# GISND

1.0

Generated by Doxygen 1.9.1

# Chapter 1

# LICENSE

# Chapter 2

# YMZ284 Sound IC library

Software Library for Yamaha YMZ284 sound chips.

author: Jay Convertino

data: 2024.05.27

license: MIT

## 2.1 Release Versions

### 2.1.1 Current

- release_v0.0.1

### 2.1.2 Past

- none

## 2.2 Requirements

- xc8-cc V2.32
- PIC18F45K50 (MCU can be changed in makefile)
- PICerino development board

## 2.3 Building

- make : builds all
- make dox_gen : doxygen only
- make test : test only
- make libYMZ284.a : static library only
- make clean : remove all build outputs.

## 2.4 Documentation

- See doxygen generated document
- Method for ready check is universal, NOT efficent. Optimize send data for your application!

### 2.4.1 Example Code

```
void main(void)
{
  struct s_ymz284 ymz284;
  /* OSCCON SETUP */
  OSCCONbits.IRCF = 0x7;
  OSCCONbits.OSTS = 0;
  OSCCONbits.SCS  = 0x3;
  OSCCON2bits.PLLEN = 1;
  /* PORT E SETUP */
  INTCON2bits.nRBPU = 1;
  /* disable analog inputs */
  ANSELA = 0;
  ANSELC = 0;
  ANSELD = 0;
  ANSELE = 0;
  /* wait for chip to be ready */
  __delay_ms(100);
  initYMZ284port(&ymz284, &TRISA, &TRISD, 0, 1);
  initYMZ284(&ymz284, &LATA, &LATD);
  setYMZ284mixer(&ymz284, ~0, ~1);
  setYMZ284channel_attn(&ymz284, 'A', 15, 0);
  setYMZ284channel_freq(&ymz284, 'A', 254);
  /* play this lovely tune forever */
  for(;;);
}
```

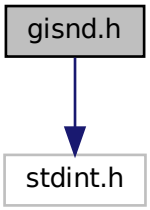
# Chapter 3

# File Documentation

## 3.1 gisnd.h File Reference

```
#include <stdint.h>
```

Include dependency graph for gisnd.h:




### Functions

- void initGISND ()

  *Initialize gisnd and mute.*
- uint16_t getGISND_FreqDiv (uint32_t refClk, uint32_t channelFreq)

  *Calculate frequency from hertz to binary value.*
- uint16_t getGISND_EnvFreqDiv (uint32_t refClk, uint32_t channelFreq)

  *Calculate envelope frequency from hertz to binary value.*
- void setGISNDchannel_freq (char channel, uint16_t freqDiv)

  *Set gisnd channel frequency.*
- void setGISNDchannel_attn (char channel, uint8_t attenuate, uint8_t select)

  *Set gisnd channel attenuation.*
- void setGISNDmixer (uint8_t noise, uint8_t tone)

  *Set gisnd mixer setting.*
- void setGISNDnoise_freq (uint8_t freqDiv)

  *Set gisnd noise frequency.*
- void setGISNDenv_freq (uint16_t freqDiv)

  *Set gisnd envelope frequency.*
- void setGISNDenv_shape (uint8_t shape)

  *Set gisnd envelope shape.*

### 3.1.1 Function Documentation

#### 3.1.1.1 getGISND_EnvFreqDiv()

```
uint16_t getGISND_EnvFreqDiv (
            uint32_t refClk,
            uint32_t channelFreq )
```

Calculate envelope frequency from hertz to binary value.

**Parameters**

| | |
|---|---|
| *refClk* | is the reference clock in hertz for the sound chip. |
| *channelFreq* | is the target frequency in hertz. |

**Returns**

A unsigned 16 bit number that will result in the freqency wanted. ($*$ 512)

#### 3.1.1.2 getGISND_FreqDiv()

```
uint16_t getGISND_FreqDiv (
            uint32_t refClk,
            uint32_t channelFreq )
```

Calculate frequency from hertz to binary value.

**Parameters**

| | |
|---|---|
| *refClk* | is the reference clock in hertz for the sound chip. |
| *channelFreq* | is the target frequency in hertz. |

**Returns**

A unsigned 16 bit number that will result in the freqency wanted. ($*$ 32)

#### 3.1.1.3 initGISND()

```
void initGISND ( )
```

Initialize gisnd and mute.

#### 3.1.1.4 setGISNDchannel_attn()

```
void setGISNDchannel_attn (
            char channel,
            uint8_t attenuate,
            uint8_t select )
```

Set gisnd channel attenuation.

**Parameters**

| | |
|---|---|
| *channel* | Select channel A, B, or C (character input, upper case). |
| *attenuate* | A 4 bit value (0 to 15) |
| *select* | When select is 1, volume control is set by envelope generator, 0 by attenuate. |

#### 3.1.1.5 setGISNDchannel_freq()

```
void setGISNDchannel_freq (
            char channel,
            uint16_t freqDiv )
```

Set gisnd channel frequency.

**Parameters**

| | |
|---|---|
| *channel* | Select channel A, B, or C (character input, upper case). |
| *freqDiv* | is binary number to set the frequency (f = refClk/(32∗freqDiv)) |

#### 3.1.1.6 setGISNDenv_freq()

```
void setGISNDenv_freq (
            uint16_t freqDiv )
```

Set gisnd envelope frequency.

**Parameters**

| | |
|---|---|
| *freqDiv* | is binary number to set the frequency (f = refClk/(512∗freqDiv)) |

#### 3.1.1.7 setGISNDenv_shape()

```
void setGISNDenv_shape (
            uint8_t shape )
```

Set gisnd envelope shape.

**Parameters**

| | |
|---|---|
| *shape* | A 4 bit value that can change the envelope shape, see datasheet. |

#### 3.1.1.8 setGISNDmixer()

```
void setGISNDmixer (
            uint8_t noise,
            uint8_t tone )
```

Set gisnd mixer setting.

**Parameters**

| | |
|---|---|
| *noise* | 0 is enable, 1 is off. bit order C = 2, B = 1, A = 0. |
| *tone* | 0 is enable, 1 is off. bit order C = 2, B = 1, A = 0. |

#### 3.1.1.9 setGISNDnoise_freq()

```
void setGISNDnoise_freq (
            uint8_t freqDiv )
```

Set gisnd noise frequency.

**Parameters**

| | |
|---|---|
| *freqDiv* | is binary number to set the frequency (f = refClk/(32∗freqDiv)) |

## 3.2 LICENSE.md File Reference

## 3.3 README.md File Reference

# Index