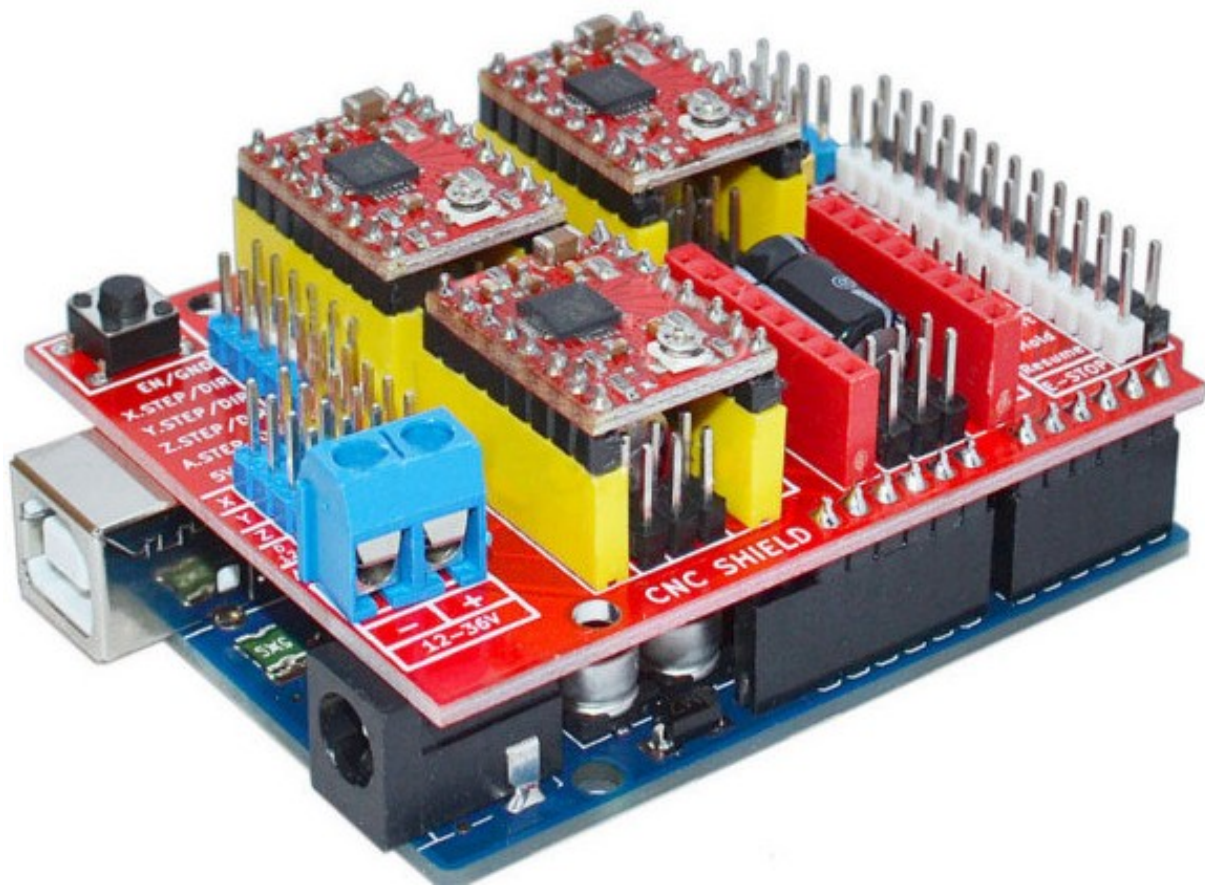


Konfigurace firmware GRBL

Lukáš kořínek – SakulRaider@seznam.cz – www.sakul.cz



Firmware GRBL se používá pro řízení CNC strojů ve spojení s řídicí elektronikou osazenou většinou procesorem ATMEGA328P. Nejčastěji pak jde o desky Arduino UNO a další založené na tomto procesoru. Firmware je dostupný jak ve zdrojovém kódu, tak i již předkompilovaná verze. Já se zaměřím na předkompilovanou verzi a to konkrétně na verzi 1.1. Nicméně následující postup bude aplikovatelný s drobnými odchylkami i na jiné verze.

Stažení firmware:

Firmware lze stáhnout zdarma na [GitHubu](https://github.com). Zde si můžete vybrat požadovanou verzi a zda chcete zkompilovanou verzi (HEX) nebo zdrojové kódy. Jak už jsem psal, já se budu zabývat verzí HEX.

Instalace firmware:

Jak je patrné na úvodním obrázku, budu vše popisovat na desce Arduino UNO, protože je asi nejrozšířenější. Nicméně je samozřejmě možné použít i jinou desku (Arduino NANO, MINI), případně jen samotný procesor ATMEGA328P osazený do Vaší aplikace. Taktéž jsou dostupné již připravené desky pro řízení CNC strojů přímo navržené pro použití s tímto firmwarem.

Takže pro nahrání firmware ve formátu HEX do desky Arduino UNO budeme potřebovat prográmk [XLoader](#). Tento program je naprosto triviální a tak nemá smysl zde nějak rozsáhle popisovat jak s ním pracovat. Prostě s ním nahrajte firmware do desky UNO a je hotovo.

Konfigurace firmware:

Pokud máme firmware úspěšně nahraný v desce (procesoru), přichází na řadu nastavení několika parametrů. Protože každý CNC stroj je jiný je nutno sladit nastavení firmware a Vašeho CNC stroje. Pro názornost budu popisovat všechna nastavení ve spojení s rozšiřující deskou [CNC Shield v3](#) (viz úvodní foto) a mnou navrženou portálovou frézku.

Nejprve musíme znát nějaké parametry použité elektroniky. Na CNC Shieldu jde konfigurovat nastavení pro drivery a to konkrétně nastavení mikrokrokování. Podle použitého driveru můžeme nastavit od Full step až po 128 mikrokroků. Na obrázku v úvodu jsou použity drivery založené na čipu A4988, který podporuje až 16 mikrokroků. Další často používané drivery založené na čipu DRV8825 mohou mít až 32 mikrokroků. Existují i jiné drivery, ale ty se nepoužívají tak často a pokud je hodláte použít, asi víte proč a také jak je správně nastavit. Já doporučuji používat spíše nižší hodnoty mikrokroků. U CNC frézky požadujeme co největší sílu (moment) motorů, která s vyšším nastavením mikrokroků klesá. Já jsem zvolil 8 mikrokroků, které mi poskytnou dobrou přesnost a zároveň rozumnou sílu.

V následujícím seznamu najdete všechny potřebné parametry, které se vztahují k nastavení firmware:

Počet mikrokroků 8 (1-128)

Počet kroků motoru na otáčku 200 (Běžné krokové motory mají 1,8° na krok, což je 200 kroků na otáčku. Avšak pozor jsou i výjimky)

Stoupání trapézového šroubu 8mm/1ot (Já jsem použil trapéz se stoupáním 8mm na otáčku, ale dají se pořídít šrouby i s jiným stoupáním 1,2,4,6,8 atd.)

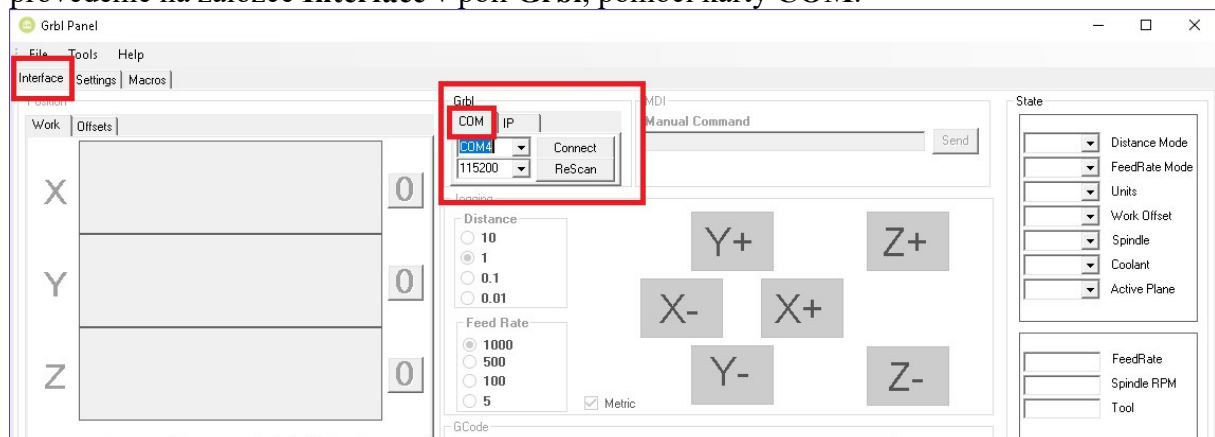
Z těchto údajů můžete vypočítat potřebný počet kroků na posun o 1mm. V našem případě $8(\text{mikrokroky driveru}) \times 200(\text{kroky motoru na otáčku}) / 8(\text{stoupání šroubu/trapézu}) = 200(\text{počet kroků na 1mm}) - (8 \times 200 / 8 = 200)$

Tento údaj budeme muset zadat do firmware.

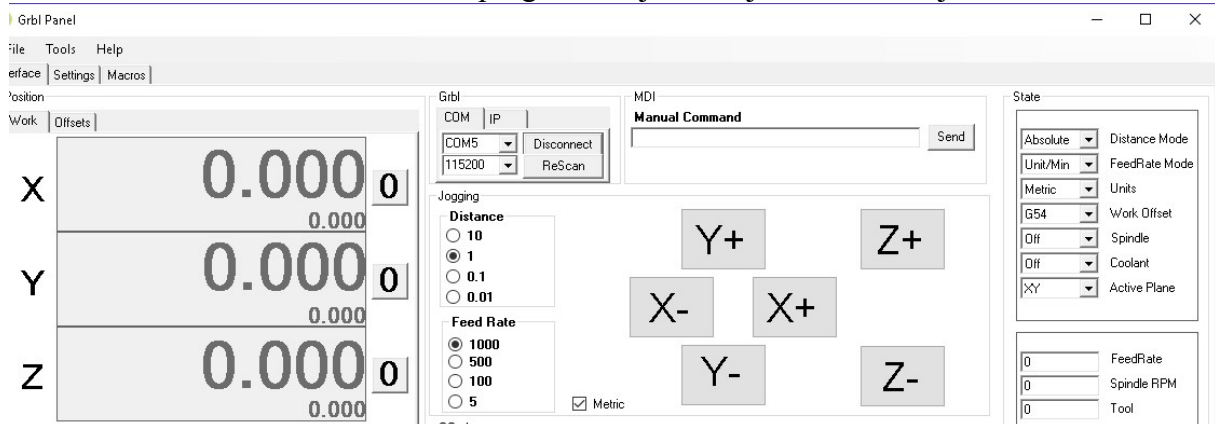
Nyní se tedy konečně dostáváme k samotné konfiguraci firmware. Firmware můžeme konfigurovat v podstatě 3mi metodami:

1. Zeditovat zdrojový kód a všechny potřebné parametry uložit ještě před kompilací firmware. Toto dělat nebudeme, protože používáme již zkompileovaný firmware, který má uloženy nějaké defaultní hodnoty.
2. Pomocí nějakého sériového terminálu. Touto metodou se přepisuje konfigurace uložená v EEPROM procesoru. Nicméně je to poměrně nepřehledné a zdlouhavé.
3. Pomocí nějakého konfiguračního nástroje. Toto je asi nejlepší metoda, kterou zde popíšu.

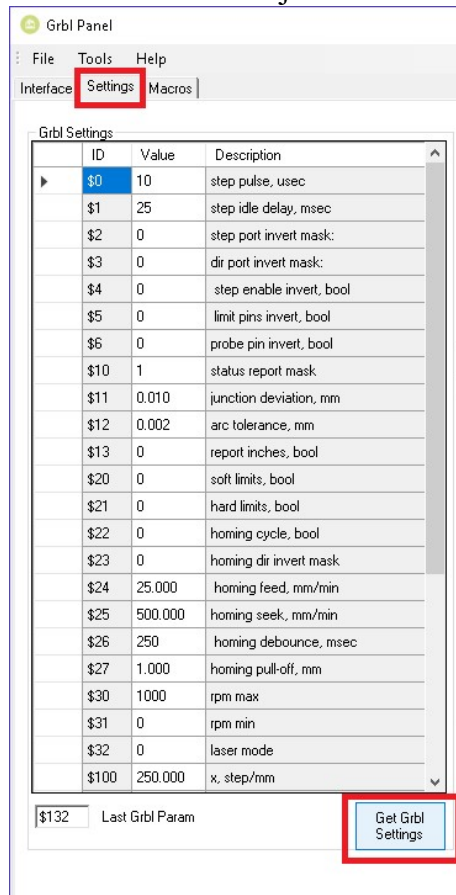
Takže budeme potřebovat program [GrblPanel](#), který je současně později možno použít k ovládání CNC stroje. Po spuštění tohoto programu je nutné navázat komunikaci s elektronikou CNC stroje, tedy s deskou Arduino UNO s nahraným firmware GRBL v1.1. To provedeme na záložce **Interface** v poli **Grbl**, pomocí karty **COM**.



Viz obrázek. Na kartě COM vybereme port, pod kterým máme s PC propojenu desku Arduino UNO (v ukázce com5) a zvolíme komunikační rychlost **115200bps**. Následně stiskneme **Connect**. Pokud je vše v pořádku měl by se program spojit s elektronikou, což poznáme podle toho, že nám naskočí do všech oken programu nějaké údaje. Viz následující obrázek:



Nyní se přepneme na záložku **Settings** a pod levým oknem **Grbl Settings** klikneme na tlačítko **Get Grbl Settings**. Tím se nám do okna načtou všechna nastavení, která můžeme editovat. Viz následující obrázek:



Zde jsou patrné 3 sloupce. **ID**, **Value** a **Description**. Kdy ID nám udává daný parametr a dále popíšu všechny tyto parametry. Value je nastavená hodnota daného parametru a konečně Description je popis daného parametru.

Takže sami vidíte, že na tom nic není a i bez hlubší znalosti můžete již některé parametry nastavit. Důležité je vědět, že nastavovat lze vždy pouze jeden parametr. To provedeme tak, že myší klikneme do pole Value u parametru, který chceme editovat. Následně se nám zobrazí kurzor a můžeme hodnotu upravit. Pokud máme novou hodnotu zapsanu, musíme ji ještě uložit stiskem klávesy Enter. Poté se hodnota uloží do EEPROM procesoru a tabulka se aktualizuje.

Nyní se tedy podíváme blíže na jednotlivé parametry a něco si k nim řekneme. Nebudu popisovat zcela všechny, protože kromě těch, které máme vypsány v tabulce existují i další vesměs řídicí, pomocí kterých komunikuje program Grbl Panel s firmwarem v desce elektroniky (Arduino UNO). Pokud Vás to přesto zajímá, ale i pokud ne, doporučuji prostudovat dokumentaci k firmware [GRBL](#), kde se dozvíte mnoho důležitých věcí.

\$0 – Step pulse, microseconds – Defaultní hodnota je 10

Toto nastavení má spojitost s použitým driverem a určuje minimální délku kroku. V praxi není třeba s touto hodnotou nic dělat, nicméně nadměrné zvýšení hodnoty může způsobit problémy.

\$1 - Step idle delay, sec - Defaultní hodnota je 25 (1-255)

Tato hodnota řídí časovou prodlevu pro vypnutí napájení krokových motorů. To znamená, že pokud se zastaví pohyby, za definovaný čas se vypne napájení motorů. Pro použití s CNC strojem je toto nežádoucí. Potřebujeme totiž, aby byly motory stále pod napětím a tím pádem vlastně zabržděné. Proto je dobré tuto hodnotu nastavit na 255, čímž se zakáže vypínání motorů a okamžitě po zapnutí výkonového napájení stroje dojde k zabrždění všech motorů, čímž se eliminuje riziko nechtěného posunutí osy například při výměně nástroje.

Také se může stát, že některé drivery si po vypnutí napájení motorů nepamatují, v kterém mikrokroku zastavily a při dalším spuštění dojde ke ztrátě původního mikrokrokování, čímž může dojít k desynchronizaci kroků.

Na druhou stranu je třeba říci, že mohou být použity motory, které nemají rádi, pokud jsou zabržděny dlouhou dobu bez pohybu. V takovém případě je dobré nastavit nějakou prodlevu pro jejich vypnutí dle jejich specifikací.

\$2 – Step port invert mask: Default je 0 (0-7)

Toto nastavení dle následující tabulky nastavuje logické úrovně pro řízení driverů krokových motorů. Respektive může invertovat signály **Step**. Toto nastavení se může použít pro speciální drivery. Bližší info viz originální popis [firmware](#). Pro zápis do firmware se používá hodnota **Setting Value**, dle následující tabulky:

Setting Value	Mask	Invert X	Invert Y	Invert Z
0	00000000	N	N	N
1	00000001	Y	N	N
2	00000010	N	Y	N
3	00000011	Y	Y	N
4	00000100	N	N	Y
5	00000101	Y	N	Y
6	00000110	N	Y	Y
7	00000111	Y	Y	Y

\$3 – Direction port invert mask: 0 (0-7)

Tímto nastavením můžeme invertovat signál pro řízení směru pohybu os (DIR). Toto se může hodit, pokud se Vám některá osa točí obráceně a potřebujete ji reverzovat. To může jít i pouhým otočením konektoru, ale někdy to není možné a je nutno to provést ve firmware. Pro reverzaci se používá předchozí tabulka. Takže pokud potřebujeme například reverzovat osu Y (což je u strojů s pohyblivým stolkem běžné) zadáme hodnotu **2**.

\$4 - Step enable invert: 0 (0-1)

Invertuje logickou úroveň pro zapnutí driverů. Záleží na použitých driverech. Při běžně používaných driverech není třeba nic měnit. Hodnota pro invertování je **1**.

\$5 - Limit pins invert: 0 (0-1)

Invertuje logiku vyhodnocení koncového spínače. V běžném režimu (0) pokud je na vstupu koncového spínače logická 1, je to bráno jako rozpojený kontakt a spínač nevybavený. Přivedením LOG 0 se vyhodnotí sepnutí daného koncového spínače. Změna tohoto nastavení může vyžadovat restart procesoru pro uplatnění změny. Bližší info najdete v originální dokumentaci. Pro CNC stroje se prakticky nepoužívá.

\$6 - Probe pin invert: 0 (0-1)

Stejně jako u předchozího parametru s tím, že se jedná o Sondu v ose Z, používanou pro korekci nerovnosti pracovní plochy nebo nastavení nulové polohy osy Z.

\$10 - Status report mask: 1 (1-x)

Toto nastavení určuje, jaká data jsou zpětně hlášena v reálném čase do ovládacího software. Defaultně se odesílají pouze aktuální koordináty stroje. Lze však zapnout mnohem rozsáhlejší diagnostiku. Nicméně čím více dat je posíláno tím může docházet k latencím stroje. Proto v běžném provozu je doporučeno používat minimum těchto dat. Zapnout lze následující data:

Report Type	Value
Machine Position	1
Work Position	2
Planner Buffer	4
RX Buffer	8
Limit Pins	16

Přičemž pokud chceme posílat například data **Machine Position** a **Work Position** zadáme hodnotu **3**. Tedy součet všech položek, které chceme zasílat.

\$11 - Junction deviation: 0.010 mm

Toto nastavení zadává brzdění motorů. Například pokud se dráha nástroje blíží k prudké změně směru, je nutné zajistit, aby motor začal včas brzdit a tím předešel ztrátě kroků. Výpočty správné hodnoty jsou poměrně složité, ale obecně platí, že čím větší nastavená hodnota tím rychleji bude stroj najíždět do změny směru. Pokud tedy dochází ke ztrátě kroků, je dobré tuto hodnotu snížit. Další info najdete [zde](#).

\$12 – Arc tolerance: 0.002 mm

Toto nastavení definuje, jak budou nástrojem vykreslovány kruhové objekty. Z principu není možné, aby se nástroj pohyboval čistě v kruhu, ale vždy se bude pohybovat po křivkách. Tento údaj tak definuje délku této křivky. Čím nižší je hodnota, tím bude kruh vykreslen přesněji neb se bude skládat z většího počtu křivek. Na druhou stranu to povede k vyššímu výpočetnímu zatížení a může dojít ke zpomalení. Ve většině případů není nutno s touto hodnotou nijak manipulovat. Opět pokud se chcete dozvědět více o tomto parametru, podívejte se do originální dokumentace.

\$13 - Report inches: 0 (0-1)

Udává, v jakých jednotkách se zpětně hlásí údaje do software. Defaultně jsou data hlášena v mm, pro změnu na palce zadejte **1**.

\$20 - Soft limits: 0 (0-1)

Toto nastavení udává, zda jsou na stroji zapnuty softwarové koncové polohy. Defaultně jsou vypnuté (**0**), pro zapnutí dejte (**1**). Nicméně pokud je chcete zapnout, je nutné, aby bylo povoleno referování všech os, byl nastaven pracovní prostor všech os a samozřejmě byly na stroji namontovány referenční koncové spínače.

Poté bude načtený kód analyzován a pokud bude zjištěno, že některá osa by se při pohybu měla dostat mimo pracovní prostor bude stroj zastaven vypnuto vřeteno a chlazení s následnou indikací problému.

\$21 - Hard limits: 0 (0-1)

Stejně jako minulý parametr s tím rozdílem, že jsou použity fyzické koncové spínače. Najetí na jakýkoli spínač bude mít za následek okamžité zastavení stroje, vřetene a chlazení. Taktéž bude vyhlášen alarm. Stroj již nebude moci dále vykonávat započatý program a bude nutná nová reference neb zastavení je neřízené a může u něj dojít ke ztrátě krokování na motorech. Zde se doporučuje, pokud je toto nastavení povoleno, zabezpečit tento vstup proti rušení, které by mohlo stroj nekontrolovaně zastavit.

\$22 - Homing cycle: 0 (0-1)

Pokud je tento parametr povolen bude nutné při každém spuštění stroje provést referenci. Tento parametr má i spojitost s houmovacím tlačítkem v ovládacím software. Pokud tato funkce není povolena, bude ovládací software při jeho stisku hlásit error. Bližší info k tomuto parametru najdete opět v originální dokumentaci.

\$23 - Homing dir invert mask: 0 (0-7)

Toto nastavení určuje, kam se rozjedou osy při pokusu o referenci os. Defaultně pojedou Z do plusu, dokud nenajde home pozici. Pak osy X a Y opět do kladné hodnoty. Pokud potřebujete, aby se některá osa referovala do opačné (záporné) hodnoty lze to změnit dle tabulky z bodu **\$2**.

\$24 - Homing feed: 25.000 mm/min

Tento parametr určuje, jakou rychlostí bude provedeno finální najetí referenční pozice dané osy. Pozor není to rychlost, jakou se osa rozjede při hledání koncového spínače (viz následující parametr). Toto je rychlost, na kterou se stroj přepne až poté co koncový spínač najde.

\$25 - Homing seek: 500.000 mm/min

Touto rychlostí se stroj pokusí najít koncový spínač.

\$26 - Homing hebounce: 250 ms

Tímto parametrem se definuje ochrana proti zákmitu mechanického spínače. Běžně stačí zadat 10-30ms. Každý mechanický spínač při sepnutí totiž sepne mnohokrát za sebou, než se ustálí v sepnutém stavu. Protože procesor dokáže číst stav tohoto spínače velice rychle a tyto zákmity by detekoval, nastavuje se krátká prodleva, které tento jev eliminuje.

\$27 - Homing pull-off:1.000 mm

Tento parametr se použije, pokud používáte stejné referenční a koncové spínače, aby se předešlo náhodnému spuštění limitního spínače dané osy.

\$100, \$101 and \$102 – [X,Y,Z]: 250.000 steps/mm

Toto je jeden z nejdůležitějších parametrů. Zde zadáváme počet kroků motoru na 1mm posuvu dané osy. Toto jsem přesně vysvětloval už v úvodu tohoto textu. Včetně vzorce jak to spočítat pro Váš stroj.

\$110, \$111 and \$112 – [X,Y,Z] Max rate: 500.000 mm/min

Tento parametr určuje maximální rychlost dané osy.

\$120, \$121, \$122 – [X,Y,Z] Acceleration: 10.000 mm/sec²

Tento parametr nastavuje akceleraci dané osy. Čím větší číslo, tím rychleji osa akceleruje. Ideální je otestovat každou osu jakou zvládá akceleraci a poté hodnotu o 10-20% snížit.

\$130, \$131, \$132 – [X,Y,Z] Max travel: 200 mm

Tento parametr vlastně definuje software koncové polohy, neboli definuje pracovní prostor stroje. Aby to fungovalo, musí být nastavena funkce **\$22**.

Tímto bychom měly probrány ty nejdůležitější parametry firmware. Netvrdím, že můj popis je 100% a můžou se v něm vyskytovat chyby a nepřesnosti. Proto si veškerá nastavení a jejich popis projděte i na oficiálním webu [zde](#). Taktéž tu nemusí být popsány všechny parametry, neb toto byly parametry z verze firmware 0.9 a u novějších mohou být nějaké další, případně některé pozměněné. Tento materiál má sloužit pouze k prvnímu seznámení s problematikou a hlubší seznámení je již na každém uživateli.

Hodnoty všech proměnných zde uvedené jsou defaultní a nemusí zaručovat, že Váš stroj s tímto nastavením bude pracovat správně. Vždy je nutné provést korekce nastavení přesně pro Váš stroj a jeho parametry.