

Linux w systemach wbudowanych

Ćwiczenie 3

przygotował: dr inż. Michał Kruszewski

2025-04-09

1 Cel ćwiczenia

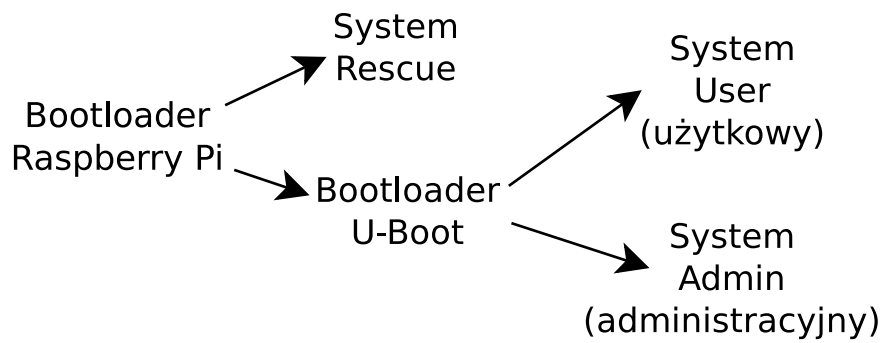
Celem ćwiczenia jest:

- zapoznanie się z bootloader'em U-Boot,
- implementacja prostego serwera uruchamianego na platformie Raspberry Pi.

Przed przystąpieniem do ćwiczenia należy zapoznać się z częścią wykładu 6 dotyczącą bootloader'a U-Boot.

2 Zadania do wykonania

1. Przygotowanie **administracyjnego** obrazu systemu Linux (podobnego do używanego podczas laboratorium systemu ratunkowego). System powinien:
 - Pracować z `initramfs`.
 - Zawierać narzędzia niezbędne do zarządzania kartą SD (partycjonowanie, formatowanie, kopiowanie na kartę nowego obrazu systemu przez sieć, naprawianie systemu plików itp.). W szczególności ten system zostanie wykorzystany do podziału karty na 3 partycje:
 - VFAT z systemem ratunkowym (w katalogu `rescue`), oraz administracyjnym i jądrem systemu użytkowego (w katalogu `user`) (w praktyce powinna tu zostać nasza oryginalna pierwsza partycja).
 - `ext4` - z systemem plików systemu użytkowego.
 - `ext4` - z danymi systemu użytkowego.
2. Przygotowanie **użytkowego** systemu Linux pracującego z systemem plików `ext4` na drugiej partycji, realizującego jedną z poniższych funkcji:
 - Sieciowy serwer plików (np. SMB/CIFS), udostępniający trzecią partycję karty SD jako dysk sieciowy, z możliwością tworzenia kont użytkowników i ich uwierzytelniania.
 - Serwer WWW (zrealizowany na bazie jednego z serwerów HTTP oferowanych w Buildroot, albo jako aplikacja webowa w środowisku Flask, Tornado lub podobnym). Serwer powinien udostępniać pliki z partycji 3 na karcie SD, wyświetlając listę tych plików i pozwalając na wybranie pliku do załadowania. Serwer powinien umożliwiać uwierzytelnionym użytkownikom wgrywanie nowych plików na tę partycję.
 - Serwer wybranego protokołu komunikatora internetowego, umożliwiający tworzenie kont użytkowników i prowadzenie rozmów między zalogowanymi użytkownikami.
3. Przygotowanie bootloader'a, umożliwiającego określenie, który system (administracyjny czy użytkowy) ma zostać załadowany. **Uwaga!** Nasz bootloader będzie współpracować z dotychczasowym bootloader'em bazującym na oryginalnym firmware. Powinien on być uruchamiany jako system użytkownika. Z uwagi na ograniczenia Raspberry Pi, bootloader musi przekazać parametry jądra przekazane przez oryginalny firmware, nie może też zmieniać drzewa urządzeń (bo jest ono dostosowane przez ten firmware).



Przygotowany bootloader powinien sygnalizować swoje działanie w następujący sposób:

1. **Żółta** dioda powinna zasignalizować, że za chwilę zostaną sprawdzone przyciski.
2. Po sekundzie należy odczytać stan przycisków. Jeśli żaden przycisk nie zostanie wciśnięty, powinien zostać załadowany system użytkowy.
3. Po sprawdzeniu przycisków **żółta** dioda powinna zostać zgaszona. Jeśli został wybrany system użytkowy, powinna zostać zapalona dioda **zielona**, a jeśli został wybrany system administracyjny, powinna zostać zapalona dioda **czerwona**.