

## My Project

Generated by Doxygen 1.8.17



|                                      |          |
|--------------------------------------|----------|
| <b>1 File Index</b>                  | <b>1</b> |
| 1.1 File List                        | 1        |
| <b>2 File Documentation</b>          | <b>3</b> |
| 2.1 definitions.h File Reference     | 3        |
| 2.2 main.c File Reference            | 3        |
| 2.2.1 Function Documentation         | 3        |
| 2.2.1.1 ADCinit()                    | 4        |
| 2.2.1.2 externalInterruptsInit()     | 4        |
| 2.2.1.3 ISR() [1/2]                  | 4        |
| 2.2.1.4 ISR() [2/2]                  | 4        |
| 2.2.1.5 main()                       | 4        |
| 2.2.1.6 portInit()                   | 5        |
| 2.2.1.7 timersInit()                 | 5        |
| 2.3 motor.c File Reference           | 5        |
| 2.3.1 Function Documentation         | 6        |
| 2.3.1.1 checkShutterPosition()       | 6        |
| 2.3.1.2 EEPROMread()                 | 6        |
| 2.3.1.3 EEPROMwrite()                | 6        |
| 2.3.1.4 isMotorRunning()             | 7        |
| 2.3.1.5 isMotorRunningDown()         | 7        |
| 2.3.1.6 isMotorRunningUp()           | 7        |
| 2.3.1.7 ISR() [1/2]                  | 8        |
| 2.3.1.8 ISR() [2/2]                  | 8        |
| 2.3.1.9 motorDown()                  | 8        |
| 2.3.1.10 motorDownAsLongAs()         | 8        |
| 2.3.1.11 motorOff()                  | 8        |
| 2.3.1.12 motorUp()                   | 9        |
| 2.3.1.13 motorUpASLongAS()           | 9        |
| 2.3.1.14 restoreShutterPosition()    | 9        |
| 2.3.1.15 saveShutterPosition()       | 9        |
| 2.3.1.16 timer0_Off()                | 10       |
| 2.3.1.17 timer0_On()                 | 10       |
| 2.3.1.18 timer2_Off()                | 10       |
| 2.3.1.19 timer2_On()                 | 10       |
| 2.3.1.20 ventilationPosition()       | 11       |
| 2.4 motor.h File Reference           | 11       |
| 2.4.1 Detailed Description           | 11       |
| 2.4.2 Macro Definition Documentation | 12       |
| 2.4.2.1 timer0_ticks                 | 12       |
| 2.4.2.2 timer2_ticks                 | 12       |
| 2.4.3 Function Documentation         | 12       |

---

|                                   |           |
|-----------------------------------|-----------|
| 2.4.3.1 checkShutterPosition()    | 12        |
| 2.4.3.2 EEPROMread()              | 12        |
| 2.4.3.3 EEPROMwrite()             | 13        |
| 2.4.3.4 isMotorRunning()          | 13        |
| 2.4.3.5 isMotorRunningDown()      | 13        |
| 2.4.3.6 isMotorRunningUp()        | 14        |
| 2.4.3.7 motorDown()               | 14        |
| 2.4.3.8 motorDownAsLongAs()       | 14        |
| 2.4.3.9 motorOff()                | 14        |
| 2.4.3.10 motorUp()                | 15        |
| 2.4.3.11 motorUpASLongAS()        | 15        |
| 2.4.3.12 restoreShutterPosition() | 15        |
| 2.4.3.13 saveShutterPosition()    | 15        |
| 2.4.3.14 timer0_Off()             | 16        |
| 2.4.3.15 timer0_On()              | 16        |
| 2.4.3.16 timer2_Off()             | 16        |
| 2.4.3.17 timer2_On()              | 16        |
| 2.4.3.18 ventilationPosition()    | 16        |
| <b>Index</b>                      | <b>17</b> |

# Chapter 1

## File Index

### 1.1 File List

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

|                               |    |
|-------------------------------|----|
| <a href="#">definitions.h</a> | 3  |
| <a href="#">main.c</a>        | 3  |
| <a href="#">motor.c</a>       | 5  |
| <a href="#">motor.h</a>       | 11 |



## Chapter 2

# File Documentation

### 2.1 definitions.h File Reference

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

### 2.2 main.c File Reference

```
#include <stdint.h>
#include <avr/io.h>
#include <avr/portpins.h>
#include <util/delay.h>
#include <stdbool.h>
#include <avr/interrupt.h>
#include <avr/common.h>
#include "definitions.h"
#include "motor.h"
```

Include dependency graph for main.c:

#### Functions

- void [portInit](#) ()
- void [timersInit](#) ()
- void [ADCinit](#) ()
- void [externalInterruptsInit](#) ()
- int [main](#) ()
- [ISR](#) (INT1\_vect)
- [ISR](#) (ADC\_vect)

#### 2.2.1 Function Documentation

### 2.2.1.1 ADCinit()

```
void ADCinit ( )
```

Inicjuje przetwornik nalogowo cyfrowy

#### Returns

Funkcja nic nie zwraca

### 2.2.1.2 externalInterruptsInit()

```
void externalInterruptsInit ( )
```

Inicjuje obsługę przerwań zewnętrznych

#### Returns

Funkcja nic nie zwraca

### 2.2.1.3 ISR() [1/2]

```
ISR (
    ADC_vect )
```

Przerwanie wywoływane gdy zostanie zakończony pomiar ADC z wyjścia potencjometru. Podczas wykonywania przerwania następuje aktualizacja rejestru OCR1A, który jest odpowiedzialny za wypełnienie sygnału PWM

### 2.2.1.4 ISR() [2/2]

```
ISR (
    INT1_vect )
```

przerwanie odpowiadające za korekcję fazy sygnału PWM z fazą napięcia sieciowego

### 2.2.1.5 main()

```
int main ( )
```

< Zmienna służąca do zapamiętywania który z przycisków jest aktualnie naciśnięty



### 2.2.1.6 portInit()

```
void portInit ( )
```

Inicjuje wszystkie potrzebne porty

#### Returns

Funkcja nic nie zwraca

### 2.2.1.7 timersInit()

```
void timersInit ( )
```

Inicjuje timery i odmaskowuje przerwania

#### Returns

Funkcja nic nie zwraca

## 2.3 motor.c File Reference

```
#include "definitions.h"
#include "motor.h"
Include dependency graph for motor.c:
```

### Functions

- int [isMotorRunning](#) ()
- int [isMotorRunningUp](#) ()
- int [isMotorRunningDown](#) ()
- void [motorUp](#) ()
- void [motorDown](#) ()
- void [motorOff](#) ()
- void [ventilationPosition](#) ()
- void [motorUpASLongAS](#) (uint8\_t \*\_keyLock)
- void [motorDownASLongAs](#) (uint8\_t \*\_keyLock)
- void [timer0\\_On](#) ()
- void [timer0\\_Off](#) ()
- void [timer2\\_On](#) ()
- void [timer2\\_Off](#) ()
- void [checkShutterPosition](#) ()
- [ISR](#) (TIMER2\_OVF\_vect)
- [ISR](#) (TIMER0\_OVF\_vect)
- void [saveShutterPosition](#) (unsigned int \_psotion)
- void [restoreShutterPosition](#) ()
- void [EEPROMwrite](#) (unsigned int \_adress, unsigned char \_data)
- unsigned char [EEPROMread](#) (unsigned int \_adress)

## 2.3.1 Function Documentation

### 2.3.1.1 checkShutterPosition()

```
void checkShutterPosition ( )
```

sprawdza pozycję żaluzji, jeżeli pozycja żaluzji jest skrajna (czyli maksymalnie otwarta/zamknięta) to bezwzględnie wyłącza silnik. Funkcja także sprawdza, czy żaluzja znajduje w pozycji wentylacji i zatrzymuje ślinik, jeśli w takiej pozycji się znajduje pod warunkiem że taka była intencja użytkownika

#### Returns

Funkcja nie zwraca nic.

### 2.3.1.2 EEPROMread()

```
unsigned char EEPROMread (
    unsigned int _address )
```

Funkcja odczytuje z pamięci EEPROM pojedynczy bajt

#### Parameters

|                       |   |
|-----------------------|---|
| <code>_address</code> | adres pamięci z którego jest odczytywany bajt |
|-----------------------|---|

#### Returns

odczytany bajt z pamięci EEPROM

### 2.3.1.3 EEPROMwrite()

```
void EEPROMwrite (
    unsigned int _address,
    unsigned char _data )
```

Funkcja zapisuje pojedynczy bajt pod podany adres w pamięci EEPROM

#### Parameters

|                       |   |
|-----------------------|---|
| <code>_address</code> | adres pamięci pod który ma być zapisany bajt                        |
| <code>_data</code>    | bajt danych który jest zapisywany pod wskazany adres pamięci EEPROM |

**Returns**

funckcja nie zwraca nic.

**2.3.1.4 isMotorRunning()**

```
int isMotorRunning ( )
```

Funkcja sprawdza czy silnik jest w ruchu

**Returns**

1, jeśli silnik jest w ruchu

0, jeśli silnik jest w stanie spoczynku

**2.3.1.5 isMotorRunningDown()**

```
int isMotorRunningDown ( )
```

Funkcja sprawdza czy silnik zamyka żaluzje

**Returns**

1, jeśli silnik zamyka żaluzje

0, jeśli silnik nie zamyka żaluzji

**2.3.1.6 isMotorRunningUp()**

```
int isMotorRunningUp ( )
```

Funkcja sprawdza czy silnik otwiera żaluzje

**Returns**

1, jeśli silnik otwiera żaluzje

0, jeśli silnik nie otwiera żaluzji

**2.3.1.7 ISR() [1/2]**

```
ISR (
    TIMER0_OVF_vect )
```

Funkcja obsługi przerwania dla otwierania żaluzji wywoływana co 0.021 s Dekrementuje pozycję żaluzji oraz każde przerwanie sprawdza czy żaluzja osiągnęła zadaną pozycję

**2.3.1.8 ISR() [2/2]**

```
ISR (
    TIMER2_OVF_vect )
```

funkcja obsługi przepełnienia timera2 która zamyka żaluzje timer2 włącza się tylko gdy użytkownik chce zamknąć żaluzje

**2.3.1.9 motorDown()**

```
void motorDown ( )
```

Funkcja włącza silnik, który zamyka żaluzje.

**Returns**

Funkcja nie zwraca nic.

**2.3.1.10 motorDownAsLongAs()**

```
void motorDownAsLongAs (
    uint8_t * _keyLock )
```

Funkcja zamyka żaluzje tak długo aż przycisk jest wcisnięty

**Parameters**

|         |          |  |
|---------|----------|--|
| in, out | _keyLock | zmienna która przechowuje które przyciski są naciśnięte. |
|---------|----------|--|

**Returns**

Funkcja nic nie zwraca

**2.3.1.11 motorOff()**

```
void motorOff ( )
```

Funkcja włącza silnik, aby zamykał żaluzje, jeżeli silnik jest w ruchu funkcja nie zmienia pracy silnika

### 2.3.1.12 motorUp()

```
void motorUp ( )
```

Funkcja włącza silnik, aby zamykał żaluzje, jeżeli silnik jest w ruchu funkcja nie zmienia pracy silnika.

#### Returns

Funkcja nie zwraca nic.

### 2.3.1.13 motorUpASLongAS()

```
void motorUpASLongAS (
    uint8_t * _keyLock )
```

Funkcja otwiera żaluzje tak długo aż przycisk jest wciśnięty.

#### Parameters

|                      |                       |  |
|----------------------|-----------------------|--|
| <code>in, out</code> | <code>_keyLock</code> | zmienna która przechowuje które przyciski są naciśnięte. |
|----------------------|-----------------------|--|

#### Returns

Funkcja nie zwraca nic.

### 2.3.1.14 restoreShutterPosition()

```
void restoreShutterPosition ( )
```

Funkcja odczytuje z pamięci EEPROM zapisaną pozycję żaluzji.

### 2.3.1.15 saveShutterPosition()

```
void saveShutterPosition (
    unsigned int _position )
```

Funkcja zapisuje pozycję żaluzji do pamięci EEPROM

#### Parameters

|                        |                          |
|------------------------|--------------------------|
| <code>_position</code> | Aktualna pozycja żaluzji |
|------------------------|--------------------------|

#### 2.3.1.16 timer0\_Off()

```
void timer0_Off ( )
```

Funkcja wyłącza timer i zeruje jego rejestr

##### Returns

Funkcja nie zwraca nic.

#### 2.3.1.17 timer0\_On()

```
void timer0_On ( )
```

Funkcja włącza timer 0 który odpowiada za wywoływanie przepełnienia co 0.021s Każde wywołanie przepełnienia powoduje inkrementację zmiennej shutterPosition.

##### Returns

Funkcja nie zwraca nic.

#### 2.3.1.18 timer2\_Off()

```
void timer2_Off ( )
```

Funkcja wyłącza timer2, który jest odpowiedzialny za liczenie czasu gdy żaluzja jest zamykana

#### 2.3.1.19 timer2\_On()

```
void timer2_On ( )
```

Funkcja włącza timer2, który jest odpowiedzialny za liczenie czasu gdy żaluzja jest zamykana

##### Returns

Funkcja nie zwraca nic.

### 2.3.1.20 ventilationPosition()

```
void ventilationPosition ( )
```

Wprowadza żaluzje w pozycje wentylacji. Jeśli żaluzja jest w pozycji wentylacji to funkcja nie wprowadza żadnych zmian

#### Returns

Funkcja nie zwraca nic.

## 2.4 motor.h File Reference

```
#include <avr/io.h>
#include <avr/portpins.h>
#include <util/delay.h>
#include <avr/interrupt.h>
```

Include dependency graph for motor.h: This graph shows which files directly or indirectly include this file:

### Macros

- #define [timer2\\_ticks](#) (256 - 156)
- #define [timer0\\_ticks](#) (256 - 164)

### Functions

- void [motorUp](#) ()
- void [motorDown](#) ()
- void [motorOff](#) ()
- int [isMotorRunning](#) ()
- int [isMotorRunningUp](#) ()
- int [isMotorRunningDown](#) ()
- void [motorDownAsLongAs](#) (uint8\_t \*\_keyLock)
- void [motorUpAsLongAs](#) (uint8\_t \*\_keyLock)
- void [checkShutterPosition](#) ()
- void [ventilationPosition](#) ()
- void [timer0\\_On](#) ()
- void [timer0\\_Off](#) ()
- void [timer2\\_On](#) ()
- void [timer2\\_Off](#) ()
- void [saveShutterPosition](#) (unsigned int \_position)
- void [restoreShutterPosition](#) ()
- void [EEPROMwrite](#) (unsigned int \_address, unsigned char \_data)
- unsigned char [EEPROMread](#) (unsigned int \_address)

### 2.4.1 Detailed Description

Plik nagłówkowy pliku [motor.c](#)

## 2.4.2 Macro Definition Documentation

### 2.4.2.1 timer0\_ticks

```
#define timer0_ticks (256 - 164)
```

Liczba od której następuje zliczenia aby uzyć czas przepełnienia 0.021

### 2.4.2.2 timer2\_ticks

```
#define timer2_ticks (256 - 156)
```

Liczba od której następuje zliczenia aby uzyć czas przepełnienia 0.020

## 2.4.3 Function Documentation

### 2.4.3.1 checkShutterPosition()

```
void checkShutterPosition ( )
```

sprawdza pozycję żaluzji, jeżeli pozycja żaluzji jest skrajna (czyli maksymalnie otwarta/zamknięta) to bezwzględnie wyłącza silnik. Funkcja także sprawdza, czy żaluzja znajduje w pozycji wentylacji i zatrzymuje ślinik, jeśli w takiej pozycji się znajduje pod warunkiem że taka była intencja użytkownika

#### Returns

Funkcja nie zwraca nic.

### 2.4.3.2 EEPROMread()

```
unsigned char EEPROMread (
    unsigned int _adress )
```

Funkcja odczytuje z pamięci EEPROM pojedynczy bajt

#### Parameters

|                      |   |
|----------------------|---|
| <code>_adress</code> | adres pamieci z którego jest odczytywany bajt |
|----------------------|---|



**Returns**

odczytany bajt z pamięci EEPROM

**2.4.3.3 EEPROMwrite()**

```
void EEPROMwrite (
    unsigned int _address,
    unsigned char _data )
```

Funkcja zapisuje pojedynczy bajt pod podany adres w pamięci EEPROM

**Parameters**

|                       |   |
|-----------------------|---|
| <code>_address</code> | adres pamięci pod który ma być zapisany bajt                        |
| <code>_data</code>    | bajt danych który jest zapisywany pod wskazany adres pamięci EEPROM |

**Returns**

funckcja nie zwraca nic.

**2.4.3.4 isMotorRunning()**

```
int isMotorRunning ( )
```

Funkcja sprawdza czy silnik jest w ruchu

**Returns**

1, jeśli silnik jest w ruchu  
0, jeśli silnik jest w stanie spoczynku

**2.4.3.5 isMotorRunningDown()**

```
int isMotorRunningDown ( )
```

Funkcja sprawdza czy silnik zamyka żaluzje

**Returns**

1, jeśli silnik zamyka żaluzje  
0, jeśli silnik nie zamyka żaluzji

#### 2.4.3.6 isMotorRunningUp()

```
int isMotorRunningUp ( )
```

Funkcja sprawdza czy silnik otwiera żaluzje

##### Returns

1, jeśli silnik otwiera żaluzje

0, jeśli silnik nie otwiera żaluzji

#### 2.4.3.7 motorDown()

```
void motorDown ( )
```

Funkcja włącza silnik, który zamyka żaluzje.

##### Returns

Funkcja nie zwraca nic.

#### 2.4.3.8 motorDownAsLongAs()

```
void motorDownAsLongAs (
    uint8_t * _keyLock )
```

Funkcja zamyka żaluzje tak długo aż przycisk jest wcisnięty

##### Parameters

|         |          |   |
|---------|----------|---|
| in, out | _keyLock | zmiennna która przechowuje które przyciski są naciśnięte. |
|---------|----------|---|

##### Returns

Funkcja nic nie zwraca

#### 2.4.3.9 motorOff()

```
void motorOff ( )
```

Funkcja włącza silnik, aby zamykał żaluzje, jeżeli silnik jest w ruchu funkcja nie zmienia pracy silnika

#### 2.4.3.10 motorUp()

```
void motorUp ( )
```

Funkcja włącza silnik, aby zamykał żaluzje, jeżeli silnik jest w ruchu funkcja nie zmienia pracy silnika.

##### Returns

Funkcja nie zwraca nic.

#### 2.4.3.11 motorUpASLongAS()

```
void motorUpASLongAS (
    uint8_t * _keyLock )
```

Funkcja otwiera żaluzje tak długo aż przycisk jest wciśnięty.

##### Parameters

|                      |                       |  |
|----------------------|-----------------------|--|
| <code>in, out</code> | <code>_keyLock</code> | zmienna która przechowuje które przyciski są naciśnięte. |
|----------------------|-----------------------|--|

##### Returns

Funkcja nie zwraca nic.

#### 2.4.3.12 restoreShutterPosition()

```
void restoreShutterPosition ( )
```

Funkcja odczytuje z pamięci EEPROM zapisaną pozycję żaluzji.

#### 2.4.3.13 saveShutterPosition()

```
void saveShutterPosition (
    unsigned int _position )
```

Funkcja zapisuje pozycję żaluzji do pamięci EEPROM

##### Parameters

|                        |                          |
|------------------------|--------------------------|
| <code>_position</code> | Aktualna pozycja żaluzji |
|------------------------|--------------------------|

#### 2.4.3.14 timer0\_Off()

```
void timer0_Off ( )
```

Funkcja wyłącza timer i zeruje jego rejestr

##### Returns

Funkcja nie zwraca nic.

#### 2.4.3.15 timer0\_On()

```
void timer0_On ( )
```

Funkcja włącza timer 0 który odpowiada za wywoływanie przepelnienia co 0.021s Każde wywołanie przepelnienia powoduje inkrementację zmiennej shutterPosition.

##### Returns

Funkcja nie zwraca nic.

#### 2.4.3.16 timer2\_Off()

```
void timer2_Off ( )
```

Funkcja wyłącza timer2, który jest odpowiedzialny za liczenie czasu gdy żaluzja jest zamykana

#### 2.4.3.17 timer2\_On()

```
void timer2_On ( )
```

Funkcja włącza timer2, który jest odpowiedzialny za liczenie czasu gdy żaluzja jest zamykana

##### Returns

Funkcja nie zwraca nic.

#### 2.4.3.18 ventilationPosition()

```
void ventilationPosition ( )
```

Wprowadza żaluzje w pozycje wentylacji. Jeśli żaluzja jest w pozycji wentylacji to funkcja nie wprowadza żadnych zmian

##### Returns

Funkcja nie zwraca nic.

# Index

- ADCinit
  - main.c, [3](#)
- checkShutterPosition
  - motor.c, [6](#)
  - motor.h, [12](#)
- definitions.h, [3](#)
- EEPROMread
  - motor.c, [6](#)
  - motor.h, [12](#)
- EEPROMwrite
  - motor.c, [6](#)
  - motor.h, [13](#)
- externalInterruptsInit
  - main.c, [4](#)
- isMotorRunning
  - motor.c, [7](#)
  - motor.h, [13](#)
- isMotorRunningDown
  - motor.c, [7](#)
  - motor.h, [13](#)
- isMotorRunningUp
  - motor.c, [7](#)
  - motor.h, [13](#)
- ISR
  - main.c, [4](#)
  - motor.c, [7](#), [8](#)
- main
  - main.c, [4](#)
- main.c, [3](#)
  - ADCinit, [3](#)
  - externalInterruptsInit, [4](#)
  - ISR, [4](#)
  - main, [4](#)
  - portInit, [4](#)
  - timersInit, [5](#)
- motor.c, [5](#)
  - checkShutterPosition, [6](#)
  - EEPROMread, [6](#)
  - EEPROMwrite, [6](#)
  - isMotorRunning, [7](#)
  - isMotorRunningDown, [7](#)
  - isMotorRunningUp, [7](#)
  - ISR, [7](#), [8](#)
  - motorDown, [8](#)
  - motorDownAsLongAs, [8](#)
  - motorOff, [8](#)
  - motorUp, [8](#)
  - motorUpAsLongAs, [9](#)
  - restoreShutterPosition, [9](#)
  - saveShutterPosition, [9](#)
  - timer0\_Off, [9](#)
  - timer0\_On, [10](#)
  - timer2\_Off, [10](#)
  - timer2\_On, [10](#)
  - ventilationPosition, [10](#)
- motor.h, [11](#)
  - checkShutterPosition, [12](#)
  - EEPROMread, [12](#)
  - EEPROMwrite, [13](#)
  - isMotorRunning, [13](#)
  - isMotorRunningDown, [13](#)
  - isMotorRunningUp, [13](#)
  - motorDown, [14](#)
  - motorDownAsLongAs, [14](#)
  - motorOff, [14](#)
  - motorUp, [14](#)
  - motorUpAsLongAs, [15](#)
  - restoreShutterPosition, [15](#)
  - saveShutterPosition, [15](#)
  - timer0\_Off, [15](#)
  - timer0\_On, [16](#)
  - timer0\_ticks, [12](#)
  - timer2\_Off, [16](#)
  - timer2\_On, [16](#)
  - timer2\_ticks, [12](#)
  - ventilationPosition, [16](#)
- motorDown
  - motor.c, [8](#)
  - motor.h, [14](#)
- motorDownAsLongAs
  - motor.c, [8](#)
  - motor.h, [14](#)
- motorOff
  - motor.c, [8](#)
  - motor.h, [14](#)
- motorUp
  - motor.c, [8](#)
  - motor.h, [14](#)
- motorUpAsLongAs
  - motor.c, [9](#)
  - motor.h, [15](#)
- portInit
  - main.c, [4](#)
- restoreShutterPosition

motor.c, [9](#)  
motor.h, [15](#)

saveShutterPosition  
motor.c, [9](#)  
motor.h, [15](#)

timer0\_Off  
motor.c, [9](#)  
motor.h, [15](#)

timer0\_On  
motor.c, [10](#)  
motor.h, [16](#)

timer0\_ticks  
motor.h, [12](#)

timer2\_Off  
motor.c, [10](#)  
motor.h, [16](#)

timer2\_On  
motor.c, [10](#)  
motor.h, [16](#)

timer2\_ticks  
motor.h, [12](#)

timersInit  
main.c, [5](#)

ventilationPosition  
motor.c, [10](#)  
motor.h, [16](#)