



IMPLEMENTING CONVENTIONAL LOGIC UNCONVENTIONALLY: PHOTOCHROMIC MOLECULAR POPULATIONS AS REGISTERS AND LOGIC GATES

J.C.CHAPLIN^{A,B}, N.A.RUSSELL^B, N.KRASNOGOR^{A,}*

^A *Automated Scheduling, Optimization and Planning Research Group, School of Computer Science, University of Nottingham, Jubilee Campus, Wollaton Road, Nottingham, NG8 1BB, UK*

^B *Institute of Biophysics, Imaging and Optical Science, Schools of Biology and Electrical and Electronic Engineering, University of Nottingham, Nottingham, NG7 2RD, UK*

** To whom correspondence should be addressed.*

jcc@cs.nott.ac.uk, noah.russell@nottingham.ac.uk, natalio.krasnogor@nottingham.ac.uk

Abstract

In this paper we detail experimental methods to implement registers, logic gates and logic circuits using populations of photochromic molecules exposed to sequences of light pulses. Photochromic molecules are molecules with two or more stable states that can be switched reversibly between states by illuminating with appropriate wavelengths of radiation. Registers are implemented by using the concentration of molecules in each state in a given sample to represent an integer value. The register's value can then be read using the fluorescence signal from one of the states. Logic Gates have been implemented using a register with inputs in the form of light pulses to implement 1-input/1-output and 2-input/1-output logic gates. A proof of concept logic circuit is also demonstrated; coupled with the software workflow describe the transition from a circuit design to the corresponding sequence of light pulses.

Keywords: *Photochromic Molecules, 6-Nitro-BIPS, Logic Gates, Vesicle Logic.*

1. INTRODUCTION

Computing is a wide field dealing with algorithmic processes and how we use them to perform useful operations on data. Traditionally a computer is an electronic device calculating with binary logic, but the field of Unconventional Computing challenges this assumption with new ideas and techniques. Unconventional Computing investigates new logical paradigms as well as new materials with which to build computers [Cooper et al. (2008)] [Adamatzky (2001)]. Like Conventional Computing, progress in Unconventional Computing has trended towards miniaturization to facilitate performance increases and energy efficiency. One of the most productive research areas has been Molecular Computing.

Molecular Computation is utilising the vast range of chemical reactions to be repurposed for computing information. Inputs can be chemicals or other stimuli into a reaction, and the output the produced chemicals, a change in properties, fluorescence etc. [REF][REF] More complex processes can be achieved with supramolecular assemblies, comprising multiple molecules and their combined behaviours [REF][REF] [1]. Beginning with breakthroughs on enzyme function, protein function and the structure of DNA, supramolecular chemistry advanced to the production of synthetic supramolecular constructs such as crown ethers [REF], cryptands [REF] and other host-guest complexes [REF]. These complexes can exhibit behaviour that can be leveraged for computation [REF][REF]. It is within the subject of supramolecular chemistry that molecular self assembly falls, relating to the spontaneous formation of complex structures from simpler molecular components without exterior direction. The production of vesicles is one such example, being self-assembled aggregates of phospholipids [REF]. Self assembly could be used to compute logical functions, EXAMPLES or even to produce cellular automata [REF].

One issue with molecular computing is the low speed of processes, taking (for example) 0.1ms for enzymes against nanoseconds for a CPU [REF]. Molecular scale electronics - the use of single molecules as electrical components [REF][REF][REF] – may go some way to address this (and is starting to expand into supramolecular electronics [REF]), but much research into molecular computing looks at ways to differentiate itself from silicon computing. One way is Natural Computing, looking at computation in nature and how it can be appropriated for human-driven applications [REF].

Research into leveraging biological components such as DNA, bacteria, neurons and other cells to build computers shows great promise for several reasons. Firstly, the computing pedigree of cells is already proven as they are extremely complex computers in their own right; with complex regulatory networks and internal structure to maintain homeostasis [Levine and Davidson (2005)]. Secondly, as they exist in nature in aggregate, they are capable of self assembly into more complex structures allowing for a greater level of system complexity than our present fabrication methods [Li et al. (2009)]. Thirdly, advances in technology in DNA sequencing and sequence synthesis allow us to create custom DNA sequences and the possibility of transfecting these sequences into cells to alter their behaviour [Felgner et al. (1987)].

In his breakthrough paper, Adleman showed how mixing specific DNA sequences in an otherwise unordered solution could solve a seven-node Hamiltonian path problem by exploiting the parallelism offered by the massive number of interacting DNA molecules [Adleman (1994)]. Engineered *Escherichia coli* strains with promoters sensitive to specific quorum molecules have been shown to exhibit the functionality of logic gates when in colonies, and multiple colonies can communicate to form logic

circuits [Tamsir et al. (2011)]. A bacterial photosynthetic protein *bacteriorhodopsin* has been used to implement high-capacity random-access memory [REF], logic gates [REF], amongst other applications [REF]. Research into the transcription of genes has shown how the promoter/repressor nature of transcription operations can be used to produce logic gates [Silva-Rocha and de Lorenzo (2008)].

The engineering of cells to implement functions and behaviours not seen in nature is known as synthetic biology, and has two distinct and competing strategies. The first is top-down and aims to take existing cells and modify them for new purposes. Taking entire genetic blocks from one organism and introducing them into another is now common, such as using Green Fluorescent Protein (GFP) to monitor the production of other proteins by introducing the GFP gene into the same regulatory sequence [Chalfie et al. (1994)]. The top-down implementation of binary logic gates by introducing new transcription regulation networks is difficult as there is no direct equivalent to a Boolean logic gate in biology and the operation of existing transcription networks interfere with implanted circuits [Silva-Rocha and de Lorenzo (2008)].

The alternative strategy is bottom-up, producing a minimal cell that contains only the essential components needed [Porcar (2009)]. These protocells or ‘chells’ (chemical cells) [Pasparakis et al. (2009b)] [Pasparakis et al. (2009a)] [Cronin et al. (2006)] are free from the interference of other cellular processes and their operation is massively simplified. One such strategy in the implementation of a protocell that implements a Boolean logic gate is to enclose a transcription regulatory network-like subsystem in a self-assembled liposome vesicle, producing a simple computational building block that

could be the foundation of more complex protocell networks [Smaldon et al. (2010)]. Like cells, protocells require some functional content to be contained within the membrane, and a method to communicate between protocells.

Vesicles act as membranes to encapsulate and separate functionality, to reduce interference and hide implementation, much in the same way a computer programmer separates code into sub units. Membranes can exist within one another, producing a hierarchal nested structure with clear boundaries around functions and well defined sources of input and output. The logic of membrane

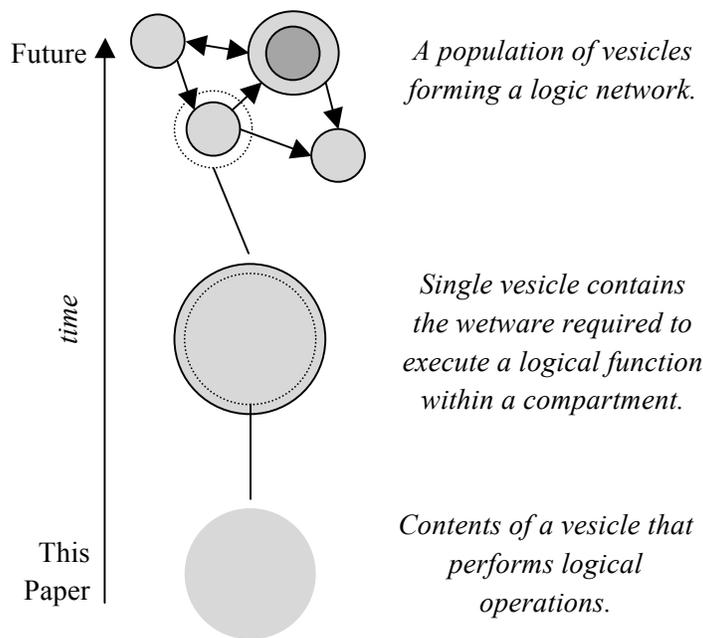


Figure 1: This paper focuses on the functional content of vesicles; a first step towards networks of vesicle logic functions.

computers is expressed as P-Systems [Păun (1998)]; a cellular modelling inspired logic which represents computation as objects contained within a membrane hierarchy, and the state of the system evolving over time in a parallel fashion via the application of object-transforming rules.

Having demonstrated *in silico* that protocells/chells can compute [Smaldon et al. (2010)], in this paper we take the first steps toward the experimental implementation of a bottom-up P-System, as shown in Figure 1. We do this by utilizing NitroBIPS as the active content in a putative liposome system.

To implement a membrane computer, we focused first on the symbols and rules that operate within a single membrane in a manner that abstracts away from the actual implementation of the membrane. We seek an implementation that represents a single ‘compartment’ containing logic-capable components acting as symbols. These symbols are unordered within the compartment, but rules must be applied to some symbols but not others. It is necessary that different symbols react and can be transformed between different states. The size of vesicles mandates the symbols be small; both to fit within the vesicle, but also possibly to move between vesicles via pores in the membrane. **LINK SENTENCE**

Photochromic molecules are those with two stable forms which can be reversibly switched between the two via the absorption of appropriate wavelength electromagnetic radiation. One of the earliest identified and most heavily studied families of photochromic molecules are the spiropyrans [Berkovic et al. (2000)]. Spiropyrans have a colourless leuco spiropyran (SP) state and a coloured trans-merocyanine (MC) state. The change from SP to MC is called colouration and is in response to ultraviolet light, and the change from MC to SP state is called decolouration and is in response to visible spectrum light. **Species of spiropyrans have already shown promise in producing logic gates [REF].**

One spiropyran molecule is NitroBIPS. NitroBIPS is a spiropyran with a nitro group on the 6-position of the benzopyran section [Görner et al. (1996)]. Colouration is with UV light and the MC state additionally decolourises when it absorbs visible wavelength light, with an absorption peak at green [Wohl and Kuciauskas (2005)] causing the MC molecules in solute to be an intense purple. A small proportion of MC molecules that absorb a visible photon change to an excited MC* fluorescent state, followed by a rapid reversion to MC and the emission of an orange photon instead of a decolouration change. As this fluorescence only occurs when the molecule is in the MC state, we can determine the ratio of SP to MC molecules via the intensity of the fluorescence when exposed to visible wavelength light. The two states of NitroBIPS can be imagined as two symbols in a P-System, with rules enacted by exposing the molecules to light.

Here, we demonstrate the computational possibilities of NitroBIPS such that we can both store integers as the relative proportion of SP and MC molecules and execute logical functions when exposed to patterns of light.

2. MOLECULAR SWITCHES

The symbols in our P-System are represented with molecular switches. Molecular Switches are molecules with two or more stable states that can be reversibly switched between the states. They can be switched with many stimuli, including light, temperature, pH value of solute, or the presence of other chemicals. Photochromic molecules are molecular switches for which the stimulus is light.

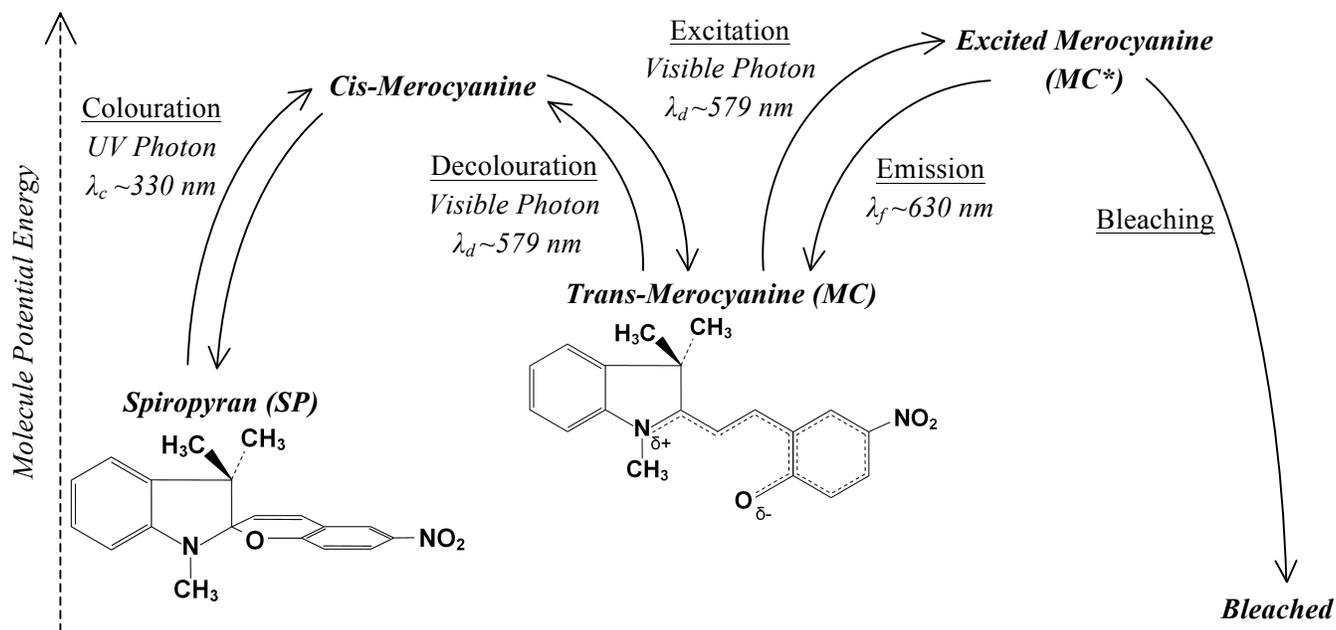


Figure 2: State diagram for NitroBIPS. The vertical axis is the approximate energy of each state representing the stability of that state. In addition to the labeled changes caused by the absorption of photons (with peak absorption wavelengths for Toluene [Wohl and Kuciauskas (2005)]), transitions without a photon label can happen spontaneously. Thermal effects can additionally cause any transition in the system. Molecule potential energy valid only for Spiropyran, Cis-Merocyanine and Trans-Merocyanine states and transitions, taken from [Wojtyk et al. (2000)]. The energy needed to realize the cis-merocyanine to trans-merocyanine isomerisation increases in polar solvents, resulting in the changing peak absorptions for colouration and decolouration as the necessary photon energy increases or decreases [Wojtyk et al. (2000)].

Of the photochromic molecules, spiropyrans are well studied and have been used in many practical applications, including their most famous use in light-reactive spectacle lenses. [Lyliane Le Naour-Sene (1981)]. NitroBIPS (1',3',3'-dihydro-1',3',3'-trimethyl-6-nitrospiro[2H-1-benzopyran-2-2'-2H-indole] or 6-nitro-BIPS and shortened to NBIPS) is one such spiropyran photochromic molecule and benefits from high quantum yields (discussed later), dissolution in many solvents and biological compatibility. Modification of NBIPS and related spiropyran compounds allows passive targeted attachment to proteins to improve imaging contrast [Mao et al. (2008)], but also more active functions such as altering the activity of enzymes depending on the molecule's state [Aizawa et al. (1977)], the reversible modulation of protein dipolar interactions [Sakata et al. (2005)], light-controlled operation of a channel in *e-coli* membranes [Koçer et al. (2005)] and light-controlled perturbation of lipid membranes [Ohya et al. (1998)]. These all provide possible further expansion to our research, but presently we focus on the behavior of NitroBIPS in isolation.

NitroBIPS possesses two stable states; SP and MC. SP is the more stable of the two and is a colourless Leuco dye. SP molecules will change to MC molecules via a short-lived cis-merocyanine state

upon absorption of an ultraviolet photon, wavelength λ_c , [Marriott et al. (2008)] with a probability referred to as the Quantum Yield of Colouration (Φ_c). The Quantum Yield gives the proportion of absorbed photons that cause a molecular state change. MC molecules are an intense purple due to a high absorption rate in the middle of the visible spectrum, peaking around green (λ_d) [Wohl and Kuciauskas (2005)] with a corresponding Quantum Yield of Decolouration (Φ_d). Absorbing these visible photons causes either a change back to the SP state, or with a lower quantum yield (Φ_f) a change to an excited MC* fluorescent state, rapid reversion to MC and the emission of an orange photon (λ_f). A summary of these state transitions can be seen in Figure 2.

Fluorescence is the emission of a photon by the relaxation of a previously excited electron to the ground state of the molecule. The photon released will be of a longer wavelength (and hence lower energy) than the exciting photon and excess energy lost as heat due to molecular vibrations. For NitroBIPS, visible photons in the green range ($\lambda_d \sim 579$ nm) excite the MC molecules, and the emission is in the orange range. ($\lambda_f \sim 630$ nm). The peak wavelength of emitted photons and the peak wavelength of excitation vary depending on the wavelength used in the colouration of the molecule; different ultraviolet wavelengths cause different Merocyanine isomers to be formed, each with a different absorption and emission spectra [Wohl and Kuciauskas (2005)]. The ability of NitroBIPS to fluoresce only in the MC state is essential as it allows us to measure the proportion of states in a sample of NitroBIPS.

NitroBIPS is solvatochromic so the peak wavelength of the absorption spectrum depends on the solvent in which it is dissolved [Wojtyk et al. (2000)]. The peak wavelength varies with polarity; the less polar the solvent, the lower the energy of the peak wavelength [Görner and Matter (2001)]. The quantum yields also improve in lower-polarity solvents [Görner and Matter (2001)]. Typically NitroBIPS is used with Toluene, Ethanol or Methanol. Although Toluene is the least polar of these two (and therefore the quantum yields would be higher), Ethanol or Methanol are more commonly used as they are cheap, readily available and safer than Toulene.

NitroBIPS is also subject to bleaching; the excited MC* molecule may also undergo inter-system crossing in which it irreversibly enters the non-fluorescent triplet state and ceases to function. Bleaching causes the dynamic range of a given sample of NitroBIPS to decrease over time as the proportion of bleached molecules increases and the maximum level of fluorescence reduces. We discuss strategies to deal with this effect later.

NitroBIPS molecules can also undergo random thermally induced transitions. As SP is the more stable, a sample of NitroBIPS will stabilise in a majority-SP equilibrium that is dependent on the temperature. This effect is one of the primary issues in using spiropyrans as a long term data storage medium [Wojtyk et al. (2000)]. Again, the rate of thermal decay is solvent dependant, with lower polarities decaying quicker. Reported rate constants at 25°C include 0.122 s⁻¹ for toluene to 0.000069 s⁻¹ for Ethanol, and unobservably low for water (over three days) [Wojtyk et al. (2000)].

Spiropyran molecules have been encapsulated in lipid vesicles between 200 and 500nm in diameter. [Carol A. Jennings et al. (1997)] The amount of NitroBIPS molecules encapsulated in one of these vesicles depends on the concentration of NBIPS and the volume of the vesicle, but could be calculated by:

$$N_{NBIPS} = \frac{1}{6} \pi N_A c_i d^3 \quad (x)$$

Where N_{NBIPS} is the number of NitroBIPS molecules, N_A is Avogadro's number, c_i is the molar concentration of NitroBIPS molecules in the solution (in mol/m³), and d is the interior diameter of the vesicle (assuming the vesicle is spherical). As an example, a 4mM solution of NitroBIPS (corresponding to 4 mol/m³) encapsulated into 500nm diameter spherical vesicles would give approximately 160,000 molecules of NitroBIPS per vesicle.

3. COMPONENT IMPLEMENTATION: REGISTERS AND LOGIC GATES

To demonstrate the computing possibilities of NitroBIPS, we first aim to replicate the conventional logic paradigm, which consists of three basic elements; registers, logic gates and logic circuits. Registers allow us to store data, logic gates to process simple logical functions and logic circuits combine logic gates into more complex logical functions.

In this section, registers and logic gates are taken in turn and described with a conceptual overview of the principles behind its operation, and the practical implementation required, showing in principle that these could be encapsulated within a single liposome. The subsequent section will demonstrate how these components can be used in combination to execute a proof-of-concept logic circuit.

A crucial factor here is that both registers and all implementable logic gate types utilize the same hardware and differences in functionality are realized by altering the patterns of light they are exposed to. This allows for non-invasive alteration of functionality and reallocation of resources.

3.1 REGISTERS

CONCEPTUAL OVERVIEW: STORAGE

A register in modern computing is a memory element used to store a number. In conventional computers, the integers are stored in a binary format as either integers or floating point numbers with finite precision.

A sample of molecular switches can be used to store data as the ratio of molecules occupying each of its two stable states, giving a non-binary integer register. An integer register can store a value $V_0, V_1, V_2 \dots V_{\max}$. Any switching molecule with two stable states can have values encoded in this manner.

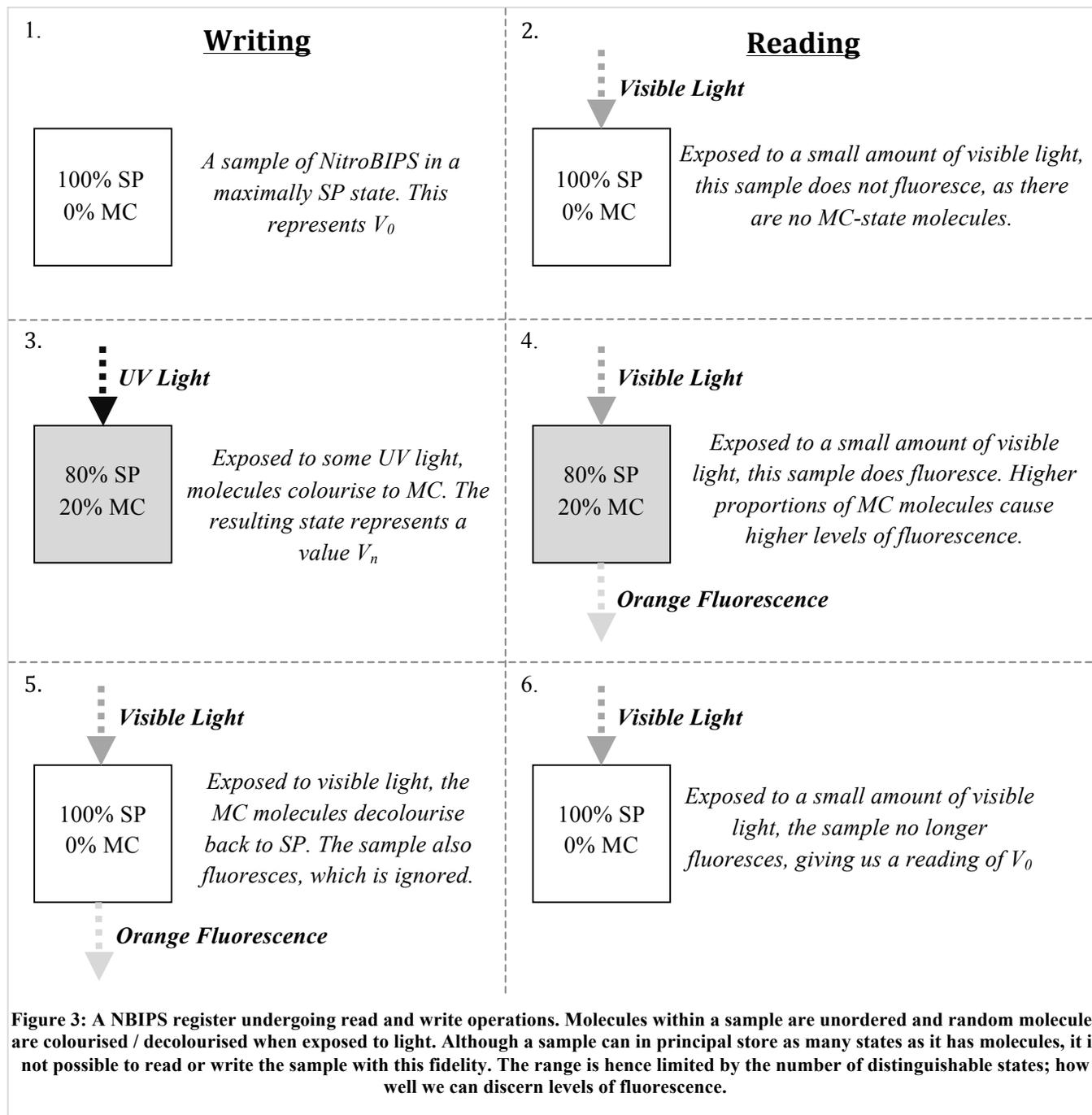
In a sample of NitroBIPS the total concentration of molecules, c_{NB} , is comprised of sub-populations of molecules in different states: $c_{NB} = c_{SP} + c_{MC} + c_{MC^*} + c_B$. The fluorescent lifetime of the excited MC* state molecules, τ_{MC^*} , and the rate of bleaching, dc_B/dt , are negligible relative to the illumination times so we can assume $c_{NB} = c_{SP} + c_{MC}$.

We designate SP as the base state (as the most stable of the two states), such that a sample in state $c_{SP}^{\max} + c_{MC}^{\min}$ represents V_0 and $c_{SP}^{\min} + c_{MC}^{\max}$ represents V_{\max} . Note that the maximum and minimum concentrations of SP or MC state molecules do not represent 100% or 0% of molecules being in the state; random thermal transitions prevent a perfectly homogenous sample being achieved.

The relative proportion of MC molecules, and thus value of the register, can be altered using ultraviolet and visible light. UV light increments the register, and visible light decrements it. As only MC molecules fluoresce when exposed to visible light, the intensity of fluorescence is relative to the proportion of MC state molecules, as shown in Figure 3.

The capacity of the register, or the total number of values it can represent, is determined by the total number of distinguishable values that it can accommodate. This is determined by the dynamic range of the system, which is limited by measurement noise. The dynamic range is limited by several sources of noise including shot noise associated with the probabilistic nature of photon absorption and subsequent fluorescence emission¹; and noise in the light source, detector and detection electronics. In practice each value will have an uncertainty associated with it (defined by the standard deviation σ_V) and adjacent values must be spaced apart by $z\sigma_V$ where greater values of z increase the reliability of value preparation and measurement. All references to V_x are, therefore, made to the set of distinguishable values.

¹ Shot noise is the fluctuation of photons detected from a light source due to the random nature of photon emission from the light source and the random conversion of photons into electrons by a photodiode. These processes follow a Poisson distribution in which the variance σ^2 in the measured signal equals the mean number of detected photons n . With many photons any random variation in the number of photons emitted is very small, in accordance with the law of large numbers, but if the intensity of the light is low the shot noise becomes more significant. Shot noise is often a less significant source of noise than many other sources in the system (such as thermal noise or electromagnetic interference) but represents an inescapable lower bound on the reliance of measurements.



WRITING

Writing values is possible with the use of ultraviolet (λ_c) and visible (λ_d) light pulses; ultraviolet colourises and is used to increment a register's value, and visible light to decrement the value via decolouration. Photons cause state changes in accordance with the quantum yields, but photons must first be absorbed by a molecule in the sample. The Optical Density (OD) of a sample is the number of orders of magnitude incident light is attenuated by (due to absorption) as it passes through the sample, given as:

$$OD = \log_{10} \left(\frac{I_{\lambda}^0}{I_{\lambda}^T} \right) \quad (1)$$

Where I_{λ}^0 is the intensity of light entering the sample, and I_{λ}^T is the transmitted light; the intensity after the light has travelled through the sample. Increasing the ratio of NBIPS in the mix increases the OD, as does increasing the thickness of the sample in accordance with the Beer-Lambert law:

$$OD = \varepsilon_x l c_x \quad (2)$$

Where ε_x is the molar absorptivity of SP or MC state NBIPS molecules (a measure of how strongly a chemical species absorbs a given wavelength of light; MC and SP will absorb different wavelengths with different strengths), l is path length of the light through the sample, and c_x is the concentration of SP or MC molecules. Optical Density is sample specific, and will be different for each wavelength of light and also changes depending on the state of the sample.

Rates of colouration and decolouration have a non-linear dependence on the relative concentrations c_{SP} and c_{MC} . For example, during colouration, as the concentration c_{SP} decreases the probability of incident photon absorption also decreases, which lowers the rate of change of the register value in an exponential manner. Increasing the Optical Density causes NitroBIPS colourations/decolourations to no longer obey the exponential formula as the illumination is attenuated significantly as it passes through the sample. However, increasing the optical density also means the available light is used more efficiently as each photon is more likely to be absorbed by the sample. Experimentally, we determined an OD of 0.1-0.2 to be an acceptable trade-off between these two effects.

During decolouration a sample of NitroBIPS at a molar concentration of $c_{NB} = N_{NB} / N_A V$, with volume $V = Al$, is assumed to be illuminated with a constant, uniform, monochromatic light source of wavelength λ_d and intensity $I_d^0 = N_{ph}^0 hc / \lambda_d A \Delta t$ over the sample area A for a short time interval Δt , where N_{ph}^0 is the number of incident photons, N_A is Avogadro's number, h is Planck's constant and c is the speed of light.

The MC state molecules in the sample absorb a fraction of the incident photons as described by the Beer-Lambert law, $N_{ph}^{ab} / N_{ph}^0 = 1 - T = 1 - 10^{-\varepsilon_{MC} c_{MC} l} = 1 - e^{-\ln(10) \varepsilon_{MC} c_{MC} l}$, and the number of molecular state transitions, from state MC to state SP, is given by $\Delta N_{MC} = -\Phi_d N_{ph}^{ab}$ where N_{ph}^{ab} is the number of absorbed photons, T is the transmittance (the proportion of unabsorbed photons), and Φ_d is the quantum efficiency of the MC \rightarrow SP decolouration transition. This results in a change of concentration $\Delta c_{MC} = \Delta N_{MC} / N_A V$

In the limit $\Delta t \rightarrow 0$ the concentration change of MC state molecules is therefore,

$$\frac{dc_{MC}}{dt} = -\frac{\Phi_d}{N_A V} \frac{I_d^0 \lambda_d A}{hc} \left(1 - e^{-\ln(10) \varepsilon_{MC} c_{MC} l} \right) \quad (3)$$

For samples with a low optical density, $\epsilon_{MC}c_{MC}l$, the incident intensity is not attenuated appreciably as it passes through the sample. In this case e^x can be approximated by a truncated Taylor series expansion, $e^x \approx 1 + x$, which simplifies the concentration change to:

$$\frac{dc_{MC}}{dt} = -\frac{\Phi_d I_d^0 \lambda_d}{N_A hc} \ln(10) \epsilon_{MC} c_{MC} l \quad (4)$$

The solution to this equation is:

$$c_{MC}(t) = c_{MC}^0 e^{-k_1 t} \quad (5)$$

where $k_1 = \frac{\Phi_d I_d^0 \lambda_d}{N_A hc} \ln(10) \epsilon_{MC}$. The concentration of molecules in the SP state is given by:

$$c_{SP}(t) = c_{NB} - c_{MC}(t) \quad (6)$$

Similarly, if the sample is illuminated at a wavelength λ_c with intensity I_c^0 , the SP state molecules will absorb a fraction of the incident photons and cause a colouration molecular state transition from state SP to state MC. In this case the concentrations are:

$$c_{SP}(t) = c_{SP}^0 e^{-k_2 t} \quad (7)$$

where $k_2 = \frac{\Phi_c I_c^0 \lambda_c}{N_A hc} \ln(10) \epsilon_{SP}$ and

$$c_{MC}(t) = c_{NB} - c_{SP}(t) \quad (8)$$

To abstract from these equations, the exposure of a register to UV light is represented as \uparrow_y and visible light as \downarrow_y , where y is the value change (positive for \uparrow , negative for \downarrow). As an example, a V_0 register exposed to \uparrow_3 light will result in a V_3 register. The duration of each pulse is calibrated to cause a change of y distinguishable values in accordance with the equations above. This formalism also allows us to abstract away from the precise wavelengths and generalize the method for other photochromic molecules.

Registers have a finite range; they cannot be decolourised below V_0 nor colourised above V_{\max} . This represents a floor and ceiling constraint on NBIPS registers. The response of a register to light pulses hence follows the following rules:

$$V_x + \uparrow_y = V_{x+y} \quad \left| \quad x + y < \max \quad (9)$$

$$V_x + \uparrow_y = V_{\max} \quad \left| \quad x + y \geq \max \quad (10)$$

$$V_x + \downarrow_y = V_{x-y} \quad \left| \quad x > y \quad (11)$$

$$V_x + \downarrow_y = V_0 \quad \left| \quad x \leq y \quad (12)$$

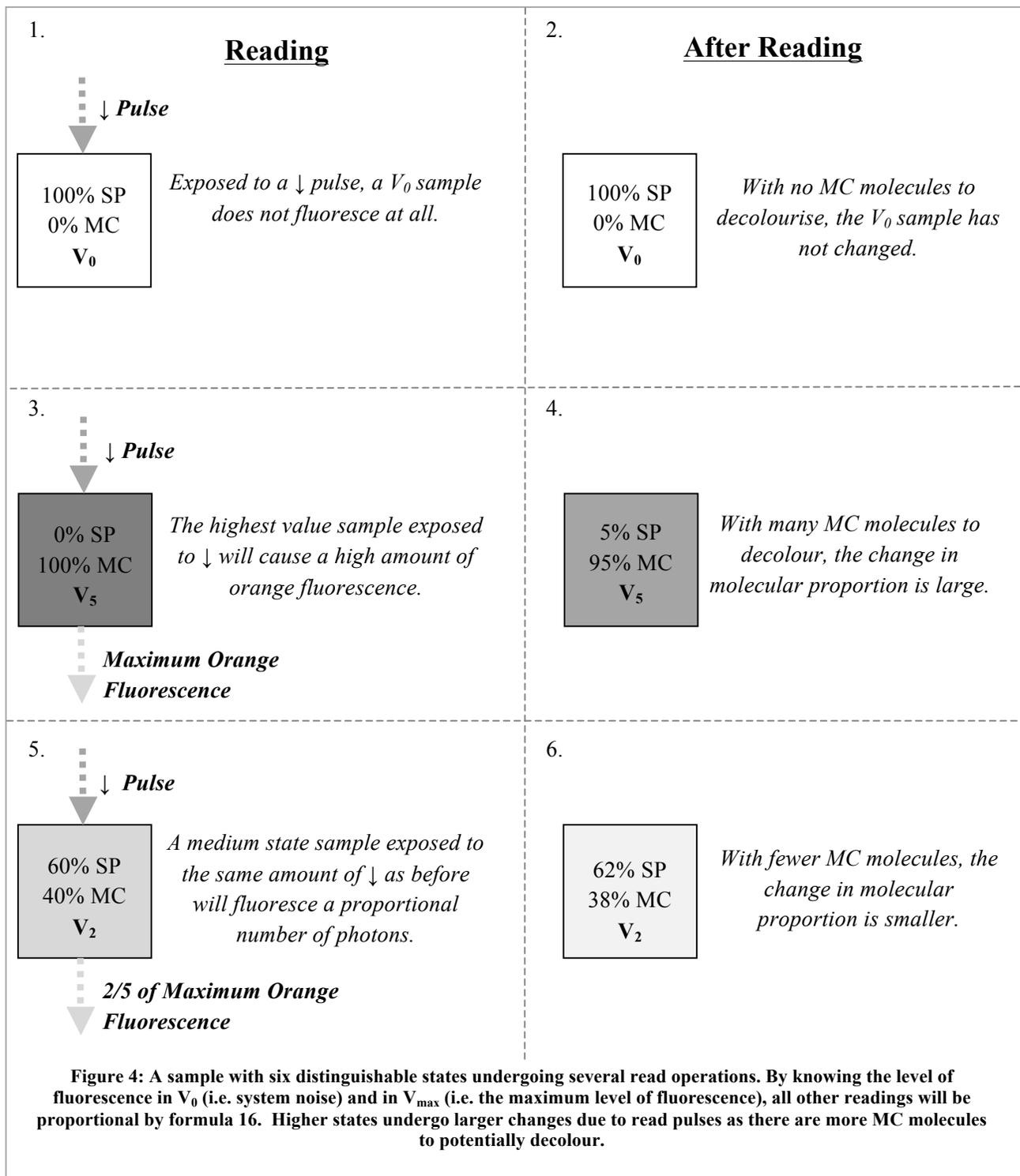
Where 9 is a sample's response to \uparrow_y unless subject to the ceiling restriction 10; and 11 is the sample's response to \downarrow_y unless subject to the floor restriction 12.

READING

NBIPS registers are read using the fluorescence signal resulting from a \downarrow pulse. The number of photons emitted is proportional to the number of MC-state molecules in the system. Note that exposing the sample to \downarrow will also cause some MC molecules to decolour to the SP state. As a result, measuring the value of a register will also change it, as summarized in Figure 4. As only MC molecules cause fluorescence, the fluorescent signal will start strongly and decrease exponentially as decolouration occurs. The value of the register is determined from the orange fluorescence emission intensity at the start of the \downarrow pulse.

During the read pulse the sample is illuminated at λ_d and a fraction of the absorbed photons (intensity $I_d^{ab} = I_d^0(1 - T)$) will cause a brief transition to the excited MC* state with an efficiency Φ_f , an emission of a photon of wavelength λ_{MC^*} and a return to the MC state, or (rarely) to the bleached state. A long pulse gives more data points that can be fitted to the exponential curve to determine the initial intensity. This is more robust to noise and gives a more reliable measurement but comes at the expense of causing a larger change to the register's value. A sufficiently short pulse minimizes this change and doesn't require curve fitting but it reduces the reliability of the measurement.

The two read strategies we use are Short-Pulse Reading and Reset Reading. The former uses a short read pulse and ignores the resulting decolouration change. The average emitted light intensity during the excitation pulse duration gives the register's current value. This method is simple and quick, but prone to noise as the read pulse is very short. It also causes a small reduction in the state of the register due to decolouration and this can accumulate with subsequent read pulses.



The second strategy is Reset Reading. Reset Reading uses longer read pulses, which provides more signal to determine the register's value but causes a large change in this value. As Reset Reading alters the register value, if we wish for a register to retain the original value it must be reset back using a \uparrow pulse of the appropriate intensity. The pulse duration required to reset the register can be calculated precisely as the value change caused by each read pulse is known.

Reset Reading has a few advantages. As the signal is longer, we can record more values with our experimental setup which helps average out noise. Provided the optical density of the sample is low, the signal will follow an exponential decay curve with equation:

$$I_{MC^*}(t) = I_{MC^*}^0 e^{-kt} \quad (13)$$

Where $I_{MC^*}(t)$ is the intensity of fluorescence emission at time t after the read pulse began, $I_{MC^*}^0$ is the initial intensity (and is hence the value we want to know) and k is a decay constant unique to the sample. The value for k can be determined experimentally for each sample during an initialization step. By fitting equation 13 to the recorded values for $I_{MC^*}(t)$ an accurate estimate for $I_{MC^*}^0$ can be produced.

Secondly, it acknowledges that reading a register alters its value and addresses the problem by resetting the state after each read. The disadvantages of this approach are the extra time and energy required; not only are the read pulses longer, but it is necessary to calculate and execute the pulse necessary to reset the register value after each read.

Regardless of the reading method used, by measuring the fluorescence emission intensity of a register prepared with the minimum value, V_0 , and the maximum value, V_{max} , we can establish the dynamic range all other measurements will fall into. To establish these values, and also determine the rates of colouration and decolouration k_1 and k_2 , we need to initialize the register. Initialization of a register requires five steps:

- i. Determine the speed of colouration k_2 . As colouration does not cause fluorescence, we use \uparrow light and many very short \downarrow pulses to track the increase in colouration over time as a measure of relative fluorescence.
- ii. Determine the maximum orange emission intensity $I_{MC^*}^{max}$ by exposing the now V_{max} register to a \downarrow pulse R of intensity I_R and duration T_R . This read pulse will be constant for all other value readings. Reset the state to V_{max} .
- iii. Determine the rate of decolouration k_1 . As decolouration does cause fluorescence, we can track this as the relative fluorescence decreases.
- iv. Determine $I_{MC^*}^{min}$ as the minimum level of orange emission intensity by exposing the now V_0 register to a read pulse of intensity I_R and duration T_R .
- v. Determine the noise by calculating the standard deviation of the fluorescence signal. This dictates the number of distinguishable values between $I_{MC^*}^{min}$ and $I_{MC^*}^{max}$.

As noise is normally distributed as shown in Figure 5, we can measure background noise to determine the standard deviation σ . The reliability of measurements is then chosen as the proportion of measurements p that will fall within z standard deviations of the desired measurement:

$$p = erf\left(\frac{z}{\sqrt{2}}\right) \quad (14)$$

Where erf is the error function:

$$erf(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-x^2} dx \quad (15)$$

As an example, 99.7% of measurements will be within 3σ either side of the expected value. We then divide the dynamic range by the value spacing:

$$d = \frac{I_{MC^*}^{\max} - I_{MC^*}^{\min}}{2r\sigma} \quad (16)$$

To determine the number of distinguishable states d . If we read emission using a read pulse of fixed duration and intensity, and the minimum orange emission is $I_{MC^*}^{\min}$ and the maximum $I_{MC^*}^{\max}$, then the orange emission ($I_{MC^*}^x$) of value V_x will be:

$$I_{MC^*}^x = \left(x \left(\frac{I_{MC^*}^{\max} - I_{MC^*}^{\min}}{d - 1} \right) \right) + I_{MC^*}^{\min} \quad (17)$$

For example, a register with 11 distinguishable states 0-10:

Register Value	Relative Orange Emission
V_0	0
V_1	1
V_2	2
...	...
V_9	9
V_{10} (i.e. V_{\max})	10

PRACTICAL IMPLEMENTATION

EXPERIMENTAL SETUP

To produce a register we first require a sample of NitroBIPS we can expose to light and record fluorescence from. For our experiments we use NitroBIPS from Sigma Aldrich [Sigma-Aldrich Co. (2011)] supplied as a powder. The concentration of NitroBIPS will change over time if it is left unencapsulated as the solvent evaporates. To solve this we use Methanol as our solvent to produce a solution and then a small amount of the mix is folded into uncured polydimethylsiloxane silicone polymer (PDMS). Once the PDMS has cured it is cut into pieces to be used. The technique requires a high concentration of NBIPS in Methanol, we use 4mM. This was then mixed into the PDMS at various ratios up to a maximum PDMS:NBIPS ratio of 5:1 to create encapsulated NBIPS at a final concentration of 0.67mM.

The desired concentration and thickness of a sample of encapsulated NBIPS depends on a trade-off between the rate of colouration/decouration fitting the exponential formula, and the rate of light

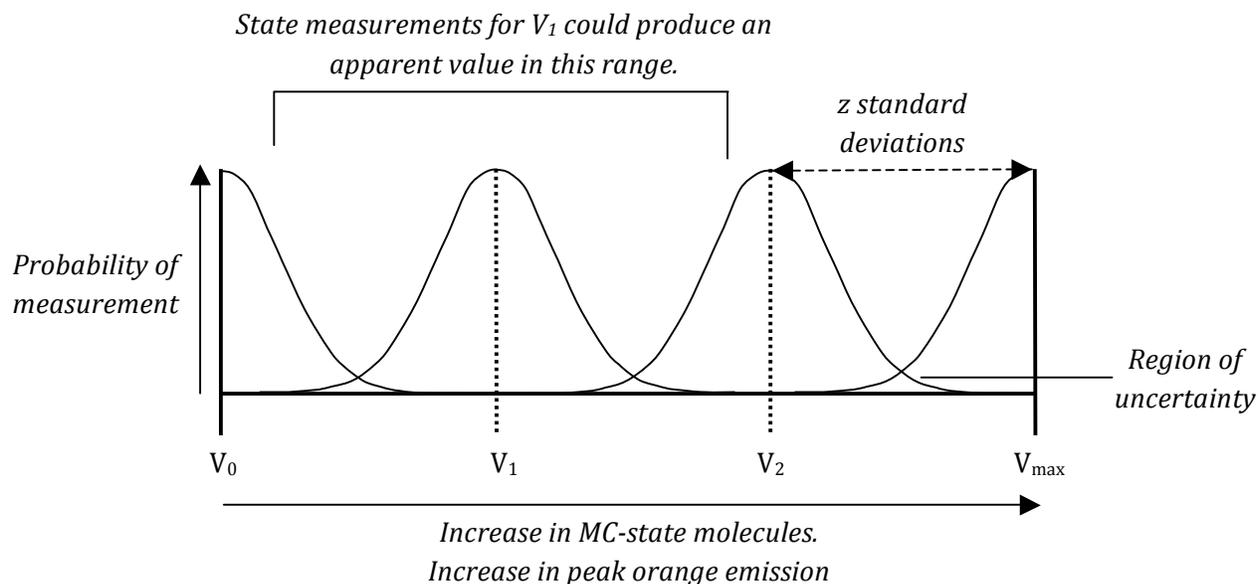


Figure 5: The probability distribution for each register value measurement is normally distributed. It is not possible to completely eliminate crossover between adjacent register values. Increasing the separation between adjacent values increases reliability but reduces the dynamic range of the register.

absorption. A 1mm thick sample of 4mM 5:1 eNBIPS has a measured optical density of 0.17 for 530nm green light when in a maximally MC state, and 0.1 for 365nm UV light when in a maximally SP state. These values are highly approximate but confirm the Optical Density is sufficiently low.

To read and write values to our NitroBIPS registers, we need to expose samples to Ultraviolet or Green light to enact \uparrow and \downarrow pulses. To do this, we constructed an optical system shown in Figure 6, featuring two high-power LEDs; one 365nm Ultraviolet and one 530nm Green. Köhler illumination arms are then used to project a uniform disc of light, with an adjustable diameter, onto the sample. Light from the two LEDs is combined with a dichroic mirror; a reflective short-pass filter which reflects below a specified wavelength and allows the transmittance of wavelengths above. This light then illuminates the sample. LEDs were used as a source because they are bright, narrow-band, and easy to control. The maximum irradiance at the sample achieved with this setup was $2.2 \mu\text{W}/\text{mm}^2$ for the UV LED and $87 \mu\text{W}/\text{mm}^2$ for the green LED.

The LEDs are controlled by an LED driver which sets the intensity and is gated by commands from a LabVIEW program running on a computer, as shown in Figure 7. LabVIEW is a graphical programming environment with a data-flow centric architecture, which is designed to interface easily with external hardware.

The relative concentration c_{MC} can be measured by the fluorescence from the excited MC^* state. Fluorescence is measured using a photodiode and an amplifier. The signal is filtered using a 10Hz low-pass Butterworth filter to remove as much background noise as possible, especially 50Hz mains interference. The signal is sampled at 50Hz, which also helps to reduce mains noise.

The emitted photons are collected by a lens (with a solid angle ratio of $\Omega/4\pi = (1 - \cos\theta)/2$) and directed to a photodiode by the optical system (with efficiency η) giving an intensity at the detector of:

$$I_{MC^*} = I_d^{ab} \Phi_f \frac{\Omega}{4\pi} \eta \quad (18)$$

The detected light is converted to a current I (with a responsivity of $R_\lambda = I/I_{MC^*}$) and amplified by a transimpedance amplifier (with gain $R_F = V/I$) to give the measured voltage for fluorescence of V_{MC^*} .

Therefore for low optical density samples,

$$V_{MC^*} = R_F R_\lambda \Phi_f \frac{\Omega}{4\pi} \eta I_d^0 \ln(10) \epsilon_{MC} c_{MC} l \quad (19)$$

i.e.

$$V_{MC^*} = k c_{MC} \quad (20)$$

where k is a constant.

As the intensity of emission I_{MC^*} is linearly proportional to the MC concentration c_{MC} , and as the measured voltage from the photodiode V_{MC^*} is linearly proportional to I_{MC^*} , it is possible to determine the rates of colouration and decolouration while working with the photodiode signal, without the need to convert back to intensities or concentrations. This gives us the following modified equations for colouration:

$$V_{MC^*}(t) = V_{MC^*}^{\max} - (V_{MC^*}^{\max} e^{-k_2 t}) + V_{offset} \quad (21)$$

Decolouration:

$$V_{MC^*}(t) = V_{MC^*}^0 e^{-k_1 t} + V_{offset} \quad (22)$$

And the determination of distinguishable states, as it is the noise in this voltage measurement that determines the ability to discern two distinct measurements:

$$d = \frac{V_{MC^*}^{\max} - V_{MC^*}^{\min}}{2r\sigma} \quad (23)$$

$$V_{MC^*}^x = \left(x \left(\frac{V_{MC^*}^{\max} - V_{MC^*}^{\min}}{d - 1} \right) \right) + V_{MC^*}^{\min} \quad (24)$$

Where V_{offset} is an additional experimental error value caused by the emission filter allowing a small proportion of green light to the photodiode, causing a fixed apparent increase in emission intensity and hence a fixed increase in measured voltage.

REGISTER USAGE

As mentioned previously, initializing a register requires five steps. The first is to determine the rate of colouration. We do this by exposing a register to UV light interspersed with very brief green pulses to cause small amounts of fluorescence to track the increase in MC state molecules. The amount of fluorescence per pulse is relative to the concentration of MC molecules in the sample; as the sample colorizes, the orange emission intensity will increase and hence so will the photodiode signal. The increase in measured voltage occurs at an exponentially decreasing rate as it approaches the maximum concentration in accordance with equation 21.

All necessary experimental values are known, with the exception of Φ_c and ϵ_{SP} , though a value for $\Phi_c \cdot \epsilon_{SP}$ can be calculated by fitting the experimental data to equation 21 which is sufficient to know the rate of concentration change. This then complete set of values allows us to determine the time taken to colourise between two given concentrations.

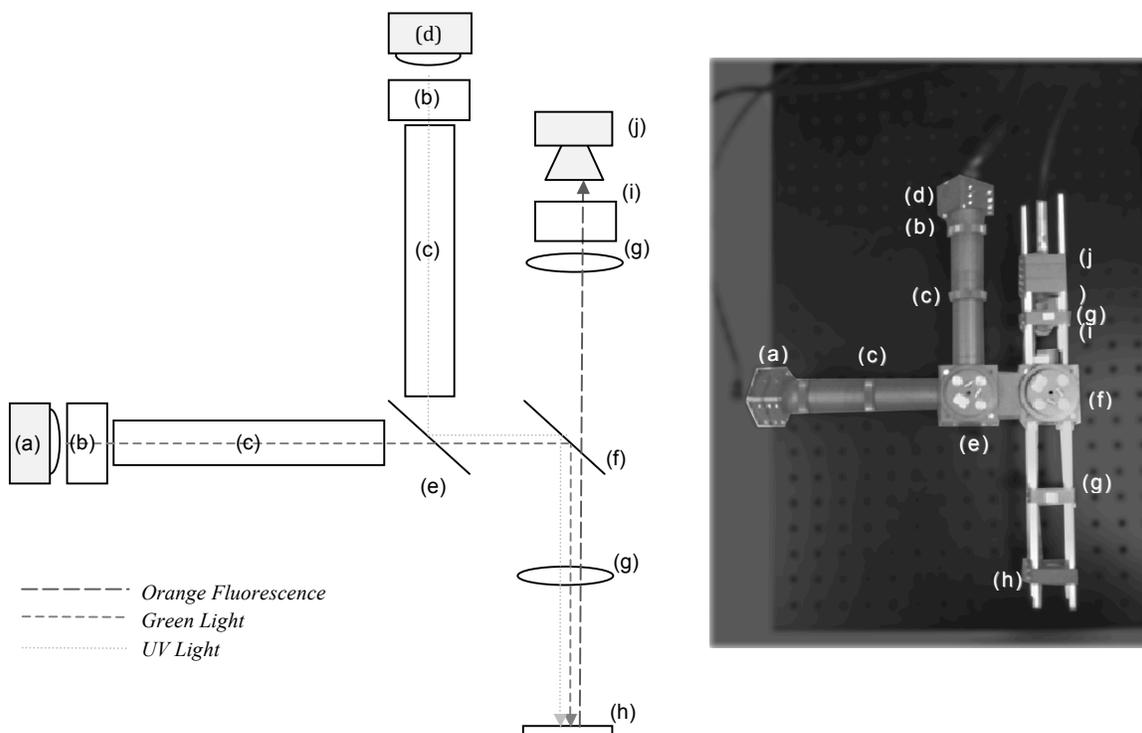


Figure 6: Optical diagram (left) and image of the setup (right). (a): 530nm Green LED. (b): Excitation Filter. (c): Kohler Illumination arms with field and aperture diaphragms. (d): 365nm UV LED. (e): 495nm Dichroic Mirror. (f): 565nm Dichroic Mirror. (g): Lens. (h): eNBIPS sample. (i): Emission Filter. (j): Photodiode.

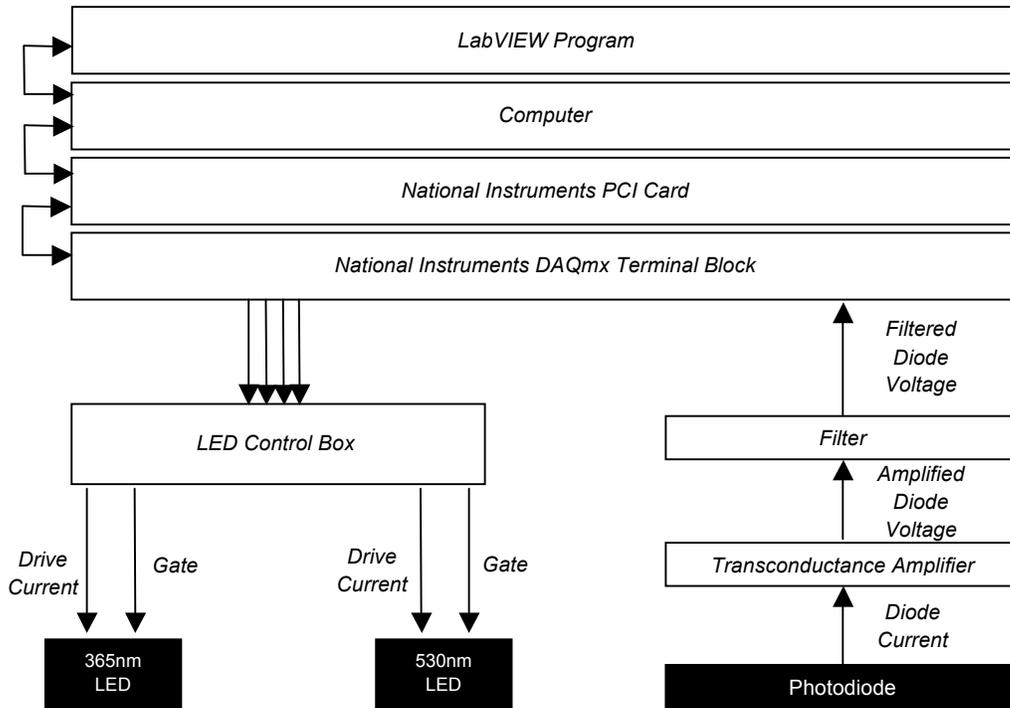


Figure 7: Information and hardware diagram. The LabVIEW program controls the LEDs and records orange emission data from the photodiode. The signals to the LEDs must pass through the National Instruments hardware and to the LED control box. The signal from the photodiode is amplified and noise filtered before being recorded.

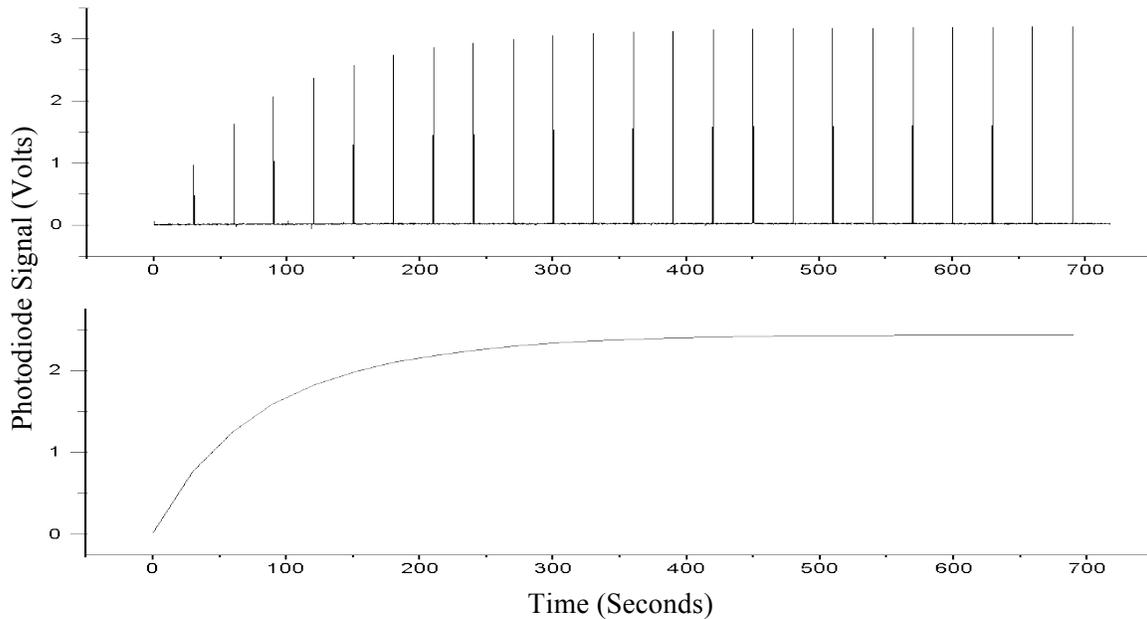


Figure 8: Plot a (top) shows orange emission over time as a V_{min} sample is coloured to V_{max} , interspersed with green pulses to measure relative fluorescence. Plot b (bottom) shows the data taken just from those pulses.

In the example shown in Figure 8a, we see how colouration is measured with repeated short green pulses; 100ms every 30 seconds. Figure 8b shows the curve given by just taking data from the measurement pulses. Knowing values for I_c^0 (2.23 W/m^2), and λ_c ($365 \times 10^{-9} \text{ m}$), we can fit our predicted data to the plot and calculate $V_{MC^*}^{\max}$ as 2.42 V , k_2 as 0.00128 and hence $\Phi_c \cdot \epsilon_{SP}$ as 81 and V_{offset} as 0.001 V .

The second step is to take a reading of the photodiode voltage for $V_{MC^*}^{\max}$ using read pulse R . This is the maximum possible fluorescence from our system. For the example, I_R was the maximum our LED allowed; 87 W/m^2 , and the T_R was 200 ms , and the reading was 3.25 V .

The third step is to determine the rate of decolouration. Decolouration obeys equation 22. In the example shown in Figure 9, knowing I_d^0 (87 W/m^2) and λ_d ($530 \times 10^{-9} \text{ m}$), we can determine $V_{MC^*}^0$ as 3.64 V , k_1 as 0.3936 and hence $\Phi_d \cdot \epsilon_{MC}$ as 443 and V_{offset} as 0.08 V .

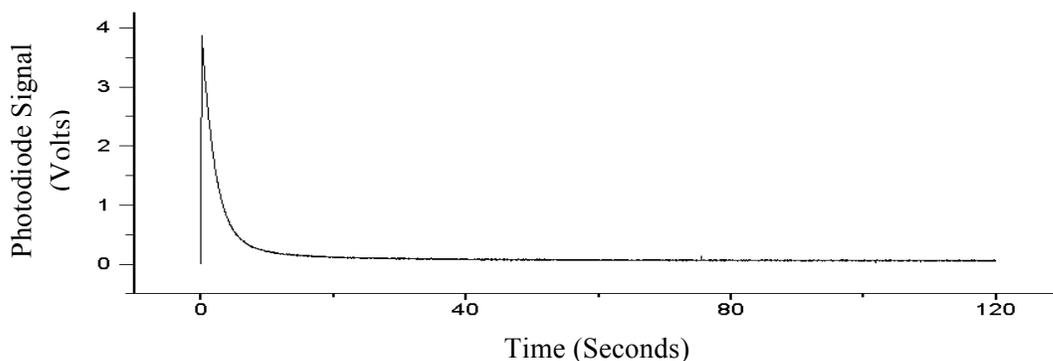


Figure 9: A V_{\max} register decolored to V_{\min} with constant green light.

Step *iv* is to expose the now V_0 register to the read pulse R to get a value for $V_{MC^*}^{\min}$. This will be non-zero due to V_{offset} . In this example, it is equal to 0.06 V .

The final step is to determine the standard deviation of background noise, which in turn determines the number of distinguishable states. Samples are taken over a few seconds, and the standard deviation calculated. The difference between the fluorescence of $V_{MC^*}^{\min}$ and $V_{MC^*}^{\max}$ and the total number of possible values in a sample can now be calculated. In the example's case, the σ of noise was 0.01 . Choosing an acceptable reliability r of 3σ (total within range 99.73%), we now know how large a separation to space values apart in terms of orange emission intensity; 0.06 V . We can then calculate the total number of distinguishable states d using equation 23. For our example, this gives a result of 53.16 . As we should refrain from using values not spaced apart the full separation (as those values would have an associated loss of reliability), the example has 53 distinguishable states.

Over time, the molecules in the sample will bleach, reducing the dynamic range. A register should frequently be reinitialized, at a rate dependant on the rate of bleaching, itself dependant on the solvent NBIPS is dissolved in. As the dynamic range decreases, the number of distinguishable states d will have to decrease or the separation $z\sigma$ and hence measurement reliability will need to be decreased. Which is more appropriate will depend on the practical application.

3.2 LOGIC GATES

CONCEPTUAL OVERVIEW

A logic gate is a device with two or more input channels, and (typically) a single output channel. In most cases, these channels are Boolean. Different patterns of inputs on the input channels will result in a certain value on the output channel, as specified by the truth table associated to the type of logic gate. The most commonly considered form of logic gate is the 2-in/1-out Boolean electronic gate which is used in all modern electronics.

Production of logic gates using NitroBIPS samples is based on the implementation of registers in combination with the floor and ceiling functions inherent to NBIPS operation. The main difference of NitroBIPS logic gates is that unlike silicon logic gates with parallel input, the inputs to the NBIPS gate must arrive serially.

To implement this, a register in V_0 is subjected to light pulses that encode the inputs as shown in Figure 10. After the inputs, the value of the register is checked. If the register is in value V_1 or higher, then the output is considered true. If the register is still in V_0 , the output is false. As a register in V_1 or higher emits orange light when the state is measured, the presence of orange emission is the indicator of a true output. The use of visible light to check the value of the register uses full reset reading; this both ensures the most reliable possible signal and also resets the gate back to V_0 in preparation for the next logical operation.

For two-input gates, there are two corresponding pulses of light acting as inputs to the system. These are potential pulses; the sample is only exposed to the pulses if the corresponding input is true, and not if it is false. However, to achieve more interesting behavior, additional pulses are required which are not dependent on inputs. These ‘moderator’ pulses are independent of the value of the input, and the sample is exposed to them every time.

The operation of a gate is defined as an ordered sequence of light pulses applied to a register. For an n -input gate, n of these pulses will be the inputs that are on or off in accordance with the input values, zero or more will be moderator pulses which are executed regardless of the input pulses, and one will be an output pulse during which the orange emission and hence gate output is measured.

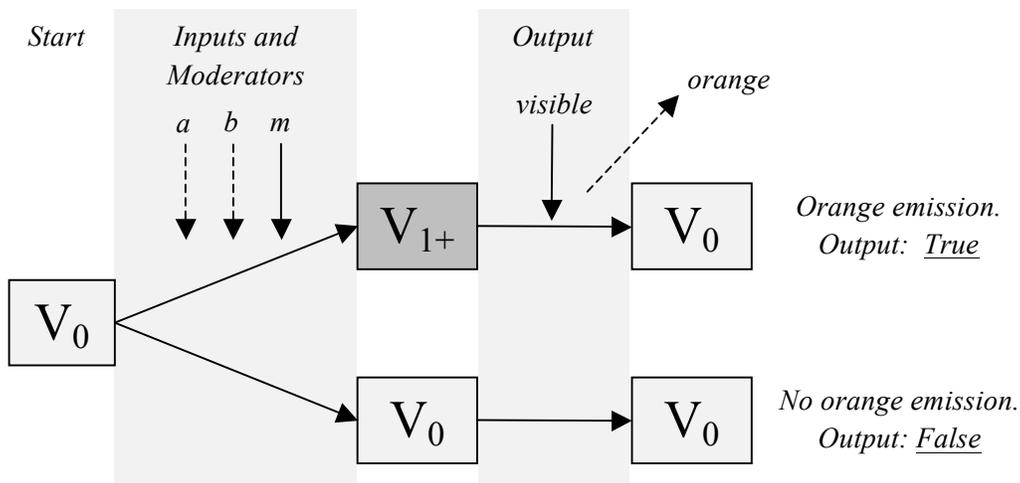


Figure 10: A logic gate's value varies depending on the inputs it receives. Only if it represents a value greater than zero during the output phase is there orange emission to correspond to a true value.

An n -input gate is defined for all possible input values as $[c, (p^*), o]$ where:

c is the capacity of the gate; the number of distinguishable values required to operate. Some gates require a minimum capacity, others an absolute capacity due to a reliance on the ceiling restriction to operate. Those requiring a minimum capacity have their capacity written as $c+$, implying c is the minimum number of values to function. Those requiring a fixed capacity due to the use of the ceiling restriction have their capacity listed as $c!$, implying the capacity must be exactly c .

(p^*) is an ordered list of n input pulses and zero or more moderator pulses.

o is the output pulse.

(p^*) is a list of pulses of length at least n . It contains exactly n input pulses and zero or more moderator pulses. Input and moderator pulses are described in different ways. We define input pulses as a tuple $\{i, \#_x\}$, defining which gate input it represents and the value change it should cause if the input is describes is true. In this general notation, all possible input pulses are listed.

i is the input to which this pulse is associated, written as a or b for a gate with two inputs.

$\#$ is the direction of value change caused by the pulse should the input it is associated to have the value True. If the input has value False, the pulse doesn't happen. For our NitroBIPS system, Ultraviolet light is a positive change, and hence written as ' \uparrow '. Visible causes a negative change, and hence ' \downarrow '. For other switching molecules, the nature of the input may change, but they will still cause a ' \uparrow ' or ' \downarrow ' change in value.

x is the amount of change in value this pulse could potentially cause. As logic gates utilize the same physical implementation as registers, they are also subject to the floor and ceiling restrictions. Gates use these restrictions to operate.

Moderator pulses are not associated with any gate input, so are defined as a single value $\#_x$, with $\#$ and x defined in the same way as input pulses.

The output pulse o does double duty, acting to determine the value of the logic gate after the execution of (p^*) and also to reset the gate to V_0 in preparation for subsequent executions. It is defined as $\{\downarrow_x\}$. x must always be equal to the highest value that a gate could achieve when subject to any set of input pulses i.e. All positive pulses' value changes added together minus the value changes of any negative constant pulses.

For example, a 2-input/1-output AND gate is represented in general as:

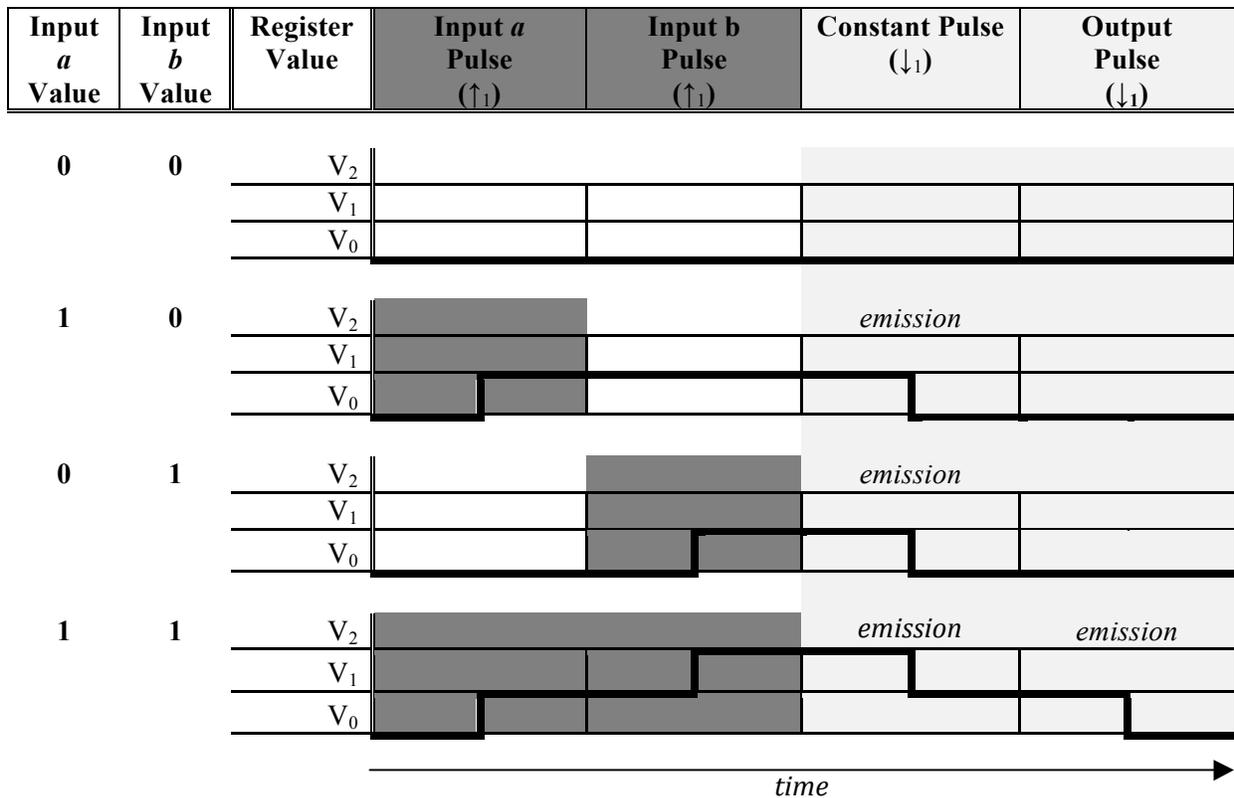
$$2/1 \text{ AND} = [3+, (\{a, \uparrow_1\}, \{b, \uparrow_1\}, \{\downarrow_1\}), \{\downarrow_1\}]$$

The AND gate has two inputs, a and b . Both are associated with \uparrow (i.e. Ultraviolet) pulses with a value change of 1. Input pulses do or do not occur depending on value of the corresponding input to the gate (i.e. a and b). A single moderator pulse is used, a \downarrow (i.e. visible) pulse of value change 1. The output pulse is \downarrow with value change 1. A truth table of the gate is as follows:

Input a Value	Input b Value	Input a Pulse Value Change	Input b Pulse Value Change	Moderator Pulse Value Change	Final Value Before Output	Output Pulse Value Change	Truth Value
0	0	0	0	-1*	0	-1*	0
1	0	+1	0	-1	0	-1*	0
0	1	0	+1	-1	0	-1*	0
1	1	+1	+1	-1	1	-1	1

*Negative value changes marked with an asterisk do not affect the system as the register was already in V_0 and those changes were subject to the floor restriction.

The change in register value over time may be more intuitively expressed as follows:



Where the line for each set of truth values charts the changing value of the logic gate over time with each pulse. A positive output corresponds to a change downwards in value (and hence orange emission) during the output phase. Note that though the logic gate only has two output values, it requires three register values to operate to ensure value changes are not incorrectly subjected to the ceiling function.

Many logic gates can be implemented in this way, including all 1-input/1-output gates and almost all 2-input/1-output gates. Though gates can be implemented with any number of input channels, only one output channel is available.

1-INPUT / 1-OUTPUT

TRUE [2+, ({ \uparrow_1 }), { \downarrow_1 }]
 FALSE [1+, (), { \downarrow_0 }]
 NOT [2+, ({ \uparrow_1 }, {a, \downarrow_1 }), { \downarrow_1 }]
 ID [2+, ({a, \uparrow_1 }), { \downarrow_1 }]

2-Input / 1-Output

FALSE [1+, (), { \downarrow_0 }]
 AND [3+, ({a, \uparrow_1 }, {b, \uparrow_1 }, { \downarrow_1 }), { \downarrow_1 }]
 a NOT b [2+, ({a, \uparrow_1 }, {b, \downarrow_1 }), { \downarrow_1 }]
 a [2+, ({a, \uparrow_1 }), { \downarrow_1 }]
 b NOT a [3+, ({ \uparrow_1 }, {a, \downarrow_1 }, {b, \uparrow_1 }, { \downarrow_1 }), { \downarrow_1 }]
 b [2+, ({b, \uparrow_1 }), { \downarrow_1 }]

OR	[2!, ({a, ↑ ₁ }, {b, ↑ ₁ }, {↓ ₁ })]
NOR	[2+, ({↑ ₁ }, {a, ↓ ₁ }, {b, ↓ ₁ }, {↓ ₁ })]
NOT b	[2+, ({↑ ₁ }, {b, ↓ ₁ }, {↓ ₁ })]
b IMP. a	[3+, ({↑ ₁ }, {a, ↑ ₁ }, {b, ↓ ₁ }, {↓ ₂ })]
NOT a	[2+, ({↑ ₁ }, {a, ↓ ₁ }, {↓ ₁ })]
A IMP. b	[3+, ({↑ ₁ }, {a, ↓ ₁ }, {b, ↑ ₁ }, {↓ ₂ })]
NAND	[3+, ({↑ ₂ }, {a, ↓ ₁ }, {b, ↓ ₁ }, {↓ ₂ })]
TRUE	[2+, ({↑ ₁ }, {↓ ₁ })]

n-INPUT / 1-OUTPUT

(where Ω is the nth unique identifier for inputs)

AND	[n+, ({a, ↑ ₁ }, {b, ↑ ₁ }, ... {Ω, ↑ ₁ }, {↓ _{n-1} }, {↓ ₁ })]
OR	[2!, ({a, ↑ ₁ }, {b, ↑ ₁ }, ... {Ω, ↑ ₁ }, {↓ ₁ })]
NOR	[2+, ({↑ ₁ }, {a, ↓ ₁ }, {b, ↓ ₁ }, ... {Ω, ↓ ₁ }, {↓ ₁ })]
NAND	[n+, ({↑ _n }, {a, ↓ ₁ }, {b, ↓ ₁ }, ... {Ω, ↓ ₁ }, {↓ _n })]

Note that some gates (specifically those with output pulses ↓₂ or ↓_n) could have more than one level of positive output depending on the values of inputs i.e. a NAND gate with one input true would finish in V₁ representing a true output, but with both inputs false would finish in V₂ which also represents a true output. Any system implementing these gates must account for this factor.

Of the 2-in/1-out gates, not listed are XOR and XNOR. No implementation exists in this scheme for these two gates. Implementing XOR is possible if an upper and lower bound on value was implemented; i.e. only a gate in V₁ would count as a positive output.

$$\text{XOR} \quad [3+, (\{a, \uparrow_1\}, \{b, \uparrow_1\}, \{\downarrow_2\})]$$

Input a Value	Input b Value	Input a Pulse Value Change	Input b Pulse Value Change	Final Value Before Output	Output Pulse Value Change	Truth Value
0	0	0	0	0	-2*	0
1	0	+1	0	1	-2* (-1)	1
0	1	0	+1	1	-2* (-1)	1
1	1	+1	+1	2	-2	0

The highlighted column shows crucially that an output equivalent to V₀ and V₂ implies false, and only an output of V₁ is true. This goes against the general principle that any orange emission represents a true output. This would require an additional layer of complexity however in determining if the output should be true or false. XNOR could be implemented in the same way with the same pulse pattern, but by designating V₀ and V₂ as true outputs, and V₁ as false. However, as the range of possible gates include many universal sets (including NOR and NAND; both sole sufficient operators), XOR and XNOR can be implemented as circuits of other gates.

PRACTICAL IMPLEMENTATION

Logic gates use the same encapsulated NBIPS samples and optical setup as registers, the only difference is the patterns of light to which they are exposed. A modified LabVIEW program is used to expose the samples to the specific patterns of light required, and monitor the orange emission during the output pulse to see if the output is true, and record the Boolean value for that gate execution.

The output of the gate is the presence or absence of orange emission during the output phase. The quantity of orange emission is not considered provided it is greater than V_1 's minimum bound as many gates can have more than one 'true' output value with varying intensities of output emission. For example, the 2/1 NAND gate has both V_1 and V_2 representing a true output.

The OR gate uses the ceiling function to operate, but practically this is problematic as it takes a long time to fully colourise a NBIPS sample in accordance with equation 7. The speed of execution of an OR gate can be massively increased by instead changing the definition to:

$$2/1 \text{ OR} \quad [2+, (\{a, \uparrow_1\}, \{b, \uparrow_1\}), \{\downarrow_2\}]$$

And like NAND, considering both V_1 and V_2 as true.

As logic gates only require a few values in a sample to operate, it is possible to use small or less concentrated sample of eNBIPS, or increase the separation between values to improve reliability.

For the gate to be used repeatedly, the check/reset pulse must be long enough to reset the gate back to V_0 's target every time. Failure to do so will result in an erroneous initial value and incorrect operation. There will always be some fluorescence during the output phase, even if the logic gate is in V_{\min} . This is due to the imperfect emission filter. This must be accounted for by having a threshold for fluorescence, above which output is considered true.

4. SYSTEM INTEGRATION: RUNNING A LOGIC CIRCUIT

CONCEPTUAL OVERVIEW

To perform more complex logical operations, we can chain together logic gates to form a logic circuit. The circuit will have one or more inputs, and one or more outputs, each corresponding to a component gate's input or output. The connections between gates, and the type of gates used governs the behavior of the logic circuit.

A logic circuit is defined as a set of logic gates, but with an extended notation to record how the gates are connected. A n-input logic gate is now represented as $[id, c, (p^*), o]$ where:

id is a unique gate identifier that other gates' input pulses will reference.

c is the register capacity as defined previously.

(p^*) is the list of n input pulses and one or more moderator pulses, as defined previously.

o is the output pulse, as defined previously.

Input pulses in (p^*) have an extended definition. They are now represented as $\{s, i, \#_x\}$ where:

s is the source for this input. This is either the id of a previous gate in the logic circuit, or the id of a circuit input. The value of the source determines the value of the input.

$i, \#$ and x remain as defined previously.

The circuit as a whole then has the definition $[(i^*), (o^*), (g^*)]$ where:

(i^*) is the list of one or more circuit inputs. Each circuit input is an identifier so gates can reference the input. No true/false value is assigned as the circuit definition is a general one.

(o^*) is the list of one or more circuit outputs. Each output is the identifier of a gate within the circuit, such that the value of the output of that gate is the value of the output of the circuit.

(g^*) is the list of gates within the circuit.

As an example, the simple circuit C_{ex} in Figure 11 would be represented as:

$$C_{ex} = [(i_0, i_1, i_2), (g_1), ([g_0, 3+, (\{i_0, a, \uparrow_1\}, \{i_1, b, \uparrow_1\}, \{\downarrow_1\}), \{\downarrow_1\}], [g_1, 3+, (\{\uparrow_2\}, \{g_0, a, \downarrow_1\}, \{i_2, b, \downarrow_1\}), \{\downarrow_2\}])]$$

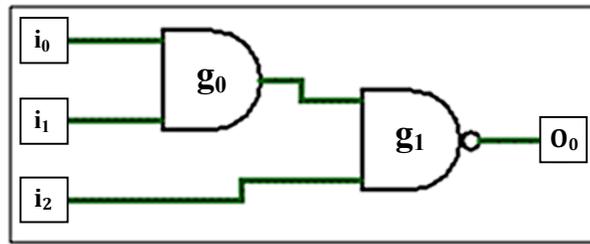


Figure 11: A simple circuit C_{ex} with three inputs, two gates and a single output. Equal to $O_0 = (i_0 \text{ AND } i_1) \text{ NAND } i_2$

PRACTICAL IMPLEMENTATION

The practical implementation of logic circuits requires several stages. Firstly, the circuit itself must be designed. Secondly, as our present implementation only supports a single gate at a time, the circuit must be serialized. Thirdly, the LabVIEW program must be given the circuit definition to execute, and lastly actual circuit execution.

We design the circuit in Logisim [Burch (2005)], a tool for designing and simulating logic circuits, offering a simple drag and drop interface and numerous optimization features allowing us to ensure any circuit is optimized before use, potentially saving time and resources, and also allowing us to check the circuit's truth table and Boolean expression, to ensure the circuit does what we want it to.

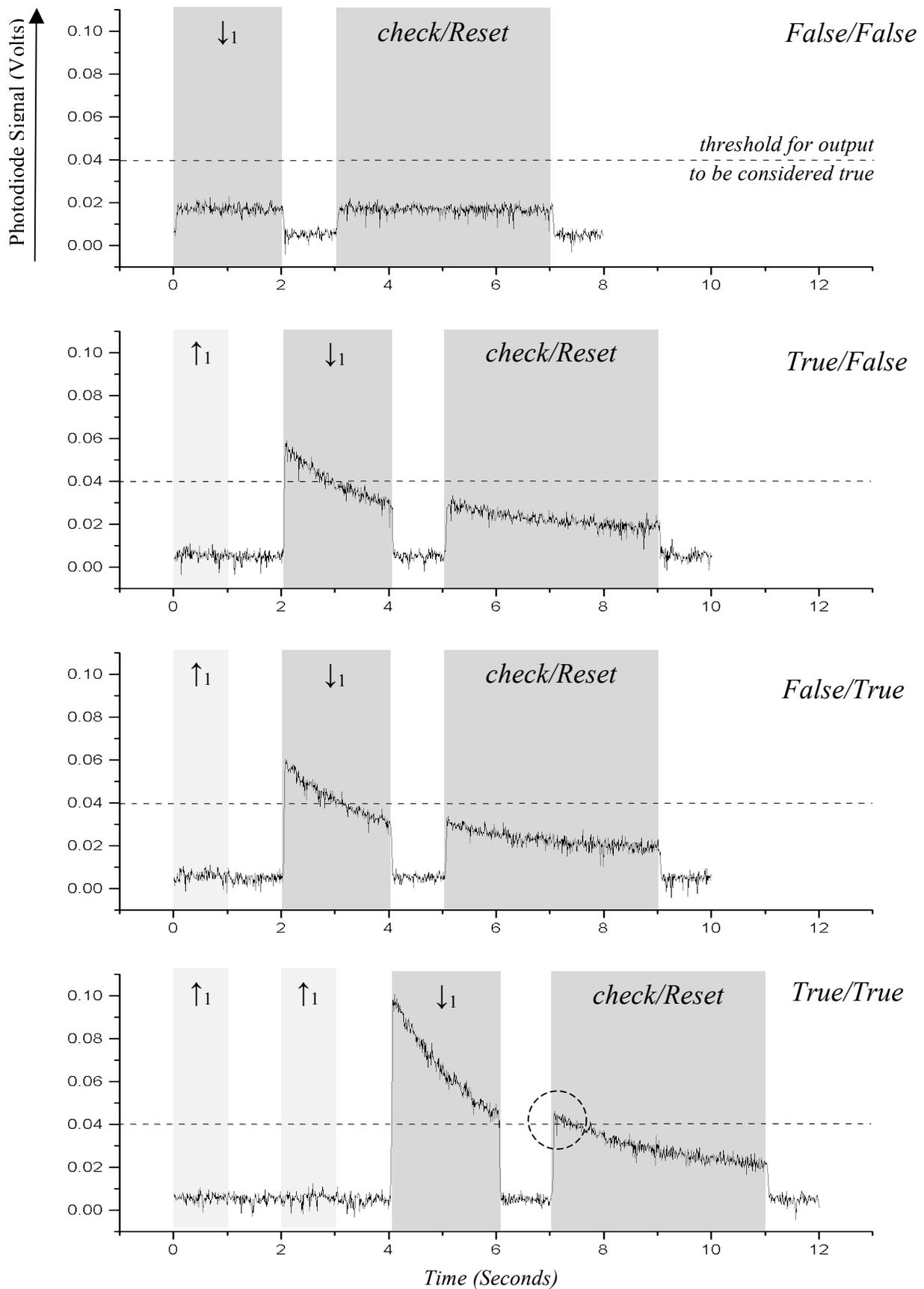


Figure 12: The fluorescence during all four possible pulse combinations for the AND gate. Each ‘True’ input has a corresponding \uparrow_1 pulse, where as ‘False’ inputs are represented by the absence of a light pulse. Only when both inputs is true is the peak fluorescence during the output phase above the threshold. A one second gap has been added between pulses to show clearly where the moderator pulse ends and the check pulse begins, this is unnecessary in practice.

Logisim generates a '.circ' file defining the circuit, and we parse it using our custom software 'Logisim Parser', which performs several tasks:

- i. File/Wire Parser: The save format of Logisim is the .circ file. This XML-based text file defines the structure of the circuit in a very literal manner using coordinates and attributes for each component in the circuit. The File Parser uses DOM parsing to turn the XML file into objects, and the Wire Parser uses the coordinates of the wires in the system to determine which components are connected to one another.
- ii. XOR/XNOR fixer (XF): As Molecular Switch Logic Gates do not currently implement XOR or XNOR gates, and as these gates are useful in many logic circuits, the XF translates any such gates in the circuit into an equivalent set of NOR gates which implements the same functionality as the XOR/XNOR gate but over a necessarily longer time. The XF can also replace the entire circuit with NOR or NAND gates, providing a uniform gate type
- iii. Serialiser: The current hardware implementation can only use a single well at a time. As such, any logic gate which has parallel elements must be refactored into a serial order. The algorithm for this is as follows:
 1. All gates and inputs have an 'order' property O_x . This starts unassigned (O_{null}). An order is a unique value representing the serial order gates must be executed in.
 2. Assign the n input nodes an order from O_0 to O_{n-1} .
 3. Loop through all gates. If all the inputs to a gate have an order assigned, assign that gate order O_x such that x is the lowest order not currently assigned. If one or more inputs have order O_{null} , skip.
 4. Repeat loop until all gates have non- O_{null} order assigned.

Pattern Output	Meaning
2	Highest register state required.
1	Number of parallel logic gates (max 1 currently)
3	Number of Inputs
0,1	Inputs: Input Reference, Input Value
1,1	
2,1	
1	Number of Outputs
0,10	Output: Output Reference, Output Source
3,0,2,UC,1	Pulses:
4,0,1,UC,1	Pulse Reference, Logic Gate id, Source Reference, Type (UV
5,0,-1,GD,1	Colouration, Green Decolouration, Green Output), State
6,0,-1,GO,1	Change.
7,0,-1,UC,2	
8,0,6,GD,1	
9,0,0,GD,1	
10,0,-1,GO,2	

Figure 13: The Pattern output for the two-gate circuit example above.

The LabVIEW program takes the pattern output from the parser as input. The user is then asked to specify values for each of the circuit inputs. A few parameters such the duration of pulses and orange emission threshold are set. Lastly, the LabVIEW program loops, executing each light pulse in turn and displaying the values of any gates which represent outputs.

For each pulse, it must firstly be determined if the pulse is actually to be run. Each pulse could be dependent on a circuit input value, or on the value of a previous output pulse. These are checked, and the pulse cleared for execution. A +5V signal is sent to the correct LED along the on/off channel and the intensity voltage along the intensity channel for the duration constant the pulse duration. If the pulse is an output pulse, the orange emission detected from the photodiode is logged and checked to see if it exceeds the threshold. The result of this reading is saved in an Output Values array. Once all pulses are executed, the Output Values array is checked for the gates that represent outputs, and their values displayed.

The output of a logic gate (i.e. orange light) is not a valid input to subsequent gates, prohibiting the direct connecting of gates as you would electronic gates. Our present system converts orange output into the necessary input for the next gate. Though this uses a computer, a simple electronic circuit could be used instead.

An example execution of the circuit in Figure 11 is shown in Figure 14. The eight traces correspond to the eight possible patterns of inputs for i_0 , i_1 and i_2 . The example circuit outputs true for all possible input combinations except when all three inputs are true.

5. DISCUSSION AND CONCLUSIONS

The field of synthetic biology [Pasparakis et al. (2009a)] has long aimed to produce cells which are engineered to perform specific tasks. The top-down approach of modifying existing cells has been a successful area of research, but is limited by evolutionary baggage that adds unwanted excess complexity and conflicts. The counter-part to this method is bottom-up, producing artificial cells or ‘chells’ [Cronin et al. (2006)] from scratch, with no unwanted components. The use of lipid membranes to form vesicles is one fledgling method of producing chells by encapsulating chemical functionality. One possible use of vesicles is to produce a vesicle computer (i.e. a physical implementation of a P-System [Smaldon et al. (2010)]), but vesicles require a content that enables computational tasks. [Pasparakis et al. (2009b)]

NitroBIPS is a photochromic molecule with two stable states; SP and MC. Ultraviolet light causes colouration reactions changing SP to MC, and green light causes decolourations from MC to SP. Green light also causes fluorescence; emitting orange photons. Similar spiropyran molecules have been shown to be containable in lipid vesicles [Carol A. Jennings et al. (1997)]. **Data can be stored as the relative concentrations of SP and MC-state molecules.**

Long term storage of NitroBIPS samples is not possible due to thermal decay. This could be overcome by cooling the sample or applying constant low intensity ultraviolet illumination proportional to the state of the sample to compensate for the decay. For short term storage

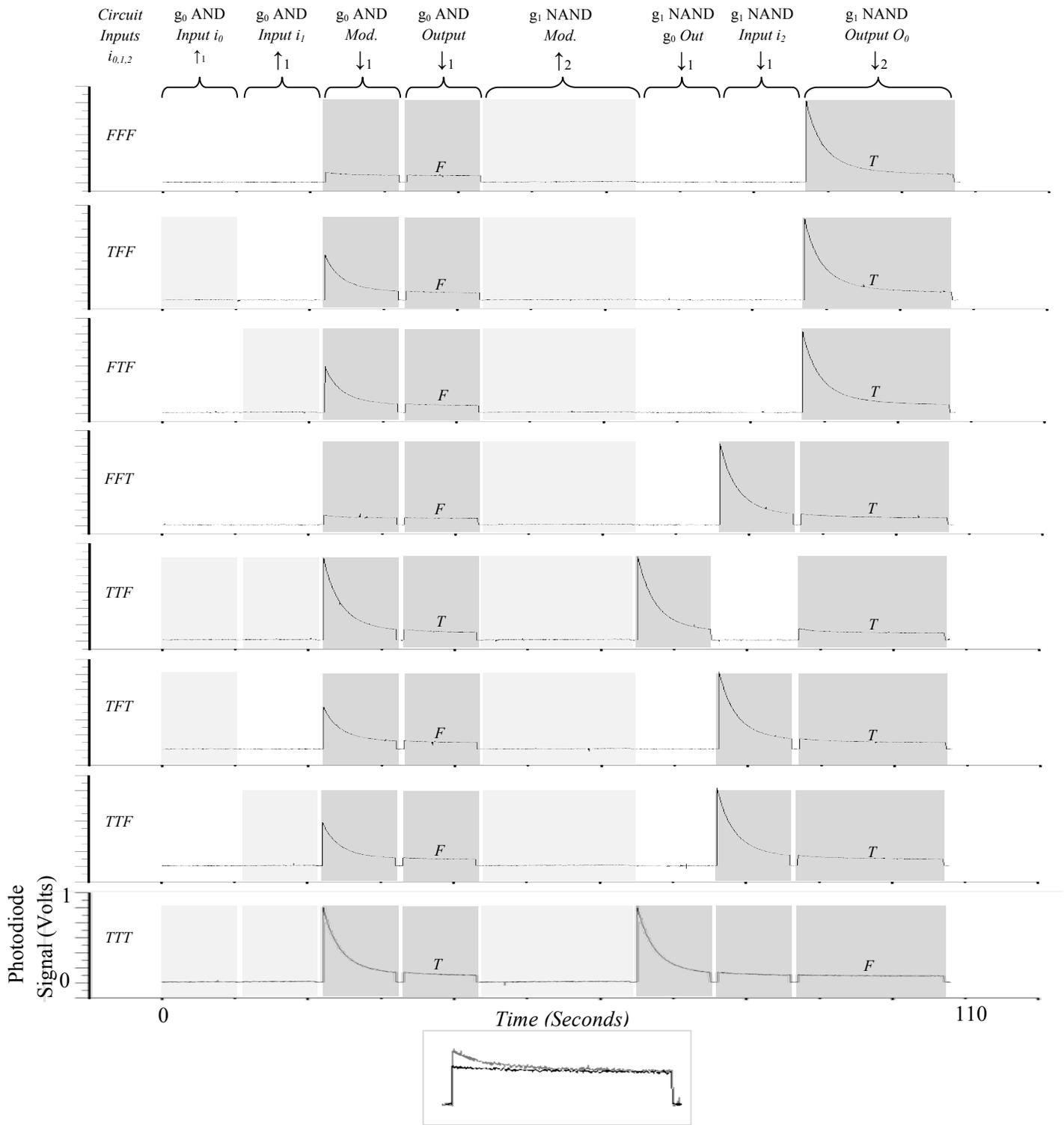


Figure 14: All eight possible sets of inputs into the example circuit in Figure 11 create eight possible circuit executions. Each trace above shows the relative orange emission over time as the well is exposed to light pulses detailed at the top of the figure. The shaded regions denote where the light pulse is being enacted versus unshaded regions where no light is incident to the well due to gate inputs being false.

The letters during the output pulses represent if that orange emission during that period corresponds to a true or a false output; the vertical compression of the figure makes the distinction from the trace alone less obvious. The inset above shows a larger view of the TTF NAND output (grey) and the TTT NAND output (black) demonstrating the difference between True and False outputs.

Note that for clarity, the execution waits during a false input rather than moving on so all eight traces align. Also, the length of pulses has been extended to make the orange spikes more obvious to the naked eye, and gaps inserted between pulses to help show where one ends and another begins.

applications as registers this effect can be mitigated provided the register is set to zero periodically; either due to using full reset reading, or in general use.

A second consideration for storage with NitroBIPS is photobleaching. As shown on the state diagram, the emission of orange photons can sometimes cause a molecule to bleach; enter a third stable state from which it cannot be switched back into the SP or MC state. A bleached molecule is for our purposes a broken molecule. It reduces $I_{MC^*}^{\max}$ and hence reduces the possible range of orange emission the sample can fluoresce and the number of distinguishable states possible. To counteract this, registers can be periodically re-initialized; establishing a new $I_{MC^*}^{\min}$, $I_{MC^*}^{\max}$ and number of distinguishable states to ensure bleaching does not cause incorrect readings, though this reduces the range of the register over time.

Logic Gates extend the functionality of registers by computing Boolean functions when exposed to specific patterns of light. A combination of three types of light pulses are used; input pulses which implement the Boolean inputs to the logic gate; moderator pulses which occur irrespective of inputs; and a final check pulse which determines the value of the logic gate. Unlike electronic logic gates where inputs must arrive in parallel, this is a serial system with each input arriving in turn. If the combination of inputs leaves the gate in a minimally-MC value, the check pulse causes no fluorescence and corresponds to the output False. If the combination leaves the gate in a higher value, the check pulse will cause orange emission and correspond to True. Using this system, we have implemented all 1-input Boolean functions, all 2-input Boolean functions with the exception of XOR and XNOR, and several n-input functions.

Logic Gates can be run serially to produce more complex logical functions such as adders. Our current system only runs a single logic gate at a time, so any logic gate must be serialized before being compiled into a sequence of light pulses, and the results of check pulses saved to be used as inputs to later gates. We have built a program that does this automatically.

NitroBIPS and related spiropyran molecules seem well suited to use in vesicle logic systems for several reasons. The optical nature of interactions allows for the communication with an external component in a non-destructive way and the capability of selective addressing of many vesicles simultaneously both in two and three dimensions (via two-photon excitation [Parthenopoulos and Rentzepis (1989)]), though appropriate wavelength light sources only exist at present for visible pulses. NitroBIPS can be labeled to specific proteins, and the state of the NitroBIPS molecule can alter the behavior of the protein [Sakata et al. (2005)], potentially offering a method of optically changing internal vesicle functions. NitroBIPS could also be incorporated into the vesicle itself, offering controlled perturbation of the membrane causing it to become selectively 'leaky' [Ohya et al. (1998)]. What is lacking is a method of other chemical processes to alter the state of the NitroBIPS molecule, allowing NitroBIPS to report on the vesicle's operations and opening the possibility of two-way optical communication with the vesicle. It is hoped future photochemical research may offer a means to achieve this.

The techniques and methods described in this paper are not just relevant to NitroBIPS, but could be implemented on many switching molecules with two states and a means to determine the proportion of molecules in each state. The switching methods need not be light, but could be any temporally-controllable stimulus. Light, electrical current and temperature all being good candidates due to ease in

controlling the stimulus, where as pH and the presence of other chemicals being less suited as the removal of the stimulus is difficult unless these stimuli were also light controlled.

An ideal photochromic compound would be one with quantum yields of 1.0 for all transitions, ensuring maximum use of incident light. It should switch on absorption of two distinct wavelengths of light with a narrow absorption spectrum, and with powerful light sources available, and preferably light sources of double wavelength available also (for two-photon excitation). It should be thermally stable, and resilient to bleaching to allow for long term storage of data, and a long lifespan of use. NitroBIPS registers have the disadvantage of changing when read, as the wavelength for decolouration and fluorescence is the same. An ideal compound would have a third 'read' wavelength which fixes this. It would be cheap, non-toxic and simple to use in equally cheap, non-toxic solvents. It would also have a structure which allows for the modification to include protein targeting, allowing it to bind into biological processes.

Our current aim is to extend the functionality of the system to multiple samples at a time, allowing for parallel execution of registers and logic gates. We aim to implement this in one of two ways. The first is to have a movable strip of PDMS divided into cells so that we can illuminate one cell at a time. This allows for parallel data storage, but not the execution of several gates simultaneously. A second approach is the use of a Spatial Light Modulator (SLM). An SLM is a grid of pixels which can be selectively made reflective or non-reflective, allowing a specific pattern to be illuminated on a sample. In conjunction with an encapsulated NBIPS sample, we can address multiple regions simultaneously, allowing for multiple registers/gates to be accessed simultaneously and run in parallel. Both methods offer the interesting possibility of dynamic reallocation of resources between data storage and logical function, as both use the same hardware.

6. REFERENCES

- [Adamatzky (2001)] Adamatzky, A., 2001. Computing in nonlinear media and automata collectives. IOP Publishing Ltd., Bristol, UK, UK.
- [Adleman (1994)] Adleman, L. M., 1994. Molecular computation of solutions to combinatorial problems. *Science* 266, 1021–1024.
- [Aizawa et al. (1977)] Aizawa, M., Namba, K., Suzuki, S., 1977. Photo control of enzyme activity of [alpha]-amylase. *Archives of Biochemistry and Biophysics* 180 (1), 41 – 48.
- [Berkovic et al. (2000)] Berkovic, G., Krongauz, V., Weiss, V., 2000. Spiropyran and spirooxazines for memories and switches. *Chemical Reviews* 100 (5), 1741–1754.
- [Burch (2005)] Burch, C., 2005. Logisim: A graphical tool for designing and simulating logic circuits. <http://ozark.hendrix.edu/~burch/logisim/> Retrieved Feb 2011
- [Carol A. Jennings et al. (1997)] Carol A. Jennings, E. C., Marcel P. Breton, M. C., MaryAnna Isabella, M. C., Eric G. Johnson, P. C. F., Trevor I. Martin, B. C., John F. Oliver, C. C., 05 1997. Ink compositions with liposomes containing photochromic compounds. Patent, uS 5633109.
- [Chalfie et al. (1994)] Chalfie, M., Tu, Y., Euskirchen, G., Ward, W., Prasher, D., 1994. Green fluorescent protein as a marker for gene expression. *Science* 263 (5148), 802–805.
- [Cooper et al. (2008)] Cooper, S., Löwe, B., Sorbi, A., 2008. New Computational Paradigms: Changing Conceptions of What is Computable.
- [Cronin et al. (2006)] Cronin, L., Krasnogor, N., Davis, B. G., Alexander, C., Robertson, N., Steinke, J., Schroeder, S., Khlobystov, A., Cooper, G., Gardner, P., Siepmann, P., Whitaker, B., 2006. The imitation game - a computational chemical approach to recognizing life. *Nature Biotechnology* 24, 1203–1206.
- [Felgner et al. (1987)] Felgner, P. L., Gadek, T. R., Holm, M., Roman, R., Chan, H. W., Wenz, M., Northrop, J. P., Ringold, G. M., Danielsen, M., 1987. Lipofection: a highly efficient, lipid-mediated dna-transfection procedure. *Proceedings of the National Academy of Sciences* 84 (21), 7413–7417.
- [Görner et al. (1996)] Görner, H., Atabekyan, L. S., Chibisov, A. K., 1996. Photoprocesses in spiropyran-derived merocyanines: singlet versus triplet pathway. *Chemical Physics Letters* 260 (1-2), 59 – 64.
- [Görner and Matter (2001)] Görner, H., Matter, S., 2001. Photochromism of nitrospiropyran: Effects of structure, solvent and temperature. *Physical Chemistry Chemical Physics* 3, 416–423.
- [Koçer et al. (2005)] Koçer, A., Walko, M., Meijberg, W., Feringa, B. L., 2005. A light-actuated nanovalve derived from a channel protein. *Science* 309 (5735), 755–758.

[Levine and Davidson (2005)] Levine, M., Davidson, E. H., 2005. Gene regulatory networks for development. *Proceedings of the National Academy of Sciences of the United States of America* 102 (14), 4936–4942.

[Li et al. (2009)] Li, H., Carter, J. D., LaBean, T. H., 2009. Nanofabrication by dna self-assembly. *Materials Today* 12 (5), 24 – 32.

[Lyliane Le Naour-Sene (1981)] Lyliane Le Naour-Sene, V. F., 09 1981. Process of integrating a photochromic substance into an ophthalmic lens and a photochromic lens of organic material. Patent, uS 4286957.

[Mao et al. (2008)] Mao, S., Benninger, R. K., Yan, Y., Petchprayoon, C., Jackson, D., Easley, C. J., Piston, D. W., Marriott, G., 2008. Optical lock-in detection of fret using synthetic and genetically encoded optical switches. *Biophysical Journal* 94 (11), 4515 – 4524.

[Marriott et al. (2008)] Marriott, G., Mao, S., Sakata, T., Ran, J., Jackson, D. K., Petchprayoon, C., Gomez, T. J., Warp, E., Tulyathan, O., Aaron, H. L., Isacoff, E. Y., Yan, Y., 2008. Optical lock-in detection imaging microscopy for contrast-enhanced imaging in living cells. *Proceedings of the National Academy of Sciences* 105 (46), 17789–17794.

[Ohya et al. (1998)] Ohya, Y., Okuyama, Y., Fukunaga, A., Ouchi, T., 1998. Photo-sensitive lipid membrane perturbation by a single chain lipid having terminal spiropyran group. *Supramolecular Science* 5 (1-2), 21 – 29.

[Parthenopoulos and Rentzepis (1989)] Parthenopoulos, D. A., Rentzepis, P. M., 1989. Three-dimensional optical storage memory. *Science* 245 (4920), 843–845.

[Pasparakis et al. (2009a)] Pasparakis, G., Krasnogor, N., Cronin, L., Davis, B., Alexander, C., 2009. Controlled polymer synthesis: from biomimicry towards synthetic biology. *Chemical Society Reviews*.

[Pasparakis et al. (2009b)] Pasparakis, G., Vamvakaki, M., Krasnogor, N., Alexander, C., 2009. Diol-boronic acid complexes integrated by responsive polymers - a route to chemical sensing and logic operations. *Soft Matter*.

[Pâun (1998)] Pâun, G., 1998. Computing with membranes. *Journal of Computer and System Sciences* 61, 108–143.

[Porcar (2009)] Porcar, M., 2009. Beyond directed evolution: Darwinian selection as a tool for synthetic biology. *Systems and Synthetic Biology* 4 (1), 1 – 6.

[Sakata et al. (2005)] Sakata, T., Yan, Y., Marriott, G., 2005. Optical switching of dipolar interactions on proteins. *Proceedings of the National Academy of Sciences of the United States of America* 102 (13), 4759–4764.

[Sigma-Aldrich Co. (2011)] Sigma-Aldrich Co., 2011. 1',3'-Dihydro-1',3',3'-trimethyl-6-nitrospiro[2H-1-benzopyran-2,2'-(2H)-indole]. Item 273619.

[Silva-Rocha and de Lorenzo (2008)] Silva-Rocha, R., de Lorenzo, V., 2008. Mining logic gates in prokaryotic transcriptional regulation networks. *FEBS Letters* 582 (8), 1237 – 1244, (1) The Digital, Democratic Age of Scientific Abstracts; (2) Targeting and Tinkering with Interaction Networks.

[Smaldon et al. (2010)] Smaldon, J., Romero-Campero, F. J., Trillo, F. F., Gheorghe, M., Alexander, C., Krasnogor, N., 2010. A computational study of liposome logic: towards cellular computing from the bottom up. *Systems and Synthetic Biology (Springer)* 4 (3), 157–179.

[Tamsir et al. (2011)] Tamsir, A., Tabor, J. J., Voigt, C. A., 2011. Robust multicellular computing using genetically encoded nor gates and chemical ‘wires’. *Nature* 469, 212–215.

[Wohl and Kuciauskas (2005)] Wohl, C. J., Kuciauskas, D., 2005. Excited-state dynamics of spiropyran-derived merocyanine isomers. *J. Phys. Chem. B* 109, 22186–22191.

[Wojtyk et al. (2000)] Wojtyk, J. T. C., Wasey, A., Kazmaier, P. M., Hoz, S., Buncel, E., 2000. Thermal reversion mechanism of n-functionalized merocyanines to spiropyrans: a solvatochromic, solvatokinetic, and semiempirical study. *The Journal of Physical Chemistry A* 104 (39), 9046–9055.