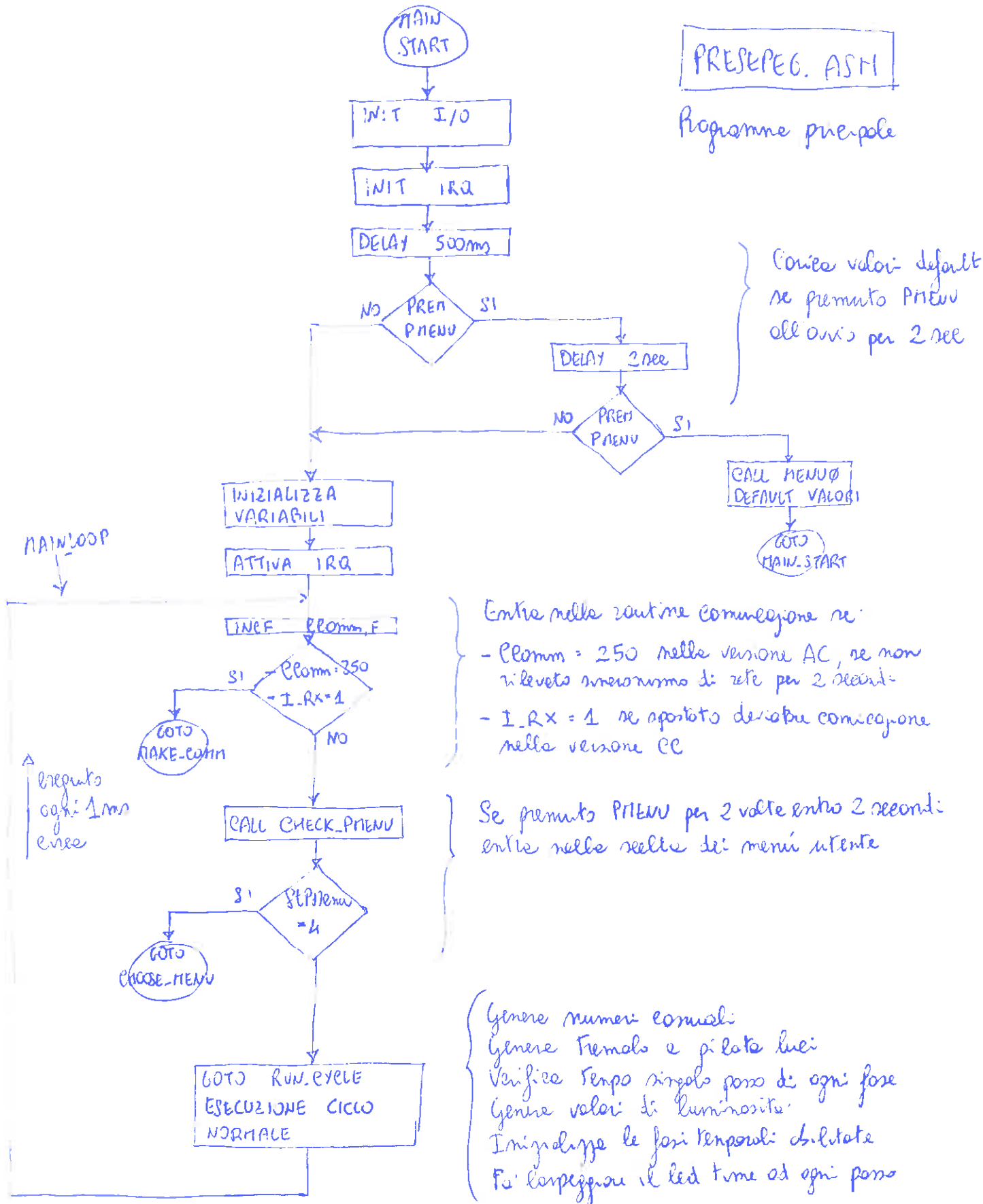


PRESEPEG. ASM

Programme pre-pole



Interrupt Timer 0

Eseguito ogni 25/54 μ s
genera i posti per il ritardo di
condizione del triac (Pwm)

Esce se il segnale
di zero-crossing
è alto (Rev.1)

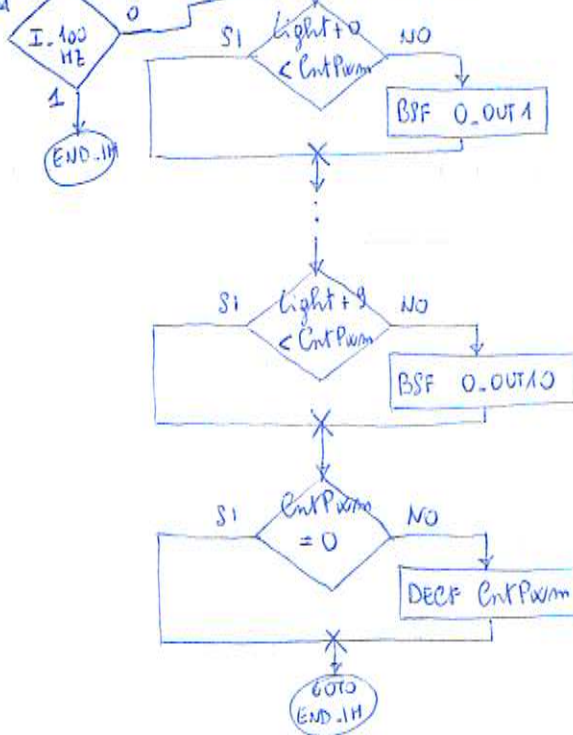
THR0 = 246
Esecuzione 25 μ s

THR0 = 238
Esecuzione 54 μ s

Stabilisce il tempo di esecuzione di
tim.int secondo il valore di EntPwm

Se Light(x) \neq EntPwm retta uscita = 1

PRESEPE6
VERSIONE
AC 50Hz

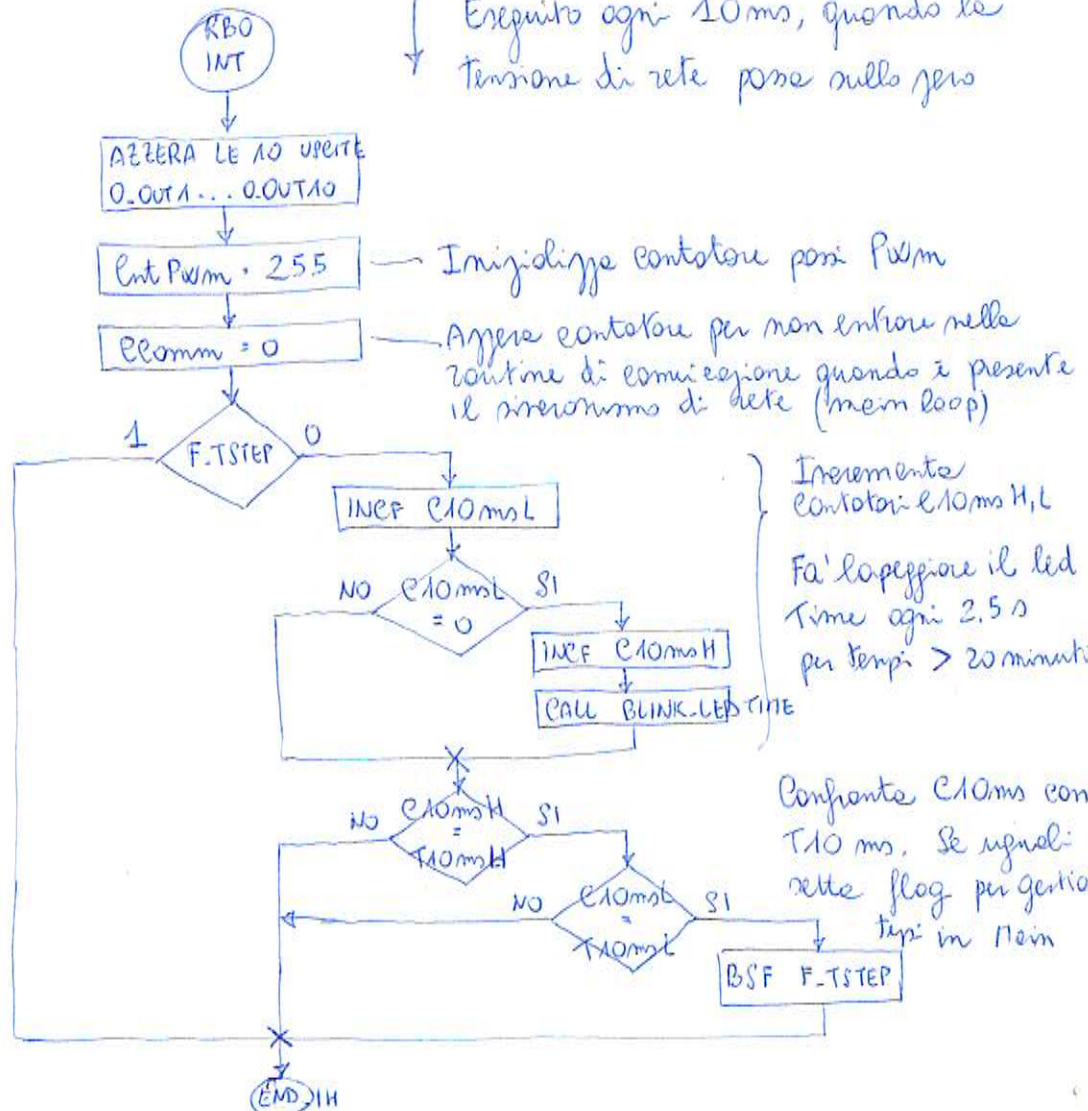


idem per uscite OUT2..OUT10

Decremento contatore posti Pwm
255... 0

Interrupt RB0

Eseguito ogni 10ms, quando la
tensione di rete passa sullo zero



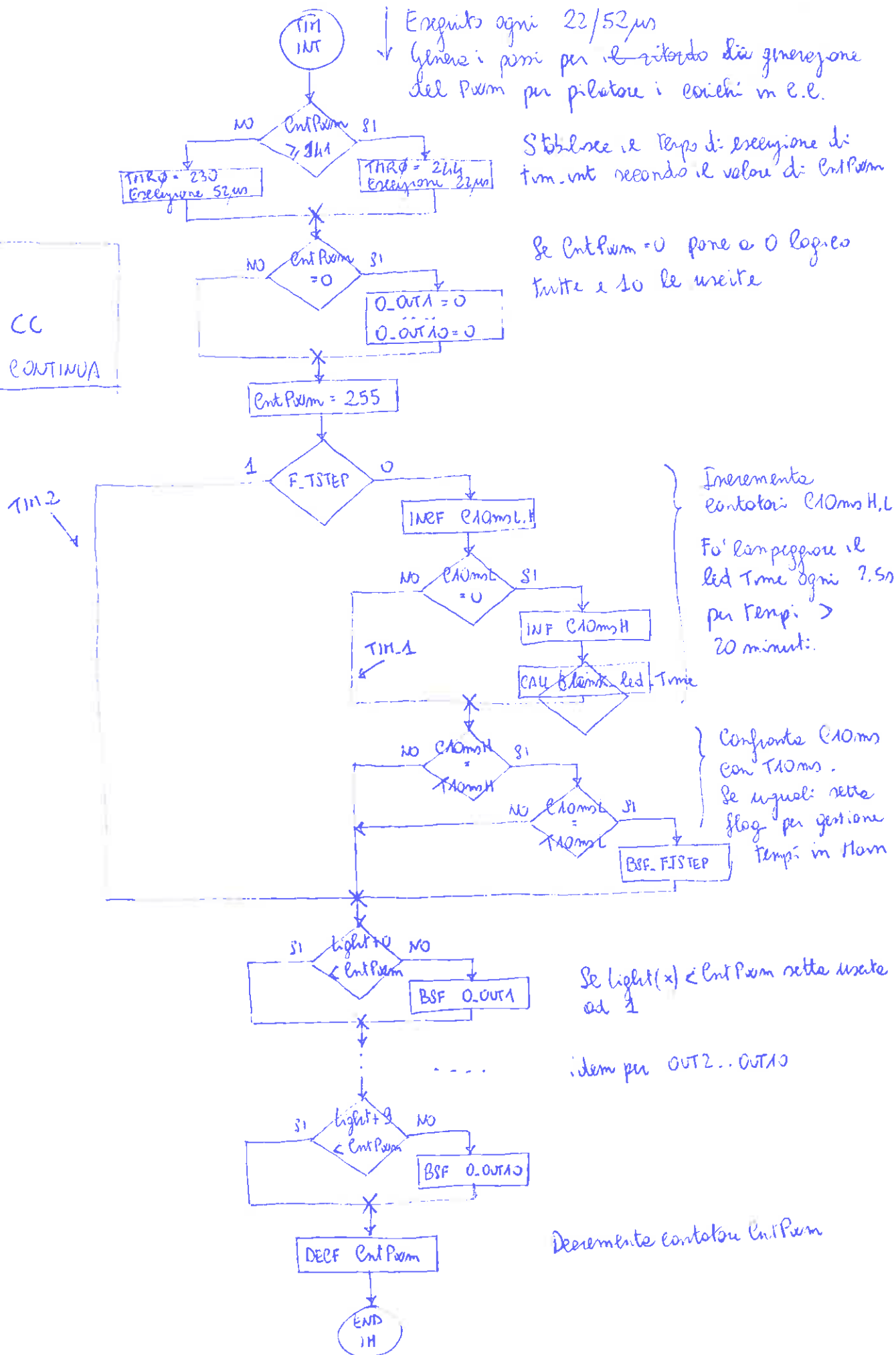
Inizializza contatore posti Pwm

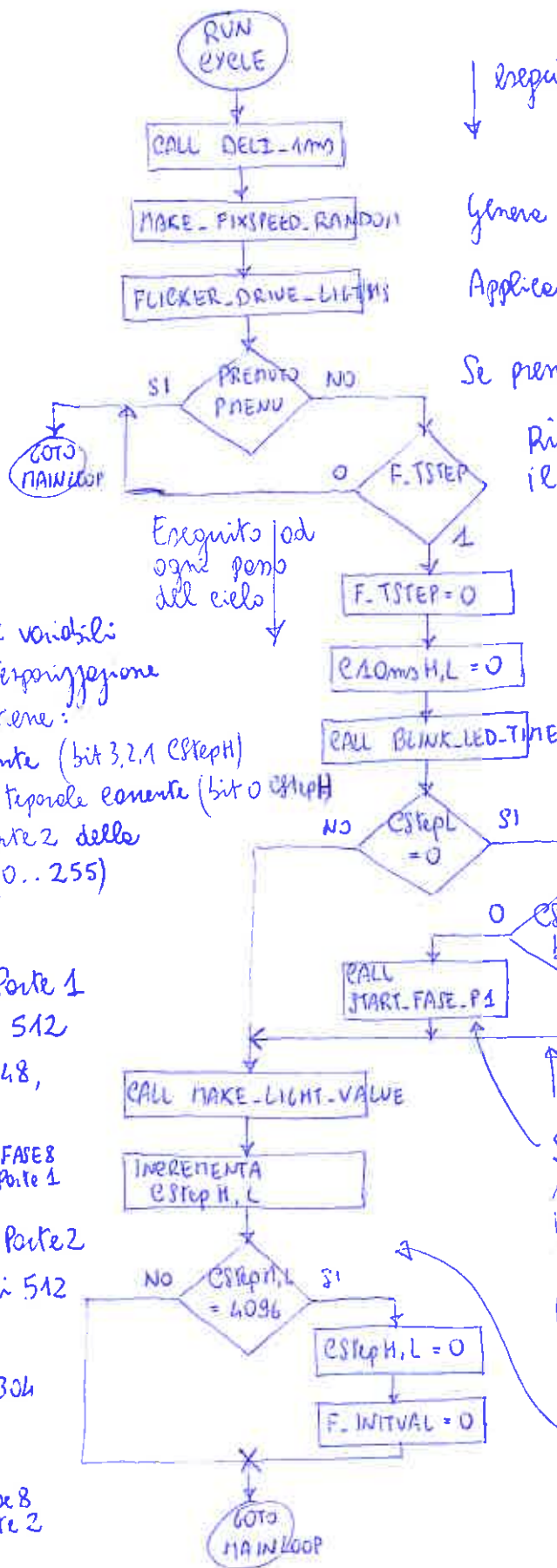
Aggiorna contatore per non entrare nelle
zone di commutazione quando è presente
il sincronismo di rete (main loop)

Incrementa
contatore 10ms H,L
Fa' lampeggiare il led
Time ogni 2.5 s
per tempi > 20 minuti

Confronta C10ms con
T10ms. Se uguali
setta flag per generare
tipi in Main

PRESEPEG
VERSIONE CC
CORRENTE CONTINUA





Esiguito ogni 1ms

Genera velocità fissa e casuale

Applica valori per fiamma e pilota uscita

Se premuto P1 o P2 per bloccare effetti, esce

Ritorna al main se non è fissato il tempo del singolo passo

Aggiorna flag e contatori del tempo del singolo passo

Fa' lampeggiare il led TIME

CSTEP H, L è la coppia di variabili per il conteggio della temporizzazione dell'intero ciclo. Contiene:

- la fase temporale corrente (bit 3,2,1 CSTEP H)
- la parte 1/2 della fase temporale corrente (bit 0 CSTEP H)
- il passo della parte 1/parte 2 della fase temporale corrente (0..255)

Conse importazioni Fase Parte 1
 di CSTEP H, L multipli di 512
 0, 512, 1024, 1536, 2048,
 2560, 3072, (3584)

FASE 1
Parte 1

FASE 8
Parte 1

Conse importazioni Fase Parte 2
 di CSTEP H, L multipli di 512
 a partire da 256

(256, 768, 1280, 1792, 2304,
 2816, 3328, (3840))

FASE 1
Parte 2

Fase 8
Parte 2

Se il tempo della fase temporale è uguale a 0, incrementa il bit FASETEMPO per prossime fase temporale (FASE = OFF) (incluso all'interno delle routine)

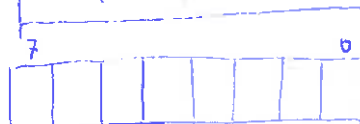
incrementa il passo dell'intero ciclo fino a 4096 (termina fase 8) → ritorna a 0 (inizio fase 1)

di = 1 (CSTEP H, L = 4096)
 indice che si è arrivati al termine del ciclo

Fase corrente 1..8 (0..7)

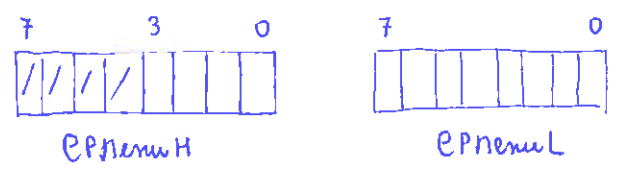
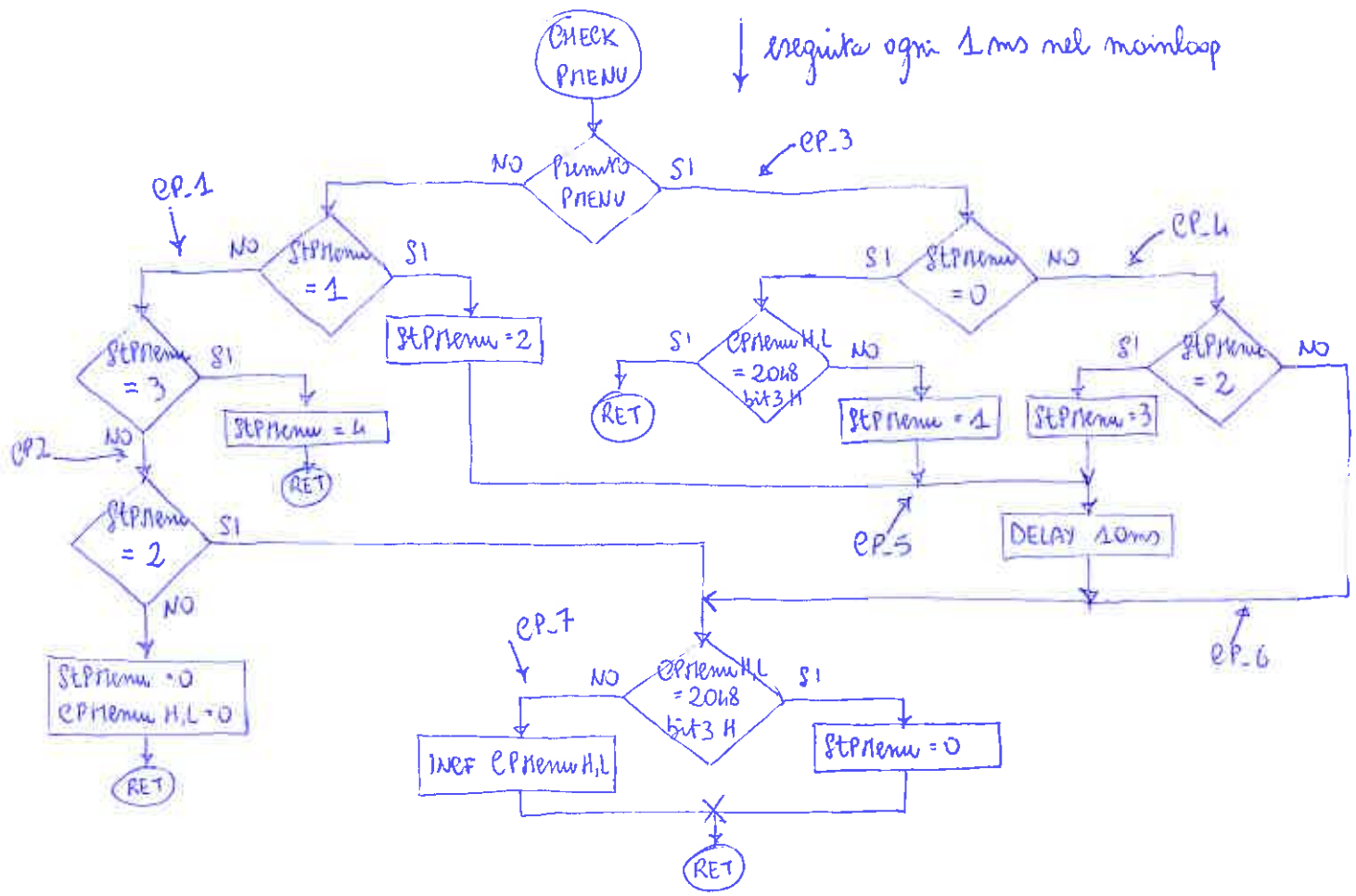
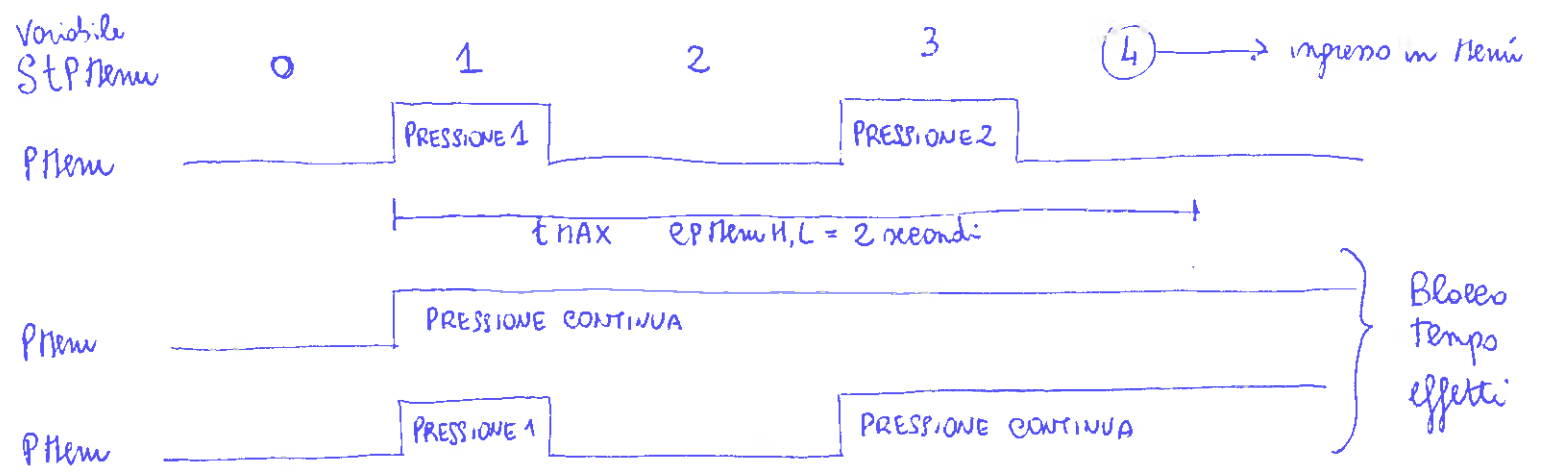
Fase Parte 1 - Parte 2 corrente (0 = Parte 1, 1 = Parte 2)
 Passo (0..255) della Parte 1/Parte 2 della fase temporale corrent.

CSTEP H, L



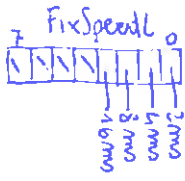
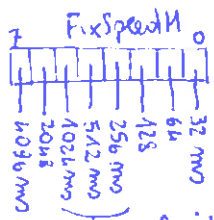
Gestione pulsante PMenu

- Pressione continua: blocco della temporizzazione degli effetti (pausa)
- Doppio click: ingresso in Menu utente. Il doppio click deve essere eseguito entro 2 secondi:



Contatore Timeout per doppio click (tMAX)

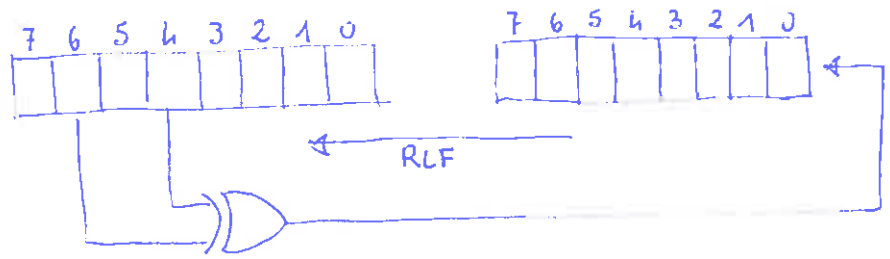
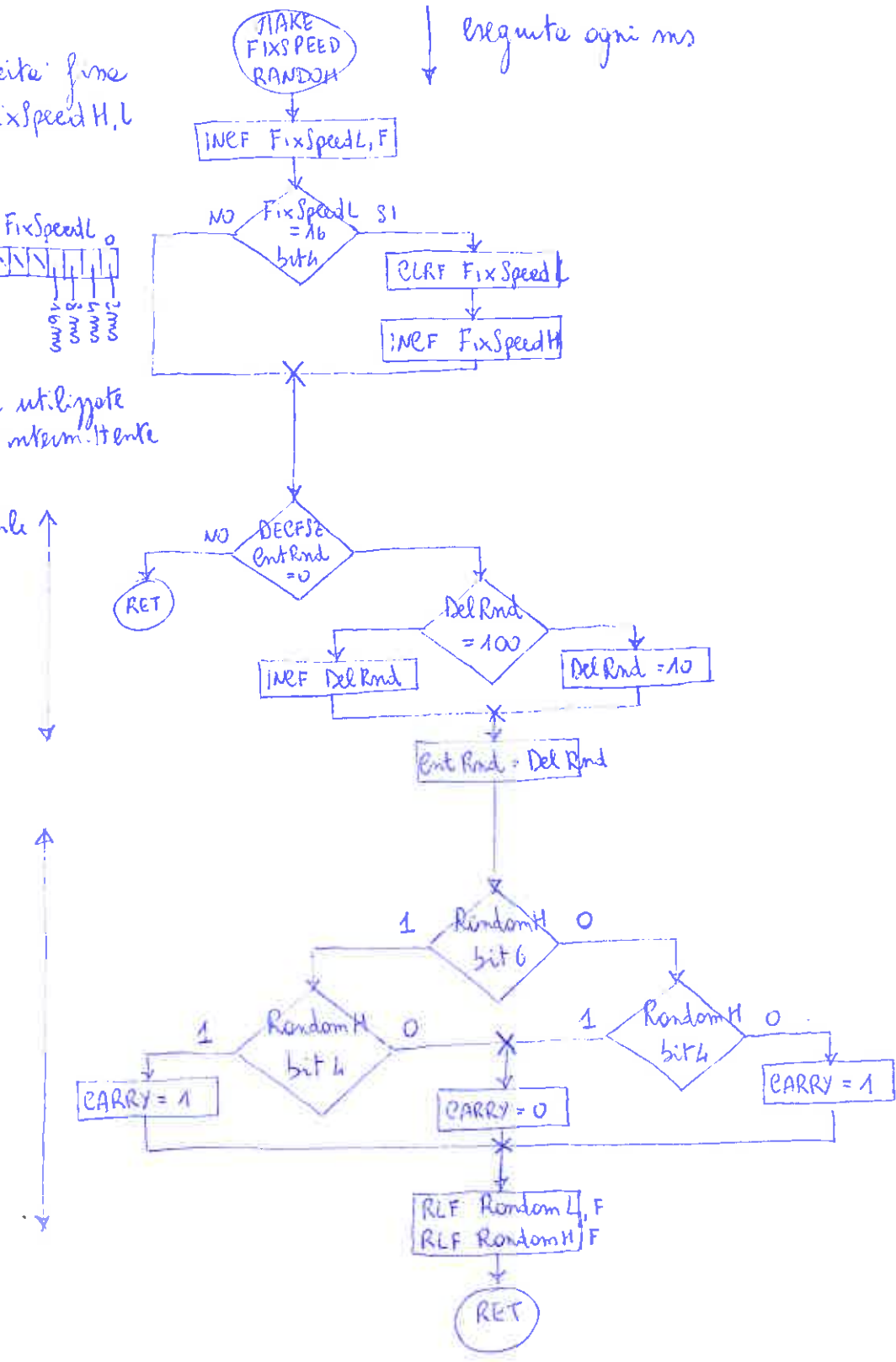
Genera una velocità fissa
nelle variabili FixSpeedH,L



velocità fissa utilizzate
per frenata intermittente

Ritardo variabile
da 10ms a
100ms

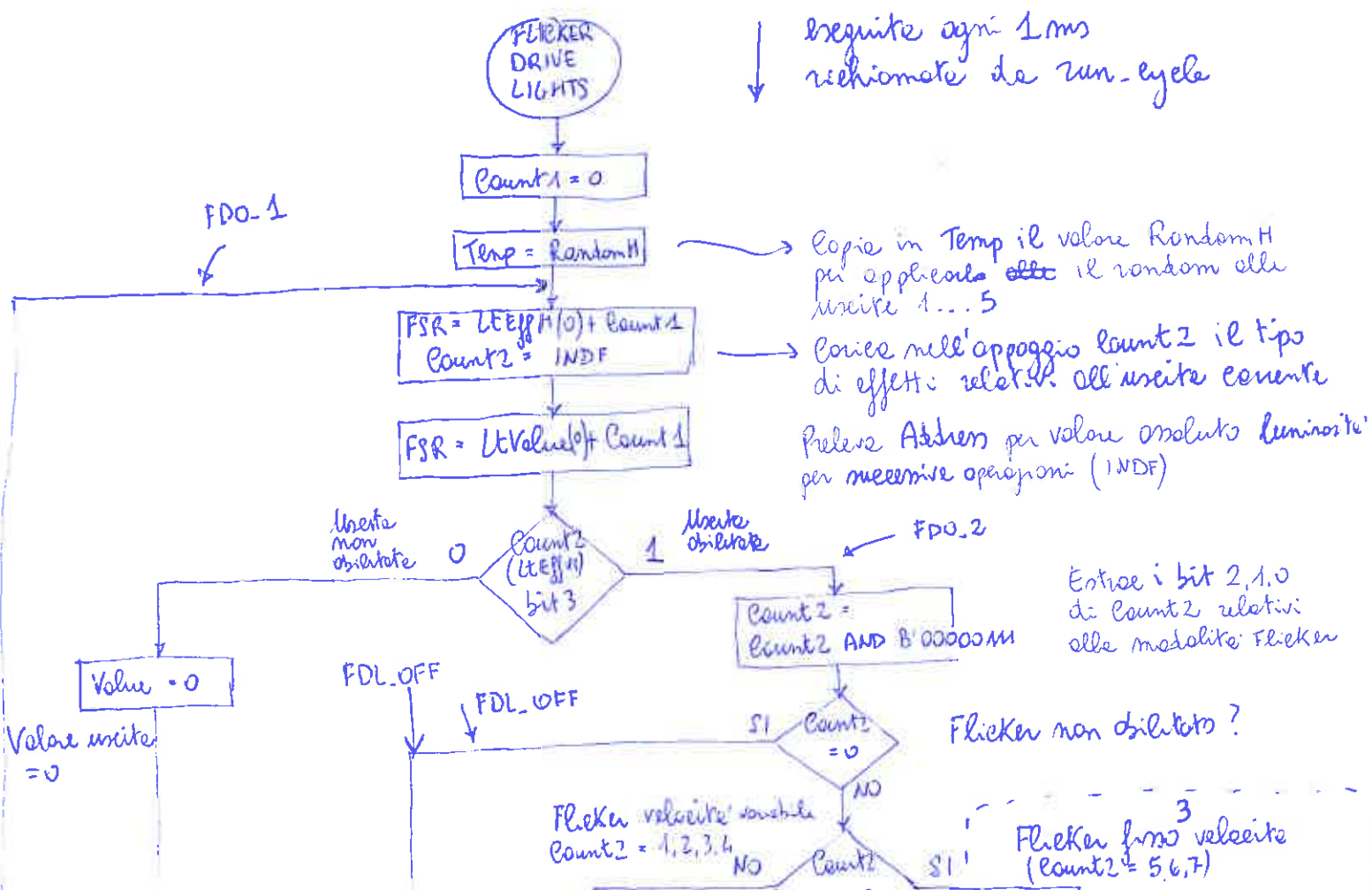
Logica LFSR
(Linear Feedback
Shift register)
per generare 16
bit casuali
a tempo variabile

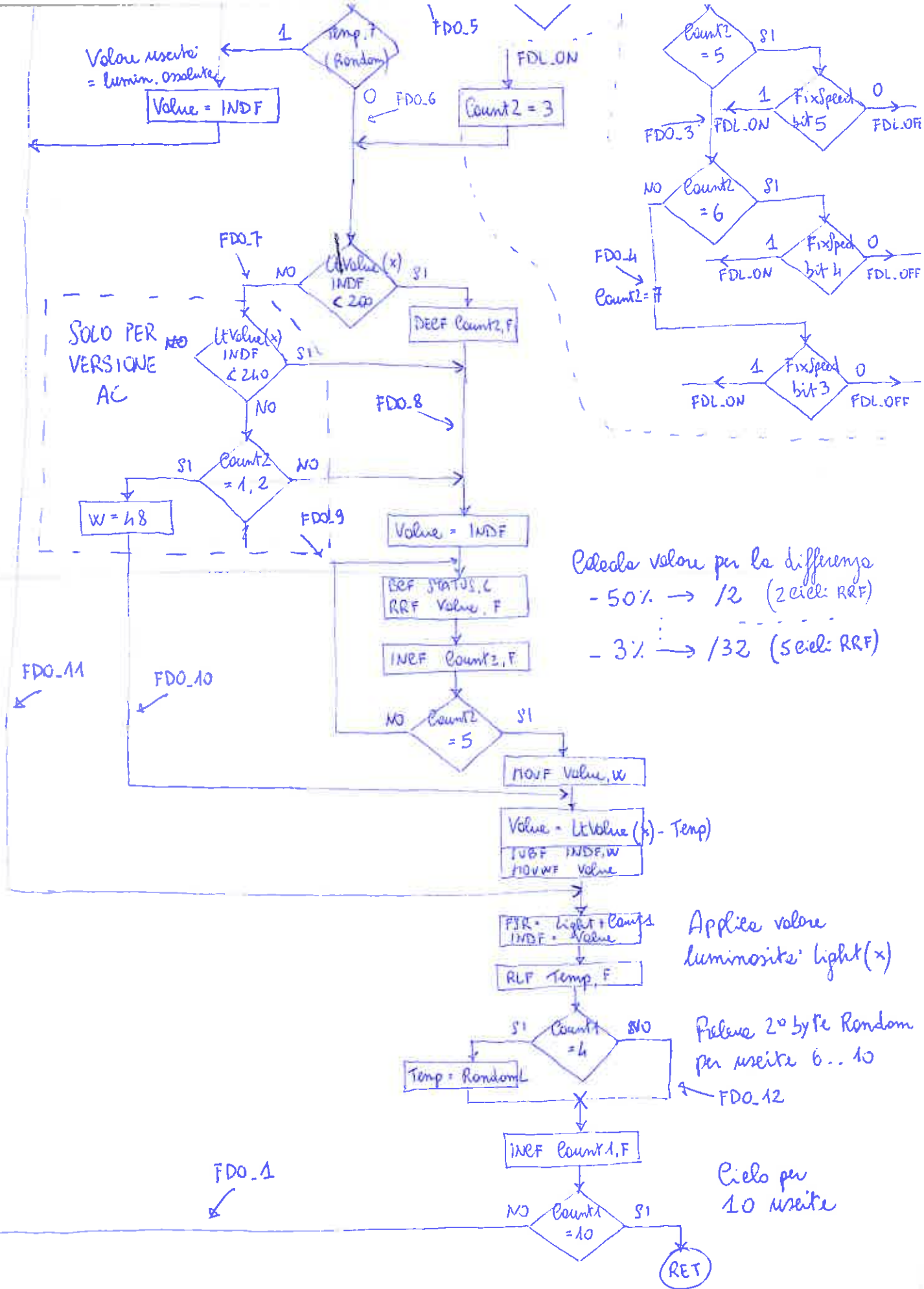


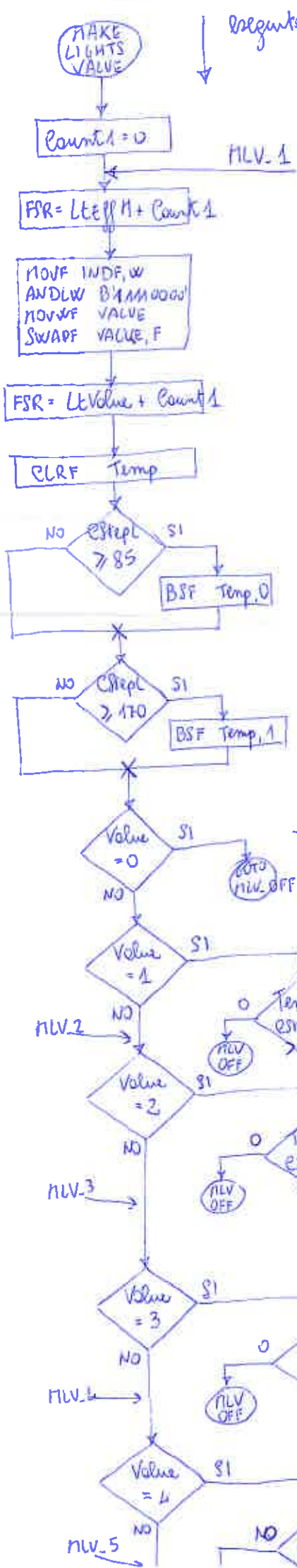
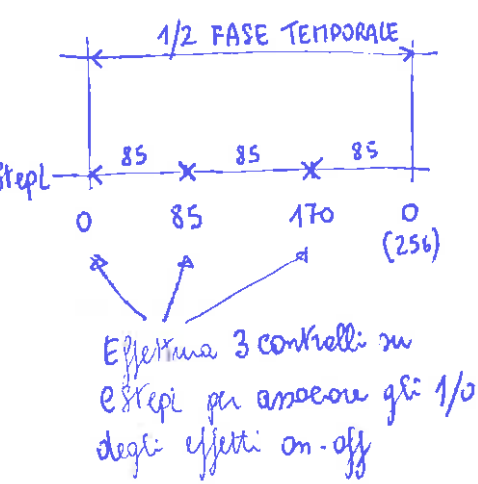
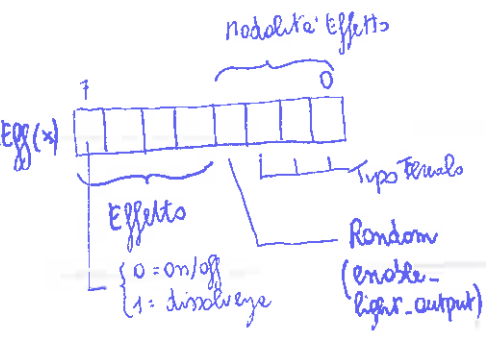
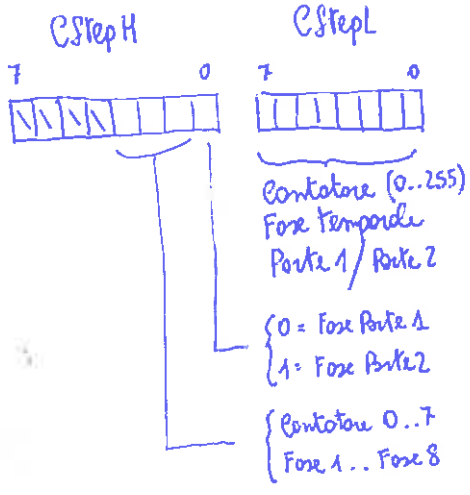
I due byte random sono utilizzati per generare il frenato delle stelle e per rendere casuale l'osil. Regione di ogni uscita all'inizio di ogni fase, a richiesta da utente

Generazione flicker (femolo)

Variabile LteffH() bit 2,1,0	Sottogruppo con livello di luminosità < 200 LtValue < 200	Sottogruppo con livello di luminosità > 200 LtValue > 200	Effetto Tremolo prodotto da:
0 (No Flicker)	0	0	Nessuno
1 Flicker luminosità -6% velocità variabile	-3% (5 cicli RRF)	-6% 4 cicli RRF -1.8 (valore fisso) se LtValue > 240	Condela
2 Flicker luminosità -12% velocità variabile	-6% (4 cicli RRF)	-12% 3 cicli RRF -1.8 (valore fisso) se LtValue > 240	Condela, Stelle
3 Flicker luminosità -25% velocità variabile	-12% (3 cicli RRF)	-25% (2 cicli RRF)	Stella, Fusco
4 luminosità -50% velocità variabile	-25% (2 cicli RRF)	-50% (1 ciclo RRF)	Fusco
5 luminosità -25% velocità fissa 1	-12% (3 cicli RRF)	-25% (2 cicli RRF)	Fusco intermittente
6 luminosità -25% velocità fissa 2	-12% (3 cicli RRF)	-25% (2 cicli RRF)	Fusco intermittente
7 luminosità -25% velocità fissa 3	-12% (3 cicli RRF)	-25% (2 cicli RRF)	Fusco intermittente







Aggiunta ad ogni passo di CStepH, L

Aggiunta variabile contatore per loop (inserta 10)

Correca l'effetto per ciascuna uscente nelle variabile value

Pone il puntatore nel valore luminosità di ogni uscente

Per comodità verificare se CStepL (contatore passi all'interno di ogni mezza fase temporale) è maggiore di due valori.

È utilizzato per il confronto negli effetti on-off, dove ogni 1/2 fase è suddivisa in 3 parti

Verificare Value per effetti ON-OFF (0...7)

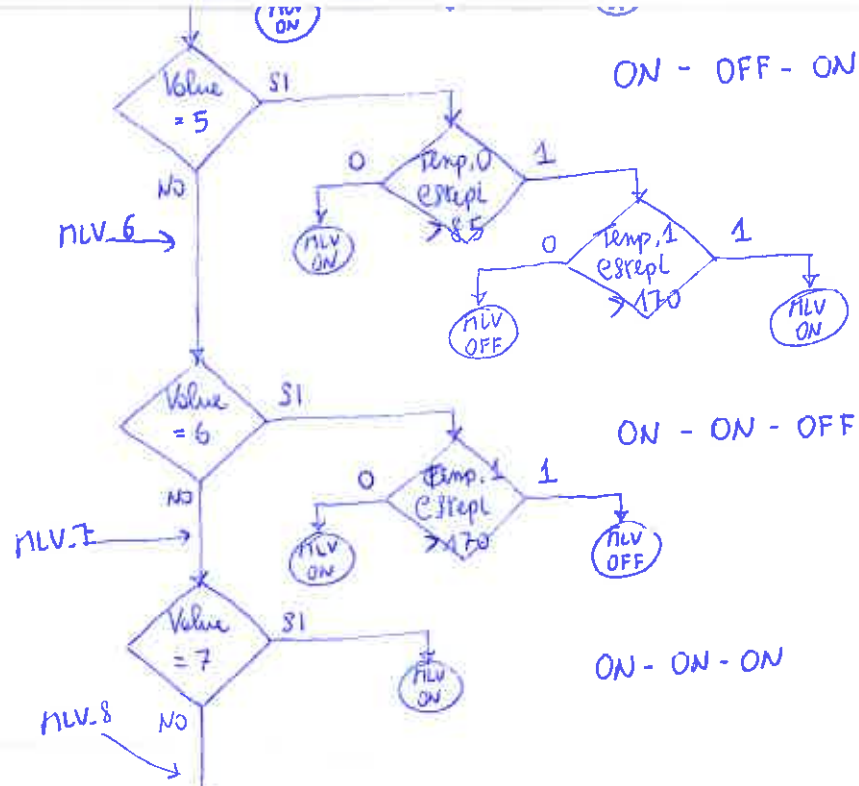
OFF - OFF - OFF

OFF - OFF - ON

OFF - ON - OFF

OFF - ON - ON

ON - OFF - OFF

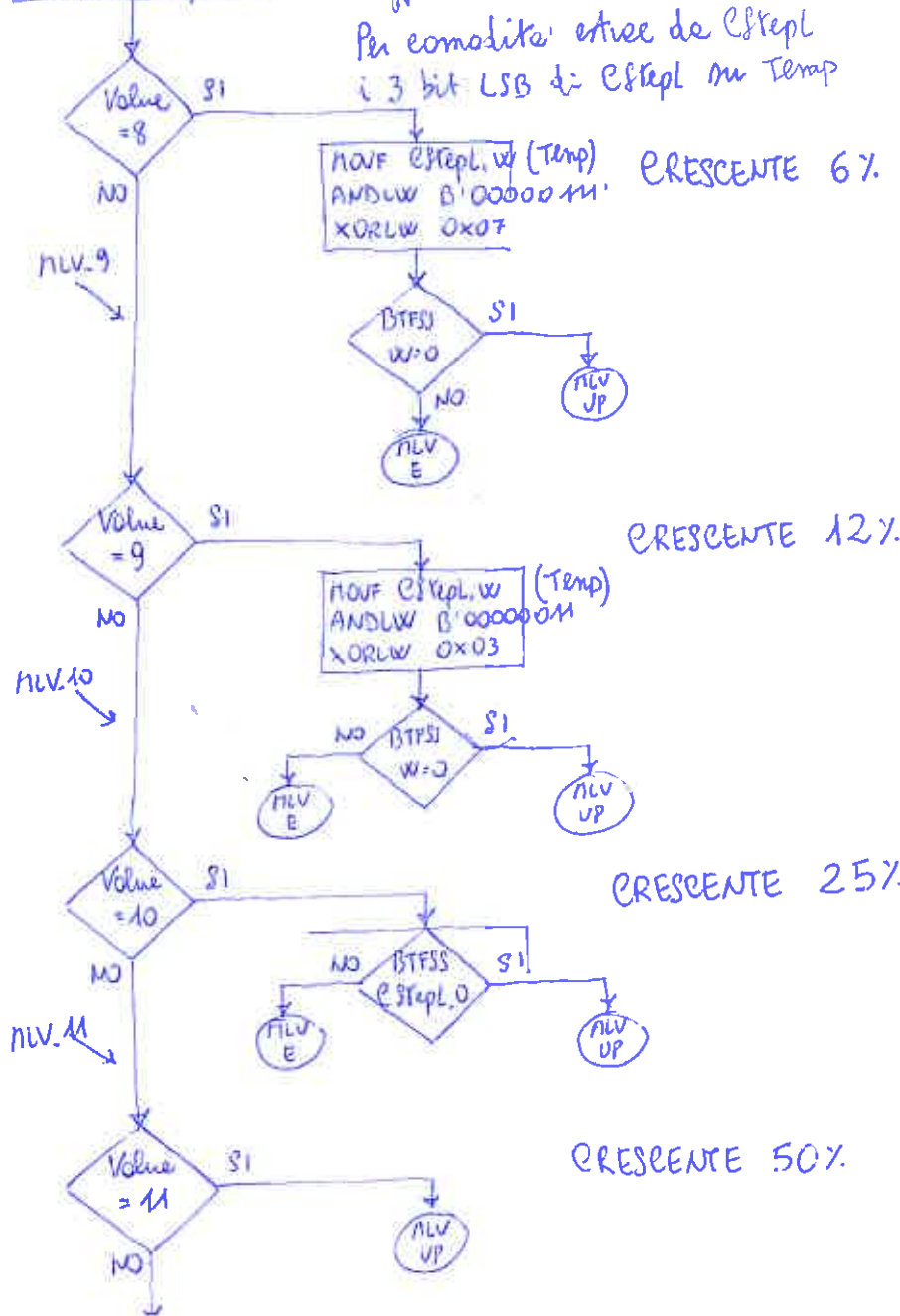


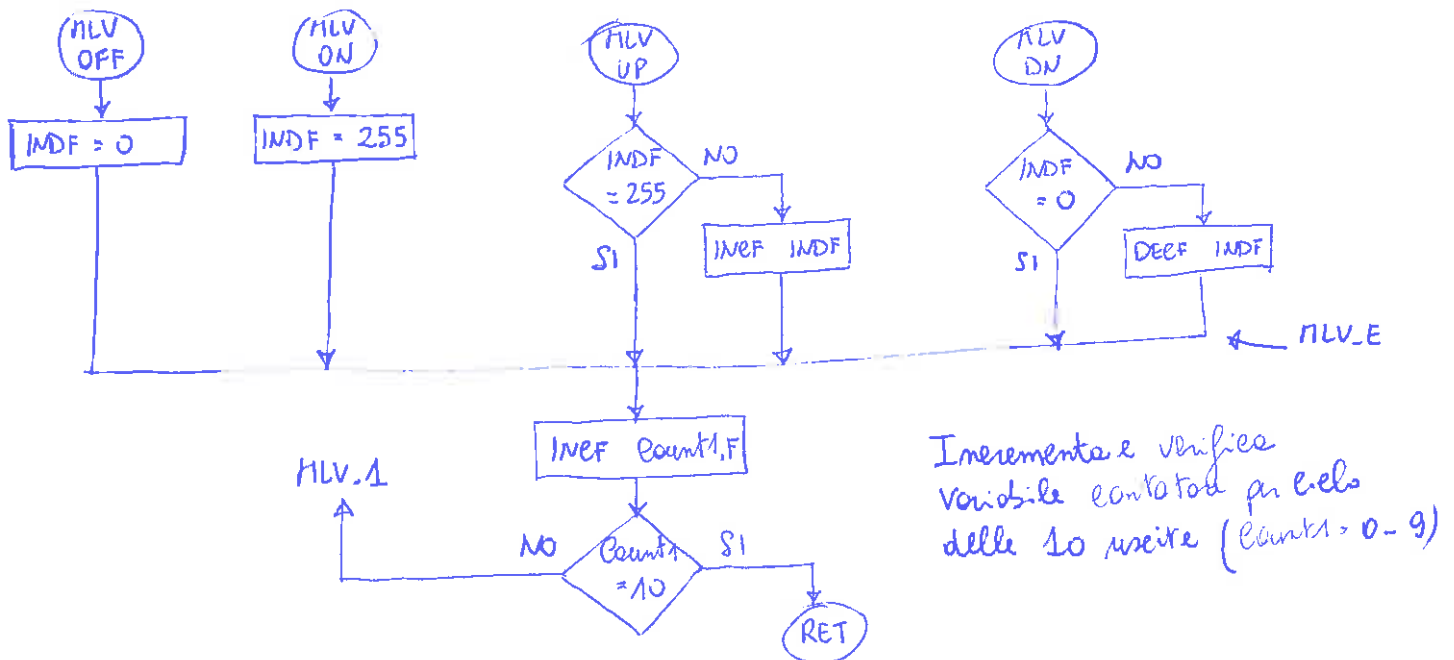
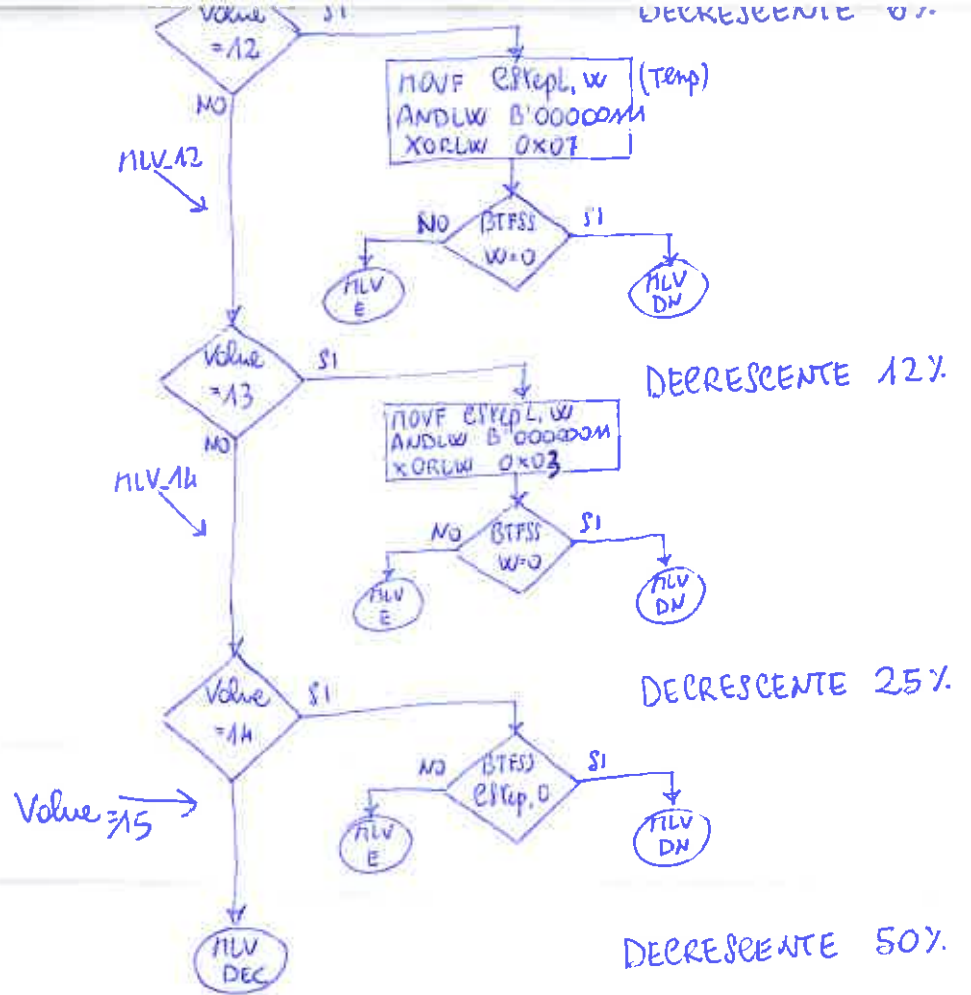
Verifica incremento/decremento luminosita'

- 50% ogni passo di Cstepl
- 25% ogni 2 passi di Cstepl bit 0 = 1
- 12% ogni 4 passi di Cstepl bit 0 e bit 1 = 1
- 6% ogni 8 passi di Cstepl bit 0, bit 1 e bit 2 = 1

Cstepl bit				2	1	0
50%				0	0	0
50%	25%			0	0	1
50%				0	1	0
50%	25%	12%		0	1	1
50%				1	0	0
50%	25%			1	0	1
50%				1	1	0
50%	25%	12%	6%	1	1	1

MOVF Cstepl, W
ANSLW B'00001111
MOVWF Temp





1

START_FASE-P1

START_FASE-P1

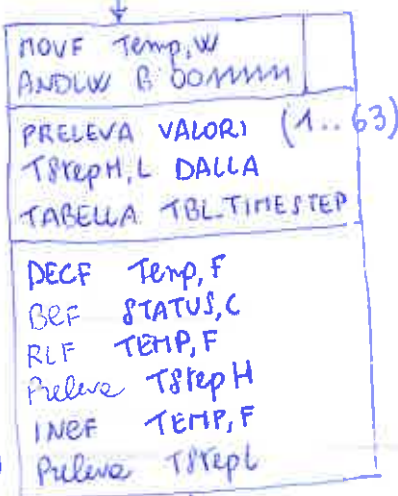
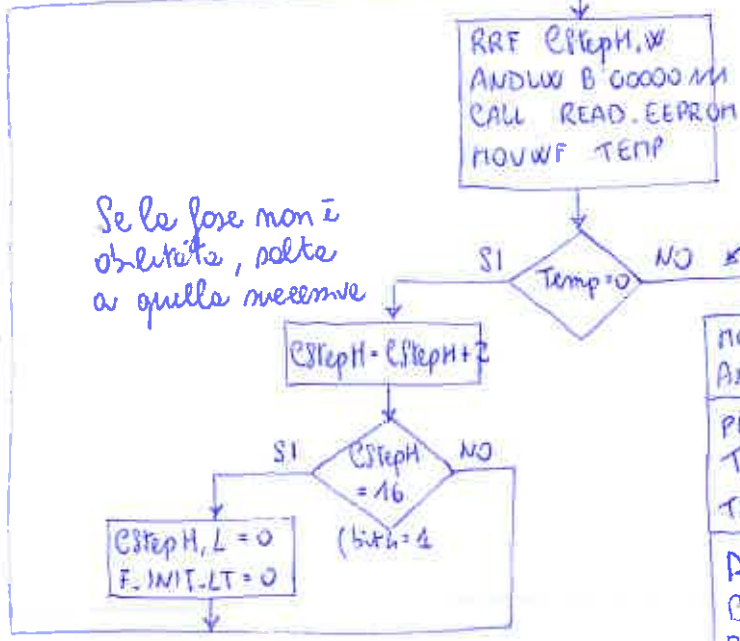
Eseguito all'inizio di ogni fase temporale (Ponte 1)

Entrate bit 1,2,3 di CStepH per prelevare la fase temporale corrente



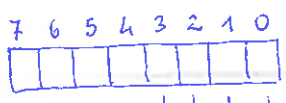
Fase Temporale

Se la fase non è obliata, salta a quella successiva



TStepH,L è il tempo del singolo passo. In base al valore prelevato su EEPROM (1..63) vengono estrapolati 63 valori diversi. I valori TStepH,L sono inseriti in sequenza.

LtEppH(x)



Effetto
0..7 = On-off
8..15 = dimming
Tremolo (0..7)
Random (0,1)

Eseguito ai metà di ogni fase temporale

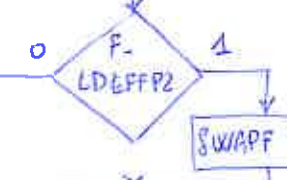
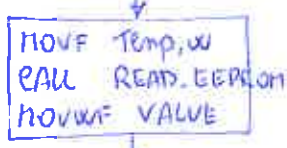
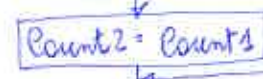
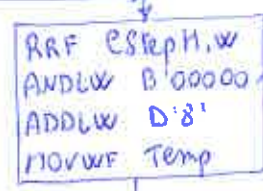
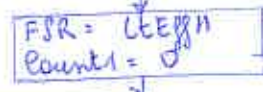
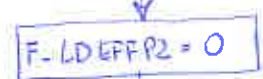
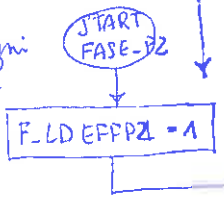
2

Conoscendo su LtEppH(x) bit 7,6,5,4 gli effetti impostati da utenti relativi alle 16 uscite

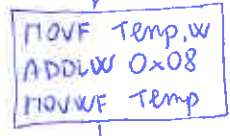
Indirizzo EEprom:
8 + Fase + Count * 8

Fase(4)	1	2	3	4	5	6	Fase(8)
Info (1)	8	9	10	11	12	13	14
1	16						
2	24						
3	32						
4	40						
5	48						
6	56						
Count (8)	64						
8	72						

START_FASE-P2

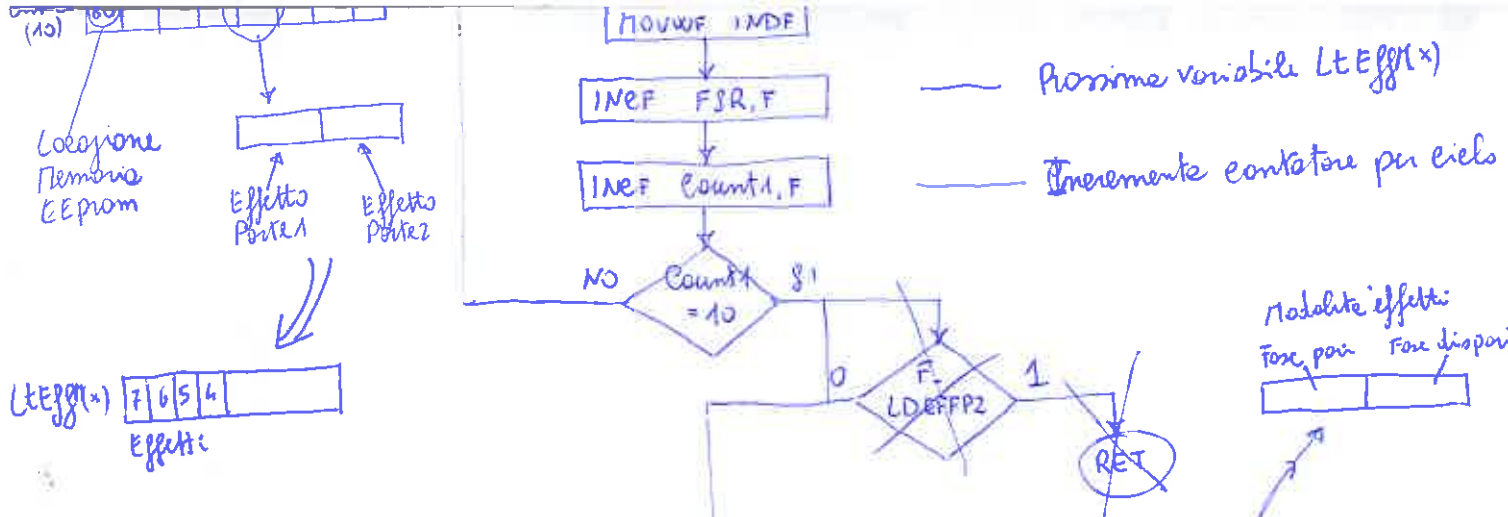


estive fase temporale



Calcola indirizzo eeprom

*



3 Come su LteffH(x) bit 3,2,1,0 le modalita' effetti

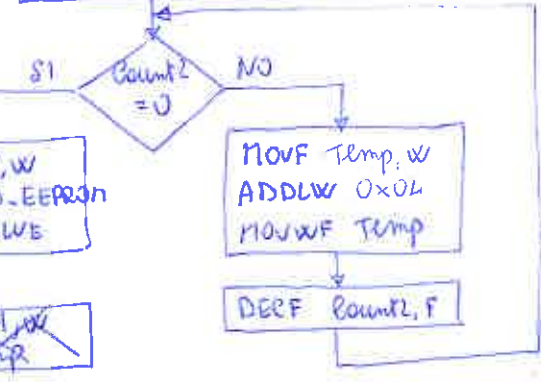
Indirizzo EEPROM = $88 + \text{Fose} / 2 + \text{Out} * 4$

0..7 0..9

```

MOVWF EFPep.w
MOVWF Temp
RRF Temp, F
RRF Temp, w
ANDLW B'00000011
ADDLW D'88
MOVWF TEMP
  
```

Count2 = Count1



Modalita' effetti: Fose pari Fose dispari

	0	1	2	3
Out(0)	88	89	90	91
1	92			
2	96			
3	100			
4	104			
5	108			
6	112			
7	116			
8	120			
9	124			

Locazione EEPROM

SF-6

SF-7

Out(9)

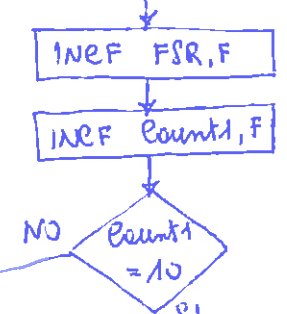
Fose 0 1 2 3 4 5 6 7 8

Calcola indirizzo EEPROM

Se fose pari (0,2,4,6) carica in LteffH bit 3,2,1,0 i 4 bit LSB di Value (eeprom)

Se fose dispari (1,3,5,7) carica in LteffH bit 3,2,1,0 i 4 bit MSB di Value (eeprom)

→ INDF = LteffH(0..9) puntatore



Proxima variabile LteffH(x)

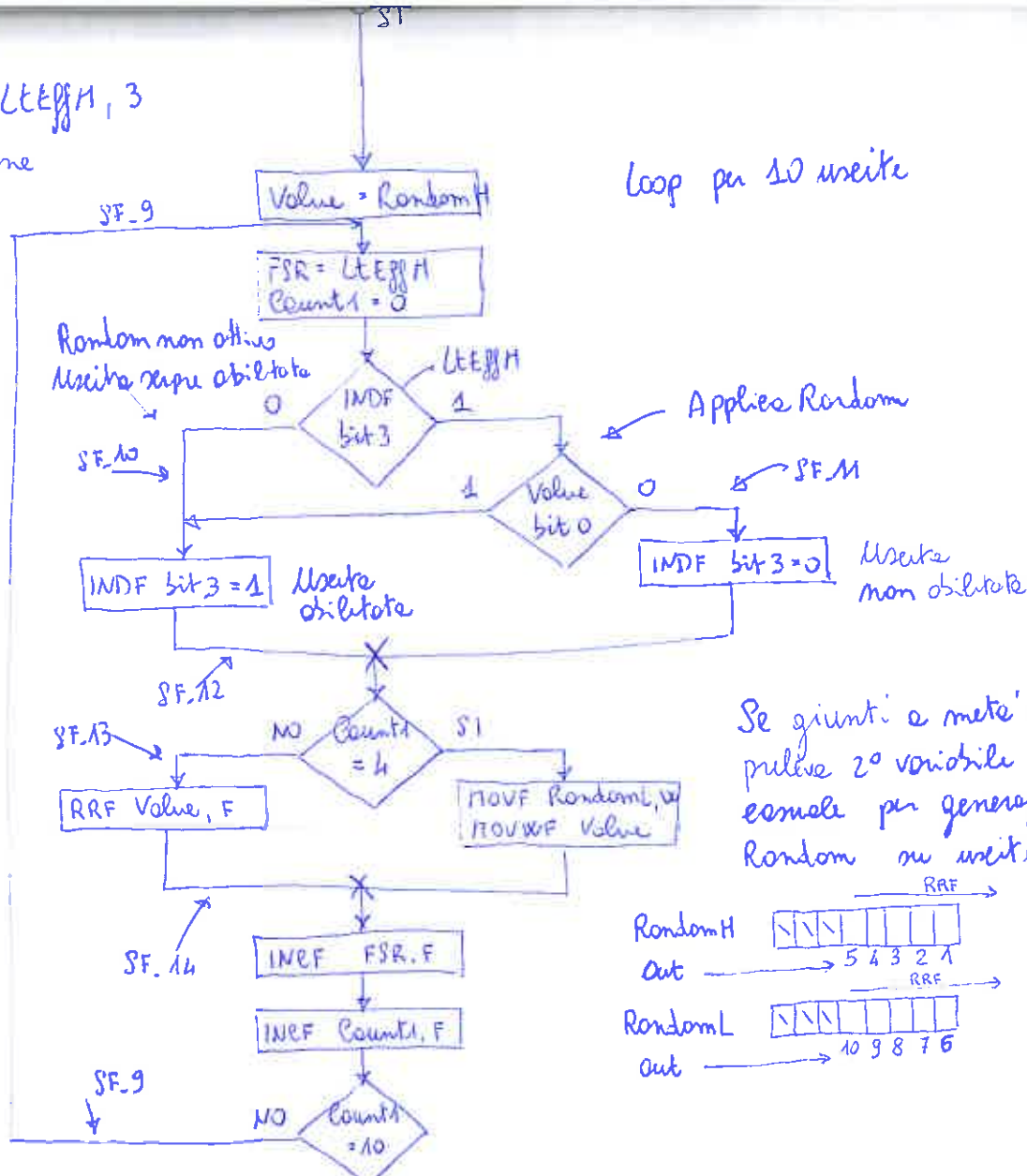
Incremento contatore per ciclo

④

Trasforma il bit LEEFFH, 3
secondo l'importazione
Random on/off

Se Random attivo,
abilita l'uscita in
modo casuale

Se Random non attivo
uscita sempre abilitata



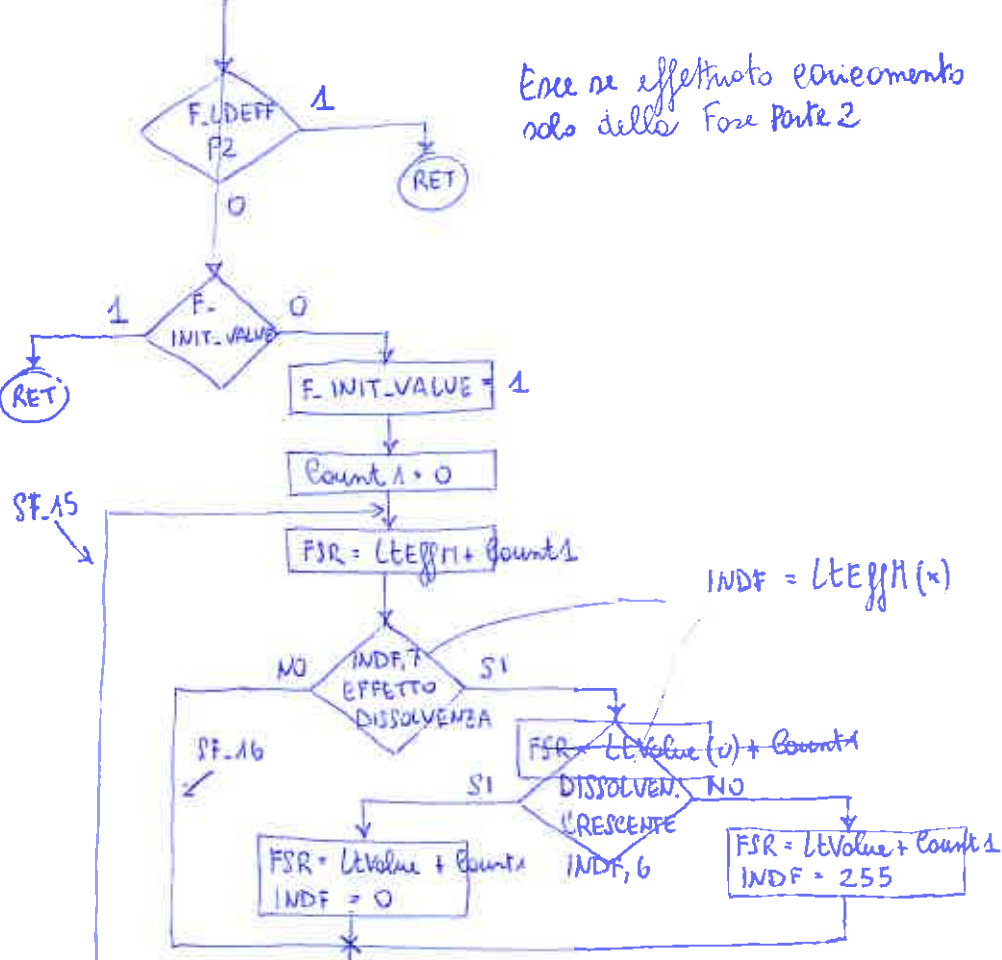
⑤

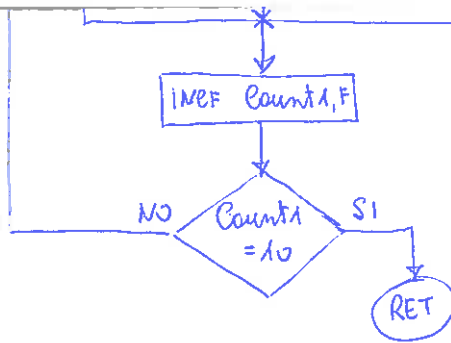
Se la forza temporale corrente è
la prima abilitata dell'intero
ciclo, inizializza i valori per
gli effetti in dissolvenza

Ecco se effettuato caricamento
solo della Forza Parte 2

Se effetto dissolvenza crescente
inizializza uscita = 0

Se effetto dissolvenza decrescente
inizializza uscita = 255



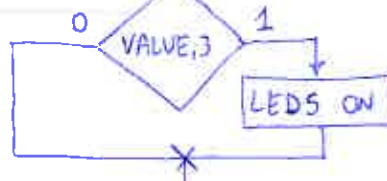
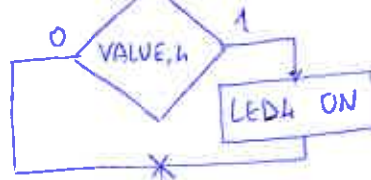
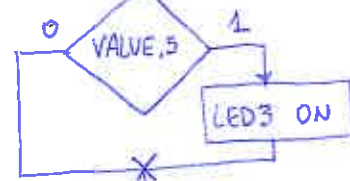


MENU1_8 PRESEPE 6

MENU1_8

ADDRESS = MENU - 1
CALL READ_EEPROM
VALUE MASK B'00111111' (63)

CALL LED_IRA_OFF



CPBUTT = 10

Ritardo 500ms



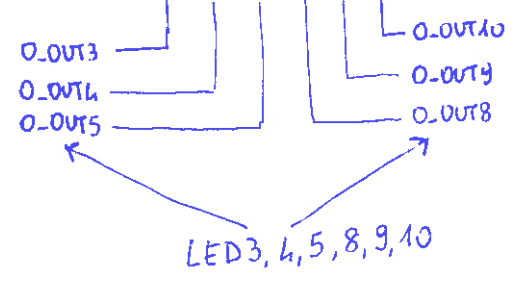
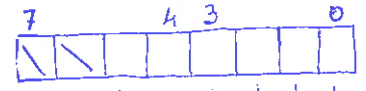
SALVA VALOR
en EEPROM
Indirizzo: MENU - 1
CALL WRITE_EEPROM

RET

Visualizzo valore mi 10 led
secondo la logica seguente

	32	16	8	4	2	1	
0 =	0	0	0	0	0	0	0
3 =	0	0	0	0	0	0	0
12 =	0	0	0	0	0	0	0
30 =	0	0	0	0	0	0	0
63 =	0	0	0	0	0	0	0

MSB LSB

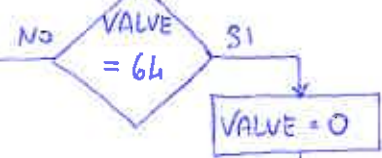


m18_1

m18_2

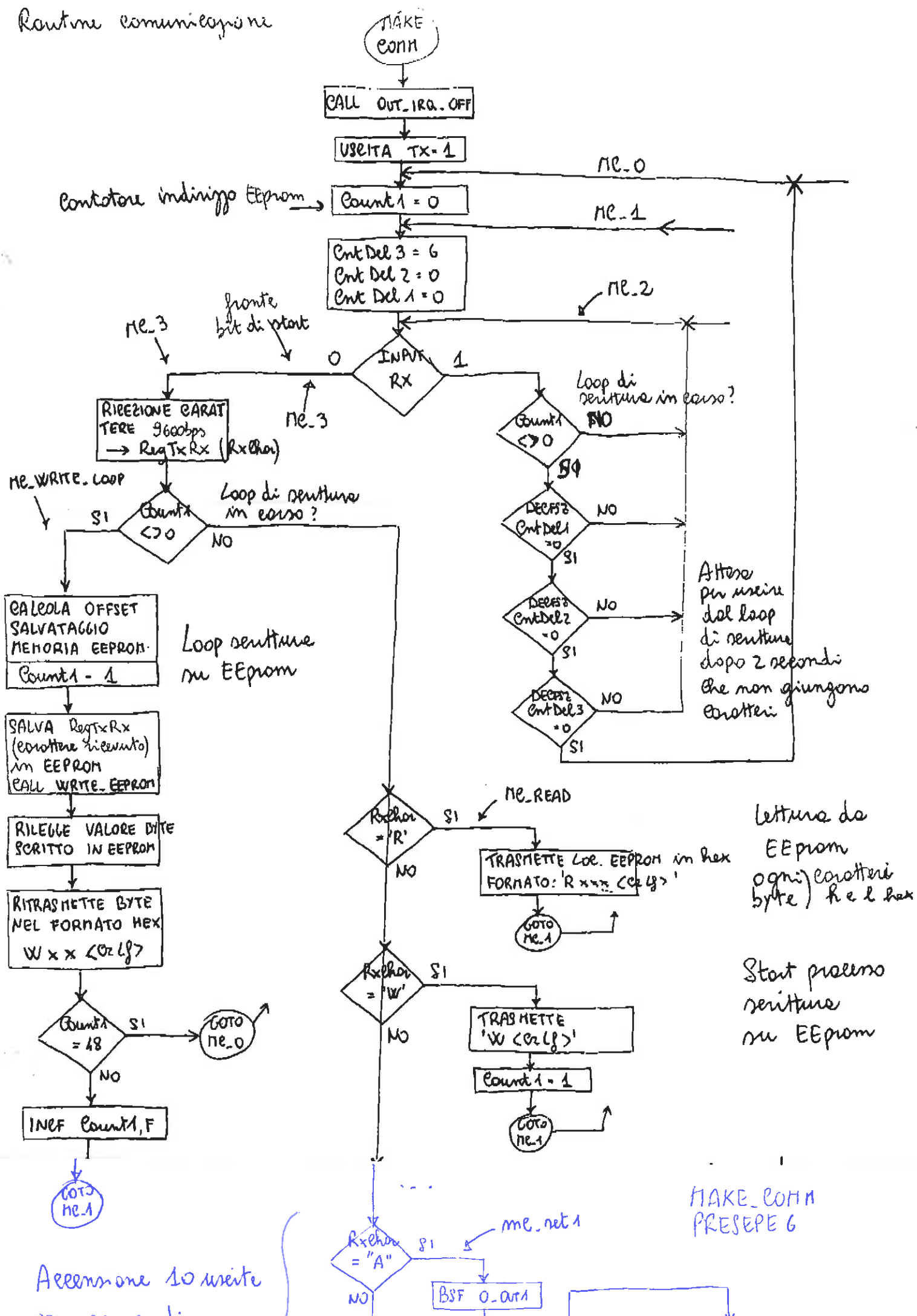
m18_3

INRF VALOR, F



m18_4

Routine communication

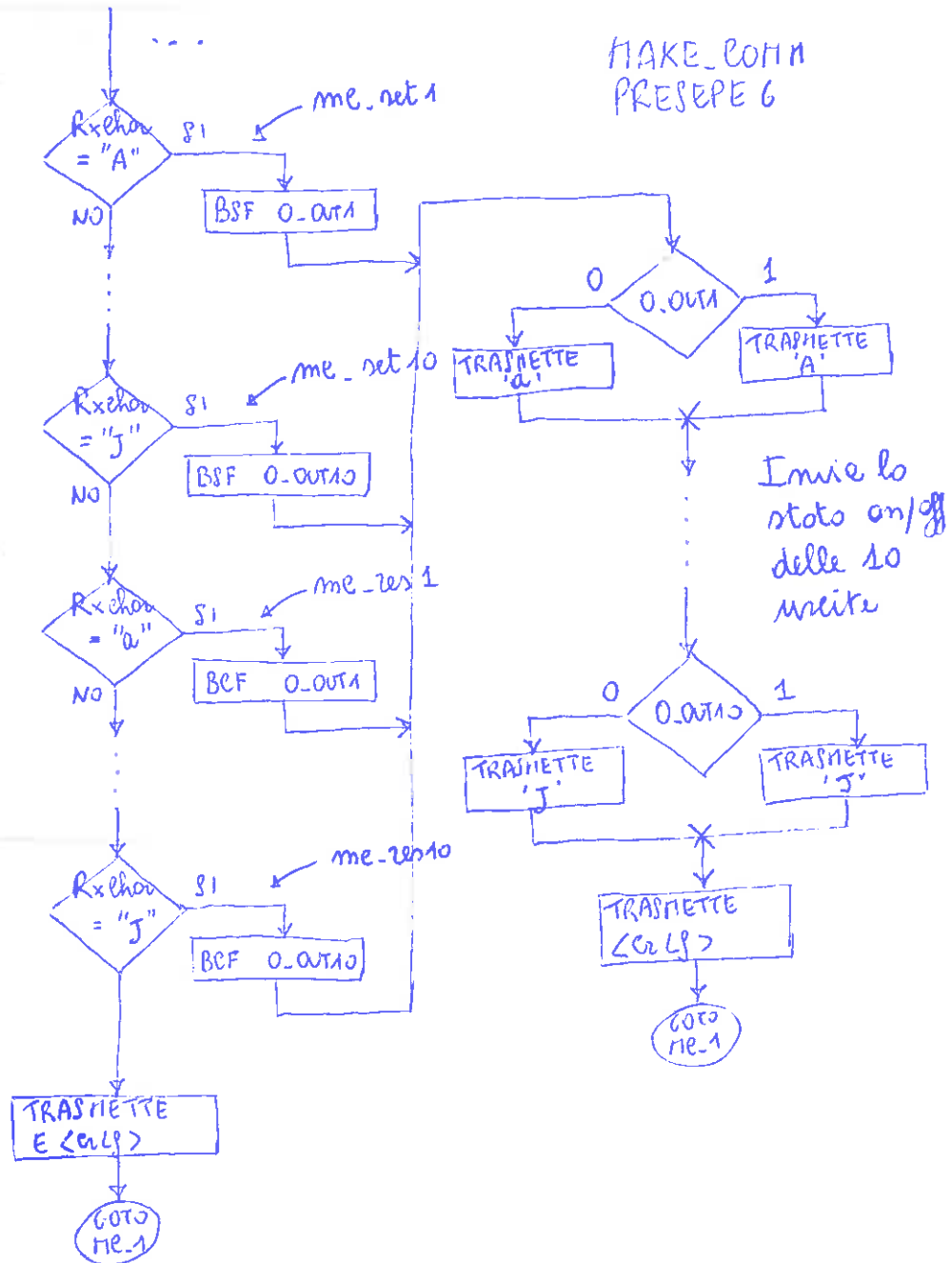


goto
ME-1

Accensione 10 uscite
con comandi
"A", ... "J"
(maiuscolo)

Spegnimento 10 uscite
con comandi
"a" ... "j"
(minuscolo)

Comando non
riconosciuto



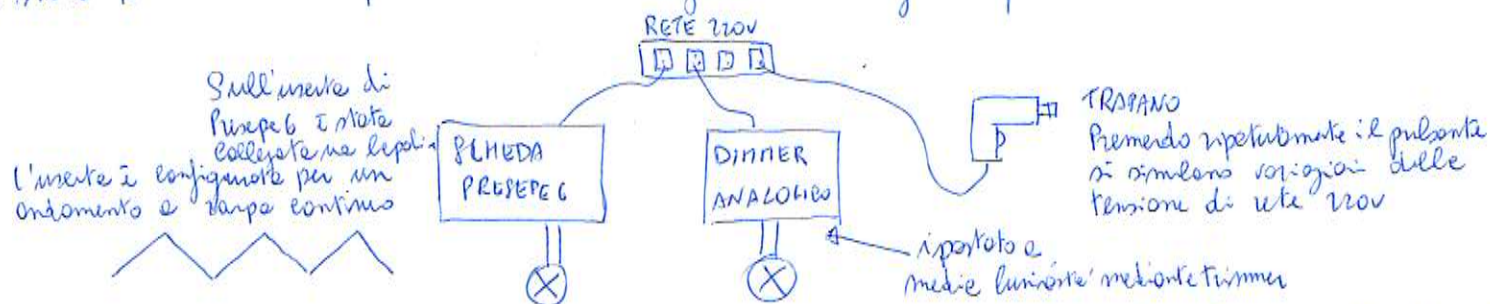
COMANDI VALIDI

- R : legge il contenuto della memoria EEPROM
- W : invia un ciclo di scrittura della EEPROM
- A, B, C, D, E, F, G, H, I, J : accende la singola uscita
- a, b, c, d, e, f, g, h, i, j : spegne la singola uscita

MODIFICA Rev. 1 FIRMWARE PRESEPE6 Versione AC 50/60 Hz

Si è notato in diverse situazioni che, durante le varpe delle uscite, le luminanze presentavano sfarfallii (rapidi aumenti e abbassamenti di luminanza).

Questo difetto si manifesta probabilmente perché ci sono rapide variazioni di fase della tensione di rete. Per simulare queste variazioni è stato collegato alla tensione di rete un Triac e è stato testato in contemporanea anche un regolatore analogico ipotizzato su medie luminanze.

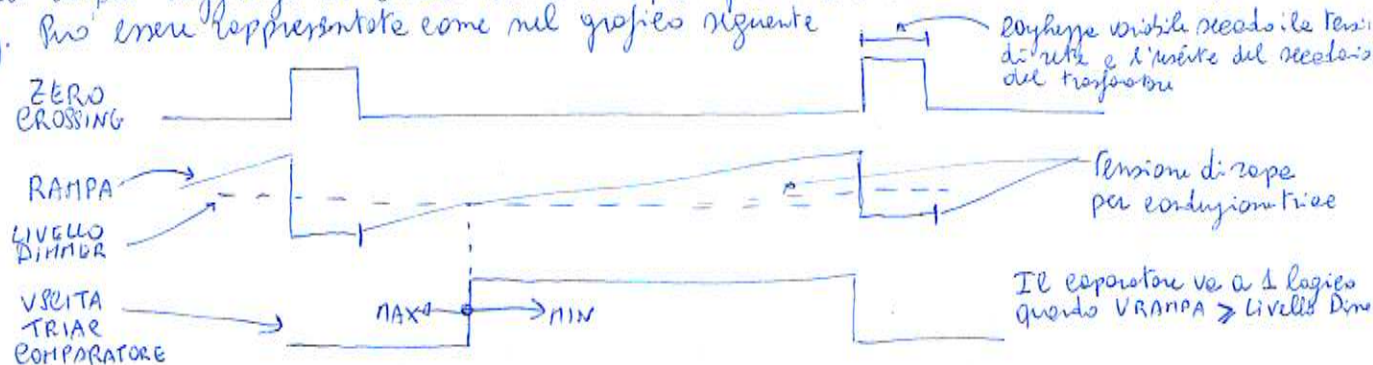


premendo ripetutamente il pulsante del Triac per simulare variazioni della tensione di rete, si può vedere che, mentre Prespe6 presenta lo sfarfallio, davanti il dimmer analogico è molto più immune ai disturbi. Si è quindi pensato di replicare (nel firmware contenuto in Prespe6) lo stesso funzionamento del dimmer analogico.

Il dimmer analogico mantiene a 0V la tensione di varpa durante tutto l'impulso di zero-crossing. La larghezza dell'impulso di zero-crossing è variabile e dipende fondamentalmente dalla tensione del trasformatore 9V che è stato utilizzato. Sono state fatte 3 misure come di seguito:

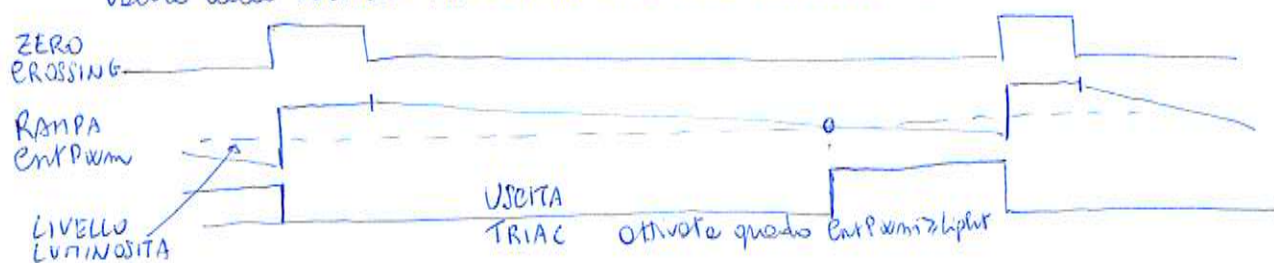
LARGHEZZA SEGNALE	9V	1,43 ms	Con un fototacciatore la larghezza dello zero-crossing si legge sui 580µs
ZERO CROSSING	10V	1,14 ms	
	12V	1,05 ms	

La tensione di varpa raggiunge il livello minimo poco prima che arriva il successivo impulso di zero-crossing. Può essere rappresentata come nel grafico seguente:

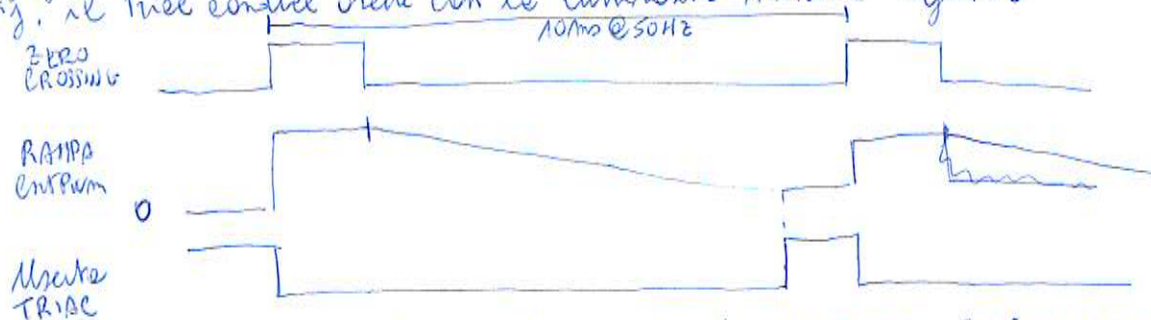


Alle routine contenute in Prespe6 (file IRAS.100) sono state apportate le seguenti modifiche:

- I/O RB0: Eliminato il ritardo EntZero di 380µs per ottenere zero-crossing reale.
- I/O TR0: Si esce dalla routine (per confronto e incremento del contatore varpe EntPwm) se l'ingresso RB0 del segnale zero-crossing è uguale a 1 logico. Questa modifica serve per documentare EntPwm solo quando non c'è lo zero-crossing. In realtà, la variabile EntPwm è decrementata da 255 a 0 per uniformare il valore delle variabili luminanze alle reali luminanze. 0 = MIN 255 = MAX



Questo significa che, se la rampa $ENTP_{min}$ va a \emptyset prima che arriva il prossimo segnale di $pro-cronig$, il tree continue anche con la luminance minima $light=0$



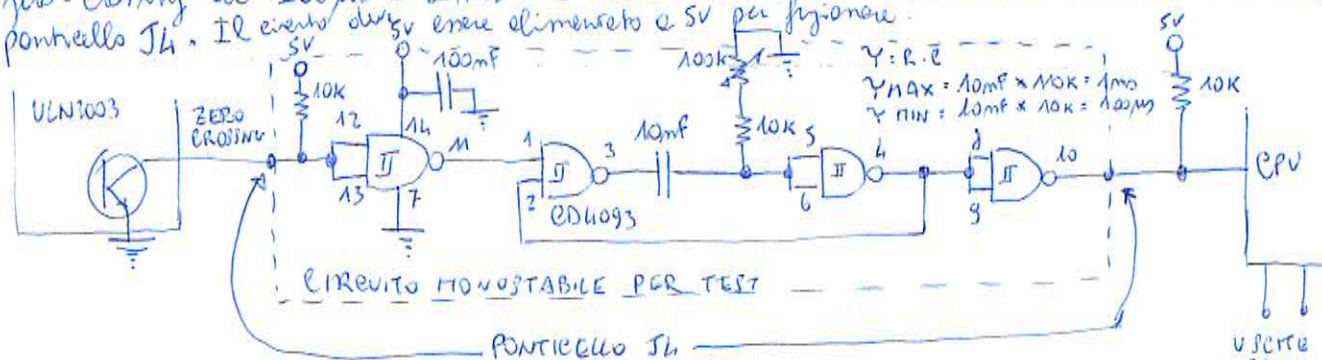
Il tempo delle rampe deve essere leggermente maggiore di $10 \mu s$ - l'ordine dei nanos. Si assume che la larghezza dello gate-cronig sia $1 \mu s$, quindi il tempo delle rampe entranti deve essere $9 \mu s$ a 50 Hz e $8 \mu s$ a 60 Hz .

I tempi di THRO sono stati riportati per avere la curva EntPwm in questo caso riproli espresso in due impulsi di zero-voltage.

Per verificare che la rampa CentPalm venga realmente eseguita nel tempo prefissato è stata fatta la seguente prova: Questa prova vale sia per il firmavoti a 50Hz che per quello a 60 Hz.

1) Inserito il circuito monostabile seguente che provvede ad allungare o restringere la lunghezza del segnale di zero-crossing da 100 μ s a 1 ms. È stato inserito tra l'inserito del VN2003 e le CPU, togliendo il ponticello J4. Il circuito è alimentato a 5V per funzionare.

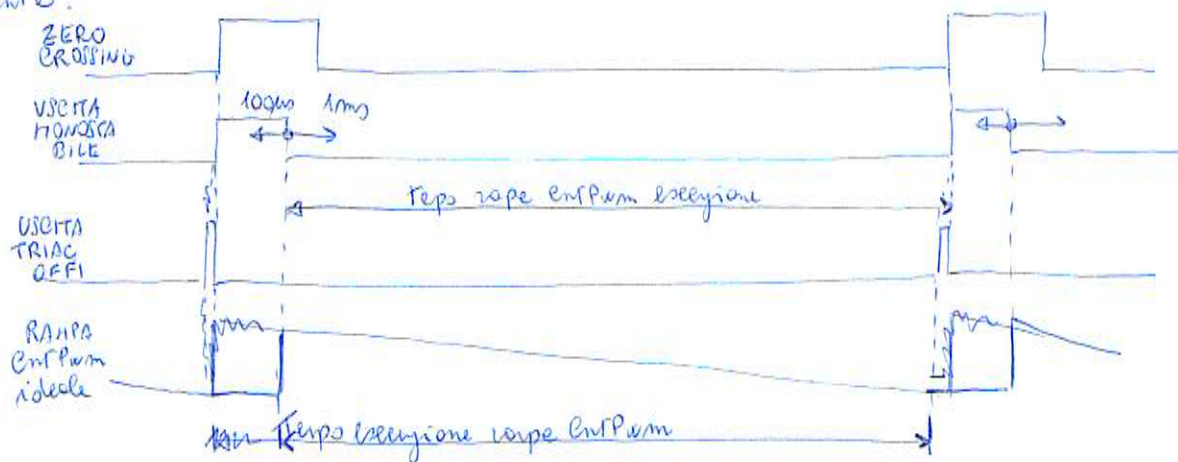
5V
100k
100nF
VN2003
10k
100nF
Y = R.C.
5V



2) Importata l'uscita 9 (e qualsiasi altre uscite di Preset 6) sopra a 0 (OFF).

3) Verifica con un oscilloscopio sincronizzato con il segnale μ -clock in uscita dell'ULN2003, che l'uscita del circuito monostabile viene da 100ms a 1ms.

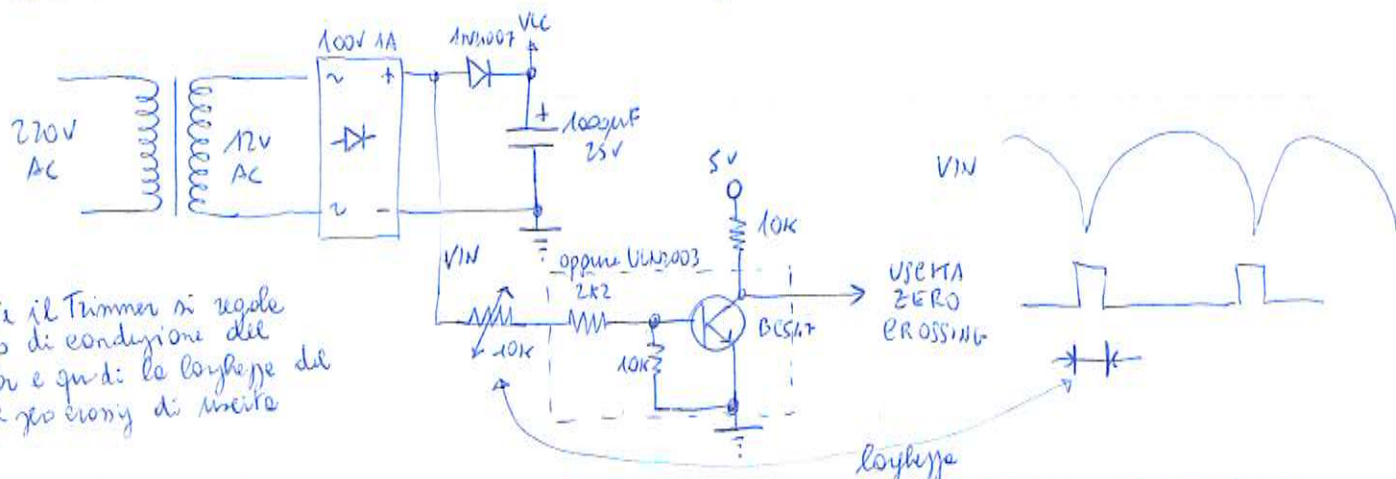
4) Collegare con un oscilloscopio sincronizzato con il segnale $\mu\text{-Clock}$ fronte, positivo verificato e collegato all'uscita riportata sopra ed OFF, girare il trimmer finché l'uscita TRIAC è: parte al livello positivo. Il tempo di esecuzione delle varie Entrata sono come indicato nel grafico seguente.



ARGINEZZA SEGNALE ZERO CROSSING

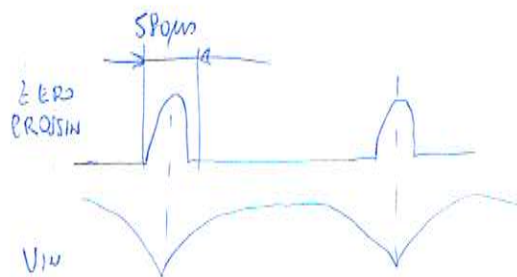
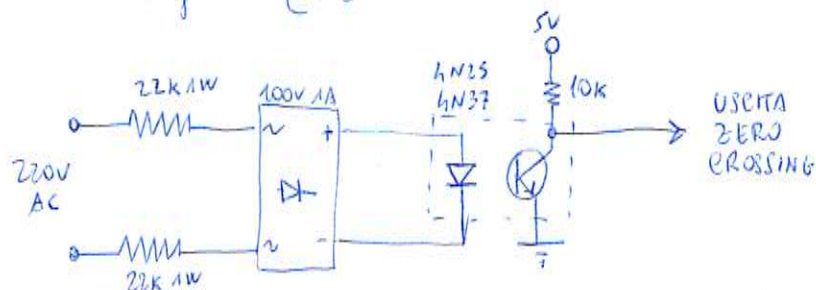
Il segnale di zero-crossing varia la sua larghezza in funzione delle tensioni applicate al transistor oppure al fotocaccoppiatore.

Puo' essere prelevato mediante secondario di un trasformatore oppure da un fotocaccoppiatore. I due circuiti mostrano le due diverse soluzioni essenziali:



La larghezza del segnale zero-crossing varia secondo le tensioni del trasformatore, del secondario

Uscente secondario trasformatore	9V	→	1,63 ms
	10V	→	1,16 ms
	12V	→	1,05 ms



Con questo tipo di circuito lo zero crossing è molto più stretto. Per aumentare la larghezza è sufficiente aumentare il valore delle due resistenze di ingresso al circuito.