Galaxia Library Suite

201

Generated by Doxygen 1.8.11

Contents

Chapter 1

Todo List

Member SWItrigger::begin (uint8_t trigger, void(∗functionSWItrigger)())

The idea is to have the same function as for Wire library with Wire.onReceive(void (∗ReceiveEvent)(int)) and void ReceiveEvent(int howMany)

Chapter 2

Deprecated List

Class SWI

SWI is no longer inlcuded in RTOS for Energia 0101E0017

Class SWItrigger

SWI and thus SWItrigger are no longer inlcuded in RTOS for Energia 0101E0017

Chapter 3

Bug List

Member Clock::begin (void(∗ClockFunction)(void), uint32_t ClockTimeOut_ms, uint32_t ClockPeriod_ms=0)

Some functions like Serial.print(); don't work :(

Class HWI

Reuse of msp432/cores/msp432/WInterrupts.c based on ti/drivers/GPIO.h. HWI impementation?

Member SWItrigger::begin (uint8_t trigger, void(∗functionSWItrigger))

functionSWItrigger(uint8_t) with uint8_t = Swi_getTrigger() doesn't work.

Member Timer::begin (void(∗timerFunction)(void), uint32_t timerPeriod_unit, uint32_t unit=1000)

Some fucntions like Serial.print(); don't work :(

Chapter 4

Class Index

4.1 Class List

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

Chapter 5

File Index

5.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 6

Class Documentation

6.1 Clock Class Reference

RTOS Clock as an object.

`#include <Clock.h>`

Public Member Functions

- Clock ()
  Define the Clock.
- ~Clock ()
  Delete the Clock.
- void begin (void(*ClockFunction)(void), uint32_t ClockTimeOut_ms, uint32_t ClockPeriod_ms=0)
  Create the Clock.
- void start ()
  Start the Clock.
- void stop ()
  Stop the Clock.

6.1.1 Detailed Description

RTOS Clock as an object.

The RTOS Clock is encapsulated as a C++ object for easier use

Note

Multiple Clock objects possible.

6.1.2 Member Function Documentation

6.1.2.1 void Clock::begin (void(*)(void) *ClockFunction*, uint32_t *ClockTimeOut_ms*, uint32_t *ClockPeriod_ms* = 0)

Create the Clock.

Parameters

ClockFunction	function to be called
ClockTimeOut_ms	initial start delay
ClockPeriod_ms	period in ms, default = 0 for one-shot

Note

The function must be void ClockPeriod_ms()

```
void ClockPeriod_ms()
       {
           digitalWrite(RED_LED, HIGH);
       }
```

Bug Some functions like Serial.print(); don't work :(

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Clock.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Clock.cpp

6.2 Event Class Reference

RTOS Event as object.

`#include <Event.h>`

Public Member Functions

- Event ()
  Define the event.
- void begin ()
  Create the event.
- void send (xdc_UInt eventId_number=Event_Id_00)
  Raise the event.
- uint32_t waitFor (xdc_UInt andMask=Event_Id_00, xdc_UInt orMask=Event_Id_NONE)
  Wait for the event.

6.2.1 Detailed Description

RTOS Event as object.

The RTOS Event is encapsulated as a C++ object for easier use

Note

Only a single Task can pend on an Event object at a time.

6.2.2 Member Function Documentation

6.2.2.1 void Event::send (xdc_UInt *eventId_number* = `Event_Id_00`)

Raise the event.

Parameters

eventId_number	event identifier, default = Event_Id_00

Note

Take event identifier among Event_Id_00..Event_Id_31

6.2.2.2 uint32_t Event::waitFor (xdc_UInt *andMask* = `Event_Id_00`, xdc_UInt *orMask* = `Event_Id_NONE`)

Wait for the event.

Parameters

andMask	AND condition, default = Event_Id_00
orMask	OR condition, default = Event_Id_NONE

Returns

events identifiers as bit, 32-bit

```
0b=1100 = Event_Id_03 + Event_Id_02
```

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Event.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Event.cpp

6.3 HWI Class Reference

RTOS HWI as object.

```
#include <HWI.h>
```

Public Member Functions

- HWI ()

  Define the HWI.
- void begin (uint8_t pinNumber, void(*functionHWI)(void), int mode)

  Create the HWI.
- void clearInterrupt ()

  Clear the interrupt.

6.3.1 Detailed Description

RTOS HWI as object.

The RTOS HWI is encapsulated as a C++ object for easier use

Bug Reuse of msp432/cores/msp432/WInterrupts.c based on ti/drivers/GPIO.h. HWI impementation?

6.3.2 Member Function Documentation

6.3.2.1 void HWI::begin (uint8_t *pinNumber,* void(∗)(void) *functionHWI,* int *mode*)

Create the HWI.

Parameters

pinNumber	function to be called
functionHWI	function to be called
mode	LOW, CHANGE, FALLING, RISING

Note

Same as attachInterrupt()
The function must be void functionHWI()

```
void functionHWI()
       {
           digitalWrite(RED_LED, HIGH);
       }
```

6.3.2.2 void HWI::clearInterrupt ( )

Clear the interrupt.

Note

Same as detachInterrupt()

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/HWI.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/HWI.cpp

6.4 Mailbox< mailboxType > Class Template Reference

RTOS Mailbox as an object.

`#include <Mailbox.h>`

Public Member Functions

- Mailbox ()

  Define the mailbox.
- void begin (uint16_t number=16)

  Create the mailbox.
- bool post (mailboxType &message, uint16_t timeout=BIOS_WAIT_FOREVER)

  Post a message to the mailbox.
- void waitFor (mailboxType &message)

  Wait for a message from the mailbox.
- uint16_t available ()

  Available messages to be read.

6.4.1 Detailed Description

template<typename mailboxType>
class Mailbox< mailboxType >

RTOS Mailbox as an object.

Warning

Header and code for template class need to be on the same unique file. I guess it isn't a bug, but a feature :/

See also

http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2003/n1426.pdf

The RTOS Mailbox is encapsulated as a C++ object for easier use

Warning

Messages must be of type typename used in declaration

Note

Only a single task can pend on an Mailbox object at a time.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 template<typename mailboxType > Mailbox< mailboxType >::Mailbox ( )

Define the mailbox.

Warning

Specify typename between brackets in declaration.

```
Mailbox<typename> myMailbox;
```

6.4.3 Member Function Documentation

6.4.3.1 template<typename mailboxType > uint16_t Mailbox< mailboxType >::available ( )

Available messages to be read.

Returns

number of available messages on the mailbox to be read

Note

0 = no messages available

6.4.3.2 template<typename mailboxType > void Mailbox< mailboxType >::begin (uint16_t *number* = 16)

Create the mailbox.

Parameters

number	number of messages of the mailbox, default = 16

6.4.3.3 template<typename mailboxType > bool Mailbox< mailboxType >::post (mailboxType & *message,* uint16_t *timeout* = BIOS_WAIT_FOREVER)

Post a message to the mailbox.

Parameters

message	message to be posted on the mailbox
timeout	default = BIOS_WAIT_FOREVER, BIOS_NO_WAIT

Returns

true if message posted, false otherwise

Note

When using BIOS_NO_WAIT, message isn't posted if mailbox is full. Check returned bool for result.

Warning

Message must be of type typename.

6.4.3.4 template<typename mailboxType > void Mailbox< mailboxType >::waitFor (mailboxType & *message*)

Wait for a message from the mailbox.

Parameters

message	message read from mailbox when available

Warning

Message must be of type typename.

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Mailbox.h

6.5 Semaphore Class Reference

RTOS Semaphore as object.

`#include <Semaphore.h>`

Public Member Functions

- Semaphore ()
  Define the semaphore.
- void begin (uint8_t count=1)
  Create the semaphore.
- void post ()
  Post a semaphore.
- uint16_t available ()
  Available count.
- void waitFor ()
  Wait for the semaphore.

6.5.1 Detailed Description

RTOS Semaphore as object.

The RTOS Semaphore is encapsulated as a C++ object for easier use

6.5.2 Member Function Documentation

6.5.2.1 uint16_t Semaphore::available ( )

Available count.

Returns

number of available count

6.5.2.2 void Semaphore::begin (uint8_t *count* = `1`)

Create the semaphore.

Parameters

count	usually number of ressources to synchronise, default = 1

Note

For serial port, take 1.

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Semaphore.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Semaphore.cpp

6.6 SWI Class Reference

RTOS SWI as object.

`#include <SWI.h>`

Public Member Functions

- SWI ()
  Define the SWI.
- void begin (void(∗functionSWI)(void))
  Create the SWI.
- void post ()
  Post a SWI.

6.6.1 Detailed Description

RTOS SWI as object.

The RTOS SWI is encapsulated as a C++ object for easier use

Note

For w SWI with trigger, see the SWItrigger library.

Deprecated SWI is no longer inlcuded in RTOS for Energia 0101E0017

6.6.2 Member Function Documentation

6.6.2.1 void SWI::begin (void(∗)(void) *functionSWI*)

Create the SWI.

Parameters

functionSWI	function to be called

Note

For serial port, take 1.
The function must be void functionSWI()

```
void functionSWI()
       {
           digitalWrite(RED_LED, HIGH);
       }
```

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/SWI.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/SWI.cpp

6.7 SWItrigger Class Reference

RTOS SWI with trigger as object.

`#include <SWItrigger.h>`

Public Member Functions

- SWItrigger ()
  Define the SWItrigger.
- void begin (uint8_t trigger, void(∗functionSWItrigger)())
  Create the SWItrigger.
- void post ()
  Post a SWItrigger.
- void inc ()
  Increment count and post.
- void dec ()
  Decrement count and post if count = 0.

6.7.1 Detailed Description

RTOS SWI with trigger as object.

The RTOS SWI with trigger is encapsulated as a C++ object for easier use

Deprecated SWI and thus SWItrigger are no longer inlcuded in RTOS for Energia 0101E0017

6.7.2 Member Function Documentation

6.7.2.1 void SWItrigger::begin (uint8_t *trigger,* void(∗)() *functionSWItrigger*)

Create the SWItrigger.

Parameters

functionSWItrigger	function to be called

Note

For serial port, take 1.
The function must be void functionSWItrigger()

```
void functionSWItrigger()
       {
           digitalWrite(RED_LED, HIGH);
       }
```

Bug functionSWItrigger(uint8_t) with uint8_t = Swi_getTrigger() doesn't work.

Todo The idea is to have the same function as for Wire library with Wire.onReceive(void (∗ReceiveEvent)(int)) and void ReceiveEvent(int howMany)

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/SWItrigger.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/SWItrigger.cpp

6.8 Task Class Reference

RTOS Task as object.

`#include <Task.h>`

Public Member Functions

- Task ()
  Define the Task.
- void begin (void(∗functionTask)(void), int8_t priority)
  Create the Task.

6.8.1 Detailed Description

RTOS Task as object.

The RTOS Task is encapsulated as a C++ object for easier use

Note

functionTask() runs once only, unless a while() { ... } is included
This implementation of Task differs from Energia MT.
Each sketch (.ino file) includes setup() and loop() for Energia MT.

6.8.2 Member Function Documentation

6.8.2.1 void Task::begin (void(∗)(void) *functionTask,* int8_t *priority*)

Create the Task.

Parameters

functionTask	function to be called
priority	0 .. Task_numPriorities - 1

Note

The function must be void functionTask()

```
void functionTask()
       {
           digitalWrite(RED_LED, HIGH);
       }
```

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Task.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Task.cpp

6.9 Timer Class Reference

RTOS Timer as an object.

```
#include <Timer.h>
```

Public Member Functions

- Timer ()
  Define the timer.
- void begin (void(∗timerFunction)(void), uint32_t timerPeriod_unit, uint32_t unit=1000)
  Create the timer.
- void start ()
  Start the timer.
- void stop ()
  Stop the timer.

6.9.1 Detailed Description

RTOS Timer as an object.

Warning

for MSP432

The RTOS Timer is encapsulated as a C++ object for easier use

Note

Only one single timer available on the MSP432.

6.9.2 Constructor & Destructor Documentation

6.9.2.1 Timer::Timer ( )

Define the timer.

Warning

Only one single timer available on the MSP432.

6.9.3 Member Function Documentation

6.9.3.1 void Timer::begin (void(*)(void) *timerFunction*, uint32_t *timerPeriod_unit*, uint32_t *unit* = 1000)

Create the timer.

Parameters

timerFunction	function to be called
timerPeriod_unit	period in unit
unit	in us, us = 1, ms = 1000, s = 1000000

Note

The function must be void timerPeriod_ms()

```
void timerPeriod_ms()
       {
           digitalWrite(RED_LED, HIGH);
       }
```

Bug Some fucntions like Serial.print(); don't work :(

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

- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Timer.h
- /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Timer.cpp

Chapter 7

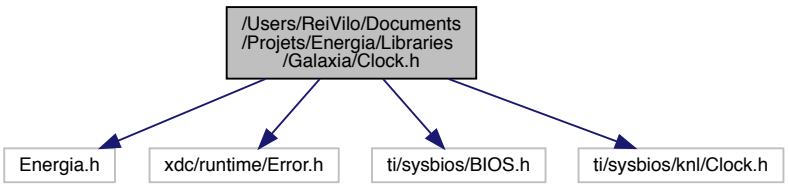
File Documentation

7.1 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Clock.h File Reference

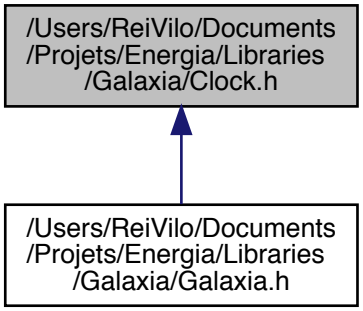
RTOS Clock, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/BIOS.h>
#include <ti/sysbios/knl/Clock.h>
```

Include dependency graph for Clock.h:



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



Classes

- class Clock

  RTOS Clock as an object.

7.1.1 Detailed Description

RTOS Clock, part of the Galaxia Library Suite.

RTOS Clock as C++ object for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Rei Vilo, Jun 17, 2015 09:59

Version

101

Copyright

See also

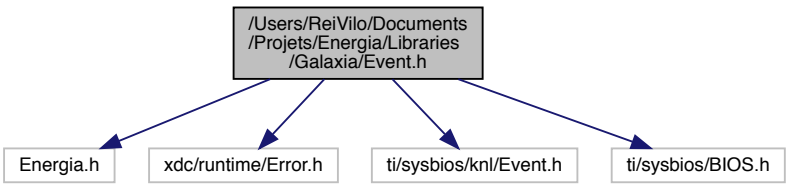
ReadMe.txt for references

7.2 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Event.h File Reference

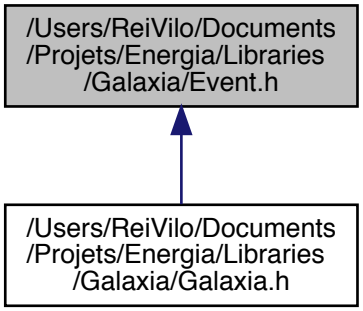
RTOS Event, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/knl/Event.h>
#include <ti/sysbios/BIOS.h>
```

Include dependency graph for Event.h:



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



Classes

- class Event

  RTOS Event as object.

7.2.1 Detailed Description

RTOS Event, part of the Galaxia Library Suite.

RTOS Event as C++ object for Energia MT

Project Energia MT 0101E0016

Author

Energia, base
Rei Vilo, enhancements

Date

Jun 22, 2015 15:53

Version

105

See also

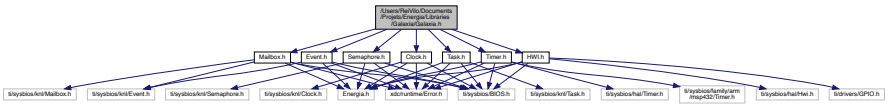
SYS/BIOS (TI-RTOS Kernel) v6.41 User's Guide (spruex3o) http://www.ti.com/lit/pdf/spruex3

7.3 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Galaxia.h File Reference

RTOS Clock, part of the Galaxia Library Suite.

```
#include "Task.h"
#include "HWI.h"
#include "Event.h"
#include "Mailbox.h"
#include "Semaphore.h"
#include "Clock.h"
#include "Timer.h"
```

Include dependency graph for Galaxia.h:



7.3.1 Detailed Description

RTOS Clock, part of the Galaxia Library Suite.

Galaxia library Simple header for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Rei Vilo, Jun 17, 2015 09:59

Version

101

Copyright

See also

ReadMe.txt for references

The Galaxia Library Suite includes

- Thread elements
  - HWI
  - SWI (1)
  - SWI with trigger (SWItrigger) (1)
  - Task
- Synchronisation elements
  - Clock
  - Event
  - Mailbox (2)
  - Semaphore
  - Timer (3)

(1) SWI has been removed from RTOS for Energia MT 0101E0017
(2) Mailbox supersedes Queue
(3) Clock recommended over Timer

Examples include

- Galaxia_Clock
- Galaxia_Event
- Galaxia_HWI
- Galaxia_Mailbox
- Galaxia_Semaphore
- Galaxia_SWI
- Galaxia_SWItrigger
- Galaxia_Task
- Galaxia_Timer
- multiBlink_with_Clock

See also

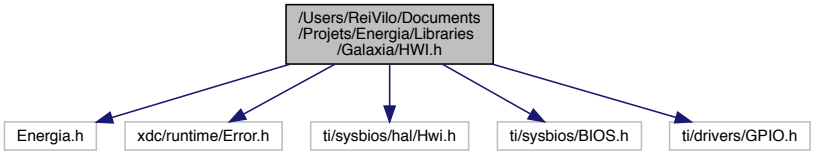
http://embeddedcomputing.weebly.com/exploring-rtos.html

7.4 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/HWI.h File Reference

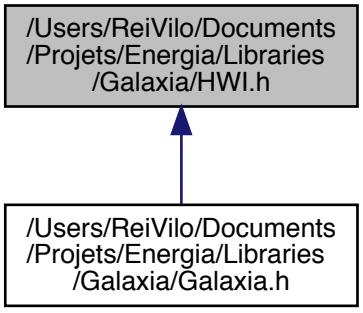
RTOS HWI, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/hal/Hwi.h>
#include <ti/sysbios/BIOS.h>
#include <ti/drivers/GPIO.h>
```

Include dependency graph for HWI.h:



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



Classes

- class HWI

  RTOS HWI as object.

7.4.1 Detailed Description

RTOS HWI, part of the Galaxia Library Suite.

RTOS HWI as C++ object for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Jun 08, 2015 09:53

Version

104

Copyright

See also

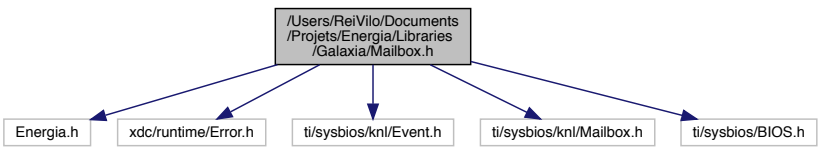
ReadMe.txt for references and example
SYS/BIOS (TI-RTOS Kernel) v6.41 User's Guide (spruex3o) http://www.ti.com/lit/pdf/spruex3

7.5 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Mailbox.h File Reference

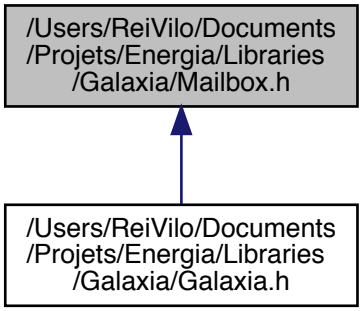
RTOS Mailbox, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/knl/Event.h>
#include <ti/sysbios/knl/Mailbox.h>
#include <ti/sysbios/BIOS.h>
```

Include dependency graph for Mailbox.h:



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



Classes

- class Mailbox< mailboxType >

  RTOS Mailbox as an object.

7.5.1 Detailed Description

RTOS Mailbox, part of the Galaxia Library Suite.

RTOS Mailbox as C++ object for Energia MT

Project Energia MT 0101E0016

Author

Energia, base
Rei Vilo, enhancements

Date

Jun 14, 2015 09:53

Version

104

See also

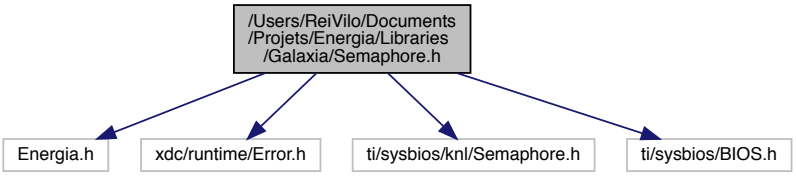
SYS/BIOS (TI-RTOS Kernel) v6.41 User's Guide (spruex3o) http://www.ti.com/lit/pdf/spruex3

7.6 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Semaphore.h File Reference

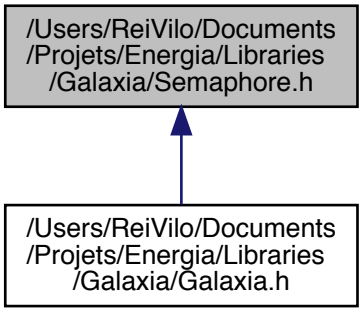
RTOS Semaphore, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/knl/Semaphore.h>
#include <ti/sysbios/BIOS.h>
```

Include dependency graph for Semaphore.h:



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



Classes

- class Semaphore

  RTOS Semaphore as object.

7.6.1 Detailed Description

RTOS Semaphore, part of the Galaxia Library Suite.

RTOS Semaphore as C++ object for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Jun 08, 2015 09:53

Version

104

Copyright

See also

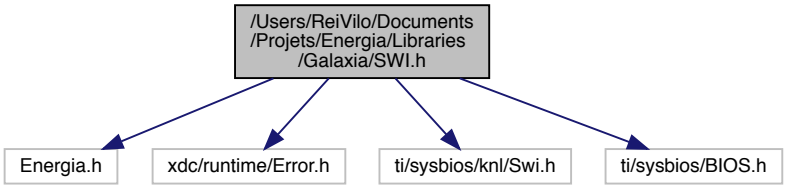
ReadMe.txt for references and example
SYS/BIOS (TI-RTOS Kernel) v6.41 User's Guide (spruex3o) http://www.ti.com/lit/pdf/spruex3

7.7 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/SWI.h File Reference

RTOS SWI, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/knl/Swi.h>
#include <ti/sysbios/BIOS.h>
```

Include dependency graph for SWI.h:



Classes

- class SWI

  RTOS SWI as object.

7.7.1 Detailed Description

RTOS SWI, part of the Galaxia Library Suite.

RTOS SWI as C++ object for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Jun 08, 2015 09:53

Version

104

Copyright

See also

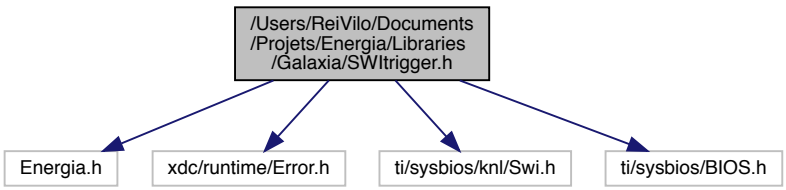
ReadMe.txt for references and example
SYS/BIOS (TI-RTOS Kernel) v6.41 User's Guide (spruex3o) http://www.ti.com/lit/pdf/spruex3

7.8 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/SWItrigger.h File Reference

RTOS SWI with trigger (SWItrigger), part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/knl/Swi.h>
#include <ti/sysbios/BIOS.h>
```

Include dependency graph for SWItrigger.h:



Classes

- class SWItrigger

  RTOS SWI with trigger as object.

7.8.1 Detailed Description

RTOS SWI with trigger (SWItrigger), part of the Galaxia Library Suite.

RTOS SWI with trigger as C++ object for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Jun 08, 2015 09:53

Version

104

Copyright

See also

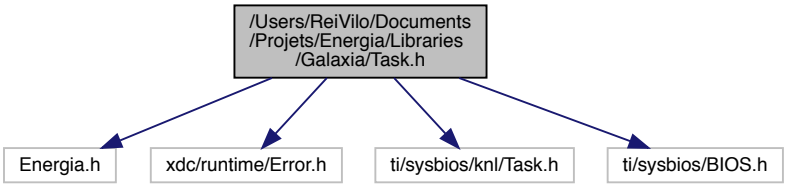
ReadMe.txt for references and example
SYS/BIOS (TI-RTOS Kernel) v6.41 User's Guide (spruex3o) http://www.ti.com/lit/pdf/spruex3

7.9 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Task.h File Reference

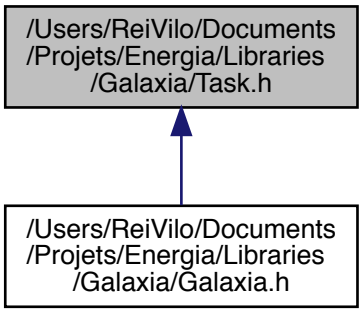
RTOS Task, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/knl/Task.h>
#include <ti/sysbios/BIOS.h>
```

Include dependency graph for Task.h:



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



Classes

- class Task

  RTOS Task as object.

7.9.1 Detailed Description

RTOS Task, part of the Galaxia Library Suite.

RTOS Task as C++ object for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Jun 23, 2015 09:53

Version

101

Copyright

See also

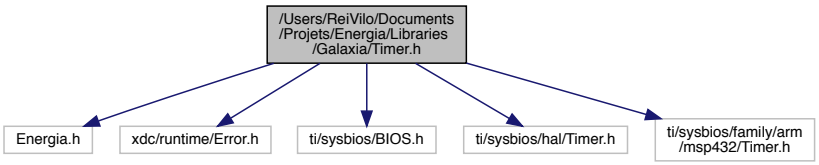
ReadMe.txt for references and example
SYS/BIOS (TI-RTOS Kernel) v6.41 User's Guide (spruex3o) http://www.ti.com/lit/pdf/spruex3

7.10 /Users/ReiVilo/Documents/Projets/Energia/Libraries/Galaxia/Timer.h File Reference

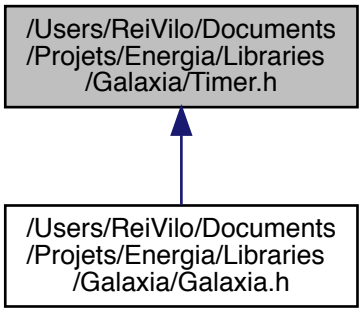
RTOS Timer, part of the Galaxia Library Suite.

```
#include <Energia.h>
#include <xdc/runtime/Error.h>
#include <ti/sysbios/BIOS.h>
#include <ti/sysbios/hal/Timer.h>
#include <ti/sysbios/family/arm/msp432/Timer.h>
```

Include dependency graph for Timer.h:



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



Classes

- class Timer

  RTOS Timer as an object.

7.10.1 Detailed Description

RTOS Timer, part of the Galaxia Library Suite.

RTOS Timer as C++ object for Energia MT

Project Galaxia library for Energia MT
Developed with embedXcode+

Author

Rei Vilo
http://embeddedcomputing.weebly.com

Date

Rei Vilo, Jun 17, 2015 09:29

Version

102

Copyright

See also

ReadMe.txt for references

Index