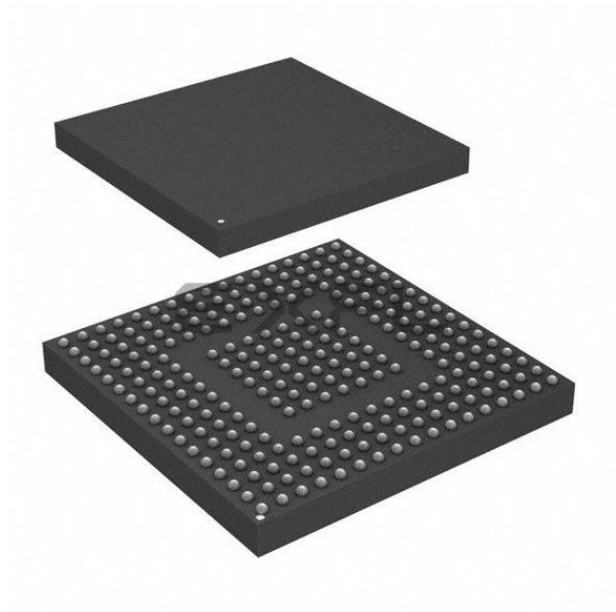

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	e200z7d
Core Size	32-Bit Dual-Core
Speed	180MHz
Connectivity	CANbus, EBI/EMI, Ethernet, FlexRay, I²C, LINbus, SPI
Peripherals	DMA, POR, PWM, WDT
Number of I/O	-
Program Memory Size	2MB (2M x 8)
Program Memory Type	FLASH
EEPROM Size	64K x 8
RAM Size	512K x 8
Voltage - Supply (Vcc/Vdd)	1.14V ~ 5.5V
Data Converters	A/D 22x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	257-LFBGA
Supplier Device Package	257-LFBGA (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/spc5675kff0vmm2r

Table 1. MPC5675K family device comparison (continued)

Features		MPC5673K	MPC5674K	MPC5675K
Packages	MAPBGA	257 pins 473 pins		
Temperature	Ambient	See the T_A recommended operating condition in the device data sheet		

1 Sphere of Replication.

2 Does not include Test or Shadow Flash memory space.

3 DSPI_0 and DSPI_1.

4 DSPI_0 has 8 chip selects; DSPI_1 and DSPI_2 have 4 chip selects each.

5 Available only on 473-pin package.

6 Any two of the three I2C can be chosen.

7 LinFlex_0, LinFlex_1, and LinFlex_2.

8 DDR available only on 473 package. Other modules available as follows:
EBI or DDR on 473 package
EBI + PDI on 473 package
DDR + PDI on 473 package
PDI only on 257 package

1.6.11 DRAM controller

The DRAM controller (available only on 473-pin devices) is a multi-port controller that monitors incoming requests on the three AHB slave ports and decides (at each rising clock edge) what command needs to be sent to the external DRAM.

The DRAM controller on this device supports the following types of memories:

- Mobile DDR (mDDR)
- DDR 1
- DDR 2 (optional)
- SDR

The controller has the following features:

- Optimized timing for 32-byte bursts and single read accesses on the AHB interface
- Optimized timing for 8-byte and 16-byte bursts on the DRAMC interface
- Supports priority elevation on the slave ports for single accesses
- 16-bit wide DRAM interface
- One chip select (CS)
- mDDR memory controller
  - 16-bit external interface
  - Address range up to 8 MB

1.6.12 Boot Assist Module (BAM)

- Enables booting via serial mode (FlexCAN, LINFlex)
- Handles static mode in case of an erroneous boot procedure
- Implemented in 8 KB ROM
- Supports Lock Step Mode (LSM) and Decoupled Parallel Mode (DPM)

1.6.13 Parallel Data Interface (PDI)

- Support for external ADC and CMOS image sensors
- Parallel interface operation up to MCU system bus frequency
- Selectable data capture from rising or falling edge
- Receive FIFO with adjustable trigger thresholds
- Data width for 8, 10, 12, 14, and 16 bits
- Data Packing Unit to pack input data on 64-bit words — data packed on 8- or 16- bit boundary, depending on input data width
- Binary increasing channel select that allows as many as eight channels to be selected
- Frame synchronization through Vsync, Hsync, PIXCLK

1.6.14 Deserial Serial Peripheral Interface (DSPI) modules

- Three serial peripheral interfaces
  - Full duplex communication ports with interrupt and eDMA request support
  - Support for all functional modes from QSPI submodule of QSMCM (MPC5xx family)
  - Support for queues in RAM
  - Six chip selects, expandable to 64 with external demultiplexers
  - Programmable frame size, baud rate, clock delay, and clock phase on a per-frame basis

- PWM outputs can operate as complementary pairs or independent channels
- Ability to accept signed numbers for PWM generation
- Independent control of both edges of each PWM output
- Synchronization to external hardware or other PWM is supported
- Double-buffered PWM registers
  - Integral reload rates from 1 to 16
  - Half-cycle reload capability
- Multiple ADC trigger events can be generated per PWM cycle via hardware
- Fault inputs can be assigned to control multiple PWM outputs
- Programmable filters for fault inputs
- Independently programmable PWM output polarity
- Independent top and bottom deadtime insertion
- Each complementary pair can operate with its own PWM frequency and deadtime values
- Individual software control for each PWM output
- All outputs can be forced to a value simultaneously
- PWMX pin can optionally output a third signal from each channel
- Channels not used for PWM generation can be used for:
  - buffered output compare functions
  - input capture functions
- Enhanced dual-edge capture functionality
- Option to supply the source for each complementary PWM signal pair from any of the following:
  - External digital pin
  - Internal timer channel
  - External ADC input, taking into account values set in ADC high and low limit registers
- Supports safety measures using DMA

1.6.20.2 Cross Triggering Unit (CTU)

The CTU provides automatic generation of ADC conversion requests on user-selected conditions without CPU load during the PWM period and with minimized CPU load for dynamic configuration.

The CTU implements the following features:

- Cross triggering between ADC, FlexPWM, eTimer, and external pins
- Double-buffered trigger generation unit with as many as eight independent triggers generated from external triggers
- Maximum operating frequency lower than or equal to platform
- Trigger generation unit configurable in sequential mode or in triggered mode
- Trigger delay unit to compensate the delay of external low-pass filter
- Double-buffered global trigger unit allowing eTimer synchronization and/or ADC command generation
- Double-buffered ADC command list pointers to minimize ADC trigger unit update
- Double-buffered ADC conversion command list with as many as twenty-four ADC commands
- Each trigger has the capability to generate consecutive commands
- ADC conversion command allows controlling ADC channel from each ADC, single or synchronous sampling, independent result queue selection
- Supports safety measures using DMA

2 Package pinouts and signal descriptions

2.1 Package pinouts

Figure 2 shows the MPC5675K in the 257 MAPBGA package. Figure 3, Figure 4, Figure 5, and Figure 6 show the MPC5675K in the 473 MAPBGA package.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
A	VSS_HV_IO	VSS_HV_IO	VDD_HV_IO	nexus MDO[5]	nexus MDO[7]	nexus MDO[9]	flexray CB_TX	flexray CA_TR_EN	VDD_HV_IO	fec RXD[2]	fec RX_CLK	fec RXD[0]	fec MDIO	fec TX_EN	fec TXD[3]	VSS_HV_IO	VSS_HV_IO	A
B	VSS_HV_IO	VSS_HV_IO	mc_cgl clk_out	can1 TXD	nexus MDO [14]	dspi2 CS1	flexray CB_TR_EN	flexray CA_TX	VSS_HV_IO	fec RXD[3]	fec RX_ER	fec RXD[1]	fec TX_ER	fec TX_CLK	can0 TXD	VDD_HV_IO	VSS_HV_IO	B
C	VDD_HV_IO	nexus MDO [15]	VSS_HV_IO	FCCU_F[1]	flexray CB_RX	etimer0 ETC[0]	etimer0 ETC[1]	etimer0 ETC[2]	etimer0 ETC[3]	JCOMP	fec CRS	fec TXD[0]	fec COL	can0 RXD	VSS_HV_PDI	pdi DATA [5]	pdi CLOCK	C
D	nexus MDO [2]	nexus MDO [3]	can1 RXD	dspi0 SOUT	RESERVED	etimer0 ETC[5]	etimer0 ETC[4]	VDD_HV_FLA	VSS_HV_FLA	fec TXD[2]	fec TXD[1]	fec RX_DV	fec MDC	VDD_HV_PDI	VSS_HV_IO	pdi DATA [0]	pdi DATA [1]	D
E	nexus MDO [0]	nexus MDO [1]	flexray CA_RX	NMI										pdi LINE_V	pdi DATA [2]	pdi DATA [3]	pdi DATA [4]	E
F	nexus MDO[6]	nexus MDO [11]	dspi1 SOUT	dspi1 SIN		VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR		mc_cgl clk_out	pdi DATA [6]	pdi DATA [7]	pdi DATA [8]	F
G	nexus MDO [4]	VDD_HV_IO	dspi0 SCK	dspi1 SCK		VDD_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VDD_LV_COR		pdi DATA [9]	pdi DATA [10]	pdi DATA [11]	pdi FRAME_V	G
H	nexus MDO [10]	VSS_HV_IO	dspi0 CS0	dspi1 CS0		VDD_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VDD_LV_COR		pdi DATA [12]	pdi DATA [13]	VDD_HV_PDI	flexpwm 0 X[0]	H
J	nexus MCKO	nexus MDO[8]	dspi2 CS0	dspi2 CS2		VDD_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VDD_LV_COR		pdi DATA [14]	pdi DATA [15]	VSS_HV_PDI	flexpwm 0 X[1]	J
K	nexus MSEO_B[0]	nexus MSEO_B[1]	nexus RDY_B	dspi0 SIN		VDD_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VDD_LV_COR		flexpwm 0 X[2]	flexpwm 0 X[3]	flexpwm 0 A[1]	flexpwm 0 B[0]	K
L	nexus EVTO_B	nexus EVTI_B	dspi2 SCK	nexus MDO [13]		VDD_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VSS_LV_COR	VDD_LV_COR		VDD_HV_DRAM_VREF	TCK	flexpwm 0 B[1]	TDO	L
M	VDD_HV_OSC	VDD_HV_IO	dspi1 CS2	nexus MDO [12]		VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR	VDD_LV_COR		flexpwm 0 B[2]	TDI	TMS	flexpwm 1 A[1]	M
N	XTALIN	VSS_HV_IO	dspi0 CS3	VSS_LV_PLL										flexpwm 0 B[3]	flexpwm 0 A[2]	flexpwm 1 A[0]	flexpwm 1 B[0]	N
P	VSS_HV_OSC	RESET	dspi0 CS2	VDD_LV_PLL	etimer1 ETC[1]	etimer1 ETC[2]	adc0 AN[0]	etimer1 ETC[3]	VSS_HV_IO	VDD_HV_IO	adc0_adc1 AN[14]	etimer1 ETC[4]	etimer1 ETC[5]	VDD_HV_IO	flexpwm 0 A[3]	flexpwm 0 A[0]	flexpwm 1 B[1]	P
R	XTAL OUT	FCCU_F[0]	VSS_HV_IO	dspi1 CS3	adc2 AN[0]	adc2 AN[3]	VDD_HV_ADR_13	adc2_adc3 AN[14]	VDD_HV_ADR_02	adc0 AN[2]	adc0_adc1 AN[13]	adc1 AN[1]	VREG_CTRL	lin0 TXD	VSS_HV_IO	flexpwm 1 A[2]	flexpwm 1 B[2]	R
T	VSS_HV_IO	VDD_HV_IO	dspi2 SOUT	adc3 AN[0]	adc3 AN[3]	adc2 AN[2]	VSS_HV_ADR_13	adc2_adc3 AN[13]	VSS_HV_ADR_02	adc0 AN[1]	adc0_adc1 AN[12]	adc1 AN[0]	adc1 AN[2]	lin0 RXD	etimer1 ETC[0]	VDD_HV_IO	VSS_HV_IO	T
U	VSS_HV_IO	VSS_HV_IO	dspi2 SIN	adc3 AN[1]	adc3 AN[2]	adc2 AN[1]	adc2_adc3 AN[11]	adc2_adc3 AN[12]	VDD_HV_ADV	VSS_HV_ADV	adc0_adc1 AN[11]	VREG_INT_ENABLE	RESET_SUP	VDD_HV_PMU	VSS_HV_PMU	VSS_HV_IO	VSS_HV_IO	U
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	

Figure 2. MPC5675K 257 MAPBGA pinout (top view)

Table 9. 257 MAPBGA pin multiplexing (continued)

Ball number	Ball type	Ball name	Alternate I/O	Additional inputs	Analog inputs	Weak pull during reset	Pad type	Power domain
F1	GPIO	nexus MDO[6][1]	A0: siul_GPIO[113] A1: _ A2: npc_wrapper_MDO[6] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
F2	GPIO	nexus MDO[11][1]	A0: siul_GPIO[108] A1: _ A2: npc_wrapper_MDO[11] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
F3	GPIO	dspi1 SOUT	A0: siul_GPIO[7] A1: dspi1_SOUT A2: _ A3: _	I: _ I: _ I: siul_EIRQ[7]	—	disabled	GP Slow/ Medium	VDD_HV_IO
F4	GPIO	dspi1 SIN	A0: siul_GPIO[8] A1: _ A2: _ A3: _	I: dspi1_SIN I: _ I: siul_EIRQ[8]	—	disabled	GP Slow/ Medium	VDD_HV_IO
F14	GPIO	mc_cgl clk_out	A0: siul_GPIO[233] A1: mc_cgl_clk_out A2: etimer2_ETC[5] A3: _	I: _ I: _ I: _	—	disabled	PDI Fast	VDD_HV_PDI
F15	GPIO	pdi DATA[6]	A0: siul_GPIO[137] A1: flexpwm2_B[0] A2: _ A3: etimer1_ETC[1]	I: pdi_DATA[6] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
F16	GPIO	pdi DATA[7]	A0: siul_GPIO[138] A1: flexpwm2_B[2] A2: _ A3: etimer1_ETC[5]	I: pdi_DATA[7] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
F17	GPIO	pdi DATA[8]	A0: siul_GPIO[139] A1: flexpwm2_A[3] A2: _ A3: _	I: pdi_DATA[8] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
G1	GPIO	nexus MDO[4][1]	A0: siul_GPIO[115] A1: _ A2: npc_wrapper_MDO[4] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO

Table 9. 257 MAPBGA pin multiplexing (continued)

Ball number	Ball type	Ball name	Alternate I/O	Additional inputs	Analog inputs	Weak pull during reset	Pad type	Power domain
M15	GPIO	TDI	A0: siul_GPIO[21] A1: _ A2: _ A3: _	I: jtagc_TDI I: _ I: _	—	pullup	GP Slow/ Medium	VDD_HV_IO
M17	GPIO	flexpwm1 A[1]	A0: siul_GPIO[157] A1: dramc_ODT A2: ebi_CS1 A3: flexpwm1_A[1]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
N3	GPIO	dspi0 CS3	A0: siul_GPIO[53] A1: dspi0_CS3 A2: i2c2_clock A3: _	I: flexpwm0_FAULT[2] I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
N14	GPIO	flexpwm0 B[3]	A0: siul_GPIO[154] A1: dramc_BA[0] A2: ebi_WE_BE_3 A3: flexpwm0_B[3]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
N15	GPIO	flexpwm0 A[2]	A0: siul_GPIO[151] A1: dramc_RAS A2: ebi_WE_BE_0 A3: flexpwm0_A[2]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
N16	GPIO	flexpwm1 A[0]	A0: siul_GPIO[155] A1: dramc_BA[1] A2: ebi_BDIP A3: flexpwm1_A[0]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
N17	GPIO	flexpwm1 B[0]	A0: siul_GPIO[156] A1: dramc_BA[2] A2: ebi_CS0 A3: flexpwm1_B[0]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
P3	GPIO	dspi0 CS2	A0: siul_GPIO[54] A1: dspi0_CS2 A2: i2c2_data A3: _	I: flexpwm0_FAULT[1] I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
P5	GPIO	etimer1 ETC[1]	A0: siul_GPIO[45] A1: etimer1_ETC[1] A2: _ A3: _	I: ctu0_EXT_IN I: flexpwm0_EXT_SYNC I: ctu1_EXT_IN	—	disabled	GP Slow/ Medium	VDD_HV_IO

Table 9. 257 MAPBGA pin multiplexing (continued)

Ball number	Ball type	Ball name	Alternate I/O	Additional inputs	Analog inputs	Weak pull during reset	Pad type	Power domain
P6	GPIO	etimer1 ETC[2]	A0: siul_GPIO[46] A1: etimer1_ETC[2] A2: ctu0_EXT_TGR A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
P7	ANA	adc0 AN[0]	—	siul_GPI[23] lin0_RXD	AN: adc0_AN[0]		Analog	VDD_HV_ADR02
P8	GPIO	etimer1 ETC[3]	A0: siul_GPIO[92] A1: etimer1_ETC[3] A2: _ A3: _	I: ctu1_EXT_IN I: mc_rgm_FAB I: siul_EIRQ[30]	—	pulldown	GP Slow/ Medium	VDD_HV_IO
P11	ANA	adc0_adc1 AN[14]	—	siul_GPI[28]	AN: adc0_adc1_AN[14]		Analog Shared	VDD_HV_ADR02
P12	GPIO	etimer1 ETC[4]	A0: siul_GPIO[93] A1: etimer1_ETC[4] A2: ctu1_EXT_TGR A3: _	I: _ I: _ I: siul_EIRQ[31]	—	disabled	GP Slow/ Medium	VDD_HV_IO
P13	GPIO	etimer1 ETC[5]	A0: siul_GPIO[78] A1: etimer1_ETC[5] A2: _ A3: _	I: _ I: _ I: siul_EIRQ[26]	—	disabled	GP Slow/ Medium	VDD_HV_IO
P15	GPIO	flexpwm0 A[3]	A0: siul_GPIO[153] A1: dramc_WEB A2: ebi_WE_BE_2 A3: flexpwm0_A[3]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
P16	GPIO	flexpwm0 A[0]	A0: siul_GPIO[147] A1: dramc_CKE A2: ebi_OE A3: flexpwm0_A[0]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
P17	GPIO	flexpwm1 B[1]	A0: siul_GPIO[163] A1: dramc_ADD[5] A2: ebi_ADD13 A3: flexpwm1_B[1]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO

Table 10. 473 MAPBGA pin multiplexing (continued)

Ball number	Ball type	Ball name	Alternate I/O	Additional Inputs	Analog Inputs	Weak pull during reset	Pad type	Power domain
A13	GPIO	fec TXD[3]	A0: siul_GPIO[204] A1: fec_TXD[3] A2: _ A3: dspi2_CS2	I: flexpwm1_FAULT[2] I: _ I: siul_EIRQ[29]	—	disabled	GP Slow/ Medium	VDD_HV_IO
A15	GPIO	pdi DATA[3]	A0: siul_GPIO[134] A1: flexpwm2_X[1] A2: _ A3: _	I: pdi_DATA[3] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
A16	GPIO	pdi DATA[1]	A0: siul_GPIO[132] A1: flexpwm2_B[3] A2: _ A3: _	I: pdi_DATA[1] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
A17	GPIO	pdi CLOCK	A0: siul_GPIO[128] A1: flexpwm2_B[1] A2: _ A3: etimer1_ETC[3]	I: pdi_CLOCK I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
A18	GPIO	pdi DATA[7]	A0: siul_GPIO[138] A1: flexpwm2_B[2] A2: _ A3: etimer1_ETC[5]	I: pdi_DATA[7] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
A19	GPIO	pdi DATA[10]	A0: siul_GPIO[141] A1: flexpwm2_X[3] A2: _ A3: _	I: pdi_DATA[10] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
A20	GPIO	pdi DATA[13]	A0: siul_GPIO[144] A1: pdi_SENS_SEL[2] A2: ctu1_EXT_TGR A3: _	I: pdi_DATA[13] I: _ I: _	—	disabled	PDI Medium	VDD_HV_PDI
A21	GPIO	pdi DATA[15]	A0: siul_GPIO[146] A1: pdi_SENS_SEL[0] A2: i2c2_data A3: _	I: pdi_DATA[15] I: ctu1_EXT_IN I: _	—	disabled	PDI Medium	VDD_HV_PDI
B3	GPIO	mc_cgl clk_out	A0: siul_GPIO[22] A1: mc_cgl_clk_out A2: etimer2_ETC[5] A3: _	I: _ I: _ I: siul_EIRQ[18]	—	disabled	GP Slow/ Fast	VDD_HV_IO

Table 10. 473 MAPBGA pin multiplexing (continued)

Ball number	Ball type	Ball name	Alternate I/O	Additional Inputs	Analog Inputs	Weak pull during reset	Pad type	Power domain
B4	GPIO	can1 TXD	A0: siul_GPIO[14] A1: can1_TXD A2: _ A3: _	I: _ I: _ I: siul_EIRQ[13]	—	disabled	GP Slow/ Medium	VDD_HV_IO
B5	GPIO	nexus MDO[14][1]	A0: siul_GPIO[219] A1: _ A2: npc_wrapper_MDO[14] A3: can3_TXD	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
B6	GPIO	dspi2 CS1	A0: siul_GPIO[9] A1: dspi2_CS1 A2: _ A3: _	I: flexpwm0_FAULT[0] I: lin3_RXD I: can2_RXD	—	disabled	GP Slow/ Medium	VDD_HV_IO
B7	GPIO	flexray CB_TR_EN	A0: siul_GPIO[52] A1: flexray_CB_TR_EN A2: _ A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Symmetric	VDD_HV_IO
B8	GPIO	flexray CA_TX	A0: siul_GPIO[48] A1: flexray_CA_TX A2: _ A3: _	I: ctu1_EXT_IN I: _ I: _	—	disabled	GP Slow/ Symmetric	VDD_HV_IO
B9	GPIO	fec RXD[3]	A0: siul_GPIO[214] A1: i2c1_data A2: _ A3: _	I: fec_RXD[3] I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
B10	GPIO	fec RX_ER	A0: siul_GPIO[215] A1: _ A2: _ A3: dspi0_CS1	I: fec_RX_ER I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
B11	GPIO	fec TXD[0]	A0: siul_GPIO[201] A1: fec_TXD[0] A2: etimer2_ETC[1] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
B12	GPIO	fec RXD[0]	A0: siul_GPIO[211] A1: i2c1_clock A2: _ A3: _	I: fec_RXD[0] I: _ I: siul_EIRQ[27]	—	disabled	GP Slow/ Medium	VDD_HV_IO

Table 10. 473 MAPBGA pin multiplexing (continued)

Ball number	Ball type	Ball name	Alternate I/O	Additional Inputs	Analog Inputs	Weak pull during reset	Pad type	Power domain
C5	GPIO	flexray CB_RX	A0: siul_GPIO[50] A1: _ A2: ctu1_EXT_TGR A3: _	I: flexray_CB_RX I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
C6	GPIO	etimer0 ETC[4]	A0: siul_GPIO[43] A1: etimer0_ETC[4] A2: _ A3: _	I: _ I: mc_rgm_ABS[0] I: _	—	pulldown	GP Slow/ Medium	VDD_HV_IO
C7	GPIO	etimer0 ETC[1]	A0: siul_GPIO[1] A1: etimer0_ETC[1] A2: _ A3: _	I: _ I: _ I: siul_EIRQ[1]	—	disabled	GP Slow/ Medium	VDD_HV_IO
C8	GPIO	etimer0 ETC[2]	A0: siul_GPIO[2] A1: etimer0_ETC[2] A2: _ A3: _	I: _ I: _ I: siul_EIRQ[2]	—	disabled	GP Slow/ Medium	VDD_HV_IO
C9	GPIO	etimer0 ETC[3]	A0: siul_GPIO[3] A1: etimer0_ETC[3] A2: _ A3: _	I: _ I: mc_rgm_ABS[2] I: siul_EIRQ[3]	—	pulldown	GP Slow/ Medium	VDD_HV_IO
C10	GPIO	fec TXD[2]	A0: siul_GPIO[203] A1: fec_TXD[2] A2: _ A3: _	I: flexpwm1_FAULT[1] I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
C11	GPIO	fec TXD[1]	A0: siul_GPIO[202] A1: fec_TXD[1] A2: _ A3: dspi2_SCK	I: flexpwm1_FAULT[0] I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
C12	GPIO	fec CRS	A0: siul_GPIO[208] A1: flexray_DBG1 A2: etimer2_ETC[3] A3: dspi0_CS5	I: fec_CRS I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
C13	GPIO	fec RX_CLK	A0: siul_GPIO[209] A1: flexray_DBG2 A2: etimer2_ETC[2] A3: dspi0_CS6	I: fec_RX_CLK I: _ I: siul_EIRQ[25]	—	disabled	GP Slow/ Medium	VDD_HV_IO

Table 10. 473 MAPBGA pin multiplexing (continued)

Ball number	Ball type	Ball name	Alternate I/O	Additional Inputs	Analog Inputs	Weak pull during reset	Pad type	Power domain
E23	GPIO	dramc BA[2]	A0: siul_GPIO[156] A1: dramc_BA[2] A2: ebi_CS0 A3: flexpwm1_B[0]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_DRAM
F1	GPIO	nexus MDO[10][1]	A0: siul_GPIO[109] A1: _ A2: npc_wrapper_MDO[10] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
F2	GPIO	nexus MDO[11][1]	A0: siul_GPIO[108] A1: _ A2: npc_wrapper_MDO[11] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
F3	GPIO	nexus MDO[6][1]	A0: siul_GPIO[113] A1: _ A2: npc_wrapper_MDO[6] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
F4	GPIO	nexus MDO[4][1]	A0: siul_GPIO[115] A1: _ A2: npc_wrapper_MDO[4] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
F20	GPIO	dramc RAS	A0: siul_GPIO[151] A1: dramc_RAS A2: ebi_WE_BE_0 A3: flexpwm0_A[2]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_DRAM
F21	GPIO	siul GPIO[194]	A0: siul_GPIO[194] A1: flexpwm0_X[0] A2: ebi_AD28 A3: _	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_DRAM
F22	GPIO	siul GPIO[148]	A0: siul_GPIO[148] A1: _ A2: ebi_CLKOUT A3: flexpwm0_B[0]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_DRAM
F23	GPIO	dramc D[5]	A0: siul_GPIO[179] A1: dramc_D[5] A2: ebi_AD13 A3: ebi_ADD29	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM

Table 40. PDI pads AC electrical characteristics (continued)

No.	Name	Prop. Delay (ns) L → H/H → L[1]		Rise/Fall Edge (ns)		Drive Load (pF)	Drive/Slew Rate Select
		Min	Max	Min	Max		MSB, LSB
2	PDI Fast	0.8/0.7 -------- 1.1/1.08	10/10	1.1/1.1	—	50	11
			15/15	2.6/2.6		200	
			15/15	2.4/2.4		50	10
			22/22	5/5		200	
			24/24	5/5		50	01
			33/33	8/8		200	
			66/66	16/16		50	00
			84/84	21/21		200	

1 L → H signifies low-to-high propagation delay and H → L signifies high-to-low propagation delay.

3.18.1 PDI pad current specifications

The power consumption of an I/O segment is dependent on the usage of the pins on a particular segment. The power consumption is the sum of all output pin currents for a particular segment. The output pin current can be calculated based on the voltage, frequency, and load on the pin.

Table 41. PDI pad current specifications

Pad Type	Frequency (MHz)	Load (pF)	Voltage (V)	Drive/Slew Rate Select	Current (mA)
PDI Medium	66	50	3.6	11	8.7
	33	50	3.6	10	3.8
	20	50	3.6	01	2.3
	3	50	3.6	00	0.38
	3	200	3.6	00	1.5
PDI Fast	66	50	3.6	11	12
	50	50	3.6	10	6.2
	33	50	3.6	01	4.0
	20	50	3.6	00	2.4
	20	200	3.6	00	8.9

3.18.2 Power Sequence Pin States for PDI Pads

Table 42. Power sequence pin states for PDI pads

VDD_LV_COR	VDD_HV_IO	VDD_HV_PDI	Pad Function
Low	Low	High	Outputs drive high

Table 49. Output drive current @ V_{DDE} = 2.5 V (±200 mV)

Pad Name	Drive Mode	Minimum I_{OH} (mA)[1]	Minimum I_{OL} (mA)[2]
DRAM ACC	011	–16.2	16.2
DRAM DQ	011		
DRAM CLK	011		

1 I_{OH} is defined as the current sourced by the pad to drive the output to V_{OH}.
2 I_{OL} is defined as the current sunk by the pad to drive the output to V_{OL}.

Table 50. DRAM pads AC electrical specifications ($V_{DD\_HV\_DRAM}$ = 2.5 V)

No.	Pad Name	Prop. Delay (ns) L → H/H → L[1]		Rise/Fall Edge (ns)		Drive Load (pF)	Drive/Slew Rate Select
		Min	Max	Min	Max		MSB, LSB
1	DRAM ACC	1.4/1.5	2.5/2.4	2.1/2.1	4.3/4.1	5	011
		1.7/1.7	2.8/2.7	0.6/0.7	1.1/1.3	20	
2	DRAM DQ	1.4/1.5	2.5/2.4	2.1/2.1	4.3/4.1	5	011
		1.7/1.7	2.8/2.7	0.6/0.7	1.1/1.3	20	
3	DRAM CLK	1.4/1.4	2.4/2.4	2.1/2.1	4.4/4.1	5	011
		1.6/1.6	2.7/2.7	0.6/0.7	1.6/1.8	20	

1 L → H signifies low-to-high propagation delay and H → L signifies high-to-low propagation delay.

3.19.3 DRAM pads electrical specification ($V_{DD\_HV\_DRAM}$ = 1.8 V)

Table 51. DRAM pads DC electrical specifications ($V_{DD\_HV\_DRAM}$ = 1.8 V)

No.	Symbol		Parameter	Condition	Min	Max	Unit
1	$V_{DD\_HV\_DRAM}$	SR	I/O supply voltage	—	1.62	1.9	V
2	$V_{DD\_HV\_DRAM\_VREF}$	CC	Input reference voltage	—	0.49 × $V_{DD\_HV\_DRAM}$	0.51 × $V_{DD\_HV\_DRAM}$	V
3	$V_{DD\_HV\_DRAM\_VTT}$	CC	Termination voltage[1]	—	$V_{DD\_HV\_DRAM\_VREF}$ – 0.04	$V_{DD\_HV\_DRAM\_VREF}$ + 0.04	V
4	V_{IH}	CC	Input high voltage	—	$V_{DD\_HV\_DRAM\_VREF}$ + 0.125	—	V
5	V_{IL}	CC	Input low voltage	—	—	$V_{DD\_HV\_DRAM\_VREF}$ – 0.125	V
6	V_{OH}	CC	Output high voltage	—	1.42	—	V
7	V_{OL}	CC	Output low voltage	—	—	0.28	V

1 BGA473: Termination voltage can be supplied via package pins. BGA257 Termination voltage internally tied as the BGA257 does not provide DRAM interface. Disable ODT.

Table 52. Output drive current @ V_{DDE} = 1.8 V (±100 mV)

No.	Pad Name	Drive Mode	Minimum I_{OH} (mA)[1]	Minimum I_{OL} (mA)[2]
1	DRAM ACC	000	–3.57	3.57
		001	–7.84	7.84
		010	–5.36	5.36
		110	–13.4	13.4
2	DRAM DQ	000	–3.57	3.57
		001	–7.84	7.84
		010	–5.36	5.36
		110	–13.4	13.4
3	DRAM CLK	000	–3.57	3.57
		001	–7.84	7.84
		010	–5.36	5.36
		110	–13.4	13.4

1 I_{OH} is defined as the current sourced by the pad to drive the output to V_{OH}.
2 I_{OL} is defined as the current sunk by the pad to drive the output to V_{OL}.

Table 53. DRAM pads AC electrical specifications ($V_{DD\_HV\_DRAM}$ = 1.8 V)

No.	Pad Name	Prop. Delay (ns) L → H/H → L[1]		Rise/Fall Edge (ns)		Drive Load (pF)	Drive/Slew Rate Select
		Min	Max	Min	Max		MSB, LSB
1	DRAM ACC	1.4/1.4	2.4/2.4	0.6/1.0	2.7/2.6	5	000
		1.7/1.7	2.8/2.7	0.2/0.4	0.5/0.6	20	
		1.4/1.5	2.4/2.5	1.1/1.1	3.0/2.7	5	001
		1.7/1.7	2.8/2.8	0.4/0.4	0.7/0.7	20	
		1.4/1.5	2.4/2.4	1.0/1.1	2.9/2.7	5	010
		1.7/1.7	2.8/2.7	0.3/0.4	0.6/0.7	20	
		1.4/1.5	2.5/2.5	1.5/1.1	3.1/2.6	5	110
		1.7/1.8	2.8/2.8	0.4/0.4	0.7/0.6	20	
2	DRAM DQ	1.4/1.4	2.4/2.4	0.6/1.0	2.7/2.6	5	000
		1.7/1.7	2.8/2.7	0.2/0.4	0.5/0.6	20	
		1.4/1.5	2.4/2.5	1.1/1.1	3.0/2.7	5	001
		1.7/1.7	2.8/2.8	0.4/0.4	0.7/0.7	20	
		1.4/1.5	2.4/2.4	1.0/1.1	2.9/2.7	5	010
		1.7/1.7	2.8/2.7	0.3/0.4	0.6/0.7	20	
		1.4/1.5	2.5/2.5	1.5/1.1	3.1/2.6	5	110
		1.7/1.8	2.8/2.8	0.4/0.4	0.7/0.6	20	

Table 58. RESET sequences

No.	Symbol		Parameter	T_{Reset}			Unit
				Min	Typ	Max[1]	
1	T_{DRB}	CC	Destructive Reset Sequence, BIST enabled	50	60	70	ms
2	T_{DR}	CC	Destructive Reset Sequence, BIST disabled	40	400	1000	µs
3	T_{ERLB}	CC	External Reset Sequence Long, BIST enabled	50	60	70	ms
4	T_{FRL}	CC	Functional Reset Sequence Long	40	300	600	µs
5	T_{FRS}	CC	Functional Reset Sequence Short	1	3	10	µs

1 The maximum value is applicable only if the reset sequence duration is not prolonged by an extended assertion of $\overline{\text{RESET}}$ by an external reset generator.

3.21.2 Reset sequence description

The figures in this section show the internal states of the MPC5675K during the five different reset sequences. The doted lines in the figures indicate the starting point and the end point for which the duration is specified in Table 58. The start point and end point conditions as well as the reset trigger mapping to the different reset sequences is specified in Section 3.21.3, Reset sequence trigger mapping.

With the beginning of DRUN mode, the first instruction is fetched and executed. At this point, application execution starts and the internal reset sequence is finished.

The following figures show the internal states of the MPC5675K during the execution of the reset sequence and the possible states of the $\overline{\text{RESET}}$ signal pin.

NOTE

$\overline{\text{RESET}}$ is a bidirectional pin. The voltage level on this pin can either be driven low by an external reset generator or by the MPC5675K internal reset circuitry. A high level on this pin can only be generated by an external pullup resistor which is strong enough to overdrive the weak internal pulldown resistor. The rising edge on $\overline{\text{RESET}}$ in the following figures indicates the time when the device stops driving it low. The reset sequence durations given in Table 58 are applicable only if the internal reset sequence is not prolonged by an external reset generator keeping $\overline{\text{RESET}}$ asserted low beyond the last PHASE3.

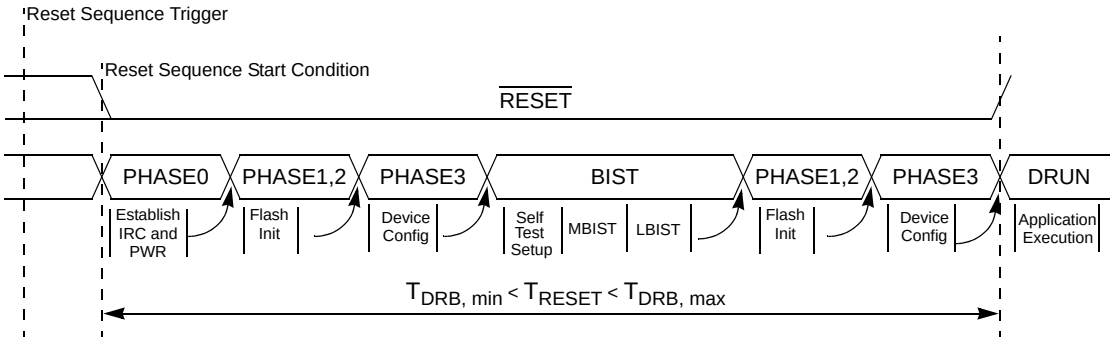

Figure 12. Destructive reset sequence, BIST enabled

Table 66. Nexus debug port timing DIVIDE by 4 DDR mode[1]

No.	Symbol		Parameter	Conditions	Min	Max	Unit
1	t_{MCKO}	CC	MCKO cycle time	—	22.22	—	ns
2	t_{MDC}	CC	MCKO duty cycle[2]	—	50	50	%
3	t_{MDOV}	CC	MCKO Low to MDO, $\overline{\text{MSEO}}$, $\overline{\text{EVTO}}$ data valid[3]	—	–2.23	4.45	ns
4	t_{EVTIPW}	CC	$\overline{\text{EVTI}}$ pulse width	—	4.0	—	t_{JCYC}
5	t_{PW}	CC	MDO, $\overline{\text{MSEO}}$,$\overline{\text{EVTO}}$ pulse width in DDR mode	—	0.5	—	t_{MCKO}

1 All Nexus timing relative to MCKO is measured from 50% of MCKO and 50% of the respective signal.Rise/Fall time for Nexus signals can be derived from Fast GPIO pad specification section.

2 Jitter/tolerance for MCKO clock is derived from PLL. Please see PLL section for jitter specification.

3 MDO, $\overline{\text{MSEO}}$, and $\overline{\text{EVTO}}$ data is held valid for half of time period. Using this time period, Data valid window for these signals is between 0.2 t_{MCKO} to 0.4 t_{MCKO} starting from each MCKO edge.

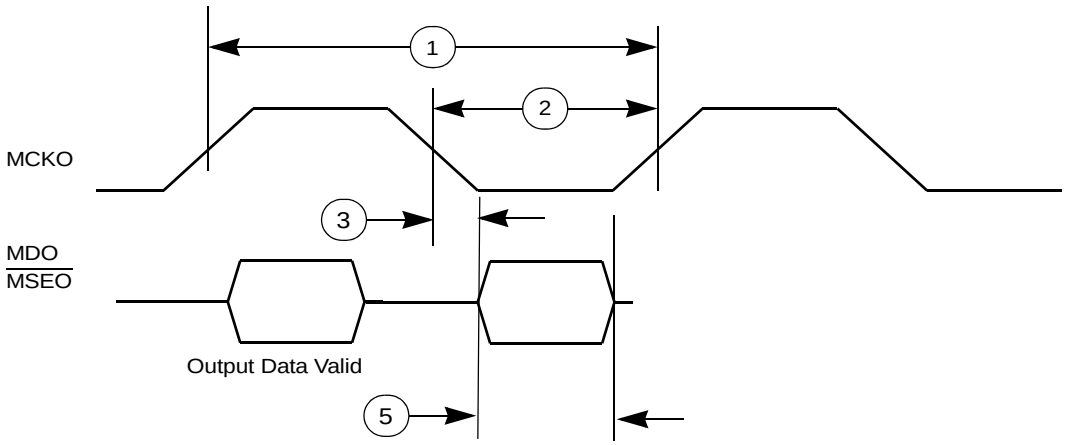

Figure 31. Nexus DDR mode timing

3.22.4 External interrupt timing (IRQ pins)

Table 67. External interrupt timing (NMI IRQ)

No.	Symbol		Parameter	Conditions	Min	Max	Unit
1	t_{IPWL}	SR	IRQ pulse width low	—	3	—	t_{CYC}
2	t_{IPWH}	SR	IRQ pulse width high	—	3	—	t_{CYC}
3	t_{ICYC}	SR	IRQ edge to edge time[1]	—	6	—	t_{CYC}

1 Applies when IRQ pins are configured for rising edge or falling edge events, but not both.

Table 68. External interrupt timing (GPIO IRQ)

No.	Symbol		Parameter	Conditions	Min	Max	Unit
1	t_{IPWL}	SR	IRQ pulse width low	—	3	—	t_{CYC}
2	t_{IPWH}	SR	IRQ pulse width high	—	3	—	t_{CYC}

Table 71. PDI electrical characteristics (continued)

No.	Symbol		Parameter	Conditions	Min	Max	Unit
2	t_{PDI_IS}	SR	Input setup time[1]	—	3	—	ns
3	t_{PDI_IH}	SR	Input hold time[1]	—	3	—	ns

1 Data can be captured at both launching and capturing edge of PDI_CLK.

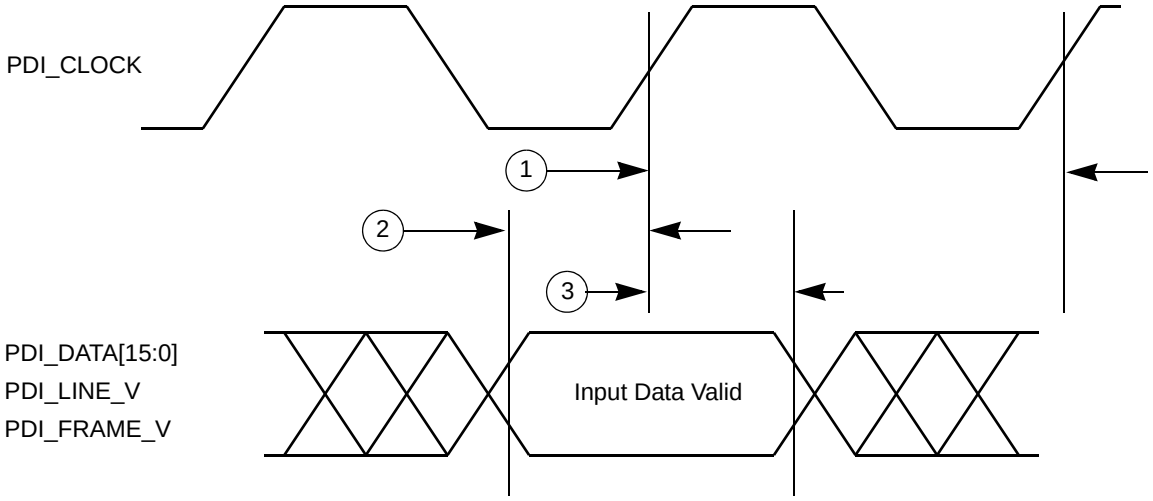

Figure 44. PDI timing

3.22.8 Fast Ethernet interface

MII signals use CMOS signal levels compatible with devices operating at either 5.0 V or 3.3 V. Signals are not TTL compatible. They follow the CMOS electrical characteristics.

3.22.8.1 MII receive signal timing (RXD[3:0], RX_DV, RX_ER, and RX_CLK)

The receiver functions correctly up to a RX_CLK maximum frequency of 25 MHz +1%. There is no minimum frequency requirement. In addition, the system clock frequency must exceed two times the RX_CLK frequency.

Table 72. MII receive signal timing

No.	Parameter	Min	Max	Unit
1	RXD[3:0], RX_DV, RX_ER to RX_CLK setup	5	—	ns
2	RX_CLK to RXD[3:0], RX_DV, RX_ER hold	5	—	ns
3	RX_CLK pulse width high	40%	60%	RX_CLK period
4	RX_CLK pulse width low	40%	60%	RX_CLK period

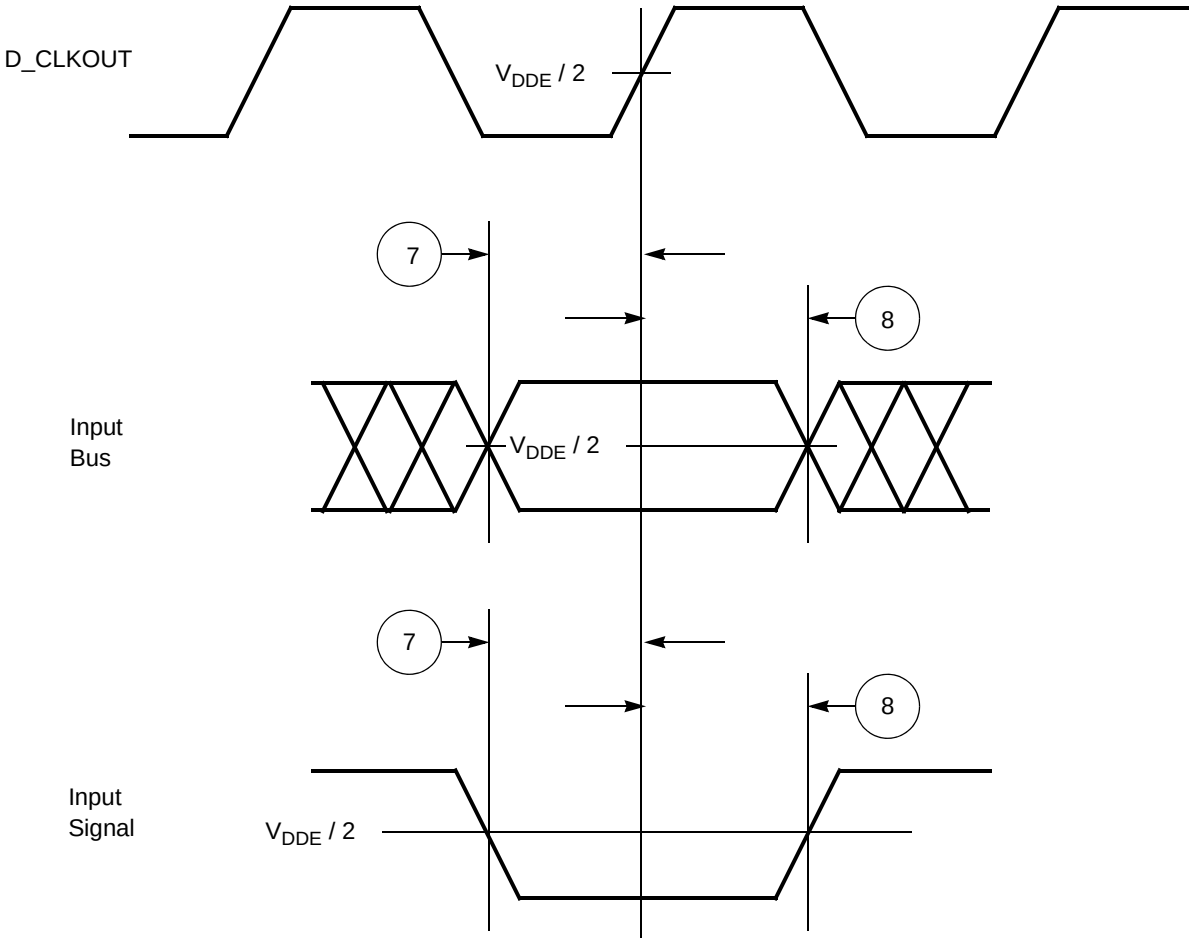

Figure 51. Synchronous input timing

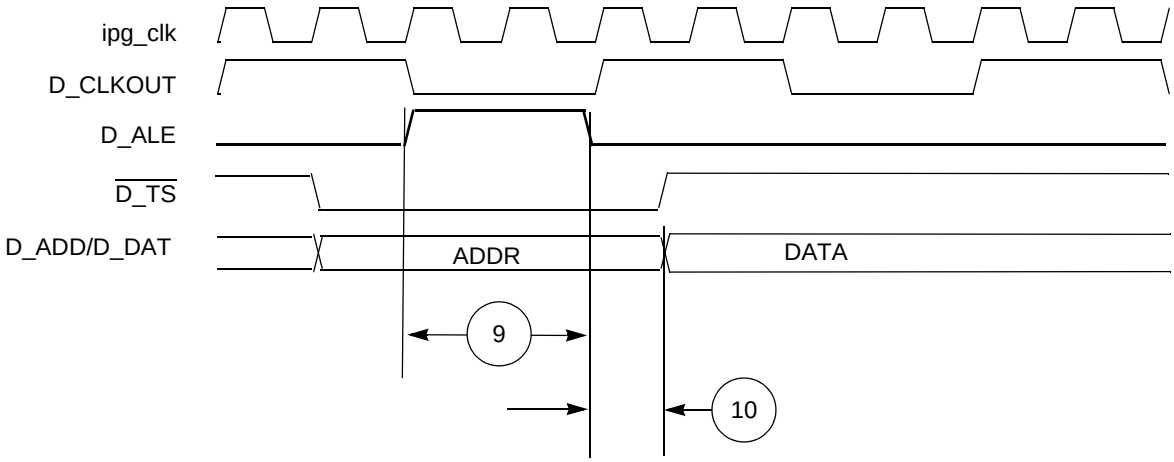

Figure 52. ALE signal timing

Table 79. Revision history (continued)

Revision	Date	Description of Changes
5 (cont.)	6 Dec 2011	In Section 3.19, DRAM pad specifications, added the note “0.7 V overshoot/undershoot can be allowed to occur repeatedly throughout the product expected lifetime and will not cause any long term reliability issue.” In Table 45 (DRAM pads DC electrical specifications ($V_{DD\_HV\_DRAM}$ = 3.3 V)): • Updated $V_{DD\_HV\_DRAM\_VTT}$ minimum value to $V_{DD\_HV\_DRAM\_VREF}$ – 0.05 (changed “x” to “–”) • Updated V_{IL} maximum value to $V_{DD\_HV\_DRAM\_VREF}$ – 0.2 (changed “x” to “–”) • Removed ODT conditions for V_{OH} and V_{OL}. • Updated V_{OL} maximum value to $V_{DD\_HV\_DRAM\_VTT}$ – 0.8 (changed “x” to “–”) In Table 48 (DRAM pads DC electrical specifications ($V_{DD\_HV\_DRAM}$ = 2.5 V)), removed ODT conditions for V_{OH} and V_{OL}. In Table 51 (DRAM pads DC electrical specifications ($V_{DD\_HV\_DRAM}$ = 1.8 V)): • Changed the minimum specification for $V_{DD\_HV\_DRAM}$ (was 1.7 V, is 1.62 V). • Removed ODT conditions for V_{OH} and V_{OL}. • Updated V_{OH} minimum value to 1.42 V • Updated V_{OL} maximum value to 0.28 V Added Section 3.20.2, RESET_SUP_B pin characteristics. Updated Note under Section 3.21.4.2, External VREG mode. Updated Figure 18 (External VREG mode, RESET_SUP rises after $V_{DD\_HV\_xxx}$ are stable) to add T_{RSTSUP}. Added Section 3.22.2.1, Standard interface timing, and revised the specifications in Table 62 (JTAG pin AC electrical characteristics). Added Section 3.22.2.2, Interface timing for Full Cycle mode. Replaced the contents of Section 3.22.3, Nexus timing, with the following: • Table 64 (Nexus debug port timing Div mode = 2) and Figure 29 (Nexus SDR (Even divisor) timing) • Table 65 (Nexus debug port timing Divide by 3 SDR mode) and Figure 30 (Nexus SDR output timing for DIV=3) • Table 66 (Nexus debug port timing DIVIDE by 4 DDR mode) and Figure 31 (Nexus DDR mode timing) In Section 5, Orderable parts, updated the orderable part numbers. Updated the entry for Rev. 4 in this revision history.

Table 79. Revision history (continued)

Revision	Date	Description of Changes
6 (cont.)	6 Feb 2012	In Table 68 (External interrupt timing (GPIO IRQ)): • Changed T_{IPWL} min value from TBD to 3. • Changed T_{IPWH} min value from TBD to 3. • Changed T_{ICYC} min value from TBD to 6. • Changed all units from ns to t_{CYC}. In Table 77 (I^2C SCL and SDA input timing specifications), corrected the line numbering.
6.1	30 Mar 2012	No content changes, technical or editorial, were made in this revision. Change bars are identical to those in Rev. 6. Removed the “preliminary” footers throughout. Changed “Data Sheet: Advance Information” to “Data Sheet: Technical Data” on page Removed the “product under development” disclaimer on page 1.
7	18 May 2012	Minor editorial changes and improvements throughout. In Section 1.3, Device comparison, Table 1 (MPC5675K family device comparison), • Changed the CPU/Data Cache entry from "16 KB, 4-way with EDC (SoR)" to "16 KB, 4-way with Parity (SoR)". • Added footnotes to stipulate the peripheral instances that are used on derivative devices: - Added footnote to **MPC5673K** DSPI module: “DSPI_0 and DSPI_1.“ - Added footnote to **MPC5673K** I2C module: “I2C_0 and I2C_1.“ - Added footnote to **MPC5673K** LinFlex module: “LinFlex_0, LinFlex_1, and LinFlex_2“ In Section 1.4, Block diagram: • Added missing modules (PMC, SPE2, VLE, and flash. • Added an arrow each from Core_0 and Core_1 to the XBAR modules to represent the data path. • Updated the Redundancy Checkers to reflect the actual implementation. • Renamed the “JTAG/Nexus” block to “Debug”, with JTAG and Nexus shown as submodules. In Section 1.5, Feature list, changed “Junction temperature sensor” to “Silicon substrate (die) temperature sensor”. In Section 1.6.1, High-performance e200z7d core processor and Section 1.6.9, Cache memory, removed the bullet “Supports tag and data parity" and added the following bullets: — Supports tag and data cache parity — Supports EDC for instruction cache In Section 1.6.19, System Timer Module (STM), changed “Duplicated periphery to guarantee that safety targets (SIL3) are achieved” to “Replicated periphery to provide safety measures respective to high safety integrity levels (for example, SIL 3, ASIL D)” In Section 1.6.20.2, Cross Triggering Unit (CTU), changed “DMA support with safety features” to “Supports safety measures using DMA”. In Section 1.6.21, Redundancy Control and Checker Unit (RCCU), changed “Duplicated module to guarantee highest possible diagnostic coverage (check of checker)” to “Duplicated module to enable high diagnostic coverage (check of checker)”. In Section 1.6.22, Software Watchdog Timer (SWT), • Changed “Duplicated periphery to guarantee that safety targets (SIL3) are achieved” to “Replicated periphery to provide safety measures respective to high safety integrity levels (for example, SIL 3, ASIL D)”. • Changed “Allows high level of safety (SIL3 monitor)” to “Provides measures to target high safety integrity levels (for example, SIL 3, ASIL D)”. In Section 1.6.25, Cyclic Redundancy Checker (CRC) unit, in the sentence “Key engine to be coupled with communication periphery where CRC application is added to allow implementation of safe communication protocol”, changed “allow” to “support”.