Pressure Alarm System

Omar Elmasri

masri.omarm@gmail.com

Learn in depth diploma

Generated by Doxygen 1.9.3

1 File Index 1

1.1 File List . . . 1

2 File Documentation 3

2.1 mainAlg.c File Reference . . . 3

2.1.1 Detailed Description . . . 4

2.1.2 Function Documentation . . . 4

2.1.2.1 STATE() [1/3] . . . 4

2.1.2.2 STATE() [2/3] . . . 4

2.1.2.3 STATE() [3/3] . . . 5

2.1.3 Variable Documentation . . . 5

2.1.3.1 AlarmFlag . . . 5

2.2 main.c File Reference . . . 5

2.2.1 Detailed Description . . . 5

2.3 globals.h File Reference . . . 6

2.3.1 Detailed Description . . . 6

2.3.2 Variable Documentation . . . 6

2.3.2.1 AlarmFlag . . . 6

2.4 driver_PressureSensor.c File Reference . . . 7

2.4.1 Detailed Description . . . 7

2.4.2 Function Documentation . . . 7

2.4.2.1 PressureSensor_set_pressure() . . . 7

2.5 driver_PressureSensor.h File Reference . . . 8

2.5.1 Detailed Description . . . 8

2.5.2 Function Documentation . . . 8

2.5.2.1 PressureSensor_set_pressure() . . . 8

Index 9

Chapter 1

File Index

1.1 File List

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

driver_PressureSensor.c
: Pressure sensor driver source file 7

driver_PressureSensor.h
: Pressure sensor driver header file 8

globals.h
: Global state pointers 6

main.c
: Pressure alarm system main file 5

mainAlg.c
: Pressure alarm system.
System shall trigger an alarm upon exceeding pressure threshold.

Chapter 2

File Documentation

2.1 mainAlg.c File Reference

Pressure alarm system.
System shall trigger an alarm upon exceeding pressure threshold.

.

```
#include "mainAlg.h"
```

Functions

- void **init** (void)

  MAINALG_H_.
- STATE (HighPressure)

  represent state where pressure value is above or equal threshold
- STATE (SafePressure)

  represent state where pressure value is below threshold
- STATE (ReceivingPressureVal)

  represent state where pressure value is being measured

Variables

- int AlarmFlag = 0

  not used since using the driver provided by the user.

2.1.1 Detailed Description

Author

Omar Elmasri (masri.omarm@gmail.com)
Embedded systems diploma. Learn in depth.

System shall read pressure value using Pressure sensor circuit provided by the user.
System shall compare pressure value to hard-coded pressure threshold provided by the user.
System shall generate an alarm signal to drive alarm circuit provided by the user.
System shall maintain alarm signal for hard-coded period provided by the user.
System shall reset alarm value only after the alarm period is over and pressure is below threshold.

Date

2022-02-21

Version

0.1

Copyright

Copyright (c) 2022

2.1.2 Function Documentation

2.1.2.1 STATE() [1/3]

```
STATE (
            HighPressure  )
```

start alarm

using the driver provided

start timer

switch state

2.1.2.2 STATE() [2/3]

```
STATE (
            ReceivingPressureVal  )
```

read pressure value

switch state

2.1.2.3 STATE() [3/3]

```
STATE (
            SafePressure  )
```

stop alarm

using the driver provided

switch state

2.1.3 Variable Documentation

2.1.3.1 AlarmFlag

```
int AlarmFlag = 0
```

GLOBALS_H_

2.2 main.c File Reference

pressure alarm system main file

Functions

- int **main** (void)

2.2.1 Detailed Description

Author

Omar Elmasri (masri.omarm@gmail.com)

Version

0.1

Date

2022-02-28

Copyright

Copyright (c) 2022

2.3 globals.h File Reference

contains global state pointers

```
#include "driver_PressureSensor.h"
#include "driver_provided.h"
#include "stdint.h"
```

Variables

- void(∗ **state_ptr_main**)(void)
- void(∗ **state_ptr_sensor_pressure**)(void)

  switch main states
- void(∗ **state_ptr_actuator_alarm**)(void)

  not used since using the driver provided by the user.
- int AlarmFlag

  not used since using the driver provided by the user.

2.3.1 Detailed Description

Author

Omar Elmasri (masri.omarm@gmail.com)

Version

0.1

Date

2022-02-26

Copyright

2.3.2 Variable Documentation

2.3.2.1 AlarmFlag

```
int AlarmFlag  [extern]
```

GLOBALS_H_

2.4 driver_PressureSensor.c File Reference

pressure sensor driver source file

```
#include "driver_PressureSensor.h"
#include "driver_provided.h"
```

Functions

- int PressureSensor_set_pressure (void)
  map sensors output to pressure value and return the value.

2.4.1 Detailed Description

Author

Omar Elmasri (masri.omarm@gmail.com)

Version

0.1

Date

2022-02-25

Copyright

Copyright (c) 2022

2.4.2 Function Documentation

2.4.2.1 PressureSensor_set_pressure()

```
int PressureSensor_set_pressure (
            void  )
```

function calls

function invoke getPressureVal function to read the value over the GPIO.
function convert the voltage read from the sensors to relative pressure value.

Returns

int

assuming a unity conversion.

2.5 driver_PressureSensor.h File Reference

pressure sensor driver header file.

```
#include "globals.h"
```

Macros

- #define **PULL_TIME** 1e3

Functions

- int PressureSensor_set_pressure (void)
  function calls

2.5.1 Detailed Description

Author

Omar Elmasri (masri.omarm@gmail.com)

Version

0.1

Date

2022-02-25

Copyright

Copyright (c) 2022

2.5.2 Function Documentation

2.5.2.1 PressureSensor_set_pressure()

```
int PressureSensor_set_pressure (
            void  )
```

DRIVER_PRESSURESENSOR_H_

function calls

function invoke getPressureVal function to read the value over the GPIO.
function convert the voltage read from the sensors to relative pressure value.

Returns

int

assuming a unity conversion.

Index

AlarmFlag
    globals.h, 6
    mainAlg.c, 5

driver_PressureSensor.c, 7
    PressureSensor_set_pressure, 7
driver_PressureSensor.h, 8
    PressureSensor_set_pressure, 8

globals.h, 6
    AlarmFlag, 6

main.c, 5
mainAlg.c, 3
    AlarmFlag, 5
    STATE, 4

PressureSensor_set_pressure
    driver_PressureSensor.c, 7
    driver_PressureSensor.h, 8

STATE
    mainAlg.c, 4