Zeno Scheltens

[z.f.scheltens@st.hanze.nl](mailto:z.f.scheltens@st.hanze.nl)

ELVE1 groep 3

Kit Yi Feng

[k.y.feng@st.hanze.nl](mailto:k.y.feng@st.hanze.nl)

ELVM1 groep 1

Ontwerp en evaluatie document programmeren

Interrupts

Ledje aan, ledje uit

Inhoudsopgave

[1. Analyse 2](#_Toc509307585)

[2. Ontwerp 3](#_Toc509307586)

[Flowcharts 3](#_Toc509307587)

[3. Implementatie 4](#_Toc509307588)

[4. Evaluatie 5](#_Toc509307589)

# 1. Analyse

Voor deze opdracht moet een programma geschreven worden met interrupt. Zoasl zijn naam, Interrupt werkt volgens een interruptie. Met interrupt wordt het hoofdprogramma tijdelijk onderbroken om een specifieke taak uit te voeren.

In de powertpoint wordt het programma al geschreven, een verandering die moet komen is de timer en de offset van de timer. In het voorbeeld programma staat de timer op 1 seconde, deze moet veranderd naar 3 seconden. Deze kan veranderd worden door een berekening:

*Voor Timer0* = 8MHz/4/… = 46875 Hz

*Voor offset:*

65536 – 46875 = 18661

*Voor TMR0H* = offset/256; =18661/256=72

72 in hexadecimale getal is 0x48.

*Voor TMR0H* = rest van TMR0H\*256; =0,8953…\*256=229

229 in hexadecimale getal is 0xE5.

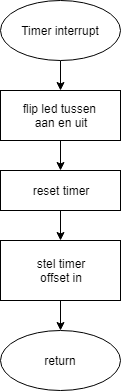
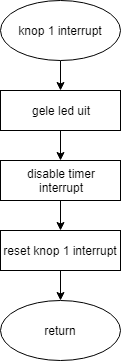
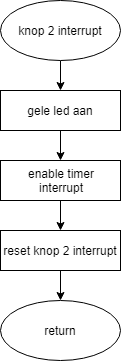
Voor deze opdracht moet de onderstaande eisen voldoet worden, de formele eisen:

* De groen led laten knipperen (3 seconden aan en 3 seconden uit).
* Met button 1 de led’s (zowel de groene als de gele led) laten branden en button 2 uitdoen.

# 2. Ontwerp

## Flowcharts

Hier staat de main flowchart en de interrupt flowcharts:



# 3. Implementatie

1. /\*
2. \*  Embedded Programming 2018
3. \*
4. \*  Opdracht: Timers
5. \*
6. \*  Code by
7. \*      - Zeno Scheltens
8. \*      - Kit Yi Feng
9. \*
10. \*/
12. #include <stdio.h>
13. #include <xc.h>
14. #include "fuses.h"
16. #define \_XTAL\_FREQ 8000000  // X-tal = 8 MHz
18. #define button1 PORTbits.RB0
19. #define button2 PORTbits.RB1
20. #define yellow LATCbits.LATC7
22. **void** interrupt myIsr (**void**) {
24. // software interrupt
25. **if** (INTCONbits.TMR0IE && INTCONbits.TMR0IF) {       // enables TMR0 overflow interrupt - TMR0 reached overflow?
26. INTCONbits.TMR0IF = 0;                          // reset TMR0, clears the overflow
27. LATCbits.LATC4 = ~LATCbits.LATC4;
28. // offset, 18661
29. TMR0H = 0x48;                                   // 72
30. TMR0L = 0xE5;                                   // 229
31. }
33. **if** (INTCONbits.INT0E && INTCONbits.INT0F)   {       // enables external interrupt - did the interrupt happen
34. yellow = 0;
35. INTCONbits.TMR0IE = 0;                          // disables TMR0 interrupt
36. INTCONbits.INT0F = 0;                           // clears this interrupt
37. }
39. **if** (INTCON3bits.INT1E && INTCON3bits.INT1F) {   // enables external interrupt - did the interrupt happen
40. yellow = 1;
41. INTCONbits.TMR0IE = 1;                          // enables TMR0 interrupt
42. INTCON3bits.INT1F = 0;                          // clears this interrupt
43. }
44. }

47. **void** main() {
49. // ini
50. TRISC = 0x00;
51. LATC = 0x00;
52. TRISBbits.TRISB0 = 1;
53. TRISBbits.TRISB1 = 1;
54. ANCON1bits.ANSEL10 = 0;
55. ANCON1bits.ANSEL8 = 0;
56. T0CON = 0b10000110;             // enables timer 1 as 16 bit with a /128 prescale
57. INTCON2bits.TMR0IP = 1;         // overflow interrupt priority
58. INTCONbits.TMR0IF = 0;          // clears the timer overflow
59. INTCONbits.TMR0IE = 1;          // enables the timer interrupt
60. INTCON2bits.INTEDG0 = 0;        // interrupt on rising/falling
61. INTCONbits.INT0E = 1;           // enables external interrupt on button 1
62. INTCON3bits.INT1E = 1;          // enables external interrupt on button 2
63. INTCON2bits.INTEDG1 = 1;        // interrupt on rising/falling
65. // enables all interrupts
66. ei();
68. **while**(1);
69. }

# 4. Evaluatie

In de analyse is een formele eis voor het programma opgesteld, deze is:

Formele eisen:

* De groen led laten knipperen (3 seconden aan en 3 seconden uit).
* Met button 1 de led’s (zowel de groene als de gele led) laten branden en button 2 uitdoen.

Om het programma te testen is simpel weg de PICkit18 pakken, een verbinding maken met de boord en het programma laten compiler.

Na het testen bleek alles te werken volgens de eisen.

Functionele eisen:

* De groen led laten knipperen (3 seconden aan en 3 seconden uit).
* Met button 1 de led’s (zowel de groene als de gele led) laten branden en button 2 uitdoen.