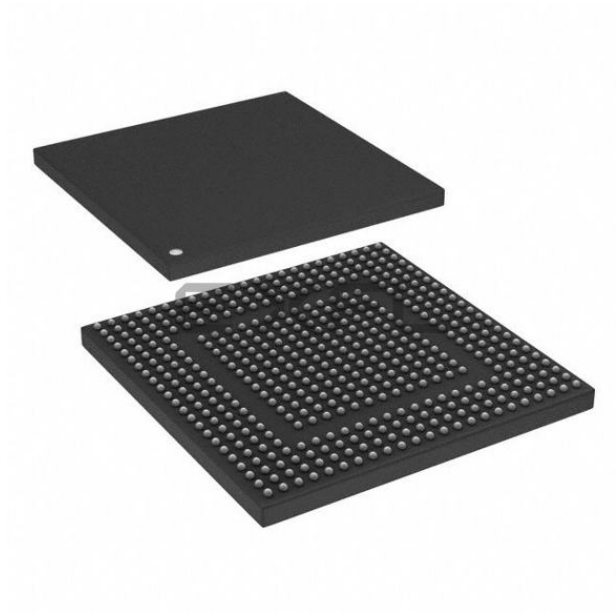

Welcome to E-XFL.COM

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	1.5MB (1.5M x 8)
Program Memory Type	FLASH
EEPROM Size	64K x 8
RAM Size	384K x 8
Voltage - Supply (Vcc/Vdd)	1.14V ~ 5.5V
Data Converters	A/D 34x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	473-LFBGA
Supplier Device Package	473-MAPBGA (19x19)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/spc5674kff0mms2

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 I2C_0 and I2C_1.

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

- 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
- DMA support with safety features

Table 3. 257 MAPBGA supply pins (continued)

Ball number	Ball name	Pad type	Ball number	Ball name	Pad type
R9	VDD_HV_ADR_02	VDD_HV_A	M9	VDD_LV_COR	VDD_LV
U9	VDD_HV_ADV	VDD_HV_A	M10	VDD_LV_COR	VDD_LV
F6	VDD_LV_COR	VDD_LV	M11	VDD_LV_COR	VDD_LV
F7	VDD_LV_COR	VDD_LV	M12	VDD_LV_COR	VDD_LV
F8	VDD_LV_COR	VDD_LV	P4	VDD_LV_PLL	VDD_LV
V_{SS}					
A1	VSS_HV_IO	VSS_HV	G7	VSS_LV_COR	VSS_LV
A2	VSS_HV_IO	VSS_HV	G8	VSS_LV_COR	VSS_LV
A16	VSS_HV_IO	VSS_HV	G9	VSS_LV_COR	VSS_LV
A17	VSS_HV_IO	VSS_HV	G10	VSS_LV_COR	VSS_LV
B1	VSS_HV_IO	VSS_HV	G11	VSS_LV_COR	VSS_LV
B2	VSS_HV_IO	VSS_HV	H7	VSS_LV_COR	VSS_LV
B9	VSS_HV_IO	VSS_HV	H8	VSS_LV_COR	VSS_LV
B17	VSS_HV_IO	VSS_HV	H9	VSS_LV_COR	VSS_LV
C3	VSS_HV_IO	VSS_HV	H10	VSS_LV_COR	VSS_LV
D15	VSS_HV_IO	VSS_HV	H11	VSS_LV_COR	VSS_LV
H2	VSS_HV_IO	VSS_HV	J7	VSS_LV_COR	VSS_LV
N2	VSS_HV_IO	VSS_HV	J8	VSS_LV_COR	VSS_LV
P9	VSS_HV_IO	VSS_HV	J9	VSS_LV_COR	VSS_LV
R3	VSS_HV_IO	VSS_HV	J10	VSS_LV_COR	VSS_LV
R15	VSS_HV_IO	VSS_HV	J11	VSS_LV_COR	VSS_LV
T1	VSS_HV_IO	VSS_HV	K7	VSS_LV_COR	VSS_LV
T17	VSS_HV_IO	VSS_HV	K8	VSS_LV_COR	VSS_LV
U1	VSS_HV_IO	VSS_HV	K9	VSS_LV_COR	VSS_LV
U2	VSS_HV_IO	VSS_HV	K10	VSS_LV_COR	VSS_LV
U16	VSS_HV_IO	VSS_HV	K11	VSS_LV_COR	VSS_LV
U17	VSS_HV_IO	VSS_HV	L7	VSS_LV_COR	VSS_LV
D9	VSS_HV_FLA	VSS_HV	L8	VSS_LV_COR	VSS_LV
P1	VSS_HV_OSC	VSS_HV	L9	VSS_LV_COR	VSS_LV
C15	VSS_HV_PDI	VSS_HV	L10	VSS_LV_COR	VSS_LV
J16	VSS_HV_PDI	VSS_HV	L11	VSS_LV_COR	VSS_LV
T9	VSS_HV_ADR_02	VSS_HV_A	N4	VSS_LV_PLL	VSS_LV
T7	VSS_HV_ADR_13	VSS_HV_A	U15	VSS_HV_PMU	VSS_LV
U10	VSS_HV_ADV	VSS_HV_A			

Table 4. 257 MAPBGA pins not populated on package

E5	E6	E7	E8	E9	E10	E11	E12
E13	F5	F13	G5	G13	H5	H13	J5
J13	K5	K13	L5	L13	M5	M13	N5
N6	N7	N8	N9	N10	N11	N12	N13

Table 5. 473 MAPBGA supply pins

Ball number	Ball name	Pad type	Ball number	Ball name	Pad type
V_{DD}					
A3	VDD_HV_IO	VDD_HV	F15	VDD_LV_COR	VDD_LV
A14	VDD_HV_IO	VDD_HV	F16	VDD_LV_COR	VDD_LV
B22	VDD_HV_IO	VDD_HV	F17	VDD_LV_COR	VDD_LV
C1	VDD_HV_IO	VDD_HV	F18	VDD_LV_COR	VDD_LV
D8	VDD_HV_IO	VDD_HV	G6	VDD_LV_COR	VDD_LV
G2	VDD_HV_IO	VDD_HV	G18	VDD_LV_COR	VDD_LV
L20	VDD_HV_IO	VDD_HV	H6	VDD_LV_COR	VDD_LV
M2	VDD_HV_IO	VDD_HV	H18	VDD_LV_COR	VDD_LV
M4	VDD_HV_IO	VDD_HV	J6	VDD_LV_COR	VDD_LV
T4	VDD_HV_IO	VDD_HV	J18	VDD_LV_COR	VDD_LV
V2	VDD_HV_IO	VDD_HV	K6	VDD_LV_COR	VDD_LV
Y13	VDD_HV_IO	VDD_HV	K18	VDD_LV_COR	VDD_LV
Y20	VDD_HV_IO	VDD_HV	L6	VDD_LV_COR	VDD_LV
AB2	VDD_HV_IO	VDD_HV	L18	VDD_LV_COR	VDD_LV
AB22	VDD_HV_IO	VDD_HV	M6	VDD_LV_COR	VDD_LV
AC12	VDD_HV_ADR_0	VDD_HV_A	M18	VDD_LV_COR	VDD_LV
AC15	VDD_HV_ADR_1	VDD_HV_A	N6	VDD_LV_COR	VDD_LV
AC7	VDD_HV_ADR_23	VDD_HV_A	N18	VDD_LV_COR	VDD_LV
AA9	VDD_HV_ADV	VDD_HV_A	P6	VDD_LV_COR	VDD_LV
H22	VDD_HV_DRAM	VDD_HV	P18	VDD_LV_COR	VDD_LV
L23	VDD_HV_DRAM	VDD_HV	R6	VDD_LV_COR	VDD_LV
P23	VDD_HV_DRAM	VDD_HV	R18	VDD_LV_COR	VDD_LV
U22	VDD_HV_DRAM	VDD_HV	T6	VDD_LV_COR	VDD_LV
R20	VDD_HV_DRAM_VREF	VDD_HV	T18	VDD_LV_COR	VDD_LV
H21	VDD_HV_DRAM_VTT	VDD_HV	U6	VDD_LV_COR	VDD_LV
L21	VDD_HV_DRAM_VTT	VDD_HV	U18	VDD_LV_COR	VDD_LV

Table 5. 473 MAPBGA supply pins (continued)

Ball number	Ball name	Pad type	Ball number	Ball name	Pad type
H11	VSS_LV_COR	VSS_LV	R16	VSS_LV_COR	VSS_LV
H12	VSS_LV_COR	VSS_LV	R17	VSS_LV_COR	VSS_LV
H13	VSS_LV_COR	VSS_LV	T7	VSS_LV_COR	VSS_LV
H14	VSS_LV_COR	VSS_LV	T8	VSS_LV_COR	VSS_LV
H15	VSS_LV_COR	VSS_LV	T9	VSS_LV_COR	VSS_LV
H16	VSS_LV_COR	VSS_LV	T10	VSS_LV_COR	VSS_LV
H17	VSS_LV_COR	VSS_LV	T11	VSS_LV_COR	VSS_LV
J7	VSS_LV_COR	VSS_LV	T12	VSS_LV_COR	VSS_LV
J8	VSS_LV_COR	VSS_LV	T13	VSS_LV_COR	VSS_LV
J9	VSS_LV_COR	VSS_LV	T14	VSS_LV_COR	VSS_LV
J10	VSS_LV_COR	VSS_LV	T15	VSS_LV_COR	VSS_LV
J11	VSS_LV_COR	VSS_LV	T16	VSS_LV_COR	VSS_LV
J12	VSS_LV_COR	VSS_LV	T17	VSS_LV_COR	VSS_LV
J13	VSS_LV_COR	VSS_LV	U7	VSS_LV_COR	VSS_LV
J14	VSS_LV_COR	VSS_LV	U8	VSS_LV_COR	VSS_LV
J15	VSS_LV_COR	VSS_LV	U9	VSS_LV_COR	VSS_LV
J16	VSS_LV_COR	VSS_LV	U10	VSS_LV_COR	VSS_LV
J17	VSS_LV_COR	VSS_LV	U11	VSS_LV_COR	VSS_LV
K7	VSS_LV_COR	VSS_LV	U12	VSS_LV_COR	VSS_LV
K8	VSS_LV_COR	VSS_LV	U13	VSS_LV_COR	VSS_LV
K9	VSS_LV_COR	VSS_LV	U14	VSS_LV_COR	VSS_LV
K10	VSS_LV_COR	VSS_LV	U15	VSS_LV_COR	VSS_LV
K11	VSS_LV_COR	VSS_LV	U16	VSS_LV_COR	VSS_LV
K12	VSS_LV_COR	VSS_LV	U17	VSS_LV_COR	VSS_LV
K13	VSS_LV_COR	VSS_LV	W4	VSS_LV_PLL	VSS_LV
K14	VSS_LV_COR	VSS_LV	AC19	VSS_HV_PMU	VSS_LV
K15	VSS_LV_COR	VSS_LV	D5	RESERVED	VSS_HV
K16	VSS_LV_COR	VSS_LV	AB20	RESERVED	VSS_HV
K17	VSS_LV_COR	VSS_LV			

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
J17	GPIO	flexpwm0 X[1]	A0: siul_GPIO[195] A1: flexpwm0_X[1] A2: ebi_AD29 A3: _	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
K1	GPIO	nexus MSEO_B[0][1]	A0: siul_GPIO[89] A1: _ A2: npc_wrapper_MSEO_B[0] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
K2	GPIO	nexus MSEO_B[1][1]	A0: siul_GPIO[88] A1: _ A2: npc_wrapper_MSEO_B[1] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
K3	GPIO	nexus RDY_B	A0: siul_GPIO[216] A1: _ A2: nexus_RDY_B A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
K4	GPIO	dspi0 SIN	A0: siul_GPIO[39] A1: _ A2: _ A3: sscm_DEBUG[7]	I: dspi0_SIN I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
K14	GPIO	flexpwm0 X[2]	A0: siul_GPIO[196] A1: flexpwm0_X[2] A2: ebi_AD30 A3: _	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
K15	GPIO	flexpwm0 X[3]	A0: siul_GPIO[197] A1: flexpwm0_X[3] A2: ebi_AD31 A3: _	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
K16	GPIO	flexpwm0 A[1]	A0: siul_GPIO[149] A1: _ A2: ebi_RD_WR A3: flexpwm0_A[1]	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_IO
K17	GPIO	flexpwm0 B[0]	A0: siul_GPIO[148] A1: _ A2: ebi_CLKOUT A3: flexpwm0_B[0]	I: _ I: _ I: _	—	disabled	DRAM ACC	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 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
G1	GPIO	nexus MCKO	A0: siul_GPIO[87] A1: _ A2: npc_wrapper_MCKO A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
G3	GPIO	nexus MDO[8][1]	A0: siul_GPIO[111] A1: _ A2: npc_wrapper_MDO[8] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
G4	GPIO	nexus MSEO_B[1][1]	A0: siul_GPIO[88] A1: _ A2: npc_wrapper_MSEO_B[1] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
G20	GPIO	siul GPIO[196]	A0: siul_GPIO[196] A1: flexpwm0_X[2] A2: ebi_AD30 A3: _	I: _ I: _ I: _	—	disabled	DRAM ACC	VDD_HV_DRAM
G21	GPIO	dramc DQS[0]	A0: siul_GPIO[190] A1: dramc_DQS[0] A2: ebi_AD24 A3: _	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM
G22	GPIO	dramc DM[0]	A0: siul_GPIO[192] A1: dramc_DM[0] A2: ebi_AD26 A3: _	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM
G23	GPIO	dramc D[7]	A0: siul_GPIO[181] A1: dramc_D[7] A2: ebi_AD15 A3: ebi_ADD31	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM
H1	GPIO	nexus EVTO_B	A0: siul_GPIO[90] A1: _ A2: npc_wrapper_EVTO_B A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
H3	GPIO	nexus MSEO_B[0][1]	A0: siul_GPIO[89] A1: _ A2: npc_wrapper_MSEO_B[0] A3: _	I: _ I: _ I: _	—	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
J23	GPIO	dramc D[6]	A0: siul_GPIO[180] A1: dramc_D[6] A2: ebi_AD14 A3: ebi_ADD30	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM
K1	GPIO	dspi0 SCK	A0: siul_GPIO[37] A1: dspi0_SCK A2: _ A3: sscm_DEBUG[5]	I: flexpwm0_FAULT[3] I: _ I: siul_EIRQ[23]	—	disabled	GP Slow/ Medium	VDD_HV_IO
K2	GPIO	dspi1 CS0	A0: siul_GPIO[5] A1: dspi1_CS0 A2: _ A3: dspi0_CS7	I: _ I: _ I: siul_EIRQ[5]	—	disabled	GP Slow/ Medium	VDD_HV_IO
K3	GPIO	dspi1 SCK	A0: siul_GPIO[6] A1: dspi1_SCK A2: _ A3: _	I: _ I: _ I: siul_EIRQ[6]	—	disabled	GP Slow/ Medium	VDD_HV_IO
K4	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
K21	GPIO	dramc D[4]	A0: siul_GPIO[178] A1: dramc_D[4] A2: ebi_AD12 A3: ebi_ADD28	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM
K22	GPIO	dramc D[8]	A0: siul_GPIO[182] A1: dramc_D[8] A2: ebi_AD16 A3: _	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM
K23	GPIO	dramc D[9]	A0: siul_GPIO[183] A1: dramc_D[9] A2: ebi_AD17 A3: _	I: _ I: _ I: _	—	disabled	DRAM DQ	VDD_HV_DRAM
L1	GPIO	dspi0 CS0	A0: siul_GPIO[36] A1: dspi0_CS0 A2: _ A3: sscm_DEBUG[4]	I: _ I: _ I: siul_EIRQ[22]	—	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
AB4	GPIO	flexpwm1 X[2]	A0: siul_GPIO[122] A1: flexpwm1_X[2] A2: etimer2_ETC[2] A3: dspi0_CS5	I: _ I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
AB5	GPIO	flexpwm1 X[3]	A0: siul_GPIO[125] A1: flexpwm1_X[3] A2: etimer2_ETC[3] A3: dspi0_CS6	I: _ I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
AB6	ANA	adc3 AN[2]	—	siul_GPI[231]	AN: adc3_AN[2]	—	Analog	VDD_HV_ADR23
AB7	ANA	adc2_adc3 AN[13]	—	siul_GPI[227]	AN: adc2_adc3_AN[13]	—	Analog Shared	VDD_HV_ADR23
AB8	ANA	adc2 AN[1]	—	siul_GPI[222]	AN: adc2_AN[1]	—	Analog	VDD_HV_ADR23
AB9	ANA	adc2 AN[2]	—	siul_GPI[223]	AN: adc2_AN[2]	—	Analog	VDD_HV_ADR23
AB10	ANA	adc0 AN[0]	—	siul_GPI[23] lin0_RXD	AN: adc0_AN[0]	—	Analog	VDD_HV_ADR0
AB11	ANA	adc0 AN[4]	—	siul_GPI[70]	AN: adc0_AN[4]	—	Analog	VDD_HV_ADR0
AB12	ANA	adc0 AN[6]	—	siul_GPI[71]	AN: adc0_AN[6]	—	Analog	VDD_HV_ADR0
AB13	ANA	adc0 AN[7]	—	siul_GPI[68]	AN: adc0_AN[7]	—	Analog	VDD_HV_ADR0
AB14	ANA	adc0_adc1 AN[13]	—	siul_GPI[27]	AN: adc0_adc1_AN[13]	—	Analog Shared	VDD_HV_ADR0

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
AB15	ANA	adc1 AN[1]	—	siul_GPI[30] etimer0_ETC[4] siul_EIRQ[19]	AN: adc1_AN[1]	—	Analog	VDD_HV_ADR1
AB16	ANA	adc1 AN[3]	—	siul_GPI[32]	AN: adc1_AN[3]	—	Analog	VDD_HV_ADR1
AB17	ANA	adc1 AN[4]	—	siul_GPI[75]	AN: adc1_AN[4]	—	Analog	VDD_HV_ADR1
AB18	GPIO	TDO	A0: siul_GPIO[20] A1: jtagc_TDO A2: _ A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Fast	VDD_HV_IO
AB21	GPIO	lin1 RXD	A0: siul_GPIO[95] A1: _ A2: i2c1_data A3: _	I: lin1_RXD I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
AC3	GPIO	dspi2 SIN	A0: siul_GPIO[13] A1: _ A2: _ A3: _	I: dspi2_SIN I: flexpwm0_FAULT[0] I: siul_EIRQ[12]	—	disabled	GP Slow/ Medium	VDD_HV_IO
AC4	GPIO	flexpwm1 A[3]	A0: siul_GPIO[126] A1: flexpwm1_A[3] A2: etimer2_ETC[4] A3: dspi0_CS7	I: _ I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
AC5	GPIO	flexpwm1 B[3]	A0: siul_GPIO[127] A1: flexpwm1_B[3] A2: etimer2_ETC[5] A3: _	I: _ I: _ I: _	—	disabled	GP Slow/ Medium	VDD_HV_IO
AC6	ANA	adc3 AN[3]	—	siul_GPI[232]	AN: adc3_AN[3]	—	GP Slow/ Medium	VDD_HV_ADR23
AC9	ANA	adc2 AN[3]	—	siul_GPI[224]	AN: adc2_AN[3]	—	Analog	VDD_HV_ADR23

Table 18. VRC SMPS recommended external devices

Reference designator	Part description	Part type	Nominal	Description
Ca	—	capacitor	20 µF, 20 V	Filter capacitor
Cb	—	capacitor	0.1 µF, 20 V	Filter capacitor
Cd	—	capacitor	20 µF, 20 V	Supply decoupling cap, ESR < 50 mΩ, as close to PMOS source as possible
Ce	—	capacitor	0.1 µF, 16 V	Ceramic
Cl	—	capacitor	20 µF, 16 V	Buck capacitor, total ESR < 100 mΩ, as close to the coil as possible
D	SS8P3L	Schottky	—	Vishay low Vf Schottky diode
L	—	inductor	4 µH, 1.5 A	Buck shielded coil low ESR
Q	FDC642P or SQ2301ES or SI3443DV	pMOS	2 A, 10 V	Low threshold PMOS V_{th} < 1.5 V, R_{dson}@4.5 V < 120 mΩ, Q_g < 16 nC
R	—	resistor	50–100 kΩ	Pullup for power PMOS gate

3.9 Supply current characteristics

Table 19. Current consumption characteristics[1]

No.	Symbol		Parameter	Conditions	Min	Typ	Max	Unit
1	I_{DD_LV}	CC	Maximum run I_{DD} (incl. digital core logic and analog block of the LV rail)	V_{DD_LV} = 1.36 V, f_{Core} = 180 MHz, 1:2 mode, DPM, both cores executing EMC test code, internal VREG mode, all caches enabled, code execution of core 0 from code flash 0, code execution of core 1 from code flash 1, FMPLL_1 active at 120 MHz.	—	600	900	mA
2	$I_{DD\_LV\_PLL}$	CC	Maximum run I_{DD} for each PLL[2]	$V_{DD\_LV\_PLL}$ = 1.36 V, f_{VCO} running at maximum frequency.	—	1.5	2	mA
3	$I_{DD\_HV\_FLA}$[3]	CC	Maximum run I_{DD} Flash	$V_{DD\_HV\_FLA}$ = 3.6 V, DPM, both cores executing EMC test code, code execution of core 0 from code flash 0, code execution of core 1 from code flash 1.	—	20	30	mA
4	$I_{DD\_HV\_OSC}$	CC	Maximum run I_{DD} OSC	f_{OSC} 4 MHz to 40 MHz, $V_{DD\_HV\_OSC}$ 3.6 V	—	1	3	mA
5	$I_{DD\_HV\_ADV}$	CC	Maximum run I_{DD} for each ADC[4]	$V_{DD\_HV\_ADV}$ = 3.6 V	—	2	4	mA
6	$I_{DD\_HV\_ADR02}$[5]	CC	Maximum reference I_{DD}[6]	ADC0 powered on[7]	—	—	2	mA
				ADC2 powered on	—	—	1.2	mA
7	$I_{DD\_HV\_ADR13}$[5]	CC	Maximum reference I_{DD}[6]	ADC1 powered on	—	—	1.2	mA
				ADC3 powered on	—	—	1.2	mA
8	$I_{DD\_HV\_ADR0}$[8]	CC	Maximum reference I_{DD}	ADC0 powered on[7]	—	—	2	mA

3.17.2 GP pads AC specifications

Table 34. GP pads AC electrical characteristics[1]

No.	Pad	Tswitchon[1] (ns)			Rise/Fall[2] (ns)			Frequency (MHz)			Current slew[3] (mA/ns)			Load drive (pF)
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
1	Slow	3	—	40	4	—	40	—	—	4	0.01	—	2	25
		3	—	40	6	—	50	—	—	2	0.01	—	2	50
		3	—	40	10	—	75	—	—	2	0.01	—	2	100
		3	—	40	14	—	100	—	—	2	0.01	—	2	200
2	Medium	1	—	15	2	—	12	—	—	40	2.5	—	7	25
		1	—	15	4	—	25	—	—	20	2.5	—	7	50
		1	—	15	8	—	40	—	—	13	2.5	—	7	100
		1	—	15	14	—	70	—	—	7	2.5	—	7	200
3	Fast	1	—	6	1	—	4	—	—	72	3	—	40	25
		1	—	6	1.5	—	7	—	—	55	7	—	40	50
		1	—	6	3	—	12	—	—	40	7	—	40	100
		1	—	6	5	—	18	—	—	25	7	—	40	200
4	Symmetric	1	—	8	1	—	5	—	—	50	3	—	25	25
5	Pullup/down (3.6 V max)	—	—	—	—	—	7500	—	—	—	—	—	—	50

1 The values provided in this table are not applicable for PDI and EBI/DRAM interface.

2 Slope at rising/falling edge.

3 Data based on characterization results, not tested in production.

3.18 PDI pads specifications

This section specifies the electrical characteristics of the PDI pads. Please refer to the tables in Section 2.2, Pin descriptions, for a cross reference between package pins and pad types.

PDI pads feature list:

- Direction
  - Input
  - Output
  - Bidirectional
- Driver
  - Push/Pull/Open Drain
  - Configurable Four Drive Strengths on Fast driver pads
  - Configurable No Slew-Rate, Slow Slew-Rate, and Fast Slew-Rate on Slow, Medium, and SLR driver pads
  - VDD_HV_PDI NOTE: All pads are NOT 5 V TOLERANT. Pads are not capable of driving to or from voltages above their respective VDD_HV_PDI. In other words, you cannot connect a 3.3V external device to a pad

Table 53. Reset sequence trigger—reset sequence

Reset Sequence Trigger	VREG Mode[1]	Reset Sequence Start Condition	Reset Sequence End Indication	Reset Sequence				
				Destructive Reset Sequence, BIST enabled[2]	*Destructive Reset Sequence, BIST disabled*[2]	*External Reset Sequence Long, BIST enabled*	*Functional Reset Sequence Long*	*Functional Reset Sequence Short*
All active internal destructive reset sources (LVDs or internal HVD during power-up and during operation)	I	Section 3.21.4.1, Internal VREG mode	Release of $\overline{\text{RESET}}$[3]	triggers		cannot trigger	cannot trigger	cannot trigger
	E	Section 3.21.4.2, External VREG mode				cannot trigger	cannot trigger	cannot trigger
Assertion of $\overline{\text{RESET\_SUP}}$[4]								
Assertion of $\overline{\text{RESET}}$[5]	I/E	Section 3.21.4.3, External reset via RESET		cannot trigger		triggers[6]	triggers[7]	triggers[8]
All internal functional reset sources configured for long reset	I/E	Sequence starts with internal reset trigger	Release of $\overline{\text{RESET}}$[9]	cannot trigger		cannot trigger	triggers	cannot trigger
All internal functional reset sources configured for short reset	I/E			cannot trigger		cannot trigger	cannot trigger	triggers

1 VREG Mode: I = Internal VREG Mode, E = External VREG Mode.

2 Whether BIST is executed or not depends on device configuration data stored in the shadow sector of the NVM.

3 End of the internal reset sequence (as specified in Table 52) can only be observed by release of $\overline{\text{RESET}}$ if it is not held low externally beyond the end of the internal sequence which would prolong the internal reset PHASE3 until $\overline{\text{RESET}}$ is released externally.

4 In external VREG mode only.

5 The assertion of $\overline{\text{RESET}}$ can only trigger a reset sequence if the device was running ($\overline{\text{RESET}}$ released) before. $\overline{\text{RESET}}$ does not gate a *Destructive Reset Sequence, BIST enabled* or a *Destructive Reset Sequence, BIST disabled*. However, it can prolong these sequences if $\overline{\text{RESET}}$ is held low externally beyond the end of the internal sequence (beyond PHASE3).

6 If $\overline{\text{RESET}}$ is configured for long reset (default) and if BIST is enabled via device configuration data stored in the shadow sector of the NVM.

7 If $\overline{\text{RESET}}$ is configured for long reset (default) and if BIST is disabled via device configuration data stored in the shadow sector of the NVM.

Table 60. 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.

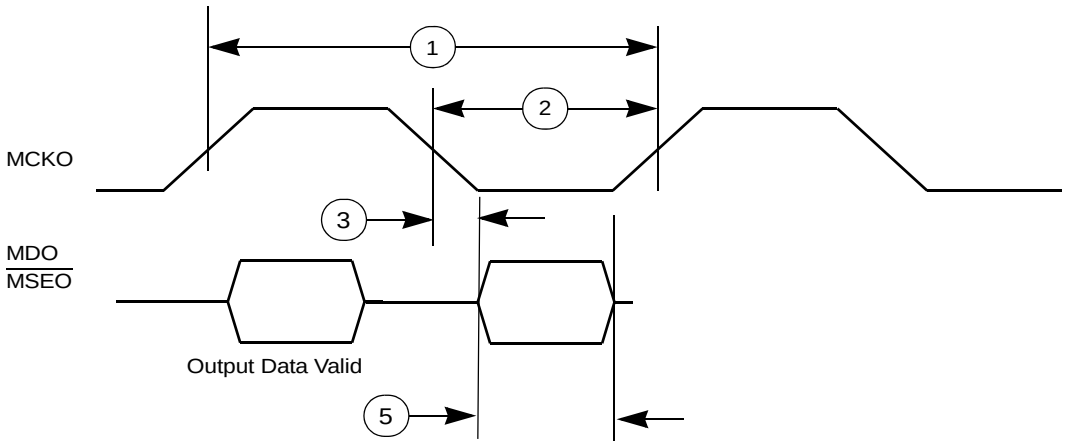

Figure 31. Nexus DDR mode timing

3.22.4 External interrupt timing (IRQ pins)

Table 61. 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 62. 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}

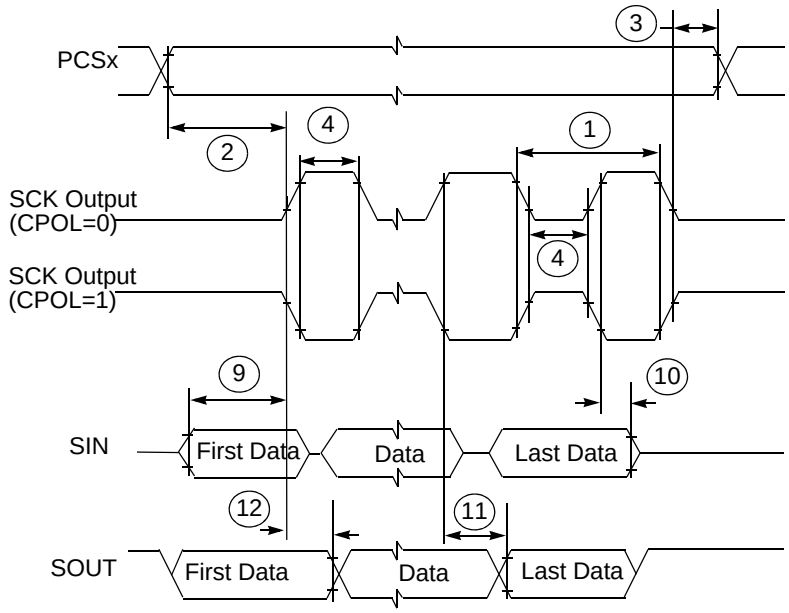

Figure 37. DSPI modified transfer format timing—master, CPHA = 0

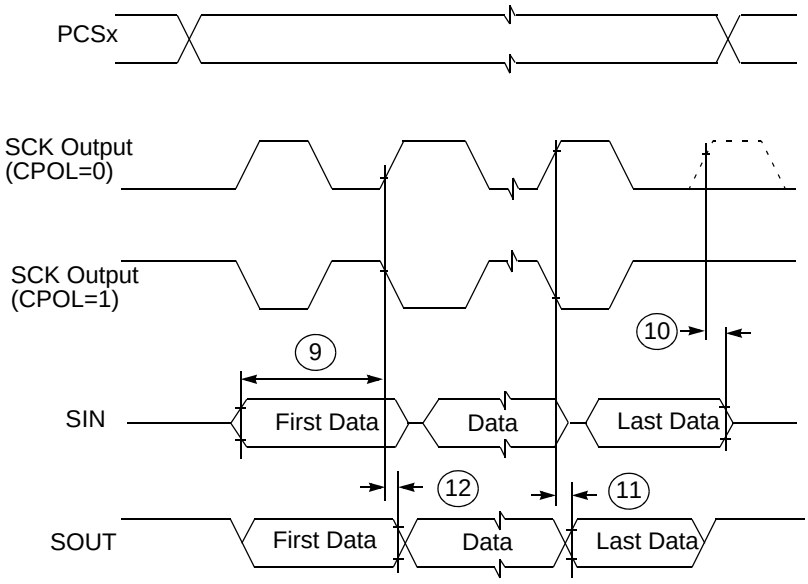

Figure 38. DSPI modified transfer format timing—master, CPHA = 1

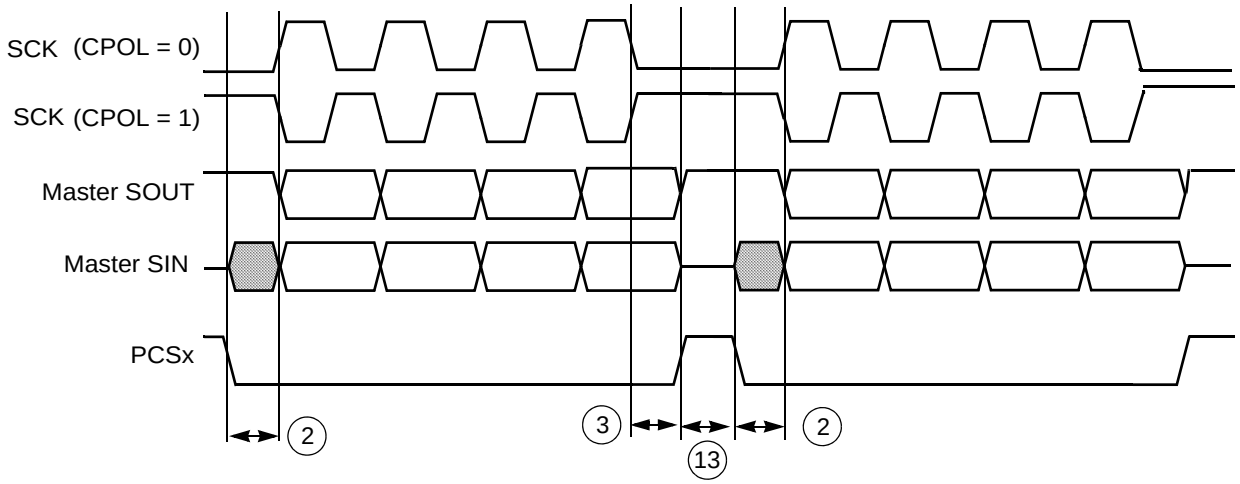

Figure 41. Example of non-continuous format (CPHA = 1, CONT = 0)

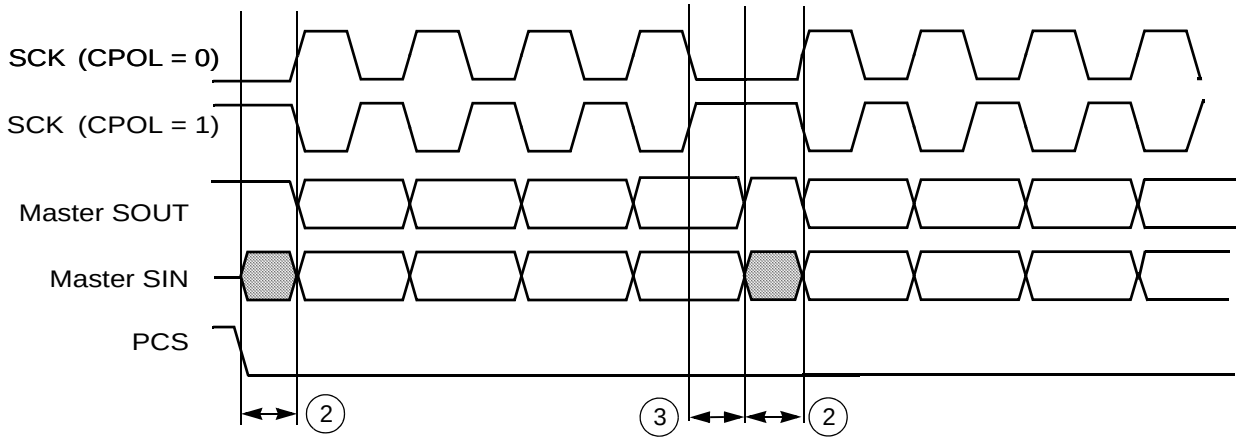

Figure 42. Example of continuous transfer (CPHA = 1, CONT = 1)

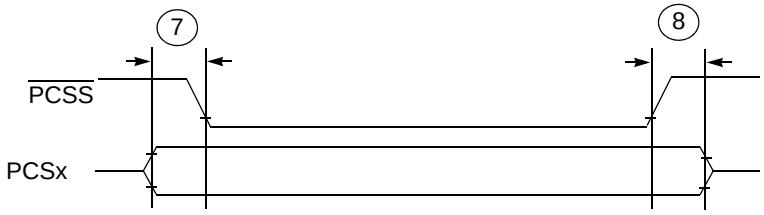

Figure 43. DSPI PCS strobe ($\overline{\text{PCSS}}$) timing

3.22.7 PDI timing

Table 65. PDI electrical characteristics

No.	Symbol		Parameter	Conditions	Min	Max	Unit
1	t_{PDI_CLOCK}	SR	PDI clock period	—	15	—	ns

4 Package characteristics

4.1 Package mechanical data

4.1.1 257 MAPBGA

4.1.2 473 MAPBGA

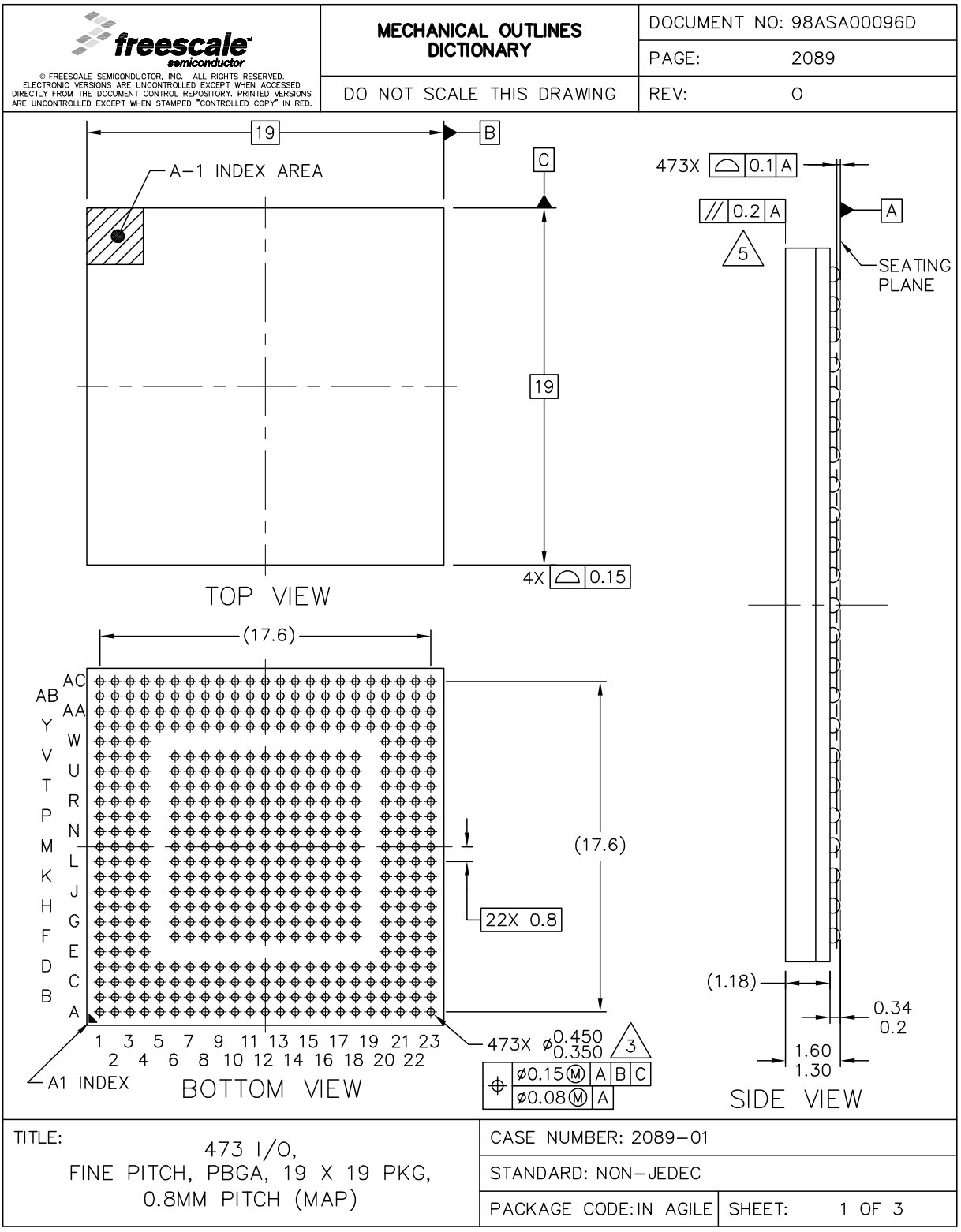

Figure 56. 473 MAPBGA package mechanical data (1 of 3)

Table 73. Revision history (continued)

Revision	Date	Description of Changes
7 (cont.)	18 May 2012	In Section 3.18, PDI pads specifications, Table 36 (PDI pads DC electrical characteristics), added footnote to table: “Over- and undershoots occurring due to impedance mismatch of the external driver and the transmission line at PDI pads in input mode can be allowed up to 0.7 V repeatedly throughout the product expected lifetime and will not cause any long term reliability issue.” In Section 5, Orderable parts, • Removed “3 = 220 MHz” under Operating frequency heading and changed the Operating frequency of the example from “3” to “2”. • Deleted Table 73 (Orderable part number summary**)**.