

Serielles Protokoll selbst programmiert - Mit offener Sprache zu mehr Wertschöpfung

Autorenangaben



Name: Armin Beck
Firma: Hilscher GmbH
Position: Produktmanager
Adresse: Rheinstraße 15
PLZ,Stadt: 65795 Hattersheim
Fon: 06190-9907-849
Fax: 06190-9907-50
Email: abeck@hilscher.com

Um Geräte mit proprietärem seriellen Protokoll an einen standardisierten Industriebus zu koppeln kommen heute Gateways zum Einsatz. Per Skriptsprache implementiert der Kunde bei Hilscher das Protokoll selbst, ohne auf fremde und kostenintensive Dienst- und Programmierleistung zurückgreifen zu müssen. Die Lösung „netSCRIPT“ verwendet hierbei bewusst als Basis die kostenlose Open-Source Skriptsprache Lua. Dank einfachster Sprachsyntax, kleinem Footprint und ihrer Schnelligkeit etablierte sich Lua innerhalb von 20 Jahren zu den meist verbreitetsten Skriptsprachen am Markt. Die offene Sprache bietet, wenn einmal erlernt, weitere Wertschöpfung im Gesamtunternehmen, wohingegen proprietäre Gateway-Programmierlösungen mit unbekannter Programmiersprache sich weiterem Nutzen verwehren.



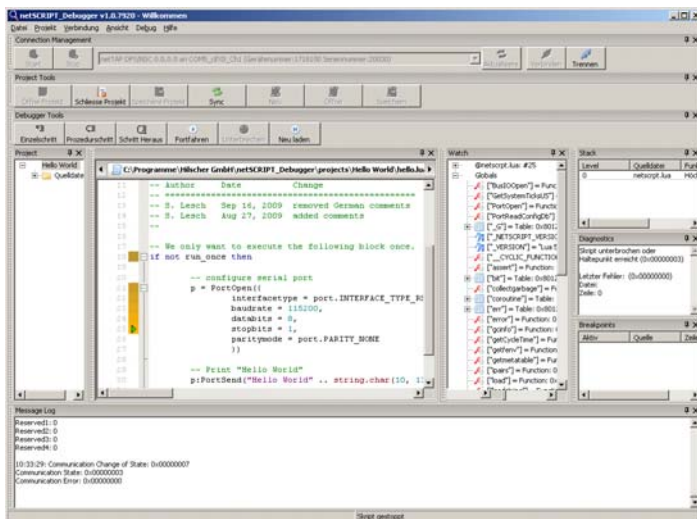
Programmierbares Gateway

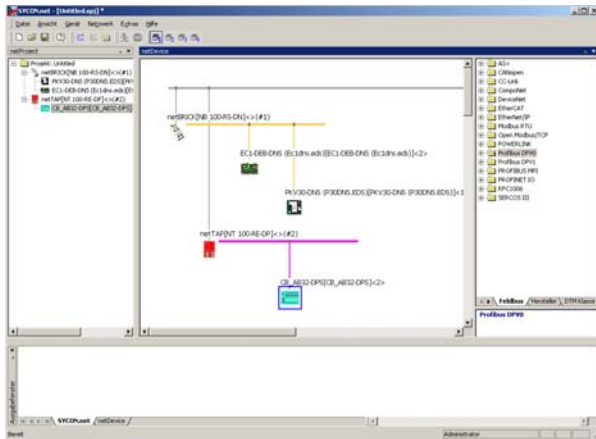
Die Sprache Lua ist in ihrer „Leichtigkeit“ selbst für Programmierunerfahrene geeignet. Das Basiswissen aus der Schulzeit reicht aus sie zu beherrschen. Die Sprache wird heute vornehmlich eingesetzt, um vom Profi in Hochsprachen wie C/C++/C# geschriebene Applikationen oder Bibliotheken über ein Lua-Frontend anzuwenden. Das eingebettete Skript interagiert dabei mit der Kernsoftware und macht dessen Funktionen einfach und schnell nutzbar. Ob Win32, Linux, Unix, MAC OS X, android, für nahezu alle Plattformen gibt es heute Lua-Interpreter mit fertigen Grafikbibliotheken, Dateisystem-Zugriffsfunktionen, Ethernet Socket-Interface oder Web-Server, hervorragend geeignet typische Applikationen in Unternehmensprozessen zu realisieren. Effektiv lassen sich beispielsweise

automatisierte Tests in der Produktion schreiben, die anfallende Mess- und Kennwerte zwecks Qualitätssicherung und Rückverfolgbarkeit erfassen oder eine neue Softwarekomponente in ihrem Verhalten per Skript-Template auf Kompatibilität zum Vorgänger prüfen.

netSCRIPT erweitert Lua um Kommunikationsfunktionen der seriellen Schnittstelle und den E/A-Datenzugriffen zum übergeordneten Bus. Die mathematische Lua-Standardbibliothek ermöglicht die Berechnung komplexer Algorithmen für Regelstrecken unmittelbar im Gerät. Mit 2MByte Skript-Speicher lassen sich ganze Kommunikations-Profile abbilden oder komplette Applikationen implementieren. Was bislang der Endanwender aufwändig in der SPS-Steuerung zu realisieren hatte, übernimmt nun das intelligente Gateway.

Die Datenzugriffe erfolgen ausschließlich über Variablen, indirekte Adressierung über Zeiger wie bei C gibt es nicht. Das schützt vor unerlaubten und ungewollten Speicherzugriffen und sichert die gesamte Gateway-Anwendung gegen unvorhersehbare Systemabstürze. Mit der netSCRIPT-Entwicklungsumgebung wird das Skript programmiert und per Debugger der korrekte Programmablauf geprüft. Schrittweise oder prozedural lassen sich alle Programmteile und Variablen systematisch kontrollieren.





SYCON.net Buskonfigurationswerkzeug

Die Erfahrung zeigt, dass selbst ein anspruchsvolles serielles Protokoll innerhalb eines Tages per netSCRIPT implementiert ist. Wie einfach netSCRIPT anzuwenden ist zeigt ein typisches Rohgerüst eines Programmes:

```
-- Öffne UART und IO Bus Schnittstellen
uart = PortOpen()
io = BusIOOpen()
...
-- Lade Ausgangsdaten und versende
output = io:BusIORead()
if output then
    uart:PortSend(output)
end
...
```

Skript-Auszug - Lesen von Ausgangsdaten und serielles Versenden

...Open() öffnet den Zugang zum seriellen und übergeordneten Bus. Das Lesekommando BusIORead() liest die Bus-Ausgangsdaten und per PortSend() werden die Daten über die serielle Schnittstelle versendet. Dazwischen lassen sich die „output“-Daten beliebig im Skript modifizieren.

Eine Open-Source basierende Programmierlösung anzubieten folgt einem allgemeinen Trend in der Automatisierung. Gerätehersteller gehen dazu über Produkte auf Basis von Linux zu entwickeln, ja sogar eigenen Quellcode zu veröffentlichen. Das Kollektiv arbeitet dann an Produktverbesserungen und das spart Entwicklungskosten. Der Anwender erhält im Gegenzug fehlerfreie, nutzerfreundliche und allseits bewährte Produkte. So ist netSCRIPT dank Lua von Beginn an eine leistungsstarke und flexibel ausgelegte Programmierlösung. Ist Lua bereits im Unternehmen bekannt, ist der Lernaufwand gering. Andere Unternehmen für die Lua neu ist, profitieren von dem vielfältigen Angebot an frei zugänglichen Lua-Applikationen mit hohem Wiederverwertbarkeitsfaktor in weiteren Geschäftsprozessen.

Mehr Infos unter www.hilscher.com, Produktbereich Gateways/netTAP.