1 byte opcodes

Rev 2, 2021-12-14

	ALU r, r/m	0	0	op		0	d	w	mod	reg	r/m	
	ALU accum, imm	0	0	op		1	0	w				+imm
	PUSH/POP segment	0	0	0	seg	1	1	op				seg:op ≠ 01:1
286+	two byte OP codes	0	0	0	0	1	1	1	1			+instruction
	segment prefixes	0	0	1	seg	1	1	0				
	BCD adjustments	0	0	1	op	1	1	1				
	INC/DEC rw	0	1	0	0	op	reg					
	PUSH/POP rw	0	1	0	1	op	reg					
186+	PUSHA/POPA	0	1	1	0	0	0	0	op			
186+/286+	BOUND/ARPL	0	1	1	0	0	0	1	op	mod	reg	r/m
386+	FS/GS/D32/A32 prefixes	0	1	1	0	0	1	op				
186+	PUSH imm	0	1	1	0	1	0	s	0			+imm
186+	IMUL r, r/m, imm	0	1	1	0	1	0	s	1	mod	reg	r/m +imm
186+	string port IO	0	1	1	0	1	1	d	w			
	conditional jumps (short)	0	1	1	1	cc		n				+rel8
	ALU r/m, imm	1	0	0	0	0	0	s	w	mod	op	r/m
	TEST/XCHG r/m, r	1	0	0	0	0	1	op	w	mod	reg	r/m
	MOV r, r/m	1	0	0	0	1	0	d	w	mod	reg	r/m
	MOV seg, r/m	1	0	0	0	1	1	d	0	mod	seg	r/m
	LEA r, m	1	0	0	0	1	1	0	1	mod	reg	r/m mod ≠ 11
	POP r/m	1	0	0	0	1	1	1	1	mod	0 0 0	r/m
	XCHG accum, rw	1	0	0	1	0	reg					
	CBW, CWD	1	0	0	1	1	0	0	op			
	CALL FAR seg:off	1	0	0	1	1	0	1	0			+seg:off
	WAIT	1	0	0	1	1	0	1	1			
	flag register access	1	0	0	1	1	1	op				
	MOV accum, off	1	0	1	0	0	0	d	w			+off
	TEST accum, imm	1	0	1	0	1	0	0	w			+imm
	string operations	1	0	1	0	op		w				op ≠ 00?, op ≠ 100
	MOV r, imm	1	0	1	1	w	reg					+imm
186+	SHIFT r/m, imm	1	1	0	0	0	0	0	w	mod	op	r/m
	RET	1	1	0	0	f	0	1	i			if i = 1 then +imm16
	LES/LDS	1	1	0	0	0	1	0	op	mod	reg	r/m mod ≠ 11
	MOV r/m, imm	1	1	0	0	0	1	1	w	mod	reg	r/m +imm

ALU op

ADD	0	0	0
OR	0	0	1
ADC	0	1	0
SBB	0	1	1
AND	1	0	0
SUB	1	0	1
XOR	1	1	0
CMP	1	1	1

SHIFT op

ROL	0	0	0
ROR	0	0	1
RCL	0	1	0
RCR	0	1	1
SHL	1		0
SHR	1	0	1
SAR	1	1	1

single operand op

NOT	0	1	0
NEG	0	1	1
MUL	1	0	0
IMUL	1	0	1
DIV	1	1	0
IDIV	1	1	1

bit test op

BT	0	0
BTS	0	1
BTR	1	0
BTC	1	1

Tabelle1

186+	ENTER	1	1	0	0	1	0	0	0		+imm16+imm8
186+	LEAVE	1	1	0	0	1	0	0	1		
	interrupt handling	1	1	0	0	1	1	op			if op = 01 then +imm8
	SHIFT r/m, 1/CL	1	1	0	1	0	0	cl	w	mod op r/m	
	AAM/AAD	1	1	0	1	0	1	0	op		+imm8
	SALC/XLAT	1	1	0	1	0	1	1	op		
	coprocessor escape	1	1	0	1	1				mod r/m	
	loop instructions	1	1	1	0	0	0	op			+imm8
	port IO	1	1	1	0	i	1	d	w		if i = 1 then +imm8
	calls and jumps	1	1	1	0	1	0	op			+...
	F0 group prefixes	1	1	1	1	0	0	op			
	HLT/CMC	1	1	1	1	0	1	0	op		
	TEST r/m, imm	1	1	1	1	0	1	1	w	mod 0 0 r/m	+imm
	ALU r/m (single operand)	1	1	1	1	0	1	1	w	mod op r/m	op ≠ 00?
	set/clear flags	1	1	1	1	1	op		c		op ≠ 11
	single operand misc	1	1	1	1	1	1	1	w	mod op r/m	

2 byte opcodes

286+	SYSTEM r/m	0F	0	0	0	0	0	0	0	op	mod op r/m	
286+	LAR/LSL r, r/m	0F	0	0	0	0	0	0	1	op	mod reg r/m	
286 only	SAVEALL/LOADALL	0F	0	0	0	0	0	1	0	op		
286+	CLTS	0F	0	0	0	0	0	1	1	0		
386 only	LOADALL	0F	0	0	0	0	0	1	1	1		
486+	INVD/WBINVD	0F	0	0	0	0	1	0	0	op		
386+	MOV cr/dr, r/m	0F	0	0	1	0	0	0	d	op	1 1 cr/dr r/m	
386+	MOV tr, r/m	0F	0	0	1	0	0	1	d	0	1 1 tr r/m	
386+	conditional jumps (near)	0F	1	0	0	0	cc			n		+rel
386+	conditional set r/m	0F	1	0	0	1	cc			n	mod 0 0 0 r/m	
386+	PUSH/POP FS/GS	0F	1	0	1	0	sg	0	0	op		
486+/386+	CPUID/RSM	0F	1	0	1	0	op	0	1	0		
386+	bit test r/m, r	0F	1	0	1	op		0	1	1	mod reg r/m	
386+	SHLD/SHRD r, r/m, imm/CL	0F	1	0	1	0	op	1	0	cl	mod reg r/m	if cl = 0 then +imm8
386+	IMUL r, r/m	0F	1	0	1	0	1	1	1	1	mod reg r/m	
486+	CMPXCHG r, r/m	0F	1	0	1	1	0	0	0	w	mod reg r/m	
386+	LSS r, r/m	0F	1	0	1	1	0	0	1	0	mod reg r/m	mod ≠ 11

condition codes

O	0	0	0
B/C	0	0	1
E/Z	0	1	0
BE	0	1	1
S	1	0	0
P/PE	1	0	1
L	1	1	0
LE	1	1	1

SYSTEM op

SLDT	0	0	0	0
STR	0	0	0	1
LLDT	0	0	1	0
LTR	0	0	1	1
VERR	0	1	0	0
VERW	0	1	0	1
SGDT	1	0	0	0
SIDT	1	0	0	1
LGDT	1	0	1	0
LIDT	1	0	1	1
SMSW	1	1	0	0
LMSW	1	1	1	0
INVLPG	1	1	1	1

FP memory format

f32	0	0
i32	0	1
f64	1	0
i16	1	1

Tabelle1

386+	LFS/LGS r, r/m	0F	1	0	1	1	0	1	0	op	mod	reg	r/m
386+	BSF/BSR r, r/m	0F	1	0	1	1	1	1	0	op	mod	reg	r/m
386+	MOVZX/MOVSX r, r/m	0F	1	0	1	1	op	1	1	w	mod	reg	r/m
386+	bit test r/m, imm	0F	1	0	1	1	1	0	1	0	mod	1 op	r/m
486+	XADD r/m, r	0F	1	1	0	0	0	0	0	w	mod	reg	r/m
486+	BSWAP r	0F	1	1	0	0	1	reg					

mod ≠ 11

FP ALU op

FADD	0	0	0
FMUL	0	0	1
FCOM	0	1	P
FSUB	1	0	R
FDIV	1	1	R

x87 instructions

	FP ALU ST(0), mem	1	1	0	1	1	MF	0	mod	op	r/m	mod ≠ 11				
	FLD mem	1	1	0	1	1	MF	1	mod	0 0 0	r/m	mod ≠ 11				
	FST(P) mem	1	1	0	1	1	MF	1	mod	0 1 P	r/m	mod ≠ 11				
	FP misc mem	1	1	0	1	1	op1	1	mod	1 op2	r/m	mod ≠ 11				
	FP ALU ST(0), ST(i)	1	1	0	1	1	d	P	0	1 1	op	ST(i)	op ≠ 01?, not DA E9, not DE D9			
	FCOM(P) ST(0), ST(i)	1	1	0	1	1	0	0	0	1 1 0 1 P	ST(i)					
387+	FUCOMPP ST(0), ST(1)	1	1	0	1	1	1	0	0	1 1 1 0 1	0 0 1					
	FCOMPP ST(0), ST(i)	1	1	0	1	1	1	1	0	1 1 0 1 1	0 0 1					
	FLD ST(i)	1	1	0	1	1	0	0	1	1 1 0 0 0	ST(i)					
	FXCH ST(0), ST(i)	1	1	0	1	1	0	0	1	1 1 0 0 1	ST(i)					
	FNOP	1	1	0	1	1	0	0	1	1 1 0 1 0	0 0 0					
	FCHS/FABS	1	1	0	1	1	0	0	1	1 1 1 0 0	0 0 op					
	FTST/FXAM	1	1	0	1	1	0	0	1	1 1 1 0 0	1 0 op					
	FP load constant	1	1	0	1	1	0	0	1	1 1 1 0 1	op					
	FP misc	1	1	0	1	1	0	0	1	1 1 1 1	op		SIN/COS/SINCOS/FPREM1: 387+			
	FENI/FDISI/FCLEX/FINIT	1	1	0	1	1	0	1	1	1 1 1 0 0	0 op					
287+	FSETPM	1	1	0	1	1	0	1	1	1 1 1 0 0	1 0 0					
	FFREE(P) ST(i)	1	1	0	1	1	1	P	1	1 1 0 0 0	ST(i)					
	FST(P) ST(i)	1	1	0	1	1	1	0	1	1 1 0 1 P	ST(i)					
387+	FUCOM(P) ST(0), ST(i)	1	1	0	1	1	1	0	1	1 1 1 0 P	ST(i)					
287+	FSTSW AX	1	1	0	1	1	1	1	1	1 1 1 0 0	0 0 0					

FP misc mem

FLDENV	0	0	0	0
FLDCW	0	0	0	1
FSTENV	0	0	1	0
FSTCW	0	0	1	1
FLD m80	0	1	0	1
FSTP m80	0	1	1	1
FRSTOR	1	0	0	0
FSAVE	1	0	1	0
FSTCW	1	0	1	1
FBLD	1	1	0	0
FILD m64	1	1	0	1
FBSTP	1	1	1	0
FISTP m64	1	1	1	1

FP load constant

FLD1	0	0	0
FLDL2T	0	0	1
FLDL2E	0	1	0
FLDPI	0	1	1
FLDLG2	1	0	0
FLDLN2	1	0	1
FLDZ	1	1	0

mod	addressing mode
reg	general purpose register
r/m	if mod = 11 like reg, else more addressing mode
seg	segment register
cr/dr/tr	control/test/debug register

w	0: byte instruction; 1: word/dword instruction
d	if 1, flip operands
s	1: force 8 bit immediate
cc	condition code
n	negate condition
f	0: return near; 1: return far
i	0: no immediate; 1: 16 bit immediate
cl	0: shift by 1 or immediate; 1: shift by CL
P	1: pop ST(0) after instruction
R	0: dest = dest op src; 1: dest = src op dest
imm8	8 bit immediate
imm16	16 bit immediate
imm	immediate of operand size
off	16/32 bit absolute address
rel8	8 bit relative address
rel	relative address of address size

FP misc

F2XM1	0	0	0	0
FYL2X	0	0	0	1
FPTAN	0	0	1	0
FPATAN	0	0	1	1
FXTRACT	0	1	0	0
FPREM1	0	1	0	1
FDECSTP	0	1	1	0
FINCSTP	0	1	1	1
FPREM	1	0	0	0
FYL2XP1	1	0	0	1
FSQRT	1	0	1	0
FSINCOS	1	0	1	1
FRNDINT	1	1	0	0
FSCALE	1	1	0	1
FSIN	1	1	1	0
FCOS	1	1	1	1

16 bit modr/m byte

	mod		mode		
indexed, absolute	0	0	1	1	0
register	1	1	reg		

mod ≠ 11, mod:mode ≠ 00:110 +disp16

16 bit modes

BX+SI	0	0	0
BX+DI	0	0	1
BP+SI	0	1	0
BP+DI	0	1	1
SI	1	0	0
DI	1	0	1
BP	1	1	0
BX	1	1	1

32 bit modr/m byte, SIB byte

	mod		reg							
indexed	0	0	1	0	1					
scaled index	0	0	1	0	0	scale	index	1	0	1
SIB	mod		1	0	0	scale	index	base		
register	1	1	reg							

mod ≠ 11, reg ≠ 100, mod:reg ≠ 00:101 +disp32

+disp32

mod ≠ 11, mod:base ≠ 00:101

mod	00: no disp, 01: disp8, 10: disp
base	base register (GPR)
scale	scale as a power of 2
index	100: no index, else index register (GPR)