AVR-ChipBASIC: BASIC-Referenz

V0.61 (c) 2006 Jörg Wolfram

1 Lizenz

2 Allgemeines

Da es reichlich Schlüsselwörter gibt, kann es sein, dass diese Referenz anfangs noch unvollständig ist. Viele Hinweise findet man auch in den Beispielprogrammen.
Jedes Programm besteht aus maximal 20 Programmzeilen (1-20). Alle Schlüsselwörter ausser Funtionen können bis auf zwei Zeichen abgekürzt werden, beim Laden werden abgekürzte Schlüsselwörter mit 2 Zeichen dargestellt. Nach jedem Schlüssselwort muss ein Leerzeichen stehen. Viele Befehle blenden nicht benutzte Bits bei den Parametern aus (z.B. COLOR) oder begrenzen auf den gültigen Wertebereich (z.B. PLOT).

3 Zahlen, Variablen und Funktionen

AVR-ChipBASIC kennt nur einen Datentyp, und das sind 16Bit Integerzahlen. Dazu gibt es 26 Variablen (A-Z) und ein Array mit 16 Elementen (@), dazu aber später mehr. Zahlen können sowohl dezimal als auch in Binär- und Hexadezimalform einegeben werden, wobei bei letzteren keine negativen Werte erlaubt sind. Hexadezimalzahlen beginnen mit **$**, Binärzahlen mit der Tilde ~. Folgende Operationen sind erlaubt:

- Addition **+**
- Subtraktion **-**
- Multiplikation ***
- Division **/**
- Modulo **%**
- Und-Verknüpfung **&**
- Oder-Verknüpfung **#**
- Vergleiche (=,<>,<,<=,>,>=) liefern 0 oder 1

Dazu kommen noch öffnende und schließende Klammern und Funktionen. Funktionen können nicht abgekürzt werden und zwischen Funktionsname und öffnender Klammer darf kein Leerzeichen stehen.

ABS(	Berechnet den Absolutwert des eingeklammerten Ausdruckes
SGN(	Berechnet das Vorzeichen (-1,0,1) des eingeklammerten Ausdruckes
NOT(	invertiert den eingeklammerten Ausdruck
SQR(	zieht die Quadratwurzel aus dem eingeklammerten Ausdruck
RND(	erzeugt eine Zufallszahl zwischen 0 und dem eingeklammerten Ausdruck
DIN(	liefert den Pegelwert des angegebenen Portpins (0-7) an der parallelen Schnittstelle
TEMP(	liefert den Temperaturwert des angegebenen LM75-Sensors (0-7)
ADC(	liefert den Analogwert der Spannung des angegebenen Portpins (0-7) an der parallelen Schnittstelle
PEEK(	liest ein Datenwort aus dem optionalen Daten-EEPROM, der Klammerausdruck bestimmt die Adresse

Zusätzlich zu den normalen Variablen gibt es noch die Arrayvariable **@**. Die Indices 0 bis 15 sprechen die 16 Array-Elemente in Wortbreite an. Die Indices 128 bis 159 sprechen dasselbe Array mit 32 Elementen in Bytebreite an, wobei das LOW-Byte zuerste gespeichert ist.

4 Schlüsselwörter

4.1 Wertzuweisung

Wertzuweiseungen beginnen nicht mit einem Schlüsselwort, sondern mit einem Varaiblennamen oder Array-Ausdruck gefolgt von einem Gleichheitszeichen und einem Ausdruck. Beispiele:

- A=-4
- B=SQR(8*2)
- @(B)=RND(6)+1

4.2 Programmsteuerung

4.2.1 END

Mit dem END-Befehl wird das Programm an der aktuellen Stelle beendet. Das gleiche geschieht auch, wenn die letzte Programmzeile abgearbeitet ist.

4.2.2 GOTO

Mit dem GOTO-Befehl kann die Programmabarbeitung mit einer anderen Zeile fortgesetzt werden. Argument ist ein beliebiger Ausdruck. Beispiele:

- GOTO 2
- GO B+1

4.2.3 IF - THEN

Die bedingte Anweisung besteht aus **IF** gefolgt von einem Ausdruck. Ist das Ergebnis des Ausdrucks Null, wird zum Anfang der nächsten Zeile gesprungen, andernfalls wird die Zeile weiter abgearbeitet. Das **THEN** kan auch weggelassen werden. Beispiele:

- IF A>5 THEN A=5
- IF A>5 A=5

4.2.4 FOR - NEXT

Bei der Schleifenabarbeitung gibt es nur die Grundform **FOR A=1 TO C** ohne die Angabe der Schrittweite. Da der Stack auf 4 Einträge begrenzt ist, lassen sich nur 4 Schleifen bzw. Unterprogrammaufrufe schachteln. Beispiel:

1. FOR A=0 TO 9
2. FOR B=0 TO 9
3. PLOT A,B,3
4. NEXT :NEXT

Dieses Programm zeichnet ein weisses Rechteck in die obere linke Ecke des Bildschirmes.

4.2.5 CALL - RETURN

Im Gegensatz zu den meisten BASIC-Dialekten heisst GOSUB hier CALL. Durch das verwendete Prinzip der Abkürzungen dürfen keine zwei Schlüsselwörter mit denselben beiden Buchstaben beginnen und GOTO beginnt schon mit „GO". **CALL Expr** ruft das Unterprogramm in der durch den Ausdruck definierten Zeile auf, mit **RETURN** wird wieder zurückgesprungen. Da der Stack auf 4 Einträge begrenzt ist, lassen sich nur 4 Schleifen bzw. Unterprogrammaufrufe schachteln.

4.3 Ausgabe

4.3.1 CLS

Mit dem CLS-Befehl wird der Bildschirm gelöscht. Beim Programmstart geschieht das automatisch.

4.3.2 POS Y,X

Mit dem POS-Befehl wird der Schreibcursor an die Stelle Y,X gesetzt. Nach jedem Löschen des Bildschirms wird der Cursor auf die Position 0,0 (links oben) gesetzt

4.3.3 PRINT

Der PRINT-Befehl dient zur Ausgabe auf den Bildschirm oder auf die serielle/parallele Schnittstelle. Zusätzlich kann die Ausgabe noch formatiert werden.

"TEXT"	der Text TEXT wird ausgegeben
#Expr	Festlegung des Ausgabekanals (0=Bildschirm, 1=seriell, 2=parallel)
!Expr	Stellt das Format ein (s.u.)
Expr	gibt das Ergebnis des Ausdrucks mit dem eingestellten Format aus
;	Trenner zwischen Ausdrücken
,	Trenner zwischen Ausdrücken, Leerzeichen bis zur nächsten durch 8 teilbaren Position

Steht am Ende des PRINT-Befehls einer der beiden Trenner, wird kein Zeilenvorschub ausgeführt. Das Format ist ein Wert zwischen 0 und 255, wobei die Bits folgende Bedeutung haben:

Bit 7	0=dezimale Ausgabe, 1=hexadezimale Ausgabe
Bit 6	1 schaltet auf Großdarstellung um
Bit 4/5	Kommaposition (0-3 Nachkommastellen), nur für dezimale Ausgabe
Bit 2/3	Anzahl der ausgegebenen Ziffern (2-5), nur für dezimale Ausgabe
Bit 0/1	0=Kompakt, 1=führende Leerzeichen 2=führende Nullen 3=führende Nullen mit Vorzeichen
Bit 0	2/4 Zeichen bei hexadezimaler Ausgabe

4.3.4 EMIT

Gibt durch Komma getrennte Zeichen auf den Bildschirm/Seriell/Drucker aus, je nachdem was zuletzt eingestellt war. Es werden keine Zeichen, sondern Zahlenwerte der Zeichen erwartet. Beispiel:

- EMIT 64,$40

gibt zwei Klammeraffen aus.

4.3.5 LCHAR n

Es werden (am oberen Bildrand beginnend) n Zeichenzeilen um ein Zeichen nach links verschoben. Am rechten Rand rücken Leerzeichen nach. Beispiel:

- LCHAR 10

verschiebt die obersten 10 Zeichenzeilen um 1 Zeichen nach links.

4.4 Pseudografik

4.4.1 COLOR

Mit dem COLOR-Befehl wird die Zeichenfarbe festgelegt. Diese wird bei PLOT, BOX und FBOX sowie bei der Ausgabe von großen Zeichen (Formatbit 6 gesetzt) ausgewertet. Akzeptiert werden Werte von 0 bis 3, dabei bedeutet 0=schwarz, 1=rot,2=cyan,3=weiss.Argument ist ein beliebiger Ausdruck. Beispiele:

- COLOR 2
- CO A

4.4.2 PLOT Y,X

Mit dem PLOT-Befehl wird ein „Pixel" im Pseudografikmodus gesetzt. Dabei kann es passieren, dass benachbarte Punkte auch ihre Farbe wechseln. Beispiel:

- PLOT 3,2+X

4.4.3 BOX Y1,X1,Y2,X2

Mit dem BOX-Befehl wird ein Rechteck im Pseudografikmodus gezeichnet. Dabei kann es passieren, dass vorhandene Punkte auch ihre Farbe wechseln. Beispiel:

- BOX 1,1,10,10

4.4.4 FBOX Y1,X1,Y2,X2

Mit dem FBOX-Befehl wird ein gefülltes Rechteck im Pseudografikmodus gezeichnet. Dabei kann es passieren, dass vorhandene Punkte auch ihre Farbe wechseln. ist Y1=Y2 oder X1=X2 werden horizontale oder vertikale Linien gezeichnet. Beispiel:

- CO 1:FBOX 0,0,0,100

zeichnet eine waagerechte rote Linie am oberen Bildschirmrand, der Wert 100 wird zur Laufzeit auf den Bildschirmbereich begrenzt.

4.4.5 LPIX n

Es werden (am oberen Bildrand beginnend) n*2 Pixelzeilen um ein Pixel nach links verschoben. Bei verschiedenfarbigen Pixeln können Farbwechsel auftreten. Amrechten Rand rücken schwarze Pixel nach. Beispiel:

- LP 20

verschiebt die obersten 40 Pixelzeilen um 1 Pixel nach links.

4.5 Audio

4.5.1 NOTE n

Ein Ton mit der Tonhöhe n (Halbtonschritte ab 220 Hz aufwärts) wird ausgegeben. Bei n=0 bis 63 werden Noten ausgegeben, bei n=255 Rauschen. Beispiel:

1. FOR N=0 to 63:NOTE N
2. WAIT 5:NEXT

gibt nacheinander alle spielbaren Noten aus.

4.6 Tastatur

4.6.1 INPUT

Es können durch Kommata getrennt Zeichenketten und Variablen angegeben werden. Die Zeichenketten werden ausgegeben, die Variablen bewirken einen Eingabecursor. Falsch eingegebene Zeichen können mit der Backspace-Taste korrigiert werden. Es ist auch möglich, Ausdrücke einzugeben die dann berechnet werden. Das folgende Beispiel zeigt einen kleinen Rechner, das letzte Ergebnis ist in der Variable M gespeichert.

1. INPUT ”AUFGABE: ”,M
2. PRINT ”ERGEBNIS:”;M
3. GOTO 1

4.6.2 RKEY V

Die aktuell gedrückte Taste wird in die Variable V geschrieben. Ist keine Taste gedrückt, wird eine 0 geschrieben. Beispiel:

- RKEY P

der aktuelle Tastaturcode wird in die Variable P geschrieben.

4.6.3 WKEY V

Es wird auf einen Tastendruck gewartet und die gedrückte Taste wird in die Variable V geschrieben. Beispiel:

- WKEY P

4.7 Zeit

4.7.1 WAIT n

Mit dem WAIT-Befehl wird N*0,1 Sekunden gewartet. N kann wieder ein beliebiger Ausdruck sein. Beispiel:

- WA 10

wartet 1 Sekunde, bis das Programm fortgesetzt wird.

4.7.2 SYNC n

Mit dem SYNC-Befehl wird auf N Bildsynchronimpuse gewartet. N kann wieder ein beliebiger Ausdruck sein. Beispiel:

- SYNC 300

wartet 6 Sekunden, bis das Programm fortgesetzt wird. Das gilt nur für PAL, bei NTSC wird nur 5 Sekunden gewartet.

4.7.3 TSET n

Der interne Timer (10Hz) wird auf den Wert n gesetzt. Beispiel:

- TSET 0

setzt den Timer auf 0.

4.7.4 TGET V

Der interne Timer (10Hz) wird ausgelesen und in die Varaiable V gespeichert. Beispiel:

- TGET Z

Damit kann z.B. die Laufzeit von Programmen bestimmt werden.

4.8 Ein-/Ausgabe

4.8.1 DIR n

setzt die I/O-Richtung der 8 Portpins an der parallelen Schnittstelle. Eine 0 bedeutet Eingang, eine 1 Ausgang. Beispiel:

- DIR $F0

Pin D0-D3 werden als Eingang, D4-D7 als Ausgang konfiguriert.

4.8.2 OUT n,b

Setzen (b=1) oder Rücksetzen (b=0) des Ausganges n. Beispiel:

- OUT 7,0

Setzt D0 auf 0-Pegel.

4.9 Serielles

4.9.1 PUMP n

Schaltet die Ladungspumpe aus (n=0) oder ein (n=1). Beispiel:

- PU 1

schaltet die Ladungspumpe ein. Ist der AutoRun-Jumper gesetzt, ist die Ladungspumpe per default nach dem Start ausgeschaltet.

4.9.2 SPUT n

das Byten wird an die serielle Schnittstelle ausgegeben. Beispiel:

- SPUT 10

gibt einen Zeilenvorschub an die serielle Schnittstelle aus.

4.9.3 SGET V

Ein Zeichen von der seriellen Schnittstelle wird eingelesen und in die Varaiable V gespeichert. Beispiel:

- SGET C

5 Changelog

17.12.2006 Erste öffentliche Version (0.61)