1	Introduction
2	Basic Program Instructions
3	STL Programming
4	Devices in Detail
5	Applied Instructions
6	Diagnostic Devices
7	Instruction Execution Times
8	PLC Device Tables
9	Assigning System Devices
10	Points of Technique
11	Index

Chapter Contents

6. Diagnostic Devices 6-1
6.1 Device Lists 6-2
6.2 PLC Status (M8000 to M8009 and D8000 to D8009)........................ 6-7
6.3 Clock Devices (M8010 to M8019 and D8010 to D8019) 6-8
6.4 Operation Flags 6-9
6.5 PLC Operation Mode (M8030 to M8039 and D8030 to D8039) 6-10
6.6 Step Ladder (STL) Flags (M8040 to M8049 and D8040 to D8049) 6-11
6.7 Interrupt Control Flags (M8050 to M8059 and D8050 to D8059) 6-12
6.8 Error Detection Devices (M8060 to M8069 and D8060 to D6069) 6-13
6.9 Link And Special Operation Devices (M8070 to M8099 and D8070 to D8099) 6-14
6.10 Miscellaneous Devices (M8100 to M8119 and D8100 to D8119) 6-16
6.11 Communication Adapter Devices, i.e. 232ADP, 485ADP (M8120 to M8129 and D8120 to D8129)........ 6-16
6.12 High Speed Zone Compare Table Comparison Flags (M8130 to M8148 and D8130 to D8148) 6-17
6.13 Miscellaneous Devices (M8160 to M8199) 6-19
6.14 Miscellanious Devices (D8158 to D8164) and Index Registers (D8182 to D8199) 6-20
6.15 N:N Network Related Flags and Data Registers 6-21
6.16 Up/Down Counter Control (M8200 to M8234 and M8219 to D8234) 6-22
6.17 High Speed Counter Control (M8235 to M8255 and D8235 to D8255) 6-22
6.18 Error Code Tables 6-23

6. Diagnostic Devices

FX1S FX1N FX2N FX2NC

The following special devices are used by the PLC to highlight the current operational status and identify any faults or errors that may be occurring. There are some variations in the application of these devices to members of the FX PLC family, these are noted where appropriate.
The Internal diagnostic devices consist of both auxiliary (M) coils and data (D) registers.
Often there is a correlation between both M and D diagnostic devices for example M8039 identifies that the PLC is in constant scan mode but D8039 contains the value or length of the set constant scan.

Devices unable to be set by user:

Any device of type M or D that is marked with a “(✗)” cannot be set by a users program. In the case of M devices this means the associated coil cannot be driven BUT all contacts can be read. For data devices (D) new values cannot be written to the register by a user BUT the register contents can be used in a data comparison.

Default Resetting Devices:

- Certain devices reset to their default status when the PLC is turned from OFF to ON. These are identified by the following symbol “(✍)”.

Symbol summary:

- ✗ not able to be set by user
- ✍ automatically reset to default at power ON.
- ✍R Also reset to default when CPU is switched to RUN.
- ✍S Also reset to default when CPU is switched to STOP.

6.1 Device Lists

Device	FX1S	FX1N	FX2N	FX2NC
M8000	✩	✩	✩	✩
M8001	✩	✩	✩	✩
M8002	✩	✩	✩	✩
M8003	✩	✩	✩	✩
M8004	✩	✩	✩	✩
M8005	-	-	✩	✩
M8006	-	-	✩	✩
M8007	-	-	✩	✩
M8008	-	-	✩	✩
M8009	-	-	✩	✩
M8010	Reserved			
M8011	✩	✩	✩	✩
M8012	✩	✩	✩	✩
M8013	✩	✩	✩	✩
M8014	✩	✩	✩	✩
M8015	✩	✩	✩	✩
M8016	✩	✩	✩	✩
M8017	✩	✩	✩	✩
M8018	✩	✩	✩	✩
M8019	✩	✩	✩	✩
M8020	✩	✩	✩	✩
M8021	✩	✩	✩	✩
M8022	✩	✩	✩	✩
M8023	Reserved			
M8024	-	-	✩	✩
M8025	-	-	✩	✩
M8026	-	-	✩	✩
M8027	-	-	✩	✩
M8028	(✩) *1	-	✩	✩
M8029	✩	✩	✩	✩
M8030	-	-	✩	✩
M8031	✩	✩	✩	✩
M8032	✩	✩	✩	✩
M8033	✩	✩	✩	✩
M8034	✩	✩	✩	✩
M8035	✩	✩	✩	✩
M8036	✩	✩	✩	✩
M8037	✩	✩	✩	✩
M8038	✩	✩	✩	✩
M8039	✩	✩	✩	✩
M8040	✩	✩	✩	✩
M8041	✩	✩	✩	✩
M8042	✩	✩	✩	✩
M8043	✩	✩	✩	✩
M8044	✩	✩	✩	✩
M8045	✩	✩	✩	✩
M8046	✩	✩	✩	✩
M8047	✩	✩	✩	✩
M8048	-	-	✩	✩
M8049	-	-	✩	✩

Device	FX1S	FX1N	FX2N	FX2NC
D8000	✩	✩	✩	✩
D8001	✩	✩	✩	✩
D8002	✩	✩	✩	✩
D8003	✩	✩	✩	✩
D8004	✩	✩	✩	✩
D8005	-	-	✩	✩
D8006	-	-	✩	✩
D8007	-	-	✩	✩
D8008	-	-	✩	✩
D8009	-	-	✩	✩
D8010	✩	✩	✩	✩
D8011	✩	✩	✩	✩
D8012	✩	✩	✩	✩
D8013	✩	✩	✩	✩
D8014	✩	✩	✩	✩
D8015	✩	✩	✩	✩
D8016	✩	✩	✩	✩
D8017	✩	✩	✩	✩
D8018	✩	✩	✩	✩
D8019	✩	✩	✩	✩
D8020	✩	✩	✩	✩
D8021	Reserved			
D8022				
D8023				
D8024				
D8025				
D8026				
D8027				
D8028	✩	✩	✩	✩
D8029	✩	✩	✩	✩
D8030	✩	✩	-	-
D8031	✩	✩	-	-
D8032	Reserved			
D8033				
D8034				
D8035				
D8036				
D8037				
D8038				
D8039	✩	✩	✩	✩
D8040	✩	✩	✩	✩
D8041	✩	✩	✩	✩
D8042	✩	✩	✩	✩
D8043	✩	✩	✩	✩
D8044	✩	✩	✩	✩
D8045	✩	✩	✩	✩
D8046	✩	✩	✩	✩
D8047	✩	✩	✩	✩
D8048	Reserved			
D8049	-	-	✩	✩

Note *1: M8028 offers a different functionality for FX1S than it does for FX2N and FX2NC. See page 6-9 for details

Device	FX1S	FX1N	FX2N	FX2NC
M8050	✩	✩	✩	✩
M8051	✩	✩	✩	✩
M8052	✩	✩	✩	✩
M8053	✩	✩	✩	✩
M8054	✩	✩	✩	✩
M8055	✩	✩	✩	✩
M8056	-	-	✩	✩
M8057	-	-	✩	✩
M8058	-	-	✩	✩
M8059	-	-	✩	✩
M8060	-	-	✩	✩
M8061	✩	✩	✩	✩
M8062	-	-	✩	✩
M8063	✩	✩	✩	✩
M8064	✩	✩	✩	✩
M8065	✩	✩	✩	✩
M8066	✩	✩	✩	✩
M8067	✩	✩	✩	✩
M8068	✩	✩	✩	✩
M8069	-	-	✩	✩
M8070	✩	✩	✩	✩
M8071	✩	✩	✩	✩
M8072	✩	✩	✩	✩
M8073	✩	✩	✩	✩
M8074	Reserved			
M8075	-	-	✩	✩
M8076	-	-	✩	✩
M8077	-	-	✩	✩
M8078	-	-	✩	✩
M8079	-	-	✩	✩
M8080	Reserved			
M8081				
M8082				
M8083				
M8084				
M8085				
M8086				
M8087				
M8088				
M8089				
M8090	Reserved			
M8091				
M8092				
M8093				
M8094				
M8095				
M8096				
M8097				
M8098				
M8099	-	-	✩	✩

Device	FX1S	FX1N	FX2N	FX2NC
D8050	Reserved			
D8051				
D8052				
D8053				
D8054				
D8055				
D8056				
D8057				
D8058				
D8059				
D8060	-	-	✩	✩
D8061	✩	✩	✩	✩
D8062	-	-	✩	✩
D8063	✩	✩	✩	✩
D8064	✩	✩	✩	✩
D8065	✩	✩	✩	✩
D8066	✩	✩	✩	✩
D8067	✩	✩	✩	✩
D8068	✩	✩	✩	✩
D8069	✩	✩	✩	✩
D8070	✩	✩	✩	✩
D8071	Reserved			
D8072				
D8073				
D8074			✩	✩
D8075	-	-	✩	✩
D8076	-	-	✩	✩
D8077	-	-	✩	✩
D8078	-	-	✩	✩
D8079	-	-	✩	✩
D8080	-	-	✩	✩
D8081	-	-	✩	✩
D8082	-	-	✩	✩
D8083	-	-	✩	✩
D8084	-	-	✩	✩
D8085	-	-	✩	✩
D8086	-	-	✩	✩
D8087	-	-	✩	✩
D8088	-	-	✩	✩
D8089	-	-	✩	✩
D8090	-	-	✩	✩
D8091	-	-	✩	✩
D8092	-	-	✩	✩
D8093	-	-	✩	✩
D8094	-	-	✩	✩
D8095	-	-	✩	✩
D8096	-	-	✩	✩
D8097	-	-	✩	✩
D8098	-	-	✩	✩
D8099	-	-	✩	✩

Device	FX1S	FX1N	FX2N	FX2NC
M8100	Reserved			
M8101				
M8102				
M8103				
M8104				
M8105				
M8106				
M8107				
M8108				
M8109	-	-	✭	✭
M8110	Reserved			
M8111				
M8112				
M8113				
M8114				
M8115				
M8116				
M8117				
M8118				
M8119				
M8120	Reserved			
M8121	✭	✭	✭	✭
M8122	✭	✭	✭	✭
M8123	✭	✭	✭	✭
M8124	✭	✭	✭	✭
M8125	Reserved			
M8126	✭	✭	✭	✭
M8127	✭	✭	✭	✭
M8128	✭	✭	✭	✭
M8129	✭	✭	✭	✭
M8130	-	-	✭	✭
M8131	-	-	✭	✭
M8132	-	-	✭	✭
M8133	-	-	✭	✭
M8134	Reserved			
M8135				
M8136				
M8137				
M8138				
M8139				
M8140	✭	✭	-	-
M8141	Reserved			
M8142				
M8143				
M8144				
M8145	✭	✭	-	-
M8146	✭	✭	-	-
M8147	✭	✭	-	-
M8148	✭	✭	-	-
M8149	Reserved			

Device	FX1S	FX1N	FX2N	FX2NC
D8100	Reserved			
D8101				
D8102	✭	✭	✭	✭
D8103	Reserved			
D8104				
D8105				
D8106				
D8107				
D8108				
D8109	-	-	✭	✭
D8110	Reserved			
D8111				
D8112				
D8113				
D8114				
D8115				
D8116				
D8117				
D8118				
D8119				
D8120	✭	✭	✭	✭
D8121	✭	✭	✭	✭
D8122	✭	✭	✭	✭
D8123	✭	✭	✭	✭
D8124	✭	✭	✭	✭
D8125	✭	✭	✭	✭
D8126	Reserved			
D8127	✭	✭	✭	✭
D8128	✭	✭	✭	✭
D8129	✭	✭	✭	✭
D8130	-	-	✭	✭
D8131	-	-	✭	✭
D8132	-	-	✭	✭
D8133	-	-	✭	✭
D8134	-	-	✭	✭
D8135	-	-	✭	✭
D8136	✭	✭	✭	✭
D8137	✭	✭	✭	✭
D8138	Reserved			
D8139				
D8140	✭	✭	✭	✭
D8141	✭	✭	✭	✭
D8142	✭	✭	✭	✭
D8143	✭	✭	✭	✭
D8144	Reserved			
D8145	✭	✭	-	-
D8146	✭	✭	-	-
D8147	✭	✭	-	-
D8148	✭	✭	-	-
D8149	Reserved			

Device	FX1S	FX1N	FX2N	FX2NC
M8150	Reserved			
M8151				
M8152				
M8153				
M8154				
M8155				
M8156				
M8157				
M8158				
M8159				
M8160	-	-	☆	☆
M8161	☆	☆	☆	☆
M8162	☆	☆	☆	☆
M8163	Reserved			
M8164	-	-	☆	☆
M8165	Reserved			
M8166				
M8167	-	-	☆	☆
M8168	-	-	☆	☆
M8169	Reserved			
M8170	☆	☆	☆	☆
M8171	☆	☆	☆	☆
M8172	☆	☆	☆	☆
M8173	☆	☆	☆	☆
M8174	☆	☆	☆	☆
M8175	☆	☆	☆	☆
M8176	Reserved			
M8177				
M8178				
M8179				
M8180	Reserved			
M8181				
M8182				
M8183	☆ M504	☆	☆	☆
M8184	☆ M505	☆	☆	☆
M8185	☆ M506	☆	☆	☆
M8186	☆ M507	☆	☆	☆
M8187	☆ M508	☆	☆	☆
M8188	☆ M509	☆	☆	☆
M8189	☆ M510	☆	☆	☆
M8190	☆ M511	☆	☆	☆
M8191	☆ M503	☆	☆	☆
M8192	Reserved			
M8193				
M8194				
M8195				
M8196				
M8197				
M8198				
M8199				

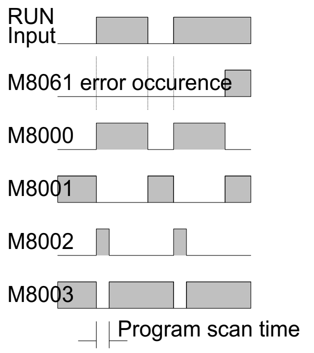
Device	FX1S	FX1N	FX2N	FX2NC
D8150	Reserved			
D8151				
D8152				
D8153				
D8154				
D8155				
D8156				
D8157				
D8158	☆	☆	-	-
D8159	☆	☆	-	-
D8160	Reserved			
D8161				
D8162				
D8163				
D8164	-	-	☆	☆
D8165	Reserved			
D8166				
D8167				
D8168				
D8169				
D8170	Reserved			
D8171				
D8172				
D8173	☆	☆	☆	☆
D8174	☆	☆	☆	☆
D8175	☆	☆	☆	☆
D8176	☆	☆	☆	☆
D8177	☆	☆	☆	☆
D8178	☆	☆	☆	☆
D8179	☆	☆	☆	☆
D8180	☆	☆	☆	☆
D8181	Reserved			
D8182	☆	☆	☆	☆
D8183	☆	☆	☆	☆
D8184	☆	☆	☆	☆
D8185	☆	☆	☆	☆
D8186	☆	☆	☆	☆
D8187	☆	☆	☆	☆
D8188	☆	☆	☆	☆
D8189	☆	☆	☆	☆
D8190	☆	☆	☆	☆
D8191	☆	☆	☆	☆
D8192	☆	☆	☆	☆
D8193	☆	☆	☆	☆
D8194	☆	☆	☆	☆
D8195	☆	☆	☆	☆
D8196	Reserved			
D8197				
D8198				
D8199				

Note;
When using an N:N network configuration with the FX1S, M503 to M511 are used in place of the regular M devices as shown above. D208 to D218 are used in place of the regular D devices shown on the next page.

Device	FX1S	FX1N	FX2N	FX2NC
M8200	-	☆	☆	☆
M8201	-	☆	☆	☆
M8202	-	☆	☆	☆
M8203	-	☆	☆	☆
M8204	-	☆	☆	☆
M8205	-	☆	☆	☆
M8206	-	☆	☆	☆
M8207	-	☆	☆	☆
M8208	-	☆	☆	☆
M8209	-	☆	☆	☆
M8210	-	☆	☆	☆
M8211	-	☆	☆	☆
M8212	-	☆	☆	☆
M8213	-	☆	☆	☆
M8214	-	☆	☆	☆
M8215	-	☆	☆	☆
M8216	-	☆	☆	☆
M8217	-	☆	☆	☆
M8218	-	☆	☆	☆
M8219	-	☆	☆	☆
M8220	-	☆	☆	☆
M8221	-	☆	☆	☆
M8222	-	☆	☆	☆
M8223	-	☆	☆	☆
M8224	-	☆	☆	☆
M8225	-	☆	☆	☆
M8226	-	☆	☆	☆
M8227	-	☆	☆	☆
M8228	-	☆	☆	☆
M8229	-	☆	☆	☆
M8230	-	☆	☆	☆
M8231	-	☆	☆	☆
M8232	-	☆	☆	☆
M8233	-	☆	☆	☆
M8234	-	☆	☆	☆
M8235	☆	☆	☆	☆
M8236	☆	☆	☆	☆
M8237	☆	☆	☆	☆
M8238	☆	☆	☆	☆
M8239	☆	☆	☆	☆
M8240	☆	☆	☆	☆
M8241	☆	☆	☆	☆
M8242	☆	☆	☆	☆
M8243	☆	☆	☆	☆
M8244	☆	☆	☆	☆
M8245	☆	☆	☆	☆
M8246	☆	☆	☆	☆
M8247	☆	☆	☆	☆
M8248	☆	☆	☆	☆
M8249	☆	☆	☆	☆
M8250	☆	☆	☆	☆
M8251	☆	☆	☆	☆
M8252	☆	☆	☆	☆
M8253	☆	☆	☆	☆
M8254	☆	☆	☆	☆
M8255	☆	☆	☆	☆

Device	FX1S	FX1N	FX2N	FX2NC
D8200	Reserved			
D8201	☆ D201	☆	☆	☆
D8202	☆ D202	☆	☆	☆
D8203	☆ D203	☆	☆	☆
D8204	☆ D204	☆	☆	☆
D8205	☆ D205	☆	☆	☆
D8206	☆ D206	☆	☆	☆
D8207	☆ D207	☆	☆	☆
D8208	☆ D208	☆	☆	☆
D8209	☆ D209	☆	☆	☆
D8210	☆ D210	☆	☆	☆
D8211	☆ D211	☆	☆	☆
D8212	☆ D212	☆	☆	☆
D8213	☆ D213	☆	☆	☆
D8214	☆ D214	☆	☆	☆
D8215	☆ D215	☆	☆	☆
D8216	☆ D216	☆	☆	☆
D8217	☆ D217	☆	☆	☆
D8218	☆ D218	☆	☆	☆
D8219	Reserved			
D8220	Reserved			
D8221				
D8222				
D8223				
D8224				
D8225				
D8226				
D8227				
D8228				
D8229				
D8230	Reserved			
D8231				
D8232				
D8233				
D8234				
D8235				
D8236				
D8237				
D8238				
D8239				
D8240	Reserved			
D8241				
D8242				
D8243				
D8244				
D8245				
D8246				
D8247				
D8248				
D8249				
D8250	Reserved			
D8251				
D8252				
D8253				
D8254				
D8255				

6.2 PLC Status (M8000 to M8009 and D8000 to D8009) FX1S FX1N FX2N FX2NC

Diagnostic Device	Operation
M8000 (✗) RUN monitor NO contact	
M8001 (✗) RUN monitor NC contact	
M8002 (✗) Initial pulse NO contact	
M8003 (✗) Initial pulse NC contact	
M8004 (✗) Error occurrence	ON when one or more error flags from the range M8060 to M8067 are ON
M8005 (✗) Battery voltage Low *(Not FX1S, FX1N)*	On when the battery voltage is below the value set in D8006
M8006 (✗) Battery error latch *(Not FX1S, FX1N)*	Latches the battery Low error
M8007 (✗) Momentary power failure *(Not FX1S, FX1N)*	See note 2
M8008 (✗) Power failure *(Not FX1S, FX1N)*	Power loss has occurred See note 2
M8009 (✗) 24V DC Down *(Not FX1S, FX1N)*	Power failure of 24V DC service supply

Diagnostic Device	Operation
D8000 (✍) Watchdog timer	FX1S, FX1N, FX2N, FX2NC: 200ms See note 1
D8001 (✗) PLC type and version	FX1S: 22 FX1N: 26 E.g. 26100 = FX1N, V1.00 FX2N: 24 FX2NC: 24
D8002 (✗) Memory capacity (see also D8102)	0002: 2K steps (FX1S only) 0004: 4K steps (FX2N, FX2NC) 0008: 8K or 16k steps (FX1N, FX2N, FX2NC)
D8003 (✗) Memory type	00H = Option RAM, 01H = Option EPROM, 02H = Option EEPROM, 0AH = Option EEPROM (protected) 10H = Built-in MPU memory
D8004 (✗) Error number M☆☆☆☆	The contents of this register ☆☆☆☆ identifies which error flag is active, i.e. if ☆☆☆☆ = 8060 identifies M8060
D8005 (✗) Battery voltage *(Not FX1S, FX1N)*	E.g. 36 = 3.6 volts
D8006 (✗) Low battery voltage *(Not FX1S, FX1N)*	The level at which a low battery voltage is detected
D8007 (✗) Power failure count *(Not FX1S, FX1N)*	The number of times a momentary power failure has occurred since power ON.
D8008 Power failure detection. *(Not FX1S, FX1N)*	The time period before shut down when a power failure occurs (default 10ms) See note 2
D8009 (✗) 24V DC failed device *(Not FX1S, FX1N)*	Lowest device affected by 24V DC power failure

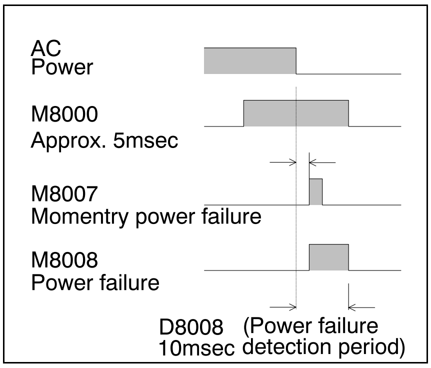
For symbol key see page 6-1.

Note 1:

- The contents of this register can be changed by the user. Settings in 1 msec steps are possible. The value should be set greater than the maximum scan time (D8012) to ensure constant scan operation.

Note 2:

- When the power supply used is 200V AC, the power down detection period is determined by the value of D8008. This can be altered by the user within the allowable range of 10 to 100msec.



6.3 Clock Devices (M8010 to M8019 and D8010 to D8019)

FX1S | FX1N | FX2N | FX2NC

Diagnostic Device	Operation
M8010	Reserved
M8011 (✗) 10 msec clock pulse	Oscillates in 10 msec cycles
M8012 (✗) 100 msec clock pulse	Oscillates in 100 msec cycles
M8013 (✗) 1 sec clock pulse	Oscillates in 1 sec cycles
M8014 (✗) 1 min clock pulse	Oscillates in 1 min cycles

Diagnostic Device	Operation
D8010 (✗) Present scan time	Current operation cycle / scan time in units of 0.1 msec (waiting time for constant scan mode is included)
D8011 (✗) Minimum scan time	Minimum cycle/ scan time in units of 0.1 msec (waiting time for constant scan mode is included)
D8012 (✗) Maximum scan time	Maximum cycle/ scan time in units of 0.1 msec (waiting time for constant scan mode is included)

The following devices apply to FX2N, FX1N and FX1S PLC's as standard and to the FX2NC PLC when a real time clock option board installed.

M8015 Time setting	When ON - clock stops, ON ➪ OFF restarts clock
M8016 Register data	When ON D8013 to 19 are frozen for display but clock continues
M8017 Min. rounding	When pulsed ON set RTC to nearest minute
M8018 (✗) RTC available	When ON Real Time Clock is installed
M8019 Setting error	Clock data has been set out of range

D8013 Seconds	Seconds data for use with an RTC (0 - 59)
D8014 Minute data	Minute data for use with an RTC (0-59)
D8015 Hour data	Hour data for use with an RTC (0-23)
D8016 Day data	Day data for use with an RTC (1-31)
D8017 Month data	Month data for use with an RTC (1-12)
D8018 Year data	Year data for use with an RTC (00-99 or 1980-2079, can be selected)
D8019 Weekday data	Weekday data for use with an RTC (0-6)

For symbol key see page 6-1.

6.4 Operation Flags (M8020 to M8029 and D8020 to D8029)

FX1S FX1N FX2N FX2NC

Diagnostic Device	Operation
M8020 (✗) Zero	Set when the result of an ADD (FNC 20) or SUB (FNC 21) is “0”
M8021 (✗) Borrow	Set when the result of a SUB (FNC 21) is less than the min. negative number
M8022 (✍) Carry	Set when ‘carry’ occurs during an ADD (FNC 20) or when an overflow occurs as a result of a data shift operation
M8024 *(Not FX1S, FX1N)*	BMOV (FNC 15) reverse mode. See note 3
M8025 *(Not FX1S, FX1N)*	When ON HSC (FNC 53 - 55) instructions are processed even when the external HSC reset input is activated
M8026 *(Not FX1S, FX1N)*	RAMP (FNC 67) hold mode
M8027 *(Not FX1S, FX1N)*	PR (FNC 77) 16 element data string
M8028 **Note:** *Separate FX1S and FX2N/2NC operation (Not FX1N)*	FX1S: Change timers T32 ~ T62 to 10ms type FX2N, FX2NC: Permit FROM/TO to interrupt program. (V3.00 and above)
M8029 (✗) Instruction execution complete	Set on the completion of operations such as DSW (FNC 72), RAMP (FNC 67) etc.

Diagnostic Device	Operation
D8020 (✍) *See note 4*	Input filter setting for devices; X000 to X017 (FX2N,FX2NC) default value = 10 msec, zero value = 50 µsec (X000, X001: 20 µsec) X000 to X007 (FX1S,FX1N) default value = 10msec zero value = 50 µsec (X000, X001: 10 µsec)
D8021 (✍) *(Not FX1N, FX@N, FX2NC* *See note 4*	Input filter setting for devices; X010 to X017 (FX1S) default value = 10 msec, zero value = 50 µsec
D8022 -D8027	Reserved
D8028 (✗)	Current value of the Z0 index register See note 5
D8029 (✗)	Current value of the V0 index register See note 5

For symbol key see page 6-1.

Note 3

- If M8024 is used with a BMOV (FNC 15) instruction, it will operate as follows;
  M8024 OFF - Normal operation (Forwarding direction is [S] to [D])
  M8024 ON - Reverse operation (Forwarding direction becomes [D] to [S])
  This device is not supported in FX1S and FX1N

Note 4

- The settings for input filters only apply to the main processing units which use 24V DC inputs. AC input filters are not adjustable.

Note 5

- For Z1~Z7 and V1~V7 (D8128~D8195) please see page 6-20.

6.5 PLC Operation Mode (M8030 to M8039 and D8030 to D8039)

FX1S FX1N FX2N FX2NC

Diagnostic Device	Operation
M8030 (✍) Battery LED OFF *(Not FX1S, FX1N)*	Battery voltage is low but BATT.V LED not lit
M8031 (✍) Non-latch memory all clear	Current device settings are reset at next END, i.e. contacts, coils and current data values for Y, M, S, T, C and D devices respectively. Special devices and file registers which have default settings are refreshed with those defaults
M8032 (✍) Latch memory all clear	
M8033 (✍) Memory hold in 'stop' mode	The device statuses and settings are retained when the PLC changes from RUN to STOP and back into RUN
M8034 (✍) All outputs disable	All of the physical switch gear for activating outputs is disabled. However, the program still operates normally.
M8035 (✍S) Forced operation mode	By using forced operation mode, i.e.M8035 is turned ON, it is possible to perform remote RUN/STOP or pulsed RUN/ STOP operation. Please see Chapter 10 for example operation
M8036 (✍S) Forced RUN signal	
M8037 (✍S) Forced STOP signal	
M8038 N to N networking	For the setting of devices when using an N to N network
M8039 (✍) Constant scan mode	When ON the PLC executes the user program within a constant scan duration. The difference between the actual end of the program operation and the set constant scan duration causes the PLC to 'pause'.

Diagnostic Device	Operation
D8030 (✗) *(Not FX2N, FX2NC)*	Value read from first setting "pot" in msec, (0 to 255)
D8031 (✗) *(Not FX2N, FX2NC)*	Value read from second setting "pot" in msec, (0 to 255)
D8032 -D8038	Reserved
D8039 (✍) Constant scan duration	This register can be written to by the user to define the duration of the constant scan. Resolutions of 1msec are possible. This register has a default setting 0 msec which will be initiated during power ON.

For symbol key see page 6-1.

6.6 Step Ladder (STL) Flags (M8040 to M8049 and D8040 to D8049)

FX1S | FX1N | FX2N | FX2NC

Diagnostic Device	Operation
M8040 (✍) STL transfer disable	When ON STL state transfer is disabled
M8041 (✍S) Transfer start	When ON STL transfer from initial state is enabled during automatic operation (ref. IST FNC 60)
M8042 (✍) Start pulse	A pulse output is given in response to a start input (ref. IST FNC 60)
M8043 (✍S) Zero return complete	On during the last state of ZERO RETURN mode (ref. IST FNC 60)
M8044 (✍S) Zero point condition	ON when the machine zero is detected (ref. IST FNC 60)
M8045 (✍) All output reset disable	Disables the 'all output reset' function when the operation mode is changed (ref. IST FNC 60)
M8046 (✗) STL state ON	ON when STL monitoring has been enabled (M8047) and there is an active STL state
M8047 (✍) Enable STL monitoring	When ON D8040 to D8047 are enabled for active STL step monitoring
M8048 (✗) Annunciator ON *(Not FX1S, FX1N)*	ON when Annunciator monitoring has been enabled (M8049) and there is an active Annunciator flag
M8049 (✍) Enable Annunciator monitoring *(Not FX1S, FX1N)*	When ON D8049 is enabled for active Annunciator state monitoring

Diagnostic Device	Operation
D8040 (✗) Lowest active STL step	Up to 8 active STL states, from the range S0 to S899, are stored in D8040 to D8047 in ascending numerical order. (Updated at END)
D8041 (✗) 2nd active STL state	
D8042 (✗) 3rd active STL state	
D8043 (✗) 4th active STL state	
D8044 (✗) 5th active STL state	
D8045 (✗) 6th active STL state	
D8046 (✗) 7th active STL state	
D8047 (✗) 8th active STL state	
D8048	Reserved
D8049 (✗) Lowest active Annunciator *(Not FX1S, FX1N)*	Stores the lowest currently active Annunciator from the range S900 to S999 (Updated at END)

For symbol key see page 6-1.

General note:

- M8046 to M8049 STL states are updated when the END instruction is executed.

6.7 Interrupt Control Flags (M8050 to M8059 and D8050 to D8059)

FX1S | FX1N | FX2N | FX2NC

<table>
<tr><th>Diagnostic Device</th><th>Operation</th></tr>
<tr><td>M8050 (✍)
I00❑ disable</td><td rowspan="9">When the EI (FNC 04) instruction is driven in the user program, all interrupts are enabled unless the special M devices noted here are driven ON. In that case for each special M coil that is ON, the associated interrupt is disabled, i.e. will not operate.
Note ❑❑ denotes all types of that interrupt</td></tr>
<tr><td>M8051 (✍)
I10❑ disable</td></tr>
<tr><td>M8052 (✍)
I20❑ disable</td></tr>
<tr><td>M8053 (✍)
I30❑ disable</td></tr>
<tr><td>M8054 (✍)
I40❑ disable</td></tr>
<tr><td>M8055 (✍)
I50❑ disable</td></tr>
<tr><td>M8056 (✍)
I6❑❑ disable
(Not FX1S, FX1N)</td></tr>
<tr><td>M8057 (✍)
I7❑❑ disable
(Not FX1S, FX1N)</td></tr>
<tr><td>M8058 (✍)
I8❑❑ disable
(Not FX1S, FX1N)</td></tr>
<tr><td>M8059(✍)
I010 to I060 disabled as a single group
(Not FX1S, FX1N)</td><td>I010 ~ I060 is disabled for high speed counter interrupt (FNC53)
When this flag is ON, the associated interrupt is disabled and therefore will not operate.</td></tr>
</table>

Diagnostic Device	Operation
D8050 -D8059	Reserved

For symbol key see page 6-1.

6.8 Error Detection Devices (M8060 to M8069 and D8060 to D6069)

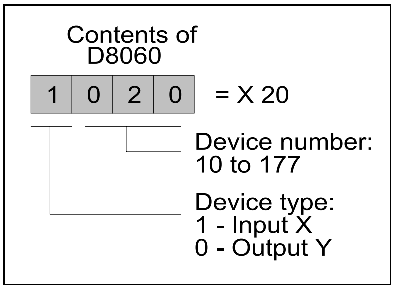
FX1S	FX1N	FX2N	FX2NC

Diagnostic Device	Operation				
	Detection			PROG.E LED	PLC STATUS
	ON-OFF	OFF-ON	Other		
M8060 (✗) I/O configuration error *(Not FX1S, FX1N)*	✓	✓	While the PLC is in RUN	OFF	RUN
M8061 (✗) PLC hardware error	✓	-		ON	STOP
M8062 (✗) PC/HPP comms error on programming port *(Not FX1S, FX1N)*	-	-	When a signal from the programming port is received	OFF	RUN
M8063(✗)(✍R) Parallel link/ RS232-C and RS485 (422) comms error on optional port			When a signal from the optional port is received		
M8064 (✗) Parameter error	✓	✓	When the program is changed (PLC in STOP) and when a program is transferred (PLC in STOP)	Flash	STOP
M8065 (✗) Syntax error					
M8066 (✗) Program error					
M8067(✗)(✍R) Operation error	-	-	While in PLC is in RUN	OFF	RUN
M8068 (✍) Operation error latch					
M8069 (✍) I/O bus error *(Not FX1S, FX1N)*			See note 7	-	-

Diagnostic Device	Operation
D8060 (✗) *(Not FX1S, FX1N)*	The first I/O number of the unit or block causing the error - See note 6
D8061 (✗)	Error code for hardware error - See appropriate error code table
D8062 (✗) *(Not FX1S, FX1N)*	Error code for PC/HPP Communications error - See appropriate error code table
D8063(✗)(-R)	Error code for parallel link error - See FX communication users manual
D8064 (✗)	Error code identifying parameter error - See appropriate error code table
D8065 (✗)	Error code identifying syntax error - See appropriate error code table
D8066 (✗)	Error code identifying program construction error See appropriate error code table
D8067(✗)(✍R)	Error code identifying operation error. See appropriate error code table
D8068 (✍)	Operation error step number latched
D8069(✗)(✍R)	Step numbers for found errors corresponding to flags M8065 to M8067

For symbol key see page 6-1.

- Please see the following page for the notes referenced in this table.



Note 6:

•If the unit or block corresponding to a programmed I / O number is not actually loaded, M8060 is set to ON and the first device number of the erroneous block is written to D8060.

Note 7:

•An I/O bus check is executed when M8069 is turned ON.
If an I/O bus error occurs, error code 6103 is written to D8069 and M8061 is turned ON.
If an Extension unit 24V failure occurs, error code 6104 is written to D8061 and M8061 is turned ON. M8009 will then be turned ON and the I/O address of the lowest numbered device affected by the 24V DC power failure is written to D8009

General note:

•HPP refers to Handy programming panel.

6.9 Link and Special Operation Devices (M8070 to M8099 and D8070 to D8099)

FX1S | FX1N | FX2N | FX2NC

Diagnostic Device	Operation
M8070 (✍R)	Driven when the PLC is a master station in a parallel link application
M8071 (✍R)	Driven when the PLC is a slave station in a parallel link application
M8072 (✗)	ON while the PLC is operating in a parallel link
M8073 (✗)	ON when M8070/ M8071 are incorrectly set during parallel link operations
M8074	Reserved
M8075 *(Not FX1S, FX1N)*	When executing Sampling trace in GX-Developer or FX-PCS/WIN-E, these devices are used by the PLC internal system
M8076 *(Not FX1S, FX1N)*	
M8077 *(Not FX1S, FX1N)*	ON during sampling trace
M8078 *(Not FX1S, FX1N)*	ON when sampling trace complete
M8079 *(Not FX1S, FX1N)*	When executing Sampling trace in GX-Developer or FX-PCS/WIN-E, this device is used by the PLC internal system
M8080 -M8098	Reserved
M8099 (✍) *(Not FX1S, FX1N)*	High speed free timer operation When ON, continue counting free ring timer (D8099) →

Diagnostic Device	Operation
D8070 (✗)	Parallel link watchdog time - 500 msec
D8071 - D8073	Reserved
D8074 *(Not FX1S, FX1N)*	When executing Sampling trace in GX-Developer or FX-PCS/WIN-E, these devices are used by the PLC internal system
D8075 *(Not FX1S, FX1N)*	
D8076 *(Not FX1S, FX1N)*	
D8077 *(Not FX1S, FX1N)*	
D8078 *(Not FX1S, FX1N)*	
D8079 *(Not FX1S, FX1N)*	
D8080 to D8095 *(Not FX1S, FX1N)*	
D8096 to D8098 *(Not FX1S, FX1N)*	
D8099 *(Not FX1S, FX1N)*	Free ring timer, range: 0-32,767 in units of 0.1 msec (for use in measuring high speed pulse input durations) See section 10.9.2

For symbol key see page 6-1.

6.10 Miscellaneous Devices (M8100 to M8119 and D8100 to D8119)

FX1S | FX1N | FX2N | FX2NC

Diagnostic Device	Operation
M8109 (✗) *(Not FX1S, FX1N)*	Output refresh error

Diagnostic Device	Operation
D8102 (✗) Memory Capacity	0002: 2K steps (FX1S only) 0004: 4K steps (FX2N, FX2NC) 0008: 8K steps (FX1N, FX2N, FX2N) 0016: 16K steps (FX2N, FX2NC)
D8109 (✗) *(Not FX1S, FX1N)*	Output refresh error, lowest device number; 0, 10, 20, etc.

6.11 Communication Adapter Devices, i.e. 232ADP, 485ADP (M8120 to M8129 and D8120 to D8129)

FX1S | FX1N | FX2N | FX2NC

Diagnostic Device	Operation
M8120	Reserved
M8121(✗)(✍R)	Data transmission delayed (RS instruction)
M8122 (✍R)	Data transmission flag (RS instruction)
M8123 (✍R)	Finished receiving data (RS instruction)
M8124(✗)	Carrier detection flag (RS instruction)
M8125	Reserved
M8126	Global flag (Computer link)
M8127 (✍)	On Demand handshake flag (Computer link)
M8128 (✍)	On Demand error flag (Computer link)
M8129 (✍)	On Demand Byte/Word change over (Computer link), Time out evaluation flag (RS instruction)

Diagnostic Device	Operation
D8120	Communications format (RS instruction, Computer link)
D8121	Station number setting (Computer link)
D8122(✗)(✍R)	Amount of remaining data to be transmitted (RS instruction)
D8123(✗)(✍R)	Amount of data already received (RS instruction)
D8124 (✍)	Data header, default STX (02H) (RS instruction)
D8125 (✍)	Data terminator, default ETX (03H) (RS instruction)
D8126	Reserved
D8127 (✍)	On Demand head device register (Computer link)
D8128 (✍)	On Demand data length register (Computer link)
D8129	Data network 'time-out' timer value (RS instruction, Computer link)

For symbol key see page 6-1.

6.12 High Speed Zone Compare Table Comparison Flags (M8130 to M8148 and D8130 to D8148)

FX1S | FX1N | FX2N | FX2NC

Diagnostic Device	Operation
M8130 *(Not FX1S, FX1N)* See note 8	Selects comparison tables to be used with the HSZ instruction
M8131 (✗)(✍) *(Not FX1S, FX1N)* See note 8	ON when the HSZ comparison table has been completed.
M8132 *(Not FX1S, FX1N)* See note 8	Selects the use of the PLSY instruction with the HSZ comparison tables
M8133 (✗)(✍) *(Not FX1S, FX1N)* See note 8	ON when the HSZ comparison table (when used with the PLSY instruction) has been completed.
M8134-M8139	Reserved

Diagnostic Device	Operation
D8130 (✗)(✍) *(Not FX1S, FX1N)*	Contains the number of the current record being processed in the HSZ comparison table
D8131 (✗)(✍) *(Not FX1S, FX1N)*	Contains the number of the current record being processed in the HSZ comparison table when the PLSY operation has been enabled
D8132 D8133 (✗)(✍) *(Not FX1S, FX1N)*	Contains the source (output pulse frequency) data for the PLSY instruction when used with the HSZ comparison table
D8134 D8135 (✗) (✍) *(Not FX1S, FX1N)*	Contains a copy of the value for the current comparison when the HSZ comparison table and combined PLSY output are used. This data is only available in 32 bit or double word format.
D8136 D8137 (✗) (✍)	Contains the total number of pulses that have been output using the PLSY (or PLSR) instruction on Y000 and Y001. This data is only available in 32 bit or double word format
D8138 - D8139	Reserved

Note 8

- See section 5.6.6 for full explanation and use.

Diagnostic Device	Operation
M8140 (✗)(✍) *(Not FX2N, FX2NC)*	When ON, clears pulse output in FNC156(ZRN) instruction
M8141 to M8144	Reserved
M8145 (✍) *(Not FX2N, FX2NC)*	Y000 Pulse output stop command
M8146 (✍) *(Not FX2N, FX2NC)*	Y001 Pulse output stop command
M8147 (✗) *(Not FX2N, FX2NC)*	Y000 Pulse output monitor (Busy/Ready)
M8148 (✗) *(Not FX2N, FX2NC)*	Y001 Pulse output monitor (Busy/Ready)

For symbol key see page 6-1

Diagnostic Device	Operation
D8140 D8141 (✗) (✍)	Contains the total number of pulses that have been output to Y0 using the PLSY or PLSR instructions. This data is only available in 32 bit or double word format.
D8142 D8143 (✗) (✍)	Contains the total number of pulses that have been output to Y1 using the PLSY or PLSR instructions. This data is only available in 32 bit or double word format.
D8145 (✍) *(Not FX2N, FX2NC)*	FNC156(ZRN), FNC158(DRVI), FNC159(DRVA) Bias value setting (default:0)
D8146 (✍) *(Not FX2N, FX2NC)* D8147 (✍) *(Not FX2N, FX2NC)*	FNC156(ZRN), FNC158(DRVI), FNC159(DRVA) Max. speed setting (default:100,000)
D8148 (✍) *(Not FX2N, FX2NC)*	FNC156(ZRN), FNC158(DRVI), FNC159(DRVA) Acceleration/ Deceleration time setting (default:100)

6.13 Miscellaneous Devices (M8160 to M8199)

FX1S FX1N FX2N FX2NC

Diagnostic Device	Operation
M8160 (✍) *(Not FX1S, FX1N)*	Selection of XCH operation to swap bytes in a single data word
M8161 (✍)	Selection of 8 bit operations for applied instructions ASC, RS, ASCI, HEX, CCD
M8162 (✍)	High speed mode for Parallel link, 2 data words Read/write only
M8164 (✍) *(Not FX1S, FX1N)*	When ON, a value in D8164 is used as the number of FROM/TO exchange points. (FX2N/2NC CPU Version 2.00 and above)
M8167 (✍) *(Not FX1S, FX1N)*	Selection of hexadecimal input mode for the HKY instruction
M8168 (✍) *(Not FX1S, FX1N)*	Selection of BCD mode for use with the SMOV instruction
M8169	Reserved
M8170 (✍R) X0 pulse catch	When the leading edge of a pulse is received at an input from the range X0 to X5 the associated M device detailed here is set ON. By resetting the same device within the user program the next pulse occurrence will again set the M coil ON. Hence, fast input pulses are 'caught' and stored. This operation requires the EI (FNC04) instruction to be active. For details see page 6-12
M8171 (✍R) X1 pulse catch	
M8172 (✍R) X2 pulse catch	
M8173 (✍R) X3 pulse catch	
M8174 (✍R) X4 pulse catch	
M8175 (✍R) X5 pulse catch	
M8176 -M8199	Reserved

For symbol key see page 6-1.

6.14 Miscellaneous devices (D8158 to D8164) and Index Registers (D8182 to D8199)

FX1S FX1N FX2N FX2NC

Diagnostic Device	Operation
D8158 (✍) (Not FX2N, FX2NC)	Control device for FX1N-5DM*1 Default: k-1
D8159 (✍) (Not FX2N, FX2NC)	Control device for FX1N-5DM*1 Default: k-1
D8164 (✍) (Not FX1S, FX1N)	Number of FROM/TO exchange points (FX2N/2NC CPU Version 2.00 and above)
D8181 (✗)	Reserved
D8182 (✗)	Value of Z1 index register
D8183 (✗)	Value of V1 index register
D8184 (✗)	Value of Z2 index register
D8185 (✗)	Value of V2 index register
D8186 (✗)	Value of Z3 index register

Diagnostic Device	Operation
D8187 (✗)	Value of V3 index register
D8188 (✗)	Value of Z4 index register
D8189 (✗)	Value of V4 index register
D8190 (✗)	Value of Z5 index register
D8191 (✗)	Value of V5 index register
D8192 (✗)	Value of Z6 index register
D8193 (✗)	Value of V6 index register
D8194 (✗)	Value of Z7 index register
D8195 (✗)	Value of V7 index register

For symbol key see page 6-1.

*1 See Chapter 10.19.2 for more information

6.15 N:N Network Related Flags and Data Registers

FX1S	FX1N	FX2N	FX2NC

Note: Functionality available for FX2N CPU Version 2.00 and above

Diagnostic Device	Operation
M8183 (✗) (For FX1S use M504)	ON when communication error in master station
M8184 (✗) (For FX1S use M505)	ON when communication error in 1st slave station
M8185 (✗) (For FX1S use M506)	ON when communication error in 2nd slave station
M8186 (✗) (For FX1S use M507)	ON when communication error in 3rd slave station
M8187 (✗) (For FX1S use M508)	ON when communication error in 4th slave station
M8188 (✗) (For FX1S use M509)	ON when communication error in 5th slave station
M8189 (✗) (For FX1S use M510)	ON when communication error in 6th slave station
M8190 (✗) (For FX1S use M511)	ON when communication error in 7th slave station
M8191 (✗) (For FX1S use M503)	ON when communicating to another station

Diagnostic Device	Operation
D8173 (✗)	Station number
D8174 (✗)	Total number of slave stations
D8175 (✗)	Refresh range
D8176 See note 10	Station number setting Default value k0
D8177 See note 10	Total number of slave stations setting Default value k7
D8178 See note 10	Refresh range setting Default value k0
D8179 See note 10	Retry count setting Default value k3
D8180 See note 10	Comms time-out setting Default value k5
D8201 (✗) (For FX1S use D201)	Current network scan time
D8202 (✗) (For FX1S use D202)	Maximum network scan time
D8203 (✗) (For FX1S use D203)	Number of communication error at master station
D8204 to D8210 (✗) (For FX1S use D204 to D210)	Number of communication error at respective slave station
D8211 (✗) (For FX1S use D2113)	Code of communication error at master station
D8212 to D8218 (✗) (For FX1S use D212 to D218)	Code of communication error at respective slave station

Note 9

- Devices M503-M511 and D201-D255 in the FX1S cannot be applied to other functions in the user program. These devices are used exclusively for the N:N Network.

Note 10

- When these devices are not being used for an N:N Network their respective default values are all ‘0’. The relevant default values are assumed at each power ON.

6.16 Up/Down Counter Control (M8200 to M8234 and D8219 to D8234)

FX1S	FX1N	FX2N	FX2NC

Diagnostic Device	Operation
M8200 - M8234 (✍)	When M8☆☆☆ is operated, counter C☆☆☆ functions as a down counter. When M8☆☆☆ is not operated the associated counter operates as an up counter

Diagnostic Device	Operation
D8219 -D8234	Reserved

For symbol key see page 6-1.

6.17 High Speed Counter Control (M8235 to M8255 and D8235 to D8255)

FX1S	FX1N	FX2N	FX2NC

Diagnostic Device	Operation
M8235 -M8245 (✍)	When M8☆☆☆ is operated, the 1 phase high speed counter C☆☆☆ functions as a down counter. When M8☆☆☆ is not operated the associated counter operates as an up counter. The available counters depends upon the PLC type.
M8246 - M8255 (✗)(✍)	When M8☆☆☆ is operated, the 2 phase high speed counter C☆☆☆ functions as a down counter. When M8☆☆☆ is not operated the associated counter operates as an up counter. The available counters depends upon the PLC type.

Diagnostic Device	Operation
D8235 -D8255	Reserved

For symbol key see page 6-1.

6.18 Error Code Tables

FX1S FX1N FX2N FX2NC

Error Detection Device	Stored Error Number	Associated Meaning	Action
D8061 **PLC Hardware error**	0000	No error	Check the cable connection between the extension unit/block and the PLC
	6101	RAM error	
	6102	Operation circuit error	
	6103	I/O bus error (M8069 = ON)	
	6104	Extension unit 24V failure (M8069=ON)	
	6105	Watch Dog Timer error	Scan time has exceeded the WDT time value set in D8000. Check user program.

Error Detection Device	Stored Error Number	Associated Meaning	Action
D8062 **PC/HPP communication error** **(Not FX1S, FX1N)**	0000	No error	Check the cable connection between the programming device and the PLC
	6201	Parity/ overrun/ framing error	
	6202	Communications character error	
	6203	Communication data sum check error	
	6204	Data format error	
	6205	Command error	

Error Detection Device	Stored Error Number	Associated Meaning	Note
D8063 **Serial communication errors**	0000	No error	Check communication settings, parameters and applicable devices. (Computer link, N:N network, Parallel link etc.) Refer to FX Communication Users Manual for wiring techniques
	6301	Parity/ overrun/ framing error	
	6302	Comms character error	
	6303	Comms data sum check error	
	6304	Comms data format error	
	6305	Command error Computer link - received command other than GW (global) when station number was FF	
	6306	Watchdog timer error	
	6312	Parallel link character error	
	6313	Parallel link data sum check error	
	6314	Parallel link data format error	

Error Detection Device	Stored Error Number	Associated Meaning	Action
D8064 **Parameter error**	0000	No error	STOP the PLC, check parameter, if incorrect change to a suitable value
	6401	Program sum check error	
	6402	Memory capacity setting error	
	6403	Latched device area setting error	
	6404	Comment area setting error	
	6405	File register area setting error	
	6406 - 6408	Reserved	
	6409	Other setting error	

Error Detection Device	Stored Error Number	Associated Meaning	Action
D8065 **Syntax error**	0000	No error	During programming, each instruction is checked as it is entered. If a syntax error is detected, re-enter the instruction correctly
	6501	Incorrect instruction/ device symbol/ device number combination	
	6502	No timer or counter coil before setting value	
	6503	1)No setting value following either a timer or a counter coil 2)Insufficient number of operands for an applied instruction	
	6504	1)The same label number is used more than once 2)The same interrupt input or high speed counter input is used more than once	
	6505	Device number is outside the allowable range	
	6506	Invalid applied instruction	
	6507	Invalid Pointer device [P] assignment for Jump or Call instruction	
	6508	Invalid Interrupt pointer device [I] assignment	
	6509	Other error	
	6510	MC nesting (N) number error	
	6511	The same interrupt input or high speed counter input is used more than once	

Error Detection Device	Stored Error Number	Associated Meaning	Action
D8066 **Circuit error**	0000	No error	A circuit error occurs if a combination of instructions is incorrect or badly specified. Select programming mode and correct the identified error.
	6601	LD and LDI is used continuously 9 or more times in succession	
	6602	1)No LD/ LDI instruction. The use of LD/LDI or ANB/ORB instruction is incorrect. 2)The following instructions are not connected to the active bus line: STL, RET, MCR, (P)ointer, (I)nterrupt, EI, DI, SRET, IRET, FOR, NEXT, FEND and END 3)When MPP is missing	
	6603	MPS is used continuously more than 12 times	
	6604	The use of MPS, MRD, MPP instruction is incorrect.	
	6605	1)The STL instruction is continuously used 9 times or more 2)MC, MCR instruction, (I)nterrupt pointer or SRET instruction is used within an STL program area 3)RET has not been used in the program or is not connected to an STL instruction	
	6606	1)No (P)ointer, (I)nterrupt pointer 2)No SRET/ IRET 3)An (I)nterrupt pointer, SRET or IRET has been used within the main program 4)STL, RET, MC or MCR have been used within either a subroutine or an interrupt routine	
	6607	1)The use of FOR and NEXT is incorrect 2)The following instructions have been used within a FOR -NEXT loop: STL, RET, MC, MCR, IRET, SRET, FEND or END	
	6608	1)The use of MC/ MCR is incorrect 2)Missing MCR N0 3)SRET, IRET instruction or an (I)nterrupt pointer has been used within an MC/ MCR instruction area	
	6609	Other error	

Continued on next page...

Error Detection Device	Stored Error Number	Associated Meaning	Action
D8066 **Circuit error**	6610	LD, LDI is used continuously 9 or more times in succession	A circuit error occurs if a combination of instructions is incorrect or badly specified. Select programming mode and correct the identified error.
	6611	Number of LD/LDI instructions is more than ANB/ORB instructions	
	6612	Number of LD/LDI instructions is less than ANB/ORB instructions	
	6613	MPS is used continuously more than 12 times	
	6614	MPS instruction missing	
	6515	MPP instruction missing	
	6616	Unauthorized use of the MPS/ MRD/ MPP instructions; possible coil missing	
	6617	One of the following instructions is not connected to the active bus line: STL, RET, MCR, (P)ointer, (I)nterrupt pointer, EI, DI, SRET, IRET, FOR, NEXT, FEND and END	
	6618	STL, RET, MC or MCR programmed within either a subroutine or an interrupt routine	
	6619	Invalid instruction programmed within a FOR - NEXT loop: STL, RET, MC, MCR, (I)nterrupt pointer, IRET and SRET	
	6620	FOR - NEXT instruction nesting levels (5) exceeded	
	6621	The number of FOR and NEXT instructions does not match	
	6622	NEXT instruction not found	
	6623	MC instruction not found	
	6624	MCR instruction not found	
	6625	The STL instruction is continually used 9 times or more	
	6626	Invalid instruction programmed within an STL - RET program area: MC, MCR, (I)nterrupt pointer, IRET and SRET	
	6627	RET instruction not found	
	6628	(I)nterrupt pointer, SRET and IRET incorrectly programmed within main program	
	6629	(P)ointer or (I)nterrupt pointer label not found	
	6630	SRET or IRET not found	
	6631	SRET programmed in invalid location	
	6632	IRET programmed in invalid location	

Error Detection Device	Stored Error Number	Associated Meaning	Action
D8067 **Operation error**	0000	No error	These error occur during the execution of an operation. When an operation error occurs, STOP the PLC enter programming ode and correct the fault. Note: operation errors can occur even when the syntax or circuit design is correct, e.g. D500Z is a valid statement within an FX1N PLC. But if Z had a value of 10000, the data register D10500 would be attempted to be accessed. This will cause an operation error as there is no D10500 device available.
	6701	1)No jump destination (pointer) for CJ or CALL instructions 2)(P)ointer is designated in a block that comes after the END instruction 3)An independent label is designated in a FOR-NEXT loop or a subroutine	
	6702	6 or more CALL instruction nesting levels have been used	
	6703	3 or more interrupt nesting levels have been used	
	6704	6 or more FOR - NEXT instruction nesting levels have been used	
	6705	An incompatible device has been specified as an operand for an applied instruction	
	6706	A device has been specified outside of the allowable range for an applied instruction operand	
	6707	A file register has been accessed which is outside of the users specified range	
	6708	FROM/ TO instruction error	
	6709	Other error, i.e. missing IRE/ SRET, unauthorized FOR - NEXT relationship	
D8067 **PID Operation error**	6730	Sampling time Ts (Ts<0 or >32767)	The identified parameter is specified outside of its allowable range Execution ceases PID instruction must be reset before execution will resume
	6732	Input filter value α (α<0 or >=101)	
	6733	Proportional gain K_P (K_P<0 or >32767)	
	6734	Integral time constant T_I (T_I<0 or >32767)	
	6735	Derivative gain K_D (K_D<0 or >=101)	
	6736	Derivative time constant T_D (T_D<0 or >32767)	
	6740	Sampling time TS is less than the program scan time.	TS is set to program scan time - Execution will continue.
	6742	Current value Δ exceeds its limits	Data affected resets to the nearest limit value. For all errors except 6745, this will either be a minimum of -32768 or a maximum of +32767. Execution will continue, but user should reset PID instruction.
	6743	Calculated error ε exceeds its limits	
	6744	Integral result exceeds its limits	
	6745	Derivative gain over, or differential value exceeds allowable range	
	6746	Derivative result exceeds its limits	
	6747	Total PID result exceeds its limits	
	6750	SV - PV_{nf} < 150, or system is unstable (SV - PV_{nf} has wide, fast variations)	The error fluctuation is outside the normal operation limits for the PID instruction. Execution ceases. PID instruction must be reset.
	6751	Large Overshoot of the Set Value	
	6752	Large fluctuations during Autotuning Set Process	

1	Introduction
2	Basic Program Instructions
3	STL Programming
4	Devices in Detail
5	Applied Instructions
6	Diagnostic Devices
7	Instruction Execution Times
8	PLC Device Tables
9	Assigning System Devices
10	Points of Technique
11	Index

Chapter Contents

7. Execution Times And Instructional Hierarchy..7-1
7.1 Basic Instructions 7-1
7.2 Applied Instructions 7-3
7.3 Hierarchical Relationships Of Basic Program Instructions 7-11
7.4 Batch Processing.. 7-13
7.5 Summary of Device Memory Allocations... 7-13
7.6 Limits Of Instruction Usage 7-15
7.6.1 Instructions Which Can Only Be Used Once In The Main Program Area 7-15
7.6.2 Instructions Which Are Not Suitable For Use With 110V AC Input Units 7-15

7. Execution Times And Instructional Hierarchy

FX1S	FX1N	FX2N	FX2NC

7.1 Basic Instructions

Mnemonic	Object Devices	Steps	Execution Time in μsec							
			FX1S		FX1N		FX2N		FX2NC	
			ON	OFF	ON	OFF	ON	OFF	ON	OFF
LD	X,Y,M,S,T,C and special M	1	0.7				0.08		0.08	
LDI										
AND			0.65							
ANI										
OR										
ORI										
LDP	X,Y,M,S,T,C	1	11.7	-	11.7	-	43.2		43.2	
LDF				-		-				
ANDP				-		-	37.4		37.4	
ANDF				-		-				
ORP				-		-				
ORF				-		-				
ANB	Not applicable	1	0.55				0.08		0.08	
ORB										
MPS			0.5							
MRD			0.55							
MPP			0.5							
INV										
MC	Nest level, M,Y	3	8.6	8.0	8.6	8.0	24.8	27.5	24.8	27.5
MCR	Nest level	2	4.1	-	4.1	-	20.8		20.8	
NOP	Not applicable	1	0.45				0.08		0.08	
END			450	-	450	-	508		508	
STL	S (see note 1)	1	15.8+ 8.2n	-	15.8+ 8.2n	-	27.3 + 12.6n		27.3 + 12.6n	
RET	Not applicable		4.8	-	4.8	-	21.6		21.6	

carried on over the page......

Mnemonic	Object Devices	Steps	Execution Time in μsec							
			FX1S		FX1N		FX2N		FX2NC	
			ON	OFF	ON	OFF	ON	OFF	ON	OFF
OUT	Y, M	1	0.7				0.08			
	S	2	4.4				24.4	24.3	24.4	24.3
	Special M	2	2.8				0.16			
	T-K	3	11.2	10.2	11.2	10.2	42.3	37.4	42.3	37.4
	T-D	3	12.2	11.2	12.2	11.2	42.2	37.2	42.2	37.2
	C-K (16 bit)	3	8.1	6.9	8.1	6.9	25.5	24.9	25.5	24.9
	C-D (16 bit)	3	9.5	8.0	9.5	8.0	25.3	25.0	25.3	25.0
	C-K (32 bit)	5	8.1	6.8	8.1	6.8	25.3	24.9	25.3	24.9
	C-D (32 bit)	5	9.5	8.0	9.5	8.0	25.2	24.9	25.2	24.9
SET	Y, M	1	0.85				0.08			
	S	2	4.2	2.4	4.2	2.4			23.7	17.2
	S when used in an STL step (see note 1)		18.6+ 6.8n	2.4	18.6+ 6.8n	2.4	27.3+ 12.6n	17.2	27.3+ 12.6n	17.2
	Special M	2	2.8				0.16			
RST	Y, M	1	0.85				0.08			
	S	2	3.8	2.4	3.8	2.4	23.1	17.3	23.1	17.3
	Special M	2	2.8				0.16			
	T, C	2	8.7	7.3	8.7	7.3	27	25	27	25
	D, V, Z and special D	3	3.8	1.1	3.8	1.1	21.9	17.1	21.9	17.1
PLS	Y, M	2	10.8				0.32			
PLF	Y, M	2					0.32			
P	0 TO 63	1	0.45				0.08			
I	I❑❑❑	1								

Note 1:

- “n” in the formulae to calculate the ON/OFF execution time, refers to the number of STL instructions at the current parallel/merge branch. Thus the value of “n” will fall in the range 1 to 8.

7.2 Applied Instructions

FX1S	FX1N	FX2N	FX2NC

Mnemonic	16/32 Bit	Execution Time in µsec											
		FX1S			FX1N			FX2N			FX2NC		
		ON	OFF	P	ON	OFF	P	ON	OFF	P	ON	OFF	P
00 CJ	16	7.1	1.2	✓	7.1	1.2	✓	29.0	6.4	✓	29.0	6.4	✓
01 CALL	16	9.3	3.2	✓	9.3	3.2	✓	32.2	6.4	✓	32.2	6.4	✓
02 SRET	16	8.3	-		8.3	-		21.2			21.2		
03 IRET	∗1	8.1			8.1			18.1			18.1		
04 EI	∗1	6.0			6.0			55.8			55.8		
05 DI	∗1	5.3			5.3			18.5			18.5		
06 FEND	∗1	450			450			508			508		
07 WDT	16	3.7	2.7	✓	3.7	2.7	✓	26.3	6.4	✓	26.3	6.4	✓
08 FOR	∗1	7.5			7.5			27.6			27.6		
09 NEXT	∗1	4.6			4.6			5.2			5.2		
10 CMP	16	40	2.5	✓	40	2.5	✓	87.6	6.4	✓	87.6	6.4	✓
	32	41	4.5		41	4.5		91.9	6.4		91.9	6.4	
11 ZCP	16	45	2.5	✓	45	2.5	✓	103.2	6.4	✓	103.2	6.4	✓
	32	47	4.5		47	4.5		108.9	6.4		108.9	6.4	
12 MOV	16	19	2.5	✓	19	2.5	✓	1.52	1.52	✓	1.52	1.52	✓
	32	22	3.0		22	3.0		1.84	1.84		1.84	1.84	
13 SMOV	16	Not Available						155.2	6.4	✓	155.2	6.4	✓
14 CML	16							51.4	6.4	✓	51.4	6.4	✓
	32							55.9	6.4		55.9	6.4	
15 BMOV ∗2	16	78+ 22n	2.5	✓	78+ 22n	2.5	✓	97.0+ 1.7n	6.4	✓	97.0+ 1.7n	6.4	✓
16 FMOV ∗2	16	Not Available						69.1+ 2.8n	6.4	✓	69.1+ 2.8n	6.4	✓
	32							73.2+ 5.2n	6.4		73.2+ 5.2n	6.4	
17 XCH	16							57.2	6.4	✓	57.2	6.4	✓
	32							64.0	6.4		64.0	6.4	
18 BCD	16	30	2.5	✓	30	2.5	✓	37.9	6.4	✓	37.9	6.4	✓
	32	38.6	3.0		38.6	3.0		57.6	6.4		57.6	6.4	
19 BIN	16	30	2.5	✓	30	2.5	✓	32.4	6.4	✓	32.4	6.4	✓
	32	35.5	3.0		35.5	3.0		44.5	6.4		44.5	6.4	

See end of section for ∗ notes...

Mnemonic	16/32 Bit	Execution Time in µsec											
		FX1S			FX1N			FX2N			FX2NC		
		ON	OFF	P	ON	OFF	P	ON	OFF	P	ON	OFF	P
20 ADD	16	37.5	2.5	✓	37.5	2.5	✓	27.6	6.4	✓	27.6	6.4	✓
	32	40.2	4.5		40.2	4.5		28.9	6.4		28.9	6.4	
21 SUB	16	37.5	2.5	✓	37.5	2.5	✓	27.6	6.4	✓	27.6	6.4	✓
	32	40.5	4.5		40.5	4.5		28.9	6.4		28.9	6.4	
22 MUL	16	38.2	2.5	✓	38.2	2.5	✓	25.2	6.4	✓	25.2	6.4	✓
	32	50.3	4.5		50.3	4.5		31.4	6.4		31.4	6.4	
23 DIV	16	39.2	2.5	✓	39.2	2.5	✓	32.0	6.4	✓	32.0	6.4	✓
	32	63.5	4.5		63.5	4.5		36.4	6.4		36.4	6.4	
24 INC	16	14.5	2.5	✓	14.5	2.5	✓	18.8	6.4	✓	18.8	6.4	✓
	32	16.7	4.5		16.7	4.5		20.2	6.4		20.2	6.4	
25 DEC	16	14.5	2.5	✓	14.5	2.5	✓	18.9	6.4	✓	18.9	6.4	✓
	32	16.7	4.5		16.7	4.5		20.0	6.4		20.0	6.4	
26 WAND	16	35.7	2.5	✓	35.7	2.5	✓	23.4	6.4	✓	23.4	6.4	✓
	32	37.3	4.5		37.3	4.5		24.7	6.4		24.7	6.4	
27 WOR	16	35.7	2.5	✓	35.7	2.5	✓	23.5	6.4	✓	23.5	6.4	✓
	32	37.3	4.5		37.3	4.5		24.7	6.4		24.7	6.4	
28 WXOR	16	35.7	2.5	✓	35.7	2.5	✓	23.5	6.4	✓	23.5	6.4	✓
	32	37.3	4.5		37.3	4.5		25.0	6.4		25.0	6.4	
29 NEG	16	Not Available						35.3	6.4	✓	35.3	6.4	✓
	32							38.4	6.4		38.4	6.4	
30 ROR ∗3	16							61.7	6.4	✓	61.7	6.4	✓
	32							65.3	6.4		65.3	6.4	
31 ROL ∗3	16							61.2	6.4	✓	61.2	6.4	✓
	32							65.2	6.4		65.2	6.4	
32 RCR ∗3	16							66.3+ 2.2n	6.4	✓	66.3+ 2.2n	6.4	✓
	32							69.7+ 2.6n	6.4		69.7+ 2.6n	6.4	
33 RCL ∗3	16							65.8+ 2.2n	6.4	✓	65.8+ 2.2n	6.4	✓
	32							69.5+ 2.6n	6.4		69.5+ 2.6n	6.4	
34 SFTR ∗4	16	55+ 1.25n	2.5	✓	55+ 1.25n	2.5	✓	107+ 53.8n	6.4	✓	107+ 53.8n	6.4	✓
35 SFTL ∗4	16	56.1+ 1.25n	2.5	✓	56.1+ 1.25n	2.5	✓	104.9 53.8n	6.4	✓	104.9 53.8n	6.4	✓

See end of section for ∗ notes...

Mnemonic	16/32 Bit	Execution Time in µsec											
		FX1S			FX1N			FX2N			FX2NC		
		ON	OFF	P	ON	OFF	P	ON	OFF	P	ON	OFF	P
36 WSFR ∗2	16	Not Available						126+ 11.7n	6.4	✓	126+ 11.7n	6.4	✓
37 WSFL ∗2	16							125+ 11.7n	6.4	✓	125+ 11.7n	6.4	✓
38 SFWR ∗5	16	41.6	2.5	✓	41.6	2.5	✓	83.9	6.4	✓	83.9	6.4	✓
39 SFRD ∗5	16	52.3	2.5	✓	52.3	2.5	✓	80.2	6.4	✓	80.2	6.4	✓
40 ZRST ∗6	16(D)	32.4+ 0.5n	2.5	✓	32.4+ 0.5n	2.5	✓	77+ 1.7n	6.4	✓	77+ 1.7n	6.4	✓
	16(S)	37.8+ 0.9n			37.8+ 0.9n			83+ 11.1n			83+ 11.1n		
	16(C)												
	16(T)												
	16(M)	51.8+0.8n			51.8+0.8n			89.2+ 9.4n			89.2+ 9.4n		
	16(Y)												
41 DECO	16	65.6	2.5	✓	65.6	2.5	✓	76.0	6.4	✓	76.0	6.4	✓
42 ENCO	16	46.7	2.5	✓	46.7	2.5	✓	81.8	6.4	✓	81.8	6.4	✓
43 SUM	16	Not Available						72.8	6.4	✓	72.8	6.4	✓
	32							94.6	6.4		94.6	6.4	
44 BON	16							78.2	6.4	✓	78.2	6.4	✓
	32							82.3	6.4		82.3	6.4	
45 MEAN Q7	16							83.8+ 3.4n	6.4	✓	83.8+ 3.4n	6.4	✓
	32							90.9+ 6.7n	6.4		90.9+ 6.7n	6.4	
46 ANS	16							100.8	96.2		100.8	96.2	
47 ANR	16							37.7	6.4	✓	37.7	6.4	✓
48 SQR	16							150.2	6.4	✓	150.2	6.4	✓
	32							154.8	6.4		154.8	6.4	
49 FLT	16							66.8	6.4	✓	66.8	6.4	✓
	32							66.8	6.4		66.8	6.4	
50 REF ∗8	16	19.5+ 4.3n	2.5	✓	19.5+ 4.3n	2.5	✓	99.6+ 0.6n	6.4	✓	99.6+ 0.6n	6.4	✓

See end of section for ∗ notes...

Mnemonic	16/32 Bit	Execution Time in μsec											
		FX1S			FX1N			FX2N			FX2NC		
		ON	OFF	P	ON	OFF	P	ON	OFF	P	ON	OFF	P
51 REFF ∗9	16	Not Available						65.3+ 1.7n	6.4	✓	65.3+ 1.7n	6.4	✓
52 MTR	16	22.6	9.8		22.6	9.8		39.1	23.6		39.1	23.6	
53 HSCS ∗10	32	46.8	4.5		46.8	4.5		87.8	6.4		87.8	6.4	
54 HSCR ∗10	32	46.8	4.5		46.8	4.5		88.6	6.4		88.6	6.4	
55 HSZ ∗10	32	Not Available						100.6	6.4		100.6	6.4	
56 SPD	∗1	39.5	43.8		39.5	43.8		80.2	80.2		80.2	80.2	
57 PLSY	16	82.6	22.8		82.6	22.8		85.0	73.3		85.0	73.3	
	32	100.6	34.9		100.6	34.9		86.6	75.8		86.6	75.8	
58 PWM	16	38.7	42.6		38.7	42.6		70.4	73.3		70.4	73.3	
59 PLSR	16	91.6	27.8		91.6	27.8		122.6	87.5		122.6	87.5	
	32	113.7	41.6		113.7	41.6		125.6	90.5		125.6	90.5	
60 IST	16	81.7	2.5		81.7	2.5		114.3	6.4		114.3	6.4	
61 SER ∗14	16	Not Available						129.2 +8.6n	22.9	✓	129.2 +8.6n	22.9	✓
	32							147+ 9.0n	22.9		147+ 9.0n	22.9	
62 ABSD ∗11	16	56.5+ 6.3n	2.5		56.5+ 6.3n	2.5		91.8+ 20.2n	6.4		91.8+ 20.2n	6.4	
	32	62.7+ 11n	2.5		62.7+ 11n	2.5		97.5+ 21.5n	6.4		97.5+ 21.5n	6.4	
63 INCD	16	60.5	52.7		60.5	52.7		110.5	19.5		110.5	19.5	
64 TTMR	16	Not Available						54.9	44.9		54.9	44.9	
65 STMR	16							84.4	84.4		84.4	84.4	
66 ALT	16	21.8	2.5		21.8	2.5		50.1	6.4	✓	50.1	6.4	✓
67 RAMP	16	52.5	44.8		52.5	44.8		98.1	81.6		98.1	81.6	
68 ROTC	16	Not Available						118.4	107.2		118.4	107.2	
69 SORT ∗15	16							50.5	19.5		50.5	19.5	

See end of section for ∗ notes...

Mnemonic	16/32 Bit	Execution Time in μsec											
		FX1S			FX1N			FX2N			FX2NC		
		ON	OFF	P	ON	OFF	P	ON	OFF	P	ON	OFF	P
70 TKY	16	Not Available						97.2	22.2		97.2	22.2	
	32							98.7	22.2		98.7	22.2	
71 HKY	16							92.2	27.4		92.2	27.4	
	32							65.0	6.4		65.0	6.4	
72 DSW	16	95.0	92.6		95.0	92.6		92.2	27.4		92.2	27.4	
73 SEGD	16	Not Available						65.0	6.4	✓	65.0	6.4	✓
74 SEGL	16	84.5	40.7		84.5	40.7		105.9	26.5		105.9	26.5	
75 ARWS	16	Not Available						134.4	22.1		134.4	22.1	
76 ASC	16							49.5	6.4		49.5	6.4	
77 PR	16-printing							114.8	88.5		114.8	88.5	
	16-ready							88.0			88.0		
78 FROM *12	16	87+ 483n	2.5	✓	87+ 483n	2.5	✓	97+ 487n	6.4	✓	97+ 487n	6.4	✓
	32	102+ 973n	4.5		102+ 973n	4.5		99+ 962n	6.4		99+ 962n	6.4	
79 TO *12	16	85+ 542n	2.5	✓	85+ 542n	2.5	✓	94+ 557n	6.4	✓	94+ 557n	6.4	✓
	32	98+ 1121n	4.5		98+ 1121n	4.5		96+ 1099n	6.4		96+ 1099n	6.4	
80 RS	16	56.3	9.2		56.3	9.2		117.6	18.0		117.6	18.0	
81 PRUN *13	16	46.7+ 1.0n	2.5	✓	46.7+ 1.0n	2.5	✓	65.6+ 17.0n	6.4	✓	65.6+ 17.0n	6.4	✓
	32	47.7+ 1.0n	3.0		47.7+ 1.0n	3.0		67.0+ 17.7n	6.4		67.0+ 17.7n	6.4	
82 ASCI	16	52.8+ 5.8n	2.5	✓	52.8+ 5.8n	2.5	✓	88.2+ 10.8n	6.4	✓	88.2+ 10.8n	6.4	✓
83 HEX	16	54+ 8.9n	2.5	✓	54+ 8.9n	2.5	✓	89.7+ 20.0n	6.4	✓	89.7+ 20.0n	6.4	✓
84 CCD	16	54.3+ 4.5n	2.5	✓	54.3+ 4.5n	2.5	✓	90.5+ 4.8n	6.4	✓	90.5+ 4.8n	6.4	✓
85 VRRD	16	142.7	8.9	✓	142.7	8.9	✓	209.7	27.3	✓	209.7	27.3	✓
86 VRSC	16	142.7	8.9	✓	142.7	8.9	✓	202.4	27.3	✓	202.4	27.3	✓
87	16	Function Not Available											
	32												

See end of section for ∗ notes...

Mnemonic	16/32 Bit	Execution Time in µsec											
		FX1S			FX1N			FX2N			FX2NC		
		ON	OFF	P	ON	OFF	P	ON	OFF	P	ON	OFF	P
88 PID	16	65.5	8.5		65.5	8.5		155.0	89.0		155.0	89.0	
89 USER	16	?	?	✓	?	?	✓	?	?	✓	?	?	✓
	32	?	?		?	?		?	?		?	?	
110 ECMP	32	Not Available						104.4	6.4	✓	104.4	6.4	✓
111 EZCP	32							124.5	6.4	✓	124.5	6.4	✓
118 EBCD	32	Not Available						106.9	6.4	✓	106.9	6.4	✓
119 EBIN	32							81.3	6.4	✓	81.3	6.4	✓
120 EADD	32							117.4	6.4	✓	117.4	6.4	✓
121 ESUB	32							117.4	6.4	✓	117.4	6.4	✓
122 EMUL	32							96.4	6.4	✓	96.4	6.4	✓
123 EDIV	32							100.4	6.4	✓	100.4	6.4	✓
127 ESQR	32	Not Available						152.1	6.4	✓	152.1	6.4	✓
129 INT	16							67.5	6.4	✓	67.5	6.4	✓
	32							70.4	6.4		70.4	6.4	
130 SIN	32							199.5	6.4	✓	199.5	6.4	✓
131 COS	32							262.5	6.4	✓	262.5	6.4	✓
132 TAN	32							425.3	6.4	✓	425.3	6.4	✓

See end of section for ∗ notes...

Mnemonic	16/32 Bit	Execution Time in µsec											
		FX1S			FX1N			FX2N			FX2NC		
		ON	OFF	P	ON	OFF	P	ON	OFF	P	ON	OFF	P
147 SWAP	16	Not Available						36.1	6.4	✓	36.1	6.4	✓
	32							41.2	6.4		41.2	6.4	
155 ABS	32	86.7	85.7		86.7	85.7		86.7	85.7		86.7	85.7	
156 ZRN	16	107.8	27.8		16	107.8		Not Available					
	32	130.5	40.8		32	130.5							
157 PLSV	16	79.6	22.7		16	79.6							
	32	97.8	33.5		32	97.8							
158 DRVI	16	87.7	26.8		16	87.7							
	32	110.6	40.7		32	110.6							
159 DRVA	16	89.6	26.8		16	89.6							
	32	112.7	40.7		32	112.7							
160 TCMP	16	52.6	2.5	✓	52.6	2.5	✓	134.2	6.4	✓	134.2	6.4	✓
161 TZCP	16	64.7	2.5	✓	64.7	2.5	✓	140.2	6.4	✓	140.2	6.4	✓
162 TADD	16	42.9	2.5	✓	42.9	2.5	✓	118.8	6.4	✓	118.8	6.4	✓
163 TSUB	16	42.9	2.5	✓	42.9	2.5	✓	109.4	6.4	✓	109.4	6.4	✓
166 TRD	16	29.7	2.5	✓	29.7	2.5	✓	46.2	6.4	✓	46.2	6.4	✓
167 TWR	16	633.5	2.5	✓	633.5	2.5	✓	112.0	6.4	✓	112.0	6.4	✓
169 HOUR	16	39.7	38.7		39.7	38.7		39.7	38.7		39.7	38.7	
	32	41.9	40.6		41.9	40.6		41.9	40.6		41.9	40.6	
170 GRY	16	Not Available						102.5	6.4	✓	102.5	6.4	✓
	32							107.1	6.4		107.1	6.4	
171 GBIN	16							103.4	6.4	✓	103.4	6.4	✓
	32							107.5	6.4		107.5	6.4	
176 RD3A	16	1248.3	7.5	✓	1248.3	7.5	✓	1248.3	7.5	✓	1248.3	7.5	✓
177 WR3A	16	1263.7	7.5	✓	1263.7	7.5	✓	1263.7	7.5	✓	1263.7	7.5	✓
224-230 LD❏	16	27.6	-		27.6	-		1.52			1.52		
	32	28.2	-		28.2	-		1.84			1.84		
232-238 AND❏	16	27.6	-		27.6	-		1.52			1.52		
	32	28.2	-		28.2	-		1.84			1.84		
240-246 OR❏	16	27.6	-		27.6	-		1.52			1.52		
	32	28.2	-		28.2	-		1.84			1.84		

See end of section for ∗ notes...

∗1:

- These instructions require NO preliminary contact devices such as LD, AND, OR etc.

∗2:

- Where “n” is referred to this identifies the quantity of registers to be manipulated.
  “n” can be equal or less than 512.

∗3:

- Where “n” is referred to this identifies the quantity of bit devices to be manipulated.
  “n” can be equal or less than selected operating mode, i.e. if 32 bit mode is selected then “n” can have a value equal or less than 32.

∗4:

- Where "n" is referred to this identifies the quantity of bit devices to be manipulated.
  When an FX1N PLC is used "n" can be equal or less than 1536.
  However, when an FX1S controller is used "n" can be equal or less than 512.

∗5:

- Where "n" is referred to this identifies the quantity devices to be manipulated. "n" can have any value taken from the range 2 through 512.

∗6:

- Where "n" is referred to this identifies the range of devices to be reset. The device type being reset is identified by the device letter in brackets in the '16/32 bit' column.

∗7:

- Where "n" is referred to this identifies the number of devices the mean is to be calculated from. The value of "n" can be taken from the range 1 through 64.

∗8:

- Where "n" is referred to this identifies the range of devices to be refreshed. The value of "n" is always specified in units of 8, i.e 8, 16, 24.....128. The maximum allowable range is dependent on the number of available inputs/outputs.

∗9:

- Where "n" is referred to this identifies the time setting for the input filters operation.
  "n" can be selected from the range 0 through to 60 msec.

∗10:

- There are limits to the total combined use of these instructions. For FX_{1S} and FX_{1N} there should be no more than 4 simultaneously active instructions. However, FX_{2N} and FX_{2NC} can have 6 simultaneously active instructions.

∗11:

- Where "n" is referred to this identifies the number of output points. "n" may have a value equal or less than 64.

∗12:

- Where "n" is referred to this identifies the number of words read or written FROM/TO the special function blocks.

∗13:

- Where "n" is referred to this identifies the number of octal (8 bit) words read or written when two FX PLC’s are involved in a parallel running function.

∗14:

- Where "n" is referred to this identifies the number of elements in a stack, for 16 bit operation n has a maximum of 256. However, for 32 bit operation n has a maximum of 128.

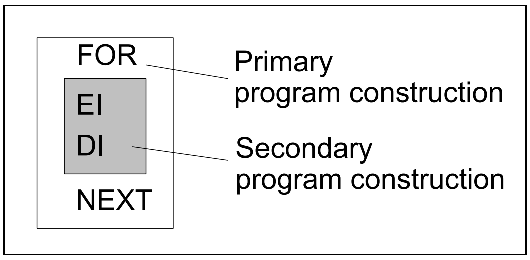
∗15:

- Where "m1" is referred to this identifies the number of elements in the data table.
  Values of m1 are taken from the range 1 to 32. For a the SORT instruction to completely process the data table the SORT instruction will be processed m1 times.

7.3 Hierarchical Relationships Of Basic Program Instructions

FX1S	FX1N	FX2N	FX2NC

The following table identifies an 'inclusive relationship'. This means the secondary program construction is included within the complete operating boundaries of the primary program construction, e.g.:



Primary Program Construction	Secondary program construction							
	MC-MCR	CJ - P	EI - DI	FOR - NEXT	STL - RET	P - SRET	I - IRET	FEND - END
MC - MCR	✓ - 8 nest levels	✓	✓	✓	✓	✗ - (6608)	✗ - (6608)	✗ - (6608)
CJ - P	✓	✓	✓	✓	✓	●	●	✗ - (6701)
EI - DI	✓	✓	✓	✓	✓	✓	✓	✓①
FOR - NEXT	✗ -(6607)	✓	✓	✓ - 5 nest levels	✗ - (6607)	✗ - (6607)	✗ - (6607)	✗ - (6607)
STL - RET	✗ - (6605)	●	✓	✓ - (within 1 STL step)	✓	✗ - (6605)	✗ - (6605)	✗ - (6605)
P - SRET	✗ - (6606)	✓	✓	✓	✗ - (6606)	✗ - (6606)	✗ - (6606)	✗ - (6709)
I - IRET	✗ - (6606)	✓	✓	✓	✗ - (6606)	✗ - (6606)	✗ - (6606)	✗ - (6606)
FEND - END	✓	✓	✓	✓	●	✓	✓	✓②
0 - FEND	✓	✓	✓	✓	✓	✗ - (6606)	✗ - (6606)	✓②
0 - END (no FEND)	✓	✓	✓	✓	✓	✗ - (6606)	✗ - (6606)	✓②

✓: Instruction combination is acceptable - for restrictions see appropriate note

✗ : Instruction combination is not allowed - bracketed number is the error code

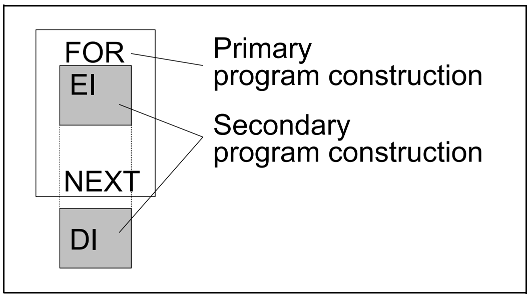
●: Instruction combination is not recommended for use even though there is no operational error

The combination of instructions with an 'inclusive relationship' is allowable. However please be aware of the following exceptions:

1) MC-MCR and STL-RET constructions cannot be used within FOR-NEXT loops, P-SRET or I-IRET subroutines.

2) Program flow may not be discontinued by using any of the following methods while inside MC-MCR, FOR-NEXT, P-SRET, I-IRET program constructions, i.e. using interrupts (I), IRET, SRET, FEND or the END instruction is not allowed.

The following table identifies an 'overlapping relationship'. This means the secondary program construction starts within the complete operating boundaries of the primary program construction but finishes outside of the primary construction, e.g.:

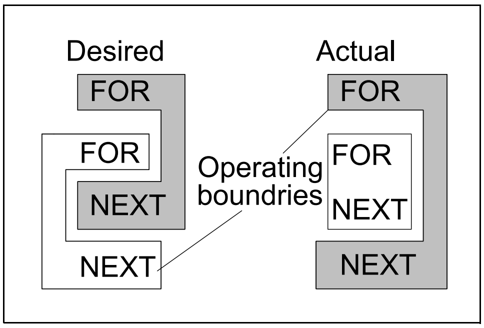


Primary Program Construction	Secondary program construction							
	MC-MCR	CJ - P	EI - DI	FOR - NEXT	STL - RET	P - SRET	I - IRET	FEND - END
MC - MCR	●	●	✓	✗ - (6607)	✗ - (6605)	✗ - (6606)	✗ - (6606)	✗ - (6608)
CJ - P	●	●	✓	●	●	●	●	✓
EI - DI	✓	✓	✓	✓	✓	✓	✓	✓
FOR - NEXT	✗ - (6607)	●	✓	✓¬	✗ - (6601)	✗ - (6607)	✗ - (6607)	✗ - (6607)
STL - RET	✗ - (6605)	●	✓	✗ - (6607)	✓	✗ - (6606)	✗ - (6606)	✗ - (6605)
P - SRET	✗ - (6608)	●	✓	✗ - (6607)	✗ - (6605)	✗ - (6606)	✗ - (6606)	✗ - (6709)
I - IRET	✗ - (6606)	●	✓	✗ - (6607)	✗ - (6606)	✗ - (6606)	✗ - (6606)	✗ - (6606)
FEND - END	✗ - (6608)	✗ - (6601)	✓①	✗ - (6607)	✗ - (6605)	✗ - (6709)	✗ - (6709)	✓②
0 - FEND	✗ - (6608)	✓	✓	✗ - (6607)	✗ - (6605)	✗ - (6709)	✗ - (6606)	✓②
0 - END (no FEND)	✗ - (6608)	✗ - (6601)	✓①	✗ - (6607)	✗ - (6605)	✗ - (6709)	✗ - (6606)	✓②

① Enters a state as if the DI instruction was missing. An error is not generated.

② The first occurrence of either an FEND or the END instruction takes priority. This would then end the program scan prematurely.

③ The sequence will not process as expected, e.g.:

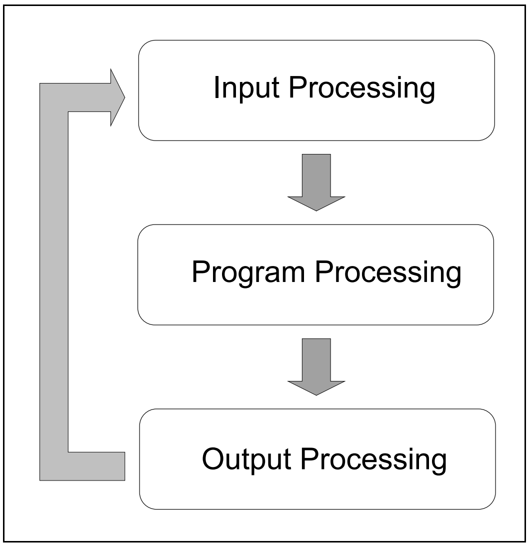


7.4 Batch Processing

FX1S	FX1N	FX2N	FX2NC

This is the system used by all members of the FX family of PLC's. The basic concept is that there are three stages to any program scan. In other words, every time the program is processed form start to end the following sequence of events occurs:

Input processing:



All of the current input statuses are read in to a temporary memory area; sometimes called an image memory. The PLC is now ready for the next program processing.......

Program processing:

All of the updated inputs are checked as the program is processed. If the new input statuses change the status of driven outputs, then these are noted in the image memory for the......

Output processing:

The new, current statuses of the outputs which have just be processed are physically updated, i.e relays are turned ON or OFF as required. The program scan starts again............

The system is known as 'Batch processing' because all of the inputs, program operation and finally the outputs are processed as batches.

7.5 Summary of Device Memory Allocations

The memory allocations of the programmable are very complex, but from a users point of view there are three main areas:

a) The Program Memory:

This memory area holds all of the data regarding: parameters, sequence program, constant values K and H, pointer information for P and I devices, nest level information, file register contents/allocations and also the program comment area.

- This memory area is latched either by battery backup or by use of EEPROM program management (dependent on the PLC being used). Any data stored in this area is kept even when the PLC is powered down. The duration and reliability of the data storage is dependent upon the condition of the battery or EEPROM being used to perform the backup process.

b) Data Memory

This memory area contains, as the title suggests, all of the data values associated with: data registers (normal and special), Index registers, current timer values, retentive timer values (if available) and current counter values.

- All of the devices which are designated as being latched (including retentive timers) are backed up in a similar method to the one mentioned under point a).
- Index registers and special data registers (D8000 to D8255) operate in the specified manner under the following circumstances:

Circumstance	Reaction
PLC's power is turned OFF	All data is cleared
PLC's power is turned ON	Certain devices are reset to their defaults see chapter 6
PLC is switched from STOP to RUN	Certain devices are reset to their defaults see chapter 6
PLC is switched from RUN to STOP	

- All other devices such as current values of non latched data registers, timers and counters behave in the following manner:

Circumstance	Reaction
PLC's power is turned OFF	All data is cleared
PLC's power is turned ON	
PLC is switched from STOP to RUN	No change
PLC is switched from RUN to STOP	Cleared (unless special M coil M8033 is active)

c) Bit Memory

This memory area contains the contact status of all inputs, outputs, auxiliary relays, state coils, timers and counters.

- All of the devices which are designated as being latched (including retentive timers) are backed up in a similar method to the one mentioned under point a).
- Special auxiliary relays (M8000 to M8255) act in a similar way to the special data registers mentioned under point b).
- All other devices are subject to the same changes as the current values of data registers, timers and counter (see the last point and table under section b).

Summary

<table>
<tr><th rowspan="2">Memory type</th><th colspan="2">Power</th><th colspan="2">PLC</th></tr>
<tr><th>OFF</th><th>OFF ➢ ON</th><th>STOP ➢ RUN</th><th>RUN ➢STOP</th></tr>
<tr><td>All devices backed by battery</td><td colspan="4">Not changed</td></tr>
<tr><td>Special M and D devices (8000 to 8255) and index registers V and Z</td><td>Cleared</td><td>Default</td><td colspan="2">Not changed</td></tr>
<tr><td rowspan="2">All other devices</td><td colspan="2" rowspan="2">Cleared</td><td>Not changed</td><td>Cleared</td></tr>
<tr><td colspan="2">Not changed when M8033 is set</td></tr>
</table>

7.6 Limits Of Instruction Usage

FX1S	FX1N	FX2N	FX2NC

7.6.1 Instructions Which Can Only Be Used Once In The Main Program Area

The following instructions can only be used once in the main program area. For PLC applicability please check either the detailed explanations of the instructions or the instruction execution tables list earlier.

- Instructions which can only be used once are:

FNC 52 MTR	FNC 60 IST	FNC 70 TKY
FNC 57 PLSY	FNC 61 SORT	FNC 71 HKY
FNC 58 PWM	FNC 62 ABSD	FNC 72 DSW
FNC 59 PLSR	FNC 63 INCD	FNC 74 SEGL
	FNC 68 ROTC	FNC 75 ARWS

- Only one of either FNC 57 PLSY or FNC 59 PLSR can be programmed at once.
  Both instructions can not be present in the same active program.

7.6.2 Instructions Which Are Not Suitable For Use With 110V AC Input Units

FX1S	FX1N	FX2N	FX2NC

When using 110V AC input units certain operations, functions and instructions are not recommended for use due to long energize/de-energize (ON/OFF) times of the 110V input devices.

- Program operations not recommended for use are:
  - Interrupt routines
  - High speed counters
- Instructions not recommended for use are:

FNC 51 REFF	FNC 68 ROTC	FNC 72 DSW
FNC 52 MTR	FNC 70 TKY	FNC 75 ARWS
FNC 56 SPD	FNC 71 HKY	

MEMO

1	Introduction
2	Basic Program Instructions
3	STL Programming
4	Devices in Detail
5	Applied Instructions
6	Diagnostic Devices
7	Instruction Execution Times
8	PLC Device Tables
9	Assigning System Devices
10	Points of Technique
11	Index

Chapter Contents

8. PC Device Tables..8-1
8.1 Performance Specification Of the FX_{1S} 8-1
8.2 Performance Specification Of The FX_{1N} 8-2
8.3 Performance Specification Of The FX_{2N} and FX_{2NC} PLC’s 8-4

8. PLC Device Tables

FX1S	FX1N	FX2N	FX2NC

8.1 Performance Specification Of The FX1S

Item		Specification	Remarks
Operation control method		Cyclic operation by stored program	
I/O control method		Batch processing method (when END instruction is executed)	I/O refresh instruction is available
Operation processing time		Basic instructions: 0.55 to 0.7 µs Applied instructions: 1.65 to several 100 µs	
Programming language		Relay symbolic language + step ladder	Step ladder can be used to produce an SFC style program
Program capacity		2K steps	Provided by built in EEPROM memory
Number of instructions		Basic sequence instructions: 29 Step ladder instructions: 2 Applied instructions: 85	A Maximum 116 applied instructions are available including all variations
I/O configuration		Max total I/O set by Main Processing Unit	
Auxiliary relay (M coils)	General	384 points	M0 to M383
	Latched	128 points (subset)	M384 to M511
	Special	256 points	From the range M8000 to M8255
State relays (S coils)	General	128 points	S0 to S127
	Initial	10 points (subset)	S0 to S9
Timers (T)	100 msec	Range: 0 to 3,276.7 sec 63 points	T0 to T55
	10 msec	Range: 0 to 327.67 sec 31 points	T32 to T62 when special M coil M8028 is driven ON
	1 msec	Range: 0.001 to 32.767 sec 1 point	T63
Counters (C)	General	Range: 1 to 32,767 counts 16 points	C0 to C15 Type: 16 bit up counter
	Latched	16 points(subset)	C16 to C31 Type: 16 bit up counter
High speed counters (C)	1 phase	Range: -2,147,483,648 to +2,147,483,647 counts FX0: Select upto four 1 phase counters with a combined counting frequency of 5kHz or less. Alternatively select one 2 phase or A/B phase counter with a counting frequency of 2kHz or less. FX0S: When multiple 1-phase counters are used the sum of the frequencies must be equal or less than 14kHz. Only 1, 2 phase high speed counter may be used at any one time. When 2 phase counters are in use the maximum counted speeds must be equal or less than 14kHz, calculated as (2 ph counter speed 5 number of counted edges) + 1 ph counter speeds.	C235 to C240 (note C235 is latched) 6 points
	1 phase c/w start stop input		C241(latched), C242 and C244 (latched) 3 points
	2 phase		C246, C247 and C249 (all latched) 3 points
	A/B phase		C251, C252 and C254 (all latched) 3 points

continued over the page....

Item		Specification	Remarks
Data registers (D)	General	128 points	D0 to D127 Type:16 bit data storage register pair for 32 bit device
	Latched	128 points (subset)	D128 to D255 Type:16 bit data storage register pair for 32 bit device
	Externally adjusted	Range: 0 to 255 2 points	D8013 or D8030 & D8031 Data is entered indirectly through the external setting potentiometer
	Special	256 points (inclusive of D8013)	From the range D8000 to D8255 Type: 16 bit data storage register
	Index	16 points	V and Z Type: 16 bit data storage register
Pointers (P)	For use with CALL	64 points	P0 to P63
	For use with interrupts	6 points	I00❑ to I30❑ (rising trigger ❑ = 1, falling trigger ❑ = 0)
Nest levels		8 points for use with MC and MCR	N0 to N7
Constants	Decimal K	16 bit: -32,768 to +32,767 32 bit: -2,147,483,648 to +2,147,483,647	
	Hexadeci-mal H	16 bit: 0000 to FFFF 32 bit: 00000000 to FFFFFFFF	

8.2 Performance Specification Of The FX1N

Item		Specification	Remarks
Operation control method		Cyclic operation by stored program	
I/O control method		Batch processing method (when END instruction is executed)	I/O refresh instruction is available
Operation processing time		Basic instructions: 0.55 to 0.7 µs Applied instructions: 1.65 to several 100 µs	
Programming language		Relay symbolic language + step ladder	Step ladder can be used to produce an SFC style program
Program capacity		8K steps	Provided by built in EEPROM memory
Number of instructions		Basic sequence instructions: 29 Step ladder instructions: 2 Applied instructions: 89	A Maximum 120 applied instructions are available including all variations
I/O configuration		Max hardware I/O configuration points 128, dependent on user selection (Max. software addressable Inputs 128, Outputs 128)	
Auxiliary relay (M coils)	General	384 points	M0 to M383
	Latched	1152 points (subset)	M384 to M1535
	Special	256 points	From the range M8000 to M8255

continued over the page....

Item		Specification	Remarks
State relays (S coils)	Latched	1000 points	S0 to S999
	Initial	10 points (subset)	S0 to S9
Timers (T)	100 msec	Range: 0 to 3,276.7 sec 200 points	T0 to T199
	10 msec	Range: 0 to 327.67 sec 46 points	T200 to T245
	1 msec	Range: 0 to 32.767 sec 4 point	T246 to T249
	100 msec retentive	Range: 0 to 3,276.7 sec 6 points	T250 to T255
Counters (C)	General	Range: 1 to 32,767 counts 16 points	C0 to C15 Type: 16 bit up counter
	Latched	184 points (subset)	C16 to C199 Type: 16 bit up counter
	General	Range: 1 to 32,767 counts 20 points	C200 to C219 Type: 32 bit bi-directional counter
	Latched	15 points (subset)	C220 to C234 Type: 32 bit bi-directional counter
High speed counters (C)	1 phase	Range: -2,147,483,648 to +2,147,483,647 counts Select upto four 1 phase counters with a combined counting frequency of 5kHz or less. Alternatively select one 2 phase or A/B phase counter with a counting fre-quency of 2kHz or less. Note all counters are latched	C235 to C240 6 points
	1 phase c/w start stop input		C241, C242 and C244 3 points
	2 phase		C246, C247 and C249 3 points
	A/B phase		C251, C252 and C254 3 points
Data registers (D)	General	7128 points	D0 to D127 & D1000 to D7999 Type: 16 bit data storage register pair for 32 bit device
	Latched	872 points (subset)	D128 to D999 Type: 16 bit data storage register pair for 32 bit device
	File	7000 points	D1000 to D6999 set by parameter in 3 blocks of 500 program steps Type: 16 bit data storage register
	Externally adjusted	Range: 0 to 255 2 points	Data is move from external setting potentiometers to registers D8030 and D8031)
	Special	256 points (inclusive of D8013, D8030 and D8031)	From the range D8000 to D8255 Type: 16 bit data storage register
	Index	16 points	V and Z Type: 16 bit data storage register
Pointers (P)	For use with CALL	128 points	P0 to P127
	For use with interrupts	6 points	I00❑ to I30❑ (rising trigger ❑ = 1, falling trigger ❑ = 0)
Nest levels		8 points for use with MC and MCR	N0 to N7
Constants	Decimal K	16 bit: -32,768 to +32,767 32 bit: -2,147,483,648 to +2,147,483,647	
	Hexadeci-mal H	16 bit: 0000 to FFFF 32 bit: 00000000 to FFFFFFFF	

8.3 Performance Specification Of The FX2N and the FX2NC PLC's

Item		Specification	Remarks
Operation control method		Cyclic operation by stored program	
I/O control method		Batch processing method (when END instruction is executed)	I/O refresh instruction is available
Operation processing time		Basic instructions: 0.08 µs Applied instructions: 1.52 to several 100 µs	
Programming language		Relay symbolic language + step ladder	Step ladder can be used to produce an SFC style program
Program capacity		8000 steps built in	Expandable to 16000 steps using additional memory cassette
Number of instructions		Basic sequence instructions: 20 Step ladder instructions: 2 Applied instructions: 125	A Maximum 125 applied instructions are available
I/O configuration		Max hardware I/O configuration points 255, dependent on user selection (Max. software addressable Inputs 255, Outputs 255)	
Auxiliary relay (M coils)	General	3072 points	M0 to M3071
	Latched	2572 points (subset)	M500 to M3071
	Special	256 points	From the range M8000 to M8255
State relays (S coils)	General	1000 points	S0 to S999
	Latched	500 points (subset)	S500 to S999
	Initial	10 points (subset)	S0 to S9
	Annunciator	100 points	S900 to S999
Timers (T)	100 msec	Range: 0 to 3,276.7 sec 200 points	T0 to T199
	10 msec	Range: 0 to 327.67 sec 46 points	T200 to T245
	1 msec retentive	Range: 0 to 32.767 sec 4 points	T246 to T249
	100 msec retentive	Range: 0 to 3,276.7 sec 6 points	T250 to T255
Counters (C)	General 16 bit	Range: 1 to 32,767 counts 200 points	C0 to C199 Type: 16 bit up counter
	Latched 16 bit	100 points (subset)	C100 to C199 Type: 16 bit up counter
	General 32 bit	Range: -2,147,483,648 to 2,147,483,647 35 points	C200 to C234 Type: 32 bit up/down counter
	Latched 32 bit	15 points (subset)	C219 to C234 Type: 16 bit up/down counter

Item		Specification	Remarks
High speed counters (C)	1 phase	Range: -2,147,483,648 to +2,147,483,647 counts General rule: Select counter combinations with a combined counting frequency of 20kHz or less. Note all counters are latched	C235 to C240 6 points
	1 phase c/w start stop input		C241 to C245 5 points
	2 phase		C246 to C250 5 points
	A/B phase		C251 to C255 5 points
Data registers (D)	General	8000 points	D0 to D7999 Type: 16 bit data storage register pair for 32 bit device
	Latched	7800 points (subset)	D200 to D7999 Type: 16 bit data storage register pair for 32 bit device
	File registers	7000 points	D1000 to D7999 set by parameter in 14 blocks of 500 program steps Type: 16 bit data storage register
	Special	256 points	From the range D8000 to D8255 Type: 16 bit data storage register
	Index	16 points	V0 to V7 and Z0 to Z7 Type: 16 bit data storage register
Pointers (P)	For use with CALL	128 points	P0 to P127
	For use with interrupts	6 input points, 3 timers, 6 counters	I00❑ to I50❑ and I6✰✰ to I8✰✰ (rising trigger ❑=1, falling trigger ❑=0, ✰✰=time in msec)
Nest levels		8 points for use with MC and MCR	N0 to N7
Numbers	Decimal K	16 bit: -32,768 to +32,767 32 bit: -2,147,483,648 to +2,147,483,647	
	Hexadecimal H	16 bit: 0000 to FFFF 32 bit: 00000000 to FFFFFFFF	
	Floating Point	32 bit: 0, $\pm 1.175 \times 10^{-38}$, $\pm 3.403 \times 10^{38}$ (Not directly enterable)	

Memo

1	Introduction
2	Basic Program Instructions
3	STL Programming
4	Devices in Detail
5	Applied Instructions
6	Diagnostic Devices
7	Instruction Execution Times
8	PLC Device Tables
9	Assigning System Devices
10	Points of Technique
11	Index

Chapter Contents

9. Assigning System Devices 9-1
9.1 Addressing Extension Modules 9-1
9.2 Real Time Clock Function 9-2
9.2.1 Setting the real time clock 9-2

9. Assigning System Devices

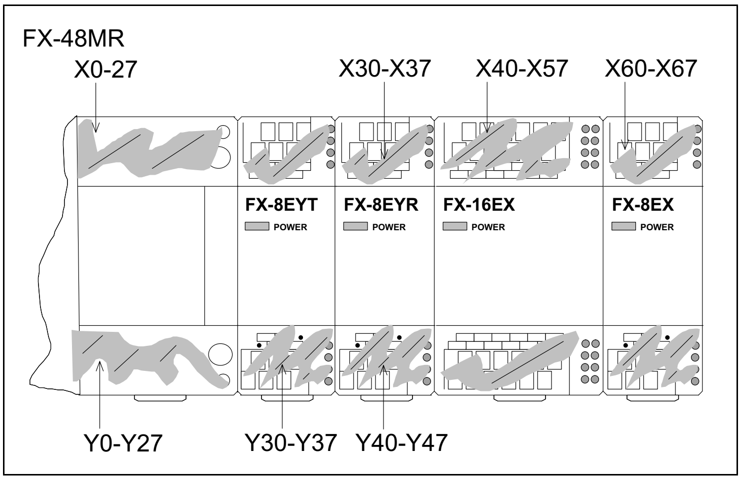
FX1S	FX1N	FX2N	FX2NC

9.1 Addressing Extension Modules

Most of the FX family of PLC's have the ability to connect additional discreet I/O and/or special function modules. To benefit from these additional units the user must address each block independently.

Addressing Additional Discrete I/O

This type of I/O is the standard input and output modules. As each extension block or powered extension unit is added to the system they assume the next available addresses. Hence, the units closest to the base unit will have the lowest I/O numbers or addresses. I/O numbers are always counted in octal. This means from 0 to 7 and 10 to 17 etc. Within a users program the additional addresses are used as normal. Discreet I/O can be added at the users discretion as long as the rules of system configuration for each PLC type are obeyed. This information can be found in the appropriate hardware manual.



For easy use and identification, each additional I/O unit should be labeled with the appropriate I/O numbers using the provided number labels.

Caution when using an FX system with FX-8ER, FX-24MR units

- When an FX-8ER or an FX-24MR are used an additional 8 points (as 4 inputs, 4 outputs) of I/O must be allowed for. This is because both units split blocks of 8 inputs and 8 outputs to obtain a physical 4 input/ 4 output configuration. Hence, an FX-8ER unit actually occupies 8 input points and 8 output points even though there are only 4 physical inputs and 4 physical outputs.

Addressing Special Function Blocks

Special function blocks are allocated a logical 'station/block number' from 0 to 7. This is used by the FROM/TO instructions to directly access each independent special function module. The lower the 'station/block number' is, the closer to the base unit it can be found. Special function blocks can be added at the users discretion but the rules of configuration for each type of PLC must be obeyed at all times. The configuration notes can be found in the appropriate hardware manual for each programmable controller.

9.2 Real Time Clock Function

FX1S | FX1N | FX2N | FX2NC

The time data of a RTC cassette or chip (built in to FX1S and FX1N) is battery backed. This means when the PLC is turned OFF the time data and settings are not lost or corrupted. The duration or storage life of the timedatails dependent upon the condition of the battery.
The real time clock has a worst case accuracy of ± 45 seconds per month at an ambient temperature of 25°C. The calendar function of the RTC caters for leap years during the period 1980 through 2079.

9.2.1 Setting the real time clock

FX1S | FX1N | FX2N | FX2NC

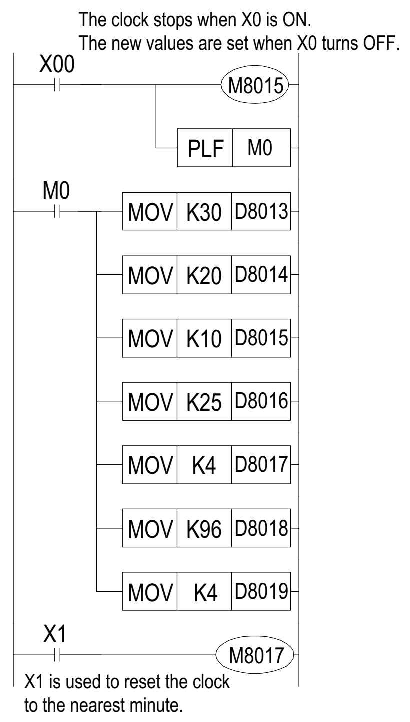
The RTC can be set using the special data registers and control flags as follows:

Device Number	Function	Range
D8013	Seconds	0 to 59
D8014	Minutes	0 to 59
D8015	Hours	0 to 23
D8016	Date	1 to 31 (correct for current Month)
D8017	Month	1 to 12
D8018	Year	00 to 99 (1980 to 2079)
D8019	Day of Week	0 to 6 (Sunday to Sat-urday

Device Number	Comments
M8015 Time setting	Set ON to stop the clock. When the clock is stopped the time values can be reset. The clock restarts when the flag is reset to OFF.
M8016 Register Hold	The clock data in the data regis-ters is held. The clock still runs. Use this to pause the data to read the current time.
M8017 Minute Rounding	When on rounds the time up or down to the nearest minute.
M8018 Clock Available	Automatically set to indicate the RTC is available.
M8019 Setting Error	ON when the values for the RTC are out of range.

These devices are used as shown in the program on the right.

Note: The FX2N and FX2NC has special instructions that simplify the setting and use of the RTC.
See section 5.14 for more details.



9.3 Analog Expansion Boards

The FX1N expansion boards can be installed on the FX1S/1N Series PLCs to provide extra analog I/O channels. Please see the respective expansion board User's Manual for more information on configuration and hardware specifications.

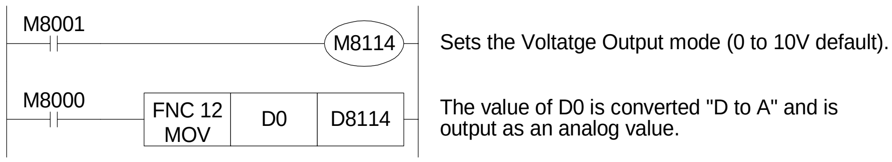
The expansion boards are not equipped with a Gain/Offset setting so that these values must be calculated in the PLC ladder program. Example programs are provided below.

9.3.1 FX1N-1DA-BD

This expansion board is used to convert a digital value in the range of 0 ~ 4000 that is stored in D8114 to an analog output value. The analog output can be in the Voltage range of 0-10 Volts DC or 4-20mA.

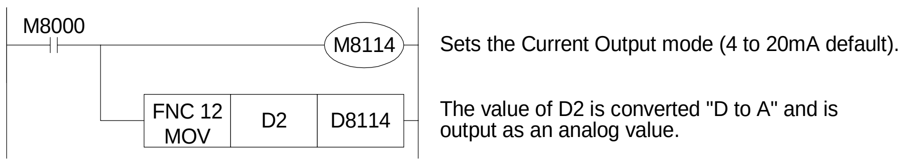
Voltage Output Mode

The following program example sets the Voltage Output mode. A digital value in D0 is converted to the analog equivalent for output.



Current Output Mode

The following program example sets the Current Output mode. A digital value in D0 is converted to the analog equivalent for output.



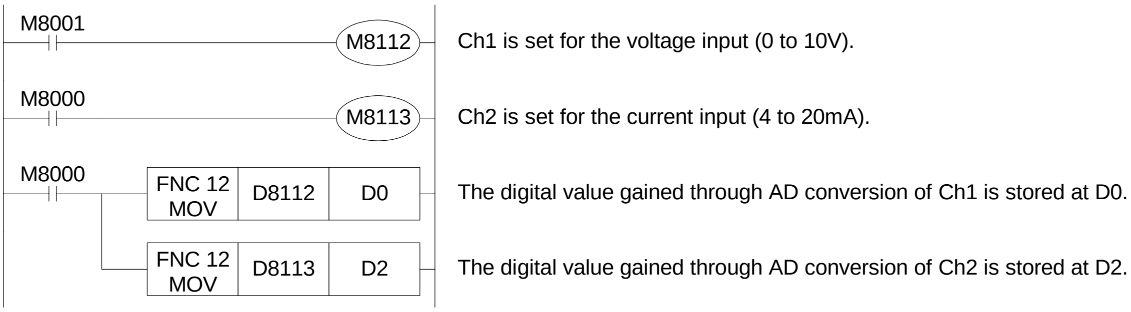
Example Application Programs

The user can use any digital value range that is convenient in the program but must convert the value to the 0 ~ 4000 range before the correct analog value can be output. In the same way, the analog outputs can be modified via PLC programming to give outputs within a certain range. Please note that outputs outside the given range are not possible.

The Please see programming examples below.

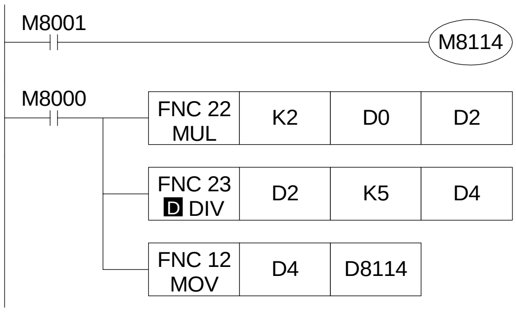
Example Application Program #1

Output an analog value in the range of 0 to 10 Volts when the digital value in the user program is 0 ~ 10000.



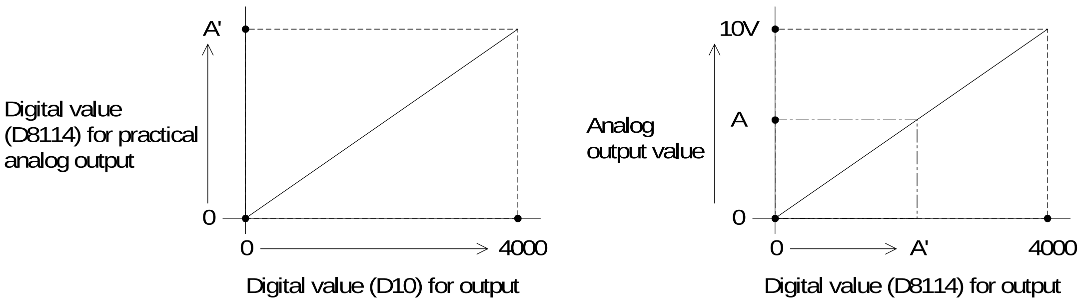
D0 ranges from 0 ~10000. To convert D0 to the 0 ~ 4000 value needed for D8114:

D8114 = [D0 x 4000] / 10000 or [D x 2] /5



Example Application Program #2

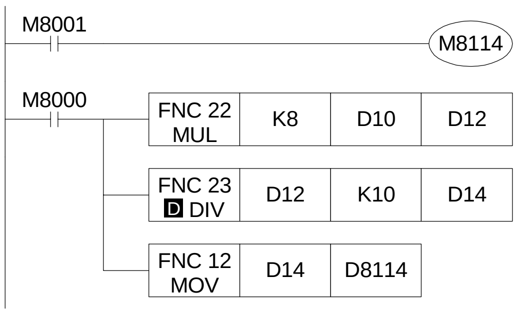
An output of 0 ~ A [0 < A < 10] is desired in the program that is using a digital range of 0~4000 that is stored in register D10.



Because A is smaller than 10 Volts, the digital value of 0~4000 must be converted to a value of 0~A' as shown in the graphs above. 4000/10V = A'/A or A' = [4000/10] x A = 400 x A

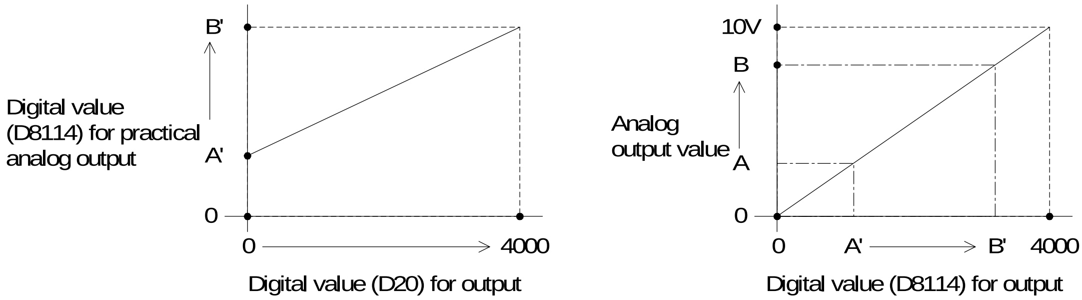
D8114 = [A'] x (D10 / 4000) = [400 x A] x [D10 / 4000) = (A x D10) / 10.

If A = 8



Example Application Program #3

The desired analog output is from values A to B where 0 < A < B < 10 and the digital values range from 0 ~ 4000 in D20.



This example is equivalent to setting an offset and gain for the analog output.

The digital values must be converted to A’ and B’ per the graphs above.

[B - A] / [10 - 0] = [B’ - A’] / [4000 - 0], therefore [B’ - A’] = [B - A] x 400.

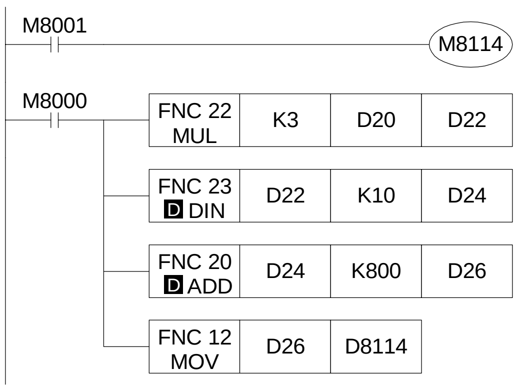
D8114 = [B’ - A’] x (D20 / 4000) + A’

B’ = 400 x B and A’ = 400 x A (see previous example programs for calculation)

D8114 = [400 x (B - A)/4000] x D20 + (400 x A)

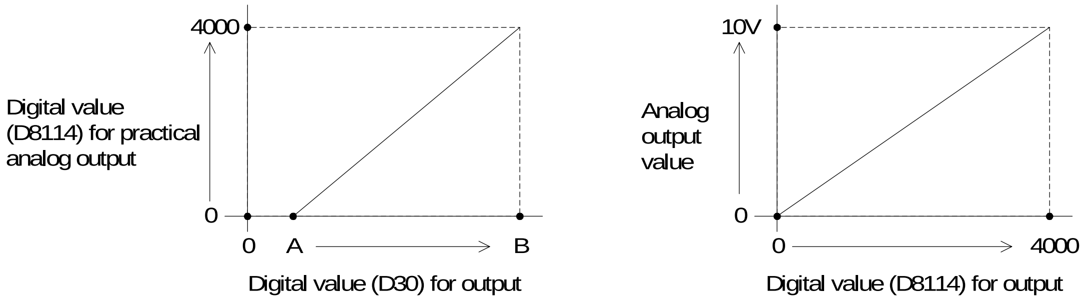
D8114 = [(B-A)/10] x D20 + (400 x A)

If A = 2 and B = 5, see the programming example below



Example Application Program #4

In Voltage Output Mode, a digital range of values A ~ B is used in the program for an analog output of 0 ~ 10 Volts. The digital range of A ~ B stored in D30 must be converted to 0 ~ 4000 before the correct analog value can be output.

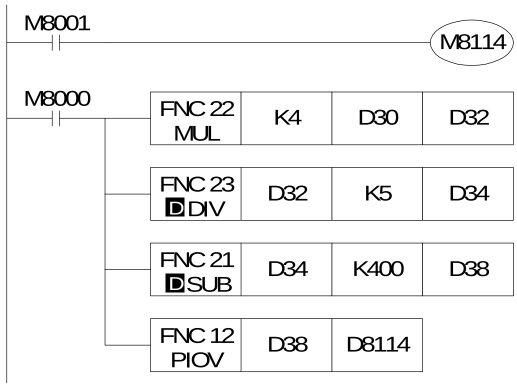


[(4000 - 0) / (B-A)] = D8114 / (D30 - A)

D8114 = [4000 x D30 / (B - A)] - [(4000 x A) / (B - A)]

If A = 500 and B = 5500, then

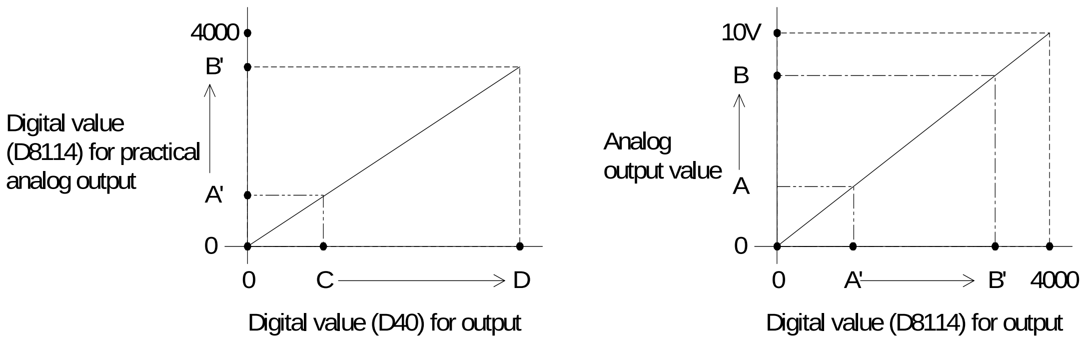
D8114 = (4/5) X D30 - 400



Example Application Program #5

If using a digital range of C ~ D in the program to output an analog value of A ~ B, the digital value must be converted to the 0 ~ 4000 equivalent and the analog value must be converted to 0 ~ 10 Volt equivalent.

Digital Values for conversion to analog are stored in D8114.



Please see prior programming examples for sample equations for the conversion of data ranges.

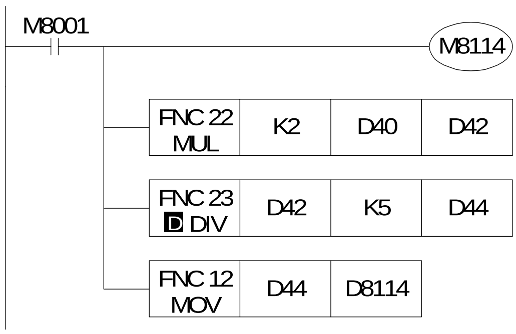
D8114 = [(B'-A') x D40] / (D-C) + [(A' x D) - (B' x C) / (D - C)

D8114 = [(400 x B - 400 x A) x D20] / (D-C) + [(400 x A x D) - (400 x B x C)] / (D - C) (from prior examples A' = 400 x A and B' = 400 x B

D8114 = [400 x (B - A)] / (D - C) + 400 x [(A x D) - (B x C)] / (D - C)

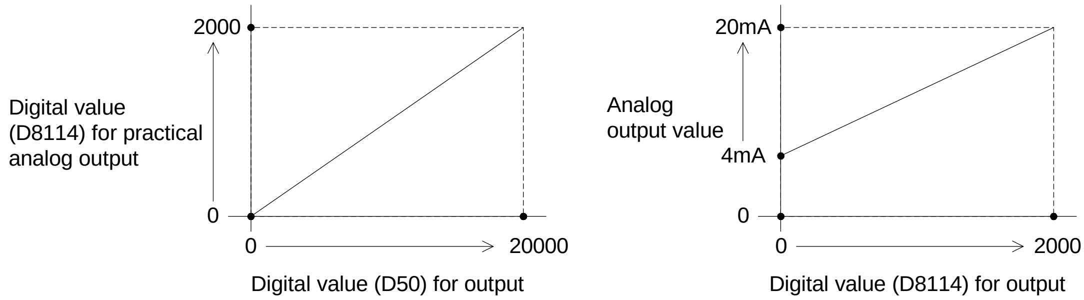
If A = 1, B = 5.5, C = 1000, and D = 5500, then

D8114 = (2 x D40) / 5



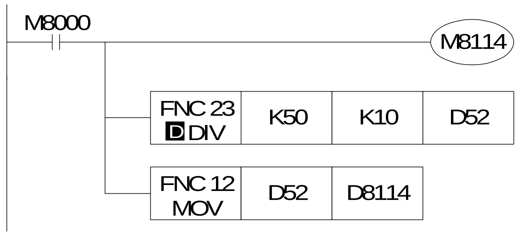
Example Application Program #6

In the Current Output Mode, the 1DA converts values from 0 ~ 2000 to the analog output of 4 ~ 20 mA. If using a digital range of 0 ~ 20000 in the program, the range must be converted to 0 ~ 2000 as shown in the programming example below. Digital values for conversion to analog are stored in D8114.



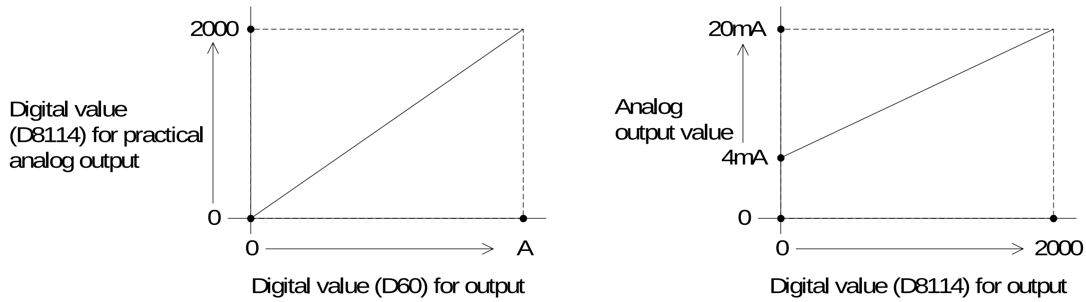
D8114 = [(2000 - 0) x D50] / (20000 - 0)

D8114 = D50 / 10



Example Application Program #7

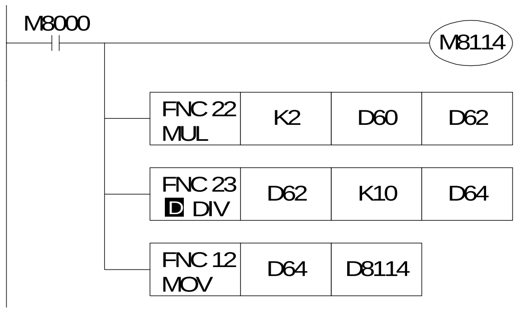
In Current Output Mode, a user wants to use a range of 0 ~ A in the program to output the analog current of 4 ~ 20mA. The user range 0 ~ A stored in D60 must be converted to the range of 0 ~ 2000 as shown below.



D8114 = [(2000 -0) x D60] / (A - 0)

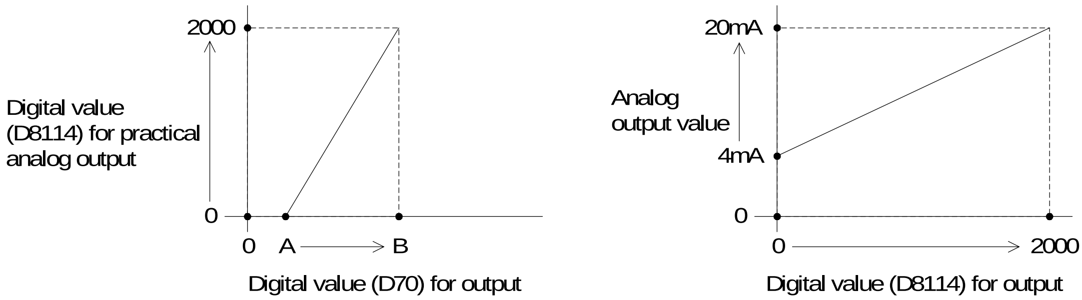
D8114 = (2000 x D60) / A, if A = 10000

D8114 = D60 / 5



Example Application Program #8

In Current Output mode, the user digital range of A ~ B is used to output a current of 4 - 20 mA. The range of A ~ B stored in D70 must be converted to a range of 0 ~ 2000 per the example program below.

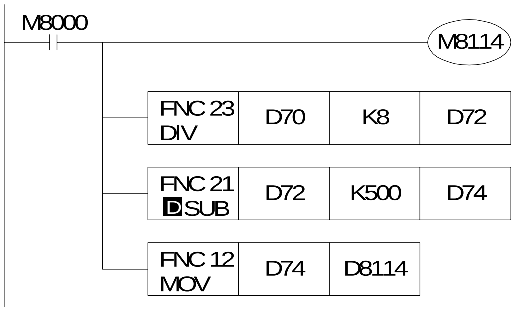


D8114/(D70 - A) = (2000 - 0)/ (B - A)

D8114 = {[(2000 - 0) x D70] / (B - A)} - {[(2000 -0) x A] / (B - A)}

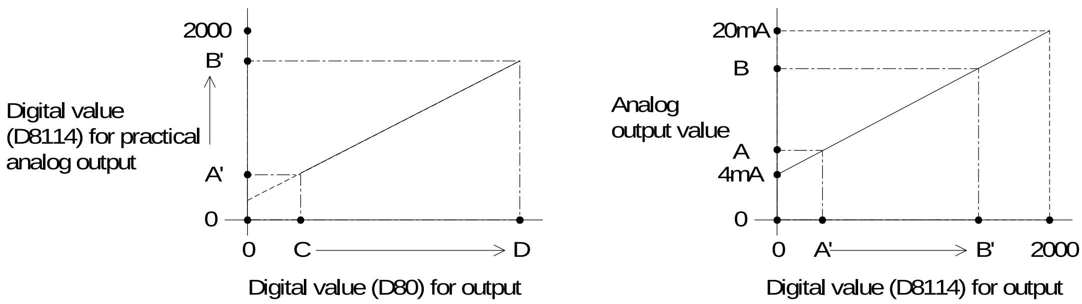
If A = 4000 and B = 20000, then [(2000 x D70 /(20000 - 4000)] - [2000 x 4000 / (20000 - 4000)]

D8114 = (D70 / 8) - 500



Example Application Program #9

In Current Output mode, a current in the range of A ~ B (4mA < A < B < 20 mA) is output by using a digital range of C ~ D that is stored in D80. The current range A ~ B must be converted to the 4 ~ 20mA equivalent value and the digital range C ~ D must be converted to the 0 ~ 2000 range equivalent value.



Please see previous programming examples for sample range conversion calculations.

D8114 = (B' - A') x D80 / (D - C) + {(A' x D) - (B' x C)} / (D - C)

A' = 125 x A - 500, B' = 125 x B - 500,

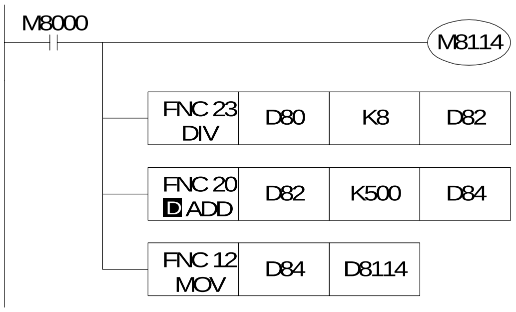
D8114 = [(125 x B - 500) - (125 x A - 500)] x D80 / (D - C) +

[(125 x A - 500) x D - (125 x B -500) x C] / (D - C)

If A = 5, B = 15, C = 5000, and D = 15000

= [125 x (15 - 5)] x D80 / (15000 - 5000) + 125 x [(5-4) x 15000 - (15-4) x 5000]/ (15000 - 5000)

D8114 = (D80 / 8) - 500

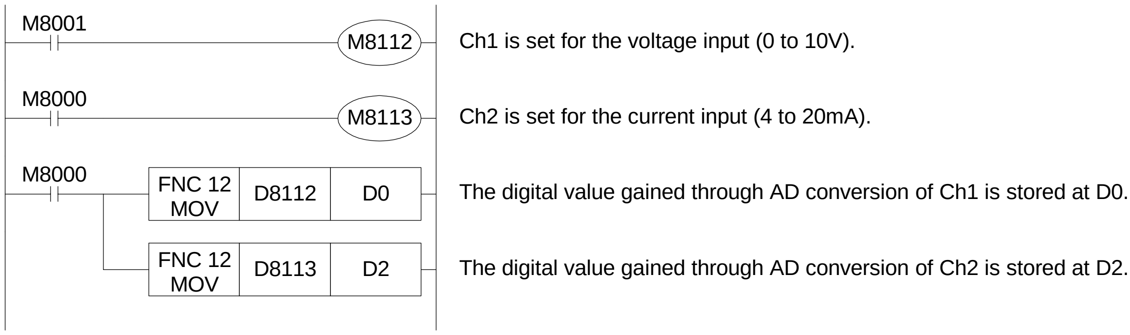


9.3.2 FX1N-2AD-BD

This expansion board is used to convert up to two channels of analog input into digital values for use by the FX1S/1N Series PLCs. Voltage input (0 ~ 10 Volts) or Current input (4 to 20 mA) for analog to digital conversion can be set by switching the auxiliary relays assigned to each channel. The output values can be adjusted after the conversion via PLC program code but resolution cannot be improved.

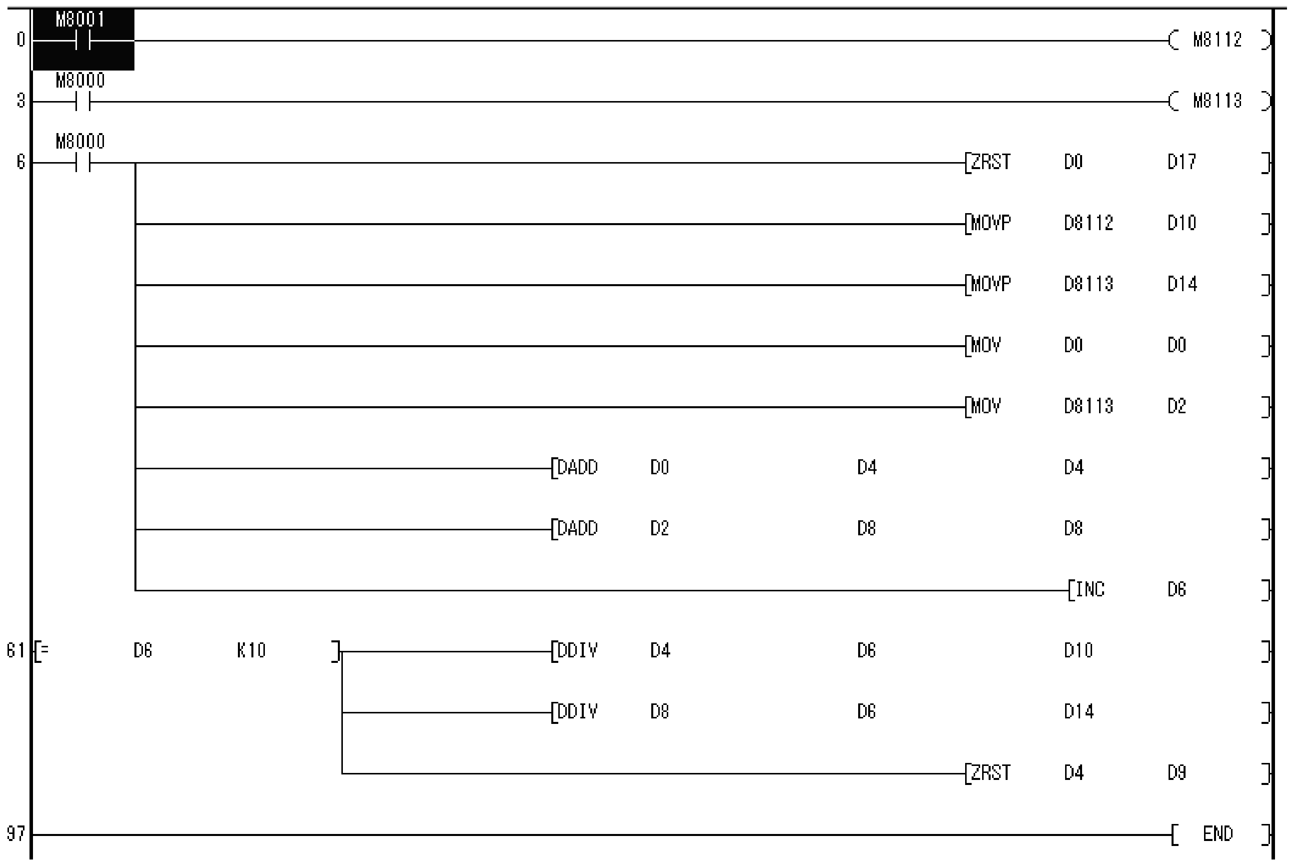
Basic Program #1

The following program sets Channel 1 in the Voltage Input mode and Channel 2 in the Current Input mode with the A/D converted digital value of each channel stored in D0 and D2 respectively.



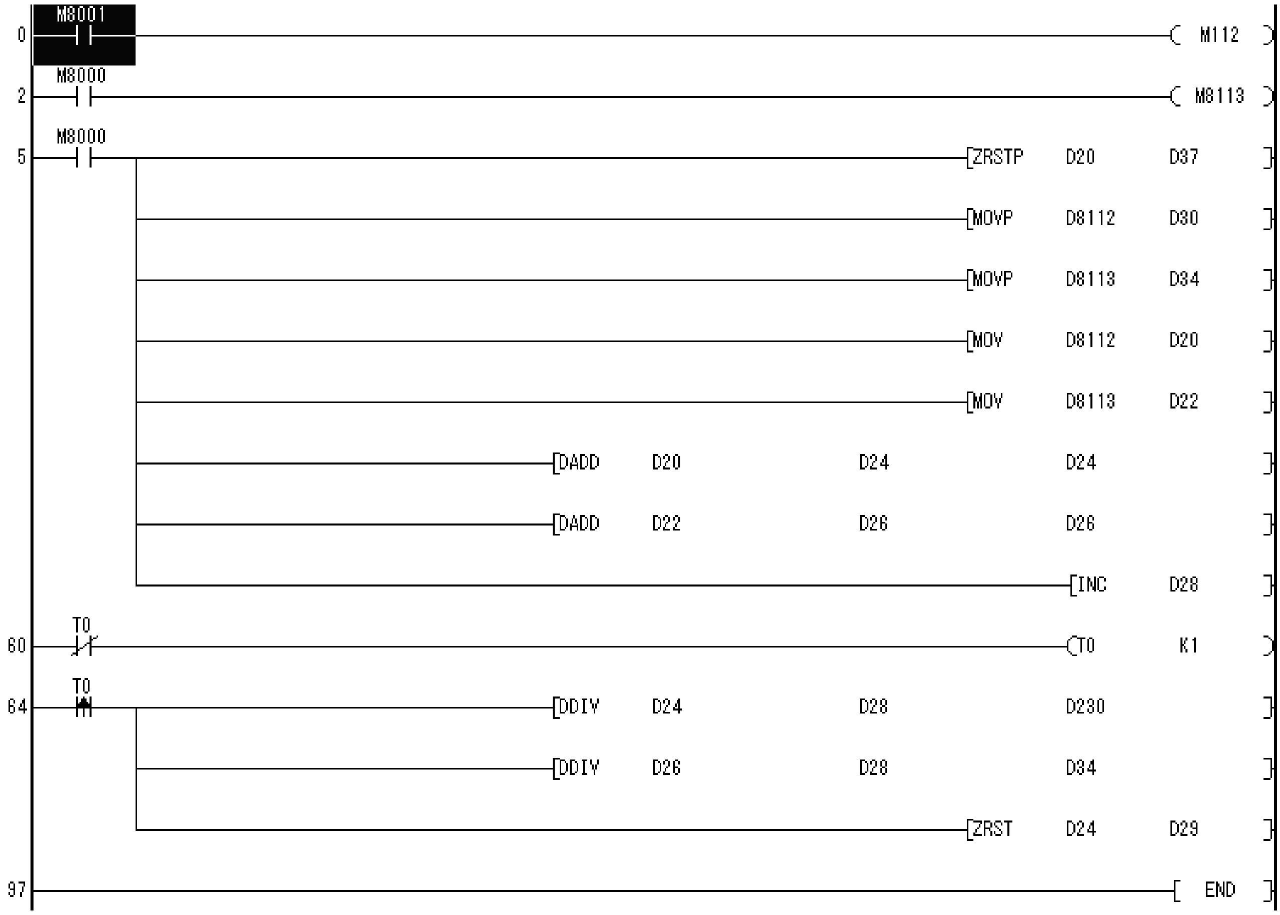
Basic Program #2

Ch1 is set to Current input, Ch2 is set to Voltage input, and the average converted digital value over a set time period is stored in D10 and D14.



Basic Program 3

Ch1 is set to Current input, Ch2 is set to Voltage input, and the average converted digital value over a set time period is stored in D30 and D34, respectively.



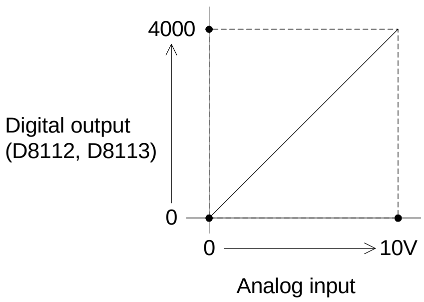
Example Application Programs

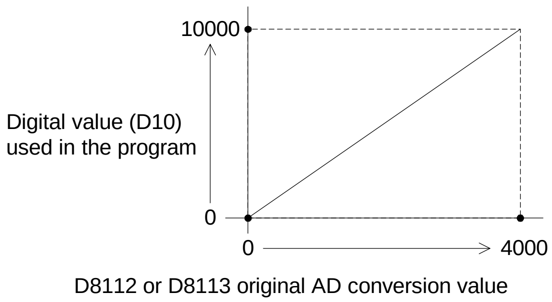
Because the 2AD does not have Offset and Gain capabilities, if values are required outside the standard specification range, additional program commands are required to either multiply or divide the conversion values.

When adjusting the conversion values, some of the resolution will be lost. The original range of the analog input does not change.

Example Application Program #1

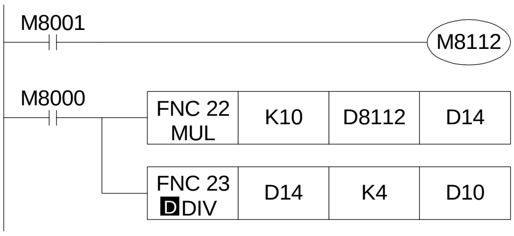
In Voltage input mode, the 2AD converts analog values from 0 ~ 10 Volts to a digital output of 0 ~ 4000. If using a digital range of 0 ~ 10000 in the program, the 0 ~ 4000 output value must be converted as shown in the programming example below. Digital values that are converted from analog values are stored in D8112 or D8113.





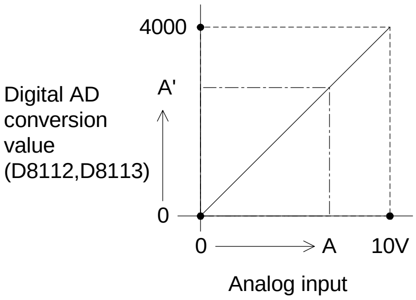
D10 = 10 x D8112 / 4, (D8113 would be used for Ch2)

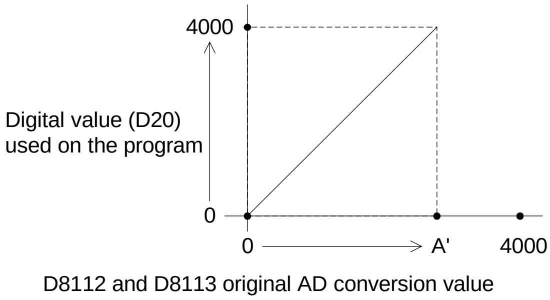
The programming code for the Equation above is given below.



Example Application Program #2

In Voltage input mode, the 2AD converts analog values from 0 ~ 10 Volts to a digital output of 0 ~ 4000. If using an analog range of 0 ~ A (where 0 < A < 10) by a digital output range of 0 ~ 4000, the range must be converted from 0 ~A' to 0 ~ 4000 as shown in the programming code below.





If a digital value of 0 ~ 4000 is used in D20,

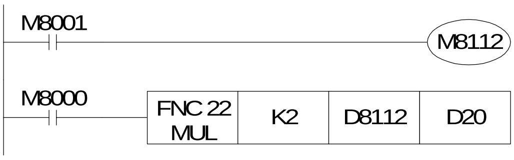
D20 = (4000) x (D8112 or D8113) / A'

4000 / (10 volts) = A' / (A volts), therefore A' = 400 x A

D20 = 4000 x (D8112 or D8113) / 400 x A

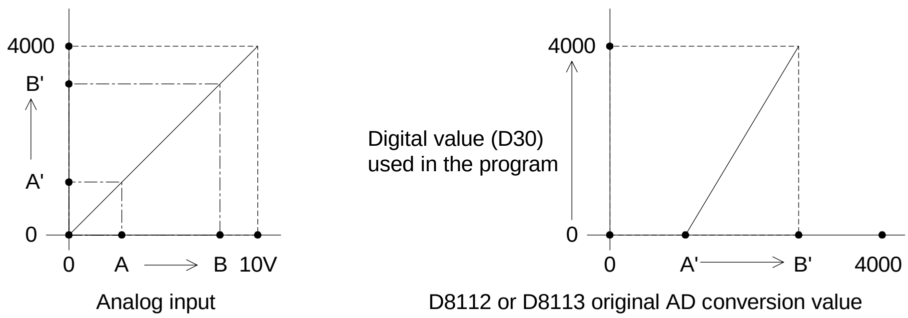
D20 = 10 x (D8112 or D8113) / A and if A = 5

D20 = 2 x (D8112 or D8113)



Example Application Program #3

If using an analog range from A ~ B by a digital range of 0 ~ 4000, the range must be converted from A' ~ B' 0 ~ 4000 in the program as shown in the example below.



If the digital range 0 ~ 4000 is desired in D30, please see the program below.

D30 = 4000 x (D8112 or D8113) / (B' - A') - 4000 x A' / (B' - A')

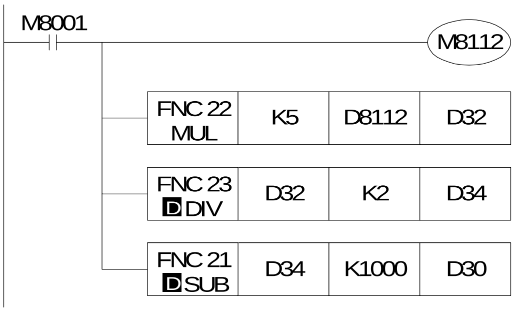
A' = 400 x A, B' = 400 x B so that

D30 = [4000 x (D8112 or D8113) / (400 x B - 400 x A)] - 4000 x (400 x A) / (400 x B - 400 x A)

D30 = [10 x (D8112 or D8113) / (B - A)] - 4000 x A / (B - A)

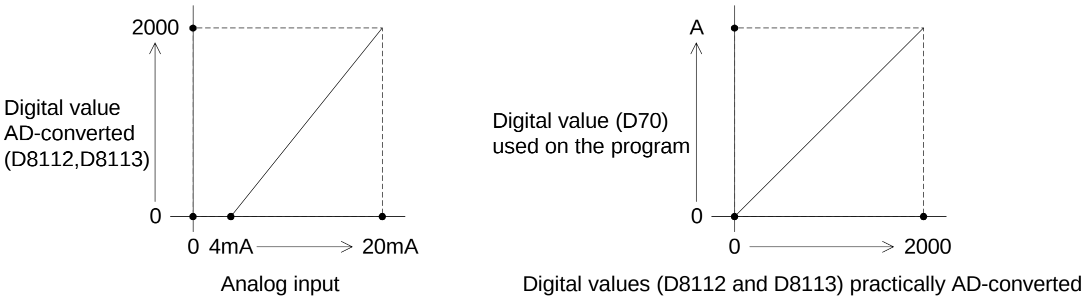
If A = 1 and B = 5

D30 = [5 x (D8112 or D8113) / 2] - 1000



Example Application Program #4

If using an analog range from 4 ~ 20mA to obtain an output range from 0 to A, the normal output range of 0 ~ 2000 be converted to the new range.

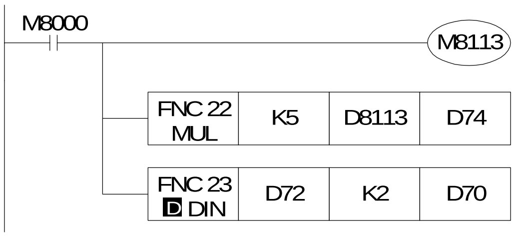


Please perform the conversion as below.

D70 = A x (D8112 or D8113) / 2000. If A = 5000 then,

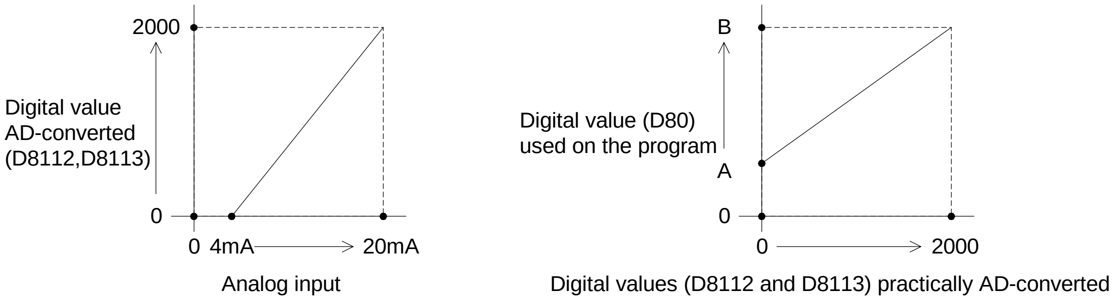
D70 = 5000 x (D8112 or D8113) / 2000

D70 = 5 x (D8112 or D8113) / 2



Example Application Program #5

If using an analog range from 4 ~ 20mA to obtain an output range from A ~ B, the normal output range of 0 ~ 2000 must be converted to the new range.

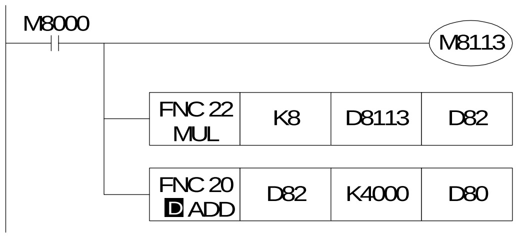


To convert the normal output range of 0 ~ 2000 to the range of A ~ B, please see below.

D80 = (B - A) x (D8112 or D8113) / (2000 - 0) + A; if A = 4000 and B = 20000

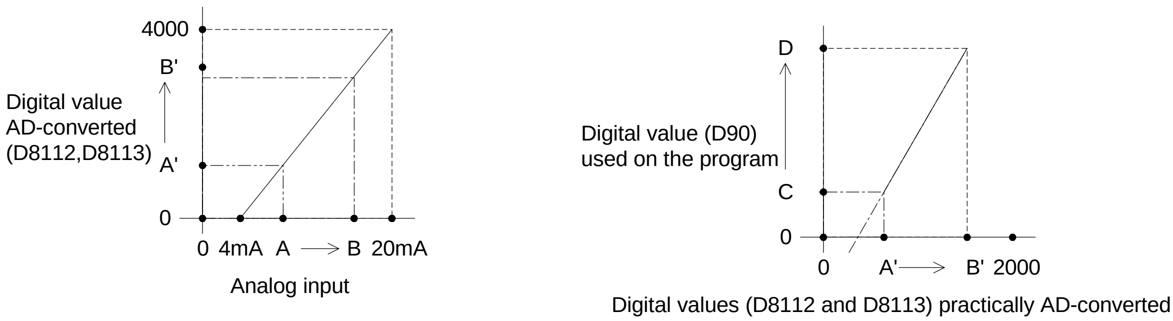
D80 = (20000 - 4000) x (D8112 or D8113) / (2000) + 4000

D80 = 8 x (D8112 or D8113) + 4000



Example Application Program #6

If using an analog range from A ~ B to obtain an output range from C ~ D, both the current and the digital ranges must be converted from the standard ranges.



To convert both ranges, please see the programming example below. More details can be found from the previous examples.

D90 = (D - C) x (D8112 or D8113) / (B’ - A’) + (B’ x C - A’ x D) / (B’ - A’)

D90 = (D - C) x (D8112 or D8113) / [(125 x B - 500) - (125 x A -500)] + [(125 x B - 500) x C - (125 x A - 500) x D] / [(125 x B - 500) - (125 x A - 500)]

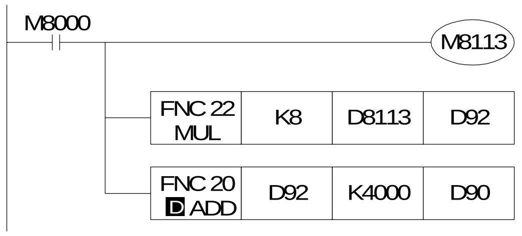
(A’ = 125 x A - 500; B’ = 125 x B - 500)

D90 = (D - C) x (D8112 or D8113) / [125 x (B - A)] + [(B - A) x C - (A - 4) x D] / (B - A)

If A = 5, B = 15, C = 5000, and D = 15000

D90 = (15000 - 5000) x (D8112 or D8113) / [125 x (15 - 5)] + [(15 - 4) x 5000 - (5 - 4) x 15000] / (15 - 5)

D90 = 8 x (D8112 or D8113) + 4000



1	Introduction
2	Basic Program Instructions
3	STL Programming
4	Devices in Detail
5	Applied Instructions
6	Diagnostic Devices
7	Instruction Execution Times
8	PLC Device Tables
9	Assigning System Devices
10	Points of Technique
11	Index

Chapter Contents

10.Points Of Technique....................10-1
10.1 Advanced Programming Points....................10-1
10.2 Users of DC Powered FX Units....................10-1
10.3 Using The Forced RUN/STOP Flags....................10-2
10.3.1 A RUN/STOP push button configuration....................10-2
10.3.2 Remote RUN/STOP control....................10-3
10.4 Constant Scan Mode....................10-4
10.5 Alternating ON/OFF States....................10-4
10.6 Using Battery Backed Devices For Maximum Advantage....................10-5
10.7 Indexing Through Multiple Display Data Values....................10-5
10.8 Reading And Manipulating Thumbwheel Data....................10-6
10.9 Measuring a High Speed Pulse Input....................10-6
10.9.1 A 1 msec timer pulse measurement....................10-6
10.9.2 A 0.1 msec timer pulse measurement....................10-7
10.10 Using The Execution Complete Flag, M8029....................10-7
10.11 Creating a User Defined MTR Instruction....................10-8
10.12 An Example System Application Using STL And IST Program Control....................10-8
10.13 Using The PWM Instruction For Motor Control....................10-15
10.14 Communication Format....................10-18
10.14.1 Specification of the communication parameters....................10-18
10.14.2 Header and Terminator Characters....................10-19
10.14.3 Timing diagrams for communications....................10-20
10.14.4 8 bit or 16 bit communications.....................10-23
10.15 PID programming techniques....................10-24
10.15.1 Keeping MV within a set range....................10-24
10.15.2 Manual / Automatic change over....................10-24
10.15.3 Using the PID alarm signals....................10-25
10.15.4 Other tips for PID programming....................10-25
10.16 Additional PID functions....................10-26
10.16.1 Output Value range control....................10-26
10.17 Pre-tuning operation....................10-27
10.17.1 Variable control....................10-27
10.18 Example Autotuning program....................10-28
10.19 Using the FX_{1N}-5DM Display module....................10-29
10.19.1 Outline of functions....................10-29
10.19.2 Control devices for 5DM....................10-30
10.19.3 Display screen protection function....................10-30
10.19.4 Specified device monitor....................10-31
10.19.5 Specified device edit....................10-32
10.19.6 Automatic Backlight OFF....................10-33
10.19.7 Error display enable / disable....................10-33

10. Points Of Technique

FX1S	FX1N	FX2N	FX2NC

10.1 Advanced Programming Points

The FX family of programmable controllers has a very easy to learn, easy to use instruction set which enables simple programs to perform complex functions. This chapter will point out one or two useful techniques while also providing the user with valuable reference programs.

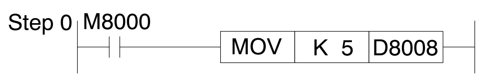
If some of these techniques are applied to user programs the user must ensure that they will perform the task or operation that they require. Mitsubishi Electric can take no responsibility for user programs containing any of the examples within this manual.

Each program will include a brief explanation of the system. Please note that the method of 'how to program' and 'what parameters are available' for each instruction will not be discussed. For this information please see the relevant, previous chapters.

10.2 Users of DC Powered FX2N Units

FX1S	FX1N	FX2N	FX2NC

When using DC powered FX2N programmable controllers, it is necessary to add the following instructions to the beginning of the installed program:



Explanation:

With AC powered FX2N programmable controllers, the power break detection period can be adjusted by writing the desired detection period to the special data register D8008.
However, in the case of DC powered units this detection period must be set to 5 msec.

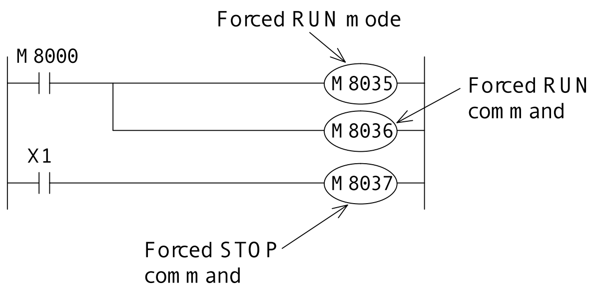
This is achieved by moving the value of -5 into D8008. Failure to do this could result in inputs being missed during the DC power 'drop'.

10.3 Using The Forced RUN/STOP Flags

FX1S	FX1N	FX2N	FX2NC

10.3.1 A RUN/STOP push button configuration

The FX programmable controller has a single RUN terminal. When power is applied to this terminal the PLC changes into a RUN state, i.e. the program contained is executed. Consequently when there is no power 'on' the RUN terminal the PLC is in a STOP state.
This feature can be utilized to provide the FX PLC with an external RUN/STOP - push button control. The following PLC wiring and program addition are required.



Explanation:

Pressing the RUN push button sets the PLC into the RUN state. This means M8000 is ON. Following the program, M8000 activates both M8035 and M8036. These two special auxiliary devices set the PLC in to forced RUN mode. Releasing the RUN push button would normally return the PLC to the STOP state, but because the two auxiliary coils, M8035 and 36 are ON, the PLC remains in RUN. To stop the, PLC pressing the STOP push button drives an input ON and consequently M8037 turns ON. This then automatically forces OFF both M8035 and 36 and resets itself. Hence, the PLC is in its STOP status and awaits the cycle to begin again.

Input priority:

- The STOP input is only processed after the programs END statement has been reached - this is because the physical input used, i.e. an X device is normally updated and processed at that time. Therefor, the RUN input is given priority when both RUN and STOP inputs are given simultaneously.
- To give priority to the STOP input and provide a 'safer' system, some form of mechanical/ circuitry interlock should be constructed between both RUN and STOP inputs. A very simple example is shown in the wiring diagram above.
- For push-button control to operate correctly, the user must set the RUN/STOP switch on FX2N and FX2NC units to the STOP position.
- FX2N and FX2NC units do not have a RUN terminal. One of the inputs X0 to X17 (X0 to X7 for FX2N-16M) on the MPU should be configured as a RUN terminal in the parameter settings.

10.3.2 Remote RUN/STOP control

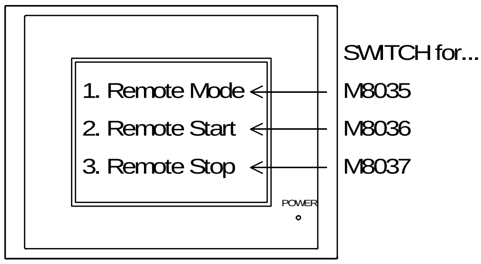
FX1S	FX1N	FX2N	FX2NC

The FX family of programmable controllers can be controlled, i.e. switched into RUN or STOP modes and have devices monitored by use of intelligent external control devices.
These includes such items as computers, the Mitsubishi FX data access units and Graphic Operator Terminals.
The following example utilizes a graphic FX-DU unit:

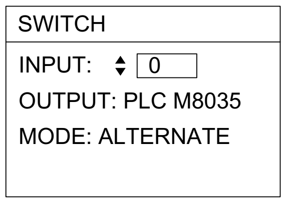
Explanation:

The programmable controller needs no special wiring or additional programming for this example.

The only condition required is that the PLC would not normally be in a RUN state, i.e., there is no connection to the RUN terminal and the RUN/STOP switch on PLC's that have one is set in the STOP position.



The HMI should be programmed with 'SWITCH' devices driving the three special M codes M8035,36 and 37. By activating the 'SWITCH' devices for M8035 and M8036 the PLC can be switched into a RUN state, while driving the 'SWITCH' device M8037 will put the PLC into a STOP state.



Example 'SWITCH' device setting opposite.
Use an 'Alternate' switch for M8035 and M8036 and use a 'Momentary' switch for M8037.
(see DU operation manual for SWITCH operation and programming)

Note: While M8035 and M8036 are ON the MPU can not be changed to STOP mode using the RUN terminal or RUN/STOP switch. Either set M8037 ON, or reset M8035 and M8036, to return to the normal operating state.

Range of Mitsubishi graphic HMI units:

FX-25DU-E - a 4 line text/graphic unit.
FX-30DU-E - a 4 line text/graphics display unit with membrane style keypad.
FX-40DU-TK-E - a 7 line, touch key, text/graphics display unit with numeric keypad.
FX-50DU-TK(S)-E - a 15 line, touch screen, color text/graphics display unit.
F930GOT-BWD - a 5 line, touch screen, monochrome text/graphics advanced display unit.
F940GOT-SWD/LWD-E - a 15 line, touch screen, color text/graphics advanced display unit.

FX1N, FX2N and FX2NC Remote STOP

FX1S	FX1N	FX2N	FX2NC

With FX1N, FX2N and FX2NC units, even if the RUN terminal or RUN/STOP switch is in the RUN position, it is still possible to do a remote STOP by forcing M8037 ON.

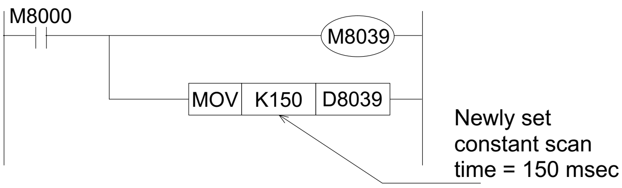
Return to RUN by resetting M8037.

10.4 Constant Scan Mode

FX1S	FX1N	FX2N	FX2NC

Some times the timing of operations can be a problem, especially if some co-ordination is being attempted with a second control system. In cases like this it is very useful to fix the PLC's scan time. Under normal conditions the PLC's scan time will vary from one scan to the next. This is simply because the natural PLC scan time is dependent on the number of and type of the active instructions. As these are continually changing between program scans the actual scan time is also a varying. Hence, by using the additional program function identified below, the PLC's scan time can be fixed so that it will be the same duration on every program scan. The actual scan duration is set by writing a scan time in excess of the current longest scan duration to special data register D8039 (in the example the value K150 is used). If the PLC scans the program quicker than the set scan time, a 'pause' will occur until the set scan duration is reached.

This program example should be placed at the beginning of a users program.

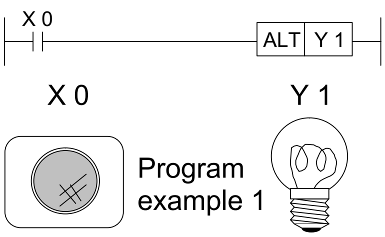


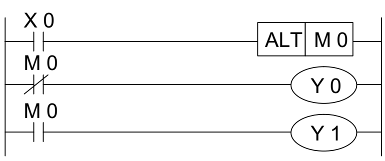
10.5 Alternating ON/OFF States

FX1S	FX1N	FX2N	FX2NC

It is often useful to have a single input control or toggle a situation. A basic, yet typical example is the switching ON/OFF of a Light. This can be easily achieved by using standard ladder program to load an input and switch an output. However, this system requires an input which is latchable. If basic ladder steps are used to latch the program then it soon becomes complex and prone to mis-programming by the user. Using the ALT instruction to toggle the ON/OFF (SET/RESET, START/STOP, SLOW/FAST) state is much simpler, quicker and more efficient.

Explanation:





Program example 2

Pressing the momentary push button X1 once will switch the lamp ON. Pressing the push button for a second time will cause the lamp to turn OFF. And if the push button is again pressed for a third time, the lamp is turned ON again and so the toggled status continues.

The second program shown identifies a possible motor interlock/control, possibly a start/stop situation.

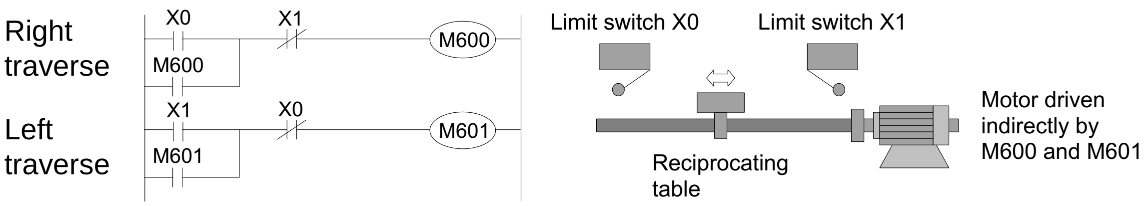
10.6 Using Battery Backed Devices For Maximum Advantage

FX1S	FX1N	FX2N	FX2NC

Battery backed devices retain their status during a PLC power down. These devices can be used for maximum advantage by allowing the PLC to continue from its last operation status just before the power failure.

For example: A table traverse system is operating, moving alternatively between two limit switches. If a PLC power failure occurs during the traversing the machine will stop.
Ideally, once the PLC regains its power the system should continue from where it left off, i.e. if the movement direction was to the left before the power down, it should continue to the left after the restoration of the power.

Explanation:



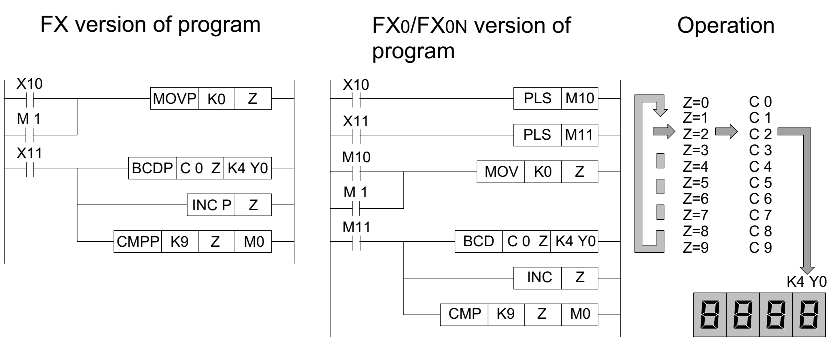
The status of the latched devices (in this example FX M coils M600 and M601) is retained during the power down. Once the power is restored the battery backed M coils latch themselves in again, i.e. the load M600 is used to drive M600.

10.7 Indexing Through Multiple Display Data Values

FX1S	FX1N	FX2N	FX2NC

Many users unwarily fall in to the trap of only using a single seven segment display to display only a single data value. This very simple combination of applied instructions shows how a user can 'page' through multiple data values displaying each in turn.

Explanation:

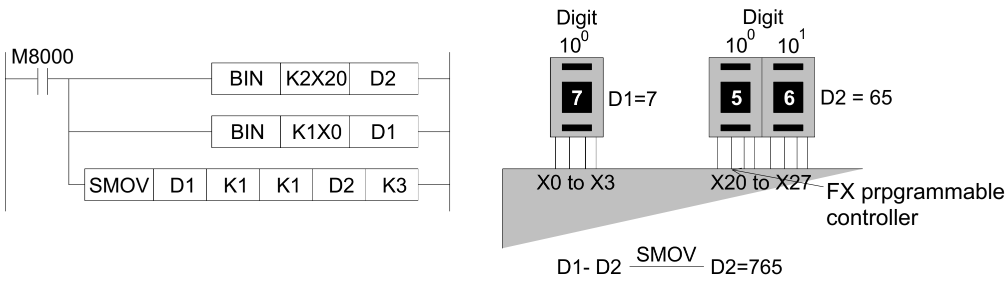


The contents of 10 counters are displayed in a sequential, 'paged' operation.
The paging action occurs every time the input X11 is received.
What actually happens is that the index register Z is continually incremented until it equals 9. When this happens the comparison instruction drives M1 ON which in turn resets the current value of Z to 0 (zero). Hence, a loop effect is created with Z varying between fixed values of 0 and 9 (10 values). The Z value is used to select the next counter to be displayed on the seven segment display.
This is because the Z index modifier is used to offset the counter being read by the BCD output instruction.

10.8 Reading And Manipulating Thumbwheel Data

FX1S	FX1N	FX2N	FX2NC

Data can be easily read into a programmable controller through the use of the BIN instruction. When data is read from multiple sources the data is often stored at different locations. It may be required that certain data values are combined or mixed to produce a new value. Alternatively, a certain data digit may need to be parsed from a larger data word. This kind of data handling and manipulation can be carried out by using the SMOV instruction. The example below shows how two data values (a single digit and a double digit number) are combined to make a final data value.



Explanation:

The two BIN instructions each read in one of the data values. The first value, the single digit stored in D1, is combined with the second data value D2 (currently containing 2 digits). This is performed by the SMOV instruction. The result is that the contents of D1 is written to the third digit of the contents of D2. The result is then stored back into register D2.

10.9 Measuring a High Speed Pulse Input

FX1S	FX1N	FX2N	FX2NC

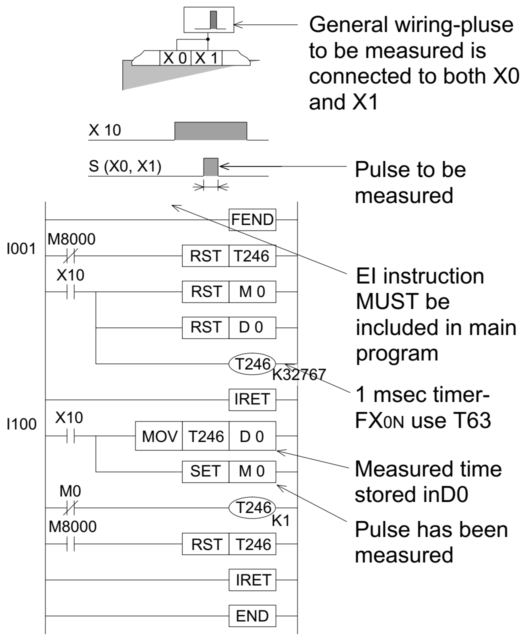
10.9.1 A 1 msec timer pulse measurement

Some times due to system requirements or even as a result of maintenance activities it is necessary to 'find out' how long certain input pulses are lasting for. The following program utilizes two interrupt routines to capture a pulse width and measure it with a 1 msec timer. The timer used in the example is one of the FX timers. However, T63 on the FX1N would be used for a similar situation on that PLC.

Explanation:

The 1 msec timer T246 is driven when interrupt I001 is activated. When the input to X1 is removed the current value of the timer T246 is moved to data register D0 by interrupt program I100. The operation complete flag M0 is then set ON.

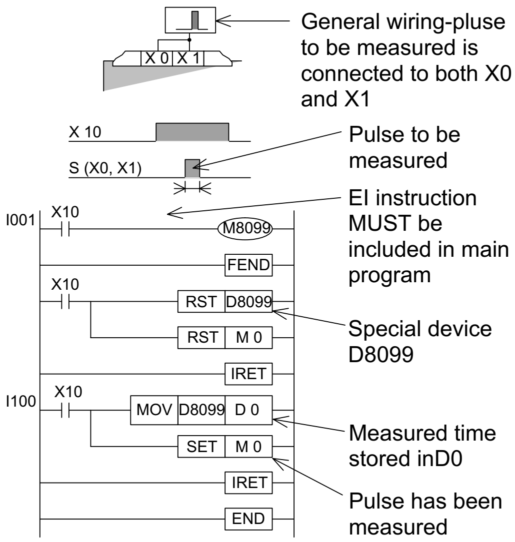
Note: X10 acts as an enable/disable flag.



10.9.2 A 0.1 msec timer pulse measurement

FX1S	FX1N	FX2N	FX2NC

This is a very accurate measuring process for pulse inputs. The use of a standard timer is not accurate enough in this case as the highest resolution is 1msec. Therefor, this example shows how the special high accuracy devices M8099 and D8099 are used to capture the 0.1 msec resolution pulse data.



Explanation:

The incoming pulse is captured between two interrupt routines. These routines operate independently of each other, one on the rising edge of the pulse input and one on the falling edge of the same input. During the pulse input the contents of special register D8099 are continually moved into data register D0. Once the pulse has completed the contents of D0 can be viewed at leisure.

Please note for this high speed/accuracy mode to be active for D8099, the corresponding special auxiliary bit device M8099 must be driven ON in the main program.

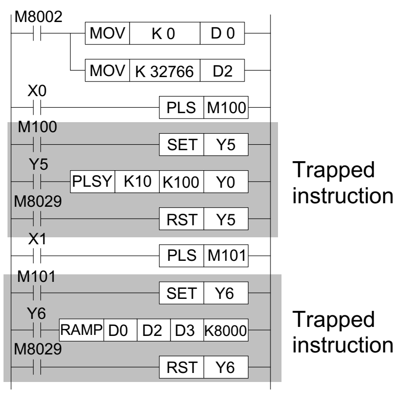
10.10 Using The Execution Complete Flag, M8029

FX1S	FX1N	FX2N	FX2NC

Some of the applied instructions take more than one program scan to complete their operation. This makes identification of the current operating state difficult. As an aid to the programmer, certainappliedinstructionsidentify theircompletionbysettinganoperationcompleteflag, M8029. Because this flag can be used by several different instructions at the same time, a method similar to the following should be used to trap the M8029 status at each of the instructions using it:

Explanation:

The M8029 'trapping' sequence takes advantage of the batch refresh of the FX family of PLC's. As the program scan passes each instruction using M8029 the status of M8029 changes to reflect the current status of the instruction. Hence, by immediately resetting (or setting) the drive flag for the instruction the current operational status of the instruction is trapped. So when the batch refresh takes place only the completed instructions are reset. The example above uses a pulse to set the drive flags so that it is easy to monitor and see when each instruction finishes (if the instructions are continuously driven it will be difficult to see when they finish!).



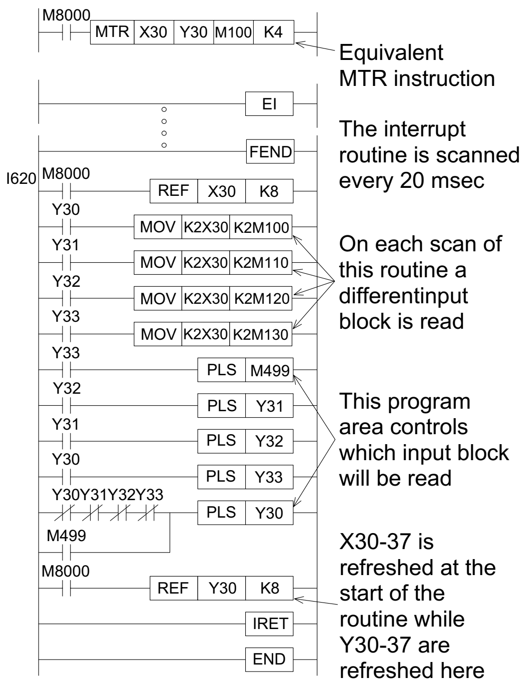
10.11 Creating a User Defined MTR Instruction

FX1S	FX1N	FX2N	FX2NC

For users who want to have the benefits of the MTR instruction for FX users who want to specify more than one MTR area, this user defined MTR function will be very useful.

Explanation:

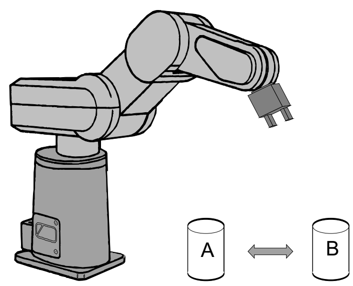
The main control of this program rests in the timer interrupt I620. This interrupt triggers every 20msec regardless of what the main program is doing. On each interruption one bank of the user defined matrix is read. The program simply consists of reading the inputs triggered by each of the multiplexed outputs.
The read data is then stored in sequential sets of auxiliary registers.
Each MOV instruction reads a new bank of multiplexed inputs.
The equivalent MTR instruction is shown immediately before the 'user defined' MTR.
See the MTR instruction on page 5-54 for more details.



10.12 An Example System Application Using STL And IST Program Control

FX1S	FX1N	FX2N	FX2NC

The following illustration shows a simple 'pick and place' system utilizing a small robotic arm. The zero point has been de-fined as the uppermost and left most position accessible by the robot arm.



A normal sequence of events

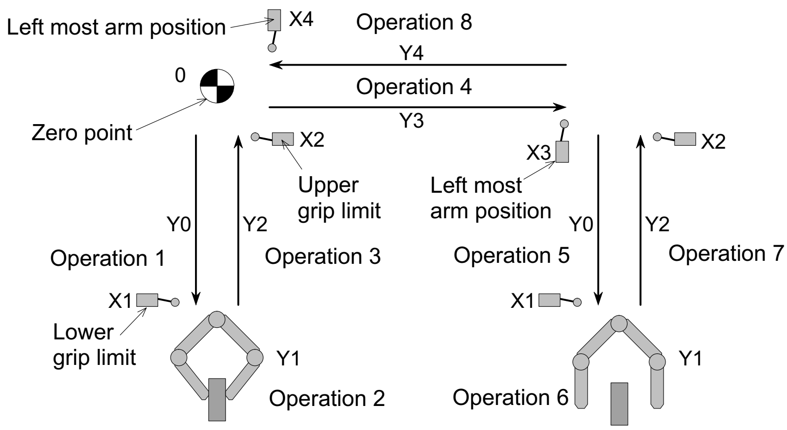
A product is carried from point 'A' to point 'B' by the robot arm. To achieve this operation the following sequence of events takes place:
Initial position: the robot arm is at its zero point.

1) The Robots grip is lowered to it lowest limit
   - output Y0: ON, input X1: ON, output Y0: OFF.
2) The grip clamped around the product at point A
   - output Y1: ON.

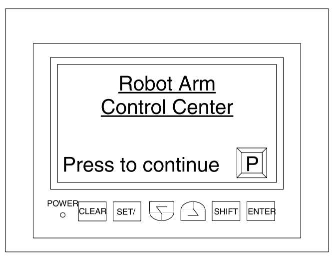
3) The grip, now holding the product, is raised to its upper limit
   - output Y2: ON, input X2: ON, output Y2: OFF.
4) The robot arm traverses to its right most position
   - output Y3: ON, input X3: ON, output Y3: OFF.
5) The grip and product are lowered to the bottom limit
   - output Y0: ON, input X1: ON, output Y0: OFF.
6) The grip is unclamped and the product is released at point B
   - output Y1: OFF.
7) The grip is retrieved back to its upper limit
   - output Y0: ON, input X2: ON, output Y0: OFF.
8) The arm traverses back to its zero point by moving to the left most limit
   - output Y4: ON, input X4: ON, output Y4: OFF.

The cycle can then start again.

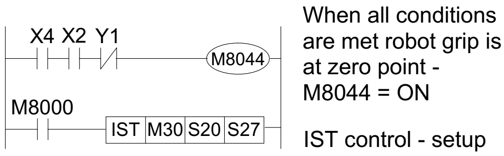
System parameters



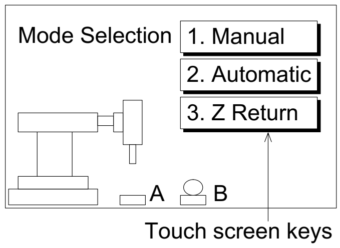
1) Double solenoid valves are used to control the up (Y2)/down (Y0) and right (Y3)/left (Y4) motion.
2) A single solenoid valve is used for the clamp (Y1)/unclamp operation.
3) The system uses an FX-40DU-TK to interface with the operator.
   The FX-40DU-TK is a touch screen data access unit.



This example uses the IST instruction (FNC 60) to control the operation mode of the robot arm. The program shown opposite identifies how the IST instruction is written into the main program.



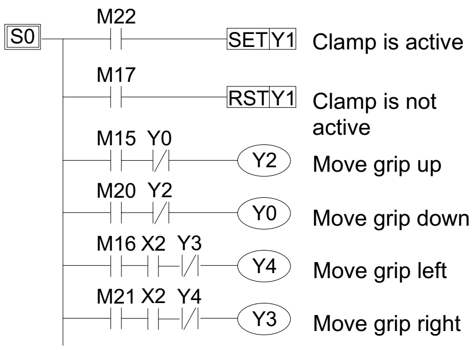
When the IST instruction is used there are 5 selectable modes which access three separate programs. This example has the following programs associated with its modes. Each mode is selected through the FX-40DU-TK. The screen shown opposite is the initial mode menu. Each of the menu options causes a screen jump to the selected mode. Menu options 1 and 3 also set ON auxiliary devices M30 and M31 respectively.
The active bits then trigger a screen change to the selected mode. Please note 'Automatic' has three further modes which are selected from a following screen/display.



An example DU screen design

Manual Mode:

In this mode ALL operations of the robot arm are controlled by the operator. An operation or movement is selected by pressing the corresponding option on the DUs screen (see below). These options then trigger DU SWITCH objects which drive associated auxiliary relays within the programmable controller. The SWITCH objects should be set to momentary so that they only operate when the key is pressed.



The status of the clamping action could be identified by two INDICATOR (SCR) functions on the DU unit. They could be monitoring the ON and OFF status of the clamp output Y1. Hence, when the clamp was ON a single black box opposite the ON button could appear. When the clamp is OFF the box would appear in front of the OFF button. At any one time only one box would be active.

Key assignment for DU screen opposite:

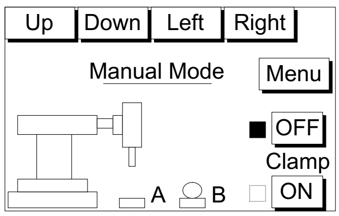
Up = M15 Down = M20

Left = M16 Right = M21

Clamp ON = M22

Clamp OFF = M17

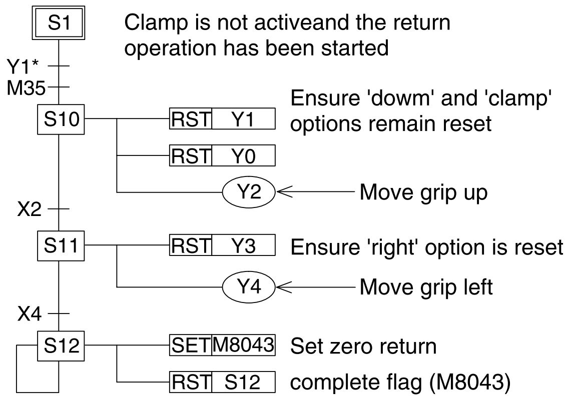
Menu = reset M30



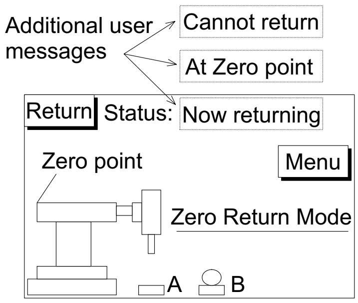
Once manual operation is completed the operator can return to the main mode selection screen by touching the 'Menu' key. This causes the manual mode bit flag, M30, to be reset. Once M30 is reset the DU screen then changes back to the desired mode selection screen.

Zero Return Mode

This mode fulfills an initialization function by returning the robot arm to a known position. Once 'Z Return' has been selected from the mode selection screen the bit device M35 is ON. At this point the DU screen changes to the 'zero return' screen.
The actual zero return operation will then start when the 'Return' push button is pressed (activating M25) and the robots grip is not active, i.e. Y1 is OFF (on the STL flow diagram opposite Y1 OFF is shown as Y1*).



The DU unit could be used to report back the status of the current returning operation. The example screen shown opposite uses 3 variable messages to indicate this status. The messages could be text strings stored in the PLC which are read and displayed by the DUs ASCII option.



Once the zero point has been returned to, the operator would also return to the mode selection screen. This is achieved by pressing the 'Menu' touch key. This then resets the zero return bit device M31 which allows the DU screen change to take place.

Key assignment for DU screen above:
Return = M25
Menu = reset M31

Automatic Mode

Under this option there are three further mode selections. The available modes are:

Step Mode:

- The automatic program is stepped through - operation by operation, on command by the user pressing the 'Start' button.

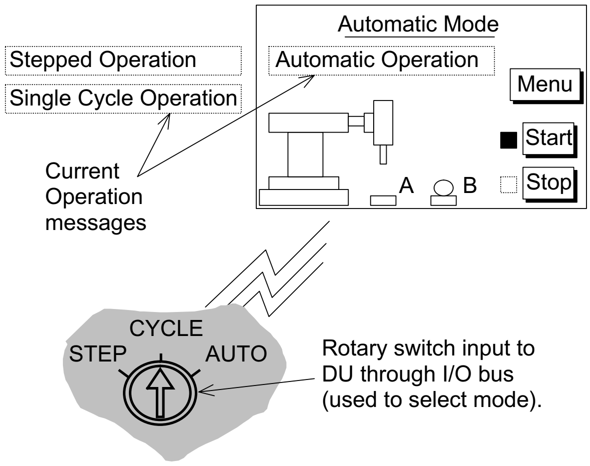
Cycle Mode:

- The automatic program is processed for one complete operational cycle. Each cycle is initiated by pressing the 'Start' button. If the 'Stop' button is pressed, the program is stopped immediately. To resume the cycle, the 'Start' button is pressed again.

Automatic Mode:

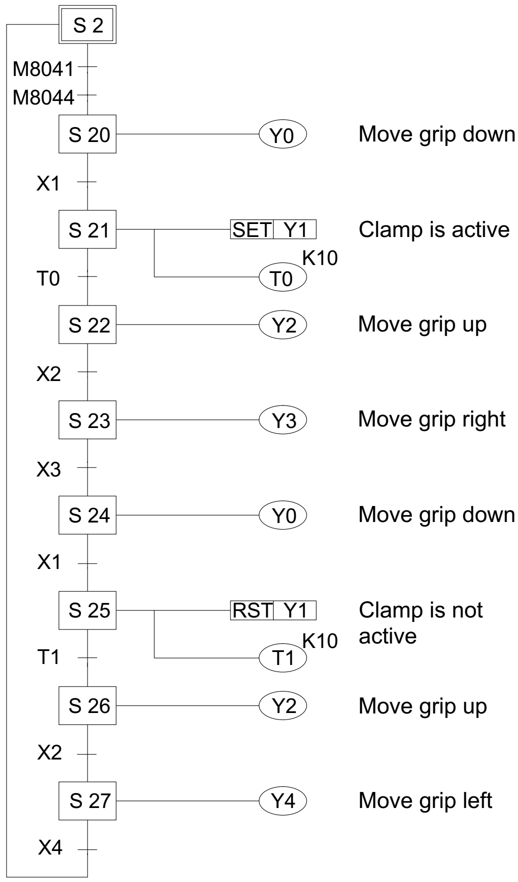
- A fully automatic, continuously cycling mode. The modes operation can be stopped by pressing the 'stop' button. However, this will only take effect after completion of the current cycle.

In this example these three modes are selected by an external rotary switch. The rotary switch is not connected to the PLC but to the I/O bus on the rear of the DU unit.
The use of the rotary switch means that the selected modes are mutually exclusive in their operation. For an operator friendly environment the currently selected mode is displayed on the DU screen (again this could be by use of the DUs ASCII function).
The start/ stop controls are touch keys on the DU screen. When a mode is selected the input received at the DU unit momentarily activates one of the following auxiliary relays:
Rotary switch:
position 1 'Step' - Step operation: DU input I0, controls bit device M32 position 2 'Cycle' - Single cycle operation:
DU input I1, controls bit device M33 position 3 'Auto' - Automatic operation: DU input I2, controls bit device M34



Key assignment for DU screen above:
Start = M36
Stop = M37

The program run in all three mode choices is shown opposite. As noted earlier, the 'Step' mode will require an operator to press the 'Start' key to start each new STL block. This could be viewed as an additional transfer condition between each state. However, the user is not required to program this as the IST instruction controls this operation automatically.



The 'Cycle' mode will process the program from STL step S2, all the way through until STL step S2 is encountered again. Once more the IST instruction ensures that only one cycle is completed for each initial activation of the 'Start' input.

Finally as suggested by the name, 'Auto' mode will continuously cycle through the program until the 'Stop' button is pressed. The actual halting of the program cycling will occur when the currently active cycle is completed.

Points of interest:

a) Users of the IST instruction will be aware that only one of the operation modes should be active at one time. In this example program the isolation of 'Manual' and 'Zero return' modes by the use of separate DU control screens, and the use of a rotary switch to isolate the three automatic modes achieves this objective. Alternatively all of the operation modes could be selected by a rotary switch.

b) For users who would like to test this example using simulator switches (i.e., without using a data access unit) the appropriate program changes are noted next to the full program listing later in this section. Alternatively, the original program could be used with all of the input conditions being given by forcing ON the contacts with a programming device e.g. a hand held programmer, Medoc etc.

c) Special flags used in this program are:

- M8040: State transfer inhibit
  - Manual mode: Always ON.
    Zero return and Cycle modes: Once the 'Stop' input is given the current state is retained until the 'Start' input is received.
    Step mode: This flag is OFF when the 'Start' input is ON. At all other times M8040 is ON, this enables the single STL step operation to be achieved.
    Auto mode: M8040 is ON initially when the PLC is switched into RUN. It is reset when the 'Start' input is given.
- M8041: State transfer start
  - Manual and Zero return modes: This flag is not used.
    Step and Cycle modes: This flag is only active while the 'Start' input is received.
    Auto mode: The flag is set ON after the 'Start' input is received. It is reset after the 'Stop' input is received.
- M8042: Start pulse
  - This is momentarily active after the 'Start' input is received.
- M8043: Zero return complete
  - This is a user activated device which should be controlled within the users program.
- M8044: At Zero position/ condition
  - This is a user activated device which should be controlled within the users program.

Full program listing:

0	LD	X	4
1	AND	X	2
2	ANI	Y	1
3	OUT	M	8044
5	LD	M	8000
6	IST		60
		M	30
		S	20
		S	27
13	STL	S	0
14	LD	M	8044
15	OUT	M	8043
17	LD	M	22
18	SET	Y	1
19	LD	M	17
20	RST	Y	1
21	LD	M	15
22	ANI	Y	0
23	OUT	Y	2
24	LD	M	20
25	ANI	Y	2
26	OUT	Y	0
27	LD	M	16
28	AND	X	2
29	ANI	Y	3
30	OUT	Y	4
31	LD	M	21
32	AND	X	2
33	ANI	Y	4
34	OUT	Y	3
	(RET)*		

↑

*: Instructions in ( ) are not necessary

necessary

35	STL	S	1
36	LD	M	35
37	RST	M	8043
39	ANI	Y	1
40	SET	S	10
42	STL	S	10
43	RST	Y	1
44	RST	Y	0
45	OUT	Y	2
46	LD	X	2
47	SET	S	11
49	STL	S	11
50	RST	Y	3
51	OUT	Y	4
52	LD	X	4
53	SET	S	12
55	STL	S	12
56	SET	M	8043
58	RST	S	12
	(RET)*		
60	STL	S	2
61	LD	M	8041
62	RST	M	8043
64	AND	M	8044
65	SET	S	20
67	STL	S	20
68	OUT	Y	0
69	LD	X	1
70	SET	S	21

This instruction returns the program flow to STL step S2. →

72	STL	S	21
73	SET	Y	1
74	OUT	T	0
		K	10
77	LD	T	0
78	SET	S	22
80	STL	S	22
81	OUT	Y	2
82	LD	X	2
83	SET	S	23
85	STL	S	23
86	OUT	Y	3
87	LD	X	3
88	SET	S	24
90	STL	S	24
91	OUT	Y	0
92	LD	X	1
93	SET	S	25
95	STL	S	25
96	RST	Y	1
97	OUT	T	1
		K	10
100	LD	T	1
101	SET	S	26
103	STL	S	26
104	OUT	Y	2
105	LD	X	2
106	SET	S	27
108	STL	S	27
109	OUT	Y	4
110	LD	X	4
111	**OUT**	**S**	**2**
113	RET		
114	END		

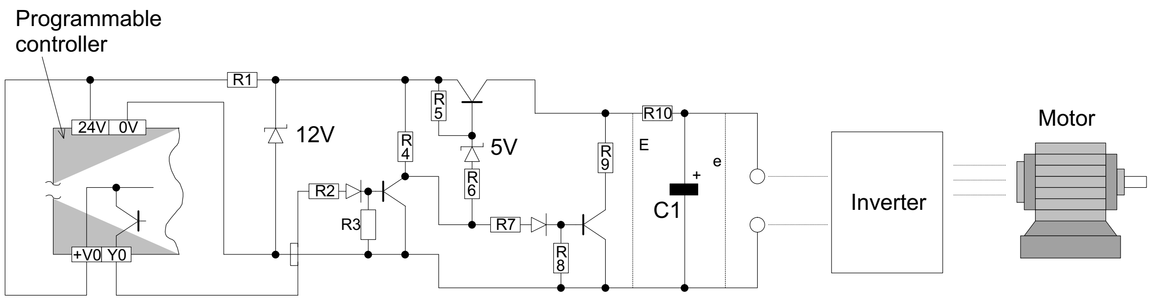
Program options:

6	IST		60	17	LD	**X**	**12**	27	LD	**X**	**6**
		X	**20**	19	LD	**X**	**7**	31	LD	**X**	**11**
		S	20	21	LD	**X**	**5**	36	LD	**X**	**25**
		S	27	24	LD	**X**	**10**				

10.13 Using The PWM Instruction For Motor Control

FX1S	FX1N	FX2N	FX2NC

The PWM instruction may be used directly with an inverter to drive a motor. If this configuration is used the following ripple circuit will be required between the PLC's PWM output and the inverters input terminals.



Circuit configuration for a PLC with source outputs

Key to component values:

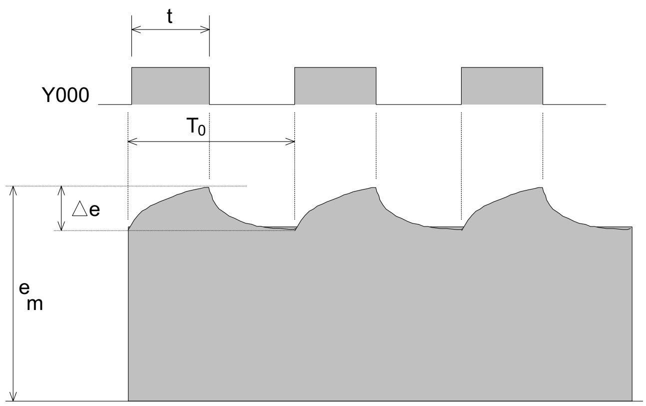
R1 - 510 Ω (1/2 W) R2 - 3.3kΩ (1/2 W)

R3 to R8 - 1kΩ (1/4 W) R9 - 22 Ω (1/4 W)

R10 - variable dependent on configuration. In this example 1kΩ (1 W)

C1 - 470 μF

Note: the values of R10 and C1 are dependent on the system configuration.



Establishing system parameters and values

It is assumed that the input impedance of the inverter is of a high order. Having established this, the values of C1 and R10 are calculated to give τ a time result (in msec) approximately 10 times bigger than the value used for T_0 in the PWM instruction:

τ = R10 (kΩ) ÅL C1 (μF)

During this calculation the value of R10 must be vastly greater than the value of R9. In the example, R9 is equal to 22Ω, where as R10 is equal to 1kΩ. This proportion is approximately 1:50 in favor of R10.

The maximum output voltage (to the inverter) including ripple voltage, can be found by using the following equation:

$$e_m \approx E \frac{t}{T_0}$$

Where:

e_m = Maximum output voltage

E= pulse (square wave) output voltage (see circuit on the previous page)

t = PWM pulse duration (see previous page for reference)

T_0 = PWM cycle time for pulse (see previous page for reference)

The average output voltage (to the inverter) including ripple voltage, can be found by using the following equation:

$$\frac{\Delta e}{e} \approx \frac{T_0 - t}{\tau} \leq \frac{T_0}{\tau}$$

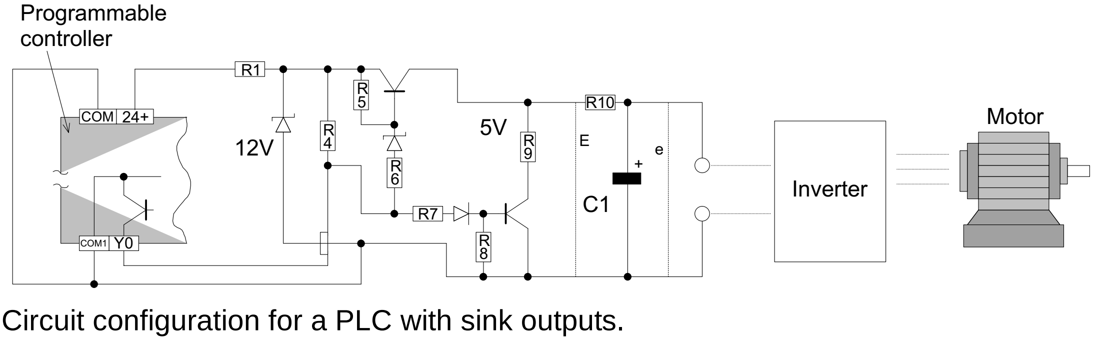
Where:

Δe = the voltage value of the ripple
e = ripple output voltage
T_0 = PWM cycle time for pulse
t = PWM pulse duration
τ = ripple circuit delay

See previous page for references.

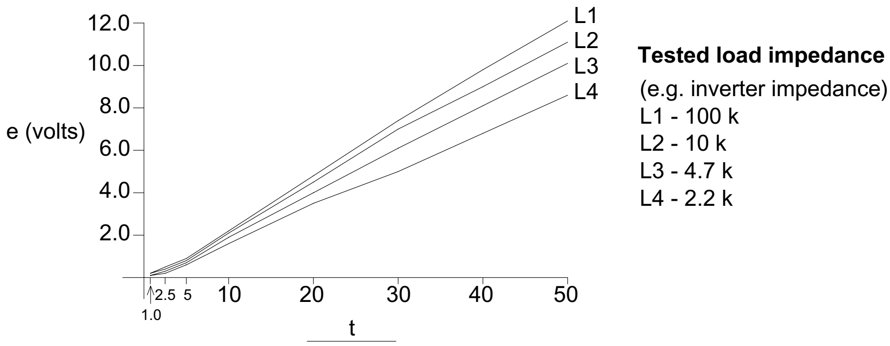
Operation

Once the system configuration has been selected and the ripple circuit has been built to suit, the motor speed may be varied by adjusting the value of 't' in the PWM instruction.
The larger the value of 't' the faster the motor speed will rotate. However, this should be balanced with the knowledge that the faster the output signal changes the greater the ripple voltage will be. On the other hand a slowly changing output signal will have a more controlled, yet smaller ripple effect. The speed of the signal change is determined by the size of C1. A large capacitive value for C1 would give a smaller ripple effect as charge is stored and released over a longer time period.



Circuit configuration for a PLC with sink outputs.
The component values are the same as stated previously

The following characteristics were noticed when the identified circuit was tested
The PWM instruction had T_0 set to K50. The value for t was varied and also the load impedance was varied to provide the following characteristics graph (see over page).



The duration of the T_0, time base also affects the ripple voltage. This can be clearly seen in the next set of test data:

PWM parameter setting			Measured ripple voltage
t	T_0	t / T_0	
100	200	0.5	1.27V
50	100		668mV
25	50		350mV
10	20		154mV
5	10		82mV

The behavior of the Sink switched circuit detailed above will be similar to that of the Source switched circuit detailed earlier.

10.14 Communication Format

FX1S	FX1N	FX2N	FX2NC

10.14.1 Specification of the communication parameters:

Items such as baud rates, stop bits and parities must be identically set between the two communicating devices. The communication parameters are selected by a bit pattern which is stored in data register D8120.

D8120				
	Description		Bit (bn)status	
			0 (OFF)	1 (ON)
b0	Data length		7 bits	8 bits
b1 b2	Parity (b2, b1)		(00): No parity (01): Odd parity (11): Even parity	
b3	Stop bits		1 bit	2bits
b4 b5 b6 b7	Baud rate - bps		(b7, b6, b5, b4) (0011): 300 bps (0100): 600 bps (0101): 1200 bps (0110): 2400 bps	(b7, b6, b5, b4) (0111): 4800 bps (1000): 9600 bps (1001): 19200 bps
b8	Header character		None	D8124, Default: STX (02H)
b9	Terminator character		None	D8125, Default: ETX (03H)
b10 b11 b12	Communication Control (see timing diagrams page 10-20 onwards)		No Protocol (b12, b11, b10) (0, 0, 0): RS Instruction is not being used (RS232C interface) (0, 0, 1): Terminal mode -RS232C interface (0, 1, 0): Interlink mode - RS232C interface (FX2N V2.00 or above) (0, 1, 1): Normal mode 1- RS232C, RS485(422) interfaces (RS485 FX2N(C) only) (1, 0, 1): Normal Mode 2 - RS232C interface (FX only)	
			Computer Link (b12, b11, b10) (0, 0, 0): RS485(422) interface (0, 1, 0): RS232C interface	
b13	FX-485 Network	Sum Check	No Check	Added automatically
b14		Protocol	No protocol	Dedicated Protocol
b15		Protocol	Format 1	Format 4

General note regarding the use of Data register D8120:

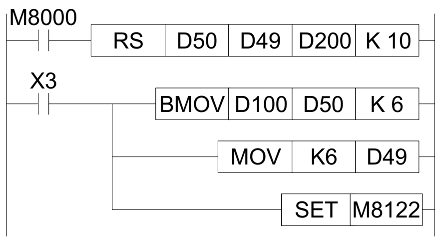
This data register is a general set-up register for all ADP type communications. Bits 13 to 15 in the 232ADP units should not be used. When using the FX-485 network with 485ADP units bits 13 to 15 should be used instead of bits 8 to 12.

10.14.2 Header and Terminator Characters

The header and terminator characters can be changed by the user to suit their requirements. The default setting for the header stored in D8124 is 'STX' (or 02H)and the terminator default setting stored in D8125 is 'ETX' (or 03H).
The header and terminator characters are automatically added to the 'send' message at the time of transmission. During a receive cycle, data will be ignored until the header is received. Data will be continually read until either the termination character is received or the receive buffer is filled. If the buffer is filled before the termination character is received then the message is considered incomplete.
If no termination character is used, then reading will continue until the receive data buffer is full. Only at this point will a message have been accepted and complete. There is no further buffering of any communications, hence if more data is sent than the available destination buffer size then the excess will be lost once the buffer is full.
It is therefore very important to specify the receive buffer length the same size as the longest message to be received.

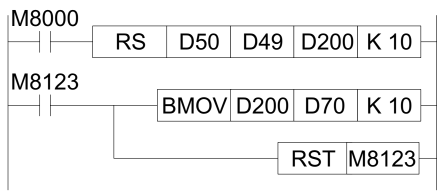
Events to complete a transmission:

The RS instruction should be set up and active.
The data to be transmitted should be moved into the transmission data buffer. If a variable is being used to identify the message length in the RS instruction this should be set to the new message length. The send flag M8122 should then be SET ON. This will automatically reset once the message has been sent. Please see the example program right.



Events encountered when receiving a message:

The RS instruction should be set up and active.
Once data is being received and an attempt is made to send out data, the special M flag M8121 is set ON to indicate the transmission will be delayed. Once the 'incoming' message is completely received the message received flag M8123 is set ON. At the same time if M8121 was ON it is automatically reset allowing further messages (delayed or otherwise) to be transmitted.



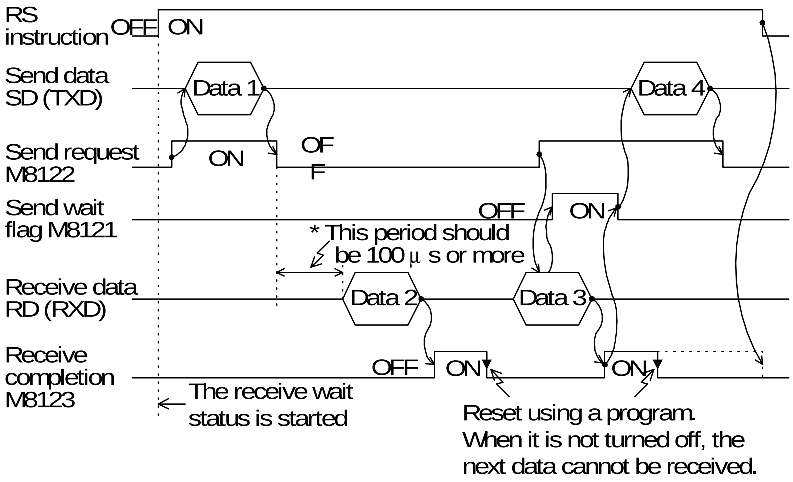
It is advisable to move the received data out of the received data buffer as soon as possible. Once this is complete M8123 should be reset by the user. This is then ready to send a message or to await receipt of a new message.

10.14.3 Timing diagrams for communications:

FX1S	FX1N	FX2N ★	FX2NC

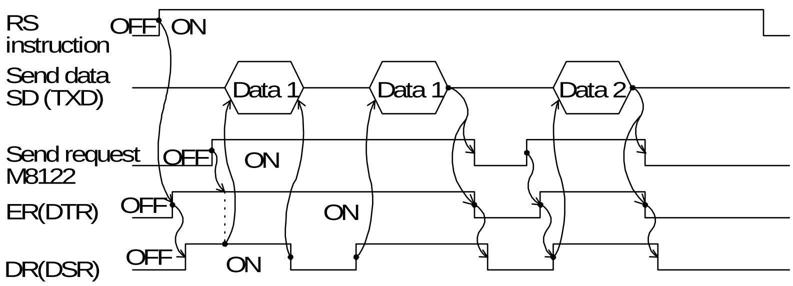
1) No Handshaking D8120 (b12, b11, b10) = (0, 0, 0)

☆FX2N below version 2.00

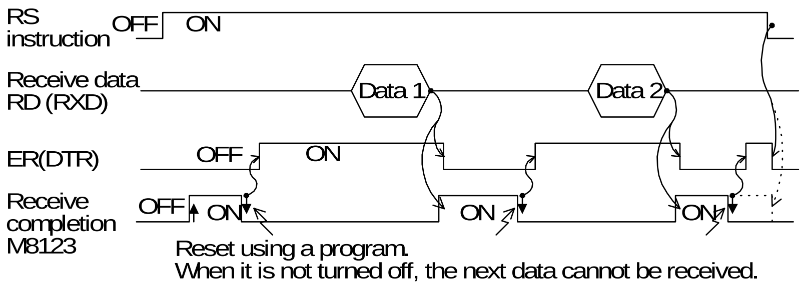


2) Terminal mode D8120 (b12, b11, b10) = (0, 0, 1)

a) Send Only



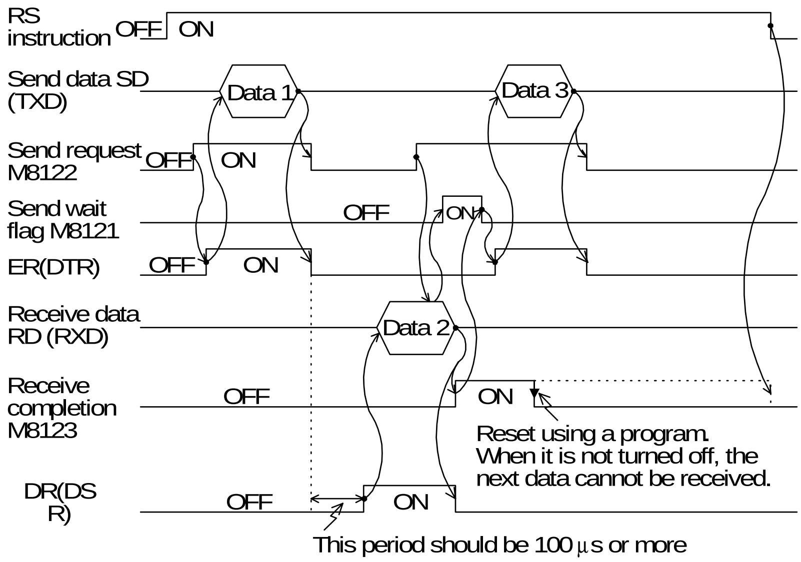
b) receive only



3) Normal Mode 1 D8120 (b12, b11, b10) = (0, 1, 1)

FX1S	FX1N	FX2N★	FX2NC

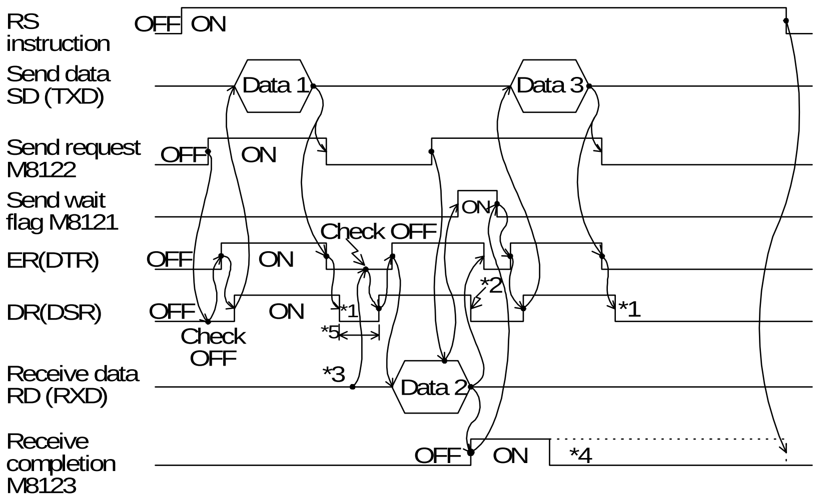
☆FX2N below V2.00.



4) Normal Mode 2 D8120 (b12, b11, b10) = (1, 0, 1)

FX1S	FX1N	FX2N★	FX2N(C

☆FX2N after V2.00

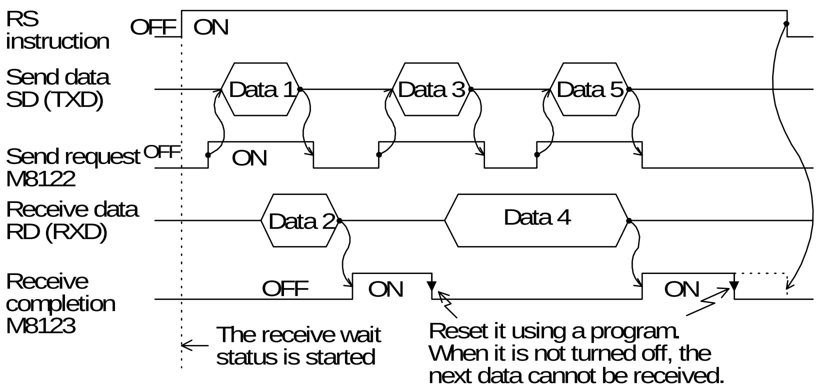


FX2N (V2.00 or above) Communications

FX1S	FX1N	FX2N	FX2NC

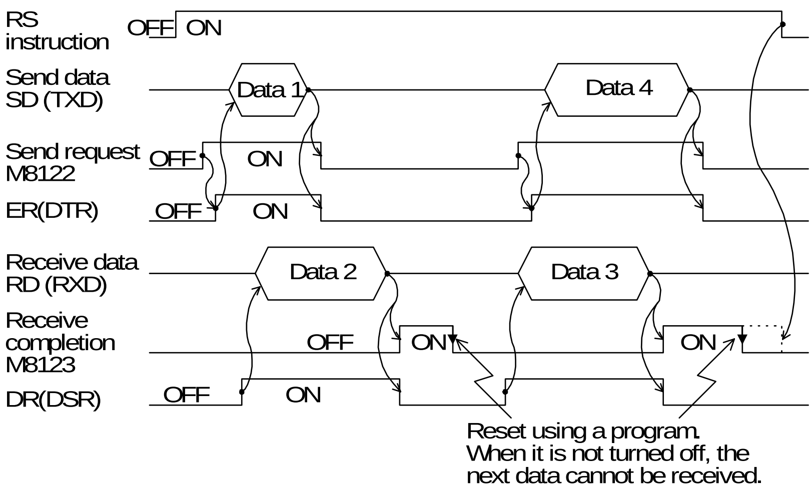
In the FX2N V2.00 or above and FX2NC, full duplex communication is performed.

1) No Hardware Handshaking D8120 (B12, b11, b10) = (0,0,0)



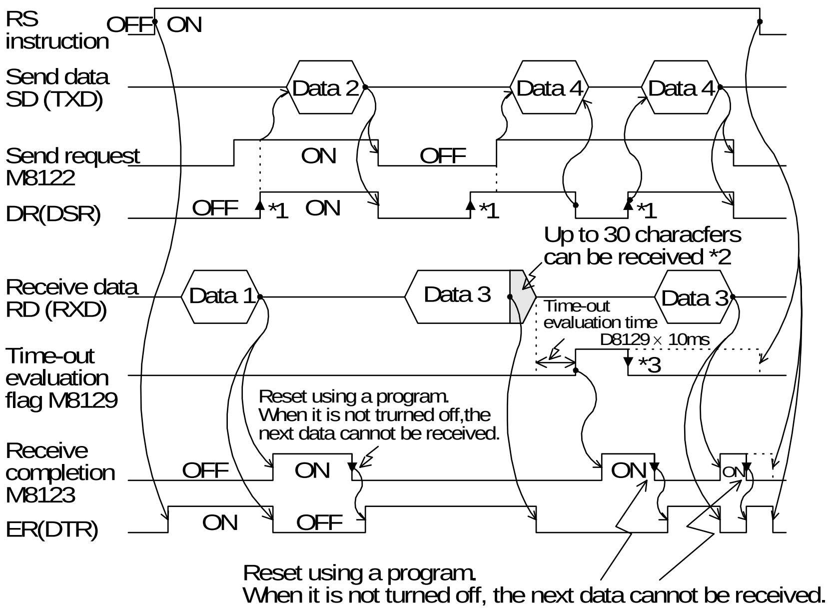
2) Terminal Mode
The control line and transmission sequence are identical to those in the FX, on page

3) Normal Mode 1 D8120 (b12, b11, b10) = (0, 1, 1)



4) Interlink Mode D8120 (b12, b11, b10) = (0, 1, 0)

FX1S	FX1N	FX2N	FX2NC



10.14.4 8 bit or 16 bit communications.

This is toggled using the Auxiliary relay M8161. When this relay is OFF 16 bit communications takes place. This actually means that both bytes of a 16 bit data device are used in both the transmission and the receipt of messages. If the M8161 device is activated then 8 bit mode is selected. In this mode only the lower 8 bits (or byte) is used to perform the transmission-receiving actions. The toggling of the M8161 device should only occur when the RS instruction is not active, i.e. it is OFF.

When a buffer area is specified in the RS instruction it is important to check whether 8 or 16bit mode has been selected, i.e. a buffer area specified as D50 K3 would produce the following results.......

16 bit mode - M8161 = OFF		
Data register	High byte	Low byte
D50	X	F
D51		0

8 bit mode - M8161 = ON		
Data register	High byte	Low byte
D50		F
D51		X
D52		0

General note regarding hardware:

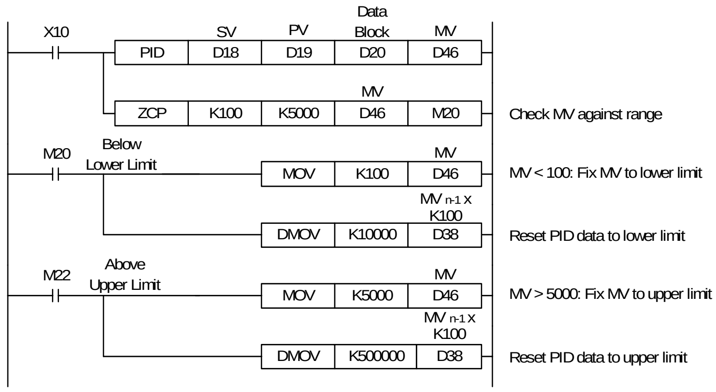
Information regarding pin outs of the respective ADP special function blocks can be found along with wiring details in the appropriate hardware manuals.

10.15 PID Programming Techniques

FX1S	FX1N	FX2N	FX2NC

10.15.1 Keeping MV within a set range

In the reserved registers of the PID data block S_3+18 and S_3+19 form a double word device that contains the previous MV x K100. The following program uses this to keep MV under control when it exceeds the operating limits.
Example Program to keep MV in the range K100 to K5000



If data registers are used to hold the limit values, it is possible to use a MUL instruction instead of the DMOV. E.g. When D50 is upper limit use: MUL D50 K100 D38 because the result of MUL is already a double word DMUL is not needed.
Resetting (S_3+19, S_3+18) in this way prevents runaway, which occurs if only MV is changed.

10.15.2 Manual/Automatic change over

In order to switch from automatic (PID) control to manual control and back to automatic it is necessary for the PID process to perform 'Manual Tracking'. Although the FX PID instruction does not have a manual tracking feature there are two methods that can be used to make the switch from manual back to automatic as trouble free as possible.
To understand the reason for the two methods the following should be noted. The PID instruction sets its initial output value based on the initial value of the output register.
When the PID instruction is switched on it can only do P as it has only 1 data reading. On the first reading the current value of the output register is used as ΔMV. Thereafter the previous output value is used (stored in S_3+18, S_3+19).

After the next reading PI can be calculated and from the third reading full PID is performed.

Please see section 5.98, PID (FNC 88), for the complete equations.

Method

It is recommended that if manual to auto switching is desired that the PID instruction is switched off during manual operation and the operator controls the value of the MV register (the Output Value). When returning to auto mode, the PID instruction is switched on again and uses the last MV input by the operator during the first PID calculation. After 3 readings full PID will be operating and the process should be under control quickly. (Assuming that manual control did not cause a move too far from the Set Point.)

10.15.3 Using the PID alarm signals

Included as part of the data block there are four alarm values. These set the maximum positive and negative change that should occur to MV and PV. The PID alarm signals are used to warn of the system going out of control.
When the system is starting from cold it is usually not good to include the Derivative numbers of the in the calculation; the changes to PV are large and the Derivative introduces too much correction. Also, if the system starts to move rapidly away from the SV then sometimes the use of D can over correct and cause chasing.
By having an 'alarm' flag for the change in PV and MV it is possible to monitor the state of the system and adjust the PID parameters to appropriate settings.
When the system is close to the SP the changes in PV (and MV) should be minimal.
In this situation using full PID is very useful in keeping the system close to the SP. (Full PID is appropriate).
However, if the conditions change (e.g. opening a refrigerator door, adding ingredients to a mixture, cold start, etc.) the system reacts. In some cases (especially cold start) the reaction is too much for the D to be useful (PI or sometimes just P only is better). In these cases the alarm flags can be used to change to PI control until the system returns to a more stable condition, when full PID can then be used.
Basically, rather than use actual values of the PV to determine the change over point from PI to PID (or PID to PI), use the size of the change in PV (or MV). This means changes to the Set Point do not require different ranges for the PI - PID change over point (at least, in theory).

10.15.4 Other tips for PID programming

- It is recommended that an input value for PV is read before the PID is activated. Otherwise, the PID will see a big change from 0 to the first value and calculate as if a big error is occurring.
- The PID instruction is not interrupt processed. It is scan dependent and as such the sampling can not occur faster the FX scan time. It is recommended that T_S is set to a multiple of the program scan time.
- To keep timing errors to a minimum it is recommended that constant scan is used.
- To improve sampling rates it is possible to put the PID instruction inside a timer interrupt routine.
- It is better to have the PID only perform P until the input value (PV) reaches the working range.
- When setting up it is a good idea to monitor the input and output of the PID instruction and check that they are about the expected values.
- If the PID system is not operating properly check the error flags for PID errors (D8067).

10.16 Additional PID functions

FX1S	FX1N	FX2N	FX2NC

The following parameter table gives the additional parameters available with FX2N(C) MPUs. These are:

- S_3+1 bit 4: Pre-tuning operation flag.
- S_3+1 bit 5: Output Value range limit flag.
- S_3+22: Output Value upper limit.
- S_3+23: Output Value lower limit.

Parameter S_3 + P	Parameter name/function	Description		Setting range
S_3+1	Action-reaction direction and alarm control	b0	Forward operation(0), Reverse operation (1)	Not applicable
		b1	Process Value (S_2) change alarm OFF(0)/ON(1)	
		b2	Output Value (*MV*) change alarm OFF(0)/ON(1)	
		b3	Reserved	
		b4	Activate pre-tuning (auto resets on completion)	
		b5	Output Value *(MV)* range limit OFF(0)/ON(1)	
		b6-15	Reserved	
S_3+22	Output Value, maximum positive change alarm	Active when S_3+1, b2 is set ON.	This is an alarm for the quantity of positive change which can occur in one PID scan. If the Output Value *(MV)* exceeds this value, bit S_3+24, b2 is set	0 to 32767
	Output Value, Upper limit restriction	Active when S_3+1, b5 is set ON.	This is an upper limit for the Output Value *(MV)*. During operation the PID instruction restricts the output so that it does not exceed this limit.	-32768 to 32767
S_3+23	Output Value, maximum negative change alarm	Active when S_3+1, b2 is set ON.	This is an alarm for the quantity of negative change which can occur in one PID scan. If the Output Value *(MV)* falls below this value, bit S_3+24, b3 is set.	0 to 32767
	Output Value, Lower limit restriction	Active when S_3+1, b5 is set ON.	This is a lower limit for the Output Value *(MV)*. During operation, the PID instruction restricts the output so that it does not fall below this limit.	-32768 to 32767

For the full list of other parameters refer to page 5-102.

Note: S_3+1 b2 and b5 should not be active at the same time. Only one value each is entered into the data registers S3+22 and S3+23.

10.16.1 Output Value range control (S_3+1 b5)

Bit 5 of parameter S_3+1, when ON, activates S_3+22 and S_3+23 to be upper and lower limits for the output value (MV).
This feature restricts the output value to the specified limits; in effect, this automatically performs the same operation as that described in section 10.15.1.

10.17 Pre-tuning operation

FX1S	FX1N	FX2N	FX2NC

10.17.1 Variable Constants

The Pre-tuning operation can be used to automatically set values for the following variables:

- The direction of the process; Forward or Reverse (S_3+1, bit 0)
- The proportional gain constant; K_P (S_3+3)
- The integral time constant; T_I (S_3+4)
- The derivative time constant; T_D (S_3+6)

Setting bit 4 of S_3+1 starts the pre-tuning process. Before starting, set all values that are not set by the pre-tuning operation: the sample time, Ts (S_3+0); the input filter α (S_3+2); the Derivative gain, K_D (S_3+5); the Set Point, SV (S_1); and any alarm or limit values, (S_3+20-23).

The Pre-tuning operation measures how fast the system will correct itself when in error. Because the P, I, and D equations all react with differing speed, the initial error must be large so that effective calculations can be made for each type of equation. The difference in values between SP and PV_{nf} must be a minimum of 150 for the Pre-tuning to operate effectively. If this is not the case, then please change SV to a suitable value for the purpose of pre-tuning.

The system keeps the output value (MV) at the initial value, monitoring the process value until it reaches one third of the way to the Set Point. At this point the pre-tuning flag (bit 4) is reset and normal PID operation resumes. SV can be returned to the normal setting without turning the PID command Off.

During the course of normal operation, the Pre-tuning will NOT automatically set new values if the SV is changed. The PID command must be turned Off, and the Pre-Tuning function restarted if it is necessary to use the Pre-tune function to calculate new values.

- Caution: The Pre-tuning can be used as many times as necessary. Because the flag resets, the set bit can be turned On again and new values will be calculated. If the system is running an oven heater and the SV is reduced from 250 to 200 C, the temperature must drop below 200 or the “Forward/Reverse” flag will be set in the wrong direction. In addition, the system error value must be large for the pre-tune variable calculations to work correctly.

- Note: Set the sampling time to greater than 1 second (1000 ms) during the pre-tuning operation. It is recommended that the sampling time is generally set to a value much greater than the program scan time.

- Note: The system should be in a stable condition before starting the pre-tuning operation. An unstable system can cause the Pre-tuning operation to produce invalid results. (e.g. opening a refrigerator door, adding ingredients to a mixture, cold start, etc.)

- Note: Even though Pre-tuning can set the above mentioned variables, additional logic may be needed in the program to "scale" all operating values to those capable of being processed by the special function devices being used.

10.18 Example Autotuning Program

The following programming code is an example of how to set up the Pre-Tuning function.

D500: SV = 500

D502: MV = 1800, initial value

D510: T_S, S_3+0 = 3000

D511: S_3+1, Bits 0-3 and 5-15 Off, Bits 4 and 5 On. Bit 4 = Pre-Tune Function
Bit 5 = MV Range Limit

D512: Input Filter, S_3+2 = 70%

D515: K_D , S_3+5 = 1800, initial value

D532: MV Max, S_3+22 = 2000

D533: MV Min, S_3+23 = 0

Pulse M1 to turn On PID command

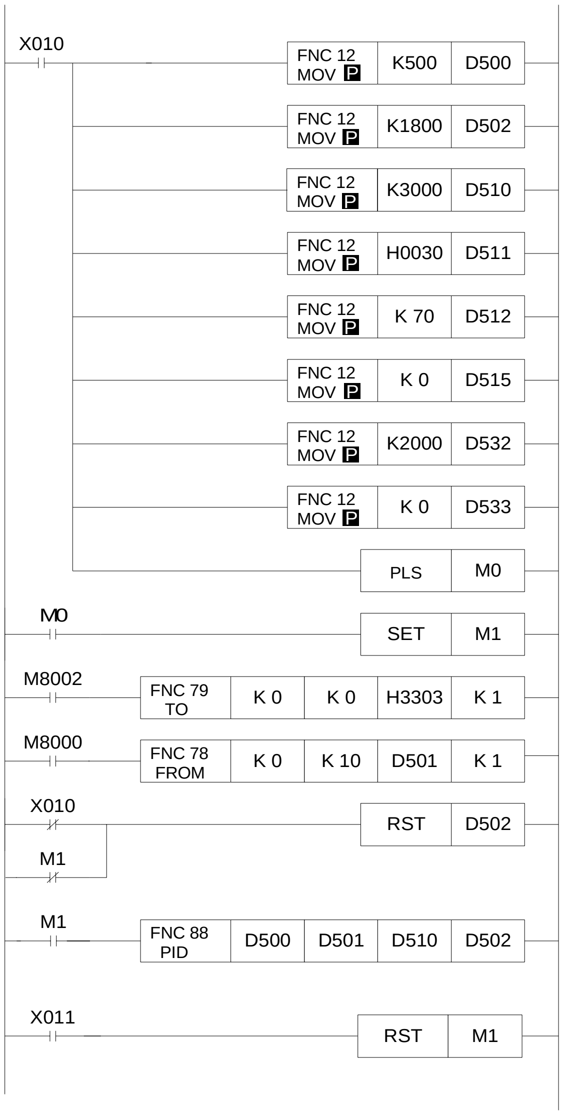
Send setting to Special Function Block

Read data from Special Function Block

Reset Output data when PID command is Off

PID Instruction Command Line

Turn Off PID Instruction



10.19 Using the FX1N-5DM Display module.

FX1S	FX1N	FX2N	FX2NC

The display module, FX1N-5DM (hereafter refered to as 5DM) can be mounted on an FX1S or FX1N PLC, allowing devices to be monitored, and data settings changed.

10.19.1 Outline of functions.

Symbols in the 5DM refer to;
X: Input, Y: Output, M: Auxillary relay, S: State, T: Timer, C: Counter, D: Data register.

Operator functions: The following functions can be used only from the operation keys on the front of the 5DM. (Refer to the 5DM Hardware manual for the correct procedure when using the operation keys).

Function		Description
Clock		
	Display	Displays built-in RTC of FX1N / FX1S
	Setting	Allows the setting of - Year, month, day, hour and minute.
Device monitor		
	Bit device	Displays the ON / OFF status of X, Y, M & S
	Word device (16-bit)	Displays the current values of T, C & D. Allows setting of T & C
	Word device (32-bit)	Displays the current values of 32bit C & D. Allows setting of 32bit C
Buffer memory monitor		Displays the buffer memory of special units and blocks (FX1N only)
Error display		Displays Error codes and error occurance step number
Forced Set / Reset		Forces ON or OFF bit devices Y, M & S
T/C reset		Clears the current values of T & C
Data change		
	Current value	Allows the changing of current values in T, C and D
	Set value	Allows the changing of set values in T, & C

5DM Control functions: The following functions can be used only when controlled by the sequence programs.

Function	Description	Section
Protect	Enables either, all operator functions, only monitor function, or only clock time display.	
Specified device monitor	Allows user to specify device type and number to be displayed	
Error display enable / disable	Enables or Disables the error display function	
Auto backlight OFF	Sets the automatic backlight off time	
Operation key status recognition	Recognised the ON/OFF status of the four operation keys	

If a key word to prohibit read or write is registered in the PLC, only the clock time display is available. All other functions shown above are not.
If an operation is performed in this state, the display flickers for 5 seconds.

10.19.2 Control devices for 5DM

When using the 5DM control functions, write the head device number of Data registers (D) and Auxiliary relays (M), to the special data registers D8158 and D8159 respectively.
D8158 and 8159 are the control devices for the 5DM.

Five data registers and 15 auxiliary relays are available for the control of a 5DM.

Special D	Control device	Description	Application
D8158 K☆	D☆	Device type to be displayed	Specified device monitor function
	D☆+1	Device number to be displayed	
	D☆+2	Backlight OFF time (minutes)	Auto backlight OFF function
	D☆+3	Display screen protection	Protect function
	D☆+4	Not available	
D8159 K☐	M☐	Request Edit of displayed device data	Specified device monitor function
	M☐+1	Edit complete response signal	
	M☐+2	Disable backlight OFF function	Auto backlight OFF function
	M☐+3	Enable / disable error display	
	M☐+4	[ESC] key status	Specified device monitor function
	M☐+5	[-] key status	
	M☐+6	[+] key status	
	M☐+7	[OK] key status	
	M☐+8	Not available	
	M☐+9		
	M☐+10		
	M☐+11		
	M☐+12		
	M☐+13		
	M☐+14		

If a nagative value or a value outside the D or M device ranges in the FX1S/FX1N is written to D8158 or D8159, the 5DM control functions are disabled. (The initial value of D8158 and D8159 is '-1' so that the functions are disabled).

10.19.3 Display screen protect function

By writing a specific numeric value to 'D☆+3' (5DM control device), operator functions with regard to display and setting can be restricted.

Control device	Current value of D☆+3	
D8158 K☆	0	All operator functions are valid, no protection
	1	Only time display is valid, current time cannot be changed
	2	Only device monitor display is valid, settings cannot be changed
	Other value	All operator functions are valid, no protection

10.19.4 Specified device monitor

It is possible to specify in the PLC, the devices to be displayed on the 5DM.
When specifying a device to be displayed, write the correcponding number shown in the table below to D☆.

Current value of D☆	Device type
1	Input (X)
2	Output (Y)
3	Auxiliary relay (M)
4	State (S)
5	Timer (T)
6	Counter (C), 16-bit current and set value or 32-bit set value
7	Data register (D) 16-bit
8	Data register (D) 32-bit
9	Time display
Any other value	Not used *1

*1 If a numeric value other than 1~9 is writen, no device will be specified. In this case all operator functions are valid.

Points to note:

a)During the monitoring of devices T or C, if a device number not used in the program is specified, the next largest existing device number is displayed. If the specified device number is beyond the range available, the largest existing device number will be displayed.
If the OUT instruction for the T or C is not present in the sequence program, '----' is displayed on the 5DM screen.

b)When scrolling and displaying consecutive devices using the operation keys, move up and down the range with the [+] and [-] keys.

c)If the device numbers are not consecutive, and scrolling is required, some additional PLC code will be needed. The range of device numbers to be displayed will have to be related to an index register, the [+] and [-] keys increment and decrement the current value of this register, and therefore change the displayed values.

d)If data registers used in D8158 are located in the non-backup area, the current values of the data registers are reset to '0' when the PLC is stopped. As a result of this, the device type to be displayed, set by D☆ becomes invalid and, the operator functions become valid.
In order to disable the operator functions, use data registers located in a battery backed area.

10.19.5 Specified device edit

This function allows the operator to edit the devices displayed by the specified device monitor. The following devices are used to achieve this.

Special D	Control device	Description
D8159 K☐	M☐	Request to edit displayed device data
	M☐+1	Edit complete response signal

Points to note;

a)In order to edit a device while it is being displayed, the control device M☐ should be ON.
If the edit request turns OFF, the function is disabled. In order to prevent this, it is recommended to drive M☐ using a set command.

b)When the edit request is turned ON, bit devices Y, M and S can be set or reset. Also the current and set values of word devices D, T and C can be edited.

- Bit devices - A cursor under the device flickers, pressing [OK] sets or resets the device. The [ESC] key signifies the end of the change process, M☐+1 set OFF and M☐ is reset.

- Word devices - The current value flickers, pressing the [+] or [-] keys will increment or decrement the current value.
  Pressing the [OK] key before the [+] or [-] keys in the case of T or C, allows the set values to be changed.
  Pressing the [OK] key after a value change, completes the change. Pressing the [ESC] key cancels the change and completes the process, for either key M☐+1 is set to OFF and M☐ is reset.

c)If the [+] and [-] keys are used for device scrolling, when the current or set value is increased or decreased for editing purposes, the program for timer scroll will be actuated. For this combination of functions please set an interlock in the sequencer program.

10.19.6 Automatic Backlight OFF

Using this function a set time until the backlight is switched OFF can be set, or it can be forced ON and OFF when necessary.

Special D	Control device	Description
D8158 K☆	D☆+2	Backlight OFF time
D8159 K□	M□+2	Disable automatic backlight OFF (Forced ON)

D☆+2 can be set in the following range;
0 (initial value) : 10 minutes
1 to 240 : 1 to 240 minutes
240 or more : 240 minutes
Negative value : Forced OFF

Points to note;

a)Once the backlight turns OFF, it will turn ON again when any key is pressed. This key will act as a trigger, not an effective key. The contents displayed before the backlight OFF, will them be shown.

b)Setting a Negative value in D☆+2 will force the backlight OFF, setting M□+2 the backlight can be forced ON.

10.19.7 Error display enable / disable

Users can specify the types of errors in the PLC to be displayed on the 5DM unit.

Special D	Control device	Description
D8159 K□	M□+3	Enable / Disable operation errors etc.

The following errors are unconditionally displayed when they occur;
PLC Hardware, parameter, grammatical and circuit errors.

While M□+3 is ON, the following errors are also displayed;
I/O configuration, parallel link and operation errors.

When any key is pressed, or when the error status is released the error display dissappears.

If two or more errors have occured, the priority is given to errors to be unconditionally displayed. Additionally the error with the smallest ‘error number’ has overall priority.

Memo

1	Introduction
2	Basic Program Instructions
3	STL Programming
4	Devices in Detail
5	Applied Instructions
6	Diagnostic Devices
7	Instruction Execution Times
8	PLC Device Tables
9	Assigning System Devices
10	Points of Technique
11	Index

Chapter contents

11.Index..11-1
11.1 Index...11-1
11.2 ASCII Character Codes...11-9
11.3 Applied Instruction List 11-10

11. Index

11.1 Index

A

Absolute current value read, ABS instruction 5-164
Absolute drum sequence, ABSD instruction 5-88
Addition of data values, ADD instruction 5-30
Addressing special function blocks 9-1
Advanced programming points
    Examples and tips 10-1
Alternated state, ALT instruction 5-92
Alternating states using ALT, example 10-4
ANB 2-12
And block instruction 2-12
And, And inverse instructions 2-6
AND, ANI 2-6
Annunciator reset, ANR instruction 5-62
Annunciator set, ANS instruction 5-61
And Pulse, And trailing Pulse instructions 2-9
ANP, ANF 2-9
Applied instr' which can only be used once 7-15
Applied instruction list 11-10
Applied instructions 5-1
Arrow switch, ARWS instruction 5-109
ASCII character codes 11-9
ASCII code (Alpha to ASCII code), ASCI instr' 5-110
ASCII to HEX conversion using HEX (FNC 83) 5-122
Assigning special function block numbers 9-1
Assigning system devices 9-1
Associated Manuals 1-4
Auxiliary relays,
    Battery backed/ latched 4-4
    Device details and example 4-3
    General information on diagnostic devices 4-5
    General use 4-3

B

Basic devices
    Outline of basic PLC devices 2-1
    X, Y, T, C, M, S 2-1
Basic devices and instructions 2-1
BCD data words - reading 4-41
BCD output (Binary Coded Decimal), BCD instr' 5-26
BIN input (Binary), BIN instruction 5-27
Binary data - reading 4-39
Bit devices 4-37
Bit on recognition, BON instruction 5-59
Bit pattern rotation left, ROL instruction 5-44
Bit pattern rotation right, ROR instruction 5-43
Bit rotation and carry left, RCL instruction 5-46
Bit rotation and carry right, RCR instruction 5-45
Bit shift left, SFTL instruction 5-47
Bit shift right, SFTR instruction 5-48
Block data move, BMOV instruction 5-23
Byte swap, SWAP instruction 5-152

C

C data devices
See Counters
Communication Parameters 10-18
Compare: And, AND❒ instruction 5-218
Compare: Load, LD❒ instruction 5-217
Compare: Or, OR❒ instruction 5-219
Comparison of data to a range, ZCP instr' 5-19
Comparison of single data values, CMP instr' 5-18
Compliment of a data value, CML instr' 5-22
Conditional Jump instruction (CJ) 5-5
Constant scan mode - how to program, example 10-4
Constants,
Numeric decimal (K) data value entry 4-14
Numeric Hexadecimal (H) data value entry 4-14
Counters,
16 bit resolution counters 4-20
32 bit resolution bi directional counters 4-21
Basic counters 2-18
Device details and examples 4-19
Ring counters 4-21

D

D data devices
See Data registers
Data registers,
Battery backed/ latched registers 4-32
Device details and examples 4-30
Externally/manually adjustable data registers 4-34
File registers of FX and FX0N PLC's 4-33
General description of diagnostic registers 4-32
General operation of data registers 4-31
Decimal to Gray code, GRY instruction 5-184
Decode data value, DECO instruction 5-56
Decrement data, DEC instruction 5-35
Device terms
Bits, words, BCD and hexadecimal 4-37
Floating Point And Scientific Notation 4-43
Diagnostic devices
Clock devices (M8010-19 and D8010-19) 6-8
Error detection devices (M8060-69, D8060-69) 6-13
High speed counter flags (M8235-55, D8235-55) 6-22
Interrupt controls (M8050-59 and D8050-59) 6-12
Link control (M8070-99 and D8070-99) 6-15
See Also Miscellaneous (M8100-19, D8100-19)
Operation flags (M8020-29 and D8020-29) 6-9
PLC operation mode (M8030-39 and D8030-39) 6-10
PLC status (M8000-9 and D8000-9) 6-7
STL/Annunciator flags (M8040-49 and D8040-49) 6-11
Up/down counter control (M8200-34, D8200-34) 6-22
Digital switch input, DSW instruction 5-104
Division of data values, DIV instruction 5-33
Double coil designation 2-5
Drive to Absolute, DRVA instruction 5-169
Drive ti Increment, DRVI instruction 5-167

E

Encode data, ENCO instruction 5-57
END 2-23
End instruction 2-23
Error codes
    Circuit (D8066) 6-25, 6-26
    Communication (D8062 - D8063) 6-23
    Hardware (D8061) 6-23
    Operation (D8067) 6-27
    Parameter (D8064) 6-24
    Syntax (D8065) 6-24
Example of interrupt use 10-6
Example system application 10-8
Example use of a timer interrupt 10-8
Exchanging data bytes, XCH instruction 5-25
Exchanging data formats
    BCD data to binary data, BIN instr' 5-27
    Binary data to BCD data, BCD instr' 5-26
    Floating point to scientific format, (FNC 18) 5-26
    Scientific format to floating point, (FNC 19) 5-27
Exchanging data values, XCH instruction 5-25
Execution complete flag, using M8029 10-7

F

FIFO data read, SFRD instruction 5-52
FIFO data write, SFWR instruction 5-51
Fill move, FMOV instruction 5-24
Float instruction, FLT 5-64
Floating point - a numbering format 4-45
Floating Point 1
    Float compare (ECMP) 5-135
    Float zone compare (EZCP) 5-136
    Float to scientific (EBCD) 5-137
    Scientific to float (EBIN) 5-138
Floating Point 2
    Float add (EADD) 5-139
    Float subtract (ESUB) 5-140
    Float multiplication (EMUL) 5-141
    Float division (EDIV) 5-142
    Float square root (ESQR) 5-143
    Float to integer (INT) 5-144
Floating Point 3
    Sine (SIN) 5-147
    Cosine (COS) 5-148
    Tangent (TAN) 5-149
Floating point application - summary 4-46
FOR-NEXT loops, FOR, Next instructions 5-14
Forced program end, FEND instruction 5-13
FX0N-3A read, RD3A instruction 5-186
FX0N-3A read, WR3A instruction 5-187
FX1N-5DM Display module 10-29
FX-8AV - externally adjustable data values 4-34
FX-8AV control instructions
    Volume read, VRRD instruction 5-124
    Volume scale, VRSC instruction 5-125
FX1S performance specification 8-1
FX1N performance specification 8-2
FX2-40AP/AW parallel run (PRUN) instruction 5-119
FX2N & FX2NC performance specification 8-4

G

Gray code to Decimal, GBIN instruction 5-185
Grouped bit devices 4-37

H

H value
    See Constants
Hex to ASCII conversion using ASCI (FNC 82) 5-121
Hexadecimal data words - reading 4-40
Hexadecimal keypad, HKY instruction 5-102
Hour meter, HOUR instruction 5-180
Hierarchy of program flow instructions 7-11
High speed counter reset, HSCR instruction 5-74
High speed counter set, HSCS instruction 5-72
High speed counter zone compare, HSZ instr' 5-77
High speed counters,
    1 phase counter - reset and start inputs 4-27
    1 phase counters - user start and reset 4-26
    2 phase bi-directional counters 4-28
    A/B phase counters 4-29
    Available counters 4-24
    Basic operation 4-23
    Counter speeds 4-25
    Glossary and examples 4-22
How to use the manual 1-2
HSZ Instruction
    Combined HSZ and PLSY operation (3) 5-77
    Standard Operation (1) 5-75
    Using HSZ with a data table (operation 2) 5-75

I

I interrupt program pointer
    See Interrupts
Incremental drum sequence, INCD instruction 5-89
Incrementing data, INC instruction 5-34
Index registers,
    Device details and examples 4-35
    General use 4-35
    Misuse of modifiers 4-36
    Modifying a constant 4-36
    Using multiple index registers 4-36
Indexing through display values, example 10-5
Initial state control, IST instruction 5-85
Input, device details and example 4-1
Instruction execution times
    Applied instructions 7-3
    Basic instructions 7-1
Interrupts,
    Counter interrupts 4-13
    Device details and pointer examples 4-11
    Disabling individual interrupts 4-13
    Input triggered interrupt routines 4-12
    Interrupt instructions: IRET, EI, DI 5-10
    Timer triggered interrupt routines 4-12
INV 2-21
Inverse instructions 2-21

K

K value
See Constants

L

LD, LDI 2-3
LDP, LDF 2-8
Load, load inverse instructions 2-3
Load Pulse, load trailing Pulse instructions 2-8

M

M bit device
See Auxiliary relay
Manipulating thumbwheel data (SMOV), example 10-6
Master control and master control reset 2-15
Matrix input sequence, MTR instruction 5-70
MC, MCR 2-15
Mean of a data set, MEAN instruction 5-60
Measuring high speed input pulses
Method using a 1msec timer + interrupts 10-6
Method using M8099, D8099 and interrupts 10-7
Motor control with the PWM instruction 10-15
Move data, MOV instruction 5-20
MPS, MRD, MPP 2-13
Multiple output circuits 2-13
Multiplication of data, MUL instruction 5-32

N

Negation of a data value, NEG instruction 5-39
No operation instruction 2-22
NOP 2-22

O

Or block instruction 2-11
Or, Or inverse instructions 2-7
OR, ORI 2-7
Or Pulse, Or trailing Pulse instructions 2-10
ORB 2-11
ORP, ORF 2-11
OUT 2-4
Timer and counter variations 2-4
Out instruction 2-4
Output, device details and example 4-2

P

P program pointer
See Pointer P
PLC operation - batch processing 7-13
PID control
Applied instruction 88 - PID. 5-126
Configuring the PID loop 5-129
Example program 10-28
PID Setup parameters 5-128
Program techniques 10-24
PLS, PLF 2-20
Pointer P,
Device details and example use 4-10
Positive/negative logic 5-108
Power failure precautions for FX DC units 10-1
Print to display, PR instruction 5-111
Program
How to read ladder logic 2-2
Program scan 2-23
Programming formats: list, ladder, STL/SFC 2-1
What do you need to program a PLC? 1-3
What is a program? 2-1
Program example featuring IST and STL control 10-8
Programmable controller
What is a programmable controller 1-3
Programming tools 1-3
FX-PCS/AT-EE SW operating precautions 3-15
Pulse
Leading and trailing edge instructions 2-20
Pulse Ramp (PLSR instruction) 5-81
Pulse train output, PLSY instruction 5-79
Pulse V, PLSV instruction 5-166
Pulse width modulation, PWM instruction 5-80

R

Ramped values, RAMP instruction 5-93
Reading from special blocks, FROM instruction 5-112
Real time clock data read, TRD instruction 5-178
Real time clock data write, TWR instruction 5-179
Real time clock memory cassettes 9-2
Refresh and filter adjust, REFF instruction 5-69
Refresh I/O status, REF instruction. 5-68
Ripple circuit for use with an inverter 10-15
Rotary table control, ROTC instruction 5-95
RS communications function (FNC80) 5-118

S

S bit device
    See State relays
Scientific Notation - a numerical format 4-44
Search, data search utility - SER instruction 5-87
Set and reset instructions 2-17
    See Also Zone reset, ZRST FNC 40
SET, RST 2-17
Seven segment decoder, SEGD instruction 5-100
Seven segment multiplexed displays 5-107
Seven segment with latch control, SEGL instr' 5-107
Shift move, SMOV instruction 5-21
    Moving BCD data 5-21
    Moving decimal data 5-21
Sort instruction, FNC 69 5-97
Special timer, STMR instruction 5-91
Speed detect, SPD instruction 5-78
Square root, SQR instruction 5-62
State relays,
    Battery backed/ latched 4-7
    Device details and example 4-6
    General use 4-6
    Use as annunciator flags 4-9
    Use as STL step numbers 4-8
Step ladder programming 3-1
    Example, simple STL flow 3-16
    Example, STL selective branch 3-18
    First state merge 3-11
    General STL branching rules 3-14
    How to start and end an STL program 3-3
    Multiple state merge 3-13
    Operational restrictions of some instructions 3-10
    Selective branch 3-11
    Some rules for the writing of STL programs 3-7
    What is STL, SFC and IEC 1131 part 3? 3-1
STL
    See Step ladder programming
Subroutine call, CALL instruction 5-7
Subroutine return, SRET instruction 5-9
Subtraction of data values, SUB instruction 5-31
Sum active data bits, SUM instruction 5-58
Sum checking using CCD (FNC 84) 5-123

T

T data devices
    See Timers
Teaching timer, TTMR instruction 5-90
Ten key keypad, TKY instruction 5-101
Thumbwheels-multiplexed
    See Digital switch input
Time add, TADD instruction 5-176
Time compare, TCMP instruction 5-174
Time subtract, TSUB instruction 5-177
Time zone compare, TZCP instruction 5-175
Timers and counters (out and reset of) 2-18
Timers,

Basic timers 2-18
Device details and examples 4-15
General accuracy 4-18
General timer operation 4-16
Retentive timers 4-17
Selectable range timers 4-16
Timers used in interrupt and CALL subroutines 4-18
Two's compliment - an explanation 4-42

U

Unsuitable instr' for 110V AC input units 7-15
User defined MTR instruction 10-8
Using battery backed/latched devices, example 10-5
Using forced RUN mode (M8035/36/37), examples
Push button configuration 10-2
Remote control with an FX graphic DU unit 10-3

V

V data device
See Index registers

W

Watchdog timer refresh, WDT instruction 5-13
Word AND instruction 5-36
Word data - interpretation 4-39
Word devices 4-39
Word exclusive OR instruction 5-38
Word OR instruction 5-37
Word shift left, WSFL instruction 5-55
Word shift right, WSFR instruction 5-49
Writing to special blocks, TO instruction 5-114

X

X bit device
See Inputs

Y

Y bit device
See Outputs

Z

Z data device
See Index registers
Zero return, ZRN instruction 5-165
Zone device reset, ZRST instruction 5-55

11.2 ASCII Character Codes

Table 11.1:

ASCII code table (HEX)		Higher bit						
		1	**2**	**3**	**4**	**5**	**6**	**7**
Lower bit	**0**	Not accessible	(SP)	0	@	P	@	p
	1		!	1	A	Q	a	q
	2		“	2	B	R	b	r
	3		#	3	C	S	c	s
	4		$	4	D	T	d	t
	5		%	5	E	U	e	u
	6		&	6	F	V	f	v
	7		‘	7	G	W	g	w
	8		(	8	H	X	h	x
	9		)	9	I	Y	i	y
	A		*	:	J	z	j	z
	B		+	;	K	[	k	{
	C		,	<	L		l	|
	D		-	=	M	]	m	}
	E		.	>	N	(SP)	n	~
	F		/	?	O	_	o	$^{C}{}_{R}$

Note:
(SP) = Space,
$^{C}{}_{R}$ = Carriage Return

11.3 Applied Instruction List

	Memonic	Fnc	Page	FX1S	FX1N	FX2NC	FX2N
A	ABS	155	5-127	■	■	■	■
	ABSD	62	5-70	■	■	■	■
	ADD	20	5-25	■	■	■	■
	ALT	66	5-73	■	■	■	■
	ANDq	232-238	5-152	■	■	■	■
	ANR	47	5-47			■	■
	ANS	46	5-47			■	■
	ARWS	75	5-87			■	■
	ASC	76	5-88			■	■
	ASCI	82	5-98	■	■	■	■
B	BCD	18	5-22	■	■	■	■
	BIN	19	5-22	■	■	■	■
	BMOV	15	5-20	■	■	■	■
	BON	44	5-45			■	■
C	CALL	1	5-7	■	■	■	■
	CCD	84	5-100	■	■	■	■
	CJ	0	5-5	■	■	■	■
	CML	14	5-19			■	■
	CMP	10	5-17	■	■	■	■
	COS	131	5-120			■	■
D	DEC	25	5-29	■	■	■	■
	DECO	41	5-43	■	■	■	■
	DI	5	5-9	■	■	■	■
	DIV	23	5-28	■	■	■	■
	DRVA	159	5-132	■	■		
	DRVI	158	5-130	■	■		
	DSW	72	5-83	■	■	■	■
E	EADD	120	5-113			■	■
	EBCD	118	5-112			■	■
	EBIN	119	5-112			■	■
	ECMP	110	5-111			■	■
	EDIV	123	5-115			■	■
	EI	4	5-9	■	■	■	■
	EMUL	122	5-114			■	■
	ENCO	42	5-44	■	■	■	■
	ESOR	127	5-115			■	■
	ESUB	121	5-114			■	■
	EZCMP	111	5-111			■	■
F	FEND	6	5-11	■	■	■	■
	FLT	49	5-49			■	■
	FMOV	16	5-21			■	■
	FOR	8	5-13	■	■	■	■
	FROM	78	5-90		■	■	■
G	GBIN	171	5-147			■	■
	GRY	170	5-147			■	■
H	HEX	83	5-99	■	■	■	■
	HKY	71	5-82			■	■
	HOUR	169	5-143	■	■	■	■
	HSCR	54	5-56	■	■	■	■
	HSCS	53	5-55	■	■	■	■
	HSZ	55	5-57			■	■
I	INC	24	5-29	■	■	■	■
	INCD	63	5-71	■	■	■	■
	INT	129	5-116			■	■
	IRET	3	5-9	■	■	■	■
	IST	60	5-67	■	■	■	■
L	LDq	224-230	5-151	■	■	■	■
M	MEAN	45	5-46			■	■
	MOV	12	5-18	■	■	■	■
	MTR	52	5-54	■	■	■	■
	MUL	22	5-27	■	■	■	■

	Memonic	Fnc	Page	FX1S	FX1N	FX2NC	FX2N
N	NEG	29	5-31			■	■
	NEXT	9	5-13	■	■	■	■
O	Orq	240-246	5-153	■	■	■	■
P	PID	88	5-102		■	■	■
	PLSR	59	5-63	■	■	■	■
	PLSV	157	5-129	■	■		
	PLSY	57	5-61	■	■	■	■
	PR	77	5-89			■	■
	PRUN	81	5-96	■	■	■	■
	PWM	58	5-62	■	■	■	■
R	RAMP	67	5-73	■	■	■	■
	RCL	33	5-36			■	■
	RCR	32	5-36			■	■
	RD3A	176	5-148		■	■	■
	REF	50	5-53	■	■	■	■
	REFF	51	5-53			■	■
	ROL	31	5-35			■	■
	ROR	30	5-35			■	■
	ROTC	68	5-75			■	■
	RS	80	5-95	■	■	■	■
S	SEGD	73	5-84			■	■
	SEGL	74	5-85	■	■	■	■
	SER	61	5-69			■	■
	SFRD	39	5-39	■	■	■	■
	SFTL	35	5-37	■	■	■	■
	SFTR	34	5-37	■	■	■	■
	SFWR	38	5-39	■	■	■	■
	SIN	130	5-119			■	■
	SMOV	13	5-18			■	■
	SORT	69	5-77			■	■
	SPD	56	5-60	■	■	■	■
	SQR	48	5-48			■	■
	SRET	2	5-8	■	■	■	■
	STMR	65	5-72			■	■
	SUB	21	5-26	■	■	■	■
	SUM	43	5-45			■	■
	SWAP	147	5-123			■	■
T	TADD	162	5-139	■	■	■	■
	TAN	132	5-120			■	■
	TCMP	160	5-137	■	■	■	■
	TKY	70	5-81			■	■
	TO	79	5-91		■	■	■
	TRD	166	5-141	■	■	■	■
	TSUB	163	5-140	■	■	■	■
	TTMR	64	5-72			■	■
	TWR	167	5-142	■	■	■	■
	TZCP	161	5-138	■	■	■	■
V	VRRD	85	5-101	■	■	■	■
	VRSC	86	5-101	■	■	■	■
W	WAND	26	5-30	■	■	■	■
	WDT	7	5-12	■	■	■	■
	WOR	27	5-30	■	■	■	■
	WR3A	177	5-148		■	■	■
	WSFL	37	5-38			■	■
	WSFR	36	5-38			■	■
	WXOR	28	5-31	■	■	■	■
X	XCH	17	5-21			■	■
Z	ZCP	11	5-17	■	■	■	■
	ZRN	156	5-128	■	■		
	ZRST	40	5-43	■	■	■	■

PROGRAMMING MANUAL II

THE FX SERIES OF PROGRAMMABLE CONTROLLER
(FX_{1S}, FX_{1N}, FX_{2N}, FX_{2NC})

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE: MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100-8310
HIMEJI WORKS: 840, CHIYODA CHO, HIMEJI, JAPAN

MODEL	FX-P2-E
MODEL CODE	09R512

JY992D88101D
(MEE)

Effective Apr. 2003
Specification are subject to change without notice.