Table 2.6 / Table 7.1 / Table 23.4

I/O	Pin	Analog Function	Digital Functions (GPIOPCTL PMCx Bit Field Encoding)							
			1	2	3	4	5	6	7	8
PA0	17	-	U0RX	-	-	-	-	-	-	CAN1RX
PA1	18	-	U0TX	-	-	-	-	-	-	CAN1TX
PA2	19	-	-	SSI0CLK	-	-	-	-	-	-
PA3	20	-	-	SSI0FSS	-	-	-	-	-	-
PA4	21	-	-	SSI0RX	-	-	-	-	-	-
PA5	22	-	-	SSI0TX	-	-	-	-	-	-
PA6	23	-	-	-	I2C1SCL	-	M1PWM2	-	-	-
PA7	24	-	-	-	I2C1SDC	-	M1PWM3	-	-	-
PB0	45	USB0ID	U1RX	-	-	-	-	-	T2CCP0	-
PB1	46	USB0VB	U1TX	-	-	-	-	-	T2CCP1	-
PB2	47	-	-	-	I2C0SCL	-	-	-	T3CCP0	-
PB3	48	-	-	-	I2C0SDC	-	-	-	T3CCP1	-
PB4	58	AIN10	-	SSI2CLK	-	M0PWM2	-	-	T1CCP0	CAN0RX
PB5	57	AIN11	-	SSI2FSS	-	M0PWM3	-	-	T1CCP1	CAN0TX
PB6	1	-	-	SSI2RX	-	M0PWM0	-	-	T0CCP0	-
PB7	4	-	-	SSI2TX	-	M0PWM1	-	-	T0CCP1	-
PC0	52	-	SWCLK	-	-	-	-	-	T4CCP0	-
PC1	51	-	SWDIO	-	-	-	-	-	T4CCP1	-
PC2	50	-	TDI	-	-	-	-	-	T5CCP0	-
PC3	49	-	TDO SWO	-	-	-	-	-	T5CCP1	-
PC4	16	C1-	U4RX	U1RX	-	M0PWM6	-	IDX1	WT0CCP0	U1RTS
PC5	15	C1+	U4TX	U1TX	-	M0PWM7	-	PHA1	WT0CCP1	U1CTS
PC6	14	C0+	U3RX	-	-	-	-	PHB1	WT1CCP0	USB0EPEN
PC7	13	C0-	U3TX	-	-	-	-	-	WT1CCP1	USB0PFLT
PD0	61	AIN7	SSI3CLK	SSI1CLK	I2C3SCL	M0PWM6	M1PWM0	-	WT2CCP0	-
PD1	62	AIN6	SSI3FSS	SSI1FSS	I2C3SDC	M0PWM7	M1PWM1	-	WT2CCP1	-
PD2	63	AIN5	SSI3RX	SSI1RX	-	M0FAULT0	-	-	WT3CCP0	USB0EPEN
PD3	64	AIN4	SSI3TX	SSI1TX	-	-	-	IDX0	WT3CCP1	USB0PFLT
PD4	43	USB0DM	U6RX	-	-	-	-	-	WT4CCP0	-
PD5	44	USB0DP	U6TX	-	-	-	-	-	WT4CCP1	-
PD6	53	-	U2RX	-	-	M0FAULT0	-	PHA0	WT5CCP0	-
PD7	10	-	U2TX	-	-	-	-	PHB0	WT5CCP1	NMI
PE0	9	AIN3	U7RX	-	-	-	-	-	-	-
PE1	8	AIN2	U7TX	-	-	-	-	-	-	-
PE2	7	AIN1	-	-	-	-	-	-	-	-
PE3	6	AIN0	-	-	-	-	-	-	-	-
PE4	59	AIN9	U5RX	-	I2C2SCL	M0PWM4	M1PWM2	-	-	CAN0RX
PE5	60	AIN8	U5TX	-	I2C2SDC	M0PWM5	M1PWM3	-	-	CAN0TX
PF0	28	-	U1RTS	SSI1RX	CAN0RX	-	M1PWM4	PHA0	T0CCP0	NMI
PF1	29	-	U1CTS	SSI1TX	-	-	M1PWM5	PHB0	T0CCP1	-
PF2	30	-	-	SSI1CLK	-	M0FAULT0	M1PWM6	-	T1CCP0	-
PF3	31	-	-	SSI1FSS	CAN0TX	-	M1PWM7	-	T1CCP1	-
PF4	5	-	-	-	-	-	M1FAULT0	IDX0	T2CCP0	USB0EPEN

Table 13-1. ADC Signals (64LQFP) / Table 7.3

Pin Name	Pin Number	Pin Mux / Pin Assignment	Pin Type	Buffer Type[a]	Description
AIN0	6	PE3	I	Analog	Analog-to-digital converter input 0.
AIN1	7	PE2	I	Analog	Analog-to-digital converter input 1.
AIN2	8	PE1	I	Analog	Analog-to-digital converter input 2.
AIN3	9	PE0	I	Analog	Analog-to-digital converter input 3.
AIN4	64	PD3	I	Analog	Analog-to-digital converter input 4.
AIN5	63	PD2	I	Analog	Analog-to-digital converter input 5.
AIN6	62	PD1	I	Analog	Analog-to-digital converter input 6.
AIN7	61	PD0	I	Analog	Analog-to-digital converter input 7.
AIN8	60	PE5	I	Analog	Analog-to-digital converter input 8.
AIN9	59	PE4	I	Analog	Analog-to-digital converter input 9.
AIN10	58	PB4	I	Analog	Analog-to-digital converter input 10.
AIN11	57	PB5	I	Analog	Analog-to-digital converter input 11.

Table 14-1. UART Signals (64LQFP)/ Table 8.36

Pin Name	Pin Number	Pin Mux / Pin Assignment	Pin Type	Buffer Type[a]	Description
U0Rx	17	PA0 (1)	I	TTL	UART module 0 receive.
U0Tx	18	PA1 (1)	O	TTL	UART module 0 transmit.
U1CTS	15 29	PC5 (8) PF1 (1)	I	TTL	UART module 1 Clear To Send modem flow control input signal.
U1RTS	16 28	PC4 (8) PF0 (1)	O	TTL	UART module 1 Request to Send modem flow control output line.
U1Rx	16 45	PC4 (2) PB0 (1)	I	TTL	UART module 1 receive.
U1Tx	15 46	PC5 (2) PB1 (1)	O	TTL	UART module 1 transmit.
U2Rx	53	PD6 (1)	I	TTL	UART module 2 receive.
U2Tx	10	PD7 (1)	O	TTL	UART module 2 transmit.
U3Rx	14	PC6 (1)	I	TTL	UART module 3 receive.
U3Tx	13	PC7 (1)	O	TTL	UART module 3 transmit.
U4Rx	16	PC4 (1)	I	TTL	UART module 4 receive.
U4Tx	15	PC5 (1)	O	TTL	UART module 4 transmit.
U5Rx	59	PE4 (1)	I	TTL	UART module 5 receive.
U5Tx	60	PE5 (1)	O	TTL	UART module 5 transmit.
U6Rx	43	PD4 (1)	I	TTL	UART module 6 receive.
U6Tx	44	PD5 (1)	O	TTL	UART module 6 transmit.
U7Rx	9	PE0 (1)	I	TTL	UART module 7 receive.
U7Tx	8	PE1 (1)	O	TTL	UART module 7 transmit.

Table 11-3. General-Purpose Timer Capabilities

Mode	Timer Use	Count Direction	Counter Size		Prescaler Size[a]		Prescaler Behavior (Count Direction)
			16/32-bit GPTM	32/64-bit Wide GPTM	16/32-bit GPTM	32/64-bit Wide GPTM	
One-shot	Individual	Up or Down	16-bit	32-bit	8-bit	16-bit	Timer Extension (Up), Prescaler (Down)
	Concatenated	Up or Down	32-bit	64-bit	-	-	N/A
Periodic	Individual	Up or Down	16-bit	32-bit	8-bit	16-bit	Timer Extension (Up), Prescaler (Down)
	Concatenated	Up or Down	32-bit	64-bit	-	-	N/A
Edge Time	Individual	Up or Down	16-bit	32-bit	8-bit	16-bit	Timer Extension (Both)
PWM	Individual	Down	16-bit	32-bit	8-bit	16-bit	Timer Extension

Table 11-2. General-Purpose Timers Signals (64LQFP)

Pin Name	Pin Number	Pin Mux / Pin Assignment	Pin Type	Buffer Type[a]	Description
T0CCP0	1 28	PB6 (7) PF0 (7)	I/O	TTL	16/32-Bit Timer 0 Capture/Compare/PWM 0.
T0CCP1	4 29	PB7 (7) PF1 (7)	I/O	TTL	16/32-Bit Timer 0 Capture/Compare/PWM 1.
T1CCP0	30 58	PF2 (7) PB4 (7)	I/O	TTL	16/32-Bit Timer 1 Capture/Compare/PWM 0.
T1CCP1	31 57	PF3 (7) PB5 (7)	I/O	TTL	16/32-Bit Timer 1 Capture/Compare/PWM 1.
T2CCP0	5 45	PF4 (7) PB0 (7)	I/O	TTL	16/32-Bit Timer 2 Capture/Compare/PWM 0.
T2CCP1	46	PB1 (7)	I/O	TTL	16/32-Bit Timer 2 Capture/Compare/PWM 1.
T3CCP0	47	PB2 (7)	I/O	TTL	16/32-Bit Timer 3 Capture/Compare/PWM 0.
T3CCP1	48	PB3 (7)	I/O	TTL	16/32-Bit Timer 3 Capture/Compare/PWM 1.
T4CCP0	52	PC0 (7)	I/O	TTL	16/32-Bit Timer 4 Capture/Compare/PWM 0.
T4CCP1	51	PC1 (7)	I/O	TTL	16/32-Bit Timer 4 Capture/Compare/PWM 1.
T5CCP0	50	PC2 (7)	I/O	TTL	16/32-Bit Timer 5 Capture/Compare/PWM 0.
T5CCP1	49	PC3 (7)	I/O	TTL	16/32-Bit Timer 5 Capture/Compare/PWM 1.

Table 11-1. Available CCP Pins

Timer	Up/Down Counter	Even CCP Pin	Odd CCP Pin
16/32-Bit Timer 0	Timer A	T0CCP0	-
	Timer B	-	T0CCP1
16/32-Bit Timer 1	Timer A	T1CCP0	-
	Timer B	-	T1CCP1
16/32-Bit Timer 2	Timer A	T2CCP0	-
	Timer B	-	T2CCP1
16/32-Bit Timer 3	Timer A	T3CCP0	-
	Timer B	-	T3CCP1
16/32-Bit Timer 4	Timer A	T4CCP0	-
	Timer B	-	T4CCP1
16/32-Bit Timer 5	Timer A	T5CCP0	-
	Timer B	-	T5CCP1

Table 2-9. Interrupts

Vector Number	Interrupt Number (Bit in Interrupt Registers)	Vector Address or Offset	Description
0-15	-	0x0000.0000 - 0x0000.003C	Processor exceptions
16	0	0x0000.0040	GPIO Port A
17	1	0x0000.0044	GPIO Port B
18	2	0x0000.0048	GPIO Port C
19	3	0x0000.004C	GPIO Port D
20	4	0x0000.0050	GPIO Port E
21	5	0x0000.0054	UART0
22	6	0x0000.0058	UART1
23	7	0x0000.005C	SSI0
24	8	0x0000.0060	I^2C0
25	9	0x0000.0064	PWM0 Fault
26	10	0x0000.0068	PWM0 Generator 0
27	11	0x0000.006C	PWM0 Generator 1
28	12	0x0000.0070	PWM0 Generator 2
29	13	0x0000.0074	QEI0
30	14	0x0000.0078	ADC0 Sequence 0
31	15	0x0000.007C	ADC0 Sequence 1
32	16	0x0000.0080	ADC0 Sequence 2
33	17	0x0000.0084	ADC0 Sequence 3
34	18	0x0000.0088	Watchdog Timers 0 and 1
35	19	0x0000.008C	16/32-Bit Timer 0A
36	20	0x0000.0090	16/32-Bit Timer 0B
37	21	0x0000.0094	16/32-Bit Timer 1A
38	22	0x0000.0098	16/32-Bit Timer 1B
39	23	0x0000.009C	16/32-Bit Timer 2A
40	24	0x0000.00A0	16/32-Bit Timer 2B
41	25	0x0000.00A4	Analog Comparator 0
42	26	0x0000.00A8	Analog Comparator 1
43	27	-	Reserved
44	28	0x0000.00B0	System Control

Vector Number	Interrupt Number (Bit in Interrupt Registers)	Vector Address or Offset	Description
45	29	0x0000.00B4	Flash Memory Control and EEPROM Control
46	30	0x0000.00B8	GPIO Port F
47-48	31-32	-	Reserved
49	33	0x0000.00C4	UART2
50	34	0x0000.00C8	SSI1
51	35	0x0000.00CC	16/32-Bit Timer 3A
52	36	0x0000.00D0	16/32-Bit Timer 3B
53	37	0x0000.00D4	I^2C1
54	38	0x0000.00D8	QEI1
55	39	0x0000.00DC	CAN0
56	40	0x0000.00E0	CAN1
57-58	41-42	-	Reserved
59	43	0x0000.00EC	Hibernation Module
60	44	0x0000.00F0	USB
61	45	0x0000.00F4	PWM Generator 3
62	46	0x0000.00F8	µDMA Software
63	47	0x0000.00FC	µDMA Error
64	48	0x0000.0100	ADC1 Sequence 0
65	49	0x0000.0104	ADC1 Sequence 1
66	50	0x0000.0108	ADC1 Sequence 2
67	51	0x0000.010C	ADC1 Sequence 3
68-72	52-56	-	Reserved
73	57	0x0000.0124	SSI2
74	58	0x0000.0128	SSI3
75	59	0x0000.012C	UART3
76	60	0x0000.0130	UART4
77	61	0x0000.0134	UART5
78	62	0x0000.0138	UART6
79	63	0x0000.013C	UART7
80-83	64-67	0x0000.0140 - 0x0000.014C	Reserved
84	68	0x0000.0150	I^2C2
85	69	0x0000.0154	I^2C3
86	70	0x0000.0158	16/32-Bit Timer 4A
87	71	0x0000.015C	16/32-Bit Timer 4B
88-107	72-91	0x0000.0160 - 0x0000.01AC	Reserved
108	92	0x0000.01B0	16/32-Bit Timer 5A
109	93	0x0000.01B4	16/32-Bit Timer 5B
110	94	0x0000.01B8	32/64-Bit Timer 0A
111	95	0x0000.01BC	32/64-Bit Timer 0B
112	96	0x0000.01C0	32/64-Bit Timer 1A

Vector Number	Interrupt Number (Bit in Interrupt Registers)	Vector Address or Offset	Description
113	97	0x0000.01C4	32/64-Bit Timer 1B
114	98	0x0000.01C8	32/64-Bit Timer 2A
115	99	0x0000.01CC	32/64-Bit Timer 2B
116	100	0x0000.01D0	32/64-Bit Timer 3A
117	101	0x0000.01D4	32/64-Bit Timer 3B
118	102	0x0000.01D8	32/64-Bit Timer 4A
119	103	0x0000.01DC	32/64-Bit Timer 4B
120	104	0x0000.01E0	32/64-Bit Timer 5A
121	105	0x0000.01E4	32/64-Bit Timer 5B
122	106	0x0000.01E8	System Exception (imprecise)
123-149	107-133	-	Reserved
150	134	0x0000.0258	PWM1 Generator 0
151	135	0x0000.025C	PWM1 Generator 1
152	136	0x0000.0260	PWM1 Generator 2
153	137	0x0000.0264	PWM1 Generator 3
154	138	0x0000.0268	PWM1 Fault

Table 5.12 Most popular Priority Registers used in the TM4C123GH6PM NVIC.

Priority Register	Priority Bits				Address
	31 – 29	23 - 21	15 - 13	7 - 5	
NVIC_PRI0_R	GPIO Port D	GPIO Port C	GPIO Port B	GPIO Port A	0xE000E400
NVIC_PRI1_R	SSI0, Rx Tx	UART1, Rx Tx	UART0, Rx Tx	GPIO Port E	0xE000E404
NVIC_PRI2_R	PWM Gen 1	PWM Gen 0	PWM Fault	I2C0	0xE000E408
NVIC_PRI3_R	ADC Seq 1	ADC Seq 0	Quad Encode	PWM Gen 2	0xE000E40C
NVIC_PRI4_R	Timer 0A	Watchdog	ADC Seq 3	ADC Seq 2	0xE000E410
NVIC_PRI5_R	Timer 2A	Timer 1B	Timer 1A	Timer 0B	0xE000E414
NVIC_PRI6_R	Comp 2	Comp 1	Comp 0	Timer 2B	0xE000E418
NVIC_PRI7_R	GPIO Port G	GPIO Port F	Flash Control	System Contrl	0xE000E41C
NVIC_PRI8_R	Timer 3A	SSI1, Rx Tx	UART2, Rx Tx	GPIO Port H	0xE000E420
NVIC_PRI9_R	CAN0	Quad Encod 1	I2C1	Timer 3B	0xE000E424
NVIC_PRI10_R	Hibernate	Ethernet	CAN2	CAN1	0xE000E428
NVIC_PRI11_R	uDMA Error	uDMA Soft Tfr	PWM Gen 3	USB0	0xE000E42C
NVIC_SYS_PRI3_R	SysTick	PendSV	--	Debug	0xE000ED20

Table 5.13 Relationship between each bit on interrupt enable register and related peripheral.

Enable Register	32 Enable Bits								
	0	1	2	3	4	5	6 - 29	30	31
NVIC_EN0_R	PORTA	PORTB	PORTC	PORTD	PORTE	UART0	.....	PORTF	PORTG
NVIC_EN1_R	PORTH	UART2	SSI1	Timer3A	Timer3B	I2C1	.....	UART6	UART7
NVIC_EN2_R	...	...	...	...	I2C2	I2C3	.....	WTimer0A	WTimer0B
NVIC_EN3_R	WT1A	WT1B	WT2A	WT2B	WT3A	WT3B	.....	GPIOQ2	GPIOQ3

CPRE 288: Datasheet Trainer (Camera Controller)

Introduction: The purpose of this document is to help those new to Datasheets work through the feeling of being overwhelmed by the information provided by datasheets. Tips are first given for how to go about using a datasheet. The remainder of the document then provides a representative section/chapter of a datasheet for a device called a Camera Controller.

Know the Organization: When first using a Datasheet, it is important to understand how information is organized. For microcontroller Datasheets, there is typically a section/chapter for each device. All sections/chapters of a given datasheet tend to have near identical organization. It is common for a section/chapter to have the following parts: **a) Overview, b) Details for each capability, and c) Register Description.** ***Overview:*** This part often gives the big picture purpose of the device, a list of capabilities, and a high-level diagram of how data moves through the device. ***Details for each capability:*** For a given capability, details are given on how the capability works, and how to configure the device to use that capability. ***Register Description:*** For each register associated with the device, this part gives details of the purpose of each bit in the register. This part is useful for quickly looking up device usage details once one has an overall understanding of how the device works. **By far** this is most often utilized part of a datasheet chapter/section.

How to Read: **1)** Take *15 minutes* to scan though the appropriate chapter/section to understand what type of information is provided and where. As examples: Is there a diagram that shows how data moves through the device? Are there areas that explain how to initialize and configure the device? Are there code examples? Where are the Register Descriptions? **2)** Take *20 minutes* to read the Overview and understanding the provided device diagram. **3)** Take *10 minutes* to scan through the details provided for the various capabilities. **4)** Take *15 minutes* to read the Register Descriptions. This is helpful for gaining insight into what bits you will need to care about. **5)** After taking ~60 minutes for steps 1-4, start to use the datasheet to accomplish the task you wish to perform.

Overview: The Camera Controller provides software accessible registers to simplify interacting with small cameras. Logic internal to the controller handles low-level details associated with the communication protocol used across the physical cable connecting a camera to the microcontroller. By abstracting away these complexities, the embedded systems developer can focus on higher level tasks. This internal logic also frees the microcontroller's CPU from implementing the communication protocol, allowing it to be used for computing higher level aspects of a system.

The software accessible registers allow a camera to be configured for different Frame Rates, Resolutions, Color modes, and Frame capture modes. They also allow developers to interact with a camera in a "Polling" or "Interrupt" based manner.

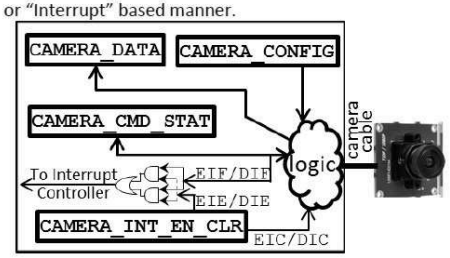

Figure 1: Camera Controller Block Diagram

Block Diagram: Figure 1 provides a high-level view of the internal workings of the Camera Controller. Its software accessible registers are shown in bold boxes.

Configuration Options: Most camera configuration options are specified using the **CAMERA_CONFIG** register. When a 1 is written to the *Enable* bit of this register, the contents of the register are sent to the camera to update its configuration and the Camera Controller is enabled. As a configuration update is being sent to the camera, the user must wait for the configuration to complete before attempting to use the camera. The *Config Pending* bit of the **CAMERA_CMD_STAT** register indicates when a configuration update has completed.

Frame Capture: Two frame capturing modes are supported: One-shot, and Continuous. In Continuous Mode, once a "Snap" command is given by writing a 1 to the *Snap* bit of the **CAMERA_CMD_STAT** register, the camera will take pictures at the specified speed until the Camera Controller is disabled. In One-shot mode, once a "Snap" command is given, the camera will only take a single picture, and then wait for another "Snap" command to be given.

Frame Retrieval: Software uses the **CAMERA_DATA** register to retrieve Frame data 8-bits at a time. The *Data Ready* bit of the **CAMERA_CMD_STAT** register indicates when a new 8-bit pixel of data has been received. Reading **CAMERA_DATA** resets the *Data Ready* bit to 0. Note: **CAMERA_DATA** is overwritten when a new pixel is received, thus if **CAMERA_DATA** is not read in a timely fashion pixel data will be lost.

Color Modes: Two color modes are supported: 8-bit color, and 8-bit gray scale. The detailed register description of the **CAMERA_DATA** register provides the 8-bit format used for each of these modes.

Interrupts: The camera controller supports the option of generating interrupts for two types of events: Data Received, and Error occurrence. When Data Received Interrupts are enabled in the **CAMERA_INT_EN_CLR** register, an interrupt signal is sent to the Interrupt Controller each time new data is received by the **CAMERA_DATA** register. Similarly, an interrupt signal is sent when an Error occurs during communication with a camera. Additionally, flags in the **CAMERA_CMD_STAT** register corresponding to the interrupts enabled will be set to 1 on the occurrence of the corresponding events.

The developer must check within their Interrupt Service Routine (ISR) which specific type of event has occurred, since the controller has only one interrupt signal for the entire device. Also within their ISR they must clear the appropriate interrupt flag, or future interrupts will not occur.

Register Descriptions: Name (Memory Map Location)

CAMERA_CONFIG (0xFFFF_A000): This register allows software to configure several aspects of a camera.

7	6	5	4	3	2	1	0
RSV	CLM	RS1	RS0	SP2	SP1	SP0	EN
RO	RW	RW	RW	RW	RW	RW	RW

Bit	Name	Function
7	Reserved (RSV)	None
6	Color Mode (CLM)	0: Gray Scale 1: Color
5:4	Resolution (RS) (Width x Height)	0: 100x100 pixels 1: 320x240 pixels 2: 640x480 pixels 3: 1280x1024 pixels
3:1	Speed (SP) (Frames per second: FPS)	0: 1 FPS 1: 30 FPS 2: 60 FPS 3: 120 FPS 4: 240 FPS >4: Undefined
0	Enable (EN)	0: Disabled 1: Enabled

CAMERA_DATA (0xFFFF_A001): This register allows software to read data from a camera one pixel at a time.

7	6	5	4	3	2	1	0
R2	R1	R0	G2	G1	G0	B1	B0
RO	RO	RO	RO	RO	RO	RO	RO

CLM = 0

Bit	Name	Function
7:0	Pixel Data	0x00: Black 0x01-0xFE: Gray Shade 0xFF: White

CLM = 1

Bit	Name	Function
7:5	Red(R)	0-7: Red Intensity level
4:2	Green(G)	0-7: Green Intensity level
1:0	Blue(B)	0-3: Blue Intensity level

Key: RO (Read Only), WO (Write Only), RW (Read and Write)

CAMERA_CMD_STAT (0xFFFF_A002): This register allows software to send commands to, and receive status from a camera.

7	6	5	4	3	2	1	0
EIF	DIF	RSV	ER	DRY	SM	S	CP
RO	RO	RW	RO	RO	RW	WO	RO

Bit	Name	Function
7	Error Int Flag(EIF)	If EIE=1, set to 1 when an error occurs
6	Data Int Flag(DIF)	If DIE=1, set to 1 when new data received
5	Reserved (RSV)	None
4	Error (ER)	1: Error has occurred 0: Reading CAMEAR_DATA sets to 0
3	Data Ready (DRY)	1: New data ready 0: Reading CAMERA_DATA sets to 0
2	Snap Mode (SM)	0:One shot Mode 1:Continuous shot Mode
1	Snap (S)	1: Take picture(s)
0	Config Pending (CP)	0: Config completed 1: Config in process

CAMERA_INT_EN_CLR (0xFFFF_A003): This allows software to enable interrupts, and to clear an interrupt once it has occurred.

7	6	5	4	3	2	1	0
RSV	RSV	EIC	DIC	RSV	RSV	EIE	DIE
RO	RO	WO	WO	RO	RO	RW	RW

Bit	Name	Function
7:6	Reserved (RSV)	None
5	Error Interrupt Clear (EIC)	0: No effect 1: Clear EIF
4	Data Interrupt Clear (DIC)	0: No effect 1: Clear DIF
3:2	Reserved (RSV)	None
1	Error Interrupt Enable (EIE)	0:Disable Error Ints 1:Enable Error Ints
0	Data Interrupt Enable (DIE)	0: Disable Data received interrupts 1: Enable Data received interrupts