
**UNIVERSITATEA SAPIENTIA DIN CLUJ-NAPOCA
FACULTATEA DE ȘTIINȚE TEHNICE ȘI UMANISTE,
TÎRGU-MUREȘ
PROGRAMUL DE STUDII ...**

**TITLUL PROIECTULUI DE
DIPLOMĂ**

PROIECT DE DIPLOMĂ

Coordonator științific:
Ș.l.dr.ing. Turos László-Zsolt

Absolvent:
Lukács Botond

2022

| | | |
|------------------------------------------------------------------------------------------------------------------------------|-------------------------------|------------------|
| UNIVERSITATEA “SAPIENTIA” din CLUJ-NAPOCA Facultatea de Științe Tehnice și Umaniste din Târgu Mureș Specializarea: ... | | Viza facultății: |
| LUCRARE DE DIPLOMĂ | | |
| Coordonator științific: | Candidat: Anul absolvirii: | |
| a) Tema lucrării de licență: | | |
| b) Problemele principale tratate: | | |
| c) Desene obligatorii: | | |
| d) Softuri obligatorii: | | |
| e) Bibliografia recomandată: | | |
| f) Termene obligatorii de consultații: săptămânal | | |
| g) Locul și durata practicii: Universitatea Sapientia, Facultatea de Științe Tehnice și Umaniste din Târgu Mureș | | |
| Primit tema la data de: | | |
| Termen de predare: | | |
| Semnătura Director Departament | Semnătura coordonatorului | |
| Semnătura responsabilului programului de studiu | Semnătura candidatului | |

Declarație

Subsemnatul/a **Lukács Botond** , absolvent al specializării **Calculatoare**, promoția 2022 cunoscând prevederile Legii Educației Naționale 1/2011 și a Codului de etică și deontologie profesională a Universității Sapientia cu privire la furt intelectual declar pe propria răspundere că prezenta lucrare de licență/proiect de diplomă/disertație se bazează pe activitatea personală, cercetarea/proiectarea este efectuată de mine, informațiile și datele preluate din literatura de specialitate sunt citate în mod corespunzător.

Târgu Mureș,

Data:

Extras

Extract

Cuvinte cheie:

**SAPIENTIA ERDÉLYI MAGYAR
TUDOMÁNYEGYETEM
MAROSVÁSÁRHELYI KAR
SZÁMÍTÁSTECHNIKA SZAK**

Elektronikai alkatrész teszter

DIPLOMADOLGOZAT

Témavezető:
S.l.dr.ing. Turos László-Zsolt

Végzős hallgató:
Lukács Botond

2022

Kivonat

Napjainkban a mikrovezérlős rendszereken sok mindenben megtalálhatóak és manapság nagy számítási kapacitással rendelkeznek, általánosan alkalmazhatóak sokféle különböző alkalmazásban. Rengeteg funkciójuk van, az egyszerű LED villogtatástól kezdve komplex rendszerek automatizálásáig. Ezen kívül könnyű külső kiegészítő tartozékokat amelyekkel sokkal szélesebb körben használhatóak.

Ennek hatására a dolgozat célkitűzése egy olyan rendszer kialakítása, amely képes meghatározni egyszerű elektronikai komponenseket és azok megközelítő értékét és ezen kívül a lábkiosztását is amennyiben ez szükséges. Az elektronikában sok féle egyszerű komponenssel találkozhatunk, mint ellenállások, kondenzátorok, tranzisztorok. Viszont egy áramkör építésénél jó tudni, hogy az a komponens pontosan mi, ez legfőképpen igaz a különböző félvezetőkre. Sok esetben az azonosítója lekopott, vagy nem található adatlap így nehéz beazonosítani, hogy pontosan mi az a komponens. Erre szolgál az „elektronikai alkatrész teszter” amely automatikusan meghatározza, vagy kiírja, hogy hibás alkatrész ha nem ismeri fel vagy sérült az tesztelt alkatrész. A rendszer egy mikrovezérlőt és egy kijelzőt használ az komponens azonosítására és arról levő adatok kijelzésére a felhasználó felé. Az azonosítás teljesen automata, csupán csatlakoztatni kell az ismeretlen komponens és egy gombot megnyomni.

A dolgozatban a mikrovezérlős alkalmazásokról és azok tervezéséről, alkatrészek felismeréséről és méréséről lesz szó.

Kulcsszavak: mikrovezérlő

Abstract

Abstract

Keywords:

Tartalomjegyzék

| | |
|----------------------------------------------------------|-----------|
| 1. Bevezető | 1 |
| 1.1. Téma meghatározása | 2 |
| 2. Elméleti megalapozás és szakirodalmi tanulmány | 3 |
| 2.1. Elméleti alapok | 4 |
| 2.1.1. Mikrovezők | 4 |
| 2.1.2. Analog Digital Converter | 5 |
| 2.1.3. Digital Analog Converter | 6 |
| 2.2. Felhasznált eszközök | 7 |
| 2.2.1. Raspberry pi pico | 7 |
| 2.2.2. ILI9341 kijelző | 8 |
| 2.2.3. Digital Analog Converter | 9 |
| 2.3. A rendszer Blokk váza | 10 |
| 2.4. Hasonló eszközök | 10 |
| 3. Rendszer specifikációja és architektúrája | 13 |
| 4. Részletes tervezés | 14 |
| 5. Üzembe helyezés és kísérleti eredmények | 15 |
| 6. A rendszer felhasználása | 16 |
| 7. Következtetések | 17 |

Irodalomjegyzék**17****A. Függelék****19**

Ábrák jegyzéke

| | |
|-----------------------------------------------|----|
| 2.1. ADC offset hibája | 6 |
| 2.2. Pico pinout | 8 |
| 2.3. ILI9341 kijelző | 9 |
| 2.4. A rendszer block váza | 11 |
| 2.5. Eredeti teszter bekötési rajza | 11 |

1. fejezet

Bevezető

A mikrovezérlős rendszerek manapság az életünk minden részében megtalálhatóak, kis méretük, alacsony áruk és meglehetősen nagy teljesítményükkel sok mindenre általánosan használhatóak. Ez nagyban csökkenti a tervezési költségeket, mivel nem kell egy specifikus logikai áramkört kialakítani minden egyes alkalmazási területre, csupán új program kódot kell feltölteni és használható egy teljesen más célra.

A mikrovezérlő könnyen összeköthetőek külső eszközökkel amellyel rengeteg mindent meg lehet valósítani és bővíteni a lehetőség és szabadon vezérelhetők a GPIO-n (General Purpose Input Output) keresztül sok mindent el lehet érni, az egyszerű LED kapcsolgatásától komplex jelek generálásáig. Általános esetben a mikrovezérlők csak a legfontosabb részeket tartalmazzák, mint az Analog Digital Converter amivel egy analóg jelet alakít egy digitális jellé amit a processzor fel tud majd dolgozni, ez legfőképpen azért van, mert nem mindenkinek van szüksége mindenre így akinek szüksége van az egyszerűen az külsőleg csatolja hozzá.

Az összeállított rendszert szabadon lehet vezérelni így automatizálni lehet vele folyamatokat, mivel egyszerű megismételhető teszteket végezni velük, miközben képesek valós időben mérni a rendszer viselkedését. Ennek a feladatnak is ez a lényege, az alkatrészek tesztelése egy gyors és automatizált módon.

1.1. Téma meghatározása

A dolgozat célja egy olyan eszköz tervezése, amit bárki elektronikai ismeret nélkül is egyszerűen használni lehet. Sok esetben a feliratok az alkatrészeken nehezen látható, lekopott vagy egyszerűen nincs feltüntetve. Ilyen esetben sok segítséget tud nyújtani egy olyan eszköz ami gyorsan meg tudja határozni a komponens és a láb kiosztását is amennyiben ez fontos.

Ez különösen nagy segítséget nyújt kezdőknek akik még kevésbé ismerik az alkatrészeket és az adatlapjai meg nagyok és komplexek számukra és a fő információk megjelenítése néhány sorban. Haladóknak is nagy segítség, mivel az ellenállást színcódjáról egyszerű meghatározni, viszont a teszterrel meg lehet határozni, hogy az alkatrész hibás-e, vagyis ha a tranzisztor kiégett akkor az is letesztelhető.

A teszternek 3 teszt terminálja van, ebbe kell az ismeretlen komponens bekötni és képes egyszerű elektronikai komponensek (ellenállás, dióda, tranzisztorok, stb.) automatikus felismerését és az adatainak meghatározására. Viszont nem képes bonyolultabb áramkörök azonosítására aminek összesen több mint 3 lába van.

Megvalósítás során a költségek csökkentése a cél, miközben a pontosság nem csökken nagyban. Két verzió is összeállítható, az egyik egyszerű ellenállásokkal és a második egy DAC (Digital Analog Converter) segítségével. Mindkettő alkalmas a komponensek meghatározására, viszont az ellenállásos verzió nem alkalmas karakterisztika diagram kirajzolására, viszont sokkal olcsóbb, mivel nem használ egy külső DAC-ot.

Mérés eredménye kikerül egy kis kijelzőre és grafikus felületen is megtekinthető amennyiben egy számítógéphez van csatolva. Viszont a 2 közül legalább az egyikre szükség van, különben a mérés eredménye nem lesz látható. A kijelzőt nem kötelező alkalmazni, viszont annélkül csak egy laptop/-számítógéphez kapcsolva lehet használni.

Ehhez szükséges egy processzor, viszont manapság a mikrovezérlők nagy számítási kapacitással rendelkeznek meglehetősen alacsony áron és néhány ellenállásból és vezetékkel otthon is összeállítható.

Karakterisztika diagram kirajzolása is fontos, viszont ez leginkább a tranzisztoroknál fontos.

2. fejezet

Elméleti megalapozás és szakirodalmi tanulmány

Első lépésként tanulmányoztam egy hasonló terméket, amely hasonlóan működik, viszont kevesebb funkcionalitással. Ebből megismerve a működési elvét és ezt fejlesztve terveztem meg a tesztteremet. Legfontosabb része a tesztternek egy mikrovezérlő, amely a rendszer magját adja, erre a célra egy Raspberry Pi Pico-t választottam, a nagyszámú GPIO-ja miatt, nagy teljesítménye miatt és nagy sebességű beépített hardveres kommunikációs protokollokkal (SPI). Ezután egy kijelző következett, amelyeken a tesztter ki tudja jelezni az adatokat az adott komponensről.

Erre a célra egy ILI9341 kijelzőt használtam, ez egy 2.2” méretű színes TFT kijelző, és erre íródnak ki az adatok a felhasználó fele ami SPI-n keresztül kommunikál a mikrovezérlővel. Ezen kívül van 2 LED, amely a rendszer státuszát jelzi egyszerű színekkel.

Az teszteléshez szükséges áramkört a DAC segítségével történik és a mikrovezérlő csak a feszültség értékeket nézi, erre a célra egy DAC8565 chipet használtam amely szintén SPI-on keresztül kommunikál a mikrovezérlővel.

Ennek 3 kimenete egy-egy 3 kimenetes analóg kapcsolón keresztül különböző ellenállásokra kapcsolódnak a nagyobb precizitás elérése érdekében és az esetleges rövidzár esetén is az áramerősség biztonságos szinten tartásáért, minden esetben a létrehozott áramkörön lesz egy ellenállás, ami az áramerősséget limitálja, hogy esetlegesen ne tegye tönkre a tesztelés alatt levő alkatrészt.

A mikrovezérlőnek 3 ADC (analog-digital converter) portja közvetlen rá van csatlakoztatva egy láb

ahová majd a tesztelni kívánt komponens kerül. A rendszernek van egy külső referencia feszültsége, ami egy stabil 3.3V-ot biztosít az ADC referenciaként és DAC referenciának.

A pontos részleteket az a következőkben lesznek leírva.

2.1. Elméleti alapok

2.1.1. Mikrovezők

A mikrovezérlők már a elterjedtek a háztartási eszközök körében is. Legfőképpen az általánosságuk miatt, vagyis egyszerű különböző eszközöket vezérelni velük miközben az általánosságukból fakadóan alacsony az áruk.

Amíg régen egyedileg kellett a hardvert felállítani különböző eszközöknek az alapoktól kezdve, ami időigényes és költséges, mivel egy mérnök csapat meg kell tervezze, tesztelje és kivitelezze ami az alacsony darabszám miatt költséges egy darabra nézve. Ennek a hátránya ezen kívül még az, hogy nehezen módosítható és még hasonló eszközök között sem cserélhetőek. Viszont maga a rendszer hatékonyabb lehetett, mintha egy általános rendszerből lett volna kialakítva, ha a működés szempontjából van vizsgálva.

Manapság sokkal jobban megéri tömeggyártani egy mikroprocesszort ami általánosan használható rengeteg különböző eszközben, miközben a rendszerhez csupán hozzá kell csatolni a szenzorokat. Ehhez még nagy segítség a standardizált kommunikációs protokollok is a szenzorokkal, amely segítségével könnyedén összekapcsolhatóak a mikrovezérlővel, legtöbb esetben hardver szinten ismeri ezeket a protokollokat így gyorsan és hatékonyan képes kommunikálni. A leggyakrabban használt protokollok az I^2C és SPI. Viszont manapság sok mikrovezérlőn a WiFi és Bluetooth is megtalálható amivel vezeték nélkül is össze lehet kötni az eszközöket, ezek az eszközök leginkább az IoT-Internet of Things (Dolgok Internete) alapú rendszerek által használatosak.

Ezek a protokollok legtöbbször egyszerűek, így lehetséges szoftveresen is megvalósítani viszonylag egyszerűen a mikrovezérlő digitális kimeneteit használva. Viszont ezzel az a gond, hogy helyet foglal a memóriában, így nagy projektek esetén memória gondok léphetnek fel. A másik nagy gond, hogy ha időzítés érzékeny a protokoll akkor nehezebb betartani az időzítéseket, különösen ha nagy

frekvenciájú jeleket kell kiküldeni. Egy ilyen példa egy VGA jel generálása, a szinkronizáló jelek előre meghatározott időben és csúszások nélkül kell megérkezzen, miközben pontos időben kell kikerüljön az adat is. Ez nehéz feladat teljes mértékben szoftveres megoldással megvalósítani.

Egyes mikrovezérlők ezért lehetővé tették a programozható be/kimeneteket. Ennek a lényege, hogy néhány kimenetet egyszer felprogramozva a processzortól függetlenül működik, mintha egy hardveres megvalósítás lenne, ezzel meg sokkal egyszerűbb egy nem alapértelmezetten támogatott protokoll megvalósítása, miközben a processzor nincs leterhelve ezzel a feladattal.

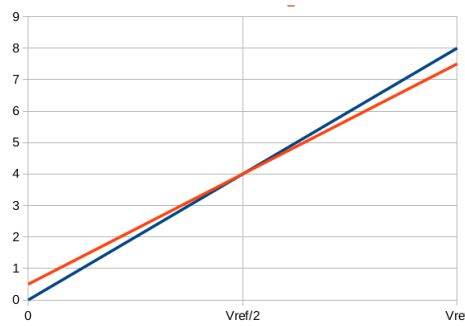
2.1.2. Analog Digital Converter

Az Analog Digital Converter (ADC) feladata, hogy egy analóg jelet digitális jellé alakítson át, mivel a processzor csak digitális jeleket képes feldolgozni. Viszont 1 bitben nem használható, mivel szinte az összes adat elveszne, ezért az ADC több biten tárolja el az analóg mérés eredményét. Ez legtöbbször 8-16 bit közti értékek közt található. A maximális érték amit az ADC eredménynek kiad az akkor történik, amikor az analóg jel feszültsége megegyezik az ADC referencia feszültség szintjével és 0 értéket ad ha a jel feszültsége 0V és a két érték között lineáris összefüggés van amely meredeksége 1. Erre több megoldás is létezik, viszont mindenik ugyan azt a célt éri el, megállapítja, hogy melyik 2 érték közé esik a feszültség szint.

Viszont ez a valóságban nem ilyen egyszerű, mivel a komponensek amik az ADC-t felépítik nem tökéletesek így zajok és csúszások jelenhetnek meg. Ilyen hibák lehetnek az offset hibák, ebben az esetben a végértékek nem felelnek meg az elvárt értékeknek, ilyen példa a amikor 0V-os feszültség esetén az ADC nem 0-t térít vissza, hanem egy nagyobb számot, míg V_{ref} esetén egy kisebb értéket.

Az alábbi ábrán egy ilyen eset látható [2.1], a kék vonal az elvárt érték és a piros az aktuális érték, amit az ADC térít vissza a mérési tartományban. Ebben az esetben csupán ez a hibája az ADC-nek az átláthatóság kedvéért.

Másik jellegzetes hiba a nem linearitás, ennek jellemzője, hogy a mérési tartományon belül még az offset hibát is figyelembe véve nem azt az eredményt kapjuk, mint amit elvártunk amelyet nehezebb korrigálni szoftveresen, mivel amíg az offset hibát egyszerűen lehet korrigálni csupán a 0 és V_{ref} feszültség szintek megméréseivel, addig ezt a hibát sokkal nehezebb kiküszöbölni, mivel az ADC ered-



2.1. ábra. ADC offset hibája

ménye nem lineáris, így több ismert mérési pontot kell felvenni és a számítás is sokkal bonyolultabbá válik. Ezen kívül lehetnek kvantálási hibák is, amikor az ADC egy bizonyos értéket sosem ad eredménynek, hanem a nála egyel kisebb/nagyobb értéket. Ez nem okoz nagy gondot azoknál az ADC-nél ahol nagy a felbontás, viszont az alacsony felbontású eszközök esetén nagyobb hibát eredményez.

2.1.3. Digital Analog Converter

A Digital Analog Converter feladata, hogy a processzor által értelmezhető digitális értékeket analóg jelekké konvertálja amik a külső perifériáknak szükségesek. Gyakran használják ahol pontos analóg jeleket kell vezérelni, ilyen egy VGA jel vagy zene lejátszása, viszont bármire felhasználhatóak ahol az egyszerű digitális jelek nem elegendőek. Ebben a projektben az alkatrészek azonosítására és a karakterisztika diagram kirajzolására van felhasználva.

Működése hasonlóképpen működik, mind az Analog Digital Converter-é, viszont fordítva. Egyes DAC-eknek van belső referencia feszültségük, vagy kell külsőleg egyet alkalmazni. Ez a jel lesz DAC maximális analóg kimenete amikor a digitális bemeneten a maximális érték érkezik és 0V lesz a kimenete amikor 0 értékű digitális érték érkezik. Ideális esetben e két pont között az átmenet lineáris, így minden köztes értékre egy lineáris összefüggés alapján ki lehetne számolni [2.1], ahol D a digitális bemenet ami 0 és 2^{res} között kell legyen.

$$V_{out} = V_{ref} \frac{D}{2^{res}} \quad (2.1)$$

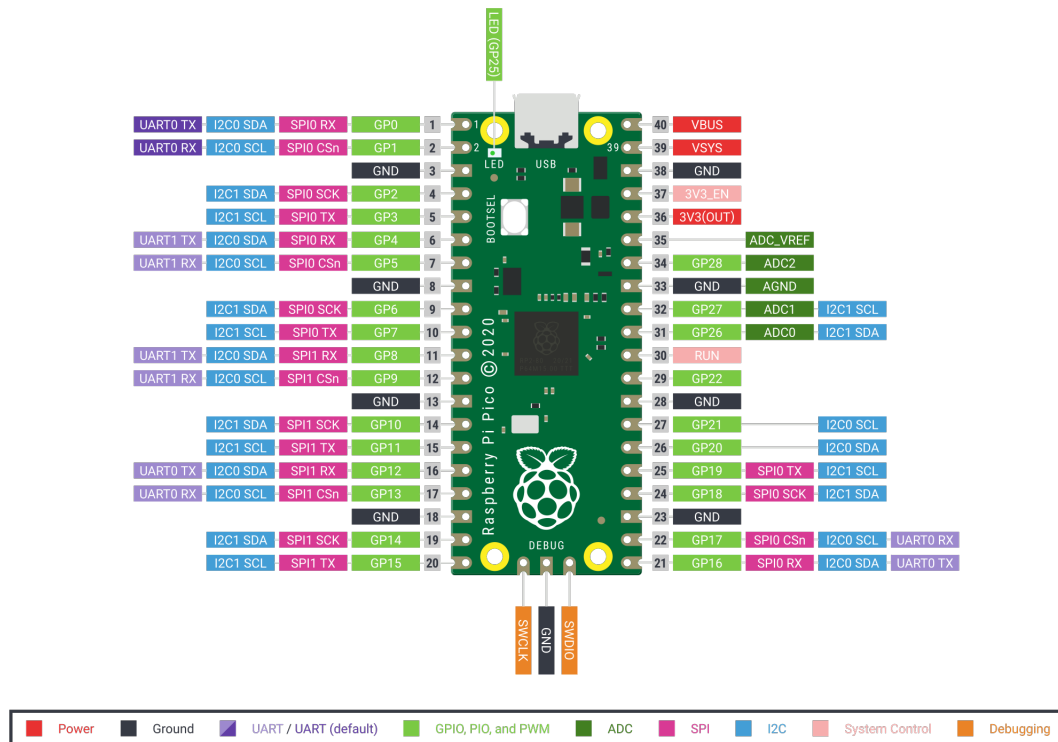
2.2. Felhasznált eszközök

2.2.1. Raspberry pi pico

Ez a teszter legfontosabb eleme [2.2], ez a vezéri a teszter többi részét. A Raspberry által kifejlesztve, viszont nem, mint a többi termékük ezen nem fut egy operációs rendszer és használható, mint egy számítógép, hanem mint egy hagyományos mikrovezérlő. A mikrovezérlő magját egy Dual-core Arm Cortex M0+ processzor, aminek a frekvenciáját dinamikusan változtatható 16Mhz és akár 280Mhz közt. A memóriája is elegendő 264KB of SRAM a változók tárolására a program futása során és 2MB Flash memória a program kód tárolására. Lehetőség van csak RAM-ból futtatni is a kódot, ilyenkor nagyobb órajel is elérhető ami elérheti a 450Mhz-et is.

A vezérlő úgy van megtervezve, hogy THC (Through Hole Component) és SMT (Surface Mount Component) komponensként is lehet alkalmazni, ezt a castellated pinnekel éri el, amelyeket lehet forrasztani közvetlen SMT komponensként és a pinek távolsága annyi, mint egy átlagos breadboard sortávja így egy male-male header forrasztásával breadboardon is alkalmazható.

Hardveresen megtalálható 2xSPI, 2xI2C, 2xUART, 3x12-bit ADC, 16xszabályozható PWM csatorna és összesen 26 GPIO van kivezetve a felhasználó fele. Egy pin több dologra is képes ez az alábbi pinout diagrammon is látható. Programozás során beállítható, hogy USB-t használja soros portként így egyszerűen használható egy számítógéppel való kommunikációra. Megtalálható 8 PIO (programmable I/O) amivel hardveres protokollokat lehet létrehozni, ha valami egyedi kell egy különleges komponens vezérlésére.



2.2. ábra. Pico pinout

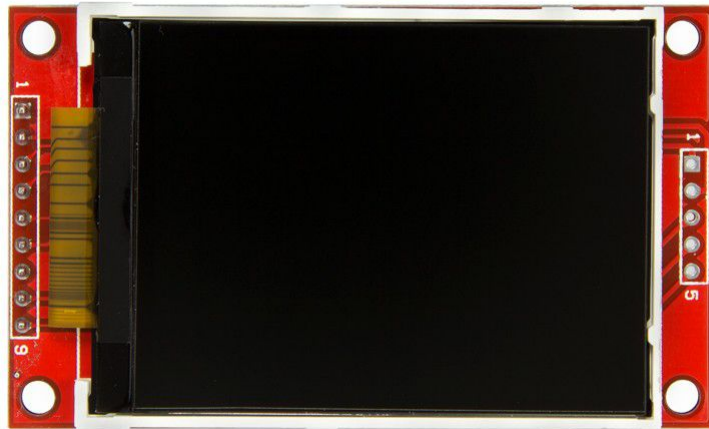
2.2.2. ILI9341 kijelző

Az adatok kijelzésére egy ILI9341 [2.3] kijelzőt használtam. Egy egyszerűbb kijelző is megfelelne a célra, akár csupán egy karaktereket megjelenítő kijelző is elegendő lenne a célra, ha nem kellene a karakterisztika diagramm. A célra megfelelne egy egyszerű OLED kijelző is. Viszont ez a kijelző már a rendelkezésre állt a projekt tervezése előtt is, így fel lehetett használni a kijelzőt erre a projektre.

Ezen fognak megjelenni az mérés eredményei, mint a komponens értékei, a lábkiosztása és a karakterisztika diagramja. A képernyő képes 16 bites színes kép megjelenítésére, viszont ez nem fontos ennél a projektnél. Viszont a nagy 240*320 as felbontása és a 2.2" kijelző mérete nagyban segít, hogy könnyen olvasható legyen. A kijelző rendelkezik beépített háttérvilágítással is, így fényes helyen is jól látható. A kijelző SPI protokollon keresztül kommunikál a mikrovezérlővel és rendkívül stabil, akár 68Mhz-es órajellel is stabilan és megbízhatóan megkapja az adatokat a mikrovezérlőtől. Ennek az eredménye a rendkívül gyors képernyő váltás, így a felhasználó nem kell arra várjon, hogy

az adatok megjelenjenek a kijelzőn.

Az adatok beírása pixel szinten történik amelyek a kijelző memóriájában elmentődnek, előre meg kell határozni, hogy milyen zónában történik az írás és azután lehetséges csak az adat küldése és belsőleg automatikusan a következő mezőt címzi meg, ahogy megérkezett az adat az előzőnek. Ennek az eredménye, hogy írás során nem kell figyelembe tartani a kijelző belső memória címét, csak az adat sorrendjét. Ez egyszerűsíti a programozó dolgát és csökkenti a küldési időt.



2.3. ábra. ILI9341 kijelző

2.2.3. Digital Analog Converter

Az analóg jelek generálására egy DAC8565IAPWR [[1]] van alkalmazva, amelynek felbontása 16 bit. Ennek 4 egymástól független kimenete van, amelyek belsőleg bufferelve vannak, így terhelés esetén sem változik.

A DAC-nak van egy belső 2.5V-os feszültség referenciája, viszont ez külsőleg felülírható egy külső feszültség referenciával. Ebben az esetben szükség volt erre, mivel a lejes ADC mérési tartományra szükség van ami 3.3V. Ez egy egyszerű 3.3V-os feszültség referenciával lett elérve, viszont ebben az esetben egy parancsot kell küldeni a DAC-nak, hogy a külső referenciát használja, mivel nem áll át automatikusan, rövid időre nem gond, hogy mindkettő aktív, viszont nem ajánlott huzamosabb ideig ilyen helyzetben tartani.

A DAC SPI protokollt használ amiből 8 bit a parancs bit és 16 bit az adat. Lehetőség van az adatok azonnali betöltésére és a kimenet változtatására, és arra is van lehetőség, hogy csak az adatot elmentse és egyszerre frissítse az összes kimenetet. Erre lehetőség van szoftveresen is, amit az SPI parancs részében lehet kódolni, ebben a projekt esetében is ez van alkalmazva.

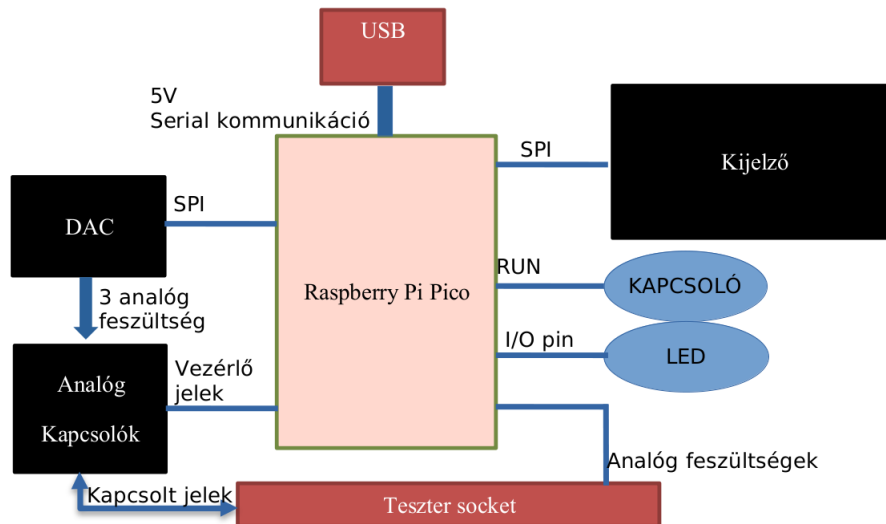
2.3. A rendszer Blokk váza

Az alábbi ábrán [2.4] látható a rendszer blockDiagrammja, a fő vezérlő jelekkel. A rendszer 2 SPI perifériával kommunikál, ezek a kijelző és a DAC, az analóg kapcsoló vezérlő jelei egyszerű digitális jelek, mint ahogyan a LED vezérlő jelei. A teszter socket mindhárom lába egy-egy ADC csatornához van csatlakoztatva, ezen méri a mikrovezérlő a tesztelés során a feszültség szinteket. A kapcsoló a RUN bemenetre van kapcsolva, ez engedélyezi a mikrovezérlő futását amíg feszültség érték található rajta vagy nincs bekötve. Amennyiben ez földre kerül akkor a mikrovezérlő leáll és resetelődik.

A rendszerher működéséhez szükséges egy 5V-os feszültség forrás, ez lehet egy általános USB tápforrás, vagy lehetséges direkt 5V csatlakoztatása is. Az áramerősség alacsony, így nem közelíti meg az 500mA-es áramerősség határt amit egy átlagos USB képes leadni. Külső akkumulátorról is táplálható. Amennyiben egy számítógéphez, vagy egy olyan eszközhöz van csatolva ami képes a serial portot olvasni akkor a mérési eredményeket automatikusan elküldi azon keresztül a másik eszköz felé.

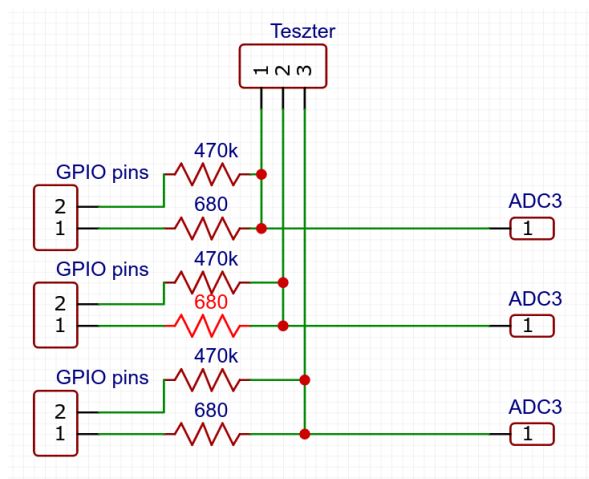
2.4. Hasonló eszközök

Egy hasonló rendszer már megvalósult [2] ami egy Arduino[3] mikroprocesszoron alapul. Ez egy egyszerűbb rendszer, ami csupán ellenállásokat használ az alkatrészek felismerésére és egy kijelzőt az adatok megjelenítésére. Ebből több fejlesztés is kialakult, több minden tesztelésére és nagyobb pontosság elérésére, miközben a rendszer egyszerűségét fenntartani. Ezek a rendszerek viszont nem használnak DAC-ot és ezért nem képesek karakterisztika diagramot készíteni. Ezen kívül nem csatlakoztatható egyszerűen számítógéphez, csupán újraprogramozás céljából, így minden esetben kell tartalmazzanak egy kijelzőt, ami növeli a költségeket. Az általános működésük hasonló, mint ebben a projektben, viszont itt precízen lehet változtatni a feszültséget, nem csak kapcsolni földre vagy



2.4. ábra. A rendszer block váza

tápfeszültségre. Az bekötése hasonlóképpen történik: lásd [2.5]



2.5. ábra. Eredeti teszter bekötési rajza

Mindegyik GPIO pin lehet csatolva földre, vagy tápfeszültségre, de le is lehet kapcsolva, így nem befolyásolja az áramkör működését. Az ADC pin meg lehet ADC üzemmódban, ilyenkor nem befolyásolja az áramkört, viszont lehet földre, vagy tápfeszültségre is kapcsolni, ilyen esetben port ellenállás nélkül csatolódik az áramkörre.

Viszont szintén hordozható egy 9V-os elem segítségével és az eredmények megjelennek egy kis

LCD kijelzőn, ebből több féle verzió is létezik, van amely csak egy karakter kijelzőt használ, van amelyik egy színes kép kirajzolására is alkalmas kijelzőt alkalmaz.

A tesztelés néhány másodpercbe telik, nagy méretű kondenzátorok esetén telhet több időbe, viszont ebben az esetben csak annyi idő, míg a legkisebb ellenálláson keresztül képes feltölteni a kondenzátort.

3. fejezet

Rendszer specifikációja és architektúrája

4. fejezet

Részletes tervezés

5. fejezet

Üzembe helyezés és kísérleti eredmények

6. fejezet

A rendszer felhasználása

7. fejezet

Következtetések

Irodalomjegyzék

- [1] „DAC.” <https://www.ti.com/cn/lit/ds/symlink/dac8565.pdf?ts=1654348067115>. [Online 2022].
- [2] M. Frejek, „AVR Transistortester.” https://www.mikrocontroller.net/articles/AVR_Transistortester. [Online 2022].
- [3] „Arduino Atmega328p datasheet.” https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf. [Online 2022].

A. függelék

Függelék