

FPUExc

COLLABORATORS

	<i>TITLE :</i> FPUExc		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 22, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	FPUExc	1
1.1	FPUExc	1
1.2	TMP:Modula-2/FPUExc.def	1

Chapter 1

FPUExc

1.1 FPUExc

Prozeduren

ClearExc	DisableExc	EnableExc
GetExcSet	GetRounding	SetExc
SetExcSet	SetRounding	

Konstanten

noExc	stdExc
-------	--------

Typ-Deklarationen

fpuExc	fpuExcSet	roundingMode
--------	-----------	--------------

1.2 TMP:Modula-2/FPUExc.def

```
DEFINITION MODULE FPUExc;  
(* $ NameChk:=FALSE *)
```

```
(*  
 * 18.7.90/bp  
 * Exception-Verwaltung für 68040/68881/68882.  
 * Wenn das Modul nur importiert wird, werden die FPU-  
 * Traps aus 'stdExc' erlaubt, d.h. die Überprüfungen der  
 * FPU sind vollständig.  
 * Falls keine FPU im System ist, passiert gar nichts,  
 * bzw. RETURN ist 0  
 * Die aufgelaufenen Exceptions im FPSR werden nur beim  
 * Start gelöscht, nicht von den Prozeduren!  
 *)
```

TYPE

```
fpuExc =(inex1,inex2,dz,unfl,ovfl,operr,snan,bsun);  
fpuExcSet =SET OF fpuExc ;
```

```
    roundingMode =
        (toNearest,toZero,toMinusInfty,toPlusInfty);

CONST
    stdExc= fpuExcSet {dz..bsun}; (* NICHT (!) inex1,inex2 *)
    noExc = fpuExcSet {};

(* default (nur importieren) ist: {dz..bsun} *)

PROCEDURE EnableExc;
    (* alle stdExc ON *)
PROCEDURE DisableExc;
    (* alles OFF, toNearest, round extended *)

PROCEDURE SetExc(e{0}: fpuExc );
    (* einzelne setzen/löschen *)
PROCEDURE ClearExc(e{0}: fpuExc );

PROCEDURE SetExcSet(s{0}: fpuExcSet );
    (* genau den Set setzen *)
PROCEDURE GetExcSet(): fpuExcSet ;
    (* aktuellen Zustand holen *)

PROCEDURE SetRounding(r{0}: roundingMode );
PROCEDURE GetRounding(): roundingMode ;

END FPUExc.
```
