

EA201 - EXERCÍCIO PRÁTICO 4

Vinícius Esperança Mantovani, 247395

Introdução:

Neste exercício, fomos incumbidos de estudar os sinais da comunicação entre uma MCU e um acelerômetro do componente LSM303DLHC. Para tanto, analisamos capturas desses sinais por meio do programa “Saleae Logic”, estudando ainda o datasheet do produto para que fosse possível entender seu funcionamento e, conseqüentemente, os sinais que nos foram dispostos. Desse modo, pudemos praticar os conhecimentos de I2C e, mais uma vez, exercitar habilidades de ler, compreender e extrair informações de datasheets de produtos.

Item 1)

Conforme o que se lê logo no começo da seção 5.1.1 do datasheet, é possível notar que após o processo de *START* da transmissão, o que se segue são 8 bits cujos 7 primeiros são o endereço (*ID*) do componente *slave* e, o último bit determina se o mestre está deseja enviar ou receber dados do slave na transmissão. Sendo assim, usando o analisador, conforme as imagens 1 e 2 a seguir, temos o que se segue após tais imagens:

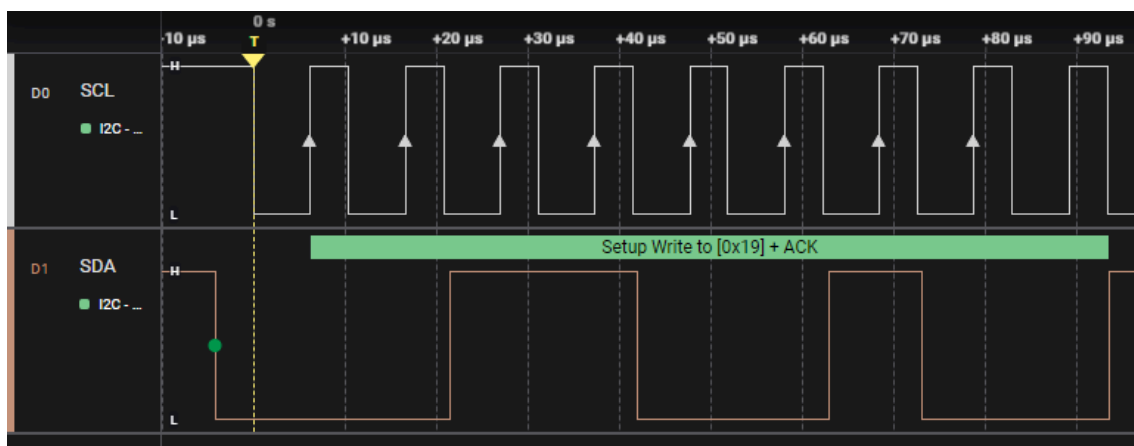


Imagem 1

	Type	Start	Duration	ack	address	read	data
■	start	-4.083 μs	42 ns				
■	address	6.167 μs	87.083 μs	true	0x19	false	
■	data	100.292 μs	91.792 μs	true			0x20
■	data	198.25 μs	87.042 μs	true			0x37

Imagem 2

Considerando o que se afirma no enunciado deste Item como uma observação de que no analisador o “address” é um valor de 7 bits, dessas imagens, nota-se que o endereço *ID* do componente é dado conforme se apresenta abaixo:

$$0x19 = 0b0011001 \rightarrow ID = 0b0011001$$

Ademais, do datasheet, temos que o *ID* de fábrica do produto é $ID_{fábrica} = 0b0011001$. Logo, podemos ver que o *ID* encontrado por meio do analisador é o mesmo que o *ID* de fábrica do acelerômetro.

Item 2)

As imagens usadas neste item estão dispostas abaixo, sendo elas as imagens 3 e 4:

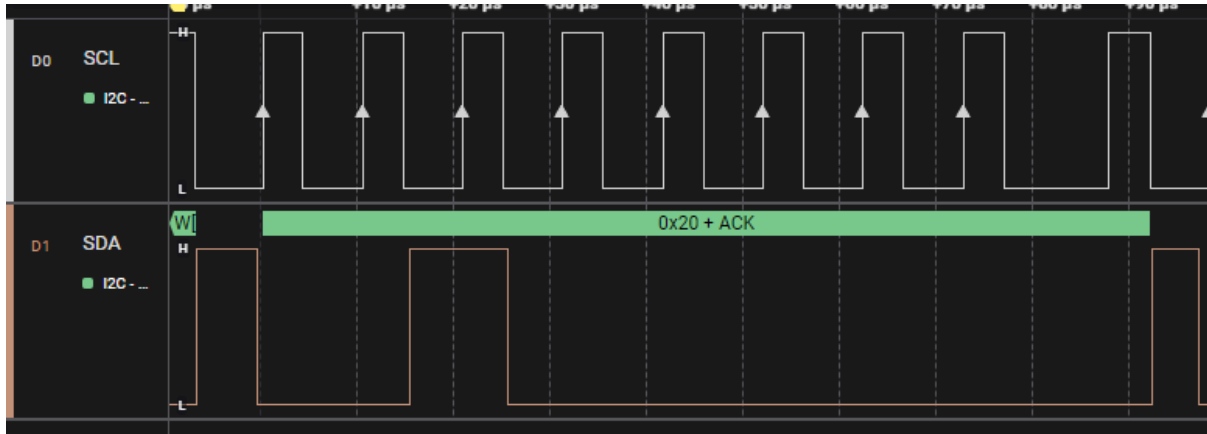


Imagem 3

■	data	100.292 μ s	91.792 μ s	true			0x20
■	data	198.25 μ s	87.042 μ s	true			0x37
■	stop	295.958 μ s	42 ns				

Imagem 4

De acordo com o datasheet do componente, os 7 bits menos significativos do conjunto de dados transferidos após o endereço do *slave* compõem o endereço do registrador a ser escrito e o bit mais significativo indica se o auto-incremento do endereço do registrador está ativo ou não. Desse modo, podemos concluir que o endereço do primeiro registrador é:

$$0x20 = 0b00100000 \rightarrow \text{sub-endereço (endereço do registrador)} = 0b0100000 = 0x20.$$

Além disso, analisando a imagem abaixo, pode-se notar que o registrador em questão é chamado de “CTRL_REG1_A”, cujo endereço é 0x20:

Table 17. Register address map

Name	Slave address	Type	Register address		Default	Comment
			Hex	Binary		
Reserved (do not modify)	Table 14		00 - 1F	--	--	Reserved
CTRL_REG1_A	Table 14	rw	20	010 0000	00000111	
CTRL_REG2_A	Table 14	rw	21	010 0001	00000000	
CTRL_REG3_A	Table 14	rw	22	010 0010	00000000	
CTRL_REG4_A	Table 14	rw	23	010 0011	00000000	
CTRL_REG5_A	Table 14	rw	24	010 0100	00000000	
CTRL_REG6_A	Table 14	rw	25	010 0101	00000000	
REFERENCE_A	Table 14	rw	26	010 0110	00000000	
STATUS_REG_A	Table 14	r	27	010 0111	00000000	
OUT_X_L_A	Table 14	r	28	010 1000	output	
OUT_X_H_A	Table 14	r	29	010 1001	output	
OUT_Y_L_A	Table 14	r	2A	010 1010	output	

Imagem 5

Seguindo, observando a imagem 6 abaixo, percebe-se que os valores dos bits do registrador após o componente ser ligado ou sofrer um refresh são:

1. ODR[3:0] = 0b0000;
2. LPen = 0;
3. Zen = 1;
4. Yen = 1;
5. Xen = 1;

7.1.1 CTRL_REG1_A (20h)

Table 18. CTRL_REG1_A register

ODR3	ODR2	ODR1	ODR0	LPen	Zen	Yen	Xen
------	------	------	------	------	-----	-----	-----

Table 19. CTRL_REG1_A description

ODR[3:0]	Data rate selection. Default value: 0000 (0000: power-down, others: refer to Table 20)
LPen	Low-power mode enable. Default value: 0 (0: normal mode, 1: low-power mode)
Zen	Z-axis enable. Default value: 1 (0: Z-axis disabled, 1: Z-axis enabled)
Yen	Y-axis enable. Default value: 1 (0: Y-axis disabled, 1: Y-axis enabled)
Xen	X-axis enable. Default value: 1 (0: X-axis disabled, 1: X-axis enabled)

Imagem 6

Da maneira como se apresenta acima, tem-se o valor inicial do registrador completo da seguinte forma:

$$CTRL_REG1_A = 0b00000111$$

Analisando, ainda, a imagem 6, pode-se concluir que os parâmetros determinados por esses componentes do registrador e seus valores de início são:

1. ODR[3:0] → Determina a frequência da captura de dados pelo componente, de acordo com a tabela da imagem 7 → inicialmente em modo *power-down*, sem captura;
2. LPen → Determina se o componente deve operar em baixo uso de energia (*Low-power mode*) ou não → inicialmente em modo normal, não em *Low-power* (influencia a frequência de captura também, conforme se nota na imagem 7);
3. Zen → Determina se o eixo Z deve estar habilitado → inicialmente habilitado;
4. Yen → Determina se o eixo Y deve estar habilitado → inicialmente habilitado;
5. Xen → Determina se o eixo X deve estar habilitado → inicialmente habilitado.

Table 20. Data rate configuration

ODR3	ODR2	ODR1	ODR0	Power mode and ODR selection
0	0	0	0	Power-down mode
0	0	0	1	Normal / low-power mode (1 Hz)
0	0	1	0	Normal / low-power mode (10 Hz)
0	0	1	1	Normal / low-power mode (25 Hz)
0	1	0	0	Normal / low-power mode (50 Hz)
0	1	0	1	Normal / low-power mode (100 Hz)
0	1	1	0	Normal / low-power mode (200 Hz)
0	1	1	1	Normal / low-power mode (400 Hz)
1	0	0	0	Low-power mode (1.620 kHz)
1	0	0	1	Normal (1.344 kHz) / low-power mode (5.376 kHz)

Imagem 7

Analisando as informações apresentadas anteriormente, nota-se que não é possível realizar medições sem que uma primeira escrita seja feita a esse registrador. Isso porque, por padrão (inicialmente, após o componente ser ligado ou sofrer reset), os componentes ODR[3:0] são nulos, o que implica que o sistema está em modo *Power-down* e, por consequência, que não há captura alguma de dados pelo acelerômetro.

Por fim, observando a imagem 4, vê-se que foi escrito no registrador *CTRL_REG1_A* o valor 0x37, ou seja, 0b00110111, o que implica os seguintes valores aos componentes:

1. ODR[3:0] = 0b0011;
2. LPen = 0;
3. Zen = 1;
4. Yen = 1;

5. Xen = 1;

Do modo exposto acima, temos agora um valor não nulo no conjunto ODR[3:0], de modo que a captura de dados pelo acelerômetro passa, portanto, a ser feita a uma frequência de 25Hz, como se observa na imagem 7. Para além disso, cabe destacar que não houveram alterações a outros componentes do registrador em questão nesta escrita, apenas aos ODR[3:0].

Item 3)

No caso do segundo registrador a ser escrito, temos o endereço 0x21 conforme se nota na imagem abaixo e, conseqüentemente, de acordo com a imagem 5, sabemos que o nome do registrador em questão é “CTRL_REG2_A”.

■	start	322.375 µs	42 ns				
■	address	332.667 µs	87.042 µs	true	0x19	false	
■	data	426.75 µs	91.792 µs	true			0x21
■	data	524.708 µs	87.083 µs	true			0x00
■	stop	622.458 µs	42 ns				

Imagem 8

Para além do exposto, sabemos, ainda da imagem acima, que o valor escrito no registrador em questão foi 0x00 = 0b00000000, o que pode ser interpretado com auxílio da imagem abaixo:

Table 22. CTRL_REG2_A description

HPM[1:0]	High-pass filter mode selection. Default value: 00 (refer to Table 23)
HPCF[2:1]	High-pass filter cutoff frequency selection
FDS	Filtered data selection. Default value: 0 (0: internal filter bypassed, 1: data from internal filter sent to output register and FIFO)
HPCLICK	High-pass filter enabled for click function (0: filter bypassed, 1: filter enabled)
HPIS2	High-pass filter enabled for AOI function on Interrupt 2 (0: filter bypassed, 1: filter enabled)
HPIS1	High-pass filter enabled for AOI function on Interrupt 1 (0: filter bypassed, 1: filter enabled)

Imagem 9

Dos dados acima, conclui-se que:

1. HPM[1:0] = 0b00 → Modo normal de filtragem passa-alta selecionado (imagem 10);
2. HPCF[2:1] = 0b00 → Frequência de corte do filtro selecionada em 0b00;
3. FDS = 0b0 → Filtro interno desviado, não há envio de dados do filtro interno nem ao registrador de saída nem à *FIFO*;

4. HPCLICK = 0b0 → Filtro passa-alta desabilitado (desviado) para função de *click*;
5. HPIS2 = 0b0 → Filtro passa-alta desabilitado (desviado) para função AOI na interrupção 2;
6. HPIS1 = 0b0 → Filtro passa-alta desabilitado (desviado) para função AOI na interrupção 1;

Table 23. High-pass filter mode configuration

HPM1	HPM0	High-pass filter mode
0	0	Normal mode (reset reading HP_RESET_FILTER)
0	1	Reference signal for filtering
1	0	Normal mode
1	1	Autoreset on interrupt event

Imagem 10

Analisando o que foi exposto acima, conclui-se que este registrador configura as operações de filtragem do acelerômetro.

Já no caso do terceiro registrador a ser escrito, tem-se o endereço 0x22 conforme a imagem abaixo e, conseqüentemente, de acordo com a imagem 5, sabemos que o nome do registrador em questão é “CTRL_REG3_A”.

■	start	648.833 μs	42 ns				
■	address	659.125 μs	87.083 μs	true	0x19	false	
■	data	753.25 μs	91.792 μs	true			0x22
■	data	851.208 μs	87.042 μs	true			0x10
■	stop	948.875 μs	42 ns				

Imagem 11

Para além do exposto, sabemos, ainda da imagem acima, que o valor escrito no registrador em questão foi 0x10 = 0b00010000, o que pode ser interpretado com auxílio da imagem abaixo:

Table 25. CTRL_REG3_A description

I1_CLICK	CLICK interrupt on INT1. Default value 0 (0: disable, 1: enable)
I1_AOI1	AOI1 interrupt on INT1. Default value 0 (0: disable, 1: enable)
I1_AOI2	AOI2 interrupt on INT1. Default value 0 (0: disable, 1: enable)
I1_DRDY1	DRDY1 interrupt on INT1. Default value 0 (0: disable, 1: enable)
I1_DRDY2	DRDY2 interrupt on INT1. Default value 0 (0: disable, 1: enable)
I1_WTM	FIFO watermark interrupt on INT1. Default value 0 (0: disable, 1: enable)
I1_OVERRUN	FIFO overrun interrupt on INT1. Default value 0 (0: disable, 1: enable)

Imagem 12

Dos dados acima, conclui-se que:

1. I1_CLICK = 0b0 → Interrupção *CLICK* em *INT1* desabilitada;
2. I1_AOI1 = 0b0 → Interrupção *AOI1* em *INT1* desabilitada;
3. I1_AOI2 = 0b0 → Interrupção *AOI2* em *INT1* desabilitada;
4. I1_DRDY1 = 0b1 → Interrupção *DRDY1* habilitada em *INT1*;
5. I1_DRDY2 = 0b0 → Interrupção *DRDY2* habilitada em *INT1*;
6. I1_WTM = 0b0 → Interrupção de marca d'água *FIFO* desabilitada em *INT1*;
7. I1_OVERRUN = 0b0 → Interrupção de *overrun FIFO* desabilitada em *INT1*.

Assim, pode-se concluir que este registrador é responsável pela configuração das interrupções no pino de interrupção 1, o pino *INT1*.

Seguindo, para o terceiro registrador, temos o endereço 0x23, conforme abaixo, de modo que, ao se analisar a imagem 5 novamente, sabe-se que o nome deste registrador é “CTRL_REG4_A”.

■	start	978.667 μs	42 ns				
■	address	988.958 μs	87.042 μs	true	0x19	false	
■	data	1.083042 ms	91.792 μs	true			0x23
■	data	1.181 ms	87.042 μs	true			0x00
■	stop	1.278708 ms	42 ns				

Imagem 13

Além disso, ainda da imagem acima, tem-se que o valor escrito neste registrador é 0x00 = 0b00000000, implicando, da imagem 14, o seguinte:

1. BDU = 0b0 → Bloco de dados continuamente atualizado;

2. BLE = 0b0 → Bit menos significativo indica menor endereço;
3. FS[1:0] = 0b00 → Full-scale = ± 2 g;
4. HR = 0b0 → Modo de alta resolução desabilitado para o *output*;
5. SIM = 0b0 → Modo *4-wire interface* selecionado para a interface serial SPI.

Table 27. CTRL_REG4_A description

BDU	Block data update. Default value: 0 (0: continuous update, 1: output registers not updated until MSB and LSB have been read)
BLE	Big/little endian data selection. Default value 0. (0: data LSB @ lower address, 1: data MSB @ lower address)
FS[1:0]	Full-scale selection. Default value: 00 (00: ± 2 g, 01: ± 4 g, 10: ± 8 g, 11: ± 16 g)
HR	High-resolution output mode: Default value: 0 (0: high-resolution disable, 1: high-resolution enable)
SIM	SPI serial interface mode selection. Default value: 0 (0: 4-wire interface, 1: 3-wire interface).

Imagem 14

Concluindo, percebe-se que este registrador é responsável, no geral, por algumas configurações relacionadas aos dados obtidos do acelerômetro.

Finalmente, temos o endereço do registrador sob perspectiva do *Logic* apresentado na imagem 15. Este registrador tem endereço 0x24, cujo nome pode ser extraído da imagem 5 e é dado por “CTRL_REG5_A”.

■	start	1.305125 ms	42 ns				
■	address	1.315375 ms	87.083 μ s	true	0x19	false	
■	data	1.4095 ms	91.792 μ s	true			0x24
■	data	1.507458 ms	87.042 μ s	true			0x00
■	stop	1.605167 ms	42 ns				

Imagem 15

Ademais, seguindo análise da imagem acima, mais uma vez, temos a escrita de 0x00 em um registrador, neste caso ao *CTRL_REG5_A*, implicando o que se segue em comparação com a imagem 16:

1. BOOT = 0b0 → Reinício de conteúdos da memória desabilitado, modo normal ativado;
2. FIFO_EN = 0b0 → FIFO desabilitada;
3. LIR_INT1 = 0b0 → Requisição de interrupção não *latched*, *INT1_SRC* não se limpa lendo a si próprio;
4. D4D_INT1 = 0b0 → Detecção de 4 dimensões desabilitada em *INT1*.
5. LIR_INT2 = 0b0 → Requisição de interrupção não *latched*, *INT2_SRC* não se limpa lendo a si próprio.
6. D4D_INT1 = 0b0 → Detecção de 4 dimensões desabilitada em *INT2*.

Table 29. CTRL_REG5_A description

BOOT	Reboot memory content. Default value: 0 (0: normal mode, 1: reboot memory content)
FIFO_EN	FIFO enable. Default value: 0 (0: FIFO disable, 1: FIFO enable)
LIR_INT1	Latch interrupt request on INT1_SRC register, with INT1_SRC register cleared by reading INT1_SRC itself. Default value: 0. (0: interrupt request not latched, 1: interrupt request latched)
D4D_INT1	4D enable: 4D detection is enabled on INT1 when 6D bit on INT1_CFG is set to 1.
LIR_INT2	Latch interrupt request on INT2_SRC register, with INT2_SRC register cleared by reading INT2_SRC itself. Default value: 0. (0: interrupt request not latched, 1: interrupt request latched)
D4D_INT2	4D enable: 4D detection is enabled on INT2 when 6D bit on INT2_CFG is set to 1.

Imagem 16

Por fim, este último registrador controla ainda algumas questões relacionadas ao conteúdo da memória, à habilitação da FIFO e aos pinos de interrupção.

Item 4)

Conforme explicado no datasheet, na sequência de um endereço de *slave* passado, ocorre o envio de um Byte, cujos 7 bits menos significativos representam o sub-endereço, o endereço do registrador a ser escrito e, o bit mais significativo controla o auto-incremento do sub-endereço para, neste caso, leitura subsequente. Sendo assim, temos que o endereço do primeiro registrador é 0x28, pois $0xA8 = 0b10101000 \rightarrow \text{end_reg} = 0b0101000 = 0x28$. Além disso, embora já soubéssemos do enunciado que a leitura é sequencial e que, por consequência, o bit controlador do auto-incremento do sub-endereço é 1, podemos notar de 0xA8 que o bit mais significativo é, de fato, 1.

Sendo assim, seguem os registradores lidos, na ordem em que o são:

1. 0x28 → nome: *OUT_X_L_A*;
2. 0x29 → nome: *OUT_X_H_A*;
3. 0x2A → nome: *OUT_Y_L_A*;
4. 0x2B → nome: *OUT_Y_H_A*;
5. 0x2C → nome: *OUT_Z_L_A*;
6. 0x2D → nome: *OUT_Z_H_A*.

Pode-se notar, portanto, que os registradores lidos são os registradores de *output* das medidas de cada um dos eixos. Sabe-se ainda, das Seções 7.1.9 a 7.1.11 do datasheet, que os valores são expressos em complemento de dois.

Item 5)

Conforme definido no campo *BLE* do registrador *CTRL_REG4_A*, o Bit menos significativo está inserido no menor endereço, logo, baseando-se na imagem 17, pode-se prosseguir para encontrar os valores decimais de cada um dos eixos da seguinte maneira:

- ❖ Determinação do valor no eixo X:
 - 0x28 → data: 0xC0 = 0b11000000;
 - 0x29 → data: 0xFE = 0b11111110;
 - Juntando: 0x29|0x28 = 0b1111111011000000 = -320;

- ❖ Determinação do valor no eixo Y:
 - 0x2A → data: 0x80 = 0b10000000;
 - 0x2B → data: 0xF8 = 0b11111000;
 - Juntando: 0x2B|0x2A = 0b1111100010000000 = -1920;
- ❖ Determinação do valor no eixo Z:
 - 0x2C → data: 0x40 = 0b01000000;
 - 0x2D → data: 0x3B = 0b00111011;
 - Juntando: 0x2D|0x2C = 0b0011101101000000 = 15168.

Type	Start	Duration	ack	address	read	data
start	-4.083 µs	42 ns				
address	6.167 µs	87.083 µs	true	0x19	false	
data	100.292 µs	87.083 µs	true			0xA8
start	204.208 µs	42 ns				
address	214.458 µs	87.083 µs	true	0x19	true	
data	307.708 µs	87.083 µs	true			0xC0
data	400.917 µs	87.083 µs	true			0xFE
data	494.167 µs	87.042 µs	true			0x80
data	587.375 µs	87.083 µs	true			0xF8
data	680.625 µs	87.042 µs	true			0x40
data	773.833 µs	87.042 µs	false			0x3B
stop	871.542 µs	42 ns				

Imagem 17

Item 6)

No caso acima, o eixo predominante é o eixo Z, pois tem o maior valor armazenado no output em base decimal. Sendo tal valor 15168, nota-se que o eixo está apontando para baixo, uma vez que tem valor positivo e, por consequência, gravidade positiva.

Item 7)

Sigamos para a análise dos arquivos *Ex04_pos2.sal* e *Ex04_pos3.sal*. Começando pelo primeiro, abaixo é exposta a imagem do analisador, expondo os valores de cada um dos registradores:

Type	Start	Duration	ack	address	read	data
start	-4.125 µs	42 ns				
address	6.125 µs	87.125 µs	true	0x19	false	
data	100.167 µs	87.083 µs	true			0xA8
start	204.125 µs	42 ns				
address	214.417 µs	87.083 µs	true	0x19	true	
data	307.667 µs	87.125 µs	true			0x40
data	400.917 µs	87.125 µs	true			0x42
data	494.208 µs	87.125 µs	true			0x80
data	587.458 µs	87.125 µs	true			0xFE
data	680.75 µs	87.125 µs	true			0x00
data	774.042 µs	87.083 µs	false			0x01
stop	871.792 µs	42 ns				

Imagem 18

Observando a imagem e repetindo o procedimento do item 5, tem-se:

- ❖ Determinação do valor no eixo X:
 - 0x28 → data: 0x40 = 0b01000000;
 - 0x29 → data: 0x42 = 0b01000010;
 - Juntando: 0x29|0x28 = 0b0100001001000000 = 16960;
- ❖ Determinação do valor no eixo Y:
 - 0x2A → data: 0x80 = 0b10000000;
 - 0x2B → data: 0xFE = 0b11111110;
 - Juntando: 0x2B|0x2A = 0b1111111010000000 = -384;
- ❖ Determinação do valor no eixo Z:
 - 0x2C → data: 0x00 = 0b00000000;
 - 0x2D → data: 0x01 = 0b00000001;
 - Juntando: 0x2D|0x2C = 0b0000000100000000 = 256.

Logo, para o segundo arquivo, o eixo predominante é o eixo X, cujo valor decimal é 16960 e, por consequência, seu sentido é coincidente com a gravidade positiva, sendo, por este motivo, um valor positivo.

Por fim, analisando *Ex04_pos3.sal*, tem-se os valores expostos na seguinte imagem:

	Type	Start	Duration	ack	address	read	data
■	start	-4.125 µs	42 ns				
■	address	6.167 µs	87.042 µs	true	0x19	false	
■	data	100.25 µs	87.042 µs	true			0xA8
■	start	204.125 µs	42 ns				
■	address	214.417 µs	87.042 µs	true	0x19	true	
■	data	307.583 µs	87.042 µs	true			0x40
■	data	400.792 µs	87.042 µs	true			0xFD
■	data	494 µs	87.042 µs	true			0x40
■	data	587.208 µs	87.042 µs	true			0xC3
■	data	680.375 µs	87.083 µs	true			0x40
■	data	773.583 µs	87.042 µs	false			0xFC
■	stop	871.292 µs	42 ns				

Imagem 19

Disso, tira-se:

- ❖ Determinação do valor no eixo X:
 - 0x28 → data: 0x40 = 0b01000000;
 - 0x29 → data: 0xFD = 0b11111101;
 - Juntando: 0x29|0x28 = 0b1111110101000000 = -704;
- ❖ Determinação do valor no eixo Y:
 - 0x2A → data: 0x40 = 0b01000000;
 - 0x2B → data: 0xC3 = 0b11000011;

- Juntando: $0x2B|0x2A = 0b1100001101000000 = -15552$;
- ❖ Determinação do valor no eixo Z:
 - $0x28 \rightarrow \text{data: } 0x40 = 0b010000000$;
 - $0x2C \rightarrow \text{data: } 0xFC = 0b11111100$;
 - Juntando: $0x2D|0x2C = 0b1111110001000000 = -960$.

Portanto, neste último arquivo, o eixo predominante é o eixo Y que, diferentemente dos casos anteriores, está voltado para cima, no sentido de gravidade negativa.