(50) Question 1. Changes shown in bold. No changes needed for `TIC1handler`.

```c
long Period; /* units of sec */
long Cnt; /* number of sec in one period */
unsigned short mcnt; // 1000 msec per sec
#define resolution 2000
#pragma interrupt_handler TOC3handler()
void TOC3handler(void){
    TFLG1 = 0x08; // Acknowledge
    TC3 = TC3 + resolution; // every 1 ms
    if(++mcnt == 1000){
        mcnt = 0;
        Cnt ++;
   // every 1 sec
    }
    if(Cnt == 0) Overflow = 0xFF;
}
void Ritual(void){
    asm(" sei"); // make atomic
    TIOS |= 0x08; // enable OC3
    TSCR = 0x80; // enable
    TMSK2 = 0x32; // 500 ns clock
    TFLG1 = 0x0A; // Clear C3F, C1F
    TMSK1 = 0x0A; // Arm OC3 and IC1
    TCTL4 = (TCTL4 & 0xF3) | 0x04;
    while((TFLG1 & 0x02) == 0); // wait rising
    TFLG1 = 0x02; // Clear C1F
    TC3 = TCNT + resolution;
    Cnt = 0; Overflow = 0; Done = 0; mcnt = 0;
    asm(" cli");
}
```

Changes shown in bold. A second solution using the MC68HC912D60.

```c
long Period; /* units of sec */
long Cnt; /* number of sec in one period */
#define resolution 62500
void Ritual(void){
    asm(" sei"); // make atomic
    TIOS = 0x08; // enable OC3
    TSCR = 0x80; // enable
    TMSK2 = 0x37; // 16 us clock
    TFLG1 = 0x0A; // Clear C3F, C1F
    TMSK1 = 0x0A; // Arm OC3 and IC1
    TCTL4 = (TCTL4 & 0xF3) | 0x04;
    while((TFLG1 & 0x02) == 0); // wait rising
    TFLG1 = 0x02; // Clear C1F
    TC3 = TCNT + resolution;
    Cnt = 0; Overflow = 0; Done = 0; mcnt = 0;
    asm(" cli");
}
```

(15) Question 2. Which of the following statements are friendly?

(3) Part a) TFLG1 |= 0x01; // no unfriendly also stupid, it clears all bits
(3) Part b) DDRT &= ~0x01; // yes friendly
(3) Part c) TCTL4 |= 0x03; // yes friendly
(3) Part d) TMSK2 = 0x32; // no unfriendly because it affects all channels
(3) Part e) TMSK1 |= 0x01; // yes friendly
(20) **Question 3.** The 7406 has the output low current (max is 40 mA) to drive this solenoid.

![Diagram of 7406 with solenoid](image)

(15) **Question 4.** Let PT1 rise before PT0 rises, but let them both rise during the same instruction (they could also both rise during a time when interrupts are disabled). Since timer channel 0 has a higher priority than timer channel 1, it will be serviced first (see table 4.9 on page 227).