

## SPECIFIKÁCIÓ

Tárcsaszabályozás elkészítése Arduino (szabályozó), illetve LabVIEW (felhasználói interfész) platformon.

### SZABÁLYOZÓ (ARDUINO)

A rendelkezésre álló hardveren kaszkádosított PI-szabályozókból álló struktúra került megvalósításra, ugyanis a motor tehetetlenségéből kifolyólag az áramjelet is vizsgálni kellett a felhasználó által beállítandó alapjel mellett – annak érdekében, hogy a beavatkozó jelet ne szabályozzuk túl.

A PI-szabályozók esetében a hibajel egy nulla középpontú szimmetrikus környezetében az integráló tagot elhagytuk (holtsáv), mivel a szabályozók által előállított beavatkozó jelek nem nulla értékűek nulla hibajel esetében, az eszköznek viszont ebben az esetben a motorok vezérléséül szolgáló H-hidat ki kell kapcsolnia. A felhasznált enkóder véges felbontásából kifolyólag az adott szöghelyzet elfogadhatónak tekinthető egy adott határértéken belül, ami a kódban külön konfigurálható.

Az enkóderek olvasása megszakítások segítségével került megvalósításra, míg a szabályozást a főciklus végzi. A PC-s interfésszel való kapcsolattartás soros porton keresztül történik, az eszköz a relatív szöghelyzet értékét küldi vissza (abszolút szöghelyzet meghatározására az opto-kapu nem megfelelő működéséből kifolyólag nincs lehetőség a jelenlegi konfiguráció mellett).

A motorvezérlés állapotgép segítségével került megvalósításra (a hibajel előjelét használjuk iránydiszkriminációra a H-híd vezérléséhez, mivel a PWM-láb csak nemnegatív bemenettel vezérelhető üzemszerűen).

A megvalósított Arduino kód lehetőséget biztosít továbbá hibakeresési célzattal a szabályozók státuszinformációinak elküldésére.

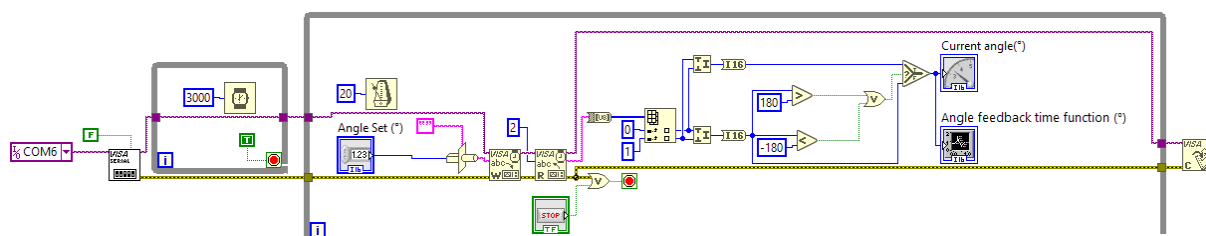
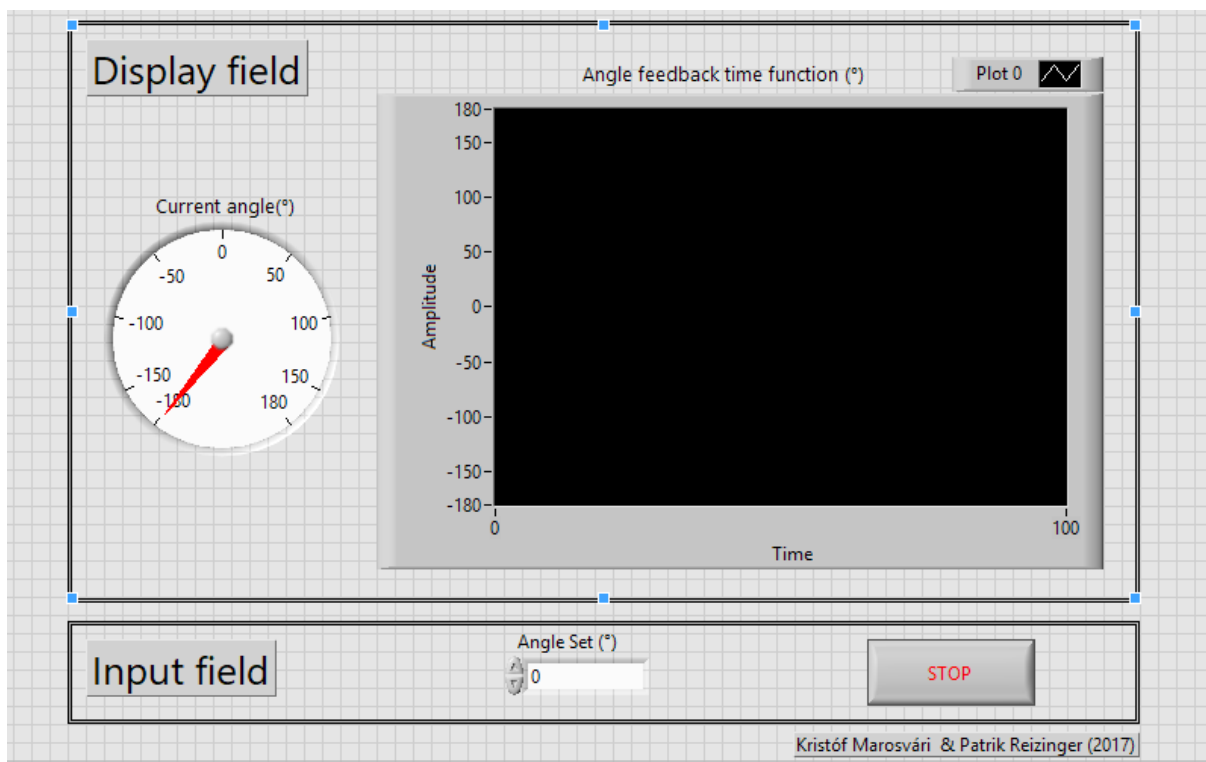
### FELHASZNÁLÓI FELÜLET (LABVIEW)

A felhasználó számára kialakított GUI (Graphical User Interface) három, az eszköz működésével kapcsolatos paraméter beállítását, illetve megjelenítését teszi lehetővé:

- Bemenet: elfordulás szöge (16 bites előjeles egész)
- Kimenet: elfordulás szöge (16 bites előjeles egész) aktuális értéként (Gauge), valamint időfüggvényként (Chart)

A LabView-ban elkészített virtuális műszer alapvetően soros porti kommunikációt folytat két irányban, a beolvasott sztringtömböt bytetömbbé, majd 16 bites intté konvertálva történik az adatok megjelenítése.

A mellékelt ábrán látható komparátoros struktúra a megolvadt enkódertárca hibáját hivatott mérsékelni, ez ugyanis gyakran okoz a vett byteok esetében túl-, illetve alulcsordulást (a nem megfelelő sorrendben vett byteokból kifolyólag). Az alkalmazott megoldás nem ad megoldást a hardveres hibára, azonban a felhasználó számára így a visszajelzés megfelelő.



Budapest, 2017. 12. 04.