

Einfach Drag & Drop

Entwicklung eines Ultraschall-Handmessgeräts. Lange Entwicklungszeiten kann sich keine Firma leisten. Außerdem müssen sich Änderungen während der Entwicklung, aber auch nach der Produktfreigabe jederzeit durchführen lassen. Dieser Praxisbericht zeigt, wie Hardwaremodule und Softwaretools nicht nur den Entwicklungsprozess unterstützen, sondern auch automatisch alle Änderungen dokumentieren.

GERHARD SCHLICHT
MARCO SCHMID

■ Ultraschallmessungen funktionieren nach dem Ortungsprinzip der Fledermäuse: Ein Schwingungspaket wird ausgesendet und die Zeit gemessen, die verstreicht, bis ein Echo zurückkommt. Dieses Verfahren lässt sich für Ortungszwecke, aber auch für die Materialanalyse nutzen. Die Signalfrequenzen liegen dabei im MHz-Bereich. Das in der Schweiz ansässige Unternehmen Sika Technology hat einen Algorithmus entwickelt, der aus dem Echo-signal die Amplitude und die genauen Antwortzeiten bestimmt. So lassen sich wichtige Informationen über den zu prüfenden Werkstoff gewinnen. Zu diesem Zweck bietet das Unternehmen ein selbst entwickeltes Handmessgerät an.

Bei der Entwicklung führte Sika Technology zunächst eine Machbarkeitsprüfung mit der grafischen Entwicklungsumgebung ›LabView‹ von National Instruments durch und ermittelte die in **Tabelle A** aufgelisteten Anforderungen an das Messgerät sowie an das Projekt. LabView sollte in allen Projektphasen mit der gleichen Entwicklungsmethode für die Softwareerstellung eingesetzt werden. Die Hardware wollte Sika nicht selbst entwickeln. Sie sollte sich im Übrigen nicht nur während der Ent-



Mit grafischem Systemdesign, leistungsfähigen Blackfin-Prozessoren und der Systemlösung ZBrain entstehen hochfunktionelle Serienprodukte in Rekordzeit

wicklung, sondern auch in der Serie aus Standardmodulen zusammensetzen, um Zeit und Kosten zu sparen. Das Entwicklungs-Know-how will Sika im Hause behalten, und last but not least soll das Design des Geräts einen hohen Wiedererkennungswert aufweisen.

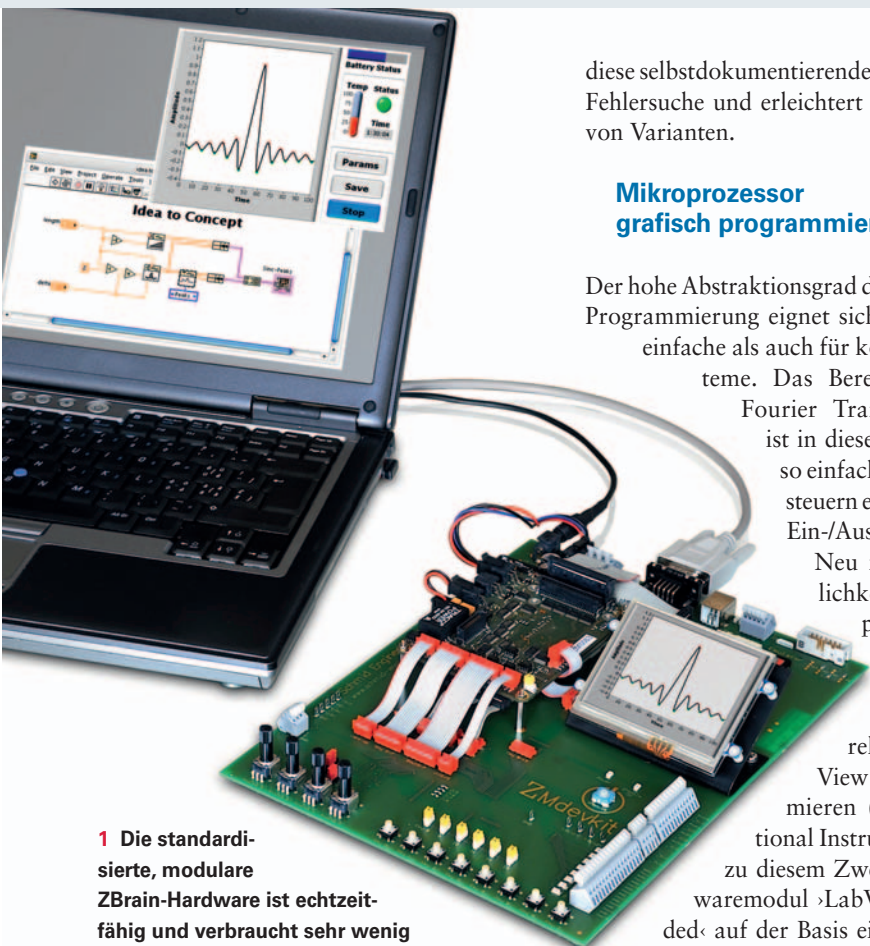
Anpassbare Funktionsblöcke beschleunigen den Prozess

Die Wahl der Hardware gestaltete sich zunächst schwieriger als erwartet, denn Untersuchungen mit PC-basierten Prozessorarchitekturen erwiesen sich als nicht zielführend.

Sika entschied sich für das ›ZBrain‹-System von Schmid Engineering. Diese Plattform basiert auf Low-power-ADI-Blackfin-Prozessoren und eignet sich speziell für Embedded-Anwendungen und batteriegestützte Systeme. Die schekkartengroßen, skalierbaren ZBrain-Hardwaremodule lassen sich sowohl grafisch mit ›LabView Embedded‹ als auch in C/C++ programmieren (**Bild 1**). Mit modernen, einfach zu handhabenden Werkzeugen und einer universellen Testumgebung eignen sie sich für folgende Entwicklungsphasen: schnelle Machbarkeitsprüfung, kostengünstige Prototypenerstellung und Serienproduktion.

KONTAKT

Schmid Engineering AG,
CH-9542 Münchwilen,
Tel. +41 71 96935-90,
www.schmid-engineering.ch
Embedded World: Halle 10, Stand 221
Halle 11, Stand 422



1 Die standardisierte, modulare ZBrain-Hardware ist echtzeitfähig und verbraucht sehr wenig Strom. Das flexible Entwicklungspaket besteht aus CPU und Prozess-E/A für Mess- und Regeltechnikaufgaben

Die Entwicklungsumgebung LabView erlaubt es dem Entwicklungsteam, sich direkt auf seine Kernkompetenzen zu konzentrieren, ohne die aufwendige Vorentwicklung von Hardware, Software und Low-Level-Treibern. Das in Gedanken bestehende Modell wird ganz einfach grafisch auf dem Bildschirm abgebildet und anschließend mit der unterlegten Hardware getestet. Mittels Drag & Drop von standardisierten Funktionsblöcken erfolgt die datenfluss- und funktionsorientierte Programmierung und führt so schnell zu verwertbaren Ergebnissen.

Unterschiedliche Funktionsblöcke, sogenannte ›Virtuelle Instrumente‹ (VIs), die auf dem Bildschirm mittels Linien verbunden werden, bilden die Basis. Diese VIs können einfache Ein- und Ausgangsfunktionen enthalten, aber auch komplexe mathematische Analyse- und Signalverarbeitungsblöcke sein. Eigene Algorithmen und C-Bibliotheken lassen sich jederzeit einbinden. Unterschiedliche Mess- beziehungsweise Regeltechnikideen oder Lösungswege können so miteinander verknüpft werden. Gleichzeitig unterstützt

diese selbstdokumentierende Methode die Fehlersuche und erleichtert das Erstellen von Varianten.

Mikroprozessor grafisch programmieren

Der hohe Abstraktionsgrad der grafischen Programmierung eignet sich sowohl für einfache als auch für komplexe Systeme. Das Berechnen einer Fourier Transformaten ist in dieser Umgebung so einfach wie das Ansteuern eines digitalen Ein-/Ausgangs.

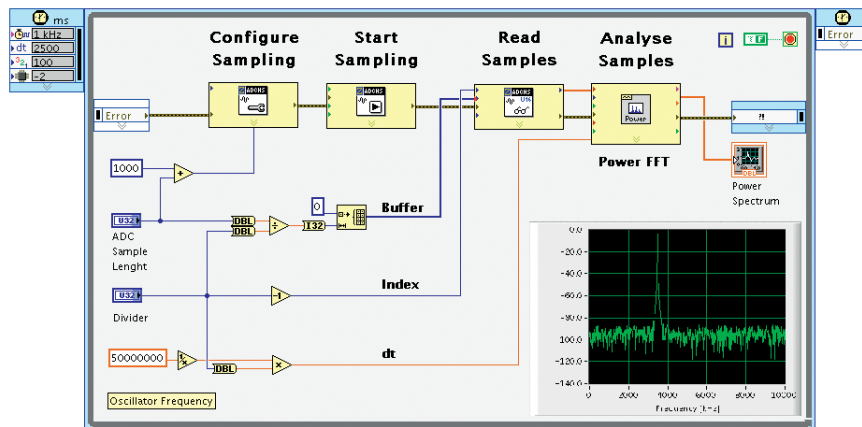
Neu ist die Möglichkeit, Mikroprozessoren wie den ADI Blackfin direkt mit LabView zu programmieren (Bild 2). National Instruments bietet zu diesem Zweck das Softwaremodul ›LabView-Embedded‹ auf der Basis eines C-Code-Generators an. Zusammen mit der entsprechenden Hardware steht dem Entwicklungsteam für alle Projektstufen eine durchgängige Entwicklungsumgebung zu Verfügung. Algorithmen oder Strategien, die in der Machbarkeitsprüfung auf dem PC entwickelt worden sind, können für Low-Power-Mikroprozessoren wiederverwendet werden. Das gilt für Prototypen ebenso wie für die Serie. Auch die ungeliebten Softwareänderungen nach dem Serienanlauf lassen sich damit einfach und transparent durchführen, sogar vor Ort.

Effizientes Team durch transparente Schnittstellen

Das Erstellen anwendungsorientierter, grafischer Bedienoberflächen für mobile Systeme erfordert normalerweise nicht nur einen hohen Programmieraufwand, sondern auch zeitraubende Abstimmungsprozesse mit der Design- und Marketingabteilung sowie den Endanwendern. Alle möchten mitreden und haben Änderungswünsche, woraus wiederum Abstimmungsbedarf entsteht.

Das ›LabView-Frontpanel-Konzept‹ für Embedded-Anwendungen unterstützt diesen Prozess mit vorgefertigten Faceplates (Bild 3). Bedieneroberflächen vom einfachen Prozessmonitor bis hin zu komplexen, mehrschichtigen Benutzerführungen lassen sich damit im Drag-&-Drop-Verfahren aufbauen. Auf der Benutzerebene können Steuerungselemente, Eingabefelder, Taster, Schieberegler und Drehknöpfe, Anzeigen, Linienplots, Tachos oder Balkendiagramme zu einer Benutzeroberfläche kombiniert werden. JPG-Bilder werden über VIs sichtbar gemacht und positioniert. Benutzereingaben werden in VIs ausgewertet und von der Programmlogik wie Events verarbeitet.

Jeder im Handheld-Projektteam hat nach einer kurzen Einführung Vorschläge am PC entworfen, realisiert und geprüft. Bereits für den Prototyp erstellte das Team anwendungsbezogene Grafikelemente mit einem gängigen Zeichnungsprogramm und lud sie mittels Entwicklungswerkzeug in das Zielsystem. Die Bedienoberfläche wurde parallel zur Hardware und zur Prozesssoftware entwickelt. Anschließend wurde der Embedded-Code generiert, der der CPU des Handhelds die gesamte Kontrolle über das GUI und den Farb-TFT mit Touchscreen übergibt. >



2 Das Gedankenmodell wird als Blockschaltbild auf dem Bildschirm grafisch entworfen. Standardwerkzeuge generieren daraus echtzeitfähigen Low-Level-Maschinencode

› Durch die transparente Darstellung und Dokumentation der Schnittstellen bei der grafischen Programmierung ließen sich die Arbeitspakete einfach aufteilen und eine hocheffiziente Teamarbeit erreichen. Messsoftware, Algorithmen und GUI wurden parallel und weitgehend unabhängig von der Zielhardware entwickelt.

Zusammen mit dem ZBrain-Entwicklungssystem stellte der Boardhersteller ein Softwareframework als Starthilfe bereit. In dieser lösungsneutralen Vorlage sind Multitasking, Echtzeitverhalten, Fehlerbehandlung, Kommunikation, Prozessein- und -ausgaben und die grafischen Bedien-schnittstellen bereits implementiert. Fehlten nur noch die Algorithmen, also das spezifische Know-how für den Handheld. Das wurde Schritt für Schritt von den Prozessfachleuten eingebettet und verifiziert.

Blockschaltbilder bieten Hardware- und Softwareentwicklern die volle Systemtransparenz. Grafische Debugginghilfsmittel erleichtern und verkürzen die Fehlersuche, besonders nach dem Kombinieren von kundenspezifischer und Standardhardware, dem Zeitpunkt, an dem das Zusammenspiel von Hard- und Softwarefunktionen auf dem Prüfstand steht. Sowohl der sequenzielle als auch der parallele Datenfluss lässt sich auf Knopfdruck visualisieren. Vom Anwender gesetzte Testpunkte zeigen die aktuellen Werte von Variablen an und Hardware-in-the-Loop (HIL) bezieht die Embedded-Hardware mit ein.

IN KÜRZE

Vom ersten Tag an produktiv

Die konsequente Nutzung aller Möglichkeiten, die die grafische Programmierung zusammen mit Off-the-Shelf Hardware bietet, führen bei der Geräteentwicklung zu einer großen Zeit- und Kostenersparnis. Entscheidend ist, dass für das Entwicklungsteam in allen Projektstufen eine durchgängige Entwicklungsumgebung bereitsteht. Die skalierbare ZBrain-Hardware wird mit lösungsneutralen Softwarevorlagen geliefert. Zusammen mit den Funktionsblöcken von LabView Embedded und Faceplates für die Bedienoberfläche schreitet die parallele Entwicklung von Software, Hardware und GUI stetig voran. Dabei sind die Schnittstellen stets für alle transparent. Trotz der marktgängigen Komponenten wurde in kürzester Zeit ein sehr spezielles Gerät für den Einsatz in einem rauen Prozessumfeld geschaffen. Dabei blieb das Schlüssel-Know-how im eigenen Haus. Bei einer konventionellen Produktentwicklung des beschriebenen Handmessgeräts hätte man von einer dreifachen Entwicklungszeit ausgehen müssen.

Zuverlässig messen in Echtzeit

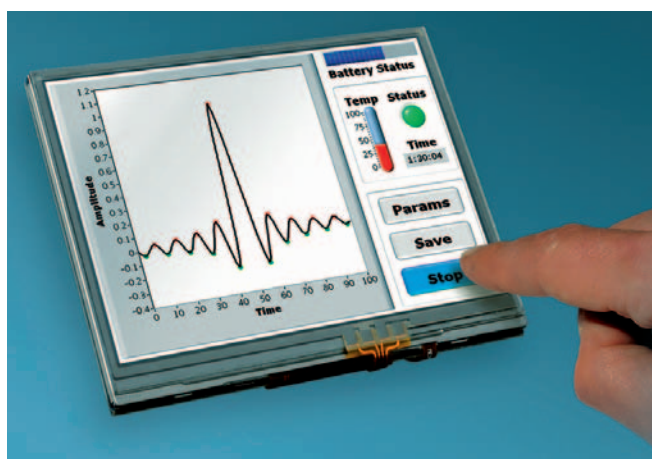
Das wichtigste Kriterium bei der Ultraschallmessung ist ein lückenloses Abtasten der Sende- und Empfangssignale. Durch ein einziges Fehlsignal würde eine Langzeitmessung ungültig. Im Zusammenspiel von GUI und Kommunikation parallel zur Datenerfassung ließ sich diese Herausforderung nur durch eine nahtlose Zusammenarbeit von Multitasking und deterministischen Echtzeitdiensten des Boardherstellers erreichen.

Auf der ZBrain-Zielhardware läuft ein echtzeitfähiger, präemptiver Multitaskingkernel von Analog Devices. Das zugehörige Board Support Package (BSP) bietet über 200 typische Embedded-Funktionen als Virtuelle Instrumente an, die per Drag & Drop in die Software integriert werden können. In wenigen Minuten wird beispielsweise mit ein paar Mausklicks eine mehrkanalige, analoge Datenerfassung mit parametrierbarer Abtastrate und Blockgröße in das Diagramm gezogen und ist somit integriert (Bild 2). Zusätzliche Funktionen im BSP sorgen für ein Echtzeitverhalten des erstellten Programms mit Zeitfenstern von Mikrosekunden auf der Kernebene.

Power Management bringt Mobilität

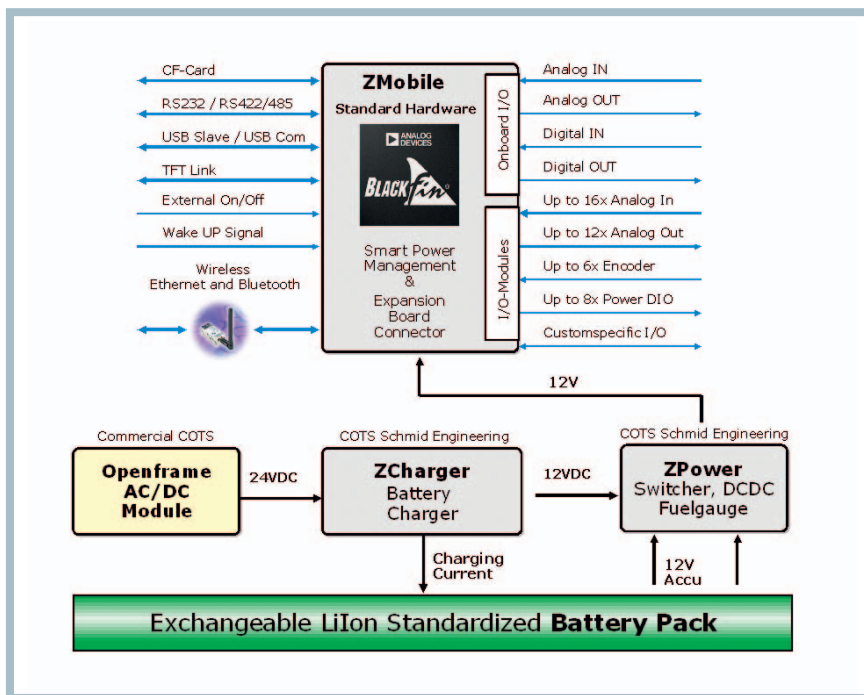
Ein niedriger Stromverbrauch ist entscheidend für den Erfolg von mobilen

Signalanalyse	Messgerät
Signalgenerierung bis 25 MHz	Stromverbrauch < 1 W
Signalerfassung bis 50 MHz	(B x L x H) 110 x 195 x 39 mm
Multiplexen der Signale für Messungen auf mehreren Kanälen	Akku- und Netzbetrieb (24 h Dauerbetrieb ohne Netz, lange Batterielaufzeiten)
Druck-, Temperatur- und Feuchteerfassung	automatisches Power-on und -off
Signalverarbeitung: 5 Blöcke (4000 Messwerte) pro Sekunde	anwendungsorientierte, ergonomische Bedienoberfläche, Touchscreen
Datenkorrelation und -analyse	Datenspeicherung auf portablem Datenträger
digitale FIR-Tiefpassfilter	Fernsteuerung durch Host via USB
digitale Ausgänge	Datenübertragung zum PC via USB und Speicherkarte



3 Interaktive grafische Bedienerschnittstellen (GUIs) werden am PC gestaltet und in das Embedded-Zielsystem geladen

A Anforderungen an Signalanalyse und Messgerät



4 Zentrale Funktionen für Messgeräte im Batteriebetrieb stehen als Referenzdesigns zur Verfügung

Geräten. Zusätzlich zu den sparsamen Blackfin-Prozessoren sind in der Zielhardware Funktionen integriert, die den Stromverbrauch so niedrig wie möglich halten: Der Stromverbrauch passt sich an die Belastung des Geräts an, das heißt, bei komplexen Berechnungen schaltet das Gerät in einen leistungsstarken Modus und kehrt danach in den Sparbetrieb zurück. Erreicht wird das entweder über einen dynamisch veränderbaren Prozessortakt oder durch Variieren der Core-Spannung. Außerdem sind Display und Messelektronik separat ein- und ausschaltbar. Wenn der Benutzer keine Eingaben macht, geht das Gerät nach einer einstellbaren Zeit in den Standby-Betrieb.

Bei Langzeitmessungen schaltet sich das Gerät nach einer definierten Zeit automatisch wieder ein oder aus. Im Netzbetrieb lädt sich gleichzeitig der Akku. Wird das Gerät vom Netz getrennt, schaltet es stoßfrei auf den Akkubetrieb um. Auch Funktionen, mit denen sich der Batteriezustand prüfen oder überwachen lässt, stehen im Board Support Package als VIs zur Verfügung und können somit einfach in die Anwendersoftware integriert oder auch modifiziert werden.

Sofort einsatzbereit

Schmid Engineering bietet verschiedene Referenzdesigns für Messgeräte auf der Basis von ZBrain an (Bild 4). Hierzu

gehören ›Measurement‹, ›Control & Motion‹, ›Sound & Vibration‹, ›High-Speed Signal Processing‹ und ›Mobile Metering‹. Mit diesen Plattformen lassen sich mobile und stationäre Geräte mit sehr unterschiedlichen Leistungsmerkmalen entwickeln. Zusammen mit LabView und mächtigen, wiederverwendbaren Softwareframeworks für typische Messaufgaben, funktionsfähigem Multitasking, Echtzeitfähigkeit, Fehlerbehandlung, Daten- und Prozessein- beziehungsweise -ausgängen ist der Anwender hiermit vom ersten Tag an produktiv. ■

Autoren

Dipl.-Ing. (FH) GERHARD SCHLICHT ist verantwortlich für den Vertrieb bei CC&I Computer Communication & Interface in Gauting bei München.
(E-Mail: g.schlicht@cciembedded.de)
Embedded World: Halle 11 / Stand 422
www.cciembedded.de

Dipl.-Ing. (FH) MARCO SCHMID, ist Geschäftsführer bei Schmid Engineering in Münchwilen (Schweiz).
(E-Mail: marco@schmid-engineering.ch)



www.mechatronik.info

Diesen Artikel finden Sie im Internet, wenn Sie im Feld ›Suche‹ die Dokumentennummer ME110013 eingeben.