

Concepts of Conversion of Base 2 Fixed Point Register Binary Number to Base 10 Decimal

Sean Rodby, Luke Snyder, Autar Kaw

University of South Florida

United States of America

kaw@eng.usf.edu

Website: <http://numericalmethods.eng.usf.edu>

Version: Maple 12; Last Revised: August 28, 2008

Introduction

The following worksheet illustrates how to convert a fixed point register binary (base-2) number to a decimal number (base-10) using loops and various conditional statements. The user inputs a binary number in the *Input* section of the program, and then an equivalent decimal number is given as an output.

Initialization

```
[ > restart : with(ArrayTools) :
```

Section 1: Input

This is the only section where the user interacts with the program.

```
[ Enter number to be converted to decimal number
```

```
[ > bin_num := "111111101.11"
```

```
bin_num := "111111101.11"
```

(3.1)

```
[ This is the end of the user section. All information must be entered before proceeding to the next section. RE-EXECUTE THE PROGRAM.
```

Section 2: Procedure

The *parse* command is used to take the Maple string *bin_num* and parse the string as if it had been entered or read from a file. The *floor* command is used to isolate the *integer* part of the base-2 number. Then the *length* command determines the length of the entire binary number and all the characters in the string.

```
[ > int_bin := parse(bin_num) :  
int_bin := floor(int_bin) :  
n := length(bin_num) :  
str_int := convert(int_bin, string) :  
m := length(str_int) :
```

Using a loop to sum values of the integer portion of the base-2 number. The loop variable *sumint* is used for summation and is initialized at 0.

```

> sumint := 0 :
  for i to m do
    bin_int[i] := parse(str_int[i]) :
    sumint := sumint + bin_int[i]*2^(m-i);
  end do:

```

Using a loop to sum values of the fractional portion of the base-2 number. The loop variable *sumfrac* is used for summation and is initialized at 0. Note that the starting point in this loop is the length of the integer portion (m), plus 2 which effectively skips the decimal point in the character array.

```

> sumfrac := 0 :
  j := 1 :
  for i from m + 2 to n do
    bin_frac[j] := parse(bin_num[i]);
    bin_frac_str[j] := bin_num[i];
    sumfrac := evalf(sumfrac + bin_frac[j]*2^(-j));
    j := j + 1;
  end do:

```

Adding the *fractional* portion of the base-2 number with the *integer* portion which yields the base-10 number.

```

> total_dec := evalf(sumint + sumfrac)
                                total_dec := 509.7500000

```

(4.1)

▼ Conclusion

This worksheet illustrates the use of Maple to convert a base-2 binary number to a base-10 number. It is important to understand the binary system as it has numerous applications. Critical to this understanding is being able to convert decimal numbers to binary numbers, and vice-versa.

▼ References

Binary Representation of Numbers.

See: http://numericalmethods.eng.usf.edu/mws/gen/01aae/mws_gen_aae_txt_binaryrepresentation.pdf

Legal Notice: The copyright for this application is owned by the author(s). Neither Maplesoft nor the author(s) are responsible for any errors contained within and are not liable for any damages resulting from the use of this material. This application is intended for non-commercial, non-profit use only. Contact the author for permission if you wish to use this application in for-profit activities.