

Tudományos diákköri dolgozat

**Impulzusüzemű és frekvenciakétszerezett Nd:YAG lézer
vezérlőprogramjának fejlesztése**

Készítették:

Jórárt Péter, III. évf. fizikus

Kopasz Katalin, IV. évf. fizika tanár

Témavezetők: Görbe Mihály, Dr. Osvay Károly

Szegedi Tudományegyetem

Optikai és Kvantumelektronikai Tanszék

Szeged, 2006

Tartalomjegyzék

1. Bevezetés	3
2. Előzmények	5
2.1. A TeWaTi lézerrendszer	5
2.2. Nyalábirány-ellenőrzés a távoli zóna figyelésével	7
2.3. Nyalábenergia szabályozás félhullám-lemez és polarizációs nyalábosztó kocka kombinációjával	9
2.4. A mérőeszközök lingua francá-ja: a LabVIEW	10
3. Célkitűzések	12
4. A vezérlőprogram szerkezete	14
5. A lézerrendszer automatikus indítása és leállítása	15
5.1. Követelmények a programmal szemben	15
5.2. A programrész működése	15
5.3. A megoldás elve	15
6. A nyalábirány stabilizálása	16
6.1. Követelmények a programmal szemben	17
6.2. A programrész működtetése	17
6.3. A stabilizálás megoldásának elve	19
6.4. Hardveres problémák és megoldásuk	25
6.4.1. Forró pixelek	25
6.4.2. Az expozíció triggerelése	25
6.6. Az elért pontosság megbecslése	27
7. Nyalábenergia-vezérlés	29
7.1. Követelmények a programmal szemben	29
7.2. A programrész működtetése	30
7.3. A nyalábenergia-vezérlés megoldásának elve	31
7.4. Az elért pontosság megbecslése	33
8. Fénymegszakító-vezérlés	34
8.1. Követelmények a programmal szemben	35
8.2. A programrész működtetése	36
8.3. A fénymegszakító-vezérlés megoldásának elve	36
9. Adatmentés és naplózás	38
9.1. Követelmények a programmal szemben	38
9.2. A programrész működése	38
9.3. A megoldás elve	38
10. Összefoglalás	39
Irodalomjegyzék	40
Köszönetnyilvánítás	41

1. Bevezetés

A természetben előforduló elemi folyamatok nagy része femto- és attoszekundumos (10^{-15} — 10^{-18} s) időskálán játszódik le. Ezen jelenségek tanulmányozásához nagyságrendileg ugyanilyen időtartamú (ultrarövid) fényimpulzusokra van szükség. Ma már előállíthatóak olyan lézerimpulzusok, amelyek elegendően rövidek ahhoz, hogy segítségükkel ezek az elemi jelenségek kísérletileg vizsgálhatók legyenek. A fény–anyag kölcsönhatást tanulmányozó kísérleteknél ebben a rövid lézerimpulzusban nagyon nagy teljesítménynek kell sűrűsödnie, ezért a vizsgálatokhoz 6-9 nagyságrenddel meg kell növelnünk a felhasználandó ultrarövid impulzus energiáját a kezdeti néhány nJ-hoz képest, és a nyalábot a céltárgy lehető legkisebb területére kell torzulásmentesen fókuszálnunk [1].

Az SZTE Optikai és Kvantumelektronikai Tanszékén felépült TeWaTi rendszer (terawatt csúcsteljesítményű titán-zafir lézerrendszer) a kelet-közép-európai régióban elsőként megépített és a mai napig egyedülálló lézerrendszer. A rendszer által kibocsátott 25 fs ($25 \cdot 10^{-15}$ s) időtartamú, terawatt (10^{12} W) csúcsteljesítményű lézerimpulzusok fókuszált intenzitása megközelíti a 10^{17} W/cm² értéket, időbeli tisztasága (kontrasztja) meghaladja a 10^{-9} szintet [2,3]. A TeWaTi laboratóriumában található impulzusüzemű Nd:YAG lézert alapvetően két célra használjuk. A frekvenciakétszerezett, 532 nm-es (zöld) fénynyalábjával egyrészt közvetlenül a terawatt csúcsteljesítményű TeWaTi lézerrendszer két erősítőfokozatát hajtjuk meg. Másrészt egy további frekvenciaátalakítás és kétszeres térszűrés után 266 nm-en hozunk létre nagy térbeli homogenitású ultraibolya impulzusokat, melyekkel lézeres anyagmegmunkálási kísérleteket végzünk.

Az erősítőrendszerhez felhasznált fizikai jelenségek természete (pl. a nemkollineáris optikai parametrikus erősítés szögérzékenysége), illetve az önálló anyagmegmunkálási alkalmazásokhoz használt kétszeres térszűrés olyan iránystabilitást követelnek meg, amilyeneket a lézer önmagában nem képes biztosítani.

Mindkét alkalmazás számára további problémát jelent a lézer által az erősítő fokozatokba táplált, illetve a céltárgyra bocsátott energia pontos és reprodukálható beállítása. Ezt korábban manuálisan forgatott félhullámlemezek és polarizációs nyalábosztó kockák segítségével végeztük el, melynek beállítási pontossága messze nem volt kielégítő.

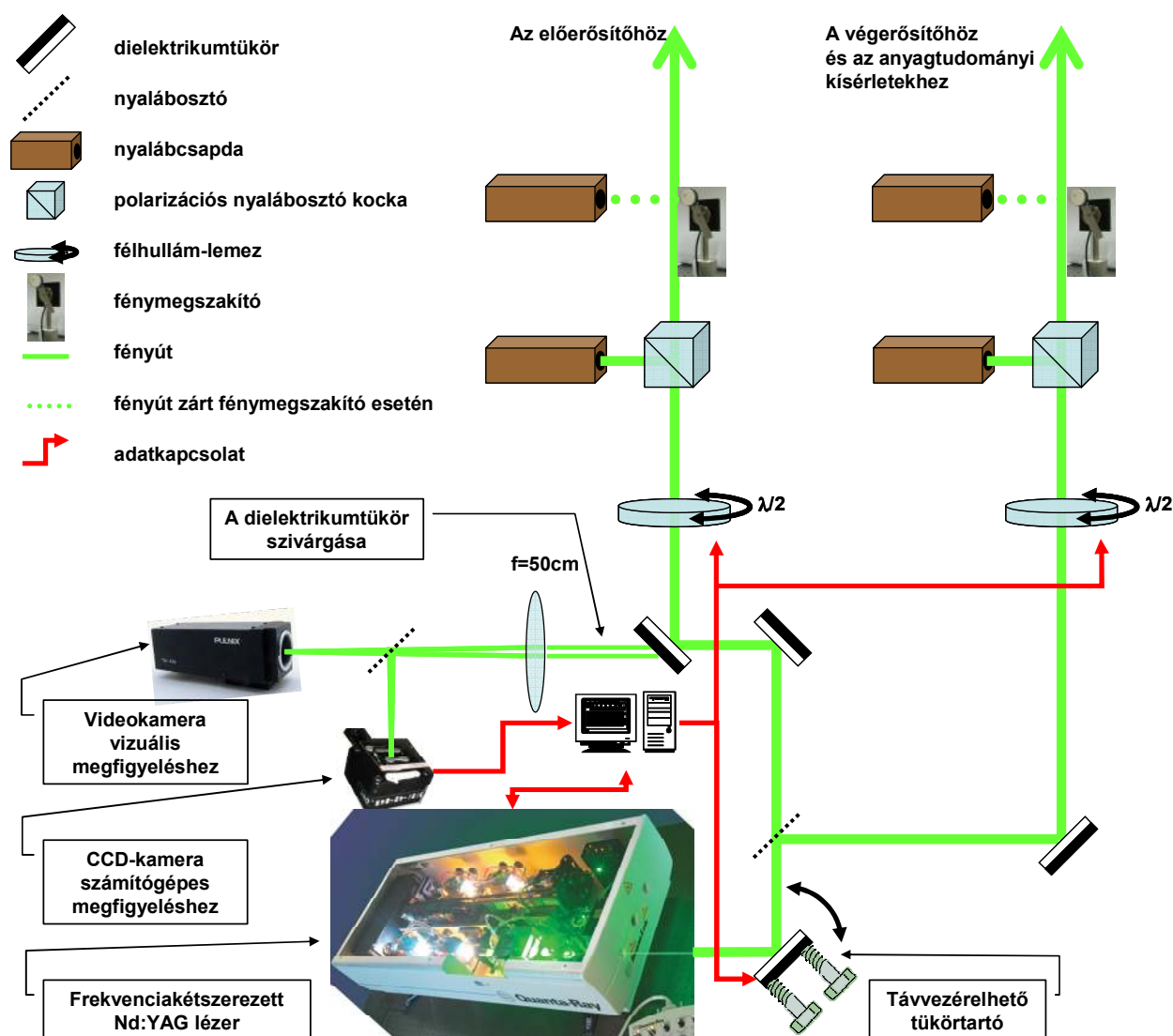
A dolgozat témája a lézer tágabb értelemben vett vezérlőprogramjának hardveres és szoftveres kiépítése. A vezérlőprogramot a fizikus-társadalomban közkedvelt, és a

laboreszköz-gyártók által is szabványnak tekintett LabVIEW fejlesztőkörnyezet grafikus programozási nyelvén, ún. virtuális műszer formájában készítettük el. A program egyes részeit külön-külön írtuk meg, egy-egy programrészlet külön problémát old meg. A programrészleteket ezután illesztettük össze, az előlap paneleket egyesítve, és a felhasználóknak alkalmas módon tagolva.

2. Előzmények

2.1. A TeWaTi lézerrendszer

A lézerrendszer 800 nm központi hullámhosszú, 60 nm sávzélességű magimpulzusait egy titán-zafir oszcillátor hozza létre [2,3]. Ezeket a 16 fs hosszú lézerimpulzusokat egy optikai rácsokból álló impulzusnyújtóban 250 ps időtartamúra nyújtjuk, hogy csúcsteljesítményüket lecsökkentsük, és így az erősítőrendszer elemeit megóvjuk a túl nagy intenzitásoktól.



1. ábra A TeWaTi lézerrendszer pumpálásának és vezérlésének egyszerűsített vázlata

A nemkollineáris optikai parametrikus erősítésen (NOPA) alapuló első erősítő fokozatot egy Nd:YAG lézerrel 10 Hz ismétlési frekvencián pumpáljuk (1. ábra), így $5 \cdot 10^4$ -szeres erősítését érjük el.

Ezután az impulzusokat a Nd:YAG lézer energiájának nagyobb részével pumpált titánzafír kristályban 40 mJ-ra erősítjük fel. A megerősített impulzusokat végül egy ugyancsak rácsos impulzus-kompresszorral összenyomjuk, hosszuk így 25 fs lesz, energiájuk közel 30 mJ marad.

Az Nd:YAG lézert önállóan is alkalmazzuk úgy, hogy egy további frekvenciakétszerezés után anyagtudományi kísérletekhez ultraibolya fényt (a negyedik felharmonikust, mely 266 nm-es hullámhosszúságú) állítunk elő [4,5].

A lézerrendszer együttes, illetve az Nd:YAG lézer önálló, tudományos célú működtetéséhez elengedhetetlenül szükséges a lézer(ek) beállítási paramétereinek paramétereinek pontos és reprodukálható beállítása. Ehhez a mindennapi gyakorlatban a következő négy szakaszban ismertetett főbb lépések tartoznak.

2.1.1. Indítás és leállítás

A lézerrendszer indítása és leállítása soklépéses folyamat, hosszabb (10-20 perces) melegedési szakaszokkal. Gyakorlatlan kezelő könnyen eltéveszti a lépések végrehajtásának sorrendjét, ami a folyamatban visszalépésekhez és így idővesztéshez vezet.

2.1.2. A nyalábirány ellenőrzése

A TeWaTi lézerrendszerben a pumpáló fénynyaláb irányát dielektrikumtükrön átszivárgó fény CCD-kamerára való leképezésével (l. 1. ábra) ellenőrzik.

A nyalábirány elállítódása esetén a korrigálás közvetlenül a lézer kimeneti nyílásánál elhelyezett tükör megdöntésével lehetséges. A tükör üzemkezdeti beállítása, és rendszeres utánállítása korábban kézi művelet volt. Ez nem kellően folyamatos, mert a kezelők nem figyelhetik egész üzemidőben a képernyőn a nyalábirányt, és nem is biztonságos, mert a kísérletező személynek kényelmetlen testhelyzetben, a kezét a lézersugártól néhány centiméterre tartva kell kezelnie a tükörtartó csavarjait, és egy óvatlan mozdulattal könnyen a nyalábra nyúlhat. A pumpáló lézer 10 ns alatt 300 mJ energiát sugároz ki, ami együttesen olyan teljesítményt ($300 \cdot 10^{-3} \text{ J} / 10 \cdot 10^{-9} \text{ s} = 30 \cdot 10^6 \text{ W} = 30 \text{ MW}$) jelent, aminek akár a szórt része is súlyos szem- és bőrkárosodást idézhet elő.)

2.1.3. Nyalábenergia-vezérlés

A TeWaTi lézerrendszer minden alkalmazása igényli a lézer által az erősítő fokozatokba táplált, illetve a céltárgyra bocsátott energia pontos és reprodukálható beállítását. Ez korábban kézzel történt, ami mindössze 15 mJ beállítási pontosságot tett lehetővé. Ez a legtöbb esetben nem kielégítő.

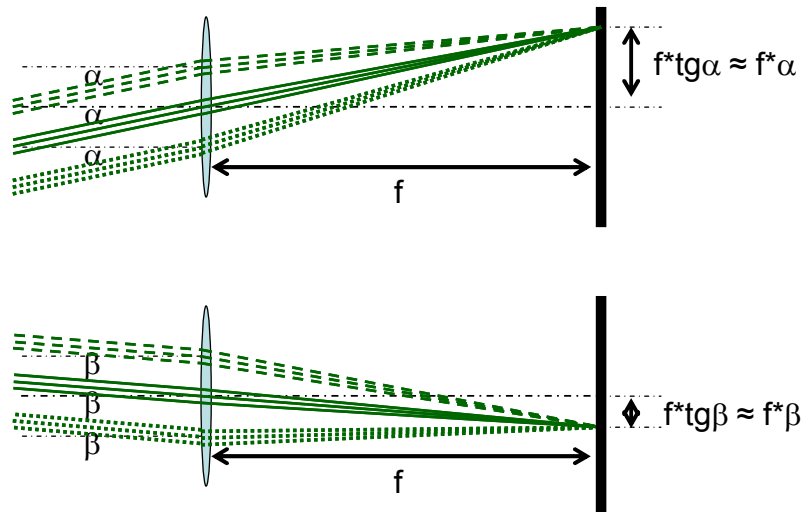
2.1.4. A fényutak nyitása-zárása

Alapvető elvárás, hogy a lézerrendszer főbb fényútjait könnyen és gyorsan el lehessen zárni ill. szabaddá lehessen tenni. Ezt korábban nyalábcspadák (többnyire jó fényelnyelő képességgel rendelkező, nagy hőkapacitású üreges fémtömbök) behelyezésével és eltávolításával oldották meg. Ez a módszer nem volt se könnyű, se gyors (gyakran a kísérlet helyétől méterekre levő nyalábcspadákat kellett mozgatni), se biztonságos.

2.2. *Nyalábirány-ellenőrzés a távoli zóna figyelésével*

Ha egy kollimált fénynyaláb útjába vékony lencsét, a lencse fókuszsjkjába pedig ernyőt helyezünk, akkor a fókuszált fényfolt helye a paraxiális optika szerint csak a bemenő fény sugar irányától függ. Amint az 2. ábrán látható, a különböző térbeli pozíciójú, de azonos irányú (folytonos, szaggatott ill. pontozott vonalakkal jelölt) fény sugarak az ernyő ugyanazon pontján találkoznak: ott, ahol a megadott irányú és a lencse közepén áthaladó fénynyaláb metszi az ernyőt. A különböző irányú fény sugarak azonban más és más helyekre képeződnek le: az 2. ábra alsó és felső részén látható kétféle irányú fény sugarak az ernyő két különböző pontján fókuszálódnak.

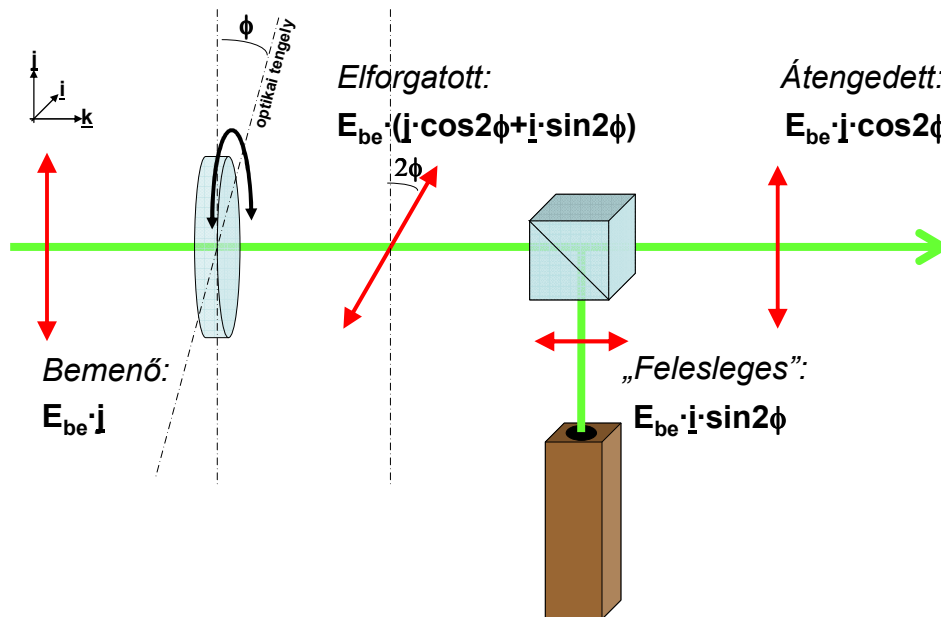
Ezt a jelenséget gyakran használják ki összetett optikai rendszerekben a nyaláb helyes irányának ellenőrzésére: az ernyőn megjelölt ponton tartják a fény sugar fókuszált képét. Gyakori az is, hogy ernyő helyett egy kamera chipjét használják és a megjelenítő képernyőn bejelölt referenciahelyen tartják a foltot.



2. ábra A nyalábirány és a fókuszált fényfolt pozíciójának összefüggése

A módszerrel elérhető pontosságot a detektor mérete (kamera esetén a pixelméret, szem esetén a csapok és pálcikák nagysága) a lefókuszált lézernyaláb átmérője és az objektív fókusztávolsága együttesen határozza meg. Ideális esetben a fénynyaláb fókusza *pontosan egy* pixelt világít ki. Kamerák esetén egyetlen pixel gerjesztése a gyakorlatban nehezen kivitelezhető, mivel a pixelek nem kör alakúak, ráadásul elektronikusan is kölcsönhatásban állnak egymással („áthallás” van közöttük). A paraxiális optika szerint a lencse fókusztávolságának növelésével egyenes arányban nőne a pontosság, mert a helyes iránytól adott szögben eltérő nyaláb nagyobb távolságra lefókuszálva messzebb kerül az ernyőn (vagy a kamera chipjén) attól a helytől, ahová a helyes irányú nyaláb fókuszálna. Így kisebb eltéréseket is lehetne észlelni, de a valóságban (mivel a lézersugarak Gauss-nyalábokként viselkednek) ilyenkor a folt mérete is növekszik, ami a pontosságot csökkenti. Ezért a pixelméret és fókusztávolság közötti ésszerű kompromisszumot kell megtalálni.

2.3. Nyalábenergia szabályozás félhullám-lemez és polarizációs nyalábosztó kocka kombinációjával



3. ábra A félhullám-lemez és polarizációs nyalábosztó kocka hatása a rajtuk átmenő lineárisan poláros fénynyalábra

A félhullám-lemez kettőtörő kristályból készült lemez, amely a benne szétváló ordinárius és extraordinárius fénykomponens között $\lambda/2$ útkülönbséget hoz létre. Ezáltal a bemenő lineárisan poláros fény polarizációs síkját képes elfordítani. Ennek az elforgatásnak a mértéke változtatható a lemeznek a rá merőleges tengely körüli forgatásával, mert a kijövő polarizáció iránya a lemez állásától függ.

A polarizációs nyalábosztó kocka lehet Glan-Thomson prizma, melyet két háromszög alapú kettőtörő anyagú hasázból állítanak össze [6]. Manapság használatos üvegből készült kocka is, ahol a hasábok között alkalmas dielektrikum rétegszerkezet található. A TeWaTi lézerrendszerben ezt az utóbbi kocka-típust használjuk. A polarizációs nyalábosztó kocka a beeső fényt aszerint engedi át, vagy irányítja oldalra, hogy annak polarizációja a ragasztási síkra való beesési síkban van, vagy arra merőleges.

A fényútban előbb egy félhullám-lemez, majd egy polarizációs nyalábosztó kocka szerepel (3. ábra). Ekkor a félhullám-lemez elforgatásával szintén elfordul az átengedett fény polarizációs síkja is, ennek pedig az a következménye, hogy változik a nyalábosztó kocka által átengedett és általa oldalra vert fény (intenzitás)aránya.

A fent leírt kombináció alkalmas az átengedett fény intenzitásának szabályozására a félhullám-lemez szimpla forgatásával. (Az oldalra vert, „fölsleges” fényt nyalábcspaddal kell felfogni.)

2.4. *A mérőeszközök lingua francá-ja: a LabVIEW*

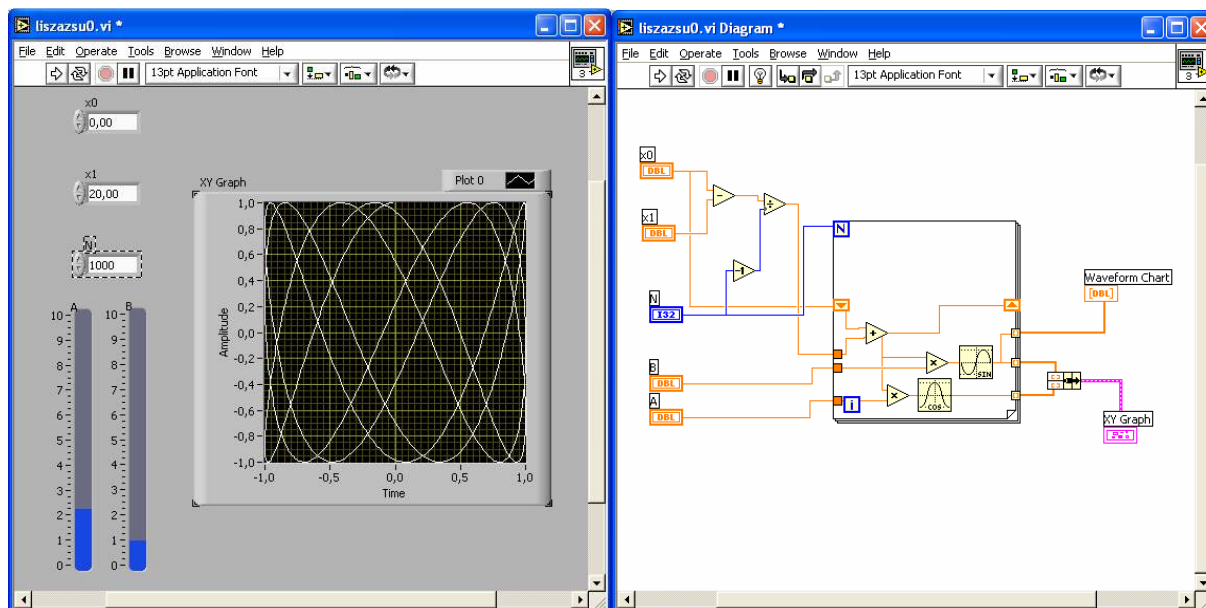
A LabVIEW programcsomagot PC bázisú műszerkészítésre fejlesztették ki. Grafikus programozású nyelv, virtuális műszerek fejlesztésére használják. Ez utóbbiakat a felhasználó készíti el az igényeinek megfelelően.

A LabVIEW egyben egy ipari, szabványos, nyitott architektúra is, amely nagy mennyiségű számítási művelet elvégzését, és ezek eredményeinek megjelenítését és tárolását teszi lehetővé. Adatgyűjtő kártyák, párhuzamos GPIB és VXI csatoló kártyák illesztésével a nyitott és szabványos buszhoz biztosíthatóak a műszer készítésének feltételei. A LabVIEW nagysebességű időkritikus alkalmazások fejlesztését is lehetővé teszi.

Segítségével könnyen és gyorsan kommunikálhatunk digitális eszközökkel. A legtöbb laborban használt lézervezérlőhöz, műszerhez mellékelnek LabVIEW-ban készült programot. Ezeket alprogramként újra felhasználhatjuk, továbbfejleszhetjük.

Egy LabVIEW környezetben írt program neve virtual instrument (VI), azaz virtuális műszer. Fő részei az előlap panel, a diagram panel (4. ábra) és az ikon (konnektor).

- 1) Az előlap panelen találhatóak a beállító és megjelenítő elemek, ezek segítségével kommunikál a felhasználó és a műszer. Ide az elképzeléseink szerinti vezérlőket (gombokat, számmezőket, szövegmezőket, csúszkákat), kijelzőket, diagramokat, grafikonokat helyezhetjük el az átláthatóság és a könnyebb kezelhetőség érdekében.
- 2) A G nyelv segítségével építhetjük fel a blokkdiagramot, ez az elkészített program „forrásnyelvi leírása”.



4. ábra Egy egyszerű LabVIEW-program előlapja és blokkdiagramja

3) A konnektor segítségével a VI-ból subVI-t (azaz alprogramot) hozhatunk létre. Így programunk szubrutinként vagy függvényként meghívható más programok számára is. Nincs korlát a szubrutinok kölcsönös hívásának mélységére vonatkozóan. A végrehajtás sorrendjét az adatáramlás szabja meg. Minden adatnak van forrása és fogadója. Egy adatnak egyetlen forrása van, de több fogadója lehet.

A diagram bármely művelete/folyamata csak akkor hajtódik végre, ha minden bemenő adata rendelkezésre áll. Ha a diagram valamely művelete számára minden bemenő adat rendelkezésre áll, akkor a művelet mindenképpen végrehajtásra kerül. Egy művelet minden kimenő adata egyszerre, a művelet befejezése után áll rendelkezésre.

A LabVIEW-nak négy alapvető struktúrája van:

- „While Loop”: feltételtől függő számú (hátultesztelős) ciklust végző utasítás.
- „For Loop”: véges számú ciklust végző utasítás.
- „Case structure”: feltételes elágazási struktúra.
- „Sequence structure”: sorrendi végrehajtás-struktúra.

A LabVIEW-ban létrehozhatunk tömböket is: ez olyan adatsorozat, melynek minden eleme azonos típusú. A tömb lehet egy- vagy több dimenziós. Ez megkönnyíti azoknak az algoritmusoknak a programozását, amelyekben például mátrixműveleteket kell végezni [7].

3. Célkitűzések

3.1. Az indítás és leállítás automatizálása

A lézerrendszer indítása és leállítása összetett, de állandóan ugyanúgy ismétlődő folyamat, melynek nagy részét (főleg a beiktatott várakozások idejének mérését) számítógép is el tudja végezni. Ezért célul tűztük ki a lézerrendszer indításának és leállításának automatizálását. Ez a mindennapi gyakorlatban megbízható, reprodukálható és gyors indítást valamint leállítást tesz lehetővé, amivel az olyan emberi hibák is elkerülhetővé válnak, mint az indítási-leállítási folyamatok részfeladatai sorrendjének cseréje vagy éppen kihagyása.

3.2. A nyalábirány stabilizálása

A nyalábirány beállítása korábban manuálisan történt. Ez a megoldás nem tette lehetővé a folyamatos korrigálást, és balesetveszélyes volt. Ezért célul tűztük ki, hogy a nyalábirány folyamatos megfigyelése mellett automatizáljuk a beállítást, és az esetlegesen szükséges korrekciót.

3.3. Nyalábenergia-vezérlés

A nyalábenergia beállítási pontosságának növelése érdekében céljaink közé felvettük a félhullám-lemezek léptetőmotorral történő forgatását.

3.4. Fénymegszakító-vezérlés

A fényutak megnyitása és elzárása korábban nehézkes volt. Ezért célul tűztük ki gyorsan, biztonságosan és kényelmesen vezérelhető fénymegszakítók (shutterek) készítését és a lézerrendszerbe történő beépítését. Mivel anyagtudományi kísérleteknél szükség van arra is, hogy a fény a kísérleti mintát bizonyos, jól meghatározott ideig érje, ezért meg kellett oldani azt is, hogy a fénymegszakítók képesek legyenek számítógépes vezérléssel adott időre kinyitni, majd automatikusan lezárni.

3.5. Távírányítás botkormánnyal

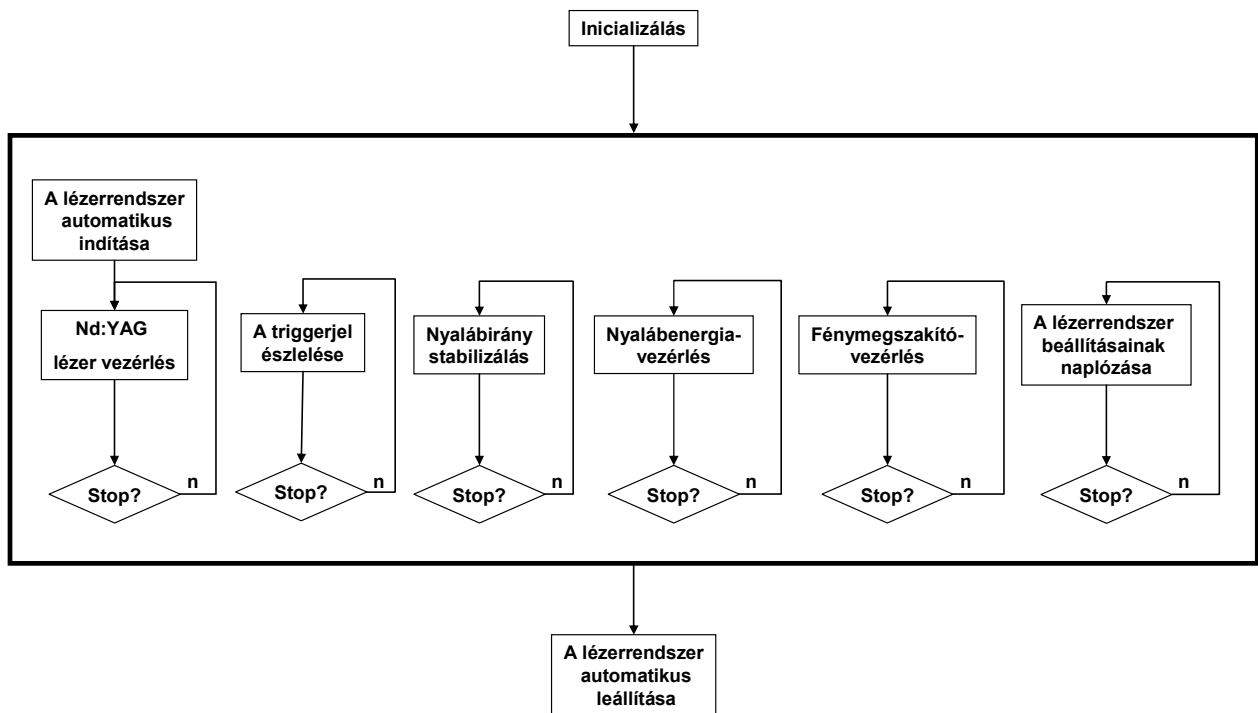
Sokban segítheti a kutató munkáját, ha az optikai asztal mellett dolgozva is szabályozni tudja az egyes nyalábok intenzitását, és nem kell minden változtatáshoz a számítógéphez mennie. Erre a célra botkormányos távvezérlési lehetőséget biztosítottunk a forgatómotorokhoz és a fénymegszakítókhoz.

3.6. *Adatmentés és naplózás*

A számítógép lefagyása esetén szükség van arra, hogy tudjuk, mik voltak az utolsó beállítások. Ez lehetővé teszi, hogy újraindítás után ne kelljen újra kezdeni a mérést, hanem folytatható legyen, esetlegesen több órányi beállítási munkát takarítva meg a kutatónak. Az is szükséges, hogy a felhasználó folyamatosan nyomon tudja követni a beállítások megváltozását, azok időpontjaival együtt.

4. A vezérlőprogram szerkezete

A lézerrendszert vezérlő program felépítését tekinthetjük át az 5. ábrán. A LabVIEW környezet lehetővé teszi, hogy a program egyes részei párhuzamosan fussanak, így a különböző funkciójú, külön kifejlesztett alprogramok egyesítésével egységes vezérlőrendszert hozhattunk létre.



5. ábra A teljes vezérlőprogram egyszerűsített blokkdiagramja

A következő fejezetekben áttekintjük az egyes programrészek felépítését és működését.

5. A lézerrendszer automatikus indítása és leállítása

5.1. *Követelmények a programmal szemben*

A programnak segítenie kell a kezelőt az egyes lépések sorrendjének betartásában, valamint mérnie kell a melegedési és hűlési időket, ezek letelte után figyelmeztetést kell kiadnia.

A rendszerrel több kutató is dolgozik, ezért is van szükség arra, hogy a program leállításakor a motorokat, fénymegszakítókat, visszaállítsuk az alapállapotba. Így a felhasználóknak mérésük megkezdésekor nem kell manuálisan beállítaniuk az elrendezés alapállapotát, és nem kell nyomon követniük, hogy az előző kutató milyen beállításokat használt.

5.2. *A programrész működése*

A program üzenetablakkal „diktálja” kezelőnek az olyan lépéseket, amelyekhez emberi beavatkozás szükséges. Ha várakozásra kerül sor, folyamatjelzőn mutatja az eltelt időt.

Leállításkor a program ellenőrzi még a motorokat, fénymegszakítókat, és amelyek nem alapállapotban van, azt visszaállítja.

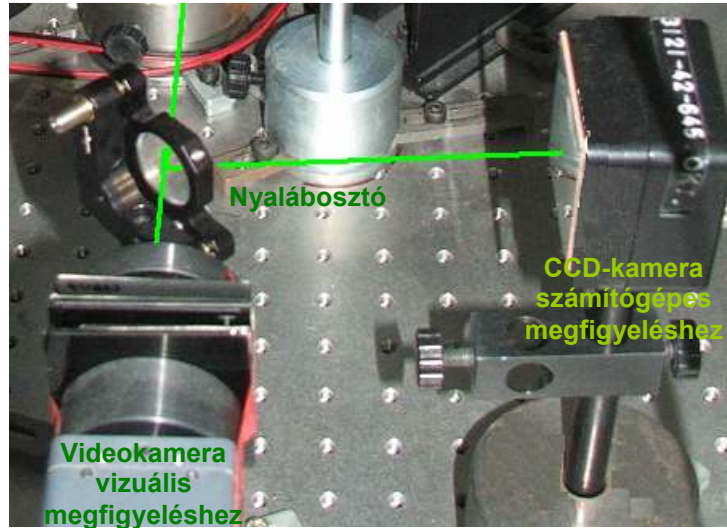
5.3. *A megoldás elve*

A vezérlőprogram a STOP gomb lenyomása után nem azonnal áll le, hanem végrehajtja a következő lépéseket (sorrendi struktúrában):

- Ellenőrzi a forgatómotorok állapotát, és ha valamelyik nem az alapállapotban van, azt visszaállítja.
- A lézeroszcillátor feszültségét 0V-ra állítja, lezárja a lézert vezérlő kommunikációs portot.
- Kikapcsolja az összes fénymegszakítót, lezárja a hozzájuk tartozó portot.
- Alapértelmezett állapotba állítja a program összes vezérlőelemét.

6. A nyalábirány stabilizálása

A munka során az 1. fejezetben ismertetett, korábban vizuális megfigyeléssel és kézi beállítással működtetett nyalábirány-ellenőrző és -helyesbítő eljárást automatizáltuk.



6. ábra Az EDC-1000HR CCD-kamera (jobbaldalt) és a Pulnix videokamera (alul)

A nyaláb helyzetének monitoron való közvetlen megfigyelésére a továbbiakban is szükség van, ezért az arra, szolgáló kamerát meghagytuk, amint az a 6. ábrán látható, de egy nyalábosztóval kettéosztottuk a lefókuszált nyalábot, és a második fényfolt megfigyelésére számítógéppel kiolvasható CCD-kamerát állítottunk be (gyártó: Electrim Corp. Típus: EDC-1000HR, térbeli felbontása: $753 \text{ pixel} \times 244 \text{ pixel} = 8,66 \text{ mm} \times 6,59 \text{ mm}$, színmélysége: 8 bit, azaz 256 szürkeségi szint).

A korrigálásra szolgáló hagyományos tükörtartót piezoelektromos úton mozgatható és számítógéppel vezérelhető típusra (gyártó: NewFocus típus: Picomotor) cseréltük. A kézi beállítás lehetősége továbbra is megvan, az erre szolgáló gombok megfigyelhetők a 7. ábrán.

Az ezután következő programozási munka alapvetően a kamera és a motorizált tükörtartó közötti megfelelő mértékű negatív visszacsatolás létrehozására irányult.



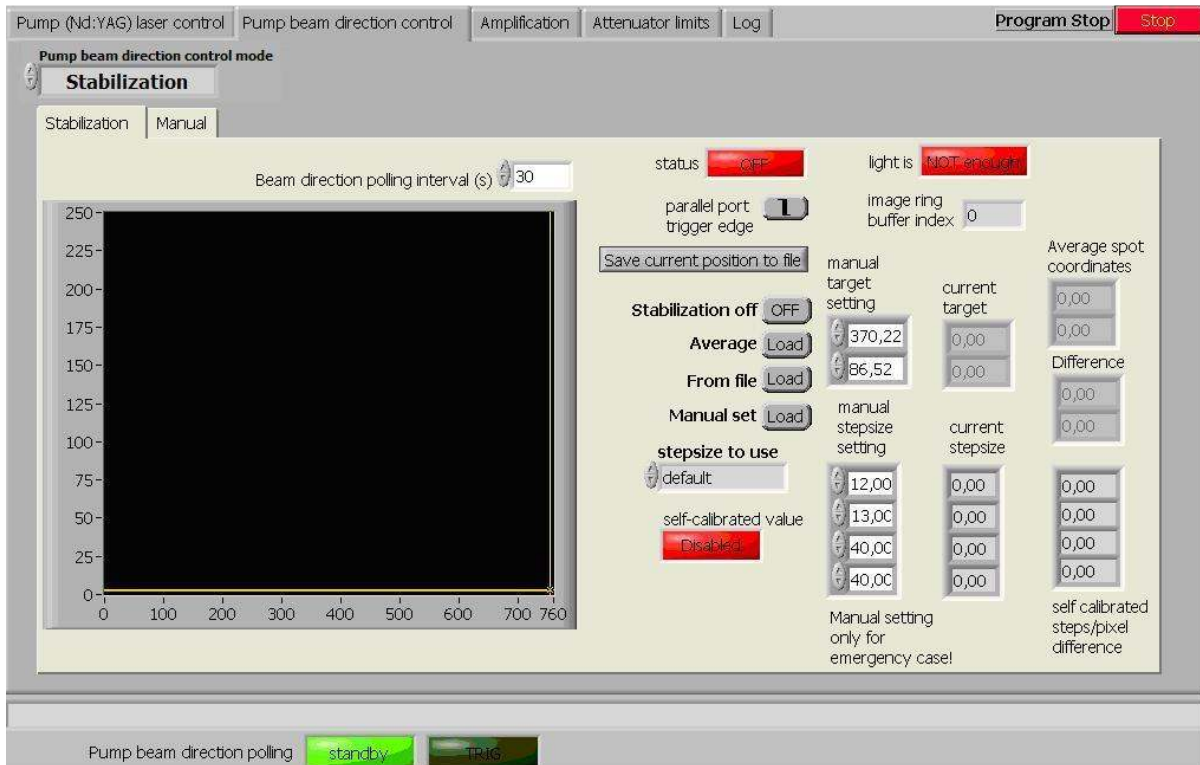
7. ábra New Focus Picomotorral felszerelt tükörtartó az Nd:YAG lézer kimeneténél

6.1. *Követelmények a programmal szemben*

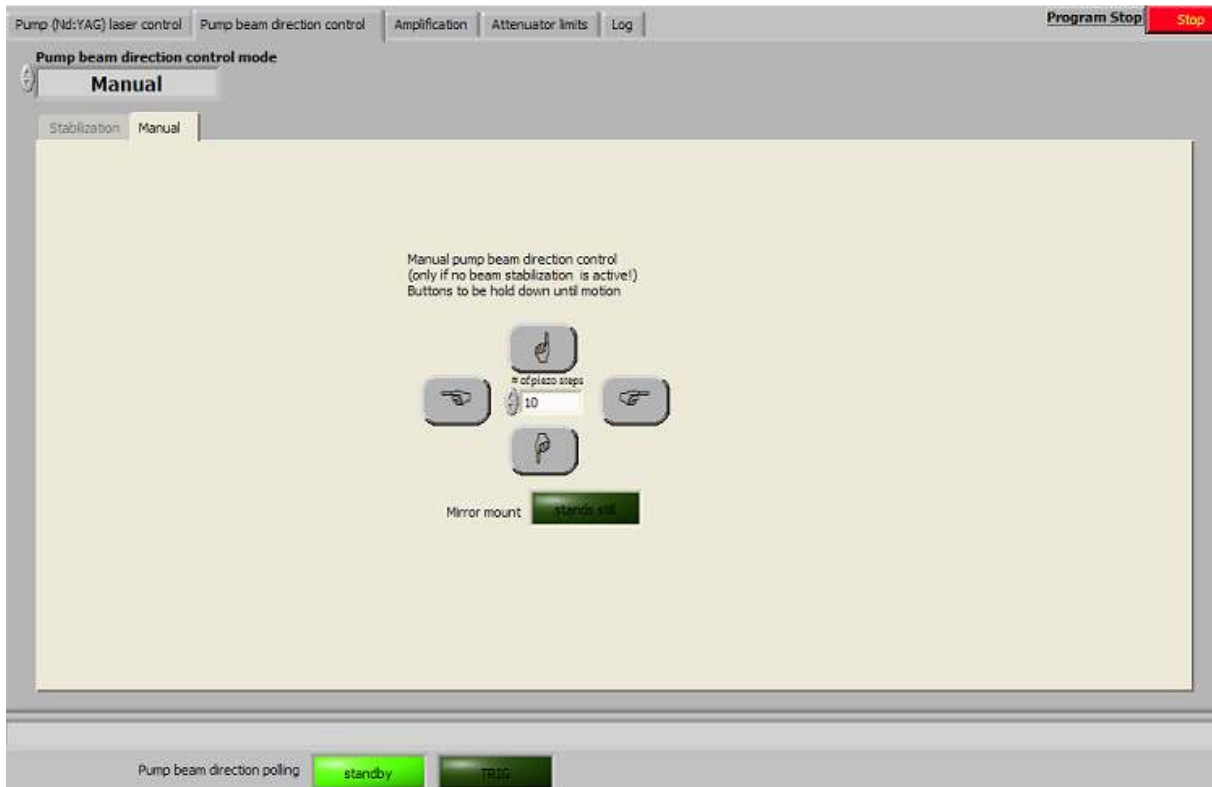
- A stabilizálás aktív állapotában abban az irányban maradjon a nyaláb, ahová előzőleg a felhasználó beállította. Ez az irány legyen elmenthető, és be lehessen tölteni a program következő indításakor.
- Ne okozhasson rendellenes működést (például a tükör rendezetlen mozgását), ha a lézerfolt „lemegy” a képről, vagy valami kitakarja a kamerát.
- A felhasználó se tudjon hibás beállításokkal rendellenes működést okozni. (Felhasználóbarát és viszonylag egyszerűen kezelhető legyen; ne vegye figyelembe a megadott adatokat, ha azok hibásak.)
- A lézerfolt pozícióját a képen minél nagyobb pontossággal kell megállapítani, hogy ezáltal a nyalábirány is minél pontosabb legyen, amivel megnöveljük a laborban végzett mérések reprodukálhatóságát.
- Manuálisan is lehessen vezérelni a nyalábirányt a programon keresztül..
- Mivel a lézer impulzusüzemű, ezért a képbeolvasást a lézerrendszer triggerjéhez kell szinkronizálni.

6.2. *A programrész működtetése*

A nyalábirány-vezérlő programrész az előlap „Pump beam direction control” címkéjű fülén érhető el. Itt egy legördülő menün választhatjuk ki azt, hogy a nyalábirányt a program gombjaival szeretnénk-e beállítani („Manual” menüpont, ez az alapértelmezett), vagy legyen aktív a nyalábstabilizálás („Stabilization” menüpont). Az utóbbi esetben háromféleképpen lehet meghatározni, hogy milyen nyalábirányt tekintszen helyesnek a program:



8. ábra Nyalábstabilizáló programrész előlapja



9. ábra A tükörtartó kézi vezérlésének előlapja

- A pillanatnyi nyalábirány (amit az előző 10, a kamera által látott pozíció átlagaként értelmezünk) tartása: „Average” gomb (8. ábra).
- Korábban fájlba mentett, és újra betöltött irány tartása: „From file” gomb.
- Az előlap „Manual target setting” címkéjű mezőibe beírt koordináták tartása: „Manual set” gomb.

A kézi vezérlés paneljén a tükörtartó négy megdöntési irányának megfelelően négy nyomógomb található (9. ábra), és egy szövegmezőben meg lehet adni, hogy hányat lépjenek a motorok egy gombnyomásra ezekbe az irányokba. Gombnyomás után a lépésszám és a döntési irány közvetlenül a motorvezérlő alprogramnak adódik át, a tükörtartó állításának folyamatát pedig egy ellenőrzőlámpa jelzi a gombok alatt.

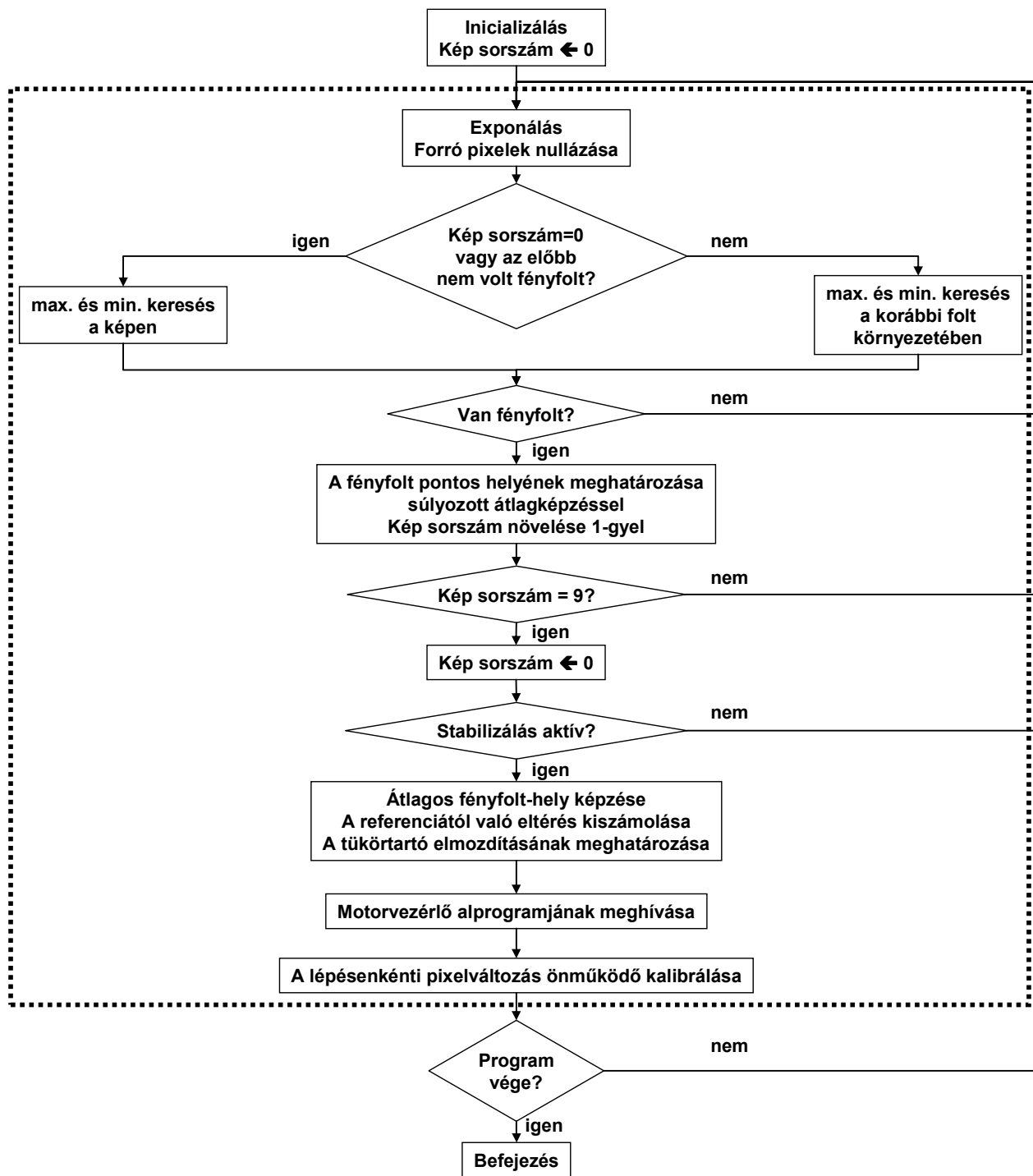
A kézi tükörállítások és a stabilizálási üzemmódban a betöltött pozíciók naplózásra kerülnek.

6.3. A stabilizálás megoldásának elve

A számítógépes algoritmus teremti meg a kamera és a motorizált tükörtartó közötti megfelelő mértékű negatív visszacsatolást. Alapvető lépései:

1. A kamera többszöri kiolvasása, a képeken a fényfolt megkeresése, és a folt-pozíciók átlagának kiszámítása;
2. A referencia és az átlagos hely különbségének megállapítása;
3. A különbségből a referenciára való visszaálláshoz szükséges tükörtartó-mozgatás kiszámítása;
4. A tükörtartó elmozdítása a kiszámolt mértékben;
5. Vissza az 1. lépésre.

Az algoritmus alapötlete a 10. ábrán látható. A szaggatott vonallal bekeretezett részt a jobb áttekinthetőség, és a következőkben ismertetett technikai nehézségek megoldása érdekében alprogramként (a beam_stabilizer_main.vi fájlból) hívjuk meg.

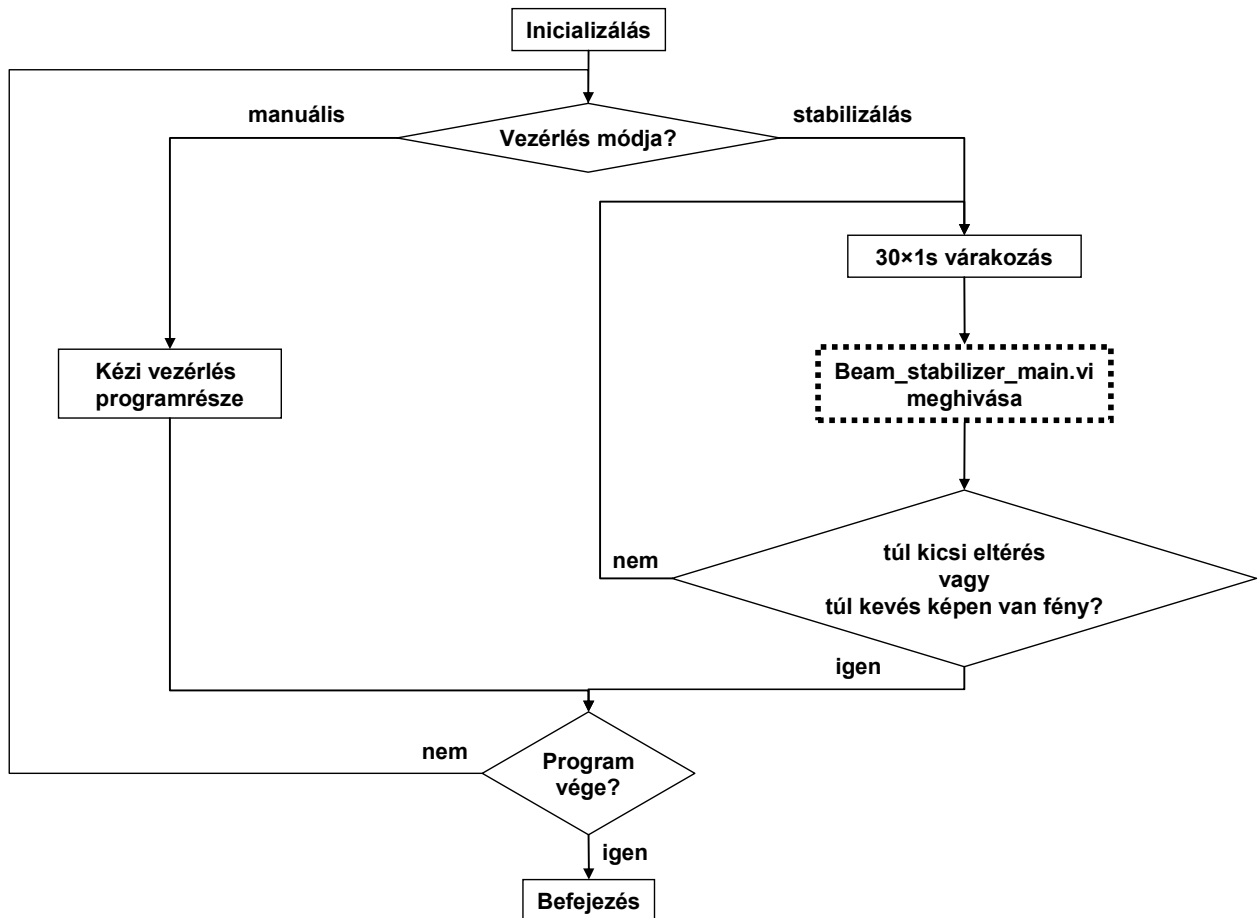


10. ábra A nyalábstabilizálás blokk-sémája

Az EDC-1000HR kamera kiolvasása hosszú művelet (több mint 300 ms), mely közben a többi párhuzamosan futó programrész is nagyon lelassul. Lehetőséget ad azonban a számítógép terhelésének csökkentésére, és a program többi részének jelentős felgyorsítására az, hogy a pumpáló Nd:YAG lézer nyalábiránya nem változik gyorsan.

Ezért a stabilizáló programrésznek elegendő hosszabb időközönként (tapasztalat szerint megfelel a 30 másodperces időköz) ellenőriznie az irányt, és csak akkor avatkozni be a

tükörtartón keresztül, ha jelentős eltérést tapasztal a beállított pozíciótól. Ez a fenti (19. oldalon ismertetett) lépéssorozat 4. és 5. lépései közé beillesztett 30 másodperces várakozást jelent.



11. ábra A nyalábstabilizáló programrész blokkdiagramja

Ennek megvalósítása a 11. ábrán látható: a fentiekben ismertetett alap-algoritmusnak a 10. ábrán szaggatott vonallal bekeretezett részét egy bővebb ciklusból hívjuk meg. A kézi vezérlő részt is a főprogram szintjén volt a legegyszerűbb megvalósítani egy feltételes elágazás másik eseteként. Az algoritmusnak ez utóbbi kiegészítése szintén a 11. ábra blokksémáján látható.

6.3.1. Az algoritmus részletes ismertetése

Inicializáló rész

Szöveges figyelmeztetést kapunk, ha a számítógép nem érzékeli a léptetőmotort (nincs bekapcsolva vagy megszakadt a kapcsolat). A szöveges figyelmeztetés belekerül a program naplójába is; a motorvezérlő alprogram sikertelensége a főprogram futása során is generálja a figyelmeztetést. A kamera forró pixeleinek helye táblázatos fájlból egy tömbbe kerül. Habár az inicializáló részbe szokták tenni a port-megnyitásokat, stabilabb megoldás, ha a

motorvezérléshez a port megnyitása, majd lezárása nem a főprogram szintjére kerül, hanem közvetlenül az alprogramba, mert ellenkező esetben a program esetleges hibás leállása esetén a port nyitva maradhatna, és a program újraindítása előtt a számítógépet is újra kellene indítani.

Expozíció

A kamerát exponáló függvény (`camera_img.vi` és `camera_variables.vi`) eredeti változata az EDC-1000HR kamera „`hrcam32.dll`” függvénytára köré épült, melyben összesen két függvény van: az egyik beállítja a gain-t és a bias-t, a másik megadott időtartamú exponálást hajt végre, és az elkészült képet beolvassa. A `camera_variables.vi` néhány fontos szám adatot tartalmaz (képméret, port, egyébek...).

Az `camera_img.vi`-t módosítottuk azzal a céllal, hogy a hardveresen nem triggerelhető kamerát lehessen szoftveresen triggerelni. Ezt úgy valósítottuk meg, hogy egy, a 6.4.2 szakaszban ismertetett „triggerjel” változó IGAZ értékére várakozó ciklust tettünk közvetlenül az exponálást elvégző DLL-hívás elé. Ha nincs jel, akkor is le kell állnia a ciklusnak 2000 ismétlődés után (ez gyakorlatilag fél másodperc). Ha a jel hiánya miatt állt le a ciklus, akkor nem fut le az exponálás lassú külső függvénye.

Az exponálás feltétele az is, hogy más programrészek nem tiltották le kamerát. A program más, időkritikus részei (pl. a fénymegszakító adott idejű nyitva tartása) ugyanis várakoztathatják a kamerát (egy globális Boole-változóba írva, hogy engedélyezik-e a kamera exponálását vagy nem). Ennek oka: ha egyszer a kamerát exponáló függvény meghívódik, fél másodpercre nagyon lelassul a program többi része.

Ha nincs tiltva a kamera, akkor a következő eljárást követjük: mivel a számítógép és a kamera nem képes azonnal reagálni a triggerjelre, de közelítőleg tudjuk, hogy a következő impulzust a lézer 100 ms múlva fogja kiadni, egy előzőleg kikapasztalt késleltetés letelte után hívjuk meg a kamera DLL-jéből az exponáló függvényt. A módszer így sem teljesen pontos, ± 20 ms szórása van.

A fényfolt pontos helyének meghatározása

- Megkeressük a képen a maximális és a minimális intenzitású pixelt. Ha intenzitásuk különbsége nem éri el a 100 szürkeségi szintet (a 255-ből), akkor úgy tekintjük, hogy nincs folt a képen (mert az általában 200 fölötti intenzitás-különbséget okoz).
- A maximumhely körül egy 10×10 pixel méretű négyzetben fényerővel súlyozottan átlagoljuk a képpontok (x,y) koordinátáit. Ezt a koordinátapárt egy tömbben tároljuk. A súlyozott átlagból kapott aktuális maximumhelyet megjelöljük az előlapon

megjelenített képen a kurzorokkal, így könnyebben ellenőrizhető a program működése (ez a fejlesztést is jelentősen megkönnyítette).

Ezek a lépések tízszer megisméltődnek úgy, hogy első alkalommal még az egész képen fut le a maximum keresése, a többi esetben már csak az előzőleg megtalált maximumhely körül felvett 20×20 pixeles négyzetben. A nem elegendő fényt tartalmazó képek maximumhelyei nem kerülnek tárolásra és feldolgozásra. A néha előforduló hibás CCD-kiolvasás miatt látszólag félreugró lézerfolt sem okoz gondot, mert a minimum és maximum különbségét is csak ebben a 20×20 pixeles négyzetben vesszük a többi képen. Ha ebben nincs fényfolt, akkor a következő végrehajtáskor ismét az egész képet dolgozzuk fel. Erre azért van szükség, mert ha nem a CCD kiolvasása volt a hibás, hanem tényleg megváltozott a nyalábirány, így akkor is meg lehet találni a lézerfoltot.

- A 10 folt pozícióinak átlaga lesz a lézerfolt pontos helye.

Ezzel a kétféle, együtt alkalmazott átlagolással együttesen akkora pontosságot lehet elérni, hogy akár egy pixel méreténél kisebb eltérés is jól detektálható, és korrigálható (mivel a motor ilyen kicsit is tud lépni). Így az irány pontatlansága legfeljebb néhány mikroradián lehet.

Tartandó pozíció ellenőrzése

Az elkészült megoldás védett a felhasználó által megadott hibás adatokkal szemben: ha bármi úton nulla, netán negatív szám került a tartandó hely változójába, illetve ha a pozícióban valamely koordináta nagyobb, mint a CCD-kép mérete, akkor a program HAMISra állítja a „Tartani” gomb változóját, és így a program későbbi működése során biztosan nem tekeri le a képről a fényfoltot egy ilyen tévedés miatt. Egy ellenőrzőlámpa (a „status” címkéjű) jelzi a felhasználó számára, ha a stabilizálás aktív állapotban van, ami csak helyes tartandó pozíció esetén lehetséges. Hibás adat esetén az ellenőrzőlámpa is elalszik.

A tükörtartó elmozdításának kiszámítása

- A motorvezérlő meghívása előtt az átlagos lézerfolt-hely és a tartandó hely különbségét képezzük. Ez a fényfolt elvárt elmozdulásvektora az aktuális helyről a tartandó helyre.
- A tükörtartó két léptetőmotorjának lépésszáma az elmozdulás komponenseinek és a motorokra jellemző *lépésenkénti pixelváltozás* szorzata lesz. A lépésenkénti pixelváltozás nem egyforma a két léptetőmotorra, sőt egy motor pozitív vagy negatív irányú lépéseire sem. Erről bővebben a 24. oldalon, „A lépésenkénti pixelváltozás önműködő kalibrálása” szakaszban fogunk írni.

A motorvezérlő alprogram

Az Newport által a léptetőmotor-vezérlőhöz adott dinamikus függvénykönyvtár (DLL) rutinjainak meghívásával eltekeri a megadott sorszámú motort megadott számú lépéssel, majd kijelzi, hogy sikeres volt-e a forgatás.

A lépésenkénti pixelváltozás önműködő kalibrálása

A tükörtartó piezoelektromos léptetőmotorjai nem ugyanakkorát lépnek pozitív és negatív irányban, és a gyártó által is elismert 20%-os pontatlanságuk van egy irányon belül is. Ezért az igazán pontos lépésszám-számításhoz nem elegendő egy mérésből meghatározni azt, hogy egy motor-lépés hány pixelnyi elmozdulást eredményez a kamera által érzékelt fényfolt helyzetében, hanem folyamatosan mérni kell, és a mérési adatokat átlagolni. Minél pontosabb értékekkel dolgozunk, annál hamarabb éri el a nyáláb a helyes irányt.

A program alapértelmezett működéséhez olyan lépésenkénti pixelváltozás értékek vannak megadva, amelyek közel optimálisak, és a program elfogadhatóan használható velük. Ezek az értékek eredetileg maguk is ebből a programrészből származnak. Ha változik az elrendezés valamely tulajdonsága (pl. az objektívlencse fókusz távolsága, vagy másfajta kamerát, tükörtartót használunk), akkor ez a részprogram képes megállapítani a lépésenkénti pixelváltozás valódi átlagértékeit, melyek azután alapértelmezetté is tehetők.

A lépésenkénti pixelváltozást kalibráló programrész csak akkor hívódik meg, ha nincs motorhiba, és a stabilizálás aktív.

A kalibráció során folyamatosan gyűjtjük a következő adatokat:

1. előző pixelváltozás (mind X, mind Y irányban)
2. előző lépésszám (mind X, mind Y irányban)
3. korábban kiszámolt lépésenkénti pixelváltozások összege (mind X, mind Y külön a pozitív és a negatív irányban)
4. korábban kiszámolt lépésenkénti pixelváltozások darabszáma (mind X, mind Y külön a pozitív és a negatív irányban)
5. korábban kiszámolt lépésenkénti pixelváltozások átlaga (mind X, mind Y külön a pozitív és a negatív irányban)

Az összeg és a darabszám külön tárolása lehetőséget ad arra, hogy az átlagolást folyamatosan új értékekkel bővítsük. Ha az adott darabszámot növeljük eggyel és az új értéket hozzáadjuk az összeghez, az átlag mindig az összeg/darabszám lesz. Az átlagértékeket azért tároljuk kiszámolva, hogy azokat ne a felhasználás helyén kelljen külön kiszámolni. A kalibrációból származó értékek felhasználását biztonsági okokból csak akkor engedélyezzük, ha már minden irányban legalább 20 mérésből származik az átlag.

6.4. Hardveres problémák és megoldásuk

6.4.1. Forró pixelek

A nyálábirány-észlelésre használt kamera *forró pixelei* félrevezethetik az algoritmust (esetleg ezeket ismerné fel lézerfoltként). Ezen hibás pixelek felismerése és feljegyzése céljából alacsony zajú beállításokkal, alaposan lesötétített kamerával készítettünk felvételeket. Amelyik pixel sötétben is világosat jelzett, azt forrónak, vagy legalábbis hibásnak tekintettük. Mivel még mindig lehetett egy kevés alapzaj a jó pixeleken is, így csak azokat a pixeleket jegyeztük fel fájlba, amelyeken a kamera által mért szürkeségi szint meghaladta a 25-öt a 256-os skálán. Ennek a kb. 10%-os értéknek a megválasztását az indokolta, hogy ez tapasztalat szerint magasabb, mint bármilyen zaj, így nem fogjuk a zajból eredő kicsiny, mindig változó helyen megjelenő szürkeséget hibás pixelnek venni. Ha pedig az érték nagyobb lenne, túl kevés forró pixelt tudnánk csak kiküszöbölni.

A forró pixelek helyének ismeretében a zavaró hatásuk könnyen kiküszöbölhető: a kamerából kiolvasott minden egyes képen szoftveresen le lehet sötétíteni őket úgy, hogy az eredményül kapott adattömbben még a képfeldolgozás előtt nullával írjuk felül a forró pixelek értékeit. Ezek után a lézerfolt megtalálását, mely maximumkereséssel történik, már nem zavarhatják meg.

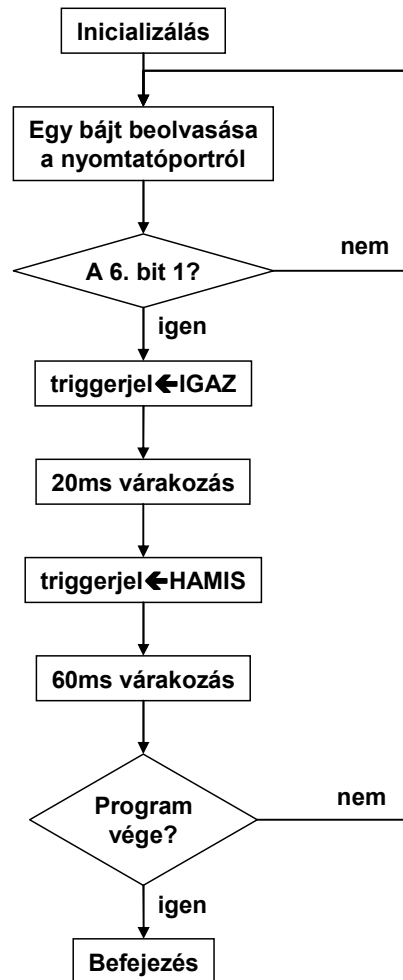
6.4.2. Az expozíció triggerelése

A TeWaTi 10 Hz-es ismétlési frekvenciájú impulzslézer-rendszer, vagyis az impulzusok viszonylag hosszú (100 ms időközönként) követik egymást. Ha 100 ms-nak választanánk a kamera expozíciós idejét, akkor azon időtartam alatt biztosan érkezne lézerimpulzus a kamerára, de a háttérfény hosszú integrálása jelentősen lerontaná a jel-zaj viszonyt. Ezért rövidebb expozíciós időre van szükség, de meg kell oldani, hogy a rövidebb időtartam alatt is biztosan érkezzen fényimpulzus a detektorra. A kamera exponálását ezért az impulzusokhoz kell szinkronizálni, *triggerelni*.

Bár léteznek különleges, triggerelhető kamerák, sőt triggerelhető illesztőkártyák is, azonban számunkra ilyenek nem álltak rendelkezésre: szabványos PC-t és hardveresen nem triggerelhető kamerát kellett használnunk.

A lézerrendszer részeinek időbeli szinkronizálására szolgáló egyik impulzusgenerátor (gyártó: Stanford Research Inc. típus: DG535) egyik kimenetével kellett összehangolnunk a program futását, ezen belül a kamera exponálását. Ehhez azt használtuk ki, hogy a nyomtatóport bizonyos érintkezőire érkező feszültséget a számítógép ki tudja olvasni. Ezek

közül mi a 12. tűhöz kötöttük a triggervezetékét, az egyik földelő-érintkezőhöz (a 24.-hez) pedig az impulzusgenerátor földjét. Ebben a kiépítésben a 0x379-es portról beolvasott bájtnál a 6. bitjének 1 értéke utal a magas TTL bemenetre, a 0 érték pedig az alacsonyra.



12. ábra A triggerjel figyelésére szolgáló programrészlet blokk-sémája

A lézerrendszer vezérlőprogramjának része egy ciklus, amelyet csak a program végén szakad meg, ennek magjában a következő lépések különíthetők el (12. ábra):

Először megvárjuk a triggerjelet: addig olvassuk a párhuzamos portról a bájtokat, amíg a 6. bit értéke magasnak nem bizonyul (vagy véget nem ér a program). Ekkor a „triggerjel” globális igaz-hamis változót IGAZra állítjuk. Ebből az IGAZ értékből tudja a többi programrész kiolvasni, hogy a triggerjel megérkezett.

Második lépésben 20 ms-ig várunk, utána HAMISra állítjuk a „triggerjel” változót, jelezvén a többi programrésznek, hogy a korábbi triggerjel már nem aktuális, meg kell várniuk a következőt.

Harmadik lépésben 60 ms várakozás után előről kezdődik a triggerre várakozás a nyomtatóport folyamatos beolvasásával. A 60 ms-os várakozás azért hasznos, mert abban az

időtartamban nem várható magas feszültség a triggervezetéken (az egész folyamat periódusideje a 10 Hz-es ismétlési frekvenciájából következően 100 ms), a port-olvasások biztosan eredménytelenek lennének; előnyösebb várakozni, mint a számítógép erőforrásait ezzel terhelni más programrészek rovására.

6.5. *A nyalábstabilizálás elért eredményei*

Eleget tettünk a 6.1 szakaszban megfogalmazott követelményeknek. Ezen kívül a következő eredményeket is sikerült elérnünk:

- Megfelelő átlagszámítások beépítésével a lézerfolt helyének meghatározási pontatlansága kevesebb, mint 1 pixel a kamerán.
- A program képes alkalmazkodni egy megváltozott elrendezéshez (pl. másik objektív-lencséhez) is a negatív visszacsatolás megfelelő mértékének önálló megállapítása révén.
- A léptetőmotor irányváltásból fakadó hiszterézisét valamint az irányfüggő lépési nagyságát korrigáltuk.
- A kamera ún. forró pixelei okozta hiba illetve zaj kiszűrése.
- A ritkán előforduló hibás CCD-kiolvasást, mikor a kép teljesen más részén jelenik meg a lézerfolt, észleljük, és a hibás képet nem vesszük figyelembe.

6.6. *Az elért pontosság megbecslése*

A léptetőmotor 30nm-es lépésfinomsága korlátot szab a tükör beállítási pontosságára (l. 13. ábra):

$$\Delta\phi_{\max, motor} = \frac{\text{lépésnagyság}}{\text{a motor távolsága a forgástengelytől}} = \frac{30\text{nm}}{42\text{mm}} \approx 0,8\mu\text{rad}.$$



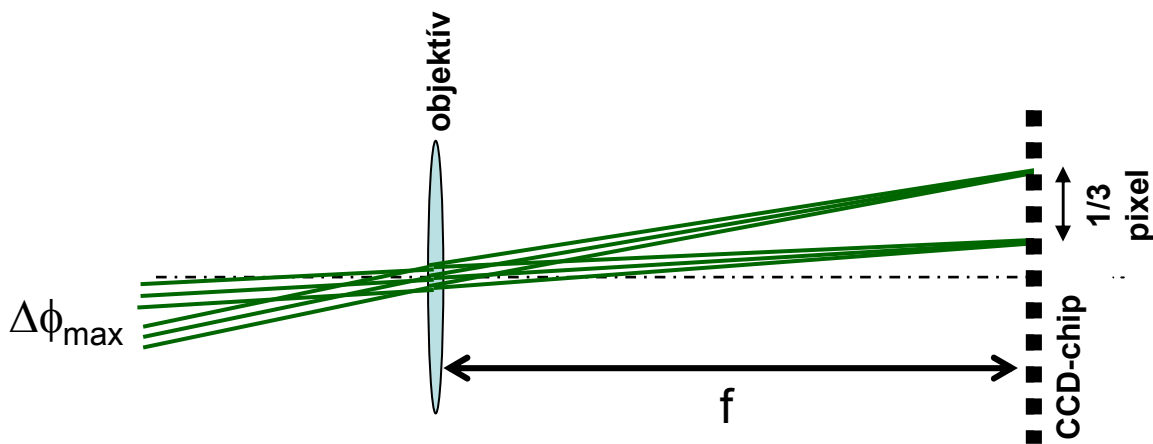
13. ábra A tükörtartó beállítási pontosságához

A fénysugár beállítási pontossága a tükör beállítási pontosságának kétszerese.

A CCD-kamerával figyeljük, hogy a fókuszált fényfolt pozíciója mennyire tér el a referenciahelytől. A fényfolt helyét a tapasztalat szerint kb. 1/3 pixel pontossággal tudjuk meghatározni, azaz ha a program 1/3 pixeles kiszámított eltérés esetén avatkozik be, még megbízható a stabilizáló algoritmus működése. Ez a 27 μ m-es pixelmérettel és a 0,5m-es objektív-kamera távolsággal (l. 14. ábra)

$$\Delta\phi_{\max} = \frac{\text{a pozíciómeghatározás pontossága}}{\text{az objektív és a kamera távolsága}} = \frac{\frac{1}{3} \text{ pixel} \cdot 27 \frac{\mu\text{m}}{\text{pixel}}}{0,5\text{m}} \approx 18 \mu\text{rad}$$

szögbeállítási pontosságot jelent, amit — mint fentebb kiszámítottuk — a léptetőmotor tud is teljesíteni.



14. ábra A nyalábirány beállítási pontosságához

7. Nyalábenergia-vezérlés

A nyalábok energiáját a 2.3 szakaszban leírtak szerint szabályozzuk. Az elforgatható félhullám-lemezeket a gyors, kényelmes, biztonságos és jól reprodukálható forgatás érdekében számítógéppel vezérelhető forgatómotorokra (gyártó: OWIS GmbH típus: DRT 40) építettük (15. ábra).



15. ábra Az OWIS DRT 40 forgatómotor a rászertelt félhullám-lemezzel

7.1. *Követelmények a programmal szemben*

A rendszerbe két motor van beépítve, ezért szükség van rá, hogy egyértelmű legyen ezek kiválasztása a program előlapján.

A vezérlési módnak (botkormány segítségével vagy az előlapon) választhatónak kell lennie.

A motoroknak botkormányos vezérlés esetén folyamatosan, lassan kell forogniuk. A precíz és gyors beállíthatóság érdekében a forgatásnak két sebességfokozatúnak kell lennie. (Erre főleg akkor van szükség, ha a kutató nem ismeri előre a megfelelő intenzitás értékét, és az elrendezést figyelve keresi meg az ideális beállítást.)

Előlapon keresztül vezérelve pedig egy szövegmezőben számmal megadott pozícióba kell fordulniuk a motoroknak.

A túlságosan nagy mértékű elforgatás megelőzése érdekében korlát beépítésére van szükség. Erre azért van szükség, hogy a félhullámlemezek helytelen állásszögével ne lehessen az eszközöket károsító intenzitást elérni.

7.2. *A programrész működtetése*

A felhasználó legördülő menüből választja ki, melyik motort szeretné elforgatni (16. ábra).

Az előlapon nyomon követhető a motorok állapota. Mindkét motorhoz 3-3 kijelző tartozik. Ezek rendre jelzik a motor aktuális helyzetét, a célszöveget, ahová a motor forogni fog és azt, hogy a motor épp mozog-e vagy áll.

A „Reset actual attenuator” gomb benyomása esetén az aktuális motor azonnal visszaforgó a kiindulási helyzetébe. Ez a megoldás az esetleges téves beállítások esetén csökkenti azt az időtartamot, amíg túl nagy intenzitás éri a kísérleti eszközöket.

A túlságosan nagy mértékű elforgatás megelőzése érdekében korlátokat építettünk be, ezt *határszögnek* neveztük el. A határszögek állíthatóak, de az előlap panel másik „fülén” („Attenuator limits” fül), mint ahol a forgatás maga látható. Ez a megoldás lehetővé teszi, hogy csak a lézerrendszer tervezői illetve a közvetlen kezelők tudják állítani a szöveget, külső felhasználók nem férnek hozzá, így akaratlanul sem tudnak kárt okozni a rossz beállítás eredményezte túl nagy intenzitással.

Nyomógomb segítségével lehet választani a botkormányos- és az előlapon történő vezérlés között.

Előlapos vezérlés esetén a „Position (deg)” szövegmezőbe beírható a kívánt szög, és az „Enter” megnyomása után a kiválasztott motor a beállított értékre áll. Ha a vezérlést váltjuk a két motor között, akkor az előzőleg beállított értékek megőrződnek. Így lehetőség nyílik a két fénypálya egyidejű szabályozására, ezáltal egyszerre többet is dolgozhatnak a lézerrel.

Botkormány-vezérlés esetén a forgás irányát a botkormány megdöntésének iránya határozza meg. Előre illetve jobbra döntés esetén pozitív irányban fordul a motor, hátra, illetve balra döntés esetén negatív irányban. Az elrendezés alapja a Descartes-féle derékszögű koordináta-rendszer, ez könnyen megjegyezhetővé teszi az irányítást.

A botkormány megdöntésének mértékétől függően $0,1^\circ$ -onként vagy 1° -onként fordul a kiválasztott motor. Az adott irányú lehetséges legnagyobb megdöntés feléig $0,1^\circ$ -onként, ennél nagyobb döntés esetén pedig 1° -onként forog a motor a megfelelő irányban.



16. ábra A nyalábenergia- és a fénymegszakító-vezérlő programrész közös előlapja

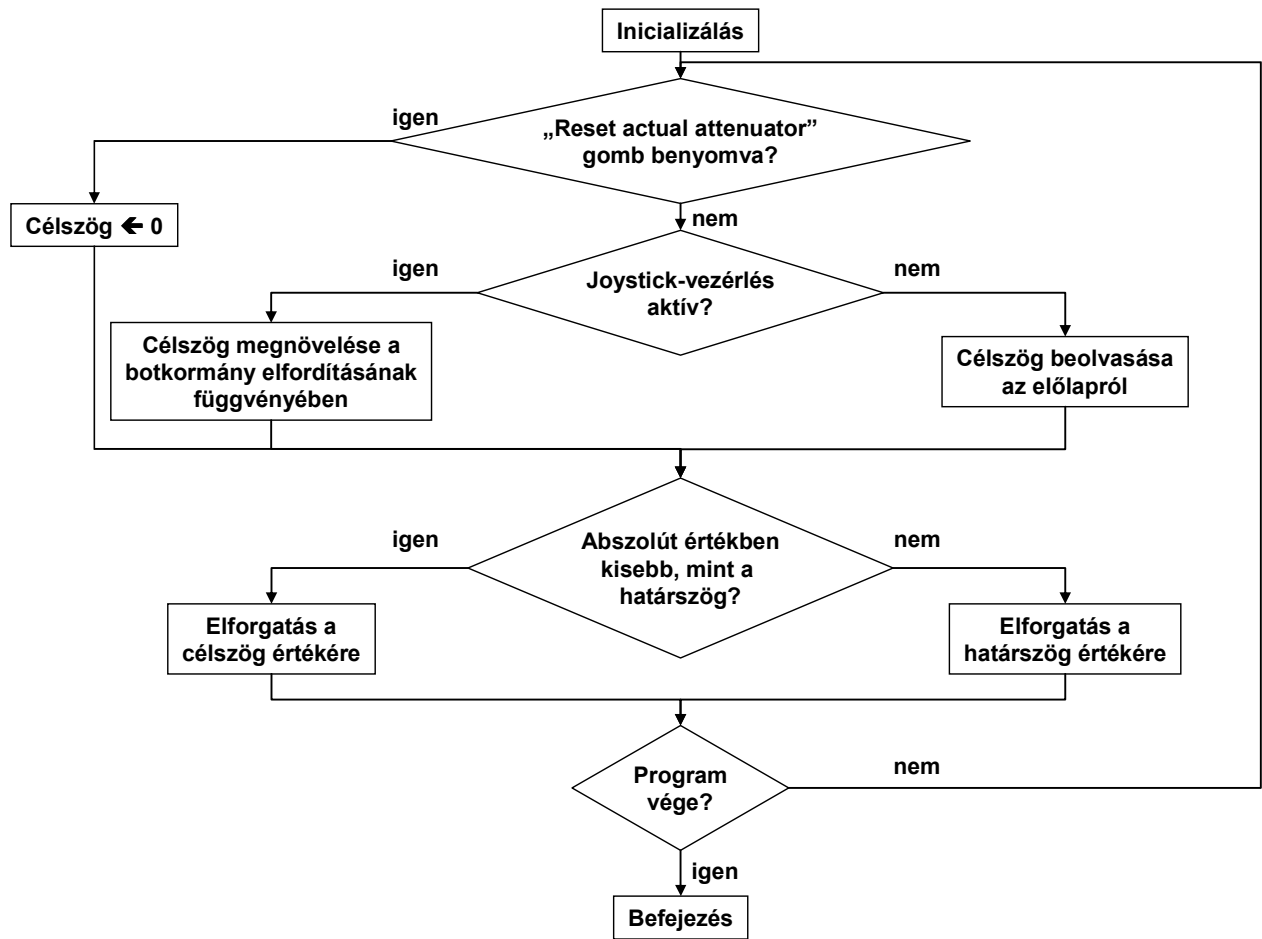
Kísérleteztünk egy olyan programverzióval is, amelyben a megdöntés mértékének függvényeként folytonosan változott a forgatás sebessége. A tesztelés során azonban ezt elvetették a felhasználók, mert a forgatás sebessége túl érzékeny volt a megdöntés mértékére, és így nehezen tudtak pontos értéket beállítani. A végül a 2 sebességfokozatú folyamatos forgatásra épülő megoldás állta ki a felhasználói teszteket.

7.3. A nyalábenergia-vezérlés megoldásának elve

A motorok forgatására adott volt egy (a gyártó által mellékelte) LabVIEW-ben írt program. Ezt alprogrammá alakítottuk át, és így használtuk fel a továbbiakban. Az átalakítás két fő momentuma: az ún. konnektorok elkészítése, hogy az alprogram subVI-ként is meghívható legyen, illetve a bekért paraméterek számának redukálása.

A lézerrendszer vezérlőprogramjának a nyalábenergia-szabályozásért felelős része a 17. ábrán tekinthető át blokkdiagram formájában.

Előlapos vezérlés esetén a célszöget összevetjük a határszöggel. Ha megadott határnál kisebb a célszög, akkor értékét továbbadjuk a forgató subVI-nak. Ellenkező esetben a határszög értékére (negatív szög esetén értelemszerűen a -1-szeresére) áll be a motor.



17. ábra A nyalábenergia-vezérlés blokkdiagramja

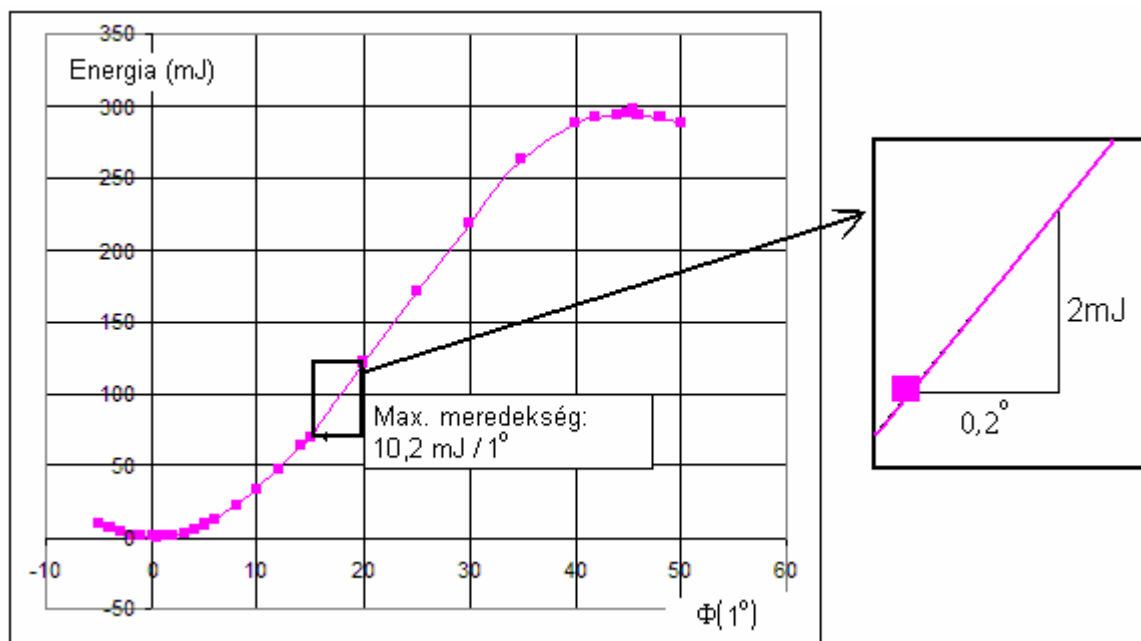
Botkormányos vezérlés esetén kétféle, egy gyors (1° -os lépésközű) és egy lassú ($0,1^\circ$ -os lépésközű) forgatási sebességfokozatot valósítottunk meg a botkormány döntésének mértékében. A forgatómotorok számára beállítandó új szögek kiszámítása a következő lépésekben történik:

1. Definiáljuk a „forgatás mértéke” lokális változót a botkormány megdöntésének függvényében, aminek lehetséges értékei a fentebb ismertetett módon $\pm 0,1^\circ$ és $\pm 1^\circ$.
2. Ellenőrizzük, hogy aktív-e a botkormányos vezérlés. Ha igen, akkor az aktuális szögállás és a „forgatás mértéke” változó összegét tekintjük a következő elforgatás céljának. Ha nem, akkor a billentyűzeten keresztül a „Position (deg)” változóba bevitt értéket.
3. Következő lépésben a forgatási cél értékét vetjük össze a kiválasztott motor beállított határszögével. Ha a kívánt elforgatás mértéke nem haladja meg a kritikus értéket, akkor továbbadjuk a forgató subVI-nak, és a motor beáll a megfelelő pozícióba. Ha a határszögnél nagyobb a kiszámított elforgatási cél, akkor a határszög eléréséig fordul el a motor, ezután csak az ellenkező irányú vezérlésre reagál.

Problémát jelentett, hogy a botkormány alapállapotban ill. elengedés után megállapodva nem pontosan középen helyezkedik el. Ezért mind a négy irány esetén külön kellett definiálni a maximális kitérés értékét. Ehhez képest állapítottuk meg, hogy az egyes irányok esetén a botkormány milyen mértékű megdöntéséig forogjon a lassabb sebességfokozatban a motor (az adott irányú maximális kitérés feléig). Az elmozdítás első 1-1%-ára nem reagál a program, így nem okoz problémát, ha elengedés után a botkormány nem pontosan az alaphelyzetbe áll vissza.

7.4. Az elért pontosság megbecslése

A forgatómotor állásszöge a gyártó specifikációja szerint legalább $\Delta\phi_{\max} = 0,2^\circ$ -ra pontosan reprodukálható. A második erősítőfokozatban az állásszög—impulzusenergia függvény maximális meredeksége $\left. \frac{dE}{d\phi} \right|_{\max} = 10,2 \frac{mJ}{1^\circ}$, azaz a forgatómotor és így a félhullámlemez 1° -os szögelfordulásának hatására az impulzusenergia legfeljebb 10,2mJ-t változik (18. ábra). Ebből az következik, hogy az energiabeállítás reprodukálhatósága a legrosszabb esetben is $\Delta E_{\max} = \left. \frac{dE}{d\phi} \right|_{\max} \cdot \Delta\phi_{\max} = 10,2 \frac{mJ}{1^\circ} \cdot 0,2^\circ \approx 2mJ$.



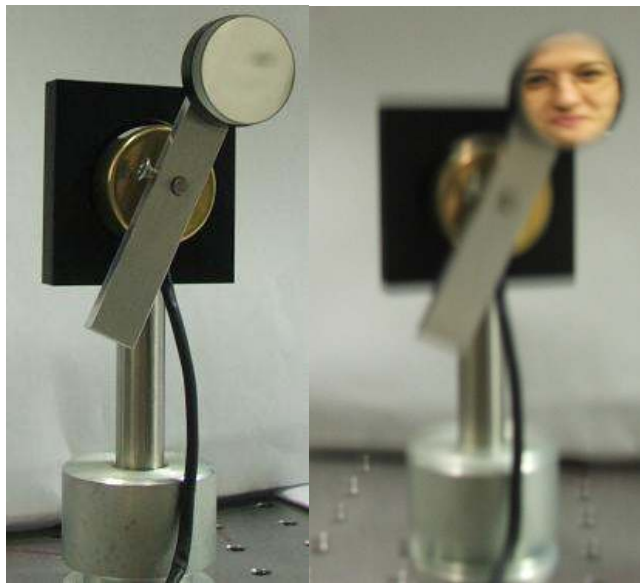
18. ábra A második erősítőfokozat pumpájának impulzusenergiája a félhullámlemez állásszögének függvényébe

Mivel az anyagtudományi kísérletek az állásszög—impulzusenergia függvény kisebb meredekségű (70mJ impulzusenergia alatti) szakaszain folynak, ezért ott a reprodukálhatóság még jobb, kb. 1mJ-nak tekinthető.

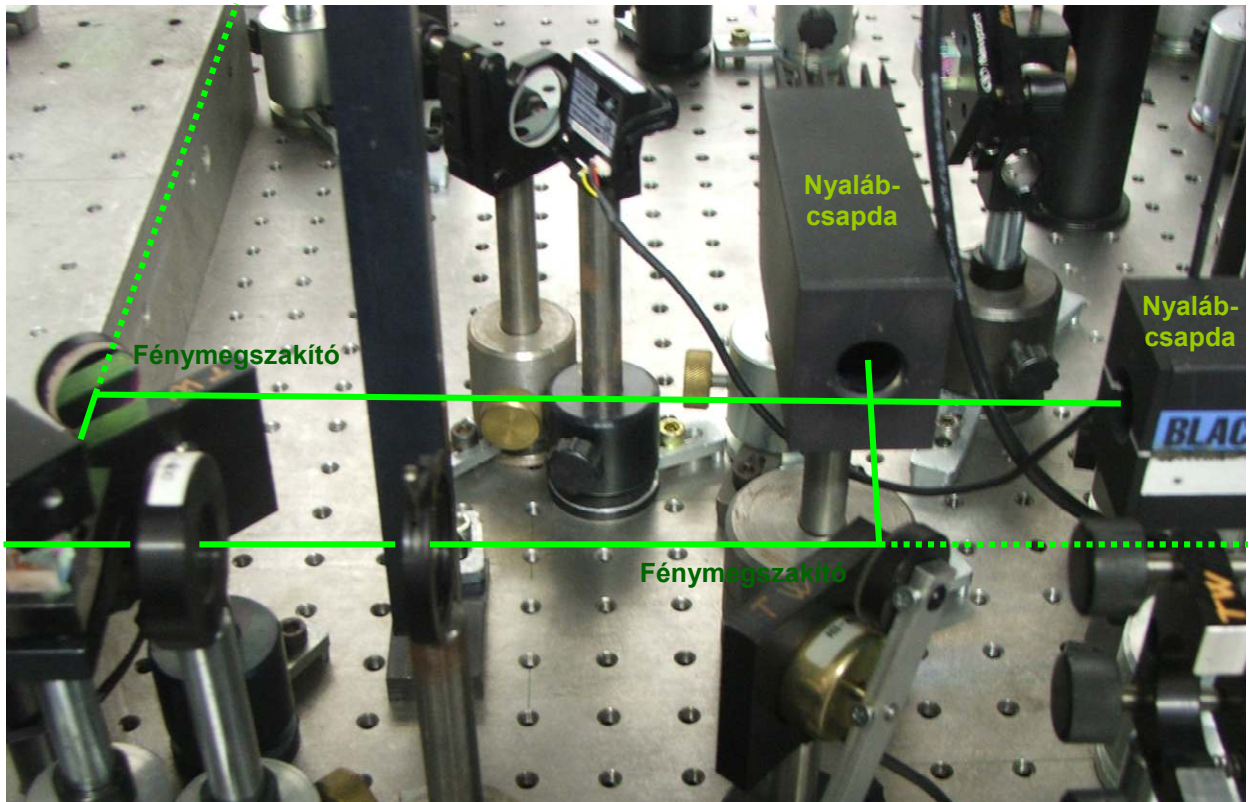
8. Fénymegszakító-vezérlés

A fénymegszakító tulajdonképpen egy csapómotor, ami egy rudacsán lévő tükröt forgat (19. és 20. ábrák). A fényt elzárását ezzel úgy oldjuk meg, hogy zárt állásban a tükör a fényt egy nyalábsapdába továbbítja. Nyitott állapotban a motor a tükröt kiforgatja a fényútból.

Összesen négy fénymegszakítót építettünk a rendszerbe, ezek közül három a Nd:YAG lézer pumpáló fénynyalábját nyitja-zárja (ezek közül kettő látható az 1. egyszerűsített ábrán).



19. ábra Fénymegszakítók



20. ábra Fénymegszakítók és nyalábcsapdák beépítve

8.1. Követelmények a programmal szemben

A LabVIEW programnak a motorokat vezérlő mikrokontrollerrel kell kommunikálnia egy egyszerű soros portos protokollal.

- Az elvárás szerint ezeket a fénymegszakítókat lehet vezérelni a program előlapjával és a rendszerbe beépített botkormány tűzgombjával is.
- Botkormány-vezérlés esetén egy bizonyos, a felhasználó által kiválasztott fénymegszakítót lehet bekapcsolni, hogy az majd egy előre beállított idő eltelte után automatikusan kikapcsoljon. Ezzel lehetővé válik, hogy a fénymegszakító pontosan meghatározott számú lézerpulzust engedjen át.
- Meg kell oldani, hogy a vezérlési mód megváltoztatása esetén a fénymegszakítók megőrizzék aktuális állapotukat. Hibás működés esetén a fénymegszakító ki- és bekapcsolhatna a vezérlési mód megváltoztatásakor, ez pedig tönkre teheti a mérést és az eszközöket is.
- Az előlapról működni kell mind a négy fénymegszakító ki- és bekapcsolásának, míg az egérrel mindig felülírható a botkormánnyal beállított állapot.

8.2. A programrész működtetése

Nyomógomb segítségével engedélyezhető a botkormányos-vezérlés (lásd 16. ábra).

Mind a négy fénymegszakító vezérelhető az előlap segítségével (ki- és bekapcsolható, megmarad a beállított állapotban).

Ha engedélyezzük a botkormány-vezérlést, akkor fontos kiválasztani a „Select shutter for joystick control” legördülő menüből, hogy melyik fénymegszakítót vezéreljük a tűzgombbal. Ekkor állítani lehet az átengedett lövések számát a „Number of shots to yield” szövegmezőben. Ez alapértelmezésben 10, a 0 lövésszámnak speciális jelentése van: ilyenkor a tűzgomb megnyomására a fénymegszakító egyszerűen átkapcsol anélkül, hogy utána visszaállna az eredeti állapotába.

Az „Emergency close” gomb megnyomásakor az összes fénymegszakító azonnal alapállapotba kerül. Ez akkor lehet különösen fontos, ha az átkapcsolás idejét túl hosszúra állították, és a lézer kárt okozhatna a kísérleti elrendezés elemeiben.

8.3. A fénymegszakító-vezérlés megoldásának elve

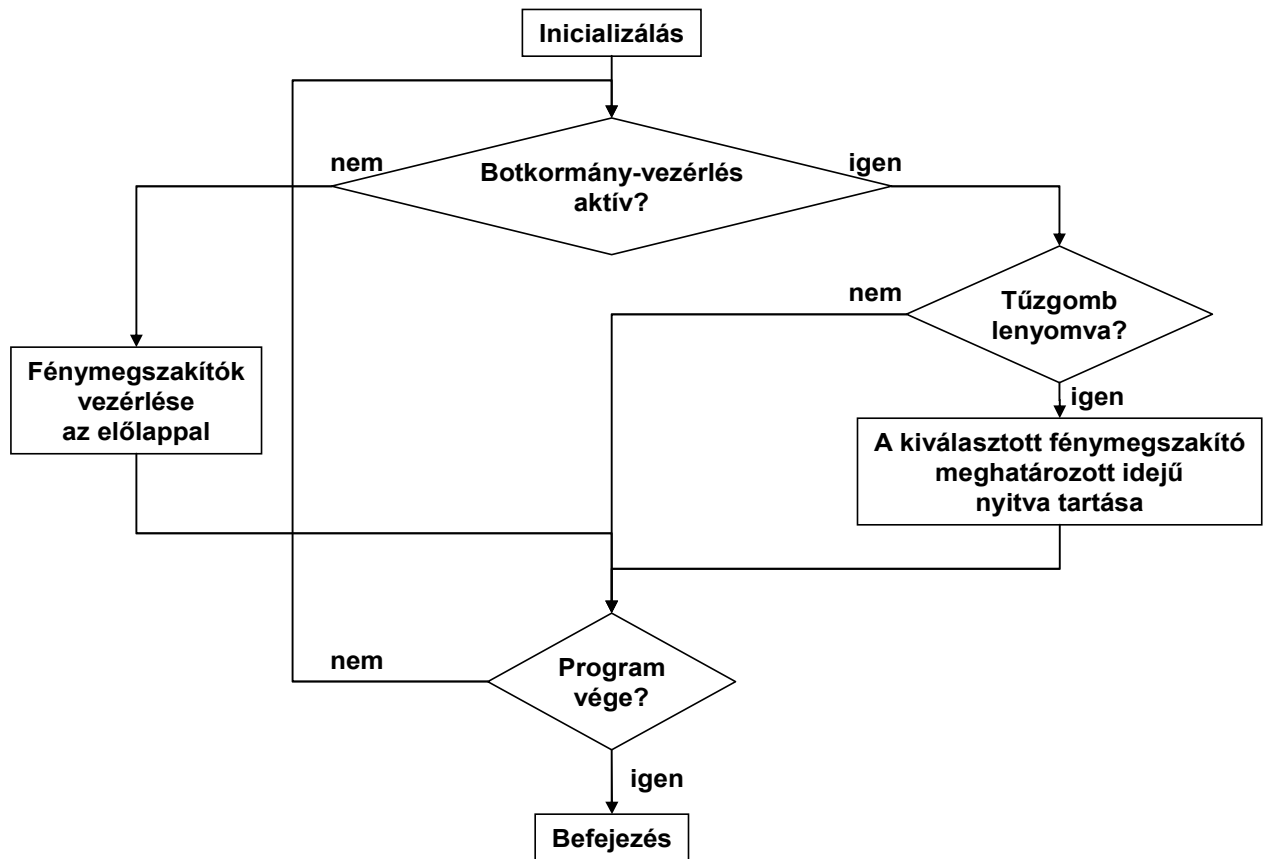
Mind a négy fénymegszakítónak megfeleltettünk egy-egy igaz-hamis (Boole) változót, melyek az előlapon kapcsolóként jelennek meg.

A ciklus felépítése a 21. ábrán található blokkdiagramon követhető nyomon.

A fénymegszakítókat vezérlő mikrokontrollerhez alprogramot írtunk. Ennek igaz-hamis változók formájában megadható a négy fénymegszakító kívánt állapota. Ezt a subVI-t hívjuk meg, amikor kapcsoljuk valamelyik fénymegszakítót.

A programrészlet minden egyes lefutás során ellenőrzi, hogy az előlapon keresztül módosították-e a fénymegszakítók beállításait. Ha igen, akkor az új beállításokat átadja a vezérlő mikrokontrollernek.

A program további futása botkormány-vezérlés engedélyezettségétől függ, mert ilyenkor a tűzgomb megnyomása esetén a fénymegszakító adott idejű átkapcsolása következik. Aktív botkormány-vezérlés esetén a botkormány tűzgombjának lenyomottságához tartozó logikai változó IGAZ értékére vár a programrész. Ezen a ponton a program leállna, és addig várna, amíg nem nyomják meg a tűzgombot. Ezért, ha nem nyomják le a gombot, a ciklus 10000-szeri lefutás után továbblép, majd ismét visszatér ehhez a ponthoz.



21. ábra A fénymegszakító-vezérlő programrész bloksémája

Gombnyomás után, a program még megvárja a következő triggerjelet a 6.4.2 szakaszban ismertetett módon, és csak ezután kapcsolja át a kívánt fénymegszakítót. Így pontosan akkor kapcsolja át a fénymegszakítót a program, amikor egy új impulzus érkezik, és nem akkor, amikor a felhasználó megnyomja a tűzgombot.

Ezután az adott impulzusszámhoz 0,1 s hosszúságú várakozás következik (mivel a lézer impulzusai 0,1 s-onként követik egymást), majd a fénymegszakító visszaáll a kiinduló állapotba.

0 impulzusszám esetén a tűzgomb lenyomása után a fénymegszakító átkapcsol, és az új állapotban marad a következő utasításig.

A nyalábirány-stabilizálás időnként lelassítja a program futását, itt viszont az időtartamok nagyon pontos betartása szükséges. Ezért, ha lenyomták a tűzgombot, akkor a program letiltja a nyalábstabilizálást a kritikus időtartamra.

9. Adatmentés és naplózás

9.1. *Követelmények a programmal szemben*

A számítógép lefagyása esetén szükség van arra, hogy tudjuk, mik voltak az utolsó beállítások.

9.2. *A programrész működése*

A program futás közben fájlba menti a szükséges paramétereket. Bármely változás esetén a meglévő adatfájlhoz hozzáír egy újabb sort.

9.3. *A megoldás elve*

A mentett adatok rendre: a két forgatómotor állapota (szögben), a fénymegszakító állapota (ki- vagy bekapcsolt, 4 változó 0-1 értékekkel), az oszcillátor feszültsége (V). Minden egyes adathoz hozzárendeltünk egy shift-regisztert. Ezzel a megoldással a ciklus minden egyes lefutásánál ellenőrizhető a változó értéke. Összehasonlítjuk a változó értékét az előző futásbeli értékkel; változás esetén az új értéket üzenetben elküldjük magyarázattal kiegészítve, a LabVIEW beépített „Send Notification.vi” függvényét használva. Egy másik programrészben a „Wait for notification.vi” függvény várja a változásokról szóló üzeneteket. Az üzeneteket azonnal fájlba mentjük új sorként, a sor elején a dátum és az idő szerepel. Az állományba mentés mellett a „Log” címkéjű programfülön megjelenik minden, az adott futás során keletkezett üzenet, szintén időponttal együtt, emellett a legutolsó üzenet a program kezelőfelülete alatt mindig látható. A rendszer rendellenes leállása esetén a labor munkatársai könnyen elérik a szükséges adatokat, és így nyomon tudják követni, hogy hogyan változtak a lézer beállításai. Ezek a fájlok a mérések dokumentálását is elősegíthetik.

10.Összefoglalás

A programfejlesztés alapjául szolgáló főbb célkitűzéseink a következők voltak: az indítás automatizálása, a lézernyaláb irányának stabilizálása, a nyalábenergia pontos szabályozása, a fényutak könnyű és gyors elzárásának biztosítása, botkormány segítségével a beállítás megkönnyítése, a leállítás segítése, naplózás. Elért eredményeinket az alábbiakban összegezzük.

- A kifejlesztett aktív nyalábirány-stabilizáló részrendszer segítségével a nyaláb iránytartását 0,02 mrad-on belül tartjuk, ami a lézer saját specifikációját (0,5 mrad) jelentősen megjavítja.
- Az aktívan vezérelt léptetőmotorok által forgatott félhullám-lemezekkel az energiabeállítás pontossága és reprodukálhatósága 1 mJ-ra növekedett a korábbi 15 mJ-lal szemben.
- Fénymegszakítóink használatával a lézerrendszer főbb fényútjai könnyen és gyorsan elzárhatók, illetve szabaddá tehetőek. A fénymegszakítók nyitva tartásának időtartama 0,1 s pontossággal beállítható.
- A létrehozott botkormányos vezérlés segítségével a fénymegszakítás és a nyalábok intenzitása az optikai asztal mellett állva is irányítható; nem szükséges, hogy a kutató a vezérlő-számítógéphez menjen, ha módosításra van szüksége.
- Az automatikus adatmentés és naplózás lehetővé teszi, hogy pontosan nyomon tudjuk követni a beállításokat. Ez segítséget jelenthet a számítógép lefagyása esetén; a beállítások gyors és pontos reprodukálása megkönnyíti a munka folytatását.

A kitűzött célokat megvalósítottuk. A programot azonban folyamatosan fejlesztjük a felmerülő felhasználói igényeknek megfelelően, hogy minél inkább megteremtjük a laborban a biztonságos, pontos és kényelmes munkavégzés feltételeit.

Irodalomjegyzék

1. Osvay K.: *Ultrarövid lézerimpulzusok lineáris terjedése, összenyomása és fókuszálása* in: *A kvantumoptika és –elektronika legújabb eredményei*, (szerk: Heiner Zs., Osvay K.), 2006
2. K. Osvay, A. P. Kovács, Zs. Heiner, M. Csatári, Zs. Bor, G. Kurdi, M. Görbe, J. Klebniczki, I. E. Ferincz: *A table-top high contrast TW laser system*, CLEO/Europe-EQEC, 2005, Munich, paper CG-13-TUE
3. Osvay K., Kovács A., Kurdi G., Bor Zs.: *Magyar Tudomány*, 2005/12., 1511-1516
4. Cs. Vass, K. Osvay, B. Hopp: *Fabrication of 150 nm period grating in fused silica by two-beam interferometric laser induced backside wet etching method* *Opt. Expr.* **14** (2006) 8354-8359
5. M. Csete, G. Szekeres, Cs. Vass, N. Maghelli, K. Osvay, Zs. Bor, M. Pietralla, O. Marti: *Surface plasmon resonance spectroscopy on rotated sub-micrometer polymer gratings generated by UV-laser based two-beam interference* *Appl. Surf. Sci.* **252** (2006) 4773-4780
6. Kovács A.: *Optikai elemek fázistulajdonságainak interferometrikus vizsgálata*, (PhD értekezés), 2000
7. <http://www.noise.physx.u-szeged.hu/digitalmeasurements/LabVIEWTutor/default.htm>

Köszönetnyilvánítás

A projektet az OTKA T47078 és az NKFP 1/0007/2005 pályázatok támogatták. Köszönetünket fejezzük Dr. Gingl Zoltánnak és munkatársainak értékes tanácsaikért. Görbe Mihálynak a gondos és figyelmes témavezetésért, Dr. Osvay Károlynak szakmai útmutatásáért, a TeWaTi labor munkatársainak a segítségért és a türelméért.