movement, the opposite values (opposite to those in Tank or Surround) are added to the target. Thus, for a key 2 press, the target's Y coordinate is increased by one, and the phaser sights appear to raise. Actually, the target is moving down. Because the target itself is continually being supplied with new and random X,Y coordinates at the same time, both the starship's and the target's movements have the appearance of being separate and distinct. They are, in reality, one and the same. I am very pleased with the realistic effects of this feature, and hope you'll try it for your own "battle" programs. A joystick would be a great addition, such as the one described in an early VIPER.

Line 0354 calls the first subroutine of the main loop. We'll discuss all the subroutines in order after we've finished with the executive routine here.

Locations 0356-035C do a quick check to see if the "fire-button" (Key F) has been pressed. (Left-handers may want to change the fire button to some other key by setting a new value into V0 at location 0356.) If the "fire button" has been pressed, the Fire Phasers Subroutine is called, a tone is sounded, and the phaser "banks" are depleted by one phaser. Upon return from this subroutine, either V0 or V1 equalling 1 indicates a "hit", and to test this condition, VF is added to V0 (for simplicity, since now only one test will be needed), and the value of V0 is checked. If V0 is not zero, then there was a hit. The number-of-hits counter, VD, is increased by one, the target is erased, and the Enemy Craft Destruct subroutine is called.

As an interesting aside: To illustrate the programming tech - nique outlined above, try taking out the subroutine call at 0372, replacing it with a 1374 (GOTO 0374) instruction. The game will function exactly as before, except that now, instead of the target exploding, it will simply disappear when hit. The destruct sequence could have been written and inserted