

C.14. Scemtec SHL-2001 Long Range HF-Reader

Bei den Lesegeräten handelt es sich um 15 HF-Industrielesegeräte *SHL-2001* von *scemtec Transponder Technologies*[®], siehe [Sce], für den 13.56 MHz-Bereich für die Standards *MIFARE*, *ISO 14443 A* und *ISO 15693*.

C.14.1. Hardware

Die Lesegeräte liegen in stabilem Metallgehäuse und Anschlüssen für zwei Antennen und RS232-Anschluss sowie eingebauten Netzteil vor⁵⁵, siehe Bild C.16.



(a) Frontseite und Anschlüsse



(b) Gehäuse geöffnet

Bild C.16.: Scemtec HF Lesegerät SHL-2001

Das Lesegerät *SHL-2001* enthält ein HF-Teil mit 2 W Leistung, welches über ein Relais über einen Software-Befehl auf den Antennenausgang 1 oder 2 geschaltet werden kann. In der Grundeinstellung ist Ausgang 1 aktiv. Eine häufige Umschaltung zum Betrieb von zwei Antennen

⁵⁵Die Aktualisierungssoftware *Flasher*, siehe Abschnitt C.14.3 zeigt die folgende Information: Device Class: LR, Board Rev.: 2, BL Ver.: 3

Gerät unter Backup gespeichert, siehe Bild C.19b. Der Vorgang darf auf keinen Fall unterbrochen werden!

C.14.4. Software

Beim Übergang von Windows 32-bit auf 64-bit müssen neuere Softwareversionen verwendet werden [Doh15], die unter Downloads⁵⁷ mit geschütztem Zugang erhältlich sind. Dabei muss eine entsprechend neue Version des Java-Runtime-Environments (JRE) verwendet werden. Meist ist diese auf den Laborrechnern vorhanden. Falls nicht, beispielsweise `jre-7u51-windows-x64.com` installieren.

Die folgende Software wird empfohlen:

vor Installation von *UniDemo* und *StxTerm* muss ein passendes JRE⁵⁸ installiert werden.

UniDemo Software in der Version 2.6.3.

StxTerm Software in der Version 4.2.6 funktioniert mit den Geräten.

Terminalprogramme Da neben den Kommandos auch immer STX⁵⁹, ETX⁶⁰ und ein CRC übertragen werden müssen, ist *TeraTermPro*, siehe Fußnote 6 auf Seite 151 weniger gut geeignet. Besser ist *Hterm*, siehe Fußnote 63 auf Seite 186.

Bei der Installation wird gegebenenfalls nach der Installation von Gerätesoftware gefragt, dann wie in Bild C.20 vorgehen.



Bild C.20.: StxTerm - Installation von Gerätesoftware

⁵⁷<http://www.stt-rfid.de/>

⁵⁸z.B. `jre-7u51-windows-x64.com`

⁵⁹Start-of-Transmission

⁶⁰End-of-Transmission

C.14.5. Inbetriebnahme



Reader besitzt vier Hochfrequenz-Ausgänge, die stets mit 50 Ω abgeschlossen sein müssen. Die Netzspannung sollte erst eingeschaltet werden, wenn alle Ausgänge mit Antennen oder Abschlusswiderständen von 50 Ω verbunden sind. Ansonsten führen Reflexionen durch Einstrahlung möglicherweise zu Störungen mit dem Hostrechner! [Kal10]

Standardmäßig sind die *scemtec*-Lesegeräte auf die folgenden RS232-Kommunikationsparameter eingestellt; gegebenenfalls müssen sie beim Arbeiten mit den o.g. Programmen angepasst werden. Beim Terminalprogramm empfiehlt sich zusätzlich beim Senden die Einstellung CR + LF.

Maximum Speed (baud rate) 9 600

Data Bits 8

Parity N

Stop Bits 1

Echo Off

Flow Control None

Erste Tests mit den Geräten, siehe Abschnitt C.14.7

C.14.6. SFTTest 3



Dieses Kapitel ist voraussichtlich obsolet geworden!

C.14.7. UniDemo

Das *SHL-2001* wird mit der RS232 des PC verbunden und die Software *UniDemo* gestartet, vergl [Cic07]. Für den ersten Test sollten beide Antenneausgänge mindestens mit 50 Ω -Abschlusswiderständen verbunden sein. Im Programm *UniDemo Launcher* werden Grundeinstellungen für den Verbindungsaufbau und die Konfiguration des Geräts vorgenommen, siehe Bild C.21.

Die Software *UniDemo* erlaubt das Lesen und Schreiben von Tags. Gerät und Software beherrschen Antikollision, siehe Bild C.22.

Dabei ist zu beachten, dass bei der im Bild gezeigten Standard-Einstellung Find Fastest (...) die Grundeinstellungen des Lesergeräts insbesondere die Baudrate, siehe Abschnitt C.14.5, geändert werden.

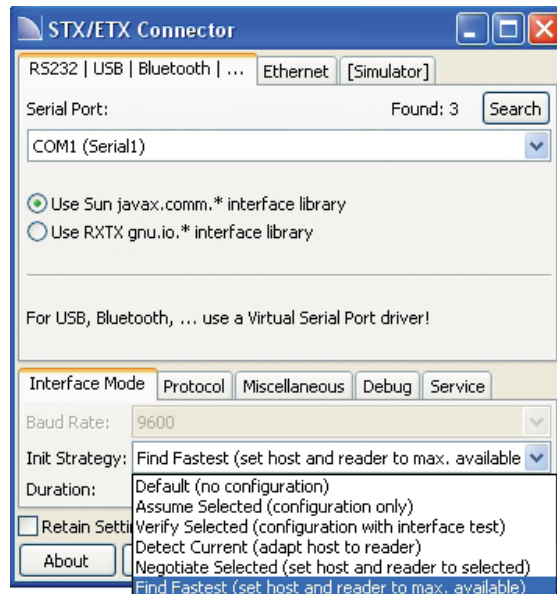


Bild C.21.: UniDemo Launcher

Das *Beschreiben* eines Tags kann nur erfolgen, wenn sich ein einzelner Tag im Lesefeld befindet und mit beispielsweise `Single Scan` erfasst wurde. Mit `Read/Write` wird ein weiteres Fenster geöffnet und gegebenenfalls mit `Read` aktualisiert. Der Karteireiter `Tag Properties`, Bild C.22c, zeigt die Tag-Eigenschaften.

Der Speicherbereich des Tags lässt sich beispielsweise im Karteireiter `Tag Data Sequential View` modifizieren, siehe Bild C.22d. Dabei können Hex- oder `ASCII`-Daten verändert werden oder Beispieldaten - `Generate Example Data` - erzeugt werden. Danach müssen mit `Insert to Block Buffer` oder `Fill Block Buffer` die geänderten Daten aktualisiert werden. Das Schreiben auf den Tag erfolgt mit `Write`.

Zur Nachvollziehbarkeit und für die Entwicklung eigener Software können die angewandten STX/ETX-Befehle aufgezeichnet werden, wie in Bild C.22b gezeigt.

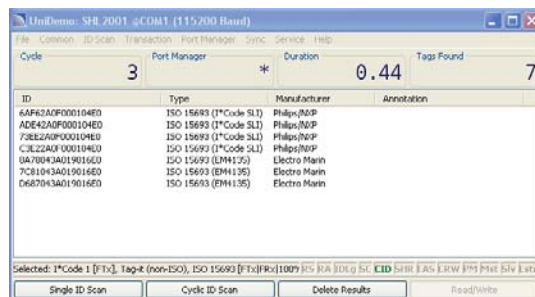
Zusätzlich kann unter dem Menü `Service → STX/ETX Log → Start for this Connection` eine Log-Datei geöffnet werden und mit `... → ... → View current Log...` angezeigt werden.

Das Starten von `StxTerm`, siehe Abschnitt C.14.8, kann ebenfalls von dieser Stelle aus erfolgen.

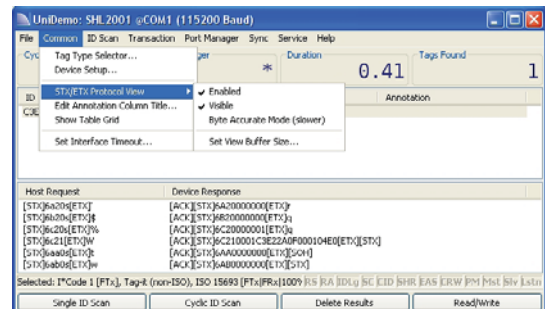
C.14.8. StxTerm

Das *SHL-2001* wird mit der *RS232* des *PC* verbunden und die Software *StxTerm* gestartet.

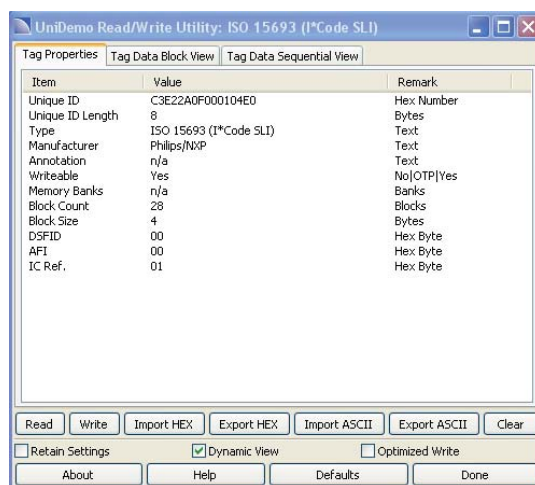
In der Kommandozeile werden folgende Befehle `command` abgesetzt und die Antwort `response`



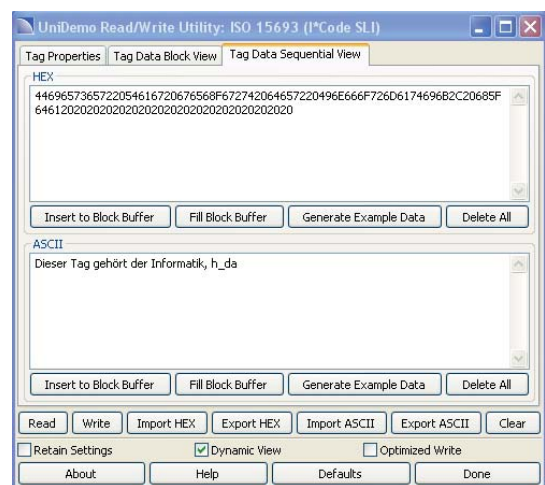
(a) Scannen von Tags



(b) Befehlsfolge



(c) Taginformationen



(d) Schreiben von Tags

Bild C.22.: Scemtec UniDemo Funktionalität

betrachtet:

1001	(command)
2 1001SHL1/0200	(response: Firmware/Version des Geräts)

Mit diesem ersten Test ist sichergestellt, dass das Gerät korrekt angeschlossen ist und eine funktionierende Verbindung besteht.

Bei den folgenden Tests muss mindestens eine Antenne⁶¹ angeschlossen sein. Für die Befehle siehe [Sti06, S. 48, 114-115]⁶². Als Transponder wird ein *ISO 15693*-Tag gewählt:

⁶¹beispielsweise *Feig*[®] ICC.ANT300/300

⁶²Die Spezifikation [Len09] ist zu neu für die vorhandene Firmware und bezieht sich zum Teil auf neuere Geräte [Len10b]. Daher wird die Spezifikation [Sti06] empfohlen

Listing C.6: Kommandos für Scemtec SHL-2001

```

F00001      (command: Einschalten der HF)
2 F000      (response: ok)

4 4C1000n    (command: Lesen 0. Block)
  4C100y77656C74 (response: Daten z.B. 77656C74)
6 4C1001n    (command: Lesen 1. Block)
  4C100y68657272 (response: Daten z.B. 68657272)
8 4C1002n    (command: Lesen 2. Block)
  4C100y73636861 (response: Daten z.B. 73636861)
10 4C1003n   (command: Lesen 3. Block)
   4C100y66742121 (response: Daten z.B. 66742121)
12
  4C120003n      (command: Lesen Blöcke 0-3)
14 4C120y77656C74686572727363686166742121 (response: Daten...)

16 4C16n      (command: lese Systeminformation)
   4C160y0FADE42A0F000104E000001B0301 (response: Tag-ID=DE42A0F0 ?)
18
  F00000      (command: Abschalten der HF)
20 F000      (response: ok)

```

Nach dem erfolgten Firmware-Update, siehe Abschnitt C.14.3, stellen sich die Befehle und Antworten beispielsweise wie folgt dar:

Listing C.7: Kommandos zum Lesen von ISO15693-Tags für Scemtec SHL-2001

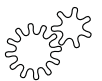
```

1001      (command)
2 1001SHLX/0065/SHL2001 (response: Firmware/Version des Geräts)

4 6C20s    (command: create inventory)
  6C20000003 (response: 3 tags in field)
6
  6C21      (command: get inventory)
8 6C2100036AF62A0F000104E073EE2A0F000104E0C3E22A0F000104E0
   (response: 3 tags, 3 id's)

```

Beispiel C.1. Bei einem der Tests wurden in 0.86s 29 Tags gefunden



C.14.9. Zusammenstzung und Senden der Kommandos

Die Kommandos *cmd*, welche über die serielle Schnittstelle gesandt werden, müssen der folgenden Struktur folgen: *[STX]cmd[ETX]{CRC}*, wobei *CRC* die Prüfsumme ist.



Die Kommandos sind in die [ASCII](#)-Steuerzeichen Start of Text (STX) und End of Text (ETX) einzubetten - siehe auch [[Wik](#), Steuerzeichen] - und mit dem entsprechenden [CRC](#) zu ergänzen.

In der Grundkonfiguration ist der *scemtec*[®] *SHL-2001* auf 9600 baud eingestellt, siehe Abschnitt [C.14.5](#).

Gut geeignet für die Tests ist das Terminalprogramm *HTerm*⁶³

Mit *StxTerm*, siehe Abschnitt [C.14.8](#), können einzelnen Befehle an den *SHL-2001* abgesetzt und die Antwort ausgewertet werden. Will man die genaue Zusammensetzung der Kommunikation sehen, empfiehlt sich ein Programm zum Sniffen der Kommunikation über die [RS232](#)-Schnittstelle. Dazu kann beispielsweise *Device Monitoring Studio* ver. 7.21.00.6146 und dabei der *Serial Monitor* von *HHD Software Ltd*^{®64} verwendet werden.

Damit kann der Rahmen (frame) der Befehle mitsamt Prüfsumme dargestellt werden, deren Antwort aus dem oberen Abschnitt ja schon bekannt ist:

Listing C.8: Hex-Werte für Scemtec SHL-2001 aus Device Monitoring Studio

```
1 02 31 30 30 31 03 01      .1001..
   02 46 30 30 30 30 31 03 76 .F00001.v
3 02 36 43 32 30 73 03 05    .6C20s..
   02 36 43 32 31 03 77      .6C21.w
5 02 46 30 30 30 30 30 03 77 .F00000.w
```

C.14.10. Java-Code für Kommandos

Ein Beispiel wurde vom Autor entwickelt und ist unter
D:/Daten/Development/Java/NetBeans/Scemtec (privates Verzeichnis) abgelegt.

Listing C.9: Berechnung CRC in Java für Scemtec SHL-2001

```
1 /**
   * Calculate CRC for scemtec SHL-2001
3  * @param bArr payload
   * @return crc for scemtec SHL-2001
5  */
   public static byte calcScemtecCRC( byte[] bArr )
7  {
       byte crc = 0x0;           // initialize CRC
9       for (int i = 0; i < bArr.length; i++ ) {
           crc ^= bArr[i];       // XOR
11      }
       return crc;
13 }
```

⁶³<http://www.der-hammer.info/terminal/>

⁶⁴<http://www.hhdsoftware.com/>

Listing C.10: Berechnung des vollständigen Befehl mit STX, ETX und CRC in Java für Scemtec SHL-2001

```

1 /**
   * Calculate full command with STX, ETX and CRC from a comand for scemtec SHL-2001
3  * @author Ralf S. Mayer, june 2015
   * @param cmd payload only (w.o. STX and ETX)
5  * @return full command string for scemtec SHL-2001 including STX, cmd, ETX, CRC
   */
7 public static byte[] calcScemtecFullCmd( byte[] cmd )
   {
9     byte bArr[] = new byte[cmd.length + 2];      // STX, cmd, ETX
       bArr[0] = STX;                               // start with STX
11    for (int i = 0; i < cmd.length; i++ ) {
       bArr[i+1] = cmd[i];                          // fill after STX
13    }
       bArr[cmd.length + 1] = ETX;                   // end with ETX
15    byte crc = calcScemtecCRC( bArr );             // get CRC

17    // new array with CRC
       byte bArr2[] = new byte[bArr.length + 1];    // STX, cmd, ETX, CRC
19    for (int i = 0; i < bArr.length; i++ ) {
       bArr2[i] = bArr[i];                          // copy
21    }
       bArr2[bArr.length] = crc;
23    return bArr2;
   }

```

Listing C.11: Formatierung byte array in Dezimalzahlen in Java für Scemtec SHL-2001

```

/**
2  * Output command array as decimals
   * @author Ralf S. Mayer, june 2015
4  * @param cmd
   * @return
6  */
public static String cmdToDecString( byte[] cmd ){
8     StringBuffer buf = new StringBuffer();
       for (int i = 0; i < cmd.length - 1; i++ ) {
10        buf.append(String.format( "%03d", cmd[i] ) + ", " );
       }
12    buf.append( String.format( "%03d", cmd[cmd.length - 1] ) );

14    return buf.toString();
}

```

Listing C.12: Formatierung byte array in Hexadezimal in Java für Scemtec SHL-2001

```

1 /**
   * Output command array as hex
3  * @author Ralf S. Mayer, june 2015

```



```

* @param cmd
* @return
*/
5 public static String cmdToHexString( byte[] cmd ){
    StringBuffer buf = new StringBuffer();
    9     for (int i = 0; i < cmd.length - 1; i++ ) {
        buf.append(String.format( "%02X", cmd[i] ) + ", " );
    11    }
    buf.append( String.format( "%02X", cmd[cmd.length - 1] ) );
    13    return buf.toString();
}

```

Listing C.13: Benutzung der Funktionen C.9 bis C.12

```

// ...
2 String command = "1001";           // Reader command as String
  byte[] cmd = command.getBytes();   // convert to bytes
4 byte[] fullCmd = calcScemtecFullCmd( cmd );
  System.out.println( command + "\t:" + cmdToDecString( fullCmd ) +
6                        "\t:" + cmdToHexString( fullCmd ) );
// ...
8 command = "F00001";                // ...
  command = "F00000";                // ...
10 command = "6C20s";                 // ...
  command = "6C21";                  // ...

```

Listing C.14: Ausgabe aus Listing C.13

```

1 1001 : 002, 049, 048, 048, 049, 003, 001      : 02, 31, 30, 30, 31, 03, 01
  F00001: 002, 070, 048, 048, 048, 048, 049, 003, 118 : 02, 46, 30, 30, 30, 30, 31, 03, 76
3 F00000: 002, 070, 048, 048, 048, 048, 048, 003, 119 : 02, 46, 30, 30, 30, 30, 30, 03, 77
  6C20s : 002, 054, 067, 050, 048, 115, 003, 005      : 02, 36, 43, 32, 30, 73, 03, 05
5 6C21 : 002, 054, 067, 050, 049, 003, 119          : 02, 36, 43, 32, 31, 03, 77

```