

Vážení zákazníci,

dovolujeme si Vás upozornit, že na tuto ukázkou knihy se vztahují autorská práva, tzv. copyright.

To znamená, že ukáзка má sloužit výhradně pro osobní potřebu potenciálního kupujícího (aby čtenář viděl, jakým způsobem je titul zpracován a mohl se také podle tohoto, jako jednoho z parametrů, rozhodnout, zda titul koupí či ne).

Z toho vyplývá, že není dovoleno tuto ukázkou jakýmkoliv způsobem dále šířit, veřejně či neveřejně např. umístováním na datová média, na jiné internetové stránky (ani prostřednictvím odkazů) apod.

redakce nakladatelství BEN – technická literatura
redakce@ben.cz



3.4 Návrh systému

Celý postup návrhu systému si ukážeme na príklade.

Popis chovania systému

Najprv podrobne definujeme systém, ktorý hodláme realizovať. Popíšeme presne čo majú vstupy a výstupy vykonávať.



Navrhujeme systém, ktorý má na vstupe štyri tlačidlá a na výstupe štyri nízkoprikonové LED.

Pri stlačení tlačidla bude na vstupe úroveň L. LED sa rozsvieti, ak bude na výstupe úroveň L.

Systém je riadený mikrokontrolérom PIC16F84(A) s RC oscilátorom s kmitočtom generátora hodín 4 MHz a bude napájaný 5 V.

Ak stlačíme tlačidlo, rozsvieti sa príslušná LED. LED teda budú indikovať stlačenie tlačidiel (*LED0 stlačenie tlačidla T10, LED1 stlačenie tlačidla T11 atď.*).

Návrh zapojenia systému

Podľa zadania navrhujeme konkrétne zapojenie systému vrátane hodnôt súčiastok.



Pretože mikrokontrolér vie na vstupoch čítať len logické úrovne, je potrebné obvody navrhnuť tak, aby mu udalosť, ktorú má čítať, bola podaná vo forme logických úrovní (*napríklad stlačené tlačidlo je indikované logickou nulou, ak nie je tlačidlo stlačené, bude na vstupe logická jednotka*). To isté platí aj pre výstupy mikrokontroléra (*napríklad LED svieti, ak je na výstupe logická nula, a nesvieti, ak je výstup v logickej jednotke*).

Nepoužitú I/O piny buď zapojíme cez odpor k V_{SS} , alebo k V_{DD} , a nastavíme ich ako vstupy, alebo ich necháme nezapojené a musíme ich programom nastaviť ako výstupy.

Je to preto, aby cez tieto piny nemohli prenikat' do mikrokontroléra rušivé signály zvonku, a tým spôsobiť jeho nesprávnu funkciu.

Navrhujeme schému zapojenia, vrátane použitých súčiastok, spočítame hodnoty odporov a kde je potrebné i kondenzátorov.

Podľa vypočítaných hodnôt zvolíme hodnoty z rady vyrábaných hodnôt a tieto potom zapíšeme do navrhutej schémy.

Nepoužité I/O piny ponecháme nezapojené a programom ich nastavíme ako výstupy.

Realizujeme zapojenie na kontaktnom poli.



2

Ako sa hovorí: „Dvakrát meraj a raz rež“. Nie je od veci si ešte overiť funkciu navrhnutého systému bez prítomnosti mikrokontroléra. To platí dvojnásobne pre výstupy, napájanie mikrokontroléra a pripojenie vstupu –MCLR.

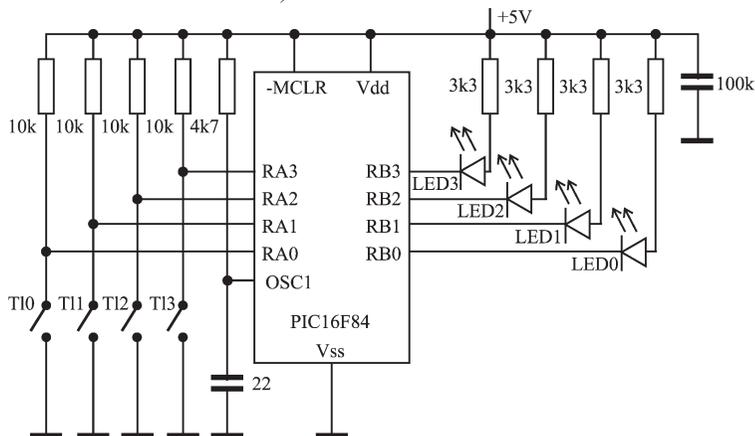
Funkciu výstupov overíme nasledovne:

Bez osadeného mikrokontroléra skontrolujeme, či nebudú naozaj k ničomu omylom pripojené nepoužité a nepripojené vývody nastavené ako výstupy mikrokontroléra, či privádzame V_{dd} a V_{ss} na správne piny a či je skutočne vstup –MCLR pripojený na V_{dd} .

Potom pripojíme napájacie napätie.

Odpor $100\ \Omega$ pripojíme k V_{dd} . Jeho druhým koncom sa dotýkame výstupov (*príslušných vývodov v päťici pre mikrokontrolér*). Tak overujeme chovanie sa výstupov v úrovni H. Potom pripojíme odpor $27\ \Omega$ k V_{ss} a opäť sa dotýkame výstupov. Overíme ich činnosť v úrovni L. Namiesto odporu môžeme použiť drôt.

Tým overíme funkciu výstupov, akoby ich riadil mikrokontrolér (*v našom prípade pripojením k V_{dd} príslušná LED nesvieti, pri pripojení k V_{ss} sa musí rozsvietiť*).



Obr.3.8 Konkrétne zapojenie systému

Podrobná analýza chovania systému

Teraz nastáva podrobná analýza chovania systému. Je potrebné navrhnuť činnosť programu tak, aby systém vykonával všetko, čo od neho požadujeme. Musíme dôkladne premyslieť činnosť programu. Pretože mikrokontrolér vykonáva jednu inštrukciu po druhej, musíme všetko popísať tak, aby sme ošetrili všetky stavy vstupov, ktoré môžu nastať, vrátane tých, ktoré môžu vzniknúť z dôvodov rušivých vplyvov, alebo z dôvodu vlastností súčiastok (*napríklad zakmitávanie kontaktov tlačidiel a podobne*).



Mikrokontrolér po zapnutí napájania má automaticky nastavený čítač programu PC-program counter („prst“, ktorý ukazuje na inštrukciu, ktorá sa má vykonať) na adresu 000 pamäte programu.



5

Prečíta inštrukciu na adrese 000 a inkrementuje čítač programu (*zvýši ho o jedna – posunie prst na nasledujúcu inštrukciu, teda na adresu 001*). Vykoná prečítanú inštrukciu z adresy 000. Po jej vykonaní prečíta inštrukciu z adresy, kam ukazuje PC (*v našom prípade z adresy 001*), inkrementuje PC (*PC teraz ukazuje na adresu 002*) a vykoná inštrukciu z adresy 001. Tak pokračuje stále.

UPOZORNENIE:

PC vždy pri vykonávaní inštrukcie ukazuje na nasledujúcu inštrukciu (na adresu inštrukcie nasledujúcu za inštrukciou práve vykonávanou).



Ako si najlepšie a najnázornejšie popísať činnosť nami navrhovaného systému?

Najlepšie je použiť **vývojový diagram**. Ide o spojenie grafického vyjadrenia s popisom činnosti. Je tu dobre vidieť, čo bude mikrokontrolér vykonávať krok za krokom.

Najskôr si vytvoríme vývojový diagram chovania celého systému, aby bolo na prvý pohľad jasné, čo a kedy bude systém vykonávať.

Program po zapnutí napájania mikrokontroléra musí najskôr nastaviť jeho vstupy a výstupy. Musí mikrokontrolér **nakonfigurovať**?

Pretože nevieme, ako budú nastavené registre výstupov a požadujeme, aby LED po zapnutí mikrokontroléra nesvietili, musíme teda výstupy RB3/0 nastaviť do H.

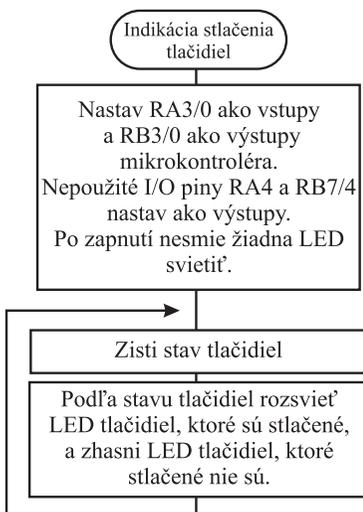
Nepoužité I/O piny nastavíme ako výstupy.

Potom mikrokontrolér zistí, v akom stave sú tlačidlá. Podľa toho rozsvieti LED tlačidiel, ktoré sú stlačené a zhasne LED tlačidiel, ktoré stlačené nie sú.



Poznámka: *I keď môžu kontakty tlačidiel odskakovať a mikrokontrolér vykonáva pri kmitočte generátora hodín 4 MHz jeden milión inštrukcií za sekundu (kmitočet generátora hodín delený 4 je doba vykonávania väčšiny inštrukcií), budú LED indikovať i tieto odskoky. Vzhľadom k zotrvačnosti oka a k tomu, že odskoky môžu trvať rádovo 10 ms (za túto dobu mikrokontrolér vykoná desiatky tisíc inštrukcií), bude zablikanie indikačných LED okom nepostrehnuteľné, a nie je preto potrebné sa nimi v našom prípade zaoberať.*

Teraz si už vytvoríme vývojový diagram chovania systému presne podľa nášho popisu. Zatiaľ sa nebudeme zamýšľať nad tým, ako tú, či onú činnosť bude vykonávať mikrokontrolér. Ide o popis chovania systému ako takého.



Obr. 3.9 Vývojový diagram chovania celého systému

Teraz podľa potreby podrobnejšie rozkreslíme jednotlivé bloky vývojového diagramu chovania celého systému tak, aby sme boli schopní pomocou inštrukcií mikrokontroléra potrebné činnosti realizovať.

Každý mikrokontrolér vie vykonávať určité inštrukcie. Všetky inštrukcie, ktoré vie mikrokontrolér vykonávať, sa nazývajú **súbor inštrukcií mikrokontroléra** alebo tiež len **inštrukčný súbor**. Pomocou týchto inštrukcií musíme realizovať činnosť, ktorú chceme, aby systém vykonával.

Poznámka: *Je to, ako by bol človek, ktorý vie dokonale, presne a rýchlo vykonávať len nejaké činnosti a rozumie, keď mu povieme, aby tú či onú činnosť vykonal. Ak potrebujeme, aby vykonal nejakú zložitú prácu, musíme mu dávať príkazy postupne tak, aby vykonal túto zložitú prácu iba pomocou činností, ktoré vie vykonať.*



Mikrokontrolér PIC16F84 má inštrukčný súbor zložený z 35 inštrukcií. Vie vykonávať základné aritmetické a logické operácie, nulovať a nastavovať jednotlivé bity registrov, nulovať obsah registrov. Dokáže presúvať (presnejšie kopírovať) obsah registra, nastaviť do registra číslo, vykonávať rotáciu bitov registra (rotáciu obsahu registra o bit vľavo/vpravo), zameniť horné štyri bity registra s dolnými štyrmi bitmi a vykonávať skoky v programe. Vie pracovať s podprogramami a prerušením programu. Zo začiatku nebudeme potrebovať používať všetky jeho inštrukcie. Dokonca aj keď už budeme písať zložitejšie programy, vystačíme spravidla s menším počtom inštrukcií, než nám ponúka celý inštrukčný súbor.

Buďme bez obáv, všetky inštrukcie si budeme postupne ukazovať a vysvetľovať na príkladoch. Tak si ani neuvedomíme, ako sa ich naučíme.

Nemusíme sa ani inštrukcie učiť spamäti. Máme po ruke tabuľku inštrukcií (kompletná tabuľka inštrukcií i s ich popisom je súčasťou publikácie [1]) a ich popis. Tabuľka inštrukcií PIC16F84 použitých v tomto dieli učebnice je jeho súčasťou.

Poznámka: *Inštrukčný súbor je zhodný s mikrokontrolérom PIC16C71 a väčšinu inštrukcií ovládajú i mikrokontroléry PIC16C5X a ďalšie. Môžeme teda s výhodou, po drobných úpravách, používať programy napísané pre PIC16F84 i pri práci s inými mikrokontrolérmi.*



Vráťme sa teraz k nášmu príkladu. Pokúsme sa vývojový diagram chovania systému rozkresliť podrobnejšie, aby sa dal podľa neho napísať program.

Po zapnutí napájania mikrokontroléra je potrebné nastaviť nepoužité vývody RA4 a RB7/4 ako výstupy. LED máme pripojené k RB3/0, preto i tie nastavíme ako výstupy. Tlačidlá sú pripojené k RA3/0, nastavíme ich ako vstupy.