Fujitsu Microelectronics Europe
Application Note

MCU-AN-301000-E-V11

TOUCH SENSOR CONTROLLER

FMA1127DC SERIES

INITIALIZATION AND REGISTER DESCRIPTION

APPLICATION NOTE

Revision History

Date	Issue
2009-04-16	CHa First version (based on FMA Tuning Manual)
2009-06-04	CHa Register Description updated including FMA1127DC version Some typos corrected I2C schematic added

This document contains 30 pages.

Warranty and Disclaimer

The use of **the deliverables** (e.g. software, application examples, target boards, evaluation boards, starter kits, schematics, engineering samples of IC's etc.) is subject to the conditions of Fujitsu Microelectronics Europe GmbH ("FME") as set out in (i) the terms of the License Agreement and/or the Sale and Purchase Agreement under which agreements the Product has been delivered, (ii) the technical descriptions and (iii) all accompanying written materials.

Please note that the deliverables are intended for and must only be used in an evaluation laboratory environment.

The software deliverables are provided without charge and therefore provided on an as-is basis. The software deliverables are to be used exclusively in connection with FME products.

Regarding hardware deliverables, FME warrants that they will be free from defects in material and workmanship under use and service as specified in the accompanying written materials for a duration of 1 year from the date of receipt by the customer.

Should a hardware deliverable turn out to be defect, FME's entire liability and the customer's exclusive remedy shall be, at FME´s sole discretion, either return of the purchase price and the license fee, or replacement of the hardware deliverable or parts thereof, if the deliverable is returned to FME in original packing and without further defects resulting from the customer's use or the transport. However, this warranty is excluded if the defect has resulted from an accident not attributable to FME, or abuse or misapplication attributable to the customer or any other third party not relating to FME or to unauthorised decompiling and/or reverse engineering and/or disassembling.

FME does not warrant that the deliverables does not infringe any third party intellectual property right (IPR). In the event that the deliverables infringe a third party IPR it is the sole responsibility of the customer to obtain necessary licenses to continue the usage of the deliverable.

In the event the software deliverables include the use of open source components, the provisions of the governing open source license agreement shall apply with respect to such software deliverables.

To the maximum extent permitted by applicable law FME disclaims all other warranties, whether express or implied, in particular, but not limited to, warranties of merchantability and fitness for a particular purpose for which the deliverables are not designated.

To the maximum extent permitted by applicable law, FME's liability is restricted to intention and gross negligence. FME is not liable for consequential damages.

Should one of the above stipulations be or become invalid and/or unenforceable, the remaining stipulations shall stay in full effect.

The contents of this document are subject to change without a prior notice, thus contact FME about the latest one.

Contents

REVISION HISTORY **2**

WARRANTY AND DISCLAIMER **3**

CONTENTS **4**

1 INTRODUCTION **6**

2 INITIALIZATION OF THE TSC **7**

2.1 I2C Host Interface and TSC connection to MCU 7

2.2 System Setup and Initialization 8

3 REGISTER MAP SUMMARY **10**

4 REGISTER DESCRIPTIONS **11**

4.1 Feature Select (0x00) 11

4.2 Alpha0-11 (0x01 – 0x0C) 12

4.3 Beta (0x0D) 12

4.4 AIC Wait (Wait before Calibration time, 0x0E) 13

4.5 Reference Delay (0x0F) 13

4.6 Hysteresis Delay (0x10 – 0x1B) 14

4.7 Strength Threshold 0 – 11 (0x1C – 0x27) 14

4.8 Sampling Interval (0x28) 14

4.9 Integration Time (0x29) 15

4.10 Idle Time (0x2A) 15

4.11 Reserved (0x2B) 15

4.12 Reserved (0x2C) 15

4.13 GPIO REG L (0x2D) 16

4.14 GPIO REG H (0x2E) 16

4.15 GPIO Configuration L (0x2F) 16

4.16 GPIO Configuration H (0x30) 17

4.17 GPIO Direction L (0x31) 17

4.18 GPIO Direction H (0x32) 17

4.19 Control (0x33) 18

4.20 Interrupt Mask (0x34) 19

4.21 Interrupt Clear (0x35) 19

4.22 Interrupt Edge (0x36) 20

4.23 Control 2 (0x37) 21

4.24 Beep Period (0x38) 21

4.25 Beep Frequency (0x39) 21
4.26 Calibration Interval (0x3A) 22
4.27 External Interrupt Enable (0x3B) 22
4.28 External Interrupt Polarity (0x3C) 22
4.29 Filter Period (0x3D) 22
4.30 Filter Threshold (0x3E) 23
4.31 CONTROL3 (0x3F) 23
4.32 Bounce Cancelling Period (0x40) 24
4.33 Strength 0-11 (0x50 – 0x5B) 24
4.34 Calibrated Impedance 0-11 (0x5C – 0x67) 24
4.35 Impedance 0-11 (0x68 – 0x73) 25
4.36 Status (0x74) 25
4.37 Touch Byte L (0x75) 25
4.38 Touch Byte H (0x76) 26
4.39 Interrupt Pending (0x79) 26
4.40 GPIO IN L (0x7A) 27
4.41 GPIO IN H (0x7B) 27
4.42 GPIO IN L FIFO (0x7C) 27
4.43 GPIO IN LBCU (0x7D) 27
4.44 BIAS OFF (0xFA) 28
4.45 BIAS ON (0xFB) 28
4.46 Wakeup SLEEP (0xFC) 28
4.47 Enter SLEEP (0xFD) 28
4.48 Cold Reset (0xFE) 29
4.49 Warm Reset (0xFF) 29

5 APPENDIX 30
5.1 Glossary 30
5.2 Related Documents 30

1 Introduction

The FMA1127 Touch Sensor Controller (TSC) offers a very effective and highly flexible way of implementing capacitive touch sensing for various applications. The TSC can be connected to virtually any host MCU by a two-wire I2C host interface, minimizing the number of connections needed.

The FMA1127 senses the touch by comparing the capacitance of the touch pad against an internal reference using the patented, purely digital technology of ATlab, Korea. This allows fast response to touch events, automatic calibration to environmental changes such as temperature or humidity using the hardware-implemented AIC™ (Automatic Impedance Calibration) as well as numerous parameters such as individual channel sensitivity to tune the sensing behaviour to fit the application's needs. Because of the unique sensing method, the sensor is highly immune to disturbances caused e.g. by water drops.

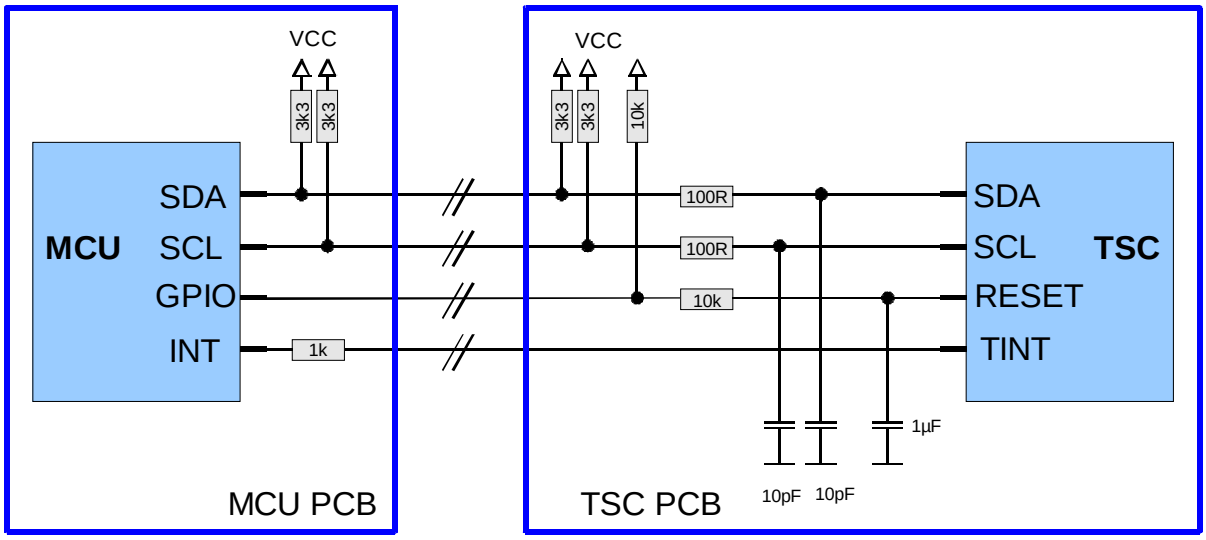
The FMA1127 touch sensor controller is developed and owned by ATLab Inc., South Korea, and is distributed by Fujitsu Microelectronics Europe.

2 Initialization of the TSC

2.1 I2C Host Interface and TSC connection to MCU

The FMA1127 uses a two-wire I2C interface for host connection. It operates at frequencies up to 400 kHz. By setting different chip IDs using the ID pins, up to four FMA1127 devices can be connected to one MCU I2C channel.

The recommended connection scheme is shown below (power and GND connections not shown):



The I2C bus requires pull-up resistors between the bus lines and VCC. The value of the pull-up resistors is dependent of factors such as bus length, supply voltage, number of slaves, etc. usually values of some few kOhms are recommended. It can be beneficial to use one pull-up resistor of twice the desired value on every end of the bus instead of one resistor only (e.g. 2x 3k3). Additional series resistors and a small (some pF) filter capacitor close to the TSC IC are recommended to increase noise rejection and over-voltage protection of the I2C pins. This especially applies when the bus length exceeds ~15cm.

For more details about I2C timing, protocol and characteristics, please refer to the FMA1127 product data sheet and the I2C bus specification.

The TINT (Touch Interrupt) and GINT (General Interrupt) output pins of the TSC optionally can be connected to external Interrupt pins of the MCU in order to avoid software polling of the touch status. Interrupt polarity can be configured.

The TSC Reset input pin should be made controllable by the host MCU circuit, either by connecting it to a GPIO pin of the MCU so that the host application can reset the TSC at startup, or by connecting it to the MCU reset circuit or power watchdog IC (if present).

2.2 System Setup and Initialization

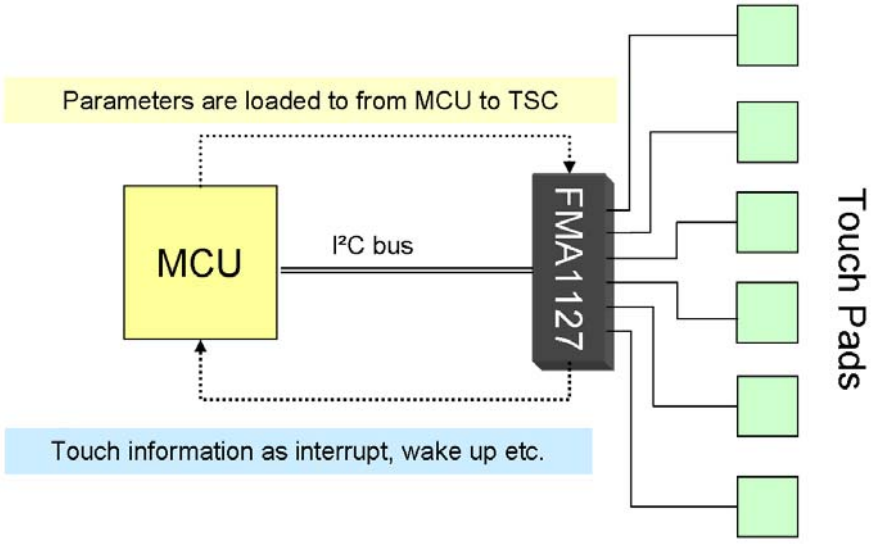


Figure 1: TSC System Setup

After system startup, the TSC configuration registers are pre-loaded with the factory default settings as shown in the register description. Therefore, the host MCU has to configure the TSC registers according to the application needs (sensitivity, APIS modes, etc.) after reset. This is done by the I2C communication bus, either by single write accesses to the relevant registers, or by a bulk transfer of all configuration registers. For details about communication timing, used I2C device address, please refer to the product data sheet.

For the following examples, the default device ID (0x58) of the FMA1127 will be used.

As mentioned, single or bulk read and write operations are supported. For single operations, the host MCU sends the register address for each register, followed by the data payload (write) or a repeated start condition with read command. So there is a full I2C sequence (START, ID, DATA, STOP) for every register access. The FMA1127 also supports bulk transfer of multiple registers with subsequent addresses. Hereby, only the first register address has to be transferred, and after every byte of transferred data, the internal address for the next transfer is incremented automatically. This is especially useful for initialization during startup or to read e.g. all strength registers. Burst mode allows higher data payload rates, because there is no additional addressing between the data byte transfers.

Example code snippets are shown below:

Single write operation:

```
void I2C_Set_Register(unsigned char i2c_address, unsigned char register_address, unsigned char
value)
  {
    I2C_Start(i2c_address, WRITE);           // I2C Device address, WRITE (for address + data)
    I2C_Write(register_address);             // Register Address
    I2C_Write(value);
    I2C_Stop();
  }
```

Single read operation:

```
unsigned char I2C_Read_Register(unsigned char i2c_address, unsigned char register_address)
  {
    unsigned char result;

    I2C_Start(i2c_address, WRITE);          // I2C Device address, WRITE for address
    I2C_Write(register_address);            // Register Address
    I2C_Continue(i2c_address, READ);        // swith to READ
    result = I2C_LastRead();                // read data from I2C
    I2C_Stop();
    return result;
  }
```

Bulk read operation:

```
void I2C_BurstRead(unsigned char i2c_address, unsigned char start_reg_address, unsigned char
count, unsigned char * buffer)
{
  I2C_Start(i2c_address, WRITE);
  I2C_Write(start_reg_address);             // send register address to be read
  I2C_Continue(i2c_address, READ);          // Restart, with READ comand
  while (--count > 0) {
    *(buffer++) = I2C_Read();               // read register and write to buffer
  }
  *buffer = I2C_LastRead();                 // read register (last read)
  I2C_Stop();
}
```

An example for device initialization (here using single transfers) could be as follows:

```
const unsigned char tsc_init_data[] = {   // This only contains the writable TSC registers.

       5,                     // 0x01: ALPHA0
       5,                     // 0x02: ALPHA1
       5,                     // 0x03: ALPHA2
  ...
  ...remaining configuration registers…
  ...
       5,                     // 0x3D: FILTER_PERIOD
       4                      // 0x3E: FILTER_THRESHOLD

}

void wait(unsigned long int i) {
       while (i--) __wait_nop();             // wait loop
}


void TSC_Init(void) {

       unsigned char i;

       I2C_Set_Register(TSC_ID, TSC_FEATURE, TSC_APIS_MODE_1);     // set APIS Mode 1

       for (i=0; i<62; i++) {                                      // set TSC registers
              I2C_Set_Register(TSC_ID, TSC_ALPHA_00 + i, tsc_init_data[i]);
       }

       wait(1000);

       I2C_Set_Register_WR(TSC_ID, TSC_WARM_RESET, 0x00);          // isue warm reset,
                                                                   // do not wait for ACK

       wait(50000);
}
```

3 Register Map Summary

Generally, there are three different register types in the FMA1127:

0x00 – 0x3E: Read/Writeable (Sensitivity/alpha, reference delay, …)

0x50 – 0x7B: Read-only (Touch strength, …)

0xFA – 0xFF: Write-only (e.g. warm reset)

Please note, that after writing to a write-only register (e.g. warm reset), no ACK is generated by the FMA1127DA, so that the host SW should not wait for the ACK signal after issuing a warm reset by writing to address 0xFF.

Addr. (HEX)	Register Name	Addr. (HEX)	Register Name	Addr. (HEX)	Register Name
00	Feature	26	Strength Threshold 10	5C	Calibrated Impedance 0
01	ALPHA 0	27	Strength Threshold 11	5D	Calibrated Impedance 1
02	ALPHA 1	28	Sampling Interval	5E	Calibrated Impedance 2
03	ALPHA 2	29	Integration Time	5F	Calibrated Impedance 3
04	ALPHA 3	2A	IDLE Time	60	Calibrated Impedance 4
05	ALPHA 4	2C	Reserved	61	Calibrated Impedance 5
06	ALPHA 5	2D	GPIO REG L	62	Calibrated Impedance 6
07	ALPHA 6	2E	GPIO REG H	63	Calibrated Impedance 7
08	ALPHA 7	2F	GPIO Configuration L	64	Calibrated Impedance 8
09	ALPHA 8	30	GPIO Configuration H	65	Calibrated Impedance 9
0A	ALPHA 9	31	GPIO Direction L	66	Calibrated Impedance 10
0B	ALPHA 10	32	GPIO Direction H	67	Calibrated Impedance 11
0C	ALPHA 11	33	Control	68	Impedance 0
0D	BETA	34	Interrupt Mask	69	Impedance 1
0E	AIC Wait	35	Interrupt Clear	6A	Impedance 2
0F	Reference Delay	36	Interrupt Edge	6B	Impedance 3
10	Reserved	37	Control 2	6C	Impedance 4
11	Reserved	38	Beep Period	6D	Impedance 5
12	Reserved	39	Beep Frequency	6E	Impedance 6
13	Reserved	3A	Calibration Interval	6F	Impedance 7
14	Reserved	3B	EINT Enable	70	Impedance 8
15	Reserved	3C	EINT Polarity	71	Impedance 9
16	Reserved	3D	FILTER Period	72	Impedance 10
17	Reserved	3E	FILTER Threshold	73	Impedance 11
18	Reserved	3F	CONTROL 3	74	Status
19	Reserved	40	Bounce Cancelling Period	75	Touch Byte L
1A	Reserved	50	Strength 0	76	Touch Byte H
1B	Reserved	51	Strength 1	79	Interrupt Pending
1C	Strength Threshold 0	52	Strength 2	7A	GPIO IN L
1D	Strength Threshold 1	53	Strength 3	7B	GPIO IN H
1E	Strength Threshold 2	54	Strength 4	7C	GPIO IN L FIFO
1F	Strength Threshold 3	55	Strength 5	7D	GPIO IN L BCU
20	Strength Threshold 4	56	Strength 6	FC	Wakeup SLEEP
21	Strength Threshold 5	57	Strength 7	FD	Enter SLEEP
22	Strength Threshold 6	58	Strength 8	FE	Cold Reset
23	Strength Threshold 7	59	Strength 9	FF	Warm Reset
24	Strength Threshold 8	5A	Strength 10		
25	Strength Threshold 9	5B	Strength 11		

Figure 2: Register Map Summary

4 Register Descriptions

The FMA1127 has three types of registers, Read-write registers, Read-only registers and Write-only registers. Read-Write registers start at 0x00, Read-only registers start at 0x50 and Write-only registers start at 0xFA. The notation has 4 lines: first, second, third and fourth lines show bit positions, bit names, register type and default values when the FMA1127 receives hard reset or cold reset, respectively. In the FMA1127DC, there are 4 additional registers: CONTROL3 Register (0x39), BOUNCE Cancelling Period (0x40), GPIO_IN_L_FIFO (0x7C) and GPIO_IN_L_BCU (0x7D).

4.1 Feature Select (0x00)

B7	B6	B5	B4	B3	B2	B1	B0
				APIS3	APIS2	APIS1	
				R/W	R/W	R/W	
0	0	0	0	0	0	0	0

- This register selects APIS mode output.
- Only B3, B2 and B1 are used. The rest are reserved.
- **Caution:** Just set only one bit among B3, B2 and B1 to 'H'.
- In order to activate APIS mode properly, Integration Time Register (0x29) and Strength Threshold Registers (0x1C~0x27) also should be initialized. Please refer to the description of these registers.

B1: If this bit is set to 'H', APIS mode I is activated. In APIS mode I, the strongest sensor output is available among twelve Strength Registers (0x50~0x5B) in the given integration period defined at Integration Time Register (0x29). This mode is suitable for the button application to screen out weakly touched buttons.

B2: If this bit is set to 'H', APIS mode II is enabled. If current strength values stored in Strength Registers (0x50 ~ 0x5B) are greater than pre-defined values of Strength Threshold Registers (0x1C~0x27), all corresponding sensor outputs are available.

B3: If this bit is set to 'H', two strongest sensor outputs are available among twelve Strength Registers (0x50~0x5B) in the given integration period defined at Integration Time Register (0x29). This is called APIS mode III. It is suitable for multi-touch applications such as a game console.

Notes:

1. The register addresses are 0x1C~0x27 to set or change sensor's Strength Threshold values.
2. Current strength values can be read at Strength Registers (0x50~0x5B).

4.2 Alpha0-11 (0x01 – 0x0C)

B7	B6	B5	B4	B3	B2	B1	B0
			ALPHA Register				
			R/W	R/W	R/W	R/W	R/W
0	0	0	0	1	0	0	0

- These registers set the sensitivity of each sensor input.
- Sensitivity of S0 through S11 can be set individually with different values.
- To assign bigger value will decrease the sensitivity and to assign smaller value will increase the sensitivity.
- One step of ALPHA is equal to 0.078pF. If an ALPHA is set to 5, 0.39pF or grater capacitance should be induced by the finger to activate touch output.
- Default value is 8.
- The minimum value of ALPHA is 4 and the maximum is 31. For proximity application, you can use smaller value than minimum ALPHA but be cautious to check if AIC works correctly.
- **B7, B6 and B5:** Reserved

4.3 Beta (0x0D)

B7	B6	B5	B4	B3	B2	B1	B0
				BETA Register			
				R/W	R/W	R/W	R/W
0	0	0	0	0	1	0	0

- This register sets AIC (Automatic Impedance Calibration) entering threshold value called BETA. It detects non-touch conditions of all twelve sensor inputs.
- The current impedances and previous impedances of all twelve sensor inputs are compared before starting AIC operation. If differences of them are all lower than BETA, AIC can start.
- Default value is 4.
- Min value of Beta is 4 and Maximum is 15.
- To increase this value will have more chance to perform AIC.
- **B7, B6, B5 and B4:** Reserved

4.4 AIC Wait (Wait before Calibration time, 0x0E)

B7	B6	B5	B4	B3	B2	B1	B0
AIC Wait Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	1	0	0	1	1	1

- This register sets AIC waiting time to stabilize AIC operation with BETA. It can eliminate false calibration when you remove the finger from the touch pads very slowly.
- AIC keeps blocked during AIC_WAIT time after all twelve sensor inputs become non-touch condition.
- If at least one input is touched during AIC_WAIT time, AIC_WAIT time is reloaded.
- If you assign bigger value of AIC_WAIT time, AIC needs longer time to start.
- Default value is 0x27 and it blocks AIC about 124.8msec under the equation below where Sensor Clock is 20KHz (0.05msec).

AIC Wait time = AIC_WAIT Register Value x 64 x Sensor Clock Period

4.5 Reference Delay (0x0F)

B7	B6	B5	B4	B3	B2	B1	B0
	Reference Delay Register						
	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register changes the value of Reference Delay Chain located in the chip which can adjust parasitic capacitance mismatch existing on all twelve sensor inputs.
- Reference Delay Chain consists of 100 unit cells whose time delay is equivalent to 0.078pF. So, the maximum value of Reference delay is equal to attaching 7.8pF of capacitor at the reference input pin denoted as Aref pin.
- If parasitic mismatch existing on sensor inputs is greater than 7.8pF, you still can adjust it by attaching external tuning capacitor to the Aref pin.
- Its maximum range is 0 through 99. Nevertheless, its recommended range is 20 through 80 to cope with mass production variation.
- Warm reset is mandatory when this value is changed.

4.6 Hysteresis Delay (0x10 – 0x1B)

- Reserved. Default value is 1.

4.7 Strength Threshold 0 – 11 (0x1C – 0x27)

B7	B6	B5	B4	B3	B2	B1	B0
Strength Threshold Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	1

- These registers set Strength Threshold values of S0~S11.
- These registers are used for APIS mode to select touch outputs greater than Strength Threshold values in the given integration time defined at Integration Time Register (0x29).
- **CAUTION:** The Strength Threshold value must be lower than or equal to the value in Integration Time Register (0x29). Otherwise, no touch output is generated.
- Default value is 1. However, this threshold value should be between approximately 1/10 and 2/10 of Integration Time. For example, if Integration Time is 100, then assign all these threshold values to between 10 and 20.
- Recommended minimum value of Strength Threshold is 5 regardless of the value of Integration Time. (Recommended minimum value of Integration Time is 10.)
- To assign higher threshold value will decrease sensitivity but increase filtering capability to screen out weekly touched sensor inputs.

4.8 Sampling Interval (0x28)

- Reserved. Default value is 0x03.

4.9 Integration Time (0x29)

B7	B6	B5	B4	B3	B2	B1	B0
Integration Time Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	1	1	1	1

- This register is to set Integration Time for APIS mode. Integration Time means the number of accumulation of touch output synchronized by Sensor Clock speed whose default value is 20KHz.
- Update Period of touch output in APIS mode is varied by the equation below:

Update Period of Touch Output = Sensor Clock Period x (Integration Time Register Value)

- The maximum value of Strength Registers (0x50~0x5B) is equal to Integration Time.
- The Integration Time must be greater than or equal to all values of Strength Threshold registers.
- The smaller this value is, the worse APIS filtering capability is. On the other hand, the bigger this value is set, the longer the response time of touch output is. For example, if sensor clock is set to be 20KHz, and integration time is set to be 100, the update period of touch output and Strength register is every 5msec.
- Default value is 15.
- The range of Integration Time is 1 through 255, but its recommended minimum value is 10 for proper APIS filtering capability.

4.10 Idle Time (0x2A)

B7	B6	B5	B4	B3	B2	B1	B0
IDLE Time Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	1	1	1	1

- This register determines entering time to IDLE mode after all inputs become non-touch status.
- If sensor clock is 20KHz, for example, IDLE time will be 3.75sec.

IDLE Time = Register value x 5000 x Sensor Clock Period

4.11 Reserved (0x2B)

- Reserved. Default value is 0.

4.12 Reserved (0x2C)

- Reserved. Default value is 0x01.

4.13 GPIO REG L (0x2D)

B7	B6	B5	B4	B3	B2	B1	B0
DIO 7	DIO 6	DIO 5	DIO 4	DIO 3	DIO 2	DIO 1	DIO 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register stores data to output through DIO0~7 ports only when corresponding bits in GPIO Configuration L register (0x2F) are configured as extended GPIOs by assigning 'H' and corresponding bits in GPIO Direction L register (0x31) are configured as Output by assigning 'L'.

4.14 GPIO REG H (0x2E)

B7	B6	B5	B4	B3	B2	B1	B0
				DIO 11	DIO 10	DIO 9	DIO 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register stores data to output through DIO8~11 ports only when corresponding bits in GPIO Configuration H register (0x30) are configured as extended GPIOs by assigning 'H' and corresponding bits in GPIO Direction H register (0x32) are configured as Output by assigning 'L'.
- **B7, B6, B5 and B4:** Reserved

4.15 GPIO Configuration L (0x2F)

B7	B6	B5	B4	B3	B2	B1	B0
DIO 7	DIO 6	DIO 5	DIO 4	DIO 3	DIO 2	DIO 1	DIO 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register configures the characteristics of DIO0~7 ports as either Touch output ports or extended GPIO ports.
- The bits with 'L' as default are configured as Touch Output ports and the bits with 'H' are configured as extended GPIO ports.
- With the configuration of Touch output ports, the status of sensor input pins is reflected to these ports. When sensor input 0~7 (S0~7) are touched, DIO0~7 generate 'H'. Otherwise, DIO0~7 generate 'L'.
- With the configuration of extended GPIO ports, data stored in GPIO REG L register (0x2D) are transmitted to DIO0~7 when the GPIO Direction L register (0x31) are configured as Output ports or data stored in GPIO IN L register (0x7A) are read to see the status of DIO0~7 when the GPIO Direction L register (0x31) are configured as Input ports.

4.16 GPIO Configuration H (0x30)

B7	B6	B5	B4	B3	B2	B1	B0
				DIO 11	DIO 10	DIO 9	DIO 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register configures the characteristics of DIO8~11 ports as either Touch output ports or extended GPIO ports.
- The bits with ‘L’ as default are configured as Touch Output ports and the bits with ‘H’ are configured as extended GPIO ports.
- With the configuration of Touch output ports, the status of sensor input pins is reflected to these ports. When sensor input 8~11 (S8~11) are touched, DIO8~11 generate ‘H’. Otherwise, DIO8~11 generate ‘L’.
- With the configuration of extended GPIO ports, data stored in GPIO REG H register (0x2E) are transmitted to DIO8~11 when the GPIO Direction H register (0x32) are configured as Output ports or data stored in GPIO IN H register (0x7B) are read to see the status of DIO8~11 when the GPIO Direction H register (0x32) are configured as Input ports.
- **B7, B6, B5 and B4:** Reserved

4.17 GPIO Direction L (0x31)

B7	B6	B5	B4	B3	B2	B1	B0
DIO 7	DIO 6	DIO 5	DIO 4	DIO 3	DIO 2	DIO 1	DIO 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register determines the direction of DIO0~7 ports as either input ports or output ports when DIO0~7 are configured as extended GPIOs by assigning ‘H’ into GPIO Configuration L register (0x29).
- If a bit is set to ‘H’, its GPIO direction is set to be Input, otherwise set to be Output as default.

4.18 GPIO Direction H (0x32)

B7	B6	B5	B4	B3	B2	B1	B0
				DIO 11	DIO 10	DIO 9	DIO 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- The register determines the direction of DIO8~11 ports as either input ports or output ports when DIO8~11 are configured as extended GPIOs by assigning ‘H’ into GPIO Configuration H register (0x30).
- If a bit is set to be ‘H’, GPIO direction is set to be Input, otherwise set to be Output as default.
- **B7, B6, B5 and B4:** Reserved

4.19 Control (0x33)

B7	B6	B5	B4	B3	B2	B1	B0
F2A	CLKSRC	P DIV[1]	P DIV[0]	N DIV	AIC	Force C	HOLD_C
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	1	0	0

B0: If B0 is set to 'L', it is auto calibration mode. Otherwise, initial calibration is done during the boot-up and AIC is waiting until Force Cal (B1) is asserted.

B1: If the status of B0 is 'H', and Force Cal bit is written with 1 by the host, calibration is executed only once then B1 becomes 'L' automatically. In other words, whenever B1 becomes 'H', calibration is done.

B2: Activates AIC function when this bit is 'H'. Otherwise AIC is off.

B3: Changes Sensor Clock Frequency

'L' ~ Sensor Clock Frequency is equal to System Clock/80.

'H' ~ Sensor Clock Frequency is equal to System Clock/160.

Initial calibration time depends on Sensor Clock Period as below:

Initial Calibration Time = Sensor Clock Period x 150

For example, if Sensor Clock is set to 20KHz, then initial calibration will take 0.05msec x 150 = 7.5msec.

B4~5: Internal Analog Clock is 1.6MHz

B[5:4] == 00: System Clock = Internal Analog Clock / 1 = 1.6MHz

B[5:4] == 01: System Clock = Internal Analog Clock / 2 = 800KHz

B[5:4] == 10: System Clock = Internal Analog Clock / 4 = 400KHz

B[5:4] == 11: System Clock = Internal Analog Clock / 8 = 200KHz

B6: If this bit is set to be 'L', internal OSC is used. Otherwise, external OSC is used. Generally this bit is set to be 'L'.

B7: If this bit is set to be 'H', power state is always in ACTIVE mode. If power state is in IDLE mode, it becomes ACTIVE mode.

4.20 Interrupt Mask (0x34)

B7	B6	B5	B4	B3	B2	B1	B0
	Interrupt Mask Register						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	1	0	1	1	0	0	0

- It is the Interrupt Source Masking Register when using GINT.
- Each bit specifies the type of interrupt.
- The bits with ‘H’ can mask corresponding interrupts and GINT is not generated even if events occur.
- User can activate necessary interrupts by unmasking corresponding bits among seven interrupt sources below.

B0: Touch Interrupt Source Mask Bit

B1: Active-to-Idle Interrupt Source Mask Bit

B2: Idle-to-Active Interrupt Source Mask Bit

B3: Reserved

B4: Reserved

B5: EINT Interrupt Source Mask Bit

B6: End of Calibration Interrupt Source Mask Bit

B7: Reserved

4.21 Interrupt Clear (0x35)

B7	B6	B5	B4	B3	B2	B1	B0
	Interrupt Clear Register						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- It is the Interrupt Clear Register when using GINT.
- This register is used to clear interrupt after completing interrupt service when GINT occurred.
- By assigning ‘H’ into the bit, the corresponding interrupt is cleared.
- It is automatically recovered to ‘L’ to generate next interrupt hence host MCU does not need to set it back to ‘L’.

B0: Touch Interrupt Source Clear Bit

B1: Active-to-Idle Interrupt Source Clear Bit

B2: Idle-to-Active Interrupt Source Clear Bit

B3: Reserved

B4: Reserved

B5: EINT Interrupt Source Clear Bit

B6: End of Calibration Interrupt Source Clear Bit

B7: Reserved

4.22 Interrupt Edge (0x36)

B7	B6	B5	B4	B3	B2	B1	B0
	Interrupt Edge Register						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
1	1	1	1	1	1	1	1

- This register specifies how to detect interrupt by level triggered or edge triggered.
- All bits are set to ‘H’ for edge triggered (Default).

B0: Touch Interrupt Edge Bit

B1: Active-to-Idle Interrupt Edge Bit

B2: Idle-to-Active Interrupt Edge Bit

B3: Reserved

B4: Reserved

B5: EINT Interrupt Edge Bit

B6: End of Calibration Interrupt Edge Bit

B7: Reserved

4.23 Control 2 (0x37)

B7	B6	B5	B4	B3	B2	B1	B0
M RST		BGEB	LDEB	Filter En		Beep En	INT Pol
R/W		R/W	R/W	R/W		R/W	R/W
0	0	0	0	0	0	0	0

B0: This bit sets Interrupt Polarity.

If 'H', GINT and TINT become falling edge, otherwise they are rising edge (Default).

B1: This bit is to enable Beep through BEEP pin. Default is 'L' and beep is disabled.

B2: Reserved

B3: This bit is to control FILTER On/Off to get more stable touch outputs. Default setting is 'L' and filter is OFF.

B4: This is an Internal LDO On/Off bit. Default is 'L' and Internal LDO is On.

B5: This is to control On/Off of internal BIAS block. Default is 'L' and BIAS block is On. This bit must be 'L' when using Internal LDO.

B6: Reserved

B7: This bit is to synchronize Sensor clock frequencies when using multiple FMA1127 chips and external clock. If this bit is set 'H', RESET signal should be transferred to DIO0 port but register values are not changed by this RESET signal. 'L' is default.

4.24 Beep Period (0x38)

B7	B6	B5	B4	B3	B2	B1	B0
Beep Period Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register determines Beep duration when detecting the change of touch status. Beep sound is generated through BEEP pin.

Beep Period (msec) = System CLK Period x (Beep Time REG x 8)

4.25 Beep Frequency (0x39)

B7	B6	B5	B4	B3	B2	B1	B0
Beep Frequency Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register determines the frequency of Beep sound when detecting the change of touch status.

Beep Frequency (Hz) = System CLK / (Beep Frequency REG x 2)

4.26 Calibration Interval (0x3A)

B7	B6	B5	B4	B3	B2	B1	B0
Calibration Interval Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	1	1	0	0	0	0

- This register sets the Auto Calibration Interval.
- If Sensor Clock is 20KHz and Register value is 100, calibration is done every 250 msec.

Calibration Interval = Sensor CLK Period x Register Value x 50

- **CAUTION:** During APIS mode, if Calibration Interval is set to be less than the Integration Time, touch output will never be generated because touch output is hold during calibration. Therefore, do not set Calibration Interval Register to a value smaller than 1/50 of the Integration Time register.

4.27 External Interrupt Enable (0x3B)

B7	B6	B5	B4	B3	B2	B1	B0
DIO 7	DIO 6	DIO 5	DIO 4	DIO 3	DIO 2	DIO 1	DIO 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register activates interrupts when detecting data changes in DIO0~7 (B0~B7) ports.
- The interrupt becomes active when the corresponding bit is ‘H’. Otherwise, it is disabled (Default).

4.28 External Interrupt Polarity (0x3C)

B7	B6	B5	B4	B3	B2	B1	B0
EINT Polarity Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- In FMA1127DA, this register determines DIO interrupt polarity. By default (‘L’), it is rising edge. Otherwise, it is falling edge.
- In FMA1127DC, this register is **RESERVED** because EINT is active both rising edge and falling edge when it configured as Edge Triggered. Default value is 0.

4.29 Filter Period (0x3D)

B7	B6	B5	B4	B3	B2	B1	B0
FILTER Period Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- Filter is the 2nd filter running with APIS. Filter is enabled first by B3 of Control2 Register (0x37).

- This register sets an additional filter period like Integration Time in APIS to ensure more stable touch output by using APIS outputs as filter inputs.
- It also can make additional delay of touch output when using high-speed sensor clock. For example, if this register is set to 10 and APIS touch output is generated every 5msec, the final touch output will be updated every 50msec.

4.30 Filter Threshold (0x3E)

B7	B6	B5	B4	B3	B2	B1	B0
FILTER Threshold Register							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

- This register determines threshold level of filtered touch output.
- Touch output is available only when accumulated value of APIS outputs during Filter Period is greater than or equal to FILTER Threshold value.
- The recommended Filter Threshold value is greater than but near to 50% of Filter Period. For example, if Filter Period is set to 10 or 11, the recommended value of Filter Threshold is 6.

4.31 CONTROL3 (0x3F)

B7	B6	B5	B4	B3	B2	B1	B0
			EINT_SRC	ECLK_EN		E_SRC	OAC
			R/W	R/W		R/W	R/W
0	0	0	1	1	0	0	0

- This register is available in the FMA1127DC version only.

B0: I2C out of address check enable: Default is ‘L’, Disable

By enabling this bit, MCU will be informed with NACK when it attempts to access invalid address area. If it is not enabled as default, ACK is always reported even if it attempts to access invalid address area and nothing happened internally in I2C write operation or unknown data will be transferred to MCU in I2C read operation.

B1: EINT Interrupt source select: default is ‘L’, De-bounced

‘L’: De-bounced GPIO input data, ‘H’: Direct GPIO input data

B2: Reserved

B3: External Interrupt Detection Block enable: Default is ‘H’, enable

In order to use EINT, you should always enable this bit. If EINT is not used, this bit can be disabled to save power consumption in active mode.

B4: This bit determines EINT generation method as either real-time edge trigger or buffered level trigger. (Default is ‘H’, Edge Trigger)

‘L’: The change of GPIO input data will be stored to FIFO memory up to 5 stages. And the host will be informed with an interrupt signal until FIFO is entirely empty. When interrupt triggers, the host can make FIFO empty by reading FIFO data. To

get FIFO data, B5 of EDGE_EN register (0x36) should be set to 'L'. Refer to GPIO_IN_L_FIFO register (0x7C) to get more information.

'H': By setting this bit to 'H', the host can detect interrupts at both falling and rising edge of GPIO status changes. If host clock speed is too slow to handle GPIO status changes, it can lose GPIO input data due to not storing GPIO data into FIFO.

4.32 Bounce Cancelling Period (0x40)

B7	B6	B5	B4	B3	B2	B1	B0
BOUNCE Cancelling Period							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	1	0	0	0	0

- This register is available in the FMA1127DC only.
- It is to set bounce cancelling period for GPIO input data.

  Xt = (1/ Sensor Clock Period) x 2 x BOUNCE Canceling Period

  With the above equation, if Sensor Clock is 20KHz and Bounce Cancelling Period is 100, any pulse width shorter than 10msec will be ignored.

4.33 Strength 0-11 (0x50 – 0x5B)

B7	B6	B5	B4	B3	B2	B1	B0
Strength Register							
R	R	R	R	R	R	R	R

- These registers are read-only and store Strength values of S0~S11 during every Integration Time in APIS mode.
- These are applicable only in APIS mode.
- The maximum value of Strength Register is equal to the value defined at Integration Time register (0x29).
- To enlarge Strength range, you need to change Integration Time first.

4.34 Calibrated Impedance 0-11 (0x5C – 0x67)

B7	B6	B5	B4	B3	B2	B1	B0
	Calibrated Impedance Register						
R	R	R	R	R	R	R	R

- These registers are read-only and store reference impedance of each touch input after AIC is performed.
- The range of Calibrated Impedance of S0~S11 is 0~99.
- Calibrated Impedances are available when all touch inputs become non-touch status.

- The value is equivalent to Impedance Value + ALPHA at the time of Touch OFF.
- The values read from these registers are noted in the tuning viewer program as ‘99-Register value’, so that a bigger value corresponds to a bigger impedance

4.35 Impedance 0-11 (0x68 – 0x73)

B7	B6	B5	B4	B3	B2	B1	B0
	Impedance Register						
R	R	R	R	R	R	R	R

- These registers are read-only and store current sensor impedances of S0~S11.
- You can monitor capacitance variation by touching the sensor pads S0~S11
- The values read from these registers are noted in the tuning view program as ‘99-Register value’, so that a bigger value corresponds to a bigger impedance and the value increases when the pad is touched

4.36 Status (0x74)

B7	B6	B5	B4	B3	B2	B1	B0
					Status Register		
R	R	R	R	R	R	R	R

- This read-only register is to store current Power State. Only B2~B0 are used and others are reserved.

B[2:0] = 000: Reset State

B[2:0] = 001: Active State

B[2:0] = 010: Idle State

B[2:0] = 100: Sleep State

4.37 Touch Byte L (0x75)

B7	B6	B5	B4	B3	B2	B1	B0
S7	S6	S5	S4	S3	S2	S1	S0
R	R	R	R	R	R	R	R

- This register is read-only and stores the status of sensor inputs (S0~S7).
- Host MCU can read this register only when TINT occurs. Or, it can read this register at any time or periodically if not using TINT.
- TINT is automatically cleared only when TOUCH Byte H (0x76) is read hence you MUST read Touch Byte H (0x76) even though S8~11 are not used. Otherwise, TINT never occurs.
- B7~B0 represent sensor output of S7~S0, respectively.

4.38 Touch Byte H (0x76)

B7	B6	B5	B4	B3	B2	B1	B0
				S11	S10	S9	S8
R	R	R	R	R	R	R	R

- This register is read-only and stores the status of sensor inputs (S8~S11).
- Host MCU can read this register only when TINT occurs. Or, it can read this register at any time or periodically if not using TINT.
- TINT is automatically cleared only when this register is read and another TINT is ready to provide when the status of sensor inputs are changed. TINT is never cleared until this register is read hence even if only S0~7 are used in the application, this register must be read to clear TINT.
- B3~B0 represent sensor outputs of S11~S8, respectively. Other upper bits are reserved.

4.39 Interrupt Pending (0x79)

B7	B6	B5	B4	B3	B2	B1	B0
	EOC	EINT			I2A	A2I	TINT
R	R	R	R	R	R	R	R

- This read-only register determines which interrupt should be handled among seven interrupt sources of GINT.
- You can check the interrupts only having unmasked bits with ‘L’ in Interrupt Mask register (0x34).
- Among unmasked interrupts, the bits with ‘H’ in this register are currently pending and need to be serviced by host MCU.

B0: Touch Interrupt Pending

B1: Active-to-Idle Interrupt Pending

B2: Idle-to-Active Interrupt Pending

B3: Reserved

B4: Reserved

B5: EINT Pending

B6: End Of Calibration Interrupt Pending

B7: Reserved

4.40 GPIO IN L (0x7A)

B7	B6	B5	B4	B3	B2	B1	B0
DIO 7	DIO 6	DIO 5	DIO 4	DIO 3	DIO 2	DIO 1	DIO 0
R	R	R	R	R	R	R	R

- This read-only register stores data to read current status of DIO0~7 ports only when corresponding bits in GPIO Configuration L register (0x2F) are configured as extended GPIOs by assigning 'H' and corresponding bits in GPIO Direction L register (0x31) are configured as Input by assigning 'H'.

4.41 GPIO IN H (0x7B)

B7	B6	B5	B4	B3	B2	B1	B0
				DIO 11	DIO 10	DIO 9	DIO 8
R	R	R	R	R	R	R	R

- This read-only register stores data to read current status of DIO8~11 ports only when corresponding bits in GPIO Configuration H register (0x30) are configured as extended GPIOs by assigning 'H' and corresponding bits in GPIO Direction H register (0x32) are configured as Input by assigning 'H'.
- **B7, B5, B5 and B4:** Reserved

4.42 GPIO IN L FIFO (0x7C)

B7	B6	B5	B4	B3	B2	B1	B0
DIO 7	DIO 6	DIO 5	DIO 4	DIO 3	DIO 2	DIO 1	DIO 0
R	R	R	R	R	R	R	R

- By activating FIFO memory in Control3 register, GPIO input data read from GPIO IN L register (0x7A) is to be stored at FIFO memory. Therefore, host MCU should read this register instead of GPIO IN L register when FIFO memory is set to be active.
- In the FMA1127DC, there are 5 steps of FIFO memory to store the events of GPIO change. If FIFO is full, first 4 events are preserved but the last 5th event is discarded when new event occurs.
- This register is available in the FMA1127DC only.

4.43 GPIO IN LBCU (0x7D)

B7	B6	B5	B4	B3	B2	B1	B0
DIO 7	DIO 6	DIO 5	DIO 4	DIO 3	DIO 2	DIO 1	DIO 0
R	R	R	R	R	R	R	R

- When B1 of Control3 register (E_SRC) is 'L', De-bounced GPIO input data is stored to this register. Therefore, it is recommended that you read this register when E_SRC bit is 'L'.
- This register is available in the FMA1127DC only.

4.44 BIAS OFF (0xFA)

B7	B6	B5	B4	B3	B2	B1	B0
BIAS OFF Register							
W	W	W	W	W	W	W	W
X	X	X	X	X	X	X	X

- This is a write-only register and data field is required, but value is ignored.
- It is Analog BIAS Block Power Off Control Register.
- BIAS OFF is used to reduce power consumption when external core power source is used and Power status is in Sleep mode because internal LDO drives some current by itself.

4.45 BIAS ON (0xFB)

B7	B6	B5	B4	B3	B2	B1	B0
BIAS ON Register							
W	W	W	W	W	W	W	W
X	X	X	X	X	X	X	X

- This is a write-only register and data field is required but value is ignored.
- This is the Analog BIAS Block Power On Control Register.
- When using external core power, power consumption can be minimized by turning off BIAS in sleep mode. On the other hand, to wake up from sleep mode BIAS block should be turned on by executing BIAS ON register.

4.46 Wakeup SLEEP (0xFC)

B7	B6	B5	B4	B3	B2	B1	B0
Wakeup SLEEP Register							
W	W	W	W	W	W	W	W
X	X	X	X	X	X	X	X

- This is a write-only register and data field is required but value is ignored.
- The device wakes up from Sleep mode by writing any value to this register.

4.47 Enter SLEEP (0xFD)

B7	B6	B5	B4	B3	B2	B1	B0
Enter SLEEP Register							
W	W	W	W	W	W	W	W
X	X	X	X	X	X	X	X

- This is a write-only register and data field is required but value is ignored.
- The device enters Sleep mode by writing any value to this register.

4.48 Cold Reset (0xFE)

B7	B6	B5	B4	B3	B2	B1	B0
Cold Reset Register							
W	W	W	W	W	W	W	W
X	X	X	X	X	X	X	X

- This is a write-only register and data field is required but value is ignored.
- Cold Reset initializes all blocks of FMA1127DC including Register block. Thus, all register values are reset to Default values. Only the register address is meaningful here.

4.49 Warm Reset (0xFF)

B7	B6	B5	B4	B3	B2	B1	B0
Warm Reset Register							
W	W	W	W	W	W	W	W
X	X	X	X	X	X	X	X

- This is a write-only register and data field is required but value is ignored.
- Warm Reset initializes all blocks of FMA1127DC except Register block. Therefore, the register values keep remained. Only the register address is meaningful here.

NOTE:

Please note, that when the Warm Reset register is written, the former FMA1127DA version of the TSC will immediately perform the desired reset operation. For this reason, the ACK signal for I2C communication might not be generated, so that the host SW should not wait for the ACK signal after issuing a warm/cold reset. The FMA1127DC performs the reset after generating the ACK signal, so that no special handling is required.

5 Appendix

5.1 Glossary

MCU	Microcontroller Unit
TSC	Touch Sensor Controller

5.2 Related Documents

More information to the FMA1127 Touch Sensor Controller can be found on the web at:
http://www.fujitsu.com/emea/services/microelectronics/tsc/

Additional information to other Fujitsu products can be found at:
http://www.fujitsu.com/emea/

