API Document

Elo Touch Solutions
Elo® Tablet

ES601068 Rev A

Change History

Rev.	Description of Change	Release Date	Changed By
A	Initial Release per **ECO-13-0697.**	09/05/2013	Cherrie Soetjipto

Disclaimer

The information in this document is subject to change without notice. Elo Touch Solutions, Inc. and its affiliates (collectively "Elo") makes no representations or warranties with respect to the contents herein, and specifically disclaims any implied warranties of merchantability or fitness for a particular purpose. Elo reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of Elo to notify any person of such revisions or changes.

Trademark Acknowledgments

AccuTouch, CarrollTouch, Elo, Elo (logo), Elo Touch, Elo Touch Solutions, Elo TouchSystems, IntelliTouch, iTouch, SecureTouch, TouchTools and VuPoint are trademarks of Elo and its affiliates. Windows is a trademark of Microsoft Corporation.

Table of Contents

Introduction 5

Application Programming Interface 6

Battery Management 6

MCU 9

POS Peripheral 10

Sensors 11

EEPROM 18

Introduction

The following Application Programming Interfaces (APIs) are used to communicate with the Elo Tablet. It will be released as Dynamic Link Library (DLL). The user of the API will need to import the DLL and call the corresponding function listed below.

The APIs are classified into 5 sections: Battery Management, MCU, POS peripherals, Sensors and EEPROM.

For further information on Tablet APIs, please contact an Elo Touch technical representative or customer service.

Visit http://www.elotouch.com/Support/TechnicalSupport/default.asp for technical support.

Refer to the last page of this user manual for Elo contact information worldwide.

Application Programming Interface

Battery Management

1. EloGetBatteryStatus(unsigned char *pStatus);

   Description: Retrieve Battery Information

   OUT: 1 byte of Battery Information

   Bit 7 - main battery present

   Bit 6 - main battery charging

   Bit 5 - main battery discharging

   Bit 4 - main battery fully charged

   Bit 3- docking battery present

   Bit 2 - docking battery charging

   Bit 1 - docking battery discharging

   Bit 0 -docking battery fully charge

   Successful Return value: STATUS_OK (0)

2. EloGetBatteryCapacityPercentage(BATTERY_LOCATION location, unsigned int *pCapacity);

   Description: Retrieve current battery capacity percentage

   IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

   OUT: 0 – 100 percent

   Successful Return value: STATUS_OK (0)

3. EloGetBatteryChargingStrength(BATTERY_LOCATION location, unsigned int *pStrength);

Description: Retrieve current battery charging Strength Percentage

IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

OUT: 0 – 100 percent

Successful Return value: STATUS_OK (0)

4. EloGetBatteryChargingThreshold(BATTERY_LOCATION location, unsigned char *pPerCent);

   Description: Retrieve current Battery charging threshold

   IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

   OUT: 0 – 100 percent

   Successful Return value: STATUS_OK (0)

5. EloGetBatteryLifeTime(BATTERY_LOCATION location, unsigned int *pLifeTime);

   Description: Retrieve current Battery Life Time

   IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

   OUT: 0 – 65535

   Successful Return value: STATUS_OK (0)

6. EloGetBatteryRemainingLifeTime(BATTERY_LOCATION location, unsigned short *pLifeTime);

   Description: Retrieve current Battery Remaining Life Time

   IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

   OUT: 0 – 65535

   Successful Return value: STATUS_OK (0)

7. EloGetBatteryVoltageCurrent(BATTERY_LOCATION location, short *pVoltage, short *pCurrent);

   Description: Retrieve the battery voltage and current

   IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

   OUT: 0 – 65536 for Voltage, -32768 – 32767 for Current

   Successful Return value: STATUS_OK (0)

8. EloSetBatteryChargingStrength(BATTERY_LOCATION location, unsigned int Strength);

   Description: Set the battery charging strength

   IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

   IN: The strength percentage value to be set, 0 - 100

   Successful Return value: STATUS_OK (0)

9. EloSetBatteryChargingThreshold(BATTERY_LOCATION location, unsigned char perCent);

   Description: Set the battery charging threshold

   IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

   IN: The charging threshold percentage value to be set, 0 - 100

   Successful Return value: STATUS_OK (0)

10. EloSetBatteryLifeTime(BATTERY_LOCATION location, unsigned int LifeTime);

    Description: Set the battery charging threshold

    IN: BATTERY_LOCATION: 0 Tablet Battery 1 Docking Battery

    IN: The charging threshold value to be set

    Successful Return value: STATUS_OK (0)

MCU

1. EloGetECVersion(unsigned int *pECVer);

   Description: Get Tablet EC version

   OUT: 2 Bytes, LSB: Version LSB, MSB: Version MSB

   Output format: MSB.LSB

   Successful Return value: STATUS_OK (0)

2. EloGetDECVersion(unsigned int *pECVer);

   Description: Get Docking EC version

   OUT: 2 Bytes, LSB: Version LSB, MSB: Version MSB

   Output format: MSB.LSB

   Successful Return value: STATUS_OK (0)

3. ELODockingMcuReset(unsigned char Data);

   Description: Reset the Docking MCU

   IN: 0 or 1

   Successful Return value: STATUS_OK (0)

4. EloGetDockingStatus(unsigned int *pDock);

   Description: Get the docking connection status

   OUT: 1 present 0 absent

   Successful Return value: STATUS_OK (0)

POS Peripheral

1. EloGetMSRControl(unsigned char *pData);

   Description: Get the MSR output device

   OUT: 0 keyboard 1 Virtual COM port

   Successful Return value: STATUS_OK (0)

2. EloEnableNFCDevice(unsigned char Data);

   Description: Turn ON/OFF the NFC device

   IN: 0 OFF 1 ON

   Successful Return value: STATUS_OK (0)

3. EloEnableSCRDevice(unsigned char Data);

   Description: Turn ON/OFF the NFC device

   IN: 0 OFF 1 ON

   Successful Return value: STATUS_OK (0)

4. EloSetMSRControl(unsigned char Data);

   Description: Set the MSR output

   IN: 0 Keyboard 1 Virtual COM port

   Successful Return value: STATUS_OK (0)

Sensors

1. EloEnableLightSensor(unsigned char Data);

   Description: This API is obsolete

2. EloGetLightSensorData(unsigned int *pLightSensorData);

   Description: Retrieve the light sensor data

   OUT: raw data 0 – 65535

   Successful Return value: STATUS_OK (0)

3. EloEnableAccelerometer(unsigned int enable);

   Description: This API is obsolete

4. EloGetAccelerometerData(unsigned short *pData, unsigned int length);

   Description: Retrieve the Accelerometer Data

   IN: the parameter obsolete

   OUT: WORD 0 of pData X, WORD 1 of pData Y, WORD 2 of pData Z

   Successful Return value: STATUS_OK (0)

5. EloGetAccelerometerSamplingPeriod(unsigned char *pData);

   Description: Retrieve the Accelerometer sampling period

   OUT: Raw data 0 – 255 units, 1 unit is 10ms.

   Successful Return value: STATUS_OK (0)

6. EloSetAccelerometerSamplingPeriod(unsigned char Data);

Description: Set the Accelerometer sampling period

IN: Raw data 0 – 255 units, 1 unit is 10ms

Successful Return value: STATUS_OK (0)

7. EloGetAccelerometerGSetting(unsigned char *pData);

   Description: Retrieve the G setting

   OUT: Bit 0 and Bit 1 of the Byte pData valid

   0: 2G

   1: 4G

   2: 8G

   Successful Return value: STATUS_OK (0)

8. EloSetAccelerometerGSetting(unsigned char Data);

   Description: Set the G setting

   IN: 0: 2G, 1: 4G, and 2: 8G

   Successful Return value: STATUS_OK (0)

9. EloGetOrient(unsigned char *pData);

   Description: Retrieve the Orientation and Freefall value

   OUT: Bit 2 of pData is Freefall value

   1 Freefall detected

   0 No freefall

   Bit 1 and Bit 0 combination of pData for Orientation

   0x00 - portrait up

   0x01 - portrait down

0x02 - landscape right

0x03 - landscape left

Successful Return value: STATUS_OK (0)

10. EloOnBacklight(unsigned int on);

    Description: This API is obsolete

11. EloGetBacklight(unsigned int *pPercentage);

    Description: This API is obsolete

12. EloSetBacklight(unsigned int percentage);

    Description: This API is obsolete

13. EloGetPowerBtnStat(unsigned char *pData);

    Description: Retrieve the power button click status

    OUT: Bit 0, Bit 1 and Bit 2 are mutual exclusively set

    Bit 0 - single click

    Bit 1 - flash light toggle

    Bit 2 - double click

    Successful Return value: STATUS_OK (0)

14. EloGetLEDStatus(ELO_LEDS led, unsigned int *pData);

    Description: Retrieve the LED data

    IN: led index

0: Power

2: Payment Amber

4: WiFi

6: Charge Amber

8: Payment Green

10: Charge Green

12: Bluetooth

14: LightBeam

16: Docking Power

18: Docking Charge Green

20: Docking Charge Amber

22: LAN

OUT: Bit details of the pData are as following:

bit 0 ~ bit 3

0 - Off

1 -blink in 200ms

2 -blin k in 500ms

3 -blink in 1s

4 -blink in 2s

5 - blink in 4s

6 - blink in 8s

7 - full on

8 - undefined

9 - fade away in step of 100ms

10 - fade away in step of 200ms

11 - fade away in step of 300ms

12 - fade away in step of 400ms

13 ~ 15 - undefined

bit 4 ~ bit 6 – brightness

0 - off

1 - 7 different level, 7 is the brightest

Successful Return value: STATUS_OK (0)

15. EloSetLEDStatus(ELO_LEDS led, unsigned int Data);

Description: Set the LEDs

IN: led index

0: Power

2: Payment Amber

4: WiFi

6: Charge Amber

8: Payment Green

10: Charge Green

12: Bluetooth

14: LightBeam

16: Docking Power

18: Docking Charge Green

20: Docking Charge Amber

22: LAN

OUT: Bit details of the byte are as following:

bit 0 ~ bit 3

0 – Off

1 -blink in 200ms

2 -blink in 500ms

3 -blink in 1s

4 -blink in 2s

5 - blink in 4s

6 - blink in 8s

7 - full on

8 – undefined

9 - fade away in step of 100ms

10 - fade away in step of 200ms

11 - fade away in step of 300ms

12 - fade away in step of 400ms

13 ~ 15 - undefined

bit 4 ~ bit 6 - brightness

0 - off

1 - 7 different level, 7 is the brightest

Successful Return value: STATUS_OK (0)

16. EloEnableCameraFlash(unsigned int enable);

    Description: This API is obsolete.

17. EloGetCameraFlashStrength(unsigned int *pStrength);

Description: This API is obsolete.

18. EloSetCameraFlashStrength(unsigned int strength);

    Description: This API is obsolete.

19. EloGetTemperature(unsigned int nSensor, char *pTemp);

    Description: Retrieve the sensor temperatures

    IN: 0 – 6

    0: CPU core

    1: Battery Pack

    2: Mainboard Top

    3: Mainboard bottom

    4: System Memory

    5: Docking

    6: Docking battery

    OUT: -20 to 127

    Successful Return value: STATUS_OK (0)

EEPROM

1. EloWriteEpr(unsigned int Data);

   Description: Write 1 byte to 1 location of 1 specific segment

   IN: Byte0 of Data is segment

      Byte1 of Data is offset

      Byte2 of Data is byte of data

   Successful Return value: STATUS_OK (0)

2. EloPreReadEpr(unsigned short Data);

   Description: Tell EC the detailed EEPROM location to read data out. This API has to be called together with the following API EloReadEpr.

   IN: Byte0 of Data is segment

      Byte1 of Data is offset

   Successful Return value: STATUS_OK (0)

3. EloReadEpr(unsigned short *pData);

   Description: Read 1 byte of data out from EEPROM. This API has to be called together with the above API EloPreReadEpr().

   OUT: pData will save the data read out from EEPROM

   Successful Return value: STATUS_OK (0)

4. EloGetMotherBoardSerialNumber_DW0(unsigned int *pSerialNumDw0);

   Description: Retrieve the MotherBoard serial number lower 4 bytes.

   OUT: The lower 4 bytes will be saved in pSerialNumDw0.

   Successful Return value: STATUS_OK (0)

5. EloGetMotherBoardSerialNumber_DW1(unsigned int *pSerialNumDw1);

   Description: Retrieve the MotherBoard serial number higher 4 bytes.

   OUT: The higher 4 bytes will be saved in pSerialNumDw1.

   Successful Return value: STATUS_OK (0)

6. EloGetTabletSerialNumber_DW0(unsigned int *pSerialNumDw0);

   Description: Retrieve the Tablet serial number lower 4 bytes.

   OUT: The lower 4 bytes will be saved in pSerialNumDw0.

   Successful Return value: STATUS_OK (0)

7. EloGetTabletSerialNumber_DW1(unsigned int *pSerialNumDw1);

   Description: Retrieve the Tablet serial number higher 4 bytes.

   OUT: The higher 4 bytes will be saved in pSerialNumDw1.

   Successful Return value: STATUS_OK (0)

8. EloGetUUID_DW0(unsigned int *pUUID0);

   Description: Retrieve UUID lowest 4 bytes.

   OUT: The lowest 4 bytes will be saved in pUUID0.

   Successful Return value: STATUS_OK (0)

9. EloGetUUID_DW1(unsigned int *pUUID1);

   Description: Retrieve UUID lower 4 bytes.

   OUT: The lower 4 bytes will be saved in pUUID1.

Successful Return value: STATUS_OK (0)

10. EloGetUUID_DW2(unsigned int *pUUID2);

    Description: Retrieve UUID higher 4 bytes.

    OUT: The higher 4 bytes will be saved in pUUID2.

    Successful Return value: STATUS_OK (0)

11. EloGetUUID_DW3(unsigned int *pUUID3);

    Description: Retrieve UUID highest 4 bytes.

    OUT: The highest 4 bytes will be saved in pUUID3.

    Successful Return value: STATUS_OK (0)

12. EloSetMotherBoardSerialNumber_DW0(unsigned int SerialNumDw0);

    Description: Set the MotherBoard Serial Number lower 4 bytes

    IN: The data to be set

    Successful Return value: STATUS_OK (0)

13. EloSetMotherBoardSerialNumber_DW1(unsigned int SerialNumDw1);

    Description: Set the MotherBoard Serial Number higher 4 bytes

    IN: The data to be set

    Successful Return value: STATUS_OK (0)

14. EloSetTabletSerialNumber_DW0(unsigned int SerialNumDw0);

    Description: Set the Tablet Serial Number lower 4 bytes

IN: The data to be set

Successful Return value: STATUS_OK (0)

15. EloSetTabletSerialNumber_DW1(unsigned int SerialNumDw1);

    Description: Set the Tablet Serial Number higher 4 bytes

    IN: The data to be set

    Successful Return value: STATUS_OK (0)

16. EloSetUUID_DW0(unsigned int UUID0);

    Description: Set the UUID lowest 4 bytes

    IN: The data to be set

    Successful Return value: STATUS_OK (0)

17. EloSetUUID_DW1(unsigned int UUID1);

    Description: Set the UUID lower 4 bytes

    IN: The data to be set

    Successful Return value: STATUS_OK (0)

18. EloSetUUID_DW2(unsigned int UUID2);

    Description: Set the UUID higher 4 bytes

    IN: The data to be set

    Successful Return value: STATUS_OK (0)

19. EloSetUUID_DW3(unsigned int UUID3);

Description: Set the UUID highest 4 bytes

IN: The data to be set

Successful Return value: STATUS_OK (0)

Check out Our Website

www.elotouch.com

Get the latest...

- Product Information
- Specifications
- Upcoming events
- Press releases
- Software drivers

Getting in Touch with us

To find out more about the extensive range of Elo touch solutions, visit our website at www.elotouch.com, or simply call the office nearest you:

North America
Elo Touch Solutions
1033 McCarthy Blvd
Milpitas, CA 95035

Tel 800-ELO-TOUCH
Tel 1-408-597-8000
Fax 1-408-597-8050
customerservice@elotouch.com

Europe
Tel +32 (0) 16 70 45 00
Fax +32 (0) 16 70 45 49
elosales@elotouch.com

Asia-Pacific
Tel +86 (21) 3329 1385
Fax +86 (21) 3329 1400
www.elotouch.com.cn

Latin America
Tel 786-923-0251
Fax 305-931-0124
www.elotouch.com

First Edition (September 2013)