AdafruitDRV2605 Library Documentation

Release 1.0

Tony DiCola

Aug 25, 2018

Contents

CircuitPython module for the DRV2605 haptic feedback motor driver.

Dependencies

This driver depends on:

- Adafruit CircuitPython
- Bus Device

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the Adafruit library and driver bundle.

Usage Example

See examples/drv2605_simpletest.py for a demo of the usage.

Contributing

Contributions are welcome! Please read our Code of Conduct before contributing to help this project stay welcoming.

Building locally

To build this library locally you'll need to install the [circuitpython-build-tools](#) package.

```
python3 -m venv .env
source .env/bin/activate
pip install circuitpython-build-tools
```

Once installed, make sure you are in the virtual environment:

```
source .env/bin/activate
```

Then run the build:

```
circuitpython-build-bundles --filename_prefix adafruit-circuitpython-drv2605 --
→library_location .
```

Table of Contents

5.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/drv2605_simpletest.py

```python
# Simple demo of the DRV2605 haptic feedback motor driver.
# Will play all 117 effects in order for about a half second each.
# Author: Tony DiCola
import time

import board
import busio

import adafruit_drv2605


# Initialize I2C bus and DRV2605 module.
i2c = busio.I2C(board.SCL, board.SDA)
drv = adafruit_drv2605.DRV2605(i2c)

# Main loop runs forever trying each effect (1-117).
# See table 11.2 in the datasheet for a list of all the effect names and IDs.
#    http://www.ti.com/lit/ds/symlink/drv2605.pdf
effect = 1
while True:
    print('Playing effect #{0}'.format(effect))
    drv.set_waveform(effect)  # Select the effect on slot 0.
    # Optionally you can assign effects to up to 7 different slots to combine
    # them in interesting ways.  Use the slot keyword and specify a slot 0 to 6
    # (0 is the default).
    #drv.set_waveform(effect, slot=1)
    drv.play()  # Play the effect.
```

```
28        time.sleep(0.5)
29        # Increment effect ID and wrap back around to 1.
30        effect += 1
31        if effect > 117:
32            effect = 1
```

5.2 `adafruit_drv2605`

CircuitPython module for the DRV2605 haptic feedback motor driver. See examples/simpletest.py for a demo of the usage.

> • Author(s): Tony DiCola

class `adafruit_drv2605.`**`DRV2605`**(*i2c*, *address=90*)

> TI DRV2605 haptic feedback motor driver module.
>
> **library**
>
> > The library selected for waveform playback. Should be a value of:
> >
> > > • LIBRARY_EMPTY: Empty
> > >
> > > • LIBRARY_TS2200A: TS2200 library A (the default)
> > >
> > > • LIBRARY_TS2200B: TS2200 library B
> > >
> > > • LIBRARY_TS2200C: TS2200 library C
> > >
> > > • LIBRARY_TS2200D: TS2200 library D
> > >
> > > • LIBRARY_TS2200E: TS2200 library E
> > >
> > > • LIBRARY_LRA: LRA library
> >
> > See the datasheet for the meaning and description of effects in each library.
>
> **mode**
>
> > The mode of the chip. Should be a value of:
> >
> > > • MODE_INTTRIG: Internal triggering, vibrates as soon as you call play(). Default mode.
> > >
> > > • MODE_EXTTRIGEDGE: External triggering, edge mode.
> > >
> > > • MODE_EXTTRIGLVL: External triggering, level mode.
> > >
> > > • MODE_PWMANALOG: PWM/analog input mode.
> > >
> > > • MODE_AUDIOVIBE: Audio-to-vibration mode.
> > >
> > > • MODE_REALTIME: Real-time playback mode.
> > >
> > > • MODE_DIAGNOS: Diagnostics mode.
> > >
> > > • MODE_AUTOCAL: Auto-calibration mode.
> >
> > See the datasheet for the meaning of modes beyond MODE_INTTRIG.
>
> **play**()
>
> > Play back the select effect(s) on the motor.
>
> **set_waveform**(*effect_id*, *slot=0*)
>
> > Select an effect waveform for the specified slot (default is slot 0, but up to 7 effects can be combined with

slot values 0 to 6). See the datasheet for a complete table of effect ID values and the associated waveform / effect.

stop()
Stop vibrating the motor.

use_ERM()
Use an eccentric rotating mass motor (the default).

use_LRM()
Use a linear resonance actuator motor.

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

a