Intel Skylake

List of instruction timings and µop breakdown

Explanation of column headings:

Instruction: Name of instruction. Multiple names mean that these instructions have the same data. Instructions with or without V name prefix behave the same unless otherwise noted.

Operands: i = immediate data, r = register, mm = 64 bit mmx register, x = 128 bit xmm register, mm/x = mmx or xmm register, y = 256 bit ymm register, v = any vector register (mmx, xmm, ymm). m = memory operand, m32 = 32-bit memory operand, etc.

µops fused domain: The number of µops at the decode, rename and allocate stages in the pipeline. Fused µops count as one.

µops unfused domain: The total number of µops for all execution port. Fused µops count as two. Fused macro-ops count as one. The instruction has µop fusion if this number is higher than the number under fused domain. Some operations are not counted here if they do not go to any execution port or if the counters are inaccurate.

µops each port: The number of µops for each execution port. p0 means a µop to execution port 0. p01means a µop that can go to either port 0 or port 1. p0 p1 means two µops going to port 0 and 1, respectively.
Port 0: Integer, f.p. and vector ALU, mul, div, branch
Port 1: Integer, f.p. and vector ALU
Port 2: Load
Port 3: Load
Port 4: Store
Port 5: Integer and vector ALU
Port 6: Integer ALU, branch
Port 7: Store address

Latency: This is the delay that the instruction generates in a dependency chain. The numbers are minimum values. Cache misses, misalignment, and exceptions may increase the clock counts considerably. Where hyperthreading is enabled, the use of the same execution units in the other thread leads to inferior performance. Subnormal numbers, NAN's and infinity do not increase the latency. The time unit used is core clock cycles, not the reference clock cycles given by the time stamp counter.

Reciprocal throughput: The average number of core clock cycles per instruction for a series of independent instructions of the same kind in the same thread.

Integer instructions

Instruction	Operands	µops fused domain	µops unfused domain	µops each port	Latency	Reciprocal through put	Comments
Move instructions							
MOV	r,i	1	1	p0156		0.25	
MOV	r8/16,r8/16	1	1	p0156	1	0.25	
MOV	r32/64,r32/64	1	1	p0156	0-1	0.25	may be elim.
MOV	r8l,m	1	2	p23 p0156		0.5	
MOV	r8h,m	1	1	p23		0.5	
MOV	r16,m	1	2	p23 p0156		0.5	
MOV	r32/64,m	1	1	p23	2	0.5	all addressing modes
MOV	m,r	1	2	p237 p4	2	1	
MOV	m,i	1	2	p237 p4		1	
MOVNTI	m,r	2	2	p23 p4	~400	1	

MOVSX MOVZX MOVSXD	r,r	1	1	p0156	1	0.25	
MOVSX MOVZX	r16,m8	1	2	p23 p0156		0.5	
MOVSX MOVZX MOVSXD	r,m	1	1	p23		0.5	all other combinations
CMOVcc	r,r	1	1	p06	1	0.5	
CMOVcc	r,m	2	2	p06 p23		0.5	
XCHG	r,r	3	3	3p0156	2	1	
XCHG	r,m	8	8		23		implicit lock
XLAT		3	3	p23 2p0156	7	2	
PUSH	r	1	2	p237 p4	3	1	
PUSH	i	1	2	p237 p4		1	
PUSH	m	2	3	p4 2p237		1	
PUSH	stack pointer	2	3	p0156 p237 p4		1	
PUSHF(D/Q)		3	4	p1 p4 p237 p06		1	
PUSHA(D)		11	19			8	not 64 bit
POP	r	1	1	p23	2	0.5	
POP	stack pointer	3	3	p23 2p0156		3	
POP	m	2	3	2p237 p4		1	
POPF(D/Q)		9	9			20	
POPA(D)		18	18			8	not 64 bit
LAHF SAHF		1	1	p06	1	1	
SALC		3	3	3p0156	1	1	not 64 bit
LEA	r16,m	2	2	p1 p05	2-4	1	16 or 32 bit address size
LEA	r32/64,m	1	1	p15	1	0.5	1 or 2 compo-nents in address
LEA	r32/64,m	1	1	p1	3	1	3 components in address
LEA	r32/64,m	1	1	p1		1	rip relative address
BSWAP	r32	1	1	p15	1	0.5	
BSWAP	r64	2	2	p06 p15	2	1	
MOVBE	r16,m16	3	3	2p0156 p23		0.5-1	MOVBE
MOVBE	r32,m32	2	2	p15 p23		0.5	MOVBE
MOVBE	r64,m64	3	3	2p0156 p23		0.75	MOVBE
MOVBE	m16,r16	2	3	p06 p237 p4		1	MOVBE
MOVBE	m32,r32	2	3	p15 p237 p4		1	MOVBE
MOVBE	m64,r64	3	4	p06 p15 p237 p4		1	MOVBE
PREFETCHNTA/0/1/2	m	1	1	p23		0.5	
PREFETCHW	m	1	1	p23		1	PREFETCHW
LFENCE		2		none counted		4	
MFENCE		4	4	p23 p4		33	
SFENCE		2	2	p23 p4		6	
Arithmetic in-structions							
ADD SUB	r,r/i	1	1	p0156	1	0.25	
ADD SUB	r,m	1	2	p0156 p23		0.5	
ADD SUB	m,r/i	2	4	2p0156 2p237 p4	5	1	
ADC SBB	r,r/i	1	1	p06	1	1	

ADC SBB	r,m	2	2	p06 p23		1	
ADC SBB	m,r/i	4	6	3p0156 2p237 p4	5	2	
CMP	r,r/i	1	1	p0156	1	0.25	
CMP	m,r/i	1	2	p0156 p23	1	0.5	
INC DEC NEG NOT	r	1	1	p0156	1	0.25	
INC DEC NOT	m	3	4	p0156 2p237 p4	5-6	1	
NEG	m	2	4	p0156 2p237 p4	5-6	1	
AAA		2	2	p1 p56	4		not 64 bit
AAS		2	2	p1 p056	4		not 64 bit
DAA DAS		3	3	p1 2p056	4		not 64 bit
AAD		3	3	p1 2p056	4		not 64 bit
AAM		11	11	p0 p1 p5 p6	23	7	not 64 bit
MUL IMUL	r8	1	1	p1	3	1	
MUL IMUL	r16	4	4	p1 p0156	4	2	
MUL IMUL	r32	3	3	p1 p0156	4	1	
MUL IMUL	r64	2	2	p1 p6	3	1	
MUL IMUL	m8	1	2	p1 p23		1	
MUL IMUL	m16	4	5	p1 3p0156 p23		2	
MUL IMUL	m32	3	4	p1 2p0156 p23		2	
MUL IMUL	m64	2	3	p1 p6 p23		1	
IMUL	r,r	1	1	p1	3	1	
IMUL	r,m	1	2	p1 p23		1	
IMUL	r16,r16,i	2	2	p1 p0156	4	1	
IMUL	r32,r32,i	1	1	p1	3	1	
IMUL	r64,r64,i	1	1	p1	3	1	
IMUL	r16,m16,i	2	3	p1 p0156 p23		1	
IMUL	r32,m32,i	1	2	p1 p23		1	
IMUL	r64,m64,i	1	2	p1 p23		1	
MULX	r32,r32,r32	3	3	p1 2p056	4	1	BMI2
MULX	r32,r32,m32	3	4	p1 2p056 p23		1	BMI2
MULX	r64,r64,r64	2	2	p1 p5	4	1	BMI2
MULX	r64,r64,m64	2	3	p1 p6 p23		1	BMI2
DIV	r8	10	10	p0 p1 p5 p6	23	6	
DIV	r16	10	10	p0 p1 p5 p6	23	6	
DIV	r32	10	10	p0 p1 p5 p6	26	6	
DIV	r64	36	36	p0 p1 p5 p6	35-88	21-83	
IDIV	r8	11	11	p0 p1 p5 p6	24	6	
IDIV	r16	10	10	p0 p1 p5 p6	23	6	
IDIV	r32	10	10	p0 p1 p5 p6	26	6	
IDIV	r64	57	57	p0 p1 p5 p6	42-95	24-90	
CBW		1	1	p0156	1		
CWDE		1	1	p0156	1		
CDQE		1	1	p0156	1		
CWD		2	2	p0156	1		
CDQ		1	1	p06	1		
CQO		1	1	p06	1		
POPCNT	r,r	1	1	p1	3	1	SSE4.2
POPCNT	r,m	1	2	p1 p23		1	SSE4.2
CRC32	r,r	1	1	p1	3	1	SSE4.2
CRC32	r,m	1	2	p1 p23		1	SSE4.2

Logic instructions							
AND OR XOR	r,r/i	1	1	p0156	1	0.25	
AND OR XOR	r,m	1	2	p0156 p23		0.5	
AND OR XOR	m,r/i	2	4	2p0156 2p237 p4	5	1	
TEST	r,r/i	1	1	p0156	1	0.25	
TEST	m,r/i	1	2	p0156 p23	1	0.5	
SHR SHL SAR	r,i	1	1	p06	1	0.5	
SHR SHL SAR	m,i	3	4	2p06 p237 p4		2	
SHR SHL SAR	r,cl	3	3	3p06	2	2	
SHR SHL SAR	m,cl	5	6	3p06 2p23 p4		4	
ROR ROL	r,1	2	2	2p06	1	1	short form
ROR ROL	r,i	1	1	p06	1	0.5	
ROR ROL	m,i	4	5	2p06 2p237 p4		2	
ROR ROL	r,cl	3	3	3p06	2	2	
ROR ROL	m,cl	5	6	3p06 p23 p4		4	
RCR RCL	r,1	3	3	2p06 p0156	2	2	
RCR RCL	m,1	4	6			3	
RCR RCL	r,i	8	8	p0156	6	6	
RCR RCL	m,i	11	11			6	
RCR RCL	r,cl	8	8	p0156	6	6	
RCR RCL	m,cl	11	11			6	
SHRD SHLD	r,r,i	1	1	p1	3	1	
SHRD SHLD	m,r,i	3	5			2	
SHLD	r,r,cl	4	4	p0156	3	2	
SHRD	r,r,cl	4	4	p0156	4	2	
SHRD SHLD	m,r,cl	5	7			4	
SHLX SHRX SARX	r,r,r	1	1	p06	1	0.5	BMI2
SHLX SHRX SARX	r,m,r	2	2	p06 p23		0.5	BMI2
RORX	r,r,i	1	1	p06	1	0.5	BMI2
RORX	r,m,i	2	2	p06 p23		0.5	BMI2
BT	r,r/i	1	1	p06	1	0.5	
BT	m,r	10	10			5	
BT	m,i	2	2	p06 p23		0.5	
BTR BTS BTC	r,r/i	1	1	p06	1	0.5	
BTR BTS BTC	m,r	10	11			5	
BTR BTS BTC	m,i	3	4	p06 p4 p23		1	
BSF BSR	r,r	1	1	p1	3	1	
BSF BSR	r,m	1	2	p1 p23		1	
SETcc	r	1	1	p06	1	0.5	
SETcc	m	2	3	p06 p237 p4		1	
CLC		1	0	none		0.25	
STC		1	1	p0156		0.25	
CMC		1	1	p0156	1	1	
CLD STD		3	3	p15 p6		4	
LZCNT	r,r	1	1	p1	3	1	LZCNT
LZCNT	r,m	1	2	p1 p23		1	LZCNT
TZCNT	r,r	1	1	p1	3	1	BMI1
TZCNT	r,m	1	2	p1 p23		1	BMI1
ANDN	r,r,r	1	1	p15	1	0.5	BMI1
ANDN	r,r,m	1	2	p15 p23	1	0.5	BMI1
BLSI BLSMSK BLSR	r,r	1	1	p15	1	0.5	BMI1

BLSI BLSMSK BLSR	r,m	1	2	p15 p23		0.5	BMI1
BEXTR	r,r,r	2	2	2p0156	2	0.5	BMI1
BEXTR	r,m,r	3	3	2p0156 p23		1	BMI1
BZHI	r,r,r	1	1	p15	1	0.5	BMI2
BZHI	r,m,r	1	2	p15 p23		0.5	BMI2
PDEP	r,r,r	1	1	p1	3	1	BMI2
PDEP	r,r,m	1	2	p1 p23		1	BMI2
PEXT	r,r,r	1	1	p1	3	1	BMI2
PEXT	r,r,m	1	2	p1 p23		1	BMI2
Control transfer instructions							
JMP	short/near	1	1	p6		1-2	
JMP	r	1	1	p6		2	
JMP	m	1	2	p23 p6		2	
Conditional jump	short/near	1	1	p6		1-2	predicted taken
Conditional jump	short/near	1	1	p06		0.5-1	predicted not taken
Fused arithmetic and branch		1	1	p6		1-2	predicted taken
Fused arithmetic and branch		1	1	p06		0.5-1	predicted not taken
J(E/R)CXZ	short	2	2	p0156 p6		0.5-2	
LOOP	short	7	7			5	
LOOP(N)E	short	11	11			6	
CALL	near	2	3	p237 p4 p6		3	
CALL	r	2	3	p237 p4 p6		2	
CALL	m	3	4	2p237 p4 p6		3	
RET		1	2	p237 p6		1	
RET	i		2			2	
BOUND	r,m	15	15			8	not 64 bit
INTO		5	5			6	not 64 bit
String instructions							
LODSB/W		3	3	2p0156 p23		1	
LODSD/Q		2	2	p0156 p23		1	
REP LODS		5n+12				~2n	
STOS		3	3	p23 p0156 p4		1	
REP STOS		<2n				~0.5n	worst case
REP STOS		2.6/32B				1/32B	best case aligned by 32
MOVS		5	5	2p23 p4 2p0156		4	
REP MOVS		~2n				< 1n	worst case
REP MOVS		4/32B				1/32B	best case aligned by 32
SCAS		3	3	p23 2p0156		1	
REP SCAS		≥6n				≥2n	
CMPS		5	5	2p23 3p0156		4	
REP CMPS		≥8n				≥2n	
Synchronization instructions							
XADD	m,r	4	5			5	

LOCK XADD	m,r	9	9			18	
LOCK ADD	m,r	8	8			18	
CMPXCHG	m,r	5	6			6	
LOCK CMPXCHG	m,r	10	10			18	
CMPXCHG8B	m,r	16	16			11	
LOCK CMPXCHG8B	m,r	20	20			19	
CMPXCHG16B	m,r	23	23			16	
LOCK CMPXCHG16B	m,r	25	25			26	
Other							
NOP (90)		1	0	none		0.25	
Long NOP (0F 1F)		1	0	none		0.25	
PAUSE		4	4	p6			
ENTER	a,0	12	12			8	
ENTER	a,b	~14+7b	~45+7b		~87+2b		
LEAVE		3	3	2p0156 p23		5	
XGETBV		15	15			9	XGETBV
CPUID		27-118				100-250	
RDTSC		20	20			25	
RDTSCP		22	22			32	RDTSCP
RDPMC		35	35			40	
RDRAND	r	16	16	p23 15p0156		~460	RDRAND
RDSEED	r	16	16	p23 15p0156		~460	RDSEED

Floating point x87 instructions

Instruction	Operands	µops fused domain	µops unfused domain	µops each port	Latency	Recipro-cal through put	Comments
Move instruc-tions							
FLD	r	1	1	p05	1	0.5	
FLD	m32/64	1	1	p23	3	0.5	
FLD	m80	4	4	2p01 2p23	4	2	
FBLD	m80	43	43		46	22	
FST(P)	r	1	1	p05	1	0.5	
FST(P)	m32/m64	1	2	p4 p237	3	1	
FSTP	m80	7	7	3p0156 2p23 2p4	4	5	
FBSTP	m80	244	226		264	266	
FXCH	r	2	0	none	0	0.5	
FILD	m	1	2	p05 p23	5	1	
FIST(P)	m	3	3	p5 p23 p4	7	1	
FISTTP	m	3	3	p1 p23 p4	7	2	SSE3
FLDZ		1	1	p05		1	
FLD1		2	2	2p05		2	
FLDPI FLDL2E etc.		2	2	2p05		2	
FCMOVcc	r	4	4	p0 p1 p56	3	2	
FNSTSW	AX	2	2	p0 p0156	6	2	
FNSTSW	m16	2	3	p0 p4 p237	6	1	
FLDCW	m16	3	3	p01 p23 p6	7	2	
FNSTCW	m16	2	3	p237 p4 p6	6	1	
FINCSTP FDECSTP		1	1	p05	0	0.5	

FFREE(P)	r	1	1	p05		0.5	
FNSAVE	m	133	133		176	176	
FRSTOR	m	89	89		175	175	
Arithmetic instructions							
FADD(P) FSUB(R)(P)	r	1	1	p5	3	1	
FADD(P) FSUB(R)(P)	m	2	3	p5 p23		1	
FMUL(P)	r	1	1	p0	5	1	
FMUL(P)	m	2	3	p0 p23		1	
FDIV(R)(P)	r	1	1	p0	14-16	4-5	
FDIV(R)(P)	m	1	2	p0 p23		4-5	
FABS		1	1	p0	1	1	
FCHS		1	1	p0	1	1	
FCOM(P) FUCOM	r	1	1	p5	3	1	
FCOM(P) FUCOM	m	1	2	p5 p23		1	
FCOMPP FUCOMPP		2	2	p0 p5		1	
FCOMI(P) FUCOMI(P)	r	3	3	p5		1	
FIADD FISUB(R)	m	3	4	2p5 p23		2	
FIMUL	m	2	3	p0 p5 p23		1	
FIDIV(R)	m	2	3	p0 p5 p23			
FICOM(P)	m	2	3	2p5 p23		2	
FTST		1	1	p5	3	1	
FXAM		2	2	2p5	6	2	
FPREM		31	31		26-30	17	
FPREM1		31	31		30-57	17	
FRNDINT		17	17		21	11	
Math							
FSCALE		27	27		130	130	
FXTRACT		17	17		11	11	
FSQRT		1	1	p0	14-21	4-7	
FSIN		53-105			50-120		
FCOS		53-105			50-130		
FSINCOS		55-120			55-150		
F2XM1		16-90			65-80		
FYL2X		40-100			103		
FYL2XP1		56			77		
FPTAN		40-112			140-160		
FPATAN		30-160			100-160		
Other							
FNOP		1	1	p05		0.5	
WAIT		2	2	p05		2	
FNCLEX		5	5	p156		22	
FNINIT		18	18			78	

Integer vector instructions

Instruction	Operands	µops fused domain	µops unfused domain	µops each port	Latency	Reciprocal through put	Comments
Move instructions							
MOVD	r32/64,mm/x	1	1	p0	2	1	
MOVD	m32/64,mm/x	1	2	p237 p4	3	1	
MOVD	mm/x,r32/64	1	1	p5	2	1	
MOVD	mm/x,m32/64	1	1	p23	2	0.5	
MOVQ	r64,mm/x	1	1	p0	2	1	
MOVQ	mm/x,r64	1	1	p5	2	1	
MOVQ	mm,mm	1		p05	1	0.5	
MOVQ	x,x	1		p015	1	0.33	
MOVQ	mm/x,m64	1	1	p23	2	0.5	
MOVQ	m64, mm/x	1	2	p237 p4	3	1	
MOVDQA/U	x,x	1	1	p015	0-1	0.25	may eliminate
MOVDQA/U	x, m128	1	1	p23	2	0.5	
MOVDQA/U	m128, x	1	2	p237 p4	3	1	
VMOVDQA/U	y,y	1	1	p015	0-1	0.25	may eliminate
VMOVDQA/U	y,m256	1	1	p23	3	0.5	AVX
VMOVDQA/U	m256,y	1	2	p237 p4	3	1	AVX
LDDQU	x, m128	1	1	p23	3	0.5	SSE3
MOVDQ2Q	mm, x	2	2	p0 p5	2	1	
MOVQ2DQ	x,mm	2	2	p0 p15	2	1	
MOVNTQ	m64,mm	1	2	p237 p4	~418	1	
MOVNTDQ	m128,x	1	2	p237 p4	~450	1	
VMOVNTDQ	m256,y	1	2	p237 p4	~400	1	AVX2
MOVNTDQA	x, m128	2	2	p23 p015	3	0.5	SSE4.1
VMOVNTDQA	y,m256	2	2	p23 p015	3	0.5	AVX2
PACKSSWB/DW PACKUSWB	mm,mm	3	3	p5	2	2	
PACKSSWB/DW PACKUSWB	mm,m64	3	3	p23 2p5		2	
PACKSSWB/DW PACKUSWB	x,x / y,y,y	1	1	p5	1	1	
PACKSSWB/DW PACKUSWB	x,m / y,y,m	1	2	p23 p5		1	
PACKUSDW	x,x / y,y,y	1	1	p5	1	1	SSE4.1
PACKUSDW	x,m / y,y,m	1	2	p23 p5		1	SSE4.1
PUNPCKH/L BW/WD/DQ	v,v / v,v,v	1	1	p5	1	1	
PUNPCKH/L BW/WD/DQ	v,m / v,v,m	1	2	p23 p5		1	
PUNPCKH/L QDQ	x,x / y,y,y	1	1	p5	1	1	
PUNPCKH/L QDQ	x,m / y,y,m	1	2	p23 p5		1	
PMOVSX/ZX BW BD BQ DW DQ	x,x	1	1	p5	1	1	SSE4.1
PMOVSX/ZX BW BD BQ DW DQ	x,m	1	2	p23 p5		1	SSE4.1
VPMOVSX/ZX BW BD BQ DW DQ	y,x	1	1	p5	3	1	AVX2
VPMOVSX/ZX BW BD BQ DW DQ	y,m	2	2	p5 p23		1	AVX2

PSHUFB	v,v / v,v,v	1	1	p5	1	1	SSSE3
PSHUFB	v,m / v,v,m	2	2	p23 p5		1	SSSE3
PSHUFW	mm,mm,i	1	1	p5	1	1	
PSHUFW	mm,m64,i	2	2	p23 p5		1	
PSHUFD	v,v,i	1	1	p5	1	1	
PSHUFD	v,m,i	1-2	2	p23 p5		1	
PSHUFL/HW	v,v,i	1	1	p5	1	1	
PSHUFL/HW	v,m,i	2	2	p23 p5		1	
PALIGNR	v,v,i / v,v,v,i	1	1	p5	1	1	SSSE3
PALIGNR	v,m,i / v,v,m,i	2	2	p23 p5		1	SSSE3
PBLENDVB	x,x,xmm0	1	1	p015	1	1	SSE4.1
PBLENDVB	x,m,xmm0	2	2	p015 p23		2	SSE4.1
VPBLENDVB	v,v,v,v	2	2	2p015	2	1	AVX2
VPBLENDVB	v,v,m,v	3	3	2p015 p23		2	AVX2
PBLENDW	x,x,i / v,v,v,i	1	1	p5	1	1	SSE4.1
PBLENDW	x,m,i / v,v,m,i	2	2	p23 p5		1	SSE4.1
VPBLENDD	v,v,v,i	1	1	p015	1	0.33	AVX2
VPBLENDD	v,v,m,i	2	2	p015 p23		0.5	AVX2
VPERMD	y,y,y	1	1	p5	3	1	AVX2
VPERMD	y,y,m	1	2	p5 p23		1	AVX2
VPERMQ	y,y,i	1	1	p5	3	1	AVX2
VPERMQ	y,m,i	2	2	p5 p23		1	AVX2
VPERM2I128	y,y,y,i	1	1	p5	3	1	AVX2
VPERM2I128	y,y,m,i	2	2	p5 p23		1	AVX2
MASKMOVQ	mm,mm	4	4	p0 p4 2p23	~450	2	
MASKMOVDQU	x,x	10	10	4p04 2p56 4p23	18-500	6	
VPMASKMOVD/Q	v,v,m	2	2	p23 p015	4	0.5	AVX2
VPMASKMOVD/Q	m,v,v	3	3	p0 p4 p23	14	1	AVX2
PMOVMSKB	r,v	1	1	p0	2-3	1	
PEXTRB/W/D/Q	r,x,i	2	2	p0 p5	3	1	SSE4.1
PEXTRB/W/D/Q	m,x,i	2	3	p23 p4 p5		1	SSE4.1
VEXTRACTI128	x,y,i	1	1	p5	3	1	AVX2
VEXTRACTI128	m,y,i	2	2	p23 p4	4	1	AVX2
PINSRB	x,r32,i	2	2	2p5	3	2	SSE4.1
PINSRB	x,m8,i	2	2	p23 p5		1	SSE4.1
PINSRW	mm/x,r32,i	2	2	p5	3	2	
PINSRW	mm/x,m16,i	2	2	p23 p5		1	
PINSRD/Q	x,r32,i	2	2	2p5	3	2	SSE4.1
PINSRD/Q	x,m32,i	2	2	p23 p5		1	SSE4.1
VINSERTI128	y,y,x,i	1	1	p5	3	1	AVX2
VINSERTI128	y,y,m,i	2	2	p015 p23	3	0.5	AVX2
VPBROADCAST B/W/D/Q	x,x	1	1	p5	1	1	AVX2
VPBROADCAST B/W	x,m8/16	2	2	p23 p5	7	1	AVX2
VPBROADCAST D/Q	x,m32/64	1	1	p23	4	0.5	AVX2
VPBROADCAST B/W/D/Q	y,x	1	1	p5	3	1	AVX2
VPBROADCAST B/W	y,m8/16	2	2	p23 p5	7	1	AVX2
VPBROADCAST D/Q	y,m32/64	1	1	p23	3	0.5	AVX2
VBROADCASTI128	y,m128	1	1	p23	3	0.5	AVX2
VPGATHERDD	x,[r+s*x],x	4	4	p0 p1 p23 p5		4	AVX2

VPGATHERDD	y,[r+s*y],y	4	4	p0 p1 p23 p5		5	AVX2
VPGATHERQD	x,[r+s*x],x	5	5	p0 p1 p23 p5		2	AVX2
VPGATHERQD	x,[r+s*y],x	4	4	p0 p1 p23 p5		4	AVX2
VPGATHERDQ	x,[r+s*x],x	5	5	p0 p1 p23 p5		2	AVX2
VPGATHERDQ	y,[r+s*x],y	4	4	p0 p1 p23 p5		4	AVX2
VPGATHERQQ	x,[r+s*x],x	5	5	p0 p1 p23 p5		2	AVX2
VPGATHERQQ	y,[r+s*y],y	4	4	p0 p1 p23 p5		4	AVX2
Arithmetic instructions							
PADD/SUB(S,US) B/W/D/Q	v,v / v,v,v	1	1	p015	1	0.33	
PADD/SUB(S,US) B/W/D/Q	v,m / v,v,m	1	2	p015 p23		0.5	
PHADD(S)W/D PHSUB(S)W/D	v,v / v,v,v	3	3	p01 2p5	3	2	SSSE3
PHADD(S)W/D PHSUB(S)W/D	v,m / v,v,m	4	4	p01 2p5 p23		2	SSSE3
PCMPEQB/W/D PCMPGTB/W/D	mm,mm	1	1	p0	1	1	
PCMPEQB/W/D PCMPGTB/W/D	x,x / y,y,y	1	1	p01	1	0.5	
PCMPEQB/W/D PCMPGTB/W/D	x,m / y,y,m	1	2	p01 p23		0.5	
PCMPEQQ	v,v / v,v,v	1	1	p01	1	0.5	SSE4.1
PCMPEQQ	v,m / v,v,m	1	2	p01 p23		0.5	SSE4.1
PCMPGTQ	v,v / v,v,v	1	1	p5	3	1	SSE4.2
PCMPGTQ	v,m / v,v,m	1	2	p5 p23		1	SSE4.2
PMULL/HW PMULHUW	mm,mm	1	1	p0	5	1	
PMULL/HW PMULHUW	x,x / y,y,y	1	1	p01	5	0.5	
PMULL/HW PMULHUW	x,m / y,y,m	1	2	p01 p23		0.5	
PMULHRSW	mm,mm	1	1	p0	5	1	SSSE3
PMULHRSW	x,x / y,y,y	1	1	p01	5	0.5	SSSE3
PMULHRSW	x,m / y,y,m	1	2	p01 p23		0.5	SSSE3
PMULLD	x,x / y,y,y	2	2	2p01	10	1	SSE4.1
PMULLD	x,m / y,y,m	3	3	2p01 p23		1	SSE4.1
PMULDQ	x,x / y,y,y	1	1	p01	5	0.5	SSE4.1
PMULDQ	x,m / y,y,m	1	2	p01 p23		0.5	SSE4.1
PMULUDQ	mm,mm	1	1	p0	5	1	
PMULUDQ	x,x / y,y,y	1	1	p01	5	0.5	
PMULUDQ	x,m / y,y,m	1	2	p01 p23		0.5	
PMADDWD	mm,mm	1	1	p0	5	1	
PMADDWD	x,x / y,y,y	1	1	p01	5	0.5	
PMADDWD	x,m / y,y,m	1	2	p01 p23		0.5	
PMADDUBSW	mm,mm	1	1	p0	5	1	SSSE3
PMADDUBSW	x,x / y,y,y	1	1	p01	5	0.5	SSSE3
PMADDUBSW	x,m / y,y,m	1	2	p01 p23		0.5	SSSE3
PAVGB/W	mm,mm	1	1	p0	1	1	
PAVGB/W	x,x / y,y,y	1	1	p01	1	0.5	
PAVGB/W	x,m / y,y,m	1	2	p01 p23		0.5	

PMIN/PMAX SB/SW/SD UB/UW/UD	mm,mm	1	1	p0	1	1	SSE4.1
PMIN/PMAX SB/SW/SD UB/UW/UD	x,x / y,y,y	1	1	p01	1	0.5	SSE4.1
PMIN/PMAX SB/SW/SD UB/UW/UD	x,m / y,y,m	1	2	p01 p23		0.5	SSE4.1
PHMINPOSUW	x,x	1	1	p0	4	1	SSE4.1
PHMINPOSUW	x,m128	1	2	p0 p23		1	SSE4.1
PABSB/W/D	mm,mm	1	1	p0	1	1	SSSE3
PABSB/W/D	x,x / y,y	1	1	p01	1	0.5	SSSE3
PABSB/W/D	x,m / y,m	1	2	p01 p23		0.5	SSSE3
PSIGNB/W/D	mm,mm	1	1	p0	1	1	SSSE3
PSIGNB/W/D	x,x / y,y,y	1	1	p01	1	0.5	SSSE3
PSIGNB/W/D	x,m / y,y,m	1	2	p01 p23		0.5	SSSE3
PSADBW	v,v / v,v,v	1	1	p5	3	1	
PSADBW	v,m / v,v,m	1	2	p5 p23		1	
MPSADBW	x,x,i / v,v,v,i	2	2	2p5	4	2	SSE4.1
MPSADBW	x,m,i / v,v,m,i	3	3	2p5 p23		2	SSE4.1
Logic instructions							
PAND PANDN POR PXOR	mm,mm	1	1	p05	1	0.5	
PAND PANDN POR PXOR	x,x / y,y,y	1	1	p015	1	0.33	
PAND PANDN POR PXOR	v,m / v,v,m	1	2	p015 p23		0.5	
PTEST	v,v	2	2	p0 p5	3	1	SSE4.1
PTEST	v,m	2	3	p0 p5 p23		1	SSE4.1
PSLLW/D/Q PSRLW/D/Q PSRAW/D/Q	mm,mm	1	1	p0	1	1	
PSLLW/D/Q PSRLW/D/Q PSRAW/D/Q	mm,m64	2	2	p0 p23		1	
PSLLW/D/Q PSRLW/D/Q PSRAW/D/Q	x,x / v,v,x	2	2	p01 p5	1	1	
PSLLW/D/Q PSRLW/D/Q PSRAW/D/Q	x,m / v,v,m	2	2	p01 p23		0.5	
PSLLW/D/Q PSRLW/D/Q PSRAW/D/Q	mm,i	1	1	p0	1	1	
PSLLW/D/Q PSRLW/D/Q PSRAW/D/Q	x,i / y,y,i	1	1	p01	1	0.5	
VPSLLVD/Q VPSRAVD VPSRLVD/Q	v,v,v	1	1	p01	1	0.5	AVX2
VPSLLVD/Q VPSRAVD VPSRLVD/Q	v,v,m	1	2	p01 p23		0.5	AVX2

PSLLDQ PSRLDQ	x,i / v,v,i	1	1	p5	1	1	
String instructions							
PCMPESTRI	x,x,i	8	8	6p05 2p16	12	4	SSE4.2
PCMPESTRI	x,m128,i	8	8	3p0 2p16 2p5 p23		4	SSE4.2
PCMPESTRM	x,x,i	9	9	3p0 2p16 4p5	9	5	SSE4.2
PCMPESTRM	x,m128,i	9	9	6p05 2p16 p23		5	SSE4.2
PCMPISTRI	x,x,i	3	3	3p0	10	3	SSE4.2
PCMPISTRI	x,m128,i	4	4	3p0 p23		3	SSE4.2
PCMPISTRM	x,x,i	3	3	3p0	9	3	SSE4.2
PCMPISTRM	x,m128,i	4	4	3p0 p23		3	SSE4.2
Encryption instructions							
PCLMULQDQ	x,x,i	1	1	p5	7	1	CLMUL
PCLMULQDQ	x,m,i	2	2	p5 p23		1	CLMUL
AESDEC, AESDECLAST, AESENC, AESENCLAST	x,x	1	1	p0	4	1	AES
AESDEC, AESDECLAST, AESENC, AESENCLAST	x,m	2	2	p0 p23		1.5	AES
AESIMC	x,x	2	2	2p0	8	2	AES
AESIMC	x,m	3	3	2p0 p23		2	AES
AESKEYGENASSIST	x,x,i	13	13	p0 p5	12	12	AES
AESKEYGENASSIST	x,m,i	13	13			12	AES
Other							
EMMS		10	10	p05		6	

Floating point XMM and YMM instructions

Instruction	Operands	µops fused domain	µops unfused domain	µops each port	Latency	Reciprocal through put	Comments
Move instructions							
MOVAPS/D	x,x	1	1	p015	0-1	0.25	may eliminate
VMOVAPS/D	y,y	1	1	p015	0-1	0.25	may eliminate
MOVAPS/D MOVUPS/D	x,m128	1	1	p23	2	0.5	
VMOVAPS/D VMOVUPS/D	y,m256	1	1	p23	3	0.5	AVX
MOVAPS/D MOVUPS/D	m128,x	1	2	p237 p4	3	1	
VMOVAPS/D VMOVUPS/D	m256,y	1	2	p237 p4	3	1	AVX
MOVSS/D	x,x	1	1	p5	1	1	
MOVSS/D	x,m32/64	1	1	p23	3	0.5	
MOVSS/D	m32/64,x	1	2	p237 p4	3	1	

MOVHPS/D	x,m64	1	2	p23 p5	4	1	
MOVHPS/D	m64,x	1	2	p4 p237	3	1	
MOVLPS/D	x,m64	1	2	p23 p5	4	1	
MOVLPS/D	m64,x	1	2	p4 p237	3	1	
MOVHLPS	x,x	1	1	p5	1	1	
MOVLHPS	x,x	1	1	p5	1	1	
MOVMSKPS/D	r32,x	1	1	p0	2	1	
VMOVMSKPS/D	r32,y	1	1	p0	3	1	
MOVNTPS/D	m128,x	1	2	p4 p237	~400	1	
VMOVNTPS/D	m256,y	1	2	p4 p237	~400	1	AVX
SHUFPS/D	x,x,i / v,v,v,i	1	1	p5	1	1	
SHUFPS/D	x,m,i / v,v,m,i	2	2	p5 p23		1	
VPERMILPS/PD	v,v,i	1	1	p5	1	1	AVX
VPERMILPS/PD	v,m,i	2	2	p5 p23		1	AVX
VPERMILPS/PD	v,v,v	1	1	p5	1	1	AVX
VPERMILPS/PD	v,v,m	2	2	p5 p23		1	AVX
VPERM2F128	y,y,y,i	1	1	p5	3	1	AVX
VPERM2F128	y,y,m,i	2	2	p5 p23		1	AVX
VPERMPS	y,y,y	1	1	p5	3	1	AVX2
VPERMPS	y,y,m	1	2	p5 p23		1	AVX2
VPERMPD	y,y,i	1	1	p5	3	1	AVX2
VPERMPD	y,m,i	2	2	p5 p23		1	AVX2
BLENDPS/PD	x,x,i / v,v,v,i	1	1	p015	1	0.33	SSE4.1
BLENDPS/PD	x,m,i / v,v,m,i	2	2	p015 p23		0.5	SSE4.1
BLENDVPS/PD	x,x,xmm0	1	1	p015	1	1	SSE4.1
BLENDVPS/PD	x,m,xmm0	2	2	p015 p23		1	SSE4.1
VBLENDVPS/PD	v,v,v,v	2	2	2p015	2	1	AVX
VBLENDVPS/PD	v,v,m,v	3	3	2p015 p23		1	AVX
MOVDDUP	v,v	1	1	p5	1	1	SSE3
MOVDDUP	v,m	1	1	p23	3	0.5	SSE3
VBROADCASTSS	x,m32	1	1	p23	2	0.5	AVX
VBROADCASTSS	y,m32	1	1	p23	3	0.5	AVX
VBROADCASTSS	x,x	1	1	p5	1	1	AVX2
VBROADCASTSS	y,x	1	1	p5	3	1	AVX2
VBROADCASTSD	y,m64	1	1	p23	3	0.5	AVX
VBROADCASTSD	y,x	1	1	p5	3	1	AVX2
VBROADCASTF128	y,m128	1	1	p23	3	0.5	AVX
MOVSH/LDUP	v,v	1	1	p5	1	1	SSE3
MOVSH/LDUP	v,m	1	1	p23	3	0.5	SSE3
UNPCKH/LPS/D	x,x / v,v,v	1	1	p5	1	1	SSE3
UNPCKH/LPS/D	x,m / v,v,m	1	2	p5 p23		1	SSE3
EXTRACTPS	r32,x,i	2	2	p0 p5		1	SSE4.1
EXTRACTPS	m32,x,i	2	3	p4 p5 p23	5	1	SSE4.1
VEXTRACTF128	x,y,i	1	1	p5	3	1	AVX
VEXTRACTF128	m128,y,i	2	2	p23 p4	6	1	AVX
INSERTPS	x,x,i	1	1	p5	1	1	SSE4.1
INSERTPS	x,m32,i	2	2	p23 p5	4	1	SSE4.1
VINSERTF128	y,y,x,i	1	1	p5	3	1	AVX
VINSERTF128	y,y,m128,i	2	2	p015 p23	5	0.5	AVX
VMASKMOVPS/D	v,v,m	2	2	p015 p23	3	0.5	AVX
VMASKMOVPS/D	m128,x,x	4	4	p0 p4 p23	13	1	AVX
VMASKMOVPS/D	m256,y,y	4	4	p0 p4 p23	13	1	AVX
VGATHERDPS	x,[r+s*x],x	4	4	p0 p1 p23 p5	12	4	AVX2

VGATHERDPS	y,[r+s*y],y	4	4	p0 p1 p23 p5	13	5	AVX2
VGATHERQPS	x,[r+s*x],x	5	5	p0 p1 p23 p5		2	AVX2
VGATHERQPS	x,[r+s*y],x	4	4	p0 p1 p23 p5		4	AVX2
VGATHERDPD	x,[r+s*x],x	5	5	p0 p1 p23 p5		2	AVX2
VGATHERDPD	y,[r+s*x],y	4	4	p0 p1 p23 p5		4	AVX2
VGATHERQPD	x,[r+s*x],x	5	5	p0 p1 p23 p5		2	AVX2
VGATHERQPD	y,[r+s*y],y	4	4	p0 p1 p23 p5		4	AVX2
Conversion							
CVTPD2PS	x,x	2	2	p01 p5	5	1	
CVTPD2PS	x,m128	2	3	p01 p5 p23		1	
VCVTPD2PS	x,y	2	2	p01 p5	7	1	AVX
VCVTPD2PS	x,m256	2	3	p01 p5 p23		1	AVX
CVTSD2SS	x,x	2	2	p01 p5	5	1	
CVTSD2SS	x,m64	2	3	p01 p5 p23		1	
CVTPS2PD	x,x	2	2	p01 p5	5	1	
CVTPS2PD	x,m64	1	2	p01 p5 p23		0.5	
VCVTPS2PD	y,x	2	2	p01 p5	7	1	AVX
VCVTPS2PD	y,m128	1	2	p01 p5 p23		0.5	AVX
CVTSS2SD	x,x	2	2	p01 p5	5	2	
CVTSS2SD	x,m32	1	2	p01 p5 p23		2	
CVTDQ2PS	x,x	1	1	p01	4	0.5	
CVTDQ2PS	x,m128	1	2	p01 p23		0.5	
VCVTDQ2PS	y,y	1	1	p01	4	0.5	AVX
VCVTDQ2PS	y,m256	1	2	p01 p23		0.5	AVX
CVT(T) PS2DQ	x,x	1	1	p01	4	0.5	
CVT(T) PS2DQ	x,m128	1	2	p01 p23		0.5	
VCVT(T) PS2DQ	y,y	1	1	p01	4	0.5	AVX
VCVT(T) PS2DQ	y,m256	1	2	p01 p23		0.5	AVX
CVTDQ2PD	x,x	2	2	p01 p5	5	1	
CVTDQ2PD	x,m64	2	2	p01 p23		0.5	
VCVTDQ2PD	y,x	2	2	p01 p5	7	1	AVX
VCVTDQ2PD	y,m128	1	2	p01 p23		0.5	AVX
CVT(T)PD2DQ	x,x	2	2	p01 p5	5	1	
CVT(T)PD2DQ	x,m128	3	3	p01 p23 p5		1	
VCVT(T)PD2DQ	x,y	2	2	p01 p5	7	1	AVX
VCVT(T)PD2DQ	x,m256	2	3	p01 p23 p5		1	AVX
CVTPI2PS	x,mm	2	2	p0 p1	6	2	
CVTPI2PS	x,m64	1	2	p01 p23		3	
CVT(T)PS2PI	mm,x	2	2	p0 p5	7	1	
CVT(T)PS2PI	mm,m128	2	2	p0 p23		1	
CVTPI2PD	x,mm	2	2	p01 p5	5	1	
CVTPI2PD	x,m64	1	2	p01 p23		0.5	
CVT(T) PD2PI	mm,x	2	2	p01 p5	5	1	
CVT(T) PD2PI	mm,m128	2	3	p01 p23 p5		1	
CVTSI2SS	x,r32	2	2	p01 p5	6	2	
CVTSI2SS	x,r64	3	3	p01 2p5	7	2	
CVTSI2SS	x,m32	1	2	p1 p23		3	
CVT(T)SS2SI	r32,x	2	2	2p01	6	1	
CVT(T)SS2SI	r64,x	3	3	2p01 p5	7	1	
CVT(T)SS2SI	r32,m32	3	3	2p01 p23		1	
CVTSI2SD	x,r32/64	2	2	p01 p5	6	2	
CVTSI2SD	x,m32	1	2	p01 p23		2	

CVT(T)SD2SI	r32/64,x	2	2	p0 p1	6	1	
CVT(T)SD2SI	r32,m64	3	3	2p01 p23		1	
VCVTPS2PH	x,v,i	2	2	p01 p5	5-7	1	F16C
VCVTPS2PH	m,v,i	3	3	p01 p4 p23		1	F16C
VCVTPH2PS	v,x	2	2	p01 p5	5-7	1	F16C
VCVTPH2PS	v,m	1	2	p01 p23		1	F16C
Arithmetic							
ADDSS/D PS/D SUBSS/D PS/D	x,x / v,v,v	1	1	p01	4	0.5	
ADDSS/D PS/D SUBSS/D PS/D	x,m / v,v,m	1	2	p01 p23		0.5	
ADDSUBPS/D	x,x / v,v,v	1	1	p01	4	0.5	SSE3
ADDSUBPS/D	x,m / v,v,m	1	2	p01 p23		0.5	SSE3
HADDPS/D HSUBPS/D	x,x / v,v,v	3	3	p01 2p5	6	2	SSE3
HADDPS/D HSUBPS/D	x,m / v,v,m	4	4	p1 2p5 p23		2	SSE3
MULSS/D PS/D	x,x / v,v,v	1	1	p01	4	0.5	
MULSS/D PS/D	x,m / v,v,m	1	2	p01 p23		0.5	
DIVSS	x,x	1	1	p0	11	3	
DIVPS	x,x	1	1	p0	11	3	
DIVSS DIVPS	x,m	1	2	p0 p23		3-5	
DIVSD	x,x	1	1	p0	13-14	4	
DIVPD	x,x	1	1	p0	13-14	4	
DIVSD DIVPD	x,m	1	2	p0 p23		4	
VDIVPS	y,y,y	1	1	p0	11	5	AVX
VDIVPS	y,y,m256	1	2	p0 p23		5	AVX
VDIVPD	y,y,y	1	1	p0	13-14	8	AVX
VDIVPD	y,y,m256	4	4	p0 p23		8	AVX
RCPSS/PS	v,v	1	1	p0	4	1	
RCPSS/PS	v,m	1	2	p0 p23		1	
CMPccSS/D CMPccPS/D	x,x / v,v,v	1	1	p01	4	0.5	
CMPccSS/D CMPccPS/D	x,m / v,v,m	2	2	p01 p23		0.5	
(U)COMISS/D	x,x	1	1	p0		1	
(U)COMISS/D	x,m32/64	2	2	p0 p23		1	
MAXSS/D PS/D MINSS/D PS/D	x,x / v,v,v	1	1	p01	4	0.5	
MAXSS/D PS/D MINSS/D PS/D	x,m / v,v,m	1	2	p01 p23		0.5	
ROUNDSS/D PS/D	v,v,i	2	2	2p01	8	1	SSE4.1
ROUNDSS/D PS/D	v,m,i	3	3	2p01 p23		1	SSE4.1
DPPS	x,x,i / v,v,v,i	4	4	3p01 p5	13	1.5	SSE4.1
DPPS	x,m,i / v,v,m,i	6	6	3p01 p23 p5 p6		1.5	SSE4.1
DPPD	x,x,i	3	3	2p01 p5	9	1	SSE4.1
DPPD	x,m128,i	4	4	2p01 p23 p5		1	SSE4.1
VFMADD... (all FMA instr.)	v,v,v	1	1	p01	4	0.5	FMA
VFMADD... (all FMA instr.)	v,v,m	1	2	p01 p23		0.5	FMA

Math							
SQRTSS/PS	x,x	1	1	p0	12	3	
SQRTSS/PS	x,m128	1	2	p0 p23		3	
VSQRTPS	y,y	1	1	p0	12	6	AVX
VSQRTPS	y,m256	4	4	p0 p23		6	AVX
SQRTSD	x,x	1	1	p0	15-16	4-6	
SQRTPD	x,x	1	1	p0	15-16	4-6	
SQRTSD/PD	x,m128	1	2	p0 p23		4-6	
VSQRTPD	y,y	1	1	p0	15-16	9-12	AVX
VSQRTPD	y,m256	4	4	p0 p23		9-12	AVX
RSQRTSS/PS	v,v	1	1	p0	4	1	
RSQRTSS/PS	v,m	1	2	p0 p23		1	
Logic							
AND/ANDN/OR/ XORPS/PD	x,x / v,v,v	1	1	p015	1	0.33	
AND/ANDN/OR/ XORPS/PD	x,m / v,v,m	1	2	p015 p23		0.5	
Other							
VZEROUPPER		4	4	none		1	AVX
VZEROALL		25	25	p0 p1 p5 p6		12	AVX, 32 bit
VZEROALL		34	34	p0 p1 p5 p6		12	AVX, 64 bit
LDMXCSR	m32	4	4	p0 p5 p6 p23	5	3	
STMXCSR	m32	3	4	p0 p4 p6 p237	5	2	
FXSAVE	m4096	106			78	78	32 bit mode
FXSAVE	m4096	136			64	64	64 bit mode
FXRSTOR	m4096	105			76	76	32 bit mode
FXRSTOR	m4096	121			77	77	64 bit mode
XSAVE		247			107	107	32 bit mode
XSAVE		304			107	107	64 bit mode
XRSTOR		257			122	122	32 bit mode
XRSTOR		257			122	122	64 bit mode
XSAVEOPT	m	168			74	74	