TZI SCIPIO WInspect Glove

Preliminary Datasheet

Description

The TZI SCIPIO WInspect Glove is a general-purpose wearable input device. It is in the form of a fingerless glove with minimal covering on the inside of the arm and wrist for comfort. It has three textile buttons that fit slugly around the fingers and do not interfere with manual tasks. It can be used in many wearable computing applications and was designed for hybrid gesture / direct selection interaction. The Glove could also be used as a simple activity/task recognition device.

Features

- Freescale MMA 7260 3-axis acceleration sensor
- cavitec MR-1000 RFID Reader 125 kHz (EM, Philips, SID, Temic)
- PromiESD02 Bluetooth Module (Class 2, RFComm)
- Microchip PIC 16F88 MCU
- Li-Ion battery pack for up to 8 hours operation time
- battery management (battery protection, charger, fuel gauge)
- modifiable firmware
- input keys, status LEDs, programmable audio feedback

Interface

The SCIPIO provides a Bluetooth interface that implements the Bluetooth profiles SDA and RFComm. The device is always in "discoverable" mode. The device identifies itself as "Scipio". In order to connect to a specific SCIPIO device, the Bluetooth MAC address of the specific device should be used instead of the generic device name.

Upon connection, the SCIPIO device starts sending sensor information. The sensor information is encoded as follows: The device sends packets of two byte length. The first byte identifies the data channel, the second byte contains the payload data for that channel. The device sends all data channels that are enabled in a round-robin schedule. The following data channels are provided:

Channel ID	Channel Data
0x41	X accelleration MSB low
0x42	X accelleration MSB high
0x43	Y accelleration MSB low
0x44	Y accelleration MSB high
0x45	Pushbutton State
0x46	RFID data
0x47	AuxAD data
0x48	Z accelleration MSB low
0x49	Z accelleration MSB high

In order to transmit 9-bit A/D data, two channel IDs are used where the channel ID used for transmission encodes the MSB of the data.

Upon RFComm connection, the SCIPIO starts the protocol with a valid channel ID identifier. Note that there is no escaping of payload data, i.e. the system takes no precausion if a channel ID identifier appears in the payload data. However, it is unlikely that all channels send in sequence contain all but valid channel IDs. In the

unlikely event that a part of the protocol stream remains in a buffer, the receiving side has to take precausions to resynchronize. This event is unlikely, as the RFComm protocol implements both CRC checks and an ARQ scheme.

Configuration and Control

The SCIPIO device can be configured and controlled by a host computer by sending a text command identifier that is followed by payload data. The commands are sent through the same RFComm connection that is used to receive data.

Command	Payload	Payload	Payload
PROTOCOL	Protocol Identifier	ON | OFF	-
SOUND	Event Identifier	freq	duration
VERSIONMAJOR	-	-	-
VERIONMINOR	-	-	-
1WIRE	-	-	-
PROGEXT	-	-	-

Valid Protocol Identifiers are SENDX, SENDY, SENDZ, SENDB, SENDRFID, SENDTYPE, SENDAUXIN and correspond to the different data channels.
Valid Event Identifiers are CONN, DISCONN, RFID, BUTTON and correspond to events that trigger a sound notification.

Application Development Support

In order to simplify application development, a number of software packages are available that support developers in integrating the device into their application software.

XSCIPIO: Linux X-Windows Pseudo-Mouse driver

The XSCIPIO software package allows a simple connection of the device to a Linux system with Bluetooth stack running X-Windows. The motion signals of the device are converted into mouse motion events. The buttons of the device trigger mouse click events. Altough not recommended, it is possible to operate a standard X-Windows Desktop with the device. The RFID scanner output is stored in a temporary file where it can be read by the application software. XSCIPIO includes functions to automatically connect and reconnect to specific or generic SCIPIO devices.

Java JSR82 Device Interface Class

For host systems with a Java VM and a JSR82 Java-Bluetooth-API, a device interface class is available. It can establish the connection to a specific device and deliver its sensor data to an application program. Possible host systems are Linux and Micrososft Windows systems and java-enabled mobile devices such as PDAs and mobile phones.

WUI-TK Interaction Device Driver

The WUI (Wearable User Interface) Toolkit, which is a part of the European Wearable Computing Framework contains a device driver for the device. It allows gesture-based interaction with the application programs using the device.

Specifications and features are subject to change without notice