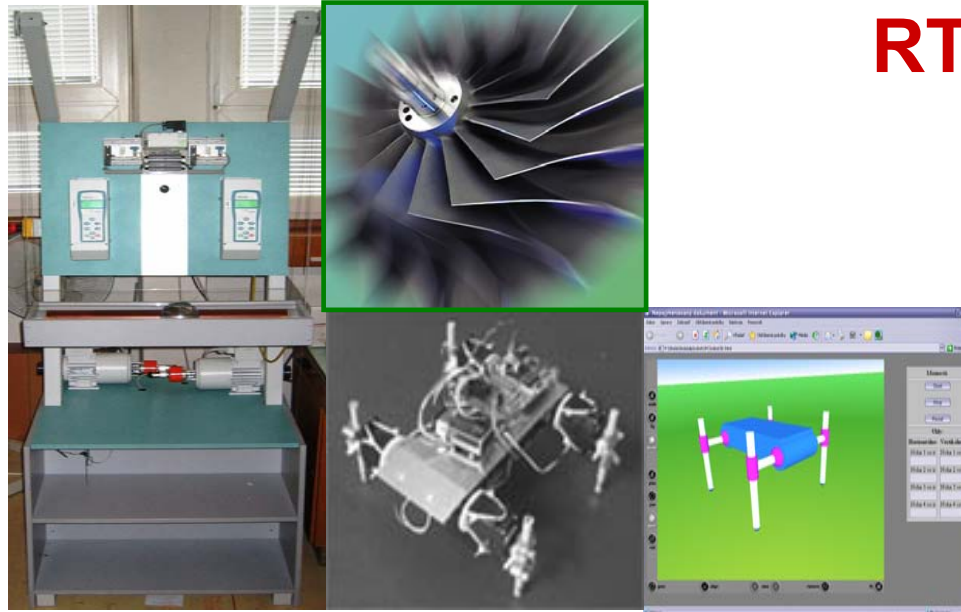
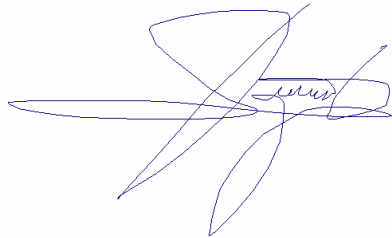




NÁVRH SEKVENČNÉHO LOGICKÉHO RIADENIA PRE PLC II. Stavové diagramy



Ing. Ján LIGUŠ, PhD.

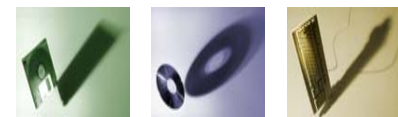


Technická univerzita v Košiciach,
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a umelej inteligencie



Obsah seminára

- Elementárne automaty
 - RS KLO kl.
 - RS KLO 0-dominant
 - D KLO
 - T KLO
- RS pomocou D
- D pomocou RS
- T pomocou D
- T pomocou RS





Rozdelenie aplikačných algoritmov

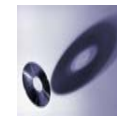
- Blokačné algoritmy
- Ochranné algoritmy
- Poruchové algoritmy
- Signalizačné algoritmy
- Sekvenčné algoritmy
- Regulačné algoritmy
- Komunikačné algoritmy –
- Diagnostické algoritmy – riadiacich, riadených
- Inicializačné algoritmy
- iné





Úvod do sekvenčného riadenia

- Veľmi odlišné riadenie vzhľadom na regulácie pomocou uzavretej regulačnej slučky
- Spojené s inteligenciou stroja
- Stroje a procesy obyčajne prechádzajú cez sekvenciu stavov pri svojej činnosti
- Rôzne akcie sú typicky vykonávané v každom stave a v cykloch
- Príklad: Píla skracovačka
 - Stav 1: podávanie kusa, brvna
 - Stav 2: Ukotvenie, upevnenie kusa, brvna
 - Stav 3: Odrezanie
 - Stav 4: Okončenie uchopenia a skok na stav 1
- Cykly potrebujeme vyberať a riadiť





Ďalšie príklady sekvenčného riadenia

- Technologické procesy
 - Manual/Auto prepínanie módu
 - Výber súčiastky v tolerancii, alebo mimo tolerancie
 - Výstražne hlásiče, majáky, zvukové zariadenia a pod.
- Automobily
 - hlásič pripútania sa bezpečnostným pásom
 - Automatické vypínanie a zapínanie svetiel pri parkovaní
 - Automatické zatváranie dverí pri pohybe
- Výtahy
 - Výtahový systém





Vstupy do sekvenčného riadenia

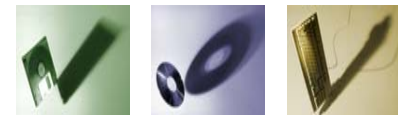
- Sekvenčná logika je riadená vstupmi
- Príklady logických vstupov
 - Tlačidlá ŠTART STOP
 - Prepínače - Manual/Auto prepínanie módu
 - Limitné spínače
 - Čítače (vypršanie časového limitu, časové oneskorenia)
 - Snímače maximálnych hodnôt (prekročenie teploty)
 - Koncové spínače
 - Signály od iných systémov (READY, BUSY,...)





Výstupy sekvenčného riadenia

- Výstupy sú ovládané sekvenčným riadením
- Príklady logických výstupov
 - Zapnutie, vypnutie motora
 - Zapnutie, vypnutie pohonu akčného člena (pohon čerpadla, frekvenčný menič ...)
 - Selenoid zap., vyp.
 - Žiarovky, osvetlenia, indikátorov, majákov a pod.
 - Alarmových hlásičov
 - Generovanie signálov pre iné systémy (READY, BUSY, ...)

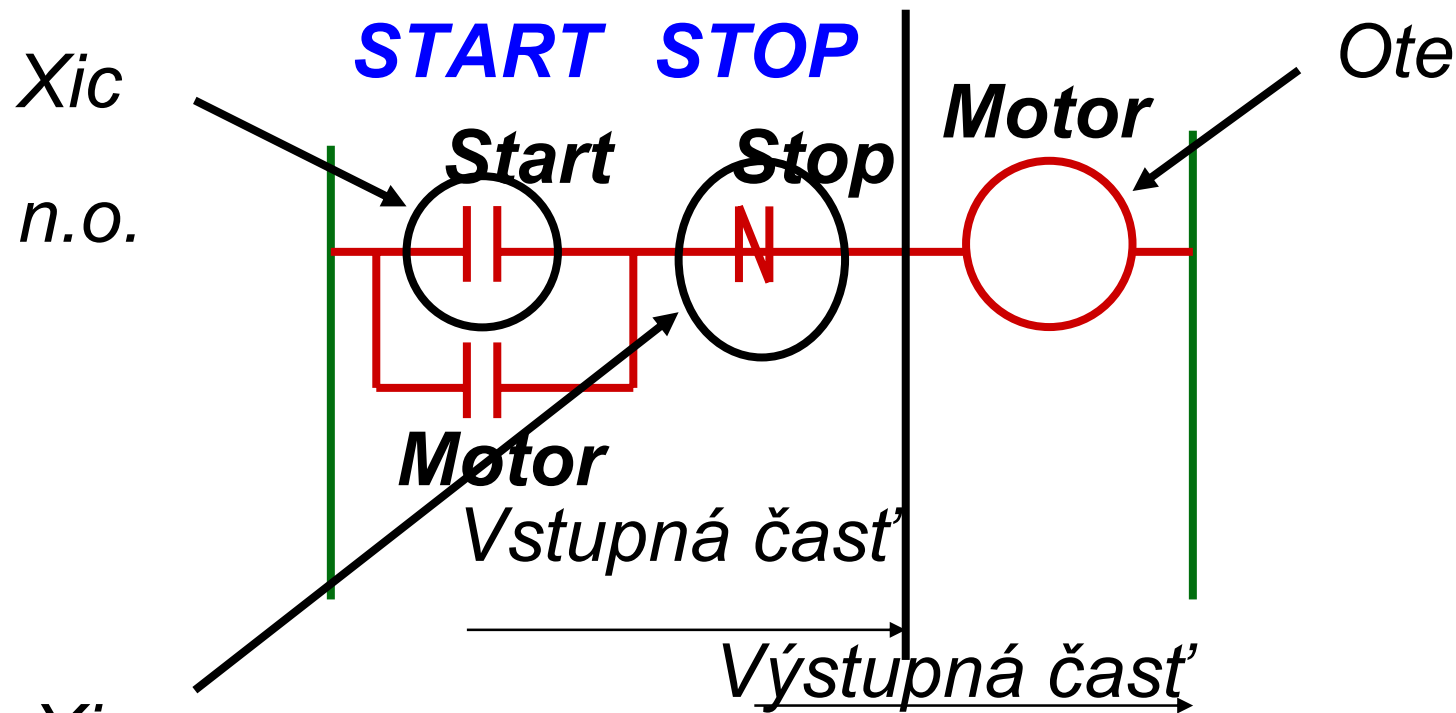




RS KLO 0-dominant

$$Q_n = S \cdot \text{NOT}(R) + Q_{n-1} \cdot \text{NOT}(R) = \text{NOT}(R) [S + Q_{n-1}]$$

$$\text{Motor} = \text{NOT}(\text{Stop}) [\text{Start} + \text{Motor}]$$



RS KLO 0-dominant !!!

n.c.



Implementácia ako počítačový program

Motor = NOT(Stop) [Start+Motor]

```
Void Motor_Control_Logic() {  
Boolean START, STOP MOTOR;  
MOTOR = false;           //poc. hodnota  
While(true) {  
    START = P1;           // nacistanie vstupu P1  
    notSTOP = P2;        // nacistanie vstupu P2  
    MOTOR = (MOTOR | START) & (notSTOP);  
    P3 = MOTOR;         // posielanie na vystup  
}  
}
```



Kroky pre návrh sekvenčného riadenia

- Špecifikácia cyklov operácie
- Špecifikácia bezpečnostných podmienok
- Identifikácia stavov v každom cykle
- Špecifikácia prechodov medzi stavmi
- Špecifikácia vstupov pre jednotlivé prechody
- Špecifikácia výstupov pre každý stav
- Verifikácia cyklov operácie
- Verifikácia bezpečnostných podmienok





Stavové prechodové diagramy

- Riadenie strojov a procesov sa uskutočňuje prechodmi sekvencií stavov zvolenej operácie
- Rôzne akcie sú vykonávané v každom stave v cykle stavov
- Stavové prechodové diagramy sú vhodné pre reprezentáciu cyklov stavov, vstupy, ktoré ich riadia, výstupy, ktoré sú ovládané nimi, a logika, ktoré ich ovláda
- Stavové prechodové diagramy hneď po ich návrhu potrebujú byť preložené do výrokovej logiky a implementované do počítačových programov





Stavové diagramy

Matematický model sekvenčního logického obvodu

- sekvenční logický obvod je závislý od sekvencie vstupov

- konečný automat definujeme $A = (X, S, Y, \delta, \lambda)$

X množina vstupov $X = \{x_1, x_2, \dots, x_N\}$

S množina stavov $S = \{s_1, s_2, \dots, s_R\}$

Y množina výstupov $Y = \{y_1, y_2, \dots, y_M\}$

δ prechodová funkcia $\delta : X \times S \rightarrow S$

λ výst. funkcia Mooly $\lambda : X \times S \rightarrow Y$

λ' výst. funkcia Moore $\lambda' : S \rightarrow Y$





Stavové diagramy

- konečný automat definujeme $A = (X, S, Y, \delta, \lambda)$

X množina vstupov $X = \{x_1, x_2, \dots, x_N\}$

S množina stavov $S = \{s_1, s_2, \dots, s_R\}$

Y množina výstupov $Y = \{y_1, y_2, \dots, y_M\}$

δ prechodová funkcia $\delta: X \times S \rightarrow S$

λ výst. funkcia Mooly $\lambda: X \times S \rightarrow Y$

λ' výst. funkcia Moore $\lambda': X \rightarrow Y$

$$S(t+1) = \delta(X(t), S(t))$$

$$Y(t) = \lambda(X(t), S(t)) \quad \text{Mooly automat}$$

$$Y'(t) = \lambda'(S(t)) \quad \text{Mooreov automat}$$



Moolyho stavové diagramy

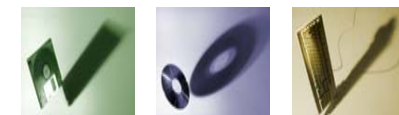
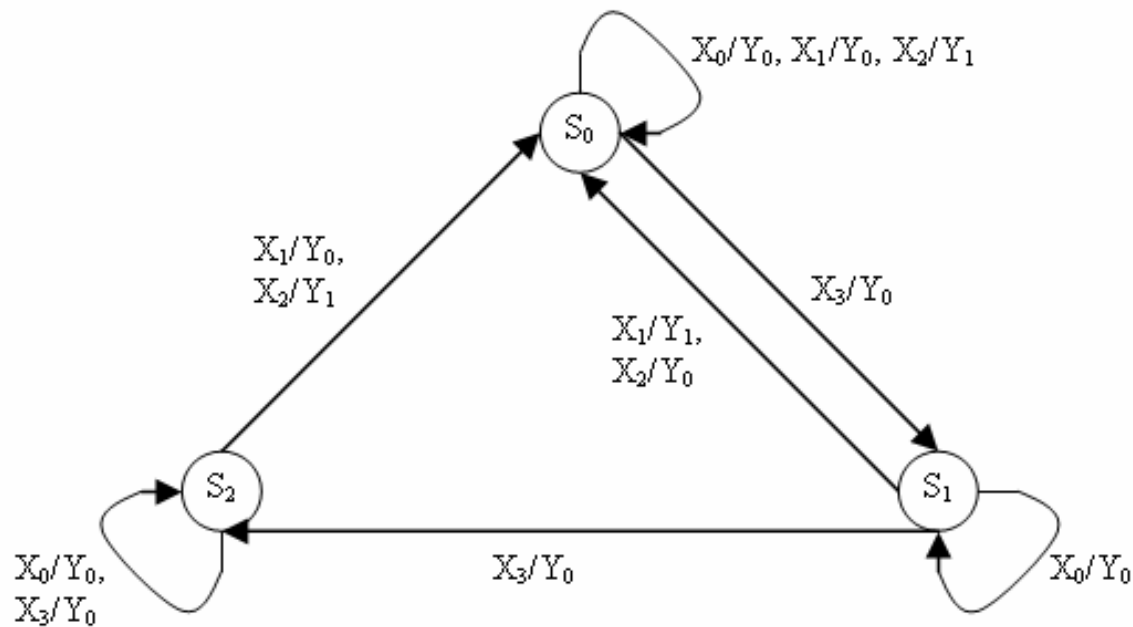
- prechodová a výstupná tabuľka
 - pre Mooly automat

$S(t) \backslash X(t)$	X_0	X_1	X_2	X_3
S_0	S_0	S_0	S_0	S_1
S_1	S_1	S_0	S_0	S_2
S_2	S_2	S_0	S_0	S_2

$S(t) \backslash X(t)$	X_0	X_1	X_2	X_3
S_0	Y_0	Y_0	Y_1	Y_0
S_1	Y_0	Y_1	Y_0	Y_0
S_2	Y_0	Y_0	Y_0	Y_0

prechodová tabuľka $\delta: S(t+1)$

výstupná tabuľka $\lambda: Y(t)$

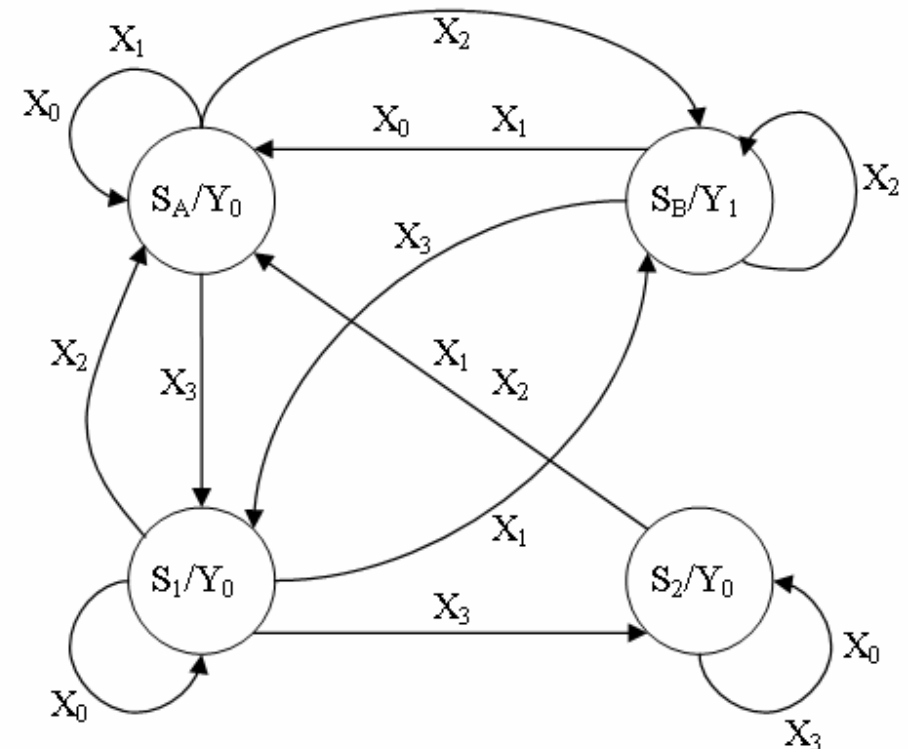




Moore stavové diagramy

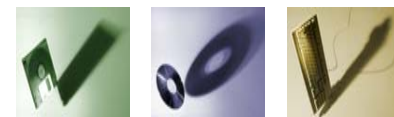
	X_0	X_1	X_2	X_3	Y
S_A	S_A	S_A	S_B	S_1	Y_0
S_B	S_A	S_A	S_B	S_1	Y_1
S_1	S_1	S_B	S_A	S_2	Y_0
S_2	S_2	S_A	S_A	S_2	Y_0

Prechodová a výstupná tabuľka



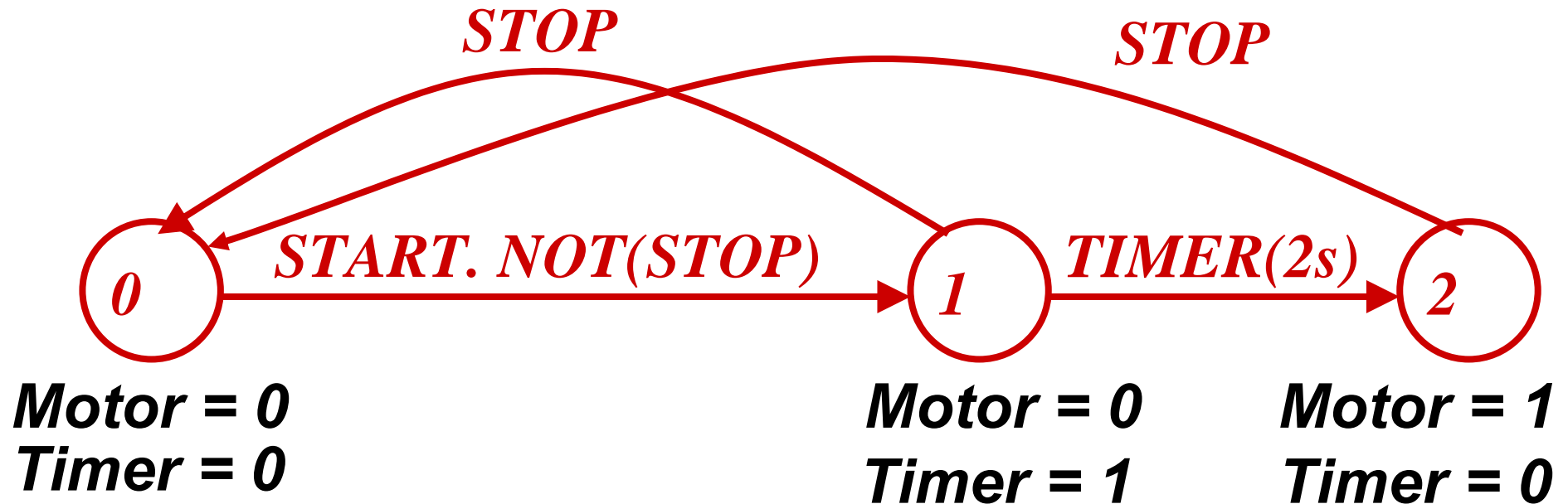
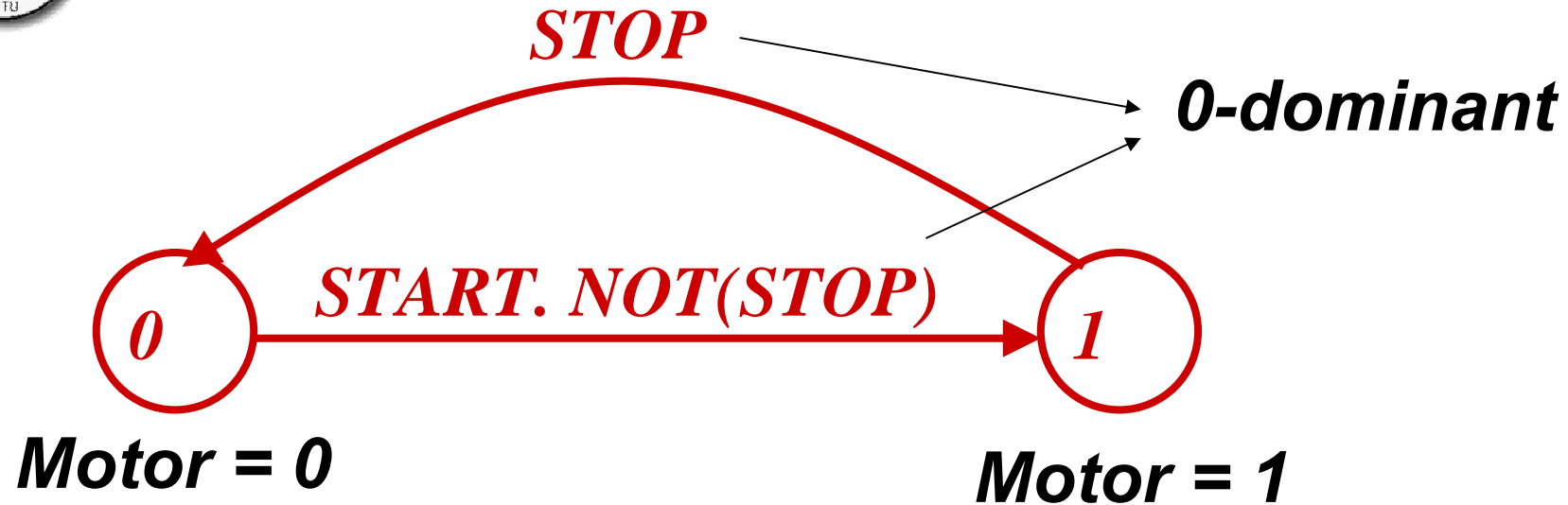


Príklad riadenie motora





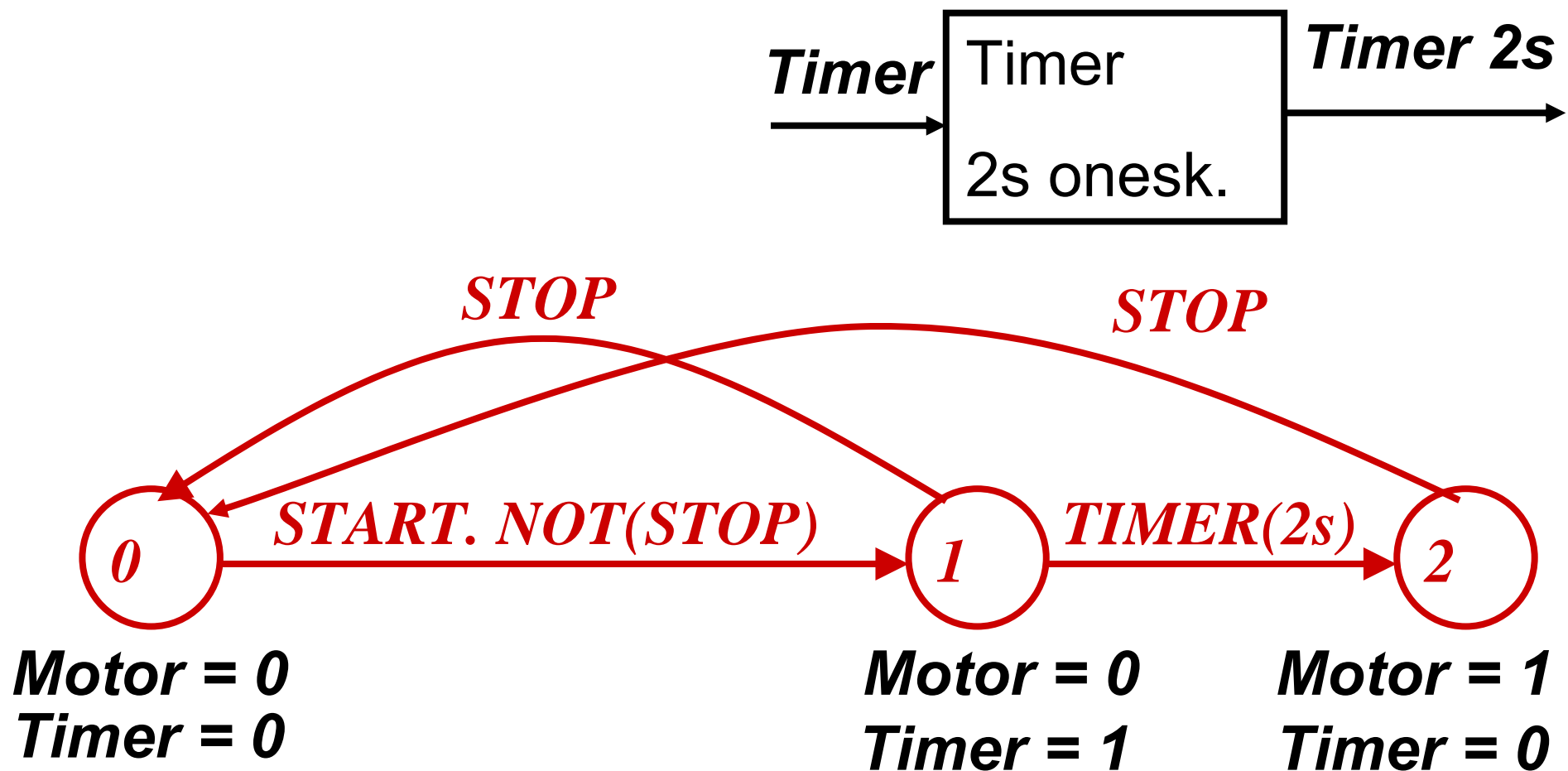
Stavový diagram riadenia motora





Stavový diagram riadenia motora

- Časovač je ovládaný logikou, stavmi
- Časové oneskorenie ovláda logiku





Moore stavový diagram riadenia motora

$X = \{START, STOP, TIMER.DN\}$

$X_0 = \{0,0,0\}, X_1 = \{0,0,1\}, X_2 = \{0,1,0\}, X_3 = \{0,1,1\}$

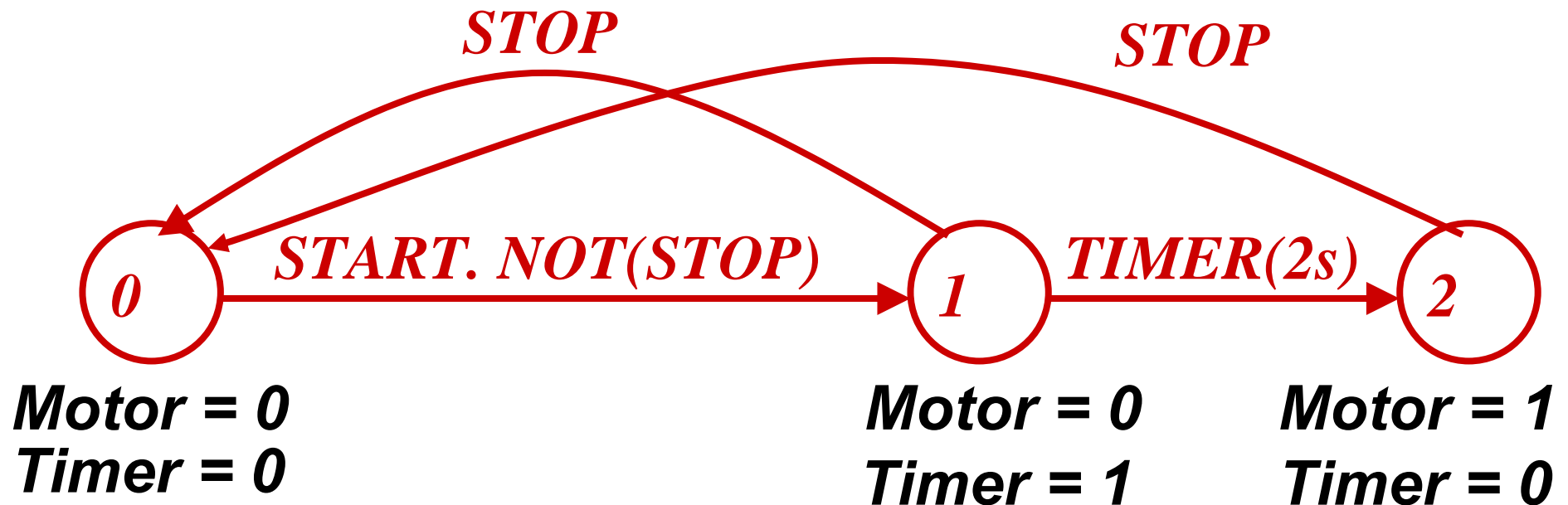
$X_4 = \{1,0,0\}, X_5 = \{1,0,1\}, X_6 = \{1,1,0\}, X_7 = \{1,1,1\}$

$Y = \{MOTOR, TIMER\}$

$Y_0 = \{0,0\}, Y_1 = \{0,1\}, Y_2 = \{1,0\}$

	X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y
S_0	S_0	S_0	S_0	S_0	S_1	S_0	S_0	S_0	Y_0
S_1	S_1	S_2	S_0	S_0	S_1	S_2	S_0	S_0	Y_1
S_2	S_2	S_2	S_0	S_0	S_2	S_2	S_0	S_0	Y_2

Prechodová a výstupná tabuľka





Moore stavový diagram riadenia motora

$X = \{START, STOP, TIMER.DN\}$

$Y = \{MOTOR, TIMER\}$

$X_0 = \{0,0,0\}, X_1 = \{0,0,1\}, X_2 = \{0,1,0\}, X_3 = \{0,1,1\}$

$Y_0 = \{0,0\}, Y_1 = \{0,1\}, Y_2 = \{1,0\}$

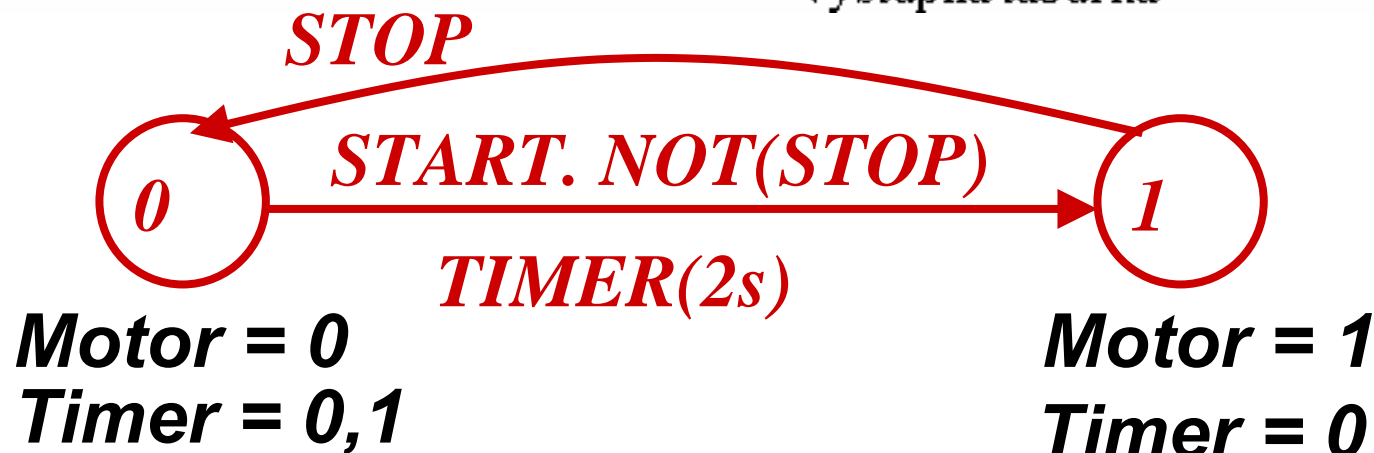
$X_4 = \{1,0,0\}, X_5 = \{1,0,1\}, X_6 = \{1,1,0\}, X_7 = \{1,1,1\}$

	X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7
S_0	S_0	S_1	S_0	S_0	S_0	S_1	S_0	S_0
S_1	S_1	S_1	S_0	S_0	S_1	S_1	S_0	S_0

Prechodová tabuľka

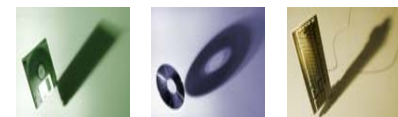
	X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7
S_0	Y_0	Y_1	Y_0	Y_1	Y_1	Y_1	Y_0	Y_1
S_1	Y_2	Y_2	Y_2	Y_2	Y_2	Y_2	Y_2	Y_2

Výstupná tabuľka





Implementácia stavových diagramov



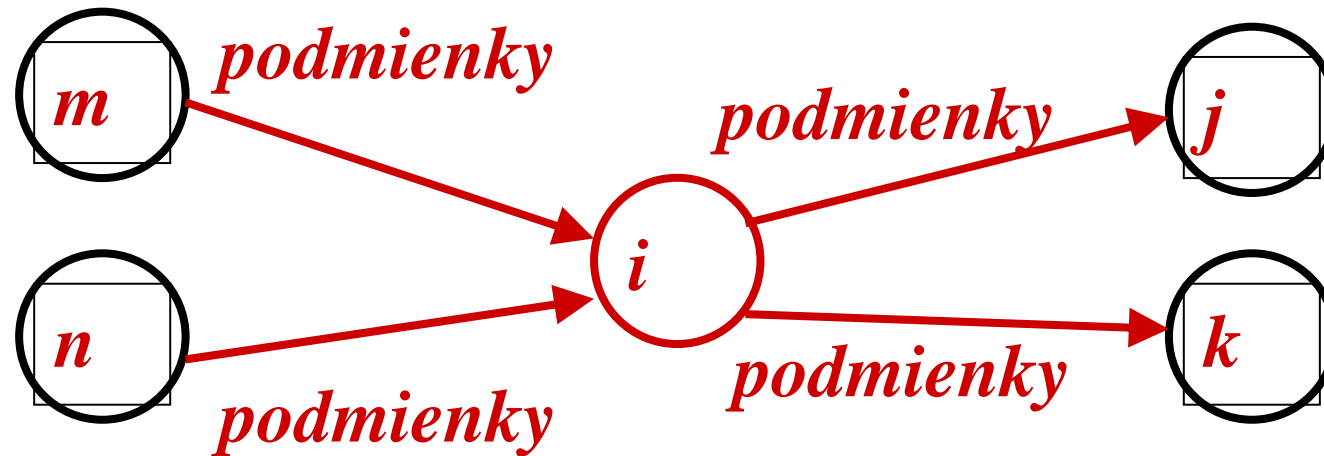


Implementácia stavových diagramov

- Implementácia do PC
- Implementácia pomocou Ladder Logic, štrukturovaný text STX
- Implementácia pomocou SFC



Generický stav S_i , diagram



**Všetky stavy,
z ktorých sa
prechádza
do S_i**

$O_i = 0$

$O_j = 1$

.....

**Všetky
výstupy**

**Všetky stavy,
do ktorých sa
prechádza
zo stavu S_i**





Generická výroková logika pre stav S_i

Pamät' pre stav S_i

Logika pre prechod z S_m do S_i

Logika pre prechod z S_n do S_i

$$***$S_i = S_i + S_m.[\dots] + \dots S_n.[\dots]$***$$

$$***$S_m = S_m. \text{Not}(S_i) \longrightarrow$ Všetky stavy z ktorých sa prechádza do S_i***$$
$$***$S_n = S_n. \text{Not}(S_i)$***$$

.....

Generická výroková logika pre výstup O_i , všetky stavy kde O_i je log.1 (true, on)

$$***$O_i = S_p + S_q + \dots$ Moore stavové diagramy***$$



Preklad stavových diagramov do výrokovovej logiky

A. Konštrukcia výrokov pre každý stav S_i

- Logika pre nastavenie daného stavu a držanie stavu S_i
- Logika pre vypnutie stavu S_i

B. Konštrukcia logiky pre každý výstup v rámci daného stavu

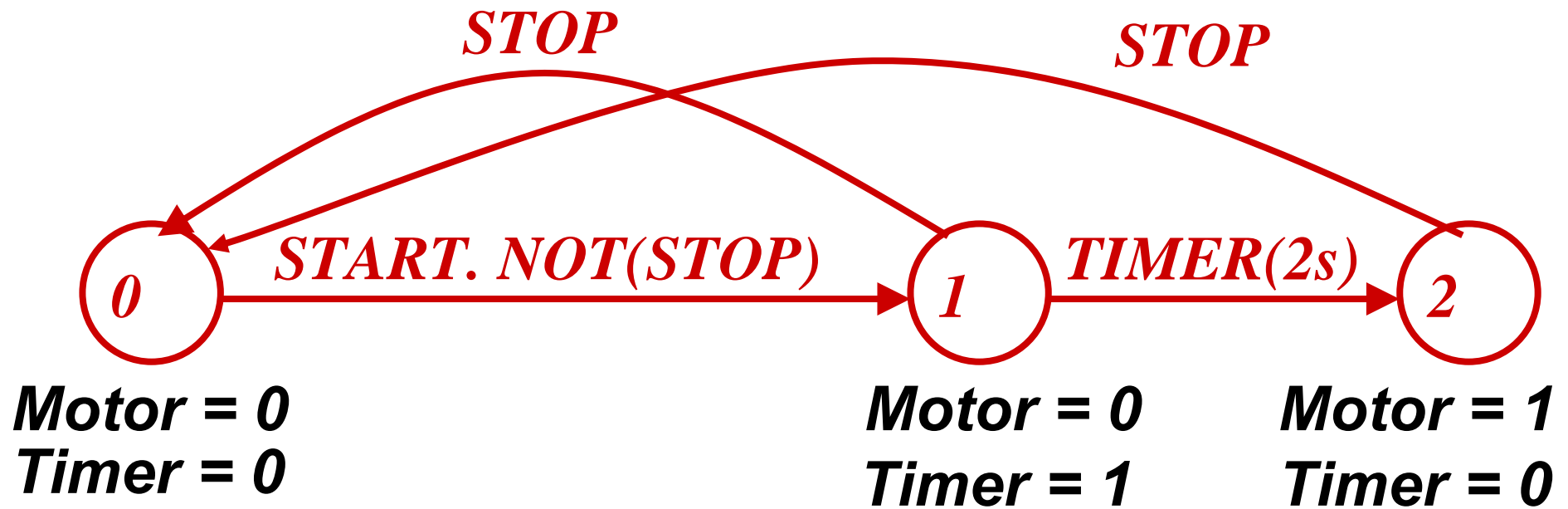


Príklad implementácia stavových diagramov do výrokovej logiky





Preklad stavového diagramu riadenia motora



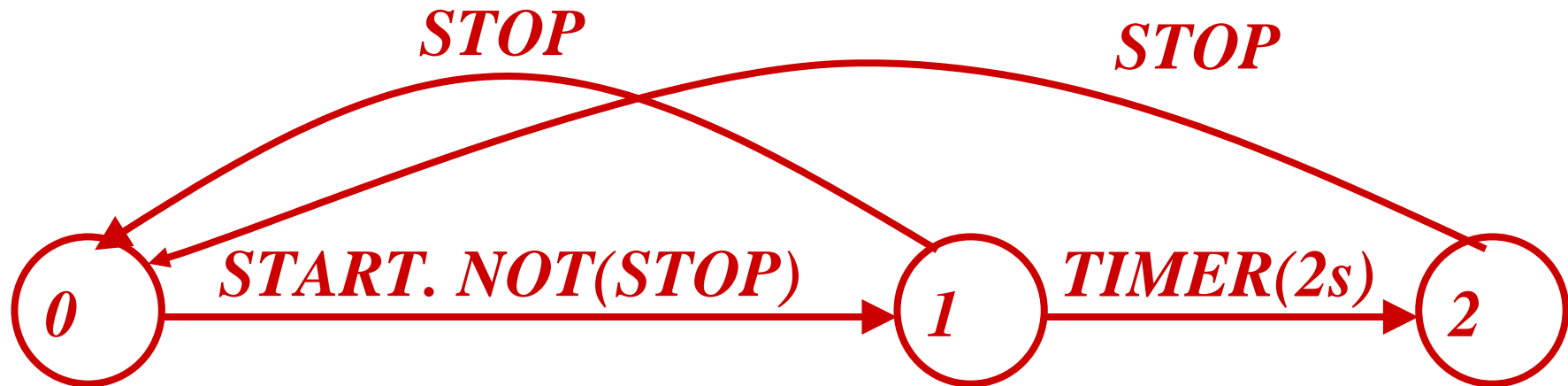
$$S0 = S0 + S1.STOP + S2.STOP$$

$$S1 = S1. Not(S0)$$

$$S2 = S2. Not(S0)$$



Preklad stavového diagramu riadenia motora



Motor = 0
Timer = 0

Motor = 0
Timer = 1

Motor = 1
Timer = 0

$S1 = S1 + S0. START.Not(STOP)$

$S0 = S0. Not(S1)$

$S2 = S2 + S1. TIMER(2s)$

$S1 = S1. Not(S2)$

$MOTOR = S2, \quad TIMER = S1$



Preklad stavového diagramu riadenia motora - pokračovanie

Stav S0:

$$S0 = S0 + S1.STOP + S2.STOP$$

$$S1 = S1. Not(S0)$$

$$S2 = S2. Not(S0)$$

Stav S1:

$$S1 = S1 + S0. START.Not(STOP)$$

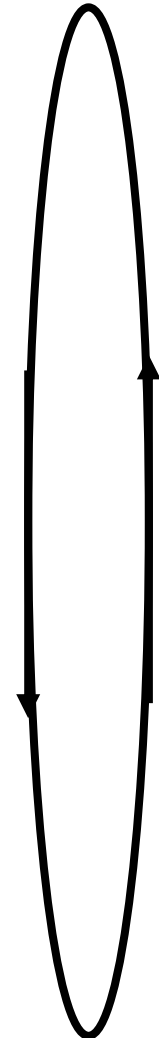
$$S0 = S0. Not(S1)$$

Stav S2:

$$S2 = S2 + S1. TIMER(2s)$$

$$S1 = S1. Not(S2)$$

Výstupy: $MOTOR = S2$, $TIMER = S1$





Diky za pozornost'

