

# Test Procedure

---





## Table of Contents

1. Initial test set up .....	1
2. TST001 Verify whether board is booting properly after powered up .....	2
2.1. Procedure .....	2
2.2. Expected result .....	2
3. TST002 Verify whether board is rebooting properly after Reset Button pressed .....	3
3.1. Procedure .....	3
3.2. Expected result .....	3
4. TST003 Verify the consistency of booting when power cycle a Device multiple times .....	4
4.1. Procedure .....	4
4.2. Expected result .....	4
5. TST004 Check release version .....	5
5.1. Procedure .....	5
5.2. Expected result .....	5
6. TST005 Check the U-boot software version .....	6
6.1. Procedure .....	6
6.2. Expected result .....	6
7. TST006 Check the Kernel version .....	7
7.1. procedure .....	7
7.2. Expected result .....	7
8. TST007 Verify the total RAM size .....	8
8.1. Procedure .....	8
8.2. Expected result .....	8
9. TST008 Verify the available RAM space size .....	10
9.1. Procedure .....	10
9.2. Expected result .....	10
10. TST009 Verify the total Storage size .....	12
10.1. Procedure .....	12
10.2. Expected result .....	12
11. TST010 Verify the Available Storage size .....	13
11.1. Procedure .....	13
11.2. Expected result .....	13
12. TST011 Verify the SOC Name and number of core .....	14
12.1. Procedure .....	14
12.2. Expected result .....	14
13. TST012 Verify the architecture of SOC .....	15
13.1. Procedure .....	15
13.2. Expected result .....	15
14. TST013 Verify whether Ethernet ports are detected .....	16
14.1. preconditions .....	16
14.2. Procedure .....	16
14.3. Expected result .....	16
15. TST014 Verify LAN Ethernet port is getting ip address when it is connected to LAN .....	18
15.1. preconditions .....	18
15.2. Procedure .....	18
15.3. Expected result .....	18
16. TST015 Verify the GPIO pins are populated in sysfs .....	19
16.1. Preconditions .....	19
16.2. Procedure .....	19
16.3. Expected result .....	19
17. TST016 Verify whether UART ports are Detected .....	20
17.1. Preconditions .....	20

17.2. Procedure .....	20
17.3. Expected result .....	20
18. TST017 Verify whether \$(n) UART ports are detected .....	22
18.1. Precondition .....	22
18.2. Procedure .....	22
18.3. Expected result .....	22
19. TST018 Verify whether \$(n) I <sup>2</sup> C ports are detected .....	23
19.1. Precondition .....	23
19.2. Procedure .....	23
19.3. Expected result .....	23
20. TST019 Verify Scanning of slave devices in I <sup>2</sup> C buses .....	24
20.1. Precondition .....	24
20.2. Procedure .....	24
20.3. Expected result .....	24
21. TST020 Verify whether \$(n) SPI ports are detected .....	26
21.1. Precondition .....	26
21.2. Procedure .....	26
21.3. Expected result .....	26
22. TST021 Verify whether \$(n) USB ports are detected .....	27
22.1. Precondition .....	27
22.2. Procedure .....	27
22.3. Expected result .....	27
23. TST022 Verify whether USB slave device is detected .....	28
23.1. Precondition .....	28
23.2. Procedure .....	28
23.3. Expected result .....	28
24. TST023 Verify Audio codec driver is detected .....	29
24.1. Precondition .....	29
24.2. Procedure .....	29
24.3. Expected result .....	29
25. TST024 Verify Audio ports are detected .....	30
25.1. Precondition .....	30
25.2. Procedure .....	30
25.3. Expected result .....	30
26. TST025 Verify whether Board enter into uboot prompt properly when press any key .....	31
26.1. Precondition .....	31
26.2. Procedure .....	31
26.3. Expected result .....	31
27. TST026 Check the board SOC in u-boot .....	32
27.1. Precondition .....	32
27.2. Procedure .....	32
27.3. Expected result .....	32
28. TST027 Check the CPU model in u-boot .....	33
28.1. Precondition .....	33
28.2. Procedure .....	33
28.3. Expected result .....	33
29. TST028 Check the board name in u-boot .....	34
29.1. Precondition .....	34
29.2. Procedure .....	34
29.3. Expected result .....	34
30. TST029 Check the U-boot version .....	35
30.1. Precondition .....	35
30.2. Procedure .....	35

30.3. Expected result .....	35
31. TST030 Verify whether kernel log is displayed using dmesg .....	36
31.1. Precondition .....	36
31.2. Procedure .....	36
31.3. Expected result .....	36
32. TST031 Verify whether Kernel data structure is accessible using sysfs .....	37
32.1. Precondition .....	37
32.2. Procedure .....	37
32.3. Expected result .....	37
33. TST032 Verify whether board shell is accessible .....	38
33.1. Precondition .....	38
33.2. Procedure .....	38
34. TST033 Check whether Rootfs have basic commands for testing .....	39
34.1. Precondition .....	39
34.2. Procedure .....	39
34.3. Expected result .....	39
35. TST034 Verify MMC_1 8 pins working as SD Card boot .....	40
35.1. precondition .....	40
35.2. procedure .....	40
35.3. Expected Result .....	40
36. TST035 Verify ETH_0 8 pins working as 10/100/1000M Ethernet(RGMII0) .....	41
36.1. Precondition .....	41
36.2. Procedure .....	41
36.3. Expected Result .....	41
37. TST036 verify AMBE Codec 4 MHz clock on boot up .....	43
37.1. preconditions .....	43
37.2. procedure .....	43
37.3. Expected result .....	43
38. TST037 verify the PHY MDIO. ....	44
38.1. preconditions .....	44
38.2. procedure .....	44
38.3. Expected result .....	44
39. TST038 verify the TLV CODEC test. ....	45
39.1. preconditions .....	45
39.2. procedure .....	45
39.3. Expected result .....	45
40. TST039 verify RP Message test. ....	46
40.1. preconditions .....	46
40.2. procedure .....	46
40.3. Expected result .....	46
41. TST040 verify IPC Big Data test. ....	49
41.1. preconditions .....	49
41.2. procedure .....	49
41.3. Expected Result .....	49
42. TST041 Verify u-boot log on UART-3/ttyS2. ....	51
42.1. preconditions .....	51
42.2. procedure .....	51
42.3. Expected result .....	51
43. TST042 verify Linux log on UART-3/ttyS2. ....	52
43.1. preconditions .....	52
43.2. procedure .....	52
43.3. Expected result .....	52
44. TST043 Verify Loopback test on UART-2/ttyS1 .....	53

44.1. preconditions .....	53
44.2. procedure .....	53
44.3. Expected result .....	53
45. TST044 Verify Loopback test on UART-4/ttys3 .....	54
45.1. preconditions .....	54
45.2. procedure .....	54
45.3. Expected result .....	54
46. TST045 Verify AMBE CODEC reset pin toggling .....	55
46.1. preconditions .....	55
46.2. procedure .....	55
46.3. Expected result .....	55
47. TST046 Verify USB Webcam test .....	56
47.1. preconditions .....	56
47.2. procedure .....	56
47.3. 1.Record Camera using GStreamer CLI .....	56
47.4. 2.Record Camera using OpenCV in Python .....	56
47.5. 3.Record Camera using OpenCV in C++ .....	57
47.6. Expected result .....	58
48. TST047 Verify SPI Loopback test. ....	59
48.1. preconditions .....	59
48.2. procedure .....	59
48.3. Expected result .....	59
49. TST048 Verify eMMC Boot test. ....	60
49.1. preconditions .....	60
49.2. procedure .....	60
49.3. eMMC Flashing instructions .....	60
49.4. Boot from SD card .....	60
49.5. Create partitions on eMMC .....	60
49.6. Create EXT4 file system type for rootfs .....	61
49.7. Mount the eMMC partitions .....	61
49.8. Copy Rootfs and U-Boot to mount points .....	61
49.9. Create a test file .....	61
49.10. Boot from eMMC .....	61
49.11. Expected result .....	61
50. TST049 Verify Loopback test on UART-n/ttyS(n-1) .....	62
50.1. preconditions .....	62
50.2. procedure .....	62
50.3. Expected result .....	62
51. TST050 Verify GPIO(port_pin) toggling .....	63
51.1. preconditions .....	63
51.2. procedure .....	63
51.3. Expected result .....	63
52. TST051 Verify GPIO PIN toggling .....	65
52.1. preconditions .....	65
52.2. Procedure .....	65
52.3. Expected result .....	65
53. TST052 Verify AMBE GPIO PIN toggling .....	66
53.1. preconditions .....	66
53.2. procedure .....	66
53.3. Expected result .....	66

# Chapter 1. Initial test set up

- AM572X-BEL-MMRFIC board
- SD card with image

# Chapter 2. TST001 Verify whether board is booting properly after powered up

## 2.1. Procedure

- Provide power supply to phytec carrier board and press power button for booting
- Check whether it is booting correctly or struck anywhere

## 2.2. Expected result

- Board should be booted successfully



# **Chapter 3. TST002 Verify whether board is rebooting properly after Reset Button pressed**

## **3.1. Procedure**

- Press the Reset Button when Board is in Running
- Check whether it is rebooting properly or struck anywhere

## **3.2. Expected result**

- Board should be rebooted successfully

# **Chapter 4. TST003 Verify the consistency of booting when power cycle a Device multiple times**

## **4.1. Procedure**

- Plug the power cable and press the power button check the board is booting properly
- Unplug and plug the power button for multiple times and check the board is properly booting

## **4.2. Expected result**

- The board should boot successfully for multiple times also

# Chapter 5. TST004 Check release version

## 5.1. Procedure

- We can check the release version by using the below command

```
$ cat /etc/os-release.
```

## 5.2. Expected result

```
ID="arago"  
NAME="Arago"  
VERSION="2019.11"  
VERSION_ID="2019.11"  
PRETTY_NAME="Arago 2019.11"
```

# Chapter 6. TST005 Check the U-boot software version

## 6.1. Procedure

- After booting the board at the initial stage of the boot log we can check the version of U-Boot

## 6.2. Expected result

```
U-Boot 2019.01-dirty (Jul 22 2022 - 13:25:15 +0000)
```

# Chapter 7. TST006 Check the Kernel version

## 7.1. procedure

- After completion of booting process we can check kernel version

```
$ uname -r
```

## 7.2. Expected result

- Kernel version should be displayed

# Chapter 8. TST007 Verify the total RAM size

## 8.1. Procedure

- We can Check ram size by using command

```
$ cat /proc/meminfo
```

## 8.2. Expected result

```
MemTotal:      1847828 kB
MemFree:       1614256 kB
MemAvailable:  1741080 kB
Buffers:       12792 kB
Cached:        133992 kB
SwapCached:    0 kB
Active:        46652 kB
Inactive:      117928 kB
Active(anon):  19176 kB
Inactive(anon): 9028 kB
Active(file):  27476 kB
Inactive(file): 108900 kB
Unevictable:   0 kB
Mlocked:      0 kB
HighTotal:     1291264 kB
HighFree:      1131312 kB
LowTotal:      556564 kB
LowFree:       482944 kB
SwapTotal:     0 kB
SwapFree:      0 kB
Dirty:         0 kB
Writeback:     0 kB
AnonPages:     17832 kB
Mapped:        26532 kB
Shmem:         10400 kB
Slab:          38532 kB
SReclaimable:  15020 kB
SUnreclaim:    23512 kB
KernelStack:   1072 kB
PageTables:    1224 kB
NFS_Unstable:  0 kB
Bounce:        0 kB
WritebackTmp:  0 kB
CommitLimit:   923912 kB
Committed_AS:  202984 kB
VmallocTotal:  245760 kB
VmallocUsed:    0 kB
VmallocChunk:  0 kB
Percpu:        312 kB
CmaTotal:      188416 kB
```

CmaFree:	175148 kB
----------	-----------

# Chapter 9. TST008 Verify the available RAM space size

## 9.1. Procedure

- we can check the available RAM size by using the command

```
$ cat /proc/meminfo
```

## 9.2. Expected result

```
MemTotal:      1847828 kB
MemFree:       1614256 kB
MemAvailable:  1741080 kB
Buffers:       12792 kB
Cached:        133992 kB
SwapCached:      0 kB
Active:        46652 kB
Inactive:      117928 kB
Active(anon):   19176 kB
Inactive(anon): 9028 kB
Active(file):   27476 kB
Inactive(file): 108900 kB
Unevictable:    0 kB
Mlocked:        0 kB
HighTotal:     1291264 kB
HighFree:      1131312 kB
LowTotal:      556564 kB
LowFree:       482944 kB
SwapTotal:      0 kB
SwapFree:      0 kB
Dirty:         0 kB
Writeback:     0 kB
AnonPages:     17832 kB
Mapped:        26532 kB
Shmem:         10400 kB
Slab:          38532 kB
SReclaimable:  15020 kB
SUnreclaim:    23512 kB
KernelStack:   1072 kB
PageTables:    1224 kB
NFS_Unstable:  0 kB
Bounce:        0 kB
WritebackTmp:  0 kB
CommitLimit:   923912 kB
Committed_AS:  202984 kB
VmallocTotal:  245760 kB
VmallocUsed:    0 kB
VmallocChunk:  0 kB
Percpu:        312 kB
CmaTotal:      188416 kB
```



CmaFree:	175148 kB
----------	-----------

# Chapter 10. TST009 Verify the total Storage size

## 10.1. Procedure

- we can check the total storage size by using the below command

```
$ df -h
```

## 10.2. Expected result

Filesystem	Size	Used	Available	Use%	Mounted on
/dev/root	7.2G	6.7G	107.4M	98%	/
devtmpfs	809.3M	4.0K	809.3M	0%	/dev
tmpfs	902.3M	8.0K	902.3M	0%	/dev/shm
tmpfs	902.3M	9.5M	892.7M	1%	/run
tmpfs	902.3M	0	902.3M	0%	/sys/fs/cgroup
tmpfs	902.3M	608.0K	901.7M	0%	/tmp
tmpfs	16.0M	0	16.0M	0%	/media/ram
tmpfs	50.0M	16.0K	50.0M	0%	/var/volatile
/dev/mmcblk0p1	4.0M	770.0K	3.2M	19%	/run/media/mmcblk0p1
/dev/mmcblk1p2	2.2G	1.8G	209.9M	90%	/run/media/mmcblk1p2
tmpfs	180.4M	0	180.4M	0%	/run/user/0

# Chapter 11. TST010 Verify the Available Storage size

## 11.1. Procedure

- we can check the available storage size by using the below command

```
$ df -h
```

## 11.2. Expected result

Filesystem	Size	Used	Available	Use%	Mounted on
/dev/root	7.2G	6.7G	107.4M	98%	/
devtmpfs	809.3M	4.0K	809.3M	0%	/dev
tmpfs	902.3M	8.0K	902.3M	0%	/dev/shm
tmpfs	902.3M	9.5M	892.7M	1%	/run
tmpfs	902.3M	0	902.3M	0%	/sys/fs/cgroup
tmpfs	902.3M	608.0K	901.7M	0%	/tmp
tmpfs	16.0M	0	16.0M	0%	/media/ram
tmpfs	50.0M	16.0K	50.0M	0%	/var/volatile
/dev/mmcblk0p1	4.0M	770.0K	3.2M	19%	/run/media/mmcblk0p1
/dev/mmcblk1p2	2.2G	1.8G	209.9M	90%	/run/media/mmcblk1p2
tmpfs	180.4M	0	180.4M	0%	/run/user/0

# Chapter 12. TST011 Verify the SOC Name and number of core

## 12.1. Procedure

- By using the below command we can check the SOC name and number of core

```
$ cat /proc/cpuinfo
```

## 12.2. Expected result

```
processor       : 0
model name     : ARMv7 Processor rev 2 (v7l)
BogoMIPS      : 12.29
Features       : half thumb fastmult vfp edsp neon vfpv3 tls vfpv4 idiva idivt vfpd3
CPU implementer : 0x41
CPU architecture: 7
CPU variant    : 0x2
CPU part       : 0xc0f
CPU revision   : 2

processor       : 1
model name     : ARMv7 Processor rev 2 (v7l)
BogoMIPS      : 12.29
Features       : half thumb fastmult vfp edsp neon vfpv3 tls vfpv4 idiva idivt vfpd3
CPU implementer : 0x41
CPU architecture: 7
CPU variant    : 0x2
CPU part       : 0xc0f
CPU revision   : 2
Hardware       : Generic DRA74X (Flattened Device Tree)
Revision       : 0000
Serial        : 0b01400e64fa0922
```

# Chapter 13. TST012 Verify the architecture of SOC

## 13.1. Procedure

- we can find the architecture of SOC by using the below command

```
$ uname -m
```

## 13.2. Expected result

```
armv7l
```

# Chapter 14. TST013 Verify whether Ethernet ports are detected

## 14.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Ethernet ports is to be enabled

## 14.2. Procedure

- Connecting ethernet port and pc through ethernet cable we can find whether ethernet ports are detected by using the below commands

```
$ ethtool eth0
```

or

```
$ ethtool eth1
```

## 14.3. Expected result

```
Settings for eth1:
  Supported ports: [ TP MII ]
  Supported link modes:   10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
                        1000baseT/Half 1000baseT/Full
  Supported pause frame use: Symmetric Receive-only
  Supports auto-negotiation: Yes
  Supported FEC modes: Not reported
  Advertised link modes:  10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
                        1000baseT/Half 1000baseT/Full
  Advertised pause frame use: No
  Advertised auto-negotiation: Yes
  Advertised FEC modes: Not reported
  Link partner advertised link modes:  10baseT/Half 10baseT/Full
                                      100baseT/Half 100baseT/Full
                                      1000baseT/Full
  Link partner advertised pause frame use: No
  Link partner advertised auto-negotiation: Yes
  Link partner advertised FEC modes: Not reported
  Speed: 1000Mb/s
  Duplex: Full
  Port: MII
  PHYAD: 2
  Transceiver: internal
  Auto-negotiation: on
  Supports Wake-on: d
  Wake-on: d
  Current message level: 0x00000000 (0)
```

Link detected: yes

Settings for eth0:

Supported ports: [ TP MII ]

Supported link modes: 10baseT/Half 10baseT/Full  
100baseT/Half 100baseT/Full  
1000baseT/Half 1000baseT/Full

Supported pause frame use: Symmetric Receive-only

Supports auto-negotiation: Yes

Supported FEC modes: Not reported

Advertised link modes: 10baseT/Half 10baseT/Full  
100baseT/Half 100baseT/Full  
1000baseT/Half 1000baseT/Full

Advertised pause frame use: No

Advertised auto-negotiation: Yes

Advertised FEC modes: Not reported

Speed: 10Mb/s

Duplex: Half

Port: MII

PHYAD: 1

Transceiver: internal

Auto-negotiation: on

Supports Wake-on: d

Wake-on: d

Current message level: 0x00000000 (0)

Link detected: no

# Chapter 15. TST014 Verify LAN Ethernet port is getting ip address when it is connected to LAN

## 15.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Ethernet ports is to be enabled

## 15.2. Procedure

- Connect ethernet port in phytec board and check whether the port is getting ip address from WAN by using the command

```
$ ifconfig
```

## 15.3. Expected result

```
eth0      Link encap:Ethernet  HWaddr 50:51:A9:92:C8:92
          inet addr:10.42.0.27  Bcast:10.42.0.255  Mask:255.255.255.0
          inet6 addr: fe80::5251:a9ff:fe92:c892/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:163 errors:0 dropped:0 overruns:0 frame:0
          TX packets:148 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:37147 (36.2 KiB)  TX bytes:26487 (25.8 KiB)
          Interrupt:98

eth1      Link encap:Ethernet  HWaddr 50:51:A9:92:C8:93
          inet6 addr: fe80::5251:a9ff:fe92:c893/64 Scope:Link
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:32 errors:0 dropped:0 overruns:0 frame:0
          TX packets:69 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7737 (7.5 KiB)  TX bytes:11460 (11.1 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:140 (140.0 B)  TX bytes:140 (140.0 B)
```



# Chapter 16. TST015 Verify the GPIO pins are populated in sysfs

## 16.1. Preconditions

- AM572X-BEL-MMRFIC board
- SD card
- GPIO pins is to be enabled

## 16.2. Procedure

- Go to /sys/bus and check whether the GPIO pins are enabled.

## 16.3. Expected result

cec	gpio	mmc	rpmsg	spmi
clockevents	hid	mmc_rpmb	scsi	usb
clocksource	i2c	nvmem	sdio	usb-serial
container	iio	pci	serial	virtio
cpu	mdio_bus	pci-epf	serio	wl
edac	media	pci_express	soc	workqueue
event_source	mipi-dsi	platform	spi	

# Chapter 17. TST016 Verify whether UART ports are Detected

## 17.1. Preconditions

- AM572X-BEL-MMRFIC board
- SD card
- UART Image is to be flashed

## 17.2. Procedure

- Go to /sys/class/ and check whether UART is detecting or not.
- ls /sys/class/tty/tty\*

## 17.3. Expected result

```
ttys0:
dev      power      subsystem uevent

ttys1:
dev      power      subsystem uevent

ttys2:
dev      power      subsystem uevent

ttys3:
dev      power      subsystem uevent

ttys4:
dev      power      subsystem uevent

ttys5:
dev      power      subsystem uevent

ttys6:
dev      power      subsystem uevent

ttys7:
dev      power      subsystem uevent

ttys8:
dev      power      subsystem uevent

ttys9:
dev      power      subsystem uevent

ttysa:
dev      power      subsystem uevent

ttysb:
dev      power      subsystem uevent
```

```
ttysc:
dev      power      subsystem uevent

ttysd:
dev      power      subsystem uevent

ttyse:
dev      power      subsystem uevent

ttysf:
dev      power      subsystem uevent
```

# Chapter 18. TST017 Verify whether \$(n) UART ports are detected

## 18.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 18.2. Procedure

- Check the UART ports are detected using below command

```
$ ls /dev/ttyS*
```

## 18.3. Expected result

- Number of UART ports detected should be 7

```
/dev/ttyS0 /dev/ttyS1 /dev/ttyS2 /dev/ttyS3 /dev/ttyS4 /dev/ttyS5 /dev/ttyS6 /de
```

# Chapter 19. TST018 Verify whether \$(n) I<sup>2</sup>C ports are detected

## 19.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 19.2. Procedure

- Check the I<sup>2</sup>C ports are detected using below command

```
$ ls /dev/i2c-*
```

## 19.3. Expected result

- Number of I<sup>2</sup>C ports detected should be 5

```
/dev/i2c-0 /dev/i2c-2 /dev/i2c-3
```

# Chapter 20. TST019 Verify Scanning of slave devices in I<sup>2</sup>C buses

## 20.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 20.2. Procedure

- Run i2cdetect command to scan the slave devices connected to i2c bus
- You will need to specify <busno> as the i2c port number

```
$ i2cdetect -y <busno>
```

## 20.3. Expected result

- It should output a table of the currently detected I<sup>2</sup>C devices

```
root@am572x-bel-mmrfic:~# i2cdetect -y 0
Warning: Can't use SMBus Quick Write command, will skip some addresses
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:
10:
20:
30: -- -- -- -- -- -- -- --
40:
50: UU -- -- -- -- -- -- -- UU UU UU 5b -- -- -- --
60:
70:
```

```
root@am572x-bel-mmrfic:~# i2cdetect -y 2
Warning: Can't use SMBus Quick Write command, will skip some addresses
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:
10:
20:
30: -- -- -- -- -- -- -- --
40:
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60:
70:
```

```
root@am572x-bel-mmrfic:~# i2cdetect -y 3
Warning: Can't use SMBus Quick Write command, will skip some addresses
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:
10:
20:
30: -- -- -- -- -- -- -- --
40:
```

50: - - - - -  
60:  
70:

# Chapter 21. TST020 Verify whether \$(n) SPI ports are detected

## 21.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 21.2. Procedure

- Check the SPI ports are detected using below command

```
$ ls /dev/spi*
```

## 21.3. Expected result

- Number of SPI ports detected should be 2

```
/dev/spidev1.0  
/dev/spidev1.1
```



# Chapter 22. TST021 Verify whether \$(n) USB ports are detected

## 22.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 22.2. Procedure

- Check the USB ports are detected using below command

```
$ lsusb
```

## 22.3. Expected result

```
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
```

# Chapter 23. TST022 Verify whether USB slave device is detected

## 23.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image
- Pendrive

## 23.2. Procedure

- Connect the pendrive in USB port 1 of board
- Check the dmesg log to verify whether pendrive is detected in USB port 1

## 23.3. Expected result

- Pendrive should be detected in USB port 1

```
root@am572x-bel-mmrfic:~# [ 3975.978165] usb 2-1: new SuperSpeed Gen 1 USB device number 1
[ 3976.009065] usb 2-1: New USB device found, idVendor=0781, idProduct=5581, bcdDevice=1.00
[ 3976.017285] usb 2-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 3976.025121] usb 2-1: Product: Ultra
[ 3976.028976] usb 2-1: Manufacturer: SanDisk
[ 3976.033093] usb 2-1: SerialNumber: 4C531003560526109191
[ 3976.129677] usb-storage 2-1:1.0: USB Mass Storage device detected
[ 3976.136532] scsi host1: usb-storage 2-1:1.0
[ 3976.142295] usbcore: registered new interface driver usb-storage
[ 3976.152572] usbcore: registered new interface driver uas
[ 3977.199035] scsi 1:0:0:0: Direct-Access      SanDisk Ultra                1.00 PQ: 0 ANSI
[ 3977.226311] sd 1:0:0:0: [sda] 30464000 512-byte logical blocks: (15.6 GB/14.5 GiB)
[ 3977.234966] sd 1:0:0:0: [sda] Write Protect is off
[ 3977.240174] sd 1:0:0:0: [sda] Write cache: disabled, read cache: enabled, doesn't support
[ 3977.255074] sda: sda1
[ 3977.259567] sd 1:0:0:0: [sda] Attached SCSI removable disk
[ 3991.098360] usb 2-1: USB disconnect, device number 2
[ 3991.268280] FAT-fs (sda1): unable to read boot sector to mark fs as dirty
```

# Chapter 24. TST023 Verify Audio codec driver is detected

## 24.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 24.2. Procedure

- Check the audio codec driver are detected using below command

```
$ dmesg | grep "audio"
```

## 24.3. Expected result

- Audio driver Initialization should be shown in dmesg log

```
root@am572x-bel-mmrfic:~# dmesg | grep "audio"
[ 9.004476] omap-hdmi-audio omap-hdmi-audio.0.auto: snd-soc-dummy-dai <-> 58040000.e
[ 9.044884] omap-hdmi-audio omap-hdmi-audio.0.auto: ASoC: no DMI vendor name!
[ 9.401116] of_get_named_gpiod_flags: can't parse 'simple-audio-card,hp-det-gpio' pr
[ 9.401134] of_get_named_gpiod_flags: can't parse 'simple-audio-card,mic-det-gpio' p
```

# Chapter 25. TST024 Verify Audio ports are detected

## 25.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 25.2. Procedure

- Check the list of audio port detected using below command

```
$ aplay -l
```

## 25.3. Expected result

- List of audio ports should be detected

```
aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: PCH [HDA Intel PCH], device 0: ALC3204 Analog [ALC3204 Analog]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 3: HDMI 0 [HDMI 0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 7: HDMI 1 [HDMI 1]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 8: HDMI 2 [HDMI 2]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

# Chapter 26. TST025 Verify whether Board enter into uboot prompt properly when press any key

## 26.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 26.2. Procedure

- Power up the board
- While booting the board, check the autoboot count is 5 seconds in boot log
- Then press enter key go to u-boot prompt

## 26.3. Expected result

- Board should be entered into uboot prompt

```
U-Boot 2019.01-dirty (Jul 22 2022 - 13:25:15 +0000)

CPU   : DRA752-GP ES2.0
Model: MMRFIC phyCORE-AM572x Carrier Board
Board: AM572X-BEL-MMRFIC
DRAM:  2 GiB
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
Loading Environment from FAT... *** Warning - bad CRC, using default environment

Loading Environment from MMC... *** Warning - bad CRC, using default environment

PHYTEC: unknown board name. Defaulting to am572x_bel_mmrfic, a MINIMAL AM5716 configura
Warning: fastboot.board_rev: unknown board revision
Net:
Warning: ethernet@48484000 using MAC address from ROM
eth0: ethernet@48484000
Hit any key to stop autoboot:  0
=>
=>
```

# Chapter 27. TST026 Check the board SOC in u-boot

## 27.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 27.2. Procedure

- Check the board soc using below command

```
$ print board_soc
```

## 27.3. Expected result

- Board SOC should be am572x

```
=> print board_soc  
board_soc=am572x
```

# Chapter 28. TST027 Check the CPU model in u-boot

## 28.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 28.2. Procedure

- Check the CPU model using below command

```
$ print cpu
```

## 28.3. Expected result

- CPU model should be armv7

# Chapter 29. TST028 Check the board name in u-boot

## 29.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 29.2. Procedure

- Check the board name using below command

```
$ print board_name
```

## 29.3. Expected result

- Board name should be am572x\_bel\_mmrfic

```
=> print board_name  
board_name=am572x_bel_mmrfic
```



# Chapter 30. TST029 Check the U-boot version

## 30.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 30.2. Procedure

- Check the U-boot using below command

```
$ print ver
```

## 30.3. Expected result

- U-boot version should be ver=U-Boot 2019.01-dirty

```
=> print ver  
ver=U-Boot 2019.01-dirty (Jul 22 2022 - 13:25:15 +0000)
```

# Chapter 31. TST030 Verify whether kernel log is displayed using dmesg

## 31.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 31.2. Procedure

- Check the kernel log is displayed in dmesg

```
$ dmesg
```

## 31.3. Expected result

- Kernel log should be shown in dmesg command output

# Chapter 32. TST031 Verify whether Kernel data structure is accessible using sysfs

## 32.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 32.2. Procedure

- check kernel data structure is accessible by getting current kernel log level from sysfs

```
$ cat /proc/sys/kernel/printk
```

## 32.3. Expected result

- Kernel data should be accessible using sysfs

```
7      4      1      7
```

# Chapter 33. TST032 Verify whether board shell is accessible

## 33.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 33.2. Procedure

- Power up the board
- After Board booted up, login the board
- Run sh command and check whether shell is accessible

```
$ sh
```

```
root@am572x-bel-mmrfic:~# ls /bin/sh
/bin/sh
root@am572x-bel-mmrfic:~# echo $SHELL
/bin/sh
```

# Chapter 34. TST033 Check whether Rootfs have basic commands for testing

## 34.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 34.2. Procedure

- Power up and login the board
- Run below commands and check whether commands are available in board

```
* dmesg
* i2cdetect -V
* aplay --version
* spidev_test -v
* ls /dev/ttyS*
* dmesg | grep <name>
```

## 34.3. Expected result

- All mentioned Basic commands should be available in board

# Chapter 35. TST034 Verify MMC\_1 8 pins working as SD Card boot

## 35.1. precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 35.2. procedure

- Insert image flashed SD card into card slot
- Power up the board

## 35.3. Expected Result

- Board Should boot, and we should get the linux prompt

# Chapter 36. TST035 Verify ETH\_0 8 pins working as 10/100/1000M Ethernet(RGMII0)

## 36.1. Precondition

- AM572X-BEL-MMRFIC board
- SD card with image

## 36.2. Procedure

- set the speed(10/100/1000M) and other properties using ethtool
- connect the live ethernet cable to the connector
- Check the configured parameters using ethtool

## 36.3. Expected Result

```
root@am572x-bel-mmrfic:~# ethtool eth0
Settings for eth0:
    Supported ports: [ TP MII ]
    Supported link modes:   10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Half 1000baseT/Full
    Supported pause frame use: Symmetric Receive-only
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  100baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Link partner advertised link modes:  10baseT/Half 10baseT/Full
                                         100baseT/Half 100baseT/Full
    Link partner advertised pause frame use: No
    Link partner advertised auto-negotiation: Yes
    Link partner advertised FEC modes: Not reported
    Speed: 100Mb/s
    Duplex: Full
    Port: MII
    PHYAD: 1
    Transceiver: internal
    Auto-negotiation: on
    Supports Wake-on: d
    Wake-on: d
    Current message level: 0x00000000 (0)

    Link detected: yes

root@am572x-bel-mmrfic:~# ethtool -s eth0 speed 1000 duplex full autoneg on
```

```
root@am572x-bel-mmrfic:~# ethtool eth0
Settings for eth0:
    Supported ports: [ TP MII ]
    Supported link modes:   10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Half 1000baseT/Full
    Supported pause frame use: Symmetric Receive-only
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  1000baseT/Full
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Link partner advertised link modes:  10baseT/Half 10baseT/Full
                                         100baseT/Half 100baseT/Full
                                         1000baseT/Full
    Link partner advertised pause frame use: No
    Link partner advertised auto-negotiation: Yes
    Link partner advertised FEC modes: Not reported
    Speed: 1000Mb/s
    Duplex: Full
    Port: MII
    PHYAD: 1
    Transceiver: internal
    Auto-negotiation: on
    Supports Wake-on: d
    Wake-on: d
    Current message level: 0x00000000 (0)

    Link detected: yes
```



# Chapter 37. TST036 verify AMBE Codec 4 MHz clock on boot up

## 37.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Digital Oscilloscope

## 37.2. procedure

- Boot up the board.
- check the resistor R51 with DSO Probe.

## 37.3. Expected result

- 4MHZ waveform on DSO.

# Chapter 38. TST037 verify the PHY MDIO.

## 38.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Ethernet Driver

## 38.2. procedure

- Boot into linux.
- Connect ethernet cable.
- Configure eth0 speed to 100Mbps/1000Mbps using `ethtool`
  - `ethtool -s eth speed <100/1000> duplex full autoneg on`

## 38.3. Expected result

```
root@am572x-bel-mmrfic:/sys/class/leds# ethtool -s eth0 speed 100 duplex half autoneg on
root@am572x-bel-mmrfic:/sys/class/leds# ethtool eth0
Settings for eth0:
    Supported ports: [ TP MII ]
    Supported link modes:   10baseT/Half 10baseT/Full
                           100baseT/Half 100baseT/Full
                           1000baseT/Half 1000baseT/Full
    Supported pause frame use: Symmetric Receive-only
    Supports auto-negotiation: Yes
    Supported FEC modes: Not reported
    Advertised link modes:  100baseT/Half
    Advertised pause frame use: No
    Advertised auto-negotiation: Yes
    Advertised FEC modes: Not reported
    Link partner advertised link modes:  10baseT/Half 10baseT/Full
                                         100baseT/Half 100baseT/Full
    Link partner advertised pause frame use: No
    Link partner advertised auto-negotiation: Yes
    Link partner advertised FEC modes: Not reported
    Speed: 100Mb/s
    Duplex: Half
    Port: MII
    PHYAD: 1
    Transceiver: internal
    Auto-negotiation: on
    Supports Wake-on: d
    Wake-on: d
    Current message level: 0x00000000 (0)

    Link detected: yes
```

# Chapter 39. TST038 verify the TLV CODEC test.

## 39.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card

## 39.2. procedure

- Boot into linux.
- Play the test file aplay.

## 39.3. Expected result

```
root@am572x-bel-mmrfic:/sys/class/leds# aplay /usr/share/sounds/alsa/Noise.wav
Playing WAVE '/usr/share/sounds/alsa/Noise.wav' : Signed 16 bit Little Endian, Rate 480
```

# Chapter 40. TST039 verify RP Message test.

## 40.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- DSP Firmware

## 40.2. procedure

- Boot the board to Linux.
- Load the firmware to DSP Engines and IPU Engines.
- Execute the below commands.

```
# DSP 1

$ ln -sf /home/root/dsp-binaries/rpmsg/server_dsp1.xe66 /lib/firmware/dra7-dsp1-fw.xe66
$ echo 40800000.dsp > /sys/bus/platform/drivers/omap-rproc/unbind
$ echo 40800000.dsp > /sys/bus/platform/drivers/omap-rproc/bind
$ cd /root/dsp-binaries/rpmsg
$ ./app_host DSP1

# DSP 2

$ ln -sf /home/root/dsp-binaries/rpmsg/server_dsp1.xe66 /lib/firmware/dra7-dsp2-fw.xe66
$ echo 41000000.dsp > /sys/bus/platform/drivers/omap-rproc/unbind
$ echo 41000000.dsp > /sys/bus/platform/drivers/omap-rproc/bind
$ cd /root/dsp-binaries/rpmsg
$ ./app_host DSP2

# IPU 1

$ ln -sf /home/root/dsp-binaries/rpmsg/server_ipu1.xem4 /lib/firmware/dra7-ipu1-fw.xem4
$ echo 58820000.ipu > /sys/bus/platform/drivers/omap-rproc/unbind
$ echo 58820000.ipu > /sys/bus/platform/drivers/omap-rproc/bin
$ cd /root/dsp-binaries/rpmsg
$ ./app_host IPU1

# IPU 2

$ ln -sf /home/root/dsp-binaries/rpmsg/server_ipu1.xem4 /lib/firmware/dra7-ipu2-fw.xem4
$ echo 55020000.ipu > /sys/bus/platform/drivers/omap-rproc/unbind
$ echo 55020000.ipu > /sys/bus/platform/drivers/omap-rproc/bin
$ cd /root/dsp-binaries/rpmsg
$ ./app_host IPU2
```

## 40.3. Expected result

```
root@am572x-bel-mmrfic:~/dsp-binaries/rpmsg# ./app_host DSP1
```

```
--> main:
--> Main_main:
--> App_create:
App_create: Host is ready
<-- App_create:
--> App_exec:
App_exec: sending message 1
App_exec: sending message 2
App_exec: sending message 3
App_exec: message received, sending message 4
.
.
App_exec: message received, sending message 15
App_exec: message received
App_exec: message received
App_exec: message received
<-- App_exec: 0
--> App_delete:
<-- App_delete:
<-- Main_main:
<-- main:

root@am572x-bel-mmrfic:~/dsp-binaries/rpmsg# ./app_host DSP2
--> main:
--> Main_main:
--> App_create:
App_create: Host is ready
<-- App_create:
--> App_exec:
App_exec: sending message 1
App_exec: sending message 2
App_exec: sending message 3
App_exec: message received, sending message 4
.
.
App_exec: message received, sending message 15
App_exec: message received
App_exec: message received
App_exec: message received
<-- App_exec: 0
--> App_delete:
<-- App_delete:
<-- Main_main:
<-- main:

root@am572x-bel-mmrfic:~/dsp-binaries/rpmsg# ./app_host IPU1
--> main:
--> Main_main:
--> App_create:
App_create: Host is ready
<-- App_create:
--> App_exec:
```

```
App_exec: sending message 1
App_exec: sending message 2
App_exec: sending message 3
App_exec: message received, sending message 4
.
.
App_exec: message received, sending message 15
App_exec: message received
App_exec: message received
App_exec: message received
<-- App_exec: 0
--> App_delete:
<-- App_delete:
<-- Main_main:
<-- main:

root@am572x-bel-mmrfic:~/dsp-binaries/rpmsg# ./app_host IPU2
--> main:
--> Main_main:
--> App_create:
App_create: Host is ready
<-- App_create:
--> App_exec:
App_exec: sending message 1
App_exec: sending message 2
App_exec: sending message 3
App_exec: message received, sending message 4
.
.
App_exec: message received, sending message 15
App_exec: message received
App_exec: message received
App_exec: message received
<-- App_exec: 0
--> App_delete:
<-- App_delete:
<-- Main_main:
<-- main:
```

# Chapter 41. TST040 verify IPC Big Data test.

## 41.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- DSP binary

## 41.2. procedure

- Boot the board to Linux.
- Load the firmware to DSP.
- Execute the below commands.

```
# DSP 1
$ ln -sf /home/root/dspl-bigdata/server_dsp.xe66 /lib/firmware/dra7-dsp1-fw.xe66
$ echo 40800000.dsp > /sys/bus/platform/drivers/omap-rproc/unbind
$ echo 40800000.dsp > /sys/bus/platform/drivers/omap-rproc/bind
$ cd /root/dspl-bigdata
$ ./app_host DSP1

# DSP 2
$ ln -sf /home/root/dspl-bigdata/server_dsp.xe66 /lib/firmware/dra7-dsp2-fw.xe66
$ echo 41000000.dsp > /sys/bus/platform/drivers/omap-rproc/unbind
$ echo 41000000.dsp > /sys/bus/platform/drivers/omap-rproc/bind
$ cd /root/dspl-bigdata
$ ./app_host DSP2
```

## 41.3. Expected Result

```
$ ./app_host DSP1
--> main:
--> Main_main:
--> App_create:
App_create: Host is ready
<-- App_create:
--> App_exec:
CMEM_init success
CMEM_allocPool success: Allocated buffer 0xaa50e000
SharedRegion_setup success
HeapMem_setup success
.
HeapMem_create success
App_exec: sending message 1
.
App_exec: sending message 2
App_exec: sending message 3
.
App_exec: message received: 16
```

```
App_exec: Data check clean
<-- App_exec: 0
--> App_delete:
<-- App_delete:
<-- Main_main:
Host: Test Passed
<-- main:

$ ./app_host DSP2
--> main:
--> Main_main:
--> App_create:
App_create: Host is ready
<-- App_create:
--> App_exec:
CMEM_init success
CMEM_allocPool success: Allocated buffer 0xaa50e000
SharedRegion_setup success
HeapMem_setup success
.
HeapMem_create success
App_exec: sending message 1
.
App_exec: sending message 2
App_exec: sending message 3
.
App_exec: message received: 16
App_exec: Data check clean
<-- App_exec: 0
--> App_delete:
<-- App_delete:
<-- Main_main:
Host: Test Passed
<-- main:
```



# Chapter 42. TST041 Verify u-boot log on UART-3/ttyS2.

## 42.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card

## 42.2. procedure

- Connect UART3 to Serial Terminal.
- Power on the Board.
- Press Enter Key on bootup.

## 42.3. Expected result

- U-Boot log should be shown as below.

```
U-Boot 2019.01-dirty (Jul 22 2022 - 13:25:15 +0000)

CPU   : DRA752-GP ES2.0
Model: MMRFIC phyCORE-AM572x Carrier Board
Board: AM572X-BEL-MMRFIC
DRAM:  2 GiB
MMC:   OMAP SD/MMC: 0, OMAP SD/MMC: 1
.
.
.
.
Hit any key to stop autoboot:  0
=>
=>
```

# Chapter 43. TST042 verify Linux log on UART-3/ttyS2.

## 43.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card

## 43.2. procedure

- Connect UART4 to Serial Terminal/PC.
- Power on the Board.

## 43.3. Expected result

- Linux boot log should be shown in Serial Terminal.

# Chapter 44. TST043 Verify Loopback test on UART-2/ttyS1

## 44.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Loopback setup

## 44.2. procedure

- 1.Connect(Loopback) Carrier Board MB Connector PIN#133(RX) & PIN#134(TX)
- 2.Run the test as below
- 3.TX and RX data should be same

## 44.3. Expected result

```
root@am572x-bel-mmrfic:~# uart_test /dev/ttyS1
Writting Data.....
Device Reading .....
mmrfhello
Loopback Test Success
```

# Chapter 45. TST044 Verify Loopback test on UART-4/ttys3

## 45.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Loopback setup

## 45.2. procedure

1.Connect(Loopback) Carrier Board MB Connector PIN#(RX) & PIN#(TX) 2.Run the test as below 3.TX and RX data should be same

## 45.3. Expected result

```
root@am572x-bel-mmrfic:~# uart_test /dev/ttyS3
Writting Data.....
Device Reading .....
mmrfhello
Loopback Test Success
```

# Chapter 46. TST045 Verify AMBE CODEC reset pin toggling

## 46.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Digital Oscilloscope

## 46.2. procedure

- Export GPIO\_13(MBC - 42)
- Set direction to `out`.
- set a number to `value` file.

```
$ cd /sys/class/gpio
$ echo 173 > export
$ cd gpio173
$ echo out > direction
$ echo 1 > value
$ echo 0 > value
```

## 46.3. Expected result

- MBC PIN#42 should toggle as per the value written.

# Chapter 47. TST046 Verify USB Webcam test

## 47.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Digital Oscilloscope

## 47.2. procedure

- Connect the USB webcam to the USB port.
- We can record the stream using the following three methods 1.gstreamer pipeline 2.Python script 3.cpp code

### 47.3. 1.Record Camera using GStreamer CLI

- The following gstreamer pipeline can be used to record camera data to a file.

```
target# gst-launch-1.0 v4l2src device=/dev/video1 name=cam_src ! videoconvert ! \
      queue ! videoconvert ! avimux ! filesink location=output2.avi
```

### 47.4. 2.Record Camera using OpenCV in Python

- This example Python script saves the camera feed into a file.

#### 47.4.1. Step 1: Save the file

- Save the below Python code on the target's filesystem as **opencv\_save.py**.

```
import cv2

def save_feed_to_file(path, fps, codec):
    # open camera
    cam = cv2.VideoCapture(1)
    width = cam.get(cv2.CAP_PROP_FRAME_WIDTH) # float
    # Get current height of frame
    height = cam.get(cv2.CAP_PROP_FRAME_HEIGHT) # float

    # setup video writer object
    codec = cv2.VideoWriter_fourcc(*codec)
    out = cv2.VideoWriter(path, codec, fps, (int(width), int(height)))

    while (cam.isOpened()):
        # read a frame
        ret, frame = cam.read()

        # write frame using video writer
        out.write(frame)

def main():
    save_feed_to_file("test.avi", 60, "MJPG")
```

```
if __name__ == "__main__":  
    main()
```

### 47.4.2. Step 2: Run the script

- Use the below command to run the script.

```
target# python3 <path-to-file>/opencv_save.py
```

- NOTE: To stop saving the feed press **ctrl+c**.

## 47.5. 3.Record Camera using OpenCV in C++

- This cpp example code saves the camera feed as a file.

### 47.5.1. Step 1: Save the file

- Save the below file on target's filesystem as **opencv\_save.cpp**.

```
#include <opencv2/core.hpp>  
#include <opencv2/highgui.hpp>  
#include <opencv2/videoio.hpp>  
#include "opencv2/imgproc/imgproc.hpp"  
  
#include <iostream>  
  
using namespace std;  
using namespace cv;  
  
int main(int argc, char *argv[])  
{  
    Mat frame;  
    VideoCapture cap(1);  
  
    int width = cap.get(cv::CAP_PROP_FRAME_WIDTH);  
    int length = cap.get(cv::CAP_PROP_FRAME_HEIGHT);  
    int fps = cap.get(cv::CAP_PROP_FPS);  
    Size sizeFrame(width, length);  
  
    if (!cap.isOpened()) {  
        cerr << "Error in opening feed\n";  
        return -1;  
    }  
  
    //Video Codec, create video writer object  
    int codec = VideoWriter::fourcc('M', 'J', 'P', 'G');  
    VideoWriter writer("./feed_save.avi", codec, fps, sizeFrame);  
  
    if (!writer.isOpened()) {  
        cerr << "Error in opening writer\n";  
        return -1;  
    }  
  
    while (1) {
```

```
        // read frame
        if (!cap.read(frame)) {
            cerr << "Error, blank frame\n";
            return -1;
        }

        // write frame
        writer.write(frame);
    }
    cap.release();
    writer.release();
    return 0;
}
```

### 47.5.2. Step 2: Compile the source

- Use the below command to compile.

```
target# arm-linux-gnueabi-g++ <path-to-file>/opencv_save.cpp \
-I/usr/include/opencv4 -L/usr/local/libc -lopencv_videoio \
-lopencv_imgproc -lopencv_highgui -lopencv_core -o opencv_save
```

### 47.5.3. Step 3: Run the executable

- Run the executable as shown below.

```
target# ./opencv_save
```

- NOTE: To stop saving the feed press **ctrl+c**.

## 47.6. Expected result

- All three of the methods described above save the camera feed as a file into the webcam.



# Chapter 48. TST047 Verify SPI Loopback test.

## 48.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Loopback setup

## 48.2. procedure

1.Download and Build the spidev-test as below(<https://github.com/rm-hull/spidev-test>)

```
$ git clone https://github.com/rm-hull/spidev-test
$ cd spidev-test
$ gcc spidev_test.c -o spidev_test
$ ./spidev_test /dev/spidev{1.0/1.1}
```

2.Connect(Loopback) Carrier Board MB Connector PIN#39(MOSI) & PIN#40(MISO) 3.Run the test as below 4.TX and RX data should be same

## 48.3. Expected result

```
root@am572x-bel-mmrfic:~/spidev-test# pwd
/home/root/spidev-test
root@am572x-bel-mmrfic:~/spidev-test# ./spidev -v -D /dev/spidev1.0
spi mode: 0x0
bits per word: 8
max speed: 500000 Hz (500 KHz)
TX | FF FF FF FF FF FF 40 00 00 00 00 95 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
RX | FF FF FF FF FF FF 40 00 00 00 00 95 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

root@am572x-bel-mmrfic:~/spidev-test# ./spidev -v -D /dev/spidev1.1
spi mode: 0x0
bits per word: 8
max speed: 500000 Hz (500 KHz)
TX | FF FF FF FF FF FF 40 00 00 00 00 95 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
RX | FF FF FF FF FF FF 40 00 00 00 00 95 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

# Chapter 49. TST048 Verify eMMC Boot test.

## 49.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Loopback setup

## 49.2. procedure

## 49.3. eMMC Flashing instructions

## 49.4. Boot from SD card

- Download `tisdk-rootfs-image-am572x-bel-mmrfic.tar` , `u-boot.img` and `MLO`
- available in the release folder on workdrive.
- Create a bootable SD card and boot the board.
- Copy the images downloaded from the release folder to the SD card's rootfs

## 49.5. Create partitions on eMMC

- Run the following command:

```
target# fdisk /dev/mmcblk1
```

- The above command displays the fdisk prompt.
- Type the below commands in fdisk prompt to create the partition.
- the explanation is mentioned in the comments following the command:

```
command:: o          # Create a DOS partition
command:: n          # Add a new partition for boot
command:: p          # Make it as primary partition
command:: 1          # Create partition number 1
command:: 2048        # Allocate size of first sector
command:: +2M         # Allocate size of last sector
command:: t          # Change partition type
Hex code (type L to list all codes): c # Select partition type as W95 FAT32 (LBA)
command:: a          # Turns on bootable flag on parition 1
command:: n          # Add a new partition for rootfs
command:: 2          # Create partition number 2
command:: <enter>     # Allocate default size for first sector
command:: <enter>     # Allocate default size for last sector
command:: w          # Write table to disk and exit
```

- Create file system type using mkfs
- Create VFAT file system type for Boot partition

```
target# mkfs.vfat /dev/mmcblk1p1
```

## 49.6. Create EXT4 file system type for rootfs

```
target# mkfs.ext4 -b 4096 /dev/mmcblk1p2
```

## 49.7. Mount the eMMC partitions

```
target# mkdir -p /mnt/uboot/  
target# mount /dev/mmcblk1p1 /mnt/uboot  
target# mkdir -p /mnt/rootfs  
target# mount /dev/mmcblk1p2 /mnt/rootfs
```

## 49.8. Copy Rootfs and U-Boot to mount points

```
target# cp </path/to/ML0> </path/to/u-boot.img> /mnt/uboot  
target# tar -xvf /path/to/tisdk-rootfs-image-am572x-bel-mmrfic.tar -C /mnt/rootfs
```

## 49.9. Create a test file

- The file eMMC is created in the root directory of rootfs.
- This is to test whether the board has booted from eMMC and the flashing procedure was successful.

```
target# touch /mnt/rootfs/eMMC
```

## 49.10. Boot from eMMC

- Follow the below step to boot from eMMC:
  1. Power OFF the board
  2. Unmount the SD card from board
  3. Power ON the board
  4. Check for the eMMC file in root directory of file system

## 49.11. Expected result

- Board should boot from eMMC without SD Card.

# Chapter 50. TST049 Verify Loopback test on UART-n/ttyS(n-1)

## 50.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Loopback setup

## 50.2. procedure

1.Connect(Loopback) Carrier Board MB Connector UART-n/ttyS(n-1) PIN#(RX) & PIN#(TX) 2.Run the test as below 3.TX and RX data should be same

## 50.3. Expected result

```
root@am572x-bel-mmrfic:~# uart_test /dev/ttyS(n-1)
Writting Data.....
Device Reading .....
mmrfhello
Loopback Test Success
```

# Chapter 51. TST050 Verify GPIO(port\_pin) toggling

## 51.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Digital Oscilloscope

## 51.2. procedure

- formula:  $N = (\text{port}-1) \times 32 + \text{pin}$
- Export GIO(N)(MBC - X)
- Set direction to out.
- set a number to value file.

```
$ cd /sys/class/gpio
$ echo N > export
$ cd gpio(N)
$ echo out > direction
$ echo 1 > value
$ echo 0 > value
```

## 51.3. Expected result

MBC PIN#X should toggle as per the value written.

MBC PIN#X	GPIO(port_pin)	$N=(\text{port}-1) \times 32 + \text{pin}$
4 pin of MB connector	GPIO8_9	233
5 pin of MB connector	GPIO2_24	56
11 pin of MB connector	GPIO8_0	224
12 pin of MB connector	GPIO8_1	225
25 pin of MB connector	GPIO2_0	32
26 pin of MB connector	GPIO2_1	33
29 pin of MB connector	GPIO2_3	35
30 pin of MB connector	GPIO2_4	36
32 pin of MB connector	GPIO2_5	37
33 pin of MB connector	GPIO2_6	38
34 pin of MB connector	GPIO2_7	39
35 pin of MB connector	GPIO2_8	40
36 pin of MB connector	GPIO2_19	51
37 pin of MB connector	GPIO2_20	52
43 pin of MB connector	GPIO2_21	53
44 pin of MB connector	GPIO2_23	55
68 pin of MB connector	GPIO8_2	226
69 pin of MB connector	GPIO8_3	227
70 pin of MB connector	GPIO8_4	228
71 pin of MB connector	GPIO8_5	229
79 pin of MB connector	GPIO8_6	230

95 pin of MB connector	GPI08_7	231
96 pin of MB connector	GPI08_8	232
105 pin of MB connector	GPI02_28	60
106 pin of MB connector	GPI02_25	57
107 pin of MB connector	GPI02_29	61
110 pin of MB connector	GPI02_26	58
127 pin of MB connector	GPI02_27	59
131 pin of MB connector	GPI02_31	63

# Chapter 52. TST051 Verify GPIO PIN toggling

## 52.1. preconditions

- AM572X-BEL-MMRFIC board
- SD card
- Digital Oscilloscope

## 52.2. Procedure

- Pins to toggle
  - GPIO6\_8(U18#5)
  - GPIO6\_9(U18#7)
  - GPIO1\_2(R75)
  - GPIO3\_31(JP3#7)
- Formula to find GPIO number:  $N = (\text{port}-1) \times 32 + \text{pin}$
- Run this script with GPIO number as argument.
- Script: `sysfs.sh`

```
#!/bin/bash
echo $1 > /sys/class/gpio/export
cd /sys/class/gpio/gpio$1
echo out > direction
for i in {1..5}
do
    echo 1 > value
    sleep 2
    echo 0 > value
    sleep 2
done
```

## 52.3. Expected result

- Script output:

```
root@am572x-bel-mmrfic:~# pwd
/home/root
root@am572x-bel-mmrfic:~# chmod +x sysfs.sh
root@am572x-bel-mmrfic:~# ./sysfs.sh 168
root@am572x-bel-mmrfic:~# ./sysfs.sh 168
root@am572x-bel-mmrfic:~# echo 168 > /sys/class/gpio/unexport
```

- MBC PIN/Register should toggle as per the value written.

# Chapter 53. TST052 Verify AMBE GPIO PIN toggling

## 53.1. preconditions

- AM572X-BEL-MMRFIC board
- eMMC / NFS
- Digital Oscilloscope

## 53.2. procedure

- $N = (\text{port}-1) \times 32 + \text{pin}$
- Export  $\text{GIO}(N)$
- Set direction to `out`.
- set a number to `value` file.

```
$ cd /sys/class/gpio
$ echo N > export
$ cd gpio(N)
$ echo out > direction
$ echo 1 > value
$ echo 0 > value
```

## 53.3. Expected result

- AMBE PIN should toggle as per the value written.

AM5728	PIN NAME	- GPIO_ADDRESS	- Sysfs number	- IC pin number
A13	-B00T0	- GPIO6_4	- 164	- 32
B16	-B00T1	- GPIO6_7	- 167	- 33
AC10	-Reset	- GPIO6_13	- 173	- 42
E14	-ENC	- GPIO4_18	- 114	- 44
D11	-DEC	- GPIO4_19	- 115	- 45