1)

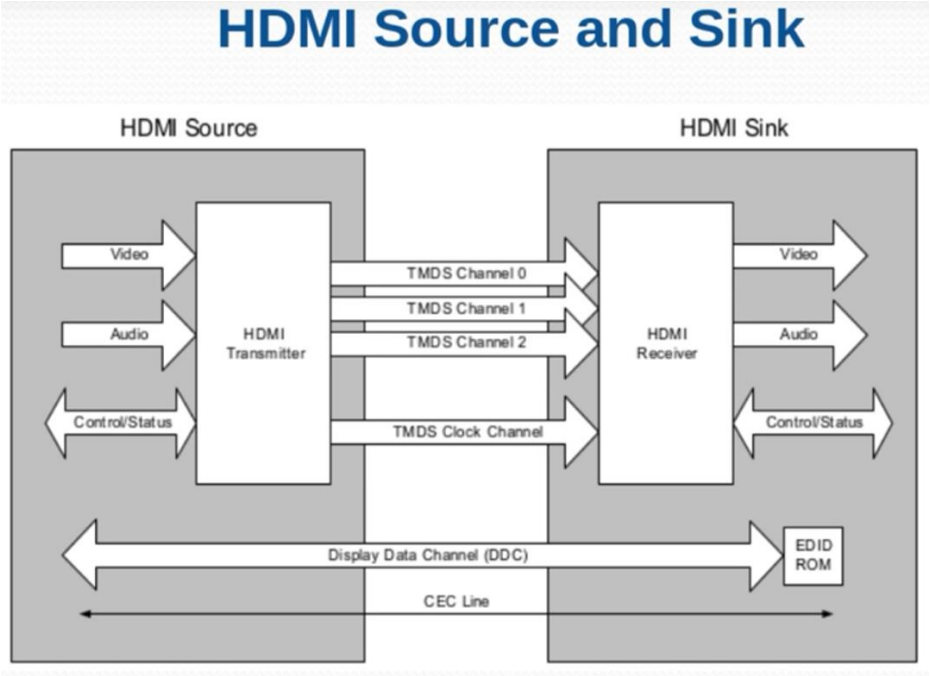
let's first understand what was the interface available before HDMI. So this is like you can see in the diagram this is DVI is digital video interface. So this was the interface which we were using before HDMI for transmitting the video data. So this was the interface which was able to transfer only the video. If you want to transmit the audio, you have to use extra cables for transmitting the audio data. And also if the size of cable will increase. Or you can say the length of cable will increase. It was causing the quality loss. So that was the limitation of DVI. It's like we need extra cable for transmitting audio data. With that, if the length of cable was increasing, the quality of data was degrading. So that was the bottleneck of our interface.

And you know, like nowadays there is a need of transmitting data of high definition data, which is like if you talk about video, if you talk about photo, if you talk about any digital media stuff. So all are basically working upon now to give you the data. So to transmit data you need an interface which can transmit the data at very high rate and with very good quality.

So that is why a group of companies came together and they developed an HDMI interface which is high definition multimedia interface. And this was developed in 2002. And after that they have sold a lot of volume. And they, they saw a market capture of I think a great market capture was there for HDMI actually. So HDMI is basically an interface for transmitting compact audio video interface for transmitting the data. And it is backward compatible with DVI, means it is backward compatible with the interface which is which was available before HDMI like DVI and HDMI basically transmit uncompressed video data and compressed or uncompressed audio data from one HDMI compliant devices to the other HDMI complete sync device.

And so now, you know, like why is popular because hdmi is support very high range of data rate transmission, which is good for our digital devices because nowadays the data demand of data is increasing. So it means the demands of data rate is also increasing. Actually we need a interface which can transmit data at very high rate.

so far we have came up came across multiple types of HDMI. You can see in this diagram we have type A, type B, type type D, type E. So these all are different types of HDMI. And the difference between all these HDMI is like number of pins. Basically they have that will vary okay. So why these many types are given by the HDMI. It's like you might have multiple devices and that multiple devices might have different kinds of connector to give the support. All these kinds of connectors we have given the support of type A, B, C, D and E okay.



So the device which is transmitting the data that is called HDMI source, you can take example your mobile or you can take example your laptop. Or you can take example of any of the digital media embedded system devices, which is basically generating some data which you want to display onto some HDMI or any other sync devices connected to the sync system.

So HDMI sink is basically the device which is receiving the data from HDMI source, and it will basically display the data over your display interface. It can be HDMI LED, it can be HDMI TV, or it can be any display.

So to go in deep you can see here like this is HDMI transmitter in the HDMI source which has video data coming, audio data coming, control data coming and these data is basically transmitted by using Tmds channel okay. So for transmitting the data we use Tmds channel. So we have four three tmds channel, one clock channel. And this clock channel will basically operate like basically operate at the clock rate which can support the high transmission data rate. It means if you want to, if you want to basically increase the data rate of the data transmission between your source and sink, you have to increase the clock rate, because your clock rate will basically define the data rate you are going to transmit between source and sink device.

On the receiver side you have HDMI receiver. Then it will forward the data. It received video audio control status. And in this transmitter you have multiple blocks. In this receiver you have multiple block. We don't need to go into details of each of the block actually, because we are going to discuss that into the into the coming sessions.

So the data which we are sending video will be uncompressed video and for audio we can send compressed as well as audio uncompressed data.

Other channel we have DDC interface. So this interface we have between our source device and the sink device basically. And it is used for display data channel. It means like our sink device has a lot of information stored in their EDID ROM.

What is the mean by EDID ROM? It's like all sink devices, let us say monitor, display, monitor, LCD monitor. All these sink devices will have small part of memory, inbuilt memory which will have information related to sink device.

Actually this information tells about the capability or the vendor and the developer option for your Display. Let's say you have connected a monitor which is coming from Dell Company. So your data will tell to the source device.

It will tell monitor model number, manufacturer like Dell and capabilities I can support resolution up to 4K or 8K or maybe 2K anything.

So why the source device need need this information. Because a source device can be connected to multiple multiple types of HDMI sync device. You can connect it to a Display which can support up to 4K, 2K or display which can support only VGA. So in that case, your source device has to know which kinds of data it has to give to the display so that display will be able to display it.

Let's say your display support only data, and your source device is sending the high resolution data 4K data to data. So your display will not be able to display it right. So to overcome this issue, what we do is our source device. Basically read this information by using DDC channel. Channel is nothing but an I2C clients.

So two I2C clients are there. So it works in the same way as I2C works. So we have data line and the clock line. So we read this information and then we pass this information. And after passing this information basically we understand the capability of sync device. And then we basically set the resolution of transmission from source to sync according to the information which we received from the EDID ROM of sync device.

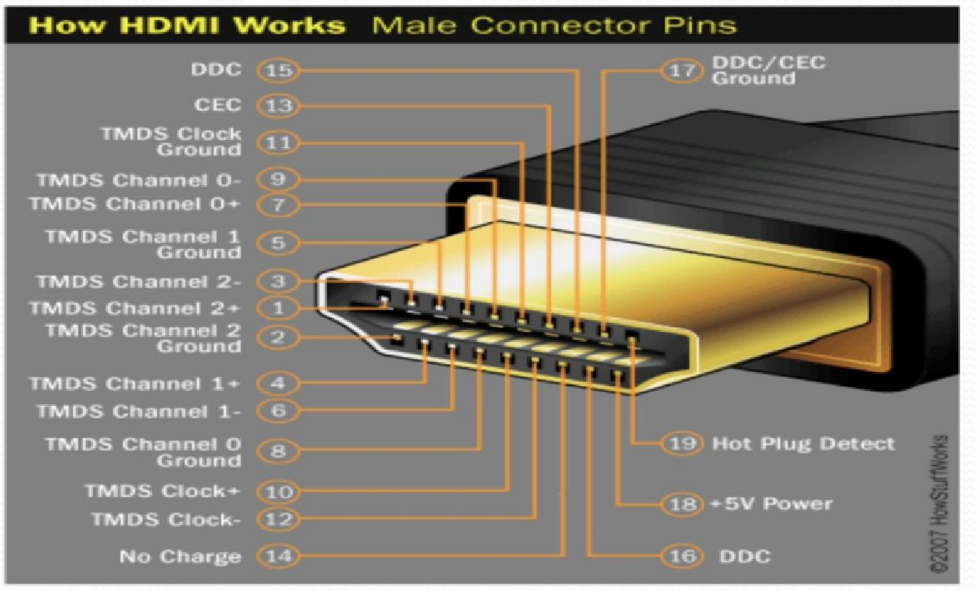
Then we have CEC line. CEC line is basically a mechanism by which you can control multiple HDMI compliant devices together. Let's say you have one remote, and by using one remote, you can control your TV as well as your compliant speaker or maybe projector or maybe any other display connected to it.

So, CEC line, basically, it's a it's a line for controlling the audio visual devices, which is connected over the network. When I say network, it means network of HDMI devices connected together.

HDMI Source and Sink

- **DDC** : To read EDID information from Sink Device.
- **TMDS:** Carry Video and Audio Auxiliary data at TMDS clock rate.
- **CEC:** High level control function across audio visual products.

HDMI Pin Assignment



we have multiple types of HDMI and each HDMI has different number of pins. HDMI type B basically we have total 19 pins okay. You can start from here from basically from 1 to 12. Basically, you have a tmds channels, pins.

So you have channel zero, you have channel zero minus plus basically negative positive negative positive all this you have and then you have ground as well. So these pins as I mentioned earlier is related to the channel which will be actually using for transmitting the data between your source device and the sink device.

And one pin you have, you can see here this is we have for DDC/CEC ground and this is for we have DDC 15. So 15 pin is a line for transmitting the for basically reading the data from the sink device as mentioned in the previous slide.

So for ground basically we use for like for ground line for DDC/CEC ground line 17 for controlling the devices.

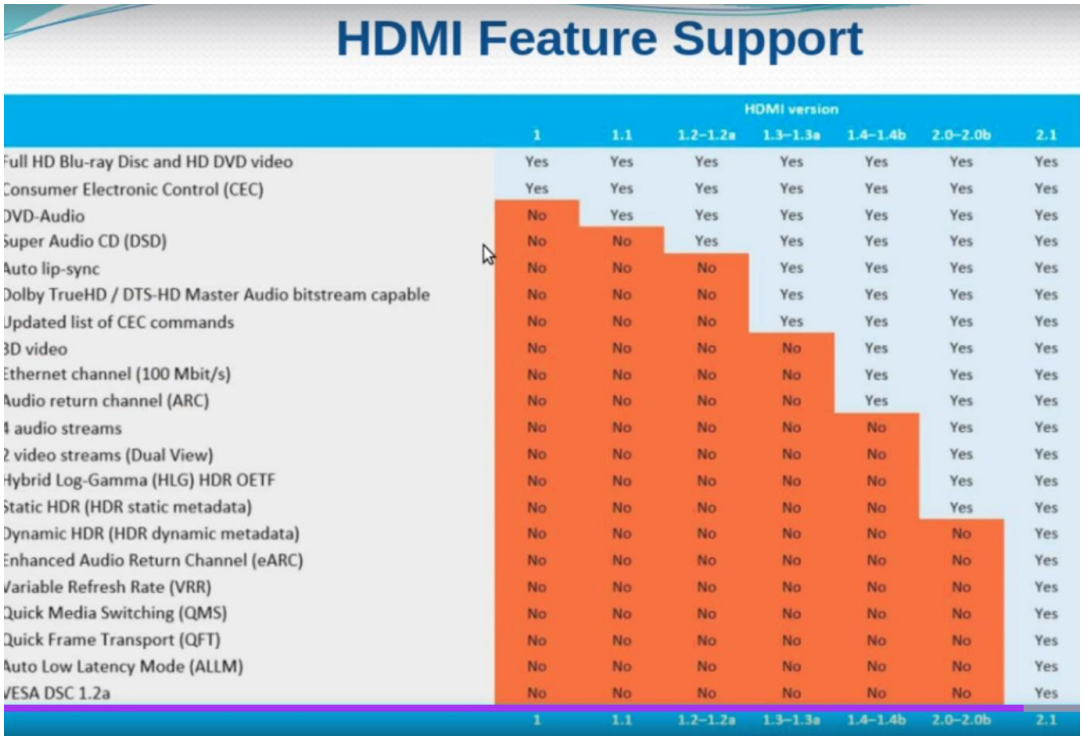
And then pin number 14 it has it has no charge basically.

Then pin number 19 is your hot plug detect. Hot plug detect is the mechanism by which you can detect if any of the sink device is connected to the source.

hot plug detect line basically which operates at five volt power supply and this supply is given by your source device. So whenever you plug any of the sink device with your source device, it completes the circuit. And it it basically generate one end up to the source device okay. So I am connected okay. So the source device will come to know okay. So there is one devices get connected.

So hot plug detect is basically detection at the runtime whether some device is connected or not. And this plus five volt power supply is there like which will be used to give power supply to the sink device by using of this pin.

DDC as I mentioned, like DDC is kind of I2C based protocol. So we know we need two lines. One is uh uh, one is data line, another is the clock line. So this 15 and 16 pin is dedicated to DDC transmission.



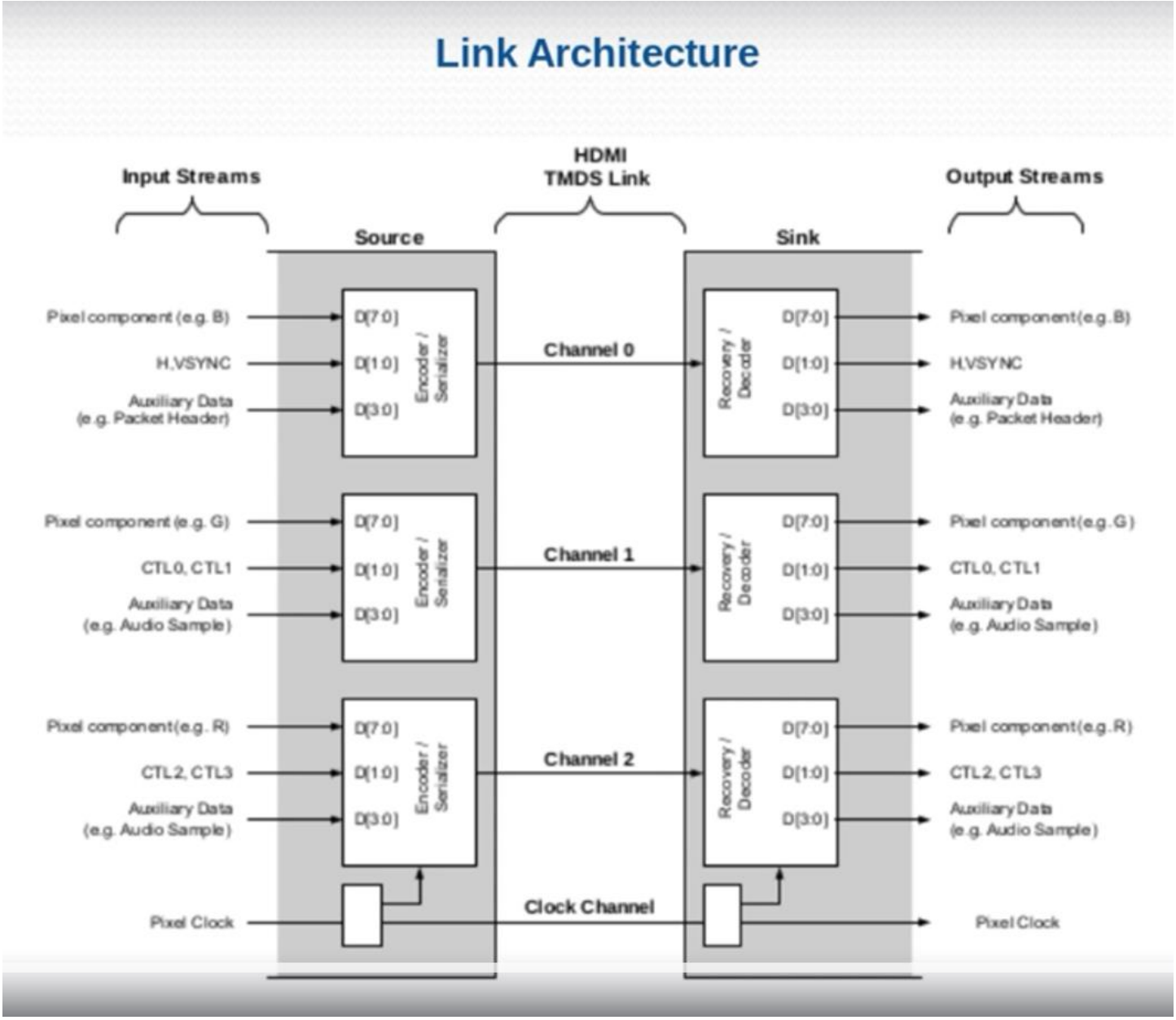
HDMI Feature Support

	HDMI version						
	1	1.1	1.2–1.2a	1.3–1.3a	1.4–1.4b	2.0–2.0b	2.1
Full HD Blu-ray Disc and HD DVD video	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Consumer Electronic Control (CEC)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
DVD-Audio	No	Yes	Yes	Yes	Yes	Yes	Yes
Super Audio CD (DSD)	No	No	Yes	Yes	Yes	Yes	Yes
Auto lip-sync	No	No	No	Yes	Yes	Yes	Yes
Dolby TrueHD / DTS-HD Master Audio bitstream capable	No	No	No	Yes	Yes	Yes	Yes
Updated list of CEC commands	No	No	No	Yes	Yes	Yes	Yes
3D video	No	No	No	No	Yes	Yes	Yes
Ethernet channel (100 Mbit/s)	No	No	No	No	Yes	Yes	Yes
Audio return channel (ARC)	No	No	No	No	Yes	Yes	Yes
4 audio streams	No	No	No	No	No	Yes	Yes
2 video streams (Dual View)	No	No	No	No	No	Yes	Yes
Hybrid Log-Gamma (HLG) HDR OETF	No	No	No	No	No	Yes	Yes
Static HDR (HDR static metadata)	No	No	No	No	No	Yes	Yes
Dynamic HDR (HDR dynamic metadata)	No	No	No	No	No	No	Yes
Enhanced Audio Return Channel (eARC)	No	No	No	No	No	No	Yes
Variable Refresh Rate (VRR)	No	No	No	No	No	No	Yes
Quick Media Switching (QMS)	No	No	No	No	No	No	Yes
Quick Frame Transport (QFT)	No	No	No	No	No	No	Yes
Auto Low Latency Mode (ALLM)	No	No	No	No	No	No	Yes
VESA DSC 1.2a	No	No	No	No	No	No	Yes
	1	1.1	1.2–1.2a	1.3–1.3a	1.4–1.4b	2.0–2.0b	2.1

And then this diagram is for feature support. As I mentioned like from the starting of development, it's like from 2002 until now we have came across different versions. And with each versions we are basically enhancing the feature supported by HDMI.

If I talk about feature, it can be like 3D data. It can be like network speed of data at the network and then basically audio return channel how many audio return channels it have. Then HDR, dynamic SDR, static HDR and then refresh rate. Many stuff you can see here we have different versions of till now 11.11.22, 1.33, 1.44822 02B and then 2.1. So these are the feature basically supported by our HDMI feature support. If you will see our latest HDMI 2.1 is supporting dynamic HDR, static HDR and then HDR Dolby basically two stream can be basically two stream can travel at a at a time. It means you are going to use some like two channels for two channel, for one stream and other channel for other stream. And then you have audio return channel. Audio return channel means like audio can be transmitted in the reverse side, like on the speakers or like that.

2)



In this diagram you can see we have one source device and we have other sink device. These source and sink device are connected through four channels. Three channels are the channels for data transmission and other is the clock lane.

So these channels are nothing but the tmds links actually or the Tmds channel okay.

Input stream is the data which we are sending from our development board to the sink device. Sink device can be like your monitor or your display device where you are basically projecting your data which is the output streams actually.

If you will see here, we are transferring the pixel component which is have the size of eight bit. Then you are sending HVsync Hsync or Vsync. Then you are sending auxiliary data. Auxiliary data is related to your audio data.

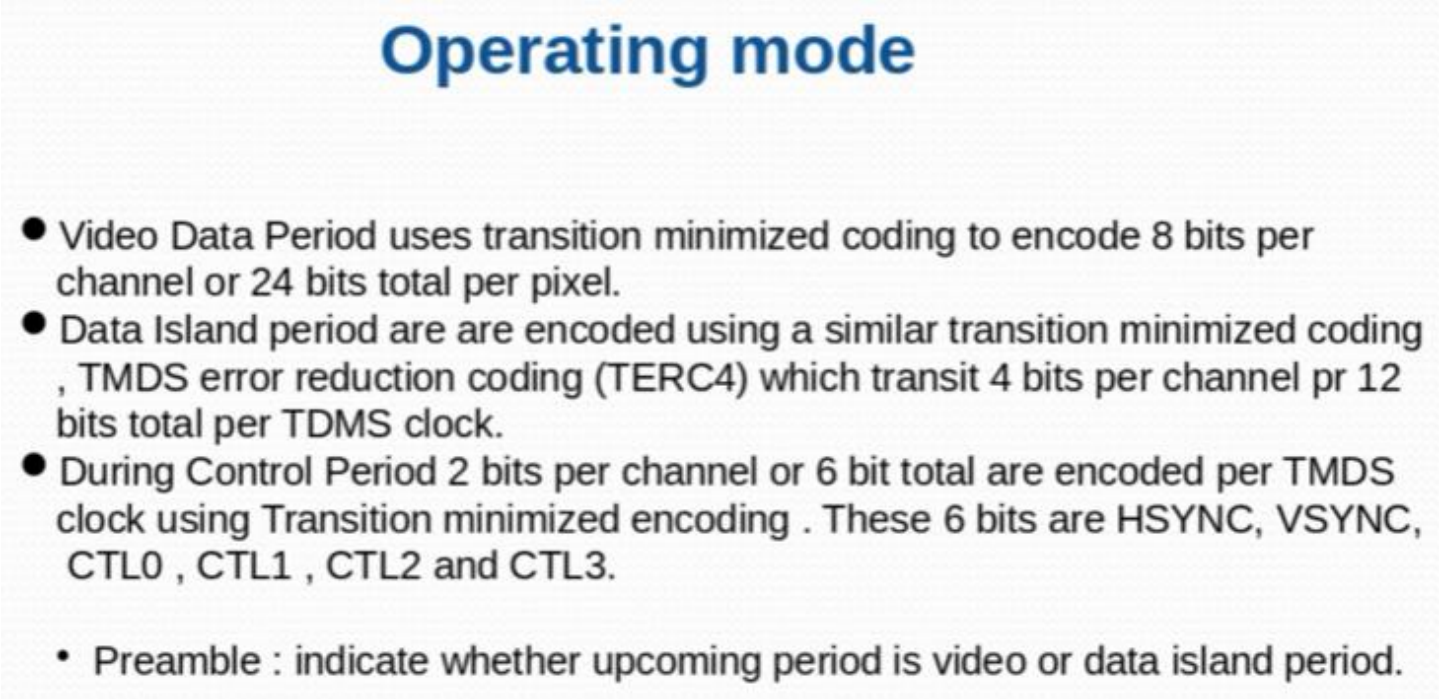
Hsync and Vsync are the control data and pixel component is your video data.

Basically HDMI operates in three mode.

So first mode is a video data period. Second is data island period. And third is a control period.

So video data period is the period basically where you transfer the actual video data pixel on that channel and tmds channel basically use transition minimized differential encoding basically, which makes the transfer of data at very good quality and without any quality loss.

Data island period is the period where you transfer the audio packets. And the control period is the period basically where you transfer the control information like Hsync, Vsync and other data related to controlling of the transmission happening between source device and the sync device.



The first operating mode as we discussed like the video data period. So video data period basically use transition minimized coding to encode eight bits per channel or 24 bits total per pixel.

It means like on per channel or on per TMDS line we are going to transfer eight bits per tmds clock, eight bits per channel per tmds clock. If you count total bits, so 24 bits basically are going to transfer at per clock of tmds.

This basically bit per pixel can vary actually depending upon the color depth you have, you have set in your HDMI. So you can transmit low rate of data as well. But generally this is the basically the most used way of communication.

So generally we transmit 24 bits of data per tmds clock actually.

And data island period are encoded using a similar transition minimized coding error. So data island period is going to transmit four bits per channel. Video were transmitting eight bits per channel. That island period is transmitting four bits per channel. It means 12 bits per TMDS clock or 12 bits of all channels.

During control period, basically you are sending two bits per channel or six bits totals are encoded per tmds clock using transition minimized encoding.

These six bits are basically control bits. As I talked about this can be accessing HSYNC, VSYNC, CTL0, CTL1, CTL2, CTL3 anything. These bits basically make a preamble. Preamble basically indicates whether the upcoming packet is your data island packet or it's a video packet.

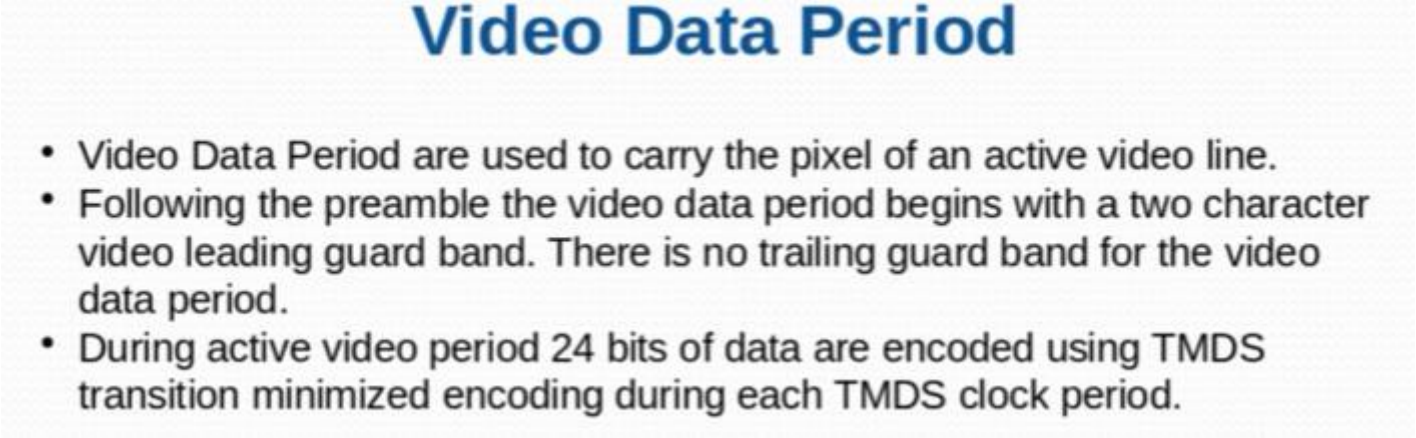
It means it will tell whether the upcoming period is Data island period or it's a video period.

pixel component is part of video data. So that's why we are transmitting eight bit. we have the source side. We have encoder serializer. So we will encode the data and we will send through some of the tmds channel and at the receiver end we will be having our receiver that will decode the data, and then it will give it to the display or whatever the output you want to display.

Other thing you have HSYNC. So as we mentioned like control period transmit two bits of data on per channel. So we are sending bit zero and bit one. In the same way we have auxiliary data. This is your packet header and it can be audio sample basically.

So as we mentioned like in the audio data island period, basically we are going to transmit four bits per channel. So that's why you can see here we are sending the bit from 0 to 3 and the same is being received at the sink side okay.

So all the data transmission rate will depend upon the tmds clock rate. Because we are going to transmit the data at the time of clock is on its peak.



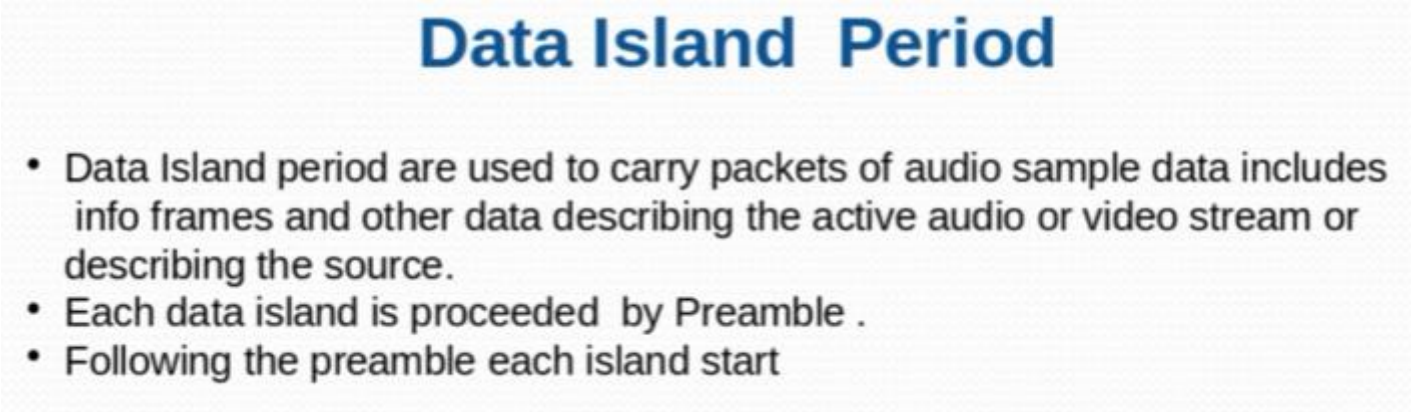
video data period is used to carry the pixel of an active video line. Following the preamble, as I mentioned, the preamble will tell whether the upcoming packet that is that will tell whether the upcoming packet is belongs to that island period or from the video period.

So following the preamble, the video data period begins with a two character video leading guard band. So after preamble we have we will get to two character of data which will basically guard the video data.

Guard means like it is boundary of your video data basically. So at the sink side, preamble will tell video data is being received. So the first two bit which will receive that we have to discard because that is the guard bit that bits doesn't belong to the video data, but there is no trailing guard band for the video data period.

And as we mentioned during active video period, we transmit 24 bits of data are encoded using Tmds (transition minimize encoding) during each tmds clock period.

Data Island Period are used to carry packets of audio sample data. Include info frames and other data describing the active audio or video stream or describing the source. So basically data island period you can consider most of the time it will be transmitting the audio packets. But with that it will also transfer transmit some of the metadata.

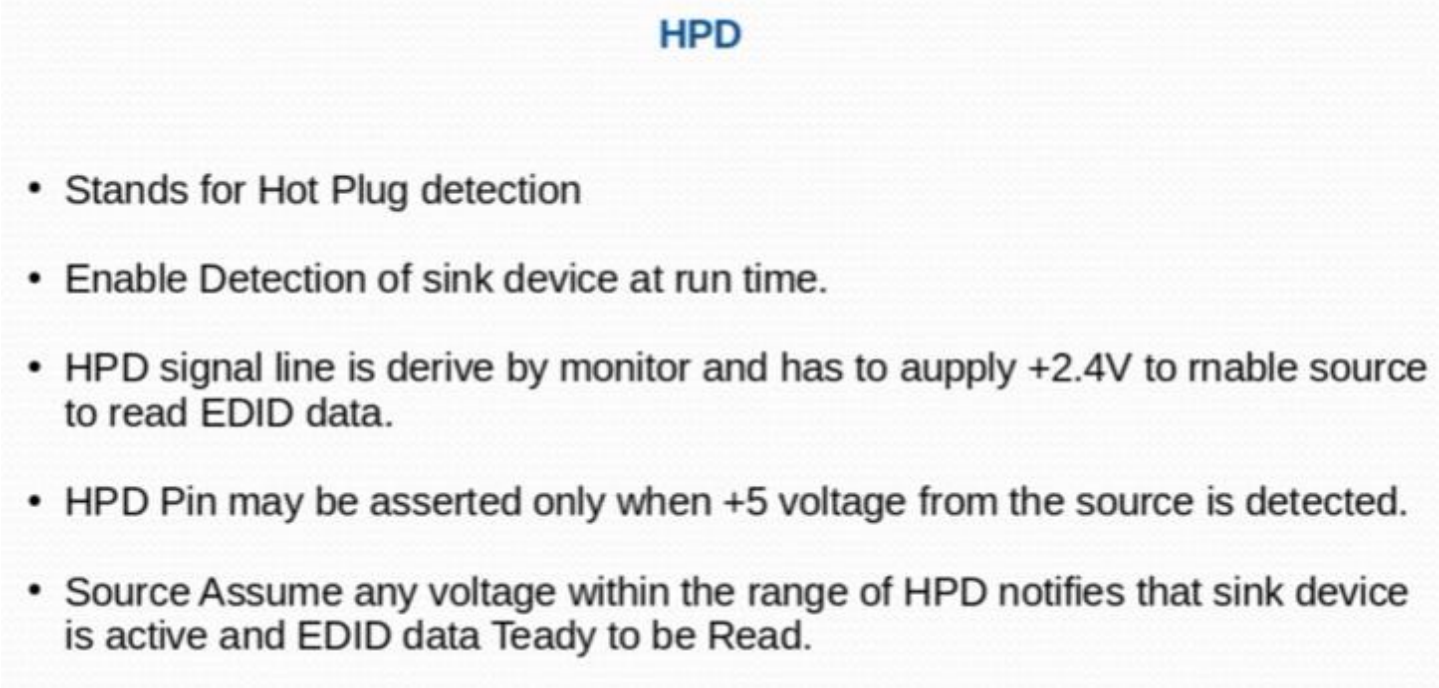


Metadata means data about data like info frames and all other stuff. describing the source basically because display like sync device also has to know information about source device. So that all information we are going to send through data Island period.

And then basically each data island is preceded by preamble. As you mentioned, in case of video preamble bits will tell the actual coming mode. Following the preamble, Each island starts so there will not be any guard bit. So after preamble directly the data Island period gets started.

You can download HDMI 1.3 specifications PDF is there on the Google. Later versions like 2.1 is not available on internet.

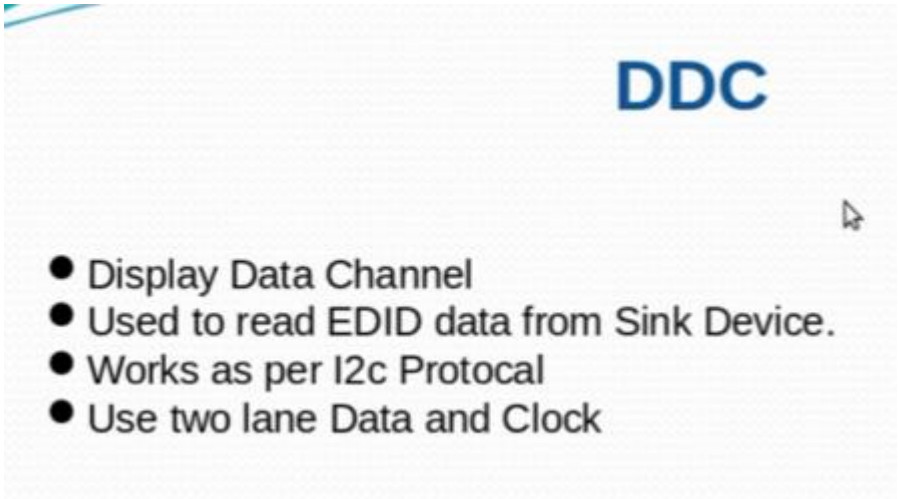
3)



what happen is you must have seen like whenever you are connect HDMI of your laptop with any of the display device. So at the runtime your display basically get detected and your laptop start transmitting data to your display device or the monitor which you have connected. when you insert any device to the source device. So in this case which the example which you have discussed. So there our laptop is behaving like a source device and monitor which you are going to connect with your laptop that is behaving like a sync device.

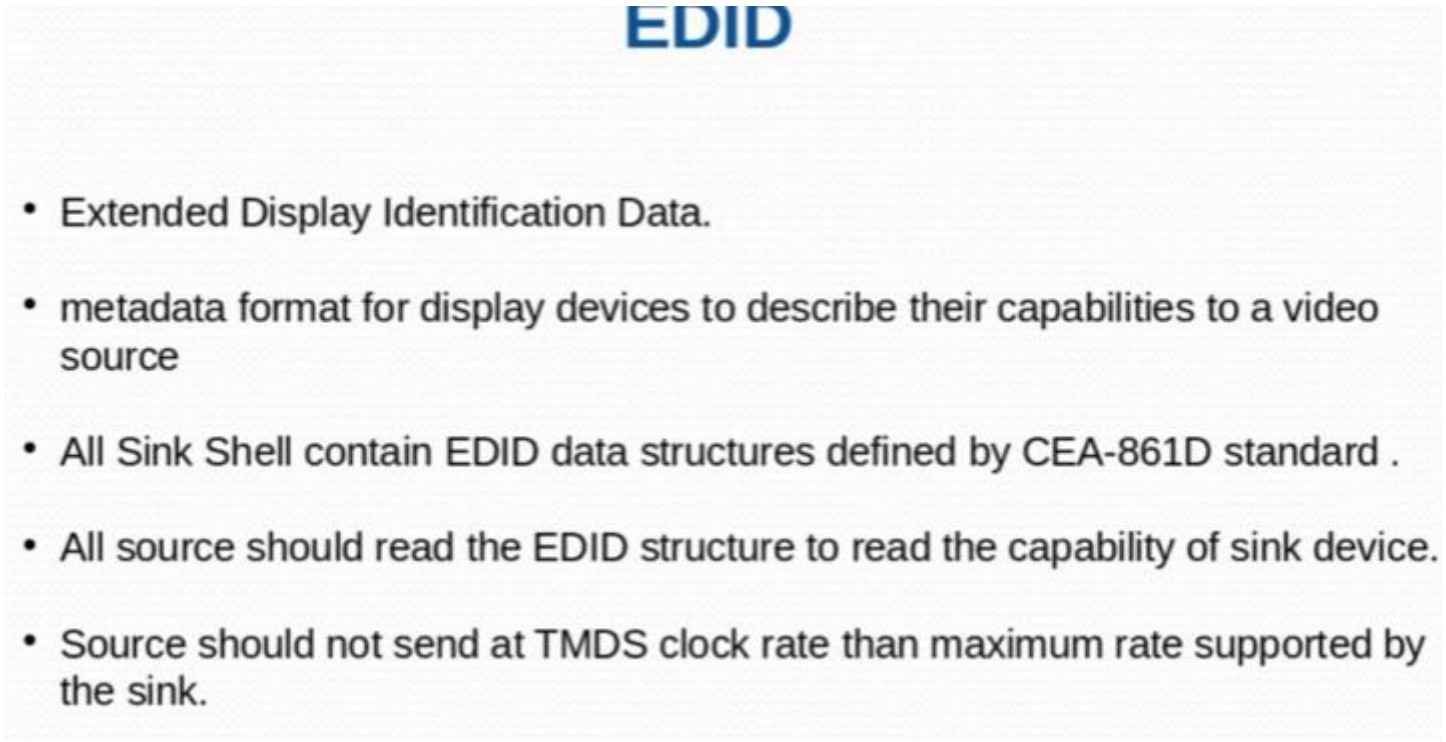
So when you basically insert HDMI cable to the HDMI port of your laptop, so that basically generate An HPD event and how HPD event is generated. So HPD signal line is derived by monitor and has to supply 2.4 voltage to enable source to read EDID data. So what does it mean? It means like when you will connect the HDMI cable to your laptop. So your monitor will be supplying 2.4 voltage power supply. which basically enable your source device. Source device is laptop to read the EDID data. HPD pin may be asserted only when five voltage From the sources detected. It means like the HDMI cable, which we have connected between our laptop and the monitor. So there is one line of five voltage power supply. So that power voltage of five voltage that has to be given by our source device to the sink. sink, and when sink will be deriving the 2.4 voltage of power At that time your source device will come to know So there is sink devices connected. And and we are ready to read the EDID data from the sink devices ROM.

Source assume any voltage within the range of HPD notifies that sync devices is active and EDID data is ready to be read. There is a range defined for HPD. it's a voltage range. Basically it can be like 5 volt to 6V, something like that. So when the voltage will be there in that range. So in that case your source device will be notified. So there is some event and you are ready to go to read the data from or sorry from the sync device. So the users of HPD already know like without HPD how your source device will come to know something is connected to it.



So after HPD event, we have to read the EDID data from sync device. So to read the data from the sync device, we should have some kind of interface by which we can read the data. So basically defines a two lane protocol - DDC. That is the basically stands for display data channel. It is used to read EDID data from sync device. So once your sync device is powered up and your source device is already powered up, and once the handshaking mechanism and handshaking mechanism means is happened and your device is using device already signalled to source, you can read the EDID information. So at that time, our source device will start reading the information from your sync device. we are going to use DDC channel. As I said it's a display data channel which use I2C protocol.

So. The way I2C protocol works the same way DDC works. One lane is a data lane. Another lane is the clock lane. ecause the data limit and the data rate is not so high actually. So that way we that's why we use to see basically for reading the data from sync device.



EDID is basically extended display identification data. we need to read some data from sync devices after getting the event.

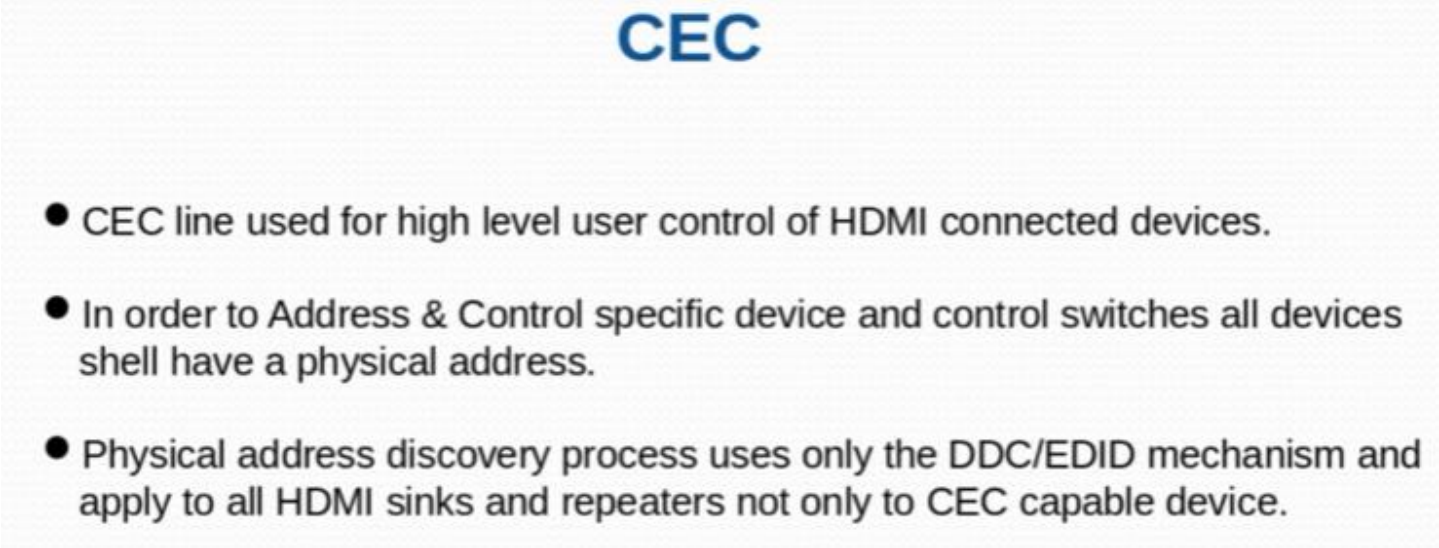
So we read the EDID data. This is a metadata format for display devices to describe their capability to a video source. So it means like your EDID data will contain the information about your devices. So laptop has to come to know like which monitor is connected with me and what is the capability of the monitor which is connected with me. When I say capability it means what are different resolution it supports and what are the data, what data rate it can support and refresh rate it can support lots of information we have to receive from the sync device. so that that all information basically is stored in the EDID structure.

So there are predefined structures type basically data structures which is defined by standard which is called CEA-861D. All source should read this data structure to read the capability of sync device. then source should not send a TMDS clock rate than maximum rate supported by the sync. It means like we have to know the capability of sync device. Then we are going to transmit the data at the rate which can be supported by our sync device.

So let's say if your monitor is supporting only video data and you're HD transmitting the data. So that will be of no use for your monitor actually, and it will not be able to support it. So that is why we have to know the capability before basically transmitting the data. And there is a mechanism which is called calibrations of resolutions. It means like our source device maintain a list of resolutions like 4K, 2K, VGA etc. Then it basically go and read the structure from the EDID sink device.

And it come to know okay. So this is the capability of your monitor. This is the capability of your sync device like this. This is the maximum resolution your display can support or your monitor can support. Then basically source device will go through the calibration table calibration resolution table, whatever it has maintained. And it will pick up the highest matching resolution from the list. And then it will start transmitting the data on that resolution so that your monitor connected to laptop can display the data.

4)



CEC stands for Consumer Electronic Control. CEC line basically used for controlling high level user control of connected device okay. It means like by using CEC lines, you can control multiple HDMI connected devices connected in a network. It's like a network of devices connected together.

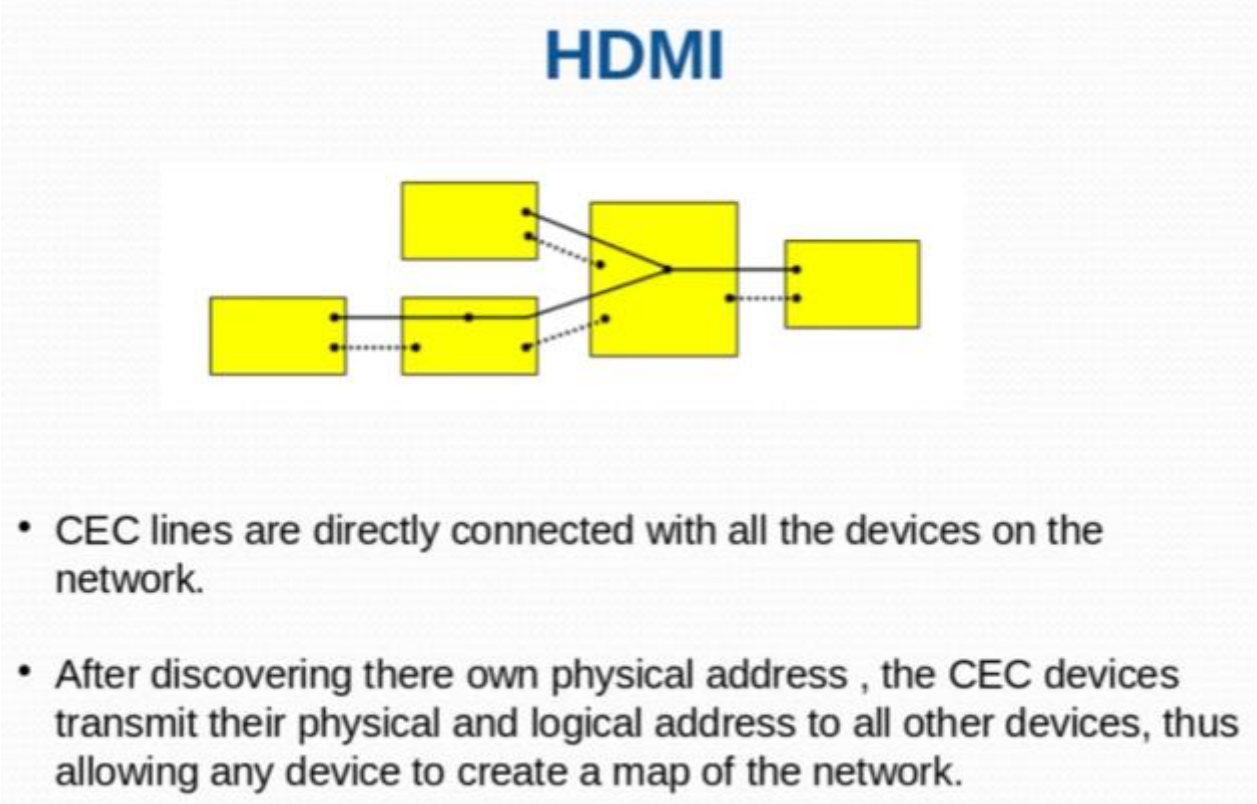
You might have your HDTV connected with your with your DVD player. And you might also have a speaker connected to it. So CEC enable us to control all these devices connected together by using one single remote controller. For example, by using a single. A remote of your HDTV. You can control your set top box, your TV, as well as other devices as well. So this feature is possible because of line.

Okay, so in order to basically address and control specific devices and control switches, all switches, all devices shall have a physical address. It means the devices which are connected in a network or cluster. Each device, each HDMI completely compliant device should have one physical address by which your remote control can identify the device it's want to communicate.

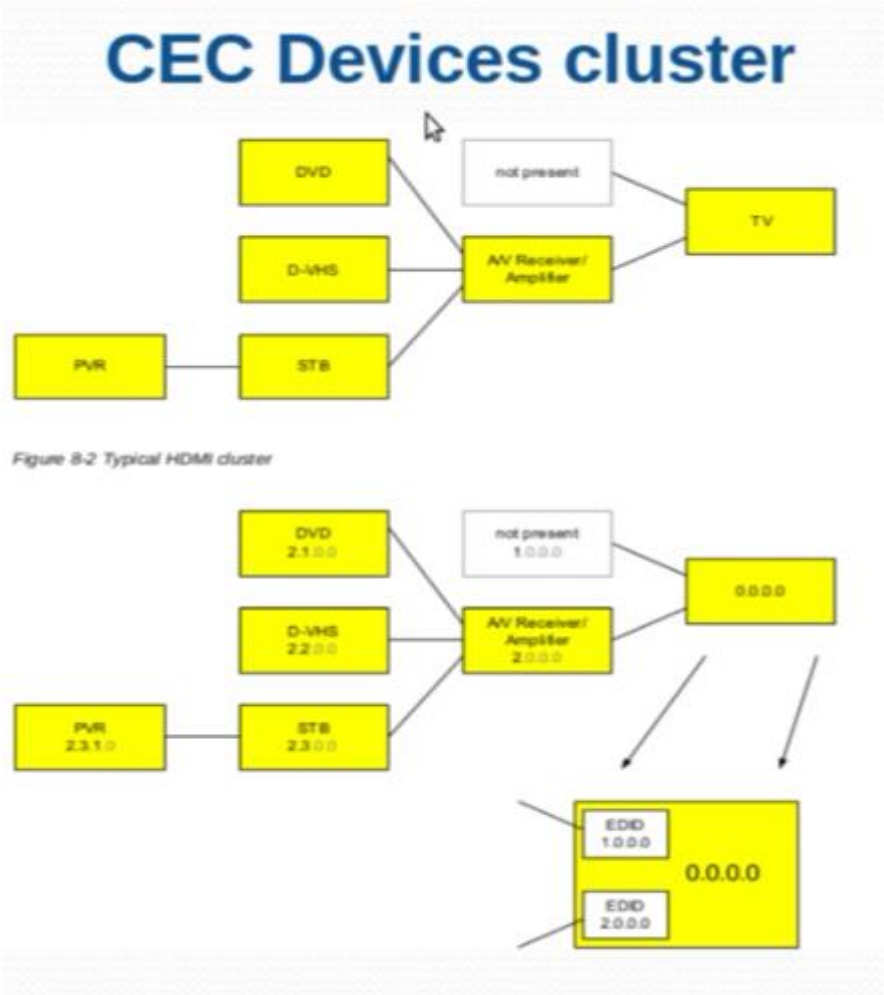
Actually, it works like a physical address discovery, process, use or mechanism to understand or to learn the addresses of other devices connected in the network. And after that CEC will send the control commands to control the other devices. And CEC line is basically one of the pins or the lines provided in HDMI cable basically. And this is bi directional cable. And for sending the control commands, we use this CEC line for sending the commands to the HDMI devices connected.

And this works similar to I2C protocol, but the difference between protocol and CEC line transmission is like transmission of data protocol is that I2C is a two wire protocol. But in case of CSI, we use one wire for both direction data transmission actually. And it's a bi directional. And the other thing is like in in case of I2C to see generally for the we use mostly we use seven bit address scheme. But in case of CEC we mostly use seven bit address scheme actually.

So the basic idea is CEC is used to control the devices connected in an HDMI compliant devices network. And we use a single wire to control the devices connected. And each device has their physical addresses by which we identify the devices connected on the network.



This linking is basically connecting all the devices together in a network. And after discovering the own physical address so what does it means. It means like each device will have their own physical address. And after discovering their own physical address, the CEC device transmits their physical and logical address to other devices. It will transmit all the physical and logical logical devices to all the other devices connected on the network, thus allowing any devices to create a map of network. so this is kind of a network created for HDMI compliant devices.

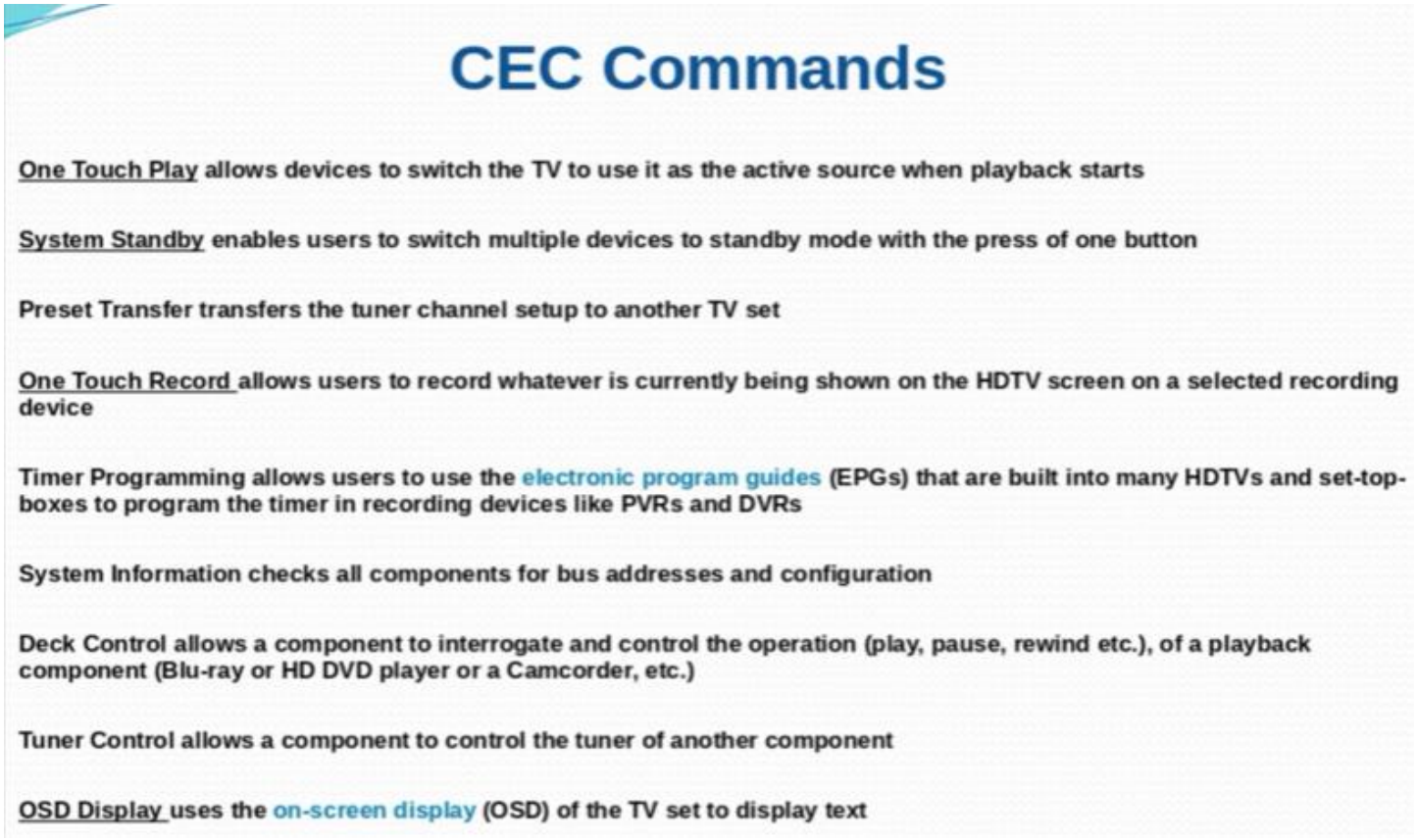


So this is kind of a example for the devices cluster. You can see here this is connected to setup box. Setup box is further connected to audio video repeater amplifier and audio receiver. Amplifier is connected to TV and then DVD connected to audio video receiver amplifier. All devices are connected to the audio video receiver. Okay, now this diagram basically show the address scheme of all these devices. So you can see here the main devices.

Or you can say the remote controller or the main controller devices or we can say the master device basically will have the the address given 0.000 means four times zero. It will have okay. And this is basically connected to node present device by means 1.00 is not present. It is connected to audio video receiver, amplifier and the. For the second layer we are given that this 2.00 this address we are given. So the device which will be connected to this audio video receiver or amplifier, this will that will get the second bit. Second bit of address will be set.

In that case you can see here like DVD will get 2.1 DV, VHS got 2.2 Set-top box got 2.3 and then PVR, then PVR is is on the third layer. It is connected to the setup box, so it will get there. Third bit set basically for particular number. So you can see here it is like PVR has got 2.3.1 address. So this is how we give the address to all physical devices present into the cluster of HDMI compliant devices.

And by using of this address, basically our remote controller or any other controller basically control the devices connected into the system.



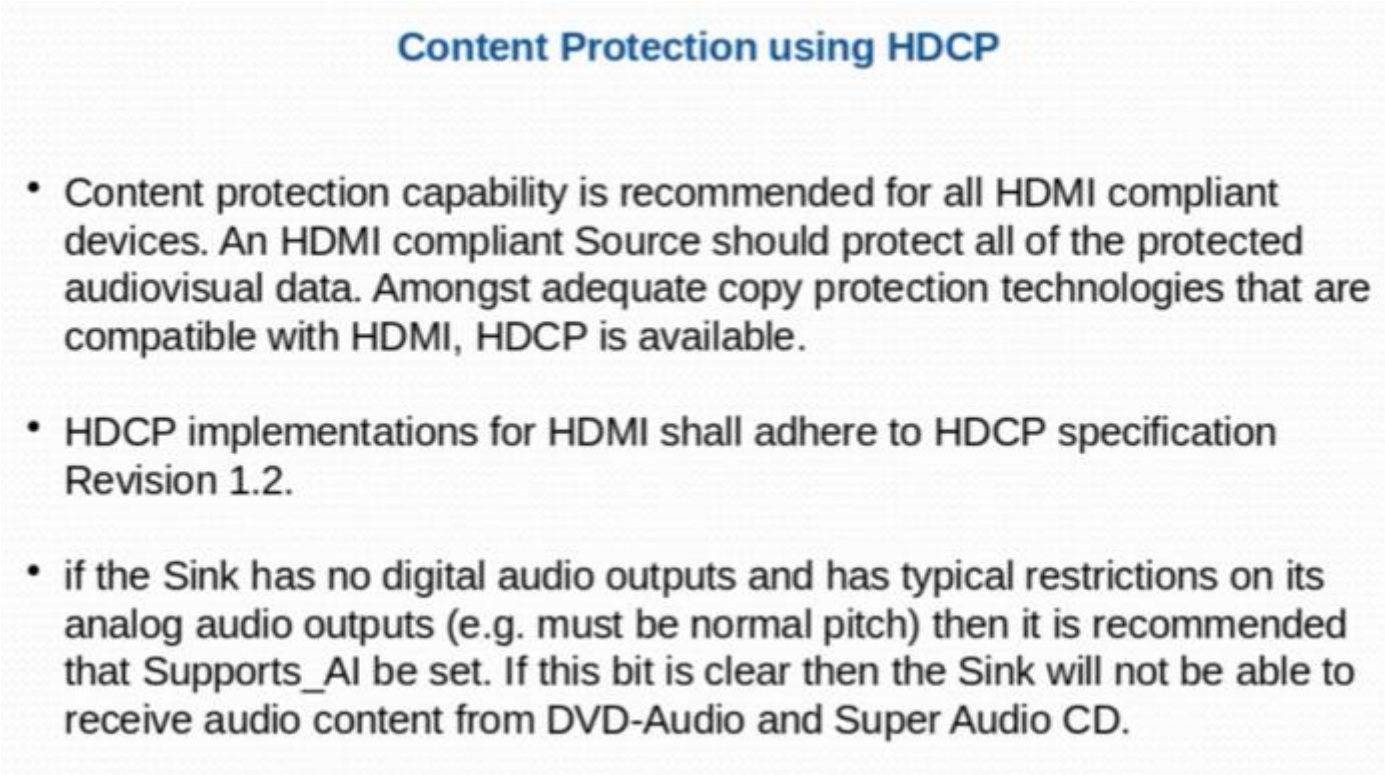
So there are a lot of example about CEC commands. So one touch play basically allow devices to switch the TV to use it as the active source when playback starts. It means like by using a one touch on your mobile device. Basically you can play anything on your TV. It's just like a Chromecast. You might have used Google Chromecast, so that is one of the example.

System standby enable users to switch multiple devices to standby mode with the press of one button. It means when you will, you will basically press the switch over standby button from the remote so the device is connected in a cluster. All devices will go into the system standby state.

So this is how you have preset transfer. Transfer that tuner channel setup to another TV set one touch record allow user to record whatever is basically currently being shown on the screen. You can record everything and it will be stored in somewhere in the secondary memory.

Okay, so all these you have deck control. You have tuner control. You have OSD display. This uses the on screen display of the TV set to display text okay.

So this was the example of CEC commands.



content protection using Hdcp. So as we have seen like in the cluster of HDMI devices connected, we transmit a load of HD data, high definition data basically from one device to another device. And you know, like this data travels through through the connection between these two devices, actually. So there are chances, like some other devices, some other basically unauthorized device which are connected on the network, can can hack the data between your two devices, which is basically authorized to transmit the data between themselves.

So third device, which is unauthorized, can basically crack the transmission of data between these two devices and get the information which it has, it don't has authorized to access. So to protect this basically our our standards follow Hdcp. It means high definition content protocol system. So which is basically used to use to basically provide the protection of the content which is being transmitted between devices actually.

So Hdcp specification revision 1.2 basically tell us about the implementation of this protocol. Like Hdcp provides a different protocols to us by which we can basically protect the data which is being transmitted actually. So there are lots of algorithms associated with it, but this is not in the scope of this training, basically, because for a for a device developer, they don't need to basically go through the algorithm until and unless they are modifying anything with related to Hdcp actually.

5)

HDMI PHY is the actual hardware, which is going to basically transmit your data between source and sink. So this basically plays a very important role because your PHY capability decides that the the capability of the data transmission between your source and sink device. Eno and Synopsys gives HDMI PHY.

6)

Find more info about colour formats like YCBCR etc.

Watch videos from 6-9.