# Signal Generator with an ATmega8

This application demonstrates a digital signal generator with an ATMEL ATmega8 with the following properties:

* Adjustabe frequency range: 0,25 c/s to 8 Mcs/s in 1024 steps
* Wide dynamic range of the frequency without mechanic switching
* Reprogramming of the frequency characteristics possible
* Adjustable pulse-width from 0.00 to 100.00% in 1024 steps
* Independant adjustment of pulse-width and frequency
* Xtal base frequency 16 Mcs/s for stable frequency
* Normal and inverted output
* Polarity change by switch (active low, active high)
* Selectable display: frequency, time, rounds per minute and pulse-width
* Suitable for single- and two-line LCDs with 16 to 40 characters per line
* English or german notation selectable

## 0. Content

* [1. Hardware](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardware)
	+ [1.1 Processor section](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardproc)
	+ [1.2 ISP-Interface](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardisp)
	+ [1.3 LCD-Display](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardlcd)
	+ [1.4 External connections](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardconn)
		- [1.4.1 Adjusting frequency and pulse-width](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardpot)
		- [1.4.2 Switches](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardswtch)
		- [1.4.3 Output connections](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardout)
	+ [1.5 Power supply](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardnet)
	+ [1.6 Hints on mounting the device](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#hardwire)
* [2. Use](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#use)
	+ [2.1 Switches](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#usekeys)
	+ [2.2 Output signals](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#usesign)
* [3. How the device works](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#funct)
	+ [3.1 Processor section](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#functproc)
	+ [3.2 LCD display](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#functlcd)
	+ [3.3 ISP interface](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#functisp)
* [4. Software](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#soft)
	+ [4.1 Prior to assembling](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#softsel)
	+ [4.2 Frequency table](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#softselfreq)
	+ [4.3 Switches](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#softselswtch)
	+ [4.4 Commented source code](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.html#softsrc)

## 1 Hardware



The hardware consists of the AVR processor, the In-System-Programming (ISP) connector, the LCD and diverse external connections.

### 1.1 Processor part

The processor is attached to the following external components.The operating voltage of 5 V is supplied via the pins 7 (+5 V) and 8 (0 V) and blocked by a ceramic capacitor.Pin 1 (= RESET-pin) is tied to the positive operating voltage. On the Xtal pins 9 und 10 (XT1, XT2) an Xtal of 16 MHz is connected. Disconnecting the internal RC clock source and activation of the Xtal clock generator is done by setting the respective fuses. Both Xtal inputs are equipped with ceramic capacitors of 22 pF to improve switching characteristics.The operating voltage for the AD converter is supplied over a LC network of a 22 µH coil and a capacitor of 100 nF on pin 20 (AVCC). As reference voltage AVCC is selected by software, so a capacitor to ground is connected to the AREF (pin 21).

### 1.2 ISP interface

The ISP interface allows programming of the AVR in the completed system without removing the chip.ISP uses the port bits 5 (SCK), 4 (MISO), 3 (MOSI) and the RESET on pin 1. MISO is also used to drive a control signal of the LCD. Pinning of the 10-pole ISP connector is compatible with the ATMEL-/KANDA- standard. The LED connected to the ISP connector signals programming currently active.

### 1.3 LCD display

As display a standard LCD is used. It is connected via a 14 pole parallel cable and connectors that fit to the 14-pole connection of the LCD.The control port of the LCD is connected to the port bits PB0 (LCD Enable) and PB4 (LCD RS input). LCD Enable is by default tied to ground over a resistor of 10 k to avoid spurious signals on that line during phases where the pin PB0 is not (yet) initialised (during startup and during ISP programming).LCD-R/W is tied to ground, because the display is not read.The data port of the LCD is connected to the port D of the AVR. Data transfer to the LCD uses the 8 bits parallel, no multiplexing takes place.

### 1.4 External connections

With the exception of the LCD ad the operating voltage supply, that use separate connections, all other external components are connected via a single 14-pole connection. This has the advantage that the processor board can be easily disconnected and testet separately. The 14-pole cable can be separated to portions reaching the different external components systematically.

#### **1.4.1 Frequency and pulse-width adjust**

The first four cables are connected to the operating voltage and the first two AD channel inputs of the AVR.

The AD channels ADC0 (Frequency adjust) and ADC1 (Pulse-width) are connected to the middle of the variable resistors. The variable resistors are of the 10-turn-type and are connected to the operating voltage. Their nominal value and linearity are uncritical, because the real values are displayed on the LCD.Directly on the variable resistors the signal lines to the ADCs are blocked by capacitors to ground to avoid noise on the ADC inputs.

#### **1.4.2 Switches**

The next five connections of the 14-pole cable are connected to the port inputs PC2 to PC5. The switch connected to PC5 is only necessary if a single-line LCD is used. Leave this line open, if you use a two-line LCD.All switches connect the port inputs with ground, if closed. The ports have software-activated pullup resistors and are so held on the operating voltage, if not pulled to ground.

#### **1.4.3 Output connections**

The following three lines of the cable are connected to the outputs of the timer (normal and inverted). These have to be soldered to the CINCH output connectors.The output yield digital signals with the typical characteristics of the AVR output drivers and are directly coupled to the output connectors. The outputs are short-circuit protected, but are not protected against voltages applied externally.

### 1.5 Power supply

The power supply provides a stabilised operating voltage of 5 V at a current of approximately 20 mA. The supply current is depending from the variable resistors. Processor and LCD require below 10 mA, so the whole generator can also be supplied by batteries.To use a small transformer with 6 V secondary voltage, a rectifier bridge with Schottky diodes and a low-drop regulator is used. With a transformer of 7,5 V secondary voltage or more these components can be exchanged to standard components.The charger capacitor has a relatively high capacity, it can be reduced.The two Tantal capacitors suppress instabilities of the regulator.

### 1.6 Hints on mounting the device



Putting the components together on a piece of experimental board is uncritical. The 14-pole connection to the LCD is located to the left, the 14-pole connection to the external components (variable resistors, switches, output connectors) to the right. For easy mounting the parallel cable can be separated into portions of four, five, three and two lines. Please don't forget the two capacitors on the middle of the variable resistors.The 10-pole ISP interface is not used very often, so it is not accessible on the outside of the casing. The power supply is located on the upper left of the plastic case.

## 2 Use

After switching power on the LCD displays a message for about 2.5 seconds, showing the machine's function, the software version and the copyright, with the format depending from the available number of characters of the line. The machine the is ready to operate.

### 2.1 Switches

The switch Time changes the display from frequency (switch open) to time (switch closed). Output of the frequency is in Hz (cs/s) with two decimals, output of the time is in microseconds. Both values are rounded and formatted with thousands separators.If frequency output is selected, the switch Rpm changes from frequency to rounds per minute (= 60 \* f). If the output of time is selected, this switch is ignored.The switch Inv inverts the output signals by software without changing the output connector.If a single-line LCD is used, the switch Pulse causes the display to show the pulse-width in % instead of the frequency/time normally displayed. In case of a two-line LCD the pulse-width is permanently displayed on line 2.

### 2.2 Output signals

The digital output signal is available in positive and inverted form on the two CINCH connectors. To avoid capacitive effects the lines should be short and not shielded.

## 3 How is it working?

This section describes the functioning of the processor, of the ISP interface and the LCD display.

### 3.1 Processor portion

The processor ATmega8 works with an external Xtal and the respective internal oscillator. Because the processor is shipped with an internal RC oscillator of 1 Mcs/s, the respective fuses must be set first to use the Xtal as clock source. The following fuse combinations must be used:

* CKOPT = 0
* CLKSEL3..CKSEL0 = 1111
* SUT1..SUT0 = 11 or 10

Reprogramming of these fuses can either be performed externally on a programming board (e.g. with a STK500 and the ATMEL Studio) or in the completed system with the ISP interface (e.g. with PonyProg2000).



By programming the fuses with an STK500 the Xtal must be connected to the device, otherwise the Mega8 will not respond any more after the fuses are programmed. The fuses are correctly programmed by selecting one of the two last options of the device.



When using PonyProg please note that the fuses are displayed inverted. To have an orientation: by default CKSEL3..CKSEL0 are 0001 and SUT1..SUT0 are 10. Read the fuses by pushing the Read-Button first. Of course, the Xtal should be mounted to the AVR before programming the fuses. CLKSEL and SUT1 fuses should be programmed like displayed here, SUT0 can be programmed to either 1 or 0. If you encounter problems during startup, then program this to 1 (SUT1:SUT0 = 11).

The open switch inputs are at the start pulled to high by software activation of the internal pull-up resistors. If the swichtes are closed, the input lines are pulled to logic zero. The switch Pulse is only needed, if a single-line LCD is used.

Signal generation is performed with the internal 16-bit timer/counter TC1 in the Fast-PWM-mode. The graph shows the function of TC1 in this mode and shows the parameters that affect the operation (blue).

The clock oscillator, controlled by an external Xtal, is divided by the TC1 prescaler either by 1, 8, 64, 256 or 1024 and drives the counter. When the counter reaches the TOP value, which was written by software to the double register ICR1, the counter is reset with the next clock cycle and the compare outputs OC1A (Portbit PB1, Pin 15) und OC1B (Portbit PB2, Pin 16) are activated. The frequency of the generator so is defined by ICR1. Depending from the selected value for the pulse-width of the signal, the compare values in the register pairs COMPA and COMPB are adjusted. If the counter reaches these compare values, the respective compare output is deactivated and stays so until the counter reaches TOP.The two outputs OC1A and OC1B are, by software, of different polarity. They produce inverted signals of the same duration. The switch Inv inverts this polarity by software.

Frequency adjustment is done with the variable resistor attached to the ADC0 input. The resistor rings a voltage of between 0.000 and 5.000 V to the ADC0 (port PC0, pin 23). After conversion, a value between 0x00 and 0x1F results. This value is used to pick a respective value for TOP in ICR1 from a table with 1024 values (Lookup-Table, Include file rectgen\_m8\_table.inc). Depending from the value read from ADC0, the prescaler control byte for TC1 is also prepared. Both values are stored in SRAM until the update of TC1 takes place.

Pulse-width adjustment is done with the second variable resistor and the voltage reaches ADC1 (port PC1, pin 24), which is also converted to a value between 0x00 und 0x1F. The TOP value is multiplied by this conversion value and then divided by 1024 to yield the compare value, next to be written to COMPA and COMPB.

During the same cycle, the switch Inv is read and the resulting control value for the correct polarity of OC1A and OC1B is determined.

With the parameters mentioned, the TC1 counter runs free without additional software overhead. In order to update these parameters the 8-bit timer/counter TC0 is used. The TC0 prescaler divides the system clock by 1024 and overflows after 256 prescaler pulses (@16Mcs/s: every 16,384 ms). TC0 interrupts and decrements a register, from 30. If this reaches zero (after 492 ms), the AD converter is connected to channel 0 and the first conversion is started.

The ADC runs with a clock that is derived from the system clock and divided by 128. After the first ADC result is complete, the ADC interrupts the CPU. The interrupt service routine reads the result to a double register, sets a flag bit, muxes channel 1 to the ADC and starts the second conversion. If the result of the second conversion interrupts the CPU, this result is read, the ADC is switched off and a signal flag is set.

The conversion of the ADC values and the update of TC1 is done asynchroneous in the main program loop, after the signal flag had been set by the interrupt service routine. The ADC values are converted and TC1 is programmed with the new parameters. After programming the TC1, the LCD is updated and the main program loop ends until the next TC0 overflow interrupt wakes up the CPU.

### 3.2 LCD display

The LCD is connected to the 8-bit data port and the two conrol lines E(nable) and R(egister)S(elect). The R(ead)/W(rite)-line is permanently on Write, the whole timing control is done in correct timing loops.

After program start and after a wait time for the internal init of the LCD the LCD is set to the following modes:

* 8-Bit interface
* Single or two line operation (depending from the values in the software)
* no display shift
* cursor off
* blinking off

Then, for 2.5 seconds a message is displayed.

After each update of the TC1 timer the LCD is also updated. First the CTC value in ICR1 is multiplicated by the prescaler value (1, 8, 64, 256, 1024). I displaying of the frequency is selected, the clock frequency, multiplied by 100, is divided by this value to yield an integer representing the frequency with a resolution of 0.01 cs/s. If displaying of the time is selected (switch Time closed), the CTC value in ICR1 is multiplied by the prescaler value, and then multiplied by the factor 25.600.000.000 / clock (@16MHz: 1.600) to yield the time in microseconds (\*100). If a two-line LCD is used, the value of frequency or time is displayed on line 1. In case of a single-line display, this is displayed only if the switch Pulse is open.

The pulse-width is calculated by multiplying the compare value in COMPA or COMPB by 10,000 and dividing by the CTC value in ICR1. The resulting integer is the pulse-width in % with a resolution of 0.01%. This value is written on line 2 (on a two-line LCD) or written on line 1 (of a single-line LCD, if switch Pulse is closed).

The display is updated approximately 2 times per second.

At higher frequencies the frequency and pulse-width cannot be adjusted to accurate values due to the limited resolution of the 16-bit counter. This is recognised by the last two decimals being displayed. Due to the fact that the displayed numbers are calculated from the real values used in TC1, these numbers are correct.

### 3.3 ISP interface

The ISP interface is for updating the software in circuit. The data and clock signals MOSI, MISO and SCK are accessed on the 1o-pole connector. Via the RESET line (port PC6, pin 1) the ISP interface brings the ATmega8 to the programming mode. After releasing RESET, the AVR restarts.

The programming LED indicates an active programming cycle, is only active in case of programming. If not build in, or if a 6-pin connector is used instead of the 10-pin standard, this does not change functioning.

## 4 Software

Te software is written exclusively in assembler language and divided into three different packages. Prior to assembling the source code a number of parameters has to be adjusted to optimise the hardware.

### 4.1 Prior to assembling

#### **4.1.1 Frequency table**

The frequency table in the file rectgen\_m8\_table.inc has 1024 words for converting the selected voltage to the CTC value for ICR1. The values were calculated with a [Spreadsheet](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8.xls) and exported as an Include-Text-file. If you make changes in that table, please note that this may have consequences for the prescaler values. In that case you will have to readjust the prescaler values in the routine Convert in the file rectgen\_m8\_v1.asm (current values: 392, 225, 60 and 3).

If changing the clock frequency of the Xtal oscillator the constant clock will have to be changed. In that case also the 5-byte-constants cDivFx and cDivUx have to be changed accordingly. These cannot be calculated by the assembler due to overflow problems. Note that changes in the clock frequency are automatically changing the LCD timing loops, no additional adjusts in the driver routine are necessary.

The constants cLcdLw and cLcdLn define the connected LCD. Inproper settings might cause strange effects on the display.

The constant cLcdMicro defines the micro character of the connected LCD. The default is not a micro but a u, because some displays do not have greek characters and do not use 0xE4 for that character.

The constant cEn enables english thousands- and decimal-separators.

#### **4.1.2 Diverse switches**

The switches Time, Rpm, Inv and Pulse can be placed to different pins, this can be re-defined with the constants pbTime, pbRpm, pbInv and pbPwm.The switches dbgon and dbghx are for debugging only. For correct function of the software these must be set to zero!

### 4.2 Commented source code

The commented source code is available in HTML-format

* [Source code of the main program](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8_v1_main.html)
* [LCD drive routines](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8_v1_lcd.html)
* [Frequency table](http://www.avr-asm-tutorial.net/avr_en/signalgen/rectgen_m8_v1_freqtab.html)

and in zipped form as assembler source code: [Assembler source code](http://www.avr-asm-tutorial.net/avr_en/signalgen/signalgen_src.zip).

**List of parts:**

uP ATMega8 1x

LCD CM160200 1x LCD 16x2 char ,16pin

coil 22uH 1x

cap 10nF 6x

cap 22pF 4x

xtal 16Mhz 1x

potenciometer 2kOhm 2x with Switch

trimmer 10kOhm 1x

CINCH 2x panel

Zoznam súčiastok:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Názov položky | Skl. č. | Objednaných ks | Rezervovaných ks | Zostáva ks | Cena/pol |  |
| [PC16MSK002.5](http://www.gme.sk/sk/-p113-023.html) | 113-023 | 2 | 0 | 2 | 3,846 |  |
| [CK 10N/100V X7R](http://www.gme.sk/sk/-p120-054.html) | 120-054 | 6 | 0 | 6 | 0,540 |  |
| [CK 22P/100V2 RM2,5](http://www.gme.sk/sk/-p120-184.html) | 120-184 | 4 | 0 | 4 | 0,264 |  |
| [Q 16MHZ](http://www.gme.sk/sk/-p131-019.html) | 131-019 | 2 | 0 | 2 | 0,756 |  |
| [7805](http://www.gme.sk/sk/-p330-001.html) | 330-001 | 2 | 0 | 2 | 0,500 |  |
| [7812-STM](http://www.gme.sk/sk/-p330-005.html) | 330-005 | 2 | 0 | 2 | 0,612 |  |
| [ATmega8-16PU](http://www.gme.sk/sk/-p432-201.html) | 432-201 | 2 | 0 | 2 | 5,764 |  |
| [GDM1602A-RN-GBS](http://www.gme.sk/sk/-p513-157.html) | 513-157 | 1 | 0 | 1 | 5,868 |  |
| [DPU022A3](http://www.gme.sk/sk/-p611-147.html) | 611-147 | 1 | 0 | 1 | 0,837 |  |
| [P-S8861](http://www.gme.sk/sk/-p624-087.html) | 624-087 | 2 | 0 | 2 | 1,070 |  |
| [P-PB303B BLACK](http://www.gme.sk/sk/-p630-097.html) | 630-097 | 3 | 0 | 3 | 0,918 |  |
| [Konektor CINCH FIRST TECH CN-RJ-008-Y](http://www.gme.sk/sk/-p808-131.html) | 808-131 | 2 | 0 | 2 | 0,638 |  |
| [DIL28PZ 7,5mm](http://www.gme.sk/sk/-p824-025.html) | 824-025 | 1 | 0 | 1 | 0,532 |  |