

Dispositius Programables

Durada: 2h

Control - Novembre 2021

1. A memòria de dades hi ha el valor 45345 de tipus `uint16_t`, això vol dir que és un valor de 16bits sense signe. Aquest valor es troba a partir de l'adreça corresponent a l'etiqueta *variable*. La disposició a memòria és de tipus Little Endian. Us pot ser d'utilitat els modificadors d'assemblador `hi8()` i `lo8()` que extreuen respectivament el byte alt i el byte baix d'un valor constant de 16 bits.
 - a) Indica el valor del byte en representació hexadecimal que es troba al registre de memòria de dades corresponent a l'etiqueta *variable*. (0.5)
 - b) Indica el valor del byte en representació hexadecimal que es troba al registre de memòria de dades corresponent a l'etiqueta *variable + 1*. (0.5)
 - c) Escriu la macro **llegeixX** que té com a paràmetre una adreça qualsevol de la memòria de dades i carrega al registre X el valor de tipus `uint16_t` que es troba a l'adreça que es passa com a paràmetre. (1)
 - d) Escriu la macro **addiX** que suma un valor constant de tipus `uint16_t` que es passa com a paràmetre al contingut del registre X. Els flags Z i C del registre d'estat SREG han de quedar com toquen. (1)
 - e) Modifica o justifica els canvis que s'haurien de fer a les macros **llegeixX** i **addiX** si considerem ara un valor de tipus `int16_t` (valor de 16 bits amb signe). En aquest cas, els flags S, V i Z han de quedar com toquen. (1)
2. Defineix una subrutina anomenada **comparaREF** que compari el valor que es passa com a paràmetre utilitzant els registres `r1:r0` amb la constant definida pel símbol `REF`. Aquesta constant assignada al símbol `REF` es defineix amb la directiva `.equ` d'assemblador. També disposeu dels modificadors `hi8()` i `lo8()` descrits a l'exercici anterior. Feu la subrutina transparent. Considereu que els valors són de tipus `uint16_t`. Si el valor de `r1:r0` i `REF` són iguals s'activarà el flag Z. Si `r1:r0 < REF` s'activarà el flag C (`C=1`) i si `r1:r0 > REF` es desactivarà el flag C (`C=0`). (1)
3. A partir de la subrutina anterior, defineix la subrutina **comparaREFescriu** que envia pel port sèrie el caràcter '=' si el valor de `r1:r0 = REF`, envia '-' si `r1:r0 < REF` i envia '+' si `r1:r0 > REF`. Considereu que disposeu de la subrutina **tx**. (1)
4. Quins canvis caldria fer a les subrutines **comparaREF** i **comparaREFescriu** per considerar que els valors siguin de tipus `int16_t` (amb signe). (1)
5. Es vol dissenyar un sistema que comprovi que si el temps entre dues pulsacions està en el marge de 1 a 2 segons. Es disposa d'un polsador a la pota corresponent a `INT0` (bit 2 del set de potes D). Es configura la interrupció `INT0` (`__vector_1`) per a que generi una interrupció de flanc descendent cada vegada que es produeix una pulsació. Es configura el timer 0 en mode CTC de manera que es genera una interrupció del comparador B (`__vector_15`) quan ha passat 1 segon des que el timer s'ha posat a 0. També es configura la interrupció del comparador A (`__vector_14`) quan han passat 2 segons. Es defineixen les macros `ledon`, `ledoff` i `restarttimer`. Les macros `ledon` i `ledoff` encenen i apaguen el led de l'Arduino. La macro `restarttimer` posa a 0 el valor del registre d'entrada / sortida `TCNT0` (registre comptador del timer 0).
 - a) Dibuixa el graf corresponent a la implementació d'aquest sistema. Heu de considerar que els esdeveniments són cadascuna de les interrupcions i que s'ha d'indicar en el graf quines són les accions que es fan. (2)
 - b) Implementa en pseudocodi el graf de la figura (1)
 - c) Implementa en assemblador el graf de la figura (1)

