

Dr. Claus Kühnel

Software:

Flexibel mit Forth

Zur Softwareproblematik in der Meßtechnik

Der Einsatz von Personal Computern hat die Meßtechnik revolutioniert. Der Meßtechniker sieht sich nicht mehr nur mit der Adaptierung des Meßobjektes sondern in gleichem Maße mit der Programmierung seiner Meßeinrichtung konfrontiert. Sich häufig ändernde Fragestellungen erfordern flexible Softwarekonzepte. Neben komfortablen und dadurch kostenintensiven

Seit dem Einzug der Computertechnik in die Meßtechnik sind zwei generelle Trends zu beobachten. Traditionelle Meßgeräte werden zur Bedienungsautomatisierung, zur Erhöhung des Funktionsumfangs u. a. m. mit „eigener Intelligenz“ ausgestattet. In das Meßgerät integrierte Computertechnik ermöglicht eine Vielzahl von Effekten, die an dieser Stelle nicht weiter erörtert werden brauchen. Es existierten eine Reihe leistungsfähiger Mikrocontroller, die ihrerseits neue, kostengünstige Systemkonzepte zulassen [1], [2]. Die Programmierung erfolgt zumeist ergebnisbezogen. Zur Entwicklung stehen Assembler, diverse Hochsprachen (PLM, C) und In-Circuit-Emulatoren zur Verfügung, die an alle gängigen Personal Computer anschließbar sind.

Der zweite Trend ist durch den direkten Einsatz von Personal Computern in der Meßtechnik gekennzeichnet. Zum einen kann der PC über standardisierte Bussysteme, wie den IEC-Bus (HP-IB, GP-IB, IEEE-488-IB) mit der Meßtechnik kommunizieren, zum anderen die Karten zur Meßwerterfassung und Steuersignalausgabe – Analog/Digital-Umsetzer (ADU), Digital/Analog-Umsetzer (DAU), Ein-Ausgabe-Baugruppe (I/O) – selbst in seinen freien Slots aufnehmen. Beide Varianten werden hardwareseitig unterstützt. Eine umfangreiche Produktpalette von (IEC-Bus-)Meßgeräten und PC-Einschubkarten steht für diese Anwendungsfälle zur Verfügung.

Nach der hardwareseitigen Adaptierung des Meßobjektes kommt das Problem der Meßwerterfassung und -verarbeitung auf den Meßtechniker zu. Oft geschieht das zu einem Zeitpunkt, zu dem noch nicht endgültig feststeht, was im einzelnen mit dem erhobenen Datenmaterial „passieren“ soll. Flexibilität ist deshalb zumin-

Softwarepaketen werden sich weiterhin „maßgeschneiderte“ Softwarelösungen behaupten müssen. Gerade in der Meßtechnik, wo der Vervielfachung der Programme aufgrund der Spezifika nahezu jeder Applikation oft enge Grenzen gesetzt sind, bietet sich die Nutzung einer Programmiersprache wie Forth vorteilhaft an.

dest zu diesem Zeitpunkt vorrangig. Für die Meßwerterfassung und -verarbeitung steht wiederum eine umfangreiche Palette an Softwareprodukten zur Verfügung. Die Entscheidung für ein solches Produkt ist abhängig von einer Reihe applikationsbedingter Größen und setzt die unbedingte Kenntnis der meßtechnischen Problematik voraus. Als Entscheidungskriterien für ein Softwareprodukt können Bedienbarkeit, Adaptierbarkeit und Verarbeitungsgeschwindigkeit angesehen werden [3].

Leicht bedienbare Systeme sind heute interaktiv, führen einen Bedienerdialog über Windows, Menüs und Icons und sind weitgehend gegen Fehleingaben gesichert. Systeme, die auf einem festen, meist durch den Nutzer selbst zu erstellenden Programm beruhen, sind im allgemeinen starr und dadurch nicht so komfortabel bedienbar. In diesen Fällen ist der gesamte Prozeß der Software-Entwicklung zu durchlaufen. Die Effektivität einer solchen Entwicklung wird dann weitgehend von den zur Verfügung stehenden Tools geprägt.

Der Adaptierbarkeit von Software zur Meßwerterfassung und -verarbeitung sollte besonderes Augenmerk geschenkt werden. Im allgemeinen ist in der Meßtechnik nicht nur eine einzige konstante Applikation vorhanden, sondern diverse. Auch bei nur einer Applikation können sich die Anforderungen im Laufe der Zeit ändern. Einfache Systeme erlauben die dann erforderlichen Anpassungen kaum oder nur mit beträchtlichen Eingriffen in Schichten der Software, die dem Anwender nicht ohne weiteres zugänglich sind. Flexible Systeme erlauben anwenderspezifische Programmänderungen und sichern damit eine ständige Adaptierbarkeit an das betreffende Problem.

Messen und Testen

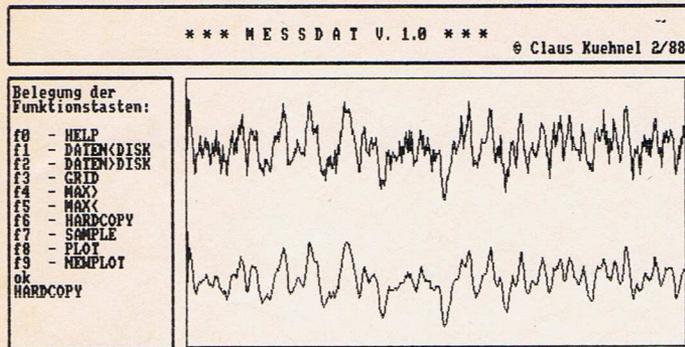


Bild 1. Darstellung von Meßwertverlauf und Funktionstastenbelegung

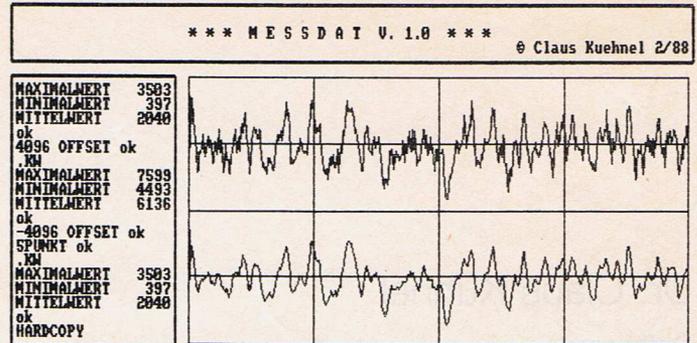


Bild 2. Darstellung von Meßwertverlauf und Bedien-Dialog

Der Begriff Verarbeitungsgeschwindigkeit kann unterschiedliche Bedeutungen haben. Zum einen versteht man darunter die Abtast- und Abspeicherungsrate (Datendurchsatz) des Systems oder aber auch die Auswerte- (Kennwertermittlung) bzw. Darstellzeiten (Datenpräsentation). Zum anderen kann damit die Reaktionszeit des Systems bei einer Echtzeitsentscheidung beschrieben werden.

Das gesamte Softwarespektrum erstreckt sich unter Verwendung der drei Kriterien Bedienbarkeit, Adaptierbarkeit und Geschwindigkeit von einfach bedienbaren, starren und langsamen bis hin zu kompliziert bedienbaren, flexiblen und schnellen Softwareprodukten.

Zur Auswahl eines die betreffende Applikation befriedigenden Softwareproduktes sind neben der Erfüllung der meßtechnischen Aufgabenstellung die erforderlichen Programmierkenntnisse, die verfügbare Programmentwicklungs- und -testzeit, die entstehenden Kosten sowie zukünftige Applikationen zu betrachten. Da der „Verschleiß“ eines verwendeten Personal Computers im allgemeinen größer als der einer umfangreicheren Meßeinrichtung ist, sollte die Applikation auch mit einem neueren PC betriebsfähig sein. Portierbare Software ist dafür eine unbedingte Voraussetzung.

Betrachtet man nun nach den beschriebenen Kriterien die im Angebot befindliche Software, kommt man mitunter recht schnell zu dem Ergebnis, daß trotz eines erheblichen Funktionsumfangs und dementsprechender Kosten eine Applikation von einem Softwareprodukt oft nicht in dem gewünschten Maße unterstützt wird und demzufolge ein weiteres Produkt hinzugezogen werden müßte. In diesen Fällen müssen dann noch die erhobenen Meßdaten in das Format der verschiedenen Analyseprogramme konvertiert werden [4].

Vor diesem Hintergrund entsteht zwangsläufig die Frage nach einer „maßgeschneiderten“ Meßtechnik-Software, die dann natürlich alle Forderungen des Meßtechnikers in geeigneter Weise erfüllt. Nicht unterschätzt werden sollte jedoch bei einer solchen Entscheidung der resultierende Programmieraufwand, da nur selten auf eine sog. „toolbox“, wie sie von Turbo-Pascal oder „C“ bekannt ist, zurückgegriffen werden kann. Geht man dennoch diesen Weg, sollte auf eine Programmier-

sprache zurückgegriffen werden, die interaktives Arbeiten und einen problemlosen Zugriff auf die Hardware-Ressourcen des Personal Computers zuläßt, eine einfache Bedienung der Schnittstellen zur Meßtechnik ermöglicht, Echtzeitanforderungen erfüllt und eine weitgehende Standardisierung aufweist, die die Portabilität der so geschaffenen Software zu sichern hilft [5].

Die angegebenen Forderungen werden in hohem Maße durch Forth sichergestellt, so daß mit diesem Software-Werkzeug ein geeignetes Mittel für die Entwicklung von Software für die Belange der Meßtechnik zur Verfügung steht. Forth-Systeme sind jedoch bei weitem nicht mehr auf Anlagen beschränkt, die mit wenig Hardware auskommen müssen [6], sondern werden für alle gängigen Personal Computer angeboten. Durch die einfache Verknüpfung von Assembler- und Hochsprachenanteilen werden an zeitkritischen Stellen die Echtzeitforderungen durch Assembler-routinen (Primitive Words) erfüllt. Desweiteren sichert die günstige Verknüpfung von Assembler und Hochsprache auch den einfachen Zugriff auf die Hardware-Ressourcen des Personal Computers. Der Nutzung von Assembler-Bibliotheken für den betreffenden Prozessor steht gleichfalls nichts im Wege. Obwohl Forth selbst ausreichend Betriebssystemeigenschaften aufweist, werden auch Systeme angeboten, die das jeweils vorhandene Betriebssystem (CP/M bzw. MS-DOS) und dessen gesamte Fähigkeiten, wie Dateiverwaltung, Datum- und Uhrzeit-Eingabe, nutzen.

Anhand des folgenden Beispiels sollen diese Aussagen untermauert werden. Für die experimentelle Analyse langsam veränderlicher Signale, wie sie beispielsweise in der Biomedizintechnik vorkommen, sollte ein einfaches Instrumentarium zur Erfassung und Auswertung von im allgemeinen als Spannungs-Zeit-Verlauf vorliegenden Signalen geschaffen werden. Als Computer fand ein preiswerter, grafik- und windowfähiger Z80-Mikrocomputer mit Diskettenlaufwerk sowie einem achtkanaligen ADU Verwendung. Das Forth-System wurde dabei so erweitert, daß der Bildschirm in drei Fenster (2 Textfenster, 1 Grafikenfenster) eingeteilt wurde. Das eine Textfenster zeigt nur den Programmtitel, wäh-

rend das zweite als Dialogfenster dient. Bei Zugriffen auf die Diskette erscheint im zweiten Fenster das komplette Directory. Der Aufruf häufig benötigter Programmfunktionen erfolgt über Funktionstasten, deren Belegung im Sinne einer Help-Funktion im Dialogfenster ausgegeben werden kann. Alle weiteren Bedieneraktionen erscheinen ebenfalls in diesem Dialogfenster und werden bei korrekter Ausführung durch das Forth-typische „ok“ quittiert. Die grafische Darstellung des Meßwerteverlaufes geschieht in einem Grafikfenster mit einer Auflösung von 464 × 300 Bildpunkten. Die Bilder 1 und 2 sollen die Gestaltung des Bildschirmes verdeutlichen. In Bild 1 wird die Belegung der Funktionstasten im Dialogfenster wiedergegeben. Im Grafikfenster ist im oberen Kurvenzug ein über den ADU erfaßtes, verrauschtes Signal wiedergegeben. Nachdem dieses Signal einem Glättungsalgorithmus unterzogen wurde, zeigt der untere Kurvenzug den resultierenden Signalverlauf. In Bild 2 hat sich nur der Inhalt des Dialogfensters verändert. Es werden die vorgenommenen Manipulationen am erfaßten Meßwerteverlauf sichtbar. Das sind im einzelnen die Veränderung des Offsets, der Aufruf des Glättungsalgorithmus, die grafische Ausgabe des so behandelten Meßwerteverlaufes einschließlich der Ermittlung von Maximal- und Minimalwert sowie das Einschalten eines Gitters (Grid) über die Darstellung.

Mit einer ebenfalls implementierten Hardcopy-Funktion kann der gewünschte Bildschirminhalt über einen Epson-kompatiblen Matrixdrucker „zu Papier“ gebracht werden.

Zur Zeit soll der Umfang der zur Verfügung stehenden Algorithmen um eine schnelle Fouriertransformation (FFT) ergänzt werden, die dann Untersuchungen im Spektralbereich zuläßt. Da Forth ursprünglich nur eine Integer-Arithmetik bereitstellt, sind bestimmten Signalverarbeitungs-Algorithmen gewisse Beschränkungen auferlegt. Bei der softwaremäßigen Realisierung einer Gleitkomma-Arithmetik muß in jeder Programmiersprache eine entsprechende Verarbeitungszeit in Kauf genommen werden. Wo dies unzumutbar erscheint, ist der Einsatz von Arithmetikprozessoren angezeigt. Für die 16-Bit-Personal-Computer ist das durch den Einsatz des 8087 kein Problem, wobei es aber für die 8-Bit-Computer mit einem AMD 9511 Lösungsmöglichkeiten gibt.

Zwei Listings (Bilder 3 und 4) sollen abschließend den einfachen Zu-

griff auf Hardware-Ressourcen und Analog/Digital-Umsetzer (ADU) verdeutlichen. Zum Einrichten des Grafikfensters wurden Firmware-Routinen des Rechners verwendet. Wie in Bild 3 zu sehen ist, wurden die Adressen der Routinen zum Setzen und Abfragen der Fensterbreite und -höhe als Konstanten definiert. Zur Parameterübergabe dienen die mittels CODE eingeleiteten Assembler-routinen. Da die Grenzen des Grafikfensters intern auf Byte-Grenzen festgelegt werden, müssen die resultierenden Grenzen nach dem Setzen abgefragt werden. Die Umrandung wird schließlich als Colon-Definition ausgeführt. Wie Screen # 8 schließlich zeigt, ist die Definition des Grafikfensters mit den zwei Angaben

168 631 SETWIDTH 11 310 SETHEIGHT erledigt. Mit den vorstehenden Angaben erstreckt sich das Grafikfenster über eine Breite von 464 Pixel, begin-

```

Screen # 6
( WINDOWS: GRAFIK1                                CK 21/12/87)

HEX

BBCF CONSTANT ADDRSETWIDTH
BBD2 CONSTANT ADDRSETHEIGHT
BBD5 CONSTANT ADDRGETWIDTH
BBD8 CONSTANT ADDRGETHEIGHT

DECIMAL

CODE SETWIDTH
  D POP  H POP  B PUSH  ADDRSETWIDTH CALL  B POP  NEXT

CODE SETHEIGHT
  D POP  H POP  B PUSH  ADDRSETHEIGHT CALL  B POP  NEXT

Screen # 7
( WINDOWS: GRAFIK2                                CK 21/12/87)

CODE GETWIDTH
  B PUSH  ADDRGETWIDTH CALL  B POP  D PUSH  H PUSH  NEXT

CODE GETHEIGHT
  B PUSH  ADDRGETHEIGHT CALL  B POP  H PUSH  D PUSH  NEXT

: GRFRAME
  GETWIDTH  GETHEIGHT  3 PICK  2 PICK  MOVE
  2 PICK  2 PICK  DRAW  2 PICK  OVER  DRAW
  3 PICK  OVER  DRAW  3 PICK  2 PICK  DRAW
  2DROP  2DROP ;

Screen # 8
( WINDOWS: DEMO                                    CK 18/12/87)

1 0 0 39 20 KOORD WINDOW1
2 0 21 79 24 KOORD WINDOW2

320 630 SETWIDTH
70 390 SETHEIGHT

: NOISE 641 320 DO I 350 RANDOM 50 + DRAW 2 +LOOP ;
: NONSENS 1000 0 DO 256 RANDOM EMIT LOOP ;

: DEMO
  WINDOW1 FRAME SELECT CLS NONSENS CR ." WINDOW 1"
  WINDOW2 FRAME SELECT CLS NONSENS CR ." WINDOW 2"
  GRCLS GRFRAME NOISE 1 STREAM ;
    
```

Bild 3. Listing zum Einrichten des Grafikfensters

```

Screen # 10
( MESSDAT: MS ADU SAMPLE DVM                                CK 22/ 2/88)

HEX F8F0 CONSTANT PORT 0000 CONSTANT CLEAR
0008 CONSTANT CONV 0000 CONSTANT KANAL DECIMAL

: ADUBLANK KANAL CLEAR OR PORT P! ;
: ADUCONV KANAL CONV OR PORT P! ;
: ADUREAD PORT P@ ;
: NORM 10000 255 */ ;
: ZEIGE 1 17 GOTOXY 10 .R ." mV" ;
: 100US 0 DO NOOP LOOP ;
: MS 0 DO 10 100US LOOP ;
: ADU ADUBLANK ADUCONV 1 100US ADUREAD ;
: SAMPLE CLS LAENGE 1+ 1 DO ADU NORM I VERLAUF ! LOOP ;
: DVM CLS BEGIN ADU NORM ZEIGE ?TERMINAL UNTIL CLS ;

Screen # 11
( MESSDAT: %ADU %SAMPLE                                       CK 6/ 3/88)

CODE %ADU
B D MOV C E MOV B POP H POP
A XRA L ORA A OUT(C)
B A MVI L ORA A OUT(C)
NOP NOP NOP NOP NOP NOP NOP NOP NOP NOP
NOP NOP NOP NOP NOP NOP NOP NOP NOP NOP
C IN(C) 0 B MVI B PUSH
D B MOV E C MOV NEXT

: %SAMPLE CLS KANAL PORT
LAENGE 1+ 1 DO 2DUP %ADU I VERLAUF ! LOOP 2DROP ;
    
```

Bild 4. Listing zur Arbeit mit dem Analog/Digital-Umsetzer

nend an der Position 168, und eine Höhe von 300 Pixel, beginnend an der Position 11.

In der gleichen Weise wird die Bedienung der ADU-Erweiterungskarte vorgenommen. Bild 4 zeigt in Screen # 10 das Ansprechen des ADU in einer Hochsprachen-Definition (ADU), während in Screen # 11 das gleiche Problem durch eine kleine Assembleroutine (%ADU) gelöst wird. Die Hardware des ADU wird durch die niederwertigen vier Bits des Datenwortes gesteuert. Die Bits 0...3 geben die Kanaladresse des Multiplexers an, während Bit 4 = L den ADU löscht und Bit 4 = H den Umsetzungsvorgang startet. Die ADU-Erweiterungskarte wird am Port mit der Adresse F8F0_H betrieben. Diese Größen werden wiederum als Konstanten definiert. Die Steuerung des ADU erfolgt nun einfach durch die Bereitstellung des betreffenden Datenwortes (Kanaladresse OR ADU-Status) und der Portadresse. Das Einlesen des Ergebnisses der Umsetzung benötigt nur die Portadresse. Diese Bemerkungen sollen zur Charakterisierung des Problems ausreichen.



Dr.-Ing. Claus Kühnel wurde in der Oberlausitz geboren. Nach dem Studium der Informationstechnik an der TU Dresden arbeitete er als Schaltungsentwickler in der Industrie. Diese Tätigkeit und sich anschließende Erfahrungen auf dem Gebiet der rechnergestützten Meßtechnik waren die Grundlage für eine freiberufliche Tätigkeit. Neben konventionellen PC-Anwendungen stehen derzeit die Verbreitung der FORTH-Philosophie sowie betreffende Meßtechnik-Applikationen an vorrangiger Stelle seiner Arbeit.

Ändert sich nun aus den unterschiedlichsten Gründen eine Komponente des Meßsystems, dann ist der Arbeitsumfang zur Adaptierung recht gering. Ist es beispielsweise erforderlich, einen ADU mit größerer Auflösung oder kürzerer Umsetzzeit im System einzusetzen, dann ist das Forth-Wort ADU (inkl. seiner Bestandteile ADUBLANK, ADUCONV und ADUREAD) zu verändern, und das System arbeitet schließlich mit der neuen ADU-Erweiterungskarte, so wie bislang mit der alten. Gleiches ist aufgrund der weitgehenden Standardisierung moderner Forth-Versionen (Standard Forth-83) und der damit prinzipiell gegebenen Portabilität der Software beim Einsatz neuer Personal Computer zu erhoffen.

Das gesamte hier vorgestellte System umfaßt das Forth-System selbst, einen Wordstar-kompatiblen Full-Screen-Editor, einen Z80-Assembler sowie den Applikationsteil und hat einen Umfang von insgesamt ca. 22- KByte, wobei auf den Applikationsteil nur ca. 4 KByte entfallen.

Literatur

- [1] Johannis, R.; Feger, O.: Peripheriebausteine integriert – Detailapplikation zum SAB 80515. ELEKTRONIK 1987, H. 1, S. 59...64.
- [2] Baum, E.: Neuer CMOS-Mikrocontroller. Elektroniker 1988, H. 3, S. 89...96.
- [3] Comparing Data Acquisition Software for Personal Computers. Application Note # 804, Keithley Instruments.
- [4] Formatting Instrumentation Data for LOTUS 1-2-3, ASYST™, and DADiSP II. Application Note # 803, Keithley Instruments.
- [5] Kühnel, C.: Eignung verschiedener Hochsprachen für die Bearbeitung von Meßdaten. Elektroniker 1988, H. 3, S. 97...100.
- [6] Bergmann, E.; Gottschalk, K.: Einfaches Hochsprachen-Echtzeitbetriebssystem für Minimalsysteme. ELEKTRONIK 1986, H. 22, S. 121...130.

Postkarte genügt,

um uns Ihre Meinung zu den Beiträgen dieses Heftes mitzuteilen. Nehmen Sie beispielsweise die am Ende des Heftes beigefügte Redaktions-Rücklaufkarte.