

**SKIP GROUP**

The intent of any skip instruction can be reversed by making Bit 5 equal to ONE.

**SHIFT ROTATE GROUP**

Shift is an arithmetic operation. The sign bit is left unchanged and vacated bits are filled with the sign. Rotate is a logical operation and cycles the bits (including sign) in a closed ring. The number of steps is the number of ONE's in bits 9-17 of the instruction (9 max).

**OPERATE GROUP**

This is a micro program set of instructions. Thus cla V cll V cll = 764207 (5 microseconds).

**CHECK STATUS**

The instruction, cks (720033), checks the status of various in-out devices and sets IO bits 0 through 6 for subsequent program interrogation as follows:

**IN-OUT TRANSFER GROUP (BASIC LIST)**

The number of variations in this group may be greatly increased for optional or special in-out equipment.

**IN-OUT TRANSFER GROUP (BASIC LIST)**

The number of variations in this group may be greatly increased for optional or special in-out equipment.

\*If the instruction part is 73 instead of 72, the computer will wait for completion.

| Mnemonic Code | Octal Code | Operation  | Oper. Time (usec.) |
|---------------|------------|--|--------------------|
| add Y         | 40         | Add C(Y) to C(AC)                                | 10                 |
| and Y         | 02         | Logical AND of C(Y) with C(AC)                   | 10                 |
| cal           | 16         | Equals jda 100                                   | 10                 |
| dac Y         | 24         | Deposit C(AC) in Y                               | 10                 |
| dap Y         | 26         | Deposit contents of address part of AC in Y      | 10                 |
| dio Y         | 32         | Deposit C(IO) in Y                               | 10                 |
| dip Y         | 30         | Deposit instruction part of AC in Y              | 10                 |
| div Y         | 56         | Divide   | 40 max             |
| dzm Y         | 34         | Make C(Y) zero                                   | 10                 |
| idx Y         | 44         | Index (add one to) C(Y) Leave in Y & AC          | 10                 |
| ior Y         | 04         | Inclusive OR of C(Y) with C(AC)                  | 10                 |
| iot           | 72         | See In-Out Transfer Group                        | —                  |
| isp Y         | 46         | Index and skip if result is positive             | 10                 |
| jda Y         | 17         | Equals dac Y plus jsp Y + 1                      | 10                 |
| jfd Y         | 12         | Jump memory field according to C(Y)              | 10                 |
| jmp Y         | 60         | Take next instruction from Y                     | 5                  |
| jsp Y         | 62         | Jump to Y and save Program Counter in AC         | 5                  |
| lac Y         | 20         | Load AC with C(Y)                                | 10                 |
| law N         | 70         | Load AC with the number N                        | 5                  |
| law - N       | 71         | Load AC with the number - N                      | 5                  |
| lio Y         | 22         | Load IO with C(Y)                                | 10                 |
| mul Y         | 54         | Multiply   | 25 max             |
| opr           | 76         | See Operate Group                                | 5                  |
| sad Y         | 50         | Skip next instruction if C(AC) differs from C(Y) | 10                 |
| sas Y         | 52         | Skip next instruction if C(AC) is same as C(Y)   | 10                 |
| shift         |            | See Shift Group                                  | 5                  |
| skp           | 64         | See Skip Group                                   | 5                  |
| sub Y         | 42         | Subtract C(Y) from C(AC)                         | 10                 |
| xct Y         | 10         | Perform instruction in Y                         | +5                 |
| xor Y         | 06         | Exclusive OR of C(AC) with C(Y)                  | 10                 |

| Character | FIO-DEC Code | Concise Code | Character  | FIO-DEC Code | Concise Code |
|-----------|--------------|--------------|------------|--------------|--------------|
| A         | 61           | 61           | 0          | 20           | 20           |
| B         | 62           | 62           | 1          | 01           | 01           |
| C         | 263          | 63           | 2          | 02           | 02           |
| D         | 64           | 64           | 3          | 203          | 03           |
| E         | 265          | 65           | 4          | 04           | 04           |
| F         | 266          | 66           | 5          | 205          | 05           |
| G         | 67           | 67           | 6          | 206          | 06           |
| H         | 70           | 70           | 7          | 07           | 07           |
| I         | 271          | 71           | 8          | 10           | 10           |
| J         | 241          | 41           | 9          | 211          | 11           |
| K         | 242          | 42           | (          | 57           | 57           |
| L         | 43           | 43           | )          | 255          | 55           |
| M         | 244          | 44           | [          | 256          | 56           |
| N         | 45           | 45           | +          | 54           | 54           |
| O         | 46           | 46           | =          | 40           | 40           |
| P         | 247          | 47           | -          | 233          | 33           |
| Q         | 250          | 50           | /          | 73           | 73           |
| R         | 51           | 51           | ?          | 221          | 21           |
| S         | 222          | 22           | Lower Case | 272          | 72           |
| T         | 23           | 23           | Upper Case | 274          | 74           |
| U         | 224          | 24           | Space      | 200          | 00           |
| V         | 25           | 25           | Bk. Sp.    | 75           | 75           |
| W         | 26           | 26           | Tab.       | 236          | 36           |
| X         | 227          | 27           | Carr. Ret. | 277          | 77           |
| Y         | 230          | 30           | Tape Feed  | 00           | 00           |
| Z         | 31           | 31           | Red*       | —            | 35           |
|           |              |              | Blk**      | —            | 34           |
|           |              |              | Stop Code  | 13           | —            |
|           |              |              | Delete     | 100          | —            |

\*Used on type-out only, not on keyboard.

**SHIFT ROTATE GROUP**

**Status Register Definitions**

IO Bit Positions

Set to 1 when light pulse strikes Light Pen

Set to 0 at the start of each dpy instruction

Set to 1 when Perforated Tape Reader Buffer has information ready to be transferred to IO Register

Set to 0 by the Reader return pulse or by the rrb instruction

Set to 1 when Typewriter is free to receive a typ instruction

Set to 0 at the start of each typ instruction

Set to 1 when Typewriter key is stuck

Set to 0 by completion of tyi instruction

Set to 1 when Tape Punch is free to receive a ppa or ppb instruction

Set to 0 at the start of each ppa or ppb instruction

Set to 1 when Type 23 Drum address equals address specified by dba instruction

Set to 0 by the dcc instruction

Set to 1 on entering Sequence Break Mode

**CHECK STATUS**

The instruction, cks (720033), checks the status of various in-out devices and sets IO bits 0 through 6 for subsequent program interrogation as follows:

**Status Register Definitions**

IO Bit Positions

Set to 1 when light pulse strikes Light Pen

Set to 0 at the start of each dpy instruction

Set to 1 when Perforated Tape Reader Buffer has information ready to be transferred to IO Register

Set to 0 by the Reader return pulse or by the rrb instruction

Set to 1 when Typewriter is free to receive a typ instruction

Set to 0 at the start of each typ instruction

Set to 1 when Typewriter key is stuck

Set to 0 by completion of tyi instruction

Set to 1 when Tape Punch is free to receive a ppa or ppb instruction

Set to 0 at the start of each ppa or ppb instruction

Set to 1 when Type 23 Drum address equals address specified by dba instruction

Set to 0 by the dcc instruction

Set to 1 on entering Sequence Break Mode

**IN-OUT TRANSFER GROUP (BASIC LIST)**

The number of variations in this group may be greatly increased for optional or special in-out equipment.

**IN-OUT TRANSFER GROUP (BASIC LIST)**

The number of variations in this group may be greatly increased for optional or special in-out equipment.

\*If the instruction part is 73 instead of 72, the computer will wait for completion.

**digital EQUIPMENT CORPORATION**  
MAYNARD, MASSACHUSETTS

**PDP 1 INSTRUCTION LIST**

| Mnemonic Code | Octal Code | Operation                                   | Oper. Time (usec.) |
|---------------|------------|---|--------------------|
| add Y         | 40         | Add C(Y) to C(AC)                           | 10                 |
| and Y         | 02         | Logical AND of C(Y) with C(AC)              | 10                 |
| cal           | 16         | Equals jda 100                              | 10                 |
| dac Y         | 24         | Deposit C(AC) in Y                          | 10                 |
| dap Y         | 26         | Deposit contents of address part of AC in Y | 10                 |
| dio Y         | 32         | Deposit C(IO) in Y                          | 10                 |
| dip Y         | 30         | Deposit instruction part of AC in Y         | 10                 |
| div Y         | 56         | Divide                                      | 40 max             |
| dzm Y         | 34         | Make C(Y) zero                              | 10                 |
| idx Y         | 44         | Index (add one to) C(Y) Leave in Y & AC     | 10                 |
| ior Y         | 04         | Inclusive OR of C(Y) with C(AC)             | 10                 |
| iot           | 72         | See In-Out Transfer Group                   | —                  |