a> (A TIM Sequencer information class)	34
<a href="#">TimSequin.h</a> (Declare a TIM Sequencer information class)	35
<a href="#">TimSequinTest.cxx</a> (Test TimSequin class)	36

## 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 ECRreset.
- TIM\_BIT\_EN\_INT\_BCR** Enable internal repetitive BCRreset.
- TIM\_BIT\_EN\_RANDOM** Enable internal trigger randomizer.
- TIM\_BIT\_EN\_INT\_FER** Enable internal repetitive FERreset.

**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: ECRreset.  
**TIM\_BIT\_EN\_EXT\_BCR** Enable external inputs: BCRreset.  
**TIM\_BIT\_EN\_EXT\_CAL** Enable external inputs: Calibrate.  
**TIM\_BIT\_EN\_EXT\_FER** Enable external inputs: FERreset.  
**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_LIID_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, `daemon`, 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

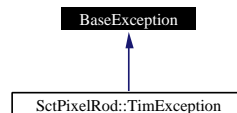
#### 6.2.2.1 ostream & std::operator<< (ostream & os, SctPixelRod::TimModule & tim)

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:

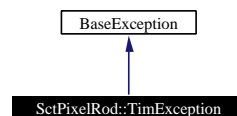
- [TimModule.h](#)

### 7.2 SctPixelRod::TimException Class Reference

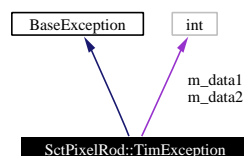
`TimException`: a general exception class for TIM errors.

```
#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](mailto: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:

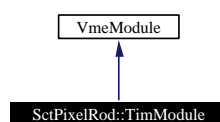
- [TimModule.h](#)
- [TimModule.cxx](#)

## 7.3 SctPixelRod::TimModule Class Reference

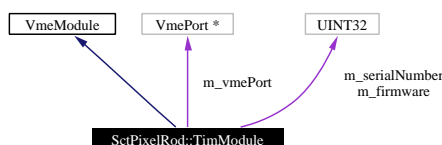
**TimModule**: A derived class for VME TIM modules.

```
#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 + LIA 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](mailto: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 fetchLIID, 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:

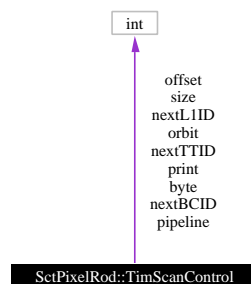
- [TimModule.h](#)
- [TimModule.cxx](#)

## 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:

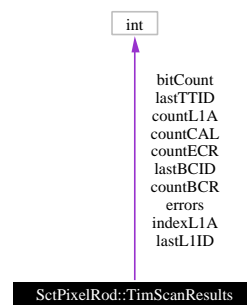
- [TimSequin.h](#)

## 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 [lastLID](#)
- int [lastBCID](#)
- int [lastTTID](#)
- int [indexLIA](#)
- int [countLIA](#)
- 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:

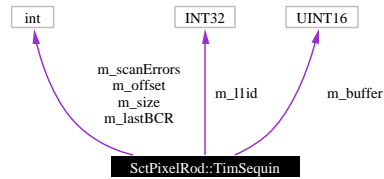
- [TimSequin.h](#)

**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 llid, const int ttid\)](#)

## Private Attributes

- [UINT16 m\\_buffer \[TIM\\_SEQ\\_SIZE\]](#)
- [INT32 m\\_llid](#)
- [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 iLIA, 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 liid, 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.

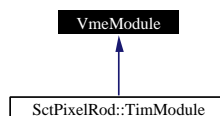
Referenced by addByIndex, addTrigger, getBuffer, getLength, reset, scanRun, and setBuffer.

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](mailto:jbl@hep.ucl.ac.uk)> - originator

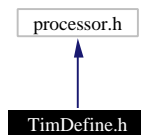
Definition in file [DoxyMain.h](#).

## 8.2 TimDefine.h File Reference

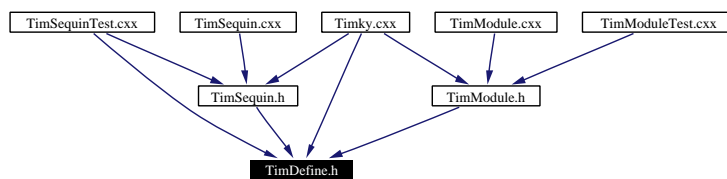
**TimDefine.h:** An incomplete prototype definition of a TIM.

```
#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 LAM\_LINUX\_HOST for processor.h (eg typedef UINT32)

Contributors: John Lane <[jbl@hep.ucl.ac.uk](mailto: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](http://www.hep.ucl.ac.uk/~jbl/SCT/TIM_registers.html)

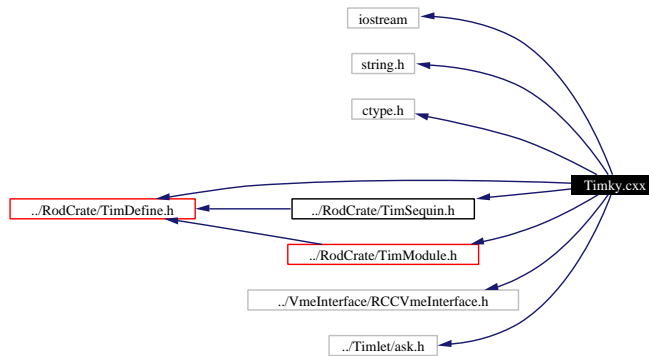
Definition in file [TimDefine.h](#).

### 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](mailto: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.

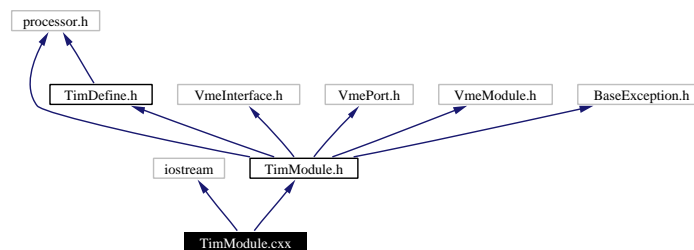
Referenced by main, test, and SctPixelRod::TimModule::TimModule.

## 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](mailto: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

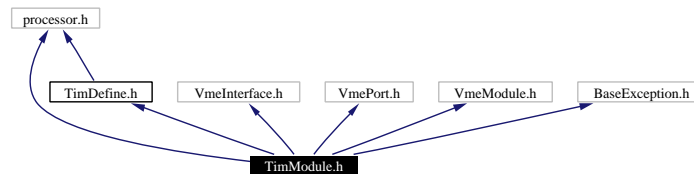
Definition in file [TimModule.cxx](#).

## 8.5 TimModule.h File Reference

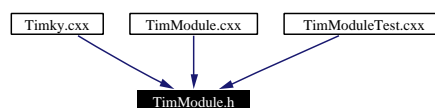
TimModule: A derived class for VME TIM modules.

```
#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](mailto: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 LAM\_LINUX\_HOST for processor.h (eg typedef UINT32)

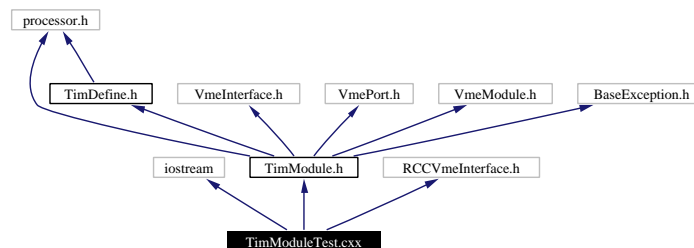
Definition in file [TimModule.h](#).

## 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

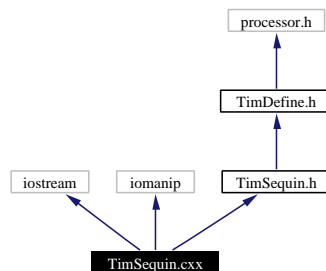
A TIM Sequencer information class.

```
#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](mailto: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

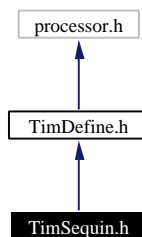
Definition in file [TimSequin.cxx](#).

## 8.8 TimSequin.h File Reference

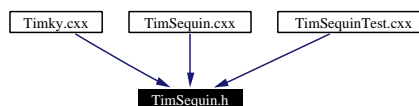
Declare a TIM Sequencer information class.

```
#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

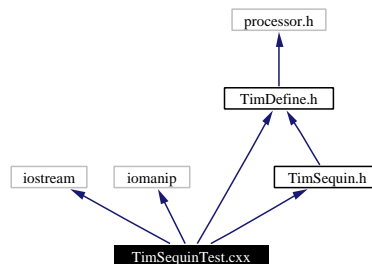
Definition in file [TimSequin.h](#).

## 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

- ~TimModule
    - SctPixelRod::TimModule, [13](#)
  - ~TimSequin
    - SctPixelRod::TimSequin, [23](#)
  - addByBunch
    - SctPixelRod::TimSequin, [23](#)
  - addByIndex
    - SctPixelRod::TimSequin, [23](#)
  - addTrigger
    - SctPixelRod::TimSequin, [23](#)
  
  - baseAddr
    - Timky.cxx, [31](#)
  - BaseException, [9](#)
  - bitCount
    - SctPixelRod::TimScanResults, [20](#)
  - byte
    - SctPixelRod::TimScanControl, [19](#)
  
  - close\_file
    - Timky.cxx, [29](#)
  - countBCR
    - SctPixelRod::TimScanResults, [20](#)
  - countCAL
    - SctPixelRod::TimScanResults, [21](#)
  - countECR
    - SctPixelRod::TimScanResults, [21](#)
  - countL1A
    - SctPixelRod::TimScanResults, [21](#)
  
  - damon
    - Timky.cxx, [29](#)
  - DoxyMain.h, [26](#)
  
  - errors
    - SctPixelRod::TimScanResults, [21](#)
  
  - fetchL1ID
    - SctPixelRod::TimModule, [14](#)
  - fetchTimID
    - SctPixelRod::TimModule, [14](#)
  - fill
    - SctPixelRod::TimSequin, [24](#)
  
  - getBCID
    - SctPixelRod::TimSequin, [24](#)
  - getBuffer
    - SctPixelRod::TimSequin, [24](#)
  - getData1
    - SctPixelRod::TimException, [10](#)
  - getData2
    - SctPixelRod::TimException, [10](#)
  - getFirmware
    - SctPixelRod::TimModule, [14](#)
  - getIndex
    - SctPixelRod::TimSequin, [24](#)
  - getLength
    - SctPixelRod::TimSequin, [24](#)
  - getSerialNumber
    - SctPixelRod::TimModule, [14](#)
  - getVmePort
    - SctPixelRod::TimModule, [14](#)
  
  - indexL1A
    - SctPixelRod::TimScanResults, [21](#)
  - initialize
    - SctPixelRod::TimModule, [14](#)
  - intTrigStart
    - SctPixelRod::TimModule, [14](#)
  - intTrigStop
    - SctPixelRod::TimModule, [15](#)
  - issueCommand
    - SctPixelRod::TimModule, [15](#)
  - issueVCAL
    - SctPixelRod::TimModule, [15](#)
  
  - lastBCID
    - SctPixelRod::TimScanResults, [21](#)
  - lastL1ID
    - SctPixelRod::TimScanResults, [21](#)
  - lastTTID
    - SctPixelRod::TimScanResults, [21](#)
  - loadBitClear
    - SctPixelRod::TimModule, [15](#)
  - loadBitSet
    - SctPixelRod::TimModule, [15](#)
  - loadByteHi
    - SctPixelRod::TimModule, [16](#)
  - loadByteLo
    - SctPixelRod::TimModule, [16](#)
  
  - m\_buffer
    - SctPixelRod::TimSequin, [25](#)
  - m\_data1
    - SctPixelRod::TimException, [11](#)
  - m\_data2
    - SctPixelRod::TimException, [11](#)
  - m\_firmware
    - SctPixelRod::TimModule, [18](#)
  - m\_l1id
    - SctPixelRod::TimSequin, [25](#)
  - m\_lastBCR
-

- SctPixelRod::TimSequin, 26
- m\_offset
  - SctPixelRod::TimSequin, 26
- m\_scanError
  - SctPixelRod::TimSequin, 24
- m\_scanErrors
  - SctPixelRod::TimSequin, 26
- m\_scanPrintCMD
  - SctPixelRod::TimSequin, 24
- m\_scanPrintL1A
  - SctPixelRod::TimSequin, 24
- m\_serialNumber
  - SctPixelRod::TimModule, 18
- m\_size
  - SctPixelRod::TimSequin, 26
- m\_vmePort
  - SctPixelRod::TimModule, 18
- main
  - Timky.cxx, 30
  - TimModuleTest.cxx, 34
  - TimSequinTest.cxx, 37
- mapSize
  - Timky.cxx, 31
- MAXLINE
  - Timky.cxx, 29
- Name
  - SctPixelRod, 4
- name
  - Timky.cxx, 29
- nextBCID
  - SctPixelRod::TimScanControl, 19
- nextL1ID
  - SctPixelRod::TimScanControl, 19
- nextTTID
  - SctPixelRod::TimScanControl, 19
- NONE
  - SctPixelRod, 4
  - Timky.cxx, 29
- offset
  - SctPixelRod::TimScanControl, 19
- open\_file
  - Timky.cxx, 30
- open\_window
  - Timky.cxx, 30
- operator<<
  - std, 9
- orbit
  - SctPixelRod::TimScanControl, 19
- pipeline
  - SctPixelRod::TimScanControl, 19
- print
  - SctPixelRod::TimScanControl, 19
- regFetch
  - SctPixelRod::TimModule, 16
- regLoad
  - SctPixelRod::TimModule, 16
- reset
  - SctPixelRod::TimModule, 16
  - SctPixelRod::TimSequin, 24
- s\_bits
  - SctPixelRod, 8
- s\_masks
  - SctPixelRod, 8
- s\_words
  - SctPixelRod, 8
- scan
  - SctPixelRod::TimSequin, 25
- scanDefaults
  - SctPixelRod::TimSequin, 25
- scanner
  - TimSequinTest.cxx, 37
- scanRun
  - SctPixelRod::TimSequin, 25
- SctPixelRod
  - NONE, 4
  - TIM\_BCID\_OFFSET, 7
  - TIM\_BCR, 5
  - TIM\_BCR\_DEADTIME, 7
  - TIM\_BIT\_EN\_CYCLIC, 6
  - TIM\_BIT\_EN\_EXT\_BCR, 6
  - TIM\_BIT\_EN\_EXT\_BUSY, 6
  - TIM\_BIT\_EN\_EXT\_CAL, 6
  - TIM\_BIT\_EN\_EXT\_CLK, 6
  - TIM\_BIT\_EN\_EXT\_ECR, 6
  - TIM\_BIT\_EN\_EXT\_FER, 6
  - TIM\_BIT\_EN\_EXT\_SEQ, 6
  - TIM\_BIT\_EN\_EXT\_TRIG, 6
  - TIM\_BIT\_EN\_ID, 6
  - TIM\_BIT\_EN\_INT\_BCR, 5
  - TIM\_BIT\_EN\_INT\_BUSY, 6
  - TIM\_BIT\_EN\_INT\_ECR, 5
  - TIM\_BIT\_EN\_INT\_FER, 5
  - TIM\_BIT\_EN\_INT\_TRIG, 5
  - TIM\_BIT\_EN\_RANDOM, 5
  - TIM\_BIT\_EN\_TYPE, 6
  - TIM\_BIT\_EN\_WINDOW, 5
  - TIM\_BIT\_SEQ\_EN\_ALL, 6
  - TIM\_BIT\_SEQ\_GO, 6
  - TIM\_BIT\_SEQ\_RESET, 6
  - TIM\_BIT\_VRESET, 5
  - TIM\_CAL, 5
  - TIM\_CAL\_DEADTIME, 7
  - TIM\_CMD, 5



- TIM\_ECR, 5
- TIM\_ECR\_DEADTIME, 7
- TIM\_FER, 5
- TIM\_L1A, 5
- TIM\_L1A\_DEADTIME, 7
- TIM\_MASK\_FER\_0\_010HZ, 6
- TIM\_MASK\_FER\_0\_100HZ, 6
- TIM\_MASK\_FER\_10\_00HZ, 6
- TIM\_MASK\_FER\_1\_000HZ, 6
- TIM\_MASK\_TRIG\_0\_10KHZ, 6
- TIM\_MASK\_TRIG\_100\_KHZ, 6
- TIM\_MASK\_TRIG\_10\_0KHZ, 6
- TIM\_MASK\_TRIG\_1\_00KHZ, 6
- TIM\_REG\_BURST\_COUNT, 7
- TIM\_REG\_COMMAND, 7
- TIM\_REG\_DELAY, 7
- TIM\_REG\_ENABLES, 7
- TIM\_REG\_FIFO\_STATUS, 7
- TIM\_REG\_FREQUENCY, 7
- TIM\_REG\_OUTPUT, 7
- TIM\_REG\_ROD\_BUSY, 7
- TIM\_REG\_ROD\_LATCH, 7
- TIM\_REG\_ROD\_MASK, 7
- TIM\_REG\_ROD\_MONITOR, 7
- TIM\_REG\_RUN\_ENABLES, 7
- TIM\_REG\_SEQ\_CONTROL, 7
- TIM\_REG\_SEQ\_END, 7
- TIM\_REG\_STATUS, 7
- TIM\_REG\_TIM\_ID, 7
- TIM\_REG\_TRIGGER\_BCID, 7
- TIM\_REG\_TRIGGER\_IDHI, 7
- TIM\_REG\_TRIGGER\_IDLO, 7
- TIM\_REG\_TRIGGER\_TYPE, 7
- TIM\_REG\_TTC\_BCID, 7
- TIM\_REG\_TTC\_DATA, 7
- TIM\_REG\_TTC\_RX, 7
- TIM\_REG\_TTC\_SELECT, 7
- TIM\_REG\_TTC\_STATUS, 7
- TIM\_REG\_WINDOW, 7
- TIM\_RES, 5
- TIM\_SID, 5
- TIM\_SPA, 5
- TIM\_STT, 5
- TIM\_TRG, 5
- TIM\_VBCR, 5
- TIM\_VCAL, 5
- TIM\_VECR, 5
- TIM\_VFER, 5
- TIM\_VSPA, 5
- TIM\_VTRG, 5
- SctPixelRod, 3
  - Name, 4
  - s\_bits, 8
  - s\_masks, 8
  - s\_words, 8
- TIM\_L1ID\_FIRST, 8
- TIM\_SEQ\_ADDR, 8
- TIM\_SEQ\_SIZE, 8
- TimBitBackplane, 4
- TimBitCommand, 5
- TimBitEnables, 5
- TimBitRunEnables, 6
- TimBitSeqControl, 6
- TimMaskFrequency, 6
- TimRegister, 6
- TimTimingSCT, 7
- SctPixelRod::TimException, 9
  - getData1, 10
  - getData2, 10
  - m\_data1, 11
  - m\_data2, 11
  - TimException, 10
  - what, 10
- SctPixelRod::TimModule, 11
  - ~TimModule, 13
  - fetchL1ID, 14
  - fetchTimID, 14
  - getFirmware, 14
  - getSerialNumber, 14
  - getVmePort, 14
  - initialize, 14
  - intTrigStart, 14
  - intTrigStop, 15
  - issueCommand, 15
  - issueVCAL, 15
  - loadBitClear, 15
  - loadBitSet, 15
  - loadByteHi, 16
  - loadByteLo, 16
  - m\_firmware, 18
  - m\_serialNumber, 18
  - m\_vmePort, 18
  - regFetch, 16
  - regLoad, 16
  - reset, 16
  - seqFetch, 16
  - seqLoad, 17
  - seqRun, 17
  - status, 17
  - TimModule, 13
  - vmeFetch, 17
  - vmeLoad, 17
- SctPixelRod::TimScanControl, 18
  - byte, 19
  - nextBCID, 19
  - nextL1ID, 19
  - nextTTID, 19
  - offset, 19

- orbit, 19
- pipeline, 19
- print, 19
- size, 19
- SctPixelRod::TimScanResults, 20
  - bitCount, 20
  - countBCR, 20
  - countCAL, 21
  - countECR, 21
  - countL1A, 21
  - errors, 21
  - indexL1A, 21
  - lastBCID, 21
  - lastL1ID, 21
  - lastTTID, 21
- SctPixelRod::TimSequin, 21
  - ~TimSequin, 23
  - addByBunch, 23
  - addByIndex, 23
  - addTrigger, 23
  - fill, 24
  - getBCID, 24
  - getBuffer, 24
  - getIndex, 24
  - getLength, 24
  - m\_buffer, 25
  - m\_l1id, 25
  - m\_lastBCR, 26
  - m\_offset, 26
  - m\_scanError, 24
  - m\_scanErrors, 26
  - m\_scanPrintCMD, 24
  - m\_scanPrintL1A, 24
  - m\_size, 26
  - reset, 24
  - scan, 25
  - scanDefaults, 25
  - scanRun, 25
  - setBuffer, 25
  - setOffset, 25
  - TimSequin, 23
- seq
  - Timky.cxx, 31
  - TimSequinTest.cxx, 37
- seqFetch
  - SctPixelRod::TimModule, 16
- seqLoad
  - SctPixelRod::TimModule, 17
- seqRun
  - SctPixelRod::TimModule, 17
- setBuffer
  - SctPixelRod::TimSequin, 25
- setOffset
  - SctPixelRod::TimSequin, 25
- SIZE
  - TimSequinTest.cxx, 37
- size
  - SctPixelRod::TimScanControl, 19
- status
  - SctPixelRod::TimModule, 17
- std, 8
  - operator<<, 9
- test
  - Timky.cxx, 30
  - TimModuleTest.cxx, 34
- testscan
  - TimSequinTest.cxx, 37
- tim
  - Timky.cxx, 31
- TIM\_BCID\_OFFSET
  - SctPixelRod, 7
- TIM\_BCR
  - SctPixelRod, 5
- TIM\_BCR\_DEADTIME
  - SctPixelRod, 7
- TIM\_BIT\_EN\_CYCLIC
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_BCR
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_BUSY
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_CAL
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_CLK
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_ECR
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_FER
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_SEQ
  - SctPixelRod, 6
- TIM\_BIT\_EN\_EXT\_TRIG
  - SctPixelRod, 6
- TIM\_BIT\_EN\_ID
  - SctPixelRod, 6
- TIM\_BIT\_EN\_INT\_BCR
  - SctPixelRod, 5
- TIM\_BIT\_EN\_INT\_BUSY
  - SctPixelRod, 6
- TIM\_BIT\_EN\_INT\_ECR
  - SctPixelRod, 5
- TIM\_BIT\_EN\_INT\_FER
  - SctPixelRod, 5
- TIM\_BIT\_EN\_INT\_TRIG
  - SctPixelRod, 5
- TIM\_BIT\_EN\_RANDOM
  - SctPixelRod, 5

---

TIM_BIT_EN_TYPE	SctPixelRod, 6	TIM_REG_FIFO_STATUS	SctPixelRod, 7
TIM_BIT_EN_WINDOW	SctPixelRod, 5	TIM_REG_FREQUENCY	SctPixelRod, 7
TIM_BIT_SEQ_EN_ALL	SctPixelRod, 6	TIM_REG_OUTPUT	SctPixelRod, 7
TIM_BIT_SEQ_GO	SctPixelRod, 6	TIM_REG_ROD_BUSY	SctPixelRod, 7
TIM_BIT_SEQ_RESET	SctPixelRod, 6	TIM_REG_ROD_LATCH	SctPixelRod, 7
TIM_BIT_VRESET	SctPixelRod, 5	TIM_REG_ROD_MASK	SctPixelRod, 7
TIM_CAL	SctPixelRod, 5	TIM_REG_ROD_MONITOR	SctPixelRod, 7
TIM_CAL_DEADTIME	SctPixelRod, 7	TIM_REG_RUN_ENABLES	SctPixelRod, 7
TIM_CMD	SctPixelRod, 5	TIM_REG_SEQ_CONTROL	SctPixelRod, 7
TIM_ECR	SctPixelRod, 5	TIM_REG_SEQ_END	SctPixelRod, 7
TIM_ECR_DEADTIME	SctPixelRod, 7	TIM_REG_STATUS	SctPixelRod, 7
TIM_FER	SctPixelRod, 5	TIM_REG_TIM_ID	SctPixelRod, 7
TIM_L1A	SctPixelRod, 5	TIM_REG_TRIGGER_BCID	SctPixelRod, 7
TIM_L1A_DEADTIME	SctPixelRod, 7	TIM_REG_TRIGGER_IDHI	SctPixelRod, 7
TIM_L1ID_FIRST	SctPixelRod, 8	TIM_REG_TRIGGER_IDLO	SctPixelRod, 7
TIM_MASK_FER_0.010HZ	SctPixelRod, 6	TIM_REG_TRIGGER_TYPE	SctPixelRod, 7
TIM_MASK_FER_0.100HZ	SctPixelRod, 6	TIM_REG_TTC_BCID	SctPixelRod, 7
TIM_MASK_FER_10.00HZ	SctPixelRod, 6	TIM_REG_TTC_DATA	SctPixelRod, 7
TIM_MASK_FER_1.000HZ	SctPixelRod, 6	TIM_REG_TTC_RX	SctPixelRod, 7
TIM_MASK_TRIG_0.10KHZ	SctPixelRod, 6	TIM_REG_TTC_SELECT	SctPixelRod, 7
TIM_MASK_TRIG_100_KHZ	SctPixelRod, 6	TIM_REG_TTC_STATUS	SctPixelRod, 7
TIM_MASK_TRIG_10_0KHZ	SctPixelRod, 6	TIM_REG_WINDOW	SctPixelRod, 7
TIM_MASK_TRIG_1_00KHZ	SctPixelRod, 6	TIM_RES	SctPixelRod, 5
TIM_REG_BURST_COUNT	SctPixelRod, 7	TIM_SEQ_ADDR	SctPixelRod, 8
TIM_REG_COMMAND	SctPixelRod, 7	TIM_SEQ_SIZE	SctPixelRod, 8
TIM_REG_DELAY	SctPixelRod, 7	TIM_SID	SctPixelRod, 5
TIM_REG_ENABLES	SctPixelRod, 7	TIM_SPA	SctPixelRod, 5

- TIM\_STT
  - SctPixelRod, 5
- TIM\_TRG
  - SctPixelRod, 5
- TIM\_VBCR
  - SctPixelRod, 5
- TIM\_VCAL
  - SctPixelRod, 5
- TIM\_VECR
  - SctPixelRod, 5
- TIM\_VFER
  - SctPixelRod, 5
- TIM\_VSPA
  - SctPixelRod, 5
- TIM\_VTRG
  - SctPixelRod, 5
- TimBitBackplane
  - SctPixelRod, 4
- TimBitCommand
  - SctPixelRod, 5
- TimBitEnables
  - SctPixelRod, 5
- TimBitRunEnables
  - SctPixelRod, 6
- TimBitSeqControl
  - SctPixelRod, 6
- TimDefine.h, 27
- TimException
  - SctPixelRod::TimException, 10
- timKeyword
  - Timky.cxx, 30
- Timky.cxx, 28
  - baseAddr, 31
  - close\_file, 29
  - damon, 29
  - main, 30
  - mapSize, 31
  - MAXLINE, 29
  - name, 29
  - NONE, 29
  - open\_file, 30
  - open\_window, 30
  - seq, 31
  - test, 30
  - tim, 31
  - timKeyword, 30
  - vme, 31
  - vme\_get\_window, 30
  - winky, 30
- TimMaskFrequency
  - SctPixelRod, 6
- TimModule
  - SctPixelRod::TimModule, 13
- TimModule.cxx, 31
  - TimModule.h, 32
  - TimModuleTest.cxx, 33
    - main, 34
    - test, 34
  - TimRegister
    - SctPixelRod, 6
  - TimSequin
    - SctPixelRod::TimSequin, 23
  - TimSequin.cxx, 34
  - TimSequin.h, 35
  - TimSequinTest.cxx, 36
    - main, 37
    - scanner, 37
    - seq, 37
    - SIZE, 37
    - testscan, 37
  - TimTimingSCT
    - SctPixelRod, 7
- vme
  - Timky.cxx, 31
- vme\_get\_window
  - Timky.cxx, 30
- vmeFetch
  - SctPixelRod::TimModule, 17
- vmeLoad
  - SctPixelRod::TimModule, 17
- VmeModule, 26
- what
  - SctPixelRod::TimException, 10
- winky
  - Timky.cxx, 30