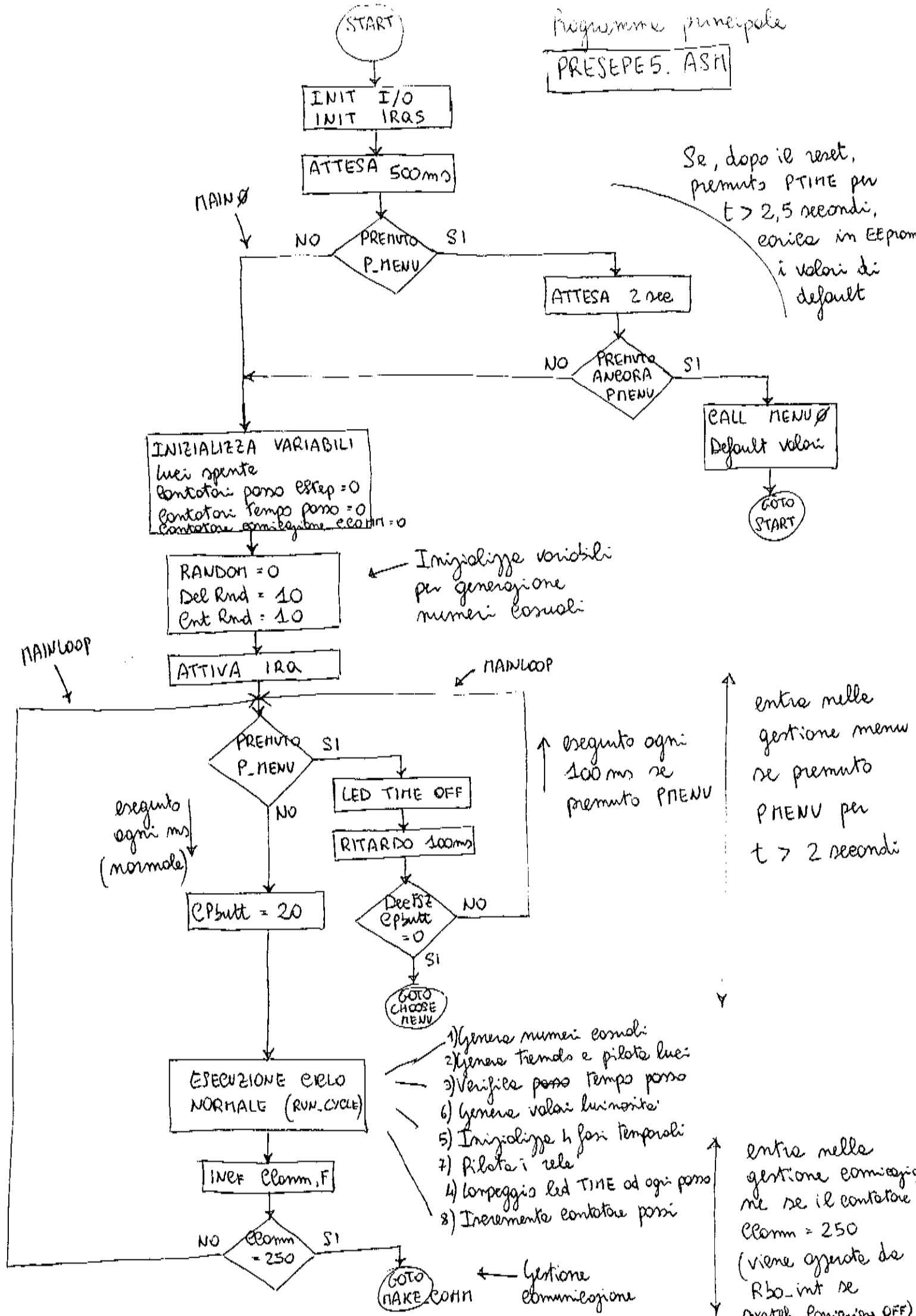


Programma principale PRESEPE5.ASH



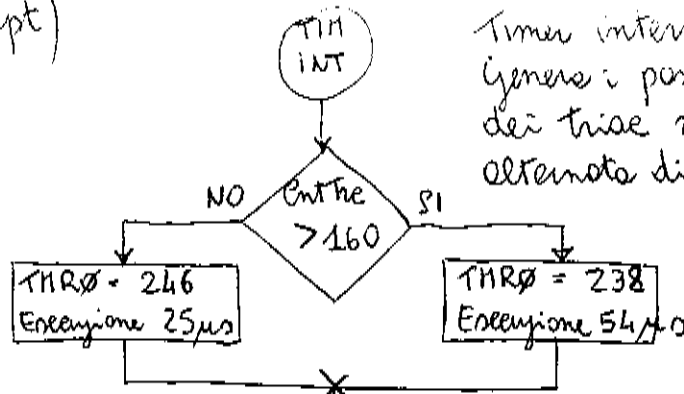
IRQS (interrupt)

Rilevare THRO
(prossima exec. irq)
secondo contatore
CntTre per entrare
seotti di luminorita
(vedi foglio mee.)

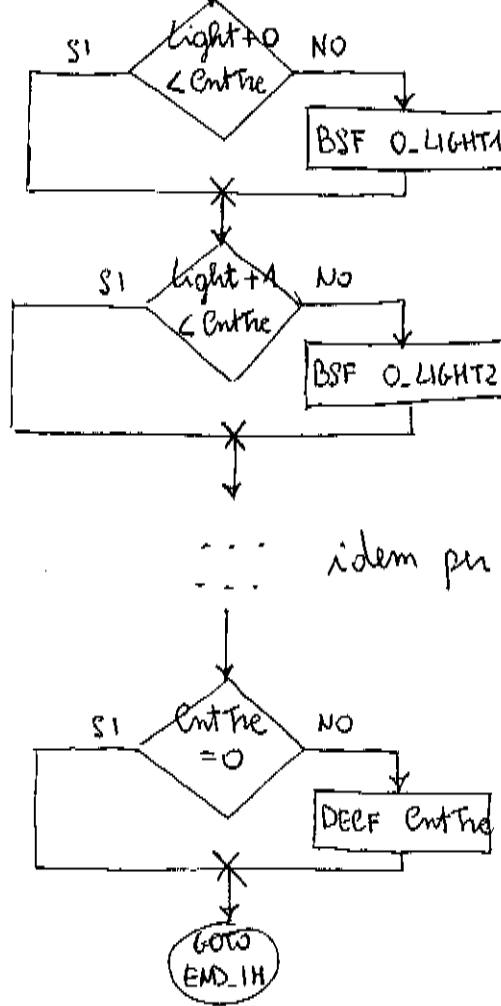
Attiva l'uscita
collegate al triee
della lampada
in ritardo rispetto
allo zero, quando
 $Light(x) \geq CntTre$

Decrements
variabile
CntTre dei
pomi condugione
del triee.

Timer interrupt
Genera i pomi del ritardo di condugione
dei triee rispetto allo 0 della tensione
alternata di rete \rightarrow luminorita
lampade con triee



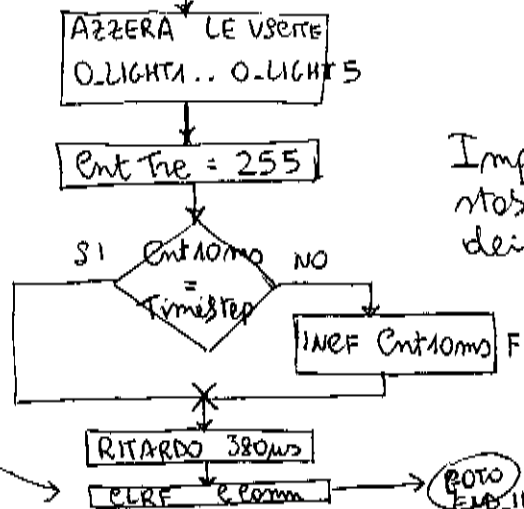
Eseguite ogni 25/54µs

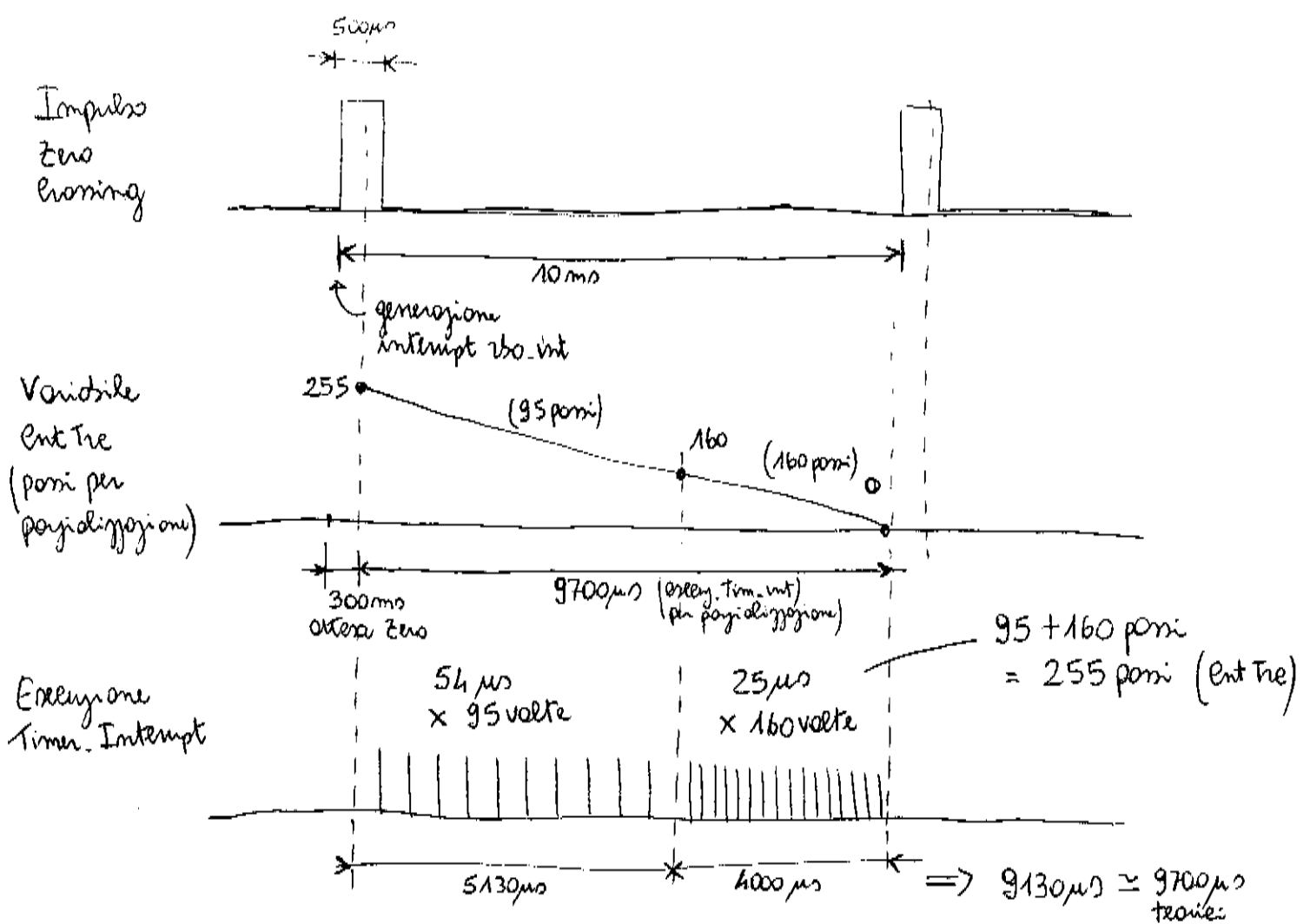


RBO interrupt
Eseguito quando la tensione di rete
porta sullo zero

Aggiore variabile contatore
Ccntm per non far
entrare il mainloop
nella routine di
comunicazione
(quando Ccntm = 250)

Importa contatore al massimo
stabilisce i pomi di condugione
dei triee.

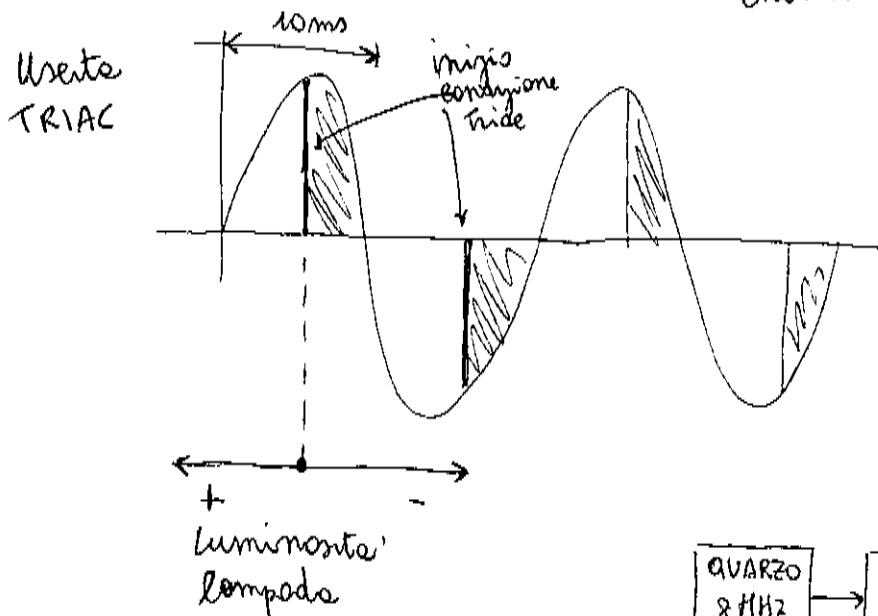




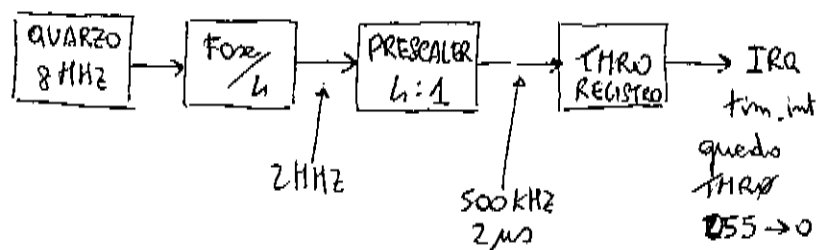
Per rendere l'aumento/decremento di luminosità più regolare, si preferisce aumentare i "ponsi" della paginazione nella parte discendente di ciascuna semionda.

La variabile Ent Tre stabilisce il ritardo di conduzione del Triac. È impostata al massimo (255) in 150-int (eseguito sullo zero crossing) e continuamente decrementata ad ogni esecuzione di tim-int (ogni 54 o 25µs).

Ent Tre > 160 ← → Ent Tre < 160



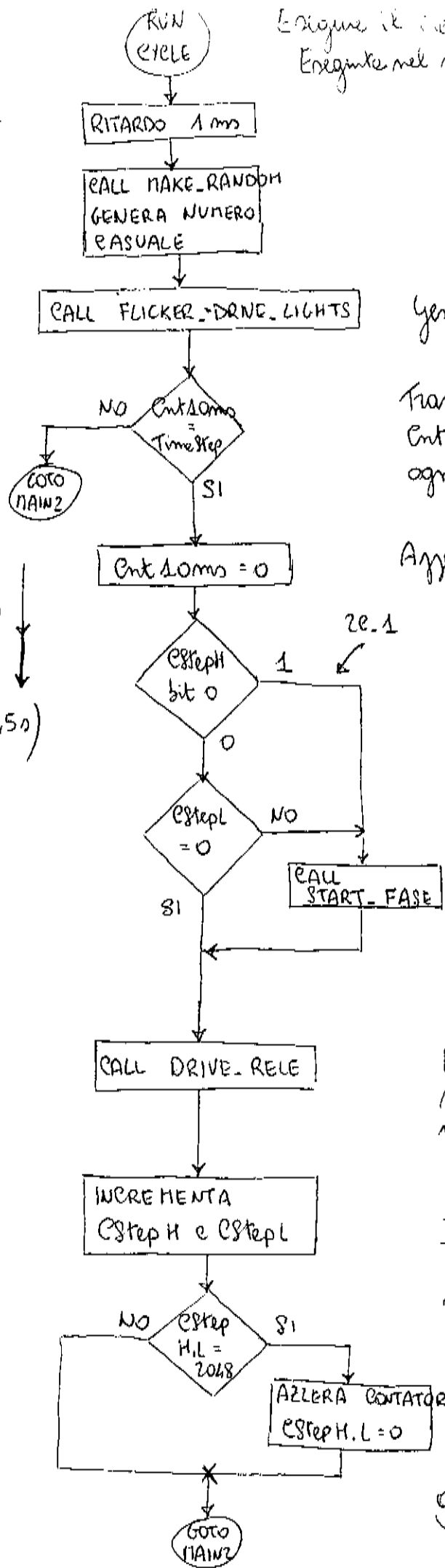
CONFIGURAZIONE TIM-INT



eseguito
ogni 1ms
linea

Esegue il ciclo di funzionamento normale
Eseguito nel mainloop

eseguito
ed ogni passo
ciclo, in
funzione del valore
timestep (da 20ms a 2,5%)



Genera effetto tremolio / pilota la luce

Trascorso tempo singolo passo?
Ent10ms è incrementata da 250 int
ogni 10ms

Aggiorna contatore

Se CStepH,L è un multiplo
di 512 (0, 512, 1024, 1536)
cioè bit 0 CStepH=0 e CStepL=0
gestisce l'inizio di ciascuna
fase prelevando i valori utente
da EEPROM

Pilota i 5 relè secondo l'onda
mento prelevato dai settaggi
in memoria EEPROM

Incrementa contatore passi del
ciclo.
2048 passi totali →
512 passi per ogni fase (4)

2048 =
0000 1000 0000 0000
CStepH CStepL

eseguito ogni ms

Genera un nuovo numero casuale che si aggiunge in un tempo variabile

eseguito da 10 a 100 ms

Ritardo variabile da 10ms a 100ms

Logica XOR per generare bit da far entrare a destra secondo i bit 5,6 di Random

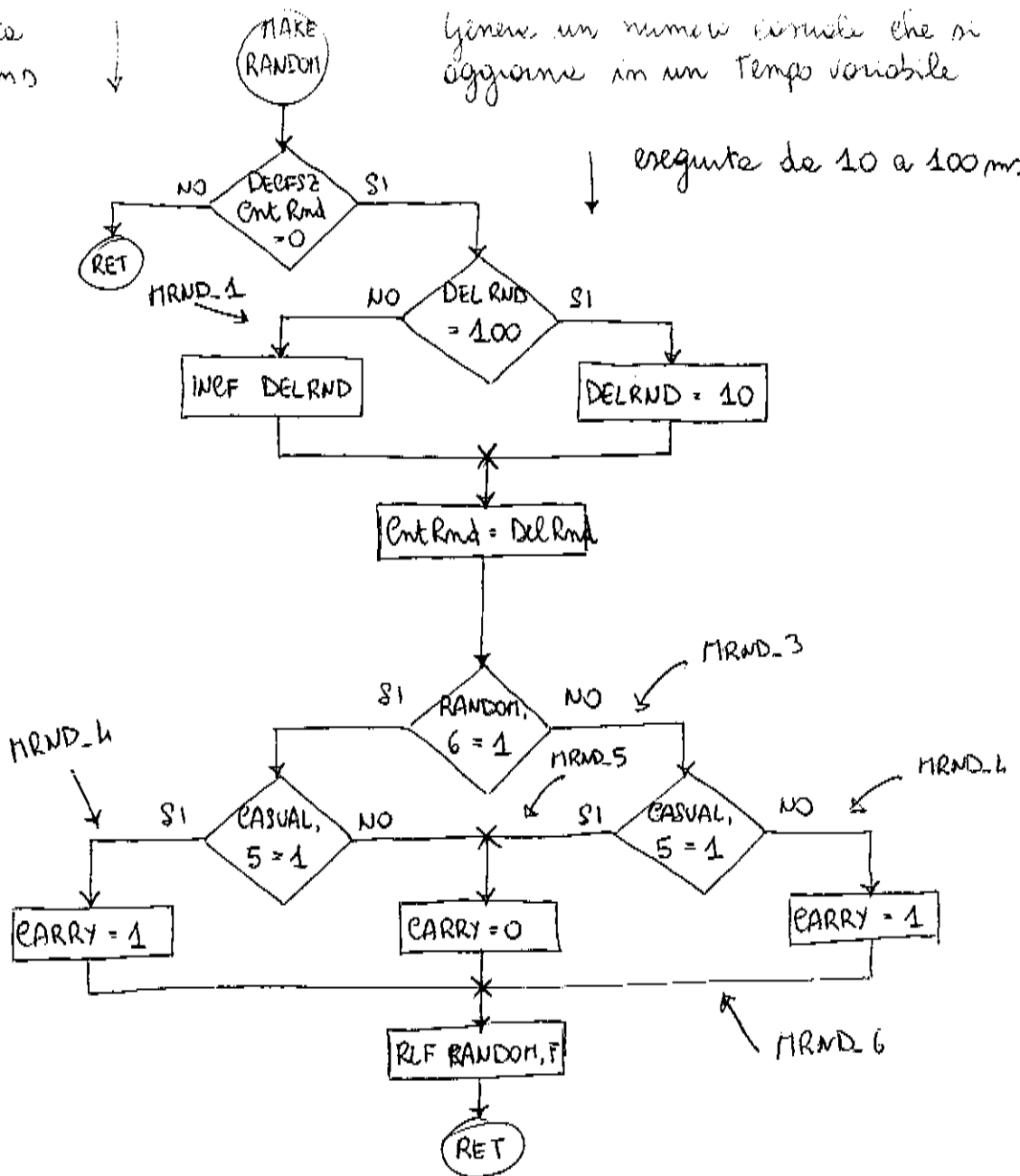
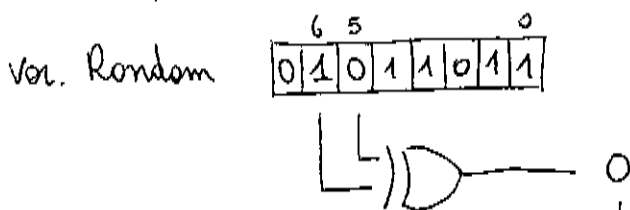


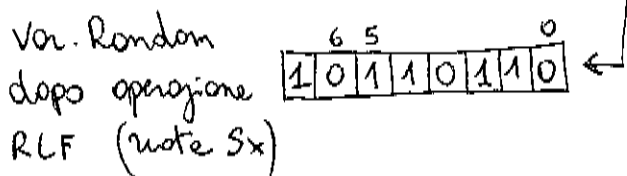
Tabelle logiche

Random. 6	Random. 5	XOR
0	0	1
0	1	0
1	0	0
1	1	1

Esempio



il risultato della XOR entrerà in Random a destra (bit 0) dopo l'operazione RLF (ruota a sinistra)



light +0
+1
+2
+3
+4 } OUTPUT

LtValue +0
+1
+2
+3
+4 } INPUT

LtMode +0
+1
+2
+3
+4 } INPUT

Input: Random
Output: Count1, Value, Temp

Pilote la luminosita' delle 5 lampade e genera l'effetto tremolo se selezionato. Emigite ogni ms circa

Valore Random per effetto tremolo

Contra in Temp il valore assoluto della lampada intersesta (la luminosita')

Individua variabile stato lampad

abilitato tremolo

valore bit random

LUMINOSITA' VALORE ASSOLUTO

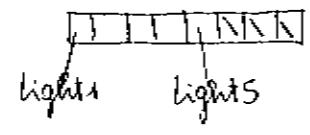
LUMINOSITA' VALORE ASSOLUTO - 25%

Rende il tremolo uguale al valore della luminosita' SI

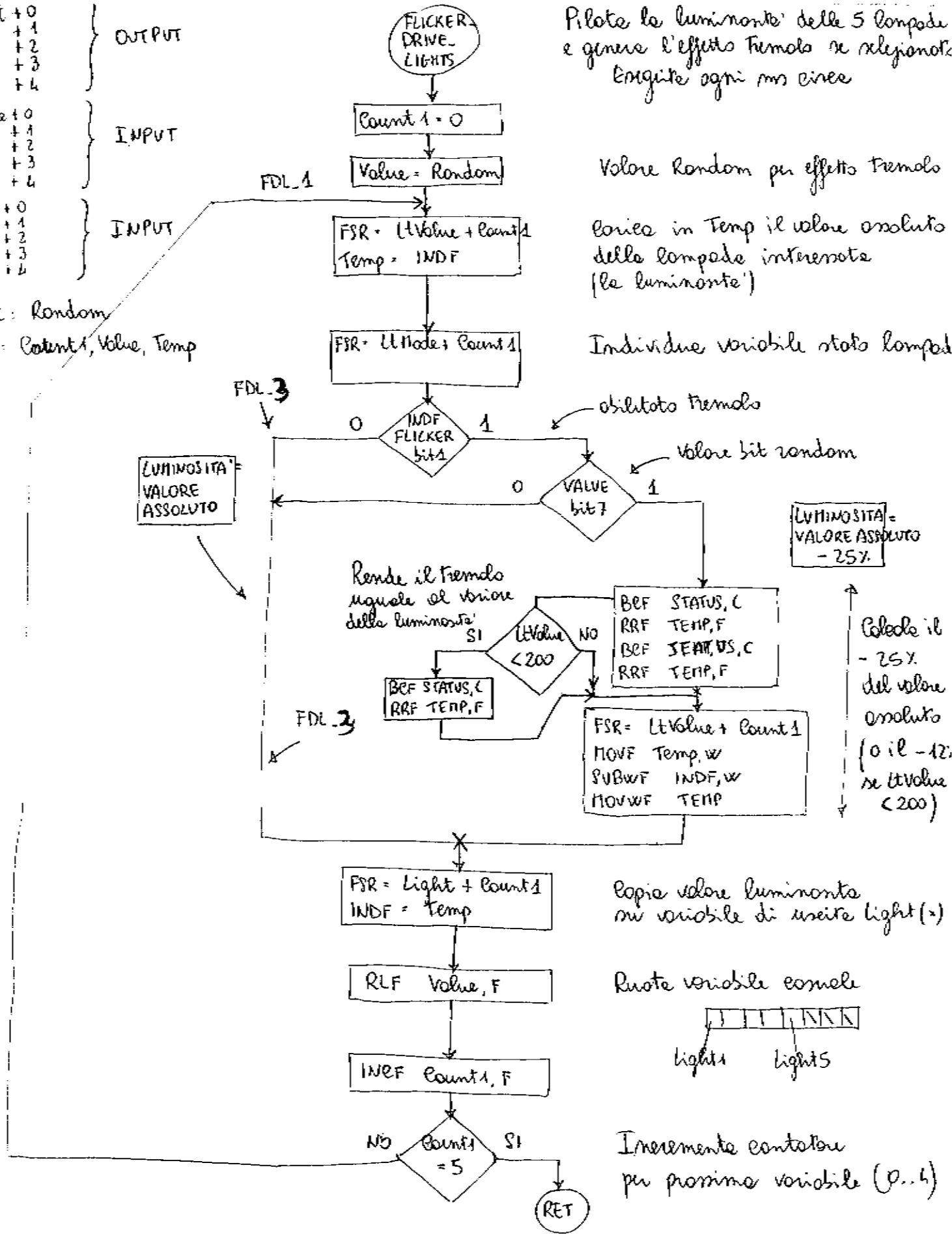
Calcola il - 25% del valore assoluto (o il -12% se LtValue < 200)

Copia valore luminosita su variabile di uscita light(*)

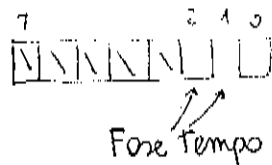
Ruota variabile come



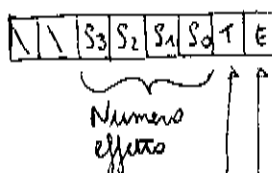
Incrementa contatore per prossima variabile (p..4)



ESTEPH



LtNode
(0..4)



Tremolo
1=ON 0=OFF

Abilitazione
1=ON 0=OFF

START
FASE

```
RRF EStepH, W
ANDLW B'0000 0011
CALL READ_EEPROM
ANDLW B'0000 1111
MOVWF TEMP

PRELEVA VALORE DA
TABELLA TBLTIMESTEP
ALL'INDIRIZZO TEMP
```

Istruzione che carica il valore della fase
prelevando dalla memoria EEPROM il
settaggio utente. Eseguita all'inizio di
ogni fase temporale.

* 1 *

Preleva settaggio Tempo (0..15) da locazione
EEPROM 0..3 secondo la fase corrente,
indicata dal bit 2,1 di EStepH
Preleva valore TimeStep dalla tabella ROM
secondo l'offset dell'impostazione letta
dalla memoria EEPROM

```
Value = Random
FSR = LtNode
Count1 = 0
```

* 2 *

Preleva ondammento + stati delle 5 lampade
della memoria EEPROM

STF-1

```
RRF EStepH, W
ANDLW B'0000 1111
ADDLW 0x04
MOVWF Temp
Count2 = Count1
```

Temp = fase 0..3

STF-2

Moltiplica per calcolare
l'indirizzo delle variabili
da raggiungere in EEPROM

STF-3

SI
Count2 = 0
NO

```
PRELEVA DA
EEPROM ALL'
INDIRIZZO Temp
MASEMERA 0011 1111
CARICA SU INDF
```

```
Temp = Temp + 4
DECF Count2, F
```

Temp = Temp + Count1 * 4

Indirizzo
EEPROM
4..23

numero lampade
0..4

INDF = LtNode(0..4)
Andamento + stati delle 5 lampade

STF-4

Enable light

INDF bit0 = 1

Random - enable

INDF bit0 (LtNode)
0
1

1

Value bit0

0

INDF bit0 = 0

STF-5

Enable light

Trasforma il bit LtNode
secondo l'impostazione
random on-off

Value, 0 contiene un bit
casuale

STF-6

```
RRF VALUE, F
```

```
INCF FSR, F
```

```
INCF Count1, F
```

Prossimo bit casuale per prossime lampade

Prossima variabile LtNode(x)

Incrementa contatore per prossime lampade

STF-1

NO
Count1 = 5
SI

* 3 *

Problema da EEPROM il valore

```

RRF CSTEP H,W
ANDLW B'0000 00M'
ADDLW D'24'
CALL READ_EEPROM
MOVWF EmRel

```

Funzione di controllo del test corrente (indirizzo 26..27)

Adden = Fore + 24
0..3

VALUE = SWAPF (Random)

Copia valore casuale invertito i bit LSB con i bit MSB

Operazione logica bit a bit
EmRel + Random
[vedi sotto] !!

```

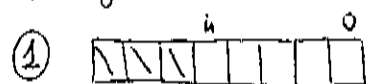
CONF EmRel, W
IORWF Value, W
ANDLW B'0001 11M'
MOVWF EmRel

```

Trasforma EmRel
- da abilitazione utente
- ad abilitazione rel'

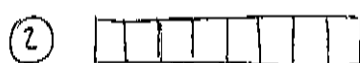
RET

Logica per calcolo EmRel (da abilitazione utente ad abilitazione rel')
Operazione bit a bit delle variabili ① e ②



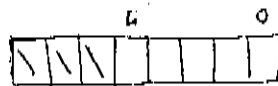
Enable Random (abilitazione random utente)

Input



Value Random (valore casuale)

Input



EmRel (abilitazione rel')

Output

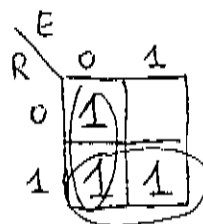
Tabella di verità

E ENABLE RANDOM	R RANDOM VALUE	OUT USCITA ENABLE	OUT
0	0	1	
0	1	1	
1	0	0	
1	1	1	

Valore casuale per on-off random

Funzione OUT = $\bar{E}\bar{R} + \bar{E}R + ER$

Mappa di Karnaugh per la minimizzazione



Funzione minimizzata

$$OUT = \bar{E} + R$$

Lt Value + 0
Lt Value + 1
Lt Value + 2
Lt Value + 3
Lt Value + 4

OUTPUT
Lt Value (x)

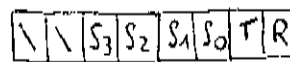
Intensità luminosa (0..255)

Lt Mode + 0
Lt Mode + 1
Lt Mode + 2
Lt Mode + 3
Lt Mode + 4

INPUT
Lt Mode (x)

Andamento + stati luminosità

Tremolo



Emble
0 = OFF
1 = ON

Effetto 0..15

Eseguito ad ogni passo fase

Genera il valore di luminosità corrispondente all'effetto selezionato

MLV.1

Count1 = 0

FSR = Lt Mode + Count1

INDF, Emble bit 0
1
0

VALUE = INDF
RRF VALUE, F
RRF VALUE, F
ANDLW B'00001111
MOVWF VALUE

VALUE = 0

NO

VALUE = 1

NO

VALUE = 2

NO

VALUE = 3

NO

Carry bit 0

0

1

Carry bit 0

0

1

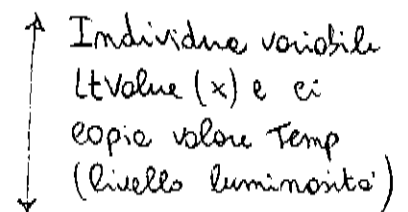
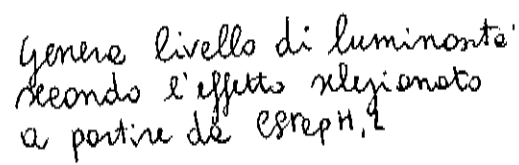
Se luce non abilitata, lampade spente

Entrata da Lt Mode
l'effetto selezionato
(bit 2 ÷ 5)

Verifica l'effetto selezionato
0..15

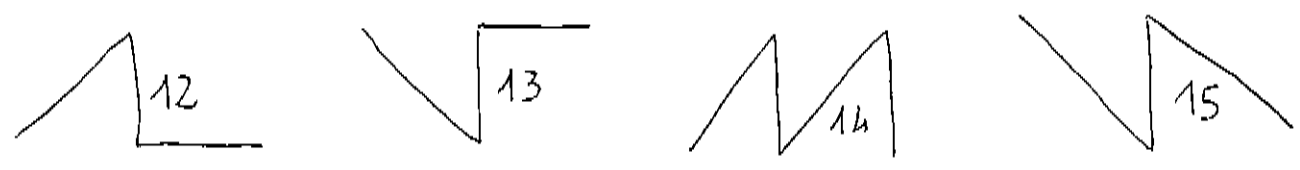
MLV.2

MLV.3



fromme variable

EFFETTI LUMINOSI generati da TAKE-LIGHTS, VALVE



Enigmas ad ogni passo fare

DRIVE
RELE

Pila 5 relè

Possibilità
0, 102, 204

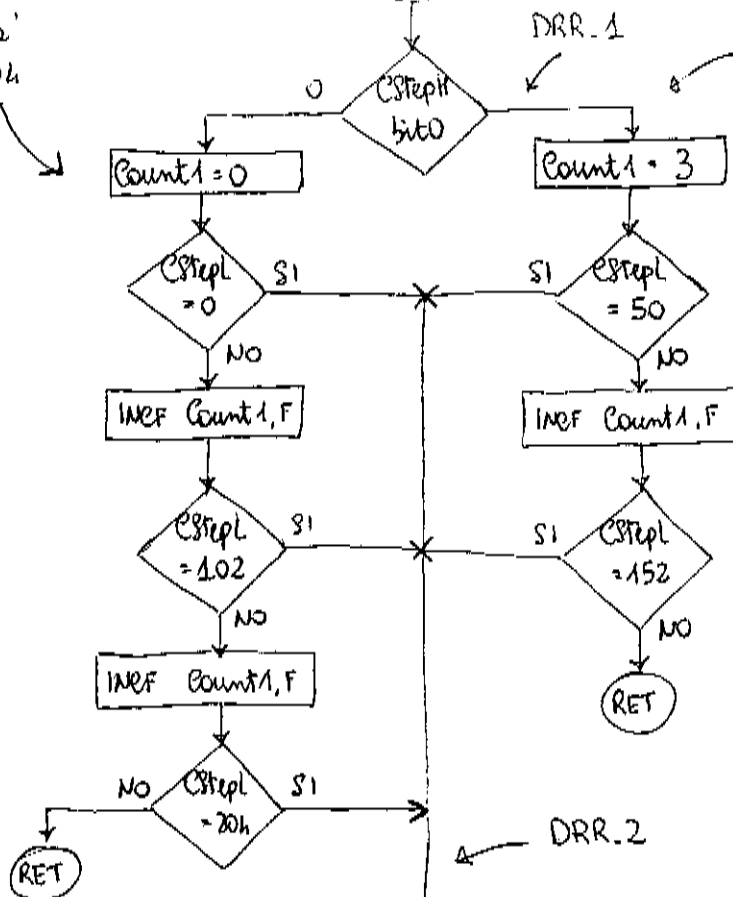
DRR-1

Possibilità 306, 408

Verifica
se raggiunto
posto
relè
(5 passi)

306 = 0000 0001 EStepH
0011 0010 EStepL

408 = 0000 0001 EStepH
1001 1000 EStepL



DRR-2

Count1 = Count1 + 28
RRF EStepH, w
ANDLW B'0000 0011
MOVWF Count2

DRR-4

Count2 = 0

DRR-3

legge
relè ON
da EEPROM
e memorizza
con EmRelè

Calcola (Conti)
l'indirizzo
EEPROM
dove leggere
lo stato
dei relè
(ad ogni
posto relè)

MOVWF Count1, w
CALL READ_EEPROM
MASCHERA CON
EmRelè
ANDWF EmRelè, w
MOVWF Value

Count1 = Count1 + 5
DECF Count2, F

SPEGNE 5 RELE

VALUE
bit 0

RELE 1 ON

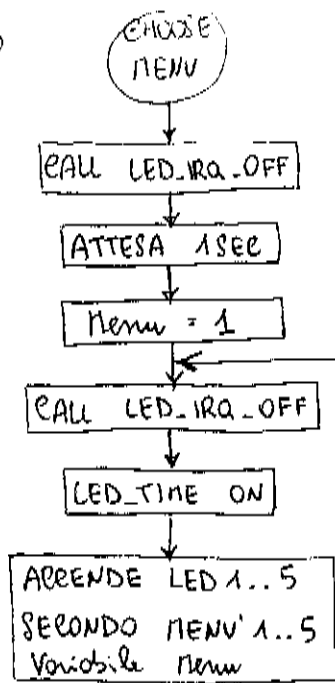
VALUE
bit 5

RELE 5 ON

Accende
o
Spegne
i
5 relè
secondo
il valore
EEPROM
e relativa
abilitazione

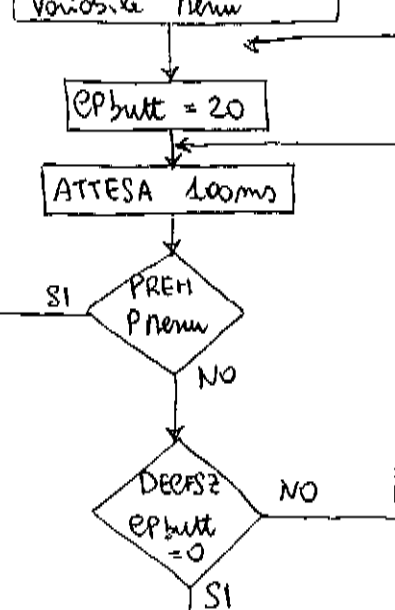
RET

Sceglie menù utente 1..5

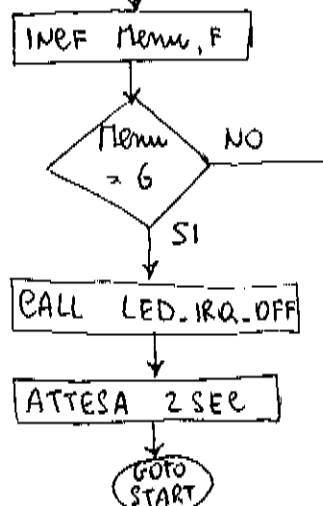
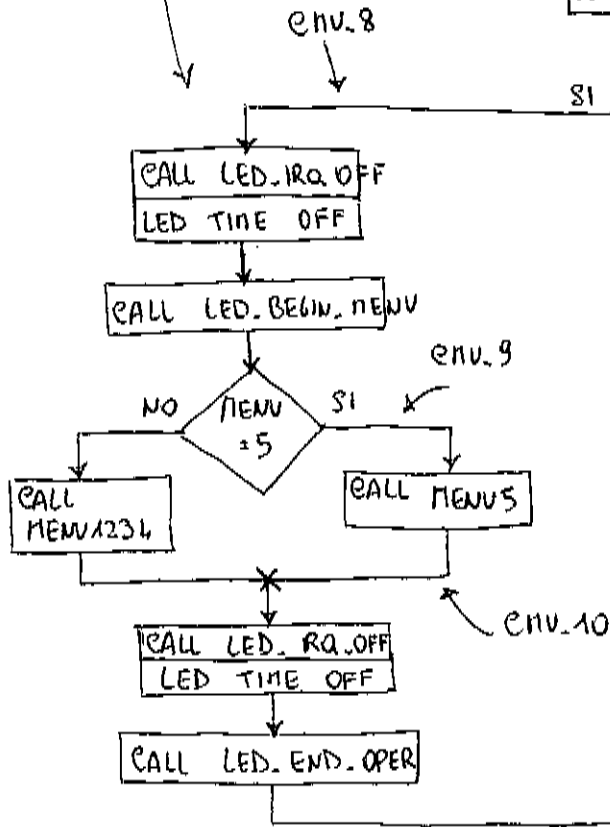


Indice scelta menù 1..5
Visualizza menù da scegliere

Richiamo menu 1234
(impostazione tempi fase)
o menu 5 (test write)

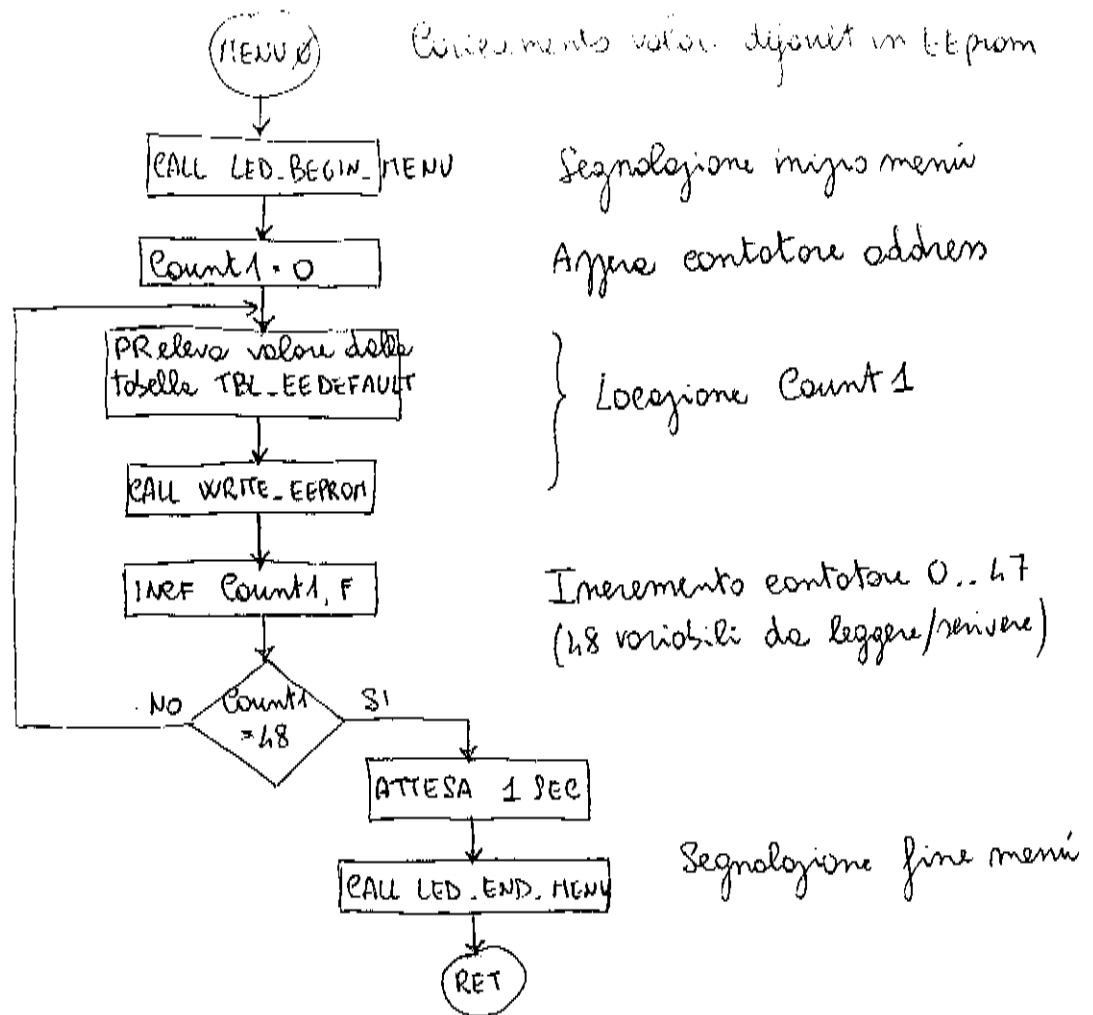


L'utente preme (e rilascia)
il pulsante quando è
stato raggiunto il menù
interessato entro 2 secondi

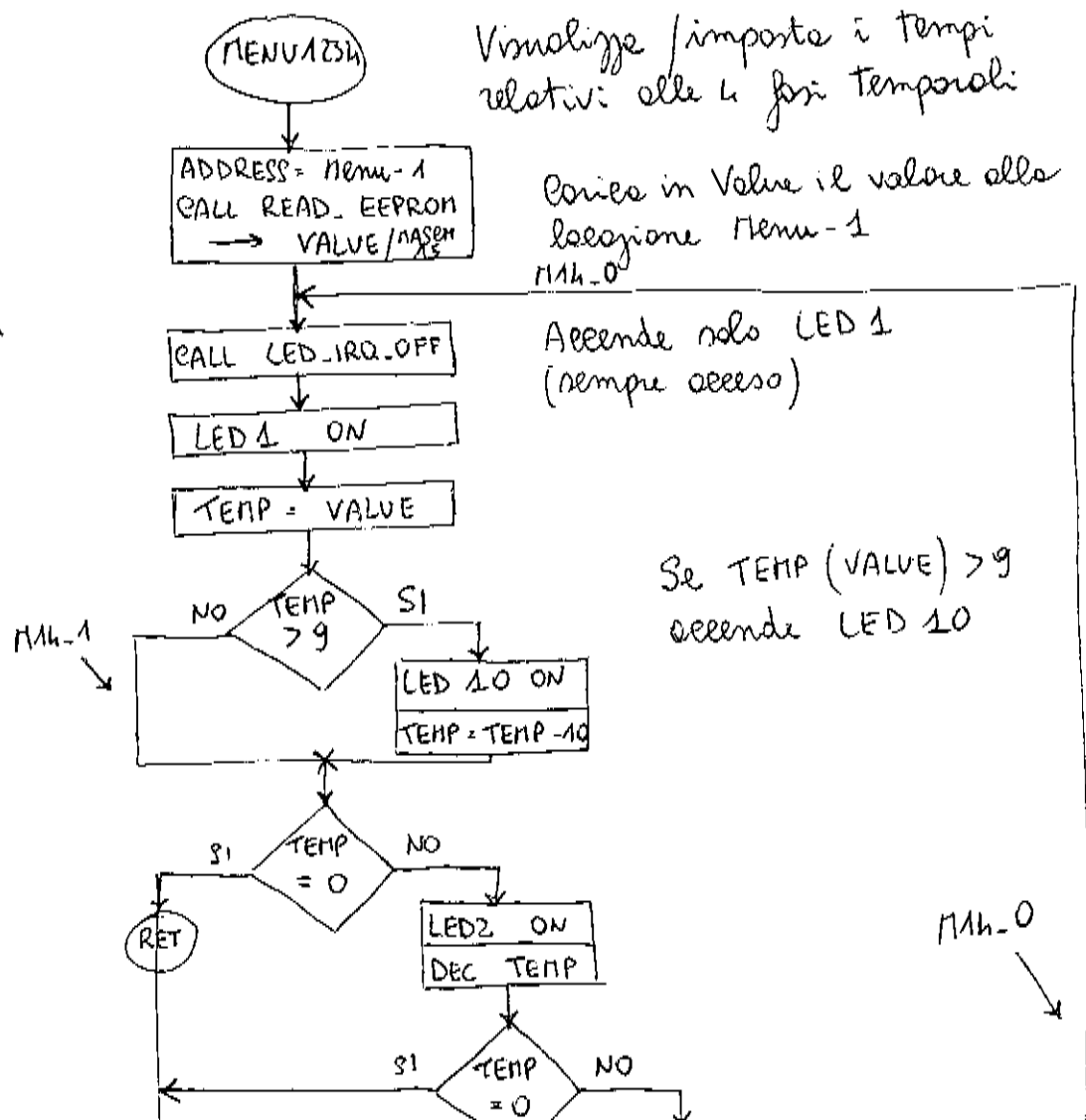


Incrementa
menù
fino al massimo

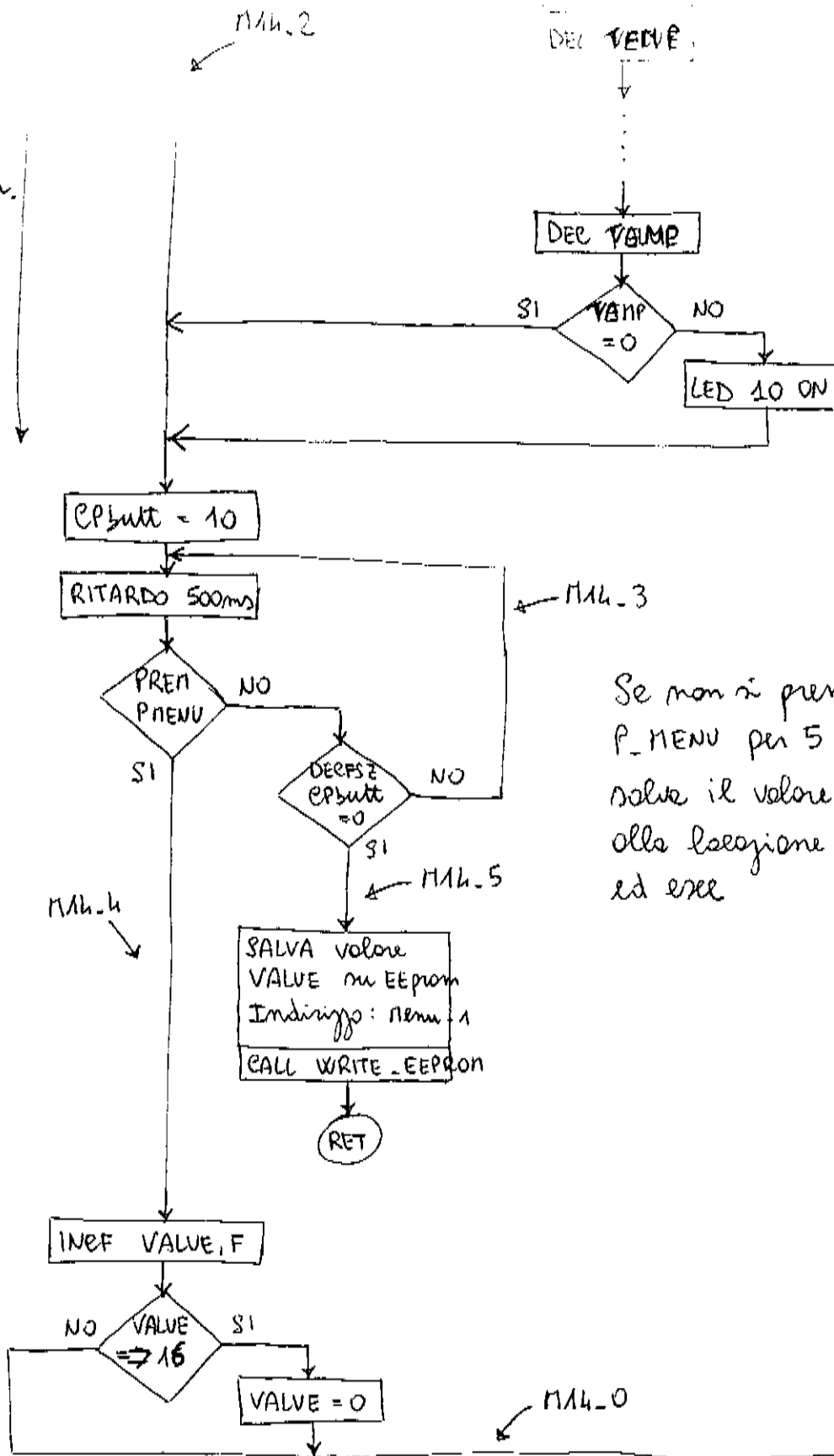
Gestione Menu 0



Gestione Menu 1..4



La sign valve
 corrisponde un
 tempo "forse"
 variabile da
 10 secondi a 20 min.

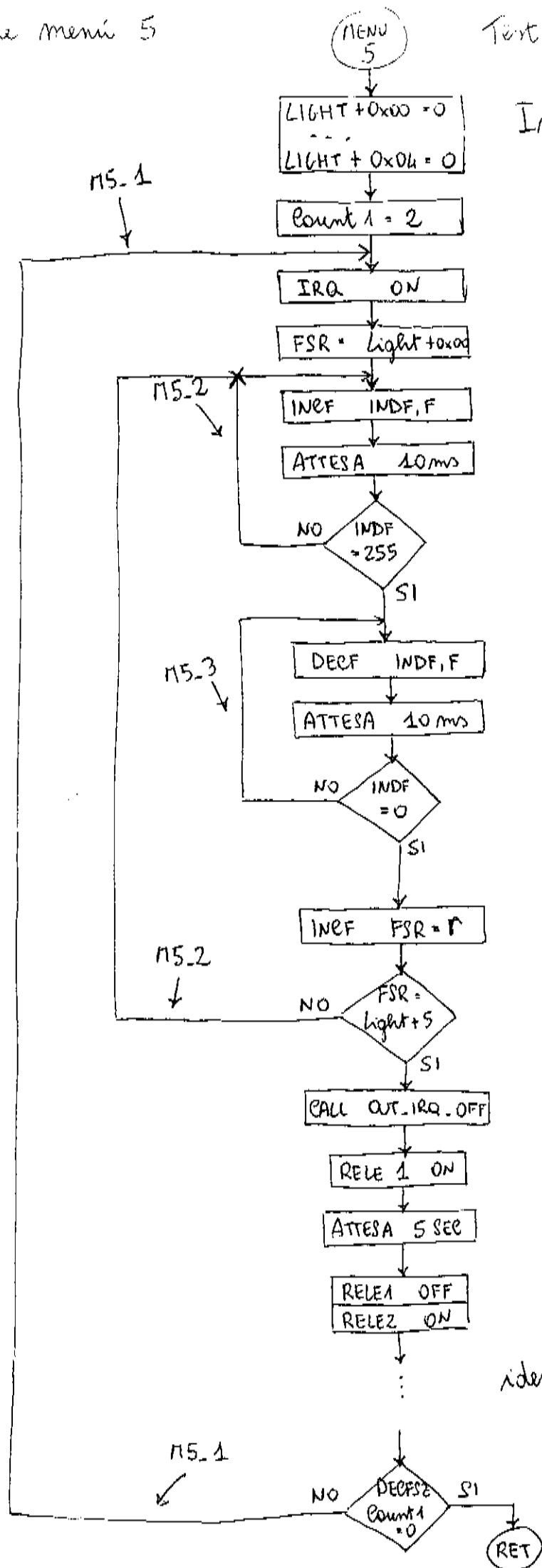


Se non si preme
 P_MENU per 5 secondi,
 salva il valore su EEPROM
 alla locazione Menu-1
 ed esce

Se premuto PMENU
 incrementa VALVE
 (fino a 15)
 e aggiorna la
 visualizzazione
 del led

Gestione menu 5

Test write 5 lampade + 5 rele'



Inizializzo valori luminante

Active interrupt

Variabile di inizio

TEST 5 LAMPAD

Incrementa luminante

0... 255

Tempo = 10ms * 255 = 2,5s

Decrementa luminante

255... 0

Tempo = 10ms * 255 = 2,5sec

Incrementa puntatore
fino al massimo

TEST 5 RELE

Active un solo rele' alla volta
per un tempo di 5 secondi

idem per Rele2.. Rele5

Routine comunicazione

MAKE
CONN

CALL OUT_IRQ.OFF

USCITA TX=1

Contatore indirizzo Eeprom

Count1 = 0

Cnt Del 3 = 6
Cnt Del 2 = 0
Cnt Del 1 = 0

NE_0

NE_1

NE_2

NE_3

fronte
bit di start

INPUT
RX

RICEZIONE CARAT
TERE 9600bps
→ RegTxRx (RxChar)

Loop di
scrittura in corso?

Count1 <> 0

SI

DECS2

Cnt Del 1 = 0

SI

DECS2

Cnt Del 2 = 0

SI

DECS2

Cnt Del 3 = 0

SI

Attese
per uscire
del loop
di scrittura
dopo 2 secondi
che non giungono
caratteri

ME_WRITE_LOOP

SI

Count1 <> 0

Loop di scrittura
in corso?

NO

CALCOLA OFFSET
SALVATAGGIO
MEMORIA EEPROM

Count1 - 1

SALVA RegTxRx
(carattere ricevuto)
in EEPROM
CALL WRITE_EEPROM

RILEGGE VALORE BYTE
SCRITTO IN EEPROM

RITRASMETTE BYTE
NEL FORMATO HEX
W x x <0zlg>

Loop scrittura
in EEPROM

Count1 = 48

SI

NO

INEF Count1, F

GOTO
NE_1

RxChar
= 'R'

SI

NO

ME_READ

TRASMETTE Loc. EEPROM in hex
FORMATO: 'R x x x <0zlg>'

GOTO
NE_1

lettura da
EEPROM
ogni carattere
byte R e L hex

RxChar
= 'W'

SI

NO

TRASMETTE
'W <0zlg>'

Count1 = 1

GOTO
NE_1

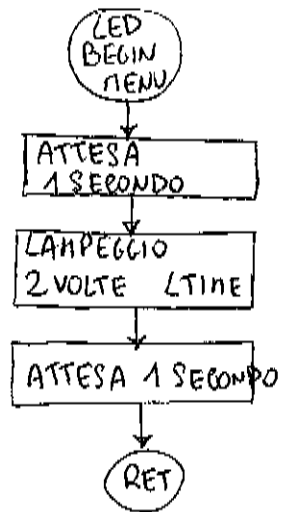
Start processo
scrittura
in EEPROM

TRASMETTE
'E <0zlg>'

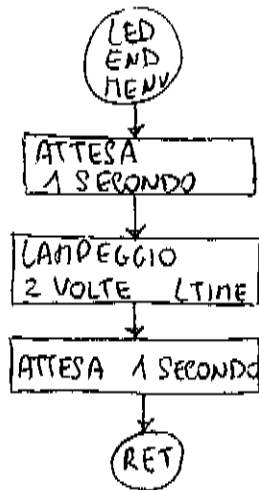
GOTO
NE_1

Comando
seconosciuto

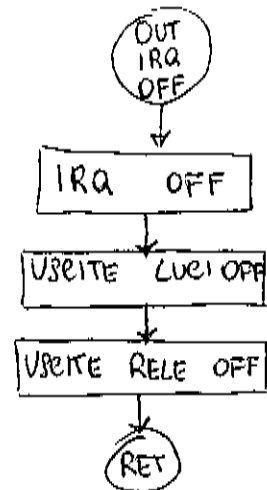
Segnalazione
Inizio menu



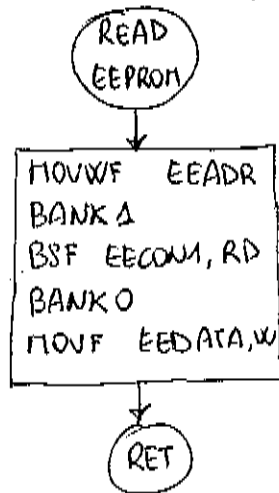
Segnalazione
Fine menu



Disattiva interrupt
fine la write e il logico



Letture da Eeprom



Input = W
indirizzo Eeprom

Scrittura su Eeprom

