

SINGLE-CHIP LIDAR BY MULTI-ORDER AND
MULTI-PULSE BEAM STEERING WITH DIGITAL MICRO
MIRROR DEVICE

by
Joshua Rodriguez

Copyright © Joshua Rodriguez 2019

A Thesis Submitted to the Faculty of the
COLLEGE OF OPTICAL SCIENCES
In Partial Fulfillment of the Requirements
For the Degree of
MASTER OF SCIENCE
In the Graduate College
THE UNIVERSITY OF ARIZONA

2019

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Master's Committee, we certify that we have read the thesis prepared by Joshua Rodriguez, titled *Single-Chip LIDAR by Multi-Order and Multi-Pulse Beam Steering with Digital Micro Mirror Device* and recommend that it be accepted as fulfilling the dissertation requirement for the Master's Degree.

Yuzuru Takashima
Dr. Yuzuru Takashima

Date: May 13, 2019

Kim Dae Wook
Dr. Dae Wook Kim

Date: May 17, 2019

Young Sik Kim
Dr. Young Sik Kim

Date: 05/15/2019

Final approval and acceptance of this thesis is contingent upon the candidate's submission of the final copies of the thesis to the Graduate College.

I hereby certify that I have read this thesis prepared under my direction and recommend that it be accepted as fulfilling the Master's requirement.

Yuzuru Takashima
Dr. Yuzuru Takashima
Master's Thesis Committee Chair
James C. Wyant College of Optical Science

Date: May 15, 2019 [®]

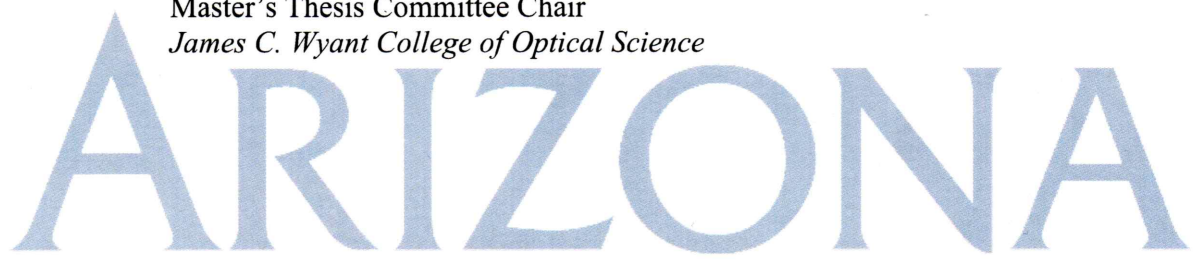


TABLE OF CONTENTS

LIST OF FIGURES	5
LIST OF TABLES	7
ABSTRACT	8
CHAPTER 1. INTRODUCTION	9
CHAPTER 2. MULTI-BEAM AND MULTI-PULSE BEAM STEERING BY DMD	11
2.1. Theory	11
2.1.1. Beam Steering	11
2.1.2. Diffraction Efficiency	13
2.2. Experimental Demonstration	14
2.2.1. Diffraction Efficiency	14
2.2.2. Multi-Pulse and Discrete Beam Steering	15
2.2.3. Multi-Beam and Discrete Beam Steering	16
CHAPTER 3. SINGLE-CHIP LIDAR IMPLEMENTATION	18
3.1. Circuitry	18
3.2. Receiver	19
3.3. Multi-Pulse Ranging	20
3.4. Ranging with Retroreflective Target	22
3.4.1. Single-Pulse Baseline Ranging	22
3.4.2. Multi-Pulse Ranging	23
3.5. Multi-Beam Ranging	24
CHAPTER 4. DISCUSSION	26
4.1. Distance Reporting Error	27
4.2. DMD Transitional Time	30
4.3. Specular Reflection of Cover Glass	31
4.4. Dynamic Interferometry	32
CHAPTER 5. SYSTEM SCALING	35
5.1. DMD	35
5.2. High Speed Laser Pulsing	36
5.3. 2D Scanning	36
5.3.1. Multiple Laser Diodes	36
5.3.2. Spatial Light Modulation	37

TABLE OF CONTENTS—*Continued*

CHAPTER 6. FUTURE RESEARCH	38
6.1. Mirror Speed	38
6.2. Bragg-Selective Retroreflective Paint	38
CHAPTER 7. CONCLUSION	39
APPENDIX A. PUBLICATIONS	40
APPENDIX B. CODE	42
B.1. Multi-Pulse LiDAR	42
B.2. Multi-Beam LiDAR	52
REFERENCES	61

LIST OF FIGURES

FIGURE 2.1. Representation of the (a) DMD diamond pixel layout (Top View); (b) a mirror in the "on" position at $+12^\circ$ (c) a mirror in the "parked" position at 0° when the DMD is powered down; (d) a mirror in the "off" position at -12° (Side View).[1]	11
FIGURE 2.2. Diagram of system control timing with respect to controller triggering for this experiment's multi-pulse configuration.	12
FIGURE 2.3. Calculated diffraction efficiency of DMD with one laser diode and three laser diodes incident on system.	14
FIGURE 2.4. Throughput efficiency of DMD.	15
FIGURE 2.5. a) Multi-pulse per frame beam steering setup. b) 2 out of total 5 spots are simultaneously created by redirecting two ns pulses with 80ns interval, in inset trigger to laser A and B (yellow in color) and APD signal (Blue in color). (Visualization 1).	16
FIGURE 2.6. a) Setup utilizing 3 laser diodes at offset angles. b) Total of 5 scanning spots from top to bottom, scanning point -7 to +7. Each photo corresponds to a single DMD transition where two pulses are transmitted in a single cycle. At the bottom of the picture, a long exposure depicts all 15 scanning spots in their sequence. (Visualization 2)	17
FIGURE 3.1. System layout of Multi-Beam Single-Chip DMD LIDAR circuitry. . .	18
FIGURE 3.2. Range finding demo of single-chip a) multi-pulse lidar and b) multi-beam lidar (Visualizations 3 and 4).	19
FIGURE 3.3. Schematic of detection system for LiDAR application. A band-pass filter (ThorLabs FL905-10) is placed in front of detection system to filter out all but 905nm light.	20
FIGURE 3.4. a) Measured distance vs. actual distance, b) RMS measurement error, and c) STD sample error of the multi-pulse configuration over 12cm in 2cm increments. 10 samples taken per measurement.	21
FIGURE 3.5. a) RMS error of distance measurements in 10cm increments. b) Standard deviation of 10 points of target at minimum measured distance and maximum measured distance. Line graph represents maximum measured distance at each point. Single-pulse configuration used with cardboard target.	23
FIGURE 3.6. a) RMS error of distance measurements in 10cm increments. b) Standard deviation of 100 points of target at minimum measured distance and maximum measured distance. Line graph represents maximum measured distance at each point. +1, 0, and -1 orders measured twice, once as first pulse in sequence and once as second pulse in sequence. Multi-pulse configuration used with corner cube array as target.	24

LIST OF FIGURES—*Continued*

FIGURE 3.7. a) Measured distance vs. actual distance, b) RMS measurement error, and c) STD sample error of the multi-beam configuration in 5cm increments. 10 samples taken per measurement	25
FIGURE 4.1. Scatter plot of distances reported from system for central and outer order. Raw digital number from timing chip displayed.	28
FIGURE 4.2. CFD circuit schematic.	29
FIGURE 4.3. CFD signal processing steps.	30
FIGURE 4.4. Throughput efficiency as a function of time.	31
FIGURE 4.5. DMD flipping sequence with respect to light path in multi-pulse mode.	31
FIGURE 4.6. Specular reflection by cover glass and residual diffracted power to 0th order.	32
FIGURE 4.7. Dynamic interferometry with Mach Zehnder interferometer configuration and 532nm pulsed source.	33
FIGURE 4.8. Fringe patterns produced by Mach Zehnder Interferometer measuring combination of DMD surface and cover glass by diffraction order with 532nm pulsed source.	34
FIGURE 5.1. Example setup of 2D Multi-Beam Single-Chip LIDAR. Laser diodes stacked in vertical direction are steered in the horizontal direction by DMD.	37
FIGURE 5.2. Setup of Multi-Beam Single-Chip LIDAR with multiple MEMS devices. The first MEMS device provides fine scanning while the DMD provides coarse scanning.	37

LIST OF TABLES

TABLE 5.1. Performance summary for two DMD types and two wavelengths of light in single-pulse and multi-beam mode[1].	35
TABLE 5.2. Performance summary for two DMD types and two wavelengths of light in multi-pulse mode.	36

ABSTRACT

We demonstrate the feasibility of enhancing the scanning rate for MEMS and diffraction based beam steering employing Digital Micromirror Device (DMD) by one to two orders of magnitude, which is configured as a programmable blazed grating. The tilt movement of micromirrors synchronizes with multiple pulses from multiple laser sources that sequentially redirect the pulses to multiple diffraction orders within μs . The approach opens up a pathway to achieve a LIDAR system with a scanning rate over 1M samples/s while leveraging a state of the art DMD and a moderate number of laser sources.

CHAPTER 1

INTRODUCTION

Laser beam steering technology is essential for Light Detection and Ranging (LIDAR) systems. For this reason, beam steering technologies have been actively researched. Along with mechanical and completely non-mechanical beam steering, Micro-Electro-Mechanical Systems (MEMS) is an emerging beam steering field that is especially suitable for LIDAR systems. Mechanical scanning including gimbals, fast-steering mirrors, Risley prisms, rotating polygon mirrors and gratings have been used for wide wavelength ranges [2]. Although mechanical beam scanning modalities are widely adopted, having fewer or no moving parts and smaller component inertia is more desirable for fast and compact beam steering devices so that size, weight, cost, and power consumption can be reduced [3, 4]. These qualities are especially required for autonomous vehicle and robotics applications. In contrast, completely non-mechanical scanning such as programmable spatial light modulators, modulo 2π optical phased arrays, solid state phase arrays, and liquid crystal electro-optic scanners are emerging, and are now actively researched [3, 5, 6, 7, 8].

In terms of reducing small component weight and inertia to fast beam steering, Micro-Electro-Mechanical Systems (MEMS) are promising due to their small size and weight, low production cost, high energy efficiency, and applicability to wide wavelength ranges. These MEMS devices include single resonant mirrors and shifting lenslet arrays [3, 9, 10]. However, in LIDAR applications for autonomous vehicles, a large steering angle as well as large beam size are needed to cover a large angle of scanning and minimize beam divergence due to diffraction. Unfortunately, resonant mirrors and shifting lenslet arrays are limited in angular range and maximum accommodated beam size. Current high-end resonant mirror MEMS scanning systems have moderate fields of view at 36° and scan rates of 21 kHz [3, 11]. However, a resonant mirrorFLs maximum beam diameter is only in-

creased at the expense of decreasing the maximum scan rate [10]. An optical amplification of the steering angle by an inverse telescope design has been reported; however, this design requires a reduced beam diameter to conserve the Lagrange invariant, which would limit the effective delivery of light over large distances due to beam spreading by diffraction [12, 13].

As our group previously demonstrated, beam steering by Digital Micromirror Device (DMD) with a short pulsed laser is not only possible, but can be implemented in highly accurate scanning LIDAR systems[1]. This beam steering technique has a high beam steering efficiency (close to 100% in theory), a larger beam size (same as DMD area), a wide field of view (48°), and a high scan rate (tens of kHz) while minimizing the number of moving parts. The limitation of this approach, however, is that the maximum number of scanning points is limited to about 5 to 9 points for a wavelength range of 0.5 to $1\mu\text{m}$ with a commercially available DMD, with specifications further outlined in Sec. 5.1. We now propose and demonstrate increasing the angular resolution of the LIDAR system by implementing multiple laser diodes.

CHAPTER 2

MULTI-BEAM AND MULTI-PULSE BEAM STEERING BY DMD

2.1 Theory

2.1.1 Beam Steering

In Fig. 2.1(a), the DMD is schematically depicted. This beam steering setup utilizes a 608x684 (horizontal by vertical) DMD chip (DLP3000, Texas Instruments). The micromirrors are positioned in a diamond configuration with a corner-to-corner period of $10.8\mu\text{m}$ as shown in Fig. 2.1(a). On this DMD, an array of micromirrors flip between an "on" and "off" state, shown in Fig. 2.1(b)-(d), by rotating $\pm 12^\circ$ about an axis defined by the diagonal of the mirror. The typical transitional time is several μs [14].

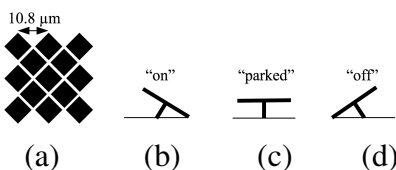


FIGURE 2.1. Representation of the (a) DMD diamond pixel layout (Top View); (b) a mirror in the "on" position at $+12^\circ$ (c) a mirror in the "parked" position at 0° when the DMD is powered down; (d) a mirror in the "off" position at -12° (Side View).[1]

Originally the DMD was developed for intensity spatial light modulation by turning pixels "on" and "off". Here, the unexplored transitional state between the "on" and "off" states is utilized with a *ns* short-pulsed laser illuminating the transitional state of the mirror. Under the circumstances, by precisely synchronizing the laser pulse and DMD mirror movement, the micromirror movement is effectively frozen at an angle between the stationary on and off states, to form a programmable and highly efficient blazed diffraction grating to discretely steer the laser beam [1].

The number of scanning points N is given by $2m + 1$ where m is the maximum allowable diffraction orders for a blazed diffraction that satisfies:

$$\sin(\theta_i) + \sin(\theta_r) = \frac{m\lambda}{a} \quad (2.1)$$

$$\theta_i + \theta_r = 2\gamma \quad (2.2)$$

where θ_i and θ_r are angle of incidence and diffraction measured from normal of blazed grating surface, λ is a wavelength, a is the pitch of the blazed grating and γ is the slope of the grating facet. For example, at 905nm, with a pitch the same as the DLP3000, the maximum number of scanning points N is limited up to 5 [1]. In Fig. 2.2 a timing diagram for DMD and multiple ns laser pulse control is depicted. A trigger signal from a microcontroller initiates flipping mirrors and launch laser pulse. The timing of emitting ns pulses is precisely synchronized and tuned to mirror movement while taking into account overall delay time of DLP3000 with respect to the trigger signal, $218 \mu s$. Note that in the timing diagram, two consecutive pulses shown, and are synchronized to the mirror movement so that each of the pulses hits mirror at a timing corresponding to two consecutive diffraction orders. Thus, the scheme of multiple pulses per mirror rotation increases the beam scan rate by factor of N .

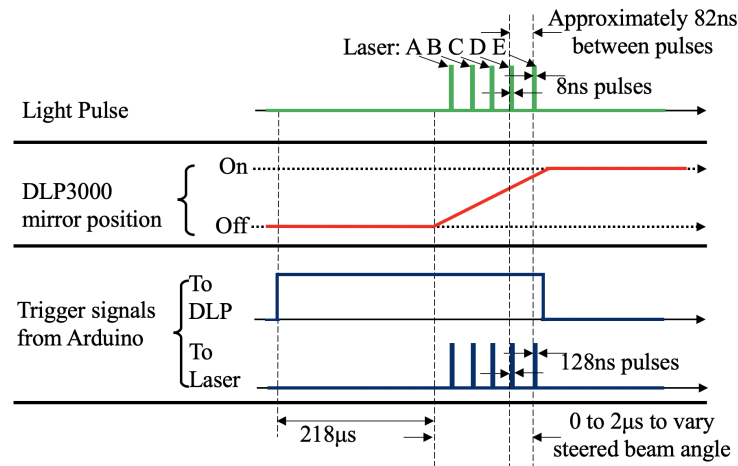


FIGURE 2.2. Diagram of system control timing with respect to controller triggering for this experiment's multi-pulse configuration.

2.1.2 Diffraction Efficiency

To estimate the efficiency of the DMD as a blazed grating, the diffraction efficiency was calculated by modeling each mirror as a series of point sources. Each had an associated phase and optical path length (OPL) induced by the tilt of mirror while taking into account the angle of incidence of the incoming beam. The field was calculated through space with the resulting diffraction efficiencies plotted in Fig 2.3.

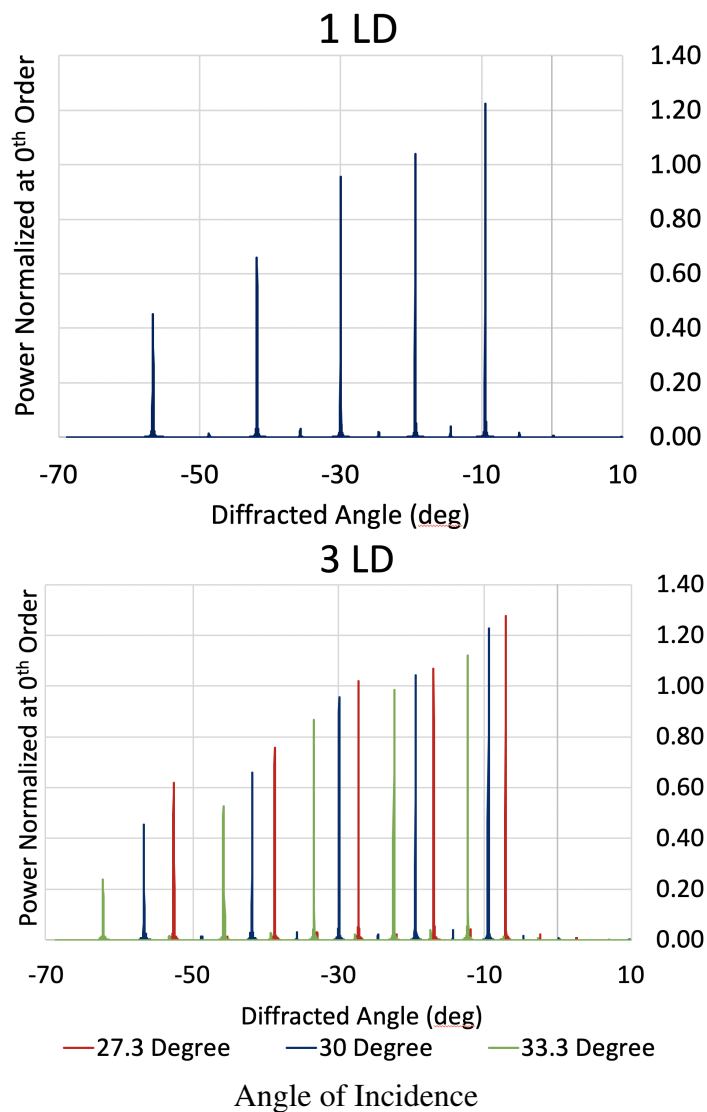


FIGURE 2.3. Calculated diffraction efficiency of DMD with one laser diode and three laser diodes incident on system.

2.2 Experimental Demonstration

2.2.1 Diffraction Efficiency

The diffraction efficiency was then measured at each steered beam angle with a 905nm source. The transmitted power was measured by aiming a steered beam onto a single pixel

APD detector (Hamamatsu C12702). The laser beam's cross-sectional area was smaller than the area of the DMD and the measurement was conducted with respect to the source's incident power on the detector. The results are plotted as discrete measurements in Fig. 2.4. Since the beams pass through a cover glass, both incoming and outgoing, the measurements were effectively the throughput efficiency, as the glass reflected a significant amount of light to the 0th order and did not diffract a large amount of light as a result. However, even at outer orders, the system is still able to steer a large amount of light from its source.

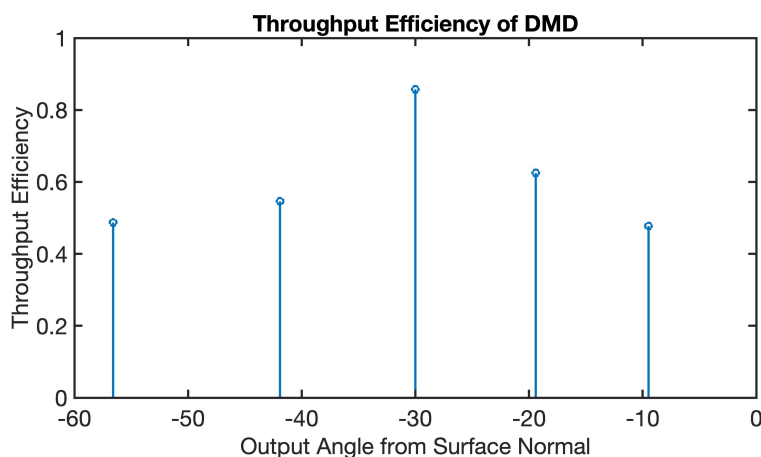


FIGURE 2.4. Throughput efficiency of DMD.

2.2.2 Multi-Pulse and Discrete Beam Steering

In Fig. 2.5a, we employed externally triggered lasers (LS9-220-8-S10-00, Laser Components, Germany) having a kHz rep rate to create two consecutive pulses equivalent to a single laser operating at MHz repetition rate. The trigger signals were sent to the first and second laser diode, (Laser A and B) with a set delay of 62.5ns. After the collimating lens (ThorLabs, model 230A), a beam splitter combines two pulses that illuminates DMD at the angle of incidence of 30° via fold mirror. A collection lens and APD module (Hamamatsu C12702) module placed at the opposite side of the fold mirror to the DMD is used to validate the multi-pulse operation for lidar, described in Section 3.

Figure 2.5b shows the pulse sequence as detected by the APD along with trigger signal to laser diode. The multi-pulse operation is confirmed as two APD outputs (blue in color) appears within the set interval. Figure 2.5b shows the beam steering results. 2 out of total 5 scanning spots are simultaneously created by redirecting two ns pulses having an approximately 560ns interval. From top to bottom, simultaneous scanning to -2 and -1, -1 and 0, and so on are depicted. Each tile of the photo corresponds to a single DMD transition where two pulses are transmitted in a single cycle. At the bottom of the picture, a long exposure to show all 5 scanning spots in their sequence is depicted.

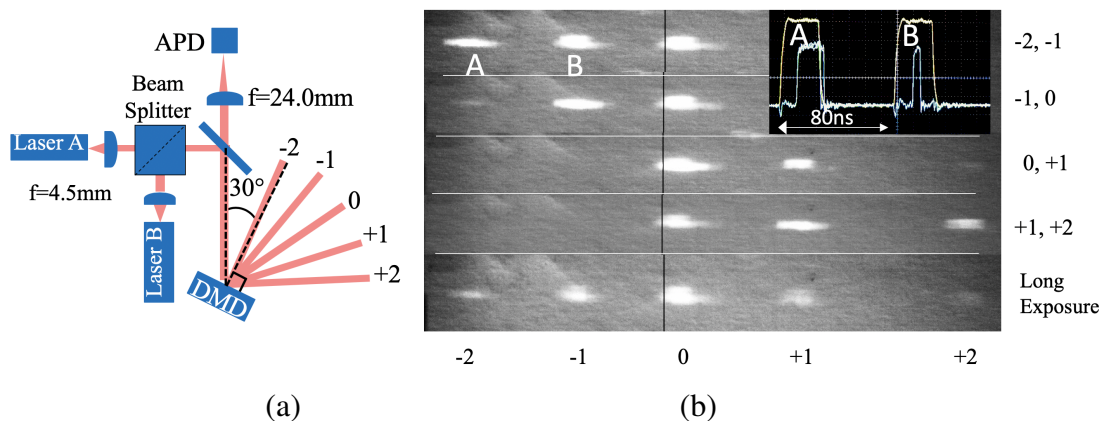


FIGURE 2.5. a) Multi-pulse per frame beam steering setup. b) 2 out of total 5 spots are simultaneously created by redirecting two ns pulses with 80ns interval, in inset trigger to laser A and B (yellow in color) and APD signal (Blue in color). (Visualization 1).

2.2.3 Multi-Beam and Discrete Beam Steering

An experimental setup similar to the multi-pulse beam steering setup was used for the multi-beam demonstration. Figure 2.6a depicts 3 laser modules (LS9-220-8-S10-00, Laser Components, Germany), collimated by an NA 0.4 objective lens (Model, Rolyn)[1] are arranged having angle of incidence to DMD, 33.3, 30.0, and 26.7 degrees, respectively. Figure 2.6b shows a screen capture of the 15 beam spots produced from a 3 laser diode beam steering setup. A screen was placed at a distance of 50cm from the DMD, and the scanned spots were captured as a movie. At the bottom of the picture, a long exposure of

all 15 scanning spots is depicted.

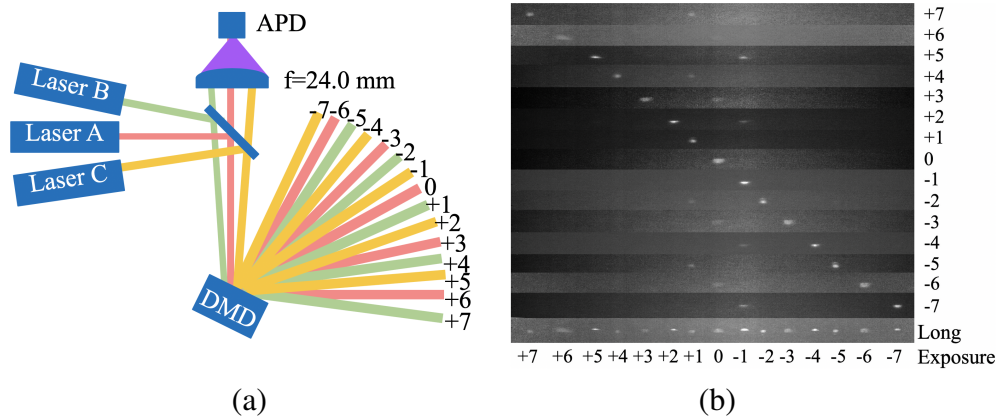


FIGURE 2.6. a) Setup utilizing 3 laser diodes at offset angles. b) Total of 5 scanning spots from top to bottom, scanning point -7 to +7. Each photo corresponds to a single DMD transition where two pulses are transmitted in a single cycle. At the bottom of the picture, a long exposure depicts all 15 scanning spots in their sequence. (Visualization 2)

CHAPTER 3

SINGLE-CHIP LIDAR IMPLEMENTATION

3.1 Circuitry

The block diagram of multi-pulse, multi-beam single-chip DMD LIDAR circuitry is depicted in Fig. 3.1. The tilt timing of the DMD mirror is synchronized to the laser pulse by an Arduino microcontroller (Arduino Uno, Arduino) with a DS1023 delay IC chip (Dallas Semiconductor). In addition to the Arduino microcontroller, several timing elements are utilized to control the beam steering system. Texas Instruments' TDC 7200 Time to Digital Converter is used to measure the time of flight of the laser pulses. It has the option to capture multiple times of flight in one cycle, a feature is used for multi-pulse lidar system. An array of AND gates is used to select and control specific laser diodes when controlling the multi-beam configuration, allowing single orders to be activated. Signal returning from an object (while cardboard) is captured by APD (Hamamatsu C12702).

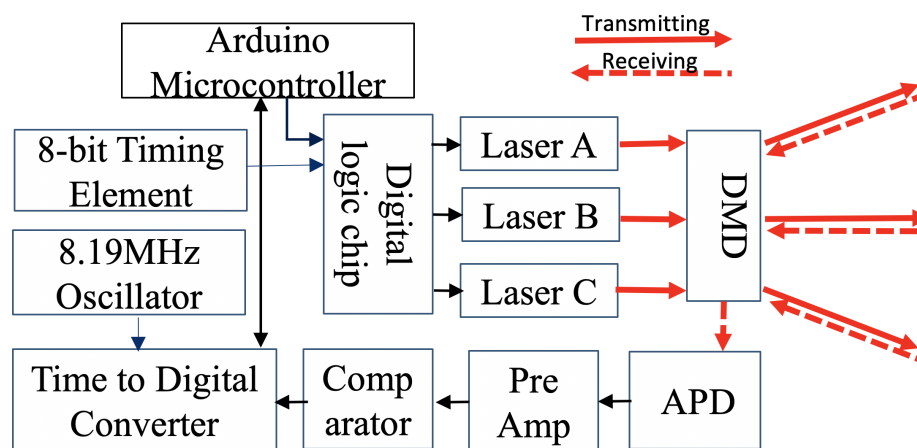


FIGURE 3.1. System layout of Multi-Beam Single-Chip DMD LIDAR circuitry.

Figure 3.2a and b shows range finding demonstrations of single-chip multi-pulse and multi-beam lidar respectively. As a moving object, a cube suspended by wire is placed

in front of the lidar. The multi-pulse lidar measures 5 angles over 48 degrees with scan rate of 6.68KHz, which is two times faster than frame rate of the DMD. Separately, the multi-beam configuration measures 15 angles over 48 degrees with 3.34KHz scan rate.

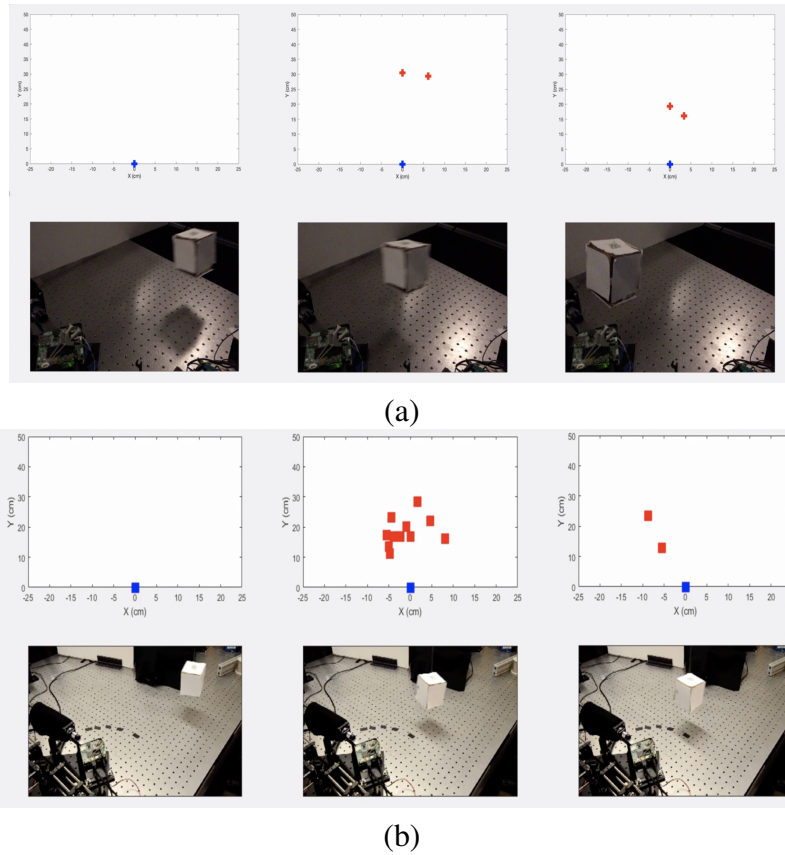


FIGURE 3.2. Range finding demo of single-chip a) multi-pulse lidar and b) multi-beam lidar (Visualizations 3 and 4).

3.2 Receiver

To make time of flight measurements possible, a detector system is required. In the case of this system, a single pixel APD (Hamamatsu C12702) is used. To capture the light coming from the relatively large reflection area of the DMD, light collection optics are needed. In this setup, an afocal telescope setup is used in combination with an integrating rod to funnel the light received from the object to the detector, shown in Fig. 3.3. This is necessary since

the detector is only 3mm in diameter and capturing as much light as possible is key to extending the range of the system.

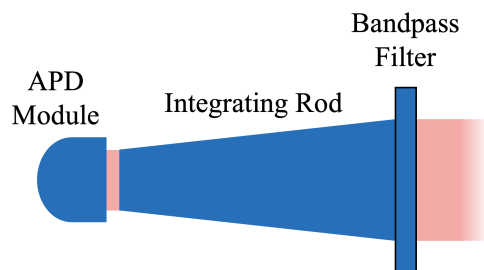
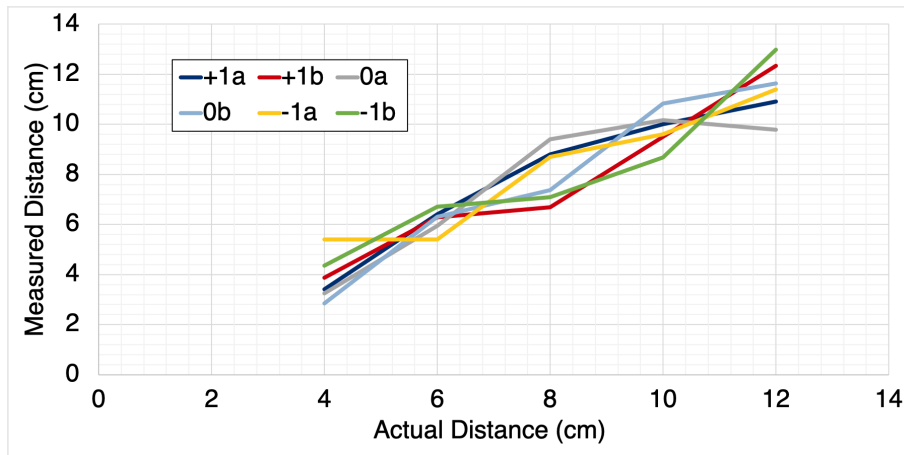


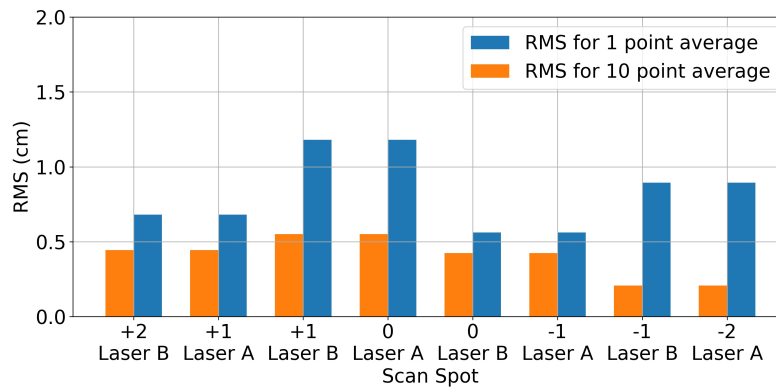
FIGURE 3.3. Schematic of detection system for LiDAR application. A band-pass filter (ThorLabs FL905-10) is placed in front of detection system to filter out all but 905nm light.

3.3 Multi-Pulse Ranging

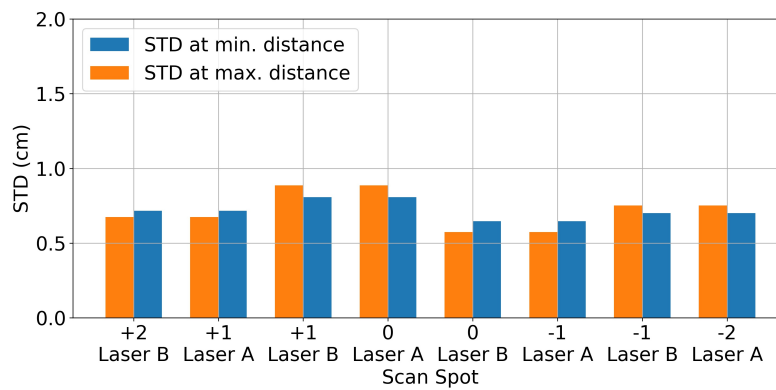
Figure 3.4 depicts the distance measurement accuracy of the multi-pulse and multi-beam setups. Figure 3.4 reports the results of certain diffraction orders as a line plot of the measured distance by the systems. An RMS error between reported and actual distance is evaluated for each of the measurement spots seen in Fig. 2.5. A piece of cardboard aligned in each of the diffraction orders was used as a target. The distance was measured at an increment of 2cm and 5 cm for multi-pulse and multi-beam configuration up to 12 and 30 cm, respectively. Overall, $< 1.5\text{cm}$ measurement accuracy has been achieved both for multi-pulse and multi-beam configurations.



(a)



(b)



(c)

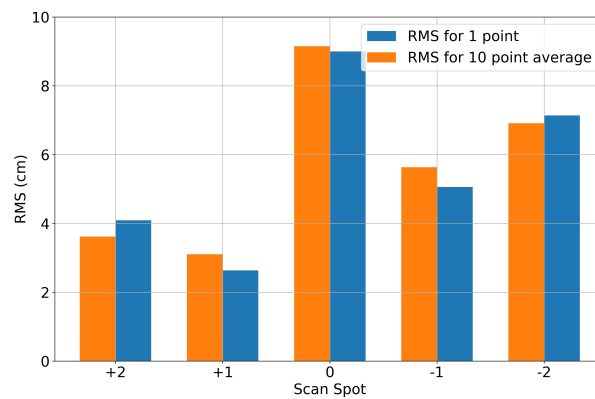
FIGURE 3.4. a) Measured distance vs. actual distance, b) RMS measurement error, and c) STD sample error of the multi-pulse configuration over 12cm in 2cm increments. 10 samples taken per measurement.

3.4 Rangefinding with Retroreflective Target

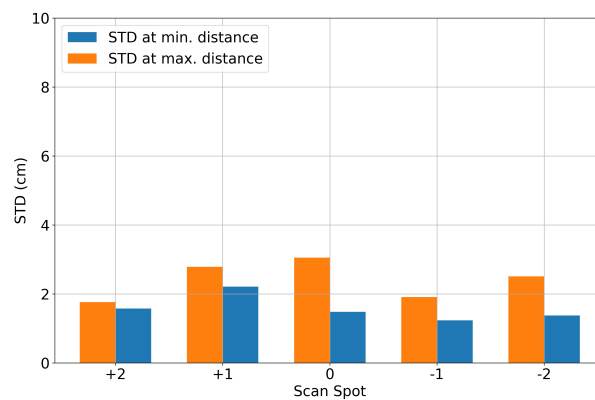
To demonstrate the system's scalability in multiple configurations, a distance measurement experiment was conducted using a retroreflective target—a corner cube array. The goal is to demonstrate that even with a smaller lab table setup, the system is able to accurately measure the distance to an object at longer distances, even though a retroreflective target is not realistic. This, however, will set a precedent for the system's performance.

3.4.1 Single-Pulse Baseline Rangefinding

A set of baseline data was captured utilizing a corner cube array to compare the system's accuracy in a multi-pulse configuration to that of a single-pulse. Additionally, the absolute maximum measurable distance with this configuration was tested at the 0th order. It was found to be 2 meters. The results are shown in Figure 3.5. The single pulse configuration produces a slightly lower overall standard deviation than the multi-pulse configuration, but has a higher RMS error for the central order.



(a)

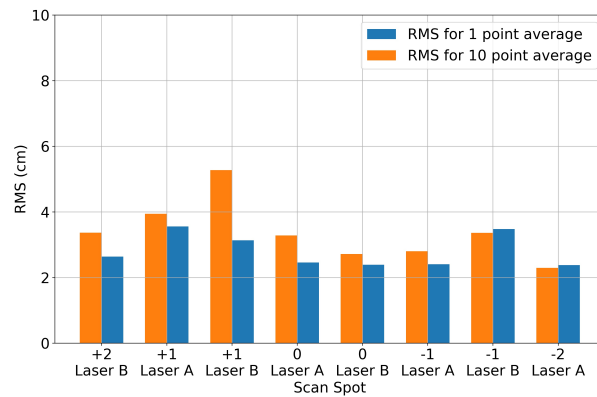


(b)

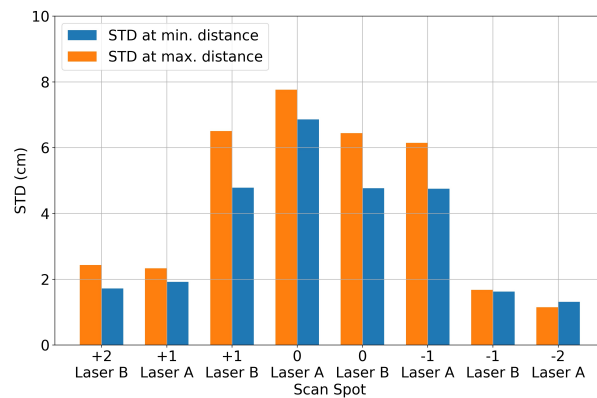
FIGURE 3.5. a) RMS error of distance measurements in 10cm increments. b) Standard deviation of 10 points of target at minimum measured distance and maximum measured distance. Line graph represents maximum measured distance at each point. Single-pulse configuration used with cardboard target.

3.4.2 Multi-Pulse Ranging

A test was performed with a corner cube array to return most of the transmitted photons to the detector since the laser power is reduced by the beam splitter for this lab setup. This would also provide an opportunity to test the accuracy of the system at longer distances. It can be seen in this setup that the system could accurately measure distances over 1m for all orders.



(a)

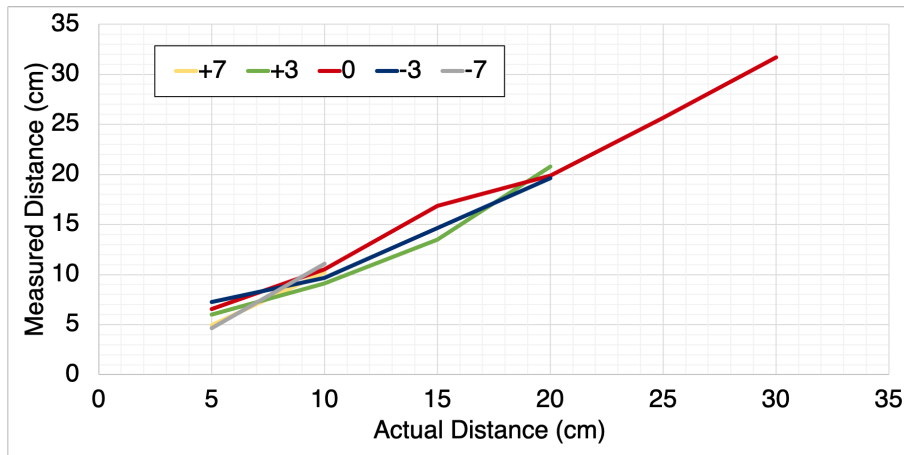


(b)

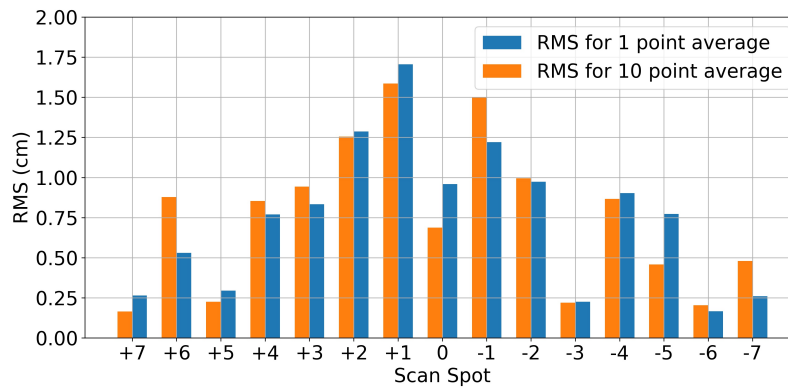
FIGURE 3.6. a) RMS error of distance measurements in 10cm increments. b) Standard deviation of 100 points of target at minimum measured distance and maximum measured distance. Line graph represents maximum measured distance at each point. +1, 0, and -1 orders measured twice, once as first pulse in sequence and once as second pulse in sequence. Multi-pulse configuration used with corner cube array as target.

3.5 Multi-Beam Ranging

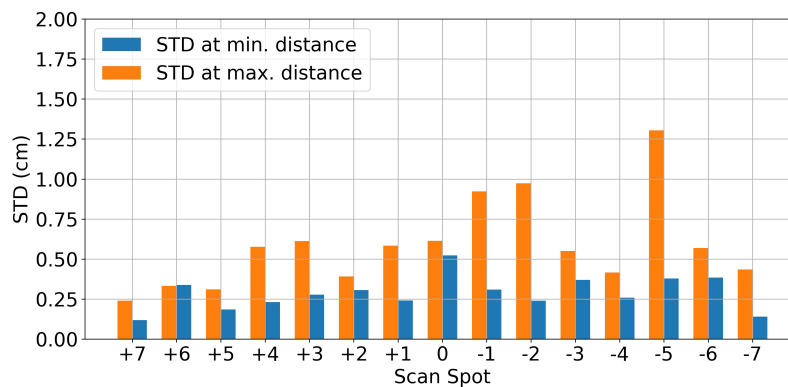
A similar setup was used for the multi-beam demonstration. Figure 3.7(a) depicts 3 laser modules. Laser A, B and C are arranged having an angle of incidence to DMD, 33.3, 30.0, and 29.3 degrees, respectively. Figure 3.7(b) shows a screen capture of the 15 beam spots produced from the 3 lasers captured on a screen placed at a distance of 50cm from the DMD.



(a)



(b)



(c)

FIGURE 3.7. a) Measured distance vs. actual distance, b) RMS measurement error, and c) STD sample error of the multi-beam configuration in 5cm increments. 10 samples taken per measurement

CHAPTER 4

DISCUSSION

Two new beam steering schemes, multi-pulse per mirror transition and multi-beam operation for DMD-based beam steering, increase the scan speed and the number of scanning points simultaneously. With minimal modification of the optical and electrical architecture, the range finding accuracy $< 1.75\text{cm}$ RMS is achieved. Combining those techniques potentially satisfies the scan rate requirement of 2MHz for front facing LIDAR for self-driving cars having a field of view of 100(H) x 25(V) degree with 0.2° resolution at 40Hz frame rate[15]. For example, the DLP7000 operating at 23kHz frames rate at a wavelength of 905nm can support 9 diffraction orders. With 11 laser sources, an overall scan rate of 2MHz is achievable. In this scenario, the lasers do NOT need to operate in a burst mode with MHz repetition rate, but it only needs to launch 9 pulses with a several *ns* interval, since the pulse train is launched with about a $40\mu\text{s}$ interval to be matched to 23kHz DMD refresh rate. With FET switching, it is possible to pulse the laser for range finding applications several times with a short duration, i.e., double 1.5ns pulses with 5ns separation[15]. For other applications that do not require as high of pulse energy as lidar applications, MHz and external triggered *ns* pulsed lasers are commercially available which assures the multi-pulse concept is applicable.

The scan rate itself indeed satisfies the requirement, but the number of scanning points still is limited by the number of diffraction orders and lasers. This indicates that further research on optical architecture to distribute laser pulses to a thousand or more points is inevitable. Also, an experimental demonstration of a FET driven *ns* laser source with a 10ns pulse interval, along with slowing down the transition of micromirrors, is expected to fully utilize the potential performance of the proposed multi-beam multi-pulse beam steering by DMD.

4.1 Distance Reporting Error

A test of the distance resolution was performed to determine the accuracy of the system's measurement. A diffusive target and retroreflector was tested, with a corner cube array chosen to test the system over a longer distance. Though the corner cube array provided a longer distance measurement, it caused issues with noise in the system. The corner cube array causes noise and measurement inconsistency due to numerous features of the returning beam to the system. For one, the light is close to being collimated, which causes the light to have slightly different optical path differences throughout the receiving optics. Secondly, the light reflected back to the system can be shifted in numerous directions depending on which corner cube in the array reflects most of the light. Both of these factors can cause the system to read false distances momentarily, causing noise in the pulse train at a particular distance. This increases the noise and subsequently the standard deviation of measurements in some diffraction orders for the multi-pulse distance test. This problem can be prevented, however, by using a faster-pulsed laser to prevent the need for a beam splitter reducing the power supply to the system and thus preventing the need for a retroreflective target to measure longer distances.

Figure 4.1 displays scatter plots of digital number distances reported by the timing chip of specific distances. It can be seen that for static targets, there is a variance in the distances reported, with a slightly greater variance for the outer order. This shows evidence of noise in the system. In a set of 20 points taken at once, occasionally values that vary greatly from the actual value are present that alter the standard deviation. However, the data remains very linear in its reporting, which corresponds to the lower RMS error of the measured distances. The error corresponds to about 2-3cm. This amount of error specification of the TDC chip due to jitter and other timing inconsistencies.

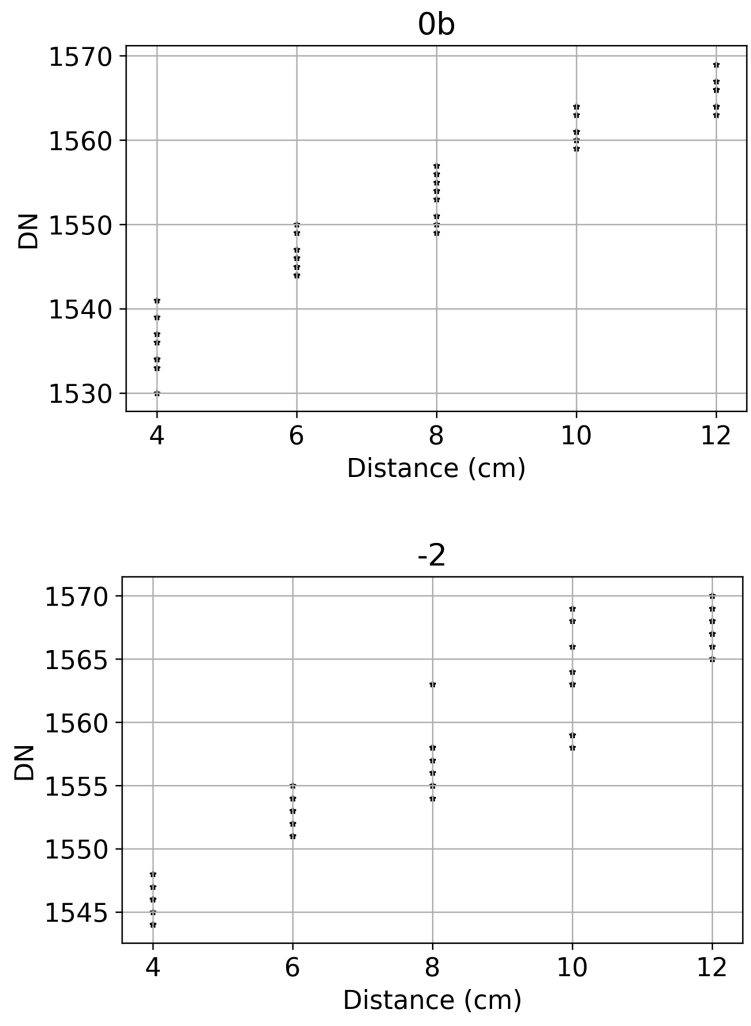


FIGURE 4.1. Scatter plot of distances reported from system for central and outer order. Raw digital number from timing chip displayed.

There are methods available to mitigate error in time of flight measurements, such as using a higher-resolution chip to measure the time of flight of the circuit and a constant fraction discriminator (CFD). A CFD utilizes an array of comparators to remove ambiguity in a received pulse's threshold. In a LiDAR system without a CFD, each received pulse has a rise time and is measured at a particular threshold. If an object is farther away, the pulse can have a weaker signal, which, if the threshold crossing is at a higher point in the pulse's

rise, will have a signal "walk" that is significant enough to correspond to a considerable distance error. The CFD takes an inverted and attenuated pulse, adds it to a delayed non-inverted signal, and looks for the resultant's zero crossing, as depicted in Fig. 4.3. This removes the ambiguity in the time of flight signal thresholding.

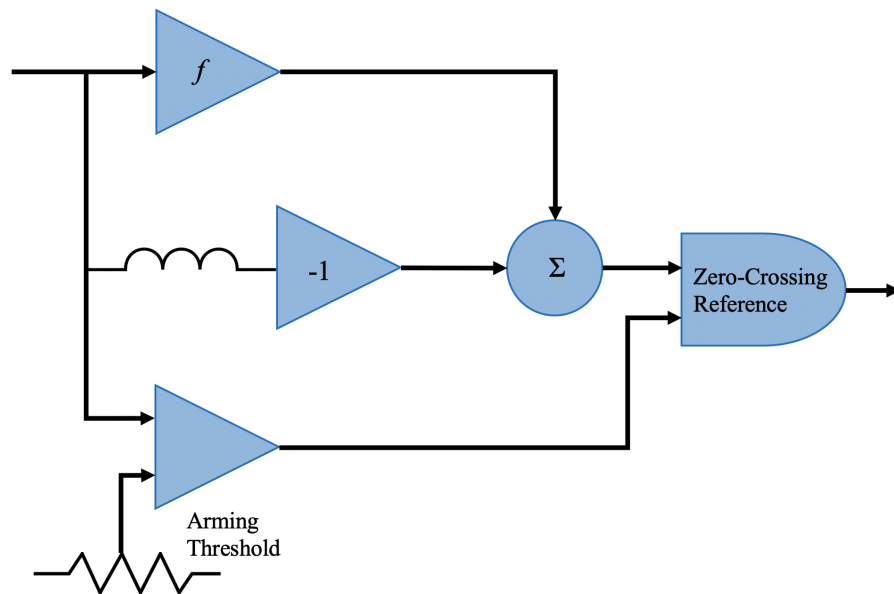


FIGURE 4.2. CFD circuit schematic.

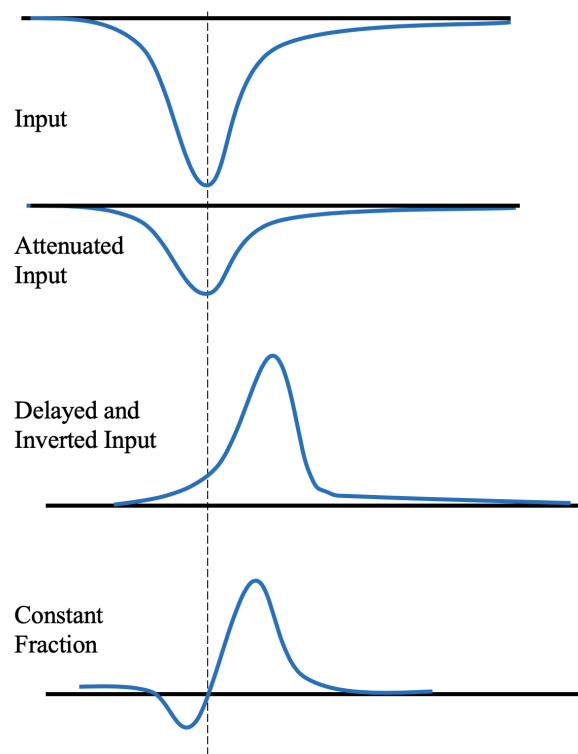


FIGURE 4.3. CFD signal processing steps.

4.2 DMD Transitional Time

The single-chip configuration employs DMD for both transmitter and receiver. In this configuration, mirror position between transmitting and receiving timing has to be small enough so that a higher receiving diffraction efficiency is assured. According to the diffraction efficiency curve (Fig. 4.4) as a function of time delay, 80% diffraction efficiency, relative to peak diffraction efficiency is achievable within a $\sim 250ns$ interval. As the micromirrors move, the diffraction efficiency decreases. The time interval limits the maximum measurable distance, 55m with current DLP3000. Slowing down the DMD is needed to extend the distance range. Figure 4.5 illustrates the flipping pattern as time goes on, illustrating why the diffraction efficiency changes over time at a specific angle.

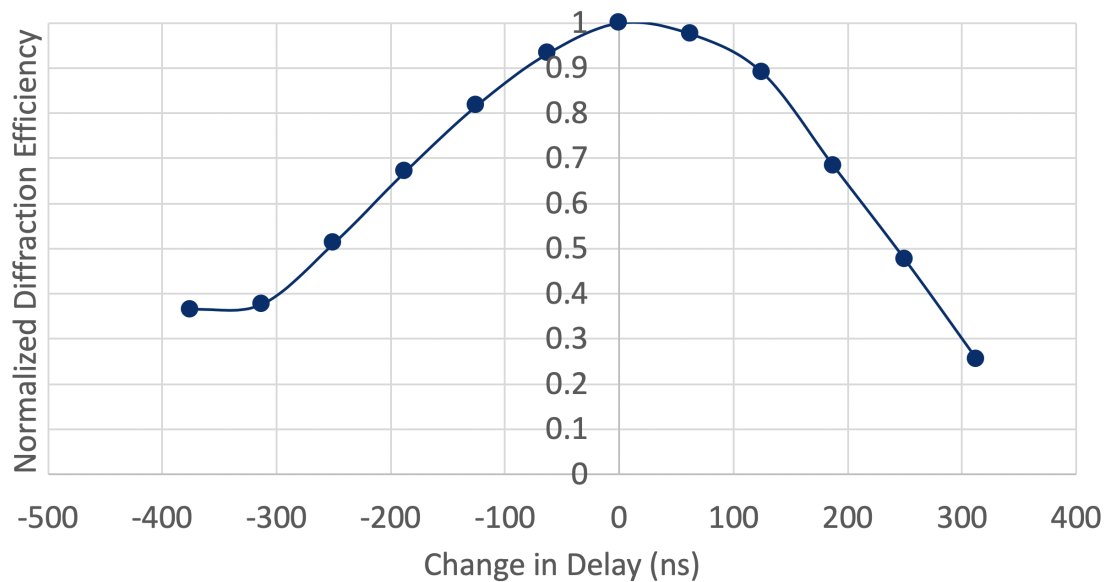


FIGURE 4.4. Throughput efficiency as a function of time.

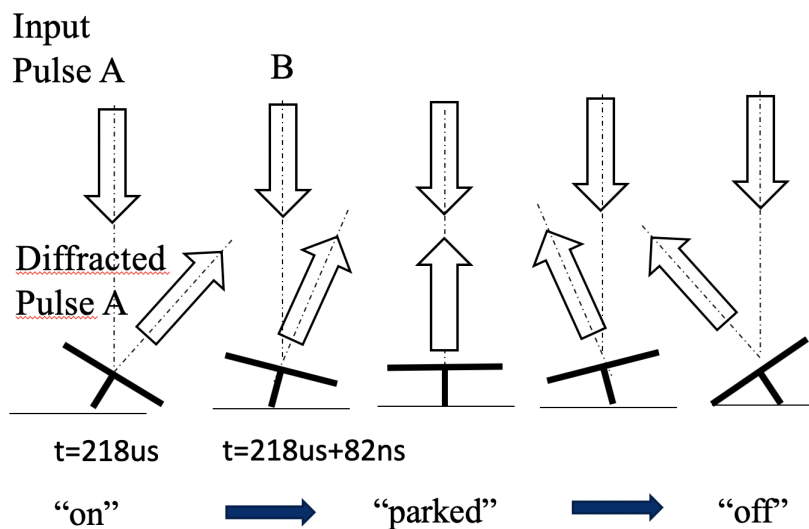


FIGURE 4.5. DMD flipping sequence with respect to light path in multi-pulse mode.

4.3 Specular Reflection of Cover Glass

At central orders, the cover glass of the DMD reflects a portion of the light sent from the laser source to the 0th order. If some of the light reflected outwards by the DMD is incident

on the object, it can interfere with the beam sent by the order measuring the object. This has been seen in previous experiments with this system [1]. It is especially problematic with retroreflective targets. This could be mitigated by replacing the cover glass of the DMD or using an anti reflective coating so that infrared light is not reflected. Thus, multiple object points are plotted for one single object located close to the DMD in the 0th order. The reflectivity of the Corning Eagle XG DMD cover glass was measured to be approximately 20% for 905nm light at a 30° angle of incidence. The mechanism causing the artifact is confirmed by the fact that the DMD cover glass has a high specular reflectivity. The fraction of power reflected by the cover glass plus the trace amount diffracted by the DMD is depicted in Fig. 4.6.

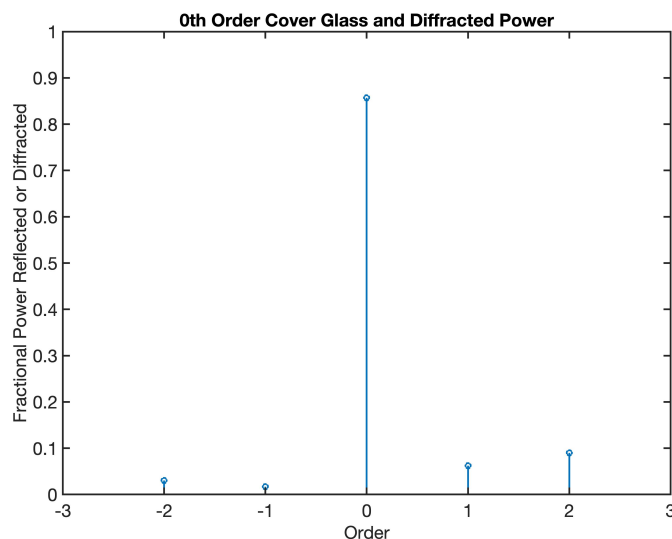


FIGURE 4.6. Specular reflection by cover glass and residual diffracted power to 0th order.

4.4 Dynamic Interferometry

Since the DMD is a key component to the beam steering concept, questions arise to the surface's flatness and any aberrations produced as a result. Dynamic interferometry was performed in a Mach Zehnder interferometer configuration to test the surface's flatness at

each diffraction order. In Fig. 4.7, schematic of the MZ interferometer is depicted. There, DMD (DLP3000) is mounted on a rotation stage. The center of rotation is aligned to the mirror surface. The timing of transition of mirror is synchronized to 532nm, 10 ns laser pulse (Model of DPSS laser). The beam was steered to each diffraction order using a combination of a pulsed laser source and the DMD beam steering in the test arm of the interferometer and measured at each position. The fringe is captured by a CMOS camera (ThorLabs DCC1645C) with imaging lens.

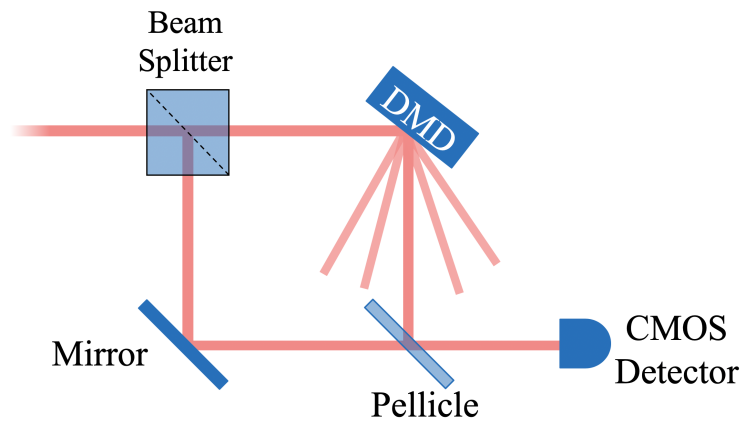


FIGURE 4.7. Dynamic interferometry with Mach Zehnder interferometer configuration and 532nm pulsed source.

The resulting fringes from the DMD are shown in Fig. 4.8. For each configuration, it can be seen that no more than one fringe of concentric optical path difference exists for $0 \sim -3$ diffraction orders at wavelength of 532nm. This corresponds to approximately $1/3.5$ waves of RMS error. Further defocusing would eliminate the residual symmetrical wave aberration. For $+1 \sim +3$ orders, about 1 wave of astigmatism is observed that would be corrected by inserting cylindrical lens that cancels astigmatism. The residual 1 waves of wave aberration in peak to valley at 532nm corresponds to about a half wave at wavelength 905nm. At worst performance in the +3 order the wavefront RMS error is approximately $1/3.5$ waves, which corresponds to the following peak-to-valley error:

$$\frac{1}{3.5} \cdot \left(\frac{905}{532} \right) \cdot \lambda = \frac{1}{6} \lambda \quad (4.1)$$

At best performance in the -3 order the wavefront RMS error the wavefront RMS error is approximately $1/6$ waves, which corresponds to the following peak-to-valley error:

$$\frac{1}{6} \cdot \left(\frac{905}{532} \right) \cdot \lambda = \frac{1}{24} \lambda \quad (4.2)$$

With a 905nm source, the RMS error calculation is approximately diffraction limited. The primary aberrations that exist are astigmatism and defocus. In both situations, the aberrations can be mitigated by lenses. However, in the current beam steering setup the aberrations should not affect the outgoing or incoming beams, or the distance measurement results significantly due to the multi mode operation of laser beam which is a non diffraction limited one.

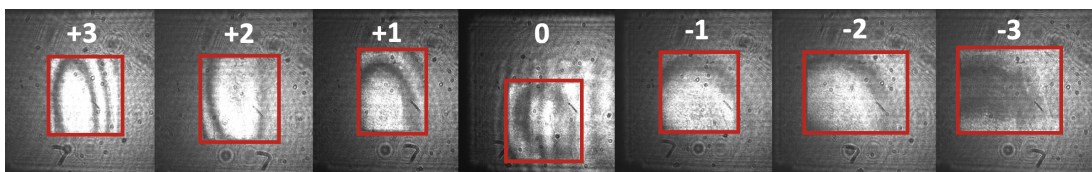


FIGURE 4.8. Fringe patterns produced by Mach Zehnder Interferometer measuring combination of DMD surface and cover glass by diffraction order with 532nm pulsed source.

CHAPTER 5
SYSTEM SCALING

5.1 DMD

The number of scan points is limited not only to a certain number of diffraction orders generated per laser diode, but also to the number of laser diodes used. With the current DMD (DLP3000), five diffraction orders are available when using a single 905 nm laser at a 30° incidence angle. The number of scanning angles can be increased by employing a larger DMD micromirror pitch and/or a shorter wavelength[1]. The limitations of two commercially available DMDs under ideal conditions for certain wavelengths of light are shown in Table 5.1. However, it is evident that we can accurately measure distance over 175m with a higher scanning rate of up to (256 lines/s) x 90 (points/line) = 23k points/sec and a field of view over 60° with 0.65° angular resolution.

TABLE 5.1. Performance summary for two DMD types and two wavelengths of light in single-pulse and multi-beam mode[1].

DMD Model	Wavelength	Range	N_{LD}	N_{ord}	Total Number of Scan Angles	FOV	Resolution	Scan Rate (lines/s)
DLP3000	905nm	55m	5	5	25	49°	1.9°	160
DLP3000	1550nm	55m	8	3	24	50°	2.1°	167
DLP9500	905nm	175m	11	7	77	48°	0.62°	299
DLP9500	1550nm	175m	18	5	90	60°	0.65°	256

When using the system in multi-pulse mode, the point production rate increases dramatically. In single-pulse mode, the system can only produce one point per "line" rate. However, in multi-pulse mode, the system produces a much faster pulse rate than in single-pulse mode. Table 5.2 shows the resolution of a single-beam multi-pulse configuration, with the DLP 3000 at 905nm configuration being used for the multi-pulse experiments of this paper.

TABLE 5.2. Performance summary for two DMD types and two wavelengths of light in multi-pulse mode.

DMD Model	Wavelength	Range	N_{ord}	FOV	Resolution	Multi-Pulse Scan Rate	
						(lines/s)	(points/s)
DLP3000	905nm	55m	5	49°	9.8°	160	800
DLP3000	1550nm	55m	3	50°	16.7°	167	501
DLP9500	905nm	175m	7	48°	6.85°	299	2095
DLP9500	1550nm	175m	5	60°	12°	256	1280

5.2 High Speed Laser Pulsing

In the multi-pulse LiDAR system as described in Sec. 2.1.1, and shown in Fig. 2.5a, two laser diodes are combined to a single optical path with a beam splitter to produce the pulse repetition rate necessary to accomplish multiple pulses per DMD transition. However, laser drivers are available to produce the necessary pulse repetition rate to accomplish this [16]. Having pulse rates $\sim MHz$, the laser would be able to pulse at a short enough interval to include all diffraction orders in the DMD's field of view in one transition period. This allows for an efficient high speed scanning process.

5.3 2D Scanning

5.3.1 Multiple Laser Diodes

A method for expanding the system's multi-beam scanning further is by scanning in the vertical direction in addition to the horizontal direction. Multiple laser diodes can be stacked in the vertical direction, or a laser diode stack can be used, to scan in the vertical direction as well as the horizontal. Adding a second dimension to the system's scanning would drastically transform the possibilities of the Multi-Beam Single-Chip LIDAR.

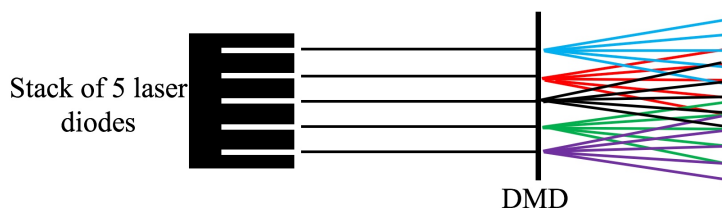


FIGURE 5.1. Example setup of 2D Multi-Beam Single-Chip LIDAR. Laser diodes stacked in vertical direction are steered in the horizontal direction by DMD.

5.3.2 Spatial Light Modulation

Another possible method of expanding the Multi-Beam Single-Chip LIDAR device is through the inclusion of another device to steer the beam between the “coarse” angular steering. Fig. 5.2 shows a proposed configuration of an optical setup involving multiple MEMS devices. Spatial light modulation, such as through a second MEMS device, would provide a means for scanning between the angular spacing of the system; drastically increasing the resolution of the system. Because of the DMD’s relatively large area, a beam of light incident on any portion of its area would be steered in the same manner. Additionally, some MEMS devices can scan in two dimensions, which would drastically expand the capability of the system’s beam steering.

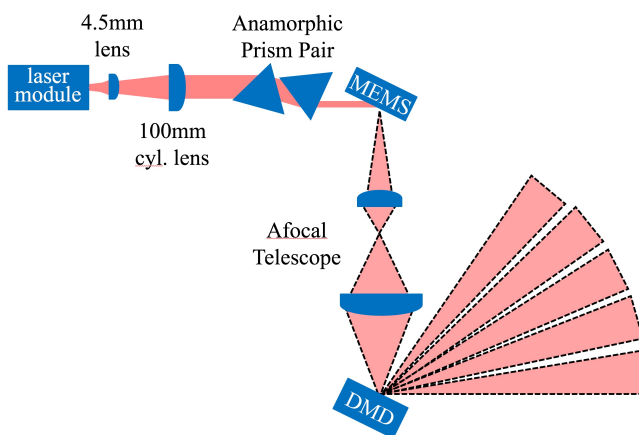


FIGURE 5.2. Setup of Multi-Beam Single-Chip LIDAR with multiple MEMS devices. The first MEMS device provides fine scanning while the DMD provides coarse scanning.

CHAPTER 6

FUTURE RESEARCH

6.1 Mirror Speed

A possible method for allowing for greater beam steering and LiDAR measurement accuracy is by slowing down the DMD being used. In doing so, the micromirrors' transition time is longer, where return pulses from distant objects would have a higher diffraction efficiency while being reflected back to the DMD since the micromirrors would not have moved much by the time the pulse returns to the system. One method is by placing the DMD in a pressure controlled chamber. The micromirrors' transition is induced by an electromagnetic field with the speed being controlled by a dense gas. The gas is held by the cover glass of the DMD. By changing the air pressure in the chamber, the micromirrors' speed can be controlled even further when the cover glass is removed.

6.2 Bragg-Selective Retroreflective Paint

It can be seen that having a high-reflectivity target helps in their detection by LiDAR devices. In an automotive application, this is not always the case. However, cars with paint that is near-retroreflective at near-infrared wavelengths would be seen better. A possible method would be Bragg-selective and retroreflective microbeads mixed with car paint that reflect 905nm light back to a LiDAR system. This would allow the LiDAR system to see the vehicle from a much further distance. Having the paint be Bragg-selective would mean that it would only be retroreflective for specific wavelengths, preventing safety hazards with sunlight. Materials have been studied that control the reflectance of light at specific wavelengths and at steep angles of incidence, particularly for the application of beam splitters [17]. Similar concepts can be applied to a particular paint or coating for vehicles to be highly reflective at 905nm.

CHAPTER 7

CONCLUSION

Multi-Beam Multi-Pulse Single-Chip DMD LIDAR provides a unique pathway to long distance range finding by leveraging the commercially available Digital Micromirror Device. Since the introduction of the legacy Single-Chip LiDAR system, methods have been effectively implemented to the system to expand its speed and angular resolution. As a proof of concept, 15 points of range finding is demonstrated by using three 905nm pulsed laser diode modules. Multiple-pulse scanning is also demonstrated as a pathway to maximize the speed of the system. With a chip-mounted array of laser diodes and state of the art detectors, the system provides a solution for long range, high speed, and high angular resolution distance measurement without bulky scanning/receiving optics. Additionally, new concepts such as dynamic interferometry, have resulted from this research, proving the flexibility of the DMD as a beam steering device.

APPENDIX A
PUBLICATIONS

In Revision

Joshua Rodriguez, Braden Smith, Brandon Hellman , Heejoo Choi , Guanghao Chen, Dae Wook Kim, Youngsik Kim, and Yuzuru Takashima, “Single-chip lidar by multi-order and multi-pulse beam steering with digital micro mirror device,” Submitted for Optics Express, Jan 2019.

Accepted

Joshua Rodriguez, Brandon Hellman, Braden Smith, Heejoo Choi, Guanghao Chen, Youngsik Kim, Dae Wook Kim, and Yuzuru Takashima, “Multi-order Laser Beam Steering with Digital Micro Mirror Device for High-speed LIDARs,” accepted for CLEO 2019.

Published

Joshua Rodriguez, Eunmo Kang, Brandon Hellman, Guanghao Chen, Braden Smith, Adley Gin, Alonzo Espinoza, and Yuzuru Takashima, “Beam steering by digital micro-mirror device for multi-beam and single-chip LIDAR,” Proc. SPIE. 10757, Optical Data Storage 2018: Industrial Optical Devices and Systems.

Joshua Rodriguez, Braden Smith, Brandon Hellman, Adley Gin, Alonzo Espinoza, Yuzuru Takashima, “Multi-beam and single-chip LIDAR with discrete beam-steering by digital micromirror device,” Proc. SPIE 10526, Physics and Simulation of Optoelectronic Devices XXVI, 105260U (23 February 2018); doi: 10.1117/12.2287485; <https://doi.org/10.1117/12.2287485>

Presentations

Joshua Rodriguez, Eunmo Kang, Brandon Hellman, Guanghao Chen, Braden Smith, Adley Gin, Alonzo Espinoza, and Yuzuru Takashima, “Beam steering by digital micro-mirror device for multi-beam and single-chip LIDAR,” Proc. SPIE. 10757, Optical Data Storage 2018: Industrial Optical Devices and Systems.

Joshua Rodriguez, Braden Smith, Brandon Hellman, Adley Gin, Alonzo Espinoza, Yuzuru Takashima, “Multi-beam and single-chip LIDAR with discrete beam-steering by digital micromirror device,” Proc. SPIE 10526, Physics and Simulation of Optoelectronic Devices XXVI, 105260U (23 February 2018); doi: 10.1117/12.2287485; <https://doi.org/10.1117/12.2287485>

APPENDIX B

CODE

B.1 Multi-Pulse LiDAR

```

#include "arduino2.h" // include the fast I/O 2 functions
#include "SPI.h" // Serial Peripheral Interface

int LD = 0; //Number of laser diodes
int Num_ave = 1; // number of averages to perform for each position
long TOF_sum1 = 0;
long TOF_sum2 = 0;
float TOF_ave_value1 = 0;
float TOF_ave_value2 = 0;
long TDC_mask = 0x7FFFFFF; // mask to remove unwanted bits from read
long TOF1 = 0; // Output from TDC
long TOF2 = 0; // Output from TDC
long byte1 = 0; // 3 bytes for output from TDC
long byte2 = 0;
long byte3 = 0;
float Time1 = 0; //TOF time measurements
float Time2 = 0; //TOF time measurements
float Distance; // to hold distance measurements for each frame

const GPIO_pin_t CS_DL = DP10; // Chip select pin for Delay Line (active
high)
const GPIO_pin_t Enable = DP4; // Chip enable pin for TDC (Active high)
const GPIO_pin_t CS_TDC = DP5; // Chip select for TDC (active low)
const GPIO_pin_t LaserOut_1 = DP14; // Pin output to select laser diode 1
const GPIO_pin_t LaserOut_2 = DP15; // Pin output to select laser diode 2
const GPIO_pin_t LaserOut = DP9; // Initialize LaserOut pin
const GPIO_pin_t DMDPin = DP2; // Pin used to flip the DMD
int mirror_flip_pin = 7; // Switch used to flip mirrors
int continue_pin = 6; // Switch used to continue after mirror flip

int LaserOut_select = 0;
int Init_mirror = false;
int Init_continue = false;
int mirror_state = false;
int Delay1023 = 0; // 0 to 255

```

```

int DelayNOP = 0;          // 0 to 32
int DelayUS = 0;          // Around 220us
int Diff1023 = 0;
int DiffDelayNOP = 0;
int n = 0;

//Calibration
long Calibration_1 = 0;
long Calibration_2 = 0;
float CalCount = 0;
float normLSB = 0;

// Subroutine Definitions
void MirrorInit();
void InitializePins();
void InitalizeTDC();

#define NOP __asm__ __volatile__ ("nop\n\t")

void setup() {
  Serial.begin(115200);

  SPI.begin();
  SPI.beginTransaction(SPISettings(100000, MSBFIRST, SPI_MODE0));
  //   data transfer rate
  //   MSB sent first
  //   SPI Mode1: Data output on clock rising edge

  // Set DS1023 delay chip to 0 delay
  digitalWrite2f(CS_DL, HIGH);
  SPI.transfer(B00000000);
  digitalWrite2f(CS_DL, LOW);

  InitializePins();
  MirrorInit();
  InitializeTDC();

  //noInterrupts();
} // End setup-----

// DMD must be set to white/black image to start off with
// Delay on DS1023 must be set the correct amount to give 8ns pulse to LD
void loop() {
  digitalWrite2f(LaserOut_1, HIGH);

```

```

digitalWrite2f(LaserOut_2, HIGH);
Serial.print("\n");
// Spot +2
Delay1023 = 0;    //0 to 255
DelayNOP = 30;    //0 to 32
DelayUS = 218;
LD = 0;
Get_Distance();

// Spot +1
Delay1023 = 0;    //0 to 255
DelayNOP = 20;    //0 to 32
DelayUS = 218;
LD = 0;
Get_Distance();

// Spot 0
Delay1023 = 0;    //0 to 255
DelayNOP = 11;    //0 to 32
DelayUS = 218;
LD = 0;
Get_Distance();

// Spot -1
Delay1023 = 0;    //0 to 255
DelayNOP = 3;     //0 to 32
DelayUS = 218;
LD = 0;
Get_Distance();

// Spot -2
Delay1023 = 0;    //0 to 255
DelayNOP = 10;    //0 to 32
DelayUS = 217;
LD = 0;
Get_Distance();

} // End Main Loop-----

void InitializePins() {
  pinMode2f(DP9, OUTPUT);
  digitalWrite2f(DP9, LOW);
  pinMode2f(DP11, OUTPUT);
  digitalWrite2f(DP11, LOW);
}

```

```

pinMode2f(DP12, OUTPUT);
digitalWrite2f(DP12, LOW);
pinMode2f(LaserOut_1, OUTPUT);
digitalWrite2f(LaserOut_1, LOW);
pinMode2f(LaserOut_2, OUTPUT);
digitalWrite2f(LaserOut_2, LOW);

pinMode2f(CS_DL, OUTPUT);
digitalWrite2f(CS_DL, LOW);

pinMode2f(CS_TDC, OUTPUT);
digitalWrite2f(CS_TDC, HIGH);

pinMode(mirror_flip_pin, INPUT);
pinMode(continue_pin, INPUT);

pinMode2f(Enable, OUTPUT);
digitalWrite2f(Enable, LOW);
} // End InitializePins-----

void MirrorInit() {
  // for (int n = 0; n < LD; n++) {
  Init_continue = digitalRead(continue_pin);
  Init_mirror = digitalRead(mirror_flip_pin);

  while (digitalRead(continue_pin) == Init_continue) {
    mirror_state = digitalRead(mirror_flip_pin);
    if (mirror_state != Init_mirror) {
      // Flip the mirrors once
      Init_mirror = mirror_state;
      digitalWrite2f(DMDPin, HIGH);
      delayMicroseconds(400);
      digitalWrite2f(DMDPin, LOW);
      delayMicroseconds(400);
    } // End If statement
    // Send pulse to laser
    digitalWrite2f(LaserOut, HIGH);
    NOP;
    digitalWrite2f(LaserOut, LOW);

    delay(1);

    //Disable interrupts
    noInterrupts();

```

```

    //   } //End for loop
  } // End while loop
} // End MirrorInit-----

void InitializeTDC() {
  delay(3);      // Delay for TDC chip to start-up
  digitalWrite2f(Enable, HIGH);
  delay(3);

  // Initialize delay chip to constant delay
  SPI.beginTransaction(SPI_Settings(20000000, MSBFIRST, SPI_MODE0)); //
    SPI_MODE0 for TDC

  // Setup TDC7200 chip
  digitalWrite2f(CS_TDC, LOW); // Config 2
  SPI.transfer(0b01000001); // Address + write
  //SPI.transfer(0b00000000); //..calibration2_periods=2
  SPI.transfer(0b01000001); // Calibration2_periods = 10
  // Perform 1 measurement cycle (No multi cycle averaging mode)
  // Single Stop mode
  digitalWrite2f(CS_TDC, HIGH);

  // set stop mask to 1 clock cycle
  digitalWrite2f(CS_TDC, LOW); // Counter stop mask high
  SPI.transfer(0b01001000); // Address + write
  SPI.transfer(0);          // Write
  digitalWrite2f(CS_TDC, HIGH);

  digitalWrite2f(CS_TDC, LOW); // Counter stop mask low
  SPI.transfer(0b01001001); // Address + write
  SPI.transfer(0);          // Write
  digitalWrite2f(CS_TDC, HIGH);

  SPI.endTransaction();
} // End TDC Init-----

void Get_Distance() {
  noInterrupts();
  digitalWrite2f(LaserOut_1, HIGH);
  digitalWrite2f(LaserOut_2, HIGH);

  TOF_sum1 = 0;
  TOF_sum2 = 0;

```

```

for (int Avg_index = 0; Avg_index < Num_ave; Avg_index++) {
    // Set delay on optional 2nd delay line to fine tune mirror position
    digitalWrite2f(CS_DL, HIGH);
    SPI.transfer(Delay1023);
    digitalWrite2f(CS_DL, LOW);

    // Calibration sequence
    digitalWrite2f(CS_TDC, LOW);           // Calibration 1
    SPI.transfer(0x1B);                    // Address for calibration 1
        register
    byte3 = SPI.transfer(0);              // upper
    byte2 = SPI.transfer(0);              // middle
    byte1 = SPI.transfer(0);              // lower
    digitalWrite2f(CS_TDC, HIGH);
    Calibration_1 = (byte3 << 16) + (byte2 << 8) + byte1;
    Calibration_1 = Calibration_1 & TDC_mask;

    digitalWrite2f(CS_TDC, LOW);           // Calibration 2
    SPI.transfer(0x1C);                    // Address for calibration 2
        register
    byte3 = SPI.transfer(0);              // upper
    byte2 = SPI.transfer(0);              // middle
    byte1 = SPI.transfer(0);              // lower
    digitalWrite2f(CS_TDC, HIGH);
    Calibration_2 = (byte3 << 16) + (byte2 << 8) + byte1;
    Calibration_2 = Calibration_2 & TDC_mask;

    digitalWrite2f(CS_TDC, LOW); // Config 1 - initiate measurement
    //SPI.transfer(0b01000000); // Address + write
    //SPI.transfer(0b10000001); // Write (perform calibration every
        interrupt - 0b10000000). Middle 1 bit is 1/0 falling/rising edge
    SPI.transfer16(0x4081); // Equivalent to the two lines above
    digitalWrite2f(CS_TDC, HIGH);

    // Send high signal to DMD to flip
    digitalWrite2f(DMDPin, HIGH);

    // Wait a critical time (adding the optional delay line time)
    delayMicroseconds(DelayUS); //16 NOP's per 1 us delay

    switch (DelayNOP) { //switch statement delays at increments of 62.5ns
        case (0):

            break;

```



```

    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    break;
case (27):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP;
    break;
case (28):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP; NOP;
    break;
case (29):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP; NOP; NOP;
    break;
case (30):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP; NOP; NOP; NOP;
    break;
case (31):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP; NOP; NOP; NOP; NOP;
    break;
case (32):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP; NOP; NOP; NOP; NOP; NOP;
    break;
} //END SWITCH-CASE STATEMENT

digitalWrite2f(LaserOut_1, LOW);
digitalWrite2f(LaserOut, HIGH);
NOP;
digitalWrite2f(LaserOut, LOW);
digitalWrite2f(LaserOut_1, HIGH);

digitalWrite2f(LaserOut_2, LOW);
digitalWrite2f(LaserOut, HIGH);

```

```

NOP;
digitalWrite2f(LaserOut, LOW);
digitalWrite2f(LaserOut_2, HIGH);

interrupts();

// Read Time 1
digitalWrite2f(CS_TDC, LOW);    // Read signal
SPI.transfer(0x10);             // TIME1 register address
byte3 = SPI.transfer(0x00);     // read 8 upper bits
byte2 = SPI.transfer(0x00);     // read 8 middle bits
byte1 = SPI.transfer(0x00);     // read 8 lower bits
digitalWrite2f(CS_TDC, HIGH);
TOF1 = (byte3 << 16) + (byte2 << 8) + byte1;
TOF1 = TOF1 & TDC_mask;
TOF_sum1 = TOF_sum1 + TOF1;

// Calculate Time
CalCount = ((float)Calibration_2 - (float)Calibration_1) / (10 - 1); //
           //(N_Periods - 1) (See setup for N_periods)
// CalCount = (float)Calibration_2 - (float)Calibration_1;
normLSB = (122 / CalCount); // period of clock in ns (8.1965MHz = 122
                             ns)
// Time1 = (float)TOF1 * normLSB * 10.0; // time in ns
Time1 = (float)TOF_sum1 / (float)Num_ave;

// Time = 0;
// TOF = 0;
Serial.print(Time1);
Serial.print(" ");

// Read Time 2
digitalWrite2f(CS_TDC, LOW);    // Read signal
SPI.transfer(0x12);             // TIME2 register address
byte3 = SPI.transfer(0x00);     // read 8 upper bits
byte2 = SPI.transfer(0x00);     // read 8 middle bits
byte1 = SPI.transfer(0x00);     // read 8 lower bits
digitalWrite2f(CS_TDC, HIGH);
TOF2 = (byte3 << 16) + (byte2 << 8) + byte1;
TOF2 = TOF2 & TDC_mask;
TOF_sum2 = TOF_sum2 + TOF2;
// Time2 = (float)TOF2 * normLSB * 10.0; // time in ns
Time2 = (float)TOF_sum2 / (float)Num_ave;
Serial.print(Time2);

```

```

Serial.print(" ");
  // Send pulse to DMD with appropriate pauses to reset mirrors
  digitalWrite2f(DMDPin, LOW);
  delayMicroseconds(300);
  digitalWrite2f(DMDPin, HIGH);
  delayMicroseconds(300);
  digitalWrite2f(DMDPin, LOW);
  delayMicroseconds(300);
} // End of averaging For Loop
} // End Get_Distance-----

```

B.2 Multi-Beam LiDAR

```

#include "arduino2.h" // include the fast I/O 2 functions
#include "SPI.h" // Serial Peripheral Interface

int LD = 3; //Number of laser diodes
int Num_ave = 10; // number of averages to perform for each position
long TOF_sum = 0;
float TOF_ave_value = 0;
long TDC_mask = 0x7FFFFFFF; // mask to remove unwanted bits from read
long TOF = 0; // Output from TDC
long byte1 = 0; // 3 bytes for output from TDC
long byte2 = 0;
long byte3 = 0;

const GPIO_pin_t CS_DL = DP10; // Chip select pin for Delay Line (active
high)
const GPIO_pin_t Enable = DP4; // Chip enable pin for TDC (Active high)
const GPIO_pin_t CS_TDC = DP5; // Chip select for TDC (active low)
const GPIO_pin_t LaserOut_1 = DP14; // Pin output to select laser diode 1
const GPIO_pin_t LaserOut_2 = DP15; // Pin output to select laser diode 2
const GPIO_pin_t LaserOut_3 = DP16; // Pin output to select laser diode 3
const GPIO_pin_t LaserOut = DP9; // Initialize LaserOut pin
const GPIO_pin_t DMDPin = DP8; // Pin used to flip the DMD
int mirror_flip_pin = 7; // Switch used to flip mirrors
int continue_pin = 6; // Switch used to continue after mirror flip

int LaserOut_select = 0;
int Init_mirror = false;
int Init_continue = false;
int mirror_state = false;
int Delay1023 = 0; // 0 to 255
int DelayNOP = 0; // 0 to 32

```

```

int DelayUS = 0;          // Around 220us
int n = 0;

// Subroutine Definitions
void MirrorInit();
void InitializePins();
void InitalizeTDC();

#define NOP __asm__ __volatile__ ("nop\n\t")

void setup() {
  Serial.begin(115200);

  SPI.begin();
  SPI.beginTransaction(SPISettings(100000, MSBFIRST, SPI_MODE0));
  //   data transfer rate
  //   MSB sent first
  //   SPI Mode1: Data output on clock rising edge

  // Set DS1023 delay chip to 0 delay
  digitalWrite2f(CS_DL, HIGH);
  SPI.transfer(B00000000);
  digitalWrite2f(CS_DL, LOW);

  InitializePins();
  MirrorInit();
  InitializeTDC();

  noInterrupts();
} // End setup-----

// DMD must be set to white/black image to start off with
// Delay on DS1023 must be set the correct amount to give 8ns pulse to LD
void loop() {
  Serial.print("\n");
  digitalWrite2f(LaserOut_1, HIGH);
  digitalWrite2f(LaserOut_2, HIGH);
  digitalWrite2f(LaserOut_3, HIGH);

  // Send high signal to DMD to flip
  digitalWrite2f(DMDPin, HIGH);

  // Wait a critical time (adding the optional delay line time)
  delayMicroseconds(DelayUS); //16 NOP's per 1 us delay

```

```

// Spot -1
Delay1023 = 128;    //0 to 255
DelayNOP = 6;      //0 to 32
DelayUS = 218;
digitalWrite2f(LaserOut_1, LOW);
Get_Distance();
digitalWrite2f(LaserOut_1, HIGH);

// Spot 0
Delay1023 = 0;     //0 to 255
DelayNOP = 7;      //0 to 32
DelayUS = 0;
digitalWrite2f(LaserOut_2, LOW);
Get_Distance();
digitalWrite2f(LaserOut_2, HIGH);

// Spot +1
Delay1023 = 0;     //0 to 255
DelayNOP = 9;      //0 to 32
DelayUS = 0;
digitalWrite2f(LaserOut_3, LOW);
Get_Distance();
digitalWrite2f(LaserOut_3, HIGH);

// Send pulse to DMD with appropriate pauses to reset mirrors
digitalWrite2f(DMDPin, LOW);
delayMicroseconds(400);
digitalWrite2f(DMDPin, HIGH);
delayMicroseconds(400);
digitalWrite2f(DMDPin, LOW);
delayMicroseconds(400);
} // End Main Loop-----

void InitializePins() {
  pinMode2f(DP9, OUTPUT);
  digitalWrite2f(DP9, LOW);
  pinMode2f(DP11, OUTPUT);
  digitalWrite2f(DP11, LOW);
  pinMode2f(DP12, OUTPUT);
  digitalWrite2f(DP12, LOW);
  pinMode2f(LaserOut_1, OUTPUT);
  digitalWrite2f(LaserOut_1, LOW);
  pinMode2f(LaserOut_2, OUTPUT);

```

```

digitalWrite2f(LaserOut_2, LOW);
pinMode2f(LaserOut_3, OUTPUT);
digitalWrite2f(LaserOut_3, LOW);
pinMode2f(CS_DL, OUTPUT);
digitalWrite2f(CS_DL, LOW);
pinMode2f(CS_TDC, OUTPUT);
digitalWrite2f(CS_TDC, HIGH);
pinMode(mirror_flip_pin, INPUT);
pinMode(continue_pin, INPUT);
pinMode2f(Enable, OUTPUT);
digitalWrite2f(Enable, LOW);
} // End InitializePins-----

void MirrorInit() {
  // for (int n = 0; n < LD; n++) {
  Init_continue = digitalRead(continue_pin);
  Init_mirror = digitalRead(mirror_flip_pin);

  while (digitalRead(continue_pin) == Init_continue) {
    mirror_state = digitalRead(mirror_flip_pin);
    if (mirror_state != Init_mirror) {
      // Flip the mirrors once
      Init_mirror = mirror_state;
      digitalWrite2f(DMDPin, HIGH);
      delayMicroseconds(400);
      digitalWrite2f(DMDPin, LOW);
      delayMicroseconds(400);
    } // End If statement
    //      Send pulse to laser
    digitalWrite2f(LaserOut, HIGH);
    NOP;
    digitalWrite2f(LaserOut, LOW);
    delay(1);
    //Disable interrupts
    noInterrupts();
    //    } //End for loop
  } // End while loop
} // End MirrorInit-----

void InitializeTDC() {
  delay(3);      // Delay for TDC chip to start-up
  digitalWrite2f(Enable, HIGH);
  delay(3);

```

```

// Initialize delay chip to constant delay
SPI.beginTransaction(SPISettings(20000000, MSBFIRST, SPI_MODE0)); //
    SPI_MODE0 for TDC

// Setup TDC7200 chip
digitalWrite2f(CS_TDC, LOW); // Config 2
SPI.transfer(0b01000001); // Address + write
//SPI.transfer(0b00000000); //..calibration2_periods=2
SPI.transfer(0b01000000); // Calibration2_periods = 10
// Perform 1 measurement cycle (No multi cycle averaging mode)
// Single Stop mode
digitalWrite2f(CS_TDC, HIGH);

// set stop mask to 1 clock cycle
digitalWrite2f(CS_TDC, LOW); // Counter stop mask high
SPI.transfer(0b01001000); // Address + write
SPI.transfer(0); // Write
digitalWrite2f(CS_TDC, HIGH);

digitalWrite2f(CS_TDC, LOW); // Counter stop mask low
SPI.transfer(0b01001001); // Address + write
SPI.transfer(0); // Write
digitalWrite2f(CS_TDC, HIGH);

SPI.endTransaction();
} // End TDC Init-----

void Get_Distance() {
    TOF_sum = 0;

    for (int Avg_index = 0; Avg_index < Num_ave; Avg_index++) {
        // Set delay on optional 2nd delay line to fine tune mirror position
        digitalWrite2f(CS_DL, HIGH);
        SPI.transfer(Delay1023);
        digitalWrite2f(CS_DL, LOW);

        // Calibration sequence for TDC7200
        SPI.beginTransaction(SPISettings(20000000, MSBFIRST, SPI_MODE0));
        digitalWrite2f(CS_TDC, LOW); // Calibration 1
        SPI.transfer(0x1B); // Address for calibration 1
            register
        byte3 = SPI.transfer(0); // upper
        byte2 = SPI.transfer(0); // middle
        byte1 = SPI.transfer(0); // lower
    }
}

```



```

digitalWrite2f(CS_TDC, HIGH);

digitalWrite2f(CS_TDC, LOW);           // Calibration 2
SPI.transfer(0x1C);                   // Address for calibration 2
    register
byte3 = SPI.transfer(0);               // upper
byte2 = SPI.transfer(0);               // middle
byte1 = SPI.transfer(0);               // lower
digitalWrite2f(CS_TDC, HIGH);

digitalWrite2f(CS_TDC, LOW); // Config 1 - initiate measurement
//SPI.transfer(0b01000000); // Address + write
//SPI.transfer(0b10000001); // Write (perform calibration every
    interrupt - 0b10000000)
SPI.transfer16(0x4081);
digitalWrite2f(CS_TDC, HIGH);

switch (DelayNOP) { //switch statement delays at increments of 62.5ns
    case (0):
        break;
    case (1):
        NOP;
        break;
    case (2):
        NOP; NOP;
        break;
    case (3):
        NOP; NOP; NOP;
        break;
    case (4):
        NOP; NOP; NOP; NOP;
        break;
    case (5):
        NOP; NOP; NOP; NOP; NOP;
        break;
    case (6):
        NOP; NOP; NOP; NOP; NOP; NOP;
        break;
    case (7):
        NOP; NOP; NOP; NOP; NOP; NOP; NOP;
        break;
    case (8):
        NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
        break;
}

```

```
case (9):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    break;
case (10):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    break;
case (11):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    break;
case (12):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    break;
case (13):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    break;
case (14):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    ;
    break;
case (15):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    ; NOP;
    break;
case (16):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    ; NOP; NOP;
    break;
case (17):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    ; NOP; NOP; NOP;
    break;
case (18):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    ; NOP; NOP; NOP; NOP;
    break;
case (19):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    ; NOP; NOP; NOP; NOP;
    break;
case (20):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    ; NOP; NOP; NOP; NOP; NOP;
    break;
case (21):
```



```

    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP; NOP; NOP; NOP; NOP;
    break;
  case (32):
    NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
      ; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP; NOP;
    NOP; NOP; NOP; NOP; NOP; NOP;
    break;
} //END SWITCH-CASE STATEMENT

digitalWrite2f(LaserOut, HIGH);
NOP;
digitalWrite2f(LaserOut, LOW);

//delay(1);
// Stop signal should hit APD and stop TDC7200 if object present
// Read output from TDC7200
digitalWrite2f(CS_TDC, LOW);    // Read signal
SPI.transfer(0x10);            // TIME1 register address
byte3 = SPI.transfer(0x00);    // read 8 upper bits
byte2 = SPI.transfer(0x00);    // read 8 middle bits
byte1 = SPI.transfer(0x00);    // read 8 lower bits
digitalWrite2f(CS_TDC, HIGH);
TOF = (byte3 << 16) + (byte2 << 8) + byte1;
TOF = TOF & TDC_mask;
TOF_sum = TOF_sum + TOF;
// Wait for mirror to stop flipping
//delayMicroseconds(300);
delayMicroseconds(400);
} // End of averaging For Loop
// Calculate averaged TOF
TOF_ave_value = (float)TOF_sum / (float)Num_ave;
// Display output
Serial.print(TOF_ave_value);
Serial.print(" ");
// } //End of LD for loop
} // End Get_Distance-----

```

REFERENCES

- [1] Smith, B., Hellman, B., Gin, A., Espinoza, A., and Takashima, Y., “Single chip lidar with discrete beam steering by digital micromirror device,” *Opt. Express* **25**, 14732–14745 (Jun 2017).
- [2] Stone, W. C., Stone, W. C., Juberts, M., Dagalakis, N., Stone, J., Gorman, J., Bond, P. J., Secretary, U., and Bement, A. L., “Performance analysis of next-generation lidar for manufacturing, construction, and mobility,” *NIST Interagency/Internal Report (NISTIR)* (2004).
- [3] Cho, A. R., Han, A., Ju, S., Jeong, H., Park, J.-H., Kim, I., Bu, J.-U., and Ji, C.-H., “Electromagnetic biaxial microscanner with mechanical amplification at resonance,” *Opt. Express* **23**, 16792–16802 (Jun 2015).
- [4] Niclass, C., Ito, K., Soga, M., Matsubara, H., Aoyagi, I., Kato, S., and Kagami, M., “Design and characterization of a 256x64-pixel single-photon imager in cmos for a mems-based laser scanning time-of-flight sensor,” *Opt. Express* **20**, 11863–11881 (May 2012).
- [5] DeRose, C. T., Kekatpure, R. D., Trotter, D. C., Starbuck, A., Wendt, J. R., Yacobbi, A., Watts, M. R., Chettiar, U., Engheta, N., and Davids, P. S., “Electronically controlled optical beam-steering by an active phased array of metallic nanoantennas,” *Opt. Express* **21**, 5198–5208 (Feb 2013).
- [6] Dou, R. and Giles, M., “Programmable phase grating and beam steerer by operating a lctv,” in [*Advanced Imaging Technologies and Commercial Applications*], Clark, N. and Gonglewski, J., eds., **2566** (Aug. 1995).
- [7] Davis, S., Rommel, S., Johnson, S., Farca, G., Rebolledo, N., Selwyn, S., and Anderson, M., “Electro-optic steering of a laser beam,” *SPIE Newsroom* (2011).
- [8] Ackerman, E., “Quanergy announces \$ 250 solid-state lidar for cars, robots, and more,” *IEEE Spectrum* **7** (2016).
- [9] Tuantranont, A., Bright, V., Zhang, J., Zhang, W., Neff, J., and Lee, Y., “Optical beam steering using mems-controllable microlens array,” *Sensors and Actuators A: Physical* **91**(3), 363 – 372 (2001). Proceedings of the Technical Digest of the 2000 Solid-State Sensors and Actuators Workshop.
- [10] Inc., M. T., “Mirrorcle technologies mems mirrors–technical overview,” *Gimbal-les Two-Axis Scanning Micromirror Devices technical overview* (2009).

- [11] Sandner, T., Wildenhain, M., Gerwig, C., Schenk, H., Schwarzer, S., and Wölfelschneider, H., “Large aperture mems scanner module for 3d distance measurement,” in [*MOEMS and Miniaturized Systems IX*], **7594**, 75940D (Feb. 2010).
- [12] Moss, R., Yuan, P., Bai, X., Quesada, E., Sudharsanan, R., Stann, B., Dammann, J., Giza, M., and Lawler, W., “Low-cost compact mems scanning lidar system for robotic applications,” in [*Laser Radar Technology and Applications XVII*], **8379**, 837903 (June 2012).
- [13] Lee, X. and Wang, C., “Optical design for uniform scanning in mems-based 3d imaging lidar,” *Appl. Opt.* **54**, 2219–2223 (Mar 2015).
- [14] Instruments, T., “Dlp system optics,” *Application Report DLPA022* (2010).
- [15] <http://www.ti.com/tool/TIDA-01573>.
- [16] https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_ID=10823&ad-image-0#ad-image-0.
- [17] Li, J., Tu, J., and Zhu, Y., “Study on 55°, 905 nm, high reflectivity and visible range beam splitter in cemented cube,” **11** (06 2013).
- [18] Rodriguez, J., Smith, B., Hellman, B., Gin, A., Espinoza, A., and Takashima, Y., “Multi-beam and single-chip lidar with discrete beam steering by digital micromirror device,” **10526** (2018).
- [19] Rodriguez, J., Smith, B., Kang, E., Hellman, B., Chen, G., Gin, A., and Alonzo Espinoza, Y. T., “Beam steering by digital micro-mirror device for multi-beam and single-chip lidar,” **10757** (2018).
- [20] Chen, G., Hellman, B., Rodriguez, J., Smith, B., Gin, A., and Takashima, Y., “Light recycling beam steering on a dmd lidar,” **10757** (2018).