

SDI-12
A Serial-Digital Interface Standard
for
Microprocessor-Based Sensors
Version 1.1

July 7, 1994

Prepared By

NR Systems, Inc.
Logan, Utah
801-752-4200

Campbell Scientific, Inc.
Logan, Utah
801-753-2342

SDI-12 is not currently registered by any professional society, standards organization, or Government agency. Document copies, current document status, and other information may be obtained by contacting the SDI-12 Support Group at 135 East Center Street, Logan, Utah 84321, 801-752-4200, FAX 801-752-1691.

TABLE OF CONTENTS

1.0 INTRODUCTION	1
2.0 ADVANTAGES OF SDI-12	1
3.0 SDI-12 ELECTRICAL INTERFACE	2
3.1 Serial Data Line	3
3.1.1 Voltage Transitions	3
3.1.2 Impedance	4
3.2 Ground Line	4
3.2.1 Transient Protection	5
3.3 12-Volt Line	5
3.4 Connectors	5
4.0 SDI-12 COMMUNICATIONS PROTOCOL	6
4.1 Baud Rate and Byte Frame Format	6
4.2 Allowable Characters	7
4.3 Device Addresses	7
4.4 SDI-12 Commands and Responses	8
4.4.1 Acknowledge Active Command (a!)	9
4.4.1.1 Examples of the Acknowledge Active Command (a!)	9
4.4.2 Send Identification Command (aI!)	10
4.4.2.1 Example of the Send Identification Command	10
4.4.3 Start Measurement Command (aM!)	11
4.4.3.1 Aborting a Measurement	12
4.4.4 Service Request	13

4.4.5	Send Data Command (aD0! . . . aD9!)	14
4.4.5.1	Return of Multiple Measurements (Parameters) by a Sensor (D1! . . . D9!)	15
4.4.5.2	Examples of the Start Measurement Command (aM!) and the Send Data Command (aD!)	15
4.4.6	Additional Measurement Commands (aM1! . . . aM9!)	17
4.4.6.1	Examples of the Additional M Commands (aMn!)	17
4.4.7	Start Verification (aV!)	18
4.4.7.1	Examples of the Start Verification Command (aV!)	18
4.4.8	Extended Commands	19
4.4.8.1	Transparent Mode	19
5.0	SDI-12 Timing	20
5.1	Rules for the Break	21
5.2	Retries	22
APPENDICES		23
Appendix A:	Suggested SDI-12 Circuits	A-1
Appendix B:	Suggested SDI-12 Flow Control for SDI-12 Data Recorders and Sensors	B-1
Appendix C:	SDI-12 Glossary	C-1
Appendix D:	Pass/Fail Criteria for SDI-12 Data Recorders and SDI-12 Sensors	D-1

LIST OF TABLES

Table 1. Logic and Voltage Levels for Serial Data	3
Table 2. SDI-12 Byte Frame Format	6
Table 3. Printable Characters	7
Table 4. Sensor Address Codes	7
Table 5. The SDI-12 Basic Command/Response Set	8
Table 6. The Acknowledge Active Command (a!)	9
Table 7. The Send Identification Command (aI!)	10
Table 8. The Start Measurement Command (aM!)	11
Table 9. The Send Data Command (aD0!, aD1 . . . aD9!)	14

LIST OF FIGURES

Figure 1. The SDI-12 Bus 3
Figure 2. Impedance Circuit 4
Figure 3. SDI-12 Timing 20

SDI-12
A SERIAL-DIGITAL INTERFACE STANDARD
FOR MICROPROCESSOR-BASED SENSORS
SDI-12 Version 1.1

1.0 INTRODUCTION

This document describes Version 1.1 of the SDI-12 standard. This is a revision of the document: "SDI-12: A Serial Digital Interface Standard for Hydrologic and Environmental Sensors, Version 1.0" dated October, 1988. The purpose of this document is to describe SDI-12 in detail, clarify the SDI-12 timing requirements, and provide examples of all SDI-12 commands and responses. There are no technical differences between versions 1.0 and 1.1 of SDI-12. This document is labeled as version 1.1 to distinguish it from the first description of SDI-12, dated October 1988.

SDI-12 is a standard for interfacing data recorders with microprocessor-based sensors. SDI-12 stands for serial/digital interface at 1200 baud. This document describes the electrical interface, the communications protocol, and the timing requirements for SDI-12 data recorders and SDI-12 sensors.

SDI-12 is intended for applications with the following requirements:

- Battery powered operation with minimal current drain
- Low system cost
- Use of a single data recorder with multiple sensors on one cable
- Up to 200 feet of cable between a sensor and a data recorder

2.0 ADVANTAGES OF SDI-12

A serial-digital interface is a logical choice for interfacing microprocessor-based sensors with a data recorder. This has advantages for sensors and data recorders.

- Unique and complex self calibration algorithms can be done in microprocessor-based sensors.
- Sensors can be interchanged without reprogramming the data recorder with calibration or other information.

- Power is supplied to sensors through the interface.
- Hybrid circuit and surface mount technologies make it practical to include the power supply regulator, a microprocessor, and other needed circuitry in small sensor packages.
- Sensors can use low cost EEPROMs (electrically erasable programmable read only memory) for calibration coefficients and other information instead of internal trimming operations.
- The use of a standard serial interface eliminates significant complexity in the design of data recorders.
- Data recorders can be designed and produced independently of future sensor development.
- SDI-12 data recorders interface with a variety of sensors.
- SDI-12 sensors interface with a variety of data recorders.
- Personnel trained in SDI-12 will have skills to work with a variety of SDI-12 data recorders and SDI-12 sensors.

3.0 SDI-12 ELECTRICAL INTERFACE

The SDI-12 electrical interface uses the SDI-12 bus to transmit serial data between SDI-12 data recorders and sensors. The SDI-12 bus is the cable that connects multiple SDI-12 devices. This is a cable with three conductors:

- 1) a serial data line
- 2) a ground line
- 3) a 12-volt line

Figure 1 shows the SDI-12 bus connecting one data recorder with two sensors. The SDI-12 bus is capable of having 10 sensors connected to it.

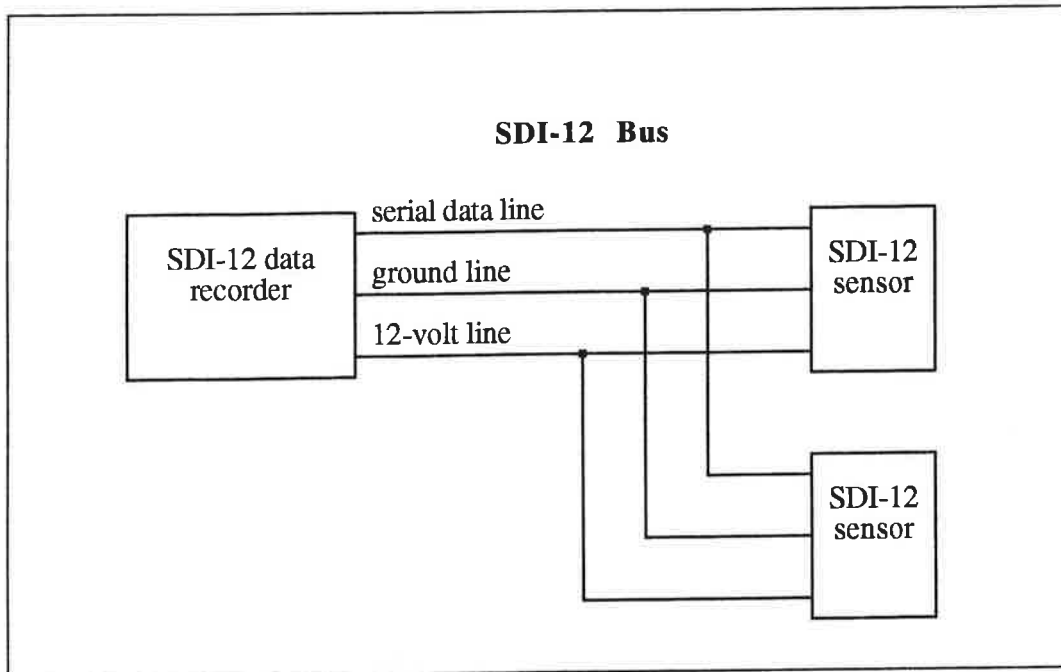


Figure 1. The SDI-12 Bus

3.1 Serial Data Line

The data line is a bidirectional, three-state, data transfer line. Table 1 shows the logic and voltage levels for the transmission of serial data for the SDI-12 standard. The data line uses negative logic.

Condition	Binary state	Voltage range
marking	1	-0.5 to 1.0 volts
spacing	0	3.5 to 5.5 volts
transition	undefined	1.0 to 3.5 volts

Table 1. Logic and voltage levels for serial data

3.1.1 Voltage Transitions

During normal operation, the data line voltage slew rate must not be greater than 1.5 volts per microsecond.

3.1.2 Impedance

When an SDI-12 device has its transmitter on, its direct current (DC) source resistance must be greater than 1000 ohms and less than 2000 ohms. When any SDI-12 device's transmitter is off, including during a low-power standby mode, the DC resistance to ground must be 200K. If an SDI-12 sensor does not use the 12-volt line for power, its data line resistance to ground while powered down must be 200K. Figure 2 shows an equivalent circuit.

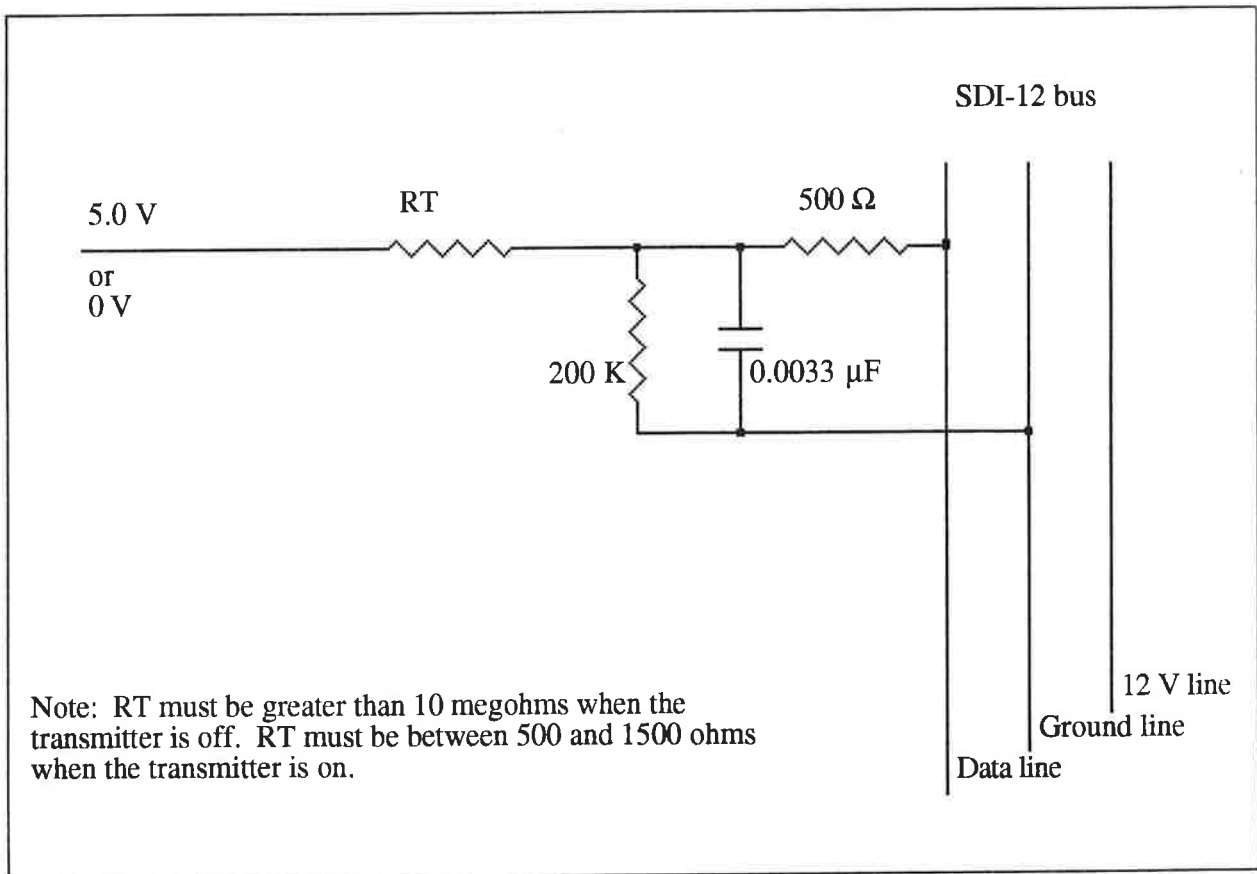


Figure 2. Equivalent circuit

3.2 Ground Line

The ground line must be connected to the circuit ground and the earth ground at the data recorder. The sensor circuit ground also must be connected to the ground line, but not normally to its own earth ground. If it is necessary to connect the sensor circuitry to earth ground, a heavy (12 AWG or larger) ground wire should be connected between the sensor earth ground and the data recorder earth ground for lightning protection.

The ground conductor should be large enough to keep the voltage drop between the data recorder and the sensor less than 0.5 volts during the maximum sensor current drain.

3.2.1 Transient Protection

Transient protection is recommended on the SDI-12 bus. See appendix A for a suggested SDI-12 transient protection method.

3.3 12 Volt-Line

The data recorder (or the external power supply) provides between 9.6 volts and 16 volts to the 12-volt line, with respect to ground, as measured under a maximum sensor load of 0.5 amperes. SDI-12 does not require the data recorder to be the source of power to the 12-volt line.

Sensors connected to the 12-volt line must not have inductive loads. SDI-12 does not require voltage limiting for transient protection in the sensor. Transient protection is, however, recommended. See appendix A for suggested circuits for transient protection.

3.4 Connectors

A connector type for SDI-12 is not specified.

4.0 SDI-12 COMMUNICATIONS PROTOCOL

SDI-12 data recorders and sensors communicate by an exchange of ASCII characters on the data line. The data recorder sends a break to wake up the sensors on the data line. A break is continuous spacing on the data line for at least 12 milliseconds. The data recorder then sends a command. The sensor, in turn, returns the appropriate response. Each command is for a specific sensor. The first character of each command is a unique sensor address that specifies with which sensor the recorder wants to communicate. Other sensors on the SDI-12 bus ignore the command and return to low-power standby mode. When a data recorder tells a sensor to start its measurement procedure, the recorder does not communicate with any other sensor until the data collection from the first sensor is complete.

A typical recorder/sensor measurement sequence proceeds as follows:

Step 1. The data recorder wakes all sensors on the SDI-12 bus with a break.

Step 2. The recorder transmits a command to a specific, addressed sensor, instructing it to make a measurement.

Step 3. The addressed sensor responds within 15.0 milliseconds returning the maximum time until the measurement data will be ready and the number of data values it will return.

Step 4. If the measurement is immediately available, the recorder transmits a command to the sensor instructing it to return the measurement(s). If the measurement is not ready, the data recorder waits for the sensor to send a request to the recorder, which indicates that the data are ready. The recorder then transmits a command to get the data.

Step 5. The sensor responds, returning one or more measurements.

4.1 Baud Rate and Byte Frame Format

The baud rate for SDI-12 is 1200. Table 2 shows the byte frame format for SDI-12.

1 start bit
7 data bits, least significant bit transmitted first
1 parity bit, even parity
1 stop bit

Table 2. SDI-12 byte frame format

4.2 Allowable Characters

All characters transmitted on the SDI-12 bus must be printable ASCII characters. Table 3 shows the printable characters.

space, 32 decimal
through
~, 126 decimal

Table 3. Printable characters

There is one exception. All responses from an SDI-12 sensor end with a carriage return (0D hex, 13 decimal) and a line feed (0A hex, 10 decimal) character. This carriage return/line feed sequence is shown as <CR><LF> in this document. Other printable characters can be used as addresses. They are not, however, defined in this specification.

4.3 Device Addresses

The first character of every command must be a sensor address. Likewise, the first character of a response is also the address character. This lets an SDI-12 recorder verify that the response has come from the correct sensor. (An address is a single character used to indicate which sensor is to respond to the command.) Table 4 shows the address codes.

ASCII address (a single character)	Decimal	Hex	Description
"0" (zero)	48	30	Default address, all sensors are initially set to "0" (zero) by the manufacturer for use in single sensor systems
"1" to "9"	49 to 57	31 to 39	Addresses for additional sensors on the SDI-12 bus

Table 4. Sensor address codes

4.4 SDI-12 Commands and Responses

Table 5 lists each basic SDI-12 command, its format, and the format of each response to a command. The terms in this table (a,ll,cccccccc,mmmmmm,vvv,xxx,<values>, etc.) are described in following sections 4.4.1 to 4.4.8.

Name	Command	Response
Break	Continuous spacing for at least 12 milliseconds	None
Acknowledge Active	a!	a<CR><LF>
Send Identification	a!	allccccccccmmmmmmvvvxxx...xx<CR><LF>
Start Measurement	aM!	attn<CR><LF>
(service request, from sensor)		a<CR><LF>
Send Data	aD0!	a<values><CR><LF>
	.	a<values><CR><LF>
	.	a<values><CR><LF>
	.	a<values><CR><LF>
	aD9!	a<values><CR><LF>
Additional Measurements	aM1!	attn<CR><LF>
	.	attn<CR><LF>
	.	attn<CR><LF>
	.	attn<CR><LF>
	aM9!	attn<CR><LF>
Start Verification	aV!	attn<CR><LF>
extended commands	Defined by the sensor manufacturer	Defined by the sensor manufacturer

Table 5. The SDI-12 basic command/response set

The first character of all commands and responses is always a device address. The last character of a command is the "!" character. The last two bytes of a response are a carriage return and line feed (<CR><LF>).

The maximum number of characters for a command and its response, not including the <CR> and <LF>, is 40. This allows a command and its response to fit on a 40-column display.

4.4.1 Acknowledge Active Command (a!)

This command is used to ensure that a sensor is responding to a data recorder or another SDI-12 device. It asks a sensor to acknowledge its presence on the SDI-12 bus. Table 6 shows the acknowledge active command.

Command	Response
a!	a<CR><LF>
a - the sensor address	a - the sensor address
! - terminates the command	<CR><LF> - terminates the response

Table 6. The acknowledge active command (a!)

4.4.1.1 Examples of the Acknowledge Active Command (a!)

0!0<CR><LF>

1!1<CR><LF>

2!2<CR><LF>

4.4.2 Send Identification Command (aI!)

This command is used to query sensors for their SDI-12 compatibility level, model number, and firmware version number. Table 7 shows the send identification command.

Command	Response
aI!	allccccccmmmmmmvvvxxx . . . xxx<CR><LF>
a - the sensor address	a - the sensor address
I - the send identification command	II - the SDI-12 version number, indicating SDI-12 version compatibility; for example, version 1.0 is encoded as 10
! - terminates the command	ccccccc - an 8 character vendor identification, usually a company name or its abbreviation
	mmmmmm - 6 characters specifying the sensor model number
	vvv - 3 characters specifying the sensor version
	xxx . . . xx - an optional field, up to 13 characters, used for a serial number or other specific sensor information that is not relevant for operation of the data recorder
	<CR><LF> - terminates the response

Table 7. The send identification command (aI!)

4.4.2.1 Example of the Send Identification Command (aI!)

0I!010NRSYSINC1000001.2101<CR><LF>

The diagram illustrates the structure of the command '0I!010NRSYSINC1000001.2101'. Vertical lines connect each part of the command to its description:

- 0**: Sensor address
- I**: SDI-12 compatibility number
- 010N**: Company name
- RSYSINC**: Sensor model number
- 1.2**: Sensor version
- 101**: Serial number

4.4.3 Start Measurement Command (aM!)

This command tells the sensor to take a measurement. The sensor does not, however, return the measurement to the data recorder after this command. It returns the time until one or more measurements will be ready and the number of measurements that it will make. The send data (D0!) command must be issued to get the measurement(s). Table 8 shows the start measurement command.

Command	Response
aM!	attm<CR><LF>
a - the sensor address	a - the sensor address
M - the start measurement	ttt - the specified time, in seconds, until the sensor will have the measurement(s) ready
! - terminates the command	n - the number of measurement values the sensor will make and return in one or more subsequent D commands

Table 8. The start measurement command (aM!)

If the sensor returns 000 (ttt), the measurement is immediately available for transfer to the data recorder. The data recorder should issue the D0 command to get the data.

If ttt is not equal to zero (000), the data recorder must wait for the specified time to elapse. Then it wakes the sensor with a break and issues the D0 command. If, however, the sensor has the measurement ready before ttt seconds elapse, it will send a service request to the recorder. This tells the recorder to stop marking time and issue the D0 command.

When a data recorder issues an M command, it must complete the command/response sequence with the sensor before it sends any command to any other sensor. For example, suppose that following the issuance of the M command, the sensor responds

0M!00101<CR><LF>

This response shows that one data value will be ready in 10 seconds. The data recorder must wait for either of two events to occur before issuing a D0 command:

- 1) receipt of a service request (a<CR><LF>) from the sensor
- 2) the specified time to elapse (10 seconds)

The recorder then issues the D0 command to get the data. After the M command and before the sensor issues the service request, the data recorder will not communicate with any other sensor. **The sensor should not drive the data line until it sends a service request. The recorder should not drive the data line while waiting for the service request.**

After a sensor finishes its measurement procedure, the sensor must retain the measured data in memory until it receives another M or V command, or another command that returns data via the D0 command.

See the send data command, aD0!, aD1! . . . aD9!, for examples of this command. Because the start measurement command is closely related to the send data command, examples for both commands are shown on page 15.

4.4.3.1 Aborting a Measurement

If a sensor detects a break after it receives an M command, but before it issues a service request, it must abort its measurement procedure. It must also empty its buffer so that no data are returned in response to a D command. (The sensor address, followed by <CR><LF>, should be returned in response to subsequent D commands.) This requirement provides a way for a data recorder to abort a measurement so that it can communicate with another sensor on the SDI-12 bus.

4.4.4 Service Request (a<CR><LF>)

A service request is a response from a sensor. It is not a command. It is sent, after an M command, to tell the data recorder that the sensor has finished its measurement(s) and the data are ready. A service request is issued by the sensor, after an M or V command, when it has finished its measurement. The entire service request must be returned before ttt seconds (see Start Measurement Command, page 11) have elapsed. The time (ttt) is the maximum time that a sensor will take before it has data available.

If a data recorder does not detect a service request, the data recorder must wait for the specified time (ttt), given in response to the M or V command, and then issue the D0 command to get the data. The minimum time before a data recorder can issue the D0 command to get the data, in this case, is ttt seconds.

A sensor is required to issue a service request if it indicates that it will take one second or longer before data is ready, in response to an M or V command.

2), 4), 5) Send Data Command (aD0! . . . aD9!)

This command is used to get groups of data from the sensor. D0! is issued after an M or V command. The sensor responds by sending the data. If the expected number of measurements is not returned in response to a D0! command, the data recorder issues D1!, D2!, etc. until all measurement values are received. (The expected number of measurements is given in the response to an M or V command.) Table 9 shows the send data command.

If the response to a D command is valid, but no data are returned, the sensor has aborted the measurement. To obtain data the recorder must issue another M or V command.

Command	Response
aD0! (aD1! . . . aD9!)	a<values><CR><LF>
a - the sensor address	a - the sensor address
D0 - the send data command, D1 . . . D9 additional send data commands	values (see below)
! - terminates the command	<CR><LF> - terminates the response
	<values> - p.d.d
	p - the polarity sign (+ or -)
	d - numeric digits before the decimal place
	. - the decimal point (optional)
	d - numeric digits after the decimal point
	the maximum number of digits for a data value is 7, even without a decimal point
	the maximum number of characters in a data value is 9 (the polarity sign + 7 digits + the decimal point)

Table 9. The send data command (aD0!, aD1! . . . aD9!)

The maximum number of characters that can be returned in response to a D command, not counting the address, <CR> and <LF>, is 33.

4.4.5.1 Return of Multiple Measurements (Parameters) by a Sensor (D1! . . . D9!)

The commands D1 . . . D9 are used with sensors that return multiple measurements. The purpose of the D commands is for the sensor to return as many measurements as possible in response to each command. The limiting constraint, however, is that the total number of characters that can be returned in response to a D command is 36. If the total number of characters is greater than 36, the sensor fragments the response, sending the first group of measurements in response to D0, the next group in response to D1, and so on. A group can have one or more measurements; data collection always begins with the D0 command. See section 4.4.5.2 for examples.

The sensor must never split individual data values, sending part of a value in response to one D command, and then sending the rest of the characters, for that value, in response to the next D command.

If possible, a sensor should return all measurements in response to the D0 command. This is not, however, a requirement.

4.4.5.2 Examples of the Start Measurement Command (aM!) and the Send Data Command (aD!)

- a. One measurement is immediately available after the M command:

```
0M!00001<CR><LF>
0D0!0+3.14<CR><LF>
```

- b. Three measurements will be ready 5 seconds after the M command, and the sensor issues a service request. All 3 measurements are returned in response to the D0 command:

```
0M!00053<CR><LF>
0<CR><LF>
0D0!0+3.14+2.718+1.414<CR><LF>
```

- c. Nine measurements will be ready 35 seconds after the M command, and the sensor issues a service request. Because the number of characters in all 9 measurements is greater than 36, a D1 command must be issued to get the second group of measurements:

```
0M!00359<CR><LF>
0<CR><LF>
0D0!0+1.11+2.22+3.33+4.44+5.55+6.66<CR><LF>
0D1!0+7.77+8.88+9.99<CR><LF>
```

d. Two measurements will be available in 1 second, and the sensor does not issue a service request. After 1 second, the data recorder sends a break to wake the sensor and issues the D0 command:

```
0M!00012<CR><LF>
0D0!0+3.14+2.718<CR><LF>
```

e. Three measurements will be ready 5 seconds after the M command, and the sensor issues a service request. Upon receipt of the service request, the data recorder issues D0 to get the data. However, only 1 measurement is returned. The data recorder then issues the D1 command to get the next group of data. In response, the second measurement is returned. Then the recorder issues D2 to get the next, and last, group of data. In the example, each group contains only one measurement.

```
0M!00053<CR><LF>
0<CR><LF>
0D0!0+3.14<CR><LF>
0D1!0+2.718<CR><LF>
0D2!0+1.414<CR><LF>
```

Note: This is in compliance with the standard. As many measurements as possible, however, should be returned in response to each D command.

4.4.6 Additional Measurement Commands (aM1! . . . aM9!)

Additional M commands provide a means to request different types of measurements from a sensor or to instruct a sensor to do a calibration or a control function. For example, a sensor could measure pressure and temperature: M tells it to measure pressure and M1 tells it to measure the temperature.

Additional M commands have the same format as the aM! command. **Data collection always begins with the D0 command.** If the sensor does not return the expected number of measurements in response to the D0 command, the recorder should issue aD1, aD2, etc. until the sensor returns all measurements.

4.4.6.1 Examples of the Additional M Commands (aMn!)

a. A sensor supports the aM1! command:

```
0M1!00011<CR><LF>
0<CR><LF>
0D0!0+3.14<CR><LF>
```

b. A sensor takes 9 measurements in response to the M2 command:

```
0M2!00359<CR><LF>
0<CR><LF>
0D0!0+1.11+2.22+3.33+4.44+5.55+6.66<CR><LF>
0D1!0+7.77+8.88+9.99<CR><LF>
```

4.4.7 Start Verification (aV!)

This command tells the sensor to return a verification in response to a subsequent D command. A verification sequence may include ROM signatures, CRC's, RAM test results, or the results of other diagnostics in the sensor. A standard response to the V command is not specified.

The format of this command is the same as the M commands. The format of the response is the same as the D commands.

4.4.7.1 Example of the Start Verification Command (aV!)

```
0V!00011<CR><LF>  
0<CR><LF>  
0D0!0+1<CR><LF>
```

4.4.8 Extended Commands

Sensors are required only to respond to the basic SDI-12 command set. Sensors, however, usually require calibration, address changes, or other setup commands. Extended commands provide the means for such functions. An extended command is a command for a specific make of sensor to tell that sensor to do a specific task. Extended commands are defined and documented by the manufacturer of each sensor.

4.4.8.1 Transparent Mode

SDI-12 data recorders must have a mode in which extended commands can be sent to sensors. This is called the transparent mode. The transparent mode has the following characteristics.

- The data recorder buffers a command string received from a computer, terminal, or modem, until the command string is terminated.
- The data recorder wakes the sensor with a break, then it sends the buffered command to the sensor, using the SDI-12 protocol.
- The data recorder receives the response from the sensor and transmits the response to the computer, the terminal, or the modem.

5.0 SDI-12 TIMING

Figure 3 shows a timing diagram for an SDI-12 command and its response. The tolerance for all SDI-12 timing is ± 0.40 milliseconds. The only exception to this is the time between the stop bit of one character and the start bit of the next character. The maximum time for this is 1.66 milliseconds, with no tolerance.

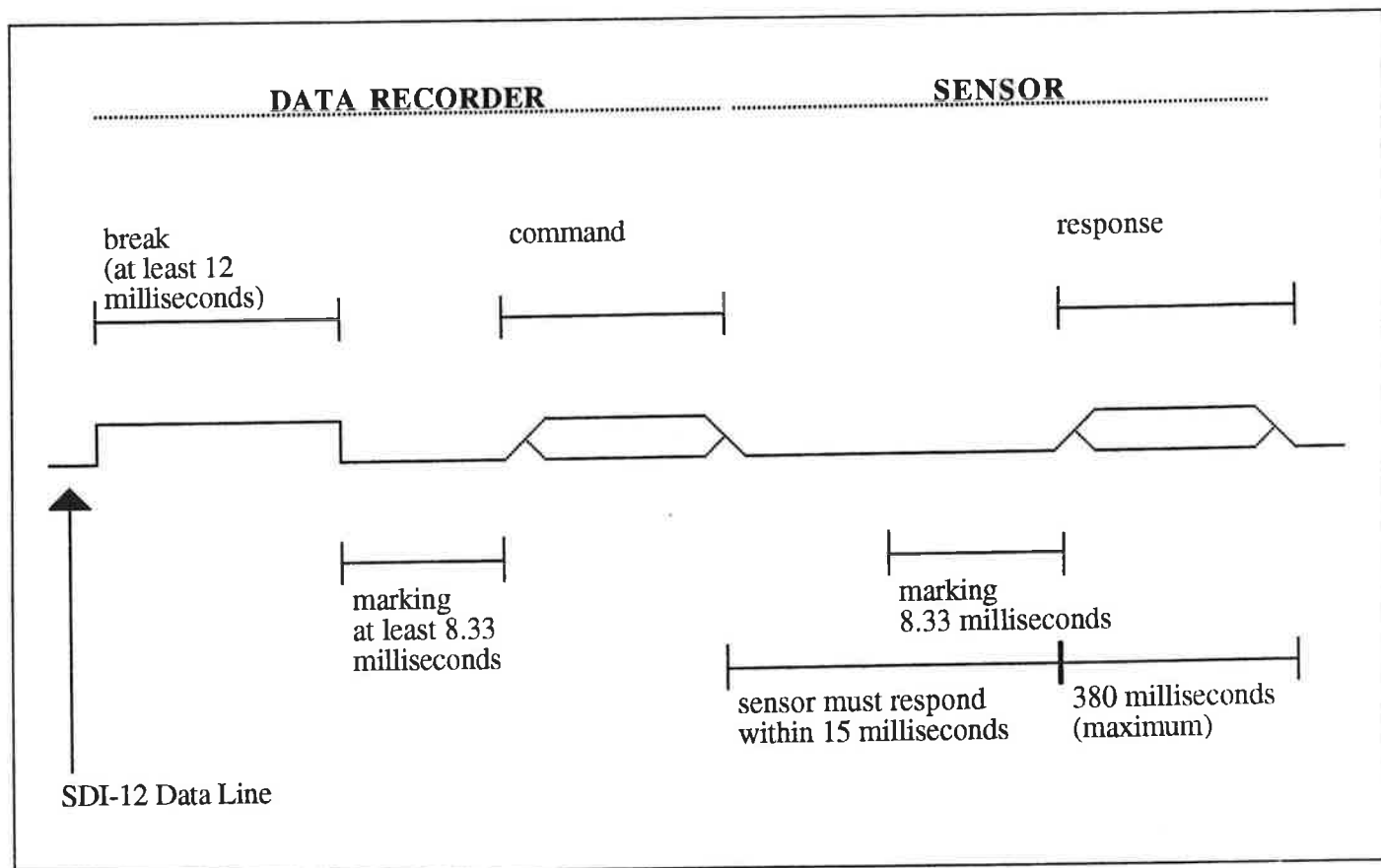


Figure 3. SDI-12 Timing

- A data recorder transmits a break by setting the data line to spacing for at least 12 milliseconds.
- The sensor will not recognize a break condition for a continuous spacing time of less than 6.5 milliseconds and will always recognize a break when the line is continuously spacing for more than 12 milliseconds.
- Upon receiving a break, a sensor must detect 8.33 milliseconds of marking on the data line before it looks for an address.

- A sensor must wake up from a low-power standby mode and be capable of detecting a start bit from a valid command within 100 milliseconds after detecting a break.
- After a data recorder transmits the last character of a command, it must relinquish control of the data line within 7.5 milliseconds following the end of the stop bit. (Tolerance: +0.40 milliseconds.)
- After receiving the break and the command, the addressed sensor sets the data line to marking for 8.33 milliseconds and then send the response. (Tolerance: -0.40 milliseconds.) The start bit of the first response byte must start within 15 milliseconds after the stop bit of the last byte of the command. (Tolerance: +0.40 milliseconds.)
- After a sensor transmits the last character of a response, it must relinquish control of the data line within 7.5 milliseconds. (Tolerance: +0.40 milliseconds.)
- No more than 1.66 milliseconds of marking are allowed between the end of the stop bit and the start bit (e.g., between characters) on any characters in the command or the response. (No tolerance.) This permits a command and a response to be sent within a 380-millisecond response window.
- Sensors must return to a low-power standby mode after receiving an invalid address or after detecting a marking state on the data line for 100 milliseconds. (Tolerance: +0.40 milliseconds.)
- When a recorder addresses a different sensor, or if the data line has been in the marking state for more than 87 milliseconds, the next command must be preceded by a break.

5.1 Rules for the Break

The data recorder sends a break when it is necessary to wake a sensor from low-power standby mode. An SDI-12 sensor is required to return to low-power standby mode after receiving an invalid address or after detecting a marking state on the data line for 100 milliseconds. For this reason, a break must precede a command whenever a new sensor is addressed and after 87 milliseconds of marking on the data line.

When a data recorder receives a service request, it does not have to send a break if it issues the D0 command within 87 milliseconds after the service request. If, however, more than 87 milliseconds elapse, the D0 command must be preceded with a break.

5.2 Retries

A data recorder must support retries. Sensors have up to 100 milliseconds to wake up after detecting a break and will not respond to any commands until they have awakened. Also, sensors will go to sleep after detecting 100 milliseconds of idle time (marking) on the SDI-12 bus.

When issuing retries, if no response is received from a sensor, the recorder must wait for at least 16.67 milliseconds after the last stop bit of the command, but no longer than 87 milliseconds, and then issue a retry (without a break). (This period of 87 milliseconds includes the 16.67 milliseconds spent waiting for a response from the sensor.) If a correct response is not received after retransmitting the command at least two more times, **with at least one of those retries more than 100 milliseconds after the end of the break**, the entire sequence (including the break and the retries) should be repeated at least two more times. The flow chart in appendix B of this document illustrates this retry logic. At least one of the retries must be issued after 100 milliseconds after the falling edge of the break to ensure that the sensor has been given the full 100 milliseconds to wake up after the break. A retry is needed if one of the following three conditions exist:

- 1) no response from the sensor;
- 2) 8.33 milliseconds of marking on the data line, after receiving the start bit of the response;
- 3) an invalid response.

Because the data recorder is the master of the SDI-12 bus, it is not required to issue a retry. The requirement is for the data recorder to support retries. If one of the above three conditions exists the data recorder will usually issue retries. As an exception to the rule, not as normal operating procedure, the data recorder has the option of not issuing retries.

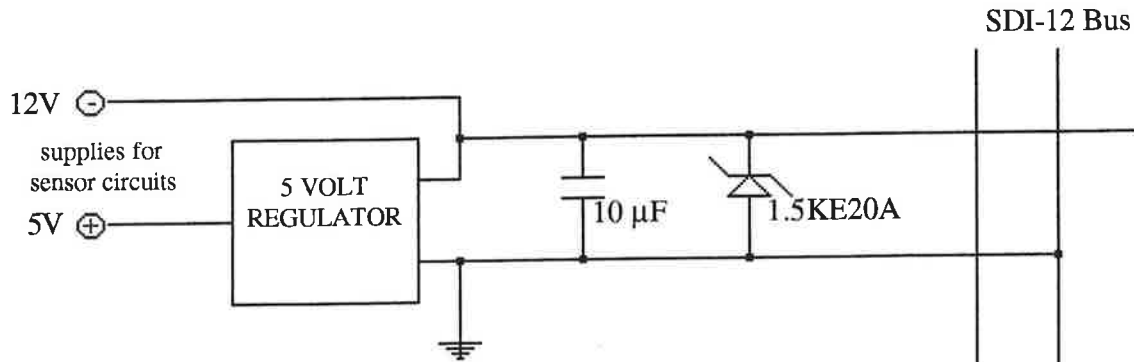
###

APPENDICES

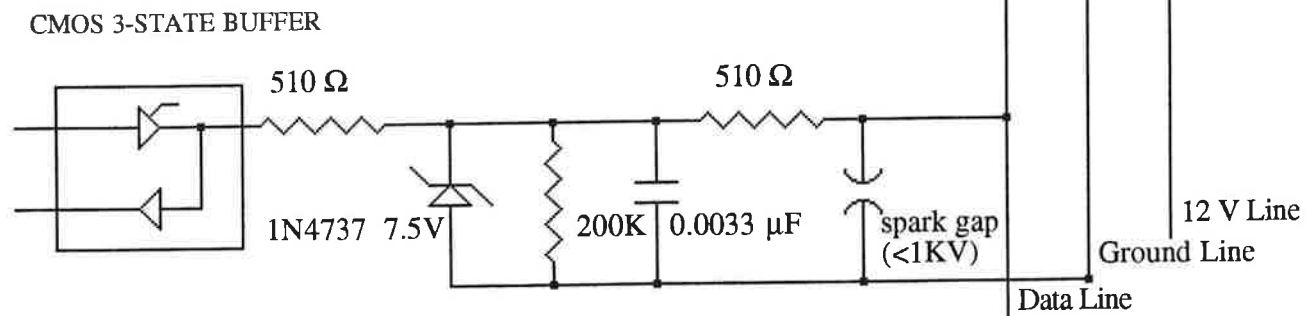
Appendix A
Suggested SDI-12 Circuits

Appendix A **Suggested SDI-12 Circuits**

sensor power circuit with transient protection



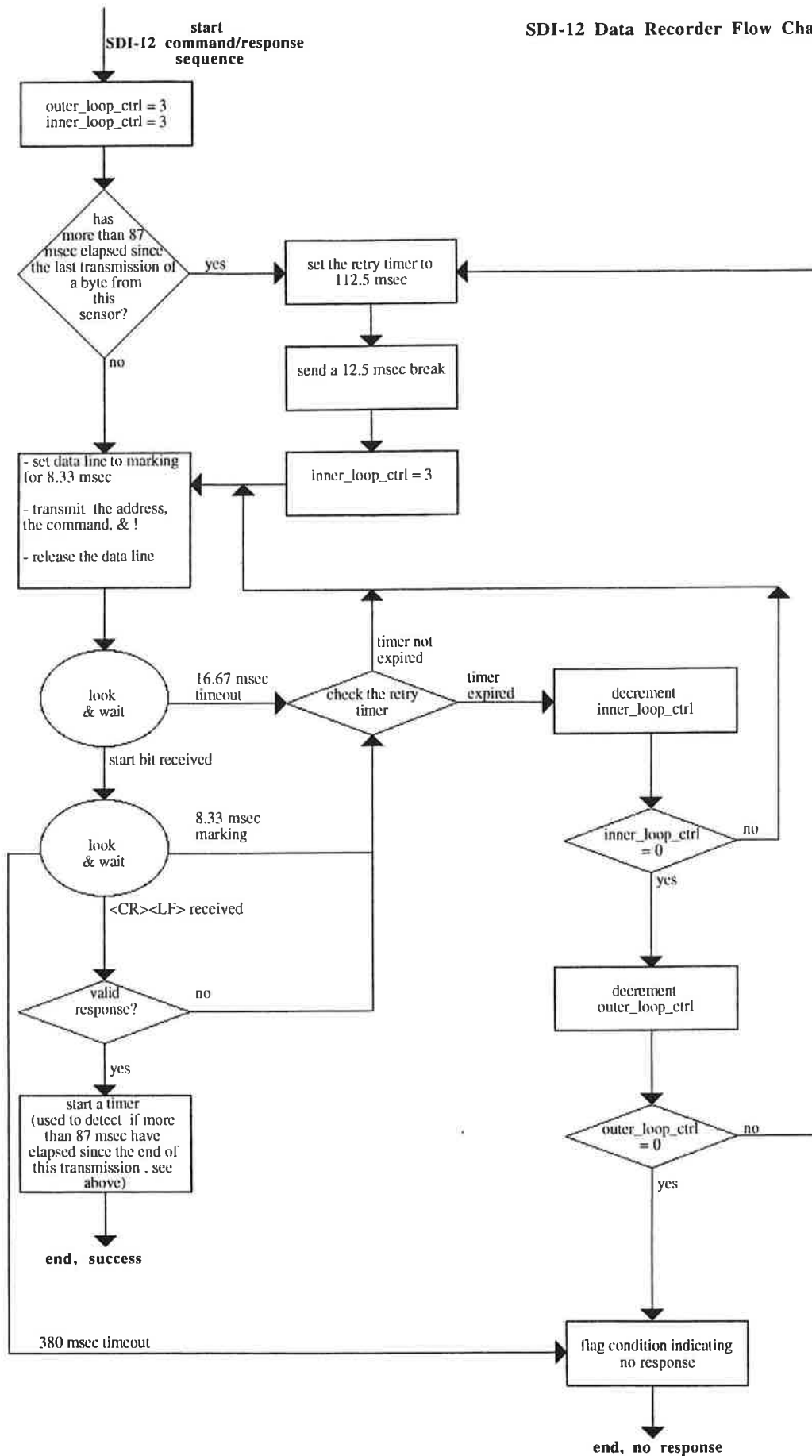
sensor or data recorder interface circuit with transient protection

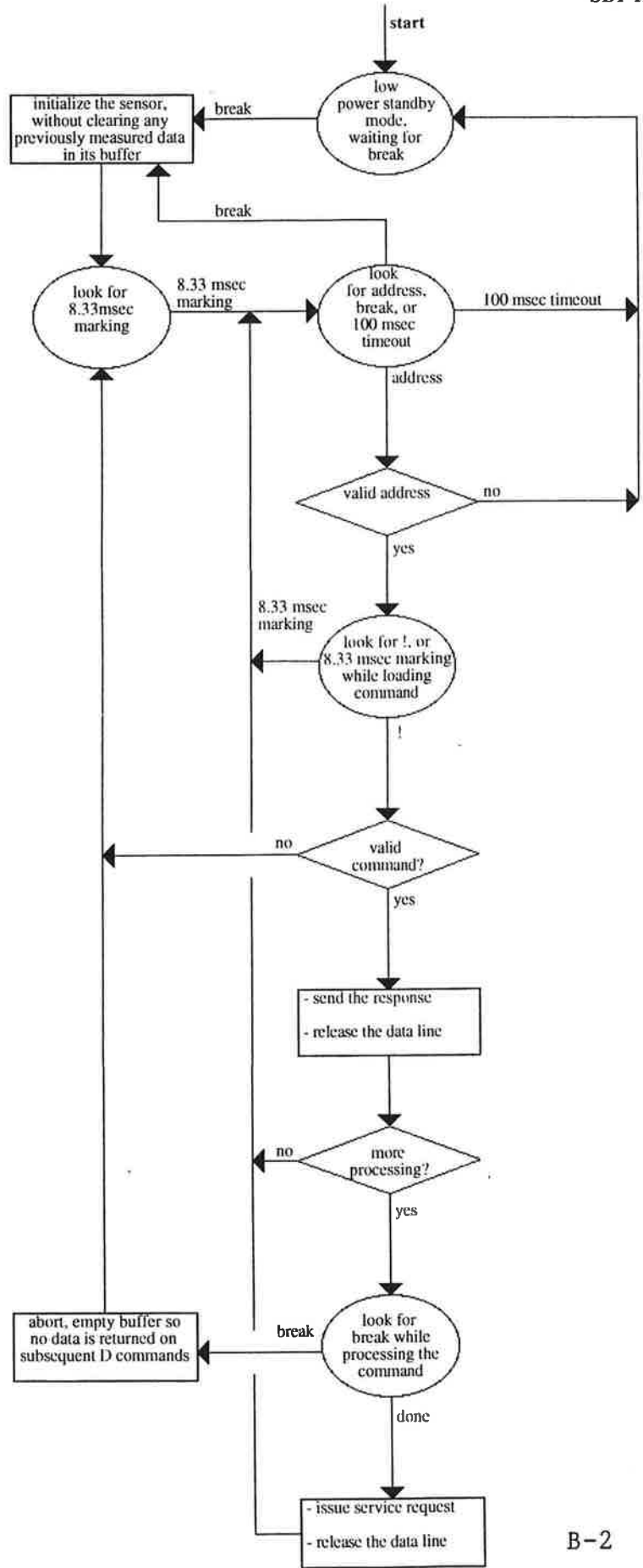


Appendix B

Suggested SDI-12 Flow Control
for
SDI-12 Data Recorders and Sensors

SDI-12 Data Recorder Flow Chart





Appendix C

SDI-12 Glossary

Appendix C

SDI-12 GLOSSARY

Address. A single character used to identify each sensor on the SDI-12 bus. The first character of every command and the first character of every response is an address. The normal address characters are "0" to "9."

Basic command set. The following SDI-12 commands: a!, aI!, aM!, aM1! . . . aM9!, aD0! . . . aD9!, aV!.

Break. Continuous spacing, by the data recorder, on the data line for at least 12 milliseconds. This is a special condition used to wake sensors from a low power standby mode.

Buffer. Memory in the sensor that holds the sensor's most recent measurement. A sensor is required to keep this information until it receives a new M or V command. When a sensor has a measurement in its buffer, repeated D commands cause it to return the same data each time the D command is issued. If, however, a break occurs while a sensor is making a measurement, the sensor must abort its measurement and empty its buffer. In this case, no data is returned in response to a D command.

Byte frame format. The manner in which a character is encoded for serial transmission. SDI-12 uses 1 start bit, 7 data bits, even parity, and 1 stop bit.

Data bits. The bits in a character that carry information, as opposed to the start bit, the parity bit, and the stop bit. SDI-12 uses 7 data bits.

Extended command. A command not in the basic SDI-12 command set. Extended commands are defined by the manufacturer of the sensor. Extended commands calibrate sensors, change addresses, run diagnostic tests in sensors, and other tasks specific to a sensor.

Even parity. Setting the parity bit to 0 or 1, as needed, to ensure that the total number of binary ones in the data bits and the parity bit are an even number. SDI-12 uses even, as opposed to odd, parity.

Marking. A binary state of 1 on the SDI-12 data line, with a voltage range of -0.5 to 1.0 volts.

Parity bit. A bit after the data bits in a character, used for error detection. SDI-12 uses even parity.

Printable characters. ASCII characters in the range 20 hex (a space) through 7E hex (~).

SDI-12 command. A string with 2 or more printable characters that will be sent to an SDI-12 sensor. The first character of an SDI-12 command is an address and the last character is an exclamation point (!). (The maximum number of characters for an SDI-12 command and its response is 40, not counting the <CR><LF>.)

SDI-12 bus. A cable with three conductors: (1) serial data, (2) ground, and (3) 12 volts--used to connect an SDI-12 data recorder with one to ten SDI-12 sensors.

SDI-12 data recorder. A data acquisition device that can be interfaced with one or more SDI-12 sensors. The recorder polls the sensor using the SDI-12 electrical interface and the SDI-12 communications protocol.

SDI-12 response. A string with 3 or more characters, returned to an SDI-12 data recorder by an SDI-12 sensor. The first character of an SDI-12 response is an address and the last two characters are a carriage return/line feed (<CR><LF>). Other than the carriage return/line feed, all characters must be printable ASCII characters. (The maximum number of characters for an SDI-12 command and its response is 40, not including the <CR><LF>.)

SDI-12 sensor. A measurement device that can be polled by an SDI-12 data recorder, using the SDI-12 electrical interface and the SDI-12 communications protocol.

Service request. An address followed by carriage return/line feed (a<CR><LF>). This is issued by a sensor, after an M or V command, to tell the data recorder that the requested measurements are ready.

Spacing. A binary state of 0 on the SDI-12 data line, with a voltage range of 3.5 to 5.5 volts.

Start bit. A bit indicating the start of a new character in an SDI-12 command or response.

Stop bit. A bit indicating the end of a character in an SDI-12 command or response. SDI-12 specifies one stop bit.

Transparent mode. A mode in an SDI-12 data recorder in which extended commands (or basic commands), received from a computer, a terminal, or a modem can be sent to a sensor. In transparent mode the data recorder buffers characters until terminated, wakes the sensor with a break, and sends the command string to the sensor. An example of transparent mode is a data recorder interfaced to a PC via a serial port. The user enters SDI-12 commands on the keyboard, and the command is transmitted to the data recorder. The recorder reads the command, transmits it to the sensor, and returns the response to the PC. The PC then displays the response.

Appendix D
Pass/Fail Criteria
for
SDI-12 Data Recorders and Sensors

Appendix D

PASS/FAIL CRITERIA FOR SDI-12 DATA RECORDERS AND SENSORS

PASS/FAIL CRITERIA FOR SDI-12 DATA RECORDERS

- The data recorder must communicate at 1200 baud, 1 start bit, 7 data bits, 1 parity bit (even parity), and 1 stop bit.
- The data recorder must send a break of at least 12.0 milliseconds when it is necessary to wake a sensor before transmitting a command.
- The data recorder must set the data line to marking for at least 8.33 milliseconds after a break, before transmitting the command.
- The data recorder must be able to transmit the following commands, as necessary, to take scheduled measurements from one or more SDI-12 sensors:

start measurement	aM!
send data	aD0!
subsequent D commands	aD1! ... aD9!

- If a service request is not received, the data recorder must issue a D command after the specified time, which given in response to the previous M command.
- The data recorder must issue subsequent D commands, incrementing from aD1!, ... aD9!, as necessary, to get all measurement groups from the sensor.
- When transmitting a command, the recorder must support retries if a response is not received from the sensor.
- The data recorder must have a transparent mode.
- The data recorder must relinquish control of the data line within 7.5 milliseconds after the last byte of the command.

PASS/FAIL CRITERIA FOR SDI-12 SENSORS

- The sensor must communicate at 1200 baud, 1 start bit, 7 data bits, 1 parity bit (even parity), and 1 stop bit.
- A sensor must respond to the following SDI-12 commands:

acknowledge active	a!
send identification	aI!
start verification	aV!
start measurement	aM!
send data	aD0!
subsequent D commands	aD1! ... aD9! (as needed)

- If the response to the send data command (aDn!) is more than 33 characters long, not including the address, <CR>, and <LF>, the sensor also must respond to the aD1!, aD2!, . . . aD9!, commands, as needed, in order to transfer all measurements to the data recorder. The sensor must never split individual data values, sending part of a value in response to one D command, and then sending the rest of the characters, for that value, in response to the next D command.
- All responses to SDI-12 commands must be in the correct format.
- After an aM!, aM1! . . . aM9!, or aV! command, the sensor must keep its measurement(s) in memory, until the next aM!, aM1! . . . aM9!, aV! command, or another command that returns data via the D command. There is, however, one exception to this. If a break occurs while the sensor is taking a measurement, after an M or V command, but before the sensor has finished its processing, the sensor must abort its measurement procedure and empty its buffer. This lets a data recorder abort the measurement process to begin communications with another sensor on the SDI-12 bus.
- The sensor must not respond to a valid command, after detecting 100 milliseconds of marking on the data line, until it detects a break, followed by a valid command.
- The sensor must not respond to a command, after an invalid address is transmitted on the data line, until it detects a break, followed by a valid command.
- The sensor must not respond to commands transmitted with no parity or odd parity.
- The sensor must detect a 12 millisecond break.
- A sensor must not detect less than 6.5 milliseconds of spacing as a break.

- The start bit of the response to an SDI-12 command must fall in the range:

7.93 milliseconds to 15.40 milliseconds,

after the stop bit of the last byte of the command. (This is the range 8.33 to 15.0 milliseconds, with a tolerance of 0.40 milliseconds.)

- The time from the start bit of the response to the stop bit of the <LF> character must not be greater than 380.40 milliseconds (380 milliseconds with a 0.40 millisecond tolerance):

$$(38 \text{ characters} * 8.33 \text{ msec/character}) + (38 * 1.66 \text{ msec}) = 379.62 \text{ msec}$$

- There must not be more than 1.66 milliseconds of marking between characters in all responses to SDI-12 commands.
- The sensor must relinquish control of the data line within 7.5 milliseconds after the last byte of the response.
- A sensor must wake up from a low-power standby mode and be capable of detecting a start bit from a valid command within 100 milliseconds after detecting a break.
- A sensor is required to issue a service request if it indicates that it will take one second or longer before data is ready, in response to an M or V command.

