

Mikrokontroller laboratórium házi feladat

Hanics Mihály Péter

Neptun-kód: UJ47SY

konzulens: Kovács Adorján

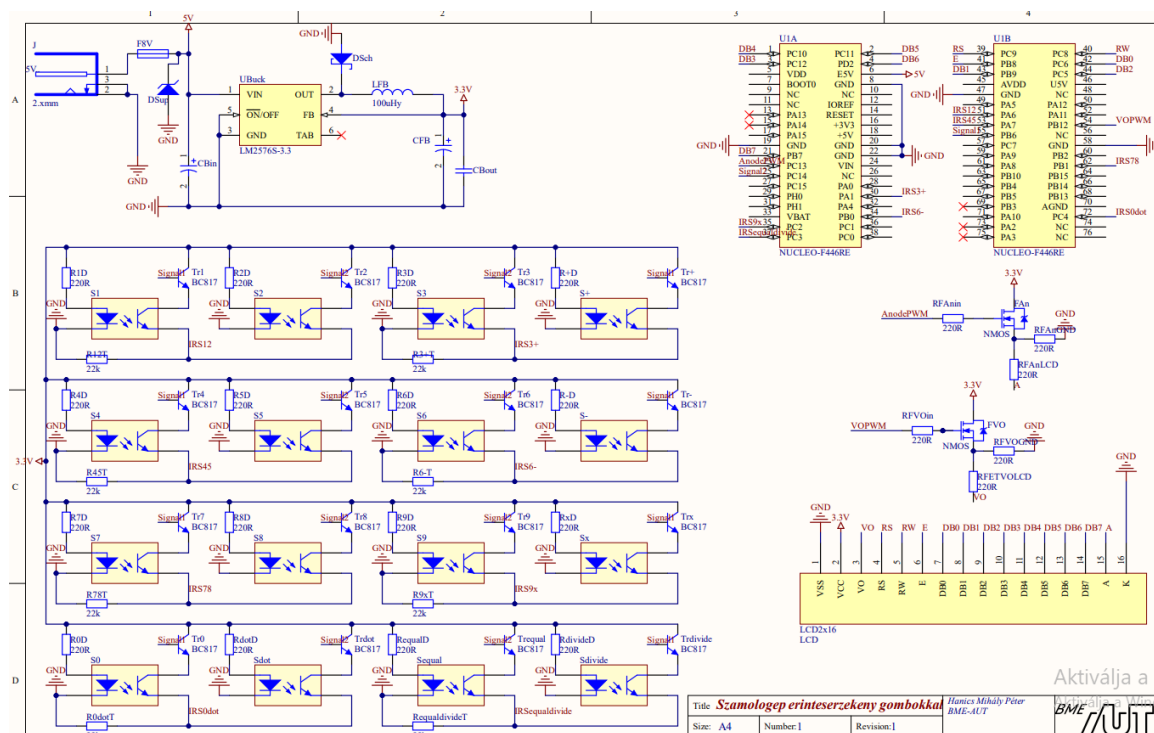
A kitűzött feladat: Számológép készítése érintésérzékeny gombokkal

Készítendő kiegészítő hardver egység az STM32 NUCLEO-F446RE kithez, amely egy 6 digitos, a négy alpművelet végrehajtására képes számológépet valósít meg. A számológép egy 2x16 karakteres alfanumerikus kijelzőt és 16 „érintés érzékeny gombot” tartalmazzon, 4x4-es mátrixban elhelyezve. Ezek infravörös szenzorok, melyek az analóg bemenethez illesztendőek. A kijelző fényereje változtatható legyen egy billentyűkombinációval. Műveletet az LCD felső során jelzünk ki, eredményt az alsó felén, képes legyen tizedestörteket kezelni. Eredményt a tizedespont hosszas lenyomása után tudunk törölni.

A házi Altium Designerben megvalósított része:

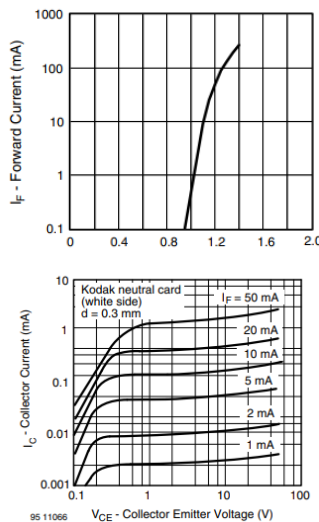
- 1db schematic kapcsolási rajz, amely tartalmazza a Nucleo-t, 16 szenzort, egy LCD-t (melynek schematicját is elkészítettem), kiegészítő áramköri elemeket és egy tápáramkört (az LM 2576S-3.3V elem köré építve adatlapja alapján)
- Footprintet készítettem: LCD-nek a ChipCAD-en talált kép alapján, 100uF-os és 1000uF-os elko-knak adatlapjaikban található méreteik alapján az Altium IPC Wizard segítségével, ugyanezzel egy 100uHy-s induktivitás és a szupresszor dióda footprintjét is elkészítettem. Felhasználtam a tanszéki könyvtárból a CNY70 footprintjét, megfelelő értékű ellenállásokat kerestem melyek méretei (0805) egyeznek a tanszéki ellenállás footprint méretével, szintén felhasználtam a tanszéki táp footprintet és a multifuse footprintet is, melyhez kerestem példányt Lomexen, amit találtam túl nagy áramot bír elviselni. A szupresszor dióda feszültségszintje is 8V ami enyhén magas. Felhasználtam a tanszéki NMOS footprintet, ehhez találtam példányt Lomexen, de az LM2576S-3.3V áramköri elemnek footprintjét a gyártói könyvtárból szereztem.
- PCB Layout, kimeneti fájlok generálása

Kapcsolási rajz:



A Nucleo külső 5V-os tápról van ellátva, a tápáramkör a bal felső sarokban található (LM2576S-3.3V Buck-konvertert használva). A Nucleonak 8db analóg bemenetére csatlakozik 16db infravörös szenzor kimenete (fototranzisztor emittere) és a hozzájuk tartozó áramköri elemek, párokban. A Nucleo PB6 (Signal1) és PC14 (Signal2) lába vezérli azon BJT-eket, melyekkel kiválasztható hogy mely szenzor analóg kimenete jelenjen meg a Nucleo egyes bemenetén. A Nucleonak ezentúl 13 kimenete van csatlakoztatva az LCD-hez, ebből két darab PWM-mel megvalósított, a világításhoz, ehhez tartoznak FETek.

A CNY70 infravörös szenzor működése a következő: egy infravörös emitterdióda adja az infravörös fényt, mely jó fényviszonyban visszatükröződik a fototranzisztor bázisa felé, sötétségben pedig nem tükröződik vissza, a tranzisztor pedig ez alapján nyit vagy zár. Az áramköri kialakítás úgy lett megoldva, hogy ha egy érzékelőhez hozzáérünk, ami nyilvánvalóan a ritkább eset, akkor nyisson a tranzisztor s az analóg bemeneten a feszültség magas (kevesebb fogyasztás). Az ellenállásokat ezen diagram alapján terveztem:



Egy leolvasható érték az $I_F = 10\text{mA}$, $V_F = 1.1\text{V}$. Ez alapján terveztem sorosan ellenállást: a kettőn 3.3V kell essen, áram 10mA, $R = \frac{3.3V - V_F}{I_F}$, ez alapján $R = 220\Omega$.

A tranzisztor kollektorának potenciálja 3.3V,

szeretnénk nyitáskor minél magasabb feszültséget az emitteren mérni, ezért minimalizálnánk a kollektor-emitter feszültséget, amit az ábra alapján a kollektoráram minimalizálásával tudunk, ezért a kimenettel sorosan kötöttem egy 22k Ω ellenállást, így minimalizálva az áramot. Az ábra alapján az áram

kb. 0.11mA lehet, így az emitterfeszültség 2.42V (ha az ellenállás pontos, ami nagy ellenállásoknál nem jellemző), V_{CE} pedig kb. 0.9V. A gyakorlatban az ellenállás értéke több, mint 22k Ω , és az emitterfeszültség 2.63V értéket vesz fel tipikusan, ha odaérintjük az ujjunk/egy tollat stb.

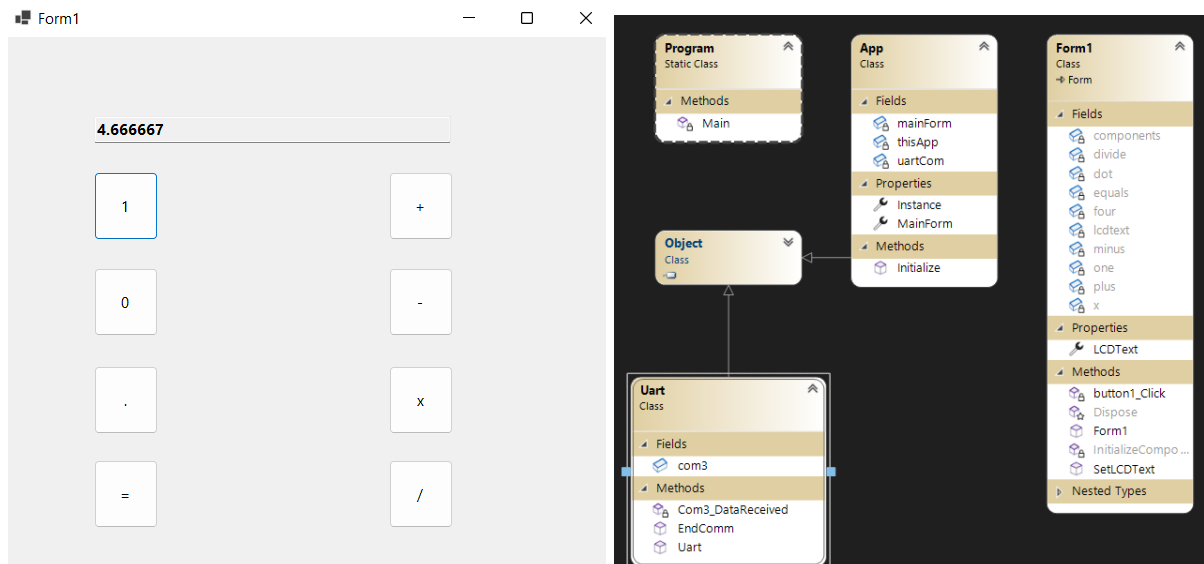
A tápáramkör a Buck-konverter adatlapja alapján készült el, ahol pontosan le volt írva, hogy ilyen áramkört kell a Buckhoz illeszteni, specifikált kapacitás és induktivitás értékekkel. A Buck kimenete táplálja az LCD-t és a szenzorokat. Az LCD lábaira digitális értékeket kell adni, időzítve (Enable lábbal).

A feladatmegoldásban az LCD nem vett fel elegendő feszültséget az anód lábán (hiába korrigáltam lebegő vezetékekkel, csak a háttérvilágítás feszültség szintjét sikerült ezzel 3.3V-ra hozni), és többszörösen hibásan működött, így ezt kivettem az áramkörből az utolsó héten és a kliensalkalmazásban szimulálom.

Azon szenzorok, melyekhez tartozó BJT-eket a Signal1 jellel vezéreljük megfelelően működnek (két szélső oszlop), a Signal2 jellel vezéreltek viszont sosem kapnak magas feszültséget, s így páronként csak 1 szenzor volt használható (a „szélsőbbik”). Határidő után derült ki ennek az oka: már a Nucleo PC14 lábán (mely a Signal2) sem jelent meg magas feszültség, hiába állítottuk be szoftveresen: a PC14 láb hardveresen oszcillátorhoz tartozó láb, nem GPIO, átállításához 2db solder bridge-t kellett volna változtatni a boardon.

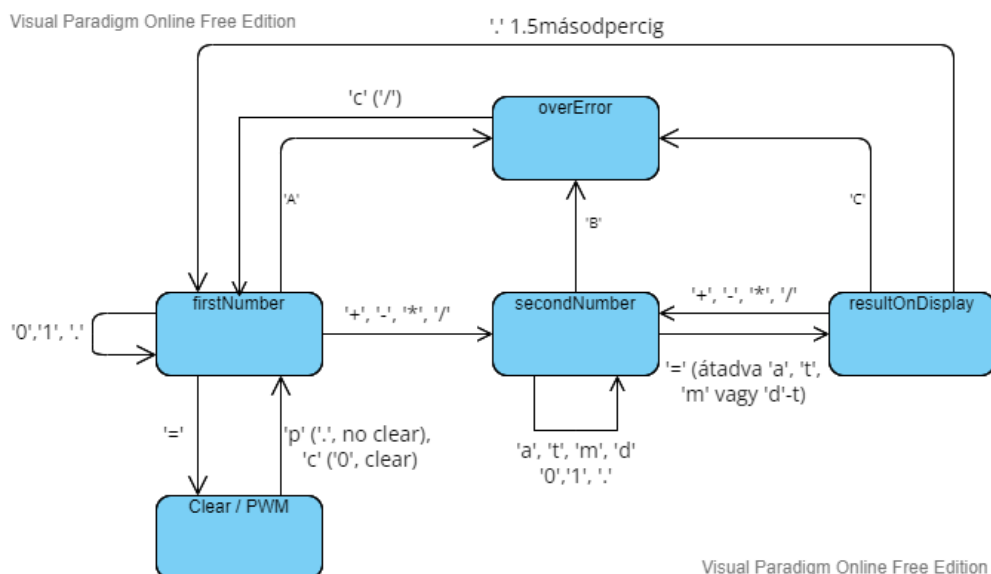
PC kliensalkalmazás:

Az imént említett probléma okán konzulenssel egyeztetve a számológép csak 8 gombját használtam, és bináris számológépet valósítottam meg (gombok: 0 1 . = + - * /). Az LCD-t a kliensalkalmazásban szimulálhattam.



Firmware blokkvázlat:

A firmware állapotgépszerűen lett megtervezve, az állapotok: firstNumber, clearORsetBrightness, secondNumber, overError, resultOnDisplay. Az állapotok közötti lépkedést egy `storedValuesType storedValues;` változóval oldjuk meg, ahol a típus tartalmaz egy char-t ami meghatározza hogy mely állapotba lépünk (például az 's' az állapotban maradást jelenti, a '+' karakter adódik át ha az első szám után a +-hoz tartozó infravörös szenzort érintjük meg, ekkor pl. firstNumberból secondNumberbe jutunk) és tartalmaz két számot műveletekhez. A blokkvázlata az állapotoknak:



Belépési pont: firstNumber

Az állapotgép magyarázata: A műveletekben szereplő első operandus a firstNumber állapotban írható, 0, 1 számjegyekkel (és tizedesponttal). Ez tárolódik a `storedValues.firstStoredNumber` változóban. Miután ezt beírtuk, ha elrontottunk valamit, akkor Clear-elhetünk egyenlőségjelet nyomva, itt visszakérdez a program, hogy biztos akarunk-e clear-elni? Pontot nyomva nem törlődik a szám, 0-t nyomva igen. Beírva a megfelelő számot, lenyomva egy operátort (+, -, *, /) áttérhetünk a második számra, ekkor a storedChar értéke az operátorhoz tartozó karakter lesz, hogy feljegyezzük, hogy mely műveletet hajtsuk majd végre az eredménynél. Itt hasonlóan meg adhatjuk a második szám értékét. Entert nyomva kiíratjuk az eredményt, itt 1.5 másodpercig lenyomva tartva a pontot visszaléphetünk egy új művelet elvégzéséhez. Bármikor meghaladjuk a 16 jegyet egy számban, a program jelzi ezt, s visszaléphetünk clearelvé.

Firmware kapcsolata a PC-vel: UART-tal kommunikálnak, méghozzá stringeket átadva. Az adó a Nucleo, a vevő a kliensalkalmazás, a Nucleo mindig olyan stringeket küld, melyek lezáró nullája előtt \n-t tartalmaznak, és így a kliensalkalmazás ReadLine-nal egyszerűen tudja olvasni.

```
HAL_StatusTypeDef UART_Init(){
    MX_USART2_UART_Init();

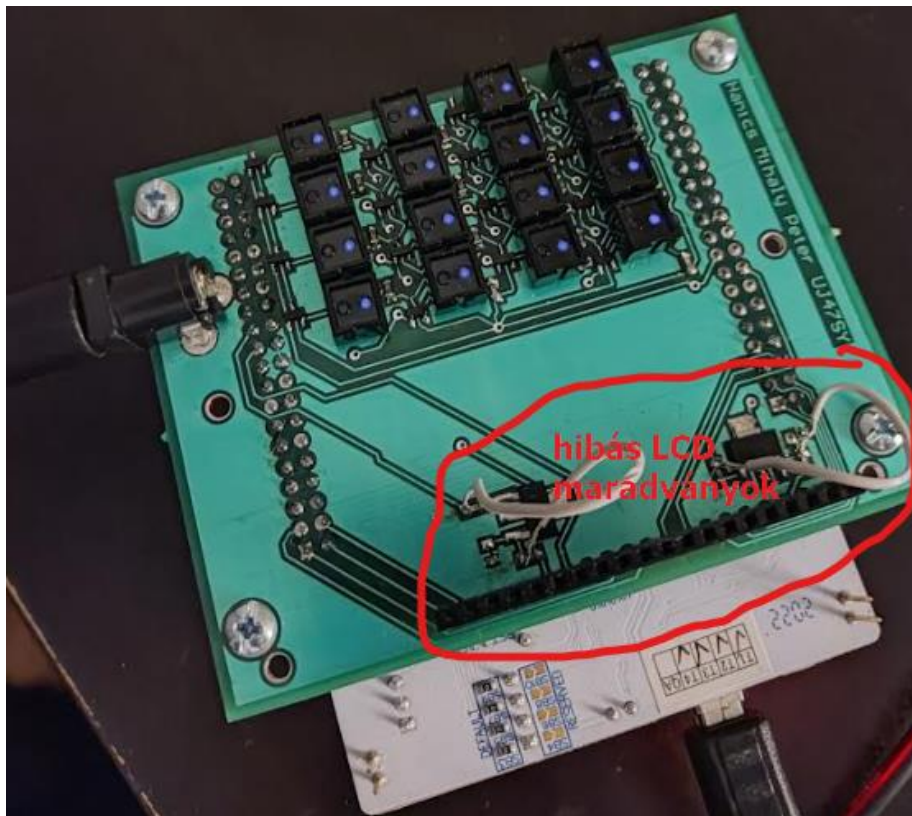
    HAL_UART_Receive_IT(&huart2, (uint8_t*) &rxBuffer, 1 );

}
HAL_StatusTypeDef UART_SendString(char *str){
    while(huart2.gState!= HAL_UART_STATE_READY && huart2.gState!=
    HAL_UART_STATE_BUSY_RX );

    strncpy(txBuffer, str, TXBUFFERSIZE);
    HAL_UART_Transmit_IT(&huart2, (uint8_t* )txBuffer, strlen(txBuffer));
}
```

A kliensalkalmazás UART kommunikációja egy külön Uart.cs fájlban van megírva. Itt megadtuk a portot, megírtuk a függvényt, mely adatfogadáskor az adatot kezeli, és a függvényt ami a kommunikációt befejezi lezáráskor.

Működés közben készített kép:



Utóbbi megjegyzések:

Ha most kezdeném el a házifeladatot, először megnézném, hogy az egyes lábak mire használhatóak, és a gyári beállítások (solder bridge-k, stb.) szerint mire használatosak ezen lábak.

Tudomásul venném, hogy idén a tanszék lényegesen több embert vett fel specializációra, mint más tanévekben, ezért egy konzulens ill. oktató sem tud annyi időt fektetni egy-egy házi feladatba, mint előző években, s többet kell hagyni magamra és társaimra.

Több önálló döntést hoztam volna.