```
#include <mc9s12dp512.h>     /* derivative information */
#include "game.h"

#define DEBOUNCE_DELAY 10

// UP     PT5
// DOWN   PT4
// LEFT   PT3
// RIGHT  PT2
// A      PT1
// B      PT0

static void (*OC6Func) (void);
unsigned static short OC6Delay;
unsigned static short OC6DelayCount1;
unsigned static short OC6DelayCount2;
unsigned static short OC6Count;

void Key_Init(void){
asm sei            // make atomic
  DDRT &= ~0x3F;    // PT7,PT6 all rows are output
  PERT  =  0x3F;    // internal pullup on PT3,PT2
  TCTL3 =  0x05;
  TCTL4 =  0x55;   // falling edges IC3,IC2
  TIOS  =  0xC0;
  TIE   =  0x3F;     // Arm only IC3,IC2
asm cli
}

void enableOC6(void (*function) (void), unsigned short delay, unsigned short delayCount, unsigned s            ¬
hort count) {
  asm sei
  TIE |= 0x40;
  OC6Func = function;
  OC6Delay = delay;
  OC6DelayCount1 = delayCount;
  OC6DelayCount2 = delayCount;
  OC6Count = count;
  TC6 = TCNT + OC6Delay;
  asm cli
}

void interrupt 8 IC0Han(void) {
  Game_B();
  TFLG1 = 0x01;
}

void interrupt 9 IC1Han(void) {
  Game_A();
  TFLG1 = 0x02;
}

void interrupt 10 IC2Han(void) {
  Game_DPad(RIGHT);
  TFLG1 = 0x04;
}

void interrupt 11 IC3Han(void) {
  Game_DPad(DOWN);
  TFLG1 = 0x08;
}

void interrupt 12 IC4Han(void) {
  Game_DPad(LEFT);
  TFLG1 = 0x10;
}

void interrupt 13 IC5Han(void){
  Game_DPad(UP);
  TFLG1 = 0x20;
}

void interrupt 14 OC6Han(void){
  TFLG1 = 0x40;
  if(!OC6DelayCount2) {
    OC6DelayCount2 = OC6DelayCount1;
    (*OC6Func)();
```

```
      OC6Count--;
      if(!OC6Count) {
        TIE &= ~0x40;
      }
    }
    else {
      OC6DelayCount2--;
    }

    TC6 = TCNT + OC6Delay;
  }
```