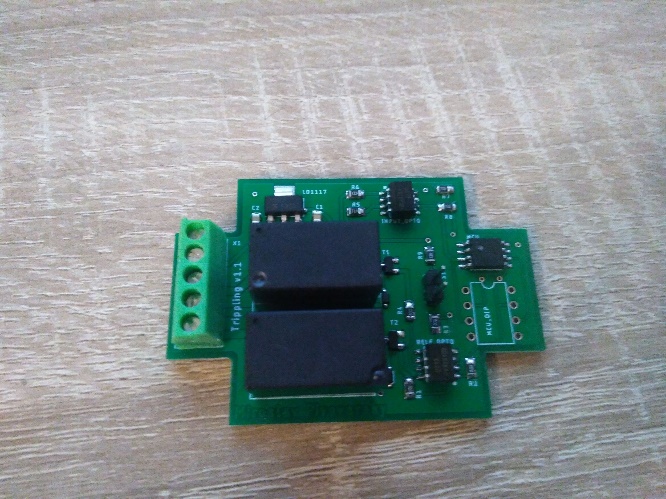
# Trippling – modul komfortného blikania smerových svetiel v aute

Trippling je moderná funkcionalita komfortného blikania smerových svetiel vozidla. Po krátkom stlačení páky smerových svetiel jednotka automaticky 3x blikne daným smerovým svetlom. Elektronika je navrhnutá ako prídavný modul k existujúcemu zapojeniu v automobile, s veľmi jednoduchým zapojením.



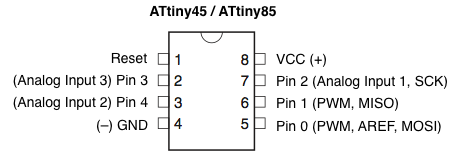
# Popis

Elektronika v zariadení využíva 5 pinov. Základné piny sú napájanie a zem. Ďalej sú to vstupno / výstupné piny pravého a ľavého smerového svetla a ich COM, čiže spoločný pin. Ak je zaznamenané napätie na ľavom alebo pravom smerovom svetle po určitý časový okamih, elektronika to vyhodnotí ako príkaz pre dokončenie trojitého blikania. Tie isté piny, ktoré detegujú napätie na smerovom svetle, sú použité aj pre spínanie smerových svetiel. Časovač zopnutia smerových svetiel pre trojité blikanie ovláda priamo MCU.

V článku použijem fotografie osadenej HW verzie 1.1, avšak opisovať budem HW verziu 1.2. V HW verzii 1.2 je pridaný blokovací kondenzátor pre MCU, pridaný ISP konektor pre pohodlnejšie programovanie MCU a sú vymenené piny procesora pre spínanie relé, s pinmi procesora, ktoré sledujú stav smerových svetiel. Táto úprava bola potrebná, aby bolo možné naprogramovať osadený MCU na PCB.

# Hardware

Samotné zapojenie pozostáva z MCU ATtiny85, dvoch optočlenov, dvoch RELE a stabilizátora napätia. ATtiny85 je jednoduchý 8 nožničkový MCU s 6 digitálnymi vstupmi/výstupmi, taktovacou frekvenciou 20Mhz bez potreby externého kryštálu. Ďalej obsahuje 8kB Flash pamäte, 512B RAM pamäte a 512B EEPROM pamäte. V tomto MCU vieme vypnúť RESET pin, a využiť ho ako štandardný vstupno/výstupný pin. Potom už musíme MCU programovať pomocou „High voltage programming“ rozhrania.



Návrh PCB je prispôsobený tak, aby všetky SMD súčiastky boli osadené z jednej strany. Pri takomto uložení súčiastok môžeme pre prispájkovanie súčiastok k PCB použiť napríklad žehličku. Spájkovanie žehličkou som opisoval v [tomto článku](https://www.elektrolab.eu/blog/osadzanie-smd-suciastok-v-domacom-prostredi-pomocou-zehlicky). Pre nanášanie spájkovacej pasty môžeme použiť podávač spájkovacej pasty. Konštrukciu pre podávač som opísal v [tomto článku](https://www.elektrolab.eu/blog/podavac-spajkovacej-pasty-solder-paste-dispenser).

Zoznam súčiastok:

* Attiny85 SOIC8 alebo DIP8 puzdro
* Stabilizátor LD1117 5V, SOT223 puzdro
* 2x optočlen ILD213T, SOIC puzdro
* 2x tranzistor MMBT3904, SOT23 puzdro
* 2x dióda 1N4148, SOD323F puzdro
* 2x relé HF32F/005-ZS, 5V a 5 pinové! (na Slovensku ho predáva napríklad avelmak.sk)
* 4x SMD rezistor 120ohm 0805
* 3x SMD rezistor 10Kohm 0805
* 2x SMD rezistor 470ohm 0805
* 3x SMD kondenzátor 100nF 0805
* 1x 5 pinový konektor PHOENIX 1751277, rozteč 3.5mm (alebo kombinácia 2 pinového a 3 pinového konektora)
* Krabička typu Z-24BU (známa aj ako KPZ 01 AU)

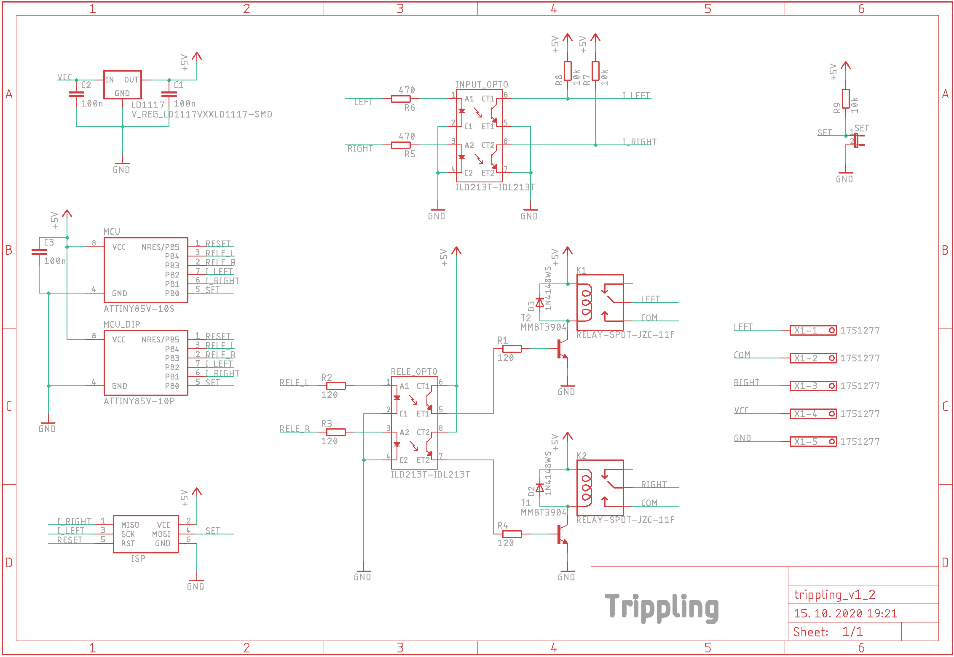
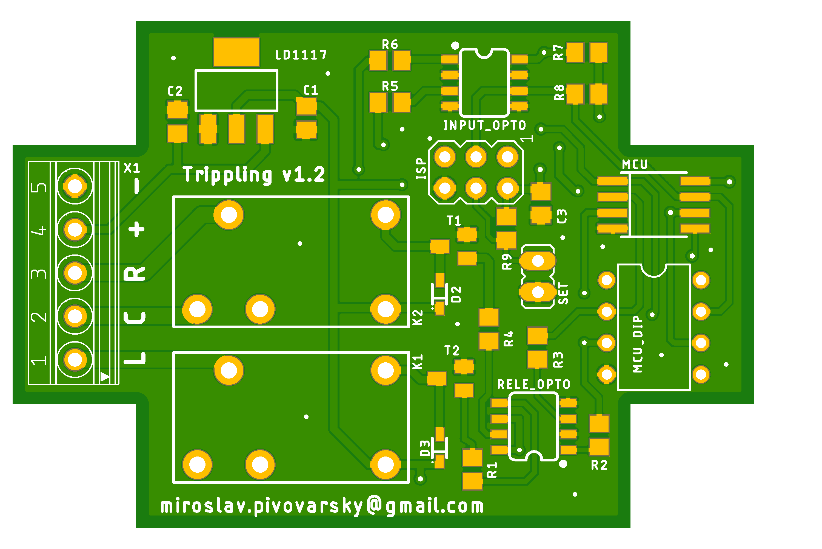
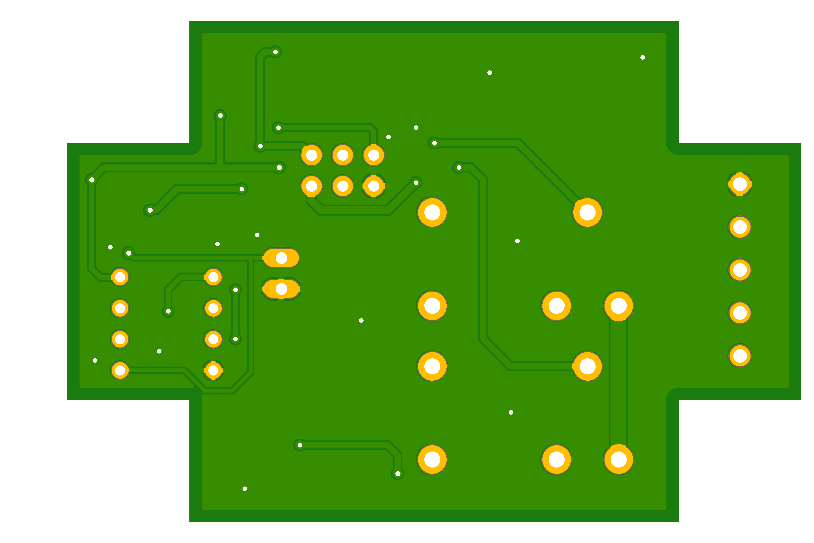


Schéma zapojenia



Vrchná strana PCB



Spodná strana PCB

# Zapojenie

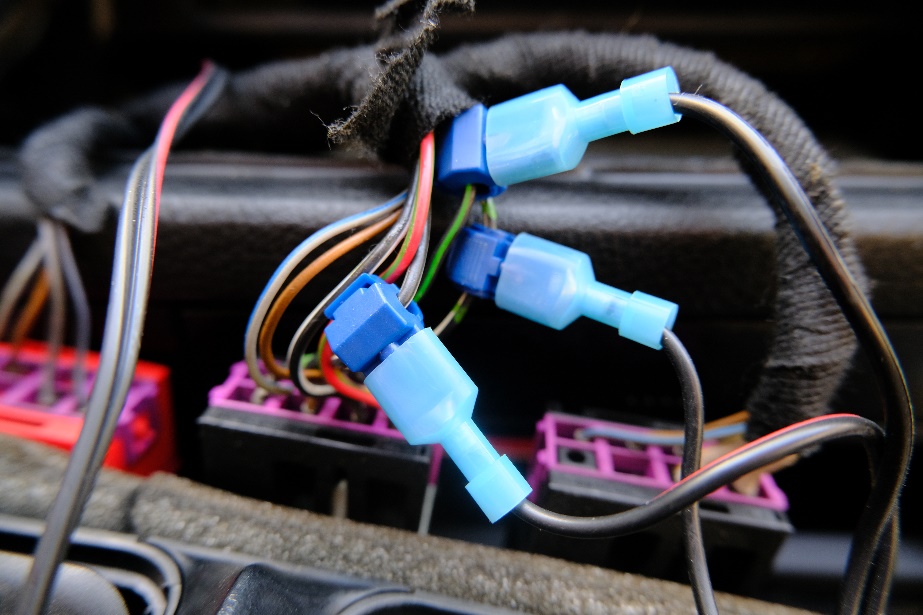
Pre napájanie elektroniky slúžia piny 4 a 5, označené plus a mínus z konektora X1. Maximálne napájacie napätie je 15V a absolútne maximálne napájacie napätie je 20V. To je pre elektroinštaláciu v bežnom aute postačujúci rozsah. Keďže ideme do automobilu pridať elektroniku neschválenú výrobcom, potrebujeme dodržať pár podmienok, aby sme predišli prípadným problémom počas technickej kontroly STK. Prvou podmienkou je pripojiť elektroniku cez vlastnú poistku. Poistka nemusí byť ľahko prístupná v prípade, že ju bude potrebné vymeniť. Druhá podmienka je, aby bolo možné vzpnúť elektroniku kedykoľvek a jednoducho. Na to je ideálny vypínač, ktorý je v dosahu ruky aj počas jazdy autom. Osobne testujem HW a SW už 2 roky, a funguje bez problémov. Nikto by si určite neželal, aby mu z ničoho nič začali blikať smerové svetlá vďaka HW/SW chybe, a nemohol to žiadnym spôsobom vypnúť. Práve z tohto dôvodu je vhodné mať vypínač vždy na dosah ruky. Ja som umiestnil vypínač zo spodnej strany palubnej dosky, blízko vodičových dverí. Vypínač je na prvý pohľad ukrytý, a súčasne vždy ľahko prístupný aj počas jazdy. Z danej poistky a vypínača, je v mojom prípade súčasne napájaná palubná kamera, cúvacia kamera a elektronika komfortného blikania. Odporúčam elektroniku pripojiť na vetvu KL15, čo je vetva, na ktorej sa objaví 12V až po otočení kľúča do prvej polohy.

Keď už máme elektroniku napájanú, ďalej je potrebné pripojiť elektroniku k smerovým svetlám. Nie je možné napísať univerzálne zapojenie pre všetky typy automobilov, keďže každý automobil ma iné farebné značenie kabeláže, a inú konštrukciu. Ja opíšem moje zapojenie, pre VW Golf 4, no myslím si, že by to malo sedieť pre automobily z koncernu VW Group.

Prvým krokom bolo rozobrať stredový výduch. Skrutky sa ukázali po vybratí plastov, pre smerovanie vyfukovaného vzduchu. Odporúčam dať pozor na gumičky, ktoré sa nachádzajú na výduchoch po oboch stranách. Môže sa stať, že gumičky zapadnú dozadu. Bolo by lepšie mať pri manipulácií s nimi tie výduchy zatvorené. Následne som musel jemne odbaliť kabeláž pre výstražné svetlá, a nájsť správne vodiče.



Pre zapojenie potrebujeme vodič pre ľavé aj pravé smerové svetlo, a ich spoločný vodič. Spoločný vodič je 12V napájacie napätie privedené z poistky pre smerové svetlá z poistkovej skrinky. To platí iba vo vozidlách s kladným ovládacím napätím smerových svietidiel a celej elektroinštalácie v aute. Zariadenie je možné použiť aj vo vozidle so záporným ovládacím napätím, keďže relé spína COM port voči príslušnému smerovému svetlu. Všetky tieto vodiče sa nachádzajú napríklad pri výstražnom tlačidle. Preto som zvolil možnosť pripojenia elektroniky k výstražnému tlačidlu. Súčasne je k výstražnému tlačidlu dotiahnutých 12V pre vetvu KL15 (napájanie na vodiči je až v prvej polohe kľúčika v spínacej skrinke) a zem, no ja som ich nevyužil. V aute mám vlastnú vetvu s 12V napájaním cez poistku a vypínač.



V nasledujúcej tabuľke opíšem význam farieb vodičov, a ich piny na výstražnom tlačidle pre Golf 4, čo by malo sedieť pre väčšinu áut daného koncernu medzi rokmi 1999-2005.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Ľavá smerovka | Pravá smerovka | Spoločný kontakt | +12V | ZEM |
| Farba vodiča | Čierno biely | Čierno zelený | Čierno bielo zelený | Čierno biely | Hnedý |
| Číslo pinu na spínači | 3 | 2 | 1 | 7 | 6 |

Pre pripojenie k existujúcej kabeláži som sa snažil využiť najmenej deštruktívnu, a najľahšie rozoberateľnú možnosť. Použil som „T-TAP faston“ konektory. V čínskych e-shopoch sa dajú kúpiť dosť lacno, zvyčajne pod názvom „Snap Splice Lock Wire Terminal Crimp“. Na jednej strane je bočník s nožom, do ktorého sa pripája klasický faston konektor.



Vstup a výstup pre ľavú smerovku je na PCB označený písmenom L, pod číslom 1. Vstup a výstup pre pravú smerovku je označený na PCB písmenom R, pod číslo 3. Ako posledný ostal spoločný vodič, čiže COM. Ten je označený písmenom C, pod číslom 2.

Celá elektronika je navrhnutá tak, aby sa zmestila do spomínanej krabičky Z-24BU.



Po pripojení napájania na pin 4 a uzemnenia cez pin 5, je elektronika ihneď funkčná a aktívna. Procesor neprechádza do žiadneho sleep módu, a nie je v ňom naprogramovaný žiadny power save mód. Preto odporúčam pripojiť elektroniku na vetvu KL15, aby bola elektronika aktívna až po vložení kľúča a pretočení do prvej polohy.

# Ovládanie

Ak je elektronika napájaná, potom automaticky neustále monitoruje smerové svetla. Pre aktiváciu trojitého blikania musíte krátko ťuknúť do páky pre aktiváciu smerového svetla. Pri krátkom ťuknutí elektronika vyhodnotí rozkaz, pre aktiváciu trojitého bliknutia. V prípade, že ste aktivovali trojité blikanie, a rozhodli ste sa zrušiť svoju voľbu, krátko ťuknite do páky pre opačný smer, čím vydáte procesoru príkaz pre núdzové ukončenie jeho činnosti. Elektronika sleduje dĺžku „ťuknutia“ do páky pre smerové svetlá, t.j. sníma dĺžku impulzu pre aktiváciu povelu. V prípade, že sú smerové svetlá zapnuté nad určitý čas (pre príklad viac ako 300ms), procesor to vyhodnotí ako spínanie prerušovačom, ktorý vložil do automobilu výrobca. Z toho dôvodu, aby nezasahoval do oficiálneho cyklovača nevykoná žiadnu operáciu. Takáto situácia môže vzniknúť napríklad pri zapnutí výstražných svetiel, alebo v prípade, keď je páka smerového svetla aretovaná (zapnutá) trvale.

# Software

Zdrojový kód je napísaný v jazyku C, v  Atmel Studio 7.0. Existujú 2 verzie zdrojového kódu. Aby si nemusel každý kompilovať zdrojový kód, pridám vykompilované binárne súbory pre oba verzie zdrojového kódu. V MCU je aktivovaný watchdog časovač, pre prípadné problémy.

Prvá verzia pracuje s možnosťou naučenia sa dĺžky zopnutia smerových svietidiel a následné uloženie hodnoty do EEPROM pamäte. Táto možnosť sa mi úplne neosvedčila. V jednom prípade sa mi stala situácia, kedy sa EEPROM pamäť prepísala. Mohlo sa to stať zákmitom, na jumpri počas štartovania auta. Aktuálne je už jumper ošetrený proti zákmitu. V tejto verzii zdrojového kódu odpojíme elektroniku od napájania, prepojíme jumper **SET** prepojkou, pripojíme napájacie napätie na elektroniku. Následne zapneme smerové svetlá pre ľavú stranu. Necháme chvíľu blikať smerové svetlá, a potom ich vypneme. Následne vypneme napájacie napätie pre elektroniku, a vyberieme prepojku pre jumper **SET**. Touto sekvenciou sme elektroniku naučili časový interval blikania prerušovača v automobile. Tento firmware nájdete pod názvom **trippling\_self\_calibration\_v1\_4\_0.hex**

V druhej verzii si nastavíme fixnú hodnotu dĺžky zapnutia smerových svetiel, a pomocou tejto hodnoty skompilujeme zdrojový kód. Vďaka tomu sa stane hodnota konštantou, uloženou vo FLASH pamäti, ktorá je v princípe neprepisovateľná (neberiem v úvahu extrémne situácie). Tento firmware nájdete pod názvom **trippling\_fix\_time\_v1\_4\_0.hex**, s pevne definovaným časom dĺžky zapnutia smerových svetiel 400ms a trojitým blikaním. V tomto zdrojovom kóde musí byť ťuknutie do páky pre smerové svetlá dlhšie ako 80ms ale kratšie ako 300ms.

V zdrojovom kóde sa nachádza súbor **Conf.h** pre konfiguráciu MCU a jednotlivých funkcionalít. Definícia **BLINK\_INTERVAL** určuje počet bliknutí, aktuálne je táto hodnota nastavená na 3 bliknutia, ale môžete si ju nastaviť na ľubovoľný počet. Dĺžku času, kedy je celé zariadenie neaktívne po núdzovom vypnutí môžeme nastaviť pomocou premennej **SAFETY\_DELAY**, ktorá je aktuálne nastavená na 500ms. Dĺžku zopnutia a vypnutia smerových svetiel počas blikania môžeme nastaviť v zdrojovom kóde pomocou premennej **TIME\_ON\_OFF\_LENGTH**. Táto premenná sa používa iba v tom prípade, kedy využívate pevne nastavený čas blikania z FLASH pamäte. V prípade, že chcete používať EEPROM pamäť pre dynamickú zmenu dĺžky blikania, je potrebné v súbore **Conf.h** nastaviť definíciu **EEPROM\_ENABLE** na hodnotu 1. Časový interval, počas ktorého ak buchneme do páky pre smerové svetlá, bude aktivované trojité blikanie, je vypočítaný automaticky na 80% z času blikania v prípade, ak použijete EEPROM pamäť. Ak použijete pevnú definíciu vo FLASH pamäti, potom tieto intervaly môžeme nastaviť pomocou definícií **TIMEOUT\_ENABLE\_BULB** a **MIN\_TIME\_FOR\_EN\_TRIPP**. Pričom definícia **TIMEOUT\_ENABLE\_BULB** vraví o maximálnej dĺžke zopnutia páky, v tomto prípade 300ms a definícia **MIN\_TIME\_FOR\_EN\_TRIPP** vraví o minimálnej dobe zopnutia páky, v tomto prípade 80ms. Minimálna doba je pridaná z dôvodu auto diagnostiky vypálenej žiarovky v niektorých automobiloch.

Nastavenie poistiek pre MCU ATtiny85 je následovné:

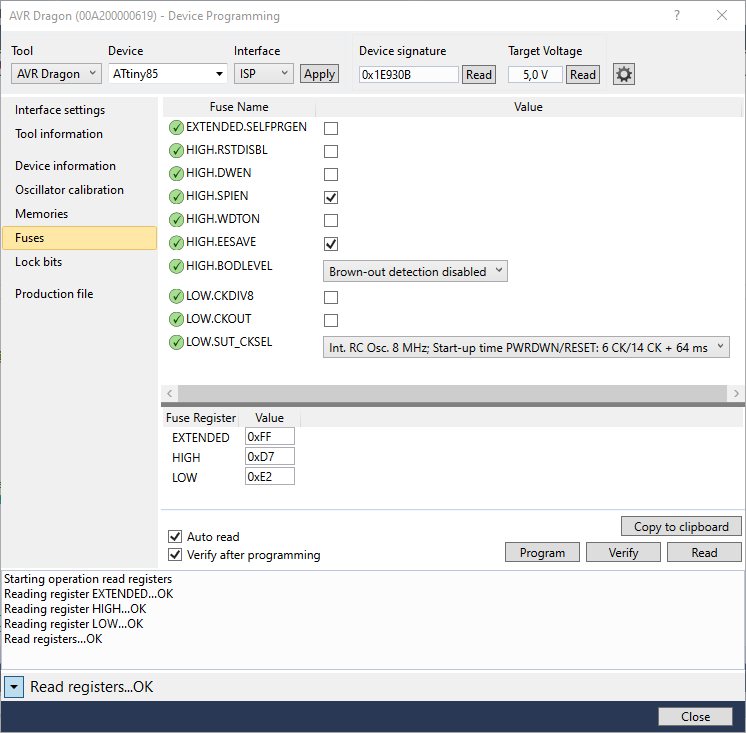
Extended: 0xFF

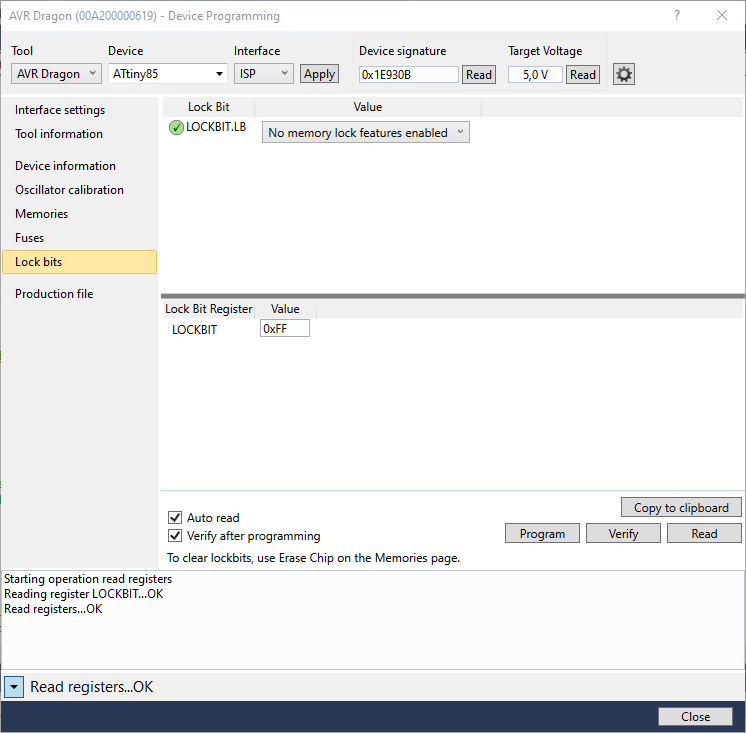
HIGH: 0xD7

LOW: 0xE2

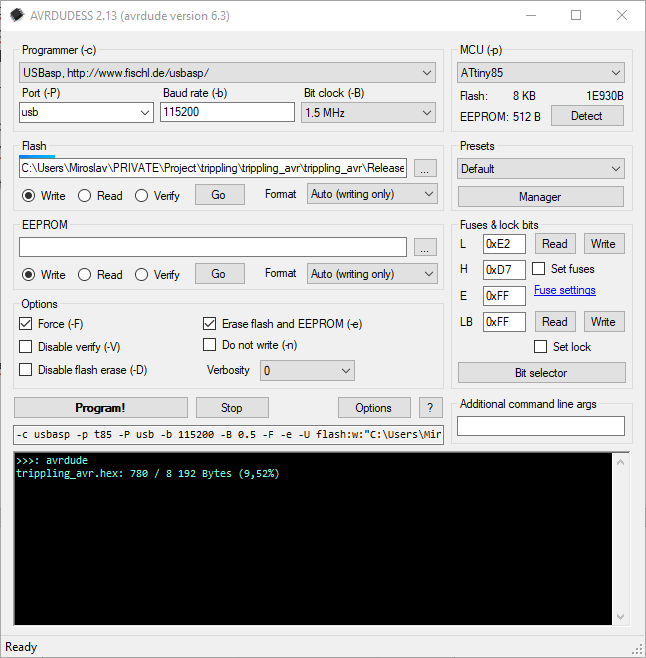
LOCKBIT: 0xFF

Prikladám aj screenshoty nastavenia poistiek z AVR štúdia





Pre nahratie binárneho súboru môžete použiť napríklad AVRDUDE v kombinácii s AVRDUDESS. AVRDUDE je konzolový SW pre nahrávanie binárneho súboru pomocou programátora do MCU, a AVRDUDESS je grafická nadstavba pre AVRDUDE. Môžeme ho stiahnuť napríklad zo stránky <https://blog.zakkemble.net/avrdudess-a-gui-for-avrdude/> . Ako programátor používam USBasp debuger k AVRDUDE a AVR dragon k atmel studiu. AVR dragon podporuje rozšírené možnosti,  napríklad možnosť debugovania SW v bežiacom MCU. USBasp je lacný programátor, ktorý viete kúpiť aj v čínskom e-shope alebo si ho vyrobiť pomocou Arduina. Osobne sa mi z lacných programátorov najviac osvedčil USBasp programátor pre Atmel MCU. Nastavenie AVRDUDESS by v tomto prípade vyzeralo následovne:



Niekedy sa môže stať, že pri kúpe nového procesora ATtiny85 ho musíme prvý krát pripojiť pomocou rozhrania „High Voltage programming“ k programátoru, a nastaviť mu správne poistky. Vytvoriť si vlastný „High voltage“ programátor je už trochu náročnejšie, práve preto využívam originálny programátor AVR dragon. V prípade záujmu viem upraviť konfiguráciu zdrojového kódu podľa požiadaviek, a nahrať vykompilovaný zdrojový kód do MCU ATtiny85. Toto zapojenie viete použiť aj ako dvojkanálový prerušovač pri menšej úprave zdrojového kódu.

# Odkazy

Zdrojový kód, potrebné knižnice, návrh PCB, schému a všetky potrebné podklady + podklady pre výrobu PCB nájdete v mojom git repozitári <https://github.com/johnyHV/ComfortTurnSignal> . Video s ukážkou funkčnosti nájdete na YouTube <https://youtu.be/xPU7Pdd0vOQ>

# Kontakt

miroslav.pivovarsky@gmail.com

Ďakujem za prečítanie článku, dúfam že vás konštrukcia zaujala.