
Wstęp

W trakcie prowadzonych w Polsce i poza jej granicami szkoleń autor wielokrotnie odpowiadał na pytania o praktyczne aspekty pracy z systemem Linux, sterownikami, aplikacjami, narzędziami wspomagającymi budowanie oraz programami i bibliotekami Open-Source. Ta książka zawiera odpowiedzi na najważniejsze pytania a także opisy dobrych praktyk pracy z Linuksem, Yocto, Buildrootem i wieloma innymi narzędziami. Powstała na podstawie doświadczeń autora w pracy nad projektami systemów wbudowanych.

Linux ma ogromny wpływ na obecny kształt rynku urządzeń embedded. Już od kilku lat osoby odpowiedzialne za tworzenie nowych produktów i rozwiązań nie zadają sobie pytania „Czy użyć Linuksa?”, ale „W jaki sposób mam użyć Linuksa w moim projekcie?”.

W książce posługujemy się przykładem rodziny układów i.MX6 firmy Freescale. Są to nowoczesne procesory zbudowane w oparciu na architekturze ARM, wykorzystujące od 1 do 4 rdzeni Cortex-A9. Ich mocną stroną są przede wszystkim funkcje multimedialne.

Praca z książką

Rozdział 1 zawiera opis zagadnień związanych z coraz powszechniej stosowanym systemem budowania Yocto. Ze względu na skomplikowaną budowę, budzi on spore kontrowersje oraz obawy użytkowników. Na podstawie praktycznych przykładów, czytelnik jest wprowadzany w mechanizmy konfiguracji narzędzia, sposoby dodawania własnych programów i zagadnienia związane z optymalizowaniem pracy.

W rozdziale 2 omawiana jest konfiguracja środowiska Eclipse do pracy z kodem źródłowym jądra i dodatkowych modułów. Także sposoby zdalnego uruchamiania i debugowania aplikacji oraz jądra Linux. Eclipse stało się de facto standardem. Używana przez nas wersja otwarcie źródłowa jest bazą dla wielu komercyjnych rozwiązań, z którymi inżynierowie i programiści stykają się w codziennej pracy.

Opisany w rozdziale 3 Buildroot jest bardzo elastycznym narzędziem służącym do budowania toolchainów i dystrybucji systemu Linux dla urządzeń wbudowanych. Zdaniem autora narzędzie to jest komplementarne z Yocto i rozwiązuje problemy pojawiające się w zupełnie innej kategorii projektów.

Rozdział 4 prezentuje rozwiązania firmy Timesys dla Linuksa Embedded. Webowe i działające lokalnie wersje narzędzia `factory` – służącego do budowania obrazu systemu wraz z SDK oraz `Timestorm` – IDE oparte na Eclipse. Są to produkty komercyjne (można uzyskać płatne wsparcie), ale niezamknięte (dostępne są wersje bez opłat).

W rozdziale 5 znajduje się dużo informacji na temat konfiguracji ekranów i wyświetlania obrazu w systemie Linux. To znaczy działania i sposobów podłączania różnego rodzaju ekranów, monitorów oraz telewizorów. Także praktyczne informacje o ich konfiguracji. Ponadto czytelnik znajdzie tu opis funkcji graficznych układów `i.MX6`.

Rozdział 6 dotyczy przetwarzania multimediiów. Przede wszystkim mowa w nim o frameworku `GStreamer` i zastosowaniu jego komponentów do obróbki obrazu z użyciem sprzętowej akceleracji implementowanej w `i.MX6`. Przykłady obejmują odtwarzanie i nagrywanie filmów, przetwarzanie obrazu z kamer, naziemnej telewizji cyfrowej i innych źródeł. W tym rozdziale opisano także bibliotekę `OpenCV` i jej zastosowania w urządzeniach wbudowanych (przede wszystkim rozpoznawanie ruchu i inne tego typu algorytmy).

Najbardziej obszerny rozdział 7 poświęcony jest zagadnieniom niskopoziomym. Omawiane są kwestie związane z bootloaderem (m.in. `U-BOOT`) i sposobem uruchamiania systemu. Dobór wersji jądra i konfiguracja urządzeń platformy: pliki `.c` (stary sposób) oraz `Device Tree`. Funkcje związane z usypianiem systemu i oszczędzaniem energii. Na praktycznych przykładach przedstawiono pracę z systemem Linux i sprzętem takie, jak: podłączanie urządzeń `I2C` i `1-wire`, magistrala `CAN` oraz protokół `Bluetooth`. Także wewnętrzną budowę systemu: skrypty startowe dla własnych aplikacji, `systemd`, reagowanie na zdarzenia związane ze sprzętem (`udev`). Wreszcie sposoby profilowania i optymalizacji czasu startu i usypiania/budzenia systemu.

Omawiane zagadnienia przenikają się wzajemnie i często wielokrotnie powracamy do różnych ich aspektów w kolejnych rozdziałach. Celem autora jest to, aby po lekturze książki czytelnik mógł podjąć pracę z własnym projektem urządzenia wbudowanego na bazie układu `i.MX6`.

Zastosowane konwencje

- Książka zawiera ponad 160 ilustracji, zdjęć i schematów ilustrujących omawiane zagadnienia.
- Świadomie zrezygnowałem ze sztucznego tłumaczenia na język polski niektórych pojęć i terminów. W tekście posługuję się ich oryginalnym brzmieniem stosowanym w dokumentacji oraz w wiadomościach na listach dyskusyjnych. Czytelnik znajdzie więc odwołania do Device Tree oraz cache'u.
- Książka zawiera fragmenty kodu źródłowego, plików konfiguracyjnych i skryptów. Zostały one wydrukowane czcionką o stałej szerokości. Jeżeli cytowane są fragmenty istniejących plików, umieszczono w nich numery linii. W tekście staram się skupiać na meritem – fragmentach rozwiązujących lub ilustrujących konkretny problem. Kompletne programy i skrypty czytelnik znajdzie w dodatkowych materiałach do książki.
- Komendy i wyniki ich działania również wydrukowano czcionką o stałej szerokości. Komendy są poprzedzane znakiem \$. Służy to oddzieleniu ich od komunikatów. Przykład:

```
# To jest kometarz -- po nim znajduje się przykładowa komenda,  
# a w~kolejnej linii skutek jej wykonania.  
$ komenda --opcje  
bash: komenda: command not found
```

Ta książka ma o kilka tysięcy stron więcej...

...dużo więcej, niż wydaje się na pierwszy rzut oka.

Współczesne procesory są bardzo skomplikowanymi układami. Zawierają nowoczesny rdzeń o rozbudowanych możliwościach, a także wiele peryferiów. Powoduje to, że dokumentacja wszystkich funkcji układu osiąga objętość kilku tysięcy stron. Nie będziemy tu przedrukowywać ani schematów, ani kompletnej listy rejestrów i ich opisów. Najbardziej aktualna ich wersja jest dostępna do pobrania ze strony producenta.

W książce można znaleźć odwołania do następujących dokumentów referencyjnych Freescale: [RM], [DLRM], [Sec]. Kompletny wykaz literatury znajduje się na stronie 371.

Na przykład: [RM, 47.5.35] oznacza, że dodatkowe informacje znajdują się w rozdziale 47.5.35 dokumentu: *i.MX 6Solo/6DualLite Applications Processor Reference Manual*. W tym przypadku jest to opis rejestru, który odwzorowuje zawartość jednorazowo programowalnej pamięci – *Value of OTP Bank4 Word2 (MAC Address) (OCOTP_MAC0)*. Jest przeznaczony do przechowywania całości lub fragmentu adresu ethernetowego karty sieciowej.

i.MX6 jest jednym z lepiej udokumentowanych układów dostępnych na rynku. Dokument [RM] na 5827 stron i zawiera, oprócz opisów rejestrów, także informacje

o ich wzajemnych relacjach i wysokopoziomowych funkcjach. Najważniejsza dokumentacja jest dostępna on-line, bez potrzeby podpisywania umów o poufności (NDA).

Oprócz specyfikacji procesora, tekst odwołuje się także do podręczników i dokumentacji poszczególnych omawianych narzędzi (na przykład systemów budowania).

Linux w systemach embedded

Ta książka stanowi uzupełnienie mojego poprzedniego tytułu – wydanej w 2011 r. [LWSE]. Wielokrotnie w tekście występują odwołania do tej pozycji, ale nie ma tu skopiowanych akapitów. Rozdziały dotyczące podobnych zagadnień zostały napisane od nowa.

W czasie, który upłynął między wydaniem obu pozycji, systemy wbudowane, jak i Linux, ewoluowały. Znacznie wzrosła wydajność dostępnych rozwiązań i liczba integrowanych funkcji. Środek ciężkości przesunął się z zagadnień związanych z budowaniem zoptymalizowanych pod konkretną platformę toolchainów i zminimalizowanych wersji narzędzi na jak najszybsze dostarczanie działających rozwiązań (zgodnie z metodologią agile).

Nowe podejście wymaga nowych narzędzi (Yocto, Buildroot, Eclipse). Pozwalają one skupić się nad clue rozwiązania (aplikacją) i nie rozdrabniać się nad uruchamianiem podstawowych funkcji systemu.

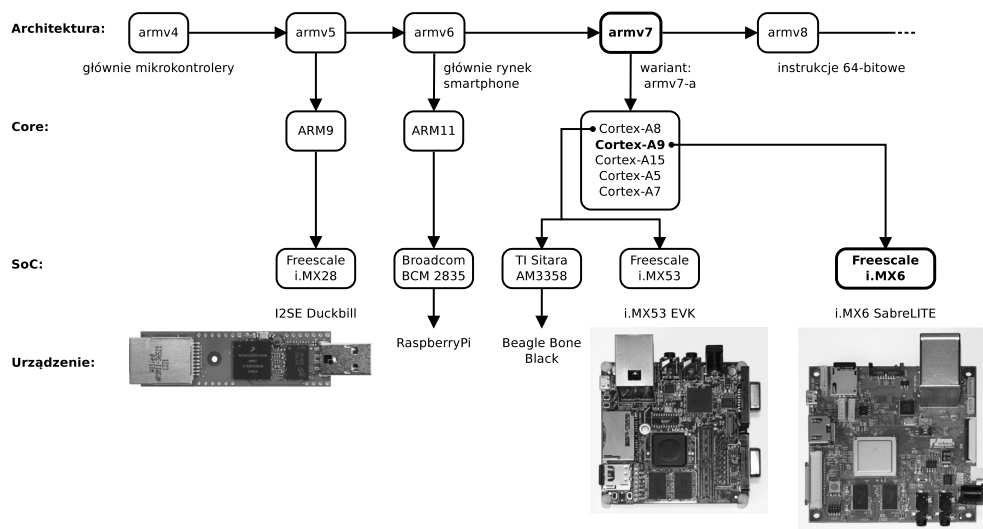
Dostęp do kodów źródłowych i środowisko pracy

Podczas uruchamiania programów, kompilowania projektów otwarto źródłowych i przeprowadzania testów autor używał komputera PC działającego pod kontrolą 64-bitowej wersji dystrybucji Ubunutu/Xubuntu 12.04 LTS oraz 14.04 LTS. Te wersje są wspierane przez 5 lat od daty wydania. Zapewniają solidną bazę do dalszej pracy. Zawierają bardzo dużo gotowych do zainstalowania pakietów. Pozwalają skupić się na pracy z systemem wbudowanym.

Dołączone do książki przykłady zadziałają również w innych aktualnych dystrybucjach.

Dodatkowe materiały do książki to przede wszystkim cytowane w tekście: specyfikacja procesora, dokumentacja dotycząca poszczególnych urządzeń oraz podręczniki systemów budowania i narzędzi. Odnośniki do dokumentów źródłowych, pliki konfiguracyjne systemów budowania, a także źródła i poprawki programów i narzędzi, o których mowa w tekście, znajdują się pod adresem:

<http://bis-linux.com/lwsi>



Rys. 1. i.MX6 w kontekście innych procesorów architektury ARM

Rodzina układów Freescale i.MX6

i.MX6 to nowoczesny układ typu SoC – *System on Chip*. Zawiera rdzeń Cortex-A9 firmy ARM oraz wiele dodatkowych peryferii. Wspierane są: zestaw instrukcji wektorowych: NEON, koprocesor numeryczny – VFPvD16, bezpieczne wykonywane instrukcje – Trustzone, liczne interfejsy wejścia-wyjścia, kontrolery pamięci niulotnych: NAND, MMC, SATA, magistala PCI Express oraz wiele sprzętowych mechanizmów przetwarzania grafiki i multimediów.

Procesory dostępne są w pięciu wariantach (rysunek 2). Cztery z nich używają tego samego układu wyprowadzeń.

i.MX6	Liczba rdzeni	Pamięć Cache L2	Grafika	Kontroler pamięci	Dodatkwe kontrolery
SoloLite	1 @ 1,0 GHz	256 KB	2D	32-bit DDR3 and LPDDR2	EPD
Solo	1 @ 1,0 GHz	256 KB	3D, 1 shader	32-bit DDR3 and LPDDR2	EPD
DualLite	2 @ 1,0 GHz	512 KB	3D, 1 shader	64-bit DDR3, 2-channel 32-bit LPDDR2	EPD
Dual	2 @ 1,2 GHz	1 MB	3D, 4 shaders, 200MT/s	64-bit DDR3, 2-channel 32-bit LPDDR2	SATA-II
Quad	4 @ 1,2 GHz	1 MB	3D, 4 shaders, 200MT/s	64-bit DDR3, 2-channel 32-bit LPDDR2	SATA-II

i.MX 6 Series at a Glance

Red indicates change from column to the left



i.MX6SoloLite

- Single ARM® Cortex™-A9 up to 1.0 GHz
- 256 KB L2 cache, Neon, VFPvD16 Trustzone
- 2D graphics
- 32-bit DDR3 and LPDDR2 at 400 MHz
- Integrated EPD controller



i.MX6Solo

- Single ARM Cortex-A9 up to 1.0 GHz
- 512 KB L2 cache, Neon, VFPvD16 Trustzone
- 3D graphics with one shader
- 2D graphics
- 32-bit DDR3 and LPDDR2 at 400 MHz
- Integrated EPD controller
- HDMIv1.4 controller plus PHY
- LVDS controller plus PHY
- PCIe controller plus PHY
- MLB and FlexCan controllers



i.MX6DualLite

- Dual ARM Cortex-A9 up to 1.0 GHz
- 512 KB L2 cache, Neon, VFPvD16 Trustzone
- 3D graphics with one shader
- 2D graphics
- 64-bit DDR3 and 2-channel 32-bit LPDDR2 at 400 MHz
- Integrated EPD controller
- HDMIv1.4 controller plus PHY
- LVDS controller plus PHY
- PCIe controller plus PHY
- MLB and FlexCan controllers



i.MX6Dual

- Dual ARM Cortex-A9 up to 1.2 GHz
- 1 MB L2 cache, Neon, VFPvD16 Trustzone
- 3D graphics with four shaders
- Two 2D graphics engines
- 64-bit DDR3 and 2-channel 32-bit LPDDR2 at 533 MHz
- Integrated SATA-II
- HDMIv1.4 controller plus PHY
- LVDS controller plus PHY
- PCIe controller plus PHY
- MLB and FlexCan controllers



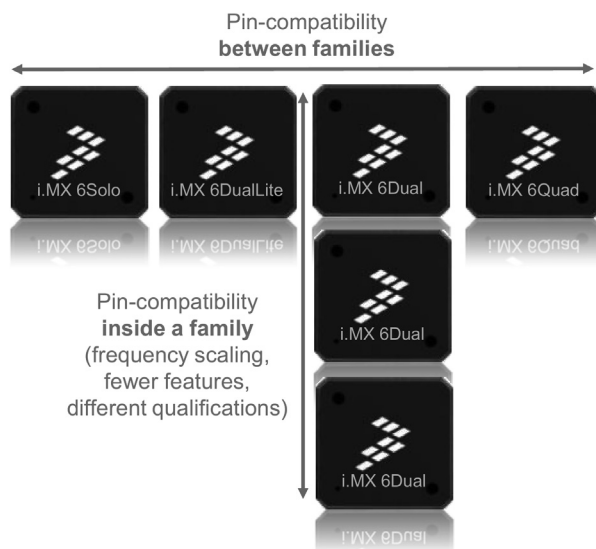
i.MX6Quad

- Quad ARM Cortex-A9 up to 1.2 GHz
- 1 MB L2 cache, Neon, VFPvD16 Trustzone
- 3D graphics with four shaders
- Two 2D graphics engines
- 64-bit DDR3 and 2-channel 32-bit LPDDR2 at 533 MHz
- Integrated SATA-II
- HDMIv1.4 controller plus PHY
- LVDS controller plus PHY
- PCIe controller plus PHY
- MLB and FlexCan controllers



Qual. Tier	Junction Temperature T _j	CPU Fmax	Operational Lifetime	Lifetime at max T _j and CPU Fmax	6Q	6D	6DL	6S	6SL
Consumer	0 to +95°C	1 GHz	5yrs	50% (2.5yrs) T _j = +95°C, 1 GHz			X	X	X
Ext. Consumer	-20°C to +105°C	1 GHz	5yrs	50% (2.5yrs) T _j = +105°C, 1 GHz	X	X	X	X	
Ext. Consumer	-20°C to +105°C	1.2 GHz	5yrs	50% (2.5yrs) T _j = +105°C, 1.2 GHz	X	X			
Ext. Consumer	-40°C to +105°C	1 GHz	5yrs	50% (2.5yrs) T _j = +105°C, 1 GHz					X
Automotive AEC-Q100	-40°C to +125°C	1 GHz	15yrs	10% (1.5yrs) T _j = +125°C, 1 GHz	X	X			
Automotive AEC-Q100	-40°C to +125°C	800 MHz	15yrs	10% (1.5yrs) T _j = +125°C, 800 MHz	X	X	X	X	
Industrial	-40°C to +105°C	800 MHz	10yrs	100% (10yrs) T _j = +105°C, 800 MHz	X	X	X	X	

Rys. 2. Rodzina układów i.MX6 firmy Freescale. Najważniejsze funkcje i kwalifikacja poszczególnych wersji układów do pracy w różnych warunkach. Źródło: materiały producenta (IMX6SRSFS)

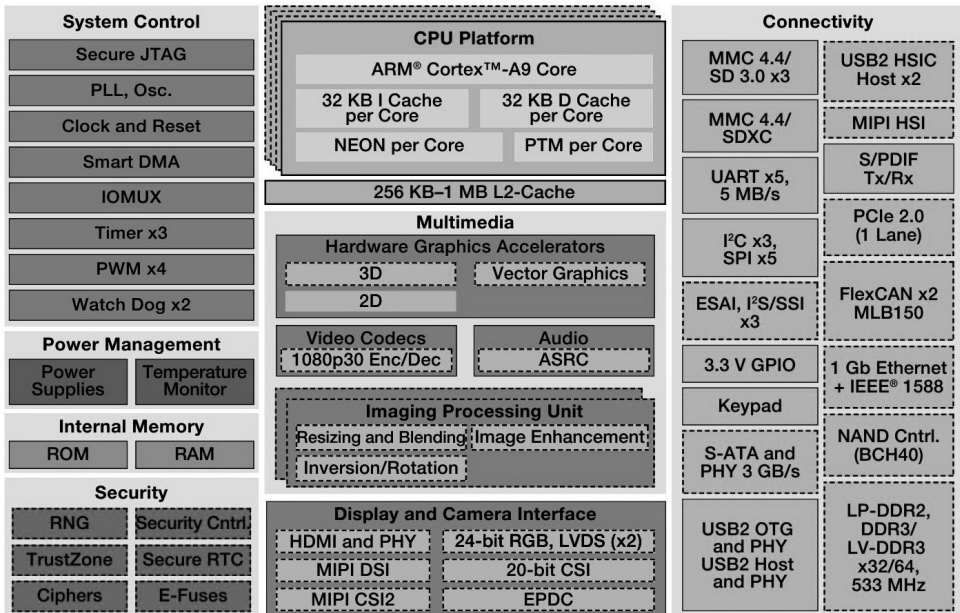


Rys. 3. Układy rodziny i.MX6 są ze sobą kompatybilne pin-do-pinu

- EPD to zintegrowany kontroler ekranów TFT oraz eINK. Znajduje się on w układzie SoloLite przeznaczonym specjalnie dla urządzeń „przenośnych” (*wearable*), zasilanych z baterii, wyposażonych w energooszczędny ekran eINK.
 - SoloLite występuje tylko w wersji Consumer. Pozostałe wersje dostępne są także w wariantach Automotive oraz Industrial. To ostatnie oznacza, że producent dopuszcza stosowanie układu w systemach, które pracują w trybie 24/7. Interpretacja poszczególnych oznaczeń znajduje się na rysunku 2.
 - Z wyjątkiem SoloLite, dla którego zaprojektowano minimalistyczną obudowę, układy wykorzystują ten sam układ wyprowadzeń. Umożliwia to łatwą aktualizację sprzętu do nowych wymagań przez dodanie mocniejszego procesora z większą liczbą rdzeni. Oczywiście kolejne układy zawierają też dodatkowe peryferie, wprowadzają więcej użytecznych sygnałów i różnią się wymaganiami dotyczącymi prądu zasilania. Projektując urządzenie, należy przewidzieć większe zapotrzebowanie na energię i większą ilość generowanego ciepła.
- Wyprowadzenia są kompatybilne pin-do-pinu, zarówno w obrębie rodziny, jak i odpowiedniego wariantu procesora. i.MX6: Solo, DualLite, Dual, Quad mają takie same wyprowadzenia w każdym z wariantów kwalifikacji: Consumer, Industrial i Automotive (rysunek 3).
- Poza SoloLite, procesory zawierają VPU – *Video Processing Unit*. Pozwala on kodować i dekodować obraz w formacie HD 1080p.

Najnowszym dodatkiem do rodziny układów i.MX6 jest **SoloX**. Układ zawiera pojedynczy rdzeń Cortex-A9 oraz długi – Cortex-M4. Dodatkowe peryferie to dwa porty Ethernet, sprzętowe wsparcie przetwarzania multimediów i inne interfejsy przydatne

i.MX 6 Series Applications Processor Block Diagram



 Available on certain product families

Rys. 4. Ogólny schemat blokowy. Elementy oznaczone linią przerywaną występują tylko w niektórych wariantach. Źródło: materiały producenta (IMX6SRSFS)

w zastosowaniach automotive. Oba rdzenie będą miały dostęp do wspólnej przestrzeni adresowej. Technologia została przeniesiona do rodziny i.MX6 z linii produktów Wybrid. Wsparcie dla nowego układu znajduje się już w nowych wersjach Linuksa (od 3.16).

Cortex-A9 pracuje pod kontrolą systemu Linux, a Cortex-M4 uruchamia aplikację bare-metal lub jeden z dostępnych RTOS-ów. Ten dodatkowy rdzeń nie obsługuje pamięci wirtualnej – co poprawia determinizm aplikacji i zmniejsza opóźnienia. Układ może być z powodzeniem używany w zastosowaniach automotive oraz w systemach sterowania wymagających reakcji w czasie rzeczywistym.

Rysunek 4 przedstawia ogólny schemat blokowy procesorów i.MX6.

Układy rodziny i.MX6 zostały zaprojektowane z myślą o aplikacjach multimedialnych. Sprawdzają się we wszelkiego rodzaju urządzeniach typu smart – od mobilnych terminali, przez urządzenia typu infotainment w samochodach i różnych środkach komunikacji, po urządzenia medyczne i instrumenty pomiarowe.

Uniwersalność i elastyczność zastosowanej architektury, pozwala na użycie jej w wielu różnych projektach. Ilość pamięci operacyjnej, a także podłączone ze-

wewnętrzne komponenty i wyprowadzone złącza zależą od konkretnego urządzenia i pomysłów projektanta.

Zestawy deweloperskie

Na rynku dostępnych jest wiele zestawów deweloperskich, które pozwalają rozpocząć pracę z układami i.MX6: od dość drogiego zestawu referencyjnego producenta, przez projekty non profit, po stosunkowo niedrogie urządzenia konsumenckie typu odtwarzacz multimedialny do telewizora.

Podstawowym kryterium wyboru powinna być jakość dostępnej dokumentacji urządzenia. Sam procesor jest dobrze udokumentowany, jednak nie wystarczy to do podłączenia do urządzenia dodatkowych peryferii czy uruchomienia innej niż dostarczona wersja jądra. Do wygodnej pracy, rozwijania i debugowania sterowników i aplikacji konieczna jest co najmniej znajomość schematu blokowego urządzenia oraz specyfikacja umieszczonych na nim dodatkowych układów.

Jak już wspomnieliśmy, wyprowadzenia układów są ze sobą kompatybilne. Dzięki temu podejściu stosunkowo łatwo jest zaprojektować urządzenie, w którym, w zależności od potrzeb, montowany będzie tańszy i słabszy lub droższy i mocniejszy układ. Ma to odzwierciedlenie w ofercie zestawów deweloperskich wiele z nich występuje w kilku wariantach.

Podczas pisania tej książki używane były niektóre z dostępnych urządzeń. Prezentowane przykłady bazują przede wszystkim na: SabreLITE, Wandboard oraz Gateworks Ventana a także innych, wymienionych poniżej (subiektywnej) liście:

- Embest SABRE Lite Development Kit – dostępny m.in. w ofercie Kamami. Jest to oficjalny zestaw deweloperski zbudowany według specyfikacji producenta – „tania” wersja oficjalnych zestawów SABRE (rysunek 5). Zawiera procesor i.MX6 Quad.

Według tej samej specyfikacji zbudowany jest produkt firmy BoundaryDevices – BD-SL-i.MX6 (<http://boundarydevices.com/products/sabre-lite-ix6-sbc/>). Także wzbogacony o dodatkowe komponenty Nitrogen6x tej samej firmy.

- Wandboard – <http://wandboard.org/>. Jest to niskokosztowy zestaw deweloperski projektowany przez organizację non profit. Urządzenie składa się z dwóch oddzielnych płyt połączonych ze sobą złączem brzegowym o standardowym układzie pinów (EDM). Występuje w trzech wariantach różniących się procesorem i peryferiami (rysunek 6).

W książce użyto wersji i.MX6 Solo oraz Quad (rysunek 7).

- Gateworks Ventana – <http://www.gateworks.com/product/item/ventana-gw5410-network-processor> jest kompletnym komputerem jednopłytkowym (z możliwością dołączania rozszerzeń). Został zbudowany z komponentów spełniających standardy przemysłowe (rysunek 8).

i.MX 6 Reference Designs

	i.MX 6 i.MX 6	i.MX 6 i.MX 6	i.MX 6 i.MX 6	i.MX 6 i.MX 6	i.MX 6 i.MX 6
	i.MX 6Quad • Dual DDR	i.MX 6Dual • Dual DDR	i.MX 6Dual Lite • Dual DDR	i.MX 6Solo • Single DDR • EPD	i.MX 6SoloLite • Single DDR • EPD
SABRE Platform for Smart Devices (\$999) Sale target: Q4'12	MCIMX6Q-SDP		MCIMX6DL-SDP		
SABRE Board for Smart Devices (\$399) Sale target: Q4'12	MCIMX6Q-SDB				
i.MX 6SLEVK (\$599) eReader Segment Sale target: Q4'12					✓

Rys. 5. Referencyjne zestawy deweloperskie oferowane przez firmę Freescale

- RIOTBoard – <http://riotboard.org/> – Oprócz procesora i.MX6 Solo zawiera także mikrokontroler Freescale Kinetis K20 oraz zintegrowany układ zasilania (rysunek 9).
- Zestaw UDOO – <http://www.udoo.org/> – dostępny jest w wariacji z układami i.MX6 Solo, Dual i Quad. Dodatkowo zawiera mikrokontroler Atmel SAM3X8E oraz złącze kompatybilne z Arduino (rysunek 10).
- Projekt iMX6 Rex – <http://www.imx6rex.com/> – to projekt otwartoźródłowego zestawu komputerowego. Cała dokumentacja, łącznie z danymi potrzebnymi do wyprodukowania PCB, dostępna jest na otwartej licencji. Dużo pracy przeznaczono na detale: moduł z procesorem jest mały, referencyjne urządzenie przeszło testy EMC i wiele innych badań (rysunek 11).
- Novena – <http://www.kosagi.com/> – to projekt laptopa open source i open hardware – przyjaznego dla zaawansowanych użytkowników. Zbudowany na bazie procesora i.MX6, wspiera liczne peryferie i dodatkowe płyty rozszerzające możliwości zestawu. Jednym z celów projektu jest wsparcie oficjalnej wersji jądra (rysunek 12).
- W przypadku procesora i.MX6 SoloLite, oprócz oficjalnej platformy ewaluacyjnej, dostępny jest także referencyjny projekt miniaturowego urządzenia przenośnego (rysunek 13).

The screenshot shows the Wandboard website in a Firefox browser. The page features the Wandboard logo, navigation links (DETAILS, DOWNLOADS, FORUMS, WIKI, FAQ, ABOUT, BUY), and a 'Specifications' section. A table compares the Solo, Dual, and Quad configurations across various hardware features. To the right, there is an image of the Wandboard Solo board with its price listed as 79 USD. Below the image is another table titled 'Wandboard Specifications' which provides a more detailed breakdown of the hardware features for each configuration.

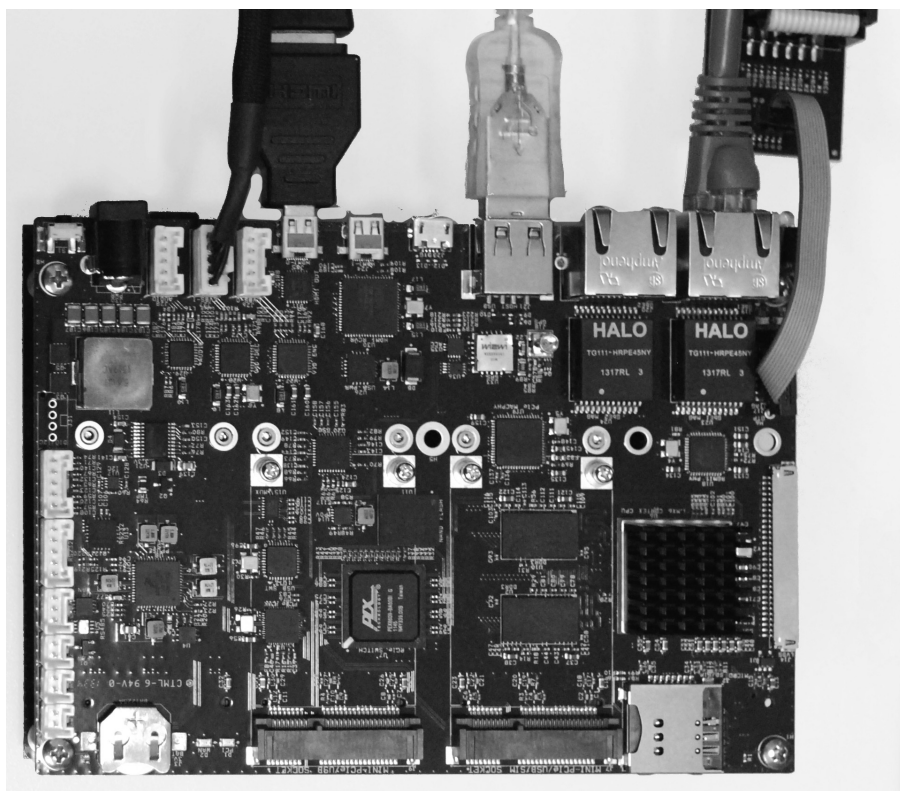
	Wandboard Solo	Wandboard Dual	Wandboard Quad
Processor	Freescal e i.MX6 Solo	Freescal e i.MX6 Duallite	Freescal e i.MX6 Quad
Cores	Cortex-A9 Single core	Cortex-A9 Dual core	Cortex-A9 Quad core
Graphic engine	Vivante GC 880 + Vivante GC 320	Vivante GC 880 + Vivante GC 320	Vivante GC 2000 + Vivante GC 355 + Vivante GC 320
Memory	512 MB DDR3	1GB DDR3	2GB DDR3
Audio	✓	✓	✓
Optical S/PDIF	✓	✓	✓
HDMI	✓	✓	✓
Camera Interface	✓	✓	✓
Micro-SD slot	2	2	2
Serial port	✓	✓	✓
Expansion Header	✓	✓	✓
USB	✓	✓	✓
USB OTG	✓	✓	✓
SATA	NO	NO	✓
Gigabit LAN	✓	✓	✓
WiFi (802.11n)		✓	✓
Bluetooth		✓	✓
	79 USD	99 USD	129 USD

	Wandboard Solo	Wandboard Dual	Wandboard Quad
Processor	Freescal e i.MX6 Solo	Freescal e i.MX6 Duallite	Freescal e i.MX6 Quad
Cores	Cortex-A9 Single core	Cortex-A9 Dual core	Cortex-A9 Quad core
Graphic engine	Vivante GC 880 + Vivante GC 320	Vivante GC 880 + Vivante GC 320	Vivante GC 2000 + Vivante GC 355 + Vivante GC 320
Memory	512 MB DDR3	1GB DDR3	2GB DDR3
Audio	✓	✓	✓
Optical S/P DIF	✓	✓	✓
HDMI	✓	✓	✓
Camera Interface	✓	✓	✓
Micro-SD slot	2	2	2
Serial port	✓	✓	✓
Expansion Header	✓	✓	✓
USB	✓	✓	✓
USB OTG	✓	✓	✓

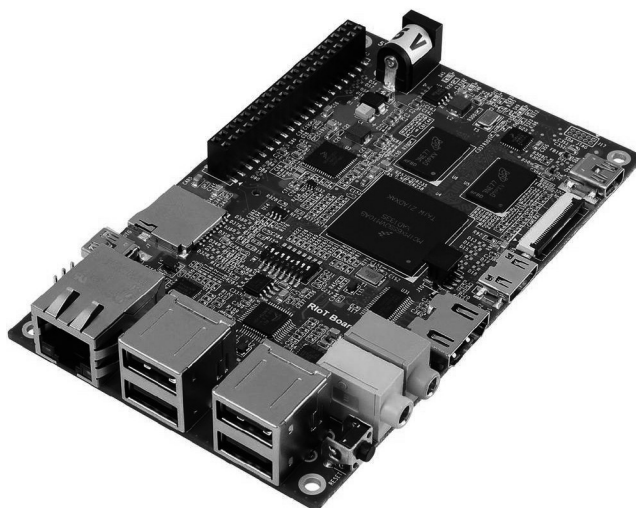
Rys. 6. Wandboard – warianty konfiguracji sprzętu



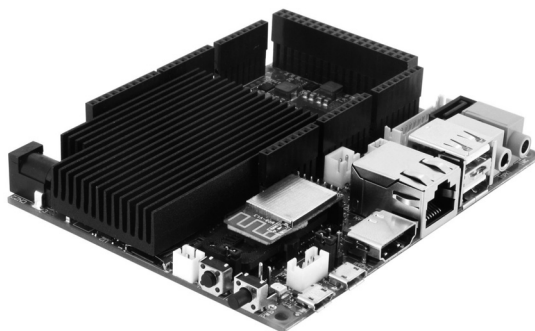
Rys. 7. Wandboard Quad w obudowie z zewnętrzną anteną Wi-Fi/Bluetooth



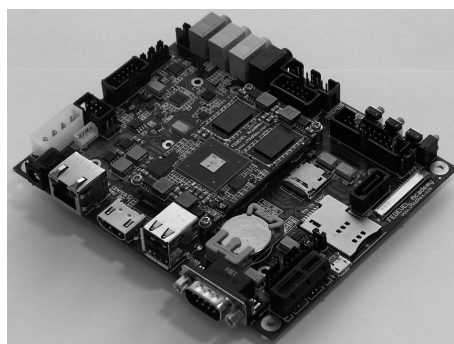
Rys. 8. Gateworks Ventana – <http://gateworks.com/>. Urządzenie działające pod kontrolą i.MX6, do którego można podłączać karty miniPCI



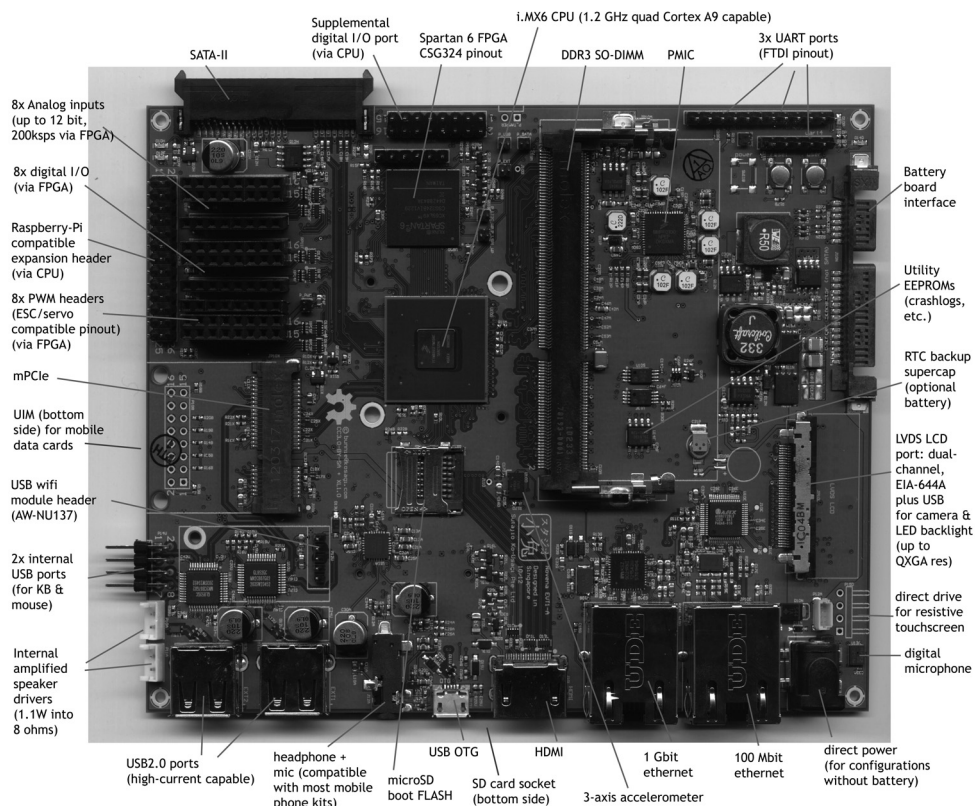
Rys. 9. RIoTBoard – <http://riotboard.org/>. Źródło: materiały prasowe



Rys. 10. UDOO – <http://www.udoo.org/>. Źródło: materiały prasowe



Rys. 11. REX – <http://www.imx6rex.com/> – projekt open source – udostępnia wszystkie pliki projektowe urządzenia. Źródło: materiały prasowe



Rys. 12. Novena – <http://www.kosagi.com/> – zdjęcie przedstawia pierwszą wersję urządzenia (z podpisanymi komponentami), obecne prototypy nieznacznie różnią się układem złączy. Źródło: materiały prasowe



Rys. 13. Referencyjna platforma – miniaturowe urządzenie przenośne z procesorem i.MX6 SoloLite. Źródło: materiały prasowe