All optical programmable logic array (PLA)

Dawit Hiluf

Ben-Gurion University of Negev department of Physical Chemistry, Be’er-Sheva 84105, Israel
Physics Department, Mekelle University, P.O.Box 231, Mekelle, Ethiopia.
E-mail: dawit@post.bgu.ac.il

Abstract. A programmable logic array (PLA) is an integrated circuit (IC) logic device that can be reconfigured to implement various kinds of combinational logic circuits. The device has a number of AND and OR gates which are linked together to give output or further combined with more gates or logic circuits. This work presents the realization of PLAs via the physics of a three level system interacting with light. A programmable logic array is designed such that a number of different logical functions can be combined as a sum-of-product or product-of-sum form. We present an all optical PLAs with the aid of laser light and observables of quantum systems, where encoded information can be considered as memory chip. The dynamics of the physical system is investigated using Lie algebra approach.

1. Introduction

The recent success of the computing industry coupled with limitations put forth by Moore’s law has opened to a search of a paradigm shift in either the way we manufacture chips, transistors and/or in the way we do computation. This range of research routes include quantum computing [1], molecular computing [2, 3, 4], ternary valued logic [5, 6, 7], parallel computation [8, 9], and multivariable logic beyond binary [10, 11].

A programmable logic array (PLA) is a type of reconfigurable logic device that can be tailored to implement the desired combinational logic circuits. The device has a number of AND and OR gates which are linked together to produce output. PLAs are advantageous as they are compact thereby, consequently, doesn’t take much space in designing complex circuits, this is more so pronounced in logic circuit such as feedback and control where one takes into account several variables in search of efficient logic design and implementation.

Nowadays PLAs are readily obtainable from the market and reconfigure (i.e ”program”) them to provide desired Boolean functions. The most commonly used method of ‘programming’ places fuses between the inputs to a gate and the gate itself. In this case by a fuse we meant a breakable, by applying high current, electrical connection. By analyzing the desired output’s Boolean expression, one readily identifies which fuses to blow and which to be left, thereby clearing connections that are not required to implement the Boolean function. It is this act of disconnection that we refer as programming, wherein all the connections are available before reconfiguring.

1 Present address: Department of Chemistry, Sungkyunkwan University, Suwon 440-746, Korea
2. The system

The physical system we consider is a very well studied, adiabatic population transfer in solids, specifically speaking on Rare-earth-metal ions doped in solids. Recently the authors of [12, 13, 14] have demonstrated the implementation of population transfer by stimulated Raman adiabatic passage (STIRAP) in Praseodymium ions (Pr$^{3+}$), doped into a Y$_2$SiO$_5$ crystal. Rare-earth ions doped into an inorganic crystal exhibit optical transitions that do not occur in free ions. These are due to the interaction of the ions with the crystal field. The Energy level scheme in Pr$^{3+}$:Y$_2$SiO$_5$, hereafter (Pr:YSO) is shown in Fig.(1). The optical transitions in Pr:YSO exhibit very narrow homogenous line widths width in the order of a few kHz. However due to crystal impurities, different ions see slightly different crystal fields and the optical line is inhomogeneously broadened by several GHz. Due to this inhomogeneous broadening, a laser field with single frequency will always address several ions in the inhomogeneously broadened line.

The benefit of using Rare-earth materials is that they exhibit sharp optical transitions and long decoherence times. This in turn facilitates the implementation of coherent adiabatic processes. It is worth noting that all the three ground states can be coupled to all three excited states, and altogether, nine transitions are possible. The hyperfine states in Pr:YSO are spaced between 4.6 MHz and 17.3 MHz from each other. The system we consider is a three level system shown in Fig.(1) in a Λ scheme quantum system of states \(|0\rangle, |1\rangle, \& |2\rangle\) with energies $\hbar \omega_0$, $\hbar \omega_1$, and $\hbar \omega_2$ respectively. These three levels of the system are coupled by a two lasers. A pump pulse drives the transition between states \(|0\rangle\) and \(|1\rangle\). A second pulse, the Stokes pulse, drives the transition between \(|1\rangle\) and \(|2\rangle\). That is, we have transitions between states \(|0\rangle\) and \(|1\rangle\), and between states \(|1\rangle\) and \(|2\rangle\), but no transition between states \(|0\rangle\) & \(|2\rangle\), it is dipole forbidden. [15, 16] Stimulated Raman adiabatic passage (STIRAP) is an adiabatic process which provides complete coherent transfer in a Λ-type quantum system. By preparing the system to be initially on the ground state \(|0\rangle\), one can transfer 100% of the population to state \(|2\rangle\) [15]. To do so we couple the states, via the pulses, counter intuitively, that is we first drive the states \(|1\rangle\) and \(|2\rangle\) using the Stokes pulse, and later we couple states \(|0\rangle\) & \(|1\rangle\), using pump laser. STIRAP is robust in that it is insensitive to the decaying intermediate state \(|1\rangle\). Our purpose here is to go beyond STIRAP.

3. The Pulse Sequence

Fractional STIRAP (FSTIRAP) is variant of STIRAP that can be used to create a superposition of states [15, 17]. We aim to create coherent superposition between two hyperfine levels, instead

![Figure 1. Three level Λ scheme, with coupling lasers. Blue (Ω$_P$) pump laser, and red (Ω$_S$) Stokes laser, along with the corresponding energy levels and the laser frequencies. Detuning $\hbar \Delta_{p/s} = \hbar \omega_{jk} - \hbar \omega_{p/s}$, where $jk = 01, 12$. We consider the case $\Delta_p = \Delta_s = \Delta$.](image-url)
of transferring 100% population between the states. To do so we have to interrupt the STIRAP evolution. In FSTIRAP scheme, like that of STIRAP, the Stokes pulse precede the Pump pulse, but the two pulses vanishes smoothly and simultaneously. If, for example, all the population is initially prepared to be in the ground state: we have Stokes pulse applied when \( t \to -\infty \) but the pump pulse is not yet applied therefore \( \frac{\Omega_P(-\infty)}{\Omega_S(-\infty)} \to 0 \). After a time, i.e when \( t \to +\infty \), we interrupt the pulses in such a manner as to get a constant from their ratio. The condition can mathematically be put as Eq.(1)

\[
\lim_{t \to -\infty} \frac{\Omega_P(t)}{\Omega_S(t)} = 0, \quad \lim_{t \to +\infty} \frac{\Omega_P(t)}{\Omega_S(t)} = \tan \alpha,
\]

where \( \alpha \) is a constant mixing angle \( 0 \leq \alpha \leq \frac{\pi}{2} \). A convenient realisation of the condition Eq.(1), using three pulses, a pump pulse and two Stokes pulses (one with the same time dependence as the pump and another coming earlier) is provided by Vitanov et al. to be [17]

\[
\Omega_P(t) = \Omega_0 \sin \alpha e^{-\frac{(t-\tau)^2}{\sigma^2}}
\]

\[
\Omega_S(t) = \Omega_0 e^{-\frac{(t+\tau)^2}{\sigma^2}} + \Omega_0 \cos \alpha e^{-\frac{(t-\tau)^2}{\sigma^2}}
\]

where \( \tau, \sigma, \) & \( \Omega_0 \) are respectively defined to be the centre, width, and the peak of the pump pulse. Note that, of the two Stokes pulses one has same time dependence as the pump and another coming earlier. Moreover \( \Omega_0 \) is also the peak of each of these two Stokes pulses.

![Figure 2](image.png)

**Figure 2.** (a) Pulse Profile, blue \( \Omega_P \) Pump laser, and red \( \Omega_S \) Stokes laser. (b) Mixing angle \( \theta \), defined by the ratio of the pulse profiles, i.e. \( \tan \theta = \frac{\Omega_P(t)}{\Omega_S(t)} \)

### 3.1. Hamiltonian of the system

For quantum systems with distinct states, external perturbations change the state of the system. The three states of the system, i.e. \(|0\rangle, |1\rangle, \) & \(|2\rangle \) respectively are the eigenstates of the the unperturbed part of the Hamiltonian \( \hat{H}_0 \), and \( \hat{H}_I \) is the part of the Hamiltonian representing the interaction of the system with the laser field. Therefore, the total Hamiltonian can be expressed as the sum of the two parts [16].

\[
\hat{H} = \hat{H}_0 + \hat{H}_I
\]
where the unperturbed part is, $\hat{H}_0 = \hbar \omega_0 |0\rangle \langle 0| + \hbar \omega_1 |1\rangle \langle 1| + \hbar \omega_2 |2\rangle \langle 2|$, and the interaction part of the Hamiltonian is $\hat{H}_I = V_P + V_S$, where $V_j = -\mu_j \cdot E_j \cos \omega_t$ with $j = P, S$. With the aid of which and considering a rotating frame, i.e. $e^{i \omega_0 t}|0\rangle \langle 0| + e^{i (\omega_1 - \Delta) t}|1\rangle \langle 1| + e^{i \omega_2 t}|2\rangle \langle 2|$, the total Hamiltonian in the interaction picture and in RWA can be rewritten as

$$\hat{H} = \frac{\hbar}{2} \begin{pmatrix} 0 & \Omega_P(t) & 0 \\ \Omega_P(t) & 2\Delta_P & \Omega_S(t) \\ 0 & \Omega_S(t) & 2(\Delta_P - \Delta_S) \end{pmatrix}$$  \hspace{1cm} (4)

Although the presence of the single detuning does not prevent population transfer, it is important that the two-photon resonance condition apply, i.e. $\hbar (\Delta_P - \Delta_S) = 0$. Because this condition is crucial for the creation of the dark state [15, 16].

3.2. Adiabaticity

The system we consider is shown Fig.(1). The population transfer mechanics, FSTIRAP, is easily understood in Hilbert space whose coordinate basis vectors are instantaneous eigenstates of the time varying Hamiltonian of Eq.(4). In section (3.1) we have obtained the Hamiltonian of a three level system whose eigenstates are the energy level of the system, these states were labeled as $|0\rangle$, $|1\rangle$, and $|2\rangle$, we refer to such basis the diabatic basis. But it turns out, it is helpful to transform into different basis to get a complete understanding of the system at hand. To this end it will be insightful if we work in the adiabatic basis. When the two-photon resonance condition $\hbar (\Delta_P - \Delta_S) = 0$ fulfilled, Eq.(4) reduces to

$$\hat{H}(t) = \frac{\hbar}{2} \begin{pmatrix} 0 & \Omega_P(t) & 0 \\ \Omega_P(t) & 2\Delta & \Omega_S(t) \\ 0 & \Omega_S(t) & 0 \end{pmatrix}$$  \hspace{1cm} (5)

It is worth recalling that for the three level system depicted in Fig.(1), the transition between levels $|0\rangle$ and $|2\rangle$ is dipole forbidden, level $|1\rangle$ can be off-resonance by certain detuning. Introducing an adiabatic $\{|a_+, a_0, a_-\}$ and diabatic basis $\{|0\rangle, |1\rangle, |2\rangle\}$ (also known as the bare states) respectively, we can make a unitary transformation $|\Phi\rangle = U|\Psi\rangle$ [18]. Following which we make the expansion of the wave function in terms of the time dependent probability amplitude corresponding to each basis as:

$$|\Psi\rangle = \sum_{j=0}^{2} C_j(t) |j\rangle, \quad |\Phi\rangle = \sum_{k=\pm, 0} a_k(t) |a_k\rangle$$  \hspace{1cm} (6)

The unitary matrix $U$ diagonalises the Hamiltonian Eq.(5) into $D = U^\dagger \hat{H} U$, adiabatic Hamiltonian, where the transformation matrix $U$ is given by [18]

$$U = \begin{pmatrix} \sin \phi & \sin \theta & \cos \phi \\ \cos \phi & \cos \theta & -\sin \phi \\ \cos \theta & \sin \phi & \cos \phi \end{pmatrix}$$  \hspace{1cm} (7)

After diagonalising the Hamiltonian Eq.(5) we find that the new basis vectors $\{|a_+, a_0, a_-\}$ corresponds to the eigenvalues $\{\lambda_+, \lambda_0, \lambda_-\}$ respectively where

$$\lambda_+ = \frac{1}{2}(\Delta + \sqrt{\Delta^2 + \Omega^2}) = \Omega \cot \phi,$$  \hspace{1cm} (8a)  

$$\lambda_0 = 0,$$  \hspace{1cm} (8b)  

$$\lambda_- = \frac{1}{2}(\Delta - \sqrt{\Delta^2 + \Omega^2}) = -\Omega \tan \phi$$  \hspace{1cm} (8c)
with $\Omega^2 = \Omega_p^2 + \Omega_s^2$ where the mixing angles are defined to be

$$\tan \theta = \frac{\Omega_p(t)}{\Omega_s(t)},$$

$$\tan 2\phi = \frac{\Omega(t)}{\Delta}.$$  \hspace{1cm} (9a, 9b)

The Hamiltonian matrix element for non-adiabatic coupling between state $|a_0\rangle$ and either one of $|a_+\rangle$ or $|a_-\rangle$ is given by $\langle a_\pm | \dot{a}_0 \rangle$. Non-adiabatic coupling is small if this matrix element is small compared to the field induced splitting $|\lambda_\pm - \lambda_0|$ of the energies of these states, that is

$$|\langle a_\pm | \dot{a}_0 \rangle| \ll |\lambda_\pm - \lambda_0|$$ \hspace{1cm} (10)

Using Eq.(7) and the transformation $|\Phi\rangle = U|\Psi\rangle$ one readily obtain

$$|a_+\rangle = \sin \theta \sin \phi |0\rangle + \cos \phi |1\rangle + \cos \theta \sin \phi |2\rangle$$ \hspace{1cm} (11a)

$$|a_0\rangle = \cos \theta |0\rangle - \sin \theta |2\rangle$$ \hspace{1cm} (11b)

$$|a_-\rangle = \sin \theta \cos \phi |0\rangle - \sin \phi |1\rangle + \cos \theta \cos \phi |2\rangle$$ \hspace{1cm} (11c)

Notice that the adiabatic state $|a_0\rangle$ is independent of state $|1\rangle$, which is the leaky state. Therefore following the adiabatic state $|a_0\rangle$ we can achieve the complete population transfer from the ground state $|0\rangle$ to the target state $|2\rangle$. For appreciable non-adiabatic transition to occur, the following conditions has to be met

$$|\dot{\theta}(t)| \geq \frac{1}{2} \Omega(t)$$ \hspace{1cm} (12a)

$$|\dot{\theta}(t)| \geq \frac{1}{\sigma}$$ \hspace{1cm} (12b)

The first condition means the adiabatic condition has to be violated, that is the non-adiabatic coupling has to be comparable or larger than the eigenvalues splitting [17]. To summarize, so far we have described our physical system which is a Rare-Earth metal ion doped in Inorganic crystal, specifically Praseodymium ions (Pr$^{3+}$), doped into a Y$_2$SiO$_5$ crystal. The ion’s electronic states are split into so called "crystal field states"; the crystal field states are split into hyperfine states. The hyperfine interaction consists of hyperfine interaction from the ions and magnetic hyperfine interaction with the crystal field. The hyperfine interaction splits the crystal field states further into doubly degenerate levels. We have identified, based on the experimental measurement, three level lambda system out of the nine possible transitions. Working in the adiabatic basis has revealed that we have two important controlling parameters: time-dependent mixing angles $\theta$ and $\phi$. With these angles we can manipulate the population transfer amongst the states of the atomic system. To avoid, for example, complete transfer of population between the initial and the target state, as one would have in STIRAP scheme, we demanded the ratio of our pulse profiles (of pump and Stokes) to be constant after interaction. This demand means our pulses after long time decay smoothly and simultaneously. Such scheme is referred to as Fractional STIRAP (FSP); with such pulse profile we managed to create superposition of the two hyperfine ground states $|0\rangle$ and $|2\rangle$. Furthermore we have obtained the Hamiltonian under two-photon resonance, RWA (Rotating Wave Approximation), in the interaction picture.

4. Dynamics of the system

The Quantum dynamic are calculated by the density matrix formalism [19]. In this method a statistical average over the assembly of the expectation values for the individual ions is included.
This in turn yields information about observable macroscopic quantities. The formalism is advantageous in that it greatly simplifies the calculations since it allows one to handle a large number of dynamical variables in a systematic way. The physical significance of the individual density matrix elements depends on the representation with which the matrix is calculated. In a representation where \( H_0 \) is diagonal, the elements of the density matrix are defined as \( \rho_{jk} = \langle \psi_j | \hat{\rho} | \psi_k \rangle \) where \( \hat{\rho} \) is the density operator.

In this representation, the diagonal elements \( \rho_{jj} \) are the ensemble averaged occupation probabilities for the state \( | \psi_j \rangle \). The off-diagonal elements \( \rho_{jk} \) give information about the ensemble averaged phases of the quantum states, these matrix elements are commonly known as coherences. The matrix \( \rho \) is Hermitian. The motion of the populations \( \rho_{jj} \) is coupled to the coherences \( \rho_{jk} \), therefore we should deal the density matrix as a whole. As our density matrix \( \rho \) is a \( 3 \times 3 \) Hermitian matrix, it contains nine distinct real variables. Because of conservation of probability, only eight of these variables are independent.

**Lie Algebra approach**

The dynamical evolution of an \( N \)-level atomic system can be expressed in terms of density matrix \( \hat{\rho} \) which satisfies the Liouville equation [20]

\[
\frac{i\hbar}{\partial t} \hat{\rho} = [\hat{H}, \hat{\rho}]
\]  

(13)

Because \( \langle \hat{P}_{mn} (t) \rangle = Tr \left( \hat{P}_{mn} \hat{\rho} (t) \right) = \rho_{nm} (t) \) the dynamics can be equally explored by studying \( \hat{P}_{nn} \), where \( \hat{P}_{nn} = |m\rangle \langle n| \). In this section we will obtain the equation of motion for the eight dimensional coherence vector, we do so by using the Alhassid-Levine formalism [21]. One can also obtain the equation of motions using the Hioe-Eberly [22, 23, 24] formalism; In Hioe-Eberly approach, the state of a three-level system is represented by pseudo-spin vector \( \vec{S} \) having 8 real components, 9 if we do not impose normalization. This description provides an elegant geometrical framework. The Hamiltonian under RWA, in the interaction picture, and two-photon resonance is expressed as

\[
\hat{H} = \hbar \left( \begin{array}{ccc} 0 & \alpha (t) & 0 \\ \alpha (t) & \Delta & \beta (t) \\ 0 & \beta (t) & 0 \end{array} \right)
\]  

(14)

where the half Rabi frequency is defined by \( \alpha (t) = \frac{1}{\hbar} \omega_P (t) \) and \( \beta (t) = \frac{1}{\hbar} \omega_S (t) \).

The \( \hat{G}_\alpha \)s are our generators, which will be expressed in terms of the operators \( |\alpha\rangle \langle \beta| \). Using this generators we will expand our Hamiltonian and density matrix as

\[
\hat{\rho} (t) = \frac{\hat{I}}{3} + \frac{1}{2} \sum_{\alpha=0}^{8} \langle G_\alpha \rangle \hat{G}_\alpha
\]  

(15a)

\[
\hat{H} (t) = \frac{\hbar}{2} \sum_{\alpha=0}^{8} h_\alpha (t) \hat{G}_\alpha
\]  

(15b)

with traceless generators \( \hat{G}_\alpha \) and \( \hat{I} \) is the identity operator. We stress here that Eq.(15a) is always true, but Eq.(15b) is a special case that is valid for our problem.

The coefficients \( \langle G_\alpha \rangle \) and \( h_\alpha (t) \) are given by

\[
\langle G_\alpha \rangle = Tr \left( \hat{\rho} (t) \hat{G}_\alpha \right)
\]  

(16a)

\[
\hbar h_\alpha (t) = Tr \left( \hat{H} (t) \hat{G}_\alpha \right)
\]  

(16b)
Furthermore the generators $\hat{G}_\alpha$ have the following properties

$$Tr\left(\hat{G}_\alpha \hat{G}_\beta\right) = 2\delta_{\alpha\beta} \quad (17a)$$

$$\left[\hat{G}_\alpha, \hat{G}_\beta\right] = 2i f_{\alpha\beta\gamma} \hat{G}_\gamma \quad (17b)$$

where the repeated indexes are summed up from 1 to 8 and $f_{\alpha\beta\gamma}$ is the completely antisymmetric structure constants of the SU(3) group is given in Table A1. The Identity operator is given by $|0\rangle\langle 0| + |1\rangle\langle 1| + |2\rangle\langle 2|$. We note that Eq.\,(17a) tells us that the trace of the product of the generators is orthogonal, and Eq.\,(17b) is the closure property.

The generators $\hat{G}$ in turn are closed under the commutation relation Eq.\,(17)

$$\left[\hat{H}, \hat{G}_\alpha\right] = i\hbar \sum_{\beta=0}^{8} \hat{G}_\beta g_{\beta\alpha} \quad (18)$$

where the $g_{\beta\alpha}$ are (possibly complex) numerical coefficients.

In their pioneering work on the information theoretic approach, Alhassid and Levine starting from Eq.\,(19), the equation of motion for observable $\hat{G}$ in the Heisenberg picture, replaced Eq.\,(13) by a set of coupled equations for the mean value of the observables as given by Eq.\,(20)

$$i\hbar \frac{d\hat{G}}{dt} = \left[\hat{G}, \hat{H}\right] \quad (19)$$

In doing so they demand that the commutation between the Hamiltonian and the generators be closed under the Lie algebra as Eq.\,(18) \cite{21, 25}. The closure means that the commutation between the Hamiltonian and the generators is a linear combination of the generators \cite{26}. Moreover the Hamiltonian is expanded in terms of the generators $\hat{G}$ via Eq.\,(15b). The equation of motion for the expectation values of the observables $\langle G_\alpha \rangle$, takes the form

$$\frac{d\langle G_\alpha \rangle}{dt} = -\sum_{\beta} \langle G_\beta \rangle g_{\beta\alpha} \quad (20)$$

Equation\,(20) is a set of coupled linear differential equations that completely determine the time evolution of the expectation values $\langle G_\alpha \rangle$, provided that one knows the corresponding initial values.

So as to exploit the SU(3) Lie algebra let us define three groups of generators as

$$\hat{s}_m = |j\rangle\langle k| + |k\rangle\langle j|$$
$$\hat{s}_n = -i (|j\rangle\langle k| - |k\rangle\langle j|)$$
$$\hat{s}_p = -\sqrt{\frac{2}{l(l+1)}} \left( |j\rangle\langle l| - l|l+1\rangle\langle l+1| \right) \quad (21)$$

where $0 \leq j \leq k \leq 2$, $0 \leq l \leq 2$, $m = 1, 2, 3$, $n = 4, 5, 6$, and $p = 7, 8$.

Using Eqs.\,(14),\,(16b), and the generators Eq.\,(21) we note that one can readily express the Hamiltonian (to within an addition of multiple of an Identity matrix) in terms of the generators as

$$\hat{H}(t) = \frac{\hbar}{2} \left[ \Omega_P \hat{s}_4 + \Omega_S \hat{s}_2 + \Delta \hat{s}_7 - \frac{\Delta}{\sqrt{3}} \hat{s}_8 \right] + \text{const} \quad (22)$$
where \( \text{const} = \frac{\hbar}{2} \left( \frac{2}{3} \sum_{j=0}^{2} \omega_j \right) \hat{s}_0 \)-commutes with the Hamiltonian– where \( \hbar \omega_j \) is energy of level \( j \). Making use of the commutation relation between the Hamiltonian Eq.(22) and the generators Eq.(21), and then equate both sides of Eq.(15b) we find the complete surviving elements \( g_{\alpha\beta} \), in a matrix form is

\[
g = \begin{pmatrix}
0 & 0 & 0 & \Delta & 0 & \beta & 0 & 0 \\
0 & 0 & 0 & -\Delta & -\alpha & 0 & 0 & 0 \\
0 & 0 & 0 & \beta & -\alpha & 0 & 0 & 0 \\
-\Delta & 0 & -\beta & 0 & 0 & 0 & 2\alpha & 0 \\
0 & \Delta & \alpha & 0 & 0 & 0 & -\beta & \sqrt{3}\beta \\
-\beta & \alpha & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -2\alpha & \beta & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\sqrt{3}\beta & 0 & 0 & 0 
\end{pmatrix}
\]  

(23)

where the half-Rabi frequency is defined by \( \alpha(t) = \frac{1}{2} \Omega_p \), \( \beta(t) = \frac{1}{2} \Omega_s \), and note that \( g \) is antisymmetric.

We now solve our equation of motion given by Eq.(20) numerically. It is worth pointing out that the equation of motion is exact. Recall we have a three level lambda system whose population is initially prepared to be on the ground state \( |0 \rangle \). We then apply FSTIRAP pulse profile and follow the evolution of the observable vector. By arranging the expectation values of the generators we can form a vector of the observables as \( \vec{S} = (\langle s_1 \rangle, \langle s_2 \rangle, \ldots, \langle s_8 \rangle)^T \) the equation of motion Eq.(20), can now be rewritten as

\[
\frac{d}{dt} \vec{S} = g \vec{S}
\]

(24)

We first solve Eq.(24) numerically for a very small value of detuning and the FSTIRAP pulse profile; we recall we prepare the system to be on the ground state \( |0 \rangle \). Of course it goes without saying that the results will vary when we take different initial state, if we take a different pulse profile, STIRAP case for example. The consequence of change in the initial state, change in the input parameters leading to different output, in digital logic term, basically means we have a Finite State Machine (FSM) where the output of the system in not only dependent on the input applied but also is dependent on the initial state of the system [27].

5. Logic Machine

The discussion in this paper focuses on isolated systems, however, it has to be pointed out that the inclusion of noise destroys most of the observables (coherences) and hence we refrain from making logic out of such observables. Yet we can design our pulse profile to yield only the desired observables that withstand the effect of noise on the system, long enough to do computation. These observables are the population on each state along with the coherence between the two lower levels (i.e. between states \( |0 \rangle \) and \( |2 \rangle \)). Using the results we obtain by such scheme we will try to show here the implementation of logic machines: a programmable logic array (PLA) it can be depicted as parallel outputted logic (endowed with Memory).

Before proceeding it is worth differentiating a ROM (Read Only Memory) and PLA (Programmable Logic Array) for the sake of clarity. Both of these used to implement logic functions, and they both use the ‘Sum of Products’ logic configuration, which consists of a primary array of AND gates and a secondary array of OR gates. The OR function (Sum) is applied to outputs of AND (product) arrays. ROM is made of an AND gates array and OR gates array. AND array provides all the combinations of inputs, and OR array is used to select the necessary combinations. Therefore, AND array is always fixed. For example, in a three input system, AND array produces all the combinations (product terms) of \( ABC, AB\overset{\Omega}{C}, AB\overset{\bar{C}}{C}, AB\overset{\bar{C}}{C}, \bar{A}BC, \bar{A}B\overset{\bar{C}}{C}, \bar{A}B\overset{\bar{C}}{C} \) where \( \bar{X} \) implies the complement (NOT) of
X. Then an OR gate can be used to select the necessary product terms to implement the given logic function. Any logic function of $A, B, C$ can be implemented using those product terms. For example

$$f(A, B, C) = \bar{A}B + \bar{B}C$$

$$= \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}B\bar{C} + \bar{A}B\bar{C}$$

(25)

In like manner an array of OR gates can implement an array of logic functions. Therefore ROM is used to store programs. Programming the ROM means configuring those OR array by selecting the necessary products. PLA too is made of two OR and AND arrays, but both the arrays are configurable unlike in ROM. This also provides a 'Sum of Products' term, but in a different way. Since terms for AND gates are also possible, it can give more product terms like $AB, \bar{C}, C$, etc. Therefore it is much easier to implement logic functions compared to ROM.

For example, $\bar{A}B + \bar{B}C$ can be directly implemented by selecting $\bar{A}, \bar{B}$ for one AND gate, $B, \bar{C}$ for another AND gate and making outputs of those AND gates to inputs of an OR gate.

A PLA provides a bank of AND gates, followed by a bank of OR gates, and optionally a flip-flop per output. It also will have an array of programmable "fuses", the "fuses" are short-circuits, so each input of each AND is connectable to any of the chip inputs or outputs, or their complements. By programming we mean fusing connections. When we program the PLA, the programmer blows the fuses to leave only the connections we want. Then, the OR array similarly can be programmed to OR together various combinations of the AND outputs. The fuse technology incurs minimal delay, so propagation delay from input pin to output pin can be as small as two gate delays or up to four, depending on how many inverted signals are needed, which automatically beats the usual ROM topology for speed. In a PLA the "OR array" is fixed and the "AND array" is programmable. Each output must be formed from a restricted number of minterms but those minterms can each cover multiple input combinations.

On top of this, a read-only memory (ROM) is a logic circuit that can generate all of the possible minterms of its inputs. With the same numbers of inputs and outputs, a ROM is capable of more general logic, in that we can burn any truth table we like into the ROM. However, as a logic device, the ROM is not optimal. If we attempt to burn several independent little functions into a ROM, we will find we have a lot of 'don’t care' cases that must nonetheless be programmed into the ROM to get the desired output. The PLA is better for logic because we don’t have to deal with the don’t-cares, but the trade-off is that we can’t program completely arbitrary functions of the bits. However, they are nonetheless sufficiently flexible to be useful for many common ‘glue-logic’ tasks, and they are generally faster at producing output.

What we aim to show here is how the system yields logic in parallel and serves as PLA, for which we will obtain the corresponding Boolean equations. In what follows we describe how we implement the PLA. To this end we apply a sequence of three pulses separated by some waiting time $t$. This means that after the system interacts with the first pulse we have some time window $t_1$ before applying the second pulse. During this time the observable vector propagates freely. Then we apply the second pulse and wait another time $t_2$ before applying the last pulse. Each of the pulses can take either the STIRAP or FSTIRAP pulse profile, here after SP and FSP respectively. We can follow the evolution of the observables before, during, and after each of the pulses are applied, but what we take as an output is the observable after the third pulse is applied. We assign logic 0, when we apply the SP pulse and logic 1 for FSP pulse profile. Because we have three inputs (the triple pulse sequence), and because each can be either 0 or 1, we have $2^3 = 8$ possible combination of inputs as seen in the truth table (1). The three pulse sequences are labeled as input variable $\text{ABC}$, where $A$ stands for the first pulse and $C$ is the last pulse in the sequence, $B$ being the intermediate pulse. We use logic assignment for the populations to be logic 1 if we have reading of populations $\geq 0.2$ else it is 0; and for the coherences it is logic 1 if the absolute value of the coherences is $\geq 0.1$, else 0. With these logic
assignments the possible input combinations, along with their outputs is shown in Table (1).

![Pulse Sequence](image1)

![Observables](image2)

**Figure 3.** (a) Plots of observables for pulse sequence 001 (see text). (b) Programmable logic array (PLA). In this case programming means fusing connections.

**Table 1.** Truth table for the triple input pulses labeled as ABC, the outputs O1, O2, O3 are populations of state $|k\rangle$, $k = 0, 1, 2$, and O4, O5 are the real and imaginary part of the coherence.

| ABC | O1 | O2 | O3 | O4 | O5 |
|-----|----|----|----|----|----|
| 000 | 0  | 0  | 1  | 0  | 0  |
| 001 | 1  | 0  | 1  | 1  | 0  |
| 010 | 1  | 1  | 1  | 1  | 1  |
| 011 | 1  | 0  | 0  | 0  | 1  |
| 100 | 1  | 0  | 1  | 1  | 0  |
| 101 | 1  | 0  | 1  | 0  | 1  |
| 110 | 1  | 0  | 0  | 1  | 0  |
| 111 | 1  | 0  | 1  | 1  | 1  |

We show in Fig. (3a), the numerical result of the observable vector, for the input sequence 001 (SP-SP-FSP), which is also shown in the table via red ink–second row. We can see from the truth Table (1) we have obtained five logic outputs simultaneously read after the system interacts with the last (i.e. third) pulse. This means we can store information in each of these outputs. Moreover, if we have many, say N, independent three level systems we can have a massive parallel logic memory, $5^N$, where we can store information. Each of these three level systems could be arranged to have different initial condition, set on different detuning, and apply
different pulse sequence to store a variety of information. From the truth table we can find an expression representing each of the five output logic functions. Such logic function is derived from the truth table by OR-ing (logic OR), logic OR is read as A OR B, is expressed as A+B, all the terms that correspond to those input lines for which the output assumes the logic value 1. Each term is an AND (logic AND), logic AND is read as A AND B, is expressed as AB, of the input variables. The input variable appears in uncomplemented form in the product (i.e. logic AND) if it has logic value 1 in the corresponding input line, and it appears in complemented form if it has logic value 0. For example, the output logic function O2 has only one logic value 1 (row 3 in Table (1)) corresponding to input line 010. Hence the product term (i.e. logic AND) that corresponds to this input line is $\overline{AB}\overline{C}$. Such product term contains each of the 3 input variables in either complemented or uncomplemented form, and is called minterm. If, however, we have logic value 1, for more than one input line combinations; then we OR all of such minterms to find the logic function of the output. An expression of this type is known as a canonical sum of products (SOP). Therefore using the SOP for the output logic functions shown in the truth table we get

$$O1 = \overline{A}BC + \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

$$O1 = \overline{A}BC + (A \oplus B) + AB$$

(26)

where we have used $(X + \overline{X}) = 1$, $A \oplus B = \overline{A}B + A\overline{B}$ for simplification, where $X$ stands $A$, $B$, and $C$, also $\oplus$ represent logic Exclusive OR (XOR). In like manner we obtain the logic function of the other outputs

$$O2 = \overline{A}BC$$

$$O3 = \overline{B} + B(A \odot C)$$

$$O4 = \overline{A}(B \oplus C) + A\overline{C} + AB$$

$$O5 = \overline{AB} + AC$$

(27)

where $(B \oplus C) = \overline{B}C + B\overline{C}$ and $(A \odot C) = (\overline{A}\overline{C} + AC) = (A \oplus C)$ and also $\odot$ represent logic Exclusive NOR (XNOR). Therefore we have here, expressed by Eqs.(26) and (27), logic output functions in parallel. As the laser pulse is very long, and as the observables are long lived, these output functions serve as memory. We can form an Integrated Circuit (IC) that implements these five output functions as shown in Fig.(3b). It comprises of an AND plane where they form input lines and are shown by the red lines, these input lines represent both the complemented and the uncomplemented form; and an OR plane where we form the outputs functions and are shown by the blue lines. The dots, in the input lines, tell us what form of the input variable we take (i.e. complemented or uncomplemented). To obtain the minterms that is present in the logic output function, we take the product of each of the input variables down the line (see the green box in Fig.(3b), for example, it is formed by taking the product of the input variables $\overline{A}$, $B$, and $C$ i.e. $\overline{AB}\overline{C}$). To obtain the output of the logic function we OR the minterms we formed for each logic functions. After we have done that we get the following Integrated Circuit (IC) for the five output functions.

We stress that the very act of applying the pulses 'fuses' the input-output connections in our system, this in turn yields fixed input combinations (an AND array in the PLA) and outputs (an OR array of PLA) as given by Eqs.(26) and (27) thereby allowing the possibility of implementing PLA using three level dynamics.

The logic functions we get will differ if we redefine our logic assignment for the input pulses. If we now assign logic value 1 for when we use the SP pulse profile and logic 0 for FSP pulse
profile we obtain the following logic equations.

\[
O_1 = \bar{A} + AB + AB\bar{C} \\
O_2 = A\bar{B}\bar{C} \\
O_3 = \bar{A}\bar{C} + C(A \oplus B) + AB \\
O_4 = \bar{A}\bar{B} + C(A \oplus B) + AB\bar{C} \\
O_5 = \bar{A}\bar{C} + AB
\]  

(28)

We can also change threshold value we took for the purpose of the logic assignments for the observables and thereby obtaining different logic functions for the observables, the five outputs. Moreover, we can also change the initial state of the system leading to different logic functions; hence we have a dynamic memory. By dynamic we mean the memory changes whenever we change either of: the initial state of the system, the inputs, redefine input logic assignments, and change the logic assignments of the observables by re binning the threshold value.

6. Conclusion
In a nut shell, in this paper we have explored the dynamics of the physical system, optically addressed three-level system, using Alhassid-Levine scheme, where we have obtained equation of motion for the average values of the observables which we numerically solve for different cases of inputs (i.e. Pulse profile, detuning, and initial state of the system). By applying sequence of three pulses we managed to propose a Memory Logic that yields its output in parallel and serves as Programmable logic array (PLA) where by programmable we mean fusing connections. Moreover we expressed the logic function of each output in terms of the input variables (and their complemented form). The execution of logic functions in parallel and storing them has advantage of yielding storing vast data in parallel, the size and complexity of the stored data increases as we increase the physical system to N-level system. Although contact with the environment washes away the information stored, we argue that we have ample time to retrieve the stored information before it goes away.

Acknowledgments
The author thanks Prof. Raphael D. Levine of Hebrew University of Jerusalem for introducing him to Lie algebra and molecular logic. This work was supported by the European Commission Future and Emerging Technologies Proactive Project MULTI (317707).

Appendix
The generators chosen to fit the GellMann SU(3) matrices are [24]

\[
\begin{align*}
\hat{G}_1 &= \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\
\hat{G}_2 &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \\
\hat{G}_3 &= \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \\
\hat{G}_4 &= \begin{pmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\
\hat{G}_5 &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -i \\ 0 & i & 0 \end{pmatrix}, \\
\hat{G}_6 &= \begin{pmatrix} 0 & 0 & -i \\ 0 & 0 & 0 \\ i & 0 & 0 \end{pmatrix}, \\
\hat{G}_7 &= \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\
\hat{G}_8 &= \frac{1}{\sqrt{3}} \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 2 \end{pmatrix}
\end{align*}
\]  

(A.1)

where the corresponding surviving structure constants \(f_{ijk}\) are given below
Table A1. Values of the surviving structure constants

| ijk  | 147 | 135 | 126 | 432 | 465 | 736 | 752 | 368 | 258 |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| \( f_{ijk} \) | \(-1\) | \(\frac{1}{2}\) | \(\frac{1}{2}\) | \(-\frac{1}{2}\) | \(-\frac{1}{2}\) | \(-\frac{\sqrt{3}}{2}\) | \(-\frac{\sqrt{3}}{2}\) |

References

[1] Michael A Nielsen, Isaac L Chuang 2010 *Quantum Computation and Quantum Information* (Cambridge University Press)
[2] Remacle F, Levine R D 2001 *J. Chem. Phys.* 114 pp 10239–46
[3] Beil F, Hallmann T, Remacle F and Levine, R D 2011 *Phys. Rev. A* 83 p 033421
[4] Remacle F, Weinkauf R and Levine R D 2006 *J Phys Chem A* 110 pp 177-84
[5] Klein M, Rogge S, Remacle F and Levine, R D 2007 *Nano letters* 7 pp 2795–99
[6] Klein M, Mol JA, Verduijn J, Lansbergen GP, Rogge S, Levine R D and Remacle F 2010 *Appl. Phys. Lett.* 96 p 043107
[7] Mol JA, Van der Heijden J, Verduijn J, Klein M, Remacle F and Rogge, S 2011 *Appl. Phys. Lett.* 99 p 263109
[8] Fresch B, Hiluf D, Collini E, Levine R D and Remacle F *Proc. Natl. Acad. Sci. U. S. A* 110 pp 17183–88
[9] Remacle F, Levine R D 2007 *Eur. Phys. J. D* 42 pp 49–59
[10] Fresch B, Bocquel J, Hiluf D, Rogge S, Levine R D and Remacle F 2017 *ChemPhysChem* 18 pp 1790-97
[11] Fresch B, Cipolloni M, Yan T, Collini E, Levine R D and Remacle F 2015 *J. Phys. Chem. Lett.* 6 pp 1714-18
[12] Klein J, Beil F and Hallmann T 2008 *Phys. Rev. A* 78 p 033416
[13] Beil F, Klein J, Nikoghosyan G and Hallmann T 2008 *J. Phys. B* 41 p 074001
[14] Klein J, Beil F and Hallmann T 2007 *Phys. Rev. Lett.* 99 p 113003
[15] Vitanov N V, Hallmann T, Shore B W and Bergmann K 2001 *Annu. Rev. Phys. Chem* 52 pp 763–809
[16] Shore B W 2008 *Acta Phys. Slovaca* 58 pp 243–486
[17] Vitanov N V, Suominen K A and Shore B W 1999 *J. Phys. B* 32 p 4535
[18] Vitanov N V, Stenholm, S 1999 *Phys. Rev. A.* 60 pp 3820–32
[19] Fano U 1957 *Rev. Mod. Phys.* 29 pp 74–93
[20] Levine R D 2011 *Quantum mechanics of molecular rate processes* (Courier Dover Publications)
[21] Alhassid Y, Levine R D 1977 *J. Chem. Phys.* 67 pp 4321–39
[22] Hioe F T, Eberly J H 1981 *Phys. Rev. Lett.* 47 pp 838–41
[23] Hioe F T, Eberly J H 1982 *Phys. Rev. A.* 25 p 2168
[24] Hioe F T 1983 *Phys. Rev. A.* 28 p 879
[25] Alhassid Y, Levine R D 1978 *Phys. Rev. A.* 18 p 89
[26] Robert G 2006 *Lie Groups, Lie Algebras, and Some of Their Applications* (Dover Publisher)
[27] Kohavi Z, Jha N K 2010 *Switching and finite automata theory* (Cambridge University Press)