Beschreibung zum hipecs Server Demo

Dieses Projekt kann entweder in Verbindung mit einem Terminal-Programm oder mit einer zweiten hipecs mit dem „hipecs TCP client demo“ Benutzt werden. Dieses Demo zeigt wie man mit der hipecs PLC einen Server im Netzwerk aufsetzen und betreiben kann.

Einführung

Um dieses Demo zu betreiben, wird mindestens eine hipecs-Steuerung mit Netzwerkanschluss benötigt. Also eine hipecs CORE10 oder eine hipecs PLC1020,1030,1220 oder 1230. Ein funktionsfähiges Netzwerk ist selbstverständlich ebenfalls nötig.

Vorbereitung

Hardwareseitig sind keine besonderen Vorbereitungen nötig, eine Netzwerkverbindung natürlich vorausgesetzt.

Ein richtig konfiguriertes system.ini file im Bootverzeichnis ist von Vorteil, aber nicht zwingend nötig. Es ist standardmäßig DHCP aktiviert und die Netzwerkparameter können aus der Visu abgelesen werden.

hipecs setup

Wenn ein frenzel + berg Visupanel vorhanden ist, kann dieses über COM3 (RS422) angeschlossen werden. Die Baudrate beträgt 460.000 Baud. Die DIP-switches am Panel auch überprüfen!

Die Datei “app_hipecssocketserver_v1r1_en.prj” ist das CODESYS project für dieses Beispiel. Die Kommunikationseinstellungen sind standardmäßig auf COM15 und 57600 Baud eingestellt. **Die muss natürlich an den jeweiligen PC angepasst werden!**

Starten und Handhabung des Projektes

Aus Gründen der Übersichtlichkeit, ist es natürlich vorteilhaft, die Visualisierung zu nutzen. Der Visubaustein “Filesystem_VISU” bietet verschiedene graue Schaltflächen. Diese führen die entsprechende Funktion einmal aus. Die blauen Schaltflächen dienen zur Eingabe von Parametern bzw. Strings. In diesem Demo zum Beispiel eine bestimmte Portnummer oder ein Hostname, der gesucht werden soll. Die Suche nach „localhost“ liefert die IP der Steuerung zurück.

Achtung!! Ganz im Gegensatz zur Vorgehensweise in diesem Beispiel, sollte bei einer realen Anwendung immer ein separater Task für Socket-Funktionen genutzt werden. Diese Bibliothek enthält blockierende Netzwerkfunktionen, die trotz des Multitasking-Betriebssytemes der hipecs die Programmabarbeitung der Steuerung bzw. den entsprechenden Task blockieren kann. Wird PLC_PRG im selben Task ausgeführt wie die Socket-Funktionen, kann durch das Blockieren der Output-Watchdog ausgelöst werden, was zum Abschalten der Ausgänge führen würde. Abhängig von der Anwendung möchte man das womöglich nicht riskieren! Deshalb: Socketfunktionen in einen extra Task -> Gefahr gebannt!

PLC_PRG

Dieses Projekt nutzt nur den PLC_PRG Baustein. Alle Funktionen warden hieraus aufgerufen. Es werden nur Funktionen der „SysLibSocket“ Bibliothek genutzt. Alle blauen Schaltflächen weisen der Variablen „Step“ einen Wert zu, sodass die entsprechende Funktion einmal aufgerufen wird. Der „Step“ wird dann wieder auf 0 zurückgesetzt. Die grünen bzw. roten Kreise in der Visu zeigen an, ob die Funktion erfolgreich ausgeführt wurde.

Reihenfolge der Befehle/Funktionen

Um einen Server zu starten sind folgende Schritte nötig:

1. Den Server-Port initialisieren (init server port)
2. Socket erstellen (create)
3. Socket binden (bind)
4. hören (listen)
5. und die Verbindung akzeptieren (accept)

Man kann diese Schritte nacheinander ausführen und danach mit dem Client verbinden

TCP client

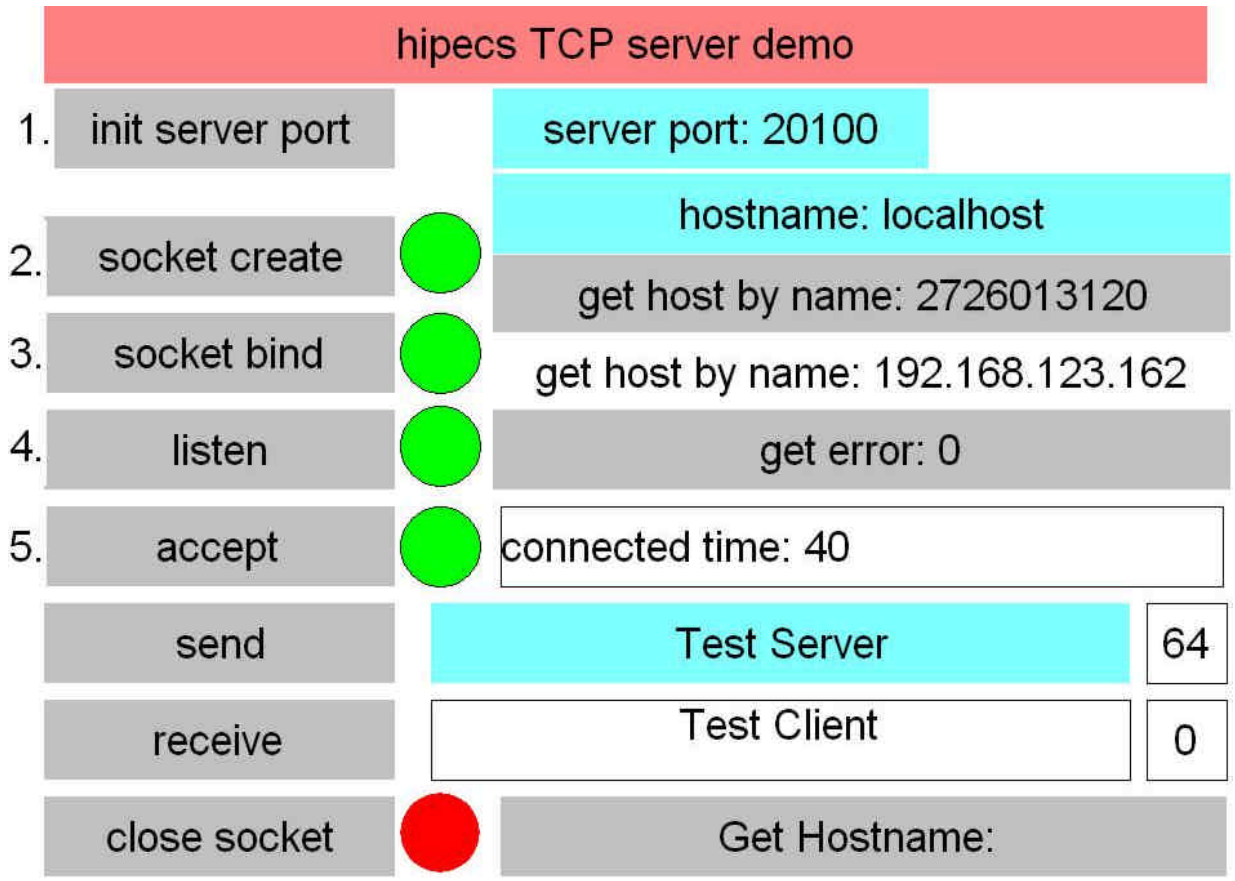
Es gibt verschiedene Möglichkeiten sich mit dem Server nun zu verbinden

1. eine weitere hipecs mit dem Demoprojekt "hipecs TCP client demo"
2. Am einfachsten mit einem Terminalprogramm wie beispielsweise PuTTY oder auch das Hyperterminal
3. eine eigene Client-Anwendung

Hier beispielsweise ein Screenshot aus PuTTY:

Beschreibung zum hipecs Server Demo

Visu



Schaltfläche	Funktion
Init server port	Setzt die Portnummer auf die verbunden warden kann. Default ist 20100
socket create	erstellt einen Spcket „SysSockCreate()“
socket bind	binded den Socket„SysSocketBind()“
listen	Ruft Funktion „SysSocketListen()“ auf und wartet auf Clients
accept	Ruft Funktion „SysSocketAccept()“ auf um eingehende Verbindung zu akzeptieren. Es wird auch ein neuer Handle vergeben!
send	Sendet die Daten, die rechts in der Box stehen
receive	ruft Funktion „SysSocketReceive“ auf Achtung: Diese Funktion wird automatisch aufgerufen, sobald eine Verbindung hergestellt wurde.
close socket	schließt die Socketverbindung
get host by name	liefert die IP zum darüber stehenden Hostnamen localhost liefert die eigene IP
get error	liefert den letzten Fehlercode zurück
Get Hostname	liefert den hostnamen der hipecs zurück