

# Systemy Komputerowe w Sterowaniu i Pomiarach

## Laboratorium 1

Aleksander Kruk

Michał Sobiech

17 października 2023

## Praca domowa    Obraz na 64-bitową maszynę wirtualną "Virt" i uruchomienie w QEMU

### Wersja używająca ramdysku startowego

#### Użyte komendy

<code>curl https://buildroot.org/downloads/buildroot-2023.08.1.tar.gz -O</code>	Pobranie buildroota
<code>tar -xf buildroot-2023.08.1.tar.bz2</code>	Rozpakowanie buildroota
<code>cd buildroot-2023.08.1</code>	
<code>make qemu_aarch64_virt_defconfig</code>	Przygotowanie domyślnej konfiguracji buildroota dla qemu
<code>make menuconfig</code>	Uruchomienie menu konfiguracyjnego

#### Wybrane opcje

#### Status

Toolchain::Toolchain type::External toolchain	✓
Filesystem images::cpio the root filesystem	✓
Filesystem images::cpio the root filesystem::Compression method	gzip
Filesystem images::ext2/3/4 root filesystem	✗
Filesystem images::initial RAM filesystem linked into linux kernel	✓
System configuration::System hostname	Kruk.Sobiech

#### Użyte komendy

<code>make clean all</code>	uruchamiamy kompilację ze sprzątaniem
<code>qemu-system-aarch64 -M virt -cpu cortex-a57 -nographic -smp 4</code>	Uruchamiamy qemu
<code>-kernel output/images/Image -append "console=ttyAMA0"</code>	

### Wersja używająca rzeczywistego systemu plików

#### Użyte komendy

<code>make linux-dirclean</code>	Posprzątanie po poprzedniej kompilacji
<code>make menuconfig</code>	Ponowne uruchomienie menu konfiguracyjnego
Wybrane opcje	Status

Toolchain::Toolchain type::External toolchain	✓
Filesystem images::cpio the root filesystem	✗
Filesystem images::ext2/3/4 root filesystem	✓
Filesystem images::ext2/3/4 root filesystem::Compression method	gzip
Filesystem images::initial RAM filesystem linked into linux kernel	✗
System configuration::System hostname	Kruk.Sobiech

#### Użyte komendy

<code>make clean all</code>	uruchamiamy kompilację ze sprzątaniem
<code>qemu-system-aarch64 -M virt -cpu cortex-a57 -nographic -smp 4</code>	Uruchomienie qemu
<code>-kernel output/images/Image -append "console=ttyAMA0"</code>	
<code>-append "root=/dev/vda console=ttyAMA0"</code>	
<code>-drive file=output/images/rootfs.ext2,if=none,format=raw,id=hd0</code>	
<code>-device virtio-blk-device,drive=hd0</code>	

## Zadanie 1 Złożenie stanowiska laboratoryjnego: zestaw z Raspberry Pi 4B (RPi)

Zestaw złożyliśmy według poleceń instrukcji.

## Zadanie 2 Pierwsze uruchomienie RPi, sprawdzenie połączenia sieciowego, wykonanie próbnych transferów plików,

### Użyte komendy

<code>minicom -D /dev/ttyUSB0</code>	Podłączenie do terminala
<code>udhcpd</code>	Zlokalizowanie komputera przez płytke
<code>ifconfig</code>	Uzyskanie adresów ip
<code>python3 -m http.server</code>	Uruchomienie serwera na porcie 8000
<code>wget http://192.168.9.127:8000/NAZWAPLIKU</code>	Pobranie pliku

## Zadanie 3 Zbudowanie za pomocą Buildroot obrazu Linuxa dla RPi, z init RAM fs

### Użyte komendy

<code>curl https://buildroot.org/downloads/buildroot-2023.08.1.tar.gz -O</code>	Pobranie buildroota
<code>tar -xjf buildroot-2023.08.1.tar.bz2</code>	Rozpakowanie buildroota
<code>cd buildroot-2023.08</code>	
<code>make raspberrypi4.64.defconfig</code>	Przygotowanie domyślnej konfiguracji buildroota
<code>make menuconfig</code>	Uruchomienie menu konfiguracyjnego

### Wybrane opcje

### Status

Toolchain::Toolchain type::External toolchain	✓
Filesystem images::cpio the root filesystem	✓
Filesystem images::cpio the root filesystem::Compression method	gzip
Filesystem images::ext2/3/4 root filesystem	✗
Filesystem images::initial RAM filesystem linked into linux kernel	✓
System configuration::System hostname	Kruk_Sobiech

### Użyte komendy

<code>make</code>	Uruchamianie kompilacji
<code>python3 -m http.server</code>	Uruchomienie serwera na porcie 8000
<code>mount /dev/mmcblk0p1 /mnt</code>	Zamontować partycję 1
<code>wget http://192.168.9.127:8000/Image</code>	Pobranie wymaganych plików na płytke
<code>wget http://192.168.9.127:8000/cmdline.txt</code>	
<code>wget http://192.168.9.127:8000/bcm2711-rpi-4-b.dtbcmdline.txt</code>	
<code>poweroff</code>	wyłączenie płytki

Po restarcie płytki z przytrzymaniem przycisku nasz skompilowany system zadziałał.

Wynikiem komendy `uname -a` było: Linux Kruk\_Sobiech 5.10.92-v8 #3 SMP PREEMPT Tue Oct 17 14:00:12 CEST 2023 aarch64 GNU/Linux

## Zadanie 4 Zbudowanie za pomocą Buildroot obrazu Linuxa dla RPi, z systemem plików na trwałym nośniku

Na drugim komputerze równocześnie uruchomiliśmy drugą kompilację po pobraniu buildroota:

Użyte komendy	
<code>make menuconfig</code>	Ponowne uruchomienie menu konfiguracyjnego
Wybrane opcje	
Toolchain::Toolchain type::External toolchain	✓
Filesystem images::cpio the root filesystem	✗
Filesystem images::ext2/3/4 root filesystem	✓
Filesystem images::ext2/3/4 root filesystem::Compression method	gzip
Filesystem images::initial RAM filesystem linked into linux kernel	✗
System configuration::System hostname	Kruk_Sobiech
Użyte komendy	
<code>make</code>	Uruchamiamy kompilację
<code>...</code>	Komendy do pobrania tak jak w poprzednim zadaniu
<code>wget http://192.168.9.127:8000/rootfs.ext2</code>	
<code>dd if=rootfs.ext2 of=/dev/mmcbk0p2 bs=4096</code>	Nagranie systemu plików na kartę SD

Po restarcie płytki z przytrzymaniem przycisku nasz skompilowany system zadziałał. Wynikiem komendy `uname -a` było:

```
Linux Kruk_Sobiech 5.10.92-v8 #1 SMP PREEMPT Tue Oct 17 14:00:09 CEST 2023 aarch64 GNU/Linux
```

Pliki utworzone przez nas po restarcie nie zniknęły.