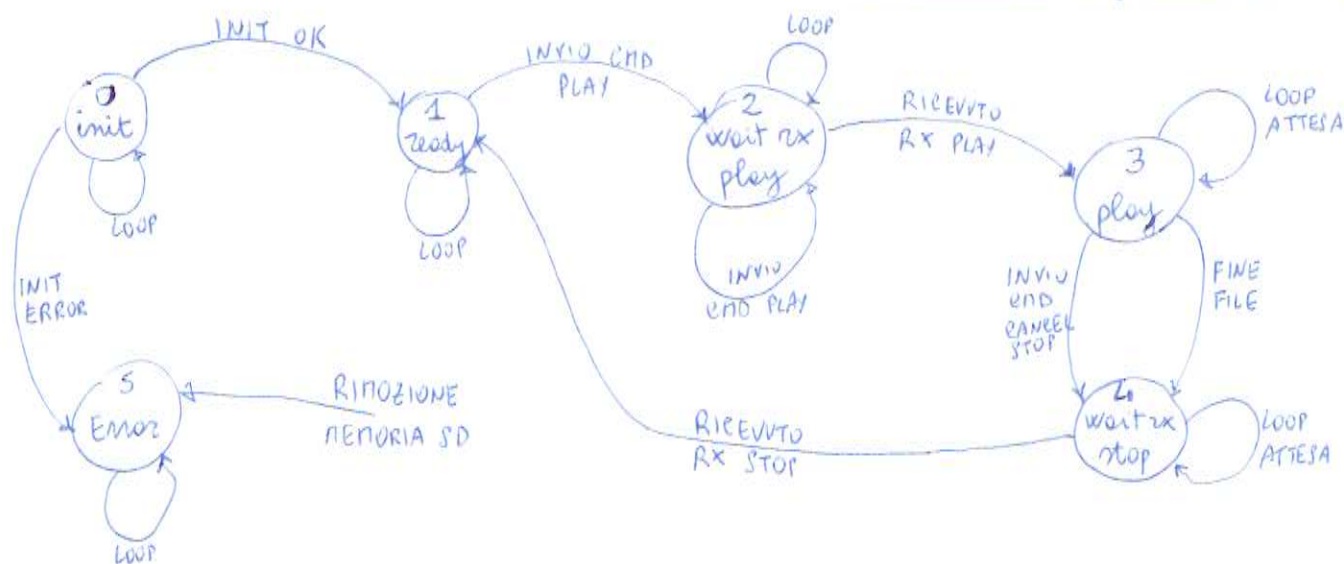


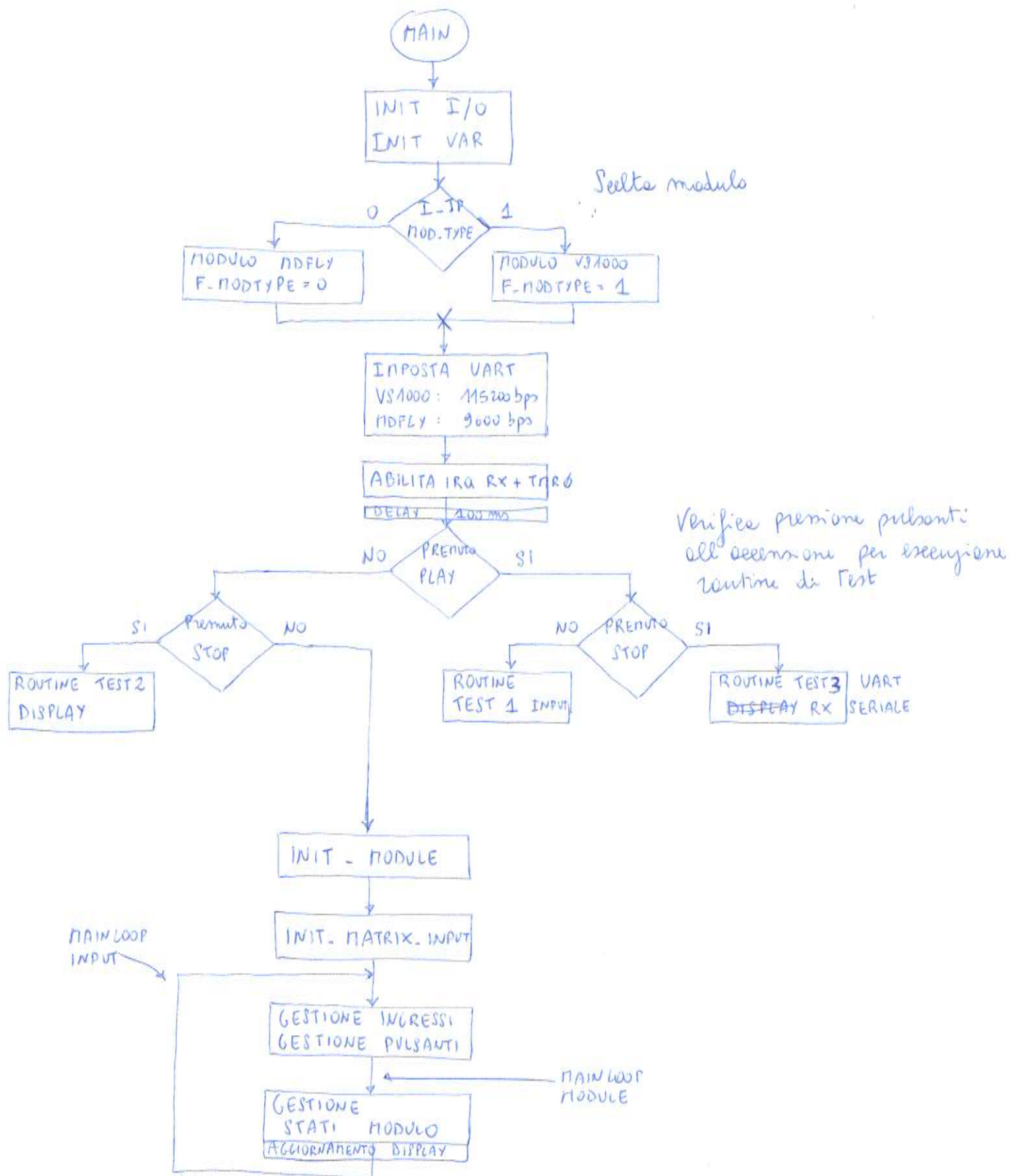
STATO	DESCRIZIONE	COMANDI POSSIBILI	DISPLAY	LED HEARTBEAT
0	Init	nessuno	spento	50% ON 50% OFF
1	Ready	play	- - fino	10% ON 90% OFF
2	Wait RX play	play	NUN lampeggiante	50% ON 50% OFF
3	Play	cancel (stop)	NUN fino	90% ON 10% OFF
4	Wait RX stop	nessuno	- - lampeggiante	50% ON 50% OFF
5	Error	nessuno	Er fino	OFF



## LOGICA INGRESSI E PULSANTI

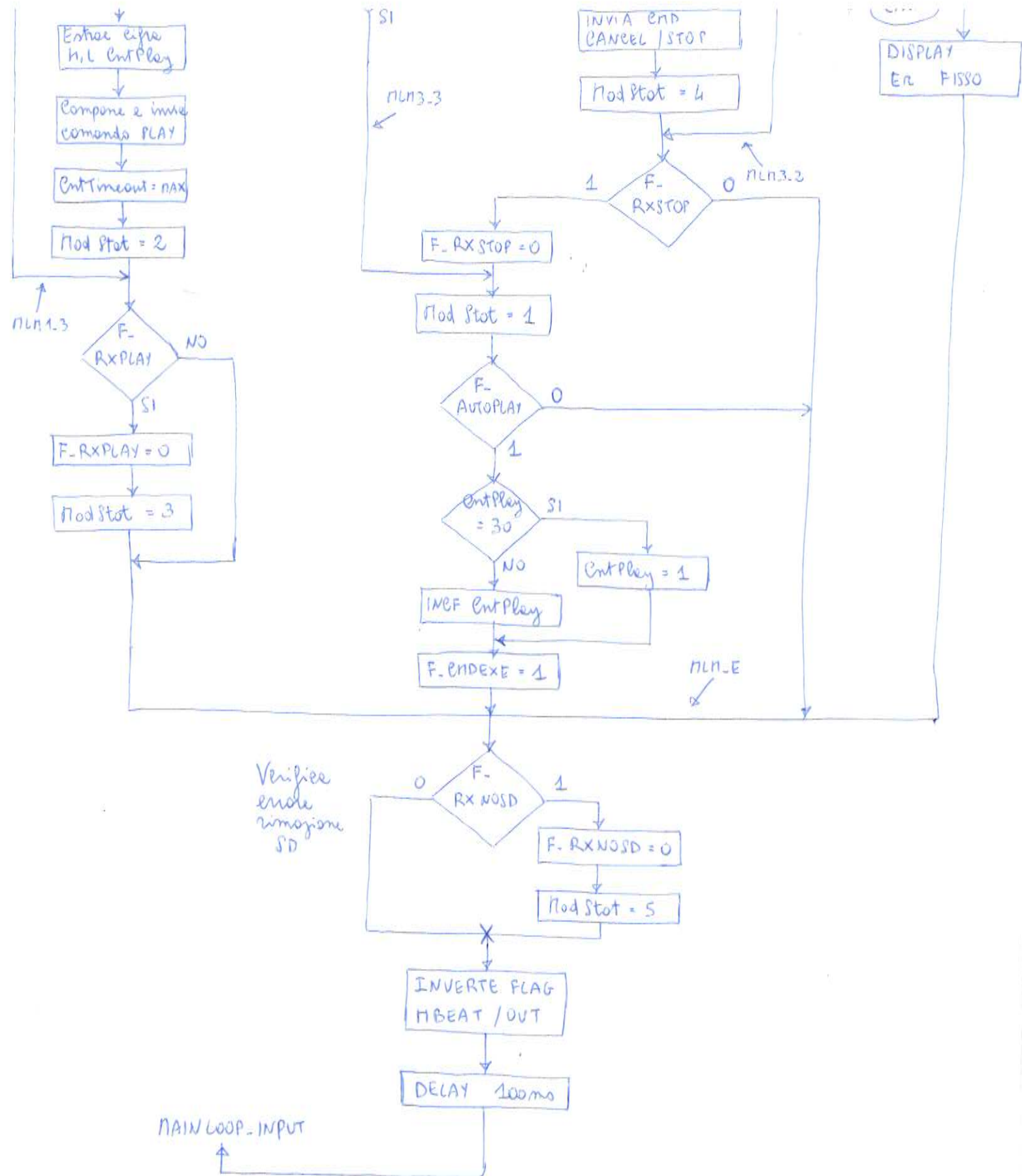
Stato precedente	Evento	Azione	Note
Qualsiasi Tranne Init e Error	Rilevato fronte 0 → 1 su ingresso 1...30	Play track 1..30	Disattiva modalità autoplay
Ready Attese play	Premuto pulsante Play / Next	Play 1 e succ.	Attiva modalità autoplay
Play	Premuto pulsante Play / Next	Play track successiva	Attiva modalità autoplay
Play	Premuto pulsante Stop	Stop Play Attese Play	Disattiva modalità autoplay

MUSIC\_BOX









COMANDI	VS 1000	MDFLY
PLAY TX	Pxx-----OGG Dove xx nome file da 01 a 30	0x01... 0x30
PLAY RX	play stringe ricevuta	0x01... 0x30 <small>leo del correttore Fiosmeno</small>
STOP TX	c <small>correttore c</small>	0xEF
STOP RX	done stringe ricevuta	0xFF (fine brano) 0xEF ( <small>comando stop</small> )

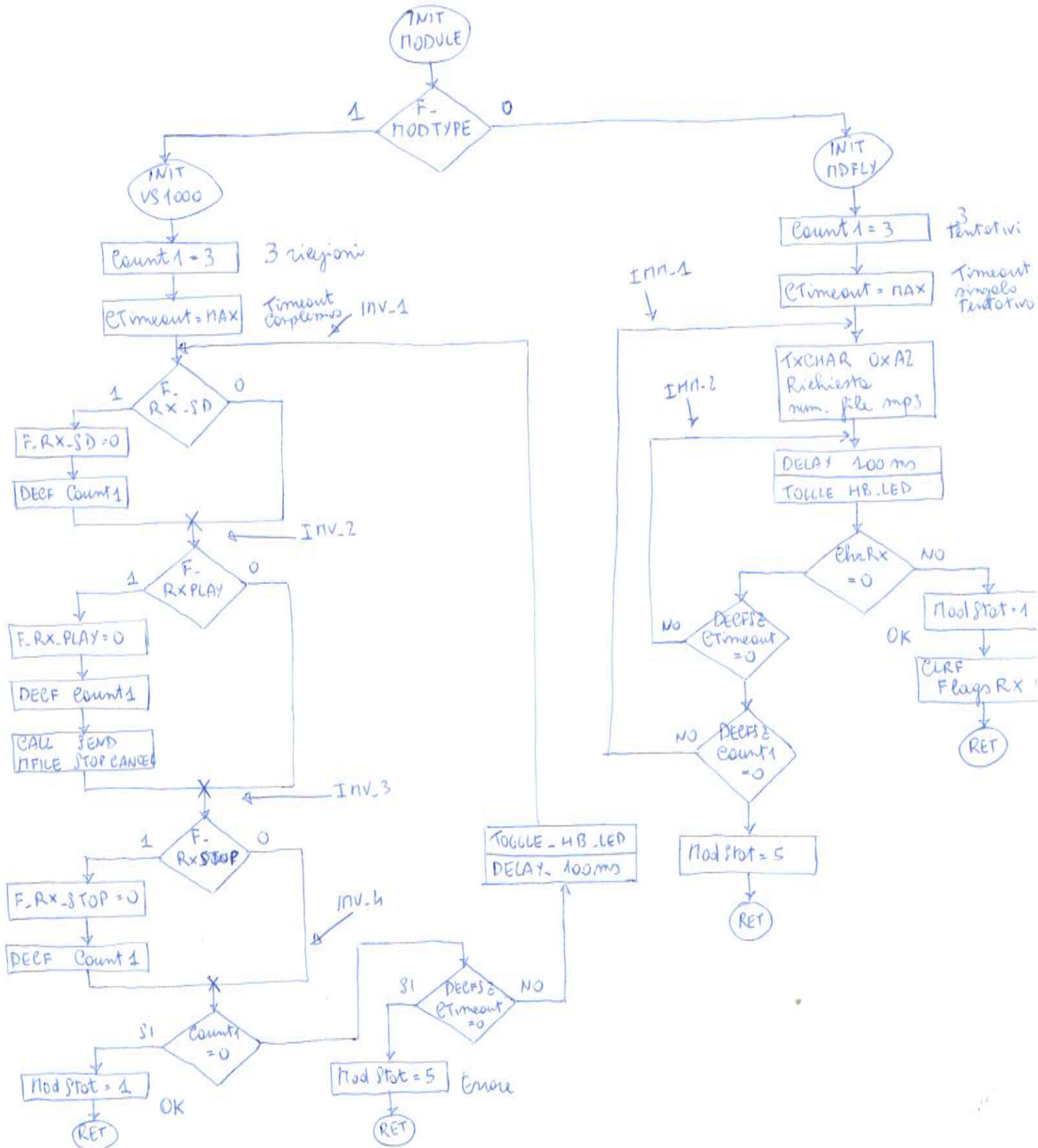
FLAG	VS 1000	MDFLY
F_RX_SD	settato quando si riceve la stringa SD che indica la posizione della memoria SD	
F_RX_NO_SD	settato quando si riceve la stringa !SD che indica rimanenza della memoria SD	
F_RX_PLAY	settato quando si riceve la stringa play dopo aver inviato un comando play	settato quando si riceve l'leo del correttore Fiosmeno 0x01... 0x030
F_RX_STOP	settato quando si riceve la stringa done a valle di un comando di stop oppure a fine traccia	settato quando si riceve il correttore 0xEF a valle di un comando stop oppure il correttore 0xFF a fine traccia

# INIZIALIZZAZIONE MODULI

VS1000 : Si attende la ricezione delle seguenti stringhe (entro 20 secondi)

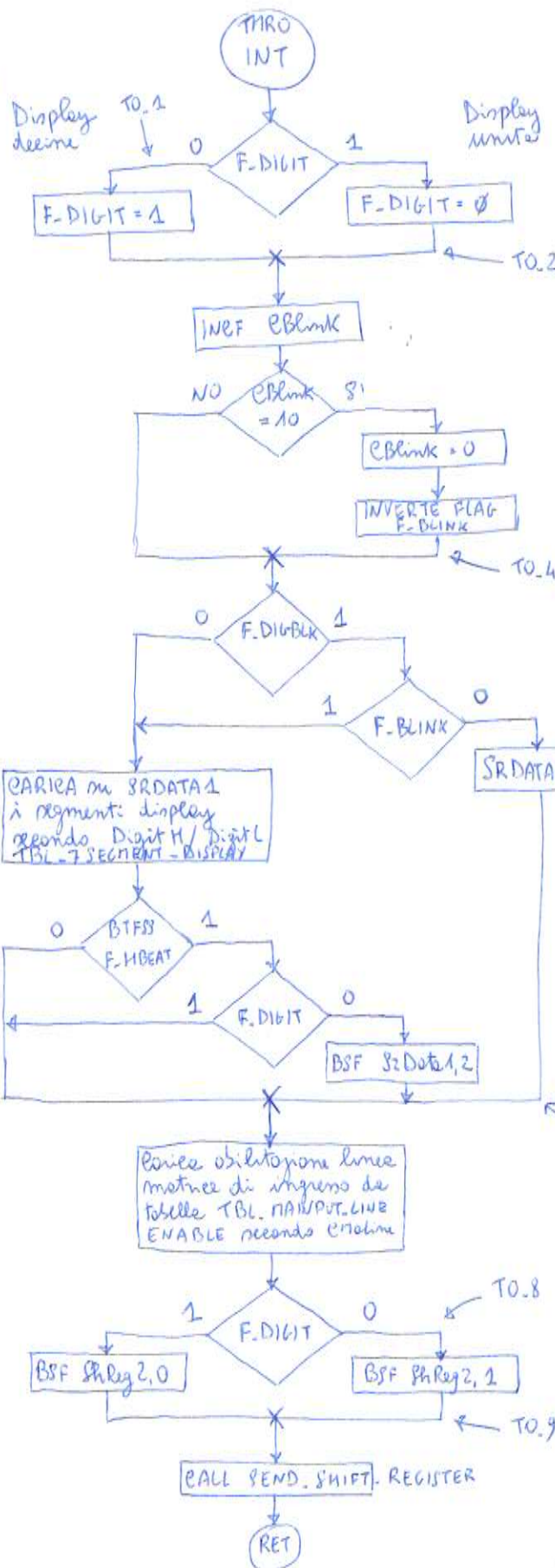
- SD : presenza memoria SD
- play : modulo in modalit  riproduzione automatica default  
Si inviano i comandi f (modalit  file) e e (stop)
- done : ricevuto comando stop. Modulo in attesa nelle modalit  file.

MDFLY : Si invia il comando 0xA2 (richiede numero file mp3 nelle SD)  
Se diverso da zero l'inizializzazione va a buon fine





Gestisce la matrice di ingresso e il multiplex dei due display



Eseguito ogni 10ms

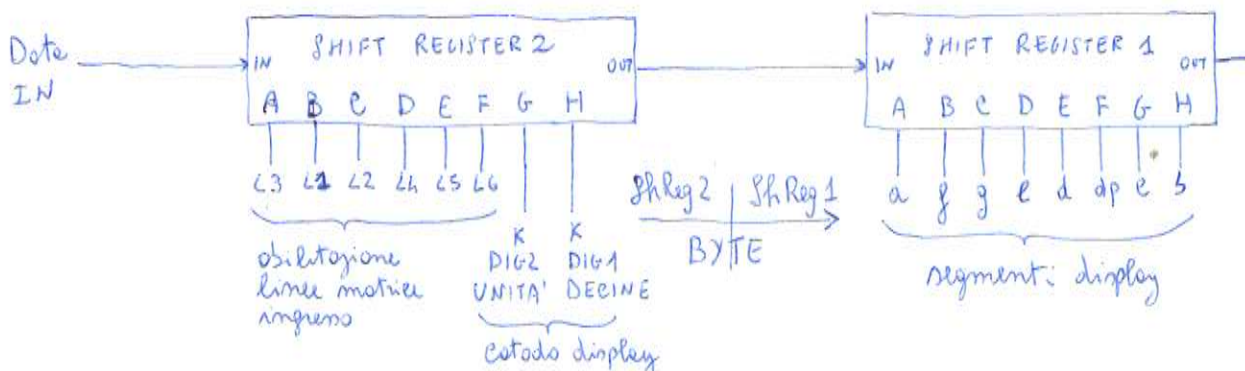
selezione display digit H, L

Inverte flag F-BLINK ogni 100ms per lampeggio display

Se lampeggio display abilitato accende / spegne segmenti secondo F-BLINK

Serve punto decimale su display LSB (F.DIGIT=0) secondo flag F-HBEAT

Accende estado display H o L secondo FLAG



Sette FlagRX  
singolo Bit  
F\_RX\_SD  
F\_RX\_NOSD  
F\_RX\_PLAY  
F\_RX\_STOP  
senza alterare  
altro bit  
(che venanno  
aspetti in Mainloop)



Preleva Input matrice  
e verifica fronti: 0 → 1

READ  
PA INPUT  
CHECK EDGE

Richiamato da Mainloop  
ed eseguito ogni 100ms circa

INCF CNoDelay, F

Attese 200ms fra letture di  
due righe consecutive

NO CNoDelay = MAX

CNoDelay = 0

FSR = CNoLine + NoInput1  
TempH = PORTA  
RLF TempH, F  
RLF TempH, W  
ANDLW B'1111000'  
MOVWF INDF

Preleva input (singole righe)  
e scrive su variabili  
NoInput (1 secondo contatore

INCF CNoLine

Incrementa contatore righe  
fino a 6

NO CNoLine = 6

RNEE-1

ELRF NoInEdge

RET

CNoLine = 0  
CNoInput = 1  
CNoByte = 0

Verifica fronti: 0 → 1

RNEE-2

CNoBit = 5

FSR = NoInput1 + CNoByte  
TempH = INDF

Individua byte stato  
input corrente → TempH

FSR = NoInput1 + CNoByte  
← INDF

Individua byte stato  
input precedente → INDF

RNEE-4

1 INDF, 7 0

Verifica fronte singolo bit

RNEE-5

RLF INDF, F  
RLF TempH, F

INCF CNoInput, F

NO DECF CNoBit = 0

INCF CNoByte

NO CNoByte = 6

SI NoInEdge = 0

PHN RnTe = 0

esse al primo  
fronte 0 → 1  
invece

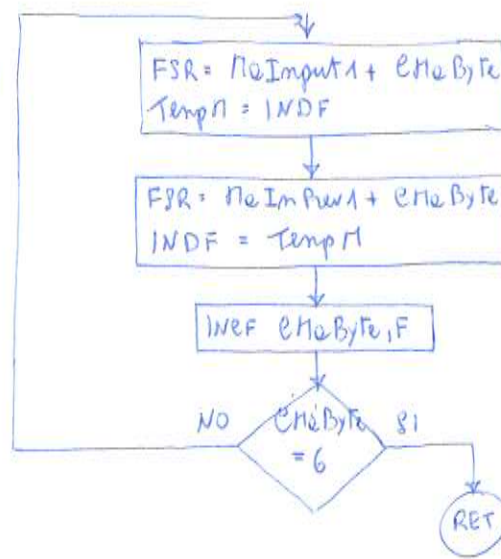
RNEE-6

Preleva  
Input

Verifica  
fronti  
0 → 1

RNEE-3

RICE-T



Copie Variabili  
NoInput (x)  
su  
NoInput (x)

Variabile  
ingresso metrice

Bit

		7	6	5	4	3	2	1	0
No Input 1		1	2	3	4	5			
= 2		6	7	8	9	10			
= 3		11	12	13	14	15			
= 4		16	17	18	19	20			
= 5		21	22	23	24	25			
No Input 6		26	27	28	29	30			

CIRCUITO  
BASE INGRESSI 1..15

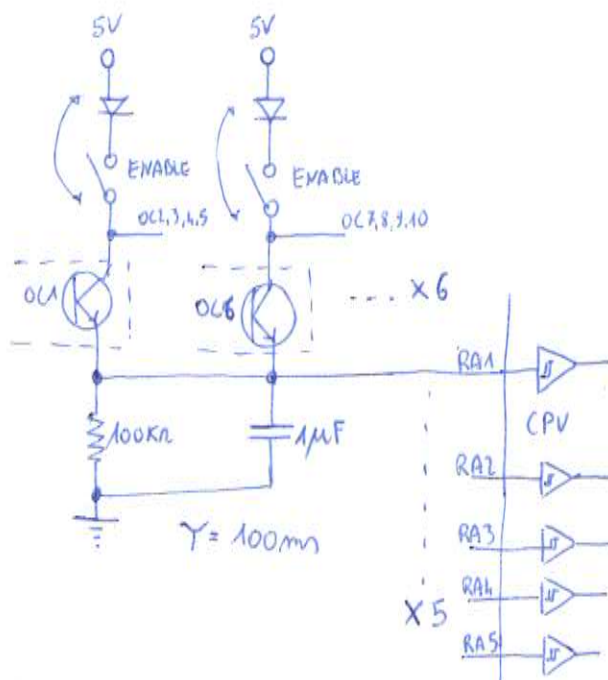
CIRCUITO ESPANSIONE  
INGRESSI 16..30

Non usati

bit ingresso precedente	bit ingresso corrente	Fronte 0 → 1
0	0	0
0	1	1
1	0	0
1	1	0

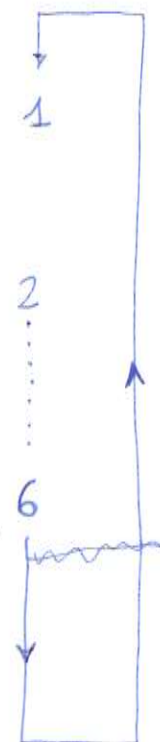
→ Rilevato fronte

## Notule ingressi



## Sequenze operazioni

- Applica ENABLE 1 (COLONNA 1)  $\Rightarrow$  alimenta fotorecettori OC1..OC5
- Aspetta 200ms
- legge ingressi e pu trigger smth e li copia su appoggio  $\overline{M}Input 1$
- Applica ENABLE 2 (COLONNA 2)  $\Rightarrow$  alimenta fotorecettori OC6..OC10
- Aspetta 200ms
- legge ingressi e pu trigger smth e li copia su appoggio  $\overline{M}Input 2$
- ...
- Confronta  $\overline{M}Input 1, 2, 3, 4, 5, 6$  con  $\overline{M}PreInput 1, 2, 3, 4, 5, 6$  fino a trovare una variazione di ingresso  $0 \rightarrow 1$ . Viene restituita  $\overline{M}Input Edge < 0 (1..30)$  se rilevato fronte  $0 \rightarrow 1$
- Copia  $\overline{M}Input 1, 2, 3, 4, 5, 6$  su  $\overline{M}PreInput 1, 2, 3, 4, 5, 6$  per prossimo confronto



Verifica funzionamento Tempo di attesa per scarica / carica RE (tempo  $\gamma$ ) su stesso ingresso CPU

$$OE1 = MAX \quad OE6 = 1/3 MAX \quad OE11 = 2/3 MAX$$

$$OE16 = 0 \quad OE21 = 2/3 MAX \quad OE26 = 1/3 MAX$$

$$MAX = \text{tensione ingresso ST} = 5V$$

OE1, OE11, OE21 devono restituire 1

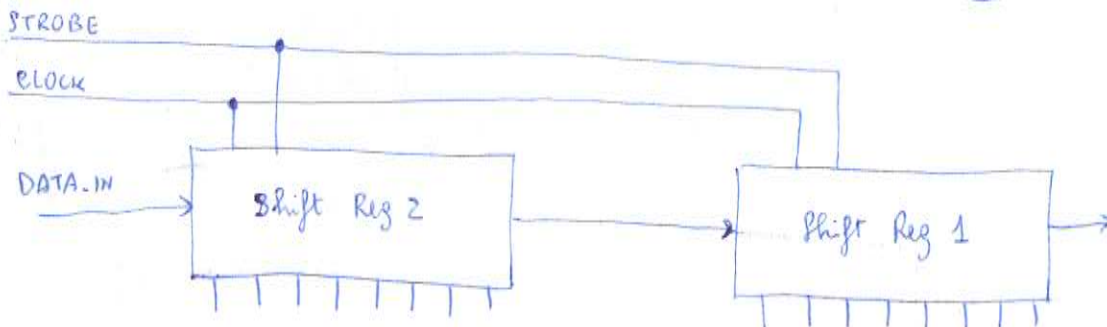
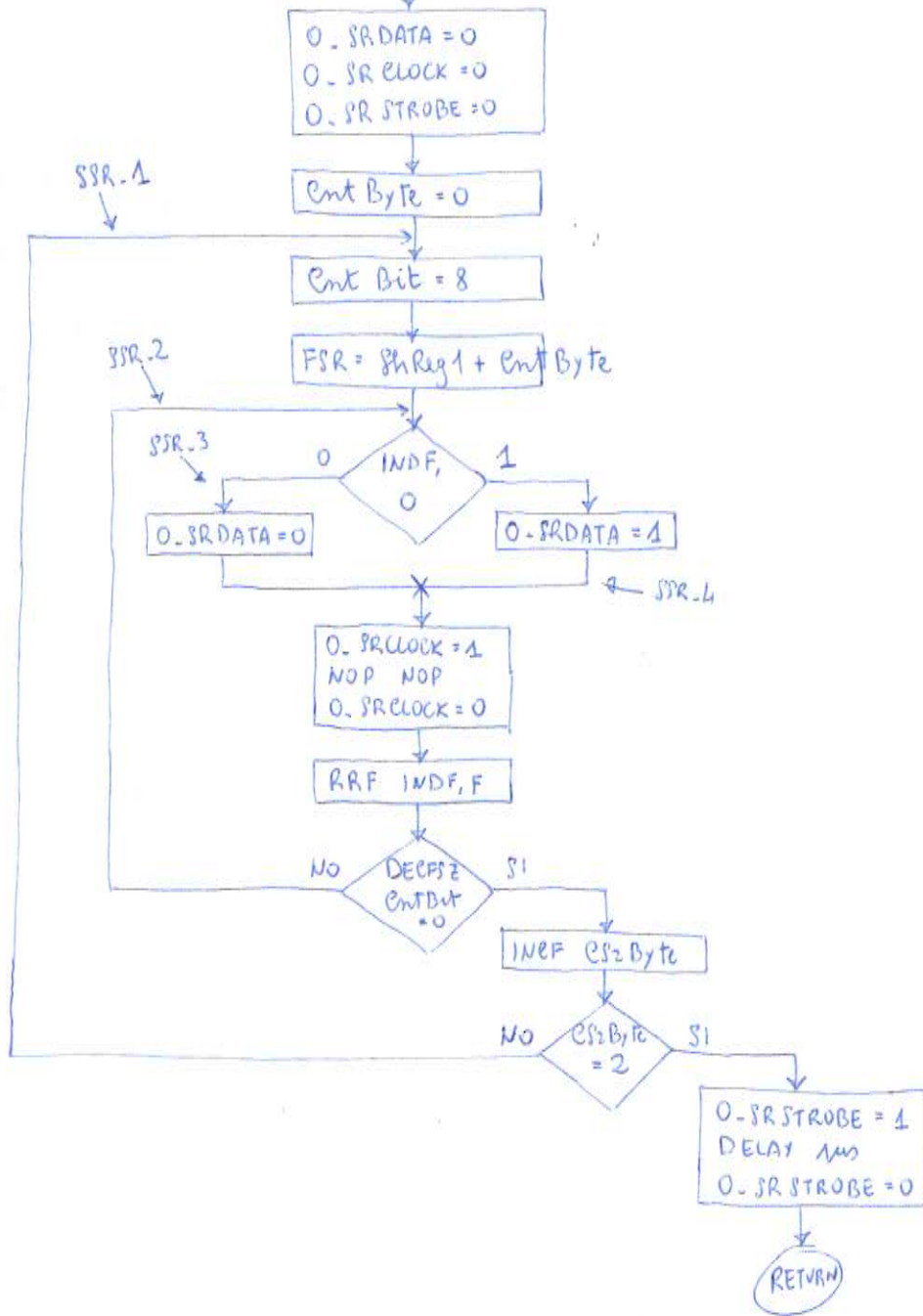
OE6, OE16, OE26 devono restituire  $\emptyset$

Il tempo di carica della RE è breve, più lungo è il tempo di scarica.

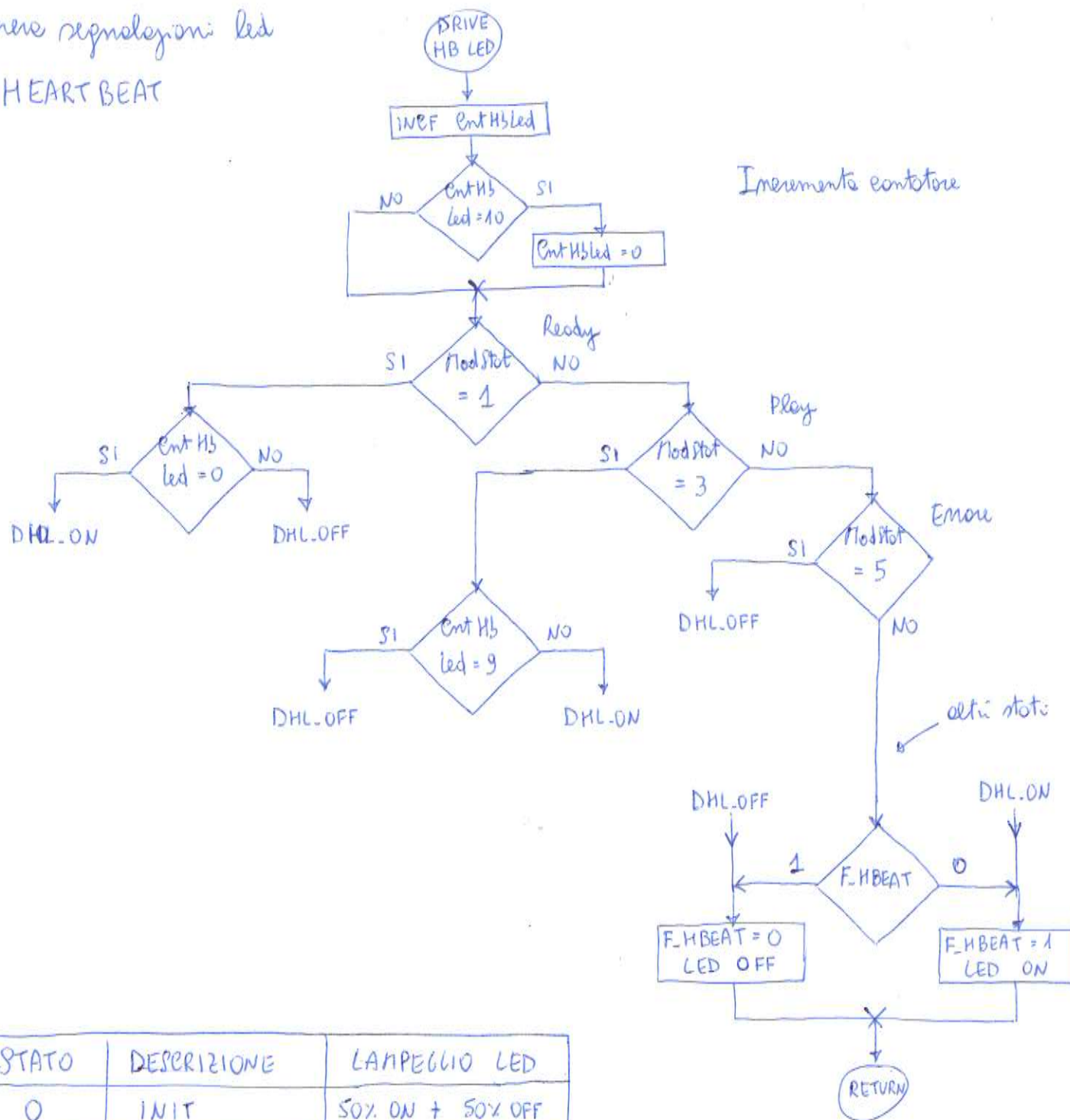


SEND  
SHIFT  
REGISTER

Initial: byte ShReg1 & ShReg2  
null shift register



# Genera segnalazione led HEART BEAT



STATO	DESCRIZIONE	LAMPEGLIO LED
0	INIT	50% ON + 50% OFF
1	READY	10% ON + 90% OFF
2	WAIT RX PLAY	50% ON + 50% OFF
3	PLAY	90% ON + 10% OFF
4	WAIT RX STOP	50% ON + 50% OFF
5	ERROR	OFF