Dauntless Concepts Underwater LED Project

1.0

Generated by Doxygen 1.8.11

Contents

1 Class Index **1**

1.1 Class List 1

2 File Index **3**

2.1 File List 3

3 Class Documentation **5**

3.1 FlashMemoryData Struct Reference 5

3.1.1 Member Data Documentation 6

3.1.1.1 blueLEDValue 6

3.1.1.2 dataRefreshTimeMet 6

3.1.1.3 debugDataRefreshRateMs 6

3.1.1.4 driverName 6

3.1.1.5 driverSerialNumber 6

3.1.1.6 greenLEDValue 6

3.1.1.7 mode 6

3.1.1.8 modeTiming1 6

3.1.1.9 modeTiming2 6

3.1.1.10 modeTiming3 6

3.1.1.11 partNumber 6

3.1.1.12 redLEDValue 6

3.1.1.13 rfTelemetryEnabled 6

3.1.1.14 serialDebuggingLevel 6

3.1.1.15 serialTelemetryEnabled 6

3.1.1.16 temperatureProtectionEnabled 6
3.1.1.17 testDataOutputEnabled 6
3.1.1.18 timeOfUseMin 6
3.1.1.19 voltageProtectionEnabled 6
3.2 SCStatus Struct Reference 7
3.2.1 Member Data Documentation 7
3.2.1.1 LEDIntensity 7
3.2.1.2 SCdriverFaultFlag 7
3.3 sensorFeedback Struct Reference 7
3.3.1 Detailed Description 7
3.3.2 Member Data Documentation 8
3.3.2.1 inputHitCriticalVoltage 8
3.3.2.2 inputHitWarningVoltage 8
3.3.2.3 inputVoltageinV 8
3.3.2.4 LEDHitCriticalTemp 8
3.3.2.5 LEDHitWarningTemp 8
3.3.2.6 LEDTempinC 8
3.3.2.7 rFDuinoOverTemp 8
3.3.2.8 rFDuinoTempinC 8
3.4 TimerClass Class Reference 8
3.4.1 Constructor & Destructor Documentation 9
3.4.1.1 TimerClass() 9
3.4.2 Member Function Documentation 9
3.4.2.1 getElapsedTimeMs(void) 9
3.4.2.2 resetTimer(void) 9
3.4.2.3 startTimer(void) 9
3.4.2.4 stopTimer(void) 10
3.4.3 Member Data Documentation 10
3.4.3.1 timeOutMs 10

4 File Documentation 11

4.1 FaultHandler.cpp File Reference 11

4.1.1 Function Documentation 12

4.1.1.1 inputVoltageErrorBehavior(void) 12

4.1.1.2 ledOverTempBehavior(void) 12

4.1.1.3 processRGBFault(void) 13

4.1.1.4 processSCFault(void) 13

4.1.1.5 reactToSensors(void) 13

4.1.2 Variable Documentation 14

4.1.2.1 TempErrorTimer 14

4.1.2.2 VoltageTimer 14

4.2 FaultHandler.h File Reference 14

4.2.1 Macro Definition Documentation 15

4.2.1.1 LED_OVERTEMP_FIRST_REDUCTION_PERCENTAGE 15

4.2.2 Function Documentation 15

4.2.2.1 processRGBFault(void) 15

4.2.2.2 processSCFault(void) 15

4.3 Globals.cpp File Reference 16

4.3.1 Function Documentation 16

4.3.1.1 loadFlashMemoryValues(void) 16

4.3.2 Variable Documentation 17

4.3.2.1 elapsedTimeMs 17

4.3.2.2 FlashMemoryData 17

4.3.2.3 gbl_SCStatus 17

4.3.2.4 gbl_systemBootFlag 17

4.3.2.5 gbl_systemState 17

4.3.2.6 gbl_systemTimerinMs 17

4.3.2.7 startTimeMs 17

4.3.2.8 stopTimeMs 17

4.4 Globals.h File Reference 17

4.4.1 Function Documentation 18
4.4.1.1 loadFlashMemoryValues(void) 18
4.4.1.2 writeAllSettingsToFlash(void) 18
4.4.1.3 writeNewTimeofUseToFlash(void) 18
4.4.2 Variable Documentation 18
4.4.2.1 FlashMemoryData 18
4.4.2.2 gbl_SCStatus 18
4.4.2.3 gbl_systemBootFlag 19
4.4.2.4 gbl_systemState 19
4.4.2.5 gbl_systemTimerinMs 19
4.5 HardwareConfiguration.h File Reference 19
4.5.1 Macro Definition Documentation 20
4.5.1.1 DEBUG_REFRESH_RATE 20
4.5.1.2 RF_TELEMETRY_ENABLED 20
4.5.1.3 RGB_DRIVER 20
4.5.1.4 row1 20
4.5.1.5 row2 20
4.5.1.6 row3 20
4.5.1.7 row4 20
4.5.1.8 row5 20
4.5.1.9 row6 20
4.5.1.10 SERIAL_TELEMETRY_ENABLED 20
4.5.1.11 SOFTWARE_VERSION 20
4.6 LEDControlManager.cpp File Reference 20
4.6.1 Function Documentation 21
4.6.1.1 setRedPWM(char percent) 21
4.6.1.2 setSCIntensity(void) 22
4.7 LEDControlManager.h File Reference 23
4.7.1 Function Documentation 23
4.7.1.1 onConnectFlash(void) 23

4.7.1.2 onDisconnectFlash(void) 24
4.7.1.3 setBluePWM(char percent) 24
4.7.1.4 setGreenPWM(char percent) 24
4.7.1.5 setRedPWM(char percent) 25
4.7.1.6 setRGBColors(void) 25
4.7.1.7 setSCIntensity(void) 26
4.8 Memory.c File Reference 26
4.9 Memory.h File Reference 26
4.9.1 Macro Definition Documentation 27
4.9.1.1 MAX_NUMBER_OF_FLASH_WRITES 27
4.9.1.2 TIME_OF_USE_PERIOD_IN_MS 27
4.10 PinDefinitions.h File Reference 27
4.10.1 Macro Definition Documentation 28
4.10.1.1 BLUE_LED_PIN 28
4.10.1.2 GREEN_LED_PIN 28
4.10.1.3 INPUT_VOLTAGE_PIN 28
4.10.1.4 LED_BOARD_TEMP_PIN 28
4.10.1.5 RED_LED_PIN 28
4.11 SensorManager.cpp File Reference 28
4.11.1 Function Documentation 29
4.11.1.1 checkSensors(void) 29
4.11.1.2 getInputVoltageinV(void) 30
4.11.1.3 getLEDTempinC(void) 30
4.11.2 Variable Documentation 30
4.11.2.1 sensorFeedback 30
4.11.2.2 VoltageTime 30
4.12 SensorManager.h File Reference 30
4.12.1 Macro Definition Documentation 31
4.12.1.1 ADC_RESOLUTION 31
4.12.1.2 ADC_VOLTAGE_REF 31

4.12.1.3 INPUT_VOLTAGE_CRITICAL_V 31
4.12.1.4 INPUT_VOLTAGE_SAG_TIME_MS 31
4.12.1.5 INPUT_VOLTAGE_WARNING_V 31
4.12.1.6 LED_CRITICAL_TEMP_C 31
4.12.1.7 LED_WARNING_TEMP_C 31
4.12.1.8 RFDUINO_CRITICAL_TEMP_C 31
4.12.2 Function Documentation 31
4.12.2.1 checkSensors(void) 32
4.12.2.2 getInputVoltageinmV(void) 32
4.12.2.3 getLEDTempinC(void) 33
4.12.2.4 getRFDuinoTempinC(void) 33
4.12.3 Variable Documentation 33
4.12.3.1 sensorFeedback 33
4.13 SerialDebug.cpp File Reference 34
4.13.1 Function Documentation 34
4.13.1.1 checkDebugSendTime(void) 34
4.13.1.2 checkSerial(void) 34
4.13.2 Variable Documentation 34
4.13.2.1 SerialDebugLevel 35
4.14 SerialDebug.h File Reference 35
4.14.1 Macro Definition Documentation 36
4.14.1.1 DEBUG 36
4.14.1.2 DEBUG_LF 36
4.14.1.3 TERMINAL 36
4.14.1.4 TERMINAL_LF 36
4.14.1.5 TIMED_DEBUG 36
4.14.2 Enumeration Type Documentation 36
4.14.2.1 DebugLevels 36
4.14.3 Function Documentation 37
4.14.3.1 checkDebugSendTime(void) 37

4.14.3.2 checkSerial(void) 37
4.14.4 Variable Documentation 37
4.14.4.1 SerialDebugLevel 37
4.15 States.cpp File Reference 37
4.15.1 Function Documentation 38
4.15.1.1 checkBootComplete(void) 38
4.15.1.2 checkState(void) 38
4.15.1.3 configurePins(void) 39
4.15.1.4 initializeSystem(void) 39
4.15.1.5 printStartUpData(void) 40
4.15.1.6 SCFaultState(void) 40
4.15.1.7 SCNominalState(void) 41
4.15.1.8 setState(void) 42
4.15.2 Variable Documentation 42
4.15.2.1 newStatePrintFlag 42
4.15.2.2 StartupTimer 42
4.16 States.h File Reference 42
4.16.1 Enumeration Type Documentation 43
4.16.1.1 States 43
4.16.2 Function Documentation 43
4.16.2.1 checkBootComplete(void) 43
4.16.2.2 configurePins(void) 44
4.16.2.3 initializeSystem(void) 44
4.16.2.4 onRadioConnect(void) 44
4.16.2.5 onRadioDisconnect(void) 44
4.16.2.6 printStartUpData(void) 44
4.16.2.7 setState(void) 45

Index **47**

Chapter 1

Class Index

1.1 Class List

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

FlashMemoryData 5
SCStatus 7
sensorFeedback
SensorFeedback struct Description 7
TimerClass 8

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

FaultHandler.cpp 11
FaultHandler.h 14
Globals.cpp 16
Globals.h 17
HardwareConfiguration.h 19
LEDControlManager.cpp 20
LEDControlManager.h 23
Memory.c 26
Memory.h 26
PinDefinitions.h 27
SensorManager.cpp 28
SensorManager.h 30
SerialDebug.cpp 34
SerialDebug.h 35
States.cpp 37
States.h 42

Chapter 3

Class Documentation

3.1 FlashMemoryData Struct Reference

```
#include <Globals.h>
```

Public Attributes

- char partNumber [10]
- char driverSerialNumber [8]
- char driverName [11]
- char redLEDValue
- char greenLEDValue
- char blueLEDValue
- unsigned char mode
- unsigned int modeTiming1
- unsigned int modeTiming2
- unsigned int modeTiming3
- unsigned long int timeOfUseMin
- bool rfTelemetryEnabled
- bool serialTelemetryEnabled
- bool serialDebuggingLevel
- bool testDataOutputEnabled
- bool voltageProtectionEnabled
- bool temperatureProtectionEnabled
- bool dataRefreshTimeMet
- unsigned int debugDataRefreshRateMs

3.1.1 Member Data Documentation

3.1.1.1 char FlashMemoryData::blueLEDValue

3.1.1.2 bool FlashMemoryData::dataRefreshTimeMet

3.1.1.3 unsigned int FlashMemoryData::debugDataRefreshRateMs

3.1.1.4 char FlashMemoryData::driverName[11]

3.1.1.5 char FlashMemoryData::driverSerialNumber[8]

3.1.1.6 char FlashMemoryData::greenLEDValue

3.1.1.7 unsigned char FlashMemoryData::mode

3.1.1.8 unsigned int FlashMemoryData::modeTiming1

3.1.1.9 unsigned int FlashMemoryData::modeTiming2

3.1.1.10 unsigned int FlashMemoryData::modeTiming3

3.1.1.11 char FlashMemoryData::partNumber[10]

3.1.1.12 char FlashMemoryData::redLEDValue

3.1.1.13 bool FlashMemoryData::rfTelemetryEnabled

3.1.1.14 bool FlashMemoryData::serialDebuggingLevel

3.1.1.15 bool FlashMemoryData::serialTelemetryEnabled

3.1.1.16 bool FlashMemoryData::temperatureProtectionEnabled

3.1.1.17 bool FlashMemoryData::testDataOutputEnabled

3.1.1.18 unsigned long int FlashMemoryData::timeOfUseMin

3.1.1.19 bool FlashMemoryData::voltageProtectionEnabled

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

- Globals.h

3.2 SCStatus Struct Reference

```
#include <Globals.h>
```

Public Attributes

- char LEDIntensity
- bool SCdriverFaultFlag

3.2.1 Member Data Documentation

3.2.1.1 char SCStatus::LEDIntensity

3.2.1.2 bool SCStatus::SCdriverFaultFlag

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

- Globals.h

3.3 sensorFeedback Struct Reference

sensorFeedback struct Description

```
#include <SensorManager.h>
```

Public Attributes

- int rFDuinoTempinC
- bool rFDuinoOverTemp
- int LEDTempinC
- bool LEDHitWarningTemp
- bool LEDHitCriticalTemp
- float inputVoltageinV
- bool inputHitWarningVoltage
- bool inputHitCriticalVoltage

3.3.1 Detailed Description

sensorFeedback struct Description

The sensorFeedback struct holds all of the elements of the light's sensor system. The struct holds the converted values (units) and flags for each of the sensors being monitored.

3.3.2 Member Data Documentation

3.3.2.1 bool sensorFeedback::inputHitCriticalVoltage

3.3.2.2 bool sensorFeedback::inputHitWarningVoltage

3.3.2.3 float sensorFeedback::inputVoltageinV

3.3.2.4 bool sensorFeedback::LEDHitCriticalTemp

3.3.2.5 bool sensorFeedback::LEDHitWarningTemp

3.3.2.6 int sensorFeedback::LEDTempinC

3.3.2.7 bool sensorFeedback::rFDuinoOverTemp

3.3.2.8 int sensorFeedback::rFDuinoTempinC

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

- SensorManager.h

3.4 TimerClass Class Reference

```
#include <Globals.h>
```

Public Member Functions

- TimerClass ()
- void startTimer (void)
- void stopTimer (void)
- void resetTimer (void)
- unsigned long getElapsedTimeMs (void)

Public Attributes

- unsigned long timeOutMs

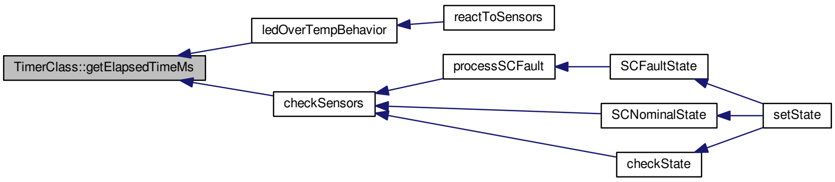
3.4.1 Constructor & Destructor Documentation

3.4.1.1 TimerClass::TimerClass ( )

3.4.2 Member Function Documentation

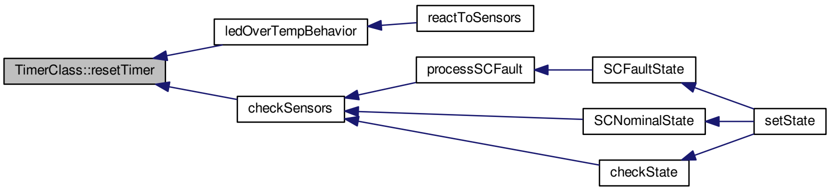
3.4.2.1 unsigned long TimerClass::getElapsedTimeMs (void)

Here is the caller graph for this function:



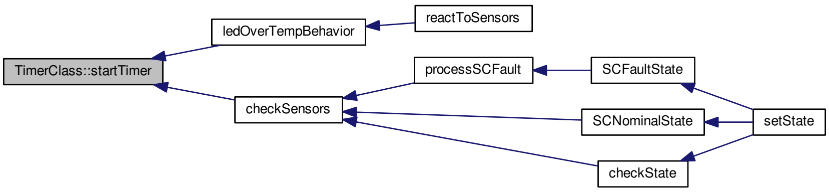
3.4.2.2 void TimerClass::resetTimer (void)

Here is the caller graph for this function:



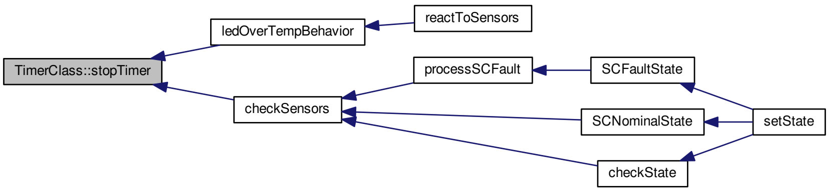
3.4.2.3 void TimerClass::startTimer (void)

Here is the caller graph for this function:



3.4.2.4 void TimerClass::stopTimer (void)

Here is the caller graph for this function:



3.4.3 Member Data Documentation

3.4.3.1 unsigned long TimerClass::timeOutMs

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

- Globals.h
- Globals.cpp

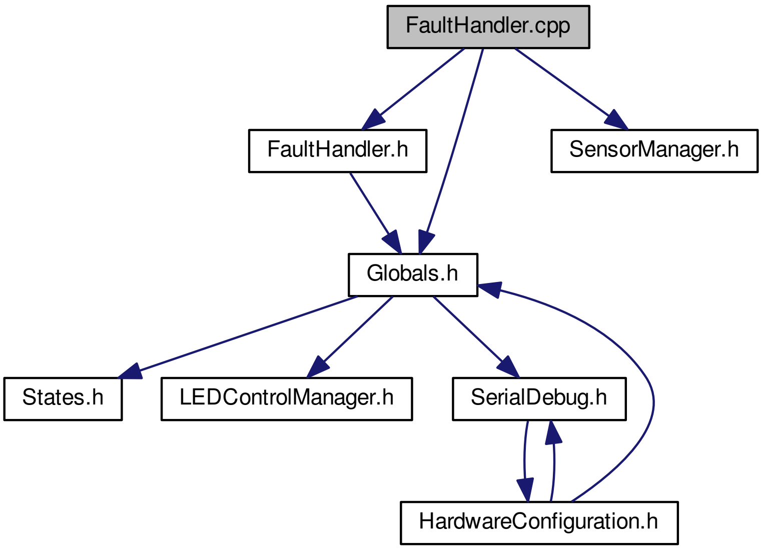
Chapter 4

File Documentation

4.1 FaultHandler.cpp File Reference

```
#include "FaultHandler.h"
#include "Globals.h"
#include "SensorManager.h"
```

Include dependency graph for FaultHandler.cpp:



Functions

- void processSCFault (void)
- void processRGBFault (void)
- void ledOverTempBehavior (void)
- void inputVoltageErrorBehavior (void)
- void reactToSensors (void)

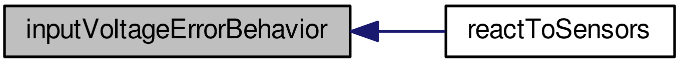
Variables

- class TimerClass VoltageTimer
- class TimerClass TempErrorTimer

4.1.1 Function Documentation

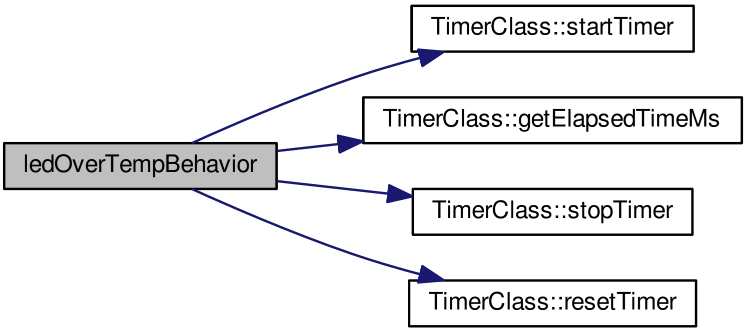
4.1.1.1 void inputVoltageErrorBehavior (void)

Here is the caller graph for this function:

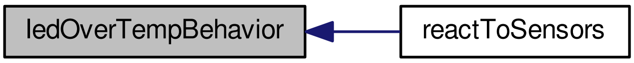


4.1.1.2 void ledOverTempBehavior (void)

Here is the call graph for this function:



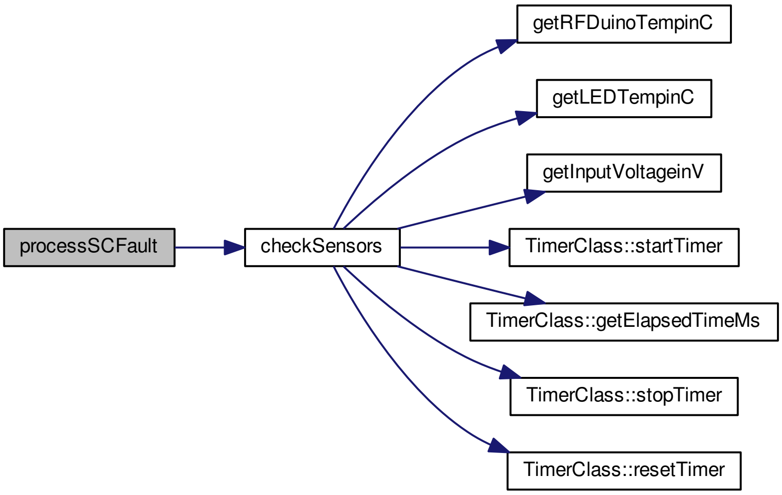
Here is the caller graph for this function:



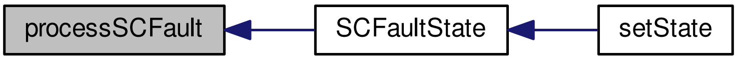
4.1.1.3 void processRGBFault (void)

4.1.1.4 void processSCFault (void)

Here is the call graph for this function:

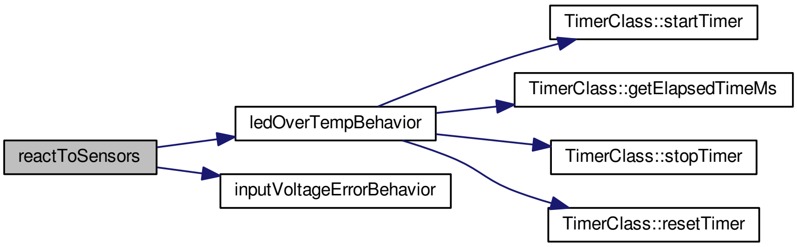


Here is the caller graph for this function:



4.1.1.5 void reactToSensors (void)

Here is the call graph for this function:



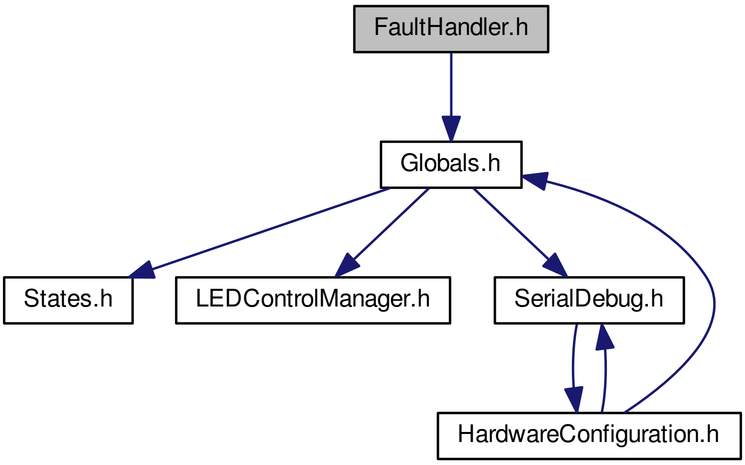
4.1.2 Variable Documentation

4.1.2.1 class **TimerClass** TempErrorTimer

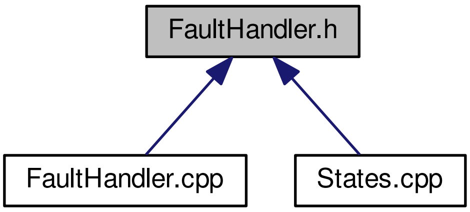
4.1.2.2 class **TimerClass** VoltageTimer

4.2 FaultHandler.h File Reference

`#include "Globals.h"`
Include dependency graph for FaultHandler.h:



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



Macros

- #define LED_OVERTEMP_FIRST_REDUCTION_PERCENTAGE 75

Functions

- void processSCFault (void)
- void processRGBFault (void)

4.2.1 Macro Definition Documentation

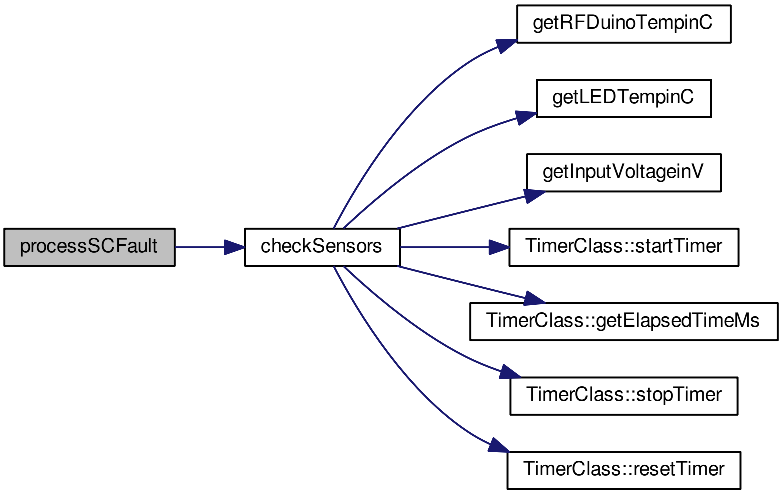
4.2.1.1 #define LED_OVERTEMP_FIRST_REDUCTION_PERCENTAGE 75

4.2.2 Function Documentation

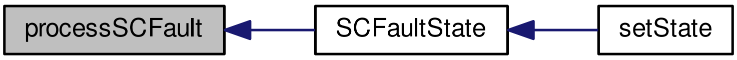
4.2.2.1 void processRGBFault (void)

4.2.2.2 void processSCFault (void)

Here is the call graph for this function:



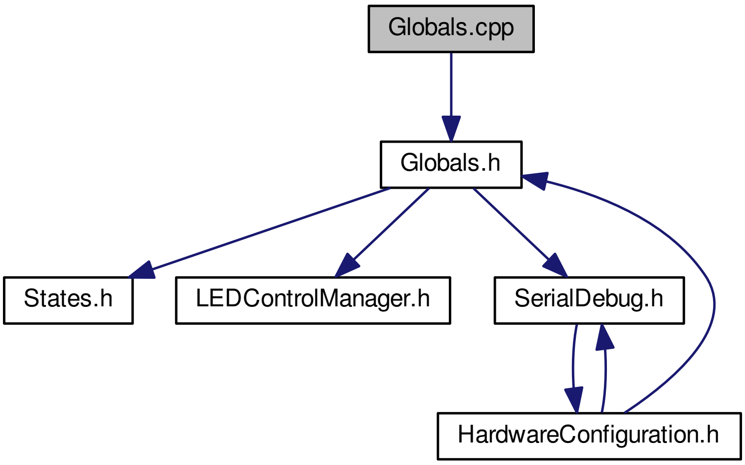
Here is the caller graph for this function:



4.3 Globals.cpp File Reference

```
#include "Globals.h"
```

Include dependency graph for Globals.cpp:



Functions

- void loadFlashMemoryValues (void)

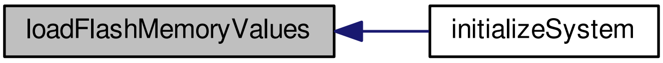
Variables

- unsigned long elapsedTimeMs = 0
- unsigned long startTimeMs = 0
- unsigned long stopTimeMs = 0
- enum States gbl_systemState = Boot
- bool gbl_systemBootFlag = false
- struct SCStatus gbl_SCStatus = {0,false}
- unsigned long gbl_systemTimerinMs = 0
- struct FlashMemoryData FlashMemoryData = {}

4.3.1 Function Documentation

4.3.1.1 void loadFlashMemoryValues (void)

Here is the caller graph for this function:



4.3.2 Variable Documentation

4.3.2.1 unsigned long elapsedTimeMs = 0

4.3.2.2 struct FlashMemoryData FlashMemoryData = {}

4.3.2.3 struct SCStatus gbl_SCStatus = {0,false}

4.3.2.4 bool gbl_systemBootFlag = false

4.3.2.5 enum States gbl_systemState = Boot

4.3.2.6 unsigned long gbl_systemTimerinMs = 0

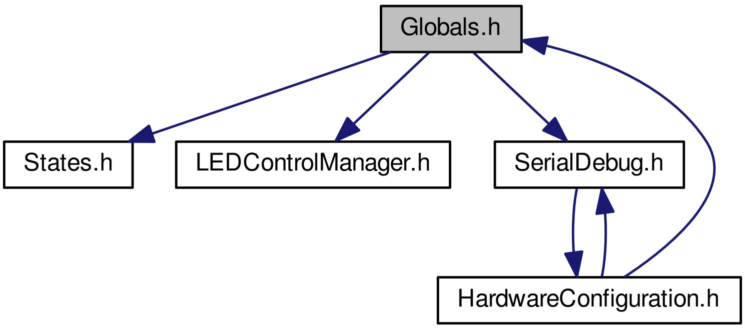
4.3.2.7 unsigned long startTimeMs = 0

4.3.2.8 unsigned long stopTimeMs = 0

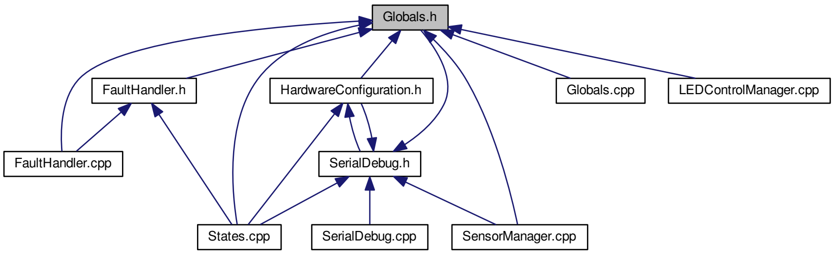
4.4 Globals.h File Reference

```
#include "States.h"
#include "LEDControlManager.h"
#include "SerialDebug.h"
```

Include dependency graph for Globals.h:



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



Classes

- class TimerClass
- struct FlashMemoryData
- struct SCStatus

Functions

- void loadFlashMemoryValues (void)
- void writeNewTimeofUseToFlash (void)
- void writeAllSettingsToFlash (void)

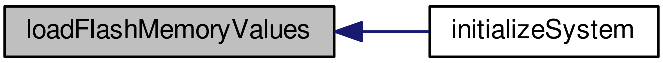
Variables

- bool gbl_systemBootFlag
- struct SCStatus gbl_SCStatus
- enum States gbl_systemState
- struct FlashMemoryData FlashMemoryData
- unsigned long gbl_systemTimerinMs

4.4.1 Function Documentation

4.4.1.1 void loadFlashMemoryValues (void)

Here is the caller graph for this function:



4.4.1.2 void writeAllSettingsToFlash (void)

4.4.1.3 void writeNewTimeofUseToFlash (void)

4.4.2 Variable Documentation

4.4.2.1 struct FlashMemoryData FlashMemoryData

4.4.2.2 struct SCStatus gbl_SCStatus

4.4.2.3 bool gbl_systemBootFlag

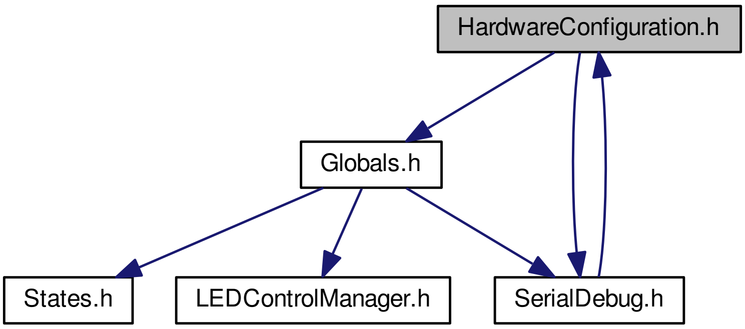
4.4.2.4 enum States gbl_systemState

4.4.2.5 unsigned long gbl_systemTimerinMs

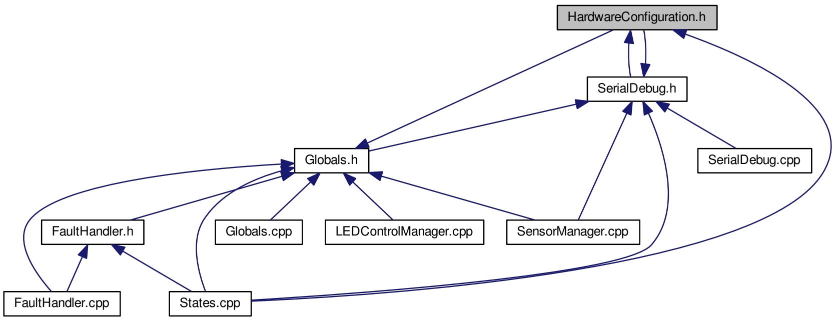
4.5 HardwareConfiguration.h File Reference

```
#include "Globals.h"
#include "SerialDebug.h"
```

Include dependency graph for HardwareConfiguration.h:



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



Macros

- #define SOFTWARE_VERSION "0.0.0"
- #define RGB_DRIVER false
- #define RF_TELEMETRY_ENABLED true
- #define SERIAL_TELEMETRY_ENABLED true

- #define DEBUG_REFRESH_RATE 1000
- #define row1 " ____ __ __ ______ __"
- #define row2 " / __ \\____ ___ ______ / /_/ /__ __________ / ____/___ ____ ________ ____ / /______"
- #define row3 " / / / / __ '/ / / / __ \\/ __/ / _ \\/ ___/ ___/ / / / __ \\/ __ \\/ ___/ _ \\/ __ \\/ __/ ___/"
- #define row4 " / /_/ / /_/ / /_/ / / / / /_/ / __(__ |__) / /___/ /_/ / / / / /__/ __/ /_/ / /_(__)"
- #define row5 "/_____/\\__,_/\\__,_/_/ /_/\__/_/\___/____/____/ \____/\____/_/ /_/\___/\___/ .___/\__/__↩
  __/"
- #define row6 " /_/"

4.5.1 Macro Definition Documentation

4.5.1.1 #define DEBUG_REFRESH_RATE 1000

4.5.1.2 #define RF_TELEMETRY_ENABLED true

4.5.1.3 #define RGB_DRIVER false

4.5.1.4 #define row1 " ____ __ __ ______ __"

4.5.1.5 #define row2 " / __ \\____ ___ ______ / /_/ /__ __________ / ____/___ ____ ________ ____ / /______"

4.5.1.6 #define row3 " / / / / __ '/ / / / __ \\/ __/ / _ \\/ ___/ ___/ / / / __ \\/ __ \\/ ___/ _ \\/ __ \\/ __/ ___/"

4.5.1.7 #define row4 " / /_/ / /_/ / /_/ / / / / /_/ / __(__ |__) / /___/ /_/ / / / / /__/ __/ /_/ / /_(__)"

4.5.1.8 #define row5 "/_____/\\__,_/\\__,_/_/ /_/\__/_/\___/____/____/ \____/\____/_/ /_/\___/\___/ .___/\__/____/"

4.5.1.9 #define row6 " /_/"

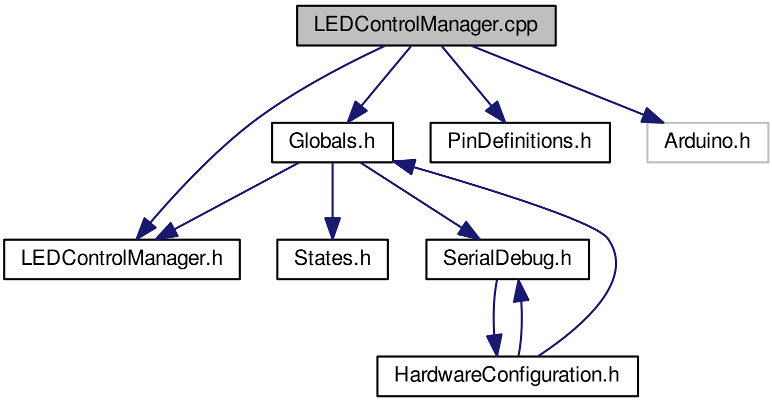
4.5.1.10 #define SERIAL_TELEMETRY_ENABLED true

4.5.1.11 #define SOFTWARE_VERSION "0.0.0"

4.6 LEDControlManager.cpp File Reference

```
#include "LEDControlManager.h"
#include "Globals.h"
#include "PinDefinitions.h"
#include <Arduino.h>
```

Include dependency graph for LEDControlManager.cpp:



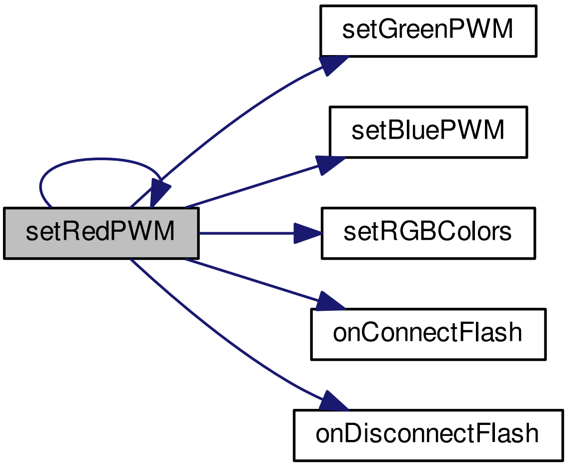
Functions

- void setRedPWM (char percent)
- void setSCIntensity (void)

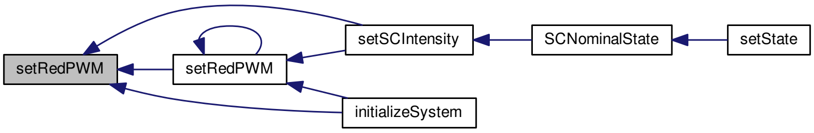
4.6.1 Function Documentation

4.6.1.1 void setRedPWM (char *percent*)

Here is the call graph for this function:

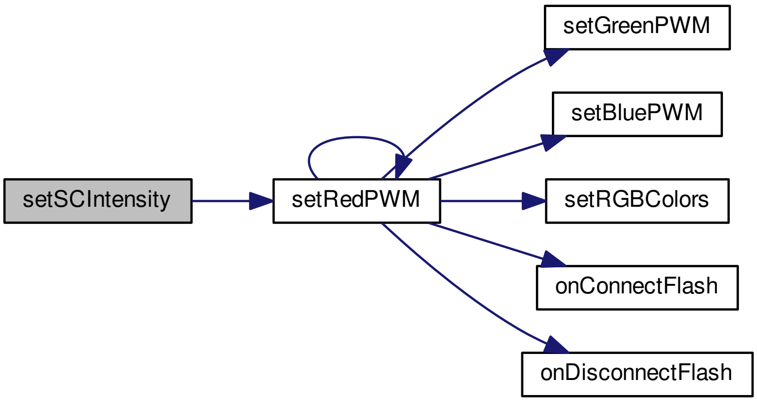


Here is the caller graph for this function:

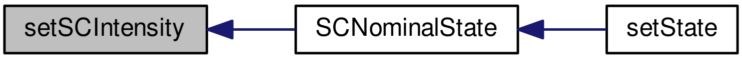


4.6.1.2 void setSCIntensity (void)

Here is the call graph for this function:

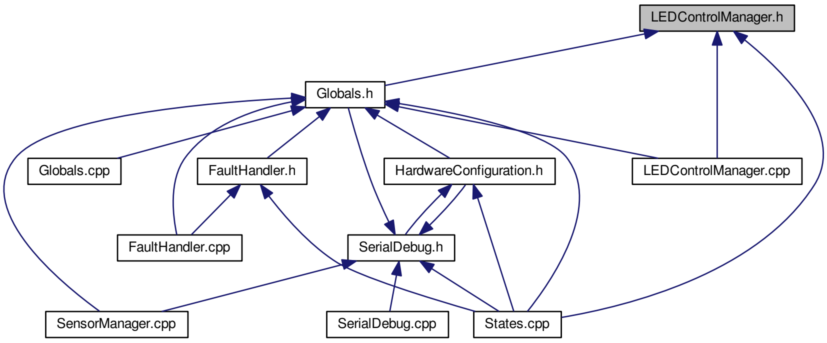


Here is the caller graph for this function:



4.7 LEDControlManager.h File Reference

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



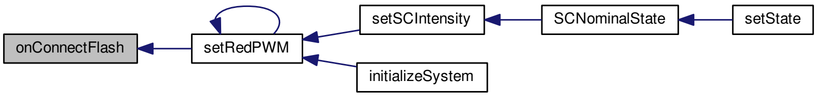
Functions

- void setSCIntensity (void)
- void setRedPWM (char percent)
- void setGreenPWM (char percent)
- void setBluePWM (char percent)
- void setRGBColors (void)
- void onConnectFlash (void)
- void onDisconnectFlash (void)

4.7.1 Function Documentation

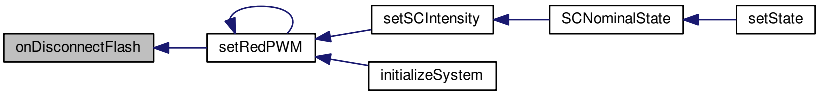
4.7.1.1 void onConnectFlash (void)

Here is the caller graph for this function:



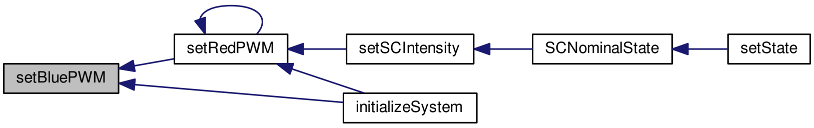
4.7.1.2 void onDisconnectFlash (void)

Here is the caller graph for this function:



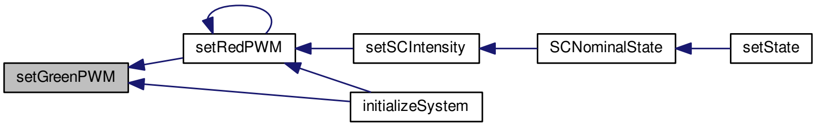
4.7.1.3 void setBluePWM (char *percent*)

Here is the caller graph for this function:



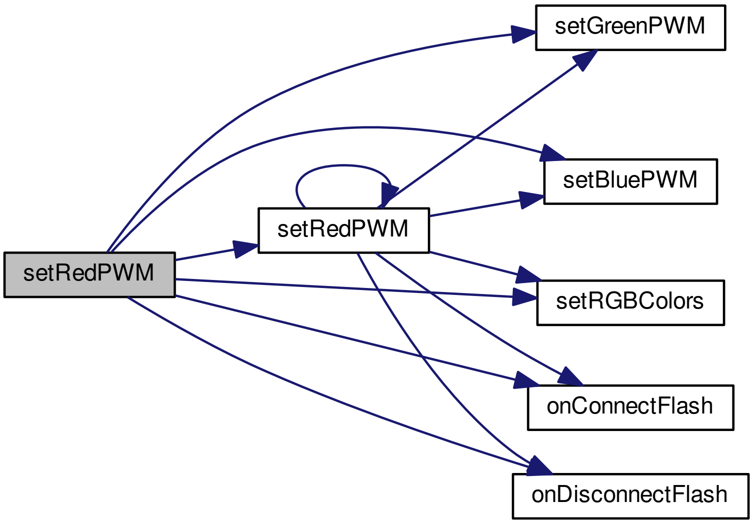
4.7.1.4 void setGreenPWM (char *percent*)

Here is the caller graph for this function:

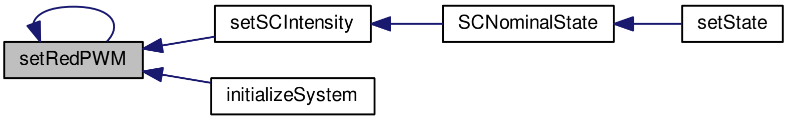


4.7.1.5 void setRedPWM (char *percent*)

Here is the call graph for this function:

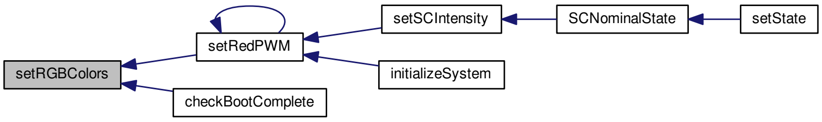


Here is the caller graph for this function:



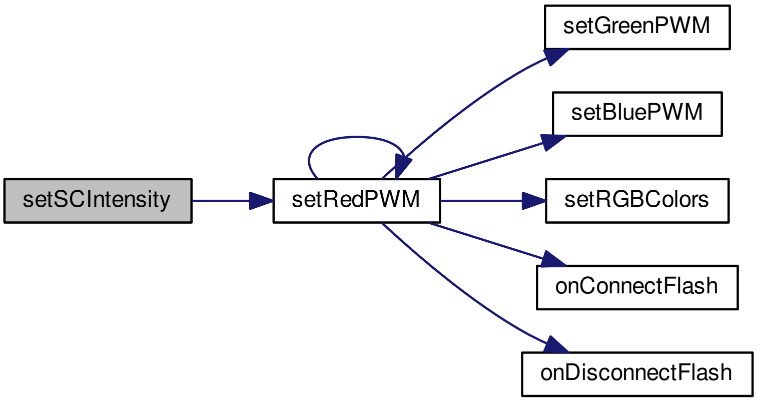
4.7.1.6 void setRGBColors (void)

Here is the caller graph for this function:

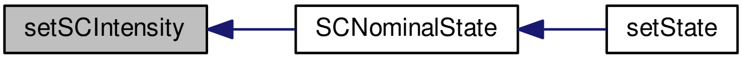


4.7.1.7 void setSCIntensity (void)

Here is the call graph for this function:



Here is the caller graph for this function:

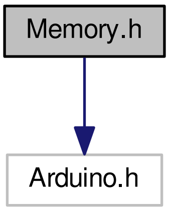


4.8 Memory.c File Reference

4.9 Memory.h File Reference

```
#include <Arduino.h>
```

Include dependency graph for Memory.h:



Macros

- #define MAX_NUMBER_OF_FLASH_WRITES 18000
- #define TIME_OF_USE_PERIOD_IN_MS 600000

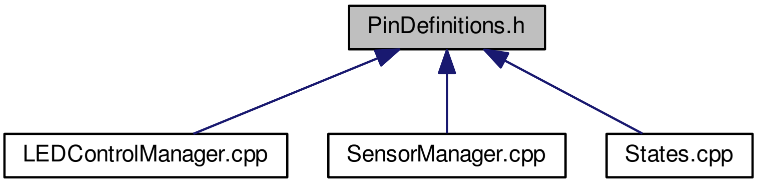
4.9.1 Macro Definition Documentation

4.9.1.1 #define MAX_NUMBER_OF_FLASH_WRITES 18000

4.9.1.2 #define TIME_OF_USE_PERIOD_IN_MS 600000

4.10 PinDefinitions.h File Reference

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



Macros

- #define RED_LED_PIN 4
- #define GREEN_LED_PIN 5
- #define BLUE_LED_PIN 6
- #define LED_BOARD_TEMP_PIN 2
- #define INPUT_VOLTAGE_PIN 1

4.10.1 Macro Definition Documentation

4.10.1.1 #define BLUE_LED_PIN 6

4.10.1.2 #define GREEN_LED_PIN 5

4.10.1.3 #define INPUT_VOLTAGE_PIN 1

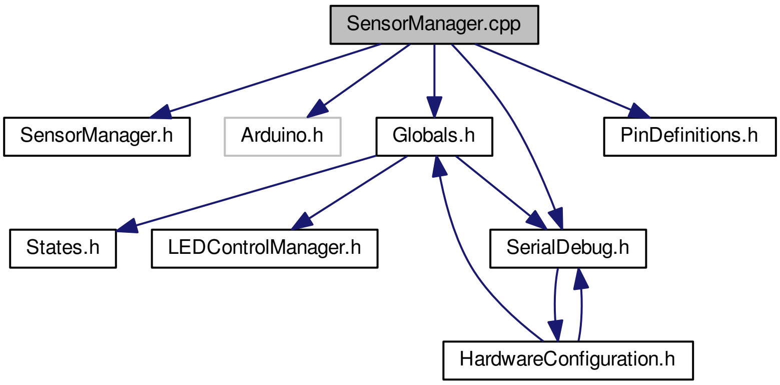
4.10.1.4 #define LED_BOARD_TEMP_PIN 2

4.10.1.5 #define RED_LED_PIN 4

4.11 SensorManager.cpp File Reference

```
#include "SensorManager.h"
#include <Arduino.h>
#include "Globals.h"
#include "PinDefinitions.h"
#include "SerialDebug.h"
```

Include dependency graph for SensorManager.cpp:



Functions

- int getLEDTempinC (void)
- float getInputVoltageinV (void)
- void checkSensors (void)

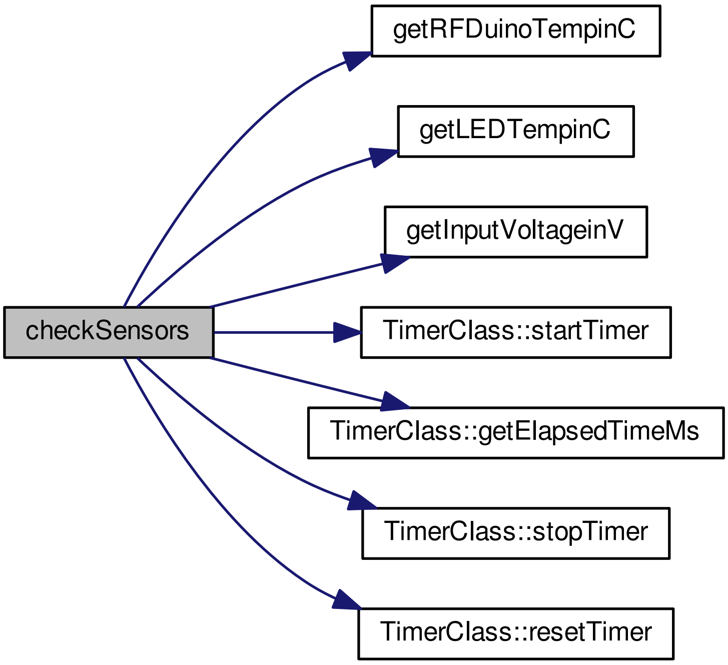
Variables

- class TimerClass VoltageTime
- struct sensorFeedback sensorFeedback

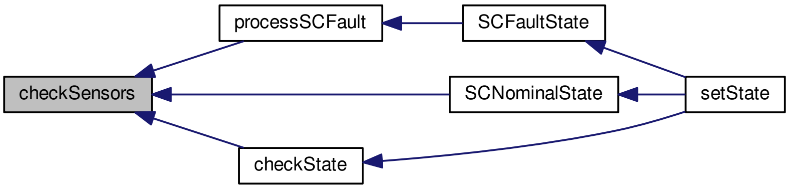
4.11.1 Function Documentation

4.11.1.1 void checkSensors (void)

Here is the call graph for this function:

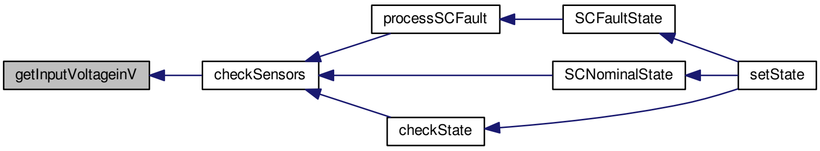


Here is the caller graph for this function:



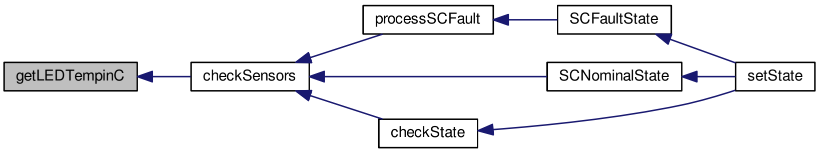
4.11.1.2 float getInputVoltageinV (void)

Here is the caller graph for this function:



4.11.1.3 int getLEDTempinC (void)

Here is the caller graph for this function:



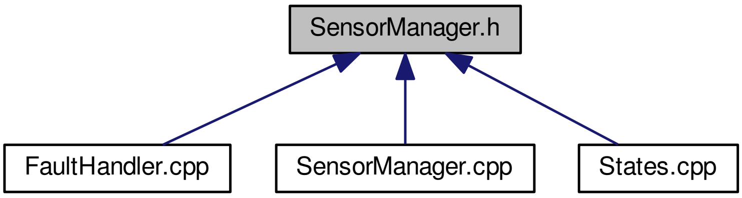
4.11.2 Variable Documentation

4.11.2.1 struct sensorFeedback sensorFeedback

4.11.2.2 class TimerClass VoltageTime

4.12 SensorManager.h File Reference

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



Classes

- struct sensorFeedback

  sensorFeedback struct Description

Macros

- #define ADC_RESOLUTION 1024
- #define ADC_VOLTAGE_REF 3.3
- #define RFDUINO_CRITICAL_TEMP_C 60
- #define LED_WARNING_TEMP_C 60
- #define LED_CRITICAL_TEMP_C 70
- #define INPUT_VOLTAGE_WARNING_V 12.7
- #define INPUT_VOLTAGE_CRITICAL_V 12.1
- #define INPUT_VOLTAGE_SAG_TIME_MS 7000

Functions

- int getRFDuinoTempinC (void)
- int getLEDTempinC (void)
- int getInputVoltageinmV (void)
- void checkSensors (void)

Variables

- sensorFeedback sensorFeedback

4.12.1 Macro Definition Documentation

4.12.1.1 #define ADC_RESOLUTION 1024

4.12.1.2 #define ADC_VOLTAGE_REF 3.3

4.12.1.3 #define INPUT_VOLTAGE_CRITICAL_V 12.1

4.12.1.4 #define INPUT_VOLTAGE_SAG_TIME_MS 7000

4.12.1.5 #define INPUT_VOLTAGE_WARNING_V 12.7

4.12.1.6 #define LED_CRITICAL_TEMP_C 70

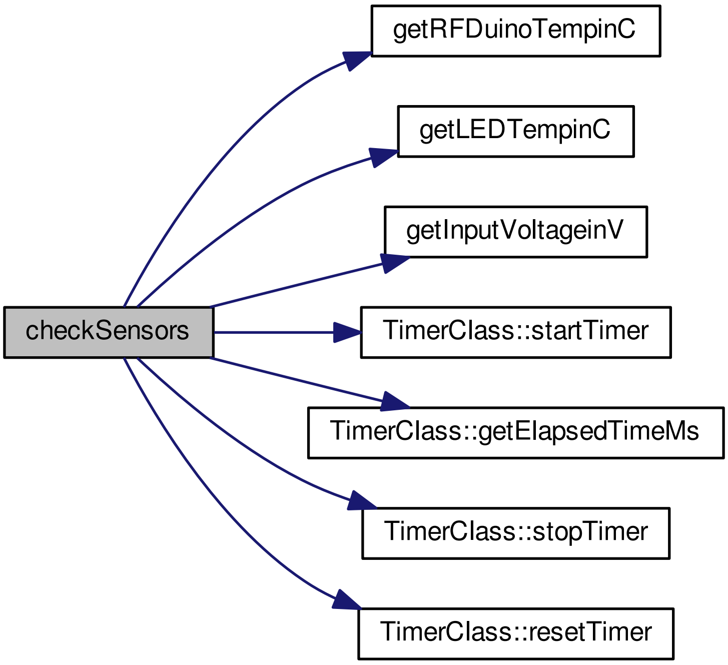
4.12.1.7 #define LED_WARNING_TEMP_C 60

4.12.1.8 #define RFDUINO_CRITICAL_TEMP_C 60

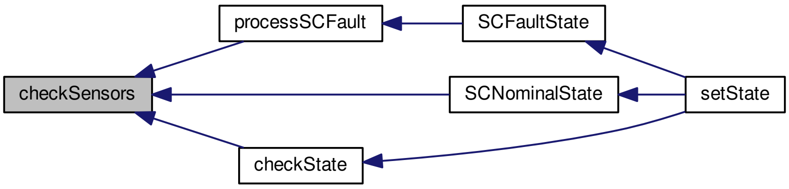
4.12.2 Function Documentation

4.12.2.1 void checkSensors (void)

Here is the call graph for this function:



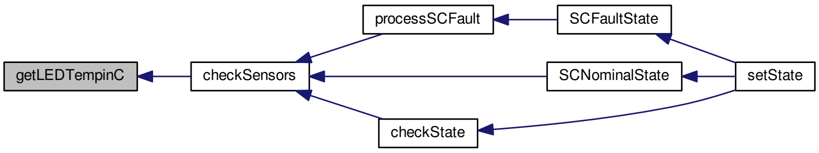
Here is the caller graph for this function:



4.12.2.2 int getInputVoltageinmV (void)

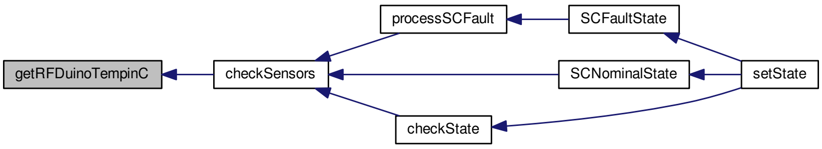
4.12.2.3 int getLEDTempinC (void)

Here is the caller graph for this function:



4.12.2.4 int getRFDuinoTempinC (void)

Here is the caller graph for this function:



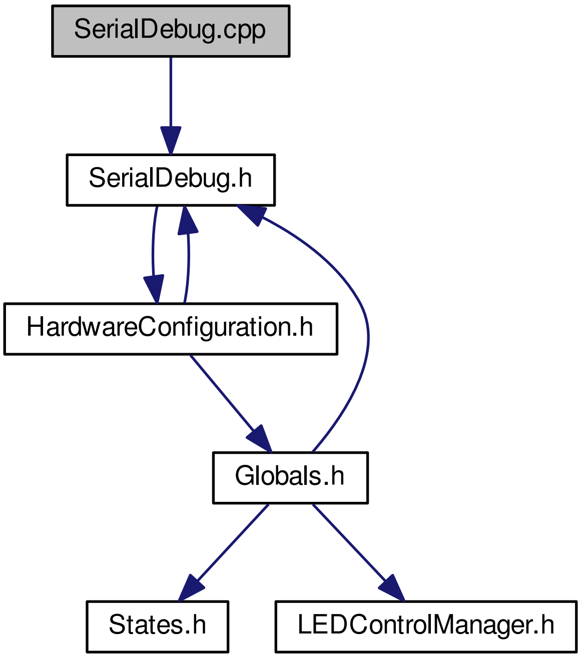
4.12.3 Variable Documentation

4.12.3.1 sensorFeedback sensorFeedback

4.13 SerialDebug.cpp File Reference

`#include "SerialDebug.h"`

Include dependency graph for SerialDebug.cpp:



Functions

- void checkSerial (void)
- void checkDebugSendTime (void)

Variables

- enum DebugLevels SerialDebugLevel = Low

4.13.1 Function Documentation

4.13.1.1 void checkDebugSendTime (void)

4.13.1.2 void checkSerial (void)

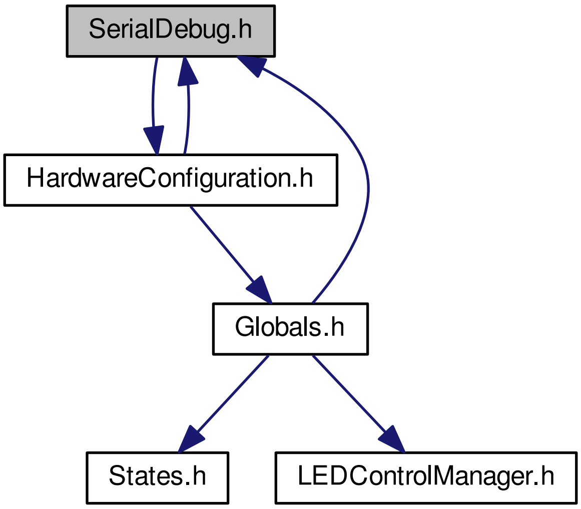
4.13.2 Variable Documentation

4.13.2.1 enum DebugLevels SerialDebugLevel = Low

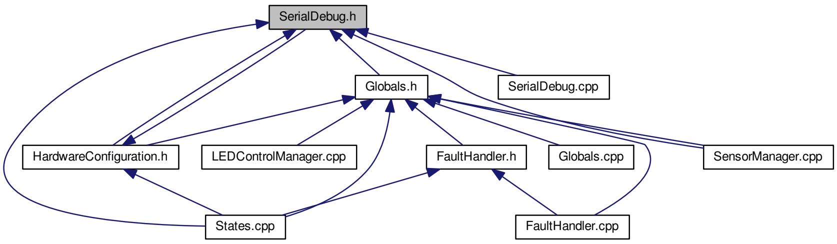
4.14 SerialDebug.h File Reference

`#include "HardwareConfiguration.h"`

Include dependency graph for SerialDebug.h:



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



Macros

- #define TERMINAL_LF Serial.println
- #define TERMINAL Serial.print
- #define DEBUG_LF(level, message) if(SerialDebugLevel!=Off && level <= SerialDebugLevel) Serial.↩ println(message)
- #define DEBUG(level, message) if(SerialDebugLevel!=Off && level <= SerialDebugLevel) Serial.↩ print(message)
- #define TIMED_DEBUG if(FlashMemoryData.dataRefreshTimeMet)

Enumerations

- enum DebugLevels {
  Off, Sensor, Low, Medium,
  High }

Functions

- void checkSerial (void)
- void checkDebugSendTime (void)

Variables

- enum DebugLevels SerialDebugLevel

4.14.1 Macro Definition Documentation

4.14.1.1 #define DEBUG(*level,* *message*) if(SerialDebugLevel!=Off && level <= SerialDebugLevel) Serial.print(message)

4.14.1.2 #define DEBUG_LF(*level,* *message*) if(SerialDebugLevel!=Off && level <= SerialDebugLevel) Serial.println(message)

4.14.1.3 #define TERMINAL Serial.print

4.14.1.4 #define TERMINAL_LF Serial.println

4.14.1.5 #define TIMED_DEBUG if(FlashMemoryData.dataRefreshTimeMet)

4.14.2 Enumeration Type Documentation

4.14.2.1 enum DebugLevels

Enumerator

Off

Sensor

Low

Medium

High

4.14.3 Function Documentation

4.14.3.1 void checkDebugSendTime (void)

4.14.3.2 void checkSerial (void)

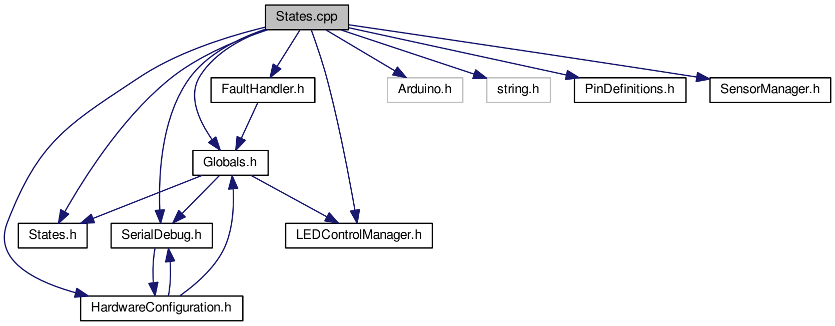
4.14.4 Variable Documentation

4.14.4.1 enum DebugLevels SerialDebugLevel

4.15 States.cpp File Reference

```
#include "States.h"
#include <Arduino.h>
#include <string.h>
#include "PinDefinitions.h"
#include "LEDControlManager.h"
#include "Globals.h"
#include "SensorManager.h"
#include "HardwareConfiguration.h"
#include "SerialDebug.h"
#include "FaultHandler.h"
```

Include dependency graph for States.cpp:



Functions

- void configurePins (void)
- void initializeSystem (void)
- void printStartUpData (void)
- void checkBootComplete (void)
- void SCNominalState (void)
- void SCFaultState (void)
- States checkState (void)
- void setState (void)

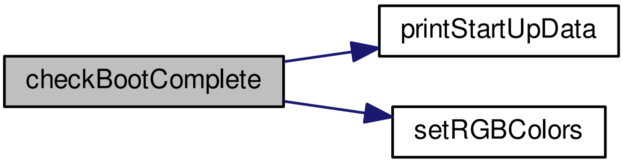
Variables

- static bool newStatePrintFlag = false
- class TimerClass StartupTimer

4.15.1 Function Documentation

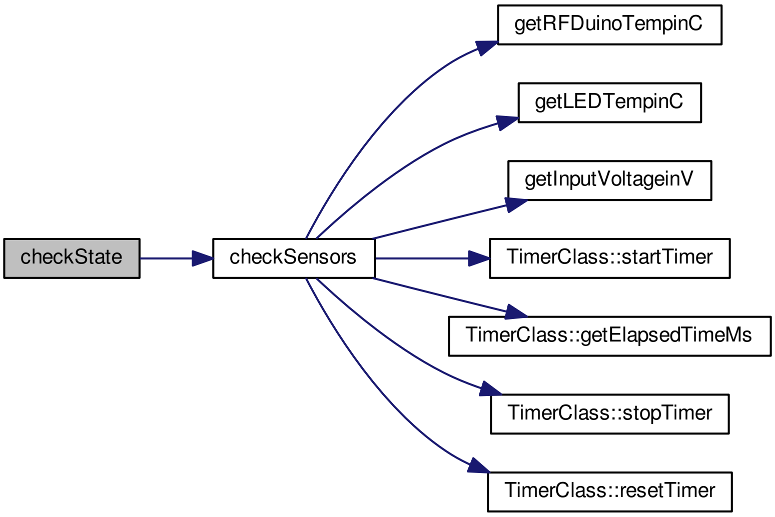
4.15.1.1 void checkBootComplete (void)

Here is the call graph for this function:

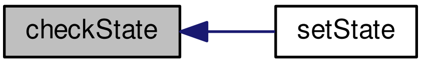


4.15.1.2 States checkState (void)

Here is the call graph for this function:

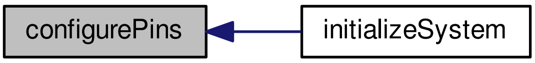


Here is the caller graph for this function:



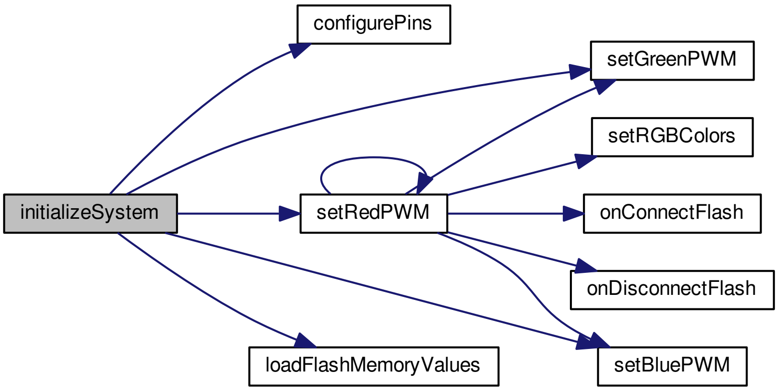
4.15.1.3 void configurePins (void)

Here is the caller graph for this function:



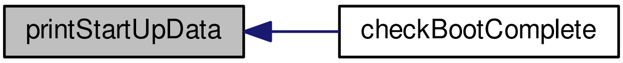
4.15.1.4 void initializeSystem (void)

Here is the call graph for this function:



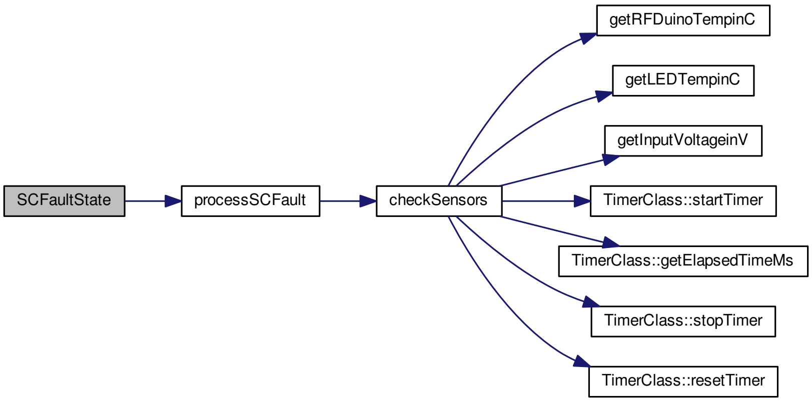
4.15.1.5 void printStartUpData (void)

Here is the caller graph for this function:

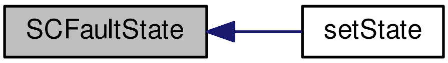


4.15.1.6 void SCFaultState (void)

Here is the call graph for this function:

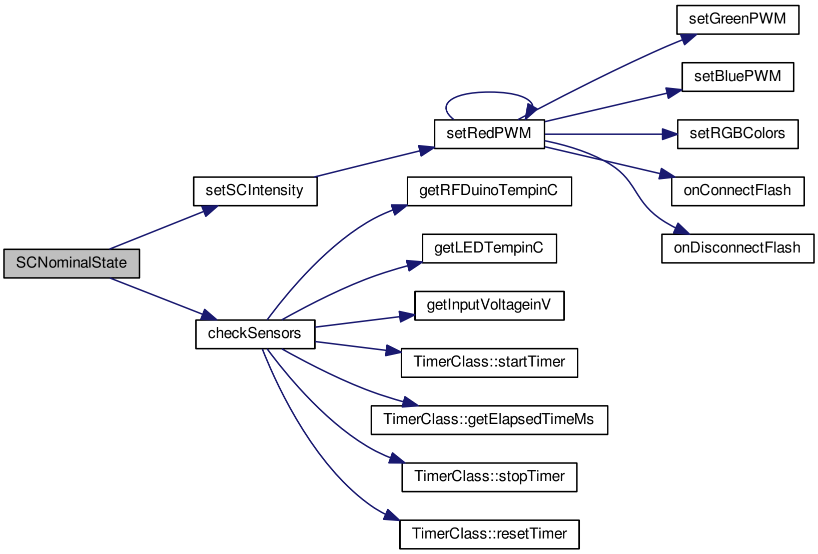


Here is the caller graph for this function:

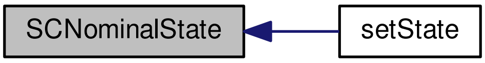


4.15.1.7 void SCNominalState (void)

Here is the call graph for this function:

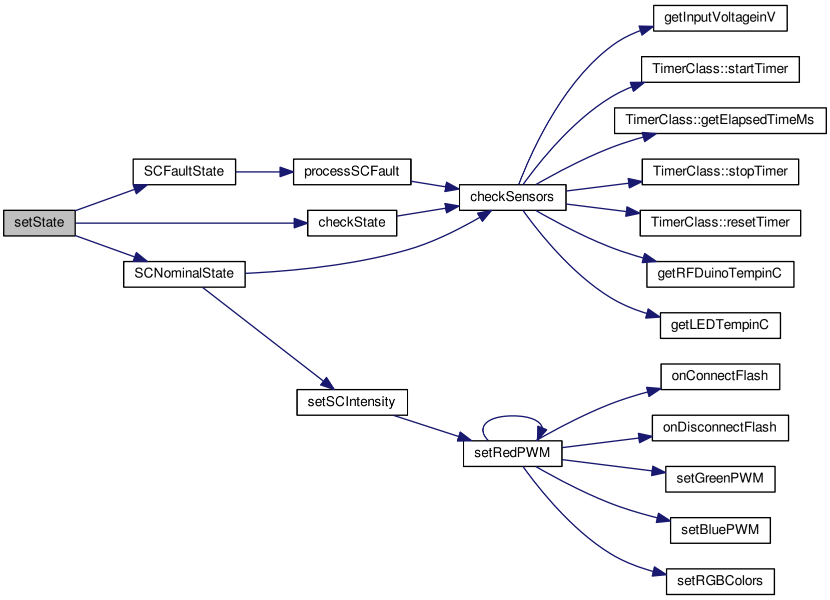


Here is the caller graph for this function:



4.15.1.8 void setState (void)

Here is the call graph for this function:



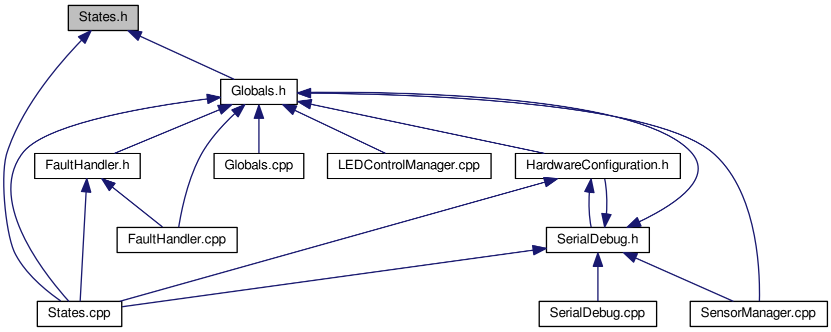
4.15.2 Variable Documentation

4.15.2.1 bool newStatePrintFlag = false [static]

4.15.2.2 class TimerClass StartupTimer

4.16 States.h File Reference

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



Enumerations

- enum States {
  RGB_Disconnected, RGB_Connected, RGB_Fault, SC_Nominal,
  SC_Fault, Boot }

Functions

- void onRadioConnect (void)
- void onRadioDisconnect (void)
- void configurePins (void)
- void initializeSystem (void)
- void printStartUpData (void)
- void checkBootComplete (void)
- void setState (void)

4.16.1 Enumeration Type Documentation

4.16.1.1 enum States

Enumerator

RGB_Disconnected

RGB_Connected

RGB_Fault

SC_Nominal

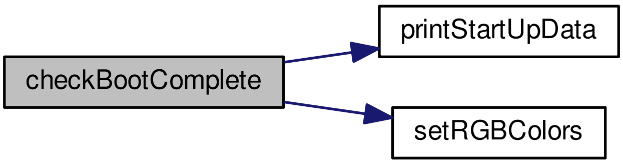
SC_Fault

Boot

4.16.2 Function Documentation

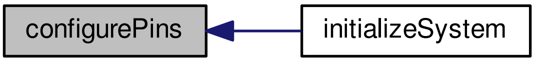
4.16.2.1 void checkBootComplete (void)

Here is the call graph for this function:



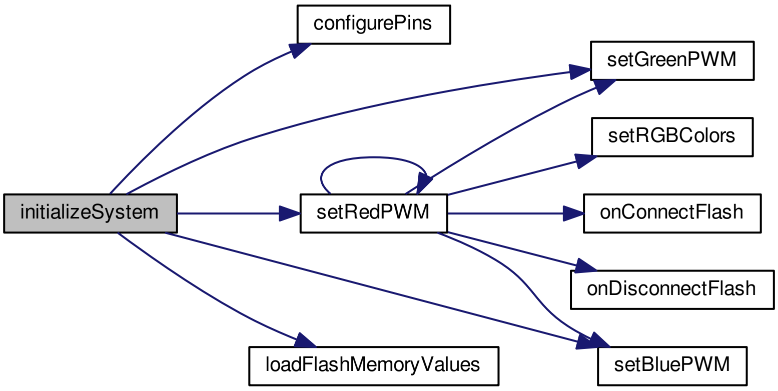
4.16.2.2 void configurePins (void)

Here is the caller graph for this function:



4.16.2.3 void initializeSystem (void)

Here is the call graph for this function:

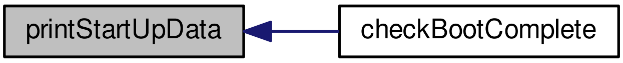


4.16.2.4 void onRadioConnect (void)

4.16.2.5 void onRadioDisconnect (void)

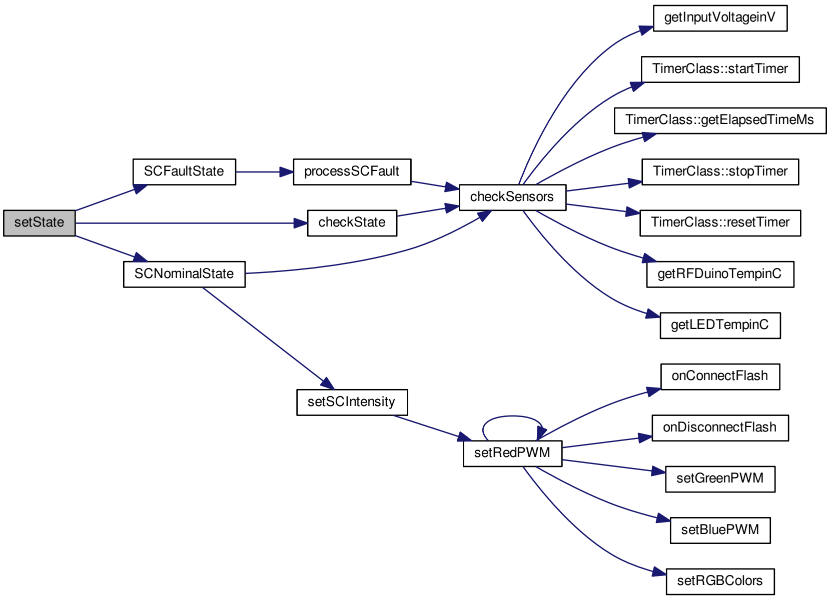
4.16.2.6 void printStartUpData (void)

Here is the caller graph for this function:



4.16.2.7 void setState (void)

Here is the call graph for this function:



Index

ADC_RESOLUTION
    SensorManager.h, 31
ADC_VOLTAGE_REF
    SensorManager.h, 31

BLUE_LED_PIN
    PinDefinitions.h, 28
blueLEDValue
    FlashMemoryData, 6
Boot
    States.h, 43

checkBootComplete
    States.cpp, 38
    States.h, 43
checkDebugSendTime
    SerialDebug.cpp, 34
    SerialDebug.h, 37
checkSensors
    SensorManager.cpp, 29
    SensorManager.h, 31
checkSerial
    SerialDebug.cpp, 34
    SerialDebug.h, 37
checkState
    States.cpp, 38
configurePins
    States.cpp, 39
    States.h, 43

DEBUG_LF
    SerialDebug.h, 36
DEBUG_REFRESH_RATE
    HardwareConfiguration.h, 20
DEBUG
    SerialDebug.h, 36
dataRefreshTimeMet
    FlashMemoryData, 6
debugDataRefreshRateMs
    FlashMemoryData, 6
DebugLevels
    SerialDebug.h, 36
driverName
    FlashMemoryData, 6
driverSerialNumber
    FlashMemoryData, 6

elapsedTimeMs
    Globals.cpp, 17

FaultHandler.cpp, 11
    inputVoltageErrorBehavior, 12
    ledOverTempBehavior, 12
    processRGBFault, 12
    processSCFault, 13
    reactToSensors, 13
    TempErrorTimer, 14
    VoltageTimer, 14
FaultHandler.h, 14
    LED_OVERTEMP_FIRST_REDUCTION_PERC↩
        ENTAGE, 15
    processRGBFault, 15
    processSCFault, 15
FlashMemoryData, 5
    blueLEDValue, 6
    dataRefreshTimeMet, 6
    debugDataRefreshRateMs, 6
    driverName, 6
    driverSerialNumber, 6
    Globals.cpp, 17
    Globals.h, 18
    greenLEDValue, 6
    mode, 6
    modeTiming1, 6
    modeTiming2, 6
    modeTiming3, 6
    partNumber, 6
    redLEDValue, 6
    rfTelemetryEnabled, 6
    serialDebuggingLevel, 6
    serialTelemetryEnabled, 6
    temperatureProtectionEnabled, 6
    testDataOutputEnabled, 6
    timeOfUseMin, 6
    voltageProtectionEnabled, 6

GREEN_LED_PIN
    PinDefinitions.h, 28
gbl_SCStatus
    Globals.cpp, 17
    Globals.h, 18
gbl_systemBootFlag
    Globals.cpp, 17
    Globals.h, 18
gbl_systemState
    Globals.cpp, 17
    Globals.h, 19
gbl_systemTimerinMs
    Globals.cpp, 17
    Globals.h, 19
getElapsedTimeMs

TimerClass, 9
getInputVoltageinmV
SensorManager.h, 32
getInputVoltageinV
SensorManager.cpp, 29
getLEDTempinC
SensorManager.cpp, 30
SensorManager.h, 32
getRFDuinoTempinC
SensorManager.h, 33
Globals.cpp, 16
elapsedTimeMs, 17
FlashMemoryData, 17
gbl_SCStatus, 17
gbl_systemBootFlag, 17
gbl_systemState, 17
gbl_systemTimerinMs, 17
loadFlashMemoryValues, 16
startTimeMs, 17
stopTimeMs, 17
Globals.h, 17
FlashMemoryData, 18
gbl_SCStatus, 18
gbl_systemBootFlag, 18
gbl_systemState, 19
gbl_systemTimerinMs, 19
loadFlashMemoryValues, 18
writeAllSettingsToFlash, 18
writeNewTimeofUseToFlash, 18
greenLEDValue
FlashMemoryData, 6

HardwareConfiguration.h, 19
DEBUG_REFRESH_RATE, 20
RF_TELEMETRY_ENABLED, 20
RGB_DRIVER, 20
row1, 20
row2, 20
row3, 20
row4, 20
row5, 20
row6, 20
SERIAL_TELEMETRY_ENABLED, 20
SOFTWARE_VERSION, 20
High
SerialDebug.h, 36

INPUT_VOLTAGE_CRITICAL_V
SensorManager.h, 31
INPUT_VOLTAGE_PIN
PinDefinitions.h, 28
INPUT_VOLTAGE_SAG_TIME_MS
SensorManager.h, 31
INPUT_VOLTAGE_WARNING_V
SensorManager.h, 31
initializeSystem
States.cpp, 39
States.h, 44
inputHitCriticalVoltage
sensorFeedback, 8
inputHitWarningVoltage
sensorFeedback, 8
inputVoltageErrorBehavior
FaultHandler.cpp, 12
inputVoltageinV
sensorFeedback, 8

LED_BOARD_TEMP_PIN
PinDefinitions.h, 28
LED_CRITICAL_TEMP_C
SensorManager.h, 31
LED_OVERTEMP_FIRST_REDUCTION_PERCENT↩AGE
FaultHandler.h, 15
LED_WARNING_TEMP_C
SensorManager.h, 31
LEDControlManager.cpp, 20
setRedPWM, 21
setSCIntensity, 22
LEDControlManager.h, 23
onConnectFlash, 23
onDisconnectFlash, 23
setBluePWM, 24
setGreenPWM, 24
setRGBColors, 25
setRedPWM, 24
setSCIntensity, 25
LEDHitCriticalTemp
sensorFeedback, 8
LEDHitWarningTemp
sensorFeedback, 8
LEDIntensity
SCStatus, 7
LEDTempinC
sensorFeedback, 8
ledOverTempBehavior
FaultHandler.cpp, 12
loadFlashMemoryValues
Globals.cpp, 16
Globals.h, 18
Low
SerialDebug.h, 36

MAX_NUMBER_OF_FLASH_WRITES
Memory.h, 27
Medium
SerialDebug.h, 36
Memory.c, 26
Memory.h, 26
MAX_NUMBER_OF_FLASH_WRITES, 27
TIME_OF_USE_PERIOD_IN_MS, 27
mode
FlashMemoryData, 6
modeTiming1
FlashMemoryData, 6
modeTiming2
FlashMemoryData, 6
modeTiming3

FlashMemoryData, 6

newStatePrintFlag
States.cpp, 42

Off
SerialDebug.h, 36
onConnectFlash
LEDControlManager.h, 23
onDisconnectFlash
LEDControlManager.h, 23
onRadioConnect
States.h, 44
onRadioDisconnect
States.h, 44

partNumber
FlashMemoryData, 6
PinDefinitions.h, 27
BLUE_LED_PIN, 28
GREEN_LED_PIN, 28
INPUT_VOLTAGE_PIN, 28
LED_BOARD_TEMP_PIN, 28
RED_LED_PIN, 28
printStartUpData
States.cpp, 39
States.h, 44
processRGBFault
FaultHandler.cpp, 12
FaultHandler.h, 15
processSCFault
FaultHandler.cpp, 13
FaultHandler.h, 15

RED_LED_PIN
PinDefinitions.h, 28
RF_TELEMETRY_ENABLED
HardwareConfiguration.h, 20
RFDUINO_CRITICAL_TEMP_C
SensorManager.h, 31
rFDuinoOverTemp
sensorFeedback, 8
rFDuinoTempinC
sensorFeedback, 8
RGB_Connected
States.h, 43
RGB_DRIVER
HardwareConfiguration.h, 20
RGB_Disconnected
States.h, 43
RGB_Fault
States.h, 43
reactToSensors
FaultHandler.cpp, 13
redLEDValue
FlashMemoryData, 6
resetTimer
TimerClass, 9
rfTelemetryEnabled
FlashMemoryData, 6
row1
HardwareConfiguration.h, 20
row2
HardwareConfiguration.h, 20
row3
HardwareConfiguration.h, 20
row4
HardwareConfiguration.h, 20
row5
HardwareConfiguration.h, 20
row6
HardwareConfiguration.h, 20

SC_Fault
States.h, 43
SC_Nominal
States.h, 43
SCFaultState
States.cpp, 40
SCNominalState
States.cpp, 40
SCStatus, 7
LEDIntensity, 7
SCdriverFaultFlag, 7
SCdriverFaultFlag
SCStatus, 7
SERIAL_TELEMETRY_ENABLED
HardwareConfiguration.h, 20
SOFTWARE_VERSION
HardwareConfiguration.h, 20
Sensor
SerialDebug.h, 36
sensorFeedback, 7
inputHitCriticalVoltage, 8
inputHitWarningVoltage, 8
inputVoltageinV, 8
LEDHitCriticalTemp, 8
LEDHitWarningTemp, 8
LEDTempinC, 8
rFDuinoOverTemp, 8
rFDuinoTempinC, 8
SensorManager.cpp, 30
SensorManager.h, 33
SensorManager.cpp, 28
checkSensors, 29
getInputVoltageinV, 29
getLEDTempinC, 30
sensorFeedback, 30
VoltageTime, 30
SensorManager.h, 30
ADC_RESOLUTION, 31
ADC_VOLTAGE_REF, 31
checkSensors, 31
getInputVoltageinmV, 32
getLEDTempinC, 32
getRFDuinoTempinC, 33
INPUT_VOLTAGE_CRITICAL_V, 31
INPUT_VOLTAGE_SAG_TIME_MS, 31

INPUT_VOLTAGE_WARNING_V, 31
LED_CRITICAL_TEMP_C, 31
LED_WARNING_TEMP_C, 31
RFDUINO_CRITICAL_TEMP_C, 31
sensorFeedback, 33
SerialDebug.cpp, 34
checkDebugSendTime, 34
checkSerial, 34
SerialDebugLevel, 34
SerialDebug.h, 35
checkDebugSendTime, 37
checkSerial, 37
DEBUG_LF, 36
DEBUG, 36
DebugLevels, 36
High, 36
Low, 36
Medium, 36
Off, 36
Sensor, 36
SerialDebugLevel, 37
TERMINAL_LF, 36
TERMINAL, 36
TIMED_DEBUG, 36
SerialDebugLevel
SerialDebug.cpp, 34
SerialDebug.h, 37
serialDebuggingLevel
FlashMemoryData, 6
serialTelemetryEnabled
FlashMemoryData, 6
setBluePWM
LEDControlManager.h, 24
setGreenPWM
LEDControlManager.h, 24
setRGBColors
LEDControlManager.h, 25
setRedPWM
LEDControlManager.cpp, 21
LEDControlManager.h, 24
setSCIntensity
LEDControlManager.cpp, 22
LEDControlManager.h, 25
setState
States.cpp, 41
States.h, 44
startTimeMs
Globals.cpp, 17
startTimer
TimerClass, 9
StartupTimer
States.cpp, 42
States
States.h, 43
States.cpp, 37
checkBootComplete, 38
checkState, 38
configurePins, 39
initializeSystem, 39
newStatePrintFlag, 42
printStartUpData, 39
SCFaultState, 40
SCNominalState, 40
setState, 41
StartupTimer, 42
States.h, 42
Boot, 43
checkBootComplete, 43
configurePins, 43
initializeSystem, 44
onRadioConnect, 44
onRadioDisconnect, 44
printStartUpData, 44
RGB_Connected, 43
RGB_Disconnected, 43
RGB_Fault, 43
SC_Fault, 43
SC_Nominal, 43
setState, 44
States, 43
stopTimeMs
Globals.cpp, 17
stopTimer
TimerClass, 9

TERMINAL_LF
SerialDebug.h, 36
TERMINAL
SerialDebug.h, 36
TIME_OF_USE_PERIOD_IN_MS
Memory.h, 27
TIMED_DEBUG
SerialDebug.h, 36
TempErrorTimer
FaultHandler.cpp, 14
temperatureProtectionEnabled
FlashMemoryData, 6
testDataOutputEnabled
FlashMemoryData, 6
timeOfUseMin
FlashMemoryData, 6
timeOutMs
TimerClass, 10
TimerClass, 8
getElapsedTimeMs, 9
resetTimer, 9
startTimer, 9
stopTimer, 9
timeOutMs, 10
TimerClass, 9

voltageProtectionEnabled
FlashMemoryData, 6
VoltageTime
SensorManager.cpp, 30
VoltageTimer
FaultHandler.cpp, 14

writeAllSettingsToFlash
Globals.h, 18
writeNewTimeofUseToFlash
Globals.h, 18