MSP432P401R Device Erratasheet

1 Functional Errata Revision History

Errata impacting device's operation, function or parametrics.

✓ The check mark indicates that the issue is present in the specified revision.

Errata Number	Rev D	Rev C
BSL16		✓
COMP11	✓	✓
PORT31	✓	✓
PORT32	✓	✓
PORT34	✓	✓
PSS4	✓	✓
RTC14	✓	✓
SYS26		✓
TA23	✓	✓
USCI42	✓	✓
USCI43	✓	✓
USCI44	✓	✓
USCI45	✓	✓
USCI46	✓	✓
USCI47	✓	✓
USCI48		✓
USCI50	✓	✓
USCI51	✓	✓

2 Preprogrammed Software Errata Revision History

Errata impacting pre-programmed software into the silicon by Texas Instruments.

✓ The check mark indicates that the issue is present in the specified revision.

Errata Number	Rev D	Rev C
UDMA2	✓	✓

3 Debug only Errata Revision History

Errata only impacting debug operation.

✓ The check mark indicates that the issue is present in the specified revision.

The device doesn't have Debug errata.

Copyright © 2014–2019, Texas Instruments Incorporated

4 Fixed by Compiler Errata Revision History

Errata completely resolved by compiler workaround. Refer to specific erratum for IDE and compiler versions with workaround.

✓ The check mark indicates that the issue is present in the specified revision.

The device doesn't have Compiler-Fixed errata.

Copyright © 2014–2019, Texas Instruments Incorporated

5 Package Markings

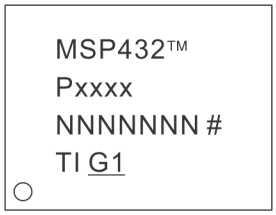
PZ100

LQFP (PZ) 100 Pin

\# = Die revision
○ = Pin 1 location
N = Lot trace code

ZXH80

BGA, 80 Pin



\# = Die revision
○ = Pin 1 location
N = Lot trace code

RGC64

QFN (RGC), 64 pin

\# = Die revision
○ = Pin 1 location
N = Lot trace code

Copyright © 2014–2019, Texas Instruments Incorporated

6 Detailed Bug Description

BSL16 *BSL Module*

Category Functional

Function On blank devices, debugger access over JTAG is disabled after the timeout window

Description On blank devices, the default device initialization sequence that resides in the boot strap loader (BSL) disables the JTAG pins after a 10 second timeout window if no activity has occurred. This causes debug and program access via JTAG to be disabled. JTAG access can be re-enabled via a device reset (either Class 0 or Class 1).

Workaround 1. Use Serial Wire Debug (SWD) as the debug/program interface instead of JTAG.

OR

2. If timeout occurs, reset device (either Class 0 or Class 1) and ensure JTAG is accessed within a 10 second window

COMP11 *COMP_E Module*

Category Functional

Function Polling comparator interrupts may result in a missed interrupt

Description Polling the comparator output interrupt and the output inverted interrupt flags (CEIFG.CExINT and CEIIFG.CExINT) may result in a missed interrupt if the flags are modified while being read.

Workaround Using an interrupt service routine to service CEIFG and CEIIFG interrupts significantly reduces the probability of the issue occurring.

PORT31 *PORT Module*

Category Functional

Function Fast transient noise on GPIOs may result in a constant high current

Description GPIOs subject to fast transient noise (e.g. electromagnetic noise) may see a high constant current. The constant high current is a result of the fast transient noise triggering the internal ESD protection structure and it persists as long as the internal or external current driver sustains the current.

1. When using in Input mode (PxDIR = 0), all GPIOs are impacted by this issue. A fast transient noise on the pin configured as an input or on an adjacent pin could cause a constant high current that is sustained as long as the external driver is present.

2. When using in Output mode (PxDIR = 1), only the high drive GPIOs (P2.0,P2.1,P2.2 and P2.3) are impacted by this issue. If the affected GPIOs are configured in high drive mode (PxDS register) and they (or adjacent pins) are subject to fast transient noise, it could cause a constant high current. Note that this issue is not seen in GPIOs configured in the output direction with the regular drive strength setting since the high drive mode is required to sustain the high current.

If the GPIO configuration is reset by a power cycle, the constant high current is no longer sustained.

Workaround 1. For GPIOs configured as input, ensure that they are driven by a current-limited source < 30mA (up to Tj=85C) or 20mA (up to Tj=125C, if allowed for device. See device

Copyright © 2014–2019, Texas Instruments Incorporated

specific datasheet for maximum allowed operating junction temperature) OR use a series resistance >100 ohm (up to Tj=85C) or 150 ohm (up to Tj=125C, if allowed for device. See device specific datasheet for maximum allowed operating junction temperature) to limit the current.

2. For high drive GPIOs configured as output, ensure adequate protection from fast transients is provided to both the high drive IOs and any adjacent pins.

3. In general, it is recommended to terminate any unused GPIOs in the output direction, driving low to minimize the occurrence of this issue.

4. For guidelines on ESD considerations refer to the document: MSP430 System-level ESD Considerations SLAA530

PORT32 *PORT Module*

Category Functional

Function Sucessive writes to port registers may cause interrupt to pend incorrectly

Description Writing to the PxIES register sets the corresponding PxIFG bit. The PxIFG bit can be cleared by writing '0' to it (clearing the register).

However if the PxIFG bit is cleared immediately (next instruction cycle) after writing to the PxIES register, the interrupt flag does not clear and stays pending.

Workaround Insert a NOP or __no_operation(); instruction after writing to the PxIES register and before clearing the PxIFG register.

PORT34 *PORT Module*

Category Functional

Function Timer_A1 & A2 outputs and EUSCIB3 pins should not be used simultaneously

Description The following pin functions cannot be simultaneously active on the device at any pin:

1) TA1.0 and UCB3STE

2) TA2.0 and UCB3CLK

3) TA2.3 and UCB3SIMO

4) TA2.4 and UCB3SOMI

Attempting to use both pin's secondary functions will cause functional conflicts and abnormal behavior of the peripherals. For example, using Timer_A2.3 output will prevent proper operation of EUSCIB3SIMO.

Workaround None.

PSS4 *PSS Module*

Category Functional

Function SVSMHLP has no impact in LPM4.5 mode

Description Upon entry into LPM4.5, the SVSMH is set to low power normal performance mode automatically regardless of the SVSMHLP setting.

Clearing the SVSMHLP bit (PSSCTL0.SVSMHLP = 0) has no impact in LPM4.5.

This bit works as expected in all other low power modes.

Copyright © 2014–2019, Texas Instruments Incorporated

Workaround None

RTC14 *RTC_C Module*

Category Functional

Function Polling RTC interrupts may result in a lost interrupt flag

Description Polling any RTC interrupt flag mapped to the RTCIV register may result in a missed interrupt if the flags are modified while being read.

Workaround Using an interrupt service routine to service flags mapped to the RTCIV register significantly reduces the probability of the issue occurring.

SYS26 *SYS Module*

Category Functional

Function IP Protection Feature not available

Description The Regional IP Protection feature is not available up to Revision C for the MSP432P401R/M devices.

If your application only requires MSP432 full-chip security to protect your software IP from JTAG read-out, this limitation does not apply to you.

Regional IP Protection is used in (i) protecting a specific block of code/data from readout by the rest of the application code and (ii) multi-party firmware development, where each software IP owner delivers an executable IP that is protected from read-out by code from other IP owners using the device.

For more information on benefits and how to use Regional IP Protection, refer to the document: Software IP Protection on MSP432P4xx Microcontrollers

Workaround None

TA23 *TIMER_A Module*

Category Functional

Function Polling timer interrupts may result in a lost interrupt flag

Description Polling any Timer interrupt flag may result in a missed interrupt if the flags are modified while being read.

Workaround Using an interrupt service routine to service timer interrupt flags significantly reduces the probability of the issue occurring.

UDMA2 *DMA Module*

Category Software in ROM

Function UDMA driver library versions < 4.20.00.03 (SimpleLink MSP432P4 SDK versions < 2.10.00.14) do not support use case where increment size and transfer size are different.

Description UDMA driver library APIs with versions < 4.20.00.03 (SimpleLink MSP432P4 SDK versions < 2.10.00.14) do not support use cases where increment size and transfer sizes are different for either the source or destination. The addresses computation are performed wrongly and this result in erroneous data transfers.

Copyright © 2014–2019, Texas Instruments Incorporated

Workaround This is fixed in driver library API versions >= 4.20.00.03 (SimpleLink MSP432P4 SDK versions >= 2.10.00.14).

- For TI Drivers users, update to SDK version 2.10.00.14 and TI Drivers based applications automatically leverage the bug fix.

- For Driver Library users, update to SDK version 2.10.00.14 and ensure that the application code uses MAP_DMA_ function calls for Driverlib DMA [instead of direct DMA_ API calls].

USCI42 *eUSCI Module*

Category Functional

Function UART asserts UCTXCPTIFG after each byte in multi-byte transmission

Description UCTXCPTIFG flag is triggered at the last stop bit of every UART byte transmission, independently of an empty buffer, when transmitting multiple byte sequences via UART. The erroneous UART behavior occurs with and without DMA transfer.

Workaround None.

USCI43 *eUSCI Module*

Category Functional

Function I2C communication stalled when polling UCBxRXIFG

Description When using the USCI_B I2C module as a receiver, if an asynchronous event occurs during the read of the UCBxRXIFG interrupt flag, the flag could be unintentionally cleared. This may result in the I2C communication being stalled.

Workaround Workaround

1. If the device functions as an I2C master receiver, use synchronous clock sources for operation.

OR

2. Avoid polling UCBxRXIFG. Using the standard interrupt service routine to service the UCBxRXIFG interrupt flag significantly reduces the probability of this errata occurring. Avoid register access to UCBxCTLW0, UCBxSTATW, UCBxRXBUF, UCBxTXBUF, UCBxIFG, & UCBxIV while transmit or receive operation is ongoing and UCBxRXIFG or UCBxTXIFG is expected to be set.

OR

3. Use the clock low time-out select feature (UCCTLO.UCBxCTLW1) to enable a timeout window. In the event that the I2C communication is stalled, use the clock low time-out interrupt to reset the eUSCI module and re-initiate communication.

USCI44 *eUSCI Module*

Category Functional

Function Differing clock sources may cause UART communication failure

Description When using the USCI_A UART module with differing clock sources for the system clock (MCLK) and the UART source clock (BRCLK), any read or write of the UCAxIFG, UCAxCTLW0, UCAxSTATW, UCAxRXBUF, UCAxTXBUF, UCAxABCTL & UCAxIV registers, while the UCATXIFG or UCARXIFG flag is being set by a UART event could unintentionally clear the UCATXIFG or UCARXIFG. This may result in the UART

Copyright © 2014–2019, Texas Instruments Incorporated

communication being stalled.

Workaround

Workaround 1: Use synchronous clocks to source BRCLK and MCLK. Note that the clock frequencies need not be identical and dividers may be used as long as they are using the same clock source.

Workaround 2: Avoid polling UCAxTXIFG and UCAxRXIFG. Using the standard interrupt service routine to service the interrupt flag significantly reduces access to the USCI registers & hence reduces the probability of this errata occurring.

Also, limit all accesses of the UCAxCTLW0, UCAxSTATW, UCAxRXBUF, UCAxTXBUF, UCAxABCTL, UCAxIFG, & UCAxIV registers while transmit or receive operation is ongoing (and UCAxRXIFG or UCAxTXIFG is expected to be set) as this can further reduce the probability of this errata occurring.

USCI45 *eUSCI Module*

Category

Functional

Function

Unexpected SPI clock stretching possible when UCxCLK is asynchronous to MCLK

Description

In rare cases, during SPI communication, the clock high phase of the first data bit may be stretched significantly. The SPI operation completes as expected with no data loss. This issue only occurs when the USCI SPI module clock (UCxCLK) is asynchronous to the system clock (MCLK).

Workaround

Ensure that the USCI SPI module clock (UCxCLK) and the CPU clock (MCLK) are synchronous to each other.

USCI46 *eUSCI Module*

Category

Functional

Function

UART may receive the first byte incorrectly

Description

The USCI UART may receive the first byte incorrectly under the following conditions:

1. If the USCI UART is setup to source BRCLK from ACLK or SMCLK and the clock source is turned off, for example when the module is in idle condition and no other peripheral is requesting the same clock source

AND

2. The UART is configured for a baud rate of 9600 and the BRCLK is setup to source a 32kHZ clock (REFO or LFXT)

If the above two conditions are satisfied, the UART start byte received interrupt flag (UCSTTIFG) may not be set and the UART may miss the first bit resulting in an incorrect data byte. Subsequent data bytes are not impacted.

Workaround

1. When setting up the UART at 9600 baud, use a source clock >32kHz or if a 32kHz clock source must be used, lower the baud rate to 4800 baud

OR

2. Ensure any other peripheral (for example a timer sourced from ACLK) is active and using the UART clock source thereby keeping the clock turned on even when the UART module is idle.

USCI47 *eUSCI Module*

Copyright © 2014–2019, Texas Instruments Incorporated

Category Functional

Function eUSCI SPI slave with clock phase UCCKPH = 1

Description The eUSCI SPI operates incorrectly under the following conditions:

1. The eUSCI_A or eUSCI_B module is configured as a SPI slave with clock phase mode UCCKPH = 1

AND

2. The SPI clock pin is not at the appropriate idle level (low for UCCKPL = 0, high for UCCKPL = 1) when the UCSWRST bit in the UCxxCTLW0 register is cleared.

If both of the above conditions are satisfied, then the following will occur:

eUSCI_A: the SPI will not be able to receive a byte (UCAxRXBUF will not be filled and UCRXIFG will not be set) and SPI slave output data will be wrong (first bit will be missed and data will be shifted).

eUSCI_B: the SPI receives data correctly but the SPI slave output data will be wrong (first byte will be duplicated or replaced by second byte).

Workaround Use clock phase mode UCCKPH = 0 for MSP SPI slave if allowed by the application.

OR

The SPI master must set the clock pin at the appropriate idle level (low for UCCKPL = 0, high for UCCKPL = 1) before SPI slave is reset (UCSWRST bit is cleared).

OR

For eUSCI_A: to detect communication failure condition where UCRXIFG is not set, check both UCRXIFG and UCTXIFG. If UCTXIFG is set twice but UCRXIFG is not set, reset the MSP SPI slave by setting and then clearing the UCSWRST bit, and inform the SPI master to resend the data.

USCI48 *eUSCI Module*

Category Functional

Function SPI communication stalled when polling UCxxIFG.UCRXIFG

Description When using the USCI_A or USCI_B SPI module as a slave, if an asynchronous event occurs during the read of the UCRXIFG (UCxxIFG.UCRXIFG) interrupt flag, the flag could be unintentionally cleared. This may result in the SPI communication being stalled.

Workaround 1. If the device functions as a SPI slave, ensure that the USCI clock source is synchronous to the system clock (MCLK).

OR

2. Avoid polling UCxxIFG.UCRXIFG. Using the standard interrupt service routine to service the UCxxIFG.UCRXIFG interrupt flag significantly reduces the probability of this errata occurring. Avoid register access to UCxxCTLW0, UCxxSTATW, UCxxRXBUF, UCxxTXBUF, UCxxIFG, & UCxxIV while transmit or receive operation is ongoing and UCRXIFG is expected to be set.

USCI50 *eUSCI Module*

Category Functional

Function Data may not be transmitted correctly from the eUSCI when operating in SPI 4-pin master mode with UCSTEM = 0

Copyright © 2014–2019, Texas Instruments Incorporated

Description When the eUSCI is used in SPI 4-pin master mode with UCSTEM = 0 (STE pin used as an input to prevent conflicts with other SPI masters), data that is moved into UCxTXBUF while the UCxSTE input is in the inactive state may not be transmitted correctly. If the eUSCI is used with UCSTEM = 1 (STE pin used to output an enable signal), data is transmitted correctly.

Workaround When using the STE pin in conflict prevention mode (UCSTEM = 0), only move data into UCxTXBUF when UCxSTE is in the active state. If an active transfer is aborted by UCxSTE transitioning to the master-inactive state, the data must be rewritten into UCxTXBUF to be transferred when UCxSTE transitions back to the master-active state.

USCI51 *eUSCI Module*

Category Functional

Function UART could lose bytes while transmitting with DMA

Description Accessing the RXIFG (which is at the same address as the TXIFG) while transmitting with the DMA, could cause a second interrupt for the DMA and overwrite the transmit buffer, before moving to the Shift register.

Workaround Clear pending UART RX-interrupt flags with dummy read and enable RX-interrupt as the below example code:

```
volatile uint8_t temp;
temp =  EUSCI_A_CMSIS(EUSCI_A2_BASE)->RXBUF;
MAP_UART_enableInterrupt(EUSCI_A2_BASE, EUSCI_A_UART_RECEIVE_INTERRUPT);
```


Copyright © 2014–2019, Texas Instruments Incorporated

7 Document Revision History

Changes from device specific erratasheet to document Revision A.

1. Errata ADC48 was added to the errata documentation.
2. Errata ADC46 was added to the errata documentation.
3. Errata ADC49 was added to the errata documentation.
4. Errata FLASH38 was added to the errata documentation.
5. Errata ADC47 was added to the errata documentation.
6. Errata DMA12 was added to the errata documentation.

Changes from document Revision A to Revision B.

1. PSS1 Description was updated.
2. Errata REF8 was added to the errata documentation.
3. SYS21 Description was updated.
4. Errata RTC11 was added to the errata documentation.
5. Errata RST2 was added to the errata documentation.
6. SYS22 Description was updated.
7. PSS1 Workaround was updated.
8. Errata PCM1 was added to the errata documentation.
9. Errata CS8 was added to the errata documentation.
10. Errata USCI36 was removed from the errata documentation.
11. SYS21 Function was updated.
12. Errata FLASH39 was added to the errata documentation.
13. Errata ADC58 was added to the errata documentation.
14. SYS24 Description was updated.
15. SYS22 Function was updated.
16. SYS24 Function was updated.
17. Errata PORT25 was added to the errata documentation.
18. Errata PSS2 was added to the errata documentation.
19. Errata REF7 was added to the errata documentation.
20. Errata COMP8 was added to the errata documentation.
21. Silicon Revision B was added to the errata documentation.
22. Errata RST1 was added to the errata documentation.

Changes from document Revision B to Revision C.

1. Errata ADC61 was added to the errata documentation.
2. Errata BSL13 was added to the errata documentation.
3. Errata RTC12 was added to the errata documentation.
4. Errata SRAM1 was added to the errata documentation.
5. Errata ADC51 was added to the errata documentation.
6. RST1 Description was updated.
7. Errata ADC57 was added to the errata documentation.
8. Errata PMAP2 was added to the errata documentation.
9. Errata ADC55 was added to the errata documentation.
10. Errata ADC56 was added to the errata documentation.
11. Errata ADC53 was added to the errata documentation.
12. Errata CS5 was added to the errata documentation.
13. Errata ADC52 was added to the errata documentation.

Copyright © 2014–2019, Texas Instruments Incorporated

14. Errata PORT27 was added to the errata documentation.
15. Errata SYSTICK1 was added to the errata documentation.
16. Errata PCM2 was added to the errata documentation.
17. Package Markings section was updated.
18. Errata ADC54 was added to the errata documentation.
19. Errata CS6 was added to the errata documentation.
20. Errata FLASH40 was added to the errata documentation.
21. RST1 Workaround was updated.
22. Errata ADC59 was added to the errata documentation.
23. RST1 Function was updated.

Changes from document Revision C to Revision D.

1. Errata CS9 was added to the errata documentation.
2. Errata COMP9 was added to the errata documentation.
3. Errata REF9 was added to the errata documentation.
4. Errata USCI42 was added to the errata documentation.
5. Errata USCI41 was added to the errata documentation.
6. Silicon Revision C was added to the errata documentation.
7. Errata AES1 was added to the errata documentation.
8. Errata FLASH41 was added to the errata documentation.
9. Errata PSS3 was added to the errata documentation.
10. Errata COMP10 was added to the errata documentation.
11. Errata CS10 was added to the errata documentation.

Changes from document Revision D to Revision E.

1. Errata PORT27 was removed from the errata documentation.
2. Errata ADC58 was removed from the errata documentation.
3. Errata REF8 was removed from the errata documentation.
4. Errata CS10 was removed from the errata documentation.
5. Errata FLASH38 was removed from the errata documentation.
6. Errata ADC59 was removed from the errata documentation.
7. Errata REF9 was removed from the errata documentation.
8. Errata CS6 was removed from the errata documentation.
9. Errata ADC57 was removed from the errata documentation.
10. Errata RTC13 was removed from the errata documentation.
11. Errata SYS21 was removed from the errata documentation.
12. Errata CS5 was removed from the errata documentation.
13. Errata PCM1 was removed from the errata documentation.
14. Errata RTC9 was removed from the errata documentation.
15. Errata SRAM1 was removed from the errata documentation.
16. Errata RTC12 was removed from the errata documentation.
17. Errata PSS2 was removed from the errata documentation.
18. Errata ADC47 was removed from the errata documentation.
19. Errata ADC49 was removed from the errata documentation.
20. Errata COMP10 was removed from the errata documentation.
21. Errata REF7 was removed from the errata documentation.
22. Errata PMAP2 was removed from the errata documentation.

Copyright © 2014–2019, Texas Instruments Incorporated

23. Errata CS8 was removed from the errata documentation.
24. Errata COMP9 was removed from the errata documentation.
25. Errata FLASH40 was removed from the errata documentation.
26. Errata AES1 was removed from the errata documentation.
27. Errata ADC55 was removed from the errata documentation.
28. Errata FLASH39 was removed from the errata documentation.
29. Errata PSS3 was removed from the errata documentation.
30. Package Markings section was updated.
31. Errata ADC61 was removed from the errata documentation.
32. Errata ADC53 was removed from the errata documentation.
33. Errata DMA12 was removed from the errata documentation.
34. Errata REF3 was removed from the errata documentation.
35. Errata CS9 was removed from the errata documentation.
36. Errata ADC54 was removed from the errata documentation.
37. Errata RST2 was removed from the errata documentation.
38. Errata ADC52 was removed from the errata documentation.
39. Errata ADC60 was removed from the errata documentation.
40. Errata REF4 was removed from the errata documentation.
41. Errata ADC51 was removed from the errata documentation.
42. Errata BSL13 was removed from the errata documentation.
43. Errata USCI41 was removed from the errata documentation.
44. Errata ADC56 was removed from the errata documentation.
45. Errata PORT25 was removed from the errata documentation.
46. Errata WDG7 was removed from the errata documentation.
47. Errata SYSTICK1 was removed from the errata documentation.
48. Errata COMP8 was removed from the errata documentation.
49. Errata PSS1 was removed from the errata documentation.
50. Errata PCM2 was removed from the errata documentation.
51. Errata RST1 was removed from the errata documentation.
52. Errata FLASH41 was removed from the errata documentation.
53. Errata SRAM2 was removed from the errata documentation.
54. Errata ADC62 was removed from the errata documentation.
55. Errata RTC11 was removed from the errata documentation.
56. Errata ADC48 was removed from the errata documentation.

Changes from document Revision E to Revision F.

1. Errata BSL16 was added to the errata documentation.

Changes from document Revision F to Revision G.

1. Errata SYS26 was added to the errata documentation.

Changes from document Revision G to Revision H.

1. TA23 was added to the errata documentation.
2. USCI44 was added to the errata documentation.
3. USCI46 was added to the errata documentation.
4. COMP11 was added to the errata documentation.
5. RTC14 was added to the errata documentation.
6. USCI43 was added to the errata documentation.

Copyright © 2014–2019, Texas Instruments Incorporated

7. Silicon Revision D was added to the errata documentation.

Changes from document Revision H to Revision I.

1. USCI47 was added to the errata documentation.

Changes from document Revision I to Revision J.

1. Function for SYS26 was updated.

Changes from document Revision J to Revision K.

1. USCI48 was added to the errata documentation.
2. Function for USCI47 was updated.
3. Description for USCI47 was updated.
4. Workaround for USCI47 was updated.

Changes from document Revision K to Revision L.

1. USCI48 is no longer impacting silicon Revision D
2. Workaround for USCI47 was updated.

Changes from document Revision L to Revision M.

1. USCI45 was added to the errata documentation.

Changes from document Revision M to Revision N.

1. Description for USCI44 was updated.
2. Workaround for USCI44 was updated.

Changes from document Revision N to Revision O.

1. PORT34 was added to the errata documentation.
2. USCI50 was added to the errata documentation.
3. Function for USCI45 was updated.
4. Workaround for PORT31 was updated.

Changes from document Revision O to Revision P.

1. Erratasheet format update.
2. Added errata category field to "Detailed bug description" section

Changes from document Revision P to Revision Q.

1. UDMA2 was added to the errata documentation.

Changes from document Revision Q to Revision R.

1. USCI51 was added to the errata documentation.

Copyright © 2014–2019, Texas Instruments Incorporated

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2019, Texas Instruments Incorporated