

Model zabezpečovacího systému pro výuku předmětu Mikropočítače a PLC

Security System Model for Microcomputers and PLC Lessons

Zdeněk Koryťák



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Zdeněk KORYŤÁK**
Osobní číslo: **A08126**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Bezpečnostní technologie, systémy a management**

Téma práce: **Model zabezpečovacího systému pro výuku
předmětu Mikropočítače a PLC**

Zásady pro vypracování:

- 1. Prostudujte hardwarové vlastnosti vývojového kitu M68EVB908GB60 a navrhnete způsob připojení modelu zabezpečovacího systému**
- 2. Provedte hardwarový návrh modelu zabezpečovacího systému, ve kterém bude činnost jednotlivých detektorů simulována tlačítky.**
- 3. Realizujte prototyp modelu a ověřte jeho funkci na vývojovém kitu.**
- 4. Vytvořte podpůrné programové vybavení pro obsluhu modelu ve formě knihovny podprogramů v jazyce symbolických adres a v C jazyce**
- 5. Vytvořte ukázkovou aplikaci s využitím realizované knihovny podprogramů, která bude demonstrovat funkci modelu.**

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. FREESCALE SEMICONDUCTOR. CPU08 Central Processor Unit Reference Manual [online]. 2001. Dostupné z: www.freescale.com
2. FREESCALE SEMICONDUCTOR. HCS08 Family Reference Manual, Rev.1. [online]. 2003. Dostupné z: www.freescale.com
3. FREESCALE SEMICONDUCTOR. MC9S08GB/GT Data Sheet, Rev.2.3. [online]. 2004. Dostupné z: www.freescale.com
4. JURÁNEK, Antonín a Miroslav HRABOVSKÝ. EAGLE pro začátečníky /: uživatelská a referenční příručka :. 2. vydání. Praha: BEN – technická literatura, 2007, 191 s. ISBN 80-730-0213-2.
5. MANN, Burkhard. C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy – linkery, práce s ATMEL AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky. Vyd. 1. Praha: BEN, 2003, 279 s. ISBN 80-730-0077-6.
6. KINDL, Jiří. Projektování bezpečnostních systémů I. Vyd. 2. Zlín: Univerzita Tomáše Bati, 2007, 134 s. ISBN 978-807-3185-541.

Vedoucí bakalářské práce:

Ing. Petr Dostálek, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

24. února 2012

Termín odevzdání bakalářské práce:

25. května 2012

Ve Zlíně dne 24. února 2012

prof. Ing. Vladimír Vašek, CSc.
děkan



doc. Mgr. Milan Adámek, Ph.D.
ředitel ústavu

ABSTRAKT

Tato bakalářská práce pojednává o návrhu, realizaci a ověření modelu zabezpečovacího zařízení. Dále řeší připojení a komunikaci s vývojovým kitem M68EVB908GB60. V praktické části je práce zaměřena na obecný popis mikropočítačů a představení použitého komponentu mikropočítače z řady HCS08 od firmy Freescale, v poslední řadě pak postup při návrhu poplachového zabezpečovacího systému. V praktické části bakalářské práce je zdokumentován postup při výrobě, popis zapojení a vysvětlení principu programu.

Klíčová slova: Freescale HCS08, SPI, poplachový zabezpečovací systém, Výukový model

ABSTRACT

This bachelor thesis describes the design, implementation and verification of the safety equipment. It also deals with the connection and communication with the development board M68EVB908GB60. In the practical part the work is focused on the general description of the microprocessors and introducing the used component of the microcomputer of a series HCS08 produced by company Freescale. There is also described the design of the process for security alarm system on the end. In the practical part of the thesis is illustrated a process for the production, description of the connection and explanation of the principle of involvement program.

Keywords: Freescale HCS08, SPI, Security Alarm System, Instructional Model

Rád bych poděkoval svému vedoucímu bakalářské práce panu Ing. Petu Dostálkovi, Ph.D. za odborné vedení, cenné rady a čas, který mi věnoval při tvorbě této práce. Dále bych chtěl poděkovat své rodině a přátelům, kteří mě podporovali po celou dobu studia.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....

podpis diplomanta

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST	10
1 MIKROPOČÍTAČE.....	11
1.1 ZÁKLADNÍ POJMY	11
1.2 ZÁKLADNÍ ARCHITEKTURY MIKROPOČÍTAČŮ	11
1.2.1 Von Neumannova koncepce.....	11
1.2.2 Harvardská architektura	12
1.3 INSTRUKČNÍ SADY	13
1.4 VÝVOJOVÝ KIT M68EVB908GB60	14
1.4.1 Technické parametry	15
1.4.2 Mikropočítač MC9S08GB60	16
1.4.3 Hlavní znaky HCS08 jádra.....	16
1.4.4 Funkce typu MC9S08GB60	16
1.4.5 Zařízení v řadě MC9S08GB60.....	17
1.5 SÉRIOVÉ ROZHRANÍ SPI	18
2 NÁVRH POPLACHOVÉHO ZABEZPEČOVACÍHO SYSTEMU.....	21
2.1 BEZPEČNOSTNÍ POSOUZENÍ OBJEKTU	21
2.1.1 Zabezpečované hodnoty - majetek	22
2.1.2 Stavební dispozice.....	22
2.1.3 Minimální úroveň střežení IAS, HAS	23
2.1.4 Ostatní vlivy	23
2.2 ZÁKLADNÍ DĚLENÍ OCHRANY OBJEKTU	23
2.3 POPLACHOVÝ ZABEZPEČOVACÍ SYSTÉM	25
2.4 STUPNĚ ZABEZPEČENÍ.....	25
2.5 TŘÍDY PROSTŘEDÍ.....	26
II PRAKTICKÁ ČÁST	27
3 MODEL ZABEZPEČOVACÍHO SYSTÉMU	28
3.1 NÁVRH ZABEZPEČENÍ OBJEKTU	28
3.2 REALIZACE MODELU.....	29
4 REALIZACE OVLÁDACÍHO ZAŘÍZENÍ MODELU.....	30
4.1 DESKA I.....	30
4.1.1 Schéma zapojení Desky I	30
4.1.2 Výroba Desky I.....	32
4.2 DESKA II.....	33
4.2.1 Schéma zapojení Desky II.....	34
4.2.2 Výroba Desky II	35
5 PROGRAMOVÉ VYBAVENÍ.....	36

5.1	JAZYK C	36
5.1.1	Inicializace	36
5.1.2	Ovladač signalizace	37
5.1.3	Ovladač klávesnice	38
5.1.4	Ovladač tlačítek	40
5.2	POKYNY PRO PROGRAMOVÉ VYBAVENÍ	42
6	UKÁZKOVÁ APLIKACE.....	46
	ZÁVĚR	47
	ZÁVĚR V ANGLIČTINĚ.....	48
	SEZNAM POUŽITÉ LITERATURY.....	49
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	51
	SEZNAM OBRÁZKŮ	52
	SEZNAM TABULEK.....	53
	SEZNAM PŘÍLOH.....	54

ÚVOD

Cílem této bakalářské práce je vytvořit výukovou pomůcku pro studenty fakulty Aplikované Informatiky na Univerzitě Tomáše Bati ve Zlíně. Konkrétně se jedná o předmět Mikropočítače a PLC. Tato práce se zabývá první částí předmětu, a to jsou mikropočítače. Zde se pomocí programovacích jazyků, například C jazyk nebo jazyk symbolických adres, studenti učí ovládat a řídit tyto mikropočítače a jejich periferie. Ve školních laboratořích jsou studentům k dispozici vývojové kity s označením M68EVB908GB60 od firmy Freescale.

Mikropočítače jsou v současné době nejrozšířenějšími počítači v celosvětovém měřítku. Rozsah jejich nasazení je až nezvykle široký. V nejjednodušších variantách se používají ke zvýšení "intelligence" takových zařízení, jako jsou výrobky spotřební elektroniky (televize, rádia, myčky nádobí apod.), výrobky měřicí techniky (spektrometry, osciloskopy, voltmetry aj.), ale i výrobky v dalších odvětvích (řízení benzinových stojanů, automatů na cigarety apod.). Mikropočítač vybavený příslušnými periferiemi se může stát výkonným kancelářským pomocníkem, může se uplatnit ve velké části aplikací, kde se dříve používaly střediskové počítače nebo minipočítače. Je možné si s ním doplňovat své znalosti a dovednosti, případně si i hrát.

Aby se studenti mohli lépe vcítit do problematiky a potřeb programátora mikropočítačů, je výukový model zasazen do oblasti bezpečnostních technologií, konkrétně simulací bezpečnostního poplachového systému. V odvětví bezpečnostních systémů se totiž studenti pohybují od prvního ročníku studia, proto pro ně bude snazší pochopit a představit si jak má takový modul správně pracovat.

Tato bakalářská práce tedy pojednává a stává se určitou formou technickou dokumentací o vytvořeném výukovém modulu pro předmět Mikropočítače a PLC.

I. TEORETICKÁ ČÁST

1 MIKROPOČÍTAČE

1.1 Základní pojmy

Mikroprocesor – nejčastěji reprezentován jako jeden integrovaný obvod. Jeho součástí jsou aritmeticko-logická jednotka (ALU), řídící jednotka (CU) a různé dalších pracovní jednotky. Mikroprocesor nemůže pracovat samostatně.

Mikropočítač – vznikne spojením mikroprocesoru a dalších obvodů. Tyto obvody jsou nejčastěji dodávány stejným výrobcem jako mikropočítačový systém (MSC). Mezi tyto obvody například patří paměti (memory), vstupně-výstupní obvody (I/O), generátory hodinových impulzů, řadiče přerušení (interrupt), časovače (timer) a jiné. Mikropočítač představuje minimální konfiguraci, která je schopná samostatně pracovat. [12]

1.2 Základní architektury mikropočítačů

1.2.1 Von Neumannova koncepce

Vznik kolem roku 1946. Používá jeden univerzální paměťový prostor jak pro data, tak pro umístění programu. Není jednoznačně určeno, kde jsou uložena data a kde program. Používá se u univerzálních výpočetních systémů.

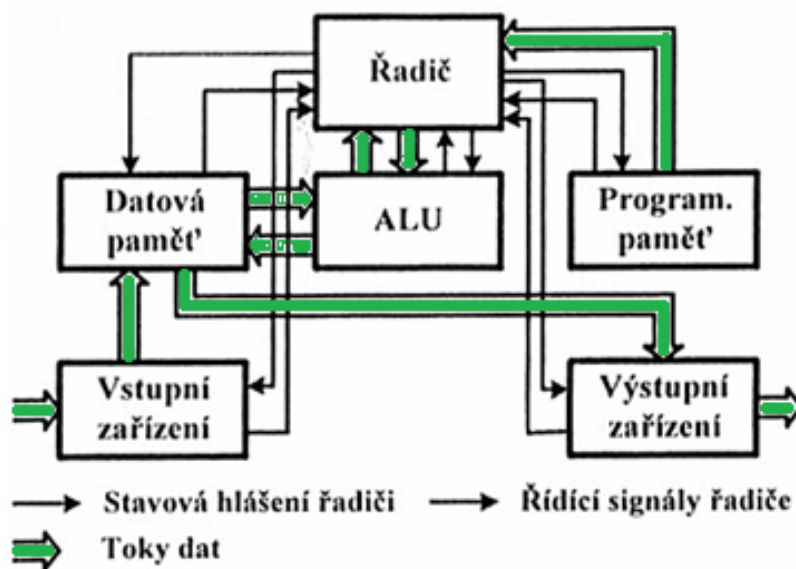


Obr. č. 1.1: Princip-Von Neumannova architekt [13]

- Do operační paměti se pomocí vstupních zařízení přes ALU umístí program, který bude provádět výpočet
- Stejným způsobem se do operační paměti umístí data, která bude program zpracovávat
- Probíhá početní operace jehož jednotlivé kroky provádí ALU. Ta je společně s ostatními moduly řízena řadičem. Mezivýsledky se ukládají do operační paměti.
- Po dokončení operace jsou výsledky zaslány přes ALU na výstupní zařízení [6], [13]

1.2.2 Harvardská architektura

Využívá dva rozdělené paměťové prostory, které představují dvě různé množiny adres pro procesor. Z jedné lze číst instrukce programu a druhou je možné používat pro ukládání zpracovaných dat. Tato paměť umožňuje zpracovávat různou délku slova pro program a pro data. To je výhodné, protože na jedno místo v paměti připadá jedna instrukce, tím dochází ke zrychlení chodu. [6], [12]



Obr. č. 1.2: Princip-Harvardská architektura [12]

1.3 Instrukční sady

Sada instrukcí procesoru pro práci s daty a pamětí. Instrukce je zapsána jako posloupnost bitů (tedy číslo). Instrukce může mít více operandů, ale také nemusí mít žádný. Operandem může být hodnota, registr či adresa v paměti. Dnešní procesory používají sadu RISC (rozšířenou o multimediální instrukce), výjimečně pak CISC,

RISC (Reduced Instruction Set Computer) – počítač s redukováným souborem instrukcí. Ze statistik vychází, že četnost opakování některých instrukcí je na tolik nízká, aby byly instrukce zařazeny do instrukčního souboru, a tím zbytečně nerozšiřovaly radič procesoru.

Vlastnosti:

- redukováný soubor instrukcí obsahuje především jednoduché instrukce;
- celkový čas jedné instrukce je jeden cyklus;
- všechny instrukce jsou stejně dlouhé.

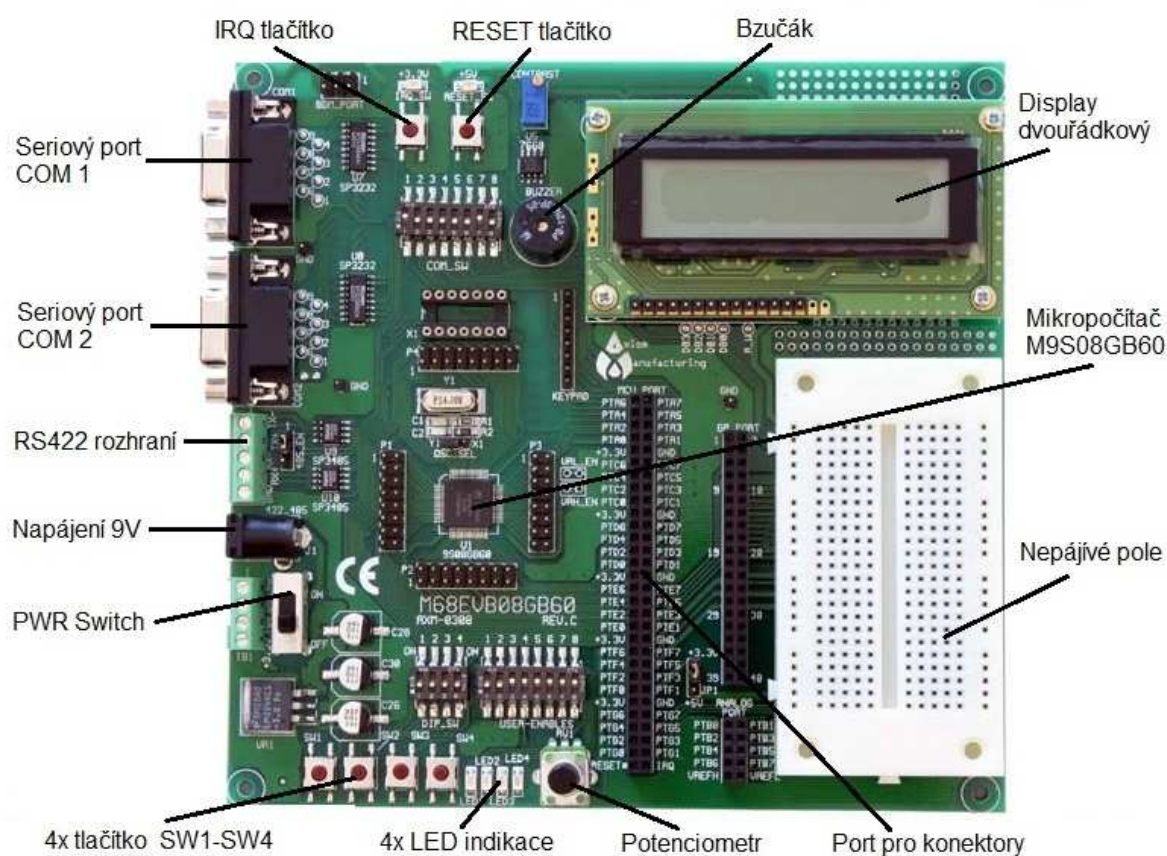
CISC (Complex Instruction Set Computer)- počítač se složitým souborem instrukcí. Mikroprocesor obsahuje velmi obsáhlý instrukční soubor, tudíž zadáním jednoho příkazu je možno realizovat složité operace.

Vlastnosti:

- obsáhlý soubor instrukcí;
- instrukce mají proměnlivou délku i dobu trvání;
- obsahuje nízký počet registrů. [7], [12]

1.4 Vývojový kit M68EVB908GB60

V laboratořích FAI jsou pro výuku studentů k dispozici vývojové kity M68EVB908GB60 z rodiny mikrokontrolerů GB60 od výrobce Freescale. Kit obsahuje řadu periférií a hardwarových prvků, které jsou popsány na obrázku níže viz Obr. č.1.3: Vývojový kit M68EVB908GB60. Aby mohli studenti s tímto zařízením pracovat a komunikovat, potřebují určité vývojové prostředí, k tomu slouží software CodeWarrior. Ten umožňuje vytvořit program v jazyce C, nebo v jazyce symbolických adres.



Obr. č.1.3: Vývojový kit M68EVB908GB60 [14]

1.4.1 Technické parametry

Vlastnosti a funkce:

- mikropočítač M9S08GB60
- krystalový oscilátor 32 kHz nebo 4 MHz
- nastavitelný oscilátor
- stabilizovaný zdroj 3,3 V a 5 V
- sériový port COM1, rozhraním RS232, konektor DB9-S
 - rozhraní na portu SCI0
- sériový port COM2 CAN, rozhraní RS232, konektor DB9-S nebo RS422/485
 - rozhraní na portu SCI1
- vypínač ON/OFF se signalizací
- uživatelské periferie
 - 4 LED indikátory (port PTF0-PTF3)
 - 4 poziční DIP přepínač (port PTB4-PTB7)
 - 4 tlačítka (port PTA4-PTA7)
 - dvouřádkový 16 znakový LCD displej (porty PTG3-PTG7 a PTE6-PTE7)
 - piezoelektrický akustický měnič (port PTD0)
- konektor MCU, umožňuje přístup ke všem digitálním I/O portům
- konektor analogového portu
- nepájivé pole [2]

Technické údaje:

- velikost desky: 6“ x 6.2“
- vstupní napětí: 6V až 20V stejnosměrného napětí, běžně 9V
- spotřeba proudu: 50mA při 9V stejnosměrného napětí [2]

1.4.2 Mikropočítač MC9S08GB60

Součástí vývojového kitu je mikropočítač Freescale MC9S08GB60. Ten patří do série 8-bitových mikropočítačů s velmi výkonným jádrem HCS08. V následujících kapitolách bude série HCS08 blíže specifikována.

1.4.3 Hlavní znaky HCS08 jádra

- 8 bitový CPU HCS08, možný pracovat na frekvenci až 40 MHz
- HC08 instrukční sada s přidanou BGND instrukcí
- Ladící modul obsahující dva komparátory a devět spouštěcích režimů
- Podpora až 32 přerušení
- Režimy úspory energie
- Funkční systémová ochrana:
 - možnost WatchDog (COP)
 - detekce nízkého napětí pomocí (reset/přerušení)
 - detekce nepovoleného operačního kodu (reset)
 - detekce nepovolených adres (reset) [1]

1.4.4 Funkce typu MC9S08GB60

- 60 KB FLASH paměti
- 4 KB RAM paměti
- 56 vstupně/výstupních linek na sedmi portech (porty A-G)
- kanálový TPM1 časovač
- 5 kanálový TPM2 časovač
- 1 x synchronní sériové periferní rozhraní (SPI)
- 1 x I2C rozhraní
- 2 x asynchronní sériové komunikační rozhraní (SCI)
- interní generátor hodinového kmitočtu s PLL obvodem (32KHz – 20MHz frekvence sběrnice)

- 8 kanálový, 10 bitový A/D převodník [2]

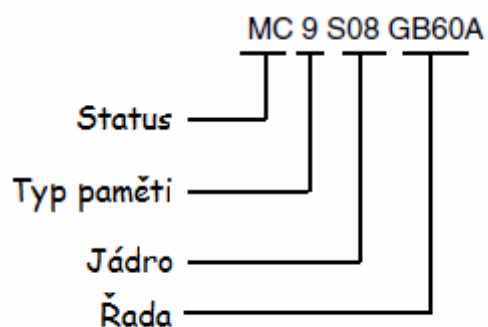
1.4.5 Zařízení v řadě MC9S08GB60

Pro přehled a srovnání parametrů je v tabulce uveden výčet hlavních znaků jednotlivých zařízení, které jsou k dispozici v této řadě.

Zařízení	Paměť Flash	Paměť RAM	TPM- Časovače	Počet I/O – vstupů/výstupů
MC9S08GB60A	60K	4K	3 kanálový 16-bitový časovač 5 kanálový 16-bitový časovač	56
MC9S08GB32A	32K	2K	3 kanálový 16-bitový časovač 5 kanálový 16-bitový časovač	56
MC9S08GT60A	60K	4K	dvojice 2 kanálových 16-bitových časovačů	39
MC9S08GT32A	32K	2K	dvojice 2 kanálových 16-bitových časovačů	39

Tab. č. 1.1: Zařízení z řady MC9S08 [3]

Vysvětlení označení názvů mikropočítače lze pochopit z následujícího obrázku. Zde jsou znázorněny informace, které název obsahuje.

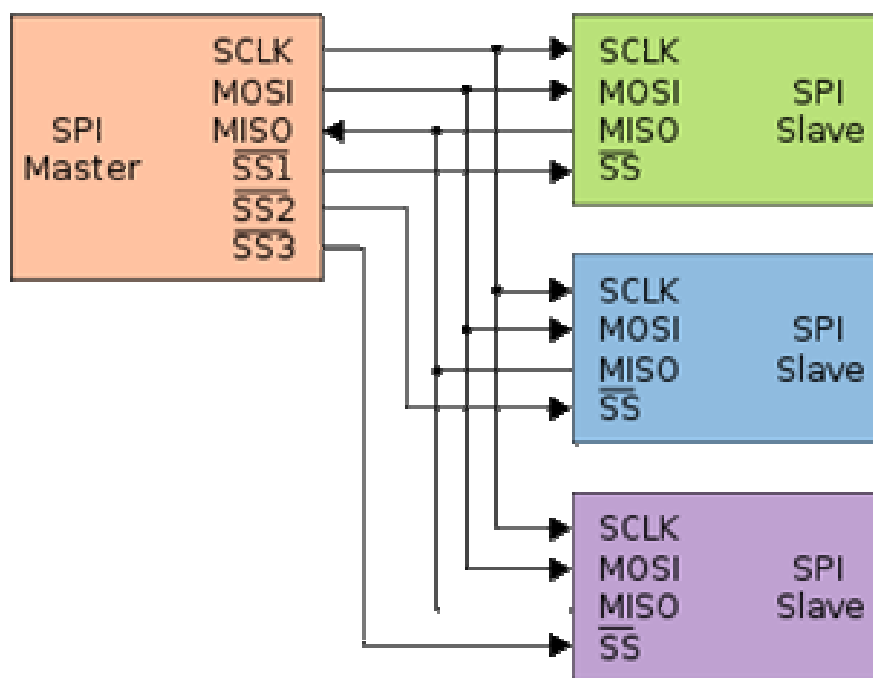


Obr. č. 1.4: Označení mikropočítače [3]

Sériové rozhraní SPI

SPI (Serial Peripheral Interface) je sériové periferní rozhraní používající se především pro připojení vnějších pamětí, A/D převodníků a jiných obvodů k mikropočítači. Dále lze rozhraní využít pro vzájemnou komunikaci mezi mikropočítači. U vybraných mikropočítačů lze SPI využít i pro programování jejich vnitřní paměti Flash. Tento způsob připojení je využit pro komunikaci mezi vývojovým kitem M68EVB908GB60 a zabezpečovacím modulem. [8]

Na následujícím obrázku je zobrazeno základní rozvržení systému využívající rozhraní SPI.

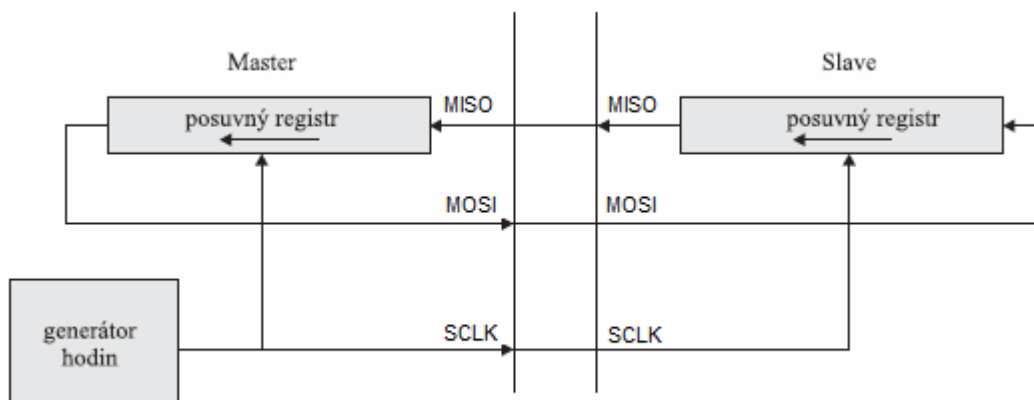


Obr. č. 1.6: Koncepce Sběrnice SPI [15]

- Datový výstup MOSI (Master Out, Slave In) obvodu Master je připojen na vstupy MOSI všech obvodů Slave.
- Datový vstup MISO (Master In, Slave Out) obvodu Master je propojen s výstupy MISO všech obvodů Slave.
- Výstup hodinového signálu SCLK je připojen na vstupy SCLK všech obvodů Slave.

- Každý obvod Slave, má vstup SS (Slave Select) pro výběr obvodu. Je-li SS v neaktivní úrovni, je rozhraní SPI daného obvodu neaktivní a jeho výstup MISO je ve vysoko impedančním stavu. Vstupy SS jednotlivých obvodů jsou samostatnými vodiči propojeny s obvodem Master. Je-li obvodem Master mikropočítač, bývají tyto vodiče připojeny k některému z jeho portů. Tak lze snadno vybírat obvod, se kterým má být v daném okamžiku vedena komunikace.

Po SPI sběrnici probíhají přenosy vždy mezi obvodem Master a některým z obvodů Slave. Oba obvody obsahují posuvné registry, komunikace mezi nimi je zobrazena na obrázku č. 1.7: Propojení Master a Slave



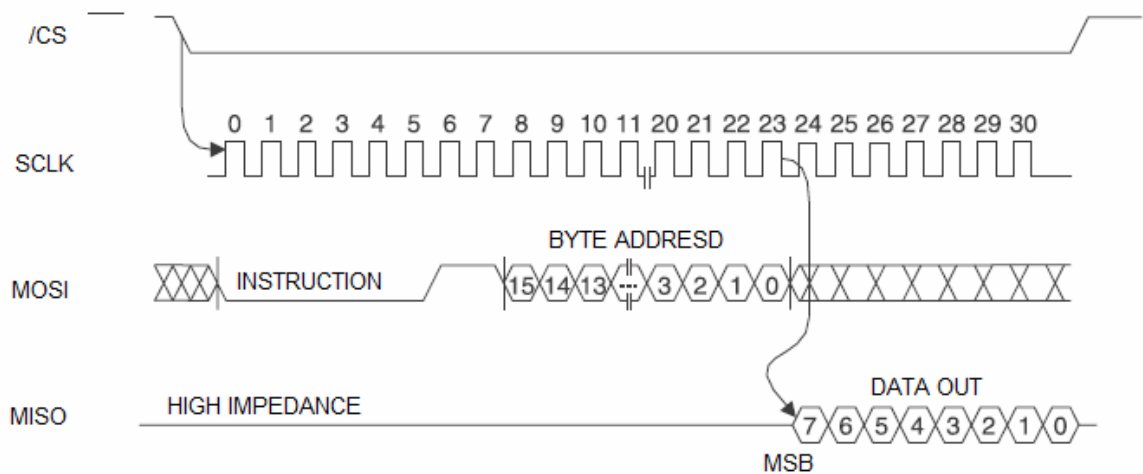
Obr. č. 1.7: Propojení Master a Slave [1]

Řízení posunu obou posuvných registrů má na starosti hodinový signál, který je generován obvodem Master. Klidová úroveň signálu SCLK a vztah mezi datovým a hodinovým signálem je dán parametry CPOL (Clock Polarity) a CPHA (Clock Phase). Je-li je u rozhraní SPI použit specializovaný řadič, jde většinou tyto parametry v řadiči nastavit. Je-li rozhraní SPI realizováno programově, musí být okamžiky změny úrovně datových a hodinových signálů zvoleny tak, aby přijímající obvod vzorkoval ustálená data. [8]

CPOL při začátku přenosu řídí, zda hodinový signál začíná na náběžné, nebo sestupové hraně. Pokud je $CPOL = 0$, pak hodinový signál začíná náběžnou hranou. Zatím co pokud je $CPOL = 1$, hodinový signál začíná hranou sestupnou. CPHA řídí přijímání dat z MOSI a MISO a to buď z náběžné, nebo sestupné hrany. $CHPA = 0$ znamená přijímání z náběžné hrany a $CHPA = 1$ přijímání dat z hrany sestupné.

Napětíové úrovně jednotlivých signálů rozhraní SPI jsou dána zvolenou technologií. Maximální frekvence hodinového signálu je 2 MHz.

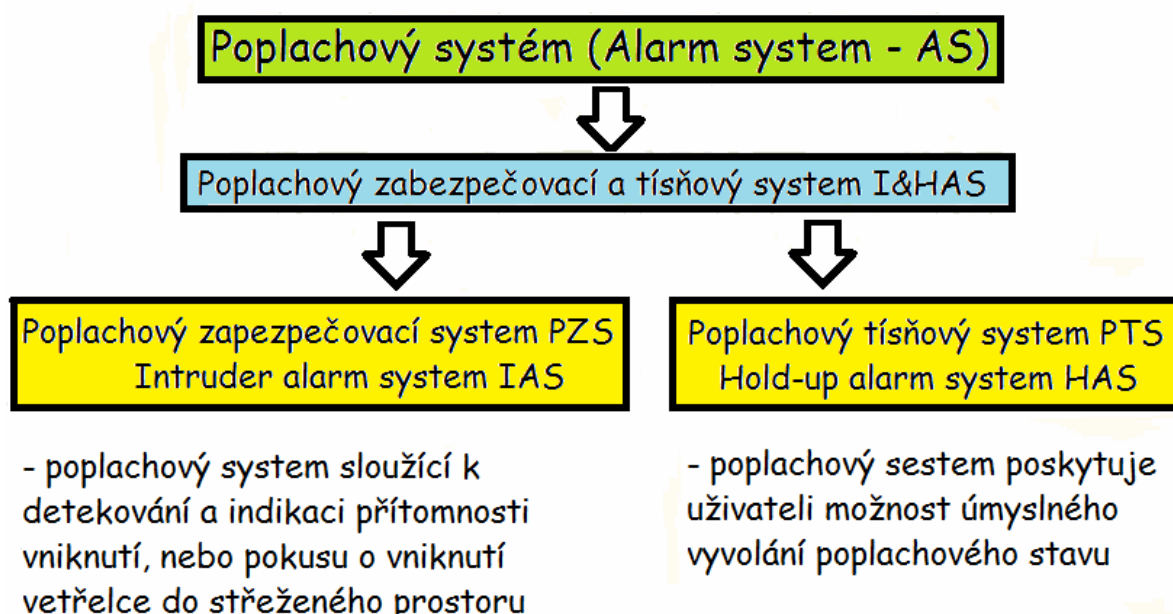
Jako příklad je uveden Obr. č. 1.9: Čtení dat z paměti pro SPI, kde je znázorněna komunikace se sériovou pamětí při čtení dat. Mikropočítač musí nejprve do paměti zapsat povel (čtení) a adresu dat. Potom jsou z paměti přečtena příslušná data. [8]



Obr. č. 1.9: Čtení dat z paměti připojené na SPI [8]

2 NÁVRH POPLACHOVÉHO ZABEZPEČOVACÍHO SYSTEMU

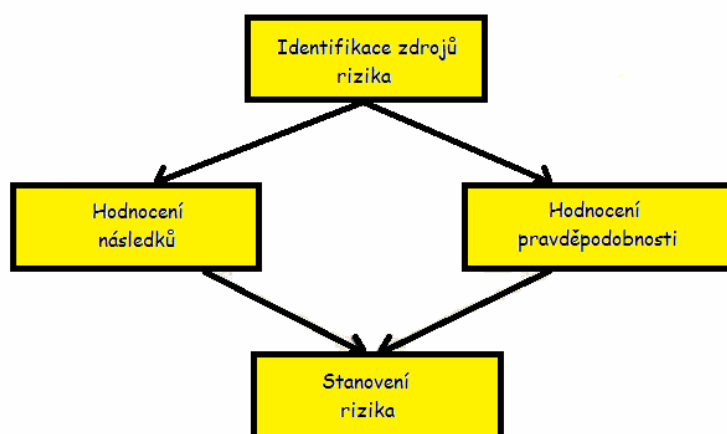
V této části bude popsán obecný postup při návrhu poplachového zabezpečovacího systému a jeho hlavní kritéria.



Obr. č. 2.1: Členění poplachových systémů [10]

2.1 Bezpečnostní posouzení objektu

Identifikace zdrojů rizik vychází z hodnocení následků ze v zniklého rizika a z hodnocení pravděpodobnosti vzniku rizika. Za pomocí těchto kritérií lze stanovit do jaké míry riziko hrozí.



Obr. č. 2.2: Kroky k analýze rizik [10]

2.1.1 Zabezpečované hodnoty - majetek

Potenciální riziko vloupání do střeženého objektu závisí na charakteru střeženého majetku, mezi které patří následující faktory:

Druh majetku: atraktivita pro pachatele, snadnost zpeněžení, nebezpečí vloupání.

Druh majetku: maximální hodnota přímé ztráty, následné výdaje související se ztrátou, osobní vztah k jednotlivým položkám majetku.

Objem majetku: snadnost/náročnost odcizení, snadnost/náročnost přepravy, (počítač, automobil), snadnost přístupu do střežených prostor (budova, areál, komunikace, vrátnice).

Historie krádeží: četnost a způsoby předcházejících incidentů.

Nebezpečí: zneužití střeženého majetku, nebezpečí majetku pro osoby, nebezpečí majetku pro okolí.

Poškození: vandalismus, zhářství. [9], [10]

2.1.2 Stavební dispozice

Při posuzování rizik jednotlivých hrozeb nebezpečí vniknutí, jako podklad ke zpracování systémového návrhu zabezpečení, je podstatným faktorem fyzická struktura objektů. Je kladeno za cíl identifikovat slabá místa v rámci stavební dispozice objektu. Mezi taková místa patří:

Konstrukce: stěny, střecha, podlahy, stropy a sklepy.

Otvory: okna, dveře, střešní světlíky, ventilační vstupy a další části pláště budovy (šachta, vstup výtahu atd.).

Provozní režim objektu: doba obývání objektu, přítomnost ostrahy, přístup veřejnosti, návštěvy, exkurze doprava.

Držitelé klíčů: přístup, dosažitelnost, evidence uložení.

Lokalita: riziko kriminality v oblasti, sousední budovy a jejich vliv (např. na vloupání), rychlost reakce na signalizaci poplachu a zásah, vzdálenost (vztah) sousedních obydlených objektů.

Stávající zabezpečení: kvalita a rozsah mechanického zabezpečovacího zařízení, kvalita a rozsah poplachového zabezpečovacího systému.

Historie: krádeží, loupeží, výhružek, počet předcházejících incidentů, způsob jejich realizace.

Místní legislativa a předpisy: bezpečnostní požadavky, požární předpisy, požadavky vzhledem ke konstrukci objektu.

Prostředí střeženého objektu: městská zástavba/venkov, typ osídlení, reliéf krajiny - přírodní překážky, nadmořská výška. [9], [10]

2.1.3 Minimální úroveň střežení IAS, HAS

Zpracovatel odhadne pravděpodobný způsob narušení v jednotlivých místech objektu podle výsledků o zabezpečovaných hodnotách a stavebních dispozic daného objektu. Na základě těchto analýz stanoví stupeň zabezpečení a skladbu systému. [9]

2.1.4 Ostatní vlivy

Je nutno brát v úvahu stávající a potencionální podmínky u střežených prostor. Tyto podmínky se rozdělují na vlivy působící uvnitř, nebo vně střežených prostor. Vlivy mající původ uvnitř střežených prostor jsou ovlivnitelné samotným uživatelem, proto lze ovlivnit výběr, umístění a nastavení detektorů použitých k zabezpečení daných míst. Na rozdíl vlivy působící vně těchto prostor jsou faktory bez možnosti ovlivnění uživatelem objektu. Oba dva druhy vlivů mají neblahý účinek na správnou funkci detektorů. Na příklad vodovodní plastové potrubí ve stěně, ve kterém proudí voda, by mohlo ovlivnit funkci mikrovlnného detektoru, nebo domácí zvířata, která by mohla mít vliv na pohybové detektory. U vnějších vlivů jsou to pak faktory jako na příklad železnice v blízkosti objektu, nebo vlivy počasí. [9], [10]

2.2 Základní dělení ochrany objektu

Kompletní bezpečnostní systém u kteréhokoli střeženého objektu či prostoru je dán vhodnou kombinací klasické, technické, fyzické a režimové ochrany.

Klasická ochrana:

tato ochrana představuje v obecném pojetí především klasické prvky objektu jako jsou zdi, střechy, podlahy, dveře a okna. V konkrétnějším pojetí se jedná především o prvky

mechanických zábranných systémů (MZS). V časovém měřítku představuje nejstarší typ ochrany proti páchání trestné činnosti. Měla by být součástí každého zabezpečovacího systému. Tvoří souhrn opatření pro přímé zabezpečení příslušného objektu mechanickými zábrannými systémy, které napomáhají tento objekt chránit. Mechanické zábranné prostředky jsou určité typy zábran a překážek, které říkají, jak dlouho je konkrétní prostředek schopen odolávat kvalifikovanému napadení dostupnými metodami a nástroji, tedy jejich průlomovou odolnost. Mezi tyto prostředky řadíme například uzamykací systémy, mříže, trezory, bezpečnostní folie, vrstvené sklo a jiné. [11]

Technická ochrana:

se skládá především z elektronických zařízení a prostředků. Ty napomáhají a jdou v souladu s prvky ochrany klasické. Patří zde poplachové zabezpečovací systémy (PZS), kamerové systémy (CCTV), elektrické požární systémy (EPS) a přístupové systémy (ACC).

Prostorové členění technické ochrany:

- obvodová (perimetrická, venkovní) ochrana,
- plášťová ochrana (vnitřní) ochrana,
- prostorová ochrana (plošná / klíčová, vnitřní / venkovní),
- předmětová ochrana (bodová, vnitřní),
- tísňová ochrana (prolíná se více prostory).

Fyzická ochrana:

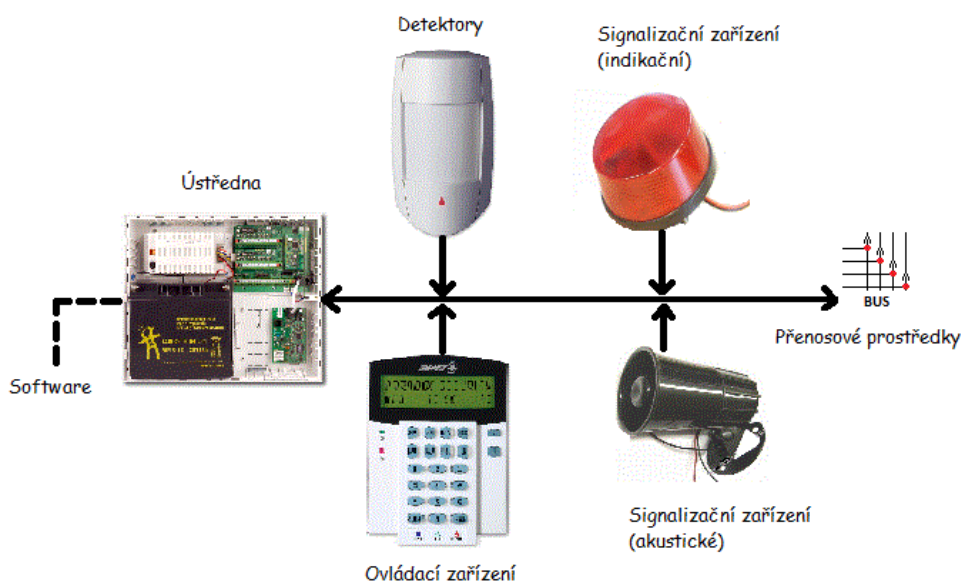
klasická ani technická ochrana by neměla téměř žádný význam, nebýt ochrany fyzické. Ta zaštiťuje oba z výše uvedených typů ochrany, neboť především na kvalitě její úrovně závisí výsledná účinnost ochrany osob a majetku. Klasická a technická ochrana by pozbyla významu kdyby prvky MZS nebyly pravidelně kontrolovány, nebo kdyby při vyhlášení poplachu neměl kdo analyzovat poplachový signál a účinně na něj reagovat. Fyzická ochrana je ze všech nejzákladnější, je tedy na místě vhodně kombinovat dostupné prostředky ochrany tak, aby bylo dosaženo co nejúčinnějšího bezpečnostního systému. Menší nevýhodou je vysoká nákladnost na finanční prostředky, za to je ale velmi efektivní. Fyzickou ochranu objektů lze provádět vlastními silami, strážnými, zaměstnanci soukromých bezpečnostních služeb, případně policií či armádou. [16]

Režimová ochrana:

jedná se o sjednocující a řídicí prvek celého komplexního zabezpečovacího systému objektu. Zajišťují ji především organizační a administrativní opatření. Jde o jednoznačné režimové pokyny, určení zodpovědnosti a pravomocí pro pracovníky zajišťující bezpečnost v objektu. Dále jsou zde uvedeny pokyny pro zaměstnance, návštěvy a strážní službu. [16]

2.3 Poplachový zabezpečovací systém

Poplachový systém slouží pro detekci a identifikaci přítomnosti, vniknutí nebo pokusu o vniknutí narušitele do střežených prostor objektů. Každý systém je složen z těchto základních prvků.



Obr. č. 2.3: Blokové schéma PZS

2.4 Stupně zabezpečení

Stupeň 1: Nízké riziko

Předpokládá se, že vetřelec, nebo lupič má malou znalost I&HAS a má k dispozici omezený sortiment snadno dostupných nástrojů.

Stupeň 2: Nízké až střední riziko

Předpokládá se, že vetřelec, nebo lupič má omezené znalosti I&HAS a používá běžného náradí a přenosných přístrojů.

Stupeň 3: Střední až vysoké riziko

Předpokládá se, že vetřelec, nebo lupič je obeznámen s I&HAS a má rozsáhlý sortiment nástrojů a přenosných elektronických zařízení.

Stupeň 4: Vysoké riziko

Používá se, má-li zabezpečení prioritu před všemi ostatními hledisky. Předpokládá se, že vetřelec nebo lupič je schopen, nebo má možnost zpracovat podrobný plán vniknutí a má kompletní sortiment zařízení včetně prostředků pro náhradu rozhodujících komponentů I&HAS. [10]

2.5 Třídy prostředí

Komponenty musí být použitelné v jedné z definovaných tříd prostředí I-IV. Musí pracovat správně, jsou-li vystaveny těmto stanoveným vlivům.

Třída	Název prostředí	Teploty, příklad
I	vnitřní	+ 5 °C až + 40 °C Obytné místnosti, obchodní domy
II	vnitřní-všeobecné	- 10 °C až + 40 °C Není stálá teplota, chodby, schodiště
III	venkovní	- 25 °C až + 50 °C Vně budovy, komponenty nejsou plně vystaveny povětrnostním vlivům, přístřešky, terasy
IV	venkovní -všeobecné	- 25 °C až + 60 °C Vně budovy, komponenty jsou plně vystaveny povětrnostním vlivům

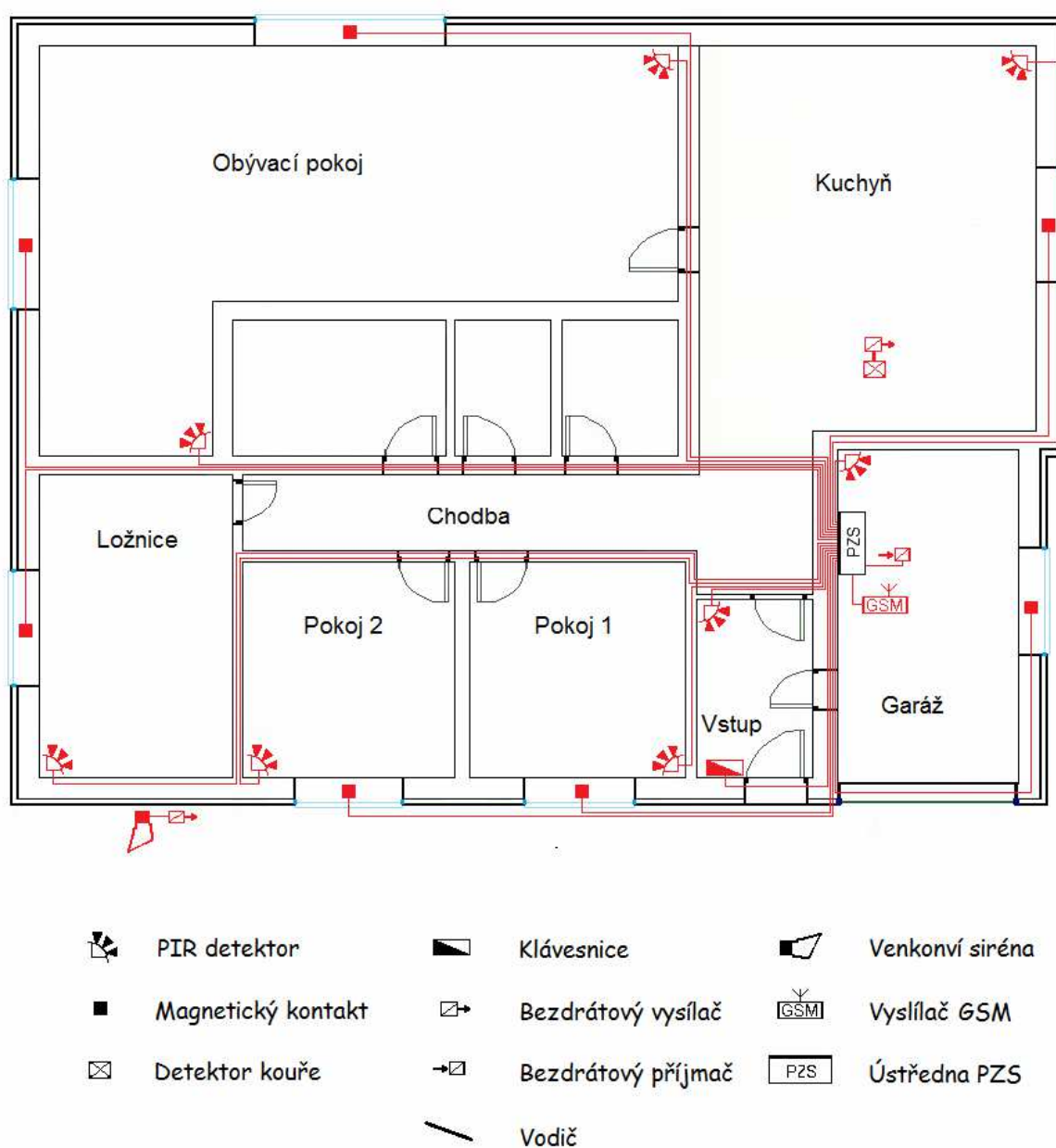
Tab. č 2.1: Třídy prostředí [10]

II. PRAKTICKÁ ČÁST

3 MODEL ZABEZPEČOVACÍHO SYSTÉMU

Jedním z bodů práce bylo navrhnout a vytvořit model zabezpečovacího systému, kde funkci detektoru mají simulovat tlačítka. Pro tento účel byl použit půdorys jednopodlažního rodinného domu, který lze vidět na Obr. č. 3.1: Výkres návrhu PZS. Následně byl u tohoto domu proveden náčrtek návrhu poplachového zabezpečovacího systému (PZS) a elektrické požární signalizace (EPS).

3.1 Návrh zabezpečení objektu

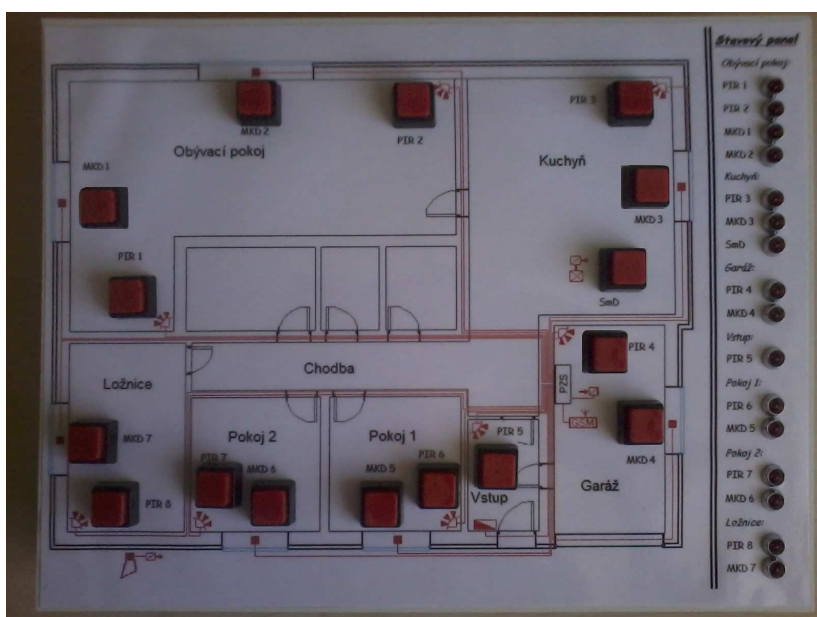


Obr. č. 3.1: Výkres návrhu PZS

Systém je navrhnut tak, aby chránil pouze vnitřní prostor objektu. Chráněny jsou veškeré prosklené plochy. Jedná se o střežení pomocí magnetických kontaktů, které mají za účel střežit nedovolenému otevření oken. Vnitřní prostory jsou střeženy pomocí PIR detektorů. Z prvků požární signalizace je zde v zastoupení kouřový detektor, který je umístěn v oblasti kuchyně. Z detekčních prvků je to vše a je to dáno z důvodu omezeného počtu těchto zařízení. Model má sloužit jako učební pomůcka tudíž muselo být přehlédnuto na rozměry a přehlednost návrhu. Veškeré prvky systému, až na dva, jsou připojeny pomocí kabeláže. Zbylé dva jsou bezdrátové a jedná se o detektor kouře a venkovní sirénu. Všechny prvky jsou připojeny k ústředně PZS, která je umístěna v prostorách garáže. Ovládá se pomocí klávesnice umístěné při vstupu do domu. K ústředně je dále připojen GSM modul, přes který lze zasílat SMS zprávy o stavu systému, nebo zprávy poplachové.

3.2 Realizace modelu

Návrh poplachového bezpečnostního systému byl použit jako podklad pro výrobu modelu. Veškeré detekční prvky, to jsou PIR detektory, magnetické kontakty a kouřový detektor, jsou v realizaci reprezentovány tlačítky. Jako venkovní siréna je využit bzučák vývojového kitu. K signalizaci je uživatelům nabídnut stavový panel, který je členěn podle jednotlivých místností v objektu. Každá místnost obsahuje určitý počet detektorů, které jsou pojmenovány a zároveň má každý detektor přiřazenou LED diodu. Vše je zapouzdřeno v krabici vyrobené z plastových desek.



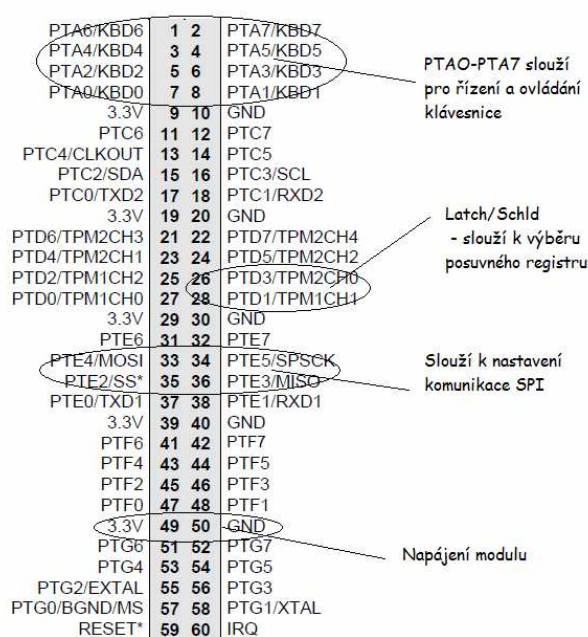
Obr. č. 3.2: Krabíčka Modelu

4 REALIZACE OVLÁDACÍHO ZAŘÍZENÍ MODELU

Model má simulovat reálné detektory pomocí stisku tlačítka, a to s pomocí vývojového kitu M68EVB908GB60. Proto bylo nutné vytvořit zařízení, které bude schopno s kitem komunikovat a zároveň reagovat na změny. Například při stisku tlačítka indikací rozsvícení LED diody, nebo rozezvučení bzučáku. Zároveň bude moct využívat integrovaný display, na kterém mohou být zobrazovány informace o stavu modulu. Dále ještě bude obsahovat obslužnou klávesnici pro zadávání jednoduchých příkazů ovládání modelu. Ke splnění těchto požadavků byly vytvořeny dvě desky plošných spojů. V následujících kapitolách budou znázorněny postupy při výrobě a schémata zapojení desek.

4.1 Deska I

První deska plošných spojů obsahuje dvouřadé pinové pole sloužící pro připojení k vývojovému kitu. Po naprogramování ovladačů zajišťuje komunikaci s klávesnicí. Dále do 14-pinového konektoru, pomocí kterých jsou spojeny obě dvě desky, přenáší signály pro řízení komunikace za pomocí SPI rozhraní, také signály pro výběr posuvných registrů a v neposlední řadě pak poskytuje napájení pro desku II. Vývojový kit je napájen 9 V, které jsou přivedeny také na Desku I, následně potom pomocí stabilizátoru LF 33 sníženy na 3,3 V, a to z důvodu, že celé zařízení pracuje na 3,3 V napěťových úrovních. Na následujícím obrázku je znázorněno využití I/O portů vývojového kitu.

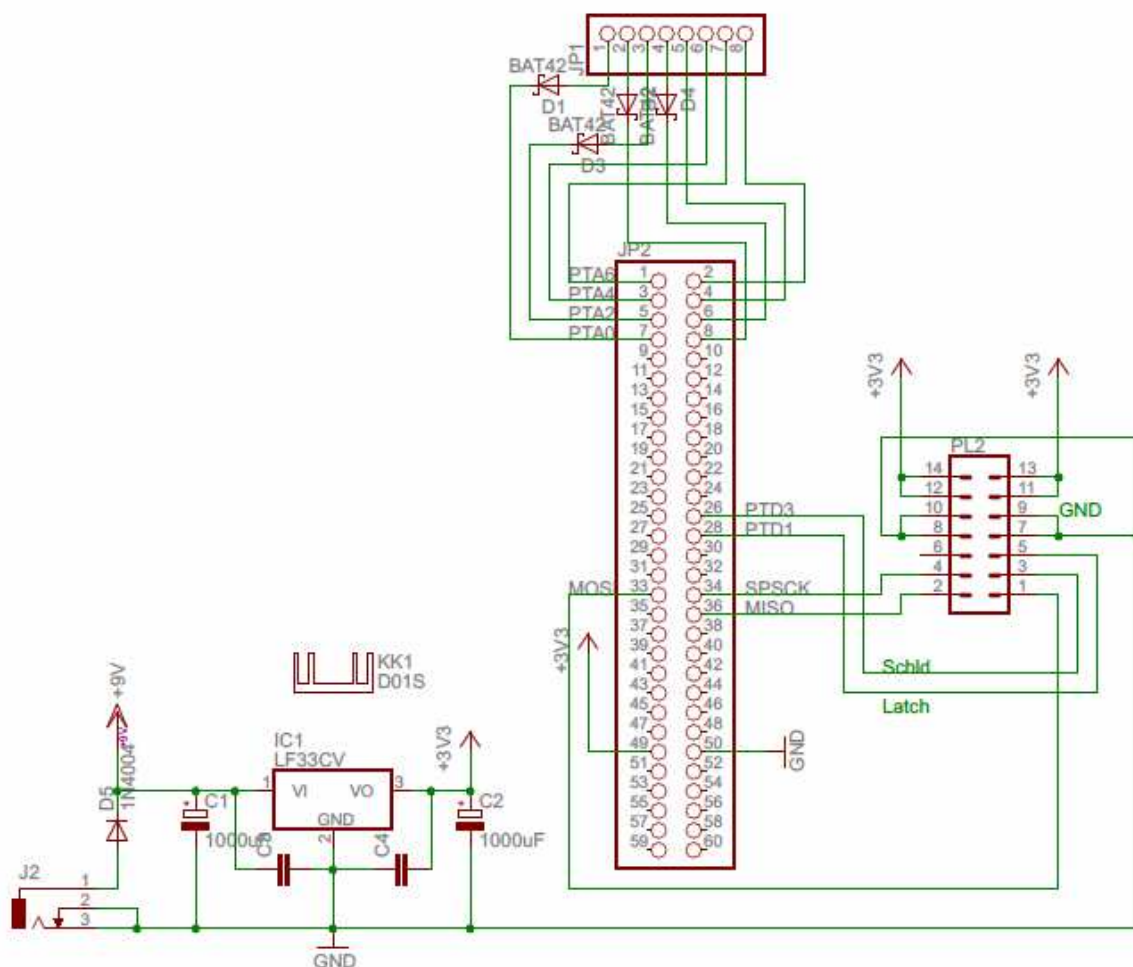


Obr. č. 4.1: Využití I/O pinů [1]

4.1.1 Schéma zapojení Desky

I

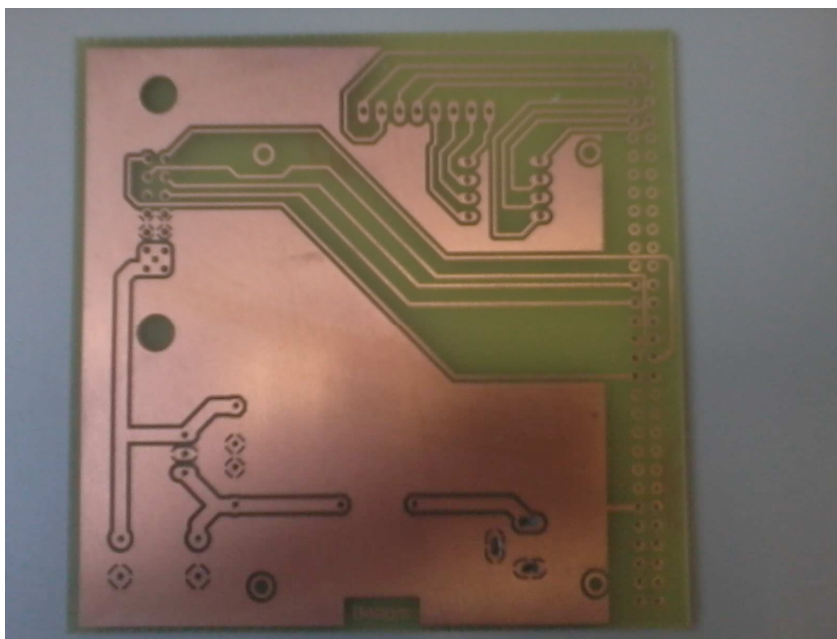
Schéma zapojení obou desek bylo navrženo ve školní licencované verzi programu Eagle. Všechny použité I/O porty jsou vyvedeny na 14-pinový konektor a pomocí plochého kabelu jsou spojeny s Deskou II. Ve schématu se nachází kolíková zdířka J2 pro napájení 9V, dále je přes filtrační kondenzátory připojen stabilizátor LF 33, ke kterému je připevněn chladič. Následně je to přivedeno na konektor. Na pinový konektor také přivádíme všechny potřebné výstupy z kitu. V vrchní části je připojena klávesnice.



Obr. č. 4.2: Schéma zapojení DeskyI

4.1.2 Výroba Desky I

Na obrázku č. 4.3: Deska plošného spoje I, lze vidět vyvrtanou desku plošných spojů, připravenou k osazení a připájení součástek k desce. V dalším obrázku č. 4.4: Osazená Deska, pak finální Desku I.



Obr. č. 4.3: Deska plošného spoje I



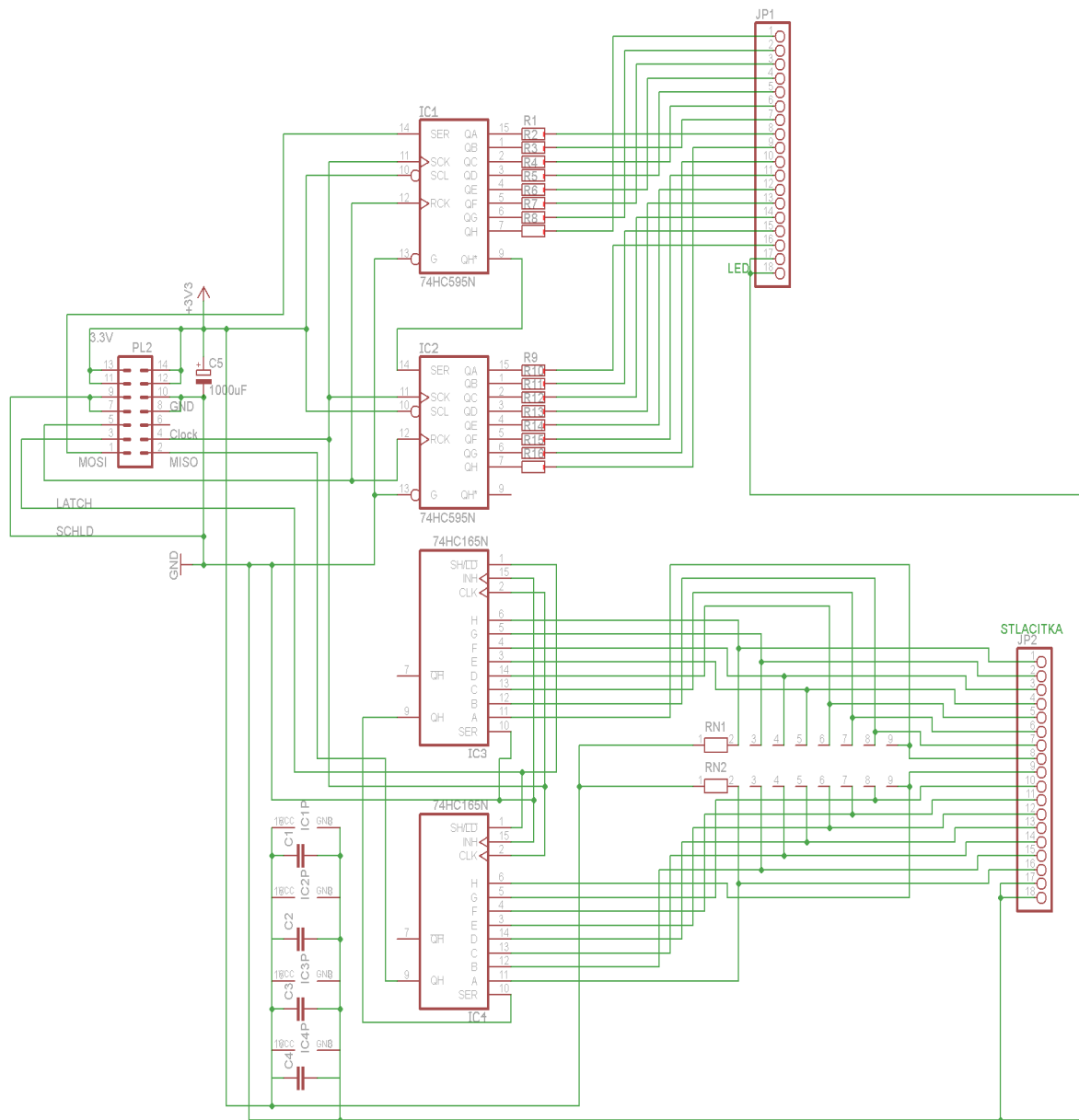
Osazená Deska

Obr. č. 4.4:

4.2 Deska II

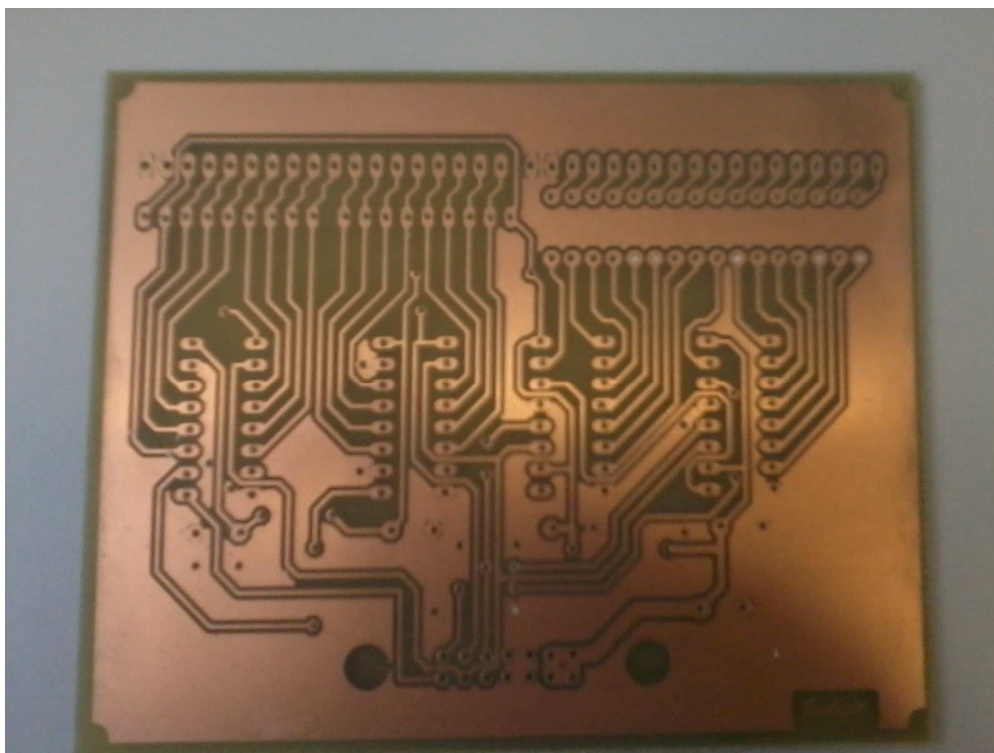
Na této desce plošných spojů se nacházejí dva posuvné registry typu PISO (paralel-in, seriál out), a jde o 74HC165 registr. Pomocí těchto prvků je umožněno snímání tlačítek, která jsou zapojena přes odporová pole na napěťovou úroveň odpovídající log 1. Dále se a desce nacházejí dva registry typu SIPO (seriál-in, paralel-out) se záchytným registrem a jde o součástku 74HC595. Ty mají za úkol umožnit rozsvěcování a zhasínání LED diod, které jsou k registům zapojeny přes rezistory o velikosti $R = 330 \Omega$. Tlačítka i LED diody jsou připojeny k Desce II přes pinovou svorku a za pomoci vodičů rozvedeny po krabici simulčního modelu, tak aby plnili funkci falešných detektorů, nebo indikace.

4.2.1 Schéma zapojení Desky II

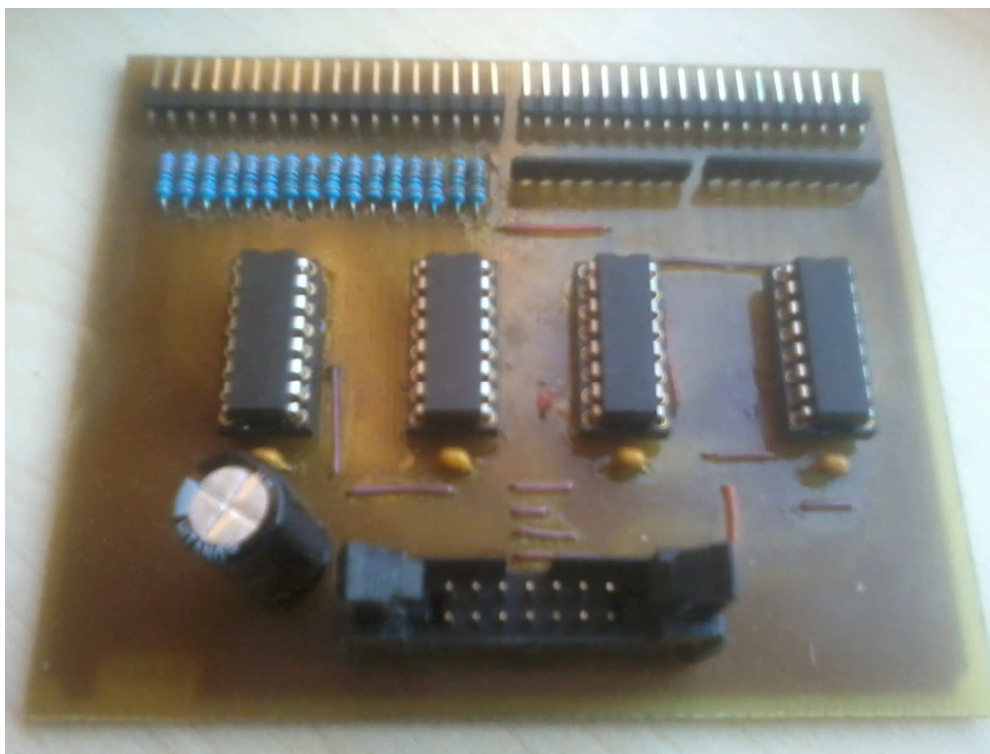


Obr. č. 4.5: Schéma zapojení Desky II

4.2.2 Výroba Desky II



Obr. č. 4.6: Deska plošného spoje II



Obr. č. 4.7: Osezená Deska II

5 PROGRAMOVÉ VYBAVENÍ

Podpůrné programové vybavení reprezentuje vytvoření podprogramů ve formě knihovny pro obsluhu modelu. Proto bylo zapotřebí naprogramovat ovládací podpůrné programy pro jednotlivé prvky modelu, které nám tuto knihovnu budou utvářet. Tyto „ovladače“ budou rozděleny do čtyř částí, a to inicializace, ovladač signalizace, ovladač klávesnice a ovladač tlačítek. Podle bodu zadání je zapotřebí vytvořit tyto podpůrné programy pro dva programovací jazyky, a to v C jazyce a v jazyce symbolických adres. V následujících kapitolách bude nejdříve představena knihovna podprogramů pro jazyk C a následně pak knihovna podprogramů pro jazyk symbolických adres.

5.1 Jazyk C

Ve zkratce C je nízko-úrovňový programovací jazyk, který počátkem 70. let 20. století vyvinuli Ken Thompson a Dennis Ritchie. Nejčastěji je užíván pro psaní systémového software, ale je velmi rozšířený i pro psaní nejrozličnějších aplikací. Je dostatečně rozsáhlý na většinu systémového programování jako jsou ovladače nebo jádro operačního systému.

5.1.1 Inicializace

```
void init(){  
    // SPI init  
  
    SPI1BR=0b10010101;           // SPI baudrate  
  
    SPI1C1=0b01010000;           // SPI  
  
    SPI1C2=0b00000000;           // SPI  
  
    PTDDD_PTDDD1=1;               // latch  
  
    PTDDD_PTDDD3=1;               // schld  
  
    //Klavesnice init  
  
    PTADD= 0b00001111;           // PTA nastaveni I/O  
  
    PTAPE= 0b11110000;           // PTA pull-up  
}
```

Inicializace SPI rozhraní - nastavení stavových registrů

- Nastavení stavového registru SPI
- nastavení přenosové rychlosti SPI
- nastavení pinu PTD1 jako výstupní pin Latch
- nastavení pinu PTD3 jako výstupní pin Schld

Inicializace klávesnice – nastavení pinu PTA do režimu vstup/výstup (I/O) a připojení pull-up rezistorů na sloupce klávesnice.

5.1.2 Ovladač signalizace

```
void zobraz(char led,char onOff){  
    if(led >= 1 && led <=16 && (onOff == 0 || onOff ==1)){           //  
        kontrola že je zadano jen to může byt zadano  
  
        led--;                // posunu index (indexuje se od nuly)  
  
        if (onOff == 1){      // chci rožnout  
  
            switch(led){  
  
            case 0:  
  
                sig = sig | 0x0001;  
  
                break;  
  
            case 1:  
  
                sig = sig | 0x0002;  
  
                break;                // toto se opakuje stále pro kazdou  
                pozici bitu  
  
            }  
  
        }else{                // chci zhasnout  
  
            switch(led){  
  
            case 0:  
  
                sig = sig & ~0x0001;
```

```

break;

case 1:

sig = sig & ~0x0002;

break;           // zhasinání led se také opakuje pro
každou pozici

    }

    }

    }

PTDD_PTDD3=0;           //vyber radice

PTDD_PTDD1=1;

while(SPI1S_SPTEF==0);   //prvni byte diod zapis
SPI1D=(sig & 0x00FF);

while(SPI1S_SPTEF==0);   //druhy byte diod zapis
SPI1D= ((sig & 0xFF00) >> 8) ;

cekej();

PTDD_PTDD3=1;

PTDD_PTDD1=0;

cekej();

}

```

Vytváří funkci, do které se zapisují dva vstupní parametry. První slouží k výběru příslušné LED diody a druhý pro nastavení stavu LED zapnuto, nebo vypnuto. Vstupní hodnoty jsou omezeny pro zadávání v rozsahu 1-16 což jsou čísla LED, a zároveň omezují zadávání stavu zap./vyp na 1/0.

5.1.3 Ovladač klávesnice

```

char charkey(void){
char cislo='N';
//prvni radek
PTAD = 0b00001110;           // PTA hodnoty I/O
    if(PTAD_PTAD4==0) {

```

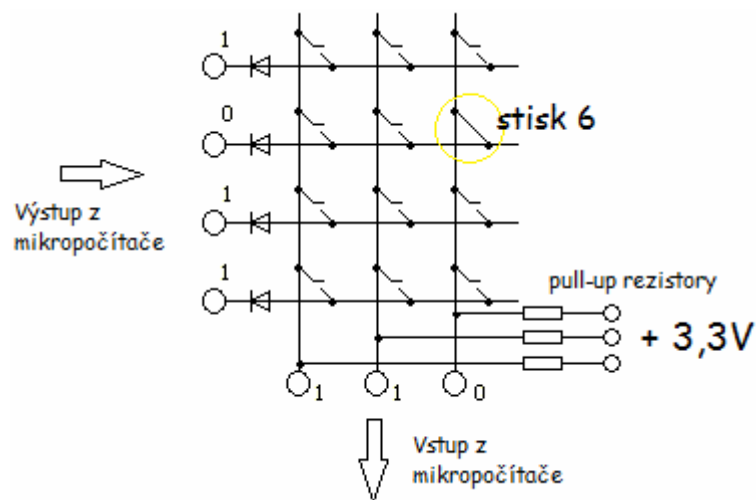
```
    cislo=2;
}
if(PTAD_PTAD5==0) {
    cislo=1;
}
if(PTAD_PTAD7==0) {
    cislo=3;
}
    //druhy radek
PTAD = 0b00001101;    // PTA hodnoty I/O
if(PTAD_PTAD4==0) {
    cislo=5;
}
if(PTAD_PTAD5==0) {
    cislo=4;
}
if(PTAD_PTAD7==0) {
    cislo=6;
}
    //treti radek
PTAD = 0b00001011;    // PTA hodnoty I/O
if(PTAD_PTAD4==0) {
    cislo=8;
}
if(PTAD_PTAD5==0) {
    cislo=7;
}
if(PTAD_PTAD7==0) {
    cislo=9;
}
    //ctvrty radek
PTAD = 0b00000111;    // PTA hodnoty I/O
if(PTAD_PTAD4==0) {
    cislo=0;
}
if(PTAD_PTAD5==0) {
    cislo='*';
```

```

}
if (PTAD_PTAD7==0) {
    cislo='#';
}
return cislo;
}

```

Voláním této funkce dostaneme nazpět znak stisknutý na klávesnici, a to tak jako je patrné z následujícího obrázku.



Obr. č. 5.1: Princip funkce klávesnice

Z výstupu mikropočítače jsou na řádky klávesnice primárně nastaveny hodnoty logické úrovně na 1. Na sloupcích jsou také nastaveny logické 1, a to za pomoci pull-up rezistorů připojených na kladné napětí 3,3 V, a tato úroveň je posílána na vstupy do mikropočítače. Aby mohlo být zjištěno stisknutí tlačítka na klávesnici, vysílá se z výstupu mikropočítače logická 0. Ta je opakovaně posílána na jednotlivé vstupy řádků klávesnice. Dojde-li ke stisku jakékoli klávesy, dojde zároveň i ke sepnutí kontaktu a tato logická nula se objeví i na kontaktu sloupce. Po té mikropočítač snadno zjistí stisknuté tlačítko tím, že porovná, do kterého řádku poslal logickou 0, a na kterém sloupci se logická 0 objevila. V případě obrázku se jedná o stisknutou klávesu číslo 6.

5.1.4 Ovladač tlačítek

```

unsigned int tlacitka(){
    char tlacitka;

```



```
char tlacitka1;

unsigned int result;

unsigned int zmacknuto;

PTDD_PTDD3 = 0;           // schld = 0, ulozeni stavu tlacitek do posuvneho reg.

PTDD_PTDD3 = 1;           // schld = 1

while(SPI1S_SPTEF==0);    // pockej na vyprazdneni vysilaciho bufferu

SPI1D = 0;                 // odesli na SPI libovolny 1B dat

while(SPI1S_SPRF==0);     // prvni pulka s tlacitkama

tlacitka=SPI1D;            // nacteni prvni pulky tlacitek

while(SPI1S_SPTEF==0);    // pockej na vyprazdneni vysilaciho bufferu

SPI1D = 0;                 // odesli na SPI libovolny 1B dat

while(SPI1S_SPRF==0);     // druha pulka s tlacitkama

tlacitka1=SPI1D;          // nacteni druhe pulky tlacitek


result = (256*tlacitka+tlacitka1);

    zmacknuto = ~result;

    if ((klv & zmacknuto)==zmacknuto){

        klv = klv & ~zmacknuto;

    }else{

        klv = klv | zmacknuto;

    }

    cekej();

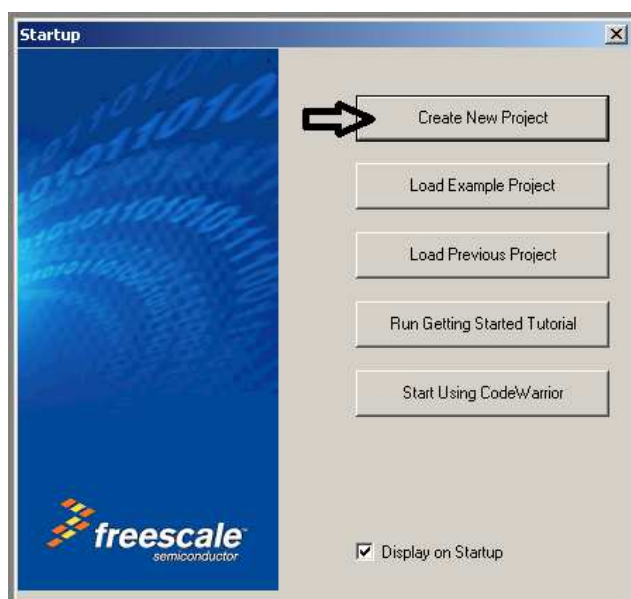
    return klv;

}
```

Stejnou formou jakou jsou uvedeny „ovladače“ v jazyce C, jsou vytvořeny i pro jazyk symbolických adres. Všechna programová vybavení jsou uložena na CD-ROM, jež je přiložen k bakalářské práci.

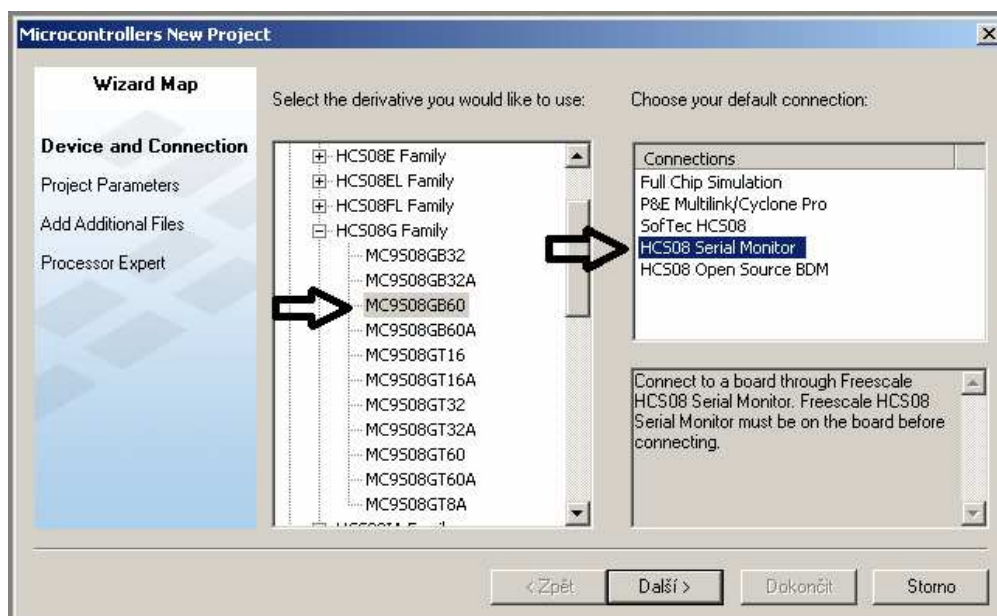
5.2 Pokyny pro programové vybavení

Programová knihovna slouží jako celek. Pro její správné fungování je při zpuštění v CodeWarrioru zapotřebí několik kroků. Ty budou vysvětleny na následujících obrázcích.

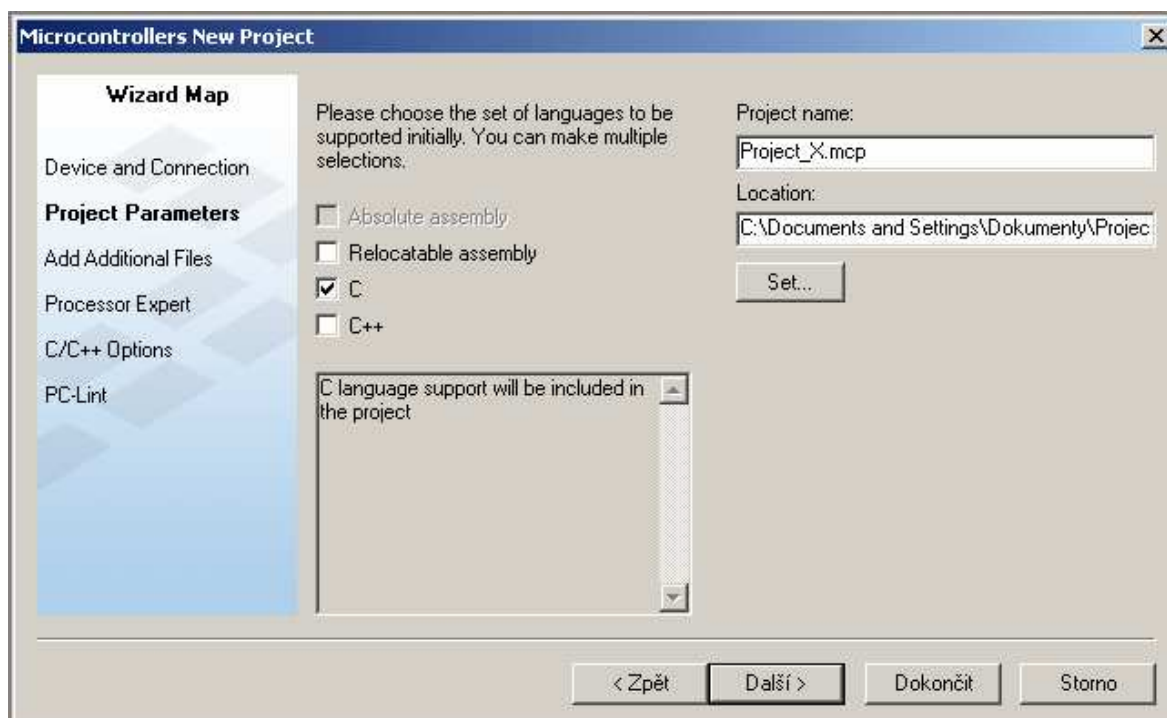


Obr. č. 5.2: Start CodeWarrior

Při zpuštění programu CodeWarrior se objeví následující okénko. Při spouštění nového projektu klikneme na „Create New Project“. V dalším kroku je za potřebí vybrat správný mikropočítač. A v okénku připojení vybrat „Serial Monitor“ jako lze vidět na obrázku.

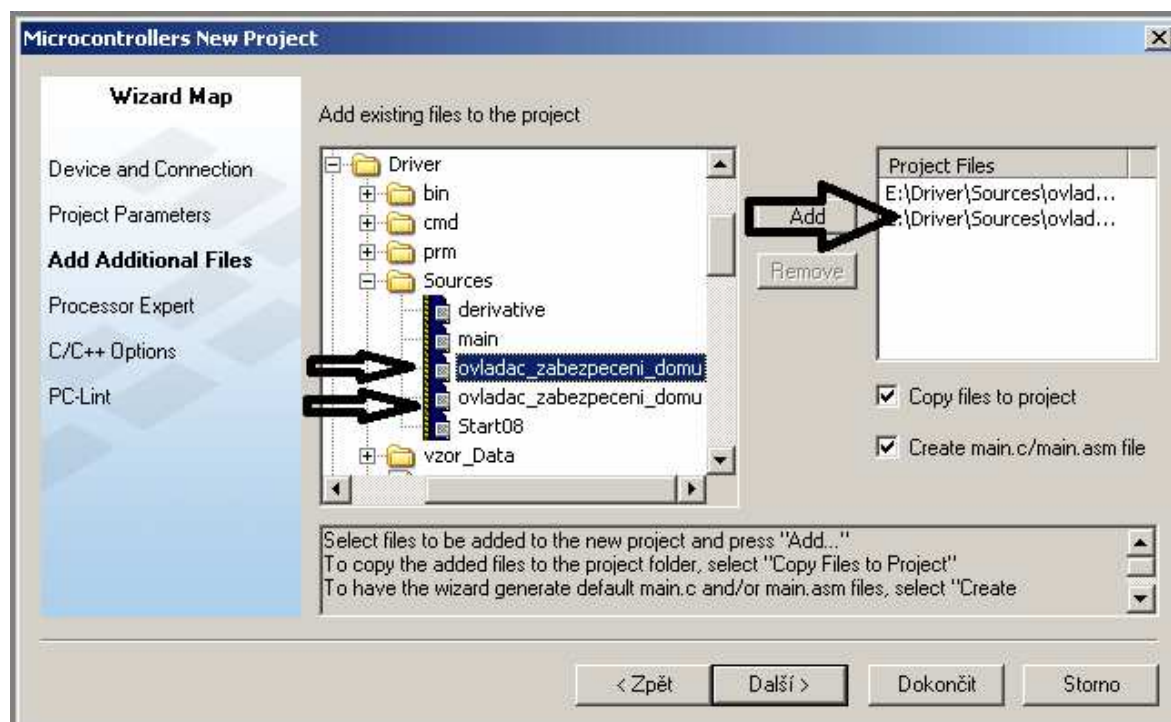


Obr. č. 5.3: Start Výběr mikropočítače



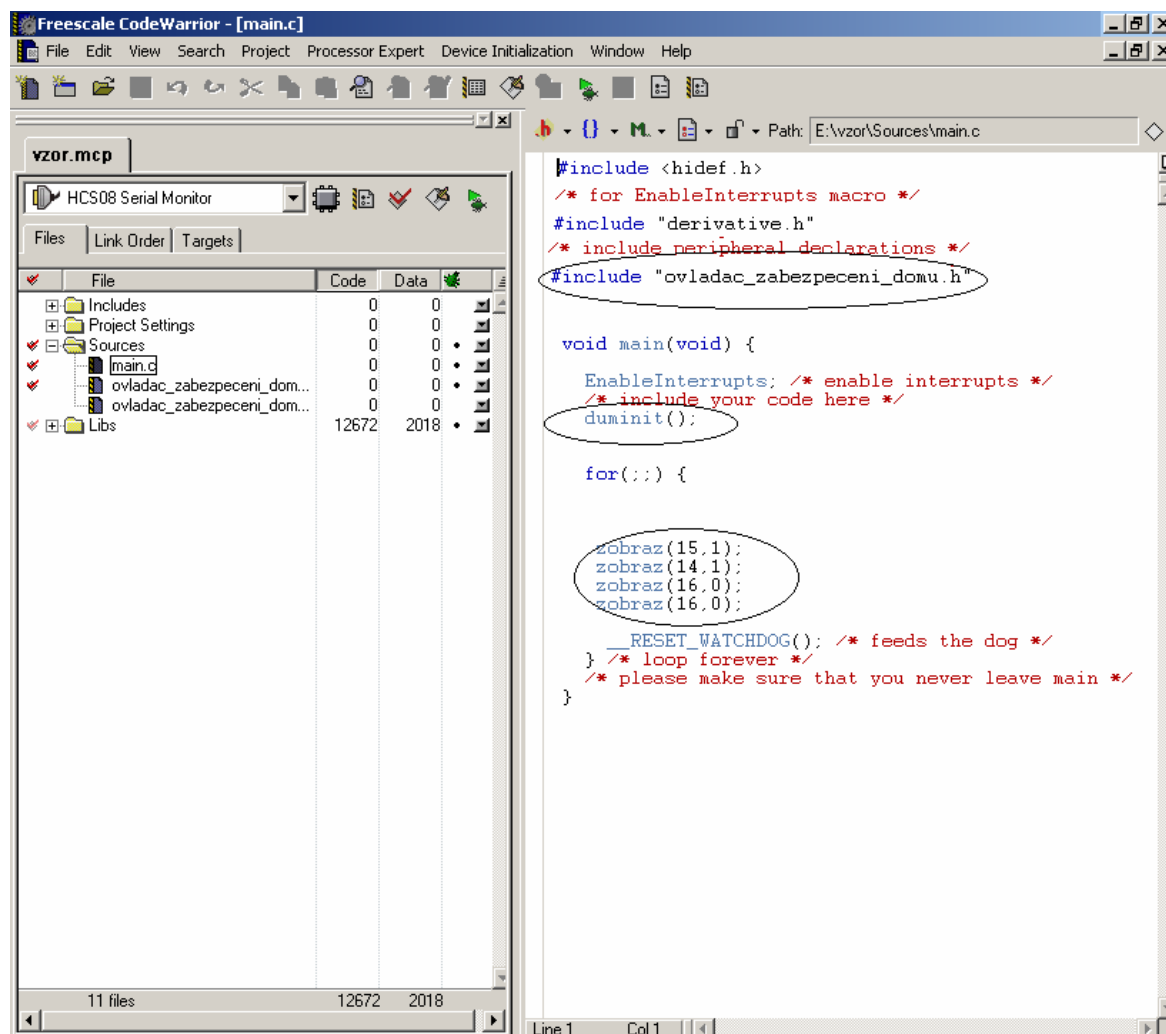
Obr. č. 5.4: výběr uložistě

V tomto kroku vybíráme programovací jazyk ve kterém chceme psát zdrojový kód programu a také název a místo uložení Vašeho projektu.



Obr. č. 5.5: Načtení ovladačů

Tento krok je velmi důležitý. Z balíčku přiložené knihovny ze složky Sources je nutno vybrat a načíst pomocí tlačítka Add dva soubory a to *Ovladac_zabezpeceni_domu.h* a *Ovladac_zabezpeceni_domu.c*. Po té co je dokončena celá operace zakládání projektu, zobrazí se následující okno jako na obrázku č. 5.6: Zadávání parametrů.



Obr. č. 5.6: Zadávání parametrů

Zde máme načtený hlavní soubor pro psaní kódu *main.c* a načtené ovladače *Ovladac_zabezpeceni_domu.h* a *Ovladac_zabezpeceni_domu.c*. Aby knihovna správně fungovala je nutné do hlavičky zapsat `#include "ovladac_zabezpeceni_domu.h"`. Dále je zapotřebí do hlavní části programu napsat příkaz `duminit()`. Po provedení těchto kroků jsou knihovna připravena k použití. Pro výběr, zhasínání, nebo rozsvěcování LED diod vypisujeme příkaz `zobraz(15,1);`, kde počtem příkazů `zobraz` můžeme vybrat libovolný počet LED. Údaje v závorce znamenají číslo LED diody 1-16 a druhá pozice je svít/zhasni 1/0.

6 UKÁZKOVÁ APLIKACE

Posledním z bodů zadání bylo vytvoření ukázkové aplikace s využitím vytvořené knihovny podprogramů, která bude demonstrovat funkci modelu. V této aplikaci je tedy za pomoci knihovny, jež je psána v jazyce C, utvořen program, který zde bude popsán. Program bude rozčleněn do jednotlivých bodů podle funkcí, které umí.

1. Při stisku příslušného tlačítka rozsvít' příslušnou LED diodu a zároveň zapni bzučák
2. Informace o stisku tlačítka a rozsvěcení LED diod ukaž na display
3. Umožnit vstup do systému pomocí hesla zadané na klávesnici
4. Pomocí klávesnice dokázat aktivovat pouze vybraná tlačítka (aktivace zón)
5. Pomocí stisku tlačítka na klávesnici vynulovat stav a vrátit do stavu původního

ZÁVĚR

Cílem této bakalářské práce bylo vytvořit model zabezpečovacího systému pro výuku předmětu Mikropočítače a PLC. Práce pojednává o mikropočítačích jako celku, jejich architekturách a instrukčních sadách. Je zde popsán vývojový kit M68EVB908GB60 od firmy Freescale, jeho hlavní funkce a technické parametry. Dále je zde popsána řada mikropočítače HCS08, který vývojový kyt obsahuje. Pro lepší pochopení komunikace mezi modelem obsahující posuvné registry a kitem je popsáno sériové periferní rozhraní (SPI). Druhý blok teoretické části je věnován postupu při návrhu poplachového zabezpečovacího systému. Je zde uvedeno bezpečnostní posouzení objektu z různých hledisek. Dále jsou pak popsány druhy ochrany objektu, stupně zabezpečení a typy prostředí pro použití komponentů.

V praktické části bakalářské práce je znázorněn návrh poplachového zabezpečovacího systému (PZS) s popisem jeho funkce. Tento návrh je využit jako podklad pro výrobu modelu, kde detektory nahradila tlačítka. V další části jsou popsány kroky postupu při návrhu a výrobě jednotlivých desek plošných spojů. Jsou zde uvedeny popisy funkce, schémata zapojení a fotodokumentace výrobků. Poslední část práce je věnována podpůrnému programovému vybavení. Zde jsou popsány jednotlivé podprogramy knihovny, které jsou napsány ve dvou programovacích jazycích a to Jazyk C a Jazyk symbolických adres. Dále je pak naznačen popis ukázkové aplikace, s využitím realizované knihovny, která slouží k demonstraci funkčnosti modelu jako celku.

ZÁVĚR V ANGLIČTINĚ

The aim of this bachelor thesis was to create a model of the signaling system for teaching the subject Microcomputers and PLC. The work deals with microcomputers as a complex, their architectures and instruction sets. There is described Development Board for Freescale MC9S08GB60, and also its main features and technological parameters. After that there is described a number of microcomputers HCS08 which includes development board too. For better understanding the communication between the model containing shift registers and the board is described serial peripheral interface (SPI).

The second part of the theoretical part aims at the process of designing security alarm system. There is the safety assessment of an object from different perspectives. In the following parts we can find the types of protection of the object, the security levels and types of environment for using the components. The practical part of my bachelor thesis represents the design of security alarm system (PZS) with a description of its function. This design is used as the base for the production of the model, where the detectors were replaced with buttons. The next part describes the steps of the method the design and production of individual printed circuit boards. There are descriptions of function, diagrams and photographs of products. The last part deals with the software support. There are described every single subprograms of the library which are written in two programming languages as language C and assembler language. It is also suggested the description of the sample applications using the implemented library, which is used for demonstration the functionality of the model as a whole.

SEZNAM POUŽITÉ LITERATURY

- [3] FREESCALE SEMICONDUCTOR. *HCS08 Microcontrollers: MC9S08GB60A Data Sheet* [online]. Rev. 2. 2008 [cit. 2012-05-15]. Dostupné z: http://cache.freescale.com/files/microcontrollers/doc/data_sheet/MC9S08GB60A.pdf?fp=1&WT_TYPE=Data%20Sheets&WT_VENDOR=FREESCALE&WT_FILE_FORMAT=pdf&WT_ASSET=Documentation
- [2] AXIOM MANUFACTURING. *M68EVB908GB60: Development Board for Freescale MC9S08GB60* [online]. Rev. 2. 2005 [cit. 2012-05-15]. Dostupné z: http://cache.freescale.com/files/soft_dev_tools/doc/user_guide/M68EVB908GB60UM.pdf
- [3] FREESCALE SEMICONDUCTOR. *HCS08 Family: Reference Manual* [online]. Rev. 1. 2007 [cit. 2012-05-15]. Dostupné z: http://cache.freescale.com/files/microcontrollers/doc/ref_manual/HCS08RMV1.pdf
- [4] JURÁNEK, Antonín a Miroslav HRABOVSKÝ. *EAGLE pro začátečníky /: uživatelská a referenční příručka* : 2. vydání. Praha: BEN - technická literatura, 2007, 191 s. ISBN 80-730-0213-2.
- [5] MANN, Burkhard. *C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy - linkery, práce s ATMEL AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky*. Vyd. 1. Praha: BEN, 2003, 279 s. ISBN 80-730-0077-6.
- [6] *Jednočipový počítač*. In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, [cit. 2012-05-15]. Dostupné z WWW: http://cs.wikipedia.org/wiki/Jednočipový_počítač.
- [7] *Procesory CICS a RICS: Studijní materiál pro předmět Architektury počítačů* [online]. Ostrava, 2010 [cit. 2012-05-15]. Dostupné z: <http://poli.cs.vsb.cz/edu/arp/down/procrisc.pdf>
- [8] *Sériová rozhraní SPI, Microwire, I2C a CAN* [online]. Plzeň, 2002 [cit. 2012-05-15]. Dostupné z: http://home.zcu.cz/~dudacek/NMS/Seriova_rozhrani.pdf
- [9] KINDL, Jiří. *Projektování bezpečnostních systémů I*. Vyd. 2. Zlín: Univerzita Tomáše Bati, 2007, 134 s. ISBN 978-80-7318-554-1.
- [10] JAN VALOUCH, Ph.D. *Projektování bezpečnostních systémů: Studijní materiál*. Ústav bezpečnostního inženýrství Fakulta aplikované informatiky, UTB Zlín, 2011.
- [11] ORSEC: EZS/I&HAS. [online]. [cit. 2012-05-24]. Dostupné z: <http://www.orsec.cz/cs/technika/produkty/ezs-i-has/>
- [12] *Vývoj architektury počítačů*. [online]. 2011 [cit. 2012-05-15]. Dostupné z: <http://statnice.obrys.cz/index.php?>

- [13] Architektura von Neumann. [online]. 2010 [cit. 2012-05-15]. Dostupné z: http://wiki.sps-pi.com/index.php/Architektura_von_Neumann
- [14] AP13172USLK with M68EVB908GB60. [online]. [cit. 2012-05-15]. Dostupné z: <http://forums.freescale.com/t5/8-Bit-Microcontrollers/AP13172USLK-with-M68EVB908GB60/td-p/97113>
- [15] Serial Peripheral Interface. [online]. 2012 [cit. 2012-05-15]. Dostupné z: http://cs.wikipedia.org/wiki/Serial_Peripheral_Interface
- [16] *Návrh zabezpečení rodinného domu v obci Vrbatův Kostelec* [online]. Zlín, 2011 [cit. 2012-05-23]. Dostupné z: <http://portal.utb.cz>. Diplomová práce. Univerzita Tomáše Bati ve Zlíně. Vedoucí práce Ing. Lubomír Macků, Ph.D.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

PLC	Programovatelný logický automat
ALU	Aritmeticko-logická jednotka
CU	Řídící systém
MSC	Mikropočítačový sestém
I/O	Vstupně/výstupní porty
RISC	Reduced instruction set computer
CISC	Komplex instruction set computer
DIP	Dual in-line package
CPU	Central processing unit
SPI	Seriál pheripheral interface
A/D	Analog/digital
MOSI	Master out- Salve in
MISO	Master in – Slave out
SCLK	Set clock
SS	Set slave
MZS	Mechanické zábranné systémy
PZS	Poplachový zabezpečovací systém
CCTV	Uzavřený televizní okruh
EPS	Elektrická požární signalizace
ACC	Systém přístupů
SMS	Short Message Service
LED	Light-Emitting Diode

SEZNAM OBRÁZKŮ

<i>Obr. č. 1.1: Princip-Von Neumannova architekt</i>	11
<i>Obr. č. 1.2: Princip-Harvardská architektura</i>	12
<i>Obr. č. 1.3: Vývojový kit M68EVB908GB60</i>	14
<i>Obr. č. 1.4: Označení mikropočítače</i>	17
<i>Obr. č. 1.5: Blokové schéma MC9S08GB60</i>	18
<i>Obr. č. 1.6: Koncepce Sběrnice SPI</i>	19
<i>Obr. č. 1.7: Propojení Master a Slave</i>	20
<i>Obr. č. 1.8: Význam parametrů CPOL a CPHA</i>	21
<i>Obr. č. 1.9: Čtení dat z paměti připojené na SPI</i>	21
<i>Obr. č. 2.1: Členění poplachových systémů</i>	22
<i>Obr. č. 2.2: Kroky k analýze rizik</i>	22
<i>Obr. č. 2.3: Blokové schéma PZS</i>	26
<i>Obr. č. 3.1: Výkres návrhu PZS</i>	29
<i>Obr. č. 3.2: Krabice Modelu</i>	30
<i>Obr. č. 4.1: Využití I/O pinů</i>	31
<i>Obr. č. 4.2: Schéma zapojení Desky I</i>	32
<i>Obr. č. 4.3: Deska plošného spoje I</i>	33
<i>Obr. č. 4.4: Osazená Deska</i>	33
<i>Obr. č. 4.5: Schéma zapojení Desky II</i>	35
<i>Obr. č. 4.6: Deska plošného spoje II</i>	36
<i>Obr. č. 4.7: Osezená Deska II</i>	37
<i>Obr. č. 5.1: Princip funkce klávesnice</i>	41
<i>Obr. č. 5.2: Start CodeWarrior</i>	43
<i>Obr. č. 5.3: Start Výběr mikropočítače</i>	43
<i>Obr. č. 5.4: výběr uložště</i>	44
<i>Obr. č. 5.5: Načtení ovladačů</i>	44
<i>Obr. č. 5.6: Zadávání parametrů</i>	45

SEZNAM TABULEK

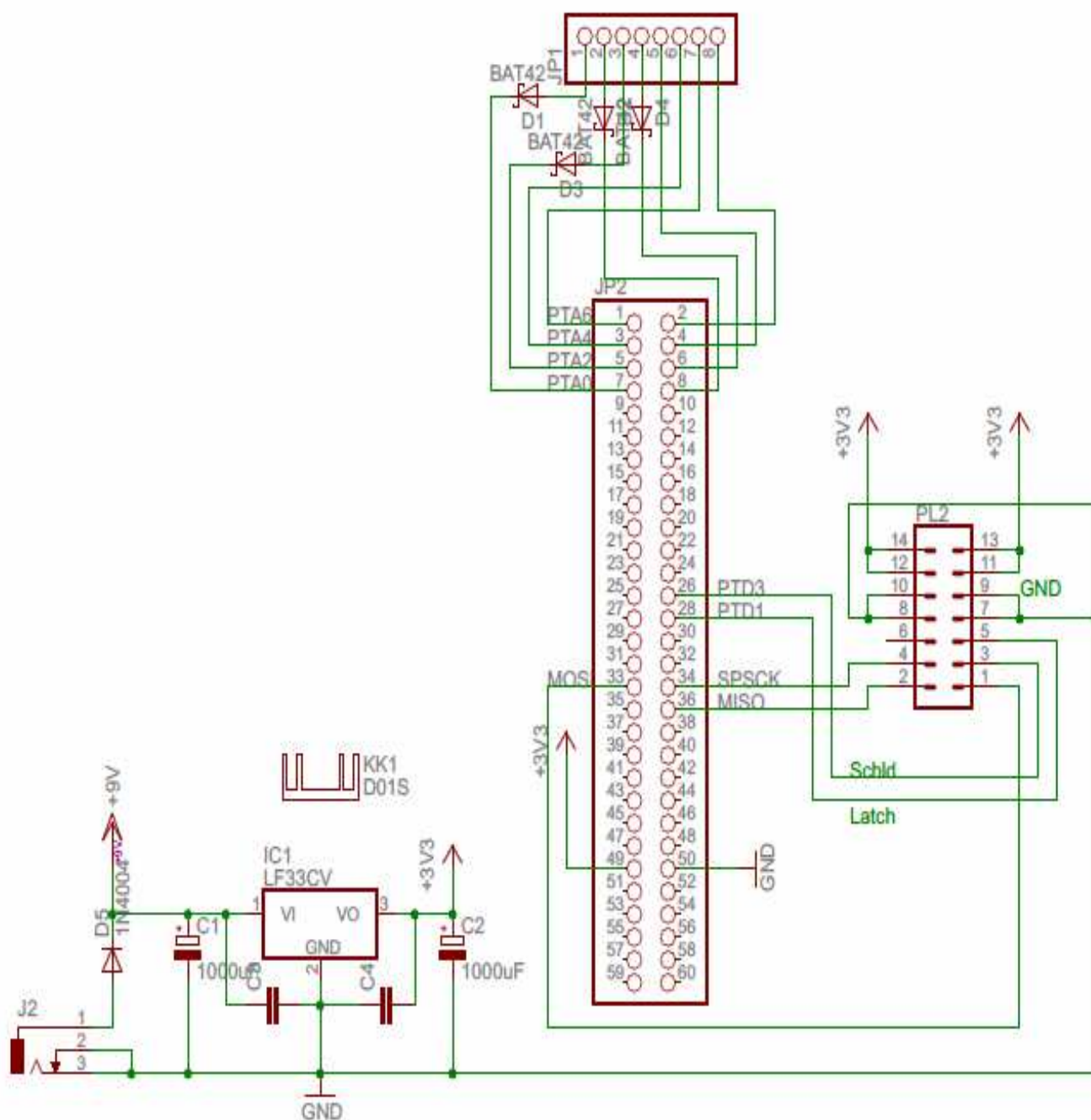
<i>Tab. č. 1.1: Zařízení z řady MC9S08</i>	<i>17</i>
<i>Tab. č 2.1: Třídy prostředí</i>	<i>27</i>

SEZNAM PŘÍLOH

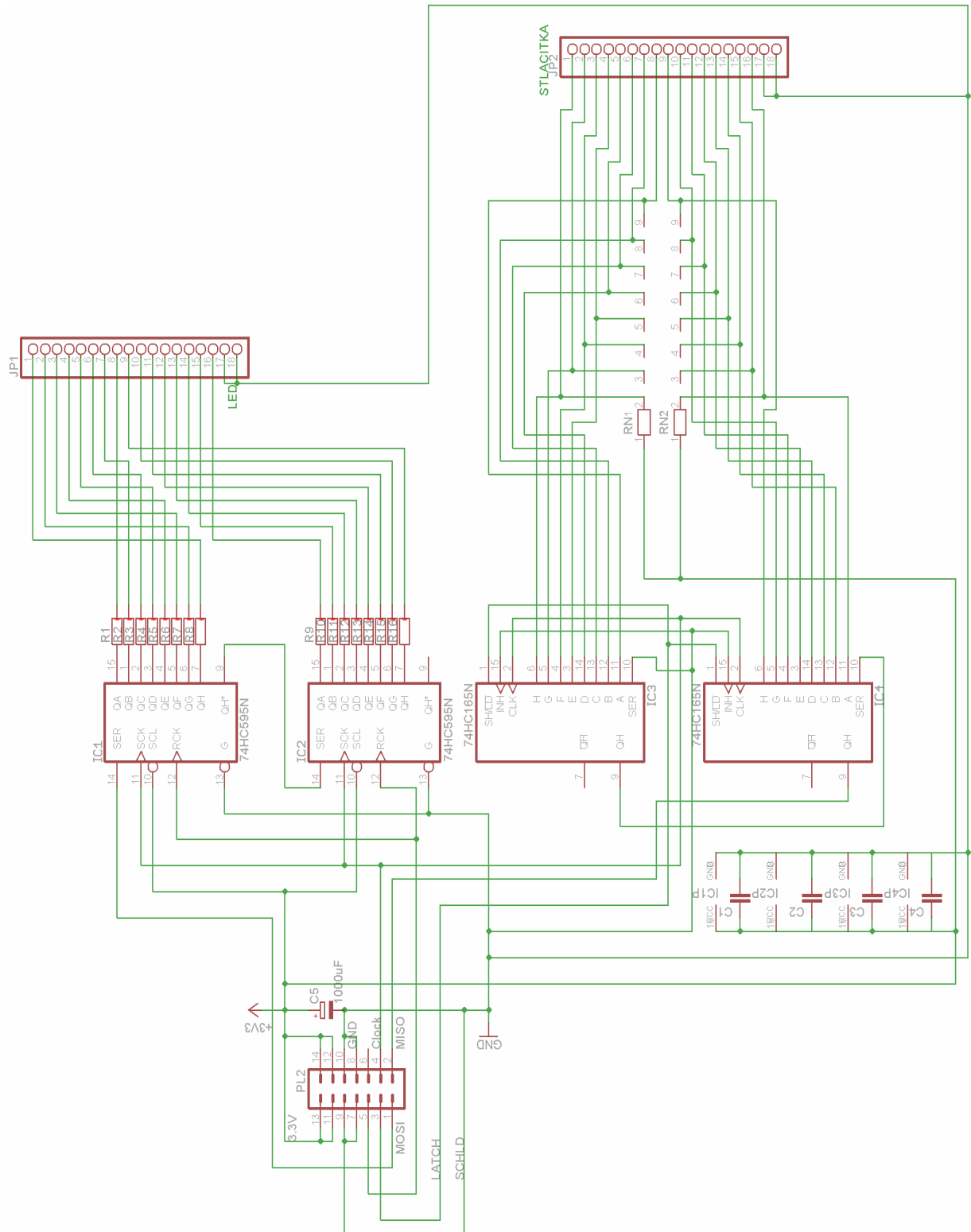
Příloha č. 1: Schéma zapojení Desky I

Příloha č. 2: Schéma zapojení Desky II

PŘÍLOHA Č I: SCHÉMA ZAPOJENÍ DESKY I



PŘÍLOHA Č II: SCHÉMA ZAPOJENÍ DESKY II



PŘÍLOHA Č III: BLOKOVÉ SCHÉMA HCS08

