

Hibakeresési és felülvizsgálati megoldások gázturbinás hajtóművek vezérlőinek fejlesztéséhez

TDK Dolgozat

Wagner Dominik

2023.11.08

1. A TKT-1 gázturbinás sugárhajtómű

1.1. Leírás

A TKT-1 a Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedés és Járműmérnöki Karának Repüléstudományi és Hajózási Tanszékén fejlesztett kísérleti gázturbinás sugárhajtómű. Az AE laborban található eszköz egy szovjet TSz-21 indító gázturbina átalakításával lett kialakítva.

A hajtómű elsődleges feladata a hallgatók számára a hajtóműtechnikai elvek és rendszerek demonstrációja, illetve a mérések/laborgyakorlatok lehetővé tétele. Ezen felül fontos másodlagos feladat a tanszéken belüli tanulmányok és tudományos kutató munka lehetővé tétele.

A dolgozat szempontjából fontos, hogy a TKT-1 teljes hatáskörű digitális hajtóműszabályzó rendszerrel (FADEC) van ellátva. Ez a FADEC egység a Teljes hatáskörű Duplikált Digitális elektronikus szabályozórendszer, rövidítve TEDDI.

2. A TEDDI hajtóművezérlő számítógép

2.1. Leírás

A TEDDI egy moduláris, opcionálisan kétcsatornás vezérlésre tervezett dedikált hajtóművezérlő mikroszámítógép. Az eszköz teljesen egyedi tervek alapján készült részben standard alkatrészekből, saját tervezésű áramköri lapokkal.

A számítógép egy ütésálló műanyag műszerdobozban lett elhelyezve. A doboz oldalfalai szükség esetén karbantartásra, vagy megfigyelésre eltávolíthatóak. A burkolaton szellőzőrács található, így a számítógép idegen anyagoktól nem védett. Ezen felül nem került kialakításra elektromágneses zavarástól való védelem sem, így minden egységnek erről magának kell gondoskodnia.

A doboz egyik végén 4 darab csatlakozó található, csatornánként 2. Ezek az összekötőelemek gyorsan oldhatóak, a karbantartás és hibakeresés megkönnyítése végett. Mind a két csatlakozópár egy 50 tűs, főképp adatcsatlakozóból és egy 20 tűs, főként tápellátást és nagyobb áramigényű aktorok vezérlőjeleit hordozó csatlakozóból áll. Normál üzem közben a számítógépnek nincs szüksége ezen a 2 (2 csatornás mód esetén 4) csatlakozón kívül semmilyen más összeköttetésre.

A FADEC kártyás rendszerű, különböző funkciók külön kártyákon kapnak helyet. Ez megkönnyíti a karbantartást és fejlesztések esetén az új alkatrészek modulárisan beépíthetőek. A kártyák ISA konnektorokkal egy közös alaplapon keresztül csatlakoznak egymáshoz. Az alaplapon semmilyen logika nem található, az egyedül az egységek közötti kapcsolat megteremtéséért felel. A be-és-kimenetek szokatlan módon nem az alaplagra, hanem az egyes kártyák megadott portjaira vannak bekötve, így a közös adatbuszba kinti információ nem kerül.

Fontos megjegyezni, hogy a TEDDI fejlesztése hosszú és komplex folyamat. A számítógép valamilyen formában már 2009 óta létezik. Emiatt a rendszerek egy része kifejezetten régi, több ember dolgozott a projekten, sokszor nem is ismerték egymást. Helyenként a szabványok nem lettek betartva, néhol pedig olyan alkatrészek lettek használva, amik már akkor sem feleltek meg a kor színvonalának. Adjuk ehhez hozzá a számítógép prototípus jellegét és látható, hogy bármi újat fejleszteni az egységre nagyon komplex és rengeteg akadályon kell átugrani.

2.2. Feladatok

A TEDDI legfőbb feladata a TKT-1 gázturbina beérkező parancsok szerinti vezérlése biztonsági paramétereken belül. Ezt megvalósítandó megfigyelésre kerül a hajtómű összes fontosabb hőmérséklete és nyomása, a hajtómű fordulatszáma, az olaj- és tüzelőanyag-rendszerek néhány paramétere, illetve természetesen a műszerfalon található kezelőszervek.

A számítógép másodlagos, ámde mégis fontos feladata a hibakeresés és a vészhelyzetek elhárítása. Ennek érdekében található az egységben dedikált hardver az önellenőrzési és hibaelhárítási funkciók megvalósítására.

A fentiek felül még nem küldetéskritikus feladat az üzemi paraméterek rögzítése az egyetemi gyakorlatok lebonyolításának és a turbina egyszerűbb üzemben tartásának érdekében.

2.3. Alkotóelemek

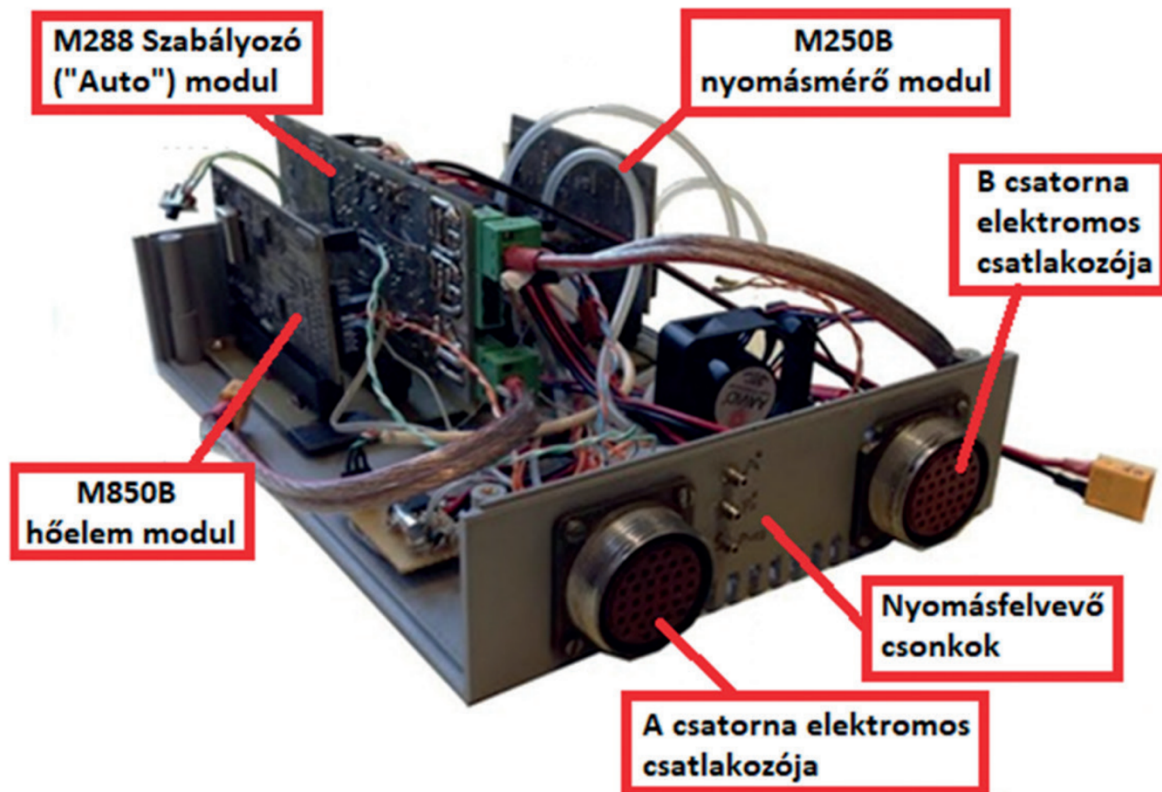


fig. 2.3.1. A TEDDI hajtóművezérlő számítógép

A TEDDI mikroszámítógép a következő fontosabb alkotóelemeket tartalmazza:

2.3.1. Alaplap

A TEDDI alaplapja egy saját tervezésű, egy oldalas, üvegszálerősítésű műanyag nyomtatott áramköri lap. 6 darab 8-bites IBM ISA (Industry Standard Architecture) szabványú kártyaél csatlakozó foglalattal rendelkezik. A csatlakozók azonos pin-jei passzívan össze vannak kötve, 2 kivétellel. Ez a 2 pin csak egy-egy oldal 3 csatlakozóját köti össze és arra szolgál, hogy a kártya tudja érzékelni, hogy melyik „csatornába” lett behelyezve, így a viselkedését tudja ilyen módon igazítani. Az alaplap 4 csavarral van a házhoz rögzítve és a 6 ISA konnektoron kívül semmilyen csatlakozót nem tudhat magáénak.

A modulok közötti kommunikáció a fent említett pin-eken keresztül már egyre kevesebb analóg, nagyobb részt digitális jelekkel történik. A hosszú távú cél az összes jel digitálisra cserélése. Az adathálózat gerincét egy 197.78Kbit/s adatátviteli sebességre képes CAN-busz

adja. Ezen a buszon kerülnek megosztásra másodpercenként 50-szer a kártyák által a hajtómű legfontosabb adatai.

2.3.2. M250

Az M250 a TEDDI nyomások értékeléséért felelős beépülő kártyája. A nevét a 250KPa maximum üzemi érzékelhető nyomásról kapta. Ez a saját tervezésű ISA konnektorral csatlakozó kétoldalas üvegszálerősítésű műanyag nyomtatott áramköri lap a rendszer egyetlen eszköze, hogy a turbina nyomásait érzékelje.

Az M250-hez tartozik ezen felül egy másodlagos PCB, amin a nyomás szenzorok jelentős része található. Ez a fő kártyához egy túsoron keresztül csatlakozik.

A modul a korábbi verziókban parancsra, a későbbi verziókban belső időzítés alapján diszkrét időnként mintát vesz a nyomásszenzorokból és a mért nyomásokat a CAN buszon keresztül továbbítja a többi kártyának.

2.3.3. M850

Az M850 a számítógép hőmérséklet érzékelő modulja. Ez a saját tervezésű kétoldalú üvegszálerősítésű műanyagból készült nyomtatott áramköri lap egy ISA élcsatlakozón keresztül csatlakozik az alaplaphoz. A legfőbb feladata a hajtómű alkatrészein található hőelemek megfigyelésre, mért értékeik továbbítása a CAN buszon. A nevét a hőelemek által érzékelhető 850C° maximális érzékelhető hőmérséklet után kapta.

Működése hasonló az M250 kártyához, diszkrét időnként a hőmérsékleteket tartalmazó csomagot a CAN buszon a többi kártyával megosztja.

2.3.4. AUTO (M288)

Az AUTO a hajtómű automatikus vezérlőegysége. Ez a saját tervezésű, kétoldalas, üvegszálerősített műanyagból készült nyomtatott áramköri lap lényegében a rendszer lelke. Ez a kártya az ISA konnektor pin-jein, a CAN buszon keresztül az M250 és M850 mikrokontrollerein, valamint a saját bemenetein keresztül folyamatosan információt gyűjt a hajtómű és a kezelőszervek állapotáról. Egy belső órajel alapján diszkrét időközönként a legutolsó beérkezett adatok alapján kiszámolja a vezérlés kimeneti paramétereit és ezeket továbbítja a megfelelő végrehajtóegységeknek.

Fontos megjegyezni, hogy a többi modullal szöges ellentétben az AUTO kártyán nem csak félvezetők találhatók, jónéhány relé és kontaktor kapott helyet, így a kártya mérete

lényegesen nagyobb a többinél. A vezérlésének nagy részét ezekkel az elektromechanikus aktorokkal végzi.

Az eredeti terv szerint a TEDDI-ben 2 AUTO kártya kapott volna helyet de jelenleg csak egy van beépítve, így a FADEC csak egycsatornás módban tud üzemelni. A TEDDI moduláris kialakításának köszönhetően bármikor beépítésre kerülhetne egy azonos kártya és ezt követően a számítógép azonnal képes lenne a kétszatornás vezérlésre. Ez jelenleg azért nem valósul meg, mert csak egy autó modul készült el. A kártyák megkülönböztetéséért és hierarchiába rendeléséért a konnektor azonosító pin-ek felelnek.

2.3.5. BITE-M

A BITE-M egy beépülő kártya, ami a rendszer felülvizsgálataért és teszteléséért felelős, innen származik nevének első fele, BITE (Built-In Test Equipment). A nevének második felében az M betű azt hivatott jelezni, hogy ebbe a modulba van beépítve hajtómű vészhelyzeti, teljesen manuális szabályzásához szükséges hardver.

A történet egyetlen szépséghibája azonban, hogy a BITE-M habár a sugárhajtómű szakutasításában erősen szerepel, a valóságban hol papíron, hol erős prototípusállapotban létezett csak. A többi alkatrész úgy van tervezve, hogy a vezérlési ciklus megvalósításához semmilyen módon ne támaszkodjon erre a kártyára így ez az elem kihagyható, és jelen állapotban ki is van hagyva, ezzel nyilvánvalóan feláldozva a biztonsági funkciók jelentős részét, illetve a manuális vezérlés lehetőségét

2.3.6. Egyéb alkatrészek

A felsoroltakon kívül még az egység részét képezi a házként szolgáló műanyag műszerdoboz, illetve a különböző konnektorok.

2.4. CAN busz

A TEDDI alaplap legfontosabb kommunikációs csatornája a CAN (Controller Area Network) busz. Az eredeti elképzelés szerint a kártyák egy az AUTO kártyán diszkrét időpontokban generált központi jelet érzékelve sorban adták volna ki az adataikat, de az ehhez kapcsolódó időzítési rendszer jelenleg nem működik, és pillanatnyilag a kártyák saját belső időzítés alapján adott frekvenciával osztják meg az információkat. Ennek megfelelően az AUTO kártya nem az „általa megrendelt” információkat használja fel, hanem az utolsó buszon megjelent információt tárolja, és a vezérlési kimenet generálásának időpontjában azt tekinti

legfrissebbek. Ez a rendszer, habár szuboptimális, a hozzávetőlegesen 50Hz frekvenciának köszönhetően elég gyors frissítést kínál, hogy az adatokat valós idejűnek lehessen tekinteni.

A 2.1. bekezdésben említett prototípus jelenleg a CAN buszra is kiterjed. A busz baud rátája nem konformál semmilyen standardhoz, 197.78Kbit/s. Emiatt a tesztelése nehézkes, jobb esetben a használt logikai analizáló eszköz részletes beállítása szükséges, rosszabb esetben a használt eszköz nem is képes erre a feladatra. Mivel a frekvencia kisebb mint 250Mbit/s, a busz a CAN standard szerint nem számít nek, így a szabvány szerint megkövetelt 120Ω ellenállások nem szükségesek a busz lezárásához, ennek ellenére néhol megtalálhatók, néhol nem.

Mivel a CAN standard nem tartalmaz eszköz azonosítókat, Ezért csak a különböző üzeneteknek vannak azonosítóik. A CAN arbitrációs logika miatt a kisebb azonosítószámú üzenetek lesznek prioritizálva. Ennek megfelelően a legkisebb azonosítók a legfontosabb adatokhoz lettek rendelve, főleg azokhoz, amik elsődleges paraméterek a hajtómű vezérlésében. Csak 2 kártya, az M850 és az M250 képes üzemszerűen CAN üzenetek küldésére. A rendszer 7 különböző üzenetet ismer, mindegyik üzenet 8 bájtos, 2 byte-onként egy adattal. A számok 16 bites fixpontos ábrázolással kerülnek küldésre, így elvárt, hogy a fogadó mikrokontroller ismerje a számok értelmezésének módját.

Jelenleg a buszon rengeteg fel nem használt küldési idő van, a sáv szélesség koránt sincs kihasználva. Ennek nyomán sokszor szóba került már a monitorozó és szervíz funkciók ellátására történő adoptálás. Példája ennek, hogy a jelenleg tervezés alatt lévő M850 utód a hideg ponti kalibrációját a tervek szerint már nem szervíz kábellel, UART-on, hanem távolról a CAN busz segítségével fogja elvégezni.

2.5. Hibakeresés

A TEDDI hosszú fejlesztése alatt védhetetlenül rengeteg hibát vétett a csapat. Emiatt a ki és beérkező jeleket rendszeresen meg kell tekinteni, a kártyák belső logikáját felül kell vizsgálni, a gázturbina állapotát meghatározandó pedig sokszor futás alatt feljegyzett adatokat le kell menteni. Erre az évek alatt sok megoldás született.

Hatalmas segítséget jelent az UART. Majdnem minden mikrokontroller, ami a számítógépben található, rendelkezik egy ilyen csatlakozóval. Példának okáért a jelenleg beépített M250 verzió alapvető adatainak lekérését és változtatását egy ilyen konzolon végezzük.

Az évek során voltak már próbálkozások a CAN busz külső megfigyelésére többkevesebb sikerrel. Jelenleg nincsen olyan kész eszköz megépítve, ami megbízhatóan különösebb előzetes konfiguráció nélkül az adatfolyam közvetlen külső megfigyelését tenné lehetővé.

A bemenő információk tesztelésében az UART parancsokon kívül jelentős szerepet játszottak a próbalap alapú jelgenerátorok. Ezek a nehezen konfigurálható és megkérdőjelezhető megbízhatóságú szerkezetek nyújtották a hibák üzemén kívüli keresésének legegyszerűbb módját. Ezen felül használva voltak a mikrokontrollerekbe előre programozott tesztek is, de nyilván ezek a módosítása, illetve aktiválása/deaktiválása a mikrokontroller újra programozását vonta magával.

Fontos hangsúlyozni, hogy a fent elhangzott tulajdonságok egyike se előnyös. Habár klasszikusan a hardver fejlesztés elsődleges fókuszja a végtermék, esetünkben a különböző beépülő modulok, ezeknek a fejlesztése nagyságrendekkel komplexebb lehet megfelelő teszt felszerelés nélkül. Hab a tortán, hogy jellegéből adódóan sok ember, aki dolgozik a projekten még tanul, így a szakmai képességeik limitáltak, és a teszt adatokat nekik még nehezebb beszerezni, mint egy képzését már befejezett szakembernek. Saját tapasztalatból tudom mondani, hogy az igazat megvallva archaikus hardvert nehéz navigálni és még nehezebb teszt adatokat kihozni belőle.

Az elégedetlenség több fél felől is akkor tetőzött, amikor múlt novemberben egy mérés után a külső SD kártya modult ellenőrizve azt tapasztaltuk, hogy az adatok nem lettek normálisan rögzítve, és ezt követően nem voltak értelmezhetőek. Ez az alkalom egybeesik a gázturbina utolsó tesztüzemével. Mint utólag kiderült, a mai napig nem teljesen megoldott gyújtási hiba volt a háttérben, és az okok feltárását sokkal nehezebbé tette, hogy az összes rögzített adatunk megbízhatatlan volt.

2.6. A TEMEC projekt

Ezt követően kerestem meg Dr. Beneda Károly tanár urat azzal, hogy tervezni kívánok egy beágyazott tesztelő és megfigyelő berendezést, ami képes ellenőrizni a sugárhajtóművet, az üzemi paramétereket megbízhatóan rögzíteni és számunkra a belső adatfolyamokhoz kényelmes hozzáférést nyújtani.

A projekt a TEMEC (TEDDI Embedded Monitoring and Evaluation Computer) nevet kapta. A cél egy olyan eszközt fejleszteni, ami tesztüzemben a FADEC üzemi rendszereit nem

zavarva a számítógépen belül elhelyezhető, távolról hozzáférhető, és később más feladatokra adaptálható.

Ezen rendszer fejlesztésének első lépései, az alapvető kutatás, a rendszer központi elemeinek megtervezése, a komponensek tesztelése és az első prototípus legyártása remélhetőleg ennek a TDK dolgozatnak a keretein belül zajlik majd le.

3. A TEMEC projekt célkitűzései

3.1. Alapfeladat

A TEMEC a tervek szerint az alaplap egyik ISA csatlakozójába fog majd illeszkedni. Feladata a fent említett buszon a fontosabb adatok megfigyelése, beleértve legfőképp a két CAN vonalat. Kell rendelkeznie olyan képességgel, amivel az adatokat maradandóan és könnyen megszerzhető módon lokálisan tárolni tudja. Képesnek kell lennie ezen felül könnyen kapcsolatot teremteni a külvilággal és a funkcióit gyorsan újra programozhatóvá kell tenni. Fontos emellett, hogy ne zavarjon semmilyen már elkészült rendszert és ne kelljen a működéséhez a jelenlegi hardveren komoly változtatásokat végezni.

Repüléstechnikáról lévén szó fontos a megbízhatóság és még fontosabb. Emiatt a rendszernek képesnek kell lennie önmagát diagnosztizálni és amint zavaró tényezővé válik magát a célrendszerből kizárni. A TEDDI lassan 15 éve fejlesztés alatt áll. Nem elképzelhetetlen, hogy 10 év múlva még mindig dolgozni fognak rajta, ezért olyan hardvert kell építeni, ami képes hosszú ideig megbízhatóan működni és egy jelenleg is modern eszközkészletet használ, hogy biztosak lehessünk abban, hogy az üzemeltetéshez szükséges technikai tudás 10 év múlva még fennáll.

Konstrukciójában tartós, iparban is használt technológiákat kell alkalmazni. Ahol lehet szem előtt kell tartani a szabványokat, hogy a jövőben könnyű legyen az eszköz működését felülvizsgálni és funkcionalitását bővíteni.

Nem utolsó szempont, hogy a gázturbina egy egyetemen található, így nyilván a működése gyakran van hallgatóknak demonstrálva. Így jó lenne, ha lehetne beépíteni funkcionalitást, ami segítene a tanároknak a hajtómű technika szemléltetésében és egy interaktív ezt közt kínálna az érdeklődőknek a kísérletezésre és kutatásra. Mindezt olyan biztonsági háttérrel ellátni, ami nyomán sem az eszközök, sem a kutatók kívánók érdemi kárt nem szenvedhetnek.

3.2. Geometriai/anyagtechnikai követelmények

A TEMEC modul egy TEDD beépülő kártya formátumában legyen elkészítve. A hajtómű vezérlőhöz kártya széli ISA csatlakozóval kapcsolódjon, külső támaszt ne igényeljen. Ez garantálni fogja a jelenlegi konstrukcióval való hosszútávú kompatibilitást, és mivel nem igényel extra csatlakozókat, ezért a ki-be szerelése igény esetén könnyen megoldható.

A mérete legyen minimális, nem nagyobb a jelenlegi kártyáknál. A modul legyen elég keskeny ahhoz, hogy bármelyik oldalára egy másik beépülő modult el lehessen helyezni. A műszerdoboz belseje így is szűkös, a kis méret garantálja, hogy a jövőbeni modulok geometriája nem fogja a felhasználókat kényszeríteni a TEMEC modul eltávolítására.

A modul nyomtatott áramköri lapra készüljön, vastagsága az ISA standardnak megfelelően 1,6mm. A PCB felületi kezelésben részesüljön. A konstrukcióba ólom ne kerüljön, a választott alkatrészek RoHS kompatibilitási listán szerepeljenek. Az előbbi pont nekem kifejezetten fontos, ugyanis véleményem szerint nem etikus közoktatási intézményben elkerülhető helyeken veszélyes anyagokat használni.

3.3. Hardveres követelmények

A modul képes legyen az ISA konnektorban elérhető 24V feszültségről a saját tápfeszültségeit előállítani, a TEDDI-nek elérhető erőforrások keretein belül.

A megfigyelési feladatokat egy modern programozható számítástechnikai eszköz, innentől CDP (Central Data Processor) végezze, ennek szükséges perifériái a kártyán kerüljenek elhelyezésre. A CDP egység legyen távolról elérhető, üzemi feltételek mellett vezeték nélkül, vészhelyzet esetén szervíz csatlakozóval.

A kártya működését egy dedikált megfigyelő egység végezze. Ez az egység képes legyen egy vészhelyzet automatikus felismerésére és ha szükséges a veszélyes faktorokat a rendszerből eltávolítani. További biztonsági kritérium, hogy a gázturbina kezelő pultján kerüljön elhelyezésre valamilyen eszköz, amivel a főbb rendszerek működésétől teljesen függetlenül egy kialakuló vészhelyzet elhárítható.

Az egység kialakításából adódóan ne keltsen zavaró jeleket a többi kártyával közösen használt kommunikációs csatornákon, ugyanis ez eddig nem tapasztalt viselkedést okozhat a minimális tesztelésnek alávetett másik modulokon és nem utolsó sorban a saját mért eredményt is meghamisíthatja.

A kártyán kapjon helyet hardveres adatrögzítő eszköz, tápellátásfüggetlen memóriával, és ez az adathordozó szükség esetén legyen könnyen eltávolítható, külső számítógépen egyszerű eszközökkel olvasható.

A lehető legtöbb alkatrész legyen standard, a kártya méretét minimalizálandó, ami lehetséges készüljön SMD technológiával.

3.4. Szoftveres követelmények

A TEMEC egység legyen mérési, adatfeljegyzési és tesztelési képességeit tekintve könnyen és gyorsan újraprogramozható. Ez legyen megoldható optimális esetben vezeték nélkül, végszükség esetén szervíz kábellel. A kártya biztonsági felügyelő rendszerei is legyenek szükség esetén újraprogramozhatóak programozó csatlakozón keresztül.

A szoftver tegye lehetővé több folyamat párhuzamos futását, a kimeneti bemeneti folyamatok a felhasználók részéről szándékos vagy véletlen veszélyeztetéstől legyenek védve.

Legyen olyan folyamat, ami érzékeli a sugárhajtómű elindítását és passzívan a felhasználó bemenete nélkül kezdje el az üzemi adatokat rögzíteni. Az így készülő naplófájlok normál üzem mellett legyenek vezeték nélküli kapcsolattal elérhetőek, szükség esetén egy PC számítógéppel a fedélzeti adathordozóról kiolvashatóak.

Készüljön olyan felület, amin a hallgatók működés közben a hajtómű adatait vezeték nélkül eléri a lehető legnagyobb kompatibilitással.

4. Hardver

4.1. Mikrovezérlő, vagy egykártyás számítógép

Szerintem ebben a tervezési feladatban a legfontosabb kérdés a CDP megfelelő hardverének kiválasztása. Itt szerintem két komolyabb választási lehetőség jöhet szóba, egy modern, nagy teljesítményű 32-bites mikrokontroller vagy egy beágyazott egykártyás számítógép (Single Board Computer, SBC).

4.1.1. Mikrokontroller

Kapható jelenleg a piacon számos nagyon-nagy teljesítményű, modern 32-bites mikrokontroller. Például vessünk egy pillantást az STM32H7x7 mikrokontroller családra. Ezek a kétmagos, ARM Cortex alapú, 480MHz maximum órajellel rendelkező mikrovezérlők számos beépített perifériával rendelkeznek, például ADC és DAC átalakítókkal, SPI (Serial

System SMPS, LDO, USB and backup regulators POR/PDR/PVD/BOR Multi-power domains Xtal oscillators 32 kHz + 4 ~48 MHz Internal RC oscillators 32 kHz + 4, 48 & 64 MHz 3x PLL Clock control RTC/AWU 1x SysTick timer 2x watchdogs (independent and window) 82/114/140/168 I/Os Cyclic redundancy check (CRC) Unique ID	Chrom-ART AcceleratorTM JPEG Codec Acceleration Cache I/D 16+16 Kbytes Arm[®] Cortex[®]-M7 480 MHz + Arm[®] Cortex[®]-M4 240 MHz	2-Mbyte dual-bank Flash memory RAM 1056 Kbytes incl. 64 Kbytes ITCM FMC/SRAM/NOR/NAND/SDRAM Dual-mode Quad-SPI 1024-byte + 4-Kbyte backup SRAM
Control 2x 16-bit motor control PWM synchronized AC timer 10x 16-bit timers 2x 32-bit timers 5x Low-power timer 16-bit high-resolution timer	Floating point unit (DP-FPU) Nested vector interrupt controller (NVIC) JTAG/SW debug/ETM Memory Protection Unit (MPU) ROP, PC-ROP anti-tamper	Connectivity TFI LCD controller MPI-DSI HDMI-CEC 6x SPI, 3x I2S, 4x I2C Camera interface Ethernet MAC 10/100 with IEEE 1588 MDIO slave 2x FDCAN (Flexible Data rate) 1x USB 2.0 OTG FS/HS 1x USB 2.0 OTG FS 2x SDMMC 4x USART + 4 UART LIN, smartcard, IrDA, modem control 1x Low-power UART 4x SAI (Serial audio interface) SPDIF input x4 DFSDM (8 inputs/4 filters) SWP (Single Wire Protocol)
Crypto/Hash processor 3DES, AES 256, GCM, CCM SHA-1, SHA-256, MD5, HMAC Security services SFI and SB-SFU	AXI and Multi-AHB bus matrix 4x DMA True random number generator (TRNG)	Analog 2x 12-bit, 2-channel DACs 3 x 16-bit ADC (up to 3.6 Msps) 20 channels/up to 2 MSPS Temperature sensor 2x COMP 2x Op amp

fig. 4.1.1.1. Az STM32H757X szürreális beépített periféria készlete

Forrás: st.com

Peripheral Interface), CAN és I²C perifériákkal, de akár komplexebb DSI csatlakozási lehetőségekkel és kriptográfiai modulokkal.

Hátrányuk viszont, hogy ha ezt a lehetőséget választom, akkor kénytelen vagyok ezeknek a mikrokontrollereknek a magas tervezési igényeiknek eleget tenni. Ugyanis egy ilyen komplex eszköznél fontos PCB szigorú szabványokra való tervezése, az elektromágneses zavarás minimalizálása, illetve hasonlóan komplex perifériák beépítése.

Nagy előnye ennek a megoldásnak a nagyon kicsi mérete és a kifejezetten beágyazott eszközökhöz igazított áramfelvételi karakterisztikája. Az előbbiből következik viszont, hogy egy ilyen chip-nek a kifejezetten apró lábkiosztása, vagy az esetleges BGA kivitel miatt a leforrasztása a gyakorlatlan kéznek komplex feladatot, és ezzel nem vagyok benne biztos, hogy meg tudnék komoly gyakorlás nélkül birkózni. Ha pedig én nem tudok vele megbirkózni, akkor

nem elképzelhetetlen, hogy akinek a jövőben feladata lesz ezen eszköz javítása vagy módosítása szintén nem fog rendelkezni a megfelelő képességekkel.

Ezek a mikrokontrollerek rendelkeznek a Bluetooth és WiFi konnektivitás beépítéséhez szükséges perifériákkal, viszont az antennákat és a szükséges driver modulokat nekünk kell utólag hozzáadni. Az ezeket a összekötő kommunikációs csatornák további tervezésbeli nehézségeket okozhatnak.

Hátránya még ennek a megoldásnak, hogy az ilyen típusú hardver általában nincs felkészítve egy általános operációs rendszer futtatására. Habár a párhuzamos feladatok időzítése egy valós idejű operációs rendszerrel megoldható lenne, a programozhatóság jelentősen sérülne, ugyanis a mikrokontrollert vagy csak külső programozóval, vagy esetleg egy a firmware-t tároló memória integrált áramkörrel, és annak távoli újraprogramozásával lehetne frissíteni.

4.1.2. SBC

A másik optimális jelölt a CDP szerepére egy modern SBC (Single Board Computer – egykártyás számítógép) lenne. Ezek olyan miniatürizált mikroszámítógépek, amik egy mikroprocesszort és az összes hozzá tartozó perifériát egy, általában nagyon kicsi áramkörtől lapra helyeznek.

Ennek a konstrukciónak nagy előnye, hogy az összes érzékeny periféria már a gyárban leforrasztásra kerül, így az összeszerelés alatt ezekből jelentős probléma nem származhat. Nyilvánvalóan ennek a hátulütője, hogy sokkal kevésbé rugalmasak a kommunikációs csatornáink a CPU-val, így meg kell elégednünk azzal, amit a gyártó elérhetővé tesz. Viszont az asztali számítógépekkel ellentétben itt legtöbbször sokkal több alacsony szintű periféria elérhető. Az egyszerűbb kimeneti séma annyiban megkönnyíti a dolgomat, hogy az ezt az egységet befogadó PCB tervezési igénye lényegesen kisebb, mint amire az előbbi pontban említett mikrovezérlőnek szüksége lett volna.

Nyilvánvaló hátulütő, hogy a sok fedélzeti periféria miatt ezek az egységek sokkal nagyobbak, mint a mikrovezérlő társaik. A külvilághoz is legtöbbször asztali számítógépeken is használt konnektorokkal, vagy egyszerű tűsorokkal csatlakoznak. A sok bizonyos feladatokhoz felesleges periféria a fogyasztásnál sem segít. Egy ilyen egykártyás számítógép akár nagyságrendekkel több áramot vehet fel, mint az egy szekcióval ezelőtt vitatott mikrovezérlők.

SZÁMOK!!!!

Hatalmas előny, hogy ezek a számítógépek legtöbbször Linux támogatással érkeznek. Így a felhasználónak a kezdetektől hozzáférése van egy robusztus operációs rendszerhez kiváló támogatással. A perifériákhoz is a legtöbb esetben a könyvtárak nem a gyártók kegyességétől függenek, hanem a közösség már írt a megfelelő feladatra célkódot.

Nem szabad elfelejteni, hogy ezek a számítógépek sokszor modern ARM alapú több magos CPU-kkal büszkélkedhetnek. Nem ritkák a több gigahertzes órajelek és a modern enkódolási/dekódolási hardvereszközök sem. Így ezek az eszközök sokkal nagyobb teljesítményt nyújthatnak, mint a feljebb említett társaik.

A vezeték nélküli konnektivitás eléréséhez kiváló segítséget nyújt, az, hogy az SBC-k általában már rendelkeznek valamilyen vezeték nélküli kommunikációs rendszerrel. Így a hardvertervezési feladatot megkerülve elegendő csak a megfelelő driverekkel megismerkedni.

Komoly árnyoldala viszont ennek a megoldásnak, hogy a hardver és szoftver általában nincs annyira felkészítve küldetéskritikus applikációkhoz, mint egy mikrovezérlőnél, így a kiberbiztonságra és a lehetséges veszélyhelyzetekre jobban kell gondolni.

4.1.3. Következtetés

Ennek a projektnek a megvalósításához egy SBC-t fogok választani. Meggyőző faktor a kisebb tervezési komplexitás, a könnyebben elérhető támogatás és a nagyobb teljesítmény. A szerteágazó és a potenciális jövőbeli hallgatók limitált programozási tudásából adódó optimalizálatlan kód futtatásában valószínűleg nagy segítséget nyújt majd a jelentős extra teljesítmény.

Nem is beszélve arról, hogy pár híresebb egykártyás számítógép sokszor kerül ajánlásra kezdő programozóknak és hardver fejlesztőknek, így lehetséges, hogy a jövőbeli karbantartók már érkezésükkor ismeretesebbek lesznek a rendszer sajátosságaival.

Viszont a hátrányokra is gondolva oda kell majd figyelni a lehetséges csökkent biztonságra és az esetlegesen kialakuló veszélyhelyzetek érzékelésére és elhárítására.

4.2. Raspberry Pi zero 2 W

Megfelelő körbetekintést követően a választás a Raspberry Pi zero 2 W egykártyás számítógépre esett.



fig. 4.2.1 Raspberry Pi zero 2 W

Forrás: <https://www.raspberrypi.com/products/raspberry-pi-zero-2-w/>

A Raspberry Pi Foundation által gyártott SBC 65mm*30mm-es méretével játszi könnyedséggel elfér a TEDDI-ben elérhető szűkös helyen. A 4 darab M2.5-ös rögzítési pont segítségével a TEMEC kártya áramkörti lapjához könnyen rögzíthető, a 40 tűs csatlakozójával pedig megfelelően képes a többi alkatrészsel kommunikálni.

A gyártó cég által tervezett SoC (System-on-Chip – Rendszer a chip-en) 4 modern ARM magot és 512MB rendszer memóriát hordoz magában. A külvilág eléréséhez található a kártyán egy vezeték nélküli modem, ha pedig egyéb fix csatlakozások kellenek elérhető 2 micro-USB, az egyik opcionálisan tápellátásra is, illetve videó kimenet is megvalósítható egy micro-HDMI csatlakozó segítségével.

A számítógép háttértáraként egy felhasználó által hozzáadott SD-kártya szolgál, amire az operációs rendszer is telepítendő. Ez az SD-kártya később másik számítógépeken is olvasható, így könnyű elveszettnek vélt adatokat visszahozni. Az előbb említett operációs rendszer a Raspberry Pi OS, elérhető asztali környezettel és szimpla konzollal is, utóbbi minimális erőforrást igényel. Ez a szoftver a Debian Linux egy erősen modifikált változata.

Specification

1GHz quad-core 64-bit Arm Cortex-A53 CPU
512MB SDRAM
2.4GHz 802.11 b/g/n wireless LAN
Bluetooth 4.2, Bluetooth Low Energy (BLE), onboard antenna
Mini HDMI® port and micro USB On-The-Go (OTG) port
microSD card slot
CSI-2 camera connector
HAT-compatible 40-pin header footprint (unpopulated)
H.264, MPEG-4 decode (1080p30); H.264 encode (1080p30)
OpenGL ES 1.1, 2.0 graphics
Micro USB power
Composite video and reset pins via solder test points
65mm x 30mm

fig. 4.2.2. Raspberry Pi zero 2 W specifikációk

Forrás: <https://www.raspberrypi.com/products/raspberry-pi-zero-2-w/>

Nem szabad elfelejteni, hogy a világon több millió Raspberry Pi van forgalomban, így az elérhető támogatás a platformhoz meglehetősen magas. Ez nemcsak nekem, hanem az esetleges jövőbeli karbantartónak is nagyon jó hír. A termék támogatottságát hangsúlyozza, hogy a gyártó ígéretet tett arra, hogy 2028-ig gyártásban maradnak ezek az egységek, így 5 éves élettartammal számolva akár 2033-ban is fogunk tudni még találkozni ilyen számítógépekkel. Ez nagyon jó hír abban az esetben, ha esetleg hiba miatt pótolni kell.

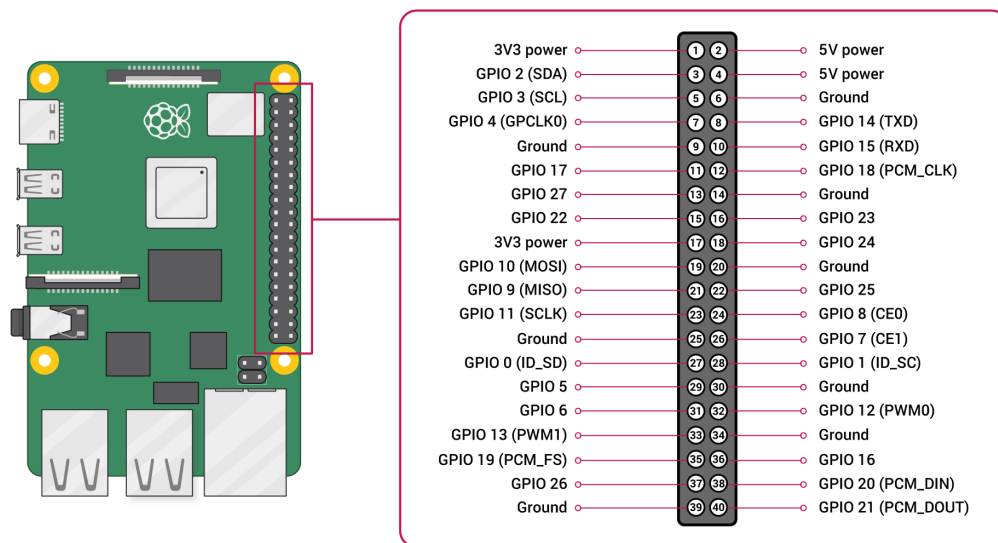


fig. 4.2.3.. Raspberry Pi pinout, ábrázolt modelltől függetlenül mindegyik számítógépen ugyan az.

Forrás: raspberrypi.com/documentation

4.3. CAN interfész

A szemfüles olvasónak feltűnhetett, hogy a CDP szerepét betöltő egykártyás mikroszámítógép 40 tűs csatlakozója nem rendelkezik CAN funkcionalitással. Ez a tervezési szempontból érthető, habár nekem előnytelen tény nem jelenti azt, hogy nem tudjuk ezt a funkcionalitást utólag hozzáadni.

Az SBC rendelkezik SPI és I²C perifériákkal, így a megfelelő integrált áramkör használatával könnyedén hozzá tudunk adni CAN perifériát. Az MCP2515 egy a Microchip cég által gyártott, viszonylag ismert SPI-CAN interfész integrált áramkör. Az 1MBit/s maximum sebességével és az általa támogatott standardok széles listájával bőven megfelel a jelen feladatra.

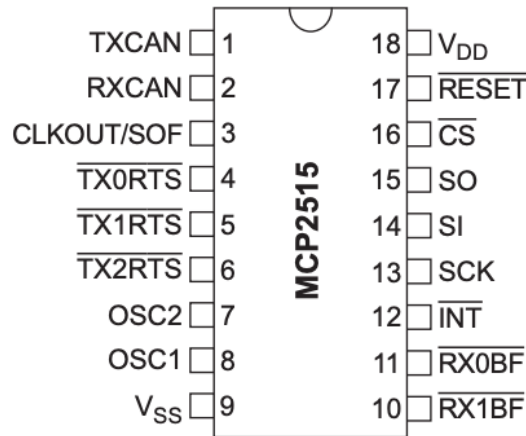


fig. 4.3.1. MCP2515 lábkiosztás

Forrás: MCP2515 adatlap

A működéséhez szüksége van az egyértelmű SPI-csatlakozáson kívül egy 8-20MHz-s kristályoszillátorra és a hozzátartozó kondenzátorokra, 3,3-5V tápfeszültségre és reset logikára. Fontos megjegyezni, hogy direkt a CAN busszal nem tud kommunikálni, így szüksége van egy CAN adó-vevőre.

A beérkező és küldeni kívánt információt átmenetileg belső regiszterekben tárolja, az SPI buszon keresztül ennek írására-olvasására nyílik lehetőség. A hozzá kötött periféria figyelmeztetésére kapott egy interrupt pin-t, illetve a belső regiszterek opcionálisan további pin-eken manipulálhatók.

Az áramköri lapok tervezésénél fontos szempont a kondenzátorok és a kristály elhelyezése, mert egy 20MHz-es órajelet a helytelen vezető geometria, vagy a túl hosszú vezetők már jelentősen befolyásolnak.

4.4. Biztonságtechnikai problémák a CDP-vel

Egyszerű belátni, hogy a 4.3.-ban említett MCP2515 csak és kizárólag a CDP befolyása alá tartozik. Mivel az SPI busz jelenlegi felhasználási módjában nem rendelkezik multi-master móddal, és ha ez nem lenne így, akkor is az IC-hez történő hozzáférésben a CDP kooperációjától függünk. Így nyilvánvaló, hogyha a beérkező adatokat manipulálni nem lehet, a legegyszerűbb megoldás az, ha a kimenő adatokat figyeljük, és nem várt történés esetén a CAN IC-k kapcsolatát a TEDDI-vel megszakítjuk.

Ezt és még egyéb biztonságkritikus változókat, például a hajtómű üzemi adatait, vagy a kártyába érkező feszültséget egy másik programozható vezérlőnek kell figyelnie. Itt nem lesz

annyira komplex a választás, mint az előző kérdésnél tökéletesen megfelel egy megbízható közép kategóriás 32-bites mikrokontroller.

Erre a szerepre saját tapasztalataim miatt az STM32F103 mikrokontrollert fogom választani, azon egyszerű indokból, hogy már tudom programozni ezt az egységet. Itt viszont a programozásnak szerepet kell kapnia a biztonság kritikus gondolkodásnak, illetve a támogató rendszereket, mint például a tápellátást el kell a CDP perifériáitól különíteni.

Ez a mikrokontroller ezentúl a BIT-MC, avagy Built-In-Test-MicroController (beépített ellenőrző mikrokontroller) névre fog hallgatni.

A BIT-MC szintén képesnek kell, hogy legyen a CAN busz olvasására, itt viszont annyival könnyebb a dolgom, hogy ez a képesség a választott mikrokontrollerbe már be van építve, így csak egy adóvevőt kell rákötni. Ezen felül ez a chip rá lesz még kötve különböző ellenállás-osztókkal és műveleti erősítőkkal az ISA busz különböző pin-jeire, hogy hajtómű vezérlő számítógép más fontos funkcióit is meg tudja majd figyelni.

A CDP-vel ez a mikrokontroller képes kommunikálni, de csak az előbbi I²C (Inter-Integrated Circuit) buszán keresztül. Fontos itt megjegyezni, hogy a master itt a CDP számítógép lesz, így a BIT-MC csak akkor tud üzenetet küldeni, ha a CDP engedélyezi. Azért nem probléma ez, mert ezen a csatornán csak tájékoztató információk kerülnek megosztásra, az üzemi feladatát, avagy hibás információ küldése esetén a CDP ellehetetlenítését a BIT-MC így is el tudja végezni, mivel a megszakító körébe a mikroszámítógép nem tud beleszólni.

4.5. Elektromechanikus/elektronikus biztonság

Itt fontos leszögezni, hogy a követelményekben megállapított módon mindenféleképp be kell építenünk egy alternatív rendszert, ami ezt a biztonsági megszakítást el tudja végezni. Így mindenképp egy komplex rendszerben lenne érdemes gondolkodni, ami mind a két feladatot egy lépcsőben ellátja. Feltétel még, hogy a vészhelyzeti rendszer ne legyen könnyen befolyásolható a kártya áramköri elemei által, azoknak esetleges hibája ne legyen kihatással a biztonsági rendszer azon képességére, hogy egy vészhelyzetet el tudjon hárítani.

Ennek megfelelően az alternatív rendszer esetében mindenképp elektromechanikus kivitelben gondolkozok, mert habár egy régebbi és kevésbé rugalmas technológia, a repülésben széles körben alkalmazzák a magas biztonsága miatt. Így célszerű lenne az adatfolyam megszakítására egy relét bevetni. Komoly probléma viszont, hogy nem optimális reléken adatot

átvinni, mert a kontaktok esetleges instabilitása komoly károsodást okozhat a küldött adatfolyam integritásában.

Ezt kiküszöbölendő, mivel a gyártó tesztjei alapján, a műszaki adatlapon indikált adatokból egyértelműen kiderül, hogy egy nulla feszültség alatt lévő MCP2515 kommunikációs jelek szempontjából szigetel, így elég lenne az IC tápfeszültségét megszakítani. Annak érdekében, hogy failsafe kialakítást érjünk el, a tápegység feszültség alatt lévő tekerecs mellett fog tudni működni, így elég a szolenoid áramát megszakítani, és a CAN IC kapcsolata a külvilággal azonnal megszakad.

Ha a relé árama akármilyen más okból, esetleg hiba miatt megszakad, a kommunikációs kapcsolat ugyanúgy bontásra kerül. Az elsődleges biztonsági rendszerhez, egyszerűen beépítettünk a vészkapcsolóval sorosan egy MOSFET-et, Így szükség esetén akár az ellenőrző szoftver is meg tudja szakítani a kapcsolatot. A kialakítás miatt akármelyik egység aktiválja a vészmegszakítást, az mindenképp végbemegy. Az elsődleges megszakító MOSFET-et egy másik ellenőrző egység irányíthatja.

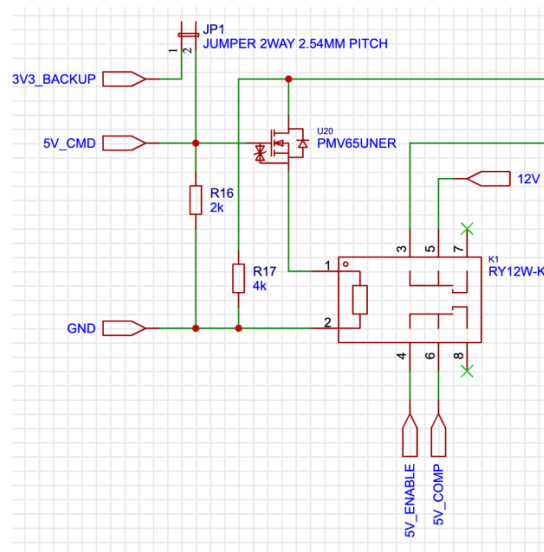


fig. 4.5.1. A relé bekötésének áramköri diagramja. Az 5V_ENABLE jel pont fordítva viselkedik a fent leírtakhoz képest, ugyanis a valós tápegység magas jelre kapcsol ki.

A sugárhajtómű kezelőjének való egyszerű visszajelzés érdekében a relé alaphelyzetben zárt kontaktjára egy a vészleoldó gombra épített LED lesz kötve, így ha a tekercs ejt, és ennek nyomán a CAN kommunikáció elveszti a tápfeszültségét az irányítópulton azonnal egy figyelmeztető fény ki fog gyulladni, ezzel jelezve a kapcsolat megszakadását.



fig. 4.5.2. Már beépített vészleállító gomb, a kétnyelvű matricájával.

Forrás: Saját fotó

A CDP-nek nem szükséges hardveres visszajelzést adni, vannak egyéb szoftveres módszereink arra, hogy az SPI buszon keresztül ezt az eseményt érzékeljük.

4.6. Valós idejű óra

Mivel a rendszer másodlagos funkciója adatrögzítés, ezért fontos az idő ismerete. A CDP nincs ellátva a valós idejű órával, avagy Real-Time Clock (RTC)-al, így a pontos időt külső forrásból kell megszerezni. Ezt a feladatot ellátandó hozzáadásra került egy Microchip MCP79402 valósid idejű óra integrált áramkör. Ez az IC egy órakristály segítségével követni tudja a pontos időt. Azért, hogy tápellátás elvesztése esetén az idő továbbra is számolva legyen a TEMEC kártyán el lett helyezve egy CR2032 típusú gomelemet fogadni képes foglalat. Ez kikapcsolt TEDDI esetén is körülbelül 3V áramot fog tudni az RTC IC-nek.

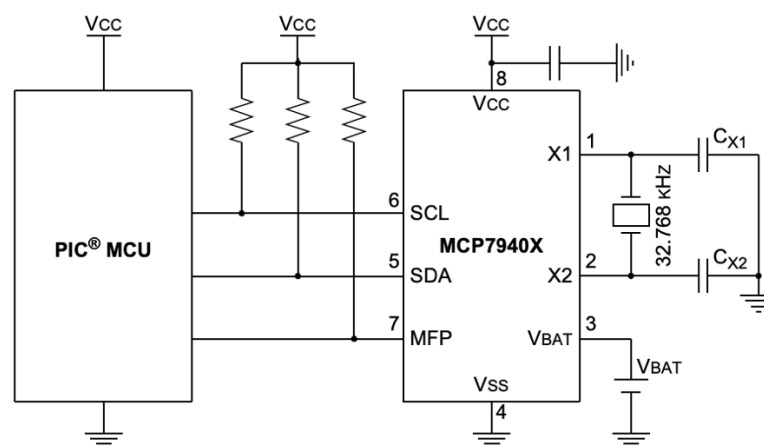


fig. 4.6.1. MCP79402 tipikus alkalmazási rajz

Forrás: MCP7940X adatlap

A CDP számítógép szükség esetén az I²C buszon keresztül fel tudja venni a kapcsolatot az RTC IC-vel, és ki tudja deríteni a pontos időt. Az integrált áramkör az év-hónap-nap adatokat is követi, így lehetőség nyílik pontos idő bélyegzők használatára. Az RTC beállítása a CDP-n keresztül történik.

4.7. Másodlagos csatlakozások

Habár üzemi feltételek mellett optimális esetben csak vezeték nélküli kapcsolatra lesz szükség a kártyával, fent kell tartani szükség esetére fizikai kommunikációs csatornákat. Ezt kiszolgáló, felhasználok a Raspberry Pi zero 2 W beépített UART port-ját, és elérhetővé teszek egy tty(tele-type) interfészt, hogy szükség esetén be lehessen lépni szervíz kábel segítségével a mikroszámítógépbe.

Ezen felül a CAN vészkapcsolónak is csatlakoznia kell a kártyához, ez pedig egy 6 eres RJ25 csatlakozón keresztül fog megtörténni, ami a kialakításának köszönhetően szükség esetén gyorsan eltávolítható, ugyanakkor az RJ25 csatlakozó megfelelő védelmet nyújt a kapcsolat véletlen bontása ellen.

Továbbiakban még a BIT-MC-hez kerül egy ST-LINK programozó csatlakozó, illetve egy UART az esetleges hibakereséshez.

4.8. Tápegységek

A beépülő kártyának alaphelyzetben 3 feszültségre van szüksége:

- 12V: Az elektromechanikus biztonsági rendszernek, illetve köztes lépcsőként a többi tápegységnek.
- 5V: A CAN eszközöknek, illetve a CDP-nek
- 3,3V: Az RTC IC-nek, illetve a BIT-MC-nek

Könnyen belátható, hogy 5V-os tápegységből 2 kell, mert ha megszüntetem a tápellátást a CAN rendszernek, attól még a fedélzeti mikroszámítógépnek kell áramot kapnia. Ezt kiküszöbölendő a tápegységek végső száma négyre duzzad.

A tápegységek integrált áramkörei nincsenek olyan precízen megszabva, mint a többi a rendszernél, mivel ezek az alkatrészek minimális speciális egyéni tulajdonságokkal rendelkeznek. Ennek megfelelően a végső alkatrészek megrendelése előtti készlet alapján születik meg a döntés a típust illetően.

De nagy általánosságban elmondható, hogy modern, magas frekvenciájú buck-kapcsolóüzemű tápegységekről beszélünk.

4.9. Távoli adminisztráció

Normális üzemben ezek a tevékenységek vezeték nélküli kapcsolaton keresztül történnek. A CDP-t evidens módon a vezeték nélküli modem-jén keresztül, a BIT-MC-t pedig a CDP I²C buszán keresztül lehet szükség esetén elérni.

Amennyiben a vezeték nélküli kapcsolat nem működőképes akkor az adminisztrációs feladatokat a két UART port-on keresztül lehet ellátni egy közeli PC-ről.

Azért, hogy a beépülő kártya felügyelhető legyen, illetve a hallgatók által demonstrációkra hozzáférhető legyen, a vezeték nélküli modem-nek aktívan Wifi hálózathoz kell csatlakoznia. Az AElabor híresen gyenge vezeték nélküli internet szimulációját figyelembe véve beszerzésre került egy új vezeték nélküli hozzáférési pont, ami lehetőséget teremt a rendszerhez való hozzáféréshez mobil és hordozható eszközökről.

4.10. Külső eszközök

Az elkészítendő külső eszközök listája a különböző UART szervíz kábelekben és a vészmegszakító gombban merül ki. Míg az előző egy primitív kábel a két végén megfelelő konnektorokkal, addig a második egy kicsit komplexebb.

Beszerzésre került egy ipari kivitelű, vandálbiztos 2 állású nyomókapcsoló. Különlegessége ennek az alkatrésznek, hogy a benyomható elem körül világító gyűrű lett kialakítva, az, amiben elhelyezett LED-eket egy külső áramforrásból megtáplálva világításra lehet bírni. Erre a kapcsolóra lett rákötve egy 6 eres kábel, ami egy RJ25 csatlakozóban végződik. Ez kerül a TEMEC egységhez csatlakoztatásra. A másik vég fixre van forrasztva. A kábel 6-ból 4 aktív ere viszi mind a kapcsoló, mind a LED elektromos jeleit.

5. Szoftver

5.1. Linux

Eleinte elkezdtem az RTOS/OS vonalon gondolkodni, de a CDP hardver választás egyértelműen kiadta, hogy milyen operációs rendszert fogok használni.

A Raspberry Pi OS egy modern Debian-on alapuló Linux disztribúció, amit a gyártó frissít. Ebből következik, hogy kiváló a támogatása a választott platformra, és mivel alapból

egy olyan eszközzel beszélünk, amit rengetegen használnak, elérhető egy nagy közösség, akiknek a megoldásaiból lehet tanulni.

Mivel az elérhető hardver erőforrások amúgy sem a legnagyobbak, és nem lesz hosszú távon a CDP-hez kijelző csatlakoztatva, ezért nem éreztem szükségességét az asztali környezettel együtt érkező verzió használatának. Így a választott operációs rendszer a Raspberry Pi OS Lite lett, ami csak parancssoros interakciót támogat. Így maga az operációs rendszer közel nulla erőforrást fogyaszt, de élvezhetjük az előnyeit egy teljes Linux kernelnek. Például könnyebben megszakíthatjuk a felhasználóink jogköreit a Linux fiók-rendszerével.

```

0[          0.0%] Tasks: 30, 63 thr; 1 running
1[          0.0%] Load average: 0.10 0.22 0.13
2[          0.0%] Uptime: 00:06:42
3[          2.0%]
Mem[|||||] 57.7M/427M
Swp[          0K/100.0M]

```

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
763	wagner	20	0	8732	3244	2572	R	1.3	0.7	0:00.78	htop
1	root	20	0	33800	8828	7044	S	0.0	2.0	0:04.58	/sbin/init
132	root	20	0	45724	11460	10792	S	0.0	2.6	0:00.84	/lib/systemd/systemd-journald
159	root	20	0	19924	4844	3364	S	0.0	1.1	0:01.42	/lib/systemd/systemd-udev
312	avahi	20	0	6912	2616	2356	S	0.0	0.6	0:00.09	avahi-daemon: running [TEMEC.local]
314	root	20	0	8200	2376	2200	S	0.0	0.5	0:00.02	/usr/sbin/cron -f
316	messagebu	20	0	7844	3636	3144	S	0.0	0.8	0:00.46	/usr/bin/dbus-daemon --system --address=system
322	root	20	0	39548	5756	5224	S	0.0	1.3	0:00.11	/usr/libexec/polkitd --no-debug
335	root	20	0	26544	2884	2560	S	0.0	0.7	0:00.12	/usr/sbin/rsyslogd -n -iNONE
343	avahi	20	0	6756	260	0	S	0.0	0.1	0:00.00	avahi-daemon: chroot helper
344	root	20	0	12940	5788	5232	S	0.0	1.3	0:00.36	/lib/systemd/systemd-logind
349	nobody	20	0	5312	2244	2060	S	0.0	0.5	0:00.01	/usr/sbin/thd --triggers /etc/triggerhappy/t
355	root	20	0	11788	4028	3652	S	0.0	0.9	0:00.05	/sbin/wpa_supplicant -u -s -O /run/wpa_suppl
408	root	20	0	26544	2884	2560	S	0.0	0.7	0:00.03	/usr/sbin/rsyslogd -n -iNONE
409	root	20	0	26544	2884	2560	S	0.0	0.7	0:00.00	/usr/sbin/rsyslogd -n -iNONE
410	root	20	0	26544	2884	2560	S	0.0	0.7	0:00.04	/usr/sbin/rsyslogd -n -iNONE
416	root	20	0	39548	5756	5224	S	0.0	1.3	0:00.00	/usr/libexec/polkitd --no-debug
417	root	20	0	27696	80	0	S	0.0	0.0	0:00.06	/usr/sbin/rngd -r /dev/hwrng
418	root	20	0	27696	80	0	S	0.0	0.0	0:00.06	/usr/sbin/rngd -r /dev/hwrng
419	root	20	0	27696	80	0	S	0.0	0.0	0:00.00	/usr/sbin/rngd -r /dev/hwrng
420	root	20	0	27696	80	0	S	0.0	0.0	0:00.00	/usr/sbin/rngd -r /dev/hwrng
431	root	20	0	39548	5756	5224	S	0.0	1.3	0:00.02	/usr/libexec/polkitd --no-debug
440	root	20	0	48576	8772	7520	S	0.0	2.0	0:00.34	/usr/sbin/ModemManager

fig. 5.1.1 htop tevékenységfigyelő egy magában webszervert futtató Raspberry Pi zero 2 W-n. A teljes terhelés a CPU 2%-nak felel meg, ebből 1.3% a tevékenységfigyelő. A használt RAM 57.7MB

Forrás: Saját képernyőfotó

A szoftver biztonsága nagyon erősen támaszkodik a Linux jogosultság rendszerére, mivel a Raspberry Pi OS csak adminisztrátor felhasználóknak engedi a GPIO(General Purpose Input/Output, általános kimenet/bemenet) manipulálását, ezért a kimenő jelek engedély nélküli manipulációja ellen az első védelmi vonalunk az, hogy a felhasználókat nem engedjük direkt parancsokat írni, hanem a hajtómű vezérlő számítógéppel csak egy adminisztrátor privilégiumokkal futó központi szolgáltatáson keresztül érintkezhetnek. Itt nyilvánvalóan még be lehet iktatni egy utolsó biztonsági ellenőrzést, így optimális esetben káros vagy nem kívánt parancsok nem is tudják elhagyni a CDP-t.

5.2. Fejnélküli eszközök

A CDP, mivel üzemi állapotban sohasem lesz csatlakoztatva kijelzőhöz, fej nélküli eszköznek számít. Ezeket az eszközöket távolról kell adminisztrálni, úgy, hogy egy másik számítógépről egy terminálba lépünk be.

```
MacBook-Pro-3:~ wagnerdominik$ ssh wagner@temec.local
wagner@temec.local's password:
Linux TEMEC 5.15.61-v7+ #1579 SMP Fri Aug 26 11:10:59 BST 2022 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Oct 19 17:41:27 2023
wagner@TEMEC:~$
```

fig. 5.2.1. SSH bejelentkezés a TEMEC tesztszámítógépébe

Forrás: Saját lépernyőfotó

A legegyszerűbb módja annak, hogy egy távoli számítógépen terminált tudjunk használni, egy SSH kapcsolat létrehozása. Az SSH (Secure Shell host) egy titkosított hálózati protokoll, ami fejnélküli számítógépek távoli adminisztrációjára lett tervezve. Két titkosítási kulcsot használ, hogy a küldött és fogadott adatok sohase legyenek egyszerű szöveg formájában. Ahhoz, hogy ezt a protokollt használni tudjuk kell a célszámítógépen egy SSH szervert futtatni és a szolgáltatáshoz tartozó 22-es számú hálózati port-nak nyitva kell lennie. Fontos még megjegyezni, hogy a Unix alapú rendszerek az ilyen típusú kapcsolatokat általában alpból támogatják míg a Windows-os számítógépekhez valószínűleg egy 3. féltől származó terminál programot kell letölteni.

Fájlok átvitelére pedig normál üzemben az SCP protokoll lesz használva. Ez az SSH standard része, és ugyanazt a kétszeresen titkosított hálózati protokollt használja, csak ebben az esetben fájlok átvitelére. Az SCP mind a 2 irányba tud adatot átvinni.

Az operációs rendszer alpból tartalmaz egy SSH szerveret, és könnyen konfigurálható a szolgáltatás használatára. A segéd alkalmazást, amit a Raspberry Pi Foundation az operációs rendszer lemezképeinek elkészítése bocsájt a felhasználó részére tartalmazza a beépített eszközöket ahhoz, hogy egy teljesen fej nélküli installációt hajtsunk végre, így a mikroszámítógépet soha nem kell kijelzőhöz csatlakoztatni.

Advanced options X

Image customization options for this session only ▼

Set hostname: .local

Enable SSH

- Use password authentication
- Allow public-key authentication only

Set authorized_keys for 'pi':

Set username and password

Username:

Password:

Configure wireless LAN

SSID:

Hidden SSID

Password:

fig.5.2.1. Raspberry Pi imager szoftver. Az SSH és WLAN beállítható az operációs rendszer kiírása előtt, így amikor a friss lemezt behelyezzük a számítógépbe, azonnal látható lesz a hálózaton

Forrás: Saját képernyőfotó

5.3. Webserverek

A tanulók felé irányuló felület klasszikus webtechnológiával lesz elkészítve. A CDP számítógépen egy HTML szerver fog futni, ami elérhetővé fog tenni a lokális hálózaton egy weblapot, amin követni lehet majd a hajtómű műszaki adatait. Erről a weblapról lehet majd letölteni a tesztek után a naplófájlokat is.

A hajtómű adatok folyamatos közvetítését nem lehet standard HTML-lel megoldani, mivel ott a weboldalt csak egyszer kéri le a böngésző. Ennek a megoldására jó pár technológia szóba jött. Elsőként szó volt arról, hogy a webes felület WebAssembly-ben lesz megírva, és a kérdéses gépen lokálisan fog futni, ahonnan nyit egy TCP port-ot, amin majd az információt

fogadja. Ez aránylag korán elvetésre került, a hatalmas komplexitás és a széles körű kompatibilitás nehéz implementálása miatt.

Utána következőre a WebSocket-ekre irányult a figyelem. Ezek az internetes, a böngészőnek küldött parancsokkal nyitott kétirányú csatornák, amiken sok adatot nagy frekvenciával lehet 2 számítógép között küldeni. Ez egy sokkal realisztikusabb megoldás, habár a duplex információ küldési képesség nem kell. Ezenfelül az, hogy lehetségesen folyamatos kommunikáció van sok erőforrást fogyaszt a szerver részéről ez az én esetemben, ahol egy Raspberry Pi zero 2 W az összes erőforrás, ami rendelkezésre áll.

Így marad a harmadik és kevésbé ismert megoldás: az SSE (Server Side Events). Ez egy olyan webtechnológia, ami lehetővé teszi, hogy a szerver folyamatosan diszkrét időközönként adatsomagokat küldjön egy felhasználónak, akinek csak egyszer kell a weboldalt betöltenie. Ez kiváló alapot fog teremteni működés közben a hajtómű adatainak sugárzásához.

Fontos itt kis kitérőt tenni a kiberbiztonságra. Onnantól, hogy a CDP webtechnológiákat használ, ki van téve az ezekkel a technológiákkal járulékos internetes fenyegetéseknek. Nem is beszélve arról, hogy nyitva van az SSH port folyamatosan. Így, mivel a számítógép biztonság kritikus helyen dolgozik, az a döntés született, hogy légréssel le lesz választva a hálózatról.



fig 5.3.1. A beszerzett TP-LINK router, ami a laboron belüli adatforgalom lebonyolításáért fog felelni

Forrás: Saját fotó

Ezt megvalósítandó a belépés egy új routerről fog történni, amit kifejezetten erre a célra vásároltam és nem lesz rákötve az internetre, így százszázalékosan biztonságossá téve a rendszert ilyen típusú fenyegetésektől.

5.4. GPIO interfész

A Raspberry Pi egy egyszerűen programozható GPIO interfésszel érkezik. A 40 tűske funkcióit a majdnem minden nagyobb programozási nyelven kiadott könyvtárak segítségével lehet programozni. A GPIO UART port-ja a rendszer beállításokon belül átállítható vészhelyzeti konzollá. Ez az UART lesz továbbítva a CDP szervíz konnektorjába.

5.5. CDS

A CDS, avagy Central Data Service (Központi Adatszolgáltatás) a legfőbb adatszolgáltatás, ami a CDP-n fut. Ez a program felelős minden bejövő adat érzékeléséért, azoknak továbbításáért, illetve lejegyzéséért. A CDS egy 3 szál program, egy szálon az emberek felé nyitott webszerver fut, a másik szálon egy belső adatszerver, a harmadikon pedig az adatrögzítés, a CAN kommunikáció és a kiegészítő funkciók.

Fontos biztonsági lépcső, hogy ezzel a szolgáltatással csak rendszergazda jogokkal lehet érintkezni a webszervereken kívül. Ez azért van, mert ha a felhasználók esetleg írnak egy scriptet, amivel tesztelni akarják a hajtóművet, akkor annak a lépéseit csak a hálózaton tudják elküldeni, még akkor is, ha magán a CDP-n fut a kódjuk. Így, mielőtt a parancsok a GPIO-ra ki lesznek küldve, a beépített szűrőknek van esélyük kiszűrni azokat, amelyek potenciálisan károsak lehetnek.

5.6. Rust vs. C

A Rust egy kortárs rendszerprogramozási nyelv, nagyon mély hardverirányítási képességekkel és modern magasszintű absztrakciókkal. Ami miatt viszont ez a nyelv komoly hírnévre tett szert, az a totális biztonság fókusz. A Rust compiler rengeteg memóriavédelmi és programozásarchitektúra integritását megőrző szabályt tartat be a programozóval. Ezek a szabályok garantálják, hogy a kimenet egyáltalán nem is lesz képes sok olyan hibába beleesni, amibe más rendszerprogramozási nyelvekben, például C-ben, vagy C++-ban rutin szerűen belekavarodnak.

Egyre több nagyvállalat, és programozó bízik meg a 2012 óta létező nyelvben. Én azért választottam a Rust-ot a küldetés-kritikus kód nagy részének a megírására, mert sokkal nehezebb benne hibát véteni, mint C-ben és hasonló a futási sebessége.

5.7. Könyvtárak és API-k

Azok számára, akik programokat kívánnak írni a rendszerre, rendelkezésükre lesz bocsájtva egy olyan könyvtár, ami tartalmaz API-eket, amikkel adatokat olvashatnak ki a CDS-ből és parancsokat küldhetnek végrehajtásra.

Ez remélhetőleg egy egyszerű módszert fog biztosítani a kártya képességeinek bővítésére.

5.8. Frontend

A weboldal frontend-je nem klasszikus HTML elemekből és nem is valamilyen keretrendszer elemeiből, hanem egy vászon alapú pixelenként rajzolt lényegében képből fog állni. Ez a döntés azért született, mert így a grafikát nagyon hasonlóra meg lehet csinálni, mint a forgalomban lévő repülőgépek hajtómű adatokat megjelenítő kijelzői. A dinamikus elemek és a vászon JavaScript-el kerül megjelenítésre.

5.9. Ellenőrző szoftver

A kiválasztott STM32 mikrokontrollerre C nyelven teljesen egyéni szoftver fog íródni, melynek a feladata a CAN busz kommunikációs folyamatok monitorozása, az egyéb rendszer paraméterek monitorozása és vészhelyzet esetén a CAN kommunikációs rendszer tápegységének kikapcsolása. Ezen felül ez a chip képes lesz kommunikálni a CDP-vel az I²C buszon keresztül, ahol a felhasználók különböző adatokat kérdezhetnek le és beállíthatják ezen mikrokontroller működési módjait.

A CDP mielőtt kimenő adatfolyamot próbálna indítani ellenőrizni fogja a BIT-MC írási mód regiszterét, amiben engedélyezettnek kell lennie a kimenő adatforgalomnak. Ez csak akkor kerülhet engedélyezésre, ha azt előbb a CDP kéri, és a hajtómű nem megy. Az utóbbit a mikrokontroller folyamatosan ellenőrzi. Ha a kimenő adatforgalom nem engedélyezett és ez az egység mégis adatforgalmat érzékel, akkor a CAN rendszer tápegysége kikapcsolásra kerül a tápegységet visszakapcsolni csak a CDP explicit kérésére lehet. A manuális megszakításokat ez a kontroller szintén ellenőrzi. Ha manuális megszakítás történt, akkor is vissza kell állítani a tápegységet ezen a mikrovezérlőn keresztül.

6. TEMEC v0.1

6.1. Miért van szükség v0.1-re

Az MCP2515 tesztelése közben napvilágot látott a probléma, hogy a modul, amit forrasztás nélküli tesztelésre vásároltam, nem tud a TEDDI egyedi CAN busz paramétereivel, illetve a nem éppen ideális teszt összeállítás problémáival megbirkózni. Emiatt muszáj volt minimum ezt az áramkört külön tesztelni, de így, hogy a rendszerek 70 százaléka valamilyen módon a CAN-hez kapcsolódik, egyszerűbb egy "nulladik" prototípust építeni kissé gyengébb specifikációkkal.

Ezen a ponton őszintén meg kell vallani, hogy a TDK dolgozat leadására a biztonsági rendszerek nem lettek volna végleges állapotban, így az időm effektívebb kihasználása érdekében inkább a kulcsfontosságú részletre fókuszáltam.

Fontos még a fejlesztés ezen pillanatában, hogy legyen egy próbaeszköz, amin a szoftvert a lehető legvéglegesebb hardveren lehet tesztelni. Így az a döntés született, hogy elkészül ez a prototípus.

6.2. TEDDI tesztelő eszköz

Az előzetes tesztekhez Dr. Beneda Károly tanár úr segítségével egy jelgeneráló eszköz készült, amivel műhely körülmények között lehet szimulálni a hajtómű pár fontos paraméterét. A konstrukció nagyon kezdetleges, de kulcsfontosságú szerepet játszott a CAN busz megfigyelésében.



fig. 6.2.1. A TEDDI tesztelő eszköz

Ez az eszköz az egyik 50 tűs csatlakozóba kerül csatlakoztatásra és a fekete, kézbe fogható dobozán lévő potenciométerekkel és gombokkal meg lehet szabni a hajtóműbe bemenő jelek profilját.

6.3. Tesztösszeállítás és problémái

Nem sokkal a v0.1-es kártya tervezésének megkezdése előtt haza tudtam vinni a TEDDI a hajtómű vezérlő számítógépet. Ekkor az a probléma állt elő, hogy nem volt elérhető a sugárhajtóműhöz rögzített többi alkatrész, ami például az egység áramellátásáról gondoskodott. Így az itthoni munkaközegben építenem kellett egy tesztösszeállítást, amivel el tudtam kezdeni a különböző alkatrészekkel kísérletezni a rendszeren.

A 24V elsődleges tápfeszültség szerepét egy eredetileg LED világítási rendszerek meghajtására gyártott kapcsolóüzemű tápegység töltheti be. A szóban forgó számítógép a tesztek után bizonyíthatóan teljesen kompatibilis ezzel a megváplálási módszerrel.

Ezzel ellentétben az adat-kábelek lényegesen megnehezítik a tesztelést. A vékony, instabil tűsorok sokszor mérés közben is kontakthibák tüneteit produkálják. Így például az oszcilloszkópos ellenőrzések Során rengeteg zajt kell kiszűrni a képből.

A CAN busz tesztelésére az oszcilloszkóp 1 vagy 2 (attól függ, hogy középső referencia, vagy valamelyik feszültség a referencia) szondával be van kötve. Ezután a TEDDI tesztelő eszközzel a bemenetek változtatása az oszcilloszkópon megfigyelésre kerül. A kép stabilizálására az oszcilloszkóp CAN módját használom, és egy specifikus üzenet címkére szűrök.

6.4. A „nulladik verzió” eltérő komponensei

A kártya nulladik verziójában több dolog el fog térni a végleges verziótól. Első lépésben az önellenőrzési folyamatok minimalizálásra kerültek. Ennek megfelelően nincs BIT-MC, így az ehhez tartozó passzív és aktív alkatrészek is hiányoznak. A CAN kommunikáció tápegységének a bekapcsoló pin-je át lett helyezve a CDP egyik GPIO lábára. Végző biztonságnak a gomb és az elektromechanikus leoldórendszer ugyan úgy megmaradtak.

A másik jelentős változás az, hogy az összes UART és a gomb pin-jei is egy csatlakozóra kerültek, ahol egy szalagkábelon hagyják el a TEDDI dobozát és egy külső műszerdobozban kerülnek két RJ25 csatlakozóra szétosztásra.



fig. 6.4.1. TEMEC v0.1 elosztódoboz

Ami érdekesség még, az az, hogy a rendelés pillanatában a beszállítómnak nem volt 3,3V-os CAN adóvevője, így egészében 5V a CAN kommunikációs rendszer. Emiatt 3,3V-os, 5V intoleráns Raspberry Pi zero 2 W -vel nem érintkezhet az említett két alkatrész, tehát be kellett építeni felfelé egy műveleti erősítőkből, lefelé ellenállásosztókból álló szinteltoló áramkört.

6.5. Összeszerelés

A kártya üvegszál-erősítésű műanyag két oldalas nyomtatott áramköri lapja saját tervek alapján egy külső beszállítótól érkezik, míg az ebből generált digitális alkatrészlista nyomán az alkatrészek megérkeznek egy másik külső beszállítótól.

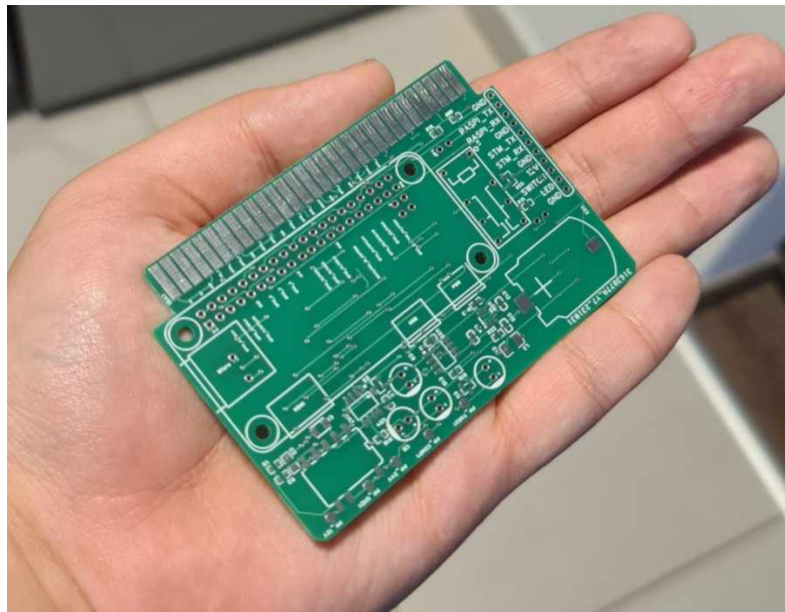


fig. 6.5.1. A TEMEC v0.1 PCB. A konstrukció a kis profil betartása érdekében hasonló méretű mint egy modern bankkártya

Forrás: Saját fotó

A prototípus modul a digitális tervrajzok nyomán lesz összeszerelve. Ez jórészt forrasztással, a CDP rögzítésénél csavarokkal és anyákkal történik. A lap jórészt SMD technológiák mentén lett tervezve, ilyen módon az összeszerelés egy kicsit körülményesebb, mint a klasszikus THT alapú áramköri lapok. A munkafolyamat közben klasszikus és újra olvasztásos forrasztási módszerek kerülnek felhasználásra.

7. TEMEC v1 és a jövőbeli fejlesztések

7.1. Miért nem fért bele ebbe a dolgozatba a végső kártya?

Azért nem tartalmaz ez a dolgozat a végső TEMEC kártya üzembe állítását, mert a különböző erőforrások tökéletlen beosztásából adódóan a projekt nem jutott el az adott idő alatt a kívánt stádiumba. Szükség lenne még STM32 fejlesztése, a felhasználó felé irányuló web felület javítására és a többi funkcióhoz a megfelelő vezérlő kód implementálása a többi beépülő kártyába.

Ugyanakkor be kell vallani, hogy madártávlatból szemlélve a történeteket ez nem egy fő projekt. Egy ilyen szintű számítógépes modul megtervezése és programozása akár éveket vehet igénybe, nekem utólag az is meglepő, hogy idáig jutottam. Viszont örülök annak, hogy legalább a vezérelveket sikerült lefektetni a projekt további fejlesztéséhez.

Fontos még megjegyezni, hogy teszteléshez rengeteg új integrált áramkör próba modul és szerszám lett vásárolva, ezek sajnos pénzügyileg is komoly tehernek bizonyultak, így jelenleg azt is várom, hogy jobban álljanak az anyagiak, ahhoz, hogy a projektet be lehessen fejezni.

7.2. A szolgálatba állás idővonala

Mivel jelenleg úgy néz ki, hogy a TKT-1 gázturbinás próbapad nem fog a közeljövőben működni, ezért a projekt nem feltétlenül halad gőzerővel. A mostani cél a legvégső verzió szolgálatba léptetésére jövő félév (2023/24-2) eleje.

7.3. Jövőbeli fejlesztések

Miután a teljes verzió szolgálatba tud állni, komoly fókusz lesz helyezve egy másodlagos hardveres kimenetre, amivel esetleg a TEMEC egy kezelő pultba épített kijelzőt tudna ellátni.

Ezen felül a közelebbi jövőben újra kell tervezni az M850 hőmérséklet mérő modult, amint ennek a projektnek vége, annak neki fogok állni. Mint az korábban a dokumentumban említésre kerül, a tervek szerint a TEMEC CAN busz kommunikációs rendszerével lehet majd

rutin kalibrációkat és ellenőrzéseket csinálni az új M850-en, így ez lesz az első kártya, ami az új hardverrel teljes integrálásra kerül.

8. Köszönetnyilvánítás

Szeretném megköszönni Dr. Beneda Károly tanár úrnak és az AElabor kollégáinak a segítségüket az elmúlt fél évben.

9. Tartalom:

Tartalomjegyzék

1.	A TKT-1 gázturbinás sugárhajtómű.....	2
1.1.	<i>Leírás.....</i>	2
2.	A TEDDI hajtóművezérlő számítógép	2
2.1.	<i>Leírás.....</i>	2
2.2.	<i>Feladatok</i>	3
2.3.	<i>Alkotóelemek</i>	4
2.3.1.	<i>Alaplap</i>	4
2.3.2.	<i>M250</i>	5
2.3.3.	<i>M850</i>	5
2.3.4.	<i>AUTO (M288).....</i>	5
2.3.5.	<i>BITE-M.....</i>	6
2.3.6.	<i>Egyéb alkatrészek</i>	6
2.4.	<i>CAN busz</i>	6
2.5.	<i>Hibakeresés.....</i>	7
2.6.	<i>A TEMEC projekt</i>	8
3.	A TEMEC projekt célkitűzései.....	9
3.1.	<i>Alapfeladat</i>	9
3.2.	<i>Geometriai/anyagtechnikai követelmények</i>	10
3.3.	<i>Hardveres követelmények.....</i>	10
3.4.	<i>Szoftveres követelmények.....</i>	11

4.	Hardver	11
4.1.	<i>Mikrovezérlő, vagy egykártyás számítógép</i>	<i>11</i>
4.1.1.	Mikrokontroller	12
4.1.2.	SBC	13
4.1.3.	Következtetés	14
4.2.	<i>Raspberry Pi zero 2 W.....</i>	<i>14</i>
4.3.	<i>CAN interfész.....</i>	<i>17</i>
4.4.	<i>Biztonságtechnikai problémák a CDP-vel.....</i>	<i>18</i>
4.5.	<i>Elektromechanikus/elektronikus biztonság</i>	<i>19</i>
4.6.	<i>Valós idejű óra</i>	<i>21</i>
4.7.	<i>Másodlagos csatlakozások</i>	<i>22</i>
4.8.	<i>Tápegységek</i>	<i>22</i>
4.9.	<i>Távoli adminisztráció</i>	<i>23</i>
4.10.	<i>Külső eszközök.....</i>	<i>23</i>
5.	Szoftver	23
5.1.	<i>Linux.....</i>	<i>23</i>
5.2.	<i>Fejnelküli eszközök</i>	<i>25</i>
5.3.	<i>Webszerverek.....</i>	<i>26</i>
5.4.	<i>GPIO interfész</i>	<i>28</i>
5.5.	<i>CDS.....</i>	<i>28</i>
5.6.	<i>Rust vs. C.....</i>	<i>28</i>
5.7.	<i>Könyvtárak és API-k</i>	<i>29</i>
5.8.	<i>Frontend.....</i>	<i>29</i>
5.9.	<i>Ellenőrző szoftver</i>	<i>29</i>
6.	TEMEC v0.1	30
6.1.	<i>Miért van szükség v0.1-re</i>	<i>30</i>
6.2.	<i>TEDDI tesztelő eszköz.....</i>	<i>30</i>
6.3.	<i>Tesztösszeállítás és problémái</i>	<i>31</i>
6.4.	<i>A „nulladik verzió” eltérő komponensei</i>	<i>31</i>

6.5.	Összeszerelés	32
7.	TEMEC v1 és a jövőbeli fejlesztések.....	33
7.1.	<i>Miért nem fért bele ebbe a dolgozatba a végső kártya?</i>	33
7.2.	<i>A szolgálatba állás idővonala</i>	33
7.3.	<i>Jövőbeli fejlesztések</i>	33
8.	Köszönetnyilvánítás.....	34
9.	Tartalom:	34
10.	Források	36

10. Források

Klasszikus akadémikus forrásaim nem voltak ezen a kutatáson, mivel ez inkább tervezőmunka volt, illetve a legfőbb forrás, a TEDDI FADEC itt volt a kezemben. A használt alkatrészek szakmai leírásait tudom linkelni.

- <https://ww1.microchip.com/downloads/en/DeviceDoc/MCP2515-Stand-Alone-CAN-Controller-with-SPI-20001801J.pdf>
- <https://ww1.microchip.com/downloads/en/DeviceDoc/MCP79400-MCP79401-MCP79402-Data-Sheet-20005009G.pdf>
- <https://www.st.com/resource/en/datasheet/stm32h757xi.pdf>
- <https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#gpio-and-the-40-pin-header>
- <https://www.st.com/en/microcontrollers-microprocessors/stm32f103/documentation.html>
- <https://doc.rust-lang.org/book/>
- TKT-1-73-MFM-00