



## CEC overview



# CEC overview

Stable: 15.04.2020 - 08:51 / Revision: 15.04.2020 - 08:46

CEC (Consumer Electronics Control) provides enhanced functions to control devices connected through HDMI. As an example, it allows the control of a recorder through TV remote control.

The CEC protocol is defined in the HDMI specification. This article gives information about the Linux<sup>®</sup>CEC framework.

## Contents

|   |          |
|---|----------|
| 1 Purpose .....                                     | 2        |
| 2 CEC overview .....                                | 3        |
| <b>2.1 Description of the main components .....</b> | <b>3</b> |
| <b>2.2 Description of the APIs .....</b>            | <b>3</b> |
| 3 Configuration .....                               | 4        |
| 4 How to use CEC framework .....                    | 4        |
| 5 Use cases .....                                   | 8        |
| 6 Generic source code location .....                | 8        |
| 7 References .....                                  | 8        |

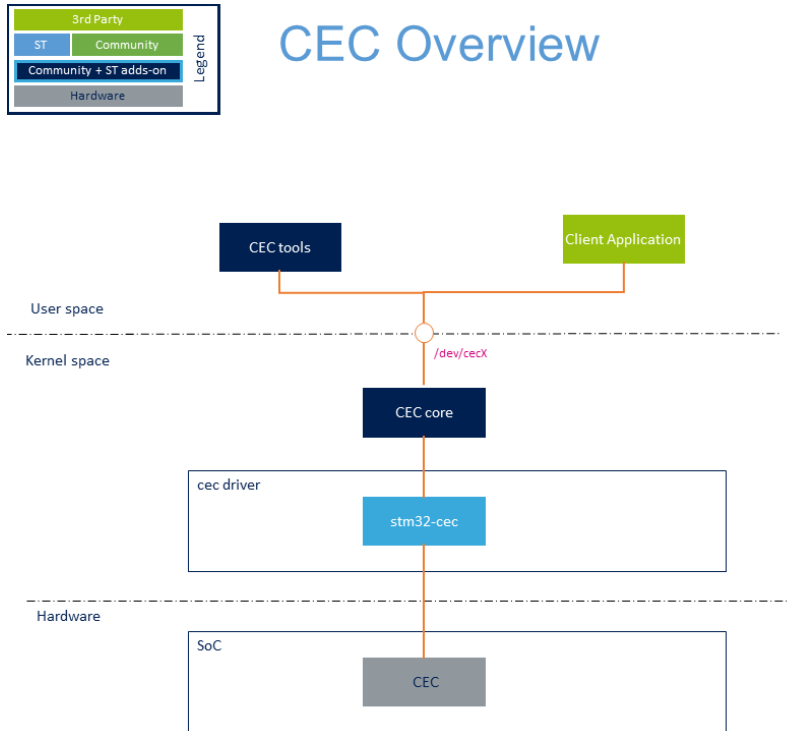
## 1 Purpose

The purpose of this article is to introduce the CEC framework:

- General information
- Main components/stakeholders
- How to use the CEC framework
- Use cases

CEC is an HDMI feature designed to command and control devices connected through HDMI by using only one remote control. As an example, the remote control of a television set can be used to control a set-top box and a DVD player. Up to 15 devices can be controlled. CEC also allows individual CEC-enabled devices to command and control each other without user intervention.<sup>[1]</sup>

## 2 CEC overview



### 2.1 Description of the main components

*From user space to hardware*

- **CEC tools** (User space)

This component contains a set of useful tools to configure CEC features. It is hosted in v4l-utils [2].

- **CEC core** (Kernel space)

CEC core is the standard Linux kernel CEC framework. It manages the CEC protocol.

- **stm32-cec** (Kernel space)

This is the ST CEC Linux driver that handles the CEC hardware block.

- **CEC** (Hardware)

This is ST microprocessor CEC hardware block.

### 2.2 Description of the APIs

The CEC kernel API is documented in the 'CEC kernel support' section of the Linux Kernel documentation [3].

The CEC Userland API is documented in the 'Consumer Electronics Control API of v4l-utils documentation' [4].



## 3 Configuration

By default, CEC is deactivated in ST deliveries. However, you can use Linux Menuconfig tool if a specific configuration is required: [Menuconfig](#) or [how to configure kernel](#) then select:

```
[*] Device Drivers --->
  [*] Multimedia support --->
    [*] HDMI-CEC support
    [*] CEC platform devices --->
      [*] STMicroelectronics STM32 HDMI-CEC driver
```

## 4 How to use CEC framework

cec-ctl and cec-compliance tools are hosted in CEC tools use space.

To register a CEC device on CEC bus, set its type and physical address:

```
{{Board$}} cec-ctl --tuner -p 1.0.0.0
Driver Info:
  Driver Name           : stm32-cec
  Adapter Name          : stm32-cec
  Capabilities          : 0x000000ef
    Physical Address
    Logical Addresses
    Transmit
    Passthrough
    Monitor All
  Driver version        : 4.19.9
  Available Logical Addresses: 4
  Physical Address      : 1.0.0.0
  Logical Address Mask  : 0x0008
  CEC Version           : 2.0
  Vendor ID             : 0x000c03 (HDMI)
  OSD Name              : 'Tuner'
  Logical Addresses     : 1 (Allow RC Passthrough)

  Logical Address       : 3 (Tuner 1)
    Primary Device Type : Tuner
    Logical Address Type : Tuner
    All Device Types    : Tuner
    RC TV Profile       : None
    Device Features     :
    None
```

To debug and monitor CEC messages on bus, use the following command line:

```
{{Board$}} cec-ctl -M
Driver Info:
  Driver Name           : stm32-cec
  Adapter Name          : stm32-cec
  Capabilities          : 0x000000ef
    Physical Address
```



```
Logical Addresses
Transmit
Passthrough
Monitor All
Driver version      : 4.19.9
Available Logical Addresses: 4
Physical Address   : 1.0.0.0
Logical Address Mask : 0x0008
CEC Version       : 2.0
Vendor ID         : 0x000c03 (HDMI)
OSD Name          : 'Tuner'
Logical Addresses  : 1 (Allow RC Passthrough)

Logical Address    : 3 (Tuner 1)
Primary Device Type : Tuner
Logical Address Type : Tuner
All Device Types   : Tuner
RC TV Profile      : None
Device Features    :
None
```

Initial Event: State Change: PA: 1.0.0.0, LA mask: 0x000

Run CEC compliance to check CEC bus typology:

```
{{Board$}} cec-compliance -A
cec-compliance SHA      : not available

Driver Info:
Driver Name              : stm32-cec
Adapter Name             : stm32-cec
Capabilities              : 0x000000ef
Physical Address         :
Logical Addresses        :
Transmit                 :
Passthrough              :
Monitor All              :
Driver version           : 4.19.9
Available Logical Addresses: 4
Physical Address         : 1.0.0.0
Logical Address Mask     : 0x0008
CEC Version              : 2.0
Vendor ID                : 0x000c03
Logical Addresses        : 1 (Allow RC Passthrough)

Logical Address          : 3
Primary Device Type      : Tuner
Logical Address Type     : Tuner
All Device Types         : Tuner
RC TV Profile            : None
Device Features          :
None

Compliance test for device /dev/cec0:

The test results mean the following:
OK                          Supported correctly by the device.
OK (Not Supported)         Not supported and not mandatory for the device.
OK (Presumed)              Presumably supported. Manually check to confirm.
OK (Unexpected)            Supported correctly but is not expected to be supported
for this device.
OK (Refused)               Supported by the device, but was refused.
FAIL                        Failed and was expected to be supported by this device.
```



## CEC overview

```
Find remote devices:  
  Polling: OK
```

### CEC API:

```
CEC_ADAP_G_CAPS: OK  
CEC_DQEVENT: OK  
CEC_ADAP_G/S_PHYS_ADDR: OK  
CEC_ADAP_G/S_LOG_ADDRS: OK  
CEC_TRANSMIT: OK  
CEC_RECEIVE: OK  
CEC_TRANSMIT/RECEIVE (non-blocking): OK (Presumed)  
CEC_G/S_MODE: OK  
CEC_EVENT_LOST_MSGS: OK
```

### Network topology:

```
System Information for device 0 (TV) from device 3 (Tuner 1):  
  CEC Version           : 1.4  
  Physical Address      : 0.0.0.0  
  Primary Device Type   : TV  
  Vendor ID            : 0x00903e  
  OSD Name              : 'TV'  
  Menu Language        : fre  
  Power Status         : On
```

```
Total: 10, Succeeded: 10, Failed: 0, Warnings: 0
```

### Send a string to OSD:

```
{{Board$}} cec-ctl --set-osd-string=disp-ctl=0x00,osd="hello" --to tv
```

#### Driver Info:

```
Driver Name           : stm32-cec  
Adapter Name         : stm32-cec  
Capabilities          : 0x000000ef  
  Physical Address  
  Logical Addresses  
  Transmit  
  Passthrough  
  Monitor All  
Driver version       : 4.19.9  
Available Logical Addresses: 4  
Physical Address     : 1.0.0.0  
Logical Address Mask : 0x0008  
CEC Version          : 2.0  
Vendor ID            : 0x000c03 (HDMI)  
OSD Name             : 'Tuner'  
Logical Addresses    : 1 (Allow RC Passthrough)  
  
  Logical Address     : 3 (Tuner 1)  
  Primary Device Type : Tuner  
  Logical Address Type : Tuner  
  All Device Types    : Tuner  
  RC TV Profile       : None  
  Device Features     :  
  None
```

```
Transmit from Tuner 1 to TV (3 to 0):
```

```
CEC_MSG_SET_OSD_STRING (0x64):  
  disp-ctl: default (0x00)  
  osd: hello  
  Sequence: 347 Tx Timestamp: 2626.350s
```



Get topology:

```

{{Board$}} cec-ctl -S
Driver Info:
  Driver Name           : stm32-cec
  Adapter Name         : stm32-cec
  Capabilities         : 0x000000ef
    Physical Address
    Logical Addresses
    Transmit
    Passthrough
    Monitor All
  Driver version       : 4.19.9
  Available Logical Addresses: 4
  Physical Address     : 1.0.0.0
  Logical Address Mask : 0x0008
  CEC Version          : 2.0
  Vendor ID            : 0x000c03 (HDMI)
  OSD Name              : 'Tuner'
  Logical Addresses    : 1 (Allow RC Passthrough)

    Logical Address     : 3 (Tuner 1)
    Primary Device Type : Tuner
    Logical Address Type : Tuner
    All Device Types    : Tuner
    RC TV Profile       : None
    Device Features     :
      None

System Information for device 0 (TV) from device 3 (Tuner 1):
  CEC Version           : 1.4
  Physical Address      : 0.0.0.0
  Primary Device Type   : TV
  Vendor ID             : 0x00903e (Philips)
  OSD Name              : TV
  Menu Language         : fre
  Power Status          : On

Topology:

0.0.0.0: TV
1.0.0.0: Tuner 1

```

Send a standby message to a tv:

```

{{Board$}} cec-ctl --standby --to tv
Driver Info:
  Driver Name           : stm32-cec
  Adapter Name         : stm32-cec
  Capabilities         : 0x000000ef
    Physical Address
    Logical Addresses
    Transmit
    Passthrough
    Monitor All
  Driver version       : 4.19.9
  Available Logical Addresses: 4
  Physical Address     : 1.0.0.0
  Logical Address Mask : 0x0008
  CEC Version          : 2.0
  Vendor ID            : 0x000c03 (HDMI)
  OSD Name              : 'Tuner'
  Logical Addresses    : 1 (Allow RC Passthrough)

```



```
Logical Address      : 3 (Tuner 1)
Primary Device Type  : Tuner
Logical Address Type : Tuner
All Device Types     : Tuner
RC TV Profile        : None
Device Features      :
None
```

```
Transmit from Tuner 1 to TV (3 to 0):
CEC_MSG_STANDBY (0x36)
Sequence: 371 Tx Timestamp: 2747.236s
```

## 5 Use cases

Applications must be added/developed in userland to support all the use cases related to CEC (eg recorder controlled by TV remote control) .

## 6 Generic source code location

- CEC core
- CEC API
- stm32 CEC driver

## 7 References

- [Wikipedia article about CEC](#)
- [CEC tools source code](#)
- [Linux CEC Kernel Support](#)
- [CEC Userland API](#)

Consumer Electronics Control (HDMI standard)

High-Definition Multimedia Interface (HDMI standard)

Application programming interface

Secure Hash Algorithm