Embedded Linux Porting (C3)

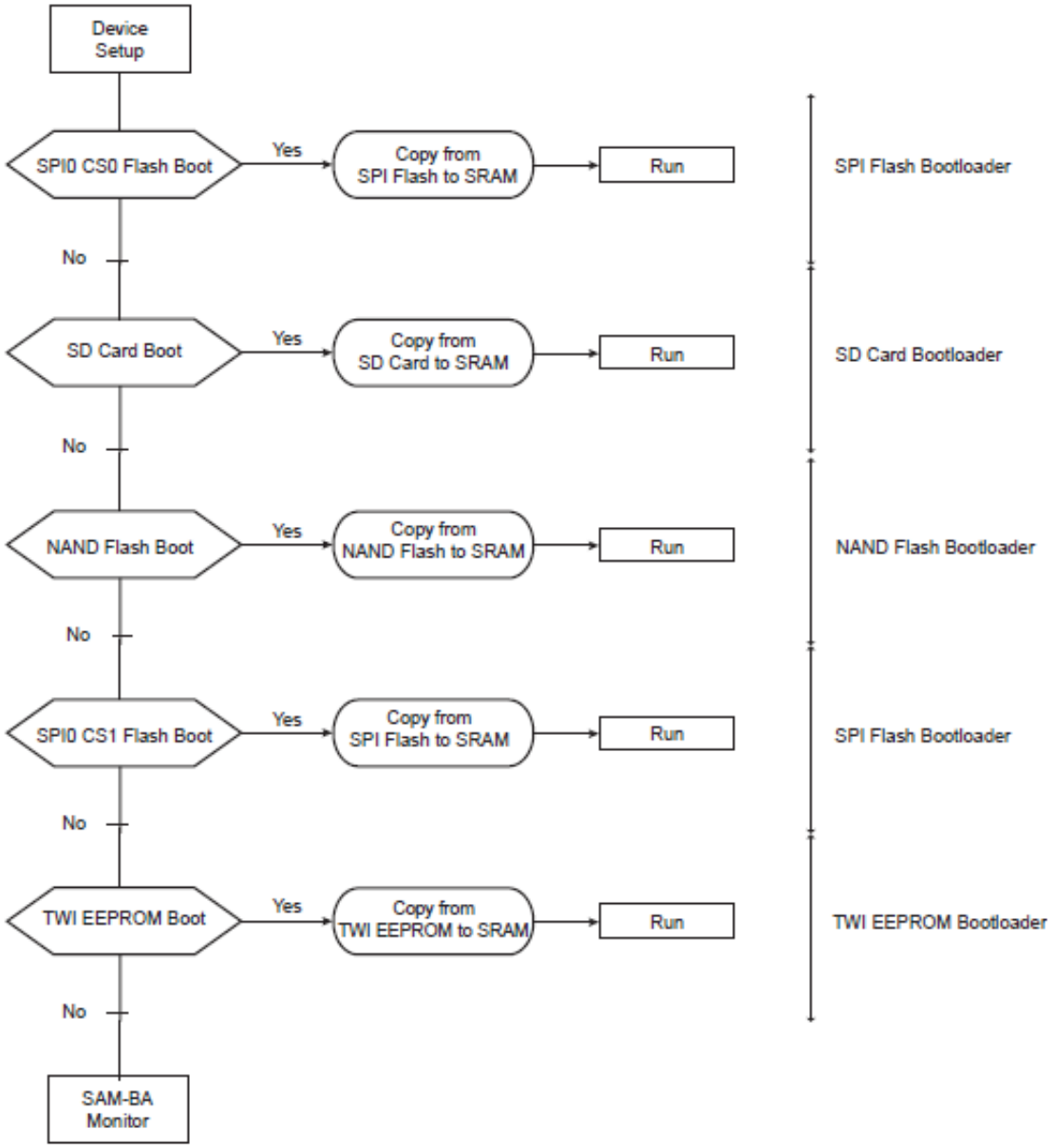
Primary Bootloader

(Atmel Bootstrap)

Agenda

- Introduction to Bootloader & PBL
- Atmel Bootstrap directory structure
- Atmel Bootstrap code flow
- Compile and Testing Atmel Bootstrap code on RB-A5D2x
- How to add new Board to the Atmel Bootstrap

ROM Code boot Sequence



Bootloader

- Why Bootloader
- Functions of Bootloader
- Why Primary Bootloader required

Bootstrap Dir Structure

```
phytec:at91bootstrap-rba5d2x$ ls
binaries              Config.in.none    driver                  KNOWN_ISSUES  README.txt
board                 Config.in.secure  elf32-littlearm.lds     lib           scripts
config                Config.in.u-boot  elf32-littlearm-tz.lds  main.c        tags
Config.in             contrib           fs                      main.o        toplevel_cpp.mk
Config.in.app-image   crt0_gnu.o        host-utilities          Makefile      uEnv.txt
Config.in.kernel      crt0_gnu.S        include                 README.md
phytec:at91bootstrap-rba5d2x$
```

bootstrap_code_flow.pdf

Bootstrap Compilation & Testing

 # Make sure you have set the toolchain path & environment

 # Clean the source dir

$ make mrproper

 # Configure the Atmel Bootstrap for Ruggedboard-A5D2x

$ make rugged_board_a5d2x_defconfig

 # Do addition Configuration if required using menuconfig

$ make menuconfig

 # Compile the source

$ make

Bootstrap Porting on New Board

- Create a new directory to hold your board specific code
  board/ directory.

- Add Files in board/<board_name>
  "board.mk", a ".c", ".h", "Config.in.board", and "Config.in.boardname“

- Create the necessary default configuration files such as “<board_name__defconfig" in your new board directory.

- Add(source) your board's "Config.in.board" in "board/Config.in.board" file.

- Add(source) your board's "Config.in.boardname" in the "board/Config.in.boardname" file.

- Add your board's ".h" in the "include/contrib_board.h" file.

Open Discussions

Connect-us