DATASHEET – epc635
3D TOF Imager 160 x 60 Pixel

General Description

The epc635 is a fully integrated 3D-TOF imager with a resolution of 160 x 60 pixels (Half-QQVGA). As a system on chip, the epc635 contains next to the CCD pixel-field the complete control logic to operate the device. The output of the chip is 12 bit DCS distance data per pixel, which are accessible through a high-speed digital 8-bit parallel video interface.

Only few additional components are needed to generate a complete 3D camera. Depending on illumination power and optical design, a resolution in the millimeter range for distances up to dozens of meters is feasible. Up to 512 full frame TOF images are delivered in rolling mode. The extremely high sensitivity of the chip allows for a reduced illumination power and reduced overall power consumption compared to other TOF imagers.

epc635 is based on the same technology and instruction set as the existing epc660 QVGA TOF imager from ESPROS.

An evaluation kit for the epc635 is available with hard- and software examples and a comprehensive manual to speed up system integration.

Applications

- People detection and counting
- Mobile postal parcel size measurement
- Machine safety
- Helicopter near terrain flight assistance
- Car collision avoidance systems
- Pedestrian detection and breaking systems
- Man-Machine interface
- Gesture control
- Body size measurement
- General volumetric mapping
- Mobil robotics
- Simultaneous localization and mapping (SLAM)

Block Diagram

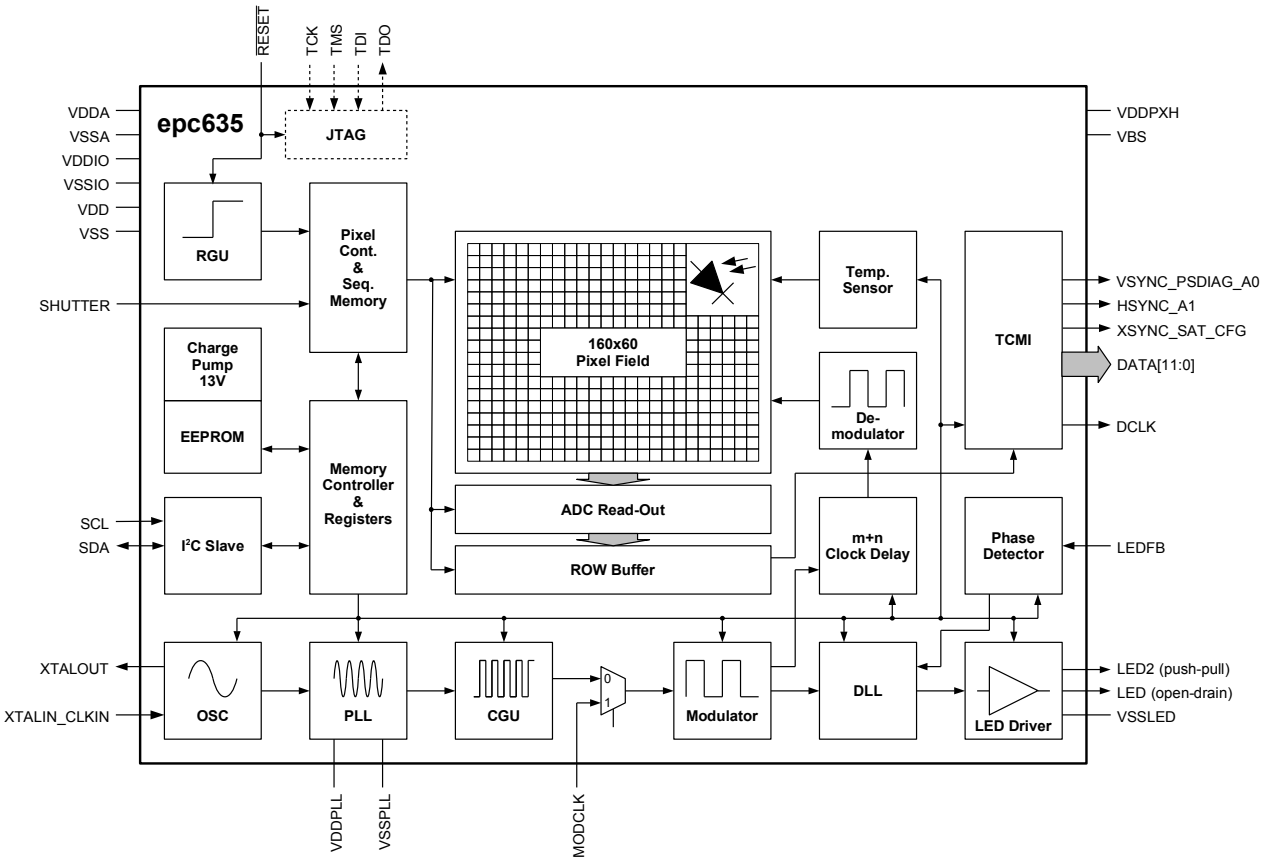


Figure 1: Functional Block Diagram

Neumüller Elektronik GmbH | Gewerbegebiet Ost 7 | 91085 Weisendorf | info@neumueller.com | Tel.: +49 9135 73666-0

Main Features

- **General**
  - 3D TOF imager in full monolithic design
  - 160 x 60 pixel-field, backside illuminated
  - 128 fps full 3D TOF frame rate, in rolling mode up to 512fps
  - Integrated temperature sensor
- **Measurement performance**
  - Absolute accuracy in the centimeter range with appropriate set-up and calibration
- **Integrated LED (or laser diode) driver**
  - LED feedback for drift compensation
  - Laser diode (LD) illumination possible
  - Open-drain LED output pad, up to 500mA drive
  - Push-pull LED2 output pad, up to 50mA drive
- **Parallel digital data interface TCMI**
  - 80MS/s max. data rate, 2.5/3.3V compatible
  - 12/8-bit DATA output + XSYNC/SAT flag on 8bit parallel TCMI interface.
  - VSYNC, HSYNC ("ITU-R656 like" HW synchronization) and DCLK outputs
- **I2C control interface (slave)**
  - 400kHz (FM) / 1MHz (FM+)
- **Integrated EEPROM 128 x 8-bit**
  - Calibration data and user programmable parameters
  - Unique chip ID
- **System / Modulation clock**
  - System clock 4MHz, internal by using crystal/resonator or using external input
  - External LED/LD modulation input MODCLK (optional) up to 80MHz
- **Power supply**
  - Supply voltages +10V, +5V, +2.5/3.3V, +1.8V, -10V
  - Power consumption approx. 210mW (average)
- **Packaging**
  - 6.3x4.2mm cost optimized 44pin CSP (chip scale package),
  - Backside illuminated flip-chip SMD mounting
- **Other data**
  - ROHS compatible

Measurement Modes

- **LED/LD modulation modes**
  - Sinusoidal modulation for single camera applications
  - Selectable modulation frequencies 0.625 ... 20MHz resulting in unambiguity distance of 7.5m … 240m
- **Distance measurement modes**
  - 128 fps 3D TOF with 4x DCS frames, full pixel-field
  - 256 fps 3D TOF with 2x DCS frames, full pixel-field
  - 512 fps 3D TOF with rolling read-out 4x DCS frames, full pixel-field
  - SHUTTER release input for precise start/stop and single/continuous measurement control
- **Non distance measurement modes**
  - Ambient-light measurement (Grayscale image without illumination)
  - Grayscale image with active illumination

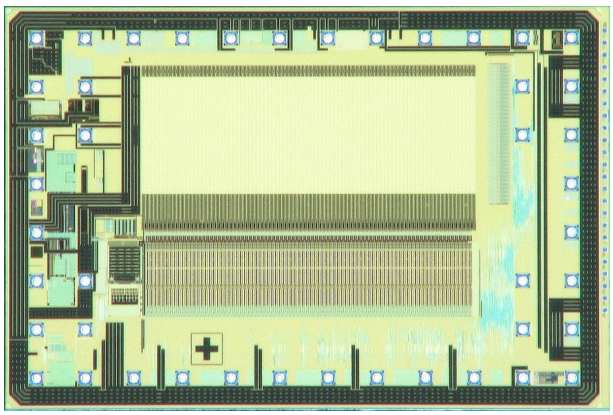


Figure 2: Picture of the epc635

Table of Contents

1. Electrical, optical and timing characteristics

1.1. Operating conditions and electrical characteristics

Typ. operational ratings, T_A = +25ºC, unless otherwise stated

Parameter	Description	Conditions/Comments	V_{SC}	Min.	Typ.	Max.	Units
V_{DD},V_{DDPLL}	Digital supply voltage	Ripple [1] < ± 20 mV	V_{DD}	1.71	1.80	1.98	V
V_{DDIO}	IO supply voltage [3]	Ripple [1] < ± 50 mV	V_{DDIO}	2.25	2.5/3.3	3.63	V
V_{DDA}	Analog supply voltage [2]	Ripple [1] < ± 20 mV	V_{DDA}	4.9	5.0	5.1	V
V_{DDPXH}	Pixel analog 2 supply voltage [2]	Ripple [1] < ± 20 mV	V_{DDPXH}	9.5	10	10.5	V
V_{BS}	Bias supply voltage	Ripple [1] < ±50 mV	V_{BS}	-10.5	-10.0	-9.75	V
I_{VDD}	Digital supply current	@nominal voltage			10	11	mA
I_{VDDPLL}	PLL supply current	@nominal voltage			4		mA
I_{VDDIO}	IO supply current [4]	DCLK = 40MHz			15	25	mA
I_{VDDA}	Analog supply current	@nominal voltage			22	33	mA
I_{VDDPXH}	Pixel analog 2 supply current	@nominal voltage			2	2	mA
I_{VBS}	Bias supply current [8]	Depending on the amount of light on the pixel-field			-2.0 [8]		mA
V_{DDLED}	LED and LEDFB voltage range		V_{DDLED}			V_{DDA}	V
V_{LED_OFF}	Off-voltage at output pin LED					V_{DDLED}	V
V_{LED_ON}	On-voltage at output pin LED (forward voltage)	@ I_{LED} = 200 mA			200		mV
I_{LED}	LED output sink current; on state	modulated peak current continuous DC current				500 250	mA mA
I_{LED_LEAK}	LED output leakage current; off state	V_{LED} max.				10	µA
V_{LEDFB}	LEDFB input voltage range		V_{DDLED}	0		8.5	V
V_{LEDFB_P2P}	LEDFB input voltage peak to peak			0.5		1.8	V
V_{IH_VDDIO}	Digital high level input voltage [5]	excluding XTALIN		0.7 x V_{DDIO}			V
V_{IL_VDDIO}	Digital low level input voltage [5]	excluding XTALIN				0.3 x V_{DDIO}	V
V_{IH_XTALIN}	Digital high level input voltage	XTALIN		1.35			V
V_{IL_XTALIN}	Digital low level input voltage	XTALIN				0.2	V
V_{OH}	Digital high level output voltage [5, 6]			0.8 x V_{DDIO}			V
V_{OL}	Digital low level output voltage [5, 6]					0.2 x V_{DDIO}	V
I_{IH}	Digital high level input current [7]	V_{IH} max.				10 [7]	µA
I_{IL}	Digital low level input current [7]	V_{IL} min.		-10 [7]			µA
I_{OH}	Digital output source current [7]	V_{OH} max.				50	mA
I_{OL}	Digital output sink current [3]	V_{OL} min.		-50			mA
C_{IO}	IO load capacitance [5]					30	pF
f_{IO}	IO switching frequency [5]				20	80	MHz
P_{Pk}	Power dissipation (average)	See Table 27			0.75		W
R_{Th}	Thermal resistance	on PCB with underfill				40	°K/W
T_J	Junction temperature			-40		85	°C

Table 1: Operating conditions and electrical characteristics

Notes:

[1] Min. and Max. voltage values include noise and ripple voltages.

[2] Analog voltage supplies have direct influence on measurement performance. They must be properly decoupled for low noise and ripple.

[3] IO voltage supply must be fixed with respect to external processor's IO supply voltage levels used in the application. It can be set to any value within min and max. operating voltage.

[4] When device is operated at max f_{DCS} frame rate, DCLK at 40MHz, driving loads 15pF each.

[5] I^2C pins SCL and SDA are open-drain outputs and need termination (Pull-up resistor) according to I^2C standards.

[6] $V_{OH/OL}$ and $I_{OH/OL}$ values are measured at max C_{IO} and max f_{IO}.

[7] Value is without termination resistors

[8] A bright illuminated white target right in front of the chip with lens leads to an I_{VBS} of approx. 3.8 mA. At room temperature, I_{VBS} is approx. 3.6 mA without any illumination. I_{VBS} with strong illumination (approx. 55 mW/cm^2, no lens) typ. 17 mA.

1.2. Absolute maximum ratings

Parameter	Conditions
Supply voltage V_{DD}, V_{DDPLL}	-0.5V ... +2.0V
Supply voltage V_{DDIO}, V_{DDA}, V_{DDLED}	-0.5V ... +5.5V
Supply voltage V_{DDPXH}	-0.5V ... +13.5V
Supply voltage V_{BS}	-12.0 ... +0.5V
Voltage to any pin in the same V_{SC} supply class. For Supply Classes refer to Table 1: Operating conditions and electrical characteristics. For Supply Classes vs. pin correspondence refer Table 8: Pin list.	$V_{SC\ min}$ - 0.3V … $V_{SC\ max}$ + 0.3V
ESD rating Note: This is a highly sensitive CMOS mixed signal device. Handling and assembly of this device should only be done at ESD protected workstations.	JEDEC HBM class 1C (1kV to < 2kV)
Storage temperature range (T_S)	-40°C to +85°C
Relative humidity	0 ... 95%, non-condensing

Table 2: Absolute maximum ratings

1.3. Temperature sensor characteristics

Typ. operational ratings, unless otherwise stated

Parameter	Description	Conditions	Min.	Typ.	Max.	Units
T_{TEMP}	Measurement range		-40		+85	ºC
P_{TEMP}	Sensor resolution			14		bit
k	Temperature sensor gain			0.067		ºK/LSB

Table 3: Temperature sensor characteristics

1.4. Timing parameters

Typ. operational ratings, T_A = +25ºC, unless otherwise stated

Parameter	Description	Conditions	Min.	Typ.	Max.	Units
$t_{STARTUP}$	Start-up time	after applying external supplies		340	1'000	µs
$t_{\overline{RESET}}$	$\overline{RESET}$		100			ns
$t_{PLLStrap\_scan}$				4x osc_clk		
t_{PLL}	PLL lock time				30	µs
$t_{EEPROM\_to\_CFG}$	Load CFG registers	Copy EEPROM to CFG registers		340		µs
f_{XTAL}	Clock frequency	of the crystal oscillator (or ceramic resonator)	---	4	---	MHz
df_{XTAL}	Clock frequency deviation	any deviation is added as a linear distance error			±100	ppm
f_{JITTER}	Clock frequency phase jitter	peak-to-peak, cycle to cycle			50	ps
f_{LED}	LED modulation frequency		0.625		40	MHz
f_{MODCLK}	Ext. modulation clock	Refer to chapter 7.10.			80	MHz
$t_{LED\_rise/fall}$	Required rise/fall time of the illumination LED/LD				12	ns
f_{DCLK}	TCMI DCLK	8 bit TCMI data + saturation flag		20	80	MHz
f_{TCMI_data}	TCMI data rate			140	560	Mbit/s
f_{SCL}	I^{2}C data rate				1	Mbit/s

Table 4: Timing parameters

1.5. Optical characteristics

Typ. operational ratings, T_A = +25ºC, unless otherwise stated

Parameter	Description	Conditions/Comments	Min.	Typ.	Max.	Units
A_{PIXEL}	Photosensitive area	1 pixel		20 x 20		µm
A_{SENSOR}	Photosensitive area	160 x 60 pixel		3.2 x 1.2		mm
H_v	Optical sensitivity			143'770		$\frac{LSB}{Lux/sec}$
A_{TOF}	TOF amplitude	dynamic range for distance measurement	25		2'046	LSB
A_{BW}	Grayscale amplitude	dynamic range for grayscale measurement	0		2'047	LSB
t_{INT}	Integration time	see Table 34	0.10		52'600	µs
ΔD_{DRIFT}	Distance drift	without DLL with DLL and fully optimized system design		tbd tbd		cm/K cm/K

Table 5: Optical characteristics

1.6. TOF and grayscale sensitivity

Typ. operational ratings, T_A = +25ºC, modulation frequency 20MHz, integration time 103µs, unless otherwise stated

Optical band	640nm	850nm	940nm
Typ. TOF sensitivity S_{TOF}	0.814 $\frac{nW/mm^2}{LSB}$	0.620 $\frac{nW/mm^2}{LSB}$	0.814 $\frac{nW/mm^2}{LSB}$
Typ. Grayscale sensitivity		0.246 $\frac{nW/mm^2}{LSB}$	

Table 6: TOF and grayscale sensitivity

1.7. Ambient-light suppression

Typ. operational ratings, T_A = +25ºC, modulation frequency 20MHz, integration time 103µs, unless otherwise stated

Optical band	640nm, ± 27.5nm	850nm, ± 32.5nm	940nm, ± 30nm
min. suppression E_{Sup}	0.265 mW/mm²	0.200 mW/mm²	0.265 mW/mm²
Sunlight equivalent	70 kLux	69 kLux	192 kLux

Table 7: Ambient-light suppression

Note: Best performance of ambient-light suppression will be achieved with integration times < 1ms.

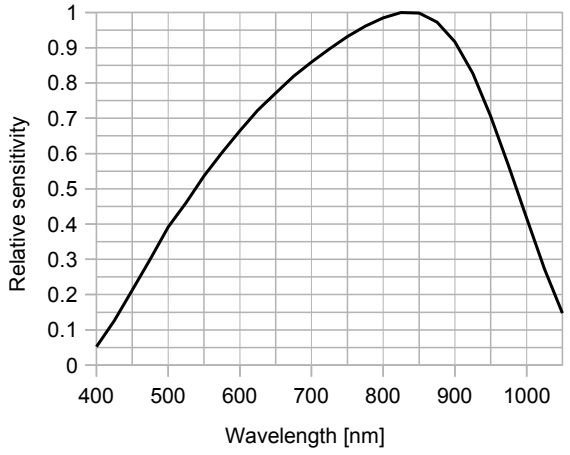


Figure 3: Relative spectral sensitivity (S_λ)
sensitivity S_{AC} vs. wavelength

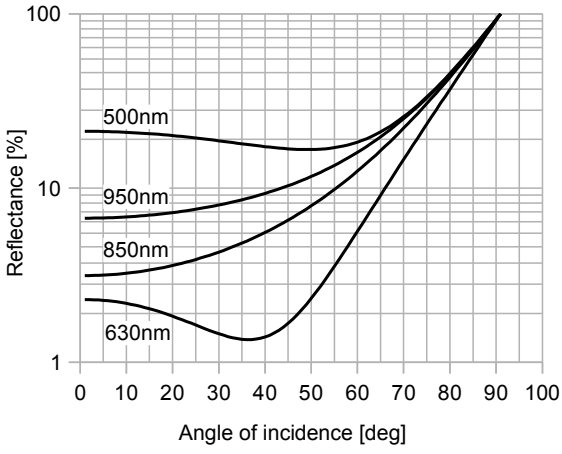


Figure 4: Reflectance vs. illumination angle (AOI)

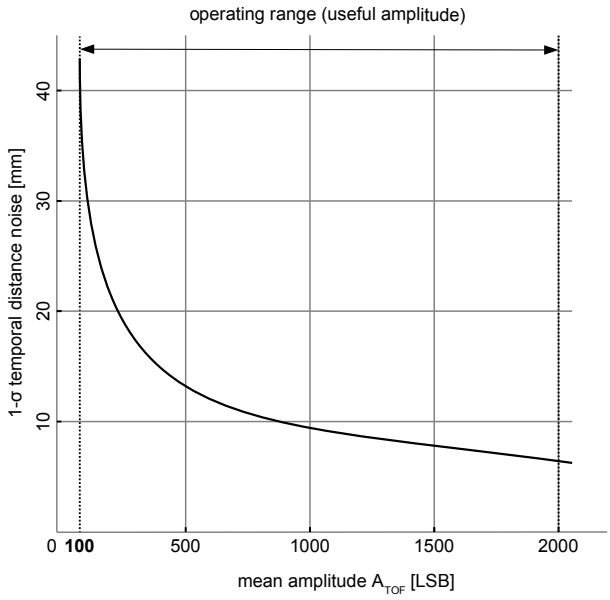


Figure 5: Typ. 1 sigma distance noise
single shot, 4 DCS. no ambient-light,
refer to chapter 6.1.2., section quality

2. Pin-out

2.1. Pin mapping

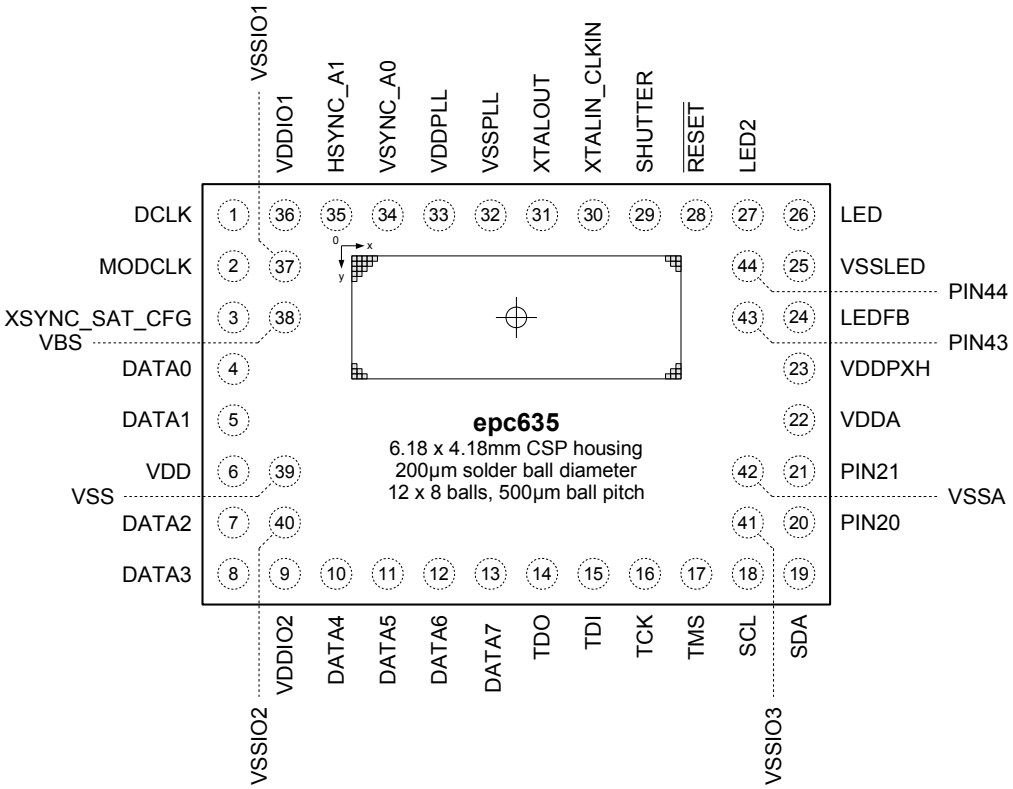


Figure 6: Pin mapping (top-view, solder balls are at the bottom, pixel-field is at the top)

2.2. Pin list

Pin No.	Pin name	Supply class V_{SC}	Pin type	Rst func	Rst level	Description
IO pins						
4	DATA0	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-0 (LSB) / Strap 0: reserved
5	DATA1	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-1 / Strap 1: reserved
7	DATA2	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-2 / Strap 2: reserved
8	DATA3	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-3 / Strap 3: reserved
10	DATA4	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-4 / Strap 4: reserved
11	DATA5	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-5 / Strap 5: reserved
12	DATA6	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-6 / Strap 6: reserved
13	DATA7	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI high-speed output bit-7 / Strap 7: reserved
1	DCLK	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI Data Clock output / Strap 12: reserved
34	VSYNC_A0	V_{DDIO}	DIO	IPD [1]	V_{OH}	TCMI Vsync output / Strap input: I^2C device address bit A0
35	HSYNC_A1	V_{DDIO}	DIO	IPD [1]	V_{OH}	TCMI Hsync output / Strap input: I^2C device address bit A1
3	XSYNC_SAT_CFG	V_{DDIO}	DIO	IPD [1]	V_{OL}	TCMI Xsync output / TCMI Saturation flag output / Strap 15: reserved
18	SCL	V_{DDIO}	DIOD	I	V_{IH}	I^2C clock input [5]
19	SDA	V_{DDIO}	DIOD	I	V_{IH}	I^2C data input/output [5]
29	SHUTTER	V_{DDIO}	DI	PD	V_{IL}	Shutter release input
28	$\overline{\text{RESET}}$	V_{DDIO}	DI	PD	V_{IL}	Reset input (active low), 600kΩ int. pull-down [4]
2	MODCLK	V_{DDIO}	DI	PD		Modulator/demodulator external clock input. If not used, connection to a test pad suggested.
27	LED2	V_{DDIO}	DO			LED driver push-pull output [3]
14	TDO	V_{DDIO}	DO		V_{OL}	JTAG test data output. Do not any electrical connection except to a test pad (suggested).
15	TDI	V_{DDIO}	DI	PU	V_{IH}	JTAG test data input. Do not any electrical connection except to a test pad (suggested).
16	TCK	V_{DDIO}	DI	PD	V_{IL}	JTAG test clock input. Do not any electrical connection except to a test pad (suggested).
17	TMS	V_{DDIO}	DI	PU	V_{IH}	JTAG test mode select input. Do not any electrical connection except to a test pad (suggested).
Digital pins						
30	XTALIN_CLKIN	V_{DDPLL}	AI			OSC input from external XTAL or Resonator / CLKIN from external clock source
31	XTALOUT	V_{DDPLL}	AO			OSC output (buffered) to ext. XTAL or Resonator

Table 8: Pin list

Pin No.	Pin name	Supply class V_{SC}	Pin type	Rst func	Rst level	Description
Analog pins						
26	LED	V_{DDLED}	AOD		V_{LED} max	LED/LD driver open-drain output [3]
24	LEDFB	V_{DDLED}	AI		V_{LEDFB} max	LED/LD driver feedback input [3]
20	PIN20	---	---			Reserved. Do not any electrical connection except to a test pad (suggested).
21	PIN21	---	---			Reserved. Do not any electrical connection except to a test pad (suggested).
43	PIN43	V_{DDPXH}	AI			Reserved. Do not any electrical connection except to a test pad (suggested).
44	PIN44	V_{DDPXH}	AI			Reserved. Do not any electrical connection except to a test pad (suggested).
Supply pins, digital						
36	VDDIO1	V_{DDIO}	PWR			IO supply VDDIO
9	VDDIO2	V_{DDIO}	PWR			IO supply VDDIO
6	VDD	V_{DD}	PWR			Digital supply VDD
33	VDDPLL	V_{DDPLL}	PWR			PLL supply
37	VSSIO1	V_{DDIO}	GND			IO ground VSSIO
40	VSSIO2	V_{DDIO}	GND			IO ground VSSIO
41	VSSIO3	V_{DDIO}	GND			IO ground VSSIO
39	VSS	V_{DD}	GND			Digital ground VSS
32	VSSPLL	V_{DDPLL}	GND			PLL ground
Supply pins, analog						
22	VDDA	V_{DDA}	PWR			Analog supply VDDA
38	VBS	V_{BS}	PWR			Bias supply
23	VDDPXH	V_{DDPXH}	PWR			Pixel analog 2 supply VDDPXH
42	VSSA	V_{DDA}	GND			Analog ground VSSA
25	VSSLED	V_{DDLED}	GND			LED/LD driver ground (return current) [2]

Table 8 cont.: Pin list

Notes:

[1] DIO pin is configured as input with internal pull-down resistor enabled during $\overline{\text{RESET}}$ = 0 (active), only. As soon as $\overline{\text{RESET}}$ is driven to 1, pins stay in that state for another 4x osc_clk periods (see chapter 5.3.3., Strap pins), then the pin is configured as output, finally the internal pull-down resistor is disabled. From this moment on, the pin can be used for normal output function.

[2] VSSLED is the dedicated, isolated GND pin for the LED/LD return-current from external circuitry. It must be connected to PCB GND plane together with the other VSSA GND pins.

[3] LED output can be used to drive an external amplifier with an addition of a pull-up resistor. The voltages at LED output and LEDFB input pins must not exceed values in Table 1: Operating conditions and electrical characteristics.

LED2 output is a push-pull driver for delivering symmetric rise/fall times to the external LED driver circuit. LED2 is internally connected to VDDIO/VSSIO supplies. During integration time, all TCMI pins are silent except for DCLK. As a result, LED2 pin will not pick up switching noise from all other TCMI pins. This can be avoided by using gated DCLK mode. When the TCMI interface is slowed down by setting DCLK <= 10MHz, data-out phase of the last two rows may overlap with the next frame's integration time as another noise source (for very short integration times <100µs). This can be avoided by SEQ_Control.pixel_seq_wait_row_done_en = 1 (default) with a small reduction on the frame rate.

LED and LED2 must not be used simultaneously for driving LED circuits on the PCB. They exhibit different insertion delays and may cause unpredicted distance offset/measurement results.

[4] RESET pin has a 600kΩ (typical) internal pull-down resistor. Therefore, this pin can be safely connected to a standard GPIO of a CPU which is initially high-Z or open-drain during power up sequence. Once the SW takes control, it can program this GPIO as output and drive 1 to release the RESET. The internal pull-down can be override by and external 10kΩ pull-up and a series capacitor to build a simple delayed power-on reset for evaluation/qualification purposes.

[5] I^2C pins SCL, SDA are according to I^2C standards. They are I^2C slave pins which need external pull-up resistors on the PCB. Values of R1 and R2 in the schematics are given only for indicative purposes and must be re-calculated according to the total capacitive load of all I^2C slave/master devices and operating mode (FM or FM+) of the I^2C (chapter 5.4.) in the application.

'Pin type' in Table 8 defines the following:

- DI: Digital Input
- DO: Digital Output
- DIO: Digital Input/Output (bidirectional)
- DIOD: Digital Input/Output (bidirectional), open-Drain
- AI: Analog Input
- AO: Analog Output
- AOD: Analog Output, open-Drain
- PWR: Supply
- GND: Ground

'Rst. Func.' in Table 8 defines the function of IO pins during reset:

- I: Input
- PU: internal Pull-Up
- PD: internal Pull-Down
- IPD: Input with internal Pull-Down

'Rst. Level' in Table 8 defines the level of the IO pins during/after reset (chapter 5.3.)

2.3. Power domain separation and ESD protection

The epc635 chip has internally 7 different power domains and 5 ground references which are interconnected with ESD protection diodes. All pins are also equipped with ESD protection diodes. The diodes have a breakthrough voltage of 0.3V. The designer has to take care that none of these diodes become conductive either at power-up, power-down or normal operation.

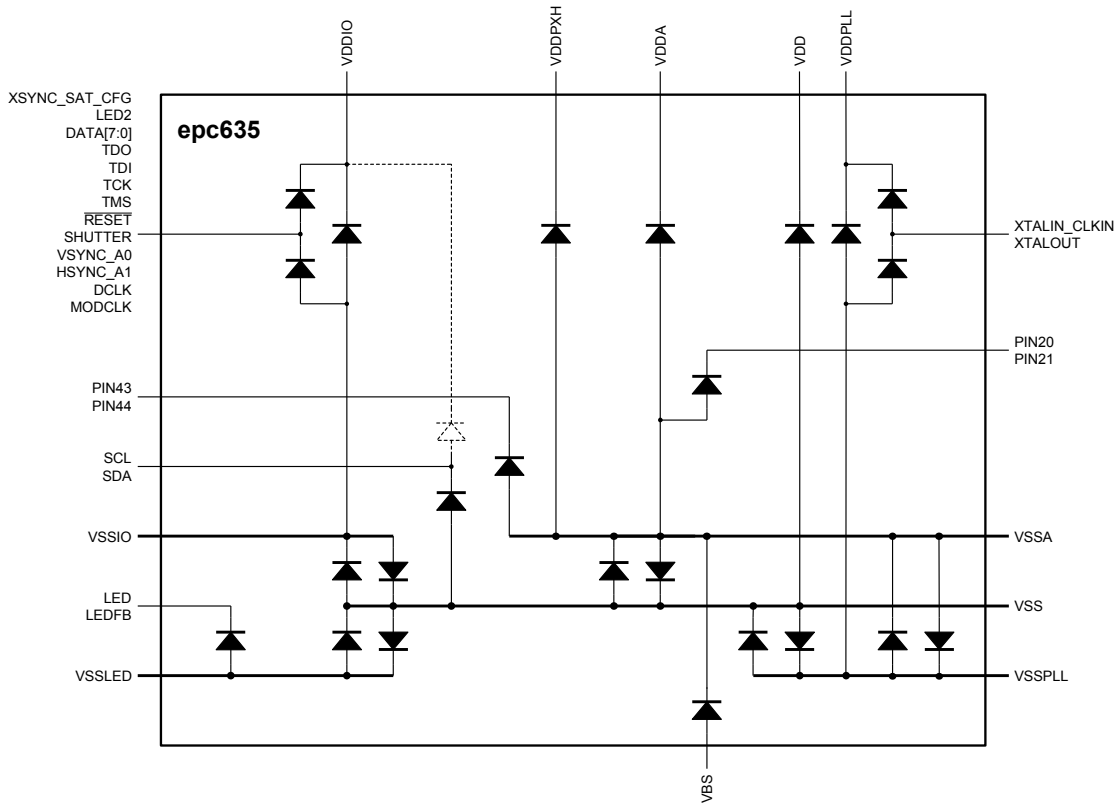


Figure 7: I/O pins and ESD protection diagram

3. Packaging and layout information

3.1. Mechanical dimensions

The center of the effective pixel-field (160x60) is positioned with respect to the center of CSP pin 1 (XSYNC_SAT_CFG) with an offset of 2700µm by 1'000µm (x,y). This point corresponds to intersection of middle of columns C84 - C85 and middle of rows R36 - R37 when mapped to pixel-field coordinate system on the die (see Figure 29).

The packaging technology is chip scale packaging (CSP).

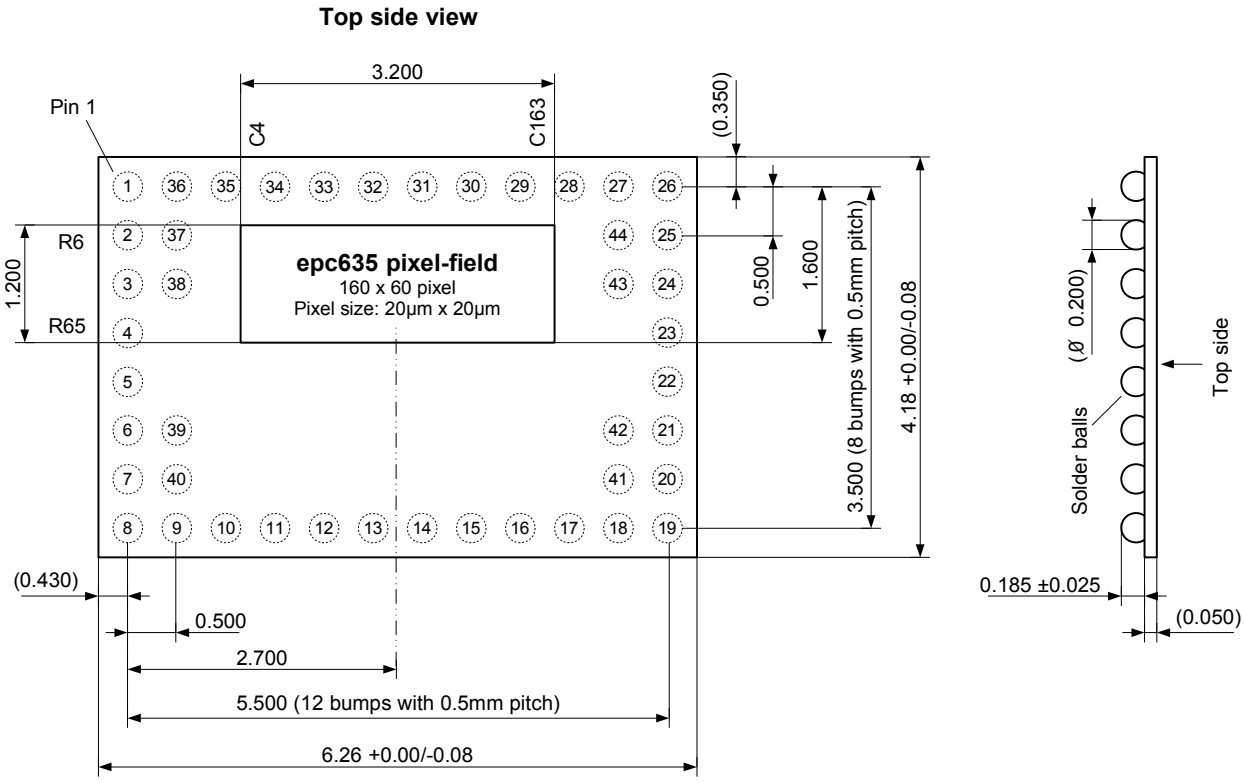


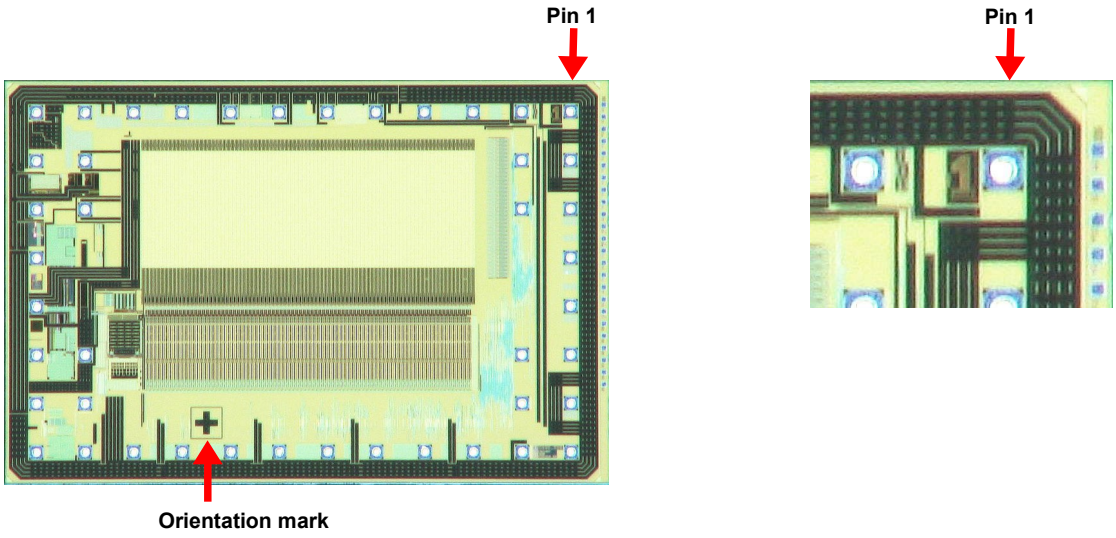
Figure 8: Mechanical dimensions

Notes:

- 
- all measures in mm
- not specified tolerances: ±0.001mm
- Dimensions in brackets: informal only
- Top side is illumination side

3.2. Pin1 marking

The following pictures shows the epc635 chip from the bottom side with view to the solder balls. Please note the location of pin 1.



epc635 chip from the solder ball side

Top right corner from the solder ball side

Figure 9: Pin 1 marking

3.3. Location of the photosensitive area

The photosensitive area is not marked (neither on the front nor on the backside of the IC). As a visible reference, a metal ring of the IC can be used. From the solder ball side it is visible. Also from the front side (photosensitive area) it can be seen with a camera which is sensitive in the near infrared wavelength domain (950 .. 1'150nm).

3.4. PCB design and SMD manufacturing process considerations

As the epc635 chip comes in a very small 44 pin chip scale package, the PCB layout should be made with special care. Because the silicon chip is small and light weight compared the the solder balls, it is highly recommended that all tracks to the chip should come straight from the side. A symmetrical design is highly recommended to achieve high production yield.

The pads and the tracks should also have exactly the same width and shall be covered by a solder resist mask in order to avoid drain of the solder tin alloy to the track.

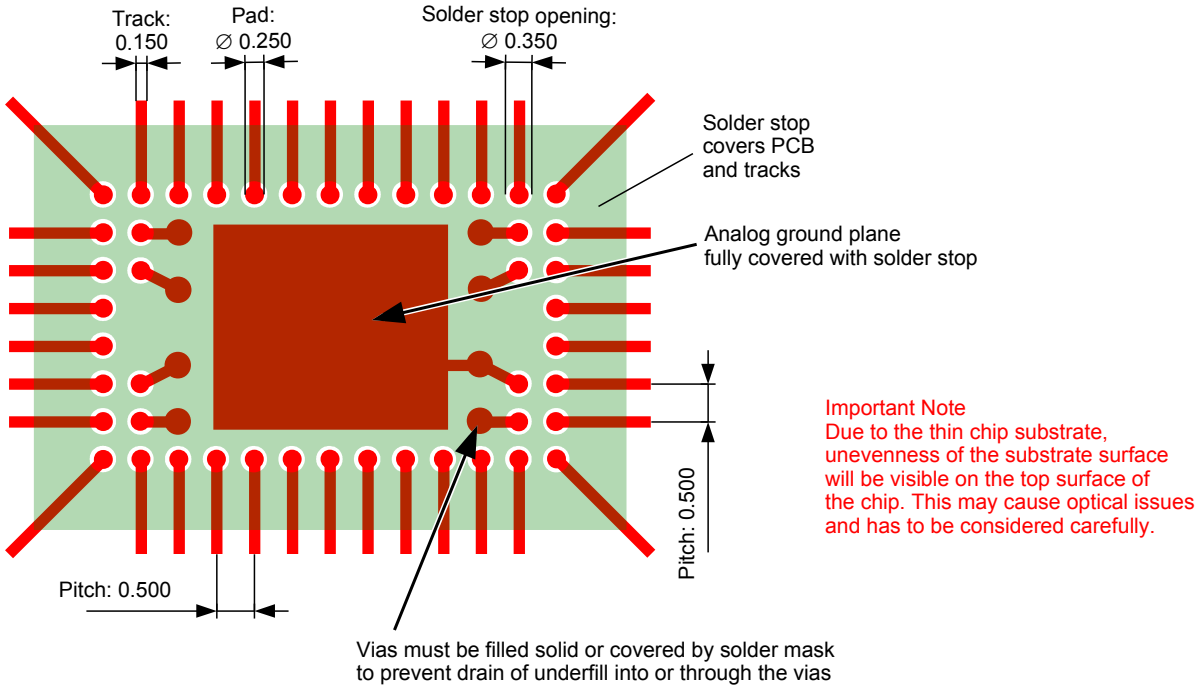


Figure 10: Recommended PCB layout (all measures in mm)

As shown in Figure 10, a ground plane shall be placed on the top PCB layer underneath the chip. This ground plane acts as a shield to suppress high frequency emission of fast interface signal lines. It is important that this plane is completely flat. Thus, the plane must not be scattered nor divided into sections. It should be rather full-faced and no via should be placed in this plane. Otherwise chip bending might occur.

Underfill of the components reduces stress to the solder pads caused by e.g. temperature cycling or mechanical bending.Furthermore the thermal and mechanical fatigue will be reduced and the longterm reliability will be increased. Underfill material and underfill selection is application specific. It shall follow JEDEC-STD JEP150: Stress-Test-Driven Qualification of and Failure Mechanisms Associated with Assembled Solid State Surface- Mount Components.

Please also, refer to the application note AN08 Process-Rules CSP Assembly which can be downloaded from the ESPROS website at www.espros.com/application-notes. Obeying these recommendations a high manufacturing yield can be achieved.

3.5. Soldering and IC handling

Because the epc635 chip is only 50µm thick, a careful handling during the assembly process shall be assured in order to avoid mechanical damage. In addition to that, careful PCB layout is needed in order to achieve reliable assembly results with a high yield. Please refer to the application note AN08 from ESPROS which contains comprehensive information to these topics. This application note can be downloaded at www.espros.com/application-notes.

4. Ordering Information

Part Number	Part Name	Package	RoHS compliance
P100 181	epc635-CSP44	CSP44	Yes
P100 404	epc635 Card Edge Connector Carrier	PCB 37.25 x 36.00 mm	Yes

Table 9: Ordering Information

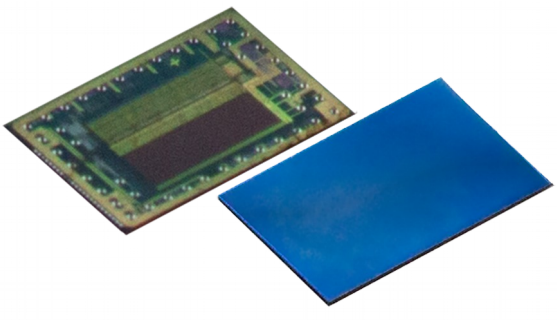


Figure 11: epc635-CSP68
(bottom and top side)

Figure 12: epc635 Card Edge Connector Carrier

5. Functional Description

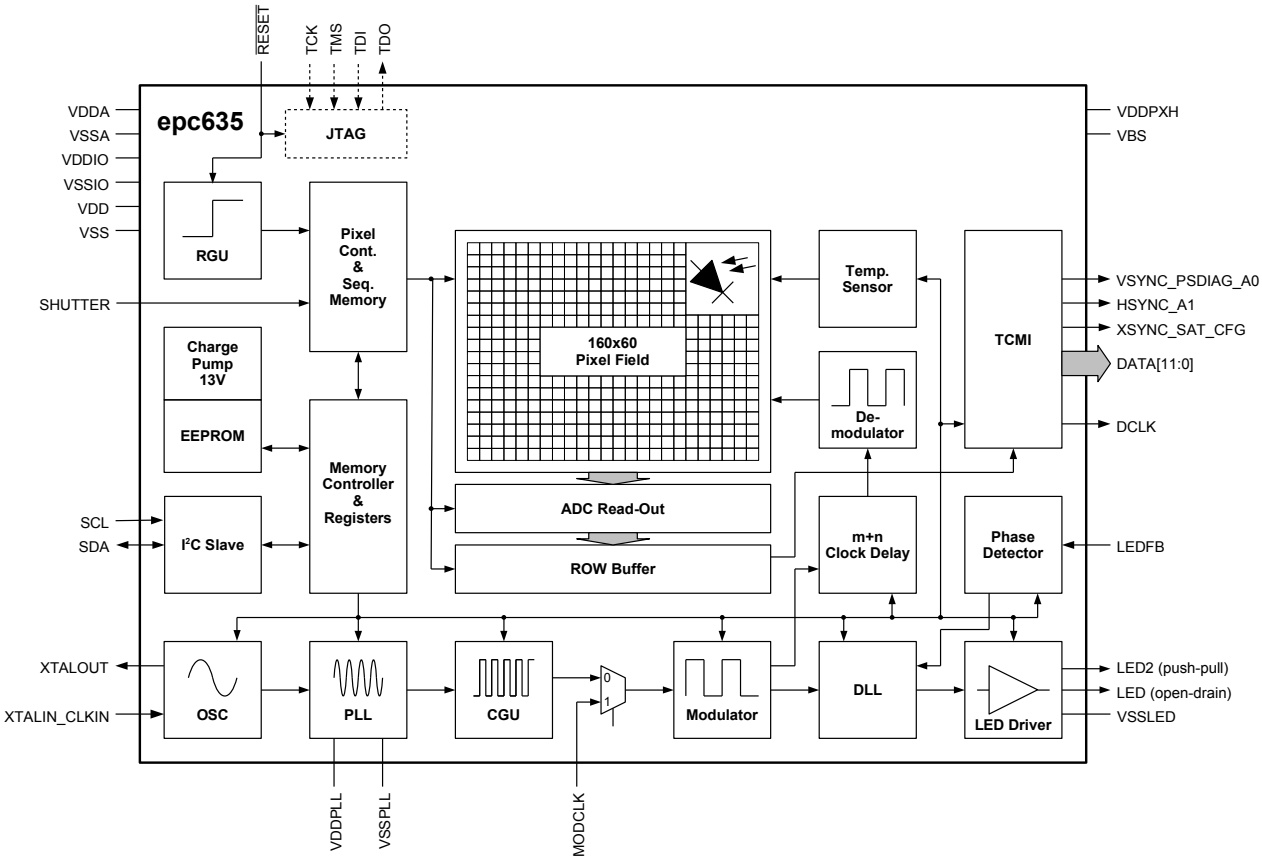


Figure 13: Block diagram

Figure 13 shows the relations between the functional blocks with signal flow diagram.

Based on the clock and mode setup of the chip, the modulation signal is generated and output over the LED driver to the external IR LED . Current flowing through the LED output pin can be monitored via LEDFB pin. This information is used for compensating delays generated due to temperature changes and aging of the LED in the system. Instead of an LED, also laserdiodes (LD) can be used. An external illumination driver can be driven by the digital signal of the LED2 output.

The compensation is used for delaying the demodulation signal going into the pixel-field.

In the pixel-field, the reflecting IR light returning from the object is captured and converted to electrons which are transferred into two storage gates within each pixel (MGA and MGB) depending on the phase information of the demodulation signal.

The parallel ADC blocks on top and bottom pixel-field efficiently convert and transfer the phase information to the digital domain where it will be formatted and finally output via the TCMI for external distance calculation.

5.1. Typical Application Diagram

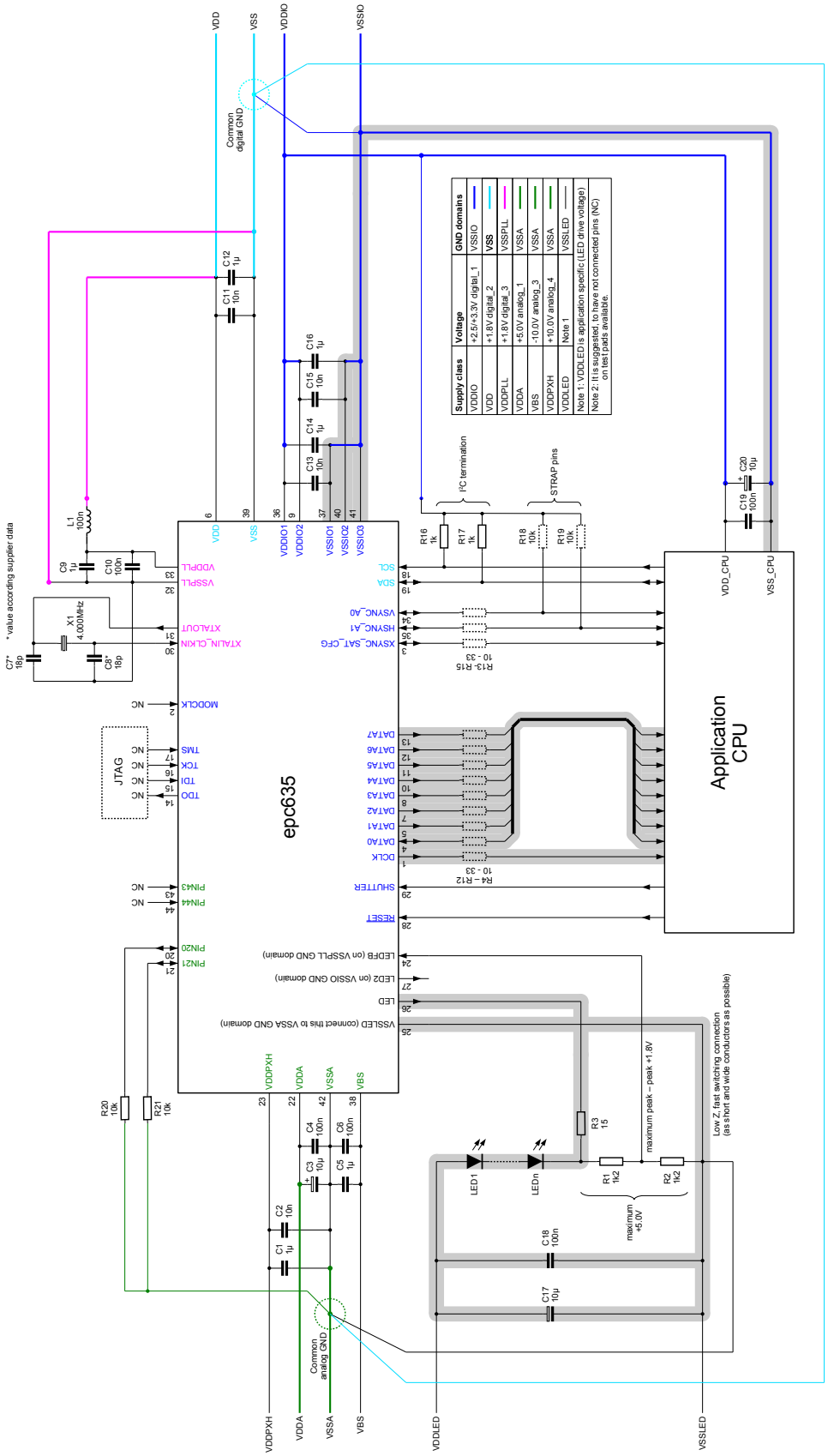


Figure 14: Typical application diagram

epc635 is supplied with four positive (+1.8V, +2.5/3.3V, +5V, +10V) and one negative (VBS) external DC supply voltage levels (Figure 14).

- Decoupling capacitors must be placed next to each supply pin pair in order to minimise the instantaneous voltage drop on internal supply rails due to fast switching high-speed signals (Table 10). They help reducing switching noise on internal/external supply rails. Therefore, they minimize the impact on system's optical/electrical performance.

+1.8V is used for supplying the digital logic (VDD), the on-chip oscillator OSC and the phase-look-loop PLL (VDDPLL).

- The digital logic creates some internal switching noise on VDD.
- When same supply is shared together with OSC and PLL, their supply wiring must be separated from the digital wires and physically isolated from each other. These supplies are marked in the application diagram as VDD and VDDPLL, respectively (Figure 14). A good practice is inserting on the PCB a series inductance of 100nH between them close to the supply source, then creating separate supply islands for both on the board. The XTAL/OSC and PLL are critical parts of the chip which directly impacts the optical system performance (i.e. distance calculation). Therefore, if the +1.8V supply is too noisy, it is recommended for the application to have a dedicated silent +1.8V supply only for OSC and PLL.

+2.5/3.3V is used for supplying the high-speed IO pins (TCMI and LED2) and the slow I^2C pins (VDDIO).

- High speed IO pins toggle at 10/20/40/80MHz during data transfer, hence generating continuously switching noise (much more dominant than the digital noise). Therefore VDDIO supply wires and layers must be carefully designed and isolated in a separate supply island on the PCB.
- I^2C pins external pull-up resistors can share the same supply on the PCB.
- As the name implies, the IO supply voltage can be one of the two +2.5V, +3.3V or in between. The specific IO voltage depends on the choice of the application CPU's IO voltage and other system supply requirements. It is not recommend to change this voltage on the fly when the TCMI, LED2 or I^2C interfaces are running. When the application needs power saving during system idle periods, it can be scaled from +3.3V down to +2.5V only after frame acquisition is stopped and both interfaces are completely inactivated. It can be increased back to +3.3V before re-activating the chip for frame acquisition, accessing I^2C, LED2 or TCMI interface. Note that, voltage scaling must be done in a controlled way having both application CPU's and epc635's IO voltages at the same time at the same level.
- Some CPU's can also work at +2.5V IO voltage but with reduced IO toggle speed. In such situations, application must make sure that the right TCMI DCLK frequency is set, both on the epc635 and on the CPU itself.

+5V is used for supplying analog blocks of the chip e.g. pixel-field drivers and ADC readout circuitry. Refer to Figure 14).

- It needs a dedicated supply to keep it biased at this voltage.

+10V is used for supplying pixel-field circuitry (VDDPXH).

- It needs a dedicated supply to keep it biased at this voltage.

VBS is used for biasing the the pixel-field.

- The use of a stable supply source with a low ripple is recommended. There is no switching or active internal circuit dependent current consumption. But, an ambient-light dependent leakage current is generated on this pin and therefore it needs a dedicated supply to keep it biased at this voltage (refer to Table 1, note 8).

A 4MHz quartz crystal or a ceramic resonator together with two load capacitors must be connected to XTALIN_CLKIN and XTALOUT pins in order to use internal oscillator OSC as time base for the epc635. The load capacitor value has to be selected according the crystal or resonator supplier's recommendation. The frequency accuracy and stability are directly related to the distance readings. Alternatively an external clock source can be connected to XTALIN_CLKIN pin, leaving XTALOUT unconnected (chapter 5.2.).

The optional MODCLK input can be used for user controlled/modulated clock. It is used for both the LED driver and the pixel-field demodulator.

SCL, SDA are I^2C slave pins which need external pull-up resistors on the PCB (see also VDDIO supply). Values of R16 and R17 are given only for indicative purposes and must be re-calculated according to the total capacitive load of all I^2C slave/master devices and the operating mode FM or FM+ of the I^2C (chapter 5.4.) in the application.

VSYNC_A0, HSYNC_A1, XSYNC_SAT_CFG, DATA[11:0], DCLK, high-speed TCMI signals (chapter 5.9.), SHUTTER and $\overline{\text{RESET}}$ control signals toggle in the VDDIO range.

- To minimize the skew, the high-speed *SYNC, DATA[11:0], DCLK signals wires must be routed equal in impedance and length less than 10cm long with less than 10mm difference on the PCB. As they are toggling all the time, they can be separated with ground wires on the side adjacent to other signals/supply lines, routed with enough distance from other sensitive signal wires on the board. Series termination resistors R4 … R15 (10 … 33Ω) are needed at high-speed outputs to control the slew.
- They must be directly connected to the application CPU. In some cases $\overline{\text{RESET}}$ can also be driven from power management device.
- There are optional pull-up resistors R19 and R19 (10kΩ) to set initial values of some configuration registers during start up of the chip. Such outputs pins are called strap pins. They are scanned one time immediately after $\overline{\text{RESET}}$ is released (chapter 5.3.3.). Two LSB of the I^2C device address bits are programmable via these strap pins during start-up (chapter 5.3.). When there is only one slave and one master on the I^2C bus as in the application diagram (Figure 14), these resistors can be omitted.

The LED pin is the internal open-drain IR LED/LD driver output. When the driver is active (on), the IR LED/LD on-current flows through the power resistor R3 into the LED pin, through the driver and comes out of the chip on the VSSLED ground pin.

- The LED pin toggles up to 20MHz or according to the MODCLK clock with a current maximum of 500mA limited by the resistor R3.
- The number of IR LEDs depends on the level of the LED supply voltage and the turned-on forward voltage drop of the IR LEDs.
- This signal creates a lot of ground noise. Therefore, VSSLED pin is decoupled from the other analog grounds internally. It must be shorted with the other analog ground pins with a low-ohmic connection as short as possible on the PCB. In this way, there will be minimal voltage differences in the ground planes of the board.
- The LED supply line must be isolated properly from any analog supply on the PCB to minimize noise coupling from the LED drivers.

The LEDFB pin is the amplifier input for the LED feedback signal. The easiest way is to pick it from a voltage divider R1 and R2 following the IR LEDs on the PCB.

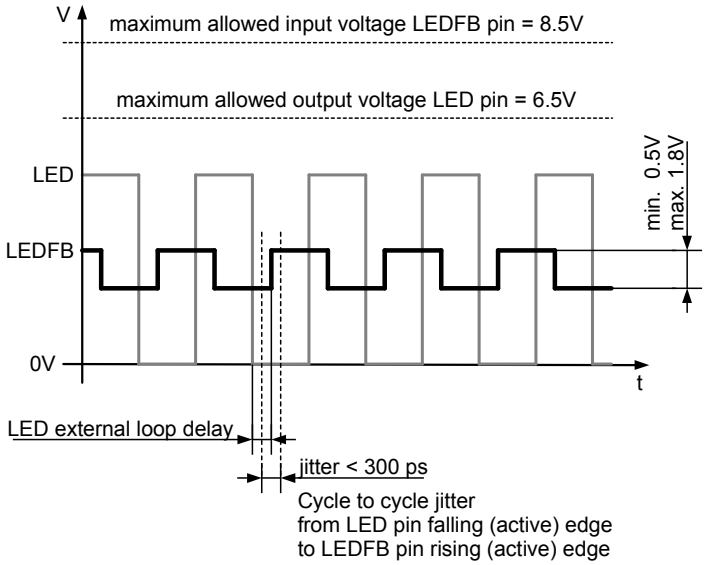


Figure 15: LED and LEDFB pin signal restrictions

The LED2 pin is the alternative push-pull driver providing symmetric rise/fall times to drive external LED driver.

- It works from the +2.5/+3.3 VDDIO supply (VSSIO GND domain) and swings in the same voltage range like the TCMI pins.
- LED2 = LOW (approx. 0v) corresponds to LED = OFF (max. output voltage).
- Because non of the TCMI pads toggle during integration time, LED2 pin is the only toggling during integration time and it is not affected from switching noise of others.
- It can toggle up to 20MHz and deliver up to +/-50mA peak.
- LED and LED2 pins must not be used at the same time for driving the external illumination. They exhibit different phase delays and this can result wrong distance measurements.

LEDFB pin is the feedback amplifier input.

- Refer to Figure 15 for input signal integrity and restrictions at the LEDFB pin.

TCK, TMS, TDI, TDO are standard JTAG controller pins (Additional information upon request)

It is recommended having “not connected pins” (NC) on test pads available. It helps e.g. to check after assembly for correct orientation of the chip or for short-cuts.

To guarantee the proper function of the chip, the remaining, not here listed pins have to be connected according Figure 14.

Part No.	Description	Pin No.	Value			Toler-ance	Supply class V_{SC}	Comments
			Min.	Typ.	Max.			
C3	VDDA	22 - 42	10 µF			±20%	V_{DDA}	Low ESR
C1	VDDPXH	23 - 42	1 µF			±20%	V_{DDPXH}	Ceramic X7R
C5	VBS	38 - 42	1 µF			±20%	V_{BS}	Ceramic X7R
C9	VDDPLL	33 - 32	1 µF			±20%	V_{DDPLL}	Ceramic X7R
C12	VDD	6 - 39	1 µF			±20%	V_{DD}	Ceramic X7R
C14, C16	VDDIO	36 – 37, 9 - 40	1 µF			±20%	V_{DDIO}	Ceramic X7R
C4	VDDA	22 - 42		100 nF		±20%	V_{DDA}	Ceramic X7R
C6	VBS	38 - 42		100 nF		±20%	V_{BS}	Ceramic X7R
C22	VDDPLL	61 - 62		100 nF		±20%	V_{DDPLL}	Ceramic X7R
C2	VDDPXH	22 - 42		10 nF		±20%	V_{DDPXH}	Ceramic X7R
C11	VDD	6 - 39		10 nF		±20%	V_{DD}	Ceramic X7R
C13, C15	VDDIO	36 – 37, 9 - 40		10 nF		±20%	V_{DDIO}	Ceramic X7R
C7, C8	XTAL	30 – 32, 31 - 32	---	18 pF [2]	---	±20%	V_{DDPLL}	Ceramic NPO
L1	VDDPLL	---		100 nH		±20%	V_{DDPLL}	Inductor
X1	XTAL	30 - 31	---	4 MHz	---	±100ppm	V_{DDPLL}	Quartz / Resonator
R4 - R15	Bus termination		0 Ohm	10 Ohm	33 Ohm	±20%	V_{DDIO}	Resistors
R16, R17	I^2C pull-up			1 kOhm		±20%	V_{DDIO}	Resistors
R18, R19	I^2C address			10 kOhm		±20%	V_{DDIO}	Resistors

Table 10: Values of component related to epc635 chip (Figure 14)

Notes:

[1] All other components are application specific.

[2] The capacitor value has to be selected according the crystal or resonator supplier's recommendation.

5.2. External Clock

epc635 can be driven by an external 4MHz clock source connected to XTALIN_CLKIN input pin. XTALOUT output pin left unconnected. Input clock signal levels must match V_{DDPLL}/V_{SSPLL} supply levels (Table 1).

When the external clock source is coming from +2.5/3.3V voltage domain, a simple resistor divider circuit can be deployed to adjust the voltage levels. The following resistor values R26/R27 are recommended: 1kΩ/2.2kΩ with +2.5V voltage domain; 1kΩ/1.2kΩ with +3.3V voltage domain (see Figure 16).

XTALOUT is not designed to drive external loads. Therefore, it is not recommended to use XTALOUT as a “buffered clock source” for another device in the application circuit.

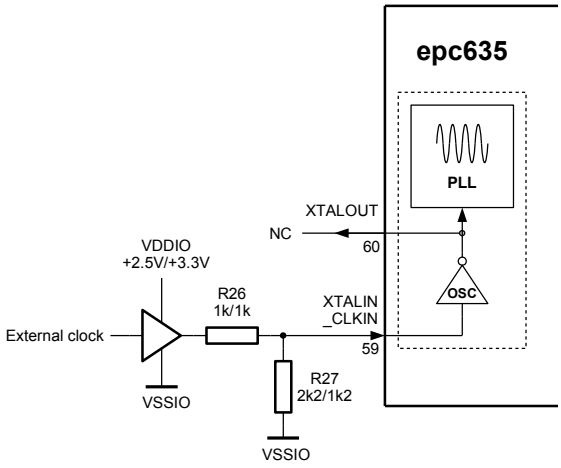


Figure 16: Resistor divider to adjust external clock voltage levels to XTALIN_CLKIN

IMPORTANT: The optical performance of the chip directly depends on the input clock precision/stability; therefore, only a clean, stable clock must be used as external clock source.

5.3. Supply, Reset and Start-up Options

5.3.1. Supply voltages and external reset

During the power-up sequence, VDD and VDDPLL supplies (Figure 17) must be applied at the same time to the epc635. VDDIO can be applied either at the same time or after VDD and VDDPLL supplies become stable. In a system where VDDIO voltage is connected in parallel to application CPU IO supply pins (see Figure 14), VDD and VDDPLL can be generated by a linear regulator directly from VDDIO supply. In this case, all these three supplies ramp together.

VDDA and VDDPXH supplies must be applied as a second group, after all VDD, VDDPLL and VDDIO supplies become stable.

The negative supply VBS must be applied after all positive supplies reached their rated levels.

The $\overline{\text{RESET}}$ input must be kept active (low) while all positive voltages are ramping-up in order to guarantee proper reset of all internal circuits. As soon as rated positive levels are reached, $\overline{\text{RESET}}$ can be released (high). On the other hand, there is no strict relation between VBS level and $\overline{\text{RESET}}$ timing. Application must guarantee that the VBS supply is turned on and reached its negative rated level before the chip is triggered via the SHUTTER for frame acquisition. In case of an external clock is applied at XTALIN_CLKIN instead of a crystal/resonator is used with on-chip OSC, clock must be present before $\overline{\text{RESET}}$ is released.

IMPORTANT:

- It is possible to shutdown entire supplies for a very low standby current. In that case, first $\overline{\text{RESET}}$ must be driven low, then supplies must be turned off in the reverse order. Refer for details to chapter 7.7., Power saving options.
- VDDA and VDDPXH supplies must be never kept on while turning off VDD, VDDPLL and VDDIO. Such condition may create a permanent damage to the chip.

For rated values of supply voltages, refer to Table 1: Operating conditions and electrical characteristics.

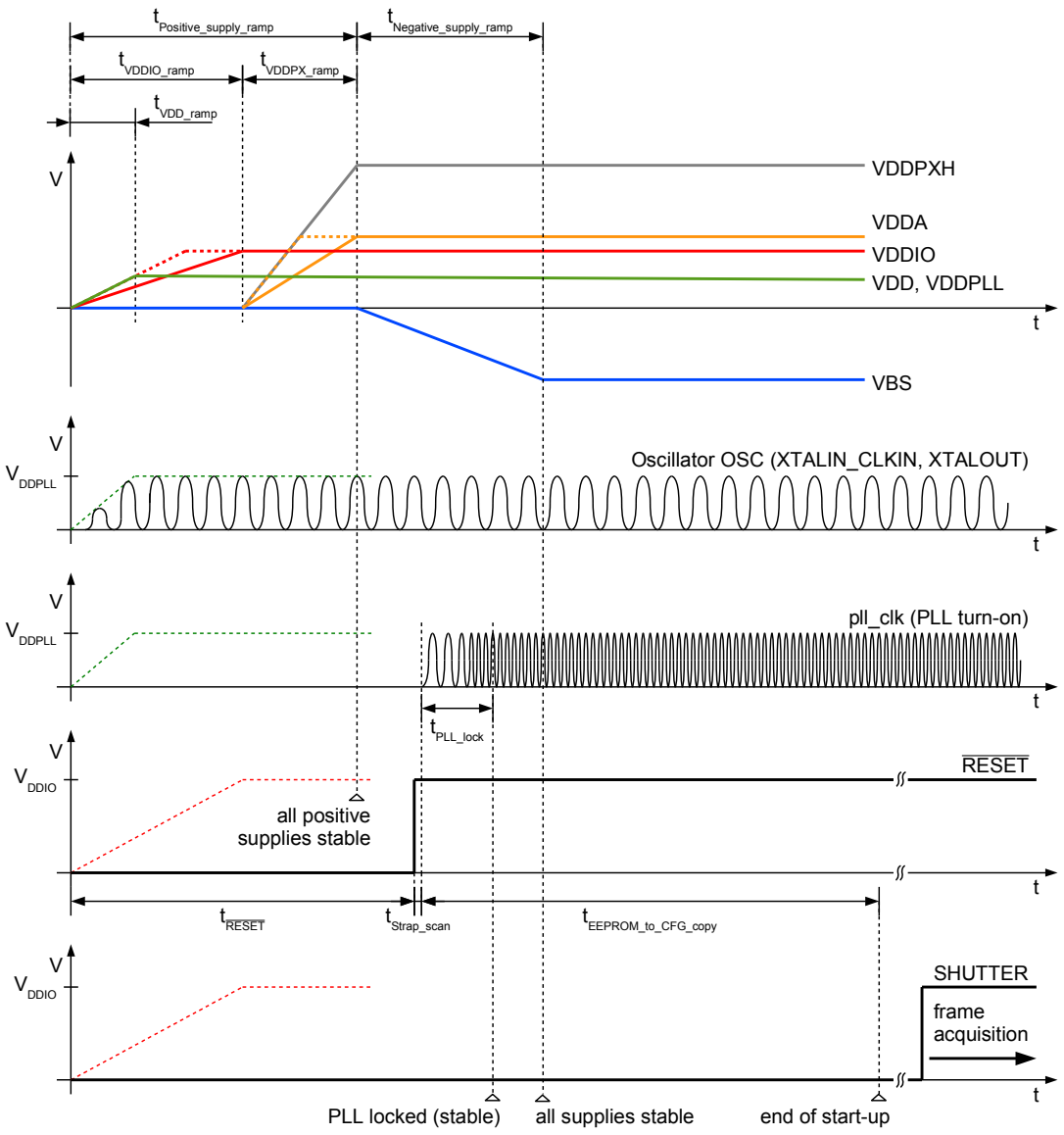


Figure 17: Start-up and reset sequence during power-up (normal mode)

5.3.2. Software reset via I²C

The epc635 can be software reset by sending I²C general call reset command (chapter 5.4.4.). Reset behaviour is similar to the RESET pin. The chip is resetting including all analog and digital blocks at the same time.

5.3.3. Strap pins

The epc635 has output pins with dual/alternative functionality for PCB level flexible start-up configuration changing, called 'strap pins'. $\overline{\text{RESET}}$ release is followed by a very short strap pin scanning step. The chip programs all of its strap pins as inputs with internal weak pull-downs enabled for 4 osc_clk periods. If there is no external pull-up resistor connected, the corresponding strap pin will be scanned as logic 0 due to internal pull-down resistor. If there is an external pull-up resistor connected (Figure 14), it will override the internal pull-down and corresponding pin will be scanned as logic 1. After the strap scan period, pins are programmed back as outputs so that they can be used for their main function. Strap pins and their definitions are listed below (Table 11).

Pin	Strap pin	Definition
XSYNC_SAT_CFG	3	reserved
HSYNC_A1	35	Set A1 bit of 7-bit I²C slave device address (section 5.4.1.).
VSYNC_A0	34	Set A0 bit of 7-bit I²C slave device address (section 5.4.1.).
DCLK, DATA7 … DATA0	1, 13 … 10, 8, 7, 5, 4	reserved

Table 11: Strap pin definition

IMPORTANT:
For having the strap pin function correctly, there should be only one 10kΩ pull-up resistor per pin be active on the line during strap-scan phase. No other pull-down/pull-up resistor, neither on the board nor in the pins of the application CPU shall be added or enabled, respectively. Destination pins in the receiving device (application CPU) must be configured always as inputs during the start-up and never programmed as outputs at any time. If the receiving device by default enables its internal pull-down/up resistors during start-up phase, the application SW must disable them before the reset is released and until the end of the strap-scan phase.

5.3.4. Start-up (Clock, PLL turn-on and EEPROM copy)

The epc635 starts using either the internal 4MHz oscillator OSC with a crystal/resonator (Figure 14) or an external 4MHz clock, followed by an EEPROM copy sequence in parallel to the PLL turn-on phase. This is the factory default configuration.

Several configuration registers are modified by copying the EEPROM content (Figure 68 and Table 41, i.e. overwrite reset values). The EEPROM copy step takes 340µs after the $\overline{\text{RESET}}$ is released. Note, the PLL lock time (stable < 32µs) is relatively faster compared to the slower EEPROM copy.

5.4. I²C Slave

The I²C-bus interface allows accessing the RW registers and the programming of the EEPROM registers which store the configuration parameters. It is the only interface through which the configuration registers can be accessed (Figure 68, Table 40 and Table 41) by the application. It works as a slave device according to the I²C specification (refer to chapter 10.2.) with a transfer rate of up to 400 kbit/s in Fast Mode (FM) or 1Mbit/s in Fast Mode plus (FM+). The I²C master such as an external CPU can set the transfer speed simply by driving the SCL input at that frequency (up to 1MHz), therefore there is no prior register configuration or setting necessary.

I²C specification is supported in epc635 with following remarks/exceptions:

- Only 7-bit addressing is supported.
- Clock stretching is supported.
- General call address: By transmitting 0x00 followed by 0x06 (issues software reset) or transmitting 0x00 followed by 0x04 (device address reload), the programmable part (A0, A1) of the I²C address pins is overwritten by the initially scanned value through strap pins during start-up or reset phase.
- Software reset is supported.
- Other uses of I²C bus are not supported.

5.4.1. Device addressing

The epc635 7-bit I²C device address is hard-wired to the value shown below in Figure 18. Two address bits A0, A1 can be optionally initialized as 1 through strap pins (chapter 5.3.3.). In a typical single-camera 3D TOF imager application in which epc635 is directly connected as a single I²C slave to a single I²C master, the strap pins can be used without any external pull-up resistors. In this case, the device address is set after reset default as 0100000. In a multi-camera application with up to 4 epc635 devices connected on the same I²C bus as slaves or together with other I²C slaves talking to a single I²C master, external pull-up resistors can be utilized on the strap pins to initialize different I²C device addresses in order to correctly identify different epc635 slaves on the bus.

MSB							LSB
0	1	0	0	0	A1	A0	**R/$\overline{W}$**

Figure 18: Device address through I²C

5.4.2. I²C bus protocol notation

The following notation is used:

- S **S**TART condition
- P STO**P** condition
- A **A**cknowledge last byte (ACK)
- $\overline{A}$ **N**ot-Acknowledge last byte (NACK)
- Shaded part of protocol: transmitted by master
- Unshaded part of protocol: transmitted by slave

5.4.3. I²C bus timing

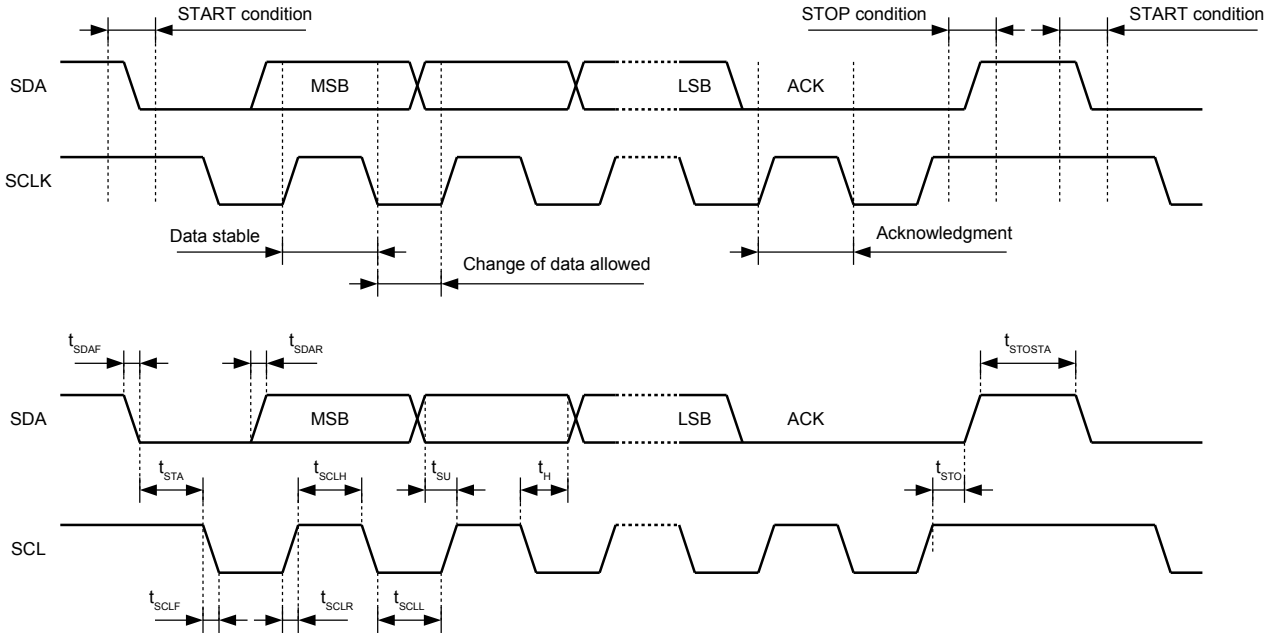


Figure 19: I²C bus timing
top: Basic communication sequence
below: Timing parameters

Symbol	Parameter	Min.	Max.	Units
f_{SCL}	I²C data rate		1	Mbit/s
t_{SCLL}	SCL clock low time	0.5		µs
t_{SCLH}	SCL clock high time	0.26		µs
t_{SU}	SDA setup time	50		ns
t_H	SDA hold time		0	ns
t_{SDAR} t_{SCLR}	SDA and SCL rise time		120	ns
t_{SDAF} t_{SCLF}	SDA and SCL fall time		120	ns
t_{STA}	Start condition hold time	0.26		µs
t_{STO}	Stop condition setup time	0.26		µs
t_{STOSTA}	Stop to start condition time (bus free)	0.5		µs
C_b	Capacitive load for each bus line		550	pF
t_{SP}	Pulse width of the spikes that are suppressed by the analog filter		50	ns

Table 12: I²C bus timing: Timing parameters (FM+)

5.4.4. General calls

epc635 supports two general call commands:

■ Software reset through I²C
(0x00, 0x06) issues a software reset, same behavior like with $\overline{\text{RESET}}$ pin.

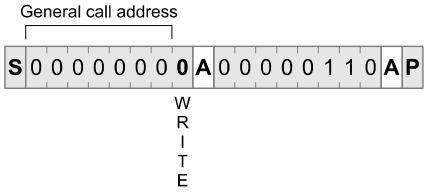


Figure 20: Software reset through I²C

■ Device address reload
(0x00, 0x04) activates the I²C address stored in register 0xCA. Note that the the values of A0 and A1 cannot be changed by software. Therefore, this general call command only works for bits 2 to 6 of register 0xCA (chapter 5.3.3).

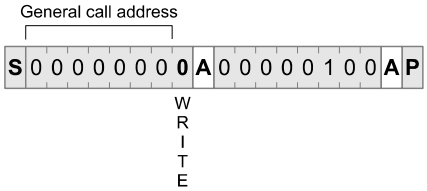


Figure 21: Device address A1, A0 reload through I²C

5.4.5. Write access

The epc635 I²C interface offers single-byte write access and multi-byte write access, where the former is basically a subset of the latter.

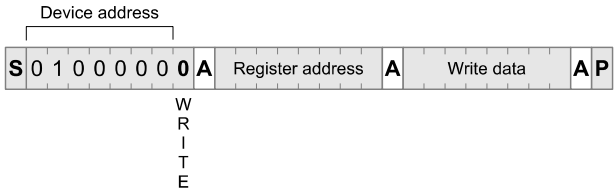


Figure 22: Single-byte Write access through I²C

During a single-byte write, only one register is written. After the device address is transmitted, the master has to transmit the register address and the write data in two I²C data packets (Figure 22). The access is terminated by a STOP condition.

During a multi-byte write operation, the master transmits the device address and the address of the first register to be written. All subsequent bytes until the STOP condition are interpreted as write data packets by the epc635 (Figure 23). The write address pointer is incremented internally. It is illegal to transmit so many data packets that the write address pointer reaches the limit of reserved address space (see chapter 8.4., Table 40, Table 41).

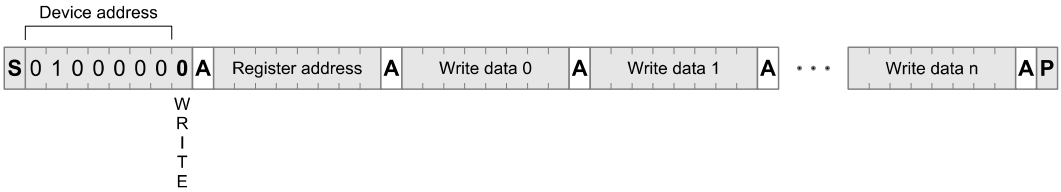


Figure 23: Multi-byte Write access through I²C

5.4.6. Read access

The epc635 I²C interface offers single-byte read access and multi-byte read access, where the former is basically a subset of the latter.

During a single-byte read, only one register is read. After the device address is transmitted, the master transmits the register address. After addressing the epc635 with a read-command, epc635 answers with the read data (Figure 24). The access is terminated by a not-acknowledge (NACK) and a STOP condition from the master.

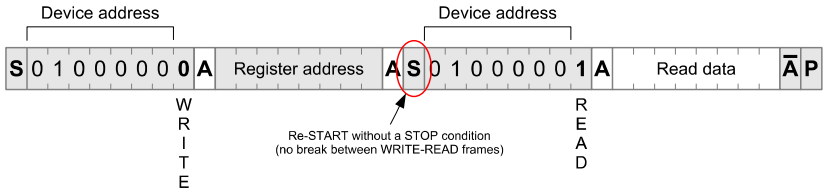


Figure 24: Single-byte Read access through I²C

During a multi-byte read operation the master transmits the device address and the address of the first register to be read. After the epc635 is addressed with a read command, epc635 answers with read data bytes until the master does not acknowledge a byte. The master is expected to terminate the access with a STOP condition thereafter (Figure 25). During the access the read address pointer is incremented epc635 internally. It is illegal to read so many data bytes that the read address pointer reaches the limit of reserved address space (see chapter 8.4., Table 40, Table 41).

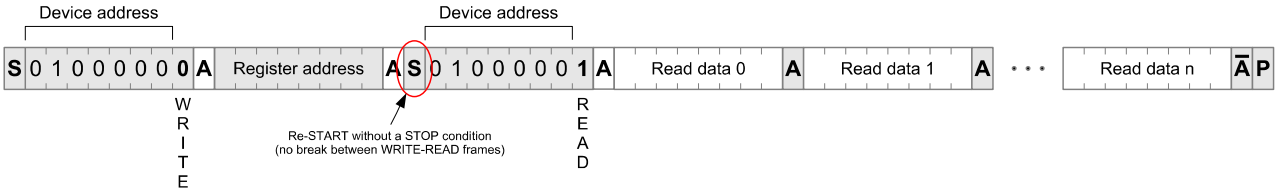


Figure 25: Multi-byte Read access through I²C

5.4.7. Control commands

The operating modes of the epc635 are initialized, activated, deactivated and monitored by sending several single or multi-byte write and read command sequences through I²C interface. This section lists and explains all available commands together with their access times (f_{SCL} = 1MHz → t_{SCL} = 1µs).

Listed in Table 13 is a summary of main control command sequences to operate the chip including the total number of command bytes transferred and their duration. There is no particular order defined for sending the commands. The only requirement is having no on-going frame acquisition process in the pipe when updating non-shadowed registers. The shadowed registers (marked with ** in the register map e.g.INTM_hi/lo, Int_len_hi/lo, etc,. see section 8.4., Table 41) can be updated on-the-fly during a frame acquisition. New values are taken into account by the next frame. The more safe approach would be finishing the last frame acquisition completely, only then update registers or change mode of operations.

Command	Description	Length [Bytes]	Time [µs]
Reset	I²C soft reset of the chip (same effect like the $\overline{\text{RESET}}$ pin)	2	20
Device address activation	I2C device address activation, see chapter 5.4.4	2	20
Single-byte Write	I²C Single-byte write to control registers	3	29
Multiple-byte Write	I²C Multiple-byte write (n bytes) to control registers	2 + n	20 + n x 9
Single-byte Read	I²C Single-byte read from control registers	4	39
Multiple-byte Read	I²C Multiple-byte read (n bytes) from control registers	3 + n	30 + n x 9
Mode set	Sine mode 4, 2, or 1 DCS set using MOD_control registers	3	29
Integration time (short) set	Integration time set (up to 800µs) using Int_len_hi/lo registers	4	38
Integration time (long) set	Integration time set (up to 26s) using INTM_hi/lo, Int_len_hi/lo registers	6	56
Dual Integration time (long) set	Dual int. time set using Int_len_mgx1_*, INTM_*, Int_len_* registers	8	74
Shutter	Soft shutter using Shutter_control registers (shutter_en bit)	3	29
Integration time (short) + Shutter	Int. time + soft shutter in one go! (Int_len_*, Shutter_control registers)	5	47
EEPROM Indirect Single Write	Indirect single write to EEPROM (bypass control registers)	9	20ms
EEPROM Direct Burst Write	Direct byte burst write (n bytes) to EEPROM (bypass control registers)	8 + n	n x 20ms
EEPROM Indirect Single Read	Indirect single read from EEPROM (bypass control registers)	10	97
EEPROM Direct Burst Read	Direct byte burst read (n bytes) from EEPROM (bypass control registers)	8 + n	78 + n x 9

Table 13: I²C Control commands summary

5.4.8. I²C control command examples:

To simplify command sequence definitions, following C-programming language style functions are defined for the I²C master CPU:

- i2cGeneralCall(byte genAdr, byte cmd); // 20 x t_{SCL} = 20µs
- i2cSingleWrite(byte devAdr, byte regAdr, byte regVal); // 29 x t_{SCL} = 29µs
- i2cMultiWrite(byte devAdr, byte regAdr, byte* regVal, byte n // 20 + (n x 9 x t_{SCL}) = 20 + (n x 9)µs
- byte i2cSingleRead(byte devAdr, byte regAdr); // 39 x t_{SCL} = 39µs
- byte* i2cMultiRead(byte devAdr, byte regAdr, byte n); // 30 + (n x 9 x t_{SCL}) = 30+(n x 9)µs

Software reset with I²C general call command

PRECONDITION: None

1. i2cGeneralCall(0x00, 0x06); // Software reset, same effect like $\overline{\text{RESET}}$ pin, 20µs
2. … // Wait for $t_{\overline{\text{RESET}}}$ (> 100ns, see chapter 5.3.2., Software reset via I2C)

continue frame acquisition

Activates device address with I2C general call command

PRECONDITION: None

1. i2cGeneralCall(0x00, 0x04); // Activates I²C device address, 20µs
2. i2cSingleWrite(0x22, 0x92, 0x34); // Start with Sine mode, 4x DCS acquisition on I²C device 0x22...
3. i2cMultiWrite(0x22, ...
4. ...

continue frame acquisition

Sine mode, 4x DCS mode: Acquire DCS 0 … 3 frames with integration time = 10µs

PRECONDITION: All other registers contain default values.

1. i2cSingleWrite(0x20, 0x92, 0x34); // MOD_Control = 0x34 (mod_sel = 00, dcs_sel = 11), 29µs
2. i2cMultiWrite(0x20, 0xA2, &(0x031F), 2); // Int_len = 0x031F (Integration time = 10µs), 38µs
3. i2cSingleWrite(0x20, 0xA4, 0x01); // Shutter_Control = 0x01, (shutter_en = 1), 29µs
4. … // Acquisition starts. Wait until all 4x DCS frames are finished.

done.

Sine mode, 4x DCS mode: Acquire DCS 0 … 3 with integration time = 10µs, followed by DCS 0 … 3 with integration time 200µs

PRECONDITION: All other registers contain default values.

1. i2cSingleWrite(0x20, 0x92, 0x34); // MOD_Control = 0x34 (mod_sel = 00, dcs_sel = 11), 29µs
2. i2cMultiWrite(0x20, 0xA2, &(0x031F), 2); // Int_len = 0x031F (integration time = 10µs), 38µs
3. i2cSingleWrite(0x20, 0xA4, 0x01); // Shutter_Control = 0x01, (shutter_en = 1), 29µs
4. … // Acquisition starts. Wait until all 4x DCS frames are finished.
5. i2cMultiWrite(0x20, 0xA2, &(0x3E7F), 2); // Int_len = 0x3E7F (integration time = 200µs), 38µs
6. i2cSingleWrite(0x20, 0xA4, 0x01); // Shutter_Control = 0x01, (shutter_en = 1), 29µs
7. … // Acquisition starts. Wait until all 4x DCS frames are finished.

done.

Sine mode, 2x DCS mode: Acquire DCS 0, 1 frames with integration time = 10µs

PRECONDITION: All other registers contain default values.

1. i2cSingleWrite(0x20, 0x92, 014); // MOD_Control = 0x14 (mod_sel = 00, dcs_sel = 01), 29µs
2. i2cMultiWrite(0x20, 0xA2, &(0x031F), 2); // Int_len = 0x031F (integration time = 10µs), 38µs
3. i2cSingleWrite(0x20, 0xA4, 0x01); // Shutter_Control = 0x01, (shutter_en = 1), 29µs
4. … // Acquisition starts. Wait until all 2x DCS frames are finished.

done.

Indirect single write to EEPROM: Store 2 bytes at address 0xE8 and 0xE9

PRECONDITION: None

1. i2cSingleWrite(0x20, 0x11, 0xE8); // EE_ADDR = 0xE8, 29µs
2. i2cSingleWrite(0x20, 0x12, 0x22); // EE_DATA = 0x22 (Temp_tl_cal1 = 0x22), 29µs + 20ms = ~20ms
3. i2cSingleWrite(0x20, 0x12, 0x28); // EE_DATA = 0x28 (Temp_tl_cal2 = 0x28), 29µs + 20ms = ~20ms
4. ...

done.

Note 1: Start address is written in EE_ADDR.

Note 2: Each EE_DATA write starts erase/programming EEPROM. Each EEPROM write takes 20ms, then it auto-increments the EE_ADDR by 1.
Note 3: Corresponding control register value is not modified. Only EEPROM register is modified.
Note 4: EEPROM content will only be copied to corresponding control register after RESET.

Direct burst write to EEPROM: Store 8 bytes at address 0xE8 ... 0xEF

PRECONDITION: None

1. i2cMultiWrite(0x20, 0xE8, &(0x22,0x28,0x20,0x26,0x18,0x17,0x19,0x19), 8); // Store 8x bytes, // calibration point registers Temp_xx_cal1/2, 20µs + 8 x (9µs + 20ms) = ~160.1ms
2. i2cSingleWrite(0x20, 0x02, 0x05); // MEM_CTRL_CONF = 0x05 (ee_direct_access = 0), 29µs

done.

Note 1: Start address is sent only one time.
Note 2: Corresponding control register value is not modified. Only EEPROM register is modified.
Note 3: EEPROM content will only be copied to corresponding control register after RESET.

Indirect single read from EEPROM: Read 2 bytes from address 0xEA and 0xEB

PRECONDITION: None

1. i2cSingleWrite(0x20, 0x11, 0xEA); // EE_ADDR = 0xEA, 29µs
2. cal1 = i2cSingleRead(0x20, 0x12); // cal1 = EE_DATA (Temp_tr_cal1 = 0x20), 39µs
3. cal2 = i2cSingleRead(0x20, 0x12); // cal2 = EE_DATA (Temp_tr_cal2 = 0x26), 39µs
4. ...

done.

Note 1: Start address is written in EE_ADDR.
Note 2: Each EE_DATA read auto-increments the EE_ADDR by 1.
Note 3: Corresponding control register value is not modified. Only EEPROM is read.

Direct burst read from EEPROM: Read 8 bytes from address 0xE8 ... 0xEF

PRECONDITION: None

1. i2cSingleWrite(0x20, 0x02, 0x07); // MEM_CTRL_CONF = 0x07 (ee_direct_access = 1), 29µs
2. cal_arr[] = i2cMultiRead(0x20, 0xE8, 8); // Read 8 bytes calibration points Temp_xx_cal1/2, 20µs + 8 x 9µs = 92µs
3. i2cSingleWrite(0x20, 0x02, 0x05); // MEM_CTRL_CONF = 0x05 (ee_direct_access = 0), 29µs

done.

Note 1: Start address is sent only one time.
Note 2: Corresponding control register value is not modified. Only EEPROM is read.

5.5. LED driver

The LED driver has two different outputs LED and LED2.

The LED output is an open-drain MOSFET device. It can sink up to 500mA peak current during operation. The modulation signal is a square-wave up to typically 20MHz with a duty cycle of 50% which depends on the mode of operation.

IMPORTANT: There are non-modulating DC modes (e.g. grayscale with LED/LD illumination) which keeps the LED driver always turned on. In this case, the user has to take care that LED driver and the epc635 chip does not exceed the maximum operating limits.

The LED2 output is a digital push-pull driver output.

The LED_driver register contains several static bit fields that can be used for setting drive strength, polarity, etc. depending on the external LED/LD circuitry used in the application (see chapter 8.7.6.). These bit fields must not be modified during the frame acquisition.

- led2_en bit enables the LED2 push-pull driver. LED and LED2 must not be simultaneously used. When LED2 is used, LED can be turned off by led_drv_en = 0
- The led_drv_sel bits set LED driver output maximum current e.g. for 200mA to 100% , 64%, 46% or 33%. Refer to chapter 8.7.6, LED_driver.
- The led_on bit permanently turns on the LED/LD (torch function). Keep care to the power dissipations if you operate the LEDs and laser diodes in this mode.
- The led_drv_en bit enables the internal LED driver circuits. The enabled after reset (default), can be disabled if there will be no frame acquisition for a long time.
- The led_inv_en bit sets the polarity of the LED driver (inverts LED and LED2 pin). This must be set/cleared with respect to external LED/LD driver circuit topology and stored in the EEPROM. It is cleared after reset (default), which matches to the simple external LED circuit shown in Figure 14.
- led_ssr_en bit sets a slower slew rate for the LED driver (part of the LED driver transistor is switched on with a delay to lower the current peaks).

5.6. DLL (Delay Locked Loop)

The LED driver and the LEDs/LDs may shift the phase depending of temperature, ageing, etc.. The deviations are from seconds up to several minutes and longer and impact the performance of the distance accuracy. The built-in DLL circuit minimizes this behaviour by compensating the delay of the critical signals. It is located between the modulator and the LED driver (see Figure 26). LED pin is used only in this description for simplification. Nevertheless, the explanation is valid for both LED and LED2 output.

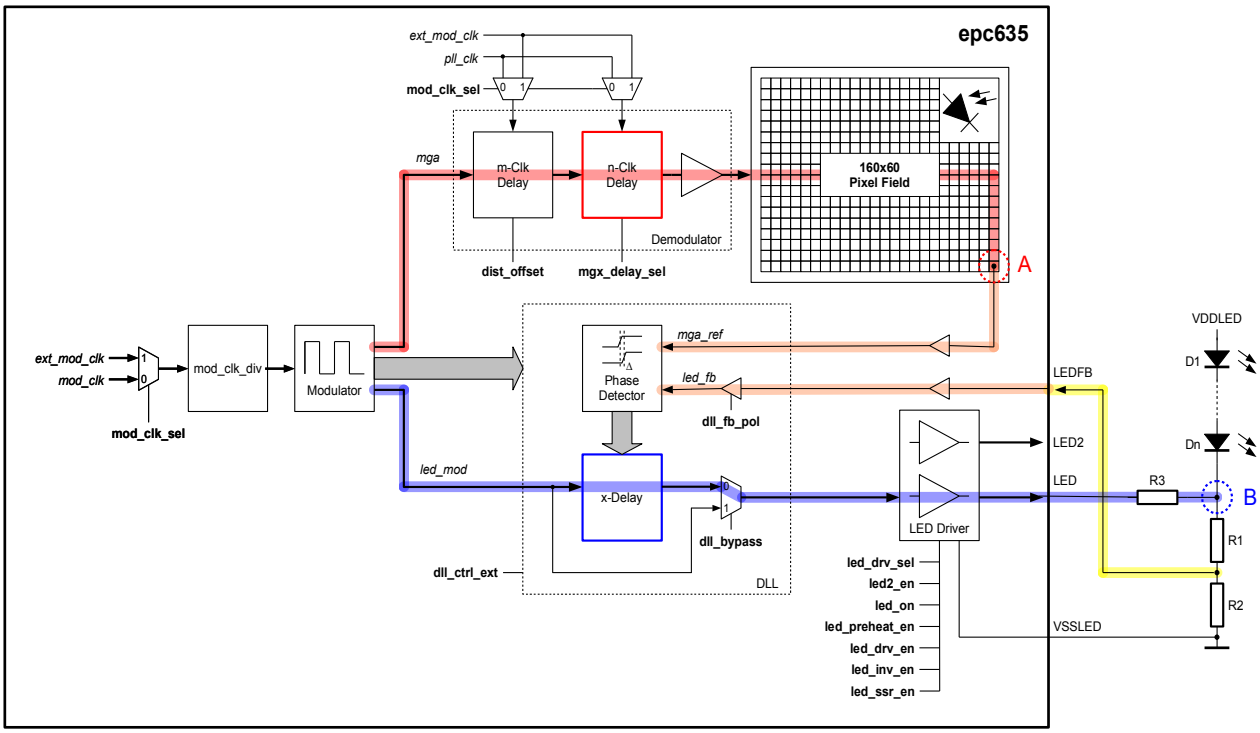


Figure 26: Simplified modulator, demodulator, DLL and LED driver diagram

The modulator invokes the DLL before DCS frame acquisition at the rate set by the DLL_measurement_rate register (see Figure 27, t_{SYNCH}). This stimulates the mga demodulation signal. After passing digital delay stages, it is distributed to the pixels in the pixel-field (Figure 26, red path, point A). The same time, the led_mod signal passes the DLL, the LED driver, the LED pin until it comes to the LED circuit (blue path, point B) which generates the modulated IR light. The DLL compares the demodulation signal phase coming from the pixel (point A) with the feedback signal (LEDFB pin) the LED circuit (point B). The distance reading error is minimal if the delays on the signals from points A and B are identical. The DLL minimizes deviations by adding a certain delay on the led_mod signal (x-Delay).

Figure 27 shows the example for 4x DCS frames and a DLL_measurement_rate = 4.

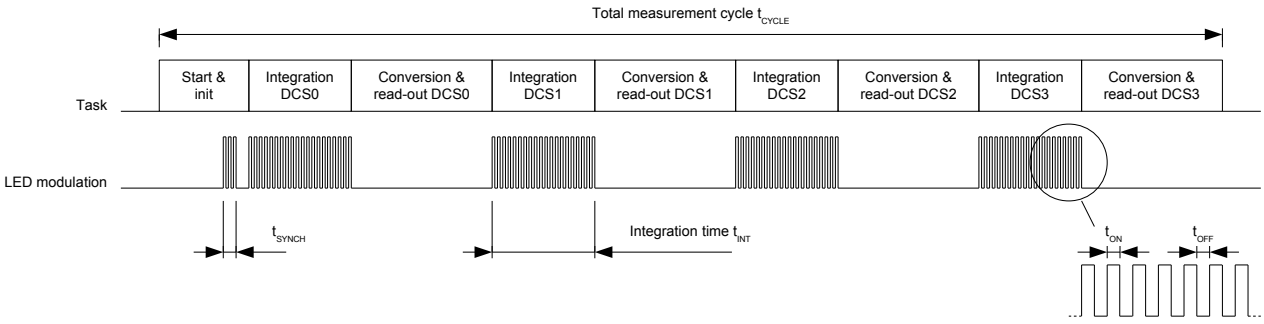


Figure 27: DLL synchronization example for 4x DCS frames

The DLL period t_{SYNCH} is skipped for DLL_measurement_rate = 0. The last compensation delay value is preserved.

Design notes:

- Minimize the path delay from the feedback node in the resistor divider to the LEDFB pin (yellow).
- Place the resistor dividers as close as possible to the LEDFB pin on the PCB.
- Noisy circuits should be placed away from the LEDFB pin and resistor dividers.

5.6.1. DLL locking operation

Before starting the DLL locking operation, the mga_ref signal must be delayed behind the led_fb signal (see Figure 28) by setting mgx_delay_sel = 1 or higher in the register Demodulation_delays. It gives an initial 12.5ns phase offset to the mga_ref signal. Next, the modulator toggles the mgx and led_mod signals without enabling the DLL (DLL pre-synchronization: DLL_en_del). Finally, the modulator enables the DLL locking operation with mgx and led_mod signals toggling until the end (DLL_en).

The default DLL pre-synchronization is 10µs and DLL lock time 30µs. It is indicated as t_{SYNC} in Figure 27. These values can be changed by DLL_en_del/DLL_en registers (see Table 41 , DLL Synchronization registers).

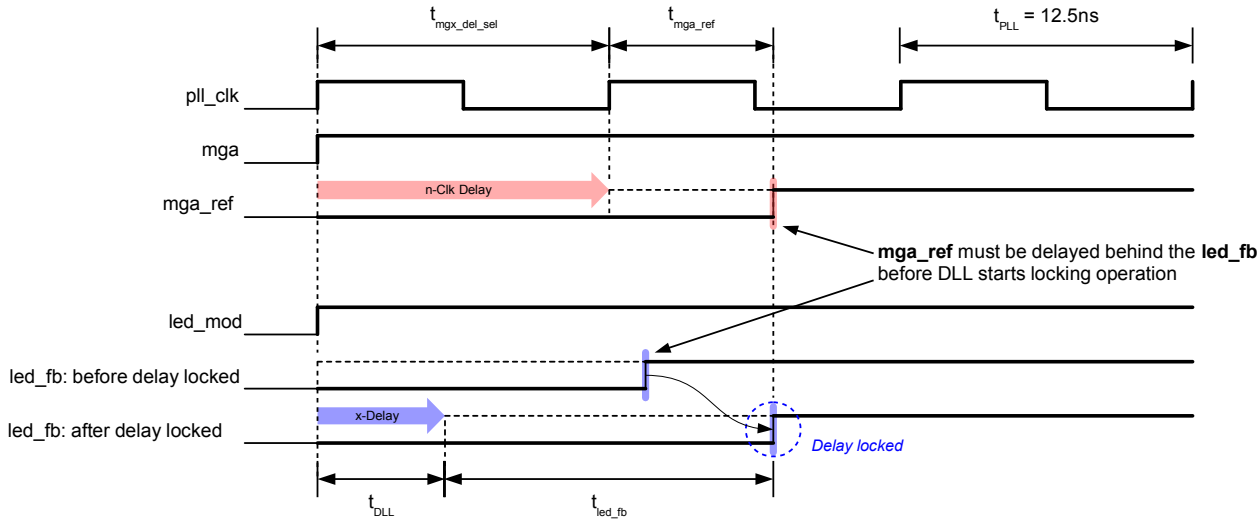


Figure 28: DLL delay locking operation

If locked, the compensation delay is frozen (x-Delay) and the mgx and led_mod signals are in phase (point A and B) until next synchronization cycle. Set the DLL synchronization rate (DLL_measurement_rate register) in multiples of the DCS frame number. Example: Set DLL_measurement_rate = 4 if the system runs the 4x DCS mode (dcs_sel = 11 in the MOD_Control.dcs_sel register). All DCS frames (0-3) are acquired this way with the same DLL compensation delay and the samples stay correlated with each other.

DLL_measurement_rate = 0: Holds last compensation delay and uses it for the next DCS frames.

IMPORTANT:

- The DLL has a delay locking range of 6ns ... 76ns with 7ps fine delay steps.
- When the DLL is enabled, mgx_del_sel must be 1 or higher because in addition to the delays from the LED driver, LED/LD circuit and LEDFB input a minimum 6ns delay is inserted on the led_mod line.
- The value of the mgx_del_sel register must be > 1 if the delay variation of LEDF signal is more than one pll_clk period. This allows to compensate this wide-range variation, e.g. for LED/LD circuits placed far away from the epc635 and causing long time offsets.

5.6.2. DLL filters

Two digital filter stages which average by counting the number of LED pulses (DLL_filter_control.dll_filter_ctrl_s1_sel) and DCS frames (DLL_filter_control.dll_filter_ctrl_s2_sel), allow to adapt the DLL locking operation to slow changing environmental conditions, such as temperature effects. The filter control loop re-applies averaged lock values only at those programmed count times back to DLL's delay line registers. DLL filter s1 allows binary count steps of 8, 16, 32, … 512 LED pulses; whereas DLL filter s2 uses 2, 4, 8, … 128 DCS frame steps. Optimal s1 and s2 settings depend on the application system.

5.7. Pixel-field

5.7.1. Pixel coordinates

The epc635 pixel-field consists of a total of 168 x 72 pixels. 4 rows top/bottom and 6 columns left/right on the periphery of the pixel-field contain dummy pixels. The active pixels for TOF and grayscale imaging modes are in the central rectangle area forming a full Half-QQVGA resolution 160 x 60 pixel (Figure 29).

The upper-left corner (top view on chip) is the origin (4/6) of the epc635 pixel-field. X-axis starts at 4 and counts up to 163 to the right. Pixel y-axis starts at 6 and counts up to 65 to the bottom. All readout modes and control registers use this coordinate system to set or change modes of the chip.

Readout starts at the bottom of the pixel-field and continues towards up, row by row.

The internal readout of a row is split in two sections: first all even pixels; second all odd pixels. Later on the TCMI interface presents the row in the regular order with even and odd pixels mixed.

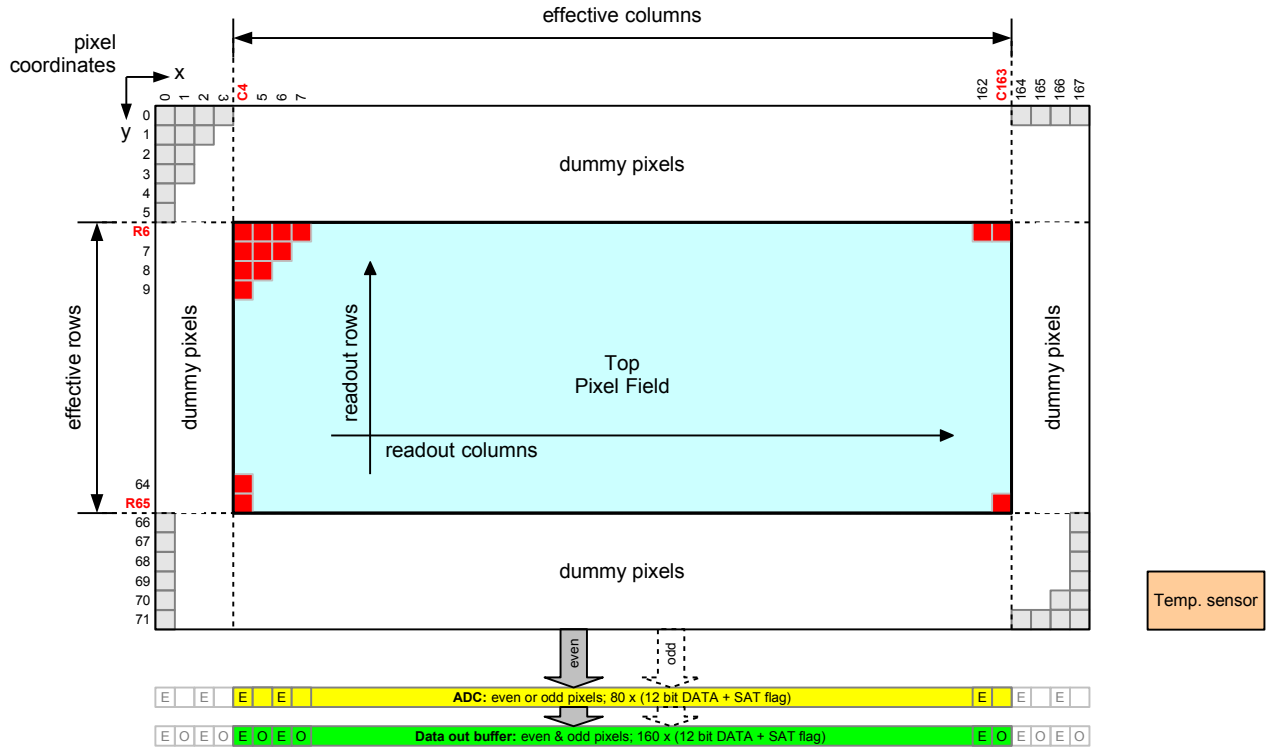


Figure 29: pixel-field coordinates with row and column numbering scheme
(top-view, solder balls are bottom side)

5.7.2. Pixel architecture

The pixels are placed in groups 2x2 pixels, called herein “pixel group”. The pixel group performs two basic operations: Measurement (integration) and readout (ADC). Pixels are named as UE (Upper-row, Even-column), UO (Upper-row, Odd-column), LE (Lower-row, Even-column) and LO (Lower-row, Odd-column) depending on their location within the pixel group (see Figure 29). Pixels with the same name are controlled simultaneously in the whole pixel-field. More precisely, pixels in the upper and lower rows are controlled simultaneously during measurement, pixels in the even and odd columns are controlled simultaneously during readout.

The pixel group architecture allows the epc635 to operate the pixel-field in different modes and in combinations thereof according the following chapters.

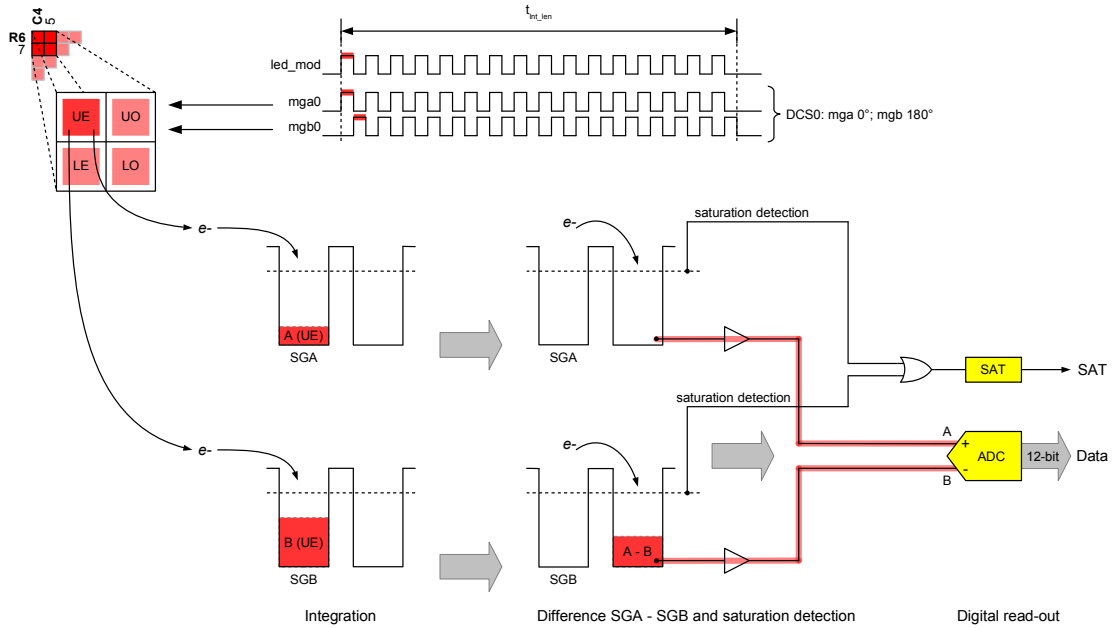


Figure 30: The 2x2 pixel group and the simplified function overview

Each pixel has its own pair of storage gates SGA and SGB. During the integration time, they accumulate the charges (e-) created by the reflected modulated light coming from the object (see section 6., Measurement Modes). They are controlled by the mga and mgb demodulation signals. After the measurement is finished, the readout phase starts. The charges stored in the storage gates SGA and SGB are read out as a difference A – B (ambient-light suppression) and converted into a single 12-bit digital value and a 1 bit saturation flag.

Depending of the operation mode the ADC readout is different:

Differential mode:	Used for 3D TOF sine. It shows the difference of the two storage gate charges. Digital output values are signed values.
Single-ended mode:	Used for grayscale images. It shows the charge of the selected storage gate. For the SGA readout the conversion result is a positive 11-bit value (except noise). For the SGB readout the conversion result is a negative 11-bit value (except noise).

IMPORTANT: Refer to chapter 8.7.21. for setting correct data format 2's complement integer (tcmi-data-polarity).
The application software must take care about the negative (-) single ended sample value, accordingly.

Single MGX mode (default)

When dual_mgx_mode = 0, all UE, UO, LE, LO storage gates work simultaneously during measurement operation. The storage gate control signals mga, mgb that are coming to all pixels are driven identically. In this mode, one DCS frame (0 …3) is acquired at the time (see Figure 31).

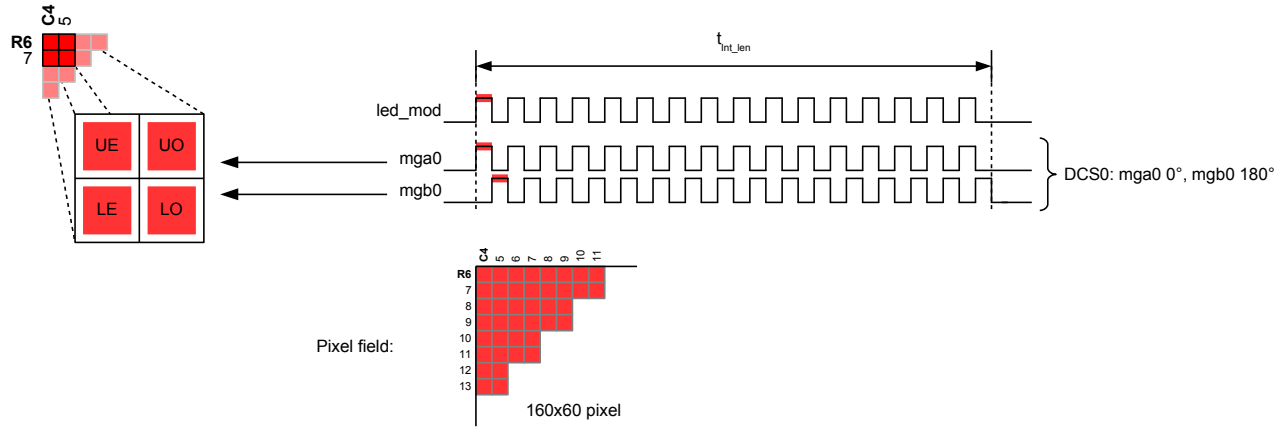


Figure 31: Single MGX mode: upper and lower pixel rows are controlled identically with mgx0

Dual MGX mode with phase-shifted integration time (motion blur reduction)

When dual_mgx_mode = 1 and int_len_mgx1_en = 0, the UE, UO, LE, LO storage gate control signals mgx (mga, mgb) are generated as two independent groups as mgx0 (mga0, mgb0) and mgx1 (mga1, mgb1). As a result, two correlation frames DCS0 and DCS1 (or DCS2 and DCS3) can be acquired simultaneously (see Figure 32). The upper row pixels store DCS0 (or DCS2) while the lower row pixels store DCS1 (or DCS3). The vertical pixel pairs (e.g. UE/LE) must be treated for distance calculation as if they are one single pixel. This comes at the cost of a reduced resolution along the y-axis. The result provides a total of 160x60 pixel-field readout with an effective 3D TOF resolution of 160x30 pixel.

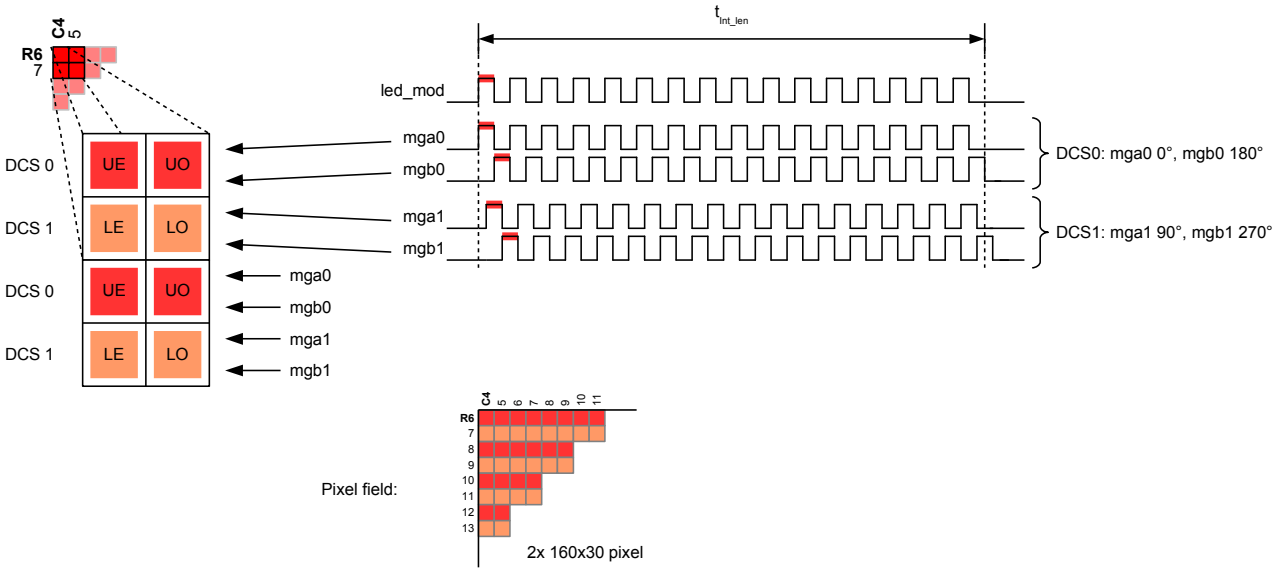


Figure 32: Dual MGX mode with phase-shifted integration time
(upper and lower rows independently controlled by mgx0 and mgx1 with different phase shifts)

IMPORTANT: This mode requires that adjacent pixels look to the same point on the target and receive the same amount of light. Otherwise, calculated distance values are not reliable.

Dual MGX mode with different integration times (High dynamic range)

When dual_mgx_mode = 1 and int_len_mgx1_en = 1, the UE, UO, LE, LO storage gate control signals mgx (mga, mgb) are generated as two independent groups as mgx0 (mga0, mgb0) and mgx1 (mga1, mgb1). Both groups provide exactly the same DCS modulation signals (phases). One stops earlier than the other due to different integration times (see Figure 33). The effect: As a consequence, the two pixels collect different amount of light simultaneously. There is no restriction about which integration time is shorter or longer with respect to the other. The upper row pixels integrate with Int_len_hi/lo register value, while the lower row pixels integrate with Int_len_mgx1_hi/lo value. The upper and lower pixels (e.g. UE, LE) can be used independently for distance calculation. This comes at the cost of a reduced resolution along the y-axis. Instead of one frame with 160x60 pixels, a single readout provides two DCS or black and white frames with an effective resolution of 160x30 pixels but with different integration times.

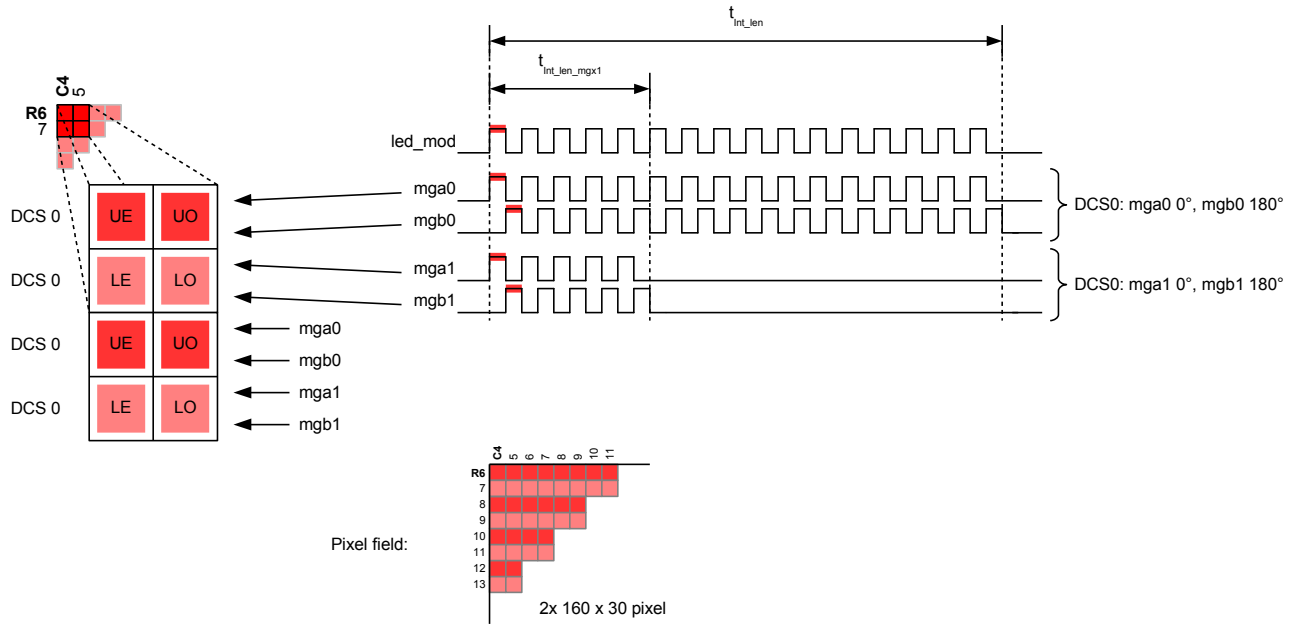


Figure 33: Dual MGX mode with different integration times
(upper and lower rows independently controlled by mgx0 and mgx1. One stops earlier than the other)

ADC conversion speed-up

Applications which need e.g. only 8-bit resolution e.g. grayscale mode, the frame rate can be increased further by reducing the resolution of the ADC conversion from 12-bit default to 8, 9, 10, 11-bit optional which reduces the conversion time according Table 14.

The trade-off for 8-bit resolution (e.g. grayscale mode, setting adc_res_sel = 1 and adc_numlsb_sel = 00) is an increase of the overall frame rate by more than 20%.

The bit-fields adc_res_sel and adc_numlsb_sel in ADC_ramp register (see chapter 8.7.22) select the ADC read-out resolution. The SIR_lo register (0x9D) and the ISOURCE_CLK_divider (0x88) must be adjusted in parallel. Table 14 shows valid combinations to get the best performance.

This mode can be combined with the 8-bit TCMI data format by setting tcmi_data_format_sel = 11 to effectively interface to an 8-bit parallel interface CPU/FPGA in the application (see chapter 8.7.20).

Register		ADC Resolution				
Name	**Address**	**8-bit**	**9-bit**	**10-bit**	**11-bit**	**12-bit**
ADC_ramp	0xCD	0x1B	0x5B	0x9B	0xDB	0x13
ISOURCE_CLK_divider	0x88	0x27	0x2E	0x34	0x3A	0x3F
SIR_lo	0x9D	0x34	0x3B	0x42	0x49	0x50
Conversion time		12.25 µs	13.125 µs	14 µs	14.875µs	15.75 µs

Table 14: ADC resolution setting for signed data format (adc_skip_signmsb_en='0', default)

Skipping sign bit in grayscale mode

Grayscale mode operates ADCs in single-ended mode (refer to chapter 5.7.2) and the sign-bit is always generated as the 1st MSB. This is regardless of the unsigned or two's-complement mode selected by the tcmi_data_pol bit in the TCMI_polarity register. Therefore, when 8-bit ADC resolution is combined with 8-bit TCMI data format, sign-bit (MSB) gives out always a fixed value 1 for unsigned, 0 for two's-complement hence samples are effectively generated as 7-bit resolution.

The sign bit will be sipped by setting adc_skip_signmsb_en = 1 together with adc_res_sel =1 and adc_numlsb_sel = 00 outputs 8-bit effective resolution. It forces ADCs to 9-bits resolution (sign-bit + 8-bit data, with a penalty of +0.875µs conversion time) during read-out and skips the sign-bit during data-out. It sends 8-bit effective data via TCMI.

Setting the adc_skip_signmsb_en =1 will yield 10, 11, 12-bits resolution when combined with adc_numlsb_sel = 01, 10, 11, respectively. Table 15 shows valid combinations to get the best performance.

Register		ADC Resolution				
Name	**Address**	**8-bit**	**9-bit**	**10-bit**	**11-bit**	**12-bit**
ADC_ramp	0xCD	0x5B	0x9B	0xDB	0x13	n/a
ISOURCE_CLK_divider	0x88	0x2E	0x34	0x3A	0x3F	n/a
SIR_lo	0x9D	0x3B	0x42	0x49	0x50	n/a

Table 15: ADC resolution setting for unsigned data format (adc_skip_signmsb_en = 1)

5.7.3. Pixel saturation detection

The pixels collect continuously modulated and non-modulated ambient light during the integration period. Depending on these light intensities, sometimes the pixels collect more charge (over-exposure) than they can accommodate in their storage gates (refer to Figure 30). In such a case, the 12 bit sample data is not valid and cannot be used. Therefore, each pixel generates a “saturation detection” flag along with the sample data, so that the data can be discarded by the application.

The saturation flag is generated in the pixel binning modes as well. It is set when one or more storage gates of the binned pixels is/are saturated or the sum thereof. Due to this architecture, two binned pixels cannot collect twice the full storage gate charge.

The saturation detection flag is transmitted via XSYNC_SAT_CFG pin synchronously to the DATA[11:0] for every 12 bit pixel sample. If XSYNC_SAT_CFG pin is programmed for an another function by setting tcmi_xsync_sat_sel = 0 or not connected at all, tcmi_data_sat_en = 1 can optionally drive all DATA[11:0] pins to 0xFFF when the pixel is saturated (see TCMI_polarity register in chapter 8.4., Table 41, TCMI registers). In this configuration, the application must discard those samples equal to 0xFFF, otherwise use it for calculation.

5.7.4. Ambient-light suppression

An important function of the 3D TOF pixel is the ambient-light suppression. It is the ability to separate the reflected modulated light and suppress the the ambient light. The user has not to take care of this because it is built-in in the pixel. It removes the common mode distortion (DC or low frequent signal) caused by foreign light sources (e.g. sunlight, daylight, room illumination, etc) automatically from the measurement signal by using only the charge difference in the storage gates (see Figure 30).

Similar to the system sensitivity to modulated light, is the ambient-light suppression also a function of the integration time. The longer the integration time, the more unwanted light will be detected.

Table 7 lists the minimum values for the ambient-light suppression as a function of the wavelength and as well as compared to sunlight.

Notes:

- The ambient-light suppression of the chip must not be mistaken with the ambient-light measurement. It is a fixed built-in functionality which is removing the ambient-light from the measurement signal only.
- DC and AC photo-signals can be generated by ambient-light (e.g. sunlight) or by cross-talk from the IR-LEDs. However, if these are above the stated maximum values, the sensor or the input electronics are saturated. This blocks the detection of the TOF modulation signal.

5.8. Temperature sensor

A temperature sensor is located near the pixel-field (Figure 29). It is not calibrated and reads the values as they are. Values can be read on-the-fly via I²C interface while frame acquisition is going on. The sensor is sampled synchronously to the row readout cycles. Four consecutive readings are summed up to the resulting 14 bit value written into the Sum_Temp_xx_hi/lo registers (see Table 40, Temperature sensor registers). After frame start, four row readout cycles must be performed for getting a valid value because every frame-start resets the temperature sensor.

An approximative temperature can be calculated from the values Sum_Temp_ti to (chapter 8.6.6), Temp_ti_cal1, Temp_ti_cal2 (chapter 8.7.24) according the following formula

$$\text{Temperture T [ºC]} = \frac{\left(\frac{\text{Sum\_Temp\_ti}}{4} - 2'048\,\text{LSB}\right) - 1'400\,\text{LSB} + \left(\text{Temp\_ti\_cal1} - 127\text{LSB}\right)}{4.68\,\text{LSB/ºC} + \left(\text{Temp\_ti\_cal2} - 127\text{LSB}\right) \cdot 0.01\,\text{LSB/ºC}} + 27\text{ºC}$$

5.9. TOF camera interface (TCMI)

The TOF Camera Module Interface (TCMI) is a programmable high-speed parallel data output interface to down-load the pixel data.

The TCMI interface provides:

- Programmable active high/low logic levels on all clock and control signals individually (default: all active high).
- Programmable output clock rates.
- 12-bit parallel output with 1 bit saturation flag transferred in two bytes on TCMI interface.
- The output data format is programmable: 12-bit mode, lsb/msb split mode, msb/lsb split mode, 8-bit mode.
- Programmable 'ITU-R 656 like' synchronization options which allows easy connection to a wide variety of high-performance low-cost SoC embedded processors. It supports hardware synchronization, but not embedded 0xFF/0x00 sync data packets.

When the integration period is terminated and ADC conversion is finished, the readout results are moved into the data out buffers to be immediately transmitted via the TCMI interface. The conversion of one row takes in total 31.5µs.

Depending of the mode selection (4x DCS, 2x DCS, ...) a programmable number of DCS frames are generated by the epc635. The generated data is streamed out as a complete block of 1 DCS frame, one after the other following the procedure described in chapter 5.7. Each row contains 12-bit DCS values and the SAT bit.

The pixel values are streamed out as 12 bit signed numbers. The rows are streamed out in sequence from the bottom to the top e.g. R65 (C4, C5, … C163), R64 (C4, C5, … C163) and so on until R6 (C4, C5, … C163). The stream-out of one row takes 16µs with default clock settings (40MHz TCMI clock rate).

The transfer of a DCS frame should not be interrupted or stopped, once it is started. The application should have enough bandwidth to receive all transmitted frames.

IMPORTANT: Refer to chapter 8.7.21. for setting correct data format 2's complement integer (tcmi-data-polarity).
The application software must take care about the negative (-) single ended sample value, accordingly.

5.9.1. TCMI clock mode

The TCMI supports two clock modes: Continuous clock (default) and gated clock. Both modes use internal tcmi_clk as a time base. The user must program the appropriate clock speed before running the TCMI. It sets the output data transfer rate (refer to signal DCLK).

The frequency is programmable to 10, 20, 40, 80MHz. The user application is responsible to set the correct clock rate in order to achieve the most efficient data acquisition throughput for the target application. The choice depends on user system factors such as e.g. application CPU/DSP's GPIO and/or bus speeds.

tcmi_clk_div (0x89)			
10MHz DCLK	**20MHz DCLK**	**40MHz DCLK**	**80MHz DCLK**
[#]	**[#]**	**[#]**	**[#]**
4	2 (default)	1	0

Table 16: DCLK settings

Continuous clock mode

This mode is selected by I2C_TCMI_control.tcmi_dclk_mode = 0. The frames are transmitted at high-speed using all *SYNC (VSYNC_A0, HSYNC_A1, XSYNC_SAT_CFG), DATA[7:0] and DCLK outputs (Figure 34).

The DCLK signal toggles continuously.

All *SYNC*, DATA[7:0] signals are synchronously updated with the positive edge of the DCLK signal when its polarity is set as active-high; with the negative edge of the DCLK signal when its polarity is set as active-low. The non-active edge of the DCLK output can be used by the receiving end (application CPU) as a sampling clock. It should approximately be in the center of the data.

By using the default configuration, the active states of VSYNC_A0 and HSYNC_A1 signals indicate blanking periods during the frame transmission. It is a legacy feature from the ITU-R 656 standard for video signal transmission. While DCLK toggles continuously, any data during the blanking periods are not valid and must be ignored.

As soon as the measurement result of the first row of the new frame is available, VSYNC_A0 and HSYNC_A1 are set consecutively with the next active edge of DCLK. VSYNC_A0 is active from the start until the end of the each complete frame. Whereas, HSYNC_A1 indicates the validity of the DATA[7:0] and XSYNC_SAT_CFG (saturation bit) from the start until the end of a row.

By default, the XSYNC_SAT_CFG pin is used for the saturation bit. Optionally, it can be programmed to indicate the end of a frame (tcmi_xsync_sat_sel = 0).

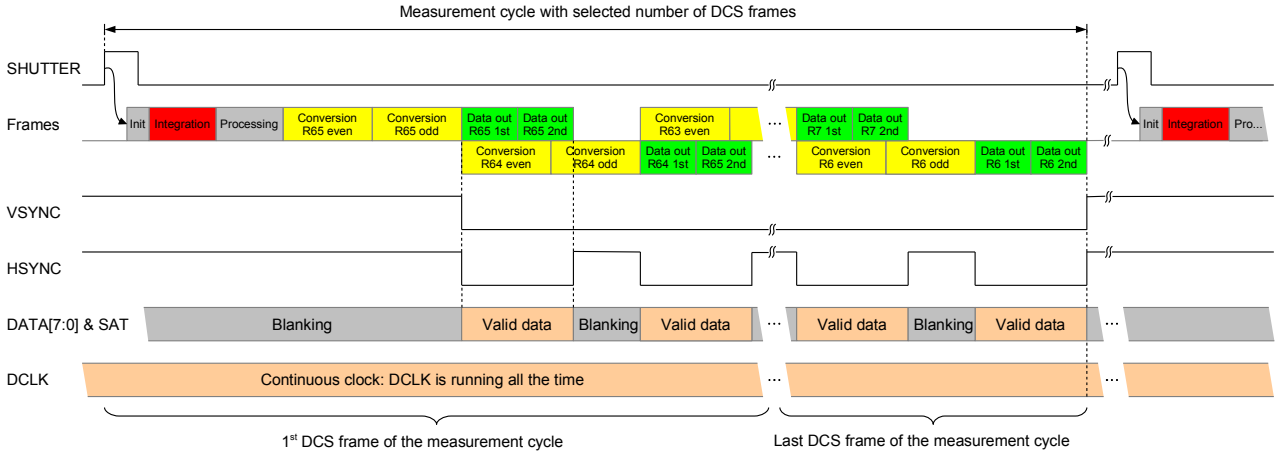


Figure 34: Continuous clock mode

Gated clock mode

This mode is selected by I2C_TCMI_control.tcmi_dclk_mode = 1. The frames are transmitted at high-speed using only the DATA and DCLK outputs (Figure 35). This allows the interfacing to embedded processors with standard parallel GPIOs. The external application should calculate begin/end of the frame/row timing before starting the acquisition.

DCLK signal is generated only during non-blanking/valid data periods (corresponds to HSYNC_A1 inactive). Its frequency is same as in the other mode. The DCLK toggle duration can be programmed to run few more cycles than HSYNC_A1 is active effectively. This allows the external processor to finish its last operations with its FIFO/DMA.

All *SYNC* signals do not toggle in this mode, but can be optionally enabled.

All *SYNC* (optional), DATA[7:0] signals are synchronously updated with the positive edge of the DCLK signal when its polarity is set as active-high; with the negative edge of the DCLK signal when its polarity is set as active-low. The non-active edge of the DCLK output can be used by the receiving end (application CPU) as a sampling clock. It should approximately be in the center of the data.

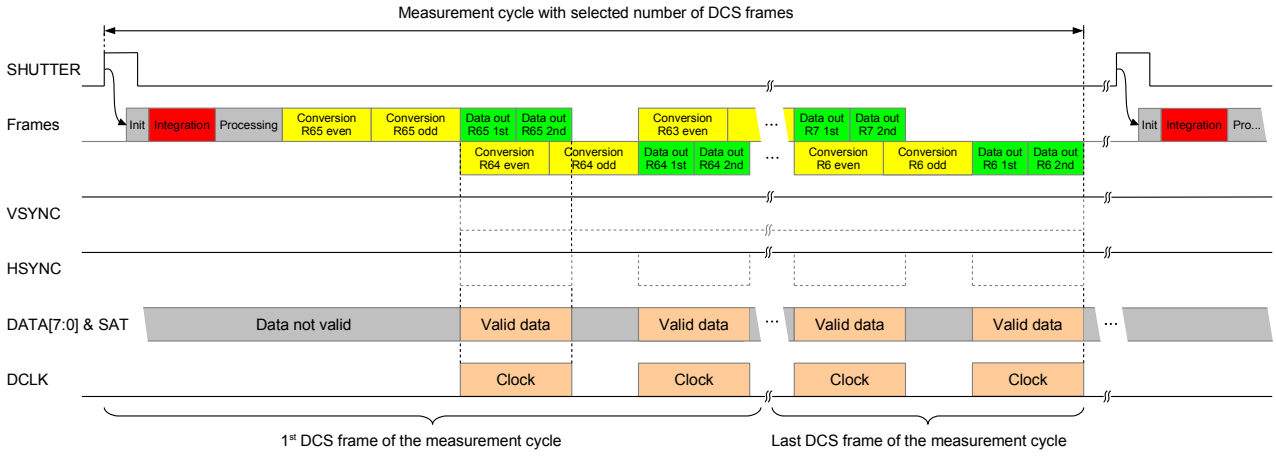


Figure 35: Gated clock mode

5.9.2. Single or continuous measurement control

Single measurement control

The selected measurement mode (4x DCS, 2x DCS, grayscale, …) defines, how many frames the chip performs by the stimulation of one SHUTTER pulse for a measurement cycle. This pulse can be applied either by the HW SHUTTER pin or by SW control of the shutter_en bit. Whereas the SW controlled SHUTTER is auto-cleared after propagation, the HW Shutter needs a minimum hold time of 250ns and must be set back manually latest before the HSYNC_A1 signal of the last row pair of the last DCS frame (last HSYNC_A1 of the last frame).

During such a measurement cycle, the next frame acquisition starts immediately after the last data readout on the TCMI interface until all frames are performed.

Continuous measurement control (auto-run)

As long as in the Shutter_Control register the multi_frame_en = 1 is set or the HW SHUTTER is applied during the readout of the last row pair of the last frame, the epc635 runs in a non-stop measurement mode. The chip starts immediately next measurement cycle if the actual one is terminated (Figure 43). If the trigger arrives before the readout of the last row pair of the last frame, then it is ignored.

5.9.3. TCMI data format

TCMI supports three 8 bit transfer formats:

- msb/lsb split mode: Transfers 12 bit pixel data with MSByte leading and LSByte trailing with 2x DCLK. Refer to Table 17 and Figure 36.
- lsb/msb split mode: Transfers 12 bit pixel data with LSByte leading and MSByte trailing with 2x DCLK. Refer to Table 18 and Figure 37.
- 8-bit mode: Transfers the 8 MSB bits of the pixel data with 1x DCLK. Refer to Table 19 and Figure 38).

The three modes require lines DATA[7:0] to be connected in the application. The TCMI data format can be selected in the register I2C_TCMI_control. tcmi_data_format_sel (refer to I2C_TCMI_ register).

IMPORTANT: The application should not set tcmi_data_format_sel = 00.

The two split modes transmit pixel values in two consecutive DCLK cycles. As a result HSYNC time is doubled. When 8 bit precision is enough, the application should use 8-bit mode and set the DCLK to half of the speed (e.g.: 10MHz instead of 20MHz default). This way, the overall DCS frame transfer time remains like in split modes and frame rate is not reduced.

1st Byte: MSByte								2nd Byte: LSByte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	0	0	0	SAT

Table 17: TCMI msb/lsb split mode, HW synchronization data format

1st Byte: LSByte								2nd Byte: MSByte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
b3	b2	b1	b0	0	0	0	SAT	b11	b10	b9	b8	b7	b6	b5	b4

Table 18: TCMI lsb/msb split mode, HW synchronization data format

Byte							
D7	D6	D5	D4	D3	D2	D1	D0
b11	b10	b9	b8	b7	b6	b5	b4

Table 19: TCMI 8-bit mode , HW synchronization data format

TCMI data formats can be combined with the ADC conversion speed up. Refer to chapter 5.7.2 for ADC resolution v.s. conversion time setting (DC_ramp.adc_res_sel, ADC_ramp.adc_numlsb_sel and SIR_hi/lo registers).

The saturation flag can be optionally inserted into the DATA[0] of the LSByte by setting tcmi_8bit_data_sat_en = 1 during the 1. or 2. DCLK cycle for the msb/lsb or lsb/msb split modes, respectively. This feature is not available for the 8-bit mode. In this case either the XSYNC_SAT_CFG pin can be used along with the DATA[*] pins or the tcmi_data_sat_en = 1 must be set to force all DATA[*] = 0xFFF when the corresponding pixel is saturated.

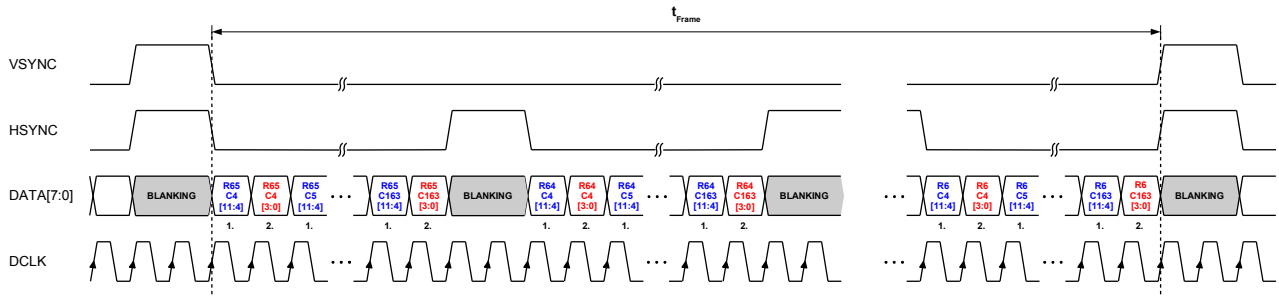


Figure 36: msb/lsb split mode

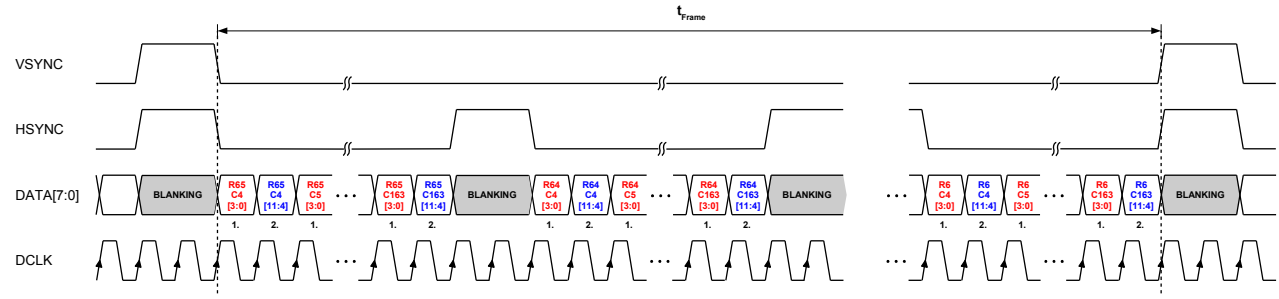


Figure 37: lsb/msb split mode

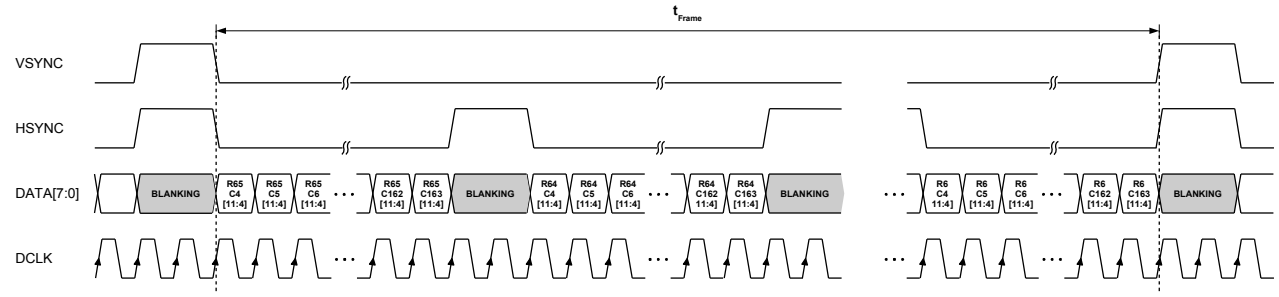


Figure 38: 8-bit mode

5.9.4. TCMI embedded synchronization mode

Embedded Synchronization Mode ESM uses only DCLK and DATA[*] bus to transmit the whole image. It eliminates using hardware VSYNC, HSYNC, XSYNC signals for synchronization. By setting I2C_TCMI_control.tcmi_esm_en = 1 (see chapter 8.7.20), ESM embeds special data packets – so called “Labels”, before and after every frame and row to mark the begin and the end of the actual pixel samples on the TCMI data bus.

The synchronization labels consist of 4 consecutive bytes, starting always with 0xFF, followed by 0x00, 0x00, ending with a unique byte as defined designator as given in Table 20. Label ending bytes can be customized by TCMI_ESM_FS, TCMI_ESM_FE, TCMI_ESM_LS and TCMI_ESM_LE registers (see 8.6.3).

Label	4-Byte Data Packet	Description
FS	0xFF 00 00 1E	Frame Start
FE	0xFF 00 00 E1	Frame End
LS	0xFF 00 00 AA	Line Start
LE	0xFF 00 00 55	Line End

Table 20: TCMI ESM labels

The receiver/application continuously parses the incoming data for ESM labels and strips out the image data marked between LS-LE pairs. Figure 39 Illustrates an example of a DCS frame transfer.

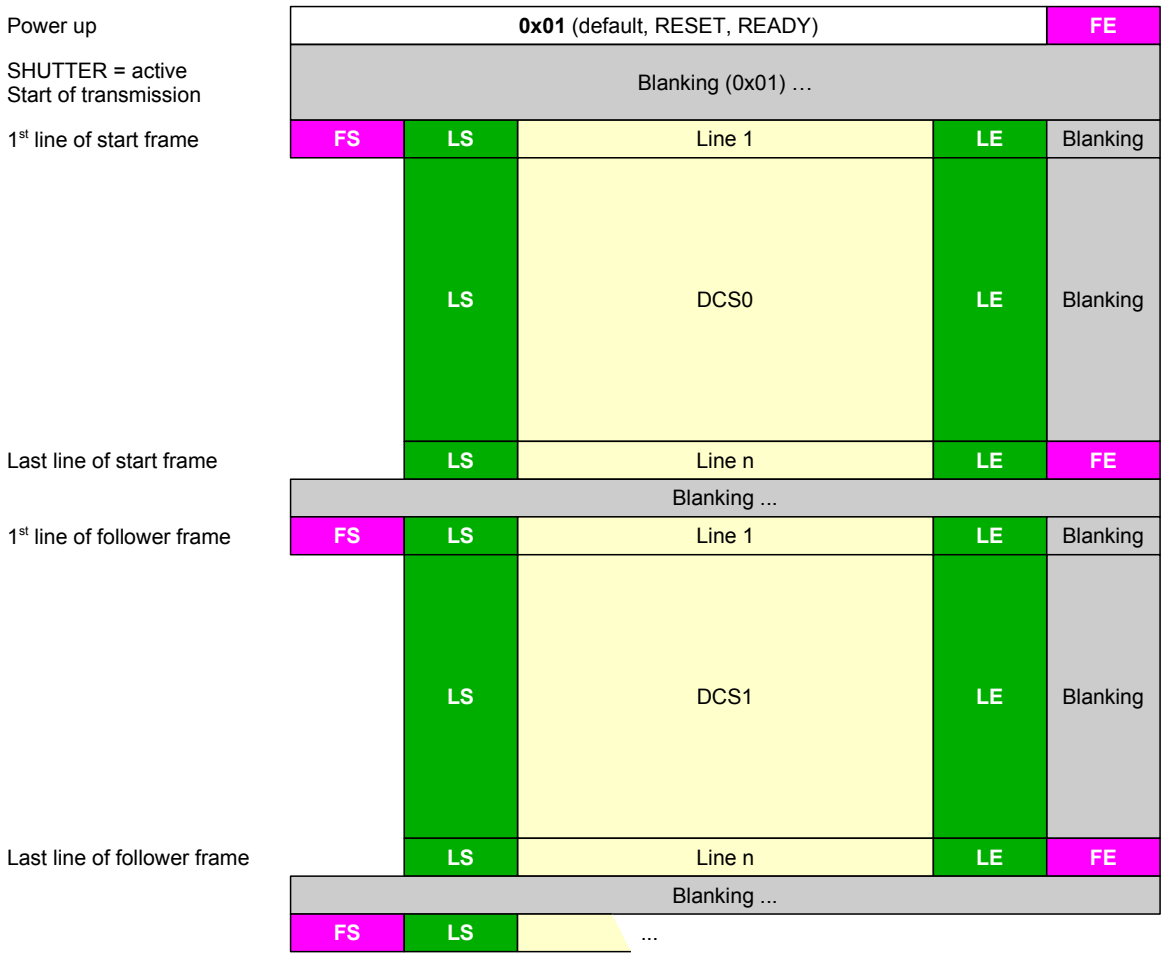


Figure 39: TCMI ESM frames

Values 0x00 and 0xFF in the actual image data will not occur due to the special construction of labels. Therefore, image data is mapped to values between 0x01 … 0xFE. Is the the TCMI bus idle (i.e. blanking), TCMI keeps DATA[7:0] = 0x01 (default value of the bus after reset).

ESM data mapping details for different TCMI data formats are defined in Table 21, Table 22 and Table 23:

D0 … D7 DATA0, DATA1, … DATA7 lines of the TCMI bus.

b0 … b11 pixel value

SAT, $\overline{SAT}$ saturation information

1st Byte: MSByte								2nd Byte: LSByte							
D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
0	1	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	$\overline{SAT}$	SAT

Table 21: TCMI msb/lsb split mode ESM data mapping

1st Byte: LSByte								2nd Byte: MSByte							
D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
b5	b4	b3	b2	b1	b0	$\overline{SAT}$	SAT	0	1	b11	b10	b9	b8	b7	b6

Table 22: TCMI lsb/msb split mode ESM data mapping

Byte							
D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
b7	b6	b5	b4	b3	b2	b1	b0

Table 23: TCMI 8-bit mode ESM data mapping

5.9.5. Detailed TCMI timing

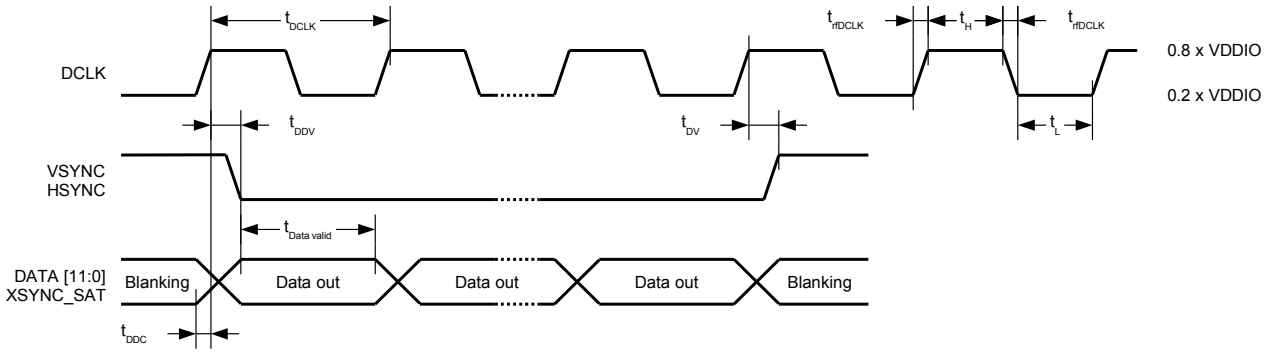
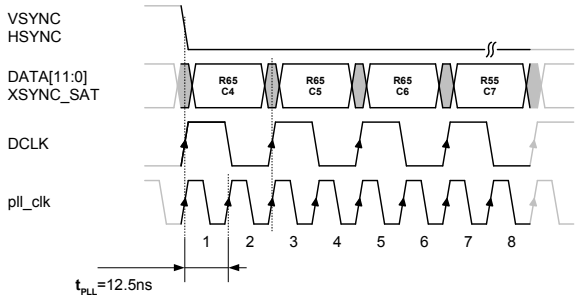


Figure 40: Detailed TCMI timing

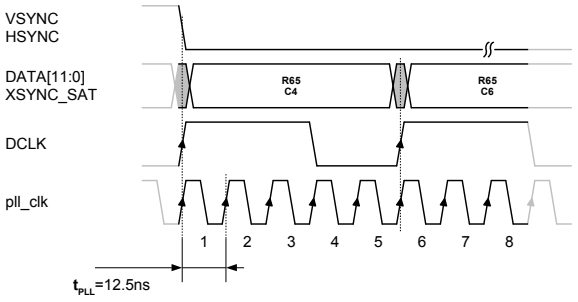
Symbol	Parameter	Min.	Typ.	Max.	Units
t_{DCLK}	TCMI readout clock: typ. f_{DCLK} = 20MHz / max. f_{DCLK} = 80MHz [1]		50	12.5	ns
t_{DDV}	Delay time after positive edge of DCLK until data are valid			2.0	ns
t_{DDC}	Data start changing before positive edge of DCLK			1.7	ns
t_{rfDCLK}	Rise and fall time of DCLK, VSYNC, HSYNC, XSYNC, Data[11:0]			2.0	ns
t_H	High period of DCLK [1]	5.0			ns
t_L	Low period of DCLK [1]	3.5			ns
$t_{Data\ valid}$	Output data on the TCMI interface are valid (depends on DCLK)	8.8			ns

Table 24: TCMI timing parameters (C_L = 20 pF max.)

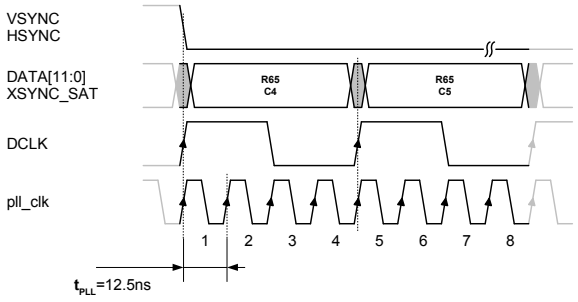
Note 1: Is DCLK >40MHz, t_H and t_L value need to be reduced by 2.5ns for covering safely delay and jitter effects of this output.



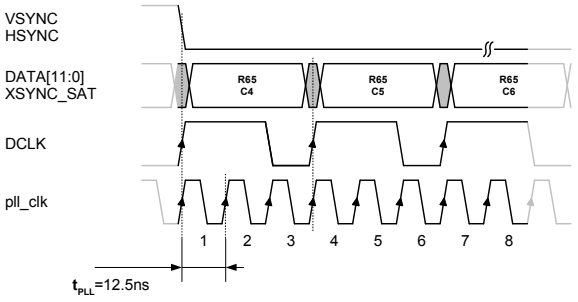
TCMI detailed bus timing: DCLK=40MHz (pll_clk / 2)



TCMI detailed bus timing: DCLK=16MHz (pll_clk / 5)



TCMI detailed bus timing: DCLK=20MHz (default, pll_clk / 4)



TCMI detailed bus timing: DCLK=26.6MHz (pll_clk / 3)

Figure 41: TCMI timing examples with symmetric and asymmetric DCLK

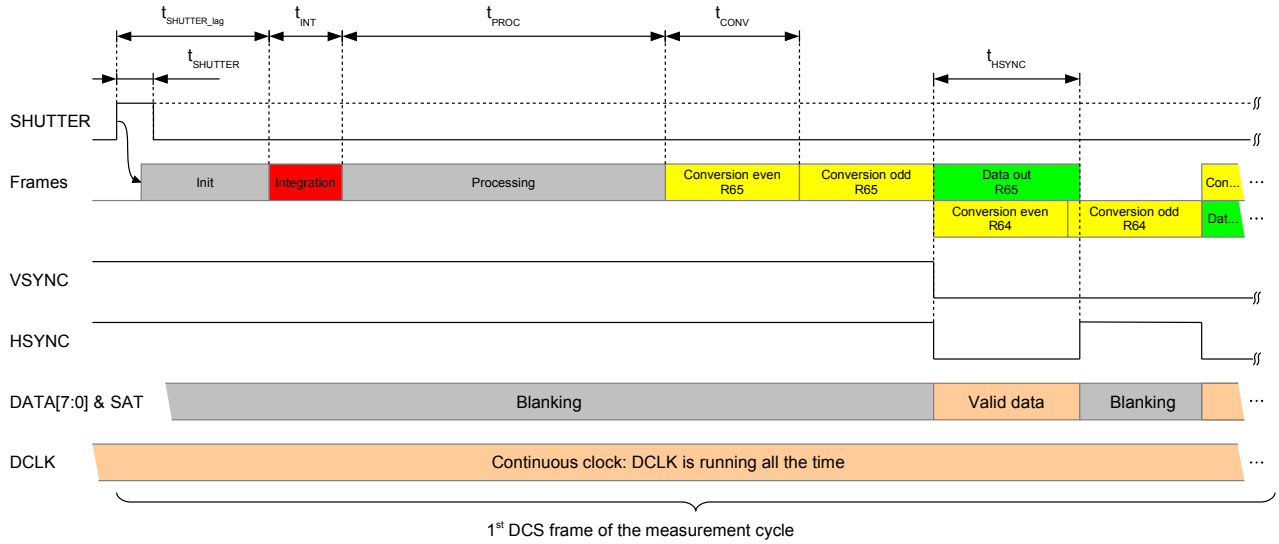


Figure 42: Frame timing: Start 1st DCS frame (DCLK: 20MHz)

Note:
To avoid readout rollover when using slower DCLKs e.g. DCLK < 5MHz with default ROI, pixel_seq_wait_row_done_en must be enabled (refer to chapter 8.7.7). It stretches HSYNC for slower TCMI interfaces. It causes a reduced DCS frame rate due to additional 2µs per ADC conversion (t_{conv} + 2µs).

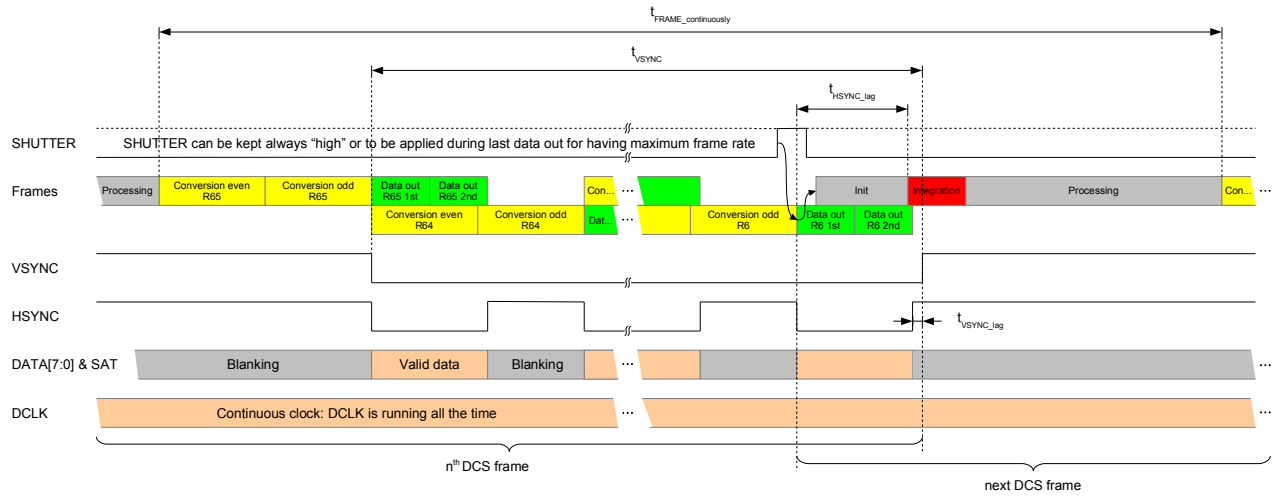


Figure 43: Frame timing: Inter frame timing, end of frame and start next frame (DCLK: 20MHz)

5.9.6. Frame rates and data-out performance

Default Frame and 3D TOF 4x DCS distance measurement

Frame rate:

The epc635 can perform a maximum of 512fps (frames per second) in any mode (1µs integration time, 80MHz mod_clk, 20MHz DCLK, DLL off, 4x DCS, continuous measurement control).

For 3D TOF, each frame is referred as a DCS frame. Either 4x (with π-delay matching) or 2x (without π-delay matching) DCS frames must be acquired for one distance calculation. Therefore, the resultant distance measurement rate turns out to be 128 fps or 256 fps respectively.

For the grayscale mode the maximum frame rate applies: 512 fps.

During the burst, each new sample is put on the DATA[7:0] TCMI bus together with XSYNC_SAT_CFG pin at every rising edge, in two consecutive DCLK cycles (e.g. 2 x 160 = 320 DCKL cycles at default f_{DCLK} = 20MHz). This gives a total burst duration time HSYNC of 32, 16 or 8µs at f_{DCLK} = 10, 20, 40MHz, respectively.

The data-out delay $t_{1st\_FRAME\_START}$ is defined as the time from the SHUTTER = 1 (or shutter_en = 1 via I^2C) to the first sample data coming out of the TCMI interface, refer to Figure 42 and Table 25. It is < 90µs for 1 µs integration time.

The time of one DCS frame is defined as the time from the SHUTTER = 1 (or shutter_en = 1 via I^2C) to the last sample data coming out of the TCMI interface.

Symbol	Parameter	Min.	Typ. 12 bit	Typ. 8bit	Units
t_{DCLK}	TCMI readout clock e.g. f_{DCLK} = 20MHz		50		ns
$t_{SHUTTER}$	Hold time for the signal on pin SHUTTER	250			ns
$t_{SHUTTER\_lag}$	Delay from the rising edge of SHUTTER signal to the 1st LED pulse		18		µs
t_{INT}	Image acquisition (integration time)				
t_{PROC}	Delay from the last LED pulse until the 1st row conversion		39.25		µs
t_{CONV}	Conversion time for a pair of half rows (even or odd)		15.75	12.25	µs
t_{HSYNC}	Readout time for a row e.g. f_{DCLK} = 20MHz		16	8	µs
t_{HSYNC_lag}	Delay from the begin of last readout until the 1st LED pulse of next DCS frame		17		µs
t_{VSYNC_lag}	Delay end of HSYNC to end of VSYNC at the end of each DCS frame		50		ns
t_{VSYNC}	Data readout time for one DCS frame e.g. f_{DCLK} = 20MHz t_{VSYNC} = (2x t_{CONV} x 59 rows) + t_{HSYNC} + t_{VSYNC_lag}		1874.55	1'453.55	µs
	Single measurement control mode:				
$t_{1st\_FRAME\_START}$	Delay from rising edge of SHUTTER signal until start of data readout of 1st frame		89.75	82.75	µs
$t_{1st\_FRAME\_TOTAL}$	Total time for reading one DCS or grayscale frame from rising edge of SHUTTER signal until end of readout of 1st frame		1964.30	1'536.10	µs
	Continuous measurement control mode:				
$t_{FRAME\_continuously}$	Total time for reading one DCS or grayscale frame $t_{FRAME\_continuously}$ = (2x t_{CONV} x 60 rows) + t_{HSYNC_lag} + t_{INT} + t_{PROC}		1'947.25	1'527.05	µs
$t_{4DCS\_continuously}$	Total time for one 3D TOF distance measurement (4 DCS) $t_{FRAME\_continuously}$ = ((2x t_{CONV} x 60 rows) + t_{HSYNC_lag} + t_{INT} + t_{PROC}) x 4 DCS		7.789	6.108	ms

Table 25: Timings for one DCS or grayscale frames and for 3D TOF distance measurements (4x DCS)
(Reference: see Figure 42 and Figure 43, f_{DCLK} = 20MHz, t_{INT} = 1µs)

Note:

t_{Init}: In frame cycles having a DLL synchronization t_{Init} = 37µs + t_{SYNC}. = 37µs + DLL pre-synchronization + DLL lock time e.g. t_{Init} = 77µs.

Action	Comments	Value	Units
DCS & grayscale frame rate	1 DCS or a grayscale frame	513	fps
Distance measurement frame rate	4 DCS measurement	128	mps

Table 26: Frame rates based on the Table 25 timing
(Continuous measurement control mode)

Memory space estimation:

The data-out operation transfers effectively 160x60pixel x 13bits/pixel (12 bit DATA + 1 bit SAT) = 124'800 bits (125kBit) pixel data with each frame via the TCMI interface.

In a typical application with a CPU having a high-speed parallel interface (GPIO, CPI, CMI, etc.), each 13 bit sample is packed into a 2-byte word, then stored in the application's frame buffer (internal SDRAM of the CPU or external SDRAM bridged via CPU). In that case, a burst (1 row) makes 320 bytes. The frame acquisition including all rows/columns requires a storage space maximum of 19'200 bytes or around 19kbyte per frame (1 kbyte = 1'024 bytes). For the 3D TOF distance measurements with 4 DCS, the application frame buffer must be large enough to accommodate at least 4 DCS frames plus (may be) another 1 DCS frame size to store back the calculated distance data. For a smooth SW programming, doubling this space is highly recommended as a bare minimum for the application frame buffer size. This corresponds to a total of 2 x (4 + 1) x 19 kbytes = 190kbytes memory space. In practice, this can be realized with a single 256 kByte SDRAM. Depending on the complexity of the application SW, this size must be increased.

5.9.7. Example applications of CPU architectures for high-speed frame data transfer

Several possible application CPU, DSP, FPGA configurations can be used with the epc635. Here are two architectures proposed as examples, to give the application developer a high-level overview.

The first one is a completely symmetric, non-blocking application frame buffer CPU architecture with a high-throughput imager to application frame buffer data transfer concept. While the frame is continuously acquired, the current frame data is streamed from the epc635 to the application frame buffer A. At the same time,the CPU can access and calculate the previously acquired data from the application frame buffer B. The next frame, the memory channels are dynamically swapped between epc635 and the CPU by re-configuring the DMA channels. As there are two physically identical, separated memory channels, both transfers can flawlessly happen at the same time (see Figure 44).

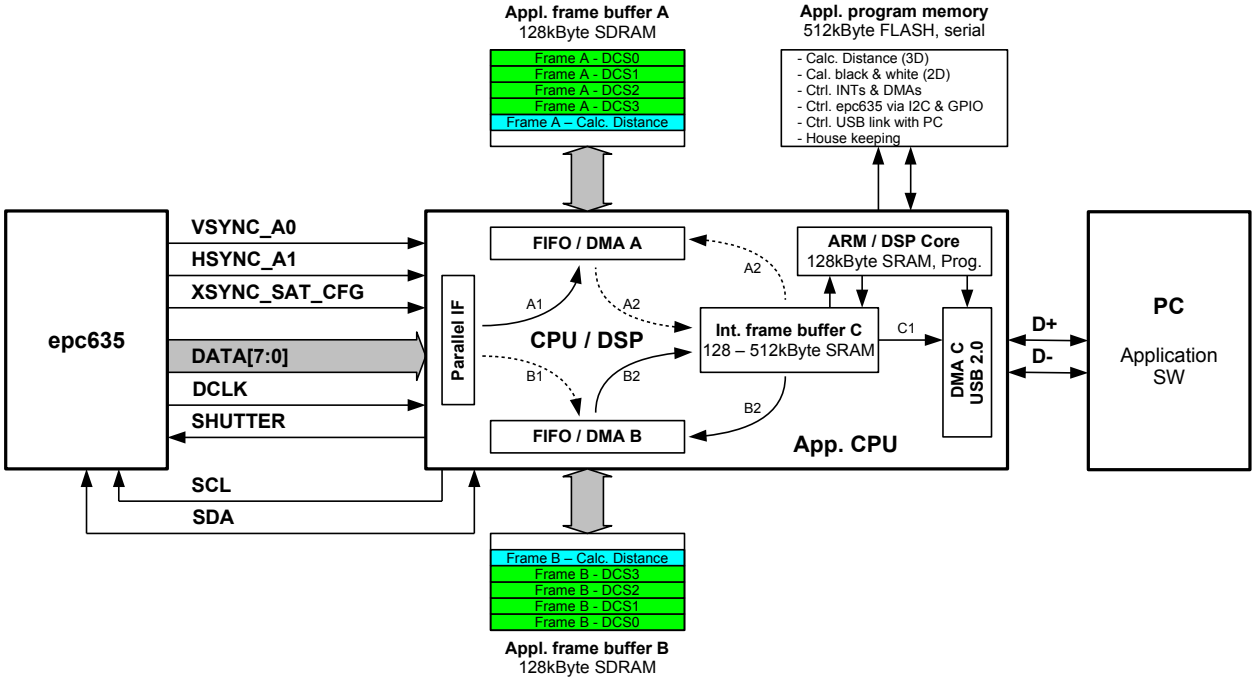


Figure 44: Symmetric, non-blocking application frame buffer CPU architecture

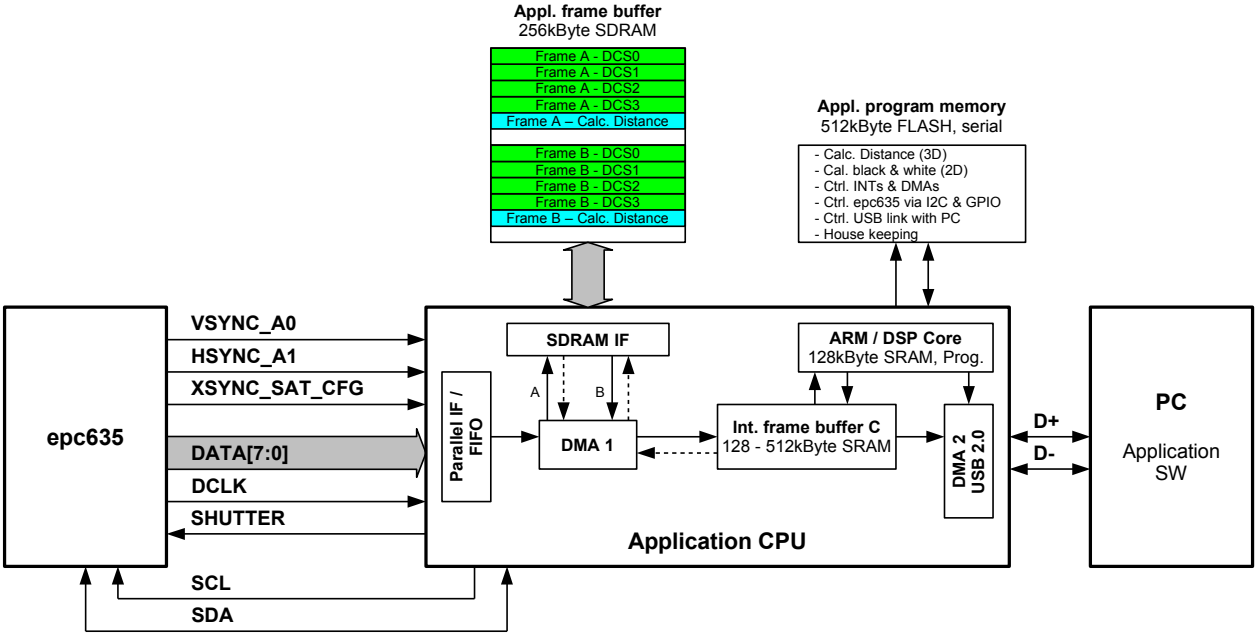


Figure 45: Time-shared application frame buffer CPU architecture

The second example is a time-shared application frame buffer CPU architecture. The epc635 must have a higher priority over CPU (see Figure 45) when accessing the application frame buffer SDRAM. The memory is blocked to the CPU as long as epc635 is transferring the burst. At the end of the burst, the memory channel is released by the DMA controller and the CPU gets access granted. In the time-shared bus architecture, the application developer must carefully tune the bus access ratio between the epc635 and the CPU. This can be adjusted by setting a different TCMI DCLK frequency (5, 10 or 20MHz) on the epc635 side. The higher the TCMI bus frequency, the faster the burst is transmitted (min. burst duration for full pixel rows: 8µs at f_{DCLK}= 20MHz).

Another option is using a CPU having a much wider and faster SDRAM interface. Example:

Assume using a CPU which has a 128 bit wide SDRAM bus running at 160MHz clock. The TCMI DCLK is set at 10MHz. This gives 4x performance increase between FIFO → application SDRAM due to the TCMI to SDRAM bus clock ratio. Another 8x increase is due to 16-bits packed samples in the FIFO and transferred over a 128 bit wide bus to the SDRAM. It results in a total of 32x faster transfers between FIFO → application SDRAM, application SDRAM → internal SRAM of the CPU and vice-versa. This gives a lot of bandwidth to the CPU to transfer data back and forth between the application SDRAM and the internal SRAM between computations.

5.10. Power consumption levels

The epc635 has mainly 7 power states/levels during the different operation phases. RESET state is the lowest, INTEGRATE is the highest average power consumption level.

Power state	Power [mW]	Operation description
RESET	TBD	All supplies are ON, $\overline{\text{RESET}}$ = 0, Oscillator is ON, PLL and all system system clocks are OFF
READY	170	$\overline{\text{RESET}}$ = 1, PLL and all system clocks ON, waiting for SHUTTER
INTEGRATE	270	SHUTTER = 1 or shutter_en= 1 via I²C, integrating
CONVERSION	TBD	Integration finished, conversion of rows
CONVERSION + DATAOUT	TBD	Transmit row data via TCMI while converting next row
DATAOUT	244	Transmit last row data via TCMI

Table 27: Typical average power consumption levels at different operating states (integration time < 5ms)

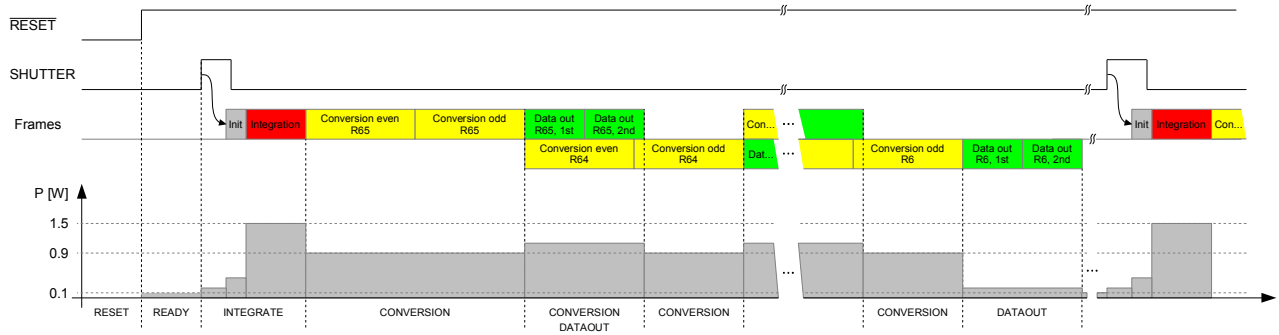


Figure 46: Power consumption levels and operating states

6. Measurement Modes

6.1. Distance measurement modes (3D TOF)

The distance measurement modes use the on chip LED driver and the external IR-LED/LD to provide modulated light on the target. Modulation control signals to the LED driver are provided by the programmable modulator (see chapter 6.1.1.).

6.1.1. Modulator

The modulator (see Figure 26) generates all signals to modulate the external IR-LED/LD via the LED driver and simultaneously all demodulation signals to the pixel-field for performing a measurement. Sine and grayscale mode with all the variants are generated here.

The mode selection takes place by the modulation table registers (refer to Table 40). More details to these modes are listed in following the chapters and in Table 32. The registers can be updated via I²C bus between the frame acquisitions. The application must take care that the last frame's integration phase is completed before modifying these registers on the fly. This time can be detected by the application by waiting for the falling-edge of VSYNC or the first falling-edge of HSYNC signal after SHUTTER = 1 was applied. This allows to run continuously at the maximum frame rate. For a full-frame readout, the margin is a 3.6ms to alter these registers via I²C on the fly. This margin scales linearly with the number of row readouts.

6.1.2. Sine mode (sinusoidal modulation)

The epc635's default modulation mode is the sinusoidal modulation. After reset, all internal register values are default to operate the chip: at 4MHz XTAL/external clock input, multiplied up to 80MHz at the PLL output, clocks the modulator with 80MHz internal mod_clk, modulates LED/LD with 20MHz and acquires 4 successive DCS frames (0 ... 3) using 25.6µs integration time.

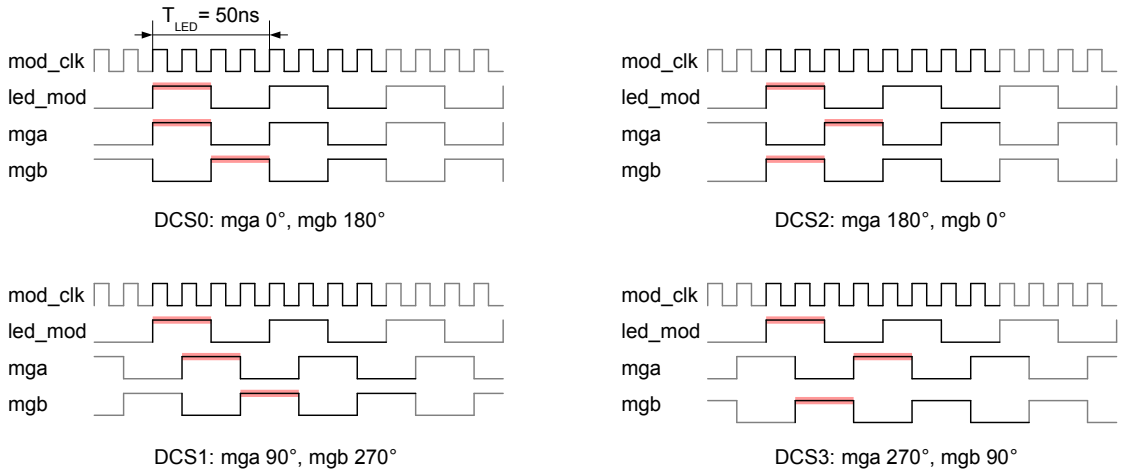
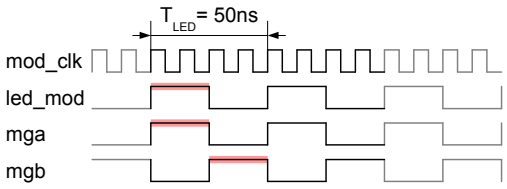


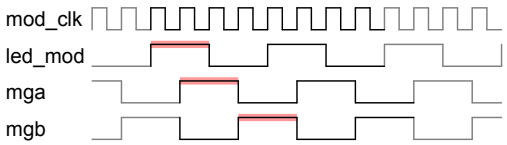
Figure 47: Sine mode 4x DCS modulation/demodulation waveforms (mod_clk = 80MHz)

With the application of the shutter trigger pulse (HW SHUTTER = 1 or SW shutter_en = 1 via I²C), the chip performs 4 successive DCS (Difference Correlation Sample) frame acquisitions. Each one of the 4 DCS frame types has a different phase relation between modulation (led_mod) and demodulation (mga, mgb) signals which makes phase-to-distance calculation possible. In case of DCS0, led_mod is phase-shifted by 0º and 180º with respect to mga and mgb, respectively. In case of DCS1, led_mod is phase-shifted by 90º and 270º. For DCS2, the phase shifts are 180º and 0º and for DCS4, the phase shifts are 270º and 90º (see Figure 47). Note that for DCS2 and DCS3, the demodulation signals mga and mgb are simply swapped with respect to DCS0 and DCS1, respectively.

By programming dcs_sel = 01 (see MOD_Control register in Table 41, Modulator/demodulator registers), shutter trigger initiates 2 successive DCS frame acquisitions (see Figure 48).



DCS0: mga 0°, mgb 180°



DCS1: mga 90°, mgb 270°

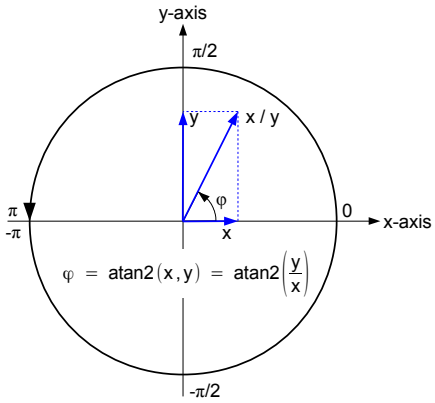
Figure 48: Sine mode 2x DCS modulation/demodulation waveforms
(mod_clk = 80MHz)

The first sample pair (DCS0 and DCS1) is enough to calculate 3D TOF distance for that pixel. The second pair (DCS2 and DCS3) can be used for calculating the π-delay matching algorithm to compensate for fixed pattern noise and device mismatches during pixel readout operation. The application SW must run the same algorithm on every pixel to compute the 3D TOF distance for the entire pixel-field.

Sine mode: Distance calculation algorithm

The use of the atan definition of the Cartesian coordinate system (atan2) guarantees a continuous distance calculation algorithm in the range of phases between -π … +π. In our case, we use the range from 0°… 360° which corresponds to the distance from 0m up to the unambiguity distance.

Definition atan2 for the range -π … π



$$\text{atan2}(x,y) := \begin{cases} \text{atan}(y/x) & \text{for } x > 0 \\ \text{atan}(y/x) + \pi & \text{for } x < 0,\ y \geq 0 \\ \text{atan}(y/x) - \pi & \text{for } x < 0,\ y < 0 \\ +\pi/2 & \text{for } x = 0,\ y > 0 \\ -\pi/2 & \text{for } x = 0,\ y < 0 \\ 0 & \text{for } x = 0,\ y = 0 \end{cases}$$

Note: Variations in notation

$\varphi = \text{atan2}(x,y)$ or $\text{atan2}(y,x)$

The arguments are reversed depending on implementation.

Figure 49: Continuous atan representation for the range -π … +π

Let the respective 12 bit samples measured for a single pixel from 4 DCS (0, 1, 2, 3) frames be DCS0, DCS1, DCS2 and DCS3.
Thereof, the sampling of 4x DCS samples corresponds to the following sinusoidal signal representation for the epc635 chip:

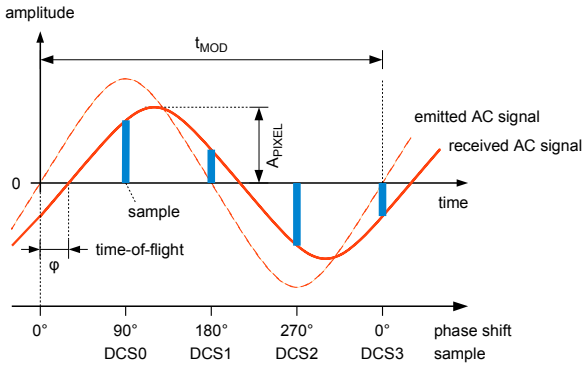


Figure 50: Sampling of the received waveform

The following terms are used for the formulas:

$D_{TOF\ sine}$	Distance in meters [m]
c	Speed of light e.g. 300'000'000m/s
f_{LED}	LED/LD modulation frequency e.g. 20MHz
DCS0 - DCS3	Sampling amplitude [LSB]
φ	Phase shift caused by the time-of-flight [rad]
D_{OFFSET}	Offset compensation [m]

In the real case of 4x DCS samples used with π-delay matching algorithm, the time of flight calculation will be:

[1] $$t_{TOF\ sine}[sec] = \frac{1}{2\pi f_{LED}} \cdot [\pi + atan2(DCS2-DCS0,\ DCS3-DCS1)] + t_{OFFSET}$$

then, the distance calculation will be (c is the speed of light):

[2] $$D_{TOF\ sine}[m] = \frac{c}{2} \cdot \frac{1}{2\pi f_{LED}} \cdot [\pi + atan2(DCS2-DCS0,\ DCS3-DCS1)] + D_{OFFSET}$$

2π phase wrap-around

If $D_{TOF\ sine} > D_{Unambiguity}$: $D_{TOF} = D_{TOF\ sine} - D_{Unambiguity}$,

if $D_{TOF\ sine} < D_{Unambiguity}$: $D_{TOF} = D_{TOF\ sine} + D_{Unambiguity}$,

else: $D_{TOF} = D_{TOF\ sine}$

$$D_{Unambiguity}[m] = \frac{c}{2} \cdot \frac{1}{f_{LED}}$$

Based on the ideal case of 2x DCS samples (without any offsets), the time of flight may be computed as follows:

[3] $$t_{TOF\ sine}[sec] = \frac{1}{2\pi f_{LED}} \cdot [\pi + atan2(-DCS0,\ -DCS1)]$$

then, the distance calculation is:

[4] $$D_{TOF\ sine}[m] = \frac{c}{2} \cdot \frac{1}{2\pi f_{LED}} \cdot [\pi + atan2(-DCS0,\ -DCS1)]$$

Sine mode: Quality of the measurement result

The epc635 provides information on the quality and the validity of the received optical signal. This reflects the confidence level of the measurement result. The better the received signal, the better and more precise the distance measurement will be.

Each distance measurement of every pixel has its own validity and quality.

The primary quality indicator for the measured distance data is the peak-to-peak amplitude value of the received modulated light $A_{TOF\ sine\ PP}$. The amplitude is in direct relation ship to the distance noise: Refer to Figure 5.

After each measurement, this needs to be calculated from the DCSx values delivered by the chip. This amplitude value is the feedback parameter that is used to set the integration time for the next measurement.

[5] $$A_{TOFsinePP} = \sqrt{\frac{(DCS2-DCS0)^2}{4} + \frac{(DCS3-DCS1)^2}{4}}$$

Amplitude $A_{TOF\ sine\ PP}$	Classification	Action
< 25 LSB	Weak illumination	Too less signal for an accurate measurement: Increase integration time for the next measurement
25 … 100 LSB	Enough signal for a useful measurement	Distance noise approx. 4 times higher than the optimum: No action necessary. See note below
100 … 2'000 LSB	Good signal strength	No action necessary. See note below
> 2'000 LSB	Overexposed	Decrease integration time for the next measurement. See note below

Table 28: Signal amplitude versus classification

Note:
Generally, the higher the received signal, the better and more precise the distance measurement will be. However, it is good practice to control the integration time such that an amplitude value between 200 ... 1'000 LSB is achieved. Higher values will only slow down the acquisition rate due to longer integration times, but are not significantly improving signal to noise ratio.

The quality indicator for the distance noise is the ratio of ambient-light E_{BW} to the peak-to-peak value of modulated light $E_{TOF\ PP}$ (AMR). This value may be calculated and used additionally to the above amplitude value if the respective application is subject to intense ambient light.

The peak-to-peak irradiance $E_{TOF\ sine\ PP}$ of the modulated signal at the surface of a pixel can be calculate out of the AC sensitivity $S_{TOF\ sine}$, the used integration time $t_{INT\text{-}TOF}$, the reference integration time $t_{INT\text{-}REF\text{-}TOF}$ and the peak-to-peak amplitude $A_{TOF\ sine\ PP}$ of the received modulated signal the following way:

$$[6] \qquad E_{TOF\ sine\ PP} = S_{TOF\ sine} \cdot \frac{t_{INT\text{-}REF\text{-}TOF}}{t_{INT\text{-}TOF}} \cdot A_{TOF\ sine\ PP} \quad \text{e.g.} \quad E_{TOF\ sine\ PP} = 0.62\,\frac{nW/mm^2}{LSB} \cdot \frac{103\,\mu s}{205\,\mu s} \cdot 1'000\ LSB = 0.31\,\mu W/mm^2$$

The formula to calculate the quality indicator “Ratio of ambient-light / modulated light” (AMR) is

$$[7] \qquad AMR[dB] = 20 \cdot \log\left(\frac{E_{BW}}{E_{TOF\,PP}}\right) \qquad \text{e.g.} \quad AMR[dB] = 20 \cdot \log\left(\frac{15.8\,\mu W/mm^2}{0.31\,\mu W/mm^2}\right) = 34dB$$

Refer for E_{BW} to chapter 6.2. Grayscale mode.

This ratio is one of the influencing factors regarding the distance noise.

AMR value	Classification	Action
< 60 dB	excellent	No action necessary.
< 70 dB	sufficient	Is a lower noise level needed, do the next measurement with a longer integration time or with an increased illumination power.
> 70 dB	weak	Do the next measurement with a longer integration time or with an increased illumination power.

Table 29: Classification ratio ambient-light to modulated light (AMR) versus distance noise

There are also validity indicators delivered by the chip after a measurement. These will help to detect saturated or not illuminated pixels as a result of too much/less illumination or too long/short integration time.

Validity indicator (per pixel)	Classification	Action
Saturation flag: = set	Pixel saturation: The pixel receives too much light (too much modulated light-signal, too reflective object or too much ambient-light) Refer to chapter 5.7.3. Pixel saturation detection and pin XSYNC_SAT_CFG.	Dump data and repeat the measurement with a decreased integration time or measure the ambient-light.
one or more of the DCSx values: | DCSx | > 2'000 LSB	DCSx value over maximum signal limit: The object is too reflective(It is too close to the sensor, there is too bright illumination or too much modulation signal is emitted)	Dump data and repeat the measurement with a decreased integration time.
all DCSx values: | DCSx | ≤ 25 LSB	DCSx values below noise level limit: Absence of an object in the operating range. The object has too little remission (reflectivity): It is too far away, the illumination is too dim or too little modulation signal is emitted)	Dump data and repeat the measurement with an increased integration time.

Table 30: Validity indicator versus classification

Table 31 shows a quality decision matrix as a summary of the validity and quality parameters for the distance measurement.

Step	Sensor status	Pixel saturation XSYNC_ SAT	Maximum signal min. one | DCSx |	Noise level all | DCSx |	Modulated light amplitude $A_{TOF\ PP}$	Ratio ambient to modulated light AMR	Action
1	Saturation or bright object within scene	flag set					Repeat measurement with decreased integration time and/or illumination
2	Saturation or bright object within scene	no	> 2'000 LSB				Repeat measurement with decreased integration time and/or illumination
4	No object detected	no	< 2'000 LSB	< 25 LSB			Repeat measurement with increased integration time or illumination
3	Overexposure or bright object within scene	no	< 2'000 LSB	> 25 LSB	> 2'000 LSB		Repeat measurement with decreased integration time or illumination
4	No object detected	no	< 2'000 LSB	> 25 LSB	< 25 LSB		Repeat measurement with increased integration time or illumination
5	Too much ambient-light	no	< 2'000 LSB	> 25 LSB	25 LSB … 2'000 LSB	> 60 db (or > 70 dB)	Repeat measurement with increased integration time or illumination
6	Object detected	no	< 2'000 LSB	> 25 LSB	Good: 100 … 2'000 LSB Enough: 25 … 100 LSB	< 60 db (or < 70 dB)	No action necessary

Table 31: Quality decision matrix

6.2. Grayscale mode

The grayscale mode allows using the epc635 as a grayscale imager. This mode can be used either without LED/LD illumination for ambient-light measurements or with LED/LD for active illumination of the scenery.

The grayscale measurement uses the regular DCS measurement but only with DCS0. It is performed in single-ended mode with readout of MGA. Refer to chapter 8.6.5, MT registers. Corresponding registers settings can be found in Table 32.

There will be a one frame acquisition with the 12 bit grayscale values.

Due to the fact that distance measurement results can be influenced by ambient-light, the grayscale measurement without illumination can thereof be used as an important quality and correction parameter for the distance measurement.

Grayscale: Quality of the measurement result

The same rules apply for data quality and validity as given in section “Sine mode: Quality of the measurement result”. The saturation flag status is invalid in this mode.

These grayscale values can also be used for estimating quality information during sine mode for the 3D TOF distance calculations.

The irradiance E_{BW} of the grayscale signal at the surface of a pixel can be calculate out of the DC sensitivity S_{BW}, the used integration time $t_{INT\text{-}BW}$, the reference integration time $t_{INT\text{-}REF\text{-}BW}$ and the amplitude DCS0 of the grayscale signal the following way:

$$[8] \qquad E_{BW} = S_{BW} \cdot \frac{t_{INT\text{-}REF\text{-}BW}}{t_{INT\text{-}BW}} \cdot DCS0 \qquad \text{e.g.} \quad E_{BW} = 0.246 \frac{nW/mm^2}{LSB} \cdot \frac{103\,\mu s}{1.6\,\mu s} \cdot 1'000\ LSB = 15.84\,\mu W/mm^2$$

7. Application information

As a help for the user to have an easier understanding of the chip, this chapter list a variety of typical application examples and their configurations for the epc635.

7.1. Example sequence from the start-up to frame acquisition

1. Apply all positive supplies, while keeping $\overline{\text{RESET}}$ = 0.
2. Wait until all positive supplies reach their rated levels.
3. Apply the V_{BS} negative supply, while keeping $\overline{\text{RESET}}$ = 0.
4. Wait until V_{BS} reached its rated level.
5. Optional: Set/check the external 10kOhm pull-up resistors on the strap pins (HSYNC_A1, VSYNC_A0).
6. Release $\overline{\text{RESET}}$ = 1.
7. Wait until the start-up/reset sequence is over (t_{Strap_scan} + $t_{EEPROM\_to\_CFG\_copy}$).
8. Optional: Program the TCMI interface signal polarities with respect to the application CPU interface requirements via I²C interface.
9. Do the counterpart for the parallel data interface settings on the application CPU.
10. Optional: Set LED/LED2 driver and DLL properties and polarities with respect to external LED/LD circuit on the PCB via I²C. Set DLL measurement rate via I²C (DLL_measurement_rate_hi/lo register).
11. Select the measurement mode out of the Sine or Grayscale modes (Default: Sine mode, 4 DCS).
12. Set the integration time via I²C (INTM_hi/lo, Int_len_hi/lo registers).
13. Start the frame acquisition by using shutter trigger signal (SHUTTER = 1 or shutter_en = 1 via I²C).
14. Receive transmitted frames from TCMI interface to the external frame buffer on the application CPU.
15. Optional: Read the temperature sensor via I²C (Sum_Temp_tl_hi/lo)
16. Loop back to step 12 - 16.

Note: For corresponding I²C communication examples refer to chapter 5.4.7., Control commands.

7.2. Basic measurement mode setting

The basic measurement functions, as they are different modes of distance measurements and grayscale imaging, are selected by the appropriate settings of the corresponding control registers. For the most often used applications, Table 32 lists these configurations.

Refer to the corresponding chapters for detailed explanations to each mode.

Mode	Register setting	
	MOD_Control (0x92)	MT_8_lo (0x3C)
Sine 4x DCS [1]	0x34 [1]	0x26 [1]
Sine 2x DCS	0x14	0x26
Ambient only [2]	0xC4	0x26
Ambient & non modulated LED/LD [3]	0xC4	0x16
Ambient & modulated LED/LD [4]	0xC4	0x06

Table 32: Basic measurement mode setting TOF & grayscale

Notes:

[1] Default setting

[2] Grayscale image passively illuminated by ambient-light only.

[3] Grayscale image passively illuminated by ambient-light and actively illuminated by non modulated LED/LD.

[4] Grayscale image passively illuminated by ambient-light and actively illuminated by modulated LED/LD.
Note: LED driver is always turned on. Take care that the LED driver and the epc635 chip does not exceed the maximum operating limits.

[5] For correct setting of DLL_measurement_rate, refer to chapter 5.6., DLL (Delay Locked Loop).

7.3. 3D TOF distance measurement flow

A final 3D TOF distance image will be done with different process steps according to Figure 51. Both interfaces of the epc635 are used: The I²C for configuration, mode selection and temperature reading (blue marked in the following figures) and the high-speed TCMI for reading the frame data (red marked in the following figures).

The sequence starts with the initialization of the epc635 registers with the necessary and correct configuration parameters. Next, the TOF measurement with the expected mode (sine 4x DCS, or 2x DCS) will be performed. Depending of the application and the ambient conditions (ambient-light, changing temperature conditions), the TOF measurement needs some compensation. For the purpose of more accurate ambient-light compensation, a grayscale measurement without illumination captures the background light level. Reading of the on-chip temperature sensor (from time to time) helps to compensate thermal influences caused by e.g. the LEDs, the optical filters and the epc660 chip. After the rearrangement of the grayscale image to the correct pixel orientation, the final 3D TOF distance image can be calculated with all for the application necessary compensations.

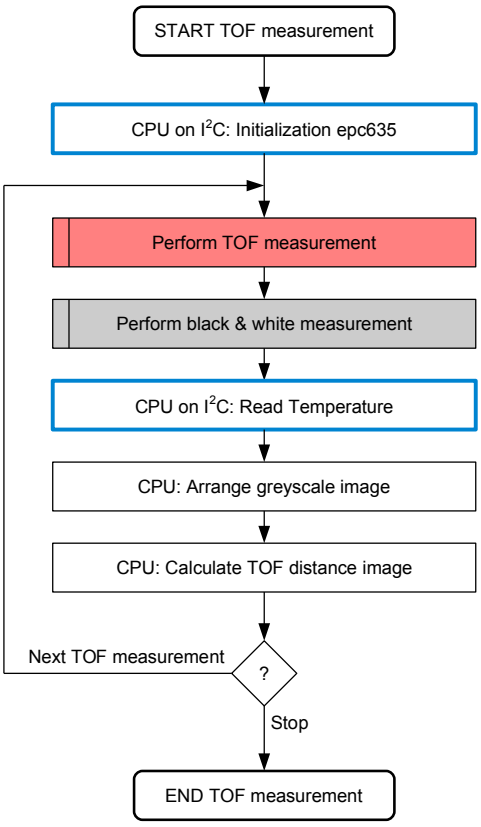


Figure 51: Generic 3D TOF distance measurement flow

The process flows for distance measurements and for grayscale images are similar, see Figure 52. The main differences are the mode selection, see Table 32 and depending thereof the number of frames, which need to be read out during a process cycle. After mode setting, the cycle will be started by applying the SHUTTER signal. Once the SHUTTER is stimulated, the epc635 executes the measurement until the end of the sequence automatically. The application CPU has to follow accordingly by reading out the collected data until the end of sequence.

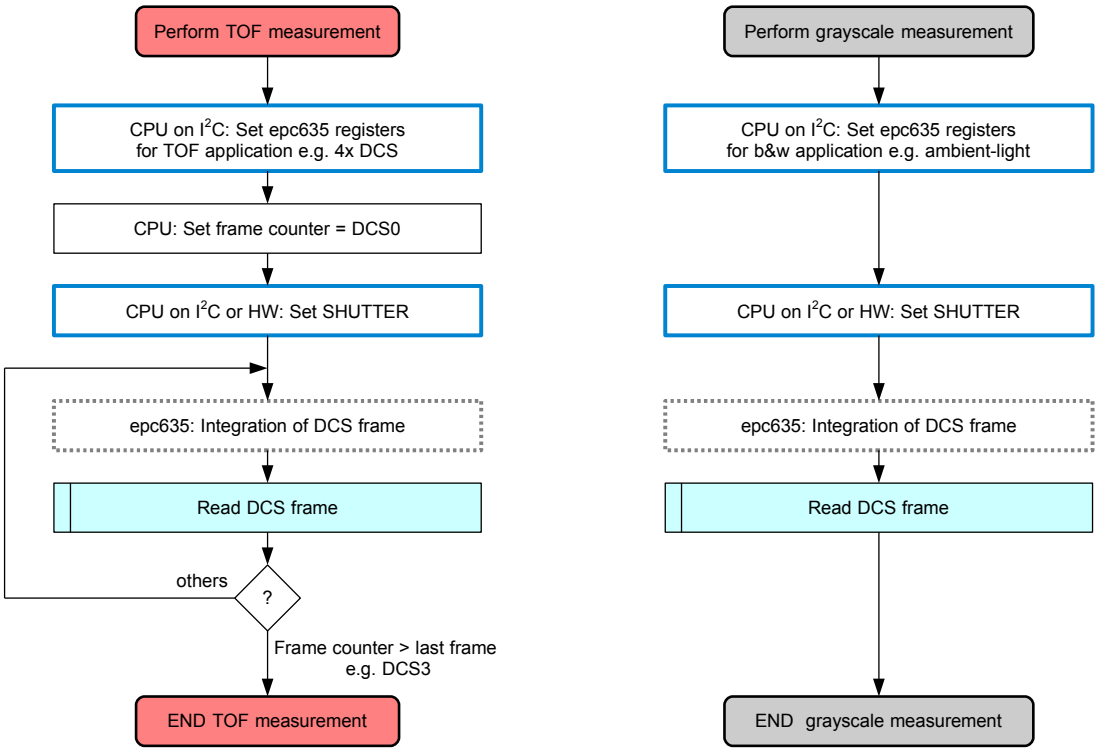


Figure 52: Generic sequences for the distance (TOF) and the grayscale measurement

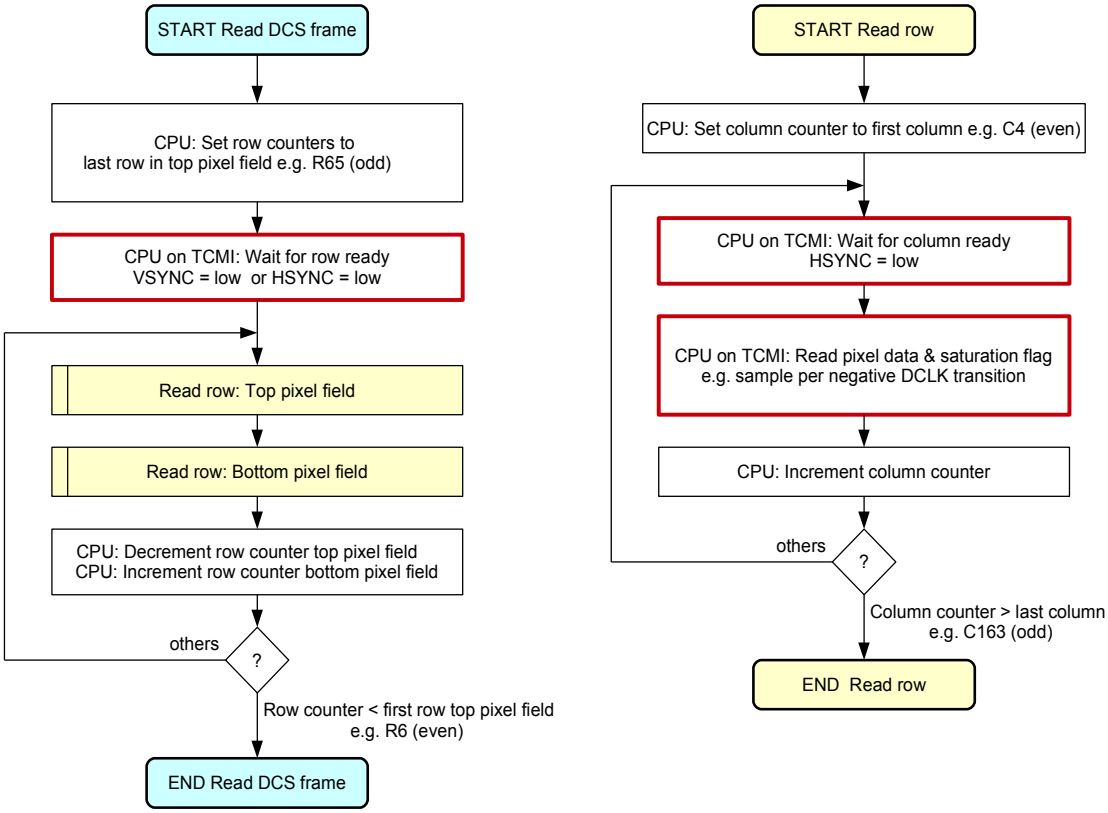


Figure 53: Generic sequences to readout frames and row by row

The generic procedures to readout frames or rows are independently of the selected modes. The application is driven only by the TCMI interface during these phases. To catch the begin of the frame, the application CPU has to wait after the measurement start until the integra-

tion period is finished and the first frame data is available. The epc635 signals this by setting VSYNC and HSYNC active. Pixel data can be read DCLK by DCLK as long the HSYNC signal is active. Refer also to Figure 42 and Figure 43. The application has to take care of its own, to update synchronously all necessary frame, row and pixel readout counters of the application during the the whole measurement cycle.

Only if the validity and the necessary quality level of the measured data are given, the result of the distance is correct and reliable. The generic validity and quality of the data is independent of any correction or compensation algorithms. It is given by the complete epc635 camera system with chip, lens, illumination and environmental conditions.

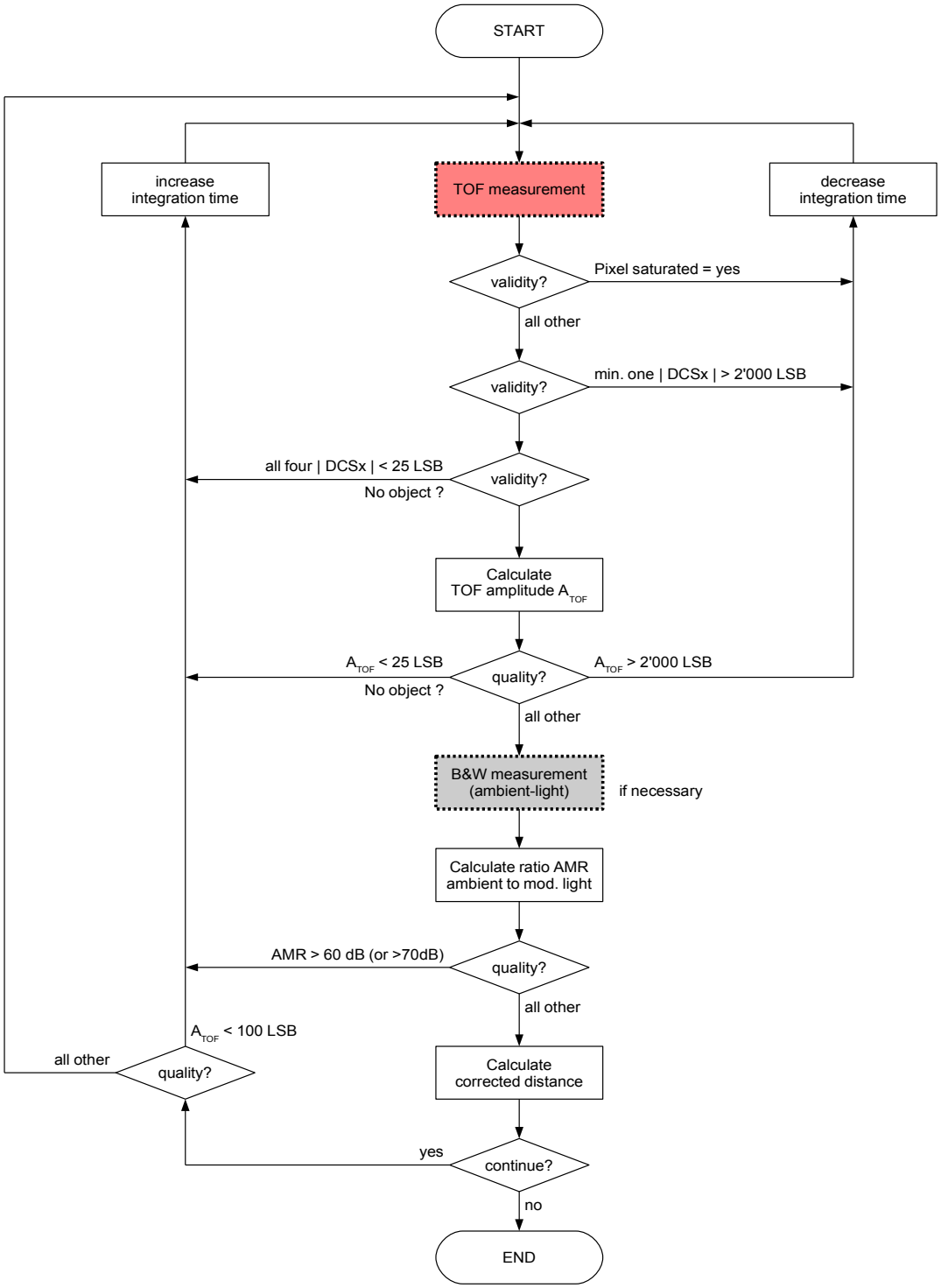


Figure 54: Generic validity and quality flow chart for a single pixel

Regarding possible correction and compensation algorithms refer to e.g. ESPROS' application note AN10 or the Handbook epc600 (AN07), which are available on the ESPROS website.

7.4. Operating and unambiguity range versus time base setting

The epc635 3D TOF imager uses the time-of-flight principle. It is implemented with a repeating, continuous-mode modulation signal during the integration phase (refer to chapter 6., Measurement Modes). Consequently, only signals returning within the maximum time slots can be detected unambiguously.

The operating range is the maximum distance which corresponds to the maximum time-of-flight inside of one period of the used modulation: for sine, it is one period of f_{LED}. Objects inside this area are detected unambiguously.

The unambiguity range defines the repetition distance, where objects outside of the targeted operating range can still be detected as far they are of very high reflectivity (remission). Strongly reflected signals outside of this range may therefore interfere with the measurement.

The operating range, the unambiguity distance, the time base for the integration time and the resolution of the distance signal are defined by the modulation clock mod_clk. This corresponds for the epc635 to a maximum default operating range of 7.5m @ mod_clk = 80MHz. It may be necessary depending on the application to adapt this parameters to other values. It can be done by a change of the mod_clk. Table 33 lists as an example some values of the modulation clock mod_clk in function of the operating range, the unambiguity distance, of the distance resolution and of the multiplier of the integration time base.

Operating range	Int. time multiplied by	Distance resolution [2]	Modulator clock	mod_clk _div	LED/LD mod. freq.	Unambiguity range
			mod_clk	(0x85)	f_{LED}	
[m]	[#]	[cm]	[MHz]	[#]	[MHz]	[m]
7.50	1	0.250	80.0	0	20.000	7.50
15.00 [1]	2 [1]	0.500	40.0	1 [1]	10.000	15.00
30.00	4	1.000	20.0	3	5.000	30.00
60.00	8	2.000	10.0	7	2.500	60.00
120.00	16	4.000	5.0	15	1.250	120.00
240.00	32	8.000	2.5	31	0.625	240.00

Table 33: Operating range versus mod_clk

Notes:

[1] Default values

[2] The distance resolution is given for an operating range corresponding to 3'000 LSB.

7.5. Integration time setting

The integration time is the active frame acquisition period (see Figure 42). Specially for moving objects or cameras, this time should be very short to reduce motion blur effects as much as possible. The integration time together with the illumination intensity also defines the effective achievable distance, which the complete epc635 camera system can see.

The following Table 34 lists the possible and mostly common used integration times per measurement mode as well as the corresponding register settings.

Integration time	Register values			
	INTM (0xA0/0xA1)		Int_len (0xA2/0xA3)	
	[DEC]	[HEX]	[DEC]	[HEX]
100 ns	1d	0x0001	7d	0x0007
200 ns	1d	0x0001	15d	0x000F
400 ns	1d	0x0001	31d	0x001F
800 ns	1d	0x0001	63d	0x003F
1.60 µs	1d	0x0001	127d	0x007F
3.20 µs	1d	0x0001	255d	0x00FF
6.40 µs	1d	0x0001	511d	0x01FF
12.8 µs [5]	1d	0x0001	1'023d	0x03FF
25.6 µs	1d	0x0001	2'047d	0x07FF
51.2 µs	1d	0x0001	4'095d	0x0FFF
102.4 µs	1d	0x0001	8'191d	0x1FFF
204.8 µs	1d	0x0001	16'383d	0x3FFF
409.60 µs	1d	0x0001	32'767d	0x7FFF
819.20 µs	1d	0x0001	65'535d	0xFFFF
1.6384 ms	2d	0x0002	65'535d	0xFFFF
3.2768 ms	4d	0x0004	65'535d	0xFFFF
6.5536 ms	8d	0x0008	65'535d	0xFFFF
13.1072 ms	16d	0x0010	65'535d	0xFFFF
26.2144 ms	32d	0x0020	65'535d	0xFFFF
52.4288 ms	64d	0x0040	65'535d	0xFFFF
104.8576 ms	128d	0x0080	65'535d	0xFFFF
209.7152 ms	256d	0x0100	65'535d	0xFFFF
419.4304 ms	512d	0x0200	65'535d	0xFFFF
838.8608 ms	1'024d	0x0400	65'535d	0xFFFF
1.6777216 s	2'048d	0x0800	65'535d	0xFFFF
3.3554432 s	4'096d	0x1000	65'535d	0xFFFF
6.7108864 s	8'192d	0x2000	65'535d	0xFFFF
13.4217728 s	16'384d	0x4000	65'535d	0xFFFF
26.8443545 s	32'768d	0x8000	65'535d	0xFFFF

Table 34: Typical TOF and grayscale integration time settings for mod_clk = 80MHz

Note:

[1] Integration time default setting is 12.8µs.

7.6. Special mode setting

In this chapter, the user will find the register setting tables for using special modes. Detailed descriptions are given in the corresponding chapters of these modes; see chapter 5.7., Pixel-field.

7.6.1. TOF dual DCS acquisition with phase-shifted integration time (motion blur reduction)

- This mode needs the following basic setting of the MT registers:
  MT_0_hi = 0x34, MT_1_hi = 0x3E.
- Reset the MT registers to the default values after leaving this mode:
  MT_0_hi = 0x30, MT_1_hi = 0x35.

Function	MOD_Control (0x92)
Sine 4x DCS	--- [1]
Sine 2x DCS [2]	0x14
Grayscale	--- [1]

Table 35: Basic measurement mode setting

Notes:

[1] Setting is not applicable for this application.

[2] Output is effectively 4x DCS in 2 DCS-frames.

7.6.2. TOF and grayscale single DCS acquisition with 2 different integration times (High dynamic range)

Mode	Register setting	
Function [1]	MOD_Control (0x92)	MT_8_lo (0x3C)
Sine 4x DCS	0x3C	0x26
Sine 2x DCS	0x1C	0x26
Ambient only [2]	0xCC	0x26
Ambient & non modulated LED/LD [3]	0xCC	0x16
Ambient & modulated LED/LD [4]	0xCC	0x06

Table 36: Measurement mode setting for high dynamic range TOF and grayscale

Notes:

[1] Output is effectively 2 DCS frames with different integration times in each 1 readout frame.

[2] Grayscale image passively illuminated by ambient-light only.

[3] Grayscale image passively illuminated by ambient-light and actively illuminated by non modulated LED/LD.

[4] Grayscale image passively illuminated by ambient-light and actively illuminated by modulated LED/LD.
Note: LED driver is always turned on. Take care that the LED driver and the epc635 chip does not exceed the maximum operating limits.

7.7. Power saving options

The epc635 has different power consumption depending on its activity level, see chapter 5.10., Power consumption levels. For power critical applications e.g. battery powered systems, it is possible to enforce the epc635 to go in so-called power saving states.

The different power saving modes and their description are listed in Table 37.

Operating mode	Power	Time		Description
		Shutdown	Recovery	
	[mW]	[ms]	[ms]	
INTEGRATION	270	TBD	TBD	Maximum average power consumption during regular 3D TOF operation
READY (Idle)	170	TBD	TBD	Minimum average power consumption during regular 3D TOF operation
SW POWER DOWN	TBD	TBD	TBD	Note: Only during regular READY mode possible. Power consumption during unnecessary functions where switched-off off by register commands.
HW POWER DOWN	TBD	TBD	TBD	Power consumption during Pin 55 $\overline{\text{RESET}}$ = 0.

Table 37: Power modes

Table 38 is a help for the electronic designer and list the current consumption versus the power saving level and per supply level.

Current	Description	Mode				Units
		INTEGRATION	READY	SW POWER DOWN	HW POWER DOWN	
I_{VDD}	Digital supply current	11	9	TBD	TBD	mA
I_{VDDPLL}	PLL supply current	6	6	TBD	TBD	mA
I_{VDDIO}	IO supply current	16	16	TBD	TBD	mA
I_{VDDA}	Analog supply current	33	17	TBD	TBD	mA
I_{VDDPXH}	Pixel analog 2 supply current	2	2	TBD	TBD	mA
I_{VBS}	Bias supply current	-2	-2	TBD	TBD	mA

Table 38: Current consumption during power modes

Is the epc635 during regular operation in the READY state, the SW POWER DOWN mode can be activated by the following command sequence listed Table 39.

No.	Register			Description
	Name	Address	Value	
1	**Regular 3D TOF operation**			
2	Power_Control (analog)	0xA5	0x00	Switch off of unnecessary supplies
3	CLK_enables	0x80	0x00	Switch off of unnecessary clocks
4	CFG_Mode_Control	0x7D	0x14	Switch system clock to XTAL clock
5	CFG_Mode_Control	0x7D	0x10	Switch off PLL
6	**SW POWER DOWN**			
7	CFG_Mode_Control	0x7D	0x14	Switch on PLL
8	**Wait > 32µs**			**Wait until PLL stable**
9	CFG_Mode_Control	0x7D	0x04	Switch system clock to PLL
10	CLK_enables	0x80	0x3F	Switch on the clocks again
11	Power_Control (analog)	0xA5	0x07	Switch on the supplies again
12	**Wait until supplies are stable**			
13	**Regular 3D TOF operation**			

Table 39: Sequence for the SW POWER DOWN mode

7.8. Rolling DCS frames

In special applications, it is possible to use all the time the same integration time in continuous distance measurement mode without any grayscale images for ambient-light compensation. Such a set-up allows enhancing the distance measurement rate by a factor of 4 by using rolling DCS frames.

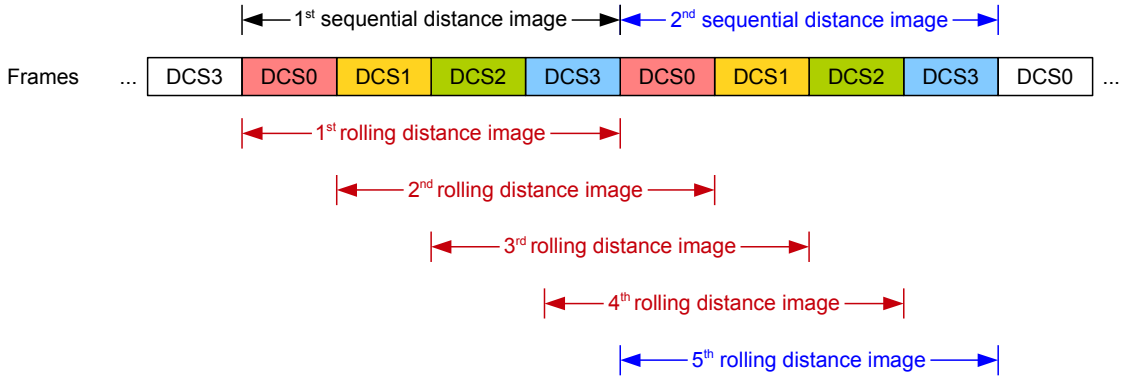


Figure 55: Rolling DCS frames

As shown in Figure 55, the algorithm performs with each new DCS frame a new distance calculation based on the new and last three DCS frames.

7.9. Enhanced rolling DCS frame mode

epc635 allows to set for each single DCS access own parameters. This opens also the possibility to acquire in time-sequence DCSx frames with e.g. different integration times for expanding the dynamic range. The example shown here is using two integration times: 50 µs for detecting short range objects and 2ms doing the same for the long range.

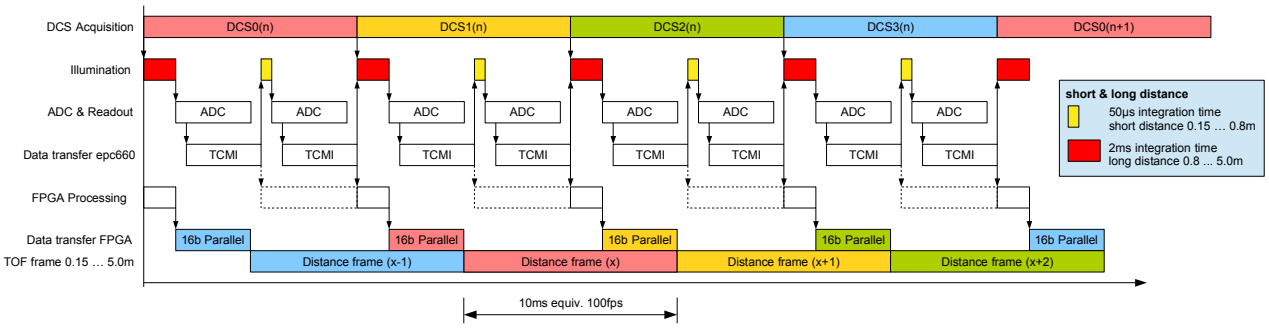


Figure 56: Enhanced rolling mode sequence

In such a case the application is responsible e.g. by FPGA to control the epc635 in the right manner and to decouple the data stream accordingly. The distance per pixel will be calculated out of the data with best exposure, means quality parameters matches best to the necessary quality conditions for good distance calculation. Figure 57 and Figure 58 show a corresponding block diagram and data flow.

The main advantage of such solution is the time equidistant acquisition of the DCSx data in a multi-integration-time (multi-exposure) mode.

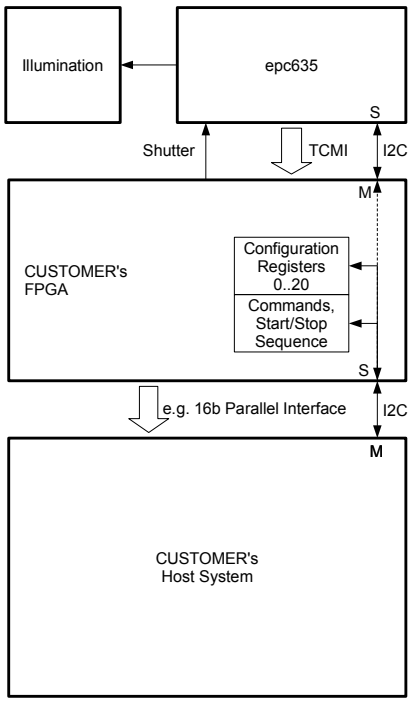


Figure 57: Block diagram

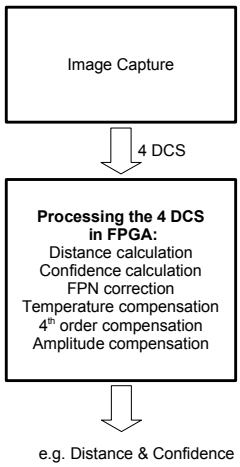


Figure 58: Data flow

7.10. External modulation MODCLK

The epc635 has for enhanced user applications also the possibility to bring an external modulation clock to the chip. The optional MODCLK input can be used for this to inject in parallel a user controlled/modulated clock for both the LED driver and the pixel demodulator, see Figure 59.

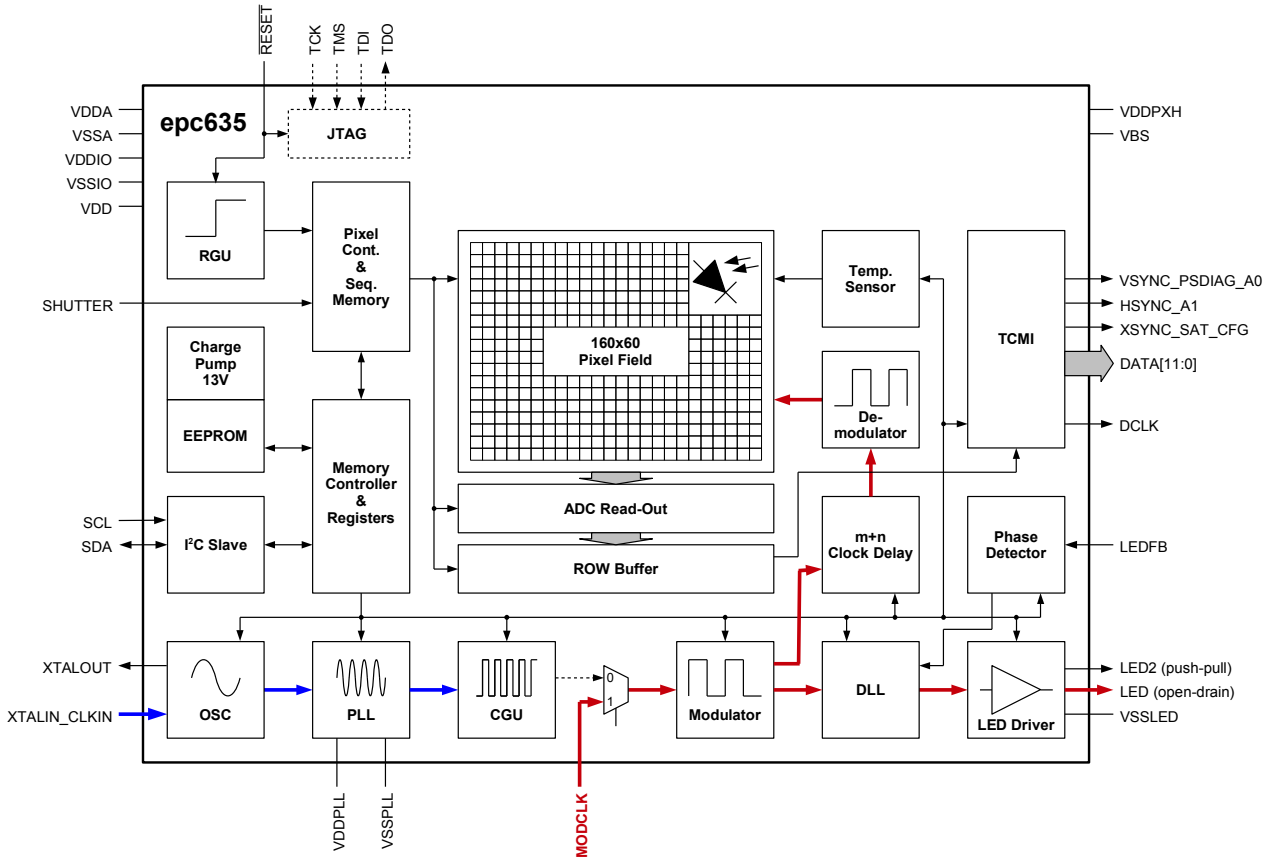


Figure 59: The MODCLK signal flow (red marked)

The external MODCLK can be used e.g. in concepts for reliable multi camera applications. It allows to use e.g. frequency-division multiple access (FDMA). In corresponding literature, the details of these concepts are explained in detail.

The user is free to apply any digital waveform up to 80MHz during frame acquisition as external MODCLK signal. Even more, he is also free to use modulations like pseudo-random edge jitter, dithering, etc.

The signal from the MODCLK pin is used as far in the register CLK_enables is set to mod_clk_sel = 1, see section 8.7.1.

IMPORTANT:

- The external modulation clock replaces the internal generated modulation clock, refer also to Table 33. Based on this clock, the modulator generates the modulation sequence according chapter 6., Measurement Modes and the setting in the MOD_Control register.
- Set DLL_measurement_rate = 0 for switching off the automatic DLL synchronization.

7.11. epc635 Card Edge Connector Carrier

7.11.1. Schematics

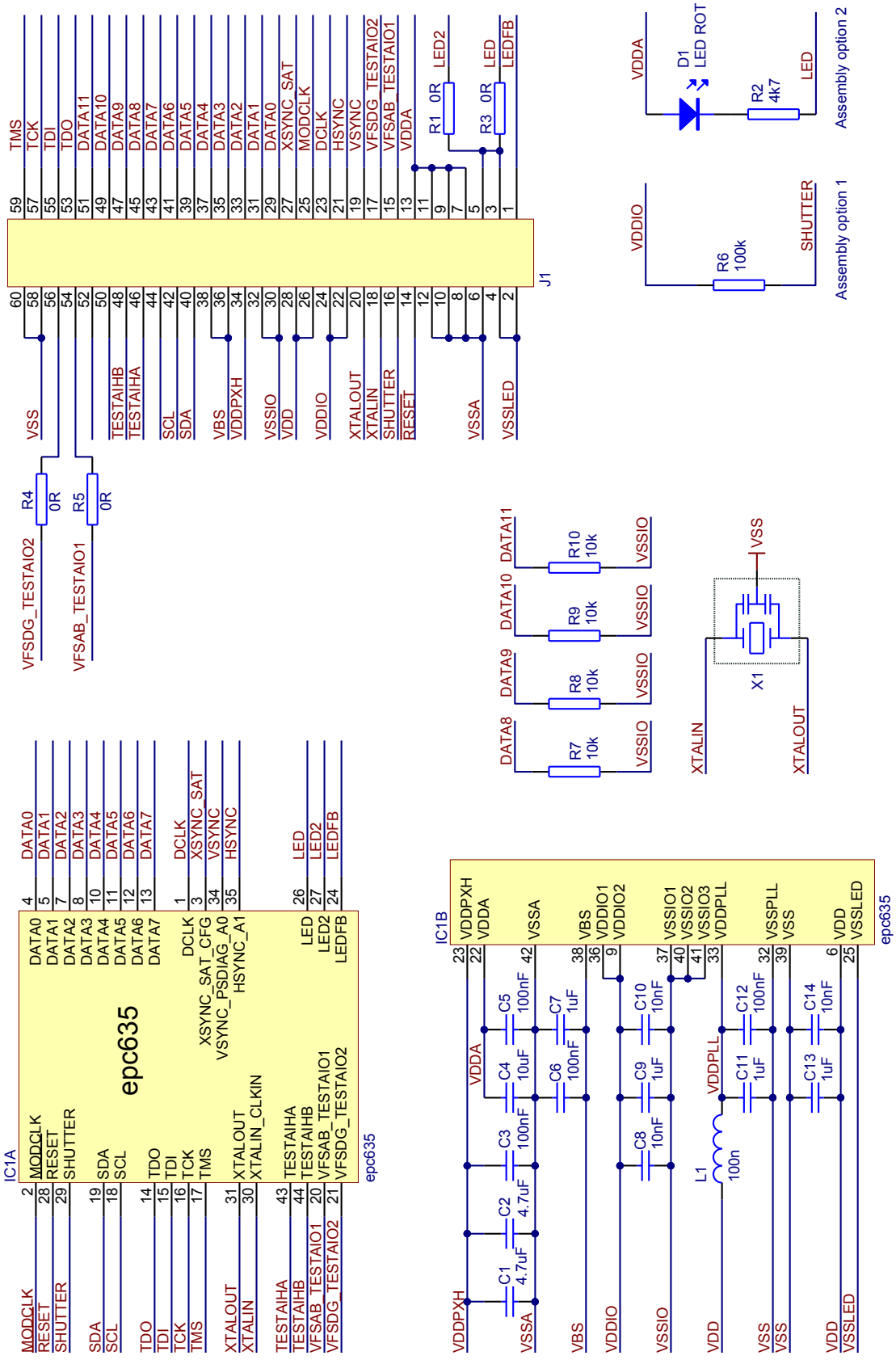


Figure 60: Schematic epc635 Card Edge Connector Carrier

7.11.2. Board layout and assembly

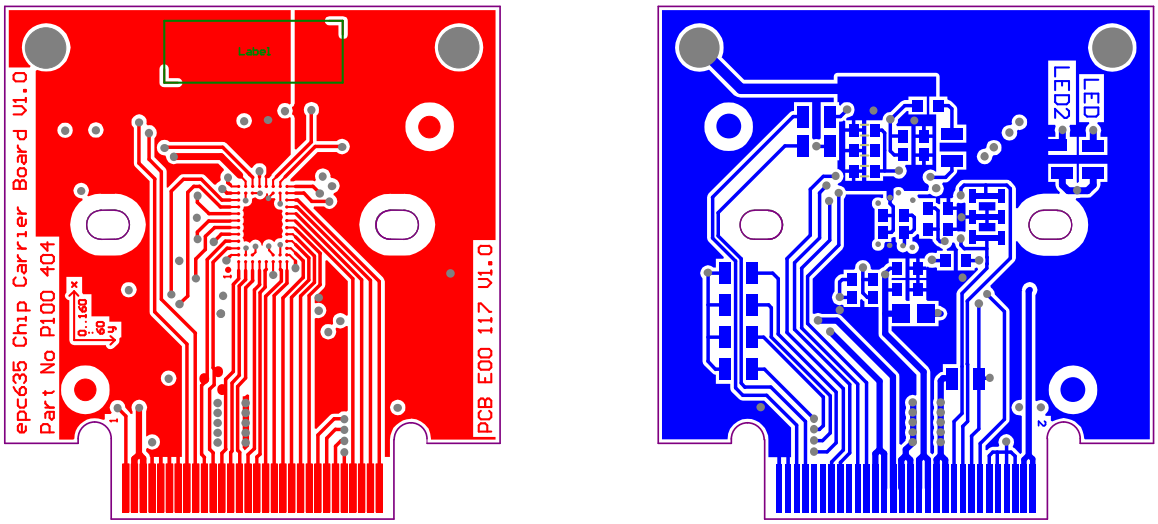


Figure 61: epc635 Card Edge Connector Carrier: Layout top and bottom

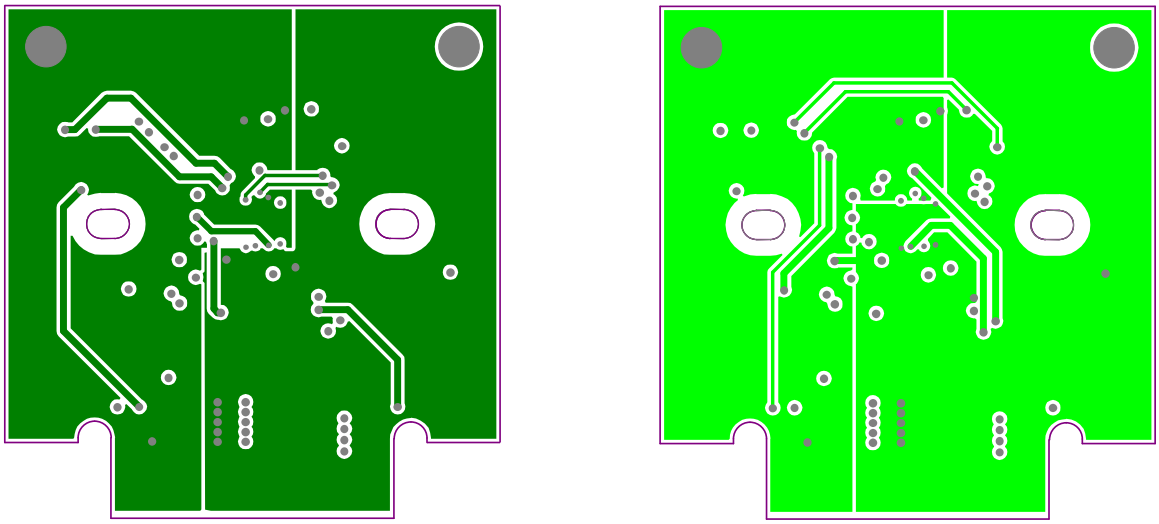


Figure 62: epc635 Card Edge Connector Carrier: Layout middle top and bottom

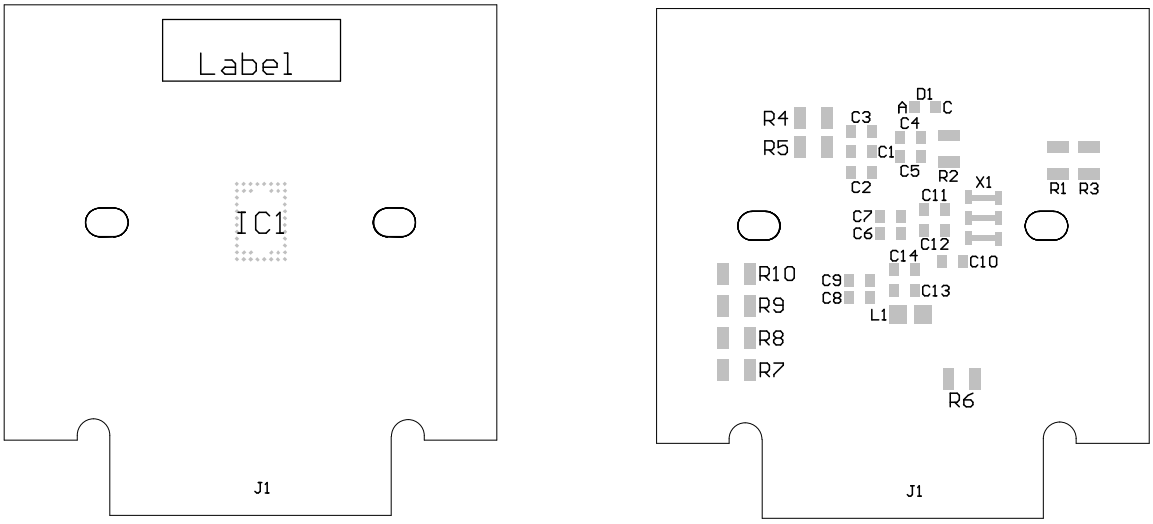


Figure 63: epc635 Card Edge Connector Carrier: Assembly top and bottom

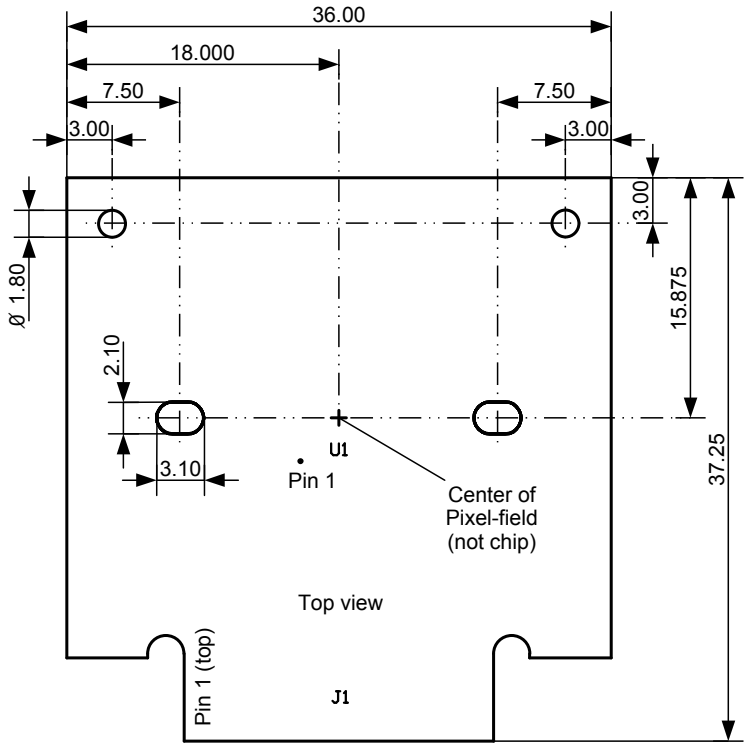


Figure 64: epc635 Card Edge Connector Carrier: Dimensions

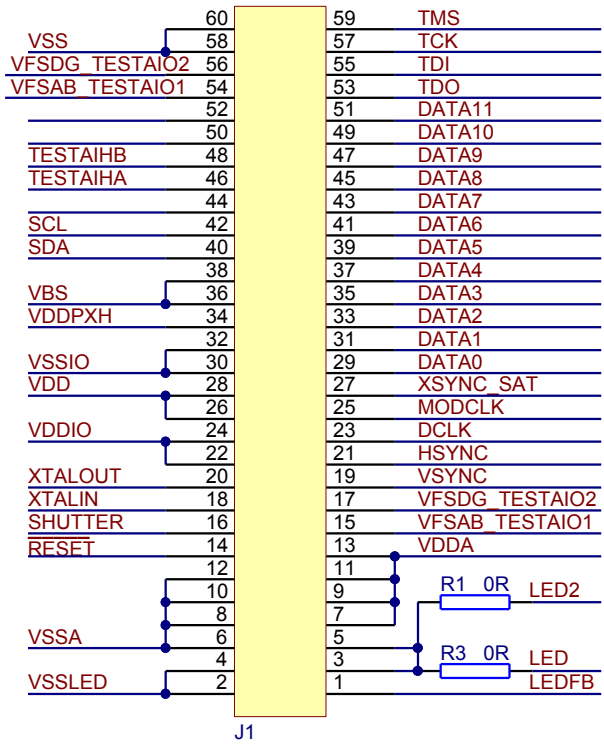


Figure 65: Pin table card connector

Figure 66 and Figure 67 show possible card connectors for interfacing the Card Edge Connector Carrier with the user's application board e.g. SAMTEC MEC6-130-02-L-DV-A / -RA1

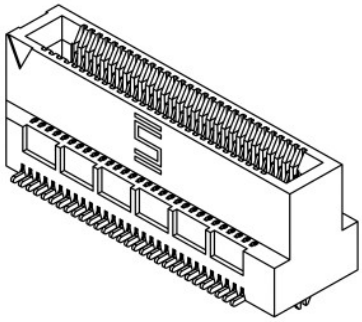


Figure 66: Vertical mount mini-edge card connector

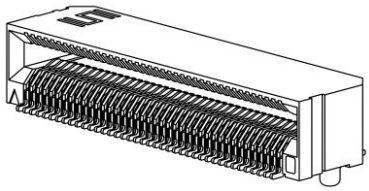


Figure 67: Right angle mini-card connector (Source: Samtec)

8. Control registers and EEPROM

8.1. Memory map

The epc635 control registers (RAM) are used for controlling all features of the chip. They are organized as 256x8 bit into 0x00 ... 0xFF address locations. The address space 0x80 ... 0xFF is EEPROM backed-up. Parameters in this section can be stored permanently between the power off/on cycles. All registers can be accessed through I²C interface by the application CPU (see chapter 5.4. I2C Slave). Multiple byte registers are stored as big-endian in the memory map.

It is further split into two pages as Control Page (0x00 ... 0x7F) and as EEPROM Page-0 (0x80 ... 0xFF).

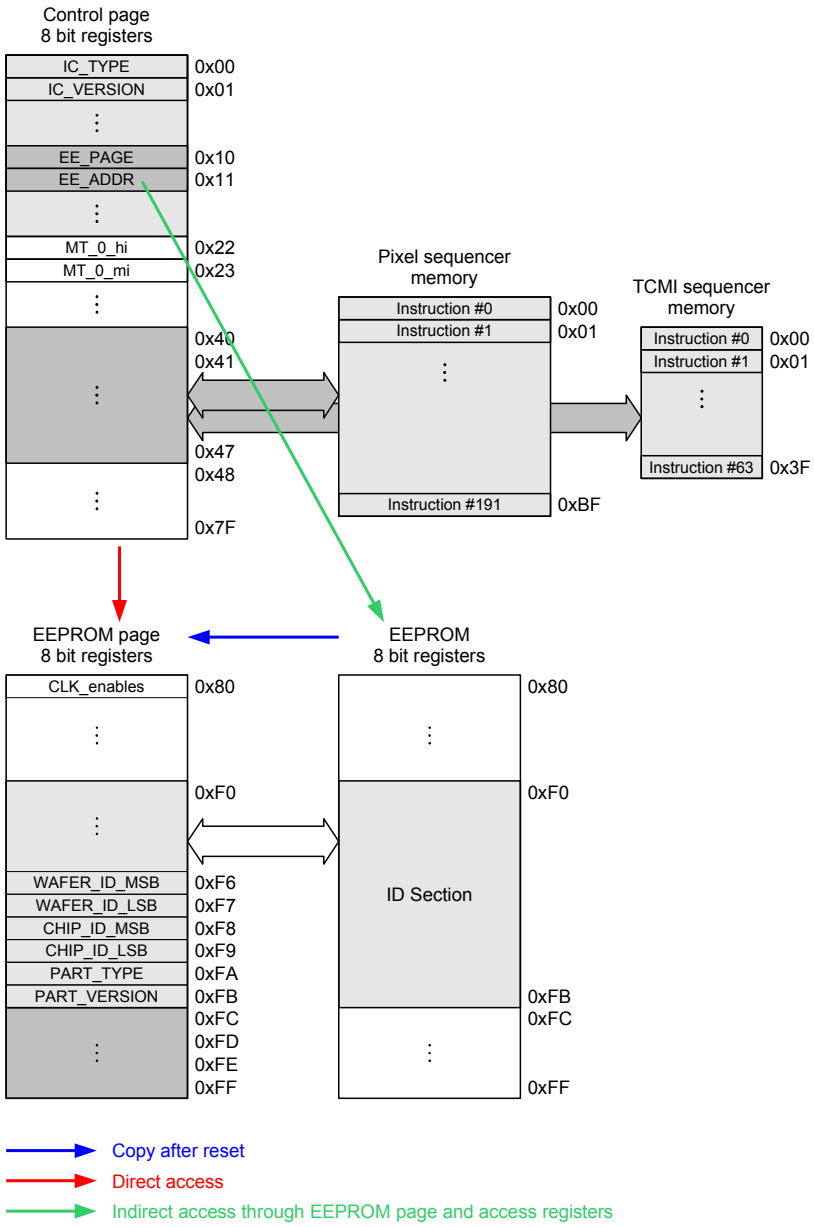


Figure 68: Memory Map

8.2. Control page

The control page contains only RW accessible configuration registers with default values during startup. The content can be changed via I²C interface. The changed values are preserved as long as the IC is powered. The registers reset back to their default values with a power-reset cycle or with a reset cycle alone.

8.3. EEPROM page, EEPROM and its charge pump

The epc635 has an 128x8-bit embedded EEPROM to store operation parameters, trimming and calibration values.

The EEPROM page address space (RAM) contains RW accessible configuration registers. Their content can be saved into the on-chip 128 x 8-bit EEPROM. It is mapped exactly one-to-one in the same RAM address space. There are various methods how the content of EEPROM can be saved or copied back and forth with the actual RW configuration registers (for details, refer to chapter 5.4.7., Control commands).

Each time a single or multiple I^2C byte-write command sequence is applied to the EEPROM address space, a charge-pump for program and erase operations is automatically enabled before the EEPROM write access starts. It is disabled after the write access is finished.

The application software must respect a 20ms/byte write time (program/erase cycle) when altering the EEPROM content through I^2C interface. When i2c_clock_stretch_en = 1, the I^2C slave interface will stretch the SCK line while the EEPROM write continues. The application CPU can monitor the SCK line with its I^2C master interface before sending the next byte-write command. If the application CPU does not support the I^2C clock stretching feature, the application should first set i2c_clock_stretch_en = 0 and keep a timer for waiting 20ms between every I^2C byte-write command sequence.

IMPORTANT:

- The built-in charge-pump for the EEPROM is a noisy circuit which shall not be enabled during frame acquisition. It can degrade the performance of the measurements. Therefore, the application must not alter the content of the EEPROM while a frame acquisition is going on.
- The number of WRITE cycles into the EEPROM is limited. It should not exceed 100 WRITE operations.

8.4. Register map

Address	Register name	Ref.	Description	Access standard	Access advanced
Memory and EEPROM controller registers					
0x00	IC_TYPE	8.6.1.	Unique IC Type for device family identification	R	R
0x01	IC_VERSION	8.6.1.	Unique IC Version for device mask identification	R	R
0x02 … 0x10	reserved			---	---
0x11	EE_ADDR	8.6.2.	Address for indirect read/write access to EEPROM	RW	RW
0x12	EE_DATA	8.6.2.	Data for indirect read/write access to EEPROM	RW	RW
0x13	EE_MASK	8.6.2.	Mask bits for indirect write access to EEPROM	RW	RW
0x14 … 0x1B	reserved			---	---
TCMI ESM registers					
0x1C	TCMI ESM_FS	8.6.3	TCMI ESM frame start label	RW	RW
0x1D	TCMI ESM_FE	8.6.3	TCMI ESM frame end label	RW	RW
0x1E	TCMI ESM_LS	8.6.3	TCMI ESM line start label	RW	RW
0x1F	TCMI ESM_LE	8.6.3	TCMI ESM line end label	RW	RW
Strap registers					
0x20 0x21	Strap_hi Strap_lo	8.6.4.	Strap Scan Value Hold Register	R	R
Modulation table registers					
0x22 0x23 0x24	MT_0_hi reserved MT_0_lo	8.6.5.	MT-vector: Sine mode DCS0	RW	RW
0x25 0x26 0x27	MT_1_hi reserved MT_1_lo	8.6.5.	MT-vector: Sine mode DCS1	RW	RW
0x28 0x29 0x2A	MT_2_hi reserved MT_2_lo	8.6.5.	MT-vector: Sine mode DCS2	RW	RW
0x2B 0x2C 0x2D	MT_3_hi reserved MT_3_lo	8.6.5.	MT-vector: Sine mode DCS3	RW	RW
0x2E … 0x39	reserved			---	---
0x3A 0x3B 0x3C	MT_8_hi reserved MT_8_lo	8.6.5.	MT-vector: Grayscale mode	RW	RW
0x3D … 0x5F	reserved			---	---
Temperature sensor registers					
0x60 0x61	Sum_Temp_tl_hi Sum_Temp_tl_lo	8.6.6.	Temperature sensor	R	R
0x62 ... 0x6F	reserved			---	---

Table 40: Address map of the control page (0x00 ~ 0x7F)

Address	Register name	Ref.	Description	Access standard	Access advanced
DLL manual control registers					
0x70	DLL_status	8.6.7.	DLL manual control: Status	---	R
0x71 0x72	DLL_fine_ctrl_ext_hi DLL_fine_ctrl_ext_lo	8.6.8.	DLL manual control: External fine control	---	RW
0x73	DLL_coarse_ctrl_ext	8.6.9.	DLL manual control: External coarse control	---	RW
0x74 0x7C	reserved			---	---
Configuration mode registers					
0x7D	CFG_Mode_Control	8.6.10.	Configuration mode: Control register	---	RW
0x7E 0x7F	reserved			---	---

Table 40 cont.: Address map of the control page (0x00 ~ 0x7F)

Address	Register name	Ref.	Description	Access standard	Access advanced
CGU registers, SEG0 *					
0x80	CLK_enables	8.7.1.	CGU register, SEG0.*: Clock enable	RW	RW
0x81 … 0x84	reserved			---	---
0x85	MOD_CLK_divider	8.7.2.	CGU register, SEG0.*: MOD clock divider	RW	RW
0x86 … 0x87	reserved			---	---
0x88	ISOURCE_CLK_divider	8.7.3	CGU register, SEG0.*: ISOURCE clock divider	RW	RW
0x89	TCMI_CLK_divider	8.7.4.	CGU register, SEG0.*: TCMI clock	RW	RW
0x8A	reserved			---	---
Modulator/demodulator registers, SEG1 *					
0x8B	Demodulation_delays	8.7.5.	Mod./demod. registers, SEG1 *: Delays	RW	RW
0x8C 0x8F	reserved			---	---
0x90	LED_driver	8.7.6.	Mod./demod. registers, SEG1 *: LED driver	RW	RW
0x91	SEQ_Control	8.7.7	Mod./demod. registers, SEG1 *: Sequencer control	RW	RW
0x92	MOD_Control **	8.7.8.	Mod./demod. registers, SEG1 *: Modulation select	RW	RW
0x93	Dist_offset **	8.7.9.	Mod. /demod. registers, SEG1 *: Distance offset	RW	RW
Pixel operating and readout control registers					
0x94	Resolution_reduction **	8.7.10.	Pixel operating and readout control	RW	RW
Frame column / row mapping registers					
0x95 … 0x9C	reserved			---	---
Shutter control registers					
0x9D	SIR_lo **	8.7.11	Shutter control: SIR (low)	RW	RW
0x9E 0x9F	Int_len_mgx1_hi ** Int_len_mgx1_lo **	8.7.12.	Shutter control: Integration time mgx1	RW	RW
0xA0 0xA1	INTM_hi ** INTM_lo **	8.7.13.	Shutter control: Integration time multiplier	RW	RW
0xA2 0xA3	Int_len_hi ** Int_len_lo **	8.7.14.	Shutter control: Integration time	RW	RW
0xA4	Shutter_Control	8.7.15.	Shutter control: Video mode / SW control	RW	RW
Power control registers					
0xA5	Power_Control (analog)	8.7.16.	Power control	RW	RW
DLL synchronization registers					
0xA6 0xA7	DLL_en_del_hi DLL_en_del_lo	8.7.17.	DLL synchronization: DLL pre-synchronization time	RW	RW
0xA8 0xA9	DLL_en_hi DLL_en_lo	8.7.17.	DLL synchronization: DLL synchronization time	RW	RW
0xAA 0xAB	DLL_measurement_rate_hi ** DLL_measurement_rate_lo **	8.7.18.	DLL synchronization: DLL measurement rate	RW	RW

Table 41: Address map of EEPROM page-0 (0x80 ~ 0xFF)

Address	Register name	Ref.	Description	Access standard	Access advanced
DLL control registers					
0xAC 0xAD	reserved			---	---
0xAE	DLL_control	8.7.19.	DLL control	RW	RW
0xAF ... 0xC9	reserved			---	---
I²C and TCMI registers, SEG4 *					
0xCA	I²C_address	8.7.20.	I²C and TCMI register, SEG4 *: Address	R	R
0xCB	I²C_TCMI_control	8.7.20.	I²C and TCMI register, SEG4 *: Control	RW	RW
TCMI registers					
0xCC	TCMI_polarity	8.7.21.	TCMI register	RW	RW
ADC control registers, SEG5 *					
0xCD	ADC_ramp	8.7.22	ADC control registers, SEG5 *		
0xCE … 0xDE	reserved			---	---
DLL filter register					
0xDF	DLL_filter_control	8.7.23	DLL filter register		
0xE0 … 0xE7	reserved			---	---
Temperature sensor calibration registers (factory settings)					
0xE8	Temp_tl_cal1	8.7.24.	Temperature sensor calibration	R	RW
0xE9	Temp_tl_cal2	8.7.24.	Temperature sensor calibration	R	RW
0xEA … 0xEF	reserved			---	---
Free EEPROM registers (for customer use), SEG6 *					
0xF0	User_1	8.7.25.	User register 1 for user data	RW	RW
0xF1	User_2	8.7.25.	User register 2 for user data	RW	RW
0xF2 ... 0xF4	reserved			---	---
CHIP ID registers (factory settings), SEG7 *					
0xF5	CUSTOMER_ID	8.7.26.	Chip ID register SEG6 *: Customer ID	R	R
0xF6 0xF7	WAFER_ID_MSB WAFER_ID_LSB	8.7.27.	Chip ID register SEG6 *: Wafer ID	R	R
0xF8 0xF9	CHIP_ID_MSB CHIP_ID_LSB	8.7.28.	Chip ID register SEG6 *: Chip ID on wafer	R	R
0xFA	PART_TYPE	8.7.29.	Chip ID register SEG6 *: Part type	R	R
0xFB	PART_VERSION	8.7.29.	Chip ID register SEG6 *: Part version	R	R
0xFC … 0xFF	reserved			---	---

Table 41 cont.: Address map of EEPROM page-0 (0x80 ~ 0xFF)

Notes:

* EEPROM protected segments. Protection will be enabled during production testing.

** Shadowed (double-buffered) registers which can be updated on-the-fly via I²C, while a frame acquisition is going on.
The new values get copied and used at the start of the next frame.
Shadow registers can be bypassed by setting Pixel_test(0x7B).shadow_regs_bypas s = 1. In this case, written values take effect immediately.

8.5. Register description: Explanation example

Register name: MOD_Control **				Description: Mod./demod. registers, SEG1 *: Modulation select				Address: 0x92
Bit No.	7	6	5	4	3	2	1	0
Bit name	mod_sel		dcs_sel		int_len _mgx1_en	lfsr_sel		
Operation	Access option for the register ■ R Read ■ W Write ■ RW Read or Write ■ (RW) Note: Needs ESPROS support to access this function by user: Read or Write							
Default	Default factory setting of the register or register bits							

8.6. Register description: Control Page (0x00 ~ 0x7F)

8.6.1. IC_

IC_TYPE	Unique IC Type for device family identification	0x00
Operation	R	
Default	according to type	

IC_VERSION	Unique IC Version for device mask identification	0x01
Operation	R	
Default	according to mask	

8.6.2. EE_ (ADDR, DATA, MASK)

EE_ADDR	Address for indirect read/write access to EEPROM	0x11
Operation	RW	
Default	0x00 (note 1)	

Note1: Default (reset) value is internally modified during start-up sequence. Therefore, when this register is read via I^2C, a different value may be observed.

EE_DATA	Data for indirect read/write access to EEPROM	0x12
Operation	RW	
Default	0x3F	

EE_MASK	Mask bits for indirect write access to EEPROM	0x13
Operation	RW	
Default	0x00	

8.6.3. TCMI ESM registers

TCMI ESM_FS				TCMI ESM frame start label				0x1C
Bit No.	7	6	5	4	3	2	1	0
Bit name	TCMI ESM_FS[7:0]							
Operation	RW							
Default	0	0	0	1	1	1	1	0
	0x1E							

TCMI ESM_FS: TCMI embedded synchronization mode frame start label. Refer to chapter 5.9.4.

TCMI ESM_FE				TCMI ESM frame end label				0x1D
Bit No.	7	6	5	4	3	2	1	0
Bit name	TCMI ESM_FE[7:0]							
Operation	RW							
Default	0	1	1	1	0	0	0	1
	0xE1							

TCMI ESM_FE: TCMI embedded synchronization mode frame end label. Refer to chapter 5.9.4.

TCMI ESM_LS				TCMI ESM line start label				0x1E
Bit No.	7	6	5	4	3	2	1	0
Bit name	TCMI ESM_LS[7:0]							
Operation	RW							
Default	1	0	1	0	1	0	1	0
	0xAA							

TCMI ESM_LS: TCMI embedded synchronization mode line start label. Refer to chapter 5.9.4.

TCMI ESM_LE				TCMI ESM line end label				0x1F
Bit No.	7	6	5	4	3	2	1	0
Bit name	TCMI ESM_LE[7:0]							
Operation	RW							
Default	0	1	0	1	0	1	0	1
	0x55							

TCMI ESM_LE: TCMI embedded synchronization mode line endt label. Refer to chapter 5.9.4.

8.6.4. Strap_

Strap_hi				Strap Scan Value Hold Register (high)				0x20
Bit No.	15	14	13	12	11	10	9	8
Bit name	reserved	strap14	strap13	reserved				
Operation	R							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see below Strap_lo register

Strap_lo				**Strap Scan Value Hold Register (low)**				**0x21**
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved							strap0
Operation	R							
Default	0	0	0	0	0	0	0	0
	0x00							

strap15:	reserved, displays pin XYSNC_SAT_CFG
strap14:	I²C Device Address: A1, on HYSNC_A1
strap13:	I²C Device Address: A0, on VYSNC_PSDIAG_A0
strap12:	reserved, displays pin DCLK
strap11 ... strap8:	reserved, not mapped to pins
strap7 ... strap0:	reserved, displays pins DATA[7:0]

Note 1: Default start-up values of these registers are only valid until end of reset phase. Values might be overwritten by external pull-up resistors during strap scan phase when reset is released.

Note 2: Read access on this register does not initiate a strap scan phase. This register simply holds the last scanned value when reset is released.

8.6.5. MT_

MT_0_hi				**MT-vector: Sine mode DCS 0 (high)**				**0x22**
Bit No.	23	22	21	20	19	18	17	16
Bit name	reserved		readout_mode		mgx1_dcs_num		mgx0_dcs_num	
Operation	RW							
Default	0	0	1	1	0	0	0	0
	0x30							

Modulation Table Vector bit definitions:

readout_mode: Readout mode of the vector
00: differential-ended readout MGA and MGB (ABS disabled, Saturation detection not active)
01: single-ended readout MGA
10: single-ended readout MGB
11: differential-ended readout MGA and MGB (ABS enabled). Default use.

mgx1_dcs_num: DCS frame number of the vector for mgx1 modulator outputs (mga1, mgb1)
00: DCS 0
01: DCS 1
10: DCS 2
11: DCS 3

mgx0_dcs_num: DCS frame number of the vector for mgx0 modulator outputs (mga0, mgb0)
00: DCS 0
01: DCS 1
10: DCS 2
11: DCS 3

MT_0_lo				**MT-vector: Sine mode DCS 0 (low)**				**0x24**
Bit No.	7	6	5	4	3	2	1	0
Bit name	override_led_ modulated_ grayscale	reserved	override_led_ off_integ	override_led_ on_integ	override_mgb _on_integ	override_mga _on_integ	override_mgb _off	override_mga _off
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

override_led_modulated_grayscale:
- Sine/grayscale: When LED_driver.led_on = 1 (torch function):
  0: LED/LD on (DC) during integration period
  1: LED/LD is modulated during integration period

override_led_off_integ:
- Sine/grayscale:
  0: LED/LD is modulated
  1: LED/LD stays off entire integration time

override_led_on_integ:
- Sine/grayscale:
  0: LED/LD is modulated
  1: LED/LD stays on the entire integration time (DC)

override_mgb_on_integ: MGB stays on (conducting) entire integration time

override_mga_on_integ: MGA stays on (conducting) entire integration time

override_mgb_off: MGB stays off (non-conducting) all the time

override_mga_off: MGA stays off (non-conducting) all the time

Note 1: Priorities: b0 over b2, b1 over b3.

Note 2: b5..b0 are ignored during DLL synchronization

MT_1_hi				**MT-vector: Sine mode DCS 1 (high)**				**0x25**
Bit No.	23	22	21	20	19	18	17	16
Bit name	reserved		readout_mode		mgx1_dcs_num		mgx0_dcs_num	
Operation	RW							
Default	0	0	1	1	0	1	0	1
	0x35							

Note: Description see MT_0_hi

MT_1_lo				**MT-vector: Sine mode DCS 1 (low)**				**0x27**
Bit No.	7	6	5	4	3	2	1	0
Bit name	override_led_ modulated_ grayscale	reserved	override_led_ off_integ/ override_led_ on_mga_ grayscale	reserved	override_mgb _on_integ	override_mga _on_integ	override_mgb _off	override_mga _off
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see MT_0_lo

MT_2_hi				**MT-vector: Sine mode DCS 2 (high)**				**0x28**
Bit No.	23	22	21	20	19	18	17	16
Bit name	reserved		readout_mode		mgx1_dcs_num		mgx0_dcs_num	
Operation	RW							
Default	0	0	1	1	1	0	1	0
	0x3A							

Note: Description see MT_0_hi

MT_2_lo				**MT-vector: Sine mode DCS 2 (low)**				**0x2A**
Bit No.	7	6	5	4	3	2	1	0
Bit name	override_led_ modulated_ grayscale	reserved	override_led_ off_integ/ override_led_ on_mga_ grayscale	reserved	override_mgb _on_integ	override_mga _on_integ	override_mgb _off	override_mga _off
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see MT_0_lo

MT_3_hi				**MT-vector: Sine mode DCS 3 (high)**				**0x2B**
Bit No.	23	22	21	20	19	18	17	16
Bit name	reserved		readout_mode		mgx1_dcs_num		mgx0_dcs_num	
Operation	RW							
Default	0	0	1	1	1	1	1	1
	0x3F							

Note: Description see MT_0_hi

MT_3_lo				**MT-vector: Sine mode DCS 3 (low)**				**0x2D**
Bit No.	7	6	5	4	3	2	1	0
Bit name	override_led_ modulated_ grayscale	reserved	override_led_ off_integ/ override_led_ on_mga_ grayscale	reserved	override_mgb _on_integ	override_mga _on_integ	override_mgb _off	override_mga _off
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see MT_0_lo

MT_8_hi				**MT-vector: Grayscale mode (high)**				**0x3A**
Bit No.	23	22	21	20	19	18	17	16
Bit name	reserved		readout_mode		mgx1_dcs_num		mgx0_dcs_num	
Operation	RW							
Default	0	0	0	1	0	0	0	0
	0x10							

Note: Description see MT_0_hi

MT_8_lo				MT-vector: Grayscale mode (low)				0x3C
Bit No.	7	6	5	4	3	2	1	0
Bit name	override_led_modulated_grayscale	reserved	override_led_off_integ/ override_led_on_mga_grayscale	reserved	override_mgb_on_integ	override_mga_on_integ	override_mgb_off	override_mga_off
Operation	RW							
Default	0	0	1	0	0	1	1	0
	0x26							

Note: Description see MT_0_lo

8.6.6. Sum_Temp_

Sum_Temp_tl_hi				Temperature sensor (high)				0x60
Bit No.	15	14	13	12	11	10	9	8
Bit name	reserved		Sum_Temp_tl[13:8]					
Operation	R							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see Sum_Temp_tl_lo

Sum_Temp_tl_lo				Temperature sensor (low)				0x61
Bit No.	7	6	5	4	3	2	1	0
Bit name	Sum_Temp_tl[7:0]							
Operation	R							
Default	0	0	0	0	0	0	0	0
	0x00							

Sum_Temp_tl_lo: Sum of 4 temperature readings. For details refer to chapter 5.8.

8.6.7. DLL_status

DLL_status				DLL manual control: Status				0x70
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved		dll_fb_lead_stat	dll_ref_lead_stat	dll_error_stat	dll_filter_stat	dll_match_stat	dll_lock_stat
Operation	R							
Default	0	0	0	0	0	0	0	0
	0x00							

dll_fb_lead_stat: Live status signal indicating the feedback waveform is leading over the reference. Used to implement the delay control by SW.

dll_ref_lead_stat: Live status signal indicating the reference waveform is leading over the feedback. Used to implement the delay control by SW.

dll_error_stat: Asserted when the delay to be compensated is out of reach for the delay line.

dll_filter_stat: Asserted when the DLL jitter filter is active.

dll_match_stat: Asserted when the residual delay between the reference and feedback is less than ~120ps.

dll_lock_stat Asserted when the residual delay between the reference and feedback is less than ~2.4ns.

8.6.8. DLL_fine_ctrl_ext

DLL_fine_ctrl_ext_hi				DLL manual control: External fine control (high)				0x71
Bit No.	15	14	13	12	11	10	9	8
Bit name	reserved						DLL_fine_ctrl_ext[9:8]	
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see DLL_fine_ctrl_ext_lo

DLL_fine_ctrl_ext_lo				DLL manual control: External fine control (low)				0x72
Bit No.	7	6	5	4	3	2	1	0
Bit name	DLL_fine_ctrl_ext[7:0]							
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

DLL_fine_ctrl_ext: Fine control from user. Valid only when DLL_control.dll_ctrl_ext = 1 (see section 8.7.19., DLL_control)

8.6.9. DLL_coarse_ctrl_ext

DLL_coarse_ctrl_ext				DLL manual control: External coarse control				0x73
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved		dll_coarse_ctrl_ext					
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

dll_coarse_ctrl_ext: Coarse control from user. Valid only when DLL_control.dll_ctrl_ext = 1 (see section 8.7.19., DLL_control)

8.6.10. CFG_Mode_

CFG_Mode_Control				Configuration mode: Control register				0x7D
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved			sys_clk _bypass	pll_bypass	pll_en	reserved	
Operation	RW							
Default	0	0	0	0	0	1	0	0
	0x04 (Note 1)							

sys_clk_bypass: Bypass OSC clock directly to the sys_clk output
0: normal
1: bypass (sys_clk = osc_clk)

pll_bypass: Bypass OSC clock directly to the PLL output as clock source to the CGU
0: normal
1: bypass (pll_clk = osc_clk)

pll_en: Enable PLL block
0: disable
1: enable

Note 1: Default (reset) value is internally modified during start-up sequence. Therefore, when this register is read via I^2C, a different value will be observed.

8.7. Register description: EEPROM Page-0 (0x80 ~ 0xFF)

8.7.1. CLK_enables

CLK_enables				CGU register, SEG0.*: Clock enable				0x80
Bit No.	7	6	5	4	3	2	1	0
Bit name	all_clks_en	mod_clk_sel	reserved				mod_clk_en	dll_clk_en
Operation	RW							
Default	0	0	1	1	1	1	1	1
	0x3F							

CGU clock enable register:

all_clks_en: Force all clock enables
0: disable
1: enable

mod_clk_sel: Select modulator clock source
0: mod_clk, from CGU
1: ext_mod_clk, from MODCLK input pin

mod_clk_en: Modulator clock
0: disable
1: enable

dll_clk_en: DLL clock
0: disable
1: enable

Note: sys_clk is always running. It cannot be disabled.

8.7.2. MOD_CLK_divider

MOD_CLK_divider				CGU register, SEG0.*: MOD clock divider				0x85
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved			mod_clk_div				
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x01							

mod_clk_div: Provides the clock to the LED / pixel-field modulator/demodulator circuits, respectively, by integer division of the PLL clock.
Default = 1: f_{mod_clk} = 80MHz / (mod_clk_div + 1) = 80MHz / 2 = 40MHz

8.7.3. ISOURCE_CLK_divider

ISOURCE_CLK_divider				CGU register, SEG0.*: ISOURCE clock divider				0x88
Bit No.	7	6	5	4	3	2	1	0
Bit name	isource_clk_div							
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x4F							

isource_clk_div: Used for ADC conversion speed-up. Refer to chapter 5.7.3, Table 14 and Table 15.

8.7.4. TCMI_CLK_divider

TCMI_CLK_divider				CGU register, SEG0.*: TCMI clock				0x89
Bit No.	7	6	5	4	3	2	1	0
Bit name	dclk_skew_en	reserved		tcmi_clk_div				
Operation	RW							
Default	0	0	0	0	0	0	1	1
	0x03							

dclk_skew_en: DCLK skew enable
0: disable
1: enable

Note1: Use dclk_skew_en = 1 to delay DCLK edge (typ. 2ns) to compensate PCB delays.
Might be particularly useful when tcmi_clk_div = 0 (divide by 1).
When set normal, DCLK edge is centered with respect to other TCMI *SYNC*, DATA[7:0] outputs.

tcmi_clk_div: Provides the clock to the TCMI by integer division of the PLL clock.
Default = 1: f_{tcmi_clk} = 80MHz / (tcmi_clk_div + 1) = 80MHz / 4 = 20MHz

8.7.5. Demodulation_delays

Demodulation_delays				Mod./demod. registers, SEG1 *: Delays				0x8B
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved				mgx_del_sel			
Operation	RW							
Default	0	0	0	0	0	0	0	1
	0x01							

mgx_del_sel: Number of pll_clk period delays on the abd, abg, mga, mgb signals in order to compensate internal LED driver delay (all modulation modes)
0: no delay
1: 1 clock
2: 2 clocks
…
12: 12 clocks
13 … 15: reserved

Note 1: When DLL is enabled, mgx_del_sel must be initialized to a higher value with respect to the external LED/LD circuit's initial delay + max. delay deviation over the system's operating conditions. Minimum recommendation: 2.

Note 2: When mgx_del_sel set > 12 (i.e. 13, 14 or 15), delay is internally limited to 12 pll_clk periods.

8.7.6. LED_driver

LED_driver				**Mod./demod. registers, SEG1 *: LED driver**				**0x90**
Bit No.	7	6	5	4	3	2	1	0
Bit name	led_drv_sel		led2_en	led_on	reserved	led_drv_en	led_inv_en	led_ssr_en
Operation	RW							
Default	1	1	0	0	0	1	0	0
	0xC4							

led_drv_sel: LED driver output max. current select e.g. for LED current 200mA ≙ 100%
00: 33%
01: 46%
10: 64%
11: 100%

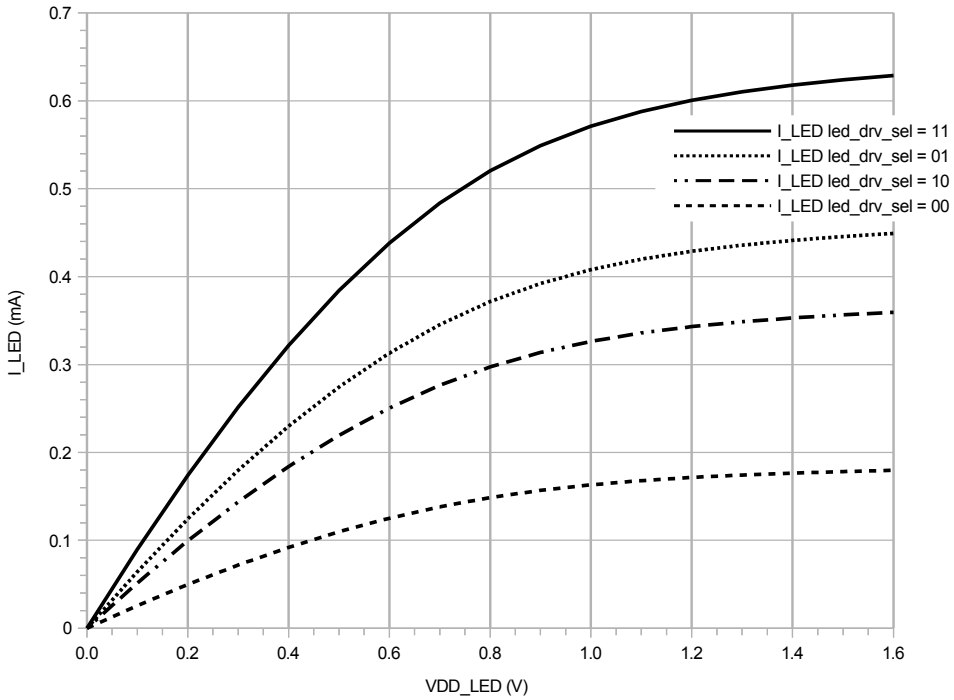


Figure 69: Output characteristic I_{LED} versus V_{DDLED}

Refer for maximum values of V_{DDLED} and I_{LED} to Table 1.

led2_en: LED2 pin enable
0: disabled (i.e. 0V)
1: enabled

led_on: LED/LD permanently on (DC) (torch function)
0: off
1: on

led_drv_en: LED driver internal power enable (independent of the modulator)
0: disable
1: enable

led_inv_en: LED signal inversion. Inversion of the output levels of the LED signal.
0: disable (signal is active when pin LED = 0, VSSLED)
1: enable (signal is active when pin LED = 1, open-drain)

Note: When led_inv_en = 0: It can directly sink the LED turn-on current without additional external driver.
When led_inv_en = 1: It needs an external pull-up resistor (open-drain) to drive the inverted value for the external LED drive circuitry.

led_ssr_en: LED slow slew rate enable.
0: disable
1: enable

8.7.7. SEQ_Control

SEQ_Control				**Mod./demod. registers, SEG1 *: Sequencer control**				**0x91**
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved	pixel_seq_ wait_row_ done_en	reserved					
Operation	RW							
Default	0	0	0	0	0	0	1	1
	0x03							

pixel_seq_wait_row_ done_en: If enabled, avoids readout rollover when using slower DCLKs e.g. DCLK < 10MHz with default ROI. Stretches HSYNC for slower TCMI interface. Causes reduced DCS frame rate due to additional 2µs per ADC conversion (t_{conv} + 2µs).
0: disable for DCLK > 10MHz (default)
1: enable for DCLK = 10MHz or lower with default ROI

8.7.8. MOD_Control **

MOD_Control **				**Mod./demod. registers, SEG1 *: Modulation select**				**0x92**
Bit No.	7	6	5	4	3	2	1	0
Bit name	mod_sel		dcs_sel		int_len _mgx1_en	reserved		
Operation	RW							
Default	0	0	1	1	0	1	0	0
	0x34							

mod_sel: Modulation select
00: Sinusoidal mode
01: reserved
10: reserved
11: Grayscale mode

dcs_sel: Number of DCS read-outs select

00: 1x DCSx: DCS0	Grayscale mode
01: 2x DCSx: DCS0, DCS1	Dual MGX mode
10: reserved	
11: 4x DCSx: DCS0, DCS1, DCS2, DCS4	Single MGX mode

Note: dcs_sel must be set to 11 for Single-MGX, 01 for Dual-MGX sine mode, or 00 for grayscale mode

int_len_mgx1_en: Int_Len_mgx1_hi/lo enable (refer to DUAL MGX mode – single DCS acquisition with different integration time)
0: disable
1: enable

Note: int_len_mgx1_en must be enabled only when dual_mgx_mode = 1, otherwise it is not effective (see Figure 33)

8.7.9. Dist_offset **

Dist_offset **				**Mod./demod. registers, SEG1 *: Distance offset**				**0x93**
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved	dist_offset						
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

dist_offset Number of pll_clk period delays on the abd, abg, mga, mgb signals.

Note: It is effective in all modulation modes.
Therefore, it is recommended to set it to 0 for all modulation modes.

8.7.10. Resolution_reduction **

Resolution_reduction **				**Pixel operating and readout control**				**0x94**
Bit No.	7	6	5	4	3	2	1	0
Bit name	dual_mgx_mode	reserved						
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

dual_mgx_mode: Dual MGX mode (refer to chapter 5.7.2, Pixel architecture)
0: single
1: dual

8.7.11. SIR_lo **

SIR_lo **				**Shutter control: SIR (low)**				**0x9D**
Bit No.	7	6	5	4	3	2	1	0
Bit name	SIR_lo							
Operation	RW							
Default	0	1	0	1	0	0	0	0
	0x50							

SIR_lo: Used for ADC conversion speed-up. Refer to chapter 5.7.3, Table 14 and Table 15.

8.7.12. Int_len_mgx1_

Int_len_mgx1_hi **				**Shutter control: Integration time mgx1 (high)**				**0x9E**
Bit No.	15	14	13	12	11	10	9	8
Operation	RW							
Default	0	0	0	0	0	1	1	1
	0x07							

Note: Description see Int_len_mgx1_lo **

Int_len_mgx1_lo **				**Shutter control: Integration time mgx1 (low)**				**0x9F**
Bit No.	7	6	5	4	3	2	1	0
Operation	RW							
Default	1	1	1	1	1	1	1	1
	0xFF							

Int_len_mgx1: Number of mod_clk periods for the integration time of mgx1 (Default: 2'047).
See Int_len_hi/lo registers (0xA2/0xA3) for functional definition details.

Note: int_len_mgx1_en = 0: These registers are updated simultaneously to the same value every time the Int_len_hi/lo (0xA2/0xA3) registers are written by I^2C.

int_len_mgx1_en =1: Int_len_mgx1_hi/lo can be updated independent of Int_len_1_hi/lo. This enables asymmetric integration times for upper and lower pixels rows

8.7.13. INTM_

INTM_hi **				Shutter control: Integration time multiplier (high)				0xA0
Bit No.	15	14	13	12	11	10	9	8
Bit name	reserved						INTM[9:8]	
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see INTM_lo **

INTM_lo **				Shutter control: Integration time multiplier (low)				0xA1
Bit No.	7	6	5	4	3	2	1	0
Bit name	INTM[7:0]							
Operation	RW							
Default	0	0	0	0	0	0	0	1
	0x01							

INTM: Multiplier of the basic integration length set in Int_len

Note: Minimum INTM value = 1

8.7.14. Int_len_

Int_len_hi **				Shutter control: Integration time (high)				0xA2
Bit No.	15	14	13	12	11	10	9	8
Operation	RW							
Default	0	0	0	0	0	1	1	1
	0x07							

Note: Description see Int_len_lo **

Int_len_lo **				Shutter control: Integration time(low)				0xA3
Bit No.	7	6	5	4	3	2	1	0
Operation	RW							
Default	1	1	1	1	1	1	1	
	0xFF							

Int_len: Number of mod_clk periods for the integration time.
Default = 2'047: Integration time = INTM * (Int_len + 1) * t_{mod_clk} = 25.6µs
Note: (Int_len + 1) should be evenly divisible by 4. Min Int_len = 7 (100ns).

8.7.15. Shutter_Control

Shutter_Control				Shutter control: Video mode / SW control				0xA4
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved						multi_frame_en	shutter_en
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

multi_frame_en: Multiple frames (video mode). Refer to chapter 5.9.2
0: disable
1: enable

shutter_en: Shutter release (SW control)
0: disable
1: enable (auto cleared). Starts acquisition

Note: shutter_en is not auto-cleared when multi_frame_en = 1

8.7.16. Power_Control (analog)

Power_Control (analog)				Power control				0xA5
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved				reserved	See note below		
Operation	RW							
Default	0	0	0	0	0	1	1	1
	0x07							

Note: Refer to chapter 7.7., Power saving options, Table 39

8.7.17. DLL_en_

DLL_en_del_hi				DLL synchronization: DLL pre-synchronization (high)				0xA6
Bit No.	15	14	13	12	11	10	9	8
Operation	RW							
Default	0	0	0	0	0	0	1	1
	0x03							

Note: Description see DLL_en_del_lo

DLL_en_del_lo				DLL synchronization: DLL pre-synchronization (low)				0xA7
Bit No.	7	6	5	4	3	2	1	0
Operation	RW							
Default	0	0	0	1	1	1	1	1
	0x1F							

DLL_en_del: Number of mod_clk periods before the DLL enable (DLL pre-synchronization time).
Default = 799: delay = (DLL_en_del + 1) * t_{mod_clk} = 10µs

DLL_en_hi				DLL synchronization: DLL synchronization time (high)				0xA8
Bit No.	15	14	13	12	11	10	9	8
Operation	RW							
Default	0	0	0	0	1	0	0	1
	0x09							

Note: Description see DLL_en_lo

DLL_en_lo				**DLL synchronization: DLL synchronization time (low)**				**0xA9**
Bit No.	7	6	5	4	3	2	1	0
Operation	RW							
Default	0	1	0	1	1	1	1	1
	0x5F							

DLL_en: Number mod_clk periods for the DLL synchronization time.
Default = 2'399: delay = (DLL_en + 1) * t_{mod_clk} = 30µs

8.7.18. DLL_measurement_rate_

DLL_measurement_rate_hi **				**DLL synchronization: DLL measurement rate (high)**				**0xAA**
Bit No.	15	14	13	12	11	10	9	8
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

Note: Description see DLL_measurement_rate_lo **

DLL_measurement_rate_lo **				**DLL synchronization: DLL measurement rate (low)**				**0xAB**
Bit No.	7	6	5	4	3	2	1	0
Operation	RW							
Default	0	0	0	0	0	1	0	0
	0x04							

DLL_measurement_rate: Number of DCS acquisitions for the DLL measurement to compensate for the phase shift of the LED driver path. When equal to 0, the modulator does not trigger the DLL to re-measure, it simply keeps the previously measured/programmed delay value during the following frame acquisition.

Note: The DLL measurement rate should be set to a multiple of the dcs frame number e.g. 4x DCS: DLL_measurement_rate = 4.

8.7.19. DLL_control

DLL_control				**DLL control**				**0xAE**
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved			dll_ctrl _ext_en	dll_rate _cnt_rst	dll_ctrl _ext	dll_fb_pol	dll_bypass
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

DLL Control:

dll_ctrl_ext_en: DLL, LED driver, feedback amplifier enable during manual control for delay locking operation (i.e. Modulator control is completely bypassed)
0: disable
1: enable

IMPORTANT
Is dll_ctrl_ext_en = 1 for manual control, the LED pin toggles continuously until this bit is cleared. If the external LED driver circuit is not designed for continuous operation, it might cause permanent damage.

Note 1: When dll_ctrl_ext = 1, DLL_filter_control.dll_filter_ctrl_s1_sel and DLL_filter_control.dll_filter_ctrl_s2_sel functions are suppressed.

Note 2: When DLL is operated manually, application SW must set dll_ctrl_ext_en = 0 while keeping dll_ctrl_ext = 1 to freeze the locked value before starting frame acquisition.

dll_rate_cnt_rst: DLL measurement rate counter reset
0: normal
1: reset (auto cleared)

dll_ctrl_ext: Manual control of the delay line using DLL_*ctrl_ext* registers
0: normal
1: manual

dll_fb_pol: Inverts the polarity of signal at LEDFB pin
0: normal
1: inverted

Note: dll_fb_pol must be set with respect to LED_driver.led_inv_en value and the external LED/LD circuit feedback point connected to the LEDFB input (external LED/LD circuit can be inverting or non-inverting depending on the application).

dll_bypass: Bypass control of DLL for system delay compensation
0: normal
1. bypass

8.7.20. I²C_TCMI_

I²C_address				**I²C_TCMI_register, SEG4 *: Address**				**0xCA**
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved	i2c_dev_adr(A6..A2)					reserved i2c_dev_adr(A1..A0)	
Operation	RW							
Default	0	0	1	0	0	0	0	0
	0x20							

i2c_dev_adr(A6..A2): I²C device address A6 … A2 of 7-bit I²C device address.
Programmable via direct access from I²C or from EEPROM during start-up, followed by an I²C general call “Device address reload” to take it into effect.

i2c_dev_adr(A1..A0): I²C device address A1, A0 of 7-bit I²C device address.
Programmable only during reset via strap pins using external pull-up resistors.

I²C_TCMI_control				**I²C and TCMI register, SEG4 *: Control**				**0xCB**
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved	tcmi_8bit_ data_sat_en	tcmi_data_format_sel		tcmi_dclk _mode	tcmi_esm_en	i2c_spike _filter_en	i2c_clock _stretch_en
Operation	RW							
Default	0	0	1	0	0	0	1	1
	0x23							

tcmi_8bit_ data_sat_en: When split modes selected (tcmi_data_format_sel = 11 or 10), forces bit DATA[0] of the LSByte = 1 when the pixel is saturated. Not effective with other TCMI data formats.
0: disabled
1: enabled

tcmi_data_format_sel: TCMI 8/12-bit data format select

00: reserved
01: 8-bit mode: Transfers the 8 MSB bits of the pixel data with 1x DCLK.
10: lsb/msb split mode: Transfers 12 bit pixel data with LSByte leading and MSByte trailing with 2x DCLK. Data are MSB aligned (default). The optional SAT bit is on the LSB.
11: msb/lsb split mode Transfers 12 bit pixel data with MSByte leading and LSByte trailing with 2x DCLK. Data are MSB aligned. The optional SAT bit is on the LSB.

tcmi_dclk_mode: TCMI DCLK mode select
0: continuous
1: gated

tcmi_esm_en: TCMI ESM mode enable
0: disabled
1: enabled

i2c_spike_filter_en: I²C pins input spike filter
0: disabled (> 1MHz)
1: enabled (≤ 1MHz, FM+)

i2c_clock_stretch_en: I²C clock stretching
0: disabled
1: enabled

Note: When i2c_spike_filter_en = 0, SDA and SCL pins can be used up to 10MHz as inputs (driven rail-to-rail by a real CMOS driver, no pull-up). They can be used up to 2MHz as outputs.

8.7.21. TCMI_polarity

TCMI_polarity				TCMI register				0xCC
Bit No.	7	6	5	4	3	2	1	0
Bit name	tcmi_data_sat_en	tcmi_xsync_sat_sel	reserved	tcmi_data_pol	tcmi_xsync_pol	tcmi_vsync_pol	tcmi_hsync_pol	tcmi_edge_sel
Operation	RW							
Default	0	1	0	0	0	0	0	1
	0x41							

tcmi_data_sat_en: Force DATA[11:0] = 0xFFF (unsigned) / 0x7FF (signed, two's complement) during data-out operation when corresponding pixel is saturated
0: disabled
1: enabled

tcmi_xsync_sat_sel: Select XSYNC / SAT output pin function (SAT function is always active high)
0: XSYNC
1: SAT

tcmi_data_pol: DATA[11:0] unsigned/signed control (for signed output, MSB bit is inverted)
0: unsigned integer (0 ... 4'095). Subtract 2'048 to get correct value (Default)
1: two's complement signed integer (-2'048 ... 2'047)

tcmi_xsync_pol: XSYNC polarity
0: XSYNC active low
1: XSYNC active high

Note: tcmi_xsync_pol is only effective when tcmi_xsync_sat_sel = 0

tcmi_vsync_pol: VSYNC polarity
0: VSYNC active low
1: VSYNC active high

tcmi_hsync_pol: HSYNC polarity
0: HSYNC active low
1: HSYNC active high

tcmi_edge_sel: DCLK edge select to align all other TCMI outputs
0: falling edge
1: rising edge

8.7.22. ADC_ramp

ADC_ramp				ADC control registers, SEG5 *				0xCD
Bit No.	7	6	5	4	3	2	1	0
Bit name	adc_numlsb_sel		reserved		adc_res_sel	adc_skip_signmsb_en	reserved	
Operation	RW							
Default	0	0	0	1	0	0	1	1
	0x13							

adc_numlsb_sel: ADC fine resolution select
00: 8 bit conversion (t_{Conv} = 12.250µs)
01: 9 bit conversion (t_{Conv} = 13.125µs)
10: 10 bit conversion (t_{Conv} = 14.000µs)
11: 11 bit conversion (t_{Conv} = 14.875µs)

Note: adc_numlsb_sel is valid for adc_res_sel = 1.

adc_res_sel: ADC resolution select
0: 12 bit conversion (t_{Conv} = 15.750µs)
1: 8, 9, 10 or 11bit (see above)

Note: When adc_res_sel = 1, the registers SIR_lo (0x9D) and ISOURCE_CLK_divider (0x88) must be set accordingly. Refer to chapter 5.7.3, Table 14 and Table 15.

adc_skip_signmsb_en ADC sign msb skipping for Grayscale mode. It forces ADC to do one additional lsb conversion (+0.875µs). It provides full 8-bit unsigned resolution for grayscale modes when combined with tcmi_data_format_sel = 01 and adc_numlsb_sel = 00. It must not be enabled for sine modes where signed output is needed for 3D TOF distance calculations.
0: disable
1: enable

Note: adc_skip_signmsb_en is valid for adc_res_sel = 1.

8.7.23. DLL_filter_control

DLL_filter_control				DLL_filter_control				0xDF
Bit No.	7	6	5	4	3	2	1	0
Bit name	reserved	dll_filter_ctrl_s2_sel			reserved	dll_filter_ctrl_s1_sel		
Operation	RW							
Default	0	0	0	0	0	0	0	0
	0x00							

dll_filter_ctrl_s2_sel: Weighted averaging DLL match values between DCS frames. This filter can be programmed to run at every 2, 4, 8, … 128 DCS frame, depending on the weight selected using this register. Once the DLL is disabled, the average values are overriding the manual control.
000: disable
001: weight 1/2
010: weight 1/4
011: weight 1/8
100: weight 1/16
101: weight 1/32
110: weight 1/64
111: weight 1/128

dll_filter_ctrl_s1_sel: Weighted averaging DLL match values during synchronization for one DCS frame. This filter can be programmed to run at every 8, 16, 32, … 512 LED pulse (i.e. runs at modulation clock frequency), depending on the weight selected using this register. Once the DLL is disabled, the average values are overriding the manual control.
000: disable
001: weight 1/8
010: weight 1/16
011: weight 1/32
100: weight 1/64
101: weight 1/128
110: weight 1/256
111: weight 1/512

Note: When DLL_control.dll_ctrl_ext = 1, dll_filter_ctrl_s1_sel and dll_filter_ctrl_s2_sel functions are suppressed.

8.7.24. Temp_

Temp_tl_cal1				Temperature sensor calibration				0xE8
Bit No.	7	6	5	4	3	2	1	0
Bit name	Temp_tl_cal1[7:0]							
Operation	RW							

Temp_tl_cal1: Temperature offset correction for calculation according the formula in chapter 5.8 by the application SW. Range approx. -27 … +27ºC with around 0.2ºC steps. The reference temperature is +27ºC.
0x7F (127) corresponds to 0ºC offset. 0xFF means not calibrated.

Temp_tl_cal2				Temperature sensor calibration				0xE9
Bit No.	7	6	5	4	3	2	1	0
Bit name	Temp_tl_cal2[7:0]							
Operation	RW							

Temp_tl_cal2: Temperature slope correction for calculation according the formula in chapter 5.8 by the application SW. 0.01LSB/ºC. 0x7F (127) corresponds to 0LSB/ºC. 0xFF means not calibrated.

8.7.25. User_

User_1	User register 1 for user data	0xF0
Operation	RW	
Default	0x00	

User_2	User register 2 for user data	0xF1
Operation	RW	
Default	0x00	

8.7.26. CUSTOMER_ID

CUSTOMER_ID	Chip ID register SEG6 *: CUSTOMER ID	0xF5
Operation	R	
Default	according engineering ID	

8.7.27. WAFER_ID_

WAFER_ID_MSB	Chip ID register SEG6 *: MSB of wafer ID	0xF6
Operation	R	
Default	according wafer ID	

WAFER_ID_LSB	Chip ID register SEG6 *: LSB of wafer ID	0xF7
Operation	R	
Default	according wafer ID	

8.7.28. CHIP_ID_

CHIP_ID_MSB	Chip ID register SEG6 *: MSB of chip ID on wafer	0xF8
Operation	R	
Default	according chip ID	

CHIP_ID_LSB	Chip ID register SEG6 *: LSB of chip ID on wafer	0xF9
Operation	R	
Default	according chip ID	

8.7.29. PART_

PART_TYPE	Chip ID register SEG6 *: Part type	0xFA
Operation	R	
Default	according part ID	

PART_VERSION	Chip ID register SEG6 *: Part version	0xFB
Operation	R	
Default	according part version	

9. Notes to various chip releases

More detailed information to the chip releases as well as latest download code for each chip version can be found in the document “Operating_Instruction_epc635-XXX-Vx.x”. It is included in the download package for the epc635 Evaluation Kit.

9.1. New features in chip versions

■ No pending items.

10. Addendum

10.1. Terms, Definitions and Abbreviations

Abbreviation	Term, Definition	Explanation
ABS	Automatic Backlight Suppression	
ADC	Analog Digital Converter	
AMR	Ratio of ambient-light / modulated light	
Big-endian		The MSB of a multi-byte register is stored in the memory location with the lowest address first. The next byte value in significance is stored at the following memory location and so on
CGU	Clock Generation Unit	
CSP	Chip Scale Package	
DCS	Differential Correlation Sample	
DLL	Delay Locked Loop	A digital circuit similar to PLL, with the main difference being the absence of an internal VCO, replaced by a delay line
Half-QQVGA	1/8 of a Quarter VGA	160x60 pixel resolution
HDR	High Dynamic Range	
IC	Integrated Circuit	
JTAG	Joint Test Action Group	
LED/LD	Light Emitting Diode / Laser Diode	
LSB	Least Significant Bit	
LSFR	Linear Feedback Shift Register	
MGA	Modulation Gate A	
MGB	Modulation Gate B	
MGX	Modulation Gate A, B	
mga	MGA control signal	
mgb	MGB control signal	
mgx	MGX control signal	
MSB	Most Significant Bit	
OSC	Oscillator	
PLL	Phase Locked Loop	
QVGA	Quarter VGA	320x240 pixel resolution
ROI	Region of Interest	
SGA	Storage Gate A	
SGB	Storage Gate B	
SGX	Storage Gate A, B	
TCMI	TOF Camera Module Interface	
TOF	Time of Flight	
VCO	Voltage Controlled OSC	
VGA	Video Graphics Array	640x480 pixel resolution
XTAL	Crystal	

Table 42: Definitions and Abbreviations

10.2. Related documents

- NXP I2C-bus specification: I2C Bus Specification and User Manual, NXP corp.
- Application note AN10 epc600/epc610 temperature and BG light compensation, ESPROS Photonics corp.
- Application note AN07 Handbook – epc600 Time-of-flight range finder chip, ESPROS Photonics corp.
- Application note AN08 Process-Rules CSP Assembly, ESPROS Photonics corp.

11. IMPORTANT NOTICE

ESPROS Photonics AG and its subsidiaries (epc) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to epc's terms and conditions of sale supplied at the time of order acknowledgment.

epc warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with epc's standard warranty. Testing and other quality control techniques are used to the extent epc deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

epc assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using epc components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

epc does not warrant or represent that any license, either express or implied, is granted under any epc patent right, copyright, mask work right, or other epc intellectual property right relating to any combination, machine, or process in which epc products or services are used. Information published by epc regarding third-party products or services does not constitute a license from epc to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from epc under the patents or other intellectual property of epc.

Resale of epc products or services with statements different from or beyond the parameters stated by epc for that product or service voids all express and any implied warranties for the associated epc product or service. epc is not responsible or liable for any such statements.

epc products are not authorized for use in safety-critical applications (such as life support) where a failure of the epc product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of epc products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by epc. Further, Buyers must fully indemnify epc and its representatives against any damages arising out of the use of epc products in such safety-critical applications.

epc products are neither designed nor intended for use in military/aerospace applications or environments unless the epc products are specifically designated by epc as military-grade. Only products designated by epc as military-grade meet military specifications. Buyers acknowledge and agree that any such use of epc products which epc has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

epc products are neither designed nor intended for use in automotive applications or environments unless the specific epc products are designated by epc as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, epc will not be responsible for any failure to meet such requirements.