NXP Semiconductors
Application Note

Document Number: AN5420
Rev. 0,02/2017

KE15Z TSI Development for Low Power Applications

Contents

1. Introduction 1
2. Features usage 1
2.1 TSI working for low power 1
2.1.1 Enable TSI working under low power 2
2.1.2 TSI Wake up MCU 2
2.1.3 Perform TSI scanning under low power 2
2.2 DMA transfer features 3
3. Implementation details 3
3.1 Data flow of DMA and TSI 3
3.2 Timing of DMA and TSI 4
3.3 DMA settings 5
3.3.1 DMA channel 0 settings 5
3.3.2 DMA channel 1 settings 6
4. Reference 7
5. Revision history 7

1. Introduction

The KE15Z series of microcontroller offers the touch sensing interface (TSI) which recognizes finger touch by sensing capacitance changes. The TSI IP supports up to 25 touch-sensing channels and wakes the microcontroller unit (MCU) from the low-power mode by out-of-range or end-of-scan interrupts. Traditionally, the low-power applications use a timer to wake up the MCU periodically, implement the TSI channel scan, and process the scan result to determine whether there is a touch event. If there is no touch, the MCU goes to sleep again. The limitation of this method is an extra power consumption caused by waking the MCU. This application notes explains how to develop the TSI software for low-power applications using DMA without waking the MCU and achieve low-power consumption. The test results show that the DMA method described in this document consumes under 1 mA current.

2. Features usage

2.1 TSI working for low power

The TSI IP provides touch-sensing detection on capacitive touch sensors. The external capacitive touch sensor is formed on PCB and the sensor electrodes are connected to TSI input channels through the I/O pins in the device.

2.1.1 Enable TSI working under low power

The KE15Z TSI is fully functional under low-power mode by setting TSI_GENCS[STPE] and can wake the MCU from the low-power mode by two kinds of interrupt: end-of-scan and out-of-range.

NOTE

> The TSI scan result is different between normal-run and low-power modes, so you need to configure different out-of-range threshold for normal-run and low-power modes.

2.1.2 TSI Wake up MCU

The end-of-scan interrupt is configured by the TSI_GENCS[ESOR] bit. When TSI scan completes, the TSI_GENCS[EOSF] bit is set, indicating the end of the TSI scan and the scan result is available in TSI_DATA[TSICNT]. If the TSI_GENCS[TSIIEN] and TSI_GENCS[ESOR] are set, and TSI_DATA[DMAEN] is clear, then an interrupt is submitted to the MCU immediately for post-processing, and it is optional for the interrupt to wake the MCU from the low-power mode.

The out-of-range interrupt is also configured by the TSI_GENCS[ESOR] bit. When enabled, the TSI IP compares TSI_DATA[TSICNT] with TSI_TSHD[HIGH] and TSI_TSHD[LOW] to check whether TSI_TSHD[HIGH] > TSI_DATA[TSICNT] > TSI_TSHD[LOW]. If the TSI scan result changes caused by the finger touch, an out-of-range interrupt is submitted to the MCU immediately for further process. The out-of-range interrupt is very useful to wake the MCU from the low-power mode when core falls asleep. The limitation of this method is that it can only monitor the given TSI channel. The TSI channel cannot be changed if the MCU is unable to execute the code under the low-power mode.

The DMA is required to change the TSI channel and the corresponding threshold in order to enable the TSI out-of-range feature monitor on all TSI channels.

2.1.3 Perform TSI scanning under low power

The TSI module allows a software or hardware trigger to start a scan by configuring the TSI_GENCS[STM] bit. In software trigger mode when the TSI_GENCS[STM] bit is configured as 0, a writing ‘1’ to TSI_DATA[SWTS] bit will start a TSI scan immediately, and the TSI channel ID to be scanned is specified by TSI_DATA[TSICH]. In hardware trigger mode when the TSI_GENCS[STM] bit is configured as 1, TSI will not start the scanning until the hardware trigger arrives. The hardware trigger source can be LPTMR/LPIT via TRGMUX. The software trigger mode is used in this application note, that DMA changes the TSI_DATA[TSICH] and writes TSI_DATA[SWTS] to perform a TSI scanning without core involved under low power mode.

2.2 DMA transfer features

KE15Z supports 8-channel eDMA (enhanced Direct Memory Access), performing complex data transfers with minimal intervention from a host processor.

This document describes the channel linking feature for DMA transfers.

Channel linking is a special eDMA feature for chaining more than one DMA channel. Channel linking is a mechanism where channel A activates channel Bor itself) DMA transfer by setting the TCDn_CSR[START] bit of channel B.

Linking channels allows you to start more than one transfer with a single request by defining a sequence of channels to be converted. The request starts the transfer on one DMA channel, and when that channel finishes, the transfer on the next channel starts. Channel linking is defined separately for major and minor loop finishes.

The minor loop-channel linking occurs at the completion of the minor loop or one iteration of the major loop. The TCDn_CITER[E_LINK] field determines whether a minor loop link is requested. When enabled, the channel link is made after each iteration of the major loop except for the last.

The major loop-channel linking occurs when all iterations of the major loop are finished. The major loop-channel link field (DMA_TCDn_CSR[MAJORELINK]) is used to determine if a channel link should be made.

NOTE

There is no minor loop-channel linking at the last minor loop. Instead, the major loop channel linking will trigger the next given DMA channel to start transfer.

3. Implementation details

3.1 Data flow of DMA and TSI

The TSI low-power example requires two DMA channels. DMA channel 0 transfers the corresponding threshold value from SRAM to TSI_TSHD register. The DMA channel 1 transfers the TSI channel ID and the start bit from the SRAM to TSI_DATA register to initiate the TSI scan of the given TSI channel. By DMA channel linking, DMA channel 0 minor loop done will start DMA channel 1 transfer, so firstly fill the threshold of the out-of-range interrupt, then start a TSI scan.

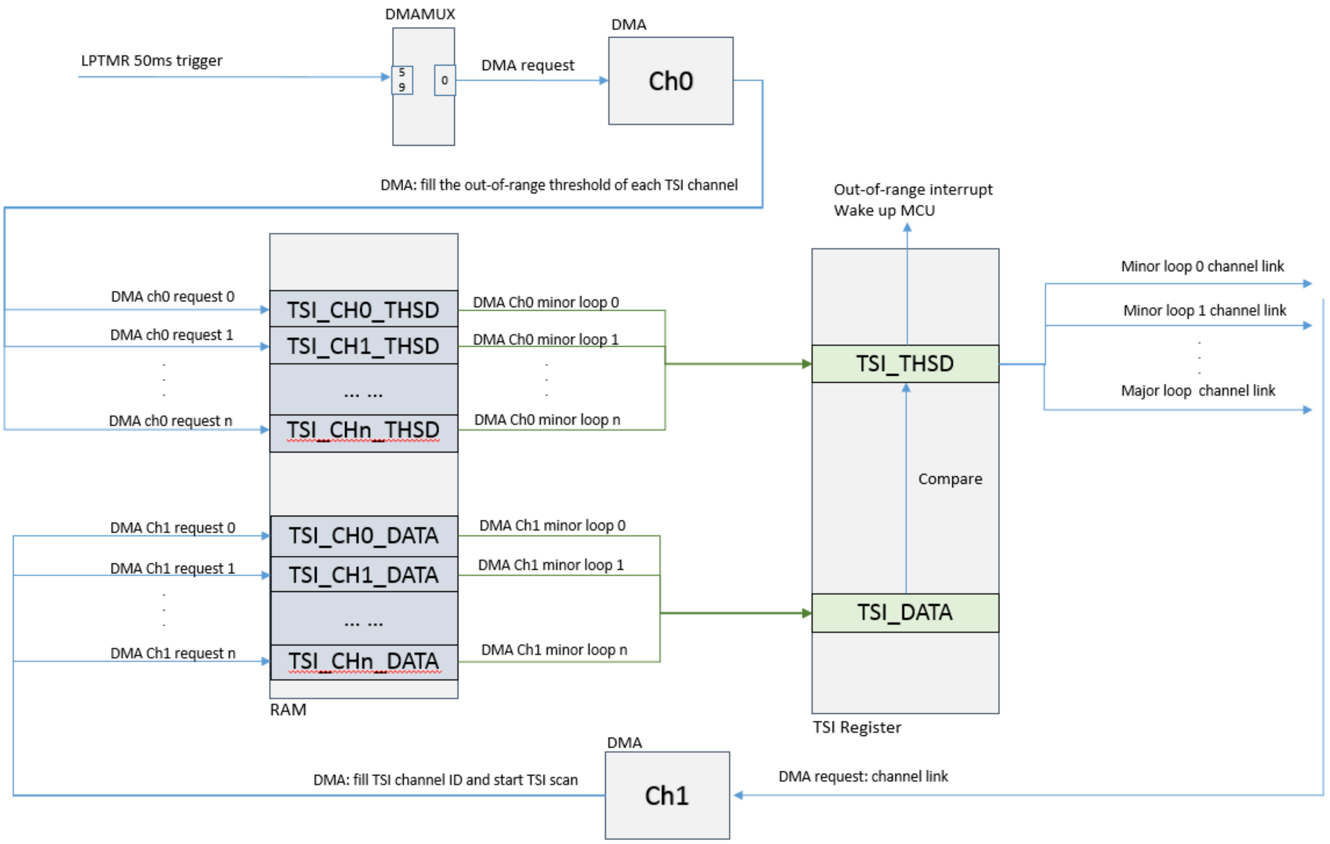


Figure 1. **Data flow between SRAM and TSI register**

3.2 Timing of DMA and TSI

The following steps occur in the TSI low-power example using DMA.

1. LPTMR generates periodic DMA request every 50 ms, to start DMA channel 0 transfer.
2. DMA channel 0 moves the out-of-range threshold value of TSI channel 0 from SRAM to TSI_TSHD register.
3. When DMA channel 0 minor loop is done, that is the out-of-range threshold of TSI channel 0 has been configured, DMA channel 0 triggers DMA channel 1 transfer.
4. DMA channel moves the corresponding TSI channel ID (channel 0) and software start bit from SRAM to TSI_DATA register(TSI_DATA[TSICH] and TSI_DATA[SWTS], once the TSI_DATA[SWTS] is written, TSI starts scan for the given channel specified by the TSI_DATA[TSICH].
5. When the TSI scan is done, the TSI IP compares the out-of-range threshold (located in TSI_TSHD) and TSI scan result (located in TSI_DATA[TSICNT]). The out-of-range interrupt will be generated immediately to wake up MCU from low power mode if the TSICNT > TSI_TSHD[THRESH] or TSICNT < TSI_TSHD[THRESL].
6. After 50 ms, the LPTMR generates another DMA request, to start next TSI channel scan and out-of-range comparison using DMA.

7. After all, TSI channels have been scanned, the DMA channel 0/1 major loop is done. Then DMA channel 0 changes the source address to the start point of TSI channel 0's out-of-range threshold, and DMA channel 1 changes the source address to the start point of TSI channel 0's channel ID and start bit.
8. Repeat from step 1 again, until there is an out-of-range interrupt caused by finger touch to wake the MCU from the low-power mode.

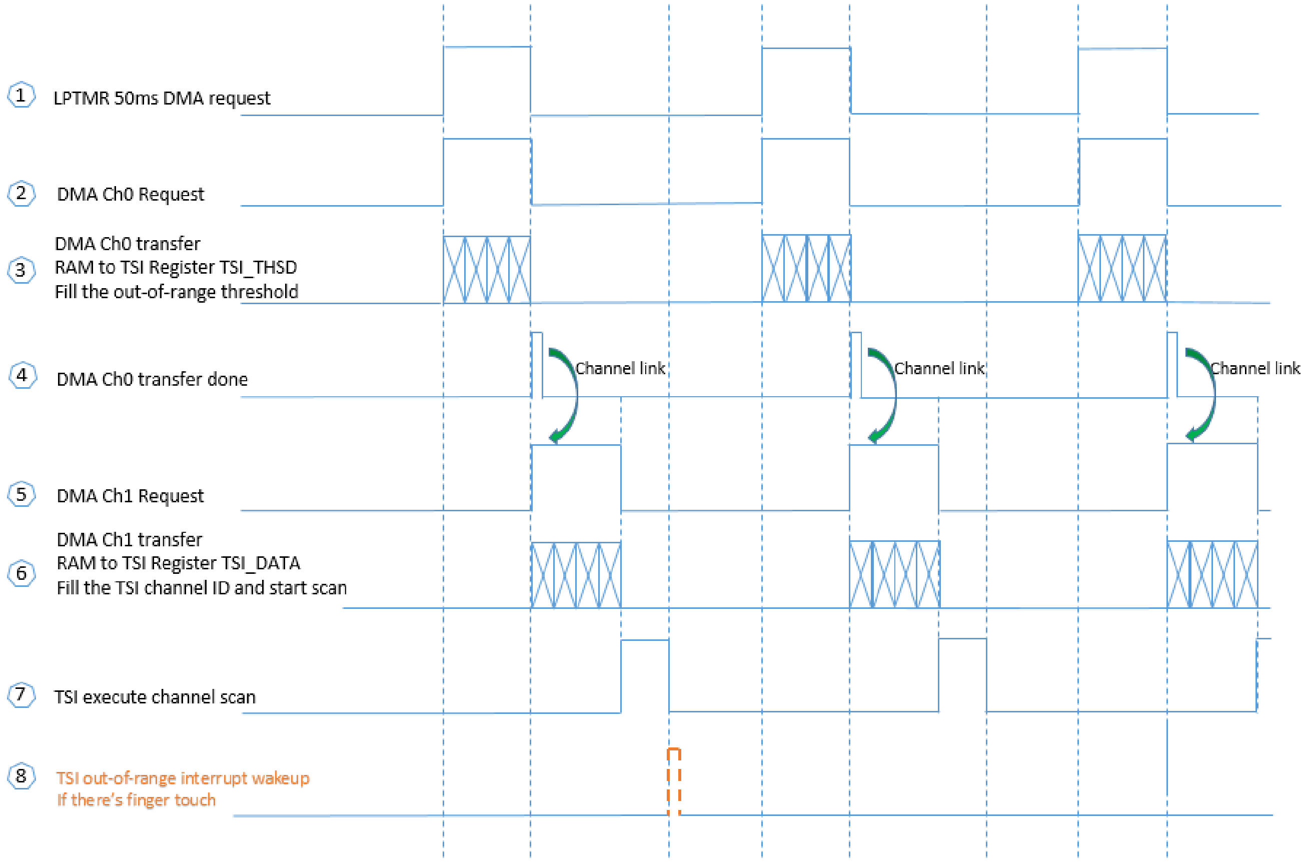
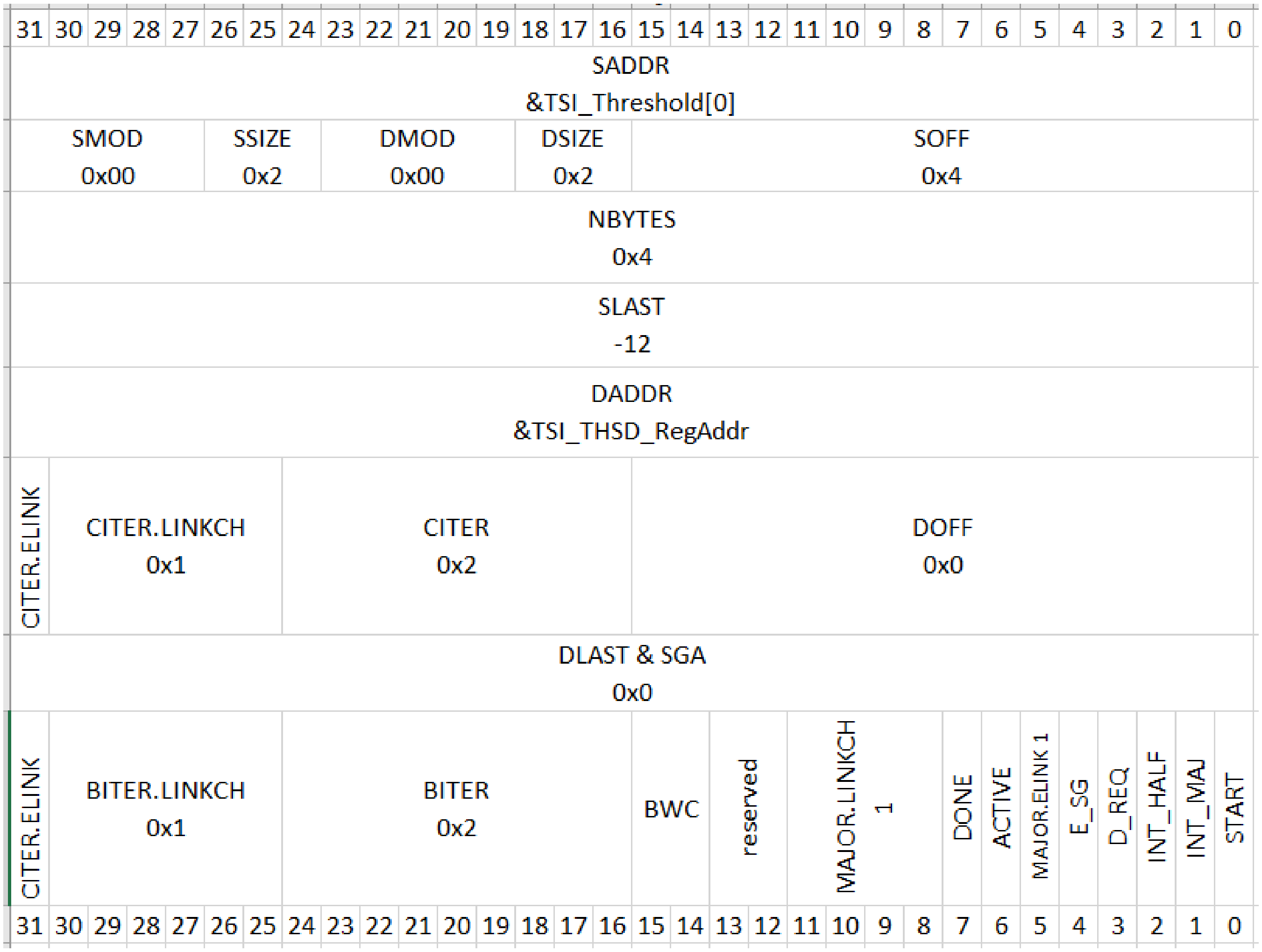

1 LPTMR 50ms DMA request
2 DMA Ch0 Request
3 DMA Ch0 transfer
RAM to TSI Register TSI_THSD
Fill the out-of-range threshold
4 DMA Ch0 transfer done
Channel link
Channel link
Channel link
5 DMA Ch1 Request
6 DMA Ch1 transfer
RAM to TSI Register TSI_DATA
Fill the TSI channel ID and start scan
7 TSI execute channel scan
8 TSI out-of-range interrupt wakeup
If there's finger touch

Figure 2. **Timing of TSI scan and DMA transfer**

3.3 DMA settings

3.3.1 DMA channel 0 settings

DMA channel 0 is used in this document to transfer the corresponding TSI out-of-range threshold value from SRAM (variable of TSI_Threshold[3] array) to TSI_TSHD register, and signal DMA channel 1 to start transfer via minor loop-channel linking. The settings of the DMA channel 0 TCD are shown below.

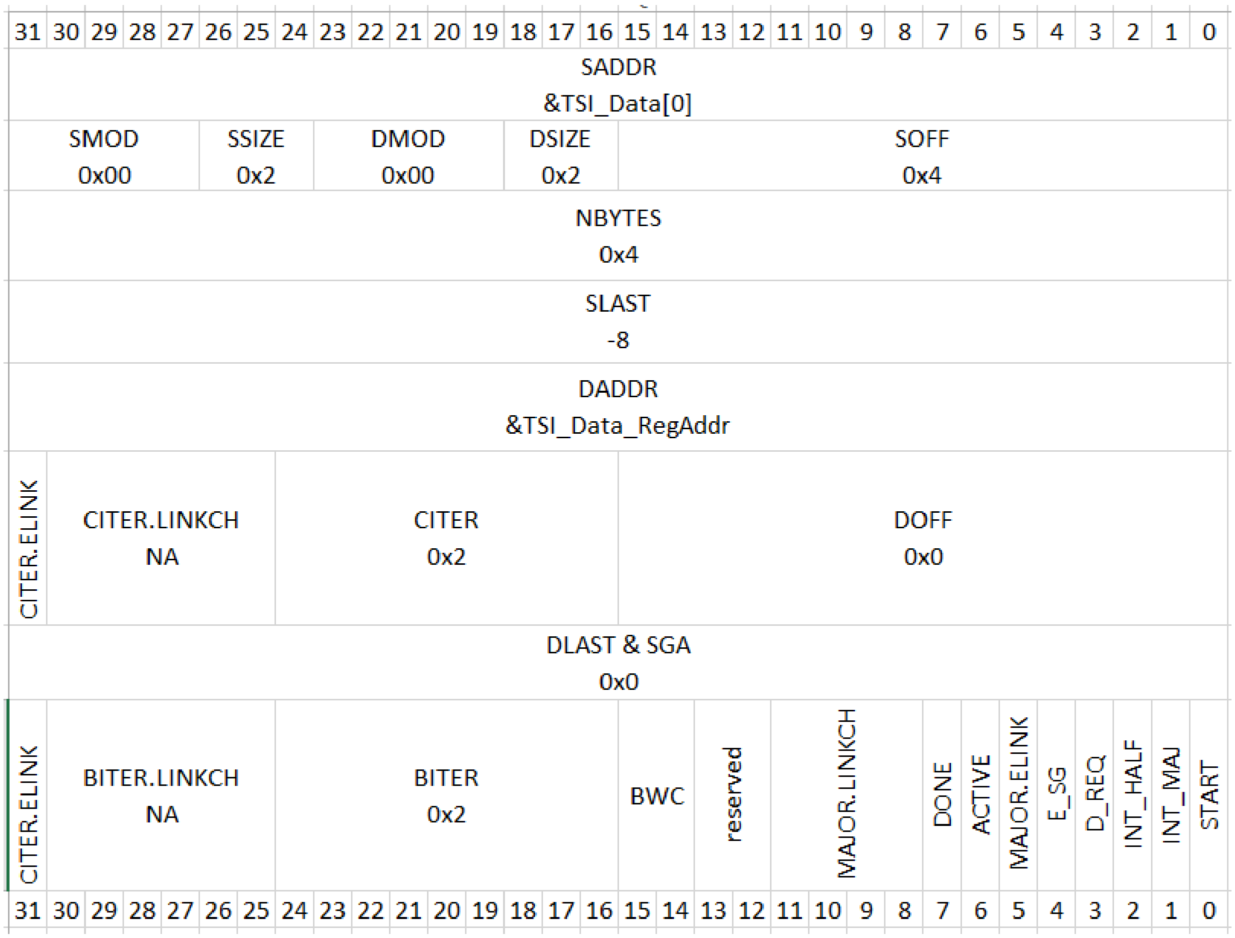


31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SADDR &TSI_Threshold[0]																															
SMOD 0x00					SSIZE 0x2			DMOD 0x00					DSIZE 0x2			SOFF 0x4															
NBYTES 0x4																															
SLAST -12																															
DADDR &TSI_THSD_RegAddr																															
CITER.ELINK	CITER.LINKCH 0x1						CITER 0x2									DOFF 0x0															
DLAST & SGA 0x0																															
CITER.ELINK	BITER.LINKCH 0x1						BITER 0x2									BWC		reserved		MAJOR.LINKCH 1				DONE	ACTIVE	MAJOR.ELINK 1	E_SG	D_REQ	INT_HALF	INT_MAJ	START
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Figure 3. **DMA Channel 0 Settings**

3.3.2 DMA channel 1 settings

The DMA channel 1 transfers the TSI channel ID and start bit from SRAM (variable of TSI_Data[2] array) to TSI_DATA register to start a TSI scan of the given TSI channel. The settings of the DMA channel 1 TCD are shown below.



<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="32">SADDR
&TSI_Data[0]</td></tr>
<tr><td colspan="5">SMOD
0x00</td><td colspan="3">SSIZE
0x2</td><td colspan="5">DMOD
0x00</td><td colspan="3">DSIZE
0x2</td><td colspan="16">SOFF
0x4</td></tr>
<tr><td colspan="32">NBYTES
0x4</td></tr>
<tr><td colspan="32">SLAST
-8</td></tr>
<tr><td colspan="32">DADDR
&TSI_Data_RegAddr</td></tr>
<tr><td>CITER.ELINK</td><td colspan="6">CITER.LINKCH
NA</td><td colspan="9">CITER
0x2</td><td colspan="16">DOFF
0x0</td></tr>
<tr><td colspan="32">DLAST & SGA
0x0</td></tr>
<tr><td>CITER.ELINK</td><td colspan="6">BITER.LINKCH
NA</td><td colspan="9">BITER
0x2</td><td colspan="2">BWC</td><td colspan="2">reserved</td><td colspan="4">MAJOR.LINKCH</td><td>DONE</td><td>ACTIVE</td><td>MAJOR.ELINK</td><td>E_SG</td><td>D_REQ</td><td>INT_HALF</td><td>INT_MAJ</td><td>START</td></tr>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

Figure 4. **DMA Channel 1 Settings**

4. Reference

The following references are available on NXP website.

1. KE15 TSI User Guide (Document: KE15ZTSIUG)
2. Using DMA to Emulate ADC Flexible Scan Mode on Kinetis K Series (AN4590)
3. KE15Z Reference Manual (Document: KE1xZP100M72SF0RM)

5. Revision history

Table 1. Revision history

Revision number	Date	Substantive changes
0	02/2017	Initial release

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer’s technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.