
SDI-12 SUPPORT GROUP

June 1, 1995

Revised June 21, 1995, after review by the Technical Committee

Revised July 5, 1995, after further review by the Technical Committee

Proposed Changes to the SDI-12 Specification by The SDI-12 Support Group's Technical Committee

In response to the 1994 meeting of the SDI-12 Support Group and the SDI-12 Technical Committee held in Chicago, the SDI-12 Support Group's Technical Committee met on 27 April 1995 to continue discussions on a change address command, an address query command, extended commands, clarification of data recorder retries, clarification of device addresses, and concurrent measurements. This document describes recommendations by the SDI-12 Technical Committee on these subjects. Proposed changes to the SDI-12 Specification are shown in **bold type** in this document. All changes agreed upon by a two-thirds majority of the Support Group will be included in Version 1.2 of the SDI-12 Specification. (Under the Group's bylaws, this is a two-thirds majority of those who return a ballot on the proposed changes.) SDI-12 is currently at Version 1.1.

CHANGE ADDRESS COMMAND

The technical committee recommends the addition of a change address command to the SDI-12 command set. The change address command will be listed on page 8 in Table 5, the SDI-12 basic command set. The command will work as follows:

<i>Command</i>	<i>Response</i>
aAb!	b<CR><LF>
a - the sensor address	b - the address of the sensor (will equal the new address or the original address if the sensor is unable to change the address)
A - the change address command	
b - the address to change to	<CR><LF> - terminates the response
! - terminates the command	

If the sensor supports software changeable addresses, it must support the address query command.

ADDRESS QUERY COMMAND

The technical committee recommends the addition of an address query command to the SDI-12 command set. The address query command will be listed on page 8 in Table 5, the SDI-12 basic command set, and will be described with the following text.

When a question mark (?) is used as the address character with the acknowledge active command (a!), the sensor will respond as if it is being addressed on the SDI-12 bus. For example, if a sensor detects ?!, it will respond with a<CR><LF>, no matter what its address may be. This will allow a user to determine the address of a sensor. The user should understand that if more than one sensor is connected to the bus, they will all respond, causing a bus contention. A sensor is required to support the wild card address character (?) only with the acknowledge active command (?!).

Note: In the Fall 1994 newsletter, this command was described using an asterisk rather than a question mark. By a vote of 7 to 2, at the April, 1995 technical committee meeting, the committee voted to use the ~~dollar~~ sign rather than an asterisk.

SDI-12 SUPPORT GROUP

EXTENDED COMMANDS

Text will be added to page 19, section 4.4.8, Extended Commands, as follows.

Extended commands have the following attributes:

- An extended command must be prefixed with an address
- An extended command must be terminated with an exclamation point
- The response must be prefixed with an address
- The response must be terminated with <CR><LF>
- The transparent mode must support basic SDI-12 commands and extended commands

Extended commands should be prefixed with an upper case X, for example, aXNNN!, where X says that an extended command follows and NNN is the extended command. NNN is not limited to three characters. This is a recommendation only and not a proposed change to the SDI-12 Specification. Future versions of SDI-12, however, may require this.

CHARACTER ADDRESSES

The technical committee proposes that the following text be added to page 7, section 4.3.

ASCII '0' through ASCII '9' are the standard addresses which all sensors and data recorders must support. All ASCII characters in the range above ASCII '9' through ASCII '~' are reserved for single character device addresses. Should there be a need for more than 10 sensors, use an address in the range ASCII 'A' through ASCII 'Z' (decimal 65 through 90) then ASCII 'a' through ASCII 'z' (decimal 97 through 122).

DATA RECORDER RETRIES

Section 5.2, Retries, on page 22 of the specification, will be reworded as follows to clarify data recorder retries. This is a clarification only and not a proposed change to the specification.

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

CONCURRENT MEASUREMENTS

There is interest in adding simultaneous or concurrent measurements to SDI-12 where multiple sensors can be commanded to collect data at the same time. At the annual meeting of the SDI-12 Support Group in November of 1994 the majority preferred the terminology "Concurrent" over "Simultaneous." This description, therefore, refers to concurrent measurements and defines an SDI-12 concurrent measurement command.

The following criteria must be met to add concurrent measurements to SDI-12:

1. Version 1.2 sensors can be read by Version 1.1 data recorders. This means that M commands must still be supported.
2. Version 1.1 sensors can be read by version 1.2 data recorders. This should not be a problem.

SDI-12 SUPPORT GROUP

3. With a mixture of Version 1.1 and 1.2 sensors connected to a version 1.2 data recorder, the data recorder or the operator must appropriately group the sensors.

The reason for using concurrent measurements is to increase data throughput on an SDI-12 network. In light of this, the 36 character constraint on the returned data should also be addressed. The 36 character constraint was intended as a debugging aid in 1988 where 40 character displays were common. For data returned from concurrent measurements, the character limit should be expanded to an 80 character display. This would limit the response to 76 characters, including the address and the <CR><LF>. Changes to the SDI-12 specification cannot make a Version 1.2 sensor incompatible with a Version 1.1 data recorder. Therefore, this expansion can only apply to the new commands because a version 1.1 data recorder issuing M commands may only have a buffer large enough for a 36 character response. (Note: 36 characters includes the address and the <CR><LF> characters.)

The Concurrent Measurement Command

Concurrent measurements use the same command structure as the other SDI-12 measurement commands. The commands are: aC!, aC1! . . . aC9!. The response to a concurrent measurement command is similar to the response to an M command, atttⁿⁿ<CR><LF>. The difference being that the number of returned values is up to 99 (nn) not 9 (n), or to 760 characters (76 * 10 D commands), whichever is less. A data recorder must be able to read and store at least 20 data values in response to a C command, e.g. nn=20. All version 1.2 sensors and data loggers must support concurrent measurements. If a sensor does not have parameters defined for C1 through C9 then it should return a00000<CR><LF> indicating that it will not take a measurement in response to the command. This saves time because the data recorder will not issue retries. The sensor will not issue a service request when the measurement is completed.

Any valid command addressed to a sensor performing a concurrent measurement will abort the measurement. An invalid command (i.e., improper parity, incomplete, etc) should not abort the measurement. Any command addressed to another sensor should not abort the measurement.

1. The concurrent measurement command will be listed on page 8 in Table 5.

<i>Name</i>	<i>Command</i>	<i>Response</i>
Start Concurrent Measurement	aC!	attt ⁿⁿ <CR><LF>
Additional Concurrent Measurements	aC1!	attt ⁿⁿ <CR><LF>
	.	attt ⁿⁿ <CR><LF>
	.	attt ⁿⁿ <CR><LF>
	.	attt ⁿⁿ <CR><LF>

The SDI-12 Basic Command/Response Set
Table 5

2. A new section of text will be added to describe the concurrent measurement command. The following text will be placed between section 4.4.4 (Service Request) on page 13 and section 4.4.5 (Send Data Command).

4.4.5 Start Concurrent Measurement Command (aC!)

This command is new to Version 1.2 of the SDI-12 Specification. For a sensor or data logger to claim support of Version 1.2 of the SDI-12 Specification, it must support this command. Claiming zero data values in response to this command (i.e. a response of a00000<CR><LF>) is not a valid support of this command. The command must initiate a true measurement cycle. Sensors or data recorders that conform to a previous

SDI-12 SUPPORT GROUP

version of SDI-12 will most likely not support this command, and will therefore not respond to this command.

This command tells the sensor to take a concurrent measurement. A concurrent measurement is one which occurs while other SDI-12 sensors on the bus are also taking measurements. The sensor does not, however, return the measurement to the data recorder after this command. It returns the time until all measurements will be ready and the number of measurement that it will make. The send data (D0!) command must be issued to collect the measurements(s). Table 9 shows the start concurrent measurement command.

<i>Command</i>	<i>Response</i>
aC!	attnn<CR><LF>
a - the sensor address	a - the sensor address
C - the start concurrent measurement command	ttt - the specified time, in seconds, until the sensor will have the measurement(s) ready
! - terminates the command	nn - the number of measurement values the sensor will make and return in response to one or more subsequent D commands (a data recorder must be able to read and store at least 20 parameters from a sensor, nn=20)

The Start Concurrent Measurement Command (aC!)
Table 9

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 before attempting to retrieve the data. During this time the data recorder can collect data from sensors at other addresses. Communicating with other sensors will not abort a concurrent measurement. After the specified time has elapsed, the data recorder wakes the sensor with a break and issues the D0 command. The sensor will not issue a service request when it has completed the measurement. The sensor should not drive the data line from the time it completes its attnn<CR><LF> response until it starts responding to the D0 command.

After a sensor finishes its measurement procedure, the sensor must retain the measured data in memory until it receives another C, 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 concurrent measurement command is closely related to the send data command, examples for both commands are shown on page xx.

4.4.5.1 Aborting a Concurrent Measurement

If a sensor receives a valid command addressed to it while it is in the process of a concurrent measurement, it should abort the 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.

SDI-12 SUPPORT GROUP

3. The last sentence on page 14, section 4.4.5, will be changed to say:

The maximum number of characters that can be returned in response to a D command issued to retrieve data from any command other than a concurrent measurement command is 36, counting the address, <CR>, and <LF>. For the concurrent measurement results the limit is 76, counting the address, <CR>, and <LF>.

4. The first paragraph on page 15, section 4.4.5.1, will be modified to say responses to D commands, after concurrent measurement commands, can return up to 76 characters:

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 (76 if the measurement command was a concurrent measurement). If the total number of characters is greater than 36 (76 if the measurement was a concurrent measurement), 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.

5. The following example of concurrent measurements will be added to page 16:

- f. Two sensors, one returning 12 readings after 45 seconds and the other returning 4 readings after 15 seconds. The measurements are taken concurrently. Fifteen seconds after requesting data from sensor 1, the data recorder issues a break followed by the D0 command. Forty-five seconds (or longer) after requesting data from the sensor at address 0, the data recorder sends a break and a D0 command to sensor 0. Note that since a concurrent measurement was requested of sensor 0, it is allowed to return up to 76 characters in its response. An M command only allows 36 characters to ensure compatibility with data recorders prior to version 1.2.

0C!004512<CR><LF>

1C!101504<CR><LF>

1D0!1+1.23+2.34+3.45+4.4678<CR><LF>

0D0!0+1.234-4.56+12354-0.00045+2.223+145.5+7.7003+4328.8+9+10+11.433+12<CR><LF>

6. The following statement will be added to page 17, section 4.4.6, Additional Measurement Commands

To comply with Version 1.2 or higher of SDI-12, sensors must respond to the additional measurement commands (aM1! . . . aM9!) and data recorders must be able to log data from the additional measurement commands. If a sensor has no data defined for an additional measurement command, it should return a0000<CR><LF>, saying that it has zero data values ready. Not responding to the command is not acceptable.

7. The following section will be placed after page 17, section 4.4.6, Additional Measurement Commands.

4.4.7 Additional Concurrent Measurement Commands (aC1! . . . aC9!)

Additional C 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. To comply with Version 1.2 or higher of SDI-12,

SDI-12 SUPPORT GROUP

sensors must respond to the additional concurrent measurement commands and data recorders must be able to log data from the additional concurrent measurement commands. If a sensor has no parameters defined for an additional concurrent measurement command, then it should return a0000<CR><LF>, saying that it has zero data values ready. Not responding to the command is not acceptable.

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

8. On page 21, Section 5.0, SDI-12 Timing, the last sentence of the 5th bullet will be changed as follows.

This permits a response to be sent within a 360 millisecond (760 milliseconds for a D command after a concurrent measurement) millisecond response window.

9. Three changes will be added to Appendix B, the SDI-12 glossary:

- a. The definition of the basic command set will be modified to include the commands ~~?~~ aC!, aC1! . . . aC9! (page C-1).
b. The next to the last sentence in the definition of buffer, on page C-1, will be modified to say:

If, however, a break occurs while a sensor is making a measurement in response to an M or V command, the sensor must abort its measurement and empty its data buffer.

- c. The sentence, "The maximum number of characters for an SDI-12 command and its response is 40, not including the <CR><LF>", will be removed from the definitions of SDI-12 command and SDI-12 response (page C-2).

Other Types of Measurements

The technical committee also discussed two other measurement schemes for SDI-12 sensors. An overview of these schemes is described here. All members of the SDI-12 Support group will vote on the following two types of measurements. If the Group votes in their favor they will also be included in Version 1.2 of the SDI-12 Specification.

Continuous Measurements

This is a proposed feature of the D commands, which will be placed in section 4.4.5, Send Data Command, on page 14.

Sensors that are able to continuously monitor the phenomena to be measured, such as a shaft encoder, do not require a start measurement command (M!, M1! . . . M9!) prior to the send data command (D0!). For example:

if (the sensor is operating in a continuous measurement mode) then

aD0! will get and return the current reading of the sensor

The technical committee feels that this is not placing a new requirement on SDI-12; it removes a requirement from the specification (the need for an M command with certain sensors).

Strobed Measurements

The following text will be added as section 4.4.9 in the specification, following the description of extended commands.

SDI-12 SUPPORT GROUP

The commands, aS! and aG!, will be listed on page 8, in Table 5 and in the glossary in appendix C, as one of the SDI-12 basic commands.

4.4.9 Strobed Measurements

Strobed measurements command two or more SDI-12 sensors to take a measurement at the same time. This is the aS! command, which is described in the following table.

<i>Data Recorder Command</i>	<i>Comments</i>	<i>Sensor(s) Response</i>
1S!	tell sensor 1 to enter the simultaneous mode	att ₁ nn ₁ <CR><LF>
2S!	tell sensor 2 to enter the simultaneous mode	att ₂ nn ₂ <CR><LF>
3S!	tell sensor 3 to enter the simultaneous mode	att ₃ nn ₃ <CR><LF>
?G!	go command, tells all sensors to begin measuring	no response

assume that $ttt_1 \geq ttt_2 \geq ttt_3$

The sensors that respond to the aS! command must remain awake and monitoring the SDI-12 bus after responding the aS! command. These sensors must remain in this state until they receive they receive the ?G! command (G for go).

The go command tells all sensors with simultaneous mode enabled to begin measuring. After ttt_3 seconds have elapsed, the data recorder will issue D commands to each sensor to get the data.

Sensors not supporting the S command will ignore the G command. Sensors do not send a response to the G command.

Sensors will not issue a service request when they have completed their measurements. A sensor should not drive the data line from the time it completes its attnn<CR><LF> response until it starts responding to the D0 command.

4.4.9.1 Aborting a Strobed Measurement

If a sensor receives a valid command addressed to it while it is in the process of a making a strobed measurement, it should abort the 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.

Appendix E, Revisions

A new appendix (Appendix E, Revisions) will be added to the SDI-12 Specification.

Version 1.2 month/day/year (date to be determined upon approval by the SDI-12 Support Group)

1. Added the concurrent measurement command (aC!) and the additional concurrent measurement commands (aC1! . . . aC9!) to the basic command set.
2. Clarified the section on data recorder retries to indicate that a data recorder must issue retries unless it has received a valid response or if it wishes to abort the measurement.

SDI-12 SUPPORT GROUP

3. Clarified the no data response to the additional measurement commands (aM! ... aM9!)
4. Added the address query command (!) to the basic command set.
5. Added the change address command (aAb!) to the basic command set.
6. Added the Strobed Measurement Command (aS!) and the Go command (aG!) to the basic command set.

Version 1.1 July 7, 1994

Rewrite and clarification of the original specification. No technical changes.

Version 1.0 October, 1988

Original release.