

Important changes to Embedded Microcomputer Systems by Jonathan W. Valvano

Page xi, Change **dem-onstration** to **demonstration**

Page 3, Change **ordinance** to **ordnance**

Page 56, Figure 1.58, Delete extra **cc** near Q1

Page 126, Section 2.11.3.6. Change

In this situation, if we suspect the error occurs when the pointer nears the end of the buffer, we could add a filter that saves in the array only when the pointer is above a certain value.

to

In this situation, if we suspect the error occurs when certain conditions are true, we could add a filter that saves in the array only when those conditions are met.

Page 180, Problem 3.1, Change **ldab #V** to **ldab #'V**

Page 195, Figure 4.8, Move "software writes new data" over on top of "asks device to output it"

Page 208, program 4.16 change

```
tpa      restore CCR to previous value
tba
```

to

```
tap      restore CCR to previous value
tba
```

Page 209, program 4.17 change

```
cpx PutPt Empty if initially the same
```

to

```
cpy PutPt Empty if initially the same
```

Page 209, program 4.17 change

```
tpa      restore CCR to previous value
tba
```

to

```
tap      restore CCR to previous value
tba
```

Page 211, program 4.20 change

```
tpa      restore CCR to previous value
tba
```

to

```
tap      restore CCR to previous value
tba
```

Page 212, program 4.21 change

```
tpa      restore CCR to previous value
tba
```

to

```
tap      restore CCR to previous value
tba
```

Page 214, program 4.24 change

```
tpa      restore CCR to previous value
tba
```

to

```
tap      restore CCR to previous value
tba
```

Page 215, program 4.25 change

```
tpa      restore CCR to previous value
tba
```

to

```
tap      restore CCR to previous value
tba
```

Page 215, program 4.25 change

```
incb
```

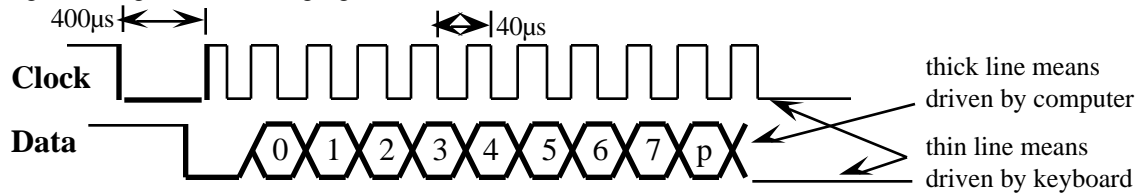
to

```
dec Size one less element in FIFO
```

incb

Page 258, Program 4.48, Delete these words from the comment "example 4.14.2"

Page 264, Figure 4.46, wrong figure. Should be



Page 279, change the sequence of Program 5.13 so it reads as follows

```
* To block a thread on semaphore S, execute SWI
SWIhan ldx RunPt    running process "to be blocked"
      sts SP,x     save Stack Pointer in its TCB
* Unlink "to be blocked" thread from RunPt list
      ldy Next,x   find previous thread
      sty RunPt   next one to run
look   cpx Next,y  search to find previous
      beq found
      ldy Next,y
      bra look
found  ldd RunPt   one after blocked
      std Next,y  link previous to next to run
* Put "to be blocked" thread on block list
      ldy BlockPt
      sty Next,x  link "to be blocked"
      stx BlockPt
* Launch next thread
      ldx RunPt
      lds SP,x    set SP for this new thread
      ldd TCNT    Next thread gets a full 10ms time slice
      addd #20000 interrupt after 10 ms
      std TOC5
      ldaa #$08   ($20 on the 6812)
      staa TFLG1  clear OC5F
      rti
```

Page 307, Program 6.10, 6811 C code, change ")" to "}"

```
First = TIC1; Count=0; Mode=1;
if(((TIC1&0x8000)==0)
    &&(TFLG2&0x80)) Count--;)
to
First = TIC1; Count=0; Mode=1;
if(((TIC1&0x8000)==0)
    &&(TFLG2&0x80)) Count--;}

```

Page 327, Program 6.24 6812 version, Change **TFLG1|=0x08;** to **TFLG1=0x08;**

Page 362, Section 7.3.1. change

More details about common mode will be presented later in Chapters 11 and 12.

to

More details about common mode will be presented later in Sections 11.2.3 and 11.2.6.

Page 362, Change **•100** to **100**

Page 363, Change

To transmit the computer ~~enables the driver by making DE active, then~~ sends the serial frame from the TxD output of the SCI port. ~~If RE is also active during transmission, the transmitted frame is echoed into the serial receiver of the SCI RxD line. To receive a frame the computer simply enables its receiver (by making RE active) and~~ accepts a serial frame on the RxD line in the usual manner.

to

Normally, we make both DE and RE active on all devices. To transmit the computer sends the serial frame from the TxD output of the SCI port. The transmitted frame is echoed into the serial receiver of the SCI RxD line. To receive a frame the computer accepts a serial frame on the RxD line in the usual manner.

Page 380 unnumbered figure, switch **PE** to **ILT** (PE is bit 1, ILT is bit 2)

Page 382, Change **SCSR** to **SCxSR1**

Page 382, Change **SCDR** to **SCxDRL**

Page 382, Change **SCxDR** to **SCxDRL** (three places)

Page 383, Add to paragraph discussing PF

Cleared by reading SC1SR1 with PF set, then reading SCxDRL.

Page 384 third line from the top, Change **Section 2.4.2** to **Section 2.7.2**

Page 399, Change **Cleared by SPSR** to **Cleared by reading SPSR**

Page 409, Program 7.22, change the 6805 and 6808 code from

```
data+SPDR; // 1sbyte of A/D
```

to

```
data+=SPDR; // 1sbyte of ADC
```

Page 435, Program 8.10, MC68HC11A8 version, Change

```
TFLG1=0.08; } // ack 0C5F
```

to

```
TFLG1=0x08; } // ack 0C5F
```

Page 435, Program 8.10, MC68HC812A4 version, Change

```
TC5=TC5+period; // ack 0C5F
```

to

```
TC5=TC5+period; TFLG1=0x20; }
```

Page 436, Figure 8.26, Delete **68HC705 PortA 68HC708 PortA 68HC11 PortC**

Page 437, Program 8.11, MC68HC812A4 version, Change

```
Ritual: clr DDRJ ;PJ3-PJ0 inputs
        rts ;PJ7-PJ0 oc outputs
```

to

```
Ritual: clr DDRJ ;PJ3-PJ0 inputs
        movb #$0F,PUPSJ
        movb #$0F,PULEJ
        rts ;PJ7-PJ0 oc outputs
```

Page 439, Program 8.12, MC68HC812A4 version, Change

```
DDRJ=0x00; } // PJ7-PJ4 are oc outputs
```

to

```
DDRJ=0x00; PUPSJ=PULEJ=0x0F; }
```

Page 446, Change "the resistor is calculated as (Figure 8.35)" to

"the resistor is calculated as (Figure 8.33)"

Page 455, Change **NOR** to **EOR** two places

Page 476 third line from the bottom, Change **peripheral** to **peripherals**

Page 485 Table 8.14, Change $(*/s^2)$ to $(*/s^2)$

Page 485 Table 8.14, Change $(*/s^3)$ to $(*/s^3)$

Page 485 Table 8.15, Change $(*/s^{**2})$ to $(*/s^2)$

Page 485 Table 8.15, Change $(*/s^{**3})$ to $(*/s^3)$

Page 506, Table 9.5, change

```
PF4 CSD 0xxxxxxxxxxxxxxxxx $0000 $7FFF 32K (CSDFH=0)
```

to

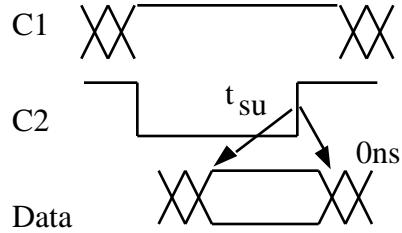
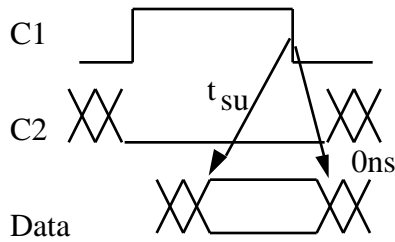
```
PF4 CSD 0xxxxxxxxxxxxxxxxx $0000 $7FFF 32K (CSDFH=1)
```

Page 521, Change **SMOD** and **MDA** bits in the **HPRIO** to **SMODN**, **MODB**, **MODA** bits in the **MODE**

Page 525, Change **SMOD** and **MDA** bits in the **HPRIO** to **SMODN**, **MODB**, **MODA** bits in the **MODE**

Page 551, Figure 9.64, Delay from rise of E to fall of DBE should be 37 rather than 49 ns

Page 569, Figure 9.82, wrong figure. Should be



Page 569, Change

"The write timing when controlled by **C1** is shown on the left in Figure 9.82; the write timing when controlled by **C2** is shown on the right in Figure 9.82."

to

"The write function occurs on either the fall of **C1** or the rise of **C2**, whichever occurs first. Let the setup time be t_{su} and assume the hold time is zero."

Page 620, Figure 11.49. change 10k /500k to 10k /510k

Page 620, Figure 11.50. change 10k /500k to 10k /510k

Page 621, Figure 11.51. change R_1/R_2 to $R_1/(R_1+R_2)$

Page 709, Section 12.4.2

change

uncertainty of counting events is n

to

uncertainty of counting events is n

Page 759, Program 13.19

change

dc. b 128, 244, 5, 8 ; EPS

to

dc. b 128, 224, 5, 8 ; EPS

Page 759, Program 13.20

change

a_tab: dc. b 244, 255, 16, 0 ; DPL

to

a_tab: dc. b 224, 255, 16, 0 ; DPL

Page 809, Program 15.2 (two places)

change

for(i=5; i>0; i++)

to

for(i=5; i>0; i--)

Page 818 Equation 51, change -13 to +13, yielding

$$y_2(n) = x(n) + x(n-2) + \frac{-14x(n-1) + 13y_2(n-1) - 231y_2(n-2)}{256} \quad (51)$$

Page 828, Figure 15.17, change **k=16** to **k=32**

Back cover, change **74505** to **74S05**