

## 例程二十三 获取 STM8S 系列的芯片的 ID 号

在 STM8 系列的芯片中，每个芯片都有一个独一无二的 ID 号。那么怎么能够获取的到这些 ID 号的呢，每个 ID 号有 96bit 组成，而 STM8 只是 8 位的单片机，所以这些 ID 号组成了 12 个字节。下面带大家看看 Datasheet 里面的描述吧。

The devices feature a 96-bit unique device identifier which provides a reference number that is unique for any device and in any context. The 96 bits of the identifier can never be altered by the user.

The unique device identifier can be read in single bytes and may then be concatenated using a custom algorithm.

The unique device identifier is ideally suited:

- For use as serial numbers
- For use as security keys to increase the code security in the program memory while using and combining this unique ID with software cryptographic primitives and protocols before programming the internal memory.
- To activate secure boot processes

**Table 14. Unique ID registers (96 bits)**

Address	Content description	Unique ID bits							
		7	6	5	4	3	2	1	0
0x48CD	X co-ordinate on the wafer	U_ID[7:0]							
0x48CE		U_ID[15:8]							
0x48CF	Y co-ordinate on the wafer	U_ID[23:16]							
0x48D0		U_ID[31:24]							
0x48D1	Wafer number	U_ID[39:32]							
0x48D2	Lot number	U_ID[47:40]							
0x48D3		U_ID[55:48]							
0x48D4		U_ID[63:56]							
0x48D5		U_ID[71:64]							
0x48D6		U_ID[79:72]							
0x48D7		U_ID[87:80]							
0x48D8		U_ID[95:88]							

给出了 ID 号的地址，我们就容易办了。只要往那个地址里面取出那个数就行。下面看看我的主函数先吧。

```
int main(void)
{
    /* Infinite loop */
    /*设置内部时钟16M为主时钟*/

    CLK_HSIPrescalerConfig(CLK_PRESCALER_HSIDIV1);
    /*!<Set High speed internal clock */

    Uart_Init();
    Get_ChipID();
    USART1_printf( "\r\n芯片的唯一ID为: %d-%d-%d-%d-%d-%d-%d-%d-%d-%d-%d \r\n",\
        ChipUniqueID[0],ChipUniqueID[1],ChipUniqueID[2],ChipUniqueID[3],\
        ChipUniqueID[4],ChipUniqueID[5],ChipUniqueID[6],ChipUniqueID[7],\
        ChipUniqueID[8],ChipUniqueID[9],ChipUniqueID[10],ChipUniqueID[11]);

    while (1)
    {

    }
}
```

在这里我是用十进制来表示每一个字节的。

```
/* *****
 * 函数名: Get_ChipID
 * 描述 : 获取芯片ID
 * 输入 : 无
 * 输出 : 无
 * ***** */
void Get_ChipID(void)
{
    ChipUniqueID[0] = *(__IO u16 *) (0X48CD);
    ChipUniqueID[1] = *(__IO u16 *) (0X48CE);
    ChipUniqueID[2] = *(__IO u16 *) (0X48CF);
    ChipUniqueID[3] = *(__IO u16 *) (0X48D0);
    ChipUniqueID[4] = *(__IO u16 *) (0X48D1);
    ChipUniqueID[5] = *(__IO u16 *) (0X48D2);
    ChipUniqueID[6] = *(__IO u16 *) (0X48D3);
    ChipUniqueID[7] = *(__IO u16 *) (0X48D4);
    ChipUniqueID[8] = *(__IO u16 *) (0X48D5);
    ChipUniqueID[9] = *(__IO u16 *) (0X48D6);
    ChipUniqueID[10] = *(__IO u16 *) (0X48D7);
    ChipUniqueID[11] = *(__IO u16 *) (0X48D8);
}
```

这个函数才是真正要掌握的东西。`*(__IO u16 *) (0X48CD)`这句话是什么意思，怎样理解的呢。这句话就是去一个地址里面的数据，先是把 0X48CD 强制转换为一个地址，然后在取这个地址里面的数据。执行完这个函数的，就会把 96bit，也就是 12Byte 的数据存放在 ChipUniqueID[12]这个数组里面。在主函数那里把它读出来就行。

风驰电子祝您学习愉快~~~~!!!!!!