"Online Training on Linux SPI & UART Drivers" by **Pradeep Tewani**

+ ***Session 1: BBB Setup & Building the Kernel***
- Readying BBB for SPI & UART drivers
- Setting up the host environment
- Patching & building the kernel

+ ***Session 2: Introduction to SPI Driver***
- SPI Protocol overview
- Understanding the SPI registers for target platform
- Writing a framework independent SPI controller driver

+ ***Session 3: Linux SPI Framework***
- SPI framework components
- Registering SPI client & Master driver
- Understanding the probing mechanism
- Adding device specific nodes in the device tree

+ ***Session 4: SPI Driver with Linux DMA Engine***
- Understanding the Linux DMA Engine
- Enhancing SPI driver to use DMA

+ ***Session 5: Platform specific UART Driver***
- UART Protocol overview
- Understanding the UART registers for the platform
- Writing a framework independent low level UART driver

+ ***Session 6: Linux TTY Framework***
- Understanding the Linux TTY Framework
- Understanding the data flow for TTY Framework
- Writing a dummy UART driver

+ ***Session 7: Integrating the UART Driver with TTY Framework***
- Adding the necessary nodes in the device tree
- Registering the required callback handlers

+ ***Session 8: Wrap Up***
- Conclusion
- What Next?

Caution: All sessions are highly interactive & hands-on with Beagle Bone Black.

Hands-On Details

+ ***Session 1: BBB Setup & Building the Kernel***
  - Building the Linux kernel
  - BBB Setup & booting up the board

+ ***Session 2: Introduction to SPI Driver***
  - Writing a low level SPI driver with loopback

+ ***Session 3: Linux SPI framework***
  - Writing a dummy SPI client and dummy Master
  - Integrating the low level driver with SPI framework

+ ***Session 4: SPI Driver with Linux DMA Engine***
  - Enhance the driver to add the support for DMA

+ ***Session 5: Platform specific UART Driver***
  - Writing a low level UART driver

+ ***Session 6: Linux TTY Framework***
  - Writing a dummy UART driver

+ ***Session 7: Integrating the UART Driver with TTY Framework***
  - Enhancing the low level driver to use TTY Framework