



Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

# Single-electron Two-qubit System in Bilayer Graphene

Group Project - Quantech Workshop

Dario Denora  
Alexander Jürgens  
Nicola Meggiato  
Emre Yazici

Integrated Systems Laboratory  
Department of Information Technology and Electrical Engineering, D-ITET  
ETH Zürich

**Supervisors:**  
Dr. Youseung Lee  
Dr. Jiang Cao  
Guido Gandus

February 3, 2023



## **Abstract**

Within the framework of the Quantech workshop project we propose an expanded model of the architecture recently developed by Computational Nanoelectronics research group in [1]. We include the spin degree of freedom in the charge qubit architecture based on bilayer graphene. Furthermore, we propose a mechanism for controlling this spin qubit as well as coupling the two degrees of freedom, allowing universal unitary manipulation of the four dimensional space. The goal of this project is the detailed formulation of the architecture and the numerical simulation of the said system. For the spin manipulation, we believe the best approach to avoid the presence of an AC magnetic field, that in principle may influence the charge qubit functionalities, is by electric dipole spin resonance controlled by spin-orbit coupling. This is necessary to perform arbitrary qubit rotations employing a time dependent electric field applied through the gates.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Device Building Blocks</b>	<b>2</b>
2.1	Bilayer Graphene . . . . .	2
2.2	Quantum Dots . . . . .	3
2.3	Novel Charge Qubit . . . . .	4
2.4	Spin Qubit . . . . .	5
2.5	Proximity effects . . . . .	7
2.6	Proposed device . . . . .	8
2.7	Project structure . . . . .	9
<b>3</b>	<b>Ab-initio Simulation Framework</b>	<b>11</b>
3.1	DFT and Wannierization . . . . .	12
3.1.1	Crystal Structure . . . . .	12
3.1.2	DFT . . . . .	12
3.1.3	Wannierization . . . . .	13
3.2	Device Hamiltonian . . . . .	14
3.2.1	Peierls Substitution . . . . .	14
3.2.2	NEGF . . . . .	15
<b>4</b>	<b>QuTiP Simulation &amp; Control Scheme</b>	<b>17</b>
4.1	Framework . . . . .	17
4.2	Landau-Zener-Stückelberg Interference . . . . .	18
4.2.1	Charge Qubit Control . . . . .	18
4.3	Electron Dipole Spin Resonance . . . . .	20
4.3.1	Spin Qubit Control . . . . .	20
4.3.2	Two Qubit Gates . . . . .	21
4.4	Use-case of the device: spin to charge conversion . . . . .	23
<b>5</b>	<b>Conclusion</b>	<b>25</b>
<b>A</b>	<b>Supplementary Figures</b>	<b>26</b>
<b>B</b>	<b>Code</b>	<b>27</b>
	<b>References</b>	<b>42</b>

## CHAPTER 1

# Introduction

---

Nanoelectronic technology is widely regarded as critical to the development of quantum computers, which will be capable of solving problems that classical machines cannot. Experiments have already demonstrated a wide range of qubit platforms, such as charge, spin, superconducting, and trapped-ion qubits [2, 3]. All of these approaches require a large number of physical qubits to realize a single logical qubit. As a result, it becomes crucial to develop a device with optimal characteristics that often corresponds to the fulfillment of the DiVincenzo criteria [4]. One possible solution is obtained by creating charge and spin qubits in bilayer graphene (BLG). Because carbon atoms are very light elements, graphene has a low spin-orbit interaction and is expected to have weak hyperfine coupling effects, which is why it was chosen over other 2D materials for the development of solid state qubits. As a result, graphene-based qubits are expected to have a long spin coherence time, which is a crucial characteristic for fault-tolerant quantum computing. The project enter in this framework where as a first step we wanted to develop an ab-initio simulation framework capable of simulating graphene spin qubit. Then, starting from the readout mechanism described in [1] the coherent manipulation of the charge and spin part separately is illustrated. The strong spin anisotropy between conduction and valence band and the valley character of the electronic states in 2D materials provide unique control knobs to manipulate spin precession. Here, by manipulating the anisotropic spin-orbit coupling in bilayer graphene by the proximity effect to  $WSe_2$ , we propose a way to coherent control spin precession in the absence of an external magnetic field, and at the same time also able to control the charge. In the following, we first present some theoretical simulation background needed in order to obtain the relevant results.

## CHAPTER 2

# Device Building Blocks

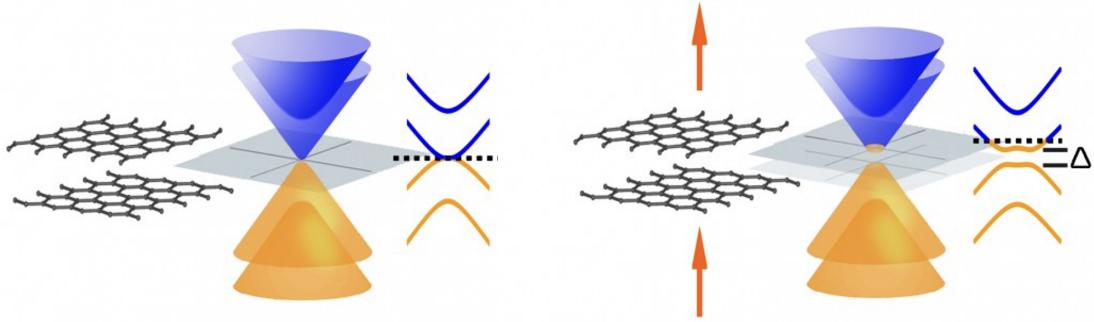
---

This chapter describes the main ingredients in order to fully justify the chosen device design. Starting from the 2D material exploited for the fabrication of the DQD charge qubit, a schematic of the building block of the device is then illustrated. The device's design is highlighted in order to emphasize the unique characteristics and benefits of using a single QD formed in a Van der Waals heterostructures of BLG and  $WSe_2$

## 2.1 Bilayer Graphene

Bilayer graphene (BLG) is a 2D material of particular interest in current research due to its remarkable bandgap tunability via an external displacement field perpendicular to the graphene plane [5]. This is related to symmetry-breaking phenomena between the two layers induced by an electric field, which is typically obtained by placing metallic gates on the top and bottom of the structure containing the BLG (Fig. 2.1). To implement qubits in a solid-state environment, precise system control at the engineering level is required. As a result, the electric field influences both the conduction and valence bands of BLG by altering the quantum superposition of wave functions localized in different layers [6–8]. Overall, in the absence of a magnetic field, charge carriers in BLG exhibit spin and valley degeneracy, yielding an extra degree of freedom that can be used to implement a qubit, as proposed in [9–11].

In terms of its production, mechanical exfoliation produces both monolayer and bilayer graphene with the best electrical and optical properties when compared to other methods [12]. Important to underline is that bilayer graphene has a different band structure than its monolayer counterpart [5, 6]. In fact, the AB-stacked configuration exhibits a zero electronic bandgap  $\Delta$ , which in turn can be tuned by applying a perpendicular electric field [13]. As a consequence, by tuning the gate voltage so that the Fermi level is in the bandgap, charge flow under the gates is hindered. By taking advantage of this property, gate structures known as split gates (SG) on top of the stack and back gates (BG) represented by the graphite, which is the bottom material of the stack, can be used to create a narrow channel of conducting carriers. Early experiments with this configuration showed conductance quantization of the channel [14]. When the Fermi energy under the SGs is tuned into the gap, the device exhibits a high resistivity of tens of  $M\Omega$ , which is the key ingredient for electrostatic confinement of charge carriers in our system. The nature of the charge carriers in reservoir regions with no top gates is determined by the sign of the voltage applied to the BG. Furthermore, an intriguing feature of electrostatic gating is that the top gate is not strictly necessary to open a band gap because the back gate suffices.



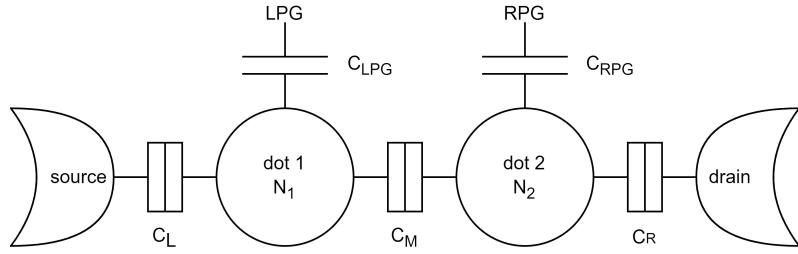
**Figure 2.1:** Bilayer graphene band gap tunability effect.

The electrostatic potential that a charge carrier experiences as it passes through the constriction must be shaped by additional gates known as finger gates or plunger gates. The ability to precisely control the confinement potential for electrons or holes, which results in a fine control of its coherence, makes this a crucial component of the creation of quantum dots in such a structure.

According to previous studies, the bilayer graphene is typically encapsulated between hexagonal boron nitride (h-BN) flakes because it is thought to increase mobility, reduce carrier inhomogeneity, and reduce intrinsic doping. [15, 16]. The reason can be attributed to the Bernal structure of hBN where boron and nitrogen atoms occupy the A and B sublattices with lattice constant almost identical to the one of the BLG. In addition, a graphite layer is employed as the back gate since it is believed to provide resistances below depletion gates on the order of  $10\text{ G}\Omega$  which is much higher compared to other materials, giving rise to the optimal confined area for Quantum Dot's purposes [17].

## 2.2 Quantum Dots

Opening a narrow channel of quantized conducting modes using split gates is not enough to define a QD. Additional gates are required to strongly confine the charge carriers in all spatial directions, leading to a discrete energy spectrum. Following the standard treatment of QD physics [18], it is possible to imagine the behavior of a QD as in Fig. 2.2. The QD is coupled to source (S) and drain (D) reservoirs through tunnel barriers, and a voltage applied to source-drain leads to the opening of a bias window  $\mu_S - \mu_D = eV_{SD}$ . By tuning the discrete energy levels of the quantum dot using a gate electrostatically coupled to the QD, it is possible to shift the levels into the bias windows and have current flowing through the dot. The amount of energy shift is defined as  $e\alpha_{PG}V_{PG}$  where  $\alpha_{PG}$  is called the lever arm of the gate. Such a quantity can be experimentally determined from the measurements as the slope of the Coulomb diamonds. We reach the Coulomb blockade regime when no energy level falls within the bias windows. Furthermore, the energy required to add an electron can be calculated as the energy difference between



**Figure 2.2:** A double QD model is depicted schematically. The two dots are coupled capacitively and via tunneling, and each dot is capacitively coupled to a plunger gate.

two resonances  $E_{add}(n)$ , which is equal to the sum of the single-particle level spacing plus the charging energy. Through tuning the voltage applied, electrons can be controlled to tunnel in and tunnel out into the dot one by one, due to Coulomb repulsion. Thus gate-controlled quantum dot offers an electrically tunable platform for studying electrons in solids down to single-particle level. This is the typical framework where it is possible to define a charge qubit. In fact, by electrical control of the energy level allow coherent charge oscillation [19].

### 2.3 Novel Charge Qubit

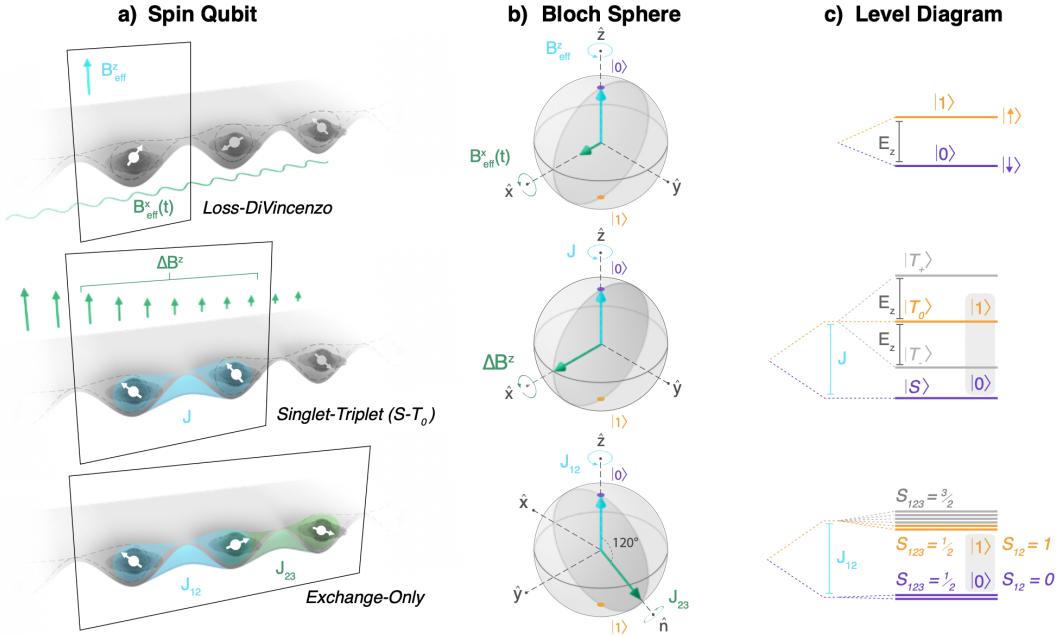
Traditional double quantum dots (DQD) charge qubits are fabricated using semiconductor heterostructures, such as Si/Ge or GaAs/GaAlAs, and they encode the two level system in the localization of the electron in either one quantum dot (QD) or the other.

An attractive alternative to the traditional approach is using van der Waals charge qubits (vdW CQs). These are based on the electronic properties of 2D materials, such as transition metal dichalcogenides (TMDCs), or bilayer graphene, that are stacked together in a vdW heterostructure. The qubits are formed in a single quantum dot, that is defined in the 2D material by electrostatic gating. The two level system, in this case, is defined by the localization of the electron wave function in the upper or the lower layer.

vdW CQs have a number of advantages over traditional DQD charge qubits:

- vdW CQs can be fabricated using dry transfer techniques, which allow for the precise alignment of different 2D materials and a high degree of control over the qubit properties
- vdW CQs have long coherence times, which could be beneficial for the implementation of quantum gates and algorithms
- vdW CQs can be integrated with existing semiconductor technology, which could facilitate the integration of quantum computing with classical computing.

The design of the charge qubit in this project is the one developed in the previous work done by the host group [1].



**Figure 2.3:** a) Spin configurations, b) Bloch spheres, and c) energy level diagrams associated with Loss-DiVincenzo (LD) single spin qubits, two-spin singlet-triplet (ST0) qubits, and three-spin exchange-only (EO) spin qubits. For the LD qubit, a static magnetic field  $B_z$  the quantization axis of the single spin, while a transverse (and smaller) ac magnetic field  $B_x(t)$  drives coherent spin rotations between spin-up and spin-down. For the ST0 qubit, exchange coupling  $J$  and a longitudinal magnetic field gradient  $\Delta B_z$  provide two orthogonal control axes. Figure taken from [20].

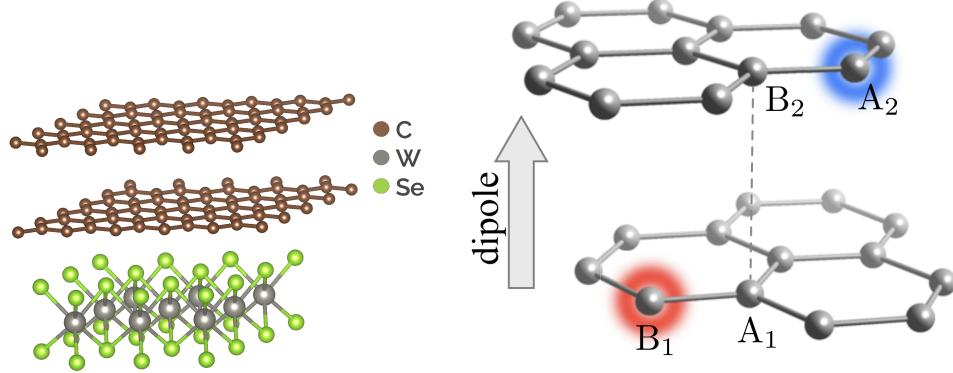
## 2.4 Spin Qubit

An electron's spin degree of freedom can serve as an excellent qubit because it provides a natural two-level system that is insensitive to electric fields, resulting in long quantum coherence times. One of the key challenges for quantum computation is finding a good balance between maintaining enough isolation to preserve the quantum state while maintaining enough interaction to permit fast qubit control, coupling, and measurements. Charge qubits are intrinsically affected by decoherence due to their interaction with the electrical environment (charge noise). That is why, over the last decade, the emphasis has shifted to spin qubits, which represent one of the most fundamental two-level systems. The Zeeman effect splits the energy of spin up and down states that are aligned with or perpendicular to the field direction when an external magnetic field is applied. These states form the computational basis of the Loss-DiVincenzo (LD) qubit [21]. Spin resonance allows for the manipulation of the electron spin state, and two-qubit gates are also possible in the presence of a weak coupling between two spins [21]. The first studies of spin qubits concentrated on the group III-V semiconductor GaAs, of which all isotopes have a nonzero nuclear spin. When the nuclei in the host crystal have a spin, they create a large (noisy) environment for the spin qubit, resulting in a high degree of decoherence. Nevertheless, thanks to the low level of disorder and high car-

rier mobility single [22, 23], double [24, 25] and array [26, 27] of spins in quantum dots were demonstrated. Electron spins initialization and readout with high fidelity was also demonstrated using energy dependent tunneling or the Pauli exclusion principle. Conversely, both silicon and germanium, group IV semiconductors, have a large abundance of net-zero nuclear spin isotopes and can be further isotopically purified to contain diminishing amounts of nuclear spins [28] leading to a remarkable improvement in the qubit coherence time [29–31] and a two-qubit gates fidelity up to 98 % [32, 33]. Also spin-qubits operation at temperature around 1 K has been studied as reported in [34, 35] and interaction mediated by superconducting resonator via a microwave photon has been shown [36, 37]. Scalability is just another reason to take advantage of the spin degree of freedom. The small intrinsic quantum dot (QD) scale of 100 nm offers itself to the development of a dense quantum computing architecture that could be mass-produced by the semiconductor microelectronics industry, given that a fully-error corrected quantum computer is likely to require at least one million physical qubits. At the same time, the small physical scale of a spin qubit might present engineering hurdles in addressing each qubit and ensuring adequate connectivity for quantum error correction.

Hole spin qubits in Ge/SiGe heterostructures are emerging as an interesting platform thanks to their intrinsic ability of controlling the qubit all electrically. In fact, the electric dipole spin resonance (EDSR) combining strong spin-orbit coupling (SOC) and electric-dipole transitions facilitates fast spin control in a scalable way, which is the critical aspect of the rapid progress made recently in germanium (Ge) hole-spin qubits.

There are three major types of electron spin qubits as shown in Fig. 2.3: LD single-spin qubits, singlet-triplet spin qubits, and exchange-only spin qubits. In our case the focus has been the LD spin qubit since we wanted to define a single electron in a single quantum dot as the element able to define two qubits by means of its charge localization and spin. In order to define the two level system for the spin a constant magnetic field is required. For the implementation of single-qubit gates, time-dependent control of magnetic field or effective g-factor is necessary; this has been done using a combination of static and oscillatory magnetic fields within the framework of electron spin resonance (ESR), or using oscillatory electric fields in combination of spin-orbit coupling ([38, 39]) or magnetic field gradients by applying electric dipole spin resonance (EDSR). Mentioning this is important because EDSR will be the mechanism we are going to exploit in order to drive the spin qubit. The first realization is documented in [39] where it was used the intrinsic SOC of GaAs to accomplish electrically driven single spin rotations. An alternating current voltage excitation applied to a gate electrode changed the orbital wave function, allowing coherent Rabi oscillations. Since the ac driving field can be delivered directly to a QD gate electrode, the switch from EDSR to ESR not only offers increased speed but also a clear mechanism for selectivity. In fact, magnetic fields are hard to confine in a small region which can cause a detrimental crosstalk when driving one qubit which is placed in the vicinity of another one. The specific model Hamiltonian for our implementation will be described in the following section. Before doing that a description of the way spin orbit coupling can be induced in bilayer graphene is presented.



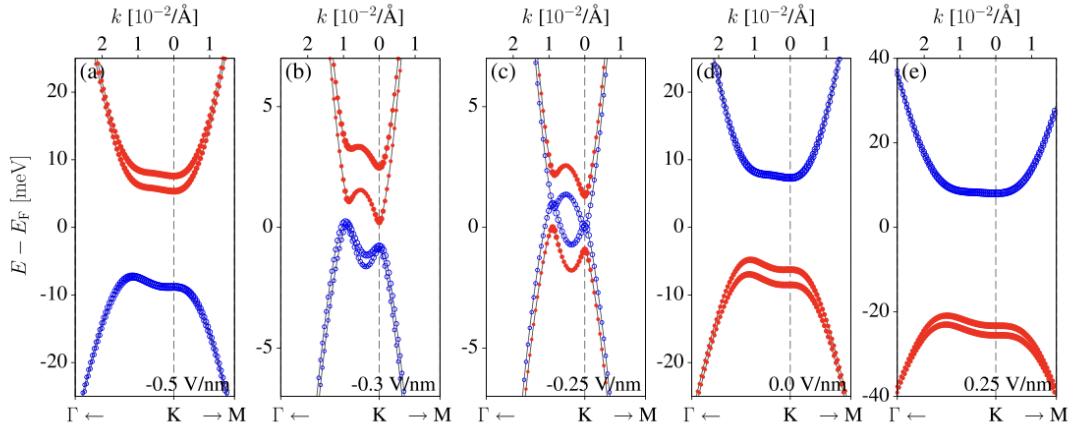
**Figure 2.4:** (a) Atomic structure of bilayer graphene on monolayer WSe<sub>2</sub> (supercell).(b) Sketch of bilayer graphene with atom labels forming a bare unit cell in Bernal stacking. Orbitals on nondimer atoms B<sub>1</sub> and A<sub>2</sub> form the low energy valence and conduction bands in the electronic structure of bilayer graphene, with B<sub>1</sub> being closer to WSe<sub>2</sub>. Figure taken from [40].

## 2.5 Proximity effects

Scaling down heterostructures, improving interface quality, and incorporating atomically thin two-dimensional materials offer a new and systematic approach to material design. Materials can be influenced by proximity effects, which enable them to inherit properties from neighboring materials, such as superconductivity, magnetism, topological nontriviality, or improved spin-orbit coupling. These proximity effects complement conventional methods of creating materials through doping or functionalization and also surpass their limitations. Properties that are not found in any constituent region of the heterostructure can be realized in proximitized materials. The proposed concepts provide a solid foundation for exploiting other proximity effects to tailor materials and uncover unexpected phenomena.

Theoretical studies of proximity effects in graphene on the entire family of TMDCs as prospective graphene substrates have recently been conducted [41]. A 1 meV increase in proximity spin-orbit coupling was estimated, which is enormous when compared to pristine graphene, where spin-orbit coupling is roughly 10  $\mu$ eV [42]. In [40] was found by performing first principle calculation BLG-WSe<sub>2</sub> heterostructure, whose model can be seen in Fig. 2.4, a displacement field emerges intrinsically, allowing for a highly efficient electric control of proximity effects. Overall, it was possible to define the following properties for the vdW heterostructure:

- Typical experimental electric fields of the order of 1 V/nm can increase, decrease, or reverse the intrinsic band gap, which is around 10 meV. The transverse field directs from WSe<sub>2</sub> to the BLG. That is the reason why B<sub>1</sub> electrons ( see Fig. 2.4) have lower energy, and form the valence band, while A<sub>2</sub> electrons have higher energy and form the conduction band.
- With a spin-orbit coupling of around 2 meV, the valence band's spin-orbit coupling is two orders of magnitude larger than that of the conduction band. This large discrepancy results from the fact that the conduction band is generated by orbitals



**Figure 2.5:** Calculated sublattice resolved band structures around the K valley for different transverse electric field. The circle radii correspond to the probability of the state being localized on carbon atoms B1 (red) and atoms A2 (blue) [40].

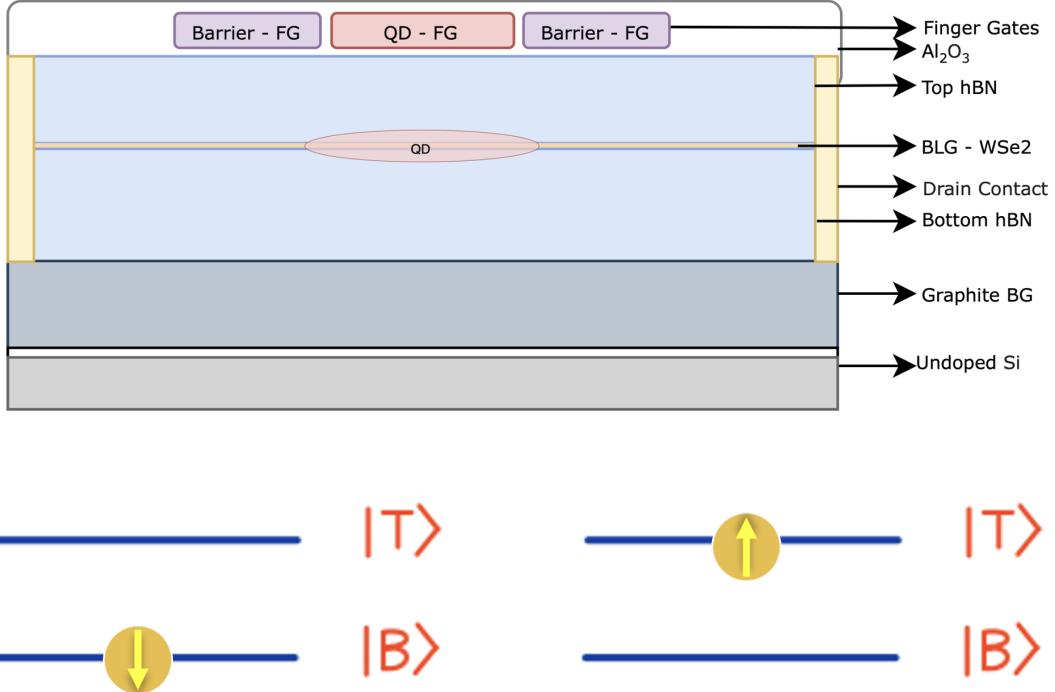
in the top layer, where proximity effects are inherently weak, while the valence band is formed by orbitals of carbon atoms close to WSe<sub>2</sub> in the bottom layer.

- By opposing the built-in field, a transverse electric field can efficiently switch on (and off) spin-orbit coupling and spin valley locking of electrons. This phenomenon is known as the spin-orbit valve [40].

Consequently, an applied transverse electric field can efficiently control proximity induced spin-orbit coupling and intrinsic electric polarization in a BLG-WSe<sub>2</sub> heterostructure, as shown in the plot of low energy band structures of BLG in the presence of applied electric fields in Fig. 2.5. This effect will be used to control the spin orbit interaction in the conduction and valence band (which represents also the computational basis state of our qubit) and thus implementing EDSR to drive the spin.

## 2.6 Proposed device

The proposed device aims to combine the previously mentioned charge and spin qubits in a single quantum dot. This should be possible because the two degrees of freedom are fully electrically driven. In fact, the charge qubit is controlled by a DC electric field that allows the charge localization in the two layers to be changed, whereas an oscillating electric field is required to perform EDSR in the case of a non-zero spin-orbit coupling. A model for such a device can be seen in Fig. 2.6. The quantum dot is expected to form underneath the central finger gate, whereas the other two gates are required in order to tune experimentally the tunnel coupling to the source and drain reservoirs. The stack of BLG on top of a WSe<sub>2</sub> monolayer is encapsulated between thicker hBN layers for the reasons explained in Sec. 2.1. Such a structure will be the starting point also for the finite element electrostatic simulation. In Fig. 2.6 it is also possible to find a potential novel feature of this type of qubit i.e. in situ spin to charge conversion. In fact, traditional spin



**Figure 2.6:** (a) Model of the single quantum dot structure realizing the proposed device.(b) Example of potential application: spin to charge conversion. We envision it may be possible

readout uses the Elzerman readout technique or the spin blockade mechanism. In order to readout the state of a spin qubit, both of these technologies require the presence of an ancilla qubit. We propose that by performing a SWAP operation between the states of the charge qubit and the spin qubit, we can perform a fast electrical readout of the charge qubit's localization, which is dependent on the state of the spin qubit. The illustrative example shows how when the spin qubit is up, the charge moves to the top layer, whereas when the spin is down, the charge remains in the bottom layer.

## 2.7 Project structure

The project was divided into two parallel pipelines. The first pipeline focused on the ab-initio simulation of material properties, starting with the DFT atomistic simulation of bilayer graphene and its dependence on interactions with other materials. This was achieved using the software QUANTUM ESPRESSO. The resulting set of electronic state wavefunctions represented by plane waves was then transformed into maximally localized Wannier functions using the Wannier90 tool. This step was necessary to derive a tight-binding-like Hamiltonian that describes the collective behavior of the atoms in the material with a smaller basis than the one output from DFT, allowing for the study of electric and magnetic field dependence. The magnetic field dependence was treated through the Peierls substitution in the tight-binding Hamiltonian, and the corresponding

## CHAPTER 2: DEVICE BUILDING BLOCKS

eigenenergy problem was resolved. Meanwhile, the electric field dependence was implemented through a finite element analysis of the potential obtained by a prototype 3D device, which was calculated using the NEGF method. This potential was then inserted into the Hamiltonian and solved for nanoribbon and quantum dot geometries. The final aim was to extract the main device parameters, such as the g factor, spin-orbit coupling, and tunneling coefficient, by performing a parametrical resolution of the Hamiltonian.

The second pipeline involved describing the high-level description of the device from an operational perspective. Using the python package QuTiP, we modeled the behavior of a charge qubit and a spin qubit encoded in the same quantum dot within the device of interest. We modeled the qubit driving by applying all-electric control pulses, using known control schemes such as LZSI and EDSR. We characterized the timescales for both qubits and evaluated the results in terms of their applicability for a novel readout mechanism for spin qubits.

# Ab-initio Simulation Framework

---

In order to test our model with realistic parameters, we need to know certain material properties. In some cases, these can be extracted from experimental data or theoretical calculations already found in the literature [5, 10, 15, 43, 44]. However, such results are not always available, especially when novel materials, complicated hetero-structures or device geometries are involved. In particular, when external electric and magnetic fields are considered the response of the material may become too complicated to capture with simple models. This is why we have worked to develop a *from first principles* simulation framework in order to model the proposed device and extract useful parameters.

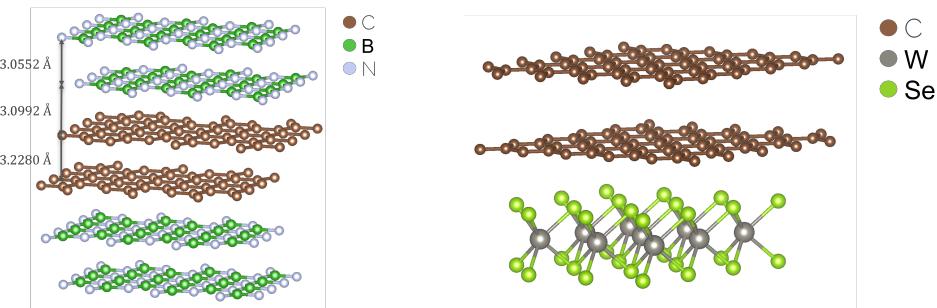
Our approach consists of first performing a DFT calculation using Quantum ESPRESSO [45] or VASP [46] to obtain fundamental material properties. Once we conclude the DFT part, we then derive a tight-binding Hamiltonian by Wannierizing [47, 48] this result. We can then construct a Hamiltonian for our device. The effect of magnetic fields are modelled using the Peierls substitution method [49]. Gate-defined electric fields are treated in parallel to this by first considering the bulk Hamiltonian of our material and using the NEGF formalism[50] to obtain the effective field acting on the device. This field is then incorporated to the device Hamiltonian before diagonalizing the system to find its eigenenergies.

### 3.1 DFT and Wannierization

We make use of the Density Functional Theory[51] and the available open-source frameworks[45, 46] to compute the fundamental properties of the material system we are interested in. Once that is done, we can fit a Hamiltonian in a Maximally Localized Wannier Function basis on the result[47].

#### 3.1.1 Crystal Structure

Any material computation starts with first defining the crystal structure of interest, along with its symmetries. In the scope of our project, these have been hetero-structures of mono- or bi-layer 2D materials.



**Figure 3.1:** (a) Crystal structure of the hBN-BLG tristack used in subsequent material calculations. (b) Crystal structure of the BLG-WSe<sub>2</sub> system proposed for tunable SOC strength device.

#### 3.1.2 DFT

Once we have decided on a structure, all that's left is to find the relevant pseudo-potentials as well as determining computational parameters such as tolerances and structural parameters like electric field.

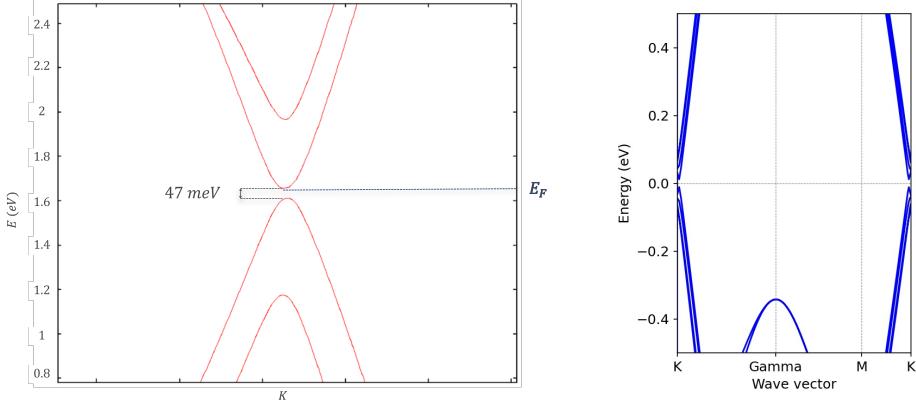
It should be noted at this stage that including SOC effects accurately requires a fully relativistic calculation with the correct psuedo-potentials as well as a careful tuning of the relevant parameters. Attaining convergence is more tricky for these calculations and may require many trials.

In general, DFT calculations are made using an iterative scheme until convergence and reasonable accuracy is reached at each stage:

1. Find optimal interlayer distance  $\delta$  by minimizing total energy in SCF calculations for different  $\delta$ .
2. Run SCF, check convergence
3. Run relax, sanity-check the result by visualizing the resulting structure
4. Run band calculation, check the resulting bands and compare with literature
5. Run NSCF, check convergence

As a result, we will have a set of wavefunctions that corresponds to the possible electronic states residing in our crystal and define the bandstructure. We can get various properties of our system by doing different operations on these bands/wavefunctions.

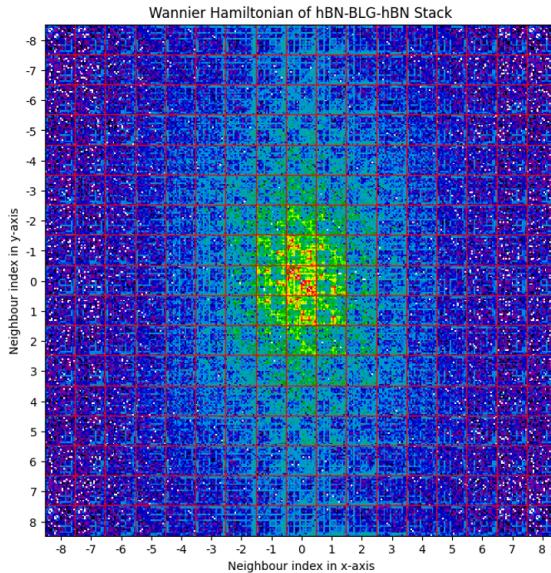
As



**Figure 3.2:** (a) Fully relativistic DFT band structure of the hBN-BLG tristack. Electric field across the layers: 0.5V/nm (b) Fully relativistic DFT band structure of the BLG-WSe<sub>2</sub> system.

### 3.1.3 Wannierization

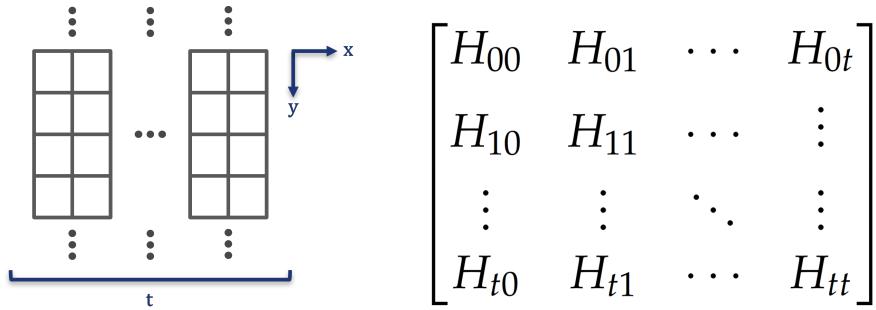
We use the Wannier90 package[47] to fit a Wannier Hamiltonian on our DFT results. Parameters we tune are mainly the placement of starting orbitals and the energy windows. For manipulating the resulting Hamiltonian, open source *pyWannier* package can be used.



**Figure 3.3:** Visualization of the Wannier Hamiltonian for the hBN-BLG stack. Each cell shows the hopping parameter from the cell at coordinate  $(x,y)$  to the cell at origin.

## 3.2 Device Hamiltonian

Once we have a tight-bonding Hamiltonian for the unit cell, it's relatively straightforward to put together a Hamiltonian that describes the behaviour of a collection of unit cells. Each unit cell consists of a set of orbital states and the hopping constants between them. Then, these will be the diagonal elements of the device Hamiltonian while coupling between different neighbors, extracted from the Wannier Hamiltonian, gives the off-diagonal terms. If we have translational invariance in a direction, we can use Bloch's theorem and assume plane wave solutions. An example of this procedure can be seen on Figure 3.4.



**Figure 3.4:** (a) Geometry of a nanoribbon structure. (b) Corresponding Hamiltonian for  $x$  direction. Since  $y$  is periodic, it can be taken into account during the calculation of the bandstructure by summing up all the neighboring interaction terms.

### 3.2.1 Peierls Substitution

In order to account for external magnetic field in a tight-bonding model, an additional phase factor can be introduced to the hopping parameters.[49] This phase depends on the path integral of the vector potential  $A$  between the two sites. Therefore, depending on the device structure, the gauge degree of freedom must be taken into account and the choice should be made carefully. In general, the acquired phase depends not only on the relative coordinates between the two sites, but also their absolute position. Calculation of the Peierls' phase can then only be done once the device structure and the real space geometry is defined.

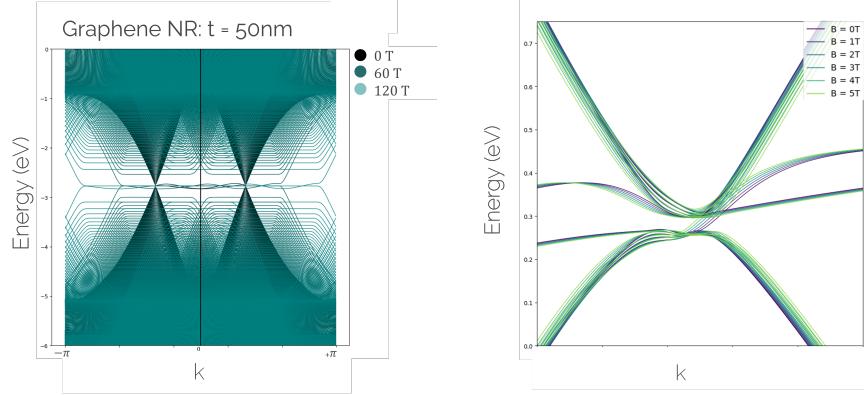
$$\phi_{1 \rightarrow 2} = \frac{e}{\hbar} \int_1^2 \vec{A} \cdot d\vec{l}$$

If the magnetic field is perpendicular to the layer plane,  $\mathbf{B} = (0, 0, B)$ , we can work in the Landau gauge,  $\mathbf{A} = (By, 0, 0)$ , to get a simple expression for the Peierls phase between sites 1 and 2.

$$\phi_{1 \rightarrow 2} = \frac{eB}{\hbar} (x_1 - x_2) \frac{(y_1 + y_2)}{2}$$

For circular devices, it may be beneficial to work in the symmetric gauge:  $\mathbf{A} = \frac{1}{2}\mathbf{B} \times \mathbf{r}$ . In this case, one can write the Peierls phase in terms of the transverse and in-plane

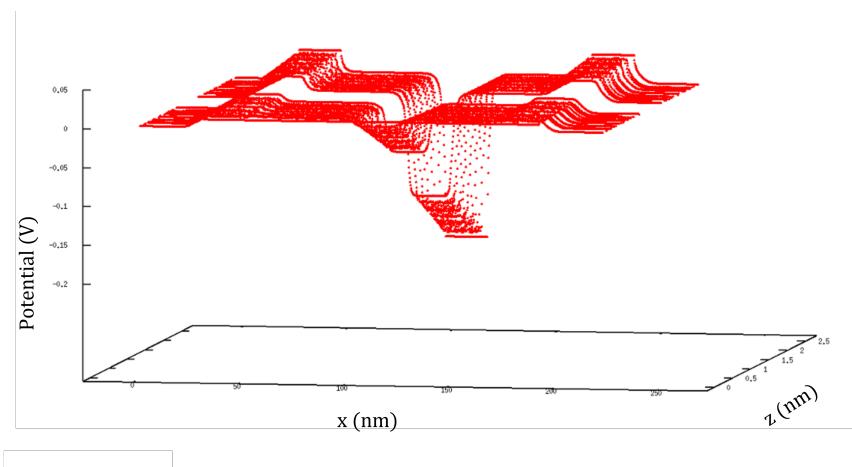
magnetic fields as well the real space coordinates of the two sites.



**Figure 3.5:** (a) Bandstructure of a zigzag-edge graphene nanoribbon ( $t=50\text{ nm}$ ). Formation of Landau-levels due to strong magnetic fields can be observed on this figure. (b) Bandstructure of a nanoribbon constructed using the hBN-BLG tristack structure under the influence of different magnetic fields.

### 3.2.2 NEGF

Gate-defined electric fields are one of the most important experimental knobs one can use to modulate the potential landscape as well as the transport properties of a device. In our framework, we use the NEGF formalism[50] to calculate the potential profile along the device. For this procedure, we need the quantum mechanical Hamiltonian of the active region (e.g. bilayer graphene flake) as well as the dielectric constants of the substrate (hBN). Gates can be incorporated as local voltage sources at given locations on the device. Once the formulation of the problem is finished, a self-consistent procedure is followed until both Poisson and Schrödinger's equations are satisfied to the determined accuracy. For typically small potentials that are used for experiments, thereby acquired solution in the active region can be introduced to the device Hamiltonian as a perturbative correction to the diagonal terms.



**Figure 3.6:** Potential profile of the device in Figure 2.6 calculated using the NEGF formalism. The shape of the potential can be adjusted by changing the gate voltages.

# QuTiP Simulation & Control Scheme

---

## 4.1 Framework

Having characterized the system from the atomistic standpoint and being able to extract its relevant material properties, we now need to take the next step and come up with a framework to describe the computational space and study the effect that the applications of various interaction Hamiltonians have on it, in order to understand to apply arbitrary two-qubit operations on our proposed device. For this we use the methods provided by the QuTiP library.

We start by acknowledging that electrons are Fermions, hence we can describe a system of electrons via its fermionic creation and annihilation operators. QuTiP provides us with the `create(N)` function generating the bosonic creation operator of an N-level system. In our case a single electron can be in one of four possible states:  $|T\uparrow\rangle, |T\downarrow\rangle, |B\uparrow\rangle, |B\downarrow\rangle$ , hence we need four creation operators:

$$\begin{aligned} c_{|T\uparrow\rangle}^* &= c^* \otimes \mathbb{I}_2 \otimes \mathbb{I}_2 \otimes \mathbb{I}_2 \\ c_{|T\downarrow\rangle}^* &= \mathbb{I}_2 \otimes c^* \otimes \mathbb{I}_2 \otimes \mathbb{I}_2 \\ c_{|B\uparrow\rangle}^* &= \mathbb{I}_2 \otimes \mathbb{I}_2 \otimes c^* \otimes \mathbb{I} \\ c_{|B\downarrow\rangle}^* &= \mathbb{I}_2 \otimes \mathbb{I}_2 \otimes \mathbb{I}_2 \otimes c^* \end{aligned}$$

However since QuTiP only provides the bosonic creation operator  $c^* = \text{create}(2)$  the above operators  $c_i^*$  describe a chain of bosons rather than a chain of fermions. We must Jordan-Wigner transform them in order to obtain operators  $a_i^*$  with fermionic commutation relations.

$$a_i^* = e^{\pi i \sum_{j < i} c_j^* c_j} c_i \quad (1)$$

Here we have slightly abused notation in the sum  $\sum_{j < i}$ , we intend some enumeration of the basis with integers, which one we use is not relevant. Having obtained these operators we identify the vacuum state  $|0\rangle$  as the common nullspace of all annihilation operators and create a basis from there, for example we would have  $|T\uparrow\rangle = c_{|T\uparrow\rangle}^* |0\rangle$ . Do notice that we are working in a 16-dimensional space. This is due to the fact that we are not limited to states where only a single electron is contained in the quantum dot. For example we could describe a singlet state of two electrons localized in the top layer within this framework as  $\frac{1}{2}(a_{|T\downarrow\rangle}^* a_{|T\uparrow\rangle}^* - a_{|T\uparrow\rangle}^* a_{|T\downarrow\rangle}^*) |0\rangle$ .

Next we can express the Pauli matrices in second quantization:

$$\begin{aligned}\sigma_x &= \sum_{L \in \{T, B\}} a_{|L\downarrow\rangle}^* a_{|L\uparrow\rangle} + a_{|L\uparrow\rangle}^* a_{|L\downarrow\rangle} \\ \sigma_y &= \sum_{L \in \{T, B\}} i a_{|L\downarrow\rangle}^* a_{|L\uparrow\rangle} - i a_{|L\uparrow\rangle}^* a_{|L\downarrow\rangle} \\ \sigma_z &= \sum_{L \in \{T, B\}} a_{|L\uparrow\rangle}^* a_{|L\uparrow\rangle} - a_{|L\downarrow\rangle}^* a_{|L\downarrow\rangle} \\ \lambda_x &= \sum_{S \in \{\uparrow, \downarrow\}} a_{|BS\rangle}^* a_{|TS\rangle} + a_{|TS\rangle}^* a_{|BS\rangle} \\ \lambda_y &= \sum_{S \in \{\uparrow, \downarrow\}} i a_{|BS\rangle}^* a_{|TS\rangle} - i a_{|TS\rangle}^* a_{|BS\rangle} \\ \lambda_z &= \sum_{S \in \{\uparrow, \downarrow\}} a_{|TS\rangle}^* a_{|TS\rangle} - a_{|BS\rangle}^* a_{|BS\rangle}\end{aligned}$$

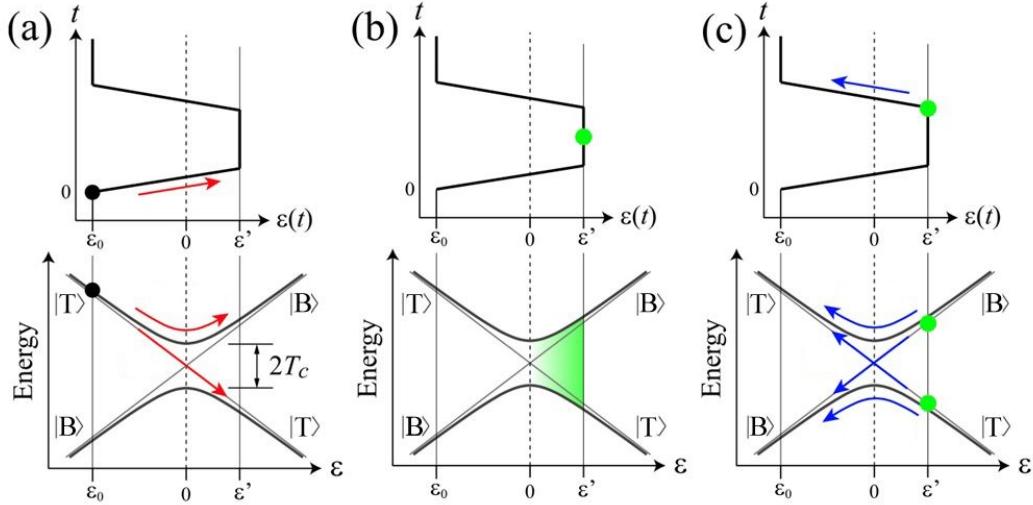
A derivation can be found in [52]. Here we write  $\sigma_i$  for the Pauli matrices operating on the spin part and  $\lambda_i$  for the Pauli matrices operating on the charge localization part of the wavefunction. Now we are ready to express any Hamiltonian we like within our QuTiP framework by decomposing it into Pauli operators. We can then use the pre-built `mesolve` function to simulate the system.

## 4.2 Landau-Zener-Stückelberg Interference

Having defined the framework for our QuTiP simulation, we can now turn to investigate how the charge qubit is modelled and driven. The underlying mechanism for driving electron-localization transitions, corresponding to charge qubit oscillations, is the so called Landau-Zener-Stückelberg Interference (LZSI). A Landau-Zener transition occurs when energy levels cross at an avoided crossing. When a two-level system (TLS) is exposed to driving with a strong enough amplitude, a series of transitions occur with a phase difference between them known as the Stückelberg phase, which can result in constructive or destructive interference. This is known as Landau-Zener-Stückelberg (LZS) interferometry and occurs in various physical systems. It has potential as an effective tool for determining the TLS's parameters and its interactions with the control fields and environment, and for fast and dependable control of the quantum system [53].

### 4.2.1 Charge Qubit Control

Our charge qubit is nothing but a TLS encoded in the localization of the electron in either one of the two layers. Hence, we can exploit LZSI to achieve qubit-manipulation, in particular being able to perform arbitrary single qubit rotations in the Bloch sphere. Thanks to the vertically applied electric field, we are able to define an energy imbalance between the two layers, measured by the detuning  $\epsilon(\vec{E})$ . At the same time, the presence of a non-zero tunnelling rate between the two layers allows us to define the avoided



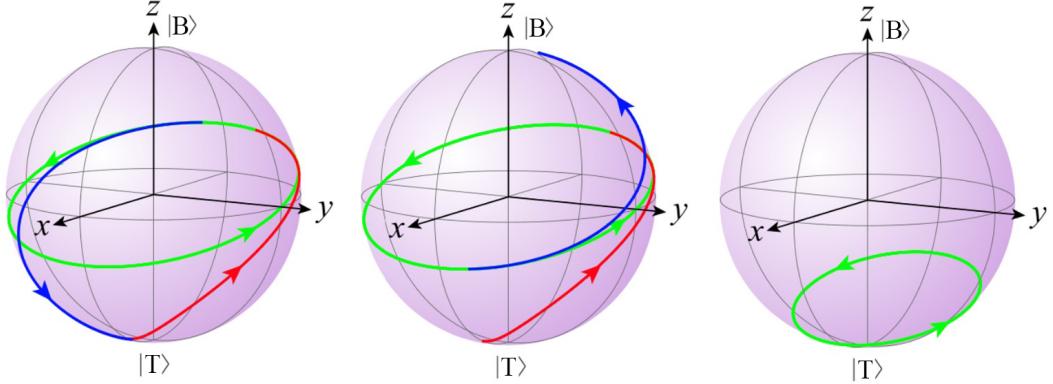
**Figure 4.1:** The three steps of a LZS drive are illustrated. In (a) the detuning is taken over the anti-crossing point and an initial phase is picked up. In the second step, (b), the free evolution with constant detuning happens. Lastly, in (c), the detuning is taken back to the original value  $\varepsilon_0$ . Picture adapted from [54].

crossing energy  $T_C$  coupling the two localization states. The Hamiltonian for the charge qubit can then be written in the basis  $|B\rangle, |T\rangle$  as:

$$\hat{H}_{CQ} = \varepsilon(E(t))\hat{\lambda}_z + T_C\hat{\lambda}_x = \begin{bmatrix} \varepsilon(E(t)) & T_C \\ T_C & -\varepsilon(E(t)) \end{bmatrix} \quad (2)$$

where  $\hat{\lambda}_i, i \in \{x, y, z\}$  are the localization-Pauli matrices for the charge qubit. Since this Hamiltonian can be mapped one-to-one to the one for a DQD charge qubit, we based our theoretical treatment for describing how the quantum system evolves under detuning modulation on [54].

A schematic of the general manipulation procedure can be found in Fig. 4.1. Here, the three different stages of the LZS approach are presented. At the beginning, the detuning is set at an initial value  $\varepsilon_0$  which ensures initialization in either  $|B\rangle$  or  $|T\rangle$ . In the first step, the detuning is changed, adiabatically or not, depending on the application, to reach a new level  $\varepsilon'$  going over the anti-crossing point situated at  $\varepsilon = 0$ . This transient stage is characterized by three independent parameters, being the initial and final values of the detuning  $\varepsilon_0$  and  $\varepsilon'$  and the falling/rising time  $t_R$ . In the second step, the detuning is kept constant for a characteristic time  $\tau_{wait}$ . It is in this stage that, if the initial transient is strictly non-adiabatic ( $t_R = 0$ ) that the anti-crossing energy starts to be effective on the qubit state. In the third and last stage, the detuning is symmetrically taken back to the original value. In the more general case in which the rising time is not zero (trapezoidal pulse), the phase picked up during the state evolution along the two eigenenergies (Fig. 4.1 a,c) plays a role in determining the final state. Theoretical calculations of the az-



**Figure 4.2:** Evolution of the Bloch vector during LZS interference operation. Color coded are the different steps as in Fig. 4.1. Notice how the green rotation during the waiting time corresponds to the situation when the detuning, determining the frequency around the  $z$  axis, is comparable with the anti-crossing energy. Picture adapted from [54].

imutual and polar angles of the final state's Bloch's vector as the pulse parameters vary can be found in the supplementary material of [55] (for the case of a triangular pulse). A representation of the evolution of the Bloch vector is illustrated in Fig. 4.2

We verified the working principle of this method in our simulation framework. We set  $\epsilon_0 = -800 \mu\text{eV}$ , the rising time  $t_R = 40 \text{ ps}$  and simulate various combination of  $\epsilon'$  and waiting time  $\tau_{\text{wait}}$ . The value of the anti-crossing energy is taken from the original work [1] to be  $T_C = \Delta/2 = 10.7 \mu\text{eV}$ . We initialized the state in the  $|B\rangle$  state and let it evolve under the resulting Hamiltonian when letting vary the waiting time and final detuning. The result is presented in Fig. 4.3.

## 4.3 Electron Dipole Spin Resonance

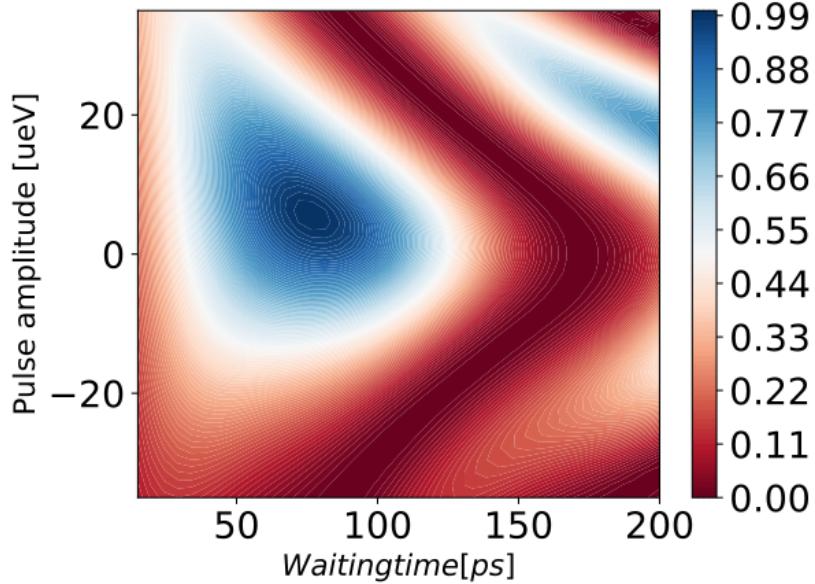
Having understood single qubit gates on the charge qubit, what remains to be designed are the single qubit gates on the spin qubit as well as the two qubit gates. We will use the effect of Electron Dipole Spin Resonance (EDSR) for both.

### 4.3.1 Spin Qubit Control

The benefit of EDSR is that it allows for fully electrical spin control. While the spin usually does not couple directly to the electric field and is controlled with magnetic field pulses, electrical control is possible in the presence of spin-orbit coupling (SOC). Recall that we can induce SOC with proximity effects. As derived in [56] when driving with an in-plane Electric field oscillating at frequency  $\omega$  with amplitude  $\tilde{E}$  we obtain an interaction Hamiltonian of the form

$$\hat{H}_{\text{int}} = \frac{2\alpha e \tilde{E} \omega}{\hbar \omega_0^2} \sin(\omega t) \sigma_x \quad (3)$$

where  $\alpha$  is the difference between the Dresselhaus and Rashba SOC contributions which we can obtain for a given setup from the atomistic simulation. This Hamiltonian, when



**Figure 4.3:** Probability of obtaining state  $|T\rangle$  after time-evolving the initial state  $|B\rangle$  with the LZSI scheme sweeping different final detunings (vertical) and waiting times (horizontal). The asymmetry is expected from the different signs of the picked-up-phase if going further the anti-crossing point or not.

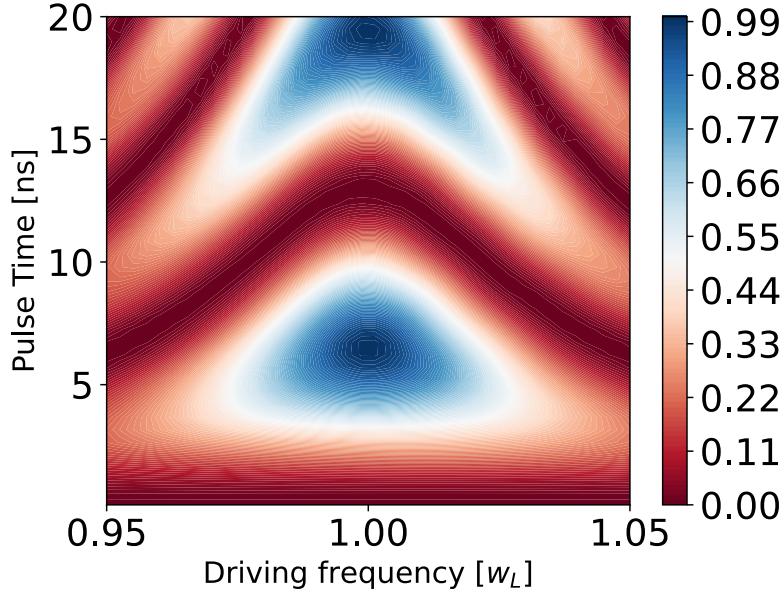
also applying a splitting field  $H_{\text{Zeeman}} = \frac{1}{2}\Delta\sigma_z$  has the usual form needed for Rabi oscillations in the rotating frame, hence are now ready to perform single qubit rotations on the spin qubit. In our framework, we simulated a system with the following parameters. Perpendicular magnetic field  $|\vec{B}| = 100\text{ mT}$ , in plane electric field  $\tilde{E} = 0.3\text{ mV/nm}$ ,  $\alpha \in [0.5; 200]\text{ }\mu\text{eV}\cdot\text{nm}$ , g factor  $g = 2$ . We visualize a heat map corresponding to the evolution under the Hamiltonian varying driving frequency  $\omega$  and pulse time resulting in the typical Chevron pattern in Fig. 4.4.

### 4.3.2 Two Qubit Gates

It is in the mechanism for two-qubit gates that we see the potential of this technology. Recall that the computational basis states for the charge qubit correspond to localization in the top and bottom layers of the BLG. Also recall that the control of the spin qubit is based on the SOC induced in the BLG through proximity effects of the BLG-WSe<sub>2</sub> heterostructure. The key to two qubit gates is to notice that we can build the heterostructure above and below the BLG asymmetrically in such a way that the SOC induced in the top and bottom layers is different. We would need to modify our interaction Hamiltonian (3) to be localization selective:

$$\hat{H}_{int} = \frac{2e\tilde{E}\omega}{\hbar\omega_0^2} \sin(\omega t)(\alpha_B\sigma_{Bx} + \alpha_T\sigma_{Tx}) \quad (4)$$

where  $\sigma_{Lx} = a_{|L\downarrow\rangle}^*a_{|L\uparrow\rangle} + a_{|L\uparrow\rangle}^*a_{|L\downarrow\rangle}$  for  $L \in \{T, B\}$ . For  $\alpha_T = \alpha_B = \alpha$  we get back the the original Hamiltonian (3) since  $\sigma_x = \sigma_{Tx} + \sigma_{Bx}$

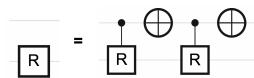


**Figure 4.4:** Rabi oscillations of the spin qubit evolving under different pulses driving the interaction Hamiltonian (3). Qubit starts in the  $| \uparrow \rangle$  state, we plot the probability of finding it in the  $| \downarrow \rangle$  state. Parameters specified in the Appendix.

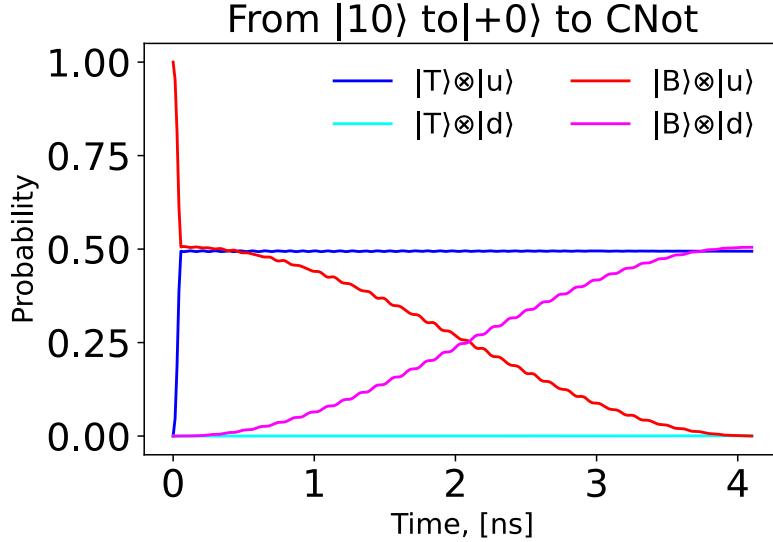
We now have a single qubit gate for the spin qubit that behaves selectively based on the state of the charge qubit: A two qubit gate! We can for example induce zero SOC in the top layer ( $\alpha_T = 0$ ) and a finite SOC in the bottom layer ( $\alpha_B = \alpha > 0$ ). If we now apply a pulse then the part of the global state localized in the  $|T\rangle$  state will be unaffected, while we drive a Rabi-oscillation on the spin qubit for the part of the state localized in the  $|B\rangle$  state. If we apply this interaction on resonance ( $\omega = \Delta/\hbar$ ) for a time  $T = \frac{\pi\hbar\omega_0^2}{\alpha_B e\tilde{E}\omega}$  then we have performed a CNOT gate with control on the charge qubit. The full process is visualized in Fig. 4.5.

The attentive reader may raise the objection that by building this selective spin-orbit coupling into the device we deprive ourselves of the ability to perform any single qubit gates on the spin qubit. This is true: Since the SOC is hard-built into the structure, all EDSR drives result in controlled rotations depending on the state of the charge qubit. This however still leads to a universal set of gates, since we can still perform true single qubit gates on the charge qubit. We can simply decompose any single qubit gate we want to apply on the spin qubit in such a way as evident in Fig. 4.6. We are only able to perform controlled rotations on the spin with control on the charge qubit, but since LZS Interference allows for arbitrary independent rotations of the charge qubit our set of gates is still universal.

We now have a complete set of gates at our disposal. Out of academic curiosity note



**Figure 4.6:** Any single qubit rotation  $R$  can be decomposed into CR gates so long as independent X gates are possible on the other qubit. In our case the upper qubit represents the charge qubit, the lower one the spin qubit.



**Figure 4.5:** Simulating a basic two-qubit gate protocol. The qubit starts in the  $|B \uparrow\rangle$  state. We rotate it to the  $\frac{1}{\sqrt{2}}(|B \uparrow\rangle + |T \uparrow\rangle) =: |+\uparrow\rangle$  state via an LZSI pulse as explained in section 4.1, which takes only  $\sim 100$  ps. We then apply a localization selective EDSR pulse as in (4) with a rectangular envelope, performing the transition  $|+\uparrow\rangle \mapsto \text{CNOT}|+\uparrow\rangle = \frac{1}{\sqrt{2}}(|B \downarrow\rangle + |T \uparrow\rangle)$ . We plot the probabilities of measuring the state in the basis states  $|T \uparrow\rangle, |T \downarrow\rangle, |B \uparrow\rangle, |B \downarrow\rangle$  at all times .Parameters specified in the Appendix.

that in particular it is possible to implement spin-to-charge conversion on this device by applying a SWAP-gate to the system. A decomposition of the SWAP-gate into gates we are able to apply directly can be found in the Appendix in Fig. A.1.

#### 4.4 Use-case of the device: spin to charge conversion

One of the possible applications of the proposed the device is spin to charge conversion. This substantially consists in exploiting the charge degree of freedom to read-out the spin state, circumventing the need for an additional quantum dot to detect the state of the spin qubit as traditionally done in the framework of Pauli Blockade [57] . The full device Hamiltonian, when the system is subject to a perpendicular electric field, perpendicular magnetic field and in-plane electric field for spin manipulation assumes the form:

$$\hat{H} = \begin{bmatrix} \epsilon(E(t)) + \Delta_Z & \Delta_{EDSR}^T(t) & T_C & 0 \\ \Delta_{EDSR}^T(t) & \epsilon(E(t)) - \Delta_Z & 0 & T_C \\ T_C & 0 & -\epsilon(E(t)) + \Delta_Z & \Delta_{EDSR}^B(t) \\ 0 & T_C & \Delta_{EDSR}^B(t) & -\epsilon(E(t)) - \Delta_Z \end{bmatrix} \quad (5)$$

when written in the basis  $\{|T \downarrow\rangle, |T \uparrow\rangle, |B \downarrow\rangle, |B \uparrow\rangle\}$ . Exploiting the location-dependent induced spin-orbit coupling, we can selectively drive EDSR pulses for spin manipulation when the electron is localized in the  $|B\rangle$  state. Whenever we want to readout the spin state we can operate a previously discussed SWAP gate, so that the spin gets mapped

## CHAPTER 4: QU TiP SIMULATION & CONTROL SCHEME

one-to-one to the localization of the electron. A total of 4 single qubit gates on the charge qubit and 5 controlled rotations on the spin qubit are necessary for this. As an implication the coherence time of both qubits must be at least  $5\frac{\pi\hbar\omega_0^2}{\alpha_B e\tilde{E}\omega}$  in order for this to be viable. We leave an analysis of the decoherence mechanisms in this device as a fruitful opportunity for future research.

# Conclusion

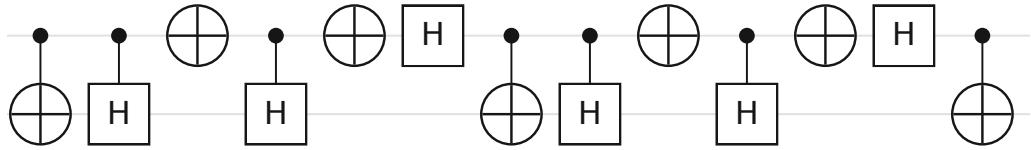
---

We are thrilled to report that we have successfully developed a comprehensive preliminary framework that demonstrates the potential of a combined architecture that incorporates bilayer graphene. This was achieved through an extensive ab-initio simulation, which began with the definition of an atomistic model and progressed through the device hamiltonian, ultimately resulting in the creation of an effective model for qubits coherent control. Our simulation results reveal that this experimental potential quantum computing platform has been fleshed out to a substantial degree, offering a promising foundation for future research and development. This project has created numerous opportunities for future exploration, including the examination of decoherence mechanisms, the investigation of fabrication techniques, and further fundamental studies on the potential for coupling these devices. In addition to these areas of future work, we are particularly interested in the characterization and modeling of coherence times for both charge and spin qubits using QuTiP, and the comparison of these results with actual experiments, such as those performed on charge qubits as outlined in [19]. It is important to acknowledge that the path to full understanding and control of this quantum technology will be a long and challenging journey. However, we are filled with hope and optimism that this platform has the potential to one day compete with other existing quantum computing platforms in terms of scalability, coherence time, and gate fidelity. Our results represent a significant step forward in the field of quantum computing and we look forward to continuing to advance our understanding of this cutting-edge technology through future research and development.

## APPENDIX A

# Supplementary Figures

---



**Figure A.1:** A decomposition of the SWAP gate into the gates described in this paper. The first qubit represents the charge qubit, the second the spin qubit. While you can see 13 gates here recall that two consecutive single qubit gates as well as two consecutive controlled rotations can be combined into a single operation. Hence we have to apply a total of 5 controlled spin rotations and 4 single qubit charge gates.

## APPENDIX B

# Code

---

```
1 %matplotlib inline
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import matplotlib
5 import scipy as sci
6 matplotlib.rc('xtick', labelsize=20)
7 matplotlib.rc('ytick', labelsize=20)
8 # !pip install qutip
9 import numpy as np
10 from qutip import *
11 from tqdm import tqdm
12 import matplotlib.colors as mcolors
13 bar=ui.progressbar.EnhancedTextProgressBar(chunk_size=.1)
14 #-----
15 M=2      #Number of Localizations
16 N=2*M #total number of operators (Two positions times two spin
        polarizations)
17 zero=tensor([Qobj(np.zeros((2,2))) for i in range(N)])
18
19 fs=[tensor([create(2) if n==m else qeye(2) for m in range(N)]) for n in
      range(N)] #Bosonic creation operators
20 def sumj(j):
21     s=tensor([Qobj(np.zeros((2,2))) for i in range(N)]) #zero operator
22     i=0
23     while i<j:
24         s=s+fs[i].dag()*fs[i]
25         i+=1
26     return s
27 cs=[(1j*np.pi*sumj(j)).expm()*fs[j] for j in range (N)] #Jordan Wigner
        Transform
28 ns=[c*c.dag() for c in cs]
29
30 longc=cs.copy()
31 longn=ns.copy()
32 cs=[[cs[2*i],cs[2*i+1]] for i in range(M)] #reshaped cs containing
        fermionic operators
33 ns=[[ns[2*i],ns[2*i+1]] for i in range(M)]
34
35 Num=zero #Total number of electrons
36 for n in longn:
37     Num=Num+n
38 #-----
39 #Check commutation relations
40 for i in range(int(N/2)):
41     for j in range(int(N/2)):
```

## APPENDIX B: CODE

```

42     for a in range(2):
43         for b in range(2):
44             if cs[i][a].dag()*cs[j][b]+cs[j][b]*cs[i][a].dag() != tensor([qeye
45 (2) for i in range(N)])*(i==j)*(a==b):
46                 print("Failure: ",i,j,a,b)
47 for i in range(int(N/2)):
48     for a in range(2):
49         if cs[i][a].dag()*cs[i][a].dag() != tensor([qeye(2) for i in range(N
50 )])*0:
51             print("Failure: ",i,a)
52 for i in range(int(N/2)):
53     for j in range(int(N/2)):
54         for a in range(2):
55             for b in range(2):
56                 if cs[i][a].dag()*cs[j][b].dag()+cs[j][b].dag()*cs[i][a].dag() !=
57 tensor([qeye(2) for i in range(N)])*0:
58                     print("Failure: ",i,j,a,b)
59 #If no Failure is printed all is fulfilled
60 #
61 #-----#
62 #Vacuum state
63 A=np.array([c.dag() for c in longc]).reshape(N*2**N,2**N)
64 vac=Qobj((sci.linalg.null_space(A)).reshape((2**N,)),shape=(2**N,1),dims
65 =[[2]*N,[1]*N],type='ket') #define the vacuum state
66 #
67 #-----#
68 #Spin operators
69 sx=[cs[i][0]*cs[i][1].dag()+cs[i][1]*cs[i][0].dag() for i in range(M)]
70 sy=[-1j*cs[i][0]*cs[i][1].dag()+1j*cs[i][1]*cs[i][0].dag() for i in range(
71 M)]
72 sz=[cs[i][0]*cs[i][0].dag()-cs[i][1]*cs[i][1].dag() for i in range(M)]
73 #Location Pauli operators (Only sensible if M=2)
74 lx=[cs[0][i]*cs[1][i].dag()+cs[1][i]*cs[0][i].dag() for i in range(2)]
75 ly=[-1j*cs[0][i]*cs[1][i].dag()+1j*cs[1][i]*cs[0][i].dag() for i in range
76 (2)]
77 lz=[cs[0][i]*cs[0][i].dag()-cs[1][i]*cs[1][i].dag() for i in range(2)]
78 #
79 ## Construct Basis
80 #
81 c={'Tu':cs[0][0],'Td':cs[0][1],
82   'Bu':cs[1][0],'Bd':cs[1][1],} #dictionary
83 #
84 # We will mainly work in the basis of the subspace of having a total of
85 # two electrons
86 # Convention {uu,ud,du,dd,S0,OS} where S = ud-du
87 # We also use the convention of first creating in the second qubit, so uu
88 # = c_iu c_2u vac
89 #
90 # total Basis: {vac,u0,d0,Ou,Od,uu,ud,du,dd,S0,OS,Su,Sd,uS,dS,SS}
91 Basis=[vac,
92       c['Tu']*vac,
93       c['Td']*vac,
94       c['Bu']*vac,
95       c['Bd']*vac,
96       c['Tu']*c['Bu']*vac,
97       c['Tu']*c['Bd']*vac,
98       c['Td']*c['Bu']*vac,
99       c['Td']*c['Bd']*vac,
```

## APPENDIX B: CODE

```

90     c[ 'Tu' ]*c[ 'Td' ]*vac/2-c[ 'Td' ]*c[ 'Tu' ]*vac/2,
91     c[ 'Bu' ]*c[ 'Bd' ]*vac/2-c[ 'Bd' ]*c[ 'Bu' ]*vac/2,
92     c[ 'Tu' ]*c[ 'Td' ]*c[ 'Bu' ]*vac/2-c[ 'Td' ]*c[ 'Tu' ]*c[ 'Bu' ]*vac/2,
93     c[ 'Tu' ]*c[ 'Td' ]*c[ 'Bd' ]*vac/2-c[ 'Td' ]*c[ 'Tu' ]*c[ 'Bd' ]*vac/2,
94     c[ 'Tu' ]*c[ 'Bu' ]*c[ 'Bd' ]*vac/2-c[ 'Tu' ]*c[ 'Bd' ]*c[ 'Bu' ]*vac/2,
95     c[ 'Td' ]*c[ 'Bu' ]*c[ 'Bd' ]*vac/2-c[ 'Td' ]*c[ 'Bd' ]*c[ 'Bu' ]*vac/2,
96     c[ 'Tu' ]*c[ 'Td' ]*(c[ 'Bu' ]*c[ 'Bd' ]*vac/2-c[ 'Bd' ]*c[ 'Bu' ]*vac/2)/2-c[ 'Td' ]*c[ 'Tu' ]*(c[ 'Bu' ]*c[ 'Bd' ]*vac/2-c[ 'Bd' ]*c[ 'Bu' ]*vac/2)/2]
97
98 ket={ '00':Basis[0] ,
99   'u0':Basis[1] ,
100  'd0':Basis[2] ,
101  'ou':Basis[3] ,
102  'od':Basis[4] ,
103  'uu':Basis[5] ,
104  'ud':Basis[6] ,
105  'du':Basis[7] ,
106  'dd':Basis[8] ,
107  'so':Basis[9] ,
108  'os':Basis[10] ,
109  'su':Basis[11] ,
110  'sd':Basis[12] ,
111  'us':Basis[13] ,
112  'ds':Basis[14] ,
113  'ss':Basis[15]}
114 bra={s:ket[s].dag() for s in ket}
115
116 xbas = {'+u': (ket['ou'] + ket['u0'])/np.sqrt(2) ,
117   '-u': (ket['ou'] - ket['u0'])/np.sqrt(2) ,
118   '+d': (ket['od'] + ket['d0'])/np.sqrt(2) ,
119   '-d': (ket['od'] - ket['d0'])/np.sqrt(2)
120   }
121 #
122 ## Construct Labframe Hamiltonian
123 #
124 # Define a function for summing up Hamiltonians of different types
125 func = lambda x:x
126 foo = type(func)
127 def Hsum(Hs,t_dep=False):
128   # Passing a list of hamiltonians Hs Hsum will return the sum of these
129   # hamiltonians ,
130   # taking care of possible time dependent and independent terms.
131   func=lambda x:x
132   Hfuncs=[h for h in Hs if type(h)==foo]
133   Hconst=[h for h in Hs if type(h)==type(zero)]
134   H=zero
135   for Hi in Hconst:
136     H=H+Hi
137   if Hfuncs==[] and not t_dep:
138     return H
139   else:
140     def out(t,args):
141       Ht=zero
142       for h in [h(t,args) for h in Hfuncs]:
143         Ht=Ht+h
144     return Ht+H

```

## APPENDIX B: CODE

```

144     return out
145 # Define a function taking operators and states to the rotating frame
146 def evolve(s,t,op,hbar=1):
147     assert type(op)==type(zero)
148     assert type(s) ==type(zero)
149     U=(1j*op*t/hbar).expm()
150     if s.type=='oper':
151         return U*s*U.dag()
152     if s.type=='ket':
153         return U*s
154     if s.type=='bra':
155         return s*U.dag()
156 #
157 #Define parameters from physics and form of Hamiltonians
158 e_ov_pl = 1519267.4605 #C/(J * ns)
159
160 ketbra=lambda s:ket[s]*bra[s]
161
162 #total spin operators
163 Sx,Sy,Sz=zero,zero,zero
164 for i in range(M):
165     Sx+=sx[i]
166     Sy+=sy[i]
167     Sz+=sz[i]
168
169 #total location pauli operators
170 Lx,Ly,Lz=zero,zero,zero
171 for i in range(2):
172     Lx+=lx[i]
173     Ly+=ly[i]
174     Lz+=lz[i]
175
176 Loc=zero
177 for i in range(M):
178     for j in range(2):
179         Loc+=(-1)**i*cs[i][j]*cs[i][j].dag()
180
181 E1      = 1           # Energies of the Basis states
182 E2      = 1.03
183 E3      = -E1
184 E4      = -E1+.03
185 t       = 2           # Tunnelling rate
186 SOT     = 3           # SpinOrbit Coupling
187 SOB     = 0
188 landau = 2           # g-factor
189
190 # def H_model(E1=E1,E2=E2,E3=E3,E4=E4,SOT=SOT,SOB=SOB,t=t):
191 #     s = zero
192 #     s += E1*ketbra('u0')+E2*ketbra('d0')+E3*ketbra('ou')+E4*ketbra('od')
193 #     s += SOT*ket['u0']*bra['d0']+SOB*ket['ou']*bra['od']+np.conj(SOB)*
194 #         ket['od']*bra['ou']+np.conj(SOT)*ket['d0']*bra['u0']
195 #     s += t*(ket['u0']*bra['ou']+ket['d0']*bra['od'])+np.conj(t)*(ket['o'
196 #         u']*bra['u0']+ket['od']*bra['d0'])
197 #     return s

```

## APPENDIX B: CODE

```

198 def H_chargesplit(E):
199     if type(E)!=foo:
200         return E*Lz
201     else:
202         return lambda t,args: E(t)*Lz
203
204 def H_spinsplit(E):
205     return E*Sz
206
207 def H_EDSR(alpha,w,w0,Ampl,phi=0):
208     if type(alpha)!=type((0,0)):
209         return lambda t,arg: 2*alpha*Ampl*e_ov_pl*w/w0**2*Sx*np.cos(w*t+
210         phi) #+detune*Sz
211     else:
212         return lambda t,arg: 2*(alpha[0]*sx[0]+alpha[1]*sx[1])*Ampl*
213         e_ov_pl*w/w0**2*np.cos(w*t+phi) #+detune*Sz
214 def H_LZSI(eps,w,phi=0):
215     if type(eps)!=foo:      #if time independent eps is passed as float
216         return lambda t,args: eps*Lx#*np.cos(w*t+phi) #+detune*Lz
217     else: #if time dependent eps is passed as a callable
218         return lambda t,args: eps(t)*Lx#*np.cos(w*t+phi) #+detune*Lz
219
220 #TODO: Build an estimator that returns E1:4 and t as a function of the
221 #       applied Field.
222 stuff=[‘u0’,’d0’,’0u’,’0d’]
223
224 stuff_x = [‘+u’,’-u’,’+d’,’-d’]
225
226 # stuff =[‘u0’,’d0’,’0u’,’0d’,’uu’,’ud’,’du’,’dd’,’S0’,’0S’,’Su’,’Sd’,’uS
227 #       ’,’dS’,’SS’]
228 def subspace(H,basis=None):
229     if basis==None:
230         basis=[ket[s] for s in stuff]
231     return np.array([[bi.dag()*H*bj for bj in basis] for bi in basis]).\
232     reshape((len(basis),len(basis)))
233
234 #-----
235 # Charge Qubit Drive
236 #-----
237 #Define the parameters for charge qubit
238
239 E_charge=-800    #ueV
240 eps   = 10.7      #ueV
241
242 #Define parameters for Spin qubit
243 #... and its driving parameters
244
245 alpha = (0,2000)    #ueV * A = ueV * Angstrom
246 Efield = 3e-7      #V/A    volt over Angstrom
247
248 mu_bohr = 5.78838e-5          #Bohr’s Magneton    eV/T
249 planck = 6.582119569e-7        # Planck’s constant eV*ns    this is just
250             # to calculate the value of w0 in GHz
251 m_e_c2 = 510998.95000 #eV
252 c2 = (299792458)**2      # (nm/ns)^2
253 r0 = 20 #nm
254 landau = 2

```

## APPENDIX B: CODE

```

248 B = 0.2      # T
249 w0 = planck*c2/(0.05*m_e_c2*r0**2)    #GHz
250 w_spin = w0*0.0005      #GHz
251
252 E_spin = (landau*mu_bohr*B)/2 *1e6   #ueV
253
254 #Define timescales for pulses and hbar for plotting
255 hbar = 6.582119569e2 #ueV*ps
256
257 on1=0/hbar          #onset of first drive
258 T1=0                 #     ps        #length of first drive
259 on2=0/hbar          #wait before of second drive
260 T2=100/hbar         #     ps        #length of second drive
261 chill=20/hbar
262
263 T=on1+T1+on2+T2+chill
264
265 w_charge = E_charge*2
266
267 omega_R = 2*2*alpha[0]*Efield*e_ov_pl*w_spin/w0**2/hbar *1e-3      #GHz
268
269 #Printing values:
270 print("E_charge = ",E_charge, "ueV")
271 print("E_spin = ", E_spin, "ueV")
272 print("w0 = ", w0, "GHz")
273 print("w_spin = ", w_spin, "GHz")
274 print("omega_Rabi = ",omega_R, "GHz")
275 print("E_EDSR = ", 2*alpha[0]*Efield*e_ov_pl*w_spin/w0**2, "ueV")
276 print("E_Charge_det = ", eps, "ueV")
277
278 def H(t,arg):
279     if t<0:
280         return zero
281     if t<on1+T1:
282         return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),H_LZSI(
283             eps,w_charge)),t_dep=True)(t,arg)
283     if t<on1+T1+on2+T2:
284         return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),H_LZSI(
285             eps,w_charge),H_EDSR(alpha,w_spin,w0,Efield)),t_dep=True)(t,arg)
285     if t<=T:
286         return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),H_LZSI(
287             eps,w_charge)),t_dep=True)(t,arg)
287     if t>T:
288         return zero
289 psi0=(ket['0u'])
290 N=300
291 tlist=np.linspace(0,T,N)
292 result=mesolve(H,psi0,tlist,progress_bar=bar,options=Options(store_states=
293     True))
294 #-----
295 #Plot probabilities
296 cols=['blue','cyan','red','magenta']
297 for thing,name,col in zip(stuff,names,cols):
298     exs=[np.abs(s.overlap(ket[thing]))**2 for s in result.states]
299     plt.plot(tlist*hbar,np.array(exs),label=name,color=col)

```

## APPENDIX B: CODE

```

300 print(np.max(np.array(exs)))
301 #plus=(ket['u0']+ket['d0'])/np.sqrt(2)
302 #exs=[np.abs(s.overlap(evolve(plus,t,Hsum((H_spinsplit(E_spin),
303 H_chargesplit(E_charge))))))**2 for t,s in zip(tlist,result.states)]
304 #plt.plot(tlist,np.array(exs),label='|+>')
305 #print(np.max(np.array(exs)))
306 plt.legend()
307 plt.title("Rabi oscillations",size=20)
308 plt.xlabel("Time, [ps]",size=15)
309 plt.ylabel("Probability",size=15)
310 # plt.savefig('rabi.eps',bbox_inches='tight')
311 #
312 # Spin Qubit Drive
313 #
314 #Define the parameters for charge qubit
315 E_charge=-800 #ueV
316 eps = 10.7 #ueV
317
318 #Define parameters for Spin qubit
319 #... and its driving parameters
320
321 alpha = (0,2000) #ueV * A = ueV * Angstrom
322 Efield = 3e-7 #V/A volt over Angstrom
323
324 mu_bohr = 5.78838e-5 #Bohr's Magneton eV/T
325 planck = 6.582119569e-7 # Planck's constant eV*ns this is just
326 to calculate the value of w0 in GHz
327 m_e_c2 = 510998.95000 #eV
328 c2 = (299792458)**2 # (nm/ns)^2
329 r0 = 15 #radius of the QD in nm
330 landau = 2 #gyromagnetic ratio
331 B = 0.1 # T , Vertical magnetic field
332 w0 = planck*c2/(2*pi*m_e_c2*r0**2) #GHz
333 wL = (landau*mu_bohr*B)/planck #GHz
334 w_drive = wL #GHz
335
336 E_spin = (landau*mu_bohr*B)/planck/2 #ueV
337
338 omega_R_spin = 2*alpha[1]*Efield*e_ov_pl*w_drive/w0**2/hbar #GHz
339
340 #Define timescales for pulses and hbar for plotting
341 hbar = 6.582119569e-1 #ueV*ns
342
343 on1=0/hbar #onset of first drive
344 T1=0/hbar # ns length of first drive
345 on2=0/hbar #wait before of second drive
346 T2=np.pi/omega_R_spin/hbar # ns length of second drive
347 chill=T1/15
348
349 T=on1+T1+on2+T2+chill
350 print("T = ", T)
351 w_charge = E_charge*2
352
353

```

## APPENDIX B: CODE

```

354 #Printing values:
355 print("E_charge = ", E_charge, "ueV")
356 print("E_spin = ", E_spin, "ueV")
357 print("w0 = ", w0, "GHz")
358 print("wL = ", wL, "GHz")
359 print("w_drive = ", w_drive, "GHz")
360 print("omega_Rabi = ", omega_R, "GHz")
361 print("E_EDSR = ", 2*alpha[0]*Efield*e_ov_pl*w_drive/w0**2, "ueV")
362 print("E_Charge_det = ", eps, "ueV")
363
364 def H(t,arg):
365     if t<0:
366         return zero
367     if t<on1+T1:
368         return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),H_LZSI(
369         eps,w_charge)),t_dep=True)(t,arg)
370     if t<on1+T1+on2+T2:
371         return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),H_LZSI(
372         eps,w_charge),H_EDSR(alpha,w_drive,w0,Efield)),t_dep=True)(t,arg)
373     if t<=T:
374         return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),H_LZSI(
375         eps,w_charge)),t_dep=True)(t,arg)
376     if t>T:
377         return zero
378 psi0=(ket['0u'])
379 N=200
380 tlist=np.linspace(0,T,N)
381 result=mesolve(H,psi0,tlist,progress_bar=bar,options=Options(store_states=
382     True))
383 #-----
384 #Plot probabilities
385 cols=['blue','cyan','red','magenta']
386 for thing,name,col in zip(stuff,names,cols):
387     exs=[np.abs(s.overlap(ket[thing]))**2 for s in result.states]
388     plt.plot(tlist/3*hbar,np.array(exs),label=name,color=col)
389     print(np.max(np.array(exs)))
390 #plus=(ket['u0']+ket['d0'])/np.sqrt(2)
391 #exs=[np.abs(s.overlap(evolve(plus,t,Hsum((H_spinsplit(E_spin),
392     H_chargesplit(E_charge))))))**2 for t,s in zip(tlist,result.states)]
393 #plt.plot(tlist,np.array(exs),label='|+>')
394 #print(np.max(np.array(exs)))
395 plt.legend(fontsize = 16)
396 plt.title("Rabi oscillations with EDSR",size=20)
397 plt.xlabel("Time, [ns]",size=18)
398 plt.ylabel("Probability",size=18)
399 plt.savefig('Rabi_spin_EDSR.svg',bbox_inches='tight')
400 #-----
401 # ColorMap for Spin Qubit
402 #
403 #Define the parameters for charge qubit
404 E_charge=-800 #ueV      #Reduce this value to fasten the simulation, it
405     doesn't influence the spin qubit behaviour
406 eps = 10.7    #ueV
407

```

## APPENDIX B: CODE

```

404 #Define parameters for Spin qubit
405 #... and its driving parameters
406
407 alpha = (0,20)      #ueV * A = ueV * Angstrom
408 Efield = 3e-5       #V/A   volt over Angstrom
409
410 mu_bohr = 5.783838e-5          #Bohr's Magneton eV/T
411 planck = 6.582119569e-7        # Planck's constant eV*ns this is just
        to calculate the value of w0 in GHz
412 m_e_c2 = 510998.95000          #eV
413 c2 = (299792458)**2           # (nm/ns)^2
414 r0 = 15                         #radius of the QD
415 landau = 2                      #gyromagnetic ratio
416 B = 0.1                          # T , Vertical magnetic field
417 w0 = planck*c2/(2*1*m_e_c2*r0**2) #GHz
418 wL = (landau*mu_bohr*B)/planck  #GHz
419 # w_drive = wL                  #GHz
420
421 i = 0
422 j = 0
423 count = 0
424 E_spin = (landau*mu_bohr*B)/planck/2    #ueV
425
426 #Define timescales for pulses and hbar for plotting
427 hbar = 6.582119569e-1 #ueV*ns
428 omega_R_spin = 2*alpha[1]*Efield*e_ov_pl*w_drive/w0**2/hbar      #GHz
429 on1=0/hbar                   #onset of first drive
430 T1=0/hbar                     # ns          #length of first drive
431 on2=0/hbar                   #wait before of second drive
432 T2=np.pi/omega_R_spin/hbar   # ns          #length of second drive
433 chill=0/hbar
434
435 time_list = np.linspace(0.1,20,40)
436 w_drive_list = np.linspace(wL*0.95,wL*1.05,40) #[0,94,188]
437 final_state = [ [0]*len(w_drive_list) for i in range(len(time_list)) ]
438
439 N=100
440 for T2 in time_list:
441     T=on1+T1+on2+T2+chill
442     tlist=np.linspace(0,T,N)
443     for w_drive in w_drive_list:
444         w_charge = E_charge*2
445         def H(t,arg):
446             if t<0:
447                 return zero
448             if t<on1+T1:
449                 return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),
450 H_LZSI(eps,w_charge)),t_dep=True)(t,arg)
451                 if t<on1+T1+on2+T2:
452                     return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),
453 H_LZSI(eps,w_charge),H_EDSR(alpha,w_drive,w0,Efield)),t_dep=True)(t,arg)
454             if t<=T:
455                 return Hsum((H_spinsplit(E_spin),H_chargesplit(E_charge),
456 H_LZSI(eps,w_charge)),t_dep=True)(t,arg)
457             if t>T:

```

## APPENDIX B: CODE

```

455         return zero
456
457     psi0=(ket['0u'])
458     result=mesolve(H,psi0,tlist,options=Options(store_states=True))
459     final_state[i][j]=np.abs(result.states[-1].overlap(ket['0d']))**2
460     j=j+1
461     count +=1
462     print("Round",count,"/",len(w_drive_list)*len(time_list))
463     j=0
464     i=i+1
465 #
466 ##### Plot color map #####
467
468 plt.contourf(w_drive_list/wL,time_list ,final_state , 100, cmap=matplotlib
    .cm.get_cmap('RdBu'))
469 plt.xlabel(r'Driving frequency $[w_L]$', fontsize=16)
470 plt.ylabel(r'Pulse Time [ns]', fontsize=16)
471 plt.colorbar()
472 plt.savefig('Colormap_EDSR_40x40.svg', dpi = 200, bbox_inches = 'tight')
473
474 print("w_drive_list = ",w_drive_list/wL,"wL")
475 print("time_list = ",time_list)
476 #
477 # ColorMap for Charge Qubit
478 #
479 ##### Solve for different pulse amplitdes and waiting times
    #####
480 E_charge= -800
481 E_spin = 1
482 E_charge_drive= 0
483
484 hbar = 6.582119569e2 #ueV*ps
485 alpha= (.1,0)
486 alph = .1
487 eps = 10.7           #ueV
488
489 # First : Drive CNot via location selective EDSR
490 # Second : Drive Hadamard on both spin and localization
491 # Third : Drive another CNot via location selective EDSR
492 # Fourth : Drive another Hadamard on both spin and localization
493 # Fifth : Drive another CNot via location selective EDSR
494
495 count = 0
496 on1=1/hbar                      #onset of drive
497 drivetime= 40/hbar                 #length of second drive
498 # stay=np.pi*60/hbar              #wait before second drive
499 chill=10/hbar
500 i = 0
501 j = 0
502 E_charge_drive_list = np.linspace(-35,35,40) #[-100,0,200,300,400]#np.
    linspace(-200,200)
503 stay_list = np.linspace(15,200,40) #[0,94,188]
504 final_state = [ [0]*len(stay_list) for i in range(len(E_charge_drive_list)
    )]
505 stay_array = np.array(stay_list)/hbar
506

```

## APPENDIX B: CODE

```

507 for E_charge_drive in E_charge_drive_list:
508     for stay in stay_array:
509         T=(on1+drivetime+stay+drivetime+chill)*1.1
510         N=400
511         tlist=np.linspace(0,T,N)
512         Es = []
513         w_charge = E_charge*2
514         w_spin = E_spin*2
515
516     def E(t):
517         if t<on1:
518             return E_charge
519         elif t<=(drivetime+on1):
520             return E_charge+(E_charge_drive-E_charge)*(t-(on1))/drivetime
521         elif t<=(drivetime+on1+stay):
522             return E_charge_drive
523         elif t<=(drivetime+on1+stay+drivetime):
524             return E_charge_drive+(E_charge-E_charge_drive)*(t-(drivetime+on1+
525                     stay))/drivetime
526         else: return E_charge
527     Es=[E(t) for t in tlist]
528
529     def H(t,arg):
530         return Hsum((H_spinsplit(E_spin),H_chargesplit(E(t)),H_LZSI(eps,
531         w_charge)),t_dep=True)(t,arg)
532         for t in tlist:
533             H(t,0)
534
535     def phase_c(t):
536         if t<0:
537             return 0
538
539     def phase_s(t):
540         None
541
542         psi0=(ket['u0'])
543         tlist=np.linspace(0,T,N)
544         result=mesolve(H,psi0,tlist,options=Options(store_states=True))
545         final_state[i][j]=np.abs(result.states[-1].overlap(ket['0u']))**2
546         # print(final_state)
547         # print(i,j)
548         j=j+1
549         count +=1
550         print("Round",count,"/",len(E_charge_drive_list)*len(stay_list))
551         j=0
552         i=i+1
553
554         #-----
555         ##### Plot color map #####
556         # print(final_state)
557         plt.contourf(stay_list,E_charge_drive_list ,final_state , 100, cmap=
558             matplotlib.cm.get_cmap('RdBu'))
559         plt.xlabel(r'$Waiting\ time\ [ps]$', fontsize=16)
560         plt.ylabel(r'$Pulse\ amplitude\ [ueV]$', fontsize=16)
561         plt.colorbar()
562         # plt.savefig('Colormap_40x40_true_values.svg', dpi = 200, bbox_inches =
563             'tight')

```

## APPENDIX B: CODE

559 #

**Listing B.1:** Complete QuTiP Simulation

# Bibliography

1. Cao, J., Gandus, G., Agarwal, T., Luisier, M. & Lee, Y. Dynamics of van der Waals charge qubit in two-dimensional bilayer materials: Ab initio quantum transport and qubit measurement. *Physical Review Research* **4**, 043073 (2022).
2. Gorman, J., Hasko, D. & Williams, D. Charge-qubit operation of an isolated double quantum dot. *Physical review letters* **95**, 090502 (2005).
3. Van Meter, R. & Horsman, D. A blueprint for building a quantum computer. *Communications of the ACM* **56**, 84–93 (2013).
4. DiVincenzo, D. P. The Physical Implementation of Quantum Computation. *Fortschritte der Physik* **48**, 771–783 (Sept. 2000).
5. McCann, E. & Koshino, M. The electronic properties of bilayer graphene. *Reports on Progress in physics* **76**, 056503 (2013).
6. McCann, E., Abergel, D. S. & Fal'ko, V. I. Electrons in bilayer graphene. *Solid state communications* **143**, 110–115 (2007).
7. Castro, E. V. *et al.* Biased bilayer graphene: semiconductor with a gap tunable by the electric field effect. *Physical review letters* **99**, 216802 (2007).
8. Oostinga, J. B., Heersche, H. B., Liu, X., Morpurgo, A. F. & Vandersypen, L. M. Gate-induced insulating state in bilayer graphene devices. *Nature materials* **7**, 151–157 (2008).
9. Banszerus, L. *et al.* Spin-valley coupling in single-electron bilayer graphene quantum dots. *Nature communications* **12**, 1–7 (2021).
10. Eich, M. *et al.* Spin and valley states in gate-defined bilayer graphene quantum dots. *Physical Review X* **8**, 031023 (2018).
11. Tong, C. *et al.* Tunable valley splitting and bipolar operation in graphene quantum dots. *Nano Letters* **21**, 1068–1073 (2021).
12. Novoselov, K. S., Colombo, L., Gellert, P., Schwab, M., Kim, K., *et al.* A roadmap for graphene. *nature* **490**, 192–200 (2012).
13. Zhang, Y. *et al.* Direct observation of a widely tunable bandgap in bilayer graphene. *Nature* **459**, 820–823 (June 2009).
14. Goossens, A. ( M. *et al.* Gate-defined confinement in bilayer graphene-hexagonal boron nitride hybrid devices. *Nano letters* **12**, 4656–4660 (2012).
15. Dean, C. R. *et al.* Boron nitride substrates for high-quality graphene electronics. *Nature nanotechnology* **5**, 722–726 (2010).
16. Wang, L. *et al.* One-dimensional electrical contact to a two-dimensional material. *Science* **342**, 614–617 (2013).
17. Overweg, H. *et al.* Electrostatically induced quantum point contacts in bilayer graphene. *Nano letters* **18**, 553–559 (2018).

## BIBLIOGRAPHY

18. Ihn, T. *Semiconductor Nanostructures: Quantum states and electronic transport* (OUP Oxford, 2009).
19. Petersson, K., Petta, J., Lu, H. & Gossard, A. Quantum coherence in a one-electron semiconductor charge qubit. *Physical Review Letters* **105**, 246804 (2010).
20. Burkard, G., Ladd, T. D., Nichol, J. M., Pan, A. & Petta, J. R. Semiconductor spin qubits. *arXiv preprint arXiv:2112.08863* (2021).
21. Loss, D. & DiVincenzo, D. P. Quantum computation with quantum dots. *Phys. Rev. A* **57**, 120–126. <https://link.aps.org/doi/10.1103/PhysRevA.57.120> (1 Jan. 1998).
22. Elzerman, J. *et al.* Single-shot read-out of an individual electron spin in a quantum dot. *nature* **430**, 431–435 (2004).
23. Hanson, R., Kouwenhoven, L. P., Petta, J. R., Tarucha, S. & Vandersypen, L. M. Spins in few-electron quantum dots. *Reviews of modern physics* **79**, 1217 (2007).
24. Petta, J. R. *et al.* Coherent manipulation of coupled electron spins in semiconductor quantum dots. *Science* **309**, 2180–2184 (2005).
25. Koppens, F. H. *et al.* Driven coherent oscillations of a single electron spin in a quantum dot. *Nature* **442**, 766–771 (2006).
26. Qiao, H. *et al.* Coherent multispin exchange coupling in a quantum-dot spin chain. *Physical Review X* **10**, 031006 (2020).
27. Shulman, M. D. *et al.* Demonstration of entanglement of electrostatically coupled singlet-triplet qubits. *science* **336**, 202–205 (2012).
28. Itoh, K. M. & Watanabe, H. Isotope engineering of silicon and diamond for quantum computing and sensing applications. *MRS communications* **4**, 143–157 (2014).
29. Yoneda, J. *et al.* A quantum-dot spin qubit with coherence limited by charge noise and fidelity higher than 99.9%. *Nature nanotechnology* **13**, 102–106 (2018).
30. Veldhorst, M. *et al.* An addressable quantum dot qubit with fault-tolerant control-fidelity. *Nature nanotechnology* **9**, 981–985 (2014).
31. Yang, C. *et al.* Silicon qubit fidelities approaching incoherent noise limits via pulse engineering. *Nature Electronics* **2**, 151–158 (2019).
32. Huang, W. *et al.* Fidelity benchmarks for two-qubit gates in silicon. *Nature* **569**, 532–536 (2019).
33. Veldhorst, M., Eenink, H., Yang, C.-H. & Dzurak, A. S. Silicon CMOS architecture for a spin-based quantum computer. *Nature communications* **8**, 1–8 (2017).
34. Yang, C. H. *et al.* Operation of a silicon quantum processor unit cell above one kelvin. *Nature* **580**, 350–354 (2020).
35. Petit, L. *et al.* High-fidelity two-qubit gates in silicon above one Kelvin. *arXiv preprint arXiv:2007.09034* (2020).
36. Landig, A. J. *et al.* Coherent spin-photon coupling using a resonant exchange qubit. *Nature* **560**, 179–184 (2018).
37. Mi, X. *et al.* A coherent spin-photon interface in silicon. *Nature* **555**, 599–603 (2018).

## BIBLIOGRAPHY

38. Nadj-Perge, S., Frolov, S., Bakkers, E. & Kouwenhoven, L. P. Spin-orbit qubit in a semiconductor nanowire. *Nature* **468**, 1084–1087 (2010).
39. Nowack, K. C., Koppens, F., Nazarov, Y. V. & Vandersypen, L. Coherent control of a single electron spin with electric fields. *Science* **318**, 1430–1433 (2007).
40. Gmitra, M. & Fabian, J. Proximity effects in bilayer graphene on monolayer WSe<sub>2</sub>: field-effect spin valley locking, spin-orbit valve, and spin transistor. *Physical review letters* **119**, 146401 (2017).
41. Gmitra, M., Kochan, D., Högl, P. & Fabian, J. Trivial and inverted Dirac bands and the emergence of quantum spin Hall states in graphene on transition-metal dichalcogenides. *Physical Review B* **93**, 155104 (2016).
42. Gmitra, M., Konschuh, S., Ertler, C., Ambrosch-Draxl, C. & Fabian, J. Band-structure topologies of graphene: Spin-orbit coupling effects from first principles. *Physical Review B* **80**, 235431 (2009).
43. Güttinger, J. *et al.* Electron-Hole Crossover in Graphene Quantum Dots. *Phys. Rev. Lett.* **103**, 046810. <https://link.aps.org/doi/10.1103/PhysRevLett.103.046810> (4 July 2009).
44. Laturia, A., Van de Put, M. L. & Vandenberghe, W. G. Dielectric properties of hexagonal boron nitride and transition metal dichalcogenides: from monolayer to bulk. *npj 2D Materials and Applications* **2**, 1–7 (2018).
45. Giannozzi, P. *et al.* QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *Journal of physics: Condensed matter* **21**, 395502 (2009).
46. Hafner, J. Ab-initio simulations of materials using VASP: Density-functional theory and beyond. *Journal of computational chemistry* **29**, 2044–2078 (2008).
47. Mostofi, A. A. *et al.* wannier90: A tool for obtaining maximally-localised Wannier functions. *Computer physics communications* **178**, 685–699 (2008).
48. Marzari, N., Mostofi, A. A., Yates, J. R., Souza, I. & Vanderbilt, D. Maximally localized Wannier functions: Theory and applications. *Reviews of Modern Physics* **84**, 1419 (2012).
49. Lado, J. L. & Fernández-Rossier, J. Landau levels in 2D materials using Wannier Hamiltonians obtained by first principles. *2D Materials* **3**, 035023. <https://dx.doi.org/10.1088/2053-1583/3/3/035023> (Sept. 2016).
50. Datta, S. Nanoscale device modeling: the Green's function method. *Superlattices and microstructures* **28**, 253–278 (2000).
51. Marzari, N. Realistic modeling of nanostructures using density functional theory. *MRS bulletin* **31**, 681–687 (2006).
52. Dr. P.J.H. Denteneer (Leiden University). *Second Quantization: Lecture notes for quantum theory* <https://homes.lorentz.leidenuniv.nl/~pjhdent/SecQuant08.pdf>.
53. Shevchenko, S. N., Ashhab, S. & Nori, F. Landau–zener–stückelberg interferometry. *Physics Reports* **492**, 1–30 (2010).

## BIBLIOGRAPHY

54. Ota, T., Hitachi, K. & Muraki, K. Landau-Zener-Stückelberg interference in coherent charge oscillations of a one-electron double quantum dot. *Scientific Reports* **8**, 1–8 (2018).
55. Cao, G. *et al.* Ultrafast universal quantum control of a quantum-dot charge qubit using Landau-Zener-Stückelberg interference. *Nature Communications* **4**, 1401 (2013).
56. Golovach, V. N., Borhani, M. & Loss, D. Electric-dipole-induced spin resonance in quantum dots. *Physical Review B* **74**. <https://doi.org/10.1103%2Fphysrevb.74.165319> (Oct. 2006).
57. Seedhouse, A. E. *et al.* Pauli blockade in silicon quantum dots with spin-orbit control. *PRX Quantum* **2**, 010303 (2021).