



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

TESTOVACÍ STOLICE PRO MIKROKONTROLÉROVÉ KITY

TEST BENCHES FOR MICROCONTROLLER KITS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Ivo Žaludek

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Zdeněk Bradáč, Ph.D.

BRNO 2024



Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Student: Bc. Ivo Žaludek

ID: 221032

Ročník: 2

Akademický rok: 2023/24

NÁZEV TÉMATU:

Testovací stolice pro mikrokontrolérové kity

POKYNY PRO VYPRACOVÁNÍ:

Navrhněte koncepci elektronického systému pro komplexní testování výukových kitů. Zaměřte se mikrokontrolerový systém v takové konfiguraci, aby umožnil bezpečné připojení spektra HW modulů a jejich komplexní obvodový test. Systém navrhněte, realizujte, osadte a oživte. Vybavte programovým vybavením pro komplexní testy a demonstруйте správnou funkci.

1. Proveďte literární rešerši a internetový průzkum.
2. Navrhněte koncepci systému tak, aby byl vysoce spolehlivý.
3. Navrhněte obvodová schémata a zrealizujte HW a oživte ho.
4. Vytvořte programové vybavení.
5. Zhodnoťte dosažené výsledky.

DOPORUČENÁ LITERATURA:

Pavel Herout: Učebnice jazyka C, KOPP, 2004, IV. přepracované vydání, ISBN 80-7232-220-6

Dle pokynů vedoucího práce.

Termín zadání: 5.2.2024

Termín odevzdání: 15.5.2024

Vedoucí práce: doc. Ing. Zdeněk Bradáč, Ph.D.

doc. Ing. Petr Fiedler, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Předložená práce se zabývá návrhem testovací stolice pro funkční testování výukových kitů. Kity jsou ve formě samostatných desek plošných spojů, například modul s mikrokontrolerem a jeho periferie jako jsou displej, klávesnice nebo externí paměť. V rámci práce je provedeno srovnání komerčně dostupných testovacích zařízení a jsou shrnuty požadavky na testování jednotlivých desek plošných spojů. Na základě požadavků je pak proveden návrh hardwarové a softwarové části testovací stolice. Součástí práce je také zhodnocení funkčnosti výrobku.

Klíčová slova

Automatizované testování, funkční testování, megaAVR, softwarová architektura MVC, návrh DPS

Abstract

This thesis deals with the design of a test bench for functional testing of educational kits. These kits are in the form of standalone printed circuit boards, for instance, a module with a microcontroller and its peripherals, such as a display, keyboard, or external memory. A comparison of commercially available testers is conducted, and the requirements for testing of each printed circuit board are identified. Based on these requirements, the design of both the hardware and the software of the test bench was developed. The final stage of the project is the evaluation of the functionality of the test bench.

Keywords

Automated testing, functional testing, megaAVR, MVC software architecture, PCB design

Bibliografická citace

ŽALUDEK, Ivo. *Testovací stolice pro mikrokontrolérové kity*. Brno, 2024. Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/159954>. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce Zdeněk Bradáč.

Prohlášení autora o původnosti díla

Jméno a příjmení studenta:	Ivo Žaludek
VUT ID studenta:	221032
Typ práce:	Diplomová práce
Akademický rok:	2023/24
Téma závěrečné práce:	Testovací stolice pro mikrokontrolérové kity

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: 13. května 2024

podpis autora

Poděkování

Děkuji vedoucímu diplomové práce doc. Ing. Zdeňku Bradáčovi, Ph.D. za pedagogickou a odbornou pomoc při zpracování mé diplomové práce a dále doc. Ing. Jakubu Armovi, Ph.D. za vstřícnost a cenné rady ohledně vyhotovení finálního výrobku. Velké poděkování patří také mé rodině a přítelkyni za podporu v průběhu celého studia.

V Brně dne: 13. května 2024

podpis autora

Obsah

SEZNAM OBRÁZKŮ	9
SEZNAM TABULEK.....	10
ÚVOD	11
1. CÍLE PRÁCE	12
2. AUTOMATIZOVANÉ TESTOVÁNÍ.....	13
2.1 ÚVOD DO TESTOVÁNÍ	13
2.2 ZPŮSOBY TESTOVÁNÍ DPS	15
3. PRŮZKUM KOMERČNĚ DOSTUPNÝCH ŘEŠENÍ PRO TESTOVÁNÍ DPS.....	17
4. STANOVENÍ KONCEPCE TESTOVACÍ STOLICE.....	20
5. ROZBOR TESTOVANÝCH ZAŘÍZENÍ	21
5.1 MODUL ATMEGA328P XPLAINED MINI	21
5.1.1 Mikrokontroler ATmega328P	22
5.1.2 Digitální vstupně-výstupní porty.....	22
5.1.3 Analogově digitální převodník	24
5.2 RELÉ MODUL	25
5.3 MODUL HODIN REÁLNÉHO ČASU A EEPROM PAMĚŤ	26
5.4 INKREMENTÁLNÍ ROTAČNÍ ENKODÉR.....	26
5.5 MATICOVÁ KLÁVESNICE.....	27
5.6 MODUL LCD DISPLEJE	28
5.7 REPRODUKTOR	30
5.8 MODUL S TERMISTOREM	30
6. NÁVRH OBVODOVÉHO ŘEŠENÍ.....	32
6.1 VÝBĚR ŘÍDICÍHO MIKROKONTROLERU	32
6.1.1 Rešerše vhodných mikrokontrolerů.....	32
6.1.2 Programování a komunikace s testovaným kitem ATmega328P.....	33
6.1.3 Spouštění testů a signalizace výsledků	33
6.2 ZAPOJENÍ ŘÍDICÍHO MIKROKONTROLERU	33
6.3 NÁVRH NAPÁJENÍ	35
6.4 PŘEVODNÍK NA USB SBĚRNICI	36
6.5 ZAPOJENÍ MODULŮ	36
6.5.1 Zapojení ATmega328P Xplained Mini.....	36
6.5.2 Zapojení relé modulu	37
6.5.3 Zapojení modulu hodin reálného času.....	37
6.5.4 Zapojení inkrementálního rotačního enkodéru	37
6.5.5 Zapojení maticové klávesnice	38
6.5.6 Zapojení modulu LCD displeje	38
6.5.7 Zapojení reproduktoru	38
6.5.8 Zapojení modulu s termistorem.....	38
7. REALIZACE DESKY PLOŠNÝCH SPOJŮ.....	39

7.1	ZPŮSOBY PŘIPOJENÍ TESTOVANÝCH MODULŮ	40
7.2	FÁZE OŽIVENÍ DESKY PLOŠNÝCH SPOJŮ	40
8.	KATEGORIE TESTŮ JEDNOTLIVÝCH MODULŮ	42
8.1	PLNĚ AUTOMATIZOVANÉ TESTY	42
8.2	TESTY, KDE UŽIVATEL INTERAGUJE S TESTOVANÝM MODULEM	42
8.3	TESTY, KDE UŽIVATEL VYHODNOCUJE SPRÁVNOU FUNKCI MODULU	42
9.	PROGRAMOVÉ VYBAVENÍ.....	43
9.1	MIKROPROGRAMOVÉ VYBAVENÍ	43
9.1.1	<i>Vývojové prostředí</i>	<i>44</i>
9.1.2	<i>Ladění</i>	<i>44</i>
9.1.3	<i>Firmware pro modul s ATmega328P</i>	<i>44</i>
9.1.4	<i>Firmware pro testovací stolici</i>	<i>45</i>
9.1.5	<i>Programování modulu s ATmega328P</i>	<i>46</i>
9.1.6	<i>Poznámky k implementaci testů</i>	<i>48</i>
9.2	GRAFICKÉ UŽIVATELSKÉ ROZHŘANÍ.....	50
9.2.1	<i>Použitý návrhový vzor.....</i>	<i>50</i>
9.2.2	<i>Ukládání protokolů o zkoušce.....</i>	<i>52</i>
9.2.3	<i>Popis grafického rozhraní pro jednotlivé kategorie testů</i>	<i>52</i>
9.2.4	<i>Spouštění aplikace grafického rozhraní.....</i>	<i>53</i>
10.	VYHODNOCENÍ TESTŮ	55
11.	ZÁVĚR.....	56
	LITERATURA.....	58
	SEZNAM SYMBOLŮ A ZKRATEK	63
	SEZNAM PŘÍLOH.....	65

SEZNAM OBRÁZKŮ

3.1	Příklad testeru s pohyblivými piny „flying probe“ [9]	18
5.1	Náhradní schéma vstupně-výstupního pinu [15]	23
5.2	Blokové schéma analogově digitálního převodníku s postupnou aproximací [17]	24
5.3	Výstup enkodéru při otáčení po (CW) a proti (CCW) směru hodinových ručiček [22]	27
5.4	Vnitřní zapojení maticové klávesnice [23]	28
5.5	Zapojení tlačítek na modulu displeje [24]	29
5.6	Vnitřní zapojení modulu s termistorem [27]	30
6.1	Zapojení konektorů ISP (vlevo) a JTAG (vpravo)	35
7.1	3D vizualizace DPS v programu KiCad (pohled shora)	39
7.2	Osazená deska plošných spojů se všemi testovanými moduly	41
9.1	Blokové schéma použití testovací stolice	43
9.2	Hlavní smyčka programu	44
9.3	Schéma zapojení reset pinu kitu ATmega328P Xplained Mini [42]	47
9.4	Vývojový diagram spuštění testu z grafického rozhraní	51
9.5	Ukázka hlavního okna grafického rozhraní	53
11.1	Testovací stolice se všemi testovanými moduly	67

SEZNAM TABULEK

5.1	Postup testování DIO	23
5.2	Postup testování ADC	25
6.1	Seznam vstupů a výstupů všech modulů	32
6.2	Seznam maximálního proudového odběru všech komponent	35
9.1	Příklad uloženého protokolu o zkoušce.....	52

ÚVOD

S nárůstem složitosti elektronických systémů se testování desek plošných spojů stává nezbytným prvkem zajišťujícím spolehlivý provoz a bezpečnost elektroniky. Testování identifikuje nejen možné výrobní chyby, ale přispívá také k odhalení chyb v návrhu obvodů nebo programu zařízení.

Tato diplomová práce se zabývá návrhem elektronického systému pro komplexní testování výukových kitů s důrazem na spolehlivost a odolnost navrženého řešení. Výukový kit se skládá z desky s mikrokontrolerem a rozšiřujících modulů. Základním požadavkem na testovací stolicí je, aby umožnila provedení obvodového testu. To však pro ověření korektní funkce některých modulů není dostačující, takže je ve většině případů prováděn také funkční test modulu.

První částí práce je úvodem do problematiky testování desek plošných spojů (DPS), jsou představeny jednotlivé fáze testování DPS a požadavky na testované zařízení. Následně je proveden průzkum trhu testerů, z ekonomických důvodů je však rozhodnuto pro vytvoření vlastního řešení. Koncepce testovací stolice je definována v kapitole 4.

Další část práce (kapitola 5) se zabývá popisem jednotlivých modulů výukového kitu. Pro každý modul je stanoven testovací plán, tedy postup, jak otestovat, zda daná komponenta funguje správně. Samotný návrh zařízení začíná od kapitoly 6, kde jsou popsány kroky návrhu obvodového řešení. Podle testovacího plánu jsou v kapitole 6.1 sestaveny požadavky na mikrokontroler, který spouští jednotlivé testy modulů. Návrh a oživení desky plošných spojů je pak popsán v kapitole 7. Kapitole 9, která se věnuje návrhu programového vybavení, předchází kapitola 8, v níž jsou testy rozřazeny do jednotlivých kategorií. Pro smysluplný návrh softwaru je tato kategorizace testů důležitým bodem. Mikroprogramové vybavení testovací stolice i testovaného modulu s mikrokontrolerem je pak popsáno v kapitole 9.1. Kapitola 9.2 se pak věnuje návrhu grafického uživatelského rozhraní, které je spouštěno na počítači. Práce je završena zhodnocením funkčnosti dodaného řešení pro testování výukových kitů v kapitole 10. Součástí práce je také návod k použití testovací stolice, který je v příloze práce. Splnění zadaných cílů a dosažené výsledky jsou zhodnoceny v závěru práce.

1. CÍLE PRÁCE

Cílem práce bylo navrhnout jednoúčelové testovací zařízení (dále testovací stolice) určené pro testování modulů použitých ve výuce. Moduly jsou ve formě samostatných desek plošných spojů (DPS). Některé moduly plní funkci uživatelského rozhraní (například displej nebo klávesnice), jiné jsou periferiemi mikrokontroleru ATmega328P, který je součástí výukové sady. Zatímco u některých modulů stačí pro ověření správné funkce provést obvodový test multimetrem, u jiných je potřeba provádět komplexnější testování ke zjištění, zda nebyl poškozen. Z těchto důvodů byl vznesen požadavek na vytvoření samostatné testovací stolice, která provede testování celé sady modulů a vypíše případné chyby.

V jedné sadě je celkem osm modulů a celkově je ve výuce používáno několik desítek sad. Požadavek na otestování všech modulů je zpravidla párkrát ročně. Vzhledem k poměrně nízké frekvenci testování bylo vedoucím práce rozhodnuto, že není potřeba navrhovat plně automatický tester za cenu vysokých nákladů na výrobu takového zařízení. Některé moduly mohou být testovány poloautomaticky, se zásahem uživatele testovací stolice.

Pro návrh zařízení byl vypracován postup, jakým má být samotné zařízení navrhováno. Tento postup byl v průběhu konzultován s vedoucím práce tak, aby výsledné zařízení a jeho funkce splňovaly kladené požadavky. Jednotlivé kroky návrhu jsou nastíněny v následujícím seznamu:

1. Průzkum trhu s komerčně dostupnými řešeními pro testování DPS (kapitola 3)
2. Stanovení koncepce testovací stolice (kapitola 4)
3. Rozbor testovaných zařízení (kapitola 5)
4. Stanovení postupu testování (kapitola 5)
5. Návrh obvodového řešení (kapitola 6)
6. Návrh DPS (kapitola 7)
7. Realizace programového vybavení (kapitola 9)
8. Vytvoření návodu k použití (v příloze práce)

Jednotlivé kroky výše popsaného postupu jsou blíže rozebrány v jednotlivých kapitolách práce, na něž se číselně odkazují.

2. AUTOMATIZOVANÉ TESTOVÁNÍ

Testování desek plošných spojů (DPS) je jednou z klíčových částí při výrobě elektrických zařízení. Cílem testování je identifikovat chyby ve výrobě, v případě testování DPS ve fázi vývoje pak může být motivací k testování také ověření správného návrhu a funkčnosti DPS. S rostoucí komplexností a miniaturizací elektroniky se stává automatizace testování nezbytnou součástí výroby. K tomu přispívá také ekonomický faktor, jelikož automatizace procesu sice přináší počáteční investici do vhodného testovacího zařízení, výrazně však urychluje testování a zvyšuje tak efektivitu výroby.

2.1 Úvod do testování

Z hlediska testování je důležité rozlišit dva pojmy: verifikace a validace. Verifikace je proces, který zjišťuje, zda byl model správně implementován, tedy zda byly splněny požadavky specifikované v koncepčním popisu modelu. Verifikace tedy srovnává přesnost vytvořeného modelu s referenčním řešením a zjišťuje, zda jsou splněny stanovené požadavky. Validace je pak proces, který ověřuje, zda vytvořený model odpovídá potřebám skutečného světa z hlediska zamýšleného použití modelu. V procesu validace je model porovnáván s výsledky experimentálně získaných dat, aby se ověřilo, zda model věrně reprodukuje chování skutečného systému. V souvislosti s testováním DPS zahrnuje verifikace ověření, zda byla deska plošných spojů správně vyrobena podle návrhu a specifikací. Kontrolují se spoje, správné osazení součástek a propojení komponent. Validace ověřuje, zda deska plošných spojů splňuje zadané požadavky, například z uživatelského hlediska. V této fázi se provádějí simulace a testy situací, ve kterých bude deska používána. [1]

Při testování se využívá dvou různých přístupů. Prvním z nich je testování „černé skříňky“ (Black box testing). Tento přístup nezkoumá vnitřní strukturu systému, pouze sleduje výstupy systému v reakci na vstupní data. Důležitým krokem při vývoji je detailně popsat specifikaci požadavků na daný systém, v opačném případě testování nemusí odhalit některé chyby. Testování „černé skříňky“ pomáhá validovat funkční požadavky na systém, testování se provádí z pohledu cílového uživatele. Výhodou tohoto přístupu je, že testeři nemusejí znát implementaci testovaného systému, ani konkrétní programovací jazyky, ve kterých byl napsán. Postačuje znalost rozhraní testovaného systému. Testování odhaluje nesrovnalosti mezi požadavky na daný systém a jeho reálnou implementací. [2]

Druhým přístupem v testování je testování „bílé skříňky“ (White box testing, či Glass box). Tento přístup zkoumá vnitřní strukturu a implementaci. Při testování softwaru jsou testy vytvářeny na základě zdrojových kódů systému se zaměřením na tok dat v programech. Cílem jednotlivých testů je vyvolat konkrétní příkazy, nebo projít konkrétní větví programu. Během testování je pak možné měřit pokrytí kódu testy.

Z analýzy pokrytí pak lze určit, jakou část programu testy prošly a ověřily jeho funkčnost. Testování „bílé skříňky“ se tedy zaměřuje na zachycení chyb v programu, je mnohem detailnější a implementace testů vyžaduje znalost vnitřní struktury. [2]

Přístupy testování „bílé a černé skříňky“ nemusí být výlučně oddělitelné, pro testování „černé skříňky“ mohou být informace o vnitřních principech vhodné pro lepší pokrytí všech stavů systému.

Testování se provádí v průběhu celého vývoje produktu, ale způsob testování v každé fázi vývoje se liší. Z tohoto hlediska lze rozdělit testování do následujících kategorií: [2]

- **Testování jednotek** – jedná se o testování „bílé skříňky“, je založené na znalosti zdrojového kódu. Jednotkou je v tomto případě samostatně testovatelná část programu. Jeden test většinou pokrývá jen dílčí část kódu, například jednu třídu.
- **Integrační testování** – ověřuje, zda dvě a více jednotek společně pracuje správně. Integrační testování kombinuje přístup „bílé a černé skříňky“, jelikož testuje i funkční požadavky na skupinu jednotek. Pro testování se využívá rozhraní jednotlivých jednotek. Po otestování integrace jednotek je systém připraven k testování systému jako celku.
- **Systémové testy** – ověřuje, že systém jako celek splňuje specifikované požadavky. Využívá se přístup „černé skříňky“.
- **Alpha testování** – cílem je ověřit základní funkčnost produktu. Testování provádí vývojáři. Po dokončení alfa testování je produkt připraven na uživatelské akceptační testování.
- **Uživatelské akceptační testování (UAT)** – jedná se o poslední fázi testování před jeho převzetím zákazníkem. Testování provádí vlastníci podniku nebo cíloví zákazníci, cílem je zjistit, zda systém splňuje obchodní požadavky a očekávání koncových uživatelů.
- **Beta testování** – principem je podobné Alpha testování, rozdíl je však v tom, že produkt již netestují vývojáři, ale vybraná skupina koncových zákazníků. Umožňuje vývojářům opravit produkt před jeho finální distribucí.
- **Regresní testování** – provádí se při vydání nové verze produktu. Ověřuje, že úpravy produktu nepřinesly nové chyby.

Testovací stolice bude provádět testy hotových desek plošných spojů, způsob testování se tedy bude lišit od testování produktu při vývoji. Přesto budou mnohé principy uplatněny. U většiny modulů bude testováno pouze vstupně výstupní chování, z tohoto pohledu se tedy jedná o testování „černé skříňky“. Při tvorbě specifikace testů však bude přihlíženo ke vnitřní struktuře a zapojení modulů. Oba přístupy tedy budou kombinovány.

Požadavkem je otestovat funkčnost jednotlivých modulů. Z tohoto hlediska se tedy jedná o funkční testování. Stanovení pokrytí testy u funkčního testování může být náročné. Existují metodiky, jak stanovit pokrytí testy pro systémové testování, embedded testy nebo pro zátěžové testy. Stanovení přesného pokrytí testy pro testované moduly by

bylo náročnou úlohou, která navíc není předmětem této práce. Přesto byla při stanovení specifikace testů brána inspirace z těchto metodik. [3]

Tři metodiky byly prezentovány v článku [3] a ze dvou metodik byla čerpána inspirace při specifikaci testů. Metoda „Deklarace“ využívá jako vstup ke stanovení pokrytí testy elektrická schémata a rozvržení desky plošných spojů. Metoda „Dedukce“ využívá formálního popisu testu. Předpokládá například, že určité části funkčního bloku jsou pokryty testy, například test externí paměti mikrokontroleru ověří funkci pinů na paměťovém zařízení a funkci určitých pinů na mikrokontroleru. Tohoto přístupu bylo využito například u modulu s ATmega328P. [3]

2.2 Způsoby testování DPS

Desky plošných spojů slouží k propojení jednotlivých součástek elektrického obvodu. Dále zajišťují mechanické ukotvení součástek na desce. DPS mohou být testovány před a po osazení. Před osazením DPS se provádí optická kontrola a testuje se kvalita vodičů a propojení mezi jednotlivými vrstvami desky. Po osazení DPS lze testovat funkčnost jednotlivých součástek, nebo ověřit funkčnost obvodu jako celku. To lze provést několika metodami. Nejjednodušší metodou je vložit DPS do cílového zařízení a zjistit, zda funguje správně. Tato metoda však nemusí odhalit některé vady. Podstatně komplexnější metodou je ruční měření jednotlivých signálů na desce za pomoci měřicích přístrojů jako jsou osciloskopy, multimetry a signálové generátory. Tato metoda je časově velmi náročná a vyžaduje znalost DPS a způsobů měření. [5]

Stejné komplexnosti testů a podstatně kratšího času testování lze dosáhnout automatizací testování. Nejčastějším způsobem automatizovaného testování desek plošných spojů je In-Circuit Testing (ICT). Při ICT testování jsou na DPS položeny kontaktní piny a testuje se existence zkratů, hodnoty odporu, impedance a další faktory určující kvalitu DPS. Kontaktní piny mohou mít pevné rozložení na testovacím přípravku, pak hovoříme o „bed of nails“ přípravku. DPS má kontaktní plochy a na testovacím přípravku jsou odpovídajícím způsobem rozloženy měřicí sondy (piny). DPS se pak vloží do přípravku a provede se měření. Každá deska plošných spojů má zpravidla jiné rozložení kontaktních ploch, proto musí mít každá deska také svůj testovací přípravek. To zvyšuje náklady na testování. Výhodou je vysoká rychlost. [5]

Kontaktní piny nemusí být rozloženy pevně, ale mohou být pohyblivé, pak hovoříme o „flying probe“. Testovací zařízení posouvá piny na určená místa na desce, kde provádí měření. Tyto testery jsou univerzální pro různé DPS, způsob testování se mění změnou programu testeru. Proto jsou tyto testery vhodné, pokud jsou DPS vyráběny v malých sériích. Oproti „bed of nails“ testování je však „flying probes“ pomalejší. [5]

Posledním krokem během testování DPS je funkční testování. Funkční testování ověřuje, že testované zařízení funguje, jak má. Zahrnuje testování funkčnosti zdrojů napájení, funkci mikrokontroleru (MCU), senzorů, uživatelských rozhraní apod. Z analýzy jednotlivých částí DPS se vytvoří testovací plán a navrhnu se body v DPS,

kde budou připojeny testovací sondy. K testování se pak využívá testovací přípravek a testovací software, který realizuje a vyhodnocuje jednotlivé testy. Funkční testování je zpravidla velmi specifické pro každou DPS a verzi programového vybavení, takže se často pro každý typ DPS vytváří vlastní testovací zařízení. [6]

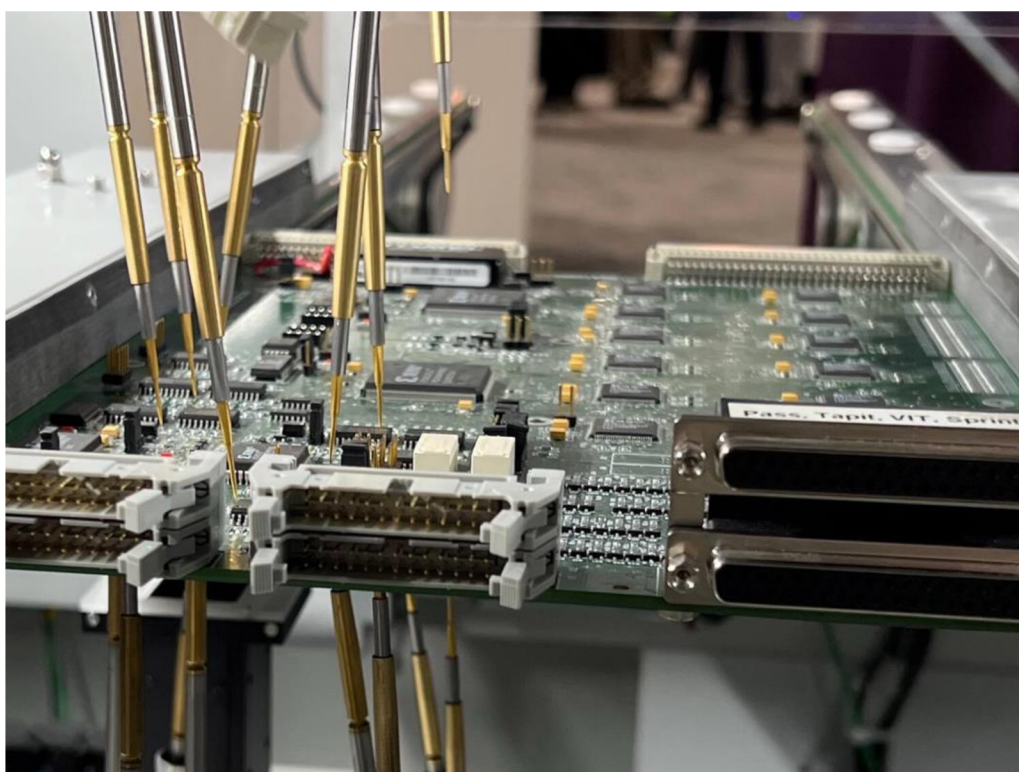
Rozdíly oproti ICT jsou, že funkční testování se provádí na hotových, osazených DPS. Funkční testování ověřuje funkčnost DPS jako celku, ne po jednotlivých částech jako je tomu u ICT. K tomu se využívá simulace skutečných vstupních signálů na konektorech DPS. ICT se zaměřuje na detekci chyb ve výrobě, funkční testování ověřuje i funkčnost programového vybavení. [7]

3. PRŮZKUM KOMERČNĚ DOSTUPNÝCH ŘEŠENÍ PRO TESTOVÁNÍ DPS

Testované moduly mají různé velikosti a různé vstupní a výstupní konektory. Z toho důvodu není příliš vhodné volit „bed of nails“ testovací přípravek s pevně rozmístěnými piny testovacích sond. Vhodnější je použít „flying probe“, tedy přípravek s pohyblivými sondami, nebo přípravek navržený na tuto konkrétní aplikaci. V případě, že by tester měl dostatečný počet testovacích sond, bylo by možné vytvořit i přípravek s pevně rozmístěnými sondami, který by umožnil připojení všech modulů kitu najednou bez nutnosti vyměňovat přípravek pro testování každého typu modulu.

Výrobci bohužel ceny svých testerů neposkytují veřejně, ale pouze v reakci na konkrétní poptávku. Z toho důvodu nelze vytvořit cenový průzkum komerčně dostupných řešení, lze však předpokládat, že ceny se budou pohybovat od několika tisíc dolarů v případě jednoduššího testeru až po statisíce dolarů a více v případě komplexního testeru. Naprostá většina výrobců nabízí možnost připojit k testeru kameru, což umožňuje provedení optické kontroly DPS.

Jedním z výrobců testerů desek plošných spojů je firma *Digitaltest*. Ta má v nabídce několik produktů rozřazených dle objemu výroby i dle rozmanitosti testovaných DPS. Nabízejí jak přípravky s pevně rozmístěnými, tak s pohyblivými piny. Pro funkční testování nabízejí zákaznický konfigurovatelný *Test Tower*, který nabízí velkou modularitu v počtu testovacích karet. Jelikož se jedná o zákaznické řešení na míru, bude cena takového testeru výrazně vyšší. V nižší cenové kategorii se dle výrobce pohybuje tester *Condor*. Ten je vhodný pro testování menšího objemu různorodých DPS, jelikož se jedná o tester s pohyblivými sondami. Nabízí 4 pohyblivé sondy a dále lze přidat až 1024 fixních pinů. Nevýhodou je, že pohyblivé sondy jsou pouze z jedné strany, kdežto fixní sondy jsou z opačné strany DPS. Maximální velikost testované DPS je 505 x 505 mm. *Digitaltest* nabízí pro své testovací zařízení vlastní programové vybavení *CITE* (Computer Integrated Test Environment). [8]



Obrázek 3.1 Příklad testeru s pohyblivými piny „flying probe“ [9]

Další význačnou firmou v oblasti testování je *Acculogic*. Jejich tester *SCORPION FLS980Dxi* má pohyblivé sondy na obou stranách DPS, je možné jich umístit až 11 z každé strany. Přesnost opakovatelnosti polohování sondy je uváděna 10 μm . Výrobce blíže specifikuje také možnosti funkčního testování. Uvádí, že lze zapojit až 4 samostatné zdroje napájení, lze programovat obvody na testované DPS a umožňuje Boundary Scan. Boundary Scan, známý také jako JTAG, je standardem IEEE Std 1149.1. Jedná se o sériové rozhraní s přístupem ke speciálním logickým obvodům umožňujícím testování výstupů, programování paměti nebo ladění integrovaných obvodů [10]. Tester je tak ve spojení s JTAG schopen provádět komplexní funkční testy například ADC (Analog-to-digital converter – analogově digitální převodník) a DAC (Digital-to-analog converter – digitálně analogový převodník), napěťových referencí a jiných částí obvodu. [9]

Firma *Takaya* se zaměřuje pouze na testery s pohyblivými sondami. Nabízí obdobné možnosti testování jako *Acculogic*, oproti ní však vyniká rychlostí testování. V sérii APT-1400F nabízí testery se 6 sondami, umístěnými z jedné strany DPS, v sérii APT-1600FD jsou testery s 10 sondami, které jsou umístěny z obou stran DPS. [10]

Dalšími výrobci testerů DPS schopných provést funkční testování jsou *Seica Automation*, *Keysight* nebo *CheckSum*. Problémem těchto výrobků je jejich cena. Možným řešením by bylo zakoupit pouze testovací přípravek se sondami a ostatní komponenty navrhnout na míru. Testované moduly jsou však navrženy jako vývojové platformy, všechny důležité piny tedy mají vyvedené na konektory. Není tedy potřeba

umísťovat měřicí sondy na jiné části DPS než právě vstupní a výstupní konektory. Pro účely funkčního a obvodového testování je tedy plně dostačující testovací přípravek, do něhož se jednotlivé moduly zapojí a provede se test. Takový přípravek může být jednoduchou deskou plošných spojů s patřičnými konektory.

Dalším důvodem, který přispěl k rozhodnutí navrhnout vlastní tester bylo, že zprovoznění testeru pro funkční testování DPS také nemusí být jednoduchou úlohou. Nejen že je nutné stanovit detailní postup testování, ale také propojit a naprogramovat všechny dílčí systémy jako jsou tester, kamera, napájecí zdroj nebo programátor pro mikrokontrolery. Hlavním faktorem pro rozhodnutí navrhnout vlastní testovací stolicí však byly finanční náklady na pořízení takového zařízení a souvisejících komponent.

4. STANOVENÍ KONCEPCE TESTOVACÍ STOLICE

V zadání práce bylo specifikováno, že má být proveden obvodový test výukových kitů. U většiny modulů však k detekci vady nestačí provést pouze tento typ testu. Je potřeba zjistit, zda všechny části modulu fungují správně. Proto je potřeba provést funkční testování výukových kitů.

U kitu Atmega328P Xplained Mini je nutné provést ověření funkce mikrokontroleru, jeho periférií a obvodu jako celku. Z hlediska testování však není nutné provádět test jednotlivých registrů, pamětí, obvodů časovačů a podobně. Podstatné je ověřit, zda nedošlo k mechanickému nebo elektrickému poškození vstupně-výstupních portů, analogově digitálního převodníku a funkčnosti mikrokontroleru.

Ve výuce jsou k testovacímu kitu ATmega328P Xplained Mini připojeny různé periferie ve formě samostatných modulů. Některé moduly je možné automatizovaně otestovat za pomoci testovacího zařízení. Moduly uživatelského rozhraní však vyžadují akci ze strany uživatele. I tyto moduly by bylo možné otestovat automaticky, vyžadovalo by to však například přidání robotického ramene, které by stlačovalo tlačítka na modulech a otáčelo enkodérem, kameru, která by snímala obrazovku LCD (Liquid crystal display – displej z tekutých krystalů) modulu a další snímače. Tato možnost byla vedoucím práce zamítnuta, jelikož vzhledem k relativně nízkému počtu testovaných modulů je složitost a cena takového řešení neúměrná.

Bylo tedy rozhodnuto, že funkčnost modulů uživatelského rozhraní bude kontrolovat osoba spouštějící testy (dále bude v práci označován za uživatele). Protože testování nebude plně automatické, nestačí pouze vypisovat, zda daný modul funguje správně, nebo je poškozený, ale je nutná bližší interakce s uživatelem. Byla zvážena možnost přidat na testovací zařízení displej a tlačítka pro ovládání, jako lepší varianta se ukázalo vytvořit grafické uživatelské rozhraní (Graphical User Interface – GUI) pro PC, na kterém bude možné konfigurovat testy a zobrazovat výsledky testování jednotlivých modulů.

Testovací zařízení tedy bude sestávat z desky plošných spojů s konektory pro připojení jednotlivých testovaných modulů a řídicího mikrokontroleru, který bude vykonávat jednotlivé testy a komunikovat s počítačem. Nedílnou součástí je také aplikace pro počítač, která bude koordinovat celý proces testování.

5. ROZBOR TESTOVANÝCH ZAŘÍZENÍ

V následujících kapitolách budou stručně popsány jednotlivé testované moduly a jejich funkce. Na základě popisu funkce a možných selhání modulů je stanovena specifikace testů, tedy postup, jak jednotlivé moduly testovat.

5.1 Modul ATmega328P Xplained Mini

Testovací kit ATmega328P Xplained Mini je založen na 8bitovém mikrokontroleru ATmega328P rodiny AVR výrobce Atmel.

Na desce je Mini Embedded Debugger (mEDBG) založený na mikrokontroleru ATmega32U4, který poskytuje přístup ke všem signálům ATmega328p. Debugger umožňuje snadné ladění mikrokontroleru v prostředí Microchip Studio a MPLAB X. Debugger je s MCU propojený prostřednictvím sběrnici UART (Universal Asynchronous Receiver/Transmitter – sériový asynchronní přijímač/vysílač) a SPI (Serial Peripheral Interface – sériové periferní rozhraní). UART slouží jako sériový port pro komunikaci mezi mikrokontrolerem a například terminálem na počítači. Sběrnice SPI je využita pro programování MCU. U mikrokontrolerů AVR je využito rozhraní In-System Programming (ISP), což umožňuje programování přímo v desce plošných spojů. To znamená že mikrokontroler nemusí být vyjmut z DPS, takže lze například aktualizovat firmware (mikroprogramové vybavení). Samotné ladění probíhá prostřednictvím protokolu debugWire, ten také umožňuje programování mikrokontroleru. Debugger dále poskytuje zdroj hodin pro MCU. [12]

Nabízejí se tři způsoby programování testovaného MCU: prostřednictvím rozhraní ISP, pomocí paralelního programování „vysokým napětím“ 12 V (High voltage programming), nebo prostřednictvím debuggeru mEDBG (připojeného k rozhraní debugWire). Programování prostřednictvím mEDBG je primárně určeno pro použití v Atmel Studiu. Dokumentace k protokolu není dostatečná a nejsou ani zveřejněné projekty, kde by někdo použil toto rozhraní k programování mikrokontroleru jinak než z počítače. Paralelní programování je nejnáročnější na implementaci, navíc vyžaduje napětí 12 voltů. Výhodou je, že dokáže přepisovat konfigurační bity (fuse) v každém režimu MCU (například po přepsání bitu SPIEN není možné mikrokontroler dále programovat jiným způsobem než přes paralelní programování). V rámci programování se nepředpokládá, že bude nutné konfigurační bity přepisovat, implementace tohoto rozhraní tak přináší komplikace navíc. Proto bylo rozhodnuto, že lepší variantou bude programovat mikrokontroler ATmega328P skrz rozhraní ISP. Navíc odpadá nutnost připojovat během testování další konektor – v tomto případě USB konektor k desce testovaného kitu.

5.1.1 Mikrokontroler ATmega328P

Rodina mikrokontrolerů AVR, do které ATmega328P spadá, zahrnuje více než 250 zařízení, rozřazených do čtyř podrodin dle výkonosti. Mikrokontrolery tinyAVR jsou zaměřeny na nízkou spotřebu, jsou dodávány v menších pouzdrech a zahrnují jednoduché periferie jako ADC nebo EEPROM (Electrically Erasable Programmable Read-Only Memory – elektricky vymazatelná paměť určená pro čtení). Jsou to také jediné MCU z rodiny AVR, které dokáží pracovat s napájecím napětím od 0,7 V. Určitou nadstavbou je podrodina megaAVR, do níž ATmega328P spadá. Nabízí větší paměti, ISP programování flash paměti a rozmanitější periferie jako USB, CAN a podobně. Jak tinyAVR, tak megaAVR jsou 8bitové mikrokontrolery. Atmel AVR XMEGA jsou 8 a 16bitové MCU, které nabízejí vysoký výpočetní výkon, schopnost pracovat v reálném čase, šifrovací algoritmy, DMA řadič a efektivní řízení spotřeby. Nejvyšší řadou jsou 32bitové mikrokontrolery Atmel AVR UC3. Ty oproti předchozím MCU nabízí digitální signálový procesor, matematický koprocesor FPU, vyšší taktovací frekvenci a další. Mikrokontrolery AVR také nabízejí hardwarové funkce umožňující splnění bezpečnostních standardů. [13] [14]

Samotný ATmega328P obsahuje 32 KB flash paměti, 2 KB SRAM a 1 KB EEPROM paměti. Je dodáván ve 3 pouzdrech, v DIP (Dual Inline Package – dvouřadé pouzdro) se 28 piny, nebo ve 2 různých SMD (Surface Mount Device – součástka pro povrchovou montáž) pouzdrech se 32 piny. Ve všech pouzdrech poskytuje 23 vstupně výstupních pinů. Dále MCU poskytuje na vybraných pinech externí přerušování, výstupy PWM (Pulse Wide Modulation – pulzně šířková modulace), 8 kanálový, 10bitový analogově digitální převodník a analogový komparátor. Z hlediska sběrnic poskytuje UART, SPI a I2C kompatibilní sběrnici nazývanou z důvodu ochranné známky TWI (Two Wire Interface – dvou vodičová sériová sběrnice). Mikrokontroler je možné napájet 2,7 V až 5,5 V a maximální taktovací frekvence je 20 MHz. [15]

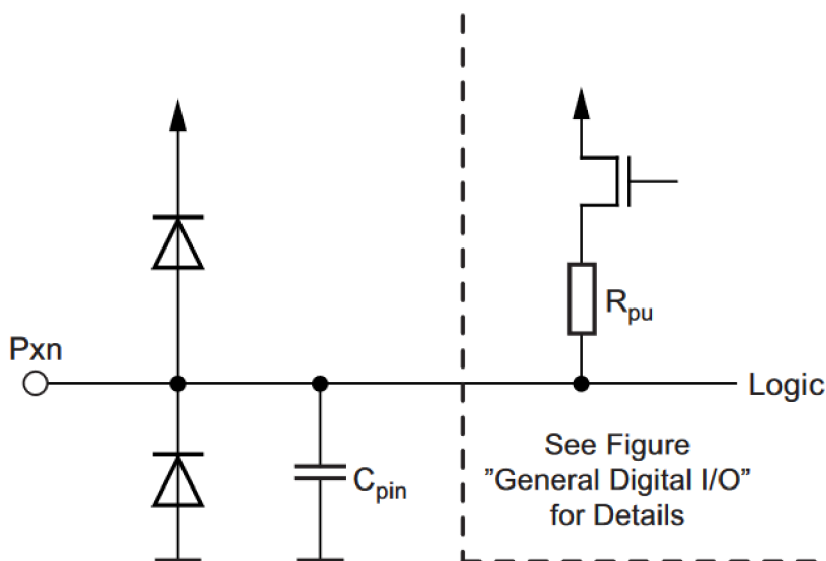
Oproti obyčejnému ATmega328 je verze „P“ (picoPower) zaměřena na optimalizaci spotřeby. Dále existuje „PB“ verze, která však již není s původním MCU kompatibilní. Přidává 4 vstupně výstupní piny, přídatné moduly pro sběrnice UART, SPI a TWI, kapacitní dotykové rozhraní a další vylepšení. [16]

5.1.2 Digitální vstupně-výstupní porty

Náhradní schéma digitálního vstupně-výstupního (DIO) portu na ATmega328P je na obrázku 5.1. Každý pin má vlastní nastavitelný, pull-up rezistor a přepětíové diody. DIO piny je důležité otestovat, neboť mohou být poškozeny několika způsoby.

Pokud je na pin přivedeno vyšší napětí než součet napájecího napětí a prahového napětí přepětíové diody, začne diodou procházet proud. Tato dioda však slouží pouze pro pokrytí krátkých impulzů, například při ESD výboji. Pokud přepětíovou diodou prochází větší proud po delší dobu, může dojít k jejímu proražení a zničení pinu, nebo celého

mikrokontroleru. Ke stejné situaci může dojít v případě přiložení napětí opačné polarity na pin mikrokontroleru.



Obrázek 5.1 Náhradní schéma vstupně-výstupního pinu [15]

Další možností, jak může dojít k poškození výstupního pinu, je zkrat mezi dvěma piny nebo zkrat vůči napájecímu napětí. Zkrat může způsobit krátkodobou nefunkčnost pinů do doby odstranění zkratu, poškození však může být i trvalé. Pokud je pin nastaven jako výstupní a nastavená logická hodnota na pinu je opačná, než je zkratová hodnota pinu, může dojít k trvalému poškození v důsledku zničení výstupního budiče.

K poškození výstupních pinů také může dojít, pokud je překročen maximální proud pro jeden port (skupina pinů, v tomto případě 8) mikrokontroleru. U ATmega328P je maximální proud jedním portem 200 mA.

Během testování je postupně vyzkoušena funkčnost všech DIO pinů mikrokontroleru. Postup testování každého pinu je dán tabulkou 5.1.

Tabulka 5.1 Postup testování DIO

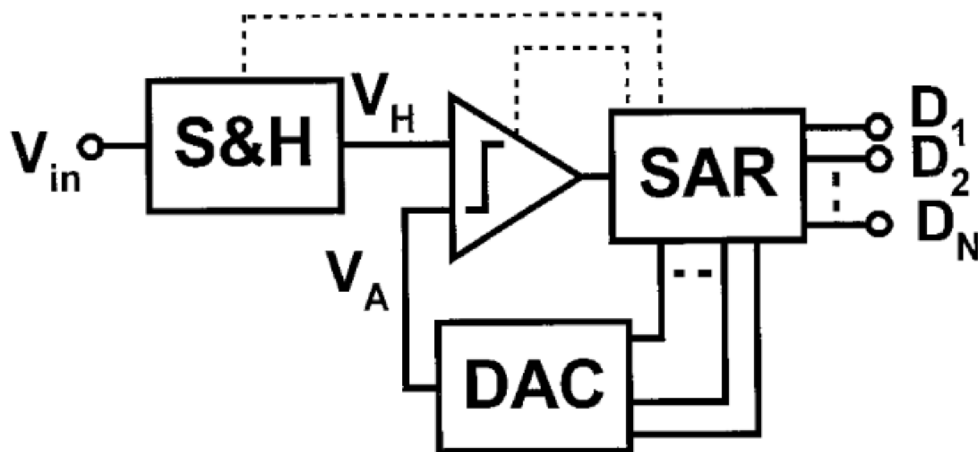
Pin ATmega328P	Pin řídicího MCU	Očekávaná hodnota
Přečti digitální vstup	Nastav logickou 1	Log. 1
Přečti digitální vstup	Nastav logickou 0	Log. 0
Nastav logickou 1	Přečti digitální vstup	Log. 1
Nastav logickou 0	Přečti digitální vstup	Log. 0

Řídicí MCU nastaví na daném pinu logickou úroveň 1 nebo 0 a odešle požadavek na přečtení digitálního vstupu. ATmega328P přečte logickou hodnotu pinu, odešle ji jako

odpověď na prvotní zprávu a řídicí MCU zkontroluje, zda se přečtená hodnota shoduje s očekávanou. Ve druhé části testu se role zařízení prohodí. Po odeslání požadavku ATmega328P nastaví pin na výstup s danou logickou hodnotou a řídicí MCU čte skutečnou hodnotu pinu. Výsledky jsou v řídicím MCU vyhodnoceny a případně je přes sériový port vypsaná do počítače chyba na daném pinu.

5.1.3 Analogově digitální převodník

Dalším rozhraním, které je vhodné otestovat je analogově digitální převodník (ADC). Analogová hodnota je převáděna pomocí postupné aproximace. Převodník je připojen k analogovému multiplexoru, který umožňuje měřit analogovou hodnotu na 8 pinech. Dále je možné vybrat jako vstup multiplexoru GND, interní referenční napětí 1,1 V nebo teplotní senzor umístěný na čipu. Jako napěťovou referenci převodníku lze zvolit vstup AVCC, tedy pin připojený k napájecímu napětí, nebo vstup AREF, ke kterému lze připojit libovolnou napěťovou referenci v rozsahu napájecího napětí. Lze též využít interní napěťovou referenci 1,1 V. Rychlost převodu lze upravit děličkou frekvence, nastavitelný kmitočet je od 50 kHz do 200 kHz. Pokud není nutné plné rozlišení 10 bitů, lze nastavit i vyšší frekvenci. Jeden převod zpravidla trvá třináct hodinových taktů.



Obrázek 5.2 Blokové schéma analogově digitálního převodníku s postupnou aproximací [17]

Samotný převodník se skládá ze vzorkovače, komparátoru, převodníku z digitální na analogový signál (DAC) a vnitřní logiky. Blokové schéma je znázorněno na obrázku 5.2 (S&H je zde vzorkovač, SAR je vnitřní logika převodníku). Vzorkovač zajišťuje zachycení vstupního signálu a udržení konstantní hodnoty po celou dobu převodu. Komparátor pak porovnává zachycenou analogovou hodnotu s výstupem DAC převodníku. Převod začíná nastavením nejvýznamnějšího bitu binárního čísla na DAC převodníku. Komparátor porovná hodnotu na DAC převodníku s analogovou hodnotou

zachycenou vzorkovačem. Podle znaménka na výstupu komparátoru se bit ponechá v logické 1 nebo 0. Vnitřní logika se pak stará o postupné nastavování nižších bitů binárního čísla na DAC převodníku a zápisu výsledného čísla. Po dokončení převodu je možné vyvolat přerušení. [17]

Funkčnost převodníku je testována na všech 8 kanálech (pínech) a dále na vnitřních napěťových referencích. Aby mohla být otestována korektní funkčnost převodníku, byly k pinům mikrokontroleru připojeny napěťové děliče, které slouží jako testovací napěťová hladina mezi GND a plným napájecím napětím VCC. Celkem je tedy každý kanál převodníku testován na 3 napěťových úrovních – GND, VCC a polovina napájecího napětí VCC z výstupu napěťového děliče.

Tabulka 5.2 Postup testování ADC

Pin ATmega328P	Pin řídicího MCU	Očekávaná výstupní hodnota ADC
Změř analogovou hodnotu	Stav vysoké impedance	512 (Vcc/2)
Změř analogovou hodnotu	Nastav logickou 1	1023 (Vcc)
Změř analogovou hodnotu	Nastav logickou 0	0 (GND)

Postup testování každého kanálu je dán tabulkou 5.2. Před testováním jednotlivých kanálů převodníku se zkontroluje korektní funkčnost samotného převodníku na dvou vnitřních napěťových referencích. Následně se testují jednotlivé kanály. Prvním krokem je nastavení testovaného pinu na straně řídicího MCU do stavu vysoké impedance. Tím se zajistí, že na pinu bude hodnota napětí z odporového děliče. Následně ATmega328P převede analogovou hodnotu na digitální a uloží výsledek. Očekávaný výsledek je $(2^n - 1)/2 = (2^{10} - 1)/2 \doteq 512$, kde n je počet bitů převodníku, v tomto případě 10. Dalšími kroky je postupné nastavení logické 1 a logické 0 na straně řídicího MCU a převod analogových hodnot na digitální. V každém kroku se porovnává, zda je převedená hodnota v očekávaném rozsahu. Pokud není, je zjištěna na daném kanálu převodníku chyba. Řídicí mikrokontroler koordinuje celý proces testování ADC a při zjištění chyby odesílá chybové hlášení přes sériový port do řídicí aplikace na počítači.

5.2 Relé modul

Relé modul obsahuje dvě relé Songle SRD-05VDC-SL-C. Relé jsou buzeny tranzistory, ty jsou pak od zbytku obvodu odděleny optočleny. Díky budícím tranzistorům je možné relé spínat přímo z mikroprocesoru. Spínací proud je 15-20 mA [18]. Přivedení napětí na vstup modulu je indikováno LED diodou. Relé jsou elektromechanického typu, z čehož pramení, že mají omezenou životnost kvůli mechanickému opotřebení. Relé je také možné poškodit, pokud je spínáno vyšší frekvencí, než je maximální frekvence uváděná výrobcem. [19]

Relé modul je testován následujícím způsobem:

1. Změř výstup relé v klidovém stavu
2. Sepni první relé a změř výstup
3. Sepni druhé relé a změř výstup
4. Ukonči test

Tento postup zaručí otestování všech kontaktů relé, každé ze dvou relé na modulu je pak testováno samostatně. Při testování jsou měřeny oba výstupy, aby byl detekován případný zkrat na modulu.

5.3 Modul hodin reálného času a EEPROM paměť

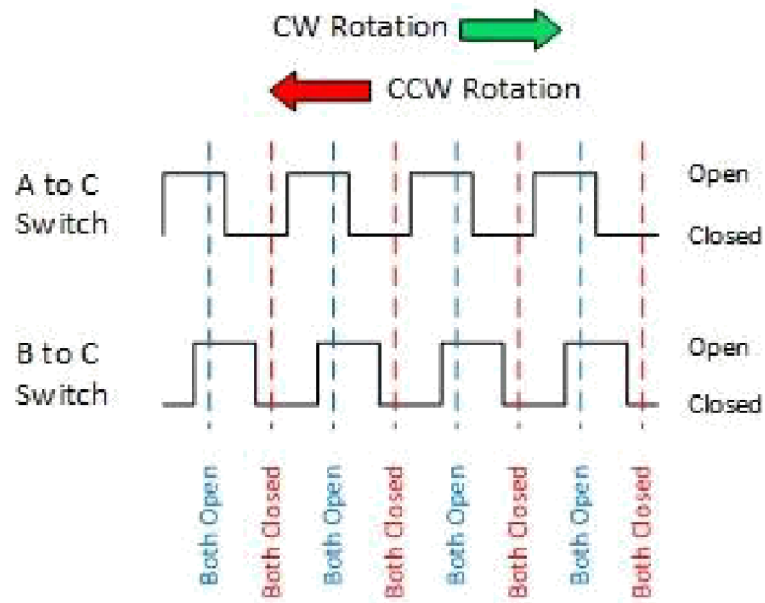
Modul obsahuje na jedné desce hodiny reálného času DS3231 a EEPROM paměť AT24C32. Společnou sběrnici modulu je I2C, hodiny reálného času (RTC) mají výstup obdélníkového signálu konfigurovatelné frekvence a výstup pevné frekvence 32 kHz. Hodiny využívají teplotně kompenzovaný krystalový oscilátor. Čas je uchováván v sekundách, minutách, hodinách, dnech, aktuálním měsíci a roce. [20]

Paměť EEPROM má 32 768 bitů, rozdělených po 4096 slovech, každé délky 8 bitů. Stejně jako u hodin reálného času je maximální frekvence I2C sběrnice 400 kHz. Adresa obvodu AT24C32 je částečně konfigurovatelná pomocí vstupů A0 – A2, na modulu je možná změna adresy propojením pájivých plošek. [21]

Funkcionalita modulu je ověřena funkčním testováním. Modul může být poškozen například v důsledku zkratu, paměť EEPROM má omezenou životnost danou počtem zápisů do paměti, poté se mohou vyskytovat vadné paměťové bity. Korektní taktovací frekvence modulu reálného času je testována spuštěním výstupu obdélníkového signálu a měřením frekvence testovací stolicí. Pokud je naměřená frekvence v pásmu tolerance, funguje oscilátor modulu správně. Dále jsou testovány registry modulu hodin. Do registrů jsou zapsány zkušební hodnoty a ty jsou zpětně přečteny. Do EEPROM paměti jsou zapsány dva testovací vzory, nejprve logické jedničky, přečten skutečný obsah paměti, poté jsou zapsány logické nuly a opět přečten skutečný obsah paměti, případně detekovány chyby.

5.4 Inkrementální rotační enkodér

Enkodér KY-040 dává na svém výstupu informaci o relativním počtu otáček ve formě dvoubitového Grayova kódu. Uvnitř enkodéru jsou 2 fázově posunuté spínače, pro zjištění relativní pozice a směru natočení jsou tedy potřeba 2 piny. Na otočení o 360° připadá 20 pozic enkodéru. V každé pozici jsou oba spínače buď otevřené, nebo zavřené. [22]



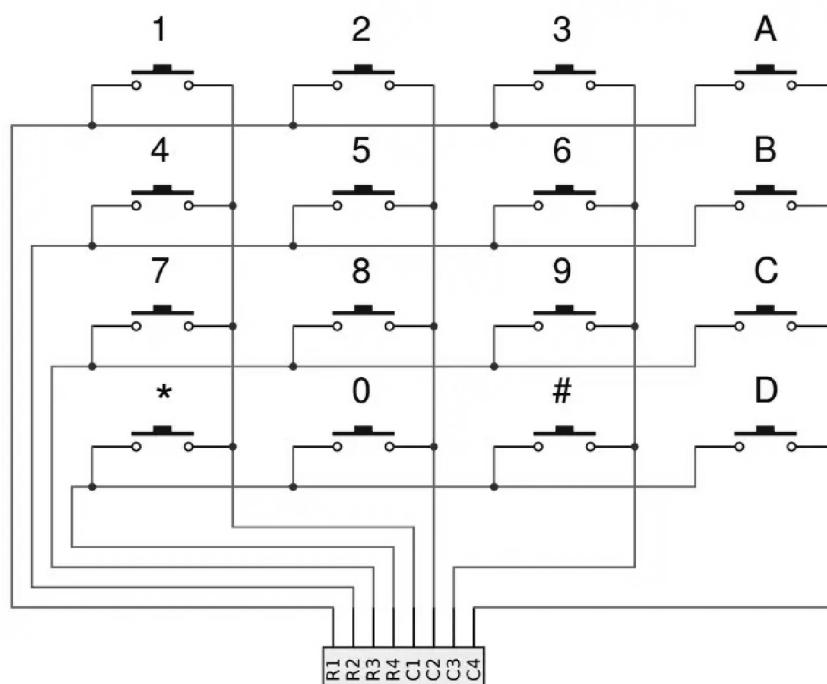
Obrázek 5.3 Výstup enkodéru při otáčení po (CW) a proti (CCW) směru hodinových ručiček [22]

Jelikož spínače uvnitř enkodéru jsou mechanické, je potřeba ošetřit zákmity. Z toho důvodu je vhodné využít pro detekci otočení stavový automat. Enkodér dále obsahuje tlačítko v ose enkodéru. Označení svorek modulu je zavádějící, pin CLK je signál A z enkodéru, pin DT je signál B (viz. obrázek 5.3). Oba jsou připojeny k pull-up rezistorům (rezistor připojený ke kladné svorce napájení). Na pinu SW je zapojeno tlačítko bez pull-up rezistoru.

Mechanické kontakty podléhají opotřebení, je tedy nutné ověřit, že správně spínají. Během testování enkodéru je uživatel vyzván k otočení enkodéru oběma směry a stisknutí tlačítka. Program detekuje, zda enkodér prošel všemi vyhovujícími stavy a stisknutí tlačítka.

5.5 Maticová klávesnice

Zapojení klávesnice do maticové topologie si klade za cíl minimalizovat počet vstupů a výstupů na daný počet kláves. V případě použité klávesnice 4x4 je počet kláves 16, ale pro detekci stisknutí klávesy stačí 8 pinů. Jak je zřejmé z obrázku 5.4, jeden kontakt tlačítka je vždy připojen ke společnému řádku a druhý kontakt ke společnému sloupci.



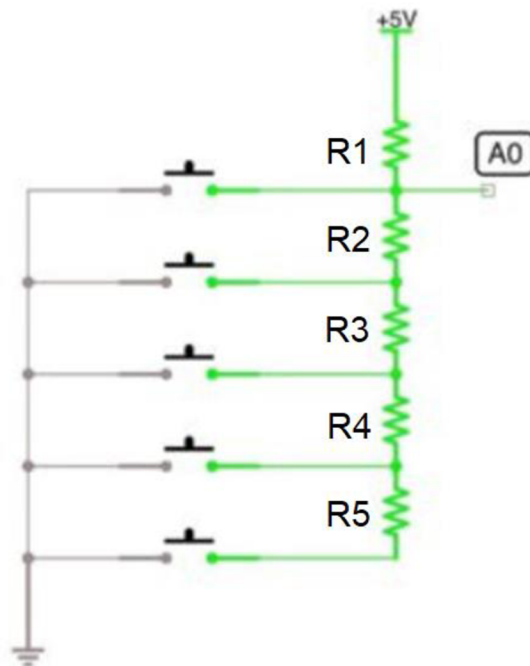
Obrázek 5.4 Vnitřní zapojení maticové klávesnice [23]

Pro vyčtení stisknuté klávesy lze využít algoritmus „postupující nuly“. Nejprve jsou všechny piny klávesnice nastaveny na logickou úroveň 1 přes pull-up rezistory. Poté se nastaví první řádek na logickou úroveň 0 a detekuje se, zda na některém sloupci došlo ke změně na logickou 0. Pokud ano, na prvním řádku a odpovídajícím sloupci bylo stisknuto tlačítko. Poté se první řádek nastaví opět na výchozí hodnotu a pokračuje se se zbývajících řádky.

Stejně jako v případě enkodéru mají mechanické kontakty tlačítek omezenou životnost. Je tedy nutné detekovat, zda všechna tlačítka umožňují spojení a rozepnutí obvodu. Pro testování maticové klávesnice je nutná interakce s uživatelem. Uživatel je během testování vyzván k posupnému stisknutí všech kláves a je detekováno, zda jsou všechny tlačítka klávesnice funkční.

5.6 Modul LCD displeje

Použitý modul displeje je ve formě zásuvného modulu (shield) kompatibilního s konektory Arduino UNO. Kromě displeje je na modulu také šest tlačítek. Pět tlačítek je zapojeno ve formě odporového děliče (viz. obrázek) a připojeno na analogový výstup. Pokud je modul zasunut do Arduino UNO kompatibilní desky, šesté tlačítko může sloužit pro reset mikroprocesoru.



Obrázek 5.5 Zapojení tlačítek na modulu displeje [24]

Samotný LCD displej 1602A, výrobce Shenzhen Eone Electronics, má dva řádky, každý o 16 znacích. Jeden znak displeje se skládá z 5x8 pixelů. Displej je podsvícen bílou LED diodou, jejíž jas lze měnit nastavením potenciometru nebo pomocí PWM (pulzně šířkovou modulací). [25] Displej je řízen pomocí řadiče HD44780U. Řadič má paměť až pro 80 znaků, funkce pro práci s kurzorem, znakový generátor a další funkce. Řadič komunikuje s mikroprocesorem pomocí paralelního rozhraní, sběrnice používá 4 nebo 8 datových pinů, dále pin RW, určující směr komunikace (zápis nebo čtení), a pin RS (Register Select), který přepíná zápis dat mezi řídicím registrem a pamětí displeje. [26]

Displej je testován v režimu 4bitové komunikace, jelikož je takto na zásuvném modulu zapojen. Testovací sekvence je následující:

1. Zapnutí podsvícení displeje, rozsvícení všech pixelů displeje
2. Uživatel potvrdí, že všechny pixely jsou funkční
3. Zobrazení posloupnosti znaků na displeji
4. Uživatel potvrdí korektní zobrazení znaků na displeji
5. Uživatel je vyzván k postupnému stisknutí všech tlačítek na displeji
6. Detekce správných napěťových úrovní na analogovém výstupu modulu
7. Po detekci stisknutí všech tlačítek je test ukončen, případně jej uživatel může předčasně ukončit

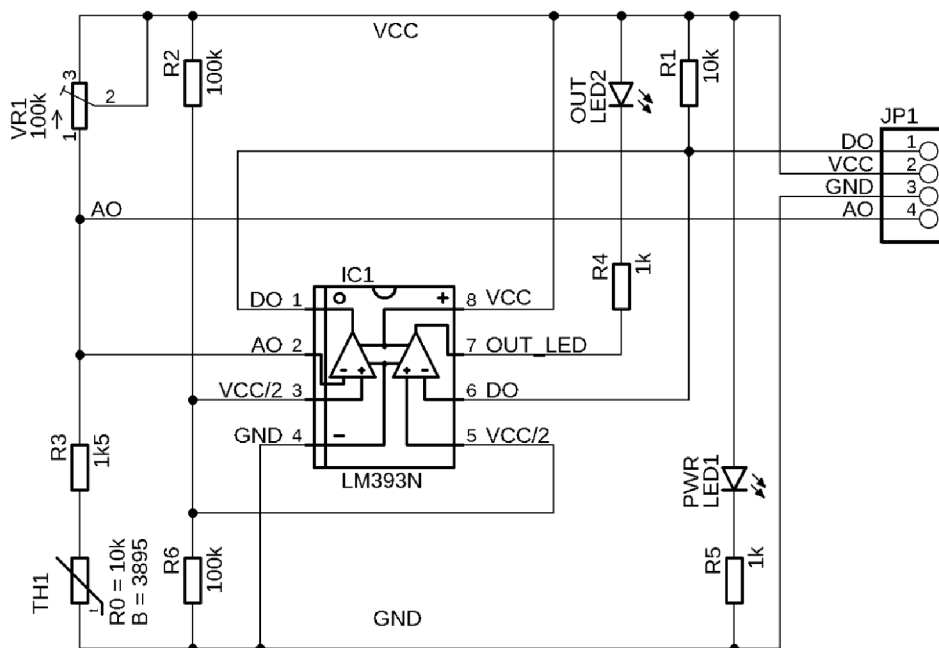
5.7 Reproduktor

Reproduktor nemá dostupný katalogový list. Známé parametry jsou impedance 8Ω a výkon 0.5 W . Jedná se o elektrodynamický typ reproduktoru, skládá se tedy z pohybující se cívky umístěné v magnetickém poli permanentního magnetu.

Funkce reproduktoru je testována interakcí s uživatelem. Testování zahrnuje pouze funkční test reproduktoru, nejsou sledovány jeho parametry ani charakteristika. Reproduktor je během testu spuštěn a uživatel potvrdí, že reproduktor generuje zvuk požadované frekvence a úměrné hlasitosti.

5.8 Modul s termistorem

Modul termistoru obsahuje na desce kromě samotného termistoru také komparátor s LED diodou a digitálním výstupem DO. Pokud je napětí na analogovém výstupu AO nižší než polovina napájecího napětí VCC, je sepnut digitální výstup a rozsvícena LED dioda LED2. Citlivost je nastavována trimrem VR1. Modul neobsahuje vlastní regulátor napětí, výstupní hodnota na analogovém výstupu AO tedy záleží na napájecím napětí modulu. Dokumentace k modulu není dostupná, ale v předmětu MPC-POR, kde je modul využit ve výuce, je schéma zapojení získané pomocí reverzního inženýrství.



Obrázek 5.6 Vnitřní zapojení modulu s termistorem [27]

Analýzou schématu zapojení lze dojít k závěru, že operační zesilovač LM393N není žádným způsobem chráněn proti připojení napětí opačné polarity. Modul lze takto jednoduše poškodit, a proto musí být funkčnost daného modulu testována.

Jak bylo uvedeno, citlivost termistoru je nastavována trimrem. Výstupní napětí modulu tedy může být (dle nastavení trimru) jakákoliv hodnota mezi cca 0.5 V až hodnota napájecího napětí. Je předpokládáno, že různé moduly budou mít trimry nastavené různě. Ruční nastavení trimru je časově náročné, protože trimr má rozsah 25 otáček. Tyto důvody vedly k rozhodnutí, že funkce modulu bude vyhodnocena pouze okometricky. Testovací stolice bude číst výstupní hodnoty modulu a zobrazovat je přes sériovou linku na počítač, ze kterého bude test spuštěn. Vyhodnocení, zda modul funguje korektně, bude na uživateli.

6. NÁVRH OBVODOVÉHO ŘEŠENÍ

V této kapitole je popsán postup návrhu jednotlivých částí obvodu. Pro návrh schématu a desky plošných spojů byl použit návrhový systém KiCad 7.0. KiCad je hojně rozšířený open source nástroj, který má za sebou dlouhý vývoj a řadí se mezi nejlepší dostupné řešení na trhu. Jeho výhodou je také široká komunita uživatelů a dostupnost knihoven.

6.1 Výběr řídicího mikrokontroleru

V kapitole 5 byly popsány způsoby testování jednotlivých modulů. Díky tomu lze sepsat seznam požadavků na mikrokontroler spouštějící testy. V tabulce 6.1 je seznam všech vstupů a výstupů, které je potřeba k mikrokontroleru připojit.

Tabulka 6.1 Seznam vstupů a výstupů všech modulů

Název modulu	Počet digitálních vstupů/výstupů	Počet analogových vstupů/výstupů	Sběrnice
ATmega328P	14		SPI
LCD modul	8	1	
Relé modul	6		
Enkodér	3		
RTC modul	1		I2C
Reproduktor	1		
Modul termistoru	1	1	
Maticová klávesnice	8		
Komunikace s PC			UART
Celkem	42	2	SPI, I2C, UART

6.1.1 Rešerše vhodných mikrokontrolerů

Testovaný kit pracuje s 5 V logikou. Použití mikrokontroleru s 3.3 V logikou by vedlo k nutnosti přidat ke každému testovanému modulu převodníky logické úrovně. Vzhledem k tomu, že jsou dostupné i mikrokontrolery pracující s 5 V logikou, je zbytečné takto komplikovat návrh DPS. Z tohoto důvodu byly z výběru řídicího MCU vyřazeny mikrokontrolery STM32 výrobce STMicroelectronics nebo MSP430 výrobce Texas Instruments.

V úvahu přišlo použití mikrokontroleru rodiny Kinetis E výrobce NXP. Tyto mikrokontrolery jsou založeny na 32bitovém jádře Arm Cortex-M0+/M4, obsahují sériové sběrnice UART, I2C, SPI, 12bitové ADC převodníky, časovače, a jsou vyráběny v různých pouzdrech s velkým rozsahem počtu vstupně výstupních pinů.

Dalšími možnostmi byly 8bitové mikrokontrolery PIC a AVR výrobce Microchip. Tyto mikrokontrolery jsou dostupné v široké škále pouzder, není tedy problém mezi nimi

vybrat mikrokontroler s potřebným počtem vstupů a výstupů. Splňují také požadavek na dostupnost sériových sběrnic a velikost paměti.

Řídicí mikrokontroler byl nakonec vybrán dle dostupnosti u výrobce DPS, který desku zároveň osadil součástkami. Jako řídicí MCU byl zvolen ATmega2561. Výhodou tohoto MCU je jeho rozšířenost (většina periférií je obdobných s ATmega2560, který je použitý na Arduino Mega) a tedy i dobrá dokumentace a podpora. Další výhodou je zachování jednotného programovací prostředí s testovaným ATmega328P. Jedinou nevýhodou pro danou aplikaci je nepatrně vyšší cena, testovací přípravek však bude pravděpodobně vyroben jen v jednotkách kusů, a proto je tento faktor zanedbán.

6.1.2 Programování a komunikace s testovaným kitem ATmega328P

Jak bylo zmíněno v kapitole 5.1, k programování kitu ATmega328P Xplained Mini je použito rozhraní ISP. ATmega328P umožňuje jak paralelní, tak sériové programování, mnohem rozšířenější je sériové programování. K tomu se využívá sběrnice SPI. Pro programování je dále potřeba ovládat reset pin mikrokontroleru.

Během testování kitu je potřeba komunikovat mezi oběma mikrokontrolery. K tomu lze využít jak sběrnici SPI, tak UART. Jelikož řídicí mikrokontroler ATmega2561 má dva samostatné UART moduly, lze pro komunikaci využít jen sběrnice UART.

6.1.3 Spouštění testů a signalizace výsledků

Konfiguraci a spouštění testů řídí aplikace na PC. Testovací stolice s PC komunikuje přes sběrnici USB. ATmega2561 nelze přímo připojit k USB, je potřeba převodník na UART. Výsledky jednotlivých testů jsou odesílány do PC opět přes USB sběrnici.

6.2 Zapojení řídicího mikrokontroleru

U mikrokontroleru ATmega2561 je na výběr několik zdrojů hodinových impulzů. Lze využít zabudovaný 128 kHz oscilátor, ten je však určen především pro módy nízké spotřeby. Druhým zabudovaným oscilátorem je 8 MHz RC oscilátor. Tento oscilátor je závislý na napájecím napětí a teplotě, a ačkoliv lze kalibrovat, neposkytuje přesný zdroj frekvence. Dalšími možnostmi je využít externí zdroj hodinových impulzů (tento mód je využít u testovaného kitu, kde debugger poskytuje hodinový impuls pro ATmega328P) nebo použít krystalový oscilátor. Jako zdroj hodin byl vybrán právě krystalový oscilátor, protože poskytuje vysokou přesnost a stabilitu kmitočtu.

Protože testovaný kit má 16 MHz oscilátor, byl i pro ATmega2561 vybrán krystalový oscilátor o frekvenci 16 MHz. Z oscilátorů dostupných na JLCPCB byl vybrán YSX321SL firmy YSC. Krystalový oscilátor je připojen mezi piny XTAL1 a XTAL2 mikrokontroleru a oba piny jsou připojeny přes zátěžovou kapacitu (load capacity) ke GND. Zátěžová kapacita je tvořena kondenzátory, hodnotu kapacit obvykle udává výrobce oscilátoru. V katalogovém listě, který je obecný pro celou sérii oscilátorů, však konkrétní hodnoty kapacit kondenzátorů uvedeny nejsou. [28] Katalogový list

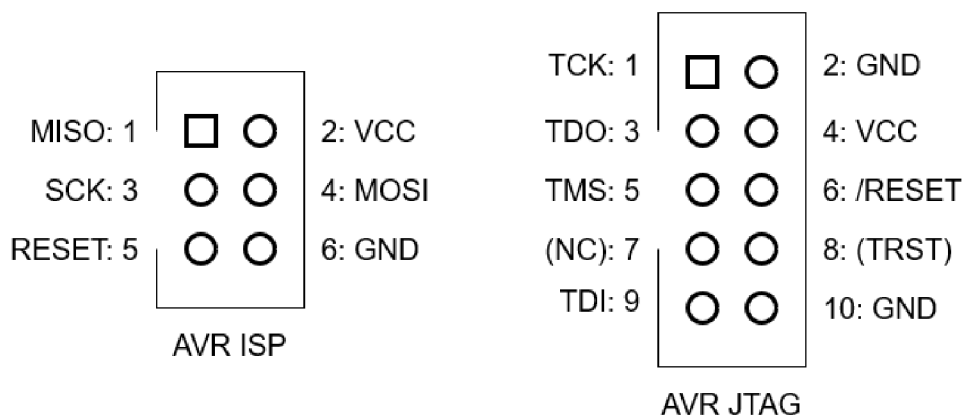
ATmega2561 udává doporučený rozsah 12 až 22 pF. Byly zvoleny kondenzátory s kapacitou 22 pF, s přihlédnutím na jiné DPS, kde jsou mikrokontrolery ATmega použity. Přesnost frekvence oscilátoru je u navrhované DPS důležitá především u UART rozhraní, to však dokáže pracovat i pokud se frekvence vysílače a přijímače liší až o 2 %. Volba kapacity zátěžových kondenzátorů má v tomto případě zanedbatelný vliv. [29]

V některých zapojení se k oscilátoru připojuje rezistor 1 M Ω . Výrobce se o jeho použití nezmiňuje, do DPS byly přidány plochy pro jeho případné umístění, osazen však není.

Reset pin mikrokontroleru je aktivní v logické 0. Pin obsahuje vnitřní pull-up rezistor. Doporučuje se však připojit i externí pull-up rezistor pro ochranu před spontánními restarty v rušivém prostředí. Pokud není použito DebugWIRE nebo PDI rozhraní, je vhodné připojit také kondenzátor 100 nF mezi reset pin a GND. Pro možnost ručního restartu MCU je připojeno tlačítko SW1 k pinu reset.

Mikrokontroler odebírá pouze malý proud. Odběr však není kontinuální, ale obsahuje krátké impulzy při každé hraně hodinového impulzu a při přepínání vstupně výstupních portů. Napájecí vodiče mikrokontroleru pak svými parazitními vlastnostmi vytvářejí anténu, která produkuje rušení. Z toho důvodu je vhodné umístit k napájecím pinům blokovací kondenzátory. Nejlepšího potlačení šíření rušení od mikrokontroleru lze docílit umístěním kondenzátorů co nejbližší k pinům mikrokontroleru. V navrženém obvodu jsou použity blokovací kondenzátory o kapacitě 100 nF. V souladu s doporučením výrobce je blokovací kondenzátor umístěn také na pin analogové reference AREF. [30]

Mikrokontroler ATmega2561 je možno programovat několika způsoby. Nejběžnějším z nich je sériové ISP programování pomocí 3 pinů, SCK, PDI a PDO. Na některých mikrokontrolerech je tato sběrnice shodná se sběrníci SPI, na ATmega2561 je společný pin SCK, ale PDI a PIO jsou na pinech UART. Tyto piny budou využity během testování k jiným účelům, a proto bylo přidáno i rozhraní JTAG (Join Test Action Group, standard IEEE 1149.1). Mikrokontroler umožňuje i programování přes JTAG. K programování přes sériové rozhraní popsané výše však stačí Arduino, do kterého je nahrán program ArduinoISP a odpadá tak použití dalšího externího programátoru. [31] Proto byly do desky plošných spojů umístěny obě rozhraní. Mezi mikrokontroler a konektor AVR-ISP byly umístěny rezistory o hodnotě 330 Ω . Výrobce doporučuje umístění rezistorů v případě, že jsou piny propojeny s dalšími periferiemi MCU. V tomto případě jsou piny dále propojeny s převodníkem na USB sběrnici. [30]



Obrázek 6.1 Zapojení konektorů ISP (vlevo) a JTAG (vpravo)

6.3 Návrh napájení

Pro návrh napájení bylo nutno spočítat maximální možný proud. K většině modulů není dostupná žádná dokumentace, jelikož jsou vyráběny různými asijskými výrobci a ani prodejci neudávají dokumentaci výrobku. Proto musel být každý modul zapojen, spuštěn a změřen jeho proudový odběr. Naměřené hodnoty jsou v tabulce. Jelikož maticová klávesnice a rotační enkodér jsou pouze pasivními prvky, nebyly do tabulky zahrnuty.

Tabulka 6.2 Seznam maximálního proudového odběru všech komponent

Komponenta	Maximální proud [mA]
ATmega2561	14
ATmega328P Xplained Mini	14
LCD modul	30
Relé modul	135
Modul hodin reálného času	8,5
Modul s termistorem	7,2
Reproduktor	7
USB převodník	20
Celkový proud	221,7

K odběru řídicího ATmega2561 je nutné připočítat proud do výstupních pinů mikrokontroleru. Výstupní piny jsou připojeny buď k pull-up rezistorům, nebo k testovaným modulům. Celkový proud se tedy nebude výrazně lišit. USB 2.0 je schopno dodat až 500 mA, bude tedy dostačující pro napájení celé desky plošných spojů se všemi moduly.

Jako nadproudová ochrana je použita vratná PPTC pojistka. Pokud pojistkou prochází nižší než maximální proud, má pojistka malý odpor. Pokud je maximální proud

překročen, pojistka se samoohřevem postupně zahřívá až po dosažení prahové teploty, kdy dojde k řádovému zvýšení odporu. Po přerušení obvodu se pojistka ochladí a opět dojde ke snížení odporu na původní hodnotu, takže není potřeba pojistku vyměňovat. Pro filtraci vstupního napětí je použit kondenzátor 4,7 μF . Použití kondenzátoru o větší kapacitě není doporučováno, jelikož celková maximální kapacita USB sběrnice by měla být do 10 μF [32]. Pro signalizaci napájení obvodu je využita LED dioda D1.

V zapojení byl použit konektor USB C. Aby mohlo být zařízení napájeno z USB C konektoru, musí být v režimu UFP (Upstream Facing Port), což znamená, že zařízení je hostem na USB sběrnici. Minimální konfigurace pro povolení UFP jsou pull-up rezistory na pinech CC1 a CC2 a spojení s USB 2.0 zařízením. [33]

6.4 Převodník na USB sběrnici

Použitý mikrokontroler není možné přímo připojit na USB sběrnici, je nutné přidat převodník z UART na USB. Rozšířenými převodníky jsou FT232RL, CP2102 nebo CH340. Z dostupných možností byl zvolen CH340. Oproti zbývajícím jmenovaným převodníkům vyžaduje (na některých operačních systémech) instalaci ovladače, je však hojně rozšířený a vyniká cenovou dostupností.

Piny D+ a D- jsou opatřeny TVS diodami (Transient Voltage Suppression – dioda pro potlačení přechodného napětí) pro ochranu proti elektrostatickému výboji. Pro účely ochrany USB zařízení existují specializované obvody, které v jednom pouzdře obsahují jak TVS diody pro potlačení elektrostatického výboje, tak diodu připojenou mezi napájecí svorky, která slouží jako ochrana proti přepětí. Pro ochranu USB převodníku tak byl vybrán právě takový obvod od výrobce ProTek Devices. Elektromagnetické rušení mezi deskou a stíněním USB kabelu je potlačeno feritovým jádrem připojeným mezi pinem stínění USB (pin SHIELD) a GND. Feritové jádro má hodnotu odporu 0,15 Ω , při frekvenci 100 MHz je hodnota impedance 100 Ω . [34]

6.5 Zapojení modulů

Všechny použité moduly pracují s napájecím napětím 5V, nebylo tedy nutné přidávat další napájecí zdroje. Většina modulů je vnitřně zapojena tak, aby vyžadovala co nejmenší množství externích součástek a bylo je možno přímo připojit k mikrokontroleru nebo k jinému nadřazenému systému. Přesto muselo být zapojení některých modulů rozšířeno o další součástky.

6.5.1 Zapojení ATmega328P Xplained Mini

Ke kitu ATmega328P Xplained Mini byly na piny ADC převodníku připojeny odporové děliče tvořené z rezistorů o hodnotě 1 k Ω . Ty vytvářejí jedno z referenčních napětí pro testování ADC převodníku. Převodník je testován na třech referenčních úrovních: 0 V, 5 V a 2,5 V z odporového děliče. Všechny vstupně výstupní piny kitu byly opatřeny

rezistory. Ty slouží jako ochrana výstupních pinů ATmega2561 v případě, že by byl na kitu ATmega328P zkrat. Maximální přípustný proud jedním pinem ATmega2561 je 40 mA, celkový proud jedním portem by však neměl překročit 100 mA. Maximální proud jedním pinem proto byl stanoven na 20 mA. Hodnota odporu ochranného rezistoru byla vypočtena z Ohmova zákona:

$$R = \frac{U}{I} = \frac{5}{0,020} = 250 \Omega, \quad (6.1)$$

Z řady E6 byl vybrán rezistor o hodnotě 330 Ω . Tím se maximální proud snížil na 15 mA. Ke kitu nebylo potřeba připojovat další součástky, neboť vše potřebné včetně filtračních kondenzátorů je již součástí DPS.

Aby bylo možno programovat kit v obvodu, musela být propojena některá sběrnice umožňující In-System Programming. K programování je využita sériová sběrnice SPI, byly tedy propojeny vodiče MOSI, MISO a SCK obou mikrokontrolerů. Dále je propojena UART sběrnice mikrokontrolerů, na straně ATmega2561 je využit UART1, protože UART0 je obsazen pro komunikaci s PC.

6.5.2 Zapojení relé modulu

Kit ATmega328P i modul LCD displeje se připojují formou zásuvného modulu s hřebínkovým konektorem. Relé modul takto připojit nelze, protože má na výstupu šroubovací svorkovnici. Vstupní konektor hřebínkového typu má dva napájecí piny a dva piny ovládající relé. Tyto dva piny jsou opět opatřeny ochrannými rezistory 330 Ω . Pohyblivý kontakt relé je připojen ke GND, zbývající kontakty NC a NO jsou připojeny přes pull-up rezistory ke kladnému pólu napájení. Pokud kontakt není sepnut, je na daném výstupu měřena logická 1, pokud kontakt sepne, je měřena logická 0.

6.5.3 Zapojení modulu hodin reálného času

Zapojení tohoto modulu je jednoduché, vodiče SDA a SCL jsou připojeny na I2C sběrnici ATmega2561 (v dokumentaci mikrokontroleru je I2C sběrnice označována jako dvouvodičové sériové rozhraní). Pull-up rezistory, které se umísťují na vodiče sběrnice jsou již součástí modulu. Výstupní pin obdélníkového signálu konfigurovatelné frekvence SQ je připojen na pin externího přerušení mikrokontroleru.

6.5.4 Zapojení inkrementálního rotačního enkodéru

Modul rotačního enkodéru má na výstupech CLK (A) a DT (B) pull-up rezistory, proto je taky nutnost modul napájet. Tlačítko umístěné v ose enkodéru pull-up rezistor nemá, proto byl přidán, aby byla jasně definovaná klidová úroveň tlačítka a nezůstal tento vstup ve stavu vysoké impedance. Kvůli zázkmitům způsobených mechanickými kontakty enkodéru je vhodné pro čtení pozice využít stavový automat. Aby bylo možné číst stavy i při vyšší frekvenci otáčení a zároveň nebyl mikrokontroler vyčerpán periodickým čtením vstupů, je vhodné využít přerušení, které je vyvoláno při změně logické hodnoty pinu.

Proto byl enkodér připojen na piny PE5 až PE7 umožňující externí přerušení od změny logické hodnoty.

6.5.5 Zapojení maticové klávesnice

Maticová klávesnice je propojena s piny mikrokontroleru. Pull-up rezistory výstupních portů mikrokontroleru budou nastavovány programově, není tedy potřeba přidávat externí rezistory. Maticová klávesnice má 8 pinů, což je shodně jako počet pinů jednoho portu mikrokontroleru ATmega2561. Pro obsluhu maticové klávesnice byl tedy vyhrazen celý port A, což následně usnadnilo implementaci programu.

6.5.6 Zapojení modulu LCD displeje

Modul LCD displeje má na výstupní piny vyvedené 4 datové vodiče, nelze jej tedy používat v režimu 8bitové komunikace. Dále jsou dostupné piny RS (Register Select) a E (Enable data read/write). Jak bylo popsáno v kapitole 5.6, pět tlačítek je připojeno na pin AO. Šesté tlačítko RST (reset) je vyvedeno na samostatný pin. V případě použití LCD modulu jako zásuvného modulu do Arduino UNO slouží jako reset mikrokontroleru.

Vstupně výstupní piny byly podobně jako kit ATmega328P Xplained Mini připojeny k řídicímu mikrokontroleru přes ochranné rezistory. Piny s tlačítky byly připojeny přímo, jelikož z nich bude pouze čtena hodnota – nebudou buzeny.

6.5.7 Zapojení reproduktoru

Kvůli odstranění stejnosměrné složky výstupního signálu z mikrokontroleru je reproduktor připojen přes jednoduchý RC článek typu horní propust. Reprodukter je testován na signálu o frekvenci 2000 Hz, požadovaná mezní frekvence RC článku byla nastavena na 1000 Hz. Odpor rezistoru byl zvolen 200 Ω. Z výchozího vztahu 4.2 dostaneme úpravou vztah pro výpočet požadované kapacity kondenzátoru.

$$f_m = \frac{1}{2 \cdot \pi \cdot R \cdot C}, \quad (6.2)$$

$$C = \frac{1}{2 \cdot \pi \cdot R \cdot f_m} = \frac{1}{2 \cdot \pi \cdot 200 \cdot 1000} = 0,796 \mu F, \quad (6.3)$$

Pro RC článek byl zvolen kondenzátor o kapacitě 1 μF. To sníží mezní frekvenci článku na hodnotu 795,78 Hz.

$$f_m = \frac{1}{2 \cdot \pi \cdot R \cdot C} = \frac{1}{2 \cdot \pi \cdot 200 \cdot 1 \cdot 10^{-6}} = 795,78 \text{ Hz}, \quad (6.4)$$

6.5.8 Zapojení modulu s termistorem

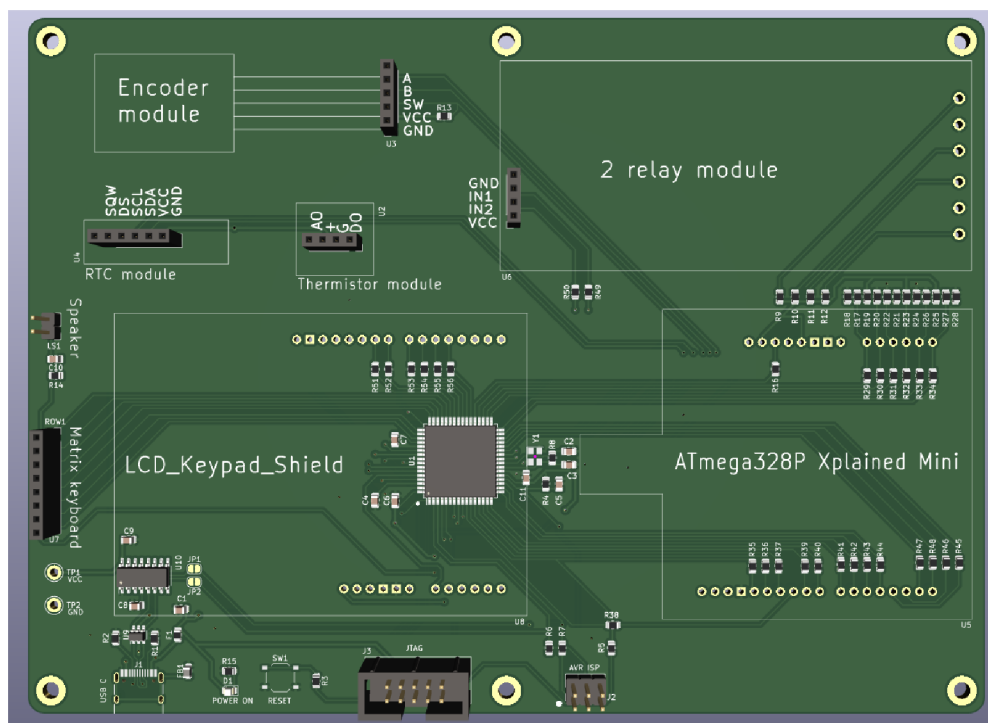
Modul má dva výstupní piny, analogový a digitální výstup. Analogový výstup je připojen na nulový kanál ADC převodníku mikrokontroleru. Digitální výstup je připojen na pin PG5.

7. REALIZACE DESKY PLOŠNÝCH SPOJŮ

Deska plošných spojů byla navržena tak, aby odpovídala nárokům na testovací stoliči. Prioritou nebylo dodržení maximálních rozměrů desky, ale vhodné rozmístění testovaných modulů tak, aby k nim byl snadný přístup a bylo je možné snadno do desky vložit i z desky vyjmout. Plně dostačující tak je dvouvrstvý plošný spoj, který navíc umožňuje i případné úpravy po výrobě DPS.

Od počátku se počítalo s profesionální výrobou DPS u JLCPCB. Výrobce stanovuje soubor pravidel, které je nutné dodržet, aby byla deska vyrobitelná. Mezi ně patří například minimální šířka izolace mezi vodiči, tloušťka vodičů, výrobní rozměry desky, ale také vlastnosti a umístění potisku. Všechna pravidla byla před návrhem DPS zohledněna a zanesena do návrhového prostředí KiCad jako omezující podmínky. Během návrhu lze dodržení pravidel kontrolovat pomocí DRC (Design Rule Check – kontrola pravidel návrhu). [35] [36]

Deska byla u výrobce JLCPCB také osazena součástkami. Některé THT součástky (Through-hole technology – klasická montáž součástek s drátovými vývody) však nebyly dostupné, takže byly osazeny ručně. Jednalo se především o hřebíkové konektory a programovací konektory JTAG a ISP. Velikost SMD součástek (Surface Mount Device – součástka pro povrchovou montáž) byla zvolena tak, aby bylo možné součástky případně vyměnit ručně. U rezistorů a kondenzátorů tedy byla zvolena pouzdra 0805.



Obrázek 7.1 3D vizualizace DPS v programu KiCad (pohled shora)

7.1 Způsoby připojení testovaných modulů

Maticová klávesnice a reproduktor jsou během testování umístěny mimo DPS. Obě zařízení mají poměrně dlouhé propojovací kabely a jejich umístění na DPS není nutné. Zbytek modulů je umístěn přímo na desce plošných spojů. Většina modulů je propojena pomocí hřebíkových konektorů.

Rotační enkodér je možné připojit do hřebíkového konektoru ve vertikální poloze, ale není to vhodné. Stisknutí tlačítka v ose hřídele enkodéru vyžaduje poměrně vysokou sílu a mechanické namáhání konektoru by bylo velmi vysoké. Pro modul enkodéru tak byl na 3D tiskárně vytvořen plastový přípravek, do něhož se modul vloží v horizontální poloze. K testovací stolici je pak modul připojen pomocí propojovacího kabelu. Plastový přípravek poskytuje oporu a snadný přístup při testování modulu.

Relé modul má na výstupu šroubovací svorky. Aby se předešlo nutnosti šroubovat při testování každého modulu šest svorek propojujících modul s testovací stolicí, byly do DPS umístěny pružinové kontakty. Tento typ kontaktů, známý také jako „pogo piny“, se využívá u magnetických konektorů, nebo také u testovacích sond. Pružinový kontakt sestává z nepohyblivé části přiletované do DPS, pružiny, a pohyblivé části s kontaktní plochou. Kontaktní plocha se pak přitiskne na cínové plochy na spodní straně DPS relé modulu v místě, kde jsou výstupní svorky. Pro správné spojení je potřeba přesné zarovnání relé modulu a pružinových kontaktů a vyvinutí síly, která stlačí pružiny kontaktů. Proto byl i pro relé modul vytvořen na 3D tiskárně plastový přípravek. [37]

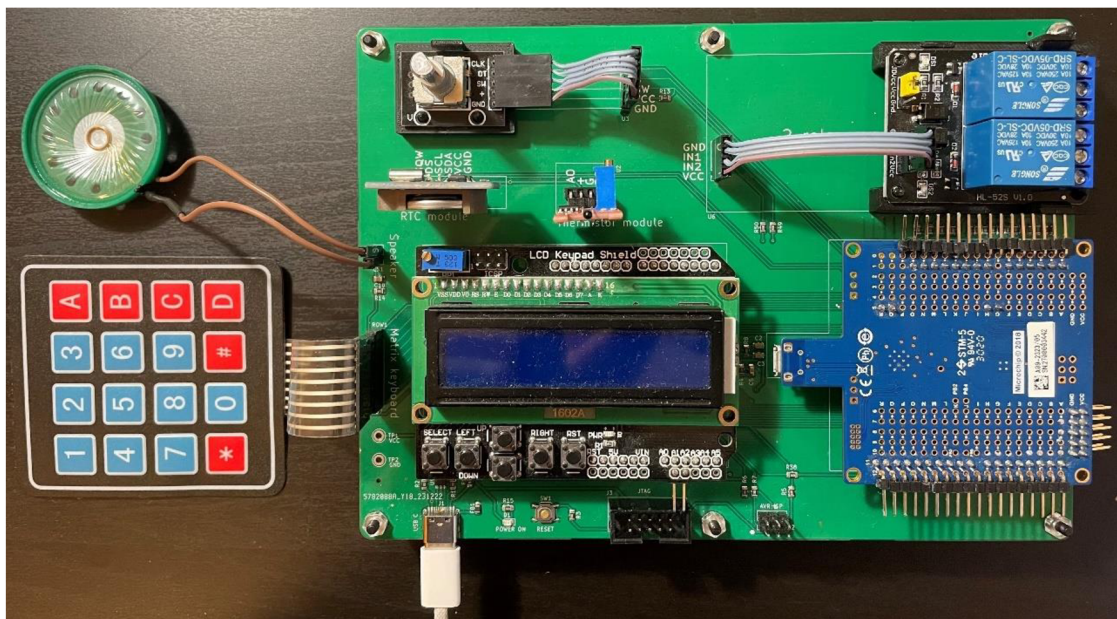
Na DPS byly vytvořeny zemnicí plochy kvůli zjednodušení spojení součástek s GND, snížení impedance a snížení elektromagnetického záření. Dále jsou na DPS umístěny dva testovací body, TP1 a TP2, na kterých lze měřit napájecí napětí desky. Rozměry DPS jsou 185 x 135 mm. Vzhledem k velikosti DPS a mechanické síle nutné pro vložení modulu LCD displeje a modulu ATmega328P do konektorů na desce bylo přidáno 6 montážních otvorů, které zamezují nadměrnému ohybu desky.

7.2 Fáze oživení desky plošných spojů

Po výrobě byla DPS oživena a bylo zkontrolováno korektní napájení všech komponent. Následně byly ověřeny funkce mikrokontroleru ATmega2561.

K programování a ladění programu byly využity obě sběrnice JTAG i ISP, jelikož sběrnice JTAG je na stejných pinech jako piny relé modulu. Pro testování relé modulu bylo tedy nutné sběrnici JTAG vypínat. Naopak u sběrnice ISP byl konflikt se sběrnici UART připojenou k USB převodníku. Po inicializaci UART sběrnice nebylo možné sběrnici ISP používat k programování mikrokontroleru. S touto skutečností bylo ve fázi návrhu DPS počítáno a na sběrnici byly umístěny pájecí propojovací můstky („solder jumpers“), které umožňují proletováním spojit obvod.

Ve fázi ladění byl mezi mikrokontroler a převodník na USB umístěn přepínač. Pokud bylo potřeba programovat MCU přes sběrnici ISP, byl obvod mezi USB převodníkem rozpojen a bylo tak možno sběrnici ISP použít.



Obrázek 7.2 Osazená deska plošných spojů se všemi testovanými moduly

Při fázi ladění byly na DPS nalezeny 2 chyby. První chybou bylo, že mezi mikrokontrolerem a převodníkem na USB nebyly překřížené vodiče sběrnice UART. Pin vysílání mikrokontroleru (pin TXD0) byl připojen k pinu vysílání převodníku CH340. Stejná situace byla u pinu příjmu dat RXD. Tato situace byla ve fázi ladění vyřešena přidáním přepínače popsaného výše, ve finální verzi byly piny překříženy vodiči mezi pájecími propojovacími můstky.

Druhou chybou na DPS byla chyba v propojení sběrnice SPI mezi ATmega2561 a testovanou ATmega328P. Sběrnice měla sloužit pro nahrávání programu do testované ATmega328P, následně však bylo zjištěno, že sběrnici pro programování nepůjde použít (viz. kapitola 9.1.5). Tato chyba tak neměla vliv na výslednou funkci.

Z hlediska přístupnosti k modulům by bylo vhodné umístit některé moduly ve větší vzdálenosti od sebe. Vzhledem k problému s programováním modulu ATmega328P by bylo vhodné modul otočit, nebo jej alespoň umístit do větší vzdálenosti od LCD modulu. Současná orientace modulu byla zvolena z důvodu zjednodušení DPS, zhoršuje však přístup ke konektoru USB.

8. KATEGORIE TESTŮ JEDNOTLIVÝCH MODULŮ

Před návrhem samotného programového vybavení se ukázalo jako stěžejní nalézt podobnosti ve způsobu testování modulů a podle toho software strukturovat. Testované moduly tedy byly rozřazeny do tří kategorií:

8.1 Plně automatizované testy

Tyto testy nevyžadují žádnou interakci ze strany uživatele, jsou tedy plně automatizované. Po spuštění testu testovací stolice vyhodnotí výsledek a ten je pak zobrazen na grafickém rozhraní.

Do této kategorie spadají testy těchto modulů:

- kit ATmega328P Xplained Mini
- modul s relé
- modul reálného času s EEPROM pamětí

8.2 Testy, kde uživatel interaguje s testovaným modulem

Při spuštění testů z této kategorie je uživatel vyzván k určité interakci s testovaným modulem (například stisknutí tlačítek klávesnice). Testovací stolice pak snímá, zda došlo k dosažení požadovaných hodnot. Pokud nějaké hodnoty chybí, je uživatel prostřednictvím grafického rozhraní vyzván k zadání těchto hodnot. Pokud dojde k dosažení všech hodnot, testovací stolice test ukončí a daný test je úspěšně absolvován. Pokud je však modul poškozen a daná hodnota není dosažitelná, může uživatel daný test ukončit a do záznamu je uložena příslušná chybějící hodnota.

Do této kategorie spadají testy těchto modulů:

- rotační enkodér
- maticová klávesnice
- tlačítka na modulu LCD displeje

8.3 Testy, kde uživatel vyhodnocuje správnou funkci modulu

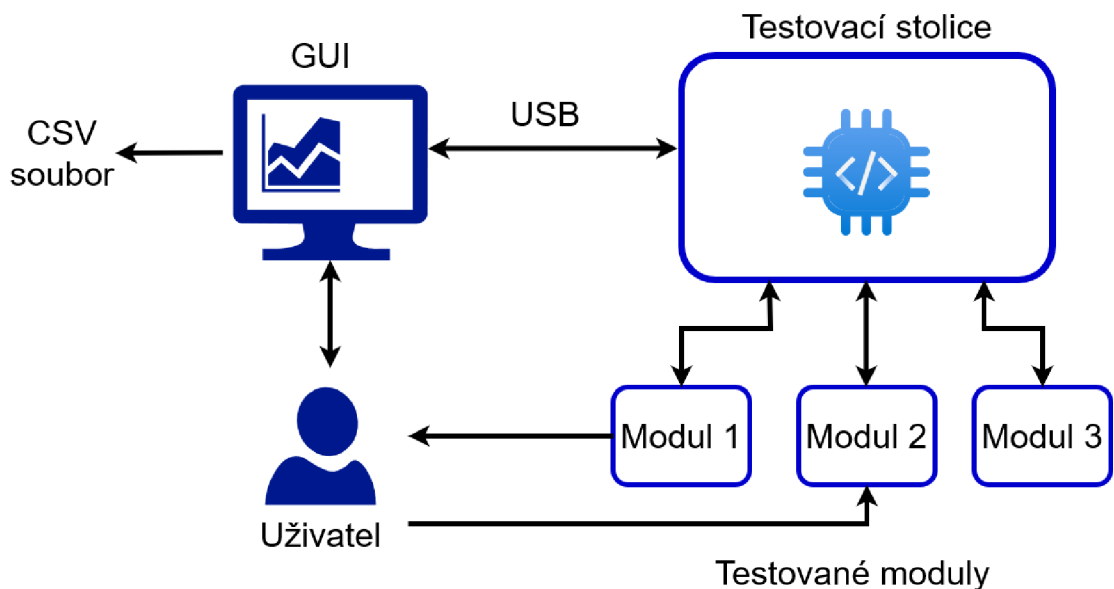
Do této kategorie patří testy, které nelze testovací stolicí vyhodnotit. Jedná se například o modul LCD displeje, kdy je na displeji zobrazen testovací obrazec a uživatel vyhodnocuje jeho korektní zobrazení. Testovací stolice v tomto případě pouze inicializuje testované moduly na potřebné hodnoty a uživatel je prostřednictvím grafického rozhraní vyzván ke kontrole hodnot.

Do této kategorie spadají testy těchto modulů:

- modul LCD displeje
- reproduktor
- modul s termistorem

9. PROGRAMOVÉ VYBAVENÍ

Celkem bylo potřeba vytvořit tři samostatné programy: firmware pro testovací stolicí, firmware pro modul ATmega328P Xplained Mini a program pro grafické uživatelské rozhraní.



Obrázek 9.1 Blokové schéma použití testovací stolice

Blokové schéma použití testovací stolice je na obrázku 9.1. Grafické uživatelské rozhraní umožňuje uživateli konfigurovat spuštění testů, nastavení ukládání výsledků a zobrazování výsledků. Testovací stolice je s počítačem, na němž je spuštěn program s GUI, propojena pomocí sběrnice USB. Jednotlivé moduly jsou pak s testovací stolicí propojeny prostřednictvím různých sběrnic. Dle kategorie testů (viz. kapitola 8) pak uživatel v průběhu testování interaguje s některými moduly, jiné testy jsou plně automatizované a nevyžadují zásah ze strany uživatele. Výsledky testování lze pak exportovat do CSV souboru.

9.1 Mikroprogramové vybavení

Pro testovací stolicí i testovaný modul s ATmega328P byla zvolena obdobná architektura hlavní smyčky programu. Oba mikrokontrolery komunikují s nadřazeným systémem pomocí sběrnice UART a u obou lze definovat určitou sadu příkazů. Příkazy jsou ve formě textových řetězců, takže jsou čitelné i pro člověka. Díky tomu lze jednoduše komunikovat se zařízeními i bez grafického rozhraní přes sériový terminál jako například PuTTY.

9.1.1 Vývojové prostředí

Firmware pro testovací stolicí i ATmega328P byl napsán v programu Microchip Studio. Jedná se o vývojové prostředí vyvíjené přímo výrobcem mikrokontrolerů, firmou Microchip Technology. Prostředí je založené na Microsoft Visual Studio, je však přizpůsobené pro mikrokontrolery Atmel a umožňuje snadné ladění pomocí debuggeru. [38]

Pro konfiguraci projektu byl využit online nástroj Atmel START. Ten umožňuje založit projekt pro konkrétní mikrokontroler, nakonfigurovat jeho periferie, přidat ovladače, middleware a případně další softwarové komponenty. Konfigurovaný projekt je pak možné stáhnout a otevřít v prostředí Microchip Studio. Podpora nástroje Atmel START byla v roce 2023 ukončena a je nahrazena nástrojem MPLAB Code Configurator. Ten je však určen pro integraci do vývojového prostředí MPLAB IDE. [39]

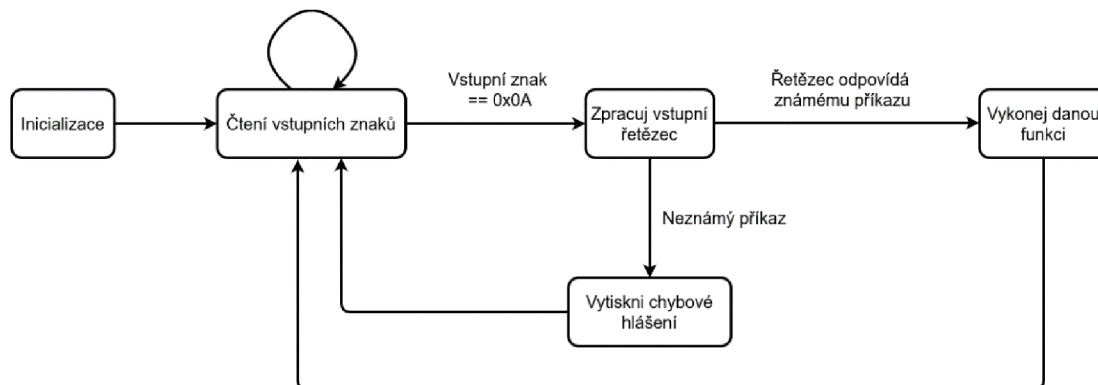
Nástroj Atmel START byl využit především ke konfiguraci projektu, většina ovladačů a middleware byly vytvořeny vlastní, protože neodpovídaly potřebám projektu, nebo měly omezené funkce a bylo by je potřeba upravit.

9.1.2 Ladění

Téměř nezbytnou součástí vývoje firmware je ladění. Jedná se o fázi odstraňování chyb a ověřování správné funkce kódu. K ladění mikrokontrolerů se využívají externí zařízení, nazývané „debugger“. V případě ATmega328P je debugger umístěn přímo na desce, jelikož se jedná o vyhodnocovací kit pro vývoj na daném MCU. V případě testovací stolice však bylo potřeba využít externí debugger. Pro ladění byl použit debugger Atmel-ICE, připojený ke sběrnici JTAG.

9.1.3 Firmware pro modul s ATmega328P

Dle specifikace testů, popsané v kapitole 3 je nutné, aby ATmega328P umožňovala nastavování obou logických úrovní na digitálních výstupech, dále čtení hodnot na digitálních vstupech a měření analogových hodnot na pinech analogově-digitálního převodníku. Tomu odpovídá i sada příkazů, které ATmega328P umí zpracovat.



Obrázek 9.2 Hlavní smyčka programu

Vývojový diagram programu je na obrázku 9.2. Po inicializaci mikrokontroleru a jeho periférií se vykonává hlavní smyčka programu. Příjem znaků na sběrnici UART je realizován v přerušení. V obslužné rutině přerušení se pak přijatý znak zapíše do cyklické fronty a posune se ukazatel na konec fronty. Hlavní smyčka programu se opakuje, dokud není přijat znak „posun o řádek“ (v angličtině „Line Feed“), v šestnáctkové soustavě 0x0A. Po přijetí tohoto znaku se z cyklické fronty přečte celý vstupní řetězec. Následně je porovnáván se známými příkazy a pokud odpovídá některému z příkazů, je vykonána příslušná funkce. Pokud neodpovídá žádnému známému příkazu, je na sběrnici UART vytisknuta chybová hláška.

Jednotlivé funkce pak analyzují vstupní řetězec a nastavují periférie mikrokontroleru. Příkladem budiž následující ukázka: po sběrnici UART je odeslána zpráva „aio_read_ADC3\n“, kde posledním znakem je posun o řádek. Testovaný modul zprávu přijme a zjistí, že se jedná o funkci čtení hodnot z ADC převodníku. Zavolá tedy odpovídající funkci. Ta nejprve zjistí, zda byl vybrán validní vstupní kanál a poté provede převod analogové hodnoty na daném vstupu převodníku. Převedená hodnota je vytisknuta na sběrnici UART ve formátu „ADC3=512\n“, kde 512 je hodnota z ADC převodníku.

Funkce pro obsluhu digitálních vstupů a výstupů fungují obdobným způsobem, místo kanálu ADC je však specifikován daný port a pin, který má být nastaven.

9.1.4 Firmware pro testovací stolici

Firmware pro testovací stolici má obdobný vývojový diagram hlavní smyčky programu jako modul s ATmega328P. Rozdílná je sada příkazů, která obsahuje samostatné příkazy pro test každého modulu a další obslužné příkazy. Dalším rozdílem je, že po nalezení příkazu ve vstupním řetězci je na UART vytisknuto potvrzení o nalezení daného příkazu. Tato funkcionalita zvýšila spolehlivost přenosu zpráv a lepší odezvu grafického uživatelského rozhraní.

Každý modul má vlastní test a pro každý test existuje samostatný příkaz ve formě „TEST *NÁZEV_MODULU*“, kde *NÁZEV_MODULU* je zkrácený název modulu. Příkladem může být příkaz „TEST RTC“, který provede test modulu s RTC a EEPROM paměti.

Pokud při testování modulu není nalezena žádná chyba, odpoví testovací stolice zprávou „PASS“. Pokud je nalezena nějaká chyba, je vytisknuta zpráva „FAIL: *CHYBOVÁ_ZPRÁVA*“. V chybové zprávě pak jsou vytisknuty všechny chyby daného modulu detekované testovací stolicí. U modulu hodin reálného času a u modulu s ATmega328P se během testování komunikuje přes sběrnici I2C, respektive UART. Pokud testované zařízení nekomunikuje, je testovací stolicí vytisknuta chyba komunikace s modulem. Každá zpráva odeslaná testovací stolicí je zakončena znakem „posun o řádek“.

Plně automatizované testy jsou spuštěny a výsledky jsou vytisknuty po sériové lince. Většina ostatních testů však běží ve smyčce, dokud nejsou dosaženy všechny stavy

testovaného modulu, nebo dokud uživatel test neukončí. Výjimkou je test reproduktoru, kdy je provedena pouze konfigurace pulzně šířkového výstupu a mikrokontroler pak může vykonávat jinou práci, dokud není test ukončen. Ukončení testů je realizováno příkazem „STOP“.

Při zapnutí napájení testovací stolice se provede inicializace periférií ATmega2561, před testováním každé sady modulů se pak dále provádí inicializace dané sady a inicializace testovacího prostředí příkazem „TEST INIT“.

Implementace samotných testů se řídí specifikací testů, definovanou v kapitole 5. Při implementaci byly brány v potaz různé zdroje poruch, které by měly být testovací stolicí detekovány a průběžně byla upravována také samotná specifikace testů.

9.1.5 Programování modulu s ATmega328P

V kapitole 5.1 bylo navrženo programovat testovaný mikrokontroler ATmega328P pomocí ISP. Před návrhem desky byla tato možnost vyzkoušena. Za pomoci ISP programátoru byl do ATmega328P úspěšně nahrán program. Ačkoliv nebyla nalezena žádná implementace knihovny pro programování mikrokontrolerů megaAVR z jiného mikrokontroleru, v dokumentaci ATmega328P je postup programování detailně popsán. Počítalo se tedy s implementací vlastní knihovny.

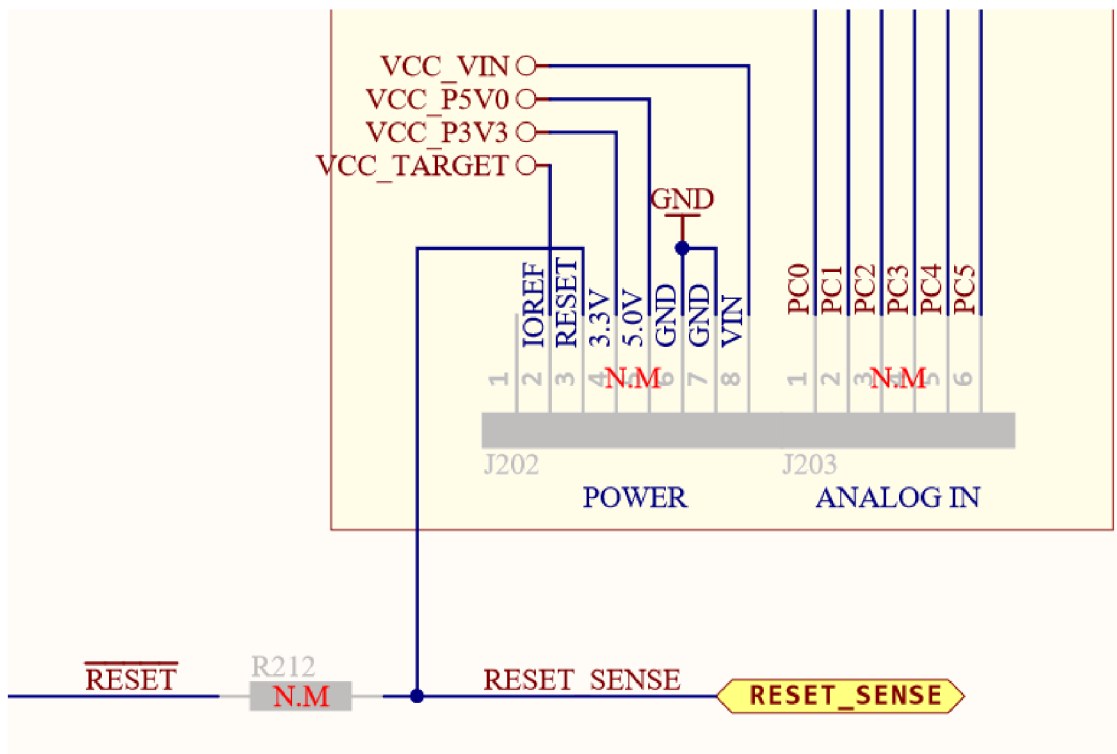
Firmware ATmega328P byl zkompileován do binárního souboru. Následně bylo potřeba soubor převést do formátu, který umožní uložení do paměti mikrokontroleru na testovací stolicí. V Linuxových prostředích lze využít k převodu binárních souborů do hexadecimálního formátu příkaz „xxd“. Pro prostředí Windows existuje samostatný program plnící stejnou funkci. Tento program slouží k zobrazení hexadecimálního obsahu binárního souboru nebo k vytvoření binárních souborů z hexadecimálního vstupu. Hexadecimální znaky pak lze uložit do paměti mikrokontroleru ve formě pole hodnot. [40]

Během implementace knihovny byl zjištěn problém, který znemožňoval mikrokontroler jednoduše přes rozhraní ISP programovat. Mikrokontroler je připojen k debuggeru na desce. Obě zařízení mezi sebou komunikují přes rozhraní AVR debugWIRE. Jedná se o rozhraní používané na mikrokontrolerech Atmel, které mají malý počet pinů. K přenosu dat se používá jediný pin, fyzicky umístěný na RESET pinu mikrokontroleru. Z toho plyne také omezení – pokud je sběrnice debugWire povolena, nelze RESET pin použít k vyvolání externího restartu MCU. Sběrnice se povoluje nastavením konfiguračního bitu DWEN. [15]

Běžně se kit s ATmega328P programuje přes debugger na desce. Rozhraní debugWIRE je tedy povolené. Problémem je, že jakmile je rozhraní povolené, nefunguje RESET pin mikrokontroleru, který je vyžadován k programování přes sběrnici ISP. Sběrnici debugWIRE lze vypnout smazáním konfiguračního bitu DWEN. To lze provést dvěma způsoby: paralelním programování s napětím 12 Voltů, nebo přes samotnou

sběrnici debugWIRE. První možnost by vyžadovala náročné úpravy DPS. Byla tedy uvažována možnost použít sběrnici debugWIRE.

Existuje implementace knihovny, která umožňuje používat sběrnici debugWIRE na platformě Arduino [41]. Arduino se tak stane debuggerem, který umožňuje ladění programu přes GDB (GNU Debugger) příkazy. Knihovna tedy byla v prostředí Arduino IDE zkompilována a nahrána do desky Arduino. Následně bylo Arduino připojeno k RESET pinu modulu ATmega328P Xplained Mini. Zapojení modulu však znemožnilo použití debugWIRE rozhraní. Schéma zapojení pinu RESET je na obrázku 9.3. Modul nemá reset pin mikrokontroleru (na obrázku vyznačen jako \overline{RESET}) připojen přímo k RESET pinu na výstupním konektoru modulu – rezistor R212 není osazen. Místo toho je připojen k debuggeru na desce (signál RESET_SENSE). Debugger pak řídí, zda bude mikrokontroler restartován. Uvedené zapojení znemožňuje použít pin RESET na výstupním konektoru modulu k připojení k ATmega328P přes debugWIRE rozhraní.



Obrázek 9.3 Schéma zapojení reset pinu kitu ATmega328P Xplained Mini [42]

Mikrokontroler má však pin RESET vyveden na konektor ISP. Tímto způsobem se povedlo přes Arduino debugger desku programovat, standardně však konektor na modulech osazen není. Existovala možnost přidat na DPS pružinový konektor, který by umožnil propojit reset pin ATmega328P na konektoru ISP a testovací stolici.

Situace byla konzultována s vedoucím práce a došlo se k závěru, že přidání pružinového konektoru a použití debugWIRE rozhraní je složité. Bylo rozhodnuto, že

namísto toho bude programování modulu integrováno do grafického uživatelského rozhraní a modul bude s počítačem propojen USB kabelem.

K programování je tedy využit program AVRDUDE. Ten umožňuje programování Flash a EEPROM paměti a konfiguračních bitů mikrokontrolerů AVR [43]. Volání programu bylo přidáno do grafického uživatelského rozhraní. Po stisknutí tlačítka program vyhledá, ke kterému portu je modul připojen a nahraje do něj binární soubor obsahující testovací program.

9.1.6 Poznámky k implementaci testů

Testování EEPROM paměti bylo nejprve provedeno na celém rozsahu paměti, tedy 32 768 bitů. Spolehlivé komunikace s EEPROM paměti a RTC modulem bylo dosaženo při frekvenci I2C sběrnice 10 kHz. Paměť umožňuje zapisovat po skupině 32 bajtů. Pokud nebudeme uvažovat adresaci na I2C sběrnici a další arbitrážní příkazy, bude samotná doba přenosu 32 768 bitů při frekvenci sběrnice 10 kHz trvat:

$$t = \frac{1}{f} \cdot n = \frac{1}{10000} \cdot 32768 = 3,2768 \text{ s}, \quad (9.1)$$

Do paměti se nejprve zapisují logické 1, přečtou a zkontrolují se zapsané hodnoty a proces se opakuje pro logické 0. Přenos dat na I2C sběrnici tedy zabere asi 13 sekund. K tomu je potřeba přičíst ostatní arbitrážní příkazy a dobu zápisu a čtení EEPROM paměti. Testování paměti by tak zabralo poměrně dlouhý čas. Bylo proto rozhodnuto, že se bude testovat pouze prvních 32 bajtů paměti.

U modulu hodin reálného času se mimo jiné testuje výstup obdélníkového signálu konfigurovatelné frekvence. Frekvence pulzů je nastavena na 32 kHz a jako reference pro srovnání se používá časovač 5 mikrokontroleru ATmega2561. Následně se změří délka trvání jedné periody obdélníkového signálu z RTC. Časovač 5 byl nastaven bez děličky frekvence na taktovací frekvenci mikrokontroleru 16 MHz. V ideální případě by se tak změřená hodnota měla rovnat 488 tikům časovače:

$$n = \frac{f_1}{f_2} = \frac{16000000}{32768} = 488,28 [-], \quad (9.2)$$

Jelikož nebyly uvedeny zvýšené požadavky na přesnost měření frekvence obdélníkového výstupu RTC modulu, byla k této hodnotě přidána tolerance asi 20 %, v níž je změřená hodnota stále považována za správnou. Pokud by bylo potřeba měřit frekvenci výstupu přesněji, bylo by vhodné změřit dobu jedné periody obdélníkového signálu několikrát a hodnoty průměrovat či jinak filtrovat.

Po nahrání programu do ATmega328P Xplained Mini přes USB konektor v některých případech dojde k chybě komunikace s modulem a modul neodpovídá například na první zprávu přijatou přes UART sběrnici. Tento problém se bohužel nepodařilo odstranit a vyžadoval by hlubší zkoumání. Chyba nastává pouze jednou za několik desítek

opakování testování, takže bylo složité najít její příčinu. Situace je prozatím vyřešena opakováním spuštěním testování modulu.

U LCD modulu tlačítka tvoří odporový dělič, stisknutí každého tlačítka pak generuje na výstupu jinou hodnotu (za předpokladu že je najednou stisknuto jen jedno tlačítko). Výstup odporového děliče je připojen na analogový vstup ATmega2561 na testovací stolici. Jelikož hodnoty rezistorů výrobce neuvádí a ani z modulu se je nepodařilo přečíst, bylo při výpočtu předpokládáno, že jsou hodnoty rezistorů stejné. Dle zapojení na obrázku 5.5 byly vypočítány předpokládané hodnoty při stisknutí každého tlačítka.

Při stisknutí prvního tlačítka s názvem „RIGHT“ dojde k přizemnění výstupu, na výstupu ADC převodníku by tedy měla být hodnota 0. Při stisknutí ostatních tlačítek lze očekávanou analogovou hodnotu vypočítat dle vzorce pro odporový dělič. Ačkoliv jsou hodnoty odporu jednotlivých rezistorů neznámé, za předpokladu, že jsou stejné se vykrátí a výstupní hodnotu lze vypočítat. Při stisknutí tlačítka „UP“ je předpokládaná hodnota na ADC převodníku:

$$ADC_{UP} = (2^n - 1) \cdot \frac{R1}{R1+R2} = (2^{10} - 1) \cdot \frac{1}{2} = 511,5 [-], \quad (9.3)$$

Měřením však bylo zjištěno, že hodnoty odporů nejsou stejné, vzorec 9.3 tedy nelze použít. Výstupní hodnota na každém tlačítku byla změřena experimentálně na dodaných modulech LCD displeje. Získané hodnoty napětí byly převedeny na předpokládané hodnoty na ADC převodníku a byly vytvořeny intervaly, které reprezentují konkrétní stisknuté tlačítko.

Pro testování reproduktoru byl využit časovač 5 mikrokontroleru ATmega2561 v režimu PWM. Reproduktor je buzen obdélníkovým signálem, což je pro účel testování funkčnosti reproduktoru dostačující. Časovač byl nastaven do módu *Clear Timer on Compare* – vymaž časovač při porovnání. V tomto módu je horní hranicí časovače hodnota v registru OCR5A. Po dosažení hodnoty v tomto registru může být generováno přerušení a následně je časovač vynulován a začíná počítat opět od nuly po hodnotu v OCR5A. V obslužné rutině přerušení od časovače dochází ke změně logické hodnoty na výstupu. Dělička frekvence byla nastavena na 1024 a hodnota, při níž se vyvolává přerušení byla vypočítána tak, aby nastavená frekvence byla 2 kHz: [44]

$$OCR5A = \frac{f_{CPU}}{2 \cdot n_{prescaler} \cdot f_{required}} = \frac{16000000}{2 \cdot 1024 \cdot 2000} = 3,9062 [-], \quad (9.4)$$

Do registru OCR5A tedy byla zapsána hodnota 4. Zastavení generování PWM signálu se pak provádí při příjetí příkazu „STOP“.

Při testování modulu termistoru je periodicky čtena hodnota na ADC převodníku na pinu, kde je připojen analogový pin termistoru. V grafickém rozhraní je pak vykreslován graf posledních dvaceti hodnot převedených na napětí v rozsahu 0 až 5 V. Nová hodnota

je vykreslena každých 300 ms. Uživatel může termistor zahrát a sledovat změny v grafu a z toho vyvodit, zda termistor funguje. Testovací stolice ukončí výpis hodnot na ADC převodníku po přijetí ukončovacího příkazu „STOP“.

9.2 Grafické uživatelské rozhraní

Na aplikaci pro počítač bylo kladeno několik požadavků. Hlavním z nich bylo, aby umožňovala komunikaci s testovací stolicí přes sériový port a zpracování přijatých zpráv. Dalšími požadavky bylo, aby umožňovala výběr jednotlivých testů (v případě, že by nebylo potřeba testovat některé moduly), přehledné uživatelské rozhraní a srozumitelné zobrazení výsledků testů.

Vývoj takové aplikace lze usnadnit použitím frameworku, který již obsahuje knihovny a moduly pro dané funkce, takže není nutné psát celou aplikaci od základů. Frameworků pro počítačové aplikace umožňující komunikaci přes sériový port je celá řada, často jsou však pro programovací jazyky C# nebo Java. Existují však i frameworky pro C++ nebo Python jako například QT Creator nebo GTK. Vzhledem ke své jednoduchosti a splnění všech požadavků byl vybrán framework Tkinter. Jedná se o standardní grafické rozhraní Pythonu a je dostupné na mnoha platformách operačních systémů. [45]

9.2.1 Použitý návrhový vzor

Pro vývoj aplikace bylo důležité vybrat vhodný návrhový vzor, aby bylo možné snadno provádět úpravy, umožnit opakované použití kódu a zachovat modularitu aplikace. Požadavkům na funkcionalitu aplikace vyhověl návrhový vzor MVC (Model-View-Controller – Model-Pohled-Ovladač). Tento vzor cílí na oddělení dat, uživatelského rozhraní a řídicí logiky aplikace a často se používá právě pro návrh uživatelského rozhraní. Model se v dané architektuře stará o uchování dat, validaci a manipulaci s daty. Model je nezávislý na ovládacím prvku a může být sdílený mezi různými částmi aplikace. Pohled (View) zobrazuje uživateli informace z modelu. Nezodpovídá za logiku aplikace ani manipulaci s daty, stará se pouze o zobrazení informací uživateli. Pohled nemusí být pouze grafický rozhraním, může se jednat i o příkazový řádek nebo API (Application Programming Interface – rozhraní pro programování aplikací). Ovladač (Controller) propojuje Model s Pohledem. Zpracovává události a požadavky uživatele, přebírá uživatelské vstupy a provádí odpovídající akce. Stejně tak musí ovladač být schopen reagovat na změny stavu Modelu. [46]

V implementaci návrhového vzoru MVC byl Model použit především ke zpracování sériové komunikace s testovací stolicí. Proto byla vytvořena samostatná třída, která uchovává informace o sériové lince, přijatá data, nebo data připravená k odeslání.

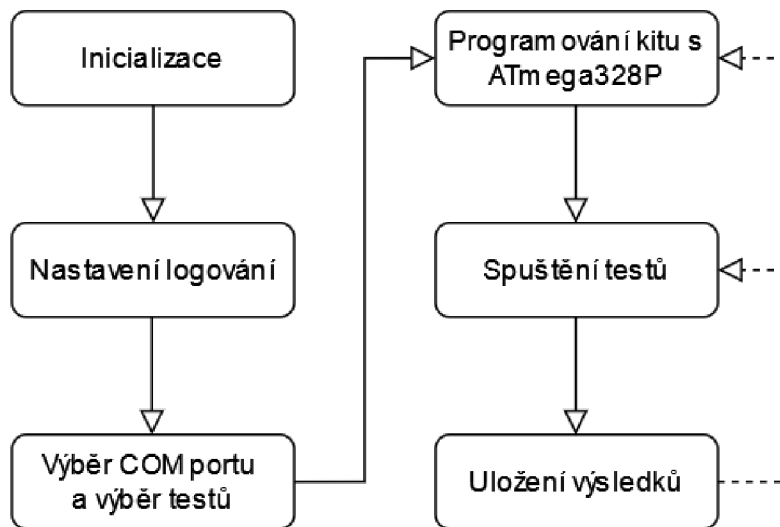
Ve vrstvě Pohled bylo implementováno grafické rozhraní. Celkem jsou v aplikaci tři druhy oken: výchozí okno aplikace, okno pro testy, kde uživatel interaguje s testovaným modulem a okno pro testy kde uživatel vyhodnocuje správnou funkci modulu. Pro každé

okno aplikace byla vytvořena samostatná třída. Logika aplikace nebyla ponechána striktně na Ovladači. Pokud bylo účelné ponechat funkcionalitu ve vrstvě Pohled a přenesení dané funkce do vrstvy Ovladače by přineslo zbytečné komplikace, byla funkcionalita ponechána ve vrstvě Pohled.

Nejkomplexnější částí programu je pak Ovladač. Ten se stará o zpracování událostí z vrstvy Model i Pohled. V této vrstvě jsou uloženy výsledky testů. Ovladač je také zodpovědný za ukládání protokolů o testování do souboru.

Použití architektury MVC pro návrh GUI se ukázalo jako vhodné, umožnilo snazší vývoj i srozumitelnost kódu. Stěžejním prvkem použití tohoto modelu je vhodné rozdělení funkcionalit do jednotlivých vrstev modelu. Nesprávném rozdělení může vést k neefektivní výměně dat mezi jednotlivými vrstvami.

Pro ladění GUI bylo potřeba komunikovat přes sériový port. K tomu byl využit Null-modem emulátor *com0com*. Jedná se o program umožňující vytvoření virtuálních sériových portů a nastavitelné propojení portů mezi sebou [47]. Během ladění byly vytvořeny dva virtuální porty a ty byly mezi sebou propojeny. K jednomu portu bylo připojeno GUI a na druhém portu byl otevřen terminál se sériovým monitorem.



Obrázek 9.4 Vývojový diagram spuštění testu z grafického rozhraní

Na obrázku 9.4 je znázorněný postup spuštění testů přes GUI z uživatelského pohledu. Po spuštění programu může uživatel zapnout ukládání protokolů o zkoušce a případně zadá název logovacího souboru. Následně uživatel vybere, který COM port z nabídky odpovídá portu testovací stolice. Program také umožňuje výběr testů, které mají být spuštěny. Před spuštěním testu s ATmega328P Xplained Mini je nutné do kitu nejprve nahrát firmware. Následně je možné spustit testování. Pokud je zapnuto ukládání protokolů o zkoušce, uživatel je na závěr vyzván k zadání čísla testovaného kitu (každý kit je označen unikátním číslem). Při dalším spuštění testů již pak není nutná další konfigurace a je možné rovnou spustit testy (v obrázku 9.4 označeno přerušovanou čarou).

9.2.2 Ukládání protokolů o zkoušce

Grafické rozhraní umožňuje ukládat výsledky testování do CSV (Comma-separated values – hodnoty oddělené čárkami) souboru. Na první řádek souboru je uložena hlavička a do každého následujícího řádku jsou ukládány výsledky testování jedné sady modulů. Spolu s výsledky je vždy uložen název (unikátní číslo) dané sady.

Tabulka 9.1 Příklad uloženého protokolu o zkoušce

Kit number	ATmega328P Xplained Mini	Rele modul	RTC a EEPROM modul	Rotacni enkoder
13	PASS	FAIL	PASS	PASS
42	PASS	PASS	PASS	PASS
69	PASS	FAIL	PASS	SKIPPED
5	PASS	PASS	SKIPPED	SKIPPED

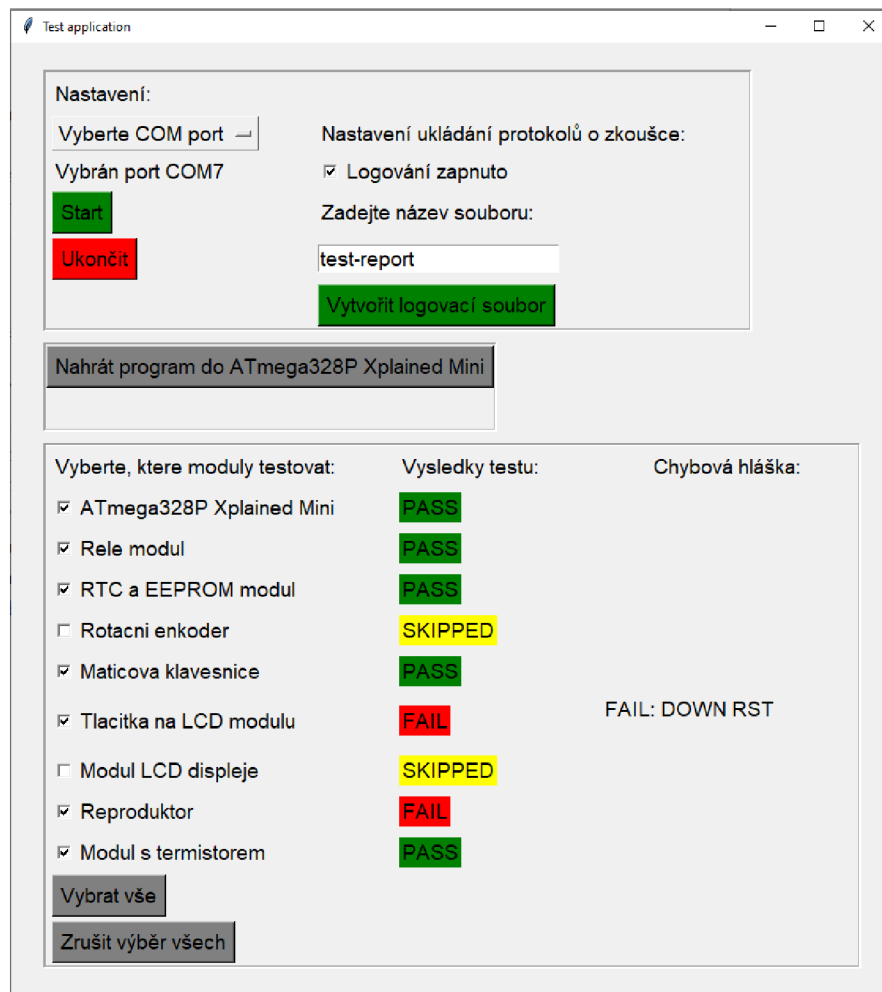
Maticova klavesnice	Tlacitka na LCD modulu	Modul LCD displeje	Reproduktor	Modul s termistorem
PASS	PASS	PASS	PASS	PASS
PASS	FAIL	PASS	PASS	PASS
SKIPPED	SKIPPED	SKIPPED	SKIPPED	FAIL
SKIPPED	SKIPPED	SKIPPED	PASS	PASS

9.2.3 Popis grafického rozhraní pro jednotlivé kategorie testů

Zatímco u plně automatizovaných testů není potřeba další interakce, testy z ostatních kategorií interakci ze strany uživatele vyžadují. Uživatel je tedy během testování informován, jak má s testovanými moduly interagovat. K tomu jsou využívány samostatná okna na grafickém rozhraní.

V případě testů, kde uživatel interaguje s testovaným modulem, je uživatel informován o tom, jaké akce má provést. Typicky se jedná například o stisknutí všech tlačítek modulu, nebo otočení enkodéru oběma směry. Pokud testovací stolice detekuje nedosažení nějakého stavu, je chybějící stav vytisknut do GUI jako chyba. Uživatel se pak může pokusit například znovu stisknout chybějící tlačítko. Pokud chyba přetrvává, je chybějící stisknutí považováno za chybu tlačítka.

Ve druhém případě se jedná o kategorii testů, kde uživatel vyhodnocuje správné chování modulu. V případě LCD displeje je vyzván k porovnání obrazovky displeje se zadaným vzorem a ke kontrole rozsvícení všech pixelů displeje. V případě testu reproduktoru je vyzván k potvrzení, zda slyší tón z reproduktoru. Test termistoru je realizován cyklickým čtením analogových hodnot na výstupu termistoru. Uživatel tak může vyzkoušet například změnu výstupní hodnoty modulu při zahřátí termistoru.



Obrázek 9.5 Ukázka hlavního okna grafického rozhraní

9.2.4 Spouštění aplikace grafického rozhraní

Výslednou aplikaci grafického rozhraní lze spustit z libovolného prostředí, v němž je instalován Python. Aplikaci je pak možné spustit zavoláním následujícího příkazu ze složky *test_app*:

```
python main.py
```

Pro počítače s operačním systémem Windows lze vygenerovat spustitelný soubor. Pro export aplikace byla využita knihovna *PyInstaller*. Jedná se o jeden z nejrozšířenějších způsobů vytvoření spustitelného souboru z Python skriptů. Knihovna je multiplatformní, lze ji tedy použít pro export do většiny operačních systémů, stará se o správu závislostí a umožňuje jednoduché použití z příkazové řádky.

Pro generování spustitelného souboru byl připraven dávkový soubor *testapp_export.bat*. V tomto souboru jsou nastaveny jednotlivé parametry. Pro usnadnění spuštění aplikace a zachování kompatibility je do výsledného balíčku vložen také program AVRDUDE a binární soubor pro programování modulu s ATmega328P.

Při spouštění aplikace na univerzitním počítači byl detekován virus ve spustitelném souboru. Při bližším zkoumání bylo zjištěno, že se jednalo o falešnou detekci antivirovým programem. Bylo vyzkoušeno jiné nastavení exportu aplikace (bez volby *onefile* umožňující aplikaci zabalit do jediného souboru), ale jednoduché řešení nebylo nalezeno. Jelikož aplikace bude využívána jen na vybraných počítačích, bylo rozhodnuto buď pro aplikaci udělit výjimku z antivirových pravidel, nebo exportovat aplikaci až na daném počítači. Tento způsob nebyl vyzkoušen, PyInstaller však na každém počítači vytváří binární soubory specifické pro daný počítač, které mohou být antivirovým programem detekovány jako virus. Vytvořením aplikace pro daný počítač by se falešné detekci viru mělo předejít. [48]

Jelikož vytvoření spustitelného souboru nebylo vedoucím považováno za prioritní, problematika detekce antivirovým programem nebyla dále zkoumána a bylo uvažováno vyzkoušení navržených řešení. Typicky bude aplikace spouštěna přímo přes Python, využití spustitelného souboru je uvažováno pouze v ojedinělých případech.

10. VYHODNOCENÍ TESTŮ

V průběhu psaní testů a ladění firmware testovací stolice byly postupně vyhodnocovány také jednotlivé testy. Dle získaných poznatků byla průběžně upravována specifikace testů a byly doplňovány testy, které nebyly definovány v původní verzi specifikace. Dále bylo nutné ošetřit stavy, kdy modul vůbec není připojen k testovací stolici, nebo je zcela vadný. V této fázi ladění byla nalezena například chyba v knihovně pro I2C komunikaci. Pokud modul nebyl připojen nebo neodesílal odpověď na odeslanou zprávu, program se zasekl v nekonečné smyčce, čekající na odpověď modulu. Tento stav byl ošetřen přidáním časovače. Pokud modul neodpoví do stanoveného času na žádost testovací stolice, je vyhodnocena chyba na sběrnici I2C.

Obvodové testy bylo v některých případech možné vyhodnocovat i bez použití samotného modulu. V době ladění testu relé modulu ještě nebyly k dispozici pružinové konektory, takže propojení s relé modulem by bylo náročné realizovat. Piny řídicího mikrokontroleru byly během ladění propojovány tak, aby bylo simulováno spínání relé. Po přidání pružinových kontaktů byly funkce ověřeny se skutečným relé modulem.

Ve fázi vyhodnocování jednotlivých testů byly také simulovány závady na jednotlivých modulech. U modulu s ATmega328P jsou postupně testovány jednotlivé piny. Bylo tedy ověřeno, že testy dokáží detekovat nejen zkrat pinu vůči napájecímu napětí, ale i mezi sousedním pinem modulu. Všechny kombinace zkratů vůči GND, vůči napájecímu napětí a propojení se sousedním pinem dokáže test detekovat. Při detekci chyby není vypsána konkrétní závada, ale pouze chyba na daném pinu. Nalezení konkrétní příčiny je tedy na uživateli.

Simulovat závady při obvodovém testu bylo poměrně snadné. Složitější byla situace u zařízení komunikujících po sběrnících. Zde bylo možné jednoduše navodit chybu při přenosu, ovlivnit výsledky přenosu na sběrnici by však vyžadovalo přidat emulátor na danou sběrnici, který by simuloval například chybu v registru hodin reálného času. ;

Pro ověření správné funkčnosti bylo požádáno o dodání skutečných vadných modulů. Byly dodány 2 kity se všemi moduly. Bližší specifikace, jaká chyba na modulech je nebyla poskytnuta, detekce závad na modulech tedy rozhodovala pouze testovací stolice. U obou modulů selhalo nahrávání programu do ATmega328P. V programu Microchip Studio bylo vyzkoušeno, že do modulů skutečně nelze nahrát program. Testovací stolice tedy v tomto případě správně zachytila vadu na modulech. Testovány byly všechny moduly dodaných kitů, žádné další závady však detekovány nebyly a chování modulů bylo korektní. Více vadných modulů bohužel nebylo k dispozici, takže ověření správné detekce závad testovací stolicí bylo u ostatních modulů realizováno pouze umělou simulací chyby. Při reálném poškození by se však modul mohl chovat jinak, než bylo očekáváno a testovací stolice by nemusela být schopna tento typ chyby zachytit. Aby se následujícím případům předešlo, byla věnována pečlivá práce tvorbě specifikace testů.

11. ZÁVĚR

V rámci této práce byl řešen návrh testovací stolice pro testování výukových kitů. Výukové kity sestávají z několika desek plošných spojů. V rámci práce byla provedena rešerše komerčně dostupných testerů pro testování desek plošných spojů. Vzhledem k finančním nákladům na pořízení takových zařízení však bylo rozhodnuto navrhnout vlastní, jednoúčelové testovací zařízení.

Diskusí s vedoucím práce byla stanovena koncepce testovací stolice. Ta je popsána v kapitole 4. Byly definovány obecné požadavky na testování a bylo navrženo testovat moduly uživatelského rozhraní poloautomaticky interakcí s osobou spouštějící testy.

Před samotným návrhem desky plošných spojů pro testovací stolicí byl proveden rozbor testovaných modulů. Kapitola 5 se zabývá stručným popisem každého modulu kitu. Jsou rozebrány možné způsoby poškození modulu, ať vlivem špatného zapojení, nebo opotřebením modulu. Dále jsou popsány důsledky poškození na funkčnost modulu. Z tohoto rozboru je pak vytvořena specifikace testů, tedy detailní způsob, jak každý modul testovat. V rámci této kapitoly byl brán důraz na co nejvyšší pokrytí testů, aby testy odhalily všechny typy závad na modulu.

V navazujících kapitolách byl popsán samotný návrh testovací stolice. Jako řídicí mikrokontroler byl vybrán ATmega2561. Požadavkem bylo, aby testovací stolice umožňovala komunikaci s počítačem, proto bylo přidáno USB rozhraní, které slouží také pro napájení stolice. Jednotlivé moduly byly zapojeny tak, aby bylo možné otestovat všechny dostupné funkce modulu a případná závada nebo zkrat nepoškodila testovací stolicí. Z elektrických schémat byla vytvořena deska plošných spojů, jejíž finální podoba s připojenými testovanými moduly je na obrázku 7.2. Pro snazší přístup k modulu enkóderu byl na 3D tiskárně vytvořen plastový přípravek, do něhož je modul zasunut. Obdobný přípravek byl vytvořen také pro relé modul. Zde přípravek slouží k zarovnání vývodů modulu s pružinovými kontakty. Pružinové kontakty umožňují rychlé a spolehlivé propojení výstupů modulu namísto původních šroubovacích svorek.

Po úspěšném oživení desky následovala fáze návrhu programového vybavení. Byl navržen firmware pro testovaný modul s ATmega328P, umožňující nastavování digitálních výstupů, čtení výsledků analogově digitálního převodu a další funkce. Modul komunikuje s testovací stolicí přes sběrnici UART. Firmware samotné testovací stolice obsahuje sadu příkazů, umožňující spouštět jednotlivé testy přes sériovou sběrnici USB. Implementace samotných testů se řídí specifikací testů definovanou v kapitole 5.

Součástí práce je také program s grafickým uživatelským rozhraním. To umožňuje nastavení připojení k testovací stolicí, konfiguraci testů a přehledné zobrazení výsledků testování, včetně tisku zpráv o konkrétním typu závady. Program dále umožňuje ukládání protokolu o testování do CSV souboru, kde jsou spolu s výsledky testů ukládány také unikátní čísla každého kitu. Pro návrh grafického rozhraní bylo využito standardní grafické rozhraní Pythonu, Tkinter. Pro danou aplikaci se osvědčilo použití návrhového

vzoru MVC, jelikož umožnilo oddělení jednotlivých funkcionalit programu a usnadnilo vývoj i srozumitelnost kódu.

Finální návrh se neobešel bez chyb, jednou z nich bylo opomenuté překřížení vodičů na sběrnici UART propojující mikrokontroler testovací stolice s převodníkem na USB. Tato chyba byla opravena. Významnějším problémem byla nemožnost programovat modul s ATmega328P přímo z testovací stolice. Ve fázi návrhu bylo ověřeno, že modul lze programovat přes rozhraní ISP. Z důvodu neobvyklého zapojení debuggeru, který je na desce umístěn, však ve finálním zapojení není možné modul přes tuto sběrnici programovat. Tato problematika je blíže rozebrána v kapitole 9.1.5. Byla navržena dvě možná řešení a bylo rozhodnuto programovat modul přes sběrnici USB. Modul je tedy sice nutné propojit s počítačem, programování je však integrováno přímo do grafického rozhraní.

Správná funkce testovací stolice byla ověřena jak na vadných modulech, tak simulací závad. Testovací stolice je schopná zachytit všechny chyby definované ve specifikaci testů a poloautomatické testování zajišťuje dobu testování v řádu desítek sekund. Vzhledem ke frekvenci testování a množství kitů je toto uspokojující výsledek, který splňuje zadaná kritéria. V navazující práci by bylo možné diskutovat o možnostech lepší automatizace celého procesu testování, svazující podmínkou je však nákladnost takového řešení.

LITERATURA

- [1] THACKER, B. H.; DOEBLING, S. W.; HEMEZ, F. M.; ANDERSON, M. C.; PEPIN, J. E. et al. *Concepts of Model Verification and Validation*. Online. 2004. USA, 2004. Dostupné z: <https://doi.org/10.2172/835920>. [cit. 2024-04-10].
- [2] NIDHRA, Srinivas. Black Box and White Box Testing Techniques - A Literature Review. Online. *International Journal of Embedded Systems and Applications*. 2012, roč. 2, č. 2, s. 29-50. ISSN 18395171. Dostupné z: <https://doi.org/10.5121/ijesa.2012.2204>. [cit. 2024-04-09].
- [3] LOTZ, Christophe; COLLINS, Peter; WIATROWSKI, Dominique a . FUNCTIONAL BOARD TEST - COVERAGE ANALYSIS what does it mean when a functional test passes? Online. *Computer Science, Engineering*. 2005, roč. 2005, s. 8. Dostupné z: <https://api.semanticscholar.org/CorpusID:6665829>. [cit. 2024-04-10].
- [4] MYSORE, G.D.; CONRAD, J.M. a NEWBERRY, B. A Microcontroller-Based Bed-of-Nails Test Fixture to Program and Test Small Printed Circuit Boards. Online. In: *Proceedings of the IEEE SoutheastCon 2006*. IEEE, 2006, s. 104-107. ISBN 1-4244-0168-2. Dostupné z: <https://doi.org/10.1109/second.2006.1629332>. [cit. 2023-12-23].
- [5] AYODELE, Abiola, 2023 PCB Testing: A Comprehensive Guide to Techniques, Tools, and Best Practices. Online. Wevolver. Dostupné z: <https://www.wevolver.com/article/test-pcb-everything-you-need-to-know>. [cit. 2023-11-05].
- [6] RETTEW, Craig, 2020. Everything You Need to Know about PCB Assembly Functional Testing. Online. Seeed Studio. Dostupné z: <https://www.seeedstudio.com/blog/2021/06/18/pcb-functional-testing/>. [cit. 2023-11-07].
- [7] HIGHLEAP ELECTRONIC, An In-Depth Guide to PCB Functional Testing, 2023. Online. Highleap Electronic. Dostupné z: <https://hilelectronic.com/pcbfunctional-testing/>. [cit. 2023-11-07].
- [8] DIGITALTEST, 2023. Test systems. Online. Digitaltest. Dostupné z: <https://www.digitaltest.com/en/>. [cit. 2023-12-19].
- [9] *FLYING PROBE TESTER*. Online. ACCULOGIC INC. Acculogic. 2023. Dostupné z: <https://www.acculogic.com/products/flying-probe-tester>. [cit. 2023-12-10].
- [10] JTAG TECHNOLOGIES, 2021. What is JTAG boundary-scan? Online. JTAG. Dostupné z: <https://www.jtag.com/what-is-jtag-boundary-scan>. [cit. 2023-12-20].
- [11] ITOCHU CORPORATION, 2023. Takaya Flying Probe Tester. Online. TAKAYA. Dostupné z: <https://www.takaya-itochu.com/>. [cit. 2023-12-20].
- [12] MICROCHIP TECHNOLOGY, 2020. ATmega328P Xplained Mini User Guide. Online. Microchip. Rev. B. Dostupné z:

- <https://ww1.microchip.com/downloads/en/DeviceDoc/ATmega328P-XplainedMini-UG-DS50002659B.pdf>. [cit. 2023-10-15].
- [13] MICROCHIP TECHNOLOGY, 2011. Atmel AVR Microcontrollers. Online. Microchip. Rev. B. Dostupné z: <https://ww1.microchip.com/downloads/en/DeviceDoc/AVROVERVIEW-02-11%20flyer%20-%20US%20-%20low%20res-4.pdf>. [cit. 2023-11-20].
- [14] KUNIKOWSKI, Wojciech, Ernest CZERWIŃSKI, Paweł OLEJNIK a Jan AWREJCEWICZ. An Overview of ATmega AVR Microcontrollers Used in Scientific Research and Industrial Applications. *Pomiary Automatyka Robotyka* [online]. 2015, 2015-03-05, 215(1), 15-20 [cit. 2023-12-23]. ISSN 14279126. Dostupné z: http://dx.doi.org/10.14313/PAR_215/15
- [15] ATMEL CORPORATION, 2015. ATmega328P Datasheet. Online. Microchip. 7810D-AVR-01/15. Dostupné z: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-AutomotiveMicrocontrollers-ATmega328P_Datasheet.pdf. [cit. 2023-10-17].
- [16] ATMEL CORPORATION, 2016. *AT15007: Differences between ATmega328/P and ATmega328PB: Application note*. Online. Microchip. 42626C. Dostupné z: https://ww1.microchip.com/downloads/en/AppNotes/Atmel-42559-Differences-between-ATmega328P-and-ATmega328PB_ApplicationNote_AT15007.pdf. [cit. 2023-12-23].
- [17] SAUERBREY, J.; SCHMITT-LANDSIEDEL, D. a THEWES, R., 2003. A 0.5-v 1-- μ W successive approximation adc. Online. *IEEE Journal of Solid-State Circuits*. Roč. 38, č. 7, s. 1261-1265. ISSN 0018-9200. Dostupné z: <https://doi.org/10.1109/JSSC.2003.813217>. [cit. 2023-12-23].
- [18] ROBOTAKI, 2023. *2 Channel 5V Relay Module Arduino Compatible*. Online. RoboTaki. Dostupné z: <https://www.robotaki.com/4-channel-5v-relay-module-arduino-compatible---copy>. [cit. 2023-11-03].
- [19] ROETTJER, Phil. *Testing Techniques to Improve Relay Reliability*. Online. Relay Testing Services, LLC. 2005. Dostupné z: http://www.relaytestingservices.co/ee_article_pdf.pdf. [cit. 2023-11-03].
- [20] MAXIM INTEGRATED, 2015. *DS3231: Extremely Accurate I2C-Integrated RTC/TCXO/Crystal*. Online. Analog Devices. 10. Dostupné z: <https://www.analog.com/media/en/technical-documentation/data-sheets/DS3231.pdf>. [cit. 2023-11-04].
- [21] ATMEL CORPORATION, 2003. *AT24C32/64: 2-Wire Serial EEPROM*. Online. Microchip. 0336K-SEEPR-7/03. Dostupné z: <https://ww1.microchip.com/downloads/en/DeviceDoc/doc0336.pdf>. [cit. 2023-11-09].
- [22] COMPONENTS101, 2021. *KY-040 Arduino Rotary Encoder User Manual*. Online. Components101. Dostupné z: https://components101.com/sites/default/files/component_datasheet/KY-04-Rotary-Encoder-Datasheet.pdf. [cit. 2023-11-04].

- [23] EETRON, 2020. *How to use 4X4 matrix keypad with Arduino*. Online. EETRON. Dostupné z: <https://ettron.com/how-to-use-4x4-matrix-keypad-with-arduino/>. [cit. 2023-11-15].
- [24] CLAUSEN, Thomas, 2017. *Walking Through the 1602 LCD Keypad Shield for Arduino*. Online. Advanced Communication Networks. Dostupné z: <https://www.master-acn.fr/walking-through-the-1602-lcd-keypad-shield-for-arduino/>. [cit. 2023-10-18].
- [25] SHENZHEN EONE ELECTRONICS CO. *Specification for LCD Module 1602A-1 (V1.2)* [online]. SHENZHEN EONE ELECTRONICS CO., LTD. Dostupné z: https://www.laskakit.cz/user/related_files/eone-1602a1_datasheet.pdf. [cit. 2023-10-19].
- [26] HITACHI, 1998. *HD44780U (LCD-II): Dot Matrix Liquid Crystal Display Controller/Driver*. Online. Sparkfun. Dostupné z: <https://www.sparkfun.com/datasheets/LCD/HD44780.pdf>. [cit. 2023-12-23].
- [27] STREIT, Jakub, 2023. *Cvičení č. 4: UARTa ADC*. Materiály k předmětu MPC-POR. Online. Brno.
- [28] SHENZHEN YANGXING TECHNOLOGY. *CRYSTAL UNIT YSX321SL*. Online. Shenzhen Yangxing Technology CO. 2023. Dostupné z: <https://image.seapx.com/mall/yangxin/3/20231030/YSX321SL-687795.pdf>. [cit. 2023-11-11].
- [29] MAXIM INTEGRATED, 2003. *Determining Clock Accuracy Requirements for UART Communications*. Online. Analog Devices. Dostupné z: <https://www.analog.com/media/en/technical-documentation/tech-articles/determining-clock-accuracy-requirements-for-uart-communications.pdf>. [cit. 2023-11-11].
- [30] MICROCHIP TECHNOLOGY, 2013. *AVR® Microcontroller Hardware Design Considerations*. Online. Microchip Technology. B. 2018. Dostupné z: <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ApplicationNotes/ApplicationNotes/AN2519-AVR-Microcontroller-Hardware-Design-Considerations-00002519B.pdf>. [cit. 2023-11-14].
- [31] CHEICH, Michael, 2023. *Arduino as ISP and Arduino Bootloaders*. Online. Arduino Documentation. Dostupné z: <https://docs.arduino.cc/built-in-examples/arduino-isp/ArduinoISP>. [cit. 2023-11-14].
- [32] COMPAQ; HEWLETT-PACKARD; INTEL; LUCENT; MICROSOFT; NEC et. al., 2000. *Universal Serial Bus Specification*. Online. PoweredUSB.org. Dostupné z: <http://www.poweredusb.org/pdf/usb20.pdf>. [cit. 2023-11-09].
- [33] MICROCHIP TECHNOLOGY, 2015. *Basic USB Type-C™ Upstream Facing Port Implementation*. Online. Microchip. Dostupné z: <https://ww1.microchip.com/downloads/en/AppNotes/00001914B.pdf>. [cit. 2023-11-09].
- [34] TEXAS INSTRUMENTS, 2013. *USB 2.0 Board Design and Layout Guidelines*. Online. Texas Instruments. Dostupné z: <https://e2e.ti.com/cfs->

- file/_key/communityserver-discussions-components-files/171/USB-2.0-Board-Design-and-Layout-Guidelines.pdf. [cit. 2023-11-09].
- [35] JLCPCB, 2023. *PCB Manufacturing & Assembly Capabilities*. Online. JLCPCB. Dostupné z: <https://jlcpcb.com/capabilities/pcb-capabilities>. [cit. 2023-12-20].
- [36] KEETH, Graham; EVANS, Jon; PETERSON, Gleen; JAHSHAN, David; HUTCHINSON, Phil et al., 2022. *Getting started in KiCad*. Online. KiCad docs. Dostupné z: https://docs.kicad.org/6.0/en/getting_started_in_kicad/getting_started_in_kicad.html. [cit. 2023-12-27].
- [37] MILL-MAX. *Pogo Pins in Your Designs: How, Where, and When to Use Them*. Online. MILL-MAX MFG. CORP. 2020. Dostupné z: <https://www.mill-max.com/engineering-notebooks/spring-loaded-pogo-pins-connectors/pogo-pins-in-your-designs>. [cit. 2024-01-18].
- [38] VISUAL MICRO. *Visual Studio or Atmel Studio?* Online. Visualmicro.com. 2015. Dostupné z: <https://www.visualmicro.com/page/User-Guide.aspx?doc=Getting-started-which-IDE.html>. [cit. 2024-03-22].
- [39] MICROCHIP TECHNOLOGY. *Atmel START*. Online. Microchip Technology. 2022. Dostupné z: <https://www.microchip.com/en-us/tools-resources/search-discover/atmel-start>. [cit. 2024-03-22].
- [40] WEIGERT, Juergen. *Xxd(1) - Linux man page*. Online. Die.net. 1996. Dostupné z: <https://linux.die.net/man/1/xxd>. [cit. 2024-03-05].
- [41] NEBEL, Bernhard. *Dw-link: An Arduino-based debugWIRE debugger*. Online. Github. 2024. Dostupné z: <https://github.com/felias-fogg/dw-link/blob/master/docs/manual.md>. [cit. 2024-04-05].
- [42] ATMEL NORWAY. *ATmega328PB Xplained Mini Schematics*. Online. Microchip Technology. 2017. Dostupné z: https://ww1.microchip.com/downloads/en/DeviceDoc/ATmega328PB_Xplained_Mini_Schematics.pdf. [cit. 2024-04-20].
- [43] NONGNU. *AVRDUDE - AVR Downloader/UploaDEr*. Online. Nongnu.org. 2010. Dostupné z: <https://www.nongnu.org/avrdude/>. [cit. 2024-04-20].
- [44] MICROCHIP TECHNOLOGY. *ATmega640/V-1280/V-1281/V-2560/V-2561/V: 8-bit Microcontroller with 16/32/64KB In-System Programmable Flash. DATASHEET*. Online. Microchip Technology. 2005, 2020. Dostupné z: <https://ww1.microchip.com/downloads/aemDocuments/documents/OTH/Product Documents/DataSheets/ATmega640-1280-1281-2560-2561-Datasheet-DS40002211A.pdf>. [cit. 2024-04-20].
- [45] PYTHON SOFTWARE FOUNDATION. *Tkinter — Python interface to Tcl/Tk*. Online. Python 3.12.3 documentation. 2024. Dostupné z: <https://docs.python.org/3/library/tkinter.html>. [cit. 2024-04-08].
- [46] DEACON, John. *Model-View-Controller (MVC) Architecture*. Online. *JOHN DEACON Computer Systems Development, Consulting & Training*. 1995, s. 6. Dostupné z:

- https://www.academia.edu/30077059/Model_View_Controller_MVC_Architecture. [cit. 2024-04-08].
- [47] VYACHESLAV, Frolov. *Null-modem emulator (com0com)*. Online. Sourceforge. 2018. Dostupné z: <https://com0com.sourceforge.net/>. [cit. 2024-04-10].
- [48] SIDDIQI, Raahim. *Pyinstaller EXE detected as Virus? (Solutions and Alternatives)*. Online. CodersLegacy. 2023. Dostupné z: <https://coderslegacy.com/pyinstaller-exe-detected-as-virus-solutions/>. [cit. 2024-04-29].

SEZNAM SYMBOLŮ A ZKRATEK

ADC převodník	Analog-to-digital converter – Analogově digitální
API programování aplikací	Application Programming Interface – Rozhraní pro
AVR	8bitový RISC mikropočítač od Atmel
CAN	Controller Area Network
CCW	Counterclockwise – Proti směru hodinových ručiček
CITE počítačové testovací prostředí	Computer Integrated Test Environment – Integrované
COM	Communication port – Komunikační sériový port
CSV	Comma-separated values – Hodnoty oddělené čárkami
CW	Clockwise – Po směru hodinových ručiček
DAC převodník	Digital-to-analog converter – Digitálně analogový
DIO	Digital Input-Output – Digitální vstupně výstupní port
DIP	Dual Inline Package – Dvouřadé pouzdro
DMA	Direct Memory Access – Přímý přístup do paměti
DPS	Deska Plošných Spojů
DRC	Design Rule Check – Kontrola pravidel návrhu
EEPROM Elektricky vymazatelná paměť určená pro čtení	Electrically Erasable Programmable Read-Only Memory –
ESD	Electrostatic Discharge – Elektrostatický výboj
FPU s pohyblivou řádovou čárkou	Floating Point Unit – Jednotka pro práci s čísly
GDB v software	GNU Debugger – Standardní nástroj pro hledání chyb
GND	ground – Záporný napájecí pól
GTK	GIMP Toolkit
GUI	Graphical User Interface – Grafické Uživatelské Rozhraní
I2C	Inter-Integrated Circuit – Dvou vodičová sériové sběrnice
ICT	In-Circuit Testing – Testování v obvodu
ISP spojů	In-System Programming – Programování v desce plošných
JTAG	Joint Test Action Group – Standard IEEE 1149.1
LCD	Liquid crystal display – Displej z tekutých krystalů
LED	Light-Emitting Diode – Elektroluminiscenční dioda
MCU	Microcontroller – Mikrokontroler
mEDBG	Mini Embedded Debugger – Mini Vestavěný Debugger

MVC	Model-View-Controller – Model-Pohled-Ovladač
PC	Personal Computer – Osobní počítač
PDI	Program and Debug Interface
PPTC	Polymeric Positive Temperature Coefficient device – Polymerové zařízení s pozitivním teplotním koeficientem
PWM	Pulse Wide Modulation – Pulzně šířková modulace
RTC	Real time clock – Hodiny reálného času
SMD	Surface Mount Device – Součástka pro povrchovou montáž
SPI	Serial Peripheral Interface – Sériové periferní rozhraní
SRAM	Static random-access memory – Statická paměť
THT	Through-hole technology – Klasická montáž součástek s drátovými vývody
TVS	Transient Voltage Suppression – Dioda pro potlačení přechodného napětí
TWI	Two Wire Interface – Dvouvodičová sériová sběrnice
UART	Universal Asynchronous Receiver/Transmitter – Sériový asynchronní přijímač/vysílač
UAT	Uživatelské Akceptační Testování
UFP	Upstream Facing Port
USB	Universal Serial Bus – Univerzální sériové sběrnice

SEZNAM PŘÍLOH

PŘÍLOHA A - NÁVOD K POUŽITÍ	66
PŘÍLOHA B - SCHÉMATA ZAPOJENÍ.....	69
PŘÍLOHA C - OBSAH PŘILOŽENÉHO CD	74

Příloha A - Návod k použití

Tato testovací stolice slouží k testování kitu MPC-POR SKU® kit edu. Testovací stolice umožňuje připojení všech modulů kitu a provedení testu jejich správné funkčnosti. Jmenovitě je možné otestovat tyto moduly:

- ATmega328P Xplained Mini
- LCD Keypad Shield 1602A
- 2-relé modul
- Enkodér
- Modul RTC
- Modul s termistorem
- Reproduktor 8 Ω 0,5 W
- Maticová membránová klávesnice

K testovací stolici je dodáváno také grafické rozhraní, které je možné spustit ze spustitelného (EXE) souboru nebo spuštěním souboru *main.py* pomocí programu Python.

A.1 Instalace software

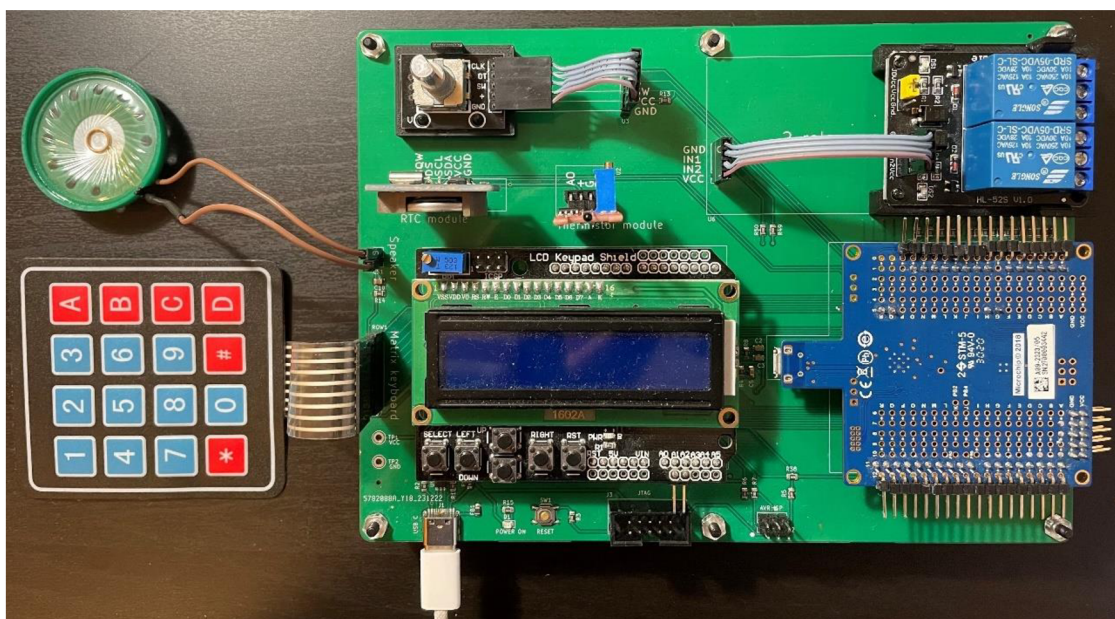
Před použitím testovací stolice je nutné mít na PC nainstalovaný ovladač pro USB převodník CH340. Dále je potřeba stáhnout program grafického rozhraní. Ten je dostupný spolu s celým repositářem projektu na adrese <https://github.com/IvoZal/DP.git>, na přiloženém CD či v elektronické příloze práce.

A.2 Použití testovací stolice

Testovací stolici připojte k počítači přes konektor USB C. Po připojení by se měla rozsvítit LED dioda „POWER ON“, indikující napájení stolice. Dalším krokem je spustit program grafického rozhraní.

Na grafickém rozhraní je nutné vybrat COM port, ke kterému je testovací stolice připojena. Kliknutím na nabídku „Vyberte COM port“ se zobrazí všechny dostupné porty počítače. Pokud si nejste jisti, pod kterým číslem portu je testovací stolice, otevřete ve Windows Správce zařízení (Device Manager) a v nabídce „Porty (COM a LPT)“ najdete port, který má v názvu CH340. Vyberte tedy v grafickém rozhraní daný port a pokračujte dalším krokem. Pro aktualizace nabídky výběru COM portu v grafickém rozhraní je nutné program restartovat.

Do testovací stolice připojte všechny moduly, které chcete otestovat. Orientace modulů je znázorněna popiskami na plošném spoji. Správné připojení modulů je zobrazeno na následujícím obrázku:



Obrázek 11.1 Testovací stolice se všemi testovanými moduly

Výběr testovaných modulů

Po připojení modulů je možné v grafickém rozhraní vybrat, které moduly mají být otestovány. Výběr je možný kliknutím na zaškrťovací políčka vedle názvu každého modulu. Dále je možné označit všechny moduly, nebo zrušit označení všech modulů tlačítky „Vybrat vše“, respektive „Zrušit výběr všech“.

Programování ATmega328P Xplained Mini

Před testováním modulu ATmega328P Xplained Mini je nutné připojit modul k počítači Micro USB kabelem a nahrát do modulu firmware tlačítkem „Nahrát program do ATmega328P Xplained Mini“. Nahrání programu trvá přibližně 10 sekund.

Nastavení ukládání výsledků

Testovací stolice umožňuje ukládat protokoly o zkoušce do CSV souboru. Pro nastavení ukládání výsledků je nutné vybrat zaškrťovací políčko v menu „Nastavení ukládání protokolů o zkoušce“. Dále je nutné zadat název logovacího souboru do políčka pod textem „Zadejte název souboru“. Po zadání názvu souboru je nutné stisknout klávesu <Enter>, nebo stisknout tlačítko „Vytvořit logovací soubor“.

Spuštění testů

Po konfiguraci grafického rozhraní je možné spustit testování tlačítkem „Start“. Uživatel je prostřednictvím vyskakovacích oken žádán o interakci s testovanými moduly. Na hlavní obrazovku jsou pak vypisovány výsledky testování.

Ve sloupci „Výsledky testů“ mohou být tři hodnoty:

- PASS – test prošel
- FAIL – na modulu byla nalezena chyba
- SKIPPED – test modulu byl přeskočen

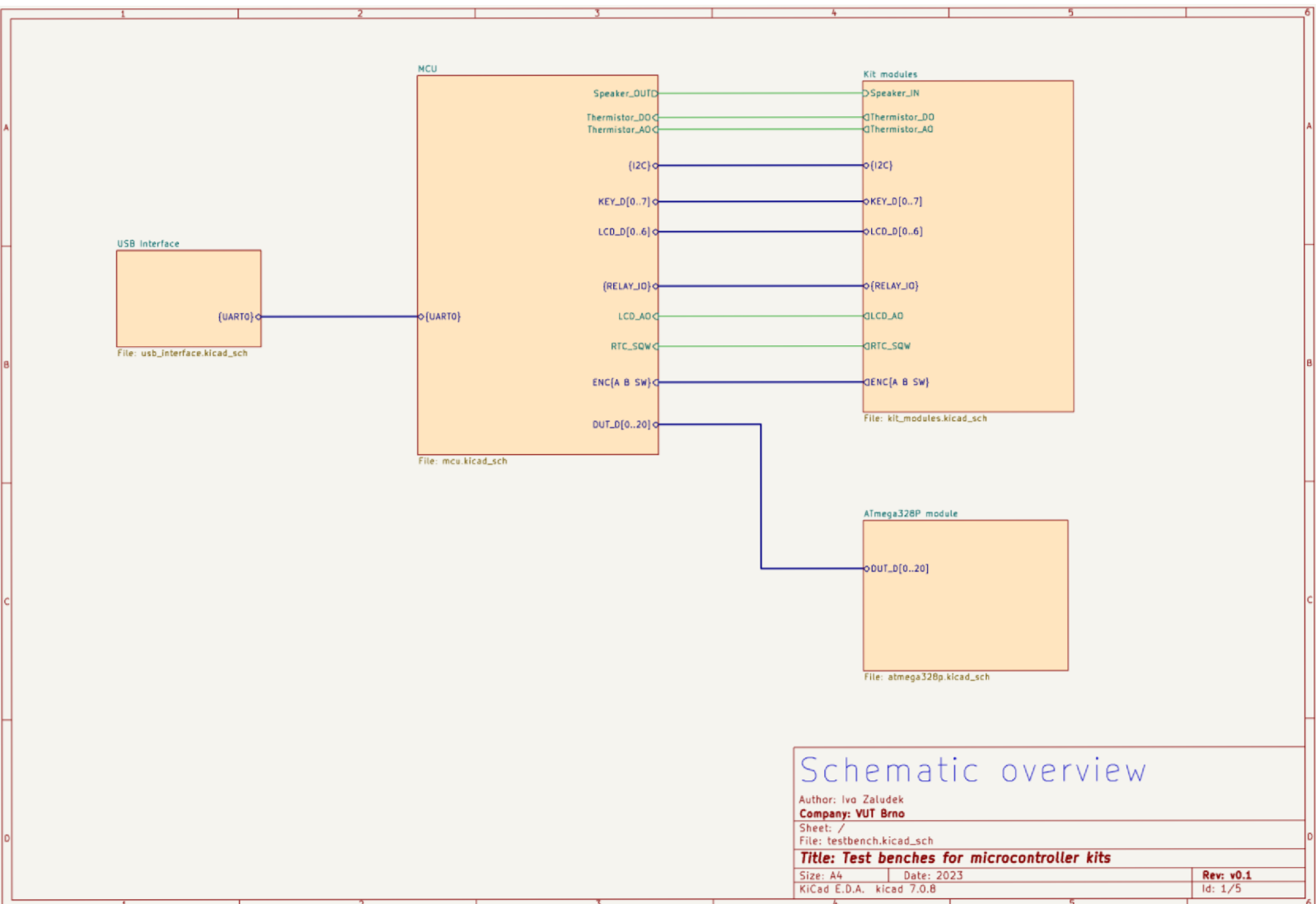
Pokud při testování modulu byla nalezena chyba, jsou do sloupce „*Chybová hláška*“ vypsány detekované chyby.

A.3 Možnosti programování testovací stolice

Do testovací stolice je nahrán firmware *atmega2561_example.hex*. V případě, že by bylo potřeba daný firmware nahradit, je to možné přes rozhraní ISP. Rozhraní JTAG je konfiguračními bity zakázáno. Před programováním přes sběrnici ISP je nutné přerušit kontakt s převodníkem CH340 (přes pájecí propojovací můstky JP1 a JP2), jelikož zde vzniká konflikt se sběrnici UART. Poté stačí připojit ISP programátor k AVR ISP konektoru a nahrát novou verzi firmware.

Příloha B - Schémata zapojení

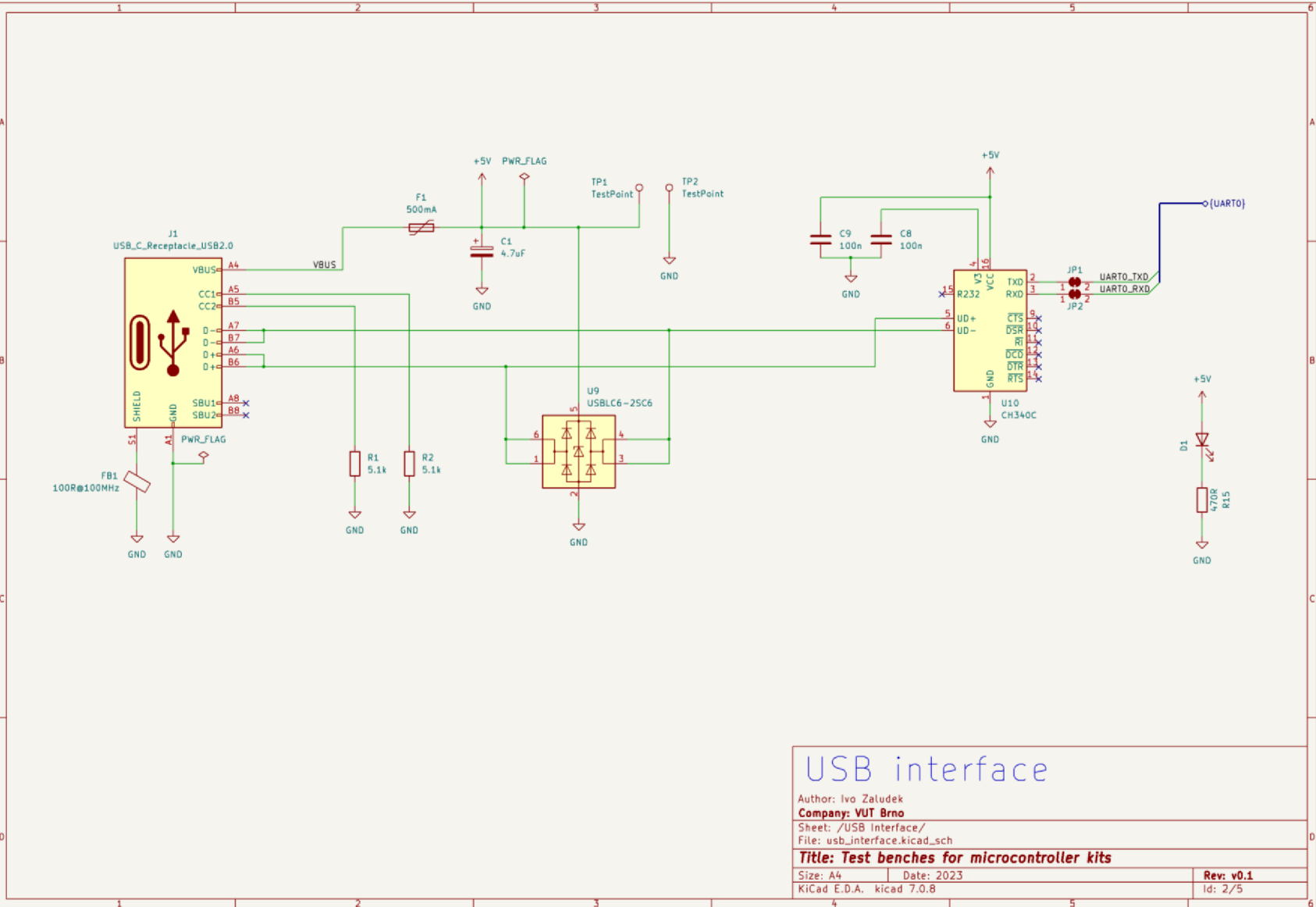
B.1 Přehledové schéma zapojení



Schematic overview

Author: Ivo Zaludek		Rev: v0.1	
Company: VUT Brno		Id: 1/5	
Sheet: /			
File: testbench.kicad_sch			
Title: Test benches for microcontroller kits			
Size: A4	Date: 2023		
KiCad E.D.A. kicad 7.0.8			

B.2 Rozhraní USB



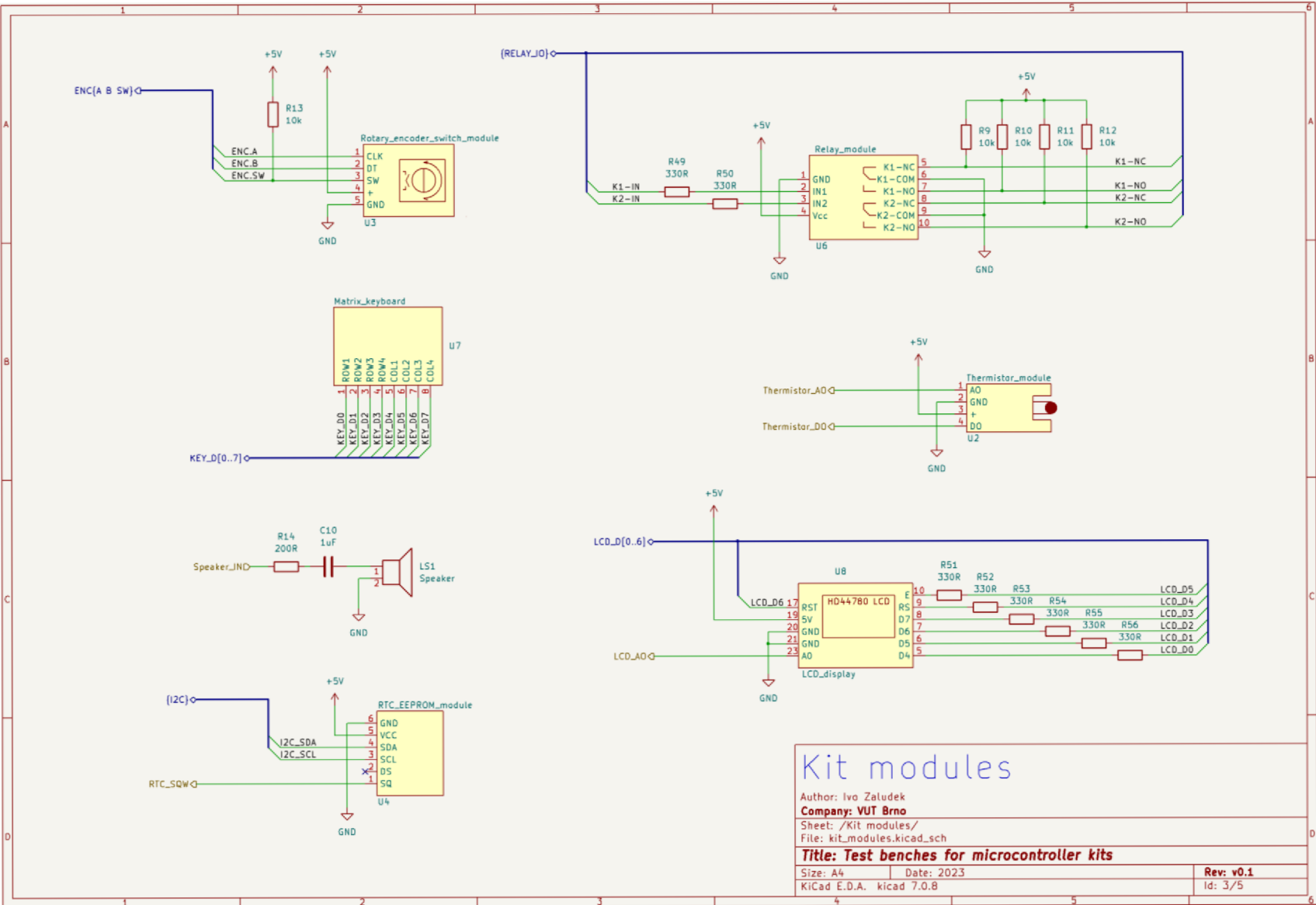
USB interface

Author: Ivo Zaludek
Company: VUT Brno
 Sheet: /USB Interface/
 File: usb_interface.kicad_sch

Title: Test benches for microcontroller kits

Size: A4	Date: 2023	Rev: v0.1
KiCad E.D.A. kicad 7.0.8		Id: 2/5

B.3 Zapojení testovaných modulů



Kit modules

Author: Ivo Zatludek

Company: VUT Brno

Sheet: /Kit modules/

File: kit_modules.kicad_sch

Title: Test benches for microcontroller kits

Size: A4

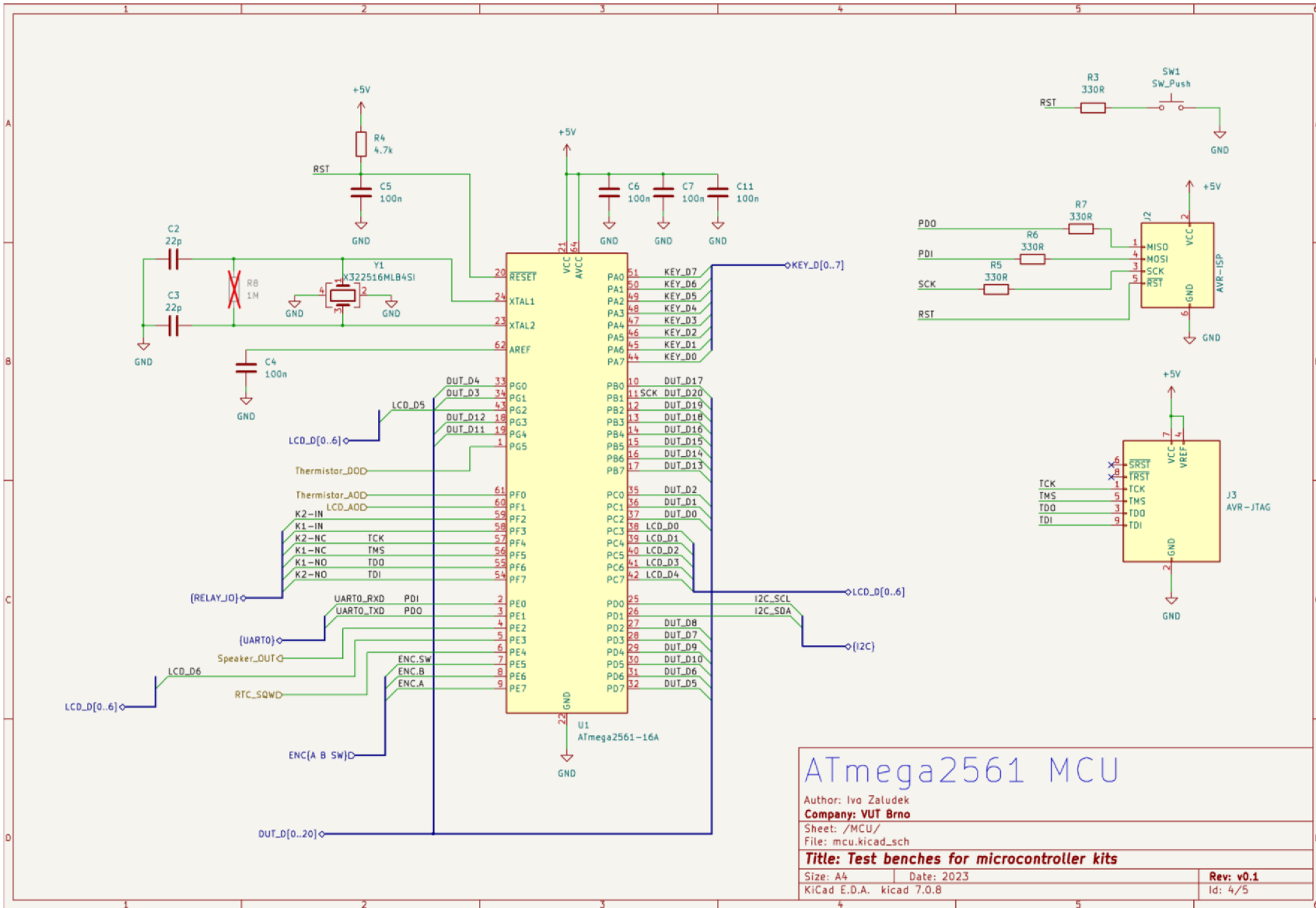
Date: 2023

Rev: v0.1

KiCad E.D.A. Kicad 7.0.8

Id: 3/5

B.4 Mikrokontroler ATmega2561



Příloha C - Obsah příloženého CD

Kapitola popisuje obsah přílohových souborů.

C.1 Projekt KiCad

Projekt v programu KiCad, obsahující obvodová schémata testovací stolice a desku plošných spojů.

C.2 Schémata ve formátu pdf

Obvodová schémata testovací stolice ve formátu pdf.

C.3 Výrobní podklady

Výrobní podklady testovací stolice exportované z KiCad projektu.

C.4 Modely pro 3D tisk

Modely mechanických dílů testovací stolice tisknutých na 3D tiskárně.

C.5 Firmware pro testovací stolici

Projekt firmwaru pro mikrokontroler ATmega2561, realizovaný ve frameworku Microchip Studio.

C.6 Firmware pro modul s ATmega328P

Projekt firmwaru pro testovaný modul ATmega328P Xplained Mini, realizovaný ve frameworku Microchip Studio.

C.7 Grafické uživatelské rozhraní

Kompletní program grafického uživatelského rozhraní v jazyce Python, včetně spustitelného (EXE) souboru.

C.8 Adresářová struktura

Organizace adresářů na přílohovém CD.

/Kořenový adresář přiloženého CD
testbench_KiCad_project.zipC.1 – Projekt KiCad
testbench_schematic/C.2 – Schémata ve formátu pdf
testbench_PCB_schematic.pdf	
testbench_fabrication_output/C.3 – Výrobní podklady
testbench_gerber_files.zip	
testbench-BOM-THT-mouser.xlsx	
testbench-BOM.xlsx	
testbench-CPL.xlsx	
testbench_3d_models/C.4 – Modely pro 3D tisk
encoder_clip.stl	
relay_clip.stl	
testbench_firmware/	
ATmega2561_test_program/C.5 – Firmware pro testovací stolici
atmega2561_project/	
main.c	
Config/	
...	
include/	
...	
src/	
...	
ATmega328P_test_prog/C.6 – Firmware pro modul s ATmega328P
ATmega328P_test_prog/	
main.c	
Debug/	
...	
Include/	
Header/	
...	
Src/	
...	
testbench_GUI/C.7 – Grafické uživatelské prostředí
example_test_report.csv	
GUI_pyinstaller_export.bat	
TestBenchGUI.exe	
main.py	
avrdude/	
...	
m328p_binary/	
...	
directory_structure.txtC.8 – Adresářová struktura