MODEL 135
16-BIT 150KHz A/D CONVERTER

LAWSON LABS, INC.
3217 Phoenixville Pike
Malvern, PA 19355

610 725-8800
or
800 321-5355

Rev. B

TABLE OF CONTENTS

INTRODUCTION

The Lawson Labs Model 135 A/D card enables an IBM PC, XT or AT computer to translate continuously variable analog voltages into their digital equivalents of ones and zeros. It also provides a means for the computer to turn external equipment on and off. These functions are the basics of monitoring and control systems. Both the analog inputs and the digital outputs are protected against overvoltages. The analog input has sample and hold capability to facilitate reading rapidly changing input signals.

NOTE: This manual discribes operation under DOS. See the readme file in the WIN135 directory on your support disk for WINDOWS instructions.

SECTION 1. INSTALLATION

NOTE: Always handle circuit cards by the edges.

There are seven DIP switches located near the bottom edge of the Model 135. These switches determine the base port address of the card. The base address can be placed virtually anywhere in the available I/O space, including places reserved by the operating system. We recommend addresses between 544 and 624 decimal.

The switch positions for base address 600 are 3,5 and 6 ON; 1,2,4 and 7 OFF. If there are other special purpose cards in your computer you should check their documentation to make sure the port addresses do not conflict. Although the Model 135 base address is a multiple of 8, only the first 4 addresses are used. The upper 4 of the 8 addresses are still available to other boards. Possible base addresses range from 256 (100 Hex) to 998 (3E6 Hex).

FIGURE 1. Recommended Base Address DIP Switch Selection Chart

			1= ON			0= OFF		
Base address	Hex	7	6	5	4	3	2	1
544	220	0	1	1	1	0	1	1
552	228	0	1	1	1	0	1	0
560	230	0	1	1	1	0	0	1
568	238	0	1	1	1	0	0	0
576	240	0	1	1	0	1	1	1
584	248	0	1	1	0	1	1	0
592	250	0	1	1	0	1	0	1
600 (standard)	258	0	1	1	0	1	0	0
608	260	0	1	1	0	0	1	1
616	268	0	1	1	0	0	1	0
624	270	0	1	1	0	0	0	1

To obtain the base address from the DIP switches, use 0 for ON and 1 for OFF in the following formula: SW7 * 64 + SW6 * 32 + SW5 * 16 + SW4 * 8 + SW3 * 4 + SW2 * 2 + SW1. Multiply the above by 8 to obtain the base address. (Note that ones and zeros in the formula are the reverse of the table.) The standard switch settings select board address 600 (258 Hex).

Static electricity can damage computer circuitry so care should be taken to control static discharge. Turn off the computer but do not unplug it. Remove the cover. First remove the screw which secures the bracket behind a vacant slot. Then remove the bracket. While holding the Model 135 in one hand touch the computer's chassis with the other. This procedure will harmlessly drain any static charge that may be present. Now insert the card into the connector. Press down firmly to insure that the card is seated and replace the screw in the bracket.

Attach a pair of wires to pins 6 and 13 of the mating 15-pin connector. The negative input is pin 13 and the positive input is pin 6. The other end of the wires should connect to the voltage being measured. A 1.5 volt battery is a convenient voltage source for test purposes. For a battery or other floating voltage source, a third connection must be made to insure that both inputs are within 6 volts of chassis ground at the computer. A wire from the negative side of the measured voltage to the computer's chassis is the best way to ground for most purposes.

NOTE: Strain-relief is recommended for all permanent wiring on the connector. Otherwise, physical stress may cause the failure of an electrical connection. The connector hood provided has a built-in strain-relief clamp.

FIGURE 2. CONNECTOR PINOUT

PIN 8...CHANNEL 1+
PIN 15...CHANNEL 1-
PIN 7...CHANNEL 2+
PIN 14...CHANNEL 2-
PIN 6...CHANNEL 0+
PIN 13...CHANNEL 0-
PIN 5...CHANNEL 3+
PIN 12...CHANNEL 3-
PIN 4...GROUND
PIN 11...DIGITAL INPUT
PIN 3...GUARD
PIN 10...OUTPUT C
PIN 2...OUTPUT D
PIN 9....OUTPUT B
PIN 1...OUTPUT A

Attach the connector and tighten the small screws that hold the halves together. Replace the cover. Turn on the power to the computer. If the power-on self-test (POST) does not behave normally, turn off the computer and refer to the troubleshooting section of this manual.

Connect the plus and minus wires to the voltage being measured. Make certain that the voltage is in the +/- 5 volt range. Remember that the positive and negative inputs on the Model 135 should both stay within 6 volts of ground potential. See FIGURE 4.

NOTE: For maximum reliability, connect all unused inputs to ground.

SECTION 2. PERFORMING A/D CONVERSIONS

In the absence of other instructions, the A/D card will measure the voltage at input channel #0. If the wires that you attached are connected to input #0, we need not concern ourselves with the channel selection process at this time. In essence, one INPut instruction begins the conversion. Six microseconds later the answer is stored on the Model 135 board where it remains until it is read with two INPut instructions.

The Model 135 uses four port addresses. An OUT statement to any of the four addresses in the range is equivalent. Reading the lowest port in the range (ADC) provides status information. Reading the next address yields the less significant byte of the answer. Read the third address for the more significant byte of the answer. A new conversion is started when the high byte is read. The highest port address is not used. Change the ADC (Analog to Digital Converter) address in line 10 of the following BASIC program if an alternate switch setting is used.

```
  10 ADC = 600
  20 X = INP(ADC+2): REM start conversion
  30 LOW = INP(ADC + 1)
  40 HIGH = (INP(ADC + 2) XOR 128): REM read the answer
  50 COUNT = LOW + HIGH * 256
  60 VOLTS = (COUNT - 32768) * 1.52588E-04: REM scale in volts
  70 LOCATE1,1: PRINT USING "##.####"; VOLTS
  80 IF INKEY$ = "" THEN GOTO 30 ELSE END
```

Line 10 assigns a variable name to the port address. Line 20 begins the conversion. Line 30 reads the less significant byte of the answer. Line 40 reads the more significant byte. Line 50 combines the bytes into a count from 0 to 65535. Notice that the less significant byte of the answer (ADC + 1) must be read first because reading the high byte starts a new conversion. Also the more significant byte is XOR'ed with 128 to change it from 2's complement format. Line 60 scales the reading into volts. Line 70 displays the result and line 80 repeats the process.

RUN the program. (Press any key to exit.) The voltage should appear in the upper left-hand corner of the screen. The Model 135 Analog Interface resolves 0.153 millivolts. A fresh D cell should read about 1.5 volts. With reasonably short wires, or with shielded wire, expect to see about +/- 3 counts (.46 millivolts) of scatter. There is virtually no filtration in the Model 135 so any noise on the input will appear as variability in the reading. Connecting both of the input wires to ground will cause a potential of zero volts. The reading with the input shorted should be within a millivolt of zero (32768 counts). An open circuit will read unpredictably. An overvoltage will read 4.9999 or -5.0000. The input channels are protected against transient overvoltages of up to 125 volts and continuous overvoltages of 65 volts. Reverse the wires and note the polarity change.
The plus sign is implicit and does not appear on the screen.

The conversion itself takes 6 microseconds. Reading the high byte of the answer also initiates a conversion. In the above program example, line 20 need only execute once because a new conversion begins each time the high byte is read in line 40. In BASIC, and in all but the fastest computer languages, the conversion will be over before the answer can be read. Routines that take advantage of the Model 135's speed must be written in fast languages. Those routines poll a status byte at address ADC to determine when the conversion has been completed.

SECTION 3. SELECTING OTHER INPUT CHANNELS

First connect the other voltage to be measured to an unused pair of input pins on the Model 135's connector. See Figure 1 and Figure 4. The four available channels are numbered 0 through 3. To select a channel, substitute the channel number to be selected for the "N" in this instruction   OUT ADC, 16 * N.

Here is a sample routine which will measure the voltage at each channel and print the data. The routine is called DEMO135.BAS and is included on the support disk.

```
   10 KEY OFF: DIM MAX(4), MIN(4)
   20 CLS
   30 INPUT "Enter ADC (check your manual) "; ADC
   40 IF ((ADC < 544) OR (ADC > 626)) THEN PRINT
         "544 <= ADC <= 626": GOTO 30
   50 START = ADC + 2: CLS
   60 SMOOTH = 30: REM number of readings to average
   70 LOCATE 24, 30: PRINT "Press Esc to exit"
   80 LOCATE 3, 31: PRINT "Volts      Max        Min"
   90 FOR N = 0 TO 3: MAX(N) = -5: MIN(N) = 5: NEXT: REM Init.
   100 LOCATE 5, 1
   110 FOR CHANNEL = 0 TO 3
   120 OUT ADC, CHANNEL * 16: REM select channel
   130 COUNT = 0: FOR S = 1 TO SMOOTH
   140 X = INP(START): REM start a conversion
   150 IF (INP(ADC) AND 2) = 0 THEN GOTO 150: REM status bit
   160 LOW = INP(ADC + 1)
   170 HIGH = (INP(ADC + 2) XOR 128): REM read the answer
   180 COUNT = COUNT + LOW + HIGH * 256
   190 NEXT S
   200 COUNT = COUNT / SMOOTH
   210 VOLTS = (COUNT - 32768!) * 1.52588E-04: REM scale in volts
   220 IF VOLTS > MAX(CHANNEL) THEN MAX(CHANNEL) = VOLTS
   230 IF VOLTS < MIN(CHANNEL) THEN MIN(CHANNEL) = VOLTS
   240 PRINT "Channel"; CHANNEL; " count = ";
   250 PRINT USING "#####"; COUNT;
   260 PRINT USING "    ##.####"; VOLTS; MAX(CHANNEL);
         MIN(CHANNEL)
   270 NEXT CHANNEL
   280 PRINT : PRINT "Digital input is ";
   290 IF INP(ADC) AND 128 THEN PRINT "OFF" ELSE PRINT "ON "
   300 IF INKEY$ = CHR$(27) THEN END
   310 GOTO 100
```

Running the DEMO135 program with no connections made on channels 1 through 3 will do no harm. However, the data from those channels will be meaningless. Make a back-up copy of the support disk. Start BASIC, BASICA or QBASIC and put the working copy of the support disk in drive A and type RUN "A:DEMO135". Provide the ADC address when prompted. You will see the raw count and voltage at each of the channels displayed on the screen. The status of the digital input is also displayed. See the next section for a description of the digital input.

The value of 30 for SMOOTH in line 60 yields a readable screen on the fastest machines, and +/- 1 count on stable DC signals. Smaller values of SMOOTH increase execution speed while larger values remove noise from DC signals. The FOR NEXT loop from line 110 to line 270 steps through the channels and displays the readings. Notice that the conversion is explicitly started in line 140. This step makes it clear that the answer obtained in line 160 AND 170 came from the input channel selected in line 120. Notice that after the program has run, channel 3 is still selected. Turning the power off and on or OUT ADC, 0 will select input channel zero.

SECTION 4.   USING THE CONTROL OUTPUTS & DIGITAL INPUT

There are four optically isolated output lines on the A/D card. These are intended primarily for expanding the number of input channels using an external multiplexer. Any channel can be expanded to 16 channels with external multiplexing. The optically isolated control lines prevent ground loops from being formed. Configurations of up to 64 analog inputs are possible with four Model 17 Multiplexers. The outputs can also be used for digital control. Think of each one as a contact closure to guard.

The output port has the same address as the input channel selector. The lower four bits appear, optically isolated, at the connector. A number from 0 to 15 sent to port ADC will appear on the output lines, coded in binary. The instruction OUT ADC, 3 will cause outputs A and B to become active. Figure 3 is a list of codes and the corresponding active outputs. Note that the sequence is that of counting in binary, 1=active.

FIGURE 3. OUTPUT CODES

CODE	OUTPUT			
	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

If it is necessary to alter the output code without affecting the input channel selection, the input selection code should be added to the output code. For example, to select input channel 1 with output code 15 use OUT ADC, 16 + 15 or OUT ADC, 31.

Electrically, each output line is the collector of an optically coupled NPN transistor whose emitter is connected to guard. Note that guard is not connected to ground on the Model 135. The guard is usually connected to the external device's ground. Each output line can drive two low-power TTL loads or many CMOS loads. An external pull-up resistor (from the collector to the external positive supply) must be used. Typically, 4.7 K ohms for low-power TTL or 20 K ohms for CMOS is appropriate. When the output code is 0, all four phototransistors are off and no current is drawn through the transistors. If the input to an inverting buffer is connected to output line A (with the appropriate pull-up resistor), the input level is forced high by the resistor so the buffer output would be 0. If the output code is 1, the A transistor will turn on and pull the buffer input to a logic level 0. The buffer output will then be a logic level 1.

The digital input can be a contact closure to ground, a TTL or 5-volt CMOS logic level, active low. This line reads the digital input.

```
     IF INP(ADC) AND 128 THEN PRINT "OFF" ELSE PRINT "ON "
```

SECTION 5.   USING THE FAST SAMPLING SOFTWARE

The DRIVR135 software is unprotected and fully listable. It can function as a self-contained data acquisition system or it can be included, all or in part, in your QuickBasic 4.5 programs. DRIVR135 is written in QuickBasic 4.5 and assembly language.

Please note that this DRIVR135 will not work with earlier versions of QuickBasic, or interpreted BASIC. Most of the QuickBasic 4.5 code is the input section which allows you to define a sample run. That definition is contained in an integer array called SAMPLE. The array is dimensioned at 90 integer elements. The address of SAMPLE(1) is passed to the assembly code in a CALL instruction. When CALLED, the specified pattern of data points are stored in a block of memory. The data can be moved to a disk as a sequential text file suitable for importing by Lotus 1 2 3 (reg. trademark) or can be saved as a binary image using the BSAVE command.

To start the DRIVR135.EXE code, first make sure DRIVR135.EXE and DRIVR135.BIN are in the current directory, then, from the DOS prompt, type DRIVR135. To start the DRIVR135 source code under QuickBasic 4.5 type the following line:

```
                QB DRIVR135/AH/LBURST135
```

This will load DRIVR135.BAS with the BURST135.QLB library. To execute DRIVR135 press F5.

DRIVR135 begins by loading a binary file from the disk, DRIVR135.BIN. DRIVR135 will look only in the default directory for that file. A beep indicates that the DRIVR135.BIN file was not found. DRIVR135.BIN is a bit image of the SAMPLE array. The last SAMPLE array that was saved on the disk will be the starting point for the session. If you mis-enter something, keep going and you will be given a chance to correct the mistake. If DRIVR135 cannot find DRIVR135.BIN, when it begins, you will start at 1), otherwise you will start at 8).

1) Enter ADC (see Section 2).

2) The interrupts may be disabled during the sample run. Interrupts can cause irregularities in the sampling interval. If the interrupts are disabled functions running in the background such as ON TIMER GOTO will be disrupted. Also, the time of day clock doesn't run when the interrupts are off.

3) From 1 to 64 data points can be taken at each scan. The data points can be taken in any sequence. For example, a scan could consist of channel 3 being sampled 3 times followed by channel 0 being sampled once. At this time, enter the number of points per scan desired.

4) For multiple channel routines, a brief delay after switching channels may be desirable. This pause allows extra time for voltages to settle for maximum accuracy. If you are using an

external multiplexer, a delay of at least 0.5 milliseconds will be needed for maximum accuracy. If you enter 0 milliseconds, then no delay is used. If you enter a delay greater than 26 or less than .006 milliseconds, you are prompted again.

NOTE: As you approach 150KHz total throughput for multi-channel acquisition, DC accuaracy will be degraded due to settling time limitations.

5) Up to 256 samples can be averaged to obtain each data point. This feature is useful for compressing slowly changing data and for noise suppression. Note that very noisy, highly averaged data, even if it is reproducible, may suffer decreased accuracy. The number of samples to average must be a power of two. If a number besides 1, 2, 4, 8, etc. is entered, a power of two will be proposed. Now if anything has been changed (or you just started DRIVR135) there will be a pause while timing is being determined.

6) Note that rates from 1 to 150,000 scans/second are possible. Normally rates slower than one scan/second can be programmed in a high level language (or use Lawson Labs PC64 software).

7) If averaging = 1, and the channel delay equals 0, then skip to step 8). Otherwise the computer timer is used to time the burst. This provides a time base for channel delays, and scan rates. You can accept the scanning rate by pressing "Y", or enter a new rate by pressing "N". Go to step 9). The program always tries first to use the timer based burst. If the timer based burst is not fast enough to achieve the desired rate, the high-speed burst routines are used. Note averaging and inter-channel delays are not available in the high-speed routines.

8) Next the program tries to achieve the scanning rate within 0.1%. If this cannot be achieved, the actual scanning rate is displayed. You can accept the scanning rate by pressing "Y", or enter a new rate by pressing "N".

9) If you enter a scanning rate greater than the fastest available rate for your configuration, the program will try to achieve that rate or prompt you for a slower rate.

10) Enter the number of scans per sample run. If you enter too large a number, you will be shown the maximum number of scans that will fit in available memory.

11) Now a review of the sampling run is presented. Press "C" to continue if all is correct. Press "M" to modify anything in the summary screen. Press "ESC" to exit.

12) If you want to load a previously BSAVEd data file answer "Y" to the prompt. Note that the file to be loaded should have the same number of points per scan and the same number of scans as the current configuration. You will be prompted for the file name, then skip to paragraph 16.

13) Finally, the control code for each data point must be entered (see Sections 3 & 4). DRIVR135 displays each control code in turn. If a proposed control code is correct, just press RETURN to leave it unchanged. You will be given a chance to review or

revise the control codes after the whole scan has been defined. Holding down the RETURN key will speed you through the review.

14) If you don't want the sampling routine to execute, Press "R" at the prompt. Otherwise press SPACE to begin sampling. You may want to save your sampling run definition on the disk before attempting to execute it. To do that, press "R" now and skip to step 18).

15) The "Sampling complete" message will appear after the data has been taken.

16) If you want your data saved as a text file answer "Y" to the prompt and provide the full file name specification including the drive designation and file name extension, if any. If you want to import your data into Lotus 1 2 3 (reg. trademark) you will need to use a .PRN extension. The text file is a standard ASCII sequential file with one entry for each data point and a new line for each scan. Versions vary, but Lotus may allow as few as 34 data points/scan and as few as 2048 scans/data run. The text files can also be read from a separate program or you can use the DOS TYPE command to review the data. With the DOS prompt, typing TYPE filename > LPT1 will send the whole file to your printer.

17) The data in volts will now begin scrolling down the screen. Press "SPACE" to pause. Press "SPACE" again to continue scrolling or press "ESC" to quit.

18) You will be offered a chance to save the sampling run definition on the disk. If you save it, the previous definition will be overwritten and the next time you run DRIVR135 you will start from the current status. If you don't save it, once you exit the program your current definition will be gone.

19) Now the process is repeated starting at step 11). If you want to save the data as a binary file, you should exit now and issue the BSAVE command(s) from the QB environment. To BSAVE your data, first type DEF SEG = X/16 where X is the start of your data table. Then type BSAVE filename, 0, TABLELENGTH. Use DEF SEG to restore the default data segment.

If you want to incorporate a sampling run into a BASIC program of your own, it isn't necessary to include all of DRIVR135 in your program. Instead you can generate and save the SAMPLE array using DRIVR135, then BLOAD that array into your application program. The uses for all the elements in the SAMPLE array are described in the DRIVR135.BAS program listing. There should be enough information for you to make small changes in the sample run by changing the SAMPLE array in your application program. You may want to use the digital input or an extra input channel as a trigger for synchronizing the sampling run.

In your application program be sure that the SAMPLE array is dimensioned as shown in DRIVR135.BAS lines 1110 and 1120. Then you must also allocate memory for data storage. DRIVR135.BAS allocates all but a small fraction of available memory for data storage. The subroutine at line 3820 performs the allocation. The use of a huge array, requires the /AH command line parameter when starting QuickBASIC 4.5. When you have thoroughly tested your program in the QuickBASIC 4.5 environment, you can generate an executable file using the ALT-R command.

NOTE: Be sure you started QuickBASIC 4.5 with the /AH/LBURST135 command line parameters. The BURST135.QLB file (containing the assembly language routines) should be in the current directory.

Only if you are familiar with assembly language programming should you attempt to modify any of the routines in BURST135.ASM. The following are the commands to type in at the DOS prompt to generate a new DRIVR135.EXE file after modifying DRIVR135.BAS and BURST135.ASM. It assumes BURST135.ASM, MASM.EXE, LINK.EXE, LIB.EXE, BC.EXE, BQLB45.LIB, BCOM45.LIB, AND DRIVR135.BAS are all in the default directory:

```
MASM BURST135/Z;
DEL BURST135.LIB
LIB BURST135.LIB+BURST135.OBJ;
LINK /Q BURST135,,,BQLB45.LIB;
BC DRIVR135/O/X/E/AH;
LINK DRIVR135,,,BCOM45.LIB+BURST135.LIB;
```

The first three lines create the BURST135.LIB file. The fourth line creates the BURST135.QLB file. The fifth and sixth lines compile and link DRIVR135.BAS. BQLB45.LIB, LIB.EXE, BC.EXE, and LINK.EXE come with QuickBASIC 4.5. MASM.EXE is an assembler, and is only needed if you modify BURST135.ASM.

If you have your own BASIC code and want to call the routines in BURST135.LIB, use the name of your code in place of DRIVR135 in the above lines to generate an executable program.

There are 4 callable routines in the BURST135 library. READCNT100 is used to determine the overhead of calling BURST1, and BURSTS2. The time it takes to run is subtracted from the total burst run time to establish the actual burst run time.

BURST1 and BURSTS2 use delay loop based timing. The calling routine must determine the delay loop value for a particular scan rate. This value will vary with machine speed and number of data points.

BURST1 is called only when the burst setup has one data point per scan, no channel delay, and no averaging. This provides the fastest scan times up to the limits of the Model 135 card.

BURSTS2 is called only when the burst setup has more than one data point per scan, no channel delay, and no averaging. You may need to run slightly slower than the maximum rate to allow for settling time.

BURSTS is the most flexible, but slowest routine. DRIVR135 uses it whenever possible. BURSTS is always called if averaging or channel delays are used. This routine uses the computer's timer for a time base and has a resolution of one microsecond.

SECTION 6. USING MACHINE LANGUAGE

Using the Model 135 from machine language requires checking the status byte to determine if the conversion is complete. The second-to-least significant bit of the status byte is reset to zero by a start conversion and set to one after the conversion is complete.

Assume the DX register contains ADC. The normal sequence to read the card would be:

```
      INC  DX
      INC  DX
      IN   AL, DX     ;Start conversion
      DEC  DX
      DEC  DX         ;Point DX at status
TST:  IN   AL, DX     ;Read status
      AND  AL, 02     ;Mask conversion complete bit
      JZ   TST        ;Wait for complete conversion
      INC  DX         ;Point DX at ADC + 1
      IN   AX, DX     ;Read both bytes
      XOR  AH, 80H    ;Convert from 2's compliment
```

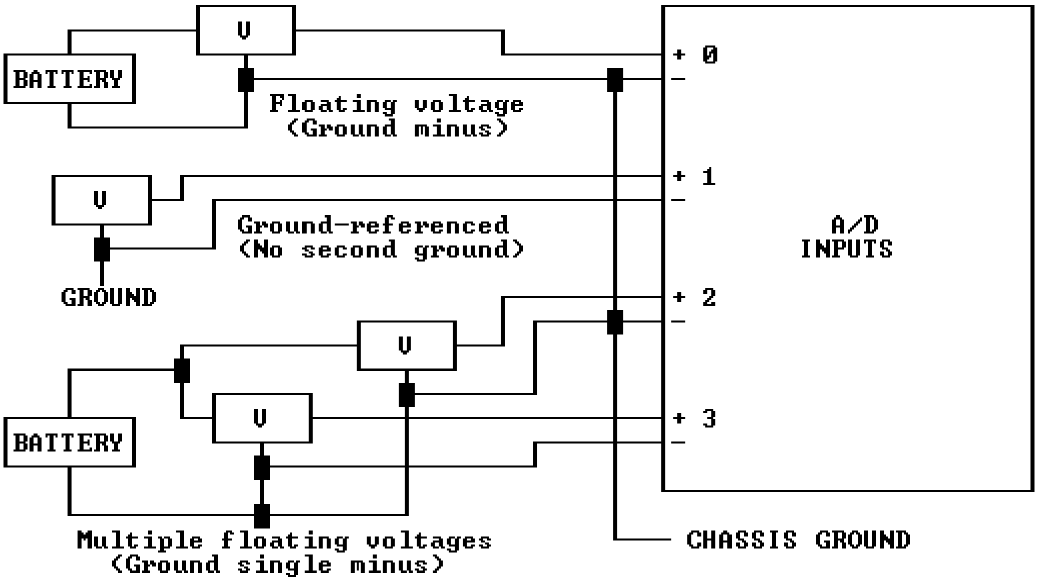
Remember that the IN AX,DX starts a new conversion. If you want that next conversion to be at a different input channel, the new control code should be output just after the previous start conversion.

SECTION 7. OPERATING SUGGESTIONS

The Model 135 A/D card features fully differential inputs. A basic understanding of differential measurements will help you to use your card to best advantage. The plus and minus input pins should be wired directly to the voltage being measured. In this way, it is assured that the only current flowing in the wires will be the input current of the A/D converter. Because the wires have finite resistance, any current flowing will cause a voltage drop and a corresponding error. The A/D card requires a vanishingly small input current so the error caused by even very long wires is negligible. For proper operation of the A/D card it is necessary that the positive and negative inputs both be within 5 volts of ground. For a floating voltage source, this is generally accomplished by connecting a third wire between the computer chassis and a ground terminal at the source of the measured voltage. Ground currents may flow in this wire, but the resulting voltage drop will not cause a measurement error. If the source of the voltage being measured is a line-connected device with a three-pronged plug, its output voltage may or may not be referenced to ground. In that case, try with and without a ground connection for best performance. Redundant grounding can cause ground loops. Ground loops can cause unpredictable behavior.

NOTE: The Model 135 may appear to behave properly most of the time with ungrounded floating input sources. Eventually one or more incorrect readings will occur when the input floats to the limit of the common mode range.

FIGURE 4. TYPICAL INPUT CONNECTIONS



The Model 135 filters very high frequency noise only. Any noise below 1 megahertz will cause increased scatter. If your voltage source is noisy, scatter will result. Best results are obtained with filtered, low impedance voltages. Electrical noise travels through the air and can be picked up by interconnecting wires. The first defense against noise is shielding. Use shielded wire with the shield connected at one end to ground. (See above). The lower the impedance of the voltage source, the less susceptible the wiring will be to electrical noise. If noise problems persist, try to locate the source of the interference and shield it. Electric motors, electric heaters and flickering fluorescent lights are potential sources of interference.

SECTION 8. TROUBLESHOOTING

GENERAL: If the A/D converter appears not to be working, first check all external wiring and all port addresses. Double check the switch settings on the Model 135. Be sure that all input voltages are within range. If another special function card is present, check for conflicting port addresses. Also, try adding a wire from the negative input to chassis ground. Make certain you are not using guard instead of ground.

1: Results seem to be meaningless numbers

The most likely culprit is a loose wire or an input with nothing attached. Try connecting the + and - inputs to ground. If the voltage does not read close to 0, the problem is in the wires themselves or in the channel selection. Check that the program has been correctly entered.

2: The answers are consistent but wrong

The ADC port address is wrong or in conflict with another card. Also, make sure all inputs are within range.

3: Readings show excessive scatter

First check for proper grounding. If there is no connection between the negative input and ground, try one. If there is such a connection, try removing it.

If the card reads a 1.5 volt battery properly but still returns widely varying numbers when measuring a voltage, the voltage could be oscillating or noisy. If you check the voltage with another voltage-measuring device remember that some meters will average out oscillations or noise to produce a steady reading. Try switching to an alternating current scale. The A/D card will filter out noise above a megahertz but any noise below those frequencies will cause increased scatter.

SECTION 9. INTERNAL ADJUSTMENTS

The two adjustments on the Model 135 are gain and offset controls. The gain adjustment varies the reference voltage used during the A/D conversion. The long-term stability of the initial calibration is excellent. If you decide to alter the original gain setting, be sure your voltage reference or test meter is sufficiently accurate and has been calibrated recently. Remember that if you change the gain setting you may not be able to restore the converter to its original accuracy.

The adjustments are at the top edge of the A/D card. The potentiometer nearer the A/D card mounting bracket is the offset adjustment. It is best to make any adjustments with the card at its normal operating temperature. First set the offset for zero volts with both inputs connected to ground. Then adjust the gain to match a known voltage.

SECTION 10. SPECIFICATIONS

TYPE:	successive approximation with sample and hold
DIFFERENTIAL INPUT RANGE:	+/- 5 volts DC
COMMON MODE INPUT RANGE:	+/- 6 volts DC
RESOLUTION:	0.153 millivolts (monotonic)
INTEGRAL NONLINEARITY:	.005% of full scale typical
ACCURACY:	0.03% of full scale typical
INPUT IMPEDANCE:	100,000 megohms typical
CONVERSION TIME:	6 microseconds
THROUGHPUT:	150,000 samples/second (using a 33MHz 486 computer)
REPRODUCIBILITY:	+/- 4 counts
ANALOG INPUTS:	4 fully differential channels

	overvoltage protected up to 100 volts transient or 50 volts continuous
DIGITAL INPUT:	single-bit, contact closure or 5-volt logic compatible
DIGITAL OUTPUTS:	4 latched optically isolated lines protected to 200 volts
TYPICAL POWER CONSUMPTION:	+5v .... 50 milliamps +12v .... 100 milliamps (after warmup) -12v .... 65 milliamps
SIZE:	3.9 x 5.2 inches, 0.6 inches max. height
INCLUDED SOFTWARE:	DRIVR135 software allows up to 64 channels of data to be acquired at high speed. Features include digital averaging, flexible sequencing and selectable inter-channel delay.

LIMITED WARRANTY

The Lawson Labs, Inc. Model 135 Analog Interface is guaranteed against defects in materials and workmanship for a period of one year from the date of delivery. Products must be returned to Lawson Labs for warranty service. Contact Lawson Labs at 800 321-5355 for return authorization before returning anything for service.

The above warranty is in lieu of all warranties express or implied. Lawson Labs will not be liable for indirect or consequential damages caused by any defect in this product. Some states do not allow the limitation of consequential damages, so the above exclusion may not apply to you.