

HT8219 使用手册

Ver 1.1

Revision

时间	原因	版本	作者	备注
2019-10-10	建立 V1.0 版本	V1.0	冯江宏	建立 V1.0 版本
2019-12-27	更新部分描述	V1.1	冯江宏	

HT8219 使用手册	1
1. 概述	4
2. SDK 文件	4
3. 宏定义	4
3.1. 函数返回值	4
3.2. 通用接口	6
3.2.1. HT8219_OpenCard	6
3.2.2. HT8219_CloseCard	6
3.2.3. HT8219_DdsSetEnable	7
3.2.4. HT8219_DdsGetEnable	8
3.2.5. HT8219_DdsSetChan0PhaseOffset	8
3.2.6. HT8219_DdsGetChan0PhaseOffset	9
3.2.7. HT8219_DdsSetChan0PhaseStep	10
3.2.8. HT8219_DdsGetChan0PhaseStep	10
3.2.9. HT8219_DdsSetChan1PhaseOffset	11
3.2.10. HT8219_DdsGetChan1PhaseOffset	12
3.2.11. HT8219_DdsSetChan1PhaseStep	12
3.2.12. HT8219_DdsGetChan1PhaseStep	13
3.2.13. HT8219_MixSetEnable	14
3.2.14. HT8219_MixGetEnable	14
3.2.15. HT8219_MixSetChan0CoefSin	15
3.2.16. HT8219_MixGetChan0CoefSin	16
3.2.17. HT8219_MixSetChan0CoefCos	16
3.2.18. HT8219_MixGetChan0CoefCos	17
3.2.19. HT8219_MixSetChan1CoefSin	18
3.2.20. HT8219_MixGetChan1CoefSin	18
3.2.21. HT8219_MixSetChan1CoefCos	19
3.2.22. HT8219_MixGetChan1CoefCos	20

1. 概述

略

2. SDK 文件

SDK 库文件包含下列文件：

ht8219_api_win7_32bit.dll

ht8219_api_win7_32bit.lib

这两个文件用于应用程序开发。

3. 宏定义

3.1. 函数返回值

Api 函数返回值统一使用 **HT8219_STATUS** 枚举值定义。

返回值定义如下：

HT8219_SUCCESS

成功

HT8219_OS_ERROR

系统错误

HT8219_LOW_MEMORY

内存不足

HT8219_INVALID_MSG_ID

无效的消息 ID

HT8219_BAD_PARAMETER_1

HT8219_BAD_PARAMETER_2

HT8219_BAD_PARAMETER_3

HT8219_BAD_PARAMETER_4

HT8219_BAD_PARAMETER_5

HT8219_BAD_PARAMETER_6

HT8219_BAD_PARAMETER_7

HT8219_BAD_PARAMETER_8

HT8219_BAD_PARAMETER_9

无效的参数 1...9

HT8219_NULL_DESCRIPTOR

空描述符

HT8219_NO_BOARD

板卡打开失败

HT8219_NOT_OPEN

板卡未打开

HT8219_READ_TIMEOUT

读操作超时

HT8219_WRITE_TIMEOUT

写操作超时

HT8219_READ_ERROR

读操作出错

HT8219_WRITE_ERROR

写操作出错

HT8219_READ_NO_IOPENDING

没有挂起的读操作

HT8219_READ_CREATE_EVENT_ERROR

读操作创建事件错误

HT8219_WRITE_NO_IOPENDING

没有挂起的写操作

HT8219_WRITE_CREATE_EVENT_ERROR

写操作创建事件错误

HT8219_SIGNALED

收到系统信号

API 说明

3.2. 通用接口

3.2.1. HT8219_OpenCard

原型

```
STDHT8219CALL HT8219_OpenCard(HT8219HANDLE * ph, HT8219_UINT8 CardId)
```

功能

打开板卡，并分配板卡资源。

参数说明

输入参数

ph: 板卡句柄指针

CardId: 板卡编号，取值为 1~255。

输出参数

无。

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表：

[HT8219_BAD_PARAMETER_1](#)

[HT8219_LOW_MEMORY](#)

[HT8219_NOT_OPEN](#)

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

HT8219HANDLE ph = NULL;

status = HT8219_OpenCard(&ph, 1);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.2. HT8219_CloseCard

原型

```
STDHT8219CALL HT8219_CloseCard(HT8219HANDLE * ph)
```

功能

关闭板卡，并释放打开板卡时申请的资源。

参数说明

输入参数

ph: 输入参数，板卡句柄指针

输出参数

无

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表：

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_CloseCard(ph);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3. 2. 3. HT8219_DdsSetEnable

原型

```
STDHT8219CALL HT8219_DdsSetEnable(HT8219HANDLE ph, HT8219_BOOL enable)
```

功能

设置 DDS 是否使能。

参数说明

输入参数

ph: 板卡句柄，由函数 [HT8219_OpenCard](#) 得到。

enable: [HT8219_TRUE](#) - 使能 DDS， [HT8219_FALSE](#) - 禁用 DDS。

输出参数

无。

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表：

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_DdsSetEnable(HT8219_TRUE);

if (status != HT8219_SUCCESS) {

    return status;

}
```

```
}
```

3.2.4. HT8219_DdsGetEnable

原型

```
STDHT8219CALL HT8219_DdsGetEnable(HT8219HANDLE ph, HT8219_BOOL * enable)
```

功能

获取 DDS 是否使能。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

enable: HT8219_TRUE - 使能 DDS, HT8219_FALSE - 禁用 DDS。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

HT8219_BOOL enable;

status = HT8219_DdsGetEnable(ph, &enable);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.5. HT8219_DdsSetChan0PhaseOffset

原型

```
STDHT8219CALL HT8219_DdsSetChan0PhaseOffset(HT8219HANDLE ph, HT8219_UINT32 offset)
```

功能

设置 DDS 通道 0 相位偏移。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

offset: 相位偏移, 32 位无符号数。

输出参数

无。

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表：

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_DdsSetChan0PhaseOffset(ph, 0xA0000000);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.6. HT8219_DdsGetChan0PhaseOffset

原型

```
STDHT8219CALL HT8219_DdsGetChan0PhaseOffset(HT8219HANDLE ph, HT8219_UINT32 *
offset)
```

功能

获取 DDS 通道 0 相位偏移。

参数说明

输入参数

ph: 板卡句柄，由函数 [HT8219_OpenCard](#) 得到。

输出参数

offset: 相位偏移，32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表：

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

HT8219_UINT32 offset;

status = HT8219_DdsGetChan0PhaseOffset(ph, &offset);
```

```
if (status != HT8219_SUCCESS) {  
    return status;  
}
```

3.2.7. HT8219_DdsSetChan0PhaseStep

原型

```
STDHT8219CALL HT8219_DdsSetChan0PhaseStep(HT8219HANDLE ph, HT8219_UINT32 step)
```

功能

设置 DDS 通道 0 相位步进。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

step: 相位步进, 32 位无符号数。

输出参数

无。

返回值

成功时返回 HT8219_SUCCESS。

失败时可能返回的返回值列表:

HT8219_BAD_PARAMETER_1

HT8219_NULL_DESCRIPTOR

HT8219_NO_BOARD

HT8219_OS_ERROR

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;  
  
status = HT8219_DdsSetChan0PhaseStep(ph, 0xAAAAAAAA);  
  
if (status != HT8219_SUCCESS) {  
    return status;  
}
```

3.2.8. HT8219_DdsGetChan0PhaseStep

原型

```
STDHT8219CALL HT8219_DdsGetChan0PhaseStep(HT8219HANDLE ph, HT8219_UINT32 *  
step)
```

功能

获取 DDS 通道 0 相位步进。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

step: 相位步进, 32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;
HT8219_UINT32 step;
status = HT8219_DdsGetChan0PhaseStep(ph, &step);
if (status != HT8219_SUCCESS) {
    return status;
}
```

3.2.9. HT8219_DdsSetChan1PhaseOffset

原型

```
STDHT8219CALL HT8219_DdsSetChan1PhaseOffset(HT8219HANDLE ph, HT8219_UINT32
offset)
```

功能

设置 DDS 通道 1 相位偏移。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

offset: 相位偏移, 32 位无符号数。

输出参数

无。

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_DdsSetChan1PhaseOffset(ph, 0xAAAAAAAA);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.10. HT8219_DdsGetChan1PhaseOffset

原型

```
STDHT8219CALL HT8219_DdsGetChan1PhaseOffset(HT8219HANDLE ph, HT8219_UINT32 *
offset)
```

功能

获取 DDS 通道 1 相位偏移。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

offset: 相位偏移, 32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

HT8219_UINT32 offset;

status = HT8219_DdsGetChan1PhaseOffset(ph, &offset);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.11. HT8219_DdsSetChan1PhaseStep

原型

```
STDHT8219CALL HT8219_DdsSetChan1PhaseStep(HT8219HANDLE ph, HT8219_UINT32 step)
```

功能

设置 DDS 通道 1 相位步进。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

step: 相位步进, 32 位无符号数。

输出参数

无。

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_DdsSetChan1PhaseStep(ph, 0xAAAAAAAA);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.12. HT8219_DdsGetChan1PhaseStep

原型

```
STDHT8219CALL HT8219_DdsGetChan1PhaseStep(HT8219HANDLE ph, HT8219_UINT32 *
step)
```

功能

获取 DDS 通道 1 相位步进。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

step: 相位步进, 32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

HT8219_OS_ERROR

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;
HT8219_UINT32 step;
status = HT8219_DdsGetChan1PhaseStep(ph, &step);
if (status != HT8219_SUCCESS) {
    return status;
}
```

3.2.13. HT8219_MixSetEnable

原型

```
STDHT8219CALL HT8219_MixSetEnable(HT8219HANDLE ph, HT8219_BOOL enable)
```

功能

设置 Mix 是否使能。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

enable: HT8219_TRUE - 使能 MIX, HT8219_FALSE - 禁用 MIX。

输出参数

无。

返回值

成功时返回 HT8219_SUCCESS。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;
status = HT8219_MixSetEnable(HT8219_TRUE);
if (status != HT8219_SUCCESS) {
    return status;
}
```

3.2.14. HT8219_MixGetEnable

原型

```
STDHT8219CALL HT8219_MixGetEnable(HT8219HANDLE ph, HT8219_BOOL * enable)
```

功能

获取 MIX 是否使能。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

enable: HT8219_TRUE - 使能 MIX, HT8219_FALSE - 禁用 MIX。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;
HT8219_BOOL enable;
status = HT8219_MixGetEnable(ph, &enable);
if (status != HT8219_SUCCESS) {
    return status;
}
```

3.2.15. HT8219_MixSetChan0CoefSin

原型

```
STDHT8219CALL HT8219_MixSetChan0CoefSin(HT8219HANDLE ph, HT8219_UINT32 sin)
```

功能

设置 MIX 通道 0 正弦系数。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

sin: 正弦系数, 32 位无符号数, 范围 0~65535。

输出参数

无

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_MixSetChan0CoefSin(ph, 0x1234);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.16. HT8219_MixGetChan0CoefSin

原型

```
STDHT8219CALL HT8219_MixGetChan0CoefSin(HT8219HANDLE ph, HT8219_UINT32 * sin)
```

功能

获取 MIX 通道 0 正弦系数。

参数说明

输入参数

ph: 板卡句柄，由函数 HT8219_OpenCard 得到。

输出参数

sin: 正弦系数，32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

HT8219_UINT32 sin;

status = HT8219_MixGetChan0CoefSin(ph, &sin);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.17. HT8219_MixSetChan0CoefCos

原型

STDHT8219CALL HT8219_Mi xSetChan0CoefCos([HT8219HANDLE](#) ph, [HT8219_UINT32](#) cos)

功能

设置 MIX 通道 0 余弦系数。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

cos: 余弦系数, 32 位无符号数, 范围 0~65535。

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_Mi xSetChan0CoefCos(ph, 0x1234);

if (status != HT8219_SUCCESS) {
    return status;
}
```

3.2.18. HT8219_MixGetChan0CoefCos

原型

STDHT8219CALL HT8219_Mi xGetChan0CoefCos([HT8219HANDLE](#) ph, [HT8219_UINT32](#) * step)

功能

获取 MIX 通道 0 余弦系数。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

cos: 相位步进, 32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

HT8219_NO_BOARD

HT8219_OS_ERROR

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;
HT8219_UINT32 cos;
status = HT8219_MixGetChan0CoefCos(ph, &cos);
if (status != HT8219_SUCCESS) {
    return status;
}
```

3.2.19. HT8219_MixSetChan1CoefSin

原型

```
STDHT8219CALL HT8219_MixSetChan1CoefSin(HT8219HANDLE ph, HT8219_UINT32 sin)
```

功能

设置 MIX 通道 1 正弦系数。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

sin: 正弦系数, 32 位无符号数, 范围 0~65535。

输出参数

无

返回值

成功时返回 HT8219_SUCCESS。

失败时可能返回的返回值列表:

HT8219_BAD_PARAMETER_1

HT8219_NULL_DESCRIPTOR

HT8219_NO_BOARD

HT8219_OS_ERROR

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;
status = HT8219_MixSetChan1CoefSin(ph, 0x1234);
if (status != HT8219_SUCCESS) {
    return status;
}
```

3.2.20. HT8219_MixGetChan1CoefSin

原型

STDHT8219CALL HT8219_Mi xGetChan1CoefSin([HT8219HANDLE](#) ph, [HT8219_UINT32](#) * sin)

功能

获取 MIX 通道 1 正弦系数。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

sin: 正弦系数, 32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219\_STATUS status = HT8219\_SUCCESS;  
HT8219\_UINT32 sin;  
status = HT8219_Mi xGetChan1CoefSin(ph, &sin);  
if (status != HT8219\_SUCCESS) {  
    return status;  
}
```

3.2.21. HT8219_MixSetChan1CoefCos

原型

STDHT8219CALL HT8219_Mi xSetChan1CoefCos([HT8219HANDLE](#) ph, [HT8219_UINT32](#) cos)

功能

设置 MIX 通道 1 余弦系数。

参数说明

输入参数

ph: 板卡句柄, 由函数 HT8219_OpenCard 得到。

输出参数

cos: 余弦系数, 32 位无符号数, 范围 0~65535。

返回值

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

status = HT8219_MixSetChan1CoefCos(ph, 0x1234);

if (status != HT8219_SUCCESS) {

    return status;

}
```

3.2.22. HT8219_MixGetChan1CoefCos

原型

```
STDHT8219CALL HT8219_MixGetChan1CoefCos(HT8219HANDLE ph, HT8219_UINT32 * step)
```

功能

获取 MIX 通道 1 余弦系数。

参数说明

输入参数

ph: 板卡句柄，由函数 HT8219_OpenCard 得到。

输出参数

cos: 相位步进，32 位无符号数。

返回值:

成功时返回 [HT8219_SUCCESS](#)。

失败时可能返回的返回值列表:

[HT8219_BAD_PARAMETER_1](#)

[HT8219_NULL_DESCRIPTOR](#)

[HT8219_NO_BOARD](#)

[HT8219_OS_ERROR](#)

函数也可能返回其它错误码。

Code example

```
HT8219_STATUS status = HT8219_SUCCESS;

HT8219_UINT32 cos;

status = HT8219_MixGetChan1CoefCos(ph, &cos);

if (status != HT8219_SUCCESS) {

    return status;

}
```