MT9V135 Developer Guide

1/4-Inch VGA NTSC/PAL CMOS Digital Image Sensor

Introduction

Micron's MT9V135 is a VGA format single-chip camera with a 1/4-inch NTSC/PAL CMOS digital image sensor that is a complete system-on-a-chip (SOC) solution. It incorporates sophisticated, on-chip camera functions and is programmable through a simple two-wire serial interface.

This developer guide is a reference for engineers developing applications with the MT9V135. The guide provides information on working with chip registers.

The MT9V135 data sheet should be used along with this guide as a reference for specific register and programming information.

Table of Contents

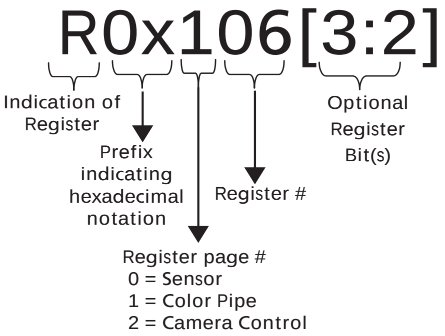
List of Figures

List of Tables

Register Operations

This developer guide refers to various registers that the user reads from or writes to for altering the MT9V135 operation. Hardware registers appear as follows and may be read from or written to by sending the address and data information over the two-wire serial interface.

Figure 1: Register Legend



The MT9V135 was designed to facilitate customizations to optimize image quality processing. Multiple parameters are allowed to be adjusted at various stages of the image processing pipeline to tune the quality of the output image.

The MT9V135 contains three register pages: sensor, colorpipe, and camera control. The register page must be set prior to writing to a register in the page. Example: to write register R0x106 (register 6 in page 1):

- Write the value of “1” to the page map register (0xF0)
- Write the desired value to register R0x06

The sensor maintains the page number once set. The page map register is located at address 0xF0 for all three register pages.

Initializing the MT9V135

Power-up Sequence

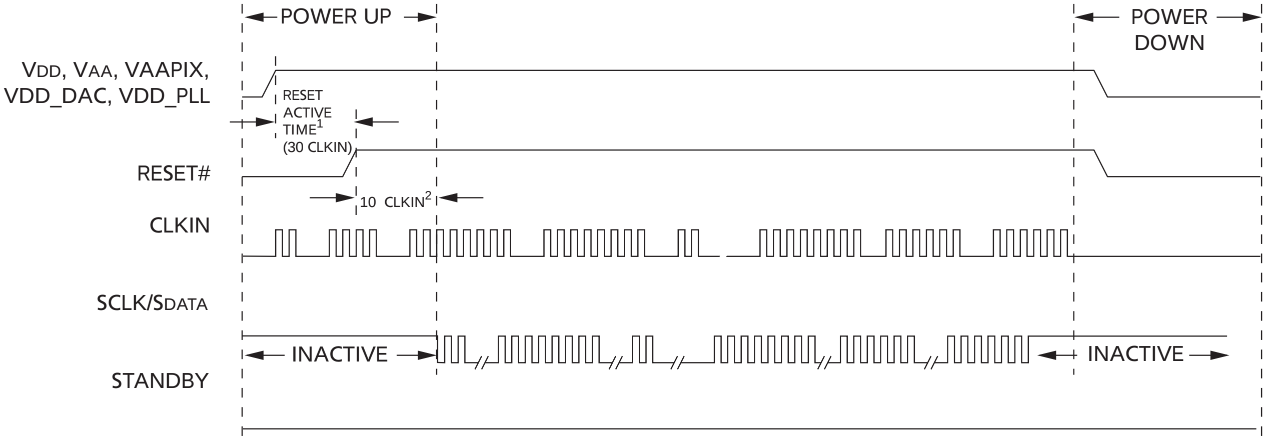
There are no specific requirements for the order in which different supplies are powered on. Once the last supply is stable within the valid ranges specified in Table 1, follow the “Hard Reset Sequence” on page 6 to complete the power-up sequence.

Table 1: Input Voltage Supplies

Voltage Supply	Minimum	Typical	Maximum	Unit
Analog voltage	2.5	2.8	3.1	V
Digital voltage	2.5	2.8	3.1	V
I/O voltage	2.5	2.8	3.1	V

Figure 2 gives an example of the power-up sequence.

Figure 2: MT9V135 Power Up/Down Sequence



Notes:
1. For a safe RESET to occur, CLKIN must be running and it is recommended that STANDBY is LOW during the RESET ACTIVE TIME, as shown in the sequence.
2. After RESET# is HIGH, wait 10 CLKIN rising edges before the two-wire serial interface communication is initiated.

Hard Reset Sequence

RESET# should be asserted for a minimum of 1µs following stabilization of the power supplies. The master clock should be on.

Prior to the first clock edge following assertion of RESET, the status of FRAME_VALID, LINE_VALID, PICXCLK, and outputs DOUT[0:7] are undefined.

With the first rising clock edge during reset, outputs settle to the following defined states:
FRAME_VALID = LINE_VALID = PICXCLK = 0 LOW
D0, D1, D2, D3, D5, D6, D7 = 0 LOW
D4 = 1 (HIGH) HIGH

Following de-assertion of RESET, wait 10 CLKIN cycles before considering the device as active and applying serial interface transactions or toggling STANDBY.

Note: Reset should not be activated while STANDBY is asserted.

Soft Reset Sequence

A soft reset to the camera can be activated by the following procedure:

1. Enable soft reset by setting R0x00D bit 0 and 5. Bit 0 resets the sensor core while bit 5 resets the SOC.
2. Disable soft reset by setting R0x00D = 0x0000.
3. Wait 10 clock cycles before using the two-wire serial interface.

Note: No access to MT9V135 registers—both page 1 and page 2—is possible during soft reset.

Identifying the Chip Version

The sensor version can be determined by reading one of the two ID registers.

```
R0x000 = 0x128D          //Sensor chip version number
R0x0FF = 0x129E          //Sensor chip version number
```

Gamma and Contrast

Because most video systems have been designed to compensate for the nonlinear light-intensity response of the CRT, the camera must output a gamma-corrected image. This means that an exponential gamma curve must be applied to the image before it leaves the camera system so that when it is viewed on a CRT or other modern imaging system, it does not appear too dark.

The MT9V135 image processing chain contains a gamma correction stage. The power-on defaults use a gamma setting of 0.45, with the luma range constricted by the ITU-R BT.656 standard. This limits the range of luma values from 16 through 240. By default, the saturation control is set to 100 percent.

Recommended Settings

The recommended settings use a gamma of 0.45 with a contrast-enhancing S-curve applied. This recommended gamma correction table improves the contrast of the default settings, lowers the black level, and reduces the noise level in low-light conditions.

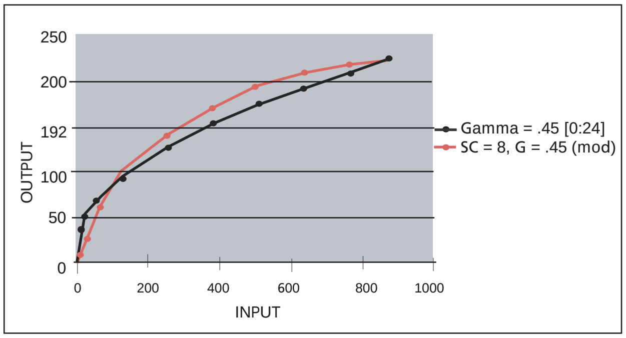
Gamma correction is implemented as a piecewise linear approximation using 11 points at specific, predetermined locations. These points work well for gamma functions. For any 10-bit input luma, an output luma will be interpolated between the two nearest points in the table.

Table 2 indicates the difference in the output values with the two different gamma settings. Figure 3 on page 8 shows this in graphical form.

Table 2: Gamma Settings: Default and S-Curve Factor

Input	Output (Gamma Default, 0.45)	Output (S-Curve, 0.45)
0	0	0
16	36	8
32	50	26
64	68	61
128	93	96
256	127	140
384	152	171
512	174	193
640	192	206
768	208	215
896	224	224

Figure 3: Plot of Gamma Settings: Default and S-Curve Factor



By default, the YCbCr output of the MT9V135 is configured to comply with Rec. ITU-R BT.656 standards (formerly known as CCIR 656).Under Rec. ITU-R BT.656, the range limits for Y are 16–240.

The following formula is used to calculate a gamma table:

$$Y_i = Y_{MAX}\left(\frac{X_i}{896}\right)^y \qquad \text{(EQ 1)}$$

where

$$X_{0....10} = \{0, 16, 32, 64, 128, 256, 384, 512, 640, 768, 896\} \qquad \text{(EQ 2)}$$

$$Y_{MAX} = 224 \quad ITU601/656YCbCr\ or\ RGB\ output \qquad \text{(EQ 3)}$$

A typical value of y is ***y*** = 0.45, which is 1/2.2 (the reciprocal of a standard monitor gamma).

Calculated values are programmed into R0x1DC–R0x1E1 for the gamma table.

The luma offset register R0x134 and the luma clip register R0x135 must remain consistent with the outputs from the gamma table. The luma offset value must be appropriate for the output format, and the luma clip register must be set to pass all the legal luma values for that output format. Table 3 illustrates appropriate values for these registers.

Table 3: Output Register Values

Output Format	Luma Offset R0x34:1	Luma Clip R0x35:1
RGB	[15:8] = 0	N/A
ITU-601/656 YCbCr	YCbCr [7:0] = 16	0xF010 (16 to 240)
Processed Bayer	[15:8] = 0	N/A
Bayer Bypass	N/A	N/A

For convenience, DevWare can calculate the appropriate register settings on the gamma table page by simply choosing ITU Rec. BT-601 and entering the desired gamma factor.

Choosing Gamma Factors

While the following is not intended to be a complete reference on the subject of gamma correction, it provides helpful background information.

Gamma correction in a display system compensates for power function transfer functions, such as occur in a CRT when the CRT converts signal voltage into output intensity. Assuming a signal range of [0.0, 1.0], the transfer function of a color channel is:

$$C_{out} = (C_{in})^{y} \tag{EQ 4}$$

If the gamma function is applied several times with different factors, the net effect on the system (precision issues notwithstanding) is as though a single gamma function is applied with a gamma that is the product of the different factors:

$$C_{out} = ((((C_{in})^{y1})^{y2})^{y3})^{y4} = (C_{in})^{y1y2y3y4} \tag{EQ 5}$$

Consider the following simple example:

- An image is captured by a camera with output gamma correction capability.
- The image is displayed on a CRT that has the standard 2.2 gamma transfer function.
- The goal is to display a faithful rendition of the image on the CRT.

To reproduce the image correctly, the camera must precorrect the image so that its colors are displayed correctly on the monitor. This requires selecting a gamma factor for the gamma correction function of the camera that counteracts the effects of the CRT gamma correction. For the output image to be the same as the input image, the product of the gamma of the camera and the gamma of the CRT must be 1.0. This, in turn, requires that the gamma factor of the camera be set to approximately 0.45 (1/2.2).

Gamma correction factors can occur in a variety of places:

- CRT electronics correct gamma for colors. As noted previously, standard gamma for a CRT is 2.2, although larger gammas can be used. The PAL TV standard uses a standard gamma of 2.8, and gamma correction ranging from 2.35 to 2.55 may be appropriate for some CRTs.
- The graphics hardware in a system can apply a gamma correction curve.
- The standards for representing images or video can require specific gamma correction. For example, Rec. ITU-R BT.709 requires image data precorrection with a gamma of 0.45.
- Additional gamma factors can be applied to make images more pleasing. For example, the 2.2 gamma of the NTSC TV standard is lower than the 2.35 to 2.55 gamma of actual monitors. This improves viewing in dark environments. The same data, viewed in a bright environment, may require an additional gamma correction of 1.1 to 1.2.

Lens Shading and Correction

This section outlines the lens shading basics featured in the MT9V135 and shows how to adjust the lens shading settings.

Camera lenses have signal degradation on sensor periphery due to optical and geometrical factors. Lens shading correction compensates for the signal degradation by digitally gaining pixels on the image periphery.

Lens Shading Approach

The digital gain to correct signal degradation can be expressed as the following:

$$Gain(x, y, color) = 1 + F_{horizontal}(x, color) + F_{vertical}(y, color) + k \times F_{horizontal}(x, color) \times F_{vertical}(y, color) \qquad \text{(EQ 6)}$$

where color = R, G, or B.

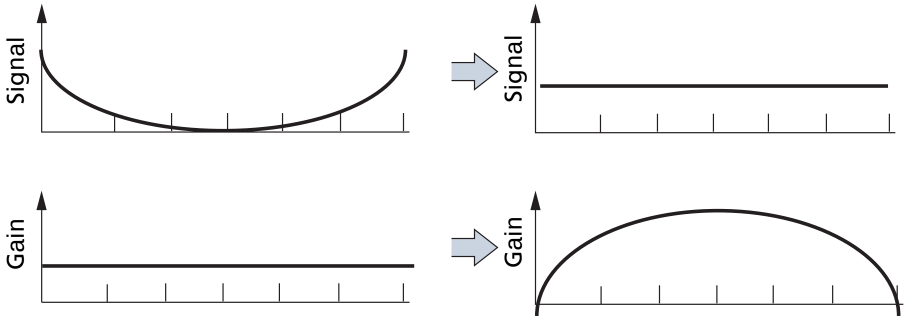
The signal of each pixel is gained as follows:

$$Signalafter\_lc = Signalbefore\_lc \times Gain(x,y,color) \qquad \text{(EQ 7)}$$

In a Bayer sensor, the value of color is a function of (x,y).

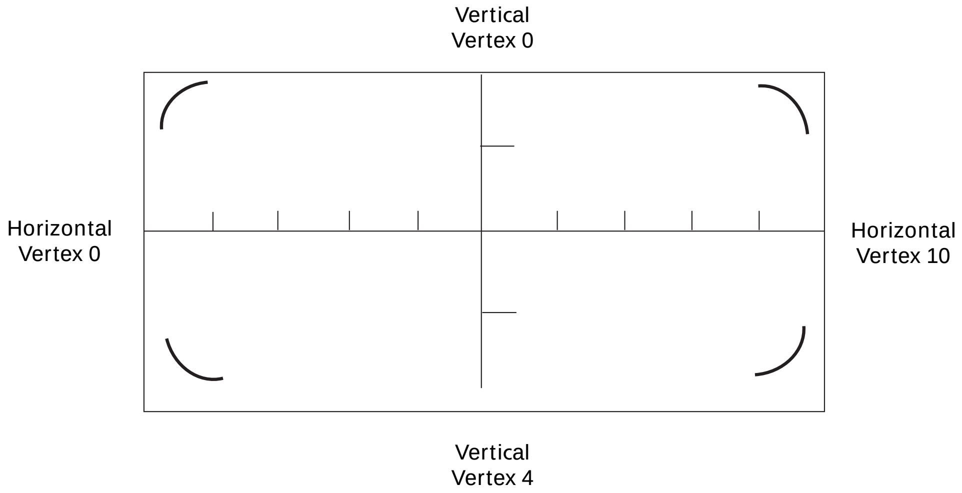
The relationship of the signal and gain before and after are shown in Figure 4. The idea is to have an even signal throughout the lens image.

Figure 4: Signal Before and After Lens Shading



In the MT9V135, $F_{horizontal}(x, color)$ and $F_{vertical}(y, color)$ are piecewise quadratic (PWQ). Eleven vertices divide the sensor area into 10 zones horizontal segments (“pieces”). Vertically, 5 vertices divide the area into 4 segments (see Figure 5 on page 11). The corner parameter “**k**” is defined by R0x180 (Off, Low, Med, High, Highest). Med is the default setting for the corners.

Figure 5: MT9V135 Lens Correction Zones

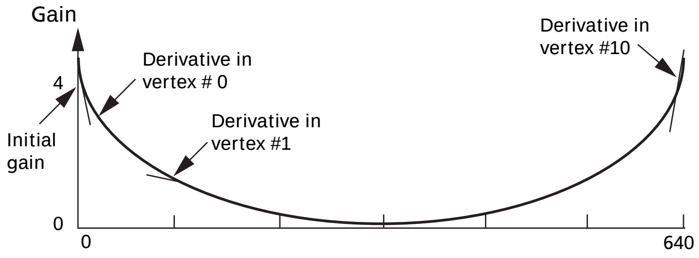


Functions $F_{horizontal}(x, color)$ and $F_{vertical}(y, color)$ are specified for each color (R, G, and B) by programming values in each vertex. For example, consider $F_{horizontal}(x, green)$. Specifying

$F_{horizontal}(x, green)$ starts by setting an "initial gain value" for green in horizontal direction.

Next, the vertex 0 initial value is set. After that, the derivatives $F_{horizontal}(vertex, green)$ for each vertex are set in order from 1 through 10 (see Figure 6).

Figure 6: Lens Correction Kneepoint Vertex Derivatives



Notes:
1. Vertex markers are not drawn to scale.
2. Lens correction occurs early in the imaging pipeline. The image is later horizontally stretched to 720 pixels.

Using DevWare Lens Shading Correction

The following section assumes that DevWare is loaded and that a Micron demo system is being used for lens shading calibration.

Before Starting

- The sensor orientation needs to be defined. If the sensor's column mirror and row mirror bits are set in the final application, they must be set during the calibration. This compensates for off-center or nonsymmetrical lenses.
- Place the sensor in front of a uniformly pigmented, uniformly illuminated, flat matte surface of uniform reflectance. A tint is acceptable, but not so much that R, G, or B contain excessive noise. Best results are achieved with a white or gray light. If possible, use equipment specifically designed to provide a uniform calibration target. Otherwise, verify uniformity by throwing the lens as far out of focus as possible, and moving the sensor up, down, right, left, and rotating it, looking for any variations in the image.
- The sensor, light source, and target cannot move once this procedure starts.

Begin

- To begin, start DevWare. Reset the sensor by pressing the reset button that is located at the top of the DevWare application.
- Use NOTEPAD.EXE to create a **"**lens correction" .ini file, and save it with a name corresponding to the module being calibrated, the date, name, and so on (for example, *lc010105.ini*). This is where the defined LC curves are to be saved.

Presets

Open the Preset dialog box by clicking the preset button. In the Presets window:

- Highlight **Lens Shading Calibration-Set Up** and click **Load**.
  This script performs the following steps:
  - R0x125—Turn off saturation attenuation.
  - R0x105—Disable aperture correction.
  - R0x106—Enable lens correction, disable auto white balance and enable manual white balance, bypass CCM, disable flicker, defect correction and AE.
  - R0x1DC to R0x1E1—Set gamma to unity.
  - R0x02F—Set global gain to 2.
- Load **Lens Correction: Calibration-Flat Curves**.
  These settings enable correction to start from zero gain (R0x181 to R0x1C4). This step is performed only on the initial LC setup. If LC is stopped and later resumed, load the previously saved values.
- Ensure that only **All Writable Registers** is selected in the **Preset -> Options** dialog.
- While developing curves, periodically save your settings. In the **Sensor Control->Lens Correction** page, click on the **Save As** button. Navigate to the file that was created in the **Begin** section. Previously saved settings can be reloaded by loading this file in DevWare.

Begin Vertical Calibration

Click the **control** icon at the top of the screen. This opens the **Sensor Control** window. In the window:

- Click on the **Video Output** selection.
- In the **Digital Video Output** area, select the **by Frames (merged)** radio button. This will enable the display of both odd and even fields.
- Click the **Lens Correction** selection.
- Click **Enable Lens Correction**.
- Set *Scale = 3 Low*. (This sets Kx in R0x180 = 3.)
- Set the Corners setting to “0 Off.” The corners setting can be adjusted later after completing lens correction.
- Click on the **graphs** icon at the top of DevWare.
- Select the **Intensity** tab on the **Analysis Graph**.
- If it is not enabled, press the **Info** button. In the Info panel under Mouse Selection, select **Row**. This displays a red horizontal line on the image. Using the left mouse button, drag the red horizontal line towards the center of the image, watching the DevWare title bar for line location. The line should be placed at location **243.**
- Under Mouse Selection, select **Column**. Using the left mouse button, drag the vertical line towards the center of the image, watching the DevWare title bar for line location. The line should be placed at location **360.**
- Now the values for column **360** are plotted in the Analysis Graph. In the Lens Correction page, click on the **Vertical** radio button. With “Red,” “Green,” and “Blue” checked, check the **Auto Adjust** button. This will automatically adjust the vertical LC setting. When the plot in the **Analysis Graph** has settled, click on **Auto Adjust** again (disabling it).
- With rough calibration done, it is now time to do fine calibration. In the **Lens Correction** page, choose one color channel (R, G, or B) to calibrate first. Use the radio buttons to select only one channel. Complete vertical calibration for that channel before tuning a different channel. Do not select **Horizontal** until all channels have been calibrated under **Vertical**.

Vertical Calibration – Set Initial Values

This section sets the initial value for the selected channel so that with LC on, the first pixel in the column has the same value as the highest-value pixel when LC is off. To see the peak value when LC is turned off, toggle the **Enable Lens Correction** check box on the **Lens Correction** page.

- In the **Analysis Graph**, right click the mouse and choose **Lens Correction Guides**. A vertical guide line indicates the location of each knee. Knees are locations where alters to the slope of the plot on the graph take place.
- Click the left mouse button inside the **Analysis Graph** window to make the x/y indicator appear. Click on the peak value of the channel being calibrated to make the horizontal indicator line pass through the peak value. Use the **Cumulative Intensity** tab on the Analysis Graph, as high precision as necessary for accurate alignment.
- Check **Enable Lens Correction**, then use the **Initial** slider and up and down buttons, “^” and “v.”
- Make the value of the first pixel match the indicator level, without going over. Do not proceed until this is done. This initial value cannot be changed later.

Vertical Calibration – Knee Values

This section makes the entire column the same value as the first pixel. To make the graph flat, bend or tilt it at several points. These points are labeled on the graph as Knee 0, Knee 1, and so on.

The goals for this step are:

1. To have the Analysis Graph line horizontal across the knee point zones
2. With “Enable Lens Correction” unchecked, the highest point in the curved line is touching where the horizontal line would be (if “Enable Lens Correction” is checked).

- Make the graph flat at Knee 0. To tilt the graph up where a knee line crosses it, raise the slider for the same Knee. To tilt the plot down, lower the slider for that knee. For fine adjustments, use the up and down arrows.
- Progress from one knee to the next, adjusting each knee value to flatten the plot at that knee. Backtracking to earlier knees (but not to **Initial**) is possible without starting over.
- Do this calibration for all three color components.

Horizontal Calibration

- After completing all vertical calibrations, perform horizontal calibration in a similar manner, using row **243,** and selecting the **Horizontal** radio button. Work with only one channel at a time, until the R, G, and B channels have all been horizontally calibrated.
- Repeat the same steps as done in **Vertical Calibration** (see above) to obtain better image quality graphs.

Auto Exposure

Auto exposure provides the following functions:

- Target luminance maintenance
  Adjusts a variety of controls within the sensor core and the IFP to maintain a target luminance (computed as a weighted average of pixels over the entire image) in response to changes in scene illumination
- AWB and flicker detection change response
  Modifies and updates sensor settings in response to changes in AWB (analog color gains) and flicker detection (shutter width)
- Color saturation and sharpening control
  Drives decreases in color saturation and sharpening for low illumination levels (The sensor control settings are used as an illumination-level metric.)

Enabling Auto Exposure

Auto exposure is enabled by setting bit 14 in the IFP operating mode register (R0x106[14]).

Auto Exposure Window and Backlight Compensation

Auto exposure defines two windows within the image: the “big” window and the “little” window. The big window is defined by the auto exposure horizontal boundaries register (R0x226) and the auto exposure vertical boundaries register (R0x227).

The little window is defined by the auto exposure center horizontal boundaries register (R0x22B) and the auto exposure center vertical boundaries register (R0x22C). The default region of the little window is within the region of the big window.

Registers of Interest

- R0x226 auto exposure horizontal boundaries register
- R0x227 auto exposure vertical boundaries register
- R0x22B auto exposure center window horizontal boundaries register
- R0x22C auto exposure center window vertical boundaries register

Backlight Compensation Control

The window that defines which pixels are used to compute the image average luminance is controlled by the auto exposure backlight compensation control, in R0x106[3:2].

A value of “0” selects the big window; a value of “1” selects the little window; and a value of “2” or “3” selects backlight compensation mode, where the pixels within the little window are weighted four times more heavily than the pixels in the region between the big and little windows.

Registers of Interest

- R0x106[3:2] auto exposure backlight compensation control
- R0x106[3:2] = 0 big window
- R0x106[3:2] = 1 little window
- R0x106[3:2] = 2 backlight compensation mode
- R0x106[3:2] = 3 backlight compensation mode

Window Coordinates

Each of the window boundary registers is split into 2 bytes. The low byte of the horizontal registers controls the window left coordinate, and the high byte controls the window right coordinate. The low byte of the vertical registers controls the window top coordinate, and the high byte controls the window bottom coordinate.

The coordinates themselves represent a fraction of the width (for horizontal coordinates) and the height (for vertical coordinates) of the currently viewed image. Calculate this fraction by dividing the byte value by 128 (0x80).

- A value of “0” is interpreted as the leftmost (horizontal) or topmost (vertical) boundary of the image.
- A value of “128” (0x80) is interpreted as the rightmost (horizontal) or bottommost (vertical) boundary of the image.

The top boundary of the big window is inset slightly from the topmost boundary of the image, to reduce the influence of the illumination source itself (ceiling lighting or sky) on auto exposure.

Target Luminance

The target luminance is contained in the low byte of register R0x22E, and the half range of the target luminance is in the high byte of register R0x22E. As long as the time-averaged luminance (low byte of R0x24D) is in the range of R0x22E[7:0] ±R0x22E[15:8], auto exposure is stable and the exposure criterion has been met.

Registers of Interest

- R0x22E[6:0] target luminance
- R0x22E[15:8] half range of the target luminance
- R0x24D[7:0] time-averaged luminance monitor

Extreme Low Light Condition

If the scene illumination drops to a level that is beyond the range of auto exposure to bring the average luminance into range, it will go into a standby state, waiting for the scene illumination to improve, or for the target luminance (under host control) to be lowered to a level that can be satisfied.

Black Level

The black level, which provides a margin for noise undershoot, is set in the sensor core register field R0x030[9:0] (row noise). In general, the value of this register field should not be changed. If it is changed, a corresponding register field in the color pipeline, R0x13B[9:0] (black level subtraction), should be changed so that the difference between R0x030[9:0] and R0x13B[9:0] is maintained.

In normal operation, this difference is 0, and the values are identical.

Note: Modifying the black level from its default values can have a negative effect on the downstream elements of the color pipeline, in some cases reducing the maximum pixel values that will be output. It can also skew the operation of some range thresholding metrics within the pipeline itself.

Registers of Interest

- R0x030[9:0] row noise
- R0x13B[9:0] black level subtraction

Flicker Avoidance

Flicker avoidance is achieved by making the shutter width a multiple of the flicker frequency. The user must indicate to the auto exposure unit how many line times constitute an integration time of 1/30 of a second for 60Hz flicker avoidance, or 1/25 of a second for 50Hz flicker avoidance. The default values for these registers are calculated based on the default line time achieved with a 27 MHz clock.

Abatement

It is possible to avoid flicker by setting the flicker mode to a manual 50Hz or 60Hz mode.

To set the default flicker state, do the following:

For 60Hz:

1. Write “1” to R0x25B[0] – put flicker control into manual mode
2. Write “1” to R0x25B[1] – choose 60Hz
3. Wait for the status bit (R0x25B[15]) to indicate “1” (60Hz mode)

For 50Hz:

1. Write “1” to R0x25B[0] – put flicker control into manual mode
2. Write “0” to R0x25B[1] – choose 50Hz
3. Wait for the status bit (R0x25B[15]) to indicate “0” (50Hz mode)

Note: Flicker cannot be avoided when auto exposure goes to Zone 0 (when the lighting is very bright). Auto exposure zone can be monitored using R0x23F.

Color Correction

Automatic White Balance (AWB) Overview

The white balance module adjusts image colors when illumination changes. White balance contains a proprietary measurement engine and a compute engine. The measurement engine processes the image stream to detect the type of illumination. The compute engine derives values for sensor gains, CCM coefficients, and SOC digital gains from these statistics.

Figure 7: AWB Sensor Control

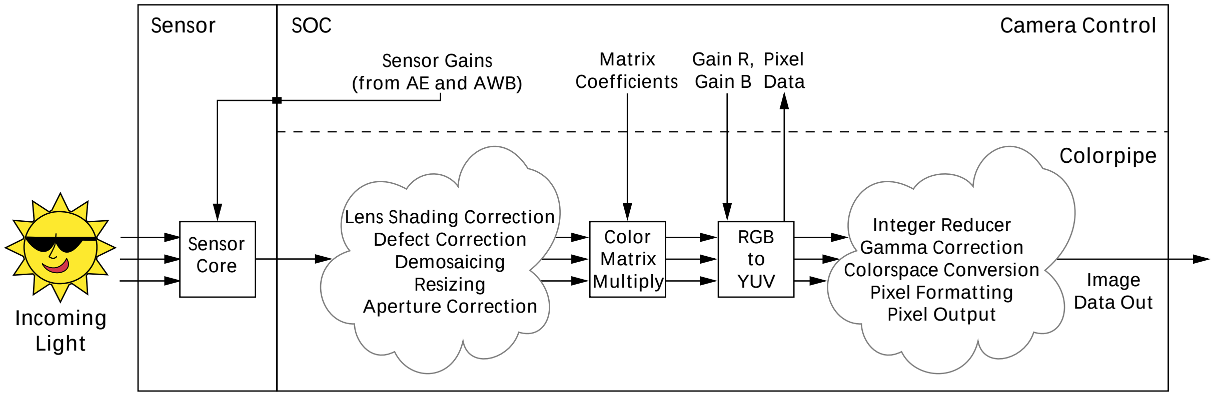


Figure 7 on page 17 shows where the controls available to the white balance unit fit into the sensor (the sensor red and blue gains) and the colorpipe (the CCM coefficients and the SOC digital gains). It also shows the place in the colorpipe where the AWB measurement engine (ME) unit obtains color data for generating statistics.

Figure 8 expands upon this and adds the processing flow that occurs in the white balance unit between frames.

Throughout a frame, the AWB ME collects data from the pixel stream and computes a set of statistics used by the other white balance computations.

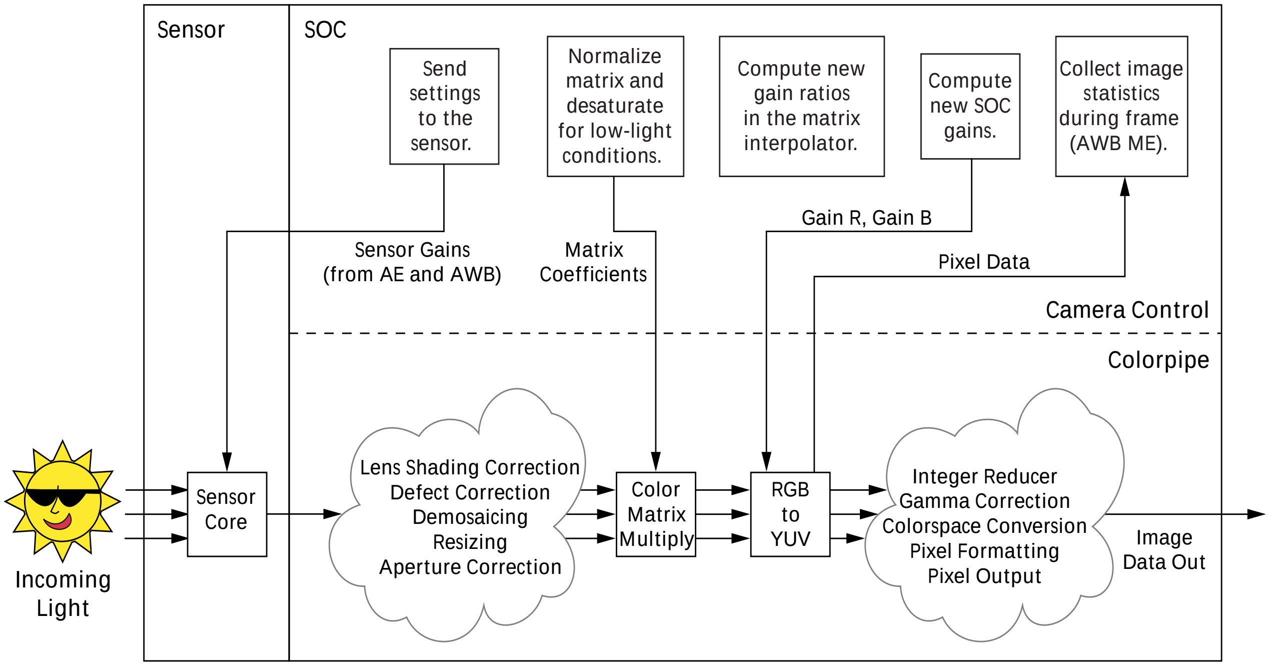
At the end of a frame, white balance begins by computing new SOC digital gains from the AWB ME statistics. It also applies a smoothing function to avoid abrupt changes in color.

Next, white balance determines if the sensor gain ratios need to change, and updates them accordingly.

This is followed by matrix normalization (if enabled) and matrix desaturation in low light conditions (if appropriate).

Finally, white balance sends any new sensor gains to the auto exposure unit, which writes them to the sensor before the next frame begins.

Figure 8: AWB Flow Processing



White Balance Settings

Simple AWB, allowing the auto white balance algorithm to adjust on its own, is the easiest approach to “preset” white balance settings. In the worst case, this takes several seconds to converge—128 frames for the matrix to move between extremes, plus a few more frames for the algorithm to fine tune the results. The slow speed is due to the algorithm switching between predominantly red-rich light, such as incandescent light, and predominantly blue -rich light, such as sunlight.

AWB Speed and Stability

In video or web camera applications, AWB should be relatively slow to avoid sharp changes in color due to slight scene changes, such as people walking by, or the camera panning a room.

The higher the frequency of AWB adjustments (that is, fewer frames between adjustments), the faster the AWB gains can move when the lighting changes. The larger the contribution of the newly computed SOC digital gain to the value actually used by the SOC (smaller values for the control), the faster the SOC digital gains adapt.

If the AWB control rates are set too fast, small changes to an image can shift the color balance measurements and cause AWB to react prematurely. The net effect is that the color balance settings appear twitchy—little things cause sweeping change. This is especially true if the digital gains react too slowly (see above description), as small, transient scene changes can trigger oscillation.

Setting the controls to react too slowly can lead to lengthy convergence times. Ensure that a suitable balance is identified for your implementation.

To accelerate AWB convergence when using slower AWB settings: if the host can determine that a large AWB change is needed, reset the AWB position to the midpoint (position 64) by switching to manual white balance for one frame by setting R0x106[15] = 1. When manual white balance is subsequently turned off by clearing R0x106[15], AWB will resume from this central position. This is helpful for transitions between extreme color balance points, such as switching between incandescent lighting and sunlight.

If AWB speed or stability are unsatisfactory for your application, contact Micron for further information.

AWB/CCM Tuning

To optimize performance and customize the MT9V135, a thorough calibration and setup has to be performed at least once to get best results. However, since the device will work in various environments, a recalibration might be necessary. For instance if the lighting condition will be singular, a narrower AWB might be applied.

Sequence

It is best to follow a procedure in the following sequence to achieve best results in the shortest time period.

Sequence for CCM Setup

1. Set Row Noise/No Row Noise Correction.
2. Set Black Level.
3. Set Gamma on grey chart to as linear as possible."
4. Set Lens Correction.
5. Set AWB.
6. Set CCM.
7. Set Adjustment Speed.

Sequence for Operating Parameters

1. Set horizontal direction.
2. Set Pedestal.

CCM Setup

There are two variations on tuning AWB and CCM for Micron Sensors.

- Color-Chart Overlay
- CCCM tuning tool.

This document focuses on the Color-Chart Overlay method and makes the following assumptions:

- DevWare is loaded and a Micron demo system is being used.
- Lens Correction Calibration (LC) is performed prior to AWB/CCM tuning.
- For most sensors, the flip/mirror bits must have the user's settings. This is true for lens correction also.

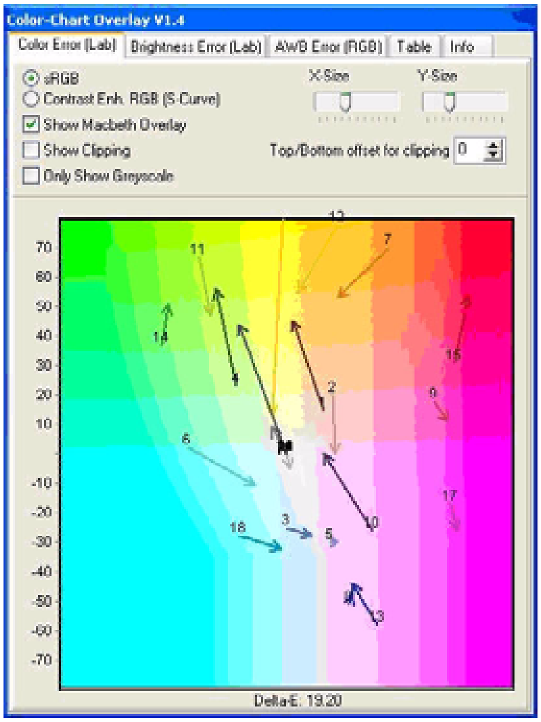
AWB Tuning

AWB is tuned first, followed by CCM.

Initialization

1. Set up the sensor in a light box to view a Macbeth chart. The chart should be centered within the sensors field of view. Ideally, the light source should be behind the imager, not on top (that is, for a Judge II light box, remove the light source from the box top and stand it up behind the sensor to be tuned).
2. Launch DevWare and reset the sensor.
3. Click on the Plug-Ins->Color-Chart Overlay menu item; this brings up the Color-Chart Overlay dialog box. On the Color Error (Lab) page, verify that sRGB is checked and that Show Macbeth Overly is enabled. See Figure 9 below.

Figure 9: Color-Chart Overlay: Color Error (Lab) Page



● sRGB
○ Contrast Enh. RGB (S-Curve)
☑ Show Macbeth Overlay
☐ Show Clipping
☐ Only Show Greyscale
Top/Bottom offset for clipping 0
Delta-E: 19.20

4. Align the reference color squares in the Devware field of view with the Macbeth chart squares; adjust the X-Size and Y-Size so the reference squares do not touch the black sides of the Macbeth chart. See Figure 10 on page 22.

Figure 10: Color-Chart Overlay: Reference Squares Aligned with Macbeth Chart

Sensor Setup

1. Switch the sensor to the snapshot context, Context B.
2. Disable scaling/zooming. The image should be full size. Set the mirror/flip bits if needed.
3. Set saturation to 100 percent. Bring up the Sensor Control->Gamma, Saturation dialog box. Set the Saturation slider to 100 percent (R0x125[5:3] = 0).
   If 150% saturation is used, the Contrast Enhanced RGB (S-Curve) setting can be used in the Color-Chart Overlay->Color Error (Lab) dialog box.
4. Set Gamma to 0.45. In the Sensor Control dialog box, Gamma, Saturation page, move the Gamma Correction Table slider to 0.45.
5. The initial tuning should be done with the S/W Gamma radio button checked. This allows AWB/CCM targets to be reached faster. After the initial tuning is complete, the H/W Gamma radio button is checked. H/W Gamma is used to verify final results and make adjustments.
6. Set the Black Correct to “8.” In the Sensor Control dialog box, Gamma, Saturation page, move the Black Correct slider to “8.”
7. Load lens correction values using the Presets dialog box.
8. Set the digital gains to 1.0. In the Register panel, set R0x148[7] to “1.”
9. Using the Devware presets dialog box, load the default Color Correction Matrix for the sensor. Do not load the fast CCM setting that is present in some Devware INI files.
10. Set the Auto Exposure Luma target. It is important to have the correct exposure when doing these procedures.
    a. In the Register panel set Auto Exposure Target and Precision Control R0x22E[7:0] to 0x44.
    b. Add the Auto Exposure Current Luma Monitor (R0x24C) to the watch page by clicking the right mouse button on 0x24C (and clicking on **Add to Watch** menu

item). Monitor this register for the correct Luma value. Wave a hand in front of the sensor if R0x24C is not 0x44.

11. Monitor register R0x212, **Current Color Correction Matrix Position**. Add this register to the watch page by right-clicking on the register in the Register panel.
12. Verify that R0x106 is set to default (AWB, AE enabled, CCM bypassed, LC enabled).
13. Set the **Delta Sensor Core Gain Ratio** register (R0x25F) equal to 0.

Procedure

The goal is to adjust the Base Sensor Core Gain Ratio register (R0x25E) to minimize the AWB error. The procedure is done twice for two light temperatures: Daylight (D65, 6500 degrees Kelvin) and Incandescent (A, 2850 degrees Kelvin).

1. Start the procedure with Incandescent light.

- For Incandescent, set R0x224 = 0000.
- For Daylight, set R0x224 = 0x7F7F.
  - 1a. Using the Color-Chart Overlay - AWB Error (RGB) page.
  - 1b. Using the Analysis Graph-RGB chart.

Use the Color-Chart Overlay method to perform the AWB tuning procedure; this method gives a numerical value (that is, the Mean Error on the AWB Error (RGB) page) for tuning precision. If desired, the Analysis Graph can later be used to check results.

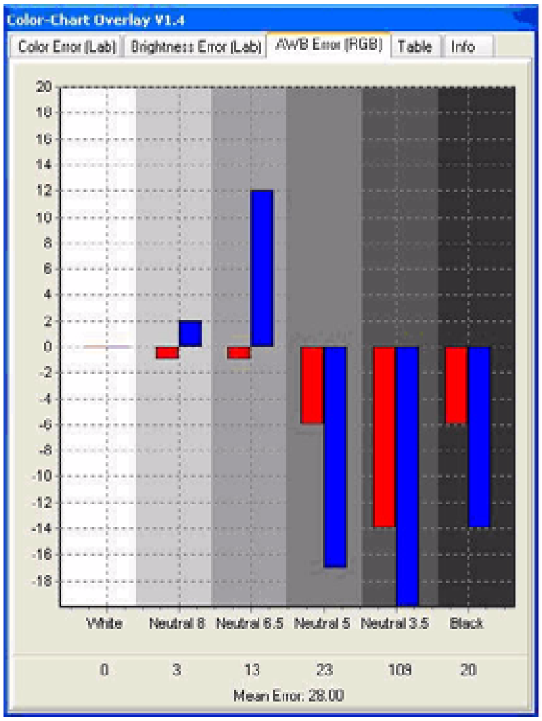
Using the Color-Chart Overlay - AWB Error (RGB) page

1. In the Color-Chart Overlay dialog box, click on the AWB Error (RGB) page. See Figure 11 below.
2. In the Register panel, adjust R0x25E to center the red and blue color bars as close to zero as possible. When a new value for R0x25E is entered, toggle R0x106[15] to 1, then 0 to load the AWB values (a checkbox to do this is located on the Sensor Control->White Balance->Color Correction Matrix page - Manual WB). R0x106[15] must be toggled; otherwise register values are not loaded into the SOC.

Note: R0x228[12] can also be used to load the new AWB (or CCM) values (this will reload the matrix based on the position in R0x2F5).

3. Wait for the AWB to settle before adjusting R0x25E again; monitor register R0x276 for settled values.
4. Focus adjustments on the middle 3 gray Macbeth chart squares (that is, Neutral 8, Neutral 6.5 and Neutral 5) while tuning. The outer squares may be thrown off by lens roll-off.
5. The Mean Error value (located at the bottom of the AWB Error (RGB) page), should be as low as possible (values of 4 to 6).

Figure 11: Color-Chart Overlay: AWB Error (RGB)



White	Neutral 8	Neutral 6.5	Neutral 5	Neutral 3.5	Black
0	3	13	23	109	20

Mean Error: 28.00

Using Analysis Graph - RGB

1. Press the Graphs button to bring up the Analysis Graph - RGB window.

Figure 12: Analysis Graph - RGB Window Showing R,G,B Components of the Macbeth Chart

2. Click on the Intensity tab.
3. In the Info panel, click on **Mouse Selection->Row** to display a horizontal line on the image. Position the line through the Macbeth chart's lowest color row (the white, neutral, … black squares). The line should intersect the reference patches displayed by the Color-Chart Overlay.
4. In the Registers page, alter R0x25E to adjust the red and blue color lines as close as possible to the green line. When a new value for R0x25E is entered, toggle R0x106[15] to 1, then 0 to load the AWB values (a checkbox to do this is located on the Sensor Control->White Balance->Color Correction Matrix page - Manual WB). R0x106[15] must be toggled otherwise register values are not loaded into the SOC.

Note: R0x228[12] can also be used to load the new AWB (or CCM) values. This will reload the matrix based on the position in R0x2F5.

5. The AWB setting is best when:
   a. the R,G,B values are at the same level and
   b. the R,G,B values for each square line up with the reference patch values (that is, when the RGB value on the side of the reference patch line up with the reference patch R,G,B values).
6. Monitor register R0x212 for settled values before doing another adjustment.
7. Once the R0x25E value for Incandescent is determined, record that number to a safe place and switch lighting to Daylight (switch R0x224 to 0x7F7F). Do the same procedure using Daylight and record that number.

Final Results

Two values are recorded, for example:

- Daylight: D650x486E
- Incandescent: A0x6836

1. Compute the resulting Base Matrix Gain Ratio (R94:2) and Delta Matrix Gain Ratio (R0x25F) values:

Base (BLUE)	(0x48 + 0x68)/2	= 0x58	R0x25E Result: 0x5852
(RED)	(0x6E + 0x36)/2	= 0x52	
Delta (BLUE)	ABS (0x48 - 0x68)	= 0x20	R0x25F Result: 0x2038
(RED)	ABS (0x6E - 0x36)	= 0x38	

Note: ABS = Absolute Value.

2. Save the resulting R0x25E and R0x25F values to an INI file. They will be used in the CCM adjustment.

To check the results, perform the following steps:

1. Set R0x224 to 0x7F00 (to allow full AWB adjustment).
2. Enable Digital Gains (R0x148[7] = 0).
3. Write the Base and Delta values to their respective registers. Toggle the MWB matrix using R0x106[15].
4. S/W Gamma can remain enabled throughout this process. H/W Gamma can be used to check the final results for both AWB and CCM.
5. Using the Color-Chart Overlay, check the AWB Error (RGB) page for both Daylight and Incandescent. A Mean Error value of 6 to 7 is considered good.
    a. Adjust the Blue and Red values as necessary using the procedure in "Using Analysis Graph - RGB" on page 29.

Color Correction Matrix Tuning

Initialization

1. For CCM tuning, follow the steps 1 through 4 in the section on Initialization of "AWB Tuning" on page 21. Then follow steps 1 through 8 in "Sensor Setup" on page 22.
2. Load the Base and Delta Sensor Core Gain Ratios (R0x25E, R0x25F) that were determined in AWB tuning.

Procedure

The goal of this procedure is to adjust the Color Correction Matrix so that Macbeth chart squares 13, 14 and 15 (Blue, Green, Red) are as close as possible to reference values. If possible, square 2 (Light Skin) should also be close to reference.

The procedure is duplicated for two light temperatures:

- Incandescent (A, 2850 degrees Kelvin).
- Daylight (D65, 6500 degrees Kelvin)

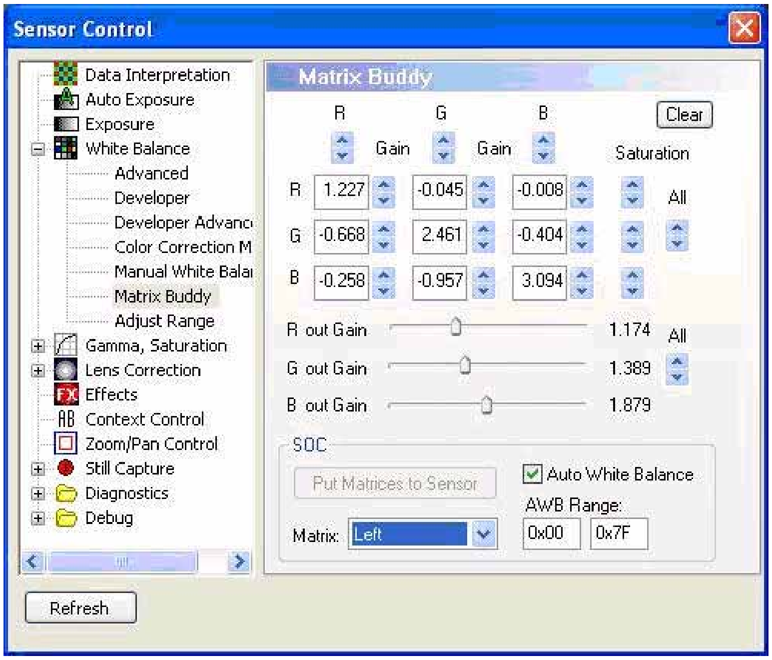
1. Start the procedure with Incandescent light.

- For Incandescent, set R0x224 = 0000.
  Incandescent uses the left Color Correction Matrix.

- For Daylight, set R0x224 = 0x7F7F.
  Daylight uses the right Color Correction Matrix.

2. Adjust the CCM values using one of the two methods given below.
   - Using the Color-Chart Overlay - Color Error (RGB) and the Table pages
     Use the Color-Chart Overlay method to perform the CCM tuning procedure; this method gives a numerical value (that is, the Delta-E value on the Color Error (Lab) page) for tuning precision.
   - Using the Analysis Graph-RGB chart
     If desired, the Analysis Graph can later be used to check results.
3. Bring up the Sensor Control Panel->White Balance->Matrix Buddy page. AWB should be enabled with the AWB range 0 to 0x7F. See Figure 5 below.
4. For the Incandescent light source, set the Matrix to left. For Daylight light source, set the Matrix to right.
5. Using either of the measurement methods below, adjust the matrix. After a single adjustment, press the Put Matrices to Sensor button to load the SOC matrices. Monitor register R0x212, Current Color Correction Matrix Position, for AWB settling.

Figure 13: Matrix Buddy Page



	R	G	B
R	1.227	-0.045	-0.008
G	-0.668	2.461	-0.404
B	-0.258	-0.957	3.094

R out Gain 1.174
G out Gain 1.389
B out Gain 1.879

☑ Auto White Balance
AWB Range: 0x00 0x7F
Matrix: Left

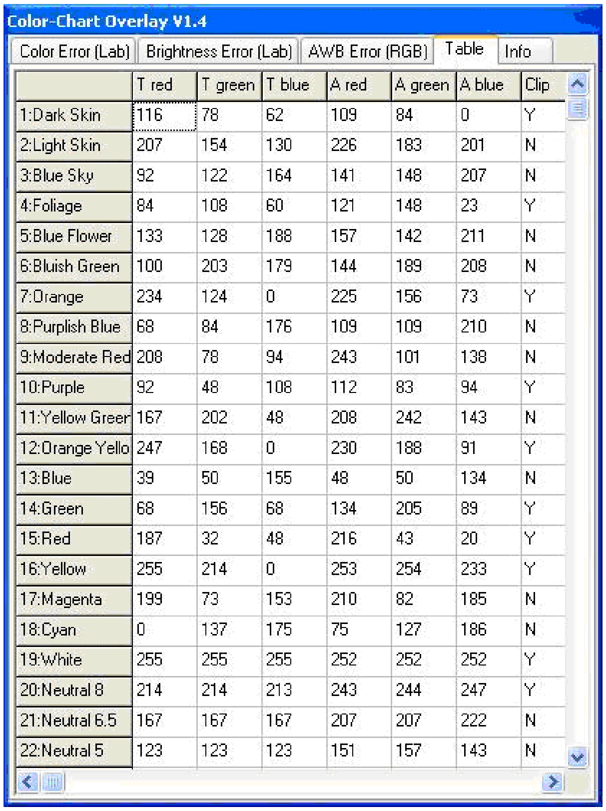
Using the Color-Chart Overlay - Color Error (RGB) and the Table pages

1. In the Color-Chart Overlay dialog box, click on the Color Error (RGB) page. Figure 9 on page 21.

Each of the Macbeth chart colors are represented by an arrow and a number. Numbers 13, 14 and 15 correspond to red, green and blue. Important considerations for this page are the length of the arrow and the direction of the arrow. The shorter the arrow, the better. Ideally, the arrow direction should correspond to the color of the patch (that is, blue arrow pointing to blue).

2. Using the Sensor Control->White Balance->Matrix Buddy page, adjust the matrix for each color component (R,G,B) of each square (R,G,B and others). Press the Put Matrices to Sensor button to write new values to the sensor.
3. Concentrate only on the red, green and blue squares.
4. Monitor register R0x212 for settled values before doing another adjustment.
5. To view the actual RGB levels versus the reference values, look at the Tables page (See Figure 14 below).
6. The T values are the theoretical (reference) values, the A values are the actual values. Ideally, the reference and actual values should match. More realistically, the differences between the reference and actual values for the three color component should match.

Figure 14: Color-Chart Overlay: Table Page



	T red	T green	T blue	A red	A green	A blue	Clip
1:Dark Skin	116	78	62	109	84	0	Y
2:Light Skin	207	154	130	226	183	201	N
3:Blue Sky	92	122	164	141	148	207	N
4:Foliage	84	108	60	121	148	23	Y
5:Blue Flower	133	128	188	157	142	211	N
6:Bluish Green	100	203	179	144	189	208	N
7:Orange	234	124	0	225	156	73	Y
8:Purplish Blue	68	84	176	109	109	210	N
9:Moderate Red	208	78	94	243	101	138	N
10:Purple	92	48	108	112	83	94	Y
11:Yellow Greer	167	202	48	208	242	143	N
12:Orange Yello	247	168	0	230	188	91	Y
13:Blue	39	50	155	48	50	134	N
14:Green	68	156	68	134	205	89	Y
15:Red	187	32	48	216	43	20	Y
16:Yellow	255	214	0	253	254	233	Y
17:Magenta	199	73	153	210	82	185	N
18:Cyan	0	137	175	75	127	186	N
19:White	255	255	255	252	252	252	Y
20:Neutral 8	214	214	213	243	244	247	Y
21:Neutral 6.5	167	167	167	207	207	222	N
22:Neutral 5	123	123	123	151	157	143	N

Using Analysis Graph - RGB

1. Press the **Graphs** button to bring up the Analysis Graph - RGB window.

Figure 15: Analysis Graph of R, G, B Macbeth Chart Squares

2. Click on the Intensity tab.
3. In the Info dialog box, click on **Mouse Selection>Row** to display a horizontal line on the image. Position the line through the Macbeth chart's third row (blue, green, red, … cyan).
   The goal is to have the similar relative values for R,G,B (that is, the offset from reference to actual is the same for R,G, B values). In Figure 15, the reference offset is the flat R, G or B line. The actual value is the distance between reference and the high/low measurement. For the third square, the R offset (that is, distance from reference to actual) is larger than the B offset and smaller than the G offset.
   Ideally, the actual R,G,B values should match the reference values but this is not always possible.
4. Using the **Sensor Control>White Balance>Matrix Buddy** page, adjust the matrix for each color component (R,G,B) of each square (R,G,B and others). Press the **Put Matrices to Sensor** button to write new values to the sensor.
5. Concentrate only on the red, green, and blue squares.
6. Monitor register R0x212 for settled values before doing another adjustment.
7. Once the CCM values for Incandescent are determined, save the current matrix values (that is, **Put Matrices to Sensor**). Save the CCM to an INI file by going to the **White Balance>Color Correction Matrix** page and pressing the **Save As** button.
8. Continue the CCM tuning by switching the light source to Daylight. Switch R0x224 to 0x7F7F and to switch the Matrix to Left. Perform the same procedure using Daylight. Then view results with Gamma set to H/W. Continue tuning for final results.

Final Results

To check the results:

1. Set R0x224 to 0x7F00 (to allow full AWB adjustment).
2. Enable Digital Gains (R0x148[7] = 0).
3. Enable H/W Gamma.
4. Load the resulting CCM matrix and AWB settings (CCM and AWB settings are automatically saved when the **Save As** button is pressed). Use the **Presets** dialog box to load the saved settings.
5. Test the settings using Incandescent, CWF, TL84, and Daylight light sources. Check the Macbeth chart versus the reference overly using the Color-Chart Overlay and the Analysis Graph.

Fast CCM

Fast CCM improves the response time for AWB and CCM to settle. It reduces the number of steps from 128 to 64.

To create Fast CCM matrices:

1. Load the Base and Delta matrices that were derived from the CCM procedure above.
2. Go to the **Sensor Control>White Balance>Adjust Range** page.
3. Press the **Load from Sensor** buttons for both left and right matrices.
4. Set the AWB position variables (this will center the matrix offsets):

- Change 0x00 to 0x20
- Change 0x7F to 0x5F

5. Press the **Recompute Matrices** button
6. Press the **Put Matrices to Sensor** button
7. Save the resulting register settings using the Sensor Control>White Balance>Color Correction Matrix **Save As** button.

NTSC/PAL Video Encoder

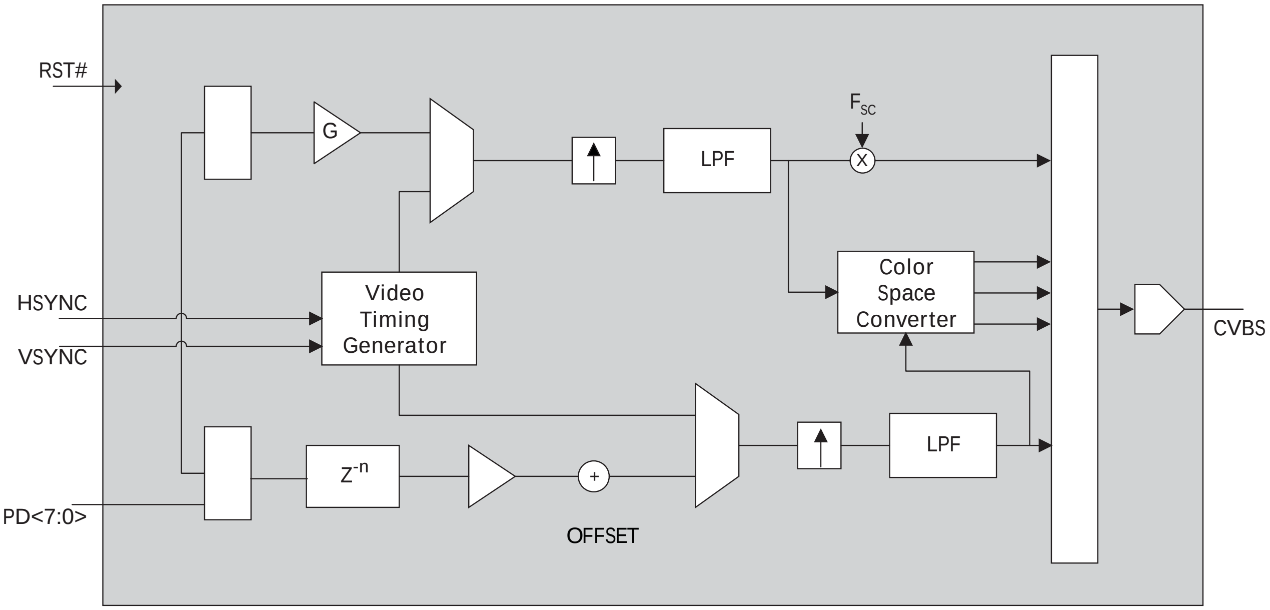
Description

The encoder block, shown in Figure 16, is composed of a video timing generator and a color space converter

The video encoder converts CCIR 601 8-bit multiplexed digital video into RGB, component YCbCr, encoded NTSC, or PAL (BDGHIMN) signals. It contains three 10-bit DACs to support simultaneous S-video and composite video; or component video display. Brightness and contrast control are also provided.

The input clock to the video encoder is the SOC clock at 27MHz (not shown in the diagram).

Figure 16: NTSC/PAL Video Encoder Block Diagram



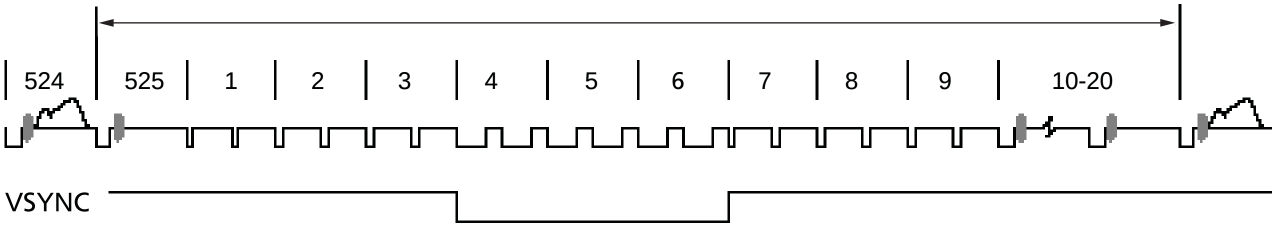
Video Timing Generation

The video encoder timing locks to the external SYNC signals. The decoder automatically detects the input format and locks the internal timing counters to the external synchronization signals. Two types of synchronization inputs are supported: (1) HSYN / VSYNC, and (2) CCIR656 EAV data.

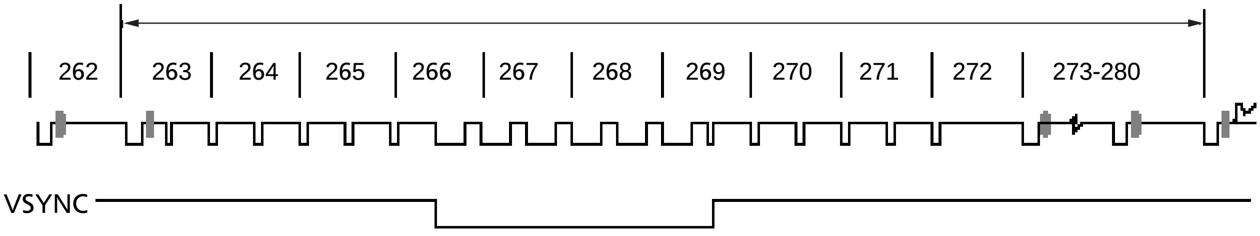
If EAV is present, the video encoder synchronized to the EAV packets according to CCIR656 specifications to generate the video timing. HSYN and VSYN signals are ignored.

If EAV is not present, the video encoder uses the signals presented on HSYN and VSYN for line and field counter increment.

Figure 17: NTSC Vertical Interval Timing

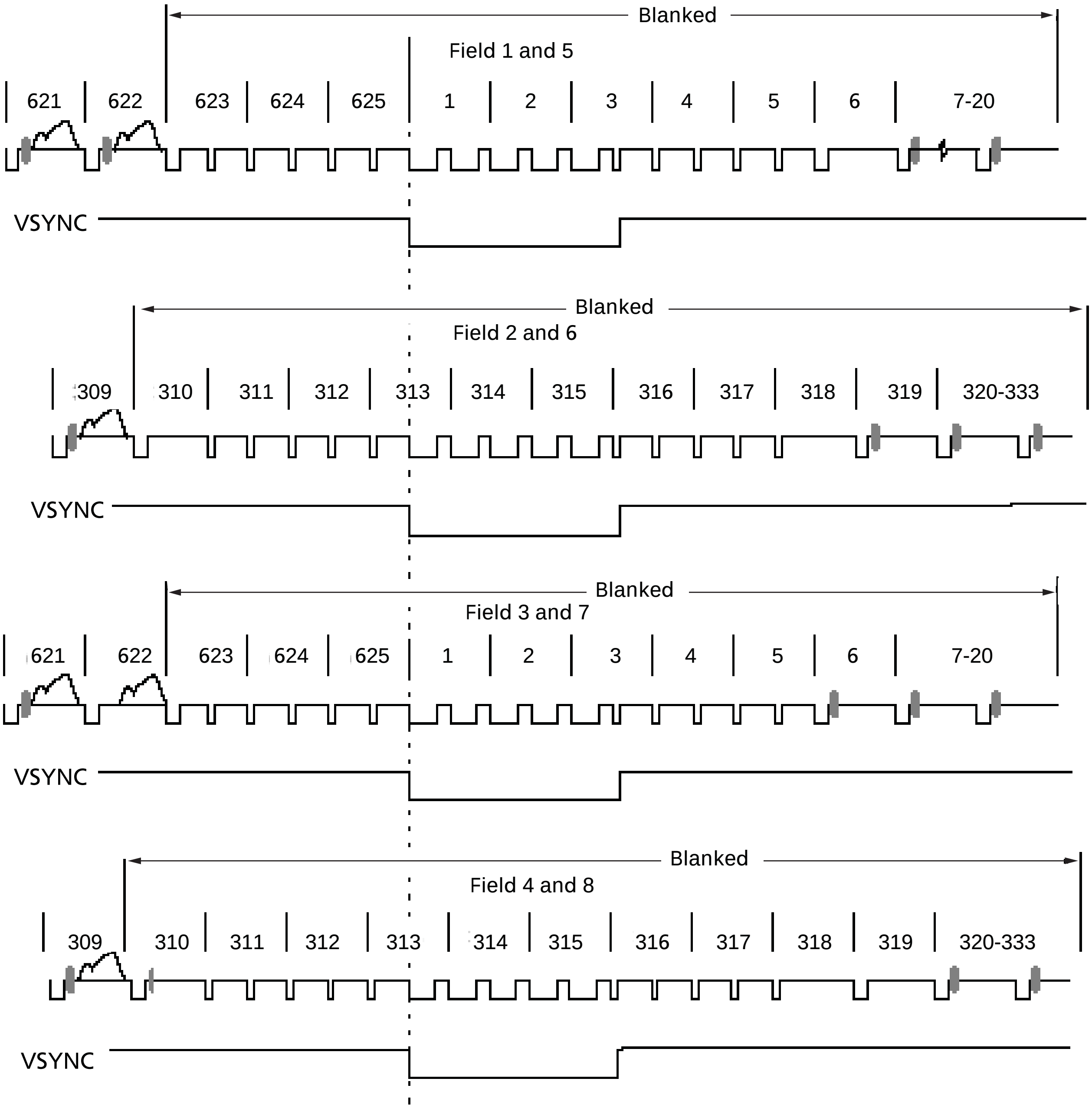


Field One



Field Two

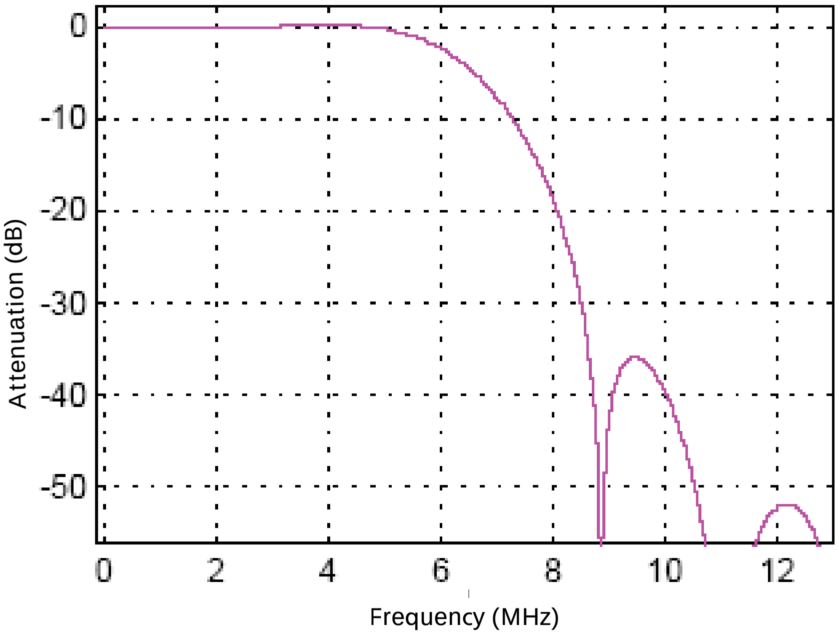
Figure 18: PAL Vertical Interval Timing



Luminance Processing

The luminance, Y, is interpolated to 27 MHz sampling rate through a multi-tap linear poly-phase filter. The filter frequency response is flat from 0 to 5 MHz. Figure 19 shows the luminance filter response:

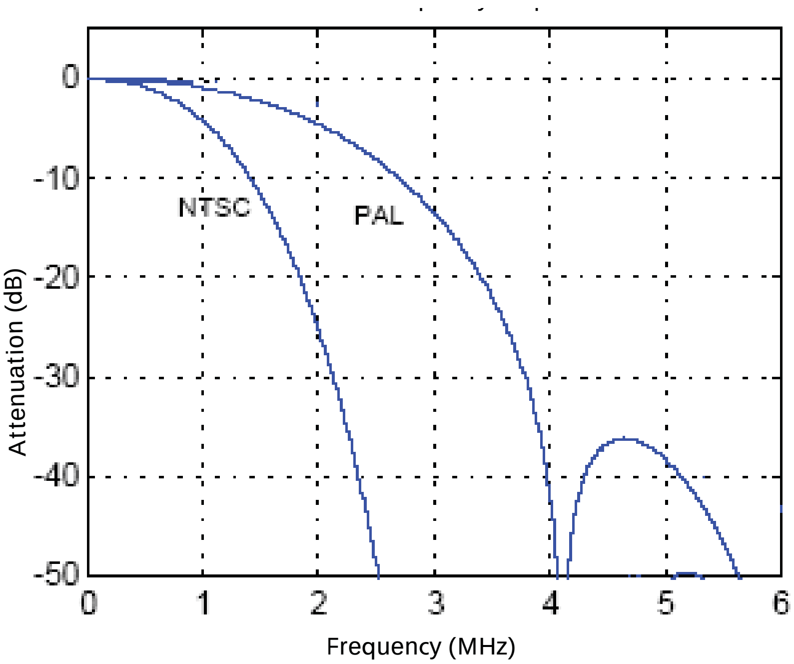
Figure 19: Luminance Filter



Chrominance Processing

The Cb and Cr signals are filtered and interpolated to 27 MHz. Figure 20 shows the chrominance filter response:

Figure 20: Chrominance Filter



Video DAC

The integrated current-based video DAC produces over 100mW. To dissipate this heat from the package, we recommend using the DGND and VDD pins as thermal conduits to PCB thermal-planes, with abundant use of 50mm fat-etch and multiple vias wherever possible.

Appendix A – Frequently Asked Questions

Miscellaneous Questions

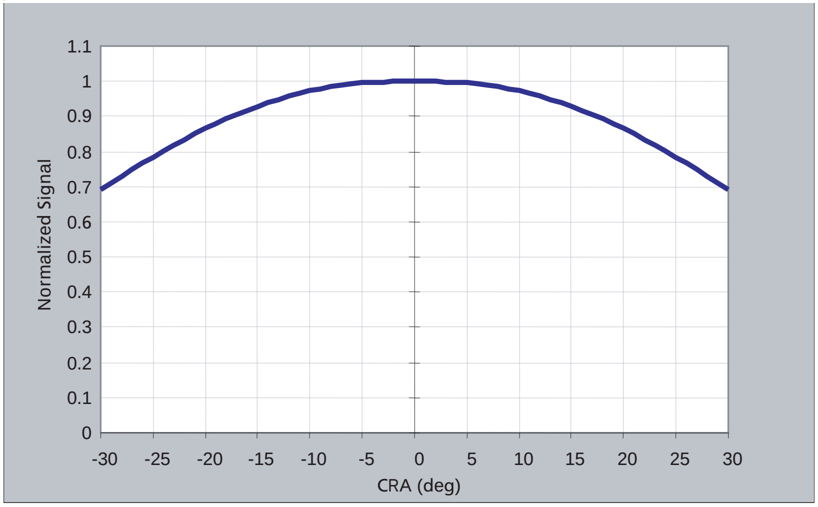
1. **When can the registers be written? When not?**

   Registers may be written any time the MT9V135 is awake and running through the two-wire serial interface. However, to avoid midframe changes in an image's appearance, avoid changing sensor core registers (page 0) or SOC registers (page 1 or 2) during the FRAME_VALID time (these may be changed at any time if appearance is not a factor, for example, upon startup).

2. **What is the chief ray angle requirement for the MT9V135?**

   The chief ray angle (CRA) requirement for the MT9V135 is captured in Figure 21.

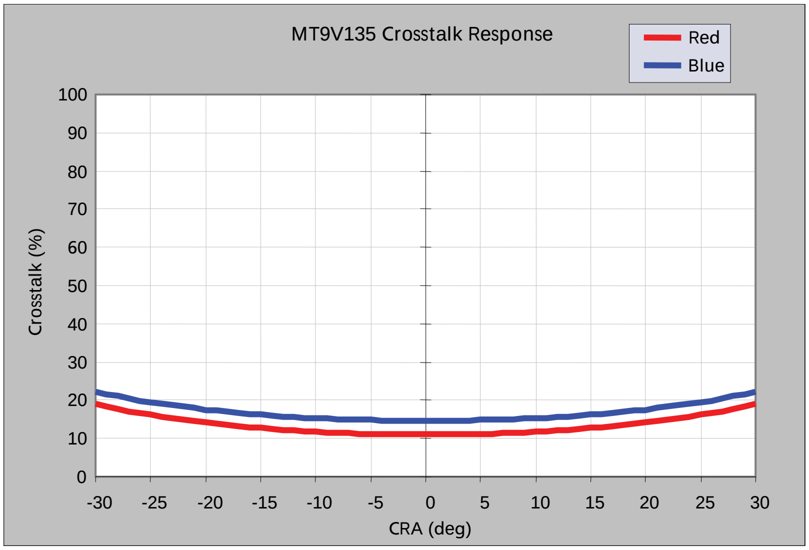
Figure 21: Angular Signal Response for MT9V135



3. What is the crosstalk response for the MT9V135?

The crosstalk response for the MT9V135 is captured in Figure 22.

Figure 22: Crosstalk Response for the MT9V135



4. How should we treat the STANDBY terminal? Should it be connected to GND or open?

STANDBY should not be left open. If it is not used, it should be tied to ground.

5. Is there color suppression in saturated areas?

These devices decrease color saturation for very bright and overexposed pixels. Color suppression is activated when pixel values exceed a relative threshold. R0x125[2:0] specifies the point at which color suppression starts.

The amount of color reduction increases with pixel brightness until 100 percent attenuation is reached at maximum luma.

6. Is there color suppression in low light conditions?

These devices automatically decrease color saturation in low light conditions. R0x23E[12]=1 enables automatic saturation reduction (default).

Registers R0x233 and R0x23E specify the thresholds at which saturation decreases.

The reduction is performed in two steps. First, decreasing values to 75 percent, and second, decreasing values to 50 percent. The two thresholds can be specified with respect to current analog gain, ADC gain, and either of the digital gains.

R0x233[7:0] and R0x23E[9:8] specify the 75 percent setting threshold; R0x233[15:8] and R0x23E[11:10] specify the 50 percent setting threshold.

7. Why is there no change when trying to change sensor gains?

In general, the SOC writes gains (R0x02B–R0x02E), integration time (R0x009, R0x00C), and ADC voltage reference (R0x029) settings to the sensor core. If an attempt is made to change sensor gains, the new values will be overwritten by the SOC.

To set gains manually, disable auto exposure (R0x106[14] = 0), AWB (R0x106[1] = 0), and flicker detection (R0x106[7] = 0).

Switching to or from manual white balance can trigger writes to the sensor core registers.

8. Can NTSC and LVDS be run simultaneously?

No. One or the other may be run, but both should not be run at the same time.

Lens Shading and Correction

9. Where are the lens shading correction control registers located?

The lens shading correction control register is located in R0x180. The lens shading feature can be turned on or off by R0x106[10]. Other lens correction settings can be found in R0x181 to R0x195, and R0x1B6 to R0x1C4. For more details on the registers, refer to the latest MT9V135 data sheet.

10. Does the lens correction need to be recalibrated if vertical or horizontal mirroring is enabled?

Lens shading needs to be recalibrated if mirroring (either vertical, horizontal, or both) is enabled to compensate for off-center or nonuniform lenses. This should be determined before lens correction is performed.

Color Correction

11. How can the auto white balance decision be biased to produce warmer or cooler images?

The R0x2F2 register can skew the measured statistics to make the image look cooler or warmer. This feature is useful for adjusting the tint to customer preference and compensating for display white point.

12. Auto white balance occasionally gets stuck. What should be done when this happens?

AWB ignores tinted areas. Therefore, when illumination color temperature changes drastically or the scene has a colored background, the camera will see the scene heavily tinted and white balance cannot compensate.

To activate AWB, move the camera around. If this is does not activate AWB, relax the settings in R0x22A. R0x22A = 0 configures the SOC to make the picture gray at all times. If this is insufficient, change the settings to R0x21F = 0 and R0x220 = 51200 (0xC800). Switch AWB to gray world mode.

13. When should the color correction matrix be calibrated?

The color correction matrix must be calibrated for each combination of lens and infrared (IR) cut filter. Each camera configuration requires its own set of optimized defaults.

14. What can be done about pictures that come out with the wrong colors?

Make sure the camera defaults are loaded. In DevWare, click on the **Defaults** button in the toolbar. Click the **Presets** button and choose from a list of available preset defaults for various modes.

Check the IR cut filter. A poor IR cut filter can ruin colors, especially with incandescent illumination.

15. What can be done when the white balance is incorrect for a test color chart?

In the MT9V135, AWB is configured to work in the real life mode by default, which does not perform well on certain test charts. Change the AWB mode to gray world.

16. How can I load a new color correction matrix?

Loading color correction matrices involves two processes. First, loading the CCM registers (R0x202–R0x211) that represent the matrices. Second, forcing the new matrices into the matrix interpolator.

Accelerated support for changing the current CCM is built into the MT9V135. Assertion of bit R0x228[12] forces the current CCM to be loaded from the CCM registers during the next vertical blank without the need to enter manual white balance mode (required for other Micron sensors). Bit R0x228[12] is auto-cleared once the CCM has taken effect.

Appendix B – Glossary of Terms

Table 4: Glossary of Terms

Term	Definition
ADC	Analog-to-digital converter.
APS	Active-pixel sensor. The CMOS active-pixel sensor is a second-generation solid state sensor technology that was invented and developed at JPL. CMOS APS technology utilizes active transistors in each pixel to buffer the photo signal. The performance of this technology is comparable to charge-coupled devices (CCDs). Because CMOS APS is inherently CMOS-compatible, it is easy to integrate on-chip timing, control, and drive electronics, reducing system cost and complexity.
AWB	Auto white balance.
Bayer color filter array	Color space jointly developed by Microsoft and Hewlett Packard as a color standard. Refer to http://www.w3.org/Graphics/Color/sRGB.
Bitmap	An image containing only raster information.
BMP	A bitmap format.
CCD	Charge-coupled device—one of the two main types of image sensors used in digital cameras.
CCM	Color correction module or color correction matrix.
Chrominance	Comprises the two components of a television signal that encode color information. Chrominance defines the difference between a color and a chosen reference color of the same luminous intensity.
CMOS	Complementary metal-oxide semiconductor.
CMYK	Cyan, magenta, yellow, and black. A color printing system that uses these colors. See RBG.
DRAM	Dynamic RAM.
ERS	Electronic rolling shutter.
FIFO	First in first out.
fps	Frames per second.
Gamma	Different platforms such as PC, Mac, and UNIX interpret color values slightly differently. Any images that look dark on a PC might look bright on a Mac. The gamma correction is a way to explain how an image should be displayed.
Gamma characteristic	A gamma characteristic is a power-law relationship that approximates the relationship between the encoded luminance in a television system and the actual desired image brightness. With this nonlinear relationship, equal steps in encoded luminance correspond to subjectively approximately equal steps in brightness.
GIF	Graphics interchange format.
Halftone	A gray-scale image represented by bi-level information.
IFP	Image flow processor—performs color recovery and correction, sharpening, gamma correction, lens shading correction, and on-the-fly defect correction.
Interlaced	Graphic data is split (usually into two parts), and displayed alternately line by line.
Interlacing	Also known as progressive display—GIF, JPG and TIFF have supported this feature since the early 1990s.
JPG	Joint photographic experts group—a file format.
LC	Lens shading correction.
LED	Light emitting diode.
LSB	Least significant bit.
Luma	Intensity or brightness component of pixel information.

Table 4: Glossary of Terms (continued)

Term	Definition
Luminance	The quality of being luminous—emitting or reflecting light. Luminosity is measured relative to that of our sun.
MSB	Most significant bit.
MTF	Modulation transfer function—the sharpness of a photographic imaging system or of a component of the system (lens, film, scanner, enlarging lens, and so on) is characterized by MTF.
NTSC	National Television Standards Committee—the North American standard (525-line interlaced raster-scanned video) for the generation, transmission, and reception of television signals.
Output Resolution	The size of the outputted image in the horizontal and vertical directions. Formats include, but are not limited to, VGA, QVGA, CIF, and QCIF.
PAL	Phase Alternating Line, Phase Alternation by Line or Phase Alternation Line—a European standard of displaying analog television signals. It consists of 625 horizontal lines of resolution at 50Hz. Also see NTSC.
Palette	A set of colors that can be used for a specific output device.
Pixels	A short name for picture element. The smallest part that can be displayed on a monitor.
PPS	Passive-pixel sensors (1960s).
PRNU	Pixel response nonlinearity.
Progressive Scan	An image sensor that gathers its data and processes each scan line one after another in sequence. Compare to Interlaced.
Resolution	A measure of how much information is stored in an image. The particular pixel density of an imager in the horizontal and vertical dimensions.
RGB	Red, green and blue. A way to represent color on a monitor. Also see CMYK.
RST	Reset.
SADDR	Sensor address.
SCLK	Serial clock.
SOC	System-on-a-chip.
SRAM	Static random access memory. SRAM is a type of memory that is faster and more reliable than the more common DRAM (dynamic RAM). The term static is derived from the fact that it does not need to be refreshed like dynamic RAM. Due to its expense, SRAM is often used only as a memory cache.
TIFF	Tagged image file format.
True Color	24-bit color. 16,777,216 colors.
VREF	Voltage reference.

8000 S. Federal Way, P.O. Box 6, Boise, ID 83707-0006, Tel: 208-368-3900
prodmktg@micron.com www.micron.com Customer Comment Line: 800-932-4992

Revision History

Rev. A 6/22/2007

- Initial release