

Code Execution and Boot-Up

The Controller can execute both downloaded program code and incoming serial port commands at the same time. Priority is given to Serial Communications to allow Host level control and override of downloaded code if required. With the exception of program flow commands, just about all Animatics SmartMotor commands may be run via serial port.

The Controller executes its downloaded program within a 500 millisecond window of power up. This window allows the user to abort program execution via serial port if necessary.

Serial Data parsing capabilities allows the controller to communicate with non-Animatics products such as Barcode Readers, Serial Encoders, Light Curtains, etc.

The Controller can therefore act at Master Controller for an entire machine.

Basic Program Flow capabilities:

The program language is similar to C or Basic. It is a Text Based language with the ability to handle up to 1000 subroutines. Decision Making Flow control allows for a broad range capability in machine control.

IF ELSEIF ELSE ENDIF structure:

```
IF a<b
    PRINT("a is less than b",#13)
ELSEIF q==123
    PRINT("q equals 123",#13)
ELSE 'if no condition above was true
    PRINT("nothing above was true",#13)
ENDIF
```

WHILE LOOP structure

Example of a loop that will execute 10 times:

```
a=0
WHILE a<10
    a=a+1
LOOP
PRINT("loop code executed 10 times",#13)
```

GOTO, GOSUB structure:

```
C1
IF a>b
    GOTO1
ELSEIF b>c
    GOSUB5
ENDIF
GOTO6

C5
PRINT("b is greater than c",#13)
RETURN
C6
END
```

SWITCH CASE BREAK structure:

```
SWITCH a
CASE 1
    PRINT("a=1",#13)
    BREAK
CASE 2
    PRINT("a=2",#13)
    BREAK
DEFAULT
    PRINT("a does not equal 1 or 2",#13)
    BREAK
ENDS
```

Data Logging and Error Handling:

The Controller can have a fault interrupt handler store status bits in non-volatile memory, then with another subroutine, recall the logged totals even in the event of power loss:

```
C1 'Interrupt Fault Routine (applies to PLS firmware ONLY)
aa=aa+Be 'Trap Pos. Error Bit
bb=bb+Bh 'Trap Over Temp Bit
cc=cc+Ba 'Trap Over Current Bit
dd=dd+Bo 'Trap Motor-Off Bit
EPTR=100 'Set EPROM Pointer
VST(aa,4) 'Store 4 consecutive variables
```

RETURN

```
C3 'Get latest Status Bit totals
EPTR=100
VLD(aa,4) 'Load 4 consecutive variables
PRINT("Error Bit Totals",#13)
PRINT("Be:",aa," (Position Error)",#13)
PRINT("Bh:",bb," (Over Temperature)",#13)
PRINT("Ba:",cc," (Peak Over Current)",#13)
PRINT("Bo:",dd," (Motor OFF)",#13)
```

RETURN

I/O Port Handling:

Each Controller has 7 User Definable I/O pins that can be assigned as General Input or Outputs.

Additionally they can be read as 10-bit analog inputs at any time.

The I/O pins are called Port Pins and are as follows:

Pin	Port	Additionally Assignable as:
1	A	Phase A Enc. Input, or Step (Pulse) Input
2	B	Phase B Enc. Input, or Direction Input
3	C	Positive Over Travel Limit
4	D	Negative Over Travel Limit
5	E	RS-485 A or AniLink Clock (I2C)
6	F	RS-485 B or AniLink Data (I2C)
7	G	Synchronize Input or "G" (GO) command

Each Controller Includes Dedicated Encoder Output Pins for ease of Master-Slave Connections between two controllers.

Mode Follow (Electronic Gearing) and

Mode-Step (Pulse and Direction) both allow the use of 24-bit resolution ratio of input count to motor slave count gearing.

Controller Addressing:

Up to 100 unique addresses per single RS-232 or RS-485 network are possible.

RS-232 is connected via Serial Daisy chain where the Transmit of one controller is connected to the Receive of the next. Each controller will ECHO out incoming data to downstream controller.

RS-485 meets the IEE standard allowing parallel connection to all controllers on the bus.

Commands may be sent individually proceeded by Controller Addresses or Globally to all controllers at once via global addressing.

Ethernet, DeviceNet, CAN Open and ProfiBus Gateway Options access the controllers via their secondary RS-485 Port. Please consult the Animatics Users Guide for full details.