

## **Środowiska pozwalające przygotować Linuksa dla systemów wbudowanych**

Celem tego ćwiczenia jest praktyczne zapoznanie się z dwoma środowiskami (Buildroot i OpenWRT), umożliwiającymi stworzenie dystrybucji Linuksa dla systemu wbudowanego, z tworzeniem własnej aplikacji w języku C w danym środowisku, oraz uruchomieniem zbudowanego systemu wraz z dodaną aplikacją w emulatorze QEMU.

### **Materiał do przejrzenia**

Slajdy z wykładów 1 i 2, ewentualnie 3.

Materiały do ćwiczenia 1 udostępnione na serwerze studia - „cw1\_eksperymenty.html” i

„cw1\_przykłady.tar.gz”.

Oczywiście korzystne może być też zapoznanie się z innymi źródłami.

### **Wprowadzenie**

Proste komputery mogące działać pod kontrolą systemu operacyjnego Linux są szeroko dostępne i pozwalają tworzyć uniwersalne systemy wbudowane. Jak to przedstawiono w wykładzie 1, możecie Państwo łatwo uzyskać dostęp do takich systemów zarówno w wersji przeznaczonej do projektów hobbystycznych lub eksperymentalnych, jak i w wersji profesjonalnej, przeznaczonej do zastosowań przemysłowych.

Istotne jest też to, że do pierwszych eksperymentów nie trzeba od razu używać rzeczywistego systemu wbudowanego. Emulator QEMU pozwala przeprowadzić realistyczną symulację działania takiego systemu, pracującego pod kontrolą systemu operacyjnego Linux. To właśnie podejście zastosowaliśmy w laboratorium, gdzie do eksperymentów zostanie użyty system „virt” z 64-bitowym procesorem ARM. Został on wybrany tak, aby umożliwić dobrą współpracę z używanymi w laboratorium środowiskami do tworzenia własnej wersji systemu Linux – Buildroot’em i OpenWRT. W tym ćwiczeniu będziecie Państwo mieli okazję uruchomić samodzielnie stworzony system Linux, z dołączoną przez Was prostą aplikacją wielowątkową.

*Jeśli chcecie Państwo przeprowadzić samodzielnie testy z systemem 32-bitowym, w przypadku środowiska OpenWRT możecie użyć systemu „virt” w wersji 32-bitowej, a w przypadku środowiska Buildroot, systemu „vexpress”.*

*Jeśli macie Państwo dostęp do rzeczywistego systemu wbudowanego obsługiwanego przez Buildroot’a lub OpenWRT, zachęcam do próby przeniesienia stworzonej przez Was aplikacji do systemu stworzonego dla tego systemu. Pozwoli to Wam przekonać się, że eksperymenty przeprowadzone w naszym laboratorium nie są jedynie sztuką dla sztuki, ale mogą bardzo szybko znaleźć praktyczne zastosowanie. Co prawda wielu producentów umożliwia uruchamianie na swoich minikomputerkach standardowych dystrybucji Linuksa (np. różne ograniczone wersje Ubuntu, specjalna wersja systemu Debian – Armbian, itp.), jednak jeśli chcemy stworzyć system oszczędnie gospodarujący zasobami sprzętowymi i mocą zasilania, jeśli chcemy mieć pełną kontrolę nad tym, jakie procesy konkurują o czas procesora, to stworzenie własnej wersji Linuksa, dostosowanej do konkretnych potrzeb może być najlepszym rozwiązaniem.*

### **Zadania:**

1. Wykorzystując środowisko Buildroot, przygotować obraz systemu Linux, dający się uruchomić na symulowanej 64-bitowej maszynie "virt" w emulatorze QEMU.
  1. Nazwa systemu, powinna mieć postać "imię\_nazwisko" zgodnie z danymi studenta.
  2. System powinien być skonfigurowany tak, aby po uruchomieniu automatycznie łączył się z siecią, oraz ustawiał systemowy czas używając protokołu NTP.
2. Proszę stworzyć i skompilować w środowisku Buildroot (bez obowiązku tworzenia pakietu) własną dwuwątkową aplikację realizującą następujące funkcje:
  1. Wątek 1 odbiera z klawiatury komunikaty (kończone znakiem nowej linii) i przekazuje je do wątku 2.
  2. Wątek 2 odbiera kolejne komunikaty i wysyła je do zadanego komputera (najlepiej stacji roboczej, na której realizowane jest ćwiczenie) do programu "netcat" (nc), nasłuchującego na porcie TCP 10000. Adres komputera powinien być specyfikowany przez parametr w linii poleceń. Program powinien umożliwiać symulowanie opóźnień komunikacji przez sieć, pozwalając określić przez parametr linii poleceń ile sekund ma zajmować transmisja każdego komunikatu.

Aplikacja powinna być napisana tak, żeby przy braku komunikatów do nadania wątek nadający nie zajmował czasu procesora, a zarazem żeby jak najszybciej rozpoczynał transmisję otrzymanego komunikatu.

*Wskazówka: Aplikację można wstępnie uruchomić i przetestować w systemie Linux na stacji roboczej, lub na dowolnym komputerze PC.*
3. Proszę uruchomić na symulowanej w QEMU 64-bitowej maszynie "virt" środowisko OpenWRT, instalując niezbędne pakiety. Należy także pobrać i rozpakować środowisko OpenWRT SDK dla 64-bitowej maszyny "virt".
4. Wykorzystując udostępnione szablony, proszę przygotować własny pakiet OpenWRT ze stworzoną wcześniej aplikacją, skompilować go i uruchomić.

### **Cennik**

zadanie 1 – 0 do 2 pkt

zadanie 2 – 0 do 4 pkt

zadanie 3 – 0 do 2 pkt

zadanie 4 – 0 do 2 pkt

czytelność kodu, komentarze – -5 do 0 pkt

prezentacja wyników – -10 do 0 pkt

Maximum: 10 pkt.