Dual-Stylus-Arm Scratch Drive Micro-robots
Controlled by an Onboard Parallax Algorithm

Mark G. Arnold
Lehigh University
Bethlehem, Pennsylvania, USA
marnold@cse.lehigh.edu

Jung H. Cho
Lehigh University
Bethlehem, Pennsylvania, USA
jung.cho@lsi.com

Abstract — A novel approach to controlling the turning operation of MEMS Scratch-Drive-Actuator (SDA) micro-robots has been developed. The operation of MEMS SDA has been well demonstrated by the research of Donald et al. [1][2][3]. An improvement of adding an additional stylus arm to control left and right rotation as well as using both arms to halt is discussed. In order to eliminate the complication of different stress curling to control multiple micro-robots, an alternative solution of controlling electrical connections between the parallel plate body and stylus arms is presented. By applying this new control, a novel onboard parallax algorithm was developed that allows a micro-robot to move towards a target without any external control.

Keywords: *MEMS, SDA, micro-robot*

I. INTRODUCTION

There has been much research to study and develop MEMS Scratch Drive Actuators (SDAs). The application of such MEMS is common in the fields of mirrors, optical gratings, variable capacitors, and accelerometers. By utilizing this extensive research on electrostatic actuation, an untethered micro-robot has been developed [3]. We can envision fabricating such a micro-robot which is capable of interacting with similar robots to engage in larger tasks. Unlike the multiple-robot system of [1] in which all robots obtain motion commands from a single external signal, we assume robots perform simple algorithms autonomously, and that the interaction of many such robots allows the performance of a complex task. The MEMS micro-robot built by Donald et al. [2] has a dimension of $60\mu m$ by $250\mu m$ by $10\mu m$. Figure 1 shows the structure of this device proposed in [2], which propels itself using scratch-drive actuation [3]. The use of a stylus steering arm was introduced to provide single-direction turning capability. The power is delivered externally through electrodes which can be multi-voltage level encoded [1][2] to take advantage of hysteresis built into the design of each SDA (chip dimensions) to control forward and turning motions. Our approach is different. We assume each robot will have identical SDA dimensions, and use digital logic to implement the local control algorithm that steers the robot. Combining the SDA and digital logic on a single chip is a fabrication challenge. The most widely used CMOS-MEMS-integration process currently is a hybrid approach of a modular assembly of CMOS and MEMS devices [10]. The unfortunate consequence of this is low performance due to assembly and packaging cost. Monolithic integration is available and can be used to integrate SDA and digital circuits. One of the approaches recommended by Witvrouw [10] is to process the integrated circuit first and the MEMS last and typically on top of the circuitry. This approach will allow a sound integration plan to build a low-cost MEMS micro-robot.
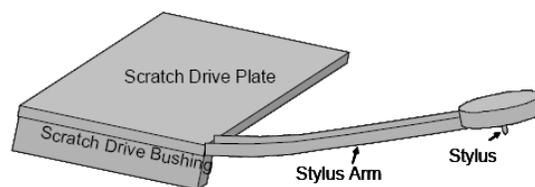
Figure 1. Illustration of MEMS Micro-robot [2].
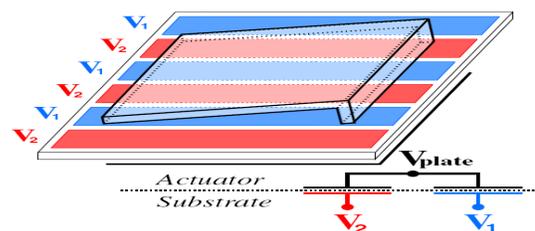
Figure 2. Illustration of SDA on power grid [2].

II. Dual-Stylus SDA

Figure 2 shows the schematic of the capacitively-coupled power grid used by Donald et al. [2]. By using these electrodes, the SDA is attracted to the electrodes when external high voltage is applied, and jumping like a spring when the voltage is removed. Therefore, in order to be propelled, a clock-like voltage waveform has to be applied [2]. In our novel approach, the voltage on the scratch drive plate can also be used to supply power to onboard digital logic as shown in Figure 3. We have demonstrated this by developing a Verilog-A model of the SDA and applying voltage regulation to provide adequate voltage swing for 1V 40nm CMOS standard-cells from the plate voltage as a power [8].
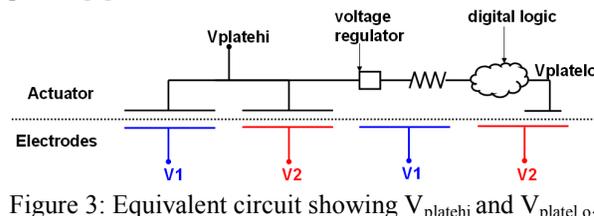
Figure 3: Equivalent circuit showing $V_{platehi}$ and $V_{platel.o}$.

In order to change direction, Donald's SDA [2] uses a stylus steering arm and requires higher voltage to achieve pull-in or snap-down voltage [2] necessary to make contact with the substrate. This requirement introduces multi-voltage level encoded power waveform which is used by Donald's SDA. In order for many SDAs to interact as shown in Donald et al. [1], all of them need to have different stresses applied during fabrication so that the stylus arms can curl differently in order to vary the pull-in voltage [11]. Our novel approach is to apply a switch or a large transistor to control the conductivity between the stylus arm and the parallel plate body. This eliminates the step needed for different stress curling, and one voltage waveform can be used to supply power to different SDA micro-robots. We also added a second arm, as seen in Figure 4, to be able to turn left and right, and use both arms to enforce a stationary position.

(a)

(b)
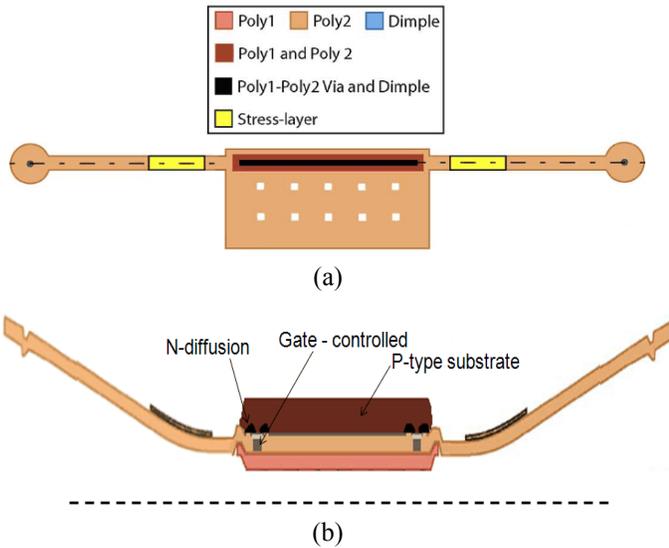
Figure 4. (a) Top view of proposed SDA modification from [2] in order to perform both left and right turn. (b) Frontal view of proposed SDA modification from [2] in order to control pull-in/snap-down voltage of the stylus steering arms.

The modified SDA consists of 3 components which are left and right stylus beams and parallel-plate capacitor body. Figure 5 shows the high-level illustration of this structure. The capacitance across the parallel plate is the most dominant energy storage part of this circuit. Since the V1 and V2 electrodes are uniformly covering the whole area of the parallel plate, the voltage across the plate can be summarized as [3],

$$V_{plate} = \frac{V_1 C_1 + V_2 C_2}{C_1 + C_2} \qquad (1)$$

When both arms are connected to the parallel plate body, the voltage across the beams ($V_{left-arm}$ and $V_{right-arm}$ in Figure 6), would be same compared to V_{plate}.
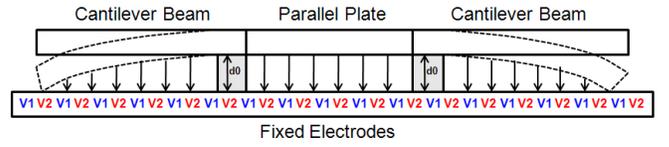
Figure 5. Deformation in cantilever and parallel plate due to applied voltage

When the beams are electrically disconnected, then each stylus beam becomes a well-known MEMS cantilever beam [11][12]. We applied the modeling technique and equations from Wei's research [12] to build a Verilog-A model to work with our previously developed model of SDAs [8]. Figure 6 shows a different voltage across the beams and the plate when arms are disconnected. Under an ideal fabrication process, $V_{left-arm}$ and $V_{right-arm}$ would be identical.
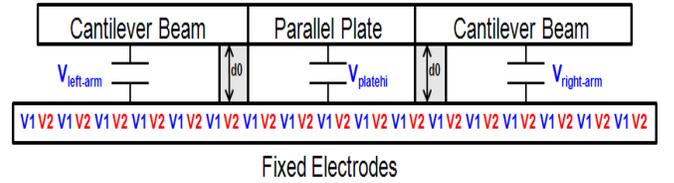
Figure 6. Illustration of different voltage across the beams and the plate capacitor due to disconnection from the parallel plate body.

Whether $V_{left-arm}$, $V_{right-arm}$ and/or V_{plate} exceed the pull-in voltage needed to cause attraction, defined in [11] as

$$V_{PI} = \sqrt{\frac{8 k d_0^{\ 3}}{27 \varepsilon_0 A}} \qquad (2)$$

determines whether left, right, forward or no motion occurs.. Here k is the spring constant, 'd_0' is the initial gap height and A is the area coverage of the cantilever. Previous work [1] chose V_{PI} based on careful selection of these parameters, k and d_0. We show that these may remain constant and by simply connecting and disconnecting the cantilever arms from the parallel plate body electrically, we can control the V_{PI}. We use this control in section V to present a parallax algorithm which can guide the micro-robots with an onboard algorithm that does not need external control.

III. Intermittent Power and Magnetic Tunnel Junction Non-Volatile Flip-Flop

As described in the previous section, a MEMS SDA micro-robot is driven using external voltage and by using the electrodes underneath to create an electrostatic field to cause actuation, which is transferred into forward or turning motions. This external voltage is applied in clocked fashion, up to perhaps 10 KHz, which we will refer to as a major cycle. Since the voltage is applied intermittently, there needs to be a solution to hold important states needed for

continuous operation of the onboard logic. This led to applying non-volatile flip-flops developed using Magnetic Tunnel Junction (MTJ) technology [4]. This flip-flop works like a standard flip-flop but information is stored in MTJs; therefore, when the SDA's major power cycle occurs, the MTJ flip flop restores to its previously saved state. In order to demonstrate this we simulated the MTJ flip-flop [4] and developed a Verilog-A model of a dynamic storage behavior of MTJ. We then simulate this MTJ flip-flop in Cadence AMS environment to co-simulate both transistors and Verilog-A/Verilog-AMS models. Figure 7 shows the actual circuit of MTJ flip-flop [4] simulated in section IV.

Figure 7. Schematic of MTJ flip-flop.

IV. Simulation Result

In order to demonstrate operation of dual-stylus SDA, we have developed a Verilog-A model to capture the voltage across the beams and the parallel plate. First, we verified that V_{plate} can still be used to supply power to onboard CMOS digital logic [8] when two arms are connected and disconnected. We have chosen a 4-bit counter to demonstrate the operation along with storage behavior of the MTJ flip-flop from section III. A 4-bit counter Verilog-RTL was synthesized into 40nm standard cells using positive edge flip-flops and they were replaced with a MTJ non-volatile flip-flop Verilog-A/AMS model. We then integrated all the models to simulate in the Cadence AMS environment. The setup is shown in Figure 8. This type of system-level simulation utilizing Verilog-A/Verilog-AMS models has been accepted in research [9].

Figure 8. Top-level simulation setup.

Figure 9 shows the simulation result of the counter. The first waveform shows the intermittently-applied and voltage-regulated output to drive the ring oscillator and the counter. The next 4 waveforms are the output of the counter x[3:0]. As the power was removed, the counter state was saved in the MTJ flip-flop as 0011. When the power returned, it restored the state 0011 and continued counting. This result provided assurance that more elaborate state machines can be pursued with this architecture.

Figure 9. Simulation result of 4-bit counter.

Then, we used the counter output bit x[3] to control the left stylus and inversion of x[3] to control the right stylus. Figure 10 shows the voltage across $V_{left-arm}$ and $V_{right-arm}$ changing as x[3] toggles. As expected, V_{plate} remained nearly constant during the x[3] change since V_{plate} has the most capacitance to hold the charge.

Figure 10. $V_{left-arm}$, $V_{right-arm}$ and V_{plate} result.

V. Parallax

Each micro-robot has no knowledge of the other robots, except via communication using some sort of sensors. We assume some sort of global communication, such as RF carbon nanotubes [13] or global optical communication, is possible. This global channel is used only while robots are navigating towards their destinations.

The more important inter-robot communication medium uses two very-short-range LEDs (like headlights on a car), together with a parallax-based algorithm described below that senses distance as the micro-robot rotates (like radar)

[14]. The left and right LEDs each flash for a brief period. If the narrow beams from the first robot's LEDs strike a photosensitive edge on another target robot, that robot can use the global communication channel to inform the original robot that it was hit. We assume, as is the case for the micro-robot proposed by Donald et al. [2], that the first robot is able to rotate around a stationary point on the 2D surface. At each step of the rotation, the first robot alternately flashes its right and left LEDs and listens for a ping from the target robot. Measuring the angles that initiate pings from the left and right LEDs provides enough information, in theory, to resolve the relative position of the first robot in relation to the target. Once the robot has maneuvered near the target, inter-robot communication without the use of the global RF channel will be more efficient because all of the energy output from the LED will be focused at its neighbor. We can obtain reasonable precision inter-robot position information since the two LEDS will be on the order of 200µm apart, and the angular resolution of the SDA-based robot proposed by Donald et al. [2] is on the order of a degree.

A. Parallax Theory

Working through the trigonometry for parallax is quite involved. For simplicity in this derivation, we will assume micro-robots are measured in 100µm units. For the 200µm size discussed previously, each side of a square micro-robot in this model is two units. The edges of this square will be photo sensors to receive the pings from other robots' LEDs; the interior of this square will hold all the digital logic and motion transducers (a large SDA for motion and two stylus arms). Assume the moving robot's center of rotation is at (x,y) and the target edge (presumably of another robot) is located between $(-1,0)$ and $(1,0)$. We define the following:

$$h_R(x,y) = \sqrt{((x-1)^2+y^2)} = \sqrt{(x^2+y^2-2x+1)} \qquad (3)$$

$$h_L(x,y) = \sqrt{((x+1)^2+y^2)} = \sqrt{(x^2+y^2+2x+1)} \qquad (4)$$

$$d_R(x,y,a) = \sqrt{(h_R^2 - a^2)} = \sqrt{(x^2+y^2-2x+1-a^2)} \qquad (5)$$

$$d_L(x,y,a) = \sqrt{(h_L^2 - a^2)} = \sqrt{(x^2+y^2+2x+1-a^2)} \qquad (6)$$

$$\phi_R(x,y,a) = \tan^{-1}(a/d_R(x,y,a))$$
$$= \tan^{-1}(a/\sqrt{(x^2+y^2-2x+1-a^2)}) \qquad (7)$$

$$\phi_L(x,y,a) = \tan^{-1}(a/d_L(x,y,a))$$
$$= \tan^{-1}(a/\sqrt{(x^2+y^2+2x+1-a^2)}) \qquad (8)$$

Here $h_R(x,y)$ and $h_L(x,y)$ are the distances from (x,y) to the left and right target edges; $d_R(x,y,a)$ and $d_L(x,y,a)$ are the distances from a LED which is a units away from the center of rotation to the left and right target edges; $\phi_R(x,y,a)$ is the angle in a right triangle with opposite side of size a, hypotenuse of $h_R(x,y,a)$ and adjacent side of $d_R(x,y,b)$; and ϕ_L is the equivalent angle for the left target side.

There are three kinds of robots that we can model with (3)-(8). The simplest kind, which we have considered in our earlier research [15], puts the center of rotation at the center of the robot, (x_c, y_c). This symmetrical arrangement allows rotation either to the left or right and makes the

implementation of a parallax algorithm easier, but is not easily realized with SDAs. The other possibilities have the center of rotation chosen to be at either the left or right side, as is the case with the single-stylus-arm SDA micro-robot fabricated by Donald et al. The direction of rotation for a single-stylus-arm SDA is chosen at fabrication time depending on which side the stylus arm is attached to. Table 1 describes how to model these three options, assuming the center of the robot at (x_c, y_c) initially faces $\theta_c = 0°$:

rotate	Left LED Angles		Right LED Angles	
both	$\phi_L(x_c, y_c,1)$	$\phi_R(x_c, y_c,1)$	$\phi_L(x_c, y_c,1)$	$\phi_R(x_c, y_c,1)$
right	$\phi_L(x_c, y_c-1,2)$	$\phi_R(x_c, y_c-1,2)$	$\phi_L(x_c, y_c-1,0)$	$\phi_R(x_c, y_c-1,0)$
left	$\phi_L(x_c, y_c+1,0)$	$\phi_R(x_c, y_c+1,0)$	$\phi_L(x_c, y_c+1,2)$	$\phi_R(x_c, y_c+1,2)$

Table 1. Modeling of different rotation centers.

The top line of the table deals with the symmetrical robot [15] (that can rotate both directions), which is not the focus of this paper. The novel micro-robot proposed in this paper has two stylus arms, which are controllable from an onboard hardware algorithm, that allow the robot to rotate either to the left or right, albeit with different centers of rotation, as described in the bottom two lines of the table.

The angles from the right LED in Table 1 to each target end are $\phi_L + \theta_L$ and $\phi_R + \theta_R$, where θ_L and θ_R are the angles between the x-axis and the respective hypotenuse. From this, is easy to calculate the relative angle subtended by the target as viewed by the robot from the right (δ_R) **LED**:

$$\delta_R = 180° - (\theta_R + \phi_R + (180° - (\theta_L + \phi_L)))$$
$$= \theta_L + \phi_L - (\theta_R + \phi_R). \qquad (9)$$

Similar triangles are involved for the left (δ_L) **LED**, except the angles from the left LED are: $\theta_L - \phi_L$ and $\theta_R - \phi_R$.

When the robot rotates left, it uses its right LED to measure the angle, $\delta_R(x,y,a_R)$ subtended by the target visible from the robot's right side at the current center of rotation, (x,y). At the rightmost, the robot's right LED hits $(-1,0)$; at the leftmost, the robot's right LED hits $(1,0)$. Thus, the angle viewed from the right LED is

$$\delta_R(x,y,a_R) = \theta_L - \phi_L(x,y,a_R) - (\theta_R - \phi_R(x,y,a_R)). \qquad (10)$$

The robot can observe $\delta_L(x,y,a_R)$ and a similar $\delta_L(x,y,a_l)$ approximately by counting rotation steps that produce pings after the respective LEDs are activated. Since θ_L and θ_R do not depend on a_L or a_R, the difference, $\delta(x,y,a_L,a_R)$, is:

$$\delta(x,y,a_L,a_R) = \delta_L(x,y,a_L) - \delta_R(x,y,a_R)$$
$$= \theta_L + \phi_L(x,y,a_L) - (\theta_R + \phi_R(x,y,a_L)) - \qquad (11)$$
$$(\theta_L - \phi_L(x,y,a_R) - (\theta_R - \phi_R(x,y,a_R)))$$
$$= \phi_L(x,y,a_L) - \phi_R(x,y,a_L) + \phi_L(x,y,a_R) - \phi_R(x,y,a_R).$$

Since here $\phi_R(x,y,0)=0$ and $\phi_L(x,y,0)=0$, (11) can simplify, i.e., $\delta(x,y,2,0) = \delta(x,y,0,2) = \phi_L(x,y,2) - \phi_R(x,y,2)$. Figure 11(a) shows $\delta(x,y,2,0)$ in degrees for $-10 < x < 10$ and $4 < y < 9$, assuming the angles could be measured without error.
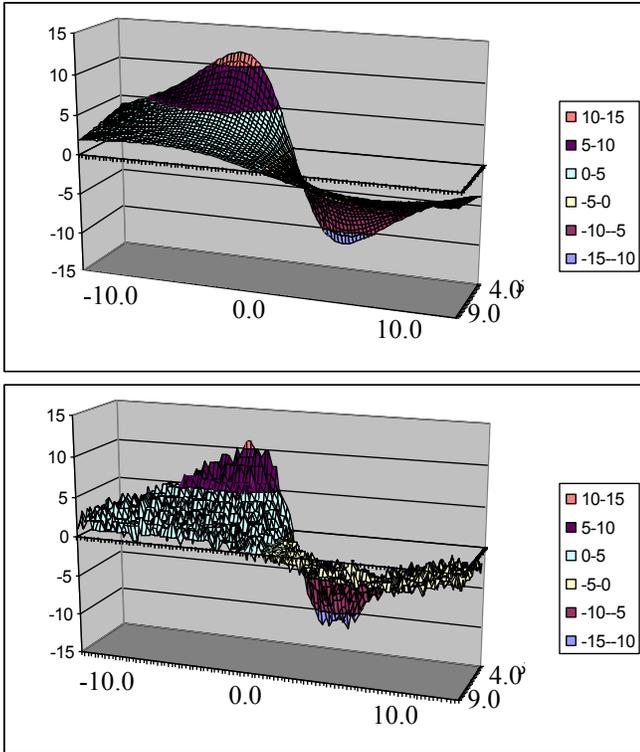
Figure 11. a) Predicted and b) Observed angles.

Absolute angles increase towards the x-axis. If $\delta > 0$, the target is closer to the left LED. If $\delta < 0$, the target is closer to the right LED. If $\delta = 0$, the robot is heading directly towards the target. The observed δ (Figure 11 b) is good enough to control the robot without expensive onboard trigonometry.

B. Verilog Digital Logic Simulation

This concept has been verified through simulation of a state machine implementing the parallax algorithm using the Verilog hardware description language. We choose this language for simulation as it is the preferred choice for modeling digital designs and state machines [6]. In order to simulate micro-robots, Verilog must model the 2D motion environment that the micro-robots operate in, a task for which Verilog was not designed to do. We developed about 2,000 lines of Verilog code, using the C-like features that are available, including integer arrays that allow us to store the scaled x, y and theta for each robot along with bit arrays that describe the state of each robot's sensors, LEDs, etc. Verilog also provides a scalar (not array) real type to perform the distance calculations needed to model the 2D geometry. Although the state machines described in Verilog do not need real numbers or trigonometric calculations to implement the parallax algorithm, the testbench environment that we developed must perform extensive calculations of this nature to simulate what would happen with physical micro-robots. For example, given the robot's x_c, y_c and θ_c, the simulation models rotation as:

$$x_c \leftarrow x_c + r\,(\cos(\theta_c + 90^o\mathrm{sgn}(\Delta)) - \cos(\theta_c + 90^o\mathrm{sgn}(\Delta) + \Delta))$$
$$y_c \leftarrow y_c + r\,(\sin(\theta_c + 90^o\mathrm{sgn}(\Delta)) - \sin(\theta_c + 90^o\mathrm{sgn}(\Delta) + \Delta))$$
$$\theta_c \leftarrow \theta_c + \Delta, \qquad\qquad (12)$$

where r is the constant distance from (x_c, y_c) to (x, y); and the rotation step, Δ, is positive for left (counterclockwise) rotation and negative for right (clockwise) rotation. Since Verilog does not provide built-in trigonometric functions, we use a package to implement these [7].

Figure 12 shows a pseudo-code fragment that uses parallax to guide one robot to approach another one. This parallax algorithm only uses integer addition, subtraction and comparison—it is well suited for economical hardware implementation. The SDAs may rotate either to the LEFT (counterclockwise about the left stylus arm) or RIGHT (clockwise about the right stylus arm), based on the value in the bit variable direction. The algorithm stops when the sum $\delta_L + \delta_R$ exceeds a constant STOPANGLE. Choosing STOPANGLE to be about 75° leaves the robot within one robot-length away from the target. Although not as well aligned as symmetrical micro-robots would be, this novel version of our parallax algorithm for dual-stylus-arm micro-robots leaves the micro-robots well positioned to establish local optical communication.

```
oldHitL = 0; oldHitR = 0;
deltaL = 0;  deltaR = 1;
delta = 0;   direction = LEFT;
while((deltaL+deltaR<=STOPANGLE)
     |(abs(delta)>1))
{
 finishL = -1;  finishR = -1;
 angle = 1;
 while (finishR==-1||finishL == -1)
 {
  rotate towards direction;
  turn left LED on and right LED off;
  hitL = test for RF ping;
  turn left LED off and right LED on;
  hitR = test for RF ping;
  turn right LED off;
  if ((hitL==1) & (oldHitL==0))
    startL = angle;
  if ((hitL==0) & (oldHitL==1))
    finishL = angle;
  if ((hitR==1) & (oldHitR==0))
    startR = angle;
  if ((hitR==0) & (oldHitR==1))
    finishR = angle;
  oldHitL = hitL; oldHitR = hitR;
  angle = angle + 1;
 }
 deltaL = finishL - startL;
 deltaR = finishR - startR;
 delta = deltaL - deltaR;
 turn on LED that points toward direction
 if(~(((delta<=0)&(direction==RIGHT))|
    ((delta>=0)&(direction==LEFT))))
      while (no RF ping)
        continue rotating towards direction
 direction = opposite direction;
}
```

Figure 12. Verilog Code for Docking.

The parallax algorithm uses separate variables for information gathered from left and right illumination, which allows complete information to be gathered in one sweep (either clockwise or counterclockwise). The inner loop notes the angles where pings start and finish under both left and right illumination. When the inner loop reaches the end of either a clockwise or a counterclockwise sweep, the algorithm makes a decision whether it should continue rotating nearly 360° to find the target again, or begin rotating in the opposite direction (about the center of the opposite stylus arm). If the robot continues rotating in the same direction, the center of the robot remains stationary; if it changes rotation direction, the robot moves slightly towards the side of the old center of rotation. The goal of the algorithm is to move in the direction that points to the more distant edge of the target, which means the robot should switch rotation direction at the end of a sweep that rotated away from pointing to the closer edge. For example, if the algorithm discovers that $\delta < 0$ (meaning the right LED is likely closer to the target) after sweeping RIGHT, it reverses rotation direction at that moment because it is now pointing to the left (further away) edge of the target.

To help us understand the results of the simulation, our testbench generates postscript for the poise and location of the robots being simulated. Figure 13 shows a snapshot (out of hundreds produced by Verilog in one particular simulation when robot "4" is heading towards robot "0"). The small gaps in each square represent the right and left LEDs. When a LED is on, it is shown as a line. When such a beam hits the edge of another robot, it is shown as a wider edge on that square.
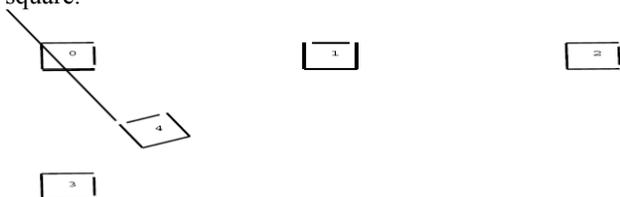
Figure 13. Postscript Simulation Output.

VI. CONCLUSION

We have proposed a novel way of controlling the turning capability of MEMS SDAs. By adding an additional stylus arm the robot can now turn both left and right as well as use both arms to stop. Fabricating a transistor connection between each stylus and the parallel plate body allows the micro-robot to control the pull-in voltage. We have demonstrated the control capability of this pull-in voltage by modeling the parallel plate SDA and the cantilever beams in Verilog-A. Using this control capability, we also presented a parallax algorithm which is suitable to operate onboard this dual-stylus micro-robot architecture. We have simulated this algorithm with the assumption of a single robot seeking a single target, with all other robots silent. Generalizing to multiple robots and targets would require time multiplexing the global communication channel, which is quite feasible

considering major (motion) cycles are much slower than the internal digital-logic clock cycle.

We envision that additional advancements in our research will lead to applying these micro-robots in diagnosis and treatment of dermatological conditions [16]. Another area of interest is a sensor-based application where these micro-robots carry different MEMS payloads to create a sensor network able to be configured for different tasks [17].

REFERENCES

[1] B.R. Donald, et al., "Planar Microassembly by Parallel Actuation of MEMS Microrobots," IEEE *Journal of Microelectro-mechanical Systems*, vol. 17, Aug. 2008.

[2] B.R. Donald, et al., "An Untethered, Electrostatic, Globally Controllable MEMS Micro-Robot," IEEE *Journal of Microelectro-mechanical Systems*, vol. 15, Feb. 2006.

[3] Bruce R. Donald et al., "Power Delivery and Locomotion of Untethered Micro-actuators," IEEE *Journal of Microelectro-mechanical Systems*, vol. 12, Dec. 2003.

[4] W. Zhao, E. Belhaire, C. Chappert, "Spin-MTJ based Non-Volatile Flip-Flop," *International Conference on Nanotechnology*, Hong Kong , Aug. 2007.

[5] M. G. Arnold, *Verilog Digital Computer Design: Algorithms into Hardware*, PTR Prentice Hall: Upper Saddle River, NJ, 1999.

[6] M. G. Arnold and J. D. Shuler, "A Synthesis Preprocessor that Converts Implicit Style Verilog into One-hot Designs," *Sixth International Verilog HDL Conference*, Mar. 1997.

[7] M. G. Arnold et al., "Verilog Transcendental Functions for Numerical Testbenches," *Tenth International HDL Conference*, Santa Clara, California, Mar, 2001.

[8] J. H. Cho and M. G. Arnold, "Powering Embedded CMOS Logic on MEMS-based Micro-Robots," *IEEE International Behavioral Modeling and Simulation Conference*, Sept. 2009.

[9] L. Mateu and F. Moll, "System-level Simulation of a Self-powered Sensor with Piezoelectric Energy Harvesting," *International Conference on Sensor Technologies and Applications*, 2007.

[10] A. Witvrouw, "CMOS-MEMS Integration: Why, How and What?," *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, pp. 826-827, 2006.

[11] S.C. Saha et al., "Modeling of Spring Constant and Pull-down Voltage of Non uniform RF MEMS Cantilever," *IEEE International Behavioral Modeling and Simulation Workshop*, pp. 56-60, 2006.

[12] L.C. Wei et al., "Analytical Modeling For Determination Of Pull-In Voltage For An Electrostatic Actuated MEMS Cantilever Beam," *Proceedings of ICSE,* 2002.

[13] T. Fukuda et al., "Perspective of Nanotube Sensors and Nanotube Actuators," *4th IEEE Conference on Nanotechnology*, Aug. 2004.

[14] D. Coleman et al., "Target-Logic Circuits Built with Holonomic Field Programmable Robot Arrays," *Proceedings of the Work-in-Progress Session of the 32nd EuroMicro Conference*, Cavtat, Croatia, ISBN 3-902457-11-2, pp. 53-54, Sept., 2006.

[15] M. G. Arnold and J. H. Cho, "Parallax-Docking and Reconfiguration of Field Programmable Robot Arrays Using an Intermittently-Powered One-Hot Controller," *International Conference on ReConFigurable Computing and FPGAs,* Cancun, Mexico, Dec. 2009.

[16] J. H. Cho and M. G. Arnold, "Low-voltage Scratch-drive Micro-scalpels Controlled by a Binary-encoded Signal," *BIOSTEC 2010*, Valencia, Spain, Jan. 2010.

[17] J. H. Cho and M. G. Arnold, "Reconfigurable Multi-Component Sensors Built from MEMS Payloads Carried by Micro-Robots," *Sensors Application Symposium 2010*, Limerick, Ireland, Feb. 2010.