Switch(GunX\_Display)//GunX\_ChgSta全部由A7控制，Display由自己控制

{

case GUN\_B\_DISPLAY（枪A显示）：

1. GunA\_ChgSta = CHG\_STA\_FREE：
   1. 显示空闲状态（全灭）；
   2. 如果GunB\_ChgSta或者GunC\_ChgSta = CHG\_STA\_PLUG即枪插入，则Display = GUN\_B\_DISPLAY或GUN\_C\_DISPLAY；
   3. Display = GUN\_B\_DISPLAY；
2. GunA\_ChgSta = CHG\_STA\_PLUG：
   1. 禁止按键中断，显示环效果1（**阻塞**），GunA\_ChgSta = CHG\_STA\_WAIT，使能按键中断；
3. GunA\_ChgSta = CHG\_STA\_WAIT：
   1. 显示等待充电中；
   2. 如果GunB\_ChgSta或者GunC\_ChgSta = CHG\_STA\_PLUG即枪插入，则Display = GUN\_B\_DISPLAY或GUN\_C\_DISPLAY；
   3. 如果"A/B/C"键按下（对应的isPressed = 1），Display = GUN\_B\_DISPLAY或者GUN\_C\_DISPLAY；
4. GunA\_ChgSta = CHG\_STA\_START：
   1. 禁止按键中断，显示环效果2（**阻塞**），GunA\_ChgSta = CHG\_STA\_CHG，使能按键中断；
5. GunA\_ChgSta = CHG\_STA\_CHG：
   1. 显示充电中（加上充电图标的定时闪烁），并开始计时；
   2. 如果GunB\_ChgSta或者GunC\_ChgSta = CHG\_STA\_PLUG即枪插入，则Display = GUN\_B\_DISPLAY或GUN\_C\_DISPLAY；
   3. 如果超时6s则Display = GUN\_B\_DISPLAY；
   4. 如果"A/B/C"键按下（对应的isPressed = 1），Display = GUN\_B\_DISPLAY或者GUN\_C\_DISPLAY；
6. GunA\_ChgSta = CHG\_STA\_ERR：
   1. 显示ERR，并开始计时；
   2. 如果GunB\_ChgSta或者GunC\_ChgSta = CHG\_STA\_PLUG即枪插入，则Display = GUN\_B\_DISPLAY或GUN\_C\_DISPLAY；
   3. 如果超时6s则Display = GUN\_B\_DISPLAY；
   4. 如果"A/B/C"键按下（对应的isPressed = 1），Display = GUN\_B\_DISPLAY或者GUN\_C\_DISPLAY；

case GUN\_B\_DISPLAY（枪B显示）：

（类似枪A）

case GUN\_C\_DISPLAY（枪C显示）：

（类似枪A）

}

以下程序流程可以不用阻塞，同级状态是枪（一次轮询只能显示其中一把枪）和按键灯：“枪不能同时显示，枪和按键灯可以同时显示”。

While(1)

{

switch(Display)//放在GunStaDisplay模块中实现

{

case A:

case B:

case C:

}

If(Keyx.isPressed)

{

Key Display;

}

}

while(1)

{

Gun\_Display();

KeyLED\_Display();

}