

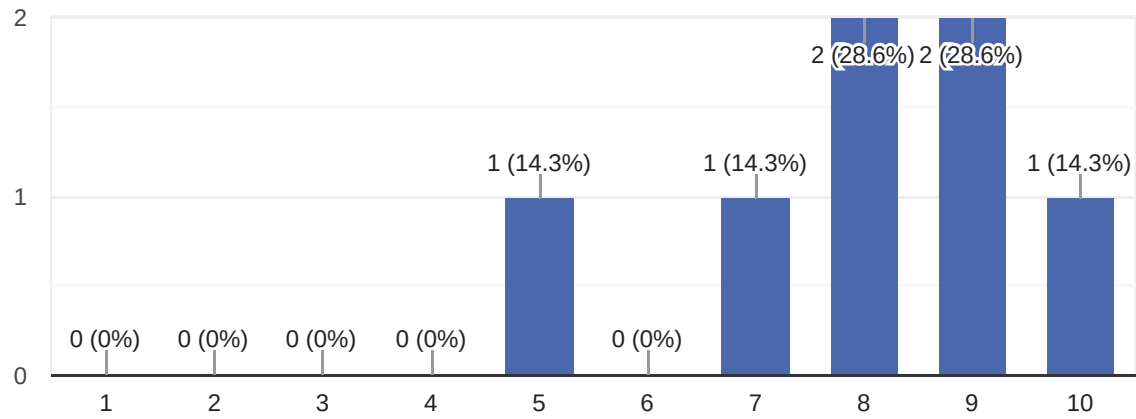
Bootlin training course evaluation

7 responses

Overall rating of the course

 [Copy chart](#)

7 responses



Comments and suggestions

5 responses

Very good.

I really liked the course and the overall stuff I have learned. However, I think the time slot was unpractical in the afternoon in summer and length of training sessions was too long. I would have preferred shorter sessions over a month or so.

1. Slow down during the practical demonstrations and explain the reasoning behind each step.
2. Reduce the amount of lecture material or spread it across a longer course.
3. Add more time for questions, troubleshooting, and independent lab work.
4. Provide additional diagrams and summaries to simplify complex topics.
5. Consider offering the course in person where possible, or make the online sessions more interactive with additional pauses for discussion.

Very nice training. The teacher explains well and the topics are interesting and relevant. The teacher has excellent knowledge about the topics. There is always space for questions. The group size is just perfect as well. Not too many people, not too few.

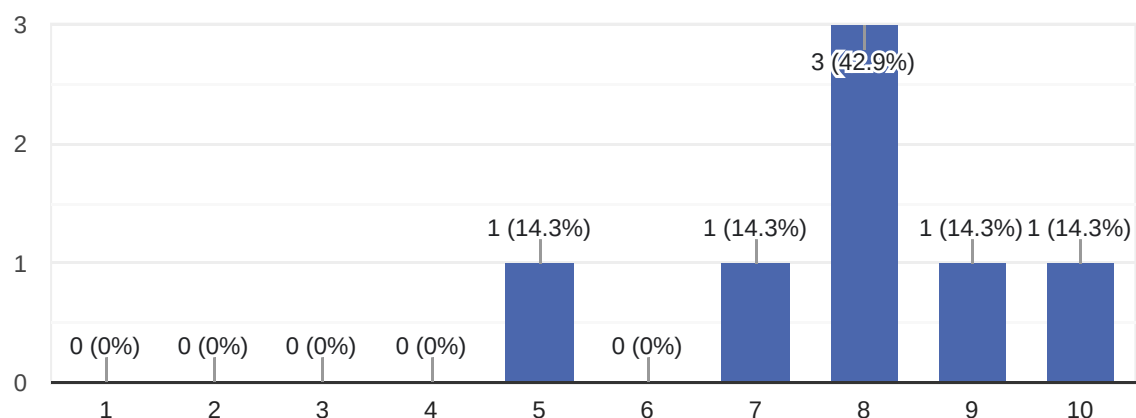
Overall this course is an excellent intro to the Linux Kernel and how device drivers are structured and implemented. I highly recommend it. It is well structured and mostly well paced. The first half of the course introduces Linux Kernel and device driver model. We get hands-on practice writing drivers for Wii Nunchuk and UART controller. I was able to follow and reproduce almost all labs (except for DMA lab). The second half focuses on other important aspects of writing a driver like concurrency, DMA, debugging.

My main criticism is that we did not spend enough time on the Power Management, DMA and mmap sections. DMA section is quite hard to follow, and the lab is somewhat difficult to implement. I also wish we would spend more time on the debugging section as it was one of the most useful parts of this training. I also wish we'd spend some time explaining magic initialization sequence for the Nunchuk and UART drivers.

How useful were the lectures?

 [Copy chart](#)

7 responses



Comments and suggestions

3 responses

The lectures were very useful. However, I sometimes had the impression that we had too many slides with not important info. I would prefer to have a more concise overview.

Suggestion: personally I would like to split up the slide decks into two pieces. One used for presenting, and one for reference/studying. Right now the slide decks are very busy. I constantly need to choose: "do I listen to the teacher and ignore the slides, or do I read the slides and ignore the teacher?" The brain can't do both. I prefer slides with diagrams / clarity over slides with lots of dense information.

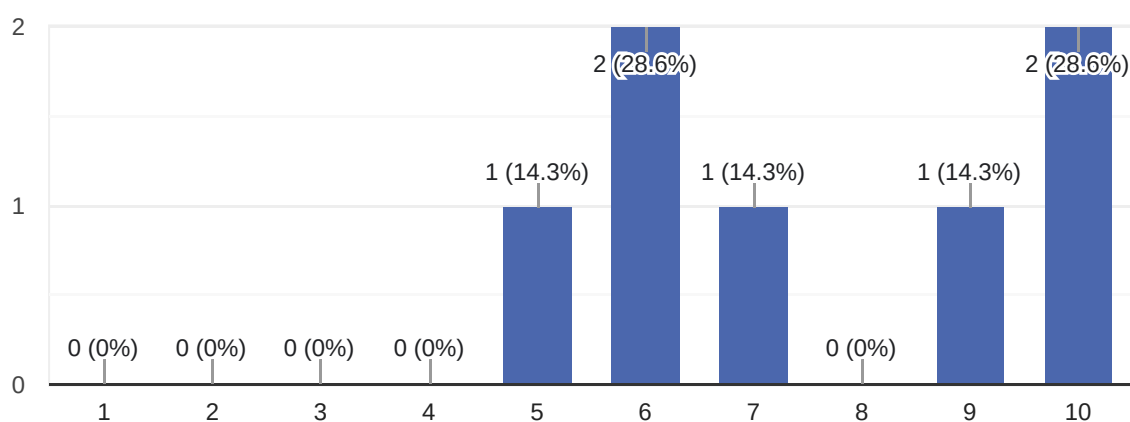
Lectures are well structured and naturally flow one after another. The first half day could be shortened a bit (intro to Linux kernel), and the last 2 sections (DMA, debugging) could take more time.

I wish there would be a section on how to write automated tests for drivers as this aspect was not covered in the lectures at all. Adding a section for writing kernel unit tests would be extremely valuable.

How useful were the practical demos?

 [Copy chart](#)

7 responses



Comments and suggestions

5 responses

Could've been done faster when the code was just pasted in instead of written per hand, but at least seeing errors and troubleshooting step was quite helpful

Better can use raspberry-pi, which uses arm64 and more up-to-date

The practical courses were useful to see how stuff is implemented. Though, following someone's code for 2 sessions a day is difficult to focus on. I would prefer to have shorter sessions over weeks where participants would need try to implement the test and then discussion the solution the next week.

It is definitely useful, but I disliked that students can choose between 3 options of development boards (beaglebone, beagleplay, imx93). I used a different development board than the teacher, which makes it less interesting to look at the demo's because things work differently for my board. This also makes it less easy to discuss findings with students.

Labs were essential in making sense out of lectures. I was able to implement all labs on my own except for the DMA lab. I do wish that the lab instructions would contain links for explaining some magic sequences (like for UART controller initialization on page 40 and long DMA priming section).

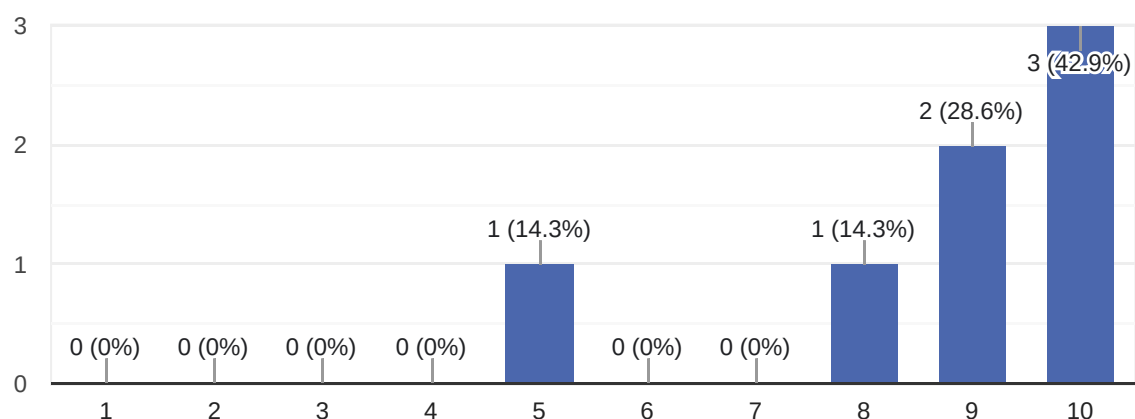
I would also suggest including kernel config options for each section as it might take 30 minutes to compile kernel only to recompile it again because one flag was missed. I also wish we'd have a lab on implementing mmap callback and actually seeing power management in action.

I also suggest adding an instruction to disable CONFIG_USB_ETH_RNDIS when setting up NFS. I had it enabled and the kernel was assigning random MAC address on every boot. Removing this config makes the lab setup stable.

How would you rate the overall organization of the course?

 [Copy chart](#)

7 responses



Comments and suggestions

2 responses

It's great that it's split up in half days with 1 day break as well. This makes it possible to do the labs in between. One suggestion is that I would prefer the time of 13.00 - 17.00 instead of 14.00 - 18.00.

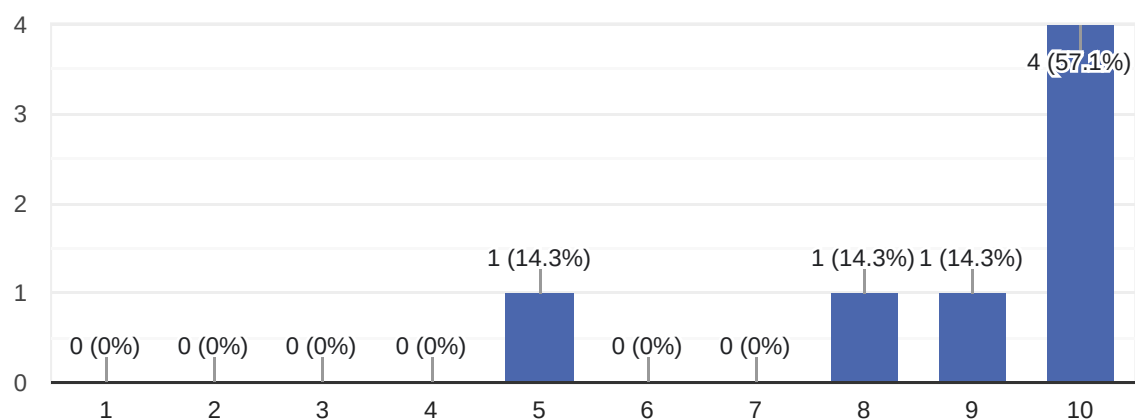
The communication was great before, during and after the course. I appreciate that we had Wednesday off because my time zone is not well suited for these courses, and it gave me a chance to get rest between lectures and time to implement the labs. I also appreciate that we had 10-15 minute breaks - very useful.

I wish we would have spent one extra day to focus on DMA and debugging sections as those did not click well. I had hard time following and wish we'd go slower.

How would you rate the trainer?

 [Copy chart](#)

7 responses



Comments and suggestions

4 responses

Very good and professional!

He was a very nice person and very competent

No comments. Very happy with the trainer.

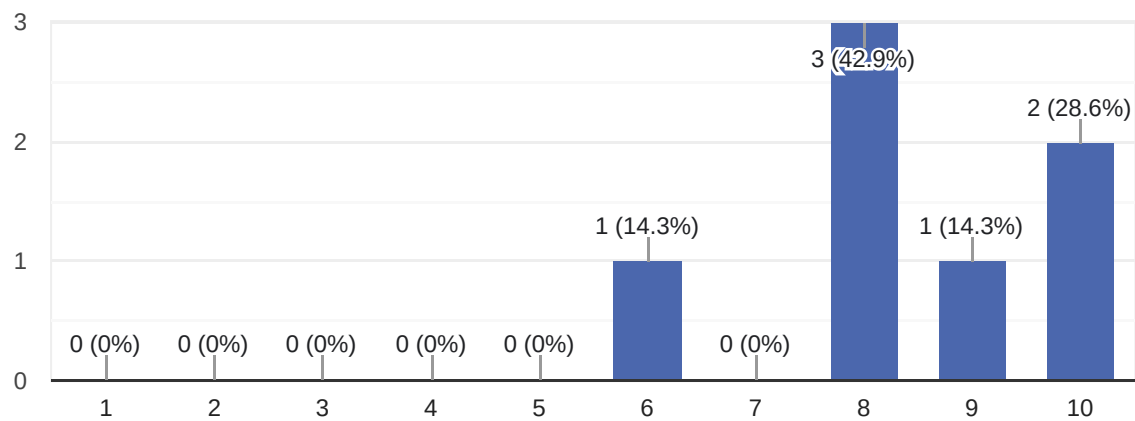
I highly recommend Gregory as the course trainer. He has a lot of experience in the field. He encouraged us to ask questions and always provided the answer. He demonstrated all practical labs and provided solutions and feedback. He communicated clearly during sections, and he was easy to follow.

My only criticism was on the pace on the last day - I wish we'd take one extra session to go slower on the DMA and debugging sections.

How did the course meet your learning objectives?

 [Copy chart](#)

7 responses



Comments and suggestions

3 responses

Gave a good intro to kernel development, really helpful to gain confidence with navigating the source code and having a map of the general concepts within the kernel

As said before: shorter sessions distributed over more days/weeks; participants being able to implement the learning by themselves (enough time between sessions); more concise slides and material

The course met most of my learning goals. I have good understanding of how device drivers are structured and where to look for help writing one.

What part(s) of the course did you like most?

6 responses

Going deep into the kernel source code

Labs were helpful to see how experience dev troubleshoots issues while writing modules

Device tree outline, driver frameworks, driver model summary

The trainer

1. The course covered many important Linux kernel and device driver concepts.
2. The hands on labs were useful because they showed how the concepts are applied in practice.
3. The trainer demonstrated real development workflows, which was valuable.

Besides the discrepancy between teacher's board and student boards, I liked doing the practical labs the most.

The course structure, breadth of the cover, hands on experience writing Nunchuk and UART drivers, debugging lab

What part(s) of the course did you like least?

5 responses

Labs could've been faster

The timeslot and the length per session, maybe it was attributed to the hot temperatures outside.

1. I found the trainer's French accent difficult to understand at times. This made it challenging to follow some of the explanations, especially during the faster parts of the lectures.
2. The course contained a very large number of slides and a lot of theory. In some sections it felt information dense, making it difficult to absorb everything during the available time.
3. The practical demonstrations sometimes moved too quickly. It felt as though the trainer was demonstrating steps from memory, so there was not always enough explanation of why each step was being performed or what could go wrong.
4. The online format made it harder to stay engaged and ask questions. I believe this course would be more effective if delivered in person, or with more interaction during the online sessions.

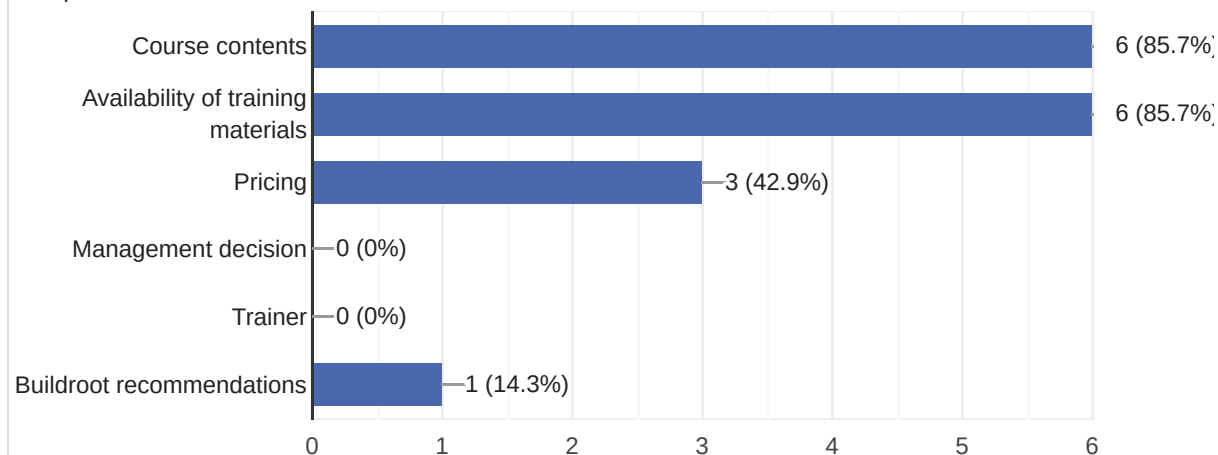
I think discussing slides that explain function prototypes doesn't add a lot of value. I'd prefer to look at implementation architecture or flow charts.

DMA part is hard to follow, Debugging part is too short.

What reasons prompted you to choose a Bootlin course?

 [Copy chart](#)

7 responses



Comments

1 response

I would've liked a diagram which shows how the components implemented during the labs play together and/or how the difference driver structs relate to each other. More diagrams in general would've been nice

Further training needs?

2 responses

Maybe adding training material for newer concepts like eBPF and netlink would be cool

Discoverable devices (PCIe and USB)

Core Kernel basics (Scheduler, Memory, Storage, Concurrency etc)

This content is neither created nor endorsed by Google. - [Terms of Service](#) - [Privacy Policy](#).

Does this form look suspicious? [Report](#)

Google Forms

