X-Plane-Plugin

Systembeschreibung

Das Plugin unterstützt nur die Windows-Version von X-Plane.

Durch das X-Plane Plugin SDK2 und X-Plane SDK3 wird Zugriff auf Kommandos und Daten-Referenzen zur Verfügung gestellt. Durch Plugins ist es möglich, mit externer Hardware Kommandos an den Flugsimulator X-Plane zu senden und Variablen zu setzen.

Das X-Plane Plugin kommuniziert direkt mit der Arduino-Hardware über das Netzwerkprotokoll UDP.

System-Voraussetzungen

Für die Verwendung der X-Plane Plugins ist eine X-Plane-Version höher 10.0 notwendig. Voraussetzung ist ein konfiguriertes Netzwerk mit automatischer Adressvergabe (DHCP), damit dem Arduino eine gültige IP-Adresse zugewiesen wird. Dies erfolgt z.B. über den Router, der die Internetverbindung realisiert.

Speicherort der X-Plane Plugins:
Die Plugins `*.xpl` müssen manuell aus dem Installationsordner in den Plugin-Ordner des X-Plane kopiert werden, dieses Verzeichnis `\Ressources\plugins` liegt unterhalb des X-Plane-Installationsorders.

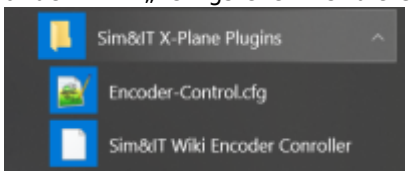
Der Speicherort der **Konfigurations-Dateien** der X-Plane Plugins ist:
Win7 ff: `C:\Users\Benutzername\AppData\Roaming\Sim&IT\X-Plane-Plugins`

Diese Ordner sind standardmäßig versteckt.

Die Konfigurationsdatei kann z.B. mit dem Editor (Programme → Zubehör) angezeigt und bearbeitet werden. Ein entsprechender Link zur Konfigurationsdatei wird bei der Installation im Programmordner erzeugt.

Software-Installation

Nach dem Start des Setup können Sie den Zielordner der Installation und den Menü-Ordner der Programm-Gruppe ändern. Mit „Fertigstellen“ ist die Client-Installation abgeschlossen.



Nach der Installation finden Sie in der bei der Installation vorzugsweise angegebenen Programmgruppe `Sim&IT X-Plane-Plugins` die Links zum Installationsordner der Plugins und zur Konfigurationsdatei für das Ändern der Konfigurationsvariablen der Plugins.

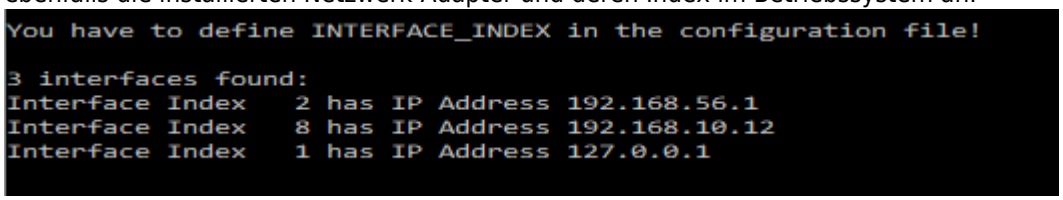
Plugin-Konfiguration

Nach dem Starten des X-Plane finden Sie im Menü **Plugins** jeweils einen Menüpunkt mit zwei Unterpunkten für Infos und für die Eingabe von Lizenzinformationen. Das Pugin funktioniert ohne Lizenz für ca. 10 Minuten.

Im Info-Bereich wird entweder die Verbindung zum Arduino-Panel und die Lizenzinformation angezeigt.

Wenn **mehr als ein Netzwerkadapter** gefunden wurde, dann werden diese mit dem Index angezeigt, der vom Betriebssystem vergeben wurde. Der Index des Netzwerkadapters, an dem die Arduino-Hardware angeschlossen ist, muss in der Konfigurationsdatei
`.\AppData\Roaming\Sim&IT\X-Plane-Plugins\Encoder-Control.cfg`
mit der Konfigurationsvariablen `INTERFACE_INDEX` definiert werden, um eine Verbindung aufzubauen.

Zum Feststellen des Index kann auch das mitinstallierte Programm `Show NetIndex` genutzt werden, dieses zeigt ebenfalls die installierten Netzwerk-Adapter und deren Index im Betriebssystem an.



Mit Windowsmitteln kann dies wie folgt ermittelt werden:
`wmic path win32_networkadapter get name, index, interfaceindex`

Konfigurationsvariablen

Den Konfigurationsvariablen folgt ein = und dann der zu setztende Wert.

Für alle Module gibt es den Abschnitt `[NET]` mit der Konfigurationsvariablen

- `INTERFACE_INDEX` zur Definition des Netzwerkadapters bei mehreren im System

Konfiguration des Encoder Controllers

Für den Encoder Controller gibt es den Abschnitt `[ENC1]`

- ARDU_PORT zur Definition des Ports des empangenden Gerätes, wenn Standardport nicht nutzbar ist
- RECV_PORT zur Definition des Ports auf dem PC, wenn Standardport nicht nutzbar ist

Die geänderten Ports müssen ebenfalls im Arduino-Sketch definiert sein.

Für jeden Encoder (1 bis 6) gibt es Konfigurationseinstellungen, um andere als die vorgegebenen Variablen bzw. Kommandos entsprechend der Dateien `Commands.txt` bzw. `DataRefs.txt` zu definieren (exemplarisch für Encoder1):

- ROTARY1_CMD_OR_VALUE 0 bedeutet, dass Kommandos definiert werden, 1 bedeutet, dass eine zu definierende Variable erhöht oder verringert wird.
- ROTARY1_VALUE definiert eine zu verändernde Variable, Voraussetzung ist die Definition von ROTARY1_CMD_OR_VALUE=1
- ROTARY1_INVERSE ändert die Drehrichtung bei der Variablenänderung
- ROTARY1_CMD_RIGHT Kommando für Drehrichtung im Uhrzeigersinn
- ROTARY1_CMD_LEFT Kommando für Drehrichtung entgegen Uhrzeigersinn
- ROTARY1_CMD_PB Kommando beim Drücken auf die Drehgeberachse (Push Button)
- ROTARY1_CMD_PB_HOLD Kommando bei längerem Drücken auf die Drehgeberachse

Für den Encoder6 gibt es noch eine zusätzliche Funktion. Wenn die Option

- ROTARY6_PUSH_TO_ACTIVATE

auf 1 gesetzt ist, dann wird der Encoder erst nach dem einmaligen Drücken aktiviert.

Es gibt noch zwei Einzeltasten 7 und 8 mit den Konfigurationsvariablen (exemplarisch für Taste7).

- KEY7_CMD Kommando beim Drücken der Taste
- KEY7_CMD_HOLD Kommando bei längerem Drücken der Taste

Bei den Tests wurde festgestellt, dass das mehrfache Aufrufen eines Kommandos während eines Loop-Durchlaufes keine Auswirkungen hat, das Kommando scheint nur einmal ausgeführt zu werden, im Gegensatz zum Inkrementieren und Dekrementieren. Falls sich im Übertragungs-Puffer mehrere gleiche Drehgeber-Ereignisse befinden, funktioniert das Ändern einer Variable besser als die Kommando-Ausführung.

Konfiguration der Port Extention für Schalter

Die Konfiguration der Schalter ist bei den Erweiterungen beschrieben.

From:
http://www.simandit.de/simwiki/ - **Wiki**

Permanent link:
http://www.simandit.de/simwiki/doku.php?id=hardware:anleitungen:ardu-enc:ardu-enc_xplane

Last update: **2020/05/05 21:51**