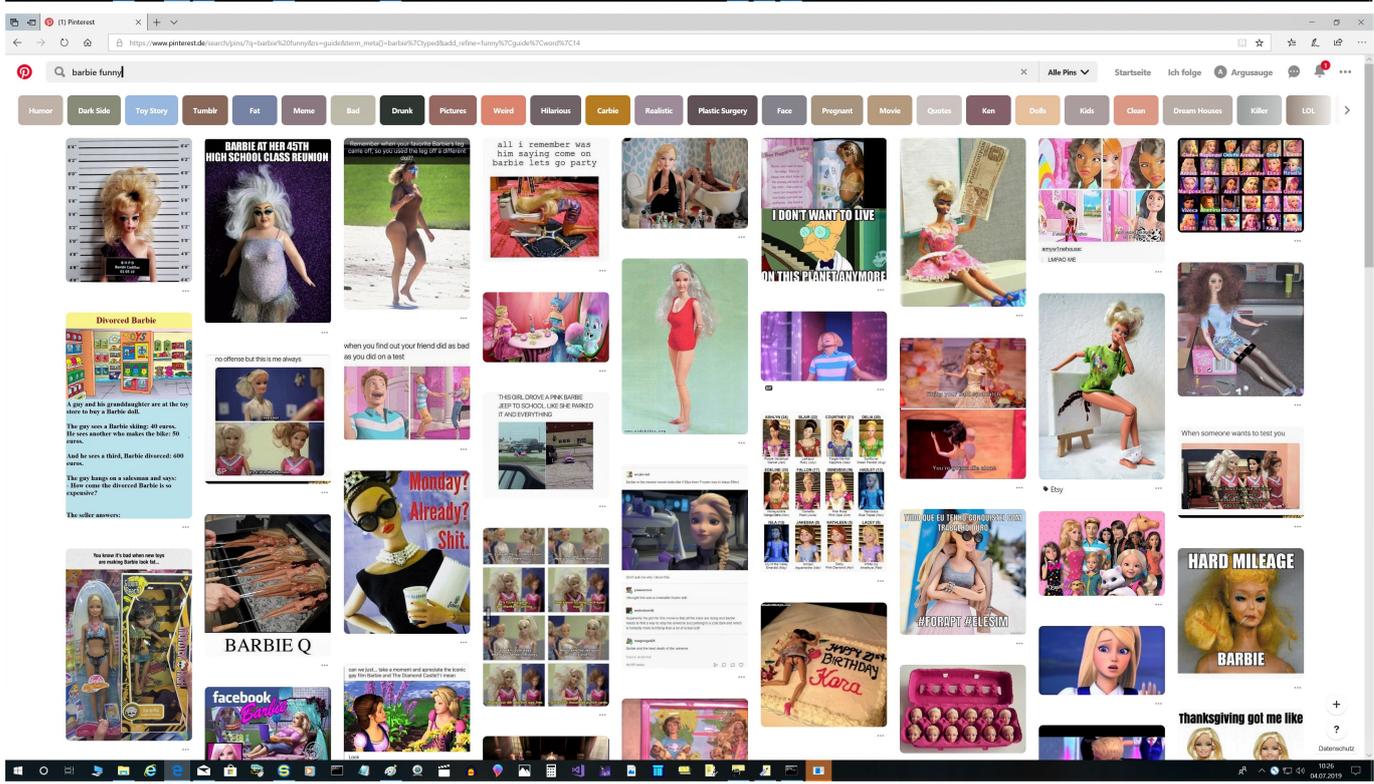
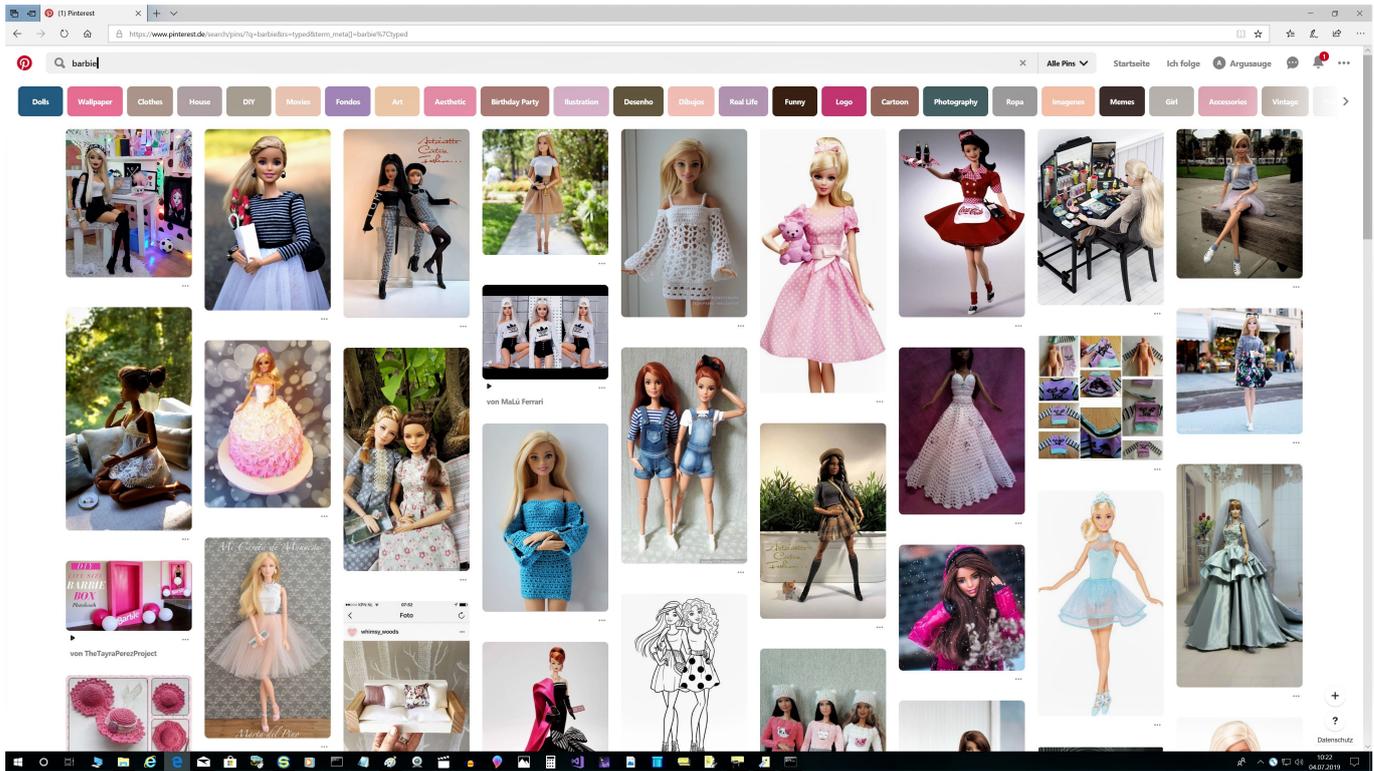


# BarbiePuppen2019a.pdf

Diese Datei ist eine Pseudo-Pinwand zum Thema Barbie Puppen von Mattel  
Außerdem dient sie für Formatierungsstudien für eine Doktorarbeit

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt achitektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf



# 1 Vorspann

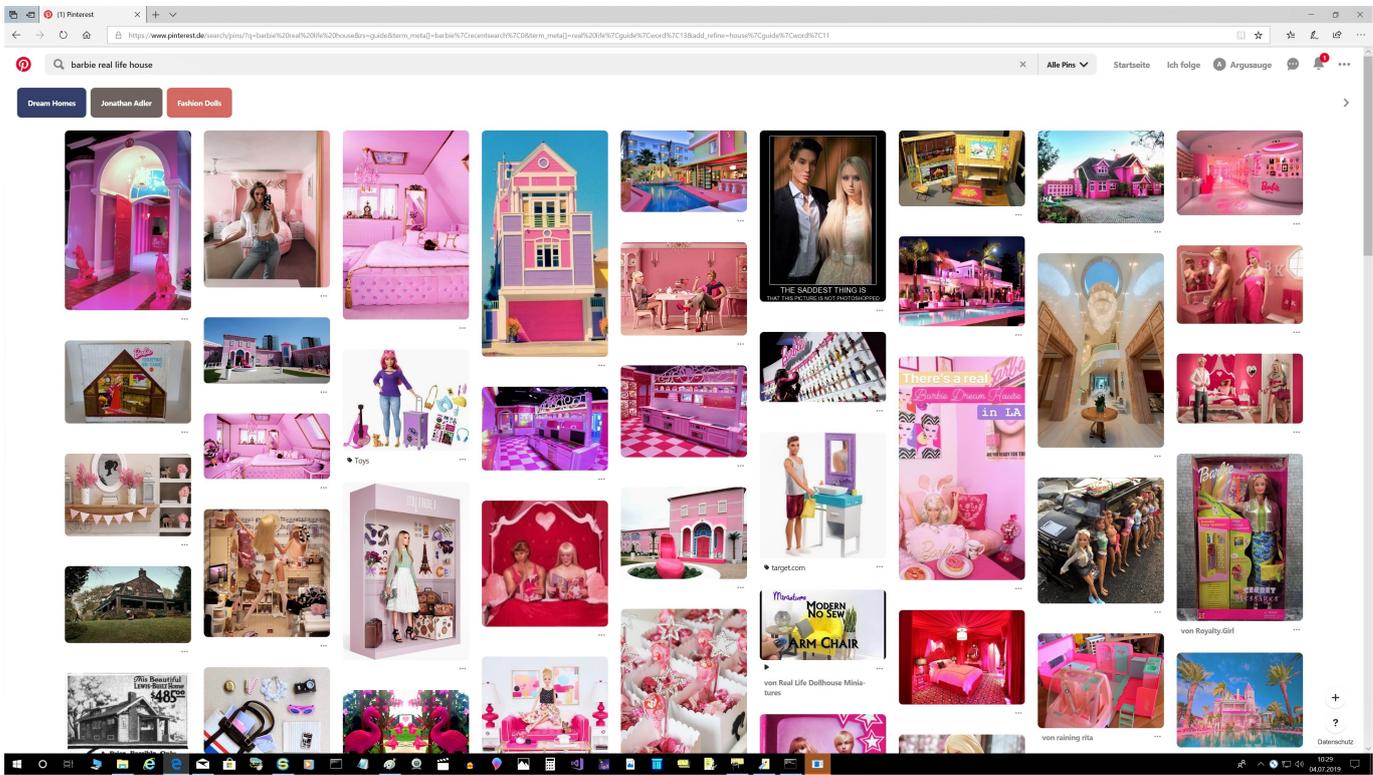
Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt architektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

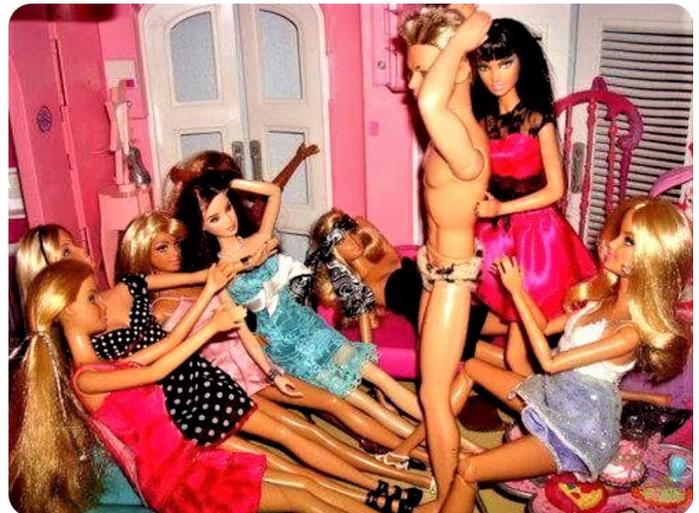
## Inhaltsverzeichnis

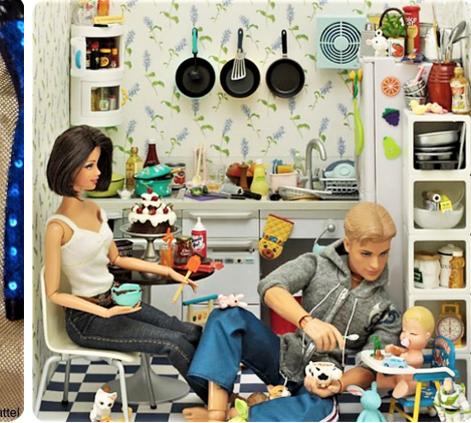
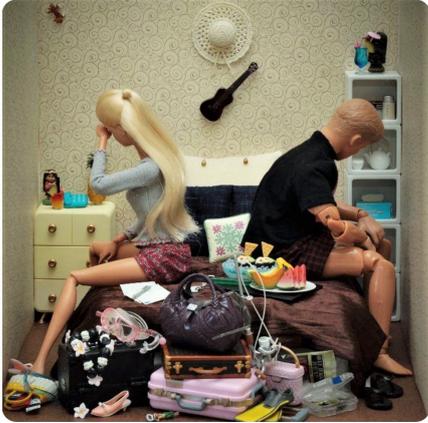
|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Vorspann</b>   | <b>2</b>  |
| <b>2</b> | <b>Parallel</b>   | <b>9</b>  |
| <b>3</b> | <b>Listings</b>   | <b>10</b> |
| <b>4</b> | <b>Verbatim</b>   | <b>11</b> |
| <b>5</b> | <b>DspLib Functions</b>                                 | <b>16</b> |
| 5.1      | MathLibrary Functions . . . . .                         | 16        |
| 5.1.1    | GETMAT GET MATH library constansts . . . . .            | 17        |
| 5.1.2    | SETMAT SET MATH library constansts . . . . .            | 18        |
| 5.1.3    | MATLIB MATH LIBrary main computation function . . . . . | 19        |
| 5.1.4    | MLWRIT Math Library WRITe function . . . . .            | 23        |
| 5.2      | Input Output Functions . . . . .                        | 35        |
| 5.2.1    | GETMAT GET MATH library constansts . . . . .            | 36        |
| 5.2.2    | FLWRIT FiLe WRITe function . . . . .                    | 40        |

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt architektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

<http://www.institut-wolfgang-renner.de/SchlossFuenfeckInhalte2019a.pdf>  
SchlossFuenfeckInhalte2019a.pdf



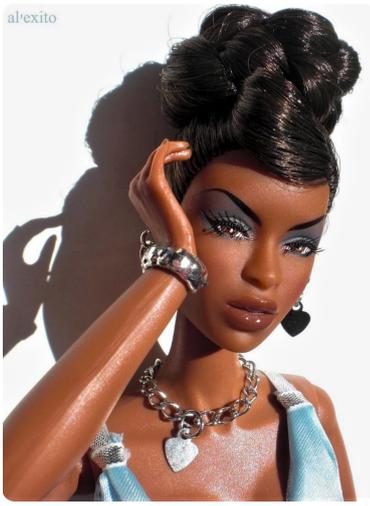




REAL BARBIE GIRLS







@Elian\_Stellar



©2017 Mattel



@LookStyleDolls



## 2 Parallel

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt achitektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

Links

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt achitektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

Rechts

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt achitektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt achitektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt achitektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

xxx

### 3 Listings

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt architektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument habe ich nie gelernt. Meine Generation hat sich vorwiegend mit den Themen beschäftigt, denen ich auch zugeneigt war. Aktiver Musikunterricht fand oftmals überhaupt nicht statt. Heute ist das Musizieren bei Jugendlichen wesentlich populärer als in meiner Generation. Dafür ist beispielsweise Elektronikbasteln und allgemeines Werken ziemlich out.

xxx

## 4 Verbatim

Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt achitektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument

XXXX



Die vorliegende Datei zeigt einige Bilder, welche ich gezeichnet habe. Am Ende sind zwei Bücher von meinem Großonkel dazugefügt. Es gibt architektonische Zeichnungen. Sowie körperliche Darstellungen. Ich habe mich von klein auf sehr für das Bauwesen interessiert. Meine guten Schulfächer waren die MINT-Fächer (Mathematik, Informatik, Naturwissenschaften und Technik), sowie graphische (Malen) und bildende Kunst (Töpfern usw.). Ein Musikinstrument





## 5 DspLib Functions

### 5.1 MathLibrary Functions







```

;* DL1 (MATLIB, 1IARXYP15, IA3ANG, P64RES, P64AR1, P64AR2) !! P64RES = ARCOS2 (P64AR1/P64AR2; IA3ANG) M1AR2COS PPPI *
;* DL1 (MATLIB, 1IARXYP16, IA3ANG, P64RES, P64AR1, P64AR2) !! P64RES = ARTAN2 (P64AR1/P64AR2; IA3ANG) M1AR2TAN PPPI *
;* DL1 (MATLIB, 1IARXYP17, IA3ANG, P64RES, P64AR1, P64AR2) !! P64RES = ARKOT2 (P64AR1/P64AR2; IA3ANG) M1AR2KOT PPPI *
;* DL1 (MATLIB, 1IARXYP18, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = EXPDUA [2.000] ~P64AR1 M1EXPDUA PPUU *
;* DL1 (MATLIB, 1IARXYP19, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = LOGDUA [00002] (P64AR1) M1LOGDUA PPUU *
;*
;* DL1 (MATLIB, 1IARXYP20, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = P64AR1 + D08AR2 M1MATADD PPDU *
;* DL1 (MATLIB, 1IARXYP21, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = P64AR1 - D08AR2 M1MATSUB PPDU *
;* DL1 (MATLIB, 1IARXYP22, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = P64AR1 * D08AR2 M1MATMUL PPDU *
;* DL1 (MATLIB, 1IARXYP23, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = P64AR1 / D08AR2 M1MATDIV PPDU *
;* DL1 (MATLIB, 1IARXYP24, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = D08AR2 - P64AR1 M1REVSUB PPDU *
;* DL1 (MATLIB, 1IARXYP25, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = D08AR2 / P64AR1 M1REVDIV PPDU *
;* DL1 (MATLIB, 1IARXYP26, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = P64AR1 ^ D08AR2 M1MATPWR PPDU *
;* DL1 (MATLIB, 1IARXYP27, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = MAROOT [D8AR2] (P64AR1) M1MAROOT PPDU *
;* DL1 (MATLIB, 1IARXYP28, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = D08AR2 ~ P64AR1 M1MATEXP PPDU *
;* DL1 (MATLIB, 1IARXYP29, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = MATLOG [D8AR2] (P64AR1) M1MATLOG PPDU *
;*
;* DL1 (MATLIB, 1IARXYP30, UNUSED, UNUSED, P64AR1, D08AR2) !! UNUSED = MATCMP (P64AR1, D08AR2) M1MATCMP UPDU *
;* DL1 (MATLIB, 1IARXYP31, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = MATMIN (P64AR1, D08AR2) M1MATMIN PPDU *
;* DL1 (MATLIB, 1IARXYP32, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = MATMAX (P64AR1, D08AR2) M1MATMAX PPDU *
;* DL1 (MATLIB, 1IARXYP33, UNUSED, P64RES, P64AR1, D08AR2) !! P64RES = MATSGN (P64AR1, D08AR2) M1MATSGN PPDU *
;* DL1 (MATLIB, 1IARXYP34, IA3ANG, P64RES, P64AR1, D08AR2) !! P64RES = ARSIN2 (P64AR1/D08AR2; IA3ANG) M1AR2SIN PPDI *
;* DL1 (MATLIB, 1IARXYP35, IA3ANG, P64RES, P64AR1, D08AR2) !! P64RES = ARCOS2 (P64AR1/D08AR2; IA3ANG) M1AR2COS PPDI *
;* DL1 (MATLIB, 1IARXYP36, IA3ANG, P64RES, P64AR1, D08AR2) !! P64RES = ARTAN2 (P64AR1/D08AR2; IA3ANG) M1AR2TAN PPDI *
;* DL1 (MATLIB, 1IARXYP37, IA3ANG, P64RES, P64AR1, D08AR2) !! P64RES = ARKOT2 (P64AR1/D08AR2; IA3ANG) M1AR2KOT PPDI *
;* DL1 (MATLIB, 1IARXYP38, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = EXPDUA [2.000] ~P64AR1 M1EXPDUA PPUU *
;* DL1 (MATLIB, 1IARXYP39, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = LOGDUA [00002] (P64AR1) M1LOGDUA PPUU *
;*
;* DL1 (MATLIB, 1IARXYP40, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = P64AR1 + I04AR2 M1MATADD PPIU *
;* DL1 (MATLIB, 1IARXYP41, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = P64AR1 - I04AR2 M1MATSUB PPIU *
;* DL1 (MATLIB, 1IARXYP42, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = P64AR1 * I04AR2 M1MATMUL PPIU *
;* DL1 (MATLIB, 1IARXYP43, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = P64AR1 / I04AR2 M1MATDIV PPIU *
;* DL1 (MATLIB, 1IARXYP44, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = I04AR2 - P64AR1 M1REVSUB PPIU *
;* DL1 (MATLIB, 1IARXYP45, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = I04AR2 / P64AR1 M1REVDIV PPIU *
;* DL1 (MATLIB, 1IARXYP46, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = P64AR1 ^ I04AR2 M1MATPWR PPIU *
;* DL1 (MATLIB, 1IARXYP47, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = MAROOT [I4AR2] (P64AR1) M1MAROOT PPIU *
;* DL1 (MATLIB, 1IARXYP48, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = I04AR2 ~ P64AR1 M1MATEXP PPIU *
;* DL1 (MATLIB, 1IARXYP49, UNUSED, P64RES, P64AR1, I04AR2) !! P64RES = MATLOG [I4AR2] (P64AR1) M1MATLOG PPIU *
;*
;* DL1 (MATLIB, 1IARXYP50, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = P64AR1 ^ 000002 M1SQSQUARE PPUU *
;* DL1 (MATLIB, 1IARXYP51, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = P64AR1 ^ 000003 M1MACUBE PPUU *
;* DL1 (MATLIB, 1IARXYP52, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = SQROOT [00002] (P64AR1) M1SQROOT PPUU *
;* DL1 (MATLIB, 1IARXYP53, UNUSED, P64RES, P64AR1, UNUSED) !! P64RES = CBROOT [00003] (P64AR1) M1CBROOT PPUU *
;* DL1 (MATLIB, 1IARXYP54, IA3BIN, P64RES, P64AR1, IA2BIN) !! P64RES = MATBIN (P64AR1, IA2BIN; IA3BIN) M1MATBIN PPII *
;* DL1 (MATLIB, 1IARXYP55, IA3GAM, P64RES, P64AR1, IA2GAM) !! P64RES = GAMFAC (P64AR1, IA2GAM; IA3GAM) M1GAMFAC PPII *
;* DL1 (MATLIB, 1IARXYP56, IA3STO, P64RES, P64AR1, UNUSED) !! P64RES = ADDSTO (P64AR1, UNUSED; IA3STO) M1ADDSTO PPUU *
;* DL1 (MATLIB, 1IARXYP57, IA3STO, P64RES, P64AR1, UNUSED) !! P64RES = SUBSTO (P64AR1, UNUSED; IA3STO) M1SUBSTO PPUU *
;* DL1 (MATLIB, 1IARXYP58, IA3STO, P64RES, P64AR1, UNUSED) !! P64RES = MATSTO (P64AR1, UNUSED; IA3STO) M1MATSTO PPUU *
;* DL1 (MATLIB, 1IARXYP59, IA3STO, P64RES, UNUSED, UNUSED) !! P64RES = RECALL (UNUSED, UNUSED; IA3STO) M1RECALL PUUI *
;*
;* DL1 (MATLIB, 1IARXYP60, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TAYSIN (P64AR1, IA2TAY; IA3ANG) M1TAYSIN PPII *
;* DL1 (MATLIB, 1IARXYP61, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TAYCOS (P64AR1, IA2TAY; IA3ANG) M1TAYCOS PPII *
;* DL1 (MATLIB, 1IARXYP62, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TAYTAN (P64AR1, IA2TAY; IA3ANG) M1TAYTAN PPII *
;* DL1 (MATLIB, 1IARXYP63, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TAYKOT (P64AR1, IA2TAY; IA3ANG) M1TAYKOT PPII *
;* DL1 (MATLIB, 1IARXYP64, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TYASIN (P64AR1, IA2TAY; IA3ANG) M1TYASIN PPII *
;* DL1 (MATLIB, 1IARXYP65, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TYACOS (P64AR1, IA2TAY; IA3ANG) M1TYACOS PPII *
;* DL1 (MATLIB, 1IARXYP66, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TYATAN (P64AR1, IA2TAY; IA3ANG) M1TYATAN PPII *
;* DL1 (MATLIB, 1IARXYP67, IA3ANG, P64RES, P64AR1, IA2TAY) !! P64RES = TYAKOT (P64AR1, IA2TAY; IA3ANG) M1TYAKOT PPII *
;* DL1 (MATLIB, 1IARXYP68, UNUSED, P64RES, P64AR1, IA2TAY) !! P64RES = TAYEXP (P64AR1, IA2TAY; IA3BAS) M1TAYEXP PPII *
;* DL1 (MATLIB, 1IARXYP69, UNUSED, P64RES, P64AR1, IA2TAY) !! P64RES = TAYLOG (P64AR1, IA2TAY; IA3BAS) M1TAYLOG PPII *
;*
;* DL1 (MATLIB, 1IARXYP70, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = MATSIN (P64AR1, IA2TAY; IA3ANG) M1MATSIN PPUU *
;* DL1 (MATLIB, 1IARXYP71, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = MATCOS (P64AR1, IA2TAY; IA3ANG) M1MATCOS PPUU *
;* DL1 (MATLIB, 1IARXYP72, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = MATTAN (P64AR1, IA2TAY; IA3ANG) M1MATTAN PPUU *
;* DL1 (MATLIB, 1IARXYP73, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = MATKOT (P64AR1, IA2TAY; IA3ANG) M1MATKOT PPUU *
;* DL1 (MATLIB, 1IARXYP74, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = ARCSIN (P64AR1, IA2TAY; IA3ANG) M1ARCSIN PPUU *
;* DL1 (MATLIB, 1IARXYP75, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = ARCCOS (P64AR1, IA2TAY; IA3ANG) M1ARCCOS PPUU *
;* DL1 (MATLIB, 1IARXYP76, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = ARCTAN (P64AR1, IA2TAY; IA3ANG) M1ARCTAN PPUU *
;* DL1 (MATLIB, 1IARXYP77, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = ARCKOT (P64AR1, IA2TAY; IA3ANG) M1ARCKOT PPUU *
;* DL1 (MATLIB, 1IARXYP78, IA3ANG, P64RES, P64AR1, UNUSED) !! P64RES = CVTANG (P64AR1, IA2TAY; IA3ANG) M1CVTANG PPUU *
;* DL1 (MATLIB, 1IARXYP79, IA3CST, P64RES, P64AR1, UNUSED) !! P64RES = GETCST (P64AR1, IA2TAY; IA3CST) M1GETCST PPUU *
;*
;* DL1 (MATLIB, 1IARXYP80, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = MATCOP (C64AR1, UNUSED; IA3CVT) M1MATCOP PCUI *
;* DL1 (MATLIB, 1IARXYP81, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = CHGSGN (C64AR1, UNUSED; IA3CVT) M1CHGSGN PCUI *
;* DL1 (MATLIB, 1IARXYP82, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = MATINT (C64AR1, UNUSED; IA3CVT) M1MATINT PCUI *
;* DL1 (MATLIB, 1IARXYP83, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = MANINT (C64AR1, UNUSED; IA3CVT) M1MANINT PCUI *
;* DL1 (MATLIB, 1IARXYP84, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = MFLOR (C64AR1, UNUSED; IA3CVT) M1MFLOR PCUI *
;* DL1 (MATLIB, 1IARXYP85, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = MACEIL (C64AR1, UNUSED; IA3CVT) M1MACEIL PCUI *
;* DL1 (MATLIB, 1IARXYP86, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = MANTIS (C64AR1, IA2TAY; IA3CVT) M1MANTIS PCUI *
;* DL1 (MATLIB, 1IARXYP87, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = FLOLOG (C64AR1, IA2TAY; IA3CVT) M1FLOLOG PCUI *
;* DL1 (MATLIB, 1IARXYP88, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = EXPDEC (C64AR1, IA2TAY; IA3CVT) M1EXPDEC PCUI *
;* DL1 (MATLIB, 1IARXYP89, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = LOGDEC (C64AR1, IA2TAY; IA3CVT) M1LOGDEC PCUI *
;*
;* DL1 (MATLIB, 1IARXYP90, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = SINHYP (C64AR1, IA2TAY; IA3CVT) M1SINHYP PCUI *
;* DL1 (MATLIB, 1IARXYP91, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = COSHYP (C64AR1, IA2TAY; IA3CVT) M1COSHYP PCUI *
;* DL1 (MATLIB, 1IARXYP92, IA3CVT, P64RES, C64AR1, UNUSED) !! P64RES = TANHYP (C64AR1, IA2TAY; IA3CVT) M1TANHYP PCUI *

```

```

;* DL1(MATLIB,1IARXYP93,IA3CVT,P64RES,C64AR1,UNUSED) !! P64RES = KOTHYP(C64AR1,IA2TAY;IA3CVT) M1KOTHYP PCUI *
;* DL1(MATLIB,1IARXYP94,IA3CVT,P64RES,C64AR1,UNUSED) !! P64RES = ARSINH(C64AR1,IA2TAY;IA3CVT) M1ARSINH PCUI *
;* DL1(MATLIB,1IARXYP95,IA3CVT,P64RES,C64AR1,UNUSED) !! P64RES = ARCOSH(C64AR1,IA2TAY;IA3CVT) M1ARCOSH PCUI *
;* DL1(MATLIB,1IARXYP96,IA3CVT,P64RES,C64AR1,UNUSED) !! P64RES = ARTANH(C64AR1,IA2TAY;IA3CVT) M1ARTANH PCUI *
;* DL1(MATLIB,1IARXYP97,IA3CVT,P64RES,C64AR1,UNUSED) !! P64RES = ARKOTH(C64AR1,IA2TAY;IA3CVT) M1ARKOTH PCUI *
;* DL1(MATLIB,1IARXYP98,IA3CVT,P64RES,C64AR1,UNUSED) !! P64RES = EXPNAT(C64AR1,IA2TAY;IA3CVT) M1EXPNAT PCUI *
;* DL1(MATLIB,1IARXYP99,IA3CVT,P64RES,C64AR1,UNUSED) !! P64RES = LOGNAT(C64AR1,IA2TAY;IA3CVT) M1LOGNAT PCUI *

```

```

;* CtrPrc <Pr> 00,01,..,98,99 P64ARG stands for all Precisions: I04,S04,D08,Q16,F32,F64,E06,E07,E08,..,N64

```

| Pr          | TYPE      | Bytes         | Ex | Fraction Bits | Maximal possible Value  | MaxDecimalFigures | ByteShf |
|-------------|-----------|---------------|----|---------------|-------------------------|-------------------|---------|
| I04VAL : 00 | sINT*04   | 04            | 00 | MAT LE 31     | +2147483647             | Little Endian 9.3 | 0008 03 |
| S04VAL : 01 | SNGL*04   | 04            | 08 | MAT LE 23     | +3.402823669E+000000038 | Little End 6.9    | 0008 03 |
| D08VAL : 02 | DBLE*08   | 08            | 11 | MAT LE 52     | +1.797693143E+000000308 | Little End 15.6   | 0008 03 |
| Q16VAL : 03 | QUAD*16   | 16            | 15 | MAT LE 111    | +1.189731581E+000004932 | Little End 33.4   | 0016 04 |
| F32VAL : 04 | GFLT*32   | 32            | 31 | MAT BE 223    | +4.226686143E+323228496 | Big Endian 67.1   | 0032 05 |
| F64VAL : 05 | PFLT*64   | 64            | 31 | MAT BE 479    | +4.226686143E+323228496 | Big Endian 144.2  | 0064 06 |
| E06VAL : 06 | XFLT*128  | 128           | 31 | MAT BE 991    | +4.226686143E+323228496 | Software 298.3    | 0128 07 |
| E07VAL : 07 | YFLT*256  | 256           | 31 | MAT BE 2 015  | +4.226686143E+323228496 | Software 606.5    | 0256 08 |
| E08VAL : 08 | ZFLT*512  | 512           | 31 | MAT BE 4 063  | +4.226686143E+323228496 | DWORD 1223.0      | 0512 09 |
| E09VAL : 09 | EFLT*1024 | 1 024         | 31 | MAT BE 8 159  | +4.226686143E+323228496 | Array 2456.1      | 1024 10 |
| E10VAL : 10 | EFLT*2048 | 2 048         | 31 | MAT BE 16 351 | +4.226686143E+323228496 | Constants 4 922   | 2048 11 |
| E11VAL : 11 | EFLT*4096 | 4 096         | 31 | MAT BE 32 735 | +4.226686143E+323228496 | Storage 9 854     | 4096 12 |
| E12VAL : 12 | EFLT*8192 | 8 192         | 31 | MAT 65 503    | +4.226686143E+323228496 | Given ! 19 718    | 8192 13 |
| E13VAL : 13 |           | 16 384        | 31 | 131 039       | +4.226686143E+323228496 | 39 446            | 16KB 14 |
| E14VAL : 14 |           | 32 768        | 31 | MAT 262 111   | +4.226686143E+323228496 | Software 78 903   | 32KB 15 |
| E15VAL : 15 |           | 65 536        | 31 | MAT 524 255   | +4.226686143E+323228496 | Software 157 816  | 64KB 16 |
| E16VAL : 16 |           | 131 072       | 31 | 1 048 543     | +4.226686143E+323228496 | 315 643           | 128K 17 |
| E17VAL : 17 |           | 262 144       | 31 | 2 097 119     | +4.226686143E+323228496 | 631 296           | 256K 18 |
| E18VAL : 18 |           | 524 288       | 31 | 4 194 271     | +4.226686143E+323228496 | DWORD 1 262 601   | 512K 19 |
| E19VAL : 19 |           | 1 048 576     | 31 | 8 388 575     | +4.226686143E+323228496 | Array 2 525 212   | 01MB 20 |
| E20VAL : 20 |           | 2 097 152     | 31 | 16 777 183    | +4.226686143E+323228496 | DWORD 5 050 435   | 02MB 21 |
| E21VAL : 21 |           | 4 194 304     | 31 | 33 554 399    | +4.226686143E+323228496 | ARRAY 10 008 880  | 04MB 22 |
| E22VAL : 22 |           | 8 388 608     | 31 | 67 108 831    | +4.226686143E+323228496 | 20 201 771        | 08MB 23 |
| E23VAL : 23 |           | 16 777 216    | 31 | 134 217 695   | +4.226686143E+323228496 | Expon 40 403 552  | 16MB 24 |
| E24VAL : 24 |           | 33 554 432    | 31 | 268 435 423   | +4.226686143E+323228496 | First 80 807 114  | 32MB 25 |
| E25VAL : 25 |           | 67 108 864    | 31 | 536 870 879   | +4.226686143E+323228496 | 161 614 238       | 64MB 26 |
| E26VAL : 26 |           | 134 217 728   | 31 | 1 073 741 791 | +4.226686143E+323228496 | 323 228 486       | 128M 27 |
| E27VAL : 27 |           | 268 435 456   | 31 | 2 147 483 615 | +4.226686143E+323228496 | 646 456 983       | 256M 28 |
| E28VAL : 28 |           | 536 870 912   | 31 | 4 294 967 263 | +4.226686143E+323228496 | 1 292 913 976     | 512M 29 |
| E29VAL : 29 |           | 1 073 741 824 | 31 | 8 589 934 559 | +4.226686143E+323228496 | 2 585 827 963     | 01GB 30 |

|             |         |    |    |              |                         |                 |         |
|-------------|---------|----|----|--------------|-------------------------|-----------------|---------|
| F02VAL : 30 | WFLT*02 | 02 | 08 | MAT SNGL 07  | +3.402823669E+000000038 | WORD Array 2.1  | 0008 03 |
| F04VAL : 31 | SFLT*04 | 04 | 15 | MAT TEMP 15  | +1.189731581E+000004932 | WORD Array 4.5  | 0008 03 |
| F08VAL : 32 | DFLT*08 | 08 | 15 | MAT TEMP 47  | +1.189731581E+000004932 | WORD Array 14.1 | 0016 04 |
| F16VAL : 33 | QFLT*16 | 16 | 15 | MAT TEMP 111 | +1.189731581E+000004932 | WORD Array 33.4 | 0032 05 |
| Q15VAL : 34 | QHLG*16 | 16 | 11 | MAT DBLE 104 | +1.797693143E+000000308 | HLG2 Dbles 31.3 | 0032 05 |
| F02VAL : 35 | WFLT*02 | 02 | 08 | X87 SNGL 07  | +3.402823669E+000000038 | WORD Array 2.1  | 0008 03 |
| F04VAL : 36 | SFLT*04 | 04 | 15 | X87 TEMP 15  | +1.189731581E+000004932 | WORD Array 4.5  | 0008 03 |
| F08VAL : 37 | DFLT*08 | 08 | 15 | X87 TEMP 47  | +1.189731581E+000004932 | WORD Array 14.1 | 0016 04 |
| F16VAL : 38 | QFLT*16 | 16 | 15 | X87 TEMP 063 | +1.189731581E+000004932 | WORD Array 18.9 | 0032 05 |
| Q15VAL : 39 | QHLG*16 | 16 | 11 | X87 DBLE 104 | +1.797693143E+000000308 | HLG2 Dbles 31.3 | 0032 05 |

| Pr          | TYPE    | Bytes | Ex | Fraction Bits | Maximal possible Value  | MaxDecimalFigures | ByteShf |
|-------------|---------|-------|----|---------------|-------------------------|-------------------|---------|
| I01VAL : 40 | uINT*01 | 01    | 00 | MAT LE 08     | (Little Endian) 255     | Little Endian 2.4 | 0008 03 |
| I01VAL : 41 | sINT*01 | 01    | 00 | MAT LE 07     | +127                    | Little End 2.1    | 0008 03 |
| I02VAL : 42 | uINT*02 | 02    | 00 | MAT LE 16     | 65535                   | Little End 4.8    | 0008 03 |
| I02VAL : 43 | sINT*02 | 02    | 00 | MAT LE 15     | +32768                  | Little End 4.5    | 0008 03 |
| I04VAL : 44 | uINT*04 | 04    | 00 | MAT LE 32     | 4294967295              | Little End 9.6    | 0008 03 |
| T10VAL : 45 | TEMP*10 | 10    | 15 | MAT LE 63     | +1.189731581E+000004932 | Little End 18.9   | 0016 04 |
| K08VAL : 46 | uINT*08 | 08    | 00 | MAT LE 64     | +1.844674407E+000000019 | Little End 19.2   | 0016 04 |
| K08VAL : 47 | sINT*08 | 08    | 00 | MAT LE 63     | +9.223372037E+000000018 | Little End 18.9   | 0016 04 |
| L16VAL : 48 | uINT*16 | 16    | 00 | MAT LE 128    | +3.402823669E+000000038 | Little End 38.5   | 0032 05 |
| L16VAL : 49 | sINT*16 | 16    | 00 | MAT LE 127    | +1.701411835E+000000038 | Little End 38.2   | 0032 05 |
| T06VAL : 50 | WTMP*06 | 06    | 31 | MAT BE 15     | +4.226686143E+323228496 | DWRD Array 4.5    | 0008 03 |
| T08VAL : 51 | STMP*08 | 08    | 31 | MAT BE 31     | +4.226686143E+323228496 | Big Endian 9.3    | 0008 03 |
| T12VAL : 52 | DTMP*12 | 12    | 31 | MAT BE 63     | +4.226686143E+323228496 | Big Endian 18.9   | 0016 04 |
| T20VAL : 53 | QTMP*20 | 20    | 31 | MAT BE 127    | +4.226686143E+323228496 | Big Endian 38.2   | 0032 05 |
| T36VAL : 54 | GTMP*36 | 36    | 31 | MAT BE 255    | +4.226686143E+323228496 | Big Endian 76.7   | 0032 05 |
| T68VAL : 55 | PTMP*68 | 68    | 31 | MAT 511       | +4.226686143E+323228496 | Big Endian 153.8  | 0064 06 |
| M32VAL : 56 | uINT*32 | 32    | 00 | MAT LE 256    | +1.157920892E+000000077 | Little End 77.1   | 0064 06 |
| M32VAL : 57 | sINT*32 | 32    | 00 | MAT LE 255    | +5.789604462E+000000076 | Little End 76.7   | 0064 06 |
| N64VAL : 58 | uINT*64 | 64    | 00 | MAT LE 512    | +1.340780793E+000000154 | Little End 154.1  | 0128 07 |
| N64VAL : 59 | sINT*64 | 64    | 00 | MAT LE 511    | +6.703903960E+000000153 | Little End 153.8  | 0128 07 |

| Pr          | TYPE    | Bytes | Ex | Fraction Bits | Maximal possible Value | MaxDecimalFigures | ByteShf |
|-------------|---------|-------|----|---------------|------------------------|-------------------|---------|
| I01VAL : 60 | uINT*01 | 01    | 00 | X86-32 08     | (Little Endian) 255    | Standard 2.4      | 0008 03 |
| I01VAL : 61 | sINT*01 | 01    | 00 | X86-32 07     | +127                   | Standard 2.1      | 0008 03 |
| I02VAL : 62 | uINT*02 | 02    | 00 | X86-32 16     | 65535                  | Standard 4.8      | 0008 03 |



### 5.1.4 MLWRIT Math Library WRITe function

The function MLWRIT reads unicode character string tokens from a diskfile or memoryarea. The diskfile or memory must be opened with the OPNFIL function. One data-token consists from an optional name string and an following optional data string. The name and the data part are separated with an equal character. Each token is ended with an token end character. This is either a comma or a semicolon.

Die Funktion MLWRIT liest unicode charakter String Tokens von einer Diskdatei oder Speicherbereich. Die Diskdatei oder der Speicherbereich muss vor mit der OPNFIL Funktion geöffnet worden sein. Ein Daten-Token besteht aus einem optionalen Namen und einem optionalen Datenteil, getrennt mit einen Gleichheitszeichen. Jeder Token wird entweder mit einem Komma oder Semikolon beendet.

```

*****
;* 12345&789012345678901234567890123456789012345678901234567890123456789012_& !!_901234567890123456789|1234567 *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !*----- *
;* !! *
;* !! 123456789012345678901234567890123456789 *
;* 12345 ERROR2 = DL1(MLWRIT,DABSIZ,DABCNT,DAECNT,DASOFF,DASCNT, & !! 01 *
;* & DACTRL,P64ARG,IA1FMT,IA2FMT,BASWSH,DASTRG) !! 02 General High Language *
;* 06 07 08 | 09 10 11 !! *
;* !! *
;* 12345 ERROR2 = DL1(MLWRIT,002560,000000,002560,DASOFF,DASCNT, & !! 01 <011 004 0016> *
;* & 1IAUSNPrBF,P64ARG,+00012,+00012,000010,DASTRG) !! 02 <0000:0458>hex *
;* !! *
;* !! *
;* 00 | 01 02 03 04 | 05 !! 12345678901234567890123456789 *
;* 12345 ERROR2 = FOR_MLWRIT(DABSIZ,DABCNT,DAECNT,DASOFF,DASCNT, & !! 01 *
;* & DACTRL,P64ARG,IA1FMT,IA2FMT,BASWSH,DASTRG) !! 02 Fortran direct call *
;* 06 07 08 | 09 10 11 !! *
;* !! *
;* 12345 ERROR2 = FOR_MLWRIT(002560,000000,002560,DASOFF,DASCNT, & !! 01 *
;* & 1IAUSNPrBF,P64ARG,+00012,+00012,000010,DASTRG) !! 02 *
;* !! *
*****
;* 1234567890123456789012345678901234567890123456789012345678901234567890123456 // 123456789|1234567 *
;* // - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* // - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* // - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* //----- *
;* // *
;* | IV IV IV IR | IR // *
;* Lab12345: Error2 = DL1(MLWRIT, DaBSiz, DaBCnt, DaECnt,&DaSOff;&DaSCnt, // 01 High Language *
;* DaCtrl,*P64Arg, IA1Fmt; IA2Fmt, BasWsh,*DaStrg); // 02 calling *
;* IV PR IV | IV IV CR // *
;* // *
;* Lab12345: Error2 = DL1(MLWRIT( 2560, 0000, 2560,&DaSOff;&DaSCnt, // 01 <011 004 0016> *
;* 1IAUSNPrBF,*P64Arg, +12; +12, 10,*DaStrg); // 02 <0000:0458>hex *
;* // *
;* // *
;* | IV IV IV IR | IR // *
;* Lab12345: Error2 = CPP_MLWRIT( DaBSiz, DaBCnt, DaECnt,&DaSOff;&DaSCnt, // 01 C/C++ direct *
;* DaCtrl,*P64Arg, IA1Fmt; IA2Fmt, BasWsh,*DaStrg); // 02 /C## call *
;* IV PR IV | IV IV CR // *
;* // *
;* Lab12345: Error2 = CPP_MLWRIT( 2560, 0000, 2560,&DaSOff;&DaSCnt, // 01 *
;* 1IAUSNPrBF,*P64Arg, +12; +12, 10,*DaStrg); // 02 *
;* // *
*****
;* *
;* CONTRL <1 IA RXY PrFu> = CTRFUN + 10*CTRBBB + 100*CTRPRC *
;* INT*4 1000*CTRDDD + 10000*CTRARY + 100000*CTRARX *
;* Typical <1 00 000 0200> 1000000*CTRRES + 10000000*CTRAX + 100000000*CTRINF *
;* *
*****
;* *
;* CtrFuncnt Presentation Style Base Int Fix Flt Rnd DecPoint NumSignChar BasStrg ExpoStrg *
;* *
;* <F> 0 -> String / Emergency 10 Not Yes Not Not Always ' '|'+''-' _ Not|Not _ May be *
;* 1 -> ES ExponentScientific BASWSH Not Yes Yes Yes Always ' '|'+''-' [_2]Not Yes *
;* 2 -> GS General Scientific BASWSH Not Yes Yes Yes Always ' '|'+''-' [_2]Not Not *
;* 3 -> EE ExponentEngineering BASWSH Not Yes Yes Yes Always ' '|'+''-' [_2]Not Yes *
;* 4 -> GE General Engineering BASWSH Not Yes Not Not Always ' '|'+''-' [_2]Not Not *
;* 5 -> Dec Decimal 10 Yes Yes Yes Yes Reals Not|'+''-' Not|'d' Not *
;* 6 -> Hex Hexadecimal 16 Yes Yes Yes Yes Reals Not|'+''-' 'h'|Not Not *
;* 7 -> Bin Binary 02 Yes Yes Yes Yes Reals Not|'+''-' 'b'|Not Not *
;* 8 -> Oct Octals 08 Yes Yes Yes Yes Reals Not|'+''-' 'o'|Not Not *
;* 9 -> Bas General Base Wished BASWSH Yes Yes Yes Yes Reals Not|'+''-' [_5]Not Not *

```



```

;* E23VAL : 23 | 16 777 216 : 31 : 134 217 695 | +4.226686143E+323228496 : Expon 40 403 552 | 16MB 24 *
;* E24VAL : 24 | 33 554 432 : 31 : 268 435 423 | +4.226686143E+323228496 : First 80 807 114 | 32MB 25 *
;* E25VAL : 25 | 67 108 864 : 31 : 536 870 879 | +4.226686143E+323228496 : 161 614 238 | 64MB 26 *
;* E26VAL : 26 | 134 217 728 : 31 : 1 073 741 791 | +4.226686143E+323228496 : 323 228 486 | 128M 27 *
;* E27VAL : 27 | 268 435 456 : 31 : 2 147 483 615 | +4.226686143E+323228496 : 646 456 983 | 256M 28 *
;* E28VAL : 28 | 536 870 912 : 31 : 4 294 967 263 | +4.226686143E+323228496 : 1 292 913 976 | 512M 29 *
;* E29VAL : 29 | 1 073 741 824 : 31 : 8 589 934 559 | +4.226686143E+323228496 : 2 585 827 963 | 01GB 30 *
*-----*
;* F02VAL : 30 | WFLT*02 02 : 08 : MAT SNGL 07 | +3.402823669E+000000038 : WORD Array 2.1 | 0008 03 *
;* F04VAL : 31 | SFLT*04 04 : 15 : MAT TEMP 15 | +1.189731581E+000004932 : WORD Array 4.5 | 0008 03 *
;* F08VAL : 32 | DFLT*08 08 : 15 : MAT TEMP 47 | +1.189731581E+000004932 : WORD Array 14.1 | 0016 04 *
;* F16VAL : 33 | QFLT*16 16 : 15 : MAT TEMP 111 | +1.189731581E+000004932 : WORD Array 33.4 | 0032 05 *
;* Q15VAL : 34 | QHLG*16 16 : 11 : MAT DBLE 104 | +1.797693143E+000000308 : HLG2 Dbles 31.3 | 0032 05 *
*-----*
;* F02VAL : 35 | WFLT*02 02 : 08 : X87 SNGL 07 | +3.402823669E+000000038 : WORD Array 2.1 | 0008 03 *
;* F04VAL : 36 | SFLT*04 04 : 15 : X87 TEMP 15 | +1.189731581E+000004932 : WORD Array 4.5 | 0008 03 *
;* F08VAL : 37 | DFLT*08 08 : 15 : X87 TEMP 47 | +1.189731581E+000004932 : WORD Array 14.1 | 0016 04 *
;* F16VAL : 38 | QFLT*16 16 : 15 : X87 TEMP 063 | +1.189731581E+000004932 : WORD Array 18.9 | 0032 05 *
;* Q15VAL : 39 | QHLG*16 16 : 11 : X87 DBLE 104 | +1.797693143E+000000308 : HLG2 Dbles 31.3 | 0032 05 *
*-----*
;* P64VAL : Pr | TYPE Bytes : Ex : Fraction Bits | Maximal possible Value : MaxDecimalFigures | ByteShf *
*-----*
;* I01VAL : 40 | uINT*01 01 : 00 : MAT LE 08 | (Little Endian ) 255 : Little Endian 2.4 | 0008 03 *
;* I01VAL : 41 | sINT*01 01 : 00 : MAT LE 07 | +127 : Little End 2.1 | 0008 03 *
;* I02VAL : 42 | uINT*02 02 : 00 : MAT LE 16 | 65535 : Little End 4.8 | 0008 03 *
;* I02VAL : 43 | sINT*02 02 : 00 : MAT LE 15 | +32768 : Little End 4.5 | 0008 03 *
;* I04VAL : 44 | uINT*04 04 : 00 : MAT LE 32 | 4294967295 : Little End 9.6 | 0008 03 *
;* T10VAL : 45 | TEMP*10 10 : 15 : MAT LE 63 | +1.189731581E+000004932 : Little End 18.9 | 0016 04 *
;* K08VAL : 46 | uINT*08 08 : 00 : MAT LE 64 | +1.844674407E+000000019 : Little End 19.2 | 0016 04 *
;* K08VAL : 47 | sINT*08 08 : 00 : MAT LE 63 | +9.223372037E+000000018 : Little End 18.9 | 0016 04 *
;* L16VAL : 48 | uINT*16 16 : 00 : MAT LE 128 | +3.402823669E+000000038 : Little End 38.5 | 0032 05 *
;* L16VAL : 49 | sINT*16 16 : 00 : MAT LE 127 | +1.701411835E+000000038 : Little End 38.2 | 0032 05 *
*-----*
;* TSTVAL : 50 | WTMP*06 06 : 31 : MAT BE 15 | +4.226686143E+323228496 : DWRD Array 4.5 | 0008 03 *
;* T08VAL : 51 | STMP*08 08 : 31 : MAT BE 31 | +4.226686143E+323228496 : Big Endian 9.3 | 0008 03 *
;* T12VAL : 52 | DTMP*12 12 : 31 : MAT BE 63 | +4.226686143E+323228496 : Big Endian 18.9 | 0016 04 *
;* T20VAL : 53 | QTMP*20 20 : 31 : MAT BE 127 | +4.226686143E+323228496 : Big Endian 38.2 | 0032 05 *
;* T36VAL : 54 | GTMP*36 36 : 31 : MAT BE 255 | +4.226686143E+323228496 : Big Endian 76.7 | 0032 05 *
;* T68VAL : 55 | PTMP*68 68 : 31 : MAT 511 | +4.226686143E+323228496 : Big Endian 153.8 | 0064 06 *
;* M32VAL : 56 | uINT*32 32 : 00 : MAT LE 256 | +1.157920892E+000000077 : Little End 77.1 | 0064 06 *
;* M32VAL : 57 | sINT*32 32 : 00 : MAT LE 255 | +5.789604462E+000000076 : Little End 76.7 | 0064 06 *
;* N64VAL : 58 | uINT*64 64 : 00 : MAT LE 512 | +1.340780793E+000000154 : Little End 154.1 | 0128 07 *
;* N64VAL : 59 | sINT*64 64 : 00 : MAT LE 511 | +6.703903960E+000000153 : Little End 153.8 | 0128 07 *
*-----*
;* P64VAL : Pr | TYPE Bytes : Ex : Fraction Bits | Maximal possible Value : MaxDecimalFigures | ByteShf *
*-----*
;* I01VAL : 60 | uINT*01 01 : 00 : X86-32 08 | (Little Endian ) 255 : Standard 2.4 | 0008 03 *
;* I01VAL : 61 | sINT*01 01 : 00 : X86-32 07 | +127 : Standard 2.1 | 0008 03 *
;* I02VAL : 62 | uINT*02 02 : 00 : X86-32 16 | 65535 : Standard 4.8 | 0008 03 *
;* I02VAL : 63 | sINT*02 02 : 00 : X86-32 15 | +32768 : Standard 4.5 | 0008 03 *
;* I04VAL : 64 | uINT*04 04 : 00 : X86-32 32 | 4294967295 : Standard 9.6 | 0008 03 *
;* I04VAL : 65 | sINT*04 04 : 00 : X86-32 31 | +2147483647 : Standard 9.3 | 0008 03 *
;* K08VAL : 66 | uINT*08 08 : 00 : X86-64 64 | +1.844674407E+000000019 : Standard 19.2 | 0016 04 *
;* K08VAL : 67 | sINT*08 08 : 00 : X86-64 63 | +9.223372037E+000000018 : Standard 18.9 | 0016 04 *
;* L16VAL : 68 | uINT*16 16 : 00 : X86-128 128 | +3.402823669E+000000038 : Future ! 38.5 | 0032 05 *
;* L16VAL : 69 | sINT*16 16 : 00 : X86-128 127 | +1.701411835E+000000038 : Future ! 38.2 | 0032 05 *
*-----*
;* I04VAL : 70 | sINT*04 04 : 00 : X87-064 31 | (LitEnd) +2147483647 : Standard 9.3 | 0008 03 *
;* S04VAL : 71 | SNGL*04 04 : 08 : X87-064 23 | +3.402823669E+000000038 : Standard 6.9 | 0008 03 *
;* D08VAL : 72 | DBLE*08 08 : 11 : X87-064 52 | +1.797693143E+000000308 : Standard 15.6 | 0016 04 *
;* Q16VAL : 73 | QUAD*16 16 : 15 : X87-128 111 | +1.189731581E+000004932 : Partial! 33.4 | 0032 05 *
;* F32VAL : 74 | GFLT*32 32 : 31 : X87-256 223 | +4.226686143E+323228496 : Future ? 67.1 | 0032 05 *
;* F64VAL : 75 | PFLT*64 64 : 31 : X87-512 479 | +4.226686143E+323228496 : Future ? 144.2 | 0064 06 *
;* M32VAL : 76 | uINT*32 32 : 00 : X86-256 256 | +1.157920892E+000000077 : Future ? 77.1 | 0064 06 *
;* M32VAL : 77 | sINT*32 32 : 00 : X86-256 255 | +5.789604462E+000000076 : Future ? 76.7 | 0064 06 *
;* N64VAL : 78 | uINT*64 64 : 00 : X86-512 512 | +1.340780793E+000000154 : Future ? 154.1 | 0128 07 *
;* N64VAL : 79 | sINT*64 64 : 00 : X86-512 511 | +6.703903960E+000000153 : Future ? 153.8 | 0128 07 *
*-----*
;* P##VAL : Pr | TYPE Bytes : Ex : Fraction Bits | Maximal possible Value : MaxDecimalFigures | ByteShf *
*-----*
;* I01VAL : 80 | uINT*01 01 : 00 : X87-64 08 | (Little Endian ) 255 : Standard 2.4 | 0008 03 *
;* I01VAL : 81 | sINT*01 01 : 00 : X87-64 07 | +127 : Standard 2.1 | 0008 03 *
;* I02VAL : 82 | uINT*02 02 : 00 : X87 LE 16 | 65535 : Standard 4.8 | 0008 03 *
;* I02VAL : 83 | sINT*02 02 : 00 : X87 LE 15 | +32768 : Standard 4.5 | 0008 03 *
;* I04VAL : 84 | uINT*04 04 : 00 : X87 LE 32 | 4294967295 : Standard 9.6 | 0008 03 *
;* T10VAL : 85 | TEMP*10 10 : 15 : X87 LE 63 | +1.189731581E+000004932 : Standard 18.9 | 0016 04 *
;* K08VAL : 86 | uINT*08 08 : 00 : X87 LE 64 | +1.844674407E+000000019 : Partial! 19.2 | 0016 04 *
;* K08VAL : 87 | sINT*08 08 : 00 : X87-64 63 | +9.223372037E+000000018 : Standard 18.9 | 0016 04 *
;* L16VAL : 88 | uINT*16 16 : 00 : X87-128 128 | +3.402823669E+000000038 : Future ! 38.5 | 0032 05 *
;* L16VAL : 89 | sINT*16 16 : 00 : X87-128 127 | +1.701411835E+000000038 : Future ! 38.2 | 0032 05 *
*-----*
;* TSTVAL : 90 | WTMP*06 ? ? 06 : 31 : X87-64 15 | +4.226686143E+323228496 : Standard 4.5 | 0008 03 *

```



```

;* WritData FNwLin FreeByteSize WIFillNull WInewLine [L1 &n L2 &z ] Style *
;* *
;* <N> 0 -> FullByteSize 00 00 [L1 L2 &z.....] Removed *
;* 1 -> FullByteSize 00 01 [L1 13,10 L2 &z.....] Windows *
;* 2 -> FullByteSize 00 02 [L1 10 L2 &z.....] Unix *
;* 3 -> FullByteSize 00 03 [L1 13 L2 &z.....] Macintosh *
;* 4 -> FullByteSize 00 05 [L1 &n L2 &z.....] Strings *
;* 5 -> FBS-NullSize 01 00 [L1 L2 &z.....|o] Removed *
;* 6 -> FBS-NullSize 01 01 [L1 13,10 L2 &z...|o] Windows *
;* 7 -> FBS-NullSize 01 02 [L1 10 L2 &z.....|o] Unix *
;* 8 -> FBS-NullSize 01 03 [L1 13 L2 &z.....|o] Macintosh *
;* 9 -> FBS-NullSize 01 05 [L1 &n L2 &z.....|o] Strings *
;* *
;* ===== *
;* *
;* Return values: *
;* ===== *
;* *
;* ERROR1 not yet defined INT*04 *
;*****
;* MyMlWrit My own MatLib Write function *
;* *
;* #=====#=====#=====#=====#=====#=====# *
;* | +12345678901234567_10E+10 : 00 |1234567890123456|1234567890123456|1234567890123456|1234567890123456| *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Exponent Scientific : A2 | ES(+16|+A2|B3) | ES(+16|-A2|B3) | ES(-16|+A2|B3) | ES(-16|-A2|B3) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.234567890123456_10E+00 : 00 | 1.E+00| 1.E+00| 1.E+00 | 1.E+00 | *
;* | -1.2345000000000000_10E+00 : 01 | -1.2E+00| -1.2E+00|-1.2E+00 | -1.2E+00 | *
;* | +1.2345000000000000_10E+00 : 02 | 1.23E+00| +1.23E+00| 1.23E+00 | +1.23E+00 | *
;* | -1.2345000000000000_10E+00 : 03 | -1.234E+00| -1.234E+00|-1.234E+00 | -1.234E+00 | *
;* | +1.2345000000000000_10E+00 : 04 | 1.234 5E+00| +1.234 5E+00| 1.234 5E+00 | +1.234 5E+00 | *
;* | -1.2345000000000000_10E+00 : 05 | -1.234 50E+00| -1.234 50E+00|-1.234 50E+00 | -1.234 50E+00 | *
;* | +1.2345000000000000_10E+00 : 06 | 1.234 500E+00| +1.234 500E+00| 1.234 500E+00 | +1.234 500E+00 | *
;* | -1.2345000000000000_10E+00 : 07 - 16 | -1.234 500 0E+00|-1.234 500 0E+00|-1.234 500 0E+00|-1.234 500 0E+00| *
;* | : | | | | | *
;* | +1.2345000000000000_10E+00 : 17 . . | 1.234 5E+00| +1.234 5E+00| 1.234 5E+00 | +1.234 5E+00 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.234567890123456_12E+00 : 00 | 1._12E+00| 1._12E+00| 1._12E+00 | 1._12E+00 | *
;* | -1.2345000000000000_12E+00 : 01 | -1.2_12E+00| -1.2_12E+00|-1.2_12E+00 | -1.2_12E+00 | *
;* | +1.2345000000000000_12E+00 : 02 | 1.23_12E+00| +1.23_12E+00| 1.23_12E+00 | +1.23_12E+00 | *
;* | -1.2345000000000000_12E+00 : 03 | -1.234_12E+00| -1.234_12E+00|-1.234_12E+00 | -1.234_12E+00 | *
;* | +1.2345000000000000_12E+00 : 04 | 1.234 5_12E+00| +1.234 5_12E+00| 1.234 5_12E+00 | +1.234 5_12E+00 | *
;* | -1.2345000000000000_10E+00 : 05 - 16 | -1.234 50_12E+00|-1.234 50_12E+00|-1.234 50_12E+00|-1.234 50_12E+00| *
;* | : | | | | | *
;* | +1.2345000000000000_12E+00 : 17 . . | 1.234 5_12E+00| +1.234 5_12E+00| 1.234 5_12E+00 | +1.234 5E+00 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Exponent Engineering : A2 | EE(+16|+A2|B3) | EE(+16|-A2|B3) | EE(-16|+A2|B3) | EE(-16|-A2|B3) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_10E+01 : 00 | 12.E+00| 12.E+00| 12.E+00 | 12.E+00 | *
;* | -1.2345000000000000_10E+01 : 01 | -12.3E+00| -12.3E+00|-12.3E+00 | -12.3E+00 | *
;* | +1.2345000000000000_10E+01 : 02 | 12.34E+00| +12.34E+00| 12.34E+00 | +12.34E+00 | *
;* | -1.2345000000000000_10E+01 : 03 | -12.345E+00| -12.345E+00|-12.345E+00 | -12.345E+00 | *
;* | +1.2345000000000000_10E+01 : 04 | 12.345 0E+00| 12.345 0E+00| 12.345 0E+00 | +12.345 0E+00 | *
;* | -1.2345000000000000_10E+01 : 05 | -12.345 00E+00| -12.345 00E+00|-12.345 00E+00 | -12.345 00E+00 | *
;* | +1.2345000000000000_10E+01 : 06 - 16 | 12.345 000E+00| 12.345 000E+00| 12.345 000E+00 | +12.345 000E+00 | *
;* | : | | | | | *
;* | +1.2345000000000000_10E+01 : 17 . . | 12.345E+00| +12.345E+00| 12.345E+00 | +12.345E+00 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_12E+01 : 00 | 12._12E+00| 12._12E+00| 12._12E+00 | 12._12E+00 | *
;* | -1.2345000000000000_12E+01 : 01 | -12.3_12E+00| -12.3_12E+00|-12.3_12E+00 | -12.3_12E+00 | *
;* | +1.2345000000000000_12E+01 : 02 | 12.34_12E+00| +12.34_12E+00| 12.34_12E+00 | +12.34_12E+00 | *
;* | -1.2345000000000000_12E+01 : 03 | -12.345_12E+00| -12.345_12E+00|-12.345_12E+00 | -12.345_12E+00 | *
;* | +1.2345000000000000_12E+01 : 04 - 16 | 12.345 0_12E+00| 12.345 0_12E+00| 12.345 0_12E+00| +12.345 0_12E+00| *
;* | : | | | | | *
;* | +1.2345000000000000_12E+01 : 17 . . | 12.345_12E+00| +12.345_12E+00| 12.345_12E+00 | +12.345_12E+00 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Exponent Engineering : A2 | EE(+16|+A2|B3) | EE(+16|-A2|B3) | EE(-16|+A2|B3) | EE(-16|-A2|B3) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_10E+02 : 00 | 123.E+00| 123.E+00| 123.E+00 | 123.E+00 | *
;* | -1.2345000000000000_10E+02 : 01 | -123.4E+00| -123.4E+00|-123.4E+00 | -123.4E+00 | *
;* | +1.2345000000000000_10E+02 : 02 | 123.45E+00| +123.45E+00| 123.45E+00 | +123.45E+00 | *
;* | -1.2345000000000000_10E+02 : 03 | -123.450E+00| -123.450E+00|-123.450E+00 | -123.450E+00 | *
;* | +1.2345000000000000_10E+02 : 04 | 123.450 0E+00| 123.450 0E+00| 123.450 0E+00 | +123.450 0E+00 | *
;* | -1.2345000000000000_10E+02 : 05 | -123.450 00E+00| -123.450 00E+00|-123.450 00E+00 | -123.450 00E+00 | *
;* | +1.2345000000000000_10E+02 : 06 - 15 | 123.450 000E+00| 123.450 000E+00| 123.450 000E+00| +123.450 000E+00| *
;* | : | | | | | *
;* | +1.2345000000000000_10E+02 : 17 . . | 123.45E+00| +123.45E+00| 123.45E+00 | +123.45E+00 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_12E+02 : 00 | 123._12E+00| 123._12E+00| 123._12E+00 | 123._12E+00 | *
;* | -1.2345000000000000_12E+02 : 01 | -123.4_12E+00| -123.4_12E+00|-123.4_12E+00 | -123.4_12E+00 | *
;* | +1.2345000000000000_12E+02 : 02 | 123.45_12E+00| +123.45_12E+00| 123.45_12E+00 | +123.45_12E+00 | *

```

```

;* | -1.2345000000000000_12E+02 : 03 - 16 | -123.450_12E+00| -123.450_12E+00|-123.450_12E+00 |-123.450_12E+00 | *
;* | : | | | | | | *
;* | +1.2345000000000000_12E+02 : 17 . . | 123.45E+00| +123.45E+00| 123.45E+00 | +123.45E+00 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | General Scientific/Engin : A2 | GS(+16|+A2|B3) | GS(+16|-A2|B3) | GS(-16|+A2|B3) | GS(-16|-A2|B3) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_10E+00 : 00 | 1.| 1.| 1. | 1. | *
;* | -1.2345000000000000_10E+00 : 01 | -1.2| -1.2|-1.2 | -1.2 | *
;* | +1.2345000000000000_10E+00 : 02 | 1.23| +1.23| 1.23 | +1.23 | *
;* | -1.2345000000000000_10E+00 : 03 | -1.234| -1.234|-1.234 | -1.234 | *
;* | +1.2345000000000000_10E+00 : 04 | 1.234 5| +1.234 5| 1.234 5 | +1.234 5 | *
;* | -1.2345000000000000_10E+00 : 05 | -1.234 50| -1.234 50|-1.234 50 | -1.234 50 | *
;* | +1.2345000000000000_10E+00 : 06 | 1.234 500| +1.234 500| 1.234 500 | +1.234 500 | *
;* | -1.2345000000000000_10E+00 : 07 | -1.234 500 0| -1.234 500 0|-1.234 500 0 | -1.234 500 0 | *
;* | -1.2345000000000000_10E+00 : 08 | 1.234 500 00| -1.234 500 00|-1.234 500 00 | -1.234 500 00 | *
;* | -1.2345000000000000_10E+00 : 09 | -1.234 500 000| -1.234 500 000|-1.234 500 000 | -1.234 500 000 | *
;* | -1.2345000000000000_10E+00 : 10 - 16 | 1.234 500 000 0| -1.234 500 000 0|-1.234 500 000 0 | -1.234 500 000 0 | *
;* | : | | | | | | *
;* | +1.2345000000000000_10E+00 : 17 . . | 1.234 5| +1.234 5| 1.234 5 | +1.234 5 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_12E+00 : 00 | 1._12| 1._12| 1._12 | 1._12 | *
;* | -1.2345000000000000_12E+00 : 01 | -1.2_12| -1.2_12|-1.2_12 | -1.2_12 | *
;* | +1.2345000000000000_12E+00 : 02 | 1.23_12| +1.23_12| 1.23_12 | +1.23_12 | *
;* | -1.2345000000000000_12E+00 : 03 | -1.234_12| -1.234_12|-1.234_12 | -1.234_12 | *
;* | +1.2345000000000000_12E+00 : 04 | 1.234 5_12| +1.234 5_12| 1.234 5_12 | +1.234 5_12 | *
;* | -1.2345000000000000_12E+00 : 05 | -1.234 50_12| -1.234 50_12|-1.234 50_12 | -1.234 50_12 | *
;* | +1.2345000000000000_12E+00 : 06 | 1.234 500_12| +1.234 500_12| 1.234 500_12 | +1.234 500_12 | *
;* | -1.2345000000000000_12E+00 : 07 | -1.234 500 0_12| -1.234 500 0_12|-1.234 500 0_12 | -1.234 500 0_12 | *
;* | -1.2345000000000000_12E+00 : 08 - 15 | 1.234 500 00_12| -1.234 500 00_12|-1.234 500 00_12 | -1.234 500 00_12 | *
;* | : | | | | | | *
;* | +1.2345000000000000_10E+00 : 17 . . | 1.234 5_12| +1.234 5_12| 1.234 5_12 | +1.234 5_12 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | General Scientific/Engin : A2 | GS(+16|+A2|B3) | GS(+16|-A2|B3) | GS(-16|+A2|B3) | GS(-16|-A2|B3) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Integer +12345_10 : 00 | 12 345| 12 345| 12 345 | 12 345 | *
;* | -12345_10 : 01 | -12 345| -12 345|-12 345 | -12 345 | *
;* | +12345_10 : 02 | 12 345| +12 345| 12 345 | +12 345 | *
;* | -12345_10 : 03 | -12 345| -12 345|-12 345 | -12 345 | *
;* | +12345_10 : 04 | 12 345| +12 345| 12 345 | +12 345 | *
;* | -12345_10 : 05 | -12 345| -12 345|-12 345 | -12 345 | *
;* | +12345_10 : 06 | 012 345| +012 345| 012 345 | +012 345 | *
;* | -12345_10 : 07 | -0 012 345| -0 012 345|-0 012 345 | -0 012 345 | *
;* | +12345_10 : 08 | 00 012 345| +00 012 345| 00 012 345 | +00 012 345 | *
;* | -12345_10 : 09 | -000 012 345| -000 012 345|-000 012 345 | -000 012 345 | *
;* | +12345_10 : 10 | 0 000 012 345| +0 000 012 345| 0 000 012 345 | +0 000 012 345 | *
;* | -12345_10 : 11 | -00 000 012 345| -00 000 012 345|-00 000 012 345 | -00 000 012 345 | *
;* | +12345_10 : 12 - 16 | 000 000 012 345| +000 000 012 345| 000 000 012 345| +000 000 012 345 | *
;* | : | | | | | | *
;* | +12345_10 : 17 . . | 12 345| +12 345| 12 345 | +12 345 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Integer +12345_12 : 00 | 12 345_12| 12 345_12| 12 345_12 | 12 345_12 | *
;* | -12345_12 : 01 | -12 345_12| -12 345_12|-12 345_12 | -12 345_12 | *
;* | +12345_12 : 02 | 12 345_12| +12 345_12| 12 345_12 | +12 345_12 | *
;* | -12345_12 : 03 | -12 345_12| -12 345_12|-12 345_12 | -12 345_12 | *
;* | +12345_12 : 04 | 12 345_12| +12 345_12| 12 345_12 | +12 345_12 | *
;* | -12345_12 : 05 | -12 345_12| -12 345_12|-12 345_12 | -12 345_12 | *
;* | +12345_12 : 06 | 012 345_12| +012 345_12| 012 345_12 | +012 345_12 | *
;* | -12345_12 : 07 | -0 012 345_12| -0 012 345_12|-0 012 345_12 | -0 012 345_12 | *
;* | +12345_12 : 08 | 00 012 345_12| +00 012 345_12| 00 012 345_12 | +00 012 345_12 | *
;* | -12345_12 : 09 | -000 012 345_12| -000 012 345_12|-000 012 345_12 | -000 012 345_12 | *
;* | : | | | | | | *
;* | +12345_10 : 17 . . | 12 345_12| +12 345_12| 12 345_12 | +12 345_12 | *
;* #-----#-----#-----#-----#-----# *
;* | General Decimal (5) : A2 | Dec(+16|+A2|3) | Dec(+16|-A2|3) | Dec(-16|+A2|3) | Dec(-16|-A2|3) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_10E+00 : 00 | 1.| 1.| 1. | 1. | *
;* | -1.2345000000000000_10E+00 : 01 | -1.2| -1.2|-1.2 | -1.2 | *
;* | +1.2345000000000000_10E+00 : 02 | 1.23| +1.23| 1.23 | +1.23 | *
;* | -1.2345000000000000_10E+00 : 03 | -1.234| -1.234|-1.234 | -1.234 | *
;* | +1.2345000000000000_10E+00 : 04 | 1.234 5| +1.234 5| 1.234 5 | +1.234 5 | *
;* | -1.2345000000000000_10E+00 : 05 | -1.234 50| -1.234 50|-1.234 50 | -1.234 50 | *
;* | +1.2345000000000000_10E+00 : 06 | 1.234 500| +1.234 500| 1.234 500 | +1.234 500 | *
;* | -1.2345000000000000_10E+00 : 07 | -1.234 500 0| -1.234 500 0|-1.234 500 0 | -1.234 500 0 | *
;* | +1.2345000000000000_10E+00 : 08 | 1.234 500 00| +1.234 500 00| 1.234 500 00 | +1.234 500 00 | *
;* | -1.2345000000000000_10E+00 : 09 | -1.234 500 000| -1.234 500 000|-1.234 500 000 | -1.234 500 000 | *
;* | +1.2345000000000000_10E+00 : 10 - 16 | 1.234 500 000 0| +1.234 500 000 0| 1.234 500 000 0 | +1.234 500 000 0 | *
;* | : | | | | | | *
;* | +1.2345000000000000_10E+00 : 17 . . | 1.234 5| +1.234 5| 1.234 5 | +1.234 5 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Integer +12345_10 : 00 | 12 345| 12 345| 12 345 | 12 345 | *
;* | -12345_10 : 01 | -12 345| -12 345|-12 345 | -12 345 | *
;* | +12345_10 : 02 | 12 345| +12 345| 12 345 | +12 345 | *

```

```

;* | -12345_10 : 03 | -12 345| -12 345|-12 345 | -12 345 | *
;* | +12345_10 : 04 | 12 345| +12 345| 12 345 | +12 345 | *
;* | -12345_10 : 05 | -12 345| -12 345|-12 345 | -12 345 | *
;* | +12345_10 : 06 | 012 345| +012 345| 012 345 | +012 345 | *
;* | -12345_10 : 07 | -0 012 345| -0 012 345|-0 012 345 | -0 012 345 | *
;* | +12345_10 : 08 | 00 012 345| +00 012 345| 00 012 345 | +00 012 345 | *
;* | -12345_10 : 09 | -000 012 345| -000 012 345|-000 012 345 | -000 012 345 | *
;* | +12345_10 : 10 | 0 000 012 345| +0 000 012 345| 0 000 012 345 | +0 000 012 345 | *
;* | -12345_10 : 11 | -00 000 012 345| -00 000 012 345|-00 000 012 345 | -00 000 012 345 | *
;* | +12345_10 : 12 - 16 | 000 000 012 345| +000 000 012 345| 000 000 012 345| +000 000 012 345| *
;* | : | | | | | *
;* | +12345_10 : 17 . . | 12 345| +12 345| 12 345 | +12 345 | *
;* +-----+-----+-----+-----+-----+
;* | General Hexadecimal (6) : A2 | Hex(+16|+A2|3) | Hex(+16|-A2|3) | Hex(-16|+A2|3) | Hex(-16|-A2|3) | *
;* +-----+-----+-----+-----+-----+
;* | +1.2345000000000000_16E+00 : 00 | 1.h| 1.h| 1.h | 1.h | *
;* | -1.2345000000000000_16E+00 : 01 | -1.2h| -1.2h|-1.2h | -1.2h | *
;* | +1.2345000000000000_16E+00 : 02 | 1.23h| +1.23h| 1.23h | +1.23h | *
;* | -1.2345000000000000_16E+00 : 03 | -1.234h| -1.234h|-1.234h | -1.234h | *
;* | +1.2345000000000000_16E+00 : 04 | 1.234 5h| +1.234 5h| 1.234 5h | +1.234 5h | *
;* | -1.2345000000000000_16E+00 : 05 | -1.234 50h| -1.234 50h|-1.234 50h | -1.234 50h | *
;* | +1.2345000000000000_16E+00 : 06 | 1.234 500h| +1.234 500h| 1.234 500h | +1.234 500h | *
;* | -1.2345000000000000_16E+00 : 07 | -1.234 500 0h| -1.234 500 0h|-1.234 500 0h | -1.234 500 0h | *
;* | +1.2345000000000000_16E+00 : 08 | 1.234 500 00h| +1.234 500 00h| 1.234 500 00h | +1.234 500 00h | *
;* | -1.2345000000000000_16E+00 : 09 - 16 | -1.234 500 000h| -1.234 500 000h|-1.234 500 000h | -1.234 500 000h | *
;* | : | | | | | *
;* | +1.2345000000000000_16E+00 : 17 . . | 1.234 5h| +1.234 5h| 1.234 5h | +1.234 5h | *
;* +-----+-----+-----+-----+-----+
;* | Integer +12345_16 : 00 | 12 345h| 12 345h| 12 345h | 12 345h | *
;* | -12345_16 : 01 | -12 345h| -12 345h|-12 345h | -12 345h | *
;* | +12345_16 : 02 | 12 345h| +12 345h| 12 345h | +12 345h | *
;* | -12345_16 : 03 | -12 345h| -12 345h|-12 345h | -12 345h | *
;* | +12345_16 : 04 | 12 345h| +12 345h| 12 345h | +12 345h | *
;* | -12345_16 : 05 | -12 345h| -12 345h|-12 345h | -12 345h | *
;* | +12345_16 : 06 | 012 345h| +012 345h| 012 345h | +012 345h | *
;* | -12345_16 : 07 | -0 012 345h| -0 012 345h|-0 012 345h | -0 012 345h | *
;* | +12345_16 : 08 | 00 012 345h| +00 012 345h| 00 012 345h | +00 012 345h | *
;* | -12345_16 : 09 | -000 012 345h| -000 012 345h|-000 012 345h | -000 012 345h | *
;* | +12345_16 : 10 | 0 000 012 345h| +0 000 012 345h| 0 000 012 345h | +0 000 012 345h | *
;* | -12345_16 : 11 - 16 | -00 000 012 345h| -00 000 012 345h|-00 000 012 345h | -00 000 012 345h | *
;* | +12345_16 : 12 - 16 | 000 000 012 345h| | | | *
;* | : | | | | | *
;* | +12345_16 : 17 . . | 12 345h| +12 345h| 12 345h | +12 345h | *
;* +-----+-----+-----+-----+-----+
;* | General Binary (7) : A2 | Bin(+16|+A2|3) | Bin(+16|-A2|3) | Bin(-16|+A2|3) | Bin(-16|-A2|3) | *
;* +-----+-----+-----+-----+-----+
;* | +1.1011000000000000_02E+00 : 00 | 1.b| 1.b| 1.b | 1.b | *
;* | -1.1011000000000000_02E+00 : 01 | -1.1b| -1.1b|-1.1b | -1.1b | *
;* | +1.1011000000000000_02E+00 : 02 | 1.10b| +1.10b| 1.10b | +1.10b | *
;* | -1.1011000000000000_02E+00 : 03 | -1.101b| -1.101b|-1.101b | -1.101b | *
;* | +1.1011000000000000_02E+00 : 04 | 1.101 1b| +1.101 1b| 1.101 1b | +1.101 1b | *
;* | -1.1011000000000000_02E+00 : 05 | -1.101 10b| -1.101 10b|-1.101 10b | -1.101 10b | *
;* | +1.1011000000000000_02E+00 : 06 | 1.101 100b| +1.101 100b| 1.101 100b | +1.101 100b | *
;* | -1.1011000000000000_02E+00 : 07 | -1.101 100 0b| -1.101 100 0b|-1.101 100 0b | -1.101 100 0b | *
;* | +1.1011000000000000_02E+00 : 08 | 1.101 100 00b| +1.101 100 00b| 1.101 100 00b | +1.101 100 00b | *
;* | -1.1011000000000000_02E+00 : 09 - 16 | -1.101 100 000b| -1.101 100 000b|-1.101 100 000b | -1.101 100 000b | *
;* | : | | | | | *
;* | +1.1011000000000000_02E+00 : 17 . . | 1.101 1b| +1.234 5b| 1.101 1b | +1.101 1b | *
;* +-----+-----+-----+-----+-----+
;* | Integer +11011_02 : 00 | 11 011b| 11 011b| 11 011b | 11 011b | *
;* | -11011_02 : 01 | -11 011b| -11 011b|-11 011b | -11 011b | *
;* | +11011_02 : 02 | 11 011b| +11 011b| 11 011b | +11 011b | *
;* | -11011_02 : 03 | -11 011b| -11 011b|-11 011b | -11 011b | *
;* | +11011_02 : 04 | 11 011b| +11 011b| 11 011b | +11 011b | *
;* | -11011_02 : 05 | -11 011b| -11 011b|-11 011b | -11 011b | *
;* | +11011_02 : 06 | 011 011b| +011 011b| 011 011b | +011 011b | *
;* | -11011_02 : 07 | -0 011 011b| -0 011 011b|-0 011 011b | -0 011 011b | *
;* | +11011_02 : 08 | 00 011 011b| +00 011 011b| 00 011 011b | +00 011 011b | *
;* | -11011_02 : 09 | -000 011 011b| -000 011 011b|-000 011 011b | -000 011 011b | *
;* | +11011_02 : 10 | 0 000 011 011b| +0 000 011 011b| 0 000 011 011b | +0 000 011 011b | *
;* | -11011_02 : 11 - 16 | -00 000 011 011b| -00 000 011 011b|-00 000 011 011b | -00 000 011 011b | *
;* | +11011_02 : 12 - 16 | 000 000 011 011b| | | | *
;* | : | | | | | *
;* | +11011_02 : 17 . . | 11 011b| +11 011b| 11 011b | +11 011b | *
;* +-----+-----+-----+-----+-----+
;* | General Octal (8) : A2 | Oct(+16|+A2|3) | Oct(+16|-A2|3) | Oct(-16|+A2|3) | Oct(-16|-A2|3) | *
;* +-----+-----+-----+-----+-----+
;* | +1.2345000000000000_08E+00 : 00 | 1.o| 1.o| 1.o | 1.o | *
;* | -1.2345000000000000_08E+00 : 01 | -1.2o| -1.2o|-1.2o | -1.2o | *
;* | +1.2345000000000000_08E+00 : 02 | 1.23o| +1.23o| 1.23o | +1.23o | *
;* | -1.2345000000000000_08E+00 : 03 | -1.234o| -1.234o|-1.234o | -1.234o | *
;* | +1.2345000000000000_08E+00 : 04 | 1.234 5o| +1.234 5o| 1.234 5o | +1.234 5o | *

```

```

;* | -1.2345000000000000_08E+00 : 05 | -1.234 50o| -1.234 50o|-1.234 50o | -1.234 50o | *
;* | +1.2345000000000000_08E+00 : 06 | 1.234 500o| +1.234 500o| 1.234 500o | +1.234 500o | *
;* | -1.2345000000000000_08E+00 : 07 | -1.234 500 0o| -1.234 500 0o|-1.234 500 0o | -1.234 500 0o | *
;* | -1.2345000000000000_08E+00 : 08 | 1.234 500 00o| -1.234 500 00o| 1.234 500 00o | -1.234 500 00o | *
;* | -1.2345000000000000_08E+00 : 09 - 16 | -1.234 500 000o| -1.234 500 000o|-1.234 500 000o | -1.234 500 000o | *
;* | : | | | | | *
;* | +1.2345000000000000_08E+00 : 17 . . | 1.234 5h| +1.234 5o| 1.234 5o | +1.234 5o | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Integer +12345_08 : 00 | 12 345o| 12 345o| 12 345o | 12 345o | *
;* | -12345_08 : 01 | -12 345o| -12 345o|-12 345o | -12 345o | *
;* | +12345_08 : 02 | 12 345o| +12 345o| 12 345o | +12 345o | *
;* | -12345_08 : 03 | -12 345o| -12 345o|-12 345o | -12 345o | *
;* | +12345_08 : 04 | 12 345o| +12 345o| 12 345o | +12 345o | *
;* | -12345_08 : 05 | -12 345o| -12 345o|-12 345o | -12 345o | *
;* | +12345_08 : 06 | 012 345o| +012 345o| 012 345o | +012 345o | *
;* | -12345_08 : 07 | -0 012 345o| -0 012 345o|-0 012 345o | -0 012 345o | *
;* | +12345_08 : 08 | 00 012 345o| +00 012 345o| 00 012 345o | +00 012 345o | *
;* | -12345_08 : 09 | -000 012 345o| -000 012 345o|-000 012 345o | -000 012 345o | *
;* | +12345_08 : 10 | 0 000 012 345o| +0 000 012 345o| 0 000 012 345o | +0 000 012 345o | *
;* | -12345_08 : 11 - 16 | -00 000 012 345o| -00 000 012 345o|-00 000 012 345o | -00 000 012 345o | *
;* | +12345_08 : 12 - 16 | 000 000 012 345o| | | | *
;* | : | | | | | *
;* | +12345_08 : 17 . . | 12 345o| +12 345o| 12 345h | +12 345o | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | General BaseWished (9) : A2 | Bas(+16|+A2|3) | Bas(+16|-A2|3) | Bas(-16|+A2|3) | Bas(-16|-A2|3) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.2345000000000000_12E+00 : 00 | 1._12| 1._12| 1._12 | 1._12 | *
;* | -1.2345000000000000_12E+00 : 01 | -1.2_12| -1.2_12|-1.2_12 | -1.2_12 | *
;* | +1.2345000000000000_12E+00 : 02 | 1.23_12| +1.23_12| 1.23_12 | +1.23_12 | *
;* | -1.2345000000000000_12E+00 : 03 | -1.234_12| -1.234_12|-1.234_12 | -1.234_12 | *
;* | +1.2345000000000000_12E+00 : 04 | 1.234 5_12| +1.234 5_12| 1.234 5_12 | +1.234 5_12 | *
;* | -1.2345000000000000_12E+00 : 05 | -1.234 50_12| -1.234 50_12|-1.234 50_12 | -1.234 50_12 | *
;* | +1.2345000000000000_12E+00 : 06 | 1.234 500_12| +1.234 500_12| 1.234 500_12 | +1.234 500_12 | *
;* | -1.2345000000000000_12E+00 : 07 | -1.234 500 0_12| -1.234 500 0_12|-1.234 500 0_12 | -1.234 500 0_12 | *
;* | -1.2345000000000000_12E+00 : 08 - 15 | 1.234 500 00_12| -1.234 500 00_12|-1.234 500 00_12| -1.234 500 00_12 | *
;* | : | | | | | *
;* | +1.2345000000000000_12E+00 : 16 . . | 1.234 5_12| +1.234 5_12| 1.234 5_12 | +1.234 5_12 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Integer +12345_12 : 00 | 12 345_12| 12 345_12| 12 345_12 | 12 345_12 | *
;* | -12345_12 : 01 | -12 345_12| -12 345_12|-12 345_12 | -12 345_12 | *
;* | +12345_12 : 02 | 12 345_12| +12 345_12| 12 345_12 | +12 345_12 | *
;* | -12345_12 : 03 | -12 345_12| -12 345_12|-12 345_12 | -12 345_12 | *
;* | +12345_12 : 04 | 12 345_12| +12 345_12| 12 345_12 | +12 345_12 | *
;* | -12345_12 : 05 | -12 345_12| -12 345_12|-12 345_12 | -12 345_12 | *
;* | +12345_12 : 06 | 012 345_12| +012 345_12| 012 345_12 | +012 345_12 | *
;* | -12345_12 : 07 | -0 012 345_12| -0 012 345_12|-0 012 345_12 | -0 012 345_12 | *
;* | +12345_12 : 08 | 00 012 345_12| +00 012 345_12| 00 012 345_12 | +00 012 345_12 | *
;* | -12345_12 : 09 - 15 | -000 012 345_12| -000 012 345_12|-000 012 345_12 | -000 012 345_12 | *
;* | : | | | | | *
;* | -12345_12 : 16 . . | -12 345_12| -12 345_12|-12 345_12 | -12 345_12 | *
;* #=====#=====#=====#=====#=====# *
;* #=====#=====#=====#=====#=====# *
;* | +1.23456789012345678_10E+10 |123456789012345678|123456789012345678|123456789012345678|123456789012345678| *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Exponent Scientific | ES(+18|+06|B03) | ES(+18|-06|B03) | ES(-18|+06|B03) | ES(-18|-06|B03) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.23456789012345678_10E-02 | 1.234 568E-02| +1.234 568E-02| 1.234 568E-02 | +1.234 568E-02 | *
;* | -1.23456789012345678_10E-01 | -1.234 568E-01| -1.234 568E-01|-1.234 568E-01 | -1.234 568E-01 | *
;* | +1.23456789012345678_10E+00 | 1.234 568E+00| +1.234 568E+00| 1.234 568E+00 | +1.234 568E+00 | *
;* | -1.23456789012345678_10E+01 | -1.234 568E+01| -1.234 568E+01|-1.234 568E+01 | -1.234 568E+01 | *
;* | +1.23456789012345678_10E+02 | 1.234 568E+02| +1.234 568E+02| 1.234 568E+02 | +1.234 568E+02 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Exponent Engineering | ES(+18|+06|B03) | ES(+18|-06|B03) | ES(-18|+06|B03) | ES(-18|-06|B03) | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.23456789012345678_10E-06 | 1.234 568E-06| +1.234 568E-06| 1.234 568E-06 | +1.234 568E-06 | *
;* | -1.23456789012345678_10E-05 | -12.345 679E-06| -12.345 679E-06|-12.345 679E-06 | -12.345 679E-06 | *
;* | +1.23456789012345678_10E-04 | 123.456 789E-06| +123.456 789E-06| 123.456 789E-06 | +123.456 789E-06 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.23456789012345678_10E-03 | 1.234 568E-03| +1.234 568E-03| 1.234 568E-03 | +1.234 568E-03 | *
;* | -1.23456789012345678_10E-02 | -12.345 679E-03| -12.345 679E-03|-12.345 679E-03 | -12.345 679E-03 | *
;* | +1.23456789012345678_10E-01 | 123.456 789E-03| +123.456 789E-03| 123.456 789E-03 | +123.456 789E-03 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.23456789012345678_10E+00 | 1.234 568E+00| +1.234 568E+00| 1.234 568E+00 | +1.234 568E+00 | *
;* | -1.23456789012345678_10E+01 | -12.345 679E+00| -12.345 679E+00|-12.345 679E+00 | -12.345 679E+00 | *
;* | +1.23456789012345678_10E+02 | 123.456 789E+00| +123.456 789E+00| 123.456 789E+00 | +123.456 789E+00 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | +1.23456789012345678_10E+03 | 1.234 568E+03| +1.234 568E+03| 1.234 568E+03 | +1.234 568E+03 | *
;* | -1.23456789012345678_10E+04 | -12.345 679E+03| -12.345 679E+03|-12.345 679E+03 | -12.345 679E+03 | *
;* | +1.23456789012345678_10E+05 | 123.456 789E+03| +123.456 789E+03| 123.456 789E+03 | +123.456 789E+03 | *
;* +-----+-----+-----+-----+-----+-----+ *
;* | Exponent Engineering | ES(+18|+17|B03) | ES(+18|-17|B03) | ES(-18|+17|B03) | ES(-18|-17|B03) | *

```

```

;* +-----+-----+-----+-----+-----+-----+
;* | +1.23456789012345678_10E-03 | 1.234 567 9E-03| +1.234 567 9E-03| 1.234 567 9E-03| +1.234 567 9E-03| *
;* | -1.23456789012345678_10E-02 | -12.345 678 9E-03| -12.345 678 9E-03| -12.345 678 9E-03| -12.345 678 9E-03| *
;* | +1.23456789012345678_10E-01 | 123.456 789 0E-03| +123.456 789 0E-03| 123.456 789 0E-03| +123.456 789 0E-03| *
;* +-----+-----+-----+-----+-----+-----+
;* | +1.23456789012345678_10E+00 | 1.234 567 9E+00| +1.234 567 9E+00| 1.234 568 9E+00| +1.234 567 9E+00| *
;* | -1.23456789012345678_10E+01 | -12.345 678 9E+00| -12.345 678 9E+00| -12.345 678 9E+00| -12.345 678 9E+00| *
;* | +1.23456789012345678_10E+02 | 123.456 789 0E+00| +123.456 789 0E+00| 123.456 789 0E+00| +123.456 789 0E+00| *
;* +-----+-----+-----+-----+-----+-----+
;*
;*
;* +-----+-----+-----+-----+-----+-----+
;* | Hexadecimal | Hex(+18|+04|B02) | Hex(+18|-04|B02) | Hex(-18|+04|B02) | Hex(-18|-04|B02) | *
;* +-----+-----+-----+-----+-----+-----+
;* | +1.23456789012345678_16E-02 | 0.00 00h | -0.00 00h | 0.00 00h | -0.00 00h | *
;* | +1.23456789012345678_16E-02 | 0.00 01h | +0.00 01h | 0.00 01h | +0.00 01h | *
;* | +1.23456789012345678_16E-02 | 0.00 12h | -0.00 12h | 0.00 12h | -0.00 12h | *
;* | +1.23456789012345678_16E-02 | 0.01 23h | +0.01 23h | 0.01 23h | +0.01 23h | *
;* | +1.23456789012345678_16E-02 | 0.12 34h | -0.12 34h | 0.12 34h | -0.12 34h | *
;* | +1.23456789012345678_16E+00 | 1.23 45h | +1.23 45h | 1.23 45h | +1.23 45h | *
;* | +1.23456789012345678_16E+01 | 12.34 56h | -12.34 56h | 12.34 56h | -12.34 56h | *
;* | +1.23456789012345678_16E+02 | 1 23.45 67h | +1 23.45 67h | 1 23.45 67h | +1 23.45 67h | *
;* | +1.23456789012345678_16E+03 | 12 34.56 78h | -12 34.56 78h | 12 34.56 78h | -12 34.56 78h | *
;* | +1.23456789012345678_16E+04 | 1 23 45.67 89h | +1 23 45.67 89h | 1 23 45.67 89h | +1 23 45.67 89h | *
;* | +1.23456789012345678_16E+05 | 12 34 56.78 90h | -12 34 56.78 90h | 12 34 56.78 90h | -12 34 56.78 90h | *
;* | +1.23456789012345678_16E+06 | 1 23 45 67.89 01h | +1 23 45 67.89 01h | 1 23 45 67.89 01h | +1 23 45 67.89 01h | *
;* | | | | | | *
;* | +1.23456789012345678_16E+07 | -1.23 45_16E+07 | -1.23 45_16E+07 | -1.23 45_16E+07 | -1.23 45_16E+07 | *
;* +-----+-----+-----+-----+-----+-----+
;* | Hexadecimal | Hex(+18|+08|B02) | Hex(+18|-08|B02) | Hex(-18|+08|B02) | Hex(-18|-08) | *
;* +-----+-----+-----+-----+-----+-----+
;* | Integer Input 00000000_16 | 00 00 00 00h | +00 00 00 00h | 00 00 00 00h | +00 00 00 00h | *
;* | 00000001_16 | 00 00 00 01h | -00 00 00 01h | 00 00 00 01h | -00 00 00 01h | *
;* | 00000012_16 | 00 00 00 12h | +00 00 00 12h | 00 00 00 12h | +00 00 00 12h | *
;* | 00000123_16 | 00 00 01 23h | -00 00 01 23h | 00 00 01 23h | -00 00 01 23h | *
;* | 00001234_16 | 00 00 12 34h | +00 00 12 34h | 00 00 12 34h | +00 00 12 34h | *
;* | 00012345_16 | 00 01 23 45h | -00 01 23 45h | 00 01 23 45h | -00 01 23 45h | *
;* | 00123456_16 | 00 12 34 56h | +00 12 34 56h | 00 12 34 56h | +00 12 34 56h | *
;* | 01234567_16 | 01 23 45 67h | -01 23 45 67h | 01 23 45 67h | -01 23 45 67h | *
;* | 12345678_16 | 12 34 56 78h | +12 34 56 78h | 12 34 56 78h | +12 34 56 78h | *
;* +-----+-----+-----+-----+-----+-----+
;* | 123456789_16 | 1 23 45 67 89h | -1 23 45 67 89h | 1 23 45 67 89h | -1 23 45 67 89h | *
;* | 123456789A_16 | 12 34 56 78 9Ah | +12 34 56 78 9Ah | 12 34 56 78 9Ah | +12 34 56 78 9Ah | *
;* | 123456789AB_16 | 1 23 45 67 89 ABh | -1 23 45 67 89 ABh | 1 23 45 67 89 ABh | -1 23 45 67 89 ABh | *
;* | 123456789ABC_16 | 12 34 56 78 9A BCh | | | | *
;* | | | -1.23 45 67_16E+11 | -1.23 45 67_16E+11 | -1.23 45 67_16E+11 | *
;* | 123456789ABCD_16 | -1.23 45 67_16E+12 | 1.23 45 67_16E+12 | 1.23 45 67_16E+12 | +1.23 45 67_16E+12 | *
;* +-----+-----+-----+-----+-----+-----+
;*
;* #=====#=====#=====#=====#=====#=====#=====#=====#=====#=====#=====#=====#=====#=====#=====#
;* | +1.23456789012345678_10E+10 | 123456789012345678 | 123456789012345678 | 123456789012345678 | 123456789012345678 | *
;* +-----+-----+-----+-----+-----+-----+
;*
;* +-----+-----+-----+-----+-----+-----+
;* | General Engineering (4) | ES(+18|+08|B00) | ES(+18|-08|B00) | ES(-18|+08|B00) | ES(-18|-08|B00) | *
;* +-----+-----+-----+-----+-----+-----+
;* | +1.23456789012345678_10E-12 | 1.23456789E-12 | +1.23456789E-12 | 1.23456789E-12 | +1.23456789E-12 | *
;* | -1.23456789012345678_10E-11 | -12.34567890E-12 | -12.34567890E-12 | -12.34567890E-12 | -12.34567890E-12 | *
;* | +1.23456789012345678_10E-10 | 123.45678901E-12 | +123.45678901E-12 | 123.45678901E-12 | +123.45678901E-12 | *
;* | -1.23456789012345678_10E-09 | -1.23456789E-09 | -1.23456789E-09 | -1.23456789E-09 | -1.23456789E-09 | *
;* +-----+-----+-----+-----+-----+-----+
;* | +1.23456789012345678_10E-08 | 0.00000001 | +0.00000001 | 0.00000001 | +0.00000001 | *
;* | -1.23456789012345678_10E-07 | -0.00000012 | -0.00000012 | -0.00000012 | -0.00000012 | *
;* | +1.23456789012345678_10E-06 | 0.00000123 | +0.00000123 | 0.00000123 | +0.00000123 | *
;* | -1.23456789012345678_10E-05 | -0.00001234 | -0.00001234 | -0.00001234 | -0.00001234 | *
;* | +1.23456789012345678_10E-04 | 0.00012345 | +0.00012345 | 0.00012345 | +0.00012345 | *
;* | -1.23456789012345678_10E-03 | -0.00123456 | -0.00123456 | -0.00123456 | -0.00123456 | *
;* | +1.23456789012345678_10E-02 | 0.01234567 | +0.01234567 | 0.01234567 | +0.01234567 | *
;* | -1.23456789012345678_10E-01 | -0.12345678 | -0.12345678 | -0.12345678 | -0.12345678 | *
;* | +1.23456789012345678_10E+00 | 1.23456789 | +1.23456789 | 1.23456789 | +1.23456789 | *
;* | -1.23456789012345678_10E+01 | -12.34567890 | -12.34567890 | -12.34567890 | -12.34567890 | *
;* | +1.23456789012345678_10E+02 | 123.45678901 | +123.45678901 | 123.45678901 | +123.45678901 | *
;* | -1.23456789012345678_10E+03 | -1234.56789012 | -1234.56789012 | -1234.56789012 | -1234.56789012 | *
;* | +1.23456789012345678_10E+04 | 12345.67890123 | +12345.67890123 | 12345.67890123 | +12345.67890123 | *
;* | -1.23456789012345678_10E+05 | -123456.78901234 | -123456.78901234 | -123456.78901234 | -123456.78901234 | *
;* | +1.23456789012345678_10E+06 | 1234567.89012345 | +1234567.89012345 | 1234567.89012345 | +1234567.89012345 | *
;* | -1.23456789012345678_10E+07 | -12345678.90123456 | -12345678.90123456 | -12345678.90123456 | -12345678.90123456 | *
;* | +1.23456789012345678_10E+08 | 123456789.0123456 | +123456789.0123456 | 123456789.0123456 | +123456789.0123456 | *
;* | -1.23456789012345678_10E+09 | -1234567890.123456 | -1234567890.123456 | -1234567890.123456 | -1234567890.123456 | *
;* | +1.23456789012345678_10E+10 | 12345678901.23456 | +12345678901.23456 | 12345678901.23456 | +12345678901.23456 | *
;* | -1.23456789012345678_10E+11 | -123456789012.3456 | -123456789012.3456 | -123456789012.3456 | -123456789012.3456 | *
;* | +1.23456789012345678_10E+12 | 1234567890123.456 | +1234567890123.456 | 1234567890123.456 | +1234567890123.456 | *
;* | -1.23456789012345678_10E+13 | -12345678901234.56 | -12345678901234.56 | -12345678901234.56 | -12345678901234.56 | *
;* | +1.23456789012345678_10E+14 | 123456789012345.6 | +123456789012345.6 | 123456789012345.6 | +123456789012345.6 | *

```





```

;*
;*=====
;* | Input Value          | ES(+18|+18)    | ES(+18|-18)    | ES(-18|+18)    | ES(-18|-18)    | *
;* +-----+-----+-----+-----+-----+ *
;* | +1234567890123456789_10E+10 | 123456789012345678 | 123456789012345678 | 123456789012345678 | 123456789012345678 | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* +-----+-----+-----+-----+-----+ *
;* | Input Value          | ES(+18|+18)    | ES(+18|-18)    | ES(-18|+18)    | ES(-18|-18)    | *
;* +-----+-----+-----+-----+-----+ *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* +-----+-----+-----+-----+-----+ *
;* | Input Value          | ES(+18|+18)    | ES(+18|-18)    | ES(-18|+18)    | ES(-18|-18)    | *
;* +-----+-----+-----+-----+-----+ *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* +-----+-----+-----+-----+-----+ *
;* | Input Value          | ES(+18|+18)    | ES(+18|-18)    | ES(-18|+18)    | ES(-18|-18)    | *
;* +-----+-----+-----+-----+-----+ *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* |                      |                  |                  |                  |                  | *
;* +-----+-----+-----+-----+-----+ *
;*
;*=====
;*
;*****

```

## 5.2 Input Output Functions

## 5.2.1 GETMAT GET MATH library constansts

The function FLREAD reads unicode character string tokens from a diskfile or memoryarea. The diskfile or memory must be opened with the OPNFIL function. One data-token consists from an optional name string and an following optional data string. The name and the data part are separated with an equal character. Each token is ended with an token end character. This is either a comma or a semicolon.

Die Function FLREAD liest unicode charakter String Tokens von einer Diskdatei oder Speicherbereich. Die Diskdatei oder der Speicherbereich muss vor mit der OPNFIL Funktion geöffnet worden sein. Ein Daten-Token besteht aus einem optionalen Namen und einem optionalen Datenteil, getrennt mit einen Gleichheitszeichen. Jeder Token wird entweder mit einem Komma oder Semikolon beendet.

```

*****
;* 12345&789012345678901234567890123456789012345678901234567890123456789012_& !!_901234567890123456789|1234567 *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !*-----*
;*                                     !! 12345&78901234567890123456789 *
;*                                     !! *
;*          00 | 01   02   03   04 | 05          !! *
;* 12345 ERROR2 = DL1(FLREAD,FILNUM,TOKCNT,PAGCNT,LINCNT,COLCNT, & !! 01 Fortran direct call *
;* &          NACTRL,NABSIZ,NABCNT,NAECNT,NASOFF,NASCNT, & !! 02 *
;* &          DACTRL,DABSIZ,DABCNT,DAECNT,DASOFF,DASCNT, & !! 03 <023 014 0025> *
;* &          DATLEN,P64RES,BASFND,CHRCNT,DASTRG,NASTRG) & !! 04 <007F:861E>hex *
;*          06   07   08 | 09   10   11          !! *
;*          12 | 13   14   15   16 ! 17          !! *
;*          18   19   20 | 21   22   23          !! *
;*                                     !! *
;* 12345 ERROR2 = DL1(FLREAD,000001,TOKCNT,PAGCNT,LINCNT,COLCNT, & !! 01 Fortran direct call *
;* &          1LAUSNEMCT,002560,000000,002560,NASOFF,NASCNT, & !! 02 *
;* &          1IAUSNPrBF,002560,000000,002560,DASOFF,DASCNT, & !! 03 <023 014 0025> *
;* &          DATLEN,P64RES,BASFND,CHRCNT,DASTRG,NASTRG) & !! 04 <007F:861E>hex *
;*                                     !! *
*****
;* 1234567890123456789012345678901234567890123456789012345678901234567890123456 // 123456789|1234567 *
;* // - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* // - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* // - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* //-----*
;*                                     // 12345678901234567 *
;*                                     // *
;* Lab12345: Error2 = DL1(FLREAD, FilNum,&TokCnt,&PagCnt,&LinCnt,&ColCnt, // 01 C/C++ direct *
;*          NaCtrl, NaBSiz, NaBCnt, NaECnt,&NaSOFF,&NaSCnt, // 02 *
;*          DaCtrl, DaBSiz, DaBCnt, DaECnt,&DaSOFF,&DaSCnt, // 03 *
;*          &DatLen,*P64Res,&BasFnd,&ChrCnt,*DaStrg,*NaStrg); // 04 *
;*          IV   IV   IV | IV   IR   IR // *
;*          IV | IV   IV   IV   IR ! IR // *
;*          IR   PR   IR | IR   CR   CR // *
;*                                     // *
;* Lab12345: Error2 = DL1(FLREAD, 000001,&TokCnt,&PagCnt,&LinCnt,&ColCnt, // 01 C/C++ direct *
;*          1LAUSNEMCT, 2560, 0000, 2560,&NaSOFF,&NaSCnt, // 02 /C## call *
;*          1IAUSNPrBF, 2560, 0000, 2560,&DaSOFF,&DaSCnt, // 03 *
;*          &DatLen,*P64Res,&BasFnd,&ChrCnt,*DaStrg,*NaStrg); // 04 *
;*                                     // *
*****
;* 12345&789012345678901234567890123456789012345678901234567890123456789012_& !!_901234567890123456789|1234567 *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !* - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - *
;* !*-----*
;*                                     !! 12345&78901234567890123456789 *
;*                                     !! *
;*          00 | 01   02   03   04 | 05          !! *
;* 12345 ERROR2 = FOR_FLREAD(FILNUM,TOKCNT,PAGCNT,LINCNT,COLCNT, & !! 01 Fortran direct call *
;* &          NACTRL,NABSIZ,NABCNT,NAECNT,NASOFF,NASCNT, & !! 02 *
;* &          DACTRL,DABSIZ,DABCNT,DAECNT,DASOFF,DASCNT, & !! 03 <023 014 0025> *
;* &          DATLEN,P64RES,BASFND,CHRCNT,DASTRG,NASTRG) & !! 04 <007F:861E>hex *
;*          06   07   08 | 09   10   11          !! *
;*          12 | 13   14   15   16 ! 17          !! *
;*          18   19   20 | 21   22   23          !! *
;*                                     !! *
;* 12345 ERROR2 = FOR_FLREAD(000001,TOKCNT,PAGCNT,LINCNT,COLCNT, & !! 01 Fortran direct call *
;* &          1LAUSNEMCT,002560,000000,002560,NASOFF,NASCNT, & !! 02 *
;* &          1IAUSNPrBF,002560,000000,002560,DASOFF,DASCNT, & !! 03 <023 014 0025> *
;* &          DATLEN,P64RES,BASFND,CHRCNT,DASTRG,NASTRG) & !! 04 <007F:861E>hex *
;*                                     !! *
*****

```





```

;* IsNext ! jmp      BJMP F1FLRea@01                ; 08 -> Proces Architecture 128 Version 512 (128)
;*
F1FLRea@01: mov      LVAL[lab+FFR_ExFILNUM], ecx      ; Store ECX => ShadowSpace InputArgument001 [LCX]
F1FLRea@02: mov      LVAL[lab+FFR_ExTOKCNT], edx      ; Store EDX => ShadowSpace InputArgument002 [LDX]
F1FLRea@08: mov      LVAL[lab+FFR_ExPAGCNT], ebx      ; Store EBX => ShadowSpace InputArgument003 [LOB]
F1FLRea@09: mov      LVAL[lab+FFR_ExLINCNT], ecx      ; Store ECX => ShadowSpace InputArgument004 [LO9]
;=====
;* LRea@10: call     SyGThMem                        ; Call: LAX := GetThread_Thr.Mai.Mem Memory Pnter
;*
;* LRea@11: lea     lbp,[lab+tmm_eml+0000]           ; Fetch LBP := ThrMaiMem_Err.Mat.Lib Memory Pnter
;* LRea@12: lea     lsi,[lab+tmm_iol+0000]           ; Fetch LSI := ThrMaiMem_Inp.Out.Lib Memory Pnter
;* LRea@13: lea     ldi,[lab+tmm_mlx+0000]           ; Fetch LDI := ThrMaiMem_Mat.Lib.Xxx Memory Pnter
;*
F1FLRea@10: lea     lbp,[lax+gmm_eml+0000]           ; Fetch LBP := GloMaiMem_Err.Mat.Lib Memory Pnter
mov      PNTR[lbp+eml_SFBotPtr], lbp                ; Store LBP => ExcMatLib_Stack.Frame.Bottom Pnter
;*
F1FLRea@16: lea     lsi,[lax+gmm_iol+0000]           ; Fetch LSI := GloMaiMem_Inp.Out.Lib Memory Pnter
mov      PNTR[lsi+iol_SFBotPtr], lsi                ; Store LSI => InpOutLib_Stack.Frame.Bottom Pnter
;*
;* LRea@17: lea     ldi,[lax+gmm_mlx+0000]           ; Fetch LDI := GloMaiMem_Mat.Lib.Xxx Memory Pnter
;* Unused ! mov    PNTR[ldi+mlx_SFBotPtr], ldi      ; Store LDI => MatLibXxx_Stack.Frame.Bottom Pnter
;=====
F1FLRea@21: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage <IAUSN????>
mov      LVAL[lbp+eml_MbContrl], lab                 ; Store LAX => ExcMatLib Informations <IAUSN????>
;*
F1FLRea@22: lea     lab,[ljp+dat_F1FLREAD]           ; Fetch LAX := ErrMsgBox WindowsTitle FOR_FLREAD(
mov      PNTR[lbp+eml_MbBoxTit], lab                 ; Store LAX => ExcMatLib Informations StrgPointer
;*
F1FLRea@23: lea     lab,[ljp+dat_MbHlgFil]           ; Fetch LAX := ErrMsgBox HighLanguage FileName
mov      PNTR[lbp+eml_MbHlgFil], lab                 ; Store LAX => ExcMatLib Informations StrgPointer
;*
F1FLRea@24: lea     lab,[ljp+dat_MbHlgFun]           ; Fetch LAX := ErrMsgBox HighLanguage Function
mov      PNTR[lbp+eml_MbHlgFun], lab                 ; Store LAX => ExcMatLib Informations StrgPointer
;*
F1FLRea@25: mov      eax, 0000000010220333           ; Fetch LAX := ErrMsgBox HighLanguage LineNumber
mov      LVAL[lbp+eml_MbHlgLin], lab                 ; Store LAX => ExcMatLib Informations IntegerValu
;*
F1FLRea@26: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage Null Parent
mov      LVAL[lbp+eml_MbParent], lab                 ; Store LAX => ExcMatLib Informations Wind Handle
;=====
F1FLRea@31: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage <IAUSN????>
mov      LVAL[lbp+eml_JqCONTRL], lab                 ; Store LAX => ExcMatLib Informations <IAUSN????>
;*
F1FLRea@32: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage <IAUSN????>
mov      LVAL[lbp+eml_JqCONNUM], lab                 ; Store LAX => ExcMatLib Informations <IAUSN????>
;*
F1FLRea@33: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage <IAUSN????>
mov      LVAL[lbp+eml_JqCMUTEX], lab                 ; Store LAX => ExcMatLib Informations <IAUSN????>
;*
F1FLRea@34: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage <IAUSN????>
mov      LVAL[lbp+eml_JqWCOHND], lab                 ; Store LAX => ExcMatLib Informations <IAUSN????>
;*
F1FLRea@35: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage <IAUSN????>
mov      LVAL[lbp+eml_JqRCOHND], lab                 ; Store LAX => ExcMatLib Informations <IAUSN????>
;*
F1FLRea@36: mov      eax, 0000000000000000           ; Fetch LAX := ErrMsgBox HighLanguage <IAUSN????>
mov      LVAL[lbp+eml_JqHEXFMT], lab                 ; Store LAX => ExcMatLib Informations <IAUSN????>
;*****
F1FLRead@9: mov      eax,'FFR@'                     ; fetch EAX := FortranFileReads <?> 000000
mov      edx, 00000BEH                             ; 00000-EAX-na-[NEBT:MSiz:LFun]-ebt-uI04B1Text-ML
call     EML1QPut                                    ; Write Out
;* Debugs ! jmp    LJMP F1FLReadGT                  ; Goto: End
;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```







