3. INSTRUCTION

All instructions of the μPD77C25 are one word instruction; one instruction is composed of 24 bits. The instructions are divided into the following four types.

OP instruction : This instruction is used to perform operations such as ordinary arithmetic operations and data transfer.

RT instruction : This instruction is return instruction.

JP instruction : This instruction causes unconditional or conditional jump of program execution, or calls a subroutine.

LD instruction : This loads 16-bit immediate data to the specified register.

3.1 OP Instruction

<table>
<tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">00</td><td colspan="2">P-SELECT</td><td colspan="4">ALU</td><td>ASL</td><td colspan="2">DP_L</td><td colspan="4">DP_H·M</td><td>RPDCR</td><td colspan="4">SRC</td><td colspan="4">DST</td></tr>
</table>

OP instruction has the following functions.

(i) This instruction consists of six fields and two bits.

(ii) Program Counter is incremented to the address hold at that time.

(1) P-SELECT Field

This field selects P input of ALU.

(2) ALU Field

This field specifies ALU operation.

(3) ASL (Acc Selection) bit

This bit selects Q input of ALU and specifies AccA or AccB.

(4) DP_L Field

This field specifies operation of lower 4 bits of DP (Data Pointer). The new value of DP_L specified by this field becomes valid from the next instruction.

(5) DP_H•M (DP_H Modify) Field

This field qualifies changes value of the upper 4 bits of DP (Data Pointer). The value of this field is qualified exclusively ORed with the value of the upper 4 bits in DP (Data Pointer). The qualified value of DP_H valid from the next instruction.

(6) RPDCR (RP Decrement) bit

This bit specifies decrement operation of RP (ROM Pointer). The decremented value becomes valid from the next instruction.

(7) SRC (Source) Field

This field specifies source register to the internal data bus for transfer instruction.

(8) DST (Destination) Field

This field specifies the destination register for a transfer instruction. The data specified in SRC Field (output to the internal data bus) is written to the specified register.

3.2 RT Instruction

<table>
<tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">01</td><td colspan="2">P-SELECT</td><td colspan="4">ALU</td><td>ASL</td><td colspan="2">DP_L</td><td colspan="4">DP_H·M</td><td>RPDCR</td><td colspan="4">SRC</td><td colspan="4">DST</td></tr>
</table>

RT instruction is a return instruction and has the following functions.

(1) This instruction consists of six fields and two bits like the OP instruction.

(2) It restores the return address saved to the stack to PC after performing the same operation as the OP instruction.

3.3 JP Instruction

<table>
<tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">10</td><td colspan="9">BRCH</td><td colspan="11">NA</td><td colspan="2"></td></tr>
</table>

JP instruction causes program execution of jump unconditionally or conditionally. It also calls a subroutine.

(1) BRCH (Branch) Field

This field specifies type of jump instruction and conditional jump instruction.

(2) NA (Next Address) Field

This field specifies jump address.

3.4 LD Instruction

<table>
<tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">11</td><td colspan="16">ID</td><td colspan="2"></td><td colspan="4">DST</td></tr>
</table>

This is an immediate data load instruction which loads 16-bit data to the specified register.

(1) ID (Immediate Data) Field

This field specifies the immediate 16-bit data. The immediate data is loaded to the register specified in DST field.

(2) DST (Destination) Field

This field specifies the register that the data specified by ID field is loaded to. This field is the same as DST field in OP instruction.

3.5 Instruction Code

THE LIST OF INSTRUCTION CODE

<table>
<tr><th>Bit
Instruction</th><th>D23</th><th>D22</th><th>D21</th><th>D20</th><th>D19</th><th>D18</th><th>D17</th><th>D16</th><th>D15</th><th>D14</th><th>D13</th><th>D12</th><th>D11</th><th>D10</th><th>D9</th><th>D8</th><th>D7</th><th>D6</th><th>D5</th><th>D4</th><th>D3</th><th>D2</th><th>D1</th><th>D0</th></tr>
<tr><td>OP</td><td colspan="2">0 0</td><td colspan="2">P-SELECT</td><td colspan="4">ALU</td><td>ASL</td><td colspan="2">DP$_L$</td><td colspan="4">DP$_H$•M</td><td>RPDCR</td><td colspan="4">SRC</td><td colspan="4">DST</td></tr>
<tr><td>RT</td><td colspan="2">0 1</td><td colspan="22">Same as OP instruction</td></tr>
<tr><td>JP</td><td colspan="2">1 0</td><td colspan="9">BRCH</td><td colspan="11">NA</td><td colspan="2"></td></tr>
<tr><td>LD</td><td colspan="2">1 1</td><td colspan="16">ID</td><td colspan="2"></td><td colspan="4">DST</td></tr>
</table>

ALL INSTRUCTIONS

Instruction	OP field		Contents of instruction
	D$_{23}$	D$_{22}$	
OP	0	0	Arithmetic and Transfer Operation
RT	0	1	Return Instruction
JP	1	0	Jump Instruction
LD	1	1	Immediate Data Load Instruction

OP, RT INSTRUCTIONS

Mnemonic	P-SELECT field		Input Data
	D$_{21}$	D$_{20}$	
RAM	0	0	RAM
IDB	0	1	Internal Data Bus
M	1	0	M register
N	1	1	N register

OP, RT INSTRUCTIONS

Mnemonic	ALU field				Function	
	D_{19}	D_{18}	D_{17}	D_{16}		
NOP	0	0	0	0	No Operation	
OR	0	0	0	1	OR	(Acc) ← (Acc) ∨ (P)
AND	0	0	1	0	AND	(Acc) ← (Acc) ∧ (P)
XOR	0	0	1	1	Exclusive OR	(Acc) ← (Acc) ⊻ (P)
SUB	0	1	0	0	Subtract	(Acc) ← (Acc) – (P)
ADD	0	1	0	1	Add	(Acc) ← (Acc) + (P)
SBB	0	1	1	0	Subtract with Borrow	(Acc) ← (Acc) – (P) – (C)
ADC	0	1	1	1	Add with Carry	(Acc) ← (Acc) + (P) + (C)
DEC	1	0	0	0	Decrement Acc	(Acc) ← (Acc) – 1
INC	1	0	0	1	Increment Acc	(Acc) ← (Acc) + 1
CMP	1	0	1	0	Complement Acc (1's complement)	(Acc) ← ($\overline{\text{Acc}}$)
SHR1	1	0	1	1	1-bit R-Shift	
SHL1	1	1	0	0	1-bit L-Shift	
SHL2	1	1	0	1	2-bit L-Shift	
SHL4	1	1	1	0	4-bit L-Shift	
XCHG	1	1	1	1	8-bit Exchange	

OP, RT INSTRUCTIONS

Mnemonic	ASL bit	Selection for Acc and FLAG
	D_{15}	
ACCA	0	Acc A
ACCB	1	Acc B

OP, RT INSTRUCTIONS

Mnemonic	DP_L field		Operation
	D_{14}	D_{13}	
DPNOP	0	0	No Operation
DPINC	0	1	Increment DP_L
DPDEC	1	0	Decrement DP_L
DPCLR	1	1	Clear DP_L

OP, RT INSTRUCTIONS

Mnemonic	DP$_H$•M field				Exclusive OR
	D_{12}	D_{11}	D_{10}	D_9	
M0	0	0	0	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 0\ 0\ 0)$
M1	0	0	0	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 0\ 0\ 1)$
M2	0	0	1	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 0\ 1\ 0)$
M3	0	0	1	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 0\ 1\ 1)$
M4	0	1	0	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 1\ 0\ 0)$
M5	0	1	0	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 1\ 0\ 1)$
M6	0	1	1	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 1\ 1\ 0)$
M7	0	1	1	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (0\ 1\ 1\ 1)$
M8	1	0	0	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 0\ 0\ 0)$
M9	1	0	0	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 0\ 0\ 1)$
MA	1	0	1	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 0\ 1\ 0)$
MB	1	0	1	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 0\ 1\ 1)$
MC	1	1	0	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 1\ 0\ 0)$
MD	1	1	0	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 1\ 0\ 1)$
ME	1	1	1	0	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 1\ 1\ 0)$
MF	1	1	1	1	$(DP_7\ DP_6\ DP_5\ DP_4) \veebar (1\ 1\ 1\ 1)$

OP, RT INSTRUCTIONS

Mnemonic	RPDCR bit	Operation
	D_8	
RPNOP	0	No Operation
RPDEC	1	Decrement RP

OP, RT INSTRUCTIONS

Mnemonic	SRC field				Specified register
	D_7	D_6	D_5	D_4	
NON[Note 1], TRB	0	0	0	0	TRB register
A	0	0	0	1	Acc A register
B	0	0	1	0	Acc B register
TR	0	0	1	1	TR register
DP	0	1	0	0	DP register
RP	0	1	0	1	RP register
RO	0	1	1	0	RO register
SGN	0	1	1	1	SGN register
DR	1	0	0	0	DR register
DRNF	1	0	0	1	DR register[Note 2]
SR	1	0	1	0	SR register
SIM	1	0	1	1	SI register (1st → MSB)[Note 3]
SIL	1	1	0	0	SI register (1st → LSB)[Note 4]
K	1	1	0	1	K register
L	1	1	1	0	L register
MEM	1	1	1	1	RAM

Notes **1.** The contents of the TRB register are also output if NON is specified.

2. Although the contents of the DR register are output to the internal data bus, the RQM flag is not set. Neither is the DRQ flag in DMA mode.

3. With the 16-bit data, the serial data input first is output to the MSB of the internal data bus; the data input last is output to the LSB.

4. With the 16-bit data, the serial data input first is output to the LSB of the internal data bus; the data input last is output to the MSB.

OP, RT, LD INSTRUCTIONS

Mnemonic	DST field				Specified register
	D_3	D_2	D_1	D_0	
@NON	0	0	0	0	No specified register
@A	0	0	0	1	Acc A register
@B	0	0	1	0	Acc B register
@TR	0	0	1	1	TR register
@DP	0	1	0	0	DP register
@RP	0	1	0	1	RP register
@DR	0	1	1	0	DR register
@SR	0	1	1	1	SR register
@SOL	1	0	0	0	SO register (LSB → 1st)Note 1
@SOM	1	0	0	1	SO register (MSB → 1st)Note 2
@K	1	0	1	0	K register
@KLR	1	0	1	1	KLRNote 3
@KLM	1	1	0	0	KLMNote 4
@L	1	1	0	1	L register
@TRB	1	1	1	0	TRB register
@MEM	1	1	1	1	RAM

Notes **1.** With 16-bit data, the serial output is sequentially performed from the LSB bit of the internal data bus.

2. With 16-bit data, the serial output is sequentially performed from the MSB of the internal data bus.

3. The data on the internal bus and the output from the RO register (ROM) are set to the K and L registers, respectively.

4. The data on the internal bus and the contents of RAM (DP_7, "1", DP_5, DP_4, DP_3, DP_2, DP_1, and DP_0) specified by DP_6 = ("1") are set to the L and K registers, respectively.

Remark Following combination are prohibited in OP or RT instruction

DST field = @KLR, SRC field = K or L register
DST field = @KLM, SRC field = K or L register
DST field and SRC field specify the same register
P-SELECT field = RAM, DST field = @MEM (for ALU operation)

JP INSTRUCTION

Mnemonic	BRCH field									Condition
	D_{21}	D_{20}	D_{19}	D_{18}	D_{17}	D_{16}	D_{15}	D_{14}	D_{13}	
JMP	1	0	0	0	0	0	0	0	0	Unconditional jump
CALL	1	0	1	0	0	0	0	0	0	Unconditional jump
JNCA	0	1	0	0	0	0	0	0	0	CA = 0
JCA	0	1	0	0	0	0	0	1	0	CA = 1
JNCB	0	1	0	0	0	0	1	0	0	CB = 0
JCB	0	1	0	0	0	0	1	1	0	CB = 1
JNZA	0	1	0	0	0	1	0	0	0	ZA = 0
JZA	0	1	0	0	0	1	0	1	0	ZA = 1
JNZB	0	1	0	0	0	1	1	0	0	ZB = 0
JZB	0	1	0	0	0	1	1	1	0	ZB = 1
JNOVA0	0	1	0	0	1	0	0	0	0	OVA0 = 0
JOVA0	0	1	0	0	1	0	0	1	0	OVA0 = 1
JNOVB0	0	1	0	0	1	0	1	0	0	OVB0 = 0
JOVB0	0	1	0	0	1	0	1	1	0	OVB0 = 1
JNOVA1	0	1	0	0	1	1	0	0	0	OVA1 = 0
JOVA1	0	1	0	0	1	1	0	1	0	OVA1 = 1
JNOVB1	0	1	0	0	1	1	1	0	0	OVB1 = 0
JOVB1	0	1	0	0	1	1	1	1	0	OVB1 = 1
JNSA0	0	1	0	1	0	0	0	0	0	SA0 = 0
JSA0	0	1	0	1	0	0	0	1	0	SA0 = 1
JNSB0	0	1	0	1	0	0	1	0	0	SB0 = 0
JSB0	0	1	0	1	0	0	1	1	0	SB0 = 1
JNSA1	0	1	0	1	0	1	0	0	0	SA1 = 0
JSA1	0	1	0	1	0	1	0	1	0	SA1 = 1
JNSB1	0	1	0	1	0	1	1	0	0	SB1 = 0
JSB1	0	1	0	1	0	1	1	1	0	SB1 = 1
JDPL0	0	1	0	1	1	0	0	0	0	$DP_L = 0$
JDPLN0	0	1	0	1	1	0	0	0	1	$DP_L \neq 0$
JDPLF	0	1	0	1	1	0	0	1	0	DP_L = F (HEX)
JDPLNF	0	1	0	1	1	0	0	1	1	$DP_L \neq$ F (HEX)
JNSIAK	0	1	0	1	1	0	1	0	0	SI ACK = 0
JSIAK	0	1	0	1	1	0	1	1	0	SI ACK = 1
JNSOAK	0	1	0	1	1	1	0	0	0	SO ACK = 0
JSOAK	0	1	0	1	1	1	0	1	0	SO ACK = 1
JNRQM	0	1	0	1	1	1	1	0	0	RQM = 0
JRQM	0	1	0	1	1	1	1	1	0	RQM = 1

JP INSTRUCTION

NA field											Jump address
D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	
0	0	0	0	0	0	0	0	0	0	0	Specify address 0 as jump address
0	0	0	0	0	0	0	0	0	0	1	Specify address 1 as jump address
0	0	0	0	0	0	0	0	0	1	0	Specify address 2 as jump address
to											to
1	1	1	1	1	1	1	1	1	1	1	Specify address 2047 as jump address

LD INSTRUCTION

ID field																HEX
D_{21}	D_{20}	D_{19}	D_{18}	D_{17}	D_{16}	D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0 0 0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 0 0 1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0 0 0 2
to																to
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	F F F F

EFFECT OF ALU OPERATION ON FLAGS

	Selected FLAG						Nonselected FLAG					
	S1	S0	C	Z	OV1	OV0	S1	S0	C	Z	OV1	OV0
NOP	●	●	●	●	●	●	●	●	●	●	●	●
OR	X	↕	0	↕	0	0	●	●	●	●	●	●
AND	X	↕	0	↕	0	0	●	●	●	●	●	●
XOR	X	↕	0	↕	0	0	●	●	●	●	●	●
SUB	↕	↕	↕	↕	↕	↕	●	●	●	●	●	●
ADD	↕	↕	↕	↕	↕	↕	●	●	●	●	●	●
SBB	↕	↕	↕	↕	↕	↕	●	●	←	●	●	●
ADC	↕	↕	↕	↕	↕	↕	●	●	←	●	●	●
DEC	↕	↕	↕	↕	↕	↕	●	●	●	●	●	●
INC	↕	↕	↕	↕	↕	↕	●	●	●	●	●	●
CMP	X	↕	0	↕	0	0	●	●	●	●	●	●
SHR1	X	↕	↕	↕	0	0	●	●	●	●	●	●
SHL1	X	↕	↕	↕	0	0	●	●	←	●	●	●
SHL2	X	↕	0	↕	0	0	●	●	●	●	●	●
SHL4	X	↕	0	↕	0	0	●	●	●	●	●	●
XCHG	X	↕	0	↕	0	0	●	●	●	●	●	●

←	:	Affects the result of operation
↕	:	Affected by the result of operation
0	:	Cleared to 0
1	:	Set to 1
●	:	Retains the previous state
X	:	Undefined