

# Komercyjne środowiska sprzętowo-programowe dla programistów mikrokontrolerów ARM Cortex-M

Na rynku dostępne są różnorodne narzędzia dla programistów przygotowujących aplikacje na mikrokontrolerowy ARM, z których najczęściej są stosowane zintegrowane środowiska programistyczne składające się z IDE, kompilatorów, debuggerów, w niektórych przypadkach także programowych symulatorów oraz narzędzi pomocniczych. Od takich właśnie pakietów zaczniemy nasz przegląd.

## Keil/ARM MDK-ARM z IDE mVision i DS-5

Nadal najpopularniejszym pakietem narzędziowym wśród programistów piszących aplikacje dla mikrokontrolerów wyposażonych w rdzenie ARM w naszym kraju jest MDK-ARM firmy ARM (dawniej Keil). Napisałem „nadal”, bowiem producent tego narzędzia nie rozpuszcza swoich fanów, którzy są zmuszeni do kupowania stosunkowo drogich wersji komercyjnych tego narzędzia, producent – niestety – nie oferuje rozwiązań „budżetowych”, co jest to usprawiedliwione wysoką jakością tego narzędzia. W pewnym stopniu sytuację ratują dedykowane wersje oprogramowania – dzięki współpracy z producentami mikrokontrolerów – udostępniane bezpłatnie, ale ograniczone do wybranej rodziny mikrokontrolerów. Dla prostych aplikacji alternatywą jest także dostępna bezpłatnie wersja Lite, która umożliwia kompilowanie programów o objętości kodu wynikowego do 32 kB.

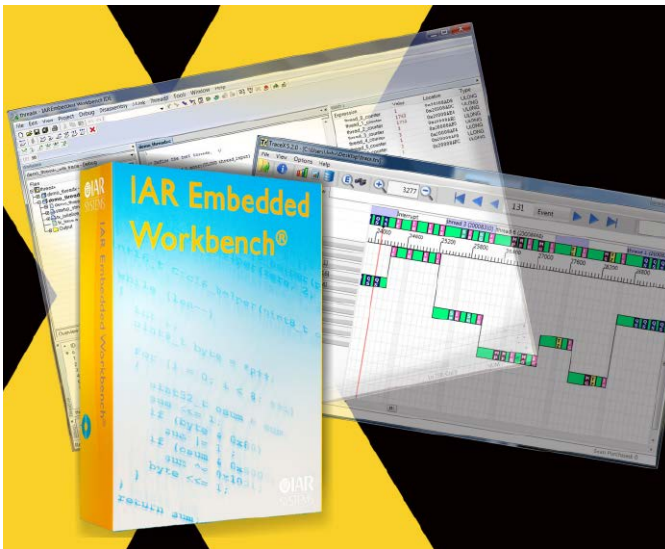
Pakiet ARM-MDK jest kompletnym systemem integrującym w jednym, łatwym w obsłudze środowisku wszystkie narzędzia

*Komercyjne środowiska programistyczne, pomimo naporu rozwiązań bezpłatnych, ciągle dobrze się trzymają na rynku. Są one coraz lepiej wyposażone i właśnie na zaawansowanych możliwościach funkcjonalnych, niedostępnych w narzędziach bezpłatnych, budują i utrwalają swoją pozycję na rynku.*

*W artykule przedstawiamy trzy najpopularniejsze w Polsce komercyjne pakiety programistyczne oraz współpracujące z nimi narzędzia sprzętowe, niezbędne w każdym poważnym, mikrokontrolerowym laboratorium projektowym.*

programistyczne (**rysunek 1**): kompilatory języków C/C++ oraz assemblera, linker, programowy symulator, debugger oraz „opakowujące” te narzędzia w interfejs graficzny IDE (*Integrated Development Environment*) o nazwie mVision (**rysunek 2**) lub DS-5 (bazujący na Eclipse).

Standardowym wyposażeniem wszystkich wersji MDK są biblioteki CMSIS (**tabela 1**), system operacyjny CMSIS-RTOS RTX, wersja najdroższa – MDK-Professional – jest dostarczana także z kompletem bibliotek obsługujących interfejsy komunikacyjne, w tym TCP/IP,



**Keil/ARM MDK-ARM w skrócie:**

- zintegrowane środowisko projektowe
- dla wszystkich mikrokontrolerów z rdzeniami ARM7/9/Cortex-M/Cortex-R/Cortex-A
- dostępny w bezpłatnej wersji ewaluacyjnej
- kompilator C/C++
- zintegrowany system operacyjny
- obsługa wielu interfejsów programująco-debugujących
- zintegrowane biblioteki

**Tabela 1. Zestawienie najważniejszych cech dostępnych wersji pakietów MDK-ARM firmy Keil/ARM**

Cecha	Professional	Plus	Cortex-M	Lite
IDE with Editor, Pack Installer	+	+	+	+
Debugger	+	+	+	32 kB
Fixed Virtual Platforms Simulation Models	+	-	-	-
Compiler, Linker, Libraries, and Utilities	+	+	+	32 kB
ARM Cortex-M	+	+	+	+
ARM SecurCore®	+	+	-	+
ARM7, ARM9, ARM Cortex-R4	+	+	-	+
DS-MDK (systemy heterogeniczne)	+	-	-	-
ARM v8-M	+	-	-	-
CMSIS-RTOS RTX	+	+	+	+
Middleware IPv4 Network, USB Device, File System, Graphics	+	+	-	-
Middleware IPv6 Network, USB Host, mbed TLS	+	-	-	-
ULINK Debug Adapters	+	+	+	+
Trace Streaming Instruction Trace, Embedded Trace Buffer, Micro Trace Buffer	+	+	+	+
Advanced Analysis Tools Code Coverage, Performance Analyzer, Execution Profiler, Logic Analyzer	+	-	+	+

USB Device i Host, interfejs CAN, system plików FAT oraz interfejsy graficzne. Nowe wersje pakietu MDK-Professional będą obsługiwać także architektury ARMv8-M oraz dwurdzeniowe Cortex-A/Cortex-M.

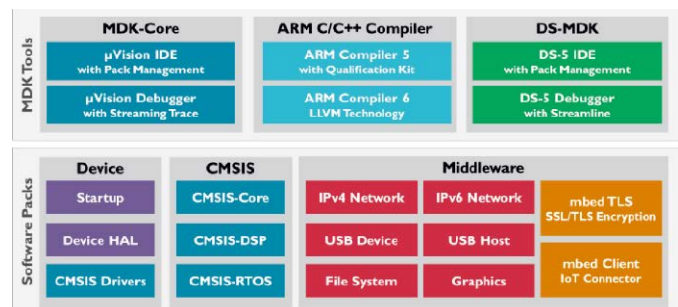
Pakiet MDK-ARM jest przystosowany do współpracy z większością popularnych interfejsów programująco-debugujących (m.in. J-Link firmy Segger, ST-Link firmy STMicroelectronics, OpenSDA, Nuvoton Nu-Link), jest także przystosowany do łatwego upgrade'owania (poprzez dodanie odpowiednich bibliotek DLL) obsługi nowych interfejsów. Producent pakietu oferuje jego użytkownikom własne interfejsy – serię urządzeń o nazwie ULink – których rodzina składa się obecnie z trzech modeli:

- ULINK2 – model podstawowy, obsługujący mikrokontrolery ARM7/ARM9, Cortex-M, starsze mikrokontrolery z rdzeniem 8051: XC800/mPSD/LPC950 oraz 16-bitowe XC166. Obsługuje interfejsy SWO z prędkością do 1 Mb/s, nie jest obsługiwany przez IDE DS-5,
- ULINKproD (fotografia 3) – interfejs przystosowany do współpracy z interfejsami zasilanymi napięciem od 1,2 V, obsługą SWO do 100 Mb/s, obsługiwany przez obydwa IDE,
- ULINKpro – interfejs przystosowany do współpracy z interfejsami zasilanymi napięciem od 1,2 V, obsługą SWO do 100 Mb/s, ETM do 800 Mb/s, obsługiwany przez obydwa IDE.

Pakiet MDK-ARM jest oferowany w czterech podstawowych wersjach, których najważniejsze cechy zestawiono w tabeli 1.

**IAR Embedded Workbench**

Dużą elastycznością oraz zaawansowanymi możliwościami funkcjonalnymi charakteryzuje się także pakiet Embedded Workbench firmy IAR. Jest on w naszym kraju nieco mniej popularny niż MDK, ale oferuje nie mniejsze możliwości, zwłaszcza podczas projektowania i optymalizacji systemów energooszczędnych. Firma IAR wyposaża pakiet Workbench w pluginy umożliwiające współpracę ze wszystkimi liczącymi się systemami na rynku (m.in. AVIX, CMX-RTX, e-Force μC3/Compact, FreeRTOS, OpenRTOS, SafeRTOS, Freescale MQX, Micrium μC/OS-III, Quadros RTXC, Segger embOS). IAR Workbench używa własnego IDE (rysunek 4), o nieco innym sposobie obsługi niż wcześniej prezentowany MDK, ale rozwiązanie proponowane przez firmę IAR ma także wielu zwolenników, dostępne są także narzędzia integrujące środowisko Workbench pod „skrzydłami” Eclipse.



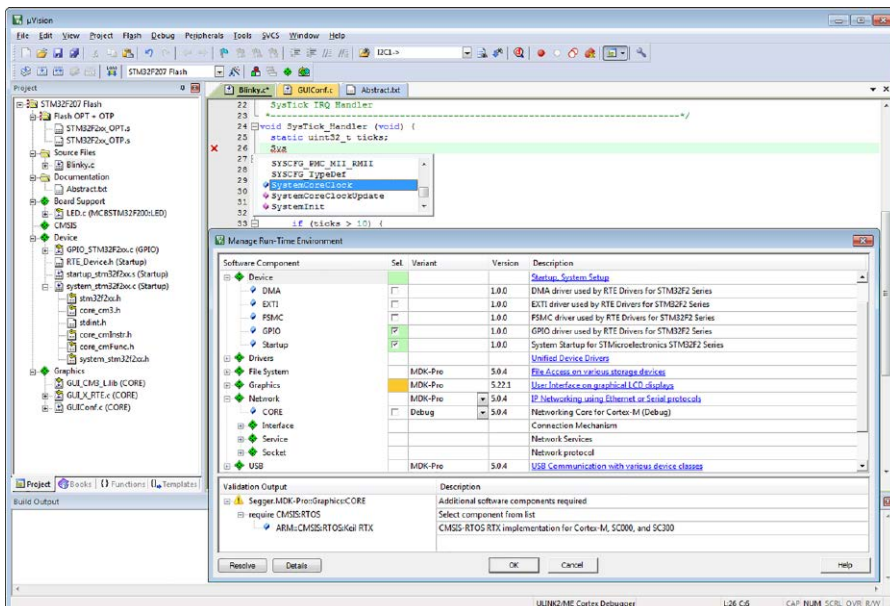
**Rysunek 1. Narzędzia programistyczne zawarte w MDK-ARM**

**IAR Embedded Workbench w skrócie:**

- zintegrowane środowisko projektowe
- dla wszystkich mikrokontrolerów z rdzeniami ARM7/9/ARM11/Cortex-M/Cortex-R oraz mikroprocesorów z rdzeniami Cortex-A
- dostępny w bezpłatnej wersji kick-start z ograniczeniami funkcjonalnymi (m.in. kompilacja do 32 kB lub 16 kB w przypadku Cortex-M0/M0+/M1, brak obsługi MISRA C) lub 30-dniowej wersji pełnej
- kompilator C/C++
- możliwość współpracy z systemami operacyjnymi dostarczonymi przez zewnętrznych producentów
- obsługa wielu interfejsów programująco-debugujących
- zintegrowane biblioteki

**Atollic TrueSTUDIO w skrócie:**

- zintegrowane środowisko projektowe
- dla mikrokontrolerów z rdzeniami ARM7/ARM9, Cortex-M, Cortex-A5
- dostępny w bezpłatnej wersji TrueSTUDIO, wersji komercyjnej TrueSTUDIO Pro oraz w 30-dniowej wersji ewaluacyjnej TrueSTUDIO Pro bez ograniczeń funkcjonalnych
- kompilator C/C++
- możliwość współpracy z systemami operacyjnymi dostarczonymi przez zewnętrznych producentów
- współpraca interfejsów programująco-debugujących: Segger J-Link, STMicroelectronics ST-LINK oraz ST-LINK/V2, Atmel SAM-ICE oraz *open-source* zgodny z *gdbserver*
- zintegrowane biblioteki



**Rysunek 2. Interfejs graficzny mVision**

Pakiet Embedded Workbench jest przystosowany do współpracy z wieloma standardowymi interfejsami do debugowania i programowania (jak na przykład rodzina interfejsów J-Link firmy Segger), producent pakietu oferuje także kilka typów własnych interfejsów, jak na przykład I-jet (fotografia 5, taktowanie JTAG do 48 MHz) lub I-jet Trace (fotografia 6, taktowanie JTAG do 100 MHz). Mocną stroną pakietu Embedded Workbench jest zaimplementowana obsługa najpopularniejszych interfejsów komercyjnych (m.in. Segger J-Link, ST-Link, XDS100) oraz *open-source* jak na przykład Stellaris FTDI/ICDI, Macraigor OCDemon, OpenSDA itp.

**Atollic TrueSTUDIO**

Firma Atollic opracowując pakiet TrueSTUDIO wybrała inną drogę niż firmy, których narzędzia przedstawiliśmy dotychczas: zamiast budować własne oprogramowanie od podstaw, jej inżynierowie wykorzystali oprogramowanie *open-source* (IDE Eclipse, kompilator ARM-GCC, serwer debugera *gdbserver*), które charakteryzuje się – w wersjach natywnych – trudną instalacją, koniecznością wykonywania wielu ręcznych modyfikacji i poprawek, a także monitorowania zmian wprowadzanych przez zespoły *freelancerów* rozwijających oprogramowanie.

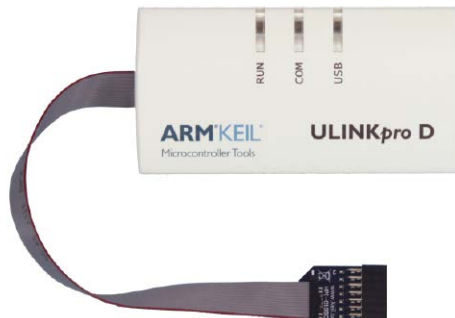
Pakiet TrueSTUDIO (rysunek 7) – pomimo tego, że składa się z elementów *open-source'owych* – jest wygodny w instalacji i nie wymaga podczas eksploatacji przesadnej troski ze strony użytkowników. Nie dzieje się to oczywiście za darmo: wersja TrueSTUDIO Pro wymaga zakupu licencji, ale jej koszt jest konkurencyjny w stosunku do innych narzędzi dostępnych na rynku. Pakiet TrueSTUDIO Pro jest dostępny w 30-dniowej wersji trial.

Wersja komercyjna TrueSTUDIO Pro jest wyposażona w zaawansowane narzędzia profilujące, debugujące i optymalizujące projekty, które odróżniają je od bezpłatnej wersji TrueSTUDIO, wśród nich m.in.: CPU Fault Analyzer, Statistic Profiler, Exception and Interrupt Tracing, Instruction Tracing, Real-Time Data Access Tracing, a także możliwość debugowania pracy systemów wielordzeniowych.

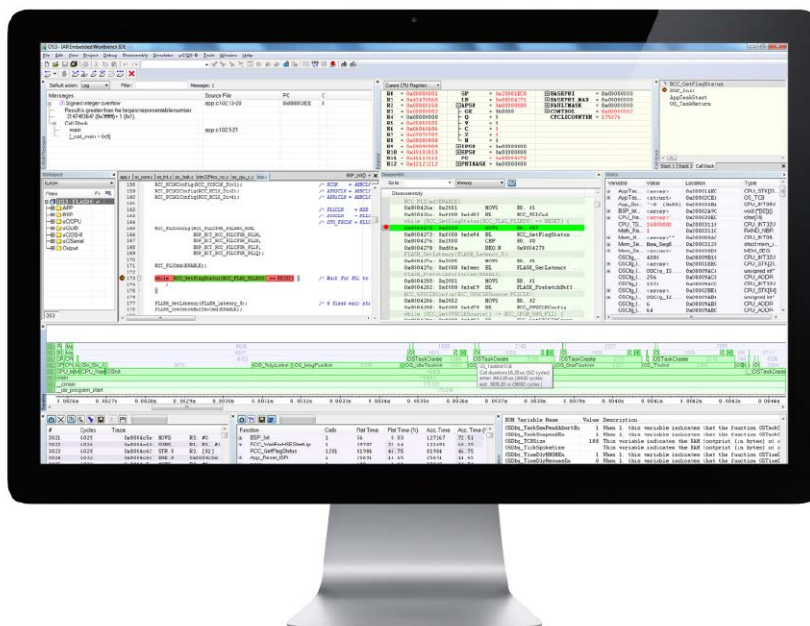
Pakiet TrueSTUDIO nie został przez producenta wyposażony w zintegrowany system operacyjny, ale może współpracować z kilkoma systemami komercyjnymi (m.in.

HCC-Embedded eTaskSync, Micrium uC/OS, CMX-RTX/CMX-TINY+, Express Logic ThreadX, FreeRTOS, Segger embOS, HighIntegrity-Systems/Wittenstein OpenRTOS/SafeRTOS, Code Time Technologies Abassi, Quadros RTX).

Opracowując komercyjną wersję TrueSTUDIO producent założył, że obsługiwane będą przede wszystkim interfejsy programująco-debugujące zgodne z *gdbserver*, ale dość szybko zintegrowano w nim także



**Fotografia 3. Programator/debugger ULINKpro D**



**Rysunek 4. Interfejs graficzny IAR Workbench**



**Tabela 2. Zestawienie najważniejszych cech dostępnych wersji pakietów Embedded Workbench firmy IAR**

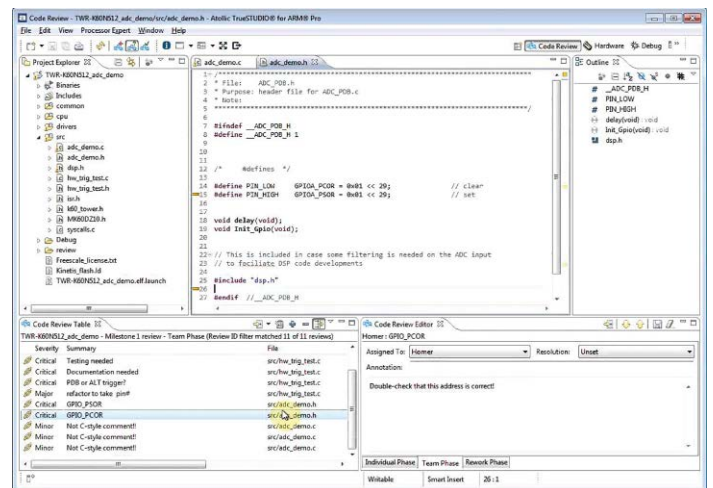
Cecha	Functional Safety	Standard	Cortex-M	Cortex-M0	Limited	Baseline
Obsługiwane rdzenie	ARM Cortex-M0, M0+, M1, M3, M4, M7, R4, R5, R7, A5, A7, A8, A9, A15 and ARM11, ARM9, ARM7, SecurCore	ARM Cortex-M0, M0+, M1, M3, M4, M7, M23, M33, R4, R5, R7, A5, A7, A8, A9, A15 and ARM11, ARM9, ARM7, SecurCore	ARM Cortex-M0, M0+, M1, M3, M4/ M4F, M7, M23, M33	ARM Cortex-M0, M0+, M1, M23	ARM Cortex-M0, M0+, M1, M3, M4, M7, M23, M33, R4, R5, R7, A5, A7, A8, A9, A15 and ARM11, ARM9, ARM7, SecurCore	ARM Cortex-M0, M0+, M1, M3, M4, M7, M23, M33, R4, R5, R7, A5, A7, A8, A9, A15 and ARM11, ARM9, ARM7, SecurCore
Edytor	+	+	+	+	+	+
Project Manager	+	+	+	+	+	+
IAR C/C++ Compiler for ARM	+	+	+	+	+	256 kB
IAR Assembler for ARM	+	+	+	+	+	+
Linker and librarian	+	+	+	+	+	+
MISRA C checker	+	+	+	+	-	-
C-SPY Debugger	+	+	+	+	-	+
Run-time libraries incl source code	+	+	+	+	+	+
Power debugging	+	+	+	-	-	-
Hardware debugging support	+	+	+	+	-	+
RTOS plugins	+	+	+	+	-	+
Support and Update Agreement (SUA)	+	+	+	+	+	+
Static code analysis	Opcja	Opcja	Opcja	Opcja	-	-
Runtime analysis	Opcja	Opcja	Opcja	Opcja	-	-
Certified for Functional Safety	+	-	-	-	-	-



Fotografia 5. Programator/debugger I-jet



Fotografia 6. Programator/debugger I-jet Trace



Rysunek 7. Interfejs graficzny TRUE Studio

obsługę interfejsów J-Link firmy Segger, Atmel SAM-ICE oraz interfejsów zgodnych z ST-LINK oraz ST-LINK/V2 firmy STMicroelectronics.

**Podsumowanie**

W tej części artykułu przedstawiliśmy trzy najpopularniejsze w naszym kraju, komercyjne pakiety narzędziowe dla mikrokontrolerów z rdzeniami firmy ARM. Oferowane przez niektórych producentów mikrokontrolerów pakiety bezpłatne oraz dodatkowe narzędzia programowe dla programistów przedstawimy w kolejnych wydaniach EP.

Tomasz Starak, EP