Diy-efis 2021 Code Preview

Presented by Kotuku Aerospace Limited

Overview of the project

- Has been in the GitHub diy-efis repository since 2015
- Original code base was written in C and was designed to be ported to own hardware. A port of the display to run on Raspberry Pi was provided.
- Kotuku has changed the architecture of diy-efis. Still uses the same API libraries which abstract Linux on a Raspberry Pi or Windows as an emulator. Kotuku products are all embedded processors.
- Still had the issue that the user needs to re-deploy the code base to upload changes.
- Solution is to create an ‘app’ that is written in C# or any Ecma334 language (with some restrictions).

Application design

- Kotuku initially ported the nanoFramework to CanFly but soon found there are issues with this. Decision was to write a completely new interpreter.
- All CanFly widgets that were written in C are not ported to C# and released.
- The CoreLibrary comes from .NET/nanoFramework but is designed to have a stable API so applications can be developed to work on any system.
- Current status is that the libraries are built with the nanoFramework using Visual Studio 2019 Community edition, and deployed to msh using a powershell script.
- So for a demo.....

Work being done

- Moving all of the code away from the nanoFramework. Have ported the metadata processor that converts .NET to the nanoFramework/CanFly format.
- Build will be using CMake so VSCode can be used to develop applications.
- Source level debugger will be available, for VSCode.
- Kotuku will release binary images of the MSH interpreter/runtime and msh on linux for ia36 and Raspberry Pi.
- Eventually the msh code base will be available, once some copyright issues are resolved.