

# Lösungen der Aufgaben

## ASCII

### Aufgabe 1

```
...
Do
    printbin 77
loop
```

### Aufgabe 2

```
dim b as byte
...
For b = 65 to 90
    printbin b
next b
end
```

### Aufgabe 3.2

```
Dim b as byte
Do
    inputbin b
    b = b - 32
    printbin b
loop
```

### Aufgabe 4

```
wait 5
printbin 72
printbin 97
printbin 108
printbin 108
printbin 111
printbin 32
wait 5
printbin 71
printbin 101
printbin 111
printbin 114
printbin 104
end
```

# Lösungen der Aufgaben

## Aufgabe 5

```
dim eingabe as byte
declare sub hallo
declare sub tschuess
...
Do
    inputbin eingabe
    if eingabe = 1 then call hallo
    if eingabe = 2 then call tschuess
loop

sub call
    printbin 72
    printbin 97
    printbin 108
    printbin 108
    printbin 111
end sub
```

Unterprogramm tschuess analog

## Aufgabe 6

```
Dim Asciiicode As Byte
...
Do
    Inputbin Asciiicode
    if Asciiicode < 90 then Asciiicode = Asciiicode + 1 else
    Asciiicode = 65
    Printbin Asciiicode
Loop
```

Beachten Sie: Es dürfen nur Großbuchstaben eingegeben werden.

# Lösungen der Aufgaben

## Variablen und Terminal

### Aufgabe 1

```
Pause = 200      <- Schreibvorgang
wait Pause      <- Lesevorgang
```

### Aufgabe 2

Die Schublade mit dem Etikett "Pause" wird geöffnet; die Zahl in dieser Schublade wird gelesen (30). Zu dieser Zahl wird 10 addiert; das Ergebnis (40) wird in derselben Schublade abgelegt (... geschrieben).

Zusammengefasst: Der Wert in der Schublade "Pause" wird um 10 erhöht.

Abbruchbedingung: pause = 100

Der Wert muss 100 sein.

Alle LEDs bei PortB gehen an.

Die Schleife wird insgesamt 9 mal durchlaufen. Beachten Sie: am Ende des ersten Schleifendurchlaufs hat die Variable Pause bereits den Wert 20.

### Aufgabe 3 (Strong Fist)

```
dim zaehler as byte
...
Zaehler = 0
Do
    do
        loop until pind.2 = 0
        zaehler = zaehler + 1    'Zähler um 1 erhöhen
        portb.0 = 1
        waitms 100
        portb.0 = 0
    loop until zaehler = 20
portb = 255
end
```

# Lösungen der Aufgaben

## Aufgabe 5

$\&B01010101 = 85$

## Aufgabe 6

Nach einer Wartezeit von Sekunden wird das Bitmuster des Nachfolgers der eingegebenen Zahl dargestellt.

## Aufgabe 7

Es dürfte nur die LED von Portb.0 angehen.

## Aufgabe 8

```
dim schwingungsdauer as byte
dim phasendauer as byte
...
inputbin schwingungsdauer
phasendauer = schwingungsdauer / 2
do
    portb.0 = 1
    waitms phasendauer
    portb.0 = 0
    waitms phasendauer
loop
```

Bemerkung: Mit Hilfe einer Zählvariable könnte man den Ton auch zeitlich begrenzen.

# Lösungen der Aufgaben

## Zeichenketten

### Aufgabe 3 und 5

```
Dim Antwort As String * 12
...
Print "Wie heißt Du? ";
Input Antwort Noecho
Print "Hallo ";
Print Antwort;
Print "! Geht es Dir gut? ";
Input Antwort Noecho
Antwort = Ucase(antwort)
If Antwort = "JA" Then Print "Schön!" ; Else Print "Schade!";
End
```

### Aufgabe 4

Funktion mit Parameter	Ergebnis
Ucase(zk)	INFO IST TOLL.
Lcase(zk)	info ist toll.
Len(zk)	14
Chr(c)	Z
Chr(68)	D
Mid(zk,1,1)	I
Mid(zk,p,1)	O
Mid(zk,5,a)	(Leerzeichen)
Mid(zk,1,4)	INFO

### Aufgabe 6

b)    laenge = len(zk)  
      b = mid(zk, laenge, 1)  
      print b

c)    b = mid(zk, 1, 3)  
      print b

## Lösungen der Aufgaben

```
d)  position = len(zk) - 2
     b = mid(zk, position, 3)
     print b
```

f)

```
Dim Text As String * 20
Dim Laenge As Byte
Dim Buchstabe As String * 1
Dim I As Byte
...
Do
  Input Text Noecho
  Laenge = Len(text)
  For I = 1 To Laenge
    Buchstabe = Mid(text , I , 1)
    Print Buchstabe;
    Wait 1
  Next I
  Print " ";
Loop
```

### Projekt Morsezeichen (verkürzt)

```
Declare Sub Lang
Declare Sub Kurz
Declare Sub Morsezeichen_a
Declare Sub Morsezeichen_s
Declare Sub Morsezeichen_o
Declare Sub Morsezeichen_g
Dim Eingabe As String * 10
Dim I As Byte
Dim L As Byte
Dim Buchstabe As String * 1
...
Do
  Input Eingabe Noecho
  L = Len(eingabe)
  For I = 1 To L
    Buchstabe = Mid(eingabe , I , 1)
    If Buchstabe = "S" Then Call Morsezeichen_s
    If Buchstabe = "A" Then Call Morsezeichen_a
    If Buchstabe = "O" Then Call Morsezeichen_o
    If Buchstabe = "G" Then Call Morsezeichen_g
  Next I
...
Next I
```

# Lösungen der Aufgaben

Loop

```
'*****  
'***** Unterprogramme *****
```

```
Sub Lang  
  Portb.0 = 1  
  Waitms 300  
  Portb.0 = 0  
  Waitms 100  
End Sub
```

```
Sub Kurz  
  Portb.0 = 1  
  Waitms 100  
  Portb.0 = 0  
  Waitms 100  
End Sub
```

```
Sub Morsezeichen_s  
  Call Kurz  
  Call Kurz  
  Call Kurz  
  Waitms 200  
End Sub
```

```
Sub Morsezeichen_o  
  Call Lang  
  Call Lang  
  Call Lang  
  Waitms 200  
End Sub
```

```
Sub Morsezeichen_g  
  Call Lang  
  Call Lang  
  Call Kurz  
  Waitms 200  
End Sub
```

...

Hinweise: Besser ist es, wenn man für dit eine Variable einführt und die restlichen Zeiten daraus berechnet. Für das Leerzeichen (Worttrenner) muss man einen entsprechenden wait-Befehl benutzen.

Eine besonders elegante Lösung besteht darin, jedem Buchstaben eine Zeichenkette für den Morsecode zuzuordnen (z. B. "klk" für den Buchstaben "r"); dadurch könnte man alle Morezeichen-Unterprogramme durch ein einziges ersetzen.

# Lösungen der Aufgaben

## LCD

### Aufgabe 2.1

```
Cls          `Kann am Anfang eines Programms nie schaden...  
lcd "Hallo!"  
Wait 3  
cls  
end
```

### Aufgabe 2.2

```
Cls  
lcd "Hallo"  
Locate 2,1  
lcd "Welt!"  
end
```

### Aufgabe 2.3

```
Dim zk as string*8  
...  
Do  
    cls  
    input zk noecho  
    lcd zk  
loop
```

### Aufgabe 3.3

```
' Miniroulette-Datei für Attiny-Platine von E. Eube,  
' G. Heinrichs und U. Ihlefeldt  
' LCDisplay  
' leicht überarbeitete Schülerlösung  
'-----
```

```
$regfile = "attiny2313.dat"  
'Attiny2313
```

```
'*****  
'***** Deklarationen *****
```

```
Dim Pause As Byte  
Dim I As Byte
```



## Lösungen der Aufgaben

```
Dim X As Byte
Dim Zz As Byte
Dim Merke As Byte
Declare Sub Intro
Declare Sub Spiel

'***** Initialisierung *****

Ddrb = &B11111111 'Port B als Ausgangsport
Ddrd = &B01101000 'D4, D5, D6 als Ausgang; Rest als Eingang
Portd = &B10001111 'Eingänge auf high legen

Cursor Off
Waitms 50

'*****
'***** Hauptprogramm *****

Pause = 50

Do
  Call Intro
  Call Spiel
  Wait 5
Loop

'*****
'***** Unterprogramme *****

Sub Intro
  Cls
  Lcd "roulette"
  Wait 3

  Cls
  Locate 1 , 1
  Lcd "start"
  Locate 2 , 1
  Lcd "-> Ta0"

  Do
  Loop Until Pind.2 = 0
End Sub

Sub Spiel
  Cls
  Lcd "123456"
  X = 0 'Anzahl der Kugelbewegungen
  I = 1 'Position
```

## Lösungen der Aufgaben

```
Zz = Rnd(30) 'Zufallszahl zwischen 0 und 29
Zz = 40 + Zz 'Zufallszahl zwischen 40 und 69
Do
  Locate 2 , I
  Lcd "o"      'o setzen
  Waitms Pause
  Locate 2 , I
  Lcd " "      'o löschen
  Waitms Pause
  X = X + 1
  Merke = I    'Merke die letzte Position
  If I < 6 Then I = I + 1 Else I = 1 'Nächste Position
Loop Until X = Zz

Locate 2 , Merke
Lcd "o"      'gelöschtes o noch einmal zeigen
End Sub
```