**XT1511控制芯片**

I2C接口定义说明文档

编写：宋博伟

版本： V0.1

日期：2018-05-29

1. 主控芯片ESP32
   1. 主控芯片为硬件I2C，3.3V
   2. 主控I2C支持标准100kbit/s和快速400kbit/s
2. 采用标准I2C协议
   1. 7-bit寻址，第8位读/写位
   2. 一次输出传输始于START位，结束于STOP位
   3. 数据按照每次一个8-bit字节进行传输，每个传输的字节后有一个ACK位
3. 控制芯片(I2C从机)
   1. I2C从机地址0x11
   2. 只有写操作，没有读操作
   3. 从机控制寄存器地址0x00
      1. 寄存器内存储总共控制XT1511灯的个数。(即每次刷新的灯的个数)
      2. 默认为12个灯
   4. 从机灯状态地址0x01~0x63 (根据芯片实际情况可以更改，需将更改结果告知我们)

|  |  |  |  |
| --- | --- | --- | --- |
| 灯存储器地址 | R灯数据 | G 灯数据 | B 灯数据 |
| 0x01 | 0~255 | 0~255 | 0~255 |
| 0x02 | 0~255 | 0~255 | 0~255 |
| 0x03 | 0~255 | 0~255 | 0~255 |
| 0x--- | 0~255 | 0~255 | 0~255 |

0:代表灯灭，255代表灯最亮

灯存储器地址代表第n个灯，即0x01存的是第1个灯的数据

灯数据写入后只做存储不刷新灯，等待刷新指令后刷新。

单个存储器写操作方式如下：

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ST | 从机地址+W | ACK | 灯存储器地址 | ACK | R值 | ACK | G值 | ACK | B值 | ACK | SP |

连续存储器写操作方式如下：(地址自动增加，超出范围无效)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ST | 从机地址+W | ACK | 灯存储器地址 | ACK | R值 | ACK | G值 | ACK | B值 | ACK |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| R值 | ACK | G值 | ACK | B值 | ACK |

……此次省略N个灯

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| R值 | ACK | G值 | ACK | B值 | ACK | SP |

* 1. 指令
     1. 0xA0，所有灯全部关掉，立即刷新

指令操作方式如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ST | 从机地址+W | ACK | 指令0xA0 | ACK | SP |

* + 1. 0xA1，按照灯状态存储器内的灯数据立即刷新

指令操作方式如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ST | 从机地址+W | ACK | 指令0xA1 | ACK | SP |

* + 1. 0xA2，按照预存灯信息立即刷新灯数据

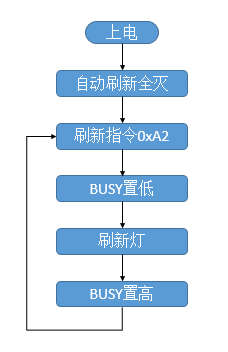
指令操作方式如下：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ST | 从机地址+W | ACK | 指令0xA2 | ACK | 预存图案编号 | ACK | SP |

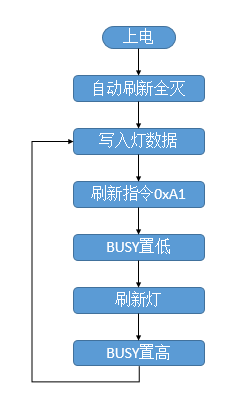
预存图案编号从0x01开始递增

* 1. 灯刷新时禁止I2C通信
     1. 刷新前将I0口(P3.5)拉低，灯刷新完成后将I0口(P3.5)置高
  2. 每次芯片上电自动刷新一次全灭的数据

1. 流程图
   1. 按照内置灯数据刷新



* 1. 自定义灯刷新



* 1. 自定义与内置灯数据混合刷新

(略)