

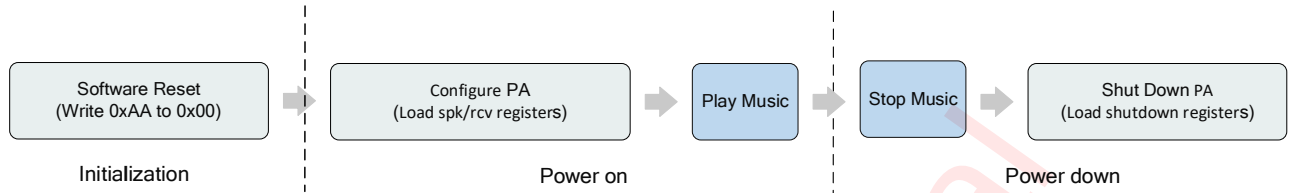
AW87390 Software Design Guide

Awinic Confidential

1	Software Design.....	3
1.1	Process	3
1.2	Power on/off timing sequence	3
1.2.1	Power on	3
1.2.2	Power off	4
1.3	I ² C Address	4
1.4	Register Configuration	4
1.4.1	Speaker Mode.....	4
1.4.2	Receiver Mode	6
1.4.3	Shutdown Mode	7
2	Appendix.....	8
2.1.1	Multi-PA Compatibility.....	8
2.1.2	I2C communication failure	8
2.1.3	Machine power failure during playback	8

1 SOFTWARE DESIGN

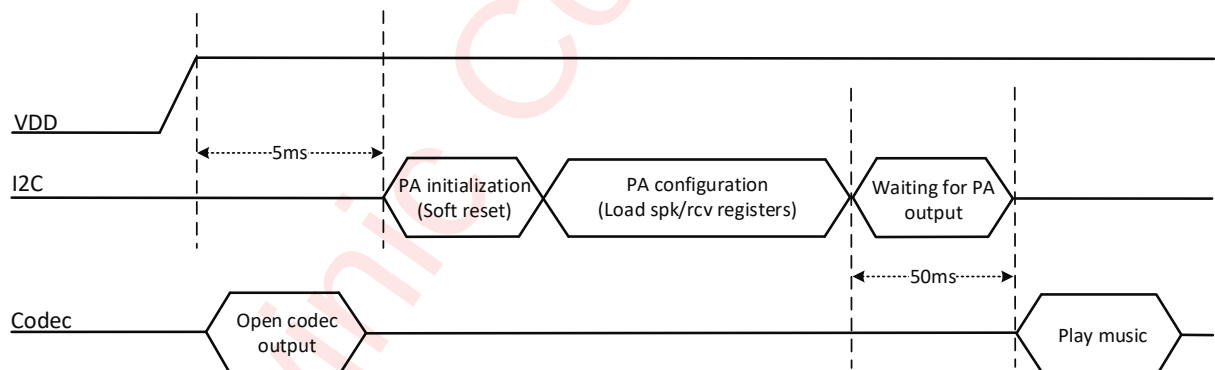
1.1 Process



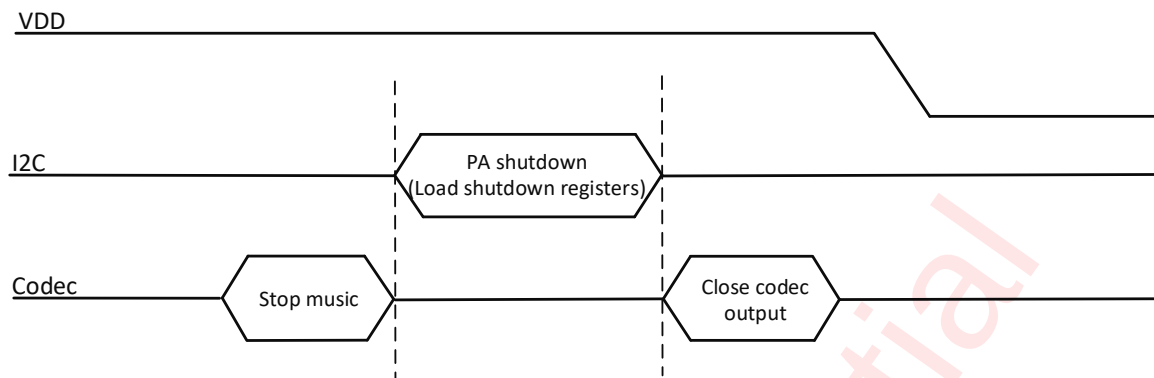
Writing 0xAA to the 0x00 register can achieve a soft reset. After the reset is completed, operate PA after a delay of at least 5ms. After each completion of playing music, a PA shutdown action is required.

1.2 Power on/off timing sequence

1.2.1 Power on



1.2.2 Power off



1.3 I2C Address

AW87390 can configure different I2C addresses through the high and low levels of the AD pin.

The following table provides instructions for configuring I2C addresses.

AD2	AD1	I2C address
0	0	0x58
0	1	0x59
1	0	0x5A
1	1	0x5B

1.4 Register Configuration

0xFE is a non-valid register used for delay purposes. The delay time corresponds to the value of the 0xFE register, measured in milliseconds.

1.4.1 Speaker Mode

```
typedef struct {  
    uint8_t reg_addr;  
    uint8_t reg_data;
```

```
}aw87xxx_i2c_type;

static const aw87xxx_i2c_type aw87390_kspk_reg[] = {
    {0x02, 0x07},
    {0x02, 0x00},
    {0x02, 0x0C},
    {0x66, 0x38},
    {0x01, 0x03},
    {0xFE, 0x02},
    {0x67, 0x23},
    {0x02, 0x07},
    {0x02, 0x00},
    {0x02, 0x1C},
    {0x03, 0x08},
    {0x04, 0x05},
    {0x05, 0x0C},
    {0x06, 0x07},
    {0x07, 0x4E},
    {0x08, 0x06},
    {0x09, 0x08},
    {0x0A, 0x4B},
    {0x61, 0xB3},
    {0x62, 0x24},
    {0x63, 0x09},
    {0x64, 0x24},
    {0x65, 0x15},
    {0x79, 0x7A},
    {0x7A, 0x6C},
    {0x78, 0x80},
    {0x66, 0x38},
    {0x76, 0x00},
    {0x78, 0x00},
    {0x68, 0x1B},
    {0x69, 0x5B},
    {0x70, 0x1C},
    {0x71, 0x00},
    {0x72, 0xFE},
    {0x73, 0x4F},
    {0x74, 0x24},
    {0x75, 0x02},
    {0x01, 0x07},
    {0xFF, 0x50},
};

/*write demo*/
for (int i = 0; i < sizeof(aw87390_kspk_reg)/sizeof(aw87390_kspk_reg[0]); i++)
{
```

```
if (aw87390_kspk_reg[i].reg_addr == 0xFE) {
    HAL_Delay(aw87390_kspk_reg[i].reg_data);
    continue;
}

i2c_write(aw87390_kspk_reg[i].reg_addr, aw87390_kspk_reg[i].reg_data);
}
```

1.4.2 Receiver Mode

```
typedef struct {
    uint8_t reg_addr;
    uint8_t reg_data;
}aw87xxx_i2c_type;

static const aw87xxx_i2c_type aw87390_rcv_reg[] = {
    {0x02, 0x07},
    {0x02, 0x00},
    {0x02, 0x0C},
    {0x66, 0x38},
    {0x01, 0x03},
    {0xFE, 0x02},
    {0x67, 0x03},
    {0x02, 0x07},
    {0x02, 0x00},
    {0x02, 0x00},
    {0x03, 0x08},
    {0x04, 0x05},
    {0x05, 0x13},
    {0x06, 0x0F},
    {0x07, 0x4E},
    {0x08, 0x09},
    {0x09, 0x08},
    {0x0A, 0x4B},
    {0x61, 0xB3},
    {0x62, 0x24},
    {0x63, 0x05},
    {0x64, 0x28},
    {0x65, 0x15},
    {0x79, 0x7A},
    {0x7A, 0x6C},
    {0x78, 0x80},
    {0x66, 0xB8},
    {0x76, 0x00},
    {0x78, 0x00},
}
```

```
{0x68, 0x1B},
{0x69, 0x5B},
{0x70, 0x1C},
{0x71, 0x10},
{0x72, 0xB4},
{0x73, 0x4F},
{0x74, 0x24},
{0x75, 0x02},
{0x01, 0x07},
{0xFF, 0x10},
};

/*write demo*/
for (int i = 0; i < sizeof(aw87390_rcv_reg)/sizeof(aw87390_rcv_reg[0]); i++) {
    if (aw87390_rcv_reg[i].reg_addr == 0xFE) {
        HAL_Delay(aw87390_rcv_reg[i].reg_data);
        continue;
    }

    i2c_write(aw87390_rcv_reg[i].reg_addr, aw87390_rcv_reg[i].reg_data);
}
```

1.4.3 Shutdown Mode

```
typedef struct {
    uint8_t reg_addr;
    uint8_t reg_data;
}aw87xxx_i2c_type;

static const aw87xxx_i2c_type aw87390_off_reg[] = {
    {0x01,0x00},
};

/*write demo*/
for (int i = 0; i < sizeof(aw87390_off_reg)/sizeof(aw87390_off_reg[0]); i++) {
    i2c_write(aw87390_off_reg[i].reg_addr, aw87390_off_reg[i].reg_data);
}
```

2 APPENDIX

2.1.1 Multi-PA Compatibility

To achieve compatibility across multiple PA models, it is possible to incorporate a chip ID verification process to distinguish between different chip models. By reading the 0x00 register, the chip ID value for the AW87390 is identified as 0x76.

2.1.2 I2C communication failure

1. Check if the power supply of the chip is normal.
2. Check if the bus and address configuration of I2C are correct.
3. Check if the level of the I2C bus matches the chip.
4. Check if the waveform of the I2C bus is normal.
5. Check if there are any issues with the hardware soldering.

You can refer to the document on the official website FAQ, titled " FAQ0100158: 音频芯片若 I2C 通信失败，如何排查? ". The official website FAQ link is <https://www.awinic.com/cn/faq.html>.

2.1.3 Machine power failure during playback

If the machine suddenly shuts down while playing music and the PA does not load the register for shutdown mode, causing the PA to shut down abnormally, it will cause PA leakage. It is recommended to refer to the configuration of the shutdown mode register and bind the shutdown of PA operation with the machine power down.

Revision History

Version	Date	Record

Awinic Confidential

Declaration

All trademarks are the property of their respective owners. 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 technical document 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.