Solved Exercises 1

1. Solve Problem **10.7** from the textbook. NOTE: Connect Ports **A** and **B** to the four 7-segment decoders, letting $\mathbf{PA_{7-4}}$, $\mathbf{PA_{3-0}}$, $\mathbf{PB_{7-4}}$, and $\mathbf{PB_{3-0}}$ display the first, second, third, and fourth received digits, respectively. Assume that all four digits arrive immediately after the character **H** has been received.

```
#define   RBUF    (volatile unsigned char *) 0xFFFFFFE0
#define   SSTAT   (volatile unsigned char *) 0xFFFFFFE2
#define   PAOUT   (volatile unsigned char *) 0xFFFFFFF1
#define   PADIR   (volatile unsigned char *) 0xFFFFFFF2
#define   PBOUT   (volatile unsigned char *) 0xFFFFFFF4
#define   PBDIR   (volatile unsigned char *) 0xFFFFFFF5

char temp;
char digits[4];                          /* Buffer for received digits. */
int i;

void main()
{
    /* Initialize the parallel ports. */
    *PADIR = 0xFF;                       /* Configure Port A as output. */
    *PBDIR = 0xFF;                       /* Configure Port B as output. */

    /* Transfer the characters. */
    while (1)                            /* Infinite loop. */
    {
        while ((*SSTAT & 0x1) == 0);     /* Wait for a new character. */
        if (*RBUF == 'H')
        {
          for (i = 3; i >= 0; i--)
          {
            while ((*SSTAT & 0x1) == 0); /* Wait for the next digit. */
            digits[i] = *RBUF;           /* Save the new digit (ASCII). */
          }
          temp = digits[3] << 4;         /* Shift left first digit by 4 bits, */
          *PAOUT = temp | (digits[2] & 0xF); /*   append second and send to A. */
          temp = digits[1] << 4;         /* Shift left third digit by 4 bits, */
          *PBOUT = temp | (digits[0] & 0xF); /*   append fourth and send to B. */
        }
    }
}
```

2. Solve Problem **10.8** from the textbook. NOTE: Upon detecting the character **H**, the subsequent four digits have to be saved and displayed only when the fourth digit arrives. Interrupts must be used to detect the arrival of both **H** and the four digits (the ISR has to keep track of the received characters).

```
#define   RBUF     (volatile unsigned char *) 0xFFFFFFE0
#define   SCONT    (volatile unsigned char *) 0xFFFFFFE3
#define   PAOUT    (volatile unsigned char *) 0xFFFFFFF1
#define   PADIR    (volatile unsigned char *) 0xFFFFFFF2
#define   PBOUT    (volatile unsigned char *) 0xFFFFFFF4
#define   PBDIR    (volatile unsigned char *) 0xFFFFFFF5
#define   IVECT    (volatile unsigned int *) 0x20

char temp;
char digits[4];                          /* Buffer for received digits. */
int k;

interrupt void intserv();

void main()
{
    /* Initialize the parallel ports. */
    *PADIR = 0xFF;                       /* Configure Port A as output. */
    *PBDIR = 0xFF;                       /* Configure Port B as output. */
    /* Initialize the interrupt mechanism. */
    *IVECT = (unsigned int *) &intserv;  /* Set the interrupt vector. */
    asm ("MoveControl    PSR, #0x40");   /* Respond to IRQ interrupts. */
    *SCONT = 0x10;                       /* Enable receiver interrupts. */

    /* Transfer the characters. */
    k = 0;
    while (1);                           /* Infinite loop */
}

/* Interrupt service routine. */
interrupt void intserv()
{
    if (k > 0)
    {
        k = k - 1;
        digits[k] = *RBUF;               /* Save the new digit (ASCII). */
        if (k == 0)
        {
          temp = digits[3] << 4;         /* Shift left first digit by 4 bits, */
          *PAOUT = temp | (digits[2] & 0xF);  /*   append second and send to A. */
          temp = digits[1] << 4;         /* Shift left third digit by 4 bits */
          *PBOUT = temp | (digits[0] & 0xF);  /*   append fourth and send to B. */
        }
        else if (*RBUF == 'H')
          k = 4;
    }
}
```