(10) Question 1. Write a C macro that will clear one bit in an I/O port.

```
#define BIT_CLEAR(port,n)     ((port) &= ~(1<<(n)))
#define BIT_CLEAR(port,n)     ((port) &= ~bit(n))
```

The following shows how the macro is expanded.

```
  ((PTT) &= ~(1<<(7)));     // clears PTT bit 7
  ((PTH) &= ~(1<<(3)));     // clears PTH bit 3
```

(5) Question 2. E) All of the above are true.

(5) Question 3. Saying when idle the clock is 1 means use CPOL=1. Since the ADC shifts its new data out on the falling edge of the clock, the 9S12 must shift in on the rising edge.

D) CPOL = 1; CPHA = 1

(35) Question 4. There are two positive-logic switches interfaced to PT1 and PT0.

Part a) Show the initialization ritual.

```
void Switch_Init(void){
  asm sei         // make atomic
  DDRP |= 0x03;   // PP1,PP0 are outputs
  PTP &= ~0x03;   // PP1,PP0 are off
  TIOS &= ~0x03;  // PT1,PT0 input captures
  DDRT &= ~0x03;  // PT1,PT0 are inputs
  TSCR1 = 0x80;   // enable TCNT
  TSCR2 = 0x00;   // divide by 1 clock, 42ns resolution is best
  TCTL4 = (TCTL4&0x0F)|0x0A; // falling edges on PT1,PT0
  TFLG1 = 0x03;   // Clear C1F,C0F
  TIE |= 0x03;    // Arm C1F,C0F
  asm cli
}
```

Part b) Show the PT0 input capture0 ISR.

```
void handleBoth(void){ short difference;
  difference = TC0-TC1;   // which was first?
  if(difference<0){ // PT0 first
    PTP |= 1;       // LED0
  }else if(difference == 0){ // same 42ns
    PTP |= 3;       // both LED0 and LED1
  }else{            // PT1 first
    PTP |= 2;       // LED1
  }
}
interrupt 8 void TC0handler(void){
  if((TFLG1&0x03) == 0x03){ // both
    handleBoth();
  } else{
    PTP |= 1;       // LED0
  }
  TIE &= ~0x03;     // only first one matters
}
```

Part b) Simple answer handing cases 0,1,2.

```
interrupt 8 void TC0handler(void){
  PTP |= 1;       // LED0
  TIE &= ~0x03;   // only first one matters
}
```

Part c) Show the PT1 input capture1 ISR.

```
interrupt 9 void TC1handler(void){
  if((TFLG1&0x03) == 0x03){ // both
    handleBoth();
  } else{
    PTP |= 2;       // LED1
  }
  TIE &= ~0x03;     // only first one matters
}
```

Part c) Simple answer handing cases 0,1,2.

```
interrupt 9 void TC1handler(void){
  PTP |= 2;       // LED1
  TIE &= ~0x03;     // only first one matters
}
```

(20) Question 5. A signed fixed-point system has a range of values from -99.99 to +99.99 with a resolution of 10^{-2} cm. Note: 10^{-2} equals 0.01.

Part a) The precision is 19999 alternatives, which will fit in 16 bits.

E) `short`

Part b) Multiply by 0.5 is the same as divide by 2.

```
 Position = Position/2;
```

With rounding:

```
  if(Position<0){
    Position = (Position-1)/2; // -1.5 rounds to -2
  } else{
    Position = (Position+1)/2; // 1.5 rounds to 2
  }
```

Part c) The integer part of 4 cm is 400.

```
 Position = Position+400;
```

With overflow prevention, implementing ceiling:

```
  if(Position<9600){
    Position = Position+400;  // normal addition
  } else{
    Position = 9999;          // ceiling
  }
```

(5) Question 6. A ceramic capacitor has the numbers 123 printed on it.

Part a) Ceramic capacitors are **nonpolarized**.

Part b) $12*10^3$ pF = 12000 pF = 12nF = 0.012 μF.

(20) Question 7. Design an analog circuit with the following transfer function $V_{out} = 10*(V_{in}-1)$.

First, rewrite as $V_{out} = 10*V_{in}-10$. Use reference for constant. $V_{ref} = 2.5V$. $V_{out} = 10*V_{in}-4*V_{ref}$. Use ground to make sum of gains equal to 1. $V_g = 0$. $V_{out} = 10*V_{in}-4*V_{ref} -5*V_g$. $R_f/R_{in}=10$, $R_f/R_{ref}=4$, $R_f/R_g=5$. A common multiple of 10, 4 and 5 is 100 kΩ.