# DRV2624 Ultra Low Power Closed-Loop LRA/ERM Haptic Driver with Internal Memory

## 1 Features

- Ultra Low-Power Shutdown Mode
- Low-Power Standby State
- Resistance-Based Actuator Diagnostics
- SimpleDrive One-Wire Vibration Scheme
- Automatic Resonance Tracking and Reporting
- Automatic Overdrive and Braking
- Automatic Level Calibration
- Drive Compensation Over Battery Discharge
- Configurable Battery Monitor with Power Preservation
- Off-Resonance Driving with Auto-Braking
- LRA Waveform Shape Selection
- Integrated RAM with Loopable Waveform Sequencer
- Real-Time Playback (RTP) Mode
- $I^2C$-Controlled Digital Playback Engine
- Hardware and Software Trigger Option
- Automatic Transition to Standby with Auto-Brake
- Optional Interrupt Pin
- 1.8-V Compatible, VDD Tolerant Digital Interface

(1) Patent pending control algorithm

## 2 Applications

- Mobile Phones
- Tablets

## 3 Description

The DRV2624 device is a haptic driver that relies on a proprietary closed-loop architecture to deliver sharp, strong, and consistent haptic effects while optimizing power consumption.

The internal memory and loopable waveform sequencer, together with the automatic overdrive and braking simplifies the process of generating crisp and optimum haptic effects, reducing the burden imposed into the processing unit.

The DRV2624 device features an automatic go-to-standby state and a battery preservation function to help reduce power consumption without user intervention. The NRST pin allows for a full shutdown state for additional power savings.

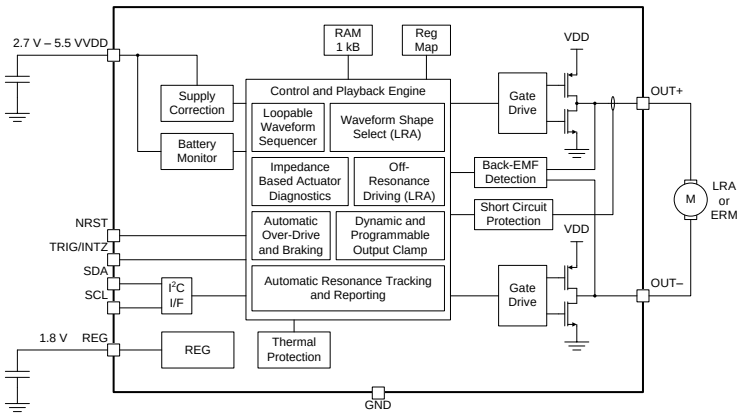
The waveform shape selection allows for sine-wave and square-wave drive to customize the haptic feel as well as the audible performance. Off-resonance driving with automatic braking simplifies the implementation of non-resonant haptic solutions.

**Device Information(1)**

| DEVICE NAME | PACKAGE | BODY SIZE (MAX) |
|---|---|---|
| DRV2624 | DSBGA (9) | 1.498 mm × 1.361 mm |

(1) For all available packages, see the orderable addendum at the end of the datasheet.

**Simplified Schematic**



# Table of Contents



## 4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

**Changes from Revision A (December 2015) to Revision B** — **Page**

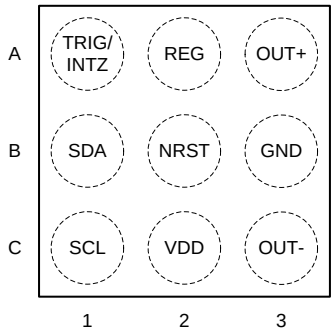
- Updated REV[3:0] from 2 to 3 ... 38
- Changed Default of 0x00 Register ... 38
- Changed the calculation value for the open loop LRA drive of register 0x2E from 'OL_LRA_PERIOD[9:0] × 24.39 µs' to 'OL_LRA_PERIOD[9:0] × 24.615 µs' ... 60

**Changes from Original (December 2015) to Revision A** — **Page**

- Changed data sheet from Product Preview to Production Data ... 1

# 5 Pin Configuration and Functions

**YFF Package**
**9-Pin DSBGA**
**Top View**




## Pin Functions

| PIN | | I/O | DESCRIPTION |
|---|---|---|---|
| NAME | NO. | | |
| VDD | C2 | P | Supply input (2.7 V to 5.5 V). A 0.1-µF capacitor is required. |
| GND | B3 | P | Supply ground |
| REG | A2 | O | 1.8 V regulator output. A 0.1-µF capacitor is required |
| OUT- | C3 | O | Negative haptic driver differential output |
| OUT+ | A3 | O | Positive haptic driver differential output |
| SDA | B1 | I/O | $I^2C$ data |
| SCL | C1 | I | $I^2C$ clock |
| TRIG/INTZ | A1 | I/O | Multi-mode pin. Selectable as input trigger (pulse), input enable, or output interrupt. This pin has an internal pull-down.<br>If pin is not used, it should be connected to ground. |
| NRST | B2 | I | Device reset pin (shutdown mode). If pin is not used, it should be connected to VDD (no internal pull-up or pull-down). |

# 6 Specifications

## 6.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted) (1)

| | | MIN | MAX | UNIT |
|---|---|---|---|---|
| Supply Voltage | $V_{DD}$ | –0.3 | 6 | V |
| Input voltage | NRST | –0.3 | 6 | V |
| | SDA | –0.3 | 6 | V |
| | SCL | –0.3 | 6 | V |
| | TRIG/INTZ | –0.3 | 6 | V |
| Operating free-air temperature range, $T_A$ | | –40 | 85 | °C |
| Operating junction temperature range, $T_J$ | | –40 | 150 | °C |
| Storage temperature, $T_{stg}$ | | –65 | 150 | °C |

(1) Stresses beyond those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Conditions*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

## 6.2 ESD Ratings

| | | | MIN | MAX | UNIT |
|---|---|---|---|---|---|
| $V_{(ESD)}$ | Electrostatic discharge | Human body model (HBM), per ANSI/ESDA/JEDEC JS-001, all pins (1) | –1500 | 1500 | V |
| | | Charged device model (CDM), per JEDEC specification JESD22-C101, all pins (2) | –500 | 500 | |

(1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.
(2) JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process.

## 6.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

| | | MIN | NOM | MAX | UNIT |
|---|---|---|---|---|---|
| $V_{DD}$ | Supply voltage | 2.7 | | 5.5 | V |
| $R_L$ | Load impedance | 8 | | | Ω |
| $C_L$ | Load capacitance | | | 100 | pF |
| $f_{(LRA)}$ | LRA frequency | 45 | | 300 | Hz |

## 6.4 Thermal Information

| THERMAL METRIC (1) | | DRV2625 DSBGA 9 PINS | UNIT |
|---|---|---|---|
| $R_{\theta JA}$ | Junction-to-ambient thermal resistance | 107 | °C/W |
| $R_{\theta JC(top)}$ | Junction-to-case (top) thermal resistance | 0.9 | °C/W |
| $R_{\theta JB}$ | Junction-to-board thermal resistance | 18.1 | °C/W |
| $\psi_{JT}$ | Junction-to-top characterization parameter | 3.8 | °C/W |
| $\psi_{JB}$ | Junction-to-board characterization parameter | 18.1 | °C/W |
| $R_{\theta JC(bot)}$ | Junction-to-case (bottom) thermal resistance | — | °C/W |

(1) For more information about traditional and new thermal metrics, see the *Semiconductor and IC Package Thermal Metrics* application report, SPRA953.

## 6.5 Electrical Characteristics

$T_A$ = 25 °C, $V_{DD}$ = 3.6 V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| $V_{(REG)}$ | Voltage at the REG pin | | | 1.84 | | V |
| $I_{IL}$ | Digital low-level input current | NRST, TRIG/INTZ, SDA, SCL<br>$V_{DD}$ = 5.5 V, $V_I$ = 0 V | | | 100 | nA |
| $I_{IH}$ | Digital high-level input current | SDA, SCL<br>$V_{DD}$ = 5.5 V, $V_I$ = $V_{DD}$ | | | 0.1 | µA |
| | | NRST<br>$V_{DD}$ = 5.5 V, $V_I$ = $V_{DD}$ | | | 1 | |
| | | TRIG/INTZ<br>$V_{DD}$ = 5.5 V, $V_I$ = $V_{DD}$ | | 2.7 | 3.5 | |
| $V_{IL}$ | Digital low-level input voltage | NRST, TRIG/INTZ, SDA, SCL | | | 0.4 | V |
| $V_{IH}$ | Digital high-level input voltage | NRST, TRIG/INTZ, SDA, SCL | 1.41 | | | V |
| $V_{OL}$ | Digital low-level output voltage | TRIG/INTZ, SDA<br>3-mA sink current | | | 0.4 | V |
| $R_{DS(on)}$ | Drain-source on-state resistance (LS + HS) | | | 0.75 | | Ω |
| $I_{(SD)}$ | Shutdown current | $V_{(NRST)}$ = 0 V | | 105 | 180 | nA |
| $I_{(STBY)}$ | Standby current | $V_{(NRST)}$ = $V_{DD}$<br>In stand-by mode | | 1.55 | 2 | µA |
| $I_{(Q)}$ | Quiescent current | $V_{(NRST)}$ = $V_{DD}$<br>In idle mode - no signal | | 2.5 | | mA |
| $Z_{O(SD)}$ | Output impedance in shutdown | OUT+ to GND, OUT– to GND | | 15 | | kΩ |
| $Z_{O(STBY)}$ | Output impedance in standby | OUT+ to GND, OUT– to GND | | 15 | | kΩ |
| $Z_{LOAD(th)}$ | Load impedance threshold for over-current detection | OUT+ to GND, OUT– to GND | | 4 | | Ω |

## 6.6 Timing Requirements

$T_A$ = 25 °C, $V_{DD}$ = 3.6 V (unless otherwise noted)

| | | MIN | NOM | MAX | UNIT |
|---|---|---|---|---|---|
| $f_{(SCL)}$ | Frequency at the SCL pin with no wait states | | | 400 | kHz |
| $t_{w(H)}$ | Pulse duration, SCL high | 0.6 | | | µs |
| $t_{w(L)}$ | Pulse duration, SCL low | 1.3 | | | µs |
| $t_{su(1)}$ | Setup time, SDA to SCL | 100 | | | ns |
| $t_{h(1)}$ | Hold time, SCL to SDA | 10 | | | ns |
| $t_{(BUF)}$ | Bus free time between stop and start condition | 1.3 | | | µs |
| $t_{su(2)}$ | Setup time, SCL to start condition | 0.6 | | | µs |
| $t_{h(2)}$ | Hold time, start condition to SCL | 0.6 | | | µs |
| $t_{su(3)}$ | Setup time, SCL to stop condition | 0.6 | | | µs |

## 6.7 Switching Characteristics

$T_A$ = 25 °C, $V_{DD}$ = 3.6 V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| $t_{(on)}$ | Device startup time | from shutdown standby | | 1 | | ms |
| $t_{(start)}$ | Waveform startup time | from trigger to output signal | | 1 | | ms |
| $f_{O(PWM)}$ | PWM output frequency (in OUT+ and OUT-) | | | 20.5 | | kHz |

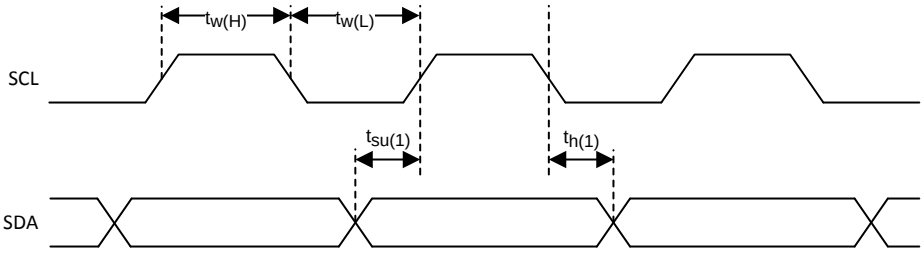



**Figure 1. SCL and SDA Timing**

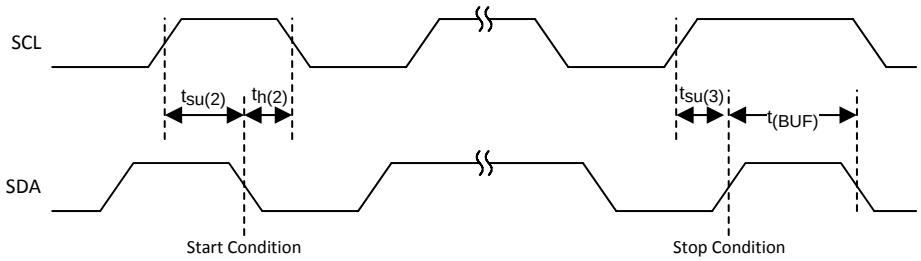



**Figure 2. Timing for Start and Stop Conditions**

## 6.8 Typical Characteristics

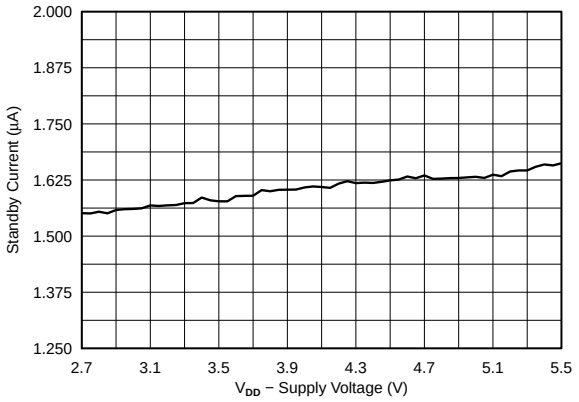
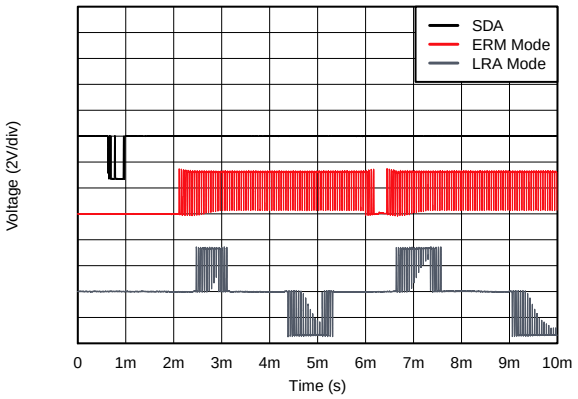
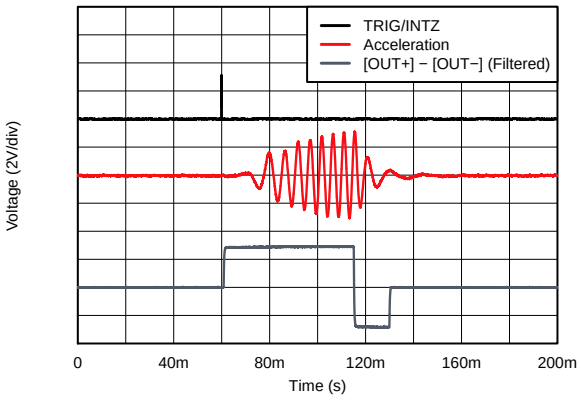



**Figure 3. Standby Current vs Supply Voltage**



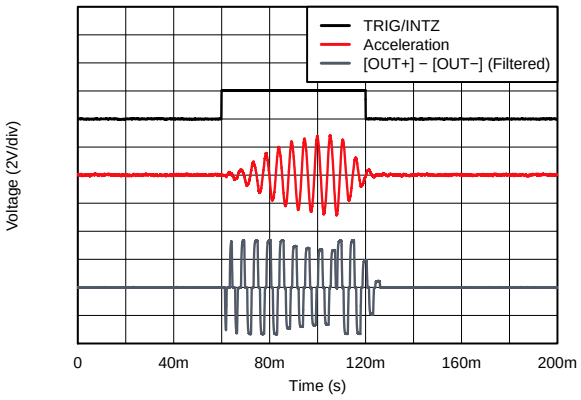

$V_{DD}$ = 3.6 V

**Figure 4. Startup Latency for ERM and LRA**



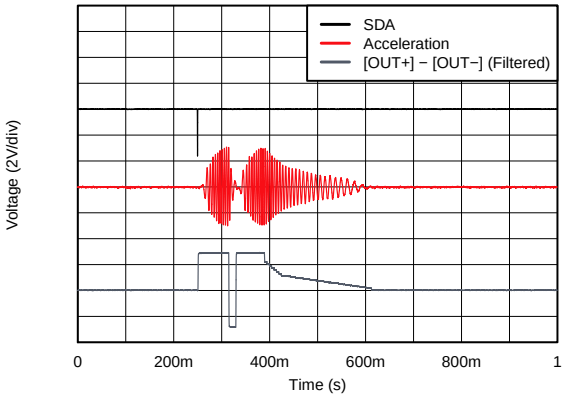

$V_{DD}$ = 3.6 V

**Figure 5. ERM Click (Open-Loop) with External Pulse-Trigger**



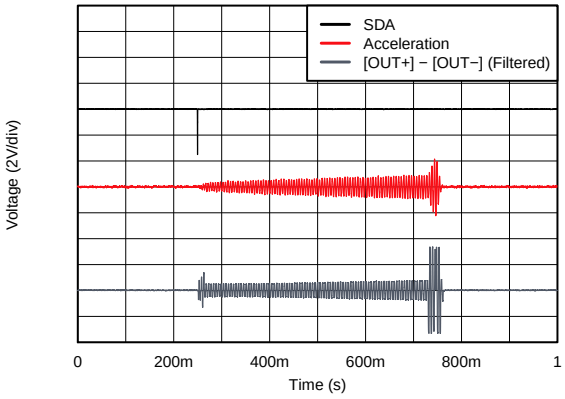

$V_{DD}$ = 3.6 V

**Figure 6. LRA Strong Click (Closed-Loop) with External Level-Trigger**




$V_{DD}$ = 3.6 V

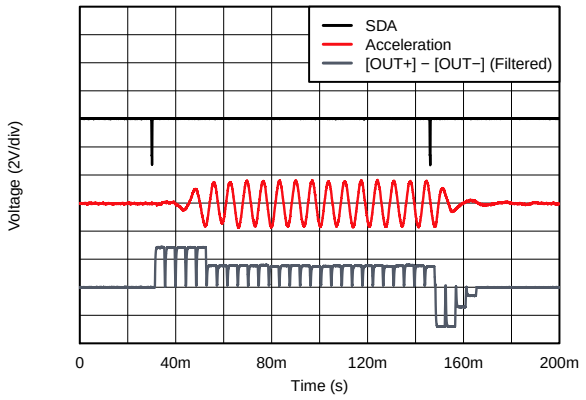
**Figure 7. ERM Click-Bounce (Open-Loop) with Internal Trigger**




$V_{DD}$ = 3.6 V

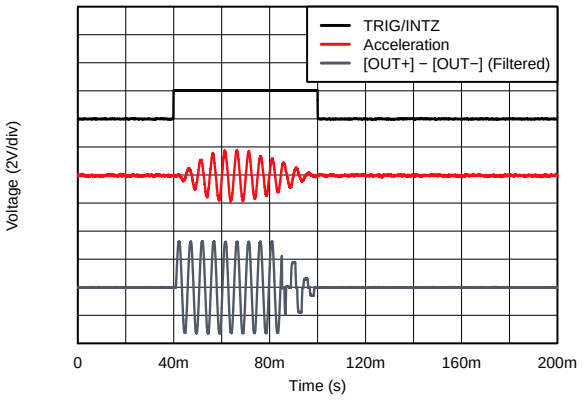
**Figure 8. LRA Transition Click (Closed-Loop) with Internal Trigger**

## Typical Characteristics (continued)



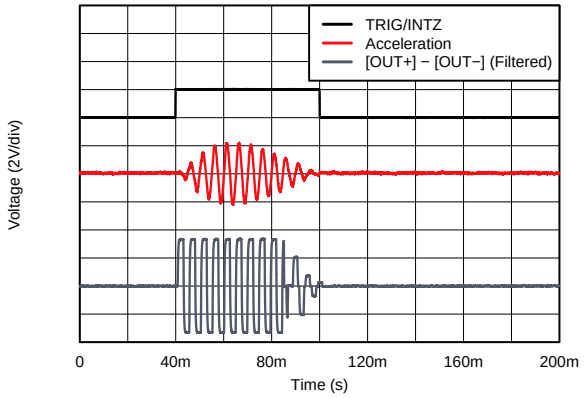

$V_{DD}$ = 3.6 V

**Figure 9. ERM RTP Buzz (Closed-Loop) with Pulse-Trigger**



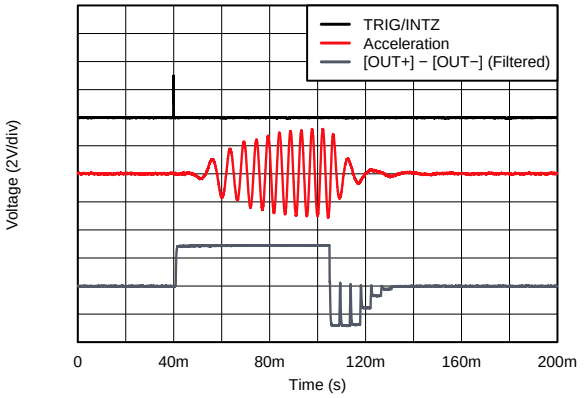

$V_{DD}$ = 3.6 V

**Figure 10. LRA Sinewave Click (Open-Loop) with Automatic Braking**



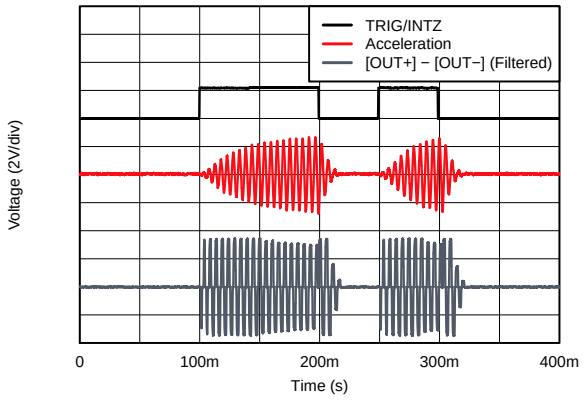

$V_{DD}$ = 3.6 V

**Figure 11. LRA Squarewave Click (Open-Loop) with Automatic Braking**



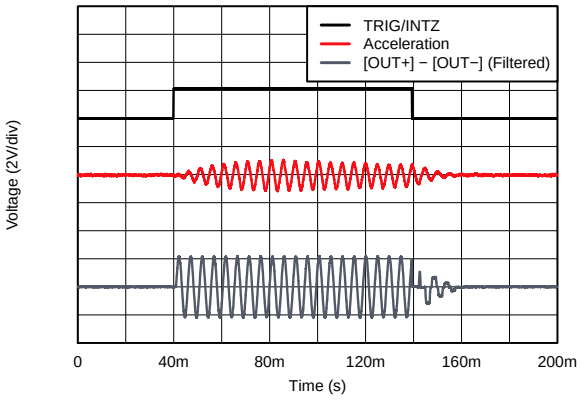

$V_{DD}$ = 3.6 V

**Figure 12. ERM Click (Open-Loop) with Automatic Braking**




$V_{DD}$ = 3.6 V

**Figure 13. LRA SimpleDrive with Closed-Loop**




$V_{DD}$ = 3.6 V

**Figure 14. LRA SimpleDrive with Sinewave**

# 7 Parameter Measurement Information

## 7.1 Test Setup for Graphs

To capture the graphs displayed in the *Typical Characteristics* section, the following first-order RC-filter setup was used with the exception of the waveform in Figure 15 which was captured without any output filter. The filter should be used when viewing output signals on an oscilloscope because output PWM modulation is present in all modes. Ensure that effective impedance of the filter is not too low because the closed-loop and auto resonance-tracking features can be affected. Therefore, TI recommends that this exact filter be used for output measurement. Most oscilloscopes have an input impedance of 1 MΩ on each channel and therefore have an approximately 1% loss in measured amplitude because of the voltage-divider effect with the filter.

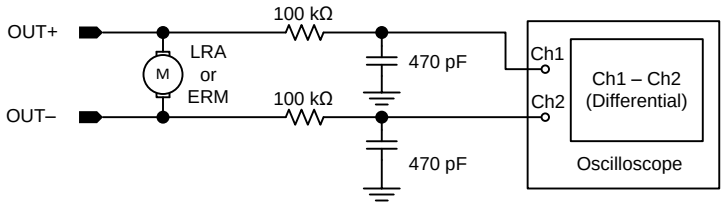



**Figure 15. Test Setup**

### 7.1.1 Default Test Conditions

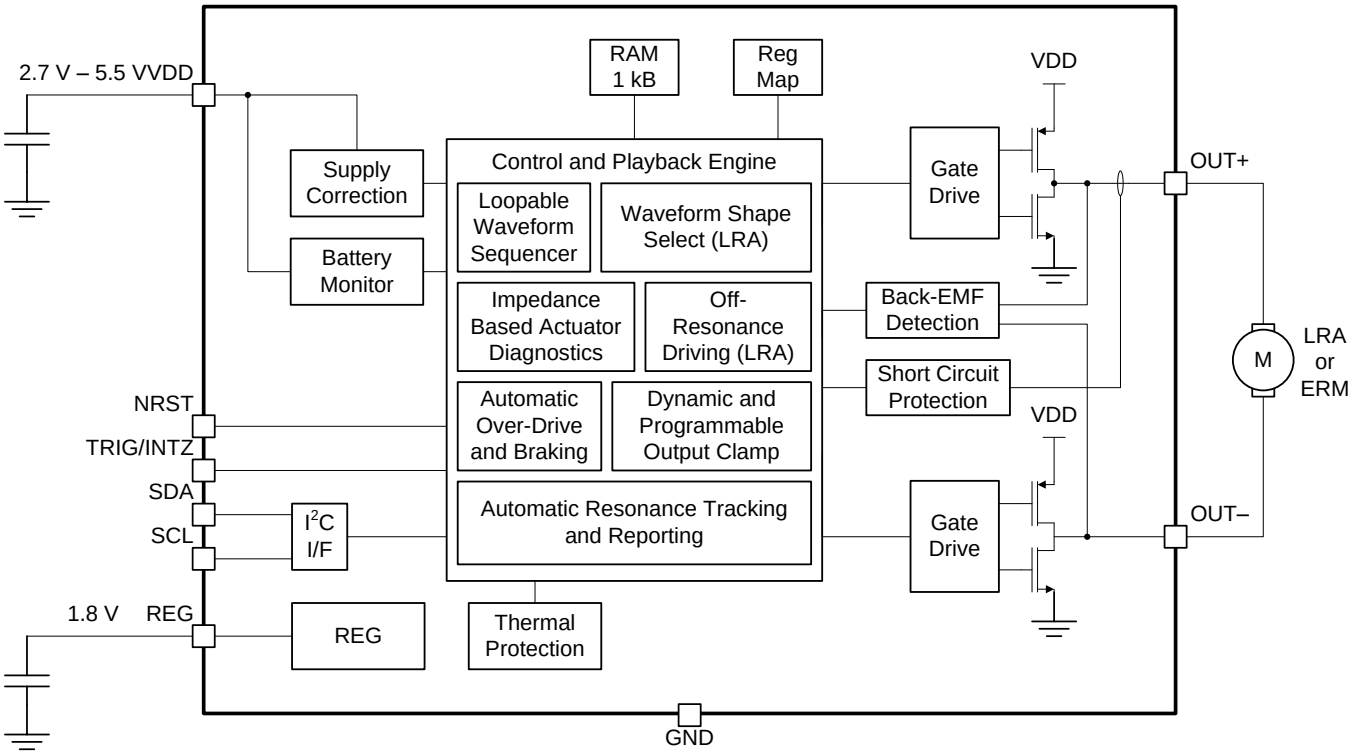
- $V_{DD}$ = 3.6 V, unless otherwise noted.
- Real actuators (as opposed to modeled actuators) were used as loads for both ERM and LRA modes unless otherwise noted.

# 8 Detailed Description

## 8.1 Overview

The DRV2624 device is a haptic driver that relies on a proprietary closed-loop architecture to deliver sharp, strong, and consistent haptic effects while optimizing power consumption. The internal memory and loopable waveform sequencer, the automatic overdrive, and the braking simplifies the process of generating crisp and optimum haptic effects, reducing the burden imposed into the processing unit. The DRV2624 device has an automatic go-to-standby state and a battery preservation function to help reduce power consumption without user intervention. The NRST pin allows for a full shutdown state for additional power savings. The waveform shape selection allows for sine-wave and square-wave drive to customize the haptic feel as well as the audible performance. Off-resonance driving with automatic braking simplifies the implementation of non-resonant haptic solutions.

## 8.2 Functional Block Diagram




## 8.3 Feature Description

### 8.3.1 Support for ERM and LRA Actuators

The DRV2624 device supports both ERM and LRA actuators. The LRA_ERM bit must be configured to select the type of actuator that the device uses.

### 8.3.2 Smart-Loop Architecture

The smart-loop architecture is an advanced closed-loop system that optimizes the performance of the actuator and allows for failure detection. The architecture consists of automatic resonance tracking and reporting (for an LRA), automatic level calibration, accelerated startup and braking, resistance based diagnostics routines, and other proprietary algorithms.

## Feature Description (continued)

### 8.3.2.1 Auto-Resonance Engine for LRA

The DRV2624 auto-resonance engine tracks the resonant frequency of an LRA in real time, effectively locking onto the resonance frequency after half of a cycle. If the resonant frequency shifts in the middle of a waveform for any reason, the engine tracks the frequency from cycle to cycle. The auto-resonance engine accomplishes tracking by constantly monitoring the back-EMF of the actuator. Note that the auto-resonance engine is not affected by the auto calibration process, which is only used for level calibration. No calibration is required for the auto resonance engine.

### 8.3.2.2 Real-Time Resonance-Frequency Reporting for LRA

The smart-loop architecture makes the resonant frequency of the LRA available through $I^2C$. Because frequency reporting occurs in real time, it must be polled while the DRV2624 device synchronizes with the LRA. The polled data should not be polled when the actuator is idle or braking.

### 8.3.2.3 Automatic Switch to Open-Loop for LRA

In the event that an LRA produces a non-valid back-EMF signal, the DRV2624 device automatically switches to open-loop operation and continues to deliver energy to the actuator in overdrive mode at a default and configurable frequency. If the LRA begins to produce a valid back-EMF signal, the auto-resonance engine automatically takes control and continues to track the resonant frequency in real time. When synchronized, this mode uses all of the benefits of the smart-loop architecture.

$$f_{(LRA\_NO\text{-}BEMF)} \approx \frac{1}{2 \times \left(t_{(DRIVE\_TIME[4:0])} - t_{(ZC\_DET\_TIME[1:0])}\right)} \quad (1)$$

The DRV2624 device offers an automatic transition to open-loop mode without the re-synchronization option. This feature is enabled by setting the LRA_AUTO_OPEN_LOOP bit. The transition to open-loop mode only occurs when the driver fails to synchronize with the LRA. The AUTO_OL_CNT[1:0] parameter can be adjusted to set the amount of non-synchronized cycles allowed before the transition to the open-loop mode. Note that the open-loop mode does not receive benefits from the smart-loop architecture, such as automatic overdrive and braking.

$$f_{(LRA\_OL)} = \frac{1}{OL\_LRA\_PERIOD[6:0] \times 97.56 \times 10^{-6}} \quad (2)$$

### 8.3.2.4 Automatic Overdrive and Braking

A key feature of the DRV2624 is the smart-loop architecture which employs actuator feedback control for both ERMs and LRAs. The feedback control desensitizes the input waveform from the motor-response behavior by providing automatic overdrive and automatic braking.

An open-loop haptic system typically drives an overdrive voltage at startup that is higher than the steady-state rated voltage of the actuator to decrease the startup latency of the actuator. Likewise, a braking algorithm must be employed for effective braking. When using an open-loop driver, these behaviors must be contained in the input waveform data. Consider the example of an ERM actuator of Motor A and another of Motor B. The ideal input waveform in open loop is different (see Figure 16). In contrast, by using the smart-loop technology with automatic overdrive and braking, the same input waveform will work optimally for both actuators (see Figure 17). The smart-loop architecture works equally well for LRAs with a combination of feedback control and an auto-resonance engine.

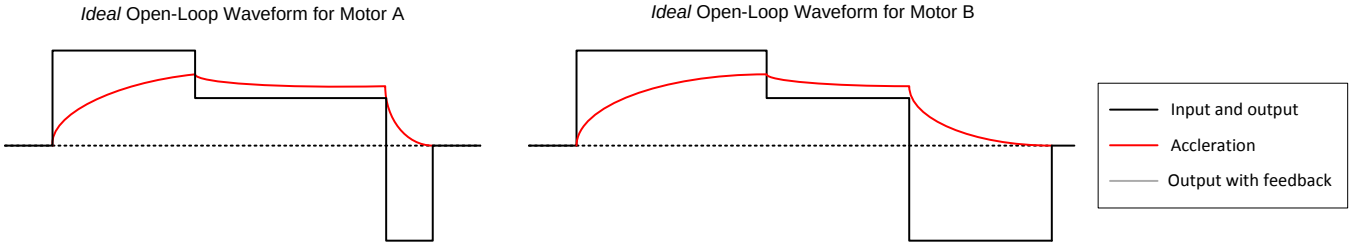



**Figure 16. Typical Open Loop Waveform**

## Feature Description (continued)

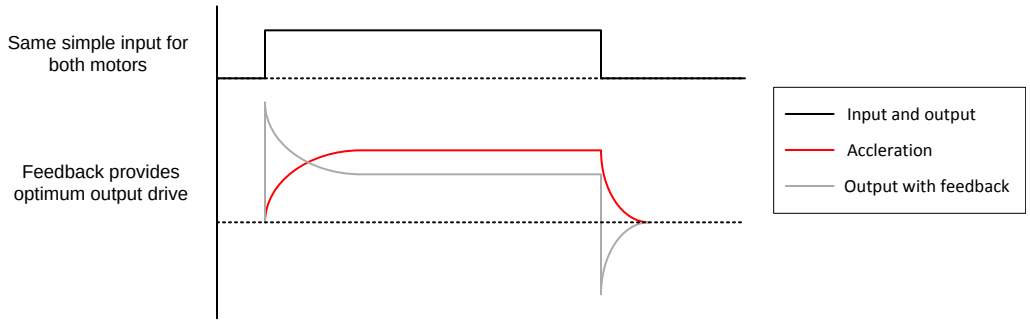



**Figure 17. Waveform Simplification With Smart Loop**

#### 8.3.2.4.1 Startup Boost

To reduce the actuator start-time performance, the DRV2624 device has an overdrive boost feature that applies higher loop gain to transient response of the actuator.

#### 8.3.2.4.2 Brake Factor

To optimize the actuator brake-time performance, the DRV2624 device provides a means to increase the gain ratio between braking and driving gain. Higher feedback-gain ratios reduce the brake time, however, these ratios also reduce the stability of the closed-loop system. The FB_BRAKE_FACTOR parameter can be adjusted to set the brake factor.

### *8.3.2.5 Automatic Level Calibration*

The smart-loop architecture uses actuator feedback by monitoring the back-EMF behavior of the actuator. The level of back-EMF voltage can vary across actuator manufacturers because of the specific actuator construction. Auto calibration compensates for this variation and also performs scaling for the desired actuator according to the specified rated voltage and overdrive clamp-register settings. When auto calibration is performed, a 100% signal level at any of the DRV2624 input interfaces supplies the rated voltage to the actuator at steady-state. The feedback allows the output level to increase above the rated voltage level for automatic overdrive and braking, but the output level does not exceed the programmable overdrive clamp voltage.

#### 8.3.2.5.1 Automatic Compensation for Resistive Losses

The DRV2624 device automatically compensates for resistive losses in the driver. During the automatic level-calibration routine, the resistance of the actuator is checked and the compensation factor is determined and stored in the A_CAL_COMP parameter.

#### 8.3.2.5.2 Automatic Back-EMF Normalization

The DRV2624 device automatically compensates for differences in back-EMF magnitude between actuators. The compensation factor is determined during the automatic level-calibration routine and the factor is stored in the A_CAL_BEMF parameter.

#### 8.3.2.5.3 Calibration Time Adjustment

The duration of the automatic level-calibration routine has an impact on accuracy. The impact is highly dependent on the start-time characteristic of the actuator. The auto-calibration routine expects the actuator to have reached a steady acceleration before the calibration factors are calculated. Because the start-time characteristic can be different for each actuator, the AUTO_CAL_TIME parameter can change the duration of the automatic level-calibration routine to optimize calibration performance. Alternatively, the duration of the calibration routine can be adjusted by the trigger by selecting the option in the AUTO_CAL_TIME parameter.

## Feature Description (continued)

#### 8.3.2.5.4 Loop-Gain Control

The DRV2624 device allows the user to control how fast the driver attempts to match the back-EMF (and thus motor velocity) and the input signal level. Higher loop-gain (or faster settling) options result in less-stable operation than lower loop gain (or slower settling). The LOOP_GAIN parameter controls the loop gain.

#### 8.3.2.5.5 Back-EMF Gain Control

The BEMF_GAIN parameter sets the analog gain for the back-EMF amplifier. The auto-calibration routine automatically populates the BEMF_GAIN bit with the most appropriate value for the actuator.

Modifying the SAMPLE_TIME parameter also adjusts the back-EMF gain. The higher the sample time the higher the gain.

### *8.3.2.6 Actuator Diagnostics*

The DRV2624 device is capable of determining whether the actuator is not present (open) or shorted. If a fault is detected during the diagnostic process, the DIAG_RESULT bit is asserted.

The DRV2624 device also features actuator resistance measurement, which is available in the DIAG_Z_RESULT parameter.

$$R_{(act)} = 478.43 \cdot \frac{DIAG\_Z\_RESULT[7:0]}{719 + 4 \cdot CURRENT\_K[7:0]} \quad (3)$$

### *8.3.2.7 Automatic Re-Synchronization*

For LRA actuators, the DRV2624 device features automatic re-synchronization, which automatically pushes the actuator in the correct direction when a waveform begins playing while the actuator is moving. If the actuator is at rest when the waveform begins, the DRV2624 device drives in the default direction.

### 8.3.3 Open-Loop Operation

In the event that open-loop operation is desired the DRV2624 device includes an open-loop drive mode that overrides any close-loop parameter and is available through the digital interface.

When activated, the digital open-loop mode is available for pre-stored waveforms as well as for RTP mode.

The dynamic range for open-loop operation is set by the OD_CLAMP[7:0], which sets the maximum peak value. Amplitude codes (either through RTP or internal memory) scales the output accordingly.

For LRA actuators, the OL_LRA_PERIOD parameter programs the operating frequency, which is derived from the PWM output frequency, $f_{O(PWM)}$.

### *8.3.3.1 Waveform Shape Selection for LRA*

The DRV2624 offers a selection of either sine-wave or square-wave waveform shape in open-loop mode. The WAVE_SHAPE_LRA parameter selects which shape to use. The WAVE_SHAPE_LRA parameter is ignored in ERM mode and in closed-loop mode.

### *8.3.3.2 Automatic Braking in Open Loop*

The DRV2624 offers automatic braking in open-loop for both ERM and LRA. To accomplish automatic braking, the DRV2624 switches to close-loop during the braking period, therefore resorting to the close-loop waveform shape. The AUTO_BRK_OL parameter can be use to enable or disable the automatic braking feature. To use the automatic braking feature, the device must be configured appropriately for closed-loop operation.

### 8.3.4 Flexible Front-End Interface

The DRV2624 device offers multiple ways to launch and control haptic effects. The MODE parameter selects from either using the waveform sequencer (and hence trigger the waveforms with either an internal or external trigger), or by using RTP mode. Additional flexibility is provided by the multi-purpose TRIG/INTZ pin, which can be configured with the TRIG_PIN_FUNC parameter.

## Feature Description (continued)

### 8.3.4.1 Internal Memory Interface

The DRV2624 device is designed with 1 kB of integrated RAM for waveform storage used by the playback engine. The data is stored in an efficient way (voltage-time pairs) to maximize the number of waveforms that can be carried. The playback engine also has the ability to generate smooth ramps (up or down) by relying on the start-waveform and end-waveform points and by using linear interpolation techniques.

Storing waveforms on the DRV2624 instead of the host processor has several advantages including:

- Offloading processing requirements, such as digital streaming (RTP).
- Improving latency by storing the waveforms on the DRV2624 and only requiring a trigger signal.
- Reducing $I^2C$ traffic by eliminating the requirement to transfer waveform data

#### 8.3.4.1.1 Library Parameterization

The waveforms stored in the internal memory are augmented by the time offset parameters. This augmentation occurs only for the waveforms stored in the internal memory and not for RTP mode. The purpose of this functionality is to add *time stretching* (or time shrinking) to the waveform. This functionality is useful for customizing the entire library of waveforms for a specific actuator rise time and fall time.

The time parameters that can be stretched or shrunk include:

**ODT** Overdrive time

**SPT** Sustain positive time

**SNT** Sustain Negative Time

**BRT** Brake Time

The time values are additive offsets and are 8-bit signed values. The default offset of these values is 0. Positive values add and negative values subtract from the *time* value of the effect that is currently played. The most positive value in the waveform is automatically interpreted as the overdrive time, and the most negative value in the waveform is automatically interpreted as the brake time. These time-offset parameters are applied to both voltage-time pairs and linear ramps. For linear ramps, linear interpolation is stretched (or shrunk) over the two operative points for the period.

$$t_{(final)} = t_{(orig)} + t_{(ofs)} \quad (4)$$

#### 8.3.4.1.2 Playback Interval

The internal memory ticks are by default interpreted as 5-ms intervals. if additional granularity is desired, then a 1-ms interval can be selected by using the PLAYBACK_INTERVAL bit.

#### 8.3.4.1.3 Waveform Sequencer

The waveform sequencer queues waveform identifiers for playback. Eight sequence registers queue up to eight waveforms for sequential playback. A waveform identifier is an integer value referring to the index position of a waveform in the internal library. Once the user has selected the Waveform Playback as the process to run in the MODE[1:0] parameter, playback begins at WAV_FRM_SEQ1 when the user triggers the process (either with the GO bit or externally, if configured to do so). When playback of that waveform ends, the waveform sequencer plays the waveform identifier held in WAV_FRM_SEQ2 if the next waveform is non-zero. The waveform sequencer continues in this way until it reaches an identifier value of zero or until all eight identifiers are played, whichever scenario is reached first.

The waveform identifier range is 1 to 127. The MSB of each sequence register can implement a delay between sequence waveforms. When the MSB is high, bits [6:0] indicate the length of the wait time. The wait time for that step then becomes WAV_FRM_SEQ[6:0] × 10 ms.

The DRV2624 allows for looping each waveform a number of times before moving onto the next waveform identifier. The waveform-looping functionality can be configured by the WAV_SEQ_LOOP parameters.

## Feature Description (continued)

The DRV2624 also allows for looping the entire waveform sequencer by configuring the WAV_SEQ_MAIN parameter. In this case, the waveform sequencer will loop all valid WAV_FRM_SEQn identifiers according to the number specified in the WAV_SEQ_MAIN. For example, if the first and second identifiers are valid (for example 1 and 2), the third identifier is 0 (signaling to stop), and the WAV_SEQ_MAIN is configured to loop once (play the waveform sequence twice), then the DRV2624 device will play waveform 1, then waveform 2, then waveform 1 then waveform 2, and then go to standby.

#### *8.3.4.2 Real-Time Playback (RTP) Interface*

The real-time playback mode is a simple, single 8-bit register interface that holds an amplitude value. When real-time playback is enabled, the RTP_INPUT parameter, which represents an amplitude value, is sent directly to the playback engine. Once triggered, the value is played until the user sends a stop trigger or removes the device from RTP mode. The RTP mode is a digital streaming mode where the user enters a register value over the $I^2C$. Because of the similarity between RTP mode and legacy PWM modes, any API (application-programming interface) designed for use with a PWM generator in the host processor can write the data values over the $I^2C$ rather than writing the data values to the host timer. This ability frees a timer in the host while retaining compatibility with the original software.

For the LRA, the DRV2624 device automatically tracks the resonance frequency unless the CONTROL_LOOP bit is set to open loop operation. If the CONTROL_LOOP bit is set to open loop, the LRA is driven according to the open-loop frequency set in the OL_LRA_PERIOD parameter.

#### *8.3.4.3 Process Trigger*

All processes (RTP, Waveform Sequencer, Calibration and Diagnostics) in the DRV2624 device are triggered processes, which means that the user has to trigger the process before the process begins. A trigger can be achieved by software using the GO bit, or by hardware using the TRIG/INTZ pin. The process stops once it finishes, or if a stop trigger is sent. For information on external trigger functionality refer to Multi-Purpose Pin Functionality.

A typical process (either RTP, Waveform Sequencer, Calibration or Diagnostics) will start and end follwing the diagram in Figure 18.

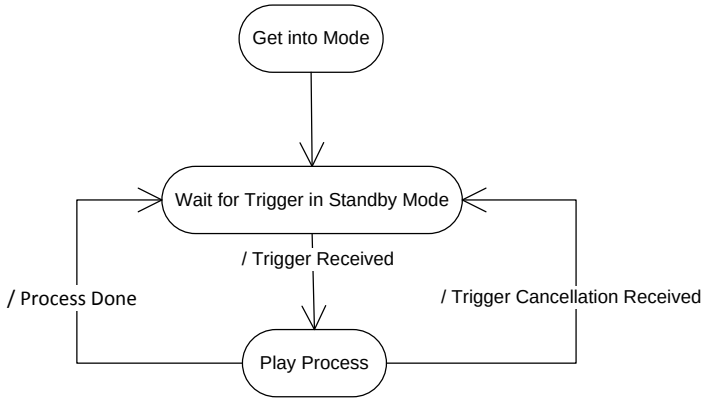



**Figure 18. Typical Process Execution**

### 8.3.5 Noise Gate Control

The DRV2624 device features a noise gate that filters out any voltage smaller than a particular threshold to prevent unintended vibrations. The NG_THRESH bit controls the threshold.

### 8.3.6 Edge Rate Control

The DRV2624 output driver implements edge rate control (ERC). This control ensures that the rise and fall characteristics of the output drivers do not emit levels of radiation that could interfere with other circuitry common in mobile and portable platforms. Because of ERC most system do not require external output filters, capacitors, or ferrite beads.

## Feature Description (continued)

### 8.3.7 Constant Vibration Strength

The DRV2624 device features power-supply feedback. If the supply voltage drifts over time (because of battery discharge, for example), the vibration strength remains the same as long as enough supply voltage is available to sustain the required output voltage.

### 8.3.8 Battery Voltage Reporting

During playback, the DRV2624 device provides cycle-by-cycle voltage measurement of the $V_{DD}$ pin. The VBAT[7:0] parameter provides this information.

### 8.3.9 Ultra Low-Power Shutdown

Setting the device into shutdown state by using the NRST pin reduces the power consumption to sub-micro levels, allowing the system to preserve power when haptics are not required. In this state, register content is not preserved.

### 8.3.10 Automatic Go-To-Stand-by (Low Power)

The DRV2624 automatically goes into a low power standby state when not in use. In this state, register content is preserved and $I^2C$ communication is available. The DRV2624 features a fast turn-on time from standby when requested to play a waveform

### 8.3.11 $I^2C$ Watchdog Timer

If an $I^2C$ stops unexpectedly, the possibility exists for the $I^2C$ protocol to remain in a *hanged* state. To allow for the recovery of the communication without having to power cycle the device, the DRV2624 device includes an automatic watchdog timer that resets the $I^2C$ protocol without user intervention after 4.33 ms.

### 8.3.12 Device Protection

The DRV2624 device has integrated protection circuits for thermal and over-current protection, as well as for UVLO. When such conditions are present, the DRV2624 device will immediately stop playback and go into the standby state. The respective status bit will be set in register 0x01, which will be cleared upon reagister read. An interrupt can be fired if the DRV2624 device is configured to do so.

If the critical condition disappears (the over-current condition goes away), the DRV2624 device will proceed with normal operation, but because the status bits are sticky, they will continue to be asserted until the status register is read.

#### *8.3.12.1 Thermal Sensor*

The DRV2624 has a thermal circuit that immediately puts the device in standby state and sets the OVER_TEMP bit in the event of an over-temperature condition.

If the interrupt functionality is selected in the TRIG_PIN_FUNC parameter and the interrupt is not masked, and interrupt will be fired to alert the host processor of a critical condition.

#### *8.3.12.2 Over-Current Protection*

During waveform playback, if the impedance at the output pin of the DRV2624 device is too low, the DRV2624 device immediately goes into standby state and latches the over-current flag (OC_DETECT bit).

If the interrupt functionality is selected in the TRIG_PIN_FUNC parameter and the interrupt is not masked, an interrupt will be fired to alert the host processor of a critical condition.

#### *8.3.12.3 VDD UVLO Protection*

The DRV2624 device has a battery monitor that monitors the VDD level to ensure that is above a configurable threshold (using UVLO_THRES[2:0] parameter).

In the event of a VDD droop, the DRV2624 device immediately goes into standby state to reduce current consumption and latches the UVLO flag (UVLO bit).

## Feature Description (continued)

If the interrupt functionality is selected in the TRIG_PIN_FUNC parameter and the interrupt is not masked, and interrupt will be fired to alert the host processor of a critical condition.

#### *8.3.12.4 Brownout Protection*

The DRV2624 device has on-chip brownout protection. When activated, a reset signal is issued that returns the DRV2624 device to the initial default state. If the regulator voltage $V_{(REG)}$ goes below the brownout protection threshold ($V_{(BOT)}$) the DRV2624 device automatically shuts down. When $V_{(REG)}$ returns to the typical output voltage (1.8 V) the DRV2624 device returns to the initial device state. The brownout protection threshold ($V_{(BOT)}$) is typically at 1.6 V.

### 8.3.13 POR

The POR circuit was designed to enable the device only when both VDD and REG voltages are at a healthy level. If REG voltage level were to dip below the reset threshold, the device will automatically abort any process and until appropriate power levels are available, once a valid voltage is present in both VDD and REG nodes, the device will proceed with the power-up sequence and return to the default state. If VDD drops below UVLO with VREG still at a healthy level, the device will immediately go into standby state.

### 8.3.14 Silicon Revision Control

The DRV2624 has a revision control implemented in CHIPID[3:0] and REV[3:0] parameter (located in register 0x00). This feature allows an external controller to determine which device is connected to it and select the appropriate firmware to control the device, which makes firmware development easier to port from one platform to another.

### 8.3.15 Support for LRA and ERM Actuators

The DRV2624 device supports both LRA and ERM actuators. The default state is LRA mode, but can be changed by using the LRA_ERM bit.

### 8.3.16 Multi-Purpose Pin Functionality

To enhance the flexibility of the DRV2624, the TRIG/INTZ pin is a configurable, multi-purpose pin that takes different functionality depending on the mode of operation. The pin can serve as an input trigger-pulse pin, as an input trigger-level (enable) pin and as an output interrupt pin. Note that the TRIG/INTZ pin can only execute one function at a time (either trigger-edge, trigger-level (enable) or interrupt function), therefore if a particular function is selected (for example, TRIG/INTZ configured as input trigger-edge), then the other functionality will not be available (for example, interrupt).

#### *8.3.16.1 Trigger-Pulse Functionality*

The trigger-pulse functionality allows for an external processor to initiate the process (either waveform sequencer, RTP, diagnostics or calibration) by pulsing the TRIG/INTZ pin. The process will initiate and play until it is done, at which point it will go back into standby mode to preserve power. If a stop trigger (another trigger-pulse) is received before the routine has finished, the routine will stop and the device will go back into standby state. In the case of diagnostics mode, a stop trigger causes the diagnostic routine to abort and no result will be reported. In the case of automatic level calibration routine, a stop trigger will cause the calibration to abort unless the AUTO_CAL_TIME[2:0] is set to trigger control, in which case the stop trigger is required for the calibration to complete, and the calibration will graciously finish and provide the expected output. Also note that a stop trigger can also be achieved by writing 0 to the GO bit.

The minimum pulse width duration is 1 µs.

## Feature Description (continued)

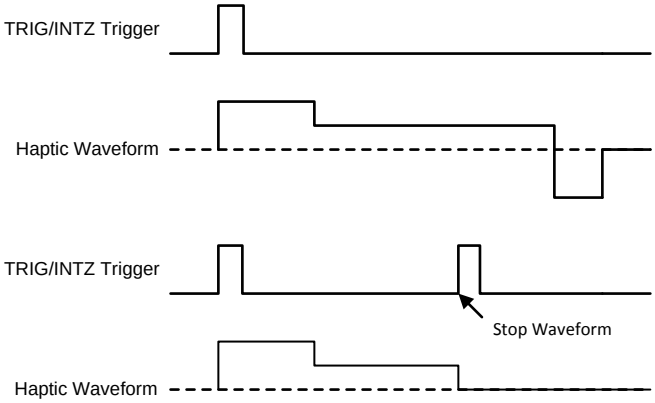



**Figure 19. TRIG/INTZ Functionality in Trigger-Pulse Mode**

### *8.3.16.2 Trigger-Level (Enable) Functionality*

The trigger-level (enable) functionality allows for an external micro-controller to wake up the DRV2624 by asserting the TRIG/INTZ pin (high), which immediately starts playing the process (either waveform sequencer, RTP value, diagnostics or auto-calibration). Once the TRIG/INTZ pin de-asserts (low) the device goes back to standby state to preserve power. If braking is desired before going into standby state, the AUTO_BRK_INTO_STBY bit can be set to allow automatic braking. Note that automatic braking is ignored during calibration.

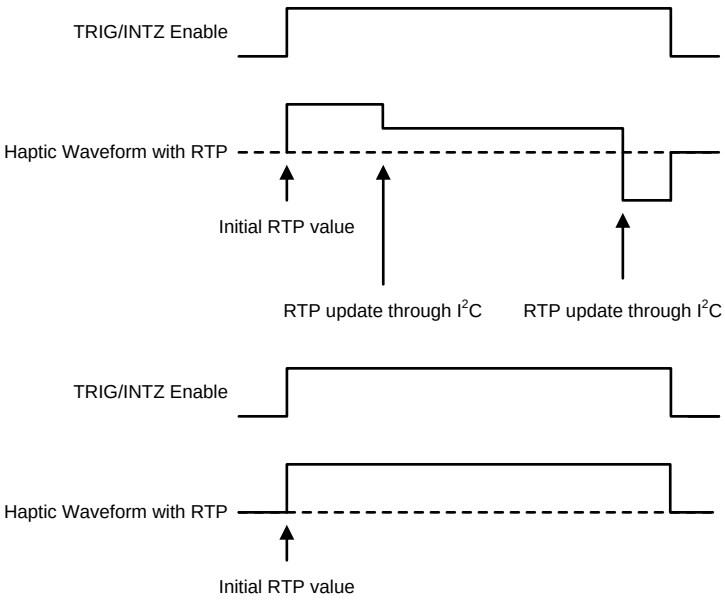



**Figure 20. TRIG/INTZ Functionality in Trigger-Level (Enable) Mode**

### *8.3.16.3 Interrupt Functionality*

The interrupt functionality allows for the DRV2624 to communicate to an external processor that a particular condition has occurred. When configured as an interrupt, the TRIG/INTZ pin becomes an output in open-drain configuration. An external pull-up is required for this mode. When asserted, the TRIG/INTZ pin will pull down the node until the interrupt is cleared (which is done by a reading the status register). All interrupts are maskable. A description of the supported interrupts is as follows:

OC_DETECT is flagged if an over-current event happens in the output stage during a process execution (such as waveform playback or auto-calibration).

## Feature Description (continued)

OVER_TEMP is flagged if the junction temperature goes above the thermal threshold during a process execution (such as waveform playback, diagnostics or auto-calibration).

UVLO is flagged if VDD drops below the VDD_THRES voltage during a process execution (such as waveform playback, diagnostics or auto-calibration).

PROCESS_DONE is flagged when the process (waveform sequencer, diagnostics or calibration) finishes. The PROCESS_DONE bit does not assert if the process is interrupted (such as with a stop trigger or by a critical condition). Note that RTP will never cause the PROCESS_DONE to assert because RTP never finishes on its own.

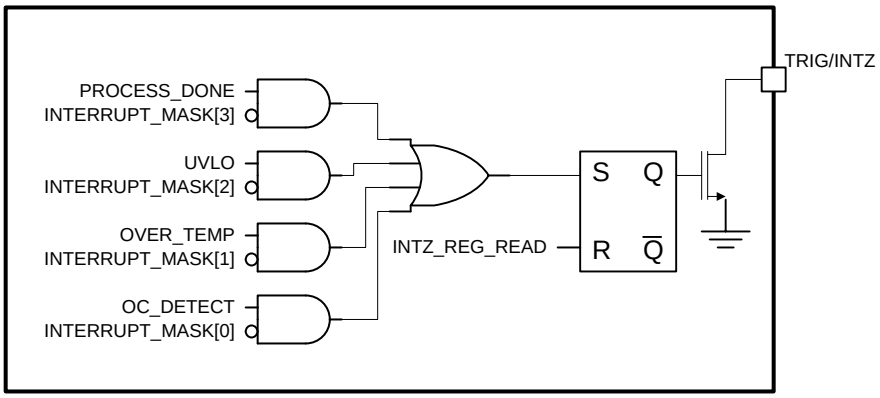



**Figure 21. TRIG/INTZ Functionality in Interrupt Mode**

Critical conditions, such as UVLO, over-temperature or over-current will not be monitored while the device is in standby state. However, UVLO and over-temperature conditions will be monitored during $I^2C$ communication is ongoing, even if the device is in standby state.

### 8.3.17 Automatic Transition to Standby State

The DRV2624 allows for automatic transition to standby state to preserve power. If the device goes into standby and a new waveform is triggered, the DRV2624 will wake-up and immediately play the requested waveform.

### 8.3.18 Automatic Brake into Standby

The DRV2624 allows for automatic braking before going into standby. If the AUTO_BRK_INTO_STBY is asserted, the device will brake the actuator (if necessary) before going into standby. This functionality will be bypassed in the event of a critical condition, such as over-temperature, over-current, UVLO, and NRST assertion.

### 8.3.19 Battery Monitoring and Power Preservation

The DRV2624 device continuously monitors the VDD voltage. In the event of a VDD voltage glitch that goes below the UVLO_THRES[2:0] voltage, the DRV2624 immediately stops any playback and goes into standby state. The UVLO status bit will assert and, if configured, the TRIG/INTZ pin will be asserted. Note that going into standby due to a VDD glitch will bypass any braking, even if AUTO_BRK_INTO_STBY is enabled. $I^2C$ communication will not be interrupted if a UVLO condition happens. However, because a UVLO condition could potentially corrupt such communication, TI recommends checking the UVLO flag after $I^2C$ transactions as a way to verify that the content was not corrupted in the process.

The DRV2624 also features a battery preservation mode that monitors the battery, and if VDD voltage drops below a specified threshold (see BAT_LIFE_EXT_LVL1[7:0] and BAT_LIFE_EXT_LVL2[7:0] parameters) will automatically clamp the maximum output voltage, as specified by the user (see OD_CLAMP_LVL1[7:0] and OD_CLAMP_LVL2[7:0] parameters).

## 8.4 Device Functional Modes

### 8.4.1 Power States

The DRV2624 device has multiple power states to optimize power consumption. In the event of a critical condition, the DRV2624 device goes immediately into the standby state. Figure 22 shows the transitions into and out of each state.

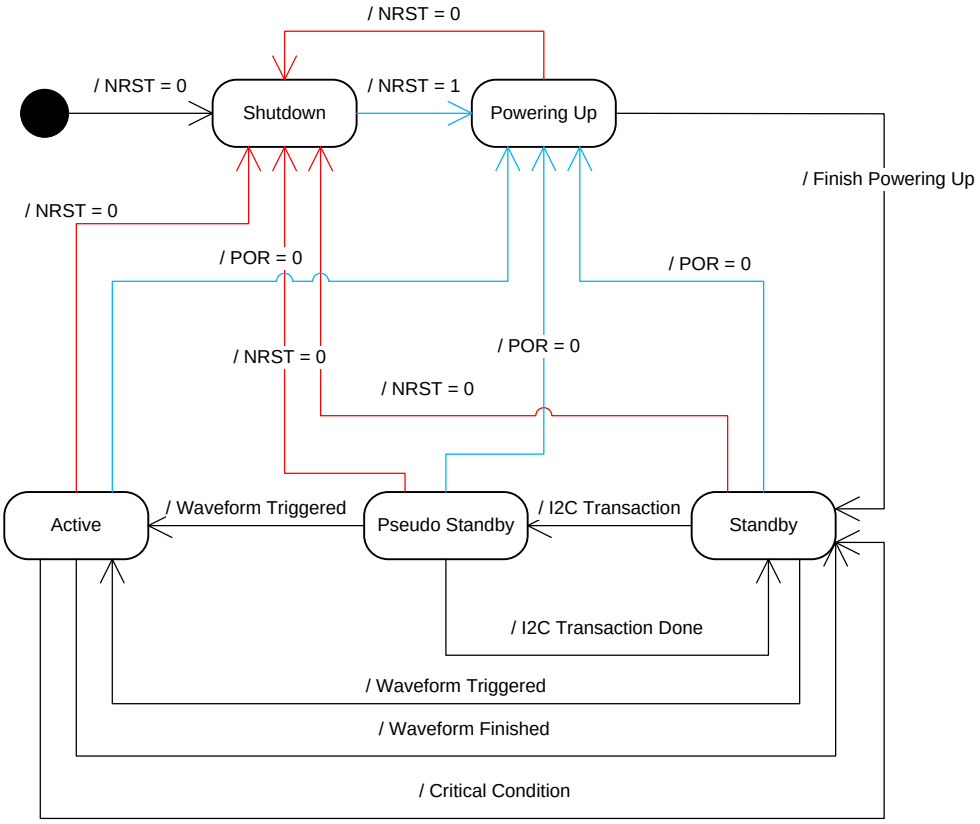



**Figure 22. Power State Diagram**

### 8.4.2 Operation With $V_{DD}$ < 2.5 V (Minimum $V_{DD}$)

Operating the device with a $V_{DD}$ value below 2.5 V is not recommended.

### 8.4.3 Operation With $V_{DD}$ > 6 V (Absolute Maximum $V_{DD}$)

The DRV2624 device is designed to operate at up to 5.5 V with an absolute maximum voltage of 6 V . If exposed to voltages above 6 V, the device can suffer permanent damage.

### 8.4.4 Operation in Shutdown State

The NRST pin of the DRV2624 device gates the power-up of the device. When NRST is asserted (logic low), all internal blocks of the device (including $I^2C$ controller) are off to achieve ultra low power.

When the NRST pin is deasserted (logic high), the DRV2624 device powers-up, loads all the default conditions and goes into standby state to preserve power.

Asserting the NRST pin has an immediate effect. Any process being executed will be aborted immediately and the device will go into shutdown state.

The DRV2624 device allows for the NRST to be permanently tied directly to VDD, in which case the shutdown state will be bypassed.

## Device Functional Modes (continued)

### 8.4.5 Operation in STANDBY State

The DRV2624 is optimized for power preservation, therefore it will automatically go into standby when not in use.

In standby state, $I^2C$ communication is available and register content is preserved.

Stand-by state turns-off all non-essential blocks to preserve power, but features a fast turn on time that will allows for low latency haptic playback from this mode.

If the host controller wants to force the DRV2624 device into standby, the host controller can do so by sending a stop trigger, which can be done by writing a 0 to the GO bit, or, if using an external trigger, by following the appropriate mechanism as described in Process Trigger.

The DRV2624 features an automatic braking option that will make the drive brake the actuator before going into standby state. The automatic braking feature will be executed every time the device goes into standby state, with the exception of a critical condition (such as over-current, thermal shutdown or UVLO). The automatic braking feature can be enabled or disabled by using the AUTO_BRK_INTO_STBY bit.

### 8.4.6 Operation in ACTIVE State

The DRV2624 goes into active mode only when it must run a process (either waveform playback, calibration or diagnostics).

When the device gets into active mode, the first thing done is to check for critical conditions (such as over-current, thermal shutdown or UVLO). If a critical condition is present, the DRV2624 device sets the appropriate flag (and fires an interrupt if configured to do so), and immediately goes into standby. If no critical condition is present, the DRV2624 device runs the routine and when finished returns to the standby state.

If a critical condition happens during a routine execution, the routine is aborted immediately and the device sets the appropriate flag (and fires an interrupt if configured to do so), and immediately goes into standby.

If a stop trigger is received while a routine is being executed, the routine will be stopped and the device will go into standby state. If the automatic braking option is enabled (AUTO_BRK_INTO_STBY bit), then the DRV2624 device will brake the actuator before going into standby state.

### 8.4.7 Changing Modes of Operation

The DRV2624 device has 8 parameters that control different aspects of modes of operation, namely: LRA_ERM, MODE[1:0], CONTROL_LOOP, TRIG_PIN_FUNC, AUTO_BRK_INTO_STBY, AUTO_BRK_OL, and LRA_WAVE_SHAPE. If any of these parameters are changed in the middle of a process execution (waveform playback, diagnostics or calibration), the DRV2624 will interpret the change as an abort and will go into standby.

LRA_ERM parameter selects the actuator type.

MODE[1:0] selects between the 4 available process that can be run, namely RTP and Waveform Sequencer for waveform playback, diagnostics and calibration. The DRV2624 device will be in standby state until a trigger is received. At that point the device will execute the process selected in the MODE[1:0] parameter. Once finished, the DRV2624 device will return into standby state.

CONTROL_LOOP selects between open loop and closed loop.

TRIG_PIN_FUNC parameter selects the functionality of the TRIG/INTZ pin among the 3 possibilities: pulse trigger, level trigger and interrupt.

AUTO_BRK_INTO_STBY parameter enables automatic braking when going into standby. The DRV2624 device will monitor the back-EMF of the actuator before going into standby, and if the back-EMF is moving, the actuator will brake and then go into standby. Note that for this function to be executed in open loop, the AUTO_BRK_OL pin must be enabled.

AUTO_BRK_OL parameter enables automatic braking for open loop mode. If enabled, every time a waveform is configured with a negative amplitude, the DRV2624 device will brake the actuator. This feature assumes that the actuator has been calibrated and works under closed-loop conditions.

LRA_WAVE_SHAPE parameter (available for LRA only) allows for selecting the waveform shape to be used when driving the LRA in open loop. In closed-loop this parameter will be ignored.

## 8.5 Operation During Exceptional Conditions

This section lists different exceptional conditions and the ways that the DRV2624 device operates during these conditions. This section also describes how the device goes into and out of these states.

### 8.5.1 Operation With No Actuator Attached

In open loop mode, the DRV2624 device will drive the waveform as intended by the user.

In LRA closed-loop mode, if a waveform is played without an actuator connected to the OUT+ and OUT– pins, the output pins toggle. However, the toggling frequency is not predictable.

In ERM closed-loop mode, the output pins will attempt to drive, however the amplitude is not predictable.

### 8.5.2 Operation With a Non-Moving Actuator Attached

In the ERM case, the DRV2624 device will attempt to overdrive the actuator until movement is detected.

The model of a non-moving actuator can be simplified as a resistor. If a resistor (with similar loading as an LRA, such as 25 Ω) is connected across the OUT+ and OUT– pins, and the DRV2624 device is in LRA closed-loop mode, the output pins toggle at a default f. In LRA open-loop mode the output pins toggle at the specified open-loop frequency.

### 8.5.3 Operation With a Short at REG Pin

If the REG pin is shorted to GND, the device turns off. When the short is removed, the device starts in the default condition.

### 8.5.4 Operation With a Short at OUT+, OUT–, or Both

During playback, if any of the output pins (OUT+ or OUT–) is shorted to $V_{DD}$, GND, or to each other, a current-protection circuit automatically enables to shut-down the output stage, the OC_DETECT bit is asserted (and an interrupt is fired if enabled) and the DRV2624 device will go into standby state.

The DRV2624 device only checks for shorts when running a process (either RTP, waveform sequencer, diagnostics or calibration). If the short occurs when the device is idle, the short is not detected until the device attempts to run a process.

## 8.6 Programming

### 8.6.1 Auto-Resonance Engine Programming for the LRA

#### *8.6.1.1 Drive-Time Programming*

The resonance frequency of each LRA actuator varies based on many factors and is generally dominated by mechanical properties. The auto-resonance engine-tracking system is optimized by providing information about the resonance frequency of the actuator. The DRIVE_TIME[4:0] bit is used as an initial guess for the half-period of the LRA. The drive time is automatically and quickly adjusted for optimum drive. For example, if the LRA has a resonance frequency of 200 Hz, then the drive time should be set to 2.5 ms.

For ERM actuators, the DRIVE_TIME[4:0] bit controls the rate for back-EMF sampling. Lower drive times imply higher back-EMF sampling frequencies which cause higher peak-to-average ratios in the output signal, and requires more supply headroom. Higher drive times imply lower back-EMF sampling frequencies which cause the feedback to react at a slower rate.

#### *8.6.1.2 Current-Dissipation Time Programming*

To sense the back-EMF of the actuator, the DRV2624 device goes into high impedance mode. However, before the device enters this mode, the device must dissipate the current in the actuator. The DRV2624 device controls the time allocated for dissipation-current through the IDISS_TIME[3:0] parameter.

#### *8.6.1.3 Blanking Time Programming*

After the current in the actuator dissipates, the DRV2624 device waits for a blanking time of the signal to settle before the back-EMF analog-to-digital (AD) conversion converts. The BLANKING_TIME[3:0] parameter controls this time.

#### *8.6.1.4 Zero-Crossing Detect-Time Programming*

When the blanking time expires, the back-EMF AD monitors for zero crossings. The ZC_DET_TIME[1:0] parameter controls the minimum time allowed for detecting zero crossings.

### 8.6.2 Automatic-Level Calibration Programming

#### *8.6.2.1 Rated Voltage Programming*

The rated voltage is the driving voltage that the driver will output during steady state. However, in closed-loop drive mode, temporarily having an output voltage that is higher than the rated voltage is possible.

The RATED_VOLTAGE[7:0] parameter sets the rated voltage for the closed-loop drive modes.

$$V_{(ERM\text{-}CL\_AV)} = 21.88 \times 10^{-3}\,\text{RATED\_VOLTAGE[7:0]} \quad (5)$$

$$V_{(LRA\text{-}CL\_RMS)} = \frac{20.58 \times 10^{-3} \times \text{RATED\_VOLTAGE[7:0]}}{\sqrt{1 - (4 \times t_{(SAMPLE\_TIME)} + 300 \times 10^{-6}) \times f_{(LRA)}}} \quad (6)$$

In open-loop mode, the RATED_VOLTAGE[7:0] parameter is ignored. Instead, the OD_CLAMP[7:0] parameter is used to set the full-scale voltage for the open-loop drive modes.

$$V_{(ERM\text{-}OL\_AV)} = 21.59 \times 10^{-3}\,\text{OD\_CLAMP[7:0]} \quad (7)$$

$$V_{(LRA\text{-}OL\_RMS)} = 21.32 \times 10^{-3} \times \text{OD\_CLAMP[7:0]} \times \sqrt{1 - f_{(LRA)} \times 800 \times 10^{-6}} \quad (8)$$

The auto-calibration routine uses the RATED_VOLTAGE[7:0] and OD_CLAMP[7:0] bits as inputs and therefore these registers must be written before calibration is performed. Any modification of this register value should be followed by calibration to appropriately set A_CAL_BEMF[7:0].

## Programming (continued)

#### 8.6.2.2 Overdrive Voltage-Clamp Programming

During closed-loop operation, the actuator feedback allows the output voltage to go above the rated voltage during the automatic overdrive and automatic braking periods. The OD_CLAMP[7:0] parameter sets a clamp so that the automatic overdrive is bounded. The OD_CLAMP[7:0] parameter also serves as the full-scale reference voltage for open-loop operation. The OD_CLAMP[7:0] parameter always represents the maximum *peak voltage* that is allowed, regardless of the mode.

**NOTE**

If the supply voltage ($V_{DD}$) is less than the overdrive clamp voltage, the output driver is unable to reach the clamp voltage value because the output voltage cannot exceed the supply voltage. If the rated voltage exceeds the overdrive clamp voltage, the overdrive clamp voltage has priority over the rated voltage.

$$V_{(ERM\_clamp)} = \frac{21.64 \times 10^{-3} \times OD\_CLAMP[7:0] \times (t_{(DRIVE\_TIME)} - 300 \times 10^{-6})}{t_{(DRIVE\_TIME)} + t_{(IDISS\_TIME)} + t_{(BLANKING\_TIME)}} \quad (9)$$

$$V_{(LRA\_clamp)} = 21.22 \times 10^{-3} \times OD\_CLAMP[7:0] \quad (10)$$

### 8.6.3 I²C Interface

#### 8.6.3.1 TI Haptic Broadcast Mode

The DRV2624 device has a TI haptic broadcast mode where, if enabled using the I²C_BCAST_EN bit, will make the device respond to the slave address 0x58 (7-bit) or 1011000 in binary. This mode is useful in the event that multiple haptic drivers implementing the TI haptic broadcast mode as installed in the system. In such a scenario, writing the GO bit to the 0x58 slave address will cause all haptic drivers to trigger the process at the same time.

#### 8.6.3.2 I²C Communication Availability

The I²C protocol is available for read/write operations during Standby, and Active states.

#### 8.6.3.3 General I²C Operation

The I²C bus employs two signals, SDA (data) and SCL (clock), to communicate between integrated circuits in a system. The bus transfers data serially, one bit at a time. The 8-bit address and data bytes are transferred with the most-significant bit (MSB) first. In addition, each byte transferred on the bus is acknowledged by the receiving device with an acknowledge bit. Each transfer operation begins with the master device driving a start condition on the bus and ends with the master device driving a stop condition on the bus. The bus uses transitions on the data pin (SDA) while the clock is at logic high to indicate start and stop conditions. A high-to-low transition on the SDA signal indicates a start, and a low-to-high transition indicates a stop. Normal data-bit transitions must occur within the low time of the clock period. Figure 23 shows a typical sequence. The master device generates the 7-bit slave address and the read-write (R/W) bit to start communication with a slave device. The master device then waits for an acknowledge condition. The slave device holds the SDA signal low during the acknowledge clock period to indicate acknowledgment. When the acknowledgment occurs, the master transmits the next byte of the sequence. Each device is addressed by a unique 7-bit slave address plus a R/W bit (1 byte). All compatible devices share the same signals through a bidirectional bus using a wired-AND connection.

The number of bytes that can be transmitted between start and stop conditions is not limited. When the last word transfers, the master generates a stop condition to release the bus. Figure 23 shows a generic data-transfer sequence.

Use external pull-up resistors for the SDA and SCL signals to set the logic-high level for the bus. Pull-up resistors between 660 Ω and 4.7 kΩ are recommended. Do not allow the SDA and SCL voltages to exceed the DRV2624 supply voltage, $V_{DD}$.

**NOTE**

The DRV2624 slave address is 0x5A (7-bit), or 1011010 in binary.

## Programming (continued)

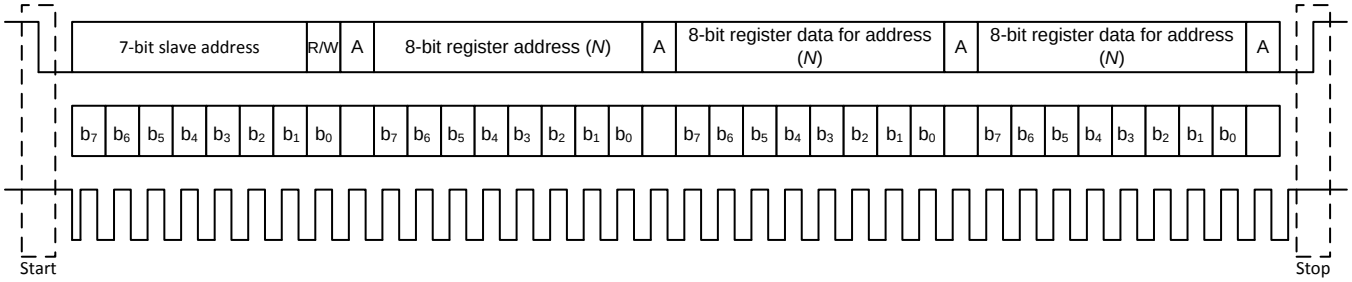



**Figure 23. Typical I²C Sequence**

The DRV2624 device operates as an I²C-slave 1.8-V logic thresholds, but can operate up to the $V_{DD}$ voltage. The device address is 0x5A (7-bit), or 1011010 in binary which is equivalent to 0xB4 (8-bit) for writing and 0xB5 (8-bit) for reading.

### *8.6.3.4 Single-Byte and Multiple-Byte Transfers*

The serial control interface supports both single-byte and multiple-byte R/W operations for all registers.

During multiple-byte read operations, the DRV2624 device responds with data one byte at a time and beginning at the signed register. The device responds as long as the master device continues to respond with acknowledges.

The DRV2624 supports sequential I²C addressing. For write transactions, a sequential I²C write transaction has taken place if a register is issued followed by data for that register as well as the remaining registers that follow. For I²C sequential-write transactions, the register issued then serves as the starting point and the amount of data transmitted subsequently before a stop or start is transmitted determines how many registers are written.

### *8.6.3.5 Single-Byte Write*

As shown in Figure 24, a single-byte data-write transfer begins with the master device transmitting a start condition followed by the I²C device address and the read-write bit. The read-write bit determines the direction of the data transfer. For a write-data transfer, the read-write bit must be set to 0. After receiving the correct I²C device address and the read-write bit, the DRV2624 responds with an acknowledge bit. Next, the master transmits the register byte corresponding to the DRV2624 internal-memory address that is accessed. After receiving the register byte, the device responds again with an acknowledge bit. Finally, the master device transmits a stop condition to complete the single-byte data-write transfer.

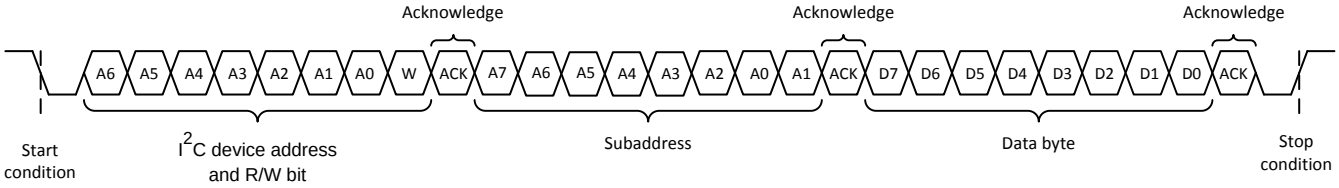



**Figure 24. Single-Byte Write Transfer**

## Programming (continued)

### 8.6.3.6 Multiple-Byte Write and Incremental Multiple-Byte Write

A multiple-byte data write transfer is identical to a single-byte data write transfer except that multiple data bytes are transmitted by the master device to the DRV2624 device as shown in Figure 25. After receiving each data byte, the DRV2624 device responds with an acknowledge bit.

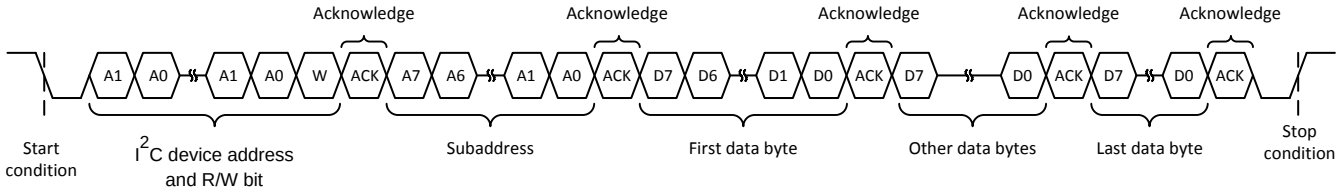



**Figure 25. Multiple-Byte Write Transfer**

### 8.6.3.7 Single-Byte Read

Figure 26 shows that a single-byte data-read transfer begins with the master device transmitting a start condition followed by the I²C device address and the read-write bit. For the data-read transfer, both a write followed by a read actually occur. Initially, a write occurs to transfer the address byte of the internal memory address to be read. As a result, the read-write bit is set to 0.

After receiving the DRV2624 address and the read-write bit, the DRV2624 device responds with an acknowledge bit. The master then sends the internal memory address byte, after which the device issues an acknowledge bit. The master device transmits another start condition followed by the DRV2624 address and the read-write bit again. On this occasion, the read-write bit is set to 1, indicating a read transfer. Next, the DRV2624 device transmits the data byte from the memory address that is read. After receiving the data byte, the master device transmits a not-acknowledge followed by a stop condition to complete the single-byte data read transfer. See the note in the *General I²C Operation* section.

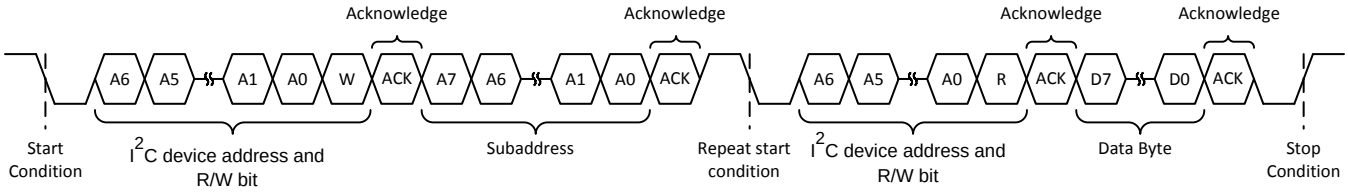



**Figure 26. Single-Byte Read Transfer**

### 8.6.3.8 Multiple-Byte Read

A multiple-byte data-read transfer is identical to a single-byte data-read transfer except that multiple data bytes are transmitted by the DRV2624 device to the master device as shown in Figure 27. With the exception of the last data byte, the master device responds with an acknowledge bit after receiving each data byte.

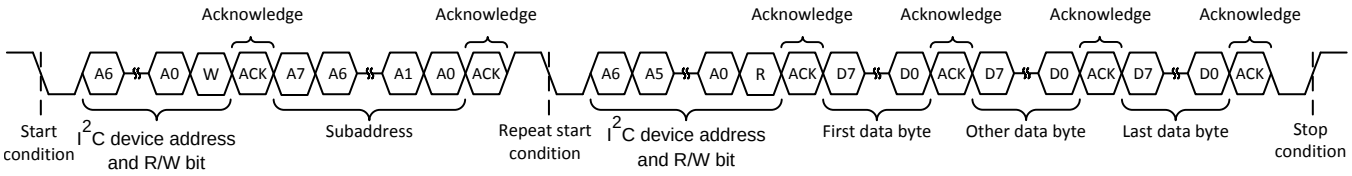



**Figure 27. Multiple-Byte Read Transfer**

## Programming (continued)

### 8.6.4 Programming for Open-Loop Operation

The DRV2624 device can be used in open-loop mode and closed-loop mode. If open-loop operation is desired, the first step is to determine which actuator type is to use, either ERM or LRA.

#### *8.6.4.1 Programming for ERM Open-Loop Operation*

To configure the DRV2624 device in ERM open-loop operation, the ERM must be selected by writing the LRA_ERM bit to 1, and the CONTROL_LOOP bit to 1.

#### *8.6.4.2 Programming for LRA Open-Loop Operation*

To configure the DRV2624 device in LRA open-loop operation, the LRA must be selected by writing the LRA_ERM bit to 0, and the CONTROL_LOOP bit to 1. Additionally, the OL_LRA_PERIOD parameter must be configured with the appropriate LRA frequency.

### 8.6.5 Programming for Closed-Loop Operation

For closed-loop operation, the device must be calibrated according to the actuator selection. When calibrated accordingly, the user is only required to provide the desired waveform. The DRV2624 device automatically adjusts the level and, for the LRA, automatically adjusts the driving frequency.

### 8.6.6 Diagnostics Routine

The DRV2624 has a diagnostic routine that can be selected by the MODE[1:0] parameter. The purpose of the routine is to determine if the actuator can be safely and correctly driven. If a problem is detected by the diagnostic routine, the DIAG_RESULT bit will assert (high). After running the diagnostic routine, the DIAG_RESULT should be checked to assess the result of the diagnostic routine. If the diagnostic routine does not finish due to a critical condition, such as a UVLO, over temperature or over-current condition, the diagnostic routine will be aborted and the DIAG_RESULT will be set to 1.

The diagnostic routine is composed of 2 sub-routines: a resistance measurement routine and a functional routine.

The resistance measurement sub-routine reports the resistance of the actuator as seen from the differential output pins (OUT+ and OUT-) and placed on the DIAG_Z_RESULT parameter. The resistance measurement sub-routine should always be executed during the diagnostics routine and the output is reported in the DIAG_Z_RESULT. The only exception is during an over-temperature or UVLO condition, in which case the diagnostic routine will abort immediately and the device will go into standby state.

**NOTE**

An over-current condition will never happen in this sub-routine, even in the presence of a short, because the resistance measurement injects a small current that will not be detected by the over-current detection circuit. The resistance measurement sub-routine is the first to be executed. Also, this sub-routine will not cause the DIAG_RESULT bit to assert.

After the resistance measurement, the diagnostic routine plays a diagnostic waveform to determine whether the actuator can be successfully driven. A short or open condition, as well as failure to detect a valid BEMF will cause the DIAG_RESULT bit to assert. Note that if a critical problem is experienced during the diagnostic routine, such as an over-current condition, the routine can be aborted, and the DIAG_RESULT will assert.

### 8.6.7 Calibration Routine

The DRV2624 has a calibration routine that automatically populates all critical parameters required for successfully driving a specific actuator (the one connected and being calibrated) in closed-loop. Variation occurs between different actuators even if the actuators are of the same model. To ensure optimal results, TI recommends that the calibration routine be run at least once for each actuator.

## Programming (continued)

The calibration engine requires a number of parameters as inputs before the calibration can be executed. When the inputs are configured, the calibration routine can be executed. After calibration execution occurs, the output parameters are written over the specified register locations. Figure 28 shows all of the required inputs and generated outputs. To ensure proper auto-resonance operation, the LRA actuator type requires more input parameters than the ERM. The LRA parameters are ignored when the device is in ERM mode.

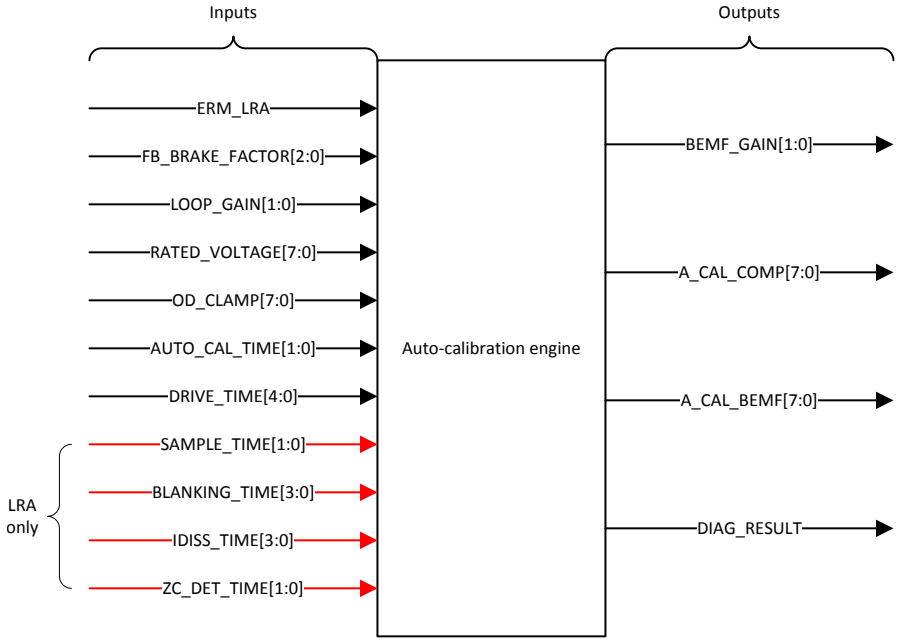



**Figure 28. Calibration-Engine Functional Diagram**

For proper calibration results, the calibration waveform must be executed long enough to achieve a steady acceleration. Therefore, the DRV2624 device has a configurable amount of time for the calibration waveform, which can be selected by the AUTO_CAL_TIME[1:0] parameter. Additionally, the option to control the calibration time by using a trigger is provided to accommodate for the cases that require longer times than those allowed by the AUTO_CAL_TIME parameter. Under the triggered control option, the calibration will start executing after the initial trigger, and then will stop execution once a stop trigger is received. At that point the output values of the calibration will be populated. Note that a minimum duration is required for the calibration to work properly.

**Table 1. Calibration Routine Behavior Under Different AUTO_CAL_TIME Selections**

| AUTO_CAL_TIME[1:0] | ACTION | COMMENTS |
|---|---|---|
| 0 | 250 ms calibration waveform | |
| 1 | 500 ms calibration waveform | |
| 2 | 1 s calibration waveform | |
| 3 | Trigger controlled | Can be triggered either using the GO bit or externally. To use the external trigger, the TRIG_PIN_FUNC parameter must be configured appropriately.<br>In this case the minimum duration should be 1 s, otherwise the result of the calibration can be corrupted. |

The following instructions list the step-by-step register configuration for auto-calibration.

1. Apply a valid supply voltage to the DRV2624 device, and then pull the NRST pin high. The supply voltage should allow for adequate drive voltage of the selected actuator.
2. Write a value of 0x03 to the MODE parameter to set the auto-calibration routine.
3. Populate the input parameters required by the auto-calibration engine:
   (a) LRA_ERM — selection will depend on desired actuator.
   (b) FB_BRAKE_FACTOR[2:0] — A value of 3 is valid for most actuators.
   (c) LOOP_GAIN[1:0] — A value of 2 is valid for most actuators.
   (d) RATED_VOLTAGE[7:0] — See the *Rated Voltage Programming* section for calculating the correct register value.
   (e) OD_CLAMP[7:0] — See the *Overdrive Voltage-Clamp Programming* section for calculating the correct register value.
   (f) AUTO_CAL_TIME[1:0] — A value of 3 is valid for most actuators.
   (g) DRIVE_TIME[3:0] — See the *Drive-Time Programming* for calculating the correct register value.
   (h) SAMPLE_TIME[1:0] — A value of 3 is valid for most actuators.
   (i) BLANKING_TIME[3:0] — A value of 1 is valid for most actuators.
   (j) IDISS_TIME[3:0] — A value of 1 is valid for most actuators.
   (k) ZC_DET_TIME[1:0] — A value of 0 is valid for most actuators.
4. Write a 1 to the GO bit to start the auto-calibration process. When auto calibration is complete, the GO bit automatically clears. The auto-calibration results are written in the respective registers as shown in Figure 28.
5. Check the status of the DIAG_RESULT bit to ensure that the auto-calibration routine is complete without faults.
6. Evaluate system performance with the auto-calibrated settings. Note that the evaluation should occur during the final assembly of the device because the auto-calibration process can affect actuator performance and behavior. If any adjustment is required, the inputs can be modified and this sequence can be repeated. If the performance is satisfactory, the user can do any of the following:
   (a) Repeat the calibration process upon subsequent power ups.
   (b) Store the auto-calibration results in host processor memory and rewrite them to the DRV2624 device upon subsequent power ups. The device retains these settings when in STANDBY mode or when the EN pin is low.

### 8.6.8 Waveform Playback Programming

#### *8.6.8.1 Data Formats for Waveform Playback*

The DRV2624 uses a signed data format (2's complement) to specify the magnitude and direction of the drive. The actuator can be driven in either closed-loop or open-loop. In closed-loop, positive numbers indicate the magnitude of the drive desired. Negative numbers are interpreted as a brake signal, which is automatic in closed-loop. In open-loop, positive and negative numbers are required to specify amplitude magnitude for both driving and braking. In the case that automatic braking is selected for open loop, then any negative number will be interpreted as a brake signal.

#### *8.6.8.2 Open-Loop Mode*

In open-loop mode, the reference level for full-scale drive is set by the OD_CLAMP[7:0] parameter. A mid-scale input value gives no drive signal, and a less-than mid-scale gives a negative drive value. For an ERM, a negative drive value results in counter-rotation, or braking. For an LRA, a negative drive value results in a 180-degree phase shift in commutation.

#### *8.6.8.3 Closed-Loop Mode*

In closed-loop mode, the DRV2624 device provides automatic overdrive and braking for both ERM and LRA devices. Positive values indicate that acceleration is desired. Negative values and 0 indicate that braking is desired.

The reference level for steady-state full-scale drive is set by the RATED_VOLTAGE[7:0] bit (when auto-calibration is performed). The output voltage can momentarily exceed the rated voltage for automatic overdrive and braking, but does not exceed the OD_CLAMP[7:0] voltage. Braking occurs automatically based on the input signal when the back-EMF feedback determines that braking is necessary.

In the event that the user is concerned that the overdrive time may be too high for a particular actuator, the OD_CLAMP_TIME[1:0] can be used to limit the amount of time spent in over-drive mode (a voltage above the rated voltage). If the overdrive time is exceeded and the DRV2624 device is still attempting to overdrive the actuator, a new clamp is enforced, which is specified in the RATED_VOLTAGE_CLAMP[7:0] parameter, which is enforced until a brake signal is received. During braking, the device will be allowed to overdrive for the time specified in the OD_CLAMP_TIME[1:0], and, if exceeded, the RATED_VOLTAGE_CLAMP[7:0] is enforced. This feature ensures that the actuator will not be overdriven continuously for longer than desired.

### 8.6.9 Waveform Setup and Playback

Playback of a haptic effect can occur in RTP mode or by using the waveform sequencer. And the process (either RTP or waveform sequencer) can be triggered by writing a 1 to the GO bit, or by using the external trigger in either trigger-pulse or trigger-level configuration. A waveform can be terminated prematurely by writing a 0 to the GO bit or by sending a stop trigger via the external TRIG/INTZ pin.

#### *8.6.9.1 Waveform Playback Using RTP Mode*

The user can enter the RTP mode by writing to the MODE[1:0] parameter. In RTP mode, when the DRV2624 device received a trigger, the device drives the actuator continuously with the amplitude specified in the RTP_INPUT[7:0] parameter. Because the amplitude tracks the value specified in the RTP_INPUT[7:0] parameter, the $I^2C$ bus can stream waveforms. To stop driving the user can either change modes or send a stop trigger (either write 0 to the GO bit or using the external trigger).

#### *8.6.9.2 Loading Data to RAM*

The DRV2624 device contains 1 kB of integrated RAM to store customer waveforms. The waveforms are represented as amplitude-time pairs. Using the waveform sequencer, the waveforms can be recalled, sequenced, and played through the $I^2C$ or an external GPIO trigger.

A library consists of a revision byte (should be set to 0), a header section, and the waveform data content. The library header defines the data boundaries for each effect ID in the data field, and the waveform data contains a sequence of amplitude-time pairs that define the effects.

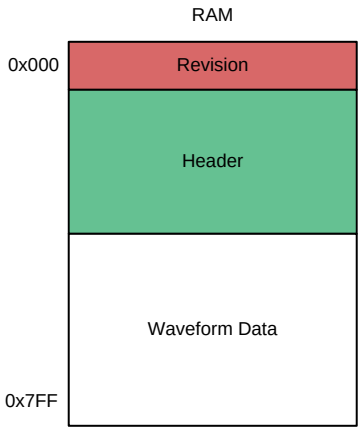



**Figure 29. RAM Memory Structure**

##### 8.6.9.2.1 Header Format

The header block consist of *N*-boundary definition blocks of 3 bytes each. *N* is the number of effects stored in the RAM (*N* cannot exceed 127). Each of the boundary definition blocks contain the start address (2 bytes) and a configuration byte.

The start address contains the location in the memory where the waveform data associated with this effect begins. The position of the effect pointer in the header becomes the effect ID. The first effect boundary definition points to the ID for effect 1, the second definition points to the ID for effect 2, and so on. The resulting effect ID is the effect ID that is used in the waveform sequencer.

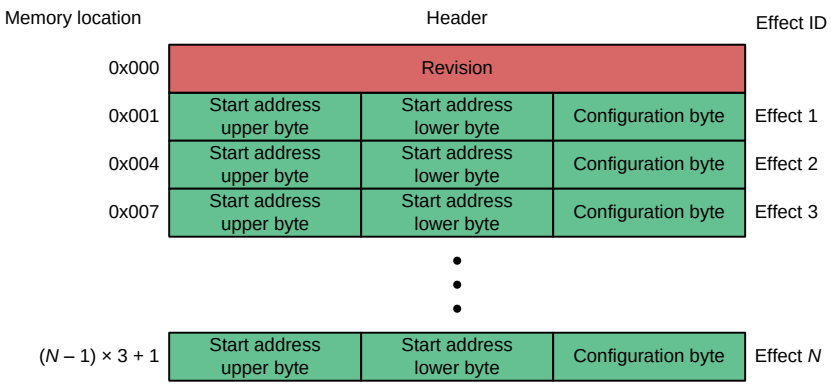



**Figure 30. Header Structure**

The configuration byte contains the following two parameters:

- The effect size contains the amount of bytes that define the waveform data. An effect size of 0 is an error state. Any odd-number effect size is an error state because the waveform data is defined as time-value (2 bytes). Therefore, the effect size must be an even number between 2 and 30.
- The WAVEFORM_REPEATS[2:0] bit is used to select the number of times the complete waveform is be played when it is called by the waveform sequencer. A value of 0 is no repeat and the waveform is played once. A value of 1 means 1 repeat and the waveform is played twice. A value of 7 means infinite repeat until the GO bit is cleared.

During waveform design, ensure that the appropriate amount of drive time is at zero amplitude on the end of the waveform so that the waveform stored in the RAM is repeated smoothly.

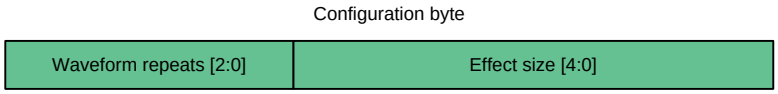



**Figure 31. Header Configuration Byte Structure**

#### 8.6.9.2.2 RAM Waveform Data Format

The library data contents can take two forms which are voltage-time pair and linear ramp. The voltage-time pair method implements a *set and wait* protocol, which is an efficient method of actuator control for most types of waveforms. The *set and wait* method becomes inefficient when ramping waveforms is desired, therefore a linear ramp method is also supported which linearly interpolates a set of voltages between two amplitude values. Both methods require only two bytes of data per set point. The linear ramp method uses a minimum of four bytes so that linear interpolation can be done to the next set point. The most significant bit of the voltage value is reserved to indicate the linear ramping mode.

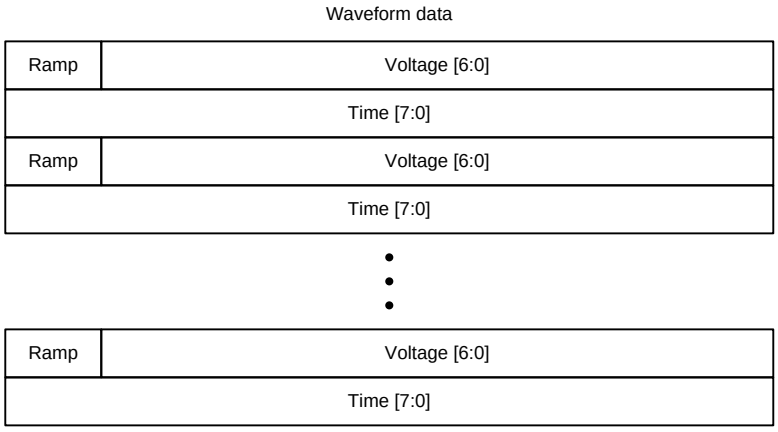



**Figure 32. Waveform Data Structure**

Data is stored as interleaved voltage-time pairs. *Voltage* in the voltage-time pair is a 7-bit signed number with range –63 to 63 when in signed data format, and a 7-bit unsigned number with a range of 0 to 127 when in unsigned data format. The MSB of the voltage byte is reserved for the linear ramping mode.

The *Time* value is the number of ticks that the Voltage will last. The size of the tick depends on the PLAYBACK_INTERVAL bit. If PLAYBACK_INTERVAL = 0 the absolute time is number of ticks × 5 ms. If PLAYBACK_INTERVAL = 1 the absolute time is number ticks × 1 ms.

When the most significant bit of the Voltage byte is high, the engine interprets a linear interpolation between that voltage and the following voltage point. The following voltage point can either be a part of a regular voltage-time pair, or a subsequent ramp. The following lists the sequence of bytes:

1. Byte1 — Voltage1 (MSB High)
2. Byte2 — Time1
3. Byte3 — Voltage2
4. Byte4 — Time2

The engine creates a linear interpolation between Voltage1 and Voltage2 over the time period Time1, where Time1 is a number of 5-ms ticks. The start value for the ramp is the 7-bit value contained in Voltage1. The end amplitude is the 7-bit value contained in Voltage2. The MSB in Voltage2 can indicate a following voltage-time pair or the starting point in a subsequent ramp.

#### *8.6.9.3 Waveform Sequencer*

To play haptic effects from the internal memory, the effects must first be loaded into the waveform sequencer, and then the effects can be launched by using any of the trigger options.

The waveform sequencer queues waveform-library identifiers for playback. Eight sequence registers queue up to eight library waveforms for sequential playback. A waveform identifier is an integer value referring to the index position of a waveform in the internal memory. Playback begins at WAV_FRM_SEQ1 when the user triggers the waveform sequencer. When playback of that waveform ends, the waveform sequencer plays the next waveform identifier held in WAV_FRM_SEQ2 (if non-zero). The waveform sequencer continues in this way until the sequencer reaches an identifier value of zero or until all eight identifiers are played whichever comes first.

The waveform identifier is a 7-bit number. The MSB of each sequence register can be used to implement a delay between sequence waveforms. When the MSB is high, bits 6-0 indicate the length of the wait time. The wait time for that step then becomes WAV_FRM_SEQ[6:0] × 10 ms.

The DRV2624 device allows for looping of individual waveforms by using the WAVn_SEQ_LOOP parameters. When used, the state machine will loop the particular waveform the number of times specified in the associated WAVn_SEQ_LOOP parameter before moving to the next waveform. Additionally, the entire sequencer of waveforms can be looped a number of times specified by the WAV_SEQ_MAIN_LOOP parameter. The waveform-looping feature is useful for long, custom haptic playbacks, such as a haptic ringtone.

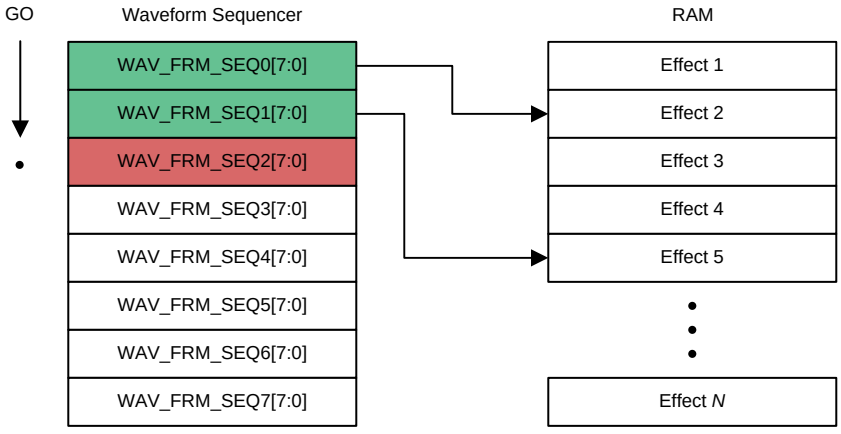



**Figure 33. Waveform Sequencer Programming**

### 8.6.9.4 Waveform Playback Triggers

The DRV2624 device has 2 modes of waveform playback: Waveform Sequencer and RTP. Both modes can be triggered externally by using the TRIG/INTZ pin or internally by using the GO bit. If using external trigger, the TRIG_PIN_FUNC most be selected appropriately.

#### 8.6.9.4.1 Playback Trigger Without Automatic Brake into Standby

When automatic braking into standby is disabled (AUTO_BRK_INTO_STBY bit set to 0), playback can be triggered in both RTP and waveform sequencer modes with internal trigger by writing a 1 to the GO bit, and stopped by writing a 0 to the GO bit. Playback can also be triggered by the external trigger by following the trigger-pulse or trigger-level specifications (see Figure 19 and Figure 20 for details). Note that internal trigger is not available if the external trigger pin is set to trigger-level (TRIG_PIN_FUNC = 1)

RTP playback (MODE[1:0] = 0), once triggered, will run indefinitely until the waveform is stopped. The waveform sequencer (MODE[1:0] = 1) will run until it reaches the end point and will automatically go into standby without a cancel trigger being received, unless an infinite loop is requested.

##### 8.6.9.4.1.1 Playback Trigger With Automatic Brake into Standby (SimpleDrive)

If automatic braking into standby is enabled (AUTO_BRK_INTO_STBY bit is set to 1), then the device part will go into brake mode before going into standby. This feature introduces new timing requirements that are described in below diagrams.

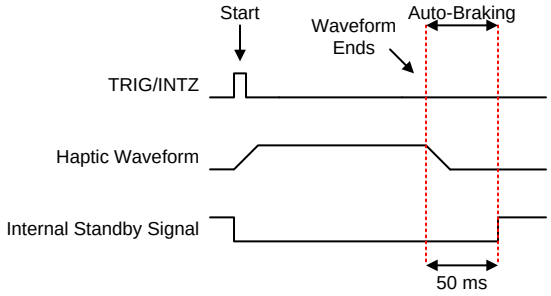



**Figure 34. Waveform Sequencer with Trigger-Pulse**

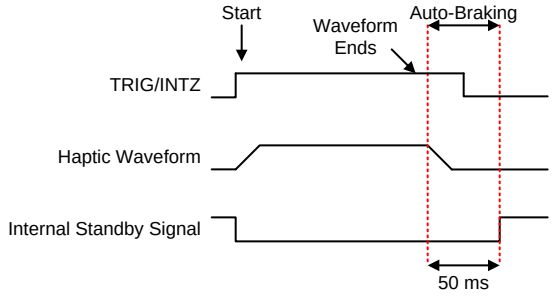



**Figure 35. Waveform Sequencer with Trigger-Level**

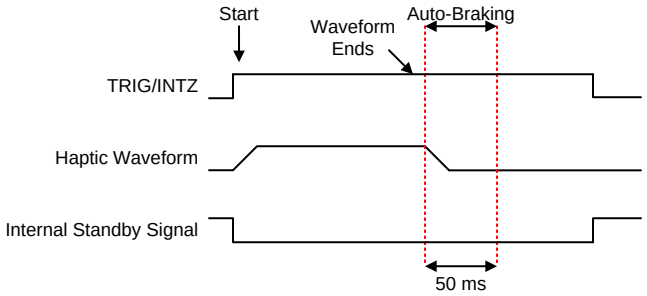



**Figure 36. Waveform Sequencer with Trigger-Level Behavior when TRIG/INTZ Pin Left High**

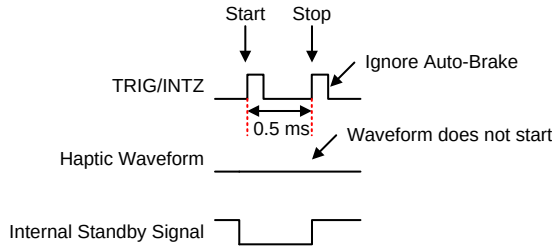



**Figure 37. Fast Start Stop with Trigger-Pulse (RTP or Waveform Sequencer)**

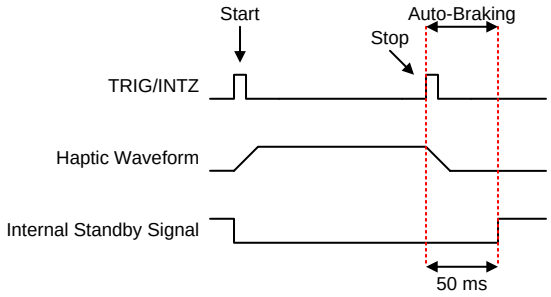



**Figure 38. Playback Start and stop Trigger (Pulse) (RTP or Waveform Sequencer)**

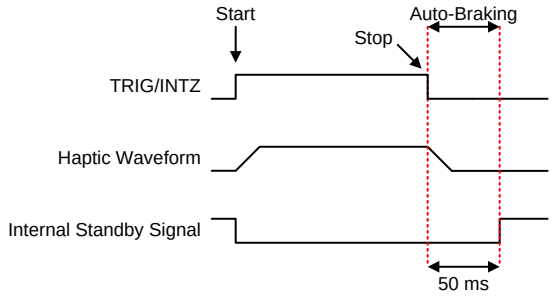



**Figure 39. Playback Start and stop Trigger (Level)(RTP or Waveform Sequencer)**

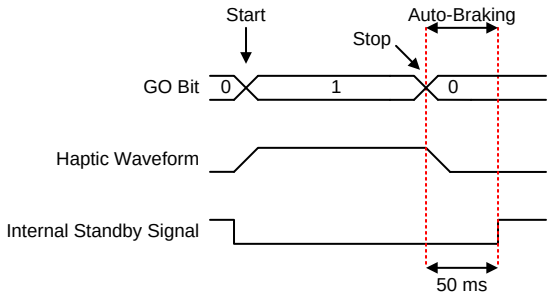



**Figure 40. Playback Start and stop Trigger (Internal GO bit) (RTP or Waveform Sequencer)**

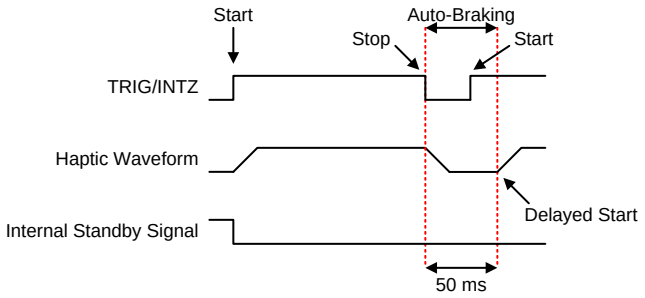



**Figure 41. Trigger-Level with Delayed Start (RTP or Waveform Sequencer)**

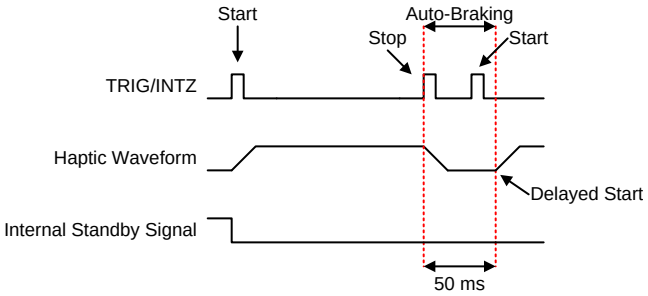



**Figure 42. Trigger-Pulse and Delayed Start (RTP or Waveform Sequencer)**

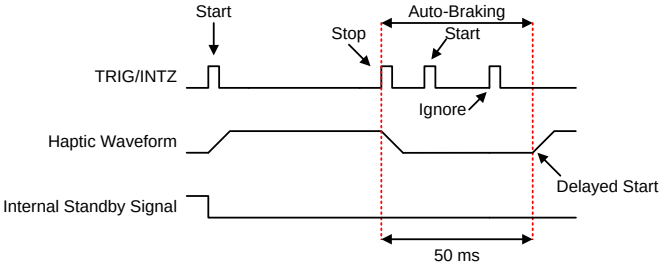



**Figure 43. Trigger-Pulse and Delayed Start with Ignored Pulse (RTP or Waveform Sequencer)**

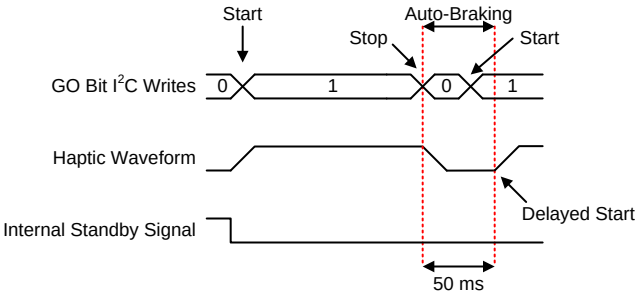



**Figure 44. GO bit Trigger with Delayed Start (RTP or Waveform Sequencer)**

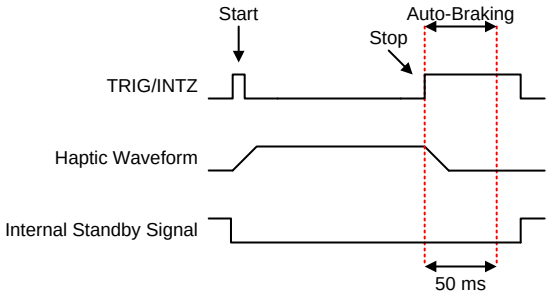



**Figure 45. Trigger-Pulse Behavior when TRIG/INTZ Pin Left High (RTP or Waveform Sequencer)**

## 8.7 Register Map

### 8.7.1 Overview

**Table 2. Register Map Overview**

| REG NO. | DEFAULT | BIT 7 | BIT 6 | BIT 5 | BIT 4 | BIT 3 | BIT 2 | BIT 1 | BIT 0 |
|---|---|---|---|---|---|---|---|---|---|
| 0x00 | 0x03 | CHIPID[3:0] | | | | REV[3:0] | | | |
| 0x01 | 0x00 | DIAG_RESULT | Reserved | | PRG_ERROR | PROCESS_DONE | UVLO | OVER_TEMP | OC_DETECT |
| 0x02 | 0x18 | Reserved | | | INTZ_MASK[4:0] | | | | |
| 0x03 | 0x00 | DIAG_Z_RESULT[7:0] | | | | | | | |
| 0x04 | 0x00 | VBAT[7:0] | | | | | | | |
| 0x05 | 0x00 | Reserved | | | | | | LRA_PERIOD[9:8] | |
| 0x06 | 0x00 | LRA_PERIOD[7:0] | | | | | | | |
| 0x07 | 0x44 | I2C_BCAST_EN | LRA_PERIOD_AVG_DIS | LINEREG_COMP_SEL[1:0] | | TRIG_PIN_FUNC[1:0] | | MODE[1:0] | |
| 0x08 | 0x88 | LRA_ERM | CONTROL_LOOP | HYBRID_LOOP | AUTO_BRK_OL | AUTO_BRK_INTO_STBY | INPUT_SLOPE_CHECK | Reserved | |
| 0x09 | 0x00 | BAT_LIFE_EXT_LVL_EN[1:0] | | Reserved | | | UVLO_THRES[2:0] | | |
| 0x0A | 0x92 | BAT_LIFE_EXT_LVL1[7:0] | | | | | | | |
| 0x0B | 0x8D | BAT_LIFE_EXT_LVL2[7:0] | | | | | | | |
| 0x0C | 0x00 | Reserved | | | | | | | GO |
| 0x0D | 0x00 | Reserved | | PLAYBACK_INTERVAL | Reserved | | | DIG_MEM_GAIN[1:0] | |
| 0x0E | 0x7F | RTP_INPUT[7:0] | | | | | | | |
| 0x0F | 0x01 | WAIT1 | WAV_FRM_SEQ1[6:0] | | | | | | |
| 0x10 | 0x00 | WAIT2 | WAV_FRM_SEQ2[6:0] | | | | | | |
| 0x11 | 0x00 | WAIT3 | WAV_FRM_SEQ3[6:0] | | | | | | |
| 0x12 | 0x00 | WAIT4 | WAV_FRM_SEQ4[6:0] | | | | | | |
| 0x13 | 0x00 | WAIT5 | WAV_FRM_SEQ5[6:0] | | | | | | |
| 0x14 | 0x00 | WAIT6 | WAV_FRM_SEQ6[6:0] | | | | | | |
| 0x15 | 0x00 | WAIT7 | WAV_FRM_SEQ7[6:0] | | | | | | |
| 0x16 | 0x00 | WAIT8 | WAV_FRM_SEQ8[6:0] | | | | | | |
| 0x17 | 0x00 | WAV4_SEQ_LOOP[1:0] | | WAV3_SEQ_LOOP[1:0] | | WAV2_SEQ_LOOP[1:0] | | WAV1_SEQ_LOOP[1:0] | |
| 0x18 | 0x00 | WAV8_SEQ_LOOP[1:0] | | WAV7_SEQ_LOOP[1:0] | | WAV6_SEQ_LOOP[1:0] | | WAV5_SEQ_LOOP[1:0] | |
| 0x19 | 0x00 | Reserved | | | | | WAV_SEQ_MAIN_LOOP[2:0] | | |
| 0x1A | 0x00 | ODT[7:0] | | | | | | | |
| 0x1B | 0x00 | SPT[7:0] | | | | | | | |
| 0x1C | 0x00 | SNT[7:0] | | | | | | | |
| 0x1D | 0x00 | BRT[7:0] | | | | | | | |
| 0x1F | 0x3F | RATED_VOLTAGE[7:0] | | | | | | | |
| 0x20 | 0x89 | OD_CLAMP[7:0] | | | | | | | |
| 0x21 | 0x0D | A_CAL_COMP[7:0] | | | | | | | |
| 0x22 | 0x6D | A_CAL_BEMF[7:0] | | | | | | | |
| 0x23 | 0x36 | NG_THRESH | FB_BRAKE_FACTOR[2:0] | | | LOOP_GAIN[1:0] | | BEMF_GAIN[1:0] | |

## Register Map (continued)

**Table 2. Register Map Overview (continued)**

| REG NO. | DEFAULT | BIT 7 | BIT 6 | BIT 5 | BIT 4 | BIT 3 | BIT 2 | BIT 1 | BIT 0 |
|---|---|---|---|---|---|---|---|---|---|
| 0x24 | 0x64 | RATED_VOLTAGE_CLAMP[7:0] | | | | | | | |
| 0x25 | 0x80 | OD_CLAMP_LVL1[7:0] | | | | | | | |
| 0x26 | 0x00 | OD_CLAMP_LVL2[7:0] | | | | | | | |
| 0x27 | 0x10 | LRA_MIN_FREQ_SEL | LRA_RESYNC_FORMAT | Reserved | DRIVE_TIME[4:0] | | | | |
| 0x28 | 0x11 | BLANKING_TIME[3:0] | | | | IDISS_TIME[3:0] | | | |
| 0x29 | 0x0C | Reserved | | OD_CLAMP_TIME[1:0] | | SAMPLE_TIME[1:0] | | ZC_DET_TIME[1:0] | |
| 0x2A | 0x02 | Reserved | | | | | | AUTO_CAL_TIME[1:0] | |
| 0x2C | 0x00 | LRA_AUTO_OPEN_LOOP | AUTO_OL_CNT[1:0] | | Reserved | | | | LRA_WAVE_SHAPE |
| 0x2E | 0x00 | Reserved | | | | | | OL_LRA_PERIOD[9:0] | |
| 0x2F | 0xC6 | OL_LRA_PERIOD[9:0] | | | | | | | |
| 0x30 | 0x00 | CURRENT_K[7:0] | | | | | | | |
| 0xFD | 0x00 | RAM_ADDR[15:8] | | | | | | | |
| 0xFE | 0x00 | RAM_ADDR[7:0] | | | | | | | |
| 0xFF | 0x00 | RAM_DATA[7:0] | | | | | | | |

### 8.7.2 Address: 0x00

**Figure 46. 0x00**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CHIPID[3:0] | | | | REV[3:0] | | | |
| R-0 | | | | R-0 | R-0 | R=1 | R=1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 3. Address: 0x00**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-4 | CHIPID[3:0] | R | 0 | Provide device identification information |
| | | | 0 | DRV2624 |
| | | | 1 | DRV2625 |
| 3-0 | REV[3:0] | R | 3 | Provides information on the device revision (1) |

(1) Rev 2 and 3 are both commercially released.

### 8.7.3 Address: 0x01

**Figure 47. 0x01**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| DIAG_RESULT[0] | Reserved | | PRG_ERROR[0] | PROCESS_DONE[0] | UVLO[0] | OVER_TEMP[0] | OC_DETECT[0] |
| R-0 | R/W-0 | | R-0 | R-0 | R-0 | R-0 | R-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 4. Address: 0x01**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | DIAG_RESULT | R | 0 | The meaning of this bit changes depending on the mode of operation. In diagnostics mode, this bit will assert if the actuator is either open or short to itself, to ground, or to vdd. For the impedance measurement mode, refer to DIAG_Z_RESULT[7:0]. For the calibration mode, this bit asserts if the calibration fails. This bit is sticky and will clear on read. |
| | | | 0 | No issue found. |
| | | | 1 | Either diagnostics, or calibration failed. |
| 6-5 | Reserved | R/W | 0 | Reserved |
| 4 | PRG_ERROR | R | 0 | If the RAM has data that does not comply with the required format, this bit will asset. This bit is sticky and will clear on read. |
| | | | 0 | No error found while reading the RAM. |
| | | | 1 | Error found while reading the RAM. |
| 3 | PROCESS_DONE | R | 0 | Shows if the process executed is done. This bit is sticky and will clear on read. |
| | | | 0 | Process is not done. |
| | | | 1 | Process is done (either waveform sequencer, diagnostics or auto-calibration). This bit is cleared when read. |
| 2 | UVLO | R | 0 | If VDD dropts below the UVLO_THRES[2:0], this bit will assert. This bit is sticky and will clear on read. |
| | | | 0 | No VDD droop has been observed. |
| | | | 1 | A VDD droop was observed. Clears on read. |
| 1 | OVER_TEMP | R | 0 | Shows current status of the thermal protection. This bit is sticky and will clear on read. |
| | | | 0 | Temperature is below over-temperature threshold |
| | | | 1 | Temperature is above over-temperature threshold. Clears on read. |
| 0 | OC_DETECT | R | 0 | Shows current status of the output overcurrent protection. This bit is sticky and will clear on read. |
| | | | 0 | No over-current detected in OUT+ or OUT- |
| | | | 1 | Over-current detected in OUT+ or OUT-. Clears on read. |

### 8.7.4 Address: 0x02

**Figure 48. 0x02**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | | INTZ_MASK[4:0] | | | | |
| R/W-0 | | | R/W-1 | R/W-1 | R/W-0 | R/W-0 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 5. Address: 0x02**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-5 | Reserved | R/W | 0 | Reserved |
| 4-0 | INTZ_MASK[4:0] | R/W | 24 | Masks status bits to configure behavior of TRIG/INTZ pin when configured in interrupt mode. Ignored otherwise. |
| | | | | 0 When INTZ_MASK[0] = 1 the OC_DETECT status will not produce an interrupt |
| | | | | 1 When INTZ_MASK[1] = 1 the OVER_TEMP status will not produce an interrupt |
| | | | | 2 When INTZ_MASK[2] = 1 the UVLO status will not produce an interrupt |
| | | | | 3 When INTZ_MASK[3] = 1 the PROCESS_DONE status will not produce an interrupt |
| | | | | 4 When INTZ_MASK[4] = 1 the PRG_ERROR status will not produce an interrupt |

### 8.7.5 Address: 0x03

**Figure 49. 0x03**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| DIAG_Z_RESULT[7:0] | | | | | | | |
| R-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 6. Address: 0x03**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | DIAG_Z_RESULT[7:0] | R | 0 | This parameter shows the impedance measurement of the actuator after running the diagnostics routine. |

### 8.7.6 Address: 0x04

**Figure 50. 0x04**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| VBAT[7:0] | | | | | | | |
| R-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 7. Address: 0x04**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | VBAT[7:0] | R | 0 | This parameter provides a real-time reading of the supply voltage at the VDD pin. The device must be actively playing a waveform to take a reading. |

### 8.7.7 Address: 0x05

**Figure 51. 0x05**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | LRA_PERIOD[9:8] | |
| R/W-0 | | | | | | R-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 8. Address: 0x05**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-2 | Reserved | R/W | 0 | Reserved |
| 1-0 | LRA_PERIOD[9:0] | R | 0 | This parameter reports the resonance frequency of the LRA in real time. Since this register is continuously being updated, the MSB section most be read first, the LSB (and the MSB) register will be retained until the LSB is read to preserve consistency. If the waveform finishes and the LSB has not been read, the device will automatically unlock both registers (MSB and LSB) and they will start to update again upon the next playback. For this reason, it is important to read both registers during the same playback to get accurate readings. LRA period = LRA_PERIOD[9:0] × 24.39 µs. The accuracy of the reported frequency is not guaranteed during braking. |

### 8.7.8 Address: 0x06

**Figure 52. 0x06**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| LRA_PERIOD[7:0] | | | | | | | |
| R-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 9. Address: 0x06**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | LRA_PERIOD[9:0] | R | 0 | This parameter reports the resonance frequency of the LRA in real time. Since this register is continuously being updated, the MSB section most be read first, the LSB (and the MSB) register will be retained until the LSB is read to preserve consistency. If the waveform finishes and the LSB has not been read, the device will automatically unlock both registers (MSB and LSB) and they will start to update again upon the next playback. For this reason, it is important to read both registers during the same playback to get accurate readings. LRA period = LRA_PERIOD[9:0] × 24.39 µs. The accuracy of the reported frequency is not guaranteed during braking. |

### 8.7.9 Address: 0x07

**Figure 53. 0x07**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| I2C_BCAST_EN[0] | LRA_PERIOD_AVG_DIS[0] | LINEREG_COMP_SEL[1:0] | | TRIG_PIN_FUNC[1:0] | | MODE[1:0] | |
| R/W-0 | R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 10. Address: 0x07**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | I2C_BCAST_EN | R/W | 0 | When enabled, the device will respond to slave address 0x58 (or 1011000). This is useful for triggering multiple drivers at the same time |
| | | | | 0 Haptic Broadcast disable. |
| | | | | 1 Haptic Broadcast enabled. |
| 6 | LRA_PERIOD_AVG_DIS | R/W | 1 | Enables/disable averaging for the resonance reporting located in LRA_PERIOD[9:0] parameter. |
| | | | | 0 LRA_PERIOD[9:0] reports the average period of the last 4 periods. Uses a shift register preloaded with 0. |
| | | | | 1 LRA_PERIOD[9:0] reports the last period. |
| 5-4 | LINEREG_COMP_SEL[1:0] | R/W | 0 | Applies a compensation factor to compensate for variations of LDO shifting. |
| | | | | 0 0% |
| | | | | 1 2% |
| | | | | 2 4% |
| | | | | 3 5% |
| 3-2 | TRIG_PIN_FUNC[1:0] | R/W | 1 | This parameter selects the function of the TRIG/INTZ pin. If this parameter is changed during process execution, the device will go into standby. |
| | | | | 0 Pin functions as external pulse trigger (input). In this mode, the GO bit can also be used to trigger or cancel processes. |
| | | | | 1 Pin functions as external level trigger - enable (input). In this mode the GO bit cannot be used. |
| | | | | 2 Pin functions as an interrupt (open drain output). In this mode, the GO bit is the only mechanism to trigger and cancel processes. |
| | | | | 3 Reserved |
| 1-0 | MODE[1:0] | R/W | 0 | This parameter is used to select the mode of operation. If the mode is changed during process execution, the device will immediately go into standby. |
| | | | | 0 RTP Mode |
| | | | | 1 Waveform Sequencer Mode |
| | | | | 2 Diagnostics Routine |
| | | | | 3 Automatic Level Calibration Routine |

### 8.7.10 Address: 0x08

**Figure 54. 0x08**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| LRA_ERM[0] | CONTROL_LOOP[0] | HYBRID_LOOP[0] | AUTO_BRK_OL[0] | AUTO_BRK_INTO_STBY[0] | INPUT_SLOPE_CHECK[0] | Reserved | |
| R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 11. Address: 0x08**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | LRA_ERM | R/W | 1 | Selects the actuator type. This bit should be set prior to running the calibration routine. |
| | | | | 0 ERM. |
| | | | | 1 LRA. |
| 6 | CONTROL_LOOP | R/W | 0 | Selects either closed loop or open loop mode. This applies to both ERM and LRA actuators. |
| | | | | 0 Closed Loop. |
| | | | | 1 Open Loop. |
| 5 | HYBRID_LOOP | R/W | 0 | Selects between full closed loop mode and hybrid closed-loop mode. |
| | | | | 0 Full closed loop mode. |
| | | | | 1 Hybrid loop mode. |
| 4 | AUTO_BRK_OL | R/W | 0 | Provides automatic braking for ERM and LRA when in open loop. To achieve this, when the data to be played is 0 or less, the DRV2625 device automatically goes into closed loop mode and then brakes the actuator. Note that when the data to be played is positive the device will play in open loop mode. This feature assumes that the actuator has been calibrated and that it functions correctly under closed-loop conditions. This feature is disabled by default. |
| | | | | 0 No automatic braking while in open loop mode. |
| | | | | 1 Automatic braking for open loop mode is enabled. Goes to closed-loop mode to break when data is 0 or less. |
| 3 | AUTO_BRK_INTO_STBY | R/W | 1 | This bit is used to enable automatic braking when the device goes into standby. If this bit is set and a waveform was playing, when a go-to-standby signal is received (either from the timer, or the EN functionality of the TRIG/INTZ pin), the device will first brake the actuator and then will transition to the standby mode. |
| | | | | 0 Go immediately to standby mode (without automatic braking). |
| | | | | 1 Before going to standby mode, check if the actuator is moving. If it is moving brake the actuator and then go to standby mode. If it is not moving then go into standby mode. |
| 2 | INPUT_SLOPE_CHECK | R/W | 0 | If bit is set, driver will operate in open loop and will only change to close loop if the transition requested is big enough. This bit is ignored if hybrid loop is disabled. |
| | | | | 0 No input slope check. |
| | | | | 1 Input slope check enabled |
| 1-0 | Reserved | R/W | 0 | Reserved |

### 8.7.11 Address: 0x09

**Figure 55. 0x09**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| BAT_LIFE_EXT_LVL_EN[1:0] | | Reserved | | | UVLO_THRES[2:0] | | |
| R/W-0 | | R/W-0 | | | R/W-0 | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 12. Address: 0x09**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-6 | BAT_LIFE_EXT_LVL_EN[1:0] | R/W | 0 | Enables the BAT_LIFE_EXT functionality. |
| | | | | 0 BAT_LIFE_EXT functionality disabled. |
| | | | | 1 BAT_LIFE_EXT_LVL1 functionality enabled. |
| | | | | 2 BAT_LIFE_EXT_LVL1 and BAT_LIFE_EXT_LVL2 functionality enabled. |
| 5-3 | Reserved | R/W | 0 | Reserved |
| 2-0 | UVLO_THRES[2:0] | R/W | 0 | Configures the UVLO threshold. If VDD voltage goes below this threshold, the output stage is immediately turn off and the device is placed into stand-by mode. |
| | | | | 0 UVLO threshold = 2.5 V. |
| | | | | 1 UVLO threshold = 2.6 V. |
| | | | | 2 UVLO threshold = 2.7 V. |
| | | | | 3 UVLO threshold = 2.8 V. |
| | | | | 4 UVLO threshold = 2.9 V. |
| | | | | 5 UVLO threshold = 3 V. |
| | | | | 6 UVLO threshold = 3.1 V. |
| | | | | 7 UVLO threshold = 3.2 V. |

### 8.7.12 Address: 0x0A

**Figure 56. 0x0A**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| BAT_LIFE_EXT_LVL1[7:0] | | | | | | | |
| R/W-1 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-0 | R/W-1 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 13. Address: 0x0A**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | BAT_LIFE_EXT_LVL1[7:0] | R/W | 146 | If VDD goes below the threshold specified by this parameter, the OD_CLAMP_LVL1 sets the overdrive clamp for the device. Note that OD_CLAMP_LVL1 should always be greater or equal to OD_CLAMP_LVL2. BAT_LIFE_EXT_LVL1 should be set higher than BAT_LIFE_EXT_LVL2. The VDD voltage is sampled at the beginning of the effect only. |

### 8.7.13 Address: 0x0B

**Figure 57. 0x0B**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| BAT_LIFE_EXT_LVL2[7:0] | | | | | | | |
| R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-1 | R/W-0 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 14. Address: 0x0B**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | BAT_LIFE_EXT_LVL2[7:0] | R/W | 141 | If VDD goes below the threshold specified by this parameter, the OD_CLAMP_LVL2 sets the overdrive clamp for the device. Note that OD_CLAMP_LVL1 should always be greater or equal to OD_CLAMP_LVL2. BAT_LIFE_EXT_LVL1 should be set higher than BAT_LIFE_EXT_LVL2. The VDD voltage is sampled at the beginning of the effect only. |

### 8.7.14 Address: 0x0C

**Figure 58. 0x0C**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | GO[0] |
| R/W-0 | | | | | | | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 15. Address: 0x0C**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-1 | Reserved | R/W | 0 | Reserved |
| 0 | GO | R/W | 0 | This bit is used to fire processes. The process fired by the GO bit is selected by the MODE parameter. The primary function of this bit is to fire playback of the waveform identifiers in he waveform sequencer (registers 0x0F to 0x16), in which case, this bit an be thought of a software trigger for haptic waveforms. The GO bit emains high until the process has completed. Clearing the GO bit during waveform playback cancels the process immediately. Using the external trigger will also assert the GO bit in a similar way as if it was written. The GO bit can be used to play effects using the waveform sequencer, run auto-calibration, and run diagnostics. |

### 8.7.15 Address: 0x0D

**Figure 59. 0x0D**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | PLAYBACK_INTERVAL[0] | Reserved | | | DIG_MEM_GAIN[1:0] | |
| R/W-0 | | R/W-0 | R/W-0 | | | R/W-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 16. Address: 0x0D**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-6 | Reserved | R/W | 0 | Reserved |
| 5 | PLAYBACK_INTERVAL | R/W | 0 | Sets the internal memory playback interval to either 5 ms or 1 ms. |
| | | | | 0 5 ms. |
| | | | | 1 1 ms. |
| 4-2 | Reserved | R/W | 0 | Reserved |
| 1-0 | DIG_MEM_GAIN[1:0] | R/W | 0 | This parameter allows for proportionally scaling down (attenuating) the effects stored in the internal library to simplify the customization of haptics. This parameter is ignored in RTP mode. |
| | | | | 0 Play effect with 100% strength. |
| | | | | 1 Play effect with 75% strength. |
| | | | | 2 Play effect with 50% strength. |
| | | | | 3 Play effect with 25% strength. |

### 8.7.16 Address: 0x0E

**Figure 60. 0x0E**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RTP_INPUT[7:0] | | | | | | | |
| R/W-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 17. Address: 0x0E**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | RTP_INPUT[7:0] | R/W | 127 | This parameter is the entry point for real-time playback (RTP) data. The digital playback engine drives the RTP_INPUT[7:0] value to the load when MODE[1:0] parameter is set to RTP mode, and the RTP process is triggered. The RTP_INPUT[7:0] value can be updated in real-time by the host controller to create haptic waveforms. The TP_INPUT[7:0] value is interpreted as an 8 bit signed number. |

### 8.7.17 Address: 0x0F

**Figure 61. 0x0F**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT1[0] | WAV_FRM_SEQ1[6:0] | | | | | | |
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 18. Address: 0x0F**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT1 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ1[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ1[6:0]. If this bit is 0, then WAV_FRM_SEQ1[6:0] is interpreted as a waveform dentifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ1[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ1[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ1[6:0] | R/W | 1 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT1 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ1[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.18 Address: 0x10

**Figure 62. 0x10**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT2[0] | WAV_FRM_SEQ2[6:0] | | | | | | |
| R/W-0 | R/W-0 | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 19. Address: 0x10**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT2 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ2[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ2[6:0]. If this bit is 0, then WAV_FRM_SEQ2[6:0] is interpreted as a waveform dentifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ2[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ2[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ2[6:0] | R/W | 0 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT2 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ2[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.19 Address: 0x11

**Figure 63. 0x11**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT3[0] | WAV_FRM_SEQ3[6:0] | | | | | | |
| R/W-0 | R/W-0 | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 20. Address: 0x11**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT3 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ3[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ3[6:0]. If this bit is 0, then WAV_FRM_SEQ31[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ3[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ3[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ3[6:0] | R/W | 0 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT3 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ3[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.20 Address: 0x12

**Figure 64. 0x12**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT4[0] | WAV_FRM_SEQ4[6:0] | | | | | | |
| R/W-0 | R/W-0 | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 21. Address: 0x12**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT4 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ4[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ4[6:0]. If this bit is 0, then WAV_FRM_SEQ4[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ4[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ4[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ4[6:0] | R/W | 0 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT4 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ4[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.21 Address: 0x13

**Figure 65. 0x13**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT5[0] | WAV_FRM_SEQ5[6:0] | | | | | | |
| R/W-0 | R/W-0 | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 22. Address: 0x13**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT5 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ5[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ5[6:0]. If this bit is 0, then WAV_FRM_SEQ5[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ5[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ5[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ5[6:0] | R/W | 0 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT5 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ5[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.22 Address: 0x14

**Figure 66. 0x14**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT6[0] | WAV_FRM_SEQ6[6:0] | | | | | | |
| R/W-0 | R/W-0 | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 23. Address: 0x14**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT6 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ6[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ6[6:0]. If this bit is 0, then WAV_FRM_SEQ6[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ6[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ6[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ6[6:0] | R/W | 0 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT6 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ6[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.23 Address: 0x15

**Figure 67. 0x15**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT7[0] | WAV_FRM_SEQ7[6:0] | | | | | | |
| R/W-0 | R/W-0 | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 24. Address: 0x15**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT7 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ7[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ7[6:0]. If this bit is 0, then WAV_FRM_SEQ7[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ7[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ7[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ7[6:0] | R/W | 0 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT7 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ7[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.24 Address: 0x16

**Figure 68. 0x16**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAIT8[0] | WAV_FRM_SEQ8[6:0] | | | | | | |
| R/W-0 | R/W-0 | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 25. Address: 0x16**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | WAIT8 | R/W | 0 | When this bit is set, the WAV_FRM_SEQ8[6:0] is interpreted as a wait time in which the playback engine idles. This bit is used to insert timed delays between sequentially played waveforms. Delay time = 10 ms × WAV_FRM_SEQ8[6:0]. If this bit is 0, then WAV_FRM_SEQ8[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 0 WAV_FRM_SEQ8[6:0] is interpreted as a waveform identifier for sequence playback. |
| | | | | 1 WAV_FRM_SEQ8[6:0] is interpreted as a delay. |
| 6-0 | WAV_FRM_SEQ8[6:0] | R/W | 0 | This parameter holds the waveform identifier of the waveform to be played. A waveform identifier is an integer value referring to the index position of a waveform in the library. Playback begins at register address 0x0F when the user asserts the GO bit (register 0x0C). hen playback of that waveform ends, the waveform sequencer plays the ext waveform identifier held in register 0x10, if the next waveform identifier is non-zero. The waveform sequencer continues in this way until he sequencer reaches an identifier value of zero, or all eight identifiers are played (register addresses 0x0F through 0x16), whichever comes first. If WAIT8 is set to 1, then this parameter is used to insert a delay given by: Delay time = 10 ms × WAV_FRM_SEQ8[6:0]. |
| | | | | 0 Signals the waveform sequencer to stop when it attempts to play this identifier. |

### 8.7.25 Address: 0x17

**Figure 69. 0x17**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAV4_SEQ_LOOP[1:0] | | WAV3_SEQ_LOOP[1:0] | | WAV2_SEQ_LOOP[1:0] | | WAV1_SEQ_LOOP[1:0] | |
| R/W-0 | | R/W-0 | | R/W-0 | | R/W-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 26. Address: 0x17**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-6 | WAV4_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT4 + WAV_FRM_SEQ4[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play only once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |
| 5-4 | WAV3_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT2 + WAV_FRM_SEQ2[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play only once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |
| 3-2 | WAV2_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT2 + WAV_FRM_SEQ2[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |
| 1-0 | WAV1_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT1 + WAV_FRM_SEQ1[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play only once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |

### 8.7.26 Address: 0x18

**Figure 70. 0x18**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| WAV8_SEQ_LOOP[1:0] | | WAV7_SEQ_LOOP[1:0] | | WAV6_SEQ_LOOP[1:0] | | WAV5_SEQ_LOOP[1:0] | |
| R/W-0 | | R/W-0 | | R/W-0 | | R/W-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 27. Address: 0x18**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-6 | WAV8_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT8 + WAV_FRM_SEQ8[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play only once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |
| 5-4 | WAV7_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT7 + WAV_FRM_SEQ7[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play only once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |
| 3-2 | WAV6_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT6 + WAV_FRM_SEQ6[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play only once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |
| 1-0 | WAV5_SEQ_LOOP[1:0] | R/W | 0 | Contains the number of times that the effect stored in WAIT5 + WAV_FRM_SEQ5[6:0] will be played before moving onto the next effect. |
| | | | | 0 No loop, play only once. |
| | | | | 1 Loop once (play twice). |
| | | | | 2 Loop twice (play 3 times). |
| | | | | 3 Loop 3 times (play 4 times). |

### 8.7.27 Address: 0x19

**Figure 71. 0x19**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | WAV_SEQ_MAIN_LOOP[2:0] | | |
| R/W-0 | | | | | R/W-0 | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 28. Address: 0x19**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-3 | Reserved | R/W | 0 | Reserved |
| 2-0 | WAV_SEQ_MAIN_LOOP[2:0] | R/W | 0 | Loop waveform sequencer for the amount of times specified in this register. The effects will be played until an identifier of 0 is reached, or until all 8 identifiers have been played. This sequence of identifiers are the ones that will be looped. |
| | | | | 0 No loop, play the identifier sequence only once. |
| | | | | 1 Loop once. |
| | | | | 2 Loop twice. |
| | | | | 3 Loop 3 times. |
| | | | | 4 Loop 4 times. |
| | | | | 5 Loop 5 times. |
| | | | | 6 Loop 6 times. |
| | | | | 7 Inifinite loops. (stop with trigger or GO bit). |

### 8.7.28 Address: 0x1A

**Figure 72. 0x1A**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| ODT[7:0] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 29. Address: 0x1A**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | ODT[7:0] | R/W | 0 | Adds a time offset to the overdrive portion of the library waveforms. Some motors require more overdrive time than others, therefore this register allows the user to add or take away overdrive time from the library waveforms. The maximum voltage value in the library waveform is automatically determined to be the overdrive portion. This register will only be useful in open loop mode. Overdrive is automatic for closed loop mode. The offset is interpreted as two's complement, so the time offset may be positive or negative. OverDrive Time Offset (ms) = ODT[7:0] × PLAYBACK_INTERVAL. |

### 8.7.29 Address: 0x1B

**Figure 73. 0x1B**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SPT[7:0] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 30. Address: 0x1B**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | SPT[7:0] | R/W | 0 | Adds a time offset to the positive sustain portion of the library waveforms. Some motors have faster/slower response time than others, therefore this register allows the user to add or take away positive sustain time from the library waveforms. Any positive voltage value other than the overdrive portion will be considered a sustain positive value. The offset is interpreted as two's complement, so the time offset may be positive or negative. Sustain Time Positive Offset (ms) = SPT[7:0] × PLAYBACK_INTERVAL. |

### 8.7.30 Address: 0x1C

**Figure 74. 0x1C**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| SNT[7:0] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 31. Address: 0x1C**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | SNT[7:0] | R/W | 0 | Adds a time offset to the negative sustain portion of the library waveforms. Some motors have faster/slower response time than others, therefore this register allows the user to add or take away negative sustain time from the library waveforms. Any negative voltage value other than the overdrive portion will be considered a sustain negative value. The offset is interpreted as two's complement, so the time offset may be positive or negative. Sustain Time Negative Offset (ms) = SNT[7:0] × PLAYBACK_INTERVAL. |

### 8.7.31 Address: 0x1D

**Figure 75. 0x1D**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| BRT[7:0] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 32. Address: 0x1D**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | BRT[7:0] | R/W | 0 | Adds a time offset to the braking portion of the library waveforms. Some motors require more braking time than others, therefore this register allows the user to add or take away brake time from the library waveforms. The most negative voltage value in the library waveform is automatically determined to be the braking portion. This register will only be useful in open loop mode. Braking is automatic for closed loop mode. The offset is interpreted as two's complement, so the time offset may be positive or negative. Time Brake Offset (ms) = BRT[7:0] × PLAYBACK_INTERVAL. |

### 8.7.32 Address: 0x1F

**Figure 76. 0x1F**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RATED_VOLTAGE[7:0] | | | | | | | |
| R/W-0 | R/W-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 33. Address: 0x1F**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | RATED_VOLTAGE[7:0] | R/W | 63 | This bit sets the reference voltage for full-scale output during closed-loop operation. The auto-calibration routine uses this register as an input, therefore this register must be written with the rated voltage value of the motor before calibration is performed. Modification of this register value should be followed by calibration to set A_CAL_BEMF appropriately. |

### 8.7.33 Address: 0x20

**Figure 77. 0x20**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| OD_CLAMP[7:0] | | | | | | | |
| R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-0 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 34. Address: 0x20**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | OD_CLAMP[7:0] | R/W | 137 | During closed-loop operation the actuator feedback allows the output voltage to go above the rated voltage during the automatic overdrive and automatic braking periods. The device limits this voltage to a maximum voltage defined in this parameter. |

### 8.7.34 Address: 0x21

**Figure 78. 0x21**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| A_CAL_COMP[7:0] | | | | | | | |
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-1 | R/W-0 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 35. Address: 0x21**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | A_CAL_COMP[7:0] | R/W | 13 | This register contains the voltage-compensation result after execution of the calibration routine. The value stored in the A_CAL_COMP bit compensates for any resistive losses in the driver. The calibration routine checks the impedance of the actuator to automatically determine an appropriate value. |

### 8.7.35 Address: 0x22

**Figure 79. 0x22**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| A_CAL_BEMF[7:0] | | | | | | | |
| R/W-0 | R/W-1 | R/W-1 | R/W-0 | R/W-1 | R/W-1 | R/W-0 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 36. Address: 0x22**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | A_CAL_BEMF[7:0] | R/W | 109 | The digital playback engine uses this value to automatically determine the appropriate feedback gain for closed-loop operation. |

### 8.7.36 Address: 0x23

**Figure 80. 0x23**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| NG_THRESH[0] | FB_BRAKE_FACTOR[2:0] | | | LOOP_GAIN[1:0] | | BEMF_GAIN[1:0] | |
| R/W-0 | R/W-0 | R/W-1 | R/W-1 | R/W-0 | R/W-1 | R/W-1 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 37. Address: 0x23**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | NG_THRESH | R/W | 0 | Output noise gate control. If the driver intends to drive a magnitude below the threshold selected in this bit, the output driver will send 0 (no output). |
| | | | | 0 4% of VDD. |
| | | | | 1 8% of VDD. |
| 6-4 | FB_BRAKE_FACTOR[2:0] | R/W | 3 | Selects the feedback gain ratio between braking gain and driving gain. In general, it is recommended to add additional feedback gain while braking so that the actuator will brake as quickly as possible. Large ratios give less stable operation than lower ones. The advanced user may choose to optimize this register. Otherwise, the default value should give good performance for most actuators. This value should be set prior to running auto calibration. |
| | | | | 0 1. |
| | | | | 1 2. |
| | | | | 2 3. |
| | | | | 3 4. |
| | | | | 4 6. |
| | | | | 5 8. |
| | | | | 6 16. |
| | | | | 7 Removes feedback during braking (braking disabled). |
| 3-2 | LOOP_GAIN[1:0] | R/W | 1 | Selects a loop gain for the feedback control. This sets how fast the loop tries to make the back-EMF (and thus motor velocity) match the input signal level. Higher loop gain (faster settling) options will give less stable operation than lower loop gain (slower settling). The advanced user may choose to optimize this register. Otherwise, the default value should give good performance for most actuators. This value should be set prior to running auto calibration. |
| | | | | 0 Very Slow. |
| | | | | 1 Slow. |
| | | | | 2 Fast. |
| | | | | 3 Very Fast. |
| 1-0 | BEMF_GAIN[1:0] | R/W | 2 | Sets the analog gain of the back-EMF amplifier. This value is interpreted differently between ERM mode and LRA mode. Auto calibration will automatically populate BEMF_GAIN with the most appropriate value for the actuator. Note that a user may overwrite this value. |
| | | | | 0 5x for LRA Mode, 0.34x for ERM Mode. |
| | | | | 1 10x for LRA Mode, 1.05x for ERM Mode. |
| | | | | 2 20x for LRA Mode, 1.82x for ERM Mode. |
| | | | | 3 30x for LRA Mode, 4x for ERM Mode. |

### 8.7.37 Address: 0x24

**Figure 81. 0x24**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RATED_VOLTAGE_CLAMP[7:0] | | | | | | | |
| R/W-0 | R/W-1 | R/W-1 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 38. Address: 0x24**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | RATED_VOLTAGE_CLAMP[7:0] | R/W | 100 | This parameter is to set a clamp for the steady state voltage provided by the driver. This clamp is enforced after the overdrive section of the waveform. Note that if the OD_CLAMP is lower than this parameter, the lower clamp will be applied. The same is true if the BAT_LIFE_EXT_LVLx is triggered. |

### 8.7.38 Address: 0x25

**Figure 82. 0x25**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| OD_CLAMP_LVL1[7:0] | | | | | | | |
| R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 39. Address: 0x25**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | OD_CLAMP_LVL1[7:0] | R/W | 128 | This parameter overwrites OD_CLAMP once VBAT is below BAT_LIFE_EXT_LVL1 value. This parameter will be ignored during autocal and diagnostics. |

### 8.7.39 Address: 0x26

**Figure 83. 0x26**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| OD_CLAMP_LVL2[7:0] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 40. Address: 0x26**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | OD_CLAMP_LVL2[7:0] | R/W | 0 | This parameter overwrites OD_CLAMP and OD_CLAMP_LVL1 once VBAT is below BAT_LIFE_EXT_LVL2 value. This parameter will be ignored during autocal and diagnostics. |

### 8.7.40 Address: 0x27

**Figure 84. 0x27**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| LRA_MIN_FREQ_SEL[0] | LRA_RESYNC_FORMAT[0] | Reserved | DRIVE_TIME[4:0] | | | | |
| R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 41. Address: 0x27**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | LRA_MIN_FREQ_SEL | R/W | 0 | Selects the minimum frequency supported by the device. |
| | | | | 0 125 Hz. |
| | | | | 1 45 Hz. |
| 6 | LRA_RESYNC_FORMAT | R/W | 0 | Selects the method for re-sync mode to operate. |
| | | | | 0 Based on LRA_MIN_FREQ_SEL. |
| | | | | 1 Based on DRIVE_TIME × 1.25. |
| 5 | Reserved | R/W | 0 | Reserved |

**Table 41. Address: 0x27 (continued)**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 4-0 | DRIVE_TIME[4:0] | R/W | 16 | LRA Mode: Sets initial guess for LRA drive time in LRA mode. Drive time is automatically adjusted for optimum drive on the fly; however, this register should be optimized for the approximate LRA frequence. If it is set too low, it may affect the actuator startup time. If it is set too high, it may cause instability. Optimum DriveTime (ms) ≈ 0.5 × LRA Period. If the LRA does not exhibit a valid BEMF, then this parameter also sets the free-running frequency when LRA is not attached or BEMF is not present.<br>ERM Mode: Sets the sample rate for the back-EMF detection. Lower drive times cause higher peak-to-average ratios in the output signal, requiring more supply headroom. Higher drive times cause the feedback to react at a slower rate. |
| | | | | 0 LRA: 0.5 ms; ERM: 1 ms. |
| | | | | 1 LRA: 0.6 ms; ERM: 1.2 ms. |
| | | | | 2 LRA: 0.7 ms; ERM: 1.4 ms. |
| | | | | 3 LRA: 0.8 ms; ERM: 1.6 ms. |
| | | | | 4 LRA: 0.9 ms; ERM: 1.8 ms. |
| | | | | 5 LRA: 1 ms; ERM: 2 ms. |
| | | | | 6 LRA: 1.1 ms; ERM: 2.2 ms. |
| | | | | 7 LRA: 1.2 ms; ERM: 2.4 ms. |
| | | | | 8 LRA: 1.3 ms; ERM: 2.6 ms. |
| | | | | 9 LRA: 1.4 ms; ERM: 2.8 ms. |
| | | | | 10 LRA: 1.5 ms; ERM: 3 ms. |
| | | | | 11 LRA: 1.6 ms; ERM: 3.2 ms. |
| | | | | 12 LRA: 1.7 ms; ERM: 3.4 ms. |
| | | | | 13 LRA: 1.8 ms; ERM: 3.6 ms. |
| | | | | 14 LRA: 1.9 ms; ERM: 3.8 ms. |
| | | | | 15 LRA: 2 ms; ERM: 4 ms. |
| | | | | 16 LRA: 2.1 ms; ERM: 4.2 ms. |
| | | | | 17 LRA: 2.2 ms; ERM: 4.4 ms. |
| | | | | 18 LRA: 2.3 ms; ERM: 4.6 ms. |
| | | | | 19 LRA: 2.4 ms; ERM: 4.8 ms. |
| | | | | 20 LRA: 2.5 ms; ERM: 5 ms. |
| | | | | 21 LRA: 2.6 ms; ERM: 5.2 ms. |
| | | | | 22 LRA: 2.7 ms; ERM: 5.4 ms. |
| | | | | 23 LRA: 2.8 ms; ERM: 5.6 ms. |
| | | | | 24 LRA: 2.9 ms; ERM: 5.8 ms. |
| | | | | 25 LRA: 3 ms; ERM: 6 ms. |
| | | | | 26 LRA: 3.1 ms; ERM: 6.2 ms. |
| | | | | 27 LRA: 3.2 ms; ERM: 6.4 ms. |
| | | | | 28 LRA: 3.3 ms; ERM: 6.6 ms. |
| | | | | 29 LRA: 3.4 ms; ERM: 6.8 ms. |
| | | | | 30 LRA: 3.5 ms; ERM: 7 ms. |
| | | | | 31 LRA: 3.6 ms; ERM: 7.2 ms. |

### 8.7.41 Address: 0x28

**Figure 85. 0x28**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| BLANKING_TIME[3:0] | | | | IDISS_TIME[3:0] | | | |
| R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-1 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 42. Address: 0x28**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-4 | BLANKING_TIME[3:0] | R/W | 1 | Time waiting for BEMF to settle. Before ADC sampling. |
| | | | 0 | LRA: 15 µs, ERM: 45 µs. |
| | | | 1 | LRA: 25 µs, ERM: 75 µs. |
| | | | 2 | LRA: 50 µs, ERM: 150 µs. |
| | | | 3 | LRA: 75 µs, ERM: 225 µs. |
| | | | 4 | LRA: 90 µs, ERM: NA. |
| | | | 5 | LRA: 105 µs, ERM: NA. |
| | | | 6 | LRA: 120 µs, ERM: NA. |
| | | | 7 | LRA: 135 µs, ERM: NA. |
| | | | 8 | LRA: 150 µs, ERM: NA. |
| | | | 9 | LRA: 165 µs, ERM: NA. |
| | | | 10 | LRA:180 µs, ERM: NA. |
| | | | 11 | LRA: 195 µs, ERM: NA. |
| | | | 12 | LRA: 210 µs, ERM: NA. |
| | | | 13 | LRA: 235 µs, ERM: NA. |
| | | | 14 | LRA: 260 µs, ERM: NA. |
| | | | 15 | LRA: 285 µs, ERM: NA. |
| 3-0 | IDISS_TIME[3:0] | R/W | 1 | Time waiting for inductor current to discharge |
| | | | 0 | LRA: 15 µs, ERM: 45 µs. |
| | | | 1 | LRA: 25 µs, ERM: 75 µs. |
| | | | 2 | LRA: 50 µs, ERM: 150 µs. |
| | | | 3 | LRA: 75 µs, ERM: 225 µs. |
| | | | 4 | LRA: 90 µs, ERM: NA. |
| | | | 5 | LRA: 105 µs, ERM: NA. |
| | | | 6 | LRA: 120 µs, ERM: NA. |
| | | | 7 | LRA: 135 µs, ERM: NA. |
| | | | 8 | LRA: 150 µs, ERM: NA. |
| | | | 9 | LRA: 165 µs, ERM: NA. |
| | | | 10 | LRA: 180 µs, ERM: NA. |
| | | | 11 | LRA: 195 µs, ERM: NA. |
| | | | 12 | LRA: 210 µs, ERM: NA. |
| | | | 13 | LRA:235 µs, ERM: NA. |
| | | | 14 | LRA: 260 µs, ERM: NA. |
| | | | 15 | LRA: 285 µs, ERM: NA. |

### 8.7.42 Address: 0x29

**Figure 86. 0x29**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | OD_CLAMP_TIME[1:0] | | SAMPLE_TIME[1:0] | | ZC_DET_TIME[1:0] | |
| R/W-0 | | R/W-0 | | R/W-1 | | R/W-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 43. Address: 0x29**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-6 | Reserved | R/W | 0 | Reserved |
| 5-4 | OD_CLAMP_TIME[1:0] | R/W | 0 | This parameter allows the user to select the maximum time the overshoot will be allowed during the overdrive and braking periods. If enabled, after this period the output voltage will clamp to the rated voltage clamp. Note that since the output is being clamped at a specified amount of time, it may not coincide with a zero-crossing, During autocal or diagnostics, this parameter is ignored. Autocal always uses automatic overdrive. |
| | | | | 0 Automatic Overdrive (overdrive time clamp is disabled) |
| | | | | 1 Clamp overdrive time to 25 ms |
| | | | | 2 Clamp overdrive time to 50 ms |
| | | | | 3 Clamp overdrive time to 100 ms |
| 3-2 | SAMPLE_TIME[1:0] | R/W | 3 | Time to wait before/after zero-crossing before adc samples BEMF amplitude. |
| | | | | 0 150 µs. |
| | | | | 1 200 µs. |
| | | | | 2 250 µs. |
| | | | | 3 300 µs. |
| 1-0 | ZC_DET_TIME[1:0] | R/W | 0 | Zero crossing detection time. |
| | | | | 0 100 µs. |
| | | | | 1 200 µs. |
| | | | | 2 300 µs. |
| | | | | 3 390 µs. |

### 8.7.43 Address: 0x2A

**Figure 87. 0x2A**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | AUTO_CAL_TIME[1:0] | |
| R/W-0 | | | | | | R/W-1 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 44. Address: 0x2A**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-2 | Reserved | R/W | 0 | Reserved |
| 1-0 | AUTO_CAL_TIME[1:0] | R/W | 2 | Duration of autocal routine. Sets the length of the auto calibration time. This should be enough time for the motor acceleration to settle when driven at the RATED_VOLTAGE value. |
| | | | | 0 250 ms. |
| | | | | 1 500 ms. |
| | | | | 2 1000 ms. |
| | | | | 3 Duration is trigger controlled (either external trigger, enable, or internal trigger). Minimum duration should be 1 s, otherwise the result of the calibration may be corrupted. Once the cancellation trigger is received, the calibration measurements will be taken, which can take several milliseconds to complete. |

### 8.7.44 Address: 0x2C

**Figure 88. 0x2C**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| LRA_AUTO_OPEN_LOOP[0] | AUTO_OL_CNT[1:0] | | Reserved | | | | LRA_WAVE_SHAPE[0] |
| R/W-0 | R/W-0 | | R/W-0 | | | | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 45. Address: 0x2C**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7 | LRA_AUTO_OPEN_LOOP | R/W | 0 | When enabled, the driver will automatically switch to open loop if ZC is not detected correctly for the number of cycles specified in AutoOpenLoop_CNT. The shape of the waveform for the open-loop will always be square, the LRA_WAVE_SHAPE bit will be ignored for this function. |
| | | | | 0 Disable auto-open-loop. |
| | | | | 1 Enable auto-open-loop. |
| 6-5 | AUTO_OL_CNT[1:0] | R/W | 0 | Counter used when AutoOpenLoop = 1 to decide when to switch to open loop |
| | | | | 0 3 attempts |
| | | | | 1 4 attempts |
| | | | | 2 5 attempts |
| | | | | 3 6 attempts |
| 4-1 | Reserved | R/W | 0 | Reserved |
| 0 | LRA_WAVE_SHAPE | R/W | 0 | Selects which shape to use for driving the LRA when in open loop mode. |
| | | | | 0 Square Wave. |
| | | | | 1 Sine Wave. |

### 8.7.45 Address: 0x2E

**Figure 89. 0x2E**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | OL_LRA_PERIOD[9:0] | |
| R/W-0 | | | | | | R/W-0 | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 46. Address: 0x2E**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-2 | Reserved | R/W | 0 | Reserved |
| 1-0 | OL_LRA_PERIOD[9:0] | R/W | 198 | This parameter sets the frequency that will be used to drive the LRA in open loop. LRA open loop period = OL_LRA_PERIOD[9:0] × 24.615 µs. |

### 8.7.46 Address: 0x2F

**Figure 90. 0x2F**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| OL_LRA_PERIOD[9:0] | | | | | | | |
| R/W-1 | R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-1 | R/W-1 | R/W-0 |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 47. Address: 0x2F**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | OL_LRA_PERIOD[9:0] | R/W | 198 | This parameter sets the frequency that will be used to drive the LRA in open loop. LRA open loop period = OL_LRA_PERIOD[9:0] × 24.615 µs. |

### 8.7.47 Address: 0x30

**Figure 91. 0x30**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CURRENT_K[7:0] | | | | | | | |
| RO-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 48. Address: 0x30**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | CURRENT_K[7:0] | RO | 0 | Stores a coefficient to be used in calculating the true impedance of the actuator from the diagnostic run. This coefficient will be used in conjunction with DIAG_Z_RESULT parameter. |

### 8.7.48 Address: 0xFD

**Figure 92. 0xFD**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RAM_ADDR[15:8] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 49. Address: 0xFD**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | RAM_ADDR[15:0] | R/W | 0 | This parameter holds the start address to read/write from/to the internal memory. |

### 8.7.49 Address: 0xFE

**Figure 93. 0xFE**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RAM_ADDR[7:0] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 50. Address: 0xFE**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | RAM_ADDR[15:0] | R/W | 0 | This parameter holds the start address to read/write from/to the internal memory. |

### 8.7.50 Address: 0xFF

**Figure 94. 0xFF**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RAM_DATA[7:0] | | | | | | | |
| R/W-0 | | | | | | | |

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

**Table 51. Address: 0xFF**

| BIT | FIELD | TYPE | DEFAULT | DESCRIPTION |
|---|---|---|---|---|
| 7-0 | RAM_DATA[7:0] | R/W | 0 | This parameter is the data entry for the internal memory. Writing to this parameter will cause the data written to be stored into the address specified in the RAM_ADDR[15:0] parameter. After a write, the controller automatically increments the address in RAM_ADDR[15:0]. This is true for both single byte and multi-byte writes. The same is true for reading from the internal memory. |

# 9 Application and Implementation

**NOTE**

Information in the following applications sections is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes. Customers should validate and test their design implementation to confirm system functionality.

## 9.1 Application Information

The typical application for a haptic driver is in a touch-enabled system that already has an application processor which makes the decision on when to execute haptic effects.

The DRV2624 device can be used fully with $I^2C$ communications (either using RTP or the memory interface). A system designer can chose to use external triggers to play low-latency effects (such as from a physical button). Figure 95 and Figure 96 show typical haptic system implementations. The system designer should not use the internal regulator (REG) to power any external load.

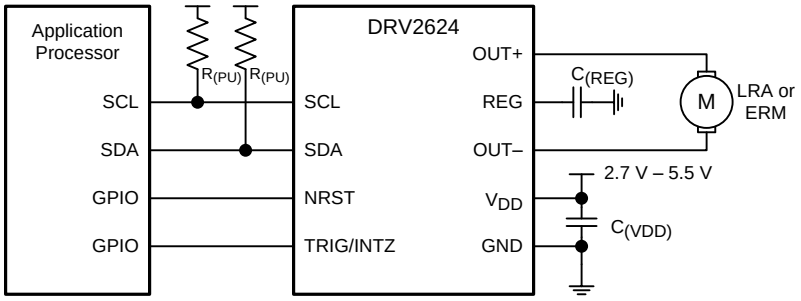



**Figure 95. $I^2C$ Control with Optional External Trigger**

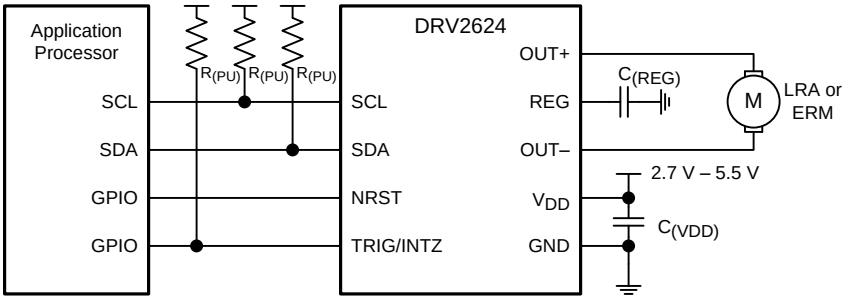



**Figure 96. $I^2C$ Control with Optional Interrupt Pin**

**Table 52. Recommended External Components**

| COMPONENT | DESCRIPTION | SPECIFICATION | TYPICAL VALUE |
|---|---|---|---|
| $C_{(VDD)}$ | Input capacitor | Capacitance | 0.1 µF |
| $C_{(REG)}$ | Regulator capacitor | Capacitance | 0.1 µF |
| $R_{(PU)}$ | Pull-up resistor | Resistance | 2.2 kΩ |

## 9.2 Typical Application

A typical application of the DRV2624 device is in a system that has external buttons which fire different haptic effects when pressed. Figure 97, Figure 98 and Figure 99 show typical schematics of such a system. The buttons can be physical buttons, capacitive-touch buttons, or GPIO signals coming from the touch-screen system.

Effects in this type of system are programmable.

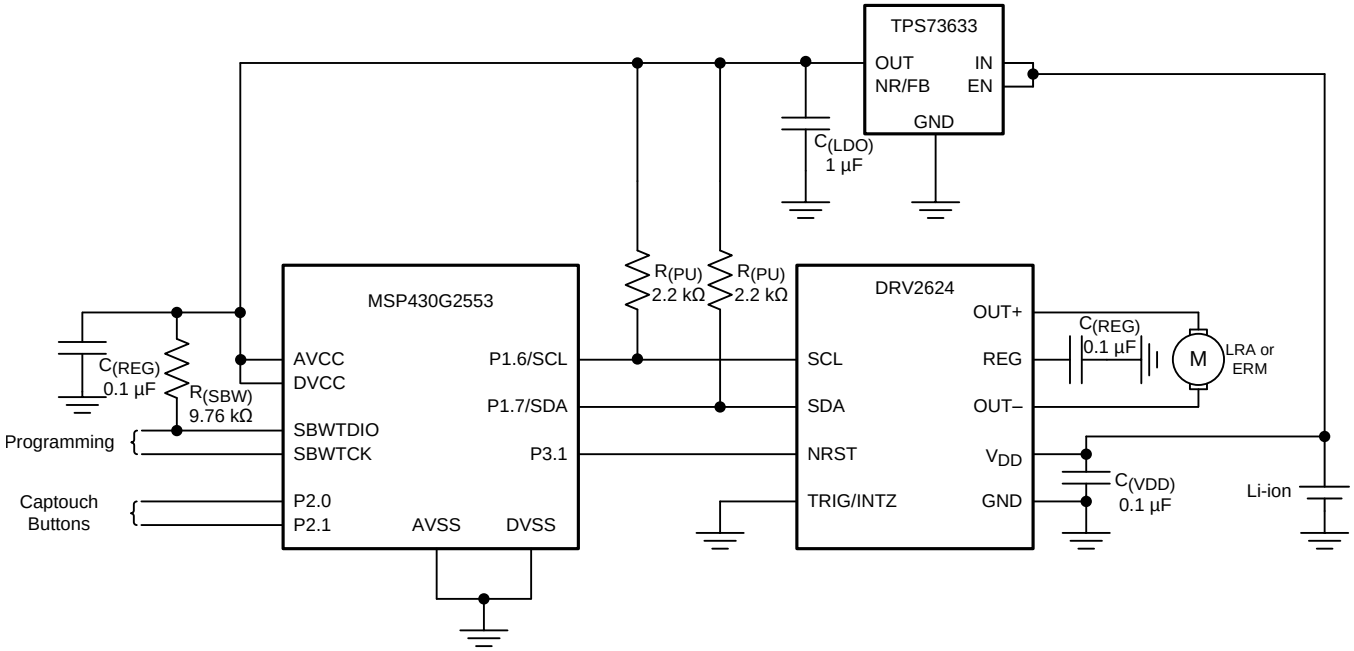



**Figure 97. Typical Application Schematic Without External Trigger or Interrupt Pin**

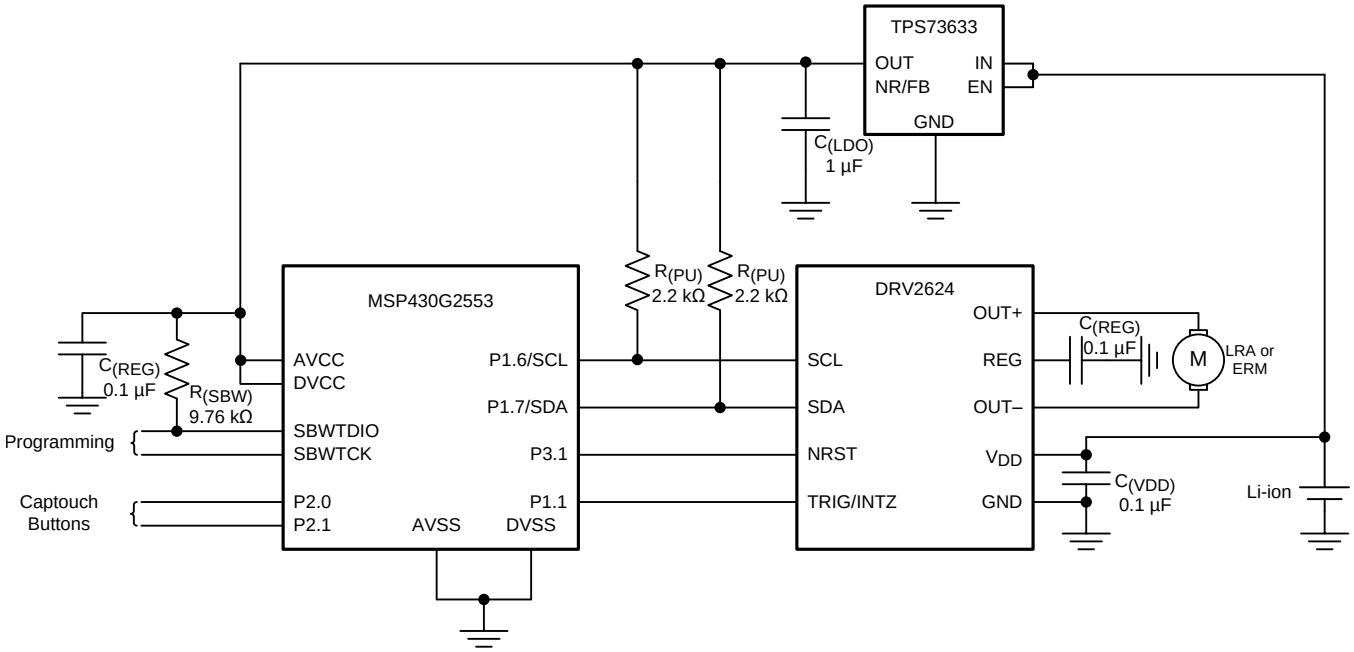



**Figure 98. Typical Application Schematic With External Trigger**

## Typical Application (continued)

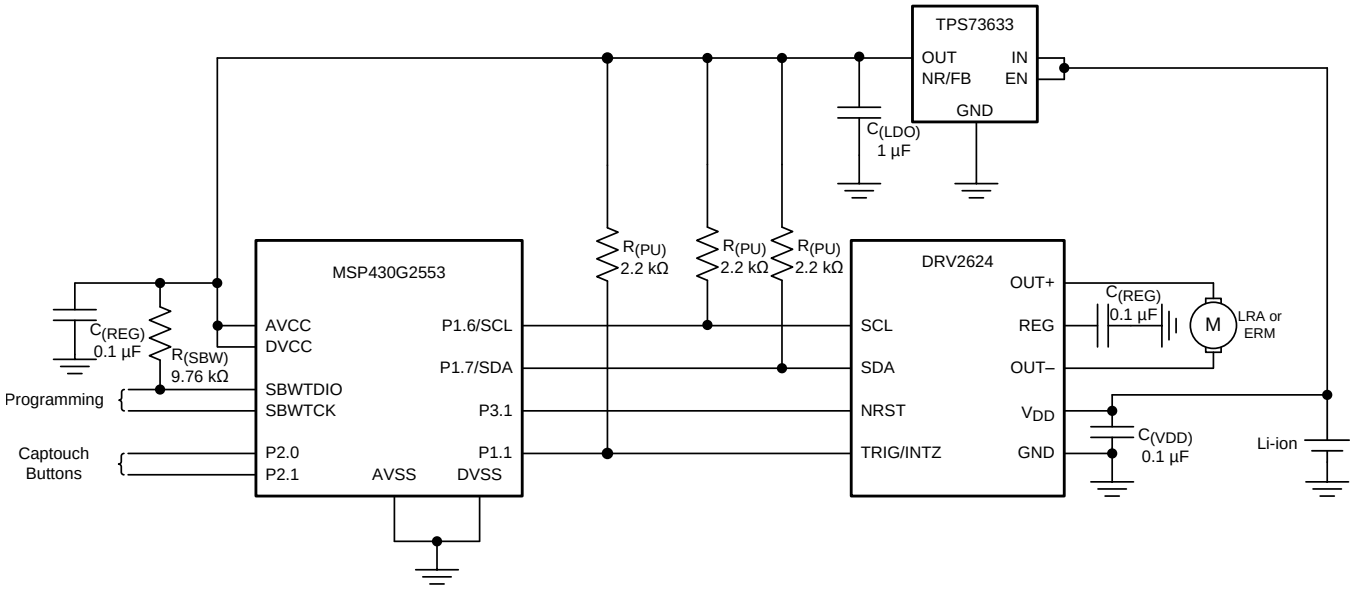



**Figure 99. Typical Application Schematic With Interrupt Pin**

### 9.2.1 Design Requirements

For this design example, use the values listed in Table 53 as the input parameters.

**Table 53. Design Parameters**

| DESIGN PARAMETER | EXAMPLE VALUE |
|---|---|
| Interface | $I^2C$, external trigger |
| Actuator type | LRA, ERM |
| Input power source | Li-ion/Li-polymer, 5-V boost |

### 9.2.2 Detailed Design Procedure

#### *9.2.2.1 Actuator Selection*

The actuator decision is based on many factors including cost, form factor, vibration strength, power-consumption requirements, haptic sharpness requirements, reliability, and audible noise performance. The actuator selection is one of the most important design considerations of a haptic system and therefore the actuator should be the first component to consider when designing the system. The following sections list the basics of ERM and LRA actuators.

##### 9.2.2.1.1 Eccentric Rotating-Mass Motors (ERM)

Eccentric rotating-mass motors (ERMs) are typically DC-controlled motors of the bar or coin type. ERMs can be driven in the clockwise direction or counter-clockwise direction depending on the polarity of voltage across the two pins. Bidirectional drive is made possible in a single-supply system by differential outputs that are capable of sourcing and sinking current. This feature helps eliminate long vibration tails which are undesirable in haptic feedback systems.

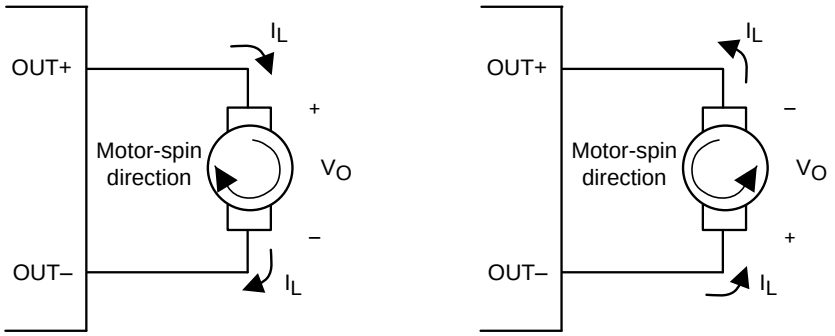



**Figure 100. Motor Spin Direction in ERM Motors**

Another common approach to driving DC motors is the concept of overdrive voltage. To overcome the inertia of the mass of the motor, these motors are often *overdriven* for a short amount of time before returning to the rated voltage of the motor to sustain the rotation of the motor. Overdrive is also used to stop (or brake) a motor quickly. Refer the data sheet of the motor for safe and reliable overdrive voltage and duration.

#### 9.2.2.1.2 Linear Resonance Actuators (LRA)

Linear resonant actuators (LRAs) vibrate optimally at the resonant frequency. LRAs have a high-Q frequency response because of a rapid drop in vibration performance at the offsets of 3 to 5 Hz from the resonant frequency. Many factors also cause a shift or drift in the resonant frequency of the actuator such as temperature, aging, the mass of the product to which the LRA is mounted, and in the case of a portable product, the manner in which the product is held. Furthermore, as the actuator is driven to the maximum allowed voltage, many LRAs will shift several hertz in frequency because of mechanical compression. All of these factors make a real-time tracking auto-resonant algorithm critical when driving LRA to achieve consistent, optimized performance.

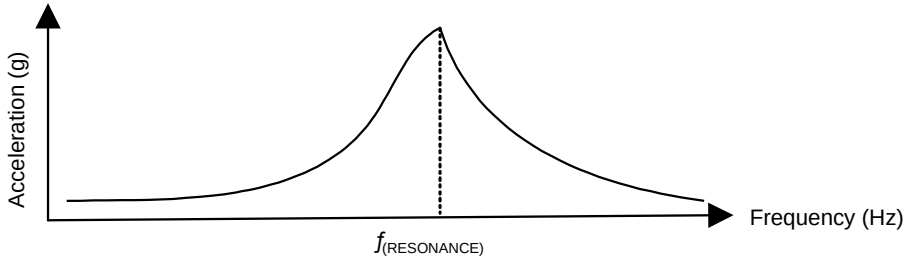



**Figure 101. Typical LRA Response**

##### *9.2.2.1.2.1 Auto-Resonance Engine for LRA*

The DRV2624 auto-resonance engine tracks the resonant frequency of an LRA in real time effectively locking into the resonance frequency after half a cycle. If the resonant frequency shifts in the middle of a waveform for any reason, the engine tracks the frequency from cycle to cycle. The auto resonance engine accomplishes the tracking by constantly monitoring the back-EMF of the actuator. Note that the auto resonance engine is not affected by the auto-calibration process which is only used for level calibration. No calibration is required for the auto resonance engine.

### *9.2.2.2 Capacitor Selection*

The DRV2624 device has a switching output stage which pulls transient currents through the $V_{DD}$ pin. Placing a 0.1-µF low equivalent-series-resistance (ESR) supply-bypass capacitor of the X5R or X7R type near the $V_{DD}$ supply pin is recommended for proper operation of the output driver and the digital portion of the device. Place a 0.1-µF X5R or X7R-type capacitor from the REG pin to ground.

#### 9.2.2.3 Interface Selection

The DRV2624 device was designed with a default configuration that supports a wide variety of LRA actuators. If an actuator compatible with the default of the device is selected, then the SimpleDrive can be used without the use of $I^2C$ interface. For this case, the user has 3 options available: controlling the NRST and TRIG/INTZ pin independently (4 pin interface: VDD, GND, NRST, TRIG/INTZ), shorting NRST with TRIG/INTZ pin and controlling them with a single GPIO (3 pin interface: VDD, GND, TRIG), or shorting NRST, TRIG/INTZ and VDD together (2 pin interface: VDD, GND). Note that for the 2-pin interface, the VDD signal must ramp faster than the startup time (about 500 µs) otherwise a UVLO condition will be detected which will prevent the device from playing the desired waveform

The $I^2C$ interface is required to configure the device. The device can be used fully with this interface with either RTP or internal memory. The advantage of using this interface is that no additional GPIO (for the TRIG/INTZ pin) is required for firing effects. Therefore the TRIG/INTZ pin can be connected to GND. Using the external trigger pin has the advantage that no $I^2C$ transaction is required to fire the pre-loaded effect, which is a good choice for interfacing with a button.

#### 9.2.2.4 Power Supply Selection

The DRV2624 device supports a wide range of voltages in the input. Ensuring that the battery voltage is high enough to support the desired vibration strength with the selected actuator is an important design consideration. The typical application uses Li-ion or Li-polymer batteries which provide enough voltage headroom to drive most common actuators.

If very strong vibrations are desired, a boost converter can be placed between the power supply and the $V_{DD}$ pin to provide a constant voltage with a healthy headroom (5-V rails are common in some systems) which is particularly true if 2 AA batteries in series are being used to power the system.

### 9.2.3 Application Curves

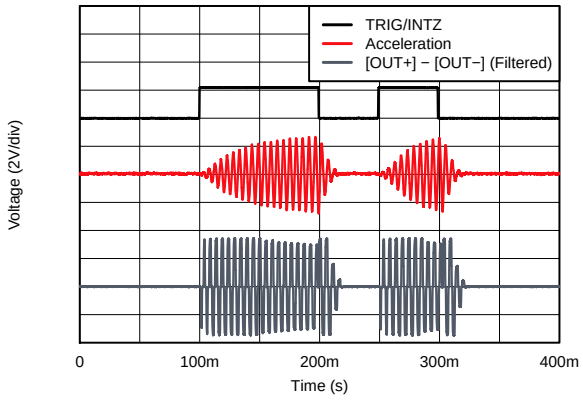



**Figure 102. LRA SimpleDrive**

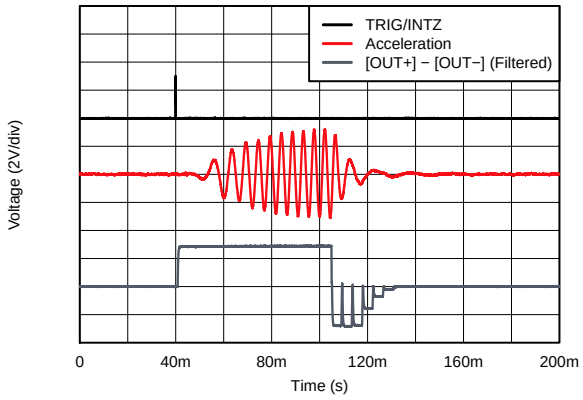



**Figure 103. ERM Open-Loop Drive with Automatic Brake**

## 9.3 Initialization Set Up

### 9.3.1 Initialization Procedure

1. After power-up, wait at least 1 ms before the DRV2624 device accepts $I^2C$ commands.
2. Assert the NRST pin (logic high). The NRST pin can be asserted any time during or after the wait period.
3. Write the MODE parameter (address 0x01) to value 0x00 to remove the device from standby mode.
4. Run auto-calibration to configure the DRV2624 device for the desired actuator. Alternatively, rewrite the results from a previous calibration.
5. If using the embedded RAM memory, populate the RAM with waveforms at this time.

## Initialization Set Up (continued)

**NOTE**

The DRV2624 device defaults to closed-loop mode with external trigger option selected in the level (enable) configuration and in RTP mode. To use other modes and features, refer to the regiter map.

### 9.3.2 Typical Usage Examples

#### *9.3.2.1 Play a Waveform or Waveform Sequence from the RAM Waveform Memory*

1. Initialize the device as listed in the *Initialization Procedure* section.
2. Select the desired TRIG/INTZ pin function by changing the TRIG_PIN_FUNC parameter.
3. Identify the waveform index to be played and populate the waveform sequencer.
4. Trigger the waveform using the desired trigger method (GO bit, or external trigger). Note that if using the interrupt functionality, only the GO bit can be used to trigger the process.
5. Device will automatically go into standby upon completion of the playback

#### *9.3.2.2 Play a Real-Time Playback (RTP) Waveform*

1. Initialize the device as shown in the *Initialization Procedure* section.
2. Write the desired drive amplitude to the real-time playback input register RTP_INPUT[7:0].
3. Trigger the waveform using the desired trigger method (GO bit, or external trigger). Note that if using the interrupt functionality, only the GO bit can be used to trigger the process.
4. Continue writing to the RTP_INPUT[7:0] if/when desired to achieve the desired haptic effect.
5. Send a stop trigger using the desired trigger method (GO bit, or external trigger). Note that if using the interrupt functionality, only the GO bit can be used to trigger the process.
6. Device will automatically go into standby upon completion of the playback

## 10 Power Supply Recommendations

The DRV2624 device is designed to operate from an input-voltage supply range between 2.7 V to 5.5 V. The decoupling capacitor for the power supply should be placed closed to the device pin.

# 11 Layout

## 11.1 Layout Guidelines

The decoupling capacitor for the power supply ($V_{DD}$ should be placed close to the device pin. The filtering capacitor for the regulator (REG) should be placed close to the device REG pin. When creating the pad size for the WCSP pins, TI recommends that the PCB layout use nonsolder mask-defined (NSMD) land. With this method, the solder mask opening is made larger than the desired land area and the opening size is defined by the copper pad width.

## 11.2 Layout Examples

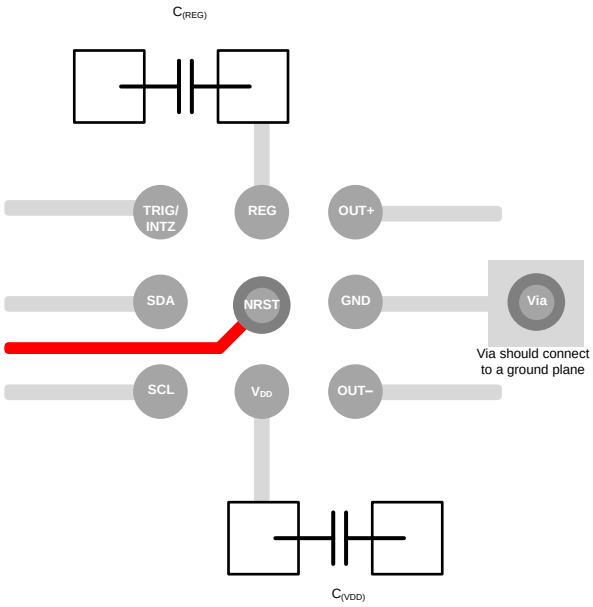



**Figure 104. Typical Layout**

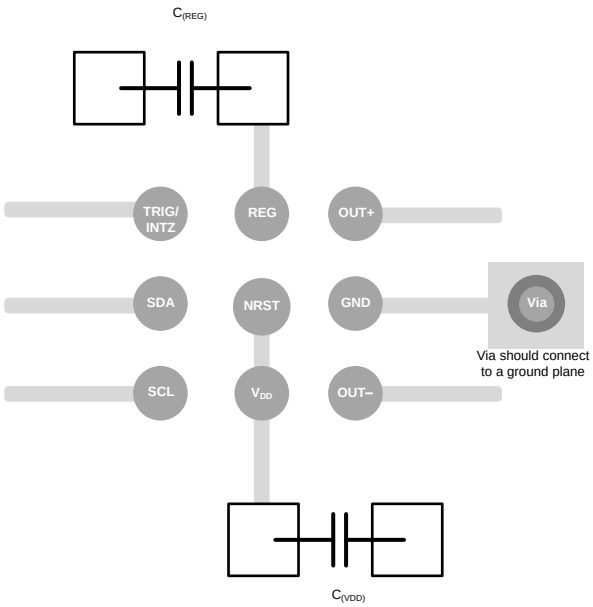



**Figure 105. Layout without NRST Functionality**

# 12 Device and Documentation Support

## 12.1 Device Support

The following links connect to TI community resources. Linked contents are provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

**TI E2E™ Online Community** ***TI's Engineer-to-Engineer (E2E) Community.*** Created to foster collaboration among engineers. At e2e.ti.com, you can ask questions, share knowledge, explore ideas and help solve problems with fellow engineers.

**Design Support** ***TI's Design Support*** Quickly find helpful E2E forums along with design support tools and contact information for technical support.

## 12.2 Trademarks

E2E is a trademark of Texas Instruments.
All other trademarks are the property of their respective owners.

## 12.3 Electrostatic Discharge Caution

These devices have limited built-in ESD protection. The leads should be shorted together or the device placed in conductive foam during storage or handling to prevent electrostatic damage to the MOS gates.

## 12.4 Glossary

SLYZ022 — *TI Glossary*.

This glossary lists and explains terms, acronyms, and definitions.

## 13 Mechanical, Packaging, and Orderable Information

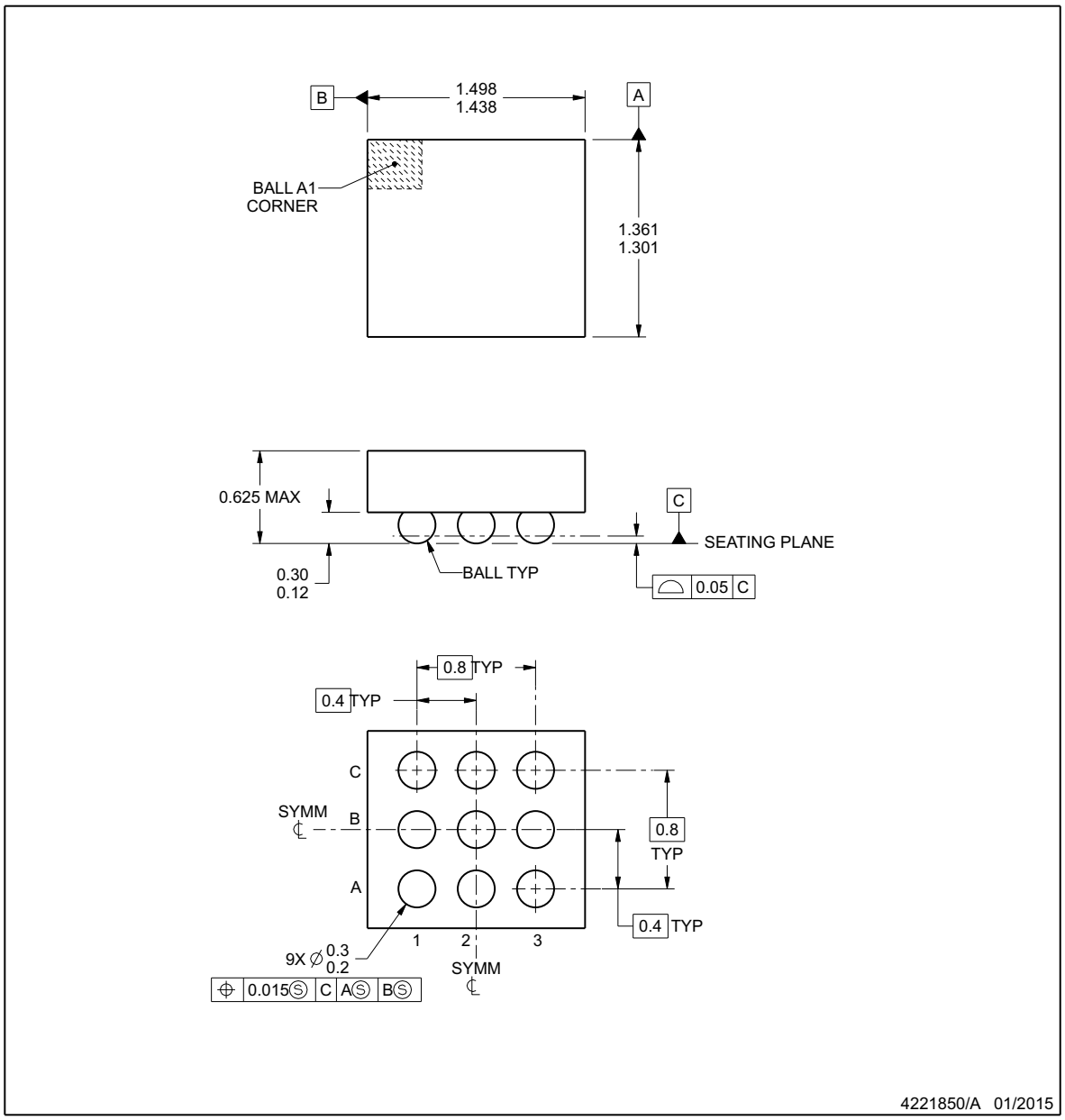
The following pages include mechanical, packaging, and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

# PACKAGE OUTLINE

## YFF0009-C01

## DSBGA - 0.625 mm max height

DIE SIZE BALL GRID ARRAY




NOTES:

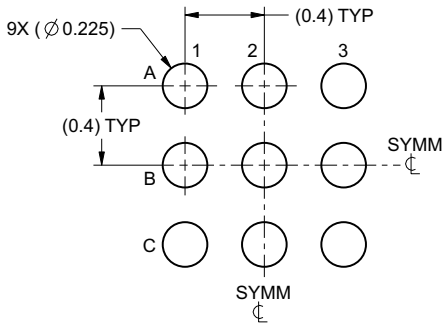
1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.

# EXAMPLE BOARD LAYOUT

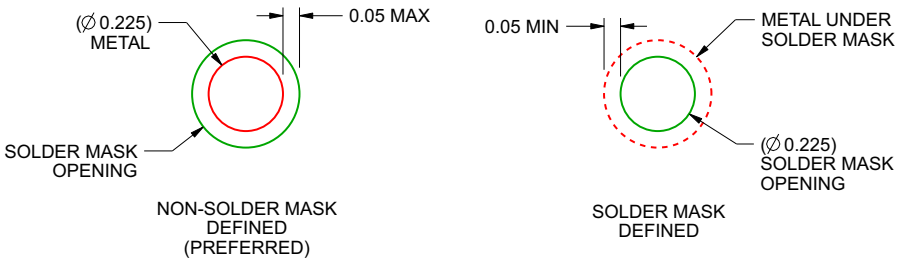
## YFF0009-C01

### DSBGA - 0.625 mm max height

DIE SIZE BALL GRID ARRAY




LAND PATTERN EXAMPLE
SCALE:30X




SOLDER MASK DETAILS
NOT TO SCALE

4221850/A 01/2015

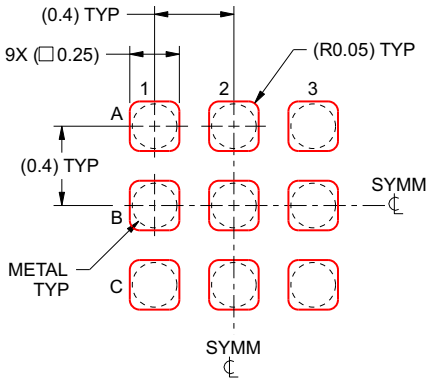
NOTES: (continued)

3. Final dimensions may vary due to manufacturing tolerance considerations and also routing constraints. For more information, see Texas Instruments literature number SNVA009 (www.ti.com/lit/snva009).

## EXAMPLE STENCIL DESIGN

**YFF0009-C01** **DSBGA - 0.625 mm max height**

DIE SIZE BALL GRID ARRAY




SOLDER PASTE EXAMPLE
BASED ON 0.1 mm THICK STENCIL
SCALE:30X

4221850/A 01/2015

NOTES: (continued)

4. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release.

## PACKAGING INFORMATION

| Orderable Device | Status (1) | Package Type | Package Drawing | Pins | Package Qty | Eco Plan (2) | Lead finish/ Ball material (6) | MSL Peak Temp (3) | Op Temp (°C) | Device Marking (4/5) | Samples |
|---|---|---|---|---|---|---|---|---|---|---|---|
| DRV2624YFFR | ACTIVE | DSBGA | YFF | 9 | 3000 | RoHS & Green | SNAGCU | Level-1-260C-UNLIM | -40 to 85 | 2624 | Samples |
| DRV2624YFFT | ACTIVE | DSBGA | YFF | 9 | 250 | RoHS & Green | SNAGCU | Level-1-260C-UNLIM | -40 to 85 | 2624 | Samples |

(1) The marketing status values are defined as follows:
**ACTIVE:** Product device recommended for new designs.
**LIFEBUY:** TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
**NRND:** Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
**PREVIEW:** Device has been announced but is not in production. Samples may or may not be available.
**OBSOLETE:** TI has discontinued the production of the device.

(2) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".
**RoHS Exempt:** TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.
**Green:** TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

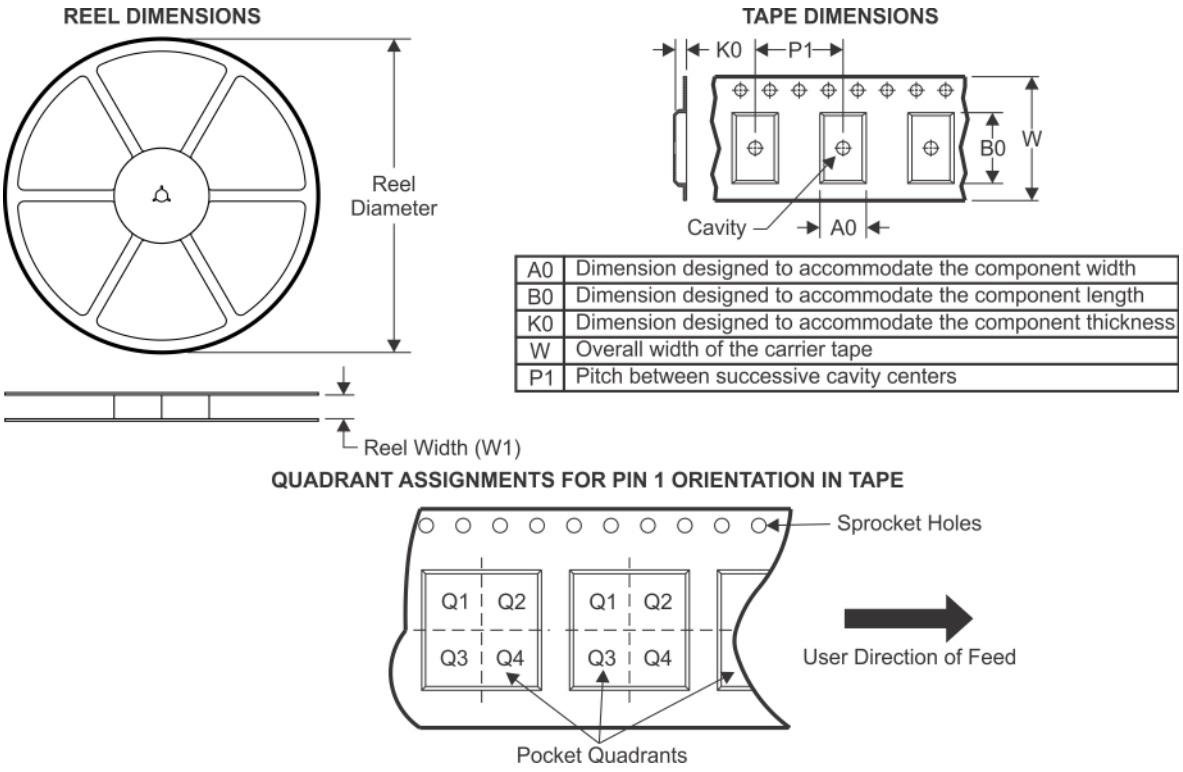
(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

(6) Lead finish/Ball material - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

## TAPE AND REEL INFORMATION

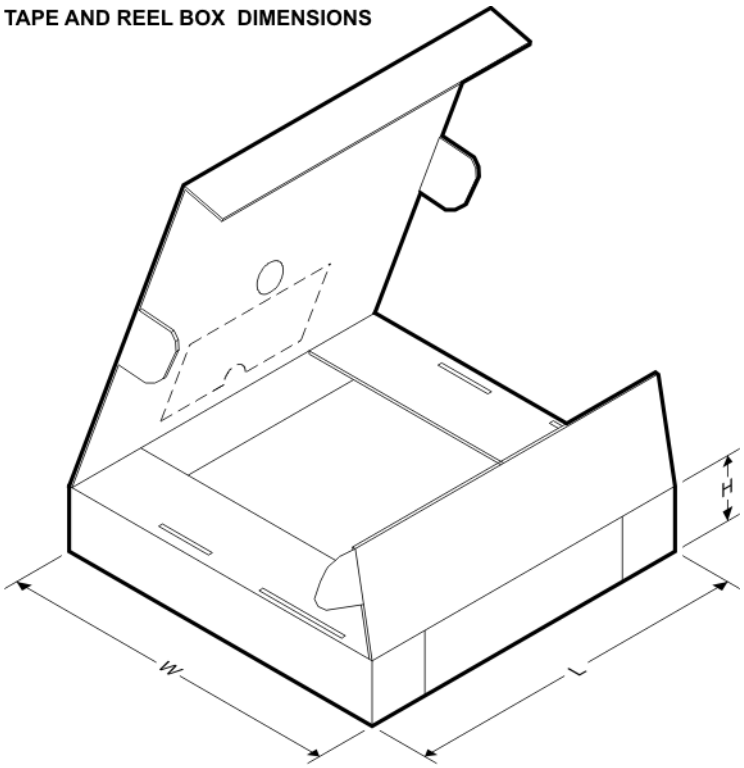



| | |
|---|---|
| A0 | Dimension designed to accommodate the component width |
| B0 | Dimension designed to accommodate the component length |
| K0 | Dimension designed to accommodate the component thickness |
| W | Overall width of the carrier tape |
| P1 | Pitch between successive cavity centers |

*All dimensions are nominal

| Device | Package Type | Package Drawing | Pins | SPQ | Reel Diameter (mm) | Reel Width W1 (mm) | A0 (mm) | B0 (mm) | K0 (mm) | P1 (mm) | W (mm) | Pin1 Quadrant |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DRV2624YFFR | DSBGA | YFF | 9 | 3000 | 180.0 | 8.4 | 1.52 | 1.56 | 0.71 | 4.0 | 8.0 | Q1 |
| DRV2624YFFT | DSBGA | YFF | 9 | 250 | 180.0 | 8.4 | 1.52 | 1.56 | 0.71 | 4.0 | 8.0 | Q1 |

TAPE AND REEL BOX DIMENSIONS




*All dimensions are nominal

| Device | Package Type | Package Drawing | Pins | SPQ | Length (mm) | Width (mm) | Height (mm) |
|---|---|---|---|---|---|---|---|
| DRV2624YFFR | DSBGA | YFF | 9 | 3000 | 182.0 | 182.0 | 20.0 |
| DRV2624YFFT | DSBGA | YFF | 9 | 250 | 182.0 | 182.0 | 20.0 |

## IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265