

MIF File

Last updated 7/21/23

MIF File

- Setting ROM values using a MIF file

- Only works for inferred ROM

- Create a signal of your ROM type

- Set 2 attributes

```
ATTRIBUTE ram_init_file: string;
```

```
ATTRIBUTE ram_init_file of your_signal: signal is "name_of_MIF_file";
```

- Place the MIF file in your project

MIF File

- MIF File Format
 - Saved as my_file.mif

```
-- comment line
--

WIDTH=8;    -- word width
DEPTH=256;  -- number of words

ADDRESS_RADIX=HEX;
DATA_RADIX=HEX;

CONTENT BEGIN
    000 : 0C;
    001 : 1E;
    [002..003] : 33;
    004 : 3F;
    [005..006] : 33;
    007 : 00;
    008 : 3F;
    [009..00A] : 66;
    00B : 3E;
    [00C..00D] : 66;
    00E : 3F;
    00F : 00;
    010 : 3C;
```

Unspecified addresses
are assigned the value
0

Options are:

BIN
OCT
HEX
DEC (signed decimal)
UNS (unsigned decimal)

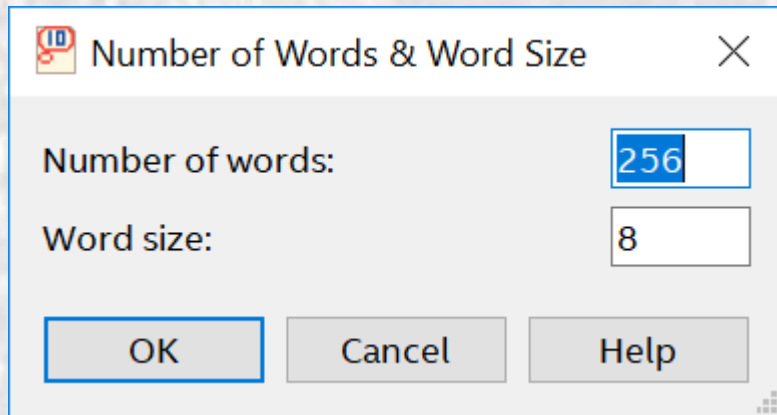
MIF File

- Addressing options

Address : Data Pairs Syntax Rules	Definition	Example
A : D	Addr[A] = D	2 : 4 Address: 01234567 Data: 00400000
[A0..A1] : D	Addr[A0] to [A1] contain data D	[0..7] : 6 Address: 01234567 Data: 66666666
[A0..A1] : D0 D1	Addr[A0] = D0, Addr[A0+1] = D1, Addr[A0+2] = D0, Addr[A0+3] = D1, until A0+n = A1	[0..7] : 5 6 Address: 01234567 Data: 56565656
A : D0 D1 D2	Addr[A] = D0, Addr[A+1] = D1, Addr[A+2] = D2	2 : 4 5 6 Address: 01234567 Data: 00456000

MIF File

- Quartus created file
 - File → New → Memory Initialization File
 - Edit the values on the screen



Addr	+0	+1	+2	+3	+4	+5	+6	+7	ASCII
0	0	3	5	7	9	11	13	15
8	17	19	21	23	25	27	29	31
16	33	35	37	39	41	0	0	0	!#%}...
24	0	0	0	0	0	0	0	0
32	0	25	0	0	23	0	0	0
40	0	0	0	0	0	0	0	0
48	0	0	0	0	0	0	0	0
56	0	0	0	0	0	0	0	0
64	0	0	0	0	0	0	0	0
72	0	0	0	0	0	0	0	0
80	0	0	0	0	0	0	0	0
88	0	0	0	0	0	0	0	0
96	0	0	0	0	0	0	0	0
104	0	0	0	0	0	0	0	0
112	0	0	0	0	0	0	0	0
120	0	0	0	0	0	0	0	0

MIF File

- Quartus created file
 - Right click in the memory table and select Custom Fill Cells

Custom Fill Cells

Allows you to custom fill an address range with either a repeating sequence, or from a starting point with incrementing or decrementing values.

Address range

The current address radix is: unsigned decimal

Starting address: 57 Ending address: 57

Custom value(s)

The current memory radix is: unsigned decimal

Repeating sequence (numbers can be delimited by either a

Incrementing / decrementing

Starting value: Increment bv

OK Cancel Help