

**Zadatak 1.** Sistem se sastoji od pokretne trake koja prenosi proizvode do mesta za testiranje na kome se odstranjuju neispravni proizvodi. Traka se pokreće motorom M. Kada proizvod stigne do mesta za testiranje, aktivira se senzor S, što zaustavlja traku. Nakon obavljenog testiranja, aktivira se jedan od dva signala I (proizvod je ispravan) ili N (proizvod je neispravan). Ako je proizvod ispravan (aktivan je signal I), traka nastavlja da se kreće. Ako je proizvod neispravan (aktivan je signal N), on se sklanja sa trake pomoću jednosmernog solenoida K, a zatim se traka ponovo pokreće.

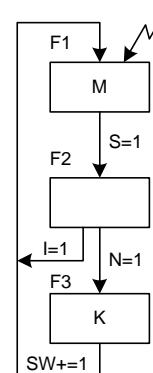
(**Ulazi:** S - proizvod prisutan na mestu za testiranje; I - proizvod je ispravan; N - proizvod je neispravan; SW+ - klip K je u krajnjem izvučenom položaju. **Izlaz:** M - pobuda motora, K - pobuda solenoida).

**Rešenje:** Najlakši način za kreiranje ispravnog lader programa, za probleme ovog tipa, jeste da se najpre rad (ponašanja) sistema opiše u vidu dijagrama stanja. Pod stanjem se podrazumeva vremenski interval u kome su izlazi sistema stabilni (ne menjaju se). Svako stanje odgovara jednoj fazi u radu sistema. Npr., u konkretnom primeru, jedna faza je vreme u kome se traka kreće (faza F1), druga faza je vreme u kome se obavlja testiranje proizvoda (faza F2), a treća vreme u kome se proizvod sklanja sa pokretne trake (faza F3). U fazi F1, motor je aktivan ( $M=1$ ), a solenoid neaktivan ( $K=0$ ), u drugoj fazi motor i solenoid su neaktivni ( $M=K=0$ ), a u trećoj motor je neaktivan, a solenoid aktivan ( $M=0, K=1$ ). Sistem prelazi iz jednog u drugo stanje kada se ispuni odgovarajući uslov, određen vrednostima ulaz. Tako, konkretan sistem prelazi iz F1 u F2 u trenutku kada se aktivira senzor ( $S=1$ ). U stanju F2 sistem čeka na aktiviranje jednog od ulaza I ili N. Ako se aktivira I, sistem se vraća u stanje F1, a ako se aktivira N, prelazi u F3. U F3 sistem čeka na aktiviranje graničnog prekidača solenoida ( $SW+=1$ ), a onda se vraća u stanje F1.

Dijagram stanja prikazan je na Sl.1.1. Stanjima odgovaraju pravougaonici, a grane predstavljaju prelaze između stanja. Svakom stanju su pridruženi izlazi koji su aktivni u tom stanju, a svakoj grani uslov koji inicira prelaz. Inicijalno stanje (tj. stanje u kome se sistem nalazi na početku rada) označeno je strelicom.

U suštini, dijagram stanja predstavlja formalni opis postavke zadatka, na osnovu koga je moguće direktno kreirati lader program. (Može se uspostaviti analogija sa programiranjem na nekom višem programskom jeziku, gde se najpre kreira algoritam, na osnovu koga se piše program).

Prvi korak u kreiranju lader programa je "kodiranje" stanja. Svakom stanju (fazi) pridružuje se jedan bit (iz interne memorije PLC kontrolera), koji kada je aktivan ukazuje da je sistem u tom stanju. Za konkretan problem, potrebna su tri takva bita. Neka su to: B3/0 (za F1), B3/1 (za F2) i B3/2 (za F3). U sledećoj tabeli navedene su vrednosti ovih bitova u svakom od tri stanja:



Sl. 1.1

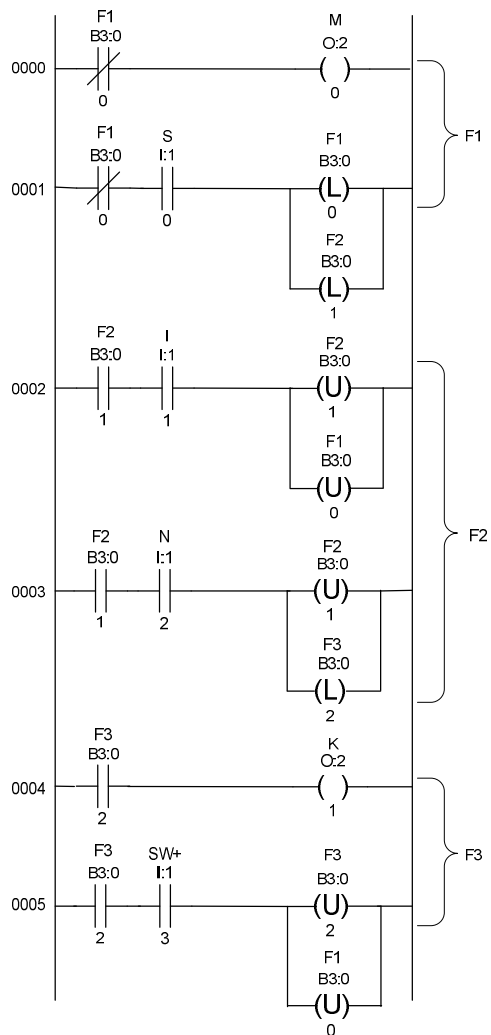
Stanje	B3/0	B3/1	B3/2
F1	0	0	0
F2	1	1	0
F3	1	0	1

Značenje pojma "bit je aktivan" razlikuje se za bit inicijalnog stanja i ostale bitove. Za bit inicijalnog stanja, u ovom slučaju B3/0, pod aktivnom vrednošću smatra se 0, dok su ostali

bitovi aktivni ako imaju vrednost 1. To je zato što su na početku rada svi bitovi ineterne memorije PLC kontrolera jednaki 0. Iz tabele se vidi je da za sve nule, sistem u stanju F1. Dakle, usvajamo sledeće: ako je  $B3/0=0$ , sistem je u F1, ako je  $B3/1=1$ , sistem je u F2, a ako je  $B3/2=1$ , sistem je u stanju F3.

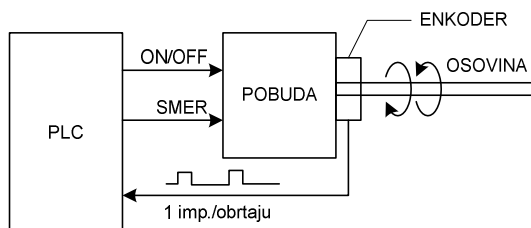
Leter dijagram, keriran na osnovu datog dijagrama stanja, prikazan je na SI.1.2. Način na koji su ulazima i izlazima dodeljeni priključci ulaznog i izlaznog PLC modula vidi se iz samog dijagrama.

Svako stanje se predstavlja sa najmanje dva ranga u leder programu. Prvi rang određuje koji su izlazi aktivni u tom stanju, a drugi vrši promenu stanja kada se ispuni odgovarajući uslov. U zavisnosti od problema, za svaku od ovde dve aktivnosti može biti potrebno više od jednog ranga. Svi rangovi koji odgovaraju istom stanju, u delu uslova imaju ispitivanje odgovarajućeg bita stanja, kako bi bili aktivni samo u tom stanju. Na primer, rangovi 0 i 1 odgovaraju stanju F1. Za ovo stanje važi  $B3/0=0$  i zato oba ranga sadrže XIO naredbu koja ispituje ovaj bit. Rang 0, "kaže" da je motor aktivan za sve vreme dok je sistem u stanju F1. Rang 1, "kaže" da sistem prelazi iz F1 u F2 ako je senzor aktivan ( $S=1$ ). Ovaj prelaz se ostvaruje tako sto se deaktivira bit tekućeg, a aktivira bit sledećeg stanja. Konkretno, za prelaz iz F1 u F2 treba setovati bitove  $B3/0$  i  $B3/1$ . U delu leder dijagrama za stanje F2, ne postoji rang koji pobuđuje izlaze, jer u ovom stanju ni jedan izlaz nije aktivan (tj.  $M=K=0$ ). Ali, zato postoje dva ranga za promenu stanja, zato što postoje dva uslova za napuštanje stanja F2. Prvi uslov je  $I=1$ , koji sistem vraća u stanje F1, što nameće promenu bitova stanja na:  $B3/0=0$  i  $B3/1=0$ . Drugi uslov,  $N=1$ , inicira prelaz u stanje F3, tj. ( $B3/1=0$  - napušta se F2; i  $B3/2=1$  - ulazi se u F3). Prvi rang u delu leder dijagram za stanje F3 pobuđuje solenoid ( $K=1$ ), a drugi ispituje uslov kraja ovog stanja ( $SW+=1$ ). U trenutku kada je solenoid izvučen do krajnjeg položaja ( $SW+=1$ ) bitovi stanja se manjaju na  $B3/2=0$ ,  $B3/1=0$ , čime se sistem vraća u polazno stanje F1. (Prelaskom su stanje F1, pobuda solenoida se ukida i on se, pod dejstvom opruge, vraća u uvučeni položaj.)



SI.1.2

**Zadatak 2.** Na Sl. 1.1 prikazan je podsistem za upravljanje rotacijom osovine koja je deo nekog složenijeg elektro-mehaničkog sistema. Blok POBUDA, koji sadrži motor i prateću elektroniku, okreće osovinu u zadatom smeru. Za ON/OFF=0, osovina miruje; za ON/OFF=1 i SMER=0, osovina se okreće u jednom, a za ON/OFF=1 i SMER=1 u drugom smeru. Blok ENKODER registruje obrtaje osovine i generiše jedan impuls za svaki obrtaj. Realizovati leder program za PLC kontroler koji će omogućiti da osovina naizmenično rotira po 100 obrtaja u oba smera sa pauzom od 1s pri svakoj promeni smera.



Sl. 1.1.

**Rešenje:** Slično kao u prethodnom zadatku, rad sistema se može razložiti na faze, koje se naizmenično ponavljaju, uvek u istom redosledu:

- F1 - rotacija u smeru SMER=0
- F2 - pauza između rotacije u smeru SMER=0 i smeru SMER=1
- F3 - rotacija u smeru SMER=1
- F4 - pauza između rotacije u smeru SMER=1 i smeru SMER=0

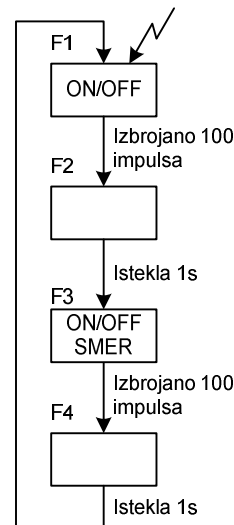
Nakon faze F4, sistem se vraća u fazu F1. Vrednosti izlaza sistema, po fazama (stanjima) su:

Stanje	ON/OFF	SMER
F1	1	0
F2	0	0
F3	1	1
F4	0	0

Sistem, iz stanja F1 prelazi u stanje F2, nakon izbrojanih 100 impulsa, a iz stanja F2 u stanje F3 nakon 1s provedene u stanju F2. Slično, za prelaz iz F3 u F4 treba čekati na izbrojanih 100 impulsa, a za prelaz iz F4 u F1 na istek 1s provedene u stanju F4. Znači, dijagram stanja oblika je kao na Sl.1.2. (Napomena: ako u pravougaoniku stanja ne postoji ime izlaznog signala, podrazumeva se da je njegova vrednost u tom stanju 0. Ako je ime signala navedeno, njegova vrednost je 1.)

Za manipulaciju stanjima biće nam potrebna četiri pomoćna bita. Neka su to: B3/0, ..., B3/3:

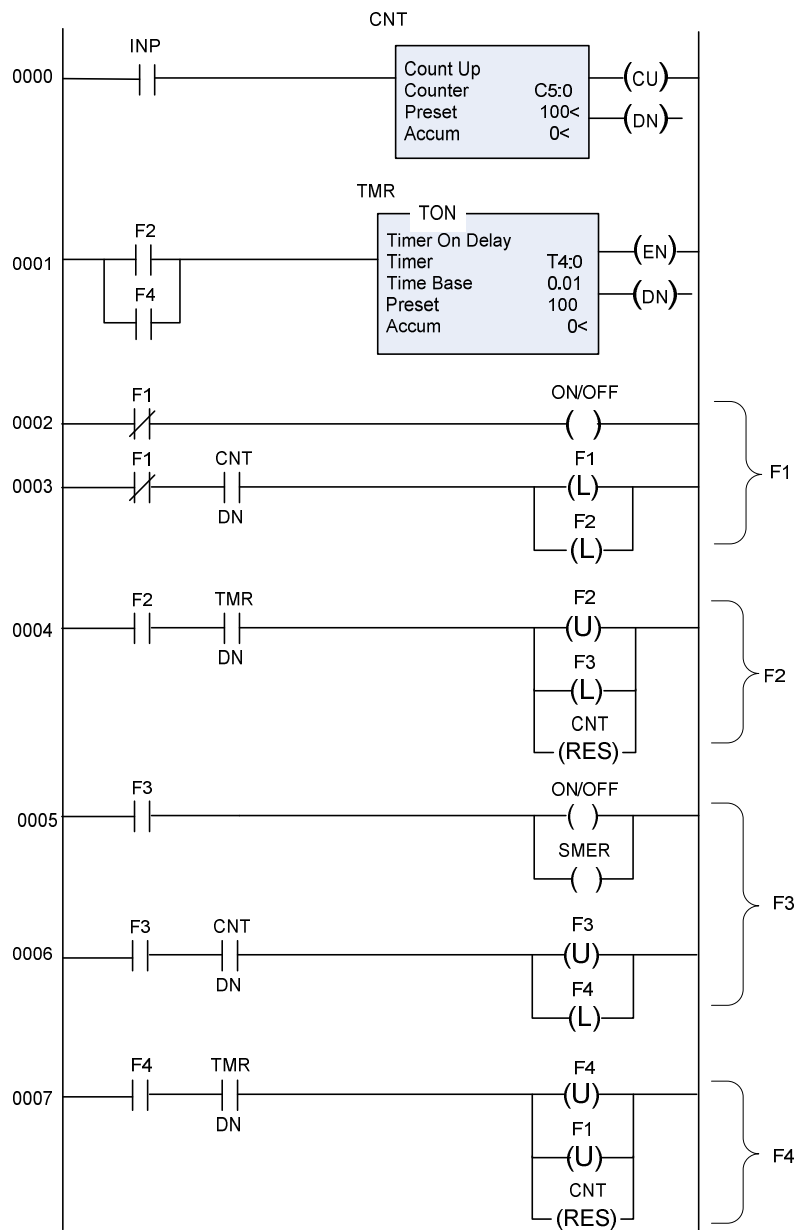
Stanje	B3/0	B3/1	B3/2	B3/3
F1	0	0	0	0
F2	1	1	0	0
F3	1	0	1	0
F4	1	0	0	1



Sl. 1.2.

Leder dijagram prikazan je na Sl. 1.3. Radi preglednijeg prikaza, adrese bitova, tajmera i brojača su izostavljene. Ulaz preko koga se PLC-u dovode impulsi označen je sa INP. Za merenje vremena koristićemo naredbu tajmera (TON), a za brojanje impulsa, naredbu brojača (CTU). Prva dva ranga upravljaju brojačem i tajmerom. Brojač,

CNT, neprekidno broji impulse, a kada izbroji 100 impulsa, aktivira bit DN. Ovaj bit se, kasnije, u stanjima F1 i F3 koristi kao uslov za prelaz u sledeće stanje (rang 3 i rang 6). Da bi u stanjima F1 i F3, brojač počeo brojanje od 0, on se resetuje prilikom prelaska iz stanja F3 u F4 (rang 4) i iz F4 u F1 (rang 7). Tajmer, TMR, radi u stanjima F2 i F4. Kada odmeri 1s, aktivira svoj bit DN, koji se koristi kao uslov za zavrsetak pauze (u stanjima F2 i F4). U stanjima F1 i F3, tajmer je resetvan, tako da po prelasku u stanje F2 (F3) počinje sa merenjem vremena iz početka.



Sl. 2.3.