WISC LIBRARY

INDEX

0.01		
.02		
.03.01	CON	Conversion-Deconversion
.04.01	ADA	Adaptation
.02	MOD	Modification
.03	MOV	Move
0.05.01	LOD	Load
.02	LAD	Load and Adapt
1.01.01	FAIR	Floating Arithmetic Interpretive Routine
2.01.01	SQR	Square Root
.02	EXS	Exponential (infinite series)
.03	SCR	Sine-Cosine Routine
.04	LNS	Logarithm (base e)
.05	ATS	Arctangent (infinite series iteration)
.06	ATP	Arctangent (polynomial approximation)
.07	CUR	Cube Root
.08	EXP	Exponential (polynomial approximation)
.09	LNP	Logarithm (either base) (polynomial approximation)
.10	GAP	Gamma Function
.02.01	CAC	Complex Arithmetic (Cartesian)
.03.01	ILT	Inverse Laplace Transform
3.01		
.02.01	RKS	Runge-Kutta Step
.02	IIT	Integral Interval Tester
.03		
.04		
.05.01	PRS	Polynomial Root Solver
3.05.02	QUF	Quadratic Formula
.06.01	SER	Simultaneous Equations Routine (algebraic)
4.01		
.02		
.03	LS3	least Squares--Cubic
5		
6.01.01	LAO	Logical And-Or
7.01.01	NIM	Nim
.02	STS	Saints
.02		
.03		
.04		
.05		
.06		
.07.01	SP4	Spangenberg--Problem 8-4
.08		
.09		
9.01.01	ITR	Intracomputational Test Routine
.02	VER	Verification
.03	RWC	Read-Write Check

WISC LIBRARY
CLASSIFICATION

0.00.00 UTILITY ROUTINES

These routines aid the programmer in preparing and assembling the program. Conversion, adaptation, coding, and assembly routines are among the types that may be found under this heading.

- 0.01.00 Program assembly
- 0.02.00 Program conversion
- 0.03.00 Data conversion
- 0.04.00 Relocation
- .05.00 Input
- .06.00 Output

1.00.00 INTERPRETIVE ROUTINES

Routines contained herein process a program of machine language orders or pseudo-orders in such a way that the orders of the program usually do not come under direct control of the computer. Tracing routines, complex number arithmetic, and certain other logical routines are among those under this heading.

- 1.01.00 Interpretive (numerical)
- 1.02.00 Interpretive (logical)
- 1.03.00 Tracing
- 1.04.00 Interpretive programming aids

2.00.00 MATHEMATICAL FUNCTION ROUTINES

Routines whose primary purpose is to calculate functions of general use in preparing programs are contained in this section. Elementary functions such as those associated with algebra (i.e., square root, sine, cosine, exponential, etc.) and higher functions such as those associated with higher mathematics (i.e., Bessel functions, etc.) may be classified under this heading.

- 2.01.00 Elementary (real number)
- 2.02.00 Elementary (complex number)
- 2.03.00 Higher mathematical functions

3.00.00 ROUTINES FOR MATHEMATICAL SOLUTIONS

The scope of the routines in this classification include equation and equation system solutions, matrix methods, integration, differentiation and differential equation solutions among others.

- 3.01.00 Integration
- 3.02.00 Differential equation solutions (ordinary)
- 3.03.00 Differential equation solutions (partial)
- 3.04.00 Differentiation
- 3.05.00 Equations of one variable
- 3.06.00 Linear sets of equations
- 3.07.00 Non-linear sets of equations
- 3.08.00 Matrices

4.00.00 INTERPOLATION, CURVE FITTING, AND APPROXIMATION ROUTINES

Interpolation and curve fitting are self explanatory. The approximations refer to routines which approximate complex functions by simpler functions.

4.01.00 Interpolation
4.02.00 Subtabulation
4.03.00 Polynomial curve fitting
4.04.00 Higher function curve fitting
4.05.00 Polynomial approximations
4.06.00 Higher function approximations

5.00.00 STATISTICAL ROUTINES

The routines contained herein are concerned with averaging, collating, and smoothing data or in other ways aiding the analysis of empirical data.

6.00.00 MISCELLANEOUS PROCESSES

This section acts as catch-all for routines not contained in the previous sections. This might include such programs as those dealing with series inversion and summation, binomial expansions, routines to find prime numbers, and certain logical routines.

6.01.00 Functions easily derived from elementary mathematical functions (2.01.00)

7.00.00 APPLIED PROGRAMS

It seems reasonable to provide a section in which persons using the WISC may store programs of those problems which occur frequently in certain fields. For example electrical engineers may want to store for general use programs for electrical bridge solutions or filter design. In this section may also be stored demonstrative routines and programs.

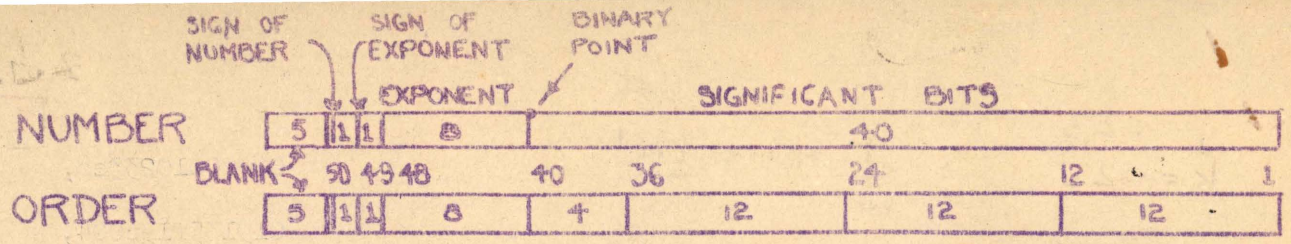
7.01.00 Demonstrative
7.02.00 Mathematics
7.03.00 Physics
7.04.00 Mechanics
7.05.00 Chemical engineering
7.06.00 Civil engineering
7.07.00 Electrical engineering
7.08.00 Mechanical engineering
7.09.00 Others

8.00.00 (No classification)

9.00.00 TESTS AND DIAGNOSTIC ROUTINES

Self-explanatory.

9.01.00 To be used by the programmer
9.02.00 For maintenance
9.03.00 For trouble shooting.



CODE	SYMBOL	OPERATION		ORDER TYPE	A	B	C
0	I	Input	00	0000	00 Address for first word.	00 Address for last word	00 Address of next order
1	E	Extract	00 00 x(6)	0001	00 Address of operand A	z(6) y(6) z(6)	00 Address for result
2	M	Multiply	0n	0010	s0 "	s0 Address of operand B	r0 "
3	D	Divide	0n	0011	s0 "	s0 "	r0 "
4	N	No Operation		0100			
5	TU	Unconditional Transfer	0t	0101	00 Address of first word	Address of last word	00 Address of next order
6	H	Halt	0h	0110	s0 Address of operand A	Address of operand B	00 "
7	O	Output	00	0111	s0 "	00 "	00 "
8	A	Add	0n	1000	s0 "	s0 "	r0 Address for result
9	AA	Add Absolute	0n	1001	s0 "	s0 "	r0 "
a	S	Subtract	0n	1010	s0 "	s0 "	r0 "
b	SA	Subtract Absolute	0n	1011	s0 "	s0 "	r0 "
c	TZ	Transfer on zero	00	1100	s0 "	s0 "	00 Next order if A-B=0
d	TZA	Transfer on zero absolute	00	1101	s0 "	s0 "	00 Next order if |A|-|B|=0
e	TN	Transfer on negative	00	1110	s0 "	s0 "	00 Next order if A-B<0
f	TNA	Transfer on negative abs.	00	1111	s0 "	s0 "	00 Next order if |A|-|B|<0

h=0 Unconditional Halt
=1 Breakpoint Halt (halt if switch set)

x = first bit to Extract
y = place to insert first bit.
z = number of bits to Extract.

n=0 Normalize result
=1 Do not shift result

r=0 Normal operation
=1 Do not deliver result to storage.

SIGN: 0 = plus
1 = minus

s=0 Normal operation.
=1 Use preceding result for this operand.

t=0 Unconditional Transfer
=1 Breakpoint Transfer (transfer if switch set)

STRUCTURE OF NUMBERS AND ORDERS

WISC
DRUM STORAGE ASSIGNMENTS

~~5/29/59~~ 1/5/60 2/19/60

DEC	HEX	USE	NO.
0	000	USED FOR INPUT	1
1 to 851	001 to 353	AVAILABLE FOR USE	851
852 to 862	354 to 35e	OPSTO	11
863	35f	LINKAGE OPSTO	1
864 to 951	360 to 3af	CONVERSION ROUTINES (0 for CON — 2 for DECON) [100 8 L+2 3ff 35f / 5 360 / 0 0 IN,[IN,[A]	~~88~~ ~~93~~ 80
952 to 976	3b0 to 3d0	MODIFICATION/ ADAPTATION ROUTINE [100 8 L+2 3ff 35f / 5 3b1 / 0 J IW,[k β]	~~33~~ ~~38~~ 34 1
977 to 999	3d1 to 3e7	TEST ROUTINE (ITR) [~~000 1 L+1 01c 35f~~ / ~~100 8 L+1 3ff 3d1 / 5 35f / β 3d3~~	~~21~~

DEAD STORAGE (5 TRACKS)

Also 3d1 3d2

1000	3e8	HALT ORDER	000	6	000	000	3e8		
1001	3e9	~~A~~	~~000~~	~~0~~	~~000~~	~~000~~	~~000~~	2^{16}	
1002	3ea	~~2^{00}~~	~~000~~	~~0~~	~~000~~	~~000~~	~~001~~	2	
1003	3eb	$2^{-12} = 10^{A}$	000	0	010	000	000		
1004	3ec	4^{C}	000	0	000	000	004		
1005	3ed	$\sqrt{2} = 1.414\,21356$	001	b	504	f33	3fa		
1006	3ee	.1111...	000	f	fff	fff	fff		
1007	3ef	4^{X}	004	0	000	000	000		
1008	3f0	1×2^{4}	005	8	000	000	000		
1009	3f1	$e = 2.718281828$	002	a	df8	545	8a3		
1010	3f2	$\pi = 3.141592654$	002	c	9a8	daa	222		
1011	3f3	2^{C}	000	0	000	000	002		
1012	3f4	10^{-9}	11d	8	970	5f4	137		
1013	3f5	10^{10}	022	9	502	f90	000		
1014	3f6	10	004	a	000	000	000		
1015	3f7	2^{8}	000	0	100	000	000		
1016	3f8	$2^{-1} = .1 = 1/2$	000	8	000	000	000		
1017	3f9	$2^{-2} = .01$	000	4	000	000	000		
1018	3fa	$2^{-4} = .0001 \equiv$ E order $= 1^{D}$	000	1	000	000	000		
1019	3fb	1^{A}	000	0	001	000	000		
1020	3fc	1^{B}	000	0	000	001	000		
1021	3fd	1^{C}	000	0	000	000	001		
1022	3fe	1	001	e	000	000	000		
1023	3ff	0	000	0	000	000	000		

PERSTO 24

0001000000000,	0007000000000,	1061502325817,	0011800000000,
0001000000000,	0007000000000,	1061502325817,	0011600000000,
0001000000000,	0007000000000,	1061502325817,	0011400000000,
0001000000000,	0007000000000,	1061502325817,	0011200000000,
0001000000000,	007000000000,	1061502325817,	0011000000000,
0001000000000	0007000000000,	1061502302137,	0008000000000, 8
0001000000000,	0007000000000,	01375596655l8,	0006000000000,
0001000000000,	0007000000000,	0354468985248,	0004000000000,
0001000000000,	0007000000000,	0591200593700,	0002000000000,
1018750000000,	0007000000000,	1075943074500,	0011800000000,
1018750000000,	0007000000000,	1075943074500,	0011600000000,
1018750000000,	0007000000000,	1075943074500,	0011400000000,
1018750000000,	0007000000000,	1075943074500,	0011200000000,
1018750000000,	0007000000000,	1075943074500,	0011000000000,
1018750000000,	0007000000000,	1075943019803,	0008000000000, 8
1018750000000,	0007000000000,	01313554191l5l,	0006000000000,
1018750000000,	0007000000000,	0344767297639,	0004000000000,
1018750000000,	0007000000000,	0581006781287,	0002000000000,
1017500000000,	0007000000000,	1072033044020,	0011800000000,
1017500000000,	0007000000000,	1072033044020,	0011600000000,
1017500000000,	0007000000000,	1072033044020,	0011400000000,
1017500000000,	0007000000000,	107203044020,	001300000000,
1017500000000,	0007000000000,	1072033044020,	0011000000000,
1017500000000,	0007000000000,	1072033033954,	0008000000000, 8
1017500000000,	0007000000000,	0121854498933,	0006000000000,
1017500000000,	0007000000000,	0333566305968,	0004000000000,
1017500000000,	0007000000000,	0565664612305,	0002000000000,
1016250000000,	0007000000000,	1085704460376,	0011800000000,
1016250000000	0007000000000,	1085704460376,	0011600000000,
1016250000000,	0007000000000,	1085704460376,	0011400000000,
1016250000000,	0007000000000,	1085704460377,	0011200000000,
1016250000000,	0007000000000,	1085704460376,	0011000000000,
1016250000000	0007000000000,	1085704446796,	0008000000000, 8
1016250000000,	0007000000000,	0111755216358,	0006000000000,
1016250000000,	0007000000000,	0321645527681,	0004000000000,
1016250000000,	0007000000000,	0551852254097,	0002000000000,
1015000000000,	0007000000000,	1081201586732,	0011800000000,
1015000000000,	0007000000000,	1081201586732,	0011600000000,
1015000000000,	0007000000000,	1081201586732,	0011400000000,
1015000000000,	0007000000000,	1081201586732,	0011200000000,
1015000000000,	0007000000000,	1081201586732,	0011000000000,
1015000000000,	0007000000000,	1081201585563,	0008000000000, 8
1015000000000,	0007000000000,	0099749071990,	0006000000000,
1015000000000,	0007000000000,	0303776928327,	004000000000,
1015000000000,	0007000000000,	0532767288590,	0002000000000,
1013750000000,	0007000000000,	1091609419952,	0011800000000,
1013750000000,	0007000000000,	1091609419952,	0011600000000,
1013750000000,	0007000000000,	1091609419952,	0011400000000,
1013750000000,	0007000000000,	1091609419952,	0011200000000,
1013750000000,	0007000000000,	1091609419952,	0011000000000,
1013750000000,	0007000000000,	1091609419458,	0008000000000, 8
1013750000000,	0007000000000,	0082334682587,	0006000000000,
1013750000000,	0007000000000,	0282881567549,	0004000000000,
1013750000000,	0007000000000,	0511203547512,	0002000000000,
1012500000000,	0007000000000,	1119442592136,	0011800000000,
101700000000,	0007000000000,	1119442592136,	0011600000000,
1012500000000,	0007000000000,	1119442592136,	0011400000000,
1012500000000,	0007000000000,	1119442592136,	0011200000000,
1012500000000,	0007000000000,	1119442592136,	0011000000000,
1012500000000,	0007000000000,	111944259563,	0008000000000, (8)
1012500000000,	0007000000000,	0061206471772,	0006000000000,

BADGER BY J. F. MCNALL

000000012b000,	0008000000800,	000a000000800,
0004000000000,	0008000000800,	000a000000800,
0004000000000,	0008000000800,	0009000000800,
1005000000000,	0008000000800,	0009000000800,
1005000000000,	0004000000000,	000a000000800,
0008000000800,	0004000000000,	000a000000800,
000f000000000,	0008000000800,	000b000000800,
0008000000800,	0008000000800,	000b000000800,
000f000000000,	0008000000800,	000b000000800,
0008000000800,	0008000000800,	000b000000800,
000f000000000,	0004000000000,	000b000000800,
0008000000800,	0004000000000,	000b000000800,
000f000000000,	1005000000000,	0008000000800,
0008000000800,	1005000000000,	0008000000800,
0008000000800,	0008000000800,	888f000000000,
000b000000800,	000f000000000,	0008000000800,
000b000000800,	0008000000800,	0009000000800,
000b000000800,	000f000000000,	0009000000800,
000f000000000,	0008000000800,	0008000000800,
000b000000800,	000f000000000,	0008000000800,
000f000000000,	0008000000800,	0008000000800,
000c3ff3ee000,	000f000000000,	0008000000800,
000f000000000,	0008000000800,	0004000000000,
000c3ff3ee000,	0008000000800,	0004000000000,
000c3ff3ee000,	000b000000800,	0008000000800,
000c3ff3ee000,	000b000000800,	0008000000800,
000f000000000,	000b000000800,	0008000000800,
000c3ff3ee000,	000f000000000,	0008000000800,
000f000000000,	000b000000800,	000b000000800,
0009000000800,	000f000000000,	000b000000800,
0009000000800,	000c3ff3ee000,	000f000000000,
0009000000800,	000f000000000,	000b000000800,
0009000000800,	000c3ff3ee000,	000c3ff3ee000,
000b000000800,	000c3ff3ee000,	000f000000000,
000b000000800,	000c3ff3ee000,	000c3ff3ee000,
000f000000000,	000f000000000,	000c3ff3ee000,
000b000000800,	000c3ff3ee000,	000c3ff3ee000,
000a000000800,	000f000000000,	000f000000000,
000a000000800,	0009000000800,	000c3ff3ee000,
000a000000800,	0009000000800,	000c3ff3ee000,
000f000000000,	0009000000800,	0009000000800,
000a000000800,	0009000000800,	0009000000800,
000f000000000,	000b000000800,	0009000000800,
000a000000800,	000b000000800,	0009000000800,
000f000000000,	000f000000000,	000b000000800,
000a000000800,	000b000000800,	000b000000800,
000a000000800,	000a000000800,	000f000000000,
000a000000800,	000a000000800,	000b000000800,
000a000000800,	000a000000800,	000a000000800,
0009000000800,	000f000000000,	000a000000800,
0009000000800,	000a000000800,	000a000000800,
000f000000000,	000f000000000,	000f000000000,
000a000000800,	000a000000800,	000a000000800,
0009000000800,	000f000000000,	000f000000000,
0009000000800,	000a000000800,	000a000000800,
	000a000000800,	000f000000000,
		000a000000800,
		000a000000800,
		000a000000800,
		000a000000f00,

```
0009000000800,      000f000000000,      000f,
0009000000800,      000a000000800,      000f,
0009000000800,      000f000000000,      000f,
000a000000800,      000a000000800,      000f,
0009000000800,      000f000000000,      000f,
0009000000800,      000a000000800,      000f,
0008000000800,      000a000000800,      000f,
0008000000800,      000a000000800,      000f,
0008000000800,      000a000000800,      000f,
0008000000800,      0009000000800,      000f,
0004000000000,      0009000000800,      000f,
0004000000000,      000a000000800,      000f,
0008000000800,      000a000000800,      000f,
0008000000800,      000b000000800,      000f,
0008000000800,      000b000000800,      000f,
0008000000800,      000b000000800,      000f,
0004000000000,      000b000000800,      0005000000000,
0004000000000,      000f,               000f,
1005000000000,      000f,               000f,
1005000000000,      000f,                000f,
0008000000800,      000f,
000f000000000,      000f,
0008000000800,      000f,
000f000000000,      000f,
0008000000800,      000f,
000f000000000,      000f,
0008000000800,      000f,
000f000000000,      000f,
0008000000800,      00070000000ff,
0008000000800,      00070000001001,
000b000000800,      00070000001011,
000b000000800,      000f,
000b000000800,      000f,
000f000000000,      000f,
000b000000800,      000f,
000f000000000,      000f,
000c3ff3ee000,      000f,
000f000000000,      000f,
000c3ff3ee000,      000f,
000c3ff3ee000,      000f,
000c3ff3ee000,      000f,
000f000000000,      000f,
000c3ff3ee000,      000f,
000f000000000,      000f,
0009000000800,      000f,
0009000000800,      000f,
0009000000800,      000f,
0009000000800,      000f,
000b000000800,      000f,
000b000000800,      000f,
000f000000000,      000f,
000b000000800,      000f,
000a000000800,      000f,
000a000000800,      000f,
000a000000800,      000f,
```

HEXADECIMAL to DECIMAL CONVERSION TABLE

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	00	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	01	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	02	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
	03	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
2	04	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
	05	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
3	06	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
	07	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
4	08	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
	09	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
5	0a	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
	0b	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
6	0c	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
	0d	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
7	0e	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
	0f	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255
8	10	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271
	11	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287
9	12	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303
	13	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319
10	14	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335
	15	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351
11	16	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367
	17	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383
12	18	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399
	19	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415
13	1a	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431
	1b	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447
14	1c	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463
	1d	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479
15	1e	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495
	1f	496	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511

000-1ff

HEXADECIMAL to DECIMAL CONVERSION TABLE

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
20	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527
21	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543
22	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559
23	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575
24	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591
25	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607
26	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623
27	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639
28	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654	655
29	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671
2a	672	673	674	675	676	677	678	679	680	681	682	683	684	685	686	687
2b	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702	703
2c	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719
2d	720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735
2e	736	737	738	739	740	741	742	743	744	745	746	747	748	749	750	751
2f	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767
30	768	769	770	771	772	773	774	775	776	777	778	779	780	781	782	783
31	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799
32	800	801	802	803	804	805	806	807	808	809	810	811	812	813	814	815
33	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831
34	832	833	834	835	836	837	838	839	840	841	842	843	844	845	846	847
35	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863
36	864	865	866	867	868	869	870	871	872	873	874	875	876	877	878	879
37	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895
38	896	897	898	899	900	901	902	903	904	905	906	907	908	909	910	911
39	912	913	914	915	916	917	918	919	920	921	922	923	924	925	926	927
3a	928	929	930	931	932	933	934	935	936	937	938	939	940	941	942	943
3b	944	945	946	947	948	949	950	951	952	953	954	955	956	957	958	959
3c	960	961	962	963	964	965	966	967	968	969	970	971	972	973	974	975
3d	976	977	978	979	980	981	982	983	984	985	986	987	988	989	990	991
3e	992	993	994	995	996	997	998	999	1000	1001	1002	1003	1004	1005	1006	1007
3f	1008	1009	1010	1011	1012	1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023

WISC. FLOATING POINT OPERATION

Number is

$$sp \times 2^{rq}$$

where s = sign of no.

r = sign of exponent.

Types of operation

Floating point	d49	if $p \neq 0$,	$s = s_R$	$r = r_R$	$q = q_R$
		if $p = 0$,	$s = 0$	$r = 0$	$q = 0$
Fixed point	d49	if $p \neq 0$,	$s = s_R$	$r = r_A$	$q = q_A$
		if $p = 0$,	$s = s_A$	$r = r_A$	$q = q_A$

↳ $s_R = s_A$ for A, S, C. $p = 0$

$s_R = s_A \times s_B$ for Mu, D.

Notation examples

$s = s_A$ means that result will have the same sign as the A operand

$r = r_R$ means result will have the normalized result exponent sign.