Byte and Message Timings

8 July 2008

Eric Hamilton

Contents

List of Figures

List of Tables

[1] `$Header: d:/Binder2/ByteTime/RCS/ByteTime.tex,v 1.6 2008-07-08 12:28:48-07 Hamilton Exp Hamilton $`

1 Message Durations

1.1 How to use the tables

A serial byte of data consists of several parts:

1. The electrical format of the data is what is called "Non-Return to Zero" or NRZ. This means that there is no return to a base value in between each bit time. Thus the only way that an end of bit may be detected is to maintain accurate and careful timing during the decode of the bits in a data byte. (In the three figures that accompany this note, there are arrows to show where each bit ends and starts. On a "real" oscilloscope there are no helpful arrows.)

2. Each byte starts out with a "start" bit. The duration of a start bit is $\frac{1}{baudrate}$ when baud rate is defined as number of equal length bits per second.

3. Following the start bit are a variable number of data bits. Most protocols used at Pelco use eight "data" bits per byte. Other protocols use as few as five and as many as twelve. (Other lengths are legal.)

4. After the data bits there may, or may not, be a "parity" bit. If a parity bit is included in the byte, then it has four different values:

   (a) Odd, this bit will cause all "1" bits in the byte to be have an odd count.

   (b) Even, this bit will cause all "1" bits in the byte to be have an even count.

   (c) Mark, in this case the bit will always be "on". Some systems will define "on" to be high and others will define it to be low.

   (d) Space, in this case the bit will always be "off". Some systems will define "off" to be low and others will define it to be high.

   Note that this means that there are five types of parity when "none" is included.

5. At the end of each byte there will be a "stop" bit(s). At Pelco one stop bit is common. Other systems use many different durations which range from 1.00 bit times to over 2.00 bit times. At one time 1.42 was common with older Tele-Type machines. (One of the UARTs in use at Pelco offers the following stop bit durations: 0.563, 0.625, 0.688, 0.750, 0.813, 0.875, 0.938, 1.000, 1.563, 1.625, 1.688, 1.750, 1.813, 1.875, 1.938 and 2.00. Most UARTs only offer 1.0, 1.5 and 2.0 for stop bit durations.)

6. The contents of each table are as follows:

   (a) All timing values in the tables are in milliseconds (ms). I.e $\frac{1}{1000}^{th}$s of a second.

   (b) Full byte timings are given for 10, 11 and 12 bit bytes. As was outlined above these are the three most common byte lengths used at Pelco. 10 bit bytes are normally used with D Protocol for PTZ control. When parity is added in, then the byte length in bits will become 11 bits long. If two stop bits are specified, then the two preceding values become 11 and 12 respectively.

   (c) The duration of a single bit is not specified, but rather the value for a full 10 bit byte may be easily divided by 10 to get a bit's duration. (This was part of an effort to save paper.)

   (d) Full message timings are given for all message lengths up to eight byte messages. At Pelco messages lengths vary from 3 → 18 and sometimes longer.

7. The figures that accompany this short note were all taken of commands being sent in RS-422 electrical format.

8. In RS-422 format, there are two signals, a "+" and a "-" version of the data. The receiver examines the input data and reports out a "1" or "0" based on the difference in the two signals.

9. On the accompanying figures the upper trace, trace 1, is the "+" signal and the lower trace, trace 2, is the "-" signal.

[2] `$Header: d:/Binder2/ByteTime/RCS/ByteTime.inc,v 1.9 2008-07-08 12:23:23-07 Hamilton Exp Hamilton $`

10. The receiver has a decision threshold of 200 mV. I.e. any signal that is greater than 200 mV different is reported out as a "1" while ones that are less are reported out as "0" logical value.
11. It should be noted that when "nothing" is being sent, that the quiescent value is an asserted high, for the "+" signal and a low for the "-" signal.
12. When a command starts, a start bit is sent that changes the value on the line to the "active" condition (and the receiver will output a logical "1") and then the rest of the character follows.
13. When a character ends, a stop bit (and the receiver will output a logical "0") is sent. It just so happens that the stop bit has the same value as the line does when it is in quiescent state. I.e. when nothing is being sent, the system always is sending stop bits.
14. Sometimes there are small timing differences between the receiver and the transmitter. Since infinite stop bits are interpreted the same as a single stop bit, to get the most reliable possible transmission of data, it is best to send data with two stop bits and receive data with one stop bit. That way if there are any timing differences, there is a built in "safety factor" of about ±10% of the baud rate. No Pelco equipment implements this logic. The only down side to implementing this logic is that messages take about 10% longer to send. At the normal rate of sending message, i.e. when ever the operator fools around with a joy stick, the additional 10% is insignificant.

1.2 The Data Pictures

A Byte without Parity As shown in Figure 1, page 6, a byte sent at 9600 baud consists of one start bit, eight data bits (which are all "1"s), no parity bit and one stop bit (10 total bits) is shown. Note that the time from the start of the byte to the end of the byte is about 1.040 ms. (A digital oscilloscope was used to make the capture and it quantizes the data so that values shown in the readout are not exact, even though they may appear to be so.)

Bytes with Parity In Figure 2, page 7 an identical byte is shown, except that this time the byte has an "odd" parity bit. Likewise in Figure 3, page 8 a byte of all data "1"s is shown that has even parity.

A Message with No Parity In Figure 4, page 9 a seven byte D Protocol command without parity is shown.

Messages with Parity In Figure 5, page 10 a seven byte D Protocol command with even parity is shown. While in Figure 6, page 11 a D Protocol command with even parity is shown.

RS-485 Byte In Figure 7, page 12 a typical RS-485 byte is shown. Note the time to acquire the data line and the release of it after the byte has been sent. Note also that the sending unit is sending two stop bits. (The character being sent is an `ACK` (`0x06`) byte.) The character was sent by a Panasonic system running at 9600 baud.

RS-485 Message In Figure 8, page 13 a typical RS-485 message is shown. Note that the handling of the data line is the same as in the previous example. I.e. the line is seized, used and the released.

1.3 The Byte Duration Tables

This is a collection of the predicted durations of bytes and messages for various baud rates. The format of all messages is 1 start bit, 8 data bits, $0 \rightarrow 2$ parity bits, 1 stop bit. Giving byte lengths of 10, 11 and 12 bits. Durations, and start/end times, of each byte in a D Protocol and P Protocol messages are listed.

300 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	33.333	36.667	40.000
2	66.667	73.333	80.000
3	100.000	110.000	120.000
4	133.333	146.667	160.000
5	166.667	183.333	200.000
6	200.000	220.000	240.000
7	233.333	256.667	280.000
8	266.667	293.333	320.000

Table 1: Timings for data sent at 300 baud

600 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	16.667	18.333	20.000
2	33.333	36.667	40.000
3	50.000	55.000	60.000
4	66.667	73.333	80.000
5	83.333	91.667	100.000
6	100.000	110.000	120.000
7	116.667	128.333	140.000
8	133.333	146.667	160.000

Table 2: Timings for data sent at 600 baud

1200 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	8.333	9.167	10.000
2	16.667	18.333	20.000
3	25.000	27.500	30.000
4	33.333	36.667	40.000
5	41.667	45.833	50.000
6	50.000	55.000	60.000
7	58.333	64.167	70.000
8	66.667	73.333	80.000

Table 3: Timings for data sent at 1200 baud

2400 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	4.167	4.583	5.000
2	8.333	9.167	10.000
3	12.500	13.750	15.000
4	16.667	18.333	20.000
5	20.833	22.917	25.000
6	25.000	27.500	30.000
7	29.167	32.083	35.000
8	33.333	36.667	40.000

Table 4: Timings for data sent at 2400 baud

4800 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	2.083	2.292	2.500
2	4.167	4.583	5.000
3	6.250	6.875	7.500
4	8.333	9.167	10.000
5	10.417	11.458	12.500
6	12.500	13.750	15.000
7	14.583	16.042	17.500
8	16.667	18.333	20.000

Table 5: Timings for data sent at 4800 baud

9600 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	1.042	1.146	1.250
2	2.083	2.292	2.500
3	3.125	3.438	3.750
4	4.167	4.583	5.000
5	5.208	5.729	6.250
6	6.250	6.875	7.500
7	7.292	8.021	8.750
8	8.333	9.167	10.000

Table 6: Timings for data sent at 9600 baud

14400 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	0.694	0.764	0.833
2	1.389	1.528	1.667
3	2.083	2.292	2.500
4	2.778	3.056	3.333
5	3.472	3.819	4.167
6	4.167	4.583	5.000
7	4.861	5.347	5.833
8	5.556	6.111	6.667

Table 7: Timings for data sent at 14400 baud

19200 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	0.521	0.573	0.625
2	1.042	1.146	1.250
3	1.563	1.719	1.875
4	2.083	2.292	2.500
5	2.604	2.865	3.125
6	3.125	3.438	3.750
7	3.646	4.010	4.375
8	4.167	4.583	5.000

Table 8: Timings for data sent at 19200 baud

28800 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	0.347	0.382	0.417
2	0.694	0.764	0.833
3	1.042	1.146	1.250
4	1.389	1.528	1.667
5	1.736	1.910	2.083
6	2.083	2.292	2.500
7	2.431	2.674	2.917
8	2.778	3.056	3.333

Table 9: Timings for data sent at 28800 baud

38400 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	0.260	0.286	0.313
2	0.521	0.573	0.625
3	0.781	0.859	0.938
4	1.042	1.146	1.250
5	1.302	1.432	1.563
6	1.563	1.719	1.875
7	1.823	2.005	2.188
8	2.083	2.292	2.500

Table 10: Timings for data sent at 38400 baud

57600 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	0.174	0.191	0.208
2	0.347	0.382	0.417
3	0.521	0.573	0.625
4	0.694	0.764	0.833
5	0.868	0.955	1.042
6	1.042	1.146	1.250
7	1.215	1.337	1.458
8	1.389	1.528	1.667

Table 11: Timings for data sent at 57600 baud

115200 baud	10 bit bytes	11 bit bytes	12 bit bytes
1	0.087	0.095	0.104
2	0.174	0.191	0.208
3	0.260	0.286	0.313
4	0.347	0.382	0.417
5	0.434	0.477	0.521
6	0.521	0.573	0.625
7	0.608	0.668	0.729
8	0.694	0.764	0.833

Table 12: Timings for data sent at 115200 baud

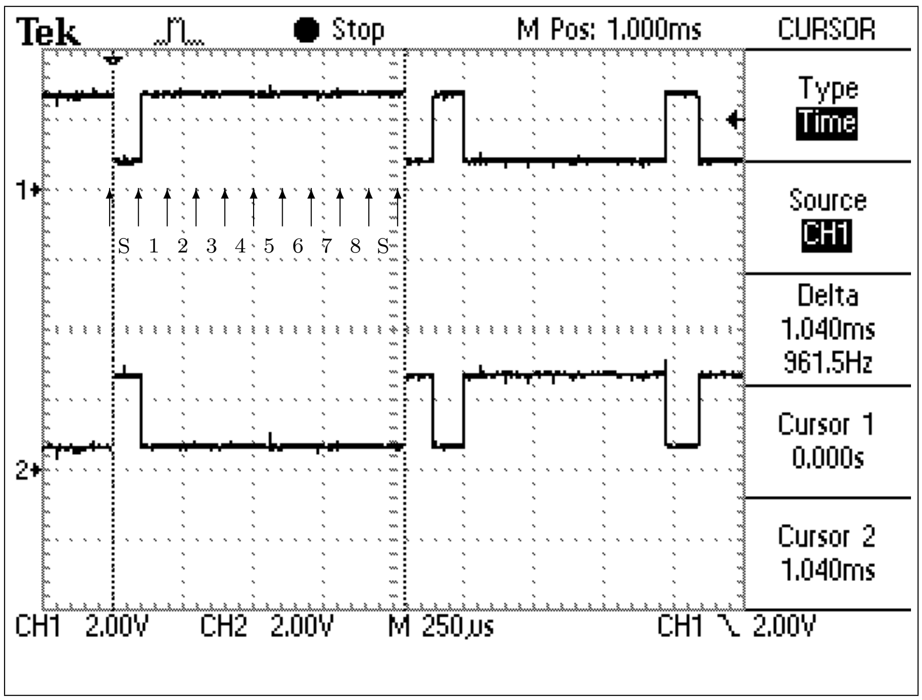

Figure 1: 1 Start, 8 data bits of 1, No parity, 1 Stop ByteTime004.ps

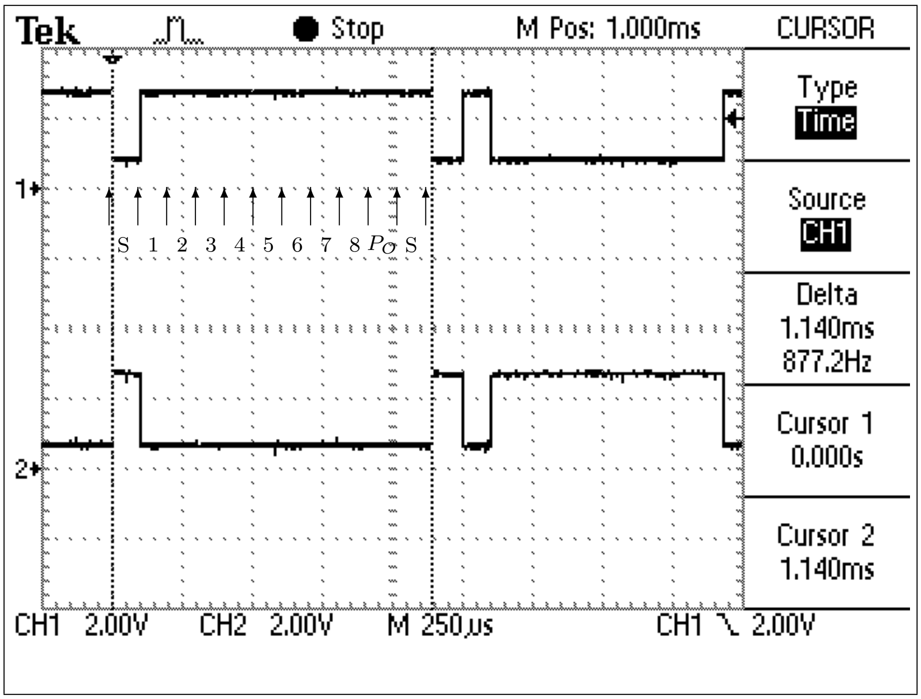

Figure 2: 1 Start, 8 data bits of 1, Odd parity, 1 Stop ByteTime001.ps

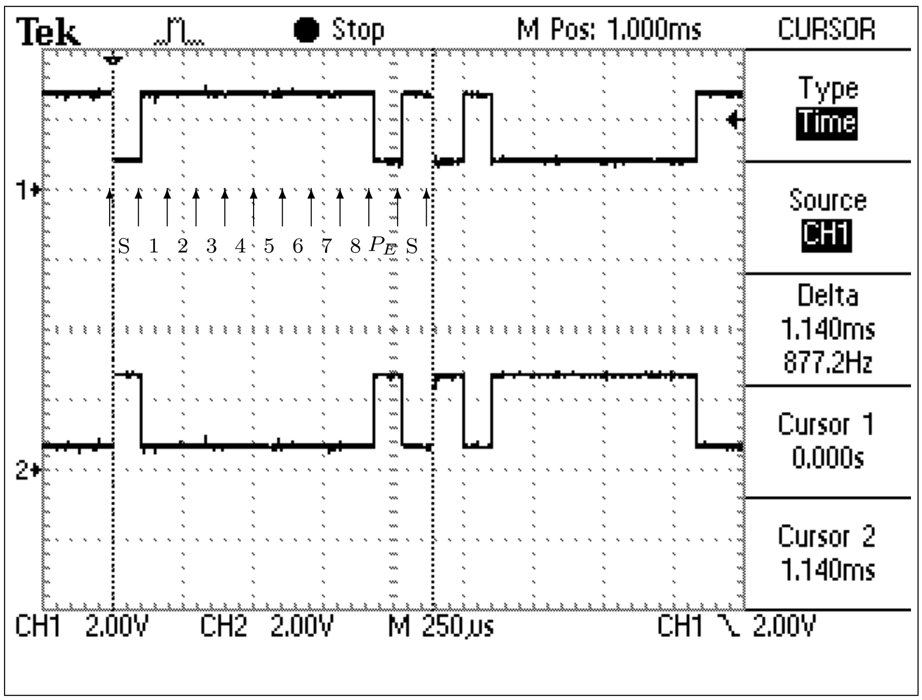

Figure 3: 1 Start, 8 data bits of 1, Even parity, 1 Stop ByteTime002.ps

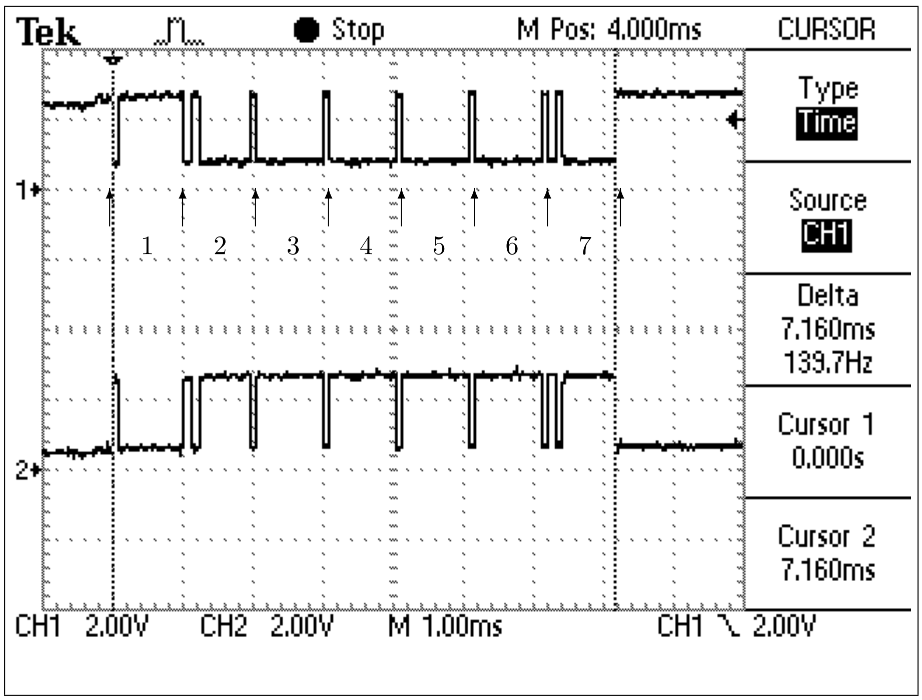

Figure 4: Full message of 1 Start, 8 data, No parity, 1 Stop ByteTime009.ps

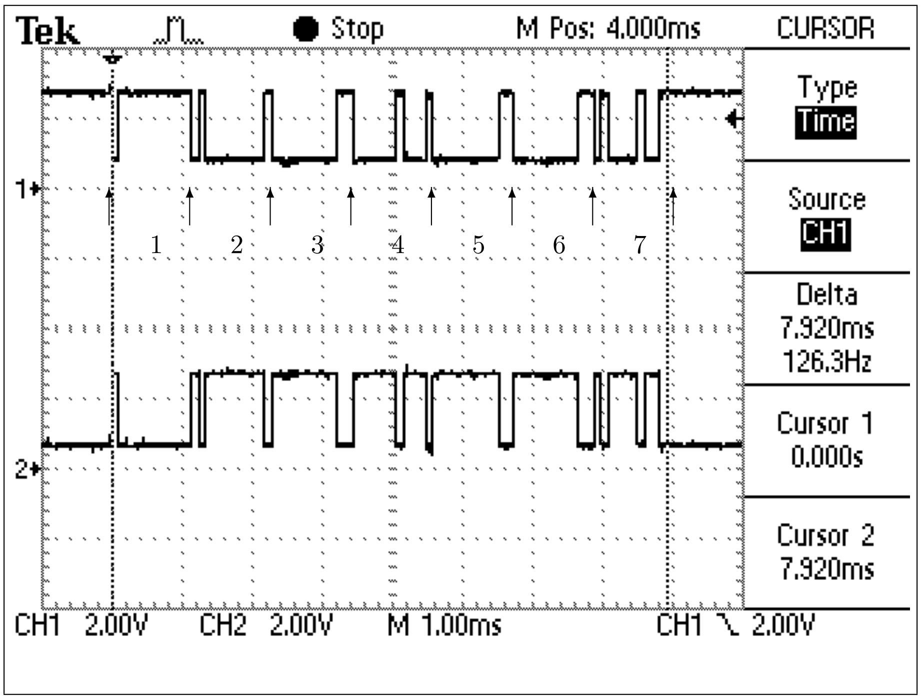

Figure 5: Full message of 1 Start, 8 data, Odd parity, 1 Stop ByteTime007.ps

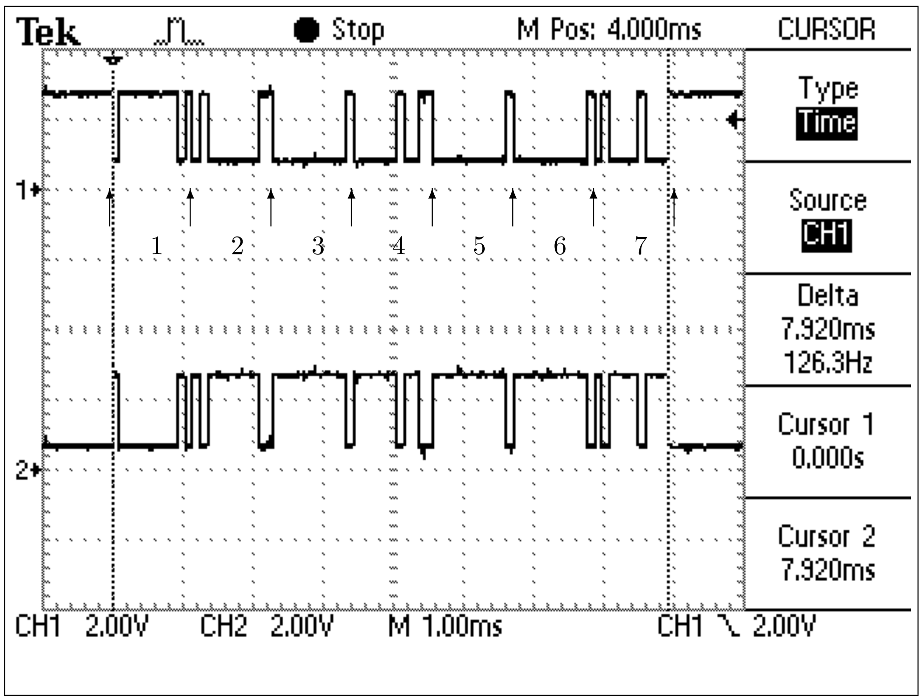

Figure 6: Full message of 1 Start, 8 data, Even parity, 1 Stop ByteTime006.ps

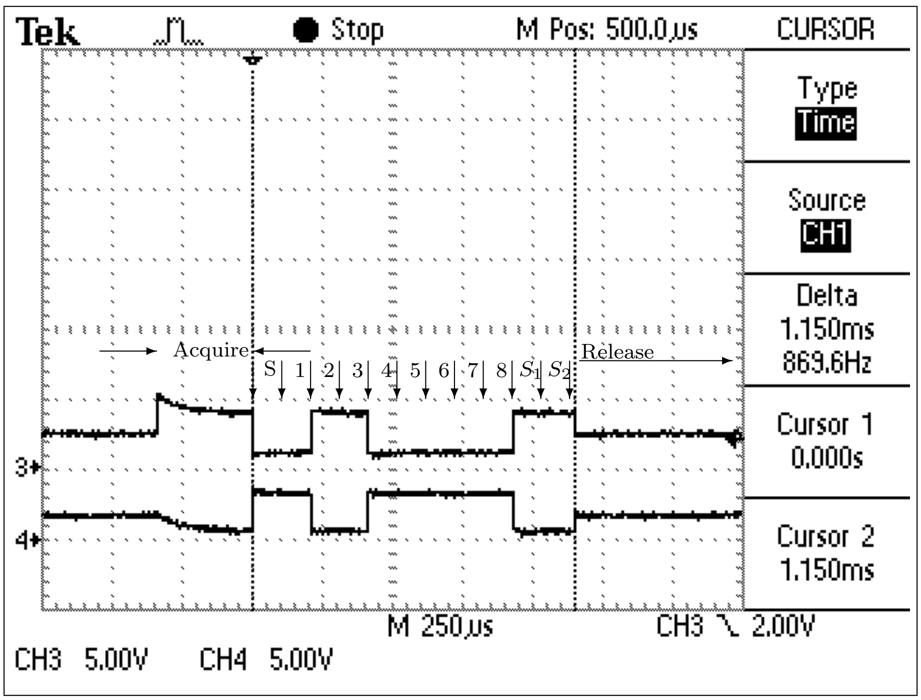

Figure 7: Single byte of RS-485 data, 1 Start, 8 data, 2 Stops 23apr07a009.ps

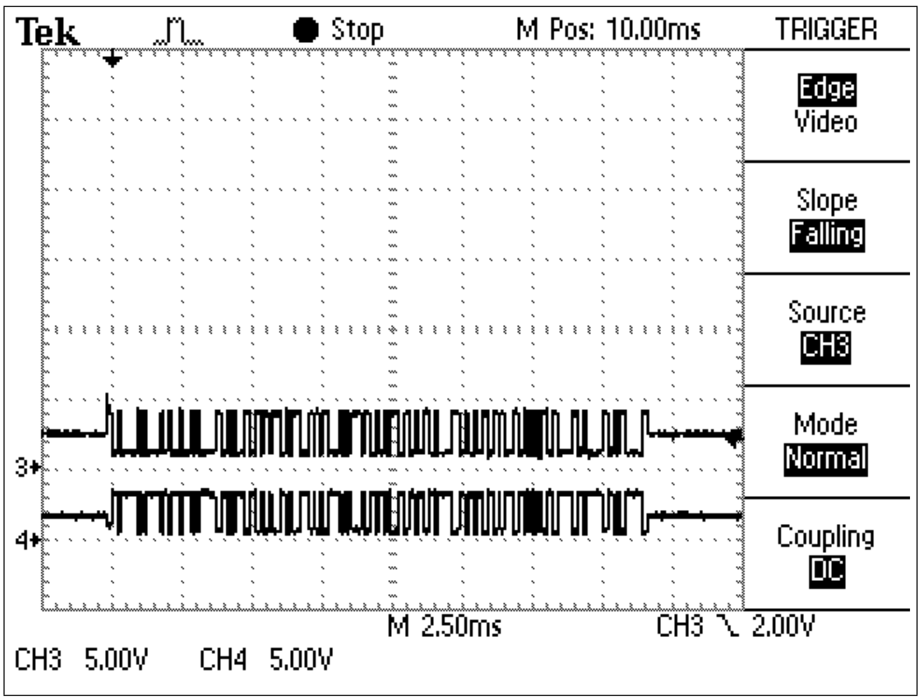

Figure 8: Full 18 Byte message RS-485 data 23apr07a011.ps

Index