USING THE PIC17CXX ADAPTER

The PIC17CXX adapter (part#APIC17) allows the Andromeda Research EPROM+ programming system to support the complete 17CXX family of 16 bit microcontrollers from Microchip. These include the 17C42A, 17C43 and 17C44 devices. Support also extends to reading and programming the configuration fuses.

INSTALLING THE ADAPTER IN THE PROGRAMMING UNIT

To install the APIC17 adapter in the programming unit, raise the handle on the 32 pin ZIF socket to more than 45 degrees. This releases the tension of the socket mechanism. Orient the APIC17 adapter such that the dot on the 40 pin socket faces the same direction as the dot on the 32 pin programming unit socket. Insert the base pins of the APIC17 adapter into the 32 pin socket. Release the 32 pin socket handle to latch the APIC17 adapter in place.

NOTE: The APIC17 adapter is held in place by the frictional force of the 32 pin socket. When you insert and remove PIC17CXX devices from the adapter socket, you may find it necessary to hold the adapter in place by using your finger or thumb on the rear portion of the adapter.

CONFIGURATION FUSES

The PIC17CXX family of microcontrollers has seven configuration fuses. The fuses allow you to configure certain device options and other settings. The EPROM+ programming system allows you to both read and program the configuration fuses. To manually read, program or change the configuration fuses, use the Z - DEVICE OPTIONS command. Due to the internal architecture of the 17CXX family, there is a brief delay before a fuse operation is performed while the system accesses the fuse area of the part.

AUTOMATIC FUSE PROGRAMMING

You may configure the system to automatically program the configuration fuses after the memory array is programmed and verified. This is accomplished by placing the configuration fuse data into the system buffer one byte past the last location which corresponds to the device memory array. **EXAMPLE:** A 17C42A part has a memory array length of 7FF words or FFF bytes. The memory array (device size) is always displayed in the top, center window of the main command screen. To cause the system to automatically program the configuration fuses, the fuse data must be placed at byte address 1000 which is one byte past FFF. You may also place the data in the low byte of word address 800 (800L) which is the same as 1000. **IMPORTANT NOTE:** If you use the Z command to change the configuration fuse data, the resulting changes will automatically be placed into the buffer after the device memory array. After the device memory array has been programmed, the system will test the buffer location past the array for fuse data. If fuse data is there (the byte is not FF), the system will automatically execute a fuse programming operation. If you wish to create a file which will automatically program the fuses, use command A to save the buffer from 0 to one byte past the memory array. For the example above this would be from 0 to 800.

NOTE: Microchip specifies that fuse data be stored in a HEX file at word address FFE0. Due to inconsistencies in file support from different sources, the fuse data may or may not be located at that address. If you do not know what the fuse settings should be or where the fuse data is located, we suggest that you load your file using command 4. Examine the load summary result. The last location loaded from the HEX SUMMARY is most likely the fuses. Use the EDITOR “M” command to enter the fuse data into the proper buffer location for the device you are using.