Marten Repeller

inaudible 90 dB

By **Jan Visser** (Elektor Labs)

A marten (or to be more precise: stone marten, *Martes foina*, also known as *beach marten*, *house marten* or *white breasted marten*) is actually a really beautiful animal. To humans the marten is not dangerous, but they can make a great nuisance of themselves — stench, noise, and under the hood even chew through wires and hoses. Because the marten is protected, even when it does become a real nuisance it may not be killed or even caught. What to do? Exactly — ask Elektor for help!

Specifications

- 9 to 12 VDC supply voltage
- 7 mA average current consumption
- 80 mA current consumption during sound burst
- 2 mA quiescent current consumption
- 20 kHz to 43 kHz adjustable frequency
- intermittent output sound

Which was precisely what this Elektor reader did, approaching our lab with the question whether there was some electronic solution to be devised to the marten problem. Some research revealed that the *mustelidae* have a terrible dislike to sound waves around 23 kHz — that's ultrasonic, i.e. inaudible for humans. This knowledge forms a good starting point for an electronic marten repeller. An old hand seemed to remember that we had already designed a marten repeller, and indeed: that was January 2003. That circuit, however, was an entirely discrete design, on a relatively large circuit board — in the present microcontroller era it should be achievable to make this much smaller (and cheaper)! It's time to get to work. The wish list

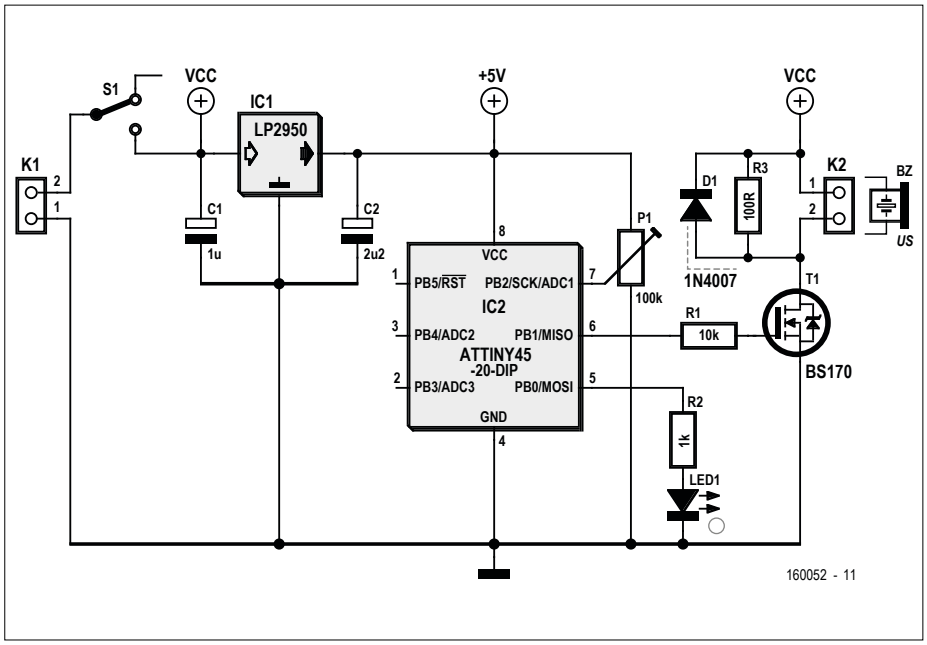

Figure 1. The schematic of the ultrasonic sound generator shows a very modest design; the circuit can be built by anyone.

PROJECT INFORMATION

ultrasonic
microcontroller
through-hole

entry level
intermediate level
expert level

1 hour approx.

normal soldering iron

£12 / € 15 / $17 approx.

from our Elektor reader was actually quite modest:

- simple, compact design
- simple construction, no finicky work with SMD components
- adjustable frequency range 20 through to 40 kHz
- substantial sound intensity
- cheap
- power supply (also) from a 9-V battery

The schematic

Figure 1 shows the modest schematic of our ultrasonic sound generator. The heart of the circuit is formed by — and how could it be otherwise — a microcontroller, in this case a small one: the ATtiny25-20. But let's start from the beginning: the power supply. The input voltage, in the range from 9 to 12 V_{DC} (derived from a 9-V battery, a line power adapter or a car battery), enters through the two-way header K1 and continues via on/off switch SW1 to a well-known, low-drop voltage regulator, the LP2950 in its 5-V variant. Capacitors C1 and C2 are part of the standard configuration and ensure the stability of the output voltage.

The microcontroller is powered from the regulated 5-V output voltage that is generated by this regulator; for the output stage this is not necessary of course, for this the 'raw' battery voltage is used.

Through to the heart

IC2 is the heart of the generator — an ATtiny25 with an absolute minimum of ancillary components. Trimpot P1 serves for setting the output frequency (with a range of about 20 kHz to about 43 kHz). The firmware has been developed such that the generator supplies an intermittent signal. We have done this to prevent the battery from being drained too quickly, and to avoid the scared animals from becoming habituated. We chose to generate a burst of roughly 1 second every 10 seconds. Since we cannot hear whether the circuit is active (that is, makes noise), we have added LED1 (in combination with a series resistor R2), which makes to operation of the circuit visible.

The output of the microcontroller (Pin 6) is nowhere near capable of delivering enough current to drive a loudspeaker

Figure 2. The MPT-001 tweeter from Monacor used for the project. After measurements in the Elektor Labs this appeared to be eminently suitable for the application at hand.

directly, so for this purpose we have added a driver stage around T1 (the well-known MOSFET BS170).

Tweeter

In order to send the amplified output signal from the controller (bursts of a frequency that is (far) above our range of hearing into the world with a substantial amount of decibels, it is best to use a piezo horn tweeter with a high efficiency. After some searching and experimenting we selected the MPT-001 from Monacor (**Figure 2**). Although this tweeter is not specifically intended for ultrasonic applications, it combines a reasonable efficiency with modest dimensions and, above all, a pleasant price of not even a tenner. In any case, good enough for our purpose. But if you happen to have another U/S tweeter or would prefer to use a different model, then go ahead! In this aspect the circuit invites experimenting.

Construction and use

A circuit board, only a few square centimeters in size, has been designed for the marten repeller (**Figure 3**). Because it uses only conventional ('through-hole') components, even a less-experienced hobbyist will not need much more than an hour to assemble it. Use a socket for the

This software repels martens...

```
//Attiny25 , running @ 1MHZ
// Using timer 1
//
//                      +-\/-+
// Ain0  (D  5)  PB5  1|    |8  VCC
// Ain3  (D  3)  PB3  2|    |7  PB2  (D  2)  INT0  Ain1 <- Potmeter input
// Ain2  (D  4)  PB4  3|    |6  PB1  (D  1)        pwm1 <- Speaker output
//               GND  4|    |5  PB0  (D  0)        pwm0 <- Led output
//                      +----+

//  calculate output frequency
//  clockspeed / prescaler / OCR1C = frequency(Hz)
//  clockspeed / prescaler / frequency = OCR1C
//  OCR1C = OCR1A

#include <avr/sleep.h>
#include <avr/interrupt.h>
#include <avr/wdt.h>

#define potmeter A1

#define adcDisable() (ADCSRA &= ~(1<<ADEN)) // disable ADC
#define adcEnable()  (ADCSRA |=  (1<<ADEN)) // re-enable ADC
byte compareValue = 0;

void setup() {
  disableWatchdog();
  adcEnable();
  compareValue = map(analogRead(potmeter), 0, 1023, 25, 10);
                 startTimer1(); //20 - 50 kHz -> OCR0A: 25 - 10
                   adcDisable();
                    pinMode(0, OUTPUT);
                    digitalWrite(0, HIGH);
                    delay(1000);
                    digitalWrite(0, LOW);
                    stopTimer1();
                    enableWatchdog();
                    enterSleep();
```

Repel undesired rodents with a handful of electronics

microcontroller! There is very little to say about the usage: there is only one trimpot (P1) that is used to set the frequency of the output signal. To verify it is best to use an oscilloscope or a frequency counter. If you have no 'scope at your disposal, you can also work by ear. Although you cannot hear the actual output signal, after switching on there will be an audible click from the tweeter roughly every 10 seconds (for each burst). Then you know that everything is working as it should.

```
}

void loop() {
}

void startTimer1(void) {
  pinMode(1, OUTPUT);
  TCNT1 = 0;
  TCCR1 = 0;
  GTCCR |= (1 << PSR1);   //section 13.3.2 reset the prescaler
  TCCR1 |= (1 << CTC1);   // section 12.3.1 CTC mode
  TCCR1 |= (1 << COM1A0); //togle pin PB1 table 12-4
  TCCR1 |= (1 << CS10);   //prescaler 1 table 12-5
  //TCCR1 |= (1 << CS11);
  //TCCR1 |= (1 << CS12);
  //TCCR1 |= (1 << CS13);
  OCR1C = compareValue;
  OCR1A = compareValue;
}

void stopTimer1(void) {
  TCNT1 = 0;
  TCCR1 = 0;
}

void enableWatchdog(void) {
  wdt_enable(WDTO_8S); //enable watchdog
}

void disableWatchdog(void) {
  MCUSR &= ~(1<<WDRF); // reset status flag
  wdt_disable();       //reset watchdog
}

void enterSleep(void) {
  set_sleep_mode(SLEEP_MODE_PWR_DOWN);
  sleep_mode();
}
```

COMPONENT LIST

Resistors (0.25 W)
R1 = 10kΩ
R2 = 1kΩ
R3 = 100Ω
P1 = 100kΩ trimpot horizontal

Capacitors
C1= 1µF 16V
C2 = 2.2µF 16V

Semiconductors
IC1 = LP2950 (5V)
IC2 = ATtiny25-20 DIP, programmed
D1 = 1N4007
T1 = BS170
LED1 = white

Miscellaneous
K1,K2 = 2-way pinheader socket, 0.1" pitch
S1 = slide switch, PCB mount
BZ = Monacor MPT-001 or similar
PCB # 160052-1

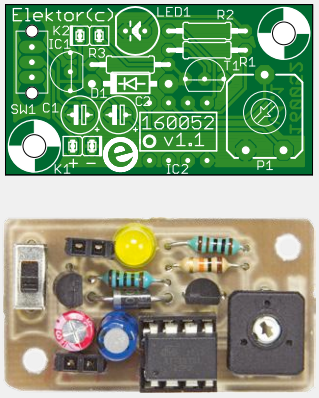

Figure 3. This single-sided circuit board for the marten repeller is available from the Elektor Store.

The software

There is not much to say about the software (see sidebar) — after switching on, the position of potentiometer P1 is read, the timer is configured correspondingly and the marten repeller begins its actual (hopefully beneficial) task. You are obviously free to experiment with the software to your heart's content. Things to consider are the duration of the signal and the repetition time; in principle you could even (if you have sufficient programming experience) write an entirely new routine that generates variable repetition times. The choice is yours!

The software is available as a free download [1]; you can also order a ready-to-go pre-programmed ATtiny25 chip from the Elektor Store for a small amount of money. Or if you prefer to program the microcontroller yourself: this goes very well with the TL866A-programmer, which is also available from the Elektor Store [2].

(160052)

Web Links

[1] www.elektormagazine.com/160052
[2] www.elektor.com/tl866a-universal-programmer

FROM THE STORE

→ 160052-1
unpopulated circuit board

→ 160052-41
programmed ATtiny25-20 DIP