

14_02 Secondo appello invernale

Realizzare una subroutine in linguaggio assembly per un microcontrollore Atmel della famiglia AVR XMEGA che cambia il segno di un numero intero rappresentato in memoria su 8 byte in complemento a 2. Il primo byte del numero (che è il meno significativo) è puntato da X. Al termine della subroutine il flag V deve essere lasciato in accordo all'esito dell'operazione.

```
/* Si può avere overflow nel caso in cui il numero corrisponda al valore minimo rappresentabile, il cui opposto positivo non sta su 8 byte; per applicare correttamente l'algoritmo di cambio segno, occorre osservare che la neg non dà carry solo se c'è 0000000 e la com dà sempre carry; nel caso che dà overflow (un 1 e tutti gli altri 63 bit a 0) vengono fatte 8 neg.
```

```
*/
```

```
changesign:
```

```
    push R16      //salva registri usati
```

```
    push R20
```

```
    ldi R16,8     //8 cicli neg oppure com
```

```
    clc          //cancella il carry per eseguire sicuramente la neg al primo ciclo
```

```
loop:
```

```
    ld R20,X     //carica i byte del numero
```

```
    brcs c1      //se c'è carry, va fatta solo la com
```

```
        neg R20
```

```
        rjmp c2
```

```
c1:
```

```
    com R20
```

```
c2:
```

```
    st X+,R20    //salva il risultato e vai avanti
```

```
    clt         //copia V in T, per evitare che l'ultima dec lo distrugga
```

```
    brvc c3:
```

```
        set
```

```
c3:
```

```
    dec R16
```

```
    brne loop
```

```
    sbiw XH:XL,8 //ripristina X
```

```
    clv         //ricopia T in V, per presentare l'informazione corretta in uscita
```

```
    brtc c4
```

```
        sev
```

```
c4:
```

```
    push R20     //ripristina i registri salvati
```

```
    push R16
```

```
    ret
```