

Untersuchungen zu Dallas DS1820 Temperatursensoren

Inhaltsverzeichnis

Abbildungsverzeichnis	2
Literaturverzeichnis	Fehler! Textmarke nicht definiert.
Buslast bei Anschluss mehrerer Dallas One Wire Sensoren	4
Referenzen:	Fehler! Textmarke nicht definiert.
Summary	5
Massnahmen (SW):	7
Massnahmen (HW):	8
Vergleich 4,7k Pullup mit anderen Werten	10
Defekter Sensor ?	14
Historie	16

Abbildungsverzeichnis

Abbildung 1: 1 Busteilnehmer, Pullup 4.7K	5
Abbildung 2: 2 Busteilnehmer, Pullup 4.7K	5
Abbildung 3: 3 Busteilnehmer, Pullup 4.7K	5
Abbildung 4: 4 Busteilnehmer, Pullup 4.7K	6
Abbildung 5: 4 Busteilnehmer, Pullup 1.93K	6
Abbildung 6: 4.7k Pullup, vorige Softwareversion	8
Abbildung 7: 1.9k Pullup, vorige Softwareversion	8
Abbildung 8: 4.7k Pullup, incl. SW modification	8
Abbildung 9: 1.9k Pullup, incl. SW modification	9
Abbildung 10: 4,7k 10k	10
Abbildung 11: 4,7k 6,8k	10
Abbildung 12: 4,7k 5,6k	11
Abbildung 13: 4,7k 4,7k	11
Abbildung 14: Einbau eines 4,7k Widerstandes in den Verteiler	12
Abbildung 15: Aufbau mit 8 Temperatursensoren	13
Abbildung 16: Normaler Sensor	14
Abbildung 17: Defekter Sensor mit 4.7k Pullup	14
Abbildung 18: Defekter Sensor mit 1.9K Pullup	14
Abbildung 19: Defekter Sensor mit 4.7k Pullup einschliesslich 162ms Timeout	15

Literaturverzeichnis

Dallas Application Note 74, Reading and Writing iButtons via Serial Interfaces, AN74.PDF)
Dallas DS18S20 High-Precision 1-Wire Digital Thermometer

Buslast bei Anschluss mehrerer Dallas One Wire Sensoren

Es wurde das Problem beobachtet das bei einem Anschluss von mehr als 3 Dallassensoren (hier: konfektionierte Dallas Temperaturfühler von CONRAD Elektronik mit 10m Anschlusskabel) die Datenerfassung nicht mehr möglich ist. Die entsprechenden ROM codes werden ab 4 Sensoren des o.g. Typs nicht mehr korrekt angezeigt. Ein verlässlicher Betrieb daher nicht mehr möglich.

Ursache ist die hohe kapazitive Belastung des One Wire Bus durch zu hohe Kabellängen und möglicherweise ungeeignete / überflüssige Abschirmungen, welche die kapazitive Last zusätzlich erhöhen.

Die entsprechende Empfehlung wäre auf den o.g. Typ (10m) weitestgehend zu verzichten, bzw. nur einen, max. 2 davon zu verwenden. Für Temperaturmessungen die keinen Tauchfühler erfordern, empfiehlt es sich diesen Fühler selber anzufertigen. (z.B. Reichelt Elektronik, einzel Bauteil: DS1820, dieser kann dann an ein vorkonfektioniertes CAT5 LAN Kabel mit RJ 45 Stecker gelötet werden. Praktisch nimmt man ein Kabel der erforderlichen Länge und schneidet einen der beiden RJ45 Stecker ab. Dort wird der Sensor dann angelötet. Dabei sollte drauf geachtet werden nur die 3 Leitungen des Sensors anzuschließen sind. Alle weiteren nicht benötigten Leitungen des CAT 5 Kabels sollten nicht angeschlossen werden um die o.g. Leitungskapazität nicht unnötig zu erhöhen. Gegeben falls auch die entsprechenden Anschlüsse am RJ45 Stecker entfernen.

Nachfolgend einige Untersuchungen mit entsprechenden Lösungsansätzen.

Im Anhang noch die Untersuchung eines defekten Sensors

Summary

Bustiming bei einem WRITE 1 Slot:

Der Controller legt den Bus für etwa $8.4\mu\text{s}$ auf Low Pegel, nimmt danach das Signal weg, der Pullup zieht den Bus wieder hoch. Der Dallas Sensor sampelt den Bus nach $15\mu\text{s}$, gemessen ab der fallenden Flanke.

Ist der Pegel zu diesem Zeitpunkt $> 2,2\text{V}$, interpretiert der Sensor das als logische 1, sonst 0.

Man erkennt, je mehr kapazitive Last (Teilnehmer) am Bus hängen, umso langsamer schwingt der Bus zum High Pegel zurück. Der Sensor muss nach $15\mu\text{s}$ eine Spannung von mind. 2.2V erkannt haben um eine logische 1 zu erkennen, andernfalls wird eine 0 erkannt.

Im letzten Bild wurde der Pullup von $4.7\text{k}\Omega$ durch Parallelschalten eines $3.3\text{k}\Omega$ auf etwa $1.93\text{k}\Omega$ verringert. Das Timing verhält sich damit in etwa der Buslast von 1 bis 2 Sensoren mit $4.7\text{k}\Omega$ Pullup

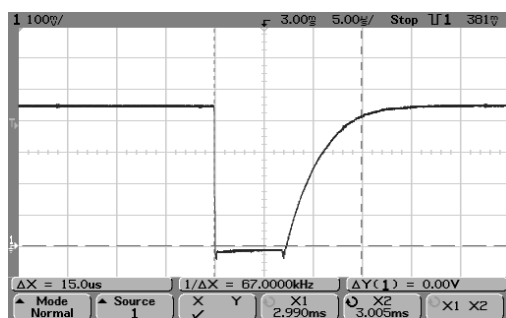


Abbildung 1: 1 Busteilnehmer, Pullup 4.7K

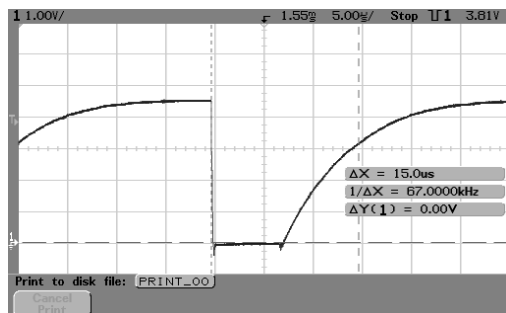


Abbildung 2: 2 Busteilnehmer, Pullup 4.7K

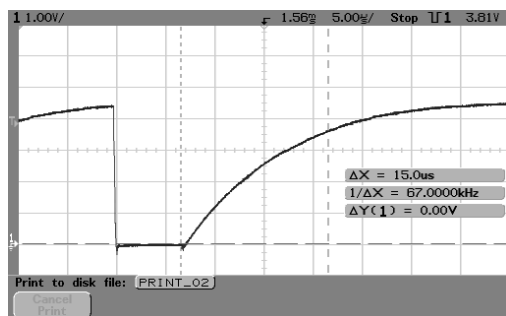


Abbildung 3: 3 Busteilnehmer, Pullup 4.7K

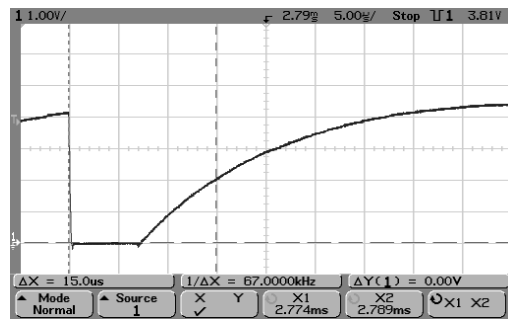


Abbildung 4: 4 Busteilnehmer, Pullup 4.7K

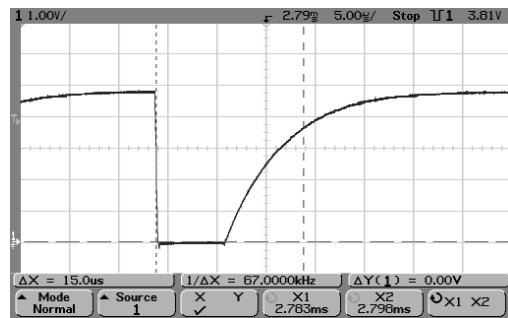


Abbildung 5: 4 Busteilnehmer, Pullup 1.93K

Massnahmen (SW):

Überlicherweise wird für die bitweise Kommunikation mit dem One Wire Bus ein UART eingesetzt.

Der Controller muss dafür sorgen das bei hoher Busbelastung der Bus genügend Zeit hat in den Idle Mode zurückzukehren bevor das nächste Bit gesendet wird. Ansonsten erkennt der Sensor die ansteigende Flanke nicht früh genug, bevor das nächste „Low“ Signal gesendet wird.

Das wird erreicht in dem ein weiteres Stoppbit hinzugefügt wird (UART Setting damit : 8, N, 2).

Bei einer Baudrate von 115.2kbit ergeben das Startbit + 7 (Null) Datenbits eine low Phase von $69\mu\text{s}$ für ein WRITE „0“ Bit. Vorgeschrieben ist eine mindest-low Time von $60\mu\text{s}$. was damit sicher erreicht wird.

Zusätzlich wird im Falle eines WRITE 0 Commands das Bit 7 bei dieser Übertragung ebenfalls auf 1 gesetzt, sodass sich zusammen mit den beiden 2 Stoppbits eine High Time von $26\mu\text{s}$ ergibt. Das ist zunächst eine ausreichende Zeit damit der Bus auch bei stärkerer Belastung sicher in den Idle (High) State zurückkehren kann.

Beim Empfang von Daten wird geprüft ob das empfangene Signal für 2 Bitperioden 0 war.

das empfangene Byte wird maskiert und die unteren beiden Bits ausgewertet. Sind beide 0 wird dieses bei einem READ 0 Commando als 0 interpretiert. Das Ganze auf drei Bits auszuweiten ist etwas kritisch, da es möglicherweise über die spezifizierte Low-Bitphase des Sensors hinausgeht.

Massnahmen (HW):

Die Einführung eines niederohmigeren Pullups bringt eine sofortige Verbesserung.

Im Versuch wurde ein Pullup von 1.93k (4.7k parallel mit 3.3k).

Der Widerstand könnte im ISDN Adapter angebracht werden (als Parallelwiderstand zwischen Vcc und Data, oder auf der Steuerplatine im Austausch für den eingebauten 4.7k Widerstand).

Nachfolgend 4 Screenshots der ersten beiden Bits eines MATCHROM Commands (0x55) mit 5 Busteilnehmern.

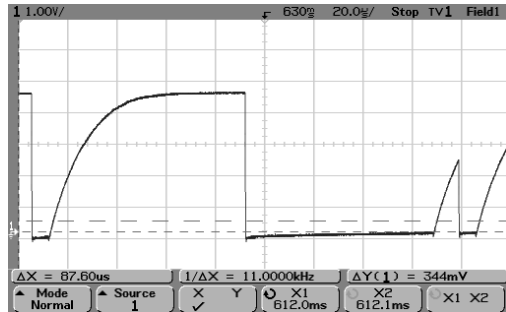


Abbildung 6: 4.7k Pullup, vorige Softwareversion

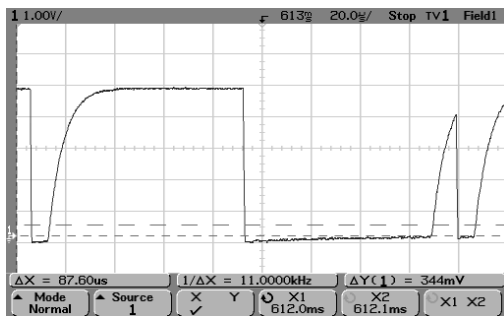


Abbildung 7: 1.9k Pullup, vorige Softwareversion

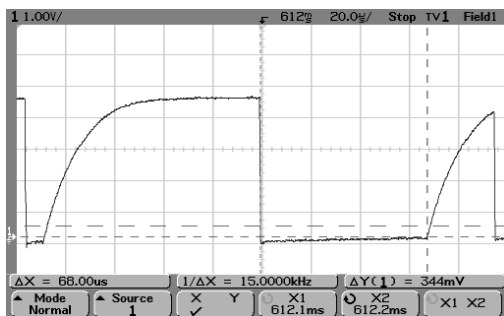


Abbildung 8: 4.7k Pullup, incl. SW modification

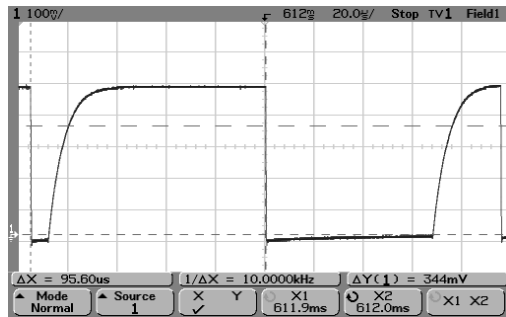


Abbildung 9: 1.9k Pullup, incl. SW modification

Erkennbar ist die schwache Idle Periode wenn der Bus nach einem WRITE 0 Commando wieder freigegeben wird und danach sofort ein neues WRITE x Commando folgt. (1. Bild oben links)

Vergleich 4,7k Pullup mit anderen Werten

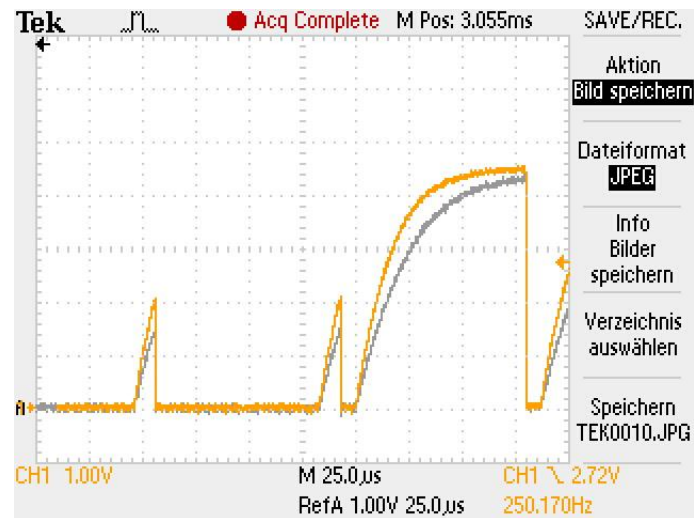


Abbildung 10: 4,7k || 10k

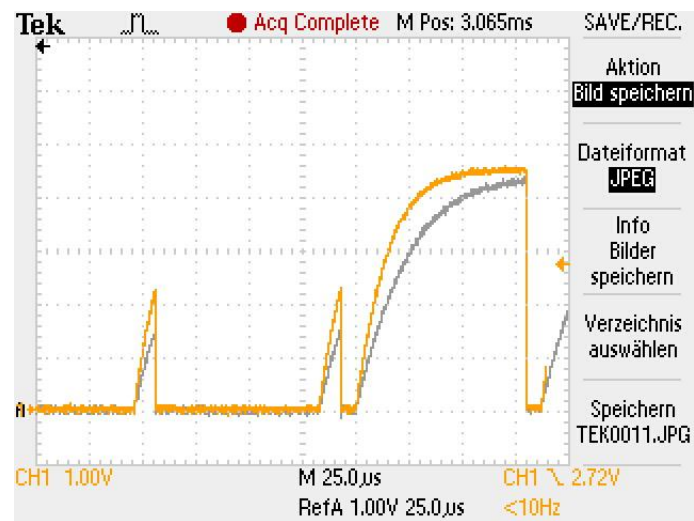


Abbildung 11: 4,7k || 6,8k

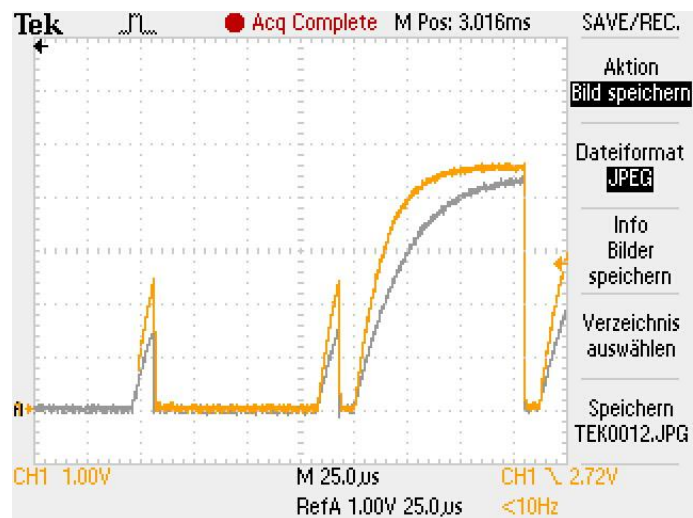


Abbildung 12: 4,7k || 5,6k

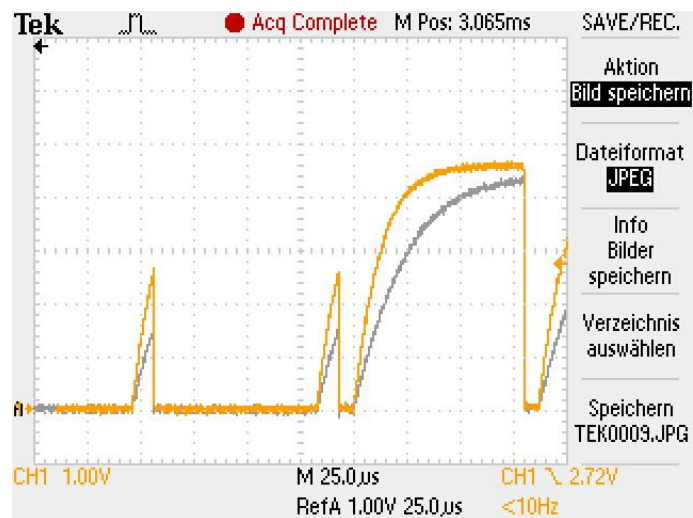


Abbildung 13: 4,7k || 4,7k

Ab 5,6k scheint der Aufbau mit 8 Temperatursensoren (insgesamt ca 65m Kabel) ohne einschalten des High Bus Load Schalters in der Software zu funktionieren

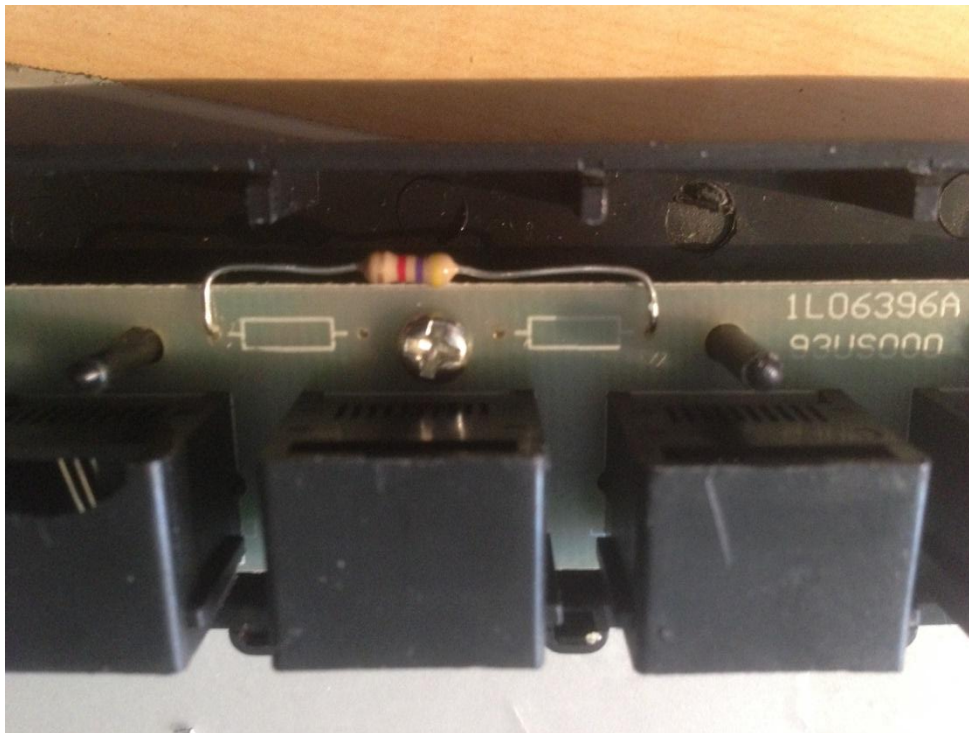


Abbildung 14: Einbau eines 4,7k Widerstandes in den Verteiler

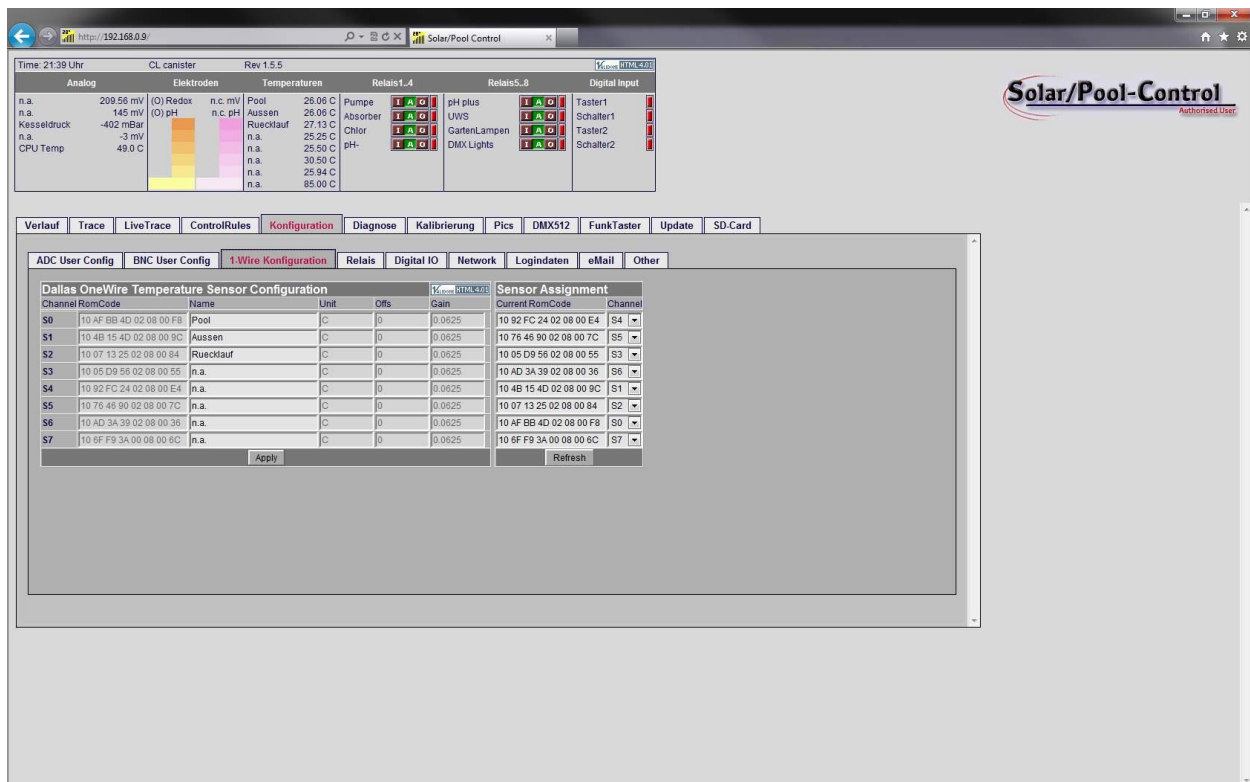


Abbildung 15: Aufbau mit 8 Temperatursensoren

Defekter Sensor ?

Bei den Untersuchungen mit den Sensoren eines Kunden ist ein Sensor auffällig geworden.

Dieser Sensor zeigt permanent 85 Grad an (lt. Spec. ist dies der Resetwert eines Temperatursensors)

Wird dieser Sensor als einziger Teilnehmer an den Bus gesteckt, zieht der Sensor nach dem CONVERT T Broadcast Commando den Bus auf halben Level runter. Nach etwa 162ms wird der Bus wieder freigegeben. Möglicherweise ist der Sensor defekt oder intern falsch angeschlossen (parasitärer Mode ?)

Nachfolgend Screenshots von einem defekten und einem funktionierenden Sensor unmittelbar nach den Commandos: RESET Pulse - SKIP ROM und CONVERT T

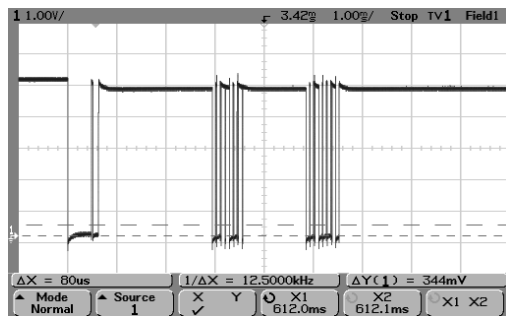


Abbildung 16: Normaler Sensor

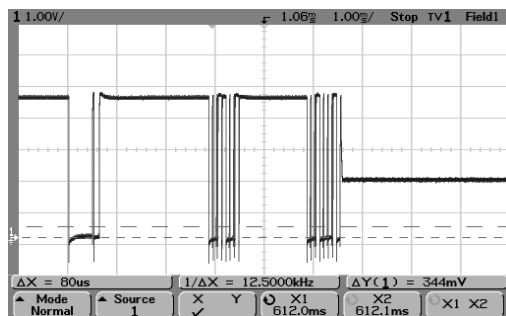


Abbildung 17: Defekter Sensor mit 4.7k Pullup

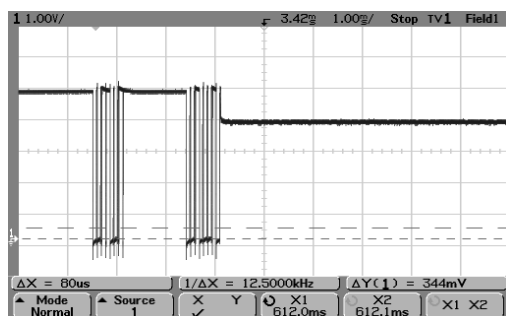


Abbildung 18: Defekter Sensor mit 1.9K Pullup

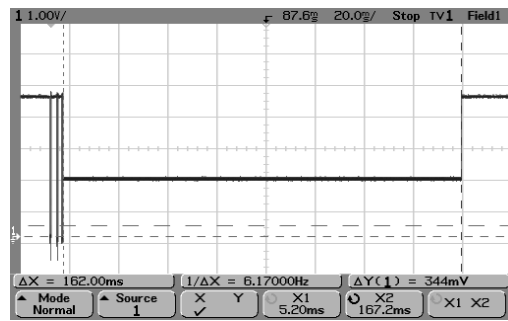


Abbildung 19: Defekter Sensor mit 4.7k Pullup einschliesslich 162ms Timeout

Historie

Datum	Name	Kommentar
06.12.2012	Frank Kiesow	Initial revision
12.07.2014	Manuel Follmann	Untersuchung mit unterschiedlichen Pullups