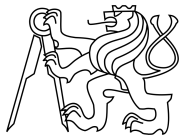
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Název:	Knihovna funkcí pro palubní informační systém v hromadném dopravním prostředku
Student:	Zbyněk Jakš
Vedoucí:	Ing. Pavel Kubalík, Ph.D.
Studijní program:	Informatika
Studijní obor:	Počítačové inženýrství
Katedra:	Katedra číslicového návrhu
Platnost zadání:	Do konce letního semestru 2016/17

Pokyny pro vypracování

Proveďte rešerši existujících řešení.
Vyberte vhodnou hardwarovou platformu pro realizaci vlastního palubního informačního systému.
Zařízení bude umožňovat zobrazovat informaci o poloze vozu s pomocí informačního displeje.
Zadávání vstupních hodnot bude umožněno s pomocí jednoduché klávesnice.
Informaci o aktuální a budoucí zastávce bude možné přehrát s pomocí zvukového zařízení v dopravním prostředku.
Zařízení bude umožňovat tisk jízdního dokladu na vhodné tiskárně.
Zařízení navrhněte a zrealizujte.
Pro navržené řešení vytvořte knihovnu funkcí umožňující ovládat dostupné periferie.
Pro předvedení funkčnosti celého zařízení vytvořte demo aplikaci využívající všechny dostupné periferie, zejména displej, zvukový systém a klávesnici.

Seznam odborné literatury

Dodá vedoucí práce.

L.S.

doc. Ing. Hana Kubátová, CSc.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
děkan

V Praze dne 15. února 2016

České vysoké učení technické v Praze
Fakulta informačních technologií
Katedra číslicového návrhu

Bakalářská práce

Knihovna funkcí pro palubní informační systém v hromadném dopravním prostředku

Zbyněk Jakš

Vedoucí práce: Ing. Pavel Kubalík, Ph.D.

15. května 2016

Poděkování

Rád bych poděkoval Ing. Pavlu Kubalíkovi, Ph.D. za vedení této práce a podělení se o své praktické zkušenosti. Dále bych rád poděkoval rodině a přátelům za psychickou podporu během psaní. V poslední řadě bych rád poděkoval své matce za gramatickou korekturu textu a svému otci za praktickou pomoc a zázemí pro manuální činnosti nezbytné pro tuto práci.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 15. května 2016 .....

České vysoké učení technické v Praze
Fakulta informačních technologií
© 2016 Zbyněk Jakš. Všechna práva vyhrazena.
Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Jakš, Zbyněk. *Knihovna funkcí pro palubní informační systém v hromadném dopravním prostředku*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.

Abstrakt

Tato práce se zabývá návrhem jednoduchého palubního počítače, který bude schopen řídit informační periférie v prostředcích hromadné dopravy, zejména autobusech a tramvajích. V úvodu ukáže již několik dostupných prostředků, které využívají dopravní podniky v České republice. Následuje ukázka metody návrhu systémů, která se v praxi pro velké projekty používá a tento princip v další části aplikuje. Po návrhu a propojení všech periférií do jednoho zařízení následuje tvorba knihovny pro jeho zjednodušení obsluhy. Systém bude předveden pomocí demoaplikace, která bude simulovat zařízení v provozu. Systém si klade důraz na jednoduché uživatelské ovládání. V závěru práce budou uvedeny výsledky testů zařízení.

Klíčová slova Arduino, LCD, přehrávání hlášení, tisk jízdenek, knihovna funkcí

Abstract

The main aim of this thesis is to design a simple, on-board computer, which is capable of controlling the information resources of public transport. The first part focuses on some of existing systems used by Czech transport companies. This part is followed by designing the on-board computer, using a method

which is typically applied in this practice. When the theoretical and hardware part of this thesis are finished, making a software library to control the connect peripherals would take place. The functionality of the whole device will be shown with a simple application, which simulates when using the device in traffic. Each part of the system will be tested and results will be shown at the end of this thesis.

Keywords Arduino, LCD, reports playback, tickets printing, library of functions

Obsah

Úvod **1**
Cíl práce 1

1 Analýza **3**
1.1 Úvod do problematiky 3
1.2 Existující řešení 3
1.3 Postup realizace 5
1.4 Rozbor komponent 7

2 Návrh **17**
2.1 Hardwarové požadavky 18
2.2 Arduino MEGA2560 20
2.3 Adafruit Thermal Printer CSN-A2-T 23
2.4 Arduino LCD display 16x2 25
2.5 Arduino klávesnice 4x4 26
2.6 SD Reader/Writer Module 28
2.7 Catalex Serial MP3 player v1.0 29

3 Realizace **31**
3.1 Hardwarové zapojení 31
3.2 Implementace řízení mp3 modulu 33
3.3 Implementace funkcí pro usnadnění ovládání 34
3.4 Implementace prototypu řídícího softwaru 39

4 Testování **43**
4.1 LCD displej 43
4.2 SD modul 43
4.3 Klávesnice 43
4.4 Tiskárna 44
4.5 Knihovna BusLib 44

4.6 Řídící software 44
4.7 Náklady 44

Závěr **45**

Literatura **47**

A Seznam použitých zkratek **51**

B Obsah přiloženého CD **53**

Seznam obrázků

1.1 Panel pro zobrazení přijezdu vozu v IPRM 4
1.2 Schema komplexního informačního systému 6
1.3 Panel pro zobrazení přijezdu vozu v IPRM 7
1.4 Zobrazovací panel Pragotron 8
1.5 Ilustrace panelu BUSE 9
1.6 Ukázka jehličkové tiskárny 10
1.7 Ukázka tepelné tiskárny 11
1.8 Ukázka GPS modulu 12
1.9 Ukázka mp3 modulu 14
1.10 Ukázka GSM modulu 15

2.1 Blokové schéma celého zařízení 18
2.2 Blokové schema zařízení pro tuto práci 20
2.3 Deska Arduino MEGA2560 22
2.4 Adafruit Thermal Printer CSN-A2-T 24
2.5 Arduino LCD display 16x2 26
2.6 Schema vnitřního zapojení klávesnice 27
2.7 Klávesnice 4x4 27
2.8 SD Reader/Writer Module 29
2.9 Catalex Serial MP3 module v1.0 30

3.1 Definování mp3 modulu 34
3.2 Funkce pro odesílání příkazu 35
3.3 Příkazy pro mp3 modul 35
3.4 Použité knihovny funckí 39
3.5 Postup inicializace klávesnice 40

Seznam tabulek

2.1 Technické parametry Arduino MEGA2560 21
2.2 Napájecí piny Arduino MEGA2560 22
2.3 Speciální funkce digitálních pinů 23
2.4 Technické parametry tiskárny Adafruit Thermal Printer CSN-A2-T 24
2.5 Technické parametry Arduino LCD display 16x2 25
2.6 Piny displeje Arduino LCD display 16x2 26
2.7 Technické parametry Arduino klávesnice 4x4 27
2.8 Technické parametry čtečky SD Reader/Writel Module 28
2.9 Technické parametry Catalex Serial MP3 player v1.0 29

3.1 Struktura příkazu pro mp3 přehrávač 33
3.2 Funkce mp3 přehrávače 34

4.1 Náklady na materiály pro zařízení 44

Úvod

Většina lidí během života využije služeb hromadné dopravy. Aby měl člověk jistotu, že nasedl do správného prostředku a zároveň věděl, že nespletl výstupní stanici, musí se řídit pomocí informačních hlášení, která dané linky nebo stanice nabízejí. Povinností cestujícího je mít platný cestovní doklad, a to buď v elektronické nebo tištěné podobě. Informace ve vozidle ovládá především řidič vozidla přes svou palubní přístrojovou desku, jejíž řízení musí být snadné a intuitivní. Malá východočeská firma Krkonošské metro z. ú., vlastnící několik dopravních prostředků a zobrazovacích panelů, si klade za cíl vytvořit vlastní informační dopravní systém, který bude poskytovat cestujícím informace pomocí retro prostředků. Celý projekt je velmi rozsáhlý, a proto ho firma rozdělila na více dílčích částí. Jednou z částí, která je pro běh takového systému nezbytná, je jeho jádro. To musí být schopné nezávislého provozu a musí zvládnout řídit veškeré připojené periférie, na kterých je systém závislý. Výsledkem je poté malý počítač s klávesnicí a displejem, umístěným na palubní desce u řidiče vozidla, do kterého jsou připojeny veškeré prvky, které slouží k poskytování služeb cestujícím.

Cíl práce

Cílem teoretické části práce je získání přehledu o existujících informačních dopravních systémech a přístupech, které společnosti používají při aplikaci systému do praxe. Hlavní náplní této práce je návrh jádra dopravního informačního systému společně s výběrem periférií pro vstup hodnot, zobrazení polohy, tisk lístků a možností přehrávání hlášení. Proto se teoretická část bude zabývat také průzkumem existujících prostředků plnící tyto funkce, které vlastní firma Krkonošské metro z. ú. Cílem tohoto průzkumu je získání znalostí o principu fungování každé periférie a stanovení hardwarových nároků, které budou nezbytné pro výběr hardwarové platformy.

Cílem praktické části je na základě zjištěných hardwarových požadavků vybrat vhodnou platformu a navrhnout a realizovat celý systém. Dalším cílem bude realizace ovládání systému pomocí knihovny funkcí, která bude využívat již existujících knihoven pro řízení jednotlivých periférií. Výsledkem celé práce bude demo aplikace využívající všechny periférie, která názorně předvede funkčnost všech součástí zařízení a zároveň předvede funkce vytvořené v knihovně.

Kapitola 1

Analýza

1.1 Úvod do problematiky

Každý člověk, který se přepravuje mezi dvěma body, musí mít informace o aktuálním stavu svého cestování. Ať už je to chodec, který zrovna jde na nákup do městského supermarketu, kde je třeba vědět, že zamířil do správné budovy (k čemuž mu stačí vlastní zrak), tak třeba řidič automobilu, který jede na pracovní cestu do úplně cizího prostředí a potřebuje správnou trasu najít pomocí mapy nebo v dnešní době již velmi často používanou navigací. Často ale mnoho lidí využívá způsobu přepravy pomocí hromadného dopravního prostředku, kde je nutno stejnou informaci podat všem cestujícím ve stejný okamžik. Takové informace zahrnují zvuková hlášení o poloze vozidla, zobrazení stavu cesty na velkém zobrazovacím panelu, který je viditelný pro všechny cestující v celém voze. Dále musí cestujícím umožnit pořídit si platnou jízdenku a další možnosti, které vedou ke zpříjemnění cesty. Ke sdělování takového typu informací se v dopravním průmyslu začaly používat tzv. informační systémy veřejné hromadné dopravy.

1.2 Existující řešení

Většina dopravních společností v ČR používá informační systém veřejné hromadné dopravy (dále jen informační systém) vlastní. Důvodem pro to jsou individuální nároky každé firmy pro konkrétní sdělovací prostředky na jiný konkrétní typ vozidla. Na českém trhu tedy neexistuje konkrétní typ informačního systému, který by fungoval jako univerzální nabídka pro české firmy zabývající se hromadnou dopravou. Je tedy zřejmé, že vyjmenování všech existujích řešení by bylo velice náročné, a proto budou uvedeny pouze některá existující řešení.

Obrázek 1.1: Panel pro zobrazení přijezdu vozu v IPRM

1.2.1 Integrovaný plán rozvoje města Mostu - DOPRAVA

Integrovaný plán rozvoje města Mostu - IPRM, jak uvádí zdroj [1], je projekt s cílem zavedení nového informačního systému a zkvalitnění služeb MHD na území města Mostu za účelem zvýšení atraktivnosti a využívání MHD co do přesnosti dodržování stanovených jízdních řádů a zvýšení informovanosti cestujících se zaměřením na návaznost a možnosti přestupů mezi jednotlivými linkami. Realizace projektu byla zahájena 6. 1. 2012 a ukončena 30. 11. 2015. Vozy Dopravního podniku města Mostu nyní nově disponují elektronickými panely pro zobrazování aktuální zastávky a zbytku cesty vozu včetně akustického hlášení zastávek. Ze stávající výbavy ve vozech zůstaly panely pro nákup jízdenek za pomoci čipových karet, které také dokáží zobrazit aktuální stanici. Zobrazení má na starosti řidič vozidla, který tak činí pomocí palubního počítače. Výjimkou jsou nové vozy tramvají, kde je zobrazení aktuální stanice nově řízeno pomocí polohy GPS. Vozy dále disponují automatem pro nákup jízdenek pro cestující, kteří nevlastní čipovou kartu. Jízdenky lze také koupit u řidiče, jehož ovládací panel má vestavěnou tiskárnu. Při příchodu revizora do vozu lze všechna zařízení pro nákup lístků zablokovat, aby se zabránilo černým pasažerům zareagovat na náhlou kontrolu. Další novinka se týká stanic. Ty nyní kromě tištěných jízdních řádů disponují elektronickým zobrazením příjezdů nejbližších vozidel podle polohy vozidla pomocí výpočtu.

1.2.2 Odbavovací a informační systém Dopravního podniku města Prahy

Dopravní podnik města Prahy pro veřejnost disponuje tištěnou verzí jízdních řádů. Provázáním offline dat jízdních řádů IDOS (Informační dopravní systém) od společnosti CHAPS spol s. r. o. s mapami Google, dokážou některé tramvajové zastávky, které jsou napojené na interní systém DORIS (Dopravní a řídící systém), zobrazovat na informačních panelech přesný online čas do příjezdu tramvaje na základě její aktuální polohy. Stanice metra navíc na hodinách v čele zobrazují čas, který uplynul od odjezdu poslední soupravy. Uprostřed nástupiště se nachází panel, který zobrazuje aktuální čas a čas do příjezdu další soupravy, jenž poskytují dispečerské systémy MATRA a ESA-M 11. Všechny vozy podniku umí hlásit linky a následující zastávky. Vnější přední část vozu zobrazuje na informačním panelu číslo linky a její konečnou stanici. Vnější boční část zobrazuje tytéž informace a navíc k tomu ukazuje i následující zastávku. Zadní část zobrazuje pouze číslo linky. Vnitřek vozu ukazuje cestujícím trasu jízdy včetně následujících zastávek. Samozřejmostí jsou také i označovače jízdenek. Systémy pro každý typ vozidla realizovala pokaždé jiná firma. Tramvajový systém instalovala firma APEX, s. r. o., autobusový systém instalovala firma MYPOL, v. o. s. a systém pro metro instalovala firma BUSE Blansko, s. r. o.

1.3 Postup realizace

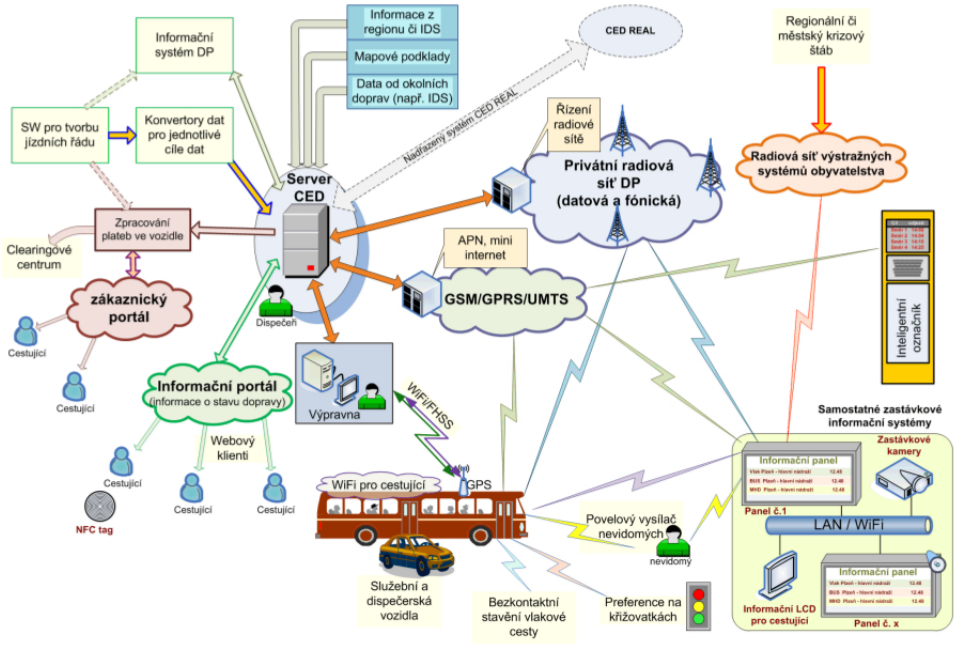
Základem pro realizaci komplexního dopravního informačního systému je vize. Ta určuje celkovou představu o výsledku. Na základě vize dopravní společnost vytvoří projekt z osobních zkušeností autorů a systémů z dalších měst. Projekt musí obsahovat mimo jiné blokové schéma, technické požadavky jednotlivých součástí a úplnou technickou zprávu. Pokud se takto připravený projekt schválí, je vyhlášeno výběrové řízení na dodavetele. Potencionální dodavatel odevzdá zprávu o technických součástech, které je schopen poskytnout a vyhovují požadavkům projektu. Dopravní společnost si podle zpráv vybere nejvhodnějšího dodavatele. Posledním krokem je kompletace. Kompletaci mohou provádět 3 subjekty. V první řadě to může být samotný dodavatel součástek. Další možností je na kompletaci najmout jinou externí firmu. Poslední variantou je, že si systém zkompletuje sama dopravní společnost.

1.3.1 Návrh a kompletace

Příklad schématu celého systému ukazuje obrázek 1.2:

Podle zdroje [2] je centrem řízení CED neboli centrální dispečink zpracovávající informace předjetí a zpoždění dopravy – jinými slovy o poloze vozidla.

Prvním důležitým krokem je zprovoznění informačních kanálů. Informační kanály pro cestující jsou:



Obrázek 1.2: Schema komplexního informačního systému

[2]

1. Zastávkové informační panely
2. Informační panely ve vozidlech
3. Internetové stránky

1.3.2 Návrh a kompletace

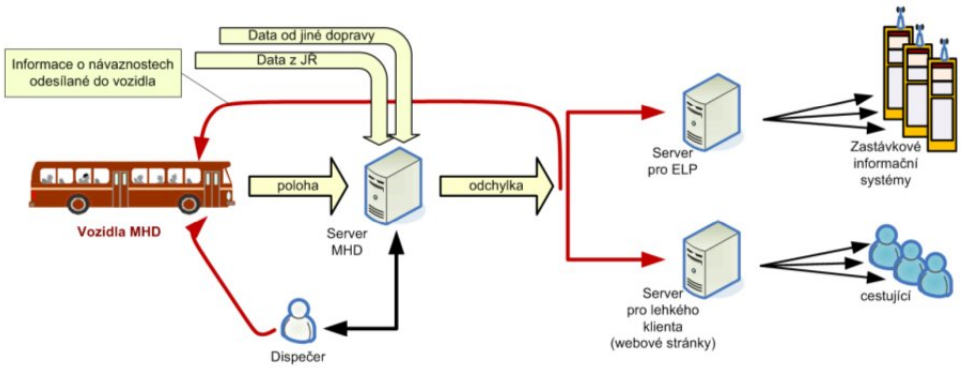
Příklad schématu celého systému ukazuje obrázek 1.3:

Dále je nutné připravit pro cestující data. Příprava má podle [2] následující postup:

1. Tvorba a zpracování jízdních řádů (organizace dopravy)
2. Příprava dat pro informační systémy vozidel (panely, displeje a hlášení), zastávkové systémy (off-line data a scénáře) a pro webové stránky
3. Nahrávání do cílových zařízení

Dalším krokem je příprava dat spojená s odbavením cestujících, kterou [2] uvádí jako:

1. Tvorba a zpracování jízdních řádů (organizace dopravy)



Obrázek 1.3: Panel pro zobrazení přijezdu vozu v IPRM

[2]

2. Tvorba ceníků a její přiřazení jednotlivým jízdám a trasám

3. Nahrání do vozidla a na www stránky

4. Řešení nákupu jízdenek (vozidlo, portál, předprodej)

5. Zpětné získání dat z vozidla a jejich zpracování

6. Příprava dat pro clearingové centrum

Vhodnou věcí ke zvážení při návrhu jsou doplňkové informační systémy. Zdroj [2] uvádí tyto možnosti:

1. Reklama ve vozidlech

2. Varovné a výstražné systémy obyvatelstva

3. Obecné informace z města či regionu

4. Informační systémy pro nevidomé

5. Internet do vozidel či na zastávky

1.4 Rozbor komponent

V následující části budou popsány komponenty, které se využívají v praxi, a zároveň je vyžaduje firma Krkonošské metro, z. ú. pro výsledný projekt.

Obrázek 1.4: Zobrazovací panel Pragotron

[5]

1.4.1 Pragotron

Zdroj [3] uvádí, že Pragotron je překlápěcí zobrazovací panel, vyráběný od roku 1969. Jeho původní uplatnění bylo informování cestujících Československých státních drah, nicméně se začaly používat i v dalších veřejných institucích, jako např. pošty. Podle [4] je základním prvkem informačního zařízení jednotka, což je nejmenší část, kterou lze samostatně řídit. V jednotce se nachází předtištěné listy, kde vždy jeden list je překlopen do polohy, ve které je viditelný průzorem na přední straně, jak píše zdroj [3]. Překlápění listů se provádí pomocí odeslání kódu, který přivede napětí na sadu vodičů, jejichž připojení k vodivé cestě spouští elektromotor. Ten zůstává spuštěný do té doby, dokud je alespoň jedna vodivá cesta aktivní. Otáčení motoru tedy vodivou cestu mění s každou otáčkou. Jednotky jsou různě velké a používají se převážně 2 typy listů - textové a znakové. Existují ale i listy s předtištěnými obrázky či symboly. Seskupením jednotek (nejčastěji do řádků a sloupců) do jednoho panelu vzniká tabule. Ta se řídí v praxi pomocí ovládacího pultu, který vysílá specifický kód. Příkladem kódu je 3 z 8.

1.4.2 BUSE BS110

Ve zdroji [6] jsem nalezl, že se jedná o zobrazovací část informačního systému pro městskou hromadnou dopravu, která je určena k zabudování do

Obrázek 1.5: Ilustrace panelu BUSE

[9]

kabiny autobusu, kde slouží k zobrazení informací o lince spoje a stanici. Zobrazovací technologie je podle [7] realizována pomocí elektromagnetických „štripsů“ DOT. Pixel je realizován pomocí malého magnetického disku o průměru 10mm. Panel se skládá z pěti zobrazovacích segmentů s rozlišením 28x19 bodů. Celkově tedy panely pokrývají plochu s rozlišením 140x19 bodů, což činí 2660 pixelů. Plocha disku je z jedné strany matně černá a ze strany druhé polepena reflexní fólií. K překlopení terče dochází na základě krátkého proudového impulzu, na jehož konci se terč otočí do požadované polohy, v níž může setrvat neomezeně dlouhou dobu. Směr protékání proudu určuje, ve které poloze pixel zůstane. Jednotka je napojená na sběrnici IBIS, která se podle zdroje [8] stala přirozeným standartem pro evropskou veřejnou dopravu, včetně naší. IBIS je napojená na sérioparalelní převodník, který obsahuje protokol, zvaný IPIS, pro snadnější řízení panelu. Protokol je součást výrobního tajemství a pro jeho využívání je potřeba originální software od firmy BUSE.

1.4.3 Jehličková tiskárna

Tisk za pomocí jehliček je jedna z nejstarších a nejpoužívanějších technologií počítačového tisku [10]. Principem tisku je, podobně jako u psacích strojů, otisk grafiky na tiskové hlavě, která je zde složená z jehliček namočených do inkoustu (odtud anglický název dot matrix print, neboli tisk pomocí matice bodů).

Během tisku dochází k mechanickému dotyku jehliček s papírem. Toho se v minulosti dříve hojně využívalo pro znásobení počtu tisknutých kopií za pomoci kopírovacího papíru [10]. Ten se přikládal na rub papíru a společně se vložily do tiskárny.

Kvalita tisku se odvíjí počtem tiskových jehliček. Obecně platí, že čím více jehliček, tím lze dosáhnout lepší kvality. Dalšími faktory, které kvalitu ovlivňují jsou:

1. Rychlost pojezdu tiskové hlavy
2. Kvalita pásky
3. Synchronizace pohybu pásky

Obrázek 1.6: Ukázka jehličkové tiskárny

[11]

4. Posun papíru
5. Ostatní

Výhody

- Nízké náklady
- Jednoduchá obsluha a údržba
- Rychlý tisk
- Vysoká trvanlivost tisku
- Vysoká odolnost

Nevýhody

- Nižší kvalita tisku
- Hlučný provoz
- Snižování kvality tisku opotřebováváním mechaniských částí

1.4.4 Termotiskárna

Termotiskárny v moderní době nahrazují tiskárny jehličkové. Jejich princip spočívá podle [12] v zahřívání řady termoelementů (bodů), které přichází do kontaktu se speciálním papírem, jenž při zahřátí ztmavne. Tím se na něj tisknou jednotlivé body. Velikostí procházejícího proudu a rychlostí tisku lze dosáhnout podstatného množství odstínů šedi. Jsou velice vhodné pro tisk lístků či paragonů.

Termotiskárny dělíme podle principu tisku na dva druhy. Prvním typem jsou tiskárny s řádkovou hlavou. Hlavu tvoří pás tepelných prvků v šířce celého speciálního papíru a tisknou vždy celý řádek. Tisk je velice rychlý, avšak tisknout lze pouze na speciální papír. Druhým typem jsou tiskárny maticovou hlavou. Stejně, jako u předchozího typu je i zde hlava tvořena pásem termoelementů, avšak zápis se provádí přes termocitlivou pásku, která obarví potiskované médium. Tento typ zvládne tisk na kancelářský papír.

Výhody

- Kompaktnost
- Jednoduchá obsluha
- Rychlý tisk
- Spotřební materiál je pouze papír

Nevýhody

- Nižší kvalita tisku
- Potřeba speciálního papíru
- Blednutí papíru
- Pořizovací cena

Obrázek 1.7: Ukázka tepelné tiskárny

[13]

1.4.5 GPS

Projekt GPS podle zdroje [14] vznikal v 70. letech 20. století v USA, kde měl sloužit pro vojenské účely Ministerstva obrany. Cílem projektu bylo sledování aktuální polohy objektu na jakémkoliv místě planety pomocí jednoduchého přijímače.

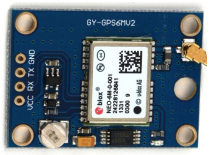
V 90. letech 20. století byl projekt uvolněn pro širokou veřejnost, jak uvádí [14], avšak obsahoval omezení jako prevence před zneužití teroristy. Omezení se týkalo zkreslování signálu, kde zařízení ukazovalo polohu s odchylkou až 30 metrů. Dalším opatřením bylo omezení dostupnosti, kde signál GPS nebyl dostupný po celé zeměkouli. 1. května 2000 se ale tato omezení zrušila a civilisté mají od té doby k dispozici stejné možnosti jako vojenský sektor.

Systém GPS se skládá ze 3 částí:

- **Kosmická část** v současnosti podle [15] disponuje 31 funkčními satelity a 3-5 neaktivními satelity, které v případě potřeby lze uvést do provozu. Nacházejí se 20 000 km nad zemským povrchem a oběhnou Zemi za 11 h a 56 min na šesti oběžných drahách skloněných o 60 stupňů. Z jakéhokoliv místa na zemi by tak v ideálním případě mělo být vidět 12 družic. Každá družice obsahuje přijímač, vysílač, cesiové atomové hodiny a další zařízení, která nejsou spojena s detekováním polohy.
- **Řídící systém** tvoří 9 pozemních stanic podél rovníku, které mají za úkol monitorovat běh družic a řešit případné problémy.
- **Uživatelská část** obsahuje veřejně dostupná zařízení, která obsahují GPS přijímač a lze je zakoupit v obchodě.

Každá družice vysílá informace o své poloze, přesný čas z atomových hodin a přibližné polohy ostatních družic. Přijímač využívá pro výpočet polohy časový rozdíl mezi okamžikem vyslání a okamžikem přijmutí dat. Při přijetí dat ze 3 různých družic lze poté určit zeměpisnou délku a šířku. Při přijetí dat ze 4 různých družic lze navíc zjistit i nadmořskou výšku.

K výpočtu je nutný přesný čas i v přijímači, který ale již netvoří atomové hodiny. Při načítání informací o družicích se čas upraví. Odchylka jedné tisíciny vteřiny může mít za následek chybu ve výpočtu v řádech stovek kilometrů.



Obrázek 1.8: Ukázka GPS modulu

[16]

1.4.6 Přehrávač MP3

MP3 je formát pro zvukové soubory, které prošly algoritmem ztrátové komprese (původní kvalitu tedy nelze navrátit). Kompresní algoritmus je založený na vlastnostech lidského sluchu a to tak, že z původní zvukové nahrávky odstraní nadbytečné zvukové signály, které buď lidské ucho neslyší nebo si je neuvědomuje. Velikost komprese závisí na volbě datového průtoku (počet zpracovaných bitů za jednotku času). Velikost nekomprimovaného souboru lze zmenšit řádově 4-krát až 10-krát, přičemž hlavní zvuková složka zůstává pro lidské ucho zřetelná. Datové průtoky se nejčastěji pohybují mezi 128–320 kbit/s, ale lze se setkat i s nižší, či vyšší hodnotou. Při příliš nízkém datovém toku lze zaznamenat vysokou ztrátu původní kvality v podobě šumu. Naopak vyšší hodnota datového průtoku přibližuje zvukový signál původní kvalitě na úkor velikosti zvukového souboru.

Přehrávač MP3 je označení pro zařízení schopné číst a přehrávat zvukové soubory ve formátu MP3. Doplňujícími funkcemi může být i podpora dalších zvukových formátů jako např. WAV, OGG, AAC nebo dnes moderních formátů FLAC a ALAC. Přehrávače se rozdělují podle typu média, na kterém jsou zvukové soubory uloženy:

- **s flash pamětí**: Tento typ přehrávačů obsahuje interní flash paměť. Pro nahrávání souborů do paměti obvykle využívají USB konektor. Jelikož čtení souborů nevyžaduje žádnou pohyblivou mechaniku, jsou tyto přístroje odolnější vůči poškození. Tento typ obvykle dosahuje nejlepší energetické spotřeby.

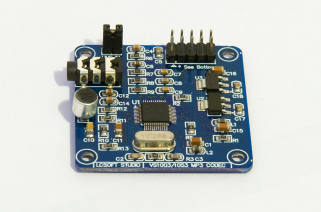
- **s pevným diskem**: Tento typ přehrávačů obsahuje pevný disk. Výhodou těchto přehrávačů je velká paměťová kapacita, která může zahrnovat až desítky tisíc skladeb. Pro svou velkou kapacitu je tento typ používaný převážně jako jukebox. Nevýhodou pevného disku je velká náchylnost vůči pádům a nárazům.

- **Diskové přehrávače**: Tento typ přehrávačů obsahuje mechaniku pro čtení přenosných disků. V dnešní době se využívají média CD, DVD a Blu-Ray. Oproti předchozím typům je zřetelný cenový rozdíl. Tento typ je výrazně levnější. Nevýhodou přehrávače jsou pohyblivé součásti v mechanice, které velmi snadno způsobí chybu ve čtení nebo přehrávání.

- **Síťové přehrávače**: Tento typ přehrávačů obsahuje síťovou kartu, přes kterou je připojeno na digitální médium (drátově či bezdrátově).

- **se čtečkou paměťových karet**: Tento typ přehrávačů obsahuje slot pro vložení paměťové karty. Typ i velikost paměťové karty se může lišit.

Hlavní součástí přehrávače je zvuková karta. K té je připojeno vstupní médium. Karta poté načte soubory ze zvukového média a dekóduje je. Dekó-



Obrázek 1.9: Ukázka mp3 modulu

[17]

dovaný výstup se odesílá na digitálně-analogový převodník, na který je napojen konektor pro výstupní zvukové zařízení. Nejčastěji používáný konektor pro MP3 přehrávač je stereo jack o průměru 3,5 mm.

1.4.7 GSM

GSM je označení standartu digitálních sítí pro mobilní telefony. První provoz byl zahájen v roce 1991 a do dvou let se počet GSM sítí rozrostl na 36 ve 22 různých státech. Evropský standart se poté dále rozšířil i do dalších zemí a v současnosti ho využívá na 80 zemí světa.

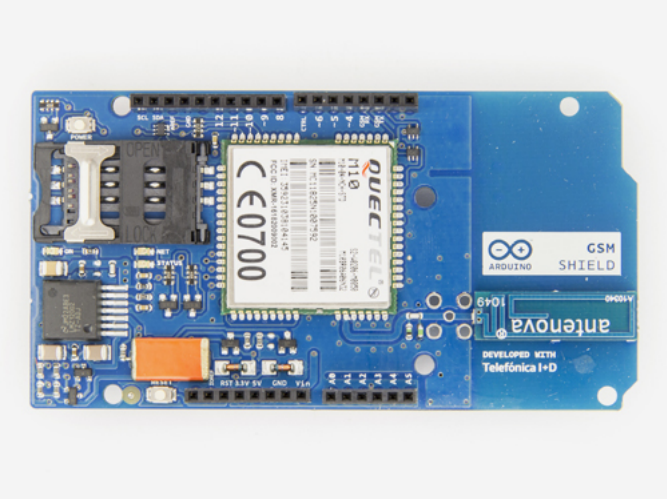
Podle [18] se systém GSM sestává jako celek ze tří částí, z nichž každá má své specifické subsystémy. První částí jsou externí telekomunikační sítě, mezi něž patří veřejné komutované sítě PSTN - tj. JTS, čili normální telefon, dále pak různé ISDN a další sítě. Systém GSM, jak již bylo řečeno, je systém otevřený, proto pod sebe může integrovat jakýkoliv z těchto systémů.

Druhou část [18] uvádí jako operátory. Operátoři jsou společnosti, které spravují systém GSM zejména po ekonomické stránce (účtují služby systému, starají se o distribuci mobilních stanic, výdávají SIM karty). Dalším pojmem je service provider, tj. firma poskytující služby operátora ve vlastní režii. Není pravidlem, že operátor sítě musí být nutně totožný se service providerem. Service providerů je v dnešní době na území České republiky mnoho a fungují tak, že si pronajmou část sítě většího operátora. S operátorem uzavřou smlouvu, která upraví způsob úhrady, a service provider tak může poskytovat služby za nižší ceny.

Poslední část, která se zabývá technickým vybavením se nazývá celulární systém. Tato část se dělí na 3 podčásti, které jsou:

- **subsystém základnových stanic BSS** s nímž komunikují mobilní stanice prostřednictvím rádiového rozhraní.
- **sítový a spínací subsystém NSS** je systém, který má podobně jako telefonní ústředny za úkol vyhledávat a přepojovat telefonní hovory
- **operační subsystém OSS** zabezpečuje provoz a údržbu obou výše uvedených subsystémů (tj. NSS i BSS), koordinuje funkci celého sys-

tému a vyřizuje záležitosti finančního charakteru (tarifikace účastníků, evidence plateb)



Obrázek 1.10: Ukázka GSM modulu

[19]

Kapitola 2

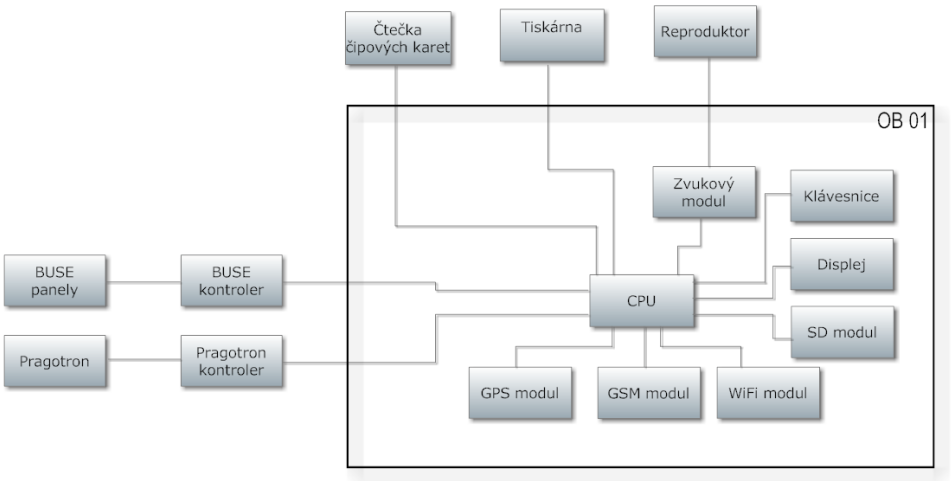
Návrh

Cílem „univerzitního“ dopravního podniku Krkonošské metro, není pouze zajistit přepravu cestujících v našich nejvyšších horách, ale zároveň tuto přepravu zajišťovat co nejkvalitněji a nejlevněji za použití dopravní telematiky a dalších inteligentních („smart“) telematických systémů. Mezi tyto navrhované systémy patří i dispečersko-palubní informační systém KISyD KrM. Hlavní funkcí hardwarové jednotky je komunikace vozidla s provozními dispečinky a informování cestující veřejnosti o dopravních údajích, které se bezprostředně váží k dopravnímu prostředku. Součástí tohoto modulu KrM OB 01, zahrnutou do OIS je ovládání panelů zobrazujících číslo linky, konečnou a nácestné zastávky na čele, bocích a konci každého vozidla (ať už se jedná o maticové nebo diodové zobrazovače Bustec, BUSE či Em-test, nebo v případě KrM o informační zařízení Pragotron), dále akustické hlášení zastávek a dalších provozních informací, včetně informací osobám se sníženou schopností pohybu a orientace (slepé občany, apod.), ovládání označovačů jízdenek v daném vozidle, tisknoucích přesný čas a datum, tarifní zónu, zastávku, evidenční číslo vozidla či další tarifně důležité informace a ovládání kasy pro doplňkový výdej jízdenek u řidiče či strojvedoucího. Další součástí mimo OIS je pak vlastní komunikace vozidla prostřednictvím této palubní jednotky s jednotlivými dispečinky a složkami. Hlavní provozní dispečink z jednotky dostává údaje o její aktuální on-line poloze a porovnává aktuální polohu každého vozidla s jeho plánovanou polohou dle jízdního řádu. Naopak díky tomuto podsystému dostává palubní jednotka informace o odchylkách od standardní situace a díky nařízenému čekání tak umožňuje zachování přípojů, aby nedošlo cestujícím k ujetí spoje při přestupu. Dopravní dispečink pak z jednotky obdrží údaje o řidiči, délce jeho směny ve službě, kde se každý hledaný řidič nachází a další údaje, potřebné pro mzdové oddělení a výpočet mzdové odměny za odvedenou práci. Vozový dispečink pak obdrží informaci o najetých kilometrech každého vozidla (nutné pro stanovení servisních prohlídek a později generálek), tedy jakousi knihu jízd, včetně knihy závad. Pokud se za jízdy na vozidle projeví malá závada, která ovšem umožňuje další provoz vozidla, řidič prostřednic-

tvím palubní jednotky tuto závadu „nahlásí“, a vozovna získá údaje nutné k opravě spolu s časem, kdy se daný vůz navrátí do vozovny/garáží a v daný čas tak může být k dispozici již potřebný náhradní díl, či servisní pracovníci, kteří drobnou závadu odstraní, aby mohlo vozidlo druhý den opět v naprostém pořádku vyrazit na další provozní výkon. Součástí vozového dispečinku pak bude i celková databáze veškerých servisních úkonů, které budou na každém konkrétním vozidle prováděny. Informace o poloze palubní jednotka získává ze systému satelitní navigace Galileo (GPS či Glonas), údaj o přesném čase pomocí signálu DCF77 z Frankfurtu a vlastní datová komunikace mezi vozidlem a dispečinkem je pak prováděna prostřednictvím mobilního signálu GSM, vlastní radiové sítě (na dráze je používán TRS - traťový radiový systém) a v rámci aktualizace údajů ve vozidlech během pobytu ve vozovně pak signál Wi-Fi nebo Bluetooth.

2.1 Hardwarové požadavky

Vizi hardwarové části projektu, která bude nasazena v dopravním prostředku zobrazuje obrázek 2.1.



Obrázek 2.1: Blokové schéma celého zařízení

Fyzické nároky či výběr jednotlivých periférie jsou následující:

1. **Čtečka čipových karet**: RFID kompatibilita
2. **Tiskárna**: Z důvodů jedoduchosti a snadné dostupnosti bude použita malá tepelná tiskárna s teplocitlivým papírem
3. **Zvukový modul**: Kompatibilní s formáty MP3 a WAV, čtení z SD karty

4. **Klávesnice**: Číselná klávesnice s malým počtem tlačítek navíc např. klávesnice 4x4
5. **Displej**: Malý displej pro zobrazení informací a nastavování hodnot. Dostatečný je např. LCD displej s počtem znaků 16x2 podporující standart HD44780
6. **SD modul**: Může být libovolný.
7. **WiFi modul**: Kompatibilita se standarty 802.11b a 802.11g.
8. **GSM modul**: Používání standartů GSM 07.07 & 07.05.
9. **GPS modul**: Může být libovolný.
10. **Pragotron + kontroler**: Kontroler kompatibilní se zvolenou deskou.
11. **BUSE panely + kontroler**: Kontroler kompatibilní se zvolenou deskou.

Výběr řídící desky je závislý na nárocích pro řízení vybraných periférií, které jsou následující:

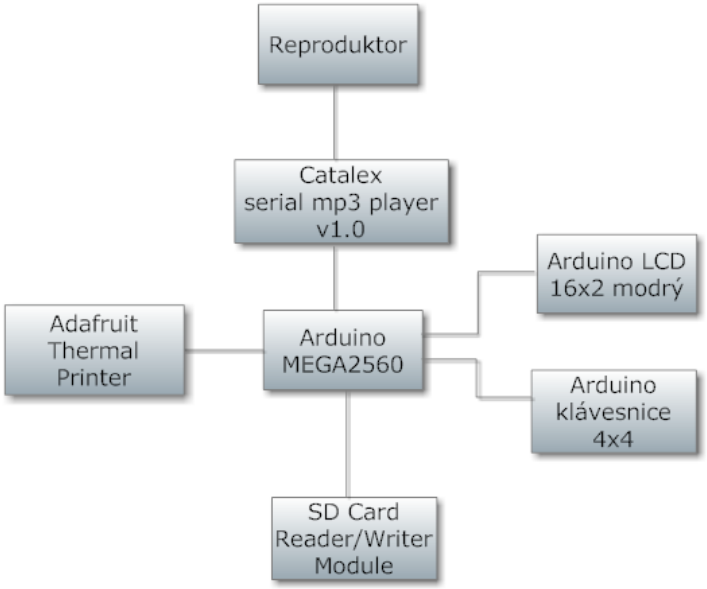
1. **Čtečka čipových karet**: SPI komunikační sběrnice (4 piny,1 pin pro resetovací puls, 1 digitální pin pro přerušení)
2. **Tiskárna**: Sériová linka (2 piny)
3. **Zvukový modul**: Sériová linka (2 piny)
4. **Klávesnice**: 8 digitálních pinů
5. **Displej**: 6 digitálních pinů, 10kΩ potenciometr
6. **SD modul** SPI komunikační sběrnice (4 piny)
7. **WiFi modul**: Sériová linka (2 piny), 1 pin pro resetovací puls, 1 digitální pin
8. **GSM modul**: Sériová linka (2 piny), 1 pin pro resetovací puls
9. **GPS modul**: Sériová linka (2 piny)
10. **Pragotron + kontroler**: Sériová linka (2 piny)
11. **BUSE panely + kontroler**: Sériová linka (2 piny)

Kromě dostatečně výkonného mikrokontroleru řídící desky jsou tedy výsledné požadavky na řídící desku:

- 7 sériových linek (14 pinů)

- SPI sběrnice
- 19 digitálních pinů

Pro tento projekt díky vyhovujícím parametrům i možnosti budoucího rozšíření byla vybrána deska **Arduino MEGA2560**. Pro snadnou dostupnost a jednoduché řízení byla vybrána tepelná tiskárna **Adafruit Thermal Printer CSN-A2-T**. Displej, klávesnice a SD modul byly vybrány jednoduché, podle nároků uvedených v odstavci pro stanovení požadavků. Realizace takto kompletního systému svou náročností přesahuje rozsah bakalářské práce, a proto se tato práce omezí jen na rozsah uvedený na obrázku 2.2.



Obrázek 2.2: Blokové schema zařízení pro tuto práci

2.2 Arduino MEGA2560

Arduino MEGA2560 je vývojová deska od firmy Arduino, která je založená na architektuře mikrokontroleru ATmega2560. Deska je zkonstruovaná za účelem složitějších projektů. Obsahuje celkem 54 digitálních I/O pinů (z toho 15 lze použít pro výstup PWM), 15 analogových pinů, 4x UART (hardwarové sériové porty), 16MHz krystalový oscilátor, USB připojení, napájecí konektor, ICSP hlavici a resetovací tlačítko. Deska je napájena buď pomocí USB kabelu nebo externího AC/DC adaptéru či baterie. Výrobce uvádí, že MEGA2560 je kompatibilní s většinou „Arduino shieldů“, které byly navrženy pro typy desek UNO a starší typy Duemilanove a Diecimila. MEGA2560 je přímým nástupcem typu Arduino MEGA.

Tabulka 2.1: Technické parametry Arduino MEGA2560

Mikrokontroler	ATmega2560
Operační napětí	5 V
Doporučené vstupní napětí	7 – 12 V
Mezní vstupní napětí	6 – 20 V
Digitální I/O piny	54
Vstupní analogové piny	16
Proud I/O pinu	20 mA
Proud na 3,3V pinu	50 mA
Flash paměť	256 kB (8kB vyhrazeno pro bootloader)
SRAM	8 kB
EEPROM	4 kB
Taktovací frekvence	16 MHz
Délka	101,52 mm
Šířka	53,3 mm
Váha	37 g

Programování desky probíhá v programu Arduino IDE, které poskytuje výrobce na svých oficiálních stránkách. Mikrokontroler desky obsahuje výrobcem předprogramovaný bootloader, což je program, který umožňuje nahrávání nového kódu bez nutnosti využití externího hardwarového programátoru. Komunikace probíhá protokolem STK500. Použitím hardwarového programátoru neboli ISP lze obejít bootloader v mikrokontroleru a vkládat nový kód.

Arduino MEGA2560 má v sobě zabudovanou pojistku, která má za úkol ochránit USB porty před zkratem či přepětím. Přestože většina počítačů tuto ochranu poskytuje, MEGA2560 poskytuje extra ochrannou vrstvu. V momentě, kdy hodnota proudu na portech USB přesáhne 500 mA, pojistka připojení přeruší do doby, dokud se zkrat nebo přepětí neopraví.

Jak již bylo zmíněno, desku lze napájet buď pomocí USB kabelu nebo pomocí externího zdroje. Při připojení více zdrojů energie deska automaticky vybere vhodnější zdroj. Externím zdrojem může být buď AC–DC adaptér nebo dostatečně výkonná baterie. Pro připojení adaptéru deska disponuje napájecí koncovkou s průměrem 2,1 mm. Pro připojení baterie disponuje deska vstupními piny Vin a GND. Provozní napětí desky se pohybuje mezi 6–20 V. Výrobce nicméně varuje před používáním desky při napětí menším než 7 V, protože napájecí pin pro 5 V nemusí dosahovat plné hodnoty napětí, což může způsobit destabilizaci desky. Při používání více než 12 V se může regulátor napětí začít přehřívat a způsobit poškození desky. Z těchto důvodů je doporučené napětí 7–12 V.

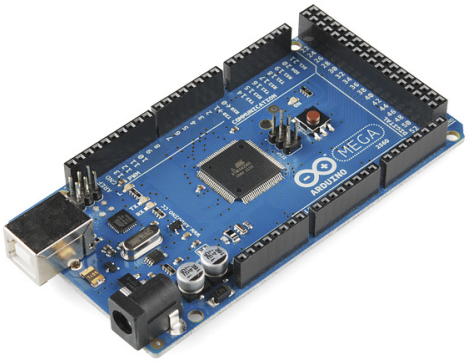
Deska dále poskytuje 256 kB flash paměti pro nahrání programu, z čehož 8 kB je vyhrazeno pro bootloader. Operační paměť SRAM je uváděna jako 8 kB. V poslední řadě má deska vlastní paměť EEPROM o velikosti 4 kB. Pro

Tabulka 2.2: Napájecí piny Arduino MEGA2560

PIN	Funkce
Vin	Vstup pro napájení z baterie
5V	Regulovaný výstup na 5 V pro napájení periférií
3.3V	Regulovaný výstup na 3,3 V pro napájení periférií
GND	Uzemňovací piny
IOREF	Neregulovaný výstup napájecího napětí
AREF	Regulace operačního napětí pro analogové piny

práci s pamětí EEPROM dodává výrobce externí knihovnu.

Každý z 54 digitálních pinů lze použít pro vstup nebo výstup. Operační napětí na pinech je 5 V. Každný z pinů přijímá/poskytuje operační napětí 20 mA, které reguluje proměnlivý rezistor v rozsahu 20–50 kΩ. Maximální přípustná mez proudu je 40 mA. Vyšší hodnota proudu by měla za následek trvalé poškození desky.



Obrázek 2.3: Deska Arduino MEGA2560

[20]

MEGA2560 dále poskytuje 16 analogových vstupních pinů. Každý z pinů pracuje v rozsahu 10 bitů (čili 1024 hodnot). Standartní měření probíhá v rozsahu 0–5 V. Horní hranici měření lze změnit pomocí pinu AREF.

Dalším způsobem přenosu dat, který deska poskytuje, je komunikace s PC za pomocí USB. USB komunikace probíhá přes virtuální COM port. Přenášené hodnoty lze v PC sledovat pomocí sériového monitoru. Arduino IDE jednoduchý monitor poskytuje, avšak existují i jiné specializované sledovací nástroje jako např. program Putty. Deska dokáže pouze signalizovat odchozí a příchozí přenos dat pomocí vestavěných LED.

Tabulka 2.3: Speciální funkce digitálních pinů

Funkce	**Piny**
Sériová linka	0 (RX),1 (TX)
Sériová linka 1	19 (RX),18 (TX)
Sériová linka 2	17 (RX),16 (TX)
Sériová linka 3	15 (RX),14 (TX)
Externí přerušení 0	2
Externí přerušení 1	3
Externí přerušení 5	18
Externí přerušení 4	19
Externí přerušení 3	20
Externí přerušení 2	21
PWM	2–13, 44–46
SPI	50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS)
LED	13
TWI	20 (SDA), 21 (SCL)

Přestože deska fyzicky disponuje čtyřmi hardwarovými sériovými linkami, lze sériovou komunikaci provádět i na ostatních digitálních pinech. Tuto speciální vlastnost umožňuje výrobcem dodávaná knihovna SoftwareSerial, která softwarovým způsobem přenáší vlastnosti sériové komunikace na digitální piny.

Pro usnadněné nahrávání kódu a práce s deskou je MEGA2560 vybavena funkcí automatického restartu. Tato funkce se spouští při každém připojení USB a některé programy tuto funkci využívají např. pro nahrání nového kódu. Tato funkce se nicméně v několika případech neukázala jako velmi stabilní a k samovolným resetům může docházet i za běhu programu. Tyto pulzy lze zablokovat vložením 10μF kondenzátoru mezi resetovací pin a GND.

2.3 Adafruit Thermal Printer CSN-A2-T

Jedná se o malou tepelnou tiskárnu od firmy Adafruit. Tento typ používá pro tisk textu a bitmap tepelnou hlavu. Výhodou tepelné hlavy je její stálá poloha, čili nedochází k pojíždění tiskové hlavice jako u jiných tiskáren. Díky této hlavici dosahuje tiskárna malých rozměrů, což ji činí vhodnou pro použití v mobilních zařízeních.

Pro vytvoření dostatečného tepla na tepelné hlavě je potřeba dostatečné množství energie. Operační napětí tiskárny se pohybuje v rozmezí 5–9 V. Pro dostatečné rozehřátí tepelné hlavy operuje tiskárna při proudu pohybujícím se okolo 1,5 A. Z toho důvodu tiskárnu nelze napájet z řídící desky ani USB portu, ale je nutno připojit vhodný napájecí adaptér.

Pro tisk se využívá teplocitlivý papír. Tiskárna Adafruit Thermal Printer CSN-A2-T využívá role o šířce 57,5 mm. Optimální délku role uvádí výrobce jako 15 metrů.

Přenos dat mezi tiskárnou a mikrokontrolerem probíhá přes 5 V TTL sériovou linku. Zelený vodič tiskárny slouží pro přenos signálu TX a žlutý pro RX. Černý vodič slouží pro uzemnění. Výrobce pro usnadněné ovládání tiskárny dodává knihovnu napsanou v jazyce C, a dále také nástroje pro usnadnění tisku bitmap.

Tabulka 2.4: Technické parametry tiskárny Adafruit Thermal Printer CSN-A2-T

Typ papíru	Teplocitlivý
Šířka papíru	57,5 mm
Délka papíru	<= 15,24 m (50 ft)
Šířka role	39 mm
Operační napětí	5–9 V
Operační proud	1,5 A
Rychlost tisku	50–80 mm/s
Rozlišení tisku	8 bodů/mm, 384 bodů/řádek
Optimální šířka tisku	48 mm
Znakové sady	ASCII, GB2312-80 (Čínština)
Protokol	TTL Serial
Přenosová rychlost	19 200 baudů
Životnost	5 000 000 řádků
Šířka	111 mm
Délka	65 mm
Výška	57 mm
Operační teplota	5 – 50 °C
Operační vlhkost	10 – 80 %
Mezní teplota	-20 – 60 °C
Mezní vlhkost	10 – 90 %

Obrázek 2.4: Adafruit Thermal Printer CSN-A2-T

[13]

2.4 Arduino LCD display 16x2

LCD displeje jsou moderní technickou variantou konstrukčně jednoduších segmentových displejů či maticových displejů. Využití mají v obrovské škále různých zařízení. Od malých displejů v tiskárnách po informační panely v hromadné dopravě. Rozdělit se dají podle barevnosti na barevné a jednobarevné (monochromatické) a podle způsobu zobrazování na znakové a grafické.

Tabulka 2.5: Technické parametry Arduino LCD display 16x2

Typ displeje	znakový LCD
Barva znaků	bílá
Barva podsvícení	modrá
Počet řádků	2
Počet znaků na řádek	16
Operační napětí	5 V
Průmyslový standard	HD44780
Šířka	80 mm
Délka	35 mm
Výška	11 mm
Šířka informační plochy	64,5 mm
Výška informační plochy	16 mm

Pro účely ovládání OB 01 je dostatečný znakový displej o rozlišení 16x2 znaky. Výhodou znakových displejů je velmi jednoduché ovládání. Obsahují totiž „slovník“ znaků, které jsou předem výrobcem definované. Pro zobrazení znaku se pouze nastaví pozice kurzoru a odešle informace o znaku.

Znakové displeje se vyrábí převážně jednobarevné. I přes jednobarevnost lze pořídit displeje s různými barvami podsvícení. Pro účely této práce bude použit displej s modrým podsvícením. Pro zlepšení viditelnosti textu displej umožňuje nastavení kontrastu znaků. K nastavení kontrastu je třeba k displeji připojit 10kΩ potenciometr. Bez připojeného odporu displej nedokáže správně zobrazit žádný znak.

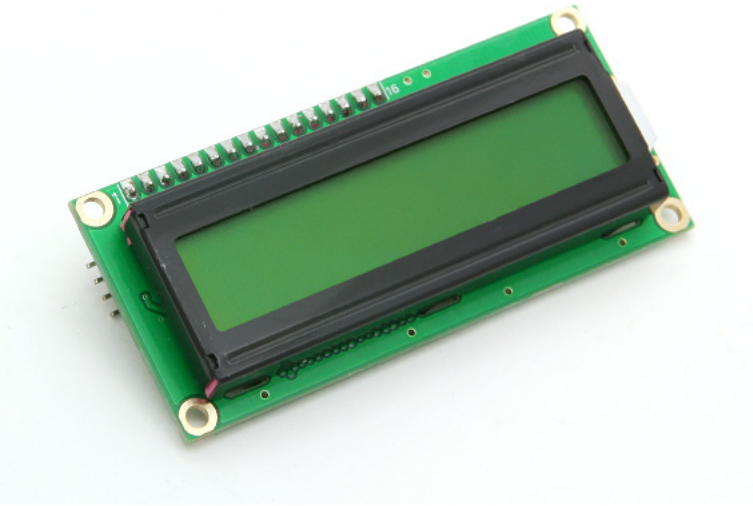
Zdroj [21] uvádí: „Pro práci se znakovými LCD displeji je Arduino vybaveno knihovnou, která je zahrnuta již v základním balíku IDE. Ta umožňuje ovládání všech znakových displejů, které jsou kompatibilní s řadičem Hitachi HD44780, což většina současných displejů je.“

Displej se připojuje pomocí 16ti pinů. Piny jsou popsány v tabulce 2.6.

Ke snadnému ovládání poskytuje Arduino knihovnu pro práci s displeji se standardem HD44780.

Tabulka 2.6: Piny displeje Arduino LCD display 16x2

Pin	Funkce
VSS	Uzemnění displeje
VCC	Napájení displeje
V0	Kontrast displeje
RS	Výběr registru (register select)
R/W	Čtení/zápis
E	Povolení zápisu
DB0 – DB7	Datové piny
A	Napájení podsvícení
K	Uzemnění podsvícení



Obrázek 2.5: Arduino LCD display 16x2

[22]

2.5 Arduino klávesnice 4x4

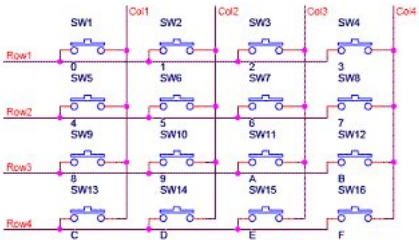
Nejvhodnější variantou pro ovládání přístroje s potřebou zadávání hodnot je klávesnice. V této práci bude použita membránová klávesnice s 16ti tlačítky s rozdělením 4x4.

Klávesnice se připojuje pomocí 8 datových pinů. Pro zapojení tlačítek je použito tzv. pole. To znamená, že každé tlačítko je jedním z vývodů zapojeno k vodiči, který slouží k označení konkrétního řádku a druhým vývodem k vodiči, který označuje sloupec.

Ke zjištění stisku tlačítka používá řídící deska jednoduchý princip. Na řádky postupně odesílá hodnotu logické 1 a testuje, zda-li se hodnota logické

Tabulka 2.7: Technické parametry Arduino klávesnice 4x4

Počet tlačítek	16
Matice	4x4
Kontaktní odpor	500 Ω
Izolační odpor	100 MΩ
Odezva	1 ms
Životnost	>100 000 000 stisků
Maximální provozní teplota	60 °C



Obrázek 2.6: Schema vnitřního zapojení klávesnice

1 objevila i na některém sloupci. Pokud se logická 1 ve sloupci objeví, tlačítko je považováno za stisknuté.

Touto funkcí ale sama deska nedisponuje. Tento princip skenování kláves implementuje přiložená knihovna keypad.h, která krom reakcí na tlačítka umožňuje implementovat i jejich reprezentaci. To znamená, že fyzická klávesa A bude v programu reprezentována např. jako znak O a i jakákoliv další reakce bude namísto A pojmenována O a zároveň návratová hodnota stisku, která je ASCII kódem znaku, který program reprezentuje, bude nastavena jako kód znaku O.

Obrázek 2.7: Klávesnice 4x4

[23]

2.6 SD Reader/Writer Module

Pro uchovávání dat, která budou přenosná a použitelná ve vícero zařízeních, je třeba vybrat vhodné paměťové médium. Pro stavbu OB 01 není třeba velké paměti, a proto se jako médium zvolila paměťová karta. Modul podporuje karty SD o standartní velikosti. Maximální kapacita média je u vybraného modulu stanovena na 64 MB.

Tabulka 2.8: Technické parametry čtečky SD Reader/Writel Module

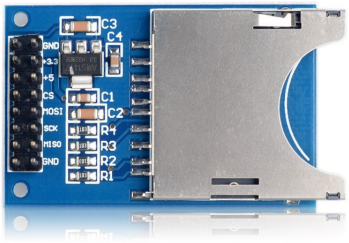
Typ karet	SD
Operační napětí	3,3 V (obsahuje stabilizátor)
Napájecí napětí	3,3 nebo 5 V
Čip	AMS1117
Šířka	51 mm
Délka	31 mm
Váha	15 g

Operační napětí modulu výrobce uvádí jako 3,3 V. Nicméně napětí na pinech moderních vývojových desek je 5 V, a proto modul obsahuje napájecí piny jak pro 3,3 V, tak pro 5 V. K napájecím pinům je připojený stabilizátor, který snižuje a udržuje napětí na operačních 3,3 V.

Komunikace s řídící deskou probíhá přes sběrnici SPI. Připojuje se pomocí 8 pinů, z čehož 2 jsou pro napájení, 2 pro uzemnění a 4 pro sběrnici SPI. Pro usnadnění ovládání poskytuje Arduino knihovnu pro SD moduly komunikující přes sběrnici SPI.

Při předchozím používání SD karty v jiných zařízeních existuje možnost, že SD karta nebude naformátována v kompatibilním souborovém systému. Knihovna pro ovládání modulu je kompatibilní pouze se souborovými systémy FAT16 a FAT32. Je proto třeba mít SD kartu naformátovanou na jeden z těchto souborových systémů, ale Arduino spíše doporučuje využívat FAT16. Kartu není vhodné formátovat příliš často, protože časté formátování výrazně ovlivňuje životnost SD karty.

Pro názvy souborů na kartě se využívá tzv. konvence 8.3. Konvence 8.3 je způsob pojmenování souborů používaných v operačním systému DOS a verzích systému Microsoft Windows 3.1 a starších. Novější verze Microsoft Windows používají konvenci 8.3 pouze v případě příliš dlouhého názvu souboru. Konvence stanovuje, že název souboru nesmí překročit délku 8 znaků a délka přípony musí obsahovat přesně 3 znaky. Název souboru a přípony je oddělen znakem „.“.



Obrázek 2.8: SD Reader/Writer Module

[24]

2.7 Catalex Serial MP3 player v1.0

Jedná se o jednoduchý MP3 přehrávač založený na MP3 čipu YX5300. Zařízení podporuje přehrávání zvukových souborů ve formátech WAV a MP3 v rozsahu vzorkovacích frekvencí 8 kHz – 48 kHz. Jako datové médium pro uchování zvukových souborů využívá zařízení paměťové karty typu mikro SD a mikro SDHC. Zdířka pro připojení karty se nachází na zadní straně zařízení. Zařízení podporuje karty o maximální kapacitě 32 GB naformátované na souborový systém FAT16 nebo FAT32.

Tabulka 2.9: Technické parametry Catalex Serial MP3 player v1.0

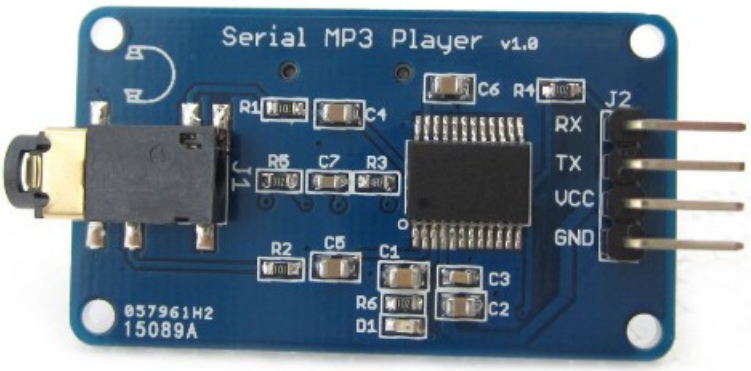
Vzorkovací frekvence (kHz)	8 / 11,025 / 12 / 16 / 22,05 / 24 / 32 / 44,1 / 48
Podporované zvukové formáty	MP3, WAV
Podporovaná datová média	micro SD, micro SDHC
Maximální podporovaná kapacita	32 GB
Podporované souborové systémy	FAT16, FAT32
Počet stupňů hlasitosti	30
Přenosová rychlost	9600 baudů
Mezní napětí	3,2 - 5,2 V
Maximální proud	200 mA
Rozhraní	3,3 V / 5V TTL
Šířka	49 mm
Délka	24 mm
Výška	8,5 mm
Váha	5 g

Rozsah napájení výrobce uvádí jako 3,2 – 5,2 V, přičemž optimální napájení je 5 V. Krom napájecích pinů obsahuje zařízení také piny pro sériovou komunikaci (RX, TX). Zařízení komunikuje s ovládací deskou rychlostí 9600

baudů / s. Zařízení je ovládáno pomocí 8B příkazů, které slouží např. pro nastavení hlasitosti nebo spuštění přehrávání. Celkový počet implementovaných příkazů v zařízení je podle výrobce 20.

Některé příkazy vyžadují přesné pojmenování souborů a složek. Při přehrávání ze složky musí být složka pojmenována číslem složeným ze 2 číslic např. 01, 02, 03 atd. Soubory ve složkách musí mít název složený z čísla složeného ze 3 číslic a doplněné znaky „xxx“ např. 001xxx.mp3.

Výrobce uvádí, že zařízení je kompatibilní s deskami typu Arduino, ARM, AVR a PIC. Jako další možnost využívání uvádí možnost připojení k PC přes rozhraní USB pomocí převodníku TTL / USB a následné využití sériového monitoru v PC pro odesílání příkazů.



Obrázek 2.9: Catalex Serial MP3 module v1.0

Kapitola 3

Realizace

3.1 Hardwarové zapojení

I přes velký počet datových pinů má Arduino MEGA2560 nevýhodu v nízkém počtu napájecích pinů. Zároveň její relativně malé rozměry zesložiťují „rozumně vypadající“ připojení většího množství periférií. Pro zmírnění těchto nedostatků jsem se rozhodl k zapojení součástek pomocí nepájivého kontaktního pole neboli breadboardu. Breadboard je pomůcka zapojování prototypů elektronických obvodů bez nutnosti pájení. Je vyrobena z plastu, který obsahuje otvory pinů. Piny jsou připojeny k vodivým cestám, obvykle ve formě řad pětic pinů propojených ve svislém směru. Horní i spodní okraj desky obvykle obsahuje 2 vodorovné vodivé plochy po celé šířce pole. Tyto vodivé plochy slouží pro společné napájení a uzemnění součástek.

První součátkou, kterou je nutno zapojit, je samotná řídící deska Arduino MEGA2560. Výběr napájení závisí na aktuálním použití. Pro vývojové účely, práci s deskou a nahrávání programu je vhodným napájecím médiem USB kabel s koncovkou typu samec B, připojený do USB portu desky, a druhou koncovkou typu samec A, připojený do USB portu v PC. Pro nasazení v praxi je neustálé napájení pomocí USB portu nepraktické a mnohem vhodnějším řešením je použití napájecího adaptéru. Adaptér by měl být schopný dodávat desce napětí v rozmezí 9–12 V a proud v rozmezí 0,5 – 1 A. Při nižších hodnotách hrozí nepředvídatelné chování desky z důvodu nedostatku energie. Při vyšší hodnotě napětí hrozí poškození desky. Napětí z 5V pinu desky je třeba vyvést na jeden z pinů vodorovné vodivé plochy nepájivého kontaktního pole, které je označené červeným pruhem. Zemnící pin označený GND je třeba připojit na jeden z pinů vodorovné plochy breadboardu, která je označena modrým pruhem. V momentě nahrání programu na desku je třeba mezi pin RESET a GND připojit 10mF kondenzátor, který má za funkci tlumení resetovacích pulzů. Tyto pulsy způsobují náhlé a nepředvídatelné restarty desky. Pro přehlednější připojení jsem vyvedl pin RESET na jednu ze svislých vodivých cest breadboardu a jeden vývod kondenzátoru zapojil do této cesty.

Druhý vývod kondenzátoru jsem připojil do modré vodorovné cesty, která je použita jako uzemnění celého obvodu.

Další připojenou součástkou je mp3 přehrávač. Ten je nutno připojit k sériové lince a uzemnění. Je tedy třeba vyvést sériové piny RX a TX z řídící desky na některé z volných vodivých cest breadboardu. Pro mp3 přehrávač jsem vyvedl piny 14 a 15. K vodivým cestám je nutno samozřejmě připojit příslušný pin TX a RX z pinů přehrávače. Přehrávač pro svou funkci vyžaduje připojení mikro SD nebo mikro SDHC karty ve slotu, který se nachází na zadní straně přehrávače. Karta musí být naformátována na souborový systém FAT16 nebo FAT32. Indexace hudebních souborů probíhá podle pořadí, ve kterém byly nahrány. Deska obsahuje výstupní port jack o průměru 3,5 mm, přes který je pro reprodukci zvuku potřeba připojit libovolné výstupní zvukové zařízení.

Pro připojení klávesnice stačí pouze vyvést 8 volných datových pinů na různé svislé vodivé plochy breadboardu a k nim připojit odpovídající piny klávesnice. Ve své práci jsem vybral piny 22 – 29.

Další komponentou zařízení je displej. Piny displeje jsou vyrobeny pouze jako vodivé plošky s malou dírkou uprostřed, a proto je nejprve potřeba ke všem pinům připájet vodič. Po připájení vodičů k displeji lze začít připojivat jednotlivé piny. Piny se připojují následovně:

- pin VSS na uzemnění obvodu
- pin VCC k 5V zdroji napájení
- pin V0 na jezdec potenciometru (obvykle prostřední nožička)
- pin RS k datovému pinu pomocí breadboardu
- pin R/W na uzemnění obvodu
- pin E k datovému pinu pomocí breadboardu
- piny DB0 – DB3 zůstávají odpojeny
- piny DB4 – DB7 k datovým pinům pomocí breadboardu
- pin A k 5V zdroji napájení
- pin K na uzemnění obvodu

Jako datové piny jsem vybral piny 1, 2, 4, 5, 6 a 7.

Pro správnou funkčnost displeje je nutno připojit potenciometr o maximálním odporu 10 kΩ. Potenciometr má za úkol řídit kontrast textu displeje. Pokud kontrastní pin displeje není připojen, kontrast textu bude minimální a text se na displeji vůbec nezobrazí. Krajní vývody potenciometru se připojují k napájecí vodivé cestě a uzemnění. Prostřední vývod, neboli jezdec, se připojuje k pinu V0 displeje.

Předposlední komponenta zařízení je SD modul. Ta vyžaduje, podobně jako mp3 přehrávač, naformátovanou paměťovou kartu na souborový systém FAT16 nebo FAT32. Rozdílem je typ karty, který je zde SD. Maximální podporovaná kapacita je 64 MB. Napájecí pin pro 5 V je třeba připojit k napájecí vodivé cestě na breadboardu. Napájecí pin pro 3,2 V lze z důvodu jedinečnosti takového napájení připojit přímo k 3,2V pinu řídící desky. Všechny piny pro GND musí být připojeny na zemnící vodivé cestě breadboardu. SD modul je z celého zařízení jediná komponenta, která využívá sběrnici SPI. K připojení používám dostatečně dlouhé vodiče, a proto piny SPI řídící desky nevyvádím na nepájivé pole. Pin MISO je tedy připojen na pin 50 řídící desky, pin MOSI na pin 51 řídící desky, pin CLK na pin 52 řídící desky a pin CS na pin 53 řídící desky.

Poslední komponentou zařízení je termotiskárna. V továrním stavu tiskárna poskytovala pro napájení pouze 2 vodiče. Kvůli vysokému operačnímu proudu je nutno tiskárnu připojit k externímu napájecímu adaptéru. K napájecím vodičům je tedy třeba připojit nejprve napájecí koncovku typu jack samice o průměru 2,1 mm. Vodič od pinu GND musí být připojen k uzemnění obvodu. Vodiče tiskárny jsou dostatečně dlouhé, a proto jsou připojeny přímo na piny řídící desky. RX a TX tiskárny se připojují k sériovým nebo datovým pinům řídící desky. Jelikož Arduino MEGA2560 disponuje konceptem Software Serial, který umožňuje datovým pinům komunikovat po sériové lince, rozhodl jsem se RX a TX tiskárny připojit na datové piny 8 a 9 řídící desky.

3.2 Implementace řízení mp3 modulu

Výrobce pro ovládání mp3 přehrávače neposkytuje připravenou řídící knihovnu. Lze však na internetu nalézt manuál popisující způsob řízení. Ten uvádí, že přístroj přijímá příkazy o velikosti 64 b. Ty se posílají přes sériovou linku. Struktura příkazu je následující:

Tabulka 3.1: Struktura příkazu pro mp3 přehrávač

Pořadí	**Význam**	**Hodnota**
1. byte	Startovací sekvence	0x7E
2. byte	Verze	0xFF
3. byte	Počet bytů příkazu bez startovací a ukončovací sekvence	0x06
4. byte	Příkaz pro přehrávač	číslo příkazu
5. byte	Zpětná vazba	0x00 nebo 0x01
6. byte	Horní byte parametru příkazu	libovolná
7. byte	Dolní byte parametru příkazu	libovolná
8. byte	Ukončovací sekvence	0xEF

Přehrávač podporuje tyto funkce:

Tabulka 3.2: Funkce mp3 přehrávače

Typ příkazu	Kód příkazu	Parametr
Inicializace	0x09	NE
Přehrání následujícího souboru	0x01	NE
Přehrání předchozího souboru	0x02	NE
Přehrání podle indexu	0x03	Index
Zvýšení hlasitosti	0x04	NE
Snížení hlasitosti	0x05	NE
Nastavení přesné hlasitosti	0x06	Hlasitost (0 – 30)
Opakované přehrávání souboru	0x08	Index
Režim spánku	0x0A	NE
Obnova z režimu spánku	0x0B	NE
Reset přehrávače	0x0C	NE
Přehrání aktuálního souboru	0x0D	NE
Pauza	0x0E	NE
Přehrání souboru ve složce	0x0F	Index složky, index souboru
Zastavení přehrávání	0x16	NE
Nastavení D/A převodníku	0x1A	Zap/vyp
Přehrávání se zvolenou hlasitostí	0x22	Index, hlasitost

Jelikož reprezentace hardwarových prvků se v prostředí Arduino provádí přes třídy, rozhodl jsem se i řízení mp3 přehrávače realizovat pomocí třídy.

```
class Mp3{
    private:
        SoftwareSerial * player;
    public:
        Mp3(const byte RX,const byte TX){
            player = new SoftwareSerial(RX, TX);
        }

        ~Mp3(){
            if(player){
             delete player;
            }
        }
```

Obrázek 3.1: Definování mp3 modulu

Pro každý příkaz jsem vytvořil funkci, která naplní buffer jeho strukturou a odešle jej do mp3 přehrávače. Tím se zjednoduší odesílání příkazů.

3.3 Implementace funkcí pro usnadnění ovládání

Knihovna, kterou jsem pojmenoval BusLib, v sobě obsahuje funkce, které využívají propojení více periférií. Implementovány byly především základní funkce, které by měly usnadnit budoucí vývoj celého zařízení.

```
void Mp3::playIndex(uint16_t index){ /* play file with index */
    static uint8_t Send_buf[8];
    Send_buf[0] = 0x7e; //starting byte
    Send_buf[1] = 0xff; //version
    Send_buf[2] = 0x06; //the number of bytes of the command without starting byte and ending byte
    Send_buf[3] = 0x03; //play track with index
    Send_buf[4] = 0x01; //0x00 = no feedback, 0x01 = feedback
    Send_buf[5] = (uint8_t)(index >> 8); //datah
    Send_buf[6] = (uint8_t)index; //datal
    Send_buf[7] = 0xef; //ending byte
    for(uint8_t i=0; i<8; i++){
        player->write(Send_buf[i]);
    }
    delay(200);
    return;
}
```

Obrázek 3.2: Funkce pro odesílání příkazu

Základem knihovny je třída s názvem BusPC, která má v sobě všechny funkce naimplementované.

```
class BusPC{
      private:
         byte lcd_cols;
         byte lcd_rows;
         char* deblank(char* input);

      public:
         BusPC(){}
         int init(LiquidCrystal & lcd, const byte lcd_col, const byte lcd_row,const byte chipSelect,Mp3 & player);
         int readInt(LiquidCrystal & lcd, Keypad & key);
         char readChar(LiquidCrystal & lcd, Keypad & key);
         byte readByte(LiquidCrystal & lcd, Keypad & key);
         long readLong(LiquidCrystal & lcd, Keypad & key);
         char* readString(LiquidCrystal & lcd, Keypad & key);
         void scrollPrint(LiquidCrystal & lcd, Keypad & key, char * text);
         int readFile(char* filename, LiquidCrystal & lcd, Keypad & key);
         int filePrint(char* filename, Adafruit_Thermal& printer);
         int loadTracks(char* filename, Track* track);
         int fileLines(char* filename);
};
```

Obrázek 3.3: Příkazy pro mp3 modul

3.3.1 Funkce init

Prvotní a velmi důležitá funkce má za úkol nainicializování některých komponent zařízení. Jejími parametry jsou třída LCD, počet řádků a sloupců displeje, číslo pinu pro signál CS a třída mp3. Po zavolání funkce nainicializuje komponenty LCD, SD modul a mp3 modul.

Funkce neumí nainicializovat klávesnici, protože třída Keypad obsahuje speciální konstruktor. Aby se tento konstruktor správně zavolal, funkce init by musela obsahovat nadměrné množství parametrů, a proto jsem se rozhodl zůstat při jednoduší variantě inicializace klávesnice. Podobně z důvodu již dostatečného množství parametrů jsem se rozhodl vynechat inicializace tepelné tiskárny.

Funkce nejprve zavolá inicializační funkci pro displej, kde by se mělo zobrazit hlášení, že displej je v pořádku. Následuje inicializace SD modulu. Při nesprávné inicializaci funkce zahlásí chybu a dále nepokračuje. Zároveň nastaví návratovou hodnotu na -1, což je reprezentováno jako chyba. Pokud se SD modul v pořádku nainicializuje, funkce přes sériovou linku odešle mp3 přehrávači inicializační příkaz a vrátí nechybovou hodnotu 0.

3.3.2 Funkce readInt

Jelikož žádná knihovna nedisponuje možností načítat vstupy do proměnných, rozhodl jsem se díky provázání displeje s klávesnicí vytvořit takové funkce pro nejčastěji využívané datové typy. Prvním datovým typem je Integer, neboli celá čísla. Přestože vývojové prostředí Arduino využívá jako hlavní programovací jazyk C, velikost datového typu Integer je zde 2 B. Jelikož se jedná o znaménkový typ, rozsah hodnot je -32,768 – 32,767. Jelikož klávesnice nedisponuje vhodným způsobem pro zadání znaménka, rozhodl jsem se implementovat pouze možnost zadávat kladná čísla. Parametry funkce jsou výstupní displej a klávesnice pro zadávání hodnot.

Funkce nejprve načítá vstupní znaky z klávesnice do pole znaků a kontroluje, zda-li není zadána nesprávná hodnota. V momentě zadání nesprávné hodnoty se funkce ukončí a vrátí hodnotu -1, která signalizuje chybný výstup. Funkce zároveň také kontroluje, zda-li není zadávaná hodnota mimo rozsah datového typu. Dále funkce čeká na potvrzení správné hodnoty, a to konkrétně stiskem tlačítka #. Po stisku tohoto tlačítka funkce provede převod z pole znaků na celočíselnou hodnotu. Převod probíhá tak, že se od každého čísla odečte hodnota 48, což je číslo, kterým je v ASCII tabulce reprezentována číslice 0. Číslice jsou v tabulce v řadě za sebou, a proto je odečítací koeficient stejný. Po odečtení se číslo vynásobí i-tou mocninou čísla 10, kde i reprezentuje pozici převáděného znaku v tabulce. Výsledné číslo se přičte k proměnné result, která na začátku algoritmu začíná číslem 0. Po skončení algoritmu bude tedy v proměnné result uložen číselný ekvivalent vstupní hodnoty.

Pro převod typu pole znaků na celé číslo existuje v jazyce C funkce s názvem atoi. Nicméně kvůli rozdílům velikostí datových typů jsem se rozhodl ji nepoužít.

3.3.3 Funkce readChar

Stejně jako v programovacím jazyce C, je zde datový typ char reprezentován jako znak o velikosti 1 B. Při stisknutí tlačítka klávesnice proběhne detekce stisku tak, že funkce postupně na řádky přivádí hodnotu logické 1 a otestuje, zda-li se hodnota logické 1 objevila i na některém ze sloupců. Pokud jsou obě podmínky splněny, stisknuté tlačítko je již funkci zřejmé. Při inicializaci displeje se pomocí dvourozměrného pole každému tlačítku přiřadí jeho znaková reprezentace.

Funkce readChar tedy pouze v nekonečné smyčce testuje stisky kláves. Klávesa # slouží pro potvrzení vstupu a klávesa * slouží pro vymazání a možnost opravit nesprávný vstup. Ostatní tlačítka klávesnice pouze ukážou na displeji zvolený znak.

Variantou funkce by mohla být možnost okamžitého přepsání znaku při stisku jiné znakové klávesy. V takovém případě by ale mohlo mnohem jednodušeji docházet k překlepům uživatelů, což by mělo na svědomí neblahé následky. Původní implementovaná varianta je tedy vhodnějším řešením.

3.3.4 Funkce readByte a readLong

Datový typ byte reprezentuje čísla v rozsahu 0 – 255. Velikost datového typu je 1 B. Je to tedy typ bezznaménkový.

Datový typ long reprezentuje čísla v rozsahu -2 147 483 648 – 2 147 483 647. Rozsahově tedy odpovídá datovému typu int v jazyce C. Velikost datového typu long jsou 4 B.

Algoritmus použitý pro čtení a převod je totožný s algoritmem funkce readInt. Rozdíly ve funkcích tvoří pouze kontrolovaná velikost vstupu. Jelikož typ byte je bezznaménkový, chybný výstup nelze reprezentovat jako číselnou hodnotu. Proto funkce readByte obsahuje navíc parametr typu boolean předávaný odkazem, který lze použít pro detekci správného výstupu.

3.3.5 Funkce readString

I přesto, že Arduino má vlastní textový datový typ string, používá tato funkce reprezentaci typickou pro jazyk C, což je pole znaků. Tato funkce může být využitelná hlavně při manipulaci s textovými soubory na SD kartě.

Funkce podobně jako funkce readChar čte stisky kláves v nekonečné smyčce, dokud nedojde k potvrzení. Hlavním rozdílem je ale možnost zadání až 50ti znaků. Funkce také předpokládá, že rozsah řádku displeje může být překročen. Tvůrce knihovny pro LCD nabízí funkci s názvem Autoscroll. Ta při překročení rozsahu řádku posune vypsaný text. Nevýhodou této funkce ale je, že při mazání znaků by uživatel přišel o možnost vidět, jaké hodnoty zadal. Z tohoto důvodu jsem se rozhodl vytvořit vlastní posouvací algoritmus, který počítá pozice znaků na základě stisků potvrzovací nebo mazací klávesy. Při překročení pravého okraje displeje algoritmus vymaže displej a zobrazuje od první pozice dále nově zadané hodnoty. V případě, že uživatel vymaže takový počet znaků, že displej zůstane prázdný, mohou nastat 2 možnosti. V prvním případě vstupní text není delší než rozměr řádku displeje. Tím pádem při mazání displej zůstane prázdný. Ve druhém případě je vstupní text delší než rozměr řádku, což znamená, že displej byl minimálně jednou posunut. Tehdy se zobrazí na displeji počet znaků o počtu velikosti řádku. Při překročení délky 50 znaků program zahlásí chybu.

3.3.6 Funkce scrollPrint

Klasické vypisovací funkce pro displej obsahují složité algoritmy pro posouvání textu po displeji. Text, který rotuje po displeji, se obtížně zastavuje tak, aby bylo možné text přečíst po jednotlivých znacích. Funkce scrollPrint umožňuje ovládání posuvu textu, který velikostí překračuje rozměr řádku. Tlačítko # posouvá text doleva a tlačítko * posouvá text doprava. Lze si všimnout, že podobný princip je již použitý ve funkci readString, kde ale probíhá posuv textu o celou délku řádku. Zde probíhá posuv pouze o jednu pozici. Maximální velikost zobrazeného textu není limitována. Čtení textu lze přerušit stisknutím tlačítka B na klávesnici.

3.3.7 Funkce readFile

Úkolem funkce readFile je umožnit uživatelem ovládané zobrazení souboru po řádcích na displeji. Prvotním problémem, na který programátor narazí při čtení souboru z karty SD, je absence funkce pro přečtení řádku ze souboru. Je tedy v prvním kroku nutné vytvořit detekci řádků.

Označení pro konec řádku se liší podle používaného operačního systému v počítači. Systém Microsoft Windows používá pro označení nového řádku po sobě jdoucí znaky `\r` a `\n`. Narozdíl od toho operační systém Linux využívá pro označení nového řádku pouze znak `\n`. Jelikož oba využívají ve své detekci znak `\n`, rozhodl jsem se použít pro označení nového řádku také znak `\n`. Aby se předešlo chybám u textových souborů vytvořených v operačním systému Microsoft Windows, byla do algoritmu přidána i reakce na posloupnost `\r\n`, kde program znak `\r` vynechá. Po načtení řádku následuje volání funkce scrollPrint, která vypíše načtený řádek na displej.

Problém nastává v momentě načítání posledního řádku. Ten totiž není zakončený znaky pro konec řádku. Proto se provede na závěr zobrazení znaků načtených od posledního konce řádku až do konce souboru.

3.3.8 Funkce filePrint

Úkolem této funkce je přečtení souboru a vytisknutí jeho obsahu po řádcích na tepelné tiskárně. Tato funkce používá naprosto stejný postup jako funkce readFile, s tím rozdílem, že namísto výstupu na displej je použita funkce pro tisk řetězce a odřádkování papíru.

3.3.9 Funkce loadTracks

Tato funkce má za cíl načíst strukturu právě jedné cesty autobusu z textového souboru. Struktura souboru byla vybrána tak, že řádek musí obsahovat právě jednu dvojici typu název města x index zvukového souboru.

Princip čtení souboru je stejný, jako u funkcí scrollPrint a readFile. Načtený řádek nejprve zpracuje funkce deblank, která má za úkol vymazání pře-

bytečných mezer před a za textem řádku. Program poté sleduje strukturu řádku ve 2 krocích. V prvním kroku program hledá název stanice. Ten může obsahovat jakékoliv znaky ASCII tabulky v rozsahu 33 – 125. Oddělovačem názvu stanice a indexem zvukového souboru byla vybrána právě jedna mezera. V okamžiku nalezení mezery program čte index souboru, který se může pohybovat v rozsahu 0 – 999. Výslednou dvojici parametrů poté uloží do pole struktur s názvem Track. Struktura Track v sobě obsahuje právě parametry název a index.

Detekce souboru má nastavená přísná pravidla a jejich nedodržení vede k okamžitému přerušení procesu a vrácení chybové hlášky.

3.3.10 Funkce fileLines

Úkolem této funkce je zjištění přesného počtu řádků v souboru. Tato funkce může být užitečná např. pro alokace velikostí datových struktur závislých na velikosti souboru.

Čtení souboru neprobíhá stejně jako u předchozích funkcí. Funkce fileLines totiž nemá zapotřebí ukládat celý řádek, ale pouze počítá počet výskytů ukončení řádků tj. počet výskytů znaku `\n`. Načítací funkce z knihovny SD umožňuje pouze načtení 1 B dat ze souboru.

Jelikož soubor není ukončen znakem `\n`, musí být k výsledné hodnotě přičtena hodnota 1.

3.4 Implementace prototypu řídícího softwaru

Prototyp řídícího softwaru je zjednodušená verze výsledného softwaru, který bude použit pro ovládání hotového systému z návrhové části práce. Aplikace si klade za cíl představení funkčnosti všech prvků zařízení a předvedení funkcí z implementovaných knihoven.

V úvodu programu se nachází vložení potřebných knihoven. Vloženy jsou jak knihovny od výrobců pro každou periférii, tak vytvořené knihovny pro mp3 přehrávač a celé zařízení.

```
#include <mp3.h>
#include <SoftwareSerial.h>
#include <buslib.h>
#include <LiquidCrystal.h>
#include <Keypad.h>
#include <SPI.h>
#include <SD.h>
#include <Adafruit_Thermal.h>
```

Obrázek 3.4: Použité knihovny funckí

Po vkládání knihoven následuje deklarování komponent. Standartní způsob je ten, že se zapíše typ objektu, a do konstruktoru jako parametry vloží čísla pinů, na které je zařízení připojeno. Jednu z výjimek tvoří tepelná tiskárna, která do konstruktoru vkládá referenci na objekt SoftwareSerial, čili sériovou linku. Druhou výjimku tvoří klávesnice. U klávesnice je nutno nejprve definovat počet řádků a sloupců. Poté je třeba vytvořit mapu tlačítek. Mapa se tvoří tak, že se vytvoří dvourozměrné pole znaků o rozměru řádky x sloupce. Každá buňka poté reprezentuje, na jaký znak se tlačítko na příslušné souřadnici přeloží. Inicializace klávesnice pokračuje deklarací dvou celočíselných polí. Ty určují, na které piny jsou připojeny řádky a na které sloupce. Na závěr přichází samotná deklarace klávesnice.

```
const byte numRows= 4; //number of rows on the keypad
const byte numCols= 4; //number of columns on the keypad

char keymap[numRows][numCols]=
{
{'1', '2', '3', 'A'},
{'4', '5', '6', 'B'},
{'7', '8', '9', 'C'},
{'*', '0', '#', 'D'}
};

byte rowPins[numRows] = {22,23,24,25}; //Rows 0 to 3
byte colPins[numCols] = {26,27,28,29}; //Columns 0 to 3

Keypad key = Keypad(makeKeymap(keymap), rowPins, colPins, numRows, numCols);
```

Obrázek 3.5: Postup inicializace klávesnice

Po inicializaci klávesnice následují deklarace globálních proměnných.

Prostředí Arduino rozděluje program na 2 hlavní části. První částí je část setup, která slouží k nastavení hardwarových komponent a k inicializacím. Druhá část je funkce loop, která zajišťuje běh programu v nekonečné smyčce. Jako první se vždy volá funkce setup.

Při startu aplikace se tedy spustí funkce setup. V té jsou vloženy příkazy a volání funkcí pro inicializace všech periférií včetně funkce init z knihovny BusLib. Pokud se všechny komponenty správně nainicializují, program přivítá uživatele a přejde do funkce loop. Pokud inicializace není úspěšná, program bude vyžadovat nápravu chyb a restart přístroje.

Funkce loop v nekonečné smyčce testuje stisk tlačítek klávesnice a na základě stisků zobrazuje na displeji aktuální položku menu. V menu se lze pohybovat stisknutím tlačítka „2“ pro směr nahoru a tlačítka „8“ pro směr dolů. Potvrzení vždy probíhá stiskem tlačítka #. Po kontrole stisku tlačítek funkce loop volá funkci showMenu, která má za úkol měnit položky menu a reagovat na potvrzení volby. Základní menu má 3 volby:

- Zobraz trasu
- Vydat listek

- Reset

Při potvrzení možnosti Zobraz trasu se opět zavolá nekonečná smyčka testující stisk tlačítek a zobrazování vnořených možností menu. Nyní přibyl test na tlačítko *, které se vrací zpět na předchozí obrazovku menu. Po kontrole stisku tlačítek funkce path volá funkci showTrackMenu, která taktéž mění položky menu a reaguje na potvrzení volby. Vnořené menu má 3 volby:

- **Vybrat soubor**: Tato funkce umožňuje vybrat soubor se strukturou pro zobrazení a hlášení zastávek. Funkce podporuje možnost vnořování se do složek. Knihovna SD při prohledávání souborového systému neumožňuje návrat k předchozí volbě, ale umožňuje cyklický pohyb mezi soubory ve složce. Pro zvolení následujícího souboru je funkce nastavena na reakci na tlačítko „8“ na klávesnici a pro výběr souboru nebo vnoření do složky reaguje na tlačítko #. Pokud uživatel nechce zvolit soubor, funkce umožňuje návrat ze složky nebo funkce pomocí tlačítka *.

- **Zacit cestu**: Tato funkce slouží k zobrazování zvolené trasy a hlášení zastávek. Na začátku, pokud je vybrán soubor, alokuje pole datové struktury Track s pomocí funkce k zjištění počtu řádků z knihovny BusLib. Poté následuje zavolání funkce loadTracks z knihovny BusLib, která naplní pole struktury Track hodnotami přečtenými z vybraného souboru. Pro soubor byl vybrán speciální formát, který dokáže soubor rozdělit na zastávky a podzastávky. Jakýkoliv název začínající malým písmenem je reprezentovaný jako podzastávka. V takovém případě hlášení oznámí nejprve hlavní zastávku, a poté podzastávku. Jelikož mezi znaky podle funkce loadTracks smí být pouze jeden znak mezery, lze od sebe slova v názvu oddělovat znakem „_“. Pokud funkce loadTracks vrátí chybovou hodnotu, funkce vypíše chybu a ukončí se. V případě, že vše bylo načteno v pořádku, program přivítá cestující a přejde do nekonečné smyčky, která čte hodnoty z klávesnice. Program pomocí funkce printTrackName vypisuje zastávky a podzastávky tak, že znak „_“ nahradí znakem mezery a první malé písmeno v názvu podzastávky nahradí za velké. Na další stanici lze přejít stiskem tlačítka „6“. V takovém případě program zahlásí aktuální zastávku a nahlásí následující zastávku, a to i včetně podzastávky (pokud nějakou obsahuje). V případě poslední zastávky program zahlásí konečnou zastávku a požádá cestující o vystoupení. Pro případ chyby se lze vrátit na předchozí zastávku stiskem tlačítka „4“. Z funkce se lze do menu vrátit stiskem tlačítka *.

- **Pokracovat**: Tato možnost volá stejnou funkci jako možnost „Zacit cestu“. Jediným rozdílem je, že nečte načtenou strukturu od začátku, ale od pozice, ve které se funkce naposledy nacházela.

Při potvrzení možnosti Vydat listek se zavolá funkce printTicket, která nejprve pomocí funkce z knihovny BusLib vytiskne soubor **ticket.txt**. Na závěr bude očekávat načtení ceny, neboli celočíselné hodnoty z klávesnice.

Při potvrzení možnosti Reset se provede pouze softwarový reset, protože hardwarový reset by vyžadoval spojení pinu Reset na řídící desce s datovým pinem. To ale kvůli kondenzátoru, který blokuje resetovací signály není možné. Jako náhrada se tedy zavolá tvrdý skok na programovou adresu 0 a jako hardwarová možnost resetu zůstává možnost zmáčknutí resetovacího tlačítka řídící desky.

Kapitola 4

Testování

4.1 LCD displej

Pro testování funkčnosti displeje byl po připojení k řídící desce využit krátký testovací program s funkcemi z knihovny LiquidCrystal, kterou poskytuje výrobce zdarma na svých oficiálních stránkách. Po softwarové stránce je displej v naprostém pořádku. Po stránce hardwarové displej vykazuje malé výpadky zobrazování, které jsou způsobeny příležitostným přerušením vodivosti kabelu, připojeným na jezdci potenciometru. Dostatečným upevněním a minimálním pohybem potenciometru lze tyto výpadky snížit na minimum.

4.2 SD modul

Pro testování SD modulu byl využit testovací program s funkcemi z knihoven SD.h a SPI.h. Zpočátku modul vykazoval špatnou schopnost načítání karty SD, která se ale ukázala jako vada vodiče. Ten byl poté vyměněn. Tím byla chyba zcela odstraněna. Čtení souborů a složek se ukázalo být v naprostém pořádku, nicméně zápis do souboru, který není vytvořen, vykazuje problémy. Proto se u stávajícího modulu doporučuje soubory pouze číst, případně modul vyměnit za kvalitnější typ.

4.3 Klávesnice

Pro testování klávesnice byla využita jednoduchá aplikace, která stisknutou klávesu zobrazila na sériovém monitoru v počítači. Odezva klávesnice se ukázala jako velmi dobrá a všechna tlačítka vracejí správný výsledek. Jedinou výtkou může být tuhost stisku klávesy, která zřídka povede k nechtěnému dvojitému stisknutí.

4.4 Tiskárna

K testování tiskárny slouží testovací program od firmy Adafruit. Ten tvoří testovací výtisk, ve kterém používá všechny funkce, které knihovna firmy poskytuje. Test vyšel v naprostém pořádku.

4.5 Knihovna BusLib

Knihovna byla testována v průběhu tvorby pro každou funkci. Funkce vykazují správné chování bez nepředvídatelných událostí.

4.6 Řídící software

Pro testování řídícího softwaru byli kontaktováni profesionálové z firmy Krkonošské metro, z. ú. Testující subjekty se shodly, že zařízení obsahuje jednoduché ovládání. Velkou výhodou systému je komunikace v českém jazyce. Časovou odezvu přístroje a algoritmus pro skládání hlasových hlášení prohlásily subjekty za velmi dobrý.

4.7 Náklady

Jelikož každý prodejce nabízí trochu jinou cenu za danou periférii/materiál, ceny jsou pouze orientační.

Tabulka 4.1: Náklady na materiály pro zařízení

Položka	**Cena (orientační) (Kč)**
Arduino Mega2560	1300
Adafruit Mini Thermal Printer	1600
Nepájivé kontaktní pole	350
Arduino LCD Display 16x2	120
Arduino klávesnice 4x4	80
Arduino čtečka SD karet	50
Serial MP3 player	150
Zdroj pro tiskárnu	280
Konektor pro tiskárnu	30
10 mF kondenzátor	30
10 kΩ potenciometr	20
Sada propojovacích vodičů	80
Plexisklo 50x50 cm	200
Závitová tyč + kryt	60
Šrouby + matice	10
Celkem	**4360**

Závěr

Tato práce se zabývala návrhem a tvorbou zařízení pro poskytování informací a obsluhy cestujících v hromadné dopravě. Nejprve ukázala existující systémy se stejným využitím, a poté ukázala metody návrhů takových systémů. Poté proběhl kompletní návrh hardwarové části systému pro firmu Krkonošské metro, z. ú. Celý systém byl pro bakalářskou práci příliš komplexní, a proto byla v další části zrealizována pouze zjednodušená verze, pro kterou byla vytvořena knihovna funkcí pro zjednodušení budoucího vývoje. Výsledek je prezentovaný pomocí jednoduché aplikace, která ukazuje funkčnost všech částí zařízení. Všechny části práce byly otestovány.

Zařízení má dispozice a je zároveň připraveno pro rozšíření o doplňující prvky jako jsou GSM a GPS moduly. Má také dispozice pro rozšíření o ovládání panelů Pragotron a BUSE podle prvotního návrhu, nicméně pro zobrazovací panely je nejprve nutno vytvořit kontroler pro komunikaci přes sériovou linku.

Literatura

[1] Dopravní podnik města Mostu: Informační systém veřejné dopravy. duben 2016. Dostupné z: http://www.dpmost.cz/iprmisvd

[2] Gelová, E.: Inteligentní dopravní systémy ve veřejné dopravě. listopad 2011.

[3] Chovanec, M.: *Audio-vizuální informační systém pro dopravní aplikace*. Bakalářská práce, České vysoké učení technické v Praze, Fakulta informačních technologií, Praha, 2013.

[4] Matějů, M.: *Návrh informačního systému pro železnici se zobrazovacím zařízením PRAGOTRON*. Diplomová práce, České vysoké učení technické v Praze, Fakulta dopravní, Praha, 2011.

[5] Hlásek, J.: Pragotron signaltron tabule. 2010, [Online]. Dostupné z: http://najd.cz/blog/postimages/signaltron/pragotron-signaltron-tabule.jpg

[6] Pilný, J.: *Informační portál s WWW rozhraním*. Diplomová práce, Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních techlogií, Brno, 2014.

[7] BUSE S. R. O., Blansko: *Technický popis panelu BUSE BS 110*. 1997.

[8] Schimmer, Z.: Řídící a informační systém MHD. květen 2010. Dostupné z: http://www.sdp-cr.cz/WD_FileDownload.ashx?wd_systemtypeid=34&wd_pk=WzY0NyxbMzBdXQ%3D%3D

[9] MHD-Ostrava: 4-segmentový boční panel. 2015, [Online]. Dostupné z: http://www.mhd-ostrava.cz/obrazky/grafika/linkova_orientace/bocni_4segmentovy_5.gif

[10] Müller, V.: Princip a historie jehličkové tiskárny. duben 2016. Dostupné z: http://www.kopirky.com/princip-a-historie-jehlickove-tiskarny/

[11] HP TRONIC Zlín, .: Tiskárna pokladní Epson TM-U220PD-052 (C31C518052) černá. [Online]. Dostupné z: https://img.kasa.cz/k-foto/ilustrace/800/5/8/8/product_1502885.jpg

[12] Pandatron: Termotiskárny, jejich princip a vlastnosti. červenec 2008. Dostupné z: http://pandatron.cz/?477&termotiskarny_jejich_princip_a_vlastnosti_-_referat

[13] Adafruit: Mini Thermal Receipt Printer. 2012, [Online]. Dostupné z: https://cdn-learn.adafruit.com/assets/assets/000/001/942/medium800/components_thermalprinter_LRG.jpg?1396777649

[14] Rydval, S.: Princip a fungování GPS. prosinec 2005. Dostupné z: http://www.rydval.cz/phprs/view.php?cisloclanku=2005110301

[15] GPS.gov: Frequently asked questions. červenec 2014. Dostupné z: http://www.gps.gov/support/faq/#sats

[16] Arduino Ublox GPS. 2016, [Online]. Dostupné z: http://cdn.instructables.com/F5X/KQ4U/HWAFTWPM/F5XKQ4UHWAFTWPM.MEDIUM.jpg

[17] Bajdi: Mp3 module. [Online]. Dostupné z: http://www.bajdi.com/wp-content/uploads/2013/02/vs1053-mp3-module.jpg

[18] Zandl, P.: Principy fungování sítě GSM. srpen 1997. Dostupné z: http://mobil.idnes.cz/principy-fungovani-site-gsm-d1t-/mob_tech.aspx?c=970825_0003060_mob_tech

[19] Arduino: Arduino GSM module. [Online]. Dostupné z: https://www.arduino.cc/en/uploads/Main/A000043_landing.jpg

[20] Alza.cz: Arduino Mega 2560. [Online]. Dostupné z: https://i.alza.cz/ImgW.ashx?fd=f4&cd=ARD206a&i=1.jpg

[21] hwkitchen.cz: Arduino a displeje II. květen 2014. Dostupné z: http://www.hwkitchen.com/news/a13-arduino-a-displeje-ii-/

[22] santy.cz: Arduino LCD 16x4. [Online]. Dostupné z: http://www.santy.cz/data/product/171_754.jpg

[23] instructables.com: Keyboard 4x4. [Online]. Dostupné z: http://cdn.instructables.com/FSM/WKU0/HX0DQ2Y3/FSMWKU0HX0DQ2Y3.MEDIUM.jpg

[24] SD Reader/Writer Module. [Online]. Dostupné z: http://g02.s.alicdn.com/kf/HTB1h4lZGXXXXXa6XFXXq6xXFXXXw/220414554/HTB1h4lZGXXXXXa6XFXXq6xXFXXXw.jpg

Příloha **A**

Seznam použitých zkratek

LCD Liquid Crystal Display

GPS Global Positioning System

GSM Global Systems for Mobile Communications

PSTN Public Switching Telecommunication Networks

JTS Jednotná telefonní síť

BSS Base Station Subsystem

NSS Network and Switching Subsystem

OSS Operation Subsystem

KISyD KrM Komplexní Informační Systém pro Dopravu Krkonošského Metra

OB 01 Ovládací bednička 01

OIS Odbavovací a informační systém

RFID Radio Frequency Identification

MP3 MPEG-2 Audio Layer III

MPEG Motion Picture Experts Group

WAV Waveform audio file format

AAC Advanced Audio Coding

FLAC Free Lossless Audio Codec

ALAC Apple Lossless Audio Codec

SPI Serial Peripheral Interface

PWM Puls Wide Modulation

UART Universal Asynchronos Reciever/Transmitter

ICSP In-Circuit Serial Programming

USB Universal Serial Bus

SRAM Static Random Access Memory

EEPROM Electronicaly Erasable Programmable Read Only Memory

ISP In-System Programmer

COM Communication

TTL Transistor Transistor Logic

ASCII American Standart Code for Information Interchange

FAT File Allocation Table

DOS Disk Operating System

Příloha **B**

Obsah přiloženého CD

- `readme.txt` ..... stručný popis obsahu CD
- `src`
  - `impl` ..... zdrojové kódy implementace
    - `buslib.cpp` ..... implementace knihovny BusLib
    - `buslib.h` ..... hlavička knihovny BusLib
    - `demo.ino` ..... implementace demoaplikace
    - `mp3.cpp` ..... implementace knihovny mp3
    - `mp3.h` ..... hlavička knihovny mp3
  - `thesis` ..... zdrojová forma práce ve formátu LaTeX
- `text` ..... text práce
  - `BP1.pdf` ..... text práce ve formátu PDF
  - `BP1.ps` ..... text práce ve formátu PS