

**Τμήμα Μηχανικών Η/Υ & Πληροφορικής**

**Εργαστήριο Προηγμένων Μικροεπεξεργαστών**

**Άσκηση 3**

|  |  |
| --- | --- |
| **Ονοματεπώνυμο** | **ΑΜ** |
| Εμμανουήλ Κατεφίδης | 6077 |
| Παναγιώτης Σταυρινάκης | 6217 |

**Κώδικας:**

|  |
| --- |
| #include <sys/types.h>  #include <sys/stat.h>  #include <fcntl.h>  #include <sys/ioctl.h>  #include <unistd.h>  #include <sys/mman.h>  #include <stdio.h>  #include <stdlib.h>  #include <header.h>  #define PIOA\_ID 2  #define TC0\_ID 17  #define BUT\_IDLE 0  #define BUT\_PRESSED 1  #define BUT\_RELEASED 2  #define LED\_IDLE 0  #define LED\_ON 1  #define LED\_HOLD 2  void FIQ\_handler(void);  PIO\* pioa = NULL;  AIC\* aic = NULL;  TC\* tc = NULL;  unsigned int button\_state = BUT\_IDLE;  unsigned int led\_state = LED\_IDLE;  int mask;  int main( int argc, const char\* argv[] )  {  char tmp;  unsigned int gen, i;  STARTUP; //ΑΡΧΙΚΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ  pioa -> PER = 0;  pioa -> OER = 0;  pioa -> CODR = 0;  for( i=0; i<8; i++ )  {  pioa -> PER |= (1)<<(i) ; //ΓΡΑΜΜΗ i: ΓΕΝΙΚΟΥ ΣΚΟΠΟΥ  pioa -> CODR |= (1)<<(i) ; //ΓΡΑΜΜΗ i: ΔΥΝΑΜΙΚΟ ΕΞΟΔΟΥ LOW  pioa -> OER |= (1)<<(i) ; //ΓΡΑΜΜΗ i: ΛΕΙΤΟΥΡΓΙΑ ΕΞΟΔΟΥ  }  tc->Channel\_0.RC = 8192; //8192(period of 1 sec) ~ 5Hz  tc->Channel\_0.CMR = 0x2084; //SLOW CLOCK , WAVEFORM , DISABLE CLK ON RC COMPARE  tc->Channel\_0.IDR = 0xFF; //ΑΠΕΝΕΡΓΟΠΟΙΗΣΗ ΟΛΩΝ ΤΩΝ ΔΙΑΚΟΠΩΝ  tc->Channel\_0.IER = 0x10; //ΕΝΕΡΓΟΠΟΙΗΣΗ ΜΟΝΟ ΤΟΥ RC COMPARE  gen = tc->Channel\_0.SR; //TC0 : ΕΚΚΑΘΑΡΙΣΗ ΑΠΟ ΤΥΧΟΝ ΔΙΑΚΟΠΕΣ  pioa->PER |= (1)<<(9); //ΓΡΑΜΜΗ 9: ΓΕΝΙΚΟΥ ΣΚΟΠΟΥ  pioa->PUER = (1)<<(9); //ΕΝΕΡΓΟΠΟΙΗΣΗ ΣΤΗ ΓΡΑΜΜΗ 9: PULL−UP  pioa->ODR = (1)<<(9); //ΓΡΑΜΜΗ 9: ΛΕΙΤΟΥΡΓΙΑ ΕΙΣΟΔΟΥ  gen = pioa->ISR; // PIOA: ΕΚΚΑΘΑΡΙΣΗ ΑΠΟ ΤΥΧΟΝ ΔΙΑΚΟΠΕΣ  pioa->IER = (1)<<(9); //ΕΝΕΡΓΟΠΟΙΗΣΗ ΔΙΑΚΟΠΩΝ ΣΤΗ ΓΡΑΜΜΗ 9  aic->FFER = (1<<PIOA\_ID) | (1<<TC0\_ID); //ΟΙ ΔΙΑΚΟΠΕΣ 2 ,17 ΕΙΝΑΙ ΤΥΠΟΥ FIQ  aic->IECR = (1<<PIOA\_ID) | (1<<TC0\_ID); //ΕΝΕΡΓΟΠΟΙΗΣΗ ΔΙΑΚΟΠΩΝ : PIOA & TC0  aic->ICCR = (1<<PIOA\_ID) | (1<<TC0\_ID); //AIC : ΕΚΚΑΘΑΡΙΣΗ ΑΠΟ ΤΥΧΟΝ ΔΙΑΚΟΠΕΣ  while( (tmp = getchar()) != 'e')  {    }  aic->IDCR = (1<<PIOA\_ID) | (1<<TC0\_ID); //ΔΙΑΚΟΠΗ ΤΩΝ AIC interrupts  tc->Channel\_0.CCR = 0x02; //ΑΠΕΝΕΡΓΟΠΟΙΗΣΗ ΤΟΥ Timer  CLEANUP;  return 0;  }  void FIQ\_handler(void)  {  unsigned int data\_in = 0;  unsigned int fiq = 0;  unsigned int data\_out;  unsigned int reset = 0;  fiq = aic->IPR; //ΕΝΤΟΠΙΣΜΟΣ ΠΕΡΙΦΕΡΕΙΑΚΟΥ ΠΟΥ ΠΡΟΚΑΛΕΣΕ ΤΗ ΔΙΑΚΟΠΗ    if( fiq & (1<<PIOA\_ID) ) //ΕΛΕΓΧΟΣ ΓΙΑ PIOA  {  data\_in = pioa->ISR; //ΕΚΚΑΘΑΡΙΣΗ ΤΗΣ ΠΗΓΗΣ ΤΗΣ ΔΙΑΚΟΠΗΣ  aic->ICCR = (1<<PIOA\_ID); //ΕΚΚΑΘΑΡΙΣΗ ΤΗΣ ΔΙΑΚΟΠΗΣ ΑΠΟ AIC  data\_in = pioa->PDSR; //ΑΝΑΓΝΩΣΗ ΤΙΜΩΝ ΕΙΣΟΔΟΥ    if( data\_in & 0x200 ) //ΔΙΑΚΟΠΤΗΣ ΠΑΤΗΜΕΝΟΣ(0010 0000 0000(b))  {  if(button\_state == BUT\_RELEASED)  {  button\_state = BUT\_PRESSED;  if( led\_state == LED\_IDLE ) //ΑΝ ΔΕΝ ΑΝΑΒΟΣΒΗΝΕΙ  {  led\_state = LED\_ON;  tc->Channel\_0.CCR = 0x02; //ΔΙΑΚΟΠΗ ΜΕΤΡΗΤΗ  data\_in = tc->Channel\_0.SR; //TC0 : ΕΚΚΑΘΑΡΙΣΗ ΑΠΟ //ΤΥΧΟΝ ΔΙΑΚΟΠΕΣ  tc->Channel\_0.RC = 8192; //8192(period of 1 sec) ~ //5Hz  tc->Channel\_0.CCR = 0x05; //ΕΝΑΡΞΗ ΜΕΤΡΗΤΗ  }else if( led\_state == LED\_ON )  {  led\_state = LED\_HOLD;  tc->Channel\_0.CCR = 0x02; //ΔΙΑΚΟΠΗ ΜΕΤΡΗΤΗ  data\_in = tc->Channel\_0.SR; //TC0 : ΕΚΚΑΘΑΡΙΣΗ ΑΠΟ //ΤΥΧΟΝ ΔΙΑΚΟΠΕΣ  tc->Channel\_0.RC = 8192/2; //8192/2(period of 0.5s //~ 2Hz  tc->Channel\_0.CCR = 0x05; //ΕΝΑΡΞΗ ΜΕΤΡΗΤΗ  }else{  led\_state = LED\_IDLE;  tc->Channel\_0.CCR = 0x02; //ΔΙΑΚΟΠΗ ΜΕΤΡΗΤΗ  pioa->CODR = 0x100 & 0x200;  //(0001 0000 0000 & 0010 0000 0000)(b)  }  }  }else{  if(button\_state == BUT\_PRESSED)  {  button\_state = BUT\_RELEASED;  reset = 0;  }  }  }    if( fiq & (1<<TC0\_ID) )  {  data\_out = tc->Channel\_0.SR;//ΕΚΚΑΘΑΡΙΣΗ ΤΗΣ ΠΗΓΗΣ ΤΗΣ ΔΙΑΚΟΠΗΣ  aic->ICCR = (1<<TC0\_ID); //ΕΚΚΑΘΑΡΙΣΗ ΔΙΑΚΟΠΗΣ ΚΑΙ ΑΠΟ AIC  if(led\_state == LED\_ON)  {  counter++;  if (counter > 59) counter = 0;  pioa->SODR = counter%10<<4 | counter/10;  if (button\_state == BUT\_PRESSED) reset += 2;  }else if( led\_state == LED\_HOLD )  {  if(pioa->ODSR & 0x100) pioa->CODR |= 0x100;  else pioa->SODR |= 0x100;  if (button\_state == BUT\_PRESSED) reset += 1;  }  if(reset > 1) {  counter = 0;  pioa->SODR = counter%10<<4 | counter/10;  reset = 0;  button\_state = BUT\_IDLE;  button\_state = BUT\_RELEASED;  }  tc->Channel\_0.CCR = 0x05; //ΕΝΑΡΞΗ ΜΕΤΡΗΤΗ  }  } |