# F0 Detect And Tracking LRA Haptic Driver

## Features

- 1MHz I2C Bus
- 12-KByte Memory
- 12k/24k/48k input wave sampling rate
- F0 detect and tracking
- Advance autobrake engine integrated
- Playback mode:
  - Real time playback(Up to 4KByte FIFO)
  - Memory playback
  - 3 Trigger playback
  - One wire playback
  - Cont playback
- Resistance-Based LRA Diagnostics
- Drive signal monitor for LRA protect
- Drive Compensation Over Battery Discharge
- Fast Start Up Time ＜1ms
- Dedicated interrupt output pin
- Support automatically switch to standby mode
- Standby current：8μA@Vbat=3.6V
- Shutdown current：0.1μA
- Supply voltage range 3 to 5.2V
- Short-Circuit Protection, Over-Temperature Protection, Under-Voltage Protection
- WBQFN 4mmX4mmX0.75mm-32L Package with Wettable Flanks
- AEC-Q100 Qualified with Grade 2

## Applications

- Steering wheel vibration feedback
- Screen vibration feedback
- Virtual key haptic design

*All trademarks are the property of their respective owners.*

## General Description

AW86205QNR-Q1 is a normal voltage H-bridge, single chip LRA haptic driver, with F0 detecting and tracking based on BEMF, supporting real time playback, memory playback, cont playback, one wire and hardware pin trigged playback. A typical startup time of 1ms makes the AW86205QNR-Q1 an ideal haptic driver for fast responses.

AW86205QNR-Q1 integrates a 12KByte SRAM for user-defined waveforms to achieve a variety of vibration experiences, supporting 3 sampling rate(12k/24k/48k) of waveforms loaded in SRAM, supporting output waveform sampling rate up-sampling to 48k.

AW86205QNR-Q1 integrates an autobrake engine to suppress the aftershocks to zero for different drive waveforms(short or long) on different LRA motor.

AW86205QNR-Q1 supports LRA fault diagnostic based on resistance measurement and protections of short-circuit, over-temperature and under-voltage.

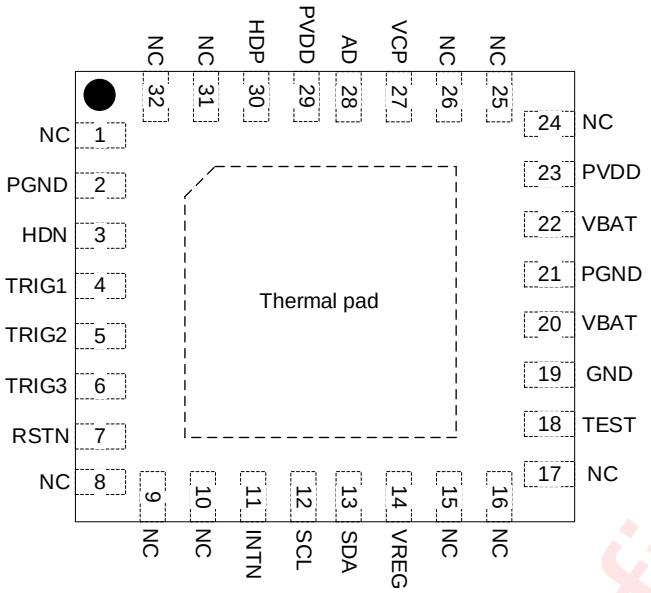
AW86205QNR-Q1 features configurable automatically switch to standby mode. This can less quiescent power consumption. Dedicated interrupt output pin can detect real time FIFO status and the error status of the chip.

AW86205QNR-Q1 features general settings are communicated via an I2C-bus interface and its I2C address is configurable.

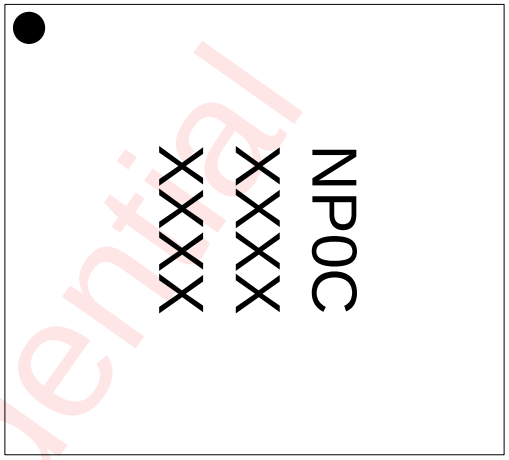
AW86205QNR-Q1 is available in a WBQFN 4mmX4mmX0.75mm-32L package.

## Pin Configuration And Top Mark

AW86205QNR-Q1
(Top View)




AW86205QNR-Q1 Mark
(Top View)




NP0C –AW86205QNR-Q1
XXXX/XXXX- Production Tracing Code

**Figure 1** Pin Configuration and Top Mark

## Pin Definition

| PIN NUMBER | NAME | I/O | DESCRIPTION |
|---|---|---|---|
| 2/21 | PGND | Ground | H-bridge driver GND |
| 3 | HDN | O | Negative haptic driver differential output |
| 4 | TRIG1 | I | Hardware trigger 1. Internal have 2MΩ pull-down resistor. |
| 5 | TRIG2 | I | Hardware trigger 2. Internal have 2MΩ pull-down resistor. |
| 6 | TRIG3 | I | Hardware trigger 3. Internal have 2MΩ pull-down resistor. |
| 7 | RSTN | I | Active low hardware reset. High: standby/active mode Low: power-down mode. Internal have 2MΩ pull-down resistor |
| 11 | INTN | O | Interrupt open drain output, low active. |
| 12 | SCL | I | I2C bus clock input |
| 13 | SDA | IO | I2C bus data input/output(open drain) |
| 14 | VREG | Power | Digital power supply |
| 18 | TEST | I | Test pin. Internal have 2MΩ pull-down resistor. |
| 19 | GND | Ground | Supply ground |
| 20/22 | VBAT | Power | Chip power supply |
| 23/29 | PVDD | Power | H driver power |
| 27 | VCP | O | Internal charge pump voltage |
| 28 | AD | I | I2C bus address selection |
| 30 | HDP | O | Positive haptic driver differential output |
| 1/8/9/10/15/16/17/ 24/25/26/31/32 | NC | NC | Not connected |

## Functional Block Diagram

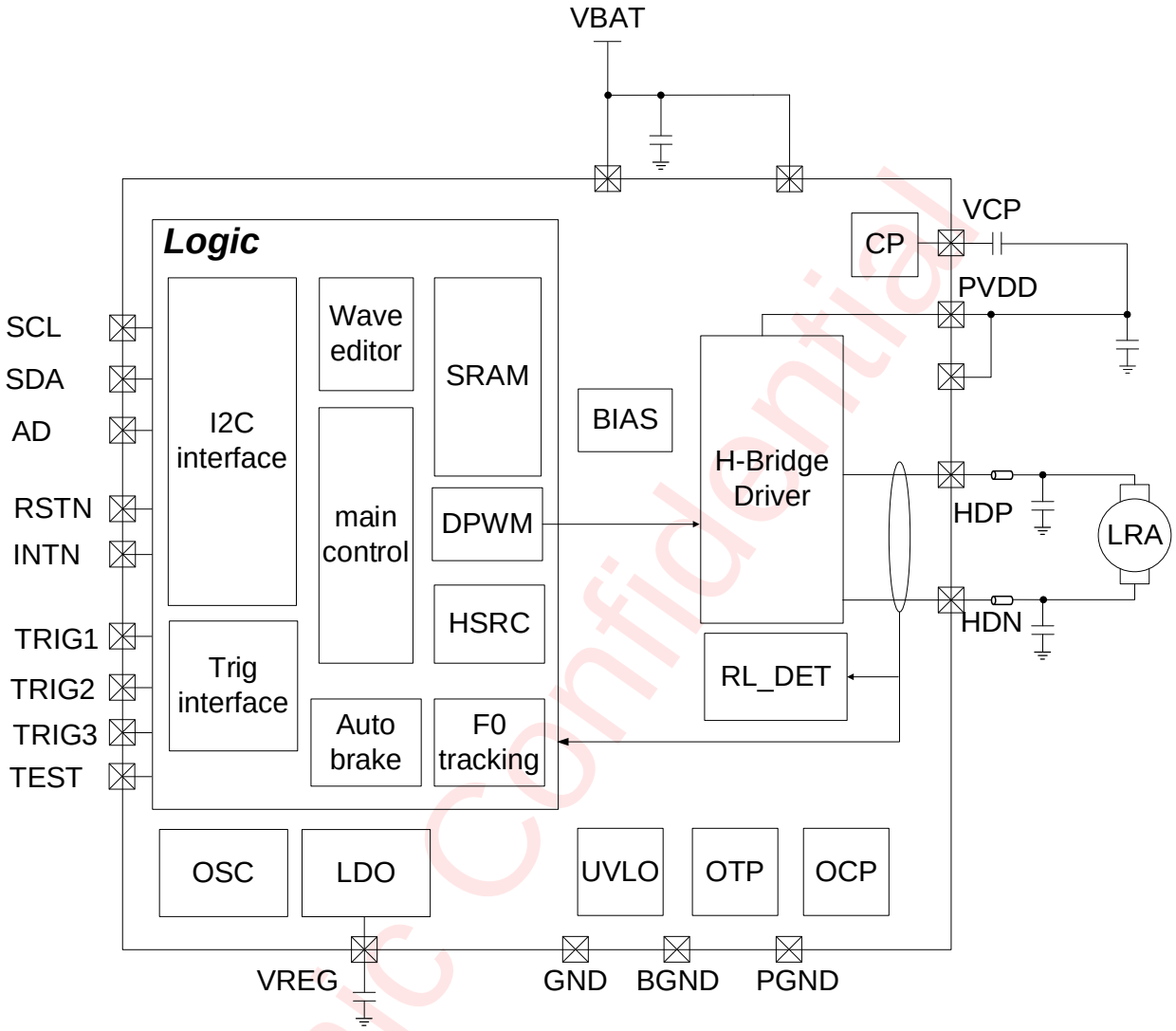



**Figure 2 FUNCTIONAL BLOCK DIAGRAM**

## Typical Application Circuits

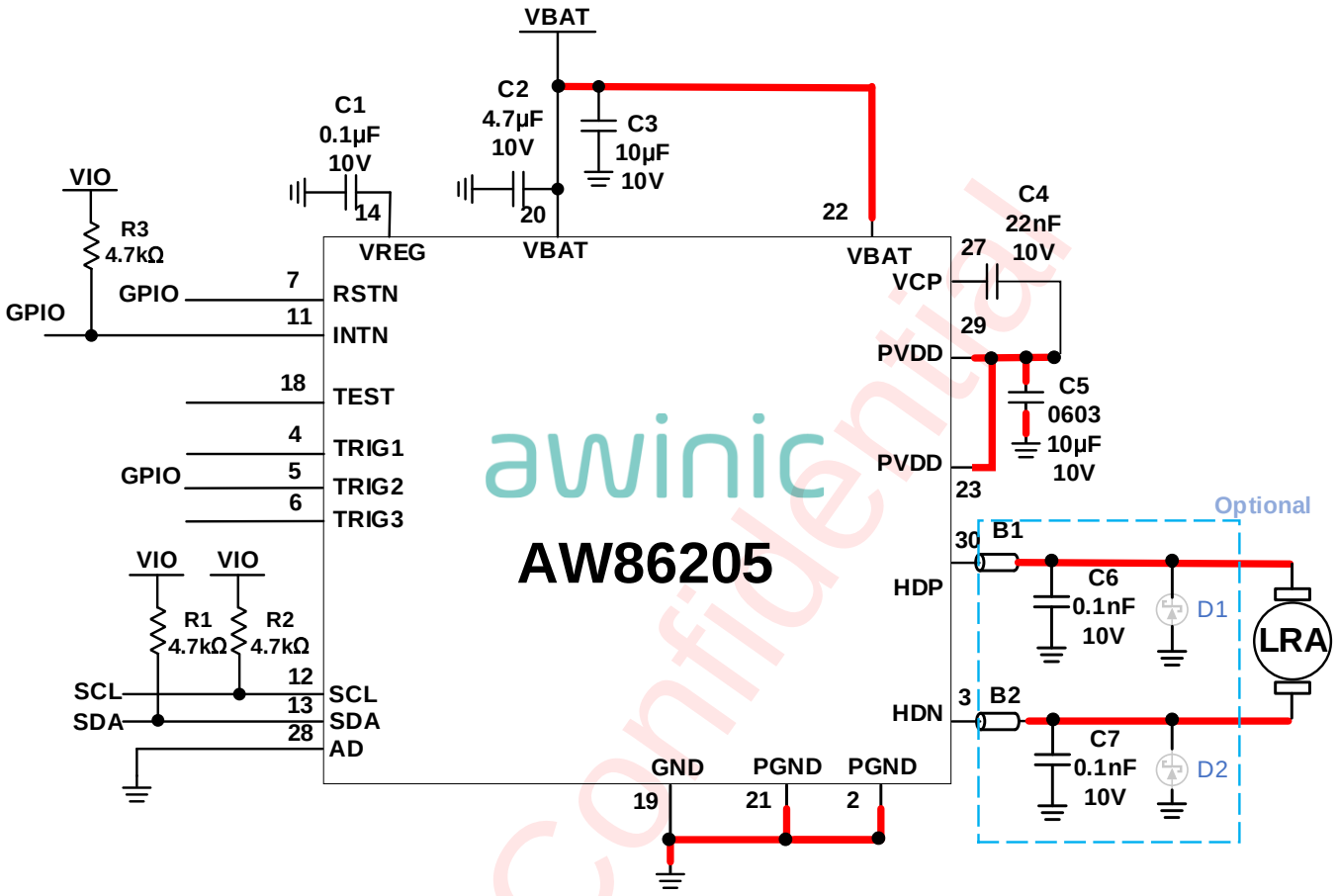



**Figure 3 Typical Application Circuit of AW86205QNR-Q1**



Notice for Typical Application Circuits:

1: Please place C1，C2，C3，C4，C5 as close to the chip as possible.

2: In order to prevent EMI problems, it is recommended to reserve bead B1/B2 (default 0Ω) and capacitor C6/C7 (default NC) at the output pin HDP/HDN, the value of C6/C7 can not exceed 100pF.The parameter of D1/D2(default NC) is recommended as follow, Vrwm is above 6V,the Vc is below 10V.

3: For the sake of driving capability, the power lines, output lines should be short and wide as possible.

## Ordering Information

| Part Number | Temperature | Package | Marking | Moisture Sensitivity Level | Environment Information | Delivery Form |
|---|---|---|---|---|---|---|
| AW86205QNR-Q1 | -40°C ~ 105°C | WBQFN 4mmX4mmX0.75mm-32L | NP0C | MSL3 | ROHS+HF | 6000 units/ Tape and Reel |

## Absolute Maximum Ratings(NOTE 1)

| Parameter | Range |
|---|---|
| Battery Supply Voltage VBAT | -0.3V to 6.0V |
| Digital power supply VREG | -0.3V to 2.0V |
| H driver power voltage PVDD | -0.3V to 13V |
| Package Thermal Resistance $\theta_{JA}$ | 45°C/W |
| Ambient Temperature Range | -40°C to 105°C |
| Maximum Junction Temperature $T_{JMAX}$ | 150°C |
| Storage Temperature Range $T_{STG}$ | -65°C to 150°C |
| Lead Temperature(Soldering 10 Seconds) | 260°C |
| ESD Rating (NOTE 2 3) | |
| HBM(Human Body Model) | ±2KV |
| CDM(Charge Device Model) | ±1.5KV |
| Latch-up | |
| Test Condition: JEDEC EIA/JESD78E | +IT: 200mA<br>-IT: -200mA |

*NOTE 1: Conditions out of those ranges listed in "absolute maximum ratings" may cause permanent damages to the device. In spite of the limits above, functional operation conditions of the device should within the ranges listed in "recommended operating conditions". Exposure to absolute-maximum-rated conditions for prolonged periods may affect device reliability.*

*NOTE 2：The human body model is a 100pF capacitor discharged through a 1.5kΩ resistor into each pin. Test method: ANSI/ESDA/JEDEC JS-001-2017.*

*NOTE 3：Charge Device Model test method: JEDEC EIA/JESD22-C101F.*

## Recommended Operating Conditions

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| VBAT | Input voltage | 3 | 3.6 | 5.2 | V |
| C1 | Vreg capacitance | 0.1 | 0.1 | 0.47 | μF |
| C2 | Input capacitance | 2.2 | 4.7 | 10 | μF |
| C3 | Input capacitance | 4.7 | 10 | 22 | μF |
| C4 | Vcp capacitance | 22 | 22 | 47 | nF |
| C5 | H driver power capacitance | 4.7 | 10 | 22 | μF |
| R1/R2/R3 | Pull up resistor | 1 | 4.7 | 10 | kΩ |

# Electrical Characteristics

## Characteristics

Test condition: TA=25°C，VBAT=4.2V，RL=8Ω+100μH(unless otherwise noted)

| Symbol | Description | Test Conditions | Min | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $V_{VBAT}$ | Battery supply voltage | On pin VBAT | 3 | | 5.2 | V |
| $V_{VREG}$ | Voltage at VREG pin | | | 1.8 | | V |
| $V_{IL}$ | Logic input low level | RSTN/TRIG1/TRIG2/TRIG3/ AD/ SDA/SCL | | | 0.5 | V |
| $V_{IH}$ | Logic input high level | RSTN/TRIG1/TRIG2/TRIG3/ AD/ SDA/SCL | 1.3 | | | V |
| $V_{OL}$ | Logic output low level | INTN/SDA<br>$I_{OUT}$=4mA | | | 0.4 | V |
| $V_{OS}$ | Output offset voltage | $I^2C$ signal input 0 | -30 | 0 | 30 | mV |
| $I_{SD}$ | Shutdown current | VBAT=4.2V, RSTN =0V | | 0.1 | 1 | μA |
| $I_{STBY}$ | Standby current | VBAT=3.6V, AD= 0V<br>TRIG1=TRIG2=TRIG3=0V<br>RSTN=SCL=SDA=1.8V | | 8 | | μA |
| $I_Q$ | Quiescent current | VBAT=3.6V@Bypass | | 5 | | mA |
| UVP | Under-voltage protection voltage | | | 2.7 | | V |
| | Under-voltage protection hysteresis voltage | | | 100 | | mV |
| $T_{SD}$ | Over temperature protection threshold | | | 160 | | °C |
| $T_{SDR}$ | Over temperature protection recovery threshold | | | 130 | | °C |
| $T_{on1}$ | Time from shutdown to standby | | | | 6 | ms |
| $T_{on2}$ | Time from standby to active | From trigger to output signal | | 1 | | ms |
| **HDRIVER** | | | | | | |
| $R_{dson}$ | Drain-Source on-state resistance | Include H and L NMOS | | 300 | | mΩ |
| $R_{ocp}$ | Load impedance threshold for over current protection | VBAT=3.6V | | 2.3 | | Ω |
| $F_{PWM}$ | PWM output frequency | VBAT=4.2V, PD_HWM=0 | | 96 | | kHz |
| | | VBAT=4.2V, PD_HWM=1 | | 48 | | kHz |

| Symbol | Description | Test Conditions | Min | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $F_{CALI\_ACC\_LRA}$ | LRA Consistency Calibration accuracy | | F0-2 | F0 | F0+2 | Hz |
| $V_{peak}$ | Output voltage | RL=16Ω+100μH VBAT=4.2V | | 3.9 | | V |
| | Output voltage | RL=8Ω+100μH VBAT=4.2V | | 3.7 | | V |

## I²C Interface Timing

| Parameter | | | fast mode | | | fast mode plus | | | UNIT |
|---|---|---|---|---|---|---|---|---|---|
| No. | Symbol | Name | MIN | TYP | MAX | MIN | TYP | MAX | |
| 1 | $f_{SCL}$ | SCL Clock frequency | | | 400 | | | 1000 | kHz |
| 2 | $t_{LOW}$ | SCL Low level Duration | 1.3 | | | 0.5 | | | μs |
| 3 | $t_{HIGH}$ | SCL High level Duration | 0.6 | | | 0.26 | | | μs |
| 4 | $t_{RISE}$ | SCL, SDA rise time | | | 0.3 | | | 0.12 | μs |
| 5 | $t_{FALL}$ | SCL, SDA fall time | | | 0.3 | | | 0.12 | μs |
| 6 | $t_{SU:STA}$ | Setup time SCL to START state | 0.6 | | | 0.26 | | | μs |
| 7 | $t_{HD:STA}$ | (Repeat-start) Start condition hold time | 0.6 | | | 0.26 | | | μs |
| 8 | $t_{SU:STO}$ | Stop condition setup time | 0.6 | | | 0.26 | | | μs |
| 9 | $t_{BUF}$ | the Bus idle time START state to STOP state | 1.3 | | | 0.5 | | | μs |
| 10 | $t_{SU:DAT}$ | SDA setup time | 0.1 | | | 0.1 | | | μs |
| 11 | $t_{HD:DAT}$ | SDA hold time | 10 | | | 10 | | | ns |

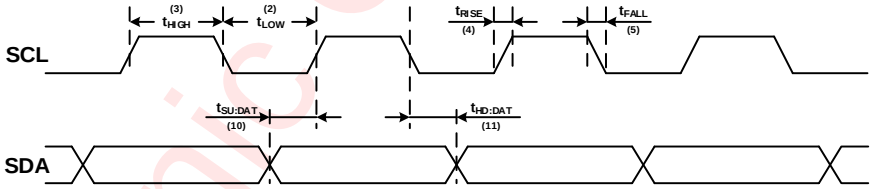



**Figure 4 SCL and SDA timing relationships in the data transmission process**

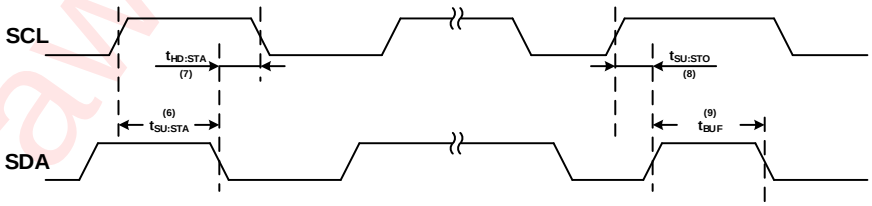



**Figure 5 The timing relationship between START and STOP state**

## Measurement Setup

AW86205QNR-Q1 features switching digital output, as shown in Figure 6. Need to connect a low pass filter to HDP/HDN output respectively to filter out switch modulation frequency, then measure the differential output of filter to obtain analog output signal.

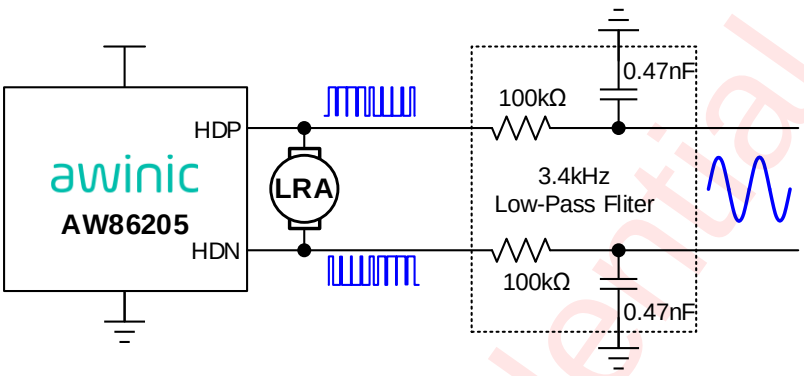



**Figure 6 AW86205QNR-Q1 test setup**

## Typical Characteristics

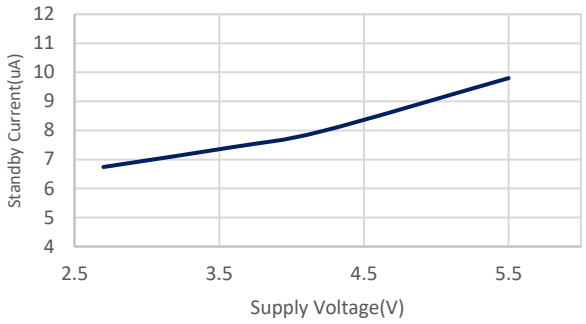



**Figure 7** Standby Current Vs Supply Voltage

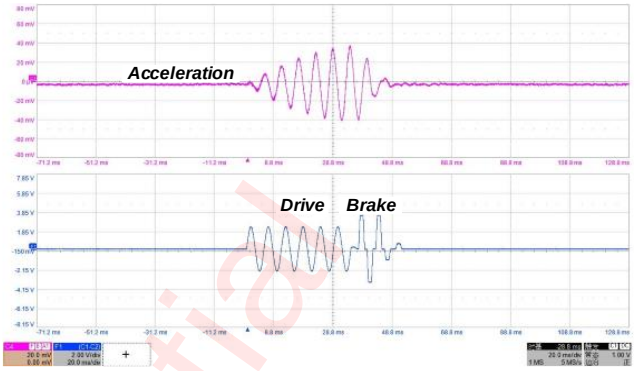



**Figure 9** LRA with Automatic Braking

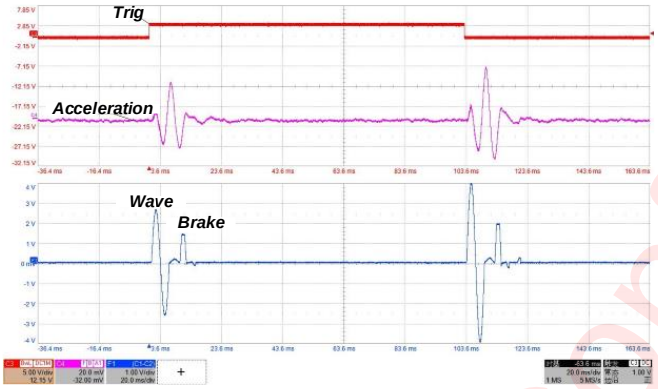



**Figure 8** Trig Application

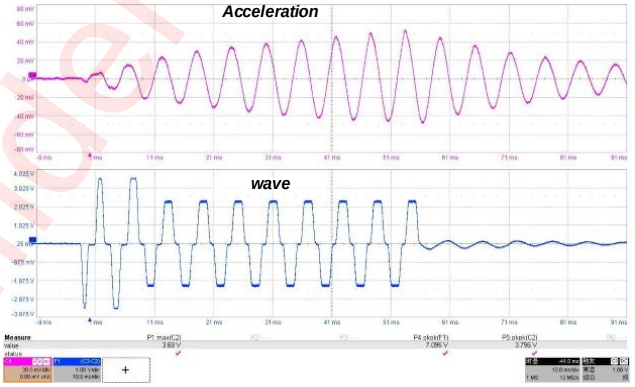



**Figure 10** Automatic Resonance Tracking

# Detailed Functional Description

## Power On Reset

The device provides a power-on reset feature that is controlled by VREG OK. The reset signal will be generated to perform a power-on reset operation, which will reset all circuits and configuration registers. When the VBAT power on, the VREG voltage raises and produce the OK indication, the reset is over.

## Operation Mode

The device supports 3 operation modes.

**Table 1 Operating Mode**

| Mode | Condition | Description |
|---|---|---|
| Power-Down | VBAT = 0V or RSTN = 0V | Power supply is not ready or RSTN is tie to low.<br>Whole chip shutdown including I²C interface. |
| Standby | VBAT > 2.7V<br>and RSTN = HIGH<br>and no wave is going | Power supply is ready and RSTN is tie to high.<br>Most parts of the device are power down for low power consumption except I²C interface and LDO. |
| Active | Playing a waveform | Most parts of the device are working |

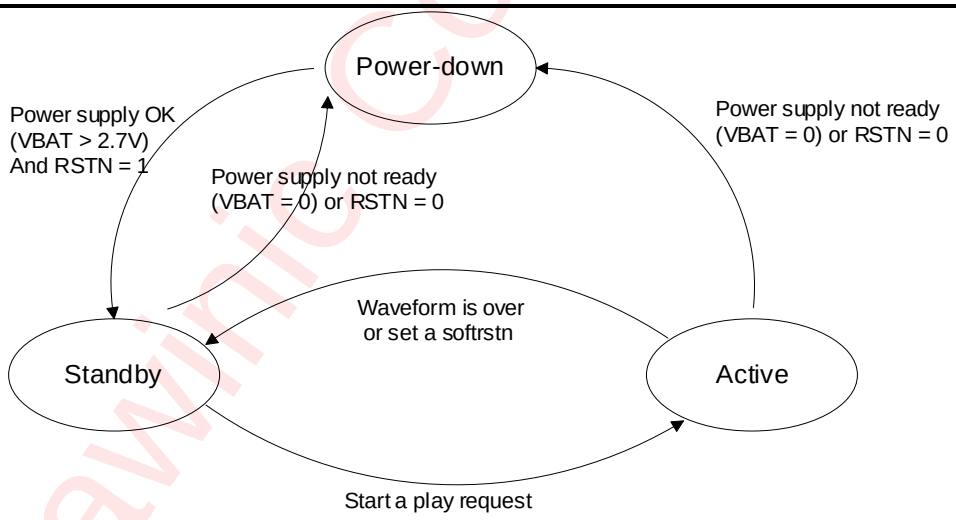



**Figure 11 Device operating modes transition**

***POWER-DOWN MODE***

The device switches to power-down mode when the supply voltage is not ready or RSTN pin is set to low.
In this mode, all circuits inside this device will be shut down. I²C interface isn't accessible in this mode, and all of the internal configurable registers and memory are cleared.
The device will jump out of the power-down mode automatically when the supply voltages are OK and RSTN pin is set to high.

***Standby Mode***

The device switches standby mode when the power supply voltages are OK and RSTN pin set to high. In this mode I²C interface is accessible, other modules except LDO module are still powered down. Also in this mode, customer can initialize waveform library in SRAM. Device will be switched to this mode after haptic waveform playback finished.

***Active Mode***

The device is fully operational in this mode. H-bridge driver circuits will start to work. Users can send a playback request to make device in this mode.

***Power On And Power Down Sequence***

This device power on and power down sequence is illustrated in the following figure:

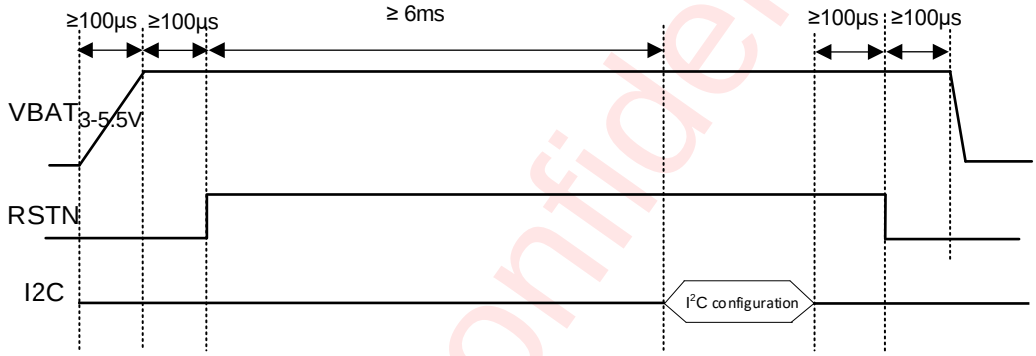



**Figure 12 Power On and Power Down Sequence**

## Playback Sequence

Make sure the device is not in POWER-DOWN MODE before sending a playback request, then the playback sequence is illustrated in the following figure:

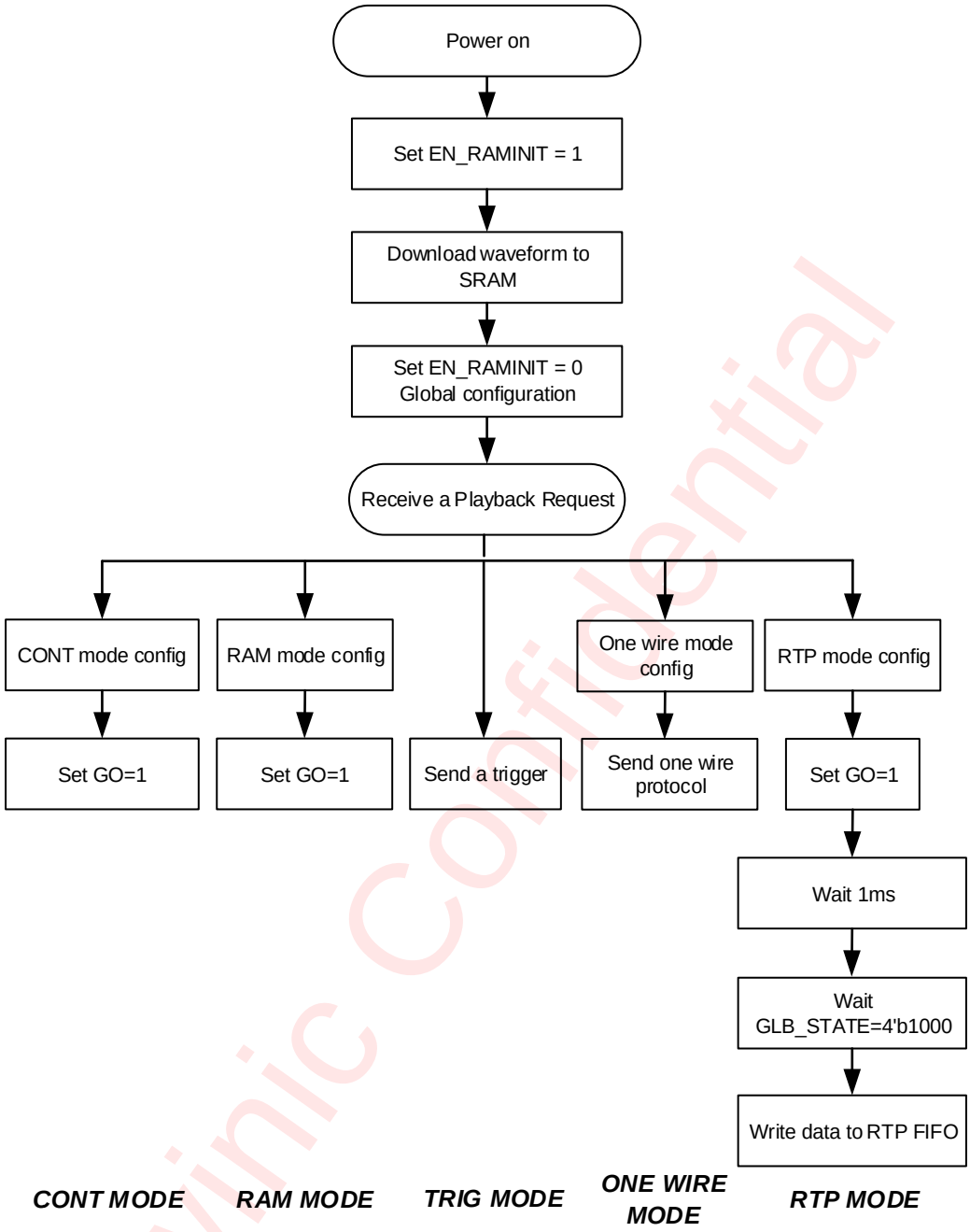



**Figure 13 Power on and playback sequence**

## Software Reset

Writing 0xAA to register CHIPID(0x00) via I²C interface will reset the device internal circuits except SRAM, including configuration registers.

## Battery Voltage Detect

Software can send command to detect the battery voltage.

**Detect steps:**

- Set EN_RAMINIT to 1 in register 0x43;

- Set VBAT_GO to 1 in register 0x52;
- Delay 3ms;
- Set EN_RAMINIT to 0 in register 0x43;
- Read VBAT_LO in register 0x57 and VBAT_H in register 0x55. Code= {VBAT_H, VBAT_LO}.

The code is a 10bit unsigned number.

$$VBAT = \frac{6.1 \times code}{1024}(V)$$

## Constant Vibration Strength

The device features power-supply feedback. If the supply voltage discharge over time, the vibration strength remains the same as long as enough supply voltage is available to sustain the required output voltage. It is especially useful for ring application.

## LRA Consistency Calibration

Different motor batches, assembly conditions and other factors can result in f0 deviation of LRA. When the drive waveform does not match the LRA monomer, the vibration may be inconsistent and the braking effect becomes worse, especially for short vibration waveforms. So it's necessary to perform consistency calibration of LRA. Firstly the power-on f0 detection can be launched to get the f0 of LRA. Secondly the waveform frequency stored in SRAM and the f0 of LRA are used to calculate the code for calibration. The f0 accuracy after LRA consistency calibration is ±2Hz.

## LRA Resistance Detect

Software can send command to detect the LRA's resistance. Based on this information host can diagnosis used LRA's status.

**Detect steps:**

- Set EN_RAMINIT to 1 in register 0x43;
- Set RL_OS to 1 in register 0x51;
- Set DIAG_GO to 1 in register 0x52;
- Delay 3ms;
- Set EN_RAMINIT to 0 in register 0x43;
- Read RL_H in register 0x53 and RL_LO in register 0x57. Code= {RL_H, RL_LO}.

The code is a 10bit unsigned number.

$$RL = \frac{678 \times code}{1024 \times \mathrm{d2s\_gain}}(\Omega)$$

## Flexible Haptic Data Playback

The device offers multiple ways to playback haptic effects data. The PLAY_MODE bits select RAM mode, RTP mode, CONT mode. Additional flexibility is provided by the three hardware TRIG pins, which can override PLAY_MODE bit to playback haptic effects data as configuration.
The device contains 12 kB of integrated SRAM to store customer haptic waveforms' data. The whole SRAM is separated to RAM waveform library and RTP FIFO region by base address. And RAM waveform library is

including waveform library version, waveform header and waveform data.

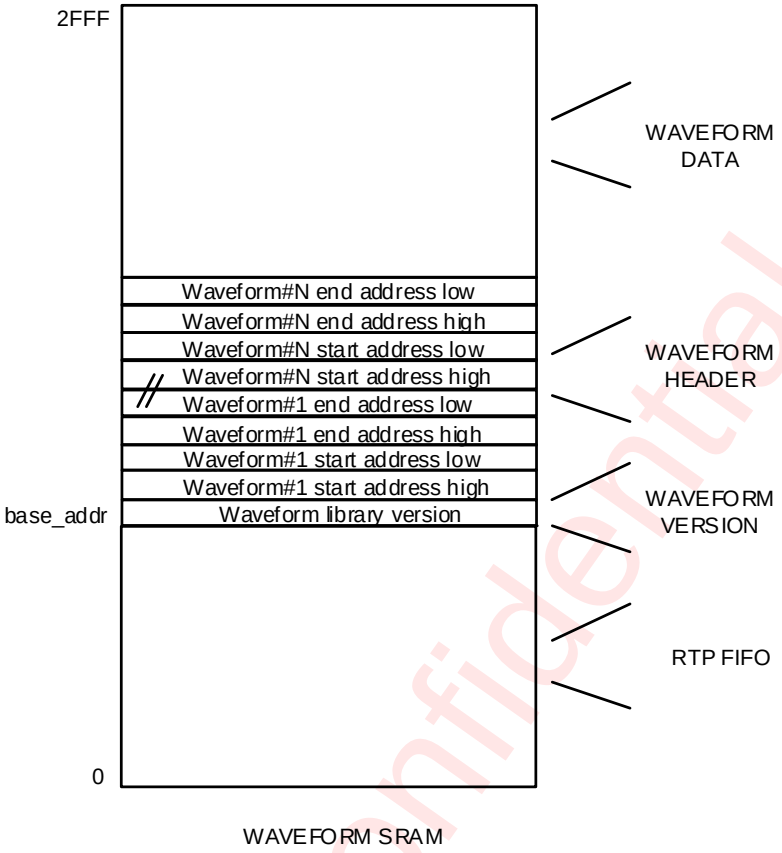



**Figure 14 Data structure in SRAM**

RAM mode and TRIG mode playback the waveforms in RAM waveform library and RTP mode playback the waveform data written in RTP FIFO, CONT mode playback non-filtered or filtered square wave with rated drive voltage.

***Sram Structure***

A SRAM waveform library consists of a waveform version byte, a waveform header section, and the waveform data content. The waveform header defines the data boundaries for each waveform ID in the data field, and the waveform data contains a signed data format (2's complement) to specify the magnitude of the drive.

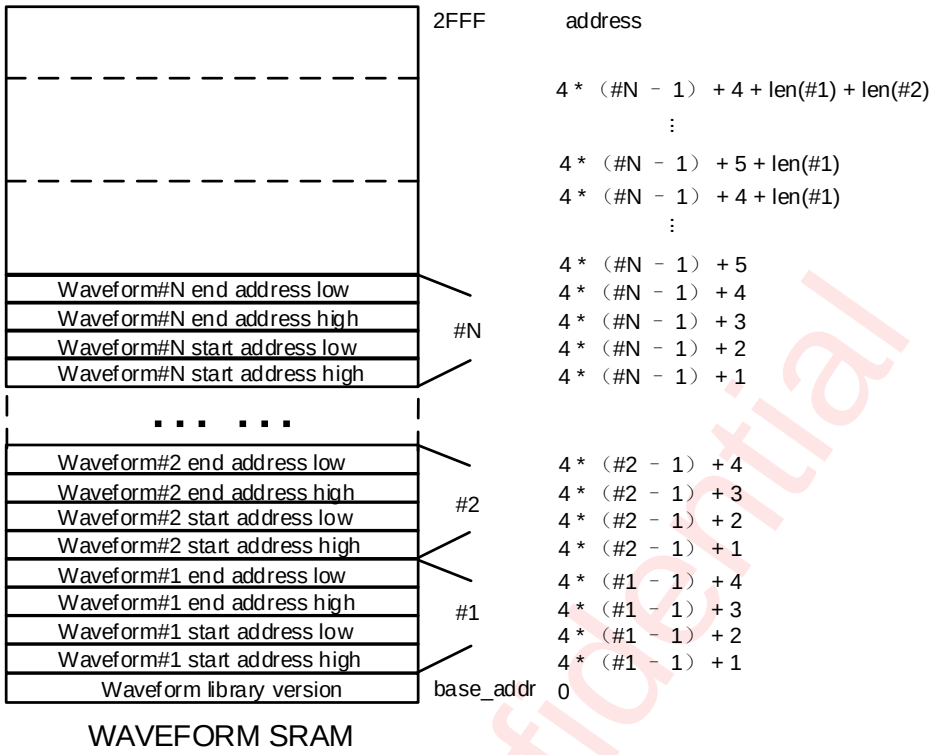



**Figure 15 Waveform library data structure**

**Waveform version:**

One byte located on SRAM base address, setting to different value to identify different version of RAM waveform library.

**Waveform header:**

The waveform header block consist of N-boundary definition blocks of 4 bytes each. N is the number of waveforms stored in the SRAM (N cannot exceed 127). Each of the boundary definition blocks contain the start address (2 bytes) and end address (2 bytes). So the total length of waveform header block are N*4 bytes.

The start address contains the location in the memory where the waveform data associated with this waveform begins.

The end address contains the location in the memory where the waveform data associated with this waveform ends.

The waveform ID is determined after base address is defined. Four bytes begins with the address next to base address are the first waveform ID’s header, and next four bytes are the second waveform ID’s header, and so on.

**Waveform data:**

The waveform data contains a signed data format (2's complement) to specify the magnitude of the drive. The begin address and end address is specified in waveform ID’s header.

**Waveform library initialization steps:**

- Prepare waveform library data including: waveform library version, waveform header fields for waveform in library and waveform data of each waveform;
- Set register EN_RAMINIT=1 in register 0x43, to enable SRAM initial;
- Delay 1ms;

- Set base address (register 0x2D, 0x2E);
- Set ram address (register 0x40, 0x41);
- Write waveform library data into register 0x42 continually until all the waveform library data written;
- Set register EN_RAMINIT=0, to disable SRAM initial.

***Ram Mode***

To playback haptic data with RAM mode, the waveform ID must first be configured into the waveform playback queue and then the waveform can be played by writing GO bit register.

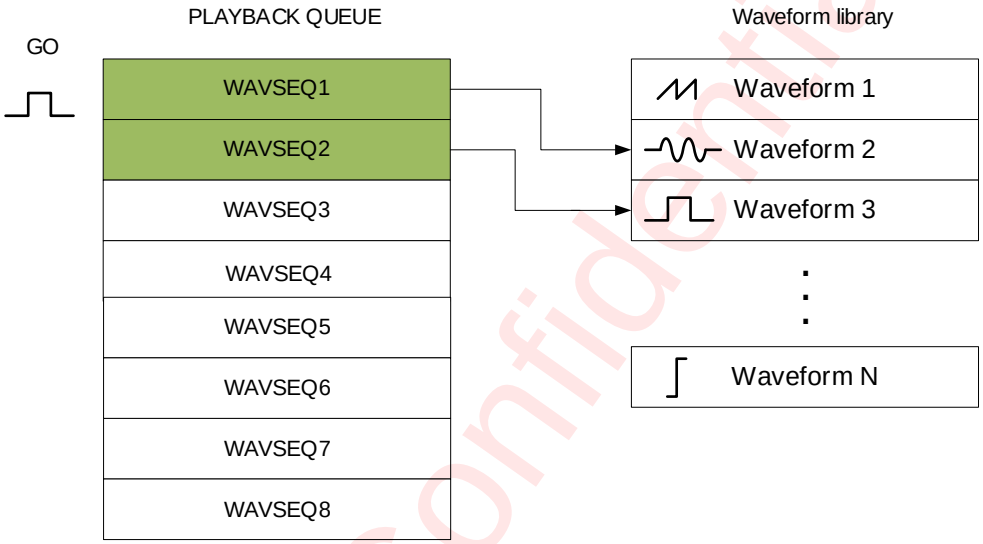



**Figure 16 RAM mode playback**

The waveform playback queue defines waveform IDs in waveform library for playback. Eight WAVSEQx registers queue up to eight library waveforms for sequential playback. A waveform ID is an integer value referring to the index of a waveform in the waveform library. Playback begins at WAVSEQ1 when the user triggers the waveform playback queue. When playback of that waveform ends, the waveform queue plays the next waveform ID held in WAVSEQ2 (if non-zero). The waveform queue continues in this way until the queue reaches an ID value of zero or until all eight IDs are played whichever comes first.

The waveform ID is a 7-bit number. The MSB of each ID register can be used to implement a delay between queue waveforms. When the SEQxWAIT is high, bits 6-0 indicate the length of the wait time. The wait time for that step then becomes WAVSEQ[6:0] × wait_time unit. Wait_time unit can be configuration of WAITSLOT register(in 0x16 register).

The device allows for looping of individual waveforms by using the SEQxLOOP registers. When used, the state machine will loop the particular waveform the number of times specified in the associated SEQxLOOP register before moving to the next waveform. The device allows for looping of the entire playback sequence by using the MAIN_LOOP register. The waveform-looping feature is useful for long, custom haptic playbacks, such as a haptic ringtone.

**Playback steps:**

- Waveform library must be initialized before playback;
- Set PLAY_MODE bit to 0 in register 0x08;
- Set playback queue registers (0x0A ~ 0x11) as desired;

- Set playback loop registers (0x12~ 0x16) as desired;
- Set GO bit to 1 in register 0x09 to trigger waveform playback;
- Device will be switched to STANDBY mode after haptic waveform playback finished.

***Rtp Mode***

The real-time playback mode is a simple, single 8-bit register interface that holds an amplitude value. When real-time playback is enabled, begin to enters a register value to RTP_DATA over the I²C will trigger the playback, the value is played until the data sending finished or removes the device from RTP mode. The maximum FIFO space is 4Kbyte. During writing data to the RTP_DATA register, the I2C speed cannot be too low, see the table below for details.

**Table 2 Minimum I2C speed**

| WAVDAT_MODE(0x44) | Sample rate of waveform | I2C speed |
|---|---|---|
| 2'b10/2'b11 | 12K | ⩾250kHz |
| 2'b00 | 24K | ⩾400kHz |
| 2'b01 | 48K | 1MHz |

After FF_AEM or FF_AFM register is set to 0, HOST can obtain the RTP FIFO almost empty or almost full status through interrupt signal(pin INTN) or read FF_AES or FF_AFS register. RTP FIFO almost empty and almost full threshold can be configured through FIFO_AE and FIFO_AF registers.

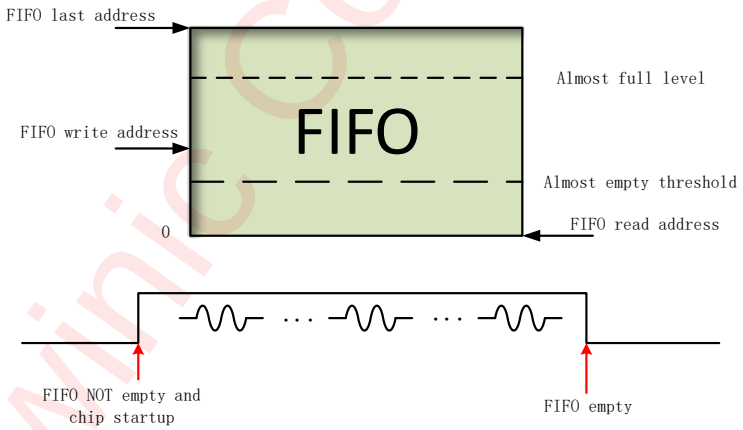



**Figure 17 RTP mode playback**

**Playback steps:**

- Prepare RTP data before playback;
- Set PLAY_MODE bit to 1 in register 0x08;
- Set GO bit to 1 in register 0x09 to trigger waveform playback;
- Delay 1ms.
- Check GLB_STATE=4'b1000, if HOST don't send data to FIFO, chip will wait for RTP data coming in this state forever;
- Write RTP data continually to register 0x32 to playback RTP waveform;

- HOST need monitor the almost full and empty status for RTP FIFO;
- Device will be switched to STANDBY mode after wave data in RTP FIFO is played empty.

***Trig Mode***

The device have three dedicated hardware pins for quickly trigger haptic data playback. Each pin can be configured positive edge/negative edge/both-edge/level trigger.

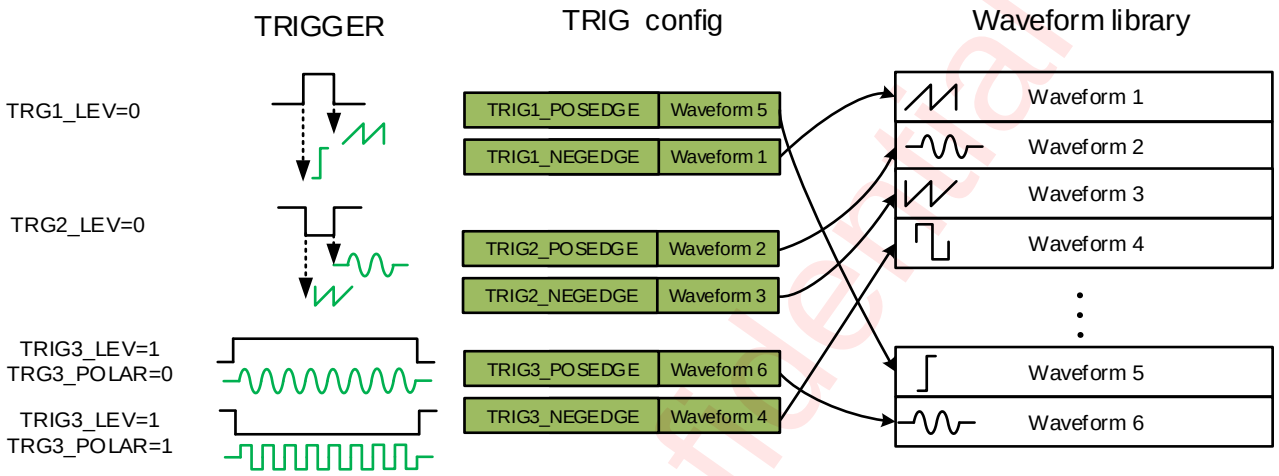



**Figure 18 TRIG mode playback**

Edge mode or level mode is accessible by configing register TRGx_LEV. When a edge mode is needed, user should set TRGx_LEV =0. In edge mode, register TRGxSEQ_P and TRGx_POS respestively represent the waveform and enable signal of positive edge, where register TRGxSEQ_N and TRGx_NEG respestively represent the waveform and enable signal of negative edge.

When a level mode is needed, user should set TRGx_LEV =1, and positive level and negative level can be supported by setting register TRGX_POLAR=0 and setting TRGX_POLAR=1.

**Table 3 TRIG MODE CONFIG**

| I2C reg | | | | Trigger | Waveform |
|---|---|---|---|---|---|
| TRGx_LVL | TRGx_POLAR | TRGx_POS | TRGx_NEG | | |
| 0 | X | 0 | 0 | - | none |
| | X | 1 | 0 | ↑ | TRGxSEQ_P |
| | X | 0 | 1 | ↓ | TRGxSEQ_N |
| | X | 1 | 1 | ↑ / ↓ | TRGxSEQ_P/ TRGxSEQ_N |
| 1 | 0 | X | X | High level | TRGxSEQ_P |
| | 1 | X | X | Low level | TRGxSEQ_N |

**Playback steps:**

- Waveform library must be initialized before playback;
- Set 0x39's bit4 & bit1 to 0;
- Set 0x3A's bit4 to 0;
- Set trigger playback registers (0x33 ~ 0x3A) as desired;
- Send trigger pulse ($\geq 1\mu s$) or trigger level ($\geq 1ms$) on TRIG pins to playback waveform;
- Device will be switched to STANDBY mode after haptic waveform playback finished.

***One wire Mode***

The function of one wire mode mainly transfer two information : sequence number and gain of waveform，TRIG1 is the interface pin.

**Playback steps:**

- Waveform library must be initialized before playback;
- Set TRG_ONEWIRE to 1 in register 0x3A to enable one wire mode;
- Set 0x39's bit4 & bit1 to 0;
- Set 0x3A's bit4 to 0;
- Set 0x3E = 0x58 to enable one wire the lowest priority of playback;
- Set 0x3C = 0x75, configure this register above this value in order to receive one wire protocol（about 2.5ms which is the minimum waiting time after chip wakeup，default value is 20μs）;
- Determine sequence number and gain of waveform which you want to playback;
- Combine sequence number and gain data into a 15 bit transformation data（low 8 bit is gain, high 7 bit is sequence number），the data is sent from the lowest bit;
- Chip will automatically enter standby mode after playing. The interval time between two sending protocol data should be greater than "3ms+time length of waveform".

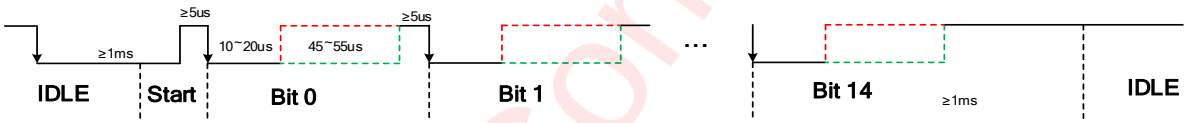



**Figure 19 One wire mode playback**

***Cont Mode***

The CONT mode mainly performs two functions: F0 detection and real-time resonance-frequency tracking. F0 detection can be launched by setting EN_F0_DET=1 and BRK_EN =1. When set TRACK_EN=1, real-time resonance-frequency tracking will be launched by tracking the BEMF of actuator constantly. It provides stronger and more consistent vibrations and lower power consumption. If the resonant frequency shifts for any reason, the function tracks the frequency from cycle to cycle. When TRACK_EN is set to 0, the width of waveform of cont mode is determined by DRV_WIDTH in register 0x1A.

When the EDGE_FRE register is set to 4'b1xxx, the CONT mode outputs a filtered square wave. The edge of filtered square wave is composed of SIN or COS wave whose frequency can be configured by EDGE_FRE register. When SIN_MODE register is set to 1, filtered square wave is composed of COS wave.

**Playback steps:**

- Set PLAY_MODE = 2 in register 0x08 to enable CONT mode ;
- (optional)Set EN_F0_DET = 1 and BRK_EN =1 to enable F0 detection;
- Set cont mode by configuring registers(0x18~0x20 and 0x22);
- Set GO bit to 1 in register 0x09 to trigger waveform playback;
- Delay 1ms;
- If enable F0 detection, read until GLB_STATE=0. then get F0 information from registers(0x25~0x28).

- Device will be switched to STANDBY mode after haptic waveform playback finished;

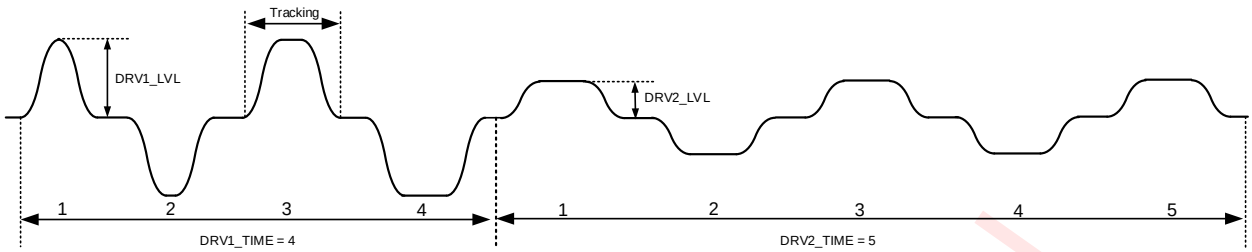



**Figure 20 Cont mode playback**

***Auto Brake Engine***

An auto-brake engine is integrated into this device. Users can adjust the brake strength by setting D2S_GAIN in register 0x49. The greater D2S_GAIN, the greater brake strength and the worse loop stability. Auto-brake engine is disabled when setting BRK_EN=0 or BRK_TIME=0.

**To enable Auto-brake engine, there are some points to note:**

- TRGx_BRK in register 0x39,0x3A should be set to 1 when in TRIG mode;
- Auto-brake engine will not work when BRK_EN=0 in register 0x08；
- Auto-brake engine will not work when EN_F0_DET in register 0x18 is set to 1;
- Auto-brake engine will not work when BRK_TIME in register 0x21 is set to 0;
- Device will be switched to STANDBY mode after haptic waveform playback finished.

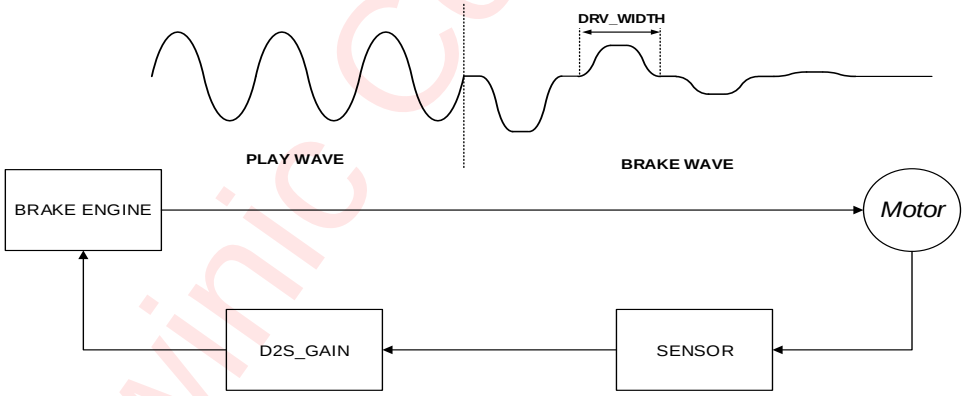



**Figure 21 Brake loop**

## Protection Mechanisms

***Over Temperature Protection (OTP)***

The device has automatic temperature protection mechanism which prevents heat damage to the chip. It is triggered when the junction temperature is larger than the preset temperature high threshold (default = 160°C). When it happens, the output stages will be disabled. When the junction temperature drops below the preset temperature low threshold (less than 130°C), the output stages will start to operate normally again

### *Over Current (Short) Protection (OCP)*

The short circuit protection function is triggered when HDP/HDN is short too PVDD/GND or HDP is short to HDN, the output stages will be shut down to prevent damage to itself. When the fault condition is disappeared, the output stages of device will restart.

### *Vbat Under Voltage Lock Out Protection (UVLO)*

The device has a battery monitor that monitors the VBAT level to ensure that is above threshold 2.7V, In the event of a VBAT drop, the device immediately power down the H-bridge driver and latches the UVLO flag.

### *Drive Data Error Protection (DDEP)*

When haptic data sent to drive LRA is error such as: a DC data or almost DC data, it will cause the LRA heat to brake. The device configurable immediately power down the H-bridge driver and latched the DDEP flag.

## I²C Interface

This device supports the I²C serial bus and data transmission protocol in fast mode at 400kHz and fast mode plus at 1000kHz. This device operates as a slave on the I²C bus. Connections to the bus are made via the open-drain I/O pins SCL and I pin SDA. The pull-up resistor can be selected in the range of 1k~10kΩ and the typical value is 4.7kΩ. This device can support different high level (1.8V~3.3V) of this I²C interface.

### *Device Address*

The I²C device address (7-bit) can be set using the AD pin according to the following table:

**Table 4 Address Selection**

| AD | I²C address (7-bit) |
|---|---|
| 0 | 0x5A |
| 1 | 0x5B |

### *Data Validation*

When SCL is high level, SDA level must be constant. SDA can be changed only when SCL is low level.

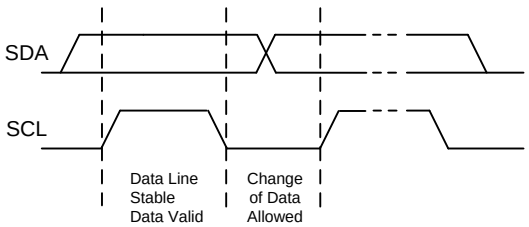



**Figure 22 Data Validation Diagram**

### *General I²C Operation*

The I²C bus employs two signals, SDA (data) and SCL (clock), to communicate between integrated circuits in a system. The device is addressed by a unique 7-bit address; the same device can send and receive data. In

addition, Communications equipment has distinguish master from slave device: In the communication process, only the master device can initiate a transfer and terminate data and generate a corresponding clock signal. The devices using the address access during transmission can be seen as a slave device.

SDA and SCL connect to the power supply through the current source or pull-up resistor. SDA and SCL default is a high level. There is no limit on the number of bytes that can be transmitted between start and stop conditions. When the last word transfers, the master generates a stop condition to release the bus.

START state: The SCL maintain a high level, SDA from high to low level

STOP state: The SCL maintain a high level, SDA pulled low to high level

Start and Stop states can be only generated by the master device. In addition, if the device does not produce STOP state after the data transmission is completed, instead re-generate a START state (Repeated START, Sr), and it is believed that this bus is still in the process of data transmission. Functionally, Sr state and START state is the same. As shown in Figure 23.

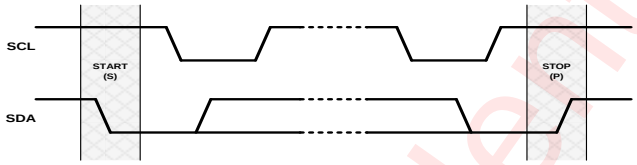



**Figure 23 START and STOP state generation process**

In the data transmission process, when the clock line SCL maintains a high level, the data line SDA must remain the same. Only when the SCL maintain a low level, the data line SDA can be changed, as shown in Figure 24. Each transmission of information on the SDA is 9 bits as a unit. The first eight bits are the data to be transmitted, and the first one is the most significant bit (Most Significant Bit, MSB), the ninth bit is an confirmation bit (Acknowledge, ACK or A ), as shown in Figure 25. When the SDA transmits a low level in ninth clock pulse, it means the acknowledgment bit is 1, namely the current transmission of 8 bits data are confirmed, otherwise it means that the data transmission has not been confirmed. Any amount of data can be transferred between START and STOP state.

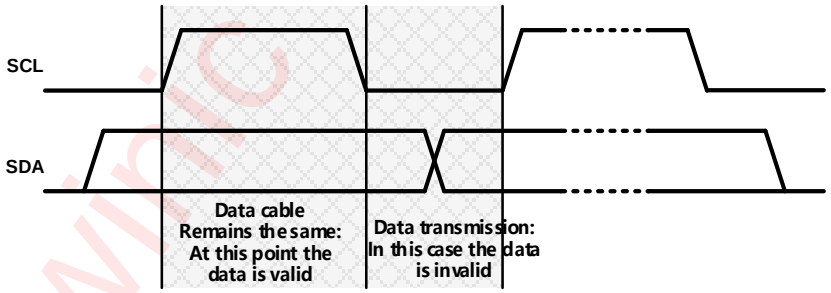



**Figure 24 The data transfer rules on the I²C bus**

The whole process of actual data transmission is shown in Figure 25. When generating a START condition, the master device sends an 8-bit data, including a 7-bit slave addresses (Slave Address), and followed by a "read / write" flag ($R/\overline{W}$). The flag is used to specify the direction of transmission of subsequent data. The master device will produce the STOP state to end the process after the data transmission is completed. However, if the master device intends to continue data transmission, you can directly send a Repeated START state, without the need to use the STOP state to end transmission.

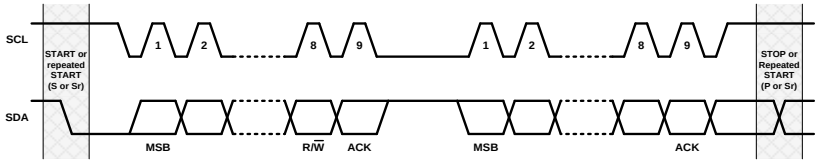



**Figure 25 Data transmission on the I²C bus**

***Write Process***

Writing process refers to the master device write data into the slave device. In this process, the transfer direction of the data is always unchanged from the master device to the slave device. All acknowledge bits are transferred by the slave device, in particular, the device as the slave device, the transmission process in accordance with the following steps, as shown in Figure 26:

Master device generates START state. The START state is produced by pulling the data line SDA to a low level when the clock SCL signal is a high level.

Master device transmits the 7-bits device address of the slave device, followed by the "read / write" flag (flag $R/\overline{W}$ = 0);

The slave device asserts an acknowledgment bit (ACK) to confirm whether the device address is correct;

The master device transmits the 8-bit register address to which the first data byte will written；

The slave device asserts an acknowledgment (ACK) bit to confirm the register address is correct;

Master sends 8 bits of data to register which needs to be written;

The slave device asserts an acknowledgment bit (ACK) to confirm whether the data is sent successfully;

If the master device needs to continue transmitting data by sending another pair of data bytes, just need to repeat the sequence from step 6. In the latter case, the targeted register address will have been auto-incremented by the device.

The master device generates the STOP state to end the data transmission.

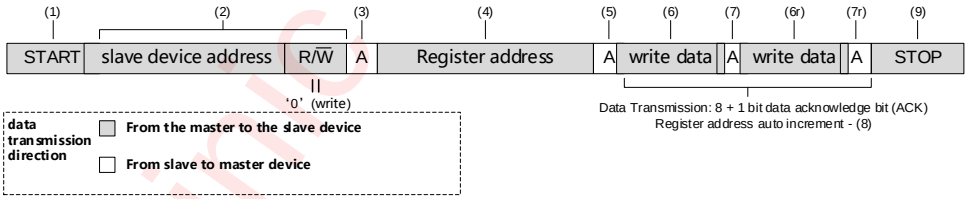



**Figure 26 Writing process (data transmission direction remains the same)**

***Read Process***

Reading process refers to the slave device reading data back to the master device. In this process, the direction of data transmission will change. Before and after the change, the master device sends START state and slave address twice, and sends the opposite "read/write" flag. In particular, AW86205QNR-Q1 as the slave device, the transmission process carried out by following steps listed in Figure 27:

Master device asserts a start condition;

Master device transmits the 7 bits address of the device, and followed by a "read / write" flag ( $R/\overline{W}$ = 0);

The slave device asserts an acknowledgment bit (ACK) to confirm whether the device address is correct;

The master device transmits the register address to make sure where the first data byte will read;

The slave device asserts an acknowledgment (ACK) bit to confirm whether the register address is correct or not;

The master device restarts the data transfer process by continuously generating STOP state and START state or a separate Repeated START;

Master sends 7-bits address of the slave device and followed by a read / write flag (flag $R/\overline{W}$ = 1) again;
The slave device asserts an acknowledgment (ACK) bit to confirm whether the register address is correct or not;
Master transmits 8 bits of data to register which needs to be read;
The slave device sends an acknowledgment bit (ACK) to confirm whether the data is sent successfully;
The device automatically increment register address once after sent each acknowledge bit (ACK),
The master device generates the STOP state to end the data transmission.

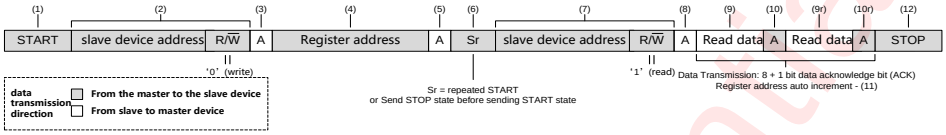



**Figure 27 Reading process (data transmission direction remains the same)**

# Register Configuration

## Register List

| ADDR | NAME | R/W | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | Default |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | SRST | RO | CHIP_ID | | | | | | | | 0x04 |
| 0x01 | SYSST | RO | | | UVLS | FF_AES | FF_AFS | OCDS | OTS | DONES | 0x10 |
| 0x02 | SYSINT | RC | | | UVLI | FF_AEI | FF_AFI | OCDI | OTI | DONEI | 0x10 |
| 0x03 | SYSINTM | RW | | | UVLM | FF_AEM | FF_AFM | OCDM | OTM | DONEM | 0xFF |
| 0x07 | PLAYCFG2 | RW | GAIN | | | | | | | | 0x80 |
| 0x08 | PLAYCFG3 | RW | | | STOP_MODE | | | BRK_EN | PLAY_MODE | | 0x44 |
| 0x09 | PLAYCFG4 | RW | | | | | | | STOP | GO | 0x00 |
| 0x0A | WAVCFG1 | RW | SEQ1WAIT | WAVSEQ1 | | | | | | | 0x01 |
| 0x0B | WAVCFG2 | RW | SEQ2WAIT | WAVSEQ2 | | | | | | | 0x00 |
| 0x0C | WAVCFG3 | RW | SEQ3WAIT | WAVSEQ3 | | | | | | | 0x00 |
| 0x0D | WAVCFG4 | RW | SEQ4WAIT | WAVSEQ4 | | | | | | | 0x00 |
| 0x0E | WAVCFG5 | RW | SEQ5WAIT | WAVSEQ5 | | | | | | | 0x00 |
| 0x0F | WAVCFG6 | RW | SEQ6WAIT | WAVSEQ6 | | | | | | | 0x00 |
| 0x10 | WAVCFG7 | RW | SEQ7WAIT | WAVSEQ7 | | | | | | | 0x00 |
| 0x11 | WAVCFG8 | RW | SEQ8WAIT | WAVSEQ8 | | | | | | | 0x00 |
| 0x12 | WAVCFG9 | RW | SEQ1LOOP | | | | SEQ2LOOP | | | | 0x00 |
| 0x13 | WAVCFG10 | RW | SEQ3LOOP | | | | SEQ4LOOP | | | | 0x00 |
| 0x14 | WAVCFG11 | RW | SEQ5LOOP | | | | SEQ6LOOP | | | | 0x00 |
| 0x15 | WAVCFG12 | RW | SEQ7LOOP | | | | SEQ8LOOP | | | | 0x00 |
| 0x16 | WAVCFG13 | RW | | WAITSLOT | | | MAINLOOP | | | | 0x00 |
| 0x18 | CONTCFG1 | RW | EDGE_FRE | | | | EN_F0_DET | | | SIN_MODE | 0xD1 |
| 0x19 | CONTCFG2 | RW | F_PRE | | | | | | | | 0x8D |
| 0x1A | CONTCFG3 | RW | DRV_WIDTH | | | | | | | | 0x6A |
| 0x1C | CONTCFG5 | RW | | | | | BRK_GAIN | | | | 0x58 |
| 0x1D | CONTCFG6 | RW | TRACK_EN | DRV1_LVL | | | | | | | 0xFF |
| 0x1E | CONTCFG7 | RW | | DRV2_LVL | | | | | | | 0x50 |
| 0x1F | CONTCFG8 | RW | DRV1_TIME | | | | | | | | 0x04 |
| 0x20 | CONTCFG9 | RW | DRV2_TIME | | | | | | | | 0x06 |
| 0x21 | CONTCFG10 | RW | BRK_TIME | | | | | | | | 0x08 |
| 0x22 | CONTCFG11 | RW | TRACK_MARGIN | | | | | | | | 0x0C |
| 0x25 | CONTRD14 | RO | F_LRA_F0_H | | | | | | | | 0x00 |
| 0x26 | CONTRD15 | RO | F_LRA_F0_L | | | | | | | | 0x00 |
| 0x27 | CONTRD16 | RO | CONT_F0_H | | | | | | | | 0x00 |
| 0x28 | CONTRD17 | RO | CONT_F0_L | | | | | | | | 0x00 |
| 0x2D | RTPCFG1 | RW | | | BASE_ADDR_H | | | | | | 0x08 |
| 0x2E | RTPCFG2 | RW | BASE_ADDR_L | | | | | | | | 0x00 |
| 0x2F | RTPCFG3 | RW | FIFO_AEH | | | | FIFO_AFH | | | | 0x26 |
| 0x30 | RTPCFG4 | RW | FIFO_AEL | | | | | | | | 0x00 |
| 0x31 | RTPCFG5 | RW | FIFO_AFL | | | | | | | | 0x00 |
| 0x32 | RTPDATA | RW | RTP_DATA | | | | | | | | 0x00 |
| 0x33 | TRGCFG1 | RW | TRG1_POS | TRG1SEQ_P | | | | | | | 0x01 |
| 0x34 | TRGCFG2 | RW | TRG2_POS | TRG2SEQ_P | | | | | | | 0x01 |
| 0x35 | TRGCFG3 | RW | TRG3_POS | TRG3SEQ_P | | | | | | | 0x01 |
| 0x36 | TRGCFG4 | RW | TRG1_NEG | TRG1SEQ_N | | | | | | | 0x01 |
| 0x37 | TRGCFG5 | RW | TRG2_NEG | TRG2SEQ_N | | | | | | | 0x01 |
| 0x38 | TRGCFG6 | RW | TRG3_NEG | TRG3SEQ_N | | | | | | | 0x01 |
| 0x39 | TRGCFG7 | RW | TRG1_POLAR | TRG1_LEV | TRG1_BRK | | TRG2_POLAR | TRG2_LEV | TRG2_BRK | | 0x33 |
| 0x3A | TRGCFG8 | RW | TRG3_POLAR | TRG3_LEV | TRG3_BRK | | TRG_ONEWIRE | TRG1_STOP | TRG2_STOP | TRG3_STOP | 0x30 |
| 0x3C | GLBCFG2 | RW | START_DLY | | | | | | | | 0x01 |
| 0x3E | GLBCFG4 | RW | GO_PRIO | | TRG3_PRIO | | TRG2_PRIO | | TRG1_PRIO | | 0x1B |
| 0x3F | GLBRD5 | RO | | | | | GLB_STATE | | | | 0x00 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0x40 | RAMADDRH | ***RWS*** | | RAMADDRH | | | | 0x00 |
| 0x41 | RAMADDRL | ***RWS*** | RAMADDRL | | | | | 0x00 |
| 0x42 | RAMDATA | ***RWS*** | RAMDATA | | | | | 0x00 |
| 0x43 | SYSCTRL1 | ***RW*** | VBAT_MODE | | EN_RAMINIT | EN_FIR | | 0x44 |
| 0x44 | SYSCTRL2 | ***RW*** | WAKE | STANDBY | | | WAVDAT_MODE | 0x20 |
| 0x49 | SYSCTRL7 | ***RW*** | | GAIN_BYPASS | | D2S_GAIN | | 0x04 |
| 0x4C | PWMCFG1 | ***RW*** | PRC_EN | PRCTIME | | | | 0xA0 |
| 0x4D | PWMCFG2 | ***RW*** | | PD_HWM | | | | 0x28 |
| 0x4E | PWMCFG3 | ***RW*** | PR_EN | PRLVL | | | | 0xBF |
| 0x4F | PWMCFG4 | ***RW*** | PRTIME | | | | | 0x32 |
| 0x51 | DETCFG1 | ***RW*** | | RL | | CLK_ADC | | 0x02 |
| 0x52 | DETCFG2 | ***RW*** | | | | VBAT_GO | DIAG_GO | 0x00 |
| 0x53 | DET_RL | ***RO*** | RL_H | | | | | 0x00 |
| 0x55 | DET_VBAT | ***RO*** | VBAT_H | | | | | 0x00 |
| 0x57 | DET_LO | ***RO*** | | VBAT_LO | | | RL_LO | 0x00 |

## Register Detailed Description

*Note: Reserved register should not be written*

SRST: (Address 00h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | CHIPID | RO | All configuration registers will be reset to default value after 0xaa is written | 0x04 |

SYSST: (Address 01h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:6 | Reserved | RO | Not used | 0 |
| 5 | UVLS | RO | 1: VBAT voltage is under UV voltage (2.7V) | 0 |
| 4 | FF_AES | RO | 1: RTP FIFO is almost empty | 1 |
| 3 | FF_AFS | RO | 1: RTP FIFO is almost full | 0 |
| 2 | OCDS | RO | 1: Over Current status | 0 |
| 1 | OTS | RO | 1: Over Temperature status | 0 |
| 0 | DONES | RO | 1: The indication of playback finished | 0 |

SYSINT: (Address 02h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:6 | Reserved | RC | Not used | 0 |
| 5 | UVLI | RC | When UVLI=1, it means UVLS has been 1 at least once since the last read | 0 |
| 4 | FF_AEI | RC | When FF_AEI=1, it means FF_AES has been 1 at least once since the last read | 1 |
| 3 | FF_AFI | RC | When FF_AFI=1, it means FF_AFS has been 1 at least once since the last read | 0 |
| 2 | OCDI | RC | When OCDI=1, it means OCDS has been 1 at least once since the last read | 0 |
| 1 | OTI | RC | When OTI=1, it means OTS has been 1 at least once since the last read | 0 |
| 0 | DONEI | RC | When DONEI=1, it means DONES has been 1 at least once since the last read | 0 |

SYSINTM: (Address 03h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:6 | Reserved | RW | Not used | 3 |
| 5 | UVLM | RW | Interrupt mask for UVLI:<br>0: INTN pin will be pulled down when UVLI=1<br>1: INTN pin will not be pulled down when UVLI=1 | 1 |

| | | | | |
|---|---|---|---|---|
| 4 | FF_AEM | RW | Interrupt mask for FF_AEI:<br>0: INTN pin will be pulled down when FF_AEI=1<br>1: INTN pin will not be pulled down when FF_AEI=1 | 1 |
| 3 | FF_AFM | RW | Interrupt mask for FF_AFI:<br>0: INTN pin will be pulled down when FF_AFI=1<br>1: INTN pin will not be pulled down when FF_AFI=1 | 1 |
| 2 | OCDM | RW | Interrupt mask for OCDI:<br>0: INTN pin will be pulled down when OCDI=1<br>1: INTN pin will not be pulled down when OCDI=1 | 1 |
| 1 | OTM | RW | Interrupt mask for OTI:<br>0: INTN pin will be pulled down when OTI=1<br>1: INTN pin will not be pulled down when OTI=1 | 1 |
| 0 | DONEM | RW | Interrupt mask for DONEI:<br>0: INTN pin will be pulled down when DONEI=1<br>1: INTN pin will not be pulled down when DONEI=1 | 1 |

PLAYCFG2: (Address 07h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | GAIN | RW | gain setting for waveform data, it is a global setting for all waveform data(expect CONT).GAIN=code/128 | 0x80 |

PLAYCFG3: (Address 08h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:6 | Reserved | RW | Not used | 1 |
| 5 | STOP_MODE | RW | 0 : stop when current wave is over<br>1: stop right now | 0 |
| 4:3 | Reserved | RW | Not used | 0 |
| 2 | BRK_EN | RW | when set 1, enable auto brake after RTP/RAM/CONT playback mode is stopped | 1 |
| 1:0 | PLAY_MODE | RW | waveform play mode for GO trig<br>b00: RAM mode<br>b01: RTP mode<br>b10: CONT mode<br>b11: no play | 0 |

PLAYCFG4: (Address 09h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:2 | Reserved | RW | Not used | 0 |
| 1 | STOP | RW | when set 1, stop the current playback mode | 0 |
| 0 | GO | RW | RAM/RTP/CONT mode playback trig bit<br>when set to 1, chip will playback one of the play mode. | 0 |

WAVCFG1: (Address 0Ah)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ1WAIT | RW | when set to 1 , WAVSEQ1 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ1 | RW | wait time (code*WAITSLOT) or wave sequence number | 1 |

WAVCFG2: (Address 0Bh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ2WAIT | RW | when set to 1 , WAVSEQ2 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ2 | RW | wait time (code*WAITSLOT) or wave sequence number | 0 |

WAVCFG3: (Address 0Ch)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ3WAIT | RW | when set to 1 , WAVSEQ3 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ3 | RW | wait time (code*WAITSLOT) or wave sequence number | 0 |

WAVCFG4: (Address 0Dh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ4WAIT | RW | when set to 1 , WAVSEQ4 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ4 | RW | wait time (code*WAITSLOT) or wave sequence number | 0 |

WAVCFG5: (Address 0Eh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ5WAIT | RW | when set to 1 , WAVSEQ5 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ5 | RW | wait time (code*WAITSLOT) or wave sequence number | 0 |

WAVCFG6: (Address 0Fh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ6WAIT | RW | when set to 1 , WAVSEQ6 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ6 | RW | wait time (code*WAITSLOT) or wave sequence number | 0 |

WAVCFG7: (Address 10h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ7WAIT | RW | when set to 1 , WAVSEQ7 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ7 | RW | wait time (code*WAITSLOT) or wave sequence number | 0 |

WAVCFG8: (Address 11h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | SEQ8WAIT | RW | when set to 1 , WAVSEQ8 means wait time, else means wave sequence number | 0 |
| 6:0 | WAVSEQ8 | RW | wait time (code*WAITSLOT) or wave sequence number | 0 |

WAVCFG9: (Address 12h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:4 | SEQ1LOOP | RW | control the loop number of the first sequence<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or SEQ1LOOP ≠0xF | 0 |
| 3:0 | SEQ2LOOP | RW | control the loop number of the second sequence<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or SEQ2LOOP ≠0xF | 0 |

WAVCFG10: (Address 13h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:4 | SEQ3LOOP | RW | control the loop number of the third sequence<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or SEQ3LOOP ≠0xF | 0 |
| 3:0 | SEQ4LOOP | RW | control the loop number of the fourth sequence<br>b0000~1110: play n+1 time<br>b1111: playback infinitely until STOP set to 1 or SEQ4LOOP ≠0xF | 0 |

| WAVCFG11: (Address 14h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:4 | SEQ5LOOP | RW | control the loop number of the fifth sequence<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or SEQ5LOOP ≠0xF | 0 |
| 3:0 | SEQ6LOOP | RW | control the loop number of the sixth sequence<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or SEQ6LOOP ≠0xF | 0 |

| WAVCFG12: (Address 15h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:4 | SEQ7LOOP | RW | control the loop number of the seventh sequence<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or SEQ7LOOP ≠0xF | 0 |
| 3:0 | SEQ8LOOP | RW | control the loop number of the eighth sequence<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or SEQ8LOOP ≠0xF | 0 |

| WAVCFG13: (Address 16h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7 | Reserved | RW | Not used | 0 |
| 6:5 | WAITSLOT | RW | unit of wait time<br>b00: (1/WAVDAT_MODE) s<br>b01: (8/WAVDAT_MODE) s<br>b10: (64/WAVDAT_MODE) s<br>b11: (512/WAVDAT_MODE) s | 0 |
| 4 | Reserved | RW | Not used | 0 |
| 3:0 | MAINLOOP | RW | control the main loop number<br>b0000~b1110: play (code+1) time<br>b1111: playback infinitely until STOP set to 1 or MAINLOOP ≠0xF | 0 |

| CONTCFG1: (Address 18h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:4 | EDGE_FRE | RW | define the edge frequency<br>b1000 : 200Hz<br>b1001 : 300Hz<br>b1010 : 400Hz<br>b1011 : 500Hz<br>b1100 : 600Hz<br>b1101 : 700Hz<br>b1110 : 800Hz<br>b1111 : 900Hz<br>b0XXX : no edge control | 13 |
| 3 | EN_F0_DET | RW | f0 detection mode enable<br>1: enable<br>0: disable | 0 |
| 2:1 | Reserved | RW | Not used | 0 |
| 0 | SIN_MODE | RW | edge mode<br>1: cos<br>0: sine | 1 |

| CONTCFG2: (Address 19h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |

| | | | | |
|---|---|---|---|---|
| 7:0 | F_PRE | RW | set the value of F0, F0=(24K/code)Hz | 0x8D |

CONTCFG3: (Address 1Ah)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | DRV_WIDTH | RW | Half cycle drive time of brake and it is also the half cycle drive time of drive when TRACK_EN=0, this value must be smaller than half cycle time of F0. Time = code/48000 (s). | 0x6A |

CONTCFG5: (Address 1Ch)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:4 | Reserved | RW | Not used | 5 |
| 3:0 | BRK_GAIN | RW | gain factor of brake | 8 |

CONTCFG6: (Address 1Dh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRACK_EN | RW | track switch<br>1: enable<br>0: disable | 1 |
| 6:0 | DRV1_LVL | RW | level for the first cont drive.<br>When VBAT_MODE=1:<br>no load output voltage= (3.05*DRV1_LVL/128)*(PVDD/VBAT);<br>if (3.05*DRV1_LVL)/VBAT > 128, no load output voltage=PVDD;<br>When VBAT_MODE=0:<br>no load output voltage=PVDD*DRV1_LVL/128 | 0x7F |

CONTCFG7: (Address 1Eh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | Reserved | RW | Not used | 0 |
| 6:0 | DRV2_LVL | RW | level for the second cont drive.<br>When VBAT_MODE=1:<br>no load output voltage= (3.05*DRV2_LVL/128)* (PVDD/VBAT);<br>if (3.05*DRV2_LVL)/VBAT > 128, no load output voltage=PVDD;<br>When VBAT_MODE=0:<br>no load output voltage=PVDD*DRV2_LVL/128 | 0x50 |

CONTCFG8: (Address 1Fh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | DRV1_TIME | RW | number of half cycle for the first cont drive | 4 |

CONTCFG9: (Address 20h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | DRV2_TIME | RW | number of half cycle for the second cont drive. | 6 |

CONTCFG10: (Address 21h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | BRK_TIME | RW | the num of half cycle of brake mode | 8 |

CONTCFG11: (Address 22h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | TRACK_MARGIN | RW | margin value of tracking, the smaller margin, the higher tracking accuracy and the lower loop stability. Time = code/480000 (s) | 12 |

CONTRD14: (Address 25h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | F_LRA_F0_H | RO | high 8 bit of the measure value for the f0 of LRA in the f0 detection mode F0=(384000/(F_LRA_F0_H*256+F_LRA_F0_L))Hz | 0 |

| CONTRD15: (Address 26h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:0 | F_LRA_F0_L | RO | low 8 bit of the measure value for the f0 of LRA in the f0 detection mode F0=(384000/(F_LRA_F0_H*256+F_LRA_F0_L))Hz | 0 |

| CONTRD16: (Address 27h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:0 | CONT_F0_H | RO | the measure value for the f0 of LRA in the continuous detection mode(high eight bits). F0=(384000/(CONT_F0_H*256+CONT_F0_L))Hz | 0 |

| CONTRD17: (Address 28h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:0 | CONT_F0_L | RO | the measure value for the f0 of LRA in the continuous detection mode(low eight bits). F0=(384000/(CONT_F0_H*256+CONT_F0_L))Hz | 0 |

| RTPCFG1: (Address 2Dh) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:6 | Reserved | RW | Not used | 0 |
| 5:0 | BASE_ADDR_H | RW | High six bits of start address of wave SRAM<br>BASE_ADDR = BASE_ADDR_H * 256 + BASE_ADDR_L | 0x08 |

| RTPCFG2: (Address 2Eh) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:0 | BASE_ADDR_L | RW | Low eight bits of start address of wave SRAM<br>BASE_ADDR = BASE_ADDR_H * 256 + BASE_ADDR_L | 0 |

| RTPCFG3: (Address 2Fh) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:4 | FIFO_AEH | RW | High four bits of RTP FIFO almost empty threshold<br>FIFO_AE = FIFO_AEH * 256 + FIFO_AEL | 0x02 |
| 3:0 | FIFO_AFH | RW | High four bits of RTP FIFO almost full threshold<br>FIFO_AF = FIFO_AFH * 256 + FIFO_AFL | 0x06 |

| RTPCFG4: (Address 30h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:0 | FIFO_AEL | RW | Low eight bits of RTP FIFO almost empty threshold<br>FIFO_AE = FIFO_AEH * 256 + FIFO_AEL | 0x00 |

| RTPCFG5: (Address 31h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:0 | FIFO_AFL | RW | Low eight bits of RTP FIFO almost full threshold<br>FIFO_AF = FIFO_AFH * 256 + FIFO_AFL | 0x00 |

| RTPDATA: (Address 32h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:0 | RTP_DATA | RW | RTP mode , data write entry, when data written into this register, the data will be written into RTP FIFO | 0 |

| TRGCFG1: (Address 33h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |

| | | | | |
|---|---|---|---|---|
| 7 | TRG1_POS | RW | trg1 rising edge enable/disable control<br>1: enable<br>0: disable | 0 |
| 6:0 | TRG1SEQ_P | RW | TRG1 positive edge trigged wave sequence number | 1 |

TRGCFG2: (Address 34h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRG2_POS | RW | trg2 rising edge enable/disable control<br>1: enable<br>0: disable | 0 |
| 6:0 | TRG2SEQ_P | RW | TRG2 positive edge trigged wave sequence number | 1 |

TRGCFG3: (Address 35h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRG3_POS | RW | trg3 rising edge enable/disable control<br>1: enable<br>0: disable | 0 |
| 6:0 | TRG3SEQ_P | RW | TRG3 positive edge trigged wave sequence number | 1 |

TRGCFG4: (Address 36h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRG1_NEG | RW | trg1 falling edge enable/disable control<br>1: enable<br>0: disable | 0 |
| 6:0 | TRG1SEQ_N | RW | TRG1 negative edge trigged wave sequence number | 1 |

TRGCFG5: (Address 37h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRG2_NEG | RW | trg2 falling edge enable/disable control<br>1: enable 0: disable | 0 |
| 6:0 | TRG2SEQ_N | RW | TRG2 negative edge trigged wave sequence number | 1 |

TRGCFG6: (Address 38h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRG3_NEG | RW | trg3 falling edge enable/disable control<br>1: enable 0: disable | 0 |
| 6:0 | TRG3SEQ_N | RW | TRG3 negative edge trigged wave sequence number | 1 |

TRGCFG7: (Address 39h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRG1_POLAR | RW | TRIG1 pin active polarity, when host supply positive level, this bit set to 0, else set to 1 | 0 |
| 6 | TRG1_LEV | RW | TRG1 mode control<br>1: level<br>0: edge | 0 |
| 5 | TRG1_BRK | RW | when set 1, enable auto brake after TRG1 playback mode is stopped | 1 |
| 4 | Reserved | RW | Not used | 1 |
| 3 | TRG2_POLAR | RW | TRIG2 pin active polarity, when host supply positive level, this bit set to 0, else set to 1 | 0 |
| 2 | TRG2_LEV | RW | TRG2 mode control<br>1: level<br>0: edge | 0 |
| 1 | TRG2_BRK | RW | when set 1, enable auto brake after TRG2 playback mode is stopped | 1 |

| | | | | |
|---|---|---|---|---|
| 0 | Reserved | RW | Not used | 1 |

TRGCFG8: (Address 3Ah)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | TRG3_POLAR | RW | TRIG3 pin active polarity, when host supply positive level, this bit set to 0, else set to 1 | 0 |
| 6 | TRG3_LEV | RW | TRG3 mode control<br>1: level<br>0: edge | 0 |
| 5 | TRG3_BRK | RW | when set 1, enable auto brake after TRG3 playback mode is stopped | 1 |
| 4 | Reserved | RW | Not used | 1 |
| 3 | TRG_ONEWIRE | RW | when set 1,enable one wire mode. | 0 |
| 2 | TRG1_STOP | RW | when set 1, TRG1 playback mode can be stopped immediately | 0 |
| 1 | TRG2_STOP | RW | when set 1, TRG2 playback mode can be stopped immediately | 0 |
| 0 | TRG3_STOP | RW | when set 1, TRG3 playback mode can be stopped immediately | 0 |

GLBCFG2: (Address 3Ch)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | START_DLY | RW | Startup delay time, unit time is (1/48k)s. | 0x01 |

GLBCFG4: (Address 3Eh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:6 | GO_PRIO | RW | Priority value of GO TRIG<br>High priority can interrupt the playback of low priority, and low priority cannot interrupt the playback of high priority. When the priority settings are consistent, the default priority will be implemented | 0 |
| 5:4 | TRG3_PRIO | RW | Priority value of TRIG3 pin<br>High priority can interrupt the playback of low priority, and low priority cannot interrupt the playback of high priority. When the priority settings are consistent, the default priority will be implemented | 1 |
| 3:2 | TRG2_PRIO | RW | Priority value of TRIG2 pin<br>High priority can interrupt the playback of low priority, and low priority cannot interrupt the playback of high priority. When the priority settings are consistent, the default priority will be implemented | 2 |
| 1:0 | TRG1_PRIO | RW | Priority value of TRIG1 pin<br>High priority can interrupt the playback of low priority, and low priority cannot interrupt the playback of high priority. When the priority settings are consistent, the default priority will be implemented | 3 |

GLBRD5: (Address 3Fh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:4 | Reserved | RO | Not used | 0 |
| 3:0 | GLB_STATE | RO | The state of glb state<br>0000: STANDBY<br>0110: CONT<br>0111: RAM<br>1000: RTP<br>1001: TRIG<br>1011: BRAKE | 0 |

RAMADDRH: (Address 40h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:6 | Reserved | RWS | Not used | 0 |
| 5:0 | RAMADDRH | RWS | SRAM address high six bits | 0 |

RAMADDRL: (Address 41h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | RAMADDRL | RWS | SRAM address low eight bits | 0 |

RAMDATA: (Address 42h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | RAMDATA | RWS | SRAM data entry | 0 |

SYSCTRL1: (Address 43h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | VBAT_MODE | RW | VBAT adjust mode, 0: software adjust mode, 1: hardware adjust mode | 0 |
| 6 | Reserved | RW | Not used | 1 |
| 5:4 | Reserved | RW | Not used | 0 |
| 3 | EN_RAMINIT | RW | Enable internal OSC clk<br>After powerup, system should initial SRAM for preload effects, to do so, this bit must be set to 1 | 0 |
| 2 | EN_FIR | RW | set enable of FIR filter | 1 |
| 1:0 | Reserved | RW | Not used | 0 |

SYSCTRL2: (Address 44h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | WAKE | RW | Chip enable control<br>1: set chip into active mode | 0 |
| 6 | STANDBY | RW | Chip disable control<br>1: set chip into standby mode | 0 |
| 5:4 | Reserved | RW | Not used | 2 |
| 3 | Reserved | RW | Not used | 0 |
| 2 | Reserved | RW | Not used | 0 |
| 1:0 | WAVDAT_MODE | RW | waveform data up sample rate selection:<br>b00: 24kHz<br>b01: 48kHz<br>others: 12kHz rate | 0 |

SYSCTRL7: (Address 49h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | Reserved | RW | Not used | 0 |
| 6 | GAIN_BYPASS | RW | 1: gain can be changed when playing<br>0: gain can not be changed when playing | 0 |
| 5:3 | Reserved | RW | Not used | 0 |
| 2:0 | D2S_GAIN | RW | Set D2S gain<br>000: 1<br>001: 2<br>010: 4<br>011: 8<br>100: 10<br>101: 16<br>110: 20<br>111: 26.7 | 4 |

PWMCFG1: (Address 4Ch)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | PRC_EN | RW | Set enable of output signal protection mode of pwm:<br>0: disable<br>1: When HDP/HDN output voltage ≥ 124/128*PVDD maintains (PRCTIME/3k)s, HDP/HDN is pulled down protectively | 1 |

| | | | | |
|---|---|---|---|---|
| 6:0 | PRCTIME | RW | set protection time of output signal protection mode of pwm, unit time is (1/3k) s. | 0x20 |

PWMCFG2: (Address 4Dh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:5 | reserved | RW | Not used | 1 |
| 4 | PD_HWM | RW | shutdown half wave modulate<br>0: half wave mode<br>1: full wave mode | 0 |
| 3:0 | reserved | RW | Not used | 8 |

PWMCFG3: (Address 4Eh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7 | PR_EN | RW | Set enable of input signal protection mode of pwm:<br>0: disable<br>1: When output voltage >= PRLVL/128*PVDD maintains (PRTIME/3k)s, HDP/HDN is pulled down protectively | 1 |
| 6:0 | PRLVL | RW | set protection voltage of input signal protection mode of pwm | 0x3F |

PWMCFG4: (Address 4Fh)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | PRTIME | RW | set protection time of input signal protection mode of pwm, unit time is (1/3k) s. | 0x32 |

DETCFG1: (Address 51h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:5 | Reserved | RW | Not used | 0 |
| 4 | RL | RW | Set diagnostic mode<br>1:RL | 0 |
| 3 | Reserved | RW | Not used | 0 |
| 2:0 | CLK_ADC | RW | Set frequency of ADC clock<br>000: 12Mhz<br>001: 6Mhz<br>010: 3Mhz<br>011: 1.5Mhz<br>100: 0.75Mhz<br>101: 0.375Mhz<br>110: 0.1875Mhz<br>111: 0.09375Mhz | 2 |

DETCFG2: (Address 52h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:2 | Reserved | RW | Not used | 0 |
| 1 | VBAT_GO | RW | Set the enabled of VBAT mode | 0 |
| 0 | DIAG_GO | RW | Set the enabled of DIAG mode | 0 |

DET_RL: (Address 53h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | RL_H | RO | the Measured value of resistance of LRA in DIAG mode(high eight bits)<br>RL=code*678/(1024*d2s_gain) (Ω) | 0 |

DET_VBAT: (Address 55h)

| Bit | Symbol | R/W | Description | Default |
|---|---|---|---|---|
| 7:0 | VBAT_H | RO | the Measured value of VBAT in VBAT mode(high eight bits)<br>VBAT=code*6.1/1024 (V) | 0 |

| DET_LO: (Address 57h) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:6 | Reserved | RO | Not used | 0 |
| 5:4 | VBAT_LO | RO | the Measured value of VBAT in VBAT mode(low two bits)<br>VBAT=code*6.1/1024 (V) | 0 |
| 3:2 | Reserved | RO | Not used | 0 |
| 1:0 | RL_LO | RO | the Measured value of resistance of LRA in DIAG mode(low two bits)<br>RL=code*678/(1024*d2s_gain) (Ω) | 0 |

| TRIMCFG3: (Address 5Ah) | | | | |
|---|---|---|---|---|
| **Bit** | **Symbol** | **R/W** | **Description** | **Default** |
| 7:6 | Reserved | RW | Not used | 0 |
| 5:0 | TRIM_LRA | RW | Change the frequency of output waveform, and one LSB is 0.24%<br>011111: 31xLSB<br>011110: 30xLSB<br>.....<br>000001: 1xLSB<br>000000: 0xLSB<br>111111: -1xLSB<br>.....<br>100001: -31xLSB<br>100000: -32xLSB | 0 |

# Application Information

### Output beads, capacitors

The device output is a square wave signal, which causing switch current at the output capacitor, increasing static power consumption, and therefore output capacitor should not be too large, 0.1nF ceramic capacitors is recommended.

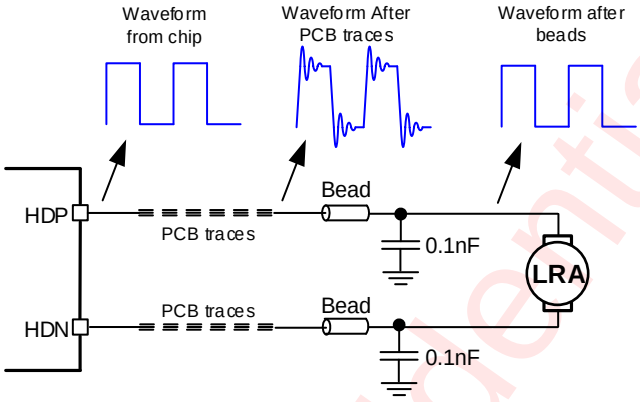



**Figure 28 Ferrite Chip Bead and capacitor**

The device output is a square wave signal. The voltage across the capacitor will be much larger than the PVDD voltage after increasing the bead capacitor. It suggested the use of rated voltage above 10V capacitor. At the same time a square wave signal at the output capacitor switching current form, the static power consumption increases, so the output capacitance should not be too much which is recommended 0.1nF ceramic capacitor rated voltage of 10V.

# PCB Layout Consideration

## Layout Considerations

The suggested Layout is illustrated in the following diagram:

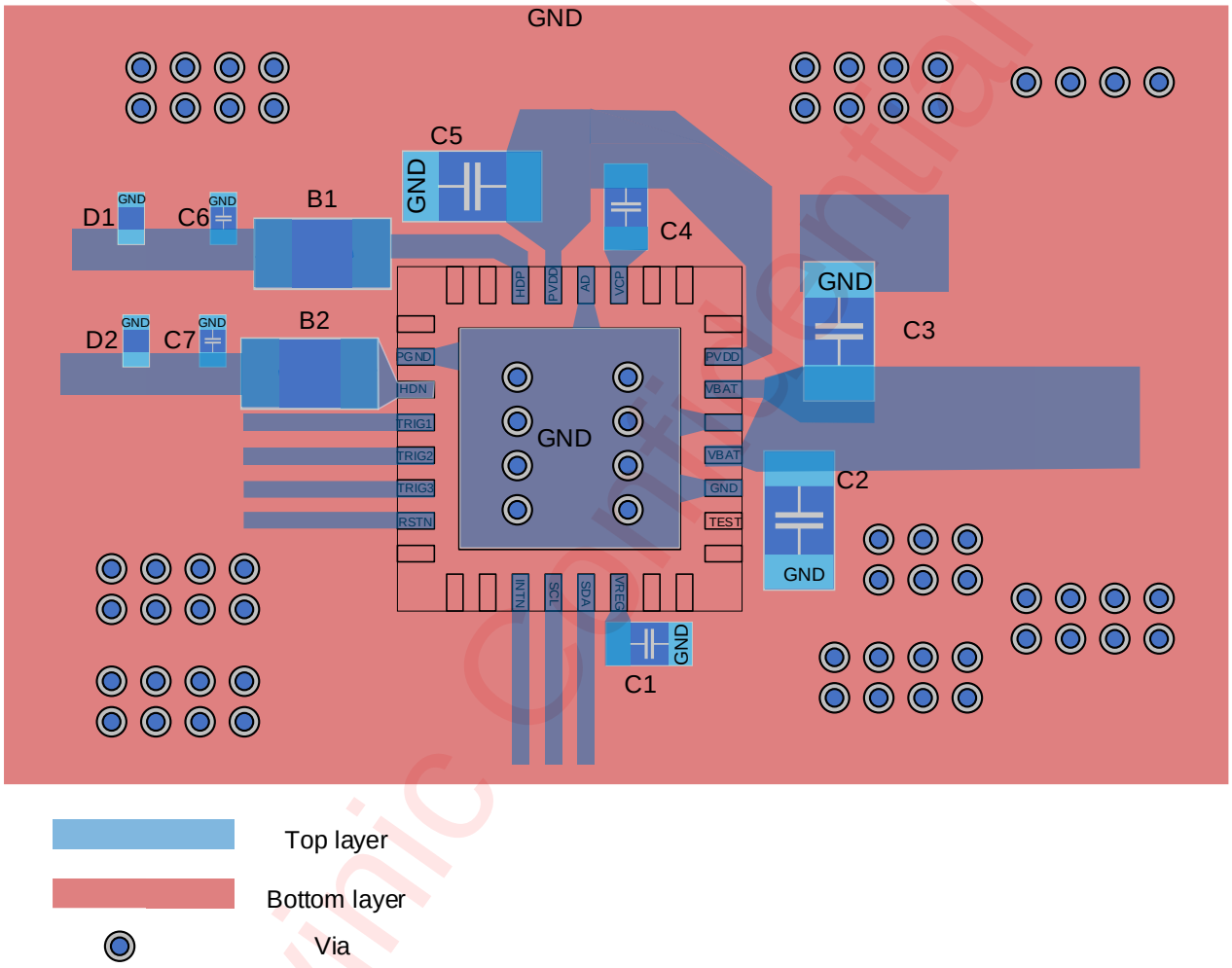


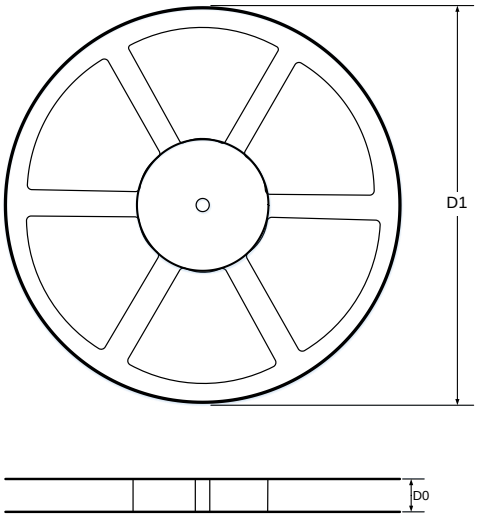

**Figure 29 AW86205QNR-Q1 Board Layout**

Here are some guidelines:

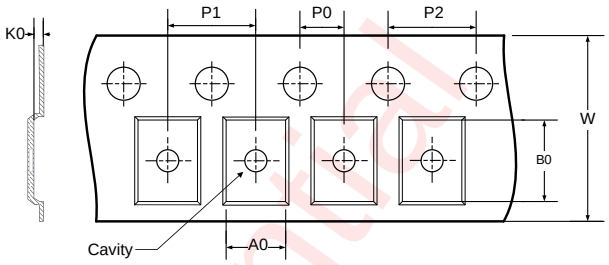
1. All of the external components should be placed as close as possible to IC in top layer PCB.
2. SCL and SDA should be shield by ground.
3. VBAT power supply lines are as short and thick as possible. The current flow capacity of the traces not be less than $\frac{PVDD}{(RL+Rdson)*\eta}$. ( RL is motor DC impedance, Rdson takes 0.3Ω, η is chip overall efficiency takes 80%)
4. Routing overcurrent capability of HDP/HDN output to the load should meet $\frac{PVDD}{R_L+R_{DSON}}$. HDP and HDN should be shield by ground and far away from the interference source especially the FLY capacitor of the high-power charging IC, otherwise it will cause the abnormal F0 detection.

# Tape And Reel Information

REEL DIMENSIONS

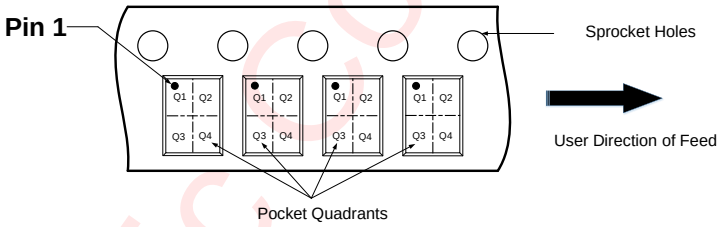



TAPE DIMENSIONS




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
P0：Pitch between successive cavity centers and sprocket hole
P1：Pitch between successive cavity centers
P2：Pitch between sprocket hole
D1：Reel Diameter
D0：Reel Width

QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE



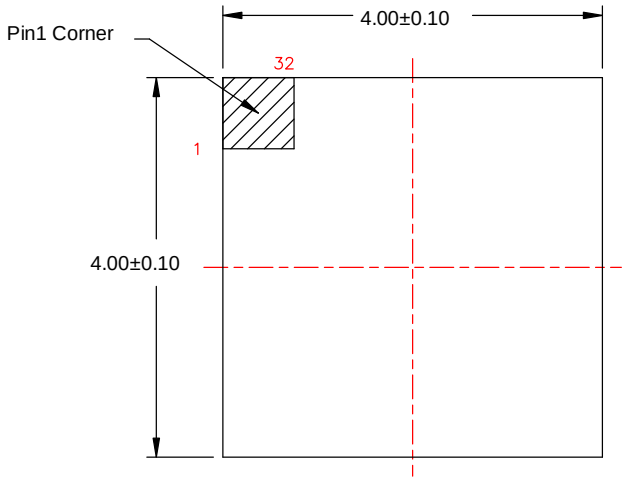

Note：The above picture is for reference only. Please refer to the value in the table below for the actual size

DIMENSIONS AND PIN1 ORIENTATION

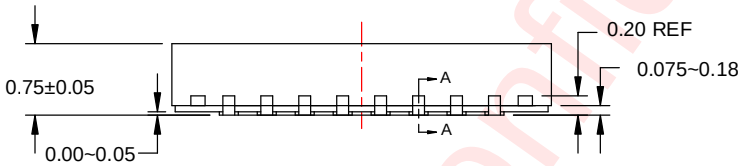
| D1 (mm) | D0 (mm) | A0 (mm) | B0 (mm) | K0 (mm) | P0 (mm) | P1 (mm) | P2 (mm) | W (mm) | Pin1 Quadrant |
|---|---|---|---|---|---|---|---|---|---|
| 330.00 | 12.40 | 4.35 | 4.35 | 1.10 | 2 | 8 | 4 | 12 | Q1 |

**All dimensions are nominal**

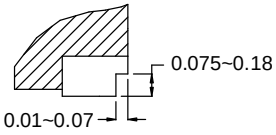
## Package Description



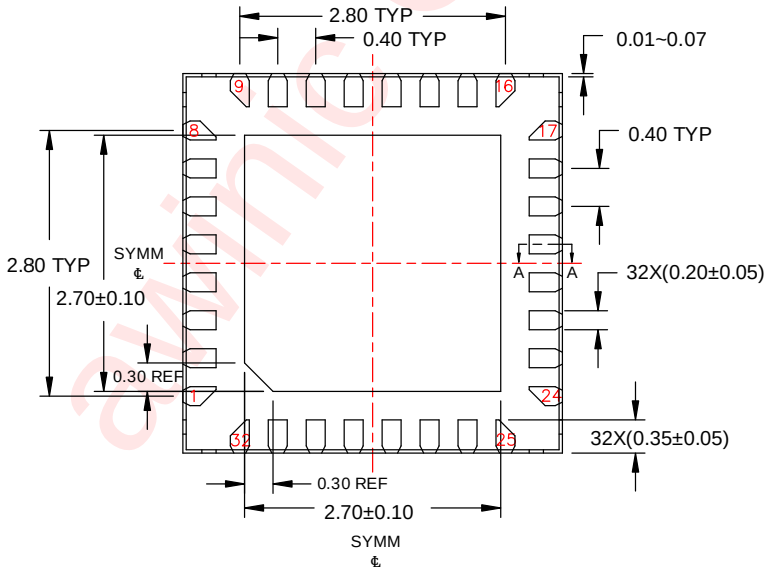

TOP VIEW




SIDE VIEW



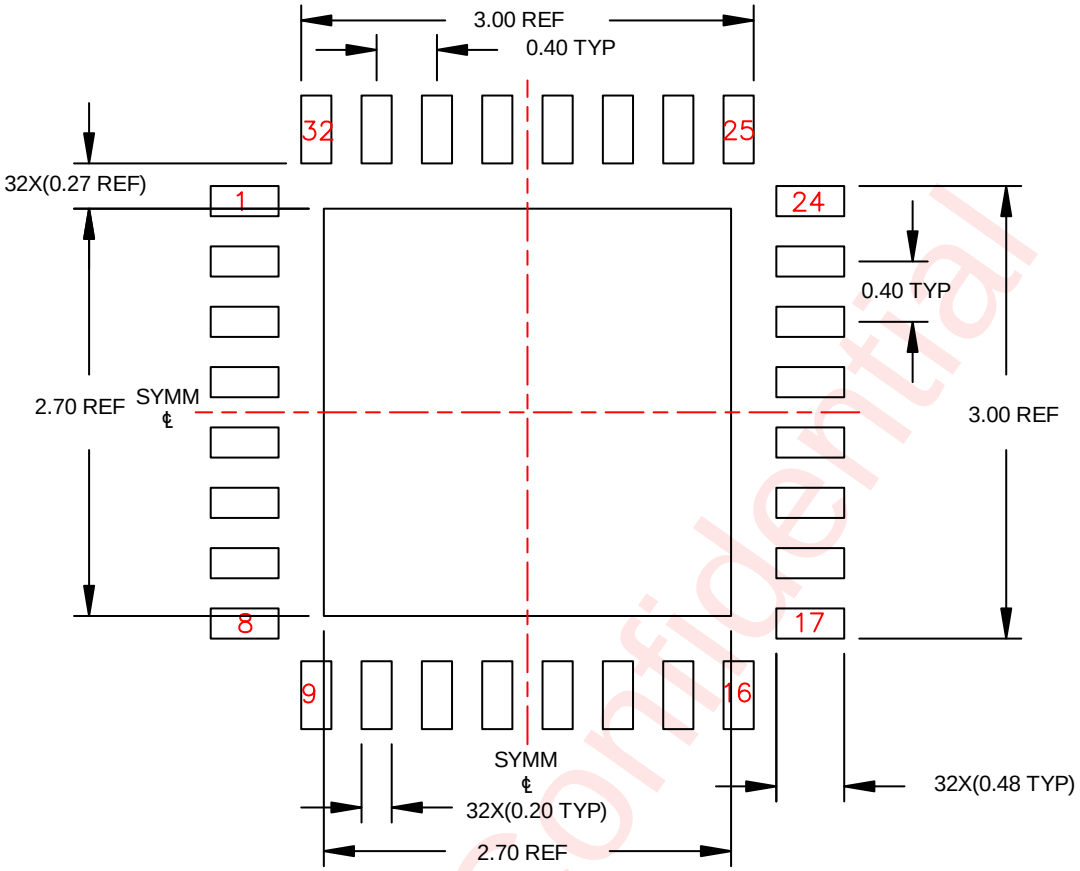

SECTION A-A
TYPICAL

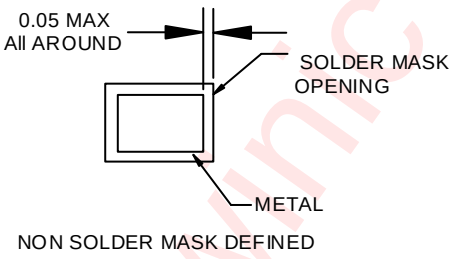



BOTTOM VIEW

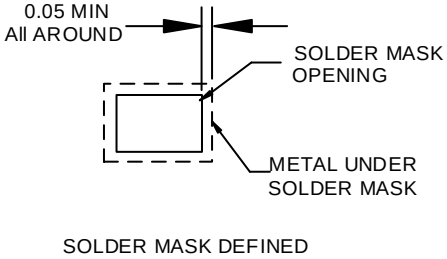
Unit: mm

## Land Pattern Data




NON SOLDER MASK DEFINED




SOLDER MASK DEFINED

Unit: mm

# Revision History

| Version | Date | Change Record |
|---|---|---|
| V1.0 | September 2023 | Initial Version |
| V1.1 | October 2023 | Revise operation voltage from 3~5.5V to 3~5.2V<br>Revise Package picture<br>Revise AW86205-Q1 to AW86205QNR-Q1<br>Add “Wettable flanks” to features |

# Disclaimer

Information in this document is believed to be accurate and reliable. However, Shanghai AWINIC Technology Co., Ltd (AWINIC Technology) does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

AWINIC Technology reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. Customers shall obtain the latest relevant information before placing orders and shall verify that such information is current and complete. This document supersedes and replaces all information supplied prior to the publication hereof.

AWINIC Technology products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of an AWINIC Technology product can reasonably be expected to result in personal injury, death or severe property or environmental damage. AWINIC Technology accepts no liability for inclusion and/or use of AWINIC Technology products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications that are described herein for any of these products are for illustrative purposes only. AWINIC Technology makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.
All products are sold subject to the general terms and conditions of commercial sale supplied at the time of order acknowledgement.

Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Reproduction of AWINIC information in AWINIC data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. AWINIC is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of AWINIC components or services with statements different from or beyond the parameters stated by AWINIC for that component or service voids all express and any implied warranties for the associated AWINIC component or service and is an unfair and deceptive business practice. AWINIC is not responsible or liable for any such statements.