SEIKO EPSON CORPORATION

A342/A542
Quick Connection Guide
(data communication)

Evaluation board/kit and development tool important notice

1. This evaluation board/kit or development tool is designed for use for engineering evaluation, demonstration, or development purposes only. Do not use it for any other purposes. It is not intended to meet the requirements of design for finished products.

2. This evaluation board/kit or development tool is intended for use by an electronics engineer and is not a consumer product. The user should use it properly and in a safe manner. Seiko Epson does not assume any responsibility or liability of any kind of damage and/or fire coursed using it. The user should cease to use it when any abnormal issue occurs even during proper and safe use.

3. The parts used in the evaluation board/kit or development tool are subject to change without notice.

目次

Revision History

Rev. No.	Date	Page	Description
20220308	2022/3/8	ALL	New release.

1. Introduction

This document is intended to provide users with a quick and easy installation of A342/A542 vibration sensor for data communication control. It explains controls in UART Auto Sampling mode. For other measurement modes (UART burst, SPI burst/normal) and detailed register specifications, please refer to the latest version of the datasheet.

In this document, the measurement under the following conditions (1) and (2) will be explained according to the steps 1 to 6 in Table 1.1.

1）Velocity RAW Sampling at 3000 Sps

- When performing vibration analysis such as frequency analysis or Lissajous analysis.
- To perform RMS calculation required for ISO10816 evaluation on a host device (e.g. PC), etc.
- Figure 1.1 shows an example of the A342/A542 Velocity RAW Output.

2）Velocity RMS Sampling at 1 Sps

- In case of ISO 10816 evaluation with the velocity RMS value calculated by the sensor, etc.
  ※Low sampling rate (0.1 s to 25.5 s) and no requirement for RMS calculation saves processing load on the host device.
- Figure 1.2 shows an example of the A342/A542 Velocity RMS Output.

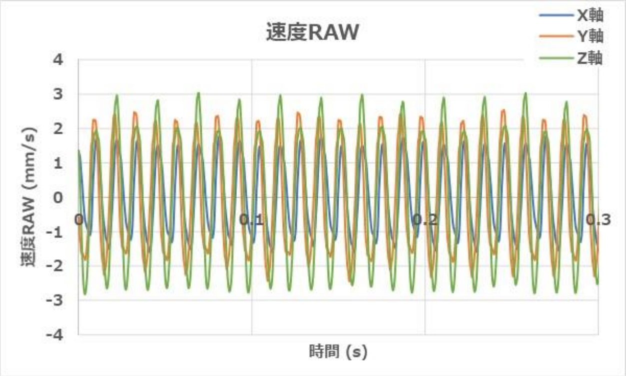


Figure 1.1 A342/A542 Velocity RAW Output

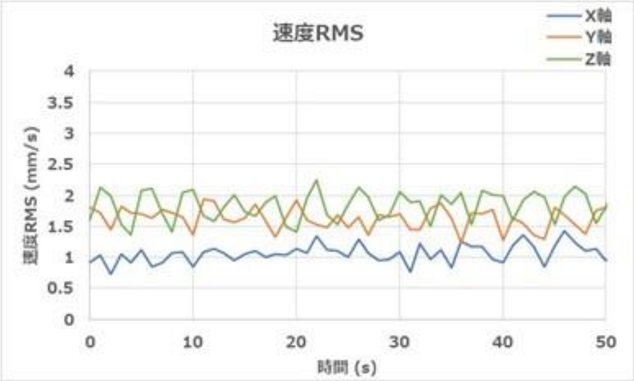


Figure 1.2 A342/A542 Velocity RMS Output

Table 1.1 Description Procedure in This Document

Step	Description
(1) Device Connection	Explanation of the device connection between the sensor and the host device.
(2) Basics of Command Control	Basics of command-based sensor control used in Steps 3 to 5.
(3) Power-On to Start-Up	Explanation of the sequence from power-On / reset to initialization completion.
(4) Configurations	Explanation of the configuration procedure for the output physical quantity.
(5) Data Acquisition	Explanation of the data acquisition procedure when UART Auto sampling is used.
(6) Converting Acquired Data to Physical Quantity Units	Explanation of converting the acquired data to physical quantity units.

2. (Step 1) Device Connection

2.1. Device Connection Configuration

The sensor is connected to the host device (MCU, PC, etc.). Connection configuration examples are shown below.

Refer to the Quick Connection Guide (device connenction) "A542A342_A552A352_Quick-Connection-Guide(device connection).pdf" for details.

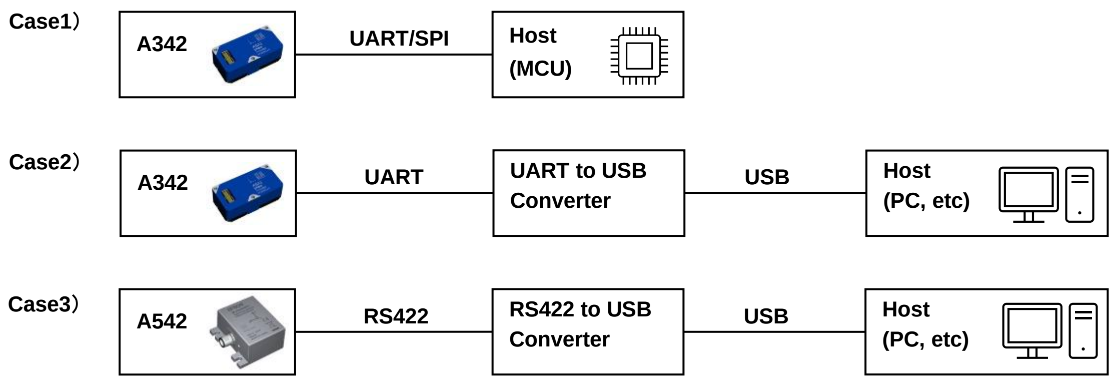
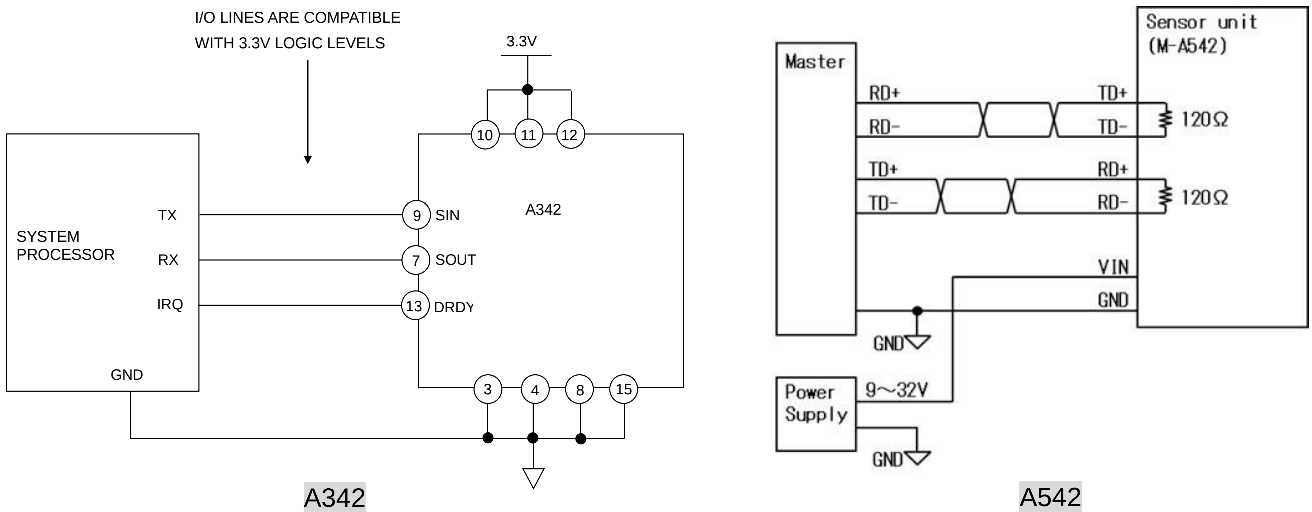


Figure 2.1 Examples of A342/A542 Connection Configuration

2.2. Serial Interface Connection Diagram

Please provide noise countermeasures for the power supply if necessary. The connection diagrams and communication format are shown below.



A342 A542

Figure 2.2 Connection Diagrams

Table 2.1 Communication Settings

Parameter	A342 settings
Transfer Rate	460.8 kbps (Factory default) ※115.2 kbps / 230.4 kbps / 921.6 kbps are also available.
Start Bit	1 bit
Data Bits	8 bits
Stop Bit	1 bit
Parity	None
Delimiter	CR (0x0D)

3. (Step 2) Basics of Command Control

The sensor control is conducted by sending a command to access the specific register where the function is assigned. Below is a description of the basics of command-based sensor control.

3.1. Register Read Command (UART)

By sending the register read command to the sensor via UART, the data stored in the register can be read out in 16-bit units. This acquires the various sensor settings and status information stored in the register.

The read command format is as follows: 1st byte is the register address, 2nd byte is unused (no argument specified), and 3rd byte is the delimiter (0x0d). The 1st byte: bit [7] of the read command is fixed to "0". For example, when reading the MODE_CTRL register at 0x02-0x03, the 1st byte is filled with the lower address 0x02 of the MODE_CTRL register as it is.

The response format is as follows: 1st byte is the register address, 2nd to 3rd bytes correspond to 16-bit read data, and 4th byte is the delimiter (0x0d).

The register read timing, read command format, and UART register read example are shown below.

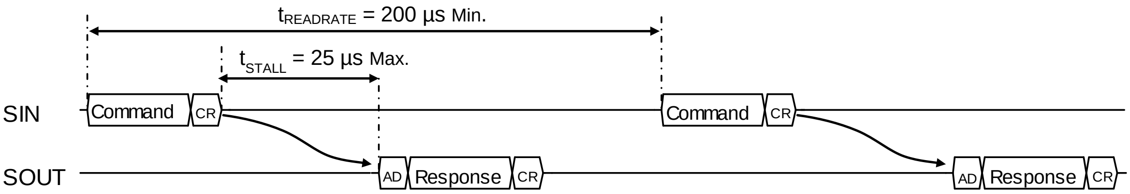


Figure 3.1 UART Read Timing (460.8 kbps)

Table 3.1 Read Command Format

1st byte								2nd byte								3rd byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	A[6:0]							XX								0x0D							

A [6:0] ··· Register address (even address)
XX ··· Don't Care
0x0D ··· Delimiter

Table 3.2 Read Response Format

1st byte								2nd byte								3rd byte								4th byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	A[6:0]							D[15:8]								D[7:0]								0x0D							

A [6:0] ··· Register address (even address)
D [15:8] ··· Register read data (upper byte)
D [7:0] ··· Register read data (lower byte)
0x0D ··· Delimiter

Register Read Example (UART)

Read the current operation mode from MODE_CTRL register.

Tx = {*0x02, 0x00, 0x0d*}. /* Read out MODE_CTRL */
Rx = {*0x02, 0x04, 0x00, 0x0d*}. /* Retrieve the response value */
1st byte: 0x02 corresponds to the lower address of MODE_CTRL.
2nd byte: 0x04 indicates the current operation state being in Configuration mode.
3rd byte: 0x00 is a dummy response.
4th byte: 0x0d corresponds to a delimiter.

3.2. Register Write Command (UART)

By sending the register write command to the sensor via UART, the register write data is written in 8-bit units. Various sensor settings and measurement operations can be controlled by this command. No corresponding response for it.

The write command format is as follows: 1st byte is the register address, 2nd byte is the write data, and 3rd byte is the delimiter (0x0d). The 1st byte: bit [7] of the read command is fixed to "1". For example, when writing data to the upper address 0x03 (00000011b) of the MODE_CTRL register, add "1" to the head of the address then 0x83 (10000011b) becomes the 1st byte.

The register write timing, write command format, and UART register write example are shown below.

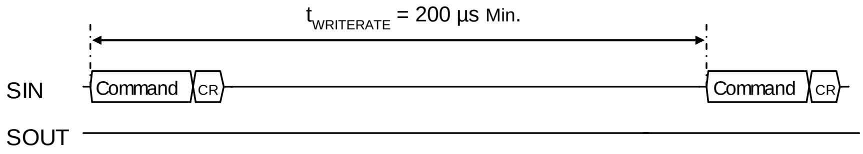


Figure 3.2 UART Write Timing (460.8 kbps)

Table 3.3 Write Command Format

1st byte								2nd byte								3rd byte							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
1	A[6:0]							D[7:0]								0x0D							

A [6:0] ･･･ Register Address (odd or even number)
D [7:0] ･･･ Register Write Data
0x0D ･･･ Delimiter

<u>Register Write Example (UART)</u>

Write “0x01” to MODE_CTRL (L) to go to Sampling mode.

Tx = {*0x83, 0x01, 0x0d*}. /* Write to MODE_CTRL (L) to go to Sampling mode */

1st byte: 0x83 is the value obtained by setting bit [7] of the upper address 0x03 of MODE_CTRL to 1.

2nd byte: 0x01 executes the transition to Sampling mode.

3rd byte: 0x0d corresponds to a delimiter.

3.3. Register Specifications

The following register specifications need to be understood for command-based sensor control. As for specific use cases, refer to the process flows in Step 3 to Step 5.

<u>Register Format and Basics of Command Control</u>

- Each register has a 16-bit data length, with an address assigned every 8 bits. Register access is performed in 16-bit units when reading and in 8-bit units when writing. The data in the 16-bit registers is stored in little-endian format.
- The registers are accessed in this device using a WINDOW method. The window number is first written to **WINDOW_ID**, then the desired register address can be accessed. The window number of each register is shown in the register map in Table 3.4, but basically grouped as 1) and 2) below.

  1) Sensor settings are made in Configuration mode with **WINDOW_ID** = 1.

  2) Readout of measurement results is made in Sampling mode with **WINDOW_ID** = 0.
- Initial values of the registers after startup are the values shown in the "Default" column in Table 3.4.

Table 3.4 shows the register map used in Step3 ~ Step5.

Table 3.4 Register map used in Step3 ~ Step5

Name	Window ID	Address	(*1) Read Command 16-bit Read	Write Command 8-bit Write	R/W	Flash Backup	Default	Function
MODE_CTRL	0	0x02	0x02XX	-	-	–	0x00	Operation mode control
		0x03		0x83	R/W		0x04	
DIAG_STAT1	0	0x04	0x04XX	-	R	–	0x00	Diagnostic result 1
		0x05		-	R		0x00	
SIG_CTRL	1	0x00	0x00XX	0x80	R/W	○	0x00	ND flag control
		0x01		0x81	R/W		0x8E	Output mode control
SMPL_CTRL	1	0x04	0x04XX	0x84	R/W	○	0x07	Sampling control
		0x05		0x85	R/W		0x0A	
GLOB_CMD	1	0x0A	0x0AXX	0x8A	R/W	–	0x00	System control
		0x0B		-	R		0x00	
BURST_CTRL	1	0x0C	0x0CXX	0x8C	R/W	○	0x00	Burst control
		0x0D		0x8D	R/W		0x47	
WIN_CTRL	0,1	0x7E	0x7EXX	0xFE	R/W	–	0x00	Register Window Control

* 1) Lower byte XX: Do not care

4. (Step 3) Power-On to Start-Up

How to activate the sensor is described below.

When the 3.3V power supply is applied to the sensor, the internal initialization starts as shown in the state transition diagram below.

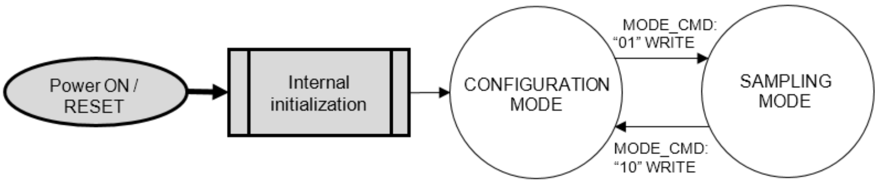


Figure 4.1 State Transition Diagram (Start-Up)

Do not access the sensor during the initialization process (Power-On Start-Up Time = 800 ms Max.). The results of the hardware check at startup can be read by DIAG_STAT1 register. After the internal initialization is complete, start the sensor control. The startup process flow is shown below.

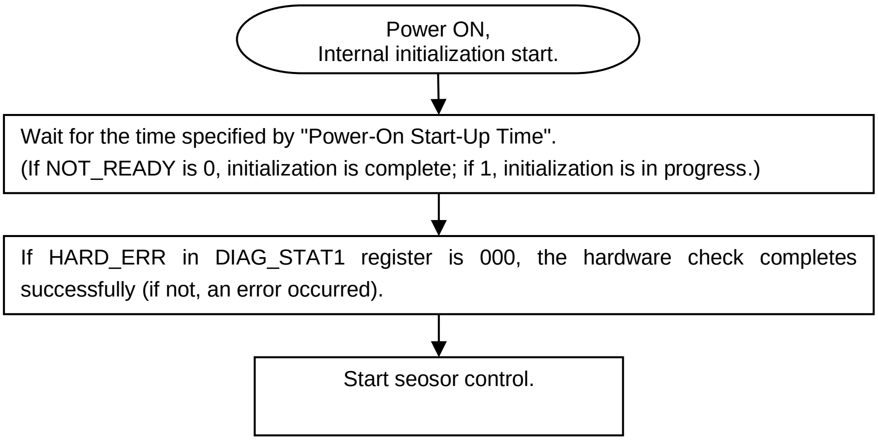


Figure 4.2 Startup Process Flow

5. (Step 4) Configurations

How to configure the sensor is explained below.

As shown in the state transition diagram below (Figure 5.1), the state after initialization automatically shifts to the configuration mode. The sensor configurations can be made in this configuration mode.

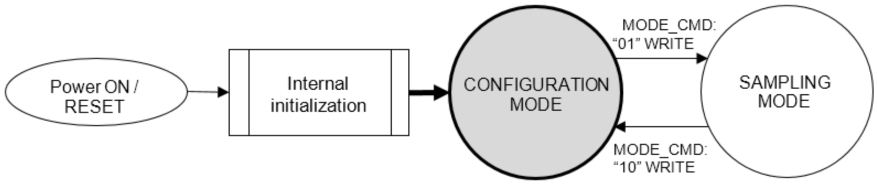


Figure 5.1 State Transition Diagram (Configuration Mode)

The following describes the specific processing steps for making typical sensor configrations shown in Table 5.1.

Table 5.1 Examples of Typical Sensor Configurations

Setting item	1) Velocity RAW data (factory default)	2) Velocity RMS data
UART baud rate	460.8 kbps	460.8 kbps
UART output mode	Auto mode	Auto mode
Output physical quantity	Velocity RAW →	Velocity RMS
Output data format	TEMP2 + Velocity-XYZ	TEMP2 + Velocity-XYZ
RAW output data rate	3000 Sps (fixed)	3000 Sps (fixed)
RMS output data rate / update rate	1 s / 0.6826 s	1 s / 0.6826 s

1) 3000 Sps, Velocity RAW data

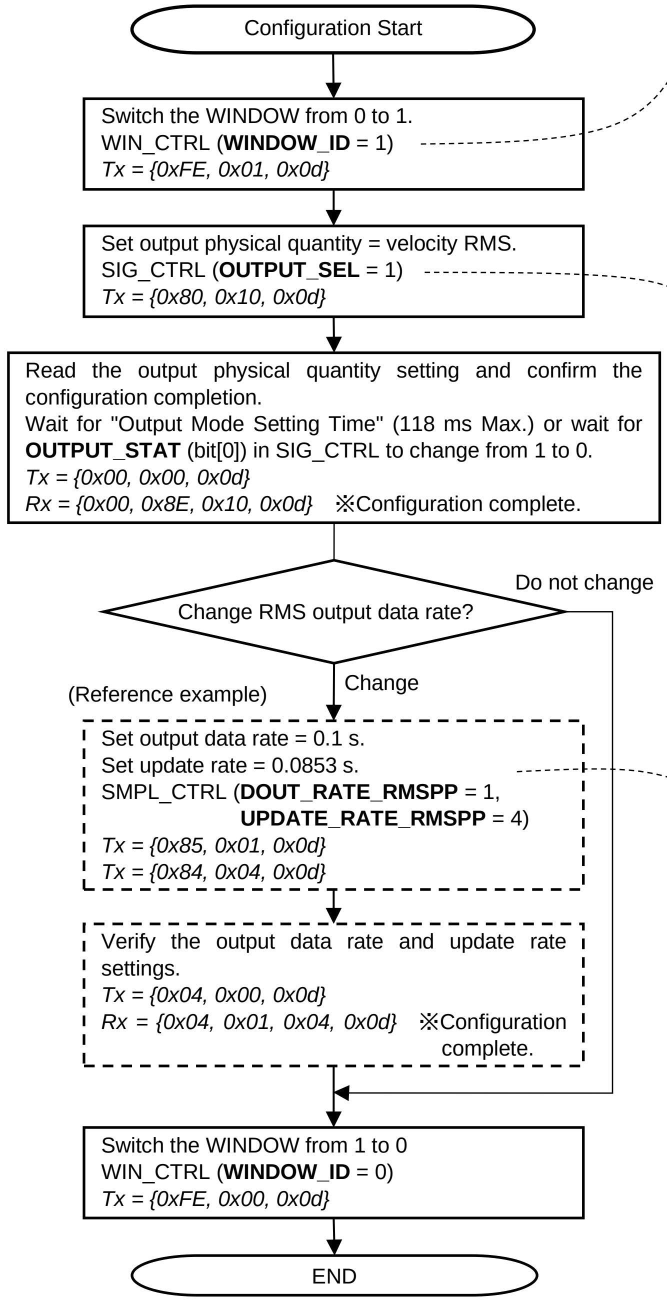
These configulations are the factory defaults that require no configuration changes. Measurement can start immediately. Go to "(Step 5) Data Acuisition".

2) 1 Sps, Velocity RMS data

How to change from the factory default settings to the velocity RMS settings is described below.

To change settings in principle, first set **WINDOW_ID** in the WIN_CTRL register to "1", then write the setting value to the register to be changed.

Figure 5.2 shows the configuration process flow for this setting (velocity RMS data). To change the output physical quantity to velocity RMS, set "velocity RMS: 0001" to OUTPUT_SEL. The output data rate and update rate can also be changed as needed. The procedure for reverting from velocity RMS to velocity RAW is the same as in this flow, except that "velocity RAW: 0000" is set to the OUTPUT_SEL.



WIN_CTRL Register

Addr (Hex)	Bit15	. . .	Bit8	R/W
0x7F		-		-

Addr (Hex)	Bit7	. . .	Bit0	R/W
0x7E		WINDOW_ID		R/W

bit [7:0] WINDOW_ID
Selects the desired register window by writing the window number to this register.
0x00: Window 0
0x01: Window 1
0x02 ~ 0xFF : Unused

SIG_CTRL Register

Addr (Hex)	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	R/W
0x01	ND_EN (Temp)	-	-	-	ND_EN (XVELC) (XDISP)	ND_EN (YVELC) (YDISP)	ND_EN (ZVELC) (ZDISP)	-	R/W

Addr (Hex)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	R/W
0x00	OUTPUT_SEL				0	-	TEMP _SEL	OUTPUT _STAT	R/W

bit [7:4] OUTPUT_SEL
Sets the output physical quantity of the XYZ axis sensor.
0000: Velocity RAW
0001: Velocity RMS
0010: Velocity P-P
0011: Reserved
0100: Displacement RAW
0101: Displacement RMS
0110: Displacement P-P
0111 - 1111: Reserved

bit [0] OUTPUT_STAT
Indicates the setting status of the output physical quantity **OUTPUT_SEL**. This bit is read-only.
1: Setting in progress
0: Setting complete

SMPL_CTRL Register

Addr (Hex)	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	R/W
0x05	DOUT_RATE_RMSPP								R/W

Addr (Hex)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	R/W
0x04	-	-	-	-	UPDATE_RATE_RMSPP				R/W

bit [15:8] DOUT_RATE_RMSPP
Sets the output rate for RMS and P-P data. The setting format is as follows.
Table 6.3 and 6.4 show setting examples.

Setting range n = 1 ~ 255
Output rate (s) Velocity: 0.1 × n
Displacement: 1 × n

bit [7:0] UPDATE_RATE_RMSPP
Sets the update rate for RMS and P-P data. The setting format is as follows.
Table 6.3 and 6.4 show setting examples.

Setting range n = 0 ~ 15
Update rate (s) Velocity: 1/3000 × 16 × 2^n
Displacement: 1/300 × 16 × 2^n

Figure 5.2 Configuration Process Flow (Velocity RMS data)

6. (Step 5) Data Acquisition

How to acquire sensor measurement data is described below.

The state transition diagram in Fugure 6.1 shows that writing "01" to **MODE_CMD** in the MODE_CTRL register shifts the state from the configuration mode to the sampling mode and starts measurement. Writing "10" to the **MODE_CMD** shifts the state from the sampling mode to the configuration mode and terminates the measurement. Writing to the MODE_CTRL register should be performed with **WINDOW_ID** = 0.

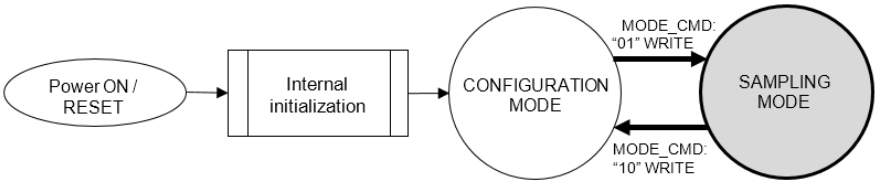


Figure 6.1 State Transition Diagram (Sampling Mode)

Measurement data can be acquired in the sampling mode. The UART Auto sampling mode is set as the factory default setting. As shown in Figure 6.2, measurement data is automatically transmitted at each sampling timing generated by the internal clock.

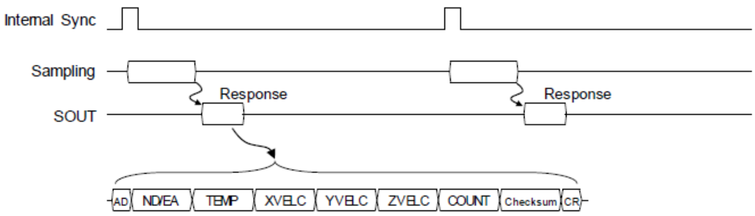


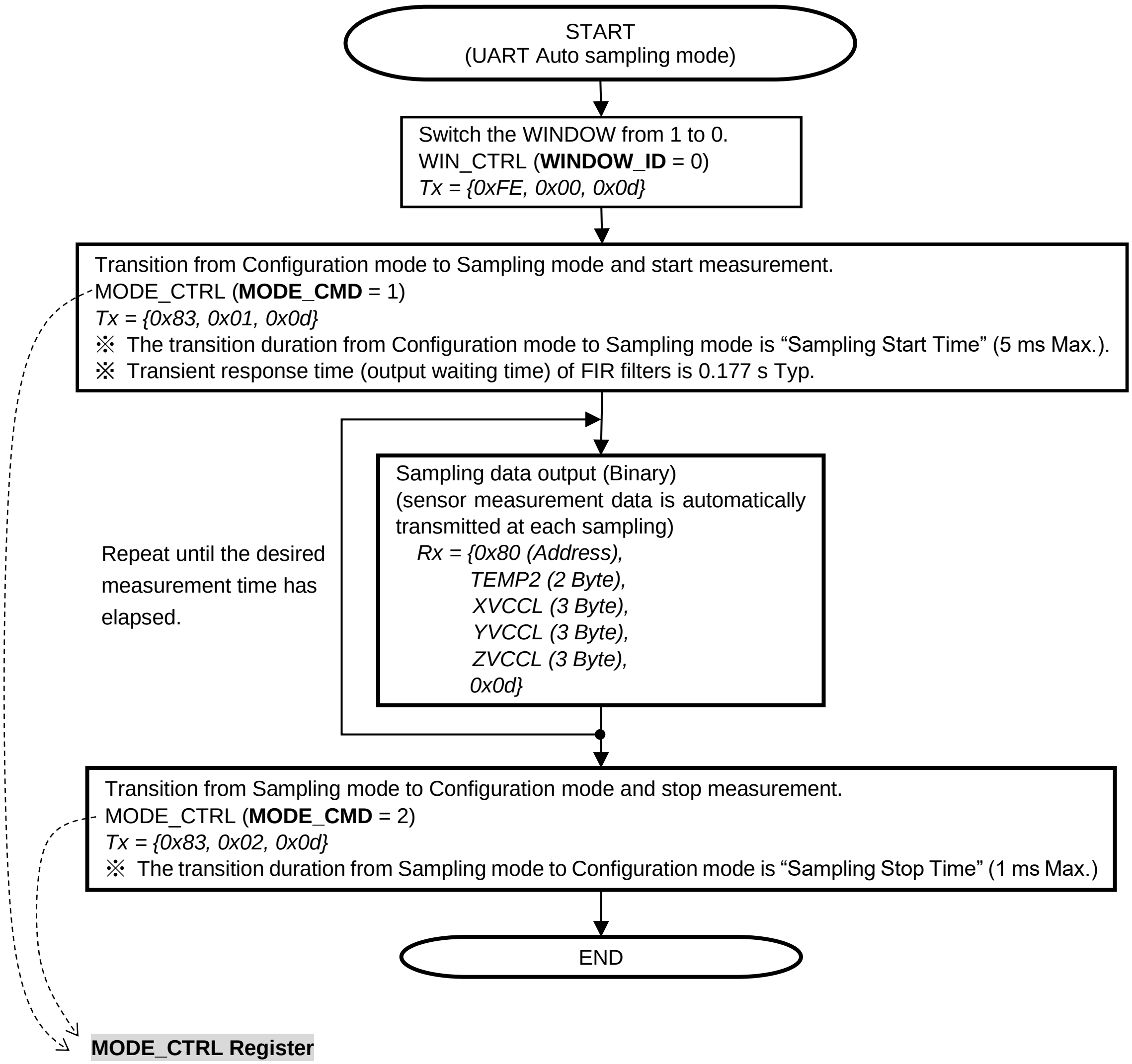
Figure 6.2 UART Auto Sampling Sequence

Table 6.1 Data Packet Format (Factory Default)

BURST_CTRL [0x0C(W1)] = 0x4700 (Burst Output: TEMP, XYZ)
SIG_CTRL [0x00(W1)] = 0x8E00 (Output Mode: Velocity, TEMP2)
Data packet size: 13 bytes

Byte No.	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	ADDRESS	0x80							
2	TEMP2_H	TEMP2 [15:8]							
3	TEMP2_L	TEMP2 [7:0]							
4	XVELC_HIGH_L	XVELC _HIGH [7:0]							
5	XVELC_LOW_H	XVELC_LOW [15:8]							
6	XVELC_LOW_L	XVELC _LOW [7:0]							
7	YVELC_HIGH_L	YVELC _HIGH [7:0]							
8	YVELC_LOW_H	YVELC _LOW [15:8]							
9	YVELC_LOW_L	YVELC _LOW [7:0]							
10	ZVELC_HIGH_L	ZVELC _HIGH [7:0]							
11	ZVELC_LOW_H	ZVELC _LOW [15:8]							
12	ZVELC_LOW_L	ZVELC _LOW [7:0]							
13	CR	0x0D							

Figure 6.3 shows the data acquisition process flow for UART auto sampling.



MODE_CTRL Register

Addr (Hex)	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	R/W
0x03	-	-	-	-	MODE_STAT		MODE_CMD		R/W *1

Addr (Hex)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	R/W
0x02	-	-	-	-	-	-	-	-	-

*1) MODE_STAT is read-only.

bit [11:10] MODE_STAT
This read-only status bit shows the current operation mode.
00: Sampling Mode
01: Configuration mode
10: Sleep Mode
11: (Not Used)

bit [9:8] MODE_CMD
Executes commands related to the operation mode.
00: Execute Complete.
01: Go to the Sampling Mode.
After the mode transition is completed, the bits automatically goes back to “00”.
10: Go to the Configuration Mode.
After the mode transition is completed, the bits automatically goes back to “00”.
11: Go to the Sleep Mode.
After the mode transition is completed, the bits automatically goes back to “00”.

Figure 6.3 Data Acquisition Process Flow for UART Auto Sampling.

7. (Step 6) Converting Acquired Data to Physical Quantity Units

Unit conversion procedures for the acquired data (binary data) are described below.

Based on the data acquired in Step 5, the meaning of each data and the calculation formulas to convert the data to physical quantity units are shown below. The calculation formulas are the same for velocity RAW data and velocity RMS. Refer to Figure 7.1 for TEMP2 register and XVELC/YVELC/ZVELC register specifications.

■ **Acquired data series (hexadecimal): 80 08 02 00 01 99 FF EC B3 00 25 5E 0D**

1）Address = 0x80

2）TEMP2 = 0x0802

2-1) 8BIT_TEMP = 0x08

Temperature Conversion Fomula: T (°C) = SF × a + 34.987
= -0.9707008 × 8 + 34.987 = 27.2213936

SF: Scale Factor = -0.9707008 (°C/LSB)
a: Acuired temperature data (decimal)

2-2) Flag information and 2-bit count value = 0x02

Structural resonance warning flags: bit [7:5] = 000: Measurement values are within normal range for all 3 axes.
Alarm detection flags: bit [4:2] = 000: No alarms are detected for any of the 3 axes.
※ Detected based on the threshold values stored in the ALARM registers.
2-bit count value: bit [1:0] = 2 ※ Incremented at each sampling (ex. 0 → 1 → 2 → 3 → 0 → ⋯).

3) VELC = 0x000199, 0xFFECB3, 0x00255E

Velocity Conversion Formula: (m/s) = a × SF

SF: Scale Factor = 2^{-22} ≒ 2.38×10^{-4} ((mm/s)/LSB)
a: Acuired velocity data (decimal)

3-1）XVELC = 0x000199 = 409
X axis velocity = 409 × 2^{-22} ≒ 0.098 (mm/s)

3-2）YVELC = 0xFFECB3 → -4941 ※ Since the number is negative ("sign" bit [23] = 1), 2's complement is used for the conversion.
Y axis velocity = -4941 × 2^{-22} ≒ -1.178 (mm/s)

3-3）ZVELC = 0x00255E = 9566
Z axis velocity = 9566 × 2^{-22} ≒ 2.281 (mm/s)

4）Delimiter = 0x0D

TEMP2 Register

Addr (Hex)	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	R/W
0x2F	8BIT_TEMP								R

Addr (Hex)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	R/W
0x2E	X_EXI _ERR	Y_EXI _ERR	Z_EXI _ERR	XALARM _ERR	YALARM _ERR	ZALARM _ERR	2BIT_COUNT		R

Note) **EXI_ERR** flags and **ALARM_ERR** flags are cleared to "0" by reading this register. These flags have the same function as those in the FLAG (ND/EA) register. However, the ND flag states are not linked and needs to be cleared separately.

bit [15:8] 8BIT_TEMP Temperature sensor output data (8 bits)
The internal temperature sensor value can be read from this register.
The output data format is 8-bit two's complement format.

Please refer to the below formula for conversion to temperature in centigrade. Please refer to Table 1.3 Sensor Specification for the scale factor value.
There is no guarantee that the value provides the absolute value of the internal temperature.

T_{8bit} (°C) = SF_{8bit} × a + 34.987
SF: Scale Factor
a: Temperature sensor output data (decimal)

bit [7] X_EXI_ERR
This bit indicates when the measured value of X-axis becomes abnormal due to structural resonance in the sensor. Until this bit is read, the upper limit of the measurement range is output as the sensor value instead of the measured value as a warning.
1: Measurement value is abnormal due to structural resonance
0: Measured value is within normal range

bit [6] Y_EXI_ERR
This bit indicates when the measured value of Y-axis becomes abnormal due to structural resonance in the sensor. Until this bit is read, the upper limit of the measurement range is output as the sensor value instead of the measured value as a warning.
1: Measurement value is abnormal due to structural resonance
0: Measured value is within normal range

bit [5] Z_EXI_ERR
This bit indicates when the measured value of Z-axis becomes abnormal due to structural resonance in the sensor. Until this bit is read, the upper limit of the measurement range is output as the sensor value instead of the measured value as a warning.
1: Measurement value is abnormal due to structural resonance
0: Measured value is within normal range

bit [4] XALARM_ERR
This bit indicates when the sensor value exceeds the value set in register: X_ALARM [0x47-0x46(W1)] in the X axis during measurement.
1: detection
0: no detection

bit [3] YALARM_ERR
This bit indicates when the sensor value exceeds the value set in register: Y_ALARM [0x49-0x48(W1)] in the Y axis during measurement.
1: detection
0: no detection

bit [2] ZALARM_ERR
This bit indicates when the sensor value exceeds the value set in register: Z_ALARM [0x4B-0x4A(W1)] in the Z axis during measurement.
1: detection
0: no detection

bit [1:0] 2BIT_COUNT
A 2-bit count value that counts up at each sampling count.

VELC Register

Addr (Hex)	Bit15	・・・	Bit8	Bit7	・・・	Bit0	R/W
0x30	-			XVELC_HIGH_L			R
0x32	XVELC_LOW_H			XVELC_LOW_L			R
0x34	-			YVELC_HIGH_L			R
0x36	YVELC_LOW_H			YVELC_LOW_L			R
0x38	-			ZVELC_HIGH_L			R
0x3A	ZVELC_LOW_H			ZVELC_LOW_L			R

bit [15:0] Velocity output data
These registers contain the 3-axis velocity data (RAW, RMS, P-P) for X, Y, and Z.

The output data format
Unit (m/s)
24-bit two's complement format
bit 23 : sign
bit 22 : integer
bit 21 ~ 0 : decimal

Figure 7.1 TEMP2 and VELC register specifications

8. Contact

AMERICA

EPSON AMERICA, INC.
214 Devcon Drive,
San Jose, CA 95112, USA
Phone: +1-800-228-3964 FAX: +1-408-922-0238

EUROPE

EPSON EUROPE ELECTRONICS GmbH
Riesstrasse 15, 80992 Munich,
GERMANY
Phone: +49-89-14005-0 FAX: +49-89-14005-110

ASIA

EPSON (CHINA) CO., LTD.
4F,Tower 1 of China Central Place,81 Jianguo Street, Chaoyang District, Beijing 100025 CHINA
Phone: +86-400-810-9972 X ext.2
Mail EPSON_MSM@ecc.epson.com.cm

EPSON SINGAPORE PTE. LTD.
438B Alexandra Road, Block B Alexandra TechnoPark, #04-01/04, Singapore 119968
Phone: +65-6586-5500 FAX: +65-6271-3182

EPSON TAIWAN TECHNOLOGY & TRADING LTD.
14F, No. 7, Song Ren Road,
Taipei 110, TAIWAN
Phone: +886-2-8786-6688 Fax: +886-2-8786-6660

EPSON KOREA Co., Ltd.
10F Posco Tower Yeoksam, Teheranro 134 Gangnam-gu,
Seoul, 06235 KOREA
Phone: +82-2-558-4270 Fax: +82-2-3420-6699

JAPAN

SEIKO EPSON CORPORATION.
DEVICE SALES & MARKETING DEPT.
29th Floor, JR Shinjuku Miraina Tower, 4-1-6 Shinjuku, Shinjuku-ku, Tokyo, 160-8801, Japan
Phone: +81-3-6682-4322 FAX: +81-3-6682-5016

Product Information on www server
https://global.epson.com/products_and_drivers/sensing_system/