

Control Dispositius Programables

8 de novembre de 2023

Enginyeria de Sistemes TIC

120 MINUTS

L'objectiu d'aquest control es aconseguir un sistema basat en l'Arduino que respon '<', '>' o '=' en funció de si la paraula que li arriba és posterior, anterior o igual a la paraula introduïda anteriorment, segons l'ordenació alfabètica. Aquesta classificació només tindrà en compte la primera lletra de la paraula i es donarà al final de la introducció de la paraula. Les següents consideracions s'han de tenir en compte per a tots els exercicis:

- a) La separació entre paraules ve donada tant per l'espai ' ' com pel retorn de carro.
- b) La introducció en majúscules o minúscules no ha d'afectar a l'ordenació.
- c) La introducció de paraules, així com la presentació de resultats es farà pel canal de comunicació sèrie.
- d) Per a cada exercici proposat podeu aprofitar qualsevol de les macros o subrutines que estiguin definides en altres exercicis, encara que no les hagueu escrit.
- e) Totes les subrutines o macros que es defineixin han de ser transparents.
- f) Considereu que en l'estat inicial ja s'ha introduït una paraula que comença per 'A'.

El caràcter corresponent al retorn de carro és el '\r' amb codi ASCII 13. L'espai té codi ASCII 32. La lletra 'a' té codi ASCII 97, la lletra 'A' té codi ASCII 65. Els caràcters de l'abecedari anglès (els de la taula ASCII) són un total de 25.

Exercici 1 [1 punt]. Dissenyeu la macro *is_letter* que comprova si el que hi ha al registre *r16* correspon a una lletra de l'abecedari anglès. En cas afirmatiu activa el flag **Z** i en cas negatiu el desactiva. El valor del registre *r16* queda com estava.

Exercici 2 [1 punt]. Dissenyeu la subrutina *majuscula* que converteix a majúscula, en cas que no ho sigui, la lletra que es troba al registre *r16*. S'assumeix que hi haurà una lletra a *r16*, i per tant no cal verificar-ho.

Exercici 3 [1 punt]. Dissenyeu la subrutina *ordena* que compara la lletra que es troba al registre *r1* amb la lletra que es troba al *r0*. Si *r1* > *r0* seguint l'ordre alfabètic, carrega a *r16* el caràcter '>'. Si als 2 registres es troba la mateixa lletra, carrega a *r16* un '='. Si *r1* < *r0* carrega '<'. S'assumeix que hi haurà lletres en els 2 registres, i per tant no cal verificar-ho.

Exercici 4 [1 punt]. Dissenyeu la macro *is_char1_or_char2* amb els paràmetres *char1* i *char2* que activa el flag **T** si el valor que es troba al registre *r16* és igual a algun d'aquests 2 paràmetres. En cas negatiu deixa el flag **T** desactivat. El valor de *r16* el deixa intacte. Aquesta macro es pot aprofitar més endavant per detectar el final de paraula.

Exercici 5 [1 punt]. Per treballar amb el LED calen un conjunt de subrutines o macros. Recordeu que aquest LED correspon al bit 5 del set de pots **B**.

Justifica per cadascun dels casos següents la idoneïtat d'usar subrutina o macro i escriu el codi corresponent.

- a) *led_init*, que configuri adequadament el LED de l'Arduino.
- b) *led_on* i *led_off*.
- c) *led_toggle*.

Exercici 6 [2 punts]. Com a objectiu previ, ens proposem realitzar un programa que encengui el led de l'Arduino just després d'escriure la primera lletra d'una paraula i que l'apagui just en el moment que s'acaba la paraula. Qualsevol caràcter que no siguin les lletres de l'abecedari o els caràcters separadors de paraula han de ser ignorats. Feu servir la interrupció de recepció de port sèrie. Es disposa de la macro o subrutina *serial_init* que configura adequadament el port sèrie.

- a) La solució proposada segueix el patró d'una màquina d'estats, dibuixeu el graf corresponent.
- b) Escriu el programa en ensamblador.

Exercici 7 [3 punts]. Finalment, escriu el programa en ensamblador que satisfà l'enunciat inicial.