LabVIEW II. jegyzőkönyv

Csomai Borbála

(Laborvezető: Tihanyi Attila, Sántha Levente, labortárs: Faragó Teodóra) Pázmány Péter Katolikus Egyetem,Információs Technológiai és Bionikai Kar 1083, Budapest, Práter utca 50/a 2018.03.05. 16:15-18:00 csomai.borbala@hallgato.ppke.hu

Abstract—Ezen jegyzőkönyv összegzi a LabVIEW mérés során elkészített programok leírását, valamint a mérés során tapasztaltakat.

I. Első Feladat

Az első feladat megoldásakor elhelyeztünk a front panelen egy boolean típusú nyomógombot, majd egy LED-et amelynek fényét sárgára állítottuk. Ezt követően, készítettünk egy "km/h" nevű numerikus vezérlőt, majd egy "m/s" feliratú kijelzőt. Ezután a block diagramon vezetékekkel összekapcsoltuk a nyomógombot a LED-del. A "km/h" és a "m/s" között egy 3,6-os konstanssal való osztással teremtettünk kapcsolatot. A program futtatásakor a nyomógomb megnyomásával a LED fénye felvillant, valamint megtörtént a mértékegységátváltás. Megfigyeltük, hogy a két folyamat egymástól teljesen függetlenül fut le.



Fig. 1. 1. feladar

II. MÁSODIK FELADAT

A második feladat elkészítésekor az első feladat megoldását módosítottuk. Először a panel nyomógombját lecseréltük egy kapcsolóra. Ezt követően a diagramon készítettünk egy elágazást (case structure-t), aminek false ágában elhelyeztük az osztást. A bemenetet hozzákapcsoltuk a kapcsolót és a LED-et összekötő vezetékkel. A m/s kijelzőt hozzákapcsoltuk a case structure kimeneti oldalához. A true ágban egy végtelen konstanst helyeztünk el. Végül a false ágban lévő osztást elhelyeztük egy subvi-ben. (LabVIEW-ban a subvi segítségével a program egy része egyetlen grafikus elemmel helyettesíthető.) A módosítás eredménye, hogy a mértékegységváltás csak a kapcsoló kikapcsolt állapotában történik meg.



Fig. 2. 2. feladat

III. HARMADIK FELADAT

A harmadik feladatban egy kockajáték szimulációt hoztunk létre. Ügyelnünk kellett arra, hogy a dobókockák ne tudjanak hamis értékeket adni. Egy hétköznapi dobókockával 1 és 6 közötti egész számokat tudunk dobni ugyanakkora esélyekkel. Első lépésként elhelyeztünk két randomszám generátort a block diagramon. Ezek a [0;1) intervallumból generálnak valós számokat. Először, ezt a random számot beszoroztuk 6tal, inkrementáltuk, majd a kapott értéket lefelé kerekítettük. Az eredményként kapott egész számok így minden esetben a [1,6] intervallumba estek. A kapott két számot összeadtuk, majd az "=" operátorral ellenőriztük, hogy az eredmény egyenlő e 7-tel. Ezután a elhelyeztünk a front panelen egy nyomógombot és egy LED-et. A diagramon összekötöttük ezt a LED-et az egyenlőségjel kimenetével, ezzel elérve, hogy a LED kigyulladjon, ha a dobott számok összege 7. Utolsó lépésként a block diagramon szereplő összes elemet behelyeztük egy case structure true ágába,

majd a bemenetet összekötöttük a nyomógombbal. Futtatásakor, a nyomógombot lenyomására a program ellenőrizte, hogy a dobott számok összege egyenlő e 7-tel, ha egyenlő, felvillant a zöld LED. (Két dobókockával való dobáskor, a kapott számok összege legnagyobb valószínűséggel 7, kb. 16,67% esély van rá.)



Fig. 3. 3. feladat

IV. NEGYEDIK FELADAT

Ebben a feladatban megmértük a 2. feladatban elkészített mértékegységváltó subvi futási sebességét. Első lépésként elhelyeztük a subvi-t egy for ciklusban, melyben az N-et 100 000 000ra állítottuk, így biztosítva, hogy a mérési hiba a lehető legkisebb legyen. Ahhoz, hogy maximum 1% eltéréssel kapjuk meg az eredményt, legalább 100-szor le kell futni a ciklusnak. Ezek után készítettünk egy Flat Sequence-t három ablakkal. A középső ablakba behelyeztük a ciklust, a két szélsőbe pedig egy-egy időmérőt tettünk. Az első óra a program futásának kezdetét méri le, ezt követően megtörténik a mértékegységváltás N-szer, majd a harmadik ablakban lévő óra leméri a program lefutásának idejét. A két mért időt kivontuk egymásból, megkapva a ciklus futásának idejét, majd a kapott eredményt leosztottuk N értékével. A kapott számot (milliszekundum) kiírattuk a Front Panel-re egy numerikus indicator felhasználásával.

V. ÖTÖDIK FELADAT

Az ötödik feladatban az emberek reakció idejének mérésére készítettünk egy programot. A Front Panel-en elhelyeztünk egy LED-et, egy kapcsolót és egy numerikus indicator-t, mely a reakcióidőt jelzi ki milliszekundumban. E feladat megoldásakor is egy Flat Sequence-t használtunk, így biztosítva a folyamatok egymásutániságát. Négy ablakot hoztunk létre. Az elsőbe egy random időgenerátort helyeztünk. Ezt úgy készítettük el, hogy a dobókocka segítségével generáltunk egy random számot a [0;1) intervallumban. A kapott



Fig. 4. 4. feladat

eredményt inkrementáltuk, beszoroztuk 3800-zal, majd hozzákötöttük egy időmérőhöz, mely 3800 ms és 7600 ms közötti ideig késlelteti a program futtatását. A második ablakba tettük be a LED-et, majd egy true konstanst csatlakoztattunk rá. Ezek mellé elhelyeztünk egy időmérőt, amely méri a LED felvillanásának idejét. A harmadik ablakba került a kapcsoló, melyet egy while ciklusba raktunk. Ez a ciklus addig fut míg a kapcsoló kikapcsolt állapotban van, vagyis míg igaz nem lesz a logikai értéke. Végül, a negyedik ablakban egy újabb időmérőt helyeztünk el, mely leméri a program lefutásának idejét. A két időmérés eredményét kivontuk egymásból, majd a kapott számot hozzákapcsoltuk a reakcióidőt kijelző numerikus indikátorhoz.



Fig. 5. 5. feladat

VI. HATODIK FELADAT

Az utolsó feladat során egy szimuláció segítségével meghatároztuk a feszültségforrás jelének effektív értékét. (sinus, négyszög, háromszög és fűrész alakú) Ezt összehasonlítottuk az elméleti úton kiszámított értékekkel. Első lépésként elhelyeztünk 4 db numerikus kontrollert a Front Panel-en, a különböző jelalakoknak megfelelően. Ezekhez egymással párhuzamosan kétféle számolást kötöttünk. Az első a szimuláció alapján méri le a feszültség effektív értékét, míg a második ág a tanult képleteknek megfelelően számolja ki. A felhasznált képleteket az alábbi táblázat tartalmazza:

Szinusz	$U_{max}/\sqrt{2}$
Négyszög	U_{max}
Háromszög	$U_{max}/\sqrt{3}$
Fűrész	$U_{max}/\sqrt{3}$

A kapott eredmények eltérését egy numerikus indikátor segítségével jelenítettük meg a Front Panel-en.



Fig. 6. 6. feladat