RF 1对6　PC Joystic_VR

作者：韦宇
部门：PE
时间：2009年10月26日

1. 概要（General description）

随着技术的不断发展，很多经典的电视游戏被移植到了电脑上，或者是通过模拟器的方式在走进电脑游戏玩家的生活中。不过在经过一段时间之后，很多用户都发现，这些原本是通过游戏手柄在操作的游戏，在很多时候使用鼠标和键盘并不能完美的进行控制，而这时候使用无线游戏手柄就是进行游戏的最好解决方案。

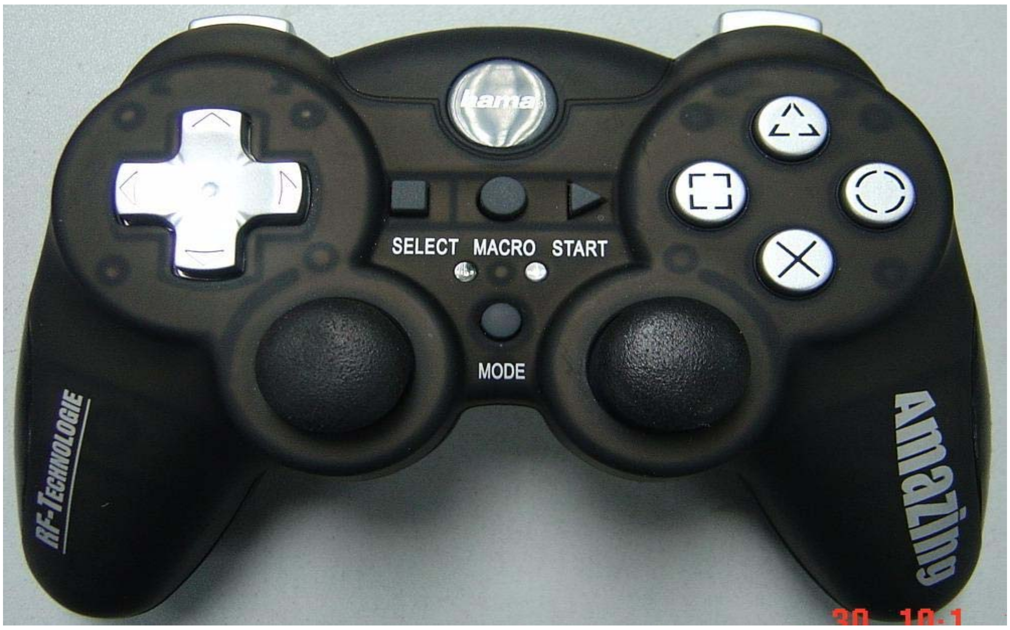
2. 特征（Feature）

采用 2.4GHz 射频无线连接方式，摆脱了连线的困扰。手柄表面质感好，配置简单，按键手感舒适，且反应较为灵敏。工作方式里，由于 2.4G 频段使用带宽越来越紧张，造成器件间波段干扰也越来越严重，由此需要经过新技术来解决此问题，其中最有效的办法就是使用“跳频”通信方式。本工程，使用一个主机六个分机进行跳频通信，完成了一体化设计的工程。下述协议与一对多兼容。主机端开发有USB通信口，如果客户需要主机的显示终端，可以再主机端做修改，而手柄在此协议下无须修改。

控制核心，手柄使用: EM78P520+EM198850; 主机使用: EM78M611+EM198850

3. 功能描述（Function descriptions）

游戏手柄采用了传统的布局方式，最左边是一个八方向的方向键，最右侧是四个按键，呈菱形分布，上面分别印着三角、方块、圆圈和叉子的标记，这和 PS/PS2 手柄的按键完全一样。在中间的则 4 个功能按键和两个迷你型的摇杆，其中 4 个按键分别对应选择（SELECT）、开始（START）以及模式选择键（MODE）和数字/模拟工作方式切换键（MACRO）。另外在手柄的顶部各有两个按钮，分别对应L1、L2和R1、R2。
原理图(见附图1,附图2).



RF游戏手柄

demo 样机

4. 电气特性（Electrical Characteristics）

- 工作环境温度：-10℃～80℃。
- 工作环境湿度：30%～95%RH。
- 工作频点：2402MHz － 2482MHz
- 供电电压
  采用 3V 电压供电系统
- 耗电电流
  搜索状态：　　　25.0-30.0mA
  正常工作时：　　Imax≤10mA (VDD=3.0V)
  待机状态：　　　Imax≤3mA (VDD=3.0V，摇杆耗电大约 2mA)
- 功耗
  正常工作时：　　Wmax≤150MW (VDD=3.0V)
- 通讯距离：
  正常通讯时：　　Lmin≥10M (VDD=3.0V，办公室环境下)
- 对码过程中，采取先到先分配原则。即对码时需要先关闭所有手柄，此时主机开启强制通讯模式，依次打开手柄的对码模式，此顺序即为手柄通道的顺序，要更改必须重新对码。

5. Communication Protocol format（通信协议格式）

以主机接收端【sink】为基准，其时序如下，20ms 为一个周期（cycle），

第一时段(ComucycleNum = 0)用于时钟同步；

第二时段(ComucycleNum = 1)分配给 发送端【transmitter】1 (GamePad1)

第三时段(ComucycleNum = 2)分配给 发送端【transmitter】2 (GamePad2)

第四时段(ComucycleNum = 3)分配给 发送端【transmitter】3 (GamePad3)

第五时段(ComucycleNum = 4)分配给 发送端【transmitter】4 (GamePad4)

第六时段(ComucycleNum = 5)分配给 发送端【transmitter】5 (GamePad5)

第七时段(ComucycleNum = 6)分配给 发送端【transmitter】6 (GamePad6)

周期号	0	1		2		3		4		5		6	
ComuClock	0-7	0-7		8-15		16-23		24-31		32-39		40-47	
时间	8ms	8ms		8ms		8ms		8ms		8ms		8ms	
接收端 (RX)	CMD_TX	RX		RX		RX		RX		RX		RX	
	TXD6 USB IN		TXD1 USB IN		TXD2 USB IN		TXD3 USB IN		TXD4 USB IN		TXD5 USB IN		TXD6 USB IN
发送端 1 (TX1)	CMD_RX	TX	keyscan CMOS GS										
发送端 2 (TX2)	CMD_RX			TX	Keyscan CMOS GS								
发送端 3 (TX3)	CMD_RX					TX	Keyscan CMOS GS						
发送端 4 (TX4)	CMD_RX							TX	Keyscam CMOS GS				
发送端 5 (TX5)	CMD_RX									TX	Keyscan CMOS GS		
发送端 6 (TX6)	CMD_RX											TX	Keyscan CMOS GS

周期号：　　　根据变量 ComucycleNum 判断

通信时钟：　　根据变量 ComuClock 判断

NUM1:　　　　主机发给手柄，用于同步的信息命令包，给手柄调整时间，确保手柄和主机同步

Num2-Num7:　　手柄回复给主机的数据封包，传输所需的资料

NUM8:　　　　预留时间给主机和手柄做数据处理。主机 USB 上传资料，手柄处理按键信息等

6. Synchronization information packet format（同步帧数据格式）

Offset	Size	Field	Note
0	1 BYTE	Length	0X10
1	1 BYTE	RX_IDH	Rand data, RX ID
2	1 BYTE	RX_IDL	
3	1 BYTE	CHN_FLAG	Gamepad status,each bit will stand for a Gamepad status
	Bit【0..5】	Gamepad X status	6 Gamepad status
	Bit【6..7】	reserve	reserve
4	1 BYTE	CommuStatusFlag	Communication function flag
	Bit【7】	DescriptorFinishFlag	
	Bit【6】	FccTestModeFlag	FCC test flag
	Bit【5】	ForceLinkModeFlag	
	Bit【4】	LinkModeFlag	
	Bit【3】	EEpromWRStatusFlag	
	Bit【2】	LoseFrameStatusFlag	
	Bit【1】	NormalStatusFlag	
	Bit【0】	SearchStatusFlag	
5	1 BYTE	DirectionCtrl	0:sink to transmitter　1:transmitter to sink
6	1 BYTE	N_CHN	Total Gamepads and Frequency index N_CHN= ((TotalGamepads<<4) & 0xF0)｜(CH_NO & 0x0F) TotalGamepads defined by Sink
7	1 BYTE	TX1_ID	Gamepad1 ID
8	1 BYTE	TX2_ID	Gamepad2 ID
9	1 BYTE	TX3_ID	Gamepad3 ID
10	1 BYTE	TX4_ID	Gamepad4 ID
11	1 BYTE	TX5_ID	Gamepad5 ID
12	1 BYTE	TX6_ID	Gamepad6 ID
..	..	..	..
24	1 BYTE	Ver. num	Defined by sink

NOTE:

1) RX_IDH、RX_IDL 两个 BYTE 是由主机在对码时随机生成的数据;

2）TX1_ID、TX2_ID、TX3_ID、TX4_ID、TX5_ID、TX6_ID

TX1_ID = (RX_IDL & 0xF0) | 0x01 ; …… ;

TX6_ID = (RX_IDL & 0xF0) | 0x06

3) 帧同步头使用 7 BYTE: RX_IDH、RX_IDL、CHN_FLAG、CommunicateStatusFlag、DirectionCtrl、N_CHN

4) 初始化与同步过程描述:

主机: 读取 EEPROM 中 RX_IDH、RX_IDL、TX1_ID、TX2_ID、TX3_ID、TX4_ID、TX5_ID、TX6_ID 数据，保存到 MCU 中;

当同步对码按键（MACRO）按下时，进入强制通信模式，随机生成 RX_IDH、RX_IDL，运算后可得 TXx_ID,

SYNC 使用默认值 0xFF、0xFF，依次开启手柄完成对码;

退出强制通信模式后，把 RX_IDH、RX_IDL、TX1_ID、TX2_ID、TX3_ID、TX4_ID、TX5_ID、TX6_ID 写入 EEPROM;

RF 中使用主机 ID【RX_IDH、RX_IDL】重写 DataAddressInRF（reg51-reg57)。

手柄: 读取 EEPROM 中 RX_IDH、RX_IDL、TXx_ID（本机 ID）数据,保存到 MCU 中;

当同步对码按键（MACRO）按下时，进入强制通信模式，首次通信上时接收到的 RX_IDH、RX_IDL、TXx_ID

保存在 MCU 中，自动结束强制通信模式，把 RX_IDH、RX_IDL、TXx_ID 写入 EEPROM;

RF 中使用主机 ID【RX_IDH、RX_IDL】重写 DataAddressInRF（reg51-reg57);

7. Communication information packet format（通讯帧数据格式）

Offset	Size	Field	Note
0	1 BYTE	Length	0x17
1	1 BYTE	RX_IDH	Rand data, RX ID
2	1 BYTE	RX_IDL	
3	1 BYTE	CHN_FLAG	Gamepad status,each bit will stand for a gamepad status
4	1 BYTE	CommuStatusFlag	Communication function flag
5	1 BYTE	DirectionCtrl	0:sink to transmitter 1:transmitter to sink
6	1 BYTE	TX_ID	Transmitter ID. Share the position of N_CHN
7	1 BYTE	Data1	rocker left-x(left-right) RF transmitter data
8	1 BYTE	Data2	rocker left-y(up-down) RF transmitter data
9	1 BYTE	Data3	rocker right-x(left-right) RF transmitter data
10	1 BYTE	Data4	rocker right-y(up-down) RF transmitter data
11	1 BYTE	Data5	Bit0 bit1 bit2 bit3 bit4 bit5 bit6 bit7 A_1 B_2 C_3 D_4 L1_5 R1_6 L2_7 R2_8
12	1 BYTE	Data6	bit7 bit6 Bit5 Bit4 Bit3 bit2 bit1 bit0 SELECT_9 START_10 LSW_11 RSW_12 MODE_13 MACRO_14 TEST1 TEST2 MODE: 1:DealWithDigital 0:DealWithAnalog (default:1)
13	1 BYTE	Data7	
	Bit【7..4】	Hat Switch	000:00’ 001:45’ 010:90’ 011:135’ 100:180’ 101:225’ 110:270’ 111:315’
	Bit【3..0】	reserve	Constan = 0
14	1 BYTE	X_Axis_H	reserve
15	1 BYTE	X_Axis_L	reserve
16	1 BYTE	Y_Axis_H	reserve
17	1 BYTE	Y_Axis_L	reserve
18	1 BYTE	Z_Axis_H	reserve
19	1 BYTE	Z_Axis_L	reserve
20	1 BYTE	VersionNum	Software version number
..	..	..	..
24	1 BYTE	Ver.Num	Defined by sink

NOTE:

1）红色字体为同步信息关键字，搜索模式和正常模式必须包含

2）蓝色字体是正常传送的手柄数据，在正常模式下传送，

3）绿色字体暂时未作使用，预留用做用新增功能的数据通道，主机可以不做任何处理

4）数据流传输模式：

主机开机初始化后，在同步头信息中包含 12 byte 的数据，依次是：

RX_IDH、RX_IDL、CHN_FLAG、CommuStatusFlag、DirectionCtrl、N_CHN、TX1_ID、TX2_ID、TX3_ID、TX4_ID、TX5_ID、TX6_ID

手柄接收到同步信息后会回复：RX_IDH、RX_IDL、CHN_FLAG、CommuStatusFlag、DirectionCtrl、TXx_ID（本机 ID）

每个通讯周期后 2ms，主机都会根据本次通信周期内的相关信息修改同步头数据，为下一次联机做准备；

手柄再次接收到同步信息，检测数据信息如果一切正常，就发送包含同步信息与所需数据每帧共 24 byte 数据：

RX_IDH、RX_IDL、CHN_FLAG、CommuStatusFlag、DirectionCtrl、TXx_ID、

Data1、Data2、Data3、Data4、Data5、Data6、【8 byte data】、Ver.Num

主机收到这一帧 24 byte 数据，也会做同步信息检测，如果正常，就把这些数据都收下，不正常做相应处理

8. 伪随机频点列表

EM198810 跳频频点从 2402MHz – 2482MHz，为配合认证测试，频点范围边沿两端各自内收 2.5MHz,即实际工作频点范围取：2405MHz – 2479MHz

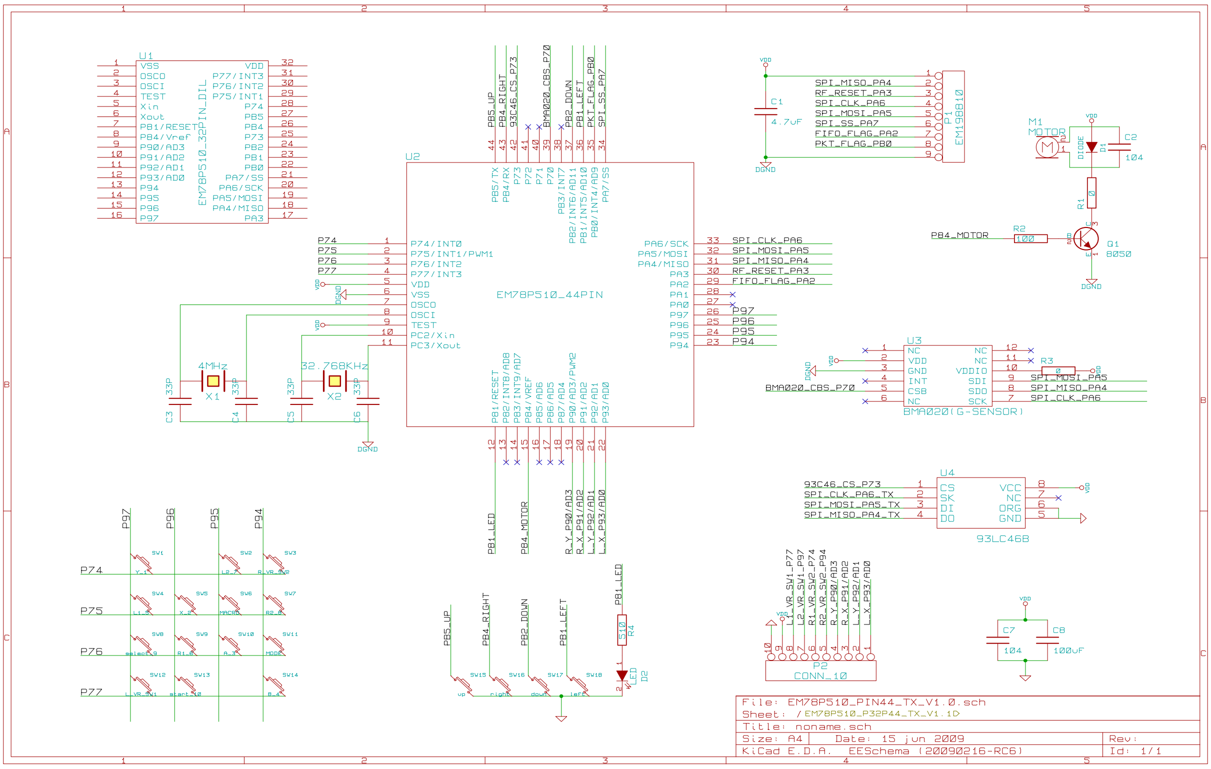
```
RF_CHNO_LIST0:
  ADD                 PC,A
  RETL                @29         ;freq:2431 ;@0X1D
  RETL                @4          ;freq:2406 ;@0X04
  RETL                @67         ;freq:2469 ;@0X43
  RETL                @32         ;freq:2434 ;@0X20

  RETL                @7          ;freq:2409 ;@0X07
  RETL                @65         ;freq:2467 ;@0X41
  RETL                @36         ;freq:2438 ;@0X24
  RETL                @10         ;freq:2412 ;@0X0A

  RETL                @62         ;freq:2464 ;@0X3E
  RETL                @38         ;freq:2440 ;@0X26
  RETL                @8          ;freq:2410 ;@0X08
  RETL                @59         ;freq:2461 ;@0X3B

  RETL                @45         ;freq:2447 ;@0X2D
  RETL                @12         ;freq:2414 ;@0X0C
  RETL                @76         ;freq:2478 ;@0X4C
  RETL                @40         ;freq:2442 ;@0X28
```

选用了 16 个频点进行伪随机，设备频点列表必须依此来跳变

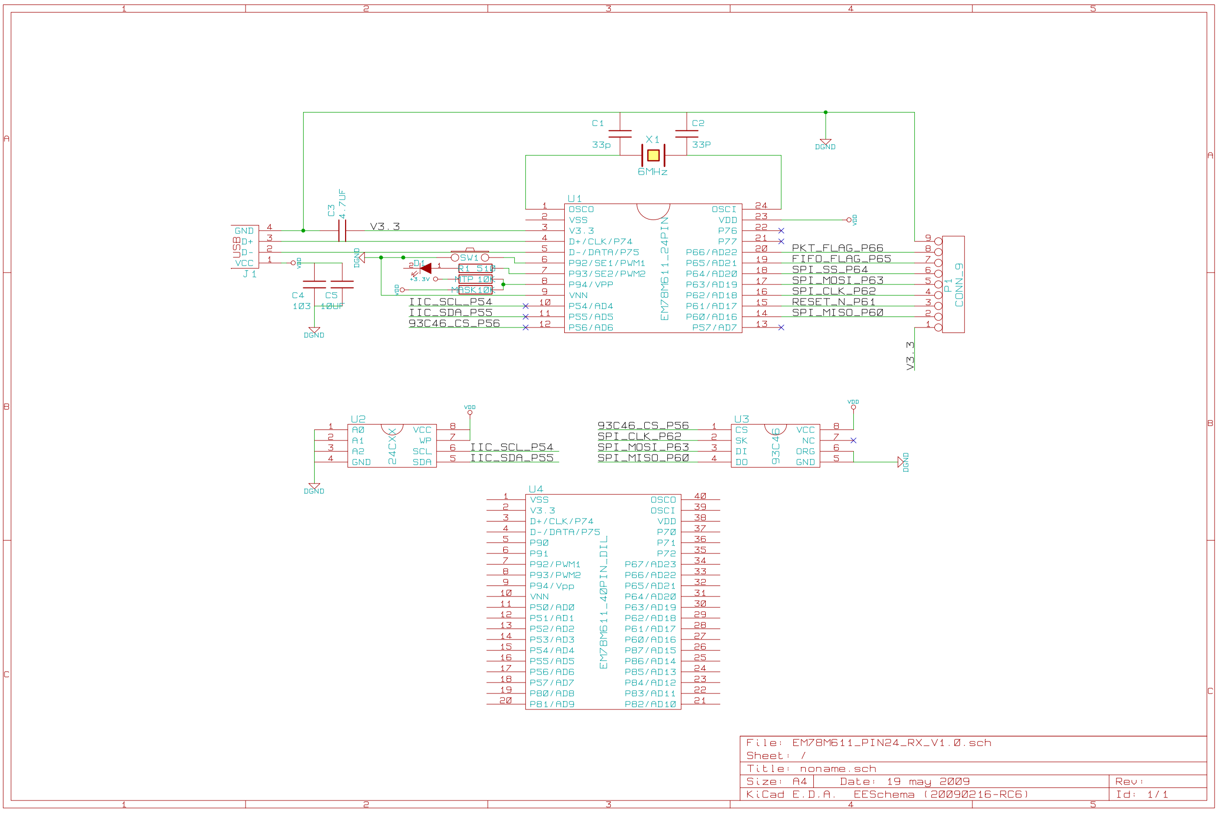
9. 心得：

已经熟练掌握义隆 EM78 系列单片机的使用方法,在此基础上已经对 EM198810 与 EM198850 模块在一对多应用上深有体会。这段时间编制的一套一对多通信协议，可以方便的移植到满足最低资源要求的其它 MCU 型号里。1 对多技术已经日趋成熟，依次为基础的通信产品可以由业务提单做推广。

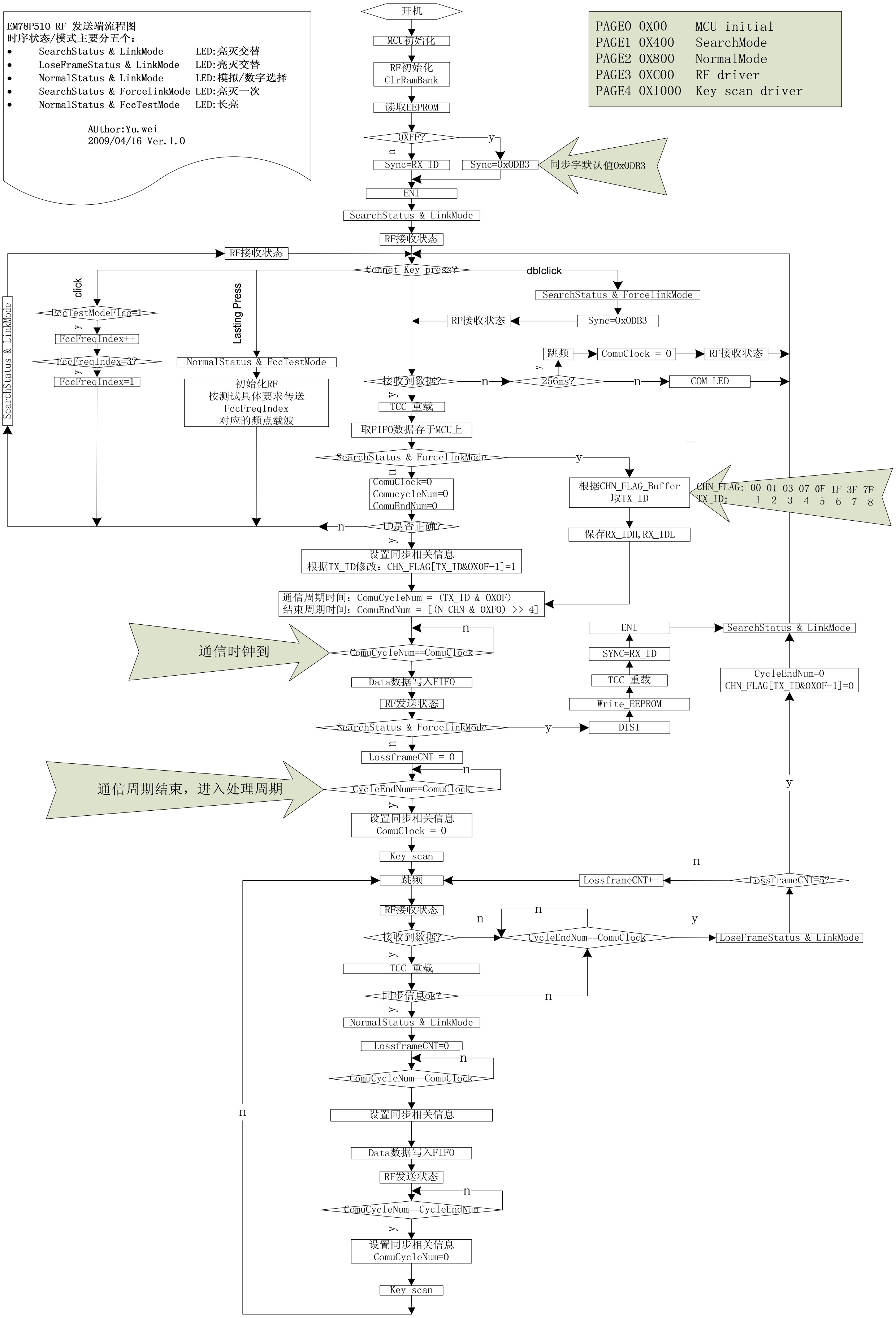
10. 原理图及流程图



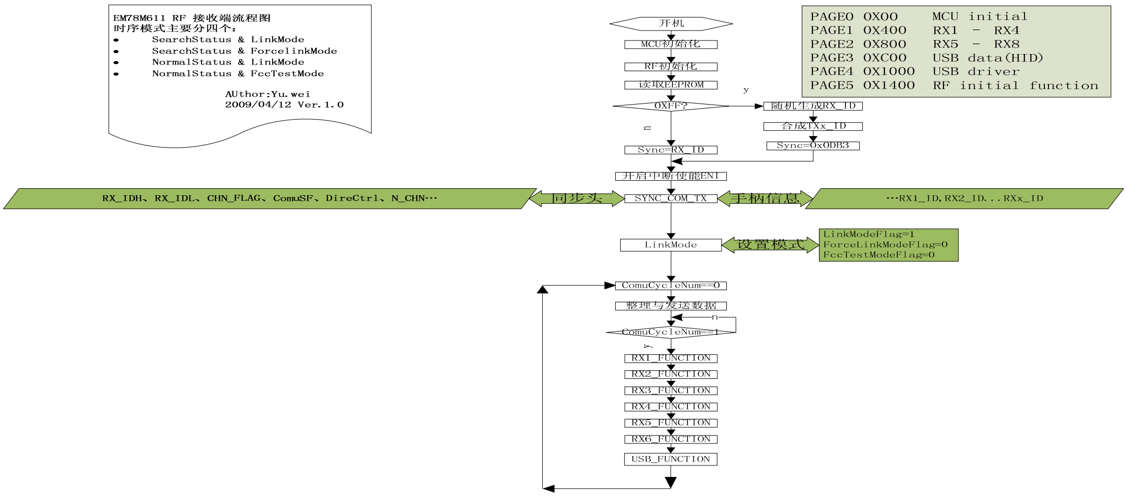
手柄原理图



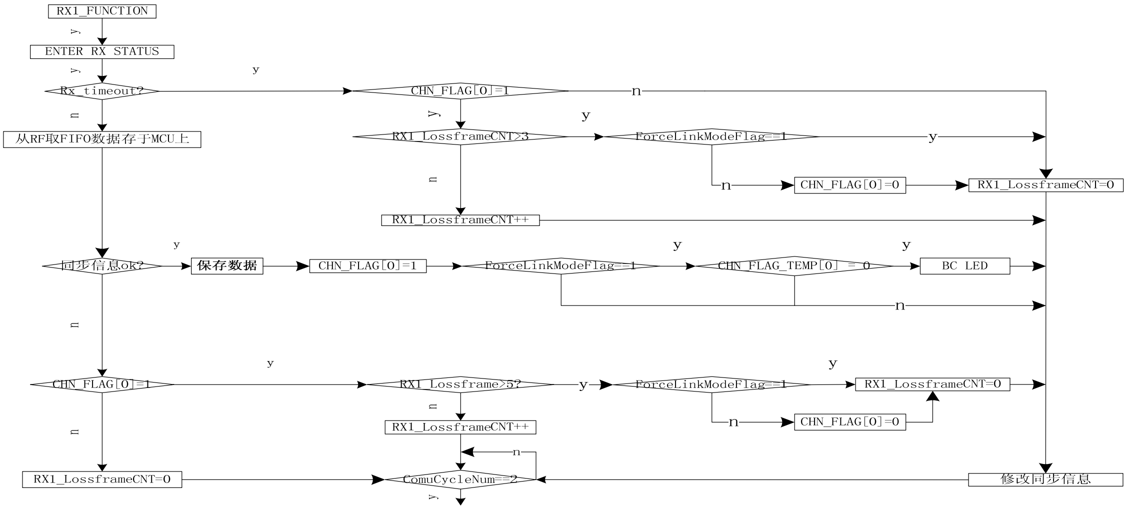
主机原理图


EM78P510 RF 发送端流程图
时序状态/模式主要分五个：
• SearchStatus & LinkMode LED:亮灭交替
• LoseFrameStatus & LinkMode LED:亮灭交替
• NormalStatus & LinkMode LED:模拟/数字选择
• SearchStatus & ForcelinkMode LED:亮灭一次
• NormalStatus & FccTestMode LED:长亮
AUthor:Yu.wei
2009/04/16 Ver.1.0
PAGE0 0X00 MCU initial
PAGE1 0X400 SearchMode
PAGE2 0X800 NormalMode
PAGE3 0XC00 RF driver
PAGE4 0X1000 Key scan driver
开机
MCU初始化
RF初始化
ClrRamBank
读取EEPROM
0XFF?
y
n
Sync=RX_ID
Sync=0x0DB3
同步字默认值0x0DB3
ENI
SearchStatus & LinkMode
RF接收状态
RF接收状态
Connet Key press?
dblclick
click
Lasting Press
SearchStatus & LinkMode
FccTestModeFlag=1
y
FccFreqIndex++
FccFreqIndex=3?
y
FccFreqIndex=1
NormalStatus & FccTestMode
初始化RF
按测试具体要求传送
FccFreqIndex
对应的频点载波
SearchStatus & ForcelinkMode
RF接收状态
Sync=0x0DB3
跳频
ComuClock = 0
RF接收状态
y
接收到数据?
n
256ms?
n
COM LED
y
TCC 重载
取FIFO数据存于MCU上
SearchStatus & ForcelinkMode
y
n
根据CHN_FLAG_Buffer
取TX_ID
CHN_FLAG: 00 01 03 07 0F 1F 3F 7F
TX_ID: 1 2 3 4 5 6 7 8
ComuClock=0
ComucycleNum=0
ComuEndNum=0
n
ID是否正确?
y
保存RX_IDH,RX_IDL
设置同步相关信息
根据TX_ID修改：CHN_FLAG[TX_ID&OXOF-1]=1
通信周期时间：ComuCycleNum = (TX_ID & 0XOF)
结束周期时间：ComuEndNum = [(N_CHN & 0XF0) >> 4]
通信时钟到
n
ComuCycleNum==ComuClock
Data数据写入FIFO
RF发送状态
SearchStatus & ForcelinkMode
y
n
LossframeCNT = 0
ENI
SearchStatus & LinkMode
SYNC=RX_ID
TCC 重载
Write EEPROM
DISI
CycleEndNum=0
CHN_FLAG[TX_ID&OXOF-1]=0
通信周期结束，进入处理周期
n
CycleEndNum==ComuClock
y
设置同步相关信息
ComuClock = 0
Key scan
y
n
跳频
LossframeCNT++
LossframeCNT=5?
RF接收状态
n
n
y
接收到数据?
CycleEndNum==ComuClock
LoseFrameStatus & LinkMode
y
TCC 重载
同步信息ok?
n
y
NormalStatus & LinkMode
LossframeCNT=0
n
ComuCycleNum==ComuClock
n
设置同步相关信息
Data数据写入FIFO
RF发送状态
n
ComuCycleNum==CycleEndNum
y
设置同步相关信息
ComuCycleNum=0
Key scan

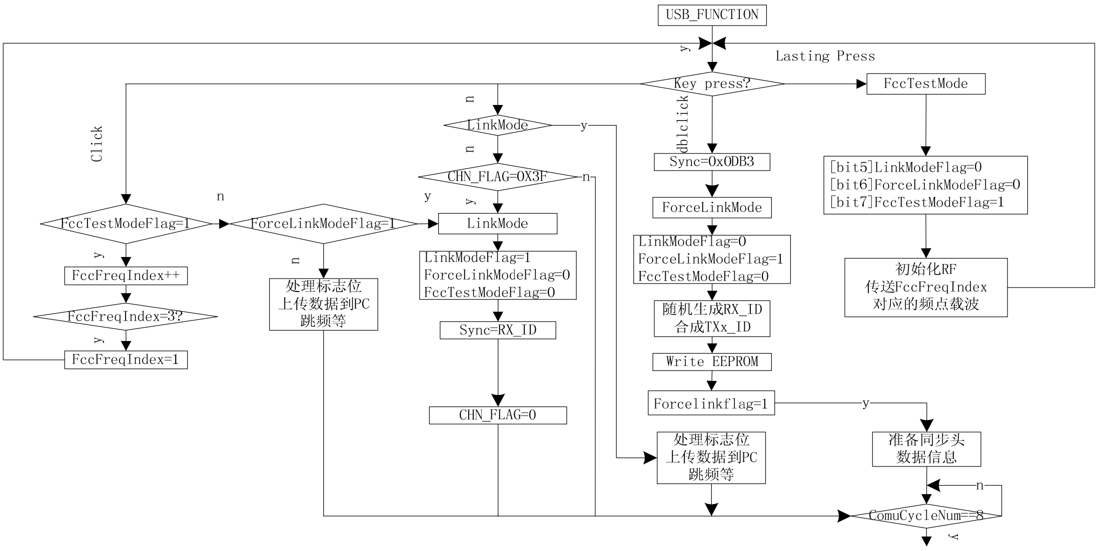
手柄流程图



主机主流程图



主机 RX1_Function



主机 Manage_Function

11 部分代码

```
/****************************************************************
* Filename        :   EM78P520_32PIN44PIN_TX.ASM
* Author          :   yu.wei
* Company         :   ELAN
* VERSION         :   1.0
* CRYSTAL         :   4MHZ
* Creat date    :   2009/9/11
* tool ver.      :   WicePlus 2.7
* Description   :   modify for code conformity
****************************************************************/
;----------------------------------------------------------------
include "D:\include\EM78xx\EM78P510.H"
include "D:\include\EM78xx\EM78Math.H"
include "D:\include\EM78xx\EM78CtrlIns.H"
include "config.h"
include "P520txP32P44V10.H"
include "EM198850_For_EM78P520.ASM"
include "P520SkipFreqFun.ASM"
include "XX93C46_For_EM78P520.ASM"
;include "FccTest.asm"

;----------------------------------------------------------------



;--------------------------MAIN PROGRAM---------------------------
        ORG                 0X00
        JMP                 INITIAL
        ORG                 0X03
        JMP                 TCC_INT
        ORG                 0X06
        JMP                 EXTERNAL_INT
        ORG                 0X09
        JMP                 WDT_INT
        ORG                 0X0C
        JMP                 TIMER1_INT
        ORG                 0X0F
        JMP                 TIMER2_INT
        ORG                 0X12
        JMP                 AD_INT
        ORG                 0X15
        JMP                 UART_INT
        ORG                 0X18
        JMP                 SPI_INT
        ORG                 0X1B
        JMP                 LVD_INT



        ORG                 0X100
;======================== TCC Interrupt Service =================================
TCC_INT:
        PUSH A_Temp,STATUS_Temp,RSR_Temp,@4,A1_Temp,STATUS1_Temp,RSR1_TEMP
        BANK                0
        BC                  ISR,TCIF                ;clear TCC interrupt flag
        JBC                 SearchStatusFlag/16,SearchStatusFlag%16
        JMP                 Search_Status_Mode
        JBC                 LoseFrameStatusFlag/16,LoseFrameStatusFlag%16
        JMP                 LoseFrame_Status_Mode
        JBC                 NormalStatusFlag/16,NormalStatusFlag%16
        JMP                 Normal_Status_Mode
        JMP                 TCC_INT_END
;-------------------------------------------------------------------------
        Normal_Status_Mode:
```

```
        MOV             A,@0x04                 ; N=125,P=32,f=4MHz ==> T=1ms
        MOV             TWTCR,A
        MOV             A,@(255-124)            ; load initial value
        MOV             TCC,A
        JMP             TCC_INT_END
;------------------------------------------------------------------
    Search_Status_Mode:
        MOV             A,@0x07                 ; N=124,P=256,f=8MHz ==> T=8ms
        MOV             TWTCR,A
        MOV             A,@(255-124)            ; load initial value
        MOV             TCC,A
        BANK            2
        INC             KeySystemTimeCNT
        MOV             A,KeySystemTimeCNT
        SUB             A,@1                    ; Control key scan clock
        JBC             STATUS,C
        JMP             TCC_INT_END
        BS              System16msFlag/16,System16msFlag%16
        CLR             KeySystemTimeCNT
        ;BANK           0
        ;MOV            A,@0X40                 ; (test)P86 exchange when intrrupt
        ;XOR            PORT8,A
        JMP             TCC_INT_END
;------------------------------------------------------------------
    LoseFrame_Status_Mode:
        MOV             A,@0x04                 ; N=125,P=32,f=4MHz ==> T=1ms
        MOV             TWTCR,A
        MOV             A,@(255-124)            ; load initial value
        MOV             TCC,A
        JMP             TCC_INT_END
TCC_INT_END:
    BANK                0
    MOV                 A,@0X80                 ; (test)P87 exchange when intrrupt
    XOR                 PORT8,A
    INC                 ComuClock
    POP A_Temp,STATUS_Temp,RSR_Temp,@4,A1_Temp,STATUS1_Temp,RSR1_TEMP
    RETI

EXTERNAL_INT:
    PUSH A_Temp,STATUS_Temp,RSR_Temp,@4,A3_Temp,STATUS3_Temp,RSR3_Temp
    BANK                1
    CLR                 EISR                    ;clear the external interrupt flag
    POP A_Temp,STATUS_Temp,RSR_Temp,@4,A3_Temp,STATUS3_Temp,RSR3_Temp
    RETI

WDT_INT:
    RETI

AD_INT:
    RETI

TIMER1_INT:
    PUSH A_Temp,STATUS_Temp,RSR_Temp,@4,A2_Temp,STATUS2_Temp,RSR2_Temp
    BANK                0
    BC                  ISR,T1IF                ;clear Timer1 interrupt flag
    ;MOV                A,@0X40                 ; (test)P86 exchange when intrrupt
    ;XOR                PORT8,A
    BANK                2
    MOV                 A,@255                  ; N=256, Auto reload
    MOV                 T1PD,A
    INC                 SleepCNT                ; 2s    at a time
    MOV                 A,SleepCNT
    SUB                 A,@SetSleepTime         ;
    JBC                 STATUS,C
    JMP                 TIMER1_INT_END
```

```
        BS                  IntoSleepFlag/16,IntoSleepFlag%16
        BANK                0
        MOV                 A,@0X40                 ; (test)P86 exchange when intrrupt
        XOR                 PORT8,A
        CLR                 SleepCNT
  TIMER1_INT_END:
        NOP
        POP A_Temp,STATUS_Temp,RSR_Temp,@4,A2_Temp,STATUS2_Temp,RSR2_Temp
        RETI

TIMER2_INT:
        RETI
UART_INT:
        RETI
SPI_INT:
        RETI
LVD_INT:
        RETI
;=======================================================================
;============================ Begin Program ============================
INITIAL:
        DISI
        NOP
        WDTC
        ClrCommRamBank
        NOP
        ClrRamBank
        NOP
        CALL                IO_INITIAL
        NOP
        BANK                0
        BC                  AT93C46_CS/16,AT93C46_CS%16       ; Disable 93C46
        BS                  SPI_SS/16,SPI_SS%16               ; Disable EM198810
        LCALL               EM198850_RESET
        NOP
        NOP

;=========================== Read/Write EEPROM(TEST) ==========================
/*
        mEWEN                                       ; erasure all ROM
        mERAL                                       ; Clear all
        mEWDS
*/

/*
        LCALL               IO_93C46_INITIAL        ; Set I/O
        BANK                0
        MOV                 A,@0XD2
        MOV                 RX_IDH_Buffer,A
        MOV                 A,@0X13
        MOV                 RX_IDL_Buffer,A
        MOV                 A,@0X11
        MOV                 TX_ID_Buffer,A

        ; write data to DataAddressInEEPROM
        NOP
        NOP
        mEWEN               ;Write enable
        NOP
        MOV                 A,@0X00
        MOV                 DataAddressInEEPROM,A
        MOV                 A,@0X60
        MOV                 DataAddressInMCU,A
        mWRITE              DataAddressInEEPROM,@0,DataAddressInMCU,@14
        NOP
        mEWDS                                   ; disable
        LCALL           IO_93C46_QUIT           ; Set I/O
        NOP
*/

;==============================================================================

        LCALL           IO_93C46_INITIAL        ; Set I/O
        BANK            0
        BC              AT93C46_CS/16,AT93C46_CS%16     ; Disable 93C46
        BS              SPI_SS/16,SPI_SS%16             ; Disable EM198810
        MOV             A,@0X00
        MOV             DataAddressInEEPROM,A
        MOV             A,@0X60
        MOV             DataAddressInMCU,A
        mREAD           DataAddressInEEPROM,@0,DataAddressInMCU,@16
        mEWDS
        LCALL           IO_93C46_QUIT           ; Set I/O
        BC              AT93C46_CS/16,AT93C46_CS%16     ; Disable 93C46
        BS              SPI_SS/16,SPI_SS%16             ; Disable EM198810


        BANK            0
        MOV             A,RX_IDH_Buffer         ; Read ID
        BANK            1
        MOV             RX_IDH,A
        BANK            0
        MOV             A,RX_IDL_Buffer
        BANK            1
        MOV             RX_IDL,A
        BANK            0
        MOV             A,TX_ID_Buffer
        BANK            1
        MOV             TX_ID,A
        BANK            0
        BC              AT93C46_CS/16,AT93C46_CS%16             ; Disable 93C46
        BS              SPI_SS/16,SPI_SS%16                     ; Disable EM198810


        BANK            1
        MOV             A,@0XFF                 ; judge RX_ID,TX_ID
        XOR             A,RX_IDH
        JBC             STATUS,Z
        JMP             Used_Default_Sync
        MOV             A,@0XFF
        XOR             A,RX_IDL
        JBC             STATUS,Z
        JMP             Used_Default_Sync
        JMP             Start_Up
Used_Default_Sync:
        MOV             A,@0X80                 ; SYNC ,used default 0X0DB3
        MOV             RX_IDH,A
        MOV             A,@0X16
        MOV             RX_IDL,A
Start_Up:
        LCALL           CHANGE_TO_INQ_VALUE
        CLR             CH_NO
        LCALL           RF_FREQ_SET

        ENI
        BANK            0
        CLR             ComuClock
        CLR             ComuCycleNum
        CLR             ComuEndNum
        BS              LED1_STATUS/16,LED1_STATUS%16                   ; PORT81,LED
        BC              AT93C46_CS/16,AT93C46_CS%16             ; Disable 93C46
        BS              SPI_SS/16,SPI_SS%16                     ; Disable EM198810

        CLR             SleepCNT
        CLR             CommuStatusFlag
        CLR             GeneralStatusFlag1
        CLR             GeneralStatusFlag2
        CLR             CHN_FLAG

        BANK            2
        MOV             A,@0XFF
        MOV             KeystokeFlag_Befor,A
        MOV             KeystokeTimeCNT,A
        CALL            SearchLinkMode_Set
        NOP


;==============================================================================
MAIN:

        LCALL           Search_Equipment
        NOP
        BANK            0
        BC              LED1_STATUS/16,LED1_STATUS%16
        NOP
        LCALL           Normal_Communicate
        NOP


        ;LCALL          Key_Scan
        ;LCALL          ConnectKey_Scan

/*
        LCALL           Enter_StandbyII_Mode
        NOP
        LCALL           ENTER_TX_BUFFER_NACK
        NOP
        LCALL           RESET_RF_FIFO
        NOP
        LCALL           WRITE_FIFO_RAM
        NOP
*/

        NOP
        JMP             MAIN


;*****************************************************************
;Function:      IO INITIAL SET
;Input:         None
;Output:        None
;*****************************************************************
IO_INITIAL:
;=========================== TCC config ===================================
        BANK            0
        MOV             A,@0x07         ; N=124,P=256,f=4MHz ==> T=8ms
        MOV             TWTCR,A
        MOV             A,@(255-124)    ; load initial value
        MOV             TCC,A
        BS              IMR,TCIE        ;set    TCIE=1  tcc enable interrupt
        CLR             ISR             ;clear interrupt flag
        MOV             A,@0x01
        BANK            1
        CLR             EISR            ;Clear External Interrupt Status Register
        CLR             EIMR            ;Clear External Interrupt Mask  Register
```

```
;============================= Timer1 config ===================================
        BANK            0
        BS             IMR,T1IE          ; Enable Timimer1
        BC             ISR,T1IF          ; Clear timer1 intrrupt flag
        BANK            2
        MOV             A,@0B00000111     ; N=256, P=256, T=2s
        MOV             T1CR,A
        MOV             A,@255            ; N=256, Auto reload
        MOV             T1PD,A
        MOV             A,@0B00011100     ; 8 bit counter ,f=32768
        MOV             TSR,A

;==============================PORT A/B======================================
        BANK            4                       ; POWER ON DISABLE PA PB PULL HIGH
        MOV             A,@00010000B      ; Set PA4 (MISO) as input pin
        MOV             PAIOCR,A          ; SO,SCK output pin
        MOV             A,@00010000B      ; PA4 PULL HIGH
        MOV             PAPHCR,A
        MOV             A,@00110111B      ; PB0 PB1 PB2 PB4 PB5 PB1AS INPUT
        MOV             PBIOCR,A          ; PKT FIFO OUTPUT PIN
        BANK            5
        MOV             A,@00110111B
        MOV             PBPHCR,A          ;PB0 PB2 ENABLE PULL HIGH
        BANK            0
        CLR            PORTA              ; PortA output logic "0"
        ;CLR            PORTB             ; PortB output logic "0"
        BANK            2
        MOV             A,@00000000B      ; Shift left, SDO delay time: 16clk,
        MOV             SPIS, A           ; disable open-drain
        MOV             A,@11001000B      ; Data shift out rising edge, is on hold during low
        MOV             SPIC,A            ; Enable SPI mode, after data output sdo remain low

;=====================PORT 7/8/9 INITIAL=================================
        BANK            4
        MOV             A,@11111111B
        MOV             IOC9,A            ;Set Input
        MOV             A,@00000000B
        MOV             IOC7,A            ;Set Output
        MOV             A,@00000000B
        MOV             IOC8,A            ;Set Output
        BANK            5
        MOV             A,@11110000B
        MOV             P9PHCR,A          ;SET PULL UP
        BANK            6
        CLR            P7ODCR
        CLR            P8ODCR
        CLR            P9ODCR
        BANK            1
        MOV             A,@00000000B      ;Disable LCD common dirver pin
        AND            LCDSCR0, A
        AND            LCDSCR1, A
        AND            LCDSCR2, A
        BANK            0
        CLR            PORT7              ; Port7 output logic "0"
        CLR            PORT8              ; Port8 output logic "0"
        ;CLR            PORT9             ; Port9 output logic "0"
        BC             AT93C46_CS/16,AT93C46_CS%16      ; Disable 93C46
        BS             SPI_SS/16,SPI_SS%16              ; disable EM198810
        RET

;*****************************************************************
;Function:       MCU SLEEP SET
;Input:          None
;Output:         None
```

```
;*****************************************************************
MCU_EnterSleep_Set:
        BANK            0
        CLR             IMR             ;Disable Timier1/TCC
        CLR             ISR
        BANK            3
        CLR             EIESH           ;SET Falling Edge
        BANK            1
        MOV             A,@0B01100000   ;Enable EX6(PB2)/EX5(PB1) Interrupt
        MOV             EIMR,A
        CLR             EISR
        BANK            4
        MOV             A,@0B01100000
        MOV             WKCR,A          ;enable EX6(PB2)/EX5(PB1) wake up
        BC              SCCR,IDLE       ;Set IDLE=0(SLEEP MODE)
        BANK            0
        BC              AT93C46_CS/16,AT93C46_CS%16     ; Disable 93C46
        BS              SPI_SS/16,SPI_SS%16             ; disable EM198810
        NOP
        RET

;*****************************************************************
;Function:      MCU wakeup SET
;Input:         None
;Output:        None
;*****************************************************************
MCU_WakeUp_Set:
        BANK            3
        CLR             EIESH           ; SET Falling Edge
        BANK            1
        MOV             A,@0B00000000   ; Disable EX6(PB2)/EX5(PB1) Interrupt
        MOV             EIMR,A
        CLR             EISR
        BANK            4
        MOV             A,@0B00000000
        MOV             WKCR,A          ; Disable EX6(PB2)/EX5(PB1) wake up
        NOP
        BANK            0
        MOV             A,@0B10000001
        MOV             IMR,A           ; Enable Timier1/TCC
        CLR             ISR
        BC              AT93C46_CS/16,AT93C46_CS%16     ; Disable 93C46
        BS              SPI_SS/16,SPI_SS%16             ; disable EM198810
        RET

;*****************************************************************
;Function:      mode select
;Input:
;Output:        None
;desciption:    set timing and select mode
;*****************************************************************
SearchLinkMode_Set:             ;0X11
        BS              SearchStatusFlag/16,SearchStatusFlag%16         ;set search mode
        BC              NormalStatusFlag/16,NormalStatusFlag%16         ;Clear normal mode
        BC              LoseFrameStatusFlag/16,LoseFrameStatusFlag%16   ;Clear LoseFreq mode
        BC              EEpromWRStatusFlag/16,EEpromWRStatusFlag%16
        BS              LinkModeFlag/16,LinkModeFlag%16
        BC              ForceLinkModeFlag/16,ForceLinkModeFlag%16
        BC              FccTestModeFlag/16,FccTestModeFlag%16
        RET

SearchForceLinkMode_Set:        ;0X25
        BS              SearchStatusFlag/16,SearchStatusFlag%16         ;set search mode
        BC              NormalStatusFlag/16,NormalStatusFlag%16         ;Clear normal mode
        BS              LoseFrameStatusFlag/16,LoseFrameStatusFlag%16   ;Clear LoseFreq mode
        BC              EEpromWRStatusFlag/16,EEpromWRStatusFlag%16
        BC              LinkModeFlag/16,LinkModeFlag%16
        BS              ForceLinkModeFlag/16,ForceLinkModeFlag%16
        BC              FccTestModeFlag/16,FccTestModeFlag%16
        RET

NormalLinkMode_Set:     ;0X12
        BC              SearchStatusFlag/16,SearchStatusFlag%16            ;Clear search status
        BS              NormalStatusFlag/16,NormalStatusFlag%16             ;set normal mode
        BC              LoseFrameStatusFlag/16,LoseFrameStatusFlag%16       ;Clear LoseFreq mode
        BC              EEpromWRStatusFlag/16,EEpromWRStatusFlag%16
        BS              LinkModeFlag/16,LinkModeFlag%16
        BC              ForceLinkModeFlag/16,ForceLinkModeFlag%16
        BC              FccTestModeFlag/16,FccTestModeFlag%16
        RET

LoseFrameLinkMode_Set:  ;0X16
        BC              SearchStatusFlag/16,SearchStatusFlag%16            ;Clear search status
        BS              NormalStatusFlag/16,NormalStatusFlag%16             ;set normal mode
        BS              LoseFrameStatusFlag/16,LoseFrameStatusFlag%16       ;Clear LoseFreq mode
        BC              EEpromWRStatusFlag/16,EEpromWRStatusFlag%16
        BS              LinkModeFlag/16,LinkModeFlag%16
        BC              ForceLinkModeFlag/16,ForceLinkModeFlag%16
        BC              FccTestModeFlag/16,FccTestModeFlag%16
        RET

LoseFrameForceLinkMode_Set: ;0X25
        BS              SearchStatusFlag/16,SearchStatusFlag%16            ;Clear search status
        BC              NormalStatusFlag/16,NormalStatusFlag%16             ;set normal mode
        BS              LoseFrameStatusFlag/16,LoseFrameStatusFlag%16       ;Clear LoseFreq mode
        BC              EEpromWRStatusFlag/16,EEpromWRStatusFlag%16
        BC              LinkModeFlag/16,LinkModeFlag%16
        BS              ForceLinkModeFlag/16,ForceLinkModeFlag%16
        BC              FccTestModeFlag/16,FccTestModeFlag%16
        RET

FccSearchLinkMode_Set:  ;0X41
        BS              SearchStatusFlag/16,SearchStatusFlag%16            ;set search mode
        BC              NormalStatusFlag/16,NormalStatusFlag%16             ;Clear normal mode
        BC              LoseFrameStatusFlag/16,LoseFrameStatusFlag%16       ;Clear LoseFreq mode
        BC              EEpromWRStatusFlag/16,EEpromWRStatusFlag%16
        BC              LinkModeFlag/16,LinkModeFlag%16
        BC              ForceLinkModeFlag/16,ForceLinkModeFlag%16
        BS              FccTestModeFlag/16,FccTestModeFlag%16
        RET
```