

.NET Gadgeteer - Bevezetés

Dr. Molnár András

2014 tavasz

Áttekintés

1. Mi a .NET Gadgeteer?
2. FEZ Spider Starter Kit
3. A demópanel használata
4. Az első Gadgeteer alkalmazás elkészítése
5. Képek a programról
6. Felkészülést segítő kérdések

1. Mi a .NET Gadgeteer? (1)

A Microsoft .NET Gadgeteer egy nyílt forrású eszközkészlet, mellyel kisméretű elektronikus eszközökön futtatható, .NET Micro Framework alapú alkalmazások készíthetők Visual Studio használatával.

1. Mi a .NET Gadgeteer? (2)

Milyen eszközök szükségesek a .NET Gadgeteer-hez?

Hardver

- Demópanel (pl.: Fez Spider, Fez Hydra, Sytech NANO)
- „piros modul” (USBClientDP), mely táplálja a próbapanelt és lehetővé teszi a program feltöltést, illetve hibakeresést
- Próbamodulok teszteléshez (pl.: különféle szenzorok, megjelenítők, tárolók, hálózati eszközök, stb.)

A **FEZ Spider Starter Kit** ezeket mind tartalmazza.

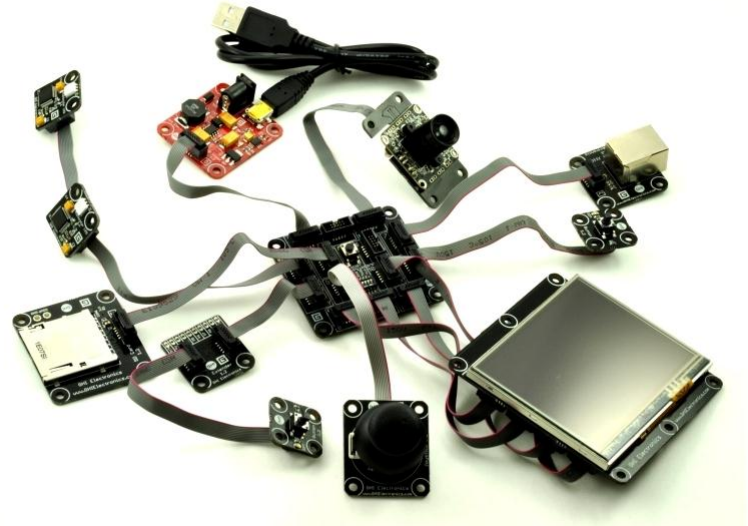
Szoftver

- Microsoft Visual Studio 2010 (vagy az ingyenes Visual C# 2010 Express)
- .NET Micro Framework SDK
- a demópanelhez szükséges illesztőprogramok és egyéb szoftverek (gyártónként eltérő)

2. FEZ Spider Starter Kit (1)

A FEZ Spider Starter Kit tartalma

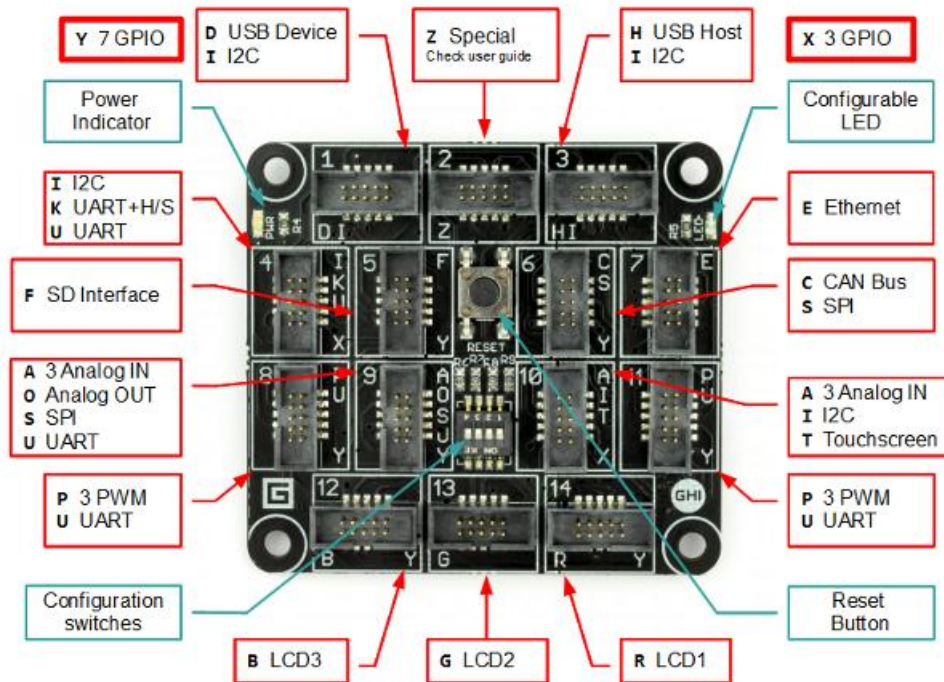
- FEZ Spider demópanel
- T35 kijelző modul (3.5" méret, érintőképernyő)
- USB Client DP modul („piros modul”)
- Kamera modul
- 2 db Multicolor LED modul
- 2 db nyomógomb modul
- J11D Ethernet modul
- SD kártya kezelő modul
- USB Host modul
- Bővítő modul
- Josytick modul
- Szalagkábelek



2. FEZ Spider Starter Kit (2)

FEZ Spider Demópanel

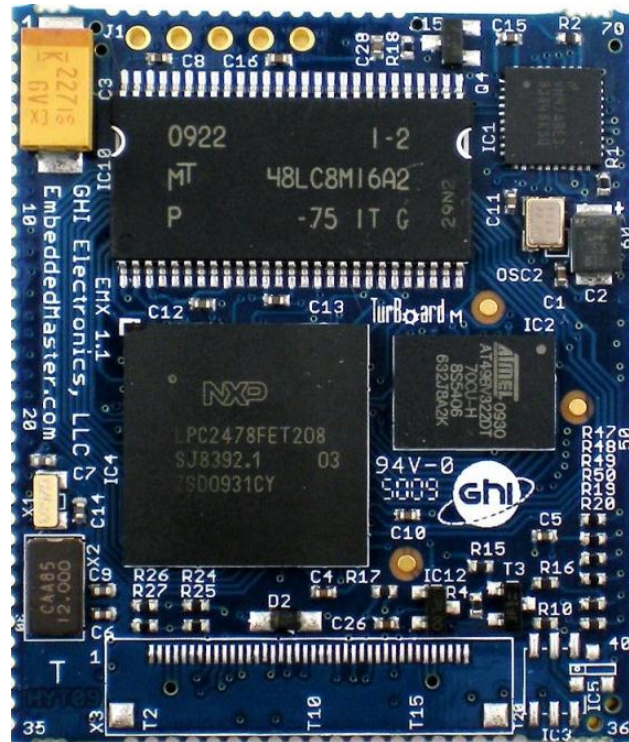
- 14 db .NET Gadgeteer kompatibilis csatlakozó aljzat a következő jelölésekkel: X, Y, A, C, D, E, F, H, I, K, O, P, S, T, U, R, G, B, Z.
- Konfigurálható LED-ek
- Konfiguráló kapcsolók
- **EMX modul**



2. FEZ Spider Starter Kit (3)

EMX modul (1)

- Az EMX modul hardver elemek összességét tartalmazza (ARM processzor, flash, RAM ethernet, stb.) egy SMT panelen, amely a .NET Micro Framework működtetését teszi lehetővé
- Tartalmazza a GHI Electronics kiegészítő kódkönyvtárait



2. FEZ Spider Starter Kit (4)

EMX modul (2)

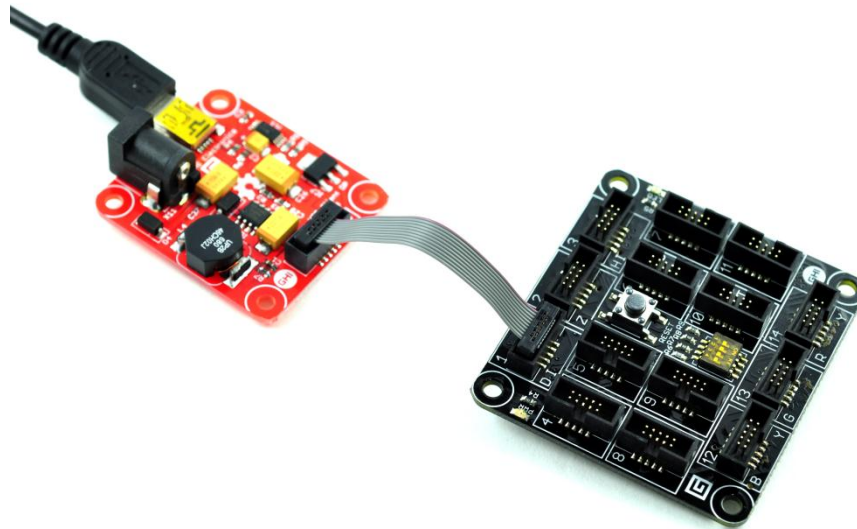
Specifikáció

- 72MHz 32-bit ARM7 processzor
- 4.5 MB Flash memória
- 16 MB RAM
- LCD vezérlő
- TCP/IP Stack SSL, HTTP, TCP, UDP, DHCP támogatással
- Ethernet, WiFi meghajtó és PPP (GPRS/ 3G modemekhez), illetve DPWS
- USB hoszt
- USB eszköz specializált könyvtárakkal különféle eszközök emulálásához, mint pl.: thumb-drive, virtual COM (CDC), egér, billentyűzet
- 76 GPIO Pin
- 2 SPI (8/16bit)
- I2C
- 4 UART
- 2 CAN busz
- 7 10-bit Analóg Bemenet
- 10-bit Analóg Kimenet (WAV audio visszajátszáshoz)
- 4-bit SD/MMC memóriakártya interfész
- 6 PWM
- Egyvezetékes interfész
- Beépített Valós Idejű Óra (RTC)
- Processzor regiszter elérés
- Kimeneti komparátor hanghullámok létrehozásához
- RLP natív kód támogatás (C/Assembly) valós idejű programozáshoz
- Matematikai osztály kétszeres pontosságú számításokhoz
- FAT Fájlrendszer
- Kriptográfia támogatás (AES és XTEA)
- alacsony fogyasztási és hibernált mód támogatása
- frissítés SD kártyáról, hálózatról, vagy más forrásból

3. A demópanel használata (1)

A demópanel működése

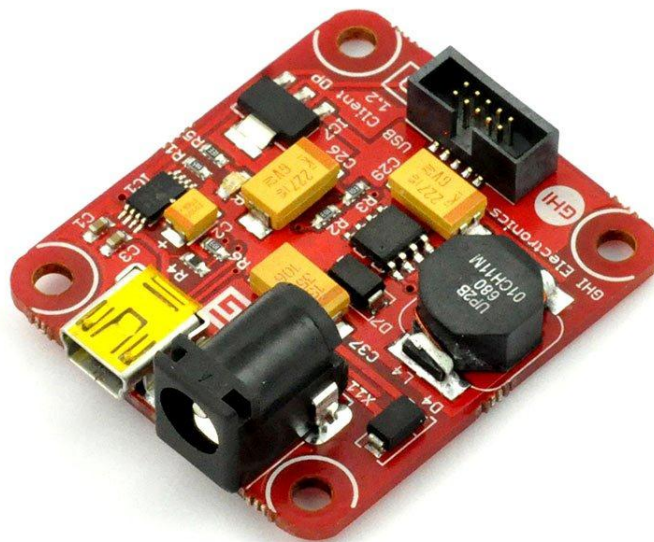
- A FEZ Spider demópanel 14 db aljzattal rendelkezik a próbamodulok illesztéséhez.
- A demópanelen elhelyezett „**Reset**” gomb segítségével bármikor újraindítható az eszköz.
- Az USB Client Dual Power modul, más néven „piros modul” táplálja árammal a demópanelt és a hozzákapcsolt eszközöket. Szintén ezzel az eszközzel csatlakoztatható PC-hez a panel (USB interfészen keresztül) program feltöltés, illetve hibakeresés céljából.



3. A demópanel használata (2)

„Piros modul”

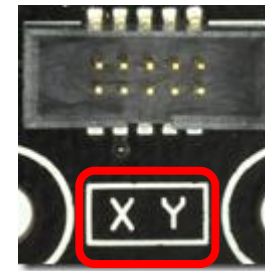
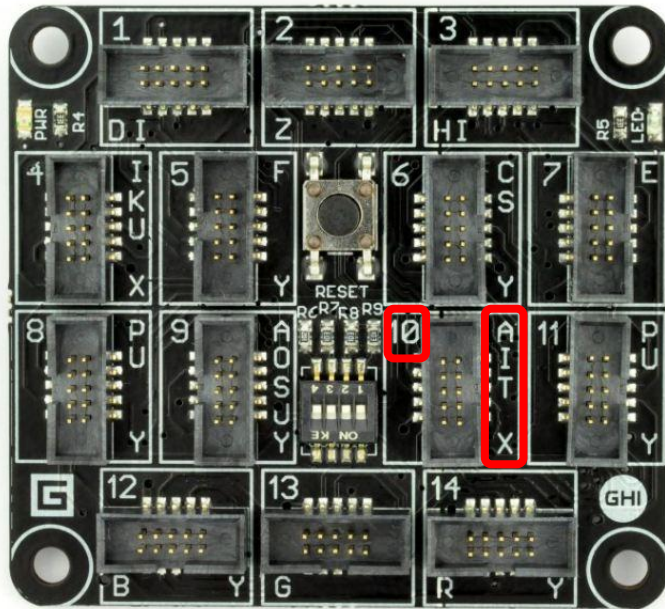
A „piros modul” egyetlen aljzattal rendelkezik, melynek ún. „D” jelölése van, ezért a demópanelen kizárólag a vele egyező „D” jelölésű aljzattal párosítható!



3. A demópanel használata (4)

Jelölések a demópanelen és a modulokon (1)

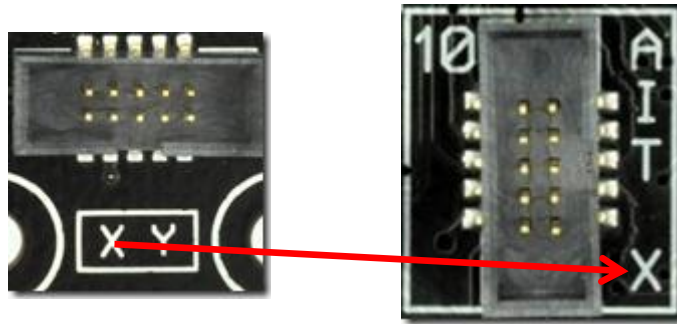
- A demópanelen és a modulok hátoldalán található aljzatokat különféle jelölésekkel látták el az egyszerű kezelhetőség érdekében:
 - azonosító
 - karakterek sorozata az elektronikus és kommunikációs csatlakoztathatóság meghatározásához



3. A demópanel használata (5)

Jelölések a demópanelen és a modulokon (2)

- Az ábrán látható XY címkéjű modul például olyan aljzathoz illeszthető, melyen megtalálható az X vagy az Y jelölés (tehát elég az egyik karakternek egyeznie)



- A csatlakoztatás során mindig ügyelni kell arra, hogy a megfelelő címkével rendelkező csatlakozó felületeket párosítsuk!

4. Az első Gadgeteer alkalmazás elkészítése (1)

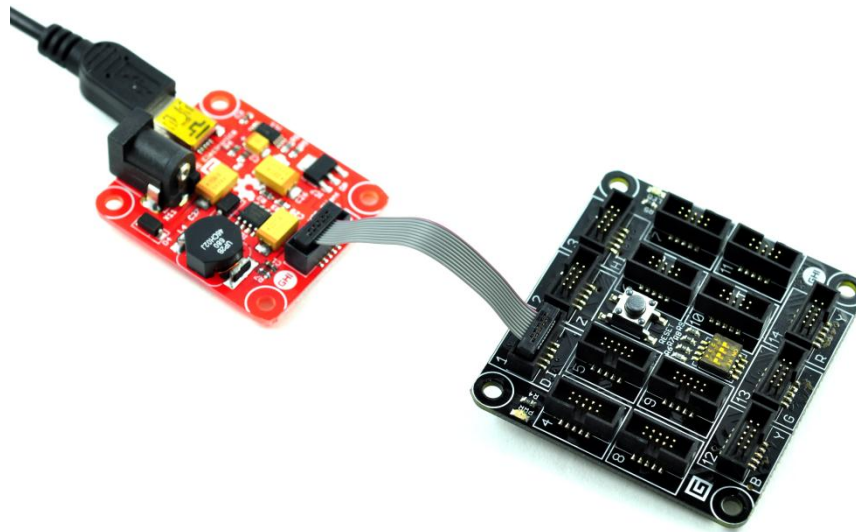
A feladat

Készítsünk egy olyan Gadgeteer alkalmazást, mely egy nyomógomb megnyomását követően folyamatosan villogtat egy LED-et!

4. Az első Gadgeteer alkalmazás elkészítése (2)

Az eszköz előkészítése

- A „piros modul” a FEZ Spider demópanel „1-es” csatlakozójához kell kapcsolni (az egyetlen aljzat, mely „D” jelzéssel rendelkezik, kizárólag ez használható fel!).
- Ezt követően lehet a demópanelhez kapcsolni a különböző próbamodulokat (ügyeljünk a jelölésekre!)



Figyelem!!!
A próbamodulok csak a kikapcsolt demópanelhez csatlakoztathatók, vagy távolíthatók el abból!

4. Az első Gadgeteer alkalmazás elkészítése (3)

Próbamodulok csatlakoztatása (1)

Az alkalmazáshoz a következő próbamodulok szükségesek:

- Multicolor LED



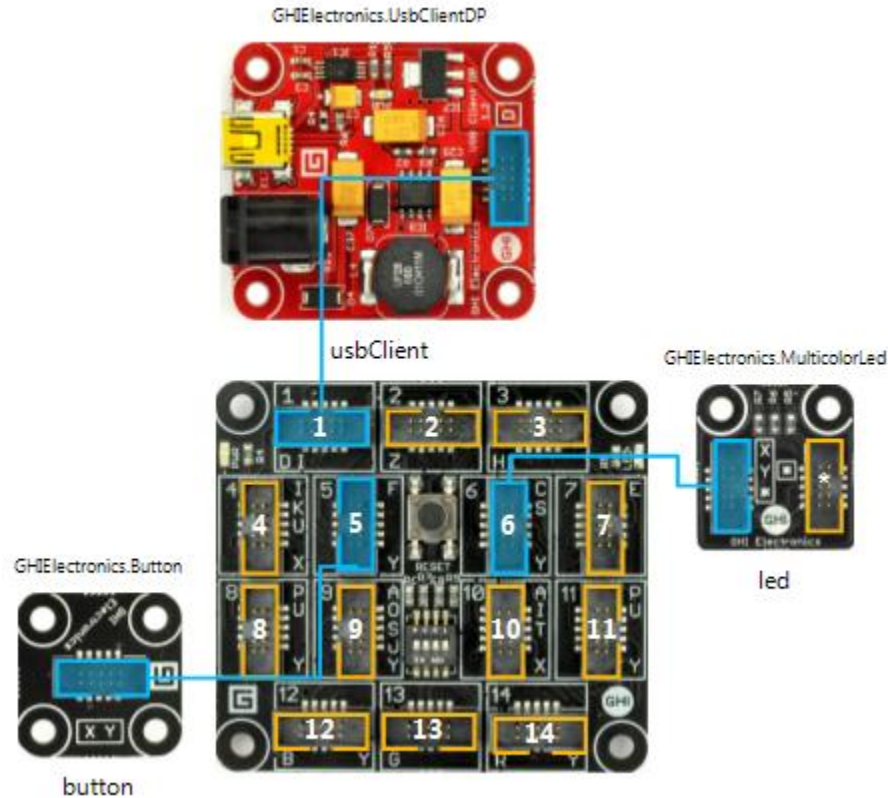
- Button



4. Az első Gadgeteer alkalmazás elkészítése (4)

Próbamodulok csatlakoztatása (2)

A próbamodulokat az ábrán látható módon csatlakoztassuk:



4. Az első Gadgeteer alkalmazás elkészítése (5)

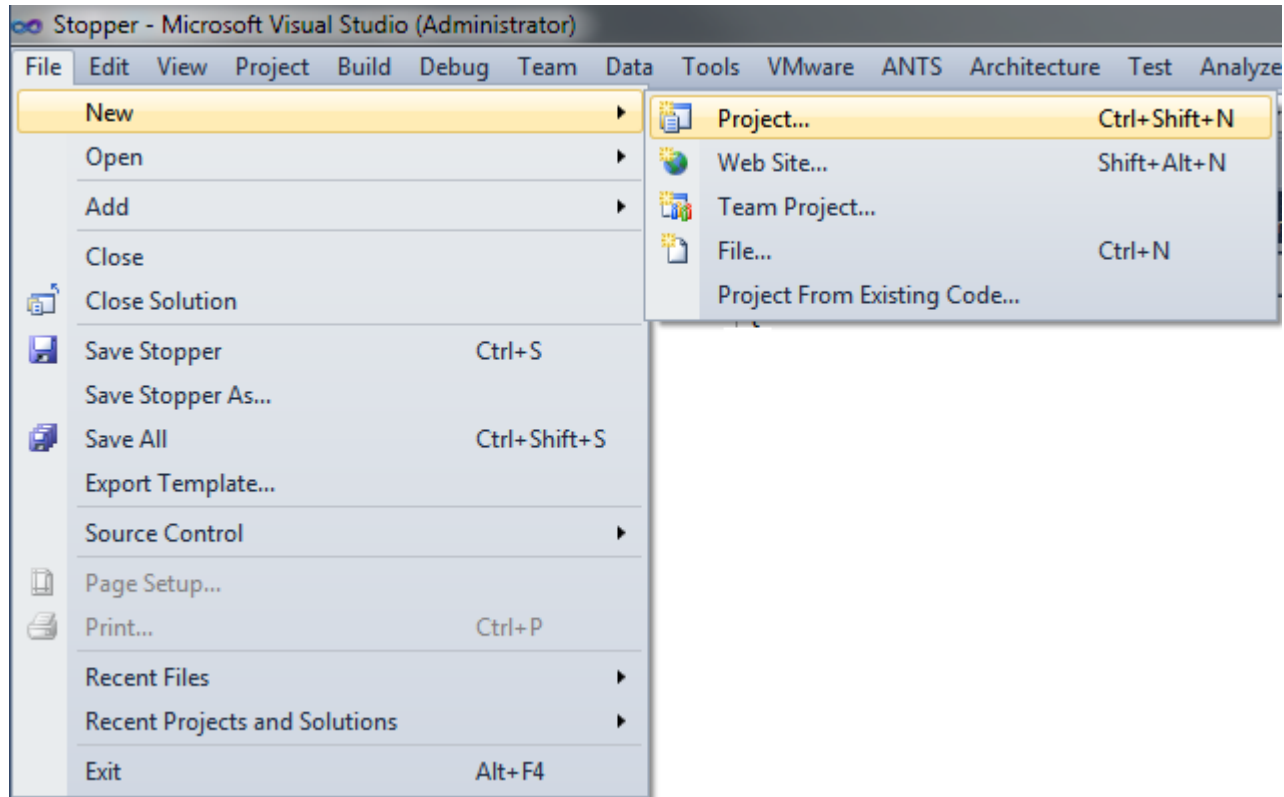
Csatlakozás számítógéphez

- A „Starter Kit”-hez mellékelt USB kábel egyik végét csatlakoztassuk a piros modulhoz, a másikat a PC-hez
- Ezután az eszköz működésbe lép, boot-ol, és elindul a rajta lévő program
- A fejlesztő által készített alkalmazás feltölthető az eszközre, és azon hibakeresést lehet végezni

4. Az első Gadgeteer alkalmazás elkészítése (6)

Nyissuk meg a Visual Studio 2010 fejlesztő környezetet.
File → New → Project...

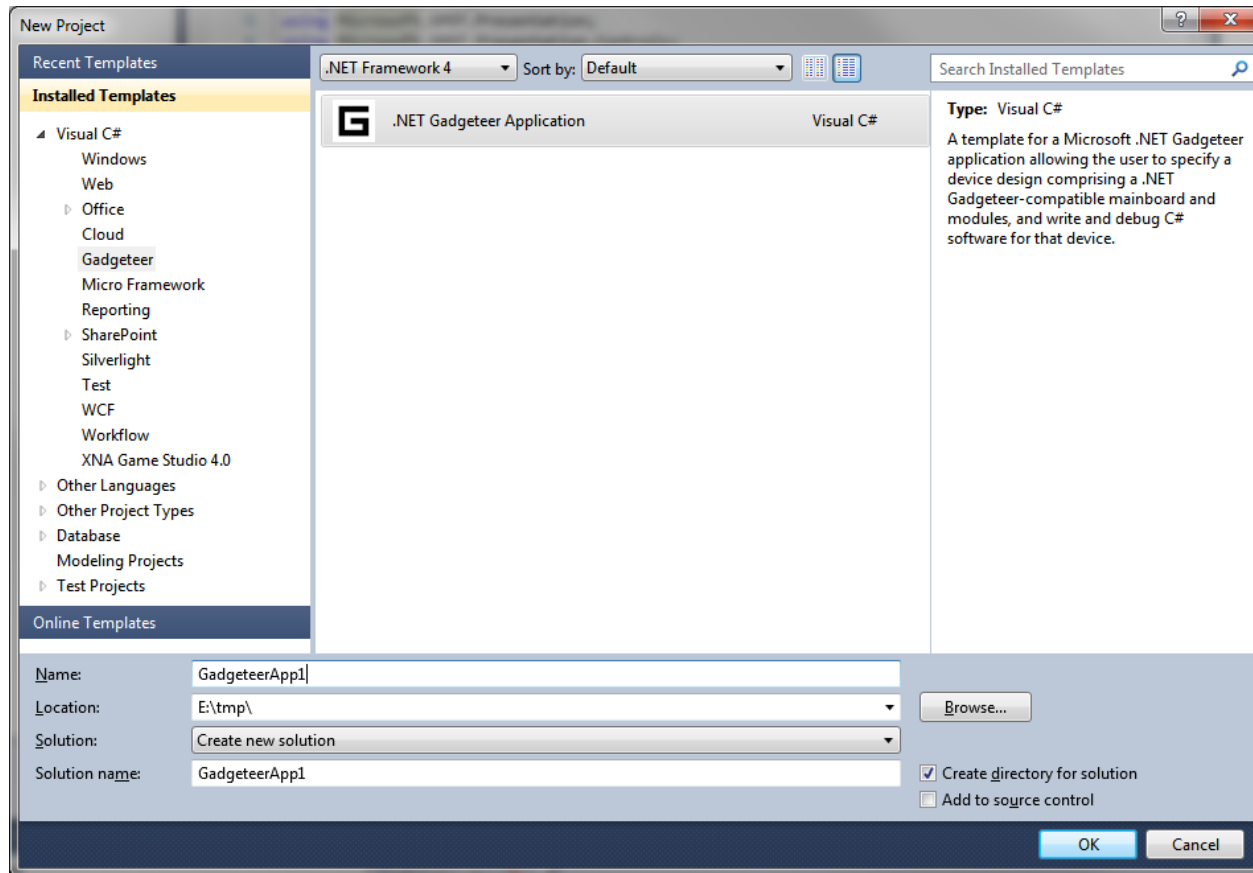
Új projekt létrehozása (1)



4. Az első Gadgeteer alkalmazás elkészítése (7)

A felugró ablakban: Visual C#/Gadgeteer/.NET Gadgeteer Application. Nevezzük el a projektet és adjunk meg egy elérési utat.

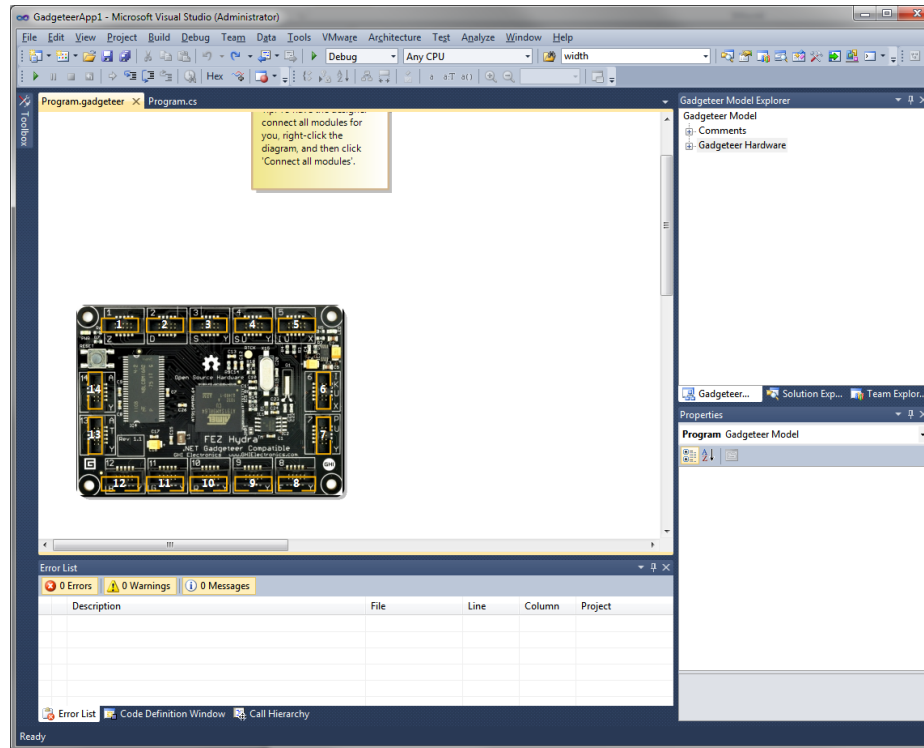
Új projekt létrehozása (2)



4. Az első Gadgeteer alkalmazás elkészítése (8)

Előfordulhat, hogy az újonnan létrehozott projekt alapesetben egy FEZ Hydra nevű demópanelt tartalmaz.

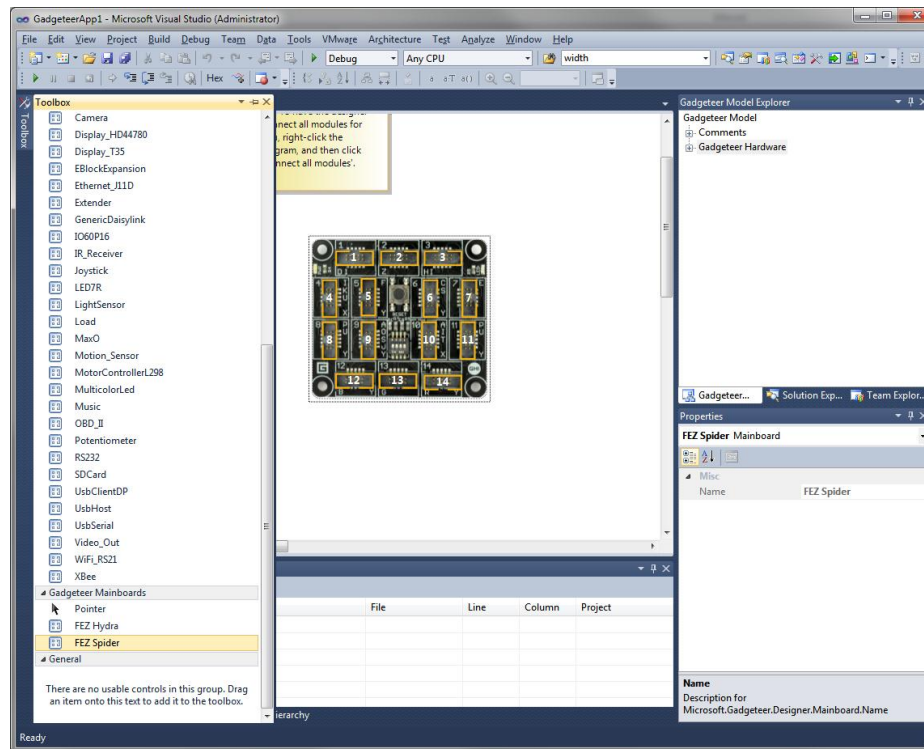
Demópanel kiválasztása (1)



4. Az első Gadgeteer alkalmazás elkészítése (9)

Mivel nekünk FEZ Spider típusú a demópanelünk, ezért csere szükséges. Jelöljük ki a FEZ Hydra-t, majd nyomjuk meg a Del billentyűt. Ezután az eszközkészletből (Toolbox) húzzunk be egy FEZ Spider típusú panelt a grafikus tervezőbe.

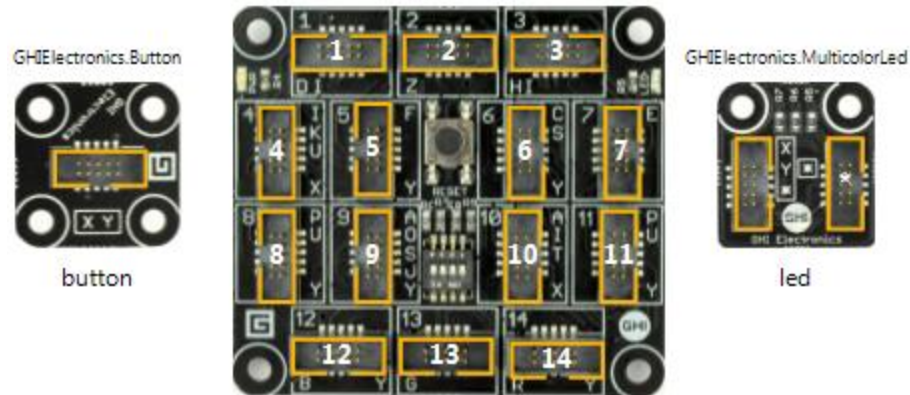
Demópanel kiválasztása (2)



4. Az első Gadgeteer alkalmazás elkészítése (10)

Konfiguráció összeállítása (1)

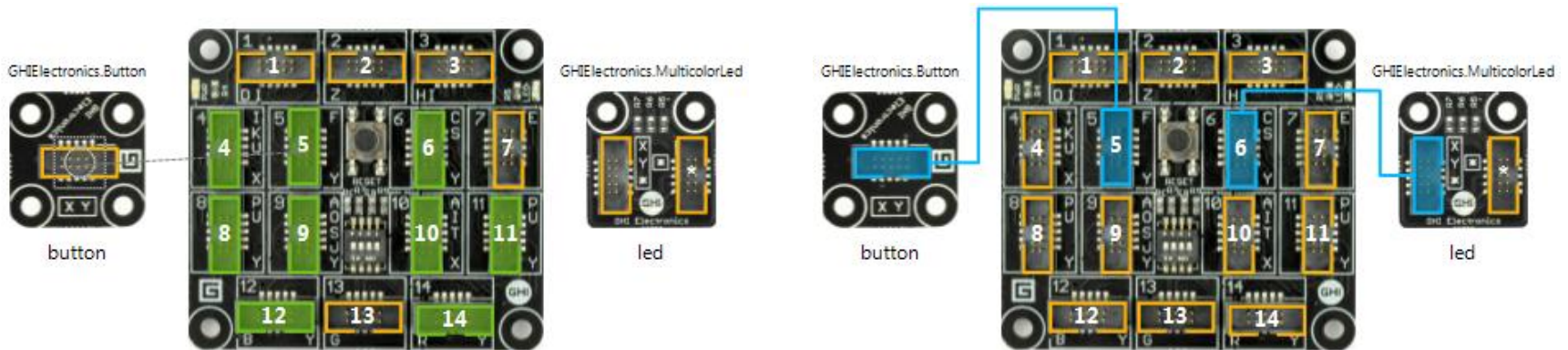
- A fordítónak is ismernie kell az összeállított demópanel konfigurációt, ezért a grafikus tervezőben meg kell jelölni, hogy az egyes aljzatokba milyen eszközök kerültek bekötésre
- Drag and drop technikával húzzunk a grafikus tervezőbe egy Button és egy Multicolor LED komponenst a Toolbox-ból



4. Az első Gadgeteer alkalmazás elkészítése(11)

Konfiguráció összeállítása (2)

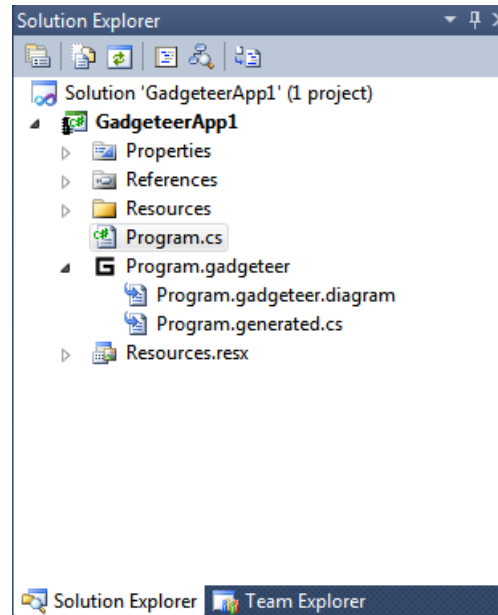
- Az egyes alkatrészek szabadon mozgathatók a tervezőben az átláthatóság javításának céljából (fizikai szerepe nincs)
- Az alkatrészek aljzatokba történő bekötéséhez először jelöljük ki kattintással a bekötni kívánt Button alkatrész csatlakozóját (sárga rész)
- Ezután a program által felajánlott kompatibilis aljzatok közül (zölden jelölt) a bal-egérgomb megnyomásával választjuk ki az 5-ös aljzatot (a demópanelen ugyanis ide kötöttük be a nyomógombot)
- A sikeres bekötést kék színű vonal jelöli
- Ezt követően a hardver konfiguráció a fordító számára is ismert lesz és elkezdhető a programkód írása



4. Az első Gadgeteer alkalmazás elkészítése (12)

A program megírása (1)

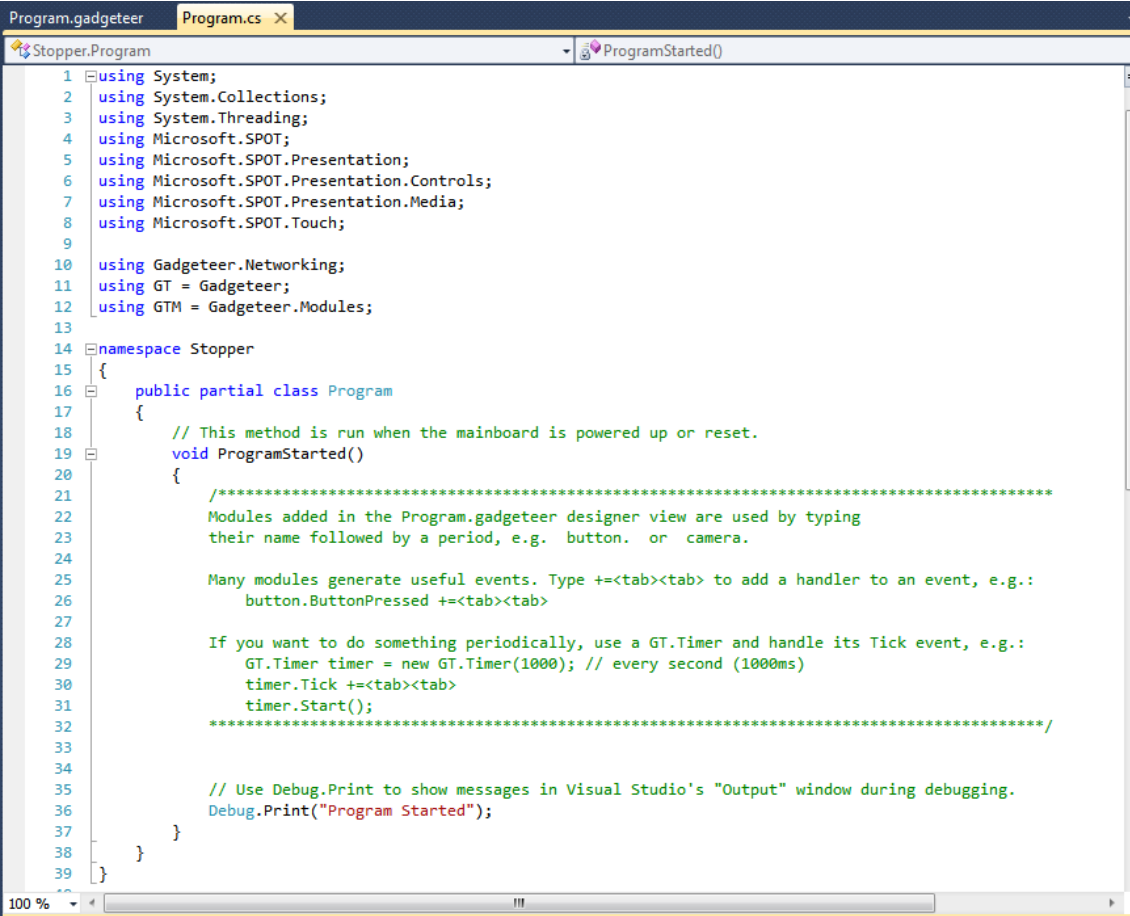
Kattintsunk duplán a Solution Explorer ablakban található Program.cs elemre!



4. Az első Gadgeteer alkalmazás elkészítése (13)

A program megírása (2)

A kiválasztott sablon alapján egy Program.cs nevű kódfájl jön létre egy Program nevű osztállyal, melynek ProgramStarted() metódusa elsőként fut le az alkalmazás indításakor.



```
1 using System;
2 using System.Collections;
3 using System.Threading;
4 using Microsoft.SPOT;
5 using Microsoft.SPOT.Presentation;
6 using Microsoft.SPOT.Presentation.Controls;
7 using Microsoft.SPOT.Presentation.Media;
8 using Microsoft.SPOT.Touch;
9
10 using Gadgeteer.Networking;
11 using GT = Gadgeteer;
12 using GTM = Gadgeteer.Modules;
13
14 namespace Stopper
15 {
16     public partial class Program
17     {
18         // This method is run when the mainboard is powered up or reset.
19         void ProgramStarted()
20         {
21             /*****
22             Modules added in the Program.gadgeteer designer view are used by typing
23             their name followed by a period, e.g. button. or camera.
24
25             Many modules generate useful events. Type +=<tab><tab> to add a handler to an event, e.g.:
26             button.ButtonPressed +=<tab><tab>
27
28             If you want to do something periodically, use a GT.Timer and handle its Tick event, e.g.:
29             GT.Timer timer = new GT.Timer(1000); // every second (1000ms)
30             timer.Tick +=<tab><tab>
31             timer.Start();
32             *****/
33
34             // Use Debug.Print to show messages in Visual Studio's "Output" window during debugging.
35             Debug.Print("Program Started");
36         }
37     }
38 }
39
```

4. Az első Gadgeteer alkalmazás elkészítése (14)

A program megírása (3)

Az ProgramStarted() metódust egészítsük ki az alábbi módon:

```
void ProgramStarted()  
{  
    button.ButtonPressed += new  
        Button.ButtonEventHandler(button_ButtonPressed);  
    Debug.Print("Program Started");  
}
```

- A gombnyomás kezeléséhez feliratkozunk a button (nyomógomb) objektum ButtonPressed eseményére egy button_ButtonPressed metódust paraméterként átadva

4. Az első Gadgeteer alkalmazás elkészítése (15)

A program megírása (4)

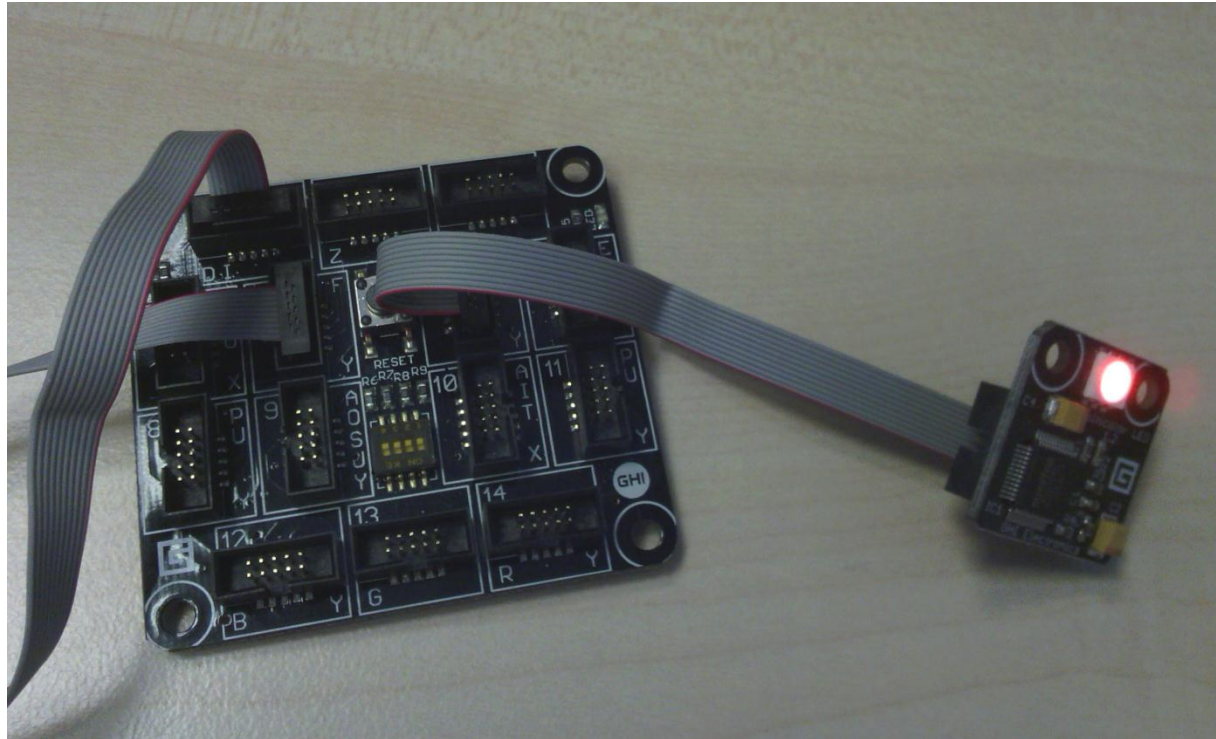
Következő lépésként a `button_ButtonPressed` metódust implementáljuk:

```
void button_ButtonPressed(Button sender, Button.ButtonState state)
{
    led.BlinkRepeatedly(GT.Color.Red);
}
```

- Amennyiben a felhasználó megnyomja a gombot, ez a metódus kerül végrehajtásra, a `sender` paraméter tartalmazza a gombhoz tartozó referenciát, a `state` pedig a gomb állapotát
- A gombnyomás bekövetkezésekor a LED-et folyamatosan villogó állapotba helyezzük piros színnel (`GT.Color.Red`)

5. Képek a programról

A program működés közben



6. Felkészülést segítő kérdések

- Milyen eszközök szükségesek .NET Gadgeteer alkalmazások készítéséhez?
- Mi a szerepe a demópanelen, illetve a próbamodulokon elhelyezett jelöléseknek (címkék)?
- Milyen lépéseket kell elvégezni egy új Gadgeteer projekt létrehozásához?
- Mire való a ProgramStarted() metódus?