

Obsah

4 Sekvenční logický systém	1
4.1 Sekvenční logický systém	1
4.1.1 Stavba	1
4.1.2 Návrh sekvenčního systému	1
4.2 SR Latch	1
4.3 SR latch with enable (gated SR latch)	1
4.4 D latch	2
4.4.1 Rising edge D flip-flop	2
4.5 JK latch	2
4.6 T latch	2
4.7 Flip Flop	4
4.8 Konečný automat	4

4 Sekvenční logický systém

4.1 Sekvenční logický systém

- též taky jen sekvenční obvod
- na rozdíl od kombinačních obvodů není závislý pouze na vstupních signálech ale také na pořadí zadání
- uchovávají předešlý stav – mají paměť
- při určení hodnoty výstupu nutno sledovat jak vstupy, tak stav paměti
- typy podle času změny výstupu a paměti
 - asynchronní – změna vstupu ihned ovlivní výstup; *latches*
 - synchronní – řízeny vnějším synchronizačním signálem (*clock signal*), změna probíhá pouze při tiku hodin, *flip-flops*

4.1.1 Stavba

- kombinační a paměťová část
- paměťová
 - uchování vnitřního stavu
 - základem SR latch
 - většinou složena z NAND nebo NOR gate
- kombinační část
 - pomocná kombinační logika doplňující paměť
 - využití AND či NOT gate...

4.1.2 Návrh sekvenčního systému

Mealyho typ

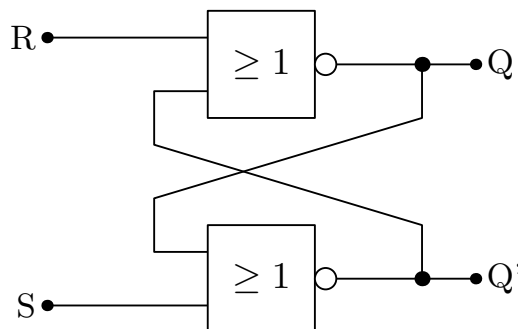
- výstup závislý na vstupu a vnitřním stavu
- rychlejší než Moor
- menší počet stavů (n vs n^2)

Moorův typ

- vstupní hodnoty se projeví až v následujícím stavu
- výstup závislý pouze na vnitřním stavu

4.2 SR Latch

- základní paměťový obvod
- sestaven z NAND nebo NOR gates
- 2 vstupy – S (set) a R (reset)
- 2 výstupy – Q a \bar{Q}
- set nastaví hodnotu Q na 1, reset na 0; opakované spouštění stejného vstupu nic nedělá
- oba vstupy zapnuté – nedefinovaný stav, latch se rychle přehazuje, konečný stav neurčitelný



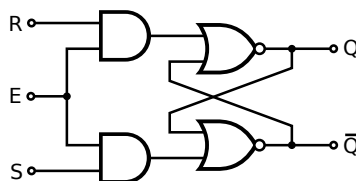
Obr. 4.1: SR latch za použití NOR gate.

S	R	Q	\bar{Q}
0	0	Last	Last
0	1	0	1
1	0	1	0
1	1	–	–

Tab. 4.1: Pravdivostní hodnoty SR latch (Last – Last state – poslední stav)

4.3 SR latch with enable (gated SR latch)

- rozšíření SR latch o kombinační obvod
- přidání *enable* vstupu – povoluje vstup
 - s $E = 0$ není možné ovlivnit latch
 - s $E = 1$ vstupy projdou a změní stav



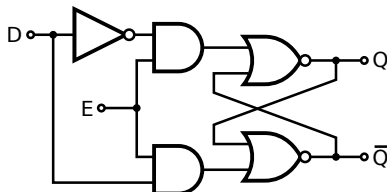
Obr. 4.2: SR latch s enable vstupem

4.4 D latch

- základem gated SR latch
- dva vstupy – E (enable) a D
- vstupy S připojeno na D , R na \bar{D}
- bez enable vstupu by Q v podstatě jen kopírovalo stav D
- s enable vstupem možnost zapamatovat si poslední stav
- D flip flop – obvod používán jako paměťový modul

D	E	Q	\overline{Q}
0	–	Last	Last
1	0	0	1
1	1	1	0

Tab. 4.2: Pravdivostní hodnoty D latch



Obr. 4.3: D latch s enable signálem

4.4.1 Rising edge D flip-flop

- uložení hodin pouze na začátku clock cyklu
- možnosti sestavení
 - master-slave – 2 D latches za sebou, u jednoho invertovaný clock signál
 - rising-edge detektor na clock signálu (viz 4.6)

4.5 JK latch

- modifikovaná gated SR latch
- zamezení nedefinovaného stavu

J	K	Q	\overline{Q}
0	0	Last	Last
0	1	0	1
1	0	1	0
1	1	Toggle	Toggle

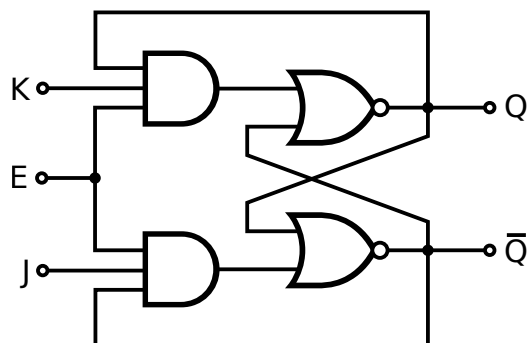
Tab. 4.3: Pravdivostní hodnoty JK latch

4.6 T latch

- modifikace JK latch
- J i K připojeny na T → stav J a K vždy stejný
 - $T = 0$ – latch si drží poslední stav
 - $T = 1$ – latch změni stav, toggles
- funguje jako přepínač

4.7 Flip Flop

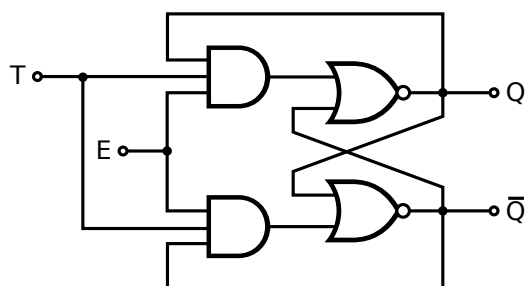
- sekvenční obvod pracující dle hodin
- připojení hodin (CLK) na enable vstup
- nutno zapsat hodnotu jednou, ne ji zapisovat po celou dobu aktivace hodin → rising nebo falling edge
 - rising edge – k rychlé aktivaci dojde při změně signálu z 0 na 1
 - falling edge – k rychlé aktivaci dojde při změně signálu z 1 na 0
 - většinou využita rising edge



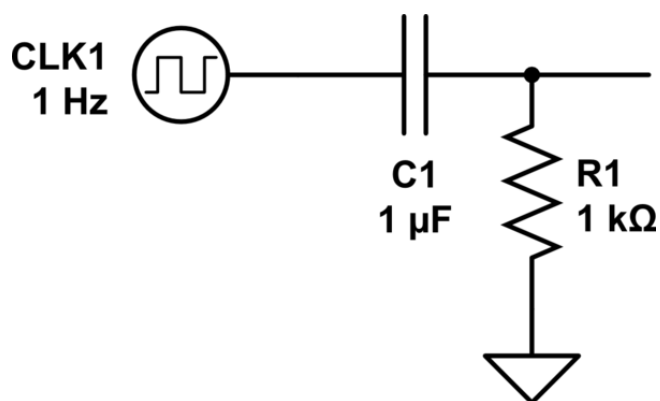
Obr. 4.4: JK latch

T	Q	\bar{Q}
0	Last	Last
1	Toggle	Toggle

Tab. 4.4: Pravdivostní hodnoty T latch

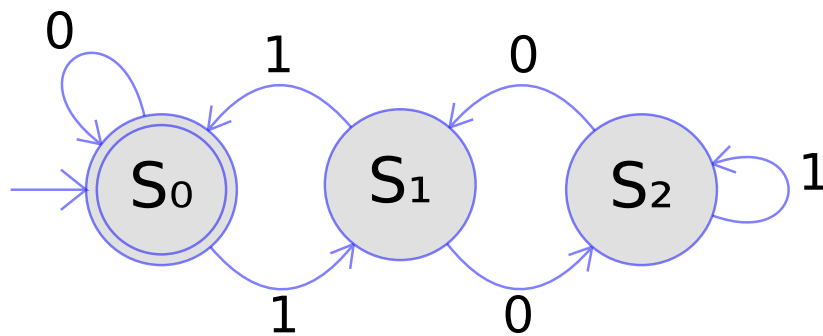


Obr. 4.5: T latch

Obr. 4.6: Obvod pro dosažení rychlého zapnutí, možno přidat diodu pro odstranění *negativních pulzů*

4.8 Konečný automat

- popis jednoduché počítače
- model systémů různých stavů
- přechod mezi stavy na základě vstupů
- vždy pouze v jednom stavu
- skladba
 - konečný počet stavů
 - konečný počet externích vstupů
 - konečný počet externích výstupů
 - explicitní specifikace přechodu mezi stavy
 - explicitní specifikace hodnoty výstupu
- části
 - logika dalšího stavu
 - registr stavu
 - * uložení dat na začátku clock cyklu
 - * data dostupná celý clock cyklus
 - logika výstupu
- příklad: výtahy, semaforey, kombinační zámky...



Obr. 4.7: Symbolický náčrtek konečného automatu