Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

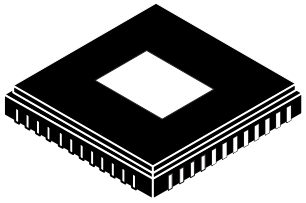
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

MT9D131

1/3.2-Inch System-On-A-Chip (SOC) CMOS Digital Image Sensor

ON Semiconductor®

www.onsemi.com



48 CLCC
CASE TBD

ORDERING INFORMATION

See detailed ordering and shipping information on page 4 of this data sheet.

General Description

The ON Semiconductor MT9D131 is a 1/3.2 inch, 2 Mp CMOS image sensor with an integrated advanced camera system. The camera system features a microcontroller (MCU) and a sophisticated image flow processor (IFP) with a real-time JPEG encoder.

The microcontroller manages all components of the camera system and sets key operation parameters for the sensor core to optimize the quality of raw image data entering the IFP. The sensor core consists of an active pixel array of 1668 x 1248 pixels, programmable timing and control circuitry including a PLL, analog signal chain with automatic offset correction and programmable gain, and two 10-bit A/D converters (ADC). The entire system-on-a-chip (SOC) has ultra-low power requirements and superior low-light performance that is particularly suitable for surveillance applications.

The excellent low-light performance of MT9D131 is one of the hallmarks of ON Semiconductor's breakthrough low-noise CMOS imaging technology that achieves CCD image quality (based on signal-to-noise ratio and low-light sensitivity) while maintaining the inherent size, cost, power consumption, and integration advantages of CMOS.

Feature Overview

The MT9D131 is a color image sensor with a Bayer color filter arrangement. Its basic characteristics are described in Table 1.

The MT9D131 has an embedded phase-locked loop oscillator (PLL) that can be used with the common wireless system clock. When in use, the PLL adjusts the incoming clock frequency, allowing the MT9D131 to run at almost any desired resolution and frame rate. To reduce power consumption, the PLL can be bypassed and powered down.

Low power consumption is a very important requirement for all components of wireless devices. The MT9D131 has numerous power conserving features, including an ultra low-power standby mode and the ability to individually shut down unused digital blocks.

Another important consideration for wireless devices is their electromagnetic emission or interference (EMI). The MT9D131 has a programmable I/O slew rate to minimize its EMI and an output FIFO to eliminate output data bursts.

The advanced IFP and flexible programmability of the MT9D131 provide a variety of ways to enhance and optimize the image sensor performance. Built-in optimization algorithms enable the MT9D131 to operate at factory settings as a fully automatic, highly adaptable camera. However, most of its settings are user-programmable.

Applications

- Network Security Cameras
- ePTZ Cameras
- High Resolution Security Camera
- Wireless Cameras
- Consumer Video Products

Features

- Superior Low-light Performance
- Ultra-low-power, Cost Effective
- Internal Master Clock Generated by on-chip Phase- locked Loop Oscillator (PLL)
- Electronic Rolling Shutter (ERS), Progressive Scan
- Integrated Image Flow Processor (IFP) for Single-die Camera Module
- Automatic Image Correction and Enhancement, Including Lens Shading Correction
- Arbitrary Image Decimation with Anti-aliasing
- Integrated Real-time JPEG Encoder
- Integrated Microcontroller for Flexibility
- Two-wire Serial Interface Providing Access to Registers and Microcontroller Memory
- Selectable Output Data Format: ITU-R BT.601 (YCbCr), 565RGB, 555RGB, 444RGB, JPEG 4:2:2, JPEG 4:2:0, and raw 10-bit
- Output FIFO for Data Rate Equalization
- Programmable I/O Slew Rate

TABLE OF CONTENTS

TABLE 1. KEY PERFORMANCE PARAMETERS

Parameter		Typical Value
Optical format		1/3.2-inch (4:3)
Full resolution		1600 x 1200 pixels (UXGA)
Pixel size		2.8 μm x 2.8 μm
Active pixel array area		4.73 mm x 3.52 mm
Shutter type		Electronic rolling shutter (ERS)
Maximum frame rate		15 fps at full resolution, 30 fps in preview mode, (800 x 600)
Maximum data rate/ master clock		80 Mp/s 6–80 MHz
Supply voltage	Analog	2.5–3.1 V
	Digital	1.7–1.95 V
	I/O	1.7–3.1 V
	PLL	2.5–3.1 V
ADC resolution		10–bit, on-die
Responsivity		1.0 V/lux-sec (550 nm)
Dynamic range		71 dB
SNR_{MAX}		42.3 dB
Power consumption		348 mW at 15 fps, full resolution
		223 mW at 30 fps, preview mode
Operating temperature		–30°C to +70°C
Package		48-pin CLCC

ORDERING INFORMATION

TABLE 2. AVAILABLE PART NUMBERS

Part Number	Product Description	Orderable Product Attribute Description
MT9D131C12STC−DP	2.0 MP 1/3” SOC	Dry Pack with Protective Film
MT9D131C12STC−DR	2.0 MP 1/3” SOC	Dry Pack without Protective Film
MT9D131C12STC−TP	2.0 MP 1/3” SOC	Tape & Reel with Protective Film
MT9D131D00STCK15LC1−305	2.0 MP 1/3” SOC	Die Sales, 305 μm Thickness

TYPICAL CONNECTION

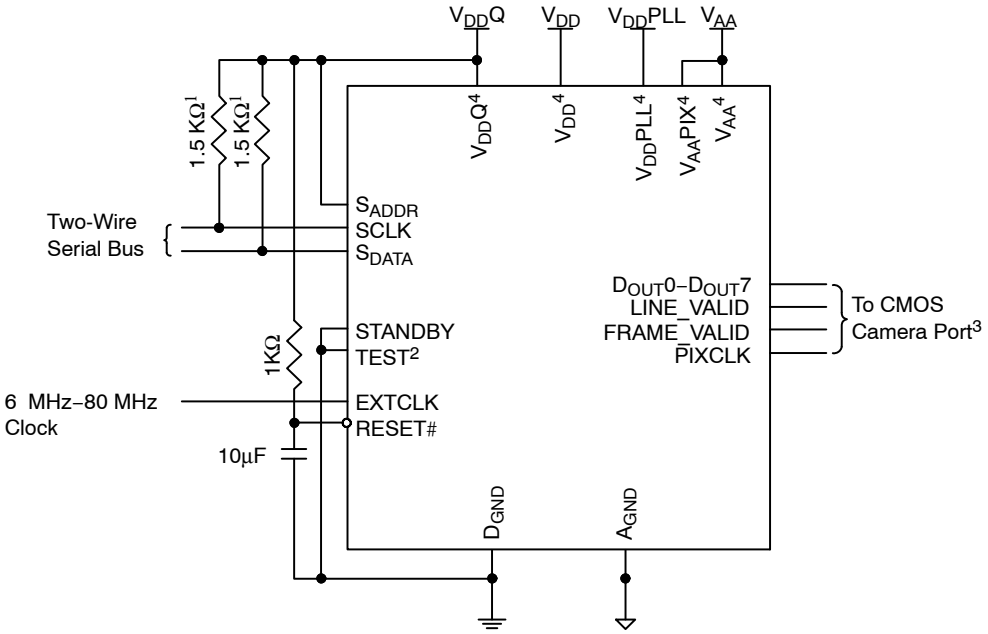


Figure 1. Typical Configuration (Connection)

Notes:
1. 1. Resistor value 1.5 KΩ is recommended, but may be greater for slower two-wire speed.
2. TEST must be connected to digital ground for normal device operation.
3. See “Standby Hardware Configuration”.
4. All power supply pads must be used.

SIGNAL DESCRIPTION

TABLE 3. SIGNAL DESCRIPTION

Name	Type	Description	Note
EXTCLK	Input	Master clock signal (can either drive the on-chip PLL or bypass it).	
RESET_BAR	Input	Master reset signal, active LOW.	
STANDBY	Input	Controls sensor's standby mode.	
TEST	Input	Reserved for factory test. Tie to digital ground during normal operation.	
SCLK	Input	Two-wire serial interface clock.	
SADDR	Input	Selects device address for the two-wire serial interface. The address is 0x90 when SADDR is tied LOW, 0xBA if tied HIGH. See also R0x0D:0[10].	
DOUT0–DOUT7	Output	Eight-bit image data output or most significant bits (MSB) of 10-bit sensor bypass mode.	1
FRAME_VALID	Output	Identifies rows in the active image.	1
LINE_VALID	Output	Identifies lines in the active image.	1
PIXCLK	Output	Pixel clock. To be used for sampling DOUT, FRAME_VALID, and LINE_VALID.	1
SDATA	I/O	Two-wire serial interface data.	
VDD	Supply	Digital power (1.8V).	
VDDPLL	Supply	PLL power (2.8V).	
VAA	Supply	Analog power (2.8V).	
VAAPIX	Supply	Pixel array power (2.8V).	
VDDQ	Supply	I/O power (nominal 1.8V or 2.8V).	
VDDGPIO	Supply	I/O power for GPIO (nominal 1.8V or 2.8V).	
AGND	Supply	Analog ground.	
DGND	Supply	Digital, I/O, and PLL ground.	

1. 1. See "Standby Hardware Configuration".

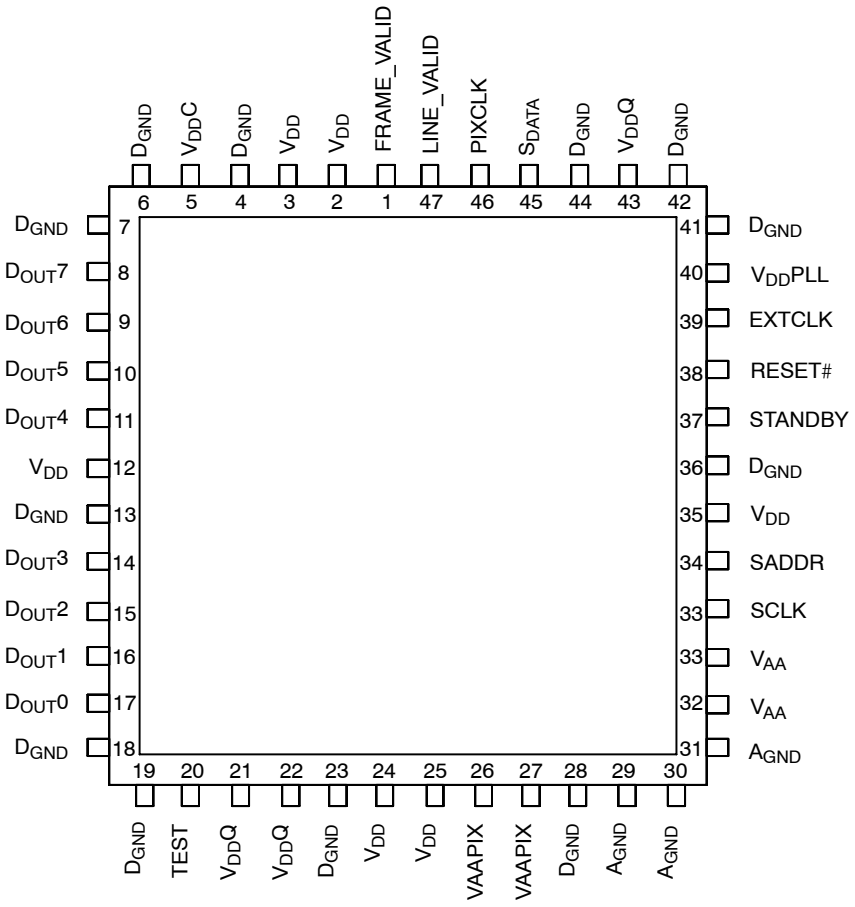


Figure 2. 48-Pin CLCC Pinout Diagram

ARCHITECTURE OVERVIEW

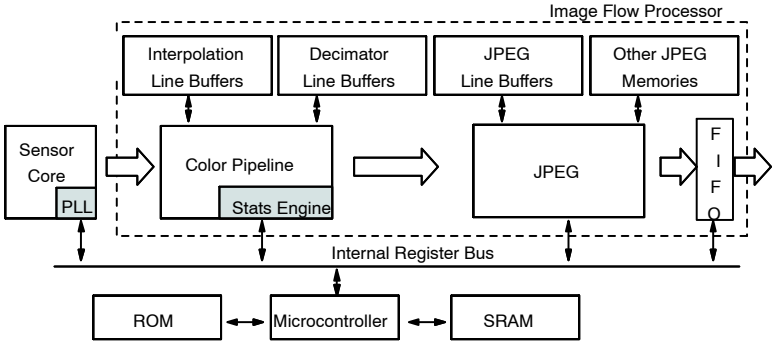


Figure 3. Block Diagram

Sensor Core

The MT9D131 sensor core is based on ON Semiconductor's MT9D011, a stand-alone, 2-megapixel CMOS image sensor with a 2.8mm pixel size. Both image sensors have the same optical size (1/3.2 inches) and maximum resolution (UXGA). Like the MT9D011, the MT9D131 sensor core includes a phase-locked loop oscillator (PLL), to facilitate camera integration and minimize the system cost for surveillance applications. When in use, the PLL generates internal master clock signal whose frequency can be set higher than the frequency of external clock signal EXTCLK. This allows the MT9D131 to run at any desired resolution and frame rate up to the specified maximum values, irrespective of the EXTCLK frequency.

Color Pipeline

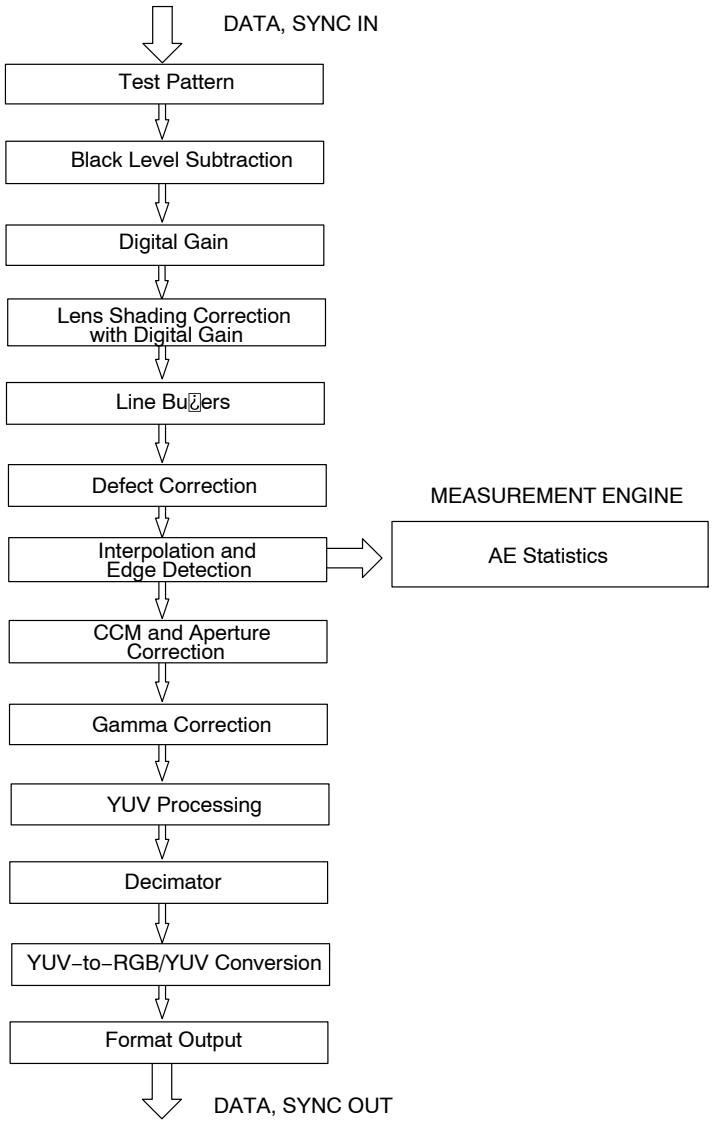


Figure 4. Color Pipeline

Test Pattern

During normal operation of MT9D131, a stream of raw image data from the sensor core is continuously fed into the color pipeline. For test purposes, this stream can be replaced with a fixed image generated by a special test module in the pipeline. The module provides a selection of test patterns sufficient for basic testing of the pipeline.

Black Level Conditioning and Digital Gain

Image stream processing starts with black level conditioning and multiplication of all pixel values by a programmable digital gain.

Lens Shading Correction

Inexpensive lenses tend to produce images whose brightness is significantly attenuated near the edges. Chromatic aberration in such lenses can cause color variation across the field of view. There are also other factors causing fixed-pattern signal gradients in images captured by image sensors. The cumulative result of all these factors is known as lens shading. The MT9D131 has an embedded lens shading correction (LC) module that can be programmed to precisely counter the shading effect of a lens on each RGB color signal. The LC module multiplies RGB signals by a two-dimensional correction function F(x,y), whose profile in both x and y direction is a piecewise quadratic polynomial with coefficients independently programmable for each direction and color.

Line Buffers

Several data processing steps following the lens shading correction require access to pixel values from up to 8 consecutive image lines. For these lines to be

simultaneously available for processing, they must be buffered. The IFP includes a number of SRAM line buffers that are used to perform defect correction, color interpolation, image decimation, and JPEG encoding.

Defect Correction

The IFP performs on-the-fly defect correction that can mask pixel array defects such as high-dark-current ("hot") pixels and pixels that are darker or brighter than their neighbors due to photoresponse nonuniformity. The defect correction algorithm uses several pixel features to distinguish between normal and defective pixels. After identifying the latter, it replaces their actual values with values inferred from the values of nearest same-color neighbors.

Color Interpolation and Edge Detection

In the raw data stream fed by the sensor core to the IFP, each pixel is represented by a 10-bit integer number, which, to make things simple, can be considered proportional to the pixel's response to a one-color light stimulus, red, green or blue, depending on the pixel's position under the color filter array. Initial data processing steps, up to and including the defect correction, preserve the one-color-per-pixel nature of the data stream, but after the defect correction it must be converted to a three-colors-per-pixel stream appropriate for standard color processing. The conversion is done by an edge-sensitive color interpolation module. The module pads the incomplete color information available for each pixel with information extracted from an appropriate set of neighboring pixels.

The algorithm used to select this set and extract the information seeks the best compromise between maintaining the sharpness of the image and filtering out high-frequency noise. The simplest interpolation algorithm is to sort the nearest eight neighbors of every pixel into three sets-red, green, and blue: discard the set of pixels of the same color as the center pixel (if there are any): calculate average pixel values for the remaining two sets, and use the averages instead of the missing color data for the center pixel. Such averaging reduces high-frequency noise, but it also blurs and distorts sharp transitions (edges) in the image. To avoid this problem, the interpolation module performs edge detection in the neighborhood of every processed pixel and, depending on its results, extracts color information from neighboring pixels in a number of different ways. In effect, it does low-pass filtering in flat-field image areas and avoids doing it near edges.

Color Correction and Aperture Correction

To achieve good color fidelity of IFP output, interpolated RGB values of all pixels are subjected to color correction. The IFP multiplies each vector of three pixel colors by a 3 x 3 color correction matrix. The three components of the resulting color vector are all sums of three 10-bit numbers. Since such sums can have up to 12 significant bits, the bit width of the image data stream is widened to 12 bits per color (36 bits per pixel). The color correction matrix can be either programmed by the user or automatically selected by the auto white balance (AWB) algorithm implemented in the IFP. Color correction should ideally produce output colors that are independent of the spectral sensitivity and color cross-talk characteristics of the image sensor. The optimal values of color correction matrix elements depend on those sensor characteristics and on the spectrum of light incident on the sensor.

To increase image sharpness, a programmable aperture correction is applied to color corrected image data, equally to each of the 12-bit R, G, and B color channels.

Gamma Correction

Like the aperture correction, gamma correction is applied equally to each of the 12-bit R, G, and B color channels. Gamma correction curve is implemented as a piecewise linear function with 19 knee points, taking 12-bit arguments and mapping them to 8-bit output. The abscissas of the knee points are fixed at 0, 64, 128, 256, 512, 768, 1024, 1280, 1536, 1792, 2048, 2304, 2560, 2816, 3072, 3328, 3584, 3840, and 4095. The 8-bit ordinates are programmable through IFP registers or public variables of mode driver (ID = 7). The driver variables include two arrays of knee point ordinates defining two separate gamma curves for sensor operation contexts A and B.

YUV Processing

After the gamma correction, the image data stream undergoes RGB to YUV conversion and, optionally, further corrective processing. The first step in this processing is removal of highlight coloration, also referred to as "color kill." It affects only pixels whose brightness exceeds a certain preprogrammed threshold. The U and V values of those pixels are attenuated proportionally to the difference between their brightness and the threshold. The second optional processing step is noise suppression by one-dimensional low-pass filtering of Y and/or UV signals. A 3- or 5-tap filter can be selected for each signal.

Image Cropping and Decimation

To ensure that the size of images output by MT9D131 can be tailored to the needs of all users, the IFP includes a decimator module. When enabled, this module performs "decimation" of incoming images (shrinks them to arbitrarily selected width and height without reducing the field of view and without discarding any pixel values). The latter point merits underscoring, because the terms "decimator" and "image decimation" suggest image size reduction by deleting columns and/or rows at regular intervals. Despite the terminology, no such deletions take place in the decimator module. Instead, it performs "pixel binning"– divides each input image into rectangular bins corresponding to individual pixels of the desired output image, averages pixel values in these bins and assembles the output image from the bin averages. Pixels lying on bin boundaries contribute to more than one bin average: their values are added to bin-wide sums of pixel values with fractional weights. The entire procedure preserves all image

information that can be included in the downsized output image and filters out high-frequency features that could cause aliasing.

The image decimation in the IFP can be preceded by image cropping and/or image decimation in the sensor core. Image cropping takes place when the sensor core is programmed to output pixel values from a rectangular portion of its pixel array - a window - smaller than the default 1600 x 1200 window. Pixels outside the selected cropping window are not read out, which results in narrower field of view than at the default sensor settings. Irrespective of the size and position of the cropping window, the MT9D131 sensor core can also decimate outgoing images by skipping columns and/or rows of the pixel array, and/or by binning 2 x 2 groups of pixels of the same color. Because decimation by skipping (that is, deletion) can cause aliasing (even if pixel binning is simultaneously enabled), it is generally better to change image size only by cropping and pixel binning.

The image cropping and decimator module can be used to do digital zoom and pan. If the decimator is programmed to output images smaller than images coming from the sensor core, zoom effect can be produced by cropping the latter from their maximum size down to the size of the output images. The ratio of these two sizes determines the maximum attainable zoom factor. For example, a 1600 x 1200 image rendered on a 160 x 120 display can be zoomed up to 10 times, since 1600/160 = 1200/120 = 10. Panning effect can be achieved by fixing the size of the cropping window and moving it around the pixel array.

YUV-to-RGB/YUV Conversion and Output Formatting

The YUV data stream emerging from the decimator module can either exit the color pipeline as-is or be converted before exit to an alternative YUV or RGB data format. See “Color Conversion Formulas” and the description of register R0x97:1 for more details.

JPEG Encoder and FIFO

The JPEG compression engine in the MT9D131 is a highly integrated, high-performance solution that can provide sustained data rates of almost 80 MB/s for image sizes up to 1600 x 1200. Additionally, the solution provides for low power consumption and full programmability of JPEG compression parameters for image quality control.

The JPEG encoding block is designed for continuous image flow and is ideal for low-power applications. After initial configuration for a target application, it can be controlled easily for instantaneous stop/restart. A flexible configuration and control interface allows for full programmability of various JPEG-specific parameters and tables.

JPEG Encoding Highlights

1. Sequential DCT (baseline) ISO/IEC 10918-1 JPEG-compliant
2. YCbCr 4:2:2 format compression
3. Programmable quantization tables
   - One each for luminance and chrominance (active)
   - Support for three pairs of quantization tables–two pairs serve as a backup for buffer overflow
4. Programmable Huffman Tables
   - 2 AC, 2 DC tables–separate for luminance and chrominance
5. Quality/compression ratio control capability
6. 15 fps MJPEG capability (header processing in external host processor)

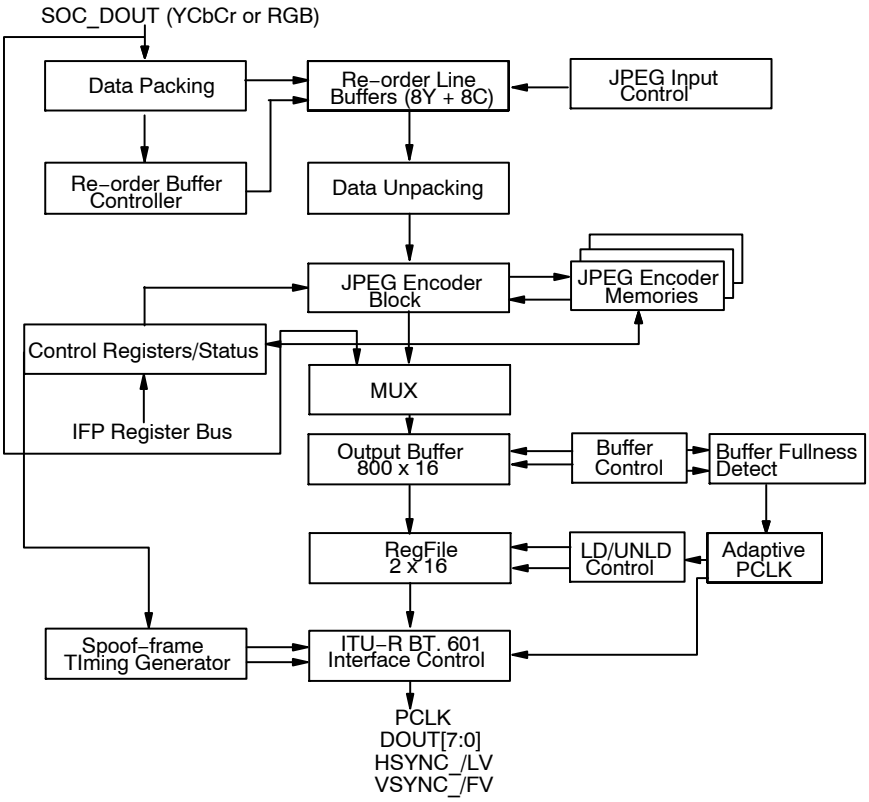


Figure 5. JPEG Encoder Block Diagram

Output Buffer Overflow Prevention

The MT9D131 integrates SRAM for the storage of JPEG data. To prevent output buffer overflow, the MT9D131 implements an adaptive pixel clock (PIXCLK) rate scheme. When the adaptive pixel clock rate scheme is enabled, PIXCLK can run at clock frequencies of *(EXTCLK freq/N1)*, *(EXTCLK freq/N2)*, *(EXTCLK freq/N3)*, where *N1*, *N2*, *N3* are register values programmed by the host through the two-wire serial interface. A clock divider block from the master clock EXTCLK generates the three clocks, PCLK1, PCLK2, and PCLK3.

At the start of the frame encode, PIXCLK is sourced by PCLK1. The buffer fullness detection block of the SOC switches PIXCLK to PCLK2 and then to PCLK3, if necessary, based on the watermark at the output buffer (that is, percentage filled up). When the output buffer watermark reaches 50 percent, PIXCLK switches to PCLK2. This increase in PIXCLK rate unloads the output buffer at a higher rate. However, depending on the image complexity and quantization table setting, the compressed image data may still be generated by the JPEG encoder faster than PIXCLK can unload it. Should the output buffer watermark equal 75 percent or higher, PIXCLK is switched to PCLK3. When the output buffer watermark drops back to 50 percent, PIXCLK is switched back to PCLK2. When the output buffer watermark drops to 25 percent, PIXCLK is switched to PCLK1.

When a decision to adapt PIXCLK frequency is made, LINE_VALID, which qualifies the 8-bit data output (DOUT), is de-asserted until PIXCLK is safely switched to the new clock. LINE_VALID is independent of the horizontal timing of the uncompressed imaged. Its assertion is strictly based on compressed image data availability.

Should an output buffer overflow still occur with PIXCLK at the maximum frequency, the output buffer and the small asynchronous FIFO are flushed immediately. This causes LINE_VALID to be de-asserted. FRAME_VALID is also de-asserted.

In addition to the adaptive PIXCLK rate scheme, the MT9D131 also has storage for three sets of quantization tables (six tables). In the event of output buffer overflow during the compression of the current frame, another set of the preloaded quantization tables can be used for the encoding of the immediate next frame. Then, the MT9D131 starts compressing the next frame starting with the nominal PIXCLK frequency.

Output Interface

Control (Two-Wire Serial Interface)

Camera control and JPEG configuration/control are accomplished through a two-wire serial interface. The interface supports individual access to all camera function registers and JPEG control registers. In particular, all tables located in the JPEG quantization and Huffman memories are accessible through the two-wire interface. To write to a particular register, the external host processor must send the MT9D131 device address (selected by SADDR or

R0x0D:0[10]), the address of the register, and data to be written to it. See “Appendix A: Two-Wire Serial Register Interface” for a description of read sequence and for details of the two-wire serial interface protocol.

Data

JPEG data is output in a BT656-like 8-bit parallel bus DOUT0–DOUT7, with FRAME_VALID, LINE_VALID, and PIXCLK. JPEG output data is valid when both FRAME_VALID and LINE_VALID are asserted. When the JPEG data output for the frame completes, or buffer overflow occurs, LINE_VALID and FRAME_VALID are de-asserted. The output clock runs at frequencies selected by frequency divisors N1, N2, and N3 (registers R0x0E:2 and R0x0F:2), depending on output buffer fullness.

Context and Operational Modes

The MT9D131 can operate in several modes, including preview, still capture (snapshot), and video. All modes of operation are individually configurable and are organized as two contexts−context A and context B. A context is defined by sensor image size, frame rate, resolution, and other associated parameters. The user can switch between the two contexts by sending a command through the two-wire serial interface.

Preview

Context A is primarily intended for use in the preview mode. During preview, the sensor usually outputs low resolution images at a relatively high frame rate, and its power consumption is kept to a minimum. Context B can be configured for the still capture or video mode, as required by the user. For still capture configuration, the user typically specifies the desired output image size; for JPEG compression, how many frames to capture, and so on. For video, the user might select a different image size and a fixed frame rate.

Snapshot

To take a snapshot, the user must send a command that changes the context from A to context B. Typical sequence of events after this command is as follows. First, the camera exposure and white balance adjusts automatically to the changed illumination of the scene. Next, the camera enables JPEG compression and captures one or more frames of desired size. Once the sequence is complete, the camera automatically returns to context A and resumes running preview.

Video

To start video capture, the user has to change relevant context B settings, such as capture mode, image size, and frame rate, and again send a context change command. Upon receiving the command, the MT9D131 switches to the modified context B settings, while continuing to output YUV-encoded image data. Auto exposure automatically switches to smooth continuous operation. To exit the video capture mode, the user has to send another context change command causing the sensor to switch back to context A.

Auto Exposure

The auto exposure (AE) algorithm performs automatic adjustments of the image brightness by controlling exposure time and analog gains of the sensor core as well as digital gains applied to the image.

Two auto exposure algorithm modes are available:

1. Preview
2. Scene-evaluative

Auto exposure is implemented by means of a firmware driver that analyzes image statistics collected by exposure measurement engine, makes a decision and programs the sensor core and color pipeline to achieve the desired exposure. The measurement engine subdivides the image into 16 windows organized as a 4 x 4 grid.

Preview Mode

This exposure mode is activated during preview or video capture. It relies on the exposure measurement engine that tracks speed and amplitude of the change of the overall luminance in the selected windows of the image.

The backlight compensation is achieved by weighting the luminance in the center of the image higher than the luminance on the periphery. Other algorithm features include the rejection of fast fluctuations in illumination (time averaging), control of speed of response, and control of the sensitivity to the small changes. While the default settings are adequate in most situations, the user can program target brightness, measurement window, and other parameters described above.

Scene-Evaluative Algorithm

A scene-evaluative AE algorithm is available for use in snapshot mode. The algorithm performs scene analysis and classification with respect to its brightness, contrast, and composure and then decides to increase, decrease, or keep original exposure target. It makes the most difference for backlight and bright outdoor conditions.

Auto White Balance

The MT9D131 has a built-in auto white balance (AWB) algorithm designed to compensate for the effects of changing spectra of the scene illumination on the quality of the color rendition. This sophisticated algorithm consists of two major parts: a measurement engine performing statistical analysis of the image and a driver performing the selection of the optimal color correction matrix, digital, and sensor core analog gains. While default settings of these algorithms are adequate in most situations, the user can reprogram base color correction matrices, place limits on color channel gains, and control the speed of both matrix and gain adjustments. Unlike simple white balancing algorithms found in many PC cameras, the MT9D131 AWB does not require the presence of gray or white elements in the image for good color rendition. The AWB does not attempt to

locate “brightest” or “grayest” element of the image but instead performs sophisticated image analysis to differentiate between changes in predominant spectra of illumination and changes in predominant colors of the scene. While defaults are suitable for most applications, a wide range of algorithm parameters can be overwritten by the user through the serial interface.

Flicker Detection

Flicker occurs when the integration time is not an integer multiple of the period of the light intensity. The automatic flicker detection block does not compensate for the flicker, but rather avoids it by detecting the flicker frequency and adjusting the integration time. For integration times below the light intensity period (10ms for 50Hz environment), flicker cannot be avoided.

REGISTERS AND VARIABLES

Three types of configuration controls are available:

- Hardware registers
- Driver variables
- MCU SRAM

The following convention is used in the text below to designate registers and variables:

R0x12:1, R0x12:1[3:0] or R18:1, R18:1[3:0]

These refer to two-wire accessible register number 18, or 0x12 hexadecimal, located on page 1. [3:0] indicate bits. Registers numbers range from 0 through FF and bits range from 15 through 0.

- ae.Target
  This refers to variable ‘Target” in the AE driver.
- SRAM 0x0400
  This refers to special function register or SRAM located at address 0x1080 in MCU memory space.

How to Access

Registers, variables are accessed in different ways.

Registers

Hardware registers are organized into several pages. Page 0 contains sensor controls. Page 1 contains color pipeline controls. Page 2 contains JPEG, output FIFO, and more color pipeline controls. The desired page is selected by writing the desired value to R0xF0. After that, all READs and WRITEs to registers from 0 through FF except R0xF0 and R0xF1, are directed to the selected page. R0xF0 and R0xF1 are special registers and are present on all pages. See “Appendix A: Two-Wire Serial Register Interface” for a description of two-wire register access..

Variables

Variables are located in the microcontroller RAM memory. Each driver, such as auto exposure, white balance, and so on, has a unique driver ID (0...31) and a set of public variables organized as a structure. Each variable in this structure is uniquely identified by its offset from the top of the structure and its size. The size can be 1 or 2 bytes, while the offset is 1 byte.

All driver variables (public and private) can be accessed through R0xC6:1 and R0xC8:1. While two access modes are available (accessed by physical address and by logical address) the public variables are typically accessed by the logical method. The logical address, which is set in R0C6:1, consists of a 5-bit driver ID number and a variable offset. Examples are provided below.

To set the variable ae.Target = 50:

- The variable has a driver ID of 2. Therefore, set R0xC6:1[12:8] = 2
- The variable has an offset of 6. Therefore, set R0xC6:1[7:0] = 6
- This is a logical access. Therefore, set R0xC6:1[14:13] = 01
- The size of the variable is 8 bits. Therefore, set R0xC6:1[15] = 1
- By combining these bits, R0xC6:1 = 0xA206.
- Set R0xC8:1 = 50 for the value of the variable

To read the variable ae.Target:

- Since this is the same variable as the above example, R0xC6:1 = 0xA206
- Read R0xC8:1 for the current variable value

MCU SRAM consists of 1K system memory and 1K user memory. Examples of access:

- Write into user SRAM. Use to upload code
  - R0xC6:1 = 0x400 // address
  - R0xC8:1 = 0x1234 // write 16-bit value
- Read from user SRAM
  - R0xC6:1 = 0x400 // address
  - Read R0xC8:1 // read 16-bit value

See R0xC6:1 and R0xC8:1 description in Table 6, “IFP Registers, Page 2” for more detail.

REGISTERS

Sensor Core Registers

TABLE 4. SENSOR CORE REGISTER DEFAULTS

Register Number (HEX)	Register Description	Default PRE MCU BOOT	Default AFTER MCU BOOT
0x00	Reserved	0x1519	0x1519
0x01	Row Start	0x001C	0x001C
0x02	Column Start	0x003C	0x003C
0x03	Row Width	0x04B0	0x04B0
0x04	Col Width	0x0640	0x0640
0x05	Horizontal Blanking B	0x015C	0x0204
0x06	Vertical Blanking B	0x0020	0x002F
0x07	Horizontal Blanking A	0x00AE	0x00FE
0x08	Vertical Blanking A	0x0010	0x000C
0x09	Shutter Width	0x04D0	N/A
0x0A	Row Speed	0x00011	0x0001
0x0B	Extra Delay	0x0000	0x0000
0x0C	Shutter Delay	0x0000	0x0000
0x0D	Reset	0x0000	0x0000
0x1F	Frame Valid Control	0x0000	0x0000
0x20	Read Mode B	0x0000	0x0300
0x21	Read Mode A	0x0490	0x8400
0x22	Dark Col/Rows	0x010F	0x010F
0x23	Reserved	0x0608	0x0608
0x24	Extra Reset	0x8000	0x8000
0x25	Line Valid Control	0x0000	0x0000
0x26	Bottom Dark Rows	0x0007	0x0007
0x2B	Green Gain	0x0020	N/A
0x2C	Blue Gain	0x0020	N/A
0x2D	Red Gain	0x0020	N/A
0x2E	Green2 Gain	0x0020	N/A
0x2F	Global Gain	0x0020	N/A
0x30	Row Noise	0x042A	0x042A
0x59	Black Rows	0x00FF	0x00FF
0x5B	Dark G1 average	N/A	N/A
0x5C	Dark B average	N/A	N/A
0x5D	Dark R average	N/A	N/A
0x5E	Dark G2 average	N/A	N/A
0x5F	Calib Threshold	0x231D	0x231D
0x60	Calib Control	0x0080	0x0080
0x61	Calib Green1	0x0000	0x0000
0x62	Calib Blue	0x0000	0x0000
0x63	Calib Red	0x0000	0x0000

TABLE 4. SENSOR CORE REGISTER DEFAULTS (continued)

Register Number (HEX)	Register Description	Default PRE MCU BOOT	Default AFTER MCU BOOT
0x64	Calib Green2	0x0000	0x0000
0x65	Clock Control	0xE000	0xE000
0x66	PLL Control 1	0x2809	0x1000
0x67	PLL Control 2	0x0501	0x0500
0xC0	Global Shutter Control	0x0000	0x0000
0xC1	Start Integration (T1)	0x0064	0x0064
0xC2	Start Readout (T2)	0x0064	0x0064
0xC3	Assert Strobe (T3)	0x0096	0x0096
0xC4	De-assert Strobe (T4)	0x00C8	0x00C8
0xC5	Assert Flash	0x0064	0x0064
0xC6	De-assert Flash	0x0078	0x0078
0xE0	External Sample 1	0x0000	0x0000
0xE1	External Sample 2	0x0000	0x0000
0xE2	External Sample 3	0x0000	0x0000
0xE3	External Sampling Control	0x0000	0x0000
0xF0	Page Register	0x0000	0x0000
0xF1	Bytewise Address	0x0000	0x0000
0xF2	Context Control	0x000B	0x0000
0xFF	Reserved	0x1519	0x1519

Registers

Notation used in the sensor register description table:

Sync'd to frame start

- *N* = No. The register value is updated and used immediately.
- *Y* = Yes. The register value is updated at next frame start as long as the synchronize changes bit is 0. Frame start is defined as when the first dark row is read out. By default, this is 8 rows before FRAME_VALID goes HIGH.

Bad frame

A bad frame is a frame where all rows do not have the same integration time, or offsets to the pixel values changed during the frame.

- *N* = No. Changing the register value does not produce a bad frame.
- *Y* = Yes. Changing the register value might produce a bad frame.
  YM = Yes, but the bad frame is masked out unless the "show bad frames" feature is (R0x0D:0[8]) is enabled.

TABLE 5. SENSOR CORE REGISTER DESCRIPTION

Bit Field	Description		Default (Hex)	Sync'd to Frame Start	Bad Frame
R0−0x00 − Reserved (R/O)					
Bits 15:0	Reserved	Reserved	1519		
R1−0x01 − Row Start (R/W)					
Bits 10:0	Row Start	The first row to be read out, excluding any dark rows that may be read. To window the image down, set this register to the starting Y value. Setting a value less than 20 is not recommended because the dark rows should be read using R0x22:0.	1C	Y	YM
R2−0x02 − Column Start (R/W)					
Bits 10:0	Column Start	The first column to be read out, excluding dark columns that may be read. To window the image down, set this register to the starting X value. Setting a value below 52 is not recommended because readout of dark columns should be controlled by R0x22:0.	3C	Y	YM
R3−0x03 − Row Width (R/W)					
Bits 10:0	Row Width	Number of rows in the image to be read out, excluding any dark rows or border rows that may be read. The minimum supported value is 2.	4B0	Y	YM
R4−0x04 − Column Width (R/W)					
Bits 10:0	Column Width	Number of columns in image to be read out, excluding any dark columns or border columns that may be read. The minimum supported value is 9 in 1 ADC mode and 17 in 2 ADC mode.	640	Y	YM
R5−0x05 − Horizontal Blanking−Context B (R/W)					
Bits 13:0	Horizontal Blanking −Context B	Number of blank columns in a row when context B is selected (R0xF2:0[0] = 1). The extra columns are added at the beginning of a row. See "Frame Rate Control" for more information on supported register values.	15C	Y	YM
R6−0x06 − Vertical Blanking−Context B (R/W)					
Bits 14:0	Vertical Blanking −Context B	Number of blank rows in a frame when context B is selected (R0xF2:0[1] = 1). The minimum supported value is (4 + R0x22:0[2:0]). The actual vertical blanking time may be controlled by the shutter width (R0x09:0). See "Raw Data Timing"	20	Y	N
R7−0x07 − Horizontal Blanking−Context A (R/W)					
Bits 13:0	Horizontal Blanking −Context A	Number of blank columns in a row when context A is selected (R0xF2:0[0] = 0). The extra columns are added at the beginning of a row. See "Frame Rate Control" for more information on supported register values.	AE	Y	YM
R8−0x08 − Vertical Blanking−Context A (R/W)					
Bits 14:0	Vertical Blanking −Context A	Number of blank rows in a frame when context A is chosen (R0xF2:0[1] = 1). The minimum supported value is (4 + R0x22:0[2:0]). The actual vertical blanking time may be controlled by the shutter width (R0x9:0). See "Raw Data Timing"	10	Y	N
R9−0x09 − Shutter Width (R/W)					
Bits 15:0	Shutter Width	Integration time in number of rows. The integration time is also influenced by the shutter delay (R0x0C:0) and the overhead time.	4D0	Y	N
R10−0x0A − Row Speed (R/W)					
Bits 15:14	Reserved	Do not change from default value.			
Bit 13	Reserved	Do not change from default value.			
Bit 8	Invert Pixel Clock	Invert PIXCLK. When clear, FRAME_VALID, LINE_VALID, and DOUT are set up relative to the delayed rising edge of PIXCLK. When set, FRAME_VALID, LINE_VALID, and DOUT are set up relative to the delayed falling edge of PIXCLK.	0	N	N

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field		Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R10–0x0A – Row Speed (R/W)					
Bits 7:4	Delay Pixel Clock	Number of half master clock cycle increments to delay the rising edge of PIXCLK relative to transitions on FRAME_VALID, LINE_VALID, and DOUT.	1	N	N
Bit 3	Reserved	Do not change from default value.			
Bits 2:0	Pixel Clock Speed	A programmed value of N gives a pixel clock period of N master clocks in 2 ADC mode and 2*N master clocks in 1 ADC mode. A value of "0" is treated like (and reads back as) a value of "1."	1	Y	YM
R11–0x0B – Extra Delay (R/W)					
Bits 13:0	Extra Delay	Extra blanking inserted between frames. A programmed value of N increases the vertical blanking time by N pixel clock periods. Can be used to get a more exact frame rate. It may affect the integration times of parts of the image when the integration time is less than one frame. Bad frame does not occur unless integration time is less than one frame.	0	Y	N[1]
R12–0x0C – Shutter Delay (R/W)					
Bits 13:0	Shutter Delay	The amount of time from the end of the sampling sequence to the beginning of the pixel reset sequence. If the value in this register exceeds the row time, the reset of the row does not complete before the associated row is sampled, and the sensor does not generate an image. A programmed value of N reduces the integration time by N/2 pixel clock periods in 1 ADC mode and by N pixel clock periods in 2 ADC mode.	0	Y	N
R13–0x0D – Reset (R/W)					
Bit 15	Synchronize Changes	By default, update of many registers are synchronized to frame start. Setting this bit inhibits this update; register changes remain pending until this bit is returned to "0." When this bit is returned to "0," all pending register updates are made on the next frame start.	0	N	N
Bit 10	Toggle SADDR	By default, the sensor serial bus responds to addresses 0xBA and 0xBB. When this bit is set, the sensor serial bus responds to addresses 0x90 and 0x91. WRITEs to this bit are ignored when STANDBY is asserted. See "Slave Address".	0	N	N
Bit 9	Restart Bad Frames	When set, a restart is forced to take place whenever a bad frame is detected. This can shorten the delay when waiting for a good frame because the delay, when masking out a bad frame, is the integration time rather than the full frame time.	0	N	N
Bit 8	Show Bad Frames	0: Outputs only good frames (default). 1: Output all frames (including bad frames). A bad frame is defined as the first frame following a change to: window size or position, horizontal blanking, pixel clock speed, zoom, row or column skip, binning, mirroring, or use of border.	0	N	N
Bit 7:6	Inhibit Standby / Drive Pins	00 or 01: setting STANDBY HIGH puts sensor into standby state with high-impedance outputs 10: Setting STANDBY HIGH only puts the outputs in High-Z 11: Causes STANDBY to be ignored	0	N	N
Bit 5	Reset SOC	When this bit is set to "1", SOC is put in reset state. It exits this state when the bit is set back to "0. " Any attempt to access SOC registers in the reset state results in a sensor hang-up. The sensor cannot recover from it without a hard reset or power cycle.	0	N	
Bit 4	Output Disable	Setting this bit to "1" puts the pin interface in a High-Z. See "Output Enable Control". If the DOUT*, PIXCLK, Frame_Valid, or Line_Valid are floating during STANDBY, this bit should be set to "0" to turn off the input buffer, reducing standby current). This bit must work together with bit 6 to take effect.	0		
Bit 3	Reserved	Keep at default value.	0		

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field	Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R13−0x0D − Reset (R/W)				
Bit 2	Standby	Setting this bit to "1" places the sensor in a low-power state. Any attempt to access registers R[0xF7:0xFD]:0 in this state results in a sensor hang-up. The sensor cannot recover from it without a hard reset or power cycle.	0	N	YM
Bit 1	Restart	Setting this bit to "1" causes the sensor to truncate the current frame and start resetting the first row. The delay before the first valid frame is read out is equal to the integration time. This bit is write - "1" but always reads back as "0".	0	N	YM
Bit 0	Reset	Setting this bit puts the sensor in reset; the frame being generated is truncated and the pin interface goes to an idle state. All internal registers (except for this bit) go to the default power-up state. Clearing this bit resumes normal operation.	0	N	YM
R31−0x1F − FRAME_VALID Control (R/W)				
Bit 15	Enable Early FRAME_VALID Fall	0: Default. FRAME_VALID goes low 6 pixel clocks after last LINE_VALID. 1: Enables the early disabling of FRAME_VALID as set in bits 14:8. LINE_VALID is still generated for all active rows.	0	N	N
Bits 14:8	Early FRAME_VALID Fall	When enabled, the FRAME_VALID falling edge occurs within the programmed number of rows before the end of the last LINE_VALID. (1 + bits 14:8)*row time + constant (constant = 3 in default mode) The value of this field must not be larger than row width R0x03:0.	0	N	N
Bit 7	Enable Early FRAME_VALID Rise	0: Default. FRAME_VALID goes HIGH 6 pixel clocks before first LINE_VALID. 1: Enables the early rise of FRAME_VALID as set in bits 6:0.	0	N	N
Bits 6:0	Early FRAME_VALID Rise	When enabled, the FRAME_VALID rising edge is set HIGH the programmed number of rows before the first LINE_VALID: (1 + bits 6:0)*row time + horizontal blank + constant (constant = 3 in default mode).	0	N	N
R32−0x20 − Read Mode−Context B (R/W)				
Bit 15	Binning −Context B	When read mode context B is selected (R0xF2:0[3] = 1): 0: Normal operation. 1: Binning enabled. See "Binning" and See "Frame Rate Control" for a full description.	0	Y	YM
Bit 13	Zoom Enable	0: Normal operation. 1: Zoom is enabled, with zoom factor [zoom] defined in bits 12:11. In zoom mode, the pixel data rate is slowed by a factor of [zoom]. This is achieved by outputting [zoom − 1] blank rows between each output row. Setting this mode allows the user to fill a window that is [zoom] times larger with interpolated data. The pixel clock speed is not affected by this operation, and the output data for each pixel is valid for [zoom] pixel clocks. Every row is followed by [zoom − 1] blank rows (with their own LINE_VALID, but all data bits = 0) of equal time. The combination of this register and an appropriate change to the window sizing registers allows the user to zoom to a region of interest without affecting the frame rate.	0	Y	YM
Bits 12:11	Zoom	When zoom is enabled by bit 13, this field determines the zoom amount: 00: Zoom 2x 01: Zoom 4x 10: Zoom 8x 11: Zoom 16x	0	Y	YM

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field	Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R32−0x20 − Read Mode−Context B (R/W)				
Bit 10	Use 1 ADC −Context B	When read mode context B is selected (bit 3, R0xF2:0 = 1): 0: Use both ADCs to achieve maximum speed. 1: Use 1 ADC to reduce power. Maximum readout frequency is now half the master clock frequency, and the pixel clock is automatically adjusted as described for the pixel clock speed register.	0	Y	YM
Bit 9	Show Border	This bit indicates whether to show the border enabled by bit 8. 0: Border is enabled but not shown; vertical blanking is increased by 8 rows and horizontal blanking is increased by 8 pixels. 1: Border is enabled and shown; FRAME_VALID time is extended by 8 rows and LINE_VALID is extended by 8 pixels. See "Pixel Border".	0	N	N
Bit 8	Over Sized	0: Normal UXGA size. 1: Adds a 4-pixel border around the active image array independent of readout mode (skip, zoom, mirror, and so on). Setting this bit adds 8 to the number of rows and columns in the frame.	0	Y	YM
Bit 7	Column Skip Enable −Context B	When read mode context B is selected (R0xF2:0[3] = 1): 0: Normal readout. 1: Enable column skip.	0	Y	YM
Bits 6:5	Column Skip −Context B	When read mode context B is selected (R0xF2:0[3] = 1) and column skip is enabled (bit 7 = 1): 00: Column Skip 2x 01: Column Skip 4x 10: Column Skip 8x 11: Column Skip 16x See "Column and Row Skip" for more information.	0	Y	YM
Bit 4	Row Skip Enable −Context B	When read mode context B is selected (R0xF2:0[3] = 1): 0: Normal readout. 1: Enable row skip.	0	Y	YM
Bits 3:2	Row Skip −Context B	When read mode context B is selected (R0xF2:0[3] = 1) and Row skip is enabled (bit 4 = 1): 00: Row Skip 2x 01: Row Skip 4x 10: Row Skip 8x 11: Row Skip 16x See "Column and Row Skip" for more information.	0	Y	YM
Bit 1	Mirror Columns	Read out columns from right to left (mirrored). When set to "1", column readout starts from column (column start + column size] and continues down to [column start + 1]. When set to "0", readout starts at column start and continues to [column start + column size − 1]. This ensures that the starting color is maintained.	0	Y	YM
Bit 0	Mirror Rows	Read out rows from bottom to top (upside down). When set, row readout starts from row [*row start* + row size] and continues down to [row start + 1]. When clear, readout starts at row start and continues to [row start + row size −1]. This ensures that the starting color is maintained.	0	Y	YM
R33−0x21 − Read Mode−Context A (R/W)				
Bit 15	Binning −Context A	When read mode context A is selected (R0xF2:0[3] = 0): 0: Normal operation. 1: Binning enabled. See "Binning".	1	Y	YM
Bit 10	Use 1 ADC −Context A	When read mode context A is selected (R0xF2:0[3] = 0): 0: Use both ADCs to achieve maximum speed. 1: Use one ADC to reduce power. Maximum readout frequency is now half of the master clock, and the pixel clock is automatically adjusted as described for the pixel clock speed register.	1	Y	YM

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field		Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R33−0x21 − Read Mode−Context A (R/W)					
Bit 7	Column Skip Enable −Context A	When read mode context A is selected (R0xF2:0[3] = 0): 0: Normal readout. 1: Enable column skip.	1	Y	YM
Bits 6:5	Column Skip −Context A	When read mode context A is selected (R0xF2:0[3] = 0) and column skip is enabled (bit 7 = 1): 00: Column Skip 2x 01: Column Skip 4x 10: Column Skip 8x 11: Column Skip 16x See "Column and Row Skip" for more information.	0	Y	YM
Bit 4	Row Skip Enable −Context A	When read mode context A is selected (R0xF2:0[3] = 0): 0: Normal readout. 1: Enable row skip.	1	Y	YM
Bits 3:2	Row Skip −Context A	When read mode context A is selected (R0xF2:0[3] = 0) and Row skip is enabled (bit 4 = 1): 00: Row Skip 2x 01: Row Skip 4x 10: Row Skip 8x 11: Row Skip 16x See "Column and Row Skip" for more information.	0	Y	YM
R34−0x22 − Show Control (R/W)					
Bit 10	Number of Dark Columns	The MT9D131 has 40 dark columns. 0: Read out 20 dark columns (4−23). 1: Read out 36 dark columns (4−39). Ignored during binning, where all 40 dark columns are used.	0	N	N
Bit 9	Show Dark Columns	When set to "1", the 20 or 36 (dependent on bit 10) dark columns are output before the active pixels in a line. There is an idle period of 2 pixels between readout of the dark columns and readout of the active image. Therefore, when set to "1", LINE_VALID is asserted 22 pixel times earlier than normal, and the horizontal blanking time is decreased by the same amount.	0	N	N
Bit 8	Read Dark Columns	0: When disabled, an arbitrary number of dark columns can be read out by including them in the active image. Enabling the dark columns increases the minimum value for horizontal blanking but does not affect the row time. 1: Enables the readout of dark columns for use in the row-wise noise correction algorithm. The number of columns used are 40 in binning mode, and otherwise determined by bit 10.	1	N	Y
Bit 7	Show Dark Rows	When set to "1", the programmed dark rows is output before the active window. FRAME_VALID is thus asserted earlier than normal. This has no effect on integration time or frame rate.	0	N	N
Bits 6:4	Dark Start Address	The start address for the dark rows within the 8 available rows (an offset of 4 is added to compensate for the guard pixels). Must be set so all dark rows read out falls in the address space 0:7.	0	N	N
Bit 3	Reserved	Do not change from default value.			
Bits 2:0	Num Dark Rows	A value of N causes n + 1 dark rows to be read out at the start of each frame when dark row readout is enabled (bit 3).	7	N	Y
R36−0x24 − Extra Reset (R/W)					
Bit 15	Extra Reset Enable	0: Only programmed window (set by R0x01:0 through R0x04:0) and black pixels are read. 1: Two additional rows are read and reset above and below programmed window to prevent blooming to active area.	1	N	N

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field	Description		Default (Hex)	Sync'd to Frame Start	Bad Frame
R36–0x24 – Extra Reset (R/W)					
Bit 14	Next Row Reset	When set, and the integration time is less than one frame time, row n + 1 is reset immediately prior to resetting row n. This is intended to prevent blooming across rows under conditions of very high illumination.	0	N	N
Bits 13:0	Reserved	Do not change from default value.			
R37–0x25 – LINE_VALID Control (R/W)					
Bit 15	Xor LINE_VALID	0: Normal LINE_VALID (default, no XORing of LINE_VALID). Ineffective if continuous LINE_VALID is set. 1: LINE_VALID = "continuous" LINE_VALID XOR FRAME_VALID.	0	N	N
Bit 14	Continuous LINE_VALID	0: Normal LINE_VALID (default, no LINE_VALID during vertical blanking). Bad frame 1: "Continuous" LINE_VALID (continue producing LINE_VALID during vertical blanking).	0	N	N[2]
R38–0x26 – Bottom Dark Rows (R/W)					
Bit 7	Show	The bottom dark rows are visible in the image if the bit is set.	0	N	N
Bits 6:4	Start Address	Defines the start address within the 8 bottom dark rows.	0	N	N
Bit 3	Enable Readout	Enables readout of the bottom dark rows.	0	N	Y
Bits 2:0	Number of Dark Rows	Defines the number of bottom dark rows to be used. (The number of rows used is the specified value + 1.)	7	N	Y
R43–0x2B – Green1 Gain (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10 + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits 6:0*0.03125.	20	Y	N
R44–0x2C – Blue Gain (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10 + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits [6:0]*0.03125.	20	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N
R45–0x2D – Red Gain (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10] + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits 6:0*0.03125.	20	Y	N
R46–0x2E – Green2 Gain (R/W)					
Bits 11:9	Digital Gain	Total gain = (bit 9 + 1)*(bit 10 + 1)*(bit 11 + 1)*analog gain (each bit gives 2x gain).	0	Y	N
Bits 8:7	Analog Gain	Analog gain = (bit 8 + 1)*(bit 7 + 1)*initial gain (each bit gives 2x gain).	0	Y	N
Bits 6:0	Initial Gain	Initial gain = bits 6:0*0.03125.	20	Y	N
R47–0x2F – Global Gain (R/W)					
Bits 11:0	Global Gain	This register can be used to simultaneously set all 4 gains. When read, it returns the value stored in R0x2B:0.	20	Y	N
R48–0x30 – Row Noise (R/W)					

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field		Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R48–0x30 – Row Noise (R/W)					
Bit 15	Frame –wise Digital Correction	By default, the row noise is calculated and compensated for individually for each color of each row. When this bit is set to "1", the row noise is calculated and applied for each color of each of the first two 2 pairs of values and the same values are applied to each subsequent row, so that new values are calculated and applied once per frame.	0	N	N
Bits 14:12	Gain Threshold	When the upper analog gain bits are equal to or larger than this threshold, the dark column average is used in the row noise correction algorithm. Otherwise, the subtracted value is determined by bit 11. This check is independently performed for each color, and is a means to turn off the black level algorithm for lower gains.	0	N	N
Bit 11	Use Black Level Average	0: Use mean of black level programmed threshold in the row noise correction algorithm for low gains. 1: Use black level frame average from the dark rows in the row noise correction algorithm for low gains. This frame average was taken before the last adjustment of the offset DAC for that frame, so it might be slightly off.	0	N	Y
Bit 10	Enable Correction	0: Normal operation. 1: Enable row noise cancellation algorithm. When this bit is set, the average value of the dark columns read out is used as a correction for the whole row. The dark average is subtracted from each pixel on the row, and then a constant is added (bits 9:0).	1	N	Y
Bits 9:0	Row Noise Constant	Constant used in the row noise cancellation algorithm. It should be set to the dark level targeted by the black level algorithm plus the noise expected between the averaged values of the dark columns. The default constant is set to 42 LSB.	2A	N	Y
R89–0x59 – Black Rows (R/W)					
Bits 7:0	Black Rows	For each bit set, the corresponding dark row (rows 0–7) are used in the black level algorithm. For this to occur, the reading of those rows must be enabled by the settings in R0x22:0.	FF	N	N
R91–0x5B – Green1 Frame Average (R/O)					
Bits 6:0	Green1 Frame Average	The frame-averaged green1 black level that is used in the black level calibration algorithm.			
R92–0x5C – Blue Frame Average (R/O)					
Bits 6:0	Blue Frame Average	The frame-averaged blue black level that is used in the black level calibration algorithm.			
R93–0x5D – Red Frame Average (R/O)					
Bits 6:0	Red Frame Average	The frame-averaged red black level that is used in the black level calibration algorithm.			
R94–0x5E – Green2 Frame Average (R/O)					
Bits 6:0	Green2 Frame Average	The frame-averaged green2 black level that is used in the black level calibration algorithm.			
R95–0x5F – Threshold (R/W)					
Bits 14:8	Upper Threshold	Upper threshold for targeted black level in ADC LSBs.	23	N	N
Bits 6:0	Lower Threshold	Lower threshold for targeted black level in ADC LSBs.	1D	N	N
R96–0x60 – Calibration Control (R/W)					
Bit 15	Disable Rapid Sweep Mode	Disables the rapid sweep mode in the black level algorithm. The averaging mode remains enabled.	0	Y	N

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field	Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R96−0x60 − Calibration Control (R/W)				
Bit 12	Recalculate	When set to "1", the rapid sweep mode is triggered if enabled, and the running frame average is reset to the current frame average. This bit is write − 1, but always reads back as "0."	0	Y	N
Bit 10	Limit Rapid Sweep	0: All dark rows can be used for the black level algorithm. This means that the internal average might not correspond to the calibration value used for the frame, so the dark row average should in this case not be used as the starting point for the digital frame-wise black level algorithm. 1: Dark rows 8−11 are not used for the black level algorithm controlling the calibration value. Instead, these rows are used to calculate dark averages that can be a starting point for the digital frame-wise black level algorithm.	0	N	N
Bit 9	Freeze Calibration	When set to "1", does not let the averaging mode of the black level algorithm change the calibration value. Use this with the feature in the frame-wise black level algorithm that allows you to trigger the rapid sweep mode when the dark column average gets away from the black level target.	0	N	N
Bit 8	Sweep Mode	When set to "1", the calibration value is increased by one every frame, and all channels are the same. This can be used to get a ramp input to the ADC from the calibration DACs.	0	N	N
Bits 7:5	Frames To Average Over	Two to the power of this value determines how many frames to average when the black level algorithm is in the averaging mode. In this mode, the running frame average is calculated from the following formula: Running frame ave = old running frame ave − (old running frame ave)/2^n + (new frame ave)/ 2^n. n = frames to average over.	4	N	N
Bit 4	Step Size Forced To 1	When set to "1", the step size is forced to 1 for the rapid sweep algorithm. Default operation (0) is to start at a higher step size when in rapid sweep mode, to converge faster to the correct value.	0	N	N
Bit 3	Switch Calibration Values	Reserved.	0		
Bit 2	Same Red/Blue	When set to "1", the same calibration value is used for red and blue pixels: Calib blue = calib red.	0	N	Y
Bit 1	Same Green	When set to "1", the same calibration value is used for all green pixels: Calib green2 = calib green1.	0	N	Y
Bit 0	Manual Override	Manual override of black level correction. 0: Normal operation (default). 1: Override automatic black level correction with programmed values. (R0x61:0−R0x64:0).	0	N	Y
R97−0x61 − Green1 Calibration Value (R/W)				
Bits 8:0	Green1 Calibration Value	Analog calibration offset for green1 pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by not ([7:0]) + 1). If R0x60:0[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If R0x60:0[0] = 1, this register is R/W and can be used to set the calibration offset manually. Green1 pixels share rows with red pixels.	0	N	Y

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field	Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R98–0x62 – Blue Calibration Value (R/W)				
Bits 8:0	Blue Calibration Value	Analog calibration offset for blue pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by Not ([7:0]) + 1). If R0x60:0[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If R0x60:0[0] = 1, this register is R/W and can be used to set the calibration offset manually.	0	N	Y
R99–0x63 – Red Calibration Value (R/W)				
Bits 8:0	Red Calibration Value	Analog calibration offset for red pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by Not ([7:0]) + 1). If R0x60:0[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If R0x60:0[0] = 1, this register is R/W and can be used to manually set the calibration offset.	0	N	Y
R100–0x64 – Green2 Calibration Value (R/W)				
Bits 8:0	Green2 Calibration Value	Analog calibration offset for green2 pixels, represented as a two's complement signed 8-bit value (if bit 8 is clear, the offset is positive and the magnitude is given by bits 7:0. If bit 8 is set, the offset is negative and the magnitude is given by Not ([7:0]) + 1.) If R0x60:0[0] = 0, this register is R/O and returns the current value computed by the black level calibration algorithm. If R0x60:0[0] = 1, this register is R/W and can be used to manually set the calibration offset. Green2 pixels share rows with blue pixels.	0	N	Y
R101–0x65 – Clock (R/W)				
Bit 15	PLL Bypass	0: Use clock produced by PLL as master clock. 1: Bypass the PLL. Use EXTCLK input signal as master clock.	1	N	N
Bit 14	PLL Power-down	0: PLL powered-up. 1: Keep PLL in power-down to save power (default).	1	N	N
Bit 13	Power-down PLL During Standby	This register only has an effect when bit 14 = 0. 0: PLL powered-up during standby. 1: Turn off PLL (power-down) during standby to save power (default).	1	N	N
Bit 2	clk_newrow	Force clk_newrow to be on continuously.	0	N	N
Bit 1	clk_newframe	Force clk_newframe to be on continuously.	0	N	N
Bit 0	clk_ship	Force clk_ship to be on continuously.	0	N	N
R102–0x66 – PLL Control 1 (R/W)				
Bits 15:8	M	M value for PLL must be 16 or higher.	10	N	N
Bits 5:0	N	N value for PLL.	00	N	N
R103–0x67 – PLL Control 2 (R/W)				
Bits 11:8	Reserved	Do not change from default value.			
Bits 6:0	P	P value for PLL.	00	N	N
R240–0xF0 – Page Register (R/W)				
Bits 2:0	Page Register	Must be kept at "0" to be able to write/read from sensor. Used in SOC to access other pages with registers.	0	N	N
R241–0xF1 – ByteWise Address (R/W)				
Bits 15:0	Bytewise Address	Special address to perform 16-bit reads and writes to the sensor in 8-bit chunks. See "8-Bit Write Sequence".	0	N	N

TABLE 5. SENSOR CORE REGISTER DESCRIPTION (continued)

Bit Field	Description	Default (Hex)	Sync'd to Frame Start	Bad Frame
R242–0xF2 – Context Control (R/W)				
Bit 15	Restart	Setting this bit causes the sensor to abandon the current frame and start resetting the first row. Same physical register as R0x0D:0[1].	0	N	YM
Bit 7	Reserved	Reserved.	0	Y	N
Bit 3	Read Mode Select	0: Use read mode context A, R0x21:0. 1: Use read mode context B, R0x20:0. Bits only found in read mode context B register are always taken from that register.	1	Y	YM
Bit 2	Reserved	Reserved.	0	Y	Y
Bit 1	Vertical Blank Select	0: Use vertical blanking context A, R0x08:0. 1: Use vertical blanking context B, R0x06:0.	1	Y	YM
Bit 0	Horizontal Blank Select	0: Use horizontal blanking context A, R0x07:0. 1: Use horizontal blanking context B, R0x05:0.	1	Y	YM
R255–0xFF – Reserved (R/O)				
Bits 15:0	Reserved	Reserved.	1519		

1. Unless integration time is less than one frame (R0x0B[13, 0]).
2. f enabled by bit 3 (R0x25[14]).