

---

---

## The Mini-Assembler

Without an assembler, you have to write your machine-language program, take the hexadecimal values for the opcodes and operands, and store them in memory using the commands covered in the previous sections. That is exactly what you did when you ran the previous examples.

The Monitor includes an assembler called the *Mini-Assembler* that lets you enter machine-language programs directly from the keyboard of your Apple. ASCII characters can be entered in Mini-Assembler programs, exactly as you enter them in the Monitor. Note that the Mini-Assembler doesn't accept labels; you must use actual values and addresses.

---

### Starting the Mini-Assembler

To start the Mini-Assembler first invoke the Monitor by typing `CALL -151` and pressing Return, and then from the Monitor, type `!` followed by Return. The Monitor prompt character then changes from `*` to `!`.

When you finish using the Mini-Assembler, press Return from a blank line to return to the Monitor.

---

### Restrictions

The Mini-Assembler supports only the subset of 65C02 instructions that are found on the 6502.

#### Original Ile

Before you can use the Mini-Assembler on the original Apple Ile, you have to be running Integer BASIC. When you start up the computer using DOS or either BASIC, the Apple Ile loads the Integer BASIC Interpreter from the file named `INTBASIC` into the bank-switched RAM. Here's how to start the Mini-Assembler on an original Apple Ile:

1. Start Integer BASIC from DOS 3.3 by typing `INT` and pressing Return.
  2. After the Integer prompt character (`>`) and a cursor appear, enter the Monitor by typing `CALL -151` and pressing Return.
  3. Now start the Mini-Assembler by typing `F666G` and pressing Return.
-