PAN1080 I2C Sample Application Note

PAN-CLT-VER-B0, Rev 0.2

PanchipMicroelectronics

www.panchip.com

修订历史

版本	修订日期	描述
V0.1	2022-10-12	初始版本创建
V0.2	2023-02-13	更新第 3 章注意事项

目录

第1章 例程演示内容

1.1 测试内容

1. 寄存器默认值
2. Slave 接收数据，Master 发送数据
3. Slave 发送数据，Master 接收数据
4. 在 10-bit Address Mode 下收发数据（Master 发，Slave 收）
5. 使用 DMA 方式，Master 发，Slave 收
6. 使用 DMA 方式，Slave 发，Master 收
7. 使用中断方式收发数据
8. Master General Call 功能

1.2 环境配置

1.2.1 软件环境

1.2.1.1 待测代码

测试工程文件：

<PAN1080-DK>\03_MCU\mcu_samples\I2C\keil\I2C.uvprojx

测试源文件目录：

<PAN1080-DK>\03_MCU\mcu_samples \I2C\src

1.2.1.2 软件工具

1、SecureCRT（用于显示 PC 与 EVB 的交互过程，打印 log 等）

2、KingstVIS（逻辑分析仪 LA1010 配套软件）

1.2.2 硬件环境

1、PAN1080 EVB 2 块

a) UART0（测试交互接口，TX: P30，RX: P31，波特率：921600 ）

b) I2C0（待测模块，SCL: P00，SDA: P01）

c) SWD（用来调试和烧录程序，SWDCLK: P46，SWDIO: P47）

2、逻辑分析仪（波形抓取工具）

3、JLink（SWD 调试与烧录工具）

第2章 例程演示流程

2.1 环境说明

2.1.1 测试程序编译烧录

打开测试工程，确保可以编译通过。

2.1.2 硬件接线

接线方面，需要：

1. 将 EVB 板的 TX0 连接 P30、RX0 连接 P31，USB->UART 与 PC 连接；

2. 将 I2C0（待测模块）的 SCL/SDA（P00/P01）与 I2C0（辅助测试模块）的 SCL/SDA（P00/P01）相连。

2. 将 I2C0（待测模块）的 SCL/SDA（P00/P01）与逻辑分析仪相连。

2.2 I2C 工作流程

参考 User Manual 文档。

2.3 测试程序初始化

硬件连线完成并烧录测试程序后，EVB 上电，观察串口是否正常打印测试主菜单。

```
CPU @ 64000000Hz

+-------------------------------------------------------------------------------+
|                      PAN1080 I2C Sample Code.                                 |
+-------------------------------------------------------------------------------+
|    Press key to start specific testcase:                                      |
|                                                                               |
|    Input '0'    Testcase 0 : Register Default Value Check.                    |
|    Input '1'    Testcase 1 : Slave  Receive Data.                             |
|    Input '2'    Testcase 2 : Slave  Send    Data.                             |
|    Input '3'    Testcase 3 : Master Receive Data.                             |
|    Input '4'    Testcase 4 : Master Send    Data.                             |
|    Input '5'    Testcase 5 : Master Send    Data with 10-bit Address.         |
|    Input '6'    Testcase 6 : Slave  Receive Data with 10-bit Address.         |
|    Input '7'    Testcase 7 : Master Write with DMA Enable.                    |
|    Input '8'    Testcase 8 : Slave  Read  with DMA Enable.                    |
|    Input '9'    Testcase 9 : Master General Call.                             |
|    Input 'A'    Testcase 10: Master Write with Interrupt Enable.              |
|    Input 'B'    Testcase 11: Slave  Read  with Interrupt Enable.              |
|    Input 'C'    Testcase 12: Master Read  with DMA Enable.                    |
|    Input 'D'    Testcase 13: Slave  Write with DMA Enable.                    |
+-------------------------------------------------------------------------------+
```

2.4 基本功能验证

2.4.1 I2C 所有寄存器默认状态

在主菜单下，输入‘0’命令 打印所有寄存器默认值：

测试目的：

检查所有 I2C 相关寄存器复位 Default 值状态。

测试预期：

寄存器默认值应和 Datasheet 上 I2C 模块默认值一致。

测试现象：

```
0
iic default value check ok
```

测试分析：

测试程序将读到的寄存器值与预期值相比较，发现完全一致，于是打印检查通过的 Log，符合预期。

2.4.2 I2C Slave 接收数据，Master 发送数据

测试目的：

验证 I2C 作为 Slave 收、Master 发功能是否正常。

测试预期：

收发数据均正常。

测试现象：

先操作 EVB 1，输入‘1’命令，进入 Slave 数据接收流程，准备接收 EVB 2 发来的数据。

再操作 EVB 2，输入‘4’命令，进入 Master 数据发送流程，向 EVB 1 发送数据(0X0~0X1F)，此时发现 EVB 1 打印出接收到的数据。

```
---------------+
1
I2C_SlaveReceiveDataCase2
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F
```

测试分析：

I2C 接收端（EVB 1）先进入 Slave 接收流程，等待发送端发送地址和数据；随后，发送端（EVB 2）进入 Master 发送流程，向接收端发送 32 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.3 I2C Slave 发送数据，Master 接收数据

测试目的：

验证 I2C 作为 Slave 发、Master 收功能是否正常。

测试预期：

收发数据均正常。

测试现象：

先操作 EVB 1，输入‘2’命令，进入 Slave 数据发送流程，准备向 EVB 2 发送数据。

再操作 EVB 2，输入‘3’命令，进入 Master 数据接收流程，接收 EVB 1 发来的数据，此时发现 EVB 2 打印出接收到的数据。

```
---------------+
3
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F

+--------------------------------------------------------------------------
```

测试分析：

I2C 发送端（EVB 1）先进入 Slave 发送流程，等待接收端发送过来的数据请求；随后，接收端（EVB 2）进入 Master 接收流程，向接收端请求接收 32 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.4 I2C Master 接收数据，Slave 发送数据

测试目的：

验证 I2C 作为 Master 收、Slave 发功能是否正常。

测试预期：

收发数据均正常。

测试现象：

先操作 EVB 2，输入‘1’命令，进入 Slave 数据接收流程，准备接收 EVB 1 发来的数据。

再操作 EVB 1，输入‘4’命令，进入 Master 数据发送流程，向 EVB 2 发送数据(0X0~0X1F)，此时发现 EVB 1 打印出接收到的数据。

```
---------------+
1
I2C_SlaveReceiveDataCase2
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F
```

测试分析：

I2C 接收端（EVB 2）先进入 Slave 接收流程，等待发送端发送地址和数据；随后，发送端（EVB 1）进入 Master 发送流程，向接收端发送 32 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.5 I2C Master 发送数据，Slave 接收数据

测试目的：

验证 I2C 作为 Master 发、Slave 收功能是否正常。

测试预期：

收发数据均正常。

测试现象：

先操作 EVB 1，输入‘2’命令，进入 Slave 数据发送流程，准备向 EVB 2 发送数据。

再操作 EVB 2，输入‘3’命令，进入 Master 数据接收流程，接收 EVB 1 发来的数据，此时发现 EVB 2 打印出接收到的数据。

```
---------------+
3
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F

+-------------------------------------------------------------------------
```

测试分析：

I2C 发送端（EVB 2）先进入 Slave 发送流程，等待接收端发送过来的数据请求；随后，接收端（EVB 1）进入 Master 接收流程，向接收端请求接收 32 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.6 I2C 10-bit Address Mode 下收发数据

测试目的：

验证 I2C 10-bit Address Mode 下工作是否正常。

测试预期：

10-bit Address Mode 下收发数据均正常。

测试现象：

先操作 EVB 1，输入‘6’命令，进入 Slave 数据接收流程（10-bit Address），准备接收 EVB 2 发送过来的数据。

再操作 EVB 2，输入‘5’命令，进入 Master 数据发送流程（10-bit Address），向 EVB 1 发送数据，此时发现 EVB 1 打印出接收到的数据。

```
---------------+
6
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F

+-------------------------------------------------------------------------
```

测试分析：

I2C 接收端（EVB 1）先进入 Slave 接收流程，等待发送端发送过来的数据；随后，发送端（EVB 2）进入 Master 发送流程，向接收端发送 32 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.7 I2C 使用 DMA 方式，Master 发，Slave 收

测试目的：

验证 I2C DMA 方式下，Master 发送数据、Slave 接收数据是否正常。

测试预期：

收发数据均正常。

测试现象：

先操作 EVB 1，输入‘8’命令，进入 Slave 数据 DMA 方式接收流程，准备接收 EVB 2 发送过来的数据。

再操作 EVB 2，输入‘7’命令，进入 Master 数据 DMA 方式发送流程，向 EVB 1 发送数据，此时发现 EVB 1 打印出接收到的数据。

```
---------------+
8
I2C_DmaPeripheral2MemTranferCase9
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17
18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f
30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d 3e 3f 40 41 42 43 44 45 46 47
48 49 4a 4b 4c 4d 4e 4f 50 51 52 53 54 55 56 57 58 59 5a 5b 5c 5d 5e 5f
60 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77
78 79 7a 7b 7c 7d 7e 7f 80 81 82 83 84 85 86 87 88 89 8a 8b 8c 8d 8e 8f
90 91 92 93 94 95 96 97 98 99 9a 9b 9c 9d 9e 9f a0 a1 a2 a3 a4 a5 a6 a7
a8 a9 aa ab ac ad ae af b0 b1 b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf
c0 c1 c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 d2 d3 d4 d5 d6 d7
d8 d9 da db dc dd de df e0 e1 e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef
f0 f1 f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff

+-----------------------------------------------------------------------
---------------+
|                    PN108C I2C Sample Code.
                 |
```

测试分析：

I2C 接收端（EVB 1）先进入 Slave DMA 接收流程，等待发送端发送过来的数据；随后，发送端（EVB 2）进入 Master DMA 发送流程，向接收端发送 256 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.8 I2C 使用 DMA 方式，Master 收，Slave 发

测试目的：

验证 I2C DMA 方式下，Master 接收数据、Slave 发送数据是否正常。

测试预期：

收发数据均正常。

测试现象：

先操作 EVB 1，输入‘D’命令，进入 Slave 数据 DMA 方式发送流程，准备向 EVB 2 发送数据。

再操作 EVB 2，输入‘C’命令，进入 Master 数据 DMA 方式接收流程，接收 EVB 1 发送过来的数据，此时发现 EVB 2 打印出接收到的数据。

```
---------------+
c
00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15 16 17
18 19 1a 1b 1c 1d 1e 1f

+-------------------------------------------------------------------------
---------------+
|                        PN108C I2C Sample Code.
               |
+-------------------------------------------------------------------------
```

测试分析：

I2C 发送端（EVB 1）先进入 Slave DMA 发送流程，等待向接收端发送数据；随后，接收端（EVB 2）进入 Master DMA 接收流程，向发送端请求发送 32 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.9 I2C 使用中断方式收发数据

测试目的：

验证 I2C 在中断方式下收发数据是否正常。

测试预期：

收发数据均正常。

测试现象：

先操作 EVB 1，输入‘B’命令，进入 Slave 数据中断方式收发流程，准备响应 EVB 2 发送过来的发送或接收数据请求。

再操作 EVB 2，输入‘A’命令，进入 Master 数据中断方式收发流程，其先向 EVB 1 发送一段数据，然后再向 EVB 1 请求接收一段数据。观察发现 EVB 1 和 EVB 2 分别打印出接收到的数据。

```
---------------+
b
I2C_SlaveInterruptCase12
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F

+-------------------------------------------------------------------------
---------------+
```

```
---------------+
a
I2C_InterruptCase11
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 AA AB AC AD AE AF B0 B1 B2 B3 B4 B5 B6 B7
B8 B9 BA BB BC BD BE BF

+-----------------------------------------------------------------------
```

测试分析：

I2C 接收端（EVB 1）先进入 Slave DMA 接收流程，等待发送端发送过来的数据；随后，发送端（EVB 2）进入 Master DMA 发送流程，向接收端发送 256 Bytes 数据。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

2.4.10 I2C Master General Call 功能

测试目的：

验证 I2C Master General Call 功能是否正常。

测试预期：

使用 Master General Call 可以正常广播数据。

测试现象：

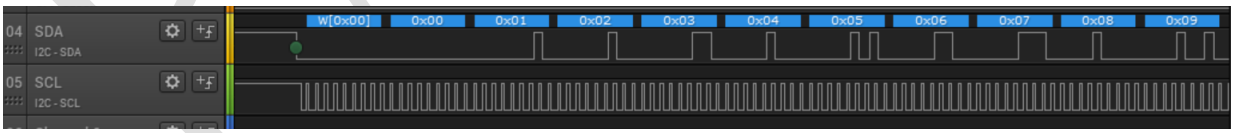
先操作 EVB 2，输入‘1’命令，进入 Slave 数据接收流程，准备接收 EVB 1 的广播数据。

再操作 EVB 1，输入‘9’命令，进入 Master General Call 发送数据流程，向总线广播数据，此时发现 EVB 2 打印出接收到的数据。

```
---------------+
1
I2C_SlaveReceiveDataCase2
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17
18 19 1A 1B 1C 1D 1E 1F

+-----------------------------------------------------------------------
```

使用逻分抓取当前波形，可以看到 Master 实际上是在向 0x00 地址发送数据：



测试分析：

I2C 接收端（EVB 1）先进入 Slave 接收流程，等待发送端发送地址和数据；随后，发送端（EVB 2）进入 Master General Call 发送流程，其向总线广播数据（实际上是向 0x00 地址发送数据）。由接收端 Log 可知，其成功接收到发送端发来的数据，符合预期。

第3章 注意事项

1、 例程中演示的 I2C 通信速率较慢（仅 100K 左右），若希望以更高的速度通信，则有以下注意事项：

A、 I2C 模块所在的 APB 总线时钟频率可能需要提高，否则驱动初始化（I2C_Init）过程中可能会返回 false。

B、 I2C IO 设置为禁用内部上拉电阻，并在电路上外挂较小的上拉电阻。原因是芯片内部上拉电阻阻值较大，充放电时间会比较久，从而导致通信频率与预期偏差较大。

2、 DMA 方式下， I2C 设置的 I2C 发送接收 triglevel 等级需与 I2C 通道的 burstlenth 设置的一致，否则在接收数据的情况下可能出现丢失数据的情况，例如：

A、 I2C_IICReceiveDataLevel(I2Cx, I2C_RX_TL_3); I2C_RX_TL_3 表示 4 个数据触发 IIC 请求

B、 TxConfigTmp1.BurstLenSrc = DMAC_BurstLen_1;1 次 I2C 请求传输一个宽度数据

以上 A、B 配置在传输 20 个字节的 case 下，I2C 端最后 3 个数据残留在 IIC FIFO 中不能触发 IIC 请求，导致丢失。