Real Time Controller Implementation for an Atomic Force Microscope using a Digital Signal Processor.

Umar Khan
Department of Electrical and Computer Engineering
University of Southampton
Southampton SO17 1BJ
Email: uaak1g10@ecs.soton.ac.uk

Mark French
Department of Electrical and Computer Engineering
University of Southampton
Southampton SO17 1BJ
Email: mcf@ecs.soton.ac.uk

Harold Chong
Department of Electrical and Computer Engineering
University of Southampton
Southampton SO17 1BJ
Email: hmhc@ecs.soton.ac.uk

***Abstract*—The objective of this contribution is to report a low cost implementation of a control system for an atomic force microscope (AFM). AFMs rely on a feedback controller as an integral part of their operation. Currently the controllers shipped with most commercial AFMs are implemented in a closed source manner and provide limited signal access. This restricts the development of novel control strategies by the end users. The solution reported in this contribution uses a commonly available Digital Signal Processor kit to implement a Proportional Integral (PI) controller. This controller is tested on a commercial AFM and experimental results are reported.**

I. Introduction

The development of Atomic Force Microscopes in 1986 by Binning *et.al* [1] initiated a seminal drive that led to the usage of this versatile instrument in a diverse range of fields ranging from material science to microbiology. The primary reason for the wide usage of this instrument is that it can be used to view samples in air, liquid or vacuum with little or no sample preparation requirements. Another factor that increases their applicability is that the sample does not need to be conducting as was the case with their predecessor, the Scanning Tunneling Microscope (STM).

A few salient application examples include the usage of AFMs for the study of biological materials, for instance single molecules like RNA polymerase [2] and cancerous cells [3]. In 2010 Ando *et.al* used an AFM to capture the motion of a Myosin V molecular motor [4]. Recently an AFM has been used for the first time to visualize the structure of DNA double helix [5]. These innovations are pivotal because they enable microbiologists to views biological materials in unprecedented detail which can potentially lead the development of lower cost and higher quality medicines and health care solutions.

Another high impact region where AFMs are being widely used is the study of material interaction forces e.g, adhesion force [6], material interface as in the case of nano contact generation [7]. Lastly AFMs are not only restricted to the study of sample topography, they can also be used for surface alteration e.g. in the case of lithographic processes [8] and nanomanipulation [9].

Although highly applicable, this instrument has two key limitations. Firstly, the image generation process is relatively *slow*, i.e it takes one to four minutes for generating a 256×256 pixel image of an approximately 10×10 μm area. Secondly, the function of an AFM is inherently dependent upon a feedback controller. Each time the AFM is used the quality of the image depends upon the optimality of the control law used. Most commercial AFMs are shipped with PI controllers, which therefore requires that the end user is able to properly tune the controller gains. In addition the time needed for image generation is also dependent on the optimality of the control law. As a consequence large number of researchers have suggested a diverse range of controller synthesis methods ranging from robust H_∞ [10], adaptive control [11] [12] to complex nonlinear controllers [13]. Since the implementation electronics in commercial AFMs is usually closed source, these researchers have mostly implemented their controllers on *off the shelf* real time platforms available from various vendors, e.g National Instruments[14], dSPACE [15] etc. These platforms although easy to use have a relatively higher cost which has consequently led some researchers to develop customized platforms for controller implementation. One possible solution is to implement the controller on a conventional desktop running a real time operation system as has been suggested in [16]. The performance of this solution although good is dependent on the computing capabilities of the desktop implementing the controller [17] and also on the softwares running in the background. The alternative is to use hardware dedicated for controller implementation for instance Digital Signal Processors (DSPs) and Field Programmable Gate Arrays (FPGAs). The key advantage is their relatively low cost and the fact that their performance is independent of the desktop they are interfaced to. Xiaokun Dong *et. al* [18] provides simulation results for implementation of a proportional integral derivative (PID) controller for an AFM using an FPGA.

This contribution reports a complete controller implementation solution using the TMS320C6713 DSP from Texas Instruments available from Digital Spectrum in the form of a kit. A proportional integral (PI) controller is implemented on this kit and tested on a commercial AFM and finally experimental results are reported. Although this DSP has been used previously for controller development [19] no implementation details are provided. Furthermore the current implementation provides a means for changing controller parameters for instance controller gains during real time execution without interrupting the control loop. This enables implementation of computationally intensive control algorithms for instance

adaptive control on a desktop. These can then run in parallel to the real time control loop and update the controller parameters as needed.

The remainder of this paper is organized as follows, section II briefly explains the fundamentals of AFM operation, section III describes the implementation details of the DSP based solution, section IV reports the experimental results and the last section provides conclusions and possible directions for future work.

II. AFM Fundamentals

The setup of a conventional AFM as illustrated in Figure 1 consists of a laser, a photo diode, a cantilever and two piezo actuators namely the *Z* and *Dither* piezo actuators. The cantilevers used in AFMs are generally made of silicon, silicon oxide or silicon nitride and are up to approximately 250 μm in length. The free end of the cantilever facing the sample is attached with a pyramid shaped tip which is used to probe the sample.

The deflection of the cantilever is measured by illuminating the top front end with a laser beam and then measuring the reflected laser radiation using the photo diode. As illustrated in Figure 1 the cantilever base is connected to the dither and Z piezos. The dither piezo is driven with a sinusoidal input voltage f_D, which in turn results in a sinusoidal cantilever tip deflection d_{TM}.

The base of the cantilever is connected to the dither and Z piezos as illustrated in Figure 1. The dither piezo is excited with a sinusoidal drive voltage signal f_D. This results in a sinusoidal cantilever deflection d_{TM}. This deflection signal passes through a lock-in amplifier to extract the amplitude of deflection referred to as A.

The functioning of the AFM then relies on the fact that if the cantilever tip is vibrating *closer* to the sample surface the vibration amplitude will be relatively smaller, whereas this amplitude will be relatively larger *farther* from the sample.

The image of the sample is then generated by moving the entire cantilever piezo actuator assembly over the sample surface in a raster pattern. This is achieved by using separate piezos attached to the cantilever piezo assembly. These piezos (not illustrated in Figure 1) allow the cantilever to move laterally over the sample. As the sample is rastered beneath the vibrating cantilever a feedback controller constantly observes the amplitude A and regulates it to a set point value A_{sp}. If the amplitude reduces due to an increase in sample height for instance due to the presence of *bump* in the sample, the controller causes the Z piezo to move the cantilever base away from the sample. This causes the amplitude to increase back to the set point value A_{sp}. Likewise, a depression in the sample causes an increases in the amplitude which results in the controller causing the Z piezo to move the cantilever base closer to the sample. Finally the controller signal referred to as the *height* signal and denoted by h is used as an estimate of the sample topography. The controller is interfaced with the Z piezo actuator through a piezo amplifier which provides the piezo drive voltage V_z.

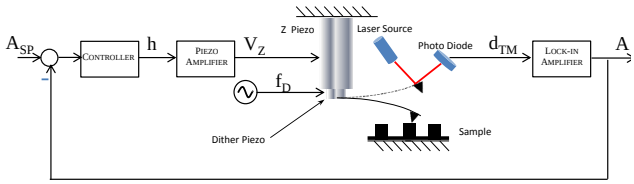

Fig. 1. Atomic Force Microscope setup.

III. DSP Implementation

This section explains the details of the DSP based controller implementation for use with the AFM. Amongst the vast range of DSPs available, the decision for using the TMS320C6713 is motivated by a number of factors namely, relatively low cost, availability in the form of a DSP kit which eliminates the requirement for development of a substantial amount of interface circuitry and reasonably fast clock frequency of 225 MHz. In addition this DSP allows floating point operations which greatly simplifies programming over the previously used fixed point operations. Finally and most importantly, this DSP has a key enabling feature called Real Time Data Exchange (RTDX). This allows data to be exchanged between the host PC and the target DSP *on the fly*, i.e without interrupting real time code execution. This implies that if a PI controller is running on the processor, the controller gains can be changed without halting the processor.

The remaining parts of this section explain in detail the data acquisition components used with the DSP Kit and details of the digital controller data flow.

A. Data Acquisition

This section explains in detail all the circuitry assembled for acquiring data from and sending to the Nanonics AFM. All Nanonics I/O signals are in the range of ± 10 V. Furthermore, it is required that the measurement circuit has an input impedance greater than $10k\Omega$. In addition the circuit that sends signals to the AFM must have an output impedance no greater than 1 $k\Omega$.

Data is read into the DSP kit using the the ADS 8361 ADC from Texas Instruments. This is a 4 channel, 16 bit ADC running at 250×10^3 samples per second (SPS). On the output side the data is written out using the 16 bit DAC 8831 also from the same manufacturer. This DAC has a maximum update rate of 1 MSPS. Both ADC and DAC are provided by Texas Instruments in the form of evaluation boards with substantial amounts of interface circuitry fabricated on them. These boards fit on the DSP kit through an interface board referred to as the 5-6K Interface Board.

Although the evaluation boards have all the necessary components which provide serial data communication between the ADC/DAC chips and the DSP kit, they lack the necessary signal conditioning circuits that will enable connecting the ADC/DAC evaluation boards to the Nanonics AFM. The problem is that the Nanonics AFM amplitude signal varies in a ± 10 V range, however the ADC is unipolar, i.e it accepts voltages in the 0-5 V range. In addition the DAC output needs to be filtered to eliminate the noise present in the AFM amplitude signal A.

To compensate for this two circuits are implemented. The first one is a differential amplifier which multiplies the input voltage with a gain of $\frac{1}{4}$ and adds an offset of 2.5V. The implementation is done using an OPA 132 amplifier from Texas Instruments. This component is selected because of its high input impedance $10^{13}\Omega||6pF$. The exact schematic can be found in the ADC's data sheet provided by Texas Instruments. The DAC output is then low pass filtered using a first order low pass filter of 15 Hz.

B. DSP Controller Data Flow

The entire DSP controller algorithm is *Interrupt Driven*, i.e the processor waits until the ADC has read a data sample and it is ready to be transferred to the processor. The ADC samples the amplitude signal A at the rate of 250×10^3 SPS using timing signals generated by the ADC evaluation module. Once the ADC has read a sample it generates a signal referred to as an *Interrupt*. The processor upon receiving this interrupt runs the control law on the sample and writes the controller output to the DAC. All these steps are explained in detail below and illustrated in the data flow graph in Figure 2.

The first step is to initialize the processor. This includes defining the algorithm variables, enabling interrupts and initializing the serial interface.

Once the processor is initialized it stays in idle state until the ADC generates an interrupt. This initiates execution of the corresponding *Interrupt Service Routine*(ISR). Once control is transferred to the ISR the first step is to read the data from the ADC and store it in a temporary buffer. The next step is to identify the ADC channel at which the sample has been read. The ADC has four input channels (A0,A1,B0,B1) and can either read two channels simultaneously at 250×10^3 SPS, or all four channels simultaneously at 125×10^3 SPS. Currently it is operated in the first mode that allows reading two data channels simultaneously namely A1 and B1. In the current implementation the AFM amplitude signal A is connected to the B1 channel. This channel identification is done by making use of the format in which the data is sent serially from the ADC to the DSP. The ADC transfers the discretized input value to the processor in a frame of 20 bits. The first two bits identify the channel that has been read, the next 16 bits contain the discretized value in two's complement form and the last 2 bits are always 0.

Each time the ADC generates an interrupt the first two bits in the received data frame are read to determine if the data on the current channel has been read, in this case B1. Once this is verified, data bits 3-18 are stored in a data buffer of type INT16. Next the data is converted to a floating variable using a standard C/C++ casting operation. At this stage the processor reads the PI controller gain parameters K_p, K_i using the *Real Time Data Exchange* (RTDX) interface. This is implemented simply by using the non-blocking routine provided by Texas Instruments. The advantage of using a non-blocking routine is that it checks for new data on the RTDX connection and returns control immediately to the ISR, thus allowing continued real time execution. On the transmission side the RTDX channel can be written to directly by Matlab using a very simple interface [20].

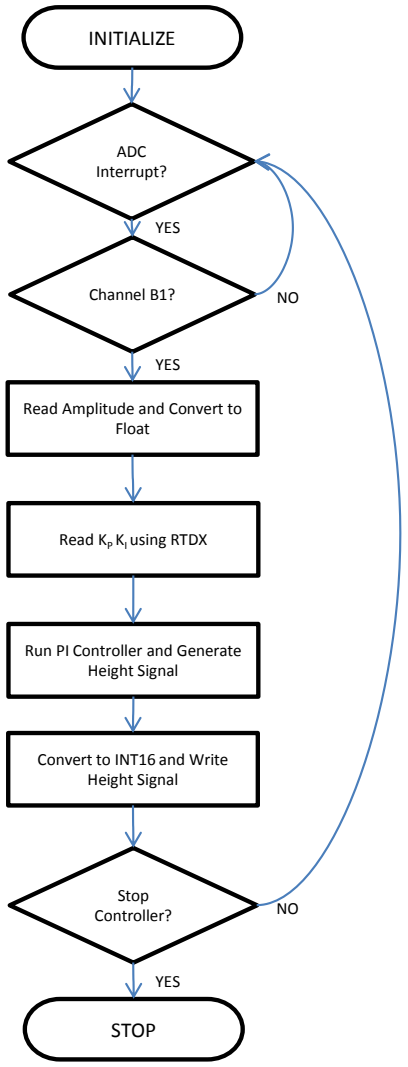

Fig. 2. DSP controller data flow.

Once both controller parameters and the latest sample have been obtained, the controller output is determined using the following control law,

$$\begin{aligned} e(i) &= A_{sp} - A(i) \\ h(i) &= K_p e(i) + K_i T_s \sum_{i=0}^{i} e(i) \end{aligned} \tag{1}$$

where A is the latest value of the amplitude signal just read, A_{sp} is the amplitude set point, T_s is the sample time, e is the error between the set point amplitude and the current value, and i is the discrete time variable. The corresponding controller output h which is the height signal sent to the AFM is generated using the standard PI control law.

Once the controller output is generated it is converted to the INT16 data type again and written to the DAC through

the serial interface. The DAC is a single channel device so no channel identification bits need to be added. The only modification that needs to be made is that since the ADC sends the data in two's complement form, this must be compensated for by executing a XOR operation. Before the data is written to the DAC it is XORed with 0×8000 and then sent to the DAC.

Figure 3 illustrates the final functional block diagram of the entire DSP based controller setup. While the DSP kit runs the controller, a separate data acquisition card (DAQ) is used to observe the AFM Height and Amplitude signals. The DAQ used in the current implementation is the National Instruments USB 6356 card which samples both Height and Amplitude signals at 1×10^6 SPS simultaneously. The terms McBSP 0 and McBSP 1 refer to the Multichannel Buffered Serial Ports used for serial communication between the ADC/DAC and the processor.

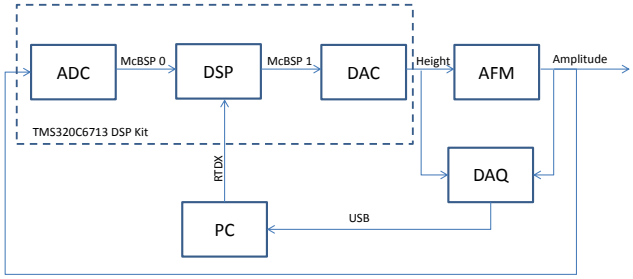

Fig. 3. DSP controller block diagram.

IV. Experimental Results

This section presents the results of scanning two samples using an AFM that uses the DSP based controller described in the previous section. The AFM in this case is the Nanonics Cryo View 2000 from Nanonics Imaging. Each image shows a scanned area of $3 \times 3\mu m$ and consists of 256 *trace* lines and 256 *retraces* lines. Here the terms trace and retrace refer to the consecutive scan lines in the raster scan pattern. Each scan line requires one second to complete therefore the scan rate is 0.5 Hz.

The PI gains were tuned using an automated PI controller tuning algorithm currently development. The controller tuning method is based on the Estimation Based Multiple Model Adaptive Control (EMMSAC) algorithm [21]. Due to the computationally intensive nature of this algorithm, it is run only three times at equally spaced intervals during each scan starting after the first scan line. Each time the EMMSAC algorithm requires 4 minutes and 50 seconds to determine the best gains on an Intel 2.67 GHz desktop. The total time required to generate each image is approximately 27 minutes. The two samples scanned in this contribution are namely the TGZ 02 calibration grid manufactured by Mikromasch and digital video disc (DVD) sample from TDK.

Figures 4 and 6 provide the images obtained. The TGZ 02 calibration calibration grid consists of SiO_2 steps on Si wafer with a height of 120 nm and pitch of 3μm. In Figure 4 a single step can be viewed. Although a greater scan area is preferable so that a higher number of steps may be viewed, however the current AFM has a maximum permissible scan area of approximately $3 \times 3\mu$m. Figure 6 shows the tracks on the DVD sample with a pitch of approximately 0.74μm as mentioned in [22].

Figures 5 and 7 provide the sample heights as obtained during a single trace and retrace line. Although the trace and retrace signals are not completely noise free and there is an error of approximately 20nm in the estimated height for the TGZ 02 sample, they are similar as is the requirement for a properly tuned controller and reveal the topography shape clearly.

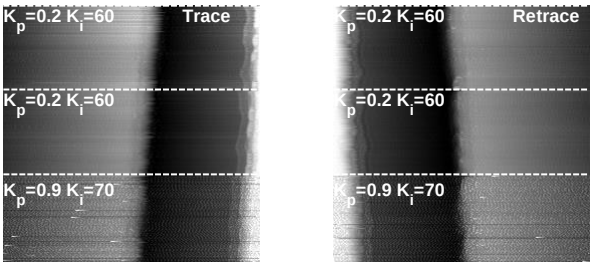

Fig. 4. TGZ calibration sample image.

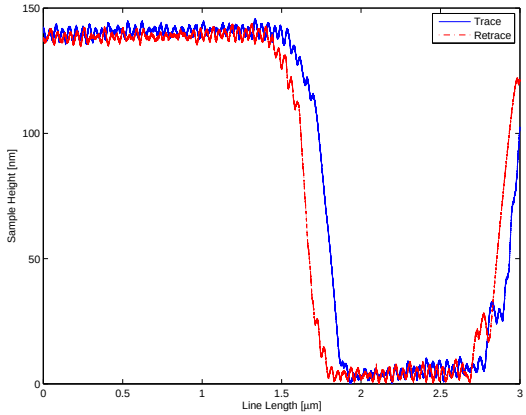

Fig. 5. TGZ calibration sample line image.

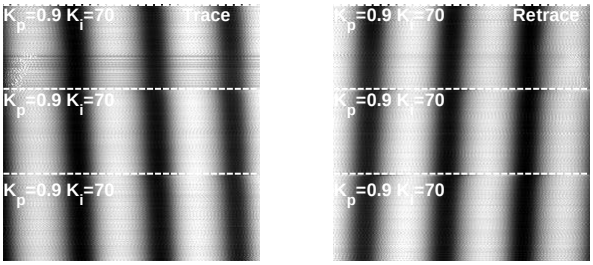

Fig. 6. DVD sample image.

V. Conclusions

This contribution provides a detailed implementation method for an enabling technology, which can save valuable development time and allow experimentation with novel controller designs at a minimal cost. This implementation can be

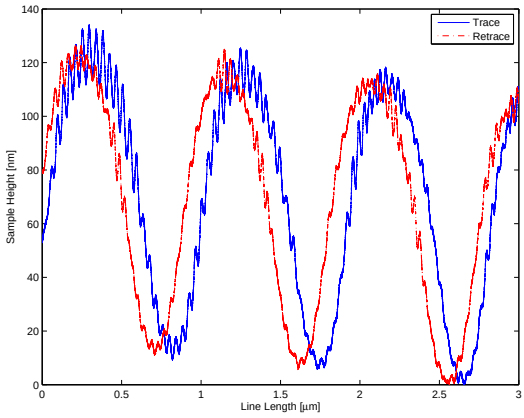

Fig. 7. DVD sample line image.

extended to multiple input multiple output controllers by using multichannel DACs. Furthermore the DSP memory can be augmented by using external memory modules thus allowing the implementation of learning controllers. These controllers improve their performance during the scan by using the data collected from the previous scan lines. In the future it needs to be explored if the system noise can be reduced and also if faster data converters with greater sampling rate can used to improve the accuracy of the estimated topography.

REFERENCES

[1] G. Binning, C.F. Quate and Ch. Gerber "Atomic Force Microscope", *in Physical Review Letters*, 1986, pp. 930.

[2] M. Guthold, M. Bezanilla, D.A. Erie, B. Jenkins, H.G. Hansama and C. Bustamante, "Following the assembly of RNA polymerase-DNA complexes in aqueous solutions with the scanning force microscope,1994,pp.12927 - 12931.

[3] S. Iyer, R. M. Gaikwad, V. Subba-Rao1, C. D. Woodworth and I. Sokolov, "Atomic force microscopy detects differences in the surface brush of normal and cancerous cells", *in Nature Nanotechnology*, 2009, pp. 389-393

[4] N. Kodera, D. Yamamoto, R. Ishikawa and T. Ando, "Video imaging of walking myosin V by high-speed atomic force microscopy",*in Nature*, 2010, pp.72-76

[5] C. Leung, A. Bestembayeva, R. Thorogate, J. Stinson, A. Pyne, C. Marcovich, J. Yang , U. Drechsler, M. Despont, T. Jankowski, M. Tschpe and B. W. Hoogenboom, "Atomic Force Microscopy with Nanoscale Cantilevers Resolves Different Structural Conformations of the DNA Double Helix",*Nano Letters*, Vol. 12, Issue 7, pp 38463850, 2012

[6] O. Dos, S. Ferreiraa, E. Gelinckb, D. de Graafb and H. Fischer, "Adhesion experiments using an AFMParameters of influence", *in Applied Surface Science*, 2010, pp. 48-55.

[7] M. Grodzicki, S. Smolareka, P. Mazura, S. Zubera and Antoni-Ciszewski, "Characterization of Cr/6H-SiC(0 0 0 1) nano-contacts by current-sensing AFM", *in Applied Surface Science* , 2009, pp. 1014-1018

[8] W. J. Dauksher, K. J. Nordquist, N. V. Le, K. A. Gehoski, D. P. Mancini, D. J. Resnick, L. Casoose, R. Bozak, R. White, J. Csuy and D. Lee, "Repair of step and flash imprint templates" , *in Journal of Vacuum Science and Technology B: Microelectronics and Nanometer Structures* , 2004, pp. 3306-3311

[9] Q. Yang and S. Jagannathan, "Atomic force microscope based nanomanipulation with drift compensation" , *in International Journal of Nanotechnology*, 2006, pp. 527-544

[10] A. Sebastian, "Nanotechnology: A Systems and Control Approach", Ph.D. Dissertation, Depatrment of Electrical Engineering, Iowa State University, 2004.

[11] D. Y. Abramovitch, S. Hoen, R. Workman,"Semi-Automatic Tuning of PID Gains for Automatic Force Microscopes", *in 2008 American Control Conference*, 2008.

[12] O. M. El Refai and K. Y. Toumi, "On Automating Atomic Force Microscopes: An Adaptive Control Approach", *in 43rd IEEE Conference on Decision and Control*, 2004.

[13] S. Misra, H. Dankowicz and M.R. Paul, "Event-driven Feedback Tracking and Control of Tapping-mode Atomic Force Microscopy", *in Proceedings of the Royal Society*, 2008.

[14] D. J. Burns, "A Systems Dynamics Approach to User Independence in High Speed Atomic Force Microscopy", Ph.D. dissertation,Dept. of Mech. Eng., Massechusetts Institute of Technology, 2008.

[15] G. Schitter, F. Allgoewer and A. Stemmer, "A new control strategy for high-speed atomic force microscopy",*Nanotechnology*, Vol. 15, pp. 108-114, 2008

[16] V. Chandrasekhar and M.M. Mehta, "RTSPM: Real-time Linux control software for scanning probe microscopy", *Review of Scientific Instruments*, Vol. 84, Issue 1, 2012.

[17] G. Aloisi, F. Bacci, M. Carla, D. Dolci and L. Lanzi, "Implementing on a desktop computer of the real time feedback control loop of a scanning probe microscope", *Review of Scientific Instruments*, Vol. 79, Issue 11, 2008.

[18] Y. Sun, Y. Fang, Y. Zhang and X. Dong, "Field Programmable Gate Array (FPGA) Based Embedded System Design for AFM Real-Time Control", *IEEE International Conference on Control Applications*, 2010.

[19] C. Lee, S. M. Salapaka,"Fast Robust Nanopositioning- A Linear-Matrix-Inequalities-Based Optimal Control Approach", *IEEE/ASME Transactions on Mechatronics*, Vol. 14, No. 4, 2009.

[20] T. Shen, "Using RTDX with a MATLAB GUI", [Online]. Available: http://cnx.org/content/m14388/latest/

[21] D. Buchstaller, "Robust Stability and Performance Multipe Model Switched Adaptive Control", Ph.D. Dissertation,Department of Electrical and Computer Engineering, University of Southampton, 2010.

[22] K. Tatebe, "Optical Data Storage", [Online]. Available:budker.berkeley.edu/Physics208/