

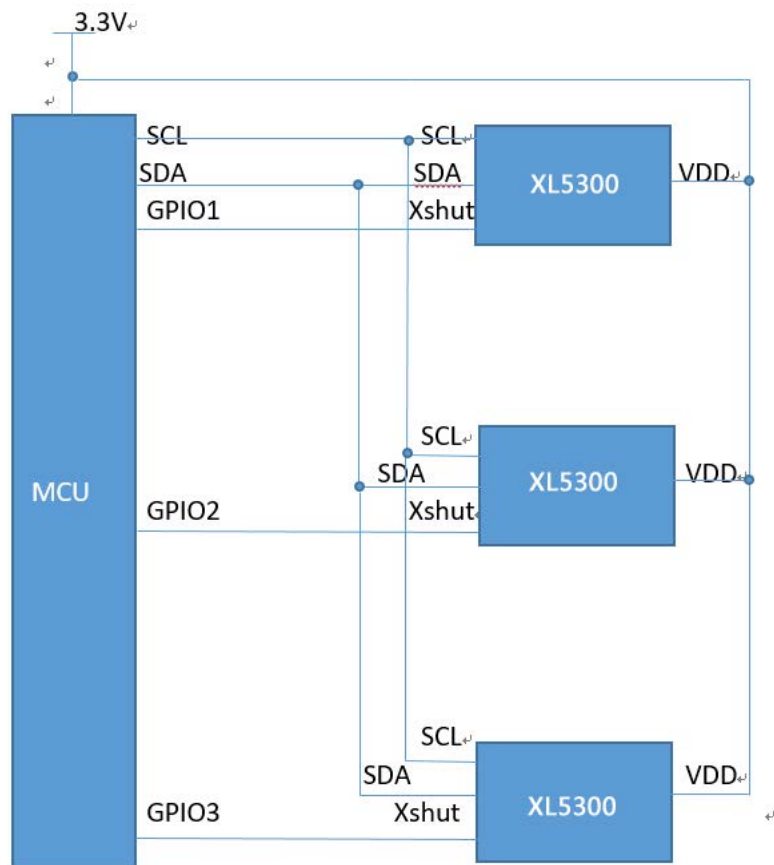
XL5300一托多使用

目录

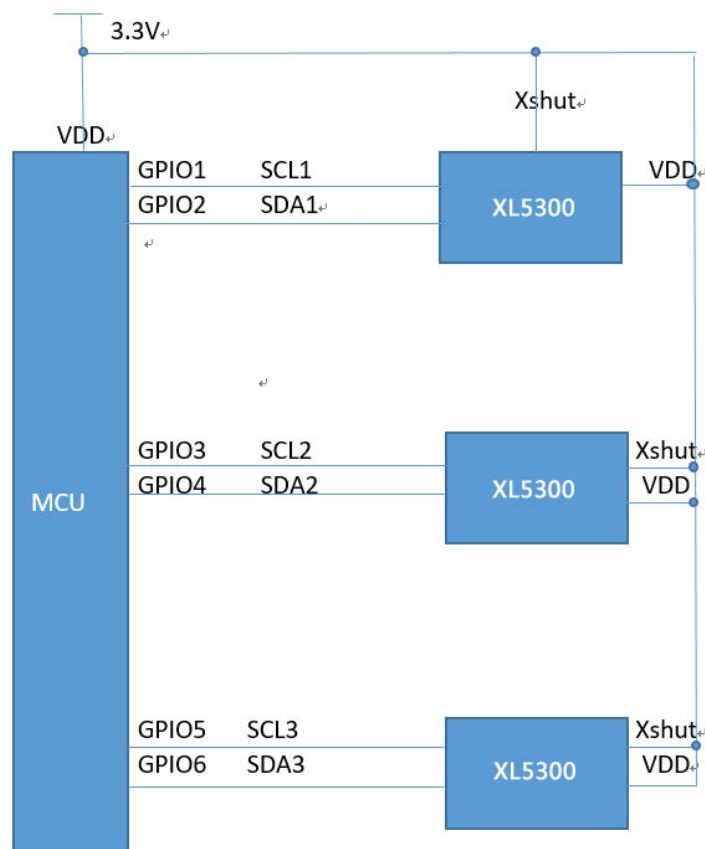
- XL5300—托多概念以及对应2种方式
- XL5300—托多驱动流程
- XL5300—托多代码说明

XL5300—一托多概念以及对应2种方式

一托多是指一个主控同时使用两个以上XL5300芯片，有以下2种方式。由于初始化时下载固件，需要一定时间，多芯片使用时，由于MCU一般是单线程时间是累加的。

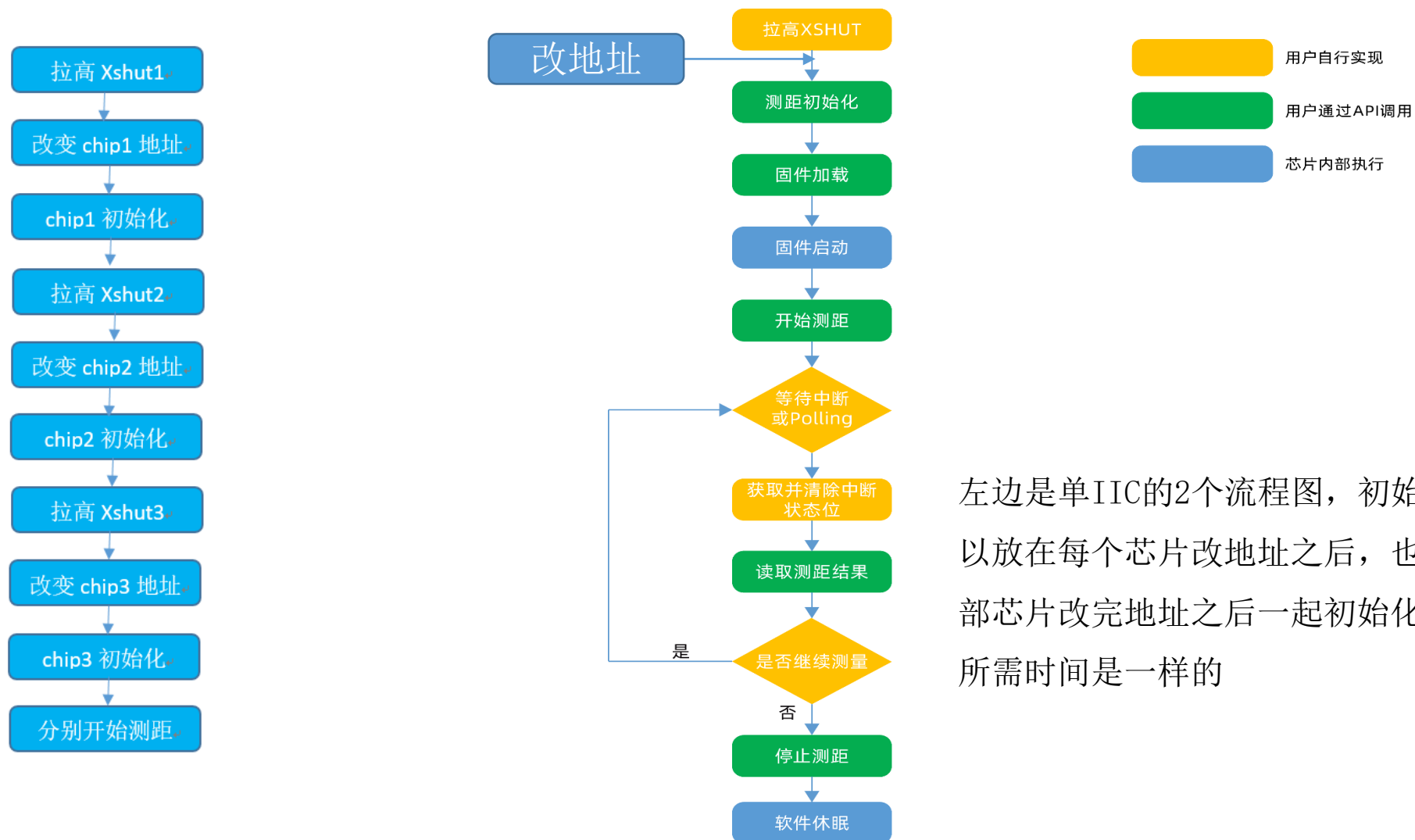


共用IIC总线



多IIC总线

XL5300—托多驱动流程



左边是单IIC的2个流程图，初始化部分可以放在每个芯片改地址之后，也可以在全部分芯片改完地址之后一起初始化。初始化所需时间是一样的

XL5300—托多代码说明

```
#ifdef CHANGE_MODEL_ADDRESS
//IIC modify
XL5300_Device_Adress_Modify(XSHUT_Pin1, XL5300_MODEL1_ADDR);
XL5300_init();
XL5300_Start_Continuous_Measure();
gSalve=VI5300_DEVICE_ADDR;
XL5300_Device_Adress_Modify(XSHUT_Pin2, XL5300_MODEL2_ADDR);
#else
XL5300_Chip_PowerON(XSHUT_Pin1);
if(XL5300_Device_Check()!=0xD8)
{
    printf("Check device ID fail!\r\n");
}
#endif
XL5300_init();
//XL5300_Start_XTalk_Calibration();
//XL5300_Start_Offset_Calibration();
XL5300_Start_Continuous_Measure();
.....
```

```
void XL5300_Chip_PowerON(uint16_t pin)
{
    HAL_GPIO_WritePin(XSHUT_GPIO_Port, pin, GPIO_PIN_RESET);//set 0
    HAL_Delay(10);//delay 30ms
    HAL_GPIO_WritePin(XSHUT_GPIO_Port, pin, GPIO_PIN_SET);//set 1
    HAL_Delay(20);
}
uint8_t XL5300_Device_Check(void)
{
    uint8_t Chip_id = 0;

    ReadOneReg(0x06, &Chip_id);

    return Chip_id;
}
```

```
uint8_t XL5300_Device_Adress_Modify(uint16_t pin,uint8_t addr_val)
{
    uint8_t err_sta = 0x00;

    XL5300_Chip_PowerON(pin);
    if(XL5300_Device_Check()!=0xD8)
    {
        printf("Check device ID fail!\r\n");
        return 0x01; //返回错误状态
    }

    err_sta = WriteOneReg(0x01,0x88);
    if(err_sta!=0x00)
    {
        return 0x01; //返回错误状态
    }
    err_sta = WriteOneReg(0x06,addr_val);
    if(err_sta!=0x00)
    {
        return 0x02; //返回错误状态
    }
    gSalve = addr_val;
    XL5300_Device_Check();
    return 0x00;
}
```

XL5300—托多代码说明

```
XL5300_Status I2C_2V1_WriteOneReg(uint8_t addr, uint8_t value)
{
    XL5300_Status ret;
    ret = vi_sw_writereg(gSalve, addr, value);
    if(ret == 1)
        ret = XL5300_OK;
    else if(ret == 0)
        ret = XL5300_ERROR;

    return ret;
}

XL5300_Status I2C_2V1_ReadOneReg(uint8_t addr, uint8_t *value)
{
    XL5300_Status ret;
    ret = vi_sw_readreg(gSalve, addr, value, 1);
    if(ret == 1)
        ret = XL5300_OK;
    else if(ret == 0)
        ret = XL5300_ERROR;

    return ret;
}
```

左边图中gSalve是全局变量的IIC地址，操作不同芯片时，改变这个变量地址就可以操作，不需要全部IIC函数全部重写

```
while (1)
{
    HAL_Delay(50);
    #ifdef CHANGE_MODEL_ADDRESS
    XL5300_Get_Measure_Data(&result2);
    gSalve = XL5300_MODEL1_ADDR;
    XL5300_Get_Measure_Data(&result1);
    gSalve = XL5300_MODEL2_ADDR;
    printf("TOF1 =%6d, confidence1=%d,TOF2 =%6d, confidence2=%d\n", result1.millimeter,
        result1.confidence, result2.millimeter, result2.confidence);
    #else
    XL5300_Get_Measure_Data(&result1);
    printf("TOF1 =%6d, confidence1=%d\n", result1.millimeter, result1.confidence);
    #endif
}
```



Thank you!

深圳市芯岭技术有限公司
www.xinlinggo.com

地址：深圳市龙华区观澜街道桂花社区观光路1233号君澜大厦二单元1416室
咨询电话: 0755-32866130