**注：以下标红的为具体修改的位置**

**48页最上方程序：**

**注释需要修改如下：**

…

MOV AL，BUF ；将01H送AL

MOV AH，BUF + 2 ；将03H送AH

…

**54页例3-17：**

程序有个别错误，修改如下：

MASC MACRO X，Y

MOV AL，X

ADD AL，30H

MOV Y，AL

ENDM

**70页（2）查询方式：**

程序有个别错误，修改如下：

START： MOV AX, 1000H

MOV DS, AX

MOV DI, 0000H

MOV CX， 8192

MOV BL， FFH

NEXT ： MOV DX， 0200H ；DX=READY/状态接口地址

WAIT ： IN AL， DX ；从接口读98C64A的状态

TEST AL， 01H ；98C64A是否忙？

JZ WAIT ；低电平表示98C64A处于忙，等待

MOV [DI], BL ；否则，98C64A处于闲，写入一个字节

INC DI

LOOP NEXT

HLT

**93页ICW4格式：**

D3和D2位，在我们课件上和某些教材上的定义是：10时，从片；11时，主片。但吴宁第四版是反过来，这个有待商榷。

**95页例5-3：**

程序中注释有错误，具体修改如下：

；初始化程序

MOV AL, 0001 1011B ；ICW1，电平触发，需要ICW4

OUT 20H, AL

MOV AL, 0000 1000B ；ICW2，类型号高5位为00001

OUT 21H, AL

MOV AL, 0000 0011B ；ICW4，一般全嵌套，自动中断结束

；，非缓冲

OUT 21H, AL

……

；读取ISR寄存器

MOV AL, 0BH ；写入OCW3，查询ISR

OUT 20H, AL

CALL DELAY

IN AL, 20H ；读取ISR的内容

**96页例5-4：**

程序中有个别错误，具体修改如下：

；主片初始化

MOV AL，0001 1001B；ICW1，电平触发，级联方式

OUT 20H，AL

MOV AL，0000 1000B；ICW2，类型号高5位为00001

OUT 21H，AL

MOV AL，0000 0100B；ICW3，从片级联到主片IRQ2

OUT 21H，AL

MOV AL，0000 0001B；ICW4，普通嵌套，非自动EOI，非缓冲

OUT 21H，AL

；从片初始化

MOV AL，0001 1001B；ICW1，电平触发，级联方式

OUT 0A0H，AL

MOV AL，0111 0000B；ICW2，类型号高5位为0111 0

OUT 0A1H，AL

MOV AL，0000 0010B；ICW3，级联到主片IRQ2

OUT 0A1H，AL

MOV AL，0000 0001B；ICW4，普通嵌套，非自动EOI，非缓冲

OUT 0A1H，AL

**106-107页例6-1：**

1. 40H-43H是8位端口，三个计数器初始化程序中的OUT指令均应改为“OUT 端口号，AL”的形式，例题题目中的0040H~0043H前面两个00去掉；

2. 计数器0控制字应为16H，计数器2的控制字应为10010100B（94H）；

具体代码修改如下：

；计数器0

MOV AL, 16H ；计数器0控制字，00010110B

OUT 43H, AL ；写入控制字端口

MOV AL, 100 ；2MHz/20kHz=100

OUT 40H, AL ；计数值写入计数器0端口

；计数器1

MOV AL, 74H ；计数器1控制字，01110100B

OUT 43H, AL ；写入控制字端口

MOV AX, 10000 ；5ms/0.5μs=10 000

OUT 41H, AL ；计数值先低后高写入计数器1端口

MOV AL, AH

OUT 41H, AL

；计数器2

MOV AL, 94H ；计数器0控制字，10010100B

OUT 43H, AL ；写入控制字端口

MOV AL, 200 ；0.1ms/0.5μs=200

OUT 42H, AL ；计数值写入计数器2端口

**107页例6-1的图6-11：**

图6-11中IOR和IOW的位置要交换。

125页2.程序存储器的特殊保留单元

（1）和（2）两点的内容交换，修改如下：

（1）0003H：外部中断0的中断服务程序入口地址；

（2）000BH：定时/计数器0溢出中断服务程序入口地址；

**137页2.(2)写片外RAM及I/O接口：**

注释要修改，修改如下：

MOVX @DPTR, A ；(DPTR) ← (A)

MOVX @Ri, A ；(Ri) ← (A)

**138页4.堆栈指令PUSH：**

注释中的（（SP））改为（SP），（direct）改为direct，修改如下：

PUSH direct ；SP←（SP）+1，（SP）←（direct）

POP direct ；direct←（（SP）），SP←（SP）-1

**141页1.加法指令1）不带进位的加法：**

注释要修改，具体如下：

ADD A，Rn ；A←（A）+（Rn）

ADD A，direct ；A←（A）+（direct）

ADD A，@Ri ；A←（A）+（（Ri））

ADD A，#data ；A←（A）+ data

**141页1.加法指令2）带进位的加法：**

注释要修改，具体如下：

ADDC A，Rn ；A←（A）+（Rn）+（CY）

ADDC A，direct ；A←（A）+（direct）+（CY）

ADDC A，@Ri ；A←（A）+（（Ri））+（CY）

ADDC A，#data ；A←（A）+ data+（CY）

**141页1.加法指令3）加1指令：**

注释要修改，具体如下：

INC A ；A←（A）+1

INC Rn ；Rn←（Rn）+1

INC direct ；direct←（direct）+1

INC @Ri ；(Ri)←（（Ri））+1

INC DPTR ；DPTR←（DPTR）+1

**141页1.加法指令4）十进制调整：**

注释要修改，具体如下：

DA A ；若（AC）=1 或 (ACC.3~ACC.0)>9，则A←（A）+ 06H

；若（CY）=1 或 (ACC.7~ACC.4)>9，则A←（A）+ 60H

**142页2.减法指令1）带借位的减法：**

注释要修改，具体如下：

SUBB A，Rn ；A←（A）-（Rn）-（CY）

SUBB A，direct ；A←（A）-（direct）-（CY）

SUBB A，@Ri ；A←（A）-（（Ri））-（CY）

SUBB A，#data ；A←（A）- data-（CY）

**142页2.减法指令2）减1指令：**

注释要修改，具体如下：

DEC A ；A←（A）-1

DEC Rn ；Rn←（Rn）-1

DEC direct ；direct←（direct）-1

DEC @Ri ；(Ri)←（（Ri））-1

**144页8.3.3逻辑运算与移位指令（1）逻辑与指令：**

注释要修改，如下：

ANL A，Rn ；A←（A）∧（Rn）

ANL A，direct ；A←（A）∧（direct）

ANL A，@Ri ；A←（A）∧（（Ri））

ANL A，#data ；A←（A）∧ data

ANL direct，A ；direct←（direct）∧（A）

ANL direct，#data ；direct←（direct）∧ data

**145页（2）逻辑与指令：**

注释要修改，如下：

ORL A，Rn ；A←（A）∨（Rn）

ORL A，direct ；A←（A）∨（direct）

ORL A，@Ri ；A←（A）∨（（Ri））

ORL A，#data ；A←（A）∨ data

ORL direct，A ；direct←（direct）∨（A）

ORL direct，#data ；direct←（direct）∨ data

**145页（3）逻辑异或指令：**

注释要修改，如下：

XRL A，Rn ；A←（A）⊕（Rn）

XRL A，direct ；A←（A）⊕（direct）

XRL A，@Ri ；A←（A）⊕（（Ri））

XRL A，#data ；A←（A）⊕ data

XRL direct，A ；direct←（direct）⊕（A）

XRL direct，#data ；direct←（direct）⊕ data

**145页（4）累加器清0和取反指令：**

注释要修改，如下：

CLR A ；A←0

CPL A ；A←

**145页（5）移位指令：**

注释要修改，如下：

RL A ；（ACC.7~ACC.1）←（ACC.6~ACC.0），（ACC.0）←（ACC.7）

RR A ；（ACC.7~ACC.1）→（ACC.6~ACC.0），（ACC.0）→（ACC.7）

RLC A ；（CY）←（ACC.7），（ACC.7~ACC.1）←（ACC.6~ACC.0），（ACC.0）←（CY）

RRC A ；（CY）→（ACC.7），（ACC.7~ACC.1）→（ACC.6~ACC.0），（ACC.0）→（CY）

**147页1.无条件转移 （1）短跳转**

注释要修改，如下：

AJMP addr11 ；PC←（PC）+ 2，PC10~0← addr11

**148页（2）长跳转**

注释要修改，如下：

LJMP addr16 ；PC← addr16

**148页（3）相对转移**

注释要修改，如下：

SJMP rel ；PC←（PC）+ 2，PC←（PC）+ rel

**148页（4）散转移**

注释要修改，如下：

JMP @A+DPTR ；PC←（PC）+1，PC←（A）+（DPTR）

**149页（1）累加器判0转移**

注释要修改，如下：

JZ rel ；若（A）= 0，则PC←（PC）+ 2 + rel

；若（A）≠ 0，则PC←（PC）+ 2

JNZ rel ；若（A）≠ 0，则PC←（PC）+ 2 + rel

；若（A）= 0，则PC←（PC）+ 2

**149页（2）比较不等转移**

注释要修改，如下：

CJNE A，direct，rel ；若(A)≠(direct)，则PC←(PC)+3+ rel

；若(A)= (direct)，则PC←(PC)+3

CJNE A，#data，rel ；若(A)≠ data，则PC←(PC)+ 3+ rel

；若(A)= data，则PC←(PC)+ 3

CJNE Rn，#data，rel ；若(Rn)≠ data，则PC←(PC)+ 3+ rel

；若(Rn)= data，则PC←(PC)+ 3

CJNE @Ri，#data，rel ；若((Ri))≠ data，则PC←(PC)+ 3+ rel

；若((Ri))= data，则PC←(PC)+ 3

**150页（2）减1不为0转移**

注释要修改，如下：

DJNZ Rn，rel ；PC← (PC)+ 2，Rn← (Rn)－1

；若（Rn）≠ 0，则PC← (PC)+ rel

；若（Rn）= 0，则结束循环，程序向下顺序执行

DJNZ direct，rel ；PC← (PC)+ 3，direct← (direct)－1

；若(direct)≠ 0，则PC← (PC)+ rel

；若(direct)= 0，则结束循环，程序向下顺序执行

**150页3.（1）子程序调用**

注释要修改，如下：

ACALL addr11 ；PC← (PC)+ 2，SP←(SP)+ 1，(SP)← (PC7~0)

；SP←(SP)+1，(SP)←(PC15~8），PC10~0← addr11

LCALL addr16 ；PC← (PC)+ 3，SP←(SP)+ 1，(SP)←(PC7~0)

；SP←(SP)+ 1，(SP)←(PC15~8)，PC← addr16

**150页3.（2）返回指令**

注释要修改，如下：

RET ；PC15~8←(SP)，SP←(SP)－1，PC7~0←(SP)，SP←(SP)－1

RETI ；PC15~8←(SP)，SP←(SP)－1，PC7~0←(SP)，SP←(SP)－1

**150页4.空操作指令**

注释要修改，如下：

NOP ；PC← (PC)+ 1

**152页8.3.5（1）位传送指令**

注释要修改，如下：

MOV bit，C ；bit←（CY）

MOV C，bit ；CY←（bit）

**152页8.3.5（2）位置位/复位指令**

注释要修改，如下：

SETB C ；CY← 1

SETB bit ；bit← 1

CLR C ；CY← 0

CLR bit ；bit← 0

**152页8.3.5（3）位逻辑运算指令**

注释要修改，如下：

ANL C，bit ；CY←（CY）∧（bit）

ANL C，/bit ；CY←（CY）∧

ORL C，bit ；CY←（CY）∨（bit）

ORL C，/bit ；CY←（CY）∨

CPL C ；CY←

CPL bit ；bit←

**153页例8-25**

注释要修改，如下：

MOV C，35H ；（CY）←（35H）

ANL C，/36H ；（CY）←（35H）∧

MOV 37H，C ；暂存运算结果

MOV C，/35H ；（CY）←

ANL C，36H ；（CY）← ∧（36H）

ORL C，37H ；（CY）←（35H）∧ + ∧（36H）

MOV 37H，C ；存放运算结果

**153页8.3.5（4）位控制指令**

注释要修改，如下：

JC rel ；（CY）=1，则PC←（PC）+ 2+ rel

；（CY）=0，则PC←（PC）+ 2

JNC rel ；（CY）=0，则PC←（PC）+ 2+ rel

；（CY）=1，则PC←（PC）+ 2

JB bit，rel ；（bit）=1，则PC←（PC）+ 3+ rel

；（bit）=0，则PC←（PC）+ 3

JNB bit，rel ；（bit）=0，则PC←（PC）+ 3+ rel

；（bit）=1，则PC←（PC）+ 3

JBC bit，rel ；（bit）=1，则PC←（PC）+ 3+ rel，并使bit←0

；（bit）=0，则PC←（PC）+ 3

**186页11.1.5节**

第一段中的此句修改如下：

……外中断1的中断服务程序入口地址是0013H……

下面程序也要相应修改，如下：

ORG 0000H ；复位入口

START: LJMP MAIN ；跳转到主程序

ORG 0013H ；外中断1中断服务程序入口

LJMP INT1

ORG 0030H ；主程序

MAIN：SETB IT0 ；设置外中断1为下降沿触发

SETB EA ；开放中断系统

SETB EX1 ；允许外中断1

SETB PX1 ；设置外中断1高优先级，若不设置，默认为低优先级

MOV SP，#03FH ；设置堆栈

...

INT1: PUSH PSW ；保护（软件）现场

PUSH ACC

...

POP ACC ；恢复现场

POP PSW

RETI ；中断返回

**188页11.2.2节第2点TMOD**

第一句话修改如下**：**

TMOD的地址为89H，不能进行位寻址，其高4位用于设置定时/计数器T1，低4位用于设置定时/计数器T0，具体定义如下：