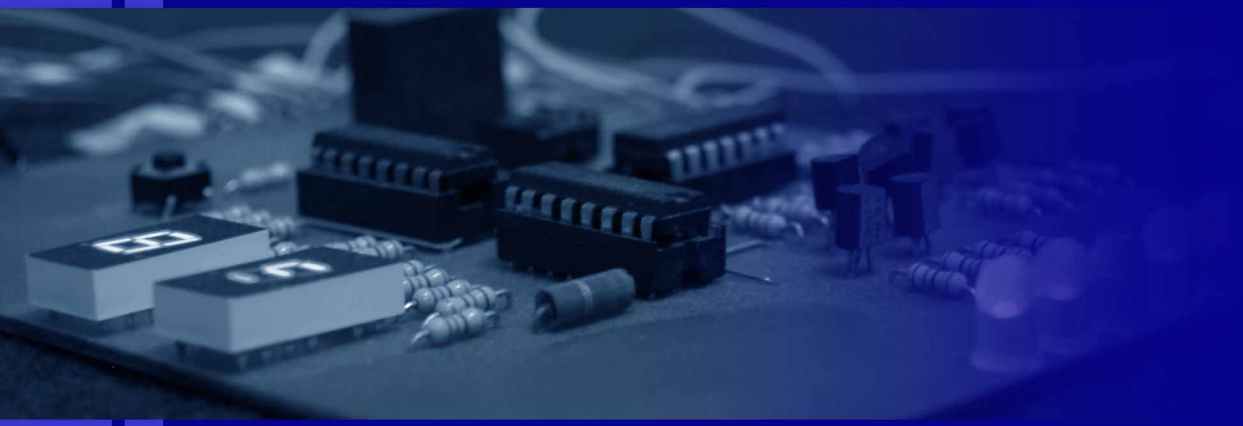

Ralf Pongratz

The Basics of Reactive Lights

A step by step guide

www.reaktivlicht.de

Ralf Pongratz

The Basics of Reactive Lights

A step by step guide
Version 1.00.00

21. Juli 2012

www.reaktivlicht.de

Inhaltsverzeichnis

1

Introduction

This book is a step by step introduction to build your own reactive light. It is based on the Cookbook[1]. Th user should have basic knowledge of electronics and programming.

1.1 Warnings

This introduction doesn't claim completeness and may contain errors. Everyone who builds and programms this circuit is doing it on his own risk. Most people doesn't know: The circuit is connected directly to the mainboard of your PC. Apart from coupling interferences and maladjusted impedance (you don't have to care whenusing this old interface) a short-circuit between a signal- and the ground-wire of the parallel interface can result in serious consequences. Anyone, who doesn't have experiences in electronics should sit down with a cup of tea in a comfortable armchair and read this introduction completely. Doing this will solve most o the problems automatically.

1.2 Order Data

Here are the order data of Reichelt of the used parts. Anybody who thinks that despite the list he needs more help is allowed to contact me[2]. If you'll pay for the shipping, I can send you the needed parts.

[1] `http://www.reaktivlicht.de`
[2] kontakt@reaktivlicht.de

Part	Reichelt
ATtiny 13V-10PU	ATTINY 13V-10PU
Socket DIL 8-pin	GS8
Resistor 56 Ω	1/4 W 56 Ω
Resistor 220 Ω	1/4 W 220 Ω
Resistor 1 MΩ	1/4 W 1 M
Capacitor 100 nF	MKS-2 100N
Light emitting diode green 13000 mcd	LED 5-13000 GN
Light emitting diode white 18000 mcd	LED 5-18000 WS
SUB-D Plug 25 pins male	D-SUB ST 25
Cap for plug	KAPPE CG25G
Photoconductive cell	A 905014
	A 906014

Tabelle 1.1. Order data of the parts

1.3 Programming interface

To load the program to the microcontroller an interface to the computer is needed. The easiest way is to use the parallel interface. Figure 1.1 shows an easy adapter therefor.

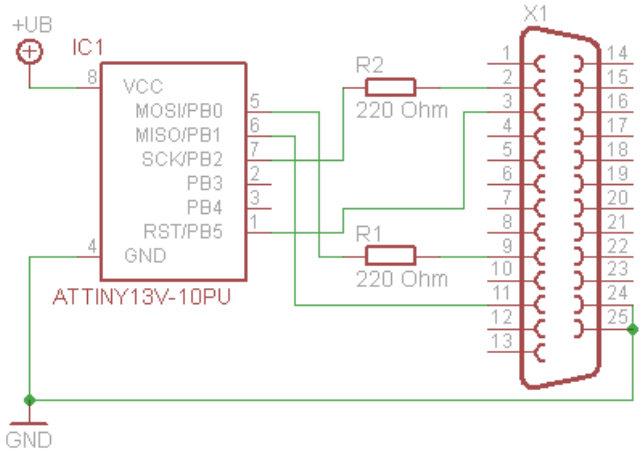

Abb. 1.1. Programming adapter for the parallel interface.

X1 is a 25-pin SUB-D-Plug, that is connected to the parallel interface of the computer. IC1 is the microcontroller to be programmed. Furthermore two resistors (220 Ohm) are needed. The power for the microcontroller will be delivered by an external power supply. To avoid interferences, the cable to the computer must be as short as possible and shielded.

As long as pin 1 is not used and the pins 5, 6 and 7 are not set to fixed potential or connected to each other, the programming adapter can be integrated into the circuit of the reactive light. This eliminates the need to replug the IC when programming.

1.4 Programming software

Various programs are available for programming microcontrollers. Here the program „Bascom AVR“ with the parallel interface adapter (Chapter 1.3) is used. A free demo verssion limited to 4 kB code size is available in the internet[3].

1.4.1 Requirements

A PC with a parallel interface (printer interface), the parallel programming adapter and the programming software is needed.

The parallel interface must be set to ECP+EPP (in- and output) in the BIOS, for example like this: Onboard Parallel Port: 378/IRQ7
Parallel Port Mode: ECP+EPP
ECP Mode Ust DMA: 3
Parallel Port EPP Type: EPP1.7

1.4.2 Using „Bascom AVR“

All explanations are based on the version 1.11.8.3. Newer versions may differ.

At the beginning „Bascom AVR“ must be started and a new program window created using the item „New“ in the menu „File“. After that the programming adapter and the chip parameters must be set. Therefor select in the menu „Options“ the entry „Programmer“.

Go to the tab „Compiler“ and there select the tab „Chip“ (Fig. 1.2). In the drop down menu „Chip“ choose the entry „attiny13.dat“ and insert the following values to the subjacent fields:
HW Stack = 2
Soft Stack = 8
Framesize = 24
After that save the values by clicking „Default“.

[3] `http://www.mcselec.com/index.php?option=com_docman&task=cat_view&gid=99&Itemid=54`

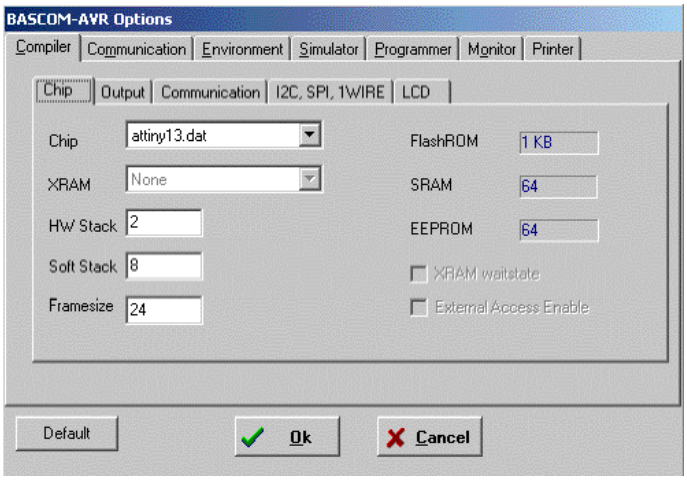

Abb. 1.2. Setting of the chip parameters.

Then go to the tab „Programmer“ and select in the drop down menu „Programmer“ the entry „Universal MCS Interface“ (Fig. 1.3). In the tab „Universal“ the programmer „WinAVR and SP12“ has to be selected. After that the dialog has to be closed by clicking „OK“.

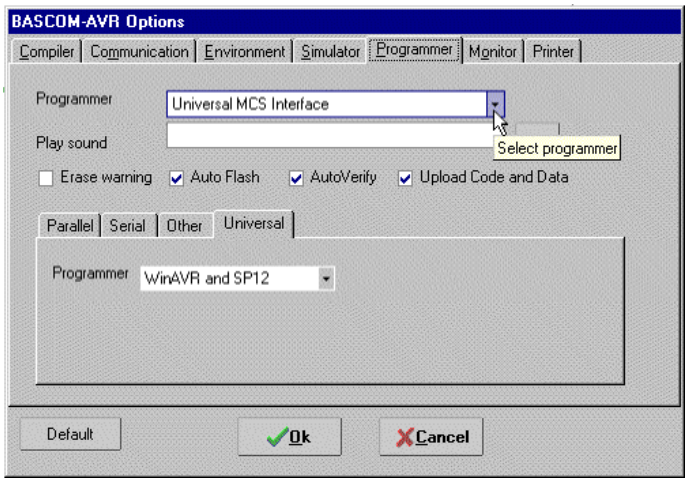

Abb. 1.3. Setting of the programming adapter.

Now the fuse bits has to be set. These are storage cells that determine the basic behaviour of the micro controller. To do that, click on the icon with the small green IC socket and select „Manual Program„ (Fig. 1.4).

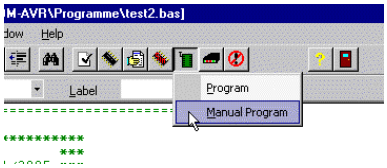

Abb. 1.4. Setting of the fuse bits.

In the dialog that openes click on the tab „Lock and Fuse Bits“. Now Bascom reads and shows the settings of the fuse bits of the controller (Fig. 1.5).

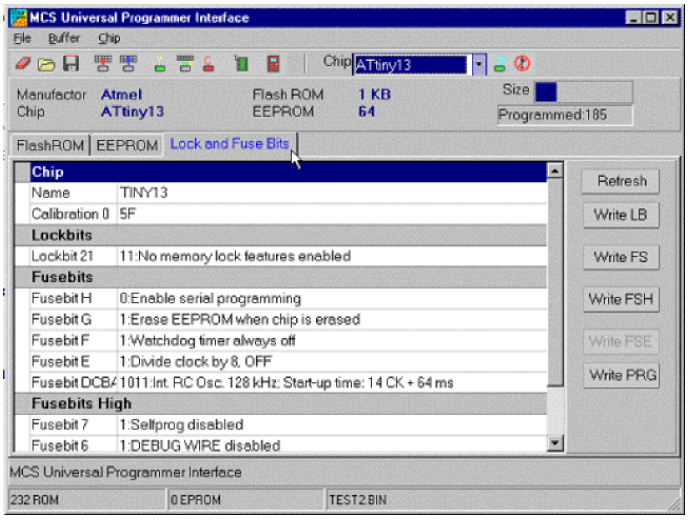

Abb. 1.5. Setting of the fuse bits.

To change a fuse bit, click on the row containing the bit to be changed. Select the new value in the drop down menu. These settings has to be done:
fuse bit DCBA to „1011:Int. Osc. 128 kHz; start-up time: 14 CK + 64 ms“
fuse bit E to „1:Divide clock by 8, OFF“

Please check the settings carefully. If everything is correct, press „Write FS“. The changes will be written to the controller.

Now write the source code of the program. After done, do the compilation by selecting „Compile“ in the menu „Program“ (Fig. 1.6). the compiler starts now. If the program containts errors, messages will be shown in the footer. After compilation succeeded, the program can be written to the micro controller. Go to the menu „Program“ and select the entry „Send to Chip“. In the dialog the entry „Autoprogram“ of the menu „Chip“ der Eintrag „Au-

toprogram“ has to be selected (Fig. 1.7). Now the compiled program will be written to the controller.

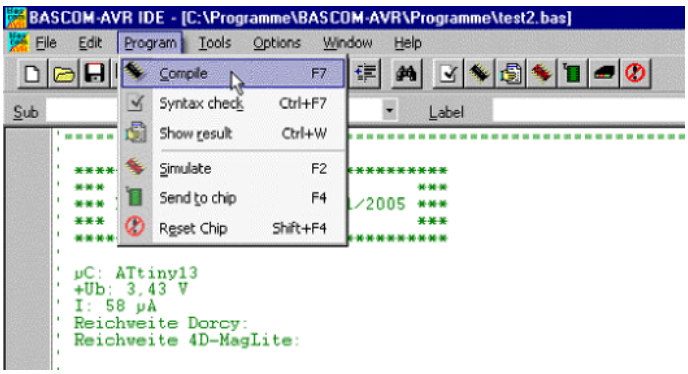

Abb. 1.6. Compilation of the program.

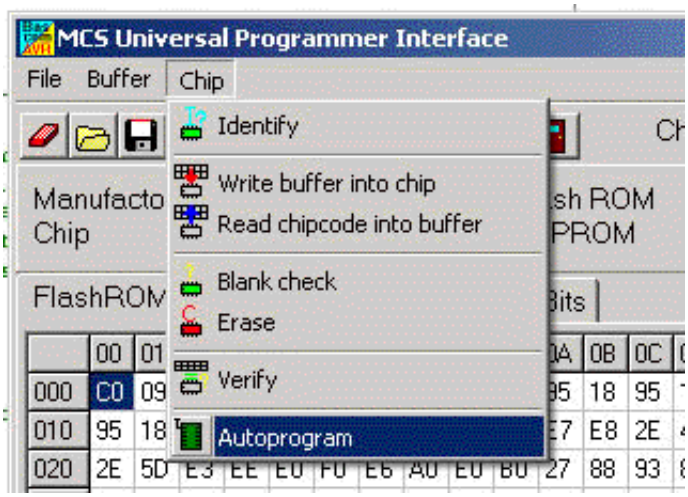

Abb. 1.7. Sending the program to the controller.

2

Design

Figure 2.1 shows the circuit diagram of the reactive light.

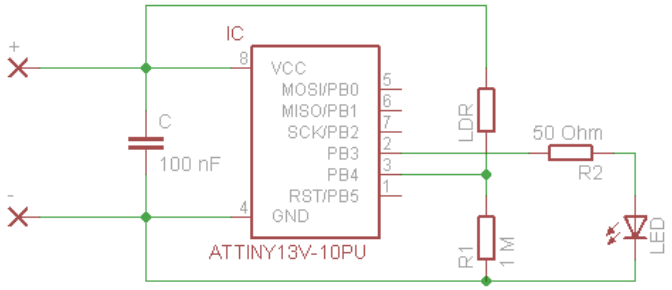

Abb. 2.1. Circuit diagram.

In the center the micro controller is located, on the left the power supply. The circuit needs a voltage of 3 V. Two normal batteries in-line are delivering this voltage. The positive pole belongs to the pin +, the negative one to the pin -. Right of the controller one can find the measurement of the brightness, consisting of R1 and the photoconductive cell LDR, and the LED with the resistor R2 for the output of the flashlight signal.

A concept for the board layout is shown in figure 2.2. Th left side shows the component side, the right one the circuit path seen from the lower side. When assembling one has to care for the polarity of the LED (the flattened side respectively the short wire has to be next to the IC) and of the IC (the denting must be next to the capacitor).

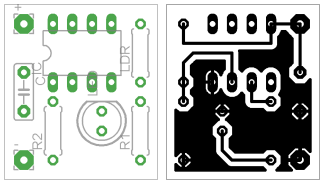

Abb. 2.2. Layout of the circuit board.

3

Programming

```
1  $regfile = "ATtiny13.DAT"
2  $crystal = 16000
3  $hwstack = 2
4
5  Config Adc = Single , Prescaler = Auto
6  Config Portb = &B00001000
7  Portb = 0
8  Stop Ac
9  Wdtcr = &B11010011
10 Enable Interrupts
11
12 Const Schwelle = 50
13 Const Tagschwelle = 800
14 Const Zwangsimpuls = 8
15 Dim A As Byte
16 Dim Tagzaehler As Byte
17 Dim Schlafzaehler As Byte
18 Dim Ldr As Integer
19 Dim Alt As Integer
20 Dim Merker As Integer
21
22 Do
23   Reset Watchdog
24   Powerdown
25   Start Adc
26   Ldr = Getadc(2)
27   Stop Adc
28   Merker = Ldr - Alt
29   Alt = Ldr
30   If Merker > Schwelle Then
```

```
31      Gosub Blinken
32    End If
33    If Ldr > Tagschwelle Then
34      If Tagzaehler < 255 Then
35        Tagzaehler = Tagzaehler + 1
36      End If
37    Else
38      Tagzaehler = 0
39    End If
40    If Tagzaehler > 200 Then
41      Gosub Pause
42    End If
43  Loop
44
45  Blinken:
46    For A = 0 To 10
47      Portb.3 = 1
48      Reset Watchdog
49      Powerdown
50      Portb.3 = 0
51      Reset Watchdog
52      Powerdown
53    Next A
54    Alt = 1023
55  Return
56
57  Pause:
58    Wdtcr = &B11110001
59    Reset Watchdog
60    Powerdown
61    Wdtcr = &B11010011
62    Schlafzaehler = Schlafzaehler + 1
63    If Schlafzaehler = Zwangsimpuls Then
64      Portb.3 = 1
65      Reset Watchdog
66      Powerdown
67      Portb.3 = 0
68      Schlafzaehler = 0
69    End If
70  Return
71
72  End
```

The listing shows the source code. It has to be copied to Bascom AVR, compiled and transfered to the controller. An explanation can be found in chap-

ter 1.4.2. For a description of the program, consult the cookbook. There it is explained in detail.

4

Commissioning

After connecting the batteries, the program starts automatically. If the sensor is lightened, the controller will go to a power down mode. This is signaled by a single flash every 64 seconds. If it is dark, the reactive light can be triggered using a light source. It will flash back 10 times.