AUTHOR

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Department** |  | **Signature** |  | **Print Name** |  | **Date** |
| R&D |  |  |  |  |  |  |

SIGNATURES

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Department** |  | **Signature** |  | **Print Name** |  | **Date** |
| R&D |  |  |  |  |  |  |

Revision History

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Rev.** |  | **Date** |  | **Initials** |  | **Description** |
| 0.1 |  | 19-Jun-2019 |  | DAT |  | Initial draft derived from 15000704 Rev A |
| 0.2 |  | 21-Jun-2019 |  | DAT |  | Updates after review, remove need for MPLAB |
| 0.3 |  | 24-Jun-2019 |  | DAT |  | Added Port B Testing |
| 0.4 |  | 26-Jun-2019 |  | DAT |  | Updates after moving Motor Table and Serial Number from 15000286 to 15008940 |
| 0.5 |  | 28-Jun-2019 |  | DAT |  | Update to consistent tense, removed EEPROM check since it is out of scope of the protocol. |
| 0.6 |  | 05-Jul-2019 |  | KW |  | Corrections from dry run, use EZ Tap instead of scope |
| 0.7 |  | 08-Jul-2019 |  | KW |  | More corrections from dry runs, cleanup of Software Update section |
| 0.8 |  | 23-Jul-2019 |  | KW |  | Change loop count from 14 to 12 to reflect previous update to array size.  Added additional test for Program Page Request CRC |
| 0.9 |  | 23-Aug-2019 |  | KW |  | Add instruction to insert programmed USB before power-on of the Shaver for the Program Page test  Correct inconsistent converter pin specification  Make all tests use TestApp only for easier execution |
| A |  | 26-Sep-2019 |  | DAT |  | Updated to Revision A |

References

1. 15000286 – DYONICS II RS485 Accessory Protocol
2. 15008940 – Reliant RS485 Protocol Specification

Equipment required for verification

1. Reliant Handpiece with Hand Controls REF (72205321) or Reliant Test Tool 20604462
2. DYOINCS II EIP Shaver System modified for debugging purposes, per 11500060 and running TestApp
3. RS-485 to RS-232 Serial Converter (used to convert the A/B RS485 differential lines into a RS232 serial output that can be read by a PC)
4. Computer with Windows 7/10 with an RS232 port and Serial Port Monitoring Program (e.g. RealTerm) installed and EZView software installed
5. EZ-Tap Pro
6. Host computer with Windows 2000/XP and the following software and hardware installed on it:
   1. Microsoft Embedded Visual C++ 4.0 Debugger for the Shaver Application
   2. Source code for Shaver application

| **Description** | **SKU/Model Number** | **Serial Number** | **Software** |
| --- | --- | --- | --- |
| DYONICS POWER II EIP | 72200873 |  |  |
| RELIANT MDU |  |  |  |
| RS-485 to RS-232 |  |  |  |
| Serial Monitoring Software |  |  |  |
| EZ-Tap Pro with EZView |  |  |  |

Table of Contents

[1 Overview 3](#_Toc13484662)

[2 Protocol Verification 3](#_Toc13484663)

[3 SUMMARY 15](#_Toc13484664)

[3.1 Notes 15](#_Toc13484665)

[3.2 Overall Pass/Fail Status 16](#_Toc13484666)

Reliant RS485 Protocol Verification

# Overview

The purpose of this document is to:

* + Show the traceability of the verification procedures against the DYONICS II RS485 Accessory Protocol and the Reliant RS485 Protocol (Reliant)
  + Provide a summary of the Verification procedures
  + Provide a summary of the Verification results
  + Provide a link to more detailed internal engineering verification reports when required

Verification is defined as the process by which the design output meets the design input requirements. The results of the design verification, including identification of the design, method(s), the date, and the individual(s) performing the verification, shall be documented in the DHF. Design verification shall be traceable to product specifications.

The verification is traceable to the following functional requirement and specification documents:

Dyonics II RS485 Accessory Protocol – Document # 15000286

Reliant RS485 Protocol Specification – Document # 15008940

# Protocol Verification

To setup for the Reliant RS485 Protocol verification:

* Connect the RS-485 to RS-232 Serial Converter to one side of the EZ-Tap Pro.
* Connect the other side of the EZ-Tap Pro to the Computer.
* Connect the USB cable from the EZ-Tap Pro to the Computer.
* Start Serial Monitoring application on the Computer. Configure the application for 57600 baud, 8N1 (8 bit data, no parity, 1 stop bit). Set the Serial Monitoring Software to display Hexadecimal values and open up the COM port.
* Start the EZView application.

There are several elements used in the configuration. They are:

|  |  |
| --- | --- |
| Element | Description |
| Shaver | Using Microsoft Embedded Visual C++ 4.0 to debug the main shaver app on TestApp |
| Monitor | Using a Serial Monitoring program on the computer, to display the RS-485 serial data. |
| EZView | Using an EZ-Tap Pro with EZView program to monitor specific timing of the RS-485 protocol. |

Each test uses one or more of the elements above.

The following is a list of all the tests. The Reliant MDU is the RS485 slave in all these tests:

| **Section Number** | **Verification Procedure Summary** | **Summary Results**  **(may include links to other verification reports)** | **Pass / Fail**  **Port** | | **Initial / Date** |
| --- | --- | --- | --- | --- | --- |
| **A** | **B** |  |
| DYONICS II 2.1 a.  2.2.1.  Reliant  2.1.2 | Verify that 8 bit bytes are sent over the bus using no parity and one stop bit at 57600 baud.  Verify that bits 9-0 of the response to the Accessory ID request, 0xB1, for the Reliant MDU w/Hand Controls is 3. | Monitor and EZView:  Hooked up the pins 1 & 14 of J1 – Port A or J6 – Port B on the EIP Board to pin - & + of the converter. Started the Serial Monitoring and EZView capture.  Shaver, Monitor and EZView:  Started the Shaver device. Started the Shaver Application. Plugged a Reliant MDU w/Hand Controls in Port A of the Shaver. Stopped the captures. Reviewed the Serial capture and verified that following serial byte sequence occurred:  B1 🡪 CC 03 | **P**  **F**  **P**  **F** | **P**  **F**  **P**  **F** |  |
| DYONICS II 2.1 b. | Verify that the master initiates all communication by sending command requests and the only time a slave transmits on the bus is in response to a master command. | Reviewed the serial capture and verified that no communication occurs prior to sending of the 0xB1 by the shaver.  Verified that after the reply to the Motor Table Command Request (0x4E) and Serial Number Command Request (0xAA), data was in the form of a 1-byte request from the DII followed by a 2-byte response from the Reliant MDU. | **P**  **F** | **P**  **F** |  |
| DYONICS II 2.2  Reliant  2.1.1 | Verify that the Software Version Command Request, 0x00, response is a 2-byte message. | The Reliant MDU w/Hand Controls responded to the Software Version Command Request 0x00 with a 2-byte Message. The lowest 2 bits of the first byte represented the Major Version, the upper 4 bits of the second byte represented the Minor Version, and the lower 4 bits of the second byte represented the Build Version.    00 🡪 B4 21  Brought up the System Information screen and verified that the software version matched the version of the MDU under test.  Handpiece Version Port A 0.02.01 | **P**  **F** | **P**  **F** |  |
| DYONICS II 2.2.5  Reliant  2.1  2.1.6 | Verify that Motor Table Command Request, 0x4E, Response is a 34 byte message and CRC.  Verify that the response to the Motor Table Command Request is within 10 milliseconds of receiving the command request. | The Reliant MDU w/Hand Controls responded to the Motor Table Command Request 0x4E with a 34 byte Message and a CRC byte  4E 🡪 1E 00 01 04 60 EA 83 20 52 70 10 27 05 19 05 20 05 20 05 20 19 00 19 10 05 10 71 42 52 03 14 00 4B 10 87  The EZView capture showed the response to the Motor Table Command Request was within 10 milliseconds of receiving the command request. | **P**  **F**  **P**  **F** | **P**  **F**  **P**  **F** |  |
| DYONICS II 2.2.5  Reliant  2.1  2.1.7 | Verify that the response to a Serial Number Command Request, 0xAA, is an 11 Byte message and CRC.  Verify that the response to the Serial Number Command Request is within 10 milliseconds of receiving the command request. | The Reliant MDU w/Hand Controls responded to the Serial Number Command Request 0xAA with an 11 bytes Message and a CRC byte.  AA 🡪 41 41 5A 31 32 33 34 35 00 00 00 D6  Brought up the System Information screen and verified that the serial number matched the serial number of the MDU under test.  SN AAZ12345  The EZView capture showed the response to the Serial Number Command Request was within 10 milliseconds of receiving the command request. | **P**  **F**  **P**  **F** | **P**  **F**  **P**  **F** |  |
| Reliant 2.1.4 | Verify that Button State Command Request 0xE4 Response is dependent upon button status. | Pressed multiple combinations of buttons on the Reliant MDU Buttons.  The Reliant MDU w/Hand Controls responded to each Button State Command Request 0xE4 with the combined status of the buttons.   | Button Pressed | | | Reliant  Response | | --- | --- | --- | --- | | Left | Middle | Right | | No | No | No | FC 00 | | Yes | No | No | 54 02 | | No | Yes | No | 64 01 | | No | No | Yes | 4C 04 | | Yes | Yes | No | CC 03 | | No | Yes | Yes | D4 05 | | Yes | No | Yes | E4 06 | | Yes | Yes | Yes | 7C 07 | | **P**  **F** | **P**  **F** |  |
| Reliant  2.1.3 | Verify that Blade ID Command Request 0xD2 Response is dependent upon button status. | The Reliant MDU w/Hand Controls responded to each Blade ID Command Request 0xD2 with the Blade Code based upon the magnet code of the blade hub inserted at the time of the request.   | Blade Code | Magnet Code\* | Blade Max Speed | Reliant  Response | | --- | --- | --- | --- | | 0 | 00 | 5000 | FC 00 | | 1 | S0 | 3000 | 64 01 | | 2 | 0S | 8000 | 54 02 | | 3 | SS | 10000 | CC 03 | | 4 | 0N | Reserved1 | 4C 04 | | 5 | N0 | Reserved1 | D4 05 | | 6 | NN | Reserved1 | E4 06 | | 7 | NS | Reserved1 | 7C 07 | | 8 | SN | Reserved1 | C4 08 |   1 Reserved – Reports “Unknown Blade” warning and Blade Max Speed is 3000. | **P**  **F** | **P**  **F** |  |
| Reliant  2.1.3  2.1.4 | Verify that Blade ID Command Request 0x63 Response is dependent upon button status. | The Reliant MDU w/Hand Controls responded to each Blade ID Command Request 0x63 with the Blade Code based upon the magnet code of the blade hub inserted at the time of the request. The Blade ID response did not change when the running state of the MDU changed.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Blade Code | Reliant  Response | Forward | Reverse | Oscillate | Window Lock | Window Lock Front Panel | | 0 | FC 00 |  |  |  |  |  | | 1 | 64 01 |  |  |  |  |  | | 2 | 54 02 |  |  |  |  |  | | 3 | CC 03 |  |  |  |  |  | | 4 | 4C 041 |  |  |  |  |  | | 5 | D4 051 |  |  |  |  |  | | 6 | E4 061 |  |  |  |  |  | | 7 | 7C 071 |  |  |  |  |  | | 8 | C4 081 |  |  |  |  |  |   1 Reports “Unknown Blade” warning and Blade Max Speed is 3000. | **P**  **F** | **P**  **F** |  |
| DYONICS II 2.1 c.  2.1.d.  2.1.e.  2.2.  Reliant  2.1 |  | Monitor and EZView:  Hooked up the pins 1 & 14 of J1 – Port A or J6 – Port B on the EIP Board to pin - & + of the converter. Started the Serial Monitoring and EZView capture.  Shaver:  Plugged a Reliant MDU w/Hand Controls in Port A of the Shaver.  Modified the shaver code software by inserting the following code segment:  {  SnWord wRequest;  static SnByte pbReqs[12] =  {0x00,0xB1,0xD2,0x63,0xE4,0x55,0x36,0x87,  0x78,0xC9,0x1B,0x2D};  SnQByte qCnt;  for (qCnt = 0; qCnt < 12; qCnt++) {  wRequest = HAND\_PORT\_CMD(PORTA, pbReqs[qCnt]);  SendSerialRequest(&wRequest, 4);  }  }  Right before the return TRUE; line of CControl::Init() in Control.cpp.  Set a break point at return TRUE; line of CControl::Init() in Control.cpp. Started the Shaver Application and waited till it stops at the breakpoint.  Monitor and EZ-View:  Stopped the captures. | **P**  **F** | **P**  **F** |  |
|  | Verify that command requests are one byte. | The serial capture showed that each command request was one byte. | **P**  **F** | **P**  **F** |  |
|  | Verify that slave responses are two bytes. | The serial capture showed that each slave response was two bytes. | **P**  **F** | **P**  **F** |  |
|  | Verify slave responses are within 3 milliseconds of receiving the command request. | The EZView capture showed that each slave response was within 3 milliseconds of receiving the command request. | **P**  **F** | **P**  **F** |  |
| DYONICS II 2.1 f. 1)  2.1 f. 2)  2.1 f. 3) |  | Monitor:  Hooked up the pins 1 & 14 of J1 – Port A or J6 – Port B on the EIP Board to pin - & + of the converter. Started the Serial Monitoring capture.  Shaver and Monitor:  Plugged a Reliant MDU w/Hand Controls in Port A of the Shaver.  Modified the shaver code software by inserting the following code segment:  {  SnWord wRequest;  wRequest = HAND\_PORT\_CMD(PORTA, SERIAL\_CMD\_DEV\_TYPE);  SendSerialRequest(&wRequest, 4);  wRequest = HAND\_PORT\_CMD(PORTA, 0xFF);  SendSerialRequest(&wRequest, 4);  }  Right before the return TRUE; line of CControl::Init() in Control.cpp.  Set a break point at return TRUE; line of CControl::Init() in Control.cpp. Started the Shaver Application and waited till it stops at the breakpoint. Verified that following serial byte sequence occurred:  0xB1 🡪 CC 03  0xFF 🡪 00 00 | **P**  **F** | **P**  **F** |  |
|  | Verify that bits 3-0 are used to encode outgoing commands and that command value 0xF gets a NAK response. | The command byte 0xB1 contained the 0x1 command in bits 3-0 which is the Accessory ID request. The response word of 0xCC03 indicated that it is Accessory ID #4 which corresponds to the Reliant MDU w/Hand Controls.  The command byte 0xFF contained the 0xF command in bits 3-0 and Bit 10 is clear indicating a NAK response. | **P**  **F** | **P**  **F** |  |
|  | Verify that bits 6-4 are used as ECC bits for outgoing commands. | The command byte 0xB1 contained the 0x1 command in bits 3-0 and bits 6-4 contain the value 0x3 which is the correct ECC value when bits 3-0 are the value 0x1. | **P**  **F** | **P**  **F** |  |
|  | Verify that bit 7 is used as the parity bit for outgoing commands. | The command byte 0xB1 contained the parity bit set in bit 7 which is the correct parity when bits 6-0 are the value 0x31. | **P**  **F** | **P**  **F** |  |
|  | Verify that return value is stored in bits 9-0 in a slave response. | The command byte 0xB1 contained the 0x1 command in bits 3-0 which is the Accessory ID request. The response word to the Accessory ID request command was 0xCC03. Bits 9-0 are the value 0x3 which corresponded to the Reliant MDU w/Hand Controls. | **P**  **F** | **P**  **F** |  |
|  | Verify that bit 10 is used as the ACK/NAK bit in a slave response. | The command byte 0xB1 contains the 0x1 command in bits 3-0 which was the Accessory ID request. The response word to the Accessory ID request command was 0xCC03. Bit 10 is set which indicated that the command request was valid. | **P**  **F** | **P**  **F** |  |
|  | Verify that bits 11 through 14 are used as ECC bits in a slave response. | The command request byte 0xB1 contained the 0x1 command in bits 3-0 which is the Accessory ID request. The response word to the Accessory ID request command was 0xCC03. Bits 14-11 contain the value 0x9 which is the correct ECC value when bits 9-0 are the value 0x3. | **P**  **F** | **P**  **F** |  |
|  | Verify that bit 15 is used as a parity bit in a slave response. | The command byte 0xB1 contains the 0x1 command in bits 3-0 which is the Accessory ID request. The response word to the Accessory ID request command was 0xCC03. Bit 15, (the parity bit), is set which was the correct parity when bits 14-0 are the value 0x4C03. | **P**  **F** | **P**  **F** |  |
|  | Verify that for erroneous requests, the NAK bit is set, bits 9-0 are cleared, and parity and ECC bits set accordingly. | The command byte 0xFF contained the 0xF command in bits 3-0 which is not a valid request. The response word to the invalid request is 0x0000. Bit 10 is clear which corresponds to a NAK. | **P**  **F** | **P**  **F** |  |
| DYONICS II 2.2.2  2.2.3  2.2.4 |  | Monitor:  Hooked up the pins 1 & 14 of J1 – Port A on the EIP Board to pin - & + of the converter.  PC:  Programed a USB Flash drive with an image of the Reliant MDU Software  Shaver:  Powered off the Shaver. Plugged the Reliant MDU w/Hand Controls into Port A of the Shaver. Plugged the programmed USB drive into the Shaver. Powered on the Shaver. Pressed the Continue button when the Check Serial Number Screen was displayed.  Monitor and EZView:  Started the Serial Monitoring and EZView capture.  Shaver:  Pressed the Start button to initiate the software upgrade. The software upgrade started. At the completion of the update the Shaver reported that the MDU was being reset and that Update was complete.  Monitor, EZView and Shaver:  Stopped the Serial Monitoring and EZView capture. Powered off the Shaver. Reviewed the Serial Monitoring and EZView capture data.  1 DII Software Update only looks for an MDU on Port A. Port B is not supported. | **P**  **F** | **N/A**1 |  |
|  |  |  |  |  |  |
|  | Verify the Software Update begins with Software Update Start Request. | The serial capture began with:  0x1B 🡪 FC 00 | **P**  **F** |  |  |
|  | Verify that the Reliant MDU ACKs the Software Update Start Request. | The response word to the Software Update Start Request (0x1B) was 0xFC00. Bit 10 is set indicating that the command was ACKed. | **P**  **F** |  |  |
|  | Verify that only Program Page Command Requests are after the Software Update Start command and before the Reset command. | The serial capture showed that after the Software Update Start Command (0x1B), only Program Page Command Requests (0x9C) appear before the Reset Request (0x2D). | **P**  **F** |  |  |
|  | Verify that the Reliant MDU Program Page Request is of the form: 0x9C, MS Page #, LS Page #, 128 Bytes of Data and CRC Byte. | The format of every Program Page Command Request was the following:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Program Page (0x9C) | MS Byte of 128 byte Page # | LS Byte of 128 byte Page # | 128 Bytes of Flash Data | CRC Byte | | **P**  **F** |  |  |
|  | Verify that the Reliant MDU ACKs the Program Page Command Requests. | The response word to every Program Page Command Request (0x9C) was 0xFC00. Bit 10 is set indicating that the command was ACKed. | **P**  **F** |  |  |
|  |  |  |  |  |  |
|  | Verify that the total size of Program Page Requests is 132 bytes. | The total number of bytes of the Program Page Command Requests was 132. | **P**  **F** |  |  |
|  | Verify that the Program Page request receives a response within 400ms after the request was sent. | Reviewed the EZView capture to verify that the Reliant MDU responded to each of the Program Page Command Requests 0x9C within 400 ms. | **P**  **F** |  |  |
|  | Verify that the last command of the Software Update was the Reset Command Request. | The last command of the Software Update was the Reset Command Request (0x2D). | **P**  **F** |  |  |
|  | Verify that the Reliant MDU ACKs the Reset Command Request. | The response word to the Reset Command Request (0x2D) was 0xFC00. Bit 10 is set indicating that the command was request was ACKed. | **P**  **F** |  |  |
| DYONICS II 2.2.2  2.2.3  2.2.4 |  | Monitor:  Hooked up the pins 1 & 14 of J1 – Port A or J6 – Port B on the EIP Board to pin - & + of the converter. Started the Serial Monitoring capture.  Shaver and Monitor:  Plugged a Reliant MDU w/Hand Controls in Port A of the Shaver.  Modified the shaver code software by inserting the following code segment:  {  SnByte pbBuf[134];  SnWord wRequest;  SnByte bCnt;  wRequest = HAND\_PORT\_CMD(PORTA, 0x1B);  SendSerialRequests(1, &wRequest, 4);  \*(SnWord\*)pbBuf = HAND\_PORT\_CMD(PORTA, SERIAL\_CMD\_REQ\_12);  pbBuf[2] = SERIAL\_CMD\_REQ\_12;  pbBuf[3] = 0;  pbBuf[4] = 128;  for (bCnt = 0; bCnt < 128; bCnt++) {  pbBuf[bCnt + 5] = bCnt;  }  pbBuf[128 + 5] = CrcMemChunk(&pbBuf[2], 128 + 3, 0);  m\_hDriver->SerialPageToDevice(134, pbBuf);  pbBuf[128 + 5] = 0x00;  m\_hDriver->SerialPageToDevice(134, pbBuf);  wRequest = HAND\_PORT\_CMD(PORTA, 0x2D);  SendSerialRequests(1, &wRequest, 4);  Sleep(4000);  wRequest = HAND\_PORT\_CMD(PORTA, 0x00);  SendSerialRequests(1, &wRequest, 4);  }  Right before the return TRUE; line of CControl::Init() in Control.cpp.  Set a break point at return TRUE; line of CControl::Init() in Control.cpp. Started the Shaver Application and waited till it stops at the breakpoint.  Monitor:  Stopped the capture.  1 DII Software Update only looks for an MDU on Port A. Port B is not supported. |  | **N/A**1 |  |
|  | Verify that response to first Program Page request is an ACK. | The Program Page request had a calculated CRC from the Shaver of 0x98, which got a response word of 0xFC00. Bit 10 is set indicating that the command was request got an ACK. | **P**  **F** |  |  |
|  | Verify that response to second Program Page request (with the invalid CRC of 0x00) is a NAK. | The Program Page request had a CRC of 0x00, which got a response word of 0x0000. Bit 10 is clear indicating that the command request got a NAK. | **P**  **F** |  |  |
|  | Verify that a Software Version Command Request sent 4 seconds after the Reset Command Request receives an ACK. | The response word to the Software Version Command Request (0x00) sent after the Reset Command Request (0x2D) was 0xFC00. Bit 10 is set indicating that the command was request was ACKed. | **P**  **F** |  |  |

# SUMMARY

## Notes

## Overall Pass/Fail Status

|  |  |
| --- | --- |
| **Overall Pass / Fail Status** |  |
| **Date** |  |
| **Signature** |  |
| **Printed Name** |  |
| **Department** |  |
| **Title** |  |