

Jak na CoCox & STM32xxDISCOVERY od píky (Win7).

Úvodní kecando (Ize přeskočit)

Rozhodl jsem se, že konečně přejdu z všelijakých komerčních velikostí kódu omezených nástrojů konečně na plnotučné gcc. Od přechodu mě pořád odrazovala příliš velká rozmanitost a zkazky, že všechny ty IDE potřebují příliš mnoho přípravy, než začnou fungovat. Kdežto IAR či KEIL člověk nainstaluje a ono to prostě funguje. Navíc ST dodává příklady k DISCOVERY kitům hlavně pro tyto dva balíky. Pak padl do mého hledáčku balík jménem CoCox. No a netrvalo ani rok a konečně jsem se k tomu kroku odhodlal. Protože se k programování dostávám sporadicky ve vlnách a vím jak je moje hlava děravá, rozhodl jsem se, že si budu dělat poznámky. Přičemž jsem se neprozřetelně pochlubil na zdejší chatu, že jsem to teda konečně rozchodil. A starý dobrák slavo, nebo možná to byl IHVI mně vyprovokoval, ať napíšu článek. Nakonec jsem si řekl, že stejně to sepsat musím alespoň pro sebe a na zdejší webu se mi to bude jednak líp hledat a jednak si tu můžu třeba později povyměňovat pár informací v diskuzi pod článkem.

Když od píky, tak tedy instalace


GCC

CooCox tvrdí, že jako první je potřeba nainstalovat gcc toolchain. Doporučuje použít ARM GCC. ColIDE podporuje i ostatní jako třeba Code Surcery GCC, ale držel jsem se doporučení. Stáhnout se dá [zde](#) No co si budeme vyprávět, kdo by četl manuál, že... Takže není potřeba instalovat první (ověřeno). No nicméně nainstalovat se někdy nakonec musí. Tak si stáhněte .exe spusťte, odsouhlaste prázdné licenční ujednání (ve verzi 4.7 to tak prostě je) a úplně obvykle nainstalujte. Na konci instalace jsem pro jistotu zvolil „Add path to enviroment variable“. Zavřete okna co vyskočí a je hotovo.

IDE

Můžeme instalovat IDE tedy řečí CoCox balíku [ColIDE](#). Dole je doporučení na stáhnutí přes CoCenter. Budiž, stáhneme CoCenter a nainstalujeme. Asi jste si všimli, že se po vás chce registrace, tak ji vyplňte. Slouží k tomu, že pak časem třeba můžete přispívat do repository kódu. Můžete zkusit stáhnout přímo ColIDE, nicméně CoCenter hlídá aktualizace atd. Na konci instalace, pokud necháte zaškrtnuté „Launch ColIDE“, tak zhavaruje na nedostatečné oprávnění. Ničemu to nevadí, prostě si ho pusťte znova z nabídky. Naskočí vám informace, že nemáte nic nainstalováno. Vyberte si co si nainstalujete, pro DISCOVERY příklady od ST je nutné jen ColIDE a CoFlash.

USB ovladač

Zatím, co se stahuje ColIDE, si můžete stáhnout ovladač pro debugger/programátor ST-Link/V2, který je na DISCOVERY kitech přítomen. Dostanete se na něj opět ze stránky ColIDE. Záměrně nedávám link přímo na st.com, protože si jeho stálost tak nějak nejsem jist  . Stažený zip obsahuje instalátor. Dál, snad nemusím vysvětlovat.

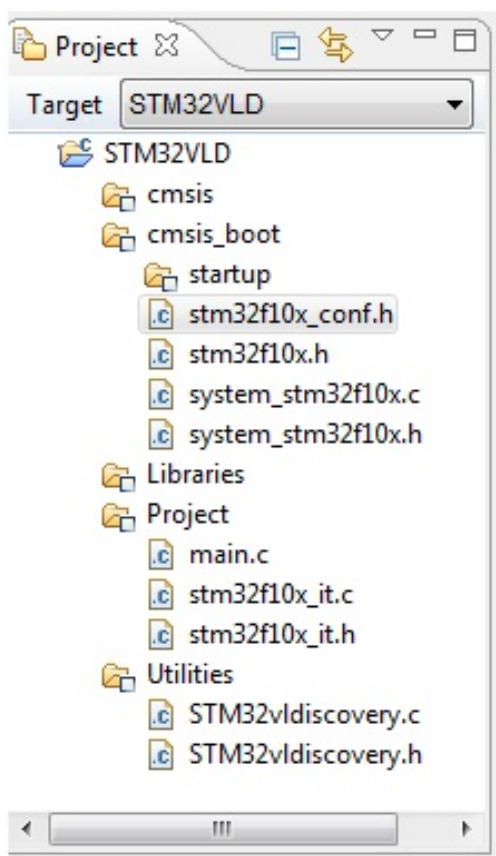
Migrace ukázkových projektů STM32VLDISCOVERY

1. Stáhneme si příklady od [ST](#). Je v sekci Associated Software, Firmware package.
2. Spustíme ColIDE a první, co je potřeba nastavit, je toolchain. Menu Project->Select Toolchain Path. Do textového pole vyplňte cestu k bináře gcc. Pokud jste použili ARM GCC a nechali vše přednastavené, bude vypadat nějak takto [C:\Program Files\GNU Tools ARM Embedded\4.7 2012q4\bin](#). Případně bude v cestě [Program Files\(x86\)](#), pokud máte 64bitovou verzi Win7. Toolchain je 32bitový.
3. Na kartě welcome zvolíme Create a New Project, vymyslíme mu jméno (např. STM32VLD) a zapamatujeme si, kam nám to CoCox uloží. V dalším okně vyberte Chip. Volba board sice má možnost perfektně připravit projekt pro některou ze známých desek, ale zatím jsou k dispozici jen dvě a žádná z nich není DISCOVERY. Dál vyberte ve stromu ST -> STM32F100x -> STM32F100RB (neplést s R8). Tím se v projektu předpřipraví správný #define pro knihovny a linker. Dostáváme se na záložku repository. Zde je potřeba vybrat CMSIS core a CMSIS boot. Periferie zatím necháme plavat, protože příklady od ST mají vlastní ovladače.
4. Teď je potřeba nakopírovat příklady do adresáře zapamatovaného v kroku 2. Zkopírujte obsah adresáře stm32vldiscovery_package, ať není zbytečně moc úrovní, až se v CoCox zorientujete, tak si můžete popřeházet a promazat dle libosti. Mělo by to vypadat nějak takto

c:\CooCox\ColIDE\workspace\STM32VLD*				
↑ Název	Příp	Velikost	Datum	Atributy
[..]		<DIR>	26.02.2013 01:09	—
[_htmresc]		<DIR>	26.02.2013 01:09	—
[cmsis]		<DIR>	26.02.2013 01:00	—
[cmsis_boot]		<DIR>	26.02.2013 01:00	—
[Libraries]		<DIR>	26.02.2013 01:09	—
[Project]		<DIR>	26.02.2013 01:09	—
[Utilities]		<DIR>	26.02.2013 01:09	—
main	c	54	26.02.2013 00:59	-a-
Release_Notes	html	22 896	13.09.2010 14:52	-a-
stm32f10x_an3268_fw	chm	762 655	13.09.2010 15:28	-a-
STM32VLD	coproj	3 733	26.02.2013 01:00	-a-

5. Od teď budu považovat adresář, jehož obsah je na obrázku za vztažný. Odeberte soubor main.c z projektu v ColIDE, je vygenerovaný a stejně v něm nic není (pravé tlačítko na main.c a Delete) Pro přehlednost je dobré ho smazat i z disku. Je to ten, co je vidět na obrázku výše.

6. Teď je potřeba přidat další soubory do projektu k těm, které ColIDE vytvořilo při konfiguraci. Pro přehlednost jsem si vytvořil skupiny Libraries, Project, Utilities podle adresářů na obrázku výše (opět přes pravé tlačítko). Do skupiny Libraries přidejte všechny soubory z adresářů Libraries\STM32F10x_StdPeriph_Driver\inc a Libraries\STM32F10x_StdPeriph_Driver\src. Do skupiny Project přidejte všechny soubory z adresáře Project\Demo\src a pouze Project\Demo\inc\stm32f10x_it.h. Do skupiny Utilities vás nechám domyslet 😊. Pokud nechcete mít hlavičkové soubory součástí, lze jejich cestu dodefinovat v konfiguraci. Otevřete soubor stm32f10x_conf.h ze skupiny cmsis_boot. A pro jednoduchost odkomentujte všechny #include. Nebo se podívejte do Project\Demo\inc\stm32f10x_conf.h a odkomentujte to stejně. Takto by měl vypadat výsledek na záložce Project.



7. Nyní je potřeba dokonfigurovat zbytek. Menu View->Configuration. Na kartě Compile do pole Defined Symbols je potřeba přidat USE_STDPERIPH_DRIVER - jinak je nám prd platné, že si odkomentujeme periferie v předchozím bodě. Do pole Includepaths si můžete přidat cesty k hlavičkovým souborům, pokud jste je nepřidali do projektu. No můžete... musíte, pokud má kompilace proběhnout. Na kartě download si můžete odškrtnout Verify After Download pro zrychlení práce při debugu. Ostatní je připravené v pořádku.

8. Můžeme zavřít záložky Configuration, Repository, welcome. Kdyby něco, jsou v menu View.

9. Chvilka napětí... Project -> Build a tadáááá

10. Debug -> Debug a ... s tím si dělejte, co chcete 😊

Závěrem

Nojo omlouvám se, je to dlouhý jak tejdén před výplatou, ale holt, kdo to chce na podnosu, musí trpět! Když mě budete přemlouvát, tak možná dám dohromady, jak na to s STM32F3DISCOVERY. STM32F3 totiž ještě není v CooCox podporovaná. U STM32F0DISCOVERY zradu nepředpokládám. STM32F4DISCOVERY je v pořadí.