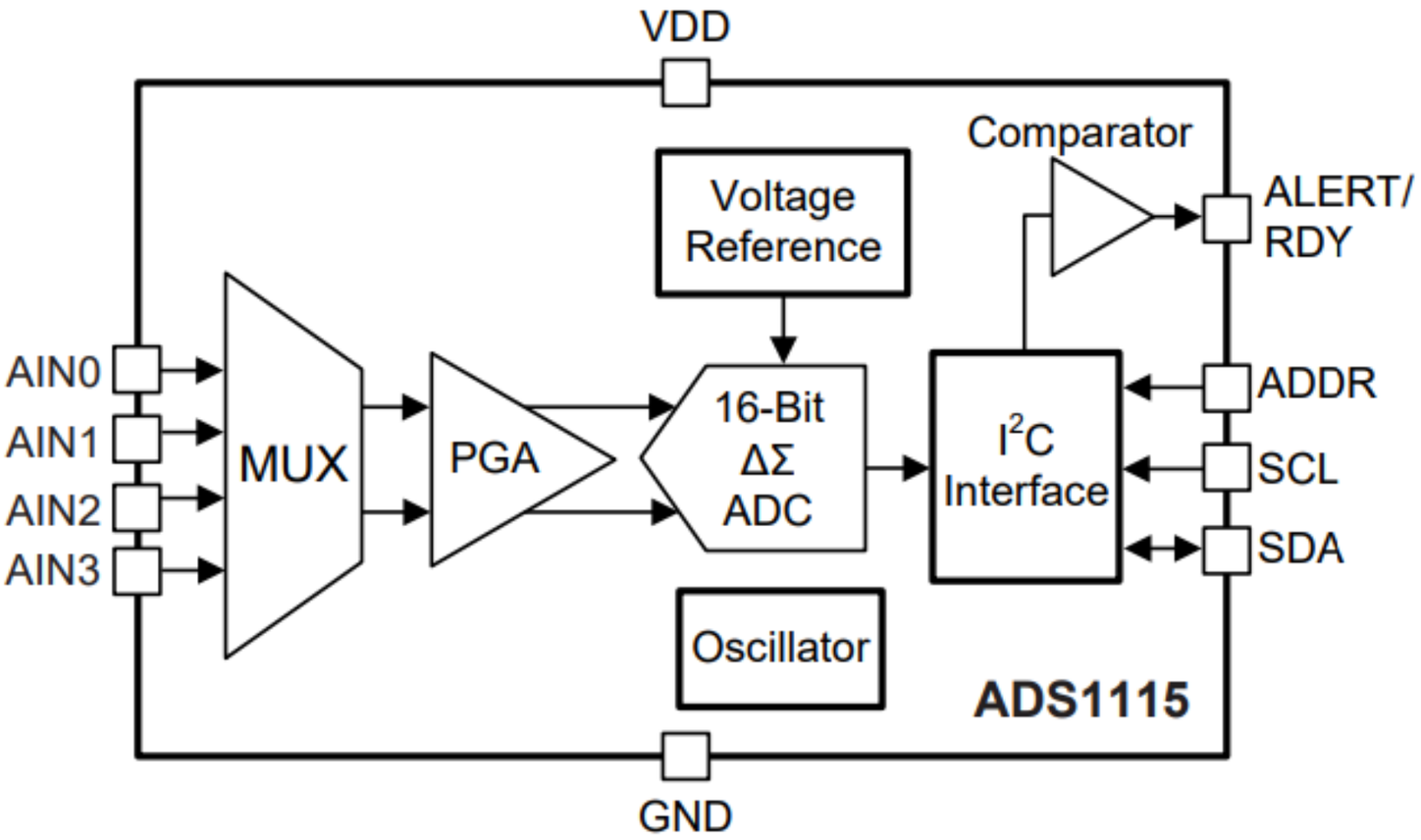
Basics of I2C:
An I2C Example

TIPL 6102
TI Precision Labs – Digital Communications

Prepared by Joseph Wu

Presented by Alex Smith

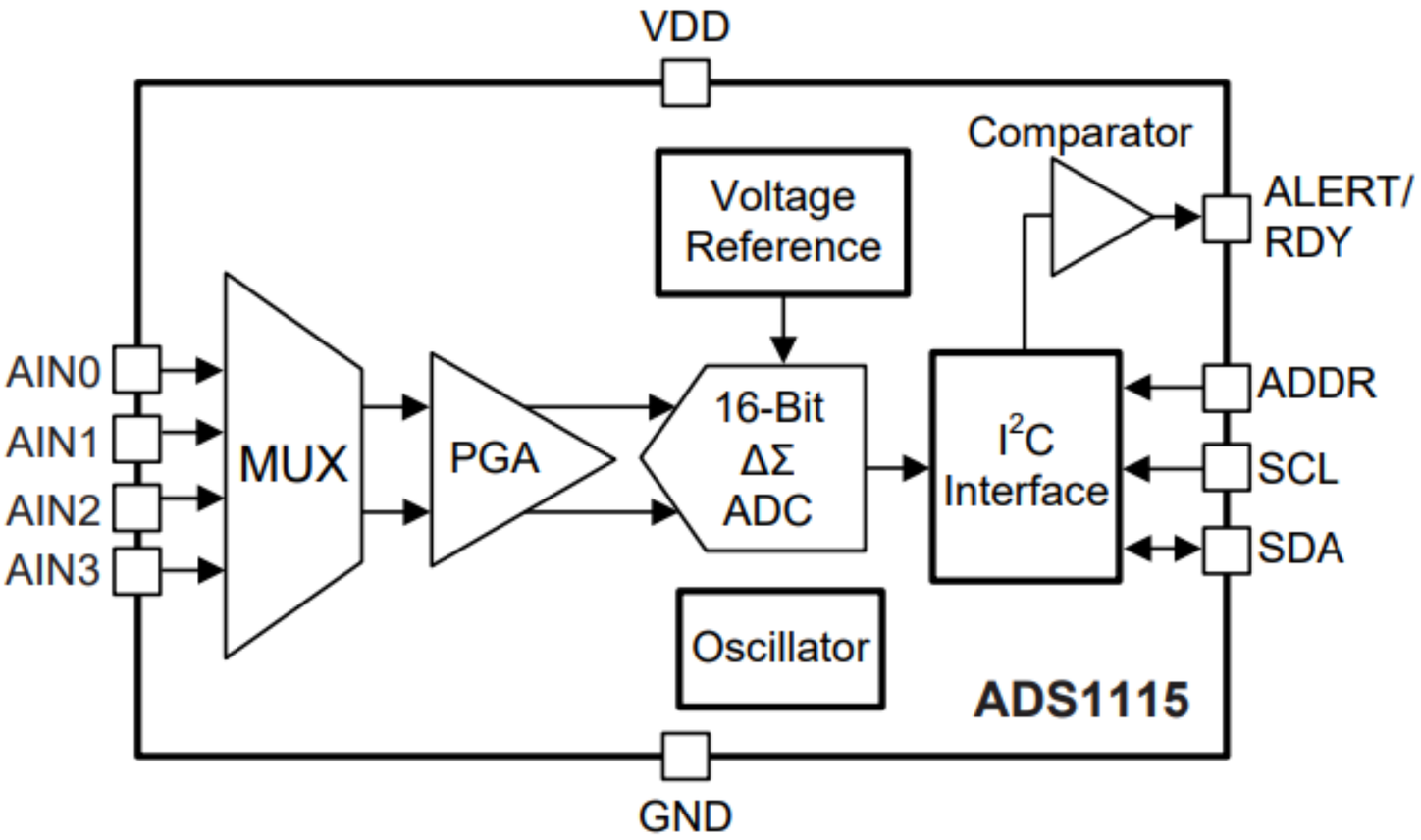
I2C Example: ADS1115



I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

ADDR Pin	I2C Address
GND	100 1000 (48h)
VDD	100 1001 (49h)
SDA	100 1010 (4Ah)
SCL	100 1011 (4Bh)

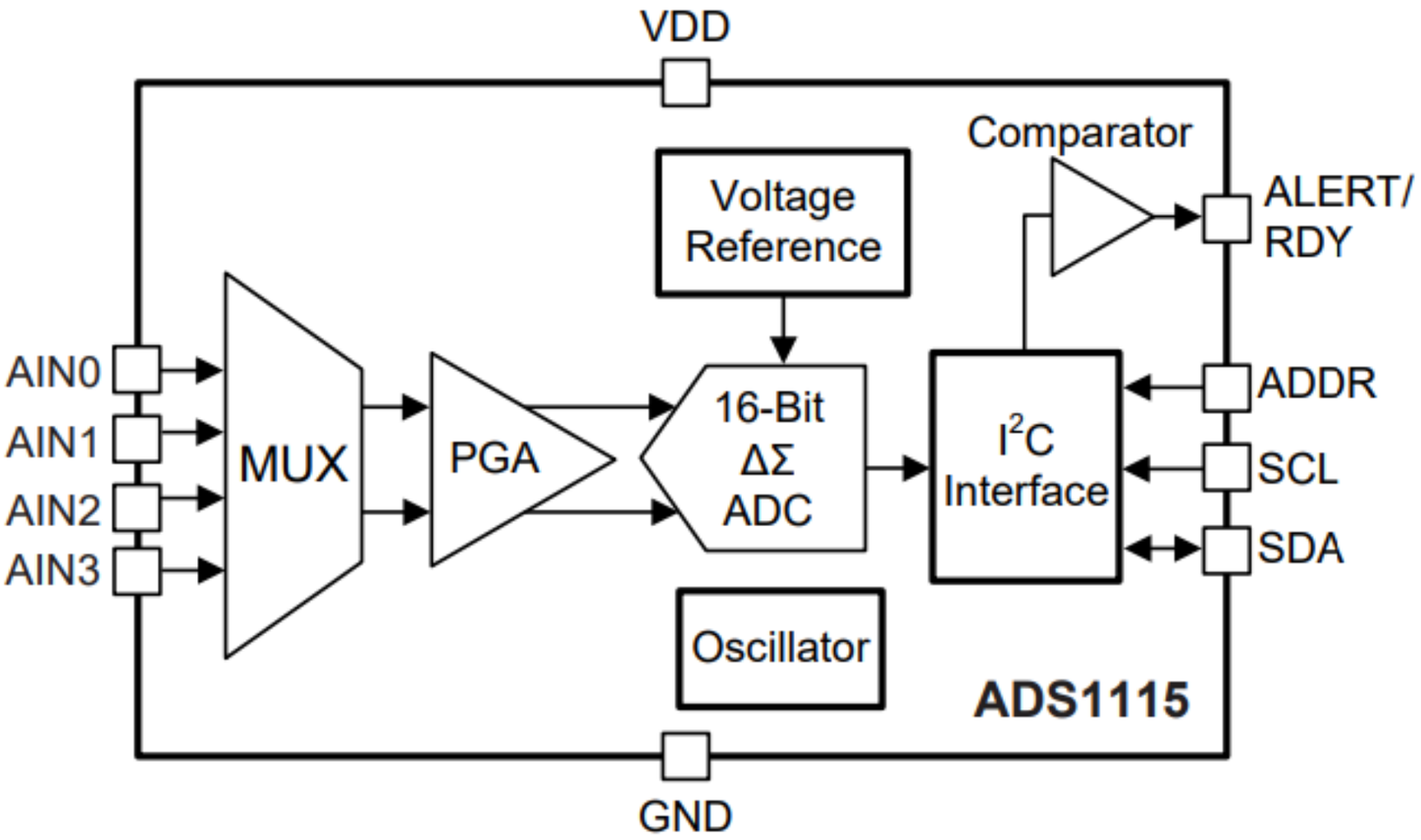
I2C Example: ADS1115



I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

Pointer	Register
00	Conversion data
01	Configuration
10	Lo_threshold register
11	Hi_threshold register

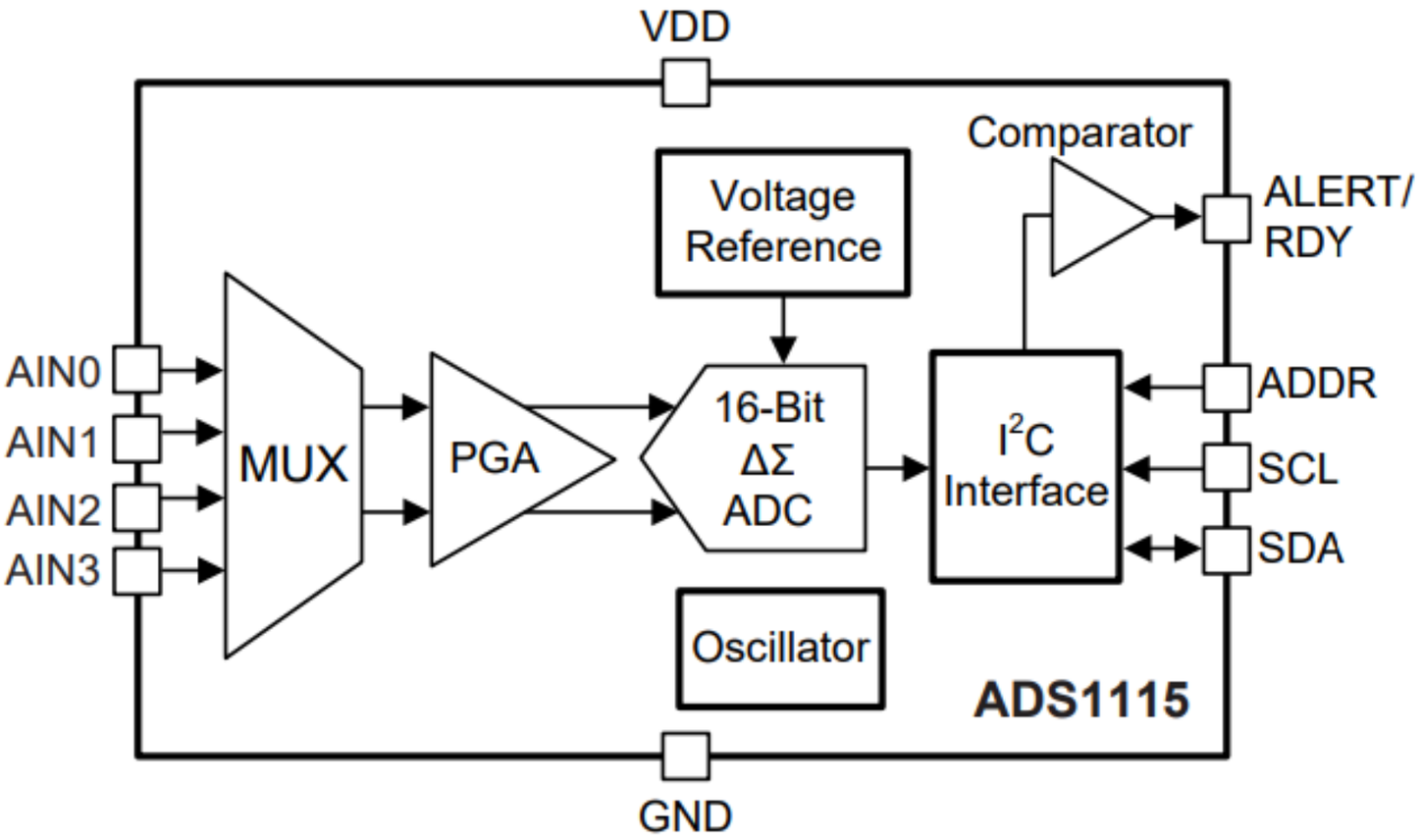
I2C Example: ADS1115



I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

Pointer	Register
00	Conversion data
01	Configuration
10	Lo_threshold register
11	Hi_threshold register

I2C Example: ADS1115



I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

Pointer	Register
00	Conversion data
01	Configuration
10	Lo_threshold register
11	Hi_threshold register

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
15	OS	R/W	1h	**Operational status or single-shot conversion start** This bit determines the operational status of the device. OS can only be written when in power-down state and has no effect when a conversion is ongoing. When writing: 0 : No effect 1 : Start a single conversion (when in power-down state) When reading: 0 : Device is currently performing a conversion 1 : Device is not currently performing a conversion

Configuration Register

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
14:12	MUX[2:0]	R/W	0h	**Input multiplexer configuration (ADS1115 only)** These bits configure the input multiplexer. These bits serve no function on the ADS1113 and ADS1114. 000 : AIN_P = AIN0 and AIN_N = AIN1 (default) 001 : AIN_P = AIN0 and AIN_N = AIN3 010 : AIN_P = AIN1 and AIN_N = AIN3 011 : AIN_P = AIN2 and AIN_N = AIN3 100 : AIN_P = AIN0 and AIN_N = GND 101 : AIN_P = AIN1 and AIN_N = GND 110 : AIN_P = AIN2 and AIN_N = GND 111 : AIN_P = AIN3 and AIN_N = GND

Configuration Register

1	1	0	0												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
11:9	PGA[2:0]	R/W	2h	**Programmable gain amplifier configuration** These bits set the FSR of the programmable gain amplifier. These bits serve no function on the ADS1113. 000 : FSR = ±6.144 V(1) 001 : FSR = ±4.096 V(1) 010 : FSR = ±2.048 V (default) 011 : FSR = ±1.024 V 100 : FSR = ±0.512 V 101 : FSR = ±0.256 V 110 : FSR = ±0.256 V 111 : FSR = ±0.256 V

Configuration Register

1	1	0	0	0	0	1									
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
8	MODE	R/W	1h	**Device operating mode** This bit controls the operating mode. 0 : Continuous-conversion mode 1 : Single-shot mode or power-down state (default)

Configuration Register

1	1	0	0	0	0	1	1								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	DR[2:0]	R/W	4h	**Data rate** These bits control the data rate setting. 000 : 8 SPS 001 : 16 SPS 010 : 32 SPS 011 : 64 SPS 100 : 128 SPS (default) 101 : 250 SPS 110 : 475 SPS 111 : 860 SPS

Configuration Register

1	1	0	0	0	0	1	1	1	1	1					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
1:0	COMP_QUE[1:0]	R/W	3h	**Comparator queue and disable (ADS1114 and ADS1115 only)** These bits perform two functions. When set to 11, the comparator is disabled and the ALERT/RDY pin is set to a high-impedance state. When set to any other value, the ALERT/RDY pin and the comparator function are enabled, and the set value determines the number of successive conversions exceeding the upper or lower threshold required before asserting the ALERT/RDY pin. These bits serve no function on the ADS1113. 00 : Assert after one conversion 01 : Assert after two conversions 10 : Assert after four conversions 11 : Disable comparator and set ALERT/RDY pin to high-impedance (default)

Configuration Register

1	1	0	0	0	0	1	1	1	1	1	0	0	0	1	1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

I2C Example: Write to the Configuration Register

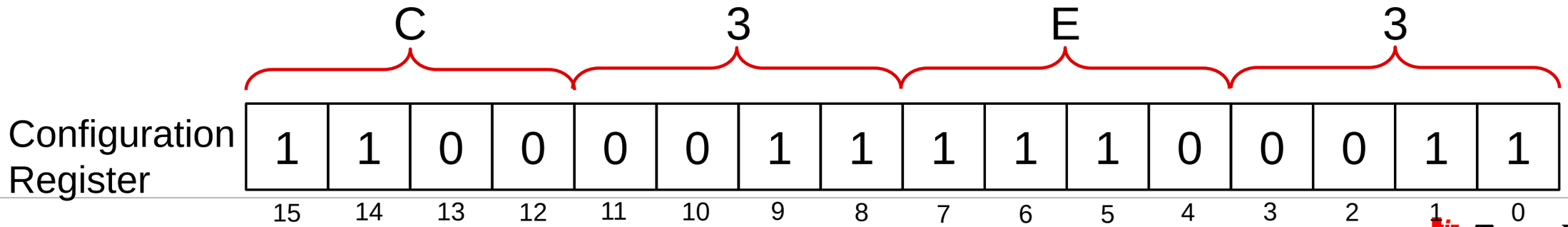
Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description

Configuration Register can also be written as C3E3 in hexadecimal



I2C Example: Write to the Configuration Register

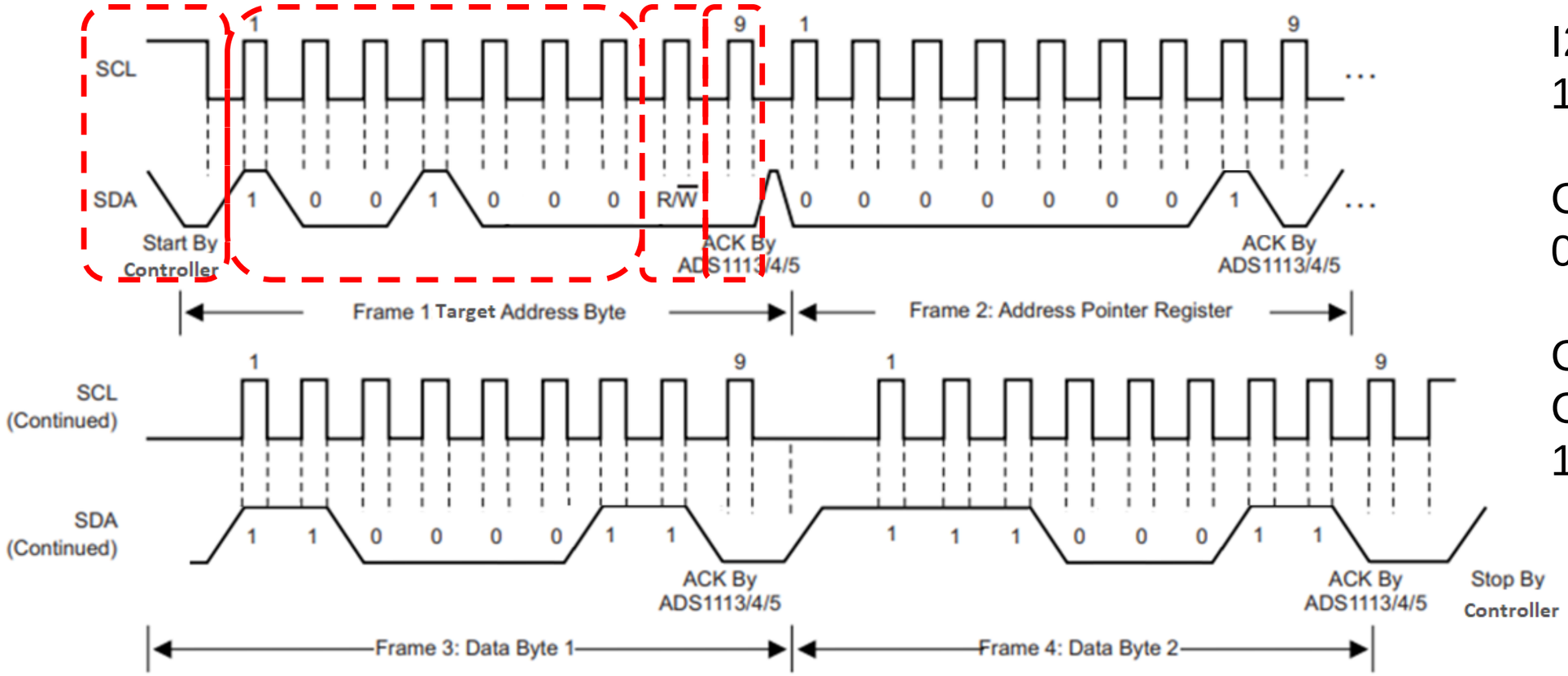


Figure 31. Timing Diagram for Writing to ADS111x

I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

START, Address 48h, Write to device, ACK by ADS1115

I2C Example: Write to the Configuration Register

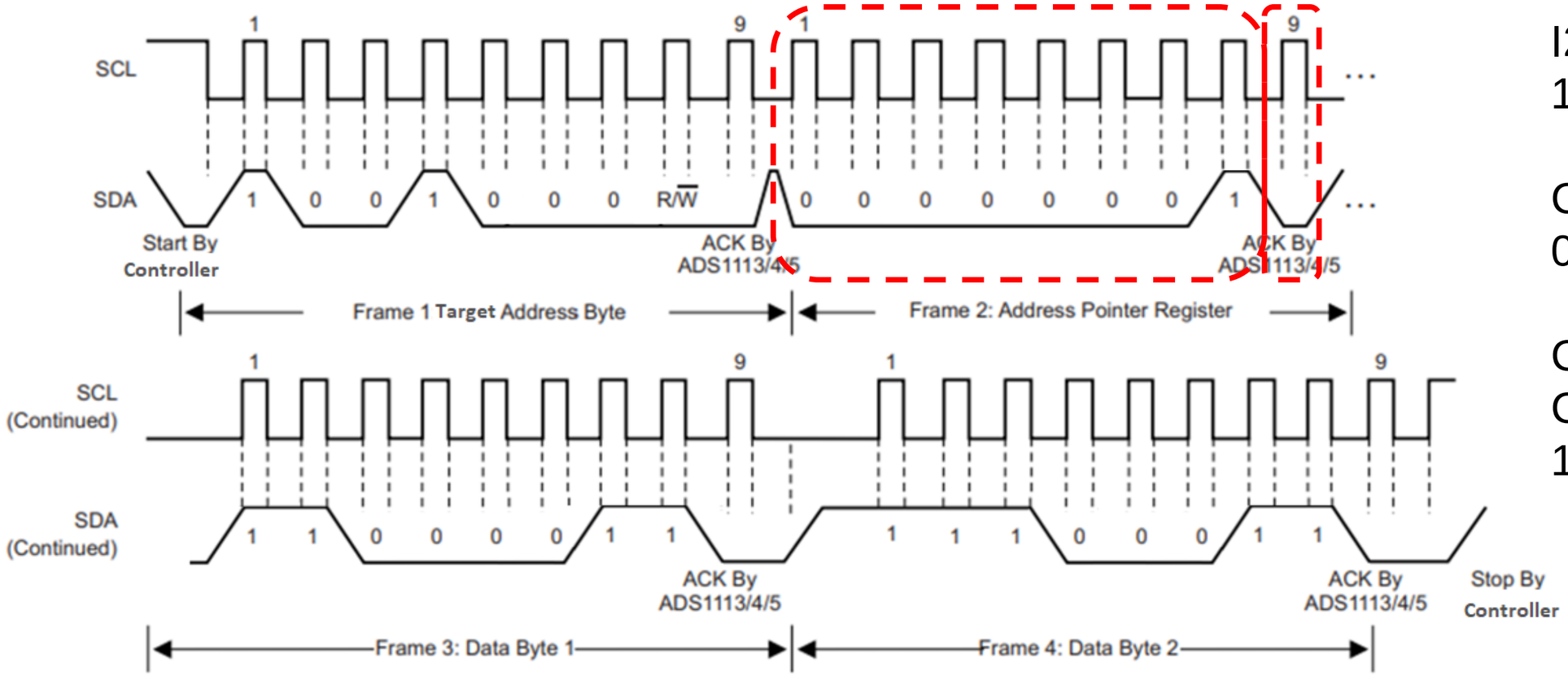


Figure 31. Timing Diagram for Writing to ADS111x

I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Send 01h to device for configuration register, ACK by ADS1115

I2C Example: Write to the Configuration Register

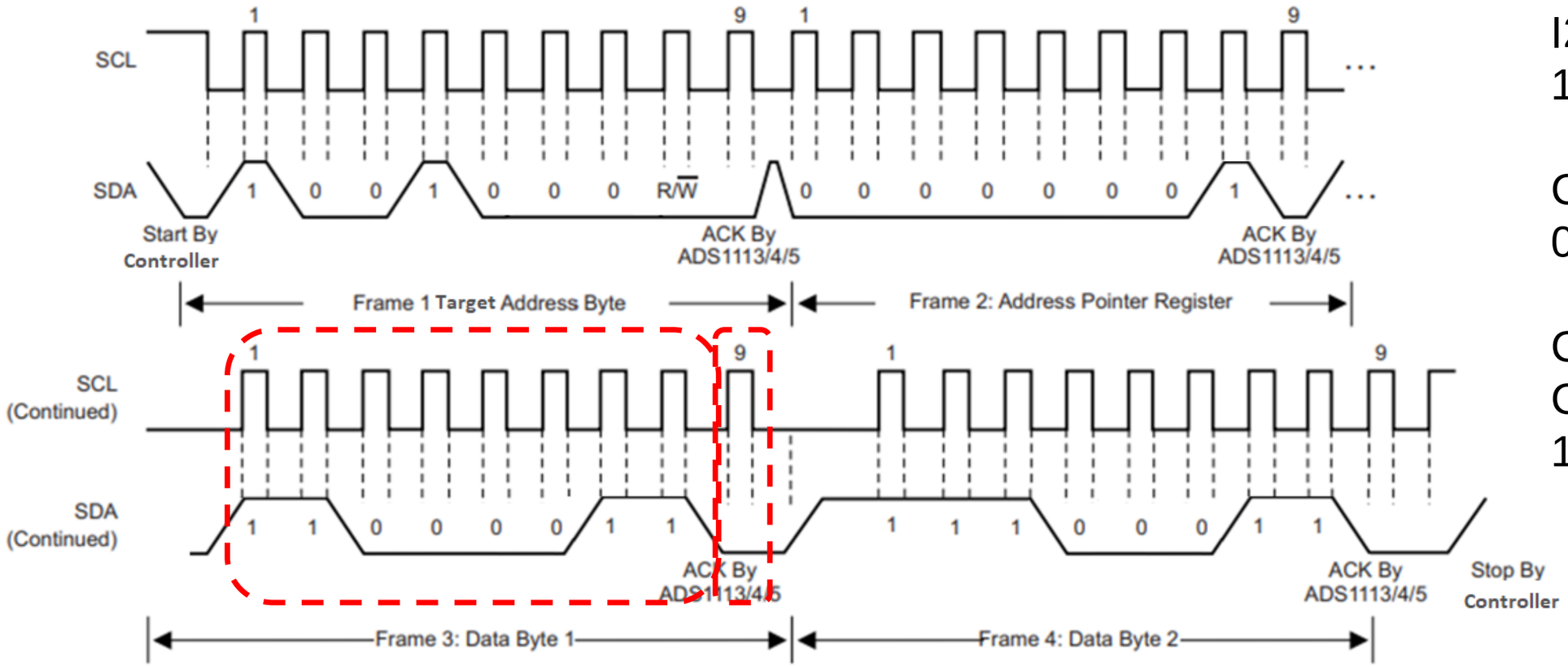


Figure 31. Timing Diagram for Writing to ADS111x

I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Send first configuration byte to device C3h, ACK by ADS1115

I2C Example: Write to the Configuration Register

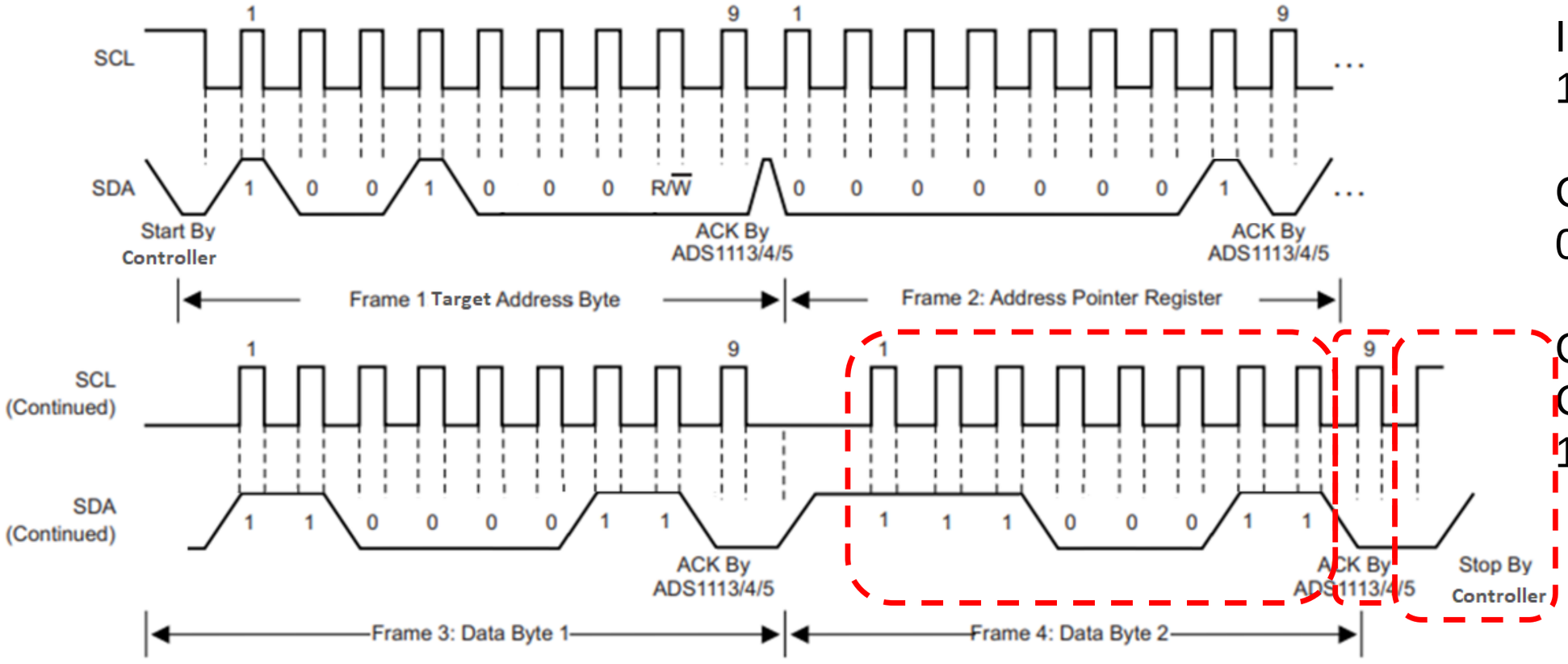


Figure 31. Timing Diagram for Writing to ADS111x

I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Send second configuration byte to device E3h, ACK by ADS1115, STOP

I2C Example: Write to the Configuration Register

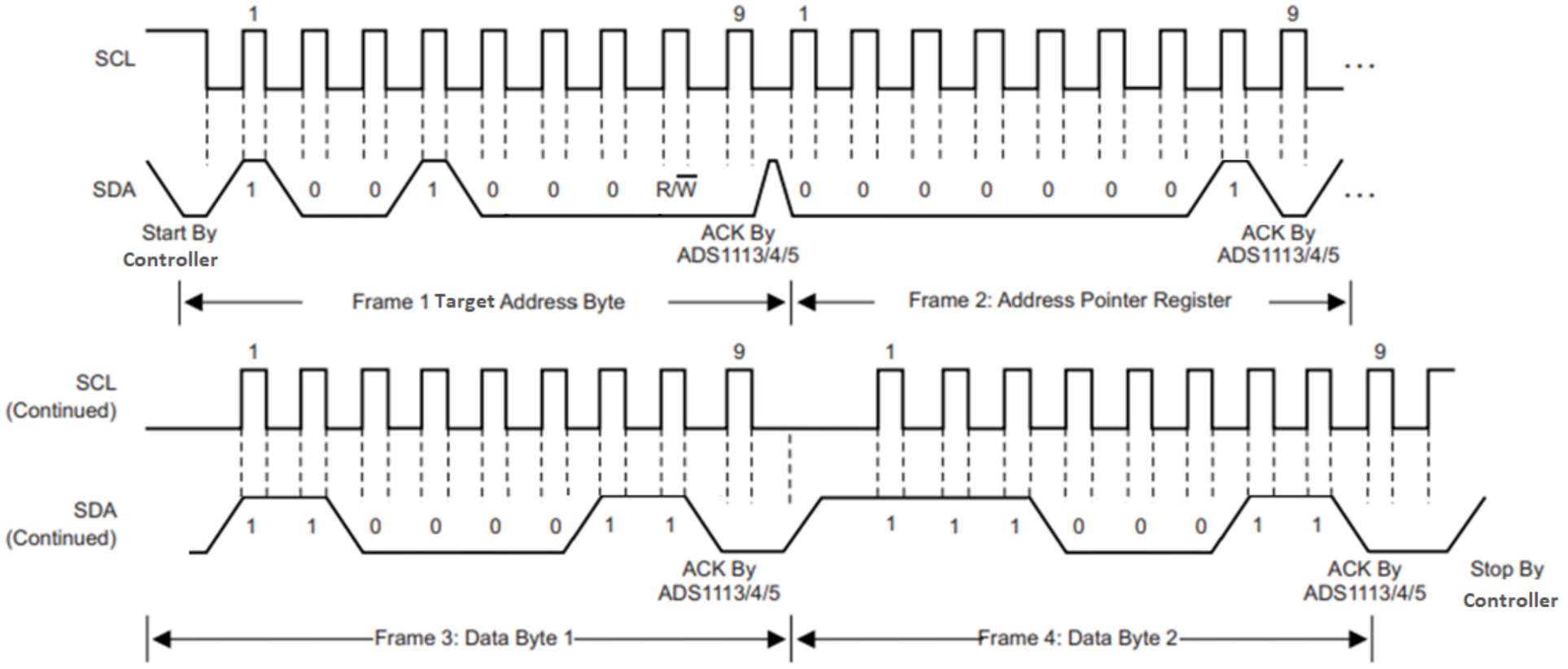


Figure 31. Timing Diagram for Writing to ADS111x

I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

I2C Example: Reading from the Conversion Register

Figure 35. Conversion Register

15	14	13	12	11	10	9	8
D15	D14	D13	D12	D11	D10	D9	D8
R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h
7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0
R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h

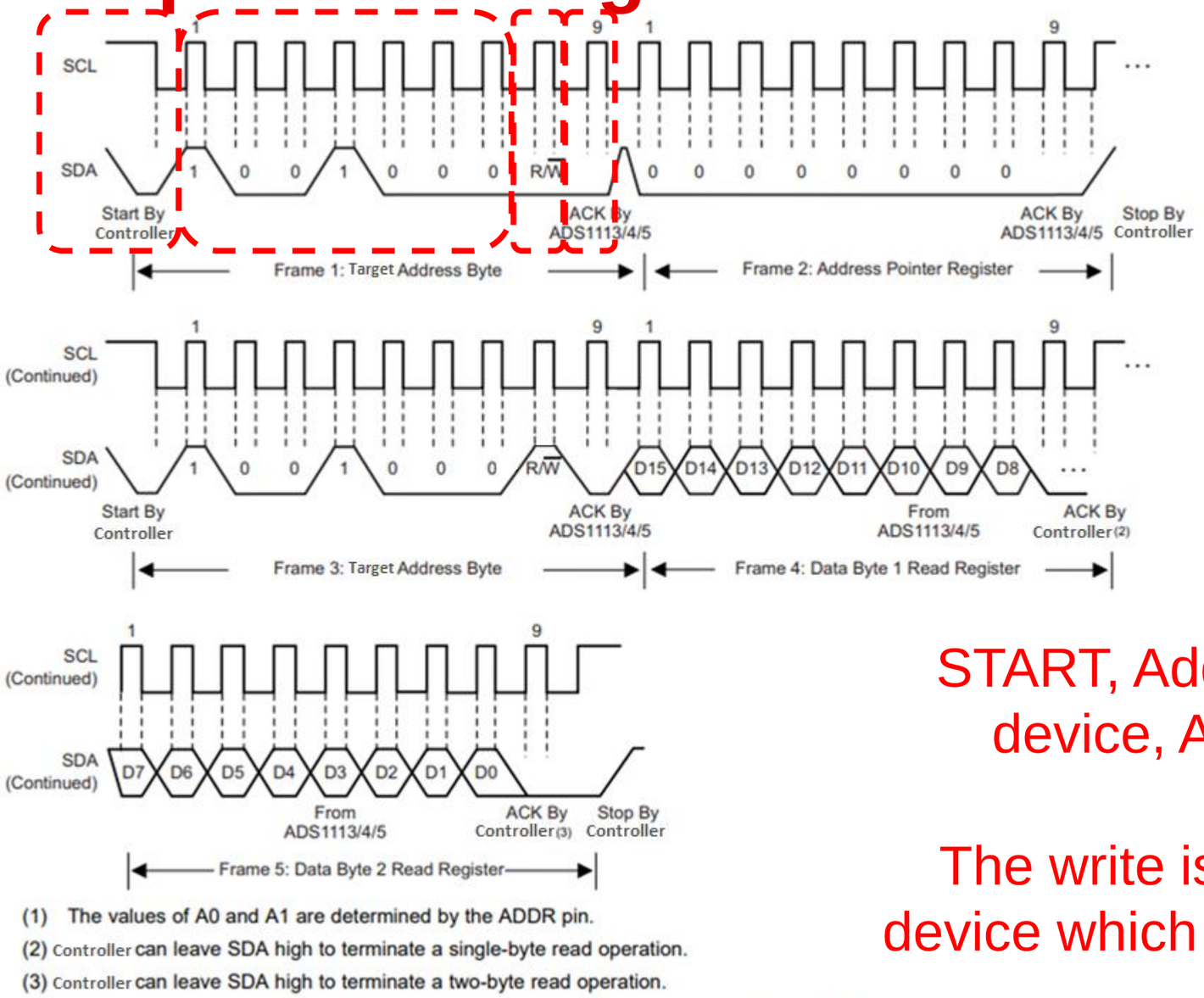
LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 7. Conversion Register Field Descriptions

Bit	Field	Type	Reset	Description
15:0	D[15:0]	R	0000h	16-bit conversion result

Conversion Register pointer: 0000 0000
Conversion data is 16 bits with MSB first

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.

(2) Controller can leave SDA high to terminate a single-byte read operation.

(3) Controller can leave SDA high to terminate a two-byte read operation.

Figure 30. Timing Diagram for Reading From ADS111x

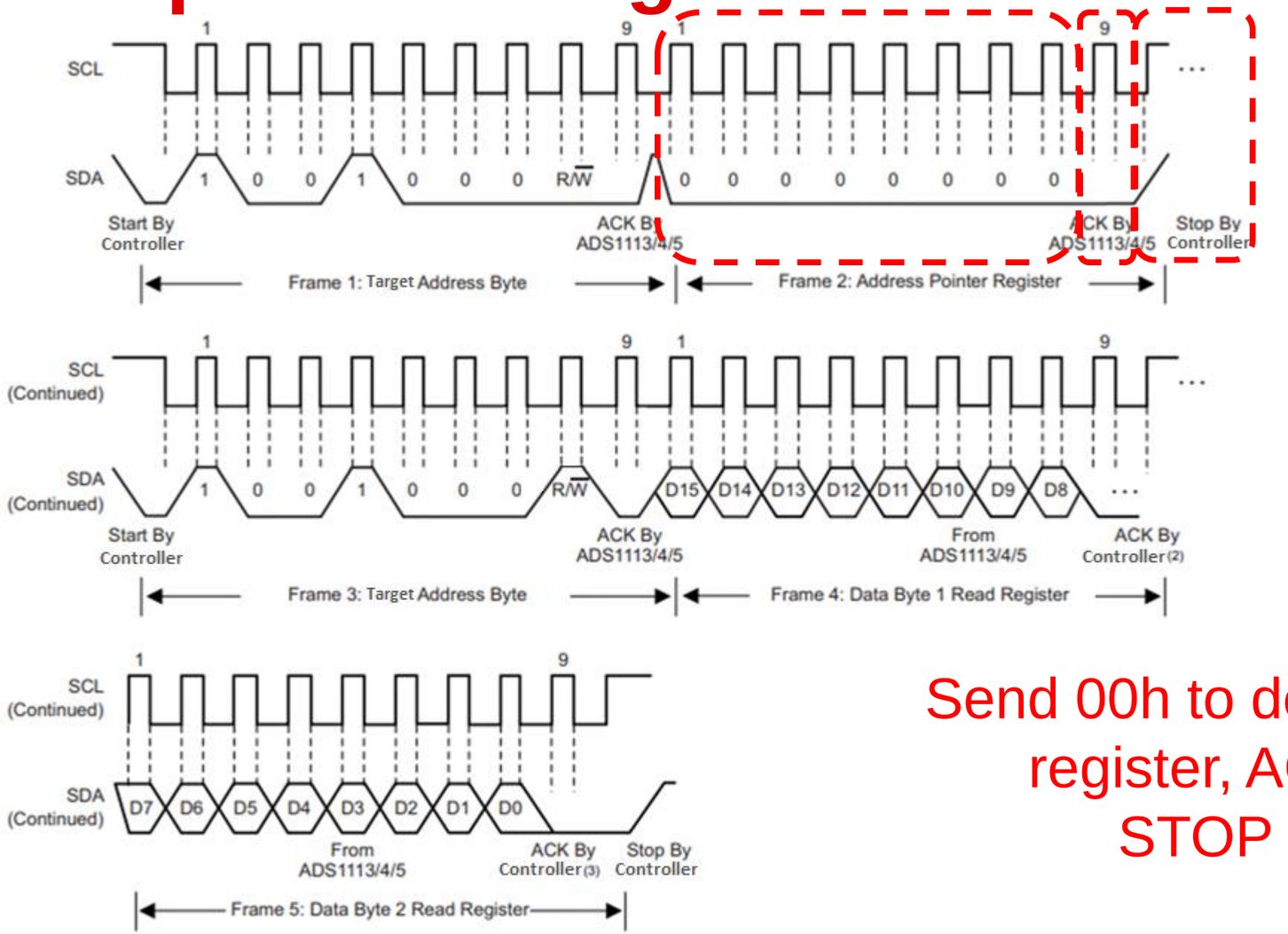
I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

START, Address 48h, Write to device, ACK by ADS1115

The write is needed to tell the device which register is to be read

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.

(2) Controller can leave SDA high to terminate a single-byte read operation.

(3) Controller can leave SDA high to terminate a two-byte read operation.

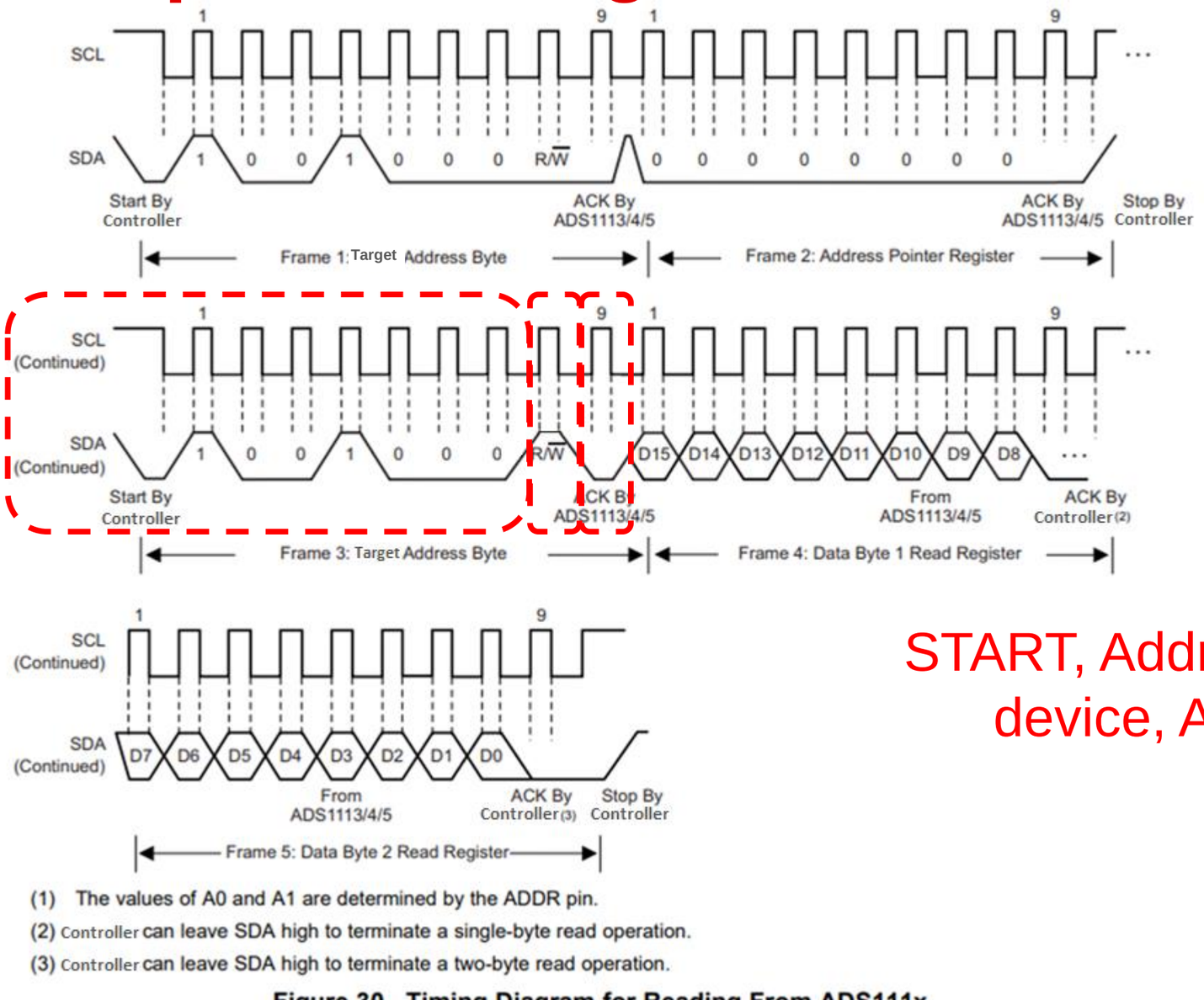
Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

Send 00h to device for conversion register, ACK by ADS1115
STOP by controller

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.

(2) Controller can leave SDA high to terminate a single-byte read operation.

(3) Controller can leave SDA high to terminate a two-byte read operation.

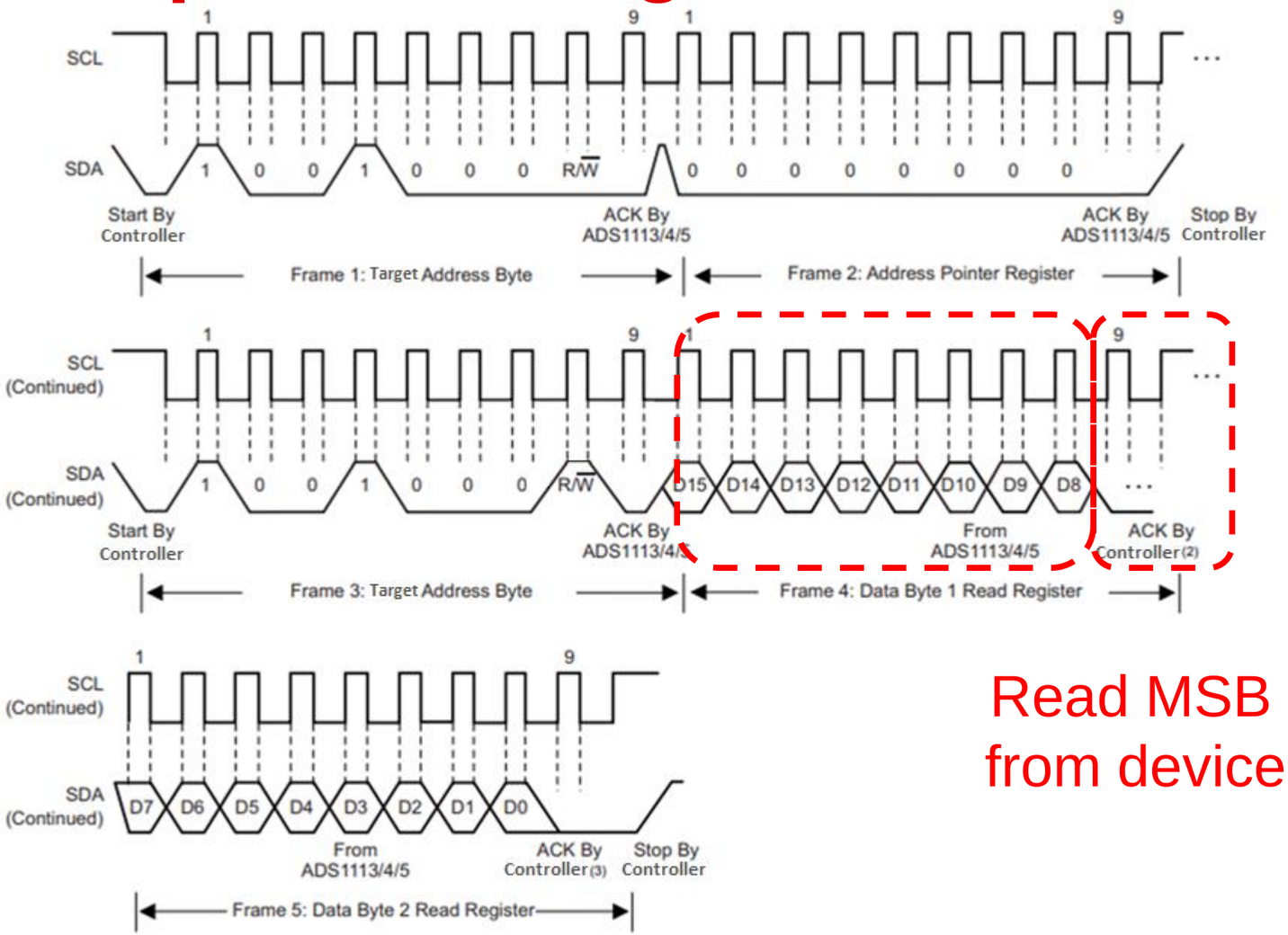
Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

START, Address 48h, Read from device, ACK by ADS1115

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.

(2) Controller can leave SDA high to terminate a single-byte read operation.

(3) Controller can leave SDA high to terminate a two-byte read operation.

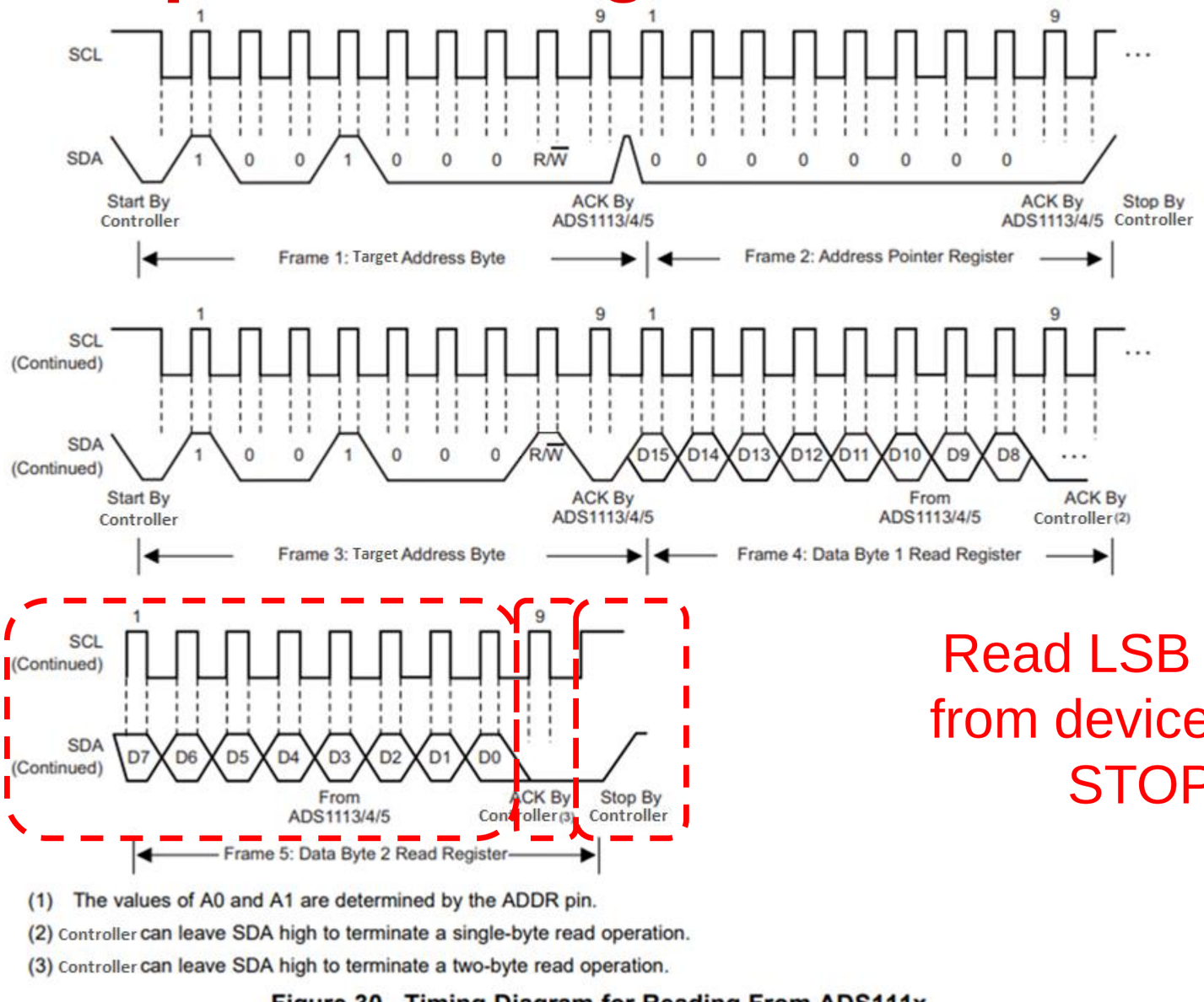
Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

Read MSB of conversion data from device, ACK by controller

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.

(2) Controller can leave SDA high to terminate a single-byte read operation.

(3) Controller can leave SDA high to terminate a two-byte read operation.

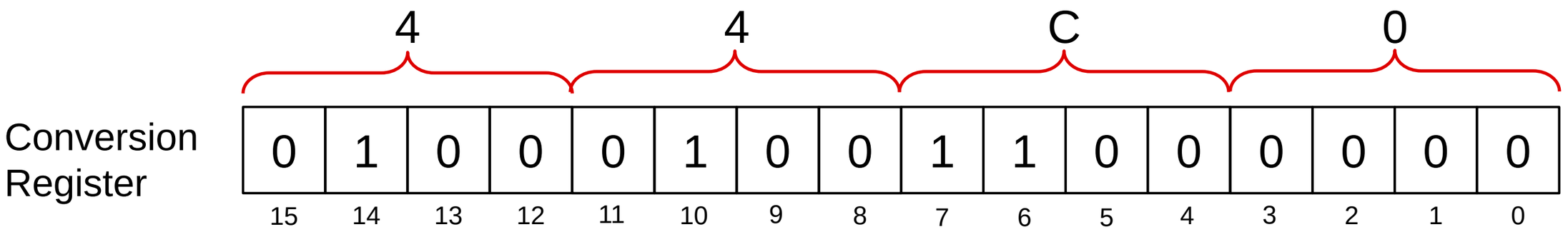
Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

Read LSB of conversion data from device, ACK by controller
STOP by controller

I2C Example: Reading from the Conversion Register



Example, PGA setting ±4.096V:

Conversion data register contents – 0100 0100 1100 0000 (44C0h)

In hex: 44C0h
In decimal: 17600

Conversion result: Measurement = (17600 / 2^{15}) × 4.096V = 2.2 V

Thanks for your time!
Please try the quiz.

Quiz: Basics of I2C: An I2C Example

1. True/false: A write of the configuration register can be written with 16 consecutive bits, instead of breaking up the register into two, one-byte transfers.
   a. True
   b. False

Quiz: Basics of I2C: An I2C Example

1. True/false: A write of the configuration register can be written with 16 consecutive bits, instead of breaking up the register into two, one-byte transfers.
   a. True
   b. False

Quiz: Basics of I2C: An I2C Example

2. What is the I2C protocol byte order of a read from the ADS1115?
   a. I2C address read, pointer address, data MSB, data LSB
   b. I2C address write, pointer address, data MSB, data LSB
   c. I2C address read, pointer address, I2C address read, data MSB, data LSB
   d. I2C address write, pointer address, I2C address read, data MSB, data LSB

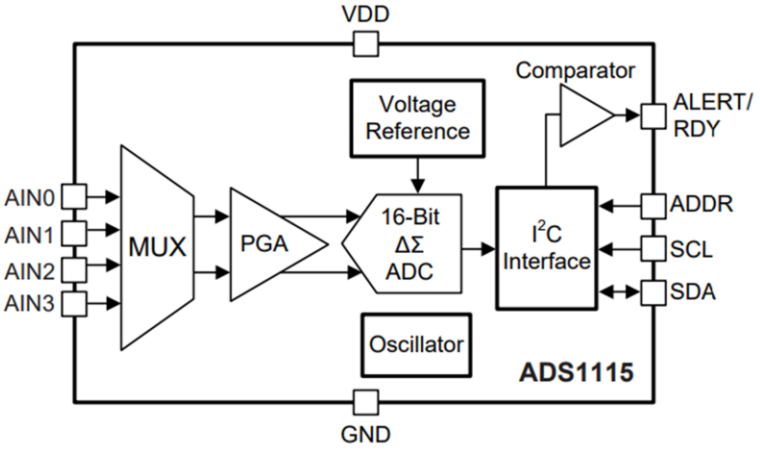
Quiz: Basics of I2C: An I2C Example

2. What is the I2C protocol byte order of a read from the ADS1115?
   a. I2C address read, pointer address, data MSB, data LSB
   b. I2C address write, pointer address, data MSB, data LSB
   c. I2C address read, pointer address, I2C address read, data MSB, data LSB
   d. I2C address write, pointer address, I2C address read, data MSB, data LSB

Thanks for your time!

Hello, and welcome to our in-depth look at communications with precision data converters. In this video, we describe I2C communication using a precision data converter as an example. We'll use an example to show how to write to and read from ADC using I2C.

I2C Example: ADS1115



I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

ADDR Pin	I2C Address
GND	100 1000 (48h)
VDD	100 1001 (49h)
SDA	100 1010 (4Ah)
SCL	100 1011 (4Bh)

Texas Instruments 2

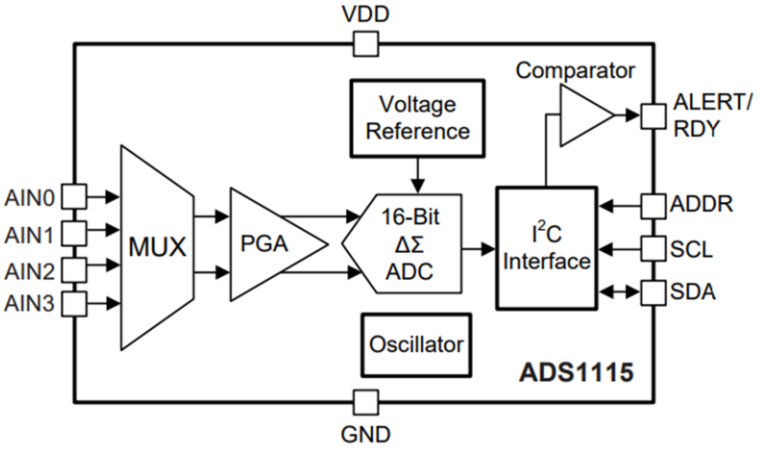
The ADS1115 is a 16-bit precision ADC. It uses an I2C interface and is capable of standard, fast, and high-speed modes. It has several settings that can be set through a configuration register, including the input range set by a programmable gain amplifier (or PGA), a variety of data rates, and a input multiplexer that can be set to make differential measurements, or single-ended measurements with respect to ground

The ADS1115 has an address pin labeled ADDR. This pin can be used to select one of four I2C addresses, meaning that four of these devices could be used on the same bus as long as the devices are programmed to different addresses. The I2C address used for the device depends on the ADDR pin connection.

With the ADDR pin connected to ground, the device has an I2C address of 48 in hex.
With the ADDR pin connected to VDD, SDA, or SCL, the device can be set to other addresses shown in the table.

For this example, we'll use the ADDR set to ground, so the address is 48 hex.

I2C Example: ADS1115



I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

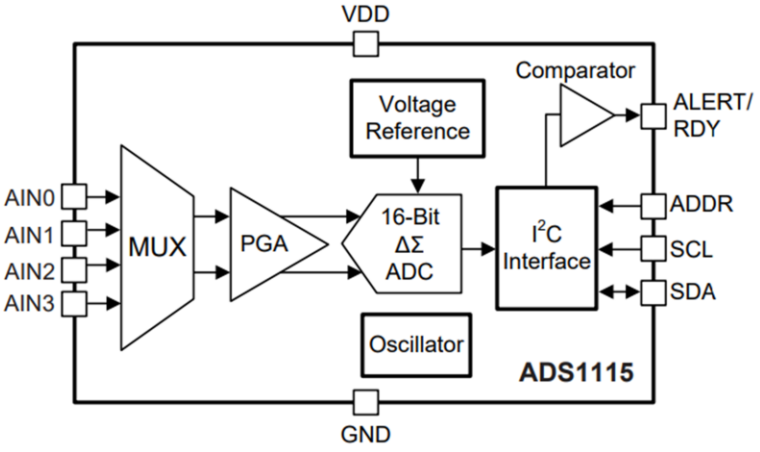
Pointer	Register
00	Conversion data
01	Configuration
10	Lo_threshold register
11	Hi_threshold register

Texas Instruments 3

The ADS1115 has four internal registers, each addressed by an internal pointer register. The pointer register scheme in this example is a common method used in I2C devices with multiple registers.

The first register is the conversion data register. When the ADC completes a conversion, the ADC data is placed in this register and the controller device reads it out from here.

I2C Example: ADS1115

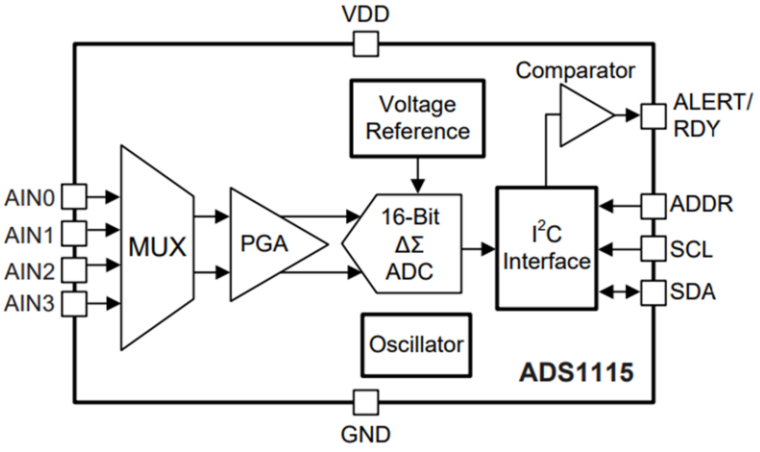


I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

Pointer	Register
00	Conversion data
01	Configuration
10	Lo_threshold register
11	Hi_threshold register

The second register is the configuration register. The controller device will write to this register to program the device and start a conversion. This register sets different aspects of the ADC conversion. It sets the programmable gain amplifier, the input channel, the data rate, and other modes of operation for the device. Later, we'll cover this register in depth, showing how to write to this register and configure the device.

I2C Example: ADS1115



I2C Example
Precision ADC with PGA, adjustable data rates, and multiplexer input
I2C address is pin configurable
Device has four internal registers

Pointer	Register
00	Conversion data
01	Configuration
10	Lo_threshold register
11	Hi_threshold register

Texas Instruments 5

The last two registers are the Lo_threshold register and the Hi_threshold register. These two registers are used to set thresholds for a digital comparator in the device. Once the conversion data goes beyond these thresholds, they can set an alert to the Alert/Ready pin. For this example, we won't configure the comparator and the thresholds.

For this example, we'll start with looking at the settings for the configuration register and using the I2C protocol to program the device.

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

The ADS1115 has a 16-bit configuration register. Writing to this register will program the configuration of the device and start a conversion. We'll discuss the settings for this configuration register and determine what to program into the device. Figure 36 from the datasheet shows the configuration register data fields. This shows the bit positions in the configuration register and what they are used for. Once we determine the all of the settings for the configuration register, we can use a I2C write to set the register.

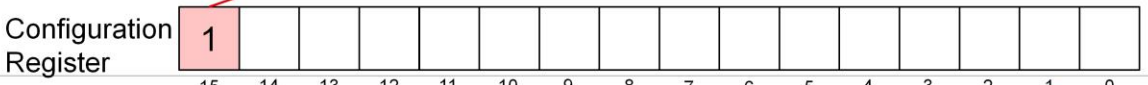
I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
15	OS	R/W	1h	**Operational status or single-shot conversion start** This bit determines the operational status of the device. OS can only be written when in power-down state and has no effect when a conversion is ongoing. When writing: 0 : No effect 1 : Start a single conversion (when in power-down state) When reading: 0 : Device is currently performing a conversion 1 : Device is not currently performing a conversion



Texas Instruments

7

Table 8 in the datasheet shows the configuration register field descriptions, giving a detailed description of the bit setting. Starting with the most significant bit, bit 15 is the single-shot conversion start bit.

Setting bit 15 to 1 will start a single conversion.

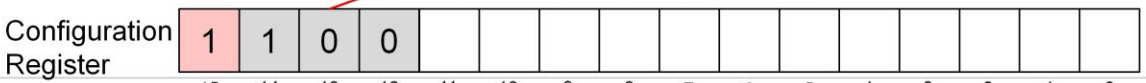
I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
14:12	MUX[2:0]	R/W	0h	**Input multiplexer configuration (ADS1115 only)** These bits configure the input multiplexer. These bits serve no function on the ADS1113 and ADS1114. 000 : AIN_P = AIN0 and AIN_N = AIN1 (default) 001 : AIN_P = AIN0 and AIN_N = AIN3 010 : AIN_P = AIN1 and AIN_N = AIN3 011 : AIN_P = AIN2 and AIN_N = AIN3 100 : AIN_P = AIN0 and AIN_N = GND 101 : AIN_P = AIN1 and AIN_N = GND 110 : AIN_P = AIN2 and AIN_N = GND 111 : AIN_P = AIN3 and AIN_N = GND



Texas Instruments

8

Bits 14 to 12 set the multiplexer setting of the device. For this example, we'll set the device to do a single-ended measurement from AIN0 with respect to ground. Using Table 8, we would set AINP to AIN0 and AINN to GND.

To do this, set bits 14 to 12 to be 100 in binary.

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
11:9	PGA[2:0]	R/W	2h	**Programmable gain amplifier configuration** These bits set the FSR of the programmable gain amplifier. These bits serve no function on the ADS1113. 000 : FSR = ±6.144 V(1) 001 : FSR = ±4.096 V(1) 010 : FSR = ±2.048 V (default) 011 : FSR = ±1.024 V 100 : FSR = ±0.512 V 101 : FSR = ±0.256 V 110 : FSR = ±0.256 V 111 : FSR = ±0.256 V

Configuration Register	1	1	0	0	0	0	1									
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Texas Instruments

9

Bits 11 to 9 set the PGA setting of the device. This is the setting for the programmable gain amplifier. This sets the full-scale range of the input measurement, setting how large of an input signal can be measured by the ADC. Here, we set the ADC to measure a signal as large as plus and minus 4.096 Volts.

Set bits 11 to 9 to be 001 in binary.

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
8	MODE	R/W	1h	**Device operating mode** This bit controls the operating mode. 0 : Continuous-conversion mode 1 : Single-shot mode or power-down state (default)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Configuration Register	1	1	0	0	0	0	1	1								

TEXAS INSTRUMENTS

10

Bit 8 sets the operating mode of the device. For this operation, we want to set the device to be in single-shot conversion mode

Set bit 8 to 1.

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	DR[2:0]	R/W	4h	**Data rate** These bits control the data rate setting. 000 : 8 SPS 001 : 16 SPS 010 : 32 SPS 011 : 64 SPS 100 : 128 SPS (default) 101 : 250 SPS 110 : 475 SPS 111 : 860 SPS

Configuration Register	1	1	0	0	0	0	1	1	1	1	1					
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Texas Instruments

11

Bits 7 to 5 set the data rate for the ADC of the device. We can set this to the highest data rate of 860 samples per second.

Set bits 7 to 5 to 111.

I2C Example: Write to the Configuration Register

Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description
1:0	COMP_QUE[1:0]	R/W	3h	**Comparator queue and disable (ADS1114 and ADS1115 only)** These bits perform two functions. When set to 11, the comparator is disabled and the ALERT/RDY pin is set to a high-impedance state. When set to any other value, the ALERT/RDY pin and the comparator function are enabled, and the set value determines the number of successive conversions exceeding the upper or lower threshold required before asserting the ALERT/RDY pin. These bits serve no function on the ADS1113. 00 : Assert after one conversion 01 : Assert after two conversions 10 : Assert after four conversions 11 : Disable comparator and set ALERT/RDY pin to high-impedance (default)

Configuration Register	1	1	0	0	0	0	1	1	1	1	1	0	0	0	1	1
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Texas Instruments

12

The last five bits from 4 down to 0 are all used for the digital comparator for this device. We won't use the digital comparator here, so we'll disable this setting with the last two bits of the register and set the remaining bits to their default setting.

Set bits 4 to 0 to 00011.

I2C Example: Write to the Configuration Register

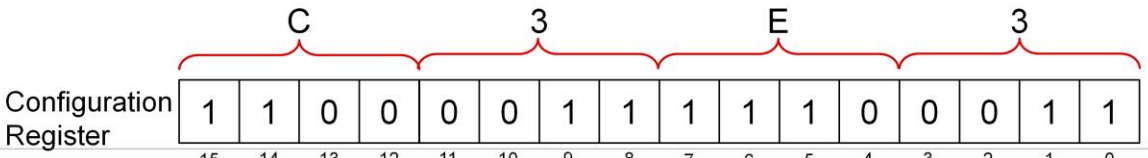
Figure 36. Config Register

15	14	13	12	11	10	9	8
OS	MUX[2:0]			PGA[2:0]			MODE
R/W-1h	R/W-0h			R/W-2h			R/W-1h
7	6	5	4	3	2	1	0
DR[2:0]			COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE[1:0]	
R/W-4h			R/W-0h	R/W-0h	R/W-0h	R/W-3h	

Table 8. Config Register Field Descriptions

Bit	Field	Type	Reset	Description

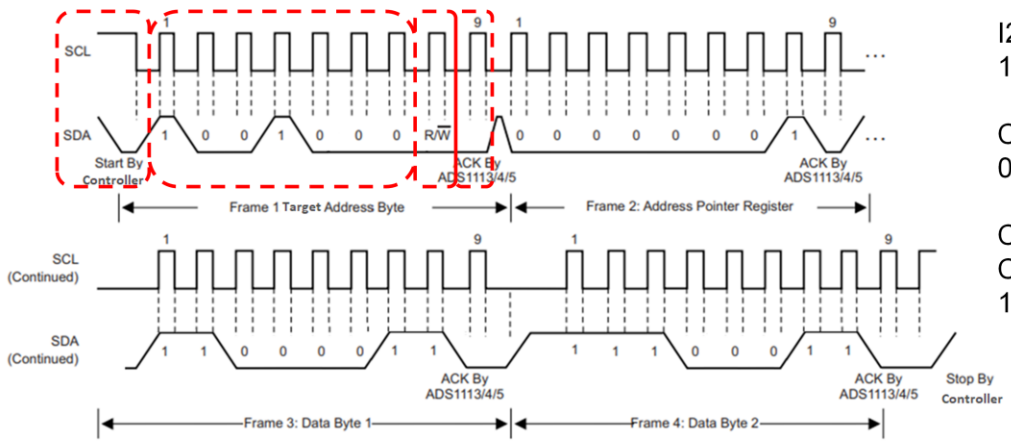
Configuration Register can also be written as C3E3 in hexadecimal



Texas Instruments 13

Now we have the complete configuration register setting for the ADS1115. We'll use these bits for the write to the register. This register can also be represented in hexadecimal as C3E3.

I2C Example: Write to the Configuration Register



I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Figure 31. Timing Diagram for Writing to ADS111x

START, Address 48h, Write to device, ACK by ADS1115

Figure 31 in the ADS1115 datasheet shows a generic diagram for writing to the device. The figure has been altered to show the exact bit transactions sent to the device. We'll use the settings that we've previously discussed to show exactly what we need to write to the configuration register. For reference, we'll put up the data we need on the right side of the slide.

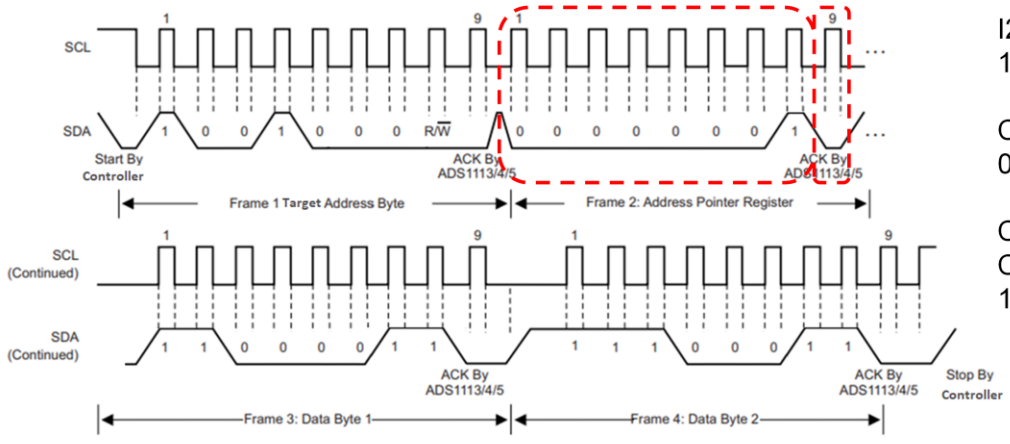
First, the I2C write starts with a START Condition. SDA is pulled low, and then SCL is pulled low.

Then we write the I2C address. With the ADDR pin connected to ground, we use an address of 100 1000 (or 48 in hex).

The Read/Write bit is then set low, indicating that we want to write to the device.

After the completion of the address frame, the ADS1115 should acknowledge the address by pulling down the SDA for the last bit of the address frame. The controller sends out the address and read/write information to all of the slaves on the bus. If a target has a matching address, they will acknowledge to let the controller know that this is a valid address and they are ready to receive information. This ACK lets the controller know that the target device is ready for communication.

I2C Example: Write to the Configuration Register



I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Figure 31. Timing Diagram for Writing to ADS111x

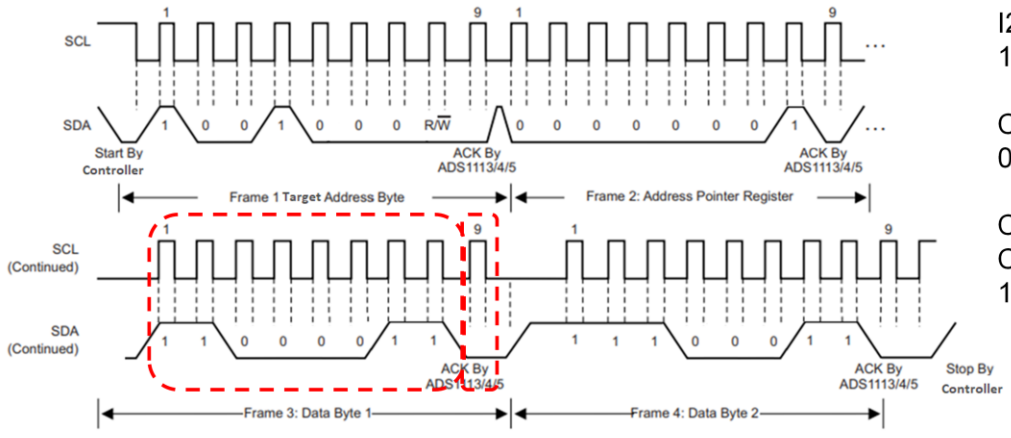
Send 01h to device for configuration register, ACK by ADS1115

After we've indicated that we want to write to the ADS1115, we need to tell the device that we want to write to the configuration register.

The second byte is the register pointer for the configuration register. Here, we send 0000 0001 to the ADS1115.

As a response, the ADS1115 pulls down on SDA for an ACK. Again, the target device is letting the controller know it has received the address pointer data.

I2C Example: Write to the Configuration Register



I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Figure 31. Timing Diagram for Writing to ADS111x

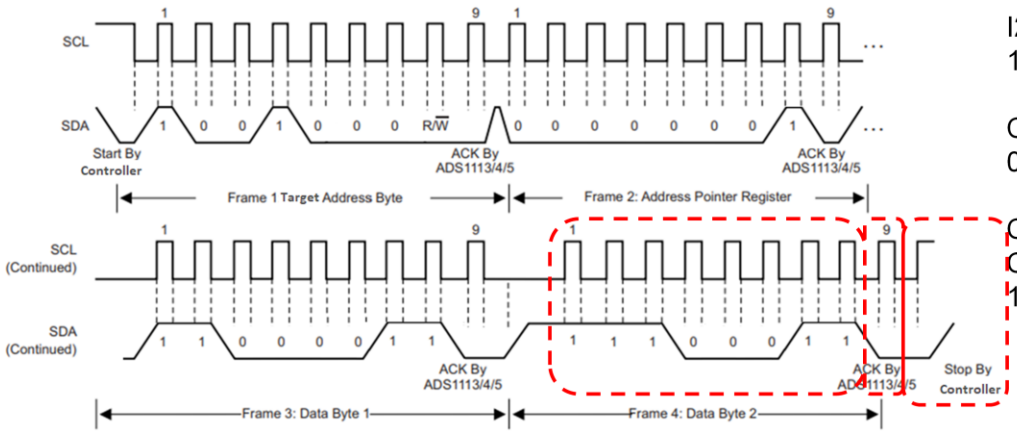
Send first configuration byte to device C3h, ACK by ADS1115

Now we start to send in the configuration register data one byte at a time.

For this byte, we send in the first byte of the configuration register. Here, we send 1100 0011 to the ADS1115.

The ADS1115 ACKs the first byte and pulls down on SDA for the last bit.

I2C Example: Write to the Configuration Register



I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Figure 31. Timing Diagram for Writing to ADS111x

Send second configuration byte to device E3h, ACK by ADS1115, STOP

Finally, we send in the last byte of the configuration register.

Here, we send 1110 0011 to the ADS1115.

The ADS1115 ACKs the first byte and pulls down on SDA for the last bit.

At the end, the controller releases the bus by issuing a STOP condition. SCL is released high and then SDA is released high.

I2C Example: Write to the Configuration Register

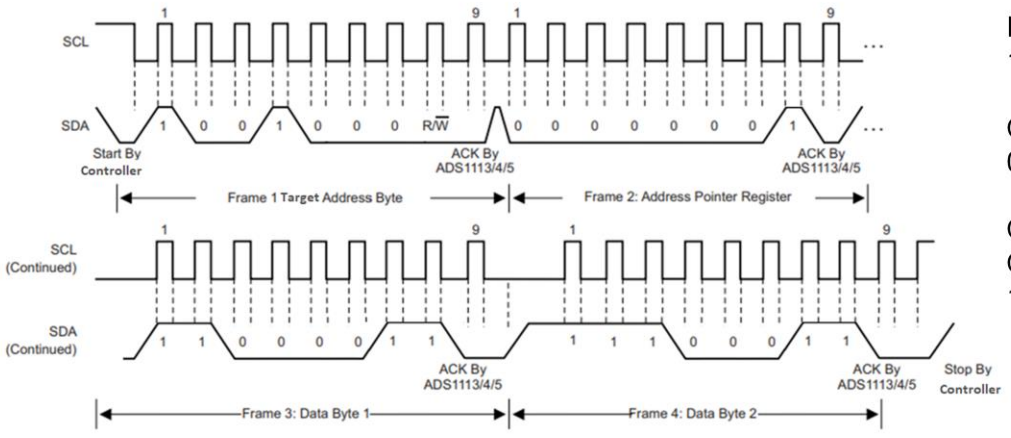


Figure 31. Timing Diagram for Writing to ADS111x

I2C Address:
100 1000

Configuration Pointer:
0000 0001

Configuration Register Contents:
1100 0011 1110 0011

Putting it all together, Figure 31 now looks like this:
[Click] Here you have the same diagram with all the proper bit settings for all frames. Figure 31 in the datasheet is very useful when debugging communications. You could plot the I2C communication with an oscilloscope, and compare this figure with the plot.

I2C Example: Reading from the Conversion Register

Figure 35. Conversion Register

15	14	13	12	11	10	9	8
D15	D14	D13	D12	D11	D10	D9	D8
R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h
7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0
R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h

LEGEND: R/W = Read/Write; R = Read only; -n = value after reset

Table 7. Conversion Register Field Descriptions

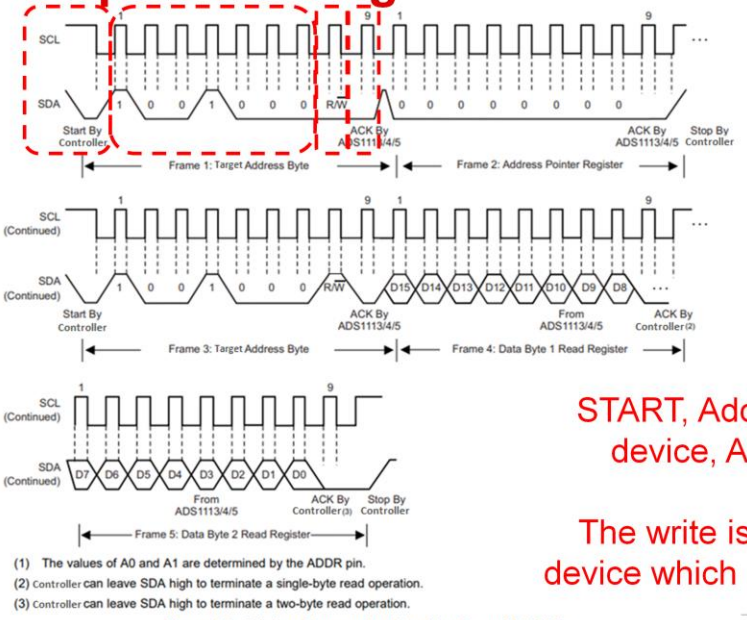
Bit	Field	Type	Reset	Description
15:0	D[15:0]	R	0000h	16-bit conversion result

Conversion Register pointer: 0000 0000
Conversion data is 16 bits with MSB first

The ADS1115 has a 16-bit ADC and therefore puts out 16-bit data conversions. To get the ADC conversion data, you need to read from a conversion register. The conversion register address pointer is 0000 0000.

Figure 35 from the datasheet shows the conversion register data field and Table 8 in the datasheet shows the configuration register field description. Conversion data appears as a 16 bit result in binary two's complement. A positive full-scale input produces an output code of 7FFFh and a negative full-scale input produces an output code of 8000h.

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.
(2) Controller can leave SDA high to terminate a single-byte read operation.
(3) Controller can leave SDA high to terminate a two-byte read operation.

Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

START, Address 48h, Write to device, ACK by ADS1115

The write is needed to tell the device which register is to be read

Figure 30 in the ADS1115 datasheet shows a generic diagram for reading from the device. Again, we'll use the settings that we've previously discussed to show exactly what we need to read from the conversion register. For reference, we'll put up the address and the register pointer that we need on the right side of the slide.

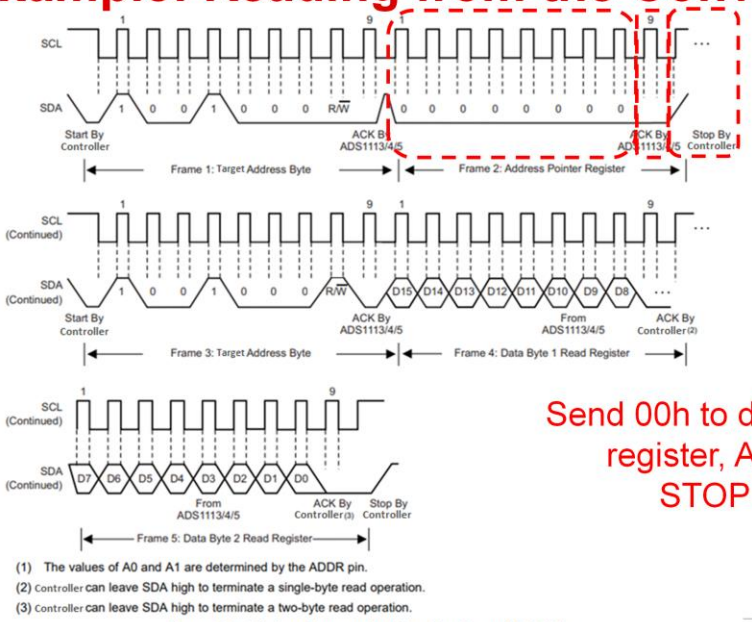
First, the I2C write starts with a START Condition. SDA is pulled low, and then SCL is pulled low.

Then we write the I2C address. With the ADDR pin connected to ground, we use an address of 100 1000 (or 48 in hex).

We need to tell the device which register we need to read from. For this, we need a *WRITE* to the device so that we can set up the read from the ADS1115. At this point, the Read/Write bit is then set low, indicating that we want to write to the device.

After the completion of the address frame, the ADS1115 acknowledges the address by pulling down the SDA for the last bit of the address frame.

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.
(2) Controller can leave SDA high to terminate a single-byte read operation.
(3) Controller can leave SDA high to terminate a two-byte read operation.

Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

Send 00h to device for conversion register, ACK by ADS1115
STOP by controller

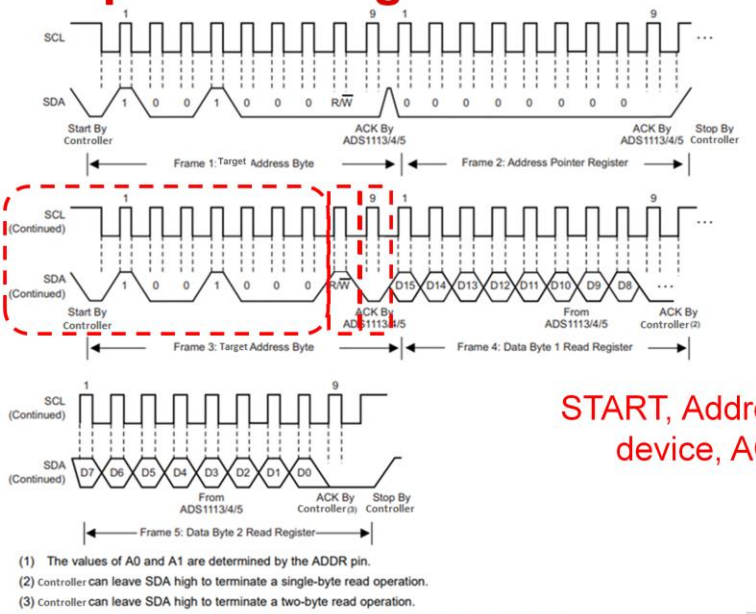
After we've indicated that we want to *WRITE* to the ADS1115, we need to tell the device that we want to access the configuration register.

The second byte is the register pointer for the configuration register. Here, we send 0000 0000 to the ADS1115.

As a response, the ADS1115 pulls down on SDA for an ACK.

Finally the controller issues a STOP to release the bus.

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.
(2) Controller can leave SDA high to terminate a single-byte read operation.
(3) Controller can leave SDA high to terminate a two-byte read operation.

Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

START, Address 48h, Read from device, ACK by ADS1115

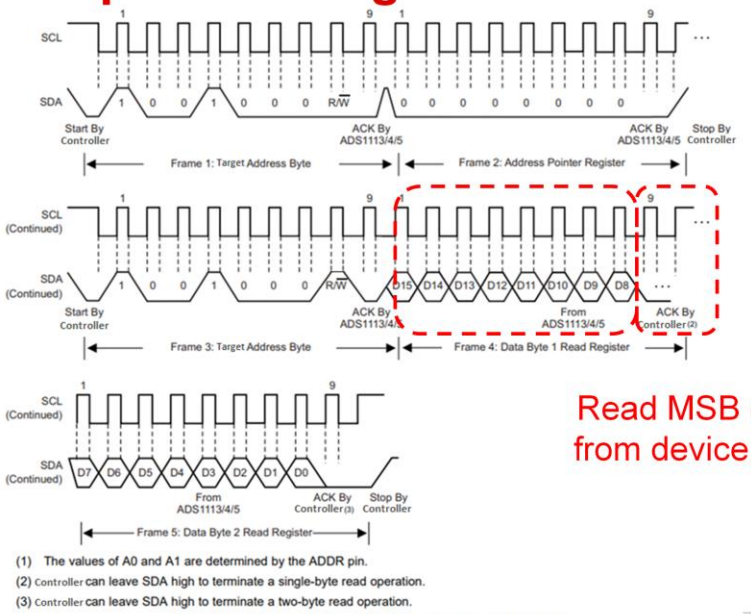
Now that we've told the device we want to access the conversion register, we follow up with the read from the conversion register.

We write the I2C address again.

Now, we need a *READ* from the device from the conversion register of the ADS1115. At this point, the Read/Write bit is then set high, indicating that we now want to read from the device.

Again after the completion of the address frame, the ADS1115 should ACK the address.

I2C Example: Reading from the Conversion Register



I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

Read MSB of conversion data from device, ACK by controller

(1) The values of A0 and A1 are determined by the ADDR pin.
(2) Controller can leave SDA high to terminate a single-byte read operation.
(3) Controller can leave SDA high to terminate a two-byte read operation.

Figure 30. Timing Diagram for Reading From ADS111x

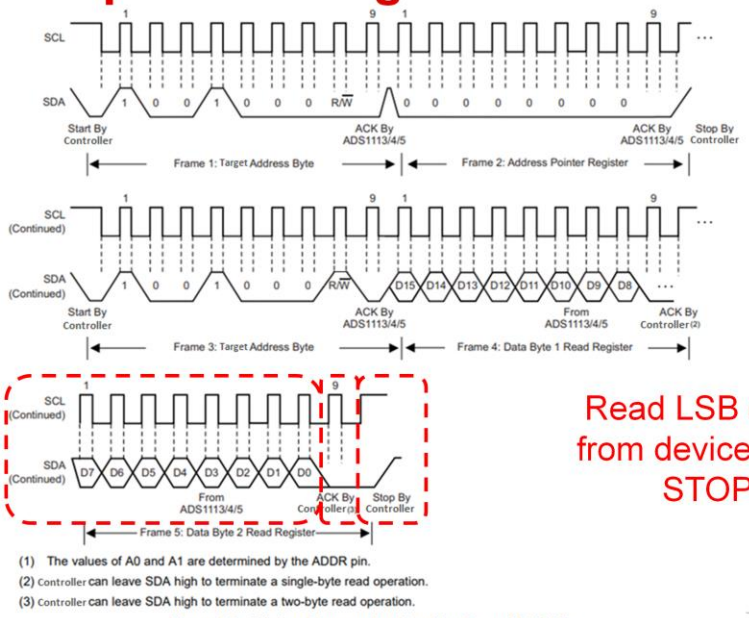
Texas Instruments 23

Now, we can read the conversion register one byte at a time.

First you read the most significant byte.

And as a response for reading the data, the controller device pulls down on SDA for an ACK. Because the target device is sending data, it is now the controller device that pulls down SDA for an ACK. This tells the target device that the data has been received by the controller device.

I2C Example: Reading from the Conversion Register



(1) The values of A0 and A1 are determined by the ADDR pin.
(2) Controller can leave SDA high to terminate a single-byte read operation.
(3) Controller can leave SDA high to terminate a two-byte read operation.

Figure 30. Timing Diagram for Reading From ADS111x

I2C Address:
100 1000 (48h)

Conversion Pointer:
0000 0000 (00h)

Read LSB of conversion data from device, ACK by controller STOP by controller

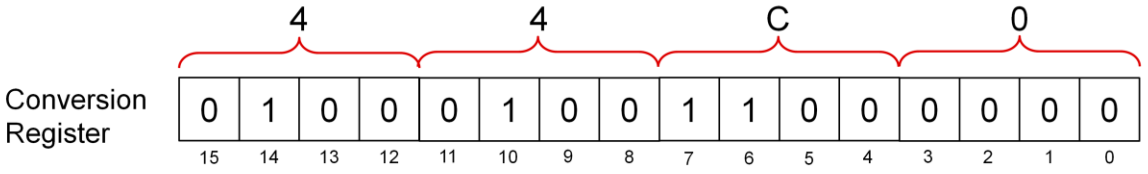
TEXAS INSTRUMENTS 24

Then follows the read of the least significant byte.

An then another ACK from the controller.

Finally, the controller sends a STOP to end the I2C communication.

I2C Example: Reading from the Conversion Register



Example, PGA setting ±4.096V:

Conversion data register contents – 0100 0100 1100 0000 (44C0h)

In hex: 44C0h
In decimal: 17600

Conversion result: Measurement = (17600 / 2^{15}) × 4.096V = 2.2 V

Just to follow up with the conversion register result, let's use one last example with reading data and making the conversion.

Let's say you read out the data from the conversion register and it reads 44C0 in hex, or 17600 in decimal. This is the ADC output based on the input voltage from the measurement.

Using this value, you can convert the conversion register to a voltage for the ADC measurement. With a positive full scale range of 4.096V, you can convert this to a measured voltage. Here, the ADC is reporting 2.2V

That concludes this video – thank you for watching! Please try the quiz to check your understanding of this video's content.

Quiz: Basics of I2C: An I2C Example

1. True/false: A write of the configuration register can be written with 16 consecutive bits, instead of breaking up the register into two, one-byte transfers.
   a. True
   b. False

Quiz: Basics of I2C: An I2C Example

1. True/false: A write of the configuration register can be written with 16 consecutive bits, instead of breaking up the register into two, one-byte transfers.
   a. True
   b. False

I2C transmissions come in single byte transmission frames. This is standardized and both controller devices and target devices know that the transmission will come in a set format.

Quiz: Basics of I2C: An I2C Example

2. What is the I2C protocol byte order of a read from the ADS1115?
   a. I2C address read, pointer address, data MSB, data LSB
   b. I2C address write, pointer address, data MSB, data LSB
   c. I2C address read, pointer address, I2C address read, data MSB, data LSB
   d. I2C address write, pointer address, I2C address read, data MSB, data LSB

Quiz: Basics of I2C: An I2C Example

2. What is the I2C protocol byte order of a read from the ADS1115?
   a. I2C address read, pointer address, data MSB, data LSB
   b. I2C address write, pointer address, data MSB, data LSB
   c. I2C address read, pointer address, I2C address read, data MSB, data LSB
   d. I2C address write, pointer address, I2C address read, data MSB, data LSB

While, the objective of this sequence is to read the conversion register, we still need to tell the target device what register the controller wants to read from. Notice that we first write to the pointer register, before we can read that register.

Thanks for your ti