





210	X	X	X	X	X	X	X	JUMP10
...	X	X	X	X	X	X	X	....
217	X	X	X	X	X	X	X	JUMP17
220	X	X	X	X	X	X	X	FJUMP00
...	X	X	X	X	X	X	X	...
227	X	X	X	X	X	X	X	FJUMP07
230	X	X	X	X	X	X	X	FJUMP10←
...	X	X	X	X	X	X	X	...
237	X	X	X	X	X	X	X	FJUMP17←
240	X	X	X	X	X	X	X	TJUMP00
...	X	X	X	X	X	X	X	...
247	X	X	X	X	X	X	X	TJUMP07
250	X	X	X	X	X	X	X	TJUMP10
...	X	X	X	X	X	X	X	...
257	X	X	X	X	X	X	X	TJUMP17
260	X	X	X	X	X	X	X	JUMPX
261	X	X	X	X	X	X	X	JUMPMX
262	X	X	X	X	X	X	X	FJUMPMX
263	X	X	X	X	X	X	X	TJUMPMX
264	X	X	X	X	X	X	X	NFJUMPMX
265	X	X	X	X	X	X	X	NTJUMPMX
266	X	-	-	-	-	-	-	ARRAYINDEX1
267	X	-	-	-	-	-	-	ARRAYINDEX2
270	X	X	X	X	X	X	X	PVAR0←
...	X	X	X	X	X	X	X	...
276	X	X	X	X	X	X	X	PVAR6←
277	X	X	X	X	X	X	X	POP
300	X	X	-	-	-	X	X	POP.N
301	X	X	-	-	-	X	X	ATOMCELL.N
302	X	X	X	X	X	X	X	GETBASEBYTE
303	-	-	-	-	-	-	-	{unused}
304	X	X	X	X	X	X	X	BLT
305	X	-	-	-	-	-	-	PIXELBLT
306	-	-	-	-	-	-	-	{unused}
307	X	X	X	X	X	X	X	PUTBASEBYTE
310	X	X	X	X	X	X	X	GETBASE.N
311	X	X	X	X	X	X	X	GETBASEPTR.N
312	X	X	X	X	X	X	X	GETBITS.N.FD
313	-	-	-	-	-	-	-	<b>{used for GETBASEFIXP}</b>
314	-	-	-	-	-	-	-	<b>{used for PUTBASEFIXP}</b>
315	X	X	X	X	X	X	X	PUTBASE.N
316	X	X	X	X	X	X	X	PUTBASEPTR.N
317	X	X	X	X	X	X	X	PUTBITS.N.FD
320	X	X	X	X	X	X	X	ADDBASE
321	X	X	X	X	X	X	X	VAG2
322	X	X	X	X	X	X	X	HILOC
323	X	X	X	X	X	X	X	LOLOC
324	X	X	X	X	X	X	X	PLUS2{see notes}
325	X	X	X	X	X	X	X	DIFFERENCE{see notes}
326	X	X	X	X	X	X	X	TIMES2{see notes}
327	X	X	X	X	X	X	X	QUOTIENT{see notes}
330	X	X	X	X	X	X	X	IPLUS2{see notes}
331	X	X	X	X	X	X	X	IDIFFERENCE{see notes}
332	X	X	X	X	X	X	X	ITIMES2{see notes}
333	X	X	X	X	X	X	X	IQUOTIENT{see notes}
334	X	X	X	X	X	X	X	IREMAINDER{see notes}
335	-	-	-	-	-	X	X	{unused} {IPLUS.N}
336	-	-	-	-	-	X	X	{unused} {IDIFFERENCE.N}
337	-	-	-	-	-	-	-	{unused}
340	X	X	X	X	X	X	X	LLSH1{see notes}
341	X	X	X	X	X	X	X	LLSH8{see notes}
342	X	X	X	X	X	X	X	LRSH1{see notes}
343	X	X	X	X	X	X	X	LRSH8{see notes}
344	X	X	X	X	X	X	X	LOGOR2{see notes}
345	X	X	X	X	X	X	X	LOGAND2{see notes}
346	X	X	X	X	X	X	X	LOGXOR2{see notes}
347	-	-	-	-	-	-	-	{unused} {new ALSH}

350	X	-	X	X	X	-	X	FPLUS2
351	X	-	X	X	X	-	X	FDIFFERENCE
352	X	-	X	X	X	-	X	FTIMES2
353	X	-	X	X	X	-	X	FQUOTIENT
354	X	-	-	-	-	-	-	UBFLOAT2 {UFADD, UFSUB, UFISUB, UFMULT, UFDIV, UGREAT, UMAX, UMIN, UREM}
355	X	-	-	-	X	-	-	UBFLOAT1 {UTOB, BTOU, UABS, UNEG, UFIX}
356	X	-	-	-	-	-	-	ARRAYREAD{GENERAL,UNBOXED}
357	X	-	-	-	-	-	-	ARRAYWRITE{GENERAL,UNBOXED}
360	X	X	X	X	X	X	X	EQ
361	X	X	X	X	X	X	X	IGREATERP
362	X	-	X	X	X	-	X	FGREATERP
363	X	X	X	X	X	X	X	GREATERP
364	X	X	?	?	X	-	X	EQUAL
365	X	-	X	X	X	-	X	MAKENUMBER
366	X	-	X	X	X	-	-	BOXIPLUS
367	X	-	X	X	X	-	-	BOXIDIFFERENCE
370	-	-	-	-	-	-	-	MISC5
371	X	-	-	-	-	-	-	FFTSTEP
372	X	-	-	-	-	-	-	MISC3 {Floating Point Array ops: EXP, MAG, FLOAT, COMPLEX, BLKMAX, BLKMIN, BLKABSMAX, BLKABSMIN, FLOATTOBYTE}
373	X	-	-	-	-	-	-	MISC4 {Floating Point Array ops: TIMES, PERM, PLUS, DIFFERENCE, MAGIC, BITMAPBIT}
374	-	-	?	-	X	-	-	{reserved for DOLPHIN}
375	X	X	X	X	X	X	X	SWAP
376	X	X	X	X	X	X	X	NOP
377	-	-	-	-	-	-	-	{unused}

## notes:

## 4K microcode:

PLUS2, DIFFERENCE, TIMES2, QUOTIENT will ufn if args not INTEGERS

IPLUS2, IDIFFERENCE will accept FIXP's as arguments, but will ufn if result is not a smallp or smallneg

ITIMES2, IQOTIENT, IREMAINDER will ufn if both args are not smallp

## 12K microcode:

PLUS2, DIFFERENCE, TIMES2, QUOTIENT will try floating point if args not INTEGERS

IPLUS2, IDIFFERENCE will accept FIXP's as arguments, and box the result if it is not a smallp or smallneg

ITIMES2, IQOTIENT, IREMAINDER will ufn if both args are not smallp

*—End of message—*