

Version 01/04



D C-CONTROL 1 MICRO / MICRO-PCB Programmer/Evaluation Board

Seite 3 - 41

GB C-CONTROL 1 MICRO / MICRO-PCB Programmer/Evaluation Board

Page 42 - 78

Best.-Nr. / Item-No.:

19 83 02

100 %
Recycling-
Papier.

Chlorfrei
gebleicht.



100 %
recycling
paper.

Bleached with-
out
chlorine.

D Impressum

Diese Bedienungsanleitung ist eine Publikation der Conrad Electronic GmbH. Alle Rechte einschließlich Übersetzung vorbehalten. Reproduktionen jeder Art, z. B. Fotokopie, Mikroverfilmung, oder die Erfassung in elektronischen Datenverarbeitungsanlagen, bedürfen der schriftlichen Genehmigung des Herausgebers. Nachdruck, auch auszugsweise, verboten. Diese Bedienungsanleitung entspricht dem technischen Stand bei Drucklegung. Änderung in Technik und Ausstattung vorbehalten.

© Copyright 2004 by Conrad Electronic GmbH. Printed in Germany.

GB Imprint

These operating instructions are published by Conrad Electronic GmbH, Klaus-Conrad-Str. 1, 92240 Hirschau/Germany. No reproduction (including translation) is permitted in whole or part e.g. photocopy, microfilming or storage in electronic data processing equipment, without the express written consent of the publisher. The operating instructions reflect the current technical specifications at time of print. We reserve the right to change the technical or physical specifications.

© Copyright 2004 by Conrad Electronic GmbH. Printed in Germany.

*01-04/AH

Technical Data

C-CONTROL MICRO/MICRO-PCB

Operating voltage	5V +- 10% (8 - 12V at sep. supply MICRO-PCB)
Current input (ports input)	6mA
max. LO inlet level ports	0.3 x Vdd
min. HI inlet level ports	0.7 x Vdd
min HI outlet level ports P3,P5 / P1,P2,P4,P6	Vdd - 1.5 / Vdd - 0.8 (at -10 mA)
max. LO outlet level ports P3,P5 / P1,P2,P4,P6	1.5V / 0.8V (at 10 mA)
max. leakage current port Input	1uA
Basic work speed	30000 instructions per second
Internal bus cycle	3.1 Mhz

C-CONTROL MICRO programmer/evaluation board

Operating voltage	8 - 12V battery or plug power supply unit
Max current input	50mA
Baud rate of serial interface	max. 9600 Baud
Switching capacity, slide switch	max. 200 mA

ⓓ Wichtig! Unbedingt lesen!

Bevor Sie das Programmer/Evaluation Board für die C-Control MICRO/MICRO-PCB oder angeschlossene Geräte in Betrieb nehmen, lesen Sie bitte diese Anleitung vollständig durch! Sie erläutert Ihnen die korrekte Verwendung und weist auf mögliche Gefahren hin.

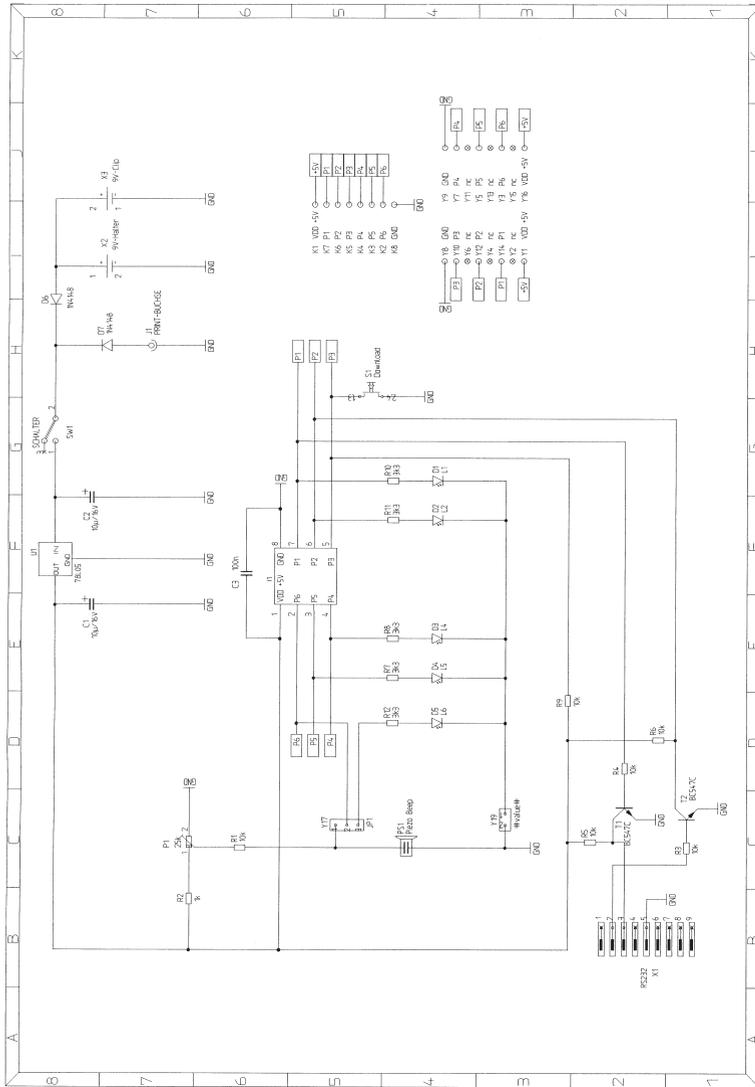
Für Schäden, die aus der Nichtbeachtung dieser Anleitung resultieren, besteht keinerlei Garantieanspruch und übernimmt Conrad Electronic keine Haftung

Inhalt

Wichtig! Unbedingt lesen!.....	3
Inhalt	3
Einleitung.....	5
Garantie	5
Service.....	6
Produktbeschreibung.....	7
Bestimmungsgemäße Verwendung.....	7
Sicherheitshinweise	7
Leistungsmerkmale.....	8
Handhabung	10
Programmer/Evaluation Board	10
Anschluss der C-Control Computer.....	10
Verbindung zum PC	10
Stromversorgung.....	10
Bedienelemente	10
Anzeigeelemente	11
Jumper zur Konfiguration.....	11
C-Control Computer MICRO / MICRO PCB.....	12
Erste Inbetriebnahme.....	13
Software-Installation	13
Die Programmiersprache CCBASIC	14
Das Konzept der C-Control MICRO/MICRO PCB.....	14
Systemressourcen des C-Control Computers	15
Echtzeituhr (RTC)	15
User-Bytes	15
Digitalports	16
Analogports.....	16
Frequenz Eingang	17
Frequenz Ausgang - BEEP.....	17

The programs can be loaded but nothing happens on "start"

- You did not configure the P/E board correctly (jumper)
- You have an application in which PORT3 does not lie on log high when it is switched on
- Your application needs too much power and overloads the voltage controller or battery



Einleitung

Wir danken Ihnen für Ihre Entscheidung zum Erwerb des Programmer/Evaluation Boards für die C-Control MICRO/MICRO-PCB.

Sie können damit beide Ausführungen der C-Control MICRO in der Entwicklungsumgebung programmieren

und die Programme auf die C-Control MICRO / MICRO PCB laden.

Das Programmer/Evaluation Board (P/E-Board) für die C-Control MICRO/MICRO-PCB stellt Ihnen außerdem eine Reihe von Ein/Ausgabemöglichkeiten (LEDs, Beeper, Taster, Poti) zur Verfügung damit sie die Beispielprogramme oder eigene Anwendungen schnell und bequem ausprobieren können.

Conrad Electronic GmbH
D-92240 Hirschau

Garantie

Bevor Sie das P/E-Board oder angeschlossene Geräte in Betrieb nehmen, lesen Sie bitte diese Anleitung vollständig durch. Sie erläutert Ihnen die korrekte Verwendung und weist auf mögliche Gefahren hin.

Für Schäden, die aus der Nichtbeachtung der Bedienungsanleitung resultieren, besteht keinerlei Garantieanspruch und übernehmen wir keine Haftung!

Hinweise zur beschränkten Garantie und Haftung

Bei der C-Control MICRO handelt es sich um einen BASIC Steuercomputer. Die im Mikroprozessor als ROM-Maske integrierte und die zugehörigen PC-Software wird in vorliegender Form geliefert.

Conrad Electronic übernimmt keine Garantie dafür, daß die Leistungsmerkmale individuellen Ansprüchen entsprechen, oder daß die Software im Mikroprozessor und die PC-Software in jedem Fall unterbrechungs- und fehlerfrei arbeiten. Der Anwender trägt das gesamte Risiko bezüglich der Qualität und der Leistungsfähigkeit des Gerätes inklusive aller Software.

Conrad Electronic garantiert die Funktion der mitgelieferten Applikationsbeispiele unter Einhaltung der in den technischen Daten spezifizierten Bedingungen. Sollte sich das P/E-Board, die C-Control MICRO oder die PC-Software darüber hinaus als fehlerhaft oder unzureichend erweisen, so übernimmt der Kunde alle entstehenden Kosten für Service, Reparatur oder Korrektur.

Die Gewährleistung von Conrad Electronic beschränkt sich ausschließlich auf den Austausch des Gerätes innerhalb der Garantiezeit bei offensichtlichen Defekten an der Hardware, wie mechanischer Beschädigung, fehlender oder falscher Bestückung elektronischer Bauteile, ausgenommen gesockelter integrierter Schalt-

PEAK TIME:

Writing the system variable `FREQ` sets the peak time (20 ms steps). `PORT4` has to be an input for this function.

The peak time is the time during which impulses are counted at the input. The peak time is stated in units of 20ms. Do not use small peak times for high frequencies.

At a peak time of 20ms, a counter status results, for example of 10Khz, in `FREQ` of: $0.02 * 10000 = 200$.

At greater peak times, lower frequencies can also be measured quite precisely. A new peak time is only effective after expiry of the previous one.

define `FREQPORT p[4]`

`FREQ=5` sets the peak time to 100ms

COUNTER STATUS

`x = FREQ`

By reading the variable `FREQ`, the counter status reached during the last peak time is supplied.

It corresponds to the frequency:

$FREQUENZ = x / TORZEIT$

FREQUENCY RANGE:

As the system variable `FREQ` only has 1 byte, only 255 occurrences can be counted. Therefore, adapt the peak time to the frequency which you wish to process. The smallest peak time is 1 (i.e. 20ms) and processes frequencies of up to 12 kHz. If you work with unknown frequencies, you can program an automatic peak-time setting. This begins with the smallest value and enlarges the peak time until a count result worth mentioning is registered in `FREQ`. The current peak time together with `FREQ` then gives precise information about the frequency measured.

RESTRICTIONS

During `PAUSE` and `BEEP`, counting is interrupted but is continued regularly after expiry of the time.

Schweiz:

Tel. 0848/80 12 88

Fax 0848/80 12 89

e-mail: support@conrad.ch

Mo. - Fr. 8.00 bis 12.00 Uhr, 13.00 bis 17.00 Uhr

Produktbeschreibung

Bestimmungsgemäße Verwendung

Der C-Control BASIC Steuercomputer dient zur programmierbaren Ansteuerung elektrischer und elektronischer Geräte, die mit Schutzkleinspannung betrieben werden. Diese Geräte können in beliebige technische Systeme integriert werden, die nicht direkt oder indirekt medizinischen, gesundheits- oder lebenssichernden Zwecken dienen oder durch deren Betrieb Gefahren für Personen oder Sachwerte entstehen können.

Zur Programmierung des Computers ist ausschließlich die mitgelieferte PC-Software und das P/E Board zu verwenden.

Sicherheitshinweise

Lesen Sie diesen Abschnitt besonders aufmerksam durch! Bei Nichtbeachtung der Sicherheitshinweise besteht Lebensgefahr durch einen Stromschlag oder Elektrobrand!

Der C-Control/BASIC -Steuercomputer MICRO/MICRO-PCP sowie das P/E Board wurden gemäß den geltenden gesetzlichen Vorschriften einer Sicherheitsprüfung unterzogen und entsprechend zertifiziert (CE). Bei sachgemäßem Gebrauch gehen normalerweise keine Gesundheitsgefährdungen von diesen Geräten aus.

Der C-Control/BASIC -Steuercomputer MICRO/MICRO-PCP sowie das P/E Board sind als elektronisches Gerät mit der dafür üblichen Vorsicht und Sorgfalt zu behandeln. Die Mißachtung der aufgeführten Hinweise oder eine andere als die bestimmungsgemäße Verwendung kann zur Beschädigung oder Zerstörung des Steuercomputers oder angeschlossener Geräte führen.

Die Geräte sind nicht gegen Lichtbogenüberschläge geschützt und dürfen nicht in Starkstromindustrieanlagen verwendet werden. Die maximalen Eingangsgrößen gemäß den Spezifikationen in den Technischen Daten dürfen nicht überschritten werden. Die Geräte sind nicht in Räumen oder Umgebungen einzusetzen, in denen brennbare oder ätzende Gase, Dämpfe oder Stäube vorhanden sind oder vorhanden sein können.

MINUTE (0...59)

SECOND (0...59)

Please observe the fact that the internal clock continues to run during access. The second value should therefore always be read first. If it shows 59 it is necessary that after reading the last interesting time information (e.g. HOUR) the second value is read again and test for =0. In this case, the real-time clock has to be read again as a new minute has been started.

Further-reaching time information is not available in the C-control MICRO but can if necessary be generated by means of a small sub-program-

EXACTITUDE:

The RTC of the C-CONTROL MICRO is fed from a free-running oscillator which in principle cannot fulfil the demands on a precise clock. The manufacturer states exactitude of 25 %. So that the RTC remains usable nevertheless, you have the possibility of calibrating the clock's cycle.

The tool CLOCK_CHECK.BAS will help you in calibrating and will also give you further tips.

FREQ2=value

writes a correction factor into the operating system which corrects the system cycle by 0.2% per digit (starting from the default value 128 after RESET). This allows the cycle to be set to an exactitude of 0.2% which corresponds to an error in the RTC of 170s/24h.

The content of FREQ2 cannot be read and an instruction such as for example

```
a=FREQ2
```

leads to uncontrollable erroneous behaviour as the corresponding TOKEN of INTERPRETER of the MICRO is not recognised.

Note:

YEAR

MONTH

DAY

DOW

are variables which are reserved for the operating system but may not be used for MICRO.

- 8 BIT A/D-Wandler
- Digitale Ports I/O
- Beep Ports
- Freq. Eingang
- Serielle. Schnittstelle 9600/19200 Baud
- 2kB BASIC-Programmspeicher
- ladbare Systemerweiterungen (z.B. DCF 77 Synchronisierung der RTC)

Für maximale Flexibilität sind Betriebssystemerweiterungen ladbar, die allerdings den nutzbaren BASIC-Programmspeicher reduzieren. Die Betriebssystemerweiterungen werden von Conrad zur Verfügung gestellt.

Programmer/Evaluation Board

Das P/E Board stellt dem Benutzer eine serielle Schnittstelle mit Pegelwandlung zur Verfügung um kompilierte Programme aus der Entwicklungsumgebung in die C-Control Computer der Ausführung MICRO und MICRO zu laden.

Zusätzlich gibt es eine Reihe von Ein/Ausgabemöglichkeiten (LEDs, Beeper, Taster, Potentiometer) zur Verfügung damit sie die Beispielprogramme oder eigene Anwendungen schnell und bequem ausprobieren können.

Die Stromversorgung erfolgt mit einer 9V Batterie oder einem Steckernetzteil 8 bis 12V

Achtung!

Das P/E-Board unterstützt 19200 Baud aus technischen Gründen nicht (Zeitkonstanten im Pegelwandler)

To switch the port over (ON to OFF; OFF to ON), you can write

P = NOT P

or the command

TOG P

used. TOG stands in English for "toggle". The TOG command requires less space in the EEPROM and is executed more quickly than the classical NOT-P construction. The port variable P may only stand in the TOG command for a single digital port.

• Deactivation of a ports with DEACT

As soon as a port variable is assigned a value for the first time, the control computer switches the associated hardware structures in the processor chip (transistors) to output operation. That means current flows out of or into the processor (max. 10 mA permitted!) in accordance with the switching connected. The command

DEACT portvar

deactivates the port given. That means, the port is switched to a high-resistance condition and works in the input area. The DEACT command may be used on single digital ports or byte ports.

Definition and applications of data tables

In standard BASIC, DATA lines serve for storing constant data blocks which can then be accessed sequentially. CCBASIC does not support any DATA lines but offers a far more flexible tool for the definition and access to data blocks. Constant data can be stored in the form of tables. Each table is assigned an identifier (table name) and can contain any desired number of entries as long as the program memory provides room. Each data entry (Cx) is stored as an integer value and thus occupies two bytes. The data can be listed directly in the source text thereby but may only accept values of up to a maximum of 255 as tables can only be processed by the byte

TABLE tablename

CO CI C2 C3 C4 C5 . . -.. . Cn

TABEND

or be imported from an external text file by the CCBASIC compiler.

TABLE tablename "tabfilename"

ACHTUNG:

Verwenden Sie Port 3 niemals in Anwendungen bei welchen nicht sichergestellt ist, dass Port 3 beim Anlegen der Betriebsspannung log. high ist, da sonst der Download-Modus aktiv ist und das Anwenderprogramm nicht läuft.

Anzeigeelemente

Im Anwenderprogramm können sie die vorhandenen Anzeigeelemente beliebig nutzen.

Die LEDs sind entsprechend Ihrer jeweiligen Bezeichnung an die zugehörigen Ports angeschlossen,

d.h. L3 ist an Port 3 angeschlossen.

Wenn die Ports 1 u. 2 in Ihrem Programm nicht aktiviert sind (INPUT), oder die MICRO auf einen Download wartet, leuchten die zugehörigen LEDs leicht, da sie über den RS 232 Pegelwandler mit Strom versorgt werden.

Zum Ausprobieren der AD-Wandler ist ein Potentiometer vorhanden, das eine variable Spannung an den AD6

liefert. Die Spannung wird größer, wenn Sie das Potentiometer gegen den Uhrzeigersinn drehen. Ein Beeper ermöglicht akustische Signale.

Jumper zur Konfiguration

Damit sie nicht an die vorhandenen Anzeigeelemente gebunden sind, können diese abgetrennt werden.

Sie können dann eine eigene Applikation an den Pfostenstecker anschließen.

JUMPER JP1

JP1 ermöglicht den Betrieb des Beepers zusammen mit der variablen Eingangsspannung an Port6/AD6

Eine Beispielprogramm, das in das Verzeichnis "BASIC" installiert wird, zeigt Ihnen wie Ports während eines Programmablaufs mit unterschiedlichen Eigenschaften genutzt werden können.

Für den Betrieb des Beepers zusammen mit der variablen Eingangsspannung an Port6 muss der Jumper auf die Position 1-2 gesteckt sein.

Wollen Sie darauf verzichten und lieber eine zusätzliche LED zur Verfügung haben, so stecken Sie den Jumper auf Position 2-3

JUMPER JP2

Wollen Sie gänzlich auf die vorhandenen Ausgabemöglichkeiten verzichten, weil Sie Ihre Applikation am Pfostenstecker angeschlossen haben, zu ziehen sie beide Jumper JP2 und JP1 ab.

Define ADPORT1 AD[1]
Define TRANSMIT PORT[1]

-
-
-

deact TRANSMIT
a=ADPORT1

deactivates the function TXD of the ports and activates it as an analogue input.
The analogues value is written in variable a.

• Data output

Data output takes place as text through the serial interface of the C-control/BASIC control computer. If a PC with a terminal program is connected by means of an interface cable, for example, the data output can be displayed there.

PRINT term

outputs the result of the calculation of term.

PRINT "text"

transmits the text contained within the exclamation marks. In both cases, a new-line character is attached to the transmission which prompts the terminal program to carry out the next output in the next screen line. The line-feed can be suppressed if a semi-colon is added to the PRINT command after the parameter (term or "text").

PRINT term;

or:

PRINT "text";

CCBASIC also supports several outputs with a PRINT command whereby the individual parameters are separated by commas or semi-colons. A comma will insert a tabulator sign into the output which, in accordance with the settings in the terminal program, appears as a number of blanks on the screen. If two outputs follow each other without any blanks, these are to be separated in the PRINT command by means of a semi-colon.

PRINT "a= ", a
PRINT "a= "; a

A single PRINT command without parameter outputs only a line-feed.

Achtung:

Ein Verstoß gegen diese Richtlinien insbesondere betreffend der max. Stromaufnahme führt zur Zerstörung empfindlicher Komponenten auf der Leiterplatte.

Die Anschlussdrähte der C-Control Micro haben folgende Funktionen:

ROT -	VDD +5V	SCHWARZ -	GND
GRÜN -	P6/AD6/BEEP	VIOLETT -	P1/AD1/BEEP/TXD
BRAUN -	P5/AD5/BEEP	BLAU -	P2/AD2/BEEP/RXD
GRAU -	P4/BEEP/FREQ	WEISS -	P3 INPUT

Erste Inbetriebnahme

Software-Installation

Legen Sie die mitgelieferte CD in das Laufwerk an Ihrem PC. Nach wenigen Sekunden erscheint selbstständig das Auswahlmenü, das Ihnen verschiedene Installationen anbietet.

1) Wählen sie zunächst > C-CONTROL I

Sie Befinden Sich jetzt in der Auswahl zu Installationen der C-Control I Geräten.

2) Wählen Sie hier C-CONTROL/BASIC – MICRO

3) Das erscheinende Fenster bietet die Option "ÖFFNEN"

Wählen Sie diese Option um die Installation zu beginnen

4) Nach Abschluss der Installation wird die Entwicklungsumgebung geöffnet.

Ein erstes Programm laden und starten

P/E Board betriebsbereit machen

Stellen Sie die Betriebsbereitschaft des P/E Boards und der C-Control MICRO, wie im Kapitel Handhabung beschrieben her.

Vergewissern Sie sich dass die Jumper gesteckt sind (JP1 auf Pos. 1-2) und der Schiebeschalter auf OFF steht.

Vergewissern Sie sich dass die Entwicklungsumgebung die richtige Schnittstelle benutzt (die Auswahl erfolgt über das Menü OPTIONEN > UMGEBUNG > LADER

Programm in die Entwicklungsumgebung laden

Wählen Sie "DATEI ÖFFNEN" in der Menüleiste. Öffnen Sie die Datei "1_EINFÜHRUNG_PORTS1.BAS" im Ordner PROGRAMME/CC1MICRO/BASIC

construction must be contained in a source text line. Instruction blocks (several instructions) after THEN are not permitted.

- **Branch instruction**

GOTO label

With the GOTO instruction, the control computer can be prompted to continue program processing at a certain point. A label identifier is stated as the branch target. The branch target can be positioned before or after the GOTO instruction in the source text.

- **Call and return from a sub-routine**

Calling-up of a sub-routine takes place by means of the instruction

GOSUB label

The label is thereby the starting point of the sub-routine. In so-called sub-routines, program sections are pooled which are required more than once in the course of processing the program. A sub-routine always begins with a label, then it contains one or more instructions and finally a RETURN

After RETURN, processing of the program is continued with the instruction to GOSUB. The program processing may never proceed to a RETURN instruction without a previous GOSUB.

The maximum allowable nesting depth for calling up sub-routines from sub-routines is four.

#main program

GOSUB sub1

...

#sub1

GOSUB sub2

...

RETURN

#sub2

GOSUB sub3

...

RETURN

#sub3

GOSUB sub4

...

RETURN

#sub4

...

Für die Editierung und Compilierung des BASIC Quellcodes wird die Standard BASIC Entwicklungsumgebung verwendet. Damit wird maximale Kompatibilität gewährleistet und die Portierung von Programmen oder Programmteilen für die Versionen Unit und M ermöglicht.

Deshalb müssen Sie, als Programmierer, unbedingt darauf achten, dass in Ihren Programmen keine Schlüsselwörter verwendet werden, die nicht Bestandteil des BASIC-Kernels auf der MICRO / MICRO PCB sind.

Die Verwendung führt zu unkontrollierbarem Fehlverhalten der C-Control und schwer zu lokalisierenden Fehlern.

Systemressourcen des C-Control Computers

Unter dem Begriff „Systemressourcen“ sind hier alle internen Funktionseinheiten zusammengefaßt, die sich aus den Eigenschaften des Mikrocontrollers ableiten oder durch das auf dem Chip befindliche Betriebssystem zur Verfügung gestellt werden. Wie diese Systemressourcen im BASIC-Programm angesprochen werden, wird weiter unten in der Befehlsübersicht beschrieben.

Echtzeituhr (RTC)

Im Hintergrund des Betriebssystems läuft ein mit 20 Millisekunden getakteter 8-Bit-Timer, dessen Wert jederzeit ausgelesen und zum Herstellen von Zeitbezügen im BASIC-Programm benutzt werden kann und Zeitbasis für die interne Echtzeituhr ist. Die RTC stellt die Uhrzeit in Stunden, Minuten und Sekunden in den entsprechenden Bezeichnern zur Verfügung.

Systembedingt ist die Genauigkeit der RTC sehr schlecht. Sie haben aber als Benutzer die Möglichkeit die Eigenschaften durch eine Kalibrierung zu verbessern. Das Beispiel TOOL_CLOCK_CHECK.BAS gibt Ihnen weitere Hinweise zu diesem Thema. Die RTC lässt sich außerdem mit einem DCF77 Empfänger und einer BASIC Systemerweiterung synchronisieren. Informationen hierzu finden Sie im Kapitel SYSTEMERWEITERUNGEN

Die internen Speicherzellen für Datum und Uhrzeit können vom BASIC-Programm aus gelesen und beschrieben werden. Durch das Beschreiben der Zeitspeicherzellen kann die Uhr also auch ohne DCF77-Empfang gestellt werden. Für Programmtests oder bei geringem Anspruch an die Ganggenauigkeit kann so auf die DCF77-Antenne oder die Kalibrierung verzichtet werden.

User-Bytes

Der Mikrocontroller verfügt über insgesamt 240 Bytes RAM. Der C-Control Steuercomputer belegt davon größten Teil für Betriebssystemfunktionen (Stack, Timer, Uhr, Zwischenspeicher für Berechnungen usw.). 24 Bytes stehen dem Anwender zur Verwendung in BASIC-Programmen zur Verfügung

Die Verwendung dieser Userbytes ist im Abschnitt zum DEFINE-Befehl weiter unten beschrieben.

tion"). For equally-ranked operators, calculation takes place from left to right. As in mathematics, however, influence can be taken on the calculation order by placing a bracket. CCBASIC supports a maximum of 3 bracket levels. For the sake of the clearness of a program, "wild" brackets should be avoided, however, and complex calculations distributed to several BASIC lines.

The following list shows the CCBASIC operator ranking order.

RANKING	OPERATORS
9	()
8	Function references
6	* / MOD SHL SHR
5	+ -
4	> >= < <= = <>
3	NOT
2	AND NAND
1	OR NOR XOR

Instructions for control of the program flow

- **Loop**

FOR variable = start TO end STEP increment

...

NEXT

The FOR loop executes the instructions up to NEXT as long as the value variable is equal to the value of the terms end. Before the first run, the value of the term start is calculated and assigned to the loop variable. In each run, the value of the incremental term is added to the loop variable. In the form

FOR variable = start TO end STEP increment

...

NEXT

The increment has the constant 1. The values of the end terms and the incremental term are recalculated for each loop run. That allows extended control of the course of the program.

A loop is always run through at least once even if **start** and **end** have the same values.

A/D-Wandler

Die C-Control MICRO hat keine eigene Referenzspannung. Als Referenz für die AD-Wandler dient die Betriebsspannung (5V). Spannungsregler stellen diese mit recht guter Genauigkeit zur Verfügung (+-5%)

Wenn Sie erhöhte Anforderungen an die Genauigkeit der AD-Wandler haben, müssen Sie die 5V Betriebsspannung mit entsprechender Genauigkeit zur Verfügung stellen.

Der angelegte Spannungswert gilt als Obergrenze des Messbereiches der A/D-Wandlung und entspricht dem Wandlungswert 255 (SFF hexadezimal).

Als Referenz für das untere Ende des Messbereiches der A/D-Wandlung dient stets das Groundpotential (Masse, „Minus“) der Betriebsspannung. An den A/D-Ports können Sensoren aller Art angeschlossen werden, die eine Ausgangsspannung von 0 bis 5 Volt liefern. In den meisten Fällen werden hier aktive Sensoren zur Anwendung kommen, um das Signal des eigentlichen Sensorelementes zu verstärken und den Ansprüchen an Auflösung, Linearität und Driftverhalten zu genügen.

Frequenz Eingang

Bei der C-Control MICRO kann nur Port4 als Frequenzeingang FREQ dienen.

FREQ2 wird von der IDE zwar akzeptiert (um die Kompatibilität zu C-Control I zu wahren) hat jedoch in der MICRO eine andere Funktion

Frequenzmessung an Pin FREQ

Über den Pin FREQ2 können Frequenzen von CMOS/TTL-kompatiblen Rechtecksignalen bis ca. 12000 Hz gemessen werden.

Frequenz Ausgang - BEEP

Bei der C-Control MICRO können alle Ports (mit Ausnahme Port 3) die Funktion eines Beep-Ports annehmen.

Die verfügbaren Ports sind PORT1, PORT2, PORT4, PORT5, PORT6.

Andere Ports werden von der Entwicklungsumgebung zwar akzeptiert (um die Kompatibilität zu C-Control I zu wahren), jedoch führt deren Verwendung zu absolut unkontrollierbarem Verhalten der MICRO

Tonausgabe an den BEEP-Pins

Am BEEP-Pin können per BEEP-Befehl (siehe unten) Rechtecksignale (0/5V) ausgegeben werden, die durch Anschluß eines piezoelektrischen Schallwandlers (Schallwandler ohne interne Elektronik zwischen BEEP und GND) als Töne hörbar gemacht werden können.

Definition of analogue ports

Ports as also variables are accessed in CCBASIC . Here also, each port used has to be previously defined.

- **Definition of one of the 4 A/D ports:**

DEFINE identifier AD[nrl]

whereby values of 1,2,5 or 6 are permitted for nr.

CAUTION:

Although other ports are accepted by the development environment (to preserve the compatibility with C-control I), their use leads to absolutely uncontrollable behaviour of the MICRO.

Mathematical and logical operators

This section provides a complete general view of the CCBASIC operators, functions and instructions.

- **Basic arithmetic operations:** + - * /
- **The module operator MOD** supplies the rest of an integer division,

a = 10 MOD 3

gives, for example for 'a' the value 1.

- **Relational operators:**

> (greater than), < (less than), >= (greater than or equal to), <= (less than or equal to), = (equals), <> (not equal to). The result of a relational operation is either -1 or 255 (comparison true) or 0 (comparison false).

a=10 <3

results, for example, for the value 'a' in the value 0.

- **Logical operators:**

NOT (negation), AND (and operation), NAND (and operation with following negation), OR (or operation), NOR (or operation with following negation), XOR (exclusive or

zesses, sie repräsentieren die gespeicherten Informationen. Der C-Control BASIC Steuercomputer verarbeitet und speichert ausschließlich ganzzahlige numerische Daten - sogenannte „Integerzahlen“ von 1, oder 8 Bit. Eine Variable von 8 Bit (Byte) kann nur nicht negative Werte von 0 bis 255 aufnehmen. Achten Sie bei allen Berechnungen darauf, daß die Ergebnisse diese Grenzwerte nicht über- oder unterschreiten, da es sonst zu sogenannten „Überläufen“ kommt.

a = 255 + 1

ergibt beispielsweise für a den Wert 0 und nicht 256, wenn a nur ein Byte repräsentiert!

Grundlegende Elemente von CCBASIC

Allgemeines

Jede Programmzeile enthält eine oder mehrere Anweisung, die durch Doppelpunkte : getrennt sind.

Zeilennummern, wie in älteren BASIC-Dialekten üblich, sind nicht notwendig. Werden dennoch Zeilennummern angegeben, so können diese als Sprungziel verwendet werden.

10 . . .

GOTO 10

Einen Einfluß auf die Reihenfolge der Programmoperationen haben die Nummern darüber hinaus nicht. Wenn beispielsweise im Quelltext auf eine mit 200 nummerierte Zeile eine Zeile 100 folgt, wird trotzdem die Zeile 200 vor der 100 abgearbeitet.

Kommentare können zur Erläuterung des geschriebenen Programms mit in den Quelltext aufgenommen werden und steigern dessen Lesbarkeit und Wartungsfreundlichkeit. Ein Kommentar in CCBASIC beginnt stets mit einem Hochkomma ' und erklärt den Rest der Zeile zum nicht zum Programm gehörigen Text.

a =b +c ´ . . . Kommentar . . .

Bezeichner

Bezeichner sind Programmelemente aus alphanumerischen Zeichen (A bis Z, 0 bis 9) die in vom Programmierer festgelegter Weise Objekte, wie Variablen und Konstanten, bezeichnen. Label-Namen und die sogenannten „reservierten Worte“ sind ebenfalls Bezeichner. Es erfolgt keine Unterscheidung von Groß- und Kleinbuchstaben. Ein Bezeichner beginnt stets mit einem Buchstaben oder mit einem Unterstrich. Leerzeichen innerhalb eines Bezeichners sind nicht erlaubt.

Definition of symbolic constants

It is good programmer style to use symbolic constants in the program

IF x > 123 THEN GOTO alarm

rather than "magic" figures.

The readability of the source text is increased by issuing significant identifiers for constants. A program is also easier to maintain when all the constants are defined globally. This applies in particular when one and the same constant is required several times in the program.

The definition of a symbolic constant takes place as follows:

DEFINE identifier value

whereby the value is either a decimal, hexadecimal or binary number. The example should therefore rather be

DEFINE limit 123

...

IF x > limit THEN GOTO alarm.

Definition of variables

The C-control/BASIC MICRO control computer makes 24-byte-storage cells of its internal storage (RAM) available to the user for use in its programs. All variables of a BASIC program are stored in this memory area. The 24 bytes can also be used bit-wise according to requirement. Contrary to standard-BASIC, in CCBASIC all variables used by the program have to be defined before being used for the first time. In doing so, the data type is to be specified (bit or byte) and a storage cell number can also (for bits has to!) be stated.

The user has to pay attention himself to the fact that no undesired overlapping arises when the memory locations are issued as otherwise the variables can be overwritten mutually. For example, bit[18] and byte[2] can occupy a part of the cell 2 of the memory area.

• **Definition of a bit variable:**

DEFINE identifier BIT [nr]

Values from 1 to 192 (24 bytes each with 8 bits) are permitted for nr.

• **Definition of a byte variable with cell number:**

DEFINE identifier BYTE[nr]

Funktionen

Eine Funktion führt eine definierte Operation - zum Beispiel eine Berechnung - durch und liefert durch ihren Aufruf einen Ergebniswert. Die meisten Funktionen erwarten ein oder mehrere Argumente, die in runden Klammern „()“ nach dem Funktionsbezeichner übergeben werden und durch Kommas getrennt sind. Einige Funktionen werden ohne Argument aufgerufen. In diesem Fall werden keine runden Klammern geschrieben.

RAND

EOF

In CCBASIC sind alle unterstützten Funktionen vordefiniert. Deren Bezeichner gehören zu den reservierten Worten. Die Formulierung anwenderdefinierter Funktionen ist in CCBASIC nicht vorgesehen.

Zuweisungen

Die Zuweisung ist die einfachste Form einer Programmanweisung. Nach dem Bezeichner einer Variablen, der ein Wert zugewiesen werden soll, folgt das Zuweisungszeichen „=“ und dann ein Term, der den zuweisenden Wert bestimmt. Eine Zuweisung entspricht damit einer einfachen mathematischen Formel.

a = 10

b = x - y

c = (a*a + b*b)

Befehle

Neben den einfachen Zuweisungen sind Befehle Anweisungen zur Ausführung von Programmoperationen durch den C-Control/BASIC Steuercomputer. Befehle beginnen stets mit einem reservierten Wort. Einige Befehle erwarten einen oder mehrere Parameter zur genauen Spezifikation der auszuführenden Programmoperation. Diese Parameter werden nach dem Befehlsbezeichner und einem Leerzeichen aufgeführt und dabei durch Kommas getrennt (Ausnahme PRINT, siehe Befehlsübersicht). Im Gegensatz zu den Argumenten beim Aufruf einer Funktion stehen die Befehlsparameter nicht innerhalb runder Klammern!

PAUSE 100

BEEP 140,50,BEEPER

Anweisungen zur Steuerung des Programmflusses

Diese Anweisungen erlauben, die Reihenfolge der an sich streng sequentiell abgearbeiteten Programmoperationen zu steuern und an Eingangswerte des Informationsverarbeitungsprozesses anzupassen. Sie bieten eine hohe Flexibilität bei der Algorithmenformulierung und sind für die Lösung mancher anwendungs-technischer

Apart from this, symbolic constants can be agreed by virtue of DEFINE lines (see below). Variables are accessed through their identifier. Before the variables are used for the first time in the program, this identifier has to be defined in a DEFINE line.

Labels

Labels mark certain points in the order of the program operations. Labels are goals of branch operations within an algorithm. In CCBASIC, labels stand at the beginning of a line and always start with a "#", followed by – without a blank – the identifier of the label.

The example shows the definition of the label "Label1" and is used in a branch command.

```
#label1 . . .  
GOTO label1
```

Terms

A term results immediately (as variable or constant) or by calculation in a certain value. Terms are parts of instructions and are positioned, for example for the assignment of a value to a variable, to the right of the assignment symbol "=". Terms are formed by combinations of operands and operators.

```
a +b=(x - 13) * 10
```

Operands and operators

In its basic form, an operand is either a constant, a variable or a function reference but can also itself be a term composed of operands and operators. Operators designate arithmetic operations which are to be executed by means of the surrounding operands. There is thereby a defined ranking order for the operators (see description of commands) which determines the order of the calculations.

Functions

A function carries out a defined operation – for example a calculation – and, when called up, supplies a result value. Most functions expect one or more arguments which are stated in round brackets "()" after the function identifier and separated by commas. Some functions are called up without an argument. In this case, no round brackets are required.

```
RAND  
EOF
```

In CCBASIC all functions supported are pre-defined. Their identifiers belong to the reserved words. The formulation of user-defined functions is not envisaged in CCBASIC.

gramms gespeichert. Die 24 Bytes können je nach Bedarf auch bitweise verwendet werden. Im Gegensatz zum Standard-BASIC müssen in CCBASIC alle vom Programm benutzten Variablen vor ihrer ersten Verwendung definiert werden. Dabei ist der Datentyp zu spezifizieren (Bit, oder Byte) und kann (für Bits muß!) eine Speicherzellennummer angegeben werden.

Der Anwender muß selbst darauf achten, daß keine unerwünschten Überlappungen bei der Vergabe der Speicherplätze entstehen, da es sonst zum gegenseitigen überschreiben der Variablen kommen kann. Beispielsweise belegen bit[18] und byte[2] jeweils einen Teil der Zelle 2 des Speicherbereiches.

• **Definition einer Bitvariablen:**

```
DEFINE bezeichner BIT [nr]
```

Dabei sind für nr Werte von 1 bis 192 (24 Bytes mit je 8 Bit) zulässig.

• **Definition einer Bytevariablen mit Zellennummer:**

```
DEFINE bezeichner BYTE[nr]
```

Dabei sind für nr Werte von 1 bis 24 (24 Bytes) zulässig.

```
DEFINE bezeichner WORD[nr]
```

werden vom Compiler zwar zugelassen dürfen, aber nicht verwendet werden, da die MICRO strikt Byte orientiert ist.

Die Verwendung Word-Variablen kann zu unkontrolliertem Verhalten der MICRO führen

Wenn bei Bytedefinitionen die Zellenangabe [nr] weggelassen wird, übernimmt der Compiler die Aufteilung auf den Speicherbereich.

```
DEFINEa BYTE  
DEFINE c BYTE
```

Die automatische Aufteilung der Variablen auf den Speicher durch den Compiler beginnt bei Zellennummer 1. Bei Definition weiterer Bits und Bytes mit Angabe der Zellennummer ist wieder auf unerwünschte Überlappung zu achten. Ein bereits definierter Variablenbezeichner darf nicht ein zweites Mal definiert werden.

Programming the C-Control Computer

CCBASIC is the BASIC dialect used for programming the C-control BASIC control computer. The syntax corresponds approximately to that of standard BASIC. In some commands there are deviations or extensions which are tailored specially to the hardware of the control computer.

The CCBASIC of the MICRO is for the most part compatible with other versions of the C-control I computer but fundamentally limited to 8-bit operations/instructions.

Examples for learning CC-BASIC

On the CD you will find short examples which will explain to you the use of the BASIC commands step-by-step.

These examples partly require a PC connection as they use the "HyperTerminal" as means of output – you will find it in Windows under ->Program ->Accessories ->Communication.

Start C-CONTROL IDE, load a BASIC program from the directory "BASIC". If you wish to load the program, you have to close HyperTerminal as otherwise the IDE will find an occupied interface.

Start HyperTerminal in the configuration 9600 baud, 8n1. It is now ready for the output which C-control generates.

What is a Program?

A program is the description of an information processing process. In the course of such a process, a number of output values is calculated from a number of variables or constant input values. The output values are either the goal of the information acquisition themselves or serve indirectly the reaction to the input values. Alongside the actual calculations, a program can contain instructions for access to the hardware of the computer or for controlling the program flow.

A BASIC program consists of several lines of a so-called source text. Each line thereby contains one or several compute statements or control instructions. Apart from these instruction themselves, their order determines quite significantly the information processing mentioned at the beginning. The execution of the operations which correspond corresponding to the instructions by the control computer takes place sequentially, i.e. one after another. A series of program instructions with a certain goal is also called an algorithm. Data are the objects of the information processing process, they represent the stored information. The C-control BASIC control computer processes and stores exclusively whole-numbers, numeric data – so-called "integers" of 1 or 8 bits. A variable of 8 bits (bytes) can only accept not negative values from 0 to 255. Pay attention in all calculations to the fact that the results of these limit values are not exceeded or fall short of as otherwise so-called "overflows" can occur.

Mathematische und logische Operatoren

Dieses Kapitel gibt einen kompletten Überblick über die CCBASIC Operatoren, Funktionen und Anweisungen.

- **Grundrechenarten:** + - * /

- **Der Modulooperator MOD** liefert den Rest einer Integerdivision,

a = 10 MOD 3

ergibt beispielsweise für a den Wert 1.

- **Vergleichsoperatoren:**

> (größer als), < (kleiner als), >= (größer oder gleich), <= (kleiner oder gleich), = (gleich), <> (ungleich)

Das Ergebnis einer Vergleichsoperation ist entweder -1 bzw. 255 (Vergleich wahr) oder 0 (Vergleich falsch).

a=10 <3

ergibt beispielsweise für a den Wert 0.

- **logische Operatoren:**

NOT (Negation), AND (Und-Verknüpfung), NAND (Und-Verknüpfung mit anschließender Negation), OR (Oder-Verknüpfung), NOR (Oder-Verknüpfung mit anschließender Negation), XOR (Exklusiv-Oder-Verknüpfung). Die logischen Operatoren können außer zur Formulierung von Bedingungen (meist in Verbindung mit Vergleichsoperationen) auch für binäre Bytemanipulationen benutzt werden.

- **Schiebeoperatoren:**

SHL (nach links schieben), SHR (nach rechts schieben) werden zum bitweisen Verschieben von Bitmustern in Byte- oder Wordvariablen benutzt. Links des Operators steht der zu schiebende Wert, rechts die Zahl, um wieviel Bits verschoben werden soll. Beim Linksschieben entspricht jede einzelne Verschiebung einer Multiplikation mit 2, beim Rechtsschieben einer Division durch 2.

a = 10 SHL 3

entspricht also: $a=10*2*2*2$ und ergibt beispielsweise für a den Wert 80.

Mathematische Funktionen und Befehle

Die Argumente x und y, je nach Funktion oder Befehl, sind stets Terme (Definition siehe oben).

Using a Digital Port as Input

Digital inputs are used to query switching states. If a digital port is used as input, it maintains an undefined level in an unswitched state that has to be defined with a PULLUP resistance, for example. For example, if a reed contact is connected to this port, a logical 1 ("true") is read off the port with open switch and a logical 0 ("false") with closed switch.

Make absolutely certain that the input value must possibly be inverted (NOT operator, see command description) depending on the switching of the port and the logical statement your program is supposed to contain.

ATTENTION!

Never use Port 3 in applications where it is not sure that Port 3 is log. high when applying the operating voltage. Otherwise, the download mode is active and the user program does not run.

Using a Digital Port as Output

If a digital port is used as output, the subsequent ICs, transistors or low-current LEDs can be operated directly. The maximum admissible load current is 10 mA. In any case, you have to ensure a sufficient current limiting, e.g. through a resistor, otherwise the microcontroller may be destroyed!

Within the microcontroller, the internal assignment of a digital port as output or input takes place during the first execution of the user program. After switching the operating voltage or after a reset, all digital ports initially behave as inputs electronically, meaning they maintain high level via a pullup-resistor.

Analogue Ports

With the C-Control MICRO, not all ports can assume the function of an analogue port.

The available analogue ports are PORT1, PORT2, PORT5, PORT6.

Other ports are accepted by the development environment (in order to maintain the compatibility to C-control I), but their use leads to absolutely uncontrollable behaviour of the MICRO.

A/D Converters

The C-Control MICRO does not have an own reference voltage. The operating voltage (5 V) serves as reference for the AD converters. Voltage stabilisers provide this with adequate accuracy (+-5%).

If you have higher demands on the accuracy of the AD converters, you must provide the 5V operating voltage with the corresponding accuracy.

The applied voltage value is the top limit of the measuring range of the A/D conversion and corresponds to the conversion value 255 (SFF hexadecimal).

The ground potential (ground "minus") of the operating voltage always serves as reference for the lower limit of the measuring range of A/D conversion. You can connect sensors of any kind to the A/D ports that provide an output voltage of 0 to 5 Volt. In

Anweisungen zur Steuerung des Programmflusses

• Schleife

FOR variable = anfang TO ende STEP schrittweite

...
NEXT

Die FOR-Schleife führt die Anweisungen bis zum NEXT solange aus, bis der Wert der variable gleich dem Wert des Terms ende ist. Vor dem ersten Durchlauf wird der Wert des Terms anfang berechnet und der Schleifen-variablen zugewiesen. In jedem Durchgang wird der Wert des schrittweite- Terms zur Schleifenvariablen addiert. In der Form

FOR variable = anfang TO ende STEP schrittweite

...
NEXT

beträgt die Schrittweite konstant 1. Die Werte des ende-Terms und des schrittweite-Terms werden mit jedem Schleifendurchlauf neu berechnet. Das gestattet eine erweiterte Kontrolle des Programmverlaufes.

Eine Schleife wird auf jeden Fall mindestens einmal durchlaufen, auch wenn **anfang** und **ende** gleiche Werte haben

FOR-Schleifen können ineinander verschachtelt werden. Die Verschachteltiefe ist nur durch den für die Schleifenvariablen erforderlichen Speicherplatz beschränkt.

FOR variable = anfang TO ende1

FOR variable = anfang TO ende2

FOR variable = anfang TO ende3

.....

NEXT

NEXT

NEXT

Jede FOR-Schleife darf im Verlauf des Programms nur über ihre eigene NEXT-Anweisung laufen. Folgender Quelltext kann zwar kompiliert und in den Steuercomputer geladen werden, wird jedoch nicht wie vielleicht erwartet funktionieren:

Download into C-CONTROL MICRO

In order to download a user program from the development environment, **keep the button "DOWNLOAD" depressed** and slide the slider switch to the position "ON" (towards the interface plug). The readiness for download is indicated with the yellow LED (L5).

Now you can load the program into C-CONTROL. Activate the download in the menu DEVELOPMENT > LOAD TO C-CONTROL.

A window shows the start and the end of the download. Close this window after the download and put the slider in the OFF position. After 2 seconds, put it to ON again in order to start the example program.

ATTENTION!

Never use Port 3 in applications where it is not sure that Port 3 is log. high when applying the operating voltage. Otherwise, the download mode is active and the user program does not run.

Remove JP2 if there are problems with the download (normally not required).

Examples

Load the examples in order to familiarise yourself with the programming language. For principle explanations about CCBASIC, refer to the following chapter.

The Programming Language CCBASIC

The Concept of C-Control MICRO/MICRO PCB

The MICRO versions of C-CONTROL are designed to realise smaller control functions with a minimum extent of components and on the smallest space. Six ports that can have several different functions (also within a program) allow multifarious use. The operating system is also tuned to fit the purpose. It consists of the KERNEL and offers the programmer an only slightly reduced functionality with respect to the C-Control M/UNIT. For special requirements, Conrad Electronic provides operating system extensions and drivers that do however reduce the usable BASIC program memory.

Without additional modules, you can use 2 kB BASIC memory which is also sufficient for complex applications in the application range.

The standard BASIC development environment is used for editing and compiling the BASIC source code. This ensures maximum compatibility and enables the porting of programs or program parts for the versions Unit and M.

Therefore you as a programmer have to make absolutely certain that no keywords are used in your programs that are not a part of the BASIC kernel on the MICRO / MICRO PCB.

Dabei ist label der Anfangspunkt der Unterroutine. In den sogenannten Unterroutinen sind Programmabschnitte zusammengefaßt, die mehrfach im Verlauf der Programmabarbeitung benötigt werden. Eine Unterroutine beginnt stets mit einem Label, enthält dann eine oder mehrere Anweisungen und abschließend ein RETURN. Nach dem RETURN wird die Programmabarbeitung mit der Anweisung nach dem GOSUB fortgesetzt. Die Programmabarbeitung darf ohne ein vorheriges GOSUB niemals an eine RETURN-Anweisung gelangen.

Die maximal zulässige Verschachtelungstiefe bei Aufrufen von Unterroutinen aus Unterroutinen ist vier.

#hauptprogramm

GOSUB sub1

...

#sub1

GOSUB sub2

...

RETURN

#sub2

GOSUB sub3

...

RETURN

#sub3

GOSUB sub4

...

RETURN

#sub4

...

RETURN

• Programmende

END

Gelangt der Steuercomputer im Verlauf der Programmabarbeitung zur END-Anweisung, wird die Programmabarbeitung beendet. Das System verharrt dann in einem inaktiven Zustand.

• Verzögerung des Programmflusses

Die Anweisung

WAIT conditionterm unterbricht die Programmausführung solange, bis die Berechnung des conditionterm einen Wert ungleich 0 ergibt.

JUMPER JP2

If you want to do without the available output possibilities altogether because you have connected your application to the post plug, pull off both jumpers JP2 and JP1.

C-Control Computer MICRO / MICRO PCB

The P/E Board provides the operating voltages for the operation of the C-Control computer on the board.

If you operate the computer in a different hardware environment, you must provide a stable operating voltage.

Version Micro

This is the "naked" version of the computer as a chip with the loaded operating system. You can see the pin assignment in the following graphic. The MICRO requires a stable power supply 5V (+- 10%) that must provide a current of at least 10 mA. Currents through connected components (e.g. LEDs) are not considered here.

For programming the computer or the operation of an application on the P/E Board, insert the C-Control MICRO into the provided socket before establishing a connection to other devices (e.g. the PC) and before applying the operating voltage.

The pins of the C-Control MICRO have the following functions:

PIN 1 - VDD +5V	PIN 8 - GND
PIN 2 - P6/AD6/BEEP	PIN 7 - P1/AD1/BEEP/TXD
PIN 3 - P5/AD5/BEEP	PIN 6 - P2/AD2/BEEP/RXD
PIN 4 - P4/BEEP/FREQ	PIN 5 - P3 INPUT

Version MICRO PCB

This is the comfortable version for direct connection to LEDs or buttons, for example, which is mainly used in model construction.

The MICRO PCB requires a stable power supply 5V (+- 10%) that must provide a current of at least 10 mA. Currents through connected components (e.g. LEDs) are not considered here. There is an additional voltage controller on the board that admits input voltages of 8 V up to max. 12 V via a separate connection.

If you use this option, the following guidelines apply:

- You may not feed any voltage to the intended connections of the 5 V supply.
- The voltage (5 V) which is then switched to the connections of the 5 V operating voltage can be used for own purposes (e.g. pull-up resistance, other components).
- The power intake of all components supplied via the controller may not exceed 50 mA.

Attention!

PRINT term

gibt das Ergebnis der Berechnung von term aus.

PRINT "text"

überträgt den in Anführungszeichen stehenden Text. In beiden Fällen wird an die Übertragung ein Zeilenvorschubzeichen angehängt, welches das Terminalprogramm veranlaßt, die nächste Ausgabe in der nächsten Bildschirmzeile vorzunehmen. Der Zeilenvorschub kann unterdrückt werden, wenn dem PRINT-Befehl nach dem Parameter (term oder „text“) ein Semikolon hinzugefügt wird.

PRINT term;

Oder:

PRINT "text";

CCBASIC unterstützt außerdem mehrere Ausgaben mit einem PRINT-Befehl, wobei die einzelnen Parameter durch Komma oder Semikolon getrennt werden. Ein Komma fügt in die Ausgabe ein Tabulatorzeichen ein, das entsprechend den Einstellungen im Terminalprogramm als eine Anzahl von Leerzeichen am Bildschirm erscheint. Sollen zwei Ausgaben ohne Zwischenraum aufeinander folgen, so sind diese im PRINT-Befehl durch ein Semikolon zu trennen.

PRINT "a= ", a

PRINT "a= "; a

Ein einzelner PRINT-Befehl ohne Parameter gibt nur einen Zeilenvorschub aus.

PRINT

• Byteweise Kommunikation über die serielle Schnittstelle

Während PRINT kurze Zeichenketten zur Darstellung eines numerischen Wertes senden beziehungsweise erwarten, kann es wünschenswert sein, einzelne Bytes seriell zu übertragen. Dafür bietet CCBASIC die Befehle PUT und GET.

PUT term

sendet den berechneten Wert eines Terms.

Programmer/Evaluation Board

The P/E Board provides the user with a serial interface with level conversion in order to load compiled programs from the development environment into the C-Control computers version MICRO and MICRO.

In addition, there are a number of input/output possibilities (LEDs, beepers, buttons, pots) so that you can quickly and comfortably try out the example programs or your own applications.

Power supply takes place with a 9 V battery or a power pack (8 to 12 V).

Attention!

For technical reasons, the P/E-Board does not support 19200 Baud (time constants in the level converter)

Handling

Programmer/Evaluation Board

Connection of the C-Control Computers

Plug one C-Control computer, version MICRO into the provided IC socket.

Observe the correct polarity. The notch on the casing of the IC is next to the condenser C3.

If you want to program the version MICRO PCB, connect the connection wires to the clamp bar according to the colour marking.

ATTENTION!

There may never be 2 C-Control MICRO plugged and connected at the same time.

Connection to the PC

Connect your PC's serial interface with the interface socket on the P/E Board.

Use the provided interface cable.

Power supply:

The P/E Board is powered with a 9 V battery or a power pack. If you use a power pack, make absolutely certain that you use the right polarity (negative must be on the outside of the plug) and the right voltage range (8 to 12 V DC).

The P/E Board is now ready for operation.

Operating Elements

In order to download a user program from the development environment, **keep the button "DOWNLOAD" depressed** and slide the slider switch to the position "ON" (towards the interface plug).

The readiness for download is indicated with the yellow LED (L5).

Port 1 and 2 now have interface function, therefore LED L1 and L2 are lit.

benutzen. TOG steht für englisch „toggle“. Der TOG Befehl benötigt weniger Platz im EEPROM und wird schneller als die klassische NOT-P-Konstruktion ausgeführt. Die Portvariable P darf beim TOG Befehl nur für einen einzelnen Digitalport stehen

• Deaktivieren eines Ports mit DEACT

Sobald einer Portvariablen erstmalig ein Wert zugewiesen wird, schaltet der Steuercomputer die zugehörigen Hardwarestrukturen im Prozessorchip (Transistoren) auf Ausgangsbetrieb. Es fließt also entsprechend der angeschlossenen Schaltung Strom aus bzw. in den Prozessor (max. 10 mA zulässig!). Der Befehl

DEACT portvar

deaktiviert den angegebenen Port. Das heißt, der Port wird in einen hochohmigen Zustand geschaltet und arbeitet im Eingangsbetrieb. Der DEACT Befehl darf auf einzelne Digitalports oder Byteports angewendet werden.

Definition und Anwendung von Datentabellen

Im Standard-BASIC dienen DATA-Zeilen zum Ablegen von konstanten Datenblöcken, auf die dann sequentiell zugegriffen werden kann. CCBASIC unterstützt keine DATA-Zeilen, bietet jedoch ein weitaus flexibleres Werkzeug zur Definition und zum Zugriff auf Datenblöcke. Konstante Daten können in Form von Tabellen abgelegt werden. Jede Tabelle bekommt einen Bezeichner (tablename) zugewiesen und kann beliebig viele Einträge enthalten, soweit der Programmspeicher Platz bietet. Jeder Dateneintrag (Cx) wird als Integerwert abgelegt und belegt somit zwei Bytes. Dabei können die Daten direkt im Quelltext aufgeführt werden, dürfen aber nur Werte bis max. 255 annehmen, da auch Tabellen nur Byteweise verarbeitet werden

TABLE tablename

CO CI C2 C3 C4 C5 Cn

TABEND

oder vom CCBASIC-Compiler aus einer externen Textdatei importiert werden

TABLE tablename "tabfilename"

Die Tabellendefinitionen müssen stets am Ende eines Programms, hinter dem END Befehl stehen, da die Daten nahtlos hinter den vorangehenden Codebytes im EEPROM-Speicherchip abgelegt werden. Die Programm-abarbeitung darf nie über Tabellendaten laufen, da die Daten sonst als BASIC Befehle interpretiert werden würden, was sicher zum Absturz des Systems führt. Der Zugriff auf die Tabellendaten erfolgt mit dem Befehl

Product Description

Intended Use

The C-Control BASIC control computer serves for the programmable access of electric and electronic devices operated with low-voltage safety current. These devices may be integrated into any technical systems that do not directly or indirectly serve for medical, health or life-sustaining purposes or whose operation may pose risks to persons or assets.

You may only use the provided PC software and the P/E Board for programming the computer.

Safety Notices

Read this section especially careful! Non-compliance with the safety instructions poses mortal danger due to electric shock or fire!

The C-Control/BASIC control computer MICRO/MICRO-PCP as well as the P/E Board were subjected to a safety test in accordance with the valid statutory regulations and certified accordingly (CE). With proper handling, this device normally does not pose a health risk.

AS electronic devices, the C-Control/BASIC control computer MICRO/MICRO-PCP as well as the P/E Board have to be treated with the usual caution and care. Non-observance of the listed notices or a utilisation purpose other than the one described herein may lead to damages or the destruction of the control computer or connected devices.

The devices are not protected against flashover and may not be used in high-voltage current industrial facilities. The maximum input parameters according to the specifications listed in the Technical Data

may not be exceeded. You may not operate the devices in rooms or surroundings with flammable or caustic gasses, vapours or dusts.

You may not commission the devices immediately after bringing them from a cold into a warm room. The condensation, which may occur, may lead to function disruptions or the breakdown of electronic components on the device.

Avoid strong magnetic fields like in proximity to machines/loudspeakers.

You must establish all electric connections from and to the device prior to connecting the supply voltage. Connecting or disconnecting cables or establishing and disconnecting connections to components of the target application during operation may lead to the destruction of the control computer or the connected devices.

For supply of the P/E Board, connect a stabilised DC voltage of 8 V to 12 V. Only use tested laboratory power packs or stabilised power packs.

Beachten Sie bitte, daß während des Zugriffs die interne Uhr weiterläuft. Der Sekundenzwert sollte daher stets zuerst ausgelesen werden. Steht er auf 59, so muß nach dem Lesen der letzten interessierenden Zeitinformation (z.B. HOUR) der Sekundenzwert nochmals gelesen und auf =0 getestet werden. In diesem Fall ist das Auslesen der Echtzeituhr zu wiederholen, da eine neue Minute angebrochen ist

Weitergehende Zeitinformatoren sind bei der C-Control MICRO nicht verfügbar, können aber bei Bedarf mit einem kleinen Unterprogramm erzeugt werden

GENAUIGKEIT:

Die RTC der C-CONTROL MICRO wird von einem frei laufenden Oszillator gespeist der Ansprüche für eine genaue Uhr prinzipiell nicht erfüllen kann. Der Hersteller gibt eine Grundgenauigkeit von 25 % an. Damit aber die RTC aber trotzdem brauchbar bleibt, haben Sie die Möglichkeit den Takt der Uhr zu kalibrieren.

Das Tool CLOCK_CHECK.BAS hilft Ihnen bei der Kalibrierung und gibt weitere Hinweise

FREQ2=Wert

schreibt einen Korrekturfaktor in das Betriebssystem, der den Systemtakt um 0,2% je Digit (ausgehend vom Defaultwert 128, nach dem RESET) korrigiert. Damit lässt sich der Takt auf 0,2% genau einstellen, was einem Fehler der RTC von 170s/24h entspricht.

Der Inhalt von FREQ2 kann nicht gelesen werden und eine Anweisung wie etwa

a=FREQ2

führt zu unkontrollierbarem Fehlverhalten, da das entsprechende TOKEN vom INTERPRETER der MICRO nicht erkannt wird.

Anmerkung:

YEAR

MONTH

DAY

DOW

Sind Variablen die für das Betriebssystem reserviert sind, aber bei der MICRO nicht verwendet werden dürfen.

Introduction

Thank you for purchasing the Programmer/Evaluation Boards for C-Control MICRO/MICRO-PCB.

It lets you program both types of the C-Control Micro in the development environment

and load the programs to C-Control MICRO / MICRO PCB.

The Programmer/Evaluation Board (P/E-Board) for C-Control MICRO/MICRO-PCB also offers you a series of input/output possibilities (LEDs, beepers, buttons, poti) so that you can quickly and comfortably try out the example programs or your own applications.

Conrad Electronic GmbH
D-92240 Hirschau

Warranty

Prior to commissioning the P/E board or connected devices, read the entire operating manual. It explains the correct handling and points out possible hazards.

Warranty cannot be claimed for damages resulting from non-compliance with this operating manual and we will not assume any liability.

Notices about the Limited Warranty and Liability

The C-Control MICRO is a BASIC control computer. The software integrated as ROM mask in the microprocessor and the corresponding PC software is delivered as is.

Conrad Electronic will not assume any guarantee that the performance features correspond to individual requirements or that the software in the microprocessor and the PC software runs without disruption or error in all cases. The user bears the entire risk regarding the quality and capacity of the device including all software.

Conrad Electronic warrants the function of the enclosed application examples under observance of the conditions specified in the technical data. Should the P/E-Board, the C-Control MICRO or the PC software turn out to be defect or insufficient, the customer assumes all resulting costs for service, repairs or correction.

The warranty of Conrad Electronic is expressly limited to the exchange of the device within the warranty period in case of apparent defects on the hardware like mechanical damage, missing or faulty equipment with electronic components with the exception of socketed integrated circuits and jumpers. We will not assume any liability for damages directly connected to the use of the Programmer/Evaluation Boards or resulting thereof.

Claims based on indispensable statutory regulations about product liability are exempt from this.

TORZEIT:

Das Beschreiben der Systemvariablen FREQ stellt die Torzeit ein (20 ms Schritte). PORT4 muss für diese Funktion ein Eingang sein.

Die Torzeit ist die Zeit, während der Impulse am Eingang gezählt werden. Die Torzeit wird in Einheiten von 20ms angegeben. Verwenden Sie kleine Torzeiten für hohe Frequenzen.

Bei einer Torzeit von 20ms ergeben z.B. 10Khz einen Zählerstand in FREQ von: $0.02 \cdot 10000 = 200$

Bei großen Torzeiten lassen sich auch tiefe Frequenzen recht genau messen. Eine neue Torzeit wird erst nach Ablauf der vorherigen wirksam.

define FREQPORT p[4]

FREQ=5 stellt die Torzeit auf 100ms

ZÄHLERSTAND:

x = FREQ

Das Lesen der Variablen FREQ liefert den während der letzten Torzeit erreichten Zählerstand.

Er entspricht der Frequenz:

$FREQUENZ = x / TORZEIT$

FREQUENZBEREICH:

Da die Systemvariable FREQ nur 1 Byte hat, können auch nur 255 Ereignisse gezählt werden.

Passen Sie deshalb die Torzeit an die Frequenz, die sie verarbeiten wollen, an. Die kleinste Torzeit ist 1 (also 20ms) und verarbeitet Frequenzen bis 12 kHz. Arbeiten Sie mit unbekanntem Frequenzen können Sie eine automatische Torzeiteinstellung programmieren. Diese beginnt mit dem kleinsten Wert und vergrößert die Torzeit solange bis sich ein nennenswertes Zählergebnis in FREQ einstellt. Die aktuelle Torzeit gibt dann zusammen mit FREQ präzise Auskunft über die gemessene Frequenz

EINSCHRÄNKUNGEN

Während PAUSE u. BEEP ist die Zählung unterbrochen, wird aber nach Ablauf der Zeit regulär fortgesetzt.

GB Important! Read this!

Prior to commissioning the Programmer/Evaluation Board for C-Control MICRO/MICRO-PCB or connected devices, read these operating instructions completely! It explains the correct handling and points out possible hazards.

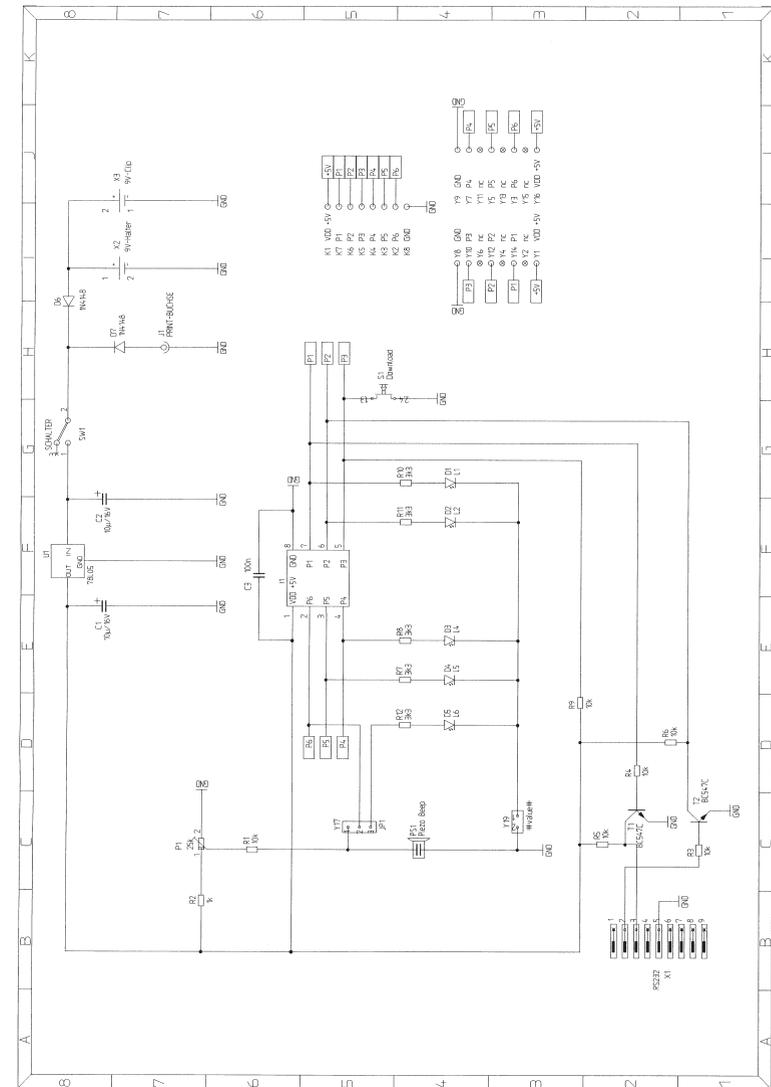
Conrad Electronics will not assume any liability for damages resulting from the non-compliance with this manual. These damages are also not covered by the warranty.

Content

Important! Read this!	42
Content.....	42
Introduction	44
Warranty	44
Service	45
Product Description	46
Intended Use	46
Safety Notices.....	46
Features	47
Handling	48
Programmer/Evaluation Board	48
Connection of the C-Control Computers	48
Connection to the PC	48
Power supply:	48
Operating Elements.....	48
Display Elements.....	49
Jumper for Configuration	49
C-Control Computer MICRO / MICRO PCB.....	50
Commissioning	51
Software Installation	51
The Programming Language CCBASIC.....	52
The Concept of C-Control MICRO/MICRO PCB	52
System Resources of the C-Control Computer	53
Realtime Clock (RTC)	53
User Bytes	53
Digital Ports.....	53
Analogue Ports	54
Frequency Input	55
Frequency Output - BEEP	55

Die Programme lassen sich laden, aber es tut sich nichts beim Start

- Sie haben das P/E-Board nicht richtig konfiguriert (Jumper)
- Sie haben eine Anwendung in welcher PORT3 beim Einschalten nicht auf log High liegt
- Ihre Anwendung braucht zu viel Strom und überlastet Spannungsregler oder Batterie



Ⓧ Diese Bedienungsanleitung gehört zu diesem Produkt. Sie enthält wichtige Hinweise zur Inbetriebnahme und Handhabung. Achten Sie hierauf, auch wenn Sie dieses Produkt an Dritte weitergeben.

Heben Sie deshalb diese Bedienungsanleitung zum Nachlesen auf!

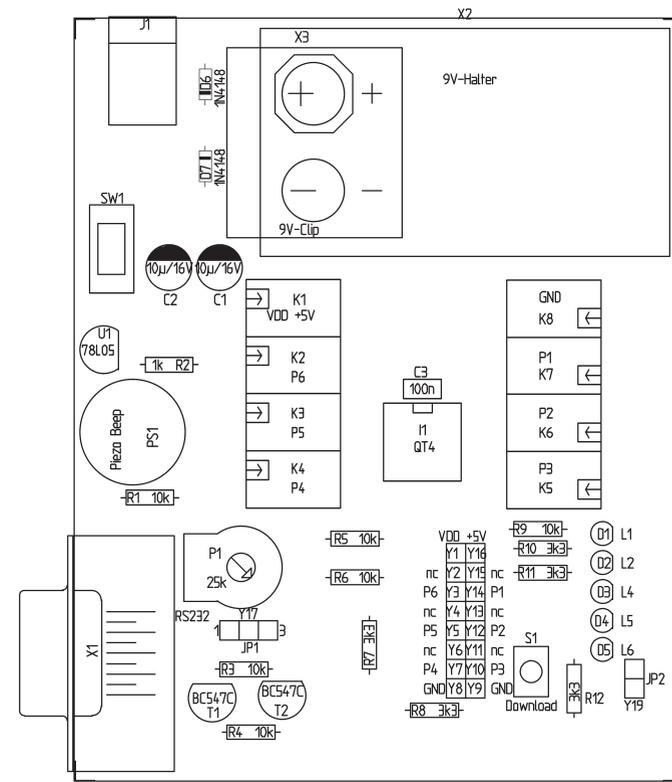
Eine Auflistung der Inhalte finden Sie in dem Inhaltsverzeichnis mit Angabe der entsprechenden Seitenzahlen auf Seite 3.

Ⓦ These Operating Instructions are part of the product. They contain important information on commissioning and installation. Please follow them, including when passing this product on to third parties.

Please keep the Operating Instructions for future reference!

The contents page on page 42 lists the contents of these instructions together with the relevant page number.

Programmierung des C-Control Computers	18
Die Beispiele zum Erlernen von CC-BASIC	18
Was ist ein Programm?	18
Grundlegende Elemente von CCBASIC	19
Allgemeines.....	19
Bezeichner	19
Variablen und Konstanten.....	20
Label	20
Terme.....	20
Operanden und Operatoren.....	20
Funktionen	21
Zuweisungen.....	21
Befehle	21
Anweisungen zur Steuerung des Programmflusses	21
Compileranweisungen	22
Definition symbolischer Konstanten	22
Definition von Variablen	22
Definition von Digitalports.....	24
Definition von Analogports.....	24
Mathematische und logische Operatoren.....	25
Rangfolge von Operatoren und Funktionsaufrufen.....	26
Anweisungen zur Steuerung des Programmflusses	27
Kommunikation über die serielle Schnittstelle.....	30
Portbefehle.....	32
Definition und Anwendungung von Datentabellen	33
Die Echtzeituhr.....	34
Interner Timer,Tonerzeugung, Frequenzmessung	36
Systemerweiterungen und Treiber.....	38
Problemlösungen.....	38
Es lässt sich kein Programm laden	38
Die Programme lassen sich laden, aber es tut sich nichts beim Start.....	39
Technische Daten	41



SERV. COMP TECHNO (1.00 11/24/103 ccontrol micro pa-board.fc)

+

kreise und Steckbrücken. Es besteht keine Haftung für Schäden, die unmittelbar durch oder in Folge der Anwendung des Programmer/Evaluation Board's stehen. Unberührt davon bleiben Ansprüche, die auf unabdingbaren gesetzlichen Vorschriften zur Produkthaftung beruhen.

Jede C-Control MICRO sowie das P/E Board verlassen das Werk in einwandfreiem und funktionsgeprüften Zustand!

Conrad Electronic bietet für diese Produkte eine **Gewährleistungs-dauer von 24 Monaten**. Innerhalb dieser Zeit werden eventuelle Transportschäden bei der Auslieferung, Fertigungsmängel oder Ausfälle am Gerät kostenfrei behoben.

Sollten die Leistungsmerkmale der C-Control MICRO oder des P/E Boards Ihren individuellen Ansprüchen nicht genügen, nutzen Sie bitte unsere **Geld-Zurück-Garantie von 14 Tagen**. Senden Sie das Gerät innerhalb dieser Zeit ohne Gebrauchsspuren und in der Originalverpackung zur Erstattung des Warenwertes oder zur Verrechnung zurück.

Alle Fristen gelten ab Datum der Rechnung beziehungsweise des Kassenbons.

Conrad Electronic übernimmt keine Haftung für Folgeschäden an Sachwerten oder Personen, die durch Anwendung dieser Produkte entstehen!

Service

Zu Ihrer Beratung stellt Conrad Electronic Ihnen ein kompetentes Team von Servicemitarbeitern zur Seite. Jede Anfrage wird schnellstmöglich bearbeitet. Spezialfragen werden an die Entwicklungsingenieure CTC weitergeleitet.

Um unnötige Verzögerungen zu vermeiden, möchten wir Sie jedoch bitten, vor einer Anfrage noch einmal diese Anleitung, die Online-Hilfen der Programmiersoftware, die Text- und Beispieldateien und nach Möglichkeit die Informationsseiten im Internet zu studieren. Meist findet sich so schon die Lösung eines Problems!

Ihre Anfragen richten Sie bitte an unsere Abteilung Technische Kundenbetreuung.

Bei Fragen wenden Sie sich an unsere Technische Beratung

Deutschland: *Tel. 0180/5 31 21 16 oder 09604/40 88 47*
 Fax 09604/40 88 48

e-mail: tkb@conrad.de
Mo. - Fr. 8.00 bis 18.00 Uhr

Österreich: *Tel. 0 72 42/20 30 60*
 Fax 0 72 42/20 30 66
e-mail: support@conrad.at
Mo. - Do. 8.00 bis 17.00 Uhr
Fr. 8.00 bis 14.00 Uhr

System Extensions and Drivers

For the C-control MICRO CCBASIC, Conrad Electronic provides system extensions and drivers for accessories. For this, PAGE1 and PAGE0 at the back end of the basic program memory are envisaged in the MICRO. Each of these pages occupies 256 bytes and reduces the basic program memory correspondingly if they are loaded.

While extensions and drivers are loaded for PAGE 0 together with the BASIC program, these extensions and drivers require their own BASIC loading program for PAGE1 which is obtainable together with examples.

These extensions and drivers are always linked with the compiler instruction

SYSCODE "ADDONS.SI9"

and

the SYSCODE command may only appear once in a CCBASIC program and should be positioned at the end, after any table definitions.

System extensions activate additional BASIC commands as a rule while system drivers contain special routines for periphery devices and are called up with the BASIC command

SYS adr

whereby adr is a constant and the address determines to where branching takes place.

You can find an exact description and examples for the application with the extensions and drivers offered.

Caution:

To prevent customers deleting the operating system by mistake when experimenting with self-written assembler programs, (flash memory !!), the execution of non-authorized machine programs is prevented.

Troubleshooting

Programs cannot be loaded

- You have selected the wrong interface in the IDE
- You are not using the interface cable which came with the delivery
- The battery is empty or the power supply unit has too little voltage
- You did not follow the instruction or downloading precisely enough.
- You connected the MICRO/MICRO-PCB incorrectly
- Remove the JP2 before downloading (normally not necessary)

Nachdem die Geräte von einem kalten in einen wärmeren Raum gebracht wurde, dürfen sie nicht sofort in Betrieb genommen werden. Das möglicherweise entstandene Kondenswasser kann zu Funktionsstörungen oder zum Ausfall elektronischer Bauelemente am Gerät führen.

Vermeiden Sie starke Magnetfelder, wie sie in der Nähe von Maschinen oder Lautsprechern vorkommen.

Alle elektrischen Verbindungen von und zum Gerät sind stets vor Anschluß der Versorgungsspannung herzustellen. Das Aufstecken oder Abziehen von Verbindungskabeln oder das Herstellen oder Lösen von Verbindungen zu Komponenten der Zielapplikation während des Betriebes können zur Zerstörung des Steuercomputers oder angeschlossener Geräte führen.

Zur Versorgung des P/E Boards ist eine stabilisierte Gleichspannung von 8V bis 12V anzuschließen. Verwenden Sie dazu nur geprüfte Labornetzgeräte oder stabilisierte Steckernetzteile.

Das P/E Board kann auch mit einer 9V Batterie betrieben werden.

Die C-Control MICRO/MICRO PCB hat Eingänge für den wahlweisen Betrieb mit 5V oder 8 bis 12V

Bei ausreichender Ladung ist auch ein Betrieb mit vier NiCd-Akkuzellen möglich.

Auf keinen Fall dürfen die Geräte an die 230 Volt Netzspannung angeklemt werden!

Die Spannungsquelle darf an beide Geräte nur im spannungsfreien Zustand angeschlossen werden. Ziehen Sie dazu den Netzstecker des Netzteils oder sehen Sie einen Schalter in der Zuführung der Betriebsspannung vor.

Bei Kurzschlüssen in der Versorgungsspannungszuführung besteht Brandgefahr! Beachten Sie unbedingt den Anschlußplan! Bei Verpolung der Versorgungsspannung kann der Steuercomputer oder das P/E Board zerstört werden.

Besonders in trockener Luft kann sich der menschliche Körper elektrostatisch aufladen. Beim Kontakt mit leitenden Gegenständen baut sich diese Ladung mit einem kleinen Funken ab. Solche Entladungen beim Berühren elektronischer Bauelemente können diese zerstören. Vor dem Hantieren mit den Geräten sollten Sie einen großen, geerdeten Gegenstand berühren (z.B.: ein PC-Metallgehäuse, eine Wasserleitung oder ein Heizungsrohr), um eventuelle Aufladungen abzubauen.

Leistungsmerkmale

C-Control MICRO, MICRO PCB

Dieser in BASIC programmierbare Computer ist zu großen Teilen kompatibel zur C-Control I in den Ausführungen M und Unit.

Die Ausführung MICRO und MICRO PCB sind für den Einsatz in miniaturisierten Anwendungen vorgesehen und haben deswegen eine reduzierte Anzahl von Ports, die aber (auch innerhalb eines Programms) wechselnde Funktionen haben können.

Internal timer, tone generation, frequency measurement

• **Timer**

The internal 20-millisecond timer can be read through the pre-defined identifier TIMER. The timer runs freely and cannot be set. It accepts a maximum value of 59 and then rolls on to 0.

x = Timer is permitted,

Timer = x is not permitted!

The timer is set back when the system variable SECOND is described.

With the timer, you can measure the duration of occurrences (duration of key pressure, for example).

• **Output of tones with BEEP**

The C-control BASIC control computer can output a square wave at almost every one of its digital ports which can be made audible by means of a sound converter.

BEEP tone t, port identifier

Constants or terms can be used for the first two parameters. Tone determines the pitch thereby; values of between 1 and 255 are permitted.

t determines the duration of the tone. The unit for the time statement is 20 milliseconds.

The command 'Port' states the digital port on which output is to take place. The available ports are PORT1, PORT2, PORT4, PORT5, PORT6.

Although other ports are accepted by the development environment (to reserve the compatibility with C-control I), their use leads to absolutely uncontrollable behaviour of the MICRO

define TONPORT port[5]

BEEP 168, 10, TONPORT

outputs for $10 \times 20 = 200$ milliseconds, that is, a tone to port 5.

• **Frequency measurement by means of the function FREQ**

PORT4 (and only this, otherwise no other) can, alongside its other functions as the counter of occurrences, be used with adjustable peak time. Measurement takes place constantly in the background parallel to the processing of the BASIC program. The system variable FREQ has two different meanings.

Handhabung

Programmer/Evaluation Board

Anschluss der C-Control Computer

Stecken Sie einen C-Control Computer der Ausführung MICRO in den dafür vorgesehenen IC-Sockel.

Achten Sie dabei unbedingt auf richtige Polarität. Die Kerbe am Gehäuse des IC ist neben dem Kondensator C3.

Wollen Sie die Ausführung MICRO PCB programmieren, so schließen sie die Anschlussdrähte entsprechend der Farbkennzeichnung an der Klemmleiste an.

ACHTUNG:

Es dürfen niemals 2 C-Control MICRO gleichzeitig gesteckt und angeschlossen sein.

Verbindung zum PC

Verbinden Sie die serielle Schnittstelle Ihres PC mit der Schnittstellenbuchse auf dem P/E Board.

Verwenden Sie dazu das mitgelieferte Schnittstellenkabel.

Stromversorgung

Versorgen Sie das P/E Board mit Strom aus einer 9V Batterie oder einem Steckernetzteil. Wenn Sie ein Steckernetzteil verwenden, so überzeugen Sie sich unbedingt von der richtigen Polarität (minus muss außen am Stecker anliegen) und dem richtigen Spannungsbereich (8 bis 12V DC)

Das P/E Board ist jetzt betriebsbereit.

Bedienelemente

Zum Download eines Anwenderprogramms aus der Entwicklungsumgebung **halten Sie den Taster "DOWNLOAD" gedrückt** und schieben den Schiebeschalter in die Position ON (in Richtung des Schnittstellensteckers)

Die Bereitschaft für den download wird mit der gelben LED (L5) angezeigt.

Port 1 und 2 haben jetzt Schnittstellenfunktion, weshalb LED L1 und L2 leuchten

Sie können jetzt ein Programm in die C-Control laden. Eine schrittweise Erläuterung hierzu finden Sie im Kapitel "Erste Inbetriebnahme".

Ist der Download beendet wird die gelbe LED deaktiviert und das Anwenderprogramm gestartet.

Wollen Sie eine bereits in den Controller programmierte Anwendung starten, so schieben sie den Schiebeschalter in die Position ON **ohne den Taster zu drücken**.

Das Programm startet bei Anlegen der Betriebsspannung (Power On Reset), eine Resettaste gibt es nicht.

The definitions must always be positioned at the end of a program after the END command as the data are stored seamlessly after the previous code bytes in the EEPROM memory chip. Program processing may never run through the table data as the data would otherwise be interpreted as BASIC commands which would lead with certainty to the system crashing. Access to the table data by means of the command

LOOKTAB tablename,index,variable

tablename designates a valid table. For the index, any desired term can be taken and the variable designates the memory cell in which the result is to be stored. The value of the index term calculated may not be negative and amount to a maximum of N-1 when the indexed table has N entries. It gives the index the value 0, in this manner CO is stored in the variable stated, for index equals 1 CI and so on. The following example outputs the content of table serially

```
DEFINE value WORD
```

```
DEFINE i BYTE
```

```
FOR i = 0 to 3
```

```
LOOKTAB mytab,i,value
```

```
PRINT "mytab["; i; "]="; value
```

```
NEXT
```

```
END
```

```
TABLE mytab 12 -20 0 1000
```

```
TABEND
```

The following should appear on the screen of the terminal program:

```
mytab[0]=12
```

```
mytab[1]=-20
```

```
mytab[2]=0
```

```
mytab[3]=1000
```

Tables are shown to be particularly useful in converting A/D values to genuine physical volumes. A conversion table then has 256 entries as a rule. The measured A/D value then enters the determination of the physical volume as a table index.

The real-time clock

To read and set the internal real-time clock, the following global variables have been defined:

HOURL (0...23)

C-Control Computer MICRO / MICRO PCB

Das P/E Board stellt die Betriebsspannungen für den Betrieb des C-Control Computers auf dem Board zur Verfügung.

Betreiben sie die Computer in einer anderen Hardwareumgebung, so müssen Sie eine stabile Betriebsspannung zur Verfügung stellen.

Ausführung Micro

Dies ist die "nackte" Ausführung des Computers als Chip, mit dem geladenen Betriebssystem. Die Pinbelegung sehen Sie in nachfolgender Grafik. Die MICRO benötigt eine stabile Stromversorgung 5V (+- 10%) die einen Strom von mindestens 10mA liefern muss. Ströme durch angeschlossene Bauteile (z.B. LEDs) sind dabei nicht berücksichtigt.

Für die Programmierung des Computers oder den Betrieb einer Anwendung auf dem P/E Board stecken Sie die C-Control MICRO in den dafür vorgesehenen Sockel bevor eine Verbindung zu anderen Geräten (z.B. dem PC) hergestellt wird und bevor die Betriebsspannung angelegt wird.

Die PINs der C-Control Micro haben folgende Funktionen:

PIN 1 - VDD +5V	PIN 8 - GND
PIN 2 - P6/AD6/BEEP	PIN 7 - P1/AD1/BEEP/TXD
PIN 3 - P5/AD5/BEEP	PIN 6 - P2/AD2/BEEP/RXD
PIN 4 - P4/BEEP/FREQ	PIN 5 - P3 INPUT

Ausführung MICRO PCB

Dies ist die komfortable Ausführung für den direkten Anschluss an z.B. LEDs oder Taster wie sie wohl hauptsächlich im Bereich Modellbau Verwendung findet.

Die MICRO PCB benötigt eine stabile Stromversorgung 5V (+- 10%) die einen Strom von mindestens 10mA liefern muss. Ströme durch angeschlossene Bauteile (z.B. LEDs) sind dabei nicht berücksichtigt. Zusätzlich ist auf der Platine ein Spannungsregler installiert, der über einen separaten Anschluss Eingangsspannungen von 8V bis max.12V zulässt.

Verwenden Sie diese Option, so gelten folgende Richtlinien.

- Es darf keine Spannung an den dafür vorgesehenen Anschlüssen der 5V Versorgung zugeführt werden.
- Die dann an den Anschlüssen der 5V Betriebsspannung anliegende Spannung (5V) kann für eigene Zwecke verwendet werden (z.B. Pullup Widerstände, andere Baugruppen)
- Die Stromaufnahme aller über den Regler versorgten Komponenten darf 50mA nicht überschreiten

PRINT

• Byte-wise communication through the serial interface

While PRINT transmits or expects short character strings for the presentation of numeric value, it can be desirable to transmit individual bytes serially. For this, CCBASIC offers the commands PUT and GET.

PUT term

transmits the calculated value of a term.

GET variable

waits for a serially received byte and then stores the value in the variables stated. Please observe the fact that GET will wait endlessly if no byte is received. Apart from this, the system clock RTC waits during the waiting time.

• Further interface commands and functions

BAUD 9600

BAUD 19200

switches, for example, the transmitter and receiver to the corresponding rate and is necessary once to initialise the serial interface.

The other interface parameters – 8-data bits, no parity bit, 1 stop bit – are fixed and cannot be altered.

CAUTION.

Conditioned by the RC oscillator, the actual baud rate is always inexact. Although the baud rate generator is calibrated during the BASIC download, the exactitude is possibly not sufficient for operation with 19200 baud (above all if the RC oscillator drifts or the operating current fluctuates) so that calibration of the oscillator is necessary.

The program TOOL_19200_CALIBRATE.BAS will help you with this.

Port commands

• The switch-over command TOG

In principle, access to the ports of the control computer takes place as for variables. To switch on a digital port, you write

P=I and **P=O** to switch it off.

Es ist ein kleines Beispiel, das den Zugriff auf die Ports der C-Control MICRO demonstriert.

Im Editor der Entwicklungsumgebung sehen Sie den Quelltext des Programms mit zusätzlichen Erläuterungen zur Funktion.

Download in die C-CONTROL MICRO

Zum Download eines Anwenderprogramms aus der Entwicklungsumgebung **halten Sie den Taster "DOWNLOAD" gedrückt** und schieben den Schiebeschalter in die Position ON (in Richtung des Schnittstellensteckers). Die Bereitschaft für den Download wird mit der gelben LED (L5) angezeigt.

Sie können jetzt ein Programm in die C-Control laden. Aktivieren Sie den Download im Menü ENTWICKLUNG > IN C-CONTROL LADEN

Ein Fenster zeigt Beginn und Ende des Downloads. Schließen Sie dieses Fenster nach dem Download, und schalten Sie den Schiebeschalter zunächst in die OFF Position und nach ca. 2s wieder in die Position ON, um das Beispielprogramm zu starten.

ACHTUNG:

Verwenden Sie Port 3 niemals in Anwendungen bei welchen nicht sichergestellt ist, dass Port 3 beim Anlegen der Betriebsspannung log. high ist, da sonst der Download-Modus aktiv ist und das Anwenderprogramm nicht läuft.

Entfernen Sie JP2, wenn es Probleme beim Download gibt (normal nicht erforderlich)

Beispiele

Laden Sie die Beispiele um sich mit der Programmiersprache vertraut zu machen. Prinzipielle Erläuterungen zu CCBASIC finden sie im folgenden Kapitel.

Die Programmiersprache CCBASIC

Das Konzept der C-Control MICRO/MICRO PCB

Die MICRO Versionen der C-Control sind dafür konzipiert kleinere Steuerfunktionen mit minimalem Bauteilaufwand und auf kleinstem Raum zu realisieren. Sechs Ports die (auch innerhalb eines Programms) mehrere verschiedene Funktionen haben können, lassen eine vielseitige Verwendung zu. Abgestimmt auf den Einsatzzweck ist auch das Betriebssystem, das aus dem KERNEL besteht und dem Programmierer bezüglich der C-Control M/UNIT eine nur leicht reduzierte Funktionalität bietet. Für besondere Anforderungen stellt Conrad Electronic Betriebssystemerweiterungen und Treiber zur Verfügung die allerdings den nutzbaren BASIC Programmspeicher reduzieren.

Ohne zusätzliche Module sind 2kB BASIC Speicher nutzbar, was auch für komplexe Anwendungen in diesem Anwendungsbereich ausreichend ist.

RETURN

- **Program end**

END

If the control computer reaches the END instruction in the course of the program processing, the program processing is terminated. The system then remains in an inactive condition.

- **Delay in the program flow**

The instruction

WAIT conditionterm interrupts execution of the program until the calculation of the conditionterm has a value not equal to 0.

define key port[1]

...

WAIT key

In this example, the program waits until a HIGH-level (= logical 1) is read from the digital port 1.

The PAUSE command interrupts execution of the program for a certain time. The calculated value of the parameter term goes as a multiplication factor with the basic unit of 20 milliseconds into the determination of the pause period.

PAUSE term

For example through the command

PAUSE 50

execution of the program is interrupted for approx. $50 \cdot 20$ milliseconds = 1 second. The maximum deviation in time of the actual pause from the given value amounts thereby principle-conditioned to + 20 milliseconds.

Communication through the serial interface

Port1 (TXD) and Port2 (RXD) can be used as serial interfaces if they have been initialised previously by means of the BAUD instruction.

RXD and TXD can also take on other functions within the program if the environment hardware supports this.

Digitalports

Bei der C-Control MICRO können alle Ports die Funktion eines Digitalports annehmen. Die verfügbaren Ports sind PORT1, PORT2, PORT3, PORT4, PORT5, PORT6. Andere Ports werden von der Entwicklungsumgebung zwar akzeptiert (um die Kompatibilität zu C-Control I zu wahren), jedoch führt deren Verwendung zu absolut unkontrollierbarem Verhalten der MICRO

Benutzung eines Digitalports als Eingang

Digitaleingänge werden zur Abfrage von Schaltzuständen verwendet. Wird ein Digitalport als Eingang benutzt, führt er im unbeschalteten Zustand undefinierten Pegel der z.B. mit einem PULLUP-Widerstand festgelegt werden muss. Ist beispielsweise ein Reedkontakt an diesem Port angeschlossen, wird bei offenem Schalter eine logische Eins („wahr“) vom Port gelesen, bei geschlossenem Schalter eine logische Null („falsch“).

Achten Sie bitte unbedingt darauf, daß je nach Beschaltung des Ports und der logischen Aussage, die Ihr Programm beinhalten soll, der eingelesene Wert eventuell invertiert werden muß (NOT-Operator, siehe Befehls- Beschreibung)!

ACHTUNG:

Verwenden Sie Port 3 niemals in Anwendungen bei welchen nicht sichergestellt ist, dass Port 3 beim Anlegen der Betriebsspannung log. high ist, da sonst der Download-Modus aktiv ist und das Anwenderprogramm nicht läuft.

Benutzung eines Digitalports als Ausgang

Wird ein Digitalport als Ausgang verwendet, können daran nachfolgende ICs, Transistoren oder Low-Current-Leuchtdioden direkt betrieben werden. Der maximal zulässige Laststrom beträgt 10 mA. In jedem Fall ist eine ausreichende Strombegrenzung, zum Beispiel durch einen Widerstand, zu gewährleisten, da es sonst zur Zerstörung des Mikrocontrollers kommen kann!

Innerhalb des Mikrocontrollers erfolgt die interne Beschaltung eines Digitalports als Ausgang oder Eingang beim ersten Ausführen des Anwenderprogramms. Nach dem Zuschalten der Betriebsspannung oder nach einem Reset verhalten sich alle Digitalports zunächst elektrisch als Eingang, sie führen also über einen Pullup-Widerstand High-Pegel.

Analogports

Bei der C-Control MICRO können nicht alle Ports die Funktion eines Analogports annehmen.

Die verfügbaren Analogports sind PORT1, PORT2, PORT5, PORT6.

Andere Analogports werden von der Entwicklungsumgebung zwar akzeptiert (um die Kompatibilität zu C-Control I zu wahren), jedoch führt deren Verwendung zu absolut unkontrollierbarem Verhalten der MICRO

FOR loops can be nested with one another. The depth of the nesting is only limited by the memory location necessary for the loop variable.

```
FOR variable = beginning TO end1
  FOR variable = beginning TO end2
    FOR variable = beginning TO end3
      .....
    NEXT
  NEXT
NEXT
```

Each FOR loop may only run through its own NEXT instruction in the course of the program. Although the following source text can be compiled and loaded into the control computer, it may not function perhaps as expected:

```
FOR v1 = beginning1 TO end1
  ...
  GOTO anothernext
...
NEXT
FOR v2 = beginning 2 TO end2
  ...
#anothernext
NEXT
```

Pay attention also to the value range of the loop variable and end term!

```
DEFINE v BYTE
FOR v = 1 TO 1000
  ...
NEXT
```

becomes an endless loop as v as byte variable can never reach the value 1000 but overruns after 255 to 0 again.

• Conditional execution

IF condition THEN instruction1

The IF...THEN...construction enables adaptation of the program flow to the conditions for the run-time of the program. Any desired term is to be inserted as a condition. If its calculation results in a value not equal to 0, then the condition is regarded as having been fulfilled and instruction1 will be carried out. The whole IF...THEN...-

Programmierung des C-Control Computers

CCBASIC ist der BASIC-Dialekt, der zur Programmierung des C-Control BASIC Steuercomputers verwendet wird. Die Syntax entspricht in etwa der des Standard-BASIC. Bei einigen Befehlen gibt es Abweichungen oder Erweiterungen, die speziell auf die Hardware des Steuercomputers zugeschnitten sind.

Das CCBASIC der MICRO ist im Wesentlichen kompatibel zu anderen Versionen der C-Control I Computer, jedoch grundsätzlich auf 8Bit Operationen/Anweisungen beschränkt.

Die Beispiele zum Erlernen von CC-BASIC

Auf der CD finden sie kleine Beispiele, die Ihnen schrittweise den Gebrauch der BASIC-Befehle erläutern.

Diese Beispiele sind erfordern teilweise eine Verbindung zum PC, da sie das "Hyperterminal"

- Sie finden es im Windows unter ->Programme ->Zubehör ->Kommunikation - als Ausgabemittel benutzen.

Starten Sie die C-CONTROL IDE , laden Sie ein BASIC-Programm aus dem Ordner "BASIC". Wenn Sie das Programm laden wollen müssen Sie die das Hyperterminal schließen, da die IDE sonst eine belegte Schnittstelle vorfindet.

Starten Sie das Hyperterminal in der Konfiguration 9600 Baud, 8n1. Es ist jetzt bereit für die Ausgaben, die C-Control erzeugt.

Was ist ein Programm?

Ein Programm ist die Beschreibung eines Informationsverarbeitungsprozesses. Im Laufe eines solchen Prozesses wird aus einer Menge von variablen oder konstanten Eingangswerten eine Menge von Ausgangswerten berechnet. Die Ausgangswerte sind entweder selbst Ziel der Informationsgewinnung oder dienen mittelbar zur Reaktion auf die Eingangswerte. Neben den eigentlichen Berechnungen kann ein Programm Anweisungen zum Zugriff auf die Hardware des Computers oder zur Steuerung des Programmflusses enthalten.

Ein BASIC-Programm besteht aus mehreren Zeilen sogenannten Quelltextes. Dabei enthält jede Zeile eine oder mehrere Rechen- oder Steueranweisung. Außer diesen Anweisungen selbst bestimmt ihre Reihenfolge ganz wesentlich die eingangs beschriebene Informationsverarbeitung. Die Ausführung der den Anweisungen entsprechenden Operationen durch den Steuercomputer erfolgt sequentiell, also nacheinander. Eine Folge von Programmanweisungen mit einem bestimmten Ziel nennt man auch Algorithmus. Daten sind die Objekte des Informationsverarbeitungspro-

operation). The logical operators can, apart from for the formulation of conditions (usually in connection with relational operations), also be used for binary byte manipulations.

• **Shift operators:**

SHL (shift to the left), SHR (shift to the right) are used for the bit-wise shifting of binary patterns in byte or word variables. The value to be shifted stands to the left of the operator, to the right, the number of the bits by which the shift is to be made. In the case of shifting to the left, each individual shift corresponds to multiplication by 2, for shifting to the right, division by 2.

a = 10 SHL 3 corresponds therefore to: $a=10*2*2*2$ and gives, for example, for 'a' a value of 80.

Mathematical functions and commands

The arguments x and y, depending on the function or command, are always terms (see above for definition).

• **The random function RAND**

supplies the next integer random value of the pseudo random generators. Random numbers are generated by means of a simple method and cannot, of course, be compared with a genuine random number but, for most purposes, they are sufficient. The value of RAND ranges from 0 to 255.

By means of clever manipulation, however, you can set any desired value ranges (e.g. for a Lotto numbers generator).

A=(rand and 31)+1 limits the values to 1 to 32
A=A+ (rand and 14) modifies the range to 1 to 46
A=A+(rand and 3) modifies the range to 1 to 49

Subtraction is not permitted as the values could become negative.

This version is more straightforward but has an incalculable program run-time:

```
#loop
x=rand
If (x=0) or (x>49) then goto loop
```

Ranking order of operators and function references

For the calculation of terms with operators and functions, their ranking order is of decisive significance. Part expressions with operators of high ranking are calculated with a lower rank (compare arithmetic rule: "multiplication has precedence over addi-

Variablen und Konstanten

Variablen und Konstanten sind Objekte des Informationsverarbeitungsprozesses. In CCBASIC speichern beide einen numerischen Wert. Während der Wert einer Konstante einmal angegeben wird und dann unverändert bleibt, kann sich der Wert einer Variablen im Lauf des Programms beliebig oft ändern. Konstanten können in CCBASIC in dezimaler, hexadezimaler und binärer Form angegeben werden. Die Syntax für Hexadezimal- und Binärzahlen sei hier am Beispiel der Zahl 46 (dezimal) gezeigt:

&H2E
&B101110

Außerdem können per DEFINE-Zeilen (siehe unten) symbolische Konstanten vereinbart werden. Auf Variablen wird stets über ihren Bezeichner zugegriffen. Dieser Bezeichner muß vor der ersten Verwendung der Variable im Programm in einer DEFINE-Zeile definiert werden.

Label

Label markieren bestimmte Punkte in der Folge der Programmoperationen. Label sind Ziele von Sprungoperation innerhalb eines Algorithmus. In CCBASIC stehen Label am Anfang einer Zeile und beginnen stets mit einem Doppelkreuz, dann folgt - ohne Leerzeichen - der Bezeichner des Labels.

Das Beispiel zeigt die Definition des Labels „Label1“ und die Verwendung in einem Sprungbefehl:

#label1 . . .
GOTO label1

Terme

Ein Term ergibt sofort (als Variable oder Konstante) oder durch Berechnung einen bestimmten Wert. Terme sind Teile von Anweisungen und stehen beispielsweise bei der Zuweisung eines Wertes an eine Variable rechts des Zuweisungszeichens „=“. Terme werden durch Kombinationen von Operanden und Operatoren gebildet.

a +b=(x - 13) * 10

Operanden und Operatoren

Ein Operand ist in der Grundform entweder eine Konstante, eine Variable oder ein Funktionsaufruf, kann aber auch selbst wieder ein aus Operanden und Operatoren zusammengesetzter Term sein. Operatoren bezeichnen Rechenoperationen, die mit den umstehenden Operanden auszuführen sind. Dabei gibt es eine definierte Rangfolge der Operatoren (siehe Befehlsbeschreibung), die die Reihenfolge der Berechnungen bestimmt.

Values from 1 to 24 (24 Bytes) are permitted for nr.

DEFINE identifier WORD[nr]

may be permitted by the compiler but not used as the MICRO is strictly byte-oriented.

The use of WORD variables can lead to uncontrolled behaviour of the MICRO.

If for byte definitions the cell statement [nr] is left out, the compiler will take over the division of the storage area.

DEFINE a BYTE
DEFINE c BYTE

The automatic division of the variables in the memory by the compiler begins at cell number 1. For the definition of further bits and bytes with statement of the cell number, it is necessary to watch out for undesired overlapping. An already defined variable identifier may not be defined for a second time.

Definition of digital ports

Ports as also variables are accessed in CCBASIC,. Here again, each port used has to be previously defined.

- **Definition of one of the 6 digital ports**

DEFINE identifier PORT[nr]

CAUTION:

For the C-control MICRO, all ports can accept the function of a digital port.

The ports available are PORT1, PORT2, PORT3, PORT4, PORT5, PORT6.

Although other ports are accepted by the development environment (to preserve the compatibility to C-control I), their use leads to absolutely uncontrollable behaviour of the MICRO.

DEFINE identifier BYTEPORT[nr]

DEFINE identifier WORDPORT[nr]

can be permitted by the compiler but not used as the MICRO is strictly byte-oriented.

Their use can lead to uncontrolled behaviour of the MICRO.

Probleme sogar Grundvoraussetzung. Anweisungen zur Steuerung des Programmflusses bestehen aus einem oder mehreren reservierten Worten und erfordern in jeweils spezieller Weise eventuell weitere Angaben.

```
GOTO label1  
IF a > b THEN GOSUBlabel2  
FOR i = 0 TO 10 STEP 2  
.._  
NEXT
```

Compileranweisungen

Zusätzlich zu den Programmanweisungen enthält ein CCBASIC-Quelltext Compileranweisungen, die zum Beispiel zum Anlegen von Datenblocks (Tabellen) oder zur Definition von Variablen- und Konstanten dienen.

Für Compileranweisungen gilt die Doppelpunktregel zum Trennen mehrerer Anweisungen in einer Zeile nicht. Es darf jeweils nur eine Compileranweisung in einer Zeile stehen. Die DEFINE-Anweisung ist eine Compileranweisung.

Definition symbolischer Konstanten

Es ist guter Programmierstil, statt „magischer“ Zahlen im Programm

```
IF x > 123 THEN GOTO alarm
```

besser symbolische Konstanten zu verwenden. Durch Vergabe signifikanter Bezeichner für Konstanten erhöht sich die Lesbarkeit des Quelltextes. Wenn alle Konstanten global definiert werden, ist ein Programm auch leichter zu warten. Das gilt besonders, wenn ein und dieselbe Konstante mehrmals im Programm benötigt wird.

Die Definition einer symbolischen Konstanten erfolgt wie folgt:

```
DEFINE bezeichner wert
```

Dabei ist wert entweder eine dezimale, hexadezimale oder binäre Zahl. So sollte das Beispiel zuvor besser

```
DEFINE limit 123  
...  
IF x > limit THEN GOTO alarm
```

lauten.

Definition von Variablen

Der C-Control/BASIC MICRO Steuercomputer stellt 24 Byte-Speicherzellen seines internen Speichers (RAM) dem Anwender zur Verwendung in seinen Programmen zur Verfügung. In diesem Speicherbereich werden alle Variablen eines BASIC-Pro-

Assignments

The assignment is the simplest form of a program instruction. After the identifier of a variable to which a value is to be assigned, the assignment symbol "=" follows and then a term which determines the value to be assigned. An assignment thus corresponds to a simple mathematical formula.

```
a = 10  
b = x - y  
c = (a*a + b*b)
```

Commands

Alongside simple assignments, commands are instruction for the execution of program operations by the C-control/BASIC control computer. Commands always begin with a reserved value. Some commands expect one or more parameters for precise specification of the program operation to be executed. These parameters are listed after the command identifier and a blank, and thereby separated by a comma (exception PRINT, see general view of commands). Contrary to the arguments for calling up a function, the command parameters are not positioned outside round brackets.

```
PAUSE 100  
BEEP 140,50,BEEPER
```

Instruction for controlling the program flow

These instructions allow the order of the in fact strictly sequentially processed program operations to be controlled and adapted to the input values of the information processing process. They offer a high degree of flexibility for algorithm formulation and are indeed a basic prerequisite for the solution of some technical application problems. Instructions for the control of the program flow consist of one or more reserved values and possibly require further information each in a special manner.

```
GOTO label1  
IF a > b THEN GOSUBlabel2  
FOR i = 0 TO 10 STEP 2
```

```
.._  
NEXT
```

Compiler instructions

Additionally to the program instructions, a CCBASIC source text contains compiler instructions which, for example, serve the creation of data blocks (tables) or the definition of variables and constants.

The colon rule does not apply to compiler instructions for the separation of several instruction in one line. There may be only one compiler instruction in one line. The DEFINE instruction is a compiler instruction.

Definition von Digitalports

In CCBASIC wird auf Ports wie auf Variablen zugegriffen. Auch hier muß jeder verwendete Port zuvor definiert sein.

- **Definition eines der 6 Digitalports:**

DEFINE bezeichner PORT[nr]

ACHTUNG:

Bei der C-Control MICRO können alle Ports die Funktion eines Digitalports annehmen.

Die verfügbaren Ports sind PORT1, PORT2, PORT3, PORT4, PORT5, PORT6.

Andere Ports werden von der Entwicklungsumgebung zwar akzeptiert (um die Kompatibilität zu C-Control I zu wahren), jedoch führt deren Verwendung zu absolut unkontrollierbarem Verhalten der MICRO

DEFINE bezeichner BYTEPORT[nr]

DEFINE bezeichner WORDPORT[nr]

werden vom Compiler zwar zugelassen dürfen, aber nicht verwendet werden, da die MICRO strikt Byte orientiert ist.

Ihre Verwendung kann zu unkontrolliertem Verhalten der MICRO führen

Definition von Analogports

In CCBASIC wird auf Ports wie auf Variablen zugegriffen. Auch hier muß jeder verwendete Port zuvor definiert sein.

- **Definition eines der 4 A/D-Ports:**

DEFINE bezeichner AD[nr]

Dabei sind für nr Werte von 1,2,5 oder 6 zulässig.

ACHTUNG:

Andere Ports werden von der Entwicklungsumgebung zwar akzeptiert (um die Kompatibilität zu C-Control I zu wahren), jedoch führt deren Verwendung zu absolut unkontrollierbarem Verhalten der MICRO

a = 255 + 1

results, for example, for 'a' in the value 0 and not 256 if 'a' only represents one byte!

Basic Elements of CCBASIC

General

Each program line contains one or several instructions which are separated by colons .:

Line numbers, as customary in former BASIC dialects, are not necessary. If line numbers are used anyway, they can be used as a branch target.

10 . . .

GOTO 10

The numbers have no influence on the order of the program operations over and above that. If, for example, in the source text a line 100 follows a line numbered 200, the line 200 is processed before the 100 nevertheless.

Commentaries can be taken up in the source text to explain the written program and increase its readability and maintainability. A commentary in CCBASIC always begins with a high comma ' and explains text in the rest of which does not belong to the program.

a = b + c ' . . . Commentary . . .

Identifiers

Identifiers are program elements made up of alphanumeric symbols (A to Z, 0 to 9) which designate objects such as variables and constants in the manner determined by the programmer. Label names and the so-called "reserved words" are also identifiers. No differentiation is made between capital and small letters. An identifier always begins with a letter or with an underline. Blanks are not allowed within an identifier.

Variables and constants

Variables and constants are objects of the information processing process.

Both store a numerical value in CCBASIC. While the value of a constant is stated once and then remains unaltered, the value of a variable can be altered in the course of the program as often as desired. Constants can be stated in CCBASIC in decimal, hexadecimal and binary form. The syntax for hexadecimal and binary figures is shown here by the example of the number 46 (decimal):

&H2E

&B101110

• **Die Zufallsfunktion RAND**

liefert den nächsten Integer-Zufallswert des Pseudo-Zufallsgenerators. Die Zufallszahlen werden mit einem einfachen Verfahren erzeugt und sind mit einer echten Zufallszahl natürlich nicht zu vergleichen, aber für die meisten Zwecke ist das ausreichend. Der Wertebereich von RAND ist 0 bis 255. Durch geschickte Manipulation können Sie aber beliebige Wertebereiche (z.B. für einen Lottozahlengenerator) einstellen.

- A=(rand and 31)+1** begrenzt die Werte auf 1 bis 32
- A=A+ (rand and 14)** modifiziert den Bereich auf 1 bis 46
- A=A+(rand and 3)** modifiziert den Bereich auf 1 bis 49

Subtraktionen sind nicht zulässig, da die Werte negativ werden könnten.

Diese Version ist einfacher, hat aber eine unberechenbare Programmlaufzeit:

```
#loop
x=rand
If (x=0) or (x>49) then goto loop
```

Rangfolge von Operatoren und Funktionsaufrufen

Bei der Berechnung von Termen mit Operatoren und Funktionen ist deren Rangfolge von entscheidender Bedeutung. Teilausdrücke mit Operatoren von hohem Rang werden vor denen mit einem niedrigerem Rang berechnet (vergleiche Rechenregel: „Punktrechnung vor Strichrechnung“). Bei gleichrangigen Operatoren erfolgt die Berechnung von links nach rechts. Wie in der Mathematik kann jedoch durch Klammersetzung zusätzlich Einfluß auf die Berechnungsreihenfolge genommen werden. CCBASIC unterstützt maximal 3 Klammerebenen. Im Sinne der Übersichtlichkeit eines Programmes sollten jedoch „wilde“ Klammersausdrücke vermieden und komplexe Berechnungen auf mehrere BASIC-Zeilen aufgeteilt werden.

Die folgende Liste zeigt die CCBASIC Operatorenrangfolge:

RANG	OPERATOREN
9	()
8	Funktionsaufrufe
6	* / MOD SHL SHR
5	+ -
4	> >= < <= = <>
3	NOT
2	AND NAND
1	OR NOR XOR

most cases, active sensors will be applied here in order to boost the signal of the actual sensor element and meet the demands on resolution, relative accuracy and drift behaviour.

Frequency Input

With the C-Control MICRO, only Port4 can serve as frequency input. FREQ2 is accepted by the IDE (in order to maintain the compatibility to C-Control I), but has a different function in the MICRO.

Frequency Measuring on Pin FREQ

Via the Pin FREQ2, it is possible to measure frequencies of CMOS/TTL compatible square-wave signals up to approx. 12,000 Hz.

Frequency Output - BEEP

With the C-Control MICRO, all ports can assume the function of a beep port (with the exception of Port 3). The available ports are PORT1, PORT2, PORT4, PORT5, PORT6. Other ports are accepted by the development environment (in order to maintain the compatibility to C-control I), but their use leads to absolutely uncontrollable behaviour of the MICRO.

Sound Output on the BEEP Pins

Square-wave signals (0/5 V) can be issued on the BEEP pin via BEEP command (see below) that can be made audible by connecting a piezoelectric sound converter (sound converter without internal electronics between BEEP and GND).

```

FOR v1 = anfang1 TO ende1
...
  GOTO anothernext
...
NEXT
FOR v2 = anfang 2 TO ende2
...
#anothernext
NEXT

```

Achten Sie außerdem auf den Wertebereich von Schleifenvariable und ende-Term!

```

DEFINE v BYTE
FOR v = 1 TO 1000
...
NEXT

```

wird zu einer Endlosschleife, da v als Bytevariable nie den Wert 1000 erreichen kann, sondern bereits nach 255 wieder auf 0 überrollt.

- **Bedingte Ausführung**

IF bedingung THEN anweisung1

Die IF...THEN...Konstruktion ermöglicht die Anpassung des Programmflusses an Bedingungen zur Laufzeit des Programms. Als bedingung ist ein beliebiger Term einzusetzen. Ergibt dessen Berechnung einen Wert ungleich 0, dann gilt die Bedingung als erfüllt, und die anweisung1 wird ausgeführt. Die gesamte IF...THEN...-Konstruktion muß in einer Quelltextzeile stehen. Anweisungsblöcke (mehrere Anweisungen) nach THEN sind nicht zulässig.

- **Sprunganweisung**

GOTO label

Mit der GOTO-Anweisung kann der Steuercomputer veranlaßt werden, die Programmabarbeitung an einer bestimmten Stelle fortzusetzen. Als Ziel des Sprungs wird ein Label-Bezeichner angegeben. Das Sprungziel kann sich vor oder nach der GOTO-Anweisung im Quelltext befinden.

- **Aufruf und Rückkehr aus einer Unterroutine**

Der Aufruf einer Unterroutine erfolgt mit der Anweisung

GOSUB label

This use leads to uncontrollable malfunctions of the C-Control and difficult to localise errors.

System Resources of the C-Control Computer

The term "system resources" here means all internal function units derived from the properties of the micro-controller or provided by the operating system on the chip. The command overview below describes how these system resources are addressed in the BASIC program.

Realtime Clock (RTC)

An 8-bit timer clocked with 20 milliseconds runs in the background of the operating system. Its value can be read off at any time and used to create time references in the BASIC program and it is the basis for the internal realtime clock.

The RTC provides the time in hours, minutes and seconds in the corresponding identifiers.

System-induced, the accuracy of the RTC is very bad. However, you as the operator have the possibility to improve the properties through calibration. The example TOOL_CLOCK_CHECK.BAS gives you further notices about this topic.

The RTC can also be synchronised with a DCF77 receiver and a BASIC system extension. For more information, refer to the chapter "SYSTEM EXPANSIONS".

The internal memory cells for date and time can be read and described from the BASIC program. With the description of the time memory cells, the clock can thus also be set without DCF77 receipt. For program tests or in case of low demands on the accuracy, you can thus do without the DCF77 antenna or the calibration.

User Bytes

The micro-controller features altogether 240 bytes RAM. Of these, the C-Control control computer occupies the largest share for operating system functions (stack, timer, clock, temporary memory for computations, etc.). 24 bytes are available to the operator for use in BASIC programs.

The Use of these user bytes is explained in the chapter about the DEFINE command further down.

Digital Ports

With the C-Control MICRO, all ports can assume the function of a digital port.

The available ports are PORT1, PORT2, PORT3, PORT4, PORT5, PORT6.

Other ports are accepted by the development environment (in order to maintain the compatibility to C-control I), but their use leads to absolutely uncontrollable behaviour of the MICRO.

define key port[1]

...

WAIT key

In diesem Beispiel wird solange gewartet, bis vom Digitalport 1 ein HIGH-Pegel (= logisch 1) gelesen wird.

Der PAUSE Befehl unterbricht die Programmausführung für eine gewisse Zeit. Der berechnete Wert des Parameterterms geht als Multiplikationsfaktor mit der Grundeinheit von 20 Millisekunden in die Festlegung der Pausenzeit ein.

PAUSE term

Beispielsweise wird durch den Befehl

PAUSE 50

die Programmausführung für ca. 50*20 Millisekunden = 1 Sekunde unterbrochen. Die maximale Zeitabweichung der tatsächlichen Pause vom angegebenen Wert beträgt dabei prinzipbedingt + 20 Millisekunden.

Kommunikation über die serielle Schnittstelle

Port1 (TXD) und Port2 (RXD) können als serielle Schnittstelle benutzt werden, wenn sie vorher mit der BAUD-Anweisung initialisiert wurden.

Auch RXD und TXD können innerhalb des Programms andere Funktionen annehmen, wenn die Umgebungshardware dies unterstützt.

Define ADPORT1 AD[1]

Define TRANSMIT PORT[1]

-
-
-

deact TRANSMIT

a=ADPORT1

Deaktiviert die Funktion TXD des Ports und aktiviert ihn als Analog-Eingang.

Der Analogwert wird in Variable a geschrieben

• Datenausgabe

Die Datenausgabe erfolgt als Text über die serielle Schnittstelle des C-Control/BASIC Steuercomputers. Ist über ein Schnittstellenkabel zum Beispiel ein PC mit einem Terminalprogramm angeschlossen, können die ausgegebenen Daten dort angezeigt werden.

A violation of these guidelines, especially with respect to the max. power intake leads to the destruction of sensitive components on the circuit board.

The wires of the C-Control MICRO have the following functions:

RED - VDD +5V

GREEN - P6/AD6/BEEP

BROWN - P5/AD5/BEEP

GREY - P4/BEEP/FREQ

BLACK - GND

VIOLET 7 - P1/AD1/BEEP/TXD

BLUE - P2/AD2/BEEP/RXD

WHITE - P3 INPUT

Commissioning

Software Installation

Insert the enclosed CD into the drive on your PC. The selection menu that offers you different installations appears automatically after a few seconds.

1) First select > C-CONTROL I

You are now in the selection for the installation of the C-Control I devices.

2) Here select C-CONTROL/BASIC – MICRO

3) The window that appears offers the option "OPEN"

Select this option to start the installation.

4) After the installation is complete, the development environment opens.

Loading and Starting the First Program

Getting the P/E Board ready for operation

Get the P/E Board and the C-Control MICRO ready for operation as described in the chapter "Handling".

Make sure that the jumpers are set (JP1 on Pos. 1-2) and that the slider is on OFF.

Make sure that the development environment uses the right interface (selection takes place via the menu OPTIONS > ENVIRONMENT > LOADER).

Loading a program into the development environment

Select "FILE OPEN" in the menu bar. Open the file "1_EINFÜHRUNG_PORTS1.BAS" in the folder "PROGRAMME/CC1MICRO/BASIC".

This is a little example that demonstrates access to the ports of the C-Control MICRO.

In the editor of the development environment, you see the source text of the program with additional explanations about its function.

GET variable

wartet auf ein seriell empfangenes Byte und speichert den Wert dann in der angegebenen Variablen.

Beachten Sie bitte, dass GET endlos wartet, wenn kein Byte empfangen wird. Außerdem wird während der Wartezeit die Systemuhr RTC nicht weitergestellt.

• Weitere Schnittstellenbefehle und -funktionen

BAUD 9600

BAUD 19200

schaltet beispielsweise Sender und Empfänger auf die entsprechende Rate um und ist einmalig zur Initialisierung der seriellen Schnittstelle erforderlich.

Die weiteren Schnittstellenparameter - 8 Datenbits, kein Paritätsbit, 1 Stopbit –sind fest und können nicht geändert werden.

ACHTUNG:

Bedingt durch den RC-Oszillator ist die tatsächliche Baudrate grundsätzlich ungenau. Der Baudratengenerator wird zwar während dem BASIC-Download kalibriert, aber die Genauigkeit ist u.U. nicht ausreichend für den Betrieb mit 19200 Baud (vor allem wenn der RC-Oszillator driftet, oder die Betriebsspannung schwankt) sodass eine Kalibrierung des Oszillators nötig ist.

Das Programm TOOL_19200_CALIBRATE.BAS hilft Ihnen dabei.

Portbefehle

• der Umschaltbefehl TOG

Prinzipiell erfolgt der Zugriff auf die Ports des Steuercomputers wie auf Variablen. Um einen Digitalport einzuschalten, schreibt man

P=I und **P=O** ,um ihn auszuschalten.

Um den Port umzuschalten (EIN nach AUS; AUS nach EIN), kann man schreiben

P = NOT P

oder den Befehl

TOG P

Now you can load the program into C-CONTROL. A step-by-step explanation is contained in the chapter "Commissioning".

Once the download is complete, the yellow LED is deactivated and the user program starts.

If you want to start an application that is already programmed in the controller, slide the slider to the position ON **without pressing the button**.

The program starts once you apply the operating voltage (power On reset), there is no reset button.

ATTENTION!

Never use Port 3 in applications where it is not sure that Port 3 is log. high when applying the operating voltage. Otherwise, the download mode is active and the user program does not run.

Display Elements

You can use the available display elements at wish in the user program.

The LEDs are connected to the corresponding ports according to their respective designation.

meaning L3 is connected to Port 3.

When Ports 1 and 2 are not activated in their program (INPUT) or the MICRO waits for a download, the corresponding LEDs are slightly lit because they are supplied with power via the RS 232 level converter.

There is a poti for trying our the AD converter, which provides a variable voltage to the AD 6

. The voltage increases when you turn the poti counter-clockwise. A beeper enables acoustic signals.

Jumper for Configuration

So that you are not tied to the available display elements, you can separate these.

Then you can connect your own application to the post plug.

JUMPER JP1

JP1 enables the operation of the beeper together with the variable input voltage on Port6/AD6.

An example program installed into the directory "BASIC" shows you how the ports can be used during a program with different properties.

For operating the beeper together with the variable input voltage on Port6, the jumper must be put onto the position 1-2.

If you want to forego this and instead have an additional LED available, put the jumper onto position 2-3.

LOOKTAB tablename,index,variable

tablename bezeichnet eine gültige Tabelle, für index kann ein beliebiger Term stehen und die variable bezeichnet die Speicherzelle, in der das Ergebnis abgespeichert werden soll. Der berechnete Wert des index-Terms darf nicht negativ sein und maximal N-1 betragen, wenn die indizierte Tabelle N Einträge hat. Ergibt index den Wert 0, so wird CO in der angegebenen Variablen gespeichert, für index gleich 1 CI und so weiter. Folgendes Beispiel gibt den Inhalt einer Tabelle seriell aus

```
DEFINE value WORD
```

```
DEFINE i BYTE
```

```
FOR i = 0 to 3
```

```
LOOKTAB mytab,i,value
```

```
PRINT "mytab["; i; "]="; value
```

```
NEXT
```

```
END
```

```
TABLE mytab 12 -20 0 1000
```

```
TABEND
```

Am Bildschirm des Terminalprogramms sollte erscheinen

```
mytab[0]=12
```

```
mytab[1]=-20
```

```
mytab[2]=0
```

```
mytab[3]=1000
```

Besonders nützlich erweisen sich Tabellen beim Umsetzen von A/D-Werten in echte physikalische Größen. Eine Umsetzungstabelle hat dann in der Regel 256 Einträge. Der gemessene A/D-Wert geht dann als Tabellenindex in die Bestimmung der physikalischen Größe ein.

Die Echtzeituhr

Um den Stand der internen Echtzeituhr auszulesen und zu setzen, sind folgende globale Variablen definiert:

HOUR Stunde (0...23)

MINUTE Minute (0...59)

SECOND Sekunde (0...59)

You can also operate the P/E Board with a 9 V battery.

The C-Control MICRO/MICRO PCB has inputs for the optional operation with 5 V or 8 V up to 12V.

With sufficient charge, operation with four NiCd accumulator cells is also possible.

Never connect the devices to the 230 V mains voltage!

You may only connect the voltage source to both devices when this is free of power. To do so, pull the power plug of the power pack or provide a switch in the operating voltage supply.

Danger of fire in case of short circuits in the supply voltage!

Observe the connection plan under all circumstances! The wrong polarity of the supply voltage may destroy the control computer or the P/E Board.

The human body can become electrostatically charged, especially in dry rooms. In case of contact with conductive items, this charge discharges with a small spark. Such discharges when touching electronic components may destroy these. Prior to handling the devices, you should touch a large, grounded object (e.g. a metal PC casing, a water pipe or a radiator pipe) in order to discharge any possible charges.

Features

C-Control MICRO, MICRO PCB

This computer programmable in BASIC is largely compatible to the C-Control I in the versions M and Unit.

The versions MICRO and MICRO PCB are intended for application in miniaturised applications and therefore have a reduced number of ports, which may however also have different functions (also within a program).

- 8 BIT A/D converter
- Digital ports I/O
- Beep ports
- Freq. input
- Serial interface 9600/19200 Baud
- 2kB BASIC program memory
- loadable system expansions (e.g. DCF 77 synchronising of the RTC)

Operating system extensions can be loaded for maximum flexibility that do however reduce the usable BASIC program memory. The operating system extensions are provided by Conrad.

- **Timer**

Der interne 20-Millisekunden-Timer kann über den vordefinierten Bezeichner TIMER ausgelesen werden. Der Timer ist freilaufend und darf nicht gestellt werden. Er nimmt maximal den Wert 59 an und rollt danach auf 0

x = Timer ist zulässig,

Timer = x ist nicht zulässig!

Der Timer wird zurückgesetzt, wenn die Systemvariable SECOND beschrieben wird. Sie können mit dem Timer die Dauer von Ereignissen messen (Dauer eines Tastendruckes z.B.)

- **Ausgabe von Tönen mit BEEP**

Der C-Control BASIC Steuercomputer kann an fast jedem seiner Digitalports eine Rechteckschwingung ausgeben die mit einem Schallwandler hörbar gemacht werden können.

BEEP ton, t, Portbezeichner

Für die ersten zwei Parameter können Konstanten oder Terme eingesetzt werden. Dabei bestimmt ton die Tonhöhe zulässig sind Werte zwischen 1 und 255

, t bestimmt die Dauer des Tons. Die Einheit für die Zeitangaben beträgt 20 Millisekunden.

Der Befehl Port gibt den Digitalport an, auf welchem die Ausgabe erfolgen soll. Die verfügbaren Ports sind PORT1, PORT2, PORT4, PORT5, PORT6.

Andere Ports werden von der Entwicklungsumgebung zwar akzeptiert (um die Kompatibilität zu C-Control I zu wahren), jedoch führt deren Verwendung zu absolut unkontrollierbarem Verhalten der MICRO

define TONPORT port[5]

BEEP 168, 10, TONPORT

gibt also für 10*20=200 Millisekunden einen Ton an Port 5 aus.

- **Frequenzmessung mit der Funktion FREQ**

PORT4 (und nur dieser, sonst kein anderer) kann, neben seinen anderen Funktionen als Zähler von Ereignissen, mit einstellbarer Torzeit, verwendet werden. Die Messung erfolgt ständig im Hintergrund, parallel zur Abarbeitung des BASIC-Programms. Der Systemvariablen FREQ kommen zwei verschiedene Bedeutungen zu.

Each C-Control MICRO as well as the P/E Board leave our factory in perfect and function-tested condition!

For these products, Conrad Electronic grants a **warranty of 24 months**. Within this time, we will remedy possible transport damages during delivery, production flaws or breakdowns on the device free of charge.

If the performance features of the C-CONTROL MICRO or the P/E Board do not meet your individual requirements, please make use of our **14-day money back guarantee**. Send the device back to us within this time without traces of use and in its original packaging for refund or offset.

All periods valid as of the invoice respectively receipt date.

Conrad Electronic will not assume any liability for consequential damages to assets or persons resulting from the application of these products!

Service

Conrad Electronic provides a competent team of service staff for your consultation. Each inquiry will be processed as fast as possible. Special inquiries will be passed on to the development engineers of CTC.

In order to avoid unnecessary delays, however, we kindly ask you to once again study this manual, the online help of the programming software, the text and example files and, if possible, the information pages in the Internet prior to your inquiry. This often already provides the solution to a problem!

Please direct your inquiries to our technical customer service department.

In case of questions, consult our technical information service

Germany: **Tel. 0180/5 31 21 16 or 09604/40 88 47**
 Fax 09604/40 88 48
 e-mail: tkb@conrad.de
 Mon - Fri 8.00 to 18.00

Austria: **Tel. 0 72 42/20 30 60**
 Fax 0 72 42/20 30 66
 e-mail: support@conrad.at
 Mon - Thu 8.00 to 17.00
 Fri. 8.00 to 14.00

Switzerland: **Tel. 0848/80 12 88**
 Fax 0848/80 12 89
 e-mail: support@conrad.ch
 Mon - Fri 8.00 to 12.00, 13.00 to 17.00

Systemerweiterungen und Treiber

Conrad Electronic stellt für die C-Control MICRO CCBASIC Systemerweiterungen und Treiber für Zubehör zur Verfügung. Dafür ist in der MICRO die PAGE 1 und PAGE 0 am hinteren Ende des Basic-Programmspeichers vorgesehen. Jede dieser Seiten belegt 256 Bytes und verringert den Basic-Programmspeicher entsprechend, wenn sie geladen werden.

Während Erweiterungen und Treiber für die PAGE 0 zusammen mit dem BASIC-Programm geladen werden, brauchen diese Erweiterungen und Treiber für die PAGE 1 ein eigenes BASIC-Ladeprogramm, das zusammen mit Beispielen erhältlich ist.

Diese Erweiterungen und Treiber werden grundsätzlich mit der Compileranweisung **SYSCODE "ADDONS.SI9"**

eingebunden und

Der SYSCODE Befehl darf nur einmal in einem CCBASIC-Programm erscheinen und sollte am Ende, noch hinter eventuellen Tabellendefinitionen stehen.

Systemerweiterungen aktivieren in der Regel zusätzliche BASIC-Befehle während Systemtreiber spezielle Routinen für periphere Geräte enthalten und mit dem BASIC-Befehl

SYS adr

aufgerufen werden, wobei adr eine Konstante ist und die Adresse bestimmt, zu der gesprungen werden soll.

Eine genaue Beschreibung und Beispiele zur Anwendung finden Sie jeweils bei den angebotenen Erweiterungen und Treibern.

Achtung:

Um zu verhindern, dass Kunden bei Experimenten mit selbst geschriebenen Assemblerprogrammen das Betriebssystem versehentlich löschen (Flash Speicher !!) wird die Ausführung nicht authentisierter Maschinenprogramme unterbunden.

Problemlösungen

Es lässt sich kein Programm laden

- Sie haben in der IDE die falsche Schnittstelle ausgewählt
- Sie verwenden ein anderes als das gelieferte Schnittstellenkabel
- Die Batterie ist leer oder das Netzteil liefert zu wenig Spannung
- Sie haben die Anweisung zum Download nicht genau befolgt.
- Sie haben die MICRO/MICRO-PCB falsch angeschlossen
- Entfernen Sie JP2 vor dem Download (normal nicht erforderlich)

Programming the C-Control Computer	56
Examples for learning CC-BASIC	56
What is a Program?	56
Basic Elements of CCBASIC	57
General	57
Identifiers.....	57
Variables and constants	57
Labels	58
Terms	58
Operands and operators	58
Functions.....	58
Assignments	59
Commands	59
Instruction for controlling the program flow	59
Compiler instructions	59
Definition of symbolic constants	60
Definition of variables	60
Definition of digital ports	61
Definition of analogue ports	62
Mathematical and logical operators	62
Ranking order of operators and function references	63
Instructions for control of the program flow	64
Communication through the serial interface	67
Port commands.....	69
Definition and applications of data tables.....	70
The real-time clock	71
Internal timer, tone generation, frequency measurement	73
System Extensions and Drivers	75
Troubleshooting	75
Programs cannot be loaded	75
The programs can be loaded but nothing happens on "start"	76
Technical Data	78

