基于N76E003实现Wiegand协议封包

NuMicro® 8 位系列微控制器范例代码介绍

文件信息

代码简述	本范例代码基于N76E003的定时器(Timer)与通用输入输出接口(GPIO)实现收发Wiegand 协议封包
BSP 版本	N76E003_BSP_Keil_C51_V1.0.6
开发平台	NT-N76E003 V2.1

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 功能介绍

1.1 简介

Wiegand通讯协议由摩托罗拉公司制定，广泛应用于门禁系统的读卡器和电子卡片，为现今的工业标准，最常见的格式为标准的26-bit封包。本范例代码基于N76E003的定时器(Timer)与通用输入输出接口(GPIO)实现收发26-bit的Wiegand协议封包。

1.2 Wiegand 协议

Wiegand界面是一种通讯协议，由3根线组成 Data0, Data1, Data return。Data0与Data1在没有数据输出时保持+5V高电平。若输出为0，则D0拉低一段时间；若输出为1，则D1拉低一段时间。

1.3 Wiegand 协议结构

标准Wiegand协议是由26位二进制数组成，定义如下：

Bit	含义
Bit 1	作为 BIT[2:13] (A Data) 隐含码(HID, Hidden ID code)的奇偶校验位
Bit [2:9]	低8位 (A Data)，作为隐含码使用，Bit 2 是MSB
Bit [10:25]	高16位 (B Data)，作为公开码(PID, Public ID code)使用 Bit 10 是MSB
Bit [26]	作为BIT [14:25] (B Data) 公开码的奇校验位

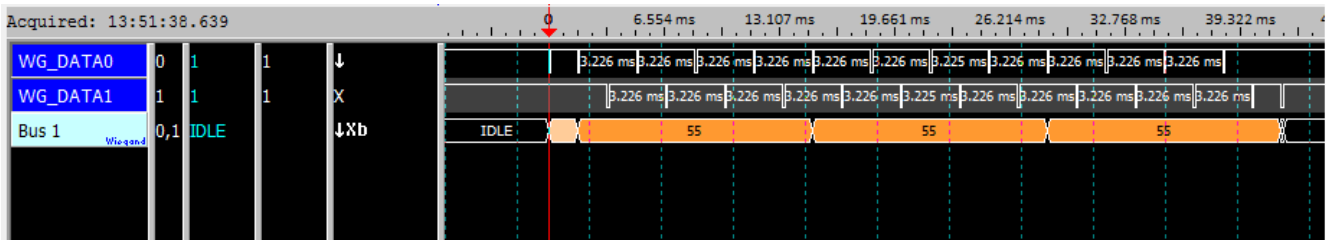
组合为26位数据如下

1	2	3	4	5		6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
P	A	A	A	A		A	A	A	A	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	P
		Even parity Field								Odd Parity Field																
		Facility Code								Card Number																

Public ID code (PID): 不同卡片具有不同的公开码，以区别用户。
Hidden ID code (HID): 隐含码存放在卡片中，也存放于读卡器中，假如卡片与读卡器中的HID码不同的话，卡片将无法在读卡器上使用。.

1.4 执行结果

送出 0x55, 0x55, 0x55 后量测到的 Weigand Bus 结果



2 代码介绍

主程序：

```
/*---------------------------------------------------------------------------*/
/* Global variables                                                          */
/*---------------------------------------------------------------------------*/
unsigned char out[3] = {0x55, 0x55, 0x55};

/*---------------------------------------------------------------------------*/
/* Functions                                                                 */
/*---------------------------------------------------------------------------*/
void main(void)
{
    //Keep wiegand bus on high state.
    P05 = 1;
    P06 = 1;
    //initial all gpio on quasi mode
    Set_All_GPIO_Quasi_Mode;
    delay_1500us();
    //send wiegand package
    send_wiegand26(out);
    while(1);
}
```

发送 26 位数据的程序

```
void send_wiegand26(unsigned char *str)
{
    unsigned char data i;
    unsigned  char check_temp;
    unsigned char data check;
    bit  even;
    bit  odd;
    unsigned char data wiegand[3];

    wiegand[0] = *str;
    wiegand[1] = *(str + 1);
    wiegand[2] = *(str + 2);
    check = 0;
    check_temp = wiegand[0];
    //even data check for data 2-9
    for(i = 0; i < 8; i++)
    {
        if((check_temp & 0x80) == 0X80)
        {
            check++;
        }
        check_temp = check_temp << 1;
    }
    if(check % 2 == 0)
        even = 0;
    else
        even = 1;
    check = 0;
    //odd data check for data 10-25
    check_temp = wiegand[1];
    for(i = 0; i < 8; i++)
    {
        if(check_temp & 0x80 == 0X80)
```

```
        {
            check++;
        }
        check_temp = check_temp << 1;
    }
    check_temp = wiegand[2];
    for(i = 0; i < 8; i++)
    {
        if(check_temp & 0x80 == 0X80)
        {
            check++;
        }
        check_temp = check_temp << 1;
    }
    if(check % 2 == 0)
        odd = 1;
    else
        odd = 0;
    //send header ep
    if(even)
    {
        WG_send_bit_1();
    }
    else
    {
        WG_send_bit_0();
    }
    //send data
    for(i = 0; i < 24; i++)
    {

        if((wiegand[0]) & 0x80)
        {
            WG_send_bit_1();
        }
        else
        {
            WG_send_bit_0();
        }
        (*(long*)&wiegand[0]) <<= 1;
    }
    //send op
    if(odd)
    {
        WG_send_bit_1();
    }
    else
    {
        WG_send_bit_0();
    }
}           …
          GPIO_CLR_INT_FLAG(PE, BIT0);
       }
```

3 软件与硬件环境

- **软件环境**
  - BSP 版本
    - N76E003_BSP_Keil_C51_V1.0.6
  - IDE 版本
    - Keil uVersion 4.6 及 PK51 Development Kit V9.52
- **硬件环境**
  - 硬件需求
    - NT-N76E003 V2.1 两组
  - 接线图

VCC	↔	VCC
P05	↔	P05
P06	↔	P06
GND	↔	GND
NT-N76E003 V2.1 (主设备)		NT-N76E003 V2.1 (从设备)

4 目录信息

- N76E003_Wiegand_V1.0
  - Common: N76E003 common function library.
  - Include: N76E003 code header.
  - Sample_Code
    - ExampleCode： Source file of example code
      - Wiegand_Master: Example code source file of wiegand master.
      - Wiegand_Slave: Example code source file of wiegand slave.
  - Startup
    - N76E003: startup code.

5 如何执行范例程序

1. 本项目支持Keil uVersion 4.6及以上版本
2. 根据目录信息进入Sample_Code\ExampleCode文件夹，双击Wiegand_Master与Wiegand_Slave文件夹中的Wiegand_Master.uvproj与Wiegand_Slave.uvproj
3. 进入编译模式接口
   a. 编译
   b. 下载代码至内存
   c. 进入 / 离开除错模式
4. 进入除错模式接口
   a. 可别进入除错模式进行调试
   b. 执行代码

6 修订纪录

日期	版本	描述
Sep. 24, 2019	1.00	1. 初始发布.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.