HFEK DRV2605 Arduino Library

PRECISION MICRODRIVES™

The DRV2605 and DRV2605L from Texas Instruments are functionally similar, the DRV2605L is available in a 10VSSOP package and is used in the Haptic Feedback Evaluation Kit. However, you should be able to use the library with either chip. If you have any issues in Development mode, please contact us.

Parent	Notes	Function Syntax	Description	Parameters	Returns
Motor.cpp	Recommended object: **Motor motor = Motor();** The Motor object encapsulates DRV2605 functionality and gives a simpler interface to playing vibration alerts and haptic effects. Only one should be instantiated at a time, as demonstrated in the IntroMode and EngineeringMode sketchs	selectMotor(**motorID**);	Selects the output motor on the grip. Required even if using external actuator	**motorID**: uint8_t	Void
		isCalibrated();	Checks to see if selected actuator has undergone calibration	None	Boolean Value True = calibrated False = not calibrated
		autoCalibrate();	Runs auto calibration on selected motor	None	Void
		playVibAlert(**waveform** , **pwr** , **onTime** , **offTime**);	Plays vibration alert on selected actuator, LRA is not supported so motorID ≠ 3	**waveform**: uint8_t **pwr**: uint8_t **onTime**: uint8_t **offTime**: uint8_t	Void
		playFullHaptic(**library** , **effect**);	Plays haptic effect from DRV2605 on selected actuator. If LRA is selected (motor ID = 3) then library 6 must be used	**library**: uint8_t **effect**: uint8_t	Void
		getMotorID();	Gets current motorID	None	Int
		isPlaying();	Checks to see if an actuator is playing a haptic effect	None	Boolean Value True = playing False = not playing
		isPlayingVib();	Checks to see if an actuator is playing a vibration alert. Note 'off times' return a true value	None	Boolean Value True = playing False = not playing
		stopVibAlert();	Immediatly stops the effect / vibration alert being played	None	Void
		getMotorName();	Returns actuator product code	None	Char string
		isPlayingAudio();	Checks to see if DRV2605 is in Audio-to-Haptic mode	None	Int value equal to 1 if in Audio-to-Haptic
		Audio2Haptic(**withKeyPress**);	Haptic-to-Audio mode with calibration for motors	**withKeyPress**: bool	Void
DRV2605.cpp	Recommended object: **extern DRV2605 drv2605;** The DRV2605 class abstracts away the I2C communications to the DRV2605 chip, providing a streamlined interface for playing effects. Only one DRV2605 object should be instantiated. If using the DRV2605 object from the Motor class use 'extern', as above	autoCal(**ratedVoltage** , **overdriveClamp** , **LRA** , **compensation** , **backEMF** , **feedback**);	Manually adjusts features in the DRV2605	**ratedVoltage**: uint8_t **overdriveClamp**: uint8_t **LRA**: boolean **compensation**: uint8_t* **backEMF**: uint8_t* **feedback**: uint8_t*	Boolean Value True = successful False = not successful
		playFullHaptic(**library** , **effect** , **ratedVoltage** , **overdriveClamp** , **compensation** , **backEMF** , **feedback**);	Plays haptic effect from DRV2605 on selected actuator. Manually adjusts the features in the DRV2605	**library**: uint8_t **effect**: uint8_t **ratedVoltage**: uint8_t **overdriveClamp**: uint8_t **compensation**: uint8_t **backEMF**: uint_8 **feedback**: uint8_t	Void
		Audio(**LRA_AUDIO** , **ratedVoltage** , **overdriveClamp** , **compensation** , **backEMF**);	Enters the Audio-to-Haptic on DRV2605. Audio signal is on PWM input pin, requires motor calibration	**LRA_AUDIO**: uint8_t **ratedVoltage**: uint8_t **overdriveClamp**: uint8_t **compensation**: uint8_t **backEMF**: uint8_t	Void
debug.cpp	N/A	freeRAM();	Returns amount of unused RAM space in DRV2605	None	Int

Parameter	Type & Limits	Values	Description
motorID	0 ≤ int ≤ 3	0 = 304-103 1 = 306-109 2 = 308-102 3 = C10-100	Selects the output actuator
waveform	0 ≤ int ≤ 3	0 = Square 1 = Sine 2 = Triangle 3 = Sawtooth	Changes the output waveform for vibration alerts
pwr	0 ≤ int ≤ 100	0% - 100%	Peak output strength of the selected waveform
onTime	0 ≤ int ≤ 255	Tenths of Seconds	Length of time the vibration alert vibrates for
offTim	0 ≤ int ≤ 255	Tenths of Seconds	Length of time between vibration alerts
library	1 ≤ int ≤ 6	1 - 5 for ERMs, 6 for LRAs	The haptic library on the DRV2605L to be used
effect	1 ≤ int ≤ 123	See Effect ID table	The haptic effect on the DRV2605L to be used
ratedVoltage	0 ≤ int ≤ 255	Voltage applied to ERM = **ratedVoltage** x 0.02118	See full DRV2605L datasheet for LRA voltage
overdriveClamp	0 ≤ int ≤ 255	Voltage applied to ERM = **overdriveClamp** x 0.02159	See full DRV2605L datasheet for LRA voltage
LRA	bool	True = LRA Mode, False = ERM Mode	Sets bit 7 of register 0x1A, setting DRV2605L in to LRA or ERM mode
Compensation	0 ≤ int ≤ 255	Auto Calibration Compensation Coefficient = 1 + **compensation** / 255	Manually adjusts compensation for resistive losses in the driver
backEMF	0 ≤ int ≤ 255	Auto Calibration Back EMF (V) = (**backEMF** / 255) x (2.88 V / **BEMFGain**)	Manually adjusts results for back EMD of the actuator, **BEMFGain** is set by **feedback**
feedback	int, specific values only	N/A	Manually adjusts feedback control register, see full DRV2605L datasheet for details

Register	Name	Datasheet	Register	Name	Datasheet
0x00	Status Register	pg 34	**0x15**	Audio-to-Vibe Maximum Output Drive Register	pg 41
0x01	Mode Register	pg 35	**0x16**	Rated Voltage Register	pg 41
0x02	Real-Time Playback Input Register	pg 36	**0x17**	Overdrive Clamp Voltage Register	pg 42
0x03	Library Selection	pg 36	**0x18**	Auto-Calibration Compensation-Result Register	pg 42
0x04:0x0B	Waveforem Sequencer Register	pg 37	**0x19**	Auto-Calibration Back-EMF Result Register	pg 42
0x0C	Go Register	pg 37	**0x1A**	Feedback Control Register	pg 43
0x0D	Overdrive Time Offset Register	pg 38	**0x1B**	Control1 Register	pg 44
0x0E	Sustain Time Offset, Positive Register	pg 38	**0x1C**	Control2 Register	pg 45
0x0F	Sustain Time Offset, Negative Register	pg 39	**0x1D**	Control3 Register	pg 48
0x10	Brake Time Offset Register	pg 39	**0x1E**	Control4 Register	pg 49
0x11	Audio-to-Vibe Control Register	pg 40	**0x1F**	Control5 Register	pg 50
0x12	Audio-to-Vibe Minimum Input Level Register	pg 40	**0x20**	LRA Open Loop Period Register	pg 50
0x13	Audio-to-Vibe Maximum Input Level Register	pg 40	**0x21**	Voltage Monitor Register	pg 51
0x14	Audio-to-Vibe Minimum Output Drive Register	pg 41	**0x22**	LRA Resonance-Period Register	pg 51