Vision SDK Linux
Cascade Radar Data Acquisition Tool

User Guide

Information in this document is subject to change without notice. Texas Instruments may have pending patent applications, trademarks, copyrights, or other intellectual property rights covering matter in this document. The furnishing of this documents is given for usage with Texas Instruments products only and does not give you any license to the intellectual property that might be contained within this document. Texas Instruments makes no implied or expressed warranties in this document and is not responsible for the products based from this document.

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgment, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Customers are responsible for their applications using TI components.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards ought to be provided by the customer so as to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, license, warranty or endorsement thereof.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations and notices. Representation or reproduction of this information with alteration voids all warranties provided for an associated TI product or service, is an unfair and deceptive business practice, and TI is neither responsible nor liable for any such use.

Resale of TI's products or services with *statements different from or beyond the parameters* stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service, is an unfair and deceptive business practice, and TI is not responsible nor liable for any such use.

Also see: Standard Terms and Conditions of Sale for Semiconductor Products.
www.ti.com/sc/docs/stdterms.htm

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265

TABLE OF CONTENTS

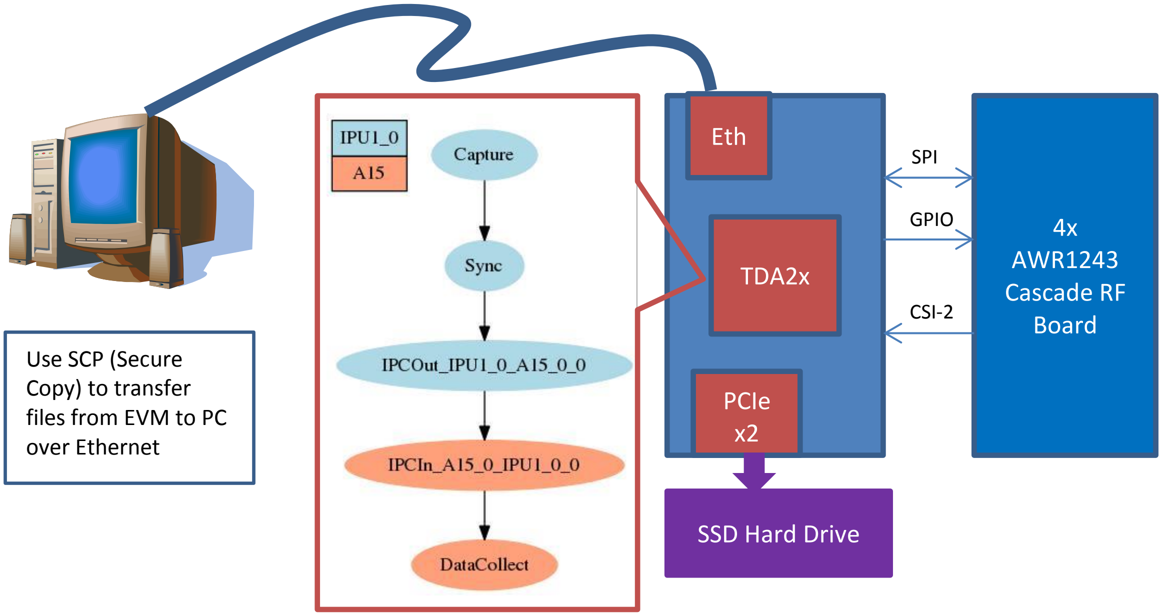
1 Introduction

This user guide provides details on how to build and utilize the data acquisition tool to capture raw data (ADC samples) from 4x AWR1243 Cascade Radar RF board with TDA2x Digital board.

For 4x AWR1243 Cascade Radar, depending on the Radar configuration, the raw data throughput could be up to ~6 Gbps. To save the raw data in real-time to a storage device, it requires an interface with very high BW capability. On TDA2x, PCIe interface can provide maximum 5Gbps BW per lane with total of 2 lanes. Using 2-lane PCIe interface on TDA2x to connect to a SSD storage device, it is sufficient to achieve real-time data acquisition for 4-chip cascade radar use case.

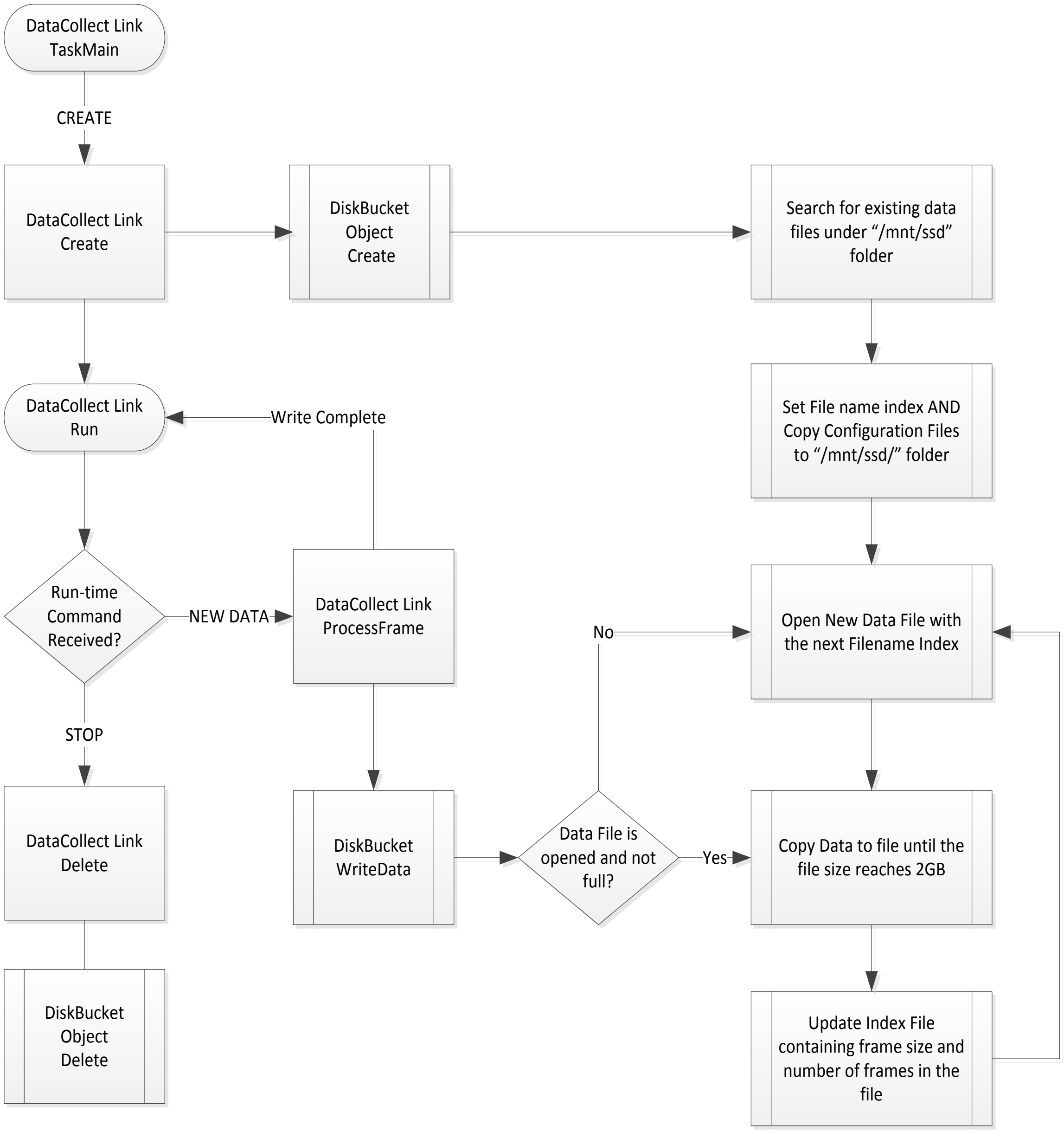
1.1 Block Diagram

The data acquisition tool implements the data scheme as shown in the following block diagram.



1.2 Program Flow

The main component of Data acquisition Tool is DataCollect Link running on A15 in TDA2x. DataCollect Link implemented the program flow as shown in the following figure.


DataCollect Link TaskMain
CREATE
DataCollect Link Create
DiskBucket Object Create
Search for existing data files under "/mnt/ssd" folder
Set File name index AND Copy Configuration Files to "/mnt/ssd/" folder
DataCollect Link Run
Write Complete
Run-time Command Received?
NEW DATA
DataCollect Link ProcessFrame
No
Open New Data File with the next Filename Index
STOP
DataCollect Link Delete
DiskBucket WriteData
Data File is opened and not full?
Yes
Copy Data to file until the file size reaches 2GB
DiskBucket Object Delete
Update Index File containing frame size and number of frames in the file

1.3 Data Output Format

Data Acquisition Tool will save 3 types of files to the SSD card.

- Radar Configuration Files

  Radar Configuration files used for the acquiring data set will be copied to SSD as "bxxxx_cfg_00.txt" ~ "bxxx_cfg_03.txt".

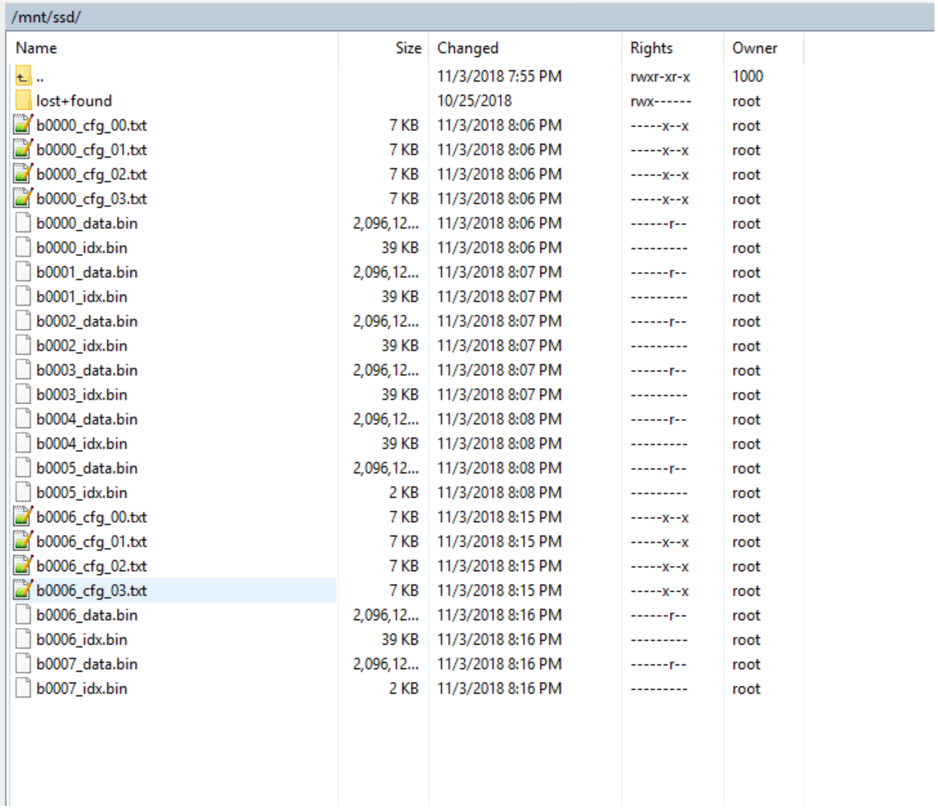
- Data Files

  Data will be saved to a new file every 2GB as "bxxxx_data.bin".

- Index Files

  Each data file has an index file, "bxxxx_idx.bin" to provide the frame size and numbers of total frames stored in the file.

The following screenshot shows an example of the files saved to SSD by Data Acquisition Tool.



/mnt/ssd/

Name	Size	Changed	Rights	Owner
..		11/3/2018 7:55 PM	rwxr-xr-x	1000
lost+found		10/25/2018	rwx------	root
b0000_cfg_00.txt	7 KB	11/3/2018 8:06 PM	-----x--x	root
b0000_cfg_01.txt	7 KB	11/3/2018 8:06 PM	-----x--x	root
b0000_cfg_02.txt	7 KB	11/3/2018 8:06 PM	-----x--x	root
b0000_cfg_03.txt	7 KB	11/3/2018 8:06 PM	-----x--x	root
b0000_data.bin	2,096,12...	11/3/2018 8:06 PM	------r--	root
b0000_idx.bin	39 KB	11/3/2018 8:06 PM	---------	root
b0001_data.bin	2,096,12...	11/3/2018 8:07 PM	------r--	root
b0001_idx.bin	39 KB	11/3/2018 8:07 PM	---------	root
b0002_data.bin	2,096,12...	11/3/2018 8:07 PM	------r--	root
b0002_idx.bin	39 KB	11/3/2018 8:07 PM	---------	root
b0003_data.bin	2,096,12...	11/3/2018 8:07 PM	------r--	root
b0003_idx.bin	39 KB	11/3/2018 8:07 PM	---------	root
b0004_data.bin	2,096,12...	11/3/2018 8:08 PM	------r--	root
b0004_idx.bin	39 KB	11/3/2018 8:08 PM	---------	root
b0005_data.bin	2,096,12...	11/3/2018 8:08 PM	------r--	root
b0005_idx.bin	2 KB	11/3/2018 8:08 PM	---------	root
b0006_cfg_00.txt	7 KB	11/3/2018 8:15 PM	-----x--x	root
b0006_cfg_01.txt	7 KB	11/3/2018 8:15 PM	-----x--x	root
b0006_cfg_02.txt	7 KB	11/3/2018 8:15 PM	-----x--x	root
b0006_cfg_03.txt	7 KB	11/3/2018 8:15 PM	-----x--x	root
b0006_data.bin	2,096,12...	11/3/2018 8:16 PM	------r--	root
b0006_idx.bin	39 KB	11/3/2018 8:16 PM	---------	root
b0007_data.bin	2,096,12...	11/3/2018 8:16 PM	------r--	root
b0007_idx.bin	2 KB	11/3/2018 8:16 PM	---------	root

2 Requirements for Running Data Acquisition Tool

It is important to make sure the required components are available before trying to use the data acquisition tool.

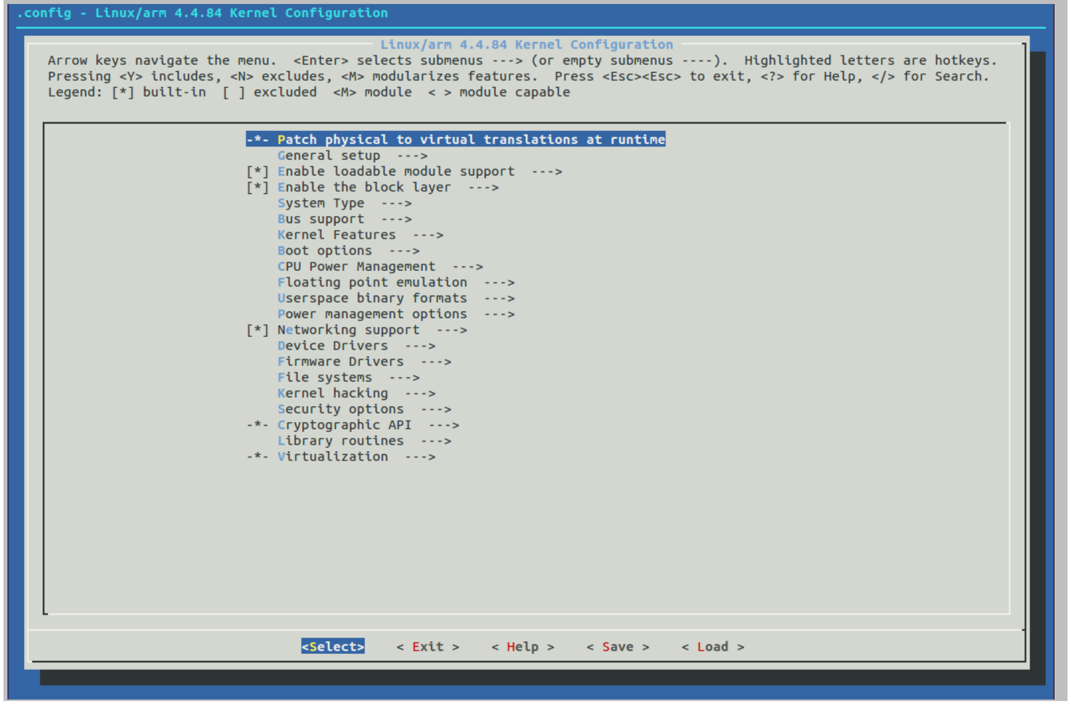
- Hardware Requirements:
  - 4-Chips AWR1243 Cascade Radar Board and TDA2xx Processing board
  - Nvme M.2 PCIe SSD card (ex. Link)
  - Windows or Ubuntu PC
- Processor SDK Vision 3.6 release or later (Linux version)
- Please Refer to VisionSDK_Linux_UserGuide.pdf for SDK requirements

3 Build Linux Vision SDK for Cascade Radar EVM

1. Follow the instructions in the Chapter 2 of VisionSDK_Linux_UserGuide.pdf to install Linux Vision SDK and set up the build environment.
2. Apply u-boot patch. Patch can be found under the following location.

   [SDK_INSTALL]/ti_components/os_tools/linux/kernel/linux-kernel-addon/u-boot-patches/
3. Follow the build instructions in Chapter 3 of the SDK User Guide. Set MAKECONFIG=tda2xx_cascade_linux_radar in Rules.make.
4. If the build is completed successfully, proceed to the next step to rebuild kernel with NVM Express block driver enabled. If not, please review the build steps in SDK User Guide carefully and try to troubleshoot the build issue.
5. To enable NVM Express block driver in Linux kernel, use the following build command.

```
$> cd $INSTALL_DIR/vision_sdk/build
$> make kernel_menuconfig
```

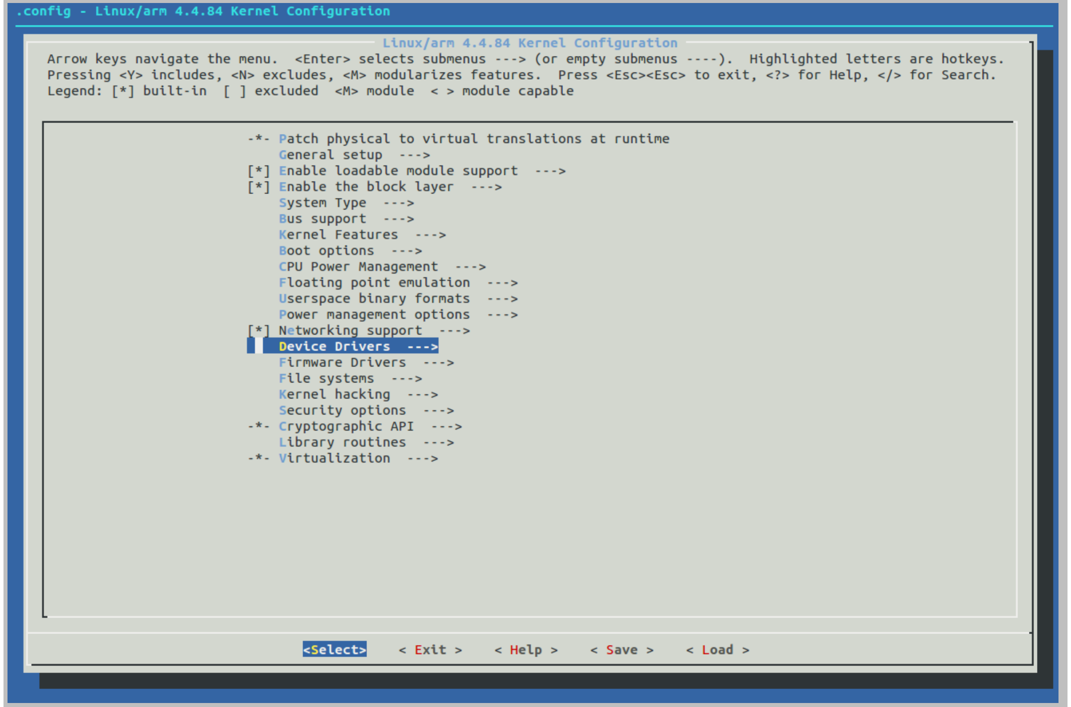


```
.config - Linux/arm 4.4.84 Kernel Configuration
                         Linux/arm 4.4.84 Kernel Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                    -*- Patch physical to virtual translations at runtime
                        General setup  --->
                    [*] Enable loadable module support  --->
                    [*] Enable the block layer  --->
                        System Type  --->
                        Bus support  --->
                        Kernel Features  --->
                        Boot options  --->
                        CPU Power Management  --->
                        Floating point emulation  --->
                        Userspace binary formats  --->
                        Power management options  --->
                    [*] Networking support  --->
                        Device Drivers  --->
                        Firmware Drivers  --->
                        File systems  --->
                        Kernel hacking  --->
                        Security options  --->
                    -*- Cryptographic API  --->
                        Library routines  --->
                    -*- Virtualization  --->

                 <Select>    < Exit >    < Help >    < Save >    < Load >
```

➔ Select “Device Drivers”

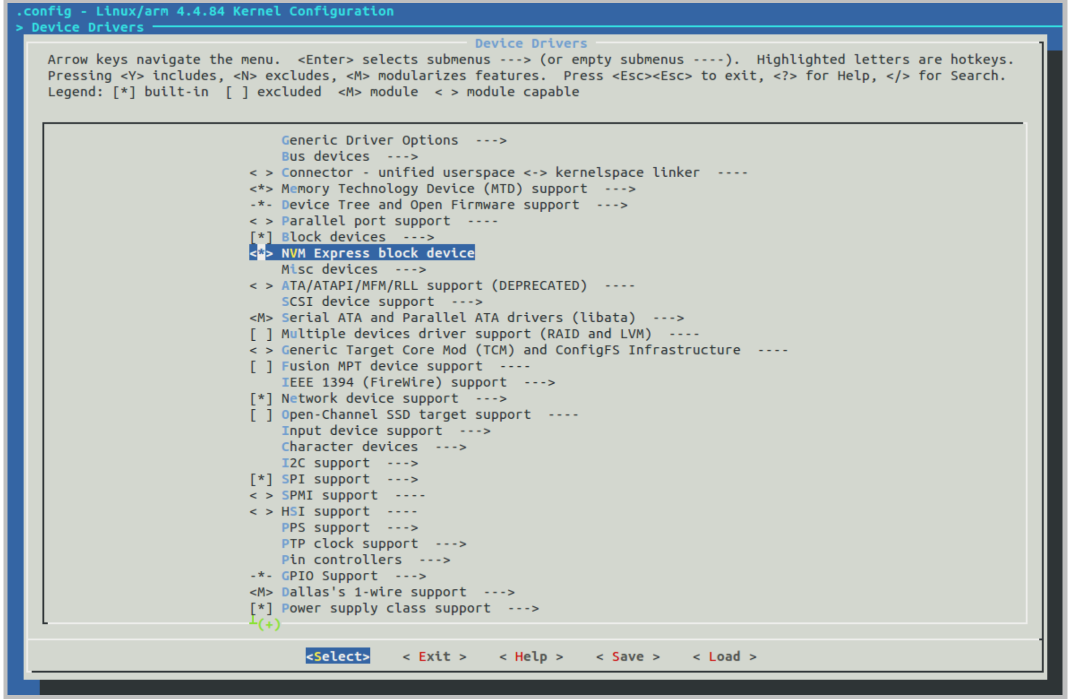


```
.config - Linux/arm 4.4.84 Kernel Configuration
                    Linux/arm 4.4.84 Kernel Configuration
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                          -*- Patch physical to virtual translations at runtime
                              General setup  --->
                          [*] Enable loadable module support  --->
                          [*] Enable the block layer  --->
                              System Type  --->
                              Bus support  --->
                              Kernel Features  --->
                              Boot options  --->
                              CPU Power Management  --->
                              Floating point emulation  --->
                              Userspace binary formats  --->
                              Power management options  --->
                          [*] Networking support  --->
                              Device Drivers  --->
                              Firmware Drivers  --->
                              File systems  --->
                              Kernel hacking  --->
                              Security options  --->
                          -*- Cryptographic API  --->
                              Library routines  --->
                          -*- Virtualization  --->

                  <Select>    < Exit >    < Help >    < Save >    < Load >
```

➔ Select “NVM Express block device” as [*] built-in



```
.config - Linux/arm 4.4.84 Kernel Configuration
> Device Drivers
                    Device Drivers
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
  Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                          Generic Driver Options  --->
                          Bus devices  --->
                      < > Connector - unified userspace <-> kernelspace linker  ----
                      <*> Memory Technology Device (MTD) support  --->
                      -*- Device Tree and Open Firmware support  --->
                      < > Parallel port support  ----
                      [*] Block devices  --->
                      <*> NVM Express block device
                          Misc devices  --->
                      < > ATA/ATAPI/MFM/RLL support (DEPRECATED)  ----
                          SCSI device support  --->
                      <M> Serial ATA and Parallel ATA drivers (libata)  --->
                      [ ] Multiple devices driver support (RAID and LVM)  ----
                      < > Generic Target Core Mod (TCM) and ConfigFS Infrastructure  ----
                      [ ] Fusion MPT device support  ----
                          IEEE 1394 (FireWire) support  --->
                      [*] Network device support  --->
                      [ ] Open-Channel SSD target support  ----
                          Input device support  --->
                          Character devices  --->
                          I2C support  --->
                      [*] SPI support  --->
                      < > SPMI support  ----
                      < > HSI support  ----
                          PPS support  --->
                          PTP clock support  --->
                          Pin controllers  --->
                      -*- GPIO Support  --->
                      <M> Dallas's 1-wire support  --->
                      [*] Power supply class support  --->
                      (+)

                  <Select>    < Exit >    < Help >    < Save >    < Load >
```

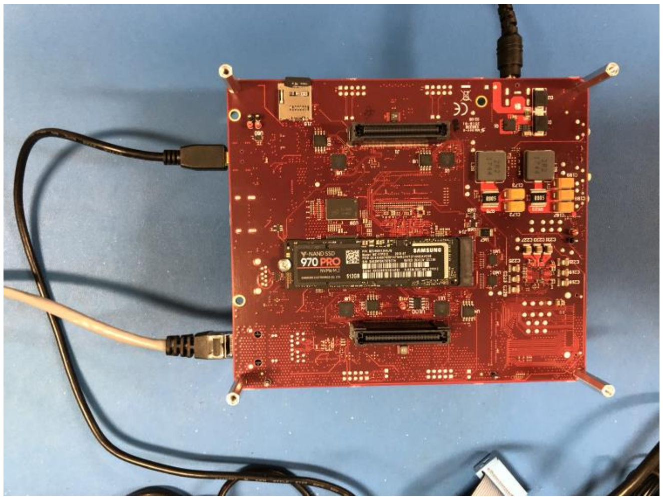
➔ Save & Exit the menu, Kernel will be built.

```
$> make linux_install
```

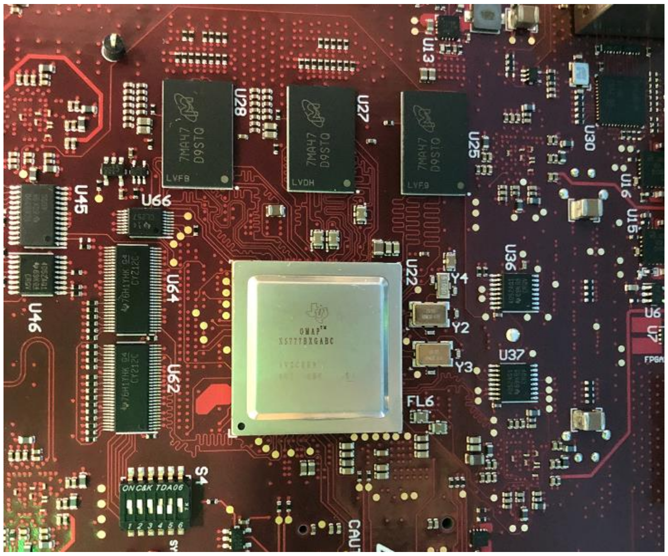
6. Follow the instructions in SDK User guide to prepare the SD card.
7. Place the radar configuration files under "/opt/vision" folder in file system on SD card. The example configuration files can be found under ~\vision_sdk\apps\src\rtos\radar\src\usecases\cascade_radar_object_detect\radar_test_vector. Please refer to ProcessorSDKRadar_UserGuide.pdf for the details regarding the configuration files.

4 Data Acquisition Tool

Before running data acquisition tool, the NVMe M.2 SSD card should be installed on the EVM as shown in the following figure. Remove the Cascade RF board first to have access to the SSD card slot on the TDA2x processing board. After installing SSD card, secure the RF board back to the connectors on the TDA2x processing board.



Before powering up the board, make sure the S4 Dip Switch, as shown in the following figure, is set to [1:6] = [111001] to boot from SD card.



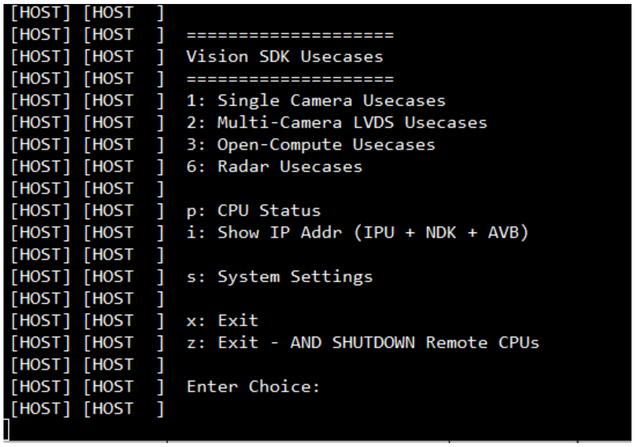
After powering up the board and booting to Linux kernel prompt, please make sure the SSD card is detected. If the SSD is detected correctly, the device will be “/dev/nvme0”.

4.1 Run Data Acquisition Tool

1. Format the partition on SSD card as needed. Use “fdisk” and “mkfs” tools. Please refer to the manual of “fdisk” and “mkfs” for details.
2. Create “/mnt/ssd” folder and mount the partition on SSD for data storage to this folder.
3. Start Vision SDK Demo Application.

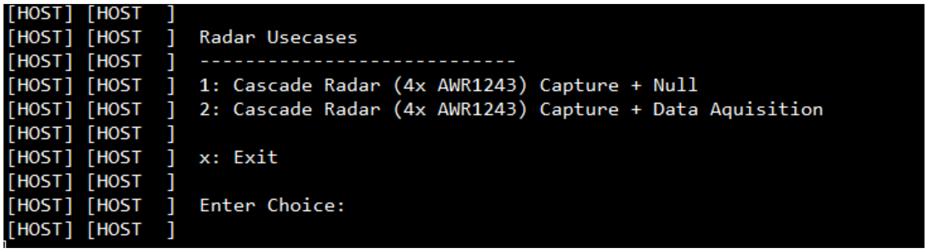
```
$> cd /opt/vision_sdk
$> source ./vision_sdk_load.sh
$> ./apps.out
```

4. Select ‘6’ for Radar Usecases.



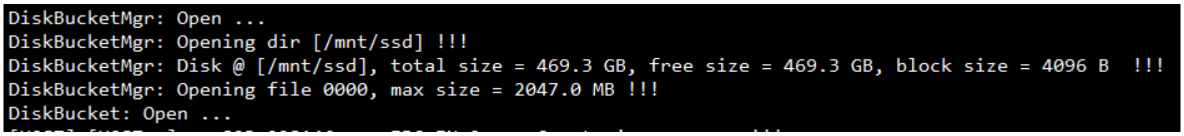
```
[HOST] [HOST  ]
[HOST] [HOST  ]  ====================
[HOST] [HOST  ]  Vision SDK Usecases
[HOST] [HOST  ]  ====================
[HOST] [HOST  ]  1: Single Camera Usecases
[HOST] [HOST  ]  2: Multi-Camera LVDS Usecases
[HOST] [HOST  ]  3: Open-Compute Usecases
[HOST] [HOST  ]  6: Radar Usecases
[HOST] [HOST  ]
[HOST] [HOST  ]  p: CPU Status
[HOST] [HOST  ]  i: Show IP Addr (IPU + NDK + AVB)
[HOST] [HOST  ]
[HOST] [HOST  ]  s: System Settings
[HOST] [HOST  ]
[HOST] [HOST  ]  x: Exit
[HOST] [HOST  ]  z: Exit - AND SHUTDOWN Remote CPUs
[HOST] [HOST  ]
[HOST] [HOST  ]  Enter Choice:
[HOST] [HOST  ]
```

5. Select ‘2’ for Cascade Radar Capture + Data Acquisition.

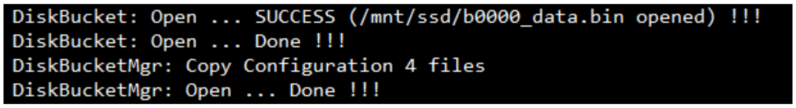


```
[HOST] [HOST  ]
[HOST] [HOST  ]  Radar Usecases
[HOST] [HOST  ]  ----------------------------
[HOST] [HOST  ]  1: Cascade Radar (4x AWR1243) Capture + Null
[HOST] [HOST  ]  2: Cascade Radar (4x AWR1243) Capture + Data Aquisition
[HOST] [HOST  ]
[HOST] [HOST  ]  x: Exit
[HOST] [HOST  ]
[HOST] [HOST  ]  Enter Choice:
[HOST] [HOST  ]
```

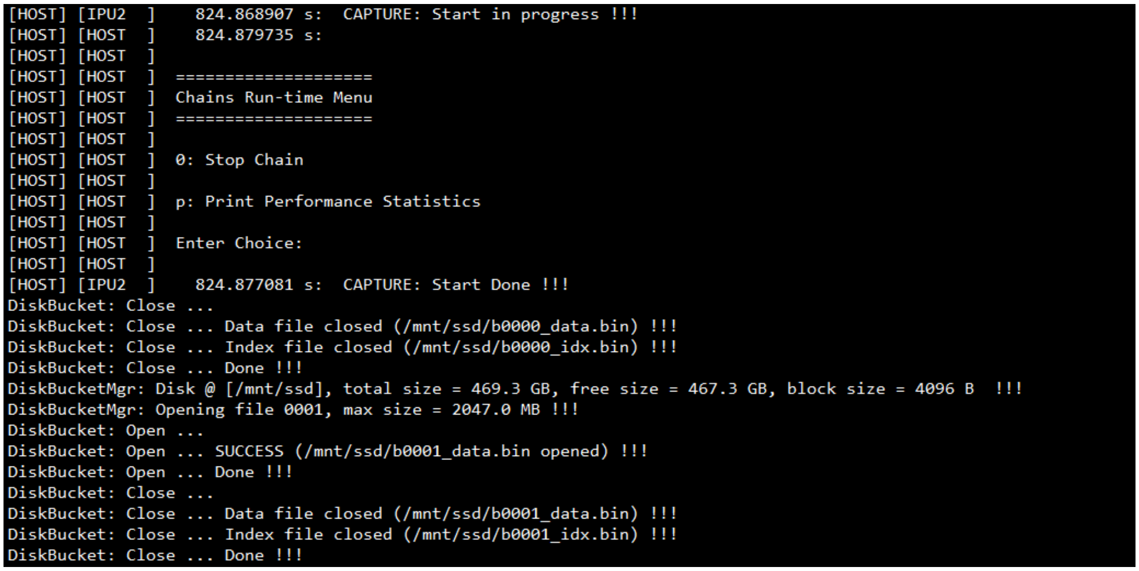
6. Once the use case starts, the tool will search the existing files under /mnt/ssd folder. If there are already data files from prior runs, it will create the new file with incrementing index for file name.



```
DiskBucketMgr: Open ...
DiskBucketMgr: Opening dir [/mnt/ssd] !!!
DiskBucketMgr: Disk @ [/mnt/ssd], total size = 469.3 GB, free size = 469.3 GB, block size = 4096 B  !!!
DiskBucketMgr: Opening file 0000, max size = 2047.0 MB !!!
DiskBucket: Open ...
```



```
DiskBucket: Open ... SUCCESS (/mnt/ssd/b0000_data.bin opened) !!!
DiskBucket: Open ... Done !!!
DiskBucketMgr: Copy Configuration 4 files
DiskBucketMgr: Open ... Done !!!
```



```
[HOST] [IPU2  ]    824.868907 s:  CAPTURE: Start in progress !!!
[HOST] [HOST  ]    824.879735 s:
[HOST] [HOST  ]
[HOST] [HOST  ]  ====================
[HOST] [HOST  ]  Chains Run-time Menu
[HOST] [HOST  ]  ====================
[HOST] [HOST  ]
[HOST] [HOST  ]  0: Stop Chain
[HOST] [HOST  ]
[HOST] [HOST  ]  p: Print Performance Statistics
[HOST] [HOST  ]
[HOST] [HOST  ]  Enter Choice:
[HOST] [HOST  ]
[HOST] [IPU2  ]    824.877081 s:  CAPTURE: Start Done !!!
DiskBucket: Close ...
DiskBucket: Close ... Data file closed (/mnt/ssd/b0000_data.bin) !!!
DiskBucket: Close ... Index file closed (/mnt/ssd/b0000_idx.bin) !!!
DiskBucket: Close ... Done !!!
DiskBucketMgr: Disk @ [/mnt/ssd], total size = 469.3 GB, free size = 467.3 GB, block size = 4096 B  !!!
DiskBucketMgr: Opening file 0001, max size = 2047.0 MB !!!
DiskBucket: Open ...
DiskBucket: Open ... SUCCESS (/mnt/ssd/b0001_data.bin opened) !!!
DiskBucket: Open ... Done !!!
DiskBucket: Close ...
DiskBucket: Close ... Data file closed (/mnt/ssd/b0001_data.bin) !!!
DiskBucket: Close ... Index file closed (/mnt/ssd/b0001_idx.bin) !!!
DiskBucket: Close ... Done !!!
```

4.2 Retrieve Data

3 ways to retrieve the radar data collected in SSD card as the followings.

- Use Linux PC (Ubuntu) – Preferred

  From EVM prompt, run "ifconfig" to obtain EVM IP address. Use SCP to transfer file to PC.

- Use Windows PC

  Download "WinSCP" for Windows and use it to transfer file from EVM to PC.

- Remove SSD card from EVM

  Use M.2 SSD-to-USB adapter/enclosure to read SSD card on PC.

5 Revision History

Version	Date	Revision History
1.0	Jan 3, 2019	First version

««« § »»»