

Reduceer het stroomverbruik van uw mollenverjager

ATtiny13 in plaats van 555



Gerhard Dürr (Duitsland) en Luc Lemmens (Elektor)

Molshopen kunnen uw zorgvuldig gemaaide gazon in een mum van tijd ruïneren. Als u uw kostbare tuin bewaakt met een elektronische mollenverjager, kan deze hardware-upgrade de levensduur van de batterijen met een factor 10 verlengen!

Als u een tuin met gazon hebt, en vooral als u in een landelijke omgeving woont, loopt u het risico onaangenaam verrast te worden door de aanwezigheid van mollen, of liever door de molshopen die ze maken. Het goede nieuws is dat u hoogstwaarschijnlijk schone, vruchtbare grond in uw tuin hebt. U zult de diertjes zelf nooit – of zelden – zien, omdat ze meestal ondergronds blijven in de gangen die ze graven, op zoek naar de wormen en insecten die op hun menu staan. De aarde die ze tijdens het graven verplaatsen, moet natuurlijk ergens heen; die duwen ze omhoog en vormen zo molshopen, het teken dat er een mol in (eigenlijk: onder) uw tuin zit. Niet veel mensen zijn hier blij mee en er zijn talloze methoden bedacht om te voorkomen dat mollen een tuin binnenkomen, of om ze te verjagen als ze zich er eenmaal hebben gevestigd. Een van de oplossingen is

de zogenaamde mollenverjager: een apparaat dat door middel van (geluids)trillingen deze ondergrondse activiteiten voorkomt of beëindigt. Het is een buis met spijkers die in de grond wordt gestoken; in het uiteinde bevindt zich een geluidsbron (meestal in de vorm van een elektronische DC-buzzer) die mollen en ander (on)gedierte zoals woelmuizen en ratten moet verjagen.

Sommige fabrikanten beweren dat de ongediertebestrijders die zij maken *ultrasone* trillingen produceren. Misschien bestaan die wel, maar de meeste apparaten die we hebben gevonden, produceren – volgens de eigen specificaties – geluiden met frequenties in het lage kilohertz-gebied of zelfs lager, wat ver onder het ultrasone bereik ligt. In de meeste gevallen zal de term sonische verjager (zonder ultra-) een betere beschrijving zijn. Het geluid wordt gedurende ongeveer een

halve seconde ingeschakeld, met tussenpozen van ongeveer 30 seconden. Deze pulsen zouden voor een mol zo onaangenaam moeten zijn dat hij verhuist en/of wegblijft. Er zijn mensen die absoluut overtuigd zijn van de effectiviteit van deze apparaten; anderen beweren dat het gewoon oplichterij is en dat alleen de fabrikanten en verkopers er baat bij hebben; daar zullen we ons niet over uitlaten.

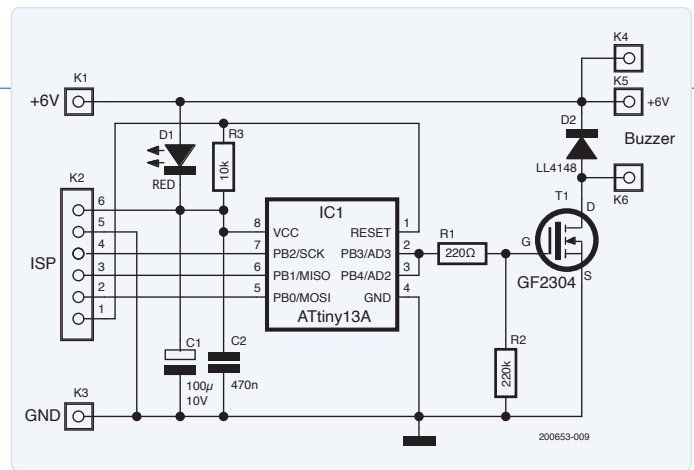
In het **kader** hebben we het kort over het verjagen van mollen in het algemeen. In dit artikel bespreken we alleen een manier om het stroomverbruik van een batterijgevoede, geluid producerende variant te reduceren.

Hoe het werkt

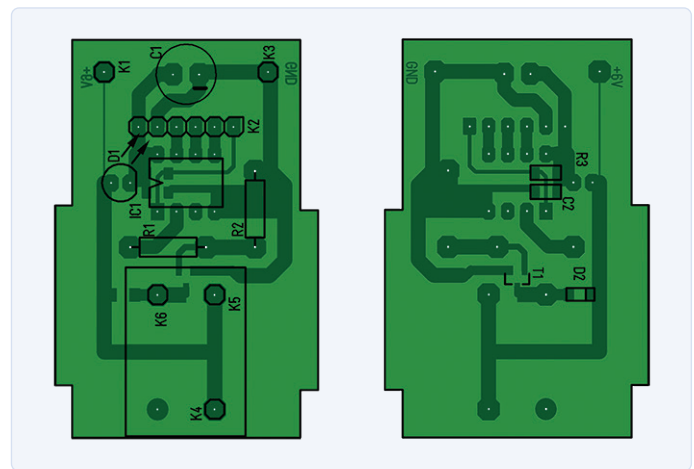
Dit type verjager bevat een timer en iets wat geluid maakt; dat laatste zal meestal een piëzo-zoemer zijn, maar we hebben ook apparaten gevonden met trilmotoren, zoals ook in mobiele telefoons worden gebruikt. Wie in de elektronica "timer-schakeling" zegt, zegt ook meteen "555", misschien wel de beroemdste geïntegreerde schakeling, die onlangs zijn vijftigste verjaardag vierde. Natuurlijk zijn er uitzonderingen zijn, maar de ontwerper van de hier besproken schakeling kwam ook een exemplaar tegen waarin de 555 werd gebruikt als bistabiele multivibrator. Het apparaat verbruikt de meeste stroom wanneer geluid wordt geproduceerd, maar er is natuurlijk ook wat stroom nodig om de 555 aan de praat te houden wanneer de zoemer is uitgeschakeld. Hoewel aanpassing van deze schakeling energie kan besparen en dus de levensduur van de batterij kan verlengen (bijvoorbeeld door een standaard 555 te vervangen door een CMOS-versie of door de waarden van de externe componenten te wijzigen), koos de ontwerper voor een andere aanpak. De complete timer-schakeling wordt hier vervangen door een kleine microcontroller die het grootste deel van de tijd in slaaptoestand verkeert; zijn watchdog-timer wekt hem ongeveer om de halve minuut en activeert de zoemer gedurende één seconde, waarna hij weer gaat slapen. Het stroomverbruik neemt zo met bijna een factor tien af: 4 mA met de originele 555-timer tegenover gemiddeld 430 μ A met microcontroller; de batterijlevensduur van minder dan 200 dagen is verlengd tot ongeveer 1450 dagen! In de praktijk zal dit minder zijn, onder meer door de zelfontlading van de batterijen, maar toch is dit een grote verbetering ten opzichte van het oorspronkelijke stroomverbruik van de mollenverjager.

De nieuwe hardware

Het schema van de microcontroller-schakeling is getekend in **figuur 1**. IC1, een ATtiny13 van Microchip (vroeger Atmel), vormt het hart van de schakeling. Het interne programmeergeheugen kan worden geflasht met een ISP-programmeerinterface die op K2 wordt aangesloten. Deze 6-polige connector kan ook worden weggelaten als de microcontroller wordt geprogrammeerd voordat hij in zijn voetje wordt gestoken. De schakeling wordt gevoed door batterijen, in dit geval vier D-cellen in serie – dus 6 V in totaal, aangesloten op K1 en K3 (resp. de plus- en minpool). Met vier verse 1,5V-batterijen en zonder (of met zeer geringe) belasting zal de totale spanning hoger zijn, ongeveer 6,5 V. De ATtiny13 heeft echter een absolute maximale voedingsspanning van 5,5 V. D1 is een standaard rode LED (bij voorkeur geen hoogrendement-type) met een doorlaatspanning van ongeveer 1,3 V, zelfs bij een doorlaatsstroom van slechts 1 μ A; dit reduceert effectief de voedingsspanning tot een veilige waarde, ruim onder het maximum van 5,5 V voor de microcontroller. Als alternatief kan D1 worden vervangen door twee of



Figuur 1. Het schema van de timer-upgrade.



Figuur 2. De print-layout.

drie standaard 1N4148-diodes in serie. Een lineaire spanningsregelaar kan hiervoor niet worden gebruikt, omdat deze het stroomverbruik van de mollenverjager zou verhogen. C1 en C2 ontkoppelen de voeding van de microcontroller.

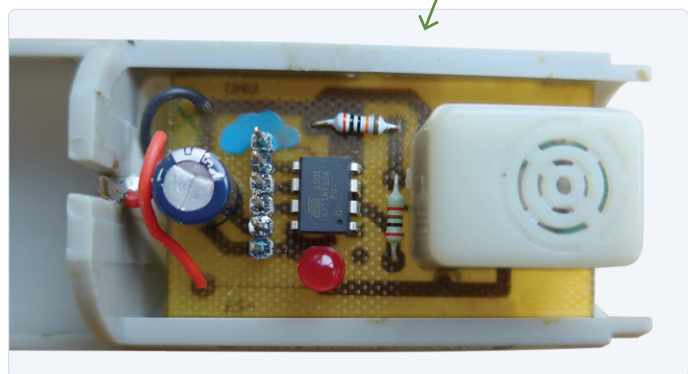
De zoemer is uit de originele hardware gesloopt en aangesloten op K4/K5 (plus) en K6 (min). Sommige zoemers zijn inductief en kunnen spikes produceren, vooral bij het uitschakelen. Diode D2 is de vrijloopdiode die voorkomt dat transistor T1 hierdoor beschadigd kan worden. De keuze van deze schakeltransistor is helemaal niet kritisch. Met een stroom door de zoemer van minder dan 10 mA, kan zelfs een standaard BC547 worden gebruikt als basisweerstand R1 wordt gewijzigd in 4,7 k Ω en R2 in 10 k Ω . Elke 'logische' NMOS-FET is ook goed, als R1 wordt vervangen door een draadbrug en R2 wordt weggelaten.

Er is zelfs een printontwerp

Het schema is ontworpen in een gratis versie van Target3001!, en de auteur heeft zelfs een kleine print gemaakt voor de microcontroller-upgrade (**figuur 2**). De ontwerpbestanden kunnen worden gedownload van de Elektor Labs-pagina van dit project [1]. Gezien het grote aantal merken en soorten mollenverjagers op de markt, zou het toeval zijn als de print even goed past in uw exemplaar als in dat van de auteur (**figuur 3**). Maar dat zal voor veel lezers een mooie gelegenheid zijn om Target3001! te testen met dit relatief eenvoudige project als u de print



Figuur 3a. Overzicht.



Figuur 3b. De nieuwe print in de originele mollenverjager.

wilt veranderen. Het zal ook niet al te moeilijk zijn om het schema en de print opnieuw te tekenen als u liever een ander CAD-programma gebruikt. Er zijn enkele SMD-componenten op de koperzijde (onderkant) van de print, maar die kunnen gemakkelijk worden vervangen door through-hole onderdelen, zelfs zonder de printplaat aan te passen.

Mollen in uw tuin? Wat (niet) te doen?

Als u op internet zoekt wat u kunt doen tegen mollen in uw tuin, of hoe u kunt voorkomen dat ze überhaupt op uw terrein komen, zult u verbaasd zijn wat er in de loop der jaren allemaal is uitgevonden om deze beestjes het leven zuur te maken. Van veel van deze middeltjes wordt de effectiviteit door (ervarings)deskundigen betwijfeld. U weet pas zeker dat zo'n methode niet werkt als – ondanks alles – uw mooi gemaaid gazon plotseling ontsierd wordt door molshopen.

Wij zijn zeker geen mollendeskundigen, maar volgens de informatie die wij hebben gevonden, lijkt helemaal niets doen het beste advies. Als er molshopen zijn, heeft een mol zijn hol al gemaakt en zal dat zal meestal alleen uitbreiden als er niet genoeg voedsel is. Als u die minibergjes storend vindt, kunt u het beste de grond voorzichtig wegscheppen; als u er op klopt of stamp, zal de mollen-gang instorten en is de kans groot dat het dier de schade herstelt en daarbij een nieuwe molshoop maakt.

Zonder firmware gaat het niet

De broncode (**listing 1**) voor de ATtiny13 is ontwikkeld in AVR Studio 4.19 met geïnstalleerde AVR-GCC compiler. Hoewel deze IDE verouderd is, kan hij nog steeds worden gedownload. Het zal echter niet moeilijk zijn om de broncode over te nemen in modernere ontwikkeltools. De broncode, het AVR Studio-project en het HEX-bestand kunnen worden gedownload van de Elektor Labs-pagina bij dit project [1]. Met dat HEX-bestand kunt u het programmeergeheugen van de ATtiny13 flashen zonder te compileren. De configuratie-fuses van deze microcontroller blijven op de fabrieksinstelling staan (ze hoeven dus niet geprogrammeerd te worden).

Hoewel het programma in C is geschreven, is de meeste code geschreven in assembler. Na het inschakelen worden de GPIO's geconfigureerd en worden de timers, de ADC, de pull-ups en de ADC uitgeschakeld. De slaap-modus wordt ingesteld op *Powerdown*. De code komt dan in een eindeloze lus (**while(1)**), die de volgende bewerkingen uitvoert:

- de zoemer wordt ingeschakeld;
- de watchdog wordt ingesteld om een interrupt te genereren na 128k cycli; met een 128kHz-klok levert dit een interrupt-interval van (ongeveer) één seconde;
- interrupts worden ingeschakeld en de ATtiny schakelt over naar *Powerdown*-modus met het assembler-commando **sleep**.
- na één seconde zal de watchdog-interrupt de microcontroller uit de *Powerdown*-modus halen;
- de zoemer wordt uitgeschakeld en de watchdog-interrupt wordt ingesteld op zijn maximum interval van 1024k cycli (is 8 seconden). Omdat we willen dat de mollenverjager ongeveer een halve minuut stil is, moet de watchdog drie interrupts genereren voordat **while(1)** weer begint met het inschakelen van de zoemer.

Geen mollen meer?

Met deze upgrade van de mollenverjager wordt de levensduur van de vier D-cellen verlengd. De effectiviteit van het apparaat zelf verandert er niet door. Zoals gezegd bestaat er veel onenigheid tussen mensen die deze mollenverjagers geweldig vinden en anderen die geloven of zelfs zeker menen te weten dat ze geen enkel effect hebben. Het grote aantal merken en verkopers suggereert dat veel mensen geloven dat ze echt werken. Het is in ieder geval een middel dat geen negatieve effecten heeft op het milieu – het enige dat we konden vinden waren berichten over bureaus die klagen over het zich herhalende zachte geluid van de apparaten. ◀

200653-03



ONDERDELENLIJST

Weerstanden:

R1 = 220 Ω
 R2 = 220 k Ω
 R3 = 10 k

Condensatoren:

C1 = 220 μ /25 V
 C2 = 470 n

Halfgeleiders:

D1 = LED rood
 D2 = LL4148
 T1 = GF2304 (zie tekst)
 IC1 = ATtiny13A (DIL-8 behuizing)

Diversen:

K2 = 6-polige SIL-header

Een bijdrage van

Ontwerp: Gerhard Dürr

Tekst: Gerhard Dürr,

Luc Lemmens

Illustraties: Gerhard Dürr,

Patrick Wielders, Luc Lemmens

Redactie: Jens Nickel, C. J. Abate

Vertaling: Eric Bogers

Layout: Giel Dols

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de redactie van Elektor via redactie@elektor.com.



GERELATEERDE PRODUCTEN

➤ **Diamex EXA-Prog Package (SKU 19912)**
www.elektor.nl/19912



Listing 1. De volledige broncode.

```
// Controlling the molechaser-buzzer with the watchdog-timer
// * MC ATtiny13A, all fuse bytes default
// * PB3 and PB4 HIGH: buzzer ON
// * Configurable with watchdog-timer-prescaler
// * Selected: 128K-cycles(~1s) ON, 3*1024K-cycles(~24s) OFF
// * Current consumption: powerdown ~6 $\mu$ A, buzzing ~7.2mA, average 0.43mA
#include <avr/io.h>
#ifndef F_CPU
#warning "Defining F_CPU 1.2MHz"
#define F_CPU 1200000UL
#endif
#include <avr/interrupt.h>
int main(void) {
    uint8_t i=0;
    DDRB = 1<<PB0|1<<PB1|1<<PB2|1<<PB3|1<<PB4;
    PRR = 1<<PRTIM0|1<<PRADC; // Power Reduction Timer0, ADC
    MCUCR = 1<<PUD|1<<SE|1<<SM1|0<<SM0; // Pullup Disable, Sleepmode: Powerdown
    ACSR = 1<<ACD; // Analog Comparator Disable
    while(1) {
        i = 0;
        PORTB |= 1<<PB3|1<<PB4; // buzzer ON
        // change Watchdog-Interrupt to 1s (measured: 1.14s @ 4.5V, 22°C)
        MCUSR = 0; // clear Reset-Flags
        WDTCR = 1<<WDCE |0<<WDE; // Watchdog Change Enable
        // Watchdog Timer Prescaler: 128K-cycles(~1s), Watchdog Timer Interrupt Enable
        WDTCR = 1<<WDTIE|0<<WDE|1<<WDP2|1<<WDP1;
        sei();
        asm volatile("sleep::");
        asm volatile("wdr::"); // Watchdog Reset, wait for WDT-ISR(~1s)
        PORTB &= ~(1<<PB3|1<<PB4); // buzzer OFF
        // change Watchdog-Interrupt to 3*8s: (measured: 27.9s @ 4.5V, 22°C)
        asm volatile("wdr::"); // Watchdog Reset
        MCUSR = 0; // clear Reset-Flags
        WDTCR = 1<<WDCE |0<<WDE; // Watchdog Change Enable
        // Watchdog Timer Prescaler: 1024K-cycles(~8s), Watchdog Timer Interrupt Enable
        WDTCR = 1<<WDTIE|0<<WDE|1<<WDP3|1<<WDP0;
        while(i < 3) { // wait 24s
            i++;
            asm volatile("sleep::");
            asm volatile("wdr::"); // Watchdog Reset, wait for WDT-ISR(~8s)
        }
    }
    EMPTY_INTERRUPT(WDT_vect); // awaking from powerdown-mode
```

WEBLINK

[1] Projectpagina op Elektor Labs: <https://bit.ly/3JWCXBk>