

Anpassung IPSACCESS für Zugriffe auf DevMan-IPS-Boxen (Anschluss über PowerPC VME-Rahmen)

IPSACCESS unter OpenVMS

Zugriffe aus IPSACCESS auf die Geräte-SW erfolgen per Userface, also über normale Properties.

Dabei werden in den meisten Fällen synchrone Einzel-Zugriffe (ein Kommando pro Auftrag) durchgeführt. Derartige Zugriffe sollten sich problemlos über die automatischen Umsetzmechanismen (Userface-Corba Server) abhandeln lassen. Es wird nicht erwartet, dass diese synchronen Aktionen sehr zeitkritisch sind. Als einzige Ausnahme mag der Text für das Display der Incremental-Encoder sein. Aber auch hier sollte eine verzögerte Update-Rate bei der Ausgabe nicht sehr stören.

Eine Besonderheit stellt dagegen das Lesen der Incremental-Encoder sowie der Keys (was ist das?) dar. Die Übergabe der Eingabe-Werte an den Anwender erfolgen über je eine Callback Funktion, die als AST-Routine aufgerufen werden. Das Lesen der Werte erfolgt hierbei ebenfalls über Userface, und zwar als konnektierter Auftrag. IPSACCESS verwendet zum Abholen der eingetroffenen Werte den directReceive-Mechanismus, also Userface-Callbacks. Somit ist IPSACCESS die hoffentlich einzige Stelle, die den Userface-Callback Mechanismus verwendet.

Diesen Userface-Callback Mechanismus auch im Userface-Server zu implementieren, wäre sehr aufwändig. Deshalb wird hier eine Anpassung in der IPSACCESS- und IPS-VME-Software vorgeschlagen, die sich sicherlich einfacher umsetzen lässt als eine allgemeingültige Behandlung im Userface selber: Übertragung der gelesenen Werte über eine spezielle UDP-Verbindung.

Die Daten werden über je ein spezielles Packetformat per UDP von den IPS-Boxen zum IPSACCESS übertragen. Da es nur je ein Datenformat gibt, ist kein spezielles Protokoll erforderlich, und die Umwandlung der Byteorder (big endian nach little endian) kann explizit ausprogrammiert werden.

IPSACCESS richtet für Incremental-Encoder und Keys je einen UDP-Port ein. Dazu wird, ab einer Anfangsnummer, so lange versucht, UDP-Ports zu öffnen, bis ein unbenutzter gefunden wird. Auf diese Weise sind auch mehrere IPS-Boxen auf einem Rechner ohne Konflikte zu handhaben.

Die gefundene Port-Nummer wird, zusammen mit dem Rechnernamen und einer Kennung (siehe unten) an die VME-Geräte-SW über einen normalen synchronen Userface-Zugriff übertragen,. Danach hört IPSACCESS an dem geöffneten Port und nimmt Datenpakete entgegen. Zum Übertragen von Portnummer und Rechnernamen könnte eine neue Property implementiert werden. Zweckmäßigerweise werden diese Informationen aber besser als Parameter der aufgerufenen Property übertragen. Da dazu ein einheitliches Datenformat erforderlich ist, sollte nicht der Rechnernamen übertragen werden, sondern dessen IP-Adresse, so dass nur numerische Integer-Werte ausgetauscht werden müssen.

Die VME-Software sendet nun gelesene Daten, wie die ausgelesenen Incremente, per UDP an die angegebene Adresse.

Aufpassen: Was ist, wenn eine Konnektierung nicht ordentlich abgeräumt wird? Dann werden munter Pakete geschickt, die beim nächsten Öffnen eines Empfangsports in IPSACCESS womöglich fälschlicherweise entgegengenommen werden können. Um das festzustellen, könnte als Kennung Datum und Uhrzeit des Zeitpunkts der Initialisierung der

Konnektierung mitgeschickt werden. Ausserdem ist über Exit-Handler darauf zu achten, dass Konnektierungen bei Ende des Programms sauber beendet werden.

Dieser speziell für OpenVMS vorgesehene Mechanismus der Datenübertragung per UDP wird nur für eine längere Übergangszeit benötigt. Mittelfristig ist vorgesehen, OpenVMS komplett abzulösen und auf der Operating-Ebene nur noch Linux einzusetzen. Unter diesem Betriebssystem können die erforderlichen Callback-Aufrufe in IPSACCESS direkt innerhalb der DevMan Callback-Aufrufe erfolgen

IPSACCESS unter Linux

Hier ist das Hauptproblem, dass die IPSACCESS-Software in Modula-2 implementiert ist. Diese Software müsste wohl in eine gängigere Sprache übertragen werden. Alternativ könnte versucht werden, einen Modula-2 Compiler zu finden (es gab mal so was wie Mocka).

Da es wieder eine Userface-kompatible Schnittstelle für Geräte-Zugriffe geben soll, können die synchronen gerätezugriffe unverändert wie unter OpenVMS bleiben. Wenn der Code ohnehin in eine andere Programmiersprache übertragen wird, ist es möglicherweise sinnvoll, direkt die CORBA-Schnittstelle für Zugriffe zu verwenden.

Für die IPSACCESS Callback-Aufrufe werden praktischerweise die CORBA-Aufrufe eingesetzt, da sie schon die Callback-Funktionalität bieten. Die etwas umständliche UDP-Datenverbindung kann dann wieder entfallen.