

## Lab 6

### Scanning

In this lab, you will write a spimbot program that uses the scanner to locate the tokens. When you get the scan results, print out the token locations. Your program should be based on the program `bonk.s`, available in the course directory.

Starting a scan requires writing values to the following four memory addresses:

- `0xffff0050` gets the  $x$ -coordinate of the center of the scan.
- `0xffff0054` gets the  $y$ -coordinate of the center of the scan.
- `0xffff0058` gets the  $L_\infty$  radius of the scan. The scan will detect all tokens whose coordinates are both within the radius of the scan's center. For example, a scan with radius 50 centered at (100, 100) will find all tokens with both coordinates between 50 and 150.
- `0xffff005c` gets the address to which the results should be written.

The scan begins when the address is written to memory location `0xffff005c`.

When the scan is complete, interrupt 13 is received (provided it is enabled). This interrupt is acknowledged by writing to address `0xffff0064`. The scan results are written as an array to the fourth argument, the address given at the start of the scan. The results are an array of integers, each a single word. The first cell gives the number of tokens found. Later cells contain the token coordinates, each token presented as an  $x$ -coordinate followed by a  $y$ -coordinate. For example, if 2 tokens with coordinates (3, 4) and (5, 6) were found, the results would be the array containing 2, 3, 4, 5, 6.

Begin by writing a program that finds all of the tokens in a single scan of radius 150 centered at (150, 150). To make running this program less painful, you can use the program `fastscan-spimbot`, which is identical to `spimbot` except that scans are much quicker.

If you have additional time, modify this program so that it performs 4 smaller scans rather than a single large one. Note that only one scan can be running at a time so you cannot start one scan until the previous one completes.