

VNS-15W01

Panel PC

Quick Reference Guide

6th Ed – 21 January 2020

Copyright Notice

Copyright © 2020 ALL RIGHTS RESERVED.

FCC Statement



THIS DEVICE COMPLIES WITH PART 15 FCC RULES. OPERATION IS SUBJECT TO THE FOLLOWING TWO CONDITIONS:

(1) THIS DEVICE MAY NOT CAUSE HARMFUL INTERFERENCE.

(2) THIS DEVICE MUST ACCEPT ANY INTERFERENCE RECEIVED INCLUDING INTERFERENCE THAT MAY CAUSE UNDESIRE OPERATION.

THIS EQUIPMENT HAS BEEN TESTED AND FOUND TO COMPLY WITH THE LIMITS FOR A CLASS "B" DIGITAL DEVICE, PURSUANT TO PART 15 OF THE FCC RULES.

THESE LIMITS ARE DESIGNED TO PROVIDE REASONABLE PROTECTION AGAINST HARMFUL INTERFERENCE WHEN THE EQUIPMENT IS OPERATED IN A COMMERCIAL ENVIRONMENT. THIS EQUIPMENT GENERATES, USES, AND CAN RADIATE RADIO FREQUENCY ENERGY AND, IF NOT INSTALLED AND USED IN ACCORDANCE WITH THE INSTRUCTION MANUAL, MAY CAUSE HARMFUL INTERFERENCE TO RADIO COMMUNICATIONS.

OPERATION OF THIS EQUIPMENT IN A RESIDENTIAL AREA IS LIKELY TO CAUSE HARMFUL INTERFERENCE IN WHICH CASE THE USER WILL BE REQUIRED TO CORRECT THE INTERFERENCE AT HIS OWN EXPENSE.

A Message to the Customer

Customer Services

Each and every product is built to the most exacting specifications to ensure reliable performance in the harsh and demanding conditions typical of industrial environments. Whether your new device is destined for the laboratory or the factory floor, you can be assured that your product will provide the reliability and ease of operation.

Your satisfaction is our primary concern. Here is a guide to our customer services. To ensure you get the full benefit of our services, please follow the instructions below carefully.

Technical Support

We want you to get the maximum performance from your products. So if you run into technical difficulties, we are here to help. For the most frequently asked questions, you can easily find answers in your product documentation. These answers are normally a lot more detailed than the ones we can give over the phone. So please consult the user's manual first.

Content

1.	Getting Started	5
1.1	Safety Precautions	5
1.2	Packing List.....	5
1.3	System Specifications	6
1.4	System Overview	9
1.4.1	Front View.....	9
1.4.2	Rear View	10
1.5	System Dimensions	11
2.	Hardware Configuration	12
2.1	VNS-15W01 connector mapping.....	13
2.1.1	Serial port connector (COM).....	13
3.	Handset Feature	14
3.1	VNS-15W01 handset function	15
3.1.1	The voice come out from speaker, When the handset is on the cradle.	15
3.1.2	The voice come out from the handset, when it is lifted up.	15
4.	LED Instruction	16
4.1	VNS-15W01 LED Programmer's Reference manual.....	17
4.1.1	Android Studio	17
4.1.2	Sample code.....	17
4.2	For win 10 LED instruction	18
4.2.1	GENERICEC Library API Programmer's Reference	18
4.2.1.1	GENERICEC_Open	18
4.2.1.2	GENERICEC_Close.....	18
4.2.1.3	GENERICEC_Close.....	18
4.2.1.4	GENERICEC_Write.....	19
4.2.2	Standard Operating Procedures.....	20
4.2.2.1	Apps	20
4.2.2.2	Bin	21
4.2.2.3	Driver.....	21
4.2.2.4	Sample	21
4.2.2.5	Demo.....	22
4.3	For Ubuntu 19.04 LED instruction	24
4.3.1	File list.....	24
4.3.2	Building	24

VNS-15W01

4.3.3	EC, I2C Definition on VNS device	24
4.3.4	Operation (vns_led)	24
4.3.5	APP & API Introduction (vns_led.c, i2c-dev.h, i2cbusses.h)	24
4.3.5.1	Open I2C driver	24
4.3.5.2	Set I2C device/slave address	25
4.3.5.3	Write I2C command & data	25
5.	Front Touch Button /Handset Function Instruction	27
5.1	Handset Function / Front Touch Button Instruction	28
1.	Handset function disable as default.....	28
2.	Front touch button function as Enable.....	29

1. Getting Started

1.1 Safety Precautions

Warning!



Always completely disconnect the power cord from your chassis whenever you work with the hardware. Do not make connections while the power is on. Sensitive electronic components can be damaged by sudden power surges. Only experienced electronics personnel should open the PC chassis.

Caution!



Always ground yourself to remove any static charge before touching the CPU card. Modern electronic devices are very sensitive to static electric charges. As a safety precaution, use a grounding wrist strap at all times. Place all electronic components in a static-dissipative surface or static-shielded bag when they are not in the chassis.

Risk of Explosion if Battery is replaced by an Incorrect Type. Dispose of Used Batteries According to the Instructions.

Français:

Attention!



Débranchez le câble d'alimentation de votre châssis chaque fois que vous travaillez avec le matériel. Ne faites pas de connexion lorsque le système est allumé. Les composants électroniques sensibles peuvent être endommagés par les surtensions soudaines. Seule les personnels expérimentés de l'électronique peuvent ouvrir le châssis du PC.

Précaution!



Il faut toujours mettre à la masse pour éliminer l'électricité statique avant de toucher la carte CPU. Les appareils électroniques modernes sont très sensibles aux électricité statique. Pour des raisons de sécurité, utilisez un bracelet électrostatique. Placez tous les composants électroniques sur une surface antistatique ou dans un sac antistatique quand ils ne sont pas dans le châssis.

Risque d'explosion si la batterie est remplacée par un type incorrect. Jetez les piles usagées selon les instructions

1.2 Packing List

- 1 x VNS-15W01
- 1 x AC/DC adapter 12V/5A 90 Screw Type (Option)
- 1 x power cord (Option)



If any of the above items is damaged or missing, contact your retailer.

1.3 System Specifications

Component	
CPU	Intel Atom Z8350
CPU Cooler (Type)	Heatsink
Memory	2GB / 4GB DDR3L
Power Supply	DC Input, 12~24V / Powered LAN 802.3AT
Microphone	1 x A-MIC
Speaker	2 x 4Ω 2.0W/2.5W(MAX) speaker(L & R)
Camera	2.0MP Camera
Wireless LAN	WIFI 802.11 b/g/n
Operating System	Windows10 IoT 2016 64bit / Android 5.1 64bit
Storage	
Other Storage Device	32GB/64GB eMMC
External I/O	
Serial Port	1 x RJ45 for RS232/422/485(configurable by BIOS setting)
USB Port	2 x USB 2.0 Type A
LAN Port	1 x 10/100/1000 Ethernet
Indicator Light	LED indicating bar
Others	1 x Headphone- 1 x 3.5mm Audio Jack (4 rings)
Mechanical	
Power Type	DC-IN(12~24V)/ Powered LAN 802.3T)
Power Connector Type	DC-IN : DC Jack 5.7mm Powered LAN+ : RJ45
Dimension	472.1 x 269 x 70.1 mm
Weight	2.5kg
Color	white
Option	1 x Handset
Software	
BIOS	64 bit
Reliability	
EMI Test	CE/ FCC class B
Dust and Rain Test	Front Panel IP65
Vibration Test	Sine Vibration test (Non-operation) Reference IEC60068-2-6 Testing procedures Test Fc : Vibration sinusoidal 1 Test Acceleration : 2G 2 Test frequency : 5~500 Hz

	<p>3 Sweep : 1 Oct/ per one minute. (logarithmic)</p> <p>4 Test Axis : X,Y and Z axis</p> <p>5 Test time :30 min. each axis</p> <p>6 System condition : Non-Operating mode</p> <p><u>Package Vibration Test</u></p> <p>Reference IEC60068-2-64 Testing procedures</p> <p>Test Fh : Vibration boardband random Test</p> <p>1. PSD: 0.026G²/Hz , 2.16 Grms</p> <p>2. Non-operation mode</p> <p>3. Test Frequency : 5-500Hz</p> <p>4. Test Axis : X,Y and Z axis</p> <p>5. 30 min. per each axis</p> <p><u>Random Vibration Operation</u></p> <p>Reference IEC60068-2-64 Testing procedures</p> <p>Test Fh : Vibration boardband random Test</p> <p>1. PSD: 0.00454G²/Hz , 1.5 Grms</p> <p>2. Operation mode</p> <p>3. Test Frequency : 5-500Hz</p> <p>4. Test Axis : X,Y and Z axis</p> <p>5. 30 minutes per each axis</p> <p>6. IEC 60068-2-64 Test:Fh</p>
Mechanical Shock Test	<p><u>Bump Test</u></p> <p>Reference IEC 60068-2-29 Testing procedures</p> <p>Test Eb : Bump Test</p> <p>1. Wave form : Half Sine wave</p> <p>2. Acceleration Rate : 10g for operation mode</p> <p>3. Duration Time : 11ms</p> <p>4. No. of Shock : Z axis 300 times</p> <p>5. Test Axis: Z axis</p> <p>6. Operation mode</p>
Drop Test	<p>1. One corner , three edges, six faces</p> <p>2. ISTA 2A, IEC-60068-2-32 Test:Ed</p>
Operating Temperature	0°C ~ 40°C
Operating Humidity	0% ~ 90% Relative Humidity, Non-condensing
Storage Temperature	-20°C ~ 60°C

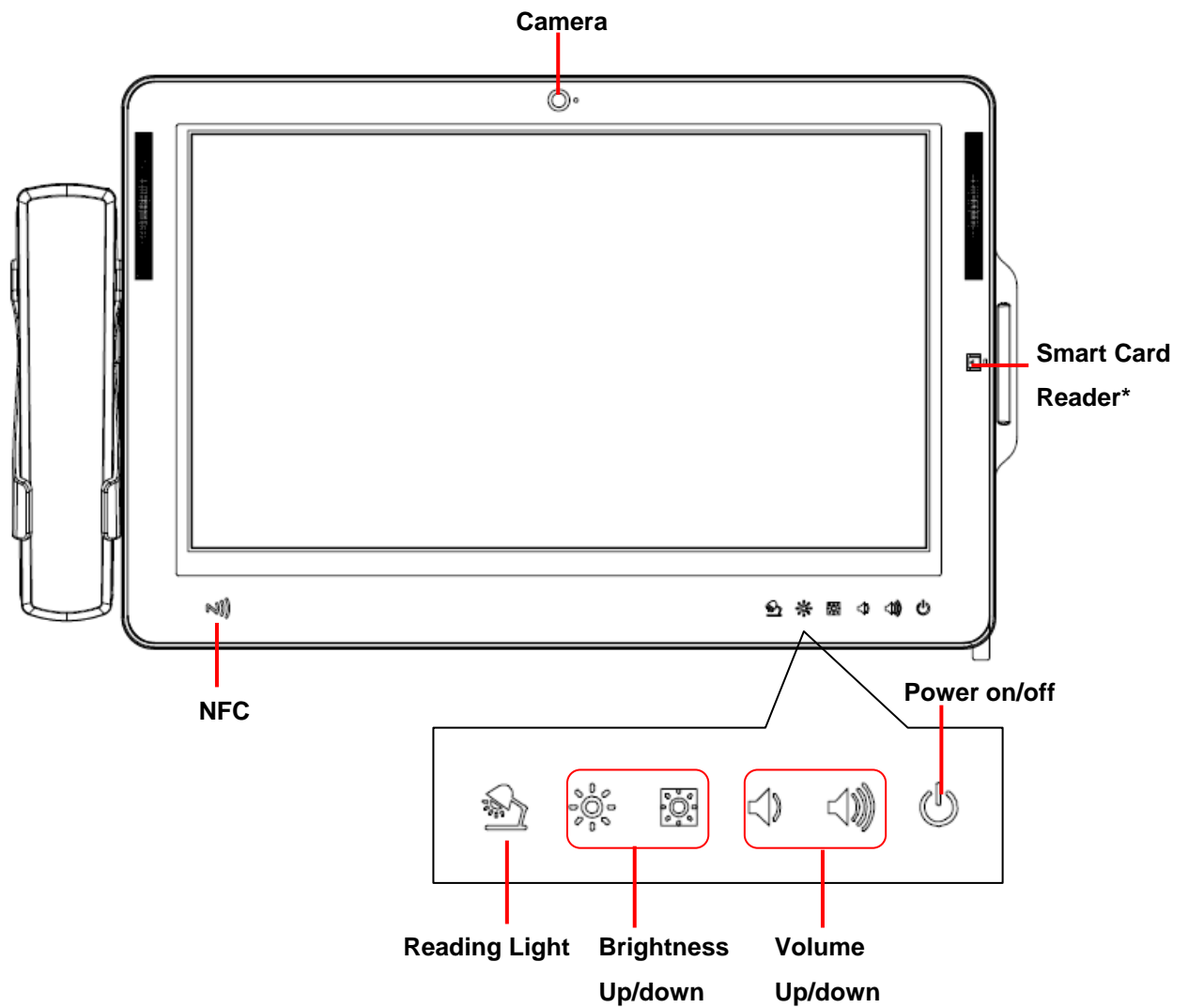
VNS-15W01



Note: Specifications are subject to change without notice.

1.4 System Overview

1.4.1 Front View

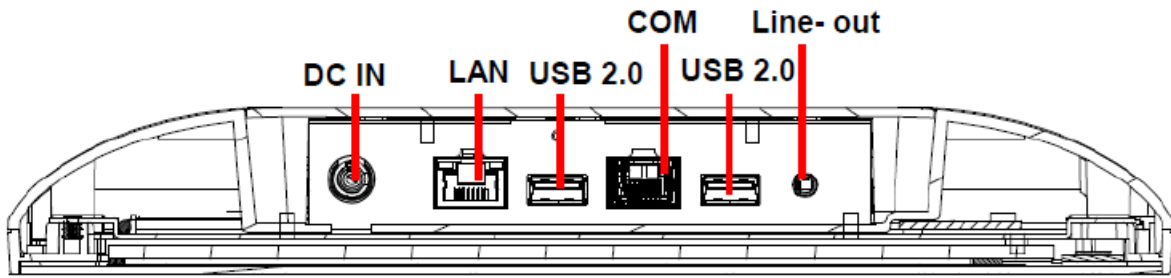


***Note:**

Insert the IC card (face down) into the Smart Card Reader slot.

VNS-15W01

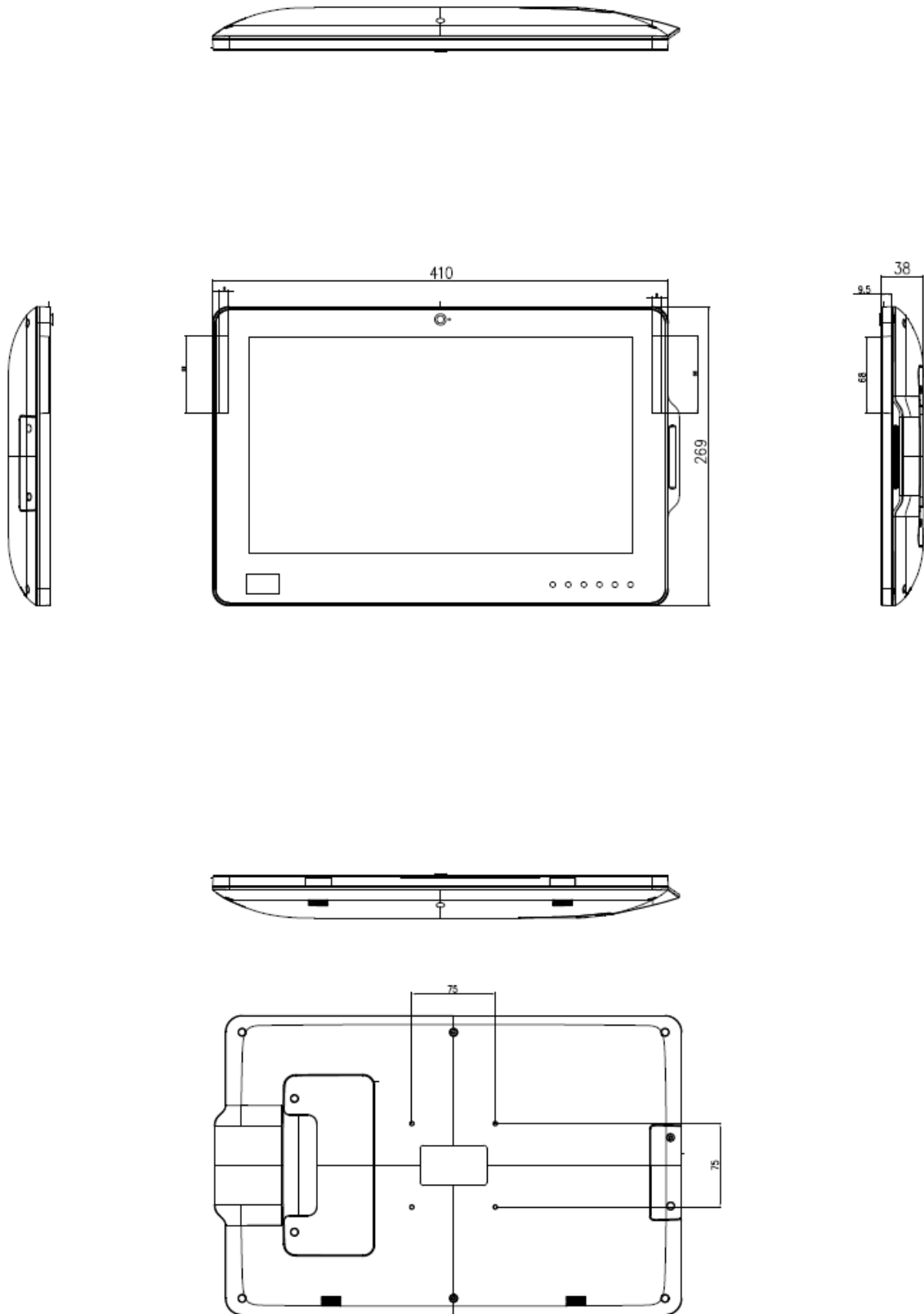
1.4.2 Rear View



Connectors

Label	Function	Note
DC IN	DC-IN power connector	
LAN	RJ-45 Ethernet	
COM	Serial port connector	
USB	2 x USB2.0 connector	
Line-out	Line-out audio jack	

1.5 System Dimensions

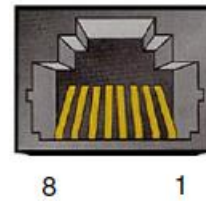
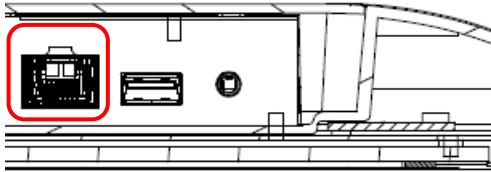


(Unit: mm)

2. Hardware Configuration

2.1 VNS-15W01 connector mapping

2.1.1 Serial port connector (COM)



RS-485

Signal	PIN
DATA-	1
DATA+	2
NC	3
NC	4
GND	5
NC	6
NC	7
NC	8

RS-232

Signal	PIN
DCD	1
RXD	2
TXD	3
DTR	4
GND	5
DSR	6
RTS	7
CTS	8

RS-422

Signal	PIN
TX-	1
TX+	2
RX+	3
RX-	4
GND	5
NC	6
NC	7
NC	8

3. Handset Feature

3.1 VNS-15W01 handset function

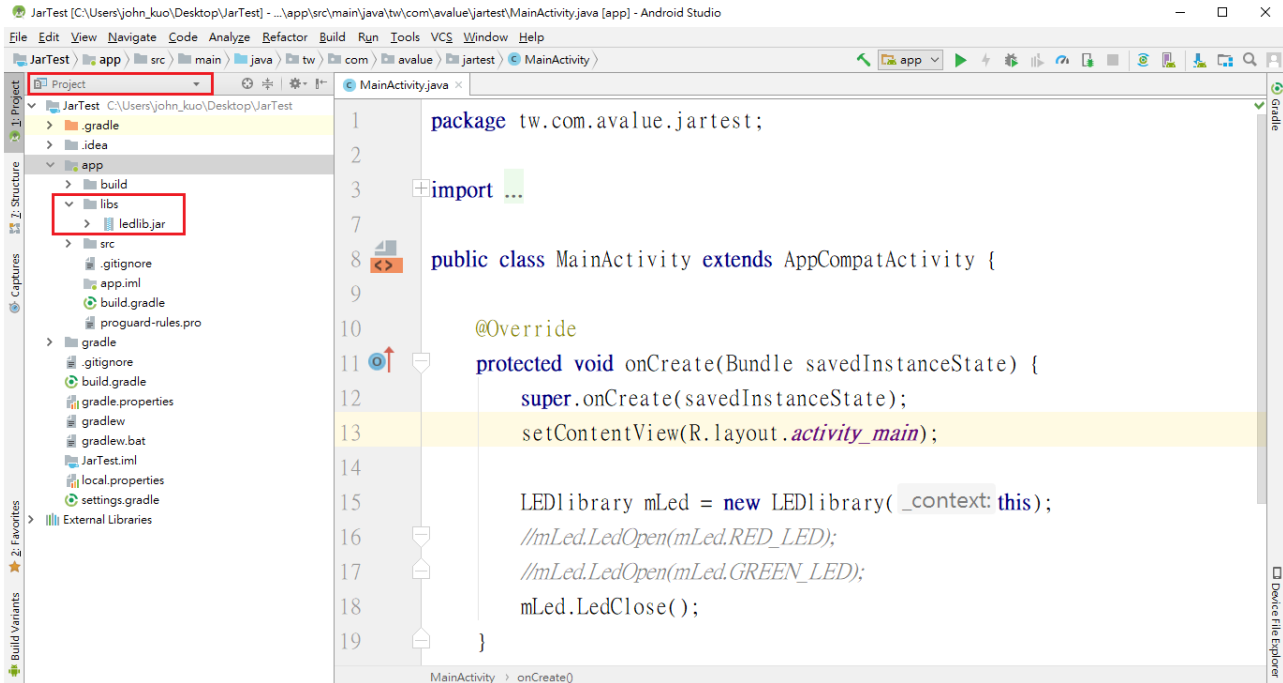
- 3.1.1 The voice come out from speaker, When the handset is on the cradle.**
- 3.1.2 The voice come out from the handset, when it is lifted up.**

4. LED Instruction

4.1 VNS-15W01 LED Programmer's Reference manual

4.1.1 Android Studio

- Put the ledlib.jar into the app "libs" folder
- Set the jar file as library



4.1.2 Sample code

- import this library in java file
- You can use the sample code to run any super user command as follows.

```
import tw.com.avalue.ledlibrary.LEDlibrary;
```

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
```

```
    LEDlibrary mLed = new LEDlibrary(this);
    //mLed.LedOpen(mLed.RED_LED);
    //mLed.LedOpen(mLed.GREEN_LED);
    mLed.LedClose();
```

```
}
```

4.2 For win 10 LED instruction

4.2.1 GENERICEC Library API Programmer's Reference

After install the GENERICEC driver on the board, the API DLL file also install into Windows system directory. The functions of API provide are list as below, refer to each function's description to get more detail information.

4.2.1.1 GENERICEC_Open

bool GENERICEC_Open()

Return Value:

True on success and **False** on failure.

Remarks:

The function will loading or initial the GENERICEC driver, for use by the application. If the return value is **False**, it indicates the driver is not installing yet.

4.2.1.2 GENERICEC_Close

bool GENERICEC_Close()

Return Value:

True on success and **False** on failure.

Remarks:

This function will close and unload the GENERICEC driver, it must be called before the application terminates. If the return value is **False**, it indicates the driver is not installing successfully.

4.2.1.3 GENERICEC_Read

bool GENERICEC_Read(unsigned char *wbuf, int wlen, unsigned char *rbuf, int rlen)

Return Value:

True on success and **False** on failure.

Parameters:

wbuf Write buffer.
wlen Write buffer length.
rbuf Read buffer.
rlen Read buffer length.

Remarks:

Write the **wbuf** data and read the result from the **rbuf**. If the return value is **False**, it indicates the driver is not installing successfully or not uses the **GENERICEC_Open** function first.

4.2.1.4 GENERICEC_Write

bool GENERICEC_Write(unsigned char *wbuf, int wlen)

Return Value:

True on success and **False** on failure.

Parameters:

wbuf Write buffer.

wlen Write buffer length.

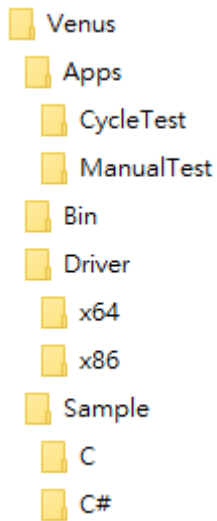
Remarks:

Write the **wbuf** data. If the return value is **False**, it indicates the driver is not installing successfully or not uses the **GENERICEC_Open** function first.

VNS-15W01

4.2.2 Standard Operating Procedures

After have extracted Venus.zip, it can see folder structure as follow.

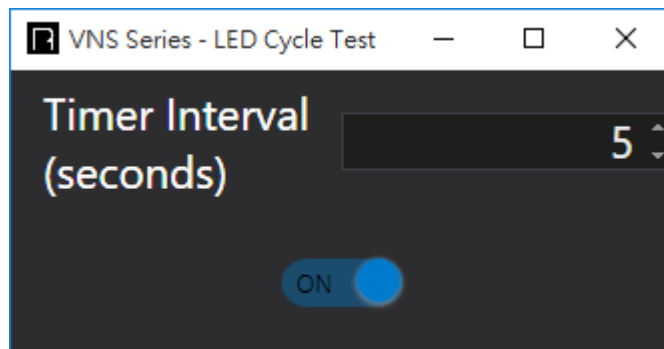


4.2.2.1 Apps

a. CycleTest

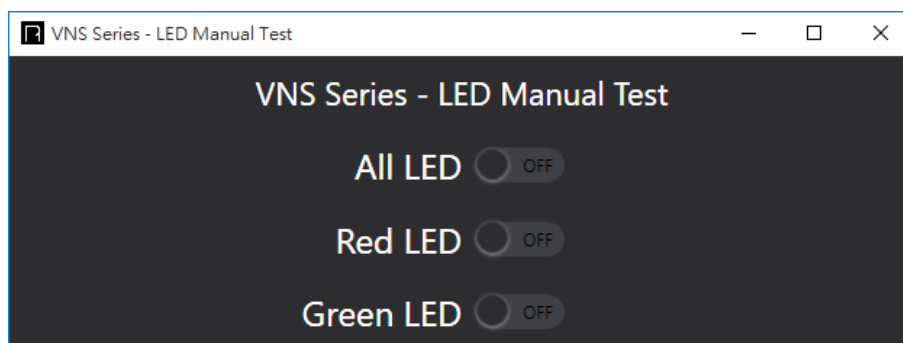
Cycle Test LED ON/OFF, auto minimizes the app after executing the program.

Timer Interval: set LED ON/OFF interval seconds.



b. ManualTest

Manual test LED ON/OFF, move the app to the center of screen after executing the program.



4.2.2.2 Bin

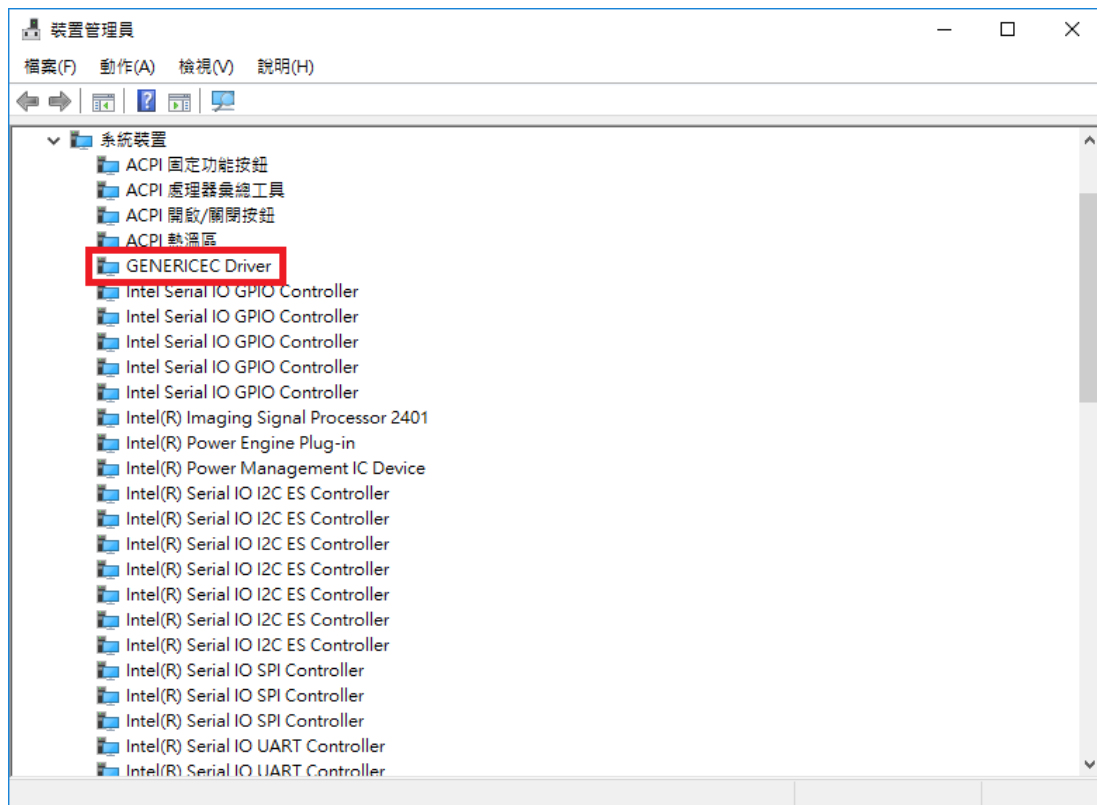
GENERICEC.dll, Dynamic-Link Library x86 Architecture.

Communicate with Generic EC Driver via this library.

4.2.2.3 Driver

GENERICEC Driver, x86/x64 Architecture.

This is Venus Series EC Driver. Please make sure target device has installed it.



4.2.2.4 Sample

C or C# Language sample code.

Reference this sample code to know how to control LED via GENERICEC.dll.

P.S. The project use Microsoft Visual Studio 2017 to develop sample code. If don't have that, it can download community version to reference sample code.

(<https://imagine.microsoft.com/en-us/Catalog/Product/530>)

CAUTION. Because only provide x86 Dynamic-Link Library, so please create x86 application.

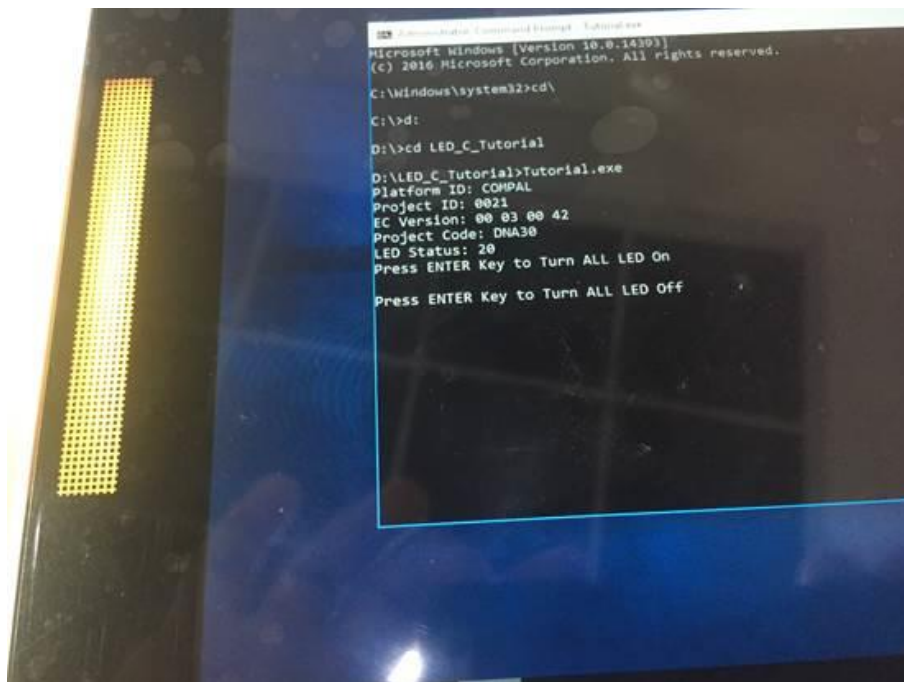
CAUTION. All the test app and sample are only execute on the Venus Series which is installing the GENERICEC Driver.

4.2.2.5 Demo

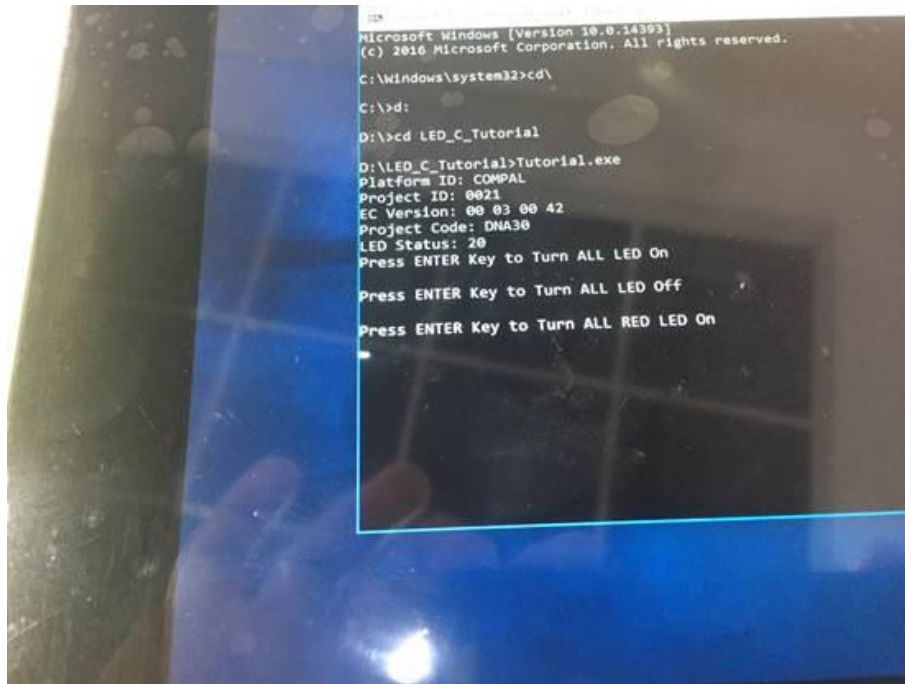
a. Run the Sample\C\Tutorial\Debug\Tutorial.exe and it will show as below picture.



b. Press enter key to turn all LED on.



c. Press enter key to turn all LED off.



4.3 For Ubuntu 19.04 LED instruction

4.3.1 File list

No	Filename	Description
1	VNS LED Control Quick Guide_v1.0.doc	Guideline document
2	VNS_LED-20181121.zip	Demo app source code
3	vns_led	Demo app execution file

4.3.2 Building

1. Prepare a VNS device with Ubuntu(16.04above) and install compiler package(Ex: gcc, make, c-library).
2. Make a working folder in Ubuntu Ex: app.
3. Unzip VNS_LED.zip to working folder.
4. In working folder, enter "\$ make" to compiler and it would create execute file "vns_led".
5. Copy "vns_led" to VNS device and set to executable file.
6. Start to test/execute in root right.

4.3.3 EC, I2C Definition on VNS device

Bus Number (on VNS): 0x02

EC I2C Device Address (on VNS): 0x70

EC I2C Read Command: 0x80

EC I2C Write Command: 0x81

LED I2C Device Address (on EC): 0x01

LED Data Address (on LED Device): 0xbc

4.3.4 Operation (vns_led)

Syntax: vns_led <led_id>

Option:

led_id: 0=off, 1=red, 2=green

Sample:

```
vns_led 1 // light on the RED LED
vns_led 2 // light on the GREEN LED
vns_led 0 // turn off LED
```

4.3.5 APP & API Introduction (vns_led.c, i2c-dev.h, i2cbusses.h)

4.3.5.1 Open I2C driver

```
int open_i2c_dev(int i2cbus, char* filename, size_t size, int quiet);
```

Open i2c driver, and return file handle.

Parameter:

int i2cbus: i2c bus number, should be 2 for EC device.
char* filename: return i2c device name, ex: "/dev/i2c-2".
size_t size: size of filename.
int quiet: quiet mode for debug message, should be 0.

Example:

```
int fd;
char filename[20];
fd = open_i2c_dev(i2cbus, filename, sizeof(filename), 0);
```

4.3.5.2 Set I2C device/slave address

```
int set_slave_addr(int fd, int address, int force);
```

Set I2C device/slave address.

Parameter:

int fd: i2c driver file handle.
int address: device/slave address, should be 0x70 for EC address.
int force: force writing, should be 1.

Example:

```
int res;
res = set_slave_addr(fd, address, 1);
```

4.3.5.3 Write I2C command & data

```
static inline __s32 i2c_smbus_write_i2c_block_data(int fd, __u8 command,
                                                    __u8 length,
                                                    const __u8 *values);
```

Parameter:

int fd: i2c driver file handle.
__u8 command: data address, should be 0x01 for LED control.
__u8 length: data length.
const __u8 *values: data buffer.

Example:

```
int res, i2cbus=2, address=0x70, daddress=0x01, led=0;
// 0x81=EC I2C Write, 0xbc=LED data address, 0x00=LED ID
unsigned char block[16] = {0x81, 0xbc, 0x00};
block[2] = led;
res = i2c_smbus_write_i2c_block_data(fd, address, 3, block);
```

VNS-15W01

Note:

- Under Android OS, when shut down by power button, the power light is from green to orange, then will be green after few seconds, the display off after shown "INTEL" again, meanwhile the power light is orange, shut down successfully.
- When get into OS under Android system, the LED on the Smart Card Reader Module is still red light, that LED will be green after insert card.
- If you MUST flash OS, from Android to Win10 only is allowed, the license label is NOT our responsibility.

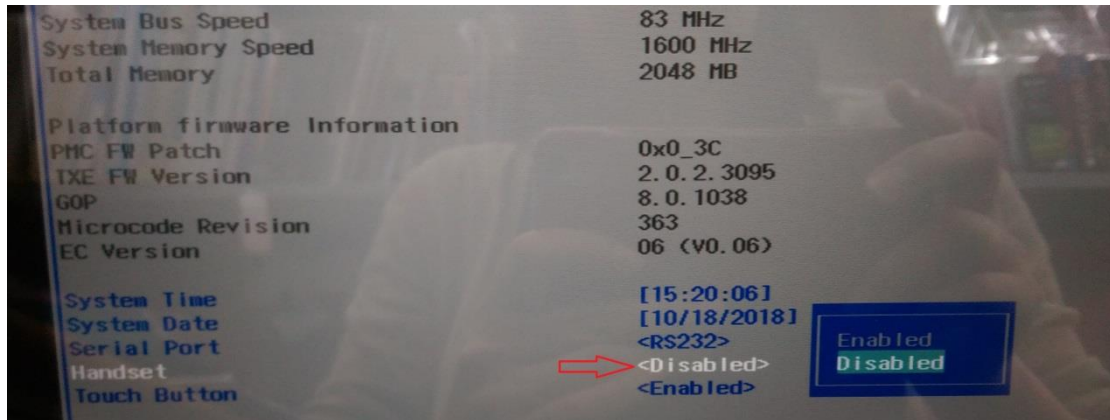
5. Front Touch Button /Handset Function Instruction

5.1 Handset Function / Front Touch Button Instruction

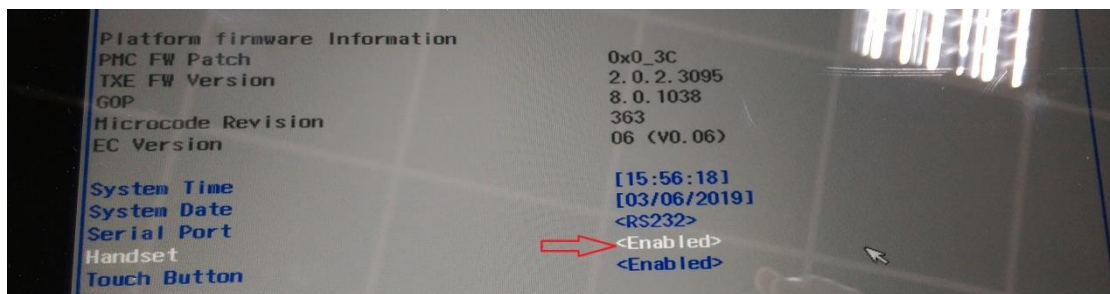
1. Handset function disable as default.

1.1 Press “F2” key into BIOS setup screen.

BIOS path: **Main/Touch Button: Enabled** as default.



1.2. Change Handset item from Disable to “Enabled”.



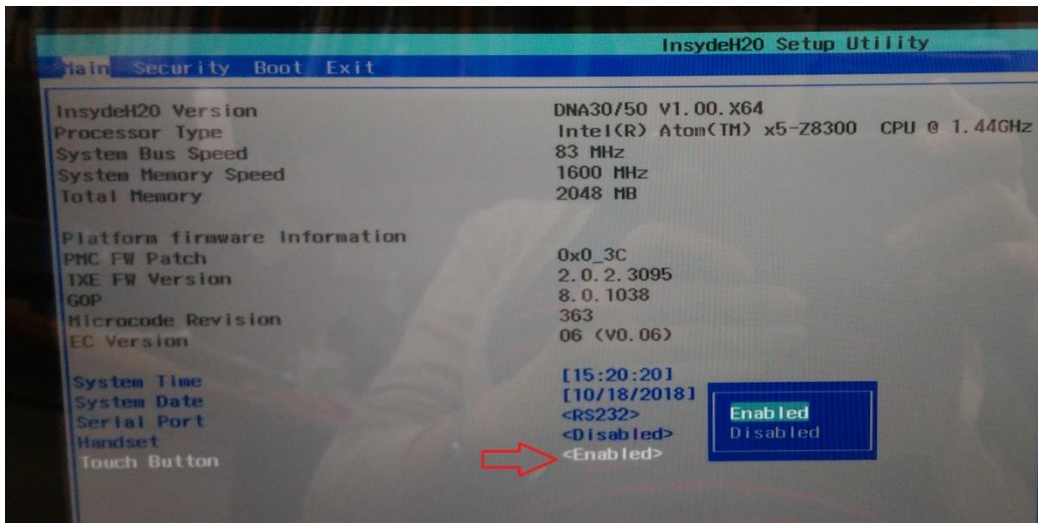
1.3. Press “F10” key to save BIOS setting.



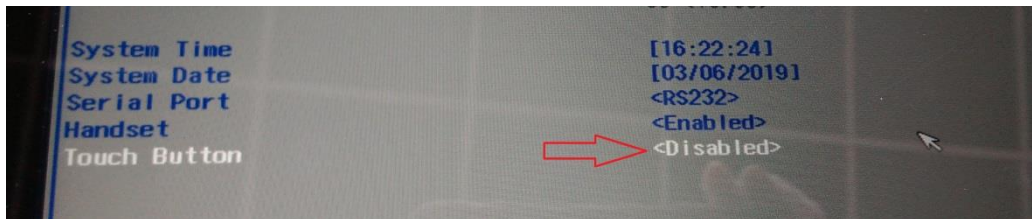
2. Front touch button function as Enable.

2.1 Press “F2” key into BIOS setup screen.

BIOS path: Main/Touch Button: Enabled as default.



2.2. Change Front touch Button item from Enable to “Disabled”.



2.3. Press “F10” key to save BIOS setting.



2.4. Restart VNS system the front touch button function will be turned off.

