AN157

Application Note

CS7620/22 FREQUENTLY ASKED QUESTIONS

What is the status of the chip at power-up?

At power up the CS7620/22 is in Normal mode with 13-bit output and compander curve turned off.

What about calibration?

The user, using register 0x04h bit 0, can perform calibration on demand. Calibration takes approximately 760 clock cycles. The goal of calibration is to ensure a monotonic digital output across the 13-bit data. Ideally calibration should be performed each time the chip is powered up, waiting 500us after power up to initiate calibration.

Which registers need to be programmed for minimal operation?

The default value is 13-bit output data. No register needs to be programmed, only CK_DT and CK_FT need to be provided.

How to put the chip in stand by mode?

By default the CS7622 is NOT in stand by mode. You can put the chip in stand by mode through register 0x01h. It takes 500us to recover from stand by. In stand by mode the CS7622 uses only 0.0825mW at 3.3V. The CS7620 uses only 0.125mW at 5V.

What is the preview mode?

The preview mode is done through register 0x04h bit 4. In this mode the output is 9-bit using DRX. This mode uses 20mA less than normal mode and is intended to be used for viewfinder or any other function where resolution is not absolutely necessary. The CS7620/22 recovers instantaneously from preview to normal mode (less than 15 clock cycles).

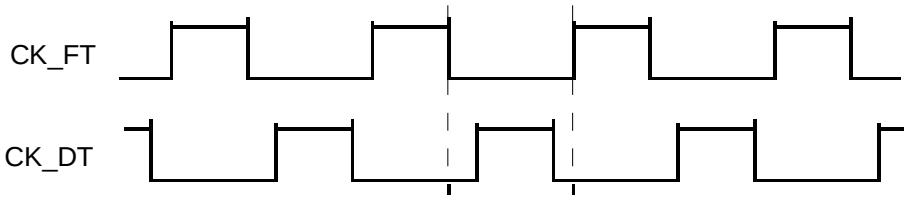
Does the CS7622 need a clock for the serial interface to work?

Yes, both CK_FT and CK_DT need to be present.

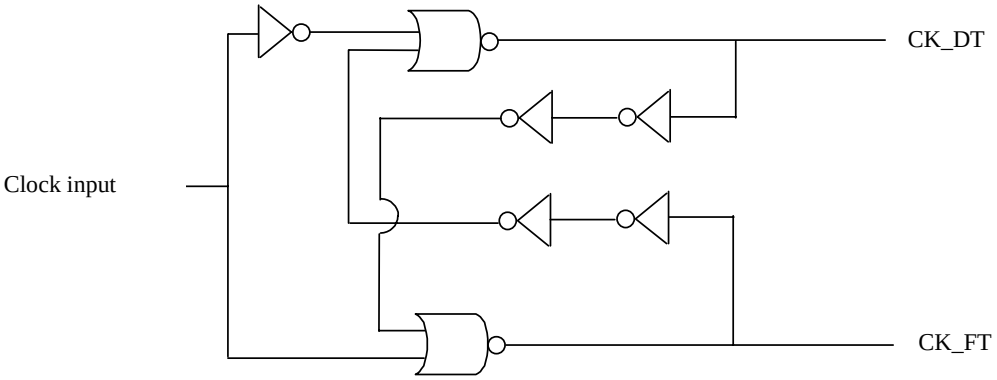
CIRRUS LOGIC
P.O. Box 17847, Austin, Texas 78760
(512) 445 7222 FAX: (512) 445 7581
http://www.cirrus.com

How to generate a non-overlapping clock from a standard clock?

The clocks used to sample the Feedthrough and Pixel level must be non-overlapping as shown below.



If your system cannot generate a non-overlapping clock, you may use the following circuit to generate CK_DT and CK_FT.



* Refer to CS7622 Datasheet for timing information.

How do the CS7622 compares to the competition?

Power usage:

The CS7622 has the lowest power consumption of the industry in stand by mode i.e.: 0.0825mW. An important feature for battery powered application like digital still camera (DSC). Another important feature is the preview mode: this mode is used when the resolution of the picture is not critical like a viewfinder. In preview mode you can save 20mA from the normal mode. Most of the time the camera will be in stand by or preview mode, taking the picture is actually a very small portion of the time that the CS7622 will be used in.

Resolution:

The CS7622 is a 10-bit ADC with 13-bit dynamic range; you will get higher resolution where you need it (dark region) without compromising the brighter part of the scene. Also, the companding curves allows for different adjustments. Note that the companding curves are also available in preview mode, therefore in an application like DSC, changes in those curves could be done in preview and these changes will still be available in normal mode.

Size:

The tiny 32-pin TQFP package and high integration make it an ideal part for handheld application.

What are voltage requirements?

The CS7620 is 5V only. The CS7622 can be powered at different voltages. The I/O of the CS7622 can have a different supply than and analog supply (see table below). Here are the possible choices:

Analog Levels	Digital Levels	Digital I/O Level
5V.	5V	CMOS
3.3V.	3.3V	CMOS
3.3V.	2.5V	CMOS

How to use serial communication?

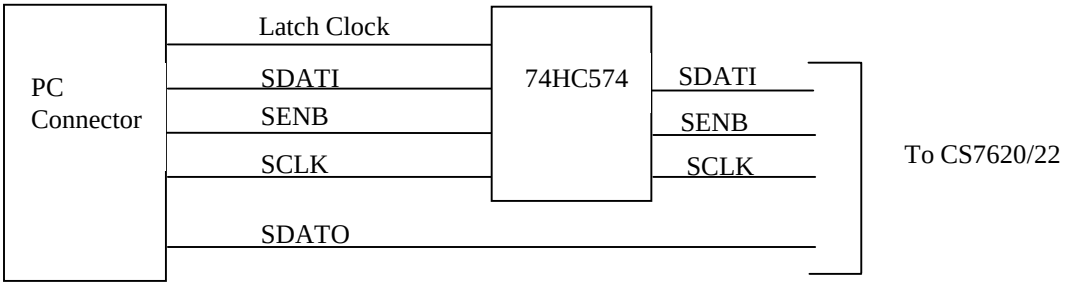
Interface:

The CS7620/22 has a general purpose serial bus interface. It is a four-wire interface, SEN, SDATO, SDATI and SCLK. This serial interface and its timing are described in detail in the datasheet. A communication software using the PC parallel port has been develop to interface with the CS7620/22. Software is available upon request.

Here is the pin description:

PC Connector	Pin Name
Pin 1	Latch
Pin 2	SDATI
Pin 3	SCLK
Pin 4	SENB
Pin 10	SDATO

The use of a latch or flip-flop is necessary only when using the PC because the PC parallel port does not guarantee its levels.



How do I program the CS7620/22?

The CS7620 can be programmed using the general purpose serial communication port. The port can be accessed in two ways. first, by directly connecting the DSP/processor to the port. Second, by using the PC paraleel port with the CS7620 software. The PC interface is targeted towards debug/development purpose.

CS7622 Specifics:

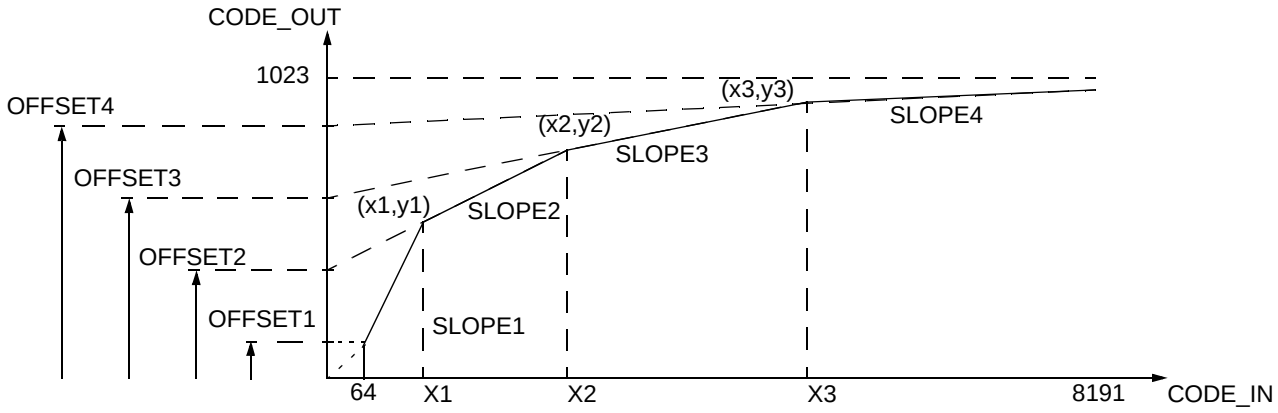
At power up the CS7622 is set to Peak mode with the compander and curves turned off. All 13 hits are available. The CS7620 is always Slave.

CS7620 Specifics:

At power up the compander curves are turned off. All 13 ibts are available. Also, the CS7620 can be Master, Partial Master, or Slave. Hardware configuration is required for settin up the mode (See datasheet and AN156 for details).

What is the compander?

The companding curves allow you to take advantage of the extended dynamic range given by the DRX module with a more standard 10-bit output. The compander module takes 13-bit data input and output either 10-bit companded output, 12-bit (MSB-clipped) data or pass through original 13-bit. By default the compader is turned off and companding curves are linear.



As you can see on the 13-to-10 bit compander figure, the curve has four programmable knees. The default values of x1, x2, x3 etc. can be found in the register description of the datasheet. The default value are set such that they compensate well for backlighting conditions. To test your own set of value you have to make sure the curves are continous. Here is how to calculate the curve:

1) Placement of the three knees:

Select the location, (x1, y1), (x2, y2), (x3, y3) of the knees. The advantage of having programmable

knees is that if you are gathering data using a histograming method you can use the results of the histogram to select the value of the x,y coordinates to reshape your histogram. Note that the value of x and y can be anything from 0 to 8191 but to insure continuity in the curves the values should be such that x1 < x2 < x3 and y1< y2 < y3.

For instance, the following are valid coordinates for x and y (512, 200) (2048, 512) (5000, 768). These values are decimals.

2) Determine the slopes.

Slope can't be higher than 1.996 (step size is 0.0039)

The slope has to be multiplied by 256 and then converted into hexadecimal to get the proper slope value to use for programming.

Using the values selected in step #1 we have:

Slope #1 : (y1-y0) / (x1-x0) = 0.42 register value in decimal is 0.42 x 256 = 110

Slope #2 : =0.203register value in decimal is 0.203 x 256 = 52

Slope #3 : =0.0867register value in decimal is 0.0867 x 256 = 22

Slope #4 : =0.0799register value in decimal is 0.0799 x 256 = 20

3) Determine the offsets.

Offset1 will be set to 8 for linear mode.

Offset2 = y1-(x1 x slope2) = 200 - (512 x 0.203) = 96

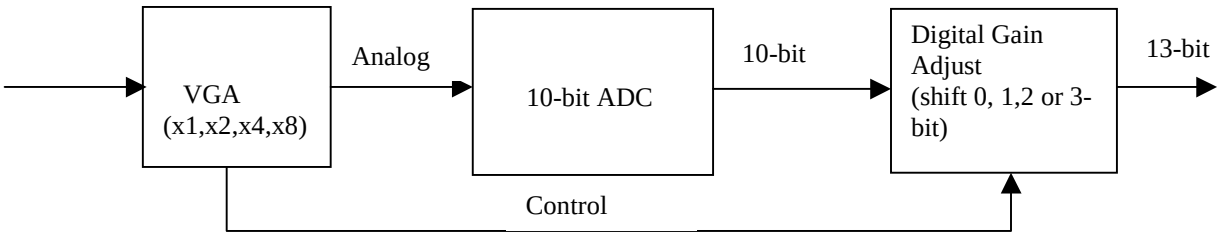
Offset3 = y2-(x2 x slope3) = 512 - (2048 x 0.0867) = 334

Offset4 = y3-(x3 x slope4) = 768 - (5000 x 0.0799) = 368

With all the values you are now ready to configure the companding curves. Again all values needs to be converted in hexadecimal. If you don't want to calculate a curve each time, you can use the curves default values which gives good results in most cases.

What is DRX?

DRX is a module that allow to extended the dynamic range of an ADC. This module comprise three block, a VGA (variable gain amplifier), a digital gain adjust block and a 10-bit ADC. The VGA provides a gain of x1,x2,x4 or x8 based on the input voltage coming from the CCD. Each pixel will be individually gained to insure a maximum use of the ADC range. After the VGA, the signal is digitized by the 10-bit ADC, then the digital gain adjust block will divide (shift right) its output bit to match the analog gain performed before digitization. This is how the extended dynamic range is performed. Note that three bits of lower significance are kept to prevent unwanted truncation.



The two tables below allow can be used to follow the path of three pixels from the VGA to the output of the CS7620/22

Example using a full scale input range of 1.0V				
	Analog Input	**VGA Gain**	**ADC Output**	**Binary ADC Value**
1st Pixel	0.1V	x8	1024 x (0.8V/1.0V) = 819	1100110011
2nd Pixel	0.45V	x2	1024 x (0.9V/1.0V) =921	1110011001
3rd Pixel	0.8V	x1	1024 x (0.8V/1.0V) =819	1100110011

The three LSB are physically added in the Digital gain adjust block, in the table below the extra bits are shown in the ADC output for clarity purpose only.

	1st Pixel	**2nd Pixel**	**3rd Pixel**
ADC Output	1100110011.000	1110011001.000	1100110011.000
Control	Shift right by 3 (gain x8)	Shift right by 1 (gain x2)	Shift right by 0 (gain x1)
Digital Gain Adjust Output	0001100110.011	0111001100.100	1100110011.000

Note that the DRX can be set to a fixed value of x1, x2, x4 or x8.

Where they can get 9-bit of ADC output in 6-bit Low Power Mode?

Actually in the 6-bit mode there are 9 bits out of the chip after using the DRX circuitry to extend the dynamic range. That is why the outputs are on DOUT[12:4]. The result using the 6-bit mode is 6-bit of resolution and 9-bit dynamic range. It should be noted that although there is only 6-bit of resolution, in the lower 1/8th of the input range is mapped to the 6-bit ADC range producing an effective resolution of 9 bits on these pixels.

How do you get 9 effective bits with 6-bit ADC?

By having the lower 1/8th of the signal mapped directly to the full ADC range you get a higher effective resolution when it is most needed i.e.: low level signal. In the case

For example if we have a 1.0V full scale signal and we are in Low resolution mode (6-bit ADC) we can make the following calculation:

For a full scale signal of 1.0V and a 9-bit ADC you have a LSB step of 0.00195 V/ step.

1.0V / 29 = 0.00195

With the DRX technology, any signal that is less than 1/8th of the full scale will use the full range of the ADC (in our case anything below 0.125V), therefor we have:

0.125V / 26 = 0.00195

We have the same quantization step (LSB) as if we had a 9-bit ADC. Also the DRX is most effective where needed, i.e. in the low level signal.

What is the Black clamp input voltage range?

We can accommodate roughly +30mV and -70mV off internal offset generated in the analog data path ((+/- 17 mV) plus CCD offsets and other external offsets with either the CS7620 or CS7622. Typically we expect to see 20mV of CCD offsets. An offset is added in the first stage. the offset range is +30 to -70 mV and a digital control loop sets the offset so that the output of the ADC is 64 during black pixel. This value is hard coded and cannot be changed so that if clamp is performed properly over black pixels the output of the chip during optically black pixels will be ~64 (digital code). The black loop may be disabled and written to manually which will allow you to select any range of offset +30 to -70 mV. With the loop off, black pixels will not necessarily output code 64.


SMART
Analog
™

littlediode
COMPONENTS FOR ALL

Littlediode supplies new, hard to find or obsolete electronic components and semiconductors all over the world.

With over two million different components listed you are sure to find the part you need.

Feel free to visit us today at our online store:

LittleDiode.com

Looking forward to providing you with the best possible service.