

*SL-264
Dallas Touch
Memory
(iButton)
decoder for PC
keyboard*

CardWare

MIV



Dallas iButton (touch memory)

- ❑ *Guaranteed uniqueness of each iButton*
- ❑ *Resistant to most kinds of abuse*
- ❑ *Unlimited areas of application*

Keyboard wedge decoder for Dallas touch memories:

- ❑ *Simple to use – no changes necessary in application software*
- ❑ *Quick and easy identification of iButton holder. Possible applications include password entry at workstation logon, issuing goods in a warehouse, attendance control at fitness clubs etc.*

 **CardWare**
Stražilovska 4, 21000 Novi Sad
www.cardware.co.yu
Tel/Fax: +381-21-612-335
Mobile: +381-63-513-741
e-mail: info@cardware.co.yu



Technical details

The **SL-264** is a Dallas touch memory (iButton) reader (decoder) which is connected to a standard PC as a keyboard "wedge". This means that the decoder is inserted between the PC and the keyboard.

Connection

- turn off the PC
- pull the keyboard cable out of the connector on the motherboard
- insert into that connector the round plug of the cable supplied with the SL-264T
- the other end of that cable (with the telephone plug) should be inserted into the appropriate female connector of the SL-264T
- insert the keyboard cable plug into the round female connector of the SL-264T
- turn on the PC

Use

The decoder reads the unique ID code (ROM) of each touched iButton. The contact cup of the decoder should be touched (best at a slight angle) with an iButton. A successful read is signaled with a beep and the green LED lights up for a short time. The decoded 16-character ID code is sent to the PC's keyboard buffer just as if it had been typed on the keyboard.

A trailing <ENTER> is also sent.

The application program cannot tell whether the data is coming from the keyboard or the decoder.

(It is, in fact, possible to ascertain whether the data is being typed or is coming from the decoder. This involves measuring the time between the first received character and the last one. It isn't possible to type a 16-character string as fast as the decoder can send it.

If this additional security measure is necessary, i.e. if manual entry of the iButton code has to be recognized, the application software on the PC has to take into account the aforementioned time between the arriving characters.)

Format of decoded data

The data is sent in the same order as engraved on the iButton itself:

1. two digits of the 8-byte **CRC** (one byte)
2. twelve digits of the serial number (six bytes)
3. two digits of the **Family Code** (one byte)
4. plus the trailing ENTER

e.g.

CCNNNNNNNNNNNNFF
<ENTER>

The characters represent hexadecimal numbers and can be any of the following: (0 – 9, A – F).

The letters A to F will be sent as capitals or not, depending on the state of the CAPS-LOCK lamp on the keyboard.

The **CRC** represents a control character for checking the validity of the read ID code.

Family Code is a byte that is unique for all iButtons of a certain type (family). For instance the DS-1990A has a family code of 01.

(The DS-1990A is the simplest and cheapest type of iButton that only has a ROM ID-code and no additional functions).

As you can see, only the ROM is read, which means that the special functions of the more complex touch memories can not be used with this decoder. This is logical, since all of these functions require write operations to the iButton and the keyboard interface is not appropriate for sending commands out from the PC.

Application

The basic application of this device is the quick identification of the iButton (or its holder):

- password entry at workstation logon
- issuing goods in warehouses: identification of issuing and receiving employee
- identification of club members
- ...

The device can be supplied such that the contact cup and signal LED are mounted in a separate sturdy wall mounting plastic box, which can be up to 30m away from the decoder (which has to be next to the PC). The decoder is connected to this contact box with a flat 4-way telephone-type cable.

This extends the possible application areas of the decoder. A simple parking time tracking system can be made for instance.