

# Dispositius Programables

Final - Gener 2016

1. Considereu el següent programa.

```
1  .global main
2  .global __vector_18
3  .global __do_copy_data
4  .global __do_clear_bss
5
6  .section .data
7  patro: .byte 'L'
8         .byte 'E'
9         .byte 'D'
10 patrof:
11 N = patrof - patro
12
13 .section .bss
14 finstr: .byte 'L'
15         .byte 'E'
16         .byte 'D'
17 finstrf:
18
19 .section .text
20 cap:    cpi r26, lo8(finstr+N)
21         breq vcap
22         ret
23 vcap:   ldi r27, hi8(finstr)
24         ldi r26, lo8(finstr)
25         ret
26
27 cua:    cpi r26, lo8(finstr)
28         breq vcua
29         ret
30 vcua:   ldi r27, hi8(finstr+N)
31         ldi r26, lo8(finstr+N)
32         ret
33
34 correl: ldi r29, hi8(patro+N)
35         ldi r28, lo8(patro+N)
36         clc
37
38         call cua
39         ld r3, -X
40         ld r4, -Y
41         sbc r3, r4
42
43 corr1:  call cua
44         ld r3, -X
45         ld r4, -Y
46         sbc r3, r4
47
48 corr2:  call cua
49         ld r3, -X
50         ld r4, -Y
51         sbc r3, r4
52
53 corr3:  brsh on
54 off:    call ledoff
55         ret
```

```

56  on:      call ledon
57          ret
58
59  __vector_18:
60          lds r16, UDR0
61          call tx
62          st X+, r16
63          call cap
64          call correl
65          reti
66
67  main:    call config_serial
68          call config_led
69
70          ldi r27, hi8(finstr)
71          ldi r26, lo8(finstr)
72
73          call ledoff
74          sei
75
76  loop:    rjmp     loop
77          ret

```

- a) Nombra i delimita en funció dels números de línia les subrutines que existeixen.
  - b) Quina funció fa la subrutina CAP? Aquesta subrutina fa servir algun paràmetre i retorna algun resultat? En cas afirmatiu, indica quin.
  - c) La subrutina CAP modifica el registre d'estat? En cas afirmatiu, a quins flags afecta?
  - d) Quina diferència hi ha entre el flag S i el N? Quina diferència hi ha entre el flag V i el C?
  - e) Transforma la rutina CAP en transparent.
  - f) Quin valor inicial tenen les posicions de memòria compreses entre patro:patrof i entre finstr:finstrf? Cal tornar a injectar el programa a l'AVR després de treure l'energia per a que aquestes posicions de memòria tinguin aquests valors inicials?
  - g) A les línies 38-51 es detecten repeticions del mateix codi. Defineix una macro per poder cridar repetidament a la macro en comptes d'aquest codi.
  - h) A les línies 38-51 es detecten repeticions del mateix codi. Substitueix aquestes línies per un bucle. Aquest bucle es pot deixar en funció de N?
  - i) Considera que pel port serie es rep la següent seqüència de caràcters ABCLEDD. Per a cada caràcter rebut indica quin serà el contingut de: la memòria de dades entre les adreces finstr:finstrf, el registre X i l'estat del led. (x3)
  - j) És possible substituir la instrucció SBC r3, r4 per CPC r3, r4 i el programa continuaria fent el mateix? Justifica la resposta.
  - k) Pots definir en una frase simple la funció que realitza aquest programa?
2. Sigui una zona de memòria de dades definida per l'adreça *taula* de mida 16 bytes amb un valor inicial corresponent al caràcter '\*'. Aquesta zona es pot omplir amb paraules que poden ser de mida variable. Per indicar fins on arriba la paraula, es pacta un sentinella corresponent al byte 0.
    - a) Defineix en ensamblador aquesta zona de memòria.
    - b) Disseny una subrutina que admeti com a paràmetre una taula d'aquest tipus i que la modifiqui deixant la paraula capgirada. (x3)
  3. Sigui la subrutina *fun* transparent que rep un paràmetre a r16 i retorna el resultat al mateix r16.
    - a) Disseny la subrutina *funp* que faci el mateix i sigui transparent, però fent servir el mecanisme de pas de paràmetres per pila. Indica igualment què cal fer en el fil principal abans de cridar a *funp*. (x3)
    - b) Dibuixa l'estat de la pila just abans de cridar a la subrutina *fun*. (x2)