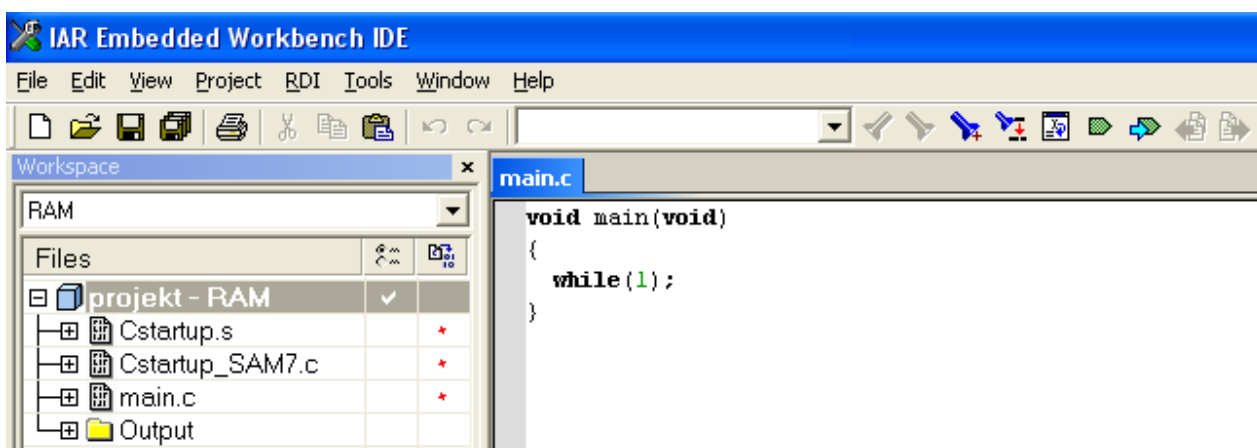


Kompilácia a ladenie v IAR


Príručka sa zaoberá stručným popisom kompilácie projektu v prostredí IAR a ladením vytvoreného programu v RAM aj FLASH pamäti. V závere je uvedený alternatívny spôsob zápisu vytvorenej aplikácie do FLASH pamäte mikrokontroléra.


1. Kompilácia projektu

Projekt je prakticky vždy zložený z viacerých zdrojových súborov. Jednotlivé súbory je možné kompilovať samostatne, alebo je možné skompilovať všetky súbory naraz. Kompilátor zvyčajne kompiluje len tie súbory, ktoré boli zmenené od poslednej kompilácie. Ak bol súbor zmenený, je označený červenou hviezdikou:



Spustenie kompilácie je možné viacerými spôsobmi: pomocou klávesových skratiek, výberom položky z menu alebo stlačením príslušnej ikony.

Kompiláciu aktuálneho súboru je možné spustiť stlačením tlačidiel Ctrl+F7, výberom z menu „Project→Compile“ alebo pomocou ikony .

Kompiláciu všetkých neskompilovaných súborov a vytvorenie výsledného programu spustíme tlačidlom F7, výberom z menu „Project→Make“ alebo pomocou ikony .

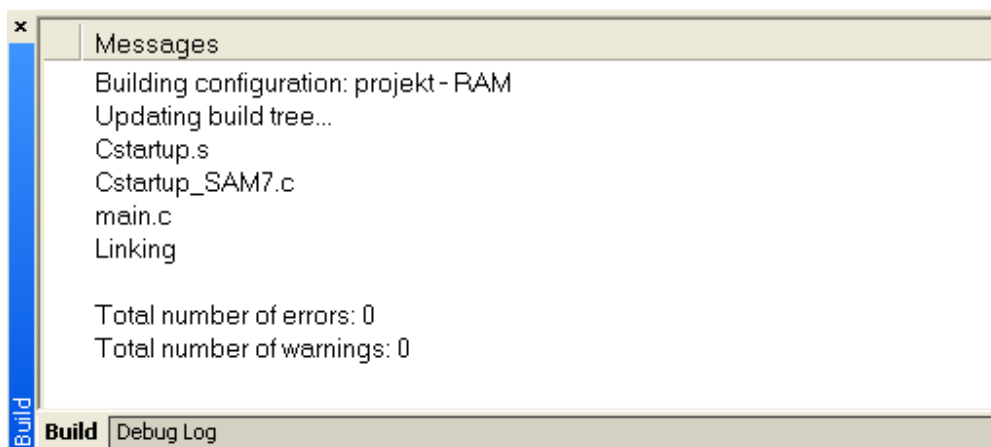
Výberom položky menu „Project→Rebuild All“ je možné vynútiť skompilovanie všetkých súborov projektu, aj tých, ktoré neboli zmenené.

Odstránenie objektových súborov, ktoré sú výsledkom kompilácie, je možné výberom položky „Project→Clean“. Objektové súbory môžeme odstrániť aj ručne vymazaním príslušného podadresára. Pre každú konfiguráciu je v hlavnom adresári vytvorený podadresár, ktorý obsahuje všetky objektové súbory. Názov podadresára je zhodný s názvom konfigurácie (RAM, FLASH, Debug, Release).

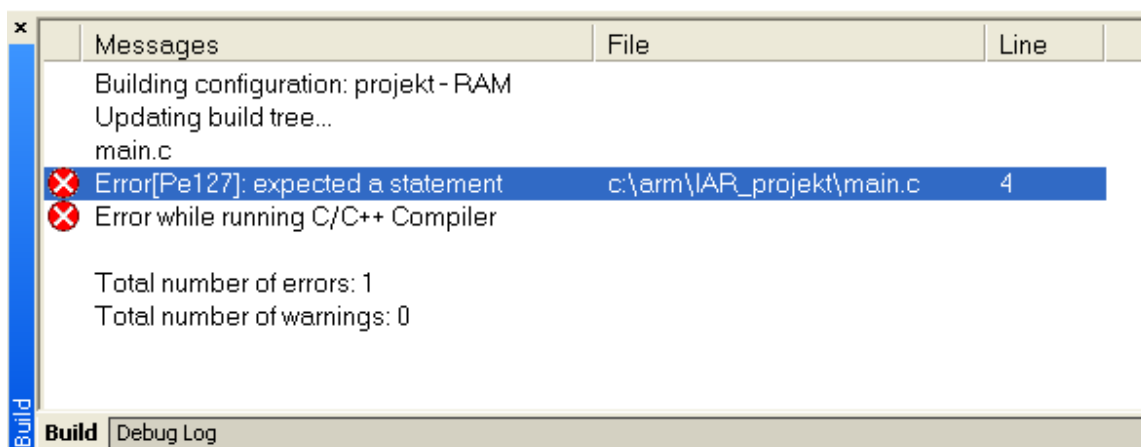
Počas kompilácie sú v spodnej časti hlavného okna vypisované informácie o priebehu kompilácie, varovania a prípadné chyby.

Skompilovaním nášho ukážkového projektu by nemali vzniknúť žiadne chyby.

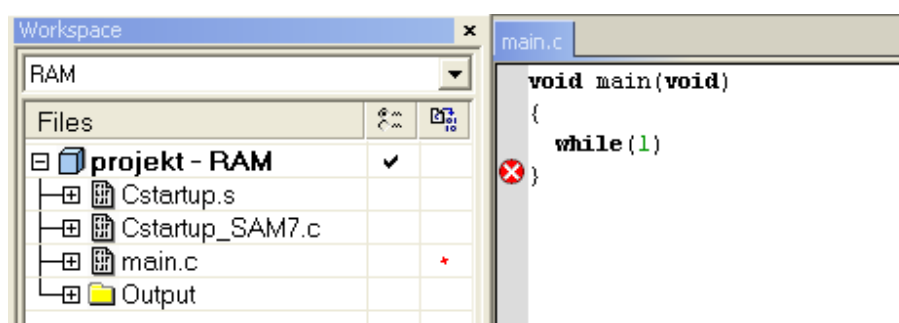
Nasledovný obrázok zobrazuje hlásenia kompilátora a linkera:



V prípade vzniku chyby (napríklad chýbajúca bodkočiarka za príkazom `while(1)`), zobrazí sa aj informácia o čísle riadku, v ktorom bola nájdená chyba:



Dvojitým kliknutím na chybové hlásenie sa kurzor nastaví na príslušné miesto v programe, ktoré je tiež označené značkou ✘.

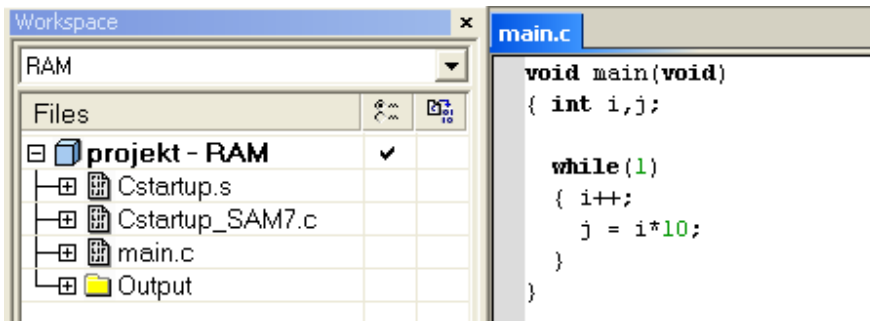


2. Ladenie projektu


Aby bolo ladenie programu možné, musia byť splnené nasledovné podmienky:

- Demonštračný modul ARM7 je pripojený na napájanie,
- modul ARM7 je prepojený plochým 20-žilovým káblom s modulom na podporu programovania,
- modul na podporu programovania je prepojený s počítačom prostredníctvom paralelného kábla,
- je spustený program H-JTAG.

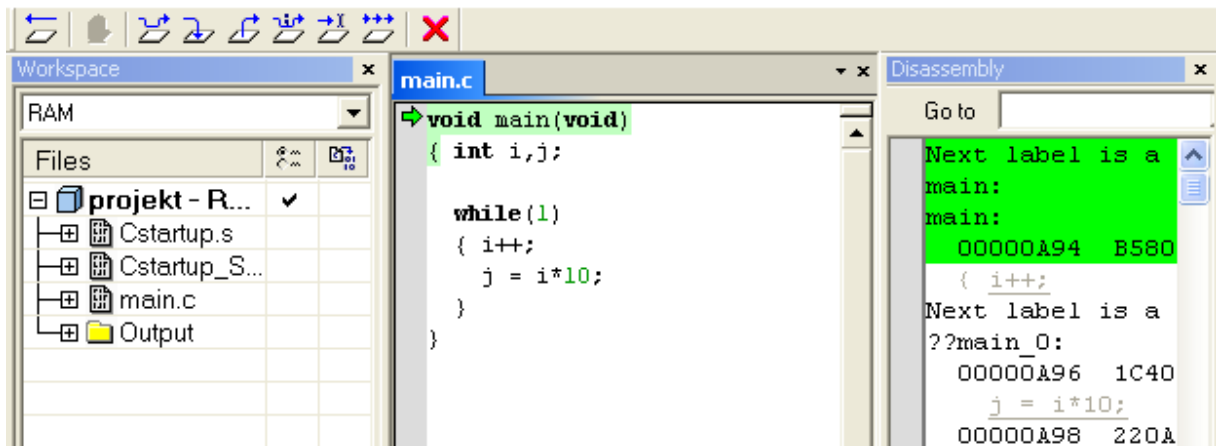
Projekt je možné ladiť v pamäti RAM aj FLASH. V oboch prípadoch je postup ladenia rovnaký. Program skladajúci sa len z prázdnej nekonečnej slučky sa nedá krokovať. Preto upravíme program tak, aby obsahoval dve premenné, i a j , ktoré budú v nekonečnej slučke modifikované. Premenná i bude inkrementovaná a premenná j bude desaťnásobok premennej i :












Počas kompilácie uvedeného príkladu vyhlási kompilátor varovanie, že premenná j bola modifikovaná, ale nebola nikde použitá. Toto varovanie môžeme ignorovať. Varovania sú v programe označené značkou ⚠.

Samotné ladenie projektu spustíme kombináciou tlačidiel Ctrl+D, výberom z menu „Project→Download and Debug“ alebo pomocou ikony . Nasleduje prenos programu do príslušnej pamäte mikrokontroléra. Ak ladíme program v pamäti RAM, prenos je pomerne rýchly. Pri ladení v pamäti FLASH musíme počkať, kým sa program zapíše (napáli) do pamäte.

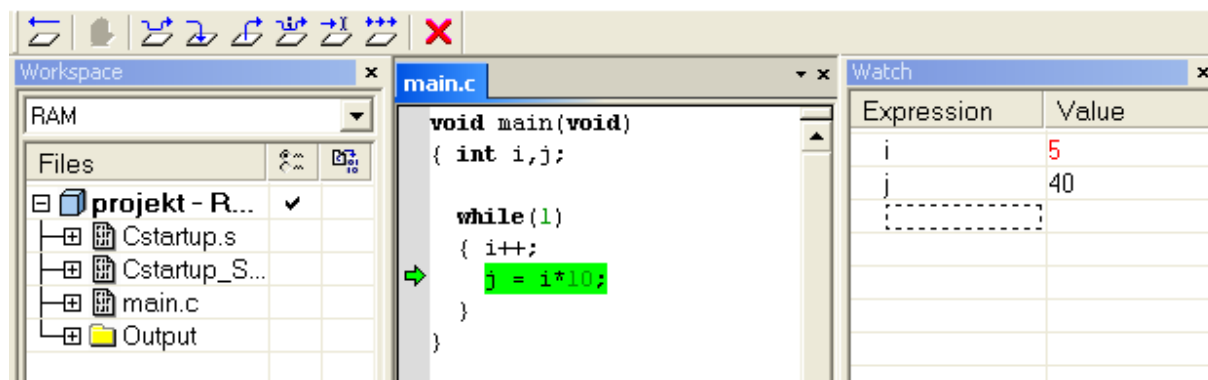
Ak bol prenos programu úspešný, zobrazia sa ikony slúžiace na ladenie a pribudne okno, v ktorom je zobrazený náš program preložený do assembleru. Príkaz, ktorý má byť vykonaný, je vyfarbený zelenou farbou a označený zelenou šípkou.



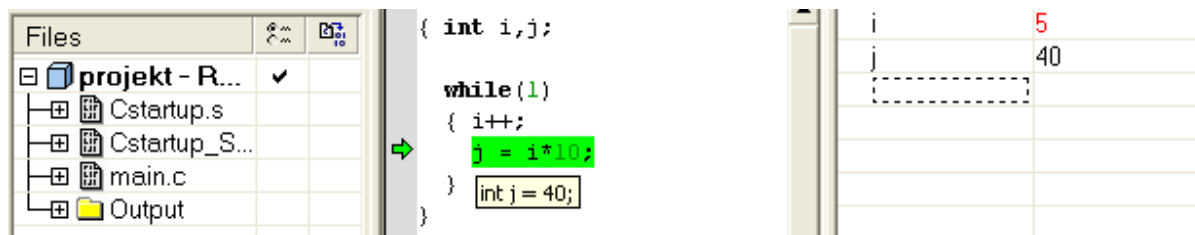
Význam jednotlivých ikon slúžiacich na ladenie programu:

Ikona	Význam	Alternatíva
	Reštart ladeného programu.	Debug→Reset
	Zastavenie vykonávaného programu. Túto ikonu je možné stlačiť, len ak je program vykonávaný plnou rýchlosťou.	Debug→Break
	Vykonanie aktuálneho (označeného) príkazu. Ak je tým príkazom podprogram, vykoná sa bez krokovania.	Debug→Step Over F10
	Ak je označeným príkazom volanie podprogramu, pokračuje sa krokovaním podprogramu.	Debug→Step Into F11
	Vykonanie celého práve krokovaného podprogramu.	Debug→Step Out Shift+F11
	Vykonanie programu po najbližší príkaz. Má význam pri krokovaní programu v asembleri.	Debug→Next Statement
	Vykonanie programu až po aktuálnu pozíciu kurzora.	Debug→Run to Cursor
	Spustenie programu plnou rýchlosťou.	Debug→Go F5
	Ukončenie ladenia programu.	Debug→Stop Debugging Ctrl+Shift+D

Počas ladenia programu je možné zobrazovať obsah premenných, registrov a pamäte prostredníctvom výberu niektorej z možností v menu „View“. Najčastejšie sa používa prezeranie obsahu premenných. Výberom položky View→Watch sa v pravej časti hlavného okna aplikácie otvorí nové okno, ktoré sa skladá z dvoch častí: Expression a Value. Do časti Expression používateľ napíše názov premennej a v časti Value sa zobrazí jej aktuálna hodnota.



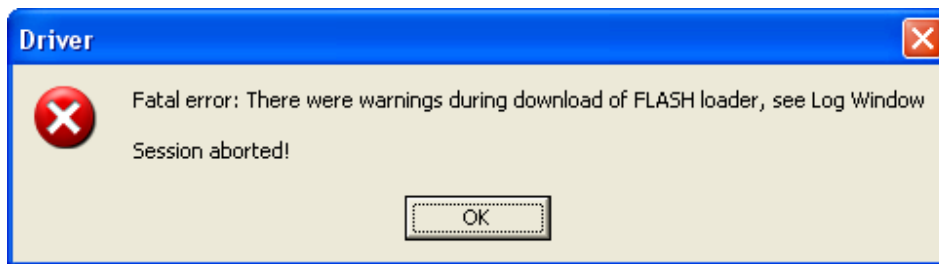
Druhou možnosťou, ako zistiť hodnotu niektorej premennej, je prostredníctvom kurzora myši. Ak nastavíte kurzor myši na niektorú premennú, po chvíli sa objaví okno s jej hodnotou:



Ak má byť v okne Watch veľa zobrazovaných hodnôt (napr. sa majú zobrazovať všetky položky väčšieho poľa), krokovanie programu sa značne spomalí. Uvedené platí v prípade použitia modulu na podporu programovania (Wiggler), ktorý sa pripája k počítaču prostredníctvom paralelného portu. Ak je použitý niektorý z USB JTAG adaptérov, rýchlosť krokovania nie je výpisom obsahu premenných výrazne ovplyvnená.

Problém pri ladení v pamäti FLASH

Počas spúšťania ladenia programu v pamäti FLASH sa môže objaviť nasledovné chybové hlásenie:



V spodnej časti hlavného okna sa zobrazí hlásenie:

*Fatal error: There were warnings during download of FLASH loader, see Log Window
Session aborted!*

Failed to load flash loader: C:\Program Files\IAR Systems\Embedded Workbench 5.4 Kickstart\arm\config\flashloader\Atmel\FlashAT91SAM7Sx.out

Uvedený problém nenastane, ak uskutočníte nasledovné kroky:

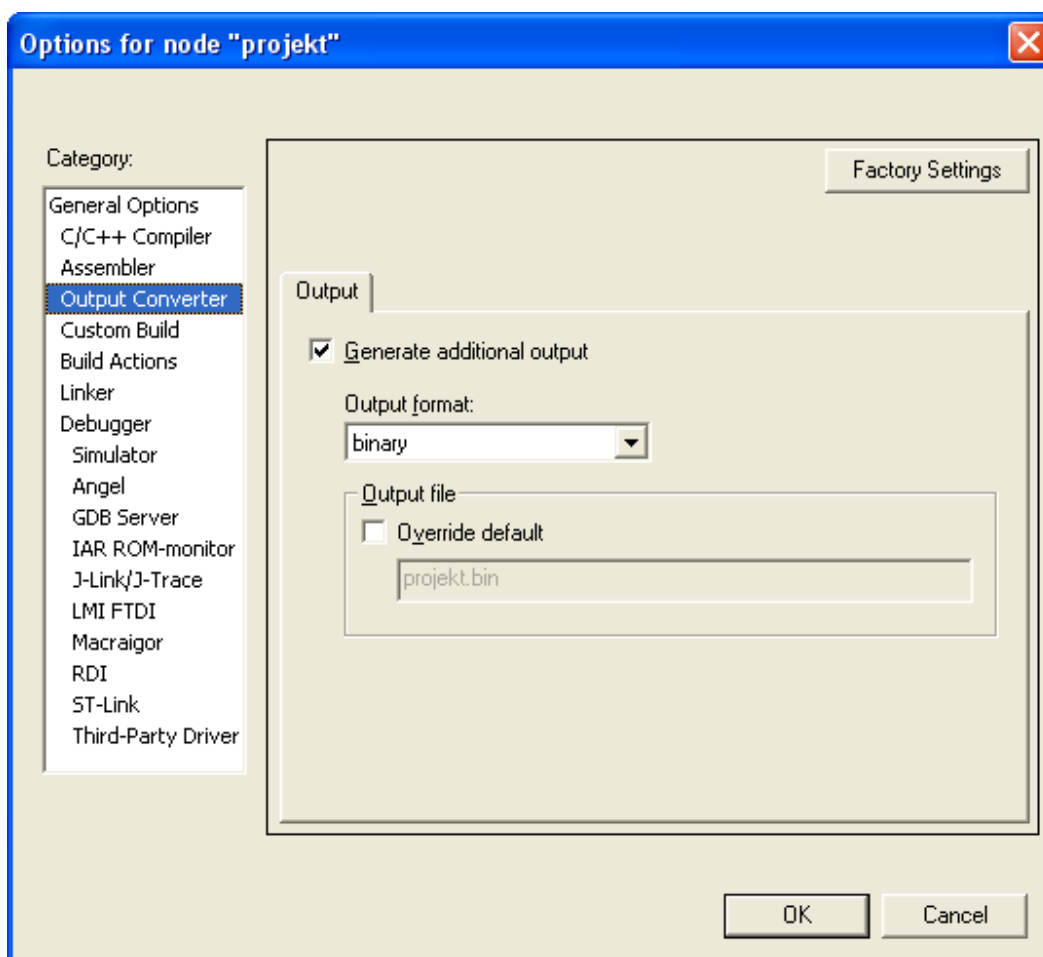
1. Nastavíte konfiguráciu RAM.
2. Spustíte ladenie programu (napr. klávesovou skratkou Ctrl+D).
3. Ukončíte ladenie programu (napr. klávesovou skratkou Ctrl+Shift+D).
4. Nastavíte konfiguráciu FLASH.
5. Spustíte ladenie programu (napr. klávesovou skratkou Ctrl+D).

Postup je potrebné opakovať prakticky pred každým ladením v pamäti FLASH. Nie je možné spustiť ladenie vo FLASH pamäti dva razy po sebe bez chyby. Uvedený problém zrejme súvisí s tým, že vo verzii 5.30 neexistuje súbor „FlashAT91SAM7Sx.out“, ako je spomenuté aj v kapitole „Nastavenie konfigurácie FLASH“ v príručke „IAR – Projekt a nastavenie.pdf“. Vo verzii 5.11 uvedené problémy nenastávali a bolo možné bez problémov ladiť programy vo FLASH pamäti. Je možné, že v ďalšej verzii prostredia IAR bude uvedená chyba odstránená.

Alternatíva zápisu výslednej aplikácie

Vytvorená aplikácia sa v prostredí IAR zapisuje do FLASH pamäte po spustení ladenia programu. Existujú aj iné spôsoby zapísania aplikácie a to pomocou programu OpenOCD alebo H-Flasher. Použitie obidvoch programov je bližšie popísané v príručke „WinARM - Projekt a kompilacia.pdf“. Aby sme mohli niektorý z uvedených programov použiť, je potrebné mať nami vytvorenú aplikáciu vo vhodnom formáte, s ktorým dokážu uvedené programy pracovať. Týmto formátom je binárny formát. Aplikácie vytvorené v prostredí IAR sú uložené vo formáte ELF s príponou *.out* (napr. *projekt.out*). Výsledný súbor je uložený v podadresári *\FLASH\Exe* hlavného adresára projektu. Aby kompilátor vytváral aj binárny súbor, je potrebné uskutočniť zmeny v nastaveniach projektu.

V prvom kroku je potrebné otvoriť okno s nastaveniami. V okne „Workspace“ vyberieme konfiguráciu FLASH, pravým tlačidlom myši klikneme na projekt a zvolíme možnosť „Options“. V ľavej časti okna zvolíme možnosť „Output Converter“, zaškrtneme políčko „Generate additional output“, vyberieme formát výstupu „binary“ a potvrdíme stlačením tlačidla „OK“.



Binárny súbor sa vytvorí po spustení kompilácie projektu. Ak projekt ešte nebol skompilovaný, stačí spustiť kompiláciu napríklad pomocou klávesovej skratky Ctrl+F7. Ak už bol projekt v minulosti skompilovaný, je potrebné vynútiť opätovnú kompiláciu výberom z menu „Projekt→Rebuild All“.