VN0003QIC04
12 Input Self Capacitance XY Touchscreen & Slider Controller

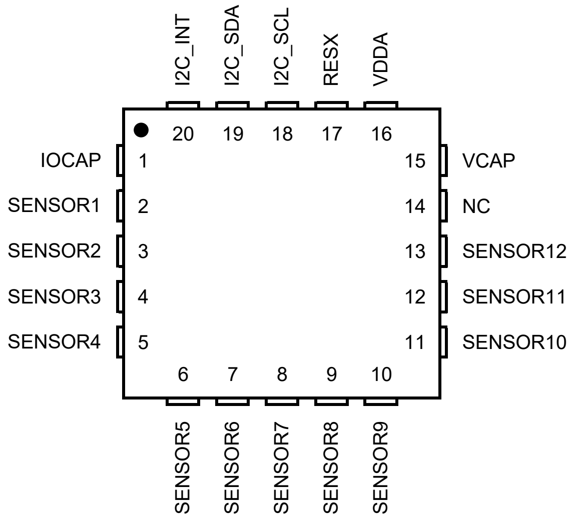
simpleTouch™ by Venntis

Overview

The patented simpleTouch™ 12 Input Self Capacitance XY Touchscreen & Slider Controller IC is a single chip touch controller which requires no tuning to work with a wide variety of substrate materials and thickness ranges. The IC is specifically designed as a XY touchscreen & slider controller for applications using up to 12 diamond shaped style electrodes, offers performance features of working with glass (or equivalent substrate) with thicknesses from 0.5mm to 4.0mm, works with water, saline and glove, and requires minimal external components for complete functionality. The IC can also be configured as a multi-element 1 dimensional linear slider and can also support 2 slider operation.

The communication interface to the IC is via standard I^{2}C bus and is configured as a slave device which communicates with a Master/Host. The configuration for the touch controller is performed by the Host writing configuration data to the simpleTouch™ Control Register Structure and simpleTouch™ Sensor Order Pointers table which are retained upon shut down and subsequent power on. In the absence of programming the simpleTouch™ Control Register Structure and simpleTouch™ Sensor Order Pointers, the IC will configure to the default values listed on those specific I^{2}C command pages.

The IC will currently only operate in self-capacitance mode and will output absolute XY data upon touch detection along with a gesture code upon touch deactivation. The touch controller will power up into Active mode which actively scans the programmed XY electrodes at $T_{activescanrate}$. For power sensitive applications, the power mode can be changed between Active and Monitor modes by the Host to reduce power consumption. Upon detection of a touch, the IC will move from Monitor mode into Active mode automatically and generate an interrupt in which the Host can read to initiate the I^{2}C communication for transfer of XY position and gesture data.



20 Pin QFN Package

Features

- Patented simpleTouch™ technology which inherently overcomes manufacturing and environmental variances—no tuning.
- Up to 12 input XY self capacitance solution (minimum of 3 electrodes in both XY directions), 3 - 12 input single slider, or 2 sliders (minimum of 3 electrodes each).
- Supports single and multiple layer XY transparent touch screens, non-transparent touch matrices, and potential absolute XY coordinates.
- Supports I^{2}C communication with a bus speed of 400kbps with interrupt notification of detected touch.
- Simple configuration structure and control attributes to the device.
- Electrodes can be designed with etched copper, printed silver, ITO (Indium Tin Oxide), AgNW (silver nanowire), and more.
- Operating Modes:
  - Monitor Mode: all electrodes are scanned and processed to maintain low power consumption to determine a touch event.
  - Active Mode: after Monitor Mode detects a touch, Active Mode will be initiated for processing XY coordinate data to be reported by I^{2}C.

Part Number Information

Part Number Format: VN0003*XYZnn*

X (Packaging): **Q** = 20 Pin QFN

Y (Temperature Range): **I** = -40C — +85C

Z (Grade): **C** = Consumer

nn (Revision): Numerical Chip Revision

Electrical Characteristics

Absolute Maximum Ratings*

Designation	Item	Condition	Rated Value	Unit
V_{dda}	Supply Voltage	V_{dda} with respect to V_{ss}	-0.3 — +5.0	V
T_{str}	Storage Temperature		-55 — +150	°C

* Exceeding the absolute maximum ratings may result in permanent damage to the device

Operating Conditions

Designation	Item	Condition	Rated Value			Unit
			Min	Nominal	Max	
V_{dda}	Supply Voltage		2.7	3.30	3.6	V
Idd	Supply Current (Monitor Mode)	3.30v			65	uA
Idd	Supply Current (Active Mode)	3.30v	755			uA
T_{amb}	Ambient Temperature	Under Bias	-40		+85	°C
$T_{activescanrate}$	Active Mode Scan Rate			200		Hz
$T_{monitorscanrate}$	Monitor Mode Scan Rate			16		hz
$T_{reportrate}$	Data XY Report Rate			200		hz
$T_{stuckkeytimeout}$	Active Stuck Key Timeout			30		seconds
T_{i2c_clock}	I^2C Bus Clock				400	kbps

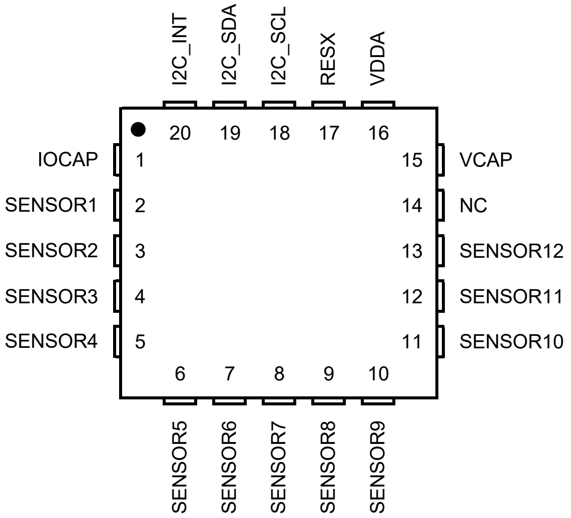
AC Characteristics

Designation	Item	Condition	Rated Value			Unit
			Min	Nominal	Max	
	Digital Inputs / Outputs					
VIH	Input High Voltage		0.7 x IOVCC		3.6	V
VIL	Input Low Voltage		GND		0.3 x IOVCC	V
VOH	Output High Voltage		0.7 x IOVCC		IOVCC	V
VOL	Output Low Voltage		GND		0.3 x IOVCC	V

Notes:

1) IOVCC = VDDA or 1.8v

Pinout Diagram



Pin	Name	Function	Description / Connection
1	IOCAP	Power	Bypass with 1.0uF Ceramic Capacitor
2	SENSOR1	Touch Sensor Input	Connect to Sensor 1 Electrode
3	SENSOR2	Touch Sensor Input	Connect to Sensor 2 Electrode
4	SENSOR3	Touch Sensor Input	Connect to Sensor 3 Electrode
5	SENSOR4	Touch Sensor Input	Connect to Sensor 4 Electrode
6	SENSOR5	Touch Sensor Input	Connect to Sensor 5 Electrode
7	SENSOR6	Touch Sensor Input	Connect to Sensor 6 Electrode
8	SENSOR7	Touch Sensor Input	Connect to Sensor 7 Electrode
9	SENSOR8	Touch Sensor Input	Connect to Sensor 8 Electrode
10	SENSOR9	Touch Sensor Input	Connect to Sensor 9 Electrode
11	SENSOR10	Touch Sensor Input	Connect to Sensor 10 Electrode
12	SENSOR11	Touch Sensor Input	Connect to Sensor 11 Electrode
13	SENSOR12	Touch Sensor Input	Connect to Sensor 12 Electrode
14	NC	No Connect	Leave Unconnected
15	VCAP	Power	Bypass with 1.0uF Ceramic Capacitor
16	VDDA	Power	Supply Voltage Connection
17	RESX	Reset Input (Active Low)	Allows for Manual Reset of IC
18	I2C_SCL	Interface	I^{2}C Clock Bus
19	I2C_SDA	Interface	I^{2}C Data Bus
20	I2C_INT	Digital Output (Active Low)	Touch Status Change Signal Line
EP	VSS	Power	Ground Connection

Application Information

Block Diagrams

Figure 1 shows a block diagram highlighting the main connections for the Venntis® simpleTouch™ IC in a 6x6 XY matrix with Table 1 showing its corresponding Sensor Order Pointers for the Master Write simpleTouch™Sensor Order I^{2}C command setup. Figure 2 shows the connections for a 3x6 XY matrix with Table 2 showing its corresponding Sensor Order Pointers. In both examples the XY electrodes are connected to the IC in any order with the connections using IC pins from SENSOR1 - SENSOR12 with no unused inputs in between the electrodes. Any unused inputs must be placed before or after all column and row connections have been made to the IC. The Sensor Order Pointer tells the simpleTouch™ IC the order in which the columns and rows have been connected to the SENSOR inputs. This allows a designer more flexibility with trace routing between the electrode sensors and IC. The order of the Sensor Order Pointers are very important and must adhere to listing all columns first and then all rows directly following starting with SENSOR 1 through SENSOR 12. The SENSOR input number for the column in which you decide is Col 1 is listed first, Col 2 second, etc. until all columns are listed, then the SENSOR input number for each Row is listed beginning with the row in which you decide is Row 1, then Row 2, etc until all rows are listed. Note: the pointer number for SENSOR 1 = 1, SENSOR 2 = 2,….. SENSOR 12 = 12 and any unused inputs must be coded with 0. To generate the Sensor Order Pointers, refer to examples shown in Tables 1-5. For XY touchscreens, the supported Number of Columns are 3 - 9 and the supported Number of Row values are 3 - 9. Note: for XY interpolation to function correctly, their must be a minimum of 3 columns and/or 3 rows used. Figures 3-5 show diagrams for slider applications. The slider electrodes are connected to the IC sequentially from SENSOR1 - SENSOR12 with no unused inputs in between the electrodes. For slider applications, the minimum number of supported elements is 3. It is possible to connect 2 sliders to the simpleTouch™ IC as shown in Figure 5. Communication to the IC is accomplished through a standard master-slave I^{2}C interface with the addition of external pullup resistors to the bus. The I2C_INT line is active low and requires an external pullup resistor to VDD.

Note: For XY operation, X = 0 & Y = 0 will always be located at the intersection of the 1st column (C1 as shown below) and the 1st Row (R1 as shown below). For slider operation, 0 will always be located at the element which is COL1 in the Sensor Order Pointer. Note: in Figure 3, the slider output data will be read from the row (Y Position) I^{2}C data.

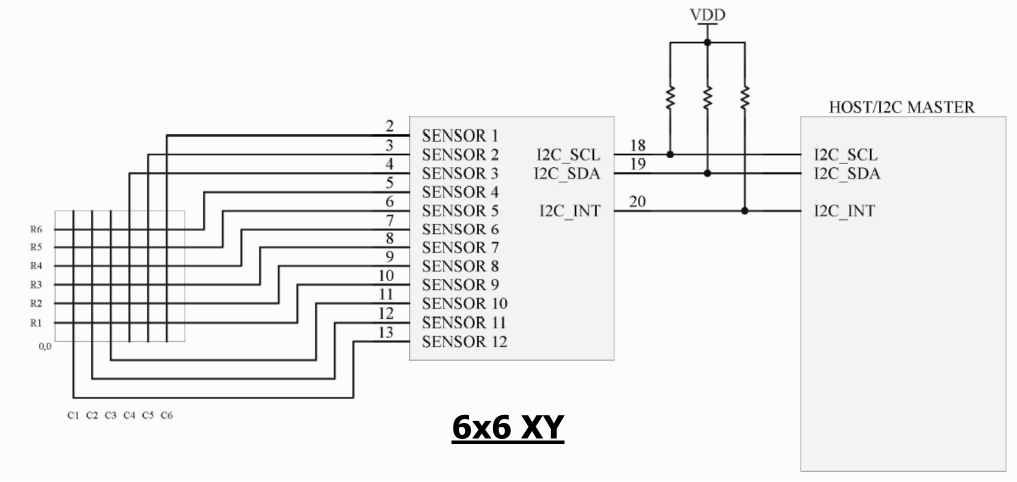


Figure 1

# of Columns:	6											
# of Rows:	6											
Sensor Pointer Order:	SENSOR 1	SENSOR 2	SENSOR 3	SENSOR 4	SENSOR 5	SENSOR 6	SENSOR 7	SENSOR 8	SENSOR 9	SENSOR 10	SENSOR 11	SENSOR 12
Example Col/Row:	Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Row 1	Row 2	Row 3	Row 4	Row 5	Row 6
Example IC Pin:	13	12	11	4	3	2	10	9	8	7	6	5
Example Pin Description:	SENSOR 12	SENSOR 11	SENSOR 10	SENSOR 3	SENSOR 2	SENSOR 1	SENSOR 9	SENSOR 8	SENSOR 7	SENSOR 6	SENSOR 5	SENSOR 4
Example Sensor Pointer:	12	11	10	3	2	1	9	8	7	6	5	4

Table 1

Application Information

Block Diagrams

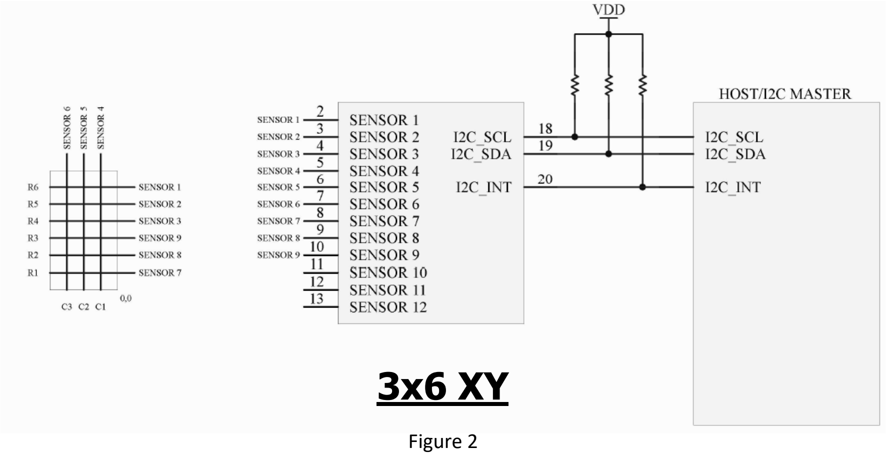


Figure 2

# of Columns:	3										
# of Rows:	6										
Sensor Pointer Order:	SENSOR 1	SENSOR 2	SENSOR 3	SENSOR 4	SENSOR 5	SENSOR 6	SENSOR 7	SENSOR 8	SENSOR 9	SENSOR 10	SENSOR 11	SENSOR 12
Example Col/Row:	COL 1	COL 2	COL 3	ROW 1	ROW 2	ROW 3	ROW 4	ROW 5	ROW 6			
Example IC Pin:	5	6	7	8	9	10	4	3	2			
Example Pin Description:	SENSOR 4	SENSOR 5	SENSOR 6	SENSOR 7	SENSOR 8	SENSOR 9	SENSOR 3	SENSOR 2	SENSOR 1			
Example Sensor Pointer:	4	5	6	7	8	9	3	2	1	0	0	0

Table 2

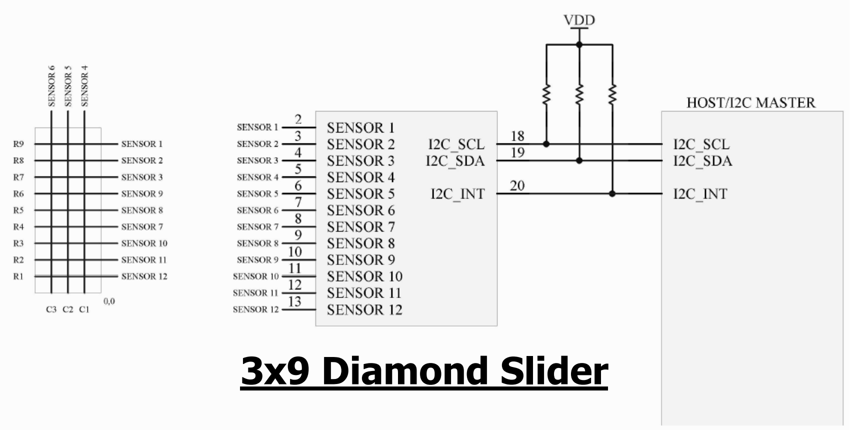


Figure 3

# of Columns:	3										
# of Rows:	9										
Sensor Pointer Order:	SENSOR 1	SENSOR 2	SENSOR 3	SENSOR 4	SENSOR 5	SENSOR 6	SENSOR 7	SENSOR 8	SENSOR 9	SENSOR 10	SENSOR 11	SENSOR 12
Example Col/Row:	COL 1	COL 2	COL 3	ROW 1	ROW 2	ROW 3	ROW 4	ROW 5	ROW 6	ROW 7	ROW 8	ROW 9
Example IC Pin:	5	6	7	13	12	11	8	9	10	4	3	2
Example Pin Description:	SENSOR 4	SENSOR 5	SENSOR 6	SENSOR 12	SENSOR 11	SENSOR 10	SENSOR 7	SENSOR 8	SENSOR 9	SENSOR 3	SENSOR 2	SENSOR 1
Example Sensor Pointer:	4	5	6	12	11	10	7	8	9	3	2	1

Table 3

Block Diagrams

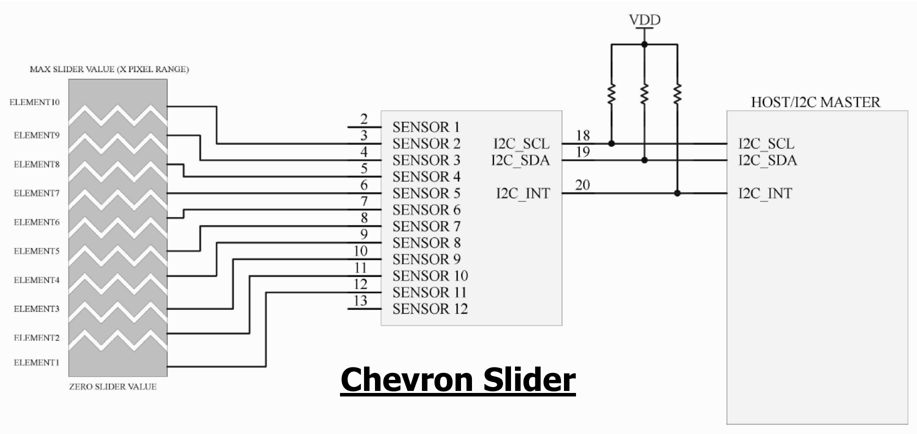


Figure 41

# of Columns:	10
# of Rows:	0

Sensor Pointer Order:	SENSOR 1	SENSOR 2	SENSOR 3	SENSOR 4	SENSOR 5	SENSOR 6	SENSOR 7	SENSOR 8	SENSOR 9	SENSOR 10	SENSOR 11	SENSOR 12
Example Col/Row:		Col 1	Col 2	Col 3	Col 4	Col 5	Col 6	Col 7	Col 8	Col 9	Col 10	
Example IC Pin:		12	11	10	9	8	7	6	5	4	3	
Example Pin Description:	SENSOR 12	SENSOR 11	SENSOR 10	SENSOR 9	SENSOR 8	SENSOR 7	SENSOR 6	SENSOR 5	SENSOR 4	SENSOR 3	SENSOR 2	SENSOR 1
Example Sensor Pointer:	0	11	10	9	8	7	6	5	4	3	2	0

Table 4

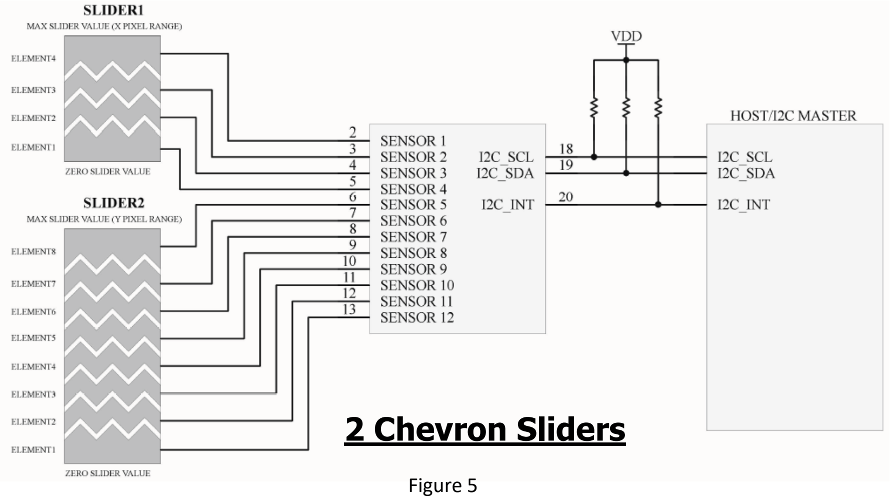


Figure 5

# of Columns:	4
# of Rows:	8

Sensor Pointer Order:	SENSOR 1	SENSOR 2	SENSOR 3	SENSOR 4	SENSOR 5	SENSOR 6	SENSOR 7	SENSOR 8	SENSOR 9	SENSOR 10	SENSOR 11	SENSOR 12
Example Col/Row:	S1 - Col 1	S1 - Col 2	S1 - Col 3	S1 - Col 4	S2 - Row 1	S2 - Row 2	S2 - Row 3	S2 - Row 4	S2 - Row 5	S2 - Row 6	S2 - Row 7	S2 - Row 8
Example IC Pin:	5	4	3	2	13	12	11	10	9	8	7	6
Example Pin Description:	SENSOR 4	SENSOR 3	SENSOR 2	SENSOR 1	SENSOR 12	SENSOR 11	SENSOR 10	SENSOR 9	SENSOR 8	SENSOR 7	SENSOR 6	SENSOR 5
Example Sensor Pointer:	4	3	2	1	12	11	10	9	8	7	6	5

Table 5

Application Information

Reference Design

Figure 6 shows a reference design for the IC used in a 7 column by 5 row XY touchscreen design. All ceramic chip capacitors should be of good quality with a temperature coefficient of X7R (class 2) or better. The external I^{2}C pullup resistors are omitted from this reference design, but the customer is responsible for including into their overall system design. R1 I2C_INT pullup is required. C5 is not required, but is considered good engineering practice. Figure 7 shows the recommended diamond pattern range and spacing for XY electrodes. Figure 8 shows the recommended chevron pattern range and spacing for slider electrodes. Slider designs can utilize other shapes such as rectangles and are not limited to chevron shaped electrodes provided the overall electrode size is in relation to what is shown in Figure 8.

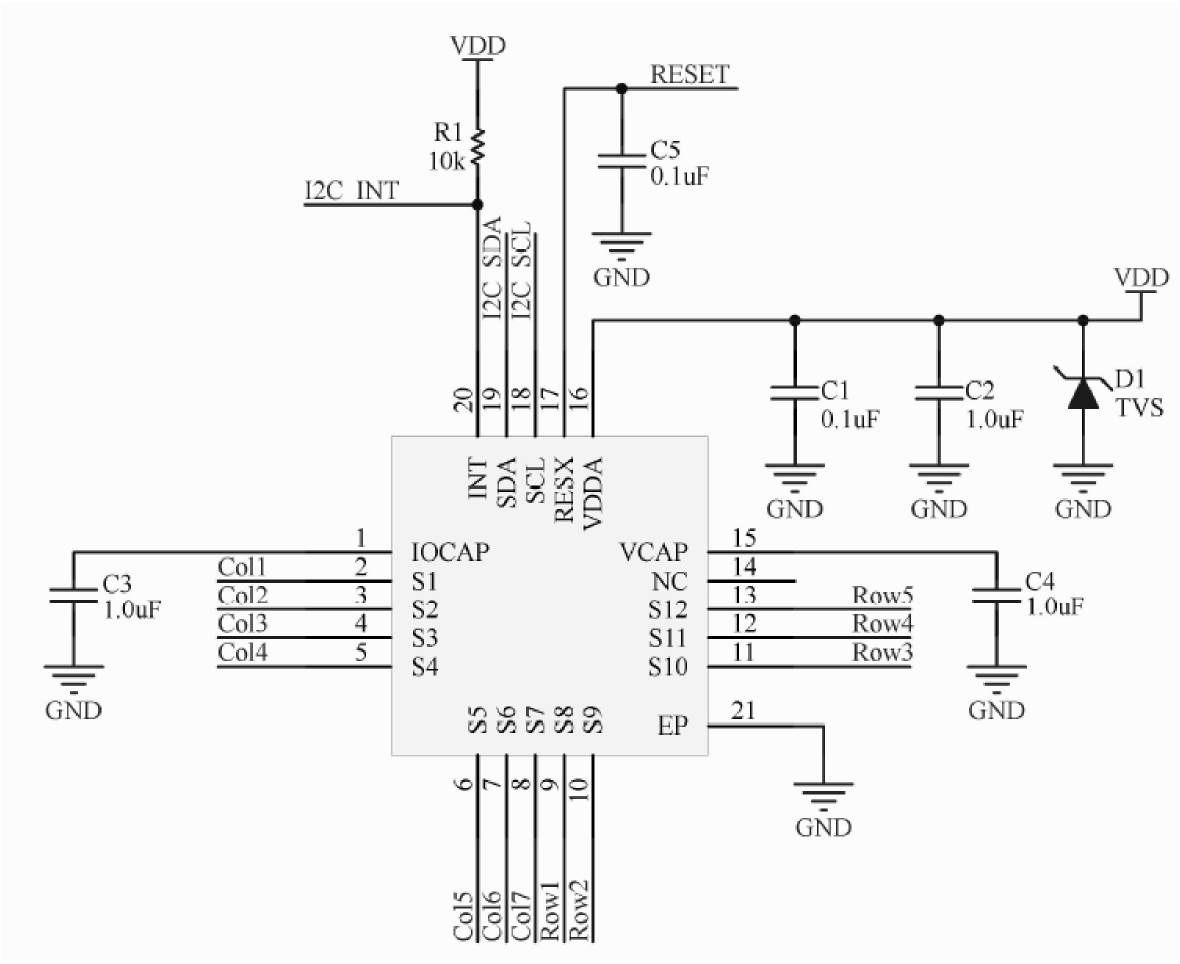


Figure 6

Reference Design

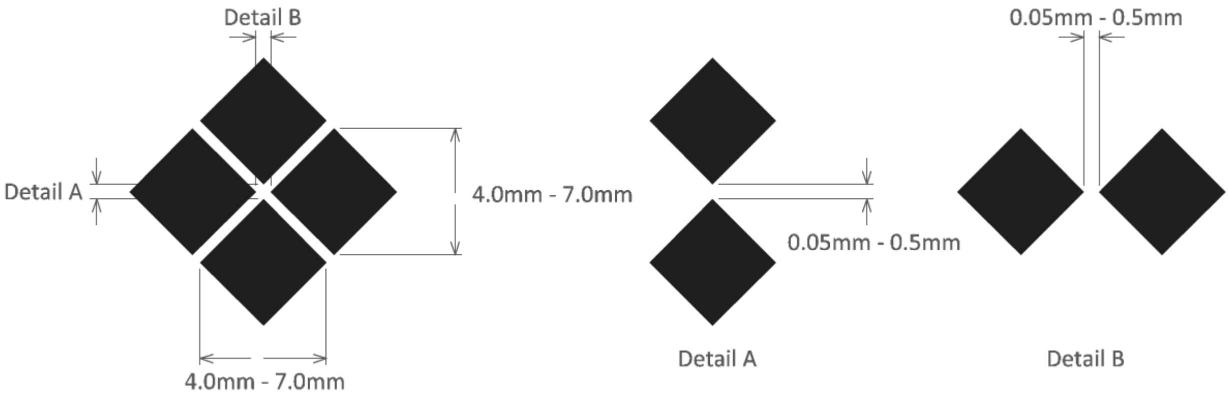


Figure 7

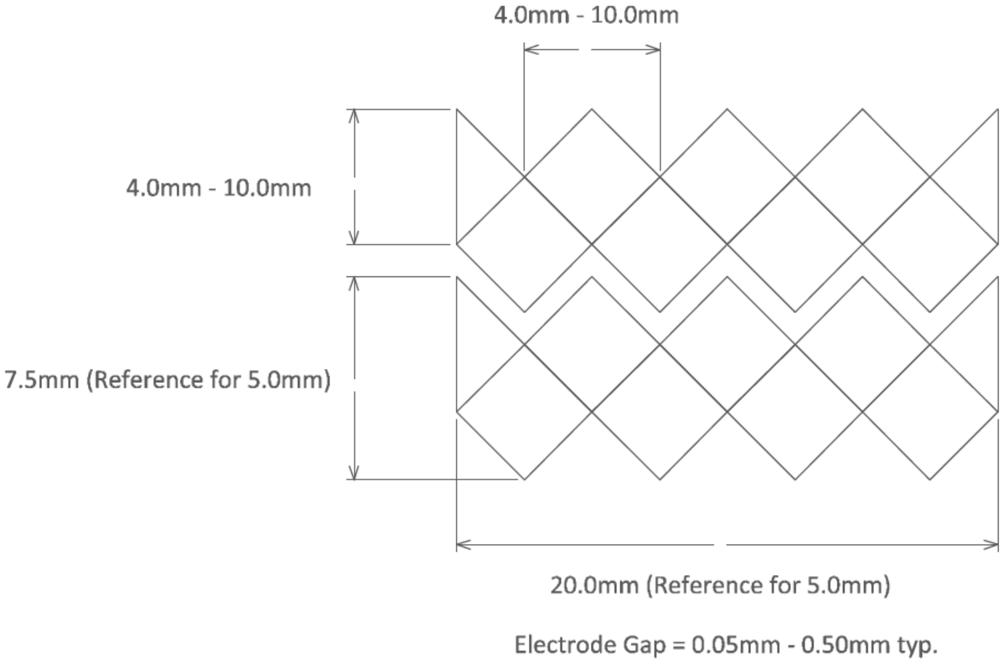

4.0mm - 10.0mm
4.0mm - 10.0mm
7.5mm (Reference for 5.0mm)
20.0mm (Reference for 5.0mm)
Electrode Gap = 0.05mm - 0.50mm typ.

Figure 8

Reference simpleTouch™ Gesture Algorithm

The Following shows reference pseudo code for the simpleTouch™ gesture algorithm implementation in the VN0003QIC04 IC. This reference pseudo code is provided as an example for similar customer implementation in the Host controller.

```
#include <stdio.h>
#include <stdlib.h>

#define NONE 0
#define SWIPE_LEFT 1
#define SWIPE_RIGHT 2
#define SWIPE_DOWN 3
#define SWIPE_UP 4
#define CLICK 5

#define CLICK_WINDOW 5 //5 pixels
#define CLICK_TIMEOUT 15 //75 ms
#define SWIPE_THRESHOLD 0 //0 pixels
#define SWIPE_TIMEOUT 2000 //2000 ms

// Define a structure to hold the data points
typedef struct {
    int X;
    int Y;
    int time;
} DataPoint;

// Function to detect the gesture based on the data points
int detect_Gesture(DataPoint downdp, DataPoint liftdp)
{
    int deltaX = liftdp.X - downdp.X;
    int deltaY = liftdp.Y - downdp.Y;
    int deltaTime = liftdp.time - downdp.time;

    if (abs(deltaX) < CLICK_WINDOW && abs(deltaY) < CLICK_WINDOW)
    {
        if (deltaTime < CLICK_TIMEOUT) {
            return CLICK;
        }
        else {
            return NONE;
        }
    }
    else
    {
        if (deltaTime < SWIPE_TIMEOUT)
        {
            if (abs(deltaX * (yPixelRange / xPixelRange)) >= abs(deltaY) && abs(deltaX) > SWIPE_THRESHOLD)
            {
                if (deltaX > 0) {
                    return SWIPE_LEFT;
                }
```

Reference simpleTouch™ Gesture Algorithm continued

```
                else {
                    return SWIPE_RIGHT;
                }
            }
            else if (abs(deltaY) >= abs(deltaX * (yPixelRange / xPixelRange)) && abs(deltaY) > SWIPE_THRESHOLD)
            {
                if (deltaY > 0) {
                    return SWIPE_DOWN;
                }
                else {
                    return SWIPE_UP;
                }
            }
            else
            {
                return NONE;
            }
        }
        else
        {
            return NONE;
        }
    }
}

int main()
{
    // Example usage of the detect_Gesture function
    DataPoint downdp = {0, 0, 1000};  // Example starting data point, first DataPoint when INT line goes LOW
    DataPoint liftdp = {10, 15, 2000};  // Example ending data point, first DataPoint when INT line goes HIGH

    int gesture = detect_Gesture(downdp, liftdp);

    switch (gesture)
    {
    case NONE:
        printf("No gesture detected.\n");
        break;
    case SWIPE_LEFT:
        printf("Swipe left detected.\n");
        break;
    case SWIPE_RIGHT:
        printf("Swipe right detected.\n");
        break;
    case SWIPE_DOWN:
        printf("Swipe down detected.\n");
        break;
    case SWIPE_UP:
        printf("Swipe up detected.\n");
        break;
```

Application Information

Reference simpleTouch™ Gesture Algorithm continued

```
    case CLICK:
        printf("Click detected.\n");
        break;
    default:
        printf("Unknown gesture.\n");
        break;
    }

    return 0;
}
```

I^{2}C Interface

The Venntis® simpleTouch™ IC is communicated with exclusively through I^{2}C commands. Unless otherwise stated, the default I^{2}C address for the simpleTouch™ IC is 0xC6 (8-bit addressing) or 0x63 (7-bit addressing) with a data rate = 400kbps. Operation is in Slave mode only. An active low signal is generated on the I2C INT pin when a change in touch state is detected and XY position and gesture data is ready for the Master/Host to request. The user can choose to use the I2C INT pin for automatic notification of a change in touch status or ignore the I2C INT and manually poll the IC for the touch status at any time (note: if polling method is used, it is recommended to request touch status and/or XY data no faster than half the $T_{reportrate}$ to avoid needlessly over burdening the IC). The action of the I2C INT pin is determined by the mode selected via the Master Write Interrupt Mode I^{2}C command. The Master can update the simpleTouch™ control register and simpleTouch™ Sensor Order Pointers registers by issuing a Master Write command with associated parameters at any time after POR.

NOTE:

The VN0003QIC04 simpleTouch™ IC requires a Stop bit after all Master write and read commands (a Restart bit must not be used). Also, a 100us minimum delay is required between a Master "write stop" and "read start" for the IC to prepare the register data for transfer (please see individual command pages for details).

Command Set

The VN0003QIC04 IC is configured and operated using the commands shown in the table below.

Command	Name
0x02	Master Request Touch Status
0x03	Master Request XY Data
0x18	Master Request Power Mode
0x19	Master Write Power Mode
0xA2	Maser Request Interrupt Mode
0xA4	Master Write Interrupt Mode
0xA6	Master Request Firmware Version
0xA8	Master Request Chip ID Number
0xD1	Master Write simpleTouch™ Configuration Bytes
0xD2	Master Request simpleTouch™ Configuration Bytes
0xD3	Master Write simpleTouch™ Sensor Order Pointers
0xD4	Master Request simpleTouch™ Sensor Order Pointers
0xD6	Master Request Serialization Parameters

Example I^{2}C Write/Read Sequence

Touch Status	I^{2}C CMD/Data	Description
0	0x02	Master Request Touch Status
0	0x00	Slave Response With No Touch
.	.	.
1	0x02	Master Request Touch Status
1	0x01	Slave Response With Touch
1	0x03	Master Request Event Flag & XY Data
1	0x80, 0xnn, 0x80, 0xnn, 0x05	Slave Response With Event Flags, XY Data, Gesture Code
.	.	.
0	0x02	Master Request Touch Status
0	0x00	Slave Responce With No Touch

Master Request Touch Status

Command Structure

Byte 1			
	S	0xC6 (Address)	A/N
Byte 2			
		0x02 (Command)	A/N
	P		
		Required 100uS delay	
Byte 3			
	S	Touch Status Value	A/N
	P		

Command Description

This command is used to **Read** the number of currently active touch points.

Touch Status Value:

0x00 = No Touch Detected

0x01 = Single Touch Detected

Note:

- Currently only 1 touch point is supported, therefore a 0x01 will always be returned for the Touch Status Value when a touch is detected.

Master Request Touch Status Example:

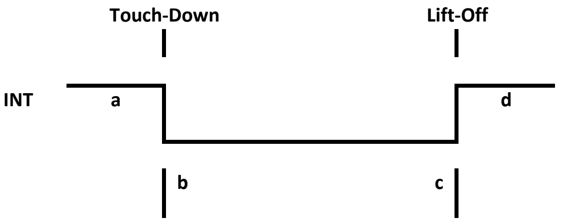
Master Out | Slave Out

Byte 1	0xC6	Address
Byte 2	0x02	Command
Byte 3	0x01	Touch Status Value

Master Request XY Data

Command Structure

Byte 1	S	0xC6 (Address)	A/N
Byte 2		0x03 (Command)	A/N
	P		
		Required 100uS delay	
Byte 3	S	Event Flag (bit7), X Position MSB (bits 6-0)	A/N
Byte 4		X Position LSB	A/N
Byte 5		Event Flag (bit7), Y Position MSB (bits 6-0)	A/N
Byte 6		Y Position LSB	A/N
Byte 7		Gesture Code	A/N
	P		



a: Gesture Code = previous gesture detected
b: Gesture Code = 0 (no gesture detected)
c: Gesture Code = current gesture detected
d: Gesture Code = gesture detected at point c

Master Request XY Data Example:

	Master Out	Slave Out	
Byte 1	0xC6		Address
Byte 2	0x03		Request XY Data
Byte 3		0x80	Event Flag & X Position MSB
Byte 4		0xA0	X Position LSB
Byte 5		0x80	Event Flag & Y Position MSB
Byte 6		0x0F	Y Position LSB
Byte 7		0x01	Gesture Code

Command Description

This command is used to **Read** the XY location of the currently active touch point and any Gestures detected. The Event Flag (bit 7 of X Position MSB and bit 7 of Y Position MSB) will be set when a valid touch is detected and clear when no touch is detected.

Gesture Code:

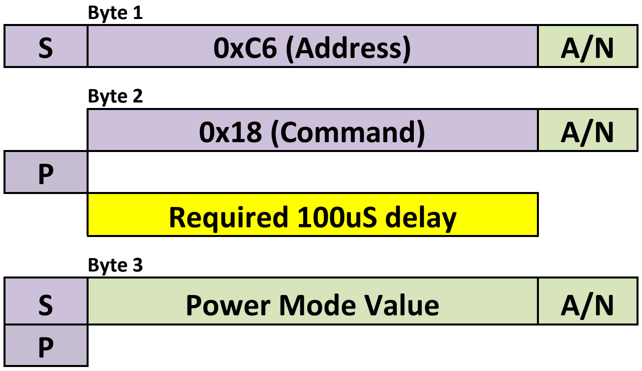
0 - No Gesture
1 - Swipe Left (Left to Right)
2 - Swipe Right (Right to Left)
3 - Swipe Down (Top to Bottom)
4 - Swipe Up (Bottom to Top)
5 - Click

Note:

- XY locations are not current unless Touch Status Value (Event Flag) = 1.
- Since gestures can only be calculated once a finger lift-off has been detected, the Gesture Code must be read AFTER the INT line transitions from a low to a high. The Gesture Code will remain unchanged until another finger touch-down is detected (INT line = low) at which point it will be set to 0 and remain 0 unless a valid gesture is detected.
- A Click is defined as touch-down and lift-off events which occur less than 300ms* apart and the change in XY locations between events is less than 10 positions.
- A Swipe is defined as touch-down and lift-off events which occur less than 2 seconds* apart and the change in XY locations between events is greater than 15 positions.
- Swipe Left = X location change > Y location change and the change is positive.
- Swipe Right = X location change > Y location change and the change is negative.
- Swipe Down = Y location change > X location change and the change is positive.
- Swipe Up = Y location change > X location change and the change is negative.
- * after Wakeup from Monitor Mode. During Wakeup mode, times may be longer.

Master Request Power Mode

Command Structure



Byte 1

S | 0xC6 (Address) | A/N

Byte 2

0x18 (Command) | A/N

P

Required 100uS delay

Byte 3

S | Power Mode Value | A/N

P

Command Description

This command is used to **Read** the current power mode of the simpleTouch™ IC.

Power Mode Value:

0x00 = Active Mode

0x01 = Idle Mode

Note:

- After issuing a Master Write Power Mode command to the IC, a minimum delay of 15 milliseconds is required prior to issuing a Master Request Power Mode command for valid Power Mode Value.

Master Request Power Mode Example:

Master Out	Slave Out

Byte 1	0xC6	Address
Byte 2	0x18	Command
Byte 3	0x01	Power Mode Value

Master Write Power Mode

Command Structure

	Byte 1	
S	**0xC6 (Address)**	**A/N**
	Byte 2	
	0x19 (Command)	**A/N**
	Byte 3	
	Power Mode Value	**A/N**
P		

Command Description

This command is used to **Write** the current power mode of the simpleTouch™ IC.

Power Mode Value:

0x00 = Active Mode

0x01 = Idle Mode

Note:

- After issuing a Master Write Power Mode command to the IC, a minimum delay of 15 milliseconds is required prior to issuing a Master Request Power Mode command for valid Power Mode Value.

Master Write Power Mode Example:

			Master Out	Slave Out
Byte 1	0xC6	Address		
Byte 2	0x19	Command		
Byte 3	0x01	Power Mode Value		

Master Request Interrupt Mode

Command Structure

	Byte 1	
S	**0xC6 (Address)**	**A/N**
	Byte 2	
	0xA2 (Command)	**A/N**
P		
	Required 100uS delay	
	Byte 3	
S	**Interrupt Mode Value**	**A/N**
P		

Command Description

This command is used to **Read** the mode of operation for the interrupt pin.

Interrupt Mode Value:

0x00 = Interrupt Polling Mode (default)

0x01 = Interrupt Trigger Mode

Note:

- In Interrupt Polling Mode, the active low I2C_INT line remains low for as long as there is a valid touch detected.
- In Interrupt Trigger Mode, the active low I2C_INT line remains low for 50 microseconds once a valid touch is detected and repeats at $T_{reportrate}$ rate as long as the valid touch is detected.

Master Request Interrupt Mode Example:			**Master Out**	**Slave Out**
Byte 1	0xC6	Address		
Byte 2	0xA2	Command		
Byte 3	0x01	Interrupt Mode Value		

Master Write Interrupt Mode

Command Structure

	Byte 1	
S	0xC6 (Address)	A/N
	Byte 2	
	0xA4 (Command)	A/N
	Byte 3	
	Interrupt Mode Value	A/N
P		

Command Description

This command is used to **Write** the mode of operation for the interrupt pin. Once written once, this value is retained for all POR's.

Interrupt Mode Value:

0x00 = Interrupt Polling Mode (default)

0x01 = Interrupt Trigger Mode

Note:

- In Interrupt Polling Mode, the active low I2C_INT line remains low for as long as there is a valid touch detected.
- In Interrupt Trigger Mode, the active low I2C_INT line remains low for 50 microseconds once a valid touch is detected and repeats at $T_{reportrate}$ rate as long as the valid touch is detected.

Master Write Interrupt Mode Example:			Master Out	Slave Out
Byte 1	0xC6	Address		
Byte 2	0xA4	Command		
Byte 3	0x01	Interrupt Mode Value		

Master Request Firmware Version

Command Structure

	Byte 1	
S	0xC6 (Address)	A/N
	Byte 2	
	0xA6 (Command)	A/N
P		
	Required 100uS delay	
	Byte 3	
S	Firmware Version Value	A/N
P		

Command Description

This command is used to **Read** the firmware version of the simpleTouch™ IC.

Master Request Firmware Version Example:

			Master Out	Slave Out
Byte 1	0xC6	Address		
Byte 2	0xA6	Command		
Byte 3	0x01	Firmware Version Value		

Master Request Chip ID

Command Structure

	Byte 1		
S	0xC6 (Address)	A/N	
	Byte 2		
	0xA8 (Command)	A/N	
P			
	Required 100uS delay		
	Byte 3		
S	Chip ID Value MSB	A/N	
	Byte 4		
	Chip ID Value LSB	A/N	
P			

Command Description

This command is used to **Read** the identification of the simpleTouch™ IC.

Master Request Chip ID Example:

Master Out	Slave Out

Byte 1	0xC6	Address
Byte 2	0xA8	Command
Byte 3	0x00	Chip ID Value MSB
Byte 4	0x03	Chip ID Value LSB

Master Write simpleTouch™ Config

Command Structure

Byte			
Byte 1	S	0xC6 (Address)	A/N
Byte 2		0xD1 (Command)	A/N
Byte 3		Screen Type	A/N
Byte 4		Number of Columns	A/N
Byte 5		Number of Rows	A/N
Byte 6		X Pixel Range MSB	A/N
Byte 7		X Pixel Range LSB	A/N
Byte 8		Y Pixel Range MSB	A/N
Byte 9		Y Pixel Range LSB	A/N
Byte 10		Substrate Material	A/N
Byte 11		simpleTouch™ Control Register	A/N
Byte 12		EEPROM Initialize Status	A/N
	P		

Master Write simpleTouch™ Config Example:

	Master Out		Slave Out
Byte 1	0xC6	Address	
Byte 2	0xD1	Command	
Byte 3	0x00	Screen Type	
Byte 4	0x06	Number of Columns	
Byte 5	0x06	Number of Rows	
Byte 6	0x01	X Pixel Range MSB	
Byte 7	0x86	X Pixel Range LSB	
Byte 8	0x01	Y Pixel Range MSB	
Byte 9	0x86	Y Pixel Range LSB	
Byte 10	0x01	Substrate Material	
Byte 11	0x00	simpleTouch™ Control Reg	
Byte 12	0x00	EEPROM Initialize Status	

Command Description

This command is used to **Write** a new simpleTouch™ configuration to the IC which automatically re-initializes the simpleTouch™ IC to the new configuration.

Screen Type:
0x00 = e_Self (default)
0x01 = e_Mutual (not supported at this time)

Number of Columns:
Supported values = 3 - 9 when used as XY electrodes, 3-12 when used as slider (default = 5)

Number of Rows:
Supported values = 3 - 9 when used as XY electrodes, value must be 0 when used as slider (default = 7)

X Pixel Range:
Supported Number of Pixels over the Number of Columns when used as XY touchscreen, maximum position when used as slider (default = 240)

Y Pixel Range:
Supported Number of Pixels over the Number of Rows when used as XY touchscreen, maximum position when used as slider (default = 320)

Substrate Material:
Enter the value which corresponds to the user interface substrate material in the end product. (default = 0)
0x00 = Glass—0.5mm to 2.0mm glass (or plastic equivalent)
0x01 = Plastic—1.0mm to 2.0mm plastic (or glass equivalent)
0x02 = Thicker or Low Relative Permittivity Materials

simpleTouch™ Control Register:
Value = Do Not Care. The value of the simpleTouch™ Control Register is subject to change and is controlled by the simpleTouch™ IC.

EEPROM Initialize Status:
Value = Do Not Care (this value is ignored by the simpleTouch™ IC)

Master Request simpleTouch™ Config

Command Structure

	Byte 1		
S		0xC6 (Address)	A/N
	Byte 2		
		0xD2 (Command)	A/N
P			
		Required 100uS delay	
	Byte 3		
S		Screen Type	A/N
	Byte 4		
		Number of Columns	A/N
	Byte 5		
		Number of Rows	A/N
	Byte 6		
		X Pixel Range MSB	A/N
	Byte 7		
		X Pixel Range LSB	A/N
	Byte 8		
		Y Pixel Range MSB	A/N
	Byte 9		
		Y Pixel Range LSB	A/N
	Byte 10		
		Substrate Material	A/N
	Byte 11		
		simpleTouch™ Control Register	A/N
	Byte 12		
		EEPROM Initialize Status	A/N
P			

Command Description

This command is used to **Read** the simpleTouch™ IC configuration settings.

simpleTouch™ Control Register:

Value = Do Not Care. The value of the simpleTouch™ Control Register is subject to change and is controlled by the simpleTouch™ IC.

EEPROM Initialize Status:

0x00 = Not Initialized (default)

0xAA = Initialized

Master Request simpleTouch™ Config Example:

Master Out | Slave Out

Byte 1	0xC6	Address
Byte 2	0xD2	Command
Byte 3	0x00	Screen Type
Byte 4	0x06	Number of Columns
Byte 5	0x06	Number of Rows
Byte 6	0x01	X Pixel Range MSB
Byte 7	0x86	X Pixel Range LSB
Byte 8	0x01	Y Pixel Range MSB
Byte 9	0x86	Y Pixel Range LSB
Byte 10	0x00	Substrate Material
Byte 11	0x00	simpleTouch™ Control Reg
Byte 12	0x00	EEPROM Init Status

Master Write simpleTouch™ Sensor Order Pointers

Command Structure

	Byte		
S	Byte 1	0xC6 (Address)	A/N
	Byte 2	0xD3 (Command)	A/N
	Byte 3	SENSOR 1 Pointer	A/N
	Byte 4	SENSOR 2 Pointer	A/N
	Byte 5	SENSOR 3 Pointer	A/N
	Byte 6	SENSOR 4 Pointer	A/N
	Byte 7	SENSOR 5 Pointer	A/N
	Byte 8	SENSOR 6 Pointer	A/N
	Byte 9	SENSOR 7 Pointer	A/N
	Byte 10	SENSOR 8 Pointer	A/N
	Byte 11	SENSOR 9 Pointer	A/N
	Byte 12	SENSOR 10 Pointer	A/N
	Byte 13	SENSOR 11 Pointer	A/N
	Byte 14	SENSOR 12 Pointer	A/N
P			

Command Description

This command is used to **Write** the simpleTouch™ IC sensor order pointers. The pointer values represents the Column & Row order of the connected sensors.

simpleTouch™ Order Pointers:

The number entered is equivalent to the Sensor Input number/name of the IC pin that the electrode is connected to (see Application Information—Block Diagrams for examples).

Default Pointers:

Byte 3 = 0x08	Byte 4 = 0x09	Byte 5 = 0x0A	Byte 6 = 0x0B
Byte 7 = 0x0C	Byte 8 = 0x04	Byte 9 = 0x05	Byte 10 = 0x06
Byte 11 = 0x07	Byte 12 = 0x03	Byte 13 = 0x02	Byte 14 = 0x01

Note:

- Columns (X) must be entered in first with the Rows (Y) following.
- Inputs must be used in order with no skipped inputs allowed in between. Any unused inputs at the end must be coded with 0.
- Example: if COL = 3 and Row = 6, then SENSOR 1-3 are column numbers, SENSOR 4-9 are used for row numbers and SENSOR 10-12 are 0.

Master Write simpleTouch™ S O Pointers Example: Master Out | Slave Out

Byte 1	0xC6	Address
Byte 2	0xD3	Command
Byte 3	0x04	Sensor 1 Pointer
Byte 4	0x05	Sensor 2 Pointer
Byte 5	0x06	Sensor 3 Pointer
Byte 6	0x07	Sensor 4 Pointer
Byte 7	0x08	Sensor 5 Pointer
Byte 8	0x09	Sensor 6 Pointer
Byte 9	0x03	Sensor 7 Pointer
Byte 10	0x02	Sensor 8 Pointer
Byte 11	0x01	Sensor 9 Pointer
Byte 12	0x00	Sensor 10 Pointer
Byte 13	0x00	Sensor 11 Pointer
Byte 14	0x00	Sensor 12 Pointer

Master Request simpleTouch™ Sensor Order Pointers

Command Structure

S	Byte 1	0xC6 (Address)	A/N
	Byte 2	0xD4 (Command)	A/N
P			
		Required 100uS delay	
S	Byte 3	SENSOR 1 Pointer	A/N
	Byte 4	SENSOR 2 Pointer	A/N
	Byte 5	SENSOR 3 Pointer	A/N
	Byte 6	SENSOR 4 Pointer	A/N
	Byte 7	SENSOR 5 Pointer	A/N
	Byte 8	SENSOR 6 Pointer	A/N
	Byte 9	SENSOR 7 Pointer	A/N
	Byte 10	SENSOR 8 Pointer	A/N
	Byte 11	SENSOR 9 Pointer	A/N
	Byte 12	SENSOR 10 Pointer	A/N
	Byte 13	SENSOR 11 Pointer	A/N
	Byte 14	SENSOR 12 Pointer	A/N
P			

Command Description

This command is used to **Read** the simpleTouch™ IC sensor order pointers. The pointer values represents the Column & Row order of the connected sensors.

Master Request simpleTouch™ S O Pointers Example:

Master Out | Slave Out

Byte 1	0xC6	Address
Byte 2	0xD4	Command
Byte 3	0x04	Sensor 1 Pointer
Byte 4	0x05	Sensor 2 Pointer
Byte 5	0x06	Sensor 3 Pointer
Byte 6	0x07	Sensor 4 Pointer
Byte 7	0x08	Sensor 5 Pointer
Byte 8	0x09	Sensor 6 Pointer
Byte 9	0x03	Sensor 7 Pointer
Byte 10	0x02	Sensor 8 Pointer
Byte 11	0x01	Sensor 9 Pointer
Byte 12	0x00	Sensor 10 Pointer
Byte 13	0x00	Sensor 11 Pointer
Byte 14	0x00	Sensor 12 Pointer

Master Request Serialization Parameters

Command Structure

	Byte 1		
S		0xC6 (Address)	A/N
	Byte 2		
		0xD6 (Command)	A/N
P			
		Required 100uS delay	
	Byte 3		
S		Test Pass/Fail Code	A/N
	Byte 4		
		Date Stamp (YYYY) MSB	A/N
	Byte 5		
		Date Stamp (YYYY) LSB	A/N
	Byte 6		
		Date Stamp (MM)	A/N
	Byte 7		
		Date Stamp (DD)	A/N
	Byte 8		
		Time Stamp MSB	A/N
	Byte 9		
		Time Stamp Byte 2	A/N
	Byte 10		
		Time Stamp LSB	A/N
P			

Command Description

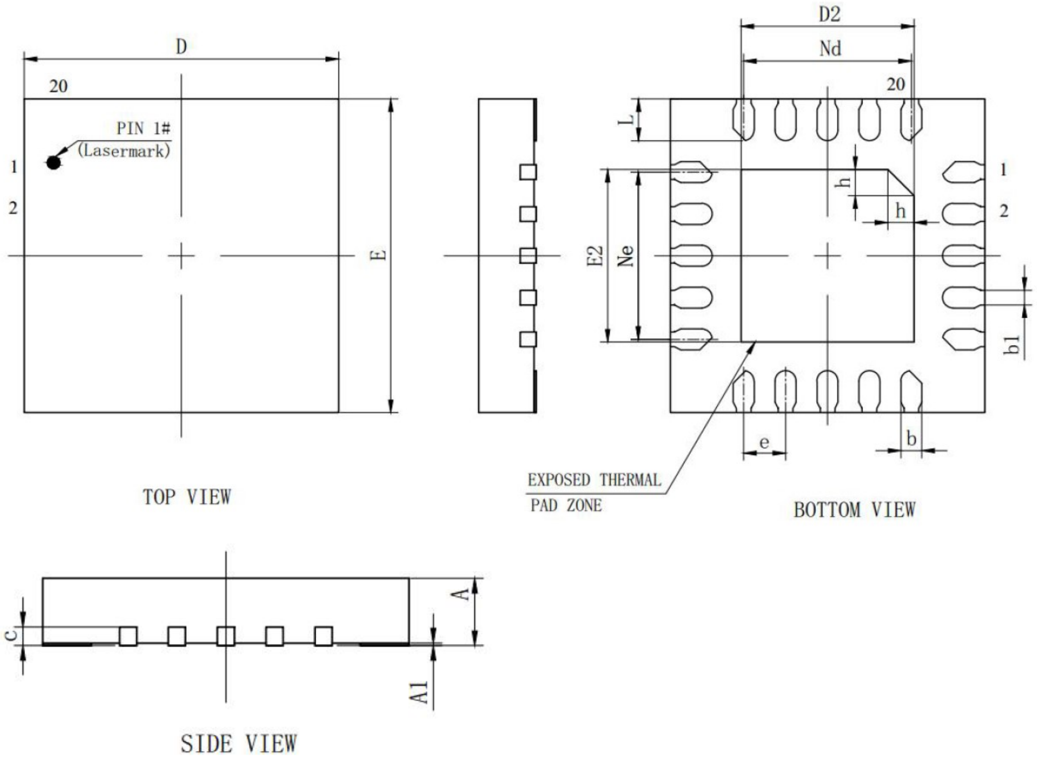
This command is used to **Read** the simpleTouch™ IC Serialization Parameters.

Master Request Serialization Parameters Example:

			🟪 Master Out	🟩 Slave Out
Byte 1	0xC6	Address		
Byte 2	0xD6	Command		
Byte 3	0x01	Test Pass/Fail Code		
Byte 4	0x14	Date Stamp (YYYY) MSB		
Byte 5	0x17	Date Stamp (YYYY) LSB		
Byte 6	0x03	Date Stamp (MM)		
Byte 7	0x31	Date Stamp (DD)		
Byte 8	0x00	Time Stamp MSB		
Byte 9	0xD6	Time Stamp Byte 2		
Byte 10	0x38	Time Stamp LSB		

Package Outline Drawing & Part Ordering Information

20-pin QFN EPAD – 3x3x0.55 mm Body [QFN] with Exposed Pad



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	0. 50	0.55	0. 60
A1	0	0. 02	0. 05
b	0. 15	0. 20	0. 25
b1	0. 14REF		
c	0. 10	0. 15	0. 20
D	2. 90	3. 00	3. 10
D2	1. 55	1. 65	1. 75
e	0. 40BSC		
Ne	1. 60BSC		
Nd	1. 60BSC		
E	2. 90	3. 00	3. 10
E2	1. 55	1. 65	1. 75
L	0. 35	0. 40	0. 45
h	0. 20	0. 25	0. 30
L/F载体尺寸	1. 95x1. 95		

Note: The center pad on the QFN package MUST be connected to system electrical ground (VSS) for operation.

Ordering Information

Part Number: AXS5106S-VST

AiXieSheng Technology co. Ltd

Floor 7, Building 3, Tingwei Industrial Park, No.6, Liufang Road,
Bao'an District, Shenzhen, Gungdong, P.R. China
ZIP: 518058

T +86 755 27677799
F +86 755 27389299
E support@aixiesheng-tech.com

www.aixiesheng-tech.com

TO OUR VALUED CUSTOMERS

We are pleased to provide our valued customers with the most up to date and accurate documentation available to help ensure your success. If you have any questions regarding this documentation or the Venntis simpleTouch™ Technology, please contact us at www.simpleTouch.xyz

Trademarks – Venntis®, the Venntis logo and simpleTouch™ are trademarks of Venntis, LLC.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. VENNTIS MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESSED OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Venntis disclaims all liability arising from this information and its use. Use of Venntis devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Venntis from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Venntis intellectual property rights.