Quantum Computing for Quantum Chemistry
Module 2

Anthony Slawski

January 2026

This lab extended the second quantisation formalism to the electronic Hamiltonian and introduced the Jordan-Wigner transformation, which maps fermionic creation and annihilation operators to Pauli operators on qubits. A comparison between simply applying X operators and the full Jordan-Wigner mapping showed that while the simple method can reproduce the correct bitstring for some state preparations, it does not implement fermionic creation and annihilation operators.

The Jordan-Wigner mapping was implemented with the parity Z string and the $(X \mp iY)/2$ structure, applied term by term to each fermionic string. Even though each elementary fermionic operator maps to only two Pauli terms, multiplying several of them together produces large strings.

The main computational consideration for PauliString and QubitOperator was to keep them as simple and general as possible. OpenFermion's QubitOperator was used for the bookkeeping, combining like terms and keeping a consistent ordering. A thin wrapper was then added to provide addition and multiplication.

Correctness was checked using the same workflow the code is meant for. The elementary Pauli multiplication rules were verified directly by matrix multiplication of X, Y, and Z. Then, the PauliString and QubitOperator classes were tested on the specific addition and multiplication examples provided, confirming that identical Pauli strings combined correctly by coefficient addition and that multiplication produced the expected phase factors and Pauli products. The second quantised H_2 Hamiltonian constructed from the Unrestricted Hartree-Fock Molecular Orbital transformation was compared against an OpenFermion reference Hamiltonian generated from PySCF output, and the term list agreed (up to numerical precision). Finally, the Jordan-Wigner Hamiltonian produced by the custom mapping was compared against OpenFermion's built in transform, and the resulting Pauli terms and coefficients matched.