

geschaffen, das zusammen mit etwas RAM auf dem Board eines Programmers steckt. Dieser Chip hat zwei Aufgaben: Zum einen programmiert man damit ganz gewöhnlich den Ziel-Mikrocontroller mit Hilfe des angeschlossenen PCs. Zum anderen stellt er Utilities zur Verfügung, mit denen man den Ziel-Controller bei der Programmausführung überwachen kann. Einige Utilities sind bekannt: ein Mehrkanal-Logic-Analyzer oder ein Mehrkanal-Oszilloskop. Andere Utilities hingegen nutzen höchstens professionelle Programmierer, doch dazu später mehr.

Wie funktioniert das?

Eigentlich ganz einfach: Der Ghost-Controller steckt in einem 100-poligen Quad-Flat-Pack-Gehäuse und verfügt daher über mehr als genügend IO-Leitungen. Der Ziel-Mikrocontroller hingegen hat maximal 35 IO-Pins. Jeder Pin dieses Ziel-Mikrocontrollers ist nun 1:1 mit einem Pin des Ghost-Mikrocontrollers verbunden, und zwar höchst „geisterhaft“ über einen hochohmigen Widerstand. Auf diese Weise wird der Betrieb des Ziel-Controllers nicht gestört. Während des Betriebs aber überwacht Ghost alle Pegeländerungen des Ziel-Controllers und speichert diese Information in einem 4 MB großen statischen RAM-Buffer. Ghost schickt die gesammelten Daten dann an den PC zur weiteren Verarbeitung. Das wars schon. Das Ghost-OS stellt auch weitere Funktionen wie das Anhalten des Ziel-Controllers während der Ausführung von Code sowie das Dumping von Register-Inhalten und die Überprüfung sonstiger Werte zur Verfügung, die das Entwickeln erleichtern. Wenn diese Daten dann in den PC übertragen wurden, müssen sie dort natürlich auch noch verarbeitet und dargestellt werden.

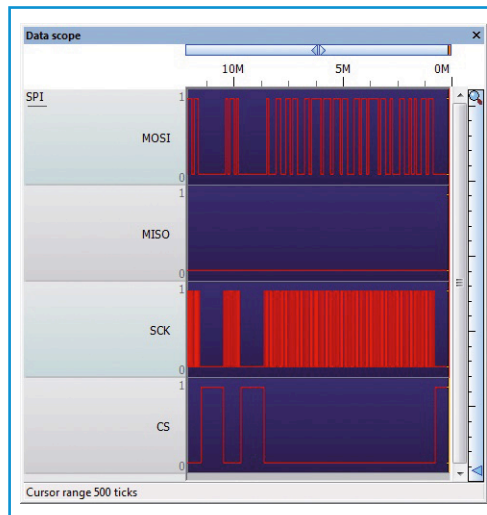


Bild 2. Die Pegel der Pins des Ziel-Chips, dargestellt mit dem Softscope.

Die PC-Seite

Auf dem PC wird Ghost durch Flowcode unterstützt, wofür gerade ein Update auf Version 6 [1] erschienen ist. Diese neue Flowcode-Version bringt nicht nur Compiler für bekannte Chips wie PIC, ARM und ATmega (Arduino) mit, sondern auch eine komplett spezifizierte Interpreter-Sprache. Die ganze Software-Suite läuft unter Windows. Um die Programmentwicklung so einfach wie möglich zu machen, basiert alles auf Flussdiagrammen.

Bezüglich der Ghost-Daten hat man die Wahl: Entweder nutzt man die Standard-Utilities oder aber man schreibt sich eigene Utilities, die diese Debug-Daten so verarbeiten, wie man dies gerne hätte. Zunächst zu den Standardfunktionen.

Utility 1: Oszilloskop

Das Tool der Wahl dürfte für die meisten Elektroniker das Oszilloskop sein. **Bild 2** zeigt

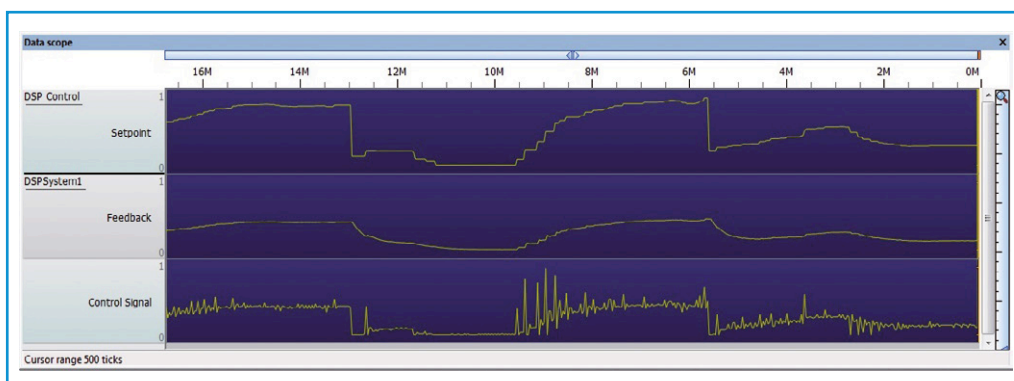


Bild 3. Analoge Ghost-Daten via Softscope.

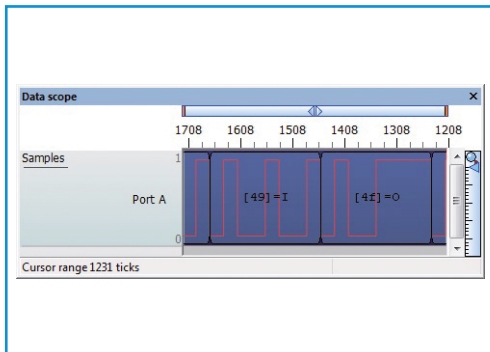


Bild 4. Anzeige der Daten im ASCII-Format.

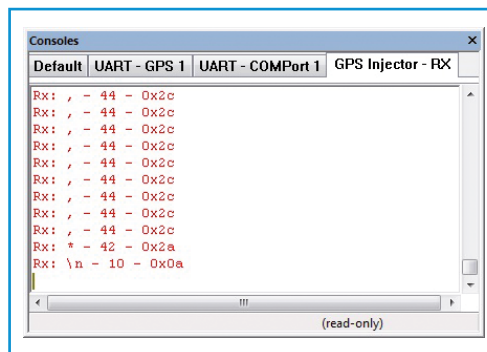


Bild 5. Low-Level-Ghost-Daten in der Konsole als scrollbarer Text.

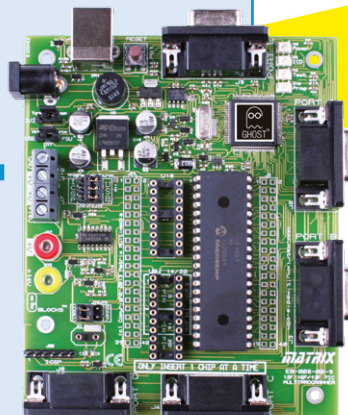
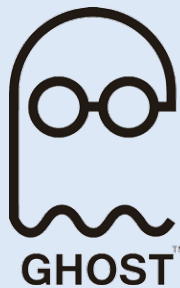
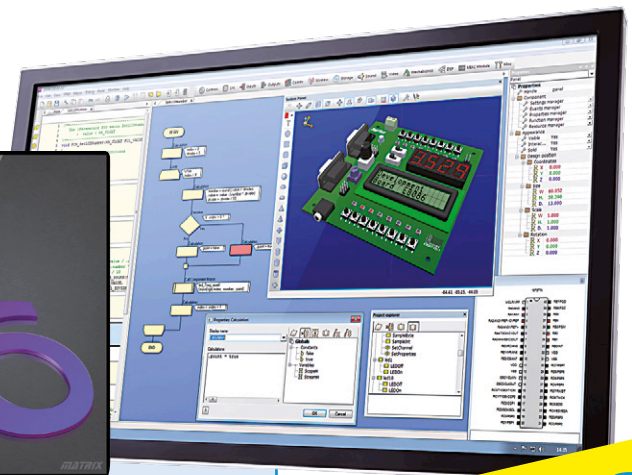
die Ausgabe des eingebauten Softscopes. Es kann mehrere analoge und digitale Kanäle gleichzeitig darstellen. Analoge Kanäle sind **Bild 3** zu sehen. Man kann so viele Kanäle wie man möchte darstellen. Zurzeit ist Ghost nicht selektiv — es werden die Signale aller Pins aufgezeichnet und man kann sie mit Hilfe der Slider oben und rechts nach Belieben

auch mit unterschiedlichen Zeitbasen inspizieren. Das Softscope ermöglicht auch die Decodierung von gängigen Protokollen und die Darstellung eines Datenaustauschs als ASCII-Zeichen (**Bild 4**). Momentan ist dies noch auf USART-Protokolle wie RS232/UART, SPI, I²C begrenzt, soll aber bald noch erweitert werden.

Spezialangebot von Elektor.POST:

Bis einschließlich des 30. Juni 2014 bekommen Elektor-Leser bei der Bestellung einer Lizenz „Flowcode Pro“ unter www.elektor.de einen kostenlosen PICmicro-Multiprogrammer dazu, der schon mit Ghost-Technik ausgestattet ist.

Wenn Sie schon eine Flowcode-Pro-Lizenz besitzen, erhalten Sie diesen PICmicro-Multiprogrammer beim Upgrade auf Flowcode Version 6 zum reduzierten Preis. Bitte beachten Sie, dass die Preise dieser Produkt-Kombis erst beim „Gang zur Kasse“ sichtbar werden. Bei Elektor wurde schon viel über diese Technologie diskutiert. Unserer Ansicht nach wäre ein Arduino-Board mit integrierter Ghost-Technik eine super Idee. Wenn Sie der gleichen Meinung sein sollten, dann lassen Sie uns das wissen. Eventuell können wir dann den Hersteller dazu überreden, so ein Board für Elektor-Leser zu entwickeln.



GRATIS

Utility 2: Konsole

Mit der „Console“ kann man die Daten an den Ziel-Pins sequenziell im ASCII-Format anzeigen (**Bild 5**). Das ist gerade dann sehr hilfreich, wenn es um Kommunikation geht. Hier operiert Ghost wirklich clever: Der Interpreter auf der PC-Seite dekodiert die Messages in einem extra Konsolenfenster und zeigt sie als systemspezifische Strings an. In **Bild 6** kann man sehen, wie die ASCII-Daten als GPS-Strings interpretiert werden. Gleichzeitig sind auch noch die anderen Fenster und Objekte aktiv, sodass man jeden Aspekt der einlaufenden Daten im Griff hat. So kann man gut überprüfen, ob empfangene und gesendete Daten auch korrekt sind.

Flowcode stellt auch Debug- und Testmöglichkeiten in Form von fertigen und maßgeschneiderten Modulen zur Verfügung, welche die Ghost-Daten in ergonomischer Art aufbereiten. **Bild 7** demonstriert die Darstellung von GPS-Daten in Form einer tachoartigen Geschwindigkeitsanzeige sowie als Ort auf einer Weltkarte. Auch ein Hardware-LC-Display wird simuliert.

Fazit

Die Mikrocontroller-Entwicklungs-Tools der nächsten Generation bringen eine Menge neuer Funktionen jenseits von einfachen Compilern mit. Vor allem der vom Anwender konfigurier- und programmierbare Interpreter auf der PC-Seite ermöglicht eine Menge leistungsfähiger Tools, welche die Entwicklung vereinfachen und dadurch beschleunigen.

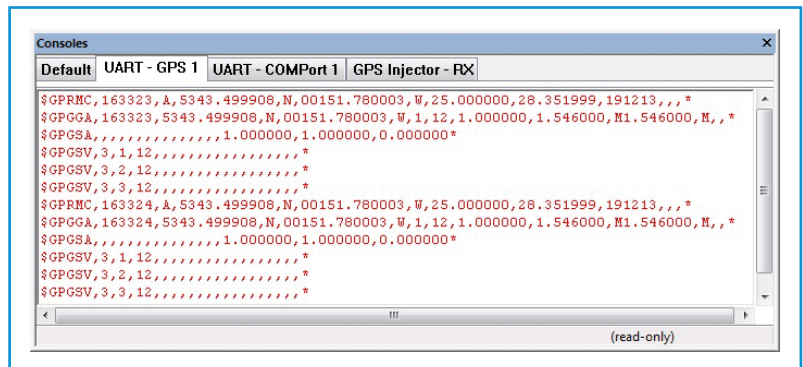


Bild 6. Konsole mit dekodierten GPS-Strings.

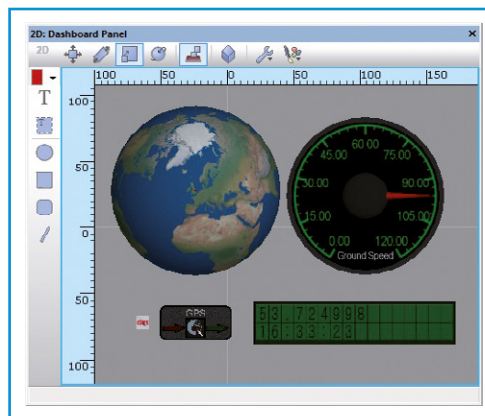


Bild 7. Ergonomisch aufbereitete Ghost-Daten dank des in Flowcode integrierten Interpreters.

Selbst mit begrenzten Programmierfähigkeiten kann man komplexere elektronische Systeme gut in den Griff bekommen.

(130582)

Weblink

- [1] www.elektor.com/development/eblocks-flowcode