Data formats used in MFS5

Version 1.2 (14.10.2025)

Table of contents

All most file formats used in MFS software are open-source, some of them are proprietary.

*.dbl

Binary file, no header

Format for 1D real data: consists of N double values

```
N×real(double-8Byte)
```

Matlab code snippet for loading:

```
fid=fopen('dataname.dbl');
data=fread(fid,'double');
fclose(fid);
plot(data)
```

Matlab code snippet for saving:

```
fid=fopen('dataname.dbl', 'w');
fwrite(fid, [data], 'double');
fclose(fid);
```

*.mpi

ASCII file

Textfile: contains information of the associated *.dbl file

ATTENTION: The first three rows are mandatory

1	Name of dataset: can be different from filename
2	Samplingrate (integer value)
3	Associated *.dbl filename
4+	Following rows can contain additional information (optional)

*.cdbl

Binary file, no header

Format for 1D complex data: consists of N complex values (real + imaginary double values)

```
N×[real(double-8Byte);imaginary(double-8Byte)]
```

Matlab code snippet for saving:

```
fid=fopen('dataname.dbl', 'w');
fwrite(fid, [real(data), imag(data)].', 'double' );
fclose(fid);
```

*.cmpi

ASCII file

Textfile: contains information of the associated *.cdbl file

ATTENTION: The first three rows are mandatory

1	Name of dataset: can be different from filename
2	Samplingrate (integer value)
3	Associated *.cdbl filename
4+	Following rows can contain additional information (optional)

*.r1da

Binary file, with header

Format for storing 1D real arrays: m×n real values

Dim(integer-4Byte)	Dim×real(double-8Byte)

Pseudo-code

```
procedure Save_1DArray ( data: 2Dreal )
     i, m: Integer;
begin
          m=Length ( data );
          file.Write ( m, 4 );
          file.Write ( data [i,0], n*8 )
end
```

*.sm or *.r2da

Binary file, with header

Format for storing 2D real arrays: m×n real values

Dim1(integer-4Byte)	Dim2(integer-4Byte)	Dim1×Dim2×real(double-8Byte)

Pseudo-code

```
procedure Save_2DArray ( data: 2Dreal )
     i, m, n: Integer;
begin
           m=Length ( data );
           n=Length ( data[0] );
           file.Write ( m, 4 );
           file.Write ( n, 4 );
           for i=0 to m-1 do
                file.Write ( data [i,0], n*8 )
end
```

*.c4da

Binary file, with header

Format for storing 4D complex arrays: x×y×z×w complex values

Dim1(integer-4Byte)	Dim2(integer-4Byte)	Dim3(integer-4Byte)	Dim4(integer-4Byte)	Dim1×Dim2×Dim3×Dim4×[real(double-8Byte);imaginary(double-8Byte)]

Pseudo-code

```
procedure Save_4DArray (data: 4Dcomplex )
     yy, zz, ww, dx, dy, dz, dw: Integer;
begin
dw=Length ( data );
           dz=Length ( data[0] );
           dy=Length ( data[0,0] );
           dx=Length ( data[0,0,0] );
           file.Write ( dx, 4 );
           file.Write ( dy, 4 );
file.Write ( dz, 4 );
file.Write ( dw, 4 );
for ww=0 to _dW-1 do
for zz=0 to _dZ-1 do
for yy=0 to _dY-1 do
file1.Write (data [ww,zz,yy,0], 2*8* dx)
end
```

*.c8da

Binary file, with header

Format for storing 8D complex arrays: x×y×z×w×d5×d6×d7×d8 complex values.

It has to be mentioned that this format can be used as a 4D-c4da array.

Dim1 (int 4Byte)	Dim2 (int 4Byte)	Dim3 (int 4Byte)	Dim4 (int 4Byte)	Dim5 (int 4Byte)	Dim6 (int 4Byte)	Dim7 (int 4Byte)	Dim8 (int 4Byte)	Dim1×Dim2×Dim3×Dim4×Dim5×Dim6×Dim7×Dim8×[real (double 8Byte); Imaginary (double 8Byte)]

Pseudo-code

```
procedure Save_8DArray (data: 8Dcomplex )
      yy, zz, ww, dd5, dd6, dd7, dd8,
dx, dy, dz, dw, d5, d6, d7, d8: Integer;
begin
d8=Length (data );
d7=Length (data[0] );
d6=Length (data[0,0] );
d5=Length (data[0,0,0] );
dw=Length (data[0,0,0,0] );
           dz=Length (data[0,0,0,0,0] );
           dy=Length (data[0,0,0,0,0,0] );
           dx=Length (data[0,0,0,0,0,0,0] );
           file.Write ( dx, 4 );
           file.Write ( dy, 4 );
file.Write ( dz, 4 );
file.Write ( dw, 4 );
file.Write ( d5, 4 );
file.Write ( d6, 4 );
file.Write ( d7, 4 );
file.Write ( d8, 4 );
for dd8=0 to _d8-1 do
 for dd7=0 to _d7-1 do
  for dd6=0 to _d6-1 do
   for dd5=0 to _d5-1 do
    for ww=0 to _dW-1 do
     for zz=0 to _dZ-1 do
      for yy=0 to _dY-1 do
       file1.Write (data [ww,zz,yy,0], 2*8* dx)
end
```

*.int

Binary file, no header

Format for 1D integer data array: consists of N values (integer values)

```
N×[int-4Byte]
```

*.3dt

Binary file, with header

Format for storing 3D real vectors:

```
N×[X[double(8Byte)]; Y[double(8Byte)]; Z[double(8Byte)]] real value triplets
```

N(integer-4Byte)	N×triplet(double-8Byte)

*.trc

Proprietary file format from Teledyne LeCroy used from digital oscilloscopes