SIEMENS

Working with Binary Numbers

Application examples

09/2020
AG101920

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠ DANGER

indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠ WARNING

indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠ CAUTION

indicates that minor personal injury can result if proper precautions are not taken.

NOTICE

indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠ WARNING

Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Digital Industries
Postfach 48 48
90026 NÜRNBERG
GERMANY

Document order number: AG101920
Ⓟ 10/2020 Subject to change

Table of contents

Introduction 1

1.1 Objective

The objective of this application guide is to help the user become familiar with the steps required to convert between decimal and binary numbers and discusses some of the problems that can occur.

1.2 Disclaimer

Note

While every effort is made to verify the following information, no warranty of accuracy or usability is expressed or implied.

Overview

2

In many Siemens devices, binary data is stored in 16 bit registers. Depending on what you are using to read this information, access to it will be either very easy or will require some effort.

Typically PLCs handle binary data very easily. For example, in an Allen Bradley SLC500 the 4th bit of register 15 is referenced as “N7:15/4”. However, if you are using a PC, accessing binary data can be more difficult depending on the type of software you are using, how it represents data, and how it lets the user access the data.

Description

After reading a 16 bit register, a particular piece of software can do one of the following:

1. Let the user reference the information as bits
2. Represent the 16 bit number as an unsigned integer
3. Represent the 16 bit number as a signed integer

Description 3

3.1 Information as Bits

If the software lets the user reference (read and write) the information as bits, then no conversion is required.

It should be noted though, that there is a known problem with Wonderware InTouch in this situation. If you use the default scaling of −32768 to 32767 for the register, then some of the bits will change due to an internal problem with InTouch. However, if the scaling is changed to be from 0 to 65536, then the data will be correct (for more details on this problem, please see application guide 99001).

3.2 Information as Unsigned Integer

If the data is represented as an unsigned integer, then the user must convert from an integer to a binary number. This can be done easily by using a conversion formula. For example, you may want to see the 4^{th} bit of Register 41,070 in an LUT400, which represents the state of the digital input 4. This is done with this formula (using the syntax found in Excel):

Bit 4 = Mod (Int (R41070/2^3),2)

Where: Mod is the Modulus function and is the remainder after a division. Int is the Integer function and is the integer part of the number.

If R41070 = 27, then 27/2^3 = 27/8 = 3.375, and Int(3.375) = 3, and the Mod (3,2) is 1. Therefore, Bit 4 would be a 1.

The complete formula is:

Let a be an integer (the number to be converted i.e. R41070 in the example above).

Let (r_n , r_{n-1} , ..., r_3 , r_2 , r_1) be the binary equivalent of a , where r_1 is the least significant bit, i.e. the low order of the number (in the example above, r_n would be r_4 and would refer to bit 4).

Then:

r_n = Mod (Int (a / 2^{n-1}) , 2)

This is a very useful formula; however, it is important to pay very close attention to how the math is done. Three potential problems are:

1. If the calculation (a / 2^{n-1}) is done using integer math, then the wrong number is generated. This needs to be done with floating point math.
2. The Int function can do one of two things, either it rounds to the nearest whole number or it truncates (i.e. chops off the digits after the decimal). For this formula, we need it to truncate.
3. The integer numbers must be unsigned (see below for converting a signed integer into a unsigned integer).

After converting the number into the series of bits, the user is faced with the problem of converting it back to an integer in order to be able to write it back to the device. This is done by virtue of the fact that each bit in a binary number has an integer value. Therefore all you have to do is add up the values. For example, if you have the following number:

011101

Then the integer equivalent is:

1*2^0+0*2^1+1*2^2+1*2^3+1*2^4+0*2^5 =

1*1+0*2+1*4+1*8+1*16+0*32 = 1+4+8+16 = 29

The formula is:

Let N be the decimal equivalent of the binary number:

$N = \sum r_a * 2^{(a-1)}$, for a= 1 to nu

3.3 Information as Signed Integers

If the integer is a signed integer, then to use the formulas stated above the number has to be converted into an unsigned integer. This can be done by recognizing that the only difference between a signed and unsigned 16 bit integer is that the 16^{th} bit is used as a sign in the signed integer.

Therefore, to change a signed integer to an unsigned integer:

If N<0 then N= N + 65536

And to convert from unsigned to signed integer:

If N>=32768 then N=N-65536