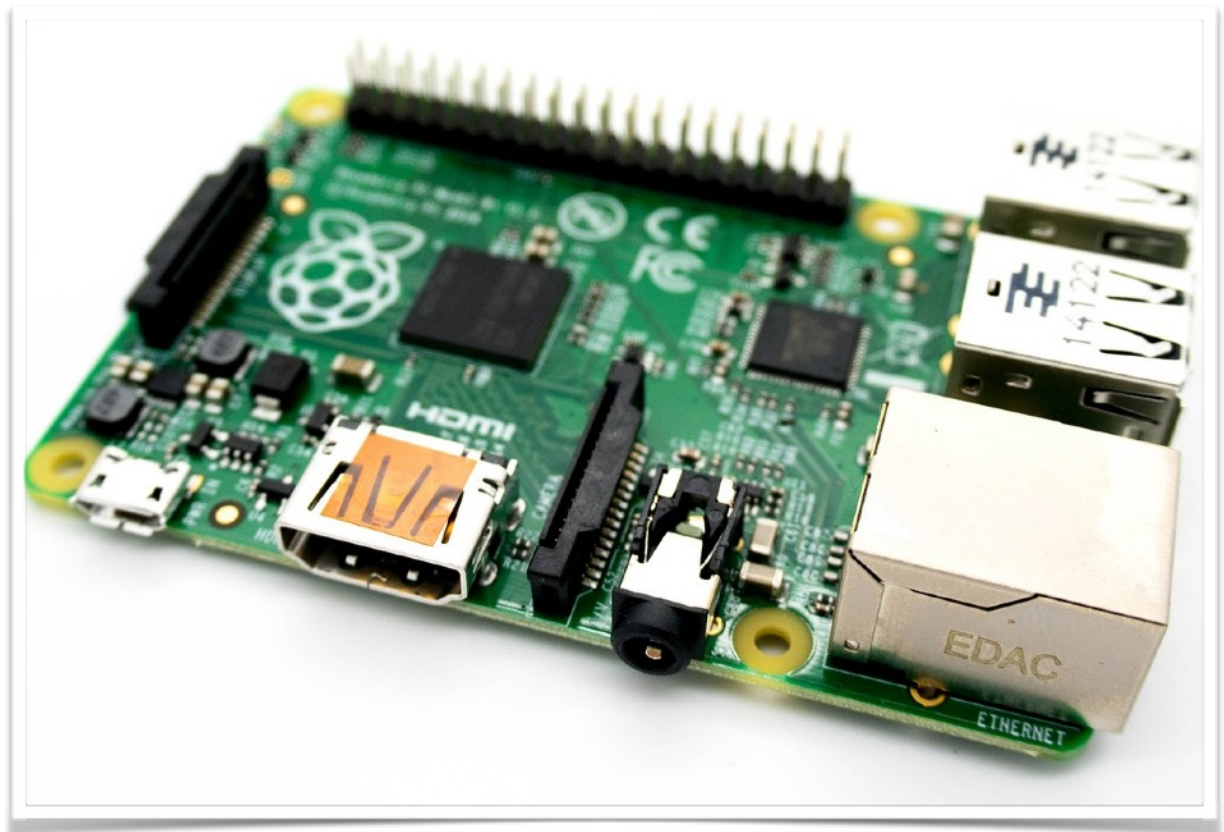


Linux w Systemach Wbudowanych



Laboratorium 1

Mateusz Sadowski
Lato 2018

1. Przygotowanie środowiska pracy

1.1. Przygotowanie środowiska Buildroot

Najnowszą wersję Buildroot'a można pobrać uruchamiając następującą komendę w katalogu roboczym:

```
wget https://buildroot.org/downloads/buildroot-2018.02.tar.bz2
```

Aby rozpakować program:

```
tar -xjf buildroot-2018.02.tar.bz2
```

Aby ustawić docelowy system na kompatybilny z Raspberry Pi:

```
cd buildroot-2018.02
```

```
make raspberrypi_defconfig
```

1.2. Podłączenie Raspberry Pi

Aby uzyskać możliwość komunikacji z Raspberry Pi należy połączyć ją z komputerem za pomocą interfejsu **UART**. Podłączenie konwertera USB/UART zostało opisane na wykładzie.

Po podłączeniu Raspberry Pi przez UART, podłączeniu karty SD oraz kabla Ethernet możemy podłączyć zasilanie za pomocą wejścia Micro USB.

Aby skomunikować się z Raspberry Pi należy skorzystać z programu **minicom**. Odpowiednia komenda:

```
minicom -D /dev/ttyUSB0 -o
```

Bardzo istotne jest, aby w ustawieniach programu wyłączyć opcję **Hardware flow control** która może powodować blokadę komunikacji. Szybkość transmisji powinna być ustawiona na **115200**.

1.3. Konfiguracja systemu

Aby rozpocząć konfigurację systemu należy wykonać polecenie:

```
make menuconfig
```

Spowoduje to otwarcie programu ułatwiającego konfigurację. Ustawienia zostaną zapisane w pliku **.config** w głównym katalogu Buildroota.

Istotne ustawienia dla naszego systemu niezwiązane z dalszymi ćwiczeniami zostały opisane w poradniku do laboratorium: ustawienie TTY port na **ttyAMA0**, Target ABI na **EABI**, Toolchain Type na

External toolchain oraz Toolchain na **Sourcery CodeBench ARM 2014.05**.

Aby znacząco skrócić kompilację systemu został udostępniony specjalny serwer przechowujący aktualne źródła pakietów pobieranych przez środowisko Buildroot. Żeby skorzystać z tej możliwości należy ustawić:

*Build options/Mirrors and Download locations/Primary download site/***http://192.168.137.24/dl**

1.4.Kompilacja systemu

Aby stworzyć obraz systemu należy wykonać komendę:

```
make
```

O ile wszystkie wcześniejsze kroki zostały wykonane poprawnie, może to zająć kilkanaście minut. Kolejne kompilacje systemu nie będą wymagały przebudowania całości i będą zajmować zdecydowanie krócej. Obraz systemu zostanie zapisany w katalogu **/output/images**.

1.5.Uruchomienie systemu

Aby uruchomić utworzony system trzeba wgrać go na kartę pamięci Raspberry Pi. Zrobimy to z komputera za pomocą protokołu **SSH**. W tym celu uruchomimy płytke w trybie **systemu ratunkowego**. Należy podłączyć Raspberry Pi jak w podpunkcie 1.2. W czasie do **2 sekund** należy wcisnąć **spację** w odpowiedzi na stosowny komunikat. Wyświetlona zostanie zachęta do wpisywania poleceń. Należy wprowadzić:

```
fatload mmc 0:1 $loadaddr rescue.scr
```

```
source $loadaddr
```

Logujemy się na konto **root** i wykonujemy następujące komendy:

```
udhcpc
```

- aby pobrać adres ip dla płytki

```
passwd
```

- aby ustawić hasło dla użytkownika root, potrzebne do przesłania obrazu

```
mkdir /tmp/d; mount /dev/mmcblk0p1 /tmp/d
```

- aby zamontować kartę SD na utworzonym katalogu /tmp/d

Następnie możemy już wgrać obraz systemu na Raspberry Pi z komputera:

```
scp path/to/image/zImage root@192.168.143.xxx:/tmp/d
```

- gdzie 192.168.143.xxx to adres nadany płytce.

2. Ćwiczenia

2.1. Raspberry Pi powinno automatycznie podłączyć się do sieci, używając DHCP do otrzymania parametrów sieci. Jeśli kabel sieciowy nie jest podłączony, połączenie powinno zostać zestawione automatycznie po podłączeniu kabla. Odłączenie kabla powinno powodować wyłączenie połączenia sieciowego, a jego ponowne podłączenie – ponowne zestawienie

Z dokumentacji Linuxa można dowiedzieć się, że daemon **netplugd** jest odpowiedzialny za obsługiwanie zdarzeń pochodzących od jądra Linuxa, związanych z połączeniami sieciowymi. M.in. uzyskaniem oraz utratą połączenia.

Aby dodać tę funkcjonalność do systemu należy wybrać odpowiedni pakiet podczas konfiguracji Buildroota:

Przejdź **Target packages**/Przejdź **Networking applications**/Wybierz **netplug**

2.2. System powinien mieć ustaloną nazwę jako: „imie_nazwisko” autora, lub „nazwisko1_nazwisko2” dla zespołu 2-osobowego

Przejdź **System configuration**/Wybierz **hostname**/Wpisz „**mateusz_sadowski**”

2.3. Przy starcie systemu, (lub po podłączeniu kabla sieciowego jeśli nie był on podłączony podczas startu), zegar systemowy powinien być automatycznie synchronizowany z jednym z serwerów NTP (najlepiej z wykorzystaniem „NTP server pool”)

Z dokumentacji Linuxa można dowiedzieć się, że daemon **ntpd** jest odpowiedzialny za ustawianie i aktualizację czasu systemowego na podstawie standardów sieciowych. Aby dodać tę funkcjonalność do systemu należy wybrać odpowiedni pakiet podczas konfiguracji Buildroota:

Przejdź **Target packages**/Przejdź **Networking applications**/Wybierz **ntpd**

Żeby ustawienia daty i godziny były poprawne należy dodatkowo ustawić prawidłową strefę czasową w konfiguracji Buildroota:

Przejdź **System configuration**/Wybierz **Install timezone info**/Wpisz **Europe/Warsaw**

2.4. Obraz systemu powinien zawierać dodatkowe programy wskazane przez prowadzącego (np. serwer HTTP udostępniający statyczne strony WWW, interpreter języka skryptowego z prostym skryptem pokazującym jego działanie itp.).

Jako dodatkowy program zostanie zainstalowany serwer HTTP. Żeby go dodać trzeba przeprowadzić następujące kroki:

Przejdź **Target packages**/Przejdź **Networking applications**/Wybierz **thttpd**

Z opisu thttpd z menu konfiguracyjnego Buildroota: „thttpd is a simple, small, portable, fast, and secure HTTP server”.

Aby wyświetlić stronę www należy ją umieścić w katalogu **/var/www/data**. Jednak ten katalog jest przywracany do stanu domyślnego przy każdym uruchomieniu systemu. Aby strona była dostępna przy każdym uruchomieniu należy skorzystać z narzędzia **overlay**.

Za pomocą metody overlay można „nadpisać” pewne katalogi i ustawienia naszego systemu wcześniej zdefiniowanymi na etapie konfiguracji systemu. W celu wykonania ćwiczenia należy stworzyć katalog *board/<boardname>/overlay/var/www/data* i umieścić w nim plik **index.html** zawierający prostą stronę www. Oraz podać tę ścieżkę w:

Przejdź **System configuration**/Przejdź **Root filesystem overlay directories**/Wpisz **board/<boardname>/overlay/var/www/data**

Od tego momentu po każdym uruchomieniu systemu w katalogu */var/www/data* będzie się znajdował plik index.html. Po wpisaniu adresu ip płytki w przeglądarce komputera wyświetli się strona zdefiniowana w pliku index.html.

2.5. Oprócz użytkownika „root” w systemie powinno zostać stworzone konto użytkownika domyślnego (o dowolnie wybranej nazwie). Zarówno hasło użytkownika „root”, jak i hasło użytkownika domyślnego powinny mieć ustawione hasła.

W tym celu należy zdefiniować **users table**. Jest to zwykły plik zawierający informacje o użytkownikach. Użytkownicy definiowani są po jednym na linię. Obowiązujący format:

username	uid	group	gid	password	home	shell	groups	comment
----------	-----	-------	-----	----------	------	-------	--------	---------

Dla przykładu wpis w users table może wyglądać następująco:

```
someusr -1 normal -1 =password /homedir/someusr /bin/sh - user
```

Wartość *-1* pod *uid* oraz pod *gid* spowoduje, że system automatycznie wygeneruje te pola. Znak *=* przed hasłem spowoduje, że będzie ono traktowane jako zwykły tekst oraz w konsekwencji umożliwi logowanie na użytkownika za pomocą hasła.

Plik należy zapisać w katalogu *board/<boardname>*. Następnie trzeba podać ścieżkę do tego pliku w konfiguracji systemu:

Przejdź **System configuration**/Przejdź **Path to the users tables**/Wpisz **board/<boardname>/<userstablefile>**

W celu ustawienia hasła dla konta **root** należy wykonać następujące kroki:

Przejdź **System configuration**/Wybierz **Root password**/Wprowadź **<nowehasło>**

3. Odniesienia

3.1.<https://linux.die.net/man/>

3.2.<https://buildroot.org/docs.html>