EE675 : Assignment 2

Microprocessor Applications in Power Electronics

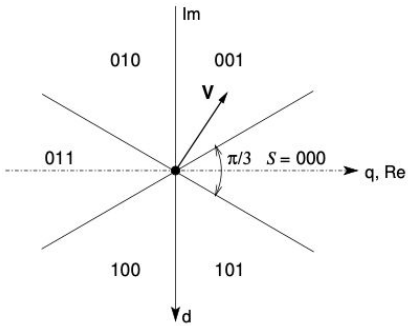
Jayesh Choudhary - 170070038

Question 1: Extracting the raw data entries from section in .obj/.out files (COFF format) using C language compiled with GNU C compiler gcc.

We first read the file which is given as an argument in the terminal (eg. - ***./rcof impy.out***)
If file is not found we display an error. Otherwise we read the file using fread and fseek which read the block of data from the stream. First 22 bytes represent the file header. Next we have optional file header (if any) which can be found out by 16-17 bytes of the file header. First 2 bytes represent the number of sections. We could use this piece of information for iteration. Then in each section header (which is of 48 bytes), first 8 bytes represent the name of the section, next 4 bytes (8-11) represent the physical address of the section, next 4 (12-15) represent the virtual address of the section. Next 4 bytes (16-19) represent the size of section. Next 4 bytes (20-23) points to the raw data entries of the section. Using this information we can create a loop within the first loop to go to the section raw data and store its value along with the address (which needs to be added to the physical section address).

Question 2: Algorithm:

- First, we check if v_d is positive or negative to realize if it lies below or above v_q axis.
- Then we check the value of v_q if it is positive or negative to see if it lies in the right or the left quadrant.
- Since in the microprocessor, it is not possible to get an accurate value of $1/\sqrt{3}$, we cannot use the slope of v_d / v_q to further distinguish between the sectors in the quadrants. So we use the square of the sine function [$v_d^2 / (v_q^2 + v_d^2)$] to find out the correct sector in which the point lies. The effective comparison is between v_d^2 and $(v_q^2 + v_d^2)/4$. Since the division is by 4 it could be done easily by right shifting it by 2 bits.



Let k = $v_d^2 / 4\,(v_q^2 + v_d^2)$

000	$v_q > 0$	$v_d > 0$	$k < 1/4$
101	$v_q > 0$	$v_d > 0$	$k > 1/4$
000	$v_q > 0$	$v_d < 0$	$k < 1/4$
001	$v_q > 0$	$v_d < 0$	$k > 1/4$
011	$v_q < 0$	$v_d < 0$	$k < 1/4$
010	$v_q < 0$	$v_d < 0$	$k > 1/4$
011	$v_q < 0$	$v_d > 0$	$k < 1/4$
100	$v_q < 0$	$v_d > 0$	$k > 1/4$

- The above table helps us to get the sector when the coordinate does not lie on the partition lines.
- For assigning the sector to a point lying on the line, we use the following table and appropriately use equality in the above table.

000	$v_q > 0$	$v_d > 0$	$k = 1/4$
111	$v_q < 0$	$v_d < 0$	$k = 1/4$
001	$v_q > 0$	$v_d < 0$	$k = 1/4$
100	$v_q < 0$	$v_d > 0$	$k = 1/4$
101	$v_q = 0$	$v_d > 0$	
010	$v_q = 0$	$v_d < 0$	

- For the case when v_q and v_d both are zero we assign it value 111.
- Since the question asks specifically to give output as unsigned integer, the final output stored in AH is one of these: 0x0000,0x0001,0x0002,0x0003,0x0004,0x0005,0x0007 for the sectors 000,001,010,011,100,101 and 111 respectively.