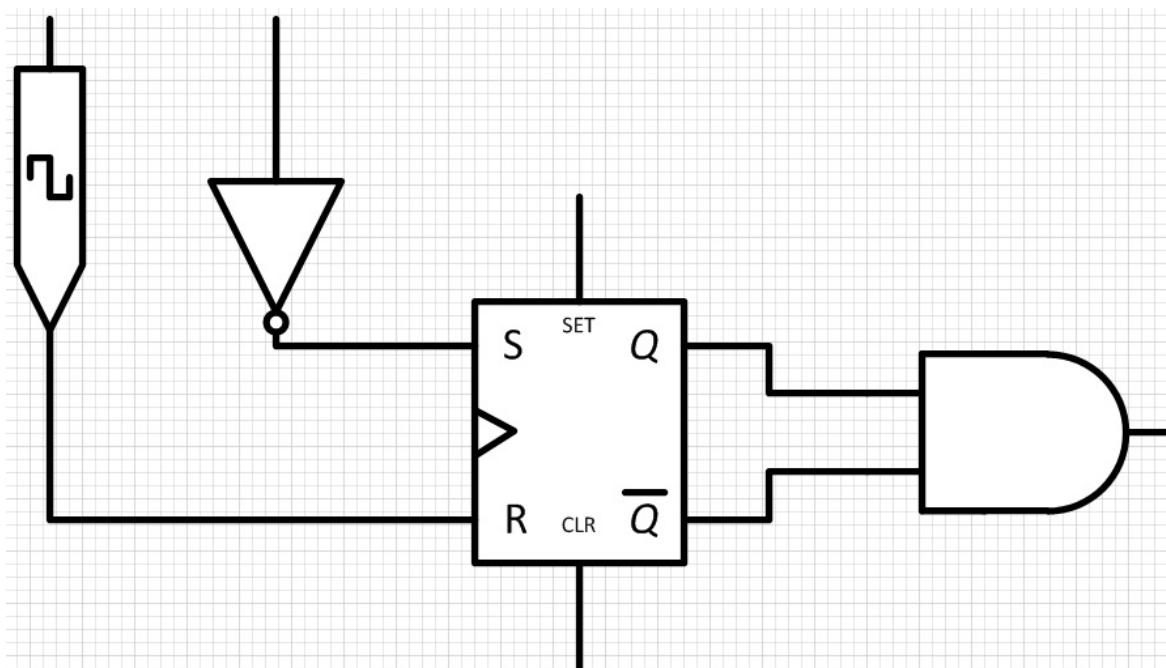


Netradiční sbírka z fyziky III

Logické obvody a jejich zapojení 3

Jakub Šafařík



Obsah

Úvod 1

I. Paměť	3
1. Postup programování.....	5
2. Světelná znamení.....	7
3. Světelná znamení se čtyřbitovým adresováním	11
4. Světelná signalizace jednotlivými diodami	13
5. Měřič reakčního času	17
6. Losovací zařízení	21
7. Hra NIM.....	23
8. Světelný nápis	27
II. Hudba	33
9. Hudba.....	33
10. Písnička	37
III. Paměťové členy v rozšířeném zapojení	39
11. Budík.....	39
12. Hudební světelné efekty.....	41
Přílohy.....	47
Paměť	47
Doporučená literatura	49

Seznam obrázků

Obr. 1.1: Paměťový člen 6116	3
Obr. 2.1: Světelná znamení	7
Obr. 3.1: Světelná signalizace se čtyřbitovým adresováním	11
Obr. 4.1: Zapojení spínačů S1–S8	13
Obr. 4.2: Světelná signalizace s jednotlivými diodami	14
Obr. 5.1: Měřič reakčního času	18
Obr. 6.1: Losovací zařízení	21
Obr. 7.1: Počty zápalek	23
Obr. 7.2: Hra NIM	25
Obr. 8.1: Světelný nápis	28
Obr. 8.2: Písmena	30
Obr. 9.1: Hudba	34
Obr. 10.1: Noty	37
Obr. 10.2: Křížek	37
Obr. 10.3: Běčko	38
Obr. 11.1: Světelný detektor	39
Obr. 12.1: Hudební světelné efekty	42
Obr. 12.2: Akustická část hudebních světelných efektů	43

Seznam tabulek

Tab. 2.1: Logické hodnoty na čítači (adresy v paměti)	9
Tab. 2.2: Svítící pole 1	9
Tab. 2.3: Svítící pole 2	10
Tab. 2.4: Svítící pole 3	10
Tab. 2.5: Svítící pole 4	10
Tab. 3.1: Počet adres v závislosti na adresových vstupech	12
Tab. 5.1: Tabulka diod – reakční čas.....	19
Tab. 7.1: Nastavení spínačů pro hru dvou hráčů.....	23
Tab. 7.2: Nastavení spínačů pro hru proti robotovi.....	24
Tab. 9.1: Tóny	35
Tab. 12.1: Hudební světelné efekty 1	44
Tab. 12.2: Hudební světelné efekty 2	45
Tab. 12.3: Hudební světelné efekty 3	45
Tab. 12.4: Hudební světelné efekty 4	46
Tab. 12.5: Hudební světelné efekty 5	46

Seznam rovnic

Rce. 5.1: Doba svitu diody	17
Rce. 5.2: Výpočet reakční doby	17

Předmluva

Dnes už si snad nikdo z nás nedokáže představit, že by žil v moderním světě bez přístrojů jako jsou mobilní telefony, MP3 přehrávače, počítače a spousta podobných elektronických zařízení. Velmi zjednodušeně jsme se pokusili vysvětlit a ověřit funkci těchto zařízení v Netradiční sbírce z fyziky I a Netradiční sbírce z fyziky II. Všechna tato zařízení, a i mnohá další, jež nemají pouze tu nejzákladnější funkci spínat a rozepínat obvody apod., potřebují nějakou součástku, která jim umožní uložit si určité informace a v případě potřeby zase tyto informace vyvolat. Takovéto části obvodu se říká paměť.

Stejně jako lidská paměť, která pracuje na principu uložení informace díky zpracování určitého podnětu a jeho následném vyvolání pokud, je k tomu dán příkaz, tak i elektronická paměť potřebuje nejprve informaci uložit v režimu zápisu a následně informaci vyvolat v režimu čtení. Tato učebnice by měla opět pomocí jednoduchých principů čtenářům objasnit, jak funguje paměť ve všech (ať už jednoduchých či složitých) zařízeních.

Tato učebnice již neobjasňuje funkce základních stavebních bloků logických obvodů a sestavování samostatných funkčních bloků obvodů jako předchozí dva díly. Text je zaměřen na využití paměti při realizaci obvodových celků. K tomuto účelu je nejvhodnější použít stavebnici VOLTÍK III, z které jsou převzata některá schémata a návrhy obvodů [5]. Není však nezbytně nutné používat danou stavebnici. Všechny uvedené úlohy je možné realizovat např. pomocí kontaktního nepájivého pole a součástek, které jsou v textu uvedeny. Předpokládáme, že čtenář má základní znalosti matematické logiky, elektrických obvodů a buď rozumí schématům zapojení z prvního a druhého dílu učebnice, nebo má tyto sbírky k dispozici.

Text je rozdělen do dvanácti různých úloh, jejichž obtížnost se postupně zvyšuje. Postupy pro sestavení a funkce jednotlivých typů obvodů jsou v učebnici podrobně vysvětleny. Na základě popsaných úloh by měl být každý schopen splnit zadané úkoly. Tyto úkoly se nachází v každé kapitole a mají ověřit znalosti a procvičit nabyté zkušenosti v zapojování funkčního celku. Pro další studium slouží seznam doporučené literatury, ve kterém naleznete informace sahající za rámec této sbírky.

Věřím, že učebnice poslouží čtenářům k pochopení principů a zapojení funkčních bloků logických obvodů. Doufám, že tak čtenáři naleznou kladný vztah k srdci techniky, která nás obklopuje na každém kroku.

Jakub Šafařík

e-mail: SafarikJ@gybot.cz

V Praze 11. 11. 2014, rev. 1.0.1

Úvod

Třetí díl sbírky netradičních úloh z fyziky rozšiřuje problematiku o využití paměťového členu. Látka přímo navazuje na předchozí díl, v němž bylo vysvětleno, jak je možné paměťové členy nejjednodušeji realizovat a jak ovlivňovat a nastavovat jejich funkci. V tomto díle sbírky již nebudeme sestavovat jednotlivé paměťové členy, ale podíváme se na jejich využití v rámci celého kompletního paměťového členu. Pokud pochopíme funkci, smysl a logiku těchto členů, jsme nadále schopni tyto poznatky rozšiřovat na mnohem složitější systémy a funkční bloky a jsme tedy opět o krouček blíže k pochopení těch nejsložitějších moderních přístrojů.

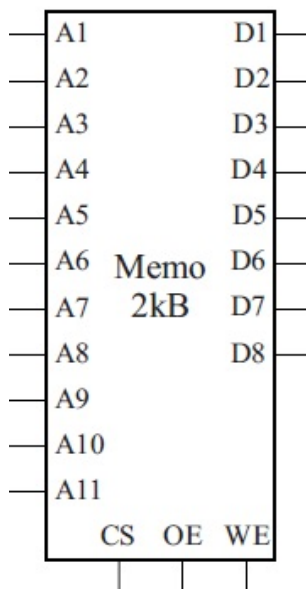
Text je rozdělen do dvanácti různých úloh, jejichž obtížnost se postupně zvyšuje. Postupy pro sestavení a funkce jednotlivých typů obvodů jsou v učebnici podrobně vysvětleny. Na základě popsaných úloh by měl být každý schopen splnit zadané úkoly. Tyto úkoly se nachází v každé kapitole a mají ověřit znalosti a procvičit nabyté zkušenosti v zapojování funkčního celku. Pro další studium slouží seznam doporučené literatury, ve které naleznete informace sahající za rámec této sbírky.

Pochopit funkci jednotlivých obvodů včetně možnosti jejich ovlivnění změnou vybraných parametrů patří k pokročilejším schopnostem technicky založeného jedince. Není to úkol jednoduchý, obzvláště v době, kdy nás výrobci složitých zařízení nutí stát se pouhými uživateli těchto přístrojů, často bez možnosti porozumět principu jejich funkce. Věřím, že odpovědi na některé otázky, které si zvědavý uživatel moderních přístrojů ohledně jejich funkce pokládá, nalezne čtenář přímo v tomto textu.

Pro práci s touto sbírkou ve školních lavicích je vhodné, aby studenti využívali tablety, pomocí nichž mohou rychle a snadnou kreslit schémata, či upravovat návrhy jednotlivých obvodů. Velmi jednoduše tak mohou své nápady pomocí připojeného projekčního zařízení prezentovat ostatním spolužákům a diskutovat o výhodách či nevýhodách jejich realizace.

I. Paměť

Paměťový člen, který máme ve stavebnici k dispozici, je typu SRAM¹ a nese označení 6116. Dané označení má svůj význam. Číslo 16 znamená, že do paměti je možné uložit informaci o maximální možné velikosti 16 Kb. Obsahuje tedy 16 384 buněk². Vždy 8 buněk je seskupeno do jednoho slova (8 bitů je 1 bajt / 8 b = 1 B). Paměť tudíž obsahuje 2 048³ slov (bajtů). Paměťový člen je uveden na Obr. 1.1.



Obr. 1.1: Paměťový člen 6116

Aby se příslušné slovo objevilo na osmi výstupech paměti (D1–D8), nebo aby mohlo být naprogramováno, musíme na adresní vstupy paměti (A1–A11) přivést příslušnou kombinaci LOG-0 a LOG-1 (adresu). D1–D8 zde fungují jako vstupy i výstupy, mezi kterými můžeme přepínat. Podle toho jaká je kombinace LOG-0, LOG-1 na A1–A11 (binární kód), taková je adresa uložené informace v paměti. Na to, abychom na adresních vstupech mohli prostřídat 2048 kombinací, je nutné připojit celkem 11 adresních vstupů, protože $2^{11} = 2048$.

¹ Zkratka SRAM znamená Static Random Access Memory, neboli statická paměť. Informace, kterou do tohoto typu paměti uložíme, zůstane v paměti zapsána do doby, než odpojíme napájecí napětí, nebo dokud nejsou data v dané buňce přepsána jinými daty. Více v kapitole Přílohy nebo [1] a [2].

² Jedna buňka umožňuje zapsat 2 stavy (LOG-0 a LOG-1), Takovýchto buněk je celkem 2^{14} , což je právě 16 384. Takovéto číslo se v technické praxi uvádí jako 16 kilobitů.

³ V technické praxi se uvádí 2 KB.

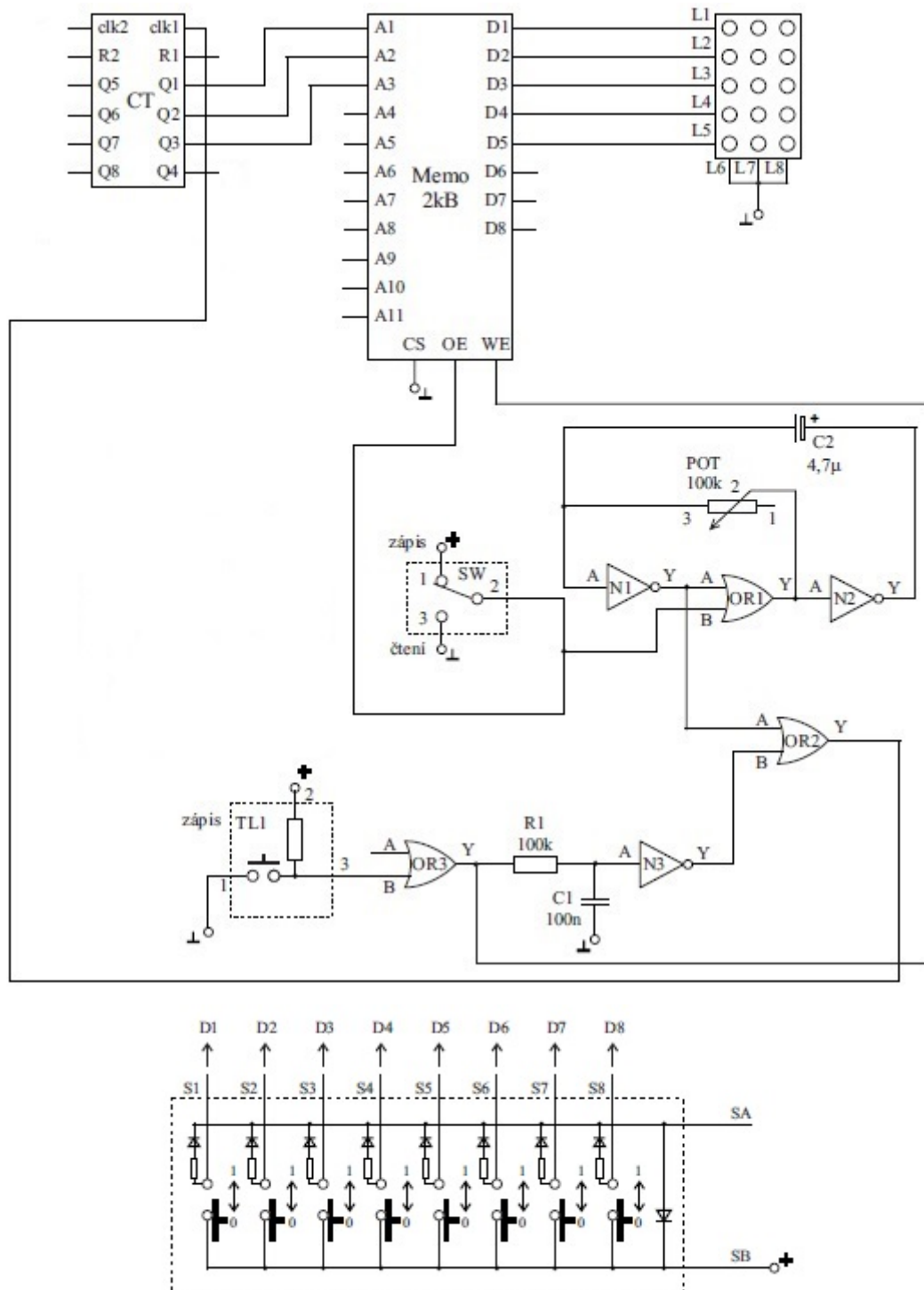
1. Postup programování

- a) Přepínač SW uvedeme do polohy „ZÁPIS“ (nahoru).
- b) Pomocí přepínačů S1–S5 nastavíme logické úrovně. Přepínač v poloze 1 znamená úroveň LOG-1 a v poloze 0 úroveň LOG-0. Připojením k diodám je možné kombinací LOG-1 (svítící dioda) a LOG-0 (nesvítící dioda) zobrazit různé varianty svítících diod (slovo).
- c) Po navolení spínačů uložíme do paměti danou kombinaci svítících diod (slovo) stiskem TL1. Na čítači se přičte 1 a posune se tak na další slovo.⁴
- d) Opakováním bodu a) a b) naprogramujeme námi požadovaný počet svítících diod. Maximální počet slov je dán počtem připojených adresních vstupů paměti (A1–A8) k výstupům čítače (Q1–Q8). Připojíme-li A1, A2, A3 ke Q1, Q2, Q3 máme k dispozici 3 bity, můžeme tedy uložit celkem 2^3 , tedy 8, slov.
- e) Pokud během programování uděláme chybu, je nutné všechna slova zapsat znovu. Čítač však není nutné nulovat, je možné začít vkládat znaky ihned.
- f) **PŘED PŘECHODEM NA ČTENÍ JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** V opačném případě se program přepíše a hrozí zničení paměti!
- g) Přepneme spínač SW do polohy „ČTENÍ“ (nahoru) a diody postupně začnou svítit tak, jak jsme je naprogramovali.

⁴ Funkce čítače je detailně popsána v předchozím díle této sbírky, viz [3] kapitola II.

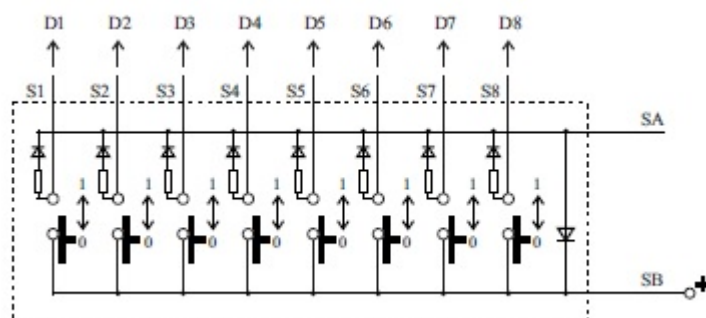
2. Světelná znamení

Pojďme nyní naprogramovat konkrétní světelnou signalizaci. Ze skutečností uvedených v předchozí kapitole je jasné, že budeme potřebovat paměť, její propojení s čítačem a polem diod. Pro jednoduchost zapojíme pět programovatelných výstupů/vstupů paměti, které budou rozsvěcet pět řádků diod. Dále pak propojíme tři výstupy čítače se třemi vstupy paměti, které se starají o adresaci uložených hodnot.



Obr. 2.1: Světelná znamení

Nyní si popíšeme vstupy a výstupy paměťového členu na



Obr. 2.1. Pomocí vstupu CS (Chip Select) paměť zapínáme a vypínáme. Je-li na CS přivedena LOG-1, je paměťový člen vypnutý, při LOG-0 se zapne. Standardně je uvnitř stavebnice Voltík III CS připojen na LOG-1, tedy vypnuto. Nyní připojíme CS na LOG-0 (zem), paměť tedy zapneme. Ze schématu je patrné, že též na vstup OE (Output Enable) můžeme přivést LOG-0, nebo LOG-1. Pokud chceme na výstupy D1–D8 (v našem případě D1–D5) přivést uložená data, je nutné na OE přivést LOG-0. V opačném případě jsou výstupy odpojeny. WE (Write Enable) se aktivuje opět po připojení na LOG-0. Z D1–D8 (v našem případě D1–D5) se stanou vstupy pro uložení hodnot do paměti. Standardně je WE připojeno na LOG-1, tedy zápis není umožněn. Je nutné si povšimnout, že spínače S1–S8 (v našem případě S1–S5) jsou též připojeny na vstupy D1–D8 (v našem případě D1–D5).

Když už víme, jak paměť funguje, pokusíme se naprogramovat světelnou signalizaci. CS je připojen na LOG-0, tudíž paměť je aktivní. Přepínač SW přesuneme do polohy 1 a umožníme tak zápis do paměti (blokuje čtení). Pomocí spínačů S1–S5 nastavíme libovolnou kombinaci svítících diod. V našem zapojení nastavujeme tedy libovolnou kombinaci svítících řádků. Po nastavení stiskneme tlačítko TL1, tím na WE přivedeme LOG-0 a stavy na vstupech se uloží do paměti. Zároveň přivedeme na hodinový vstup čítače impuls, který se projeví na výstupech čítače (Q1–Q3) jako kombinace LOG-1 a LOG-0, která adresuje uložené stavy v paměti. Takto můžeme obvodem na Obr. 2.1 naprogramovat celkem osm stavů.

Pokud jsou stavy naprogramovány, přesvědčíme se, že jsou v paměti uloženy jejich vyvoláním. **PŘEPNEME SPÍNAČE S1–S5 DO POLOHY LOG-0.** Přepínač SW nastavíme do polohy 3 (čtení). Nyní by se měly diody rozsvěcet tak, jak jsme je naprogramovali. Postupné rozsvěcení je umožněno tím, že na vstup čítače clk1 jsou přivedeny impulzy, jejichž délku je možné regulovat potenciometrem POT. Logické hodnoty na vstupech Q1–Q3, tedy hodnoty na adresních vstupech paměti A1–A3 jsou uvedeny v následující tabulce – Tab. 2.1.

Impulz	Q3/A3	Q2/A2	Q1/A1
0.	0	0	0
1.	0	0	1
2.	0	1	0
3.	0	1	1

Impulz	Q3/A3	Q2/A2	Q1/A1
4.	1	0	0
5.	1	0	1
6.	1	1	0
7.	1	1	1

Tab. 2.1: Logické hodnoty na čítači (adresy v paměti)

Úkol

1. Realizujte kombinaci svítících diod dle Tab. 2.2. Křížek znamená, že daný spínač je sepnut (LOG-1), prázdné okénko znamená, že je rozepnut (LOG-0). Nezapomeňte, že svítí vždy celý řádek!

	Spínač, který bude sepnut				
Znak	S1	S2	S3	S4	S5
0.	X				
1.		X			
2.			X		
3.				X	
4.					X
5.				X	
6.			X		
7.		X			

Tab. 2.2: Svítící pole 1

2. Realizujte kombinaci svítících diod dle Tab. 2.2. Před spuštěním zkuste určit, jak se budou diody postupně rozsvěcovat a zhasínat.

	Spínač, který bude sepnut				
Znak	S1	S2	S3	S4	S5
0.			X		
1.		X		X	
2.	X				X
3.	X				X
4.	X				X
5.	X				X
6.		X		X	
7.			X		

Tab. 2.3: Svítící pole 2

3. Realizujte kombinaci svítících diod tak, aby se řádky rozsvěcely v pořadí 2. a 4. řádek; 3. a 1. a 5. řádek; všechny řádky; žádný řádek; 1. řádek; 1. a 2. řádek; 1. a 2. a 3. řádek; 1. a 2. a 3. a 4. řádek. Zaneste do tabulky – Tab. 2.4.

	Spínač, který bude sepnut				
Znak	S1	S2	S3	S4	S5
0.					
1.					
2.					
3.					
4.					
5.					
6.					
7.					

Tab. 2.4: Svítící pole 3

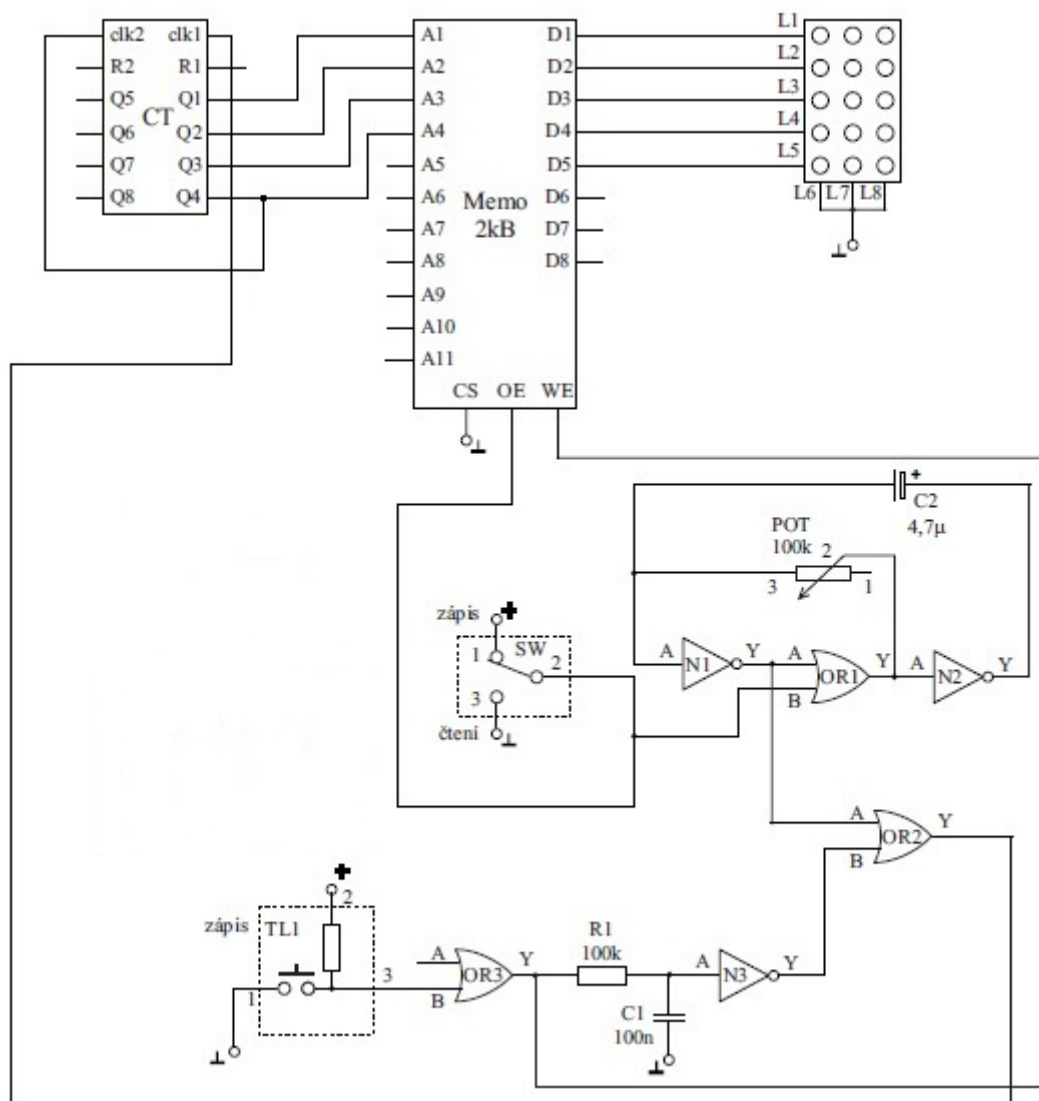
4. Vymyslete slastní schéma postupného rozsvěcení řádků diod, zaneste do Tab. 2.5 a realizujte.

	Spínač, který bude sepnut				
Znak	S1	S2	S3	S4	S5
0.					
1.					
2.					
3.					
4.					
5.					
6.					
7.					

Tab. 2.5: Svítící pole 4

3. Světelná znamení se čtyřbitovým adresováním

V předchozí úloze jsme si ukázali, jakým způsobem můžeme do paměti ukládat pětibitová slova (výstupy D1–D5). Víme, že pomocí tří adresovacích bitů jich můžeme uložit celkem osm variant. Pokud bychom chtěli ukládat větší množství slov, potřebujeme k tomu větší počet adres. Následující schéma na Obr. 3.1 ukazuje, jak rozšířit množství adres na dvojnásobek.



Obr. 3.1: Světelná signalizace se čtyřbitovým adresováním

Připojením čtvrtého výstupu čítače Q4 na adresní vstup paměti A4 jsme vytvořili čtyřbitové adresování, viz Tab. 3.1. Paměť může nyní uložit 2^4 slov, tedy celkem 16 slov. Analogicky připojením dalšího adresního vstupu paměti rozšíříme počet adres umožňujících uložení slov opět na dvojnásobek atd.

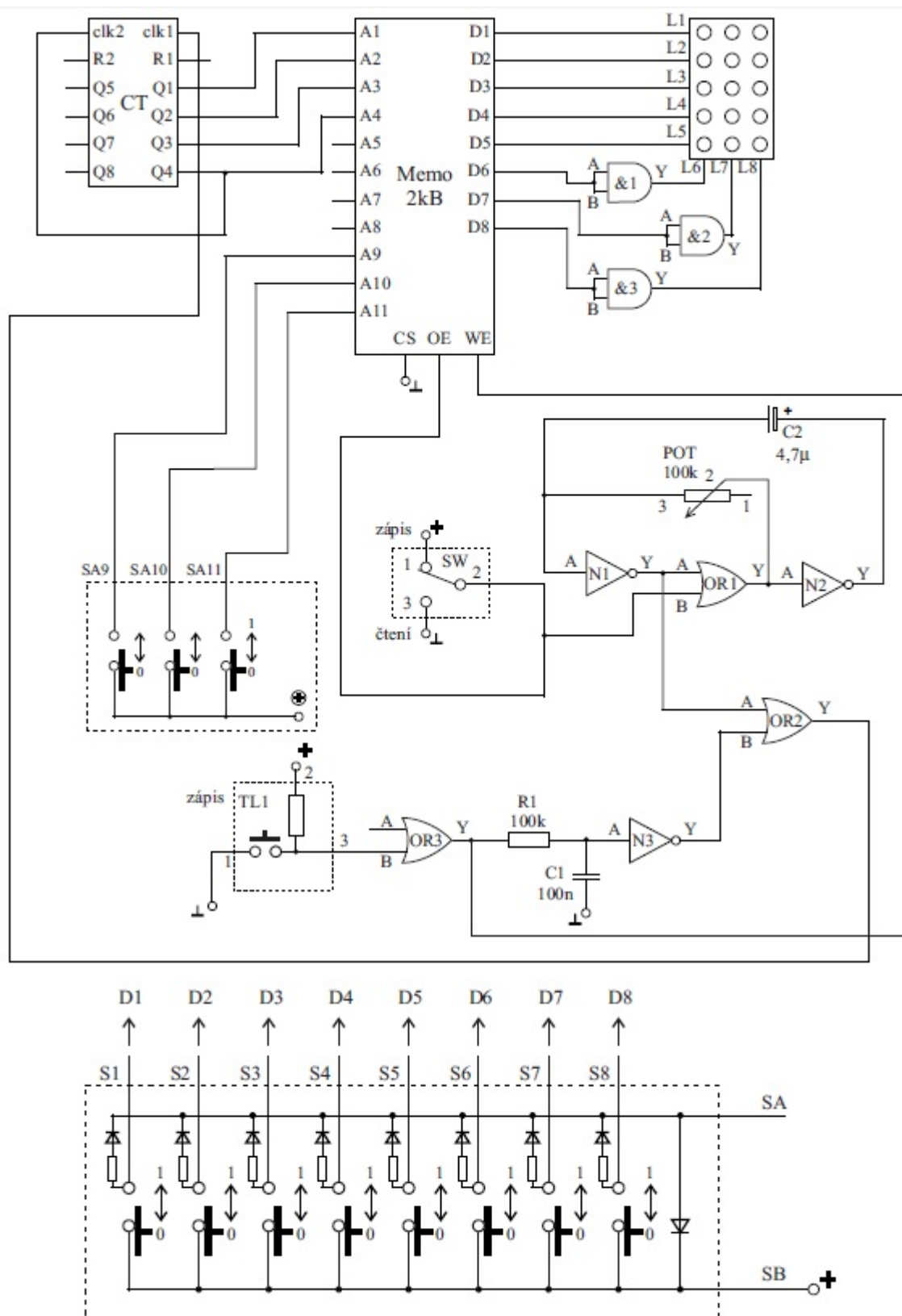
Počet adres	Počet adresních vstupů			
1	1	2	3	4
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				

Tab. 3.1: Počet adres v závislosti na adresových vstupech

Nyní zbývá opět pouze vyzkoušet funkci daného zapojení. Tentokrát můžeme uložit 16 kombinací pětibitových slov. Postup při vyvolání uložené informace je stejný jako v předchozí kapitole. **NEZAPOMEŇTE PŘEPNOUT SPÍNAČE S1–S5 DO POLOHY LOG-0!**

Úkol

1. Jaký největší počet jednotlivých buněk paměti bude obsahovat informace při použití pětibitových slov (D1–D5) a čtyřbitového adresování?
2. Jakou maximální délku mohou mít jednotlivá slova ukládaná do paměti? Kolik je to znaků, kolik bitů a kolik bajtů? Jaký maximální počet adres můžeme použít pro ukládání slov, kolik je to bitů? Odpověď zdůvodněte.



Obr. 4.2: Světelná signalizace s jednotlivými diodami

Úkol

1. Vyberte jednu diodu, kterou budete chtít rozsvítit, určete jednotlivé logické úrovně, které je nutné přivést na vstupy paměti D1–D8 a ověřte zapojením.
2. Naprogramujte do paměti postupné rozsvěcování diod, nejprve se rozsvítí první dioda, poté spolu s ní druhá, následně se k nim přidá třetí atd. Při programování postupujte dle návodu uvedeného v kapitole 1. Určitě nezapomeňte – **PŘED PŘECHODEM NA ČTENÍ JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**
3. Naprogramujte do paměti postupné rozsvěcování a zhasínání diod, rozsvítí se jedna dioda a poté zhasne, takto se vystřídají všechny diody. Kolik adres v paměti budeme potřebovat? Kolikabitový čítač musíme připojit k adresovým vstupům paměti? Při programování postupujte dle postupu uvedeného v kapitole 1. A nezapomeňte – **PŘED PŘECHODEM NA ČTENÍ JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**
4. Naprogramujte úlohy 2 a 3 současně. Uložte každou zvlášť (pomocí kombinace S9–S11). Při programování postupujte dle postupu uvedeného v kapitole 1. A nezapomeňte – **PŘED PŘECHODEM NA ČTENÍ JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

5. Měřič reakčního času

Následující úloha nám umožní porovnat si reakční časy se spolužáky. V momentě, kdy se rozsvítí všechny diody, bude nutné stisknout tlačítko. Lidský organismus na světelný podnět není schopen reagovat okamžitě, a tak od rozsvícení diod do stisku tlačítka uplyne určitá doba – reakční doba. Po rozsvícení celého pole diod se dále postupně rozsvěcují jednotlivé diody, až do stisku tlačítka. Určit reakční interval bude možné z toho, která dioda zůstala svítit. Schéma pro takovýto obvod je na Obr. 5.1.

Všechny diody budou svítit jen velice krátkou dobu, následně se pak budou postupně rozsvěcovat jednotlivé diody po sloupcích. Stiskem a držením TL2 se zastaví čítač a zůstane svítit jen jedna dioda. S klesajícími řádky a sloupci posouvajícími se postupně směrem doprava roste reakční čas. Po puštění TL2 se opět čítač rozběhne.

Pokud bude vše správně zapojeno, přibližně každé 3 sekundy se rozsvítí celé pole diod a následně problikají ostatní diody. Budeme-li chtít určit svůj reakční čas přesně, je nutné spočítat, kolikrát se všechny diody rozsvítí během časového intervalu, který změříme na stopkách (doporučen je interval 120 s). Doba svitu každé jedné diody je pak:

$$t_1 = \frac{T_s}{N \cdot 128}$$

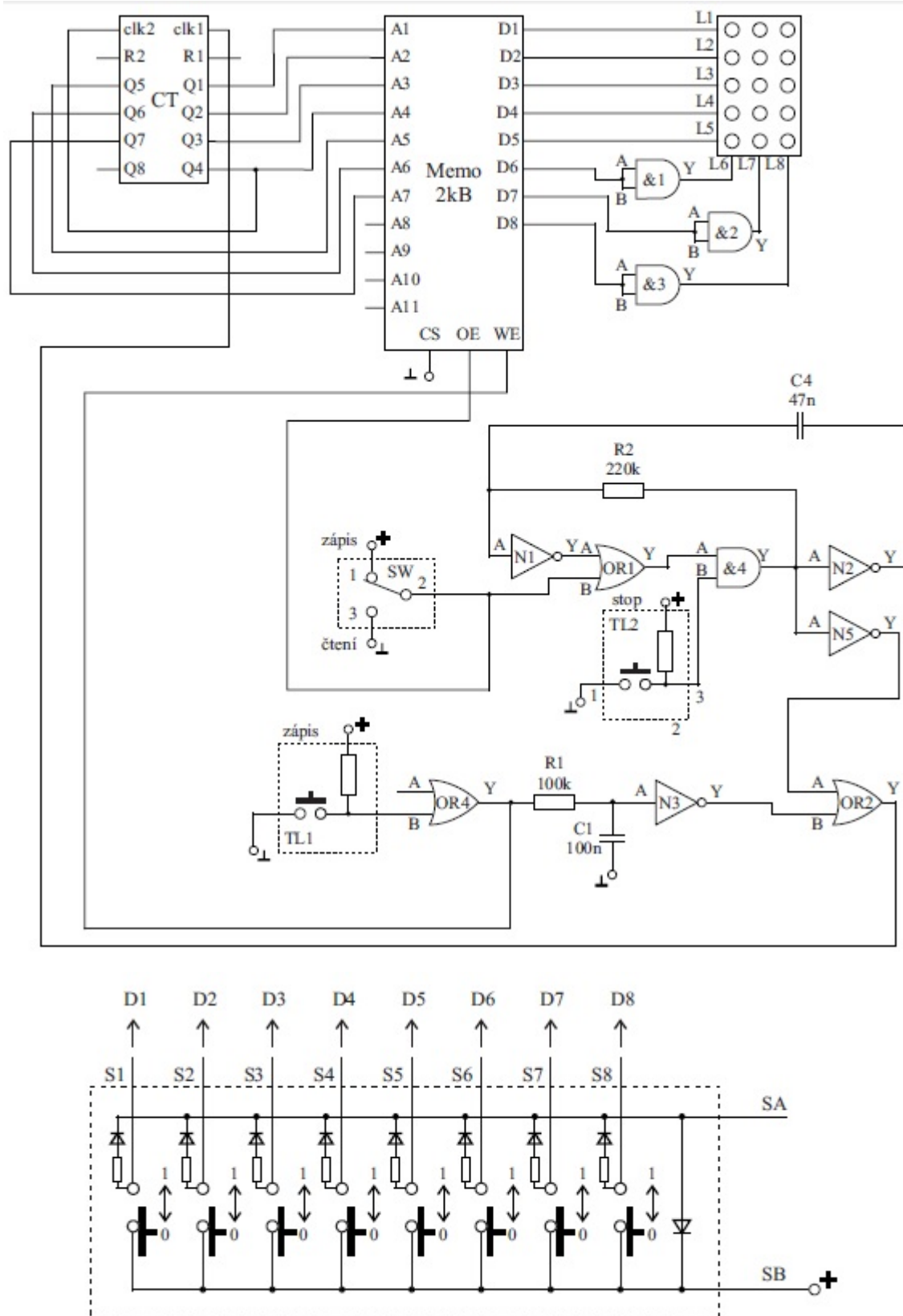
Rce. 5.1: Doba svitu diody

T_s značí dobu naměřenou na stopkách a N kolikrát se během této doby rozsvítily všechny diody, 128 značí počet uplynulých impulsů čítače (během tohoto počtu impulsů se postupně rozsvítily všechny diody celého pole). Pokud po stisknutí tlačítka zůstane svítit 5. dioda, je reakční čas:

$$t = 5 \cdot t_1 + 2 \cdot t_1$$

Rce. 5.2: Výpočet reakční doby

Druhý člen v součtu značí dobu, po kterou svítí všechny diody (dva takty čítače).



Obr. 5.1: Měřič reakčního času

Nakonec je nutné naprogramovat do paměti rozsvěcení jednotlivých diod. Jak už bylo uvedeno, všechny diody budou svítit dva takty čítače. Programování jednotlivých diod je názorně vyznačeno v Tab. 5.1.

	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.	X	X	X	X	X			
1.	X	X	X	X	X			
2.	X						X	X
3.		X					X	X
4.			X				X	X
5.				X			X	X
6.					X		X	X
7.	X					X		X
8.		X				X		X
9.			X			X		X
10.				X		X		X
11.					X	X		X
12.	X					X	X	
13.		X				X	X	
14.			X			X	X	
15.				X		X	X	
16.					X	X	X	
17. až 127.								

Tab. 5.1: Tabulka diod – reakční čas

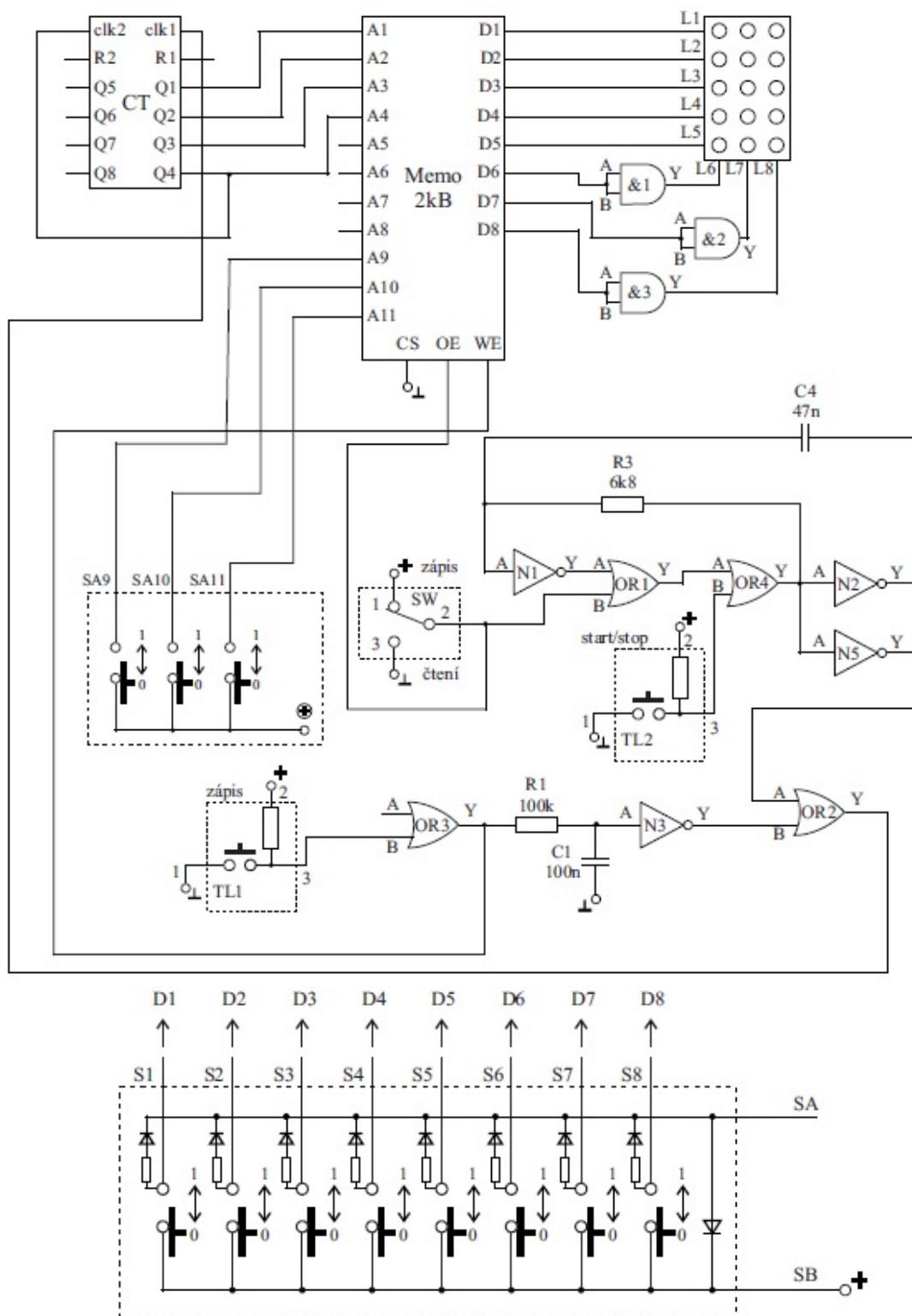
Postup programování je popsán v kapitole 1, na začátku programování nesmíme zapomenout paměť vymazat. Přepínač SW přepneme do polohy „ZÁPIS“. Všechny spínače S1–S8 přepneme do polohy „0“ a zmáčkne alespoň 128krát TL1, tím do všech adres paměti zapíšeme LOG-0. Do paměti naprogramujeme svícení jednotlivých diod dle Tab. 5.1. Na 17.–127. pozici jsou samé nuly, ty už zadávat nemusíme, neboť jsme v předchozích krocích již zbylou paměť vynulovali. Před přepnutím spínače SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** Následně přepneme SW do polohy „ČTENÍ“.

Úkol

1. Spočtete co nejpřesněji dobu svitu jedné diody dle Rce. 5.1. Čas měření volte vhodně (3–5 min). Opakujte několikrát a výsledek zprůměrujte.
2. Změřte svůj reakční čas a vypočtete jeho skutečnou hodnotu dle Rce. 5.2

6. Losovací zařízení

V následující úloze sestavíme losovací zařízení, které během stisku tlačítka losuje (probíhají zadané sekvence znaků) a při uvolnění tlačítka se objeví svítící znak. Schéma obvodu je na Obr. 6.1.



Obr. 6.1: Losovací zařízení

Schéma se od předchozího (Obr. 5.1) liší použitím jen čtyř adresových vstupů paměti v kombinaci s přepínáním paměťových bloků (S9–S11). K dispozici máme nyní jen šestnáct adres pro uložení šestnácti kombinací svítících diod, ale můžeme je uložit do osmi samostatných bloků, mezi nimiž přepínáme pomocí S8–S11. Při bližším prozkoumání si lze všimnout, že i některé další součástky byly zaměněny. Odpor R2 byl vyměněn za R3, což prodlouží časové impulzy jdoucí do čítače. Člen AND4 byl vyměněn za OR4, smysl této výměny je následující – při zapojeném členu AND stiskem tlačítka TL2 zastavíme čtení z paměti, kdežto při zapojení přes člen OR stiskem TL2 čtení z paměti spustíme.

Stejně jako v předchozí úloze je nutné naprogramovat pořadí svícení diod a jejich kombinaci do paměti. Postup programování je podrobně popsán v kapitole 1. Přepínač SW přepneme do polohy „ZÁPIS“. Nastavením spínačů S1–S8 navolíme na diodách požadovaný znak a ten pak stiskem TL1 uložíme do paměti. Takto navolíme všech šestnáct znaků. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** Následně přepneme SW do polohy „ČTENÍ“.

Uložení hodnoty v paměti je možné zkontrolovat tak, že odpojíme vodič od zdířky WE a opakovaným stiskem TL1 ověříme všechny zapsané znaky. Následně opět zasuneme vodič do zdířky WE.

Po stisku TL2 by mělo probíhat losování. Budou se postupně rozsvěcet všechny naprogramované kombinace diod. Při puštění TL2 se zobrazí vybraný znak.

Úkol

1. Jaká typy symbolů lze naprogramovat a proč? Můžeme naprogramovat jednotlivá čísla nebo písmena?
2. Naprogramujte do paměti pod jednotlivé adresy libovolné symboly a ověřte funkci losovacího zařízení.

7. Hra NIM

Nyní si ukážeme, jak můžeme předchozí zkušenosti využít k vytvoření hry. Princip hry NIM spočívá v tom, že hráči mají před sebou hromádku zápalek a z ní střídavě odebírají jednu až tři zápalky, ten kdo vezme poslední, vyhrál. Ve stavebnici Voltík III máme patnáct diod, které budou signalizovat počet zbývajících zápalek na hromádce. Každá dioda signalizuje daný počet zbývajících zápalek, viz Obr. 7.1. Na výběr máme dvě možnosti hry – buď proti druhému hráči, nebo proti robotovi.

15	10	5
14	9	4
13	8	3
12	7	2
11	6	1

Obr. 7.1: Počty zápalek

Chceme-li naprogramovat hru pro dva hráče, postupujeme následovně. Přepínač SW přepneme do polohy „ZÁPIS“. Spínač S9 přepneme do polohy 0. Nastavením spínačů S1–S8 navolíme na diodách požadovaný znak a ten pak stiskem TL1 uložíme do paměti. Postup programování je znázorněn v Tab. 7.1.

Znak	Spínač, který bude sepnut							
	S1	S2	S3	S4	S5	S6	S7	S8
0.	X						X	X
1.		X					X	X
2.			X				X	X
3.				X			X	X
4.					X		X	X
5.	X					X		X
6.		X				X		X
7.			X			X		X
8.				X		X		X
9.					X	X		X
10.	X					X	X	
11.		X				X	X	
12.			X			X	X	
13.				X		X	X	
14.					X	X	X	
15.								

Tab. 7.1: Nastavení spínačů pro hru dvou hráčů

Takto navolíme všech šestnáct znaků. Než přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** Následně přepneme SW do polohy „ČTENÍ“.

Nyní máme hru naprogramovanou a měla by svítit dioda vlevo nahoře, která signalizuje, že je na hromádce patnáct zápalek. První hráč nyní může stisknout jednou až třikrát TL1 (odebere jednu až tři zápalky), následně druhý hráč má na výběr stejné tři možnosti. Hráči se postupně střídají, dokud jeden z nich nezhasne poslední diodu (neodebere poslední zápalku) – tento hráč prohrává.

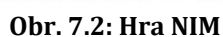
Chcete-li hrát proti robotovi, musíme do paměti krom sekvence v Tab. 7.1 naprogramovat ještě druhou sekvenci, která je uvedena v Tab. 7.2. Po naprogramování první sekvence přepneme spínač SA9 do polohy 1. Je nutné, abychom programovali sekvence synchronně. Tedy abychom naprogramovali nultý znak druhé sekvence do paměti tak, aby v daný okamžik byl v první sekvenci také nultý znak. Tuto skutečnost ověříme následovně. Po naprogramování posledního patnáctého znaku druhé sekvence (stisku TL1) čítač opět přechází na nultý znak. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** Následně přepneme SW do polohy „ČTENÍ“ a SA9 do polohy 0, nyní by měla svítit dioda vlevo nahoře, která udává plný počet zápalek na hromádce (patnáct).

	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.				X			X	X
1.				X			X	X
2.				X			X	X
3.					X		X	X
4.			X			X		X
5.			X			X		X
6.			X			X		X
7.					X	X		X
8.		X				X	X	
9.		X				X	X	
10.		X				X	X	
11.			X			X	X	
12.								
13.								
14.								
15.								

Tab. 7.2: Nastavení spínačů pro hru proti robotovi

Nyní už zbývá pouze vybrat, jestli začne hráč, nebo robot. V případě že začíná robot, stiskneme TL2 a robot si vybere, kolik zápalek sebere. V případě že začíná hráč,

Na následujícím obrázku. je uvedeno schéma (Obr. 7.2) obvodu pro hru NIM. Schéma je opět variací zapojení paměti z předchozích úloh.



Úkol

1. Naprogramujte hru NIM pro dva hráče dle postupu výše, ověřte zapojení a hru hrajte a vyhrajte.
2. Naprogramujte hru NIM pro hráče a robota dle postupu výše, ověřte zapojení a hru hrajte a vyhrajte.
3. Zkuste upravit program hry proti robotovi tak, aby ten, kdo sebere poslední zápalku, prohrál.

8. Světelný nápis

Vše co jsme se naučili v předchozích kapitolách, nyní využijeme k naprogramování světelných nápisů. Do paměťových adres budeme ukládat jednotlivá písmena. Po naprogramování budeme moci nechat pomocí daných písmen zobrazovat jednotlivá slova. Pro realizaci takového obvodu nemůžeme použít schéma Obr. 4.2, protože by tak jednotlivá písmena nešla zobrazit, schéma doplníme ještě o další část. Uděláme však obměnu a jednotlivé znaky budeme skládat ze dvou, které následují v rychlém sledu za sebou (první znak a druhý znak). Frekvence blikání musí být dostatečně velká, aby oko dané zobrazené dva symboly vnímalo jako jeden – o to se nám postará obvod 555⁷. Je tedy možné v první době nechat svítit některé diody a v druhé době buď stejné (silný svit), nebo jiné (díky poloviční době zobrazení oko vnímá jas jako poloviční). Použité schéma je na Obr. 8.1.

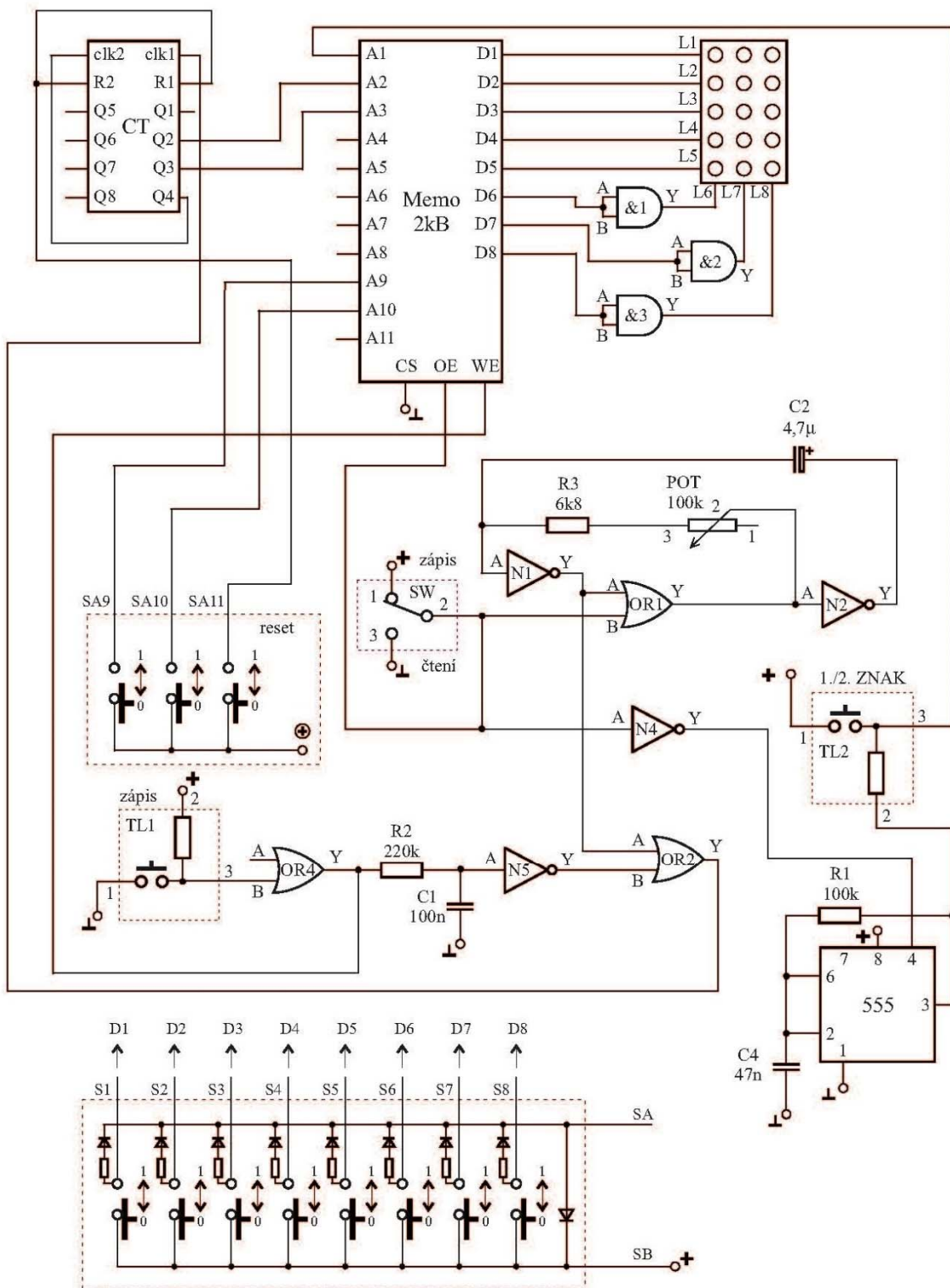
Délku slov můžeme přímo ovlivnit tím, kolik výstupů čítače (Q2–Q8) připojíme na jednotlivé adresové vstupy. Můžeme tedy vytvářet posloupnosti 2, 4, 8 až 128⁸ znaků. Pomocí spínačů SA9 a SA10 můžeme přepínat mezi jednotlivými bloky paměti, takže můžeme vytvořit až čtyři posloupnosti světelných nápisů. SA11 slouží jako RESET čítače – písmena se zobrazují opět od začátku. Obvod 555, který je připojen na adresový vstup paměti A1 se stará o přepínání mezi prvním a druhým znakem paměti v témže cyklu čítače.

Programujeme následovně – spínač SW přepneme do polohy ZÁPIS (nahoru). Vynulujeme čítač (SA11 přesuneme do polohy 1 a pak zpět do polohy 0). Nyní nastavíme první část prvního písmene dle prvního řádku tabulky pro spínače S1–S8 na Obr. 8.2. Stiskneme TL2⁹ a držíme, zároveň krátce stiskneme TL1, dále pustíme i TL2. První část prvního znaku je naprogramována. Druhou část prvního písmene nastavíme tak, že opět zvolíme vhodnou kombinaci přepínačů S1–S8, dle druhého řádku pro dané písmeno na Obr. 8.2. Stiskneme TL1 a symboly jsou uloženy. Při ukládání druhé části symbolu se TL2 již nemačká. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** Následně přepneme SW do polohy „ČTENÍ“.

⁷ Princip a funkce obvodu 555 je detailně popsána v [3].

⁸ Nejdelší sekvenci 128 znaků zobrazíme, pokud připojíme sedm vstupů (2⁷).

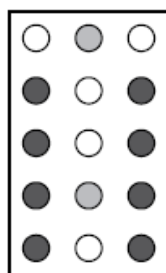
⁹ TL2 přepíná hodnoty LOG-0 a LOG-1 na nejnižším adresním bitu paměti A1.



Obr. 8.1: Světelný nápis

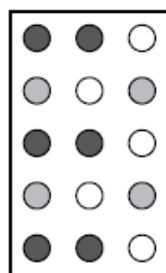
Námět pro rozsvícení jednotlivých písmenek je na Obr. 8.2. Jinak si samozřejmě každý může vymyslet vlastní variantu i s tím, které diody budou svítit slaběji (poloviční čas) a které silněji (oba dva intervaly na adrese paměti A1).

"A"



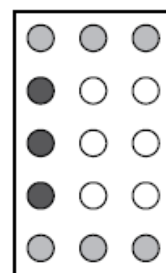
	S1	S2	S3	S4	S5	S6	S7	S8
1		•	•	•	•		•	
2	•			•		•		•

"B"



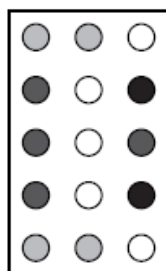
	S1	S2	S3	S4	S5	S6	S7	S8
1	•		•		•			•
2		•		•			•	

"C"



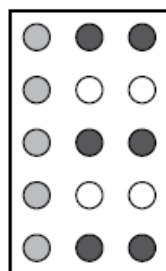
	S1	S2	S3	S4	S5	S6	S7	S8
1		•	•	•			•	•
2	•				•			

"D"



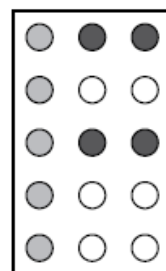
	S1	S2	S3	S4	S5	S6	S7	S8
1		•	•	•			•	
2	•				•			•

"E"



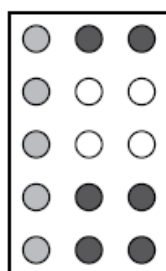
	S1	S2	S3	S4	S5	S6	S7	S8
1	•		•		•	•		
2	•	•	•	•	•		•	•

"F"



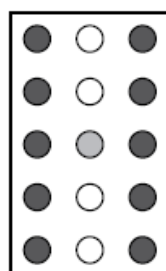
	S1	S2	S3	S4	S5	S6	S7	S8
1	•		•			•		
2	•	•	•	•	•		•	•

"G"



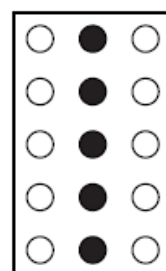
	S1	S2	S3	S4	S5	S6	S7	S8
1	•			•	•	•		
2	•	•	•	•	•		•	•

"H"



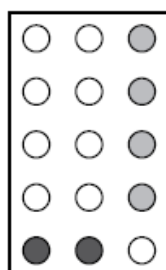
	S1	S2	S3	S4	S5	S6	S7	S8
1	•	•	•	•	•		•	
2			•			•		•

"I"



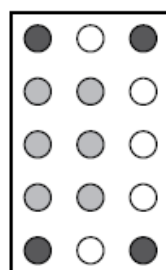
	S1	S2	S3	S4	S5	S6	S7	S8
1	•	•	•	•	•	•		•
2								

"J"



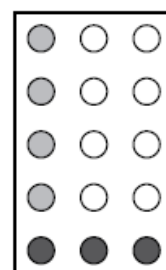
	S1	S2	S3	S4	S5	S6	S7	S8
1	•	•	•	•		•	•	
2					•			•

"K"



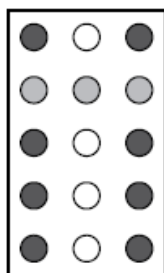
	S1	S2	S3	S4	S5	S6	S7	S8
1		•	•	•				•
2	•				•		•	

"L"



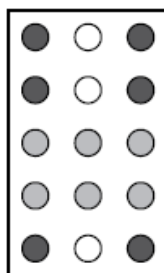
	S1	S2	S3	S4	S5	S6	S7	S8
1	•	•	•	•			•	•
2					•			

"M"



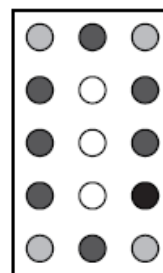
	S1	S2	S3	S4	S5	S6	S7	S8
1	●		●	●	●		●	
2		●						

"N"



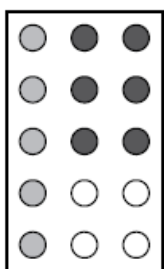
	S1	S2	S3	S4	S5	S6	S7	S8
1			●	●				
2	●	●			●		●	

"O"



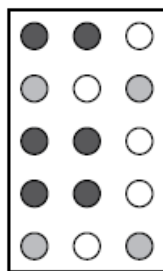
	S1	S2	S3	S4	S5	S6	S7	S8
1		●	●	●			●	
2	●				●			

"P"



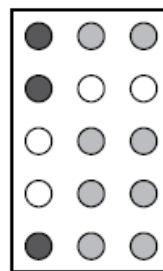
	S1	S2	S3	S4	S5	S6	S7	S8
1	●	●	●			●		
2	●	●	●	●	●		●	●

"R"



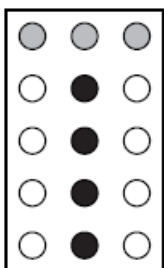
	S1	S2	S3	S4	S5	S6	S7	S8
1	●		●	●				●
2		●			●		●	

"S"



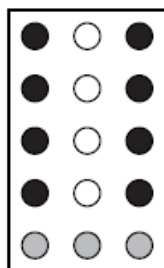
	S1	S2	S3	S4	S5	S6	S7	S8
1	●		●	●	●	●		
2	●	●			●		●	●

"T"



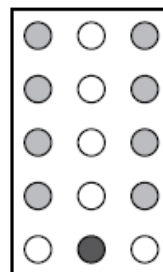
	S1	S2	S3	S4	S5	S6	S7	S8
1	●							
2		●	●	●	●	●		

"U"



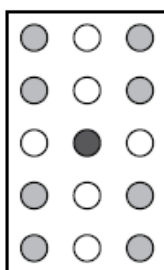
	S1	S2	S3	S4	S5	S6	S7	S8
1	●	●	●	●			●	
2					●			

"V"



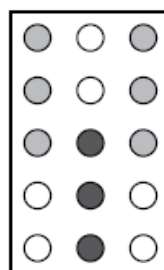
	S1	S2	S3	S4	S5	S6	S7	S8
1	●	●	●	●			●	
2					●	●		●

"X"



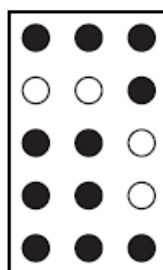
	S1	S2	S3	S4	S5	S6	S7	S8
1	●	●		●	●		●	
2			●			●		●

"Y"



	S1	S2	S3	S4	S5	S6	S7	S8
1	●	●	●				●	
2			●	●	●	●		●

"Z"



	S1	S2	S3	S4	S5	S6	S7	S8
1	●		●	●	●			●
2	●	●			●	●	●	

Obr. 8.2: Písmena

Úkol

1. Naprogramujte slovo AHOJ. Kolik adres v paměti budeme potřebovat celkem a proč? Kolik výstupů čítače musíme připojit na adresní vstupy paměti? Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**
2. Pod jednotlivé bloky paměti (spínače SA9 a SA10) naprogramujte krátký rozhovor. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**
3. Zkuste vymyslet vlastní způsob naprogramování a rozsvěcení písmen. Ověřte zapojením. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

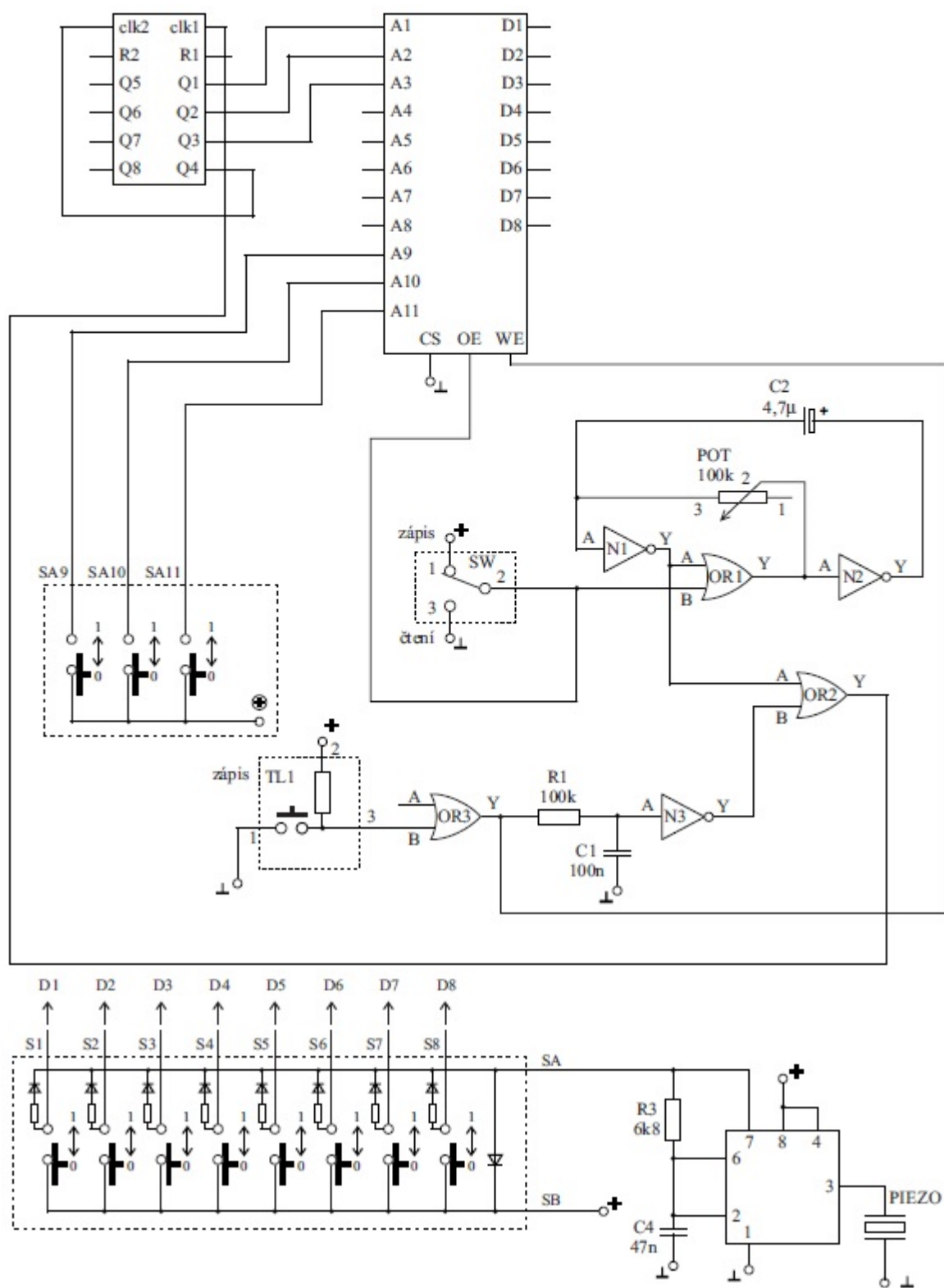
II. Hudba

9. Hudba

Po střídání rozsvěcování jednotlivých diod budeme chtít opět pokročit o kus dále. Světelnou signalizaci v této kapitole vyměníme za akustickou a ukážeme si, jakým způsobem je možné uložit do paměti jednotlivé tóny a ty pak nechat postupně přehrávat.

Stejně jako v předchozí úloze – 7 jsme rozsvěcování jednotlivých diod nastavovali pomocí spínačů S1–S8, tak i zde jednotlivé tóny budeme nastavovat pomocí těchto spínačů.¹⁰ Nejprve uložíme jednotlivé tóny do paměti, to provedeme stejně jako v předchozích úlohách. Na Obr. 9.1 je znázorněno schéma zapojení. Spínač SW přepneme do polohy ZÁPIS (nahoru). Kombinací spínačů S1–S8 nastavíme požadovaný tón (daný zvuk je slyšet) a zmáčkne TL1, tím se daný tón uloží do paměti. Na čítači se přičte 1 a můžeme ukládat další tón do další paměťové adresy. Jak nastavit jednotlivé tóny pomocí spínačů je znázorněno v Tab. 9.1. Pomocí spínačů SA9–SA10 si opět můžeme naprogramovat až osm odlišných melodií, které budou uloženy v jednotlivých paměťových blocích. Rychlost přehrávání (rychlost generování pulzů jdoucích do čítače) je možné měnit pomocí potenciometru POT. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** Následně přepneme SW do polohy „ČTENÍ“.

¹⁰ Detailně vysvětleno ve [3] úloha 10.



Obr. 9.1: Hudba

Tón	Číslo tónu	S1	S2	S3	S4	S5	S6	S7	S8
žádný	0								
F	1					X		X	X
Fis	2					X	X		
G	3					X	X		X
Gis	4				X				
A	5				X				X
Ais	6				X			X	
H	7				X			X	X
C1	8				X		X		
Cis1	9				X		X	X	
D1	10				X		X	X	X
Dis1	11				X	X			
E1	12				X	X		X	
F1	13				X	X	X		
Fis1	14				X	X	X		
G1	15			X					X
Gis1	16			X			X		
A1	17			X			X	X	X
Ais1	18			X		X		X	
H1	19			X		X	X	X	
C2	20			X	X			X	X
Cis2	21			X	X	X			
D2	22			X	X	X	X		X
Dis2	23		X				X		X
E2	24		X			X	X		
F2	25		X		X		X	X	
Fis2	26		X	X					
G2	27		X	X		X	X		
Gis2	28	X							X
A2	29	X			X		X		X
Ais2	30	X		X		X	X	X	
H2	31	X	X		X			X	
C3	32	X	X	X	X	X	X	X	X

Tab. 9.1: Tóny

Frekvence jednotlivých tónů neodpovídají jejich názvům z hlediska ladění tónu A1 na 440 Hz, nicméně tónové intervaly zůstávají zachovány. U tónů F–H je ladění už poněkud nepřesné.

Úkol

1. Kolik tónů umožňuje uložit do paměti schéma na Obr. 9.1? Jak je možné rozšířit počet uložených tónů? Jaký maximální počet tónů můžeme do paměti uložit a proč?
2. Vyzkoušejte naprogramovat melodie následujících písní. K programování využijte schéma na Obr. 9.1 a Tab. 9.1. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

Už ty pilky dořezali: 13, 17, 15, 18, 17, 13, 15 a 8.

Kočka leze dírou: 8, 10, 12, 13, 15, 0, 15 a 15.

3. Upravte schéma na Obr. 9.1 tak, aby bylo možné ukládat 16 tónů a naprogramujte následující melodie. K programování využijte Tab. 9.1. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

Pár havraních copánků: 10, 15, 14, 15, 17, 12, 17, 17, 15, 14, 12, 14, 15, 15, 15 a 0.

Hrály dudy: 13, 17, 8, 17, 13, 17, 8, 17, 20, 20, 17, 13, 12, 15, 8 a 8.

A já sám: 15, 15, 20, 20, 15, 15, 12, 12, 15, 13, 13, 10, 17, 15, 15 a 12.

4. Upravte schéma na Obr. 9.1 tak, aby bylo možné ukládat 32 tónů a naprogramujte následující melodie. K programování využijte Tab. 9.1. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

Černý muž: 13, 13, 11, 11, 10, 13, 18, 20, 22, 22, 22, 20, 18, 18, 0, 0, 15, 15, 17, 17, 18, 18, 17, 15, 13, 15, 13, 11, 10, 10, 0 a 0.

Červená řeka: 8, 13, 17, 0, 17, 15, 13, 13, 15, 13, 10, 13, 13, 13, 13, 13, 8, 13, 17, 0, 17, 17, 20, 20, 18, 17, 15, 15, 15, 15 a 15.

Okolo Hradce: 6, 13, 13, 12, 15, 15, 13, 13, 17, 20, 20, 20, 18, 18, 17, 17, 15, 15, 15, 13, 12, 12, 15, 15, 8, 8, 8, 8, 0, 0, 0 a 0.



Obr. 10.3: Béčko

Pokud už víme jaké noty naprogramovat, potřebujeme určit jejich časové intervaly. Pokud bude v notovém zápisu nejkratší nota osminová, pak pro nás bude nejmenším programovatelným časovým intervalem (možno nastavovat pomocí potenciometru POT). Všechny další noty (celá, půlová, čtvrtová) nastavíme jako daný počet not osminových apod.

Úkol

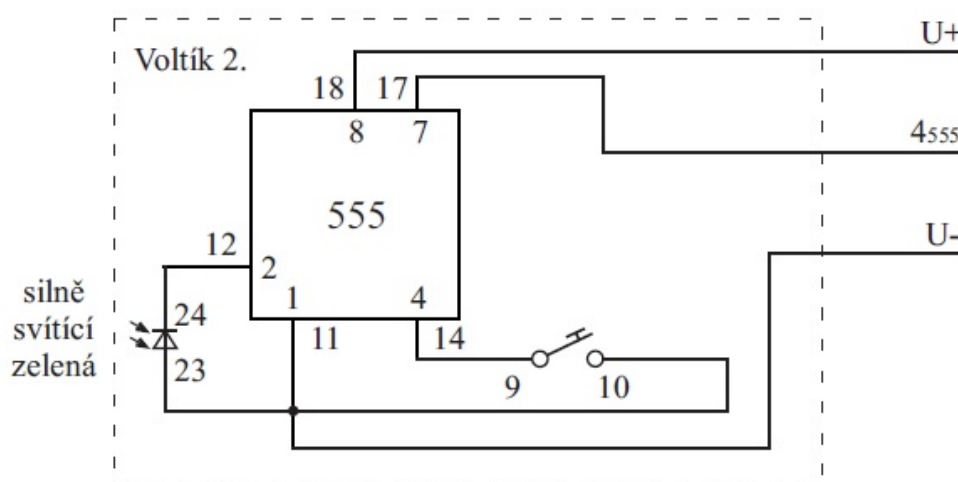
1. Vyberte si libovolný tón a naprogramujte, aby hrála nejprve nota šestnáctinová, pak osminová, čtvrtová, půlová a nakonec celá. K programování využijte schéma na Obr. 9.1 a Tab. 9.1. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**
2. Naprogramujte vámi vybranou melodii dle notového zápisu. K programování využijte schéma na Obr. 9.1 a Tab. 9.1. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

III. Paměťové členy v rozšířeném zapojení

V následujících úlohách budeme k realizaci potřebovat větší množství součástek. Z tohoto důvodu je výhodné kombinovat schémata zapojení ze stavebnic Voltík II a Voltík III. Velice se tím rozšíří možnosti využití paměťových členů.

11. Budík

Nyní budeme chtít vytvořit budík, který bude reagovat na světlo. V momentě, kdy se rozední, začne budík hrát zvolenou melodii. Pokud k zapojení předchozích schémat byla využita stavebnice Voltík III, následující obvod je nutné rozšířit o další součástky, případně o stavebnici Voltík II. Obvod sestavený dle schématu na Obr. 11.1 připojíme ke schématu na Obr. 9.1.



Obr. 11.1: Světelný detektor

Obvod na Obr. 11.1 připojíme pomocí vývodů vpravo ($U+$, 4_{555} a $U-$) k obvodu na Obr. 9.1. Napájení je připojeno jen přes obvodovou část ve schématu na Obr. 9.1. Pokud fotodiodu zakryjeme, nebude se dít nic, při osvětlení fotodiody by měla zaznít naprogramovaná melodie. Melodii naprogramujeme stejným postupem jako v kapitole II. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

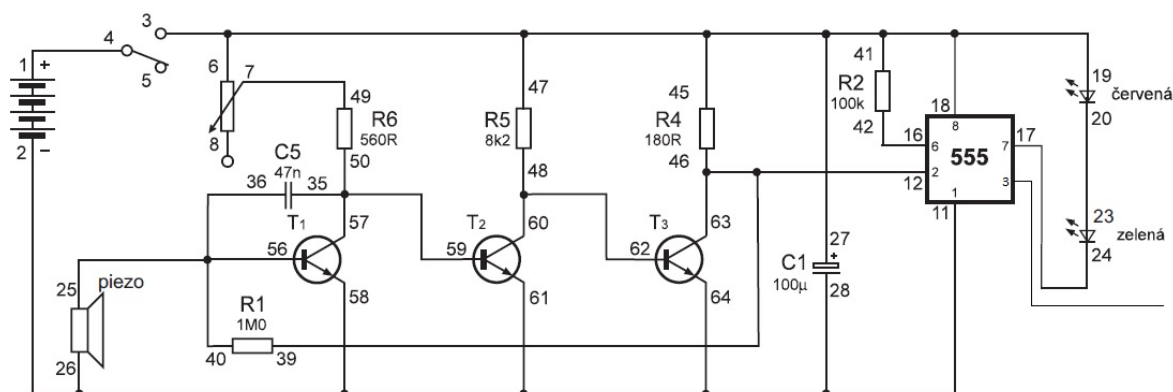
Úkol

1. Sestavte budík citlivý na světlo dle popisu výše. Naprogramujte libovolnou melodii a ověřte funkci. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**
2. Upravte schéma z předchozího úkolu tak, aby budík hrál, pokud je tma a nehrál, pokud je světlo. Jakými způsoby je tohoto možné dosáhnout? Vyzkoušejte. Nezapomeňte, že nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

12. Hudební světelné efekty

V poslední úloze sestavíme obvod, který po naprogramování bude rozsvěcovat odpovídající kombinaci diod. Spouštění obvodu bude možné provést ručně, pomocí TL2, nebo automaticky v momentě když začne hrát hudba. Pro automatické spuštění bude nutné přidat některé další součástky, nebo propojit stavebnici Voltík III se stavebnicí Voltík II. Obvod zapojíme dle schématu na Obr. 12.1.

propojky 1 a 2 ze stavebnice Voltík II připojíme k napájení a výstup 3 obvodu 555 (propojka 13) stavebnice Voltík II připojíme k TL2, jak je zobrazeno na schématu.



Obr. 12.2: Akustická část hudebních světelných efektů

Programování probíhá následovně – spínač SW přepneme do polohy ZÁPIS (nahoru). Před programováním je nutné nejprve vynulovat čítač krátkým stiskem TL2. Následně můžeme naprogramovat zvolenou sekvenci diod, kombinaci vždy uložíme pomocí stisku TL1. Nutné je též naprogramovat jeden znak navíc, který bude svítit, když nehraje hudba. V našem zapojení to znamená sedmnáctý znak.¹¹ Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!** Následně přepneme SW do polohy „ČTENÍ“. Krátce stiskneme TL2 a měla by proběhnout naprogramovaná sekvence a dále bude svítit klidový znak. Rychlost blikání diod je možno regulovat pomocí POT, nebo počtem svítících cyklů diod v programu.

Při přeprogramování je nutné odpojit výstup 3 obvodu 555 Voltíka II od zbytku obvodu!

Pokud budeme chtít sekvenci prodloužit nebo zkrátit, je to možné učinit tak, že vstup B členu OR4 připojíme k jinému vstupu čítače (Q2–Q8).

Úkol

1. Naprogramujte sekvenci dle

	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.	X	X	X	X	X			
1.	X	X	X	X	X			
2.	X	X	X	X	X			

¹¹ Vstup B členu OR4 je připojen ke Q5 čítače. Máme tedy $2^4 = 16$ sekvencí a jednu poslední sekvenci klidového znaku odpovídající stavu čítače 1 v Q5 – zablokování čítače.

3.		X	X	X	X			
4.		X	X	X	X			
5.		X	X	X	X			
6.			X	X	X			
7.			X	X	X			
8.			X	X	X			
9.				X	X			
10.				X	X			
11.				X	X			
12.					X			
13.					X			
14.					X			
15.					X			
16.					X			

2. Tab. 12.1. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.	X	X	X	X	X			
1.	X	X	X	X	X			
2.	X	X	X	X	X			
3.		X	X	X	X			
4.		X	X	X	X			
5.		X	X	X	X			
6.			X	X	X			
7.			X	X	X			
8.			X	X	X			
9.				X	X			
10.				X	X			
11.				X	X			
12.					X			
13.					X			
14.					X			
15.					X			
16.					X			

Tab. 12.1: Hudební světelné efekty 1

3. Naprogramujte sekvenci dle Tab. 12.2. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.	X					X	X	
1.					X		X	X
2.	X					X	X	
3.					X		X	X
4.		X				X	X	
5.				X			X	X

6.			X			X	X	
7.			X				X	X
8.				X		X	X	
9.		X					X	X
10.					X	X	X	
11.	X						X	X
12.					X	X	X	
13.	X						X	X
14.					X	X		X
15.	X					X		X
16.	X				X	X		X

Tab. 12.2: Hudební světelné efekty 2

4. Naprogramujte sekvenci dle Tab. 12.3. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.	X	X	X	X	X			
1.	X	X	X	X	X			
2.		X	X	X				
3.		X	X	X				
4.			X					
5.			X					
6.			X					
7.			X					
8.			X					
9.			X					
10.			X					
11.			X					
12.			X					
13.			X					
14.			X					
15.			X					
16.			X					

Tab. 12.3: Hudební světelné efekty 3

5. Naprogramujte sekvenci dle Tab. 12.4. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.	X	X	X	X	X			
1.	X	X	X	X	X			
2.		X	X	X				
3.		X	X	X				
4.			X					
5.			X					

6.			X					
7.			X					
8.			X					
9.			X					
10.			X					
11.			X					
12.			X					
13.			X					
14.			X					
15.			X					
16.			X					

Tab. 12.4: Hudební světelné efekty 4

6. Vymyslete vlastní sekvenci hudebních světelných efektů. Tuto sekvenci doplňte do Tab. 12.5 a následně naprogramujte. Nežli přepneme spínač SW do polohy „ČTENÍ“, **JE NUTNÉ SPÍNAČE S1–S8 PŘEPNOUT DO POLOHY 0!**

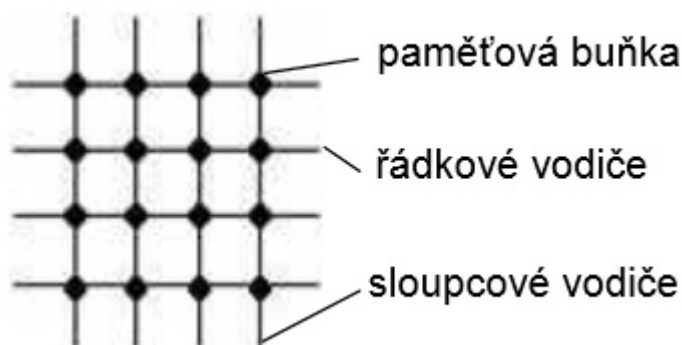
	Spínač, který bude sepnut							
Znak	S1	S2	S3	S4	S5	S6	S7	S8
0.								
1.								
2.								
3.								
4.								
5.								
6.								
7.								
8.								
9.								
10.								
11.								
12.								
13.								
14.								
15.								
16.								

Tab. 12.5: Hudební světelné efekty 5

Přílohy

Paměť

Polovodičové paměti se skládají z paměťových buněk. Paměťová buňka se skládá z obvodu, který umožňuje dočasně, nebo trvale uchovat logické hodnoty LOG-0, nebo LOG-1. Každá paměťová buňka tedy může uchovat pouze 1 bit. Paměťové buňky jsou na čipu uspořádány do mřížky, viz Obr. P. 1.



Obr. P. 1: Paměťové buňky

Adresa každé buňky je určena řádkovým a sloupcovým vodičem. O nalezení buňky na zadané pozici se stará paměťový řadič, který má také za úkol řídit zápis do buňky a čtení z buňky.

Polovodičové paměti můžeme z hlediska zápisu a čtení rozdělit do dvou skupin:

- RWM (Read/Write Memory) – paměť umožňující jak zápis, tak i čtení. Uchová informaci, jen pokud je zajištěno napájení. Paměti tohoto druhu s přímým přístupem (uspořádání do mřížky) se označují jako **RAM** (Random Access Memory)
- ROM (Read Only Memory) – Nelze do nich standardním způsobem zapisovat. Uchovávají informaci i po odpojení napájení.

Paměti RAM můžeme rozdělit podle typu obvodu, který je použit pro paměťovou buňku:

- SRAM (Static Random Access Memory) – paměťová buňka je tvořena bistabilním klopným obvodem.
- DRAM (Dynamic Random Access Memory) – paměťová buňka je tvořena kondenzátorem a tranzistorem MOSFET.

Doporučená literatura

- [1] VALÁŠEK, Pavel, LOSKOT Roman. *Polovodičové paměti*. 2. vyd. Praha: BEN – technická literatura, 2000, 235 s. ISBN 80-860-5679-1.
- [2] BARTONĚK, Luděk. Informatika I: Struktura počítačů [online]. Olomouc, 2012 [cit. 2012-07-22]. ISBN 978-80-244-2973-1.
Dostupné z: <http://fyzika.upol.cz/cs/predmety-kef-slo/informatika-1>
- [3] ŠAFAŘÍK, Jakub. *Netradiční sbírka z fyziky II – Logické obvody a jejich zapojení 2*. 1. vyd. Praha: Gymnázium Botičská, 2013, 45 s.
Dostupné z: <http://gybot.cz>
- [4] ŠAFAŘÍK, Jakub. *Netradiční sbírka z fyziky I – Logické obvody a jejich zapojení 1*. 1. vyd. Praha: Gymnázium Botičská, 2012, 51 s.
Dostupné z: <http://gybot.cz>
- [5] SVOBODA, Ivan, SVOBODOVÁ Radka. Průvodce stavebnicí VOLTÍK III. Ostrava – Petřkovice: SVOBODA, 2000.
Dostupné z: http://voltik.cz/fotky/voltik/v3/prirucka_v3.pdf