Real Time Linux
(PREEMPT_RT)

Matthias Lüscher, 3. September 2018

"Controlling a laser with Linux is crazy, but everyone in this room is crazy in his own way. So if you want to use Linux to control an industrial welding laser, I have no problem with your using PREEMPT_RT."

Linus Torvalds, Kernel Summit 2006

Content

- Real time Linux **sample applications**

- **Enabling real time Linux** by using PREEMPT_RT

- **Architectural considerations** when using PREEMPT_RT

Sample Applications

- Example 1: **NI Linux Real-Time**
  http://www.ni.com/white-paper/14627/de/
  http://www.ni.com/pdf/product-flyers/compactrio-controller.pdf

- Example 2: Maybe the **industrial welding laser** Linus Torvalds was talking about (video)

- More Examples: **Industrial Companies**
  Join the OSADL (https://www.osadl.org) and ask the members about their usage of PREEMPT_RT

Enabling Real Time Linux (Generic)

Getting the sources:

```
wget https://www.kernel.org/pub/linux/kernel/v4.x/linux-4.4.12.tar.xz
wget https://www.kernel.org/pub/linux/kernel/projects/rt/4.4/patch-4.4.12-rt19.patch.xz
```

Applying the PREEMPT_RT patch set:

```
xz -cd linux-4.4.12.tar.xz | tar xvf -
cd linux-4.4.12
xzcat ../patch-4.4.12-rt19.patch.xz | patch -p1
```

Configuring the kernel:

The only necessary configuration for real time Linux kernel is the choice of the "Fully Preemptible Kernel" preemption model (CONFIG_PREEMPT_RT_FULL).

Building the kernel:

Building the kernel and starting the kernel works similarly to a kernel without PREEMPT_RT patch.

Source: https://wiki.linuxfoundation.org/realtime/documentation/howto/applications/preemptrt_setup

Enabling Real Time Linux (Debian, amd64)

For Debian, there are **real time images available** for the amd64 architecture:

```
sudo apt install linux-image-rt-amd64
```

Enabling Real Time Linux (Results)

Without PREEMPT_RT patch:

```
pi@raspberry:~$ uname -a
Linux raspberry 4.17.0-0.bpo.1-arm64 #1 SMP Debian 4.17.8-1~bpo9+1 (2018-07-23)
aarch64 GNU/Linux
pi@raspberry:~$ sudo cyclictest -p 99 -t5 -n -i250
# /dev/cpu_dma_latency set to 0us
policy: fifo: loadavg: 0.32 0.42 0.22 1/140 2387

T: 0 ( 1856) P:99 I:250  C:1736556 Min:     9 Act:   12 Avg:   14 Max:    4975
T: 1 ( 1857) P:99 I:750  C: 578996 Min:     9 Act:   14 Avg:   14 Max:    4315
T: 2 ( 1858) P:99 I:1250 C: 347406 Min:     9 Act:   14 Avg:   15 Max:    4318
T: 3 ( 1859) P:99 I:1750 C: 248151 Min:     9 Act:   13 Avg:   14 Max:    1195
T: 4 ( 1860) P:99 I:2250 C: 193006 Min:     9 Act:   12 Avg:   14 Max:     991
```
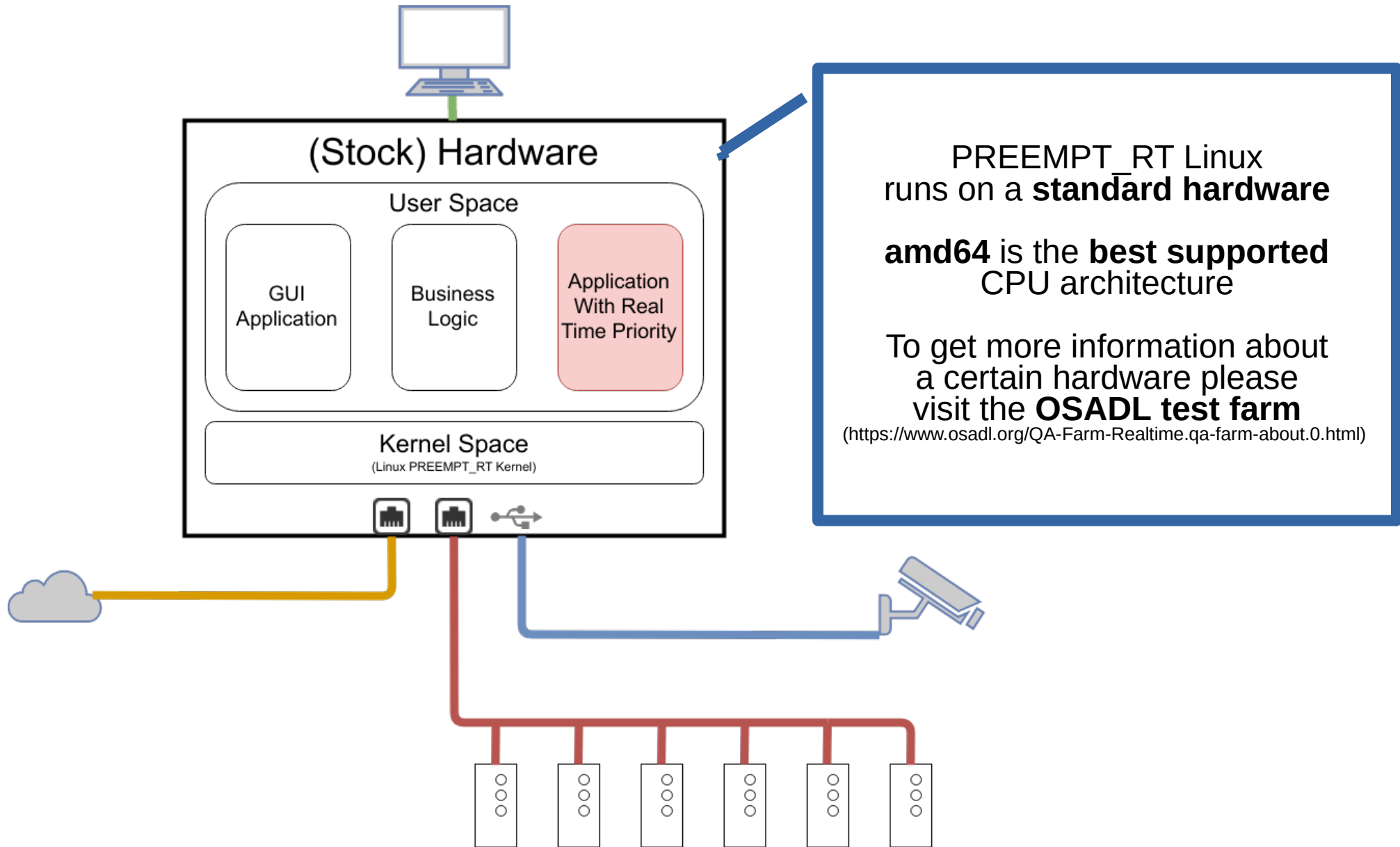
With PREEMPT_RT patch:

```
pi@raspberry:~$ uname -a
Linux raspberry 4.18.5-rt3-v8 #1 SMP PREEMPT RT Fri Aug 31 08:13:54 UTC 2018 aarch64
GNU/Linux
pi@raspberry:~$ sudo cyclictest -p99 -t5 -n -i250
# /dev/cpu_dma_latency set to 0us
policy: fifo: loadavg: 0.76 0.84 0.46 1/158 1526

T: 0 ( 1246) P:99 I:250  C: 995343 Min:    12 Act:   19 Avg:   23 Max:     152
T: 1 ( 1247) P:99 I:750  C: 331780 Min:    13 Act:   17 Avg:   19 Max:      94
T: 2 ( 1248) P:99 I:1250 C: 199068 Min:    14 Act:   21 Avg:   20 Max:     128
T: 3 ( 1249) P:99 I:1750 C: 142191 Min:    13 Act:   30 Avg:   29 Max:     151
T: 4 ( 1250) P:99 I:2250 C: 110593 Min:    14 Act:   19 Avg:   21 Max:     107
```
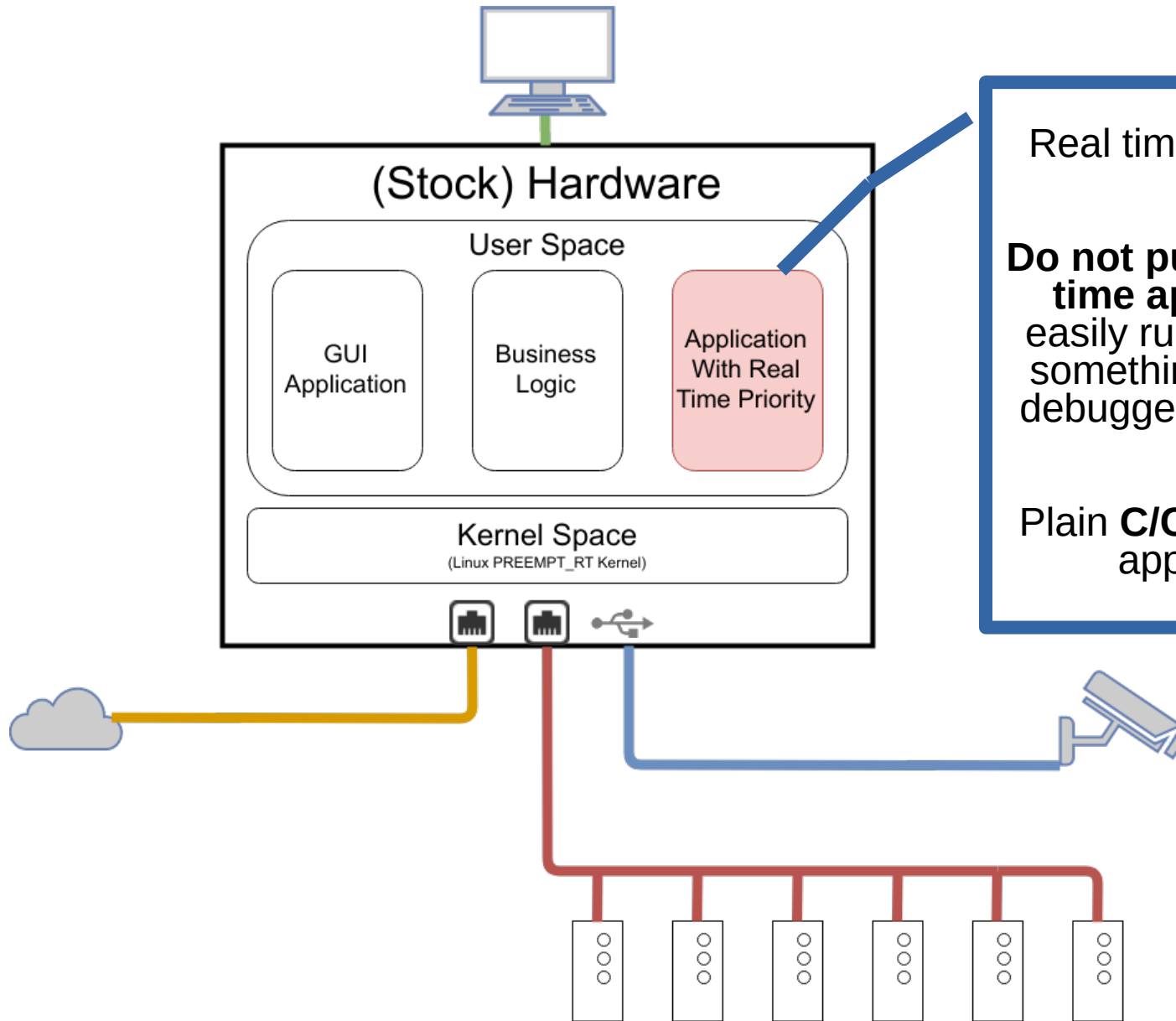
Test system: Raspberry Pi3, arm64
Setup: https://www.get-edi.io/A-new-Approach-to-Operating-System-Image-Generation/

Architectural Considerations (Hardware)

(Stock) Hardware

User Space

GUI Application

Business Logic

Application With Real Time Priority

Kernel Space
(Linux PREEMPT_RT Kernel)

PREEMPT_RT Linux
runs on a **standard hardware**

amd64 is the **best supported** CPU architecture

To get more information about a certain hardware please visit the **OSADL test farm**
(https://www.osadl.org/QA-Farm-Realtime.qa-farm-about.0.html)

Architectural Considerations (Software)

(Stock) Hardware

User Space

GUI Application

Business Logic

Application With Real Time Priority
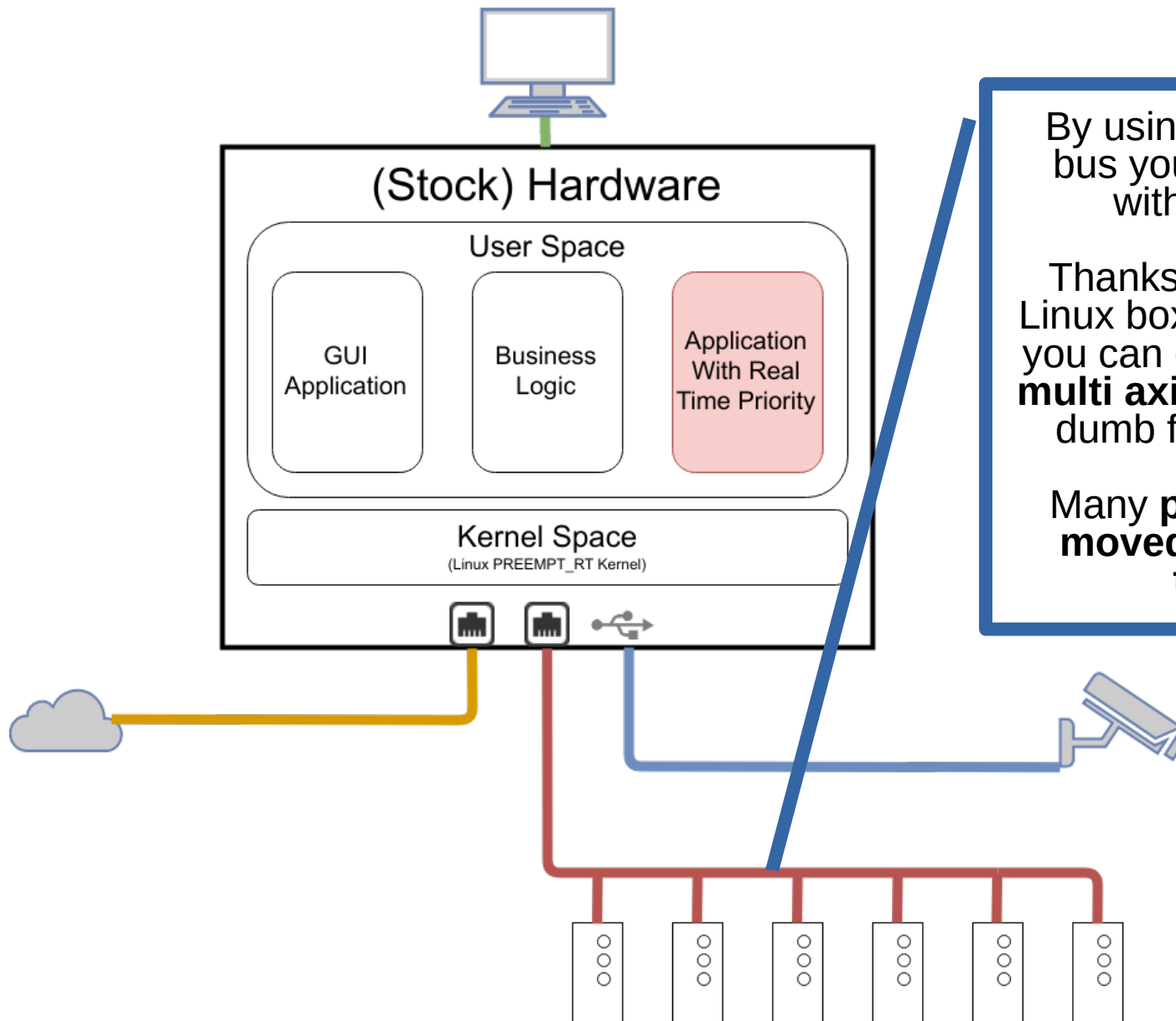
Kernel Space
(Linux PREEMPT_RT Kernel)

Real time application runs in **user space**

Do not put everything into the real time application since you can easily run into troubles when doing something wrong (and obviously a debugger would break the real time behavior)

Plain **C/C++** is suitable for real time application development

Architectural Considerations (Fieldbus)

(Stock) Hardware

User Space

GUI Application

Business Logic

Application With Real Time Priority

Kernel Space
(Linux PREEMPT_RT Kernel)

By using an Ethernet based field bus you can run **update cycles** with a frequency of **5kHz**

Thanks to the CPU power of the Linux box and its real time behavior you can centrally **control complex multi axis machines** with relatively dumb field bus controller nodes

Many **processing tasks** can be **moved** from the field bus node **to the Linux box**

Conclusion

- PREEMPT_RT patched Linux is **heavily used in industry**

- PREEMPT_RT Linux is a **brilliant combination** of a **general purpose** operating system with a **real time operating system**

- PREEMPT_RT patched Linux is in many cases a **suitable replacement for proprietary real time operating systems**

- CONFIG_PREEMPT_RT_FULL is **not yet fully merged into mainline Linux**

- Please consider to (financially) **support the great team** of Linux developers that make PREEMPT_RT available and further develop it