

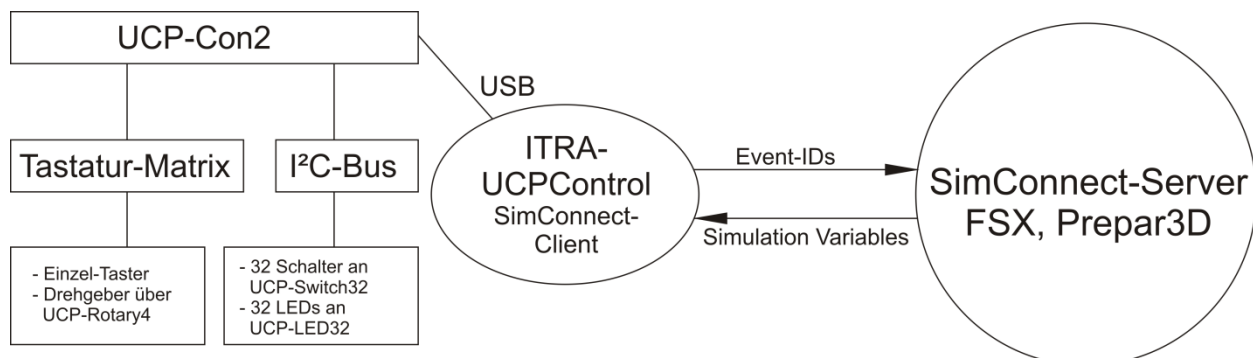
ITRA SimConnect-Client UCPCControl (für FSX und Prepar3D)



Einleitung

SimConnect ist eine von Microsoft entwickelte Schnittstelle zur Kommunikation externer Programme mit dem Flugsimulator. Der SimConnect-Server wird zusammen mit dem Flugsimulator installiert. SimConnect-Clients können unter bestimmten Voraussetzungen auch auf weiteren Netzwerk-PCs installiert werden, auf diese Möglichkeit wird hier aber nicht eingegangen, da aus Gründen der schnellen Übertragung der USB-Ereignisse z.B. von Drehimpulsgebern, eine lokale Installation der Clients bevorzugt wird.

Funktionsschema des SimConnect-Clients UCPCControl



Der SimConnect-Client kommuniziert über USB mit dem UCP-Con2. Die Ereignisse an der Tastatur-Matrix und an den Schaltern der Erweiterungsbaugruppe UCP-Switch32 werden vom USB-Controller verarbeitet und entsprechend der Definitionen in der Konfigurationsdatei werden Ereignisse (Event-IDs) zum SimConnect-Server gesendet.

Diese Ereignisse können priorisiert werden, damit ist es möglich über den SimConnect-Server auch selbst definierte Ereignisse an weitere SimConnect-Clients wie z.B. den TFT-Interface Client zu senden.

Der UCPCControl-Client kann Simulations-Variablen abfragen und entsprechend der Definition in der Konfigurationsdatei diese zur Ansteuerung von Anzeigen an den Ports der Erweiterungsbaugruppe UCP-LED32 nutzen.

System-Voraussetzungen

Für die Verwendung der ITRA-SimConnect-Client ist die Installation der aktuellen Version des FSX oder Prepar3D notwendig. Es müssen die aktuellen Treiber für die ITRA-Geräte installiert sein. Für die Verwendung der USB-Geräte installieren Sie bitte vor dem Anschluss des Gerätes an den PC das ITRA-USB-Treiberpaket.

Speicherort der ITRA-SimConnect-Clients:

C:\Program Files (x86)\ITRA\SimConnect

// Win7-64bit

C:\Program Files\ITRA\SimConnect

// Win7-32bit

Speicherort der Konfigurations-Dateien der ITRA-SimConnect-Clients

C:\Users\Benutzername\AppData\Roaming\ITRA\SimConnect

Speicherort der Konfigurations-Dateien **exe.xml** und **dll.xml** des FSX

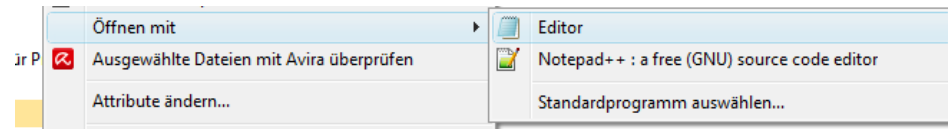
C:\Users\Benutzername\AppData\Roaming\Microsoft\FSX

Speicherort der Konfigurations-Dateien **exe.xml** und **dll.xml** des Prepar3D

C:\Users\Benutzername\AppData\Roaming\Lockheed Martin\Prepar3D

Die Konfigurationsdateien können z.B. mit dem Editor (Programme -> Zubehör) angezeigt und bearbeitet werden. Entsprechende Links zu den Konfigurationsdateien werden bei der Installation im Programmordner erzeugt.

Die Dateien öffnen Sie mit Doppelklick oder über das Kontextmenü der rechten Maustaste.

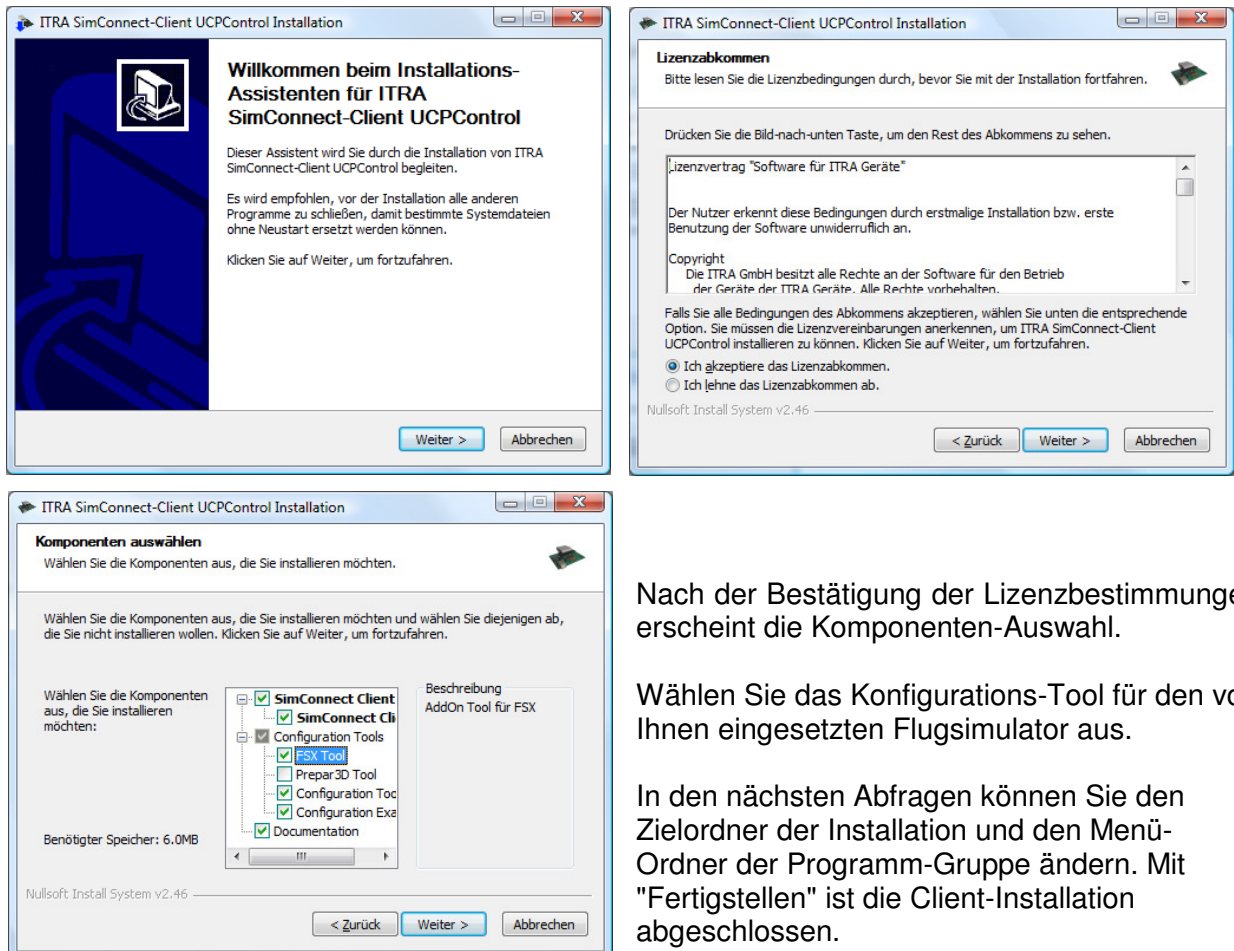


Wir empfehlen die Benutzung der mitgelieferten Konfigurations-Programme.

Software-Installation

Achtung: Installieren Sie vor dem Anschließen des USB-Gerätes an den PC unbedingt zuerst das USB-Treiber-Paket. Weitere Einzelheiten befinden sich in der Anleitung zur Treiberinstallation.

Starten Sie das Setup für die Installation der Client-Software.



Nach der Bestätigung der Lizenzbestimmungen erscheint die Komponenten-Auswahl.

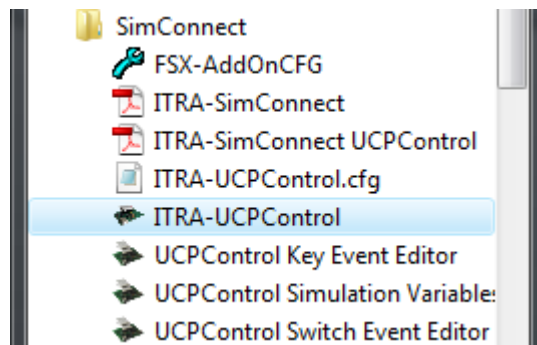
Wählen Sie das Konfigurations-Tool für den von Ihnen eingesetzten Flugsimulator aus.

In den nächsten Abfragen können Sie den Zielordner der Installation und den Menü-Ordner der Programm-Gruppe ändern. Mit "Fertigstellen" ist die Client-Installation abgeschlossen.

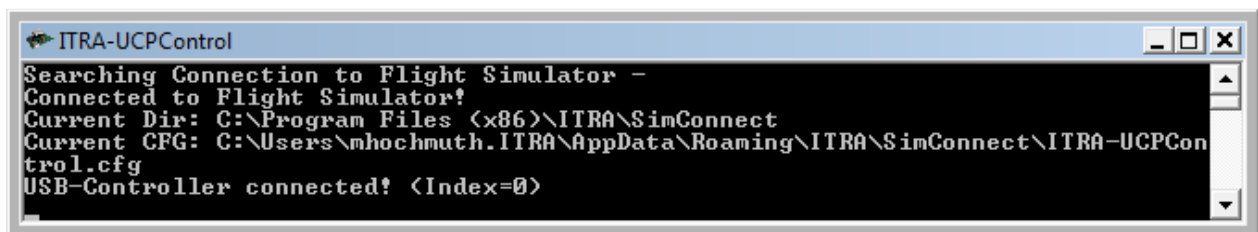
Die Beschreibungen zur Hardware entnehmen Sie bitte der Beschreibung der einzelnen UCP-Baugruppen.

SimConnect-Client-Software

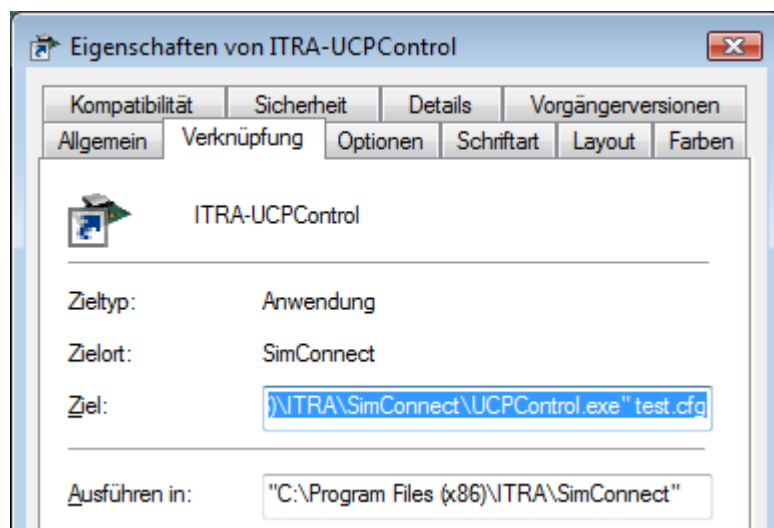
Nach der Installation finden Sie in der Programmgruppe ITRA -> SimConnect den SimConnect-Client sowie Dokumentationen, einen Link zur Konfigurationsdatei und ein Konfigurationsprogramm für den Flugsimulator.



Beim Starten des Clients wird eine Verbindung zum Flugsimulator gesucht und wenn dieser gestartet ist, aufgebaut, danach wird das Flightboard gesucht. Informationen über den Status sowie die geladene Konfigurationsdatei werden im Konsolenfenster des Clients angezeigt.



Für erfahrene Benutzer:



Jede Client-Verknüpfung kann durch Hinzufügen eines Konfigurationsdateinamens an das Ende des Verknüpfungsziels mit einer separaten Konfiguration ausgeführt werden. Die angegebene Konfigurationsdatei wird erst im aktuellen Programm-Pfad und dann im oben angegebenen Applikations-Datenpfad für die SimConnect-Clients gesucht.

Achtung! Wenn der Client über die exe.xml vom Flugsimulator gestartet wird ist der aktuelle Programm-Pfad der Flugsimulator-Pfad. Die geeignetere Ablage für Konfigurationsdateien ist der Applikations-Datenpfad der SimConnect-Clients.

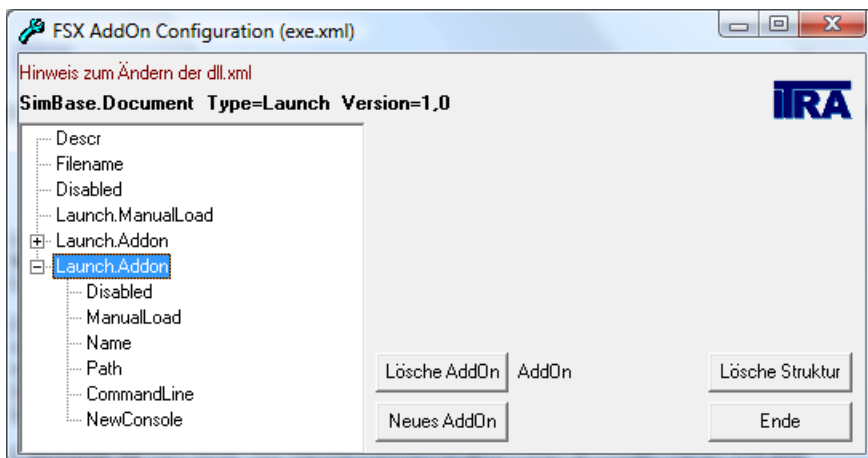
Konfigurations-Datei exe.xml für automatisches Starten der SimConnect-Clients:

```
<SimBase.Document Type="Launch" version="1,0">
  <Descr>Launch</Descr>
  <Filename>EXE.xml</Filename>
  <Disabled>False</Disabled>
  <Launch.ManualLoad>False</Launch.ManualLoad>
```

...
// für jeden Client solch einen Abschnitt erzeugen

```
<Launch.Addon>
  <Disabled>False</Disabled>
  <ManualLoad>False</ManualLoad>
  <Name>UCPCControl</Name> // hier den Namen anpassen
  <Path>C:\Program Files (x86)\ITRA\SimConnect\UCPCControl.exe</Path> // Win7-64bit
  <Path>C:\Program Files\ITRA\SimConnect\UCPCControl.exe</Path> // Win7-32bit
  <Path>C:\Programme\ITRA\SimConnect\UCPCControl.exe</Path> // XP
  <CommandLine></CommandLine>
  <NewConsole>True</NewConsole>
</Launch.Addon>
...
</SimBase.Document>
```

Für eine einfache Modifizierung dieser Datei dienen die Programme **FSX-AddOnCFG** / **Prepar3D-AddOnCFG**. Die Datei exe.xml wird im Applikations-Datenpfad des Flugsimulators gesucht, die Struktur wird ermittelt und angezeigt. Sie können AddOns hinzufügen und löschen und die Einträge ändern. Ist die Datei nicht vorhanden oder leer, dann wird eine neue Baumstruktur erzeugt.



Lösche Baum erzeugt eine leere Struktur entsprechend der Definition ohne AddOns.

Neues AddOn erzeugt einen neuen Eintrag, **Disable** ist auf True gesetzt.

Lösche AddOn löscht den markierten Eintrag, dessen Name wird neben der Taste angezeigt.

Dem Eintrag **Path** ist ein Eingabefeld zugeordnet, um den zu startenden Client zu definieren, mit dem Knopf daneben kann im Verzeichnisbaum nach dem SimConnect-Client gesucht werden. Bei dieser Aktion wird nach der Auswahl der Datei das AddOn aktiviert (**Disable**=False) und der erzeugte **Name** wird aus dem Dateinamen abgeleitet, kann dann aber geändert werden.

CommandLine kann einen vom Standard abweichenden Konfigurationsdatei-Namen enthalten.

ManualLoad erzeugt bei jedem Start des FS eine Abfrage, ob der Client gestartet werden soll.

Bei **Ende** wird gefragt, ob Sie die neue Struktur in die Datei exe.xml speichern wollen.

Die Datei wird entsprechend der Dokumentation des SimConnect-SDKs erstellt. Im Kopf muss **Disable** auf False gesetzt sein (kein Häkchen in der Auswahlbox), um die Verarbeitung zu aktivieren, alle anderen Einträge sind entsprechend der Definition vorbelegt.

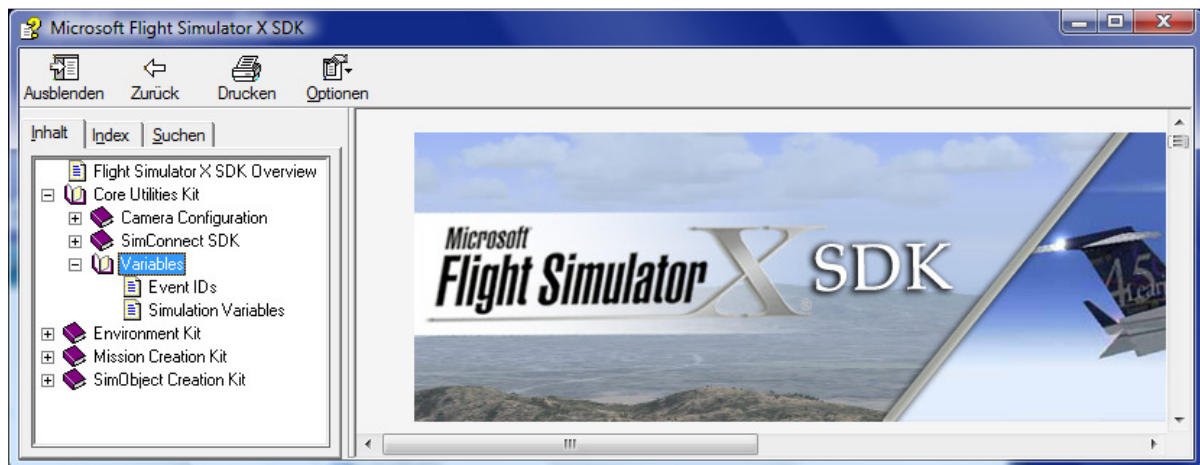
Launch.ManualLoad erzeugt für jeden Client eine Abfrage ob dieser zusammen mit dem FS gestartet werden soll.

Hinweis: Falls die SimConnect-Clients auf Ihrem System mit einer Fehlermeldung starten, dann müssen auf Ihrem System die aktuellen Laufzeitbibliotheken

Microsoft Visual C++ 2010 Redistributable Package

für die Programmumgebung installiert werden. Diese finden Sie über die Microsoft Downloads oder über einen Link auf unserer Support-Webseite

Aufbau der Konfigurationsdatei



<http://msdn.microsoft.com/en-us/library/cc527015.aspx>

In der Dokumentation des SimConnect-SDK im Abschnitt **Event-IDs** sind die Zeichenketten (**String Name**) für die Ereignisse beschrieben, die bei der Definition der Tasten und Schalter verwendet werden können.

Event ID	String Name	Description	Multiplayer
KEY_AP_MASTER	AP_MASTER	Toggles AP on/off	Shared Cockpit
KEY_AUTOPILOT_OFF	AUTOPILOT_OFF	Turns AP off	Shared Cockpit
KEY_AUTOPILOT_ON	AUTOPILOT_ON	Turns AP on	Shared Cockpit

Die Zeichenketten der Variablen, die bei der Ansteuerung der LEDs verwendet werden, befinden sich im Abschnitt **Simulation Variables**.

Simulation Variable	Description	Units	Settable	Multiplayer
AUTOPILOT AVAILABLE	Available flag	Bool	N	-
AUTOPILOT MASTER	On/off flag	Bool	N	Shared Cockpit

Der standardmäßig verwendete Dateiname ist ITRA-UCPControl.cfg.

Für den UCP-Con2 gibt es vier Sektionen in der Konfigurationsdatei des SimConnect-Clients

```
[UCPControl]      Grundlegende Definitionen
INDEX=0          -> je nach programmiertem Index des
                  UCP-Con2 bzw. TCP-USB-S1, Standard ist 0
DEBOUNCE=0      -> Entprellung (Wert*50ms)
```

[UCP_KEYS] **Definition der Tastenzuordnungen**

Z0_S0=DME_SELECT -> String-Definition für Tast-Ereignis entsprechend der Dokumentation des SimConnect-SDK im Abschnitt Event-IDs, Beispiel für DME-Auswahl

Z0_S0_ARG=2 -> Einige Kommandos erfordern einen Parameter
Beispiel für DME2
Es sind hier nur Zahlen als Argument möglich

Z0_S0_REPEAT=0 -> {0..10} Faktor zum Definieren der Wiederholfunktion, 0=aus
1 entspricht ca. 50ms
10 entspricht ca. 500ms

Z0_S0_PRIORITY=0 -> {0,1} setzen auf hohe Event-Priorität für Senden von selbst definierten Ereignissen an weitere SimConnect-Clients
z.B. zum ITRA-Interface-Client für Steuerung der ITRA A320-Software entsprechende String-Definitionen sind dort aufgeführt.

...

Z8_S15
Z8_S15_ARG
Z8_S15_REPEAT
Z8_S15_PRIORITY -> derzeit werden 144 Taster des UCP-Con2 unterstützt von Zeile0/Spalte0 bis Zeile8/Spalte15

[UCP_SWITCHES] **Schalterdefinition bei Verwendung der Erweiterungsbaugruppe UCP-Switch32**

P0_B0_OFF -> String-Definition für Ausschalt-Ereignis des Schalters entsprechend der Dokumentation des SimConnect-SDK im Abschnitt Event-IDs

P0_B0_ON -> String-Definition für Einschalt-Ereignis des Schalters entsprechend der Dokumentation des SimConnect-SDK im Abschnitt Event-IDs

P0_B0_OFF_ARG -> Einige Kommandos erfordern einen Parameter
Es sind hier nur Zahlen als Argument möglich

P0_B0_ON_ARG -> Einige Kommandos erfordern einen Parameter
Es sind hier nur Zahlen als Argument möglich

P0_B0_PRIORITY=0 -> {0,1} setzen auf hohe Event-Priorität für Senden von selbst definierten Ereignissen an weitere SimConnect-Clients

...

P3_B7_OFF
P3_B7_ON
P3_B7_OFF_ARG
P3_B7_ON_ARG
P3_B7_PRIORITY -> Es wird eine Baugruppe UCP-Switch32 mit 32 Schaltern unterstützt

Es wird jeweils **einmalig** ein Kommando beim Zustandswechsel des Schalters ausgegeben.

[UCP_LEDS] **Definition der Ausgänge bei Verwendung einer Erweiterungsbaugruppe UCP-LED32 (I²C-Adresse 24..27)**

P0_B0=SELECTED DME -> String-Definition für Simulations-Variable entsprechend der Dokumentation des SimConnect-SDK im Abschnitt Simulation Variables

P0_B0_UNIT=number -> Einheit entsprechend der Variablen-Definition. Standard ist boolean, bei Boolean-Variablen kann diese Definition entfallen

P0_B0_LEVEL=2 -> Bei der Verwendung von Integer-Variablen kann hier eine Schaltschwelle angegeben werden, ab welcher der Ausgang geschaltet wird, RANGE beeinflusst das Verhalten

P0_B0_RANGE=1 -> Bei der Verwendung von LEVEL kann hier ein Wertebereich eingetragen werden, bei dem der Ausgang geschaltet wird. (Standard ist 0)
0: alle Werte gleich und größer LEVEL aktivieren den Ausgang
1: nur bei LEVEL wird der Ausgang aktiviert
größer 1: von LEVEL bis kleiner LEVEL+RANGE ist der Ausgang aktiv

P0_B0_INVERSE=0 -> {0,1} Die Ausgangslogik wird umgekehrt.

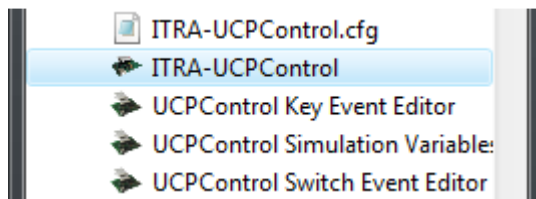
...

P3_B7
P3_B7_UNIT
P3_B7_LEVEL
P3_B7_RANGE
P3_B7_INVERSE -> Es wird eine Baugruppe UCP-LED32 mit 32 Ausgängen unterstützt

Mit dem Setup können Sie eine Beispiel-Konfiguration installieren.

Hilfsprogramme für Konfigurationsdatei

In der Programmgruppe ITRA – SimConnect befinden sich nach der Installation drei Hilfsprogramme für das Erstellen und Bearbeiten der UCPCControl-Konfigurationsdatei.

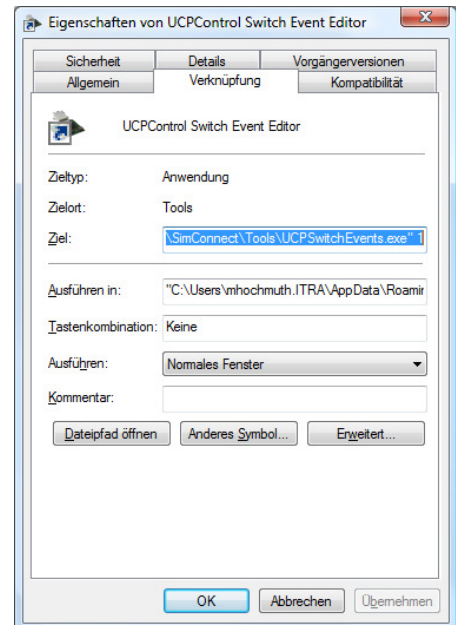


Standardmäßig wird der USB-Controller mit dem Index 0 ausgeliefert. Wenn mehrere Controller eingesetzt werden sollen, kann der Controller durch ITRA bei der Auslieferung einen anderen Index erhalten.

Alle Konfigurationsprogramme können mit einem Programm-Parameter dementsprechend aufgerufen werden. Erstellen Sie eine Kopie der Verknüpfung und benennen Sie diese entsprechend, z.B. mit der programmierten Kennung.

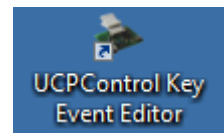
Bearbeiten Sie die Eigenschaften der neuen Verknüpfung indem Sie mit der rechten Maustaste auf die Verknüpfung klicken

Ergänzen Sie eine 1 (je nach Programmierung des Controllers) am Ende der Ziel-Zeile.



UCPCControl Key Event Editor

Mit dem Key Event Editor können der Tastatur-Matrix des USB-Controllers Simulator-Ereignisse zugeordnet werden. Zu jeder Taste kann zusätzlich eine Wiederholrate eingestellt werden.



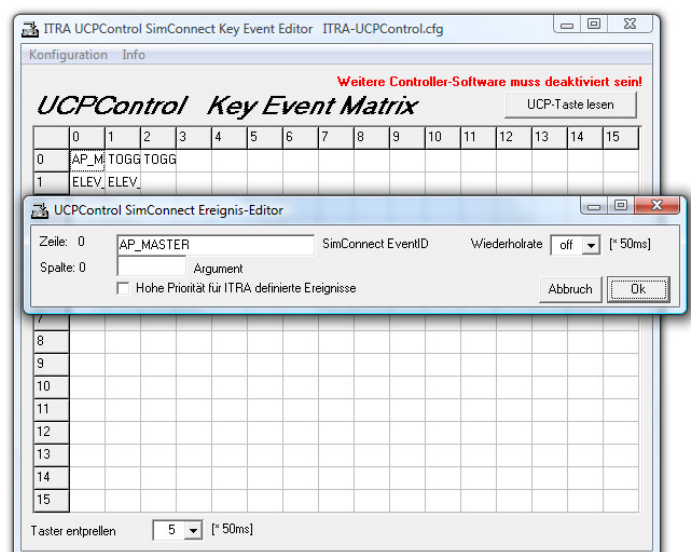
Zuerst müssen alle Programme deaktiviert werden, die mit dem USB-Controller kommunizieren.

Sie können eine bereits bestehende Konfiguration laden oder mit einer leeren Matrix beginnen und diese dann speichern.

Die Abfrage der Tasten erfolgt, indem die entsprechende Taste gedrückt wird und anschließend mit der Maus die Schaltfläche **UCP-Taste lesen** geklickt wird. Es öffnet sich die zugehörige Eingabe-Maske.

Sie können aber auch mit der Maus in der Matrix das gewünschte Feld doppelklicken, um die Eingabemaske zu öffnen.

Es werden Zeile und Spalte angezeigt sowie bereits definierte Ereignisse. Für einige Ereignisse ist ein zusätzliches Argument notwendig, so kann z.B. bei



DME_SELECT mit dem Argument 1 oder 2 das entsprechende DME ausgewählt werden.

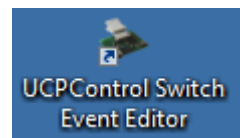
Für Ereignisse, die nicht nur zum Flugsimulator, sondern weiter zu anderen SimConnect-Clients gesendet werden sollen, wie z.B. zum TFT-Interface-Client, ist es notwendig die Ereignispriorität auf hoch zu setzen. Siehe dazu auch die Beschreibung der SimConnect-Clients A320. Dort sind auch die ITRA-SimConnect-Ereignisse definiert, die die A320-Clients verarbeiten können.

Der Standard-Abfragezyklus des SimConnect-Clients beträgt 50ms. Es kann mit **Taster entprellen** für alle Ereignisse eine Anzahl zusätzlicher Zyklen angegeben werden, in der ein Tastensignal gültig sein muss. Beachten Sie, dass dies beim Einsatz von Drehimpulsgebern nicht verwendet werden kann.

Die Belegung der Matrix kann mit **Druckvorschau** auch als Liste angezeigt werden. Diese Liste kann dann ausgedruckt werden.

UCPControl Switch Event Editor

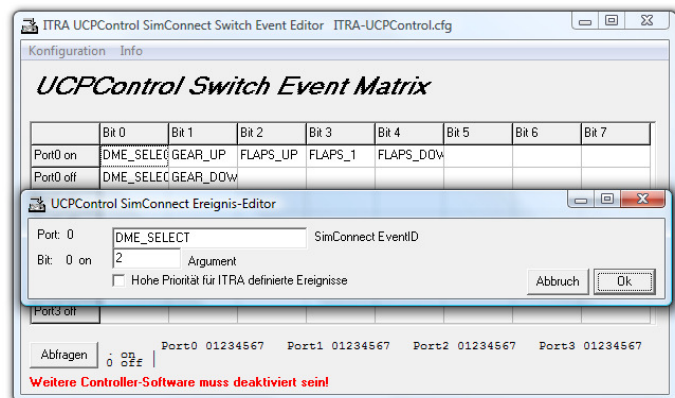
Mit dem Switch Event Editor können den Schaltwechselln der Schalter an einer Eingabe-Baugruppe UCP-Switch32 Simulator-Ereignisse zugeordnet werden. Es gibt jeweils ein ON- und ein OFF-Ereignis.



Zuerst müssen alle Programme deaktiviert werden, die mit dem USB-Controller kommunizieren.

Sie können eine bereits bestehende Konfiguration laden oder mit einer leeren Matrix beginnen und diese dann speichern.

Die Abfrage der Schalterwechsel erfolgt, indem der entsprechende Schalter betätigt wird und anschließend mit der Maus die Schaltfläche **Abfragen** geklickt wird. Es öffnet sich die zugehörige Eingabe-Maske. In der Zeile neben dem Abfrage-Knopf wird der Bit-Zustand der Ports angezeigt.



Sie können aber auch mit der Maus in der Matrix das gewünschte Feld doppelklicken, um die Eingabemaske zu öffnen.

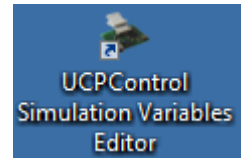
Es werden Port und Bit angezeigt sowie bereits definierte Ereignisse. Für einige Ereignisse ist ein zusätzliches Argument notwendig, so kann z.B. bei DME_SELECT mit dem Argument 1 oder 2 das entsprechende DME ausgewählt werden.

Für Ereignisse, die nicht nur zum Flugsimulator, sondern weiter zu anderen SimConnect-Clients gesendet werden sollen, wie z.B. zum TFT-Interface-Client, ist es notwendig die Ereignispriorität auf hoch zu setzen. Siehe dazu auch die Beschreibung der SimConnect-Clients A320. Dort sind auch die ITRA-SimConnect-Ereignisse definiert, die die A320-Clients verarbeiten können.

Die Belegung der Matrix kann mit **Druckvorschau** auch als Liste angezeigt werden. Diese Liste kann dann ausgedruckt werden.

UCPControl Simulation Variables Editor

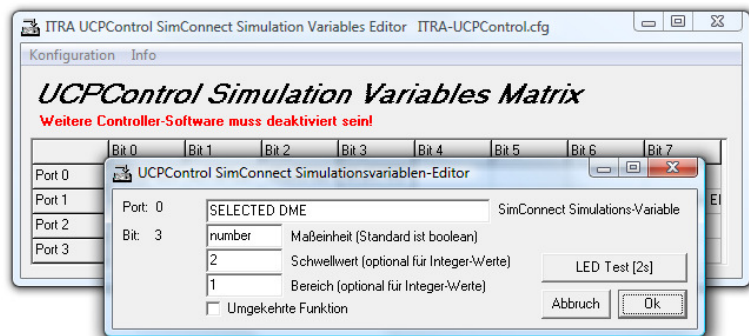
Mit dem Simulation Variables Editor können Simulator-Variablen den LED-Funktionen der Ausgabebaugruppe UCP-LED32 zugeordnet werden.



Zuerst müssen alle Programme deaktiviert werden, die mit dem USB-Controller kommunizieren.

Sie können eine bereits bestehende Konfiguration laden oder mit einer leeren Matrix beginnen und diese dann speichern.

Die Definition der Zuordnung erfolgt, indem Sie im entsprechenden Feld der Matrix doppelklicken. Die mit dem entsprechenden Ausgang verbundene LED kann mit dem Knopf **LED-Test** für 2 Sekunden getestet werden.

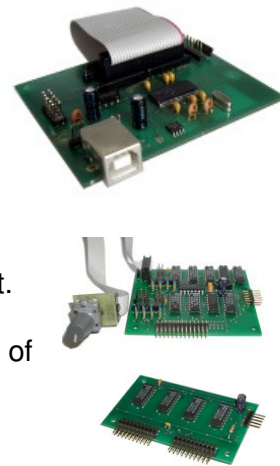


Für einfache ON-/OFF-Variablen (boolean) wird nur die Simulations-Variable in das Eingabefeld eingetragen. Im ON-Zustand ist der Ausgang aktiv. Mit dem Aktivieren von **Umgekehrte Funktion** ist der Ausgang im OFF-Zustand aktiv.

Wenn Sie Integer-Variablen verwenden wollen, dann muss die entsprechende Maßeinheit der Variablen angegeben werden (je nach Zuordnung der Variablen zu bestimmten Oberbegriffen z.B. Druck, Temperatur usw. sind mehrere Maßeinheiten möglich). Es kann ein Schwellwert (Level) angegeben werden, ab dem der Ausgang aktiv sein soll und ein Bereich (Range) oberhalb des Schwellwertes. Ein Bereich von 0 aktiviert den Ausgang bei allen Werten ab dem Schwellwert (einschließlich), ein Bereich von 1 aktiviert den Ausgang bei nur genau dem Wert des Schwellwertes. Ist ein Bereich größer 1 angegeben, dann wird der Ausgang aktiv, wenn der Wert größer/gleich dem Schwellwert und kleiner als Schwellwert+Bereich ist. Auch hierbei ist die umgekehrte Funktion des Ausgangs möglich.

Die Belegung der Matrix kann mit **Druckvorschau** auch als Liste angezeigt werden. Diese Liste kann dann ausgedruckt werden.

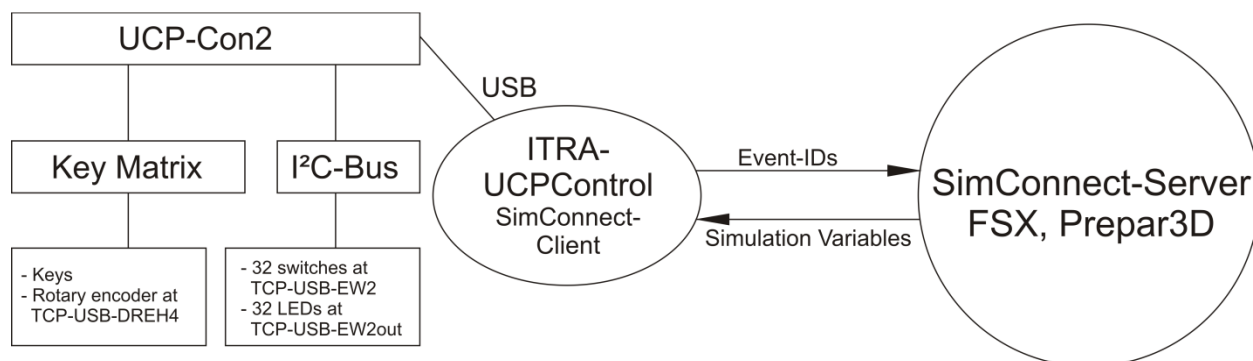
ITRA SimConnect-Client UCPCControl (for FSX and Prepar3D)



Introduction

SimConnect is an interface developed by Microsoft to realize the communication between external programs and the flight simulator. The SimConnect server is a component of the flight simulator and is installed with it. SimConnect clients may run on network PCs, but we are using only the local installation of all clients at the flight simulator PC because of the faster request of USB events of the rotary encoders.

Function diagram of the SimConnect client UCPCControl



The SimConnect client communicates over USB with the USB-Controller UCP-Con2. The events from the keys at the key matrix and the events from the switches at the input unit UCP-Switch32 are processed by the USB controller and depending on the configuration file are sent simulator events to the SimConnect server.

These events may be prioritized. So it is possible to send self-defined events over the SimConnect server to other SimConnect clients for example to the TFT-Interface client.

The UCPCControl client can request Simulation Variables. Depending on the configuration these variables are used to control e.g. LEDs at the ports of the output unit UCP-LED32.

System requirements

For the usage of the ITRA-SimConnect-Clients it is necessary to have the actual version of FSX or Prepar3D installed. You have also to install the actual drivers for the ITRA devices. Please install first the ITRA USB Driver Package before connecting the USB devices to the PC.

Installation folder of ITRA-SimConnect-Clients:

C:\Program Files (x86)\ITRA\SimConnect
C:\Program Files\ITRA\SimConnect

// Win7-64bit
// Win7-32bit

Location of configuration files of ITRA-SimConnect-Clients

C:\Users\Benutzername\AppData\Roaming\ITRA\SimConnect

Location of configuration files **exe.xml** and **dll.xml** of FSX

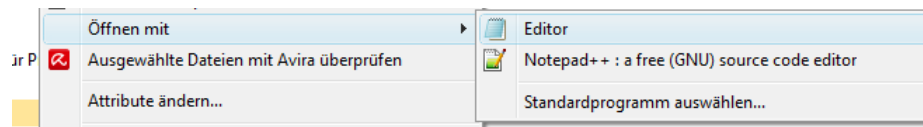
C:\Users\username\AppData\Roaming\Microsoft\FSX

Location of configuration files **exe.xml** and **dll.xml** of Prepar3D

C:\Users\username\AppData\Roaming\Lockheed Martin\Prepar3D

You can modify the configuration file manually with an editor. A link to the configuration file is created in the program folder by the setup.

You can open the file with double click or with the context menu of the right mouse button.

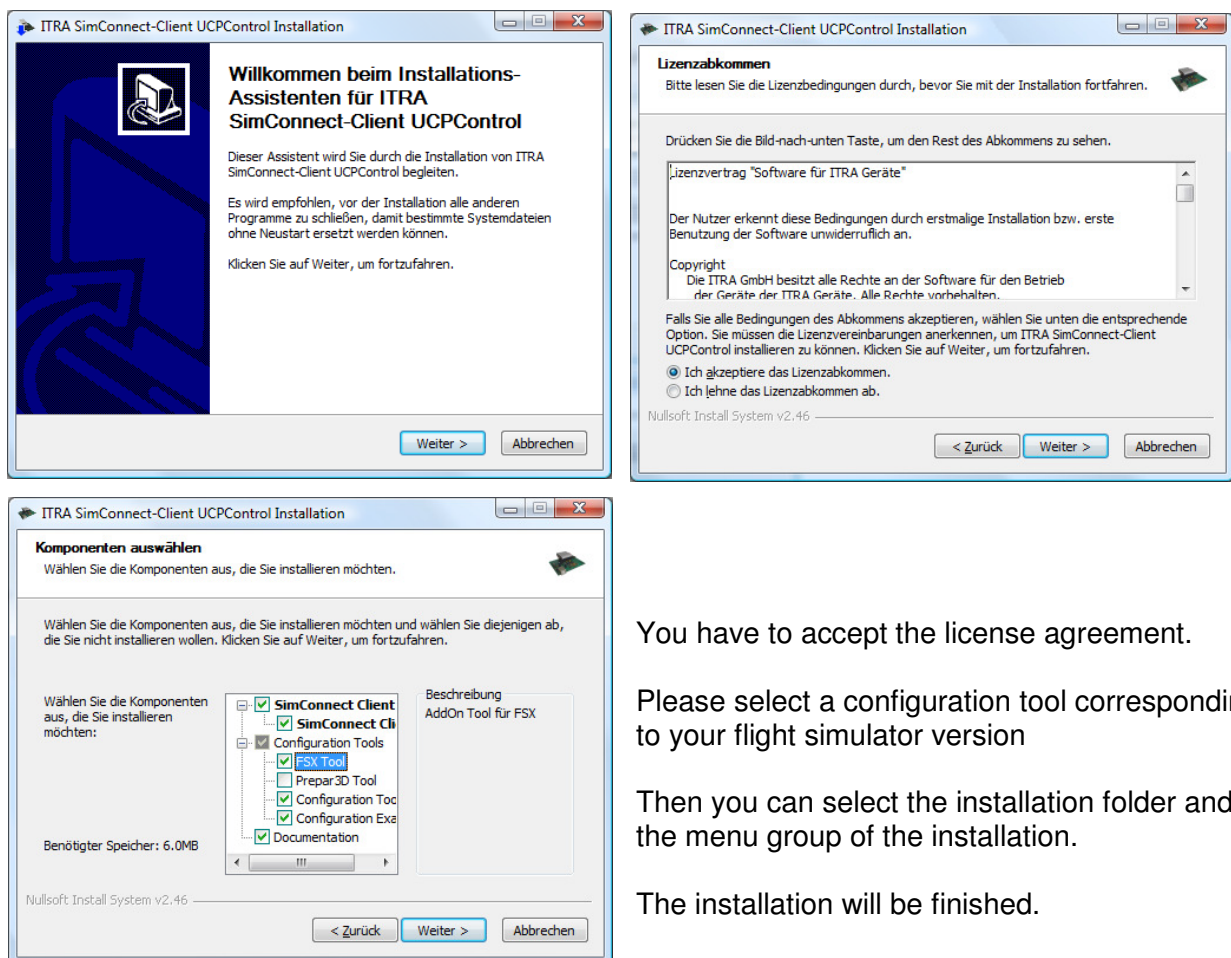


We recommend the usage of the supplied configuration programs.

Software-Installation

Attention: Please install first the USB Driver Package before connecting the controller to the PC. Additional information you will find in the installation instruction.

In order to install the software run the installation setup file.



You have to accept the license agreement.

Please select a configuration tool corresponding to your flight simulator version

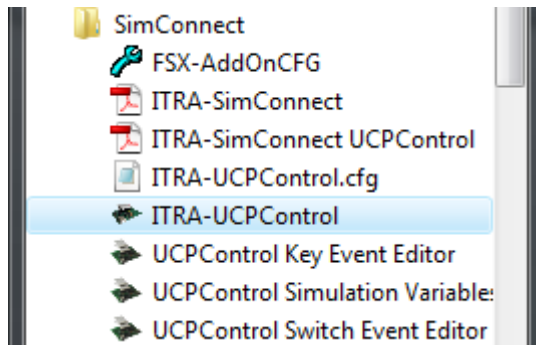
Then you can select the installation folder and the menu group of the installation.

The installation will be finished.

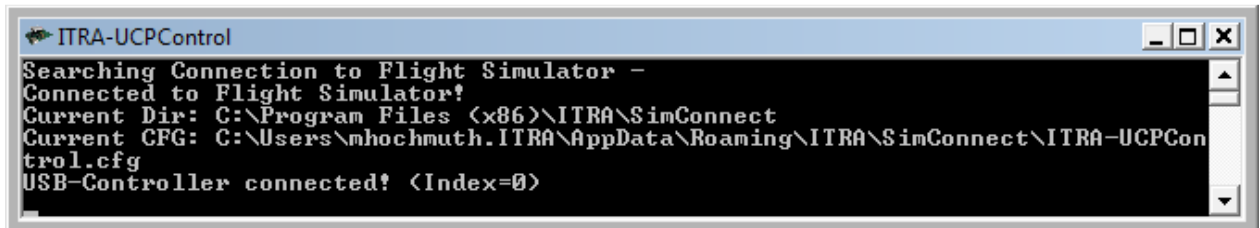
You will find the hardware description in the manuals of the UCP circuit boards.

SimConnect-Client-Software

If the installation is finished right, you will find in the program group ITRA -> SimConnect the SimConnect Client, manuals, a link to the configuration file and a program for modifying the start configuration of the flight simulator.

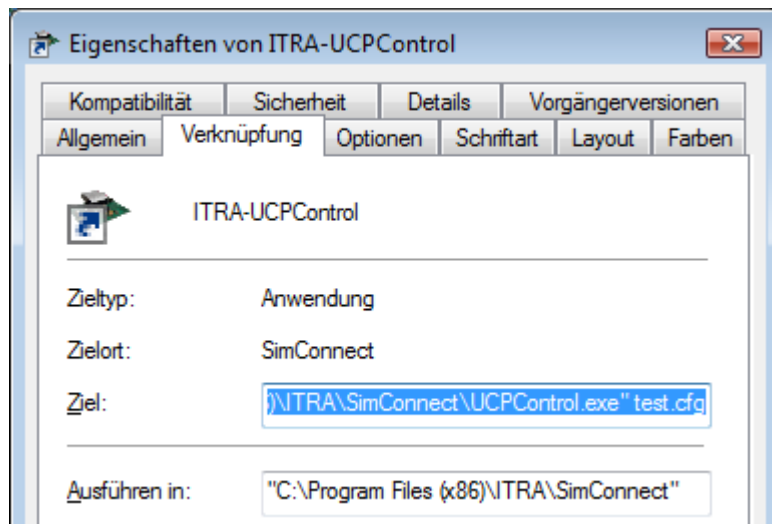


If the client is started the connection to the SimConnect server is searched. If the server is detected the connection will be opened, then the Flightboard is searched. If the device is detected the configuration file is loaded. Information about the status is written to the console output of the client.



For expert users:

Every Link to the client program can be modified with a separate configuration file at the end of the command line. The configuration file is searched in the above described AppData path and then in the application path of the SimConnect client.



Attention! If you let start the client with exe.xml from flight simulator then the application path is the path of that. Save your configuration best in the AppData path of the SimConnect Clients.

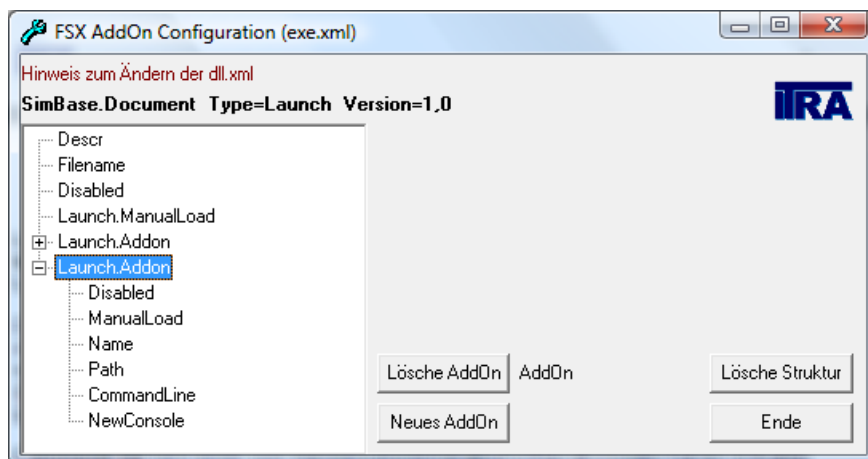
Configuration file exe.xml for automatic start of SimConnect Clients:

```
<SimBase.Document Type="Launch" version="1,0">
  <Descr>Launch</Descr>
  <Filename>EXE.xml</Filename>
  <Disabled>False</Disabled>
  <Launch.ManualLoad>False</Launch.ManualLoad>
```

...
// for every client you have to make such a section

```
<Launch.Addon>
  <Disabled>False</Disabled>
  <ManualLoad>False</ManualLoad>
  <Name>UCPControl</Name> // change name here
  <Path>C:\Program Files (x86)\ITRA\SimConnect\UCPControl.exe</Path> // Win7-64bit
  <Path>C:\Program Files\ITRA\SimConnect\UCPControl.exe</Path> // Win7-32bit
  <Path>C:\Programme\ITRA\SimConnect\UCPControl.exe</Path> // XP
  <CommandLine></CommandLine>
  <NewConsole>True</NewConsole>
</Launch.Addon>
...
</SimBase.Document>
```

For a simpler modification of this file you can use the programs **FSX-AddOnCFG** / **Prepar3D-AddOnCFG**. The file exe.xml is searched in AppData path of the flight simulator, the structure is detected and shown. You can add and remove AddOns and change the values. If the file is not found or empty, a new tree structure will be generated.



Clear Tree generates a new clean structure of the tree.

New AddOn generates a new entry for an AddOn, **Disable** is set to True.

Del AddOn removes the marked AddOn, the name is shown beside the button.

Path has an input line to define the client, with the button beside you can search the SimConnect client. If you have selected the client then the AddOn will be activated (**Disable**=False) and the generated **Name** is taken from the file name and can be changed.

CommandLine may contain a configuration file name if not the default should be used.

ManualLoad generates every time when the flight simulator is started a question if the AddOn should start.

Exit terminates the program, you will be asked if save the file or not.

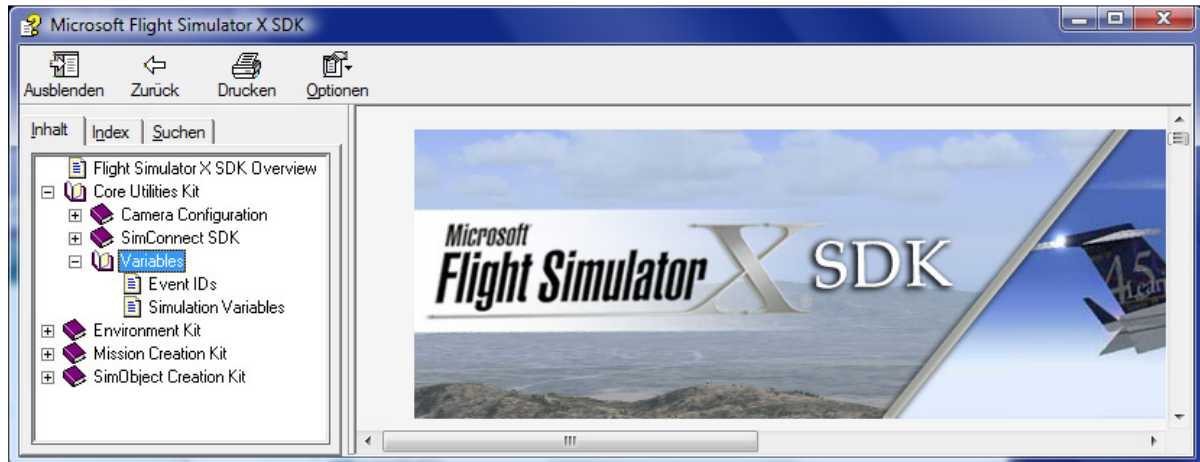
The file is generated depending on the definitions in the description of the SimConnect Interface. In the header you have to set the value **Disable** to False (not marked in the checkbox) to process the complete file, all other entries are defined according to the documentation of SimConnect. **Launch.ManualLoad** generates for every AddOn entry the question if it should be started or not.

Hint: If the SimConnect clients are starting with an error message then the actual runtime package for the programming environment have to be installed

Microsoft Visual C++ 2010 Redistributable Package

You will find this on the Microsoft download page or with a link on our support website.

Structure of the configuration file



In the documentation of the SimConnect-SDK are described **String Names** in the section **Event-IDs**, which are used for defining key and switch events of the controller.

<http://msdn.microsoft.com/en-us/library/cc527015.aspx>

Event ID	String Name	Description	Multiplayer
KEY_AP_MASTER	AP_MASTER	Toggles AP on/off	Shared Cockpit
KEY_AUTOPILOT_OFF	AUTOPILOT_OFF	Turns AP off	Shared Cockpit
KEY_AUTOPILOT_ON	AUTOPILOT_ON	Turns AP on	Shared Cockpit

The strings that are described in the section **Simulation Variables** are used to control LEDs.

Simulation Variable	Description	Units	Settable	Multiplayer
AUTOPILOT AVAILABLE	Available flag	Bool	N	-
AUTOPILOT MASTER	On/off flag	Bool	N	Shared Cockpit

The default file name of the configuration file is ITRA-UCPCControl.cfg.

In the configuration file of the SimConnect client for the UCP-Con2 are four sections defined:

```
[UCPCONTROL]           Global definitions
INDEX=0                -> Index of the UCP-Con2 or TCP-USB-S1 depending of the
                        delivered version, default is 0
DEBOUNCE=2            -> key debouncing (value*50ms)
```

[UCP_KEYS] **Key definitions**

Z0_S0=DME_SELECT -> The string definition for the key events that are defined and described in the SimConnect-SDK in the section Event-IDs, Example for DME selection

Z0_S0_ARG=2 -> Some events need an additional parameter
Example for DME2
As parameter are only numbers possible

Z0_S0_REPEAT=0 -> {0..10} factor that defines the repeat function, 0=off
1 complies to 50 ms
10 complies to 500ms

Z0_S0_PRIORITY=0 -> {0,1} to set to a high event priority for sending self-defined events to other SimConnect clients
e.g. to the ITRA-Interface client to control ITRA A320 software, the string definitions are specified there

...

Z8_S15
Z8_S15_ARG
Z8_S15_REPEAT
Z8_S15_PRIORITY -> at this time 144 keys of the UCP-Con2 are supported from Row0/Column0 to Row8/Column15

[UCP_SWITCHES] **Switch definition if using the extension unit**
For switches UCP-Switch32

P0_B0_OFF -> The string definition for an OFF-Event like defined and described in the SimConnect-SDK in the section Event-IDs

P0_B0_ON -> The string definition for an ON-Event like defined and described in the SimConnect-SDK in the section Event-IDs

P0_B0_OFF_ARG -> Some events need an additional parameter
As parameter are only numbers possible

P0_B0_ON_ARG -> Some events need an additional parameter
As parameter are only numbers possible

P0_B0_PRIORITY=0 -> {0,1} to set to a high event priority for sending self-defined events to other SimConnect clients

...

P3_B7_OFF
P3_B7_ON
P3_B7_OFF_ARG
P3_B7_ON_ARG
P3_B7_PRORITY -> One UCP-Switch32 with 32 switches is supported at the UCP-Con2

A single event is generated if the state of the switch is changed.

[UCP_LEDS] **Output definition for LED control if using unit UCP-LED32**
(I²C-Adresse 24..27)

P0_B0=SELECTED DME -> String definition for simulation variable like described in the SimConnect-SDK in the section Simulation Variables

P0_B0_UNIT=number -> Unit corresponding to the definition of the variable.
Default is Boolean; this definition is not necessary if used a boolean variable

P0_B0_LEVEL=2 -> If using an Integer variable a switch level may be defined at that the output is set. RANGE controls the manner

P0_B0_RANGE=1 -> If LEVEL is used, you may define a value range at that the output is activated (default is 0)
0: all values equal and greater LEVEL are activating the output
1: only value LEVEL activates the output
greater 1: from LEVEL to lower LEVEL+RANGE the output is activated

P0_B0_INVERSE=0 -> {0,1} The logic at the output is inverted.

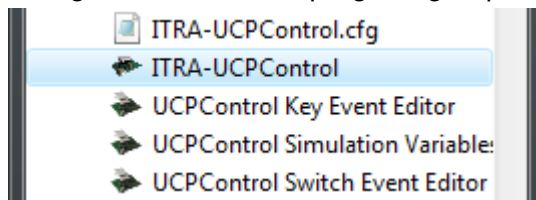
...

P3_B7
P3_B7_UNIT
P3_B7_LEVEL
P3_B7_RANGE
P3_B7_INVERSE -> One UCP-LED32 with 32 outputs is supported at the UCP-Con2

You can install an example configuration file.

Programs to modify the configuration file

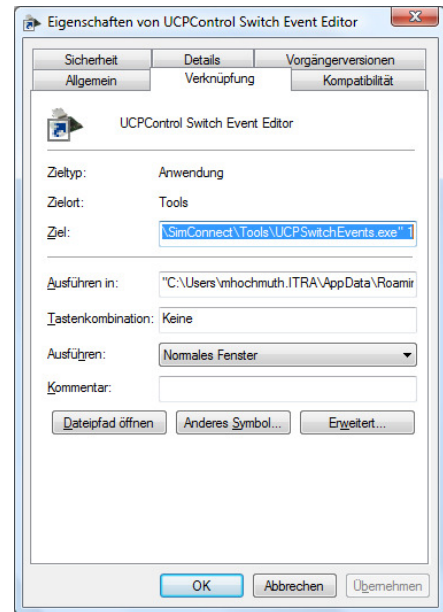
After the installation there are found three programs for creating or modifying the UCPCControl configuration file in the program group ITRA – SimConnect.



The USB-Controller is delivered with the default index 0. If more than one controller should be used it is possible to get a controller with another index programmed by ITRA.

All configuration programs may be started with a program parameter. You can make a copy of the program link and rename it depending on the index.

You can change then with a mouse click on the link the properties. Now you can modify the destination field by appending the controller index at the end of the line.



UCPCControl Key Event Editor

With the Key Event Editor it is possible to define simulator events to the keyboard matrix of the USB-Controller. For every event a repeat rate is possible.



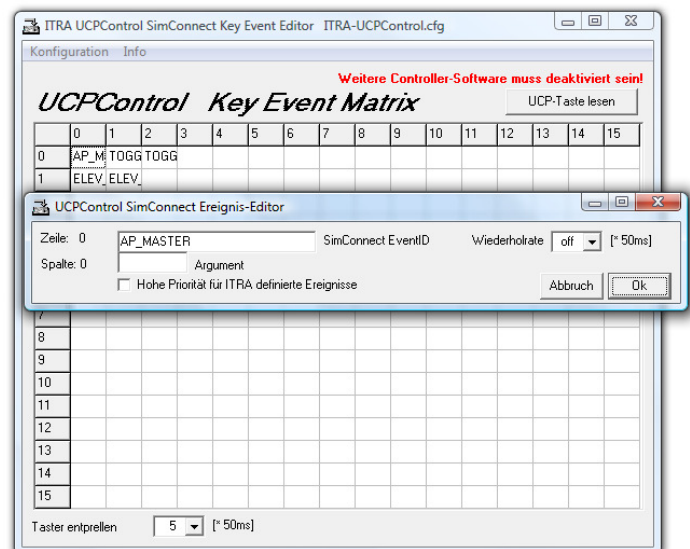
All programs that are communicating with the USB-Controller must be closed.

You can load an existing configuration file or you can begin with a clear matrix and save it at the end.

Now a key of the controller matrix is pressed and then you can request this controller event with **Read UCP-Key**. The associated input mask will open.

The input mask opens also with a double click in the matrix field.

Row and column are displayed and also already defined events. For some events it is necessary to define an argument such as DME_SELECT. With 1 or 2 the corresponding DME is selected.



If events should be sent not only to the FS but to other SimConnect clients like TFT-Interface client so you have to define a high priority. See also the description of A320 SimConnect clients. There are written the ITRA SimConnect events that the A320 clients may process.

The default request time of the SimConnect client is 50ms. With **Debounce** you can define for all events the number of additional request times in that the key event of the controller must be active. This cannot be used if rotary encoders are used.

You can show a list of the defined matrix with **Print preview** and then print this list.

UCPControl Switch Event Editor

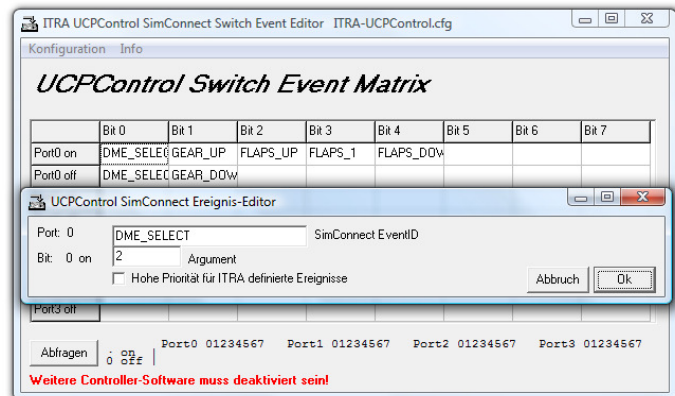
With the Switch Event Editor it is possible to define simulator events to the switches of the input unit UCP-Switch32 of the USB-Controller. For every event a repeat rate is possible. Every switch has an ON- and OFF-event



All programs that are communicating with the USB-Controller must be closed.

You can load an existing configuration file or you can begin with a clear matrix and save it at the end.

Now a switch is changed. Then you can click with the mouse on **Request**. The associated input mask will open. In the line next to the button the bit status of all ports is shown.



The input mask opens also with a double click in the matrix field.

Port and Bit are displayed and also already defined events. For some events it is necessary to define an argument such as DME_SELECT. With 1 or 2 the corresponding DME is selected.

If events should be sent not only to the FS but to other SimConnect clients like TFT-Interface client so you have to define a high priority. See also the description of A320 SimConnect clients. There are written the ITRA SimConnect events that the A320 clients may process.

You can show a list of the defined matrix with **Print preview** and then print this list.

UCPControl Simulation Variables Editor

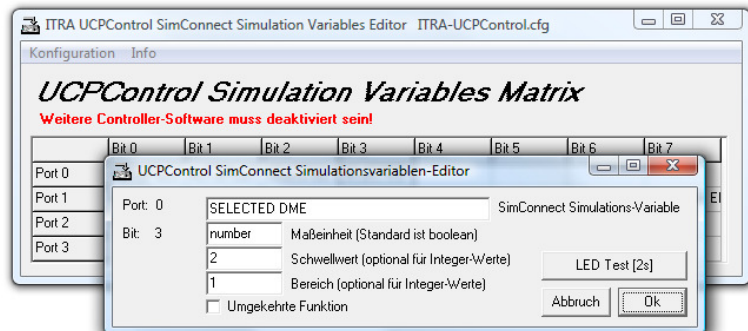
With the Simulation Variables Editor it is possible to connect the status of simulation variables of the FS to the LEDs of the output unit UCP-LED32.



All programs that are communicating with the USB-Controller must be closed.

You can load an existing configuration file or you can begin with a clear matrix and save it at the end.

To define an output you have to open the input mask with a double click in the associated matrix field. The associated LED may be checked for 2 seconds with the button **LED Test**.



For simple ON/OFF variables (boolean) only the Simulation Variable is entered. If the variable is ON the output is activated. With **Inverse Function** the output is activated if the variable is OFF.

You can use also integer variables. Then a unit must be defined (on many variables depending on generic terms like pressure, temperature etc. are different units of measurement are possible). You can define a level value at that the output is activated and a range. If the range is set to 0, all values above level will activate the output. A range value of 1 will set the output active only at the value of level. If a range greater than 1 is defined, then the output is activated from level to lower as level+range. Also the inverse function is possible.

You can show a list of the defined matrix with **Print preview** and then print this list.