The Z80 Instruction Set

TurboMON is unique in offering the ENTIRE instruction set without ANY gaps for 'bad opcodes'. Other programs may claim to include undocumented instructions, but they don't normally cover all possibilities. The table at the end of this manual shows all possible combinations of instructions. Instructions in bold text are the normal documented ones.

The TurboMON disassembler knows which instructions can have an index prefix and separates out ones that are not used. They appear in the disassembly as [Unused IX/IY prefix] and have no effect when executed. The Z80 only uses them to signal to use IX/IY instead of HL (in most cases), so LD IX,12345 is really [Signal use IX] LD *HL*,12345. If you have 2 index prefixes following each other, using the previous example you should understand that the second one will over-ride the first one. i.e. [IX prefix] [IY prefix] LD HL,0 will perform LD IY,0 (TurboMON will show the IX prefix as unused, followed by the LD IY,0). Instructions with an ED prefix cannot have an index prefix but ones with a CB prefix can.

102 undocumented instructions that are now widely accepted mainly involve the manipulation of the index registers (blue in the table) as though they were separate 8 bit registers. They were used for a while in Spectrum 'tape loader' security as most people didn't know about them or their disassembler didn't recognise them. The instructions are formed by prefixing instructions involving the H and L registers with an index prefix (DD for IX, FD for IY). The index halves are displayed as IXl, IXh, IYl, IYh. for example: LD A,IXh (formed from LD A,H)

10 of the undocumented instructions are the SLL (shift left logical) instruction that was omitted because it doesn't work properly. Bit 0 of the result is always set after the shift operation.

Other instructions are formed by using an index prefix with CB prefixed instructions that don't use (HL) (hope that was clear!). e.g. Using an IX prefix with SRL B transforms it into: LD B,SRL(IX+d). The instruction performs SRL (IX+d) and loads B with the result. The result is stored back in (IX+d) as well as B, giving 2 instructions for the price of one! Note that the length of the new instructions is always 4 since an index displacement is now required and forms the 3rd byte in the instruction (as with all index instructions).

Because normal BIT instructions only test bits and don't store a result (like SET would), adding a prefix to BIT b,r would transform it into: BIT b,(IX+d) and NOT LD r,BIT b,(IX+d). The fact that the BIT instruction is faster than a SET/RES instruction shows us that the Z80 doesn't try to store a result.

Some instructions come from patterns in the instruction set: NEG, RETN, RETI, IM 0/1/2 are all repeated 8 or 16 bytes after the normal ones.

IN r,(C), with r as (HL), reads a byte from the port in BC but junks the result. It is displayed as IN X,(C). **Note:** The flags are still affected, as with other IN r,(C) instructions.

OUT (C),r, with r as (HL) sends zero to the port in BC. It is displayed as OUT (C),0 to show its effect.

There are spaces for 4 interrupt modes but the Z80 has only 3 (0, 1 and 2). There is a gap between mode 0 and 1 in the instruction set. The missing interrupt mode is another mode 0.

The rest of the instruction set behaves as NOPs, the ones with an ED prefix are obviously 2 byte instructions and will take 8 T-states in reality.

Hex	Dec	Normal	DD/FD Prefix	CB Prefix	ED Prefix	DDCB/FDCB Prefix
00	000	**NOP**	NOP	**RLC B**	NOP	LD B,RLC (IX+d)
01	001	**LD BC,nn**	LD BC,nn	**RLC C**	NOP	LD C,RLC (IX+d)
02	002	**LD (BC),A**	LD (BC),A	**RLC D**	NOP	LD D,RLC (IX+d)
03	003	**INC BC**	INC BC	**RLC E**	NOP	LD E,RLC (IX+d)
04	004	**INC B**	INC B	**RLC H**	NOP	LD H,RLC (IX+d)
05	005	**DEC B**	DEC B	**RLC L**	NOP	LD L,RLC (IX+d)
06	006	**LD B,n**	LD B,n	**RLC (HL)**	NOP	**RLC (IX+d)**
07	007	**RLCA**	RLCA	**RLC A**	NOP	LD A,RLC (IX+d)
08	008	**EX AF,AF'**	EX AF,AF'	**RRC B**	NOP	LD B,RRC (IX+d)
09	009	**ADD HL,BC**	**ADD IX,BC**	**RRC C**	NOP	LD C,RRC (IX+d)
0a	010	**LD A,(BC)**	LD A,(BC)	**RRC D**	NOP	LD D,RRC (IX+d)
0b	011	**DEC BC**	DEC BC	**RRC E**	NOP	LD E,RRC (IX+d)
0c	012	**INC C**	INC C	**RRC H**	NOP	LD H,RRC (IX+d)
0d	013	**DEC C**	DEC C	**RRC L**	NOP	LD L,RRC (IX+d)
0e	014	**LD C,n**	LD C,n	**RRC (HL)**	NOP	**RRC (IX+d)**
0f	015	**RRCA**	RRCA	**RRC A**	NOP	LD A,RRC (IX+d)
10	016	**DJNZ d**	DJNZ d	**RL B**	NOP	LD B,RL (IX+d)
11	017	**LD DE,nn**	LD DE,nn	**RL C**	NOP	LD C,RL (IX+d)
12	018	**LD (DE),A**	LD (DE),A	**RL D**	NOP	LD D,RL (IX+d)
13	019	**INC DE**	INC DE	**RL E**	NOP	LD E,RL (IX+d)
14	020	**INC D**	INC D	**RL H**	NOP	LD H,RL (IX+d)
15	021	**DEC D**	DEC D	**RL L**	NOP	LD L,RL (IX+d)
16	022	**LD D,n**	LD D,n	**RL (HL)**	NOP	**RL (IX+d)**
17	023	**RLA**	RLA	**RL A**	NOP	LD A,RL (IX+d)
18	024	**JR d**	JR d	**RR B**	NOP	LD B,RR (IX+d)
19	025	**ADD HL,DE**	**ADD IX,DE**	**RR C**	NOP	LD C,RR (IX+d)
1a	026	**LD A,(DE)**	LD A,(DE)	**RR D**	NOP	LD D,RR (IX+d)
1b	027	**DEC DE**	DEC DE	**RR E**	NOP	LD E,RR (IX+d)
1c	028	**INC E**	INC E	**RR H**	NOP	LD H,RR (IX+d)
1d	029	**DEC E**	DEC E	**RR L**	NOP	LD L,RR (IX+d)
1e	030	**LD E,n**	LD E,n	**RR (HL)**	NOP	**RR (IX+d)**
1f	031	**RRA**	RRA	**RR A**	NOP	LD A,RR (IX+d)
20	032	**JR NZ,d**	JR NZ,d	**SLA B**	NOP	LD B,SLA (IX+d)
21	033	**LD HL,nn**	**LD IX,nn**	**SLA C**	NOP	LD C,SLA (IX+d)
22	034	**LD (nn),HL**	**LD (nn),IX**	**SLA D**	NOP	LD D,SLA (IX+d)
23	035	**INC HL**	**INC IX**	**SLA E**	NOP	LD E,SLA (IX+d)
24	036	**INC H**	INC IXh	**SLA H**	NOP	LD H,SLA (IX+d)
25	037	**DEC H**	DEC IXh	**SLA L**	NOP	LD L,SLA (IX+d)
26	038	**LD H,n**	LD IXh,n	**SLA (HL)**	NOP	**SLA (IX+d)**
27	039	**DAA**	DAA	**SLA A**	NOP	LD A,SLA (IX+d)
28	040	**JR Z,d**	JR Z,d	**SRA B**	NOP	LD B,SRA (IX+d)
29	041	**ADD HL,HL**	**ADD IX,IX**	**SRA C**	NOP	LD C,SRA (IX+d)
2a	042	**LD HL,(nn)**	**LD IX,(nn)**	**SRA D**	NOP	LD D,SRA (IX+d)
2b	043	**DEC HL**	**DEC IX**	**SRA E**	NOP	LD E,SRA (IX+d)
2c	044	**INC L**	INC IXl	**SRA H**	NOP	LD H,SRA (IX+d)
2d	045	**DEC L**	DEC IXl	**SRA L**	NOP	LD L,SRA (IX+d)
2e	046	**LD L,n**	LD IXl,n	**SRA (HL)**	NOP	**SRA (IX+d)**
2f	047	**CPL**	CPL	**SRA A**	NOP	LD A,SRA (IX+d)
30	048	**JR NC,d**	JR NC,d	SLL B	NOP	LD B,SLL (IX+d)
31	049	**LD SP,nn**	LD SP,nn	SLL C	NOP	LD C,SLL (IX+d)
32	050	**LD (nn),A**	LD (nn),A	SLL D	NOP	LD D,SLL (IX+d)
33	051	**INC SP**	INC SP	SLL E	NOP	LD E,SLL (IX+d)
34	052	**INC (HL)**	**INC (IX+d)**	SLL H	NOP	LD H,SLL (IX+d)
35	053	**DEC (HL)**	**DEC (IX+d)**	SLL L	NOP	LD L,SLL (IX+d)
36	054	**LD (HL),n**	**LD (IX+d),n**	SLL (HL)	NOP	SLL (IX+d)
37	055	**SCF**	SCF	SLL A	NOP	LD A,SLL (IX+d)
38	056	**JR C,d**	JR C,d	**SRL B**	NOP	LD B,SRL (IX+d)
39	057	**ADD HL,SP**	**ADD IX,SP**	**SRL C**	NOP	LD C,SRL (IX+d)
3a	058	**LD A,(nn)**	LD A,(nn)	**SRL D**	NOP	LD D,SRL (IX+d)
3b	059	**DEC SP**	DEC SP	**SRL E**	NOP	LD E,SRL (IX+d)
3c	060	**INC A**	INC A	**SRL H**	NOP	LD H,SRL (IX+d)
3d	061	**DEC A**	DEC A	**SRL L**	NOP	LD L,SRL (IX+d)
3e	062	**LD A,n**	LD A,n	**SRL (HL)**	NOP	**SRL (IX+d)**
3f	063	**CCF**	CCF	**SRL A**	NOP	LD A,SRL (IX+d)

Hex	Dec	Normal	DD/FD Prefix	CB Prefix	ED Prefix	DDCB/FDCB Prefix
40	064	LD B,B	LD B,B	BIT 0,B	IN B,(C)	BIT 0,(IX+d)
41	065	LD B,C	LD B,C	BIT 0,C	OUT (C),B	BIT 0,(IX+d)
42	066	LD B,D	LD B,D	BIT 0,D	SBC HL,BC	BIT 0,(IX+d)
43	067	LD B,E	LD B,E	BIT 0,E	LD (nn),BC	BIT 0,(IX+d)
44	068	LD B,H	LD B,IXh	BIT 0,H	NEG	BIT 0,(IX+d)
45	069	LD B,L	LD B,IXl	BIT 0,L	RETN	BIT 0,(IX+d)
46	070	LD B,(HL)	LD B,(IX+d)	BIT 0,(HL)	IM 0	BIT 0,(IX+d)
47	071	LD B,A	LD B,A	BIT 0,A	LD I,A	BIT 0,(IX+d)
48	072	LD C,B	LD C,B	BIT 1,B	IN C,(C)	BIT 1,(IX+d)
49	073	LD C,C	LD C,C	BIT 1,C	OUT (C),C	BIT 1,(IX+d)
4a	074	LD C,D	LD C,D	BIT 1,D	ADC HL,BC	BIT 1,(IX+d)
4b	075	LD C,E	LD C,E	BIT 1,E	LD BC,(nn)	BIT 1,(IX+d)
4c	076	LD C,H	LD C,IXh	BIT 1,H	NEG	BIT 1,(IX+d)
4d	077	LD C,L	LD C,IXl	BIT 1,L	RETI	BIT 1,(IX+d)
4e	078	LD C,(HL)	LD C,(IX+d)	BIT 1,(HL)	IM 0	BIT 1,(IX+d)
4f	079	LD C,A	LD C,A	BIT 1,A	LD R,A	BIT 1,(IX+d)
50	080	LD D,B	LD D,B	BIT 2,B	IN D,(C)	BIT 2,(IX+d)
51	081	LD D,C	LD D,C	BIT 2,C	OUT (C),D	BIT 2,(IX+d)
52	082	LD D,D	LD D,D	BIT 2,D	SBC HL,DE	BIT 2,(IX+d)
53	083	LD D,E	LD D,E	BIT 2,E	LD (nn),DE	BIT 2,(IX+d)
54	084	LD D,H	LD D,IXh	BIT 2,H	NEG	BIT 2,(IX+d)
55	085	LD D,L	LD D,IXl	BIT 2,L	RETN	BIT 2,(IX+d)
56	086	LD D,(HL)	LD D,(IX+d)	BIT 2,(HL)	IM 1	BIT 2,(IX+d)
57	087	LD D,A	LD D,A	BIT 2,A	LD A,I	BIT 2,(IX+d)
58	088	LD E,B	LD E,B	BIT 3,B	IN E,(C)	BIT 3,(IX+d)
59	089	LD E,C	LD E,C	BIT 3,C	OUT (C),E	BIT 3,(IX+d)
5a	090	LD E,D	LD E,D	BIT 3,D	ADC HL,DE	BIT 3,(IX+d)
5b	091	LD E,E	LD E,E	BIT 3,E	LD DE,(nn)	BIT 3,(IX+d)
5c	092	LD E,H	LD E,IXh	BIT 3,H	NEG	BIT 3,(IX+d)
5d	093	LD E,L	LD E,IXl	BIT 3,L	RETI	BIT 3,(IX+d)
5e	094	LD E,(HL)	LD E,(IX+d)	BIT 3,(HL)	IM 2	BIT 3,(IX+d)
5f	095	LD E,A	LD E,A	BIT 3,A	LD A,R	BIT 3,(IX+d)
60	096	LD H,B	LD IXh,B	BIT 4,B	IN H,(C)	BIT 4,(IX+d)
61	097	LD H,C	LD IXh,C	BIT 4,C	OUT (C),H	BIT 4,(IX+d)
62	098	LD H,D	LD IXh,D	BIT 4,D	SBC HL,HL	BIT 4,(IX+d)
63	099	LD H,E	LD IXh,E	BIT 4,E	LD (nn),HL	BIT 4,(IX+d)
64	100	LD H,H	LD IXh,IXh	BIT 4,H	NEG	BIT 4,(IX+d)
65	101	LD H,L	LD IXh,IXl	BIT 4,L	RETN	BIT 4,(IX+d)
66	102	LD H,(HL)	LD H,(IX+d)	BIT 4,(HL)	IM 0	BIT 4,(IX+d)
67	103	LD H,A	LD IXh,A	BIT 4,A	RRD	BIT 4,(IX+d)
68	104	LD L,B	LD IXl,B	BIT 5,B	IN L,(C)	BIT 5,(IX+d)
69	105	LD L,C	LD IXl,C	BIT 5,C	OUT (C),L	BIT 5,(IX+d)
6a	106	LD L,D	LD IXl,D	BIT 5,D	ADC HL,HL	BIT 5,(IX+d)
6b	107	LD L,E	LD IXl,E	BIT 5,E	LD HL,(nn)	BIT 5,(IX+d)
6c	108	LD L,H	LD IXl,IXh	BIT 5,H	NEG	BIT 5,(IX+d)
6d	109	LD L,L	LD IXl,IXl	BIT 5,L	RETI	BIT 5,(IX+d)
6e	110	LD L,(HL)	LD L,(IX+d)	BIT 5,(HL)	IM 0	BIT 5,(IX+d)
6f	111	LD L,A	LD IXl,A	BIT 5,A	RLD	BIT 5,(IX+d)
70	112	LD (HL),B	LD (IX+d),B	BIT 6,B	IN X,(C)	BIT 6,(IX+d)
71	113	LD (HL),C	LD (IX+d),C	BIT 6,C	OUT (C),0	BIT 6,(IX+d)
72	114	LD (HL),D	LD (IX+d),D	BIT 6,D	SBC HL,SP	BIT 6,(IX+d)
73	115	LD (HL),E	LD (IX+d),E	BIT 6,E	LD (nn),SP	BIT 6,(IX+d)
74	116	LD (HL),H	LD (IX+d),H	BIT 6,H	NEG	BIT 6,(IX+d)
75	117	LD (HL),L	LD (IX+d),L	BIT 6,L	RETN	BIT 6,(IX+d)
76	118	HALT	HALT	BIT 6,(HL)	IM 1	BIT 6,(IX+d)
77	119	LD (HL),A	LD (IX+d),A	BIT 6,A	NOP	BIT 6,(IX+d)
78	120	LD A,B	LD A,B	BIT 7,B	IN A,(C)	BIT 7,(IX+d)
79	121	LD A,C	LD A,C	BIT 7,C	OUT (C),A	BIT 7,(IX+d)
7a	122	LD A,D	LD A,D	BIT 7,D	ADC HL,SP	BIT 7,(IX+d)
7b	123	LD A,E	LD A,E	BIT 7,E	LD SP,(nn)	BIT 7,(IX+d)
7c	124	LD A,H	LD A,IXh	BIT 7,H	NEG	BIT 7,(IX+d)
7d	125	LD A,L	LD A,IXl	BIT 7,L	RETI	BIT 7,(IX+d)
7e	126	LD A,(HL)	LD A,(IX+d)	BIT 7,(HL)	IM 2	BIT 7,(IX+d)
7f	127	LD A,A	LD A,A	BIT 7,A	NOP	BIT 7,(IX+d)

Hex	Dec	Normal	DD/FD Prefix	CB Prefix	ED Prefix	DDCB/FDCB Prefix
80	128	ADD A,B	ADD A,B	RES 0,B	NOP	LD B,RES 0,(IX+d)
81	129	ADD A,C	ADD A,C	RES 0,C	NOP	LD C,RES 0,(IX+d)
82	130	ADD A,D	ADD A,D	RES 0,D	NOP	LD D,RES 0,(IX+d)
83	131	ADD A,E	ADD A,E	RES 0,E	NOP	LD E,RES 0,(IX+d)
84	132	ADD A,H	ADD A,IXh	RES 0,H	NOP	LD H,RES 0,(IX+d)
85	133	ADD A,L	ADD A,IXl	RES 0,L	NOP	LD L,RES 0,(IX+d)
86	134	ADD A,(HL)	ADD A,(IX+d)	RES 0,(HL)	NOP	RES 0,(IX+d)
87	135	ADD A,A	ADD A,A	RES 0,A	NOP	LD A,RES 0,(IX+d)
88	136	ADC A,B	ADC A,B	RES 1,B	NOP	LD B,RES 1,(IX+d)
89	137	ADC A,C	ADC A,C	RES 1,C	NOP	LD C,RES 1,(IX+d)
8a	138	ADC A,D	ADC A,D	RES 1,D	NOP	LD D,RES 1,(IX+d)
8b	139	ADC A,E	ADC A,E	RES 1,E	NOP	LD E,RES 1,(IX+d)
8c	140	ADC A,H	ADC A,IXh	RES 1,H	NOP	LD H,RES 1,(IX+d)
8d	141	ADC A,L	ADC A,IXl	RES 1,L	NOP	LD L,RES 1,(IX+d)
8e	142	ADC A,(HL)	ADC A,(IX+d)	RES 1,(HL)	NOP	RES 1,(IX+d)
8f	143	ADC A,A	ADC A,A	RES 1,A	NOP	LD A,RES 1,(IX+d)
90	144	SUB B	SUB B	RES 2,B	NOP	LD B,RES 2,(IX+d)
91	145	SUB C	SUB C	RES 2,C	NOP	LD C,RES 2,(IX+d)
92	146	SUB D	SUB D	RES 2,D	NOP	LD D,RES 2,(IX+d)
93	147	SUB E	SUB E	RES 2,E	NOP	LD E,RES 2,(IX+d)
94	148	SUB H	SUB IXh	RES 2,H	NOP	LD H,RES 2,(IX+d)
95	149	SUB L	SUB IXl	RES 2,L	NOP	LD L,RES 2,(IX+d)
96	150	SUB (HL)	SUB (IX+d)	RES 2,(HL)	NOP	RES 2,(IX+d)
97	151	SUB A	SUB A	RES 2,A	NOP	LD A,RES 2,(IX+d)
98	152	SBC A,B	SBC A,B	RES 3,B	NOP	LD B,RES 3,(IX+d)
99	153	SBC A,C	SBC A,C	RES 3,C	NOP	LD C,RES 3,(IX+d)
9a	154	SBC A,D	SBC A,D	RES 3,D	NOP	LD D,RES 3,(IX+d)
9b	155	SBC A,E	SBC A,E	RES 3,E	NOP	LD E,RES 3,(IX+d)
9c	156	SBC A,H	SBC A,IXh	RES 3,H	NOP	LD H,RES 3,(IX+d)
9d	157	SBC A,L	SBC A,IXl	RES 3,L	NOP	LD L,RES 3,(IX+d)
9e	158	SBC A,(HL)	SBC A,(IX+d)	RES 3,(HL)	NOP	RES 3,(IX+d)
9f	159	SBC A,A	SBC A,A	RES 3,A	NOP	LD A,RES 3,(IX+d)
a0	160	AND B	AND B	RES 4,B	LDI	LD B,RES 4,(IX+d)
a1	161	AND C	AND C	RES 4,C	CPI	LD C,RES 4,(IX+d)
a2	162	AND D	AND D	RES 4,D	INI	LD D,RES 4,(IX+d)
a3	163	AND E	AND E	RES 4,E	OUTI	LD E,RES 4,(IX+d)
a4	164	AND H	AND IXh	RES 4,H	NOP	LD H,RES 4,(IX+d)
a5	165	AND L	AND IXl	RES 4,L	NOP	LD L,RES 4,(IX+d)
a6	166	AND (HL)	AND (IX+d)	RES 4,(HL)	NOP	RES 4,(IX+d)
a7	167	AND A	AND A	RES 4,A	NOP	LD A,RES 4,(IX+d)
a8	168	XOR B	XOR B	RES 5,B	LDD	LD B,RES 5,(IX+d)
a9	169	XOR C	XOR C	RES 5,C	CPD	LD C,RES 5,(IX+d)
aa	170	XOR D	XOR D	RES 5,D	IND	LD D,RES 5,(IX+d)
ab	171	XOR E	XOR E	RES 5,E	OUTD	LD E,RES 5,(IX+d)
ac	172	XOR H	XOR IXh	RES 5,H	NOP	LD H,RES 5,(IX+d)
ad	173	XOR L	XOR IXl	RES 5,L	NOP	LD L,RES 5,(IX+d)
ae	174	XOR (HL)	XOR (IX+d)	RES 5,(HL)	NOP	RES 5,(IX+d)
af	175	XOR A	XOR A	RES 5,A	NOP	LD A,RES 5,(IX+d)
b0	176	OR B	OR B	RES 6,B	LDIR	LD B,RES 6,(IX+d)
b1	177	OR C	OR C	RES 6,C	CPIR	LD C,RES 6,(IX+d)
b2	178	OR D	OR D	RES 6,D	INIR	LD D,RES 6,(IX+d)
b3	179	OR E	OR E	RES 6,E	OTIR	LD E,RES 6,(IX+d)
b4	180	OR H	OR IXh	RES 6,H	NOP	LD H,RES 6,(IX+d)
b5	181	OR L	OR IXl	RES 6,L	NOP	LD L,RES 6,(IX+d)
b6	182	OR (HL)	OR (IX+d)	RES 6,(HL)	NOP	RES 6,(IX+d)
b7	183	OR A	OR A	RES 6,A	NOP	LD A,RES 6,(IX+d)
b8	184	CP B	CP B	RES 7,B	LDDR	LD B,RES 7,(IX+d)
b9	185	CP C	CP C	RES 7,C	CPDR	LD C,RES 7,(IX+d)
ba	186	CP D	CP D	RES 7,D	INDR	LD D,RES 7,(IX+d)
bb	187	CP E	CP E	RES 7,E	OTDR	LD E,RES 7,(IX+d)
bc	188	CP H	CP IXh	RES 7,H	NOP	LD H,RES 7,(IX+d)
bd	189	CP L	CP IXl	RES 7,L	NOP	LD L,RES 7,(IX+d)
be	190	CP (HL)	CP (IX+d)	RES 7,(HL)	NOP	RES 7,(IX+d)
bf	191	CP A	CP A	RES 7,A	NOP	LD A,RES 7,(IX+d)

Hex	Dec	Normal	DD/FD Prefix	CB Prefix	ED Prefix	DDCB/FDCB Prefix
c0	192	**RET NZ**	RET NZ	**SET 0,B**	NOP	LD B,SET 0,(IX+d)
c1	193	**POP BC**	POP BC	**SET 0,C**	NOP	LD C,SET 0,(IX+d)
c2	194	**JP NZ,nn**	JP NZ,nn	**SET 0,D**	NOP	LD D,SET 0,(IX+d)
c3	195	**JP nn**	JP nn	**SET 0,E**	NOP	LD E,SET 0,(IX+d)
c4	196	**CALL NZ,nn**	CALL NZ,nn	**SET 0,H**	NOP	LD H,SET 0,(IX+d)
c5	197	**PUSH BC**	PUSH BC	**SET 0,L**	NOP	LD L,SET 0,(IX+d)
c6	198	**ADD A,n**	ADD A,n	**SET 0,(HL)**	NOP	**SET 0,(IX+d)**
c7	199	**RST 0**	RST 0	**SET 0,A**	NOP	LD A,SET 0,(IX+d)
c8	100	**RET Z**	RET Z	**SET 1,B**	NOP	LD B,SET 1,(IX+d)
c9	201	**RET**	RET	**SET 1,C**	NOP	LD C,SET 1,(IX+d)
ca	202	**JP Z,nn**	JP Z,nn	**SET 1,D**	NOP	LD D,SET 1,(IX+d)
cb	203	**[See CB Prefix]**	[See DDCB/FDCB]	**SET 1,E**	NOP	LD E,SET 1,(IX+d)
cc	204	**CALL Z,nn**	CALL Z,nn	**SET 1,H**	NOP	LD H,SET 1,(IX+d)
cd	205	**CALL nn**	CALL nn	**SET 1,L**	NOP	LD L,SET 1,(IX+d)
ce	206	**ADC A,n**	ADC A,n	**SET 1,(HL)**	NOP	**SET 1,(IX+d)**
cf	207	**RST 8**	RST 8	**SET 1,A**	NOP	LD A,SET 1,(IX+d)
d0	208	**RET NC**	RET NC	**SET 2,B**	NOP	LD B,SET 2,(IX+d)
d1	209	**POP DE**	POP DE	**SET 2,C**	NOP	LD C,SET 2,(IX+d)
d2	210	**JP NC,nn**	JP NC,nn	**SET 2,D**	NOP	LD D,SET 2,(IX+d)
d3	211	**OUT (n),A**	OUT (n),A	**SET 2,E**	NOP	LD E,SET 2,(IX+d)
d4	212	**CALL NC,nn**	CALL NC,nn	**SET 2,H**	NOP	LD H,SET 2,(IX+d)
d5	213	**PUSH DE**	PUSH DE	**SET 2,L**	NOP	LD L,SET 2,(IX+d)
d6	214	**SUB n**	SUB n	**SET 2,(HL)**	NOP	**SET 2,(IX+d)**
d7	215	**RST 10H**	RST 10H	**SET 2,A**	NOP	LD A,SET 2,(IX+d)
d8	216	**RET C**	RET C	**SET 3,B**	NOP	LD B,SET 3,(IX+d)
d9	217	**EXX**	EXX	**SET 3,C**	NOP	LD C,SET 3,(IX+d)
da	218	**JP C,nn**	JP C,nn	**SET 3,D**	NOP	LD D,SET 3,(IX+d)
db	219	**IN A,(n)**	IN A,(n)	**SET 3,E**	NOP	LD E,SET 3,(IX+d)
dc	220	**CALL C,nn**	CALL C,nn	**SET 3,H**	NOP	LD H,SET 3,(IX+d)
dd	221	**[IX Prefix]**	[IX Prefix]	**SET 3,L**	NOP	LD L,SET 3,(IX+d)
de	222	**SBC A,n**	SBC A,n	**SET 3,(HL)**	NOP	**SET 3,(IX+d)**
df	223	**RST 18H**	RST 18H	**SET 3,A**	NOP	LD A,SET 3,(IX+d)
e0	224	**RET PO**	RET PO	**SET 4,B**	NOP	LD B,SET 4,(IX+d)
e1	225	**POP HL**	**POP IX**	**SET 4,C**	NOP	LD C,SET 4,(IX+d)
e2	226	**JP PO,nn**	JP PO,nn	**SET 4,D**	NOP	LD D,SET 4,(IX+d)
e3	227	**EX (SP),HL**	**EX (SP),IX**	**SET 4,E**	NOP	LD E,SET 4,(IX+d)
e4	228	**CALL PO,nn**	CALL PO,nn	**SET 4,H**	NOP	LD H,SET 4,(IX+d)
e5	229	**PUSH HL**	**PUSH IX**	**SET 4,L**	NOP	LD L,SET 4,(IX+d)
e6	230	**AND n**	AND n	**SET 4,(HL)**	NOP	**SET 4,(IX+d)**
e7	231	**RST 20H**	RST 20H	**SET 4,A**	NOP	LD A,SET 4,(IX+d)
e8	232	**RET PE**	RET PE	**SET 5,B**	NOP	LD B,SET 5,(IX+d)
e9	233	**JP (HL)**	**JP (IX)**	**SET 5,C**	NOP	LD C,SET 5,(IX+d)
ea	234	**JP PE,nn**	JP PE,nn	**SET 5,D**	NOP	LD D,SET 5,(IX+d)
eb	235	**EX DE,HL**	EX DE,HL	**SET 5,E**	NOP	LD E,SET 5,(IX+d)
ec	236	**CALL PE,nn**	CALL PE,nn	**SET 5,H**	NOP	LD H,SET 5,(IX+d)
ed	237	**[See ED Prefix]**	[Ignore DD/FD]	**SET 5,L**	NOP	LD L,SET 5,(IX+d)
ee	238	**XOR n**	XOR n	**SET 5,(HL)**	NOP	**SET 5,(IX+d)**
ef	239	**RST 28H**	RST 28H	**SET 5,A**	NOP	LD A,SET 5,(IX+d)
f0	240	**RET P**	RET P	**SET 6,B**	NOP	LD B,SET 6,(IX+d)
f1	241	**POP AF**	POP AF	**SET 6,C**	NOP	LD C,SET 6,(IX+d)
f2	242	**JP P,nn**	JP P,nn	**SET 6,D**	NOP	LD D,SET 6,(IX+d)
f3	243	**DI**	DI	**SET 6,E**	NOP	LD E,SET 6,(IX+d)
f4	244	**CALL P,nn**	CALL P,nn	**SET 6,H**	NOP	LD H,SET 6,(IX+d)
f5	245	**PUSH AF**	PUSH AF	**SET 6,L**	NOP	LD L,SET 6,(IX+d)
f6	246	**OR n**	OR n	**SET 6,(HL)**	NOP	**SET 6,(IX+d)**
f7	247	**RST 30H**	RST 30H	**SET 6,A**	NOP	LD A,SET 6,(IX+d)
f8	248	**RET M**	RET M	**SET 7,B**	NOP	LD B,SET 7,(IX+d)
f9	249	**LD SP,HL**	**LD SP,IX**	**SET 7,C**	NOP	LD C,SET 7,(IX+d)
fa	250	**JP M,nn**	JP M,nn	**SET 7,D**	NOP	LD D,SET 7,(IX+d)
fb	251	**EI**	EI	**SET 7,E**	NOP	LD E,SET 7,(IX+d)
fc	252	**CALL M,nn**	CALL M,nn	**SET 7,H**	NOP	LD H,SET 7,(IX+d)
fd	253	**[IY Prefix]**	[IY Prefix]	**SET 7,L**	NOP	LD L,SET 7,(IX+d)
fe	254	**CP n**	CP n	**SET 7,(HL)**	NOP	**SET 7,(IX+d)**
ff	255	**RST 38H**	RST 38H	**SET 7,A**	NOP	LD A,SET 7,(IX+d)