

3DTouch Auto Leveling Sensor

Uživatelská příručka



Úvod

3DTouch je dotykový snímač úrovně podložky pro 3D tiskárnu, který přesně zaměří povrch tiskové podložky, čímž pomůže zvýšit přesnost tisku.

Senzor 3D Touch je jednoduchý, inteligentní a přesný. Funguje s téměř jakýmkoli druhem podložek, jako jsou sklo, dřevo, kovy atd.

Hlavní úloha senzoru 3DTouch je stejná jako většiny senzorů pro měření úrovně podložky, které používají pro měření mikropsínač a servo. Používá i stejný způsob připojení a stejný komunikační protokol, proto je snadné jej připojit téměř ke všem řídicím jednotkám 3D tiskáren.

Použitím speciálně navrženého solenoidu a hallova senzoru 3DTouch kombinuje vysokou přesnost měření a snadné použití. Přináší více radosti z přesnějšího tisku, usnadňuje kalibraci a přidává spoustu chytrých funkcí jako self-test, detekci planého poplachu a test mód pro M119.

Funkce

Jednoduchý

3D Touch lze snadno nainstalovat. Jednoduše jej nastavíte ve firmwaru, protože 3DTouch používá standardní komunikační protokol stejně jako ostatní senzory.

Inteligentní

Autotest: Při každém zapnutí napájení provede senzor dvojité test, aby vše fungovalo tak, jak má.

Alarm: V případě problému při autotestu nebo při práci se rozblíká kontrolka LED.

Vysoká přesnost

Přesnost měření úrovně je 0,005 mm.

Díky této přesnosti bude tisková hlava v přesně nastavené výšce - nezávisle na teplotě podložky nebo přesnosti kalibrace.

Lepší elektromagnet: úspora energie

V klidovém stavu, kdy je kolík senzoru v krajní poloze (vysunut nebo zasunut) neprotéká solenoidem žádný proud. Díky tomu je odběr proudu v klidovém stavu < 15mA. Odběr při pohybu detekčního kolíku je 300 mA po dobu cca 100 ms. Nižší spotřeba proudu zabraňuje přehřívání senzoru.

Technologie

3D Touch se skládá z Atmel ATtiny13A, solenoidu a pohyblivého dotykového kolíku.

Široký výběr tiskové podložky

3D Touch nepoužívá ani optický ani indukční nebo kapacitní snímač. Dotek měří absolutně mechanickým dotykem. Proto není přesnost měření na materiálu, tvaru ani barvě podložky.

3D Touch využívá Hallova jevu, díky kterému poskytuje vysokou přesnost.

Optimalizovaná struktura: větší detekční vzdálenost

3D Touch je malý a nabitý technologií. Díky nastavení detekční vzdálenosti může být umístěn mnohem výše nad podložkou, než jiné senzory.

3D Touch využívá stávající signály ovládání servo motoru, takže jej jednoduše připojíte místo něj.

Specifikace

Napětí: 5V

Klidový proud: 15mA

Max. proud: 300mA

Délka kabelu: 150 mm

Čistá hmotnost: 10g

Přepavní hmotnost: 25g

Kabel

3pólový: hnědý (-, GND), červený (+ 5V), oranžový (řídící signál)

2-kolíkový: černý (-, GND), bílý (Z min)

Obsah balení

1x 3D Touch Autolevel senzor

1x 3 pinová propojka

1x 2 pinová propojka

2x šroub a matice s podložka

2x 3 pinový kryt konektoru

2x 2 pinový kryt konektoru

3x 1 pinový kryt konektoru

14x dutinka

1x rezistor 10kΩ / 0,125W

Připojení

3D Touch potřebuje toto připojení:

Jeden I/O pro řízení (PWM nebo Software PWM)

Jeden I/O pro Z min (Z sonda)

GND a + 5V

Zapojení:

Hnědá	-, GND
Červená	+5V
Oranžová	data

Černá	-, GND
Bílá	Z-min



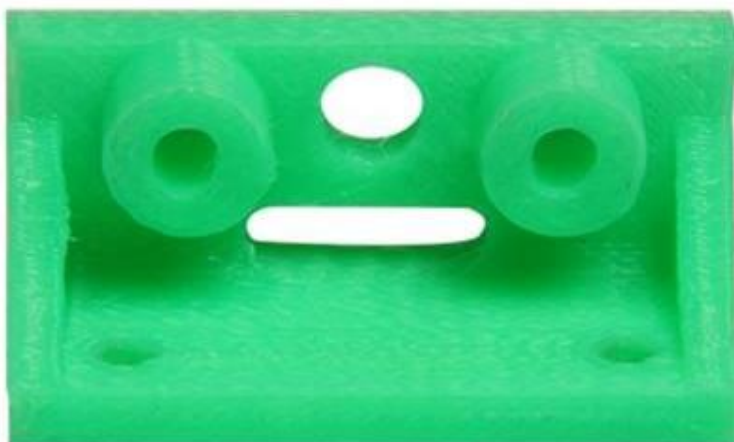
Použití

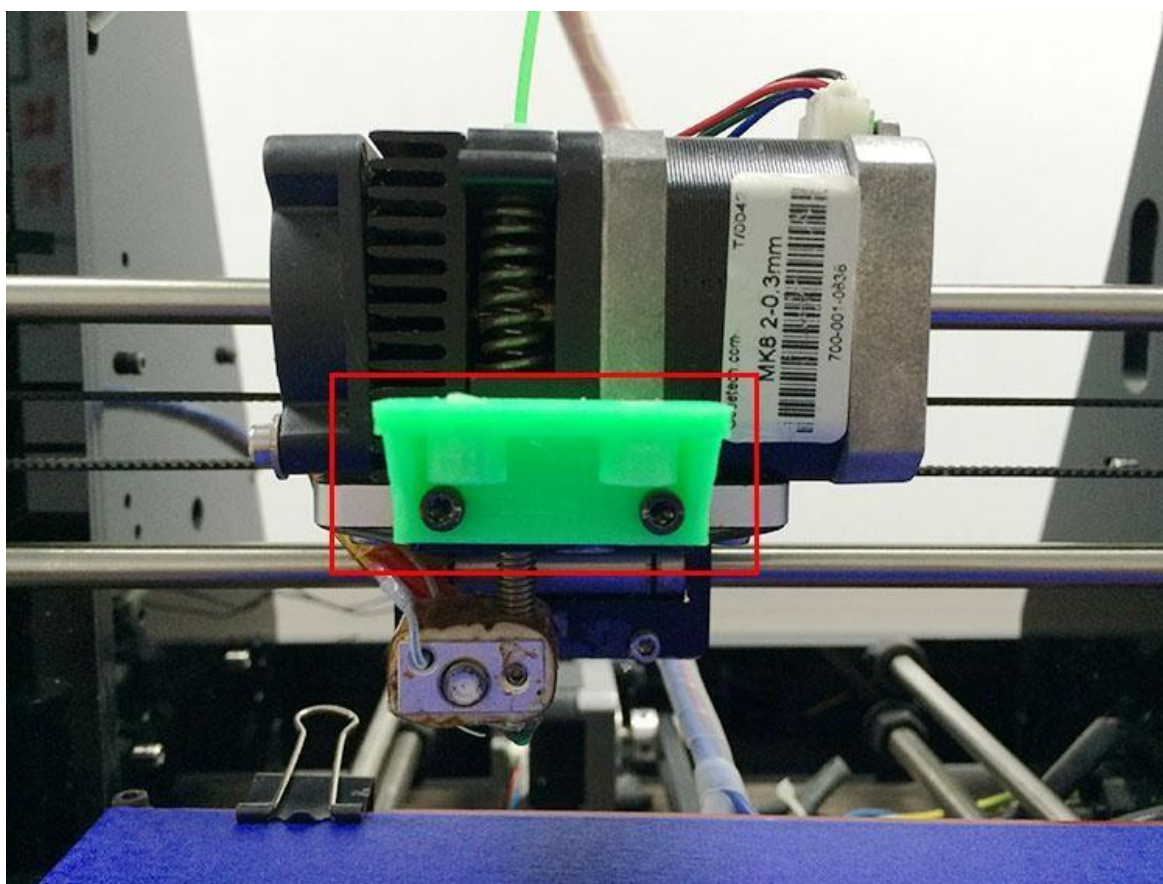
Montáž 3DTouch senzoru

3DTouch je otestován na tiskárnách Geeetech Prusa I3 Pro B, Pro C a Pro X. Dále naleznete montážní postupy pro zmíněné modely. Pro montáž budete potřebovat úchyt, který si můžete vytisknout nebo objednat.

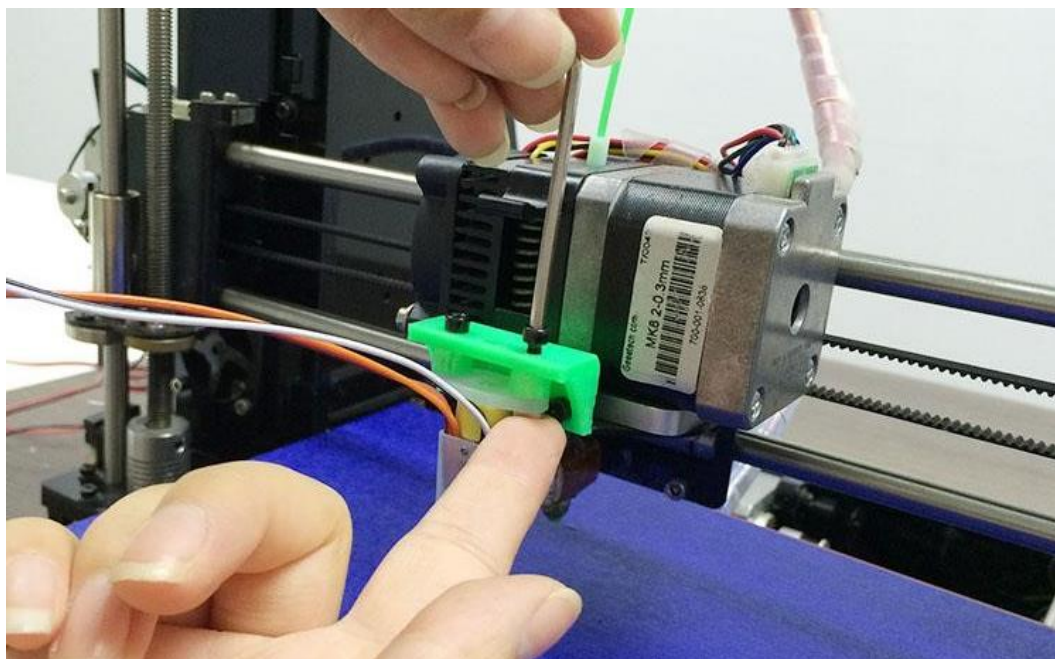
Montáž na Geeetech Prusa I3 pro B

Pro montáž budete potřebovat tento držák. Pokud jej nemáte, můžete si jej stáhnout [zde](#) a vytisknout.

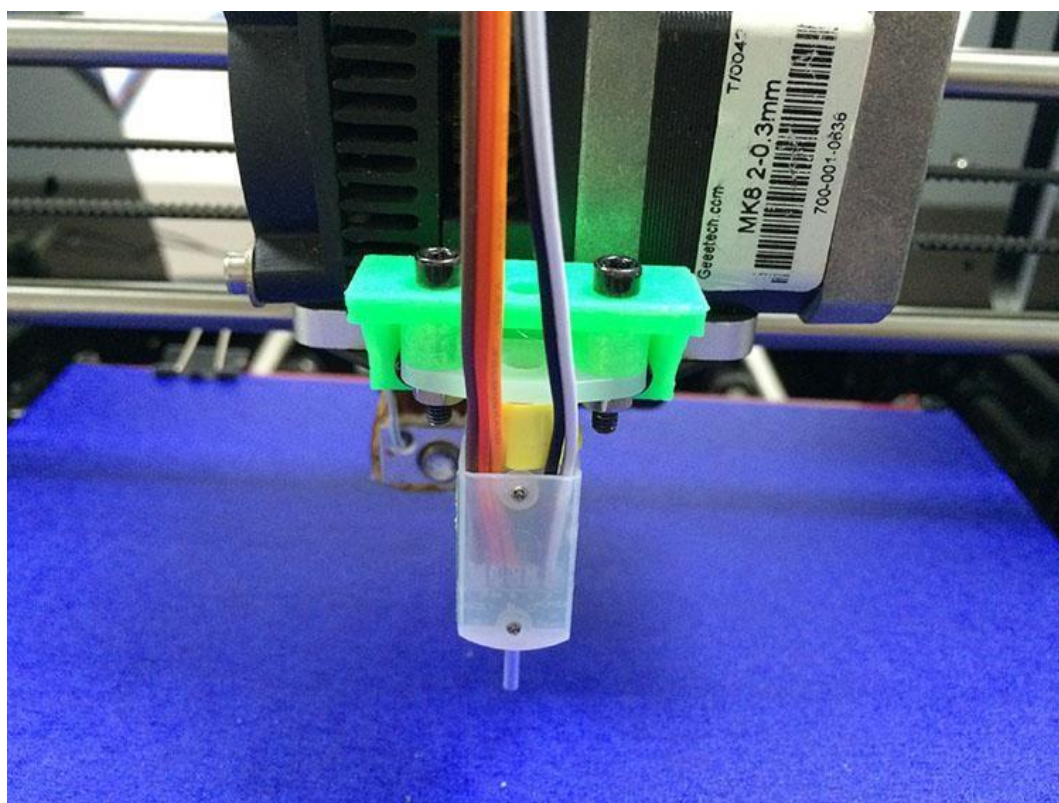




Přišroubujte držák senzoru dvěma šrouby M3x10 k držáku extruderu.



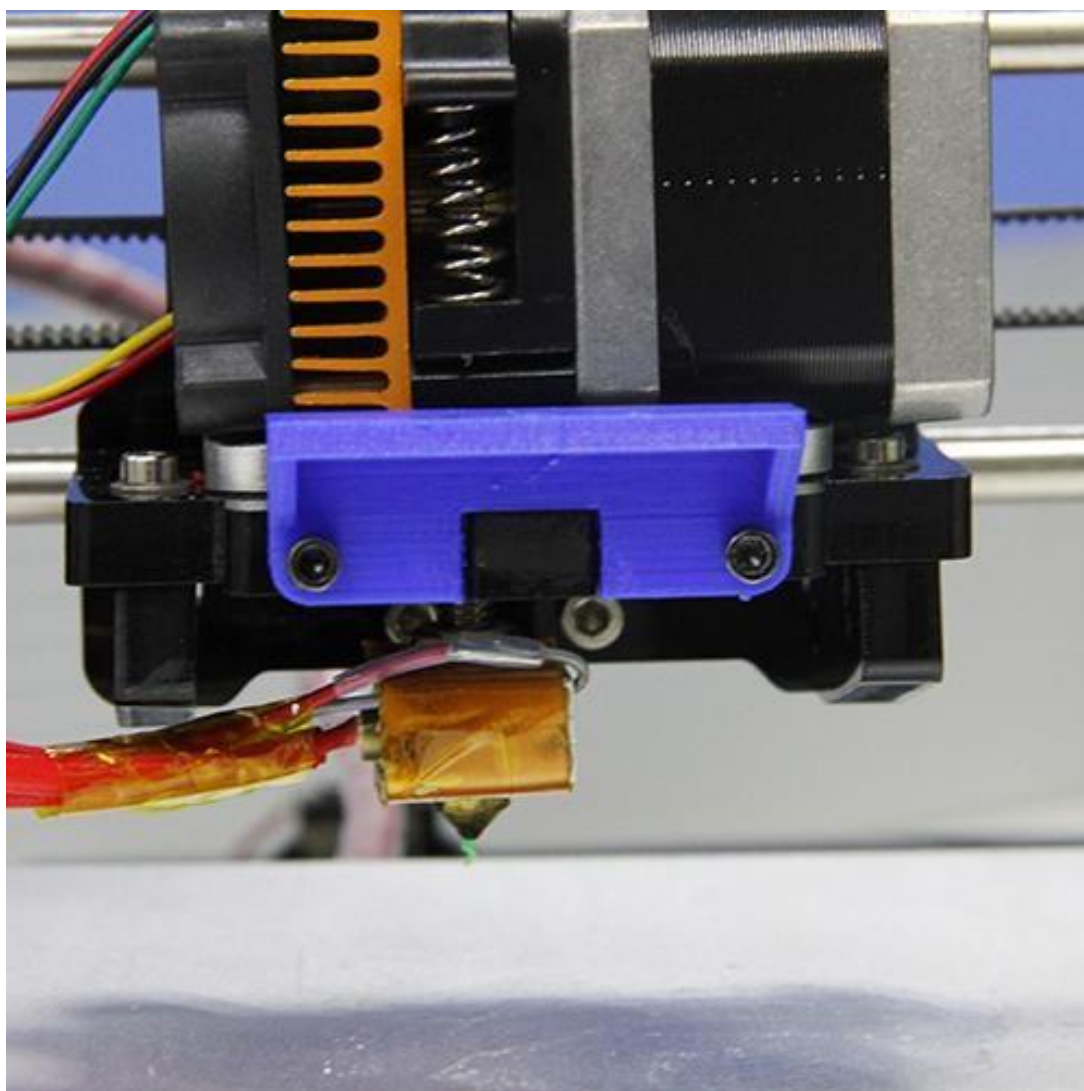
Dvěma šrouby M3x16 s maticí připevněte 3D Touch senzor k držáku.



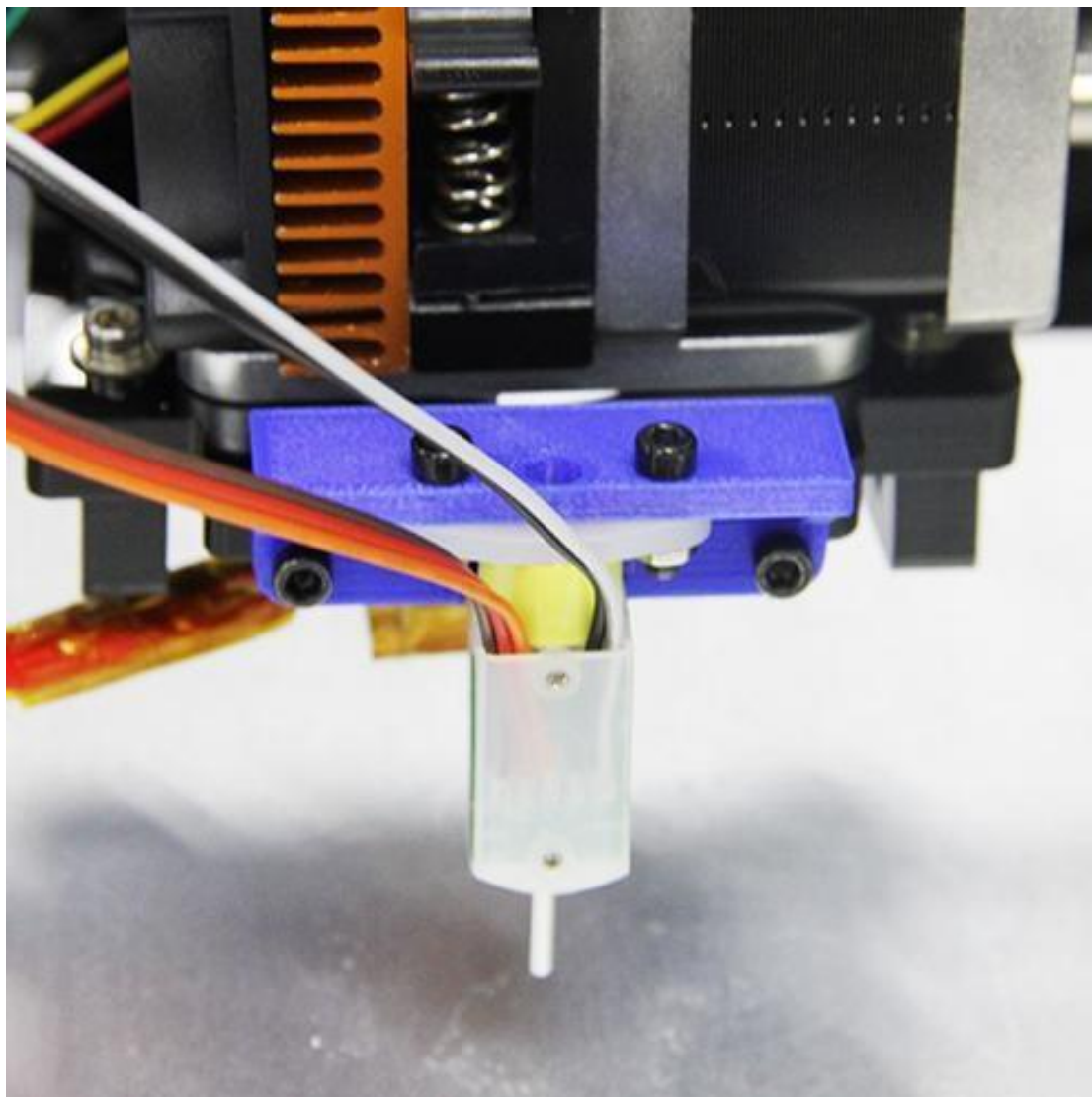
Montáž na Geeetech Prusa I3 pro X

Pro montáž budete potřebovat tento držák. Pokud jej nemáte, můžete si jej stáhnout [zde](#) a vytisknout.

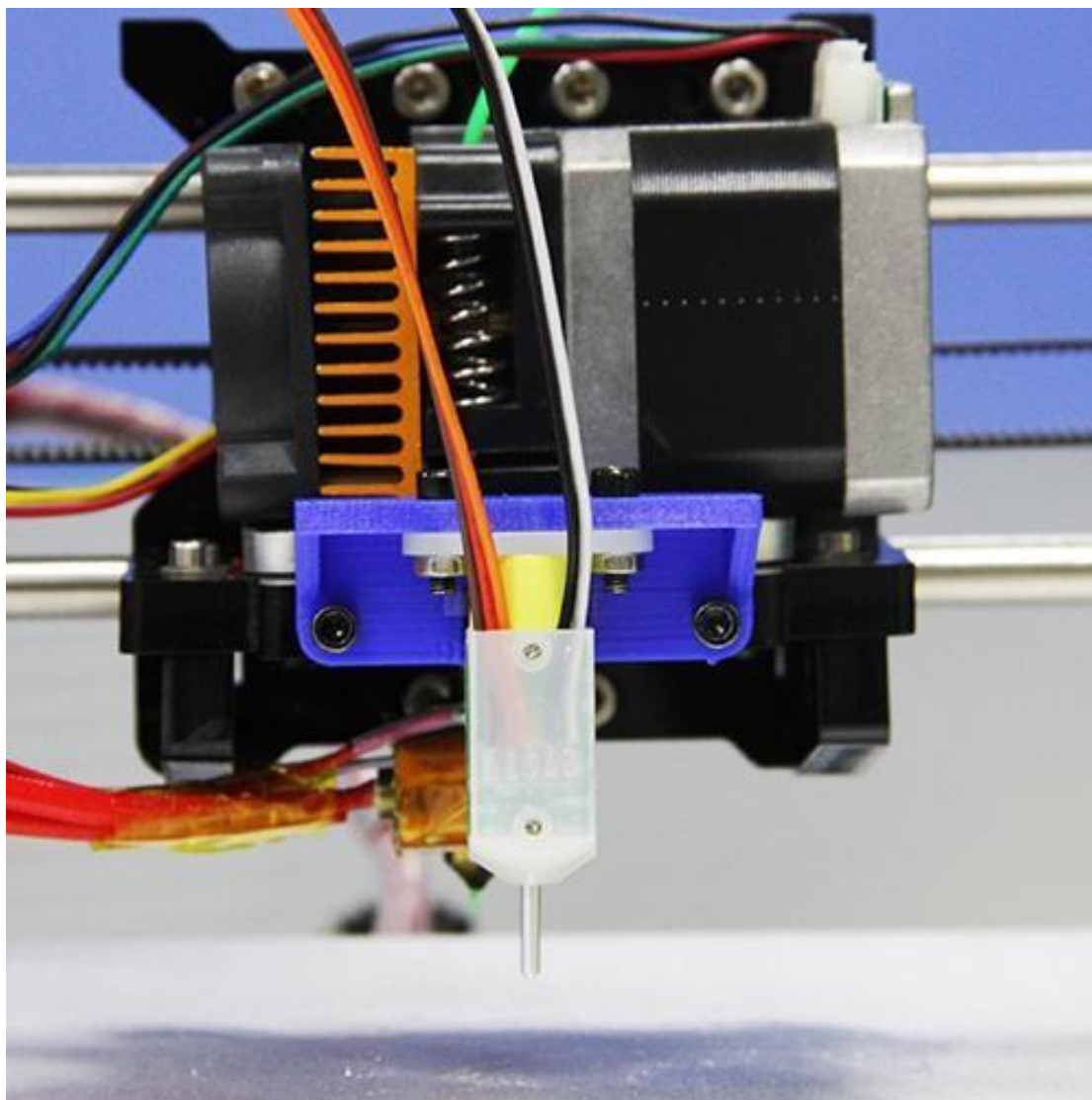




Přišroubujte držák senzoru k držáku extruderu dvěma šrouby M3x10



Přišroubujte 3DTouch senzor k držáku dvěma šrouby M3x16 s maticemi M3



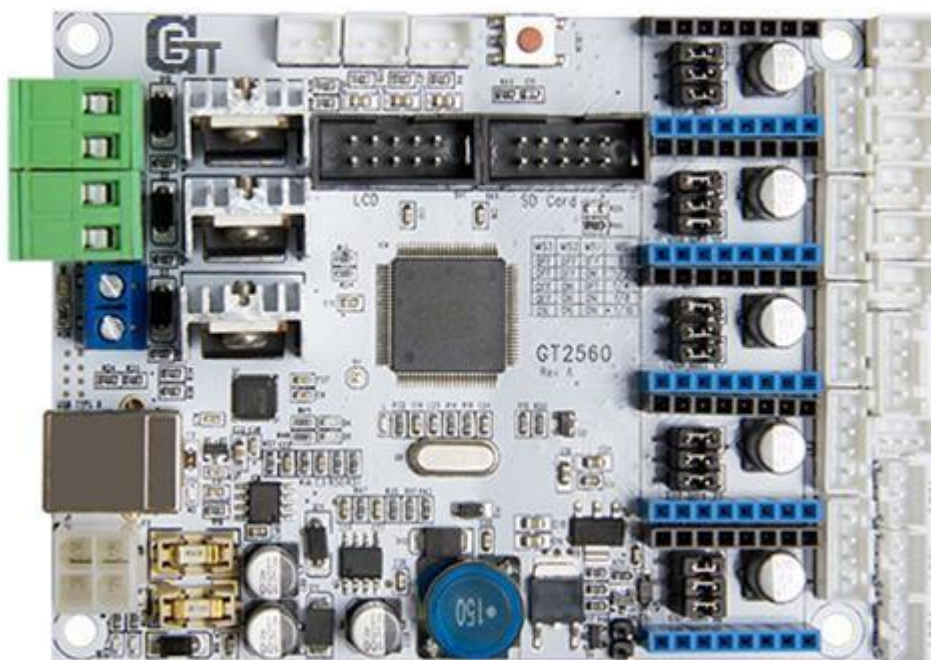
Zapojení

3DTouch senzor má celkem 5 vodičů, data, 5V, GND, Z-min signál a GND.

Pro správnou funkci potřebuje toto propojení:

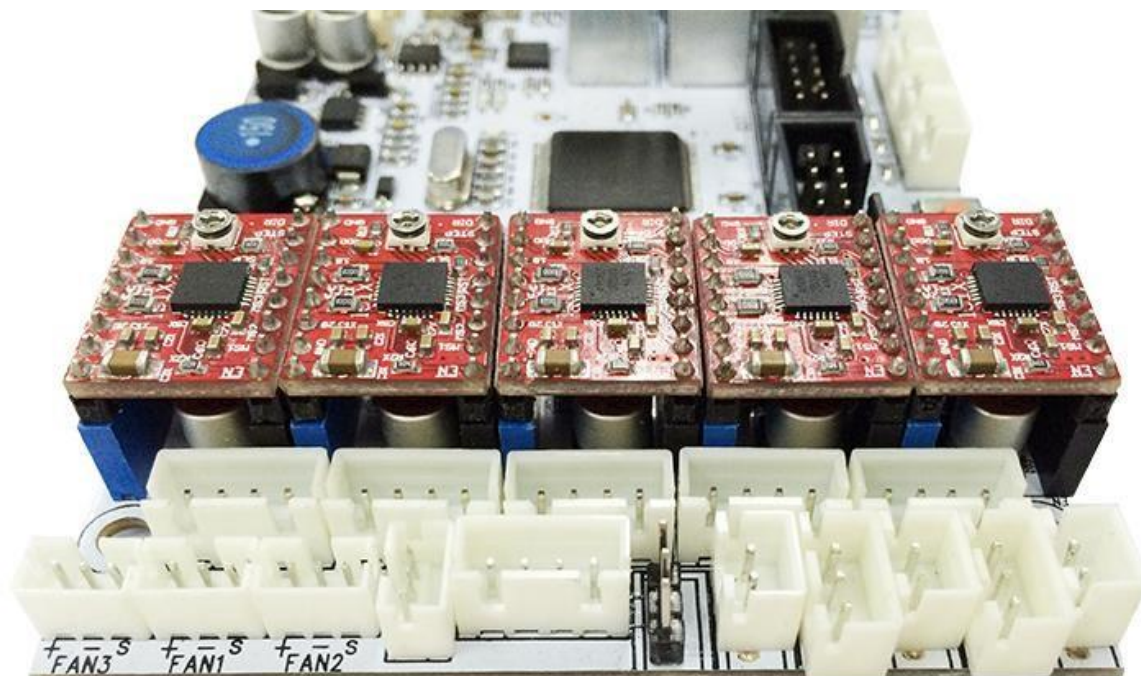
- jeden I/O pro ovládání (PWM nebo softwarové PWM)
- jeden I/O pro Z-min (koncový spínač osy Z)
- GND a napájení +5V

GT2560 Rev. A

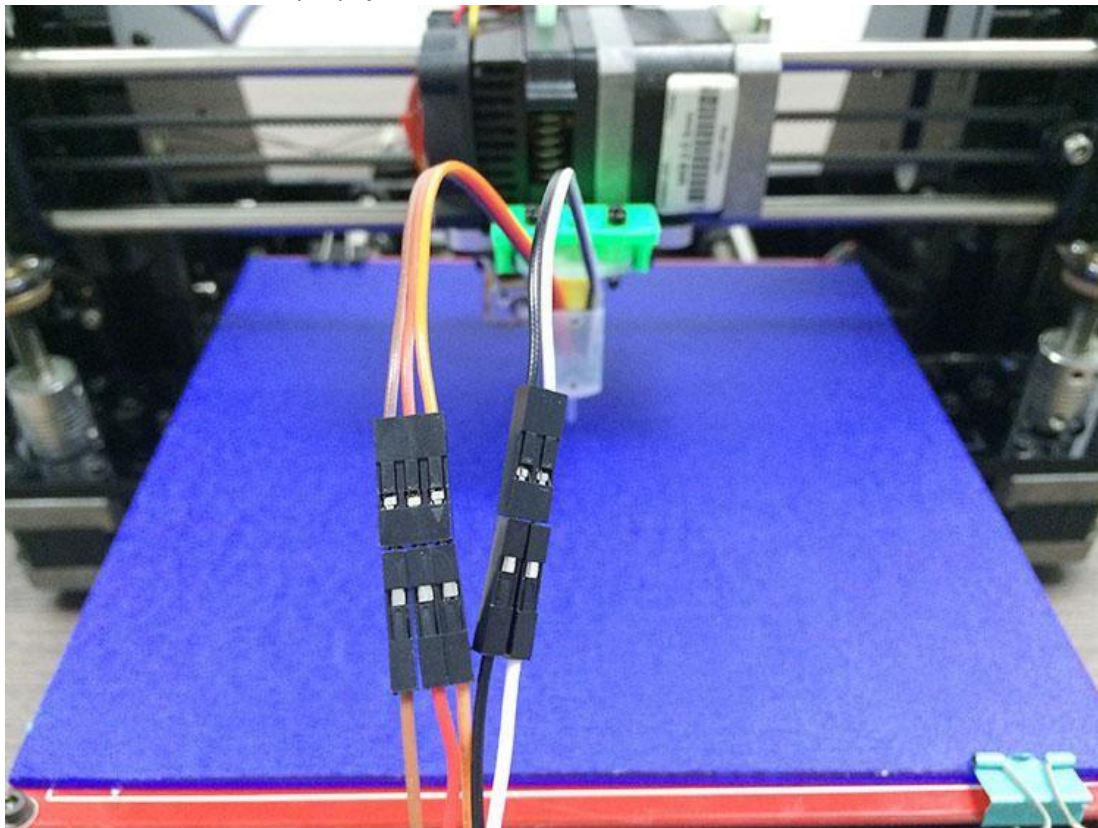


Je několik způsobů, jak senzor připojit k řídící elektronice GT2560. Zde je ten nejjednodušší způsob:

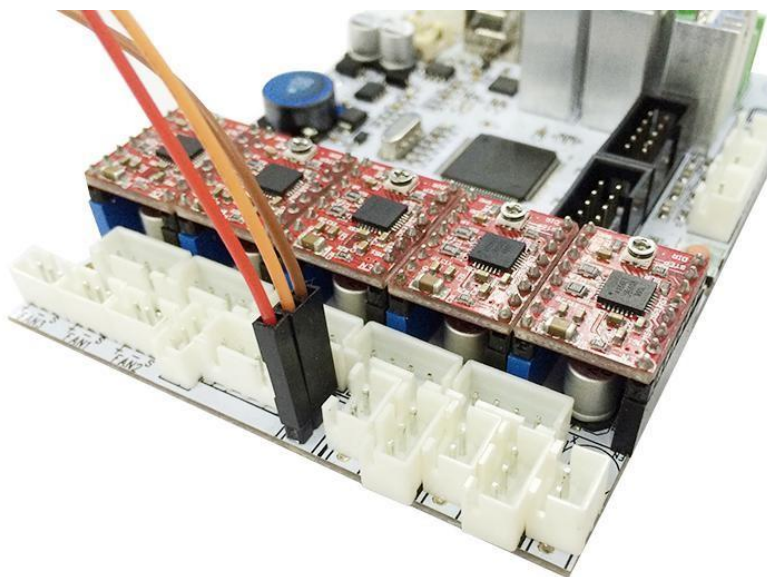
1. Páječkou odpájejte Z – max konektor z řídící desky a místo něj zapájejte 3 pinovou propojku.



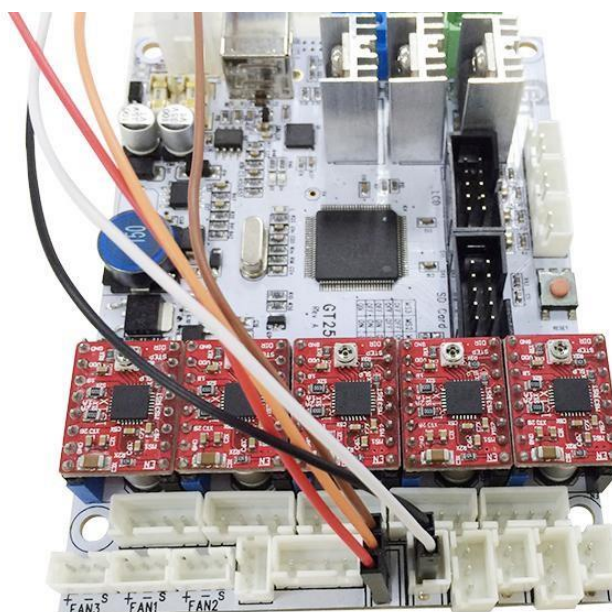
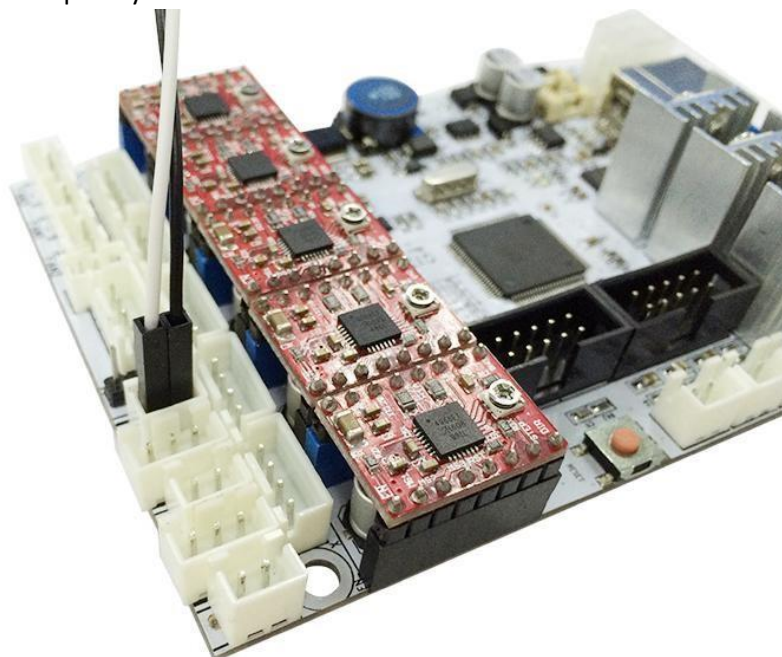
2. Prodlužovacím kabelem propojte senzor s řídicí elektronikou.



3. Prodlužovací kabel zapojte do řídicí elektroniky GT2560. 3 pinový konektor zapojte do Z – max konektoru (hnědá -, GND, červená napájení +5V, oranžová data)

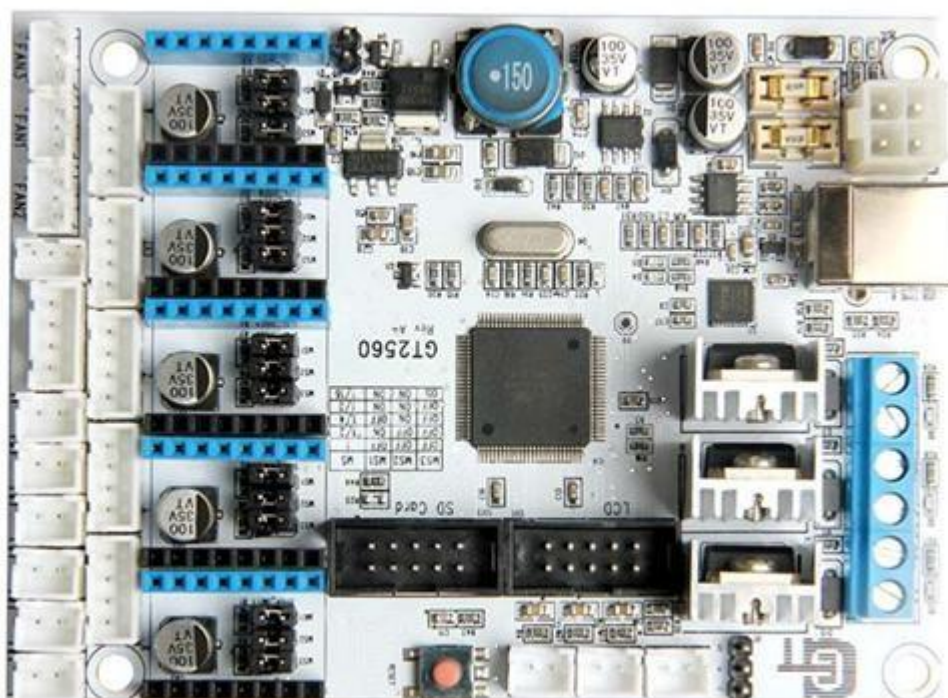


4. Zapojte dvoupinový kabel do konektoru Z – min.



Pozor na správné pořadí vodičů!

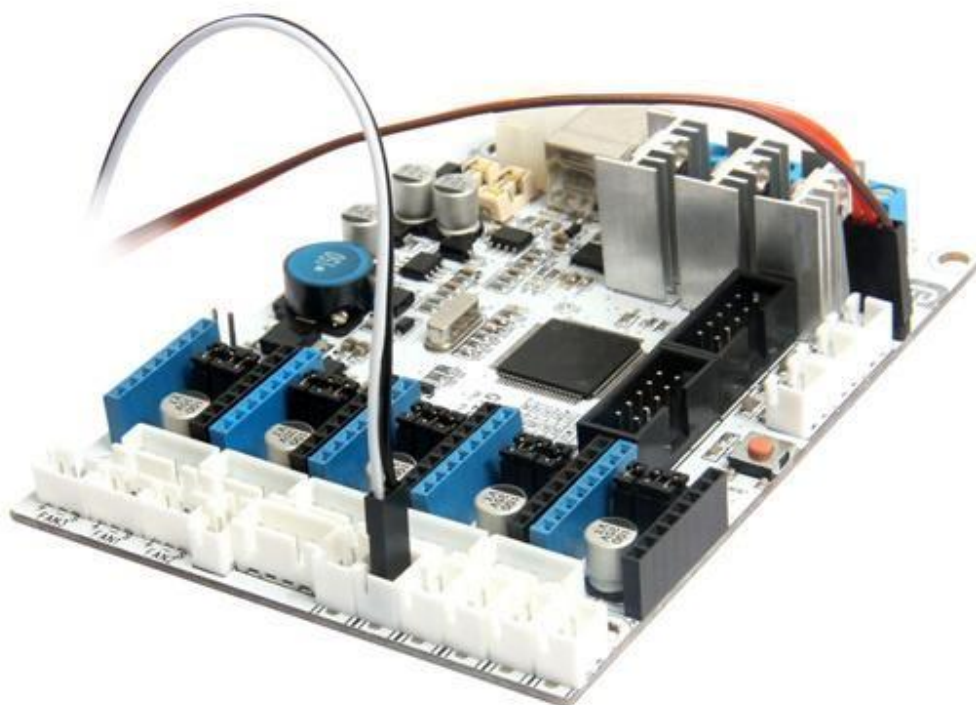
GT2560 rev. A+



Vylepšení:

Na desce GT2560 Rev. A+ je přidán 3pinový konektor a není zapotřebí žádného pájení.





Při použití 3DTouch autorevel senzoru není zapotřebí koncového snímače osy Z. Můžete jej z tiskárny odinstalovat.

Nastavení firmware

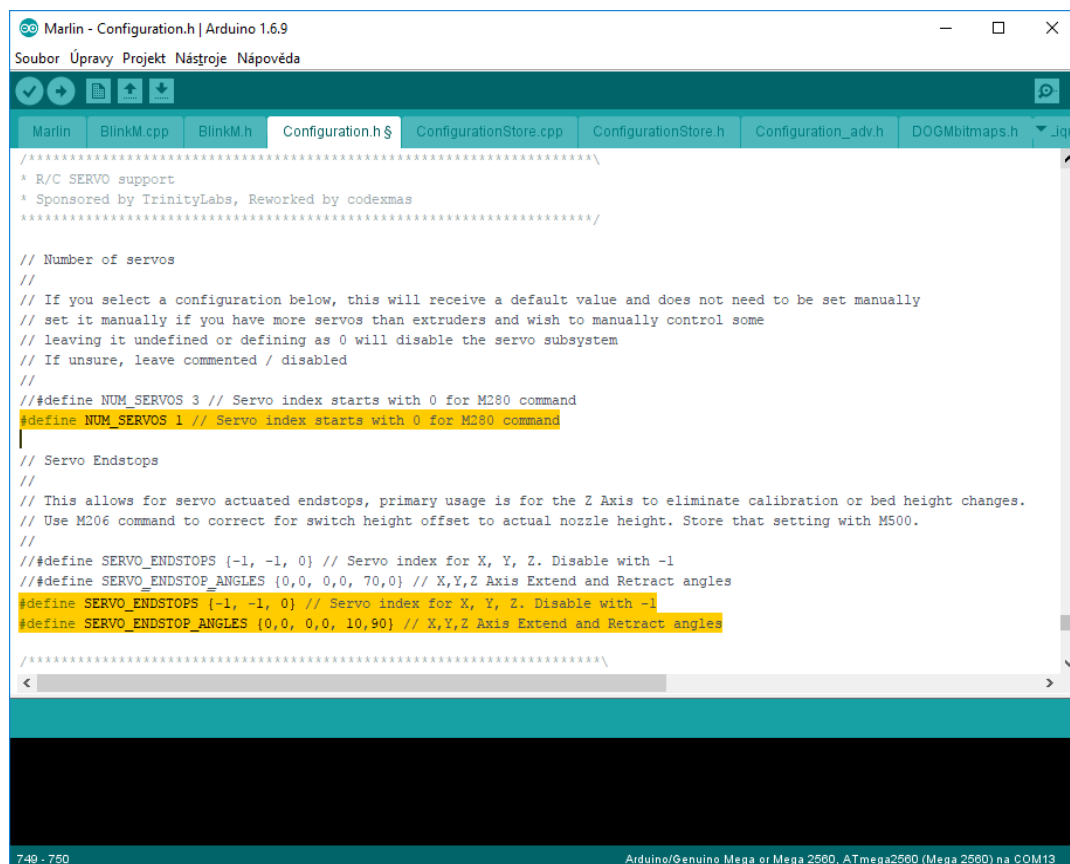
Po instalaci 3DTouch senzoru je potřeba změnit konfiguraci firmware. Změny jsou podobné jako při konfiguraci servo senzoru.

Pokud nechcete provést změnu firmwaru sami, můžete si stáhnout upravený český firmware [zde](#). Ten pouze nahrajte do své tiskárny.

Konfigurace pro modely Prusa I3 Pro B, Pro C a Pro X, je podobná. Stáhněte firmware pro vaši tiskárnu. Českou verzi firmware stahujte [zde](#).

V jednotlivých verzích firmware se může způsob úprav lišit a může být v nesouladu s informacemi níže.

Otevřete firmware v Arduino IDE a změňte následující kód v Configuration.h:



```

*****\
* R/C SERVO support
* Sponsored by TrinityLabs, Reworked by codexmas
*****/

// Number of servos
//
// If you select a configuration below, this will receive a default value and does not need to be set manually
// set it manually if you have more servos than extruders and wish to manually control some
// leaving it undefined or defining as 0 will disable the servo subsystem
// If unsure, leave commented / disabled
//
// #define NUM_SERVOS 3 // Servo index starts with 0 for M280 command
#define NUM_SERVOS 1 // Servo index starts with 0 for M280 command
//
// Servo Endstops
//
// This allows for servo actuated endstops, primary usage is for the Z Axis to eliminate calibration or bed height changes.
// Use M206 command to correct for switch height offset to actual nozzle height. Store that setting with M500.
//
// #define SERVO_ENDSTOPS {-1, -1, 0} // Servo index for X, Y, Z. Disable with -1
// #define SERVO_ENDSTOP_ANGLES {0, 0, 0, 70, 0} // X,Y,Z Axis Extend and Retract angles
#define SERVO_ENDSTOPS {-1, -1, 0} // Servo index for X, Y, Z. Disable with -1
#define SERVO_ENDSTOP_ANGLES {0, 0, 0, 10, 90} // X,Y,Z Axis Extend and Retract angles
*****\

```

Změňte kód následujícím způsobem:

```

/*****\
* R/C SERVO support
* Sponsored by TrinityLabs, Reworked by codexmas
*****/

// Number of servos
//
// If you select a configuration below, this will receive a default value and does not need to
be set manually
// set it manually if you have more servos than extruders and wish to manually control
some
// leaving it undefined or defining as 0 will disable the servo subsystem
// If unsure, leave commented / disabled
//
// #define NUM_SERVOS 3 // Servo index starts with 0 for M280 command
#define NUM_SERVOS 1 // Servo index starts with 0 for M280 command

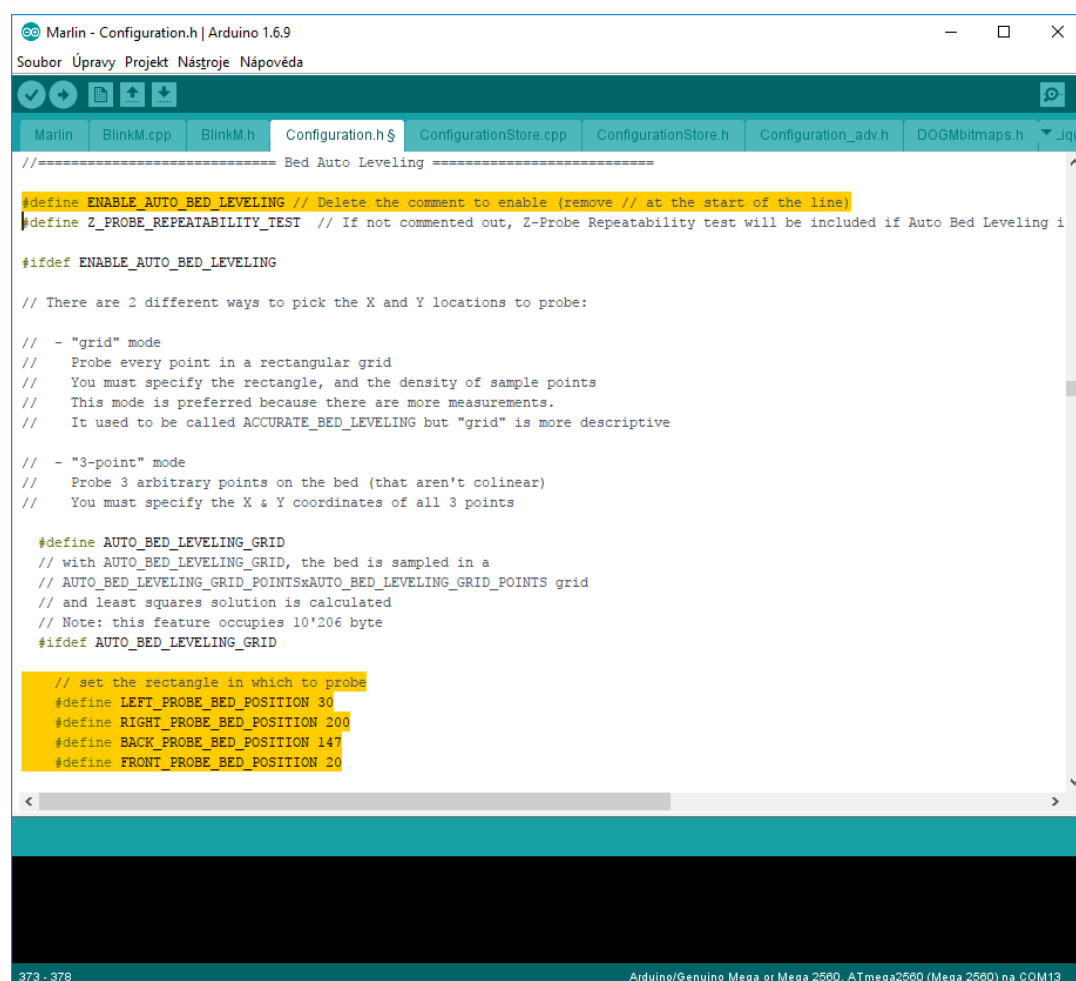
// Servo Endstops
//
// This allows for servo actuated endstops, primary usage is for the Z Axis to eliminate
calibration or bed height changes.

```



```
// Use M206 command to correct for switch height offset to actual nozzle height. Store
that setting with M500.
//
// #define SERVO_ENDSTOPS {-1, -1, 0} // Servo index for X, Y, Z. Disable with -1
// #define SERVO_ENDSTOP_ANGLES {0,0, 0,0, 70,0} // X,Y,Z Axis Extend and Retract angles
#define SERVO_ENDSTOPS {-1, -1, 0} // Servo index for X, Y, Z. Disable with -1
#define SERVO_ENDSTOP_ANGLES {0,0, 0,0, 10,90} // X,Y,Z Axis Extend and Retract angles
```

Najděte kód Bed Auto Leveling v *Configuration.h*



```
//===== Bed Auto Leveling =====
#define ENABLE_AUTO_BED_LEVELING // Delete the comment to enable (remove // at the start of the line)
#define Z_PROBE_REPEATABILITY_TEST // If not commented out, Z-Probe Repeatability test will be included if Auto Bed Leveling is Enabled.

#ifdef ENABLE_AUTO_BED_LEVELING

// There are 2 different ways to pick the X and Y locations to probe:

// - "grid" mode
//   Probe every point in a rectangular grid
//   You must specify the rectangle, and the density of sample points
//   This mode is preferred because there are more measurements.
//   It used to be called ACCURATE_BED_LEVELING but "grid" is more descriptive

// - "3-point" mode
//   Probe 3 arbitrary points on the bed (that aren't colinear)
//   You must specify the X & Y coordinates of all 3 points

#define AUTO_BED_LEVELING_GRID
// with AUTO_BED_LEVELING_GRID, the bed is sampled in a
// AUTO_BED_LEVELING_GRID_POINTSxAUTO_BED_LEVELING_GRID_POINTS grid
// and least squares solution is calculated
// Note: this feature occupies 10'206 byte
#ifdef AUTO_BED_LEVELING_GRID

// set the rectangle in which to probe
#define LEFT_PROBE_BED_POSITION 30
#define RIGHT_PROBE_BED_POSITION 200
#define BACK_PROBE_BED_POSITION 147
#define FRONT_PROBE_BED_POSITION 20
```

```
//===== Bed Auto Leveling =====
```

```
#define ENABLE_AUTO_BED_LEVELING // Delete the comment to enable (remove // at the
start of the line)
```

```
#define Z_PROBE_REPEATABILITY_TEST // If not commented out, Z-Probe Repeatability
test will be included if Auto Bed Leveling is Enabled.
```

```
#ifdef ENABLE_AUTO_BED_LEVELING
```



```
// There are 2 different ways to pick the X and Y locations to probe:

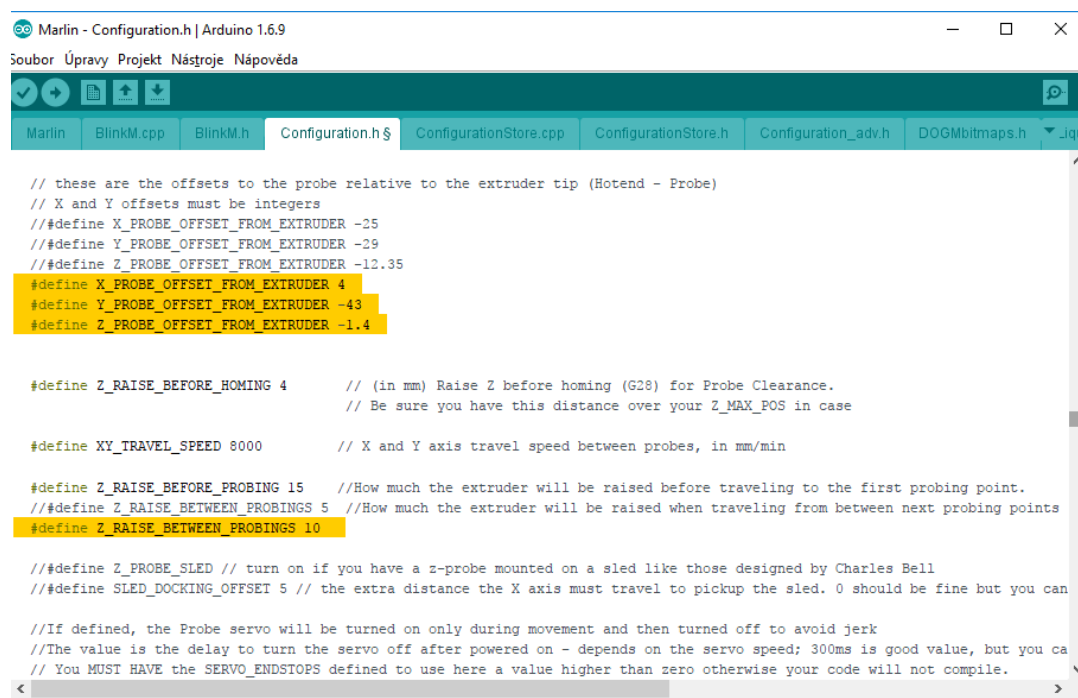
// - "grid" mode
// Probe every point in a rectangular grid
// You must specify the rectangle, and the density of sample points
// This mode is preferred because there are more measurements.
// It used to be called ACCURATE_BED_LEVELING but "grid" is more descriptive

// - "3-point" mode
// Probe 3 arbitrary points on the bed (that aren't colinear)
// You must specify the X & Y coordinates of all 3 points

#define AUTO_BED_LEVELING_GRID
// with AUTO_BED_LEVELING_GRID, the bed is sampled in a
// AUTO_BED_LEVELING_GRID_POINTSxAUTO_BED_LEVELING_GRID_POINTS grid
// and least squares solution is calculated
// Note: this feature occupies 10'206 byte
#ifdef AUTO_BED_LEVELING_GRID

// set the rectangle in which to probe
#define LEFT_PROBE_BED_POSITION 30
#define RIGHT_PROBE_BED_POSITION 200
#define BACK_PROBE_BED_POSITION 147
#define FRONT_PROBE_BED_POSITION 20
```

Kousek níže najdete definici offsetu snímače vůči tryse:



```
// these are the offsets to the probe relative to the extruder tip (Hotend - Probe)
// X and Y offsets must be integers
// #define X_PROBE_OFFSET_FROM_EXTRUDER -25
// #define Y_PROBE_OFFSET_FROM_EXTRUDER -29
// #define Z_PROBE_OFFSET_FROM_EXTRUDER -12.35
#define X_PROBE_OFFSET_FROM_EXTRUDER 4
#define Y_PROBE_OFFSET_FROM_EXTRUDER -43
#define Z_PROBE_OFFSET_FROM_EXTRUDER -1.4

#define Z_RAISE_BEFORE_HOMING 4 // (in mm) Raise Z before homing (G28) for Probe Clearance.
// Be sure you have this distance over your Z_MAX_POS in case

#define XY_TRAVEL_SPEED 8000 // X and Y axis travel speed between probes, in mm/min

#define Z_RAISE_BEFORE_PROBING 15 //How much the extruder will be raised before traveling to the first probing point.
// #define Z_RAISE_BETWEEN_PROBINGS 5 //How much the extruder will be raised when traveling from between next probing points
#define Z_RAISE_BETWEEN_PROBINGS 10

// #define Z_PROBE_SLED // turn on if you have a z-probe mounted on a sled like those designed by Charles Bell
// #define SLED_DOCKING_OFFSET 5 // the extra distance the X axis must travel to pickup the sled. 0 should be fine but you can

// If defined, the Probe servo will be turned on only during movement and then turned off to avoid jerk
// The value is the delay to turn the servo off after powered on - depends on the servo speed; 300ms is good value, but you can
// You MUST HAVE the SERVO_ENDSTOPS defined to use here a value higher than zero otherwise your code will not compile.
```


Změňte ji následujícím způsobem:

```
// these are the offsets to the probe relative to the extruder tip (Hotend - Probe)
// X and Y offsets must be integers
// #define X_PROBE_OFFSET_FROM_EXTRUDER -25
// #define Y_PROBE_OFFSET_FROM_EXTRUDER -29
// #define Z_PROBE_OFFSET_FROM_EXTRUDER -12.35
#define X_PROBE_OFFSET_FROM_EXTRUDER 4
#define Y_PROBE_OFFSET_FROM_EXTRUDER -43
#define Z_PROBE_OFFSET_FROM_EXTRUDER -1.4

#define Z_RAISE_BEFORE_HOMING 4 // (in mm) Raise Z before homing (G28) for
Probe Clearance.
// Be sure you have this distance over your Z_MAX_POS in case

#define XY_TRAVEL_SPEED 8000 // X and Y axis travel speed between probes, in
mm/min

#define Z_RAISE_BEFORE_PROBING 15 //How much the extruder will be raised before
traveling to the first probing point.
// #define Z_RAISE_BETWEEN_PROBINGS 5 //How much the extruder will be raised when
traveling from between next probing points
#define Z_RAISE_BETWEEN_PROBINGS 10

// #define Z_PROBE_SLED // turn on if you have a z-probe mounted on a sled like those
designed by Charles Bell
// #define SLED_DOCKING_OFFSET 5 // the extra distance the X axis must travel to pickup
the sled. 0 should be fine but you can push it further if you'd like.
```

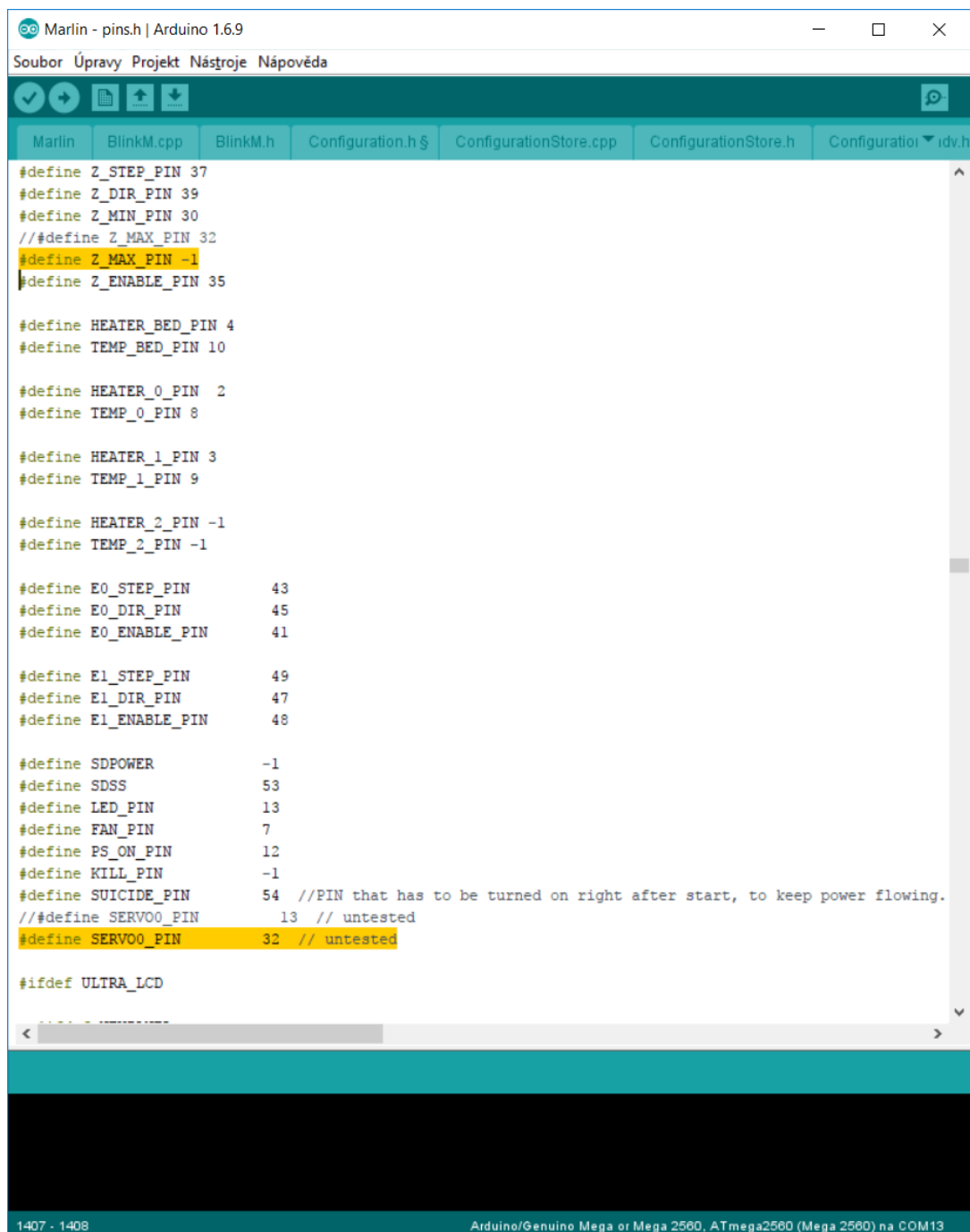
***Pozor:** pro tiskárnu Pro X, použijte následující hodnoty:

```
#define LEFT_PROBE_BED_POSITION 20
#define RIGHT_PROBE_BED_POSITION 190
#define BACK_PROBE_BED_POSITION 165
#define FRONT_PROBE_BED_POSITION 30

#define X_PROBE_OFFSET_FROM_EXTRUDER 4
#define Y_PROBE_OFFSET_FROM_EXTRUDER -50
#define Z_PROBE_OFFSET_FROM_EXTRUDER -3.26

#define Z_RAISE_BETWEEN_PROBINGS 10
```

Najděte následující kód v souboru pins.h:



```

Marlin - pins.h | Arduino 1.6.9
Soubor Úpravy Projekt Nástroje Nápověda

Marlin BlinkM.cpp BlinkM.h Configuration.h$ ConfigurationStore.cpp ConfigurationStore.h ConfigurationStore.h

#define Z_STEP_PIN 37
#define Z_DIR_PIN 39
#define Z_MIN_PIN 30
// #define Z_MAX_PIN 32
#define Z_MAX_PIN -1
#define Z_ENABLE_PIN 35

#define HEATER_BED_PIN 4
#define TEMP_BED_PIN 10

#define HEATER_0_PIN 2
#define TEMP_0_PIN 8

#define HEATER_1_PIN 3
#define TEMP_1_PIN 9

#define HEATER_2_PIN -1
#define TEMP_2_PIN -1

#define E0_STEP_PIN 43
#define E0_DIR_PIN 45
#define E0_ENABLE_PIN 41

#define E1_STEP_PIN 49
#define E1_DIR_PIN 47
#define E1_ENABLE_PIN 48

#define SDPOWER -1
#define SDSS 53
#define LED_PIN 13
#define FAN_PIN 7
#define PS_ON_PIN 12
#define KILL_PIN -1
#define SUICIDE_PIN 54 //PIN that has to be turned on right after start, to keep power flowing.
// #define SERVO0_PIN 13 // untested
#define SERVO0_PIN 32 // untested

#ifdef ULTRA_LCD

```

Změňte jej následujícím způsobem:

```

#define Z_STEP_PIN 37
#define Z_DIR_PIN 39
#define Z_MIN_PIN 30
// #define Z_MAX_PIN 32
#define Z_MAX_PIN -1
#define Z_ENABLE_PIN 35

```

```
#define HEATER_BED_PIN 4
#define TEMP_BED_PIN 10

#define HEATER_0_PIN 2
#define TEMP_0_PIN 8

#define HEATER_1_PIN 3
#define TEMP_1_PIN 9

#define HEATER_2_PIN -1
#define TEMP_2_PIN -1

#define E0_STEP_PIN    43
#define E0_DIR_PIN     45
#define E0_ENABLE_PIN  41

#define E1_STEP_PIN    49
#define E1_DIR_PIN     47
#define E1_ENABLE_PIN  48

#define SDPOWER        -1
#define SDSS           53
#define LED_PIN        13
#define FAN_PIN        7
#define PS_ON_PIN      12
#define KILL_PIN       -1
#define SUICIDE_PIN    54 //PIN that has to be turned on right after start, to keep power
                           flowing.
//#define SERVO0_PIN    13 // untested
#define SERVO0_PIN     32 // untested
```

Pozor: Pro desku GT2560 Rev. A+ použijte následující hodnotu:

```
#define SERVO0_PIN     11 // untested
```

Změny ve firmware uložte a nahrajte do mikroprocesoru desky.

Test

Když poprvé zapnete 3DTouch senzor bude proveden self-test: LED začne blikat, 3x je vsunut a vysunut testovací kolík. Po úspěšném ukončení testu LED dioda trvale svítí. Pokud LED dále bliká, byl při testu zjištěn nějaký problém nebo chyba.

Případnou chybu můžete diagnostikovat těmito G-kódy:

M280 P0 S10; vysune kolík dolů

M280 P0 S90; zasune kolík nahoru

M280 P0 S120; self-test. Ukončení provedete některým z ostatních povelů

M280 P0 S160; ukončení alarmu

Alarm: 3DTouch senzor rozpozná problém a zapne alarm mód – LED dioda bliká. Problém může být např. ve volném pohybu testovacího kolíku.

Kalibrace

Pro správné fungování senzoru je zapotřebí jej zkalibrovat. Jedná se vlastně o pouhé nastavení rozdílu úrovně výšky senzoru a ústí trysky.

Ve svém ovládacím programu proveďte následující:

Zaparkujte tiskovou hlavu do výchozí pozice. Můžete také použít příkaz G28;

G28; zaparkování na výchozí pozici

Vynulujte offset osy Z:

M851 Z0; vynulování offsetu osy Z

Uložte změny do EEPROM:

M500; uložení nastavení do EEPROM

Načtěte údaje z EEPROM zpět:

M501; načte údaje z EEPROM

Zkontrolujte vynulování offsetu:

M503; zobrazí základní nastavení načtené z EEPROM. Údaj Z-offset by měl být 0

Přesuňte tiskovou hlavu do bodu 0 v ose Z:

G1 F60 Z0; přesunutí do bodu 0 v ose Z rychlostí 60mm/s

Pokud je tryska pořád nad podložkou, vypněte sw endstop, aby bylo možné posunovat do záporných hodnot:

M211 S0; vypne sw endstop, který zamezuje posunu do záporných hodnot

Pomocí papírkové metody posuňte trysku do správné výšky nad podložku. K posunování použijte ovládací panel tiskárny nebo váš ovládací program na počítači. Hodnotu, o kterou jste museli trysku posunout si poznamenejte. Dbejte také na znaménko mínus:

Nastavte offset osy Z podle zaznamenané hodnoty z předchozího kroku, včetně znaménka:

M851 Zx.xx; nastaví offset osy z na hodnotu

Zapněte sw endstop:

M211 S1; opět zapne sw endstop

Uložte nastavení do EEPROM:

M500; uloží nastavení do EEPROM

Načtěte hodnoty z EEPROM:

M501; načte nastavení z EEPROM

Ověřte, že vámi zadaná hodnota je nastavena v parametru Z-offset

M503; zobrazí základní nastavené parametry.

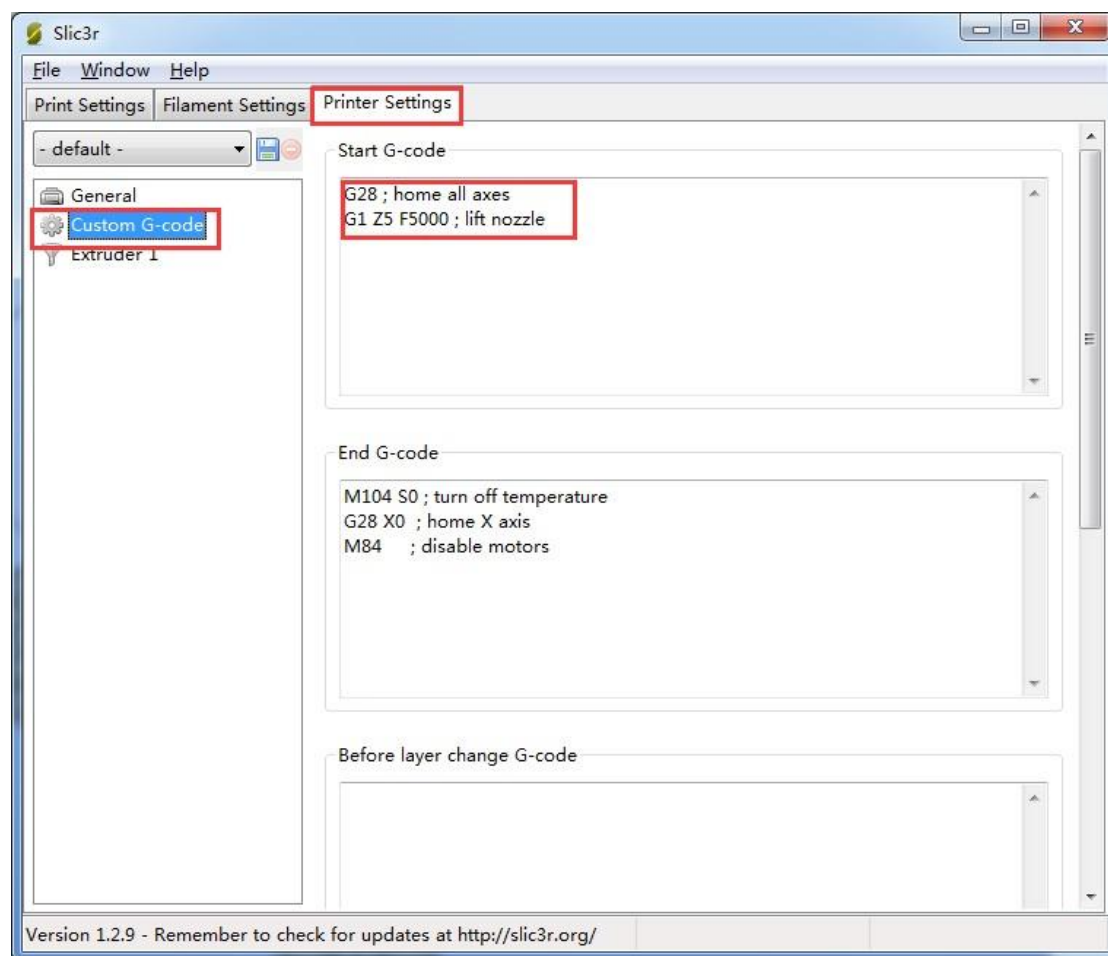
G28; zaparkování do výchozí pozice

G1 F60 Z0; přesunutí do bodu 0 v ose Z rychlostí 60 mm/s

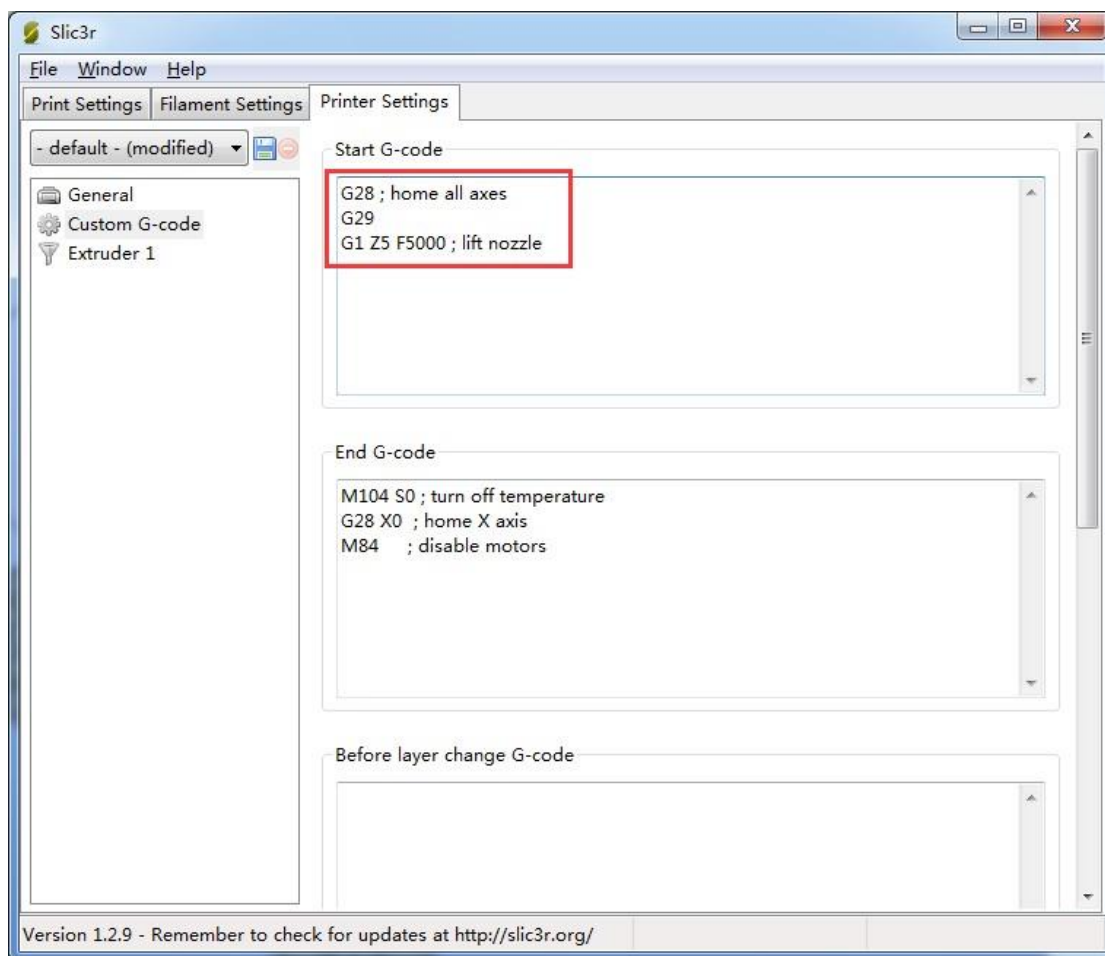
Překontrolujte, že je vzdálenost trysky od podložky správná. V případě potřeby postup opakujte.

Nastavení tiskárny

Po změně konfigurace firmware bude senzor odpovídat stejným řídícím povelům jako jiné typy senzorů, např. indukční, kapacitní nebo IR. Startovací G-code sekvence by měla obsahovat kódy G28 následovaný G29. Tím je zajištěno provedení automatického nastavení úrovně podložky. Startovací sekvenci upravíte ve svém sliceru:



Přidejte kód G29 hned za G28.



Nevkládejte další příkaz G28 za G29. G29 resetuje výsledky nastavení. G29 by měl být spouštěn před každým tiskem.

Videa

[Zde](#) je video použití 3DTouch senzoru na tiskárně Geeetech Prusa I3 pro B a [zde](#) na tiskárně Geeetech Prusa I3 pro X.