ME 5329 Project 3

Quantum Espresso Doping of Silicon Diamond Center Cubic (DCC) with Boron and Phosphorus

September 30, 2025

Dr. Aquino Fall 2025

Due: October 6, 2025 at 8:00am

Objectives:

- Understand DFT calculations with periodic boundary conditions using the Quantum Espresso program suite.
- Perform and analyze a k-points convergence test for a bulk structure.
- Recognize the significance of k-points selection in DFT calculation.
- Explore doping mechanisms and their impact on the electronic properties of a crystal.
- Differentiate between the n-type and p-type doping and their application in semiconductors.
- Learn to construct supercell models and comprehend their importance in specific Quantum Chemistry studies.

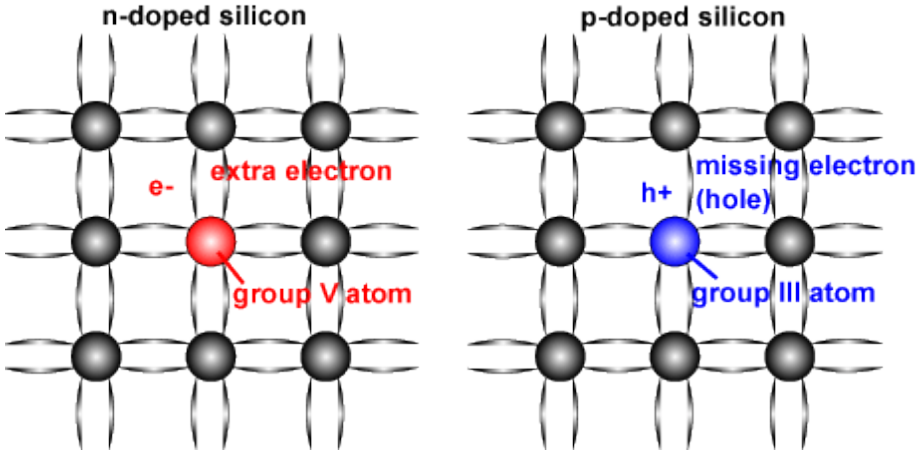

Figure 1: Illustration of the different types of doping that can be applied to silicon. Substitutional doping with elements like boron (B) introduces holes (p-type), while doping with phosphorus (P) adds extra electrons (n-type). These modifications alter the electronic properties of silicon and are essential for designing semiconductor devices such as transistors and diodes.

K Points Grid Mesh Test for Si DCC:

1. Visualize the Si Diamond Center Cubic crystal file (**Si_pristine.xsf** which is located in `/home/jerschro/Share/`).

2. Read the tutorial article on how to use Quantum Espresso linked below.

   - Quantum Espresso Tutorial

3. The following steps can be performed either manually or through an automation script in the programming language of your choice. Our goal is to evaluate the effect of different k-point meshes (listed in the table below) on the electronic structure of the Si diamond center cubic bulk system, provided in the file **Si_pristine.xsf**.

K-point Grid Meshes to Test:
$1 \times 1 \times 1$
$2 \times 2 \times 2$
$3 \times 3 \times 3$
$4 \times 4 \times 4$
$5 \times 5 \times 5$
$6 \times 6 \times 6$
$7 \times 7 \times 7$
$8 \times 8 \times 8$

4. To perform a single k-point test calculation, first create a new empty working directory. Within this directory, create the Quantum Espresso input file, **pw.in**. Set the calculation routine type to *'vc-relax'*, include the correct k-point mesh for the test, and use the **Si_pristine.xsf** geometry. Submit the calculation using the SLURM file **run_qe.sh** (which is located in `/home/jerschro/Share/`).

   - If you downloaded the **pw.in** from the Quantum Espresso tutorial, you will need to add the atomic coordinates to the **"ATOMIC_POSITIONS"** section, the lattice matrix to the **"CELL_PARAMETERS"** section and the atom species, molecular weight, and atom pseudopotential file to the **"ATOMIC_SPECIES"** section. The default k-points in the **pw.in** template is a Γ-Point ($1 \times 1 \times 1$ k-point grid). Additionally, make sure to ensure the *"nat"* (number of atoms) and *"ntyp"* (number of atom type) parameters are correct based on the initial structure given in the input file.
   - The text box below shows how the *"K_POINTS"* section of the **pw.in** file should be written for a $1 \times 2 \times 3$ k-point grid mesh. For the k-points test, this is the only section of the input file that needs to be modified between calculations.

```
K_POINTS automatic
1 2 3 0 0 0
```

5. After each k-point calculation is completed, you may gather the total SCF energy, the number of k-points, and the CPU time of the calculation.

   - The self-consistent field (SCF) energy at each iteration is recorded in the **pw.out** file on lines containing "! total energy". To extract the final SCF energy, you can use the following command: `'grep !  pw.out | tail -1'`
   - You can find the total number of k-points used by Quantum Espresso in the line that contains "number of k points" in the **pw.out** file.
   - You can find the total CPU time by looking at the bottom of the **pw.out** file under the "Parallel routines" section. The time listed there is the total CPU time.

Supercell Generation and Doping:

1. Collect the final structure from the optimal k-points identified in the k-points test and save it in a file that you can visualize.

2. The next step may be achieved using a visualization program (such as VESTA) or `molecule_lib`. Take the final structure and generate a $2 \times 2 \times 2$ supercell. Below is the python code for `molecule_lib` which uses the ABCMolecule.generate_supercell() method if you choose to use it.

```
import molecule_lib as mlb

pristine = mlb.read_vasp("Si_pristine_OPT.vasp")
supercell = pristine.generate_supercell(x=2, y=2, z=2)
supercell.to_vasp("Si_supercell.vasp")
```

3. Now we have the $2 \times 2 \times 2$ Si DCC supercell. To dope the Si supercell with boron and phosphorus, all we need to do is change **one** Si atom to the doped atom of choice.

Supercell Quantum Espresso Calculations:

1. The calculations described below will need to be done for the three supercell structures we have created, which are Si, Si+P and Si+B.

2. Create a new directory for the supercell structure you are working on. Within this directory, create a sub-directory named "relax". Generate the Quantum Espresso input file, **pw.in**, with the *'relax'* routine invoked. Add the coordinates of the supercell in the **pw.in** file. Make sure to ensure the *"nat"* and *"ntype"* parameters are correct and to modify the "ATOMIC_SPECIES" section as needed. For each supercell, you will need to add the *'occupations'*, *'smearing'* and *'degauss'* in the "SYSTEM" section of **pw.in**. This is because the doped structures either lack an electron or have an extra electron, and we want Quantum Espresso to calculate the Fermi Energy. The following code box shows how the edited "SYSTEM" section should look like in the **pw.in**.

```
&SYSTEM
    ibrav = 0,
    nat = 64,
    ntyp = 2,
    ecutwfc = 68,
    vdw_corr = 'dft-d3',
    occupations = 'smearing',
    smearing = 'mp',
    degauss = 0.01
/
```

3. Once the "relax" calculation has finished, create a directory named "scf". Copy the **pw.in** file and **run_qe.sh** from the "relax" directory into the "scf" directory. Retrieve the final atomic positions from the "relax" calculation and paste them into the "scf" **pw.in**. Change the invoked Quantum Espresso routine to *'scf'*. Make sure the changes we made in the above step to the parameters in the "SYSTEM" section of the **pw.in** are still in effect.

Deliverables for the Report:

1. Create a plot of energy versus the k-point grid mesh for the report. The k-point grid mesh axis label may be text (e.g. $1 \times 2 \times 3$) or the number of k-points found. Include a table with the values used for this plot. What do you notice?

2. Create a plot of calculation time versus the k-point grid mesh. The k-point grid mesh axis label may be text (e.g. $1 \times 2 \times 3$) or the number of k-points found. What observations can you make?

3. Based on the analysis of the results of the k-points grid mesh test (Deliverable Step 1 and 2), identify the favorable k-points grid mesh and justify your choice.

4. Visualize the favorable k-points pristine unit cell using your preferred visualization software. Include a picture of the structure. Report the number of atoms in the molecular file and estimate how many atoms you see in the visualization. If there is a difference, explain why.

5. Present the lattice matrix of unit cell for both Si bulk and Si supercell in a table.

6. Report the unit cell parameters (a,b,c in Å, α,β,γ in degrees and volume in $Å^3$) for Si bulk and Si supercell in a table. You can use VESTA or ABCMolecule.unitcell.getabc() and ABCMolecule.unitcell.getanglesdeg().

7. Visualize the final structures from each "relax" calculation for the three supercell structures. Discuss any differences observed in the lattice around the doped atoms? Have the bond lengths changed near these atoms?

8. Report the calculation total energy (in Rydbergs, Ry) and Fermi energy (in eV) from "scf" calculations for each supercell structure in a table.

9. Create a table compare the total energy and Fermi energy of each supercell normalized to the pristine supercell (Si) with units in electronvolts (eV). What effects does doping have on properties? What does this calculation inform us about semiconductors?

10. Discuss why the creation of a supercell was necessary in this Project. If doping calculations had been performed on the pristine unit cell, how would these results relate to real world doped structures? Hint: Calculate the doping concentration and compare it with is experimental values available in the literature.

Submission Requirements:

Please include the scripts you used to generate the structures or information on where the structures were found, paths to where the calculations were ran on the HPC (a parent directory for this specific project works also) and your post processing scripts that generated the graphs and results in the Canvas Assignment submission. Including this information demonstrates the student's work and understanding of the project. Please consider this carefully.

The only accepted submission format for the report is a Microsoft Word document.

Please submit it under the **'Project 3'** assignment in Canvas.