AUTOMATED WRAPPING OF SILVERWARE IN A NAPKIN

By

ARUL SELVAM SIMON JEYAPALAN

Bachelor of Engineering

Government College of Engineering at Salem

Salem, India

2001

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 2005

AUTOMATED WRAPPING OF SILVERWARE IN A NAPKIN

Thesis Approved:

DR. LAWRENCE L HOBEROCK

Thesis Adviser

DR. PRABHAKAR R. PAGILLA

DR. RONALD D. DELAHOUSSAYE

DR. GORDON EMSILE

Dean of the Graduate College

ACKNOWLEDGEMENTS

I would like to extend my sincere gratitude to my Graduate Advisor Dr. Lawrence L. Hoberock, for his constant encouragement, support and innumerable hours of guidance extended throughout my course of graduate program. I sincerely thank Dr. Prabhakar R. Pagilla and Dr. Ronald D. Delahoussaye for serving in my graduate committee.

I would also like to give my special thanks to Mr. Jerry Dale, CEAT Lab Manager and Mr. Brett Riegel, MAE Electronics Shop for their suggestions and assistance. I would like to extend my thanks to Mr. Sandeep Yeri, Ms. Shilpa Nagaraj, Mr. Venugopal Lolla, Mr. Vamshi Peddi, and Mr. Venkatesh Akella for their constant support. I would also like to express a special thanks to Mr. Aswin Ramachandran for his suggestions and assistance in building the Electronic unit of the system.

Finally, I would like to thank all my friends and family members for their unfailing support and constant encouragement in all my pursuits. I would like to dedicate my work to my parents Mr. Jeyapalan Selvanayagam and Mrs. Dulsybai Jeyapalan.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INRODUCTION

1.1 Need for a Silverware Wrapping Machine

The process of dishwashing in commercial establishments is a laborious process involving manual labor to do the bulk of the work. Much of the dishwashing process is monotonous and tedious, often performed in dirty, humid environments, which results in high labor costs, low efficiency and absenteeism (Nagaraj, 2003).

Washing and processing various types of silverware involves distinct stages. Automation of the process is divided into singulation, identification, inspection, sorting, and finally wrapping appropriate silverware in napkins. Singulation is defined as the process of removing individual pieces of silverware from a mixed batch of silverware. A prototype for singulating silverware was devised by Hashimoto (1995). A vision system was employed by Yeri (2003) in order to identify and find the orientation of silverware. The vision system was further enhanced by Lolla (2005) to inspect silverware for cleanliness. Collection of silverware into their respective collection bins was devised by Nagaraj (2003), and robust techniques were developed and implemented by Peddi (2005).

The complete process to identify and sort silverware has been automated in order to increase speed and efficiency and reduce costs. The process of wrapping silverware in a napkin requires placing a set of silverware consisting of a fork, spoon and knife on a napkin, and then rolling them up neatly. Typically, this involves significant human contact, and is repetitive and tedious. Developing an automated napkin wrapping device is needed to significantly reduce the manual labor involved and reduce costs.

1.2 Patent Search

A number of previous investigations into automated napkin wrapping have been reported in the patent literature. A patent search was conducted, and six different patents were found that involve the wrapping of either single or multiple pieces of silverware in a napkin. The search keywords “Silverware Wrapping”, “Flatware Wrapping”, “Silverware Napkin”, and “Flatware Napkin” were used to search for available patents. In what follows, we summarize these patents, whose abstracts are given in appendix A.

U.S. patent 5,469,688 “Method for wrapping silverware in a napkin”, Nov. 1995, describes a device to roll the napkin using the frictional forces between the belt and the napkin. It is doubtful that this device will be able to handle larger cloth napkins to wrap, since the longitudinal length of the napkin is longer than that of the silverware by at least a factor of two. Also, the process is not completely automated because it lacks a mechanism for the napkin to be placed on the conveyor.

U.S. Patent *6,023,908* “Method and apparatus for folding a napkin around an eating utensil”, Feb. 15, 2000, describes a device that does not wrap the silverware in an orderly fashion as expected with a human operator. This system has two hoppers for two different pieces of silverware; it does not indicate the use of a third delivery system, required for the additional silverware. The patent does not provide any information on wrapping a cloth napkin, which is larger in size than paper napkins, posing a whole new set of problems. Furthermore, the system does not indicate the use of any automated technique to deliver the napkin to the wrapping mechanism.

U.S. Patent *6,023,913* "Apparatus and method for wrapping silverware within a napkin", Feb. 15, 2000, and U.S. Patent *6,202,387* "Apparatus and method for banding wrapped silverware", Mar. 20, 2001, involve the same device which is used to wrap and band silverware in a napkin. The device mentioned in these patents does not use cloth napkins. This technique is a crude approach that does not roll the silverware in a napkin with a professional appeal. It may be suitable for paper napkins, but is likely to fail with cloth napkins. The napkin delivery system requires the napkin to be paper, owing to the fact that the napkin has to be somewhat rigid for proper delivery of the napkin to the wrapping area. Also, the silverware is expected to be stacked in an orderly fashion, requiring significant manual labor.

U.S. Patent *6,615,566* "Apparatus and method for automatically wrapping silverware in a napkin", Sep. 9, 2003, does not indicate on how a large cloth napkin will be handled by the system. The system does not have a singulation device, which will be required to minimize human operator intervention.

U.S. Patent *6,837,028* "Automated flatware and napkin assembling apparatus", Jan. 4, 2005, describes a device, suitable only for disposable paper napkins available in rolled form. Also, the singulation device is not efficient; it requires a human operator to stack the silverware in the hopper for wrapping the silverware.

Summary

It is evident from these patents that significant work has been done to construct a device capable of wrapping silverware automatically. None of the above mentioned patents have been commercialized owing to various shortcomings such as,

1. Inability to wrap silverware in a cloth napkin.
2. Inability to truly singulate silverware from a batch of silverware.
3. Significant manual labor required.

Some of these methods are crude and cannot be used to professionally wrap silverware in a napkin, thus giving significant room for improvement in techniques to automate the silverware wrapping process.

1.3 Background Work

The design of a silverware wrapping device was initialized after previous investigators had performed significant work to singulate, inspect and sort silverware in a completely automated process. A singulation device was designed and constructed by Hasimoto (1995), which uses a vibrating motor to vibrate a multilayered hopper containing mixed silverware pieces, discharging them from a slot in the hopper bottom. Subsequent work by Latvala (1999) and Loisson (1999) provided a magnetic conveying device to move the silverware underneath a camera and subsequently to a sorting device. A silverware identification system was designed by Yeri (2003) using a machine vision system to identify silverware pieces and their orientations. An improved identification system, together with inspection, was implemented by Lolla (2005), again using machine vision. A silverware sorting and orientation device was devised by Nagaraj (2003) using orientation and sorting turntables. Since the efficiency of the sorting and orientation device was low, Peddi (2005) designed and constructed a more efficient sorting and orienting device using dynamic meshing and individual sorting of random oppositely oriented silverware pieces. In order to further automate the process, a silverware wrapping device to automatically wrap the silverware in a napkin is desirable.

1.4 Objective

The objective of this thesis is to design and construct a silverware wrapping device that is capable of automatically picking cloth napkins from a napkin bin and wrapping the napkin in an appealing manner with a set of silverware comprised of a knife, fork and spoon.

CHAPTER 2

DESIGN OF SILVERWARE WRAPPING MECHANISM

2.1 Design Considerations

The wrapping process is split into napkin singulation, dropping silverware into the napkin, and folding. Napkins are neatly stacked in a pile, from which a single napkin must be removed each time a set of silverware is to be wrapped. A device to drop a set of silverware on the napkin and a complete set of wrapping mechanisms are required to finish the process. It is necessary for the silverware wrapping process to be continuous to achieve acceptable throughput. Accordingly, it appeared that a conveyor system is required to take the napkin from the napkin singulating device to a silverware dropping mechanism and eventually to a device that finishes the wrapping process. An analysis of the various methods used to wrap silverware in a napkin revealed that it is a general practice to place silverware along one of the diagonals of the napkin and wrap it. After several manual experiments, we found that the best method to wrap silverware in a napkin would consist of the following steps, illustrated in Fig 2.1:

1. Silverware is placed on the napkin along one of its diagonal lengths.
2. The first fold over the silverware is made along the diagonal on which the silverware is placed.
3. The napkin is folded on one of the two sides adjacent to the ends of the silverware.

4. The other side adjacent to the other silverware end is folded.
5. A sequence of continuous folds, or rolls, is made along the same diagonal in which the silverware is placed to complete the wrapping process.

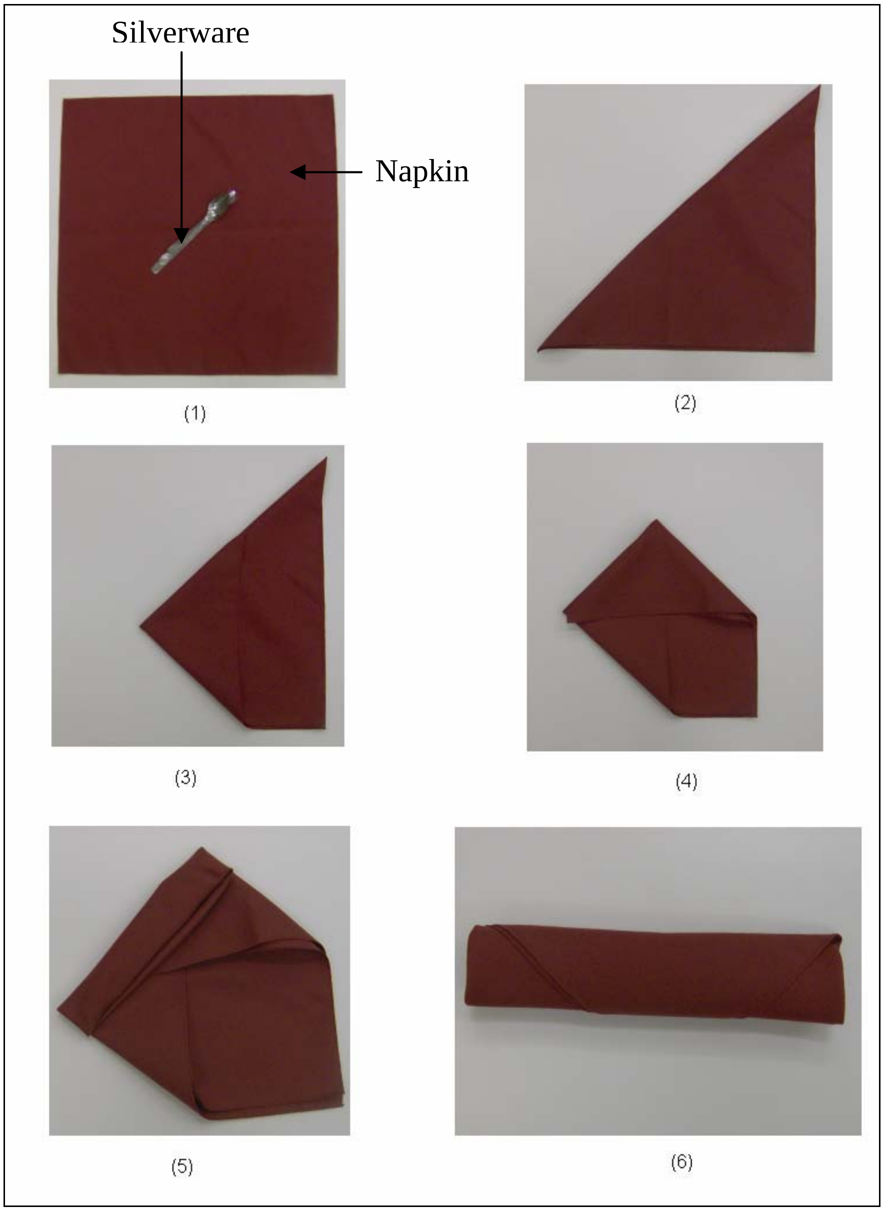


Fig. 2.1 Steps in Wrapping Silverware in a Napkin.

2.2 Design of the Napkin Bin

In this and following sections, figures are for illustration only, and are not drawn to scale. Detailed and dimensioned scaled drawings of all components are given in

Appendix B. The napkin used for the design process was a square cotton dinner napkin measuring 20” in length per side. From the wrapping process described above, the method requires the napkin to be moved to the silverware delivery system. Fig 2.2 shows the position in which silverware is placed along the diagonal of the napkin, with the first fold over the silverware along the silverware diagonal, shown in Fig 2.1. To facilitate this, napkin motion should be perpendicular to this fold. It is hence desirable to pull the napkin out of the napkin stack along one of its diagonal lengths.

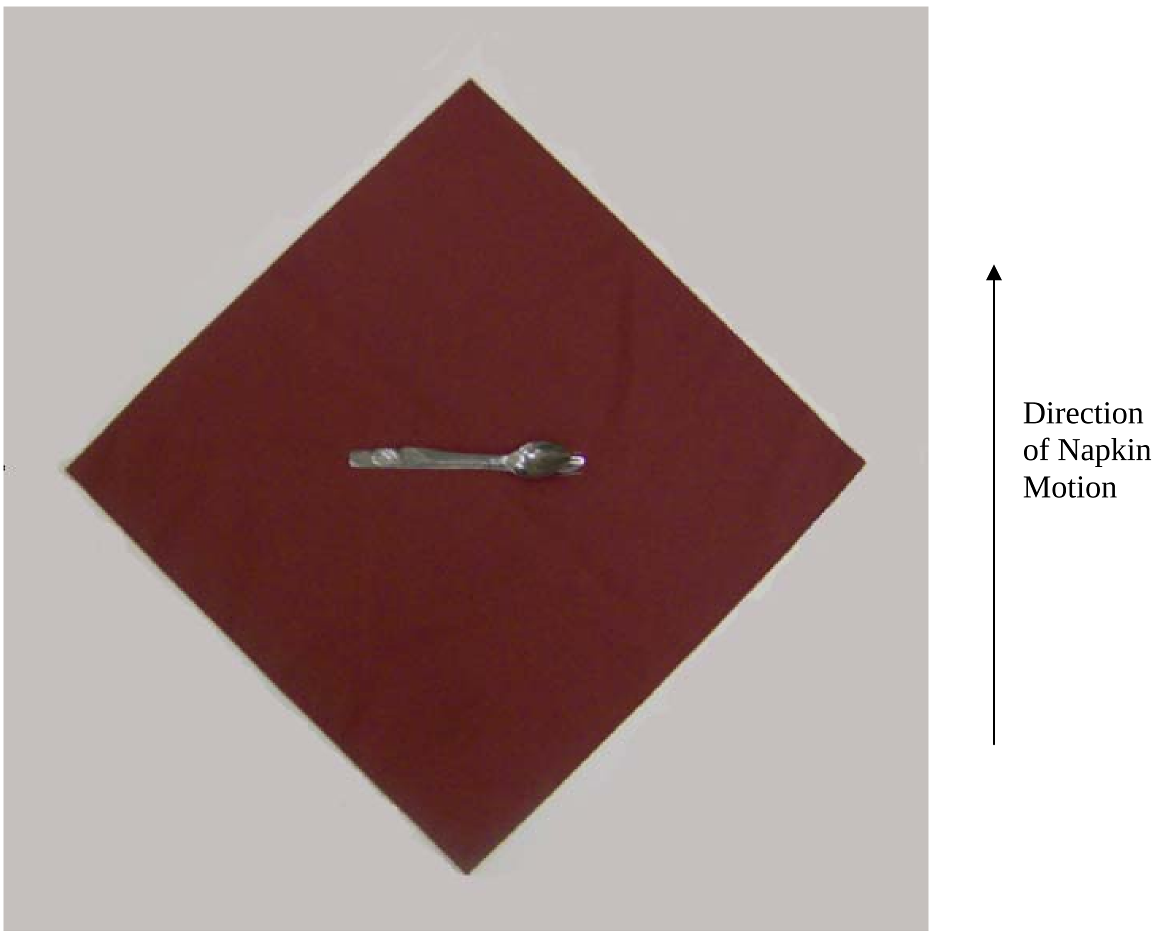


Fig 2.2. Diagonal Position of Napkin with Silverware

Initial manual experiments suggested that napkin singulation could be implemented by using a roller to pull a single napkin from the top of the napkin stack.

This is similar to the method used in desktop laser printers to singulate single sheets of paper from a stack to feed the printer. The roller presses vertically into the stack of napkins placed in the napkin bin as shown in Fig 2.3. The basic idea was that when the friction between the napkin and the roller was greater than the friction between the napkins in the napkin stack, the napkin on the top would be moved from the napkin stack. This process is illustrated in Fig.2.4

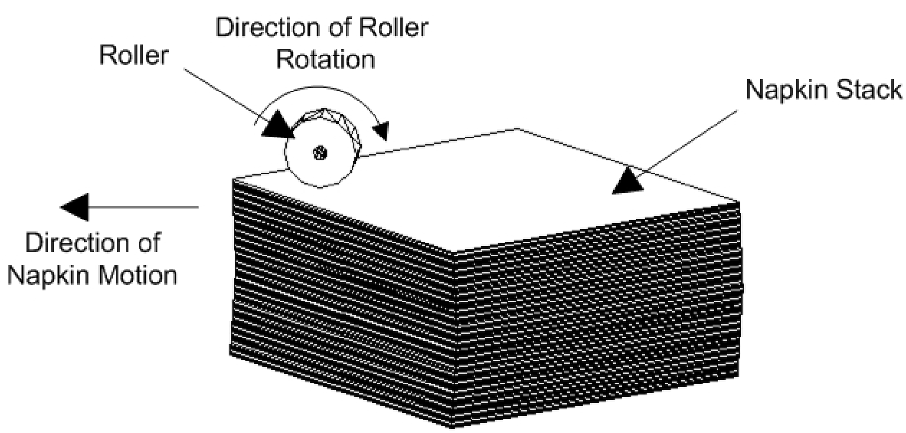


Fig 2.3 Position of Roller over a Stack of napkin

With one end of the napkin hanging out of the napkin bin, shown in Fig 2.4(2), a pickup device was found necessary to pull the partially singulated napkin out of the napkin bin and place it over a conveying mechanism to be taken to the silverware drop mechanism.

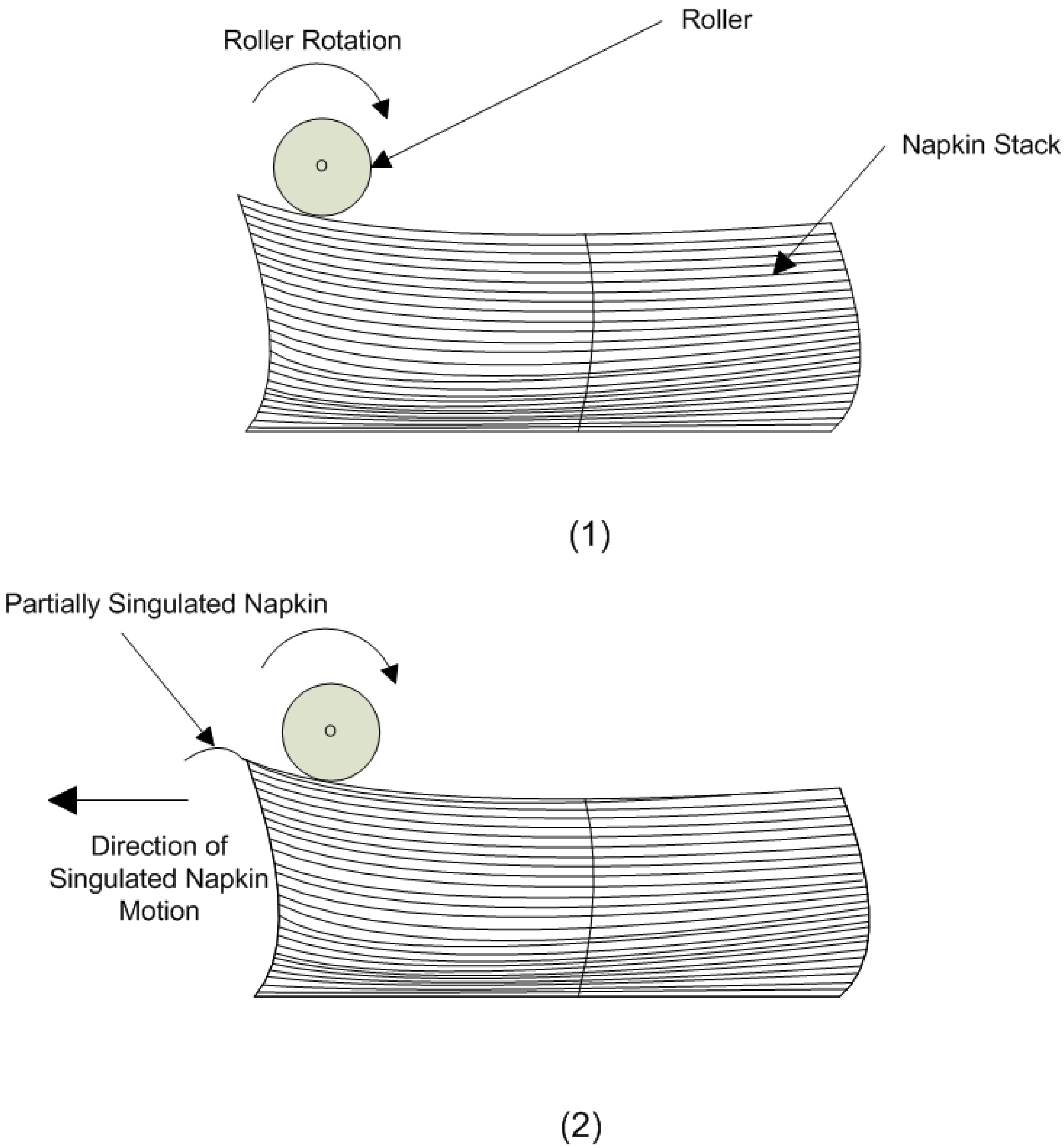


Fig. 2.4 Napkin Singulation Method

Various methods could be used to pick the napkin from the napkin bin and place it on a conveyor. Some ideas considered are discussed below.

As shown in Fig 2.5, one approach would be to place the bin adjacent to the conveyor system to allow the napkin to drop directly on the conveyor. Roller-2 is placed directly over the end of the conveyor system. This roller is moved downwards to secure the napkin between the roller and the conveyor. The pressure between Roller-2 and the moving conveyor secures the napkin between them, such that it is completely pulled from the napkin bin and placed on the conveyor.

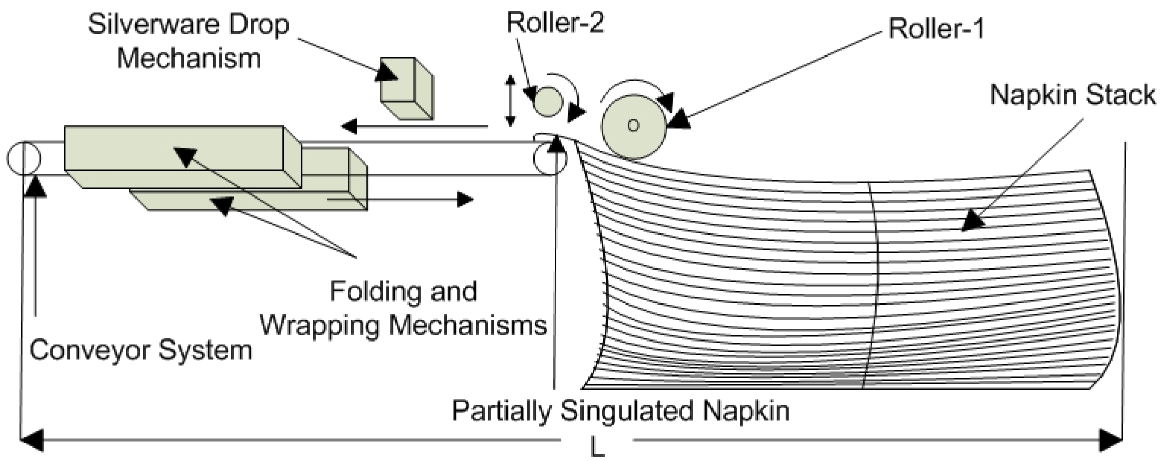


Fig 2.5 Method-I: Conveyor is Placed In-Line With Napkin Bin

This is a very straightforward approach, but the overall length ‘L’ of the complete system would be approximately 10’ with a 20” square napkin moving in its diagonal direction. This would make the device bulky, requiring significant floor space. If the napkin bin is placed underneath the silverware drop and wrapping mechanism, the length of the system could be reduced to approximately 6’. In the following section, with the napkin bin placed beneath the rest of the wrapping mechanism, several approaches are discussed.

A second method for placing the napkin over the conveyor is illustrated in Fig 2.6. This method allows the partially singulated napkin to fall between two rollers (Roller-2 and Roller-3), one of which is powered. Both the rollers are mounted on motorized rotating arms. Once the napkin falls between the two rollers, the arms on which the rollers are fixed lock against one another, and using another motor, they rotate, pulling the napkin between them and placing it over the moving conveyor. Once the rollers are positioned over the conveyor system, the powered roller is actuated to remove the napkin from the napkin bin to the conveyor system.

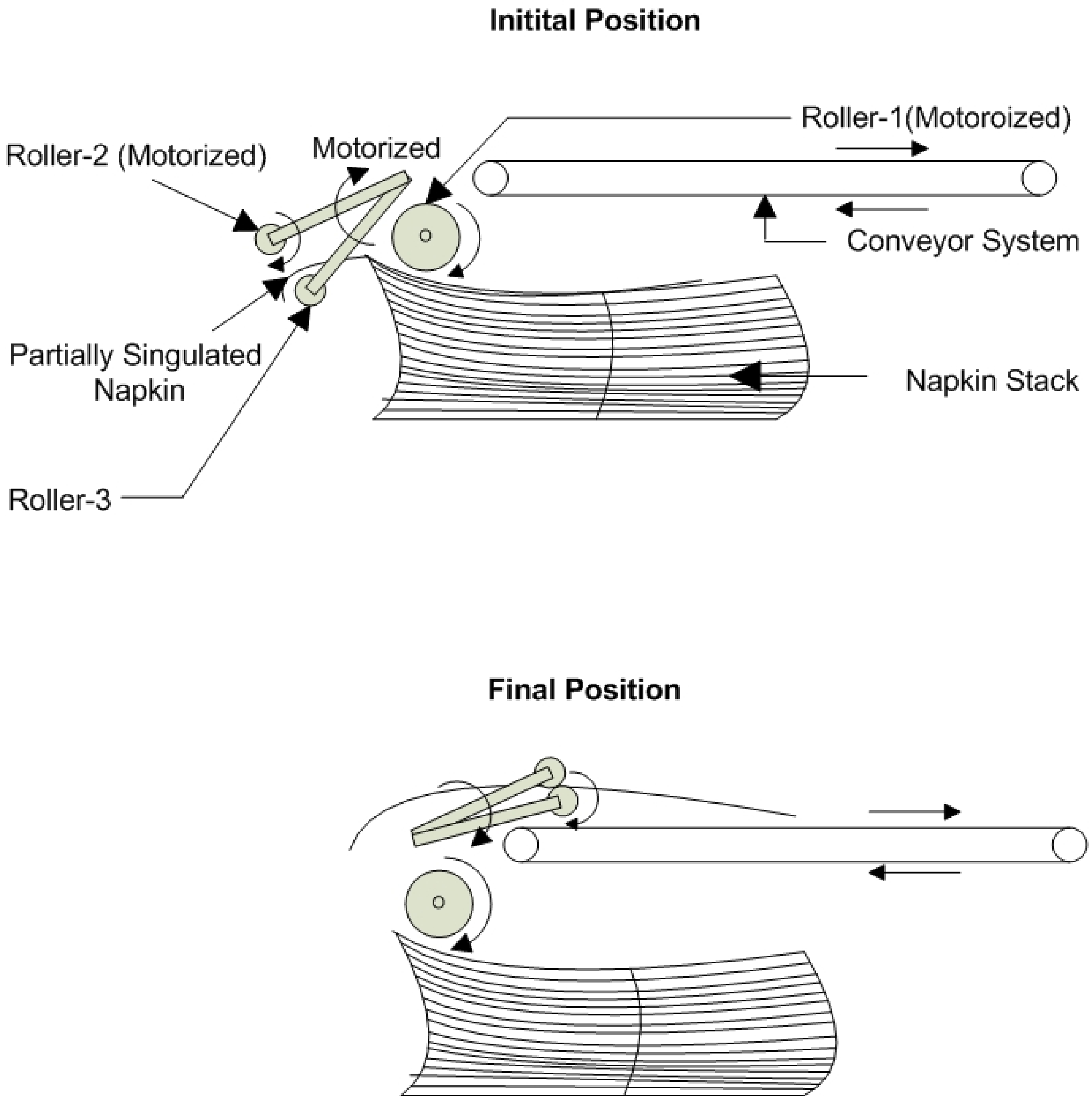


Fig 2.6 Method-II: Conveyor is Placed Above Napkin Bin

This system requires three different motorized drives to completely singulate the napkin and place it on the conveyor. Even though this concept positions the napkin bin beneath the rest of the mechanisms, the system becomes complicated with two additional drives, one to move the arms on which the rollers are placed, and the other to actuate the roller placed on one of the moveable arms.

A third concept places the napkin bin beneath the silverware drop and wrapping mechanisms and is shown in Fig 2.7. This requires a static quadrant of a cylinder, with one edge of the cylinder quadrant at the end of the napkin bin and the other edge at the beginning of the horizontal portion of the conveyor. An unpowered roller (Roller-2) is fixed on a powered rotating arm in order for it to travel along the quadrant of the cylinder. The conveyor belt runs over this static cylindrical quadrant using an additional end roller (Roller-3) at the lower end of the quadrant. Once the tip of a napkin is removed from the bin by Roller-1, Roller-2 on the powered rotating arm secures the napkin between the roller and the belt running over the static cylindrical quadrant. Roller-2 moves on its arm along with the belt on the static cylindrical quadrant, with the napkin between the roller and the belt. Once the roller reaches the discharge end of the cylindrical quadrant, the arm pauses over the belt. This assures that the napkin is completely removed from the napkin bin and placed on the horizontal portion of the conveyor. Once the napkin has cleared Roller-2, the arm rotates back to its initial position to await the next napkin. This mechanism allows the napkin bin to be placed beneath the conveyor system with only one additional drive.

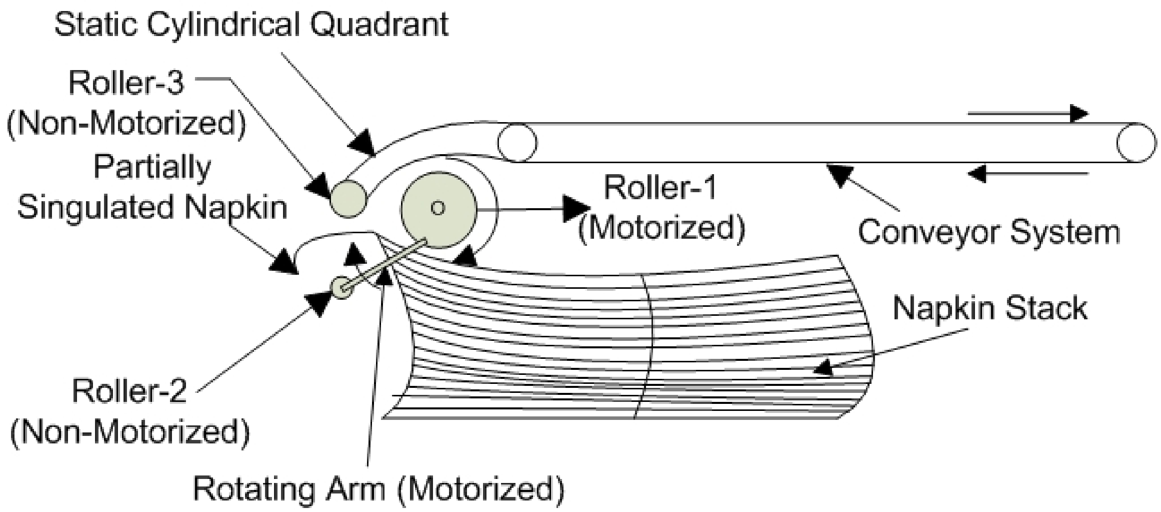


Fig 2.7 Method-III: Conveyor is Placed Above Napkin Bin

For this method to work properly, sufficient contact between Roller-1 and the napkin at the top of the napkin stack must be established. Fig 2.8 illustrates a mechanism proposed to accomplish this. This mechanism uses three leadscrew motors to achieve the vertical motion required to establish sufficient contact between the top napkin in the stack and Roller-1. Flanged nuts are fixed on a tray used to hold the stack of napkins. A leadscrew driven by an electric motor threads through each nut. Rotating the leadscrew moves the tray, and thus moving the stack up or down. All three motors are actuated synchronously to move the napkin stack upwards, and maintain the napkin bin the horizontal position, in order to ensure the top napkin has sufficient contact with the roller. When the top napkin has been removed, the three motors rotate a fixed number of turns to bring the next napkin to the proper position and contact with Roller-2. When the stack is empty the motors are reversed to lower the bin tray to its bottom position, allowing the user to reload napkins into the bin tray.

After considering the pros and cons of these concepts, it was decided to pursue the third concept (Figs 2.7 and 2.8) for implementation and experimentation.

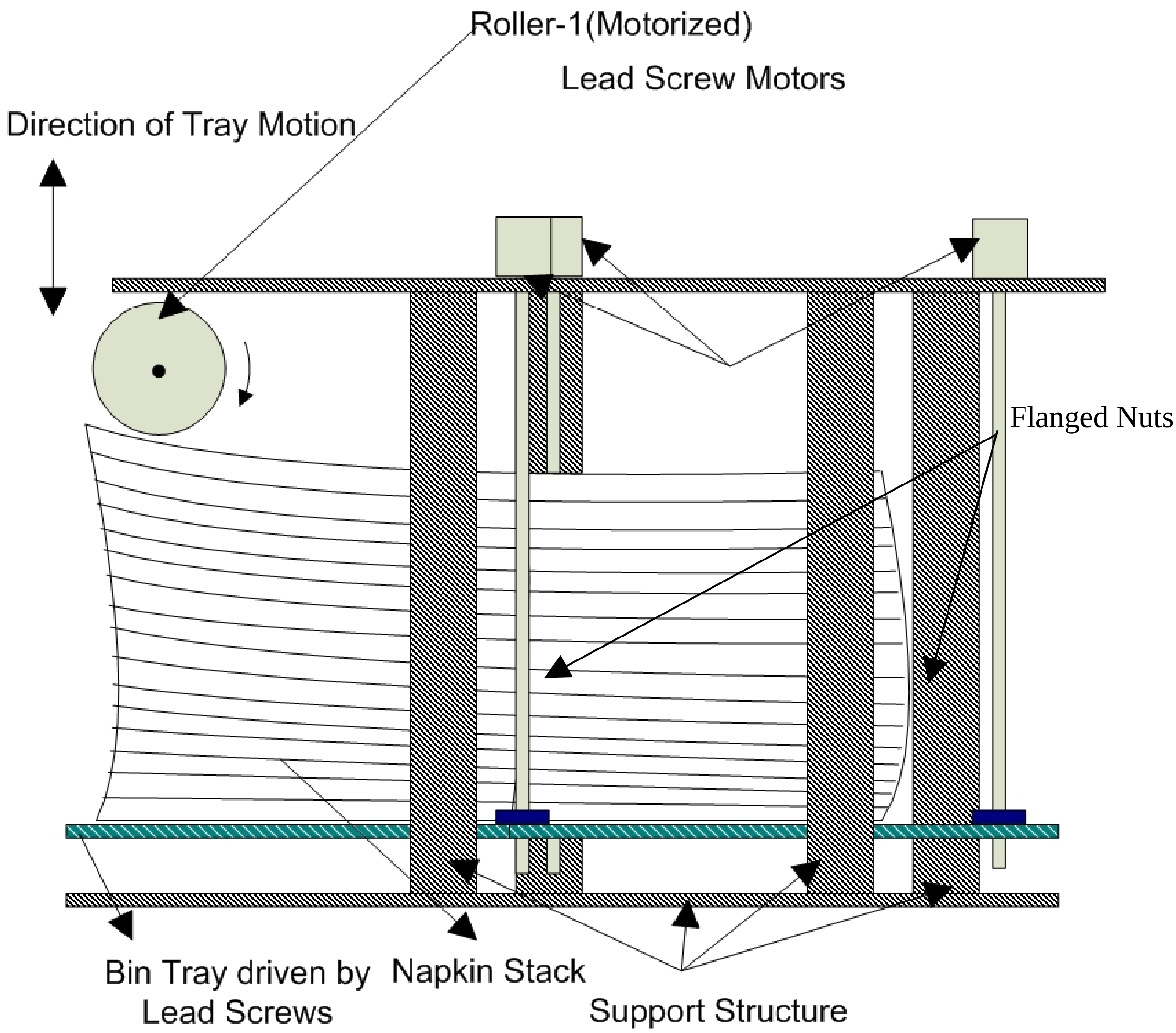


Fig 2.8 Mechanism to Ensure Proper Contact Between the Top Napkin Stack and Roller-1.

2.3 Design of the Silverware Drop Mechanism

After a napkin has been singulated from the napkin bin, a set of silverware is dropped on the napkin using a silverware drop mechanism shown in Fig 2.9. The silverware drop mechanism consists of an open-top hollow box with a hinged bottom. The hinged bottom is held shut by the activated plunger of a solenoid attached by a flexible line (Fig 2.9(a)). Suppose a set of silverware that is to be wrapped is dropped in the hollow box, with bottom closed. When the solenoid is deactivated, gravity causes the hinged bottom to open, dropping the silverware on the diagonal of the napkin, as shown in Fig 2.9(b). For a complete operating system, an automatic mechanism for feeding silverware sets to the drop box would be needed. This, however, was not included in the work herein.

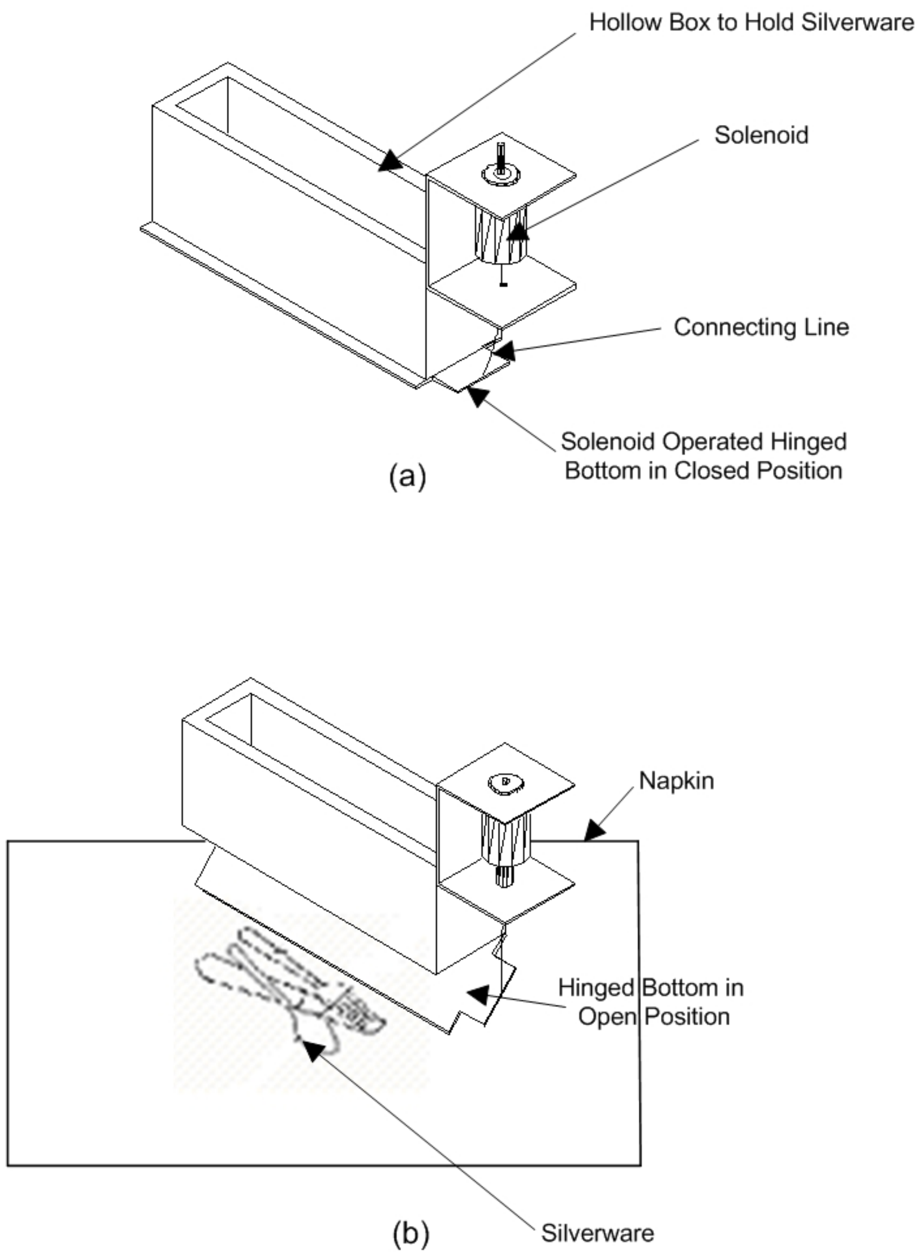


Fig 2.9 Silverware Drop Mechanism

2.4 Design of the Folding Mechanisms

As discussed earlier, four different folds are to be done on the napkin with silverware. The first fold is the longest, and is implemented as the napkin with silverware moves along the conveyor system, shown in Fig 2.10(a).

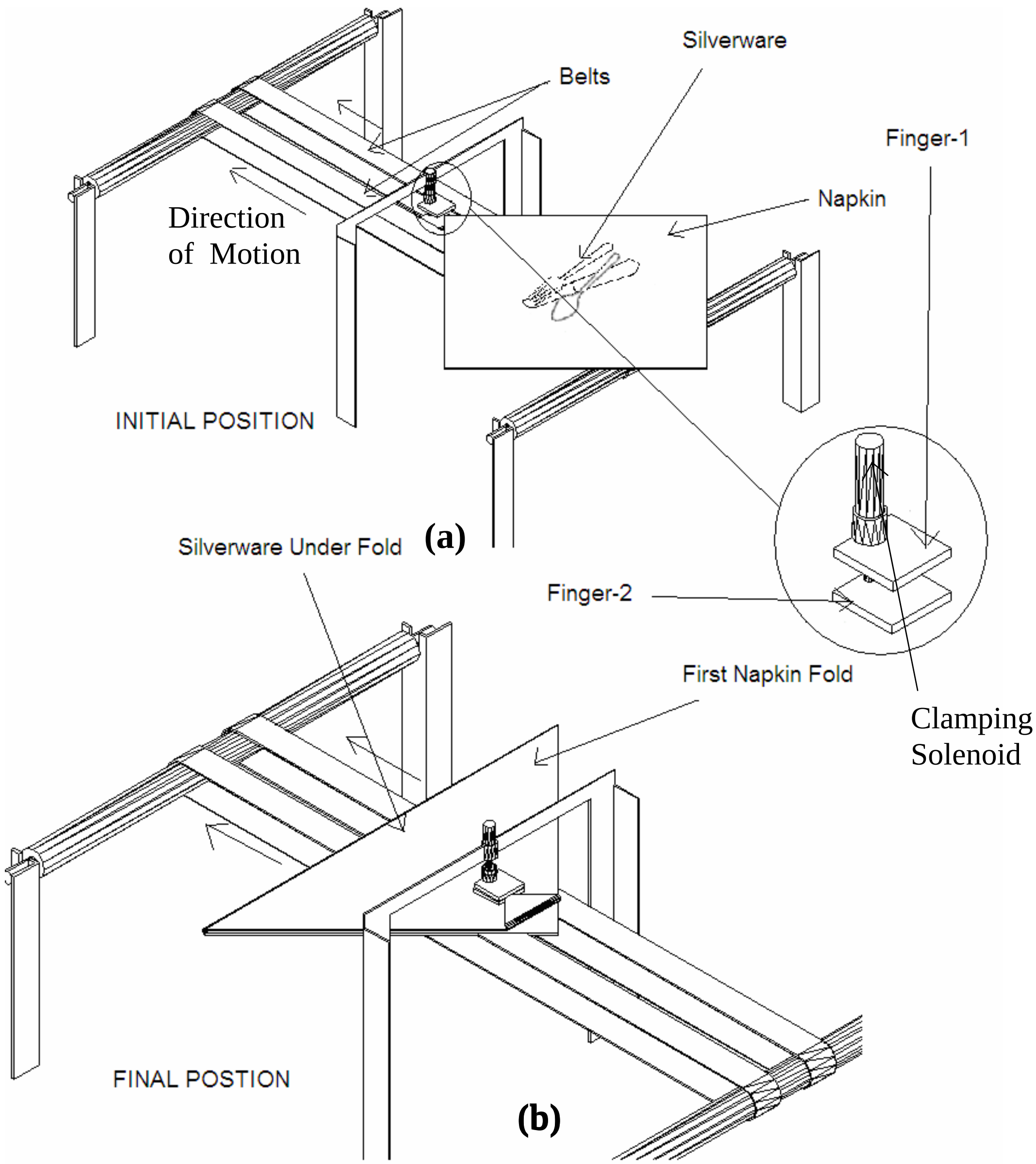


Fig. 2.10 Folding Mechanism-1: Method for Long First Fold.

As the leading napkin tip is conveyed over Finger-2 and under Finger-1, a solenoid actuates Finger-2 to clamp the napkin tip between the two fingers. This clamping raises the tip slightly above the rest of the moving napkin. The belt moves the unclamped portion of the napkin under the clamped tip, which causes the napkin to fold over the silverware along the diagonal, shown in Fig 2.10(b). Once the trailing napkin tip arrives under the clamped tip, the solenoid releases Finger-2, which releases the clamping action, completing the first fold.

The second and third folds handling the sides of the napkin are each implemented with similar mechanisms. A four-bar, crank-rocker mechanism, shown in Fig 2.11, carries a 10” long folding pad fixed to the rocker arm.

The four-bar, crank-rocker mechanism uses a motor rotating in a single direction, such that the motor does not need to reverse direction to return the folding pad to its ready position. Initial experimentation, using solenoids to actuate the folding pad were unsuccessful because of the inability of the solenoid to retrieve the folding pad back to its ready position. We also considered DC gearmotors can be reversed in order to achieve the to and fro motion needed by the folding pads. This requires the motor to move through an angle, stop, and then reverse it in order to achieve the to and fro motion. We elected not to employ such motors in order to avoid stopping the motor, and also to avoid the additional sensors that would be required to position the folding pad. The four bar mechanism, being a mechanical device, has additional moving parts, which can increase manufacturing cost. Also higher power is required to drive the motor when the flap is to be moved through large angles. However, we decided its advantages outweighed its disadvantages. An analysis of the four bar mechanism has been provided in Appendix D.

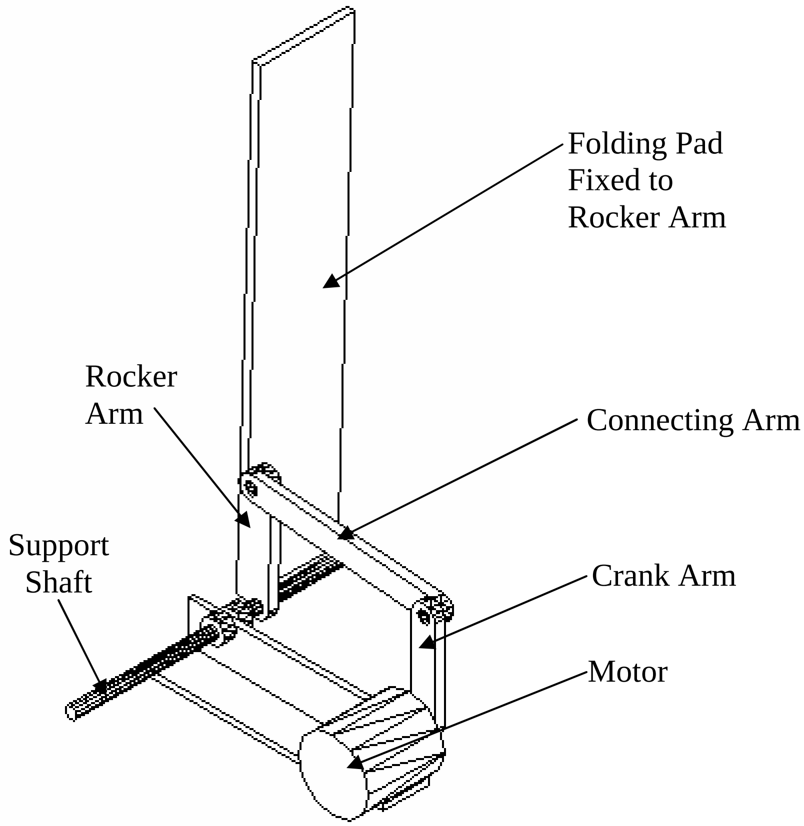


Fig. 2.11 Folding Mechanism-2: Crank and Rocker Mechanism for Side Folding.

The crank arm, shown in Fig 2.11 moves a complete revolution to move the rocker to and fro to fold the sides of the napkin. The advantage of using the crank and rocker mechanism is the use of a single drive that rotates only in one direction.

The side folds placed on the napkin are implemented as shown in Fig 2.12. The napkin with its first fold moves along the two-belt conveyor in a direction “into the page”. After the long first fold is placed using Folding Mechanism – 1, the napkin is guided on to the folding pad of Fig 2.11 using the support belts shown in Fig 2.12(a). The four-bar mechanism is actuated, causing the folding pad to move from its extreme open position to its extreme inward position. This causes the napkin end to fold over, as shown in Fig 2.12(b). The four bar mechanism then returns the folding pad to its extreme open position to await the next napkin. A similar mechanism, with the guide belts and a four bar mechanism, is placed on the opposite side of the conveyor to fold the other side of the napkin.

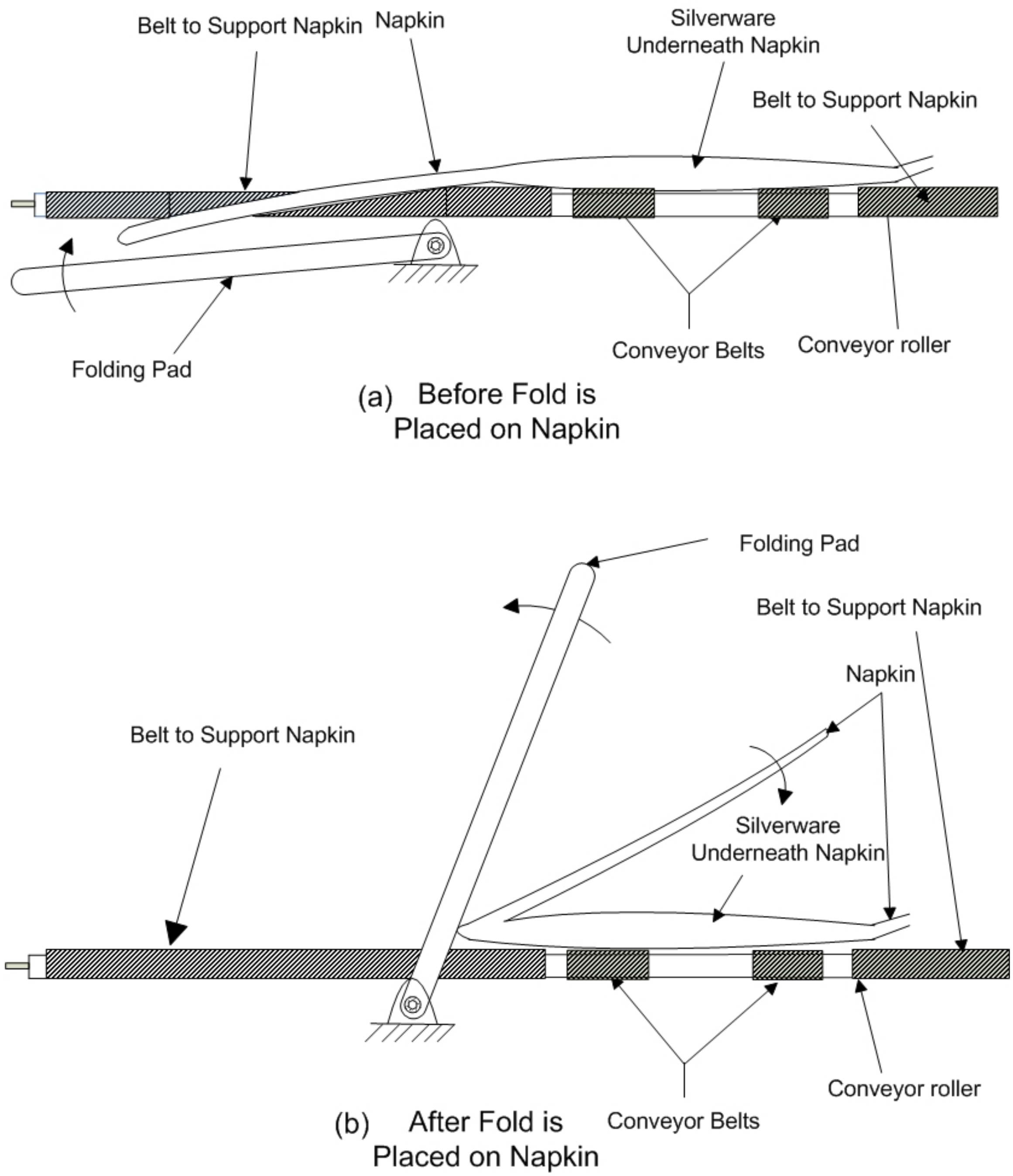


Fig 2.12 Working of Folding Mechanism-2

A final sequence is needed to roll the napkin, to complete the folding process. After both ends of the napkin have been folded, this is implemented using a four-bar mechanism similar to those used in the second and third “end” folds. However the folding pad used in this must be of different geometry than the flat pads used in the second and third folds, since the folding hinge of the napkin must be at a different location than the fixed hinge of the rocker arm. Two rolling pads, are required to place the final sequence of folds, or rolls, shown on a modified four bar crank rocker mechanism in Fig 2.13. The pads are mounted on a connecting shaft used to transfer the rocker motion to the rolling pads. The dimensions of the rolling pads (given in Appendix B) were determined by trial and error.

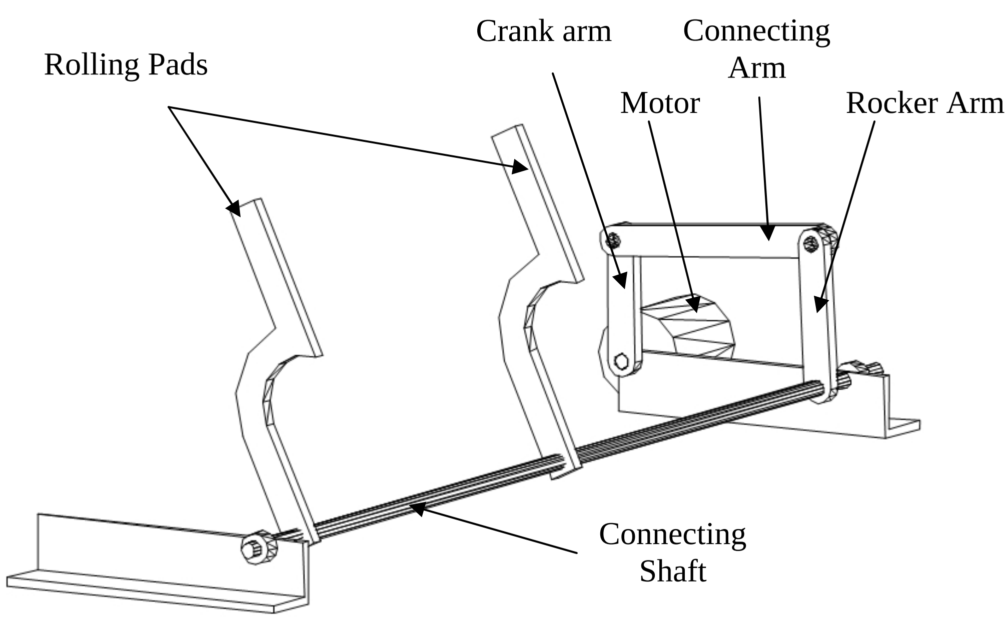


Fig 2.13 Folding Mechanism:3: Modified Crank Rocker Mechanism for Final Sequence of Folds.

A finger mechanism shown in Fig 2.14 is used to hold the napkin and the silverware in position for the first roll to take place. The mechanism consists of two motors. One is a DC gearmotor mounted on a sliding carriage supported by two sliding supports. The DC gearmotor has a spur gear, or pinion, mounted on its shaft meshing with a fixed rack, in order to facilitate the linear motion of the entire assembly. The actual finger is mounted on a servo motor as shown in the figure, which after being moved over the napkin, rotates the finger into place to hold the napkin and silverware in place for the first roll.

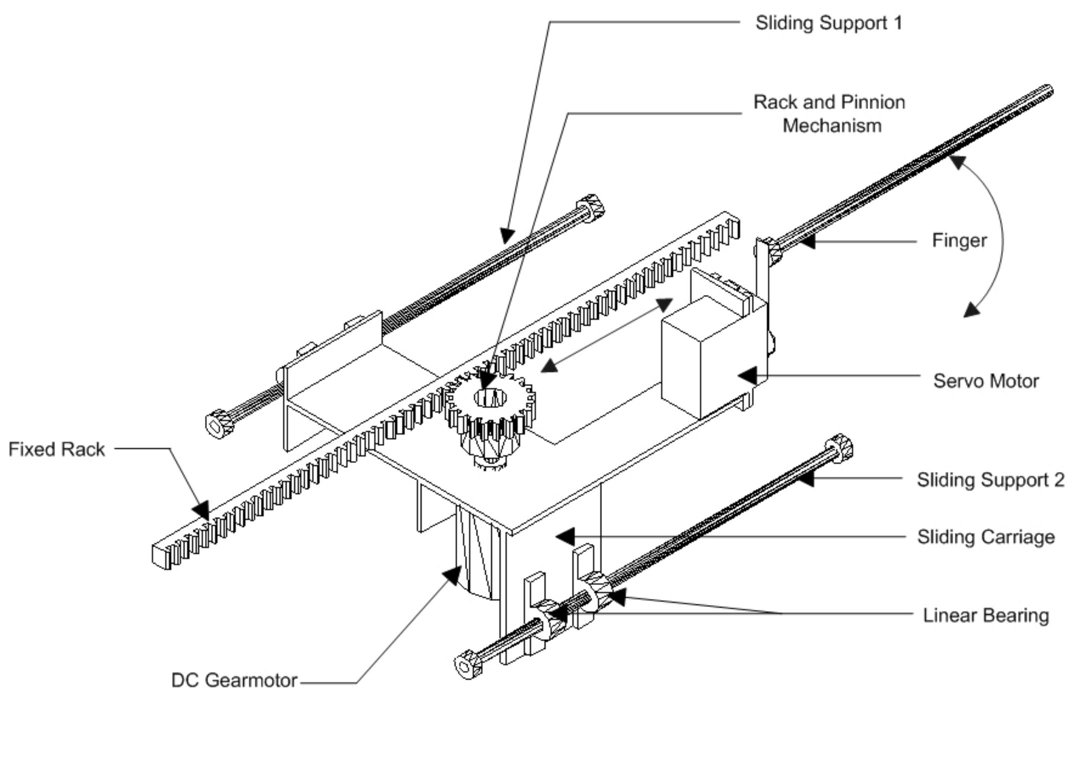


Fig 2.14 Finger Mechanism used in Folding Mechanism-3

The process of placing final rolls on the napkin is shown in Fig 2.15. The napkin is moved along by the conveyor belt, into position for the rolling pads to roll the napkin. The Finger in Figure 2.14 is brought into position as shown in Fig 2.15(a), and the first roll proceeds as shown in Figure 2.15(b). This finger is used to place the first fold only. Once the first fold is placed, the finger is retracted linearly by the rack-and-pinion mechanism in Figure 2.14 in a direction that is perpendicular to surface of the paper, and then rotated by the servomotor in Figure 2.14 into the ready position for the next napkin.

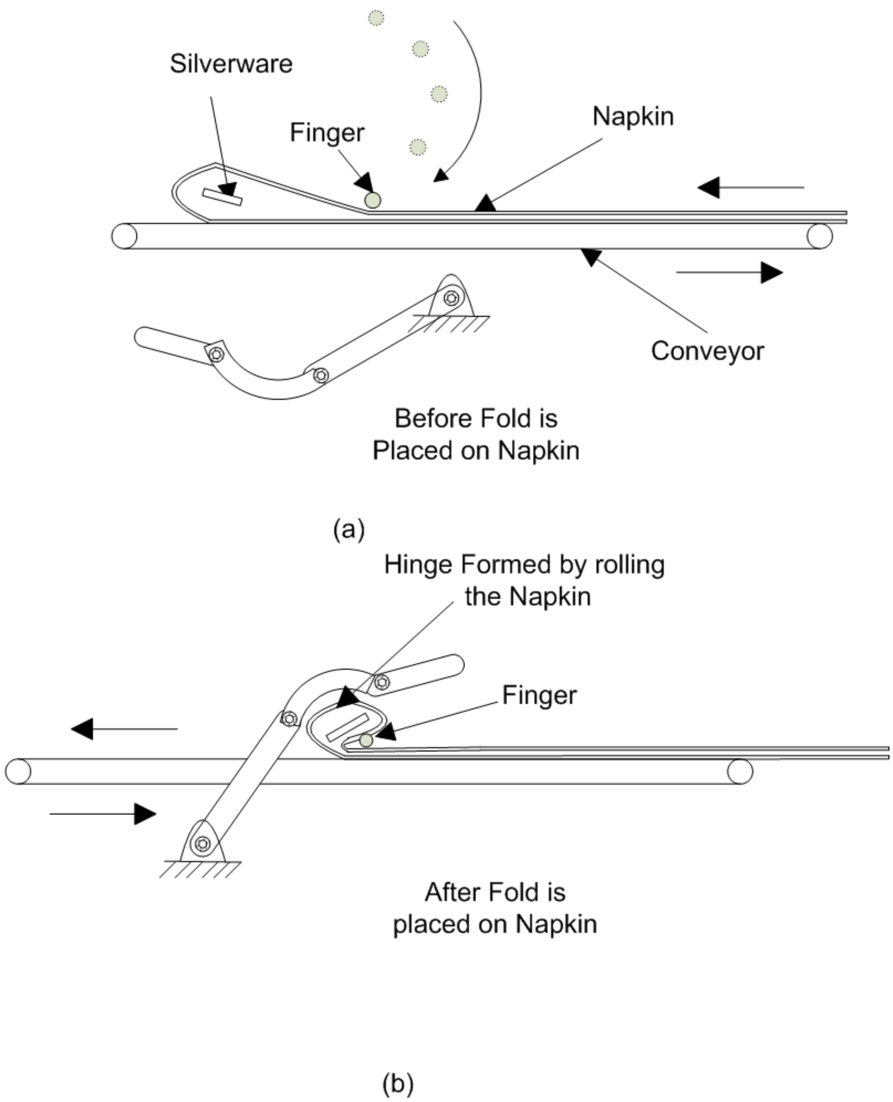
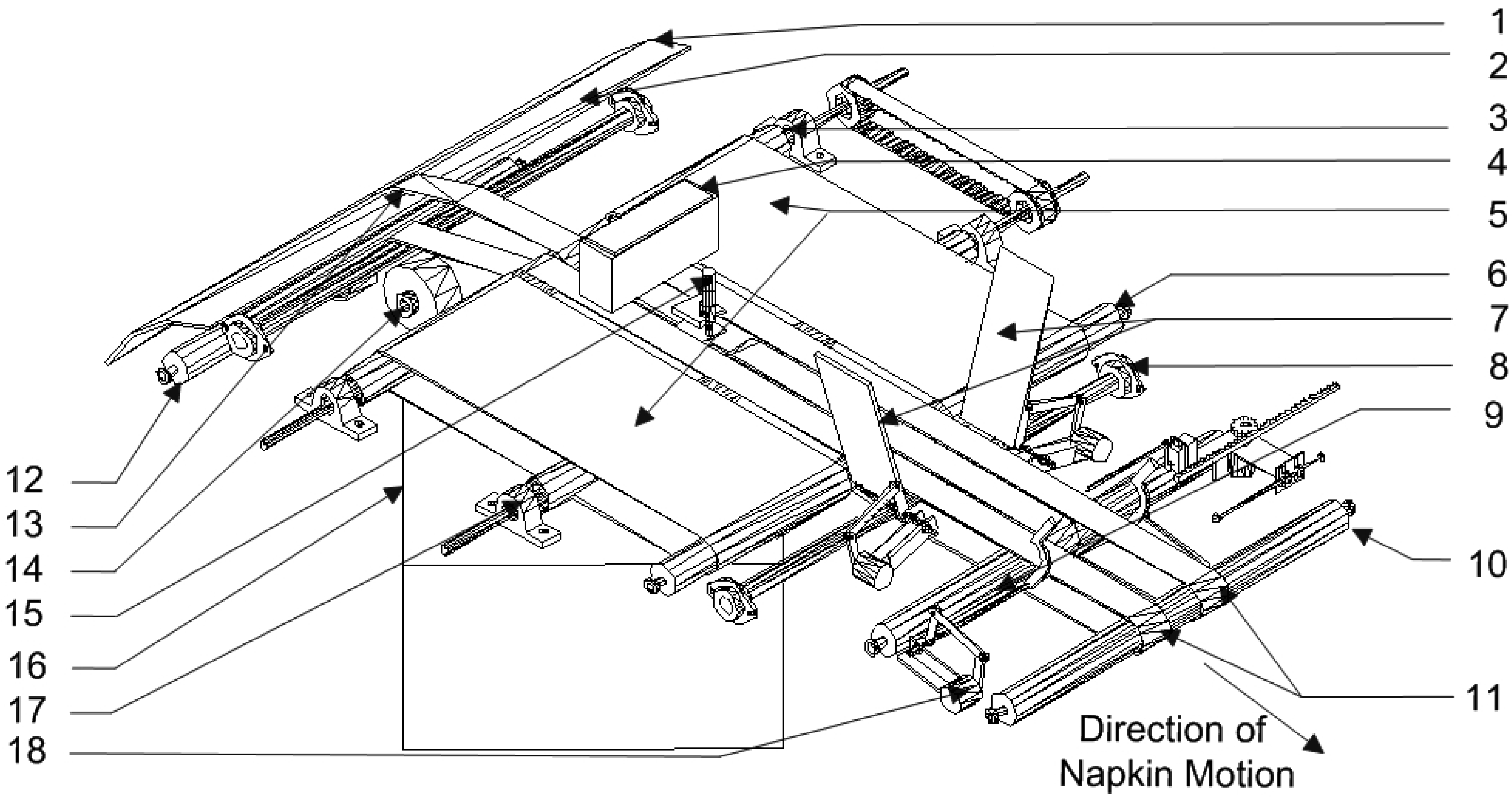


Fig 2.15 Final Sequence of Rolls Placed on the Napkin

For the rolling process to be complete, the hinge formed on the roll must be accommodated within the folding pad and the conveyor belt. The curved profile of the rolling pad provides space for the hinge of the rolled napkin. A series of rolls are placed on the napkin by moving the rolling pads forward, then back, as the conveyor continues to move the partially rolled napkin in the direction shown in Figure 2.15, until the complete length of the napkin is rolled, completing the process. The rolled napkin would then be conveyed out of the system, dropping into a collection bin.

2.5 Design of the Napkin Conveyor System

The napkin conveyor system is designed to accommodate the geometry and design of the napkin bin and the folding mechanisms, and shown in Fig. 2.16. Once the napkin is placed on the conveyor and the axis of the napkin moves beneath the silverware drop mechanism, a set of silverware is dropped on the unfolded napkin along the napkin diagonal perpendicular to the direction of motion. To accommodate the first folding mechanism, two parallel 2” wide belts (11) with a clearance of 2.5” between them, is fixed between Drive Roller-1 (3) and End Roller-3 (10) as shown in Fig 2.16. Another belt (13), 2” wide, used to transport the napkin from the Napkin Bin (16), is fixed between Drive Roller - 1(3) and End Roller-4 (12), and traverses over the Static Cylindrical Profile (1). Additional belts are placed between Drive Roller – 1(3) and End Roller – 1(6) to support and convey the lateral sides of the napkin before encountering folding mechanisms – 2(7). A belt is also placed between Drive Roller – 2(17) and End Roller – 2(8) to provide support for the napkin with its first fold containing the silverware.



1-Partial Cylindrical Profile
2-Idler Shaft-1
3-Drive Roller
4-Silverware Drop Mechanism
5-Side Belt to Support Napkin
6-End Roller-1
7-Folding Mechanism-2
8-End Roller-2
9- Folding Mechanism-3
10-End Roller-3
11-Main Belt System
12-End Roller-4
13-Belt over the Profile
14-Napkin Singulation Roller
15-Folding Mechanism-1
16-Napkin Bin
17-Drive Roller-2
18-Folding Mechanism-3

Fig 2.16 Conveyor System

The Silverware Drop Mechanism (4) is placed immediately following the Drive Roller – 1(3). Folding Mechanism-1 (15) is placed between the two main conveyor belts to make the first fold. Folding Mechanisms-2 (7) are placed adjacent to the belts downstream of the support belts (5) to perform the side folds. Two Idler rollers (2 & 9) are used to adjust belt tension and provide stability to the conveyor system. Idler Shaft-1 (2) ensures that the conveyor belt does not interfere with the Napkin Singulating Roller (14) in the napkin bin.

A Napkin Bin to store dinner napkins and dispense one napkin at a time has been described, together with three different folding mechanisms to completely fold the napkin containing silverware. A conveyor system connecting the napkin bin, silverware drop mechanism, and the folding mechanisms has also been described. A complete set of drawings with dimensions is included in Appendix B. A control system to control the various motion components required to implement the device is discussed in the next chapter.

CHAPTER 3

DESIGN OF CONTROL SYSTEM

3.1 Need for Embedded System

The silverware identification, inspection and collection process is composed of the various stages shown in Fig 3.1. The singulation device provides isolated pieces of silverware to the identification and inspection system. Based on information from the identification and inspection system, the sorting and the orientation system places like silverware with like orientation in a single collection bin. This process information is required for the process to wrap silverware in a napkin. The identification and inspection system implemented by Lolla (2005) uses a line scan camera connected to a personal computer equipped with an image acquisition board. Identification, inspection, sorting, and orienting, along with napkin wrapping, can be done using a personal computer for the needed logic and computations. However, such a PC would need to be exceptionally fast to achieve high throughput (greater than 30 silverware pieces per minute) through the system. For the silverware wrapping device to work in a stand-alone mode, (i.e., without the middle two processes in Fig 3.1), using a personal computer equipped with DAQ boards would result in high costs and underutilization of computing power. An effective way to control the silverware sorting and orientation device along, with the silverware wrapping device, is to use programmable microcontrollers. Microcontrollers offer dedicated computing power along with a variety of options that can be used to control the systems effectively. Programmable microcontrollers used to control the silverware sorting and orientation system communicate with the personal computer to operate on the information sent from the inspection and identification system. It is hence effective to use microcontrollers to control the silverware wrapping process. This offers a compact programmable system that can communicate with the silverware sorting and orientation system, and also operate in stand-alone mode for the wrapping process.

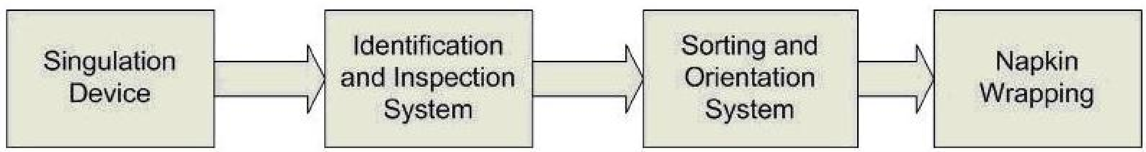


Fig 3.1 Stages in Silverware Automation Process.

The control system used to control the silverware wrapping device consists of a logic unit, a power unit, actuators and feedback systems, as shown in Fig 3.2. The logic unit communicates with the silverware sorting and orientation device to obtain information on availability and type of silverware. Based on this information and information from the feedback systems, the logic unit commences and finishes the wrapping process by actuating the various mechanisms through the power unit

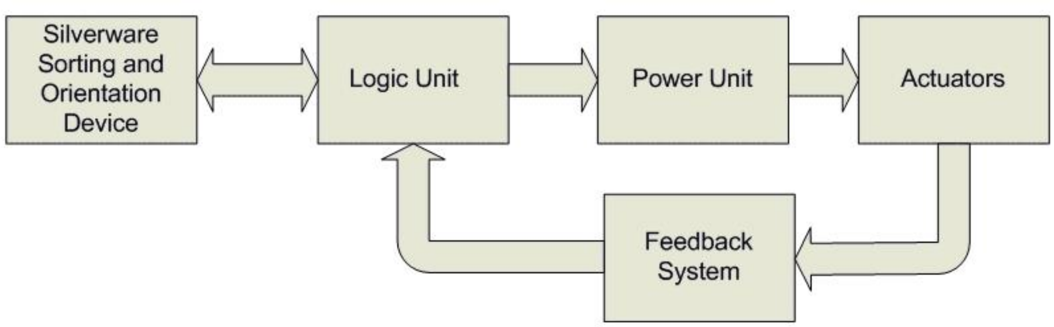


Fig 3.2 Control System used to Control the Silverware Wrapping Device.

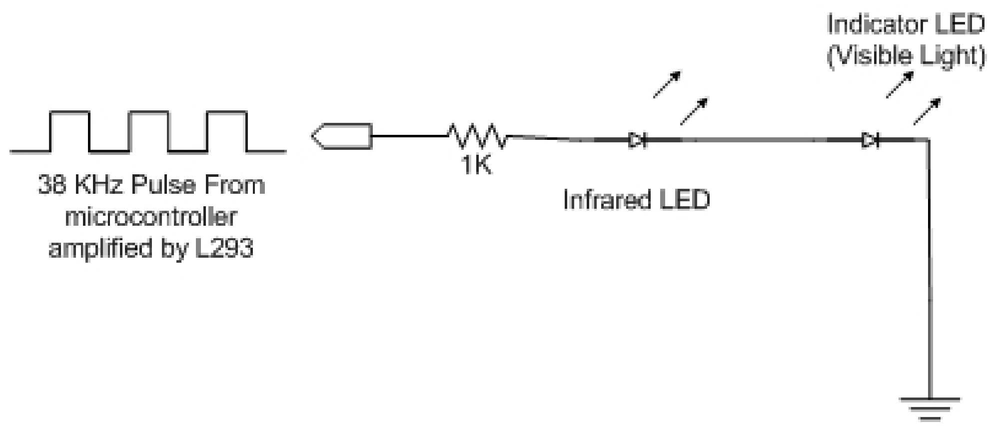
3.2 Compatibility with Existing Control System

For the silverware wrapping device to operate with the rest of the system, it is essential for the control system to be compatible with the existing sorting and orientation device. The control system used to sort and orient silverware pieces (Peddi, 2005), receives information from the serial port in the PC, which is used to identify and inspect silverware (Lolla, 2005). This information is used to sort and orient silverware pieces (Peddi, 2005), and the information required by the silverware wrapping device is forwarded to it through a RS-232 communication channel. The information available on an RS-232 communication channel must be retrieved by the control system used to control the silverware wrapping device. The control system is designed to communicate with the silverware sorting and orienting device using RS-232 communication protocol. The control system will be able to report any problems with the silverware wrapping mechanism back to the sorting and orientation system using the same RS-232, allowing it to report errors to the operator.

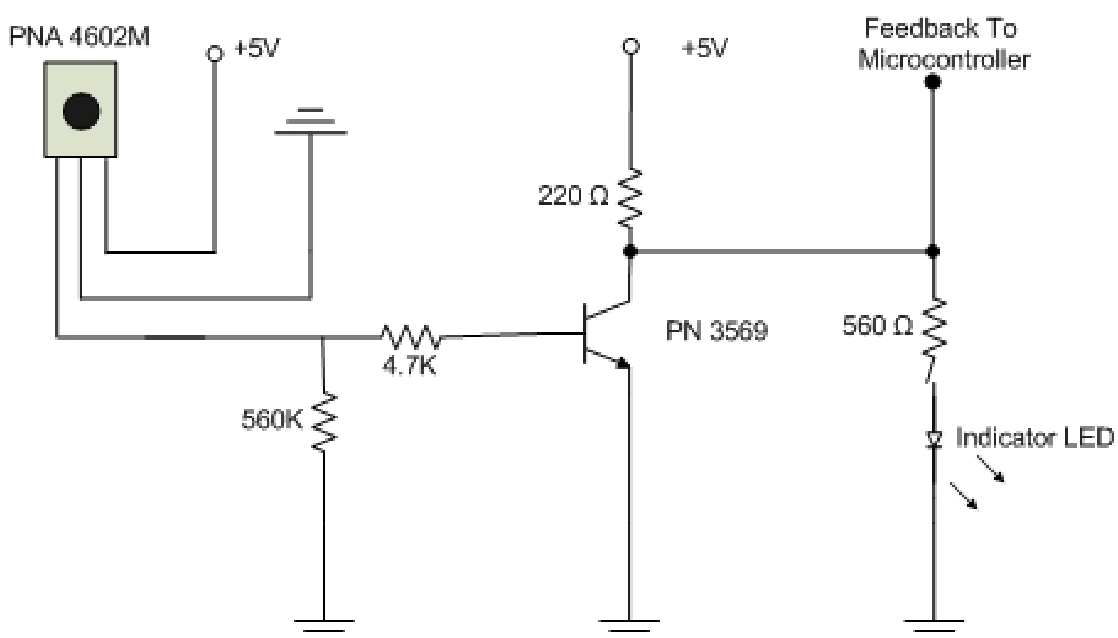
3.3 Implementation of Feedback System

A robust feedback system is essential for the control system to identify the various tasks to be executed at the proper moment. From the moment when the napkin is pulled from the bin to when it is completely wrapped, its behavior is not completely predictable. In order to wrap the silverware, it is essential to determine the position of the napkin at key points. Infrared sensors provide a reliable non-contact method of identifying the presence of an object as it moves between an infrared trans-receiver pair obstructing the path of the infrared rays. A total of 13 infrared sensors are positioned at various points in the system to detect presence of the napkin or moving components of the system. Since a large number of sensors are used in the process, it is desirable to select reliable, yet low-cost sensors. The infrared trans-receiver pair and associated circuitry selected for these purposes is described in what follows.

To avoid interference from infrared rays present in the environment, the receiver selected is sensitive only to infrared energy switched at 38 KHz. Panasonic MODEL PNA4602M is the receiver selected to detect the emitted infrared rays. The transmitter unit consists of an infrared LED emitting signal at 38 KHz. In order to power all the 11 infrared LED's, voltage to them is pulsed by generation from a separate microcontroller dedicated to the purpose, and fed to a half H-bridge circuit in the quad half H-bridge module MODEL L293 from Texas Instruments. The power of the switching pulse to the IR led is modulated using a 1KΩ resistor placed in series with the infrared transmitter, shown in Fig 3.3(a). The output from the IR receiver unit PNA 4602M can be fed directly to the microcontroller feedback. If the receiver is connected to the microcontroller directly, there will be no visible indication of the receiver unit detecting the presence of IR energy. This feature is useful in the process of debugging the system. To provide the receiver unit with a visible indicator, the signal from the IR receiver unit is fed to the amplifying transistor PN 3569 as shown in Fig 3.3(b). The output from the transistor is used to power a visible LED, and to give feedback to the microcontroller.



(a) Infrared Transmission Unit



(b) Infrared Receiver Unit

Fig 3.3 Infrared Sensor System.

The positions of the various sensors used in the silverware wrapping process are shown in Figure 3.4.1 – 3.4.3.

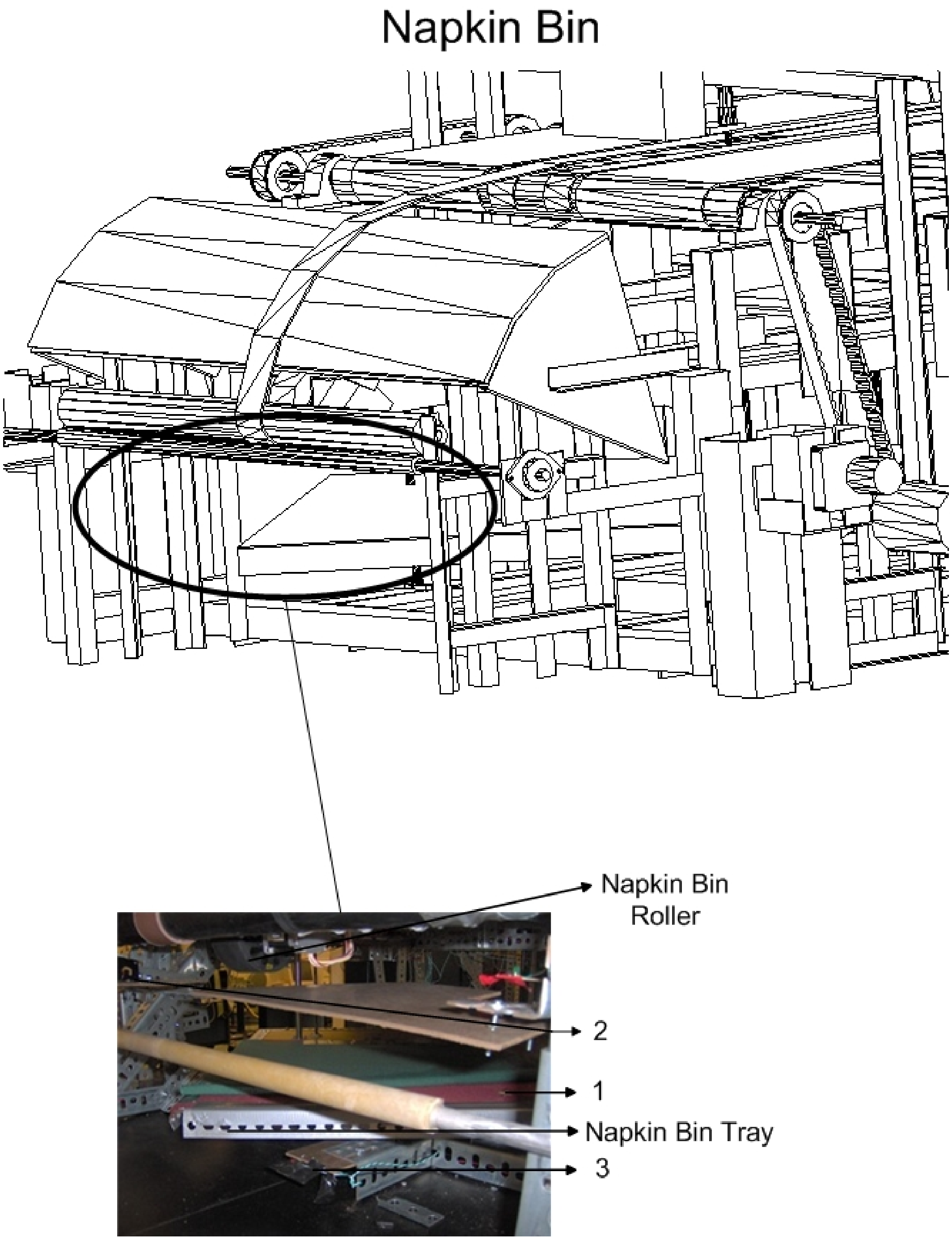


Fig 3.4.1 Position of Feedback Components in the Silverware Wrapping Device-1

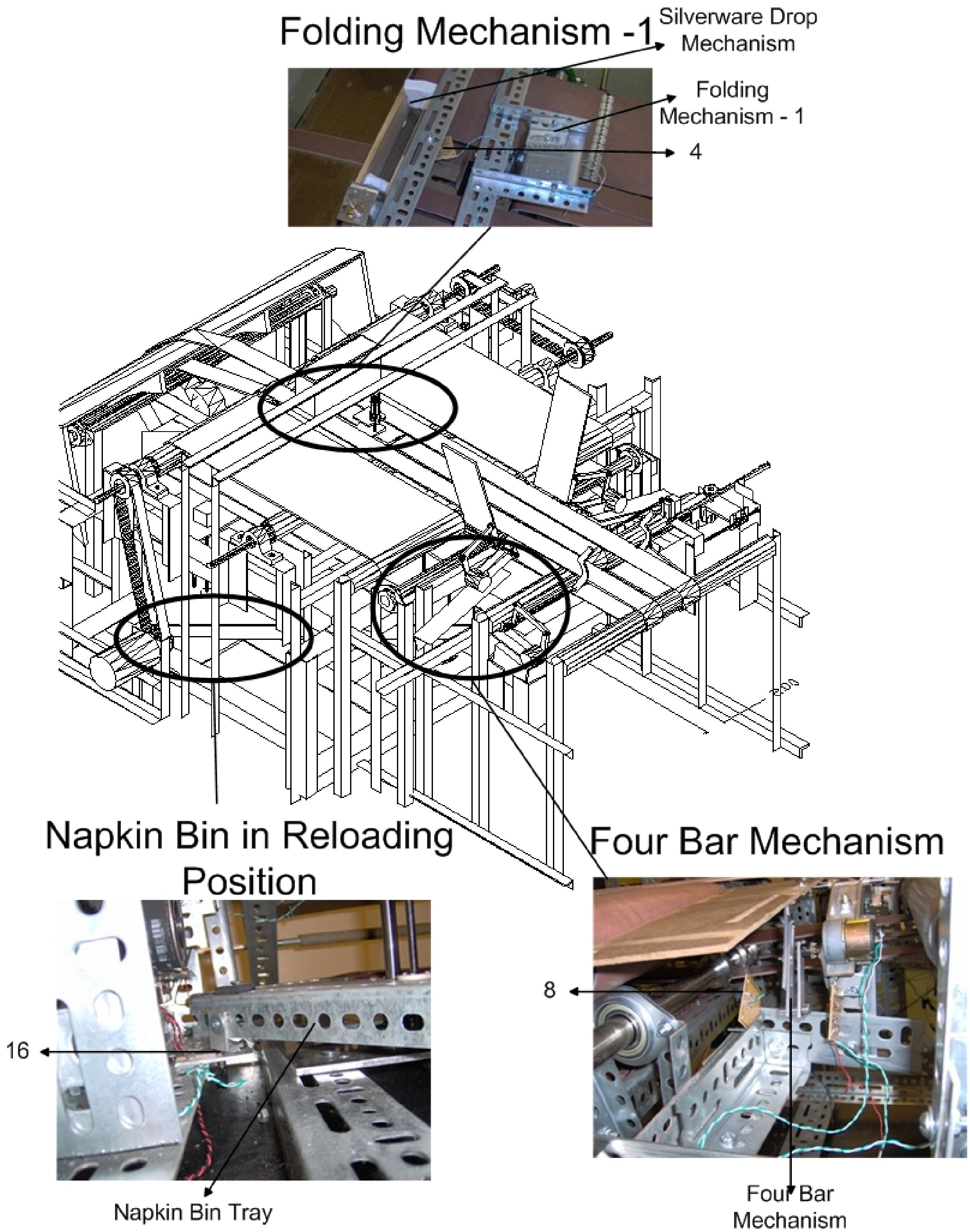


Fig 3.4.2 Position of Feedback Components in the Silverware Wrapping Device-2

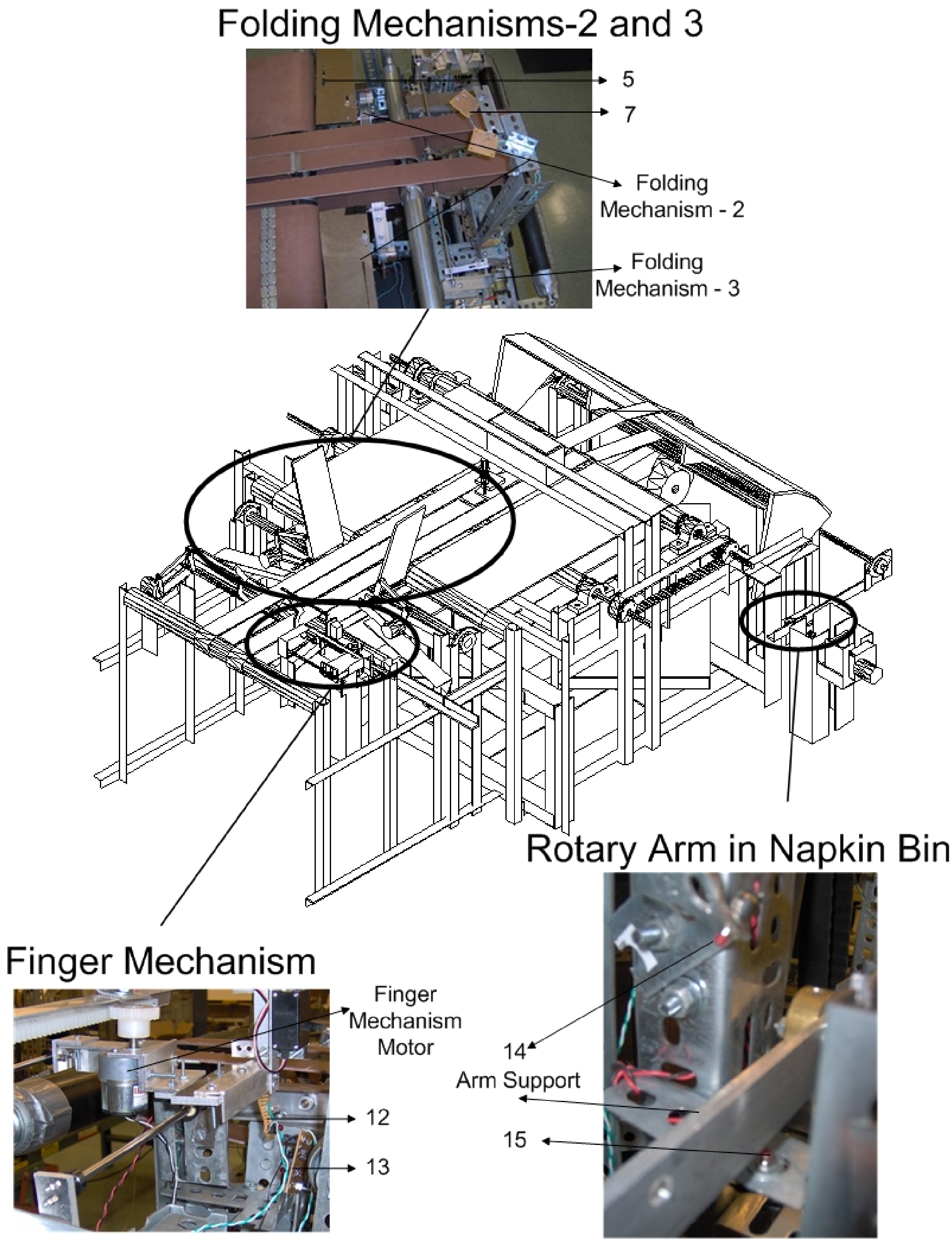


Fig 3.4.3 Position of Feedback Components in the Silverware Wrapping Device-3

Three trip switches (14, 15 & 16)(Figures 3.4.1 – 3.4.3) are also used in the feedback system, the first and second are used to identify the position of the arm mechanism used by the napkin pickup mechanism, the third is used to find out if the napkin tray has reached the bottom position for reloading napkins.

The various infrared sensors shown in Figures 3.4.1 – 3.4.3 are used to:

1) Detect if napkin bin is empty (1).
2) Detect if napkin is within the vicinity of the bin starter roller (2).
3) Detect if napkin has started to exit the napkin bin (3).
4) Detect if napkin is beneath the silverware drop mechanism (4).
5) Detect if napkin is beneath Folding Mechanism – 1 (4).
6) Detect if napkin is within range of Folding Mechanism – 2 (5).
7) Detect rotation of motor used to drive Folding Mechanism – 2 (6).
8) Detect if napkin is within range of Folding Mechanism – 3 (7).
9) Detect rotation of motor used to drive Folding Mechanism – 3 (8).
10) Detect if napkin is over Folding Mechanism – 4 (9).
11) Detect rotation of motor used to drive Folding Mechanism – 4 (10).
12) Detect position of moveable carriage in Finger Mechanism (12 & 13).

3.4 Design of the Control System

The silverware wrapping system, illustrated in Figs 2.7-2.15, uses ten independent drives. Controlling these drives requires significant computing power and a large number of I/O lines. To address this problem, the workload is shared between various microcontrollers that take care of localized operations. For the system to function effectively, the microcontrollers must communicate with one another. The various control mechanisms used to wrap silverware are:

1.Communication and silverware drop control.

2.Napkin Bin control.

3.Wrapping mechanism control.

4.Bin motors synchronous feedback control.

In order to implement these control actions, five microcontrollers are used. The system is designed to accommodate additional drives and feedback signals to provide room for future upgrades and to integrate the system completely with the rest of existing systems in the silverware handling process.

The functions of the five different microcontrollers for general communication and control are shown in Fig 3.5, and are described in what follows. Detailed circuit diagrams are given in the Appendix C.

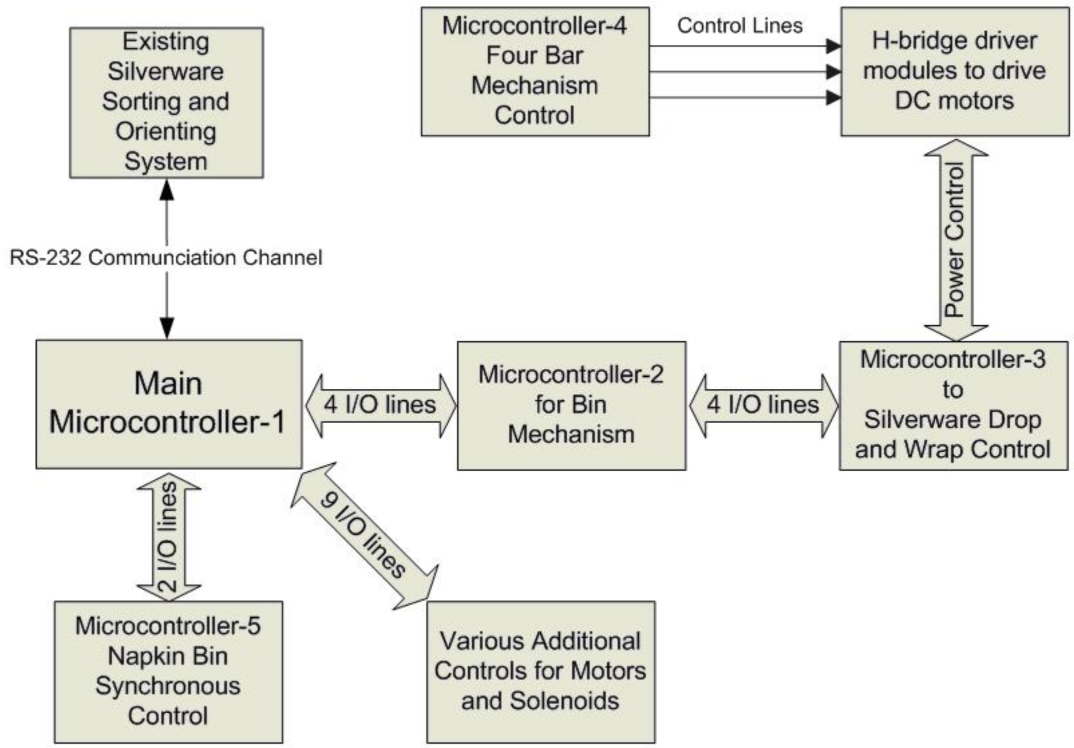


Fig 3.5 General Communication and Control Flow between the Five Microcontrollers

The Main Microcontroller-1, illustrated in Fig 3.6, is responsible for communication with the silverware sorting and orientation system using RS-232 communication standards, control mechanisms that will be used to convey silverware from the sorting and orientation device to the silverware drop mechanism, and coordinating with microcontroller-2.

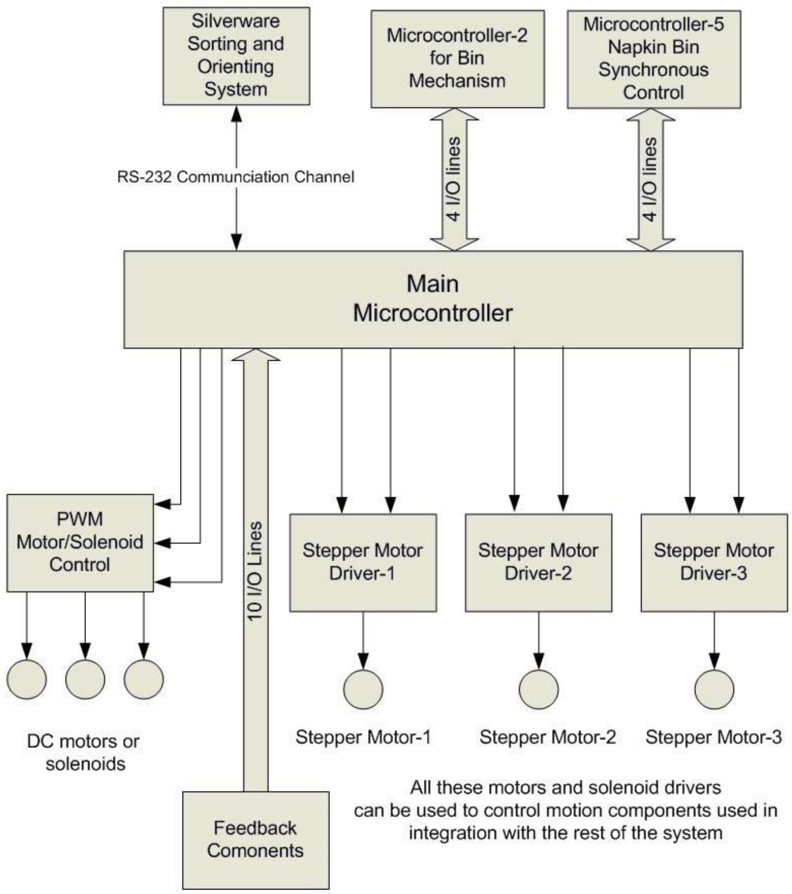


Fig 3.6 Microcontroller-1: Main Microcontroller functional diagram

The Main Microcontroller-1 communicates with Microcontroller-2, as shown in Fig 3.7, to control the napkin bin mechanism using 4 I/O lines. The main microcontroller uses 6 I/O lines to control three stepper motors that might be required to integrate the silverware wrapping process with the rest of the process. Three PWM motor/solenoid controls are controlled using 3 I/O lines that can be used to control DC motors or solenoids depending on requirements. These stepper motor drivers and the PWM motor/solenoid drivers can be used to control motion components that will be required to integrate the silverware wrapping device with the rest of the automation process. It is also equipped with 10 I/O lines to obtain feedback from sensors in the silverware wrapping device, namely photo detectors and trip switches.

The Bin Mechanism Microcontroller-2, whose functional block diagram is shown in Fig 3.7, receives information from the Main Microcontroller to dispense napkins as required. This microcontroller controls three motion components. The first controls the napkin bin tray mechanism to assure contact between Roller-1, Fig 2.7, and the top napkin in the napkin stack. This employs three stepper motors with leadscrews, Fig 2.8, to drive the bin tray. These three motors are controlled synchronously to achieve the desired motion and maintain the bin tray level as it moves up and down. To obtain synchronous motion, 2 I/O lines are used to control a single stepper motor driver, to drive three stepper motors connected in parallel. The second motion component controls rotation of Roller-1, Fig 2.8, placed above the napkin stack. A stepper motor drives Roller-1 and is controlled through a single control line of the 4 I/O lines connected to Microcontroller-3, shown in Fig 3.7, that also controls the silverware drop and wrapping mechanisms. Since the roller is rotated clockwise direction only, the stepper motor is driven if the control line is “high” and stopped when the control line is “low”. The third motion component is used to drive the rotary arm, Fig 2.7, on which the roller is mounted. Two DC gearmotors, one at each end of the rotary arm, Fig 2.7, are driven synchronously to achieve this motion. These gearmotors are driven using an H-bridge driver requiring 2 I/O lines to control the motor. A third line from the driver unit gives feedback to Microcontroller-2 if there is any error in the driver module. Also, Microcontroller-2 uses 5 I/O lines to receive feedback required to obtain the position of the napkin to control the bin motion components. Microcontroller-2 communicates with Microcontroller-3 to send and receive error messages using 4 I/O lines.

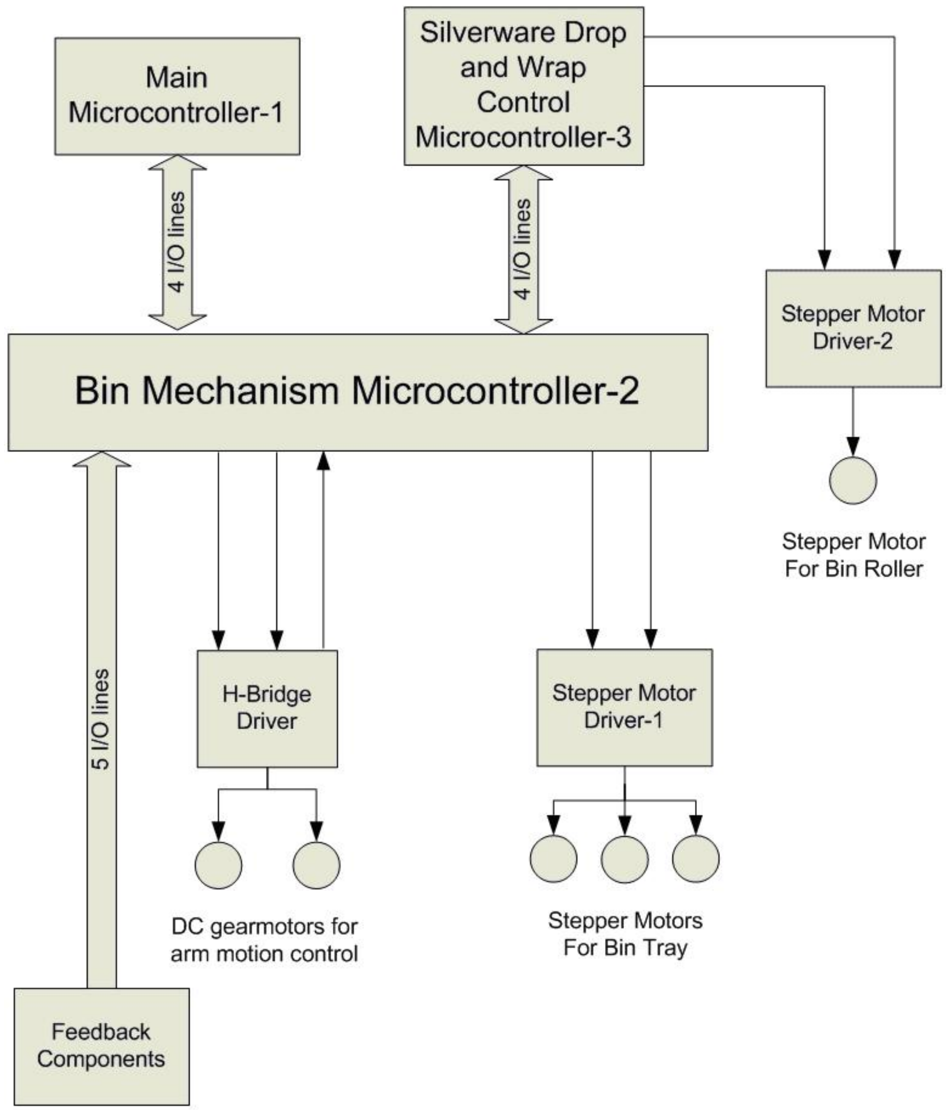


Fig 3.7 Microcontroller-2: Bin Mechanism Microcontroller Functional Diagram

Microcontroller-3, shown in Fig 3.8, is used to control the conveyor system, the silverware drop mechanism, and Folding Mechanism-1, Fig 2.10. The conveyor is powered using a high-torque stepper motor. This is driven using a stepper motor driver requiring 2 I/O lines. The silverware drop mechanism is driven by a solenoid operating the hinged bottom of the silverware drop box. Folding Mechanism-1, Fig 2.10, is driven by two solenoids. To drive these solenoids high side switches are used, in which the high side switch switches the positive terminal required to actuate the solenoids. Two high side switches, controlled using 2 I/O lines are used, one for each solenoid. Two power lines used to power Folding Mechanism-2 and Folding Mechanism-3, Fig 2.13, are switched using two high side switches controlled by 2 I/O lines. The microcontroller uses 6 I/O lines to monitor feedback signals from infrared sensors to operate the various mechanisms controlled by the microcontroller.

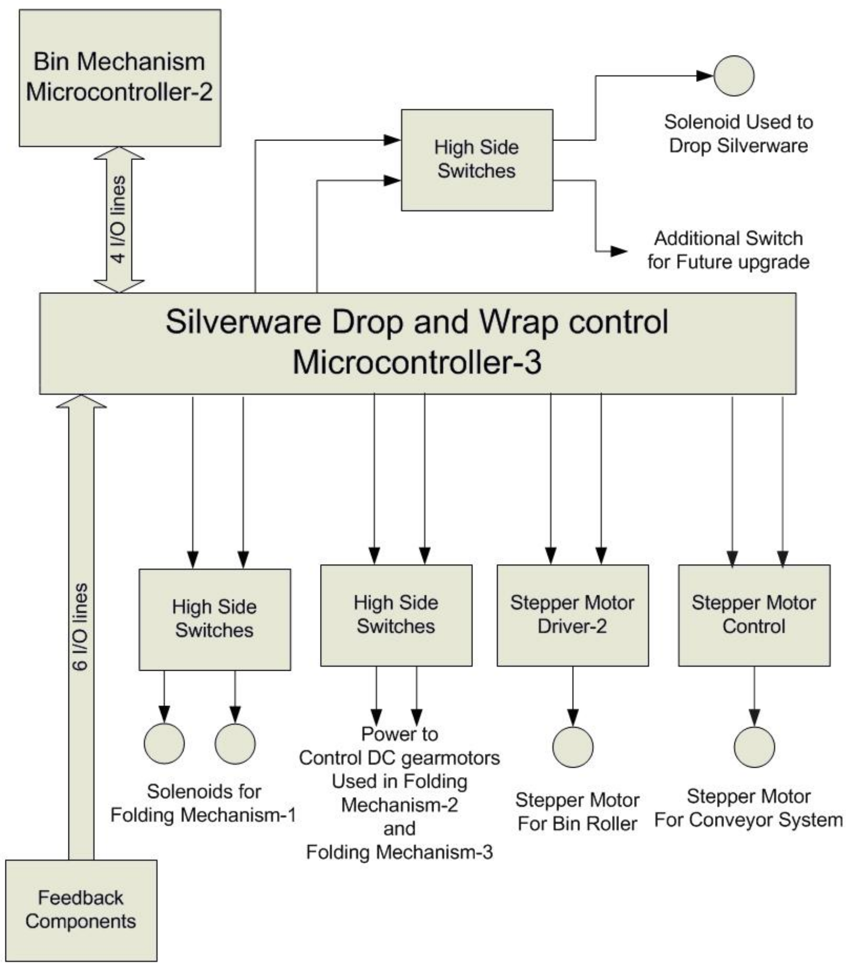


Fig 3.8 Microcontroller-3: Silverware Drop and Wrap Control Microcontroller Functional Diagram

Microcontroller-4 shown in Fig 3.9 is used to control Folding Mechanisms-2 (two of these) and Folding Mechanism-3 consisting of two DC gearmotors and a servomotor. The two folding mechanisms in Folding Mechanism-2 (Fig 2.11) each require a DC gearmotor to drive the four bar mechanism, used to fold the sides of the napkin. These are controlled using 2 I/O lines. Folding Mechanism-3 (Fig 2.13) requires a DC gearmotor to drive the four bar mechanism powering the folding pads that finish the rolling process. Additionally a servomotor is used to move the finger in position for the first fold, with a DC gearmotor powering a rack and pinion mechanism to first move the finger carriage over the napkin, and then retrieve the finger after the first roll is placed. The microcontroller uses 8 I/O lines to obtain feedback from optical sensors to obtain the position of the crank arms in the four bar mechanisms used to drive the folding mechanisms. The three DC gearmotors are driven using H-bridge circuitry. The H-bridge circuitry provides braking for the DC gearmotors, crucial for the operation of the four bar mechanisms.

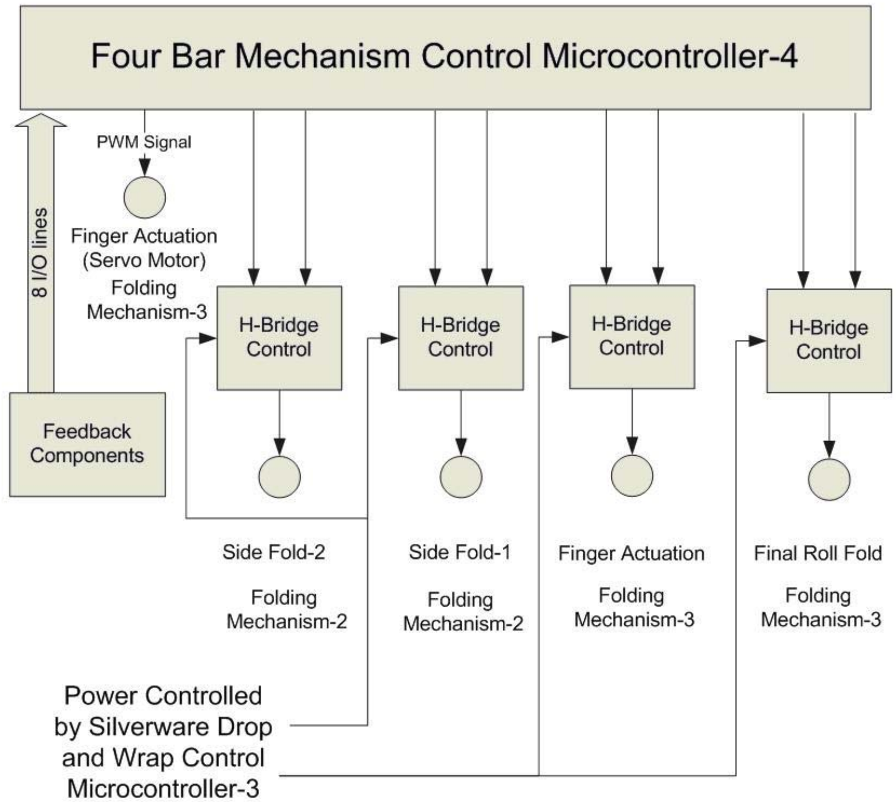


Fig 3.9 Microcontroller-4: To Control Four Bar Mechanisms Used in Folding Mechanisms-2 and Folding Mechanism-3

Microcontroller-5 shown in Fig 3.10 assures synchronous operation of the three bin stepper motors, Fig 2.8, used to move the bin tray containing the napkins. Each of the three motors is fitted with incremental optical encoders, which are connected to 8 bit counters that provide information on the position of the motor shaft. These counters are read using the microcontroller. The four bits on the high side of the byte value are read using the microcontroller, and those for each motor are compared with one another. Differences between these values for any two motors indicate that they have not moved equal amounts, which would cause the napkin tray to tilt and bend. This information is sent back to the main microcontroller to halt the system and report the problem to the operator.

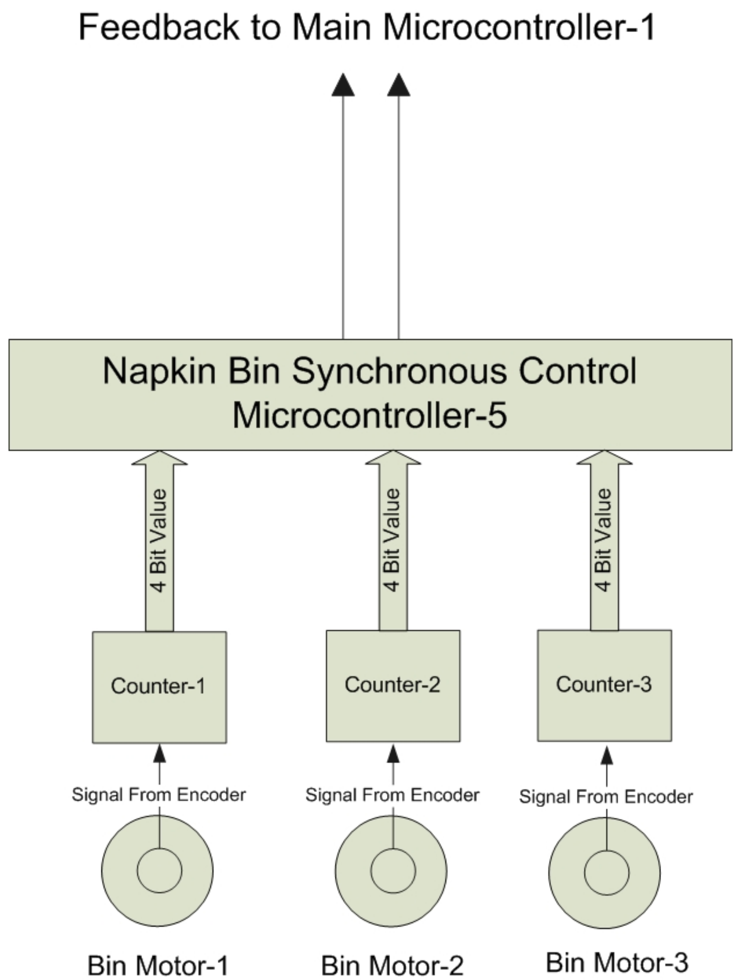


Fig 3.10 Microcontroller 5: Napkin Bin Synchronous Control Microcontroller.

3.5 Power Components Design

The various power drives described in the discussion above can be categorized as follows:

1) Stepper Motor Driver.
2) High Side Switches to control DC motors or solenoids.
3) H-bridge driver for bin arm control.
4) PWM DC Motor Driver.
5) H-bridge driver for Folding Mechanisms.

Detailed specifications for each of these are given in the Appendix. Six stepper motor drivers are used in the complete setup. All are bipolar stepper motors, which provide higher torque than unipolar stepper motors. The selected driver is model Kit-158 from kitsrus.com, a circuit adapted from Oatley Electronics, Australia, which uses two digital control lines to control a stepper motor. Every pulse on one of the control lines is used to advance the motor in clockwise or anti-clockwise direction, based on the signal on the other line. A “high” on the direction control line advances the motor in the clockwise direction, and a “low” advances the motor in the anticlockwise direction. These drivers are rated for the drive motor coils, rated at 5-30VDC, capable of providing currents as high as 6A.

High side drivers used to switch solenoids or DC motors are electronic switches that switch on or off the positive side of the power based on an input signal. The advantages of using an electronic driver are:

1) Low input power.
2) High switching speed.
3) High output power.
4) No moving parts.

The high side driver selected is 511NR from International Rectifier. The advantages of this driver are that it requires minimum external circuit and provides significant circuit protection. The disadvantage is that significant heat is produced at high currents. The driver is capable of providing 1.7A continuous current and a peak current of 5A.

The H-bridge module selected to drive the DC gearmotor for the rotary arm in the bin mechanism (Fig 2.7) is a DMOS based H-bridge driver model NMIH-0050 from newmicros.com. The advantage of using an H-bridge is that it is capable of reversing the motor based on the logic inputs. This driver is controlled using 2 I/O pins to drive the DC motor. The functional Truth table for the H-bridge driver is given in Table 3.1.

IN1	IN2	OUT1	OUT2	Function
L	L	L	L	Both Terminals of the Motor are grounded(Braking)
L	H	L	H	Motor turns counterclockwise
H	L	H	L	Motor turns clockwise
H	H	H	H	Both Terminals of the Motor are made high(Braking)

Table 3.1: Functional Truth table of H-bridge driver NMIH-0050

(L=Low; H=High)

IN1 and IN2 are the input pins that are controlled using the microcontroller I/O pins. OUT1 and OUT2 are the power outputs connected to the terminals of the gearmotor. The driver is rated at 5A continuous current with a peak of 6A and an operating voltage from 5VDC to 40VDC. The motors used are rated at 24VDC, 1.08A. Since two motors are used, (one at either end of the arm) the total current required to drive these motors is 2.16A. Both the motors are connected in parallel with alternate terminals connected to one another in order to provide the correct direction of rotation for the arm. Since the motors are DC gearmotors identical in construction, both rotate at the same speed, ensuring that they operate synchronously.

Three PWM DC motor/solenoid drivers are controlled using the Main Microcontroller-1. These are for future use in controlling the drives for conveying silverware from the silverware sorting and orienting device (Peddi, 2005) to the silverware drop mechanism (Figs 2.9; 2.15). The drivers selected are model DRV-101T from Texas Instruments. This driver is switched on or off using a single I/O pin. The PWM is controlled using a 100KΩ variable resistor included in the circuitry to build the circuit. The PWM gives control over the speed and torque of the motor. The driver is primarily intended for use with solenoids, but it can be also be used with DC motors. These drivers are rated at 1.9A max, with an operating voltage of 9VDC to 60VDC.

The H-bridge driver selected to control the four-bar crank-rocker mechanisms (Folding Mechanisms 2-3, Figs 2.11; 2.13) is model L298N from STMicroelectronics, which is a dual H-bridge driver. Two of these drivers are used to control the three DC gearmotors that drive two Folding Mechanisms-2 and one Folding Mechanism-3. H-bridge control is preferred over high side switches because of the braking capability of the H-bridge, which is vital for stopping the DC gearmotors after a single revolution. Two I/O pins are required to operate each of the two H-bridge drivers present in the L298N module. The drivers are rated at a total current of 4A. The module has an operating voltage range of upto 46VDC. The motors driven by these drivers are each rated at 12VDC, 300mA.

Power for the motion components is provided by two SMPS power supply units, normally used to power computer systems, together with a custom built SMPS system from TODD power systems (Model SC24-10709), which gives an output of 24V at a maximum current of 8A. The use of computer SMPS power supplies ensures that the power supplied is clean and that the unit is compact. This also provides a variety of output voltages, principally 3.3V, 5V and 12V. The voltages used in the silverware wrapping device are 5V and 12V. One of the SMPS units is from ASTEC MODEL: ATX202-3515, which can provide 5V at 18A and 12V at 6A. The second unit is from Power UP MODEL: P6100D, which can provide 5V at 30A and 12V at 25A, providing ample power to drive the various motion components.

An effective control system capable of communicating with the silverware sorting and orientation has been designed and implemented. The computing power required to control the silverware wrapping device has been effectively distributed between five microcontrollers. The power drivers required to drive the various drive elements based on information from the microcontrollers has also been designed and implemented. The next chapter gives an account of how the various motion components are coordinated to effectively wrap silverware in a napkin.

CHAPTER 4

IMPLEMENTAION OF THE SILVERWARE WRAPPING MECHANISM

4.1 Method Used to Program Microcontroller

Coordinating the ten drives used to operate the various mechanisms in the silverware wrapping machine is important for proper wrapping of silverware in a napkin. As mentioned in the previous chapter, five different microcontrollers are used to operate the complete system. The microcontrollers used in the system are from Microchip. Two different microcontrollers MODELS: PIC 16F876 and PIC 16F877 are used. These microcontrollers are capable of RS-232 communication standards, and these capabilities are used to program these microcontrollers using a PC, and to communicate with the rest of the silverware automation process. This provides flexibility to change the programs in the microcontrollers to accommodate any future upgrades without removing the microcontroller from the system.

To control the stepper motor driver module used in the system, a square wave signal whose frequency is given by $f = 1/n$ must be used to control the stepper motor, where n is the number of steps the stepper motor is required to move in one second. Depending on the signal to the control line controlling the direction of the stepper motor, the motor is rotated clockwise or anticlockwise. To produce a pulse of frequency ‘f’ based on a digital signal from an I/O line from the microcontroller, additional circuitry is required. In order to avoid additional circuitry, we elected to produce the square wave signal from the microcontroller. The microcontrollers are equipped with PWM, capable of producing square wave signals which does not require continuous monitoring. This method cannot be implemented as each microcontroller is limited to only two output ports capable of PWM. In order to address this problem, we program the microcontroller so that it is capable of multitasking, to produce the pulse of frequency ‘f’ using the timers present in the microcontroller without dedicating the microcontroller for the process. This method of programming the microcontroller is known as Pont’s Cooperative Scheduler, which ensures that the microprocessor keeps track of the various feedback components at all times. The microcontrollers are capable of producing an interrupt every time the built-in timer overflows. The interrupt is configured to produce an interrupt for every millisecond. In the main program, the feedback lines and communication lines are monitored, and flags are set to decide on which of the motion components are to be actuated. Based on this information, every time the timer overflows, the motion components are actuated accordingly.

The transfer in control between the interrupt service routine and the main program in the microcontroller is shown in illustrated in Figure 4.1. An interrupt is caused every millisecond to transfer the control to the interrupt service routine, where the output ports are switched on or off based on flags set in the main program, after t μs, the control is transferred to the main program. Once the execution is finished in the interrupt service routine in t μs the control is shifted back to the main program, where the various communication and feedback components are monitored in order to modify the flags required to control the motion components in the interrupt service routine.

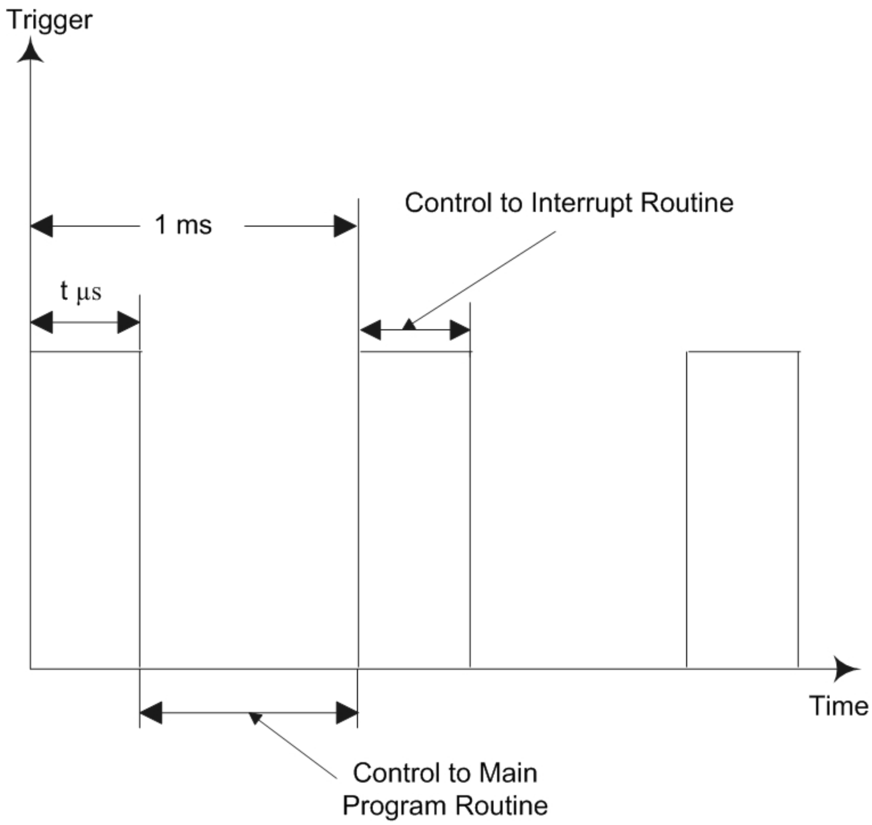


Fig 4.1 Control Transfer in Microcontroller Program

4.2 Implementation of the Napkin Bin Mechanism

The microcontroller used to control the napkin bin mechanism is a PIC 16F876 microcontroller. The bin mechanism is operated as shown in Fig 4.2.

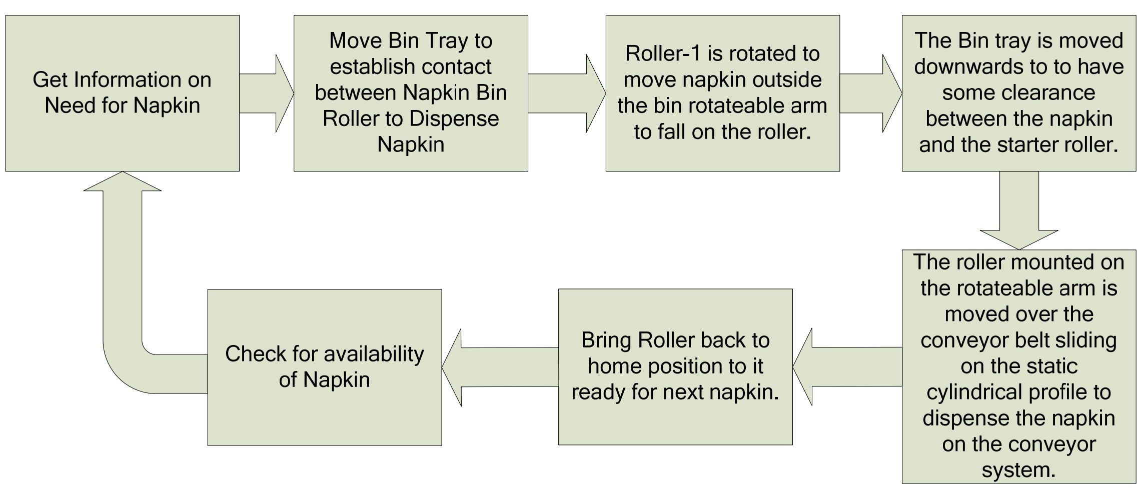


Fig 4.2 Working of the Bin Mechanism

In order to establish control over the bin mechanism, a program based on the flowchart given in Fig 4.3 was developed. The software code is given in the Appendix.

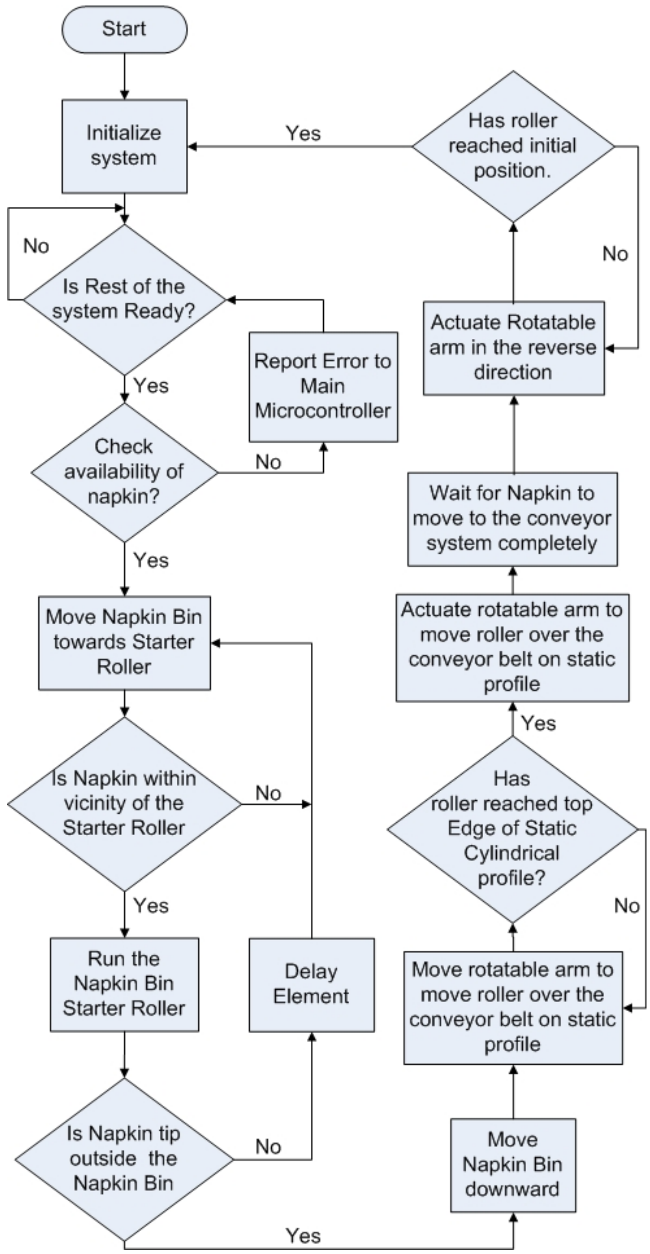


Fig 4.3 Flowchart to Control Napkin Bin

4.3 Implementation of the Silverware Drop Mechanism and Folding Mechanism-1

Once the napkin is placed on the conveyor, it is moved beneath the silverware drop mechanism where the silverware is dropped on the diagonal of the napkin that is perpendicular to the direction in which the conveyor moves. Then the long first fold is placed on the napkin using Folding Mechanism-1. The flowchart for the program used to drop silverware in the napkin and actuate Folding Mechanism-1 is shown in Fig 4.4. The software code is given in the Appendix.

The other systems driven by this program are the conveyor system and the power supply for the power H-bridge units used to control Folding Mechanisms-2 and 3. This ensures that Folding Mechanisms-2 and 3 can be shut down if there are any other problems in the system that requires a complete system shutdown.

The microcontroller used to control the silverware drop mechanism is a PIC 16F877 microcontroller, which is a 40 pin IC. It operates on an external crystal at 20MHz. This microcontroller is programmed using the Ponts Cooperative Scheduler method, since it drives the stepper motor used to drive the conveyor system, Roller-1 used in the bin mechanism, and the solenoid used to control the Folding Mechanism-1.

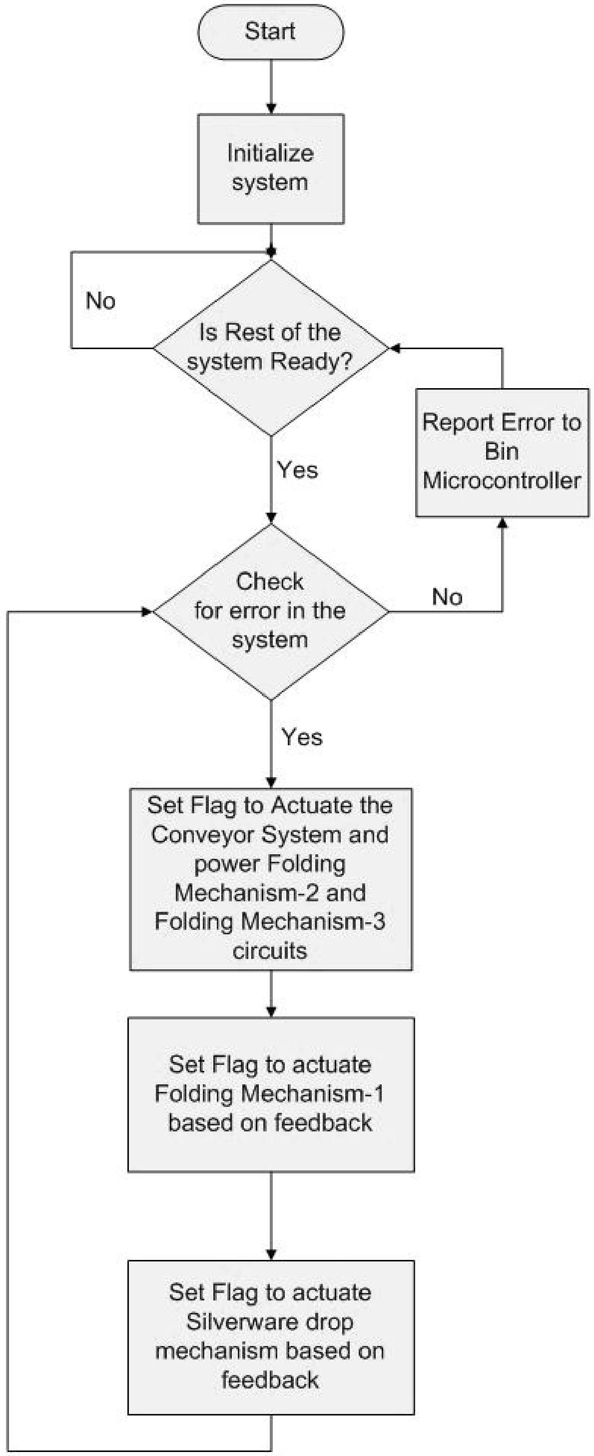


Fig 4.4 Flowchart to control Silverware Drop Mechanism and Folding Mechanism-1

4.4 Implementation of Folding Mechanisms-2 and Folding Mechaism-3

A flowchart illustrating control of folding mechanisms-2 and 3 is given in Fig 4.5.

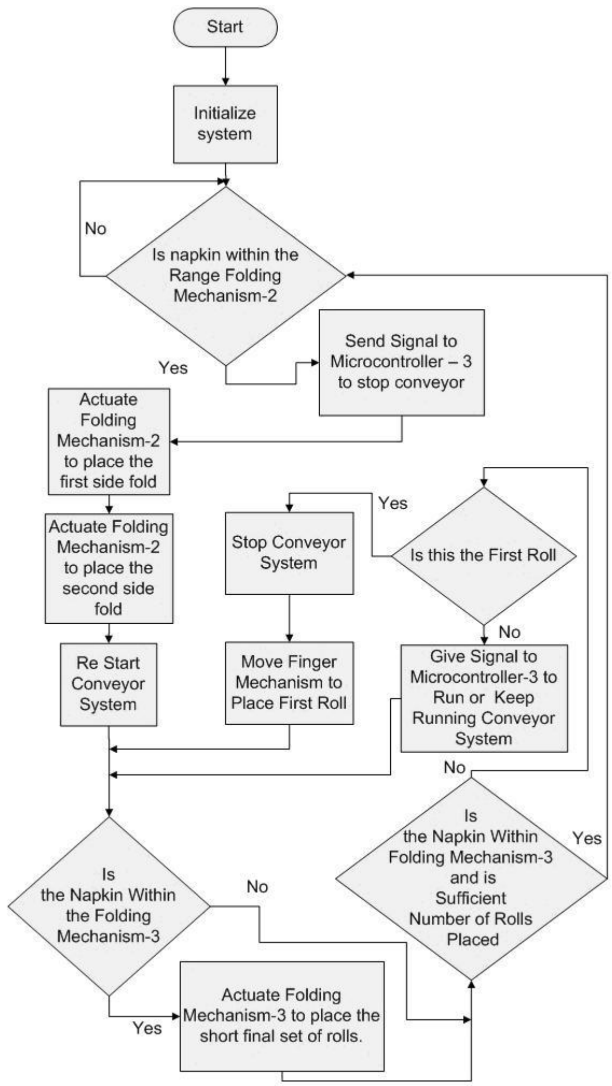


Fig 4.5 Flowchart used to control Folding Mechanism-2 and Folding Mechanism-3.

Folding Mechanisms-2 and Folding Mechanism-3 consists of three four-bar crank rocker mechanisms powered using DC gearmotors and a servo motor and a DC gearmotor to actuate the finger mechanism used to place the first fold in Folding Mechanism-3. Eight feedback lines are used to monitor the position of the napkin and the crank in the four-bar mechanism. A PIC 16F876 microcontroller is used to program the folding mechanisms. It is a 28 pin microcontroller which uses an external clock operating at 20MHz. It is desirable to use a separate microcontroller to operate the folding mechanisms owing to the number of feedback systems and the control lines required to control the gearmotors; this is an independent system with a set of H-bridge drivers, allowing the other microcontrollers to take care of the other functions relating to silverware wrapping operation when the wrapping process occurs, this feature will be particularly useful when multiple napkins need to be processed at a particular given time. The H-bridge modules are powered by external High Side switches controlled by the Silverware Drop Mechanism microcontroller.

4.5 Operating the Device at Various Speeds

The throughput of the entire device is determined by the rate at which the conveyor system is operated. Because this project was mainly a “proof of concept” undertaking, we did not provide for automatic retuning of the controls with conveyor speed changes. Accordingly, manual retuning is required for each different speed, a time-consuming process. The rotateable arm used by the napkin bin mechanism must be adjusted to the speed of the conveyor system. If the speed of the system is increased, the conveyor system will have to be stopped every time Folding Mechanism – 3 is actuated, to provide it with enough time to place the roll on the napkin before the napkin is progressed for the next roll.

A program used to control the Napkin Bin has been designed and implemented, together with a program to control the conveyor system and the various folding mechanisms, all these have been designed and implemented in both hardware and software. The next chapter presents results from experimentation with the system.

CHAPTER 5

EXPERIMENTATION AND RESULTS

5.1 Criteria for Experimentation

The design of the silverware wrapping device described in the previous chapters is implemented. It was then tested for reliability and repeatability. Individual components and the complete system were assessed as follows:

- Performance of Napkin Bin Mechanism.
- Performance of Folding Mechanism-1 used to place the first long fold.
- Performance of Folding Mechanisms-2 used to place the side folds.
- Performance of Folding Mechanism-3 used to place the final series of rolls.
- Performance of the complete system as a single unit consisting of the above mentioned mechanisms.

The performances of these components and unit were assessed by determining the percentage of napkins that were successfully manipulated by the specific component or unit. The actual number of napkins fed to a particular component was determined by the number correctly processed by the preceding component.

A batch of ten napkins each with one of ten sets of silverware consisting of a fork, knife and a spoon was used to test the silverware wrapping device. Initial testing of the silverware wrapping device indicated that in order for the napkin bin mechanism to function properly, the napkin should be relatively “stiff”, as would be exhibited by a highly starched material, to allow proper sliding of the top napkin out from the napkin stack in the napkin bin. However, Folding Mechanisms-2 and 3 required a considerably “less stiff” napkin in order to fold properly.

Numerous trials with napkins of various stiffness were made, using differing amounts of laundry starch to vary the stiffness. No single stiffness was found that would allow proper function of both the bin feed and folding mechanisms. During this testing, it was observed that higher napkin stiffness, caused by higher laundry starch, also appeared to yield lower sliding friction between the top two napkins in the bin napkin stack. However, no precise sliding friction tests were conducted. Different pressures were tried between the Feed Roller-1 and the top napkin for different napkin stiffness (with all napkins in the stack starched equally), which led to the conclusion that reducing sliding friction between napkins to an acceptable level was very important. Moreover, even for the most highly starched napkins, such friction appeared to be considerably larger than that found between the top two sheets in a paper stack in a common desktop laser or ink-jet printer, and the napkin stack compliance (in the stacking direction) appeared to be much lower.

Accordingly, to test all the components in the system, we elected to reduce sliding friction to an acceptable level by inserting between each napkin a sheet of vellum paper of the same size as the napkin. While this approach was “artificial” and required manual removal of the vellum as each napkin was fed from the bin, it did provide sufficiently low inter-layer sliding friction for the Napkin Bin Mechanism to work properly. For experimentation and in order to establish feasibility of other components, the sheets of paper present in the napkin stack were manually removed after each napkin was conveyed to the silverware drop mechanism.

5.2 Tabulation of Performance of Silverware Wrapping Device

Ten batches of napkins, each batch consisting of ten clean zero-starched napkins in a stack interleaved with vellum were used to test the system as whole, beginning with the napkin bin. Results are given in tables 5.1 through 5.5. Note that the components were tested serially, such that only those napkins that were successfully processed by an immediately processed by an immediately preceding component were used as input to its immediately following component. Napkins that were not successfully processed were removed manually. Thus, in general, the number of napkins used as inputs to serial components decreased with progression through the series.

5.2.1 Results for Napkin Bin Mechanism:

The failures that occurred in the napkin bin mechanism indicated in Table 5.1, can be attributed for the most part to variations in the amount of napkin that was delivered by the napkin bin roller. In almost all cases of failure, the napkin was held between the belt traversing over the partial cylindrical profile and the rotateable arm, but since the length of the napkin delivered by the napkin bin roller was sometimes less than needed, the napkin could not be successfully be delivered to the silverware drop and Folding Mechanism – 1.

Trial Number	No of Napkins processed	No of napkins processed correctly	Percent Correct (%)
1	10	8	80
2	10	6	60
3	10	8	80
4	10	8	80
5	10	8	80
6	10	9	90
7	10	8	80
8	10	10	100
9	10	8	80
10	10	9	90
Total	100	82	82

Table 5.1: Performance of Napkin Bin Mechanism

We note from Table 5.1 that approximately two out of ten napkins failed to be properly delivered from the napkin bin, which is well above a desired failure rate of less than one in hundred.

5.2.2 Results for Folding Mechanism-1 and Silverware Drop Mechanism:

Table 5.2 indicates an overall success rate of approximately 93% for the combined Silverware Drop and Folding Mechanism – 1. However, the failures occurred only in Folding Mechanism-1. These failures were all caused by the napkin leading tip being off-center as it approached Folding Mechanism -1, such that the fingers could not grasp the napkin tip to execute the first fold over the silverware set. Again, the failure rate was well above a desired rate of less than 1%.

Trial Number	No of Napkins processed	No of napkins processed correctly	Percent Correct (%)
1	8	7	87.5
2	6	6	100
3	8	7	87.5
4	8	7	87.5
5	8	8	100
6	9	7	77.8
7	8	8	100
8	10	10	100
9	8	7	87.5
10	9	9	100
Total	82	76	92.7

Table 5.2: Performance of Folding Mechanism-1 and Silverware Drop Mechanism

5.2.3 Results for Folding Mechanisms-2:

Table 5.3 presents results for Folding Mechanism-2. Failures occurred when the position of the napkin was shifted towards one side from the other. Again, such misalignment was caused principally by failure of the napkin bin mechanism to deliver a napkin centered on the conveyor belts. Also some of the failures can be attributed to the inability of the folding mechanism to force the side folds to be placed on the napkin, thereby allowing the napkin to spring back open. This appeared to be due to varying mechanical properties of different napkins.

Trial Number	No of Napkins processed	No of napkins processed correctly	Percent Correct (%)
1	7	7	100
2	7	6	85.7
3	7	6	85.7
4	7	7	100
5	8	6	75
6	7	6	85.7
7	8	7	87.5
8	10	10	100
9	7	6	85.7
10	9	9	100
Total	76	69	90.8

Table 5.3: Performance of Folding Mechanisms-2

Clearly, a failure rate of 9% is excessive compared to a desired rate of less than 1%.

5.2.4 Results for Folding Mechanism-3:

Folding Mechanism-3 was reasonably successful with zero-starched napkins, as shown in Table 5.4. Some of the folded napkins were poor folds, again attributed to the misalignment in the position of the napkin on the belt. Folds placed by Folding Mechanisms – 1 and 2 were not consistent in quality, due mainly to variability in napkin properties from one to another accordingly, the sensor sensing the position of the napkin being fed to Folding Mechanism – 3 was sometimes triggered either earlier or latter than desired, thereby causing a poor wrap.

Trial Number	No of Napkins processed	No of napkins processed correctly	Percent Correct (%)
1	7	7	100
2	5	5	100
3	6	6	100
4	7	7	100
5	6	6	100
6	6	6	100
7	7	7	100
8	10	9	90
9	6	6	100
10	9	9	100
Total	69	68	98.6

Table 5.4: Performance of Folding Mechanism-3

While the percentage of acceptably rolled napkins was higher for this mechanism than another, “acceptable” was very broadly defined as described in section 5.26.

5.2.5 Results for Silverware Wrapping Mechanism as a Unified Device:

Failure of any of the above mentioned devices is a failure by the Silverware Wrapping Mechanism as a unified device. The percent of napkins that have been processed correctly is given by the product of the efficiencies of the individual devices. All those napkins that have passed successfully through all the mechanisms are considered successful. Table 5.5 presents results, showing that in overall, the unified device processed correctly only 68% of the napkins starting in the napkin bin. Such a rate is well below a target level of 99% or better, leaving large opportunity for improvement

Trial Number	No of Napkins processed	No of napkins processed correctly	Percent Correct (%)
1	10	7	70
2	10	5	50
3	10	6	60
4	10	7	70
5	10	6	60
6	10	6	60
7	10	7	70
8	10	9	90
9	10	6	60
10	10	9	90
Total	100	68	68

Table 5.5: Performance of the Silverware Wrapping Device

5.2.6 Quality of Wrap

Successfully wrapped silverware is categorized into one of four types, identified and pictured in Figures 5.1 – 5.4. The first type is a Very Good Wrap, shown in Figure 5.1, a set of wrapped silverware such that opposite ends of the wrap are planar and approximately parallel with each other, and the wrapping is significantly tight.

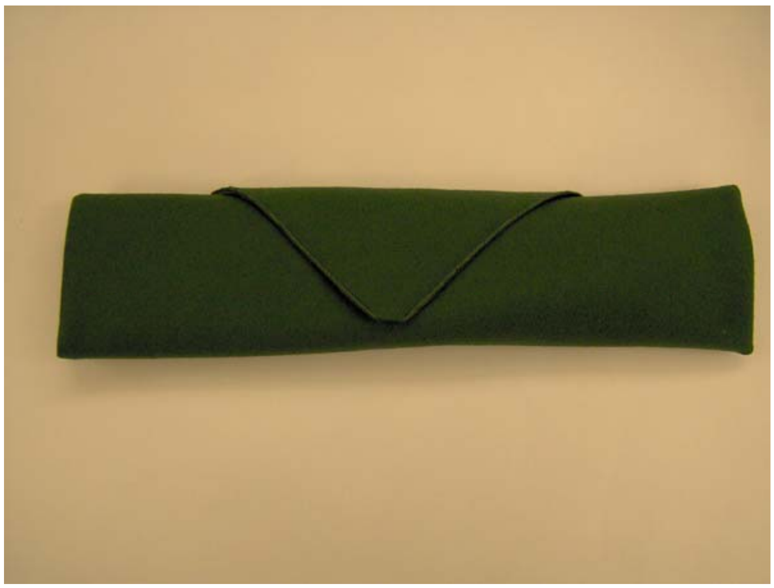


Fig 5.1 Picture of a Very Good Wrap

The second type is a Good Wrap, Figure 5.2, a set of wrapped silverware failing as a Very Good Wrap because the ends of the wrap are not parallel with each other.

Fig 5.2 Picture of a Good Wrap

The third type is a Fair Wrap, Figure 5.3, a set of wrapped silverware sufficiently tight, but either one or both ends are not planar.

Fig 5.3 Picture of a Fair Wrap

The fourth type is a Poor Wrap, Figure 5.4, a set of wrapped silverware successfully wrapped, but with an insufficiently tight or loose wrap.

Fig 5.4 Picture of a Poor Wrap

The process is considered successful only if the roll falls into the category of napkins described below and illustrated in Figures 5.1 – 5.4.

Table 5.6 gives results on the number of wrapped silverware categorized by the type of wrap specified in the above section.

Trial Number	Number of Successful Napkins	Number of Very Good Wraps	Number of Good Wraps	Number of Decent Wraps	Number of Poor Wraps
1	7	1	2	3	1
2	5	1	2	1	1
3	6	0	3	1	2
4	7	2	2	2	1
5	6	2	2	1	1
6	6	1	3	2	0
7	7	1	3	2	1
8	9	1	5	2	1
9	6	2	1	2	1
10	9	1	3	5	0
Total	68	12	26	21	9
Percentages	100	17.65	38.24	30.88	13.24

Table 5.6 Categorizing the Wrapped Silverware

Clearly, there is room for improvement in decreasing to zero the percentage of fair and poor wraps.

5.2.7 Experimentation of Wrapping Process with Manual Napkin Feed

In order to verify the above mentioned inferences, a test was conducted to determine if most of the failures are because of the napkin bin mechanism. Twenty zero-starched napkins were manually fed to the wrapping system consisting of Silverware Drop Mechanism, Folding Mechanism – 1, Folding Mechanisms – 2 and Folding Mechanism – 3. Table 5.4 presents a summary of the type of roll obtained during this process. None of the wraps can be categorized under Good Wrap as the napkins were fed manually ensuring that the opposite ends of the wrap are planar and approximately parallel.

Total Number of Napkins Processed	Number of Successful Napkins	Number of Very Good Wraps	Number of Fair Wraps	Number of poor Wraps
20	20	13	7	0
Percentages	100	65	35	0

Table 5.7: Categorizing the Wrapped Silverware by Manual Feed of Napkin

Clearly, from Table 5.7, it is evident that the failure rate is due to the erroneous feeding of napkins by the napkin bin mechanism to the folding mechanisms. Also 65% of the napkins were Very Good Wraps, and the remaining 35% were Fair Wraps, indicating that Poor Wraps were formed by the erroneous feed of the napkin bin mechanism.

Results in Tables 5.1 – 5.6, together with observations are used to derive conclusions given in the next chapter, together with Recommendations for enhancing the device.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

This research has focused on the design, development, implementation and evaluation of an Automated Silverware Wrapping Mechanism. This system is capable of retrieving a single napkin from a stack of napkins and wrapping a set of silverware placed in a silverware drop box. Contributions of this research can be summarized as follows:

- Design and construction of a mechanism for moving a stack of napkins towards and away from a napkin singulation mechanism.
- Design and implementation of a napkin retrieving mechanism consisting of a napkin bin roller, moveable arm and belt mechanism. Space savings were realized by locating the napkin bin beneath the rest of the mechanisms.
- Design and implementation of an effective conveyor system to convey the napkin retrieved from the napkin bin to the wrapping mechanisms.
- Design and implementation of a silverware drop mechanism to drop silverware on the napkin as it travels below this mechanism.
- Design and implementation of Folding Mechanism-1 used to place the first long fold on the napkin over the silverware.

- Design and implementation of Folding Mechanism-2, consisting of four bar mechanisms powered by DC gearmotors, to place side folds on the napkin containing silverware with a first fold.
- Design and implementation of Folding Mechanism-3, consisting of a four bar mechanism powered by a DC gearmotor, specially designed folding pads, and a servo and DC gearmotor powered finger holding mechanism, to place the final sequence of rolls.
- Design and implementation of the control unit using 6 microcontrollers, which co-ordinate with one another to establish the entire wrapping process. The control unit is equipped with additional ports to obtain feedback to accommodate future modifications, together with additional control units to control additional stepper, DC motors and solenoids.
- Design and implementation of a feedback mechanism using optical en-coders to co-ordinate the three stepper motors used to move the bin containing the stack of napkins.
- Design and implementation of feedback mechanisms to detect the position of napkins and trigger the appropriate mechanisms.
- Research and selection of various drive controllers to control different motors, installed appropriately depending on the mechanism.
- Provision of the system with communication capabilities to work as a unit in the existing Silverware Automation Process, consisting of singulation, identification, inspection and sorting.
- Design and implementation of Algorithms to:

    - Control the Silverware Drop Mechanism, Folding Mechanism – 1 and Conveyor unit.
    - Control Folding Mechanisms – 2 and Folding Mechanism – 3.
- Implementation of communication protocols in order for the complete system to function as a single unit.
- The entire unit consisting of various components correctly processed 68% of the napkins entering the system. Correctly processed percentages of the various units were:
    - Napkin Bin Mechanism: 82%.
    - Folding Mechanism-1 and Silverware Drop Mechanism: 92.7%
    - Folding Mechanism-2: 90.8%.
    - Folding Mechanism-3: 98.6%.

Clearly the successful processing rate of the overall system and each of components is well below a desired target of more than 99%. There is much room for improvement, and recommendations addressing these are given in the next section. The Napkin Bin feed mechanism, which not only had the lowest success rate, but also caused the low success rate of Folding Mechanism.

6.2 Recommendations

Following are recommendations for enhancement of the silverware wrapping process:

- Table 5.7 clearly indicates that the success rate of the device depends heavily on the way that napkin is fed to the wrapping section of the process. An improvised Napkin Bin Mechanism must be designed to solve the problem of sliding friction between the napkins, which requires the current Napkin Bin Mechanism to use a stack of napkins interleaved with paper. The improvised Napkin Bin Mechanism must also be designed to deliver properly aligned napkins to the silverware wrapping device, when fed to the wrapping section of the silverware wrapping device.

- Increase distance between the two fingers in Folding Mechanism – 1 to obtain an improved long fold, thereby minimizing the chances for a failure in Folding Mechanism-2.
- Improve Folding Mechanism-2 by making the four bar mechanisms more robust and stable to obtain more angular displacement for the folding pads.
- Improve Folding Mechanism-3 by adding additional sensors. It currently utilizes only a single pair of infrared trans-receiver pair to sense the position of the napkin over it, causing the sensor to trigger at times when Folding Mechanism-2 places a fold which is slightly offset, causing the ends to overhang ahead of the actual napkin containing silverware.
- Replace the stepper motors used to drive the bin roller and the conveyor system with servomotors in order to avoid heating problems.
- Design and implement a system to collect silverware sets and feed to the silverware drop mechanism.

REFERENCES

1. Brown, William Edward; Clanton, Dwight; Mudd, Reginald; Gray, Charles Woodard. Apparatus and method for banding wrapped silverware”, U.S. Patent - 6,202,387, September 2, 1999.
2. Dunbar, Michael; Vohnout, Vincent. “Method for Wrapping Silverware in a Napkin”, U.S. Patent - 5,469,688, July 26, 1993.
3. Gray, Charles Woodard; Brown, William Edward; Clanton, Dwight; “Apparatus and Method for wrapping silverware within a napkin”, U.S. Patent – 6,023,913, September 9, 1998.
4. Heisey, John. “Apparatus and method for automatically wrapping silverware in a napkin”, U.S. Patent - 6,615,566, January 14, 2002.
5. Hasimoto, Sachiko. “Separation of Silverware For Machine Vision Sorting and inspection”. M.S. thesis, School of Mechanical and Aerospace Engineering, Stillwater. OK: Oklahoma State University, May 1995.
6. Latvala, Jessica, Student, School of Mechanical and Aerospace Engineering, Oklahoma State University, Research Log Book, BSME 1999.
7. Loisson, Michael , Student, School of Mechanical and Aerospace Engineering, Research Log Book, BSME 1999.
8. Lolla, Venugopal. “Identification and Inspection of Silverware using Machine Vision”, M.S thesis, School of Mechanical and Aerospace Engineering, Stillwater, OK: Oklahoma State University, 2005.

9. Peddi, Ravi. “Silverware Sorting and Orientation System”, M.S. thesis, School of Mechanical and Aerospace Engineering, Stillwater, OK: Oklahoma State University, 2005.
10. Vetsch, Kevin. “Method and apparatus for folding a napkin around an eating utensil”, U.S. Patent - 6,023,908, May 27, 1997.
11. Yeri, Sandeep. “Classification of silverware pieces using machine vision”, M.S. thesis, School of Mechanical and Aerospace Engineering, Stillwater, OK: Oklahoma State University, 2003.

APPENDIX A

ABSTRACTS OF PATENTS

1 U.S. Patent *5,469,688*

Title: Method for wrapping silverware in a napkin

Inventors: Dunbar; Michael D. (873 Neil Ave., Columbus, OH 43215); Vohnout; Vincent J. (Columbus, OH);

Appl. No.: 097847

Filed: July 26, 1993.

Abstract

An apparatus and method for automatically wrapping at least one eating utensil in a napkin includes a receiving area for at least one utensil and at least one napkin, and a mechanism for automatically wrapping. The mechanism for automatically wrapping includes a frame, a flexible belt disposed in the frame, and a belt manipulation device. The belt has a first surface adapted for frictional contact with the napkin. At least a portion of the belt is movable to form a trough in which the napkin and utensil are manipulated and urged by frictional contact with the belt to fold and roll, automatically wrapping the utensil.

2 U.S. Patent *6,023,908*

Title: Method and apparatus for folding a napkin around an eating utensil

Inventors: Vetsch; Kevin R. (1306 NW. 63rd Ter., Kansas City, MO 64118)

Appl. No.: 864014

Filed: May 27, 1997 .

Abstract

The present invention relates to a method and apparatus for rolling a napkin around silverware. The device comprises a housing having a conveyor belt system for advancing a napkin there through along a substantially horizontal path. As the napkin traverses said path, it contacts a folding belt which causes a corner of the napkin to be folded back onto itself. A knife, fork or other piece of silverware is delivered from storage bins in timed relation to the movement of the napkin to ensure placement thereon. The napkin and silverware then contact a rapidly moving rolling belt which rolls the napkin around the silverware. The rolled napkin and silverware are then transported out of the housing and on to an external holding bin.

3 U.S. Patent *6,202,387*

Title: Apparatus and method for banding wrapped silverware.

Inventors: Brown; William Edward (Nashville, TN); Clanton; Dwight (Shelbyville, TN); Mudd; Reginald M. (503 Dawn Pl., Lebanon, TN 37087); Gray; Charles Woodard (Gallatin, TN)

Appl. No.: 388779

Filed: September 2, 1999.

Abstract

An apparatus and method for banding a wrapped silverware arrangement with a two-sided segment of banding material bearing a treatable adhesive on one side thereof

utilizes a pair of jaw members which are movable between open and closed conditions for capturing a wrapped silverware arrangement and a clamp assembly for holding an end of a banding material segment. With an end of the banding material segment held by the clamp assembly and the jaw members being disposed in an open condition, the jaw members are lowered onto a wrapped silverware arrangement so that the banding material segment is draped across the wrapped silverware arrangement and so that subsequent movement of the jaw members to the closed condition captures the silverware arrangement between the jaw members and positions the banding material segment in an inverted U-shaped orientation about the wrapped silverware arrangement. The one end of the banding material segment is thereafter released from the clamp assembly and exposed to heat to render the adhesive borne by one region of the segment tacky. The wrapped silverware arrangement is thereafter rotated about its longitudinal axis so that the tacky region of adhesive is moved into contact with another surface region of the banding material segment so that the banding material segment is thereby secured in a band about the wrapped silverware arrangement.

4 U.S. Patent *6,023,913*

Title: Apparatus and method for wrapping silverware within a napkin.

Inventors: Gray; Charles Woodard (Tallahassee, FL); Brown; William Edward (Nashville, TN); Clanton; Dwight (Shelbyville, TN)

Appl. No.: 188814

Filed: November 9, 1998.

Abstract

An apparatus and method for wrapping a napkin about silverware items into a wrapped arrangement involves a worktable within which is provided an upwardly-opening trough for accepting a napkin and silverware items positioned therein. The napkin is automatically placed over the opening of the trough in a spread condition and then silverware items are automatically directed onto the napkin so that the silverware falls to the bottom of the trough thereby positioning the napkin between the silverware and the bottom of the trough. A rotatable blade is positioned within the trough for automatically spinning the napkin and silverware items about the longitudinal axis of the trough until the napkin and silverware items are wound in a wrapped arrangement. The wrapped arrangement is thereafter removed from the trough in preparation of a subsequent napkin-wrapping operation.

5 U.S. Patent *6,615,566*

Title: Apparatus and method for automatically wrapping silverware in a napkin.

Inventors: Heisey; John L. (Phenix, AL)

Appl. No.: 047441

Filed: January 14, 2002.

Abstract

An apparatus and method are provided for wrapping a napkin around one or more utensils, wherein the utensils each have first end and a second ends defining a

longitudinal axis. The utensil wrapping apparatus includes a utensil manipulator for placing the utensil adjacent to the surface of the napkin. First and second utensil engaging devices are provided for releasably engaging the first and second ends of the utensil and the portions of said napkin respectively adjacent the first and second ends of said utensil, such that said napkin and said utensil are held in a fixed relative position. A napkin guide for wrapping the napkin around the utensil, wherein the napkin guide is manipulable to orbit the longitudinal axis such that the napkin guide contacts the napkin and thereby wraps the napkin around said utensil.

6 U.S. Patent *6,615,566*

Title: Automated flatware and napkin assembling apparatus.

Inventors: Miano; Mario L. (731 Sitka St., Fort Collins, CO 80524); Nelson; Kevin W. (1401 Sugarpine St., Fort Collins, CO 80524)

Appl. No.: 641357

Filed: August 15, 2003.

Abstract

An apparatus that wraps dinner flatware in a paper napkin in an automated manner and secures it with a paper label is disclosed. The invention includes input hoppers for knives, forks, spoons, and napkins which are located on the front and top part of the unit. Additionally, a roll of paper napkins and a roll of securing paper labels are provided on top with their own automatic dispensing means. A plastic cover secures the entire top of the invention to protect it from dust and dirt as well as accidental contact during operation. Internal mechanisms then take one of each piece of flatware and fold and roll it in a napkin. The completed napkin unit then drops out the bottom of the invention, where it is collected for use. The invention is capable of processing up to 50 sets of flatware during one operating run. The use of the invention provides time and labor savings to restaurants and other eating establishments while producing rolled flatware secured in a paper napkin in fast and sanitary manner.

APPENDIX B

Drawings of Silverware Wrapping Device

The following statements hold good for drawings given in Figures B1 – B35:

- All Dimensions indicated are in inches unless specified
- Support structures are made up of slotted angles of size 2.25” by 1.5”, and thickness 0.1” unless specified.
- Support structures are not marked for dimensions until and unless required.
- Folding Pads are made up of wooden material of thickness 0.1”.

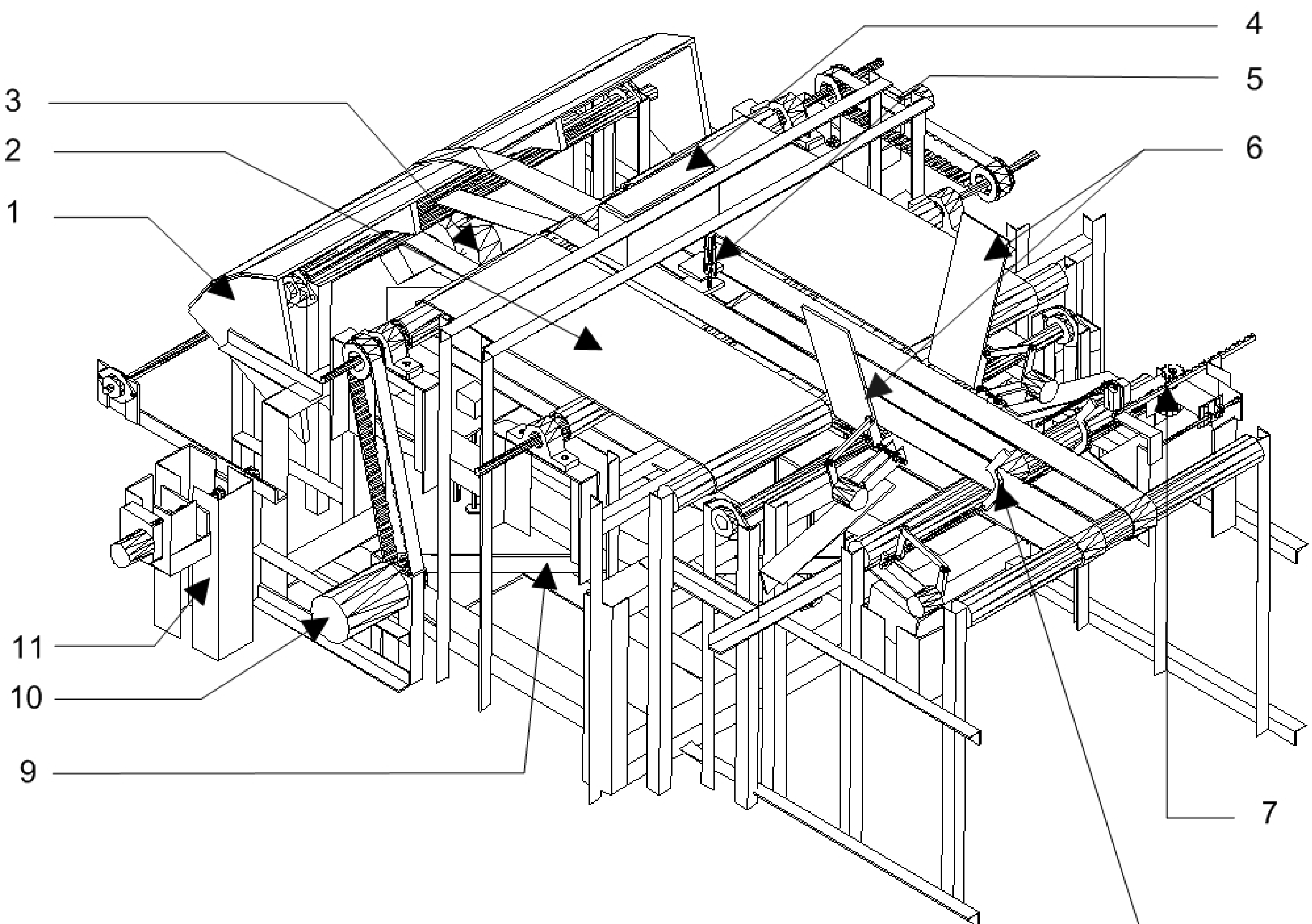
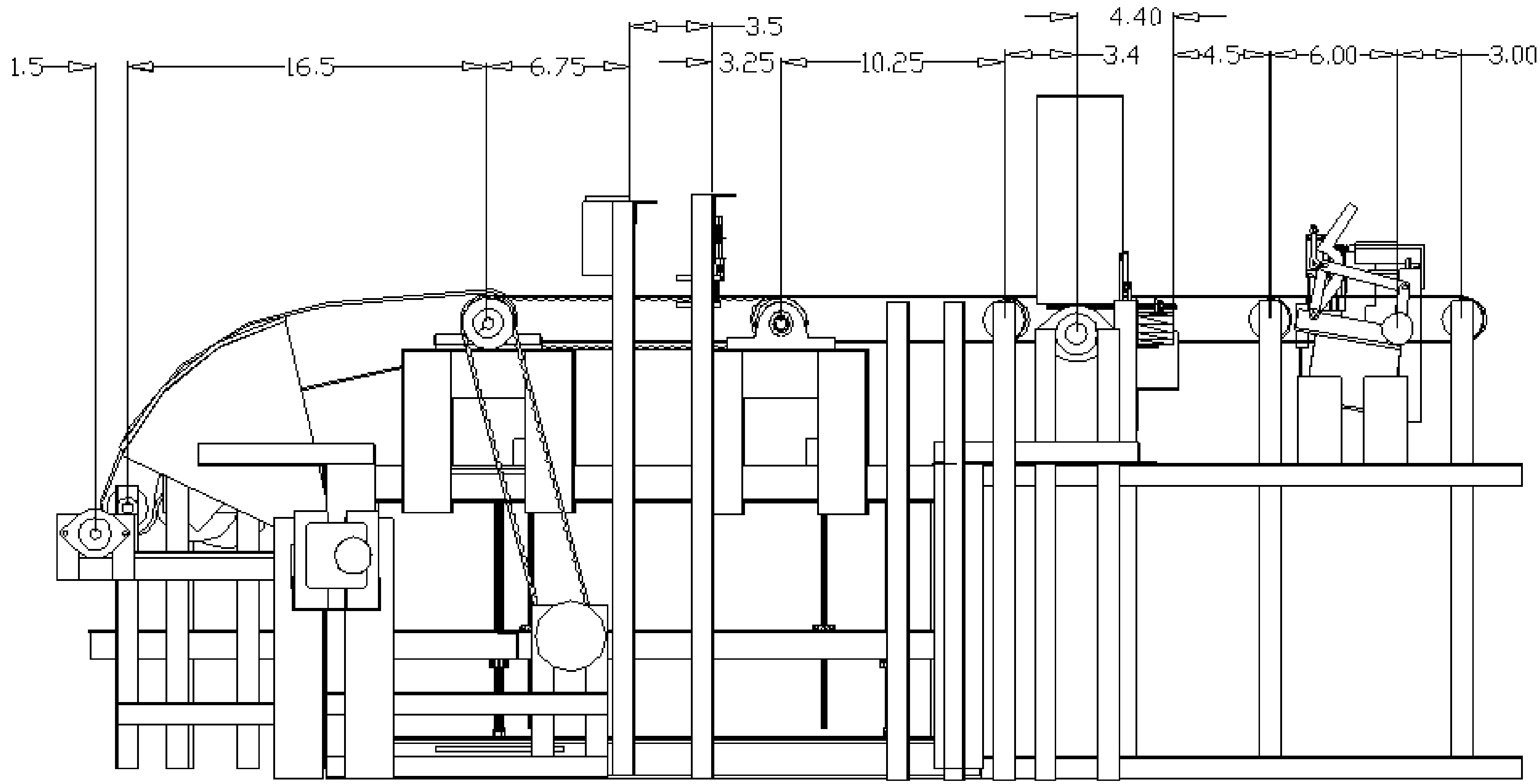


Figure B1: Orthographic View of the Silverware Wrapping Mechanism

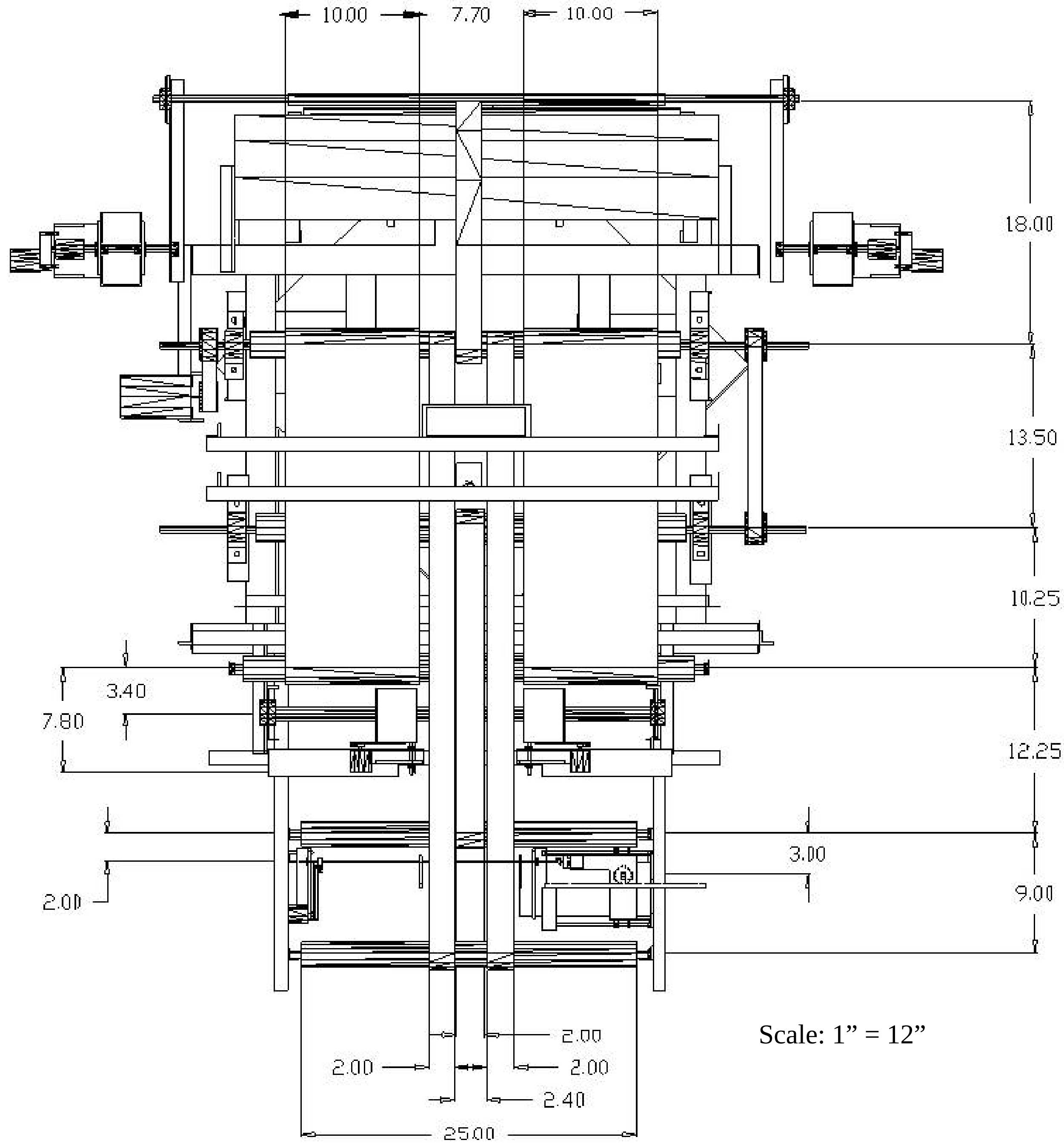
Table B1: Reference for Figure B1

Label Number	Description
1	Static Guide for over which belt to pickup the napkin from the napkin bin traverses
2	Conveyor System
3	Napkin Bin Roller
4	Silverware Drop Mechanism
5	Folding Mechanism-1
6	Folding Mechanisms-2
7	Finger Mechanism used by Folding Mechanism – 3
8	Folding Mechanism – 3
9	Napkin Bin Tray to Hold Napkins
10	Stepper Motor To Drive Conveyor System
11	Arm Mechanism containing Roller to Pickup Napkin



Scale: 1” = 8”

Figure B2: Side View of the Silverware Wrapping Mechanism



Scale: 1” = 12”

Figure B3: Top View of the Silverware Wrapping Mechanism

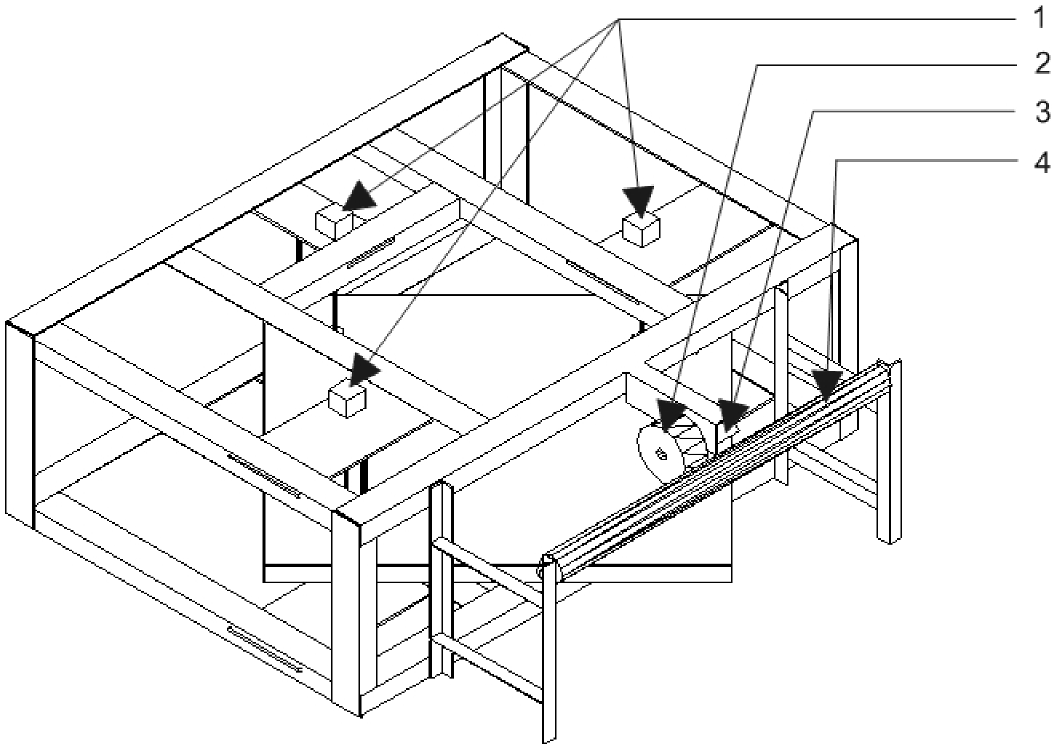


Figure B4: Orthographic view of napkin Bin Mechanism

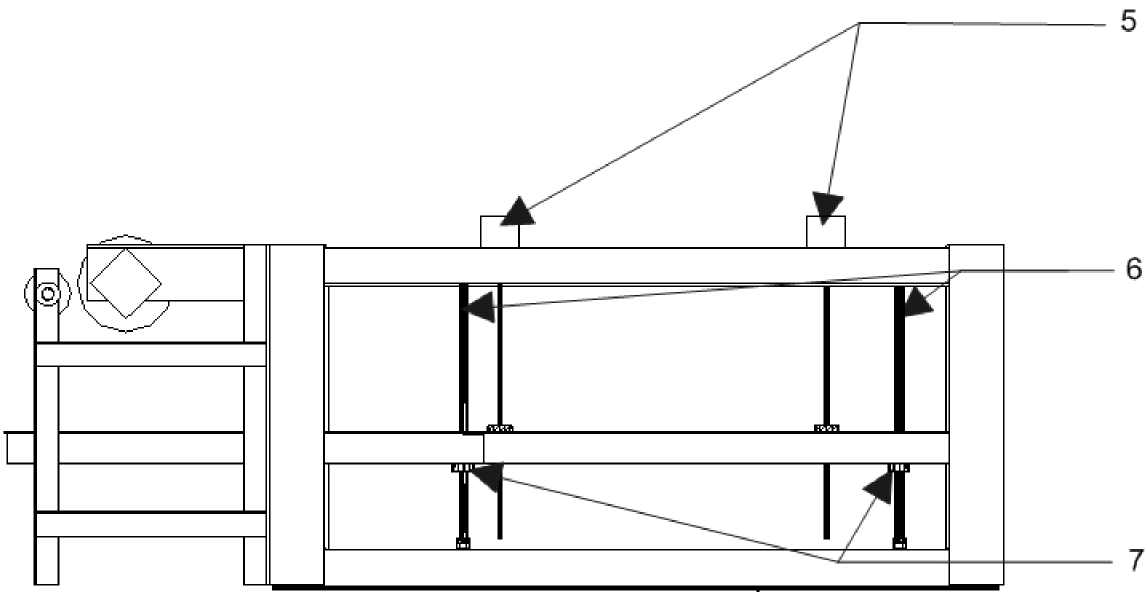


Figure B5: Side view of napkin Bin Mechanism

Table B2: Reference for Figure B4 & B5

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1&5	Lead Screw Motor	HSI motors	E43H4C-12-8070117	3
2	Bin Roller	McMaster Carr	2497K5	1
3	Stepper Motor to Power Bin Roller	Applied Motion	834-HT23-397	1
4	Conveyor Idler	McMaster Carr	2277T31	1
6	Bearing Shaft	McMaster Carr	6061K33	3
7	Linear Bearing	McMaster Carr	6483K23	3

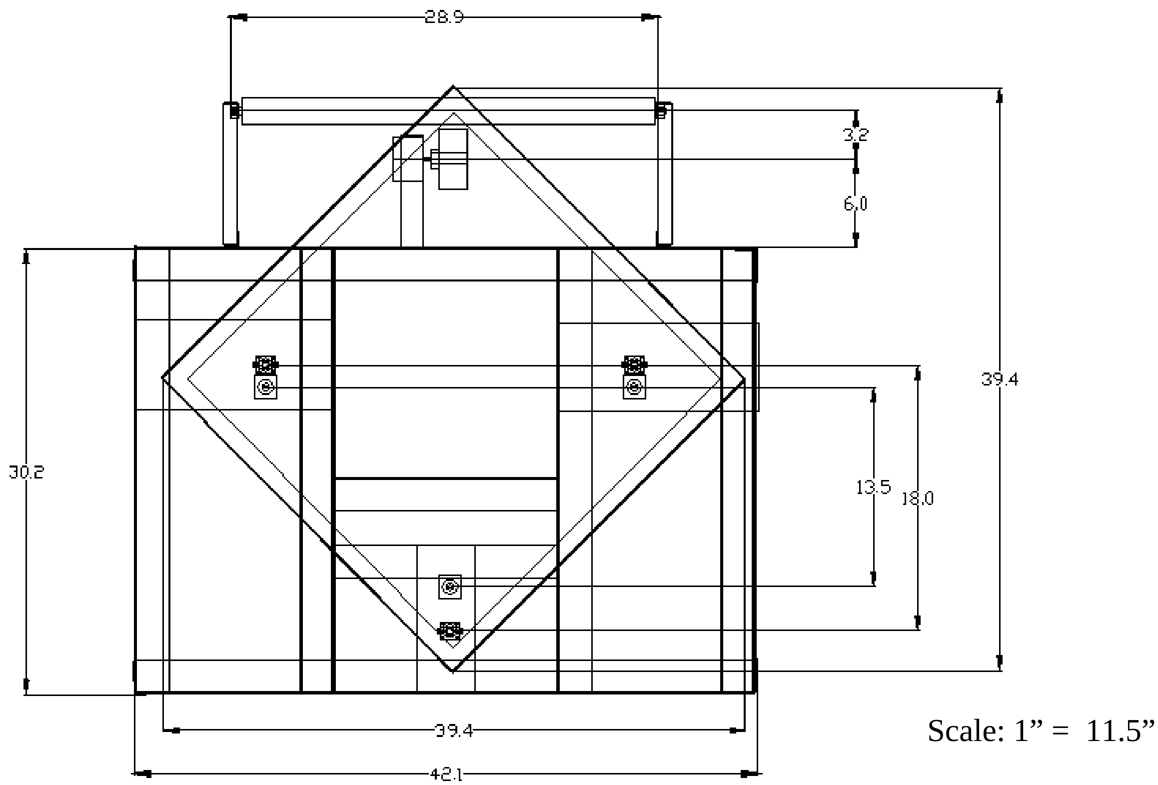


Figure B6: Top view of napkin Bin Mechanism

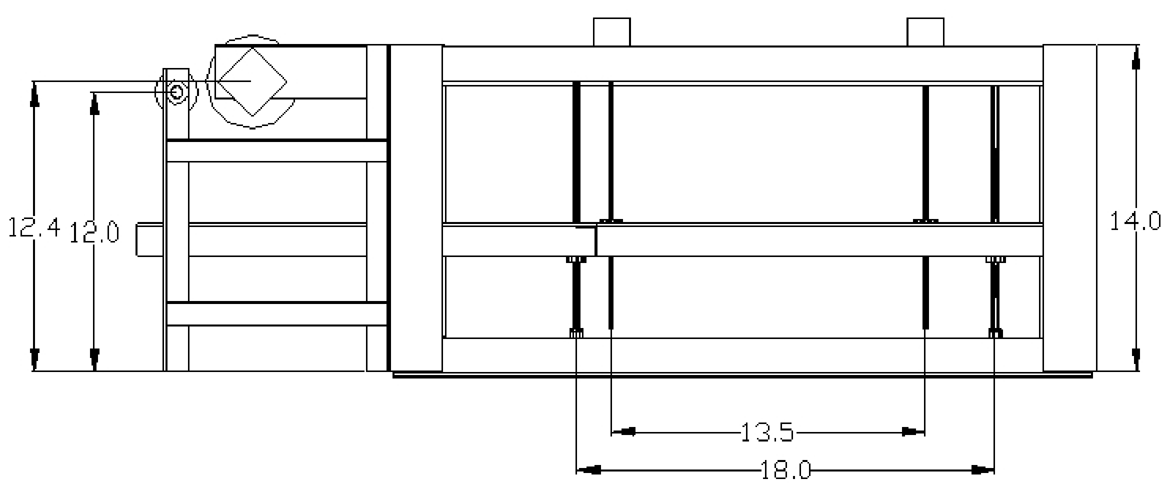


Figure B7: Side view of napkin Bin Mechanism

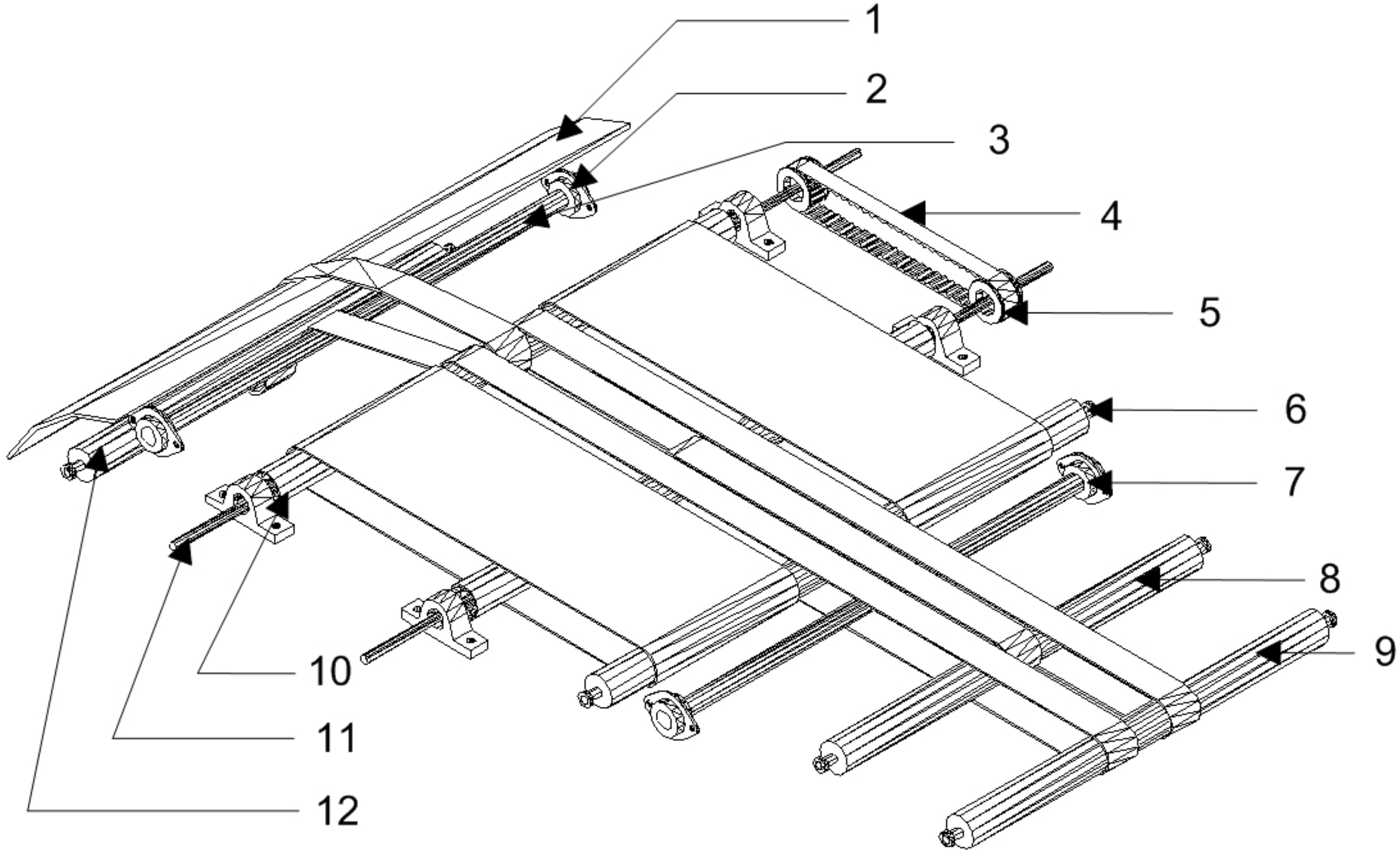
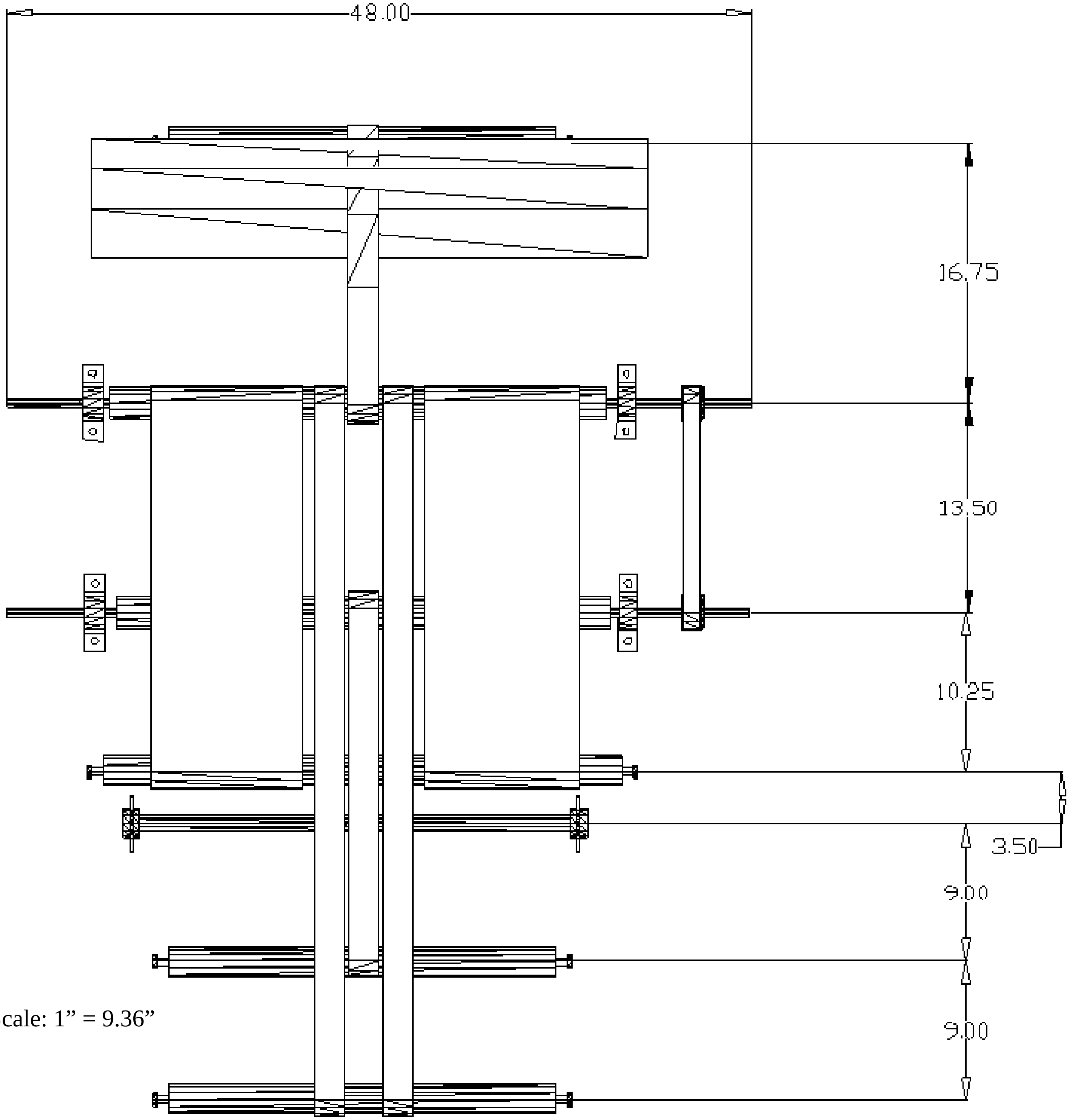


Figure B8: Orthographic View of the
Conveyor Mechanism

Table B3: Reference for Figure B8

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1	Profile Used by napkin bin mechanism to pick napkin	-	-	1
2	Bearing Shafts	McMaster Carr	6061K25	1
3&12	Bearing	McMaster Carr	5913K54	2
4	Timing Belt	McMaster Carr	6484K404	1
5	Timing Pulley and JA Bushing	Mc Master Carr	Timing Pulley: 6495K511	2
			JA Bushing: 6086K11	2
6	Conveyor Roller	McMaster Carr	2277T34	1
7	Conveyor Belt Support Shaft	McMaster Carr	Shaft: 6061K25	1
			Bearing: 5913K54	2
8	Conveyor Roller	McMaster Carr	2277T28	1
9	Conveyor Roller	McMaster Carr	2277T28	1
10	Conveyor Drive Pulley	McMaster Carr	Pulley: 5706K26	2
			Shaft: 6061K73	2
			Bearings: 6244K51	4
11	Shaft(Used in Drive Pulley)	McMaster Carr	Shaft: 6061K73	2



Scale: 1” = 9.36”

Figure B9: Top View of the Conveyor Mechanism

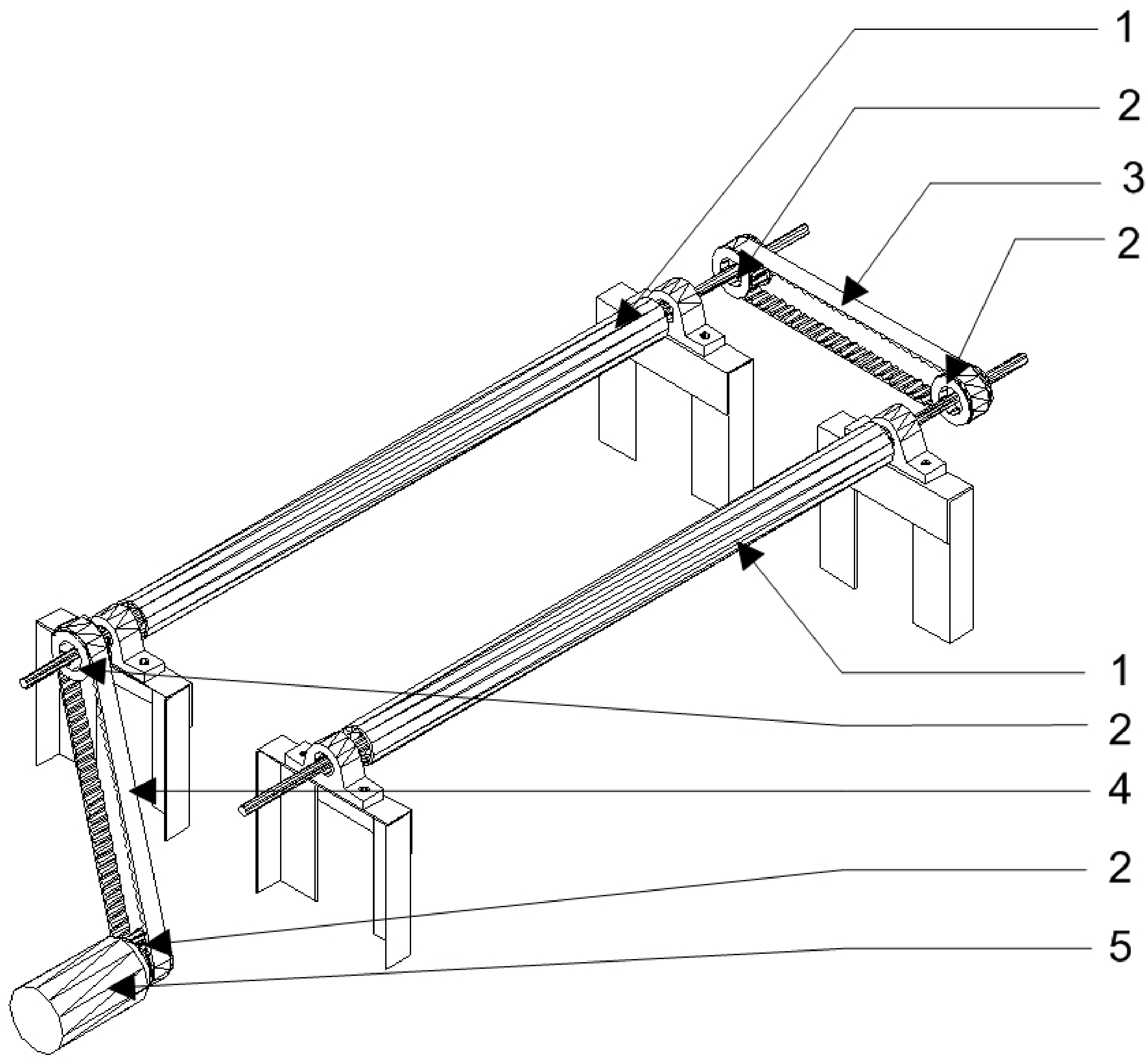


Figure B10: Orthographic View of the Conveyor Drive Mechanism

Table B4: Reference for Figure B10

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1	Conveyor Drive Roller(Consists of Pulley, Shaft and Bearings)	Mc Master Carr	Pulley: 5706K26	2
			Shaft: 6061K73	2
			Bearings: 6244K51	4
2	Timing Gears and JA Bushing	Mc Master Carr	Timing Gear: 6495K511	3
			JA Bushing: 6086K11	
3	Timing Belt	Mc Master Carr	6484K404	1
4	Timing Belt	Mc Master Carr	6484K175	1
5	Stepper Motor	Anaheim Automation	34Y307D-LW8	1

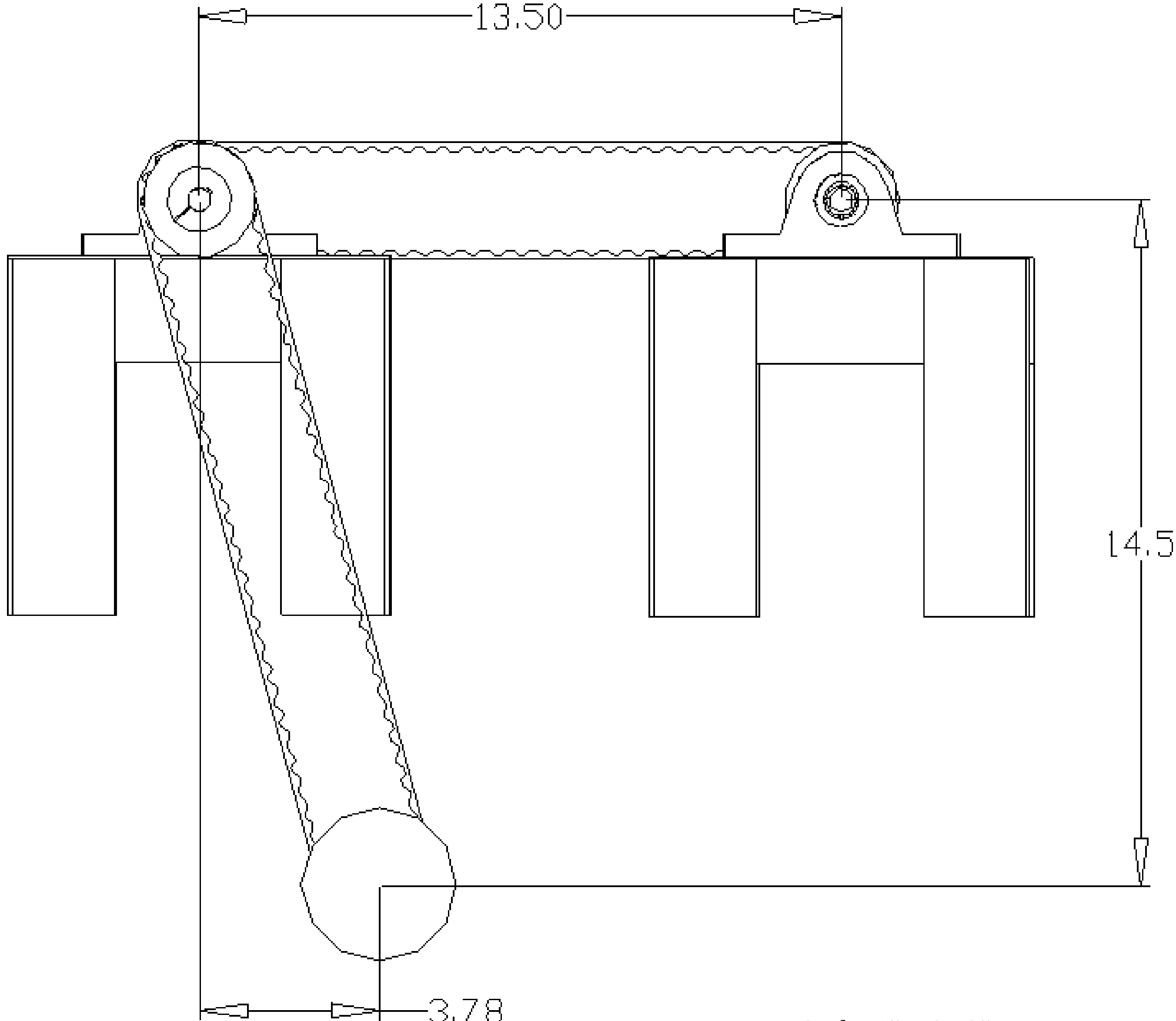
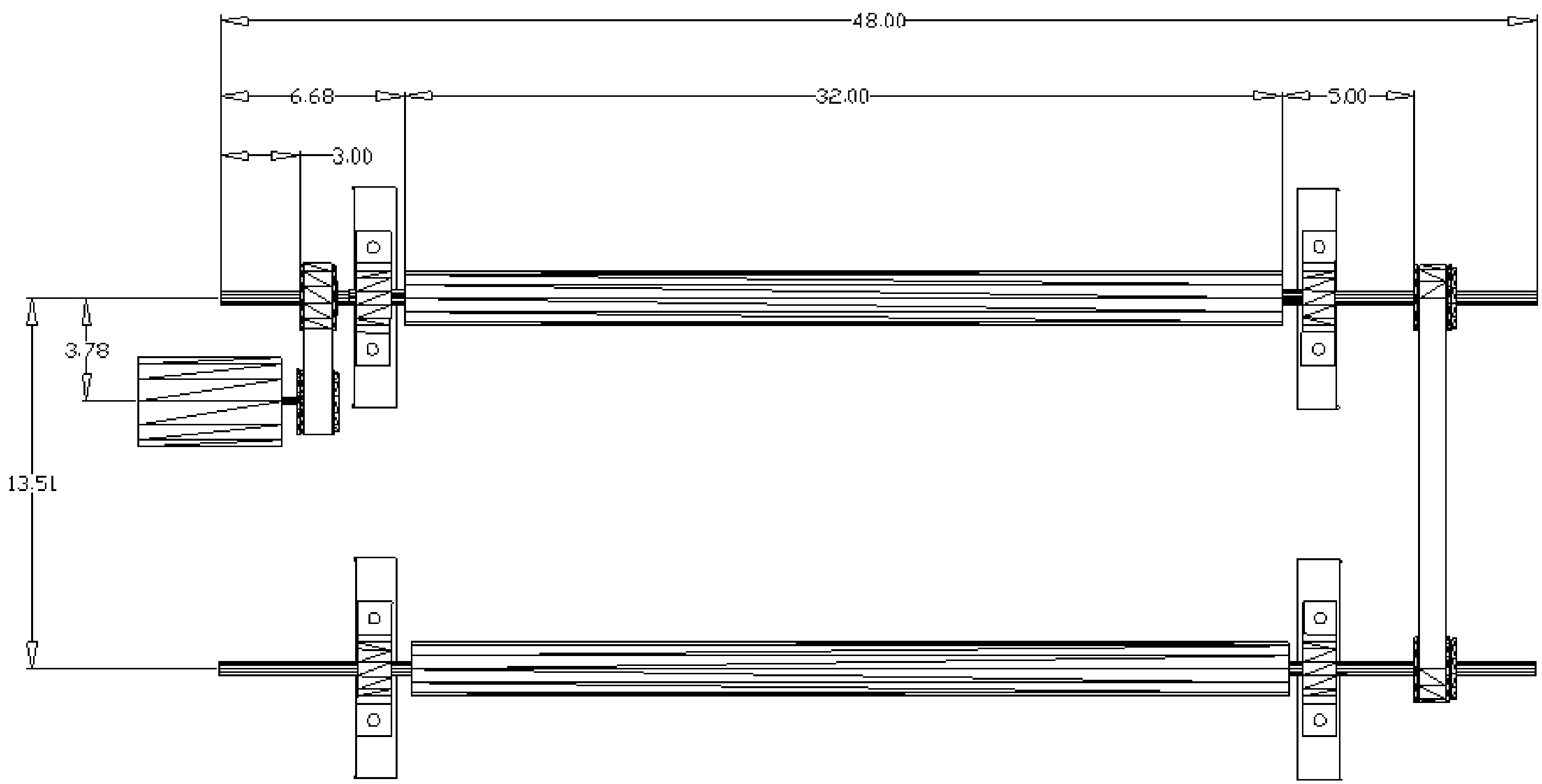


Figure B11: Side View of the Conveyor Drive Mechanism



Scale: 1” = 6.4”

Figure B12: Top View of the Conveyor Drive Mechanism

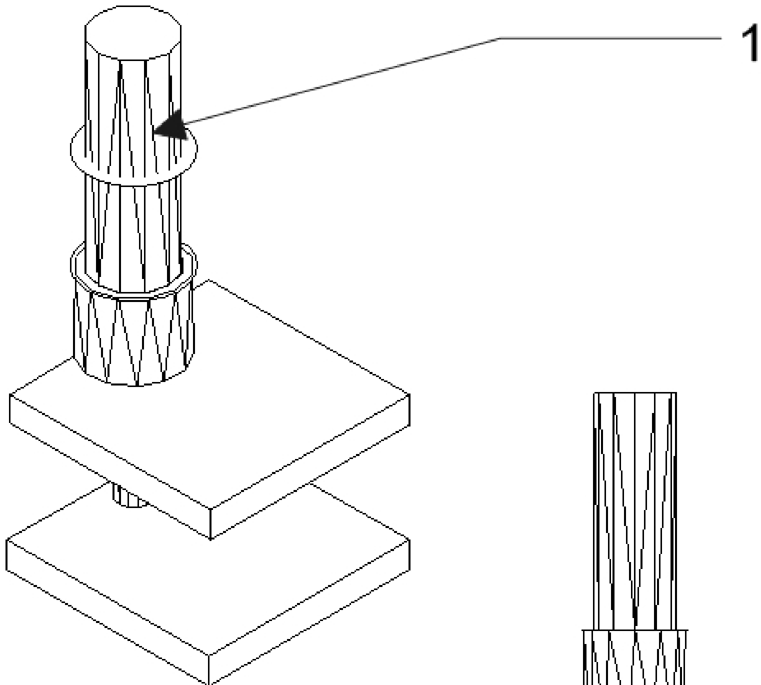
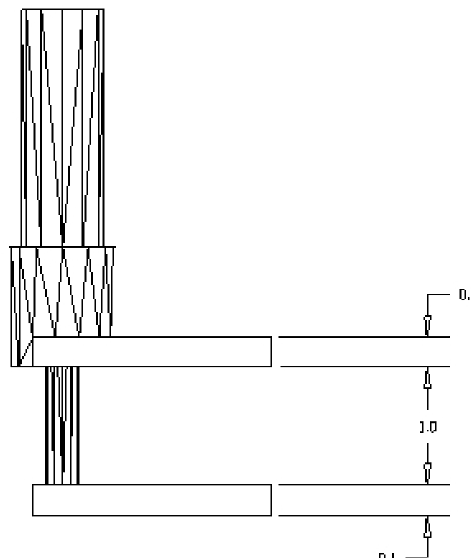
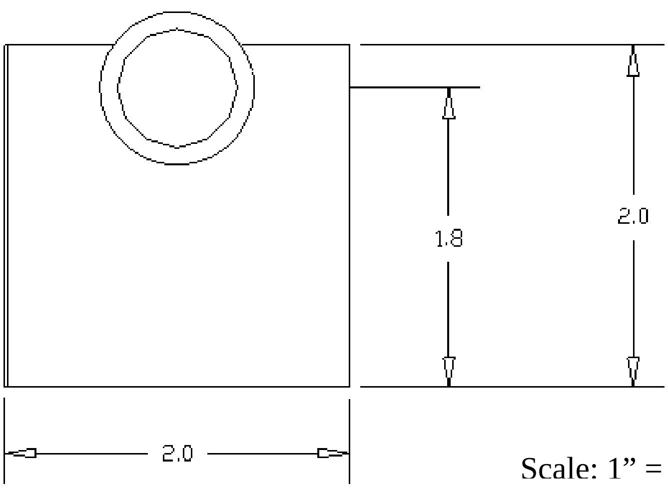


Figure B13: Orthographic View of Folding Mechanism – 1



Scale: 1” = 1.78”

Figure B14: Side View of Folding Mechanism – 1



Scale: 1” = 1.33”

Figure B15: Top View of Folding Mechanism – 1

Table B5: Reference for Figure B13

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1	Solenoid	Mc Master Carr	69905K85	1

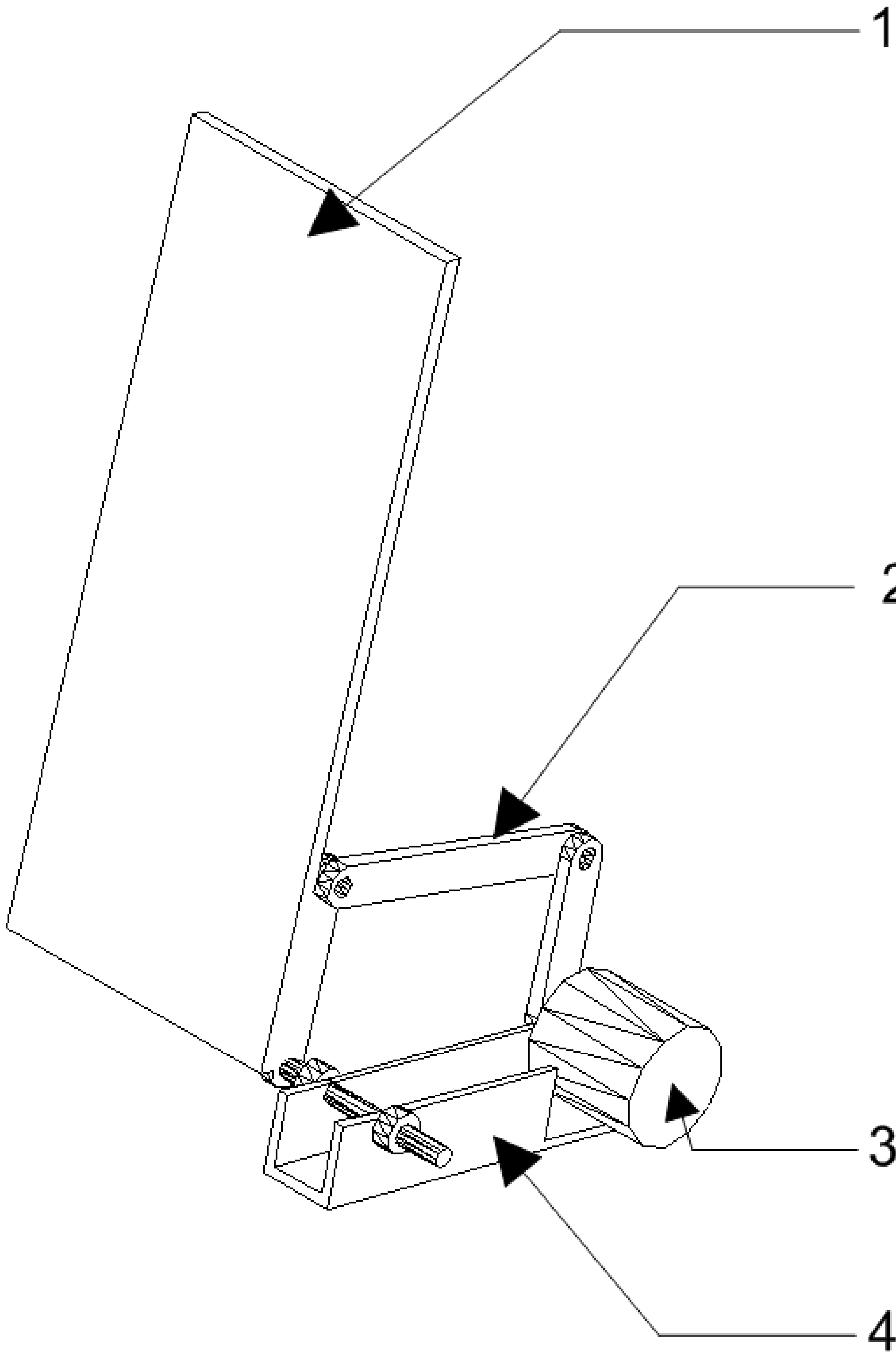
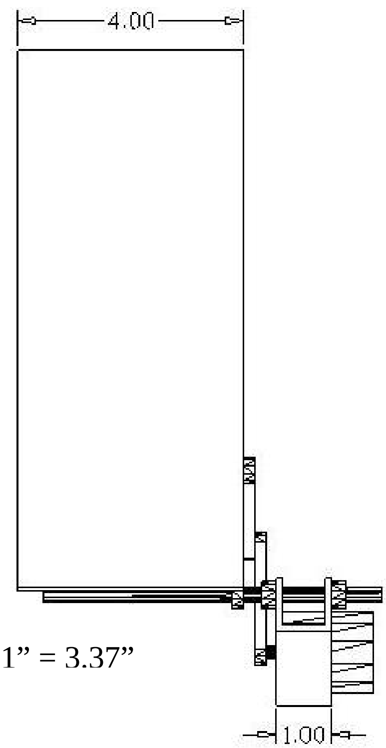


Figure B16: Orthographic View of Folding Mechanism – 2

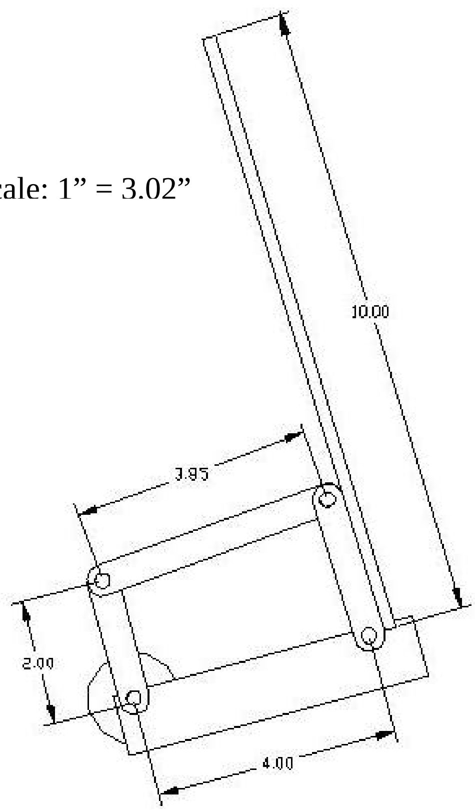
Table B6: Reference for Figure B16

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1	Folding Flap	-	-	1
2	Four Bar Mechanism	-	-	1
3	DC gearmotor	Jameco	HN-GH12-1926Y	1
4	Aluminum Frame	-	-	1



Scale: 1” = 3.37”

Figure B17: Front View of Folding Mechanism – 2



Scale: 1” = 3.02”

Figure B18: Side View of Folding Mechanism – 2

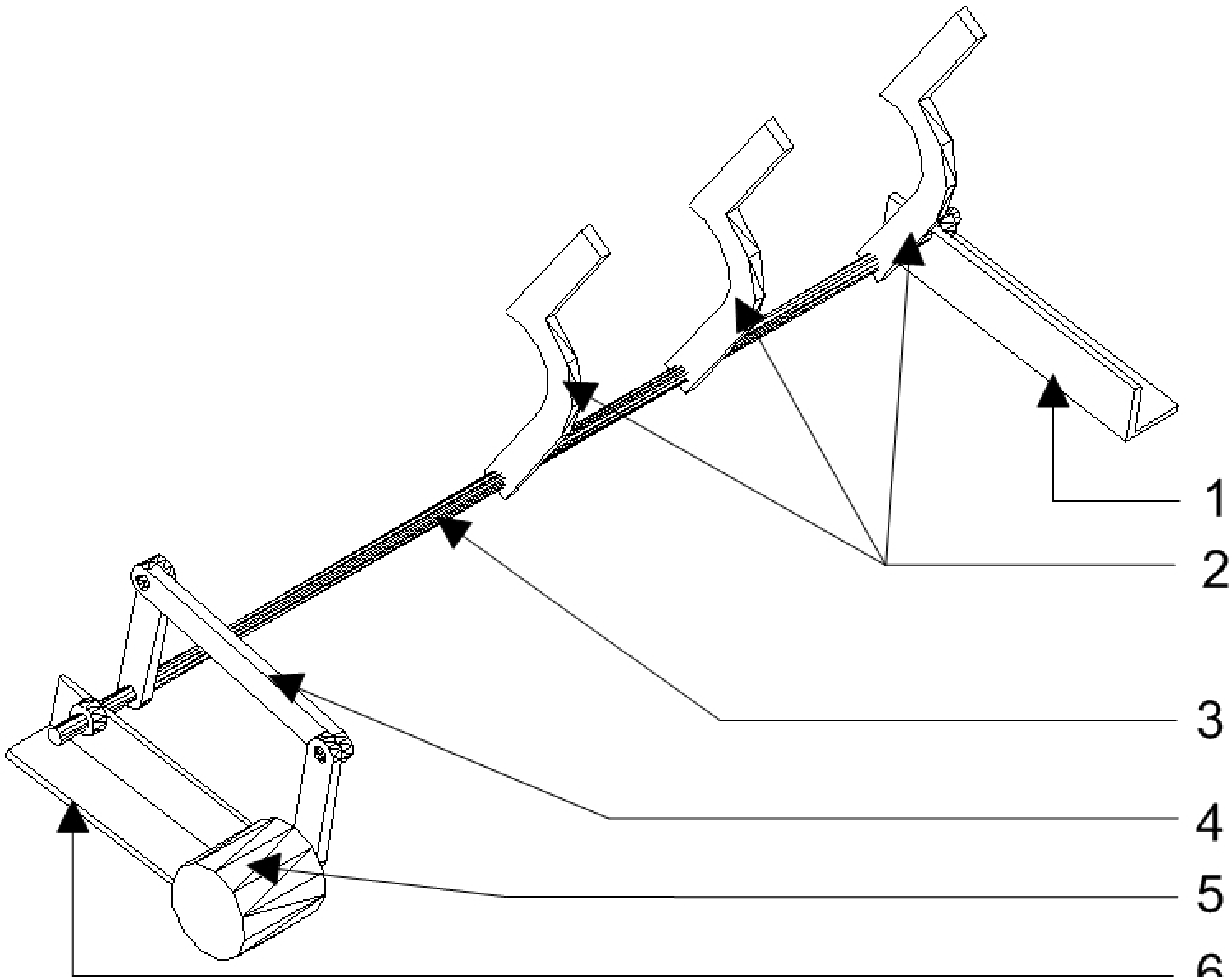
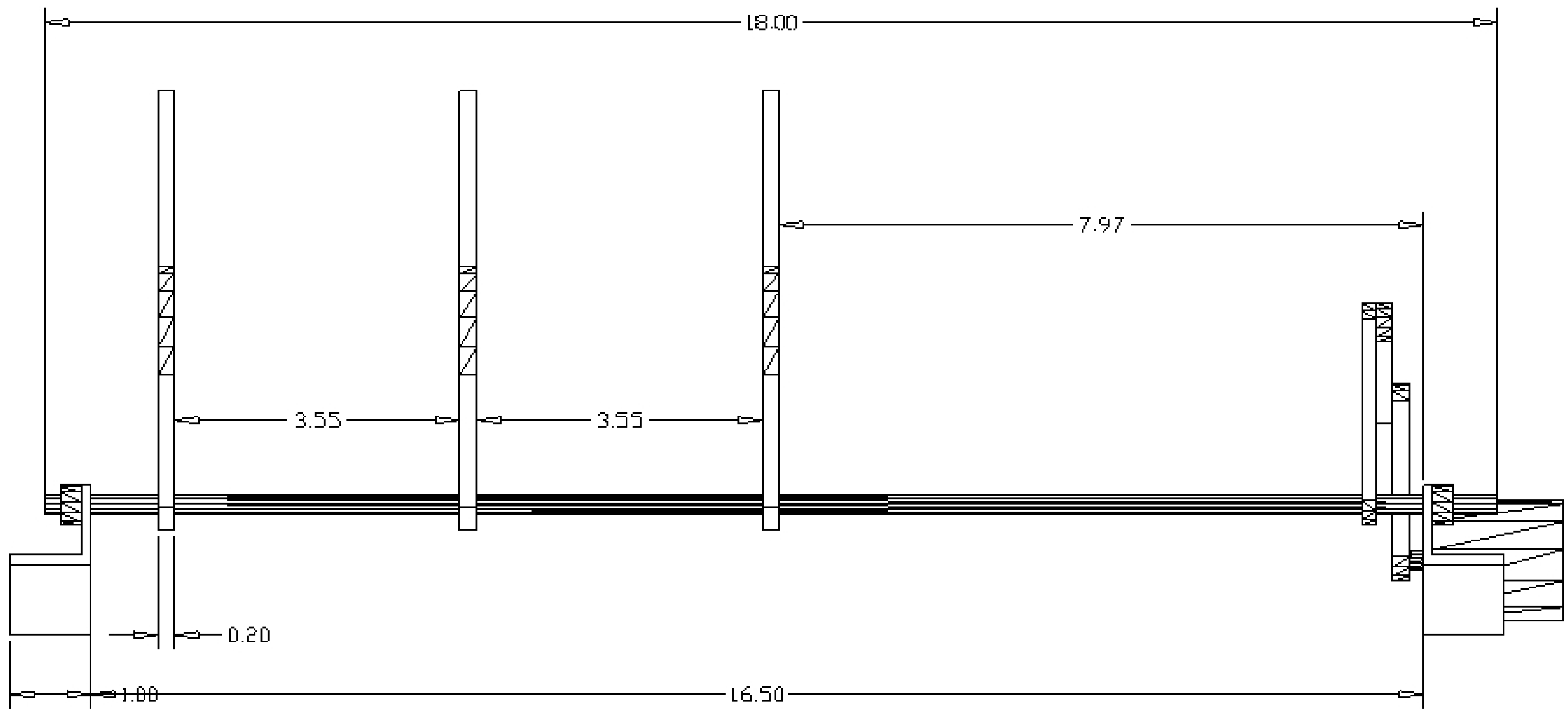


Figure B19: Orthographic View of Folding Mechanism - 3

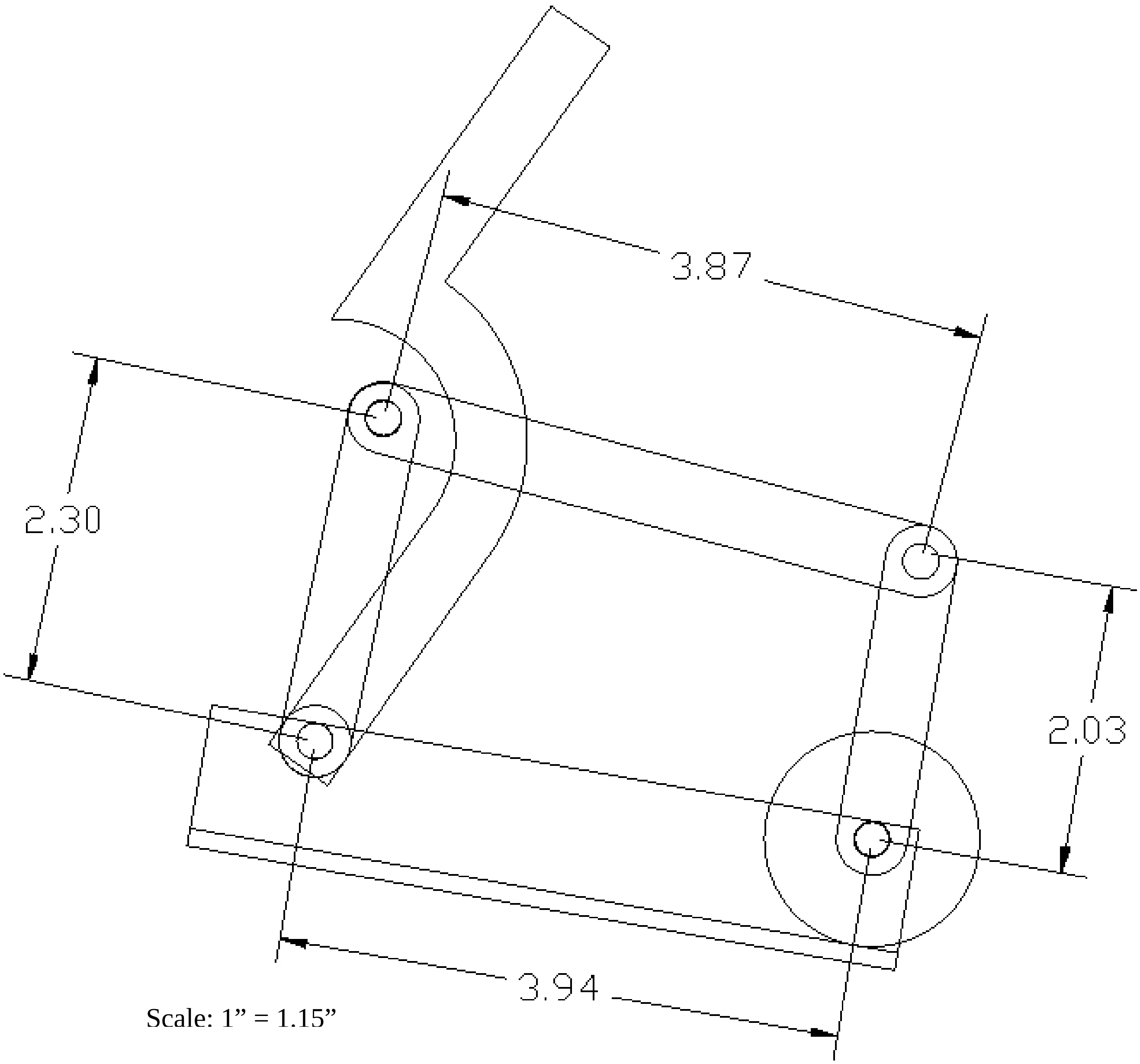
Table B7: Reference for Figure B19

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1&6	Support for Folding Mechanism-3	-	-	2
2	Folding Pads	-	-	3
3	Connecting Shaft	Mc Master Carr	6061K41	1
4	Four Bar Mechanism	Mc Master Carr	-	1
5	DC gearmotor	Jameco	HN-GH12-1926Y	1



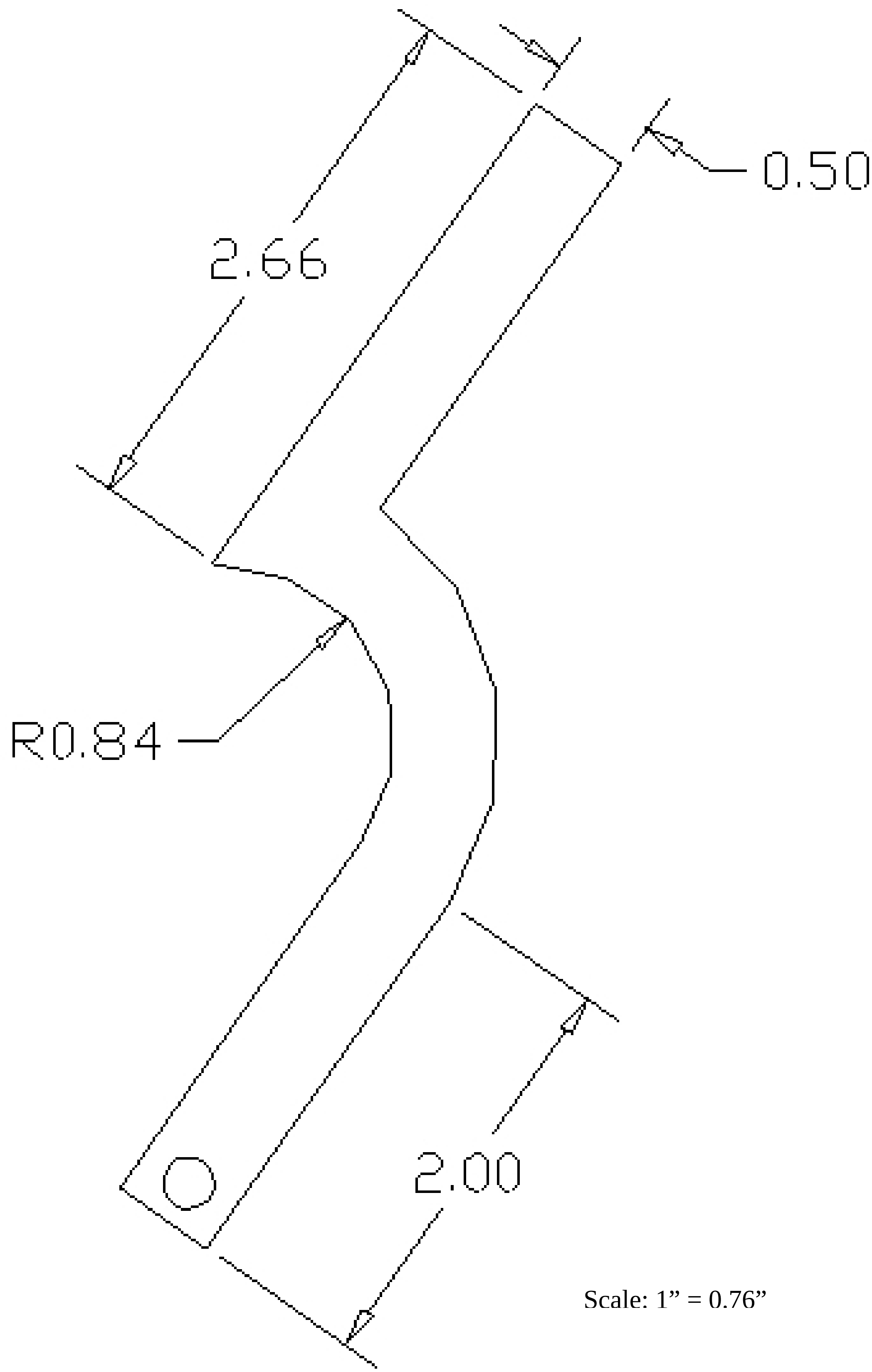
Scale: 1” = 2.17”

Figure B20: Top View of Folding Mechanism - 3



Scale: 1” = 1.15”

Figure B21: Side View of Folding Mechanism - 3



Scale: 1” = 0.76”

Figure B22: Side View of Folding Pad Used in Folding Mechanism - 3

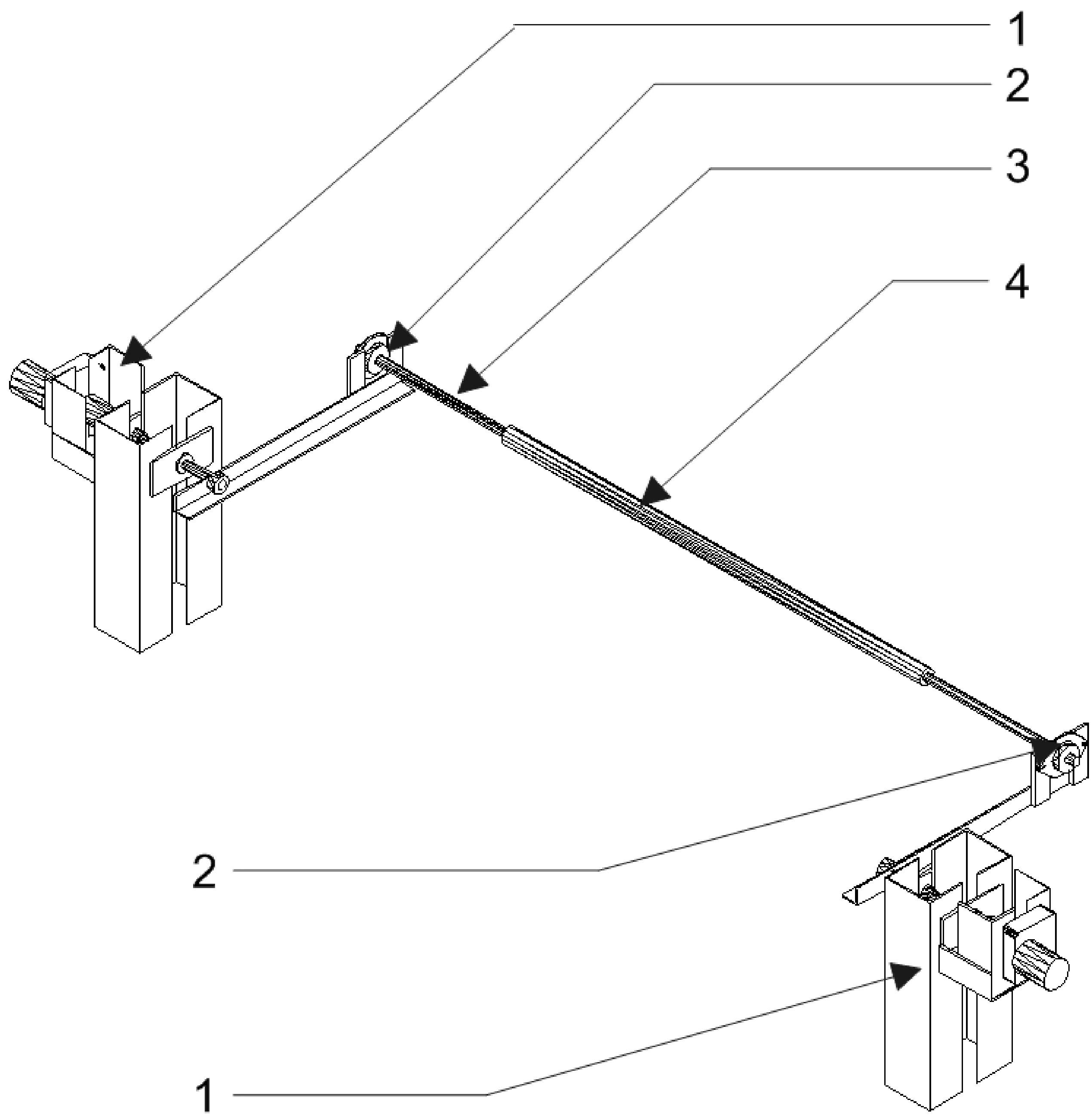
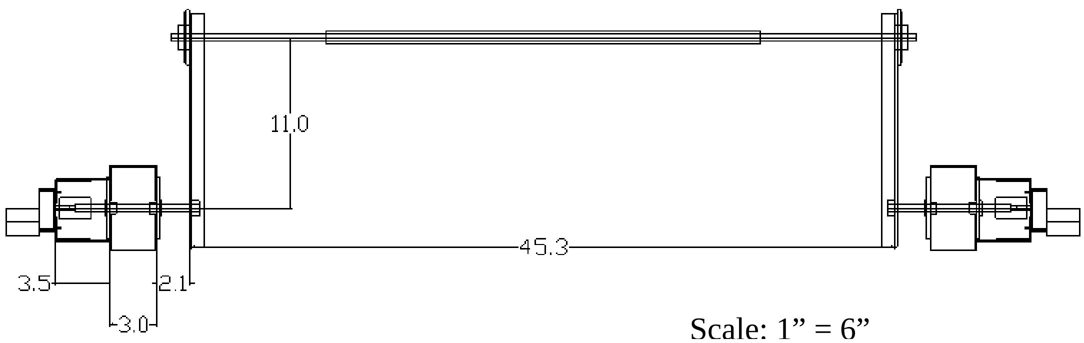


Figure B23: Orthographic View of Arm Mechanism used to pick Napkin from Napkin Bin

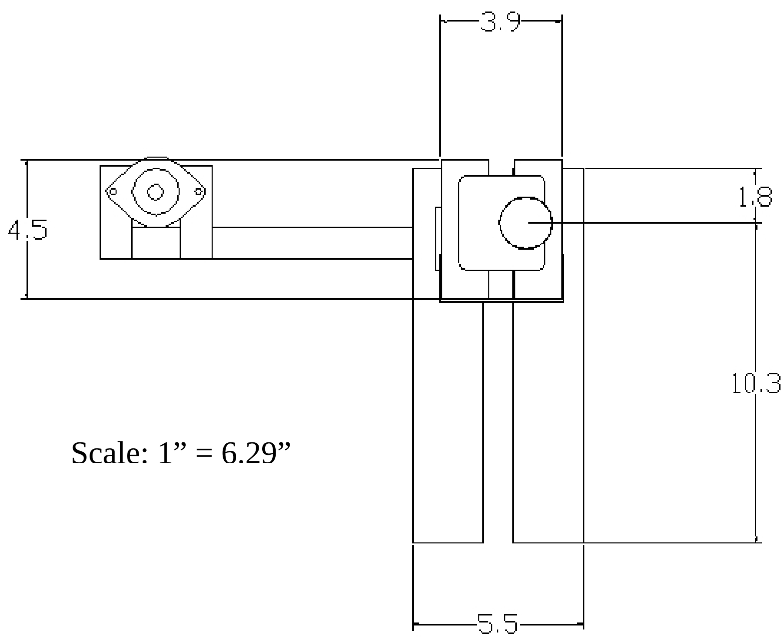
Table B8: Reference for Figure B23

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1	Drive System For Arm Mechanism	-	-	2
2	Bearing	Mc Master Carr	5913K51	2
3	Aluminum Shaft	-	-	1
4	Rubber Tube	Mc Master Carr	5235K54	28”



Scale: 1” = 6”

Figure B24: Top View of Arm Mechanism used to pick Napkin from Napkin Bin



Scale: 1” = 6.29”

Figure B25: Side View of Arm Mechanism used to pick Napkin from Napkin Bin

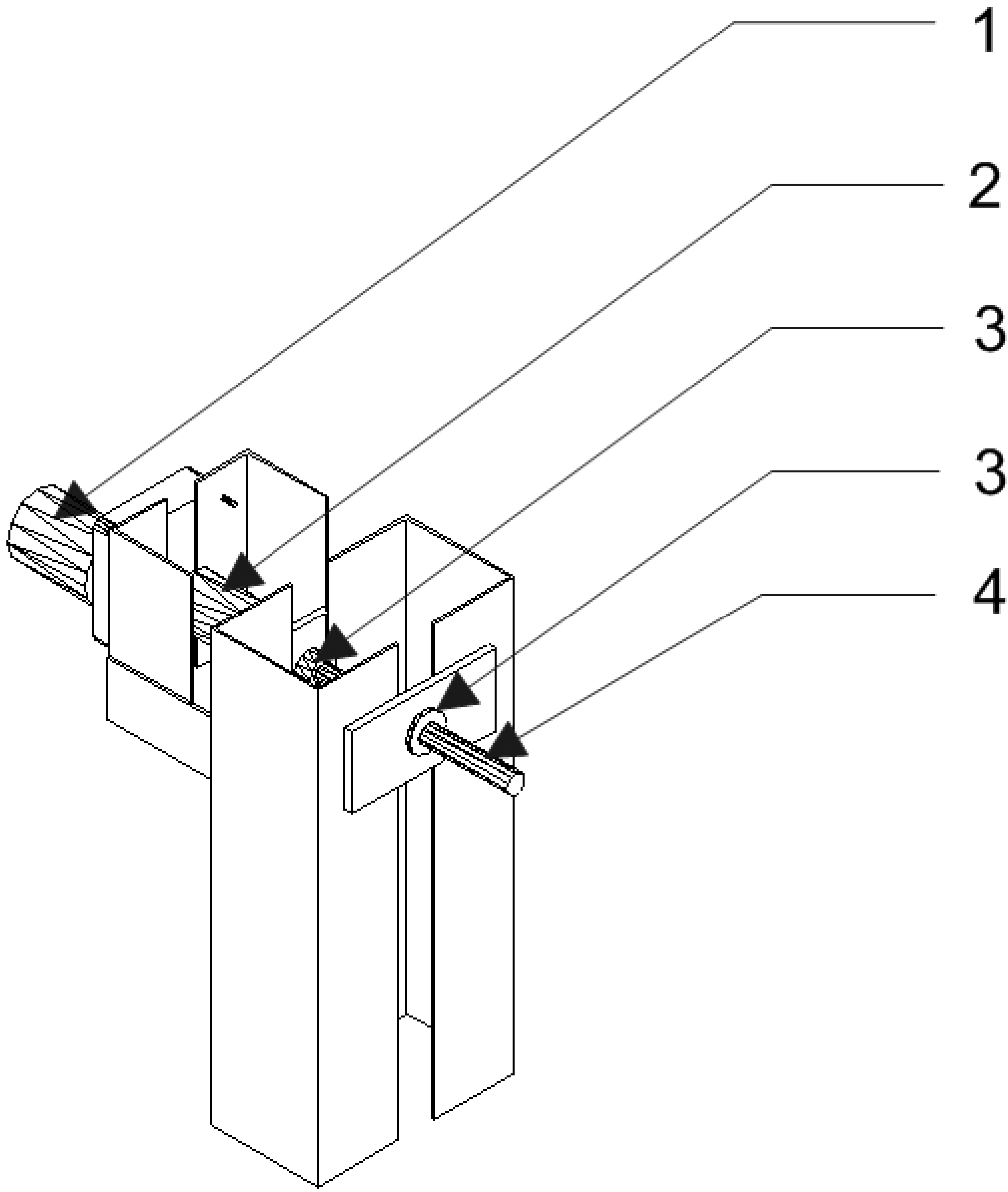


Figure B26: Orthographic View of Drive System to Drive Arm Mechanism used to pick Napkin from Napkin Bin

Table B9: Reference for Figure B26

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1	DC gearmotor	Mc Master Carr	6409K26	1
2	Shaft Coupler	Mc Master Carr	60845K55	1
3	Plain Shaft Bearing	Mc Master Carr	2938T16	2
4	Bearing Shaft	Mc Master Carr	6061K13	1

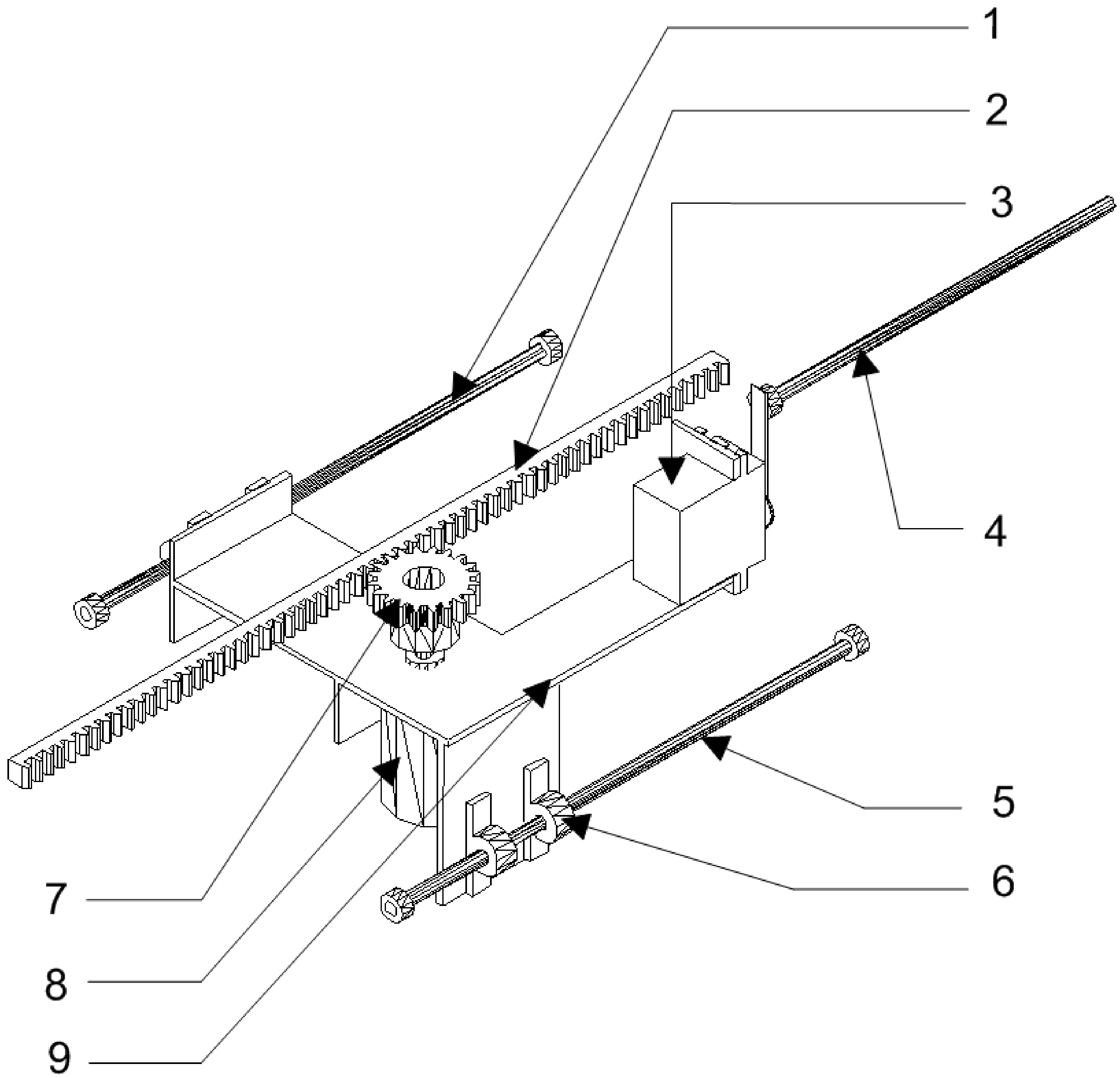
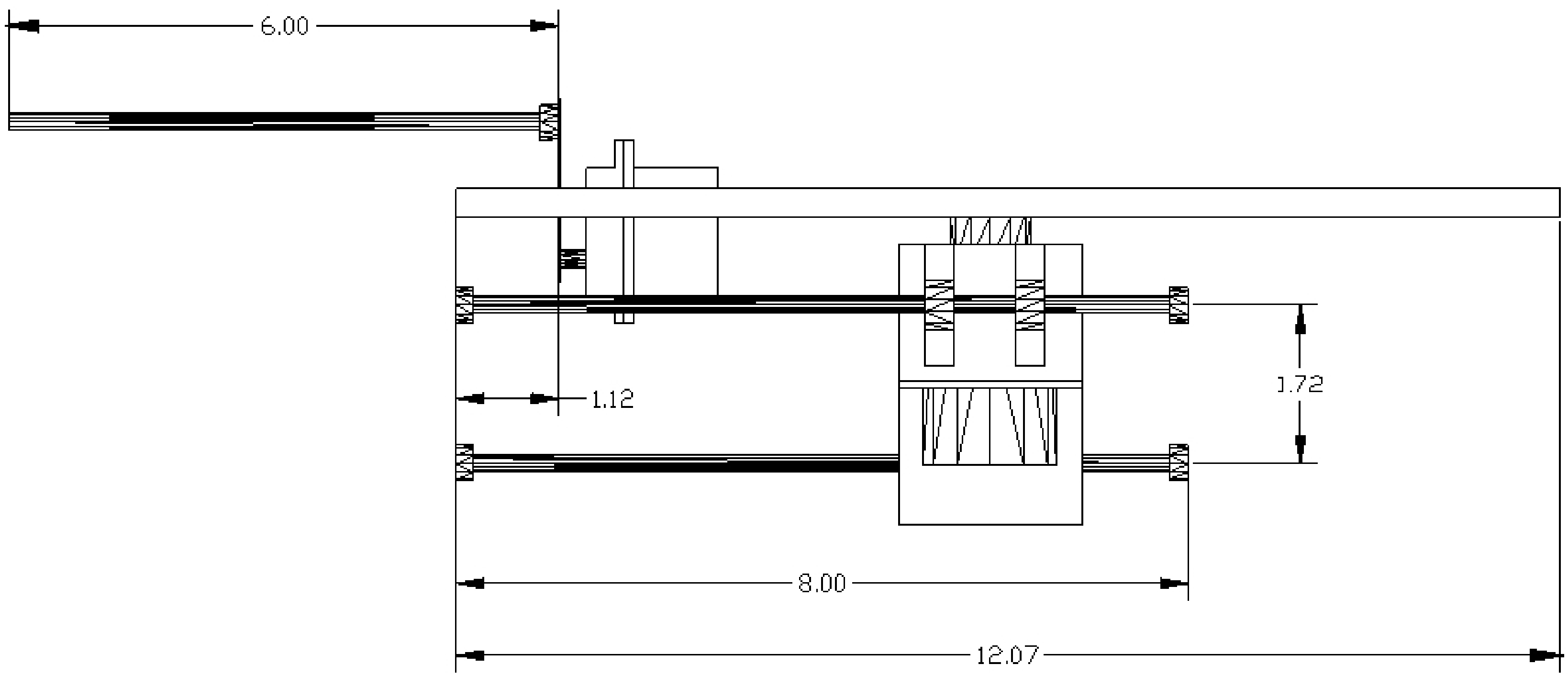


Figure B27: Orthographic View of Finger Mechanism used by Folding Mechanism - 3

Table B10: Reference for Figure B27

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1&5	Support Shaft	Mc Master Carr	6112K37	2
2	Rack	Mc Master Carr	57655K64	1
3	DC servomotor	Futaba	FP-S148	1
4	Shaft	Mc Master Carr	1162K32	1
6	Bearing	Mc Master Carr	6687K31	4
7	Spur Gear	Mc Master Carr	57655K56	1
8	DC gearmotor	Jameco	HN-GH12-1640Y	1
9	Support Carriage	-	-	1



Scale: 1” = 1.91”

Figure B28: Side View of Finger Mechanism used by Folding Mechanism - 3

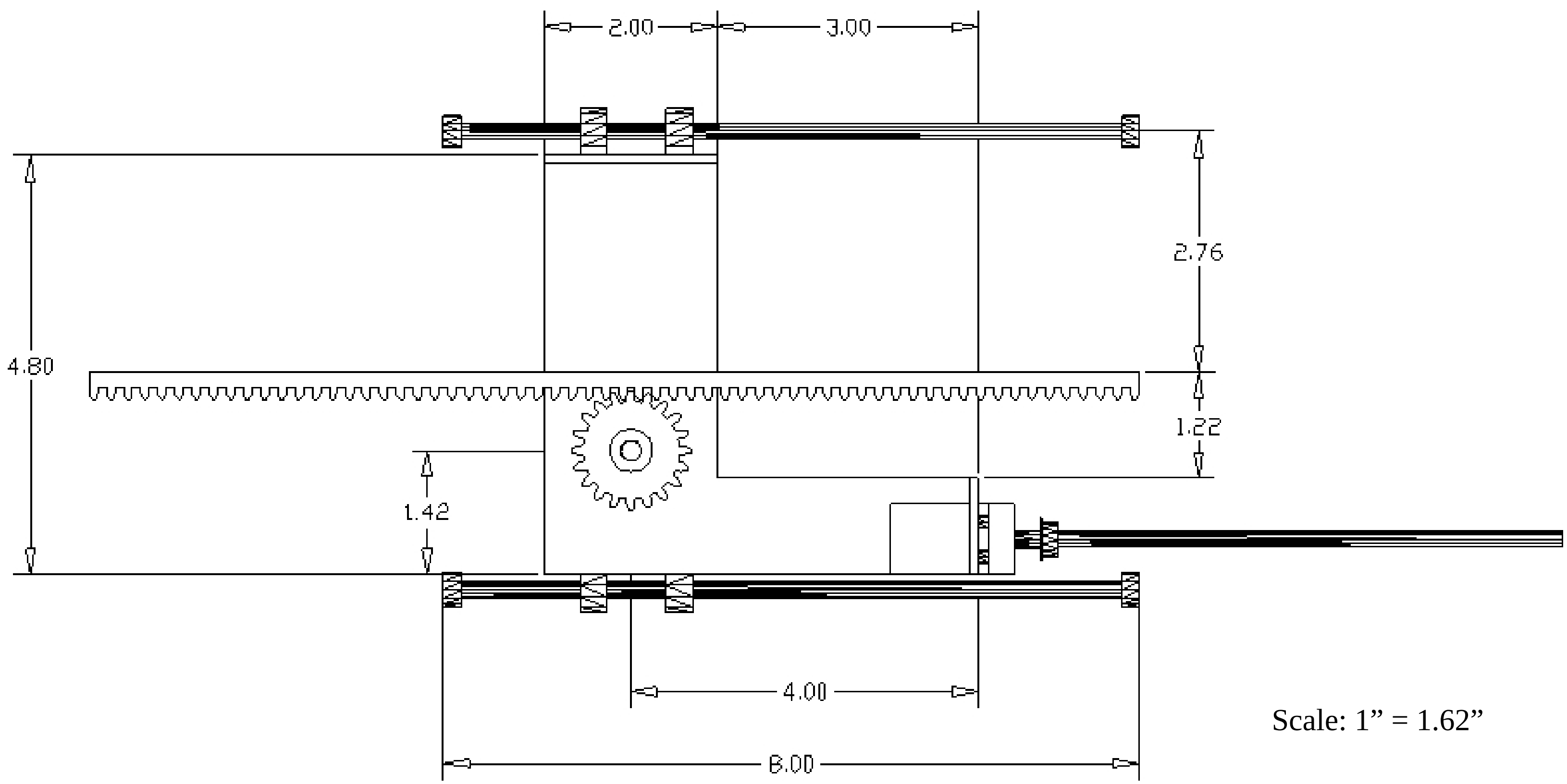


Figure B29: Top View of Finger Mechanism used by Folding Mechanism - 3

Stainless Steel Sheets of thickness 0.1”
are used to fabricate the profiles

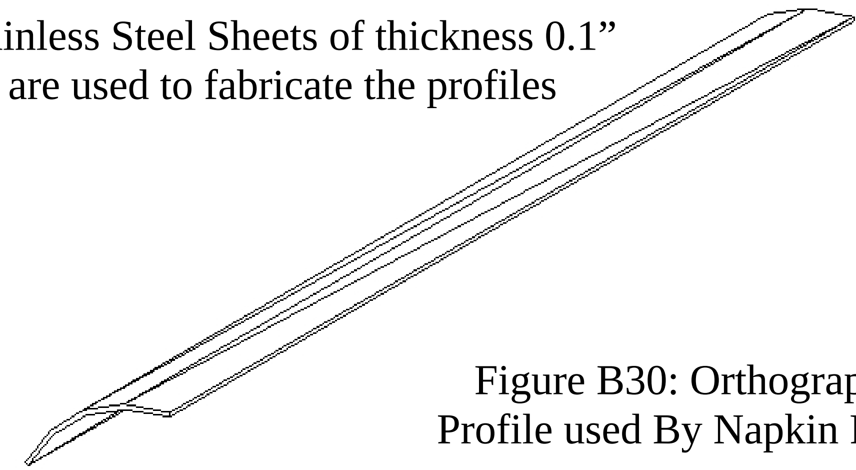
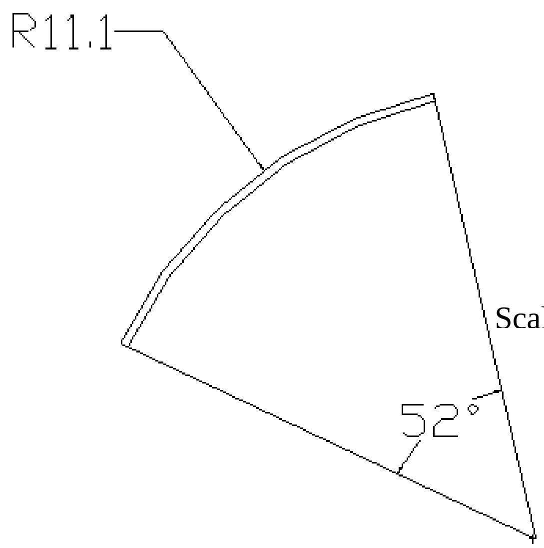
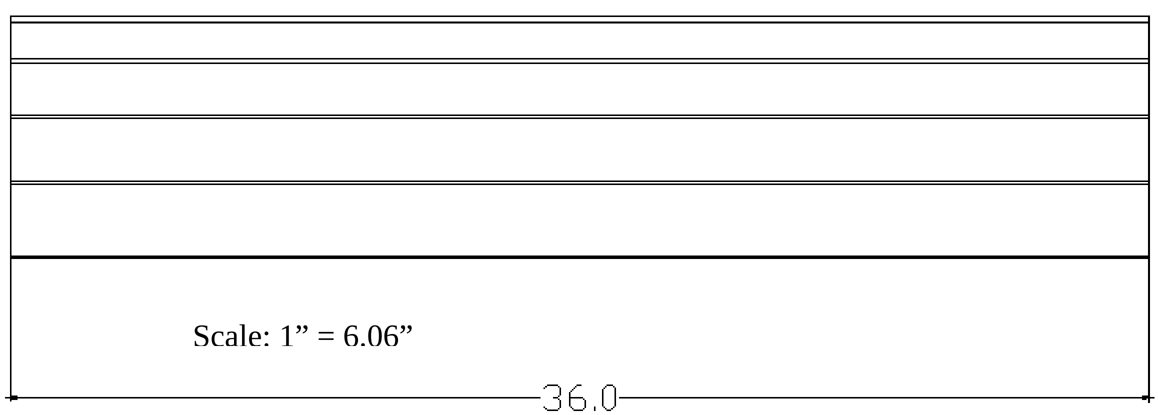


Figure B30: Orthographic View of Profile used By Napkin Pickup System



Scale: 1” = 4.67”

Figure B31: Side View of Profile used By Napkin Pickup System



Scale: 1” = 6.06”

Figure B32: Top View of Profile used By Napkin Pickup System

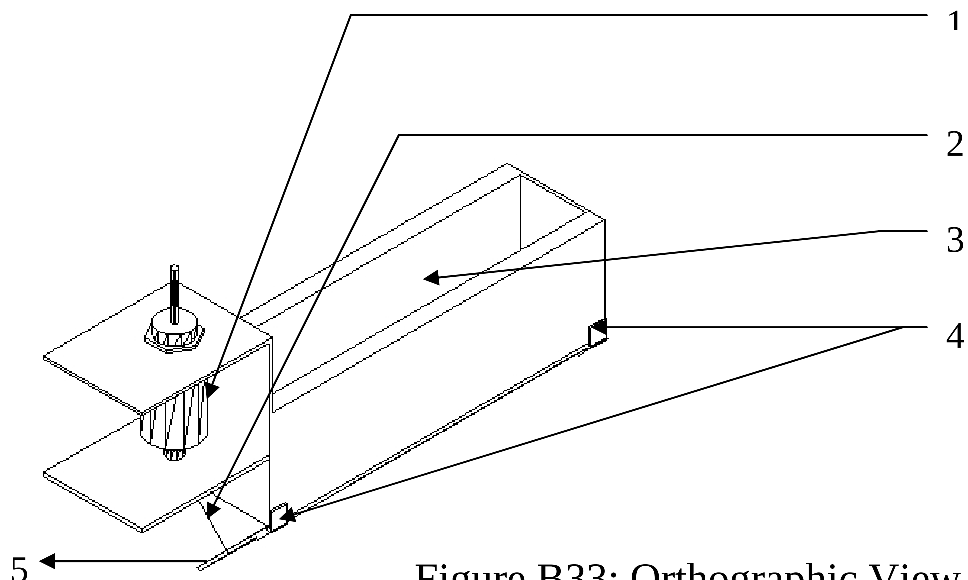
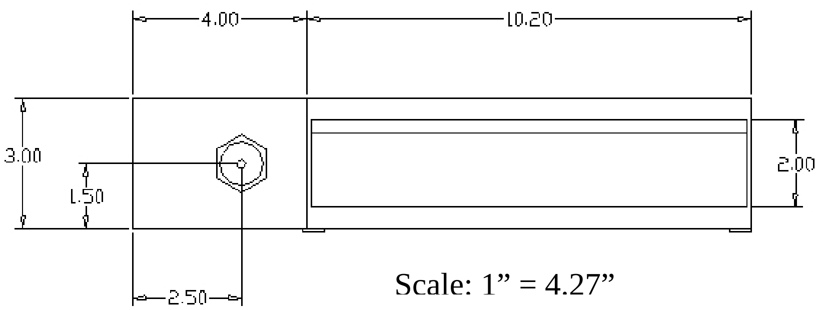
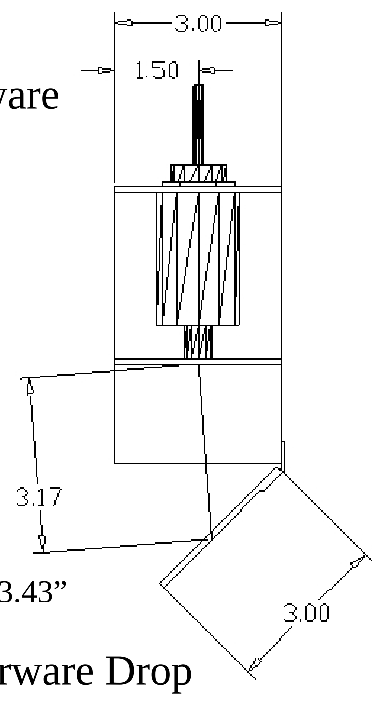


Figure B33: Orthographic View of Silverware Drop Mechanism



Scale: 1” = 4.27”

Figure B34: Top View of Profile Silverware Drop Mechanism



Scale: 1” = 3.43”

Figure B35: Side View of Silverware Drop Mechanism

Table B11: Reference for Figure B33

Label Number	Description	Manufacturer or Supplier	Manufacturer or Supplier Part Number	Quantity Used
1	Pull Solenoid	Mc Master Carr	69905K48	1
2	Flexible Line	-	-	1
3	Opening to Drop Silverware	-	-	-
4	Hinges	-	-	2
5	Solenoid Operated Hinged Bottom	-	-	1

APPENDIX C

Circuit Diagrams of Various microcontrollers

Circuit diagrams for microcontrollers 1, 2, and 3 are given in this section. The circuit diagram is given in Fig. C1, with legend as follows:

D1, D2, D3, D4, D5, D6 – Stepper motor driver module Kit-158 from kitsrus.com.
D7 – H-bridge driver unit, NMIH-0050 from newmicros.com
L1 – 12VDC pull solenoid rated at 10W continuous duty; McMaster part number: 69905K48.
L2 – Power for Folding Mechanisms-2 driver unit.
L3 – No Load Connected (For future upgrades)
L4 – Power for Folding Mechanism-3 driver unit.
L5 – 2 x 24VDC 1.08A dcgearmotor used to drive the rotary arm used in the bin mechanism.
L6 – 3 x E43H4C-12-8070117 stepper motor from HSI motors.
L7 – No Load Connected (For future upgrades)
L8 – No Load Connected (For future upgrades)
L9 – No Load Connected (For future upgrades)
L10, L11, L12 – No Load (For future upgrades)
L13 – 12VDC Solenoid rated at 11W continuous duty; McMaster part number: 70155K5
L14 – 15VDC Solenoid rated at 15W continuous duty; Magnetic Sensor system series S-20-125-H.
L15 – 8618L-02E-01 Stepper motor from Lin Engineering, used to driver conveyor system.
L16 – HT23-397 Stepper motor from Applied Motion Products used to drive Roller-1.

ZTT 20 MX – Chip Crystal used as the external clock for the microcontroller.
IPS 511 – High side switch.
DRV 101 – PWM/Solenoid driver.

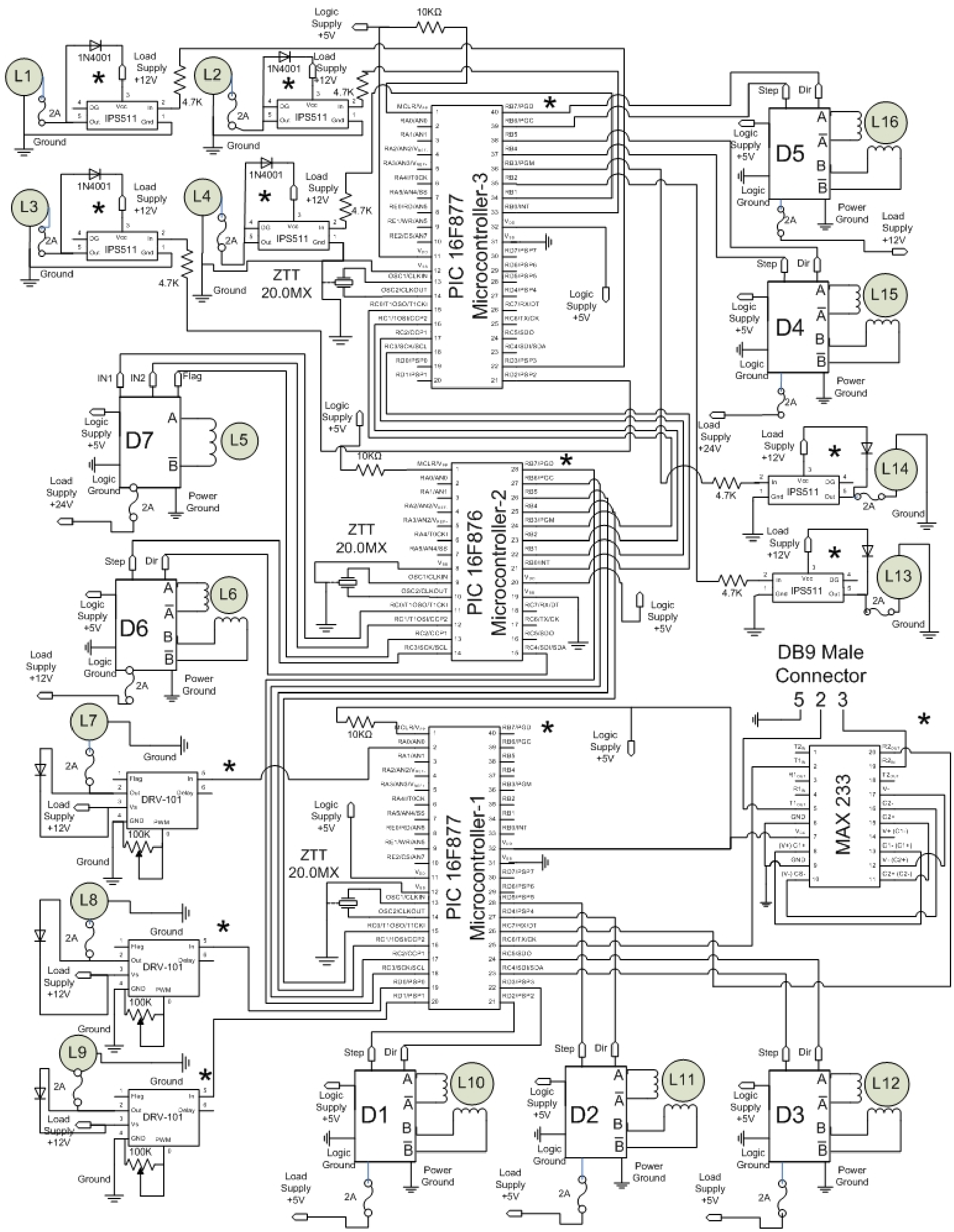


FIGURE C1
Circuit Diagram for Microcontrollers 1, 2, and 3.
* Pin labels as given by manufacturers.

Figure C2 gives the circuit diagram for the feedback lines connected from Microcontroller-1, Microcontroller-2 and Microconntroller-3 to the DB-25 breakout board from Winford Engineering. The functions of the 25 feedback lines accommodated in the system as labeled in the figure are as follows.

1 – Ground
2 – Logic Supply +5VDC
3 – Ground
4 – No Connection
5 – No Connection
6 – No Connection
7 – No Connection
8 – No Connection
9 – No Connection
10 – No Connection
11 – No Connection
12 – IR sensor to indicate if napkin bin is empty
13 – No Connection
14 – Trip switch to indicate if bin tray has reached the bottom
15 – Trip switch to indicate if rotateable arm has reached lower position
16 – Trip switch to indicate if rotateable arm has reached napkin dispensing position
17 – IR sensor to detect if napkin is outside the napkin tray
18 – IR sensor to detect if napkin the at the top of the napkin stack is in contact with Roller - 1
19 – No Connection
20 – IR sensor to detect if napkin is within range of Folding Mechanism-1.
21 – No Connection
22 – No Connection
23 – No Connection
24 – PIN C0 from Microcontroller – 4
25 – PIN C1 from Microcontroller – 4

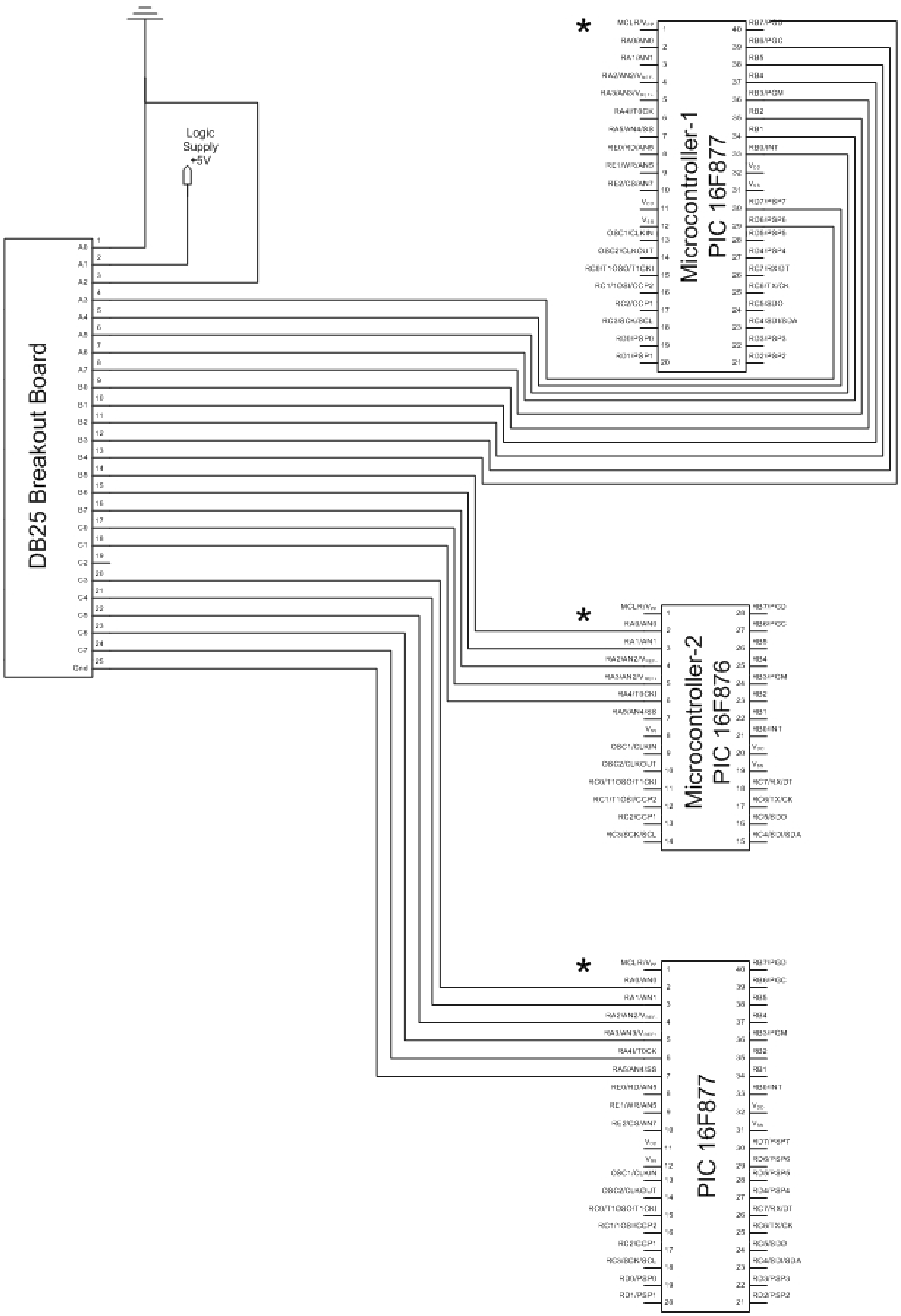


FIGURE C2
Circuit Diagram for Feedback Lines
* Pin labels as given by manufacturer of microcontrollers

Figure C3 gives the circuit diagram for microcontroller-4 used to control Folding Mechanisms-2 and Folding Mechanism-3. The load and feedback details, labeled in the figure are as follows.

L1 – 12VDC gearmotor rated at 100 RPM @ 60 mA to drive Folding mechanism -3.
L2 – No Load Connected.
L3 – 12VDC gearmotor rated at 100 RPM @ 60 mA to control the left flap in Folding Mechanism -2.
L4 – 12VDC gearmotor rated at 100 RPM @ 60 mA to control the right flap in Folding Mechanism - 2.
F1 – IR feedback used to control L4.
F2 – IR feedback used to detect napkin within Folding Mechanism – 3.
F3 – IR feedback used to detect napkin within right folding pad in Folding Mechanism–2.
F4 – IR feedback used to control L1.
F5 – IR feedback used to detect napkin within left folding pad in Folding Mechanism – 3.
F6 – IR feedback used to control L3.
F7 – Feedback to Microcontroller – 3.
F8 – Feedback to Microcontroller – 4.
F9 – Spare Feedback Line.
F10 – Spare Feedback Line.
ZTT 20 MX – Chip Crystal used as the external clock for the microcontroller.
L298N – Dual H-bride driver.

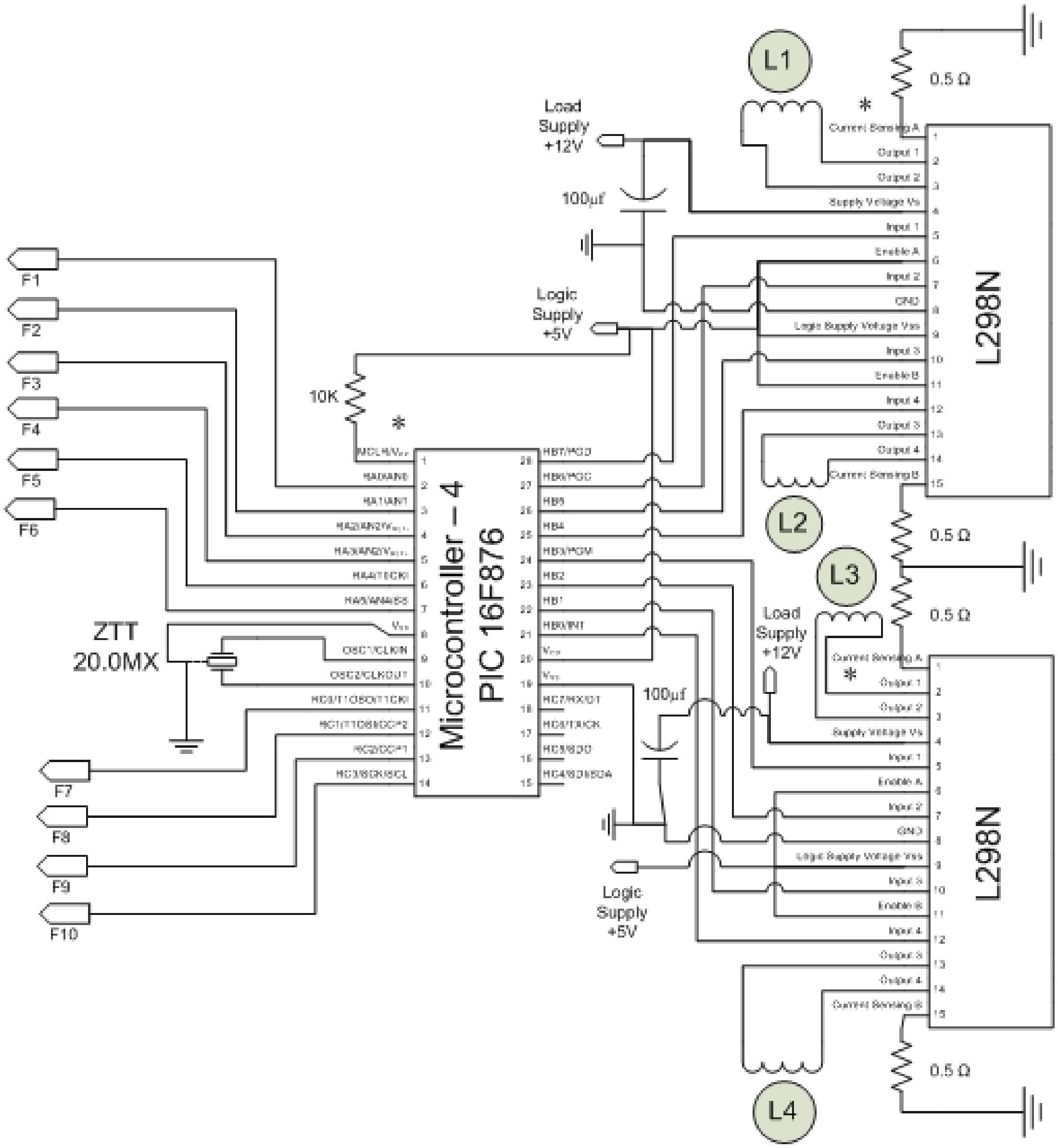


FIGURE C3
Circuit Diagram – Microcontroller 4
* Pin labels as given by corresponding manufacturers

Figure C4 gives the circuit diagram for the feedback control required to ensure that the bin motors run synchronously. The encoder details used are as follows.

E1, E2, E3 – RM – 21 encoder model from RENCO whose resolution is 1000 CPR.
MC 3486P – Line driver
CD 4516 – Counter
LS 7084 – Encoder to counter interface

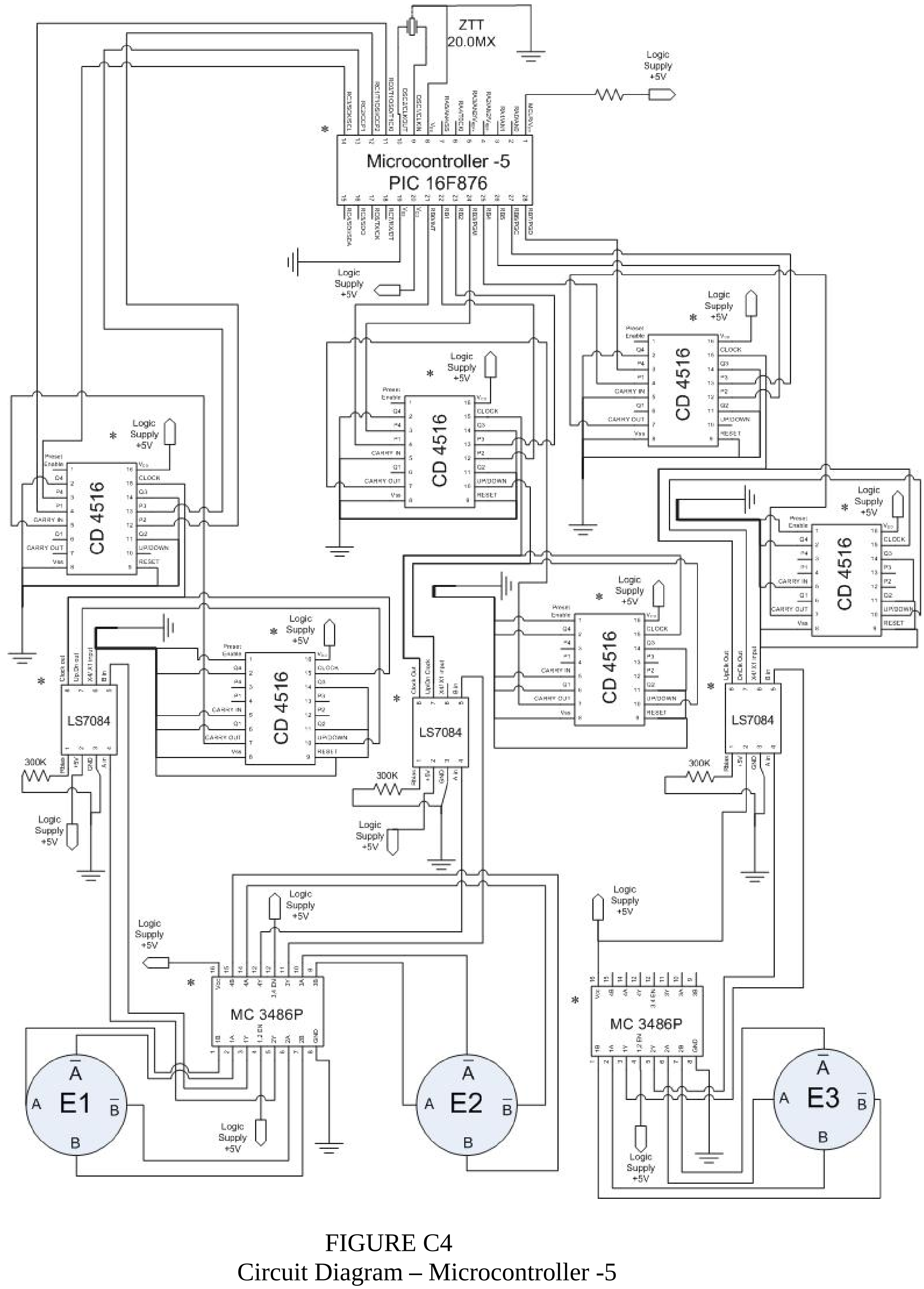


FIGURE C4
Circuit Diagram – Microcontroller -5
* Pin labels as given by corresponding manufacturers

Fig C5 is the circuit diagram used to generate the 38KHz pulse required to drive the infrared LED's used in the feedback system. The output from the microcontroller is fed to a quad H-bridge driver L293. The quad H-bridge unit is used to ensure that enough power is delivered by the system to all the IR LED's in the system.

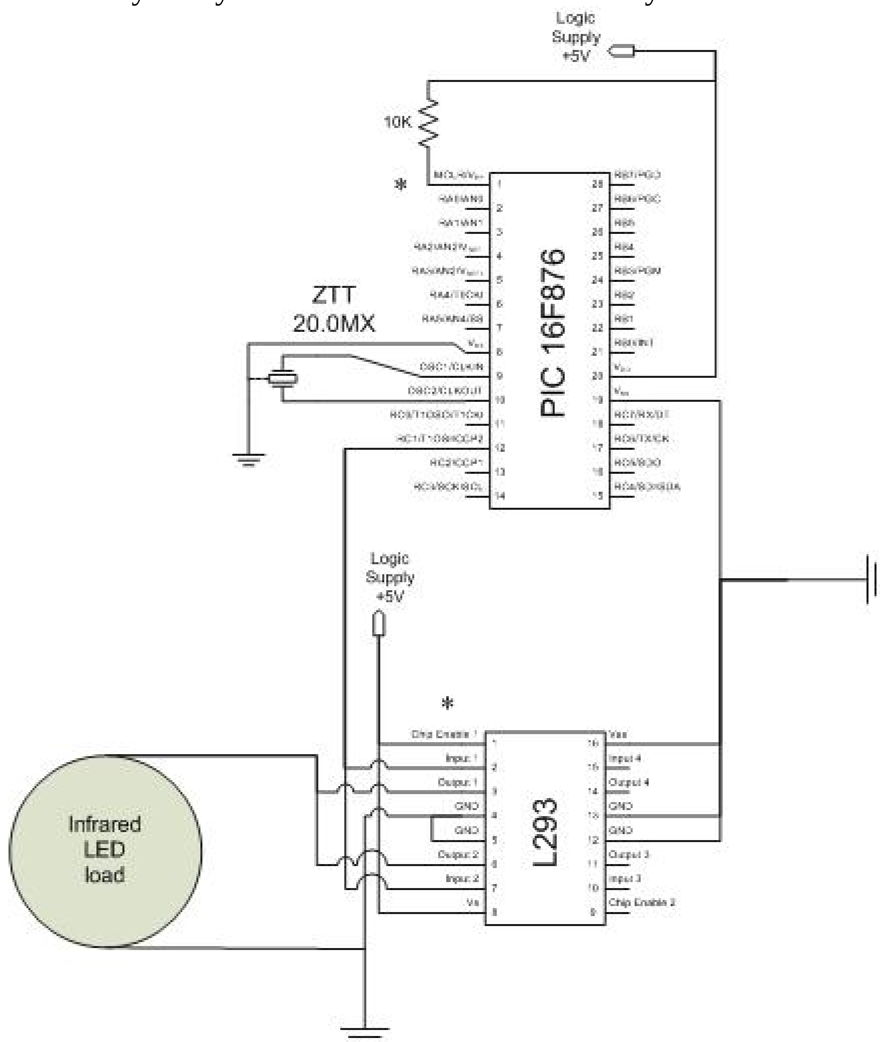


FIGURE C5
Circuit Diagram – Pulse Generator for IR LED's
* Pin labels as given by corresponding manufacturers

APPENDIX D

Four Bar Mechanism Analysis

A Four-Bar Mechanism, as described by its name consists of four links attached as shown in Fig D1.

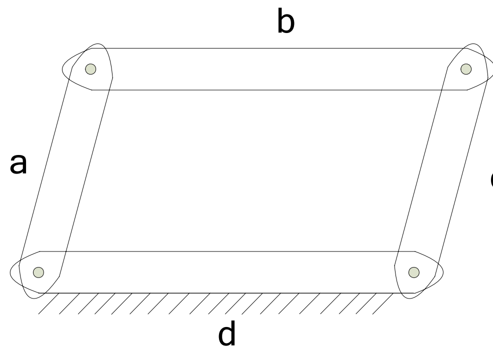


Figure D1: Four-Bar Mechanism

When the members a & c, b & d are of equal lengths, the motion from members 'a' and 'c' are rotary. But when member 'c' is longer than that of member 'a' and when member 'b' is not necessarily member 'd', and member 'd' is fixed, the mechanism becomes a crank and rocker mechanism as shown in Figure D2. This mechanism is used to implement Folding Mechanisms-2 and Folding Mechanisms-3.

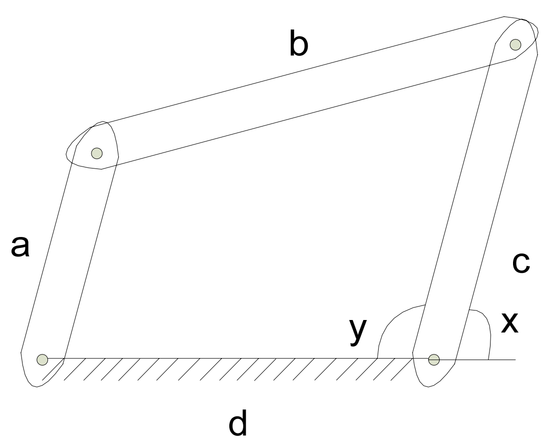


Figure D2: Four-Bar Rocker Mechanism

For a given value of length of member 'a' and member 'd', and a given value of minimum values of 'x' and 'y', we can determine the value of the lengths of members of 'b' and 'c' given by the equations.

$$b = \sqrt{\left((d+a)^2 + c^2 - 2(a+d)c\cos(x)\right)}$$

$$c = \frac{4ad}{2a(\cos(x) + \cos(\phi - y)) - 2d(\cos(\phi - y) - \cos(x))}$$

Once the distance of all the four members have been determined, the four bar mechanism can be simulated in order to obtain its performance based on various parameters.

Figure D3 is used to determine the kinematics of the four bar rocker mechanism.

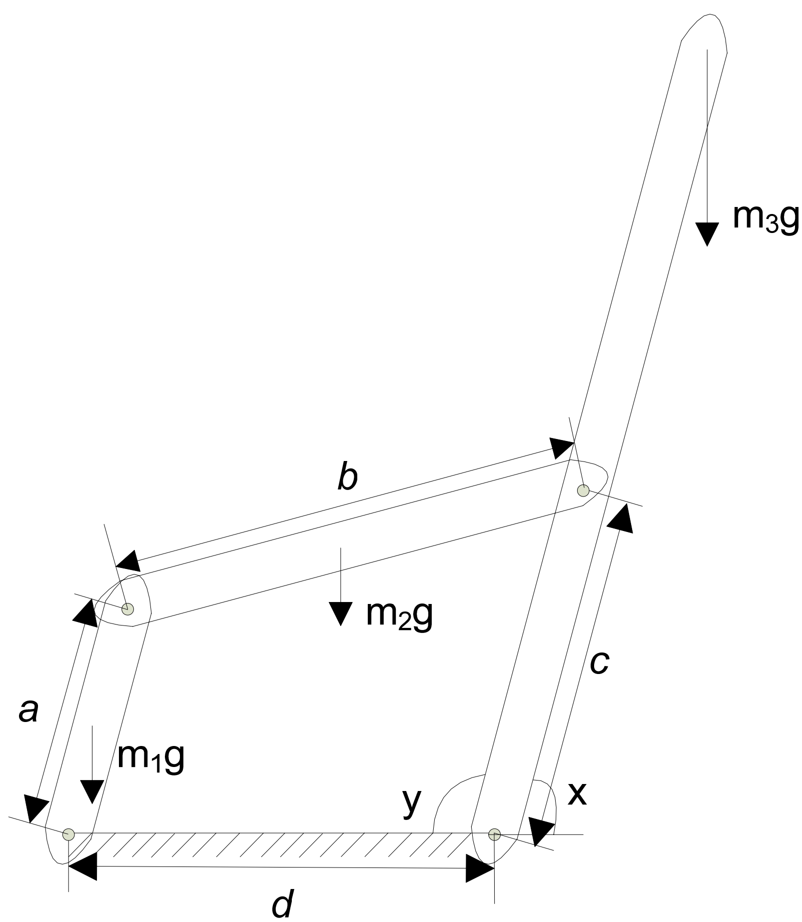


Figure D3: Four Bar Mechanism Used to Determine the Kinematics of the Four Bar Rocker Mechanism

The Free body diagrams and equations of the members of length ‘a’, ‘b’, ‘c’ are given in the following diagrams and equations.

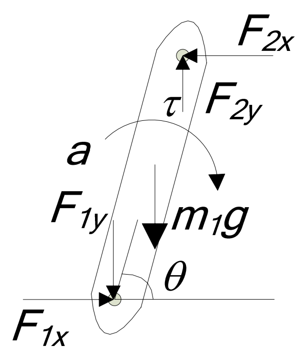


Figure D4: Free Body Diagram of the crank in the crank and rocker mechanism

The equations of motion from Figure D4 are given by:-

$$F_{1y} - F_{2y} + m_1 g = 0 \longrightarrow (1)$$

$$F_{1x} - F_{2x} = 0 \longrightarrow (2)$$

$$\tau - F_{2x} a \sin(\theta) - F_{2y} a \cos(\theta) = 0$$

Hence

$$\tau = F_{2x} a \sin(\theta) + F_{2y} a \cos(\theta) \longrightarrow (3)$$

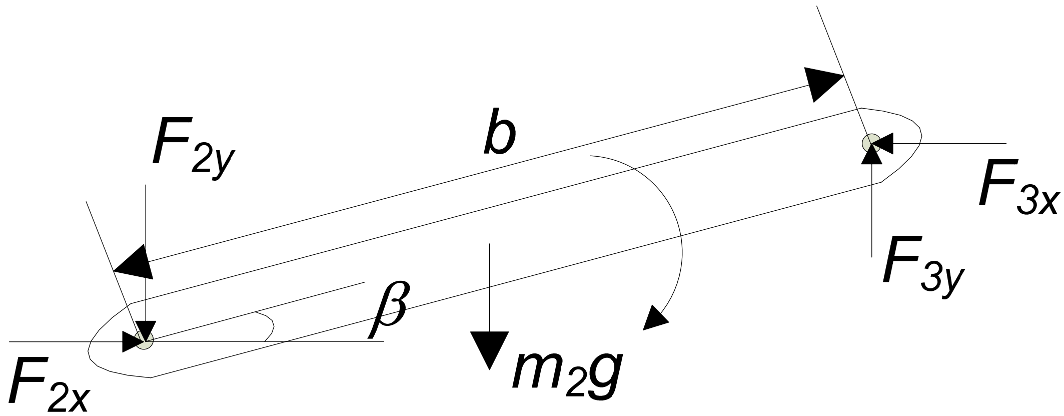


Figure D5: Free Body Diagram of the Connecting member in the Four Bar Rocker Mechanism

The Equations of motion for the member shown in Figure D5 are as follows:

$$F_{3y} - m_2 g - F_{2y} = 0 \longrightarrow (4)$$
$$F_{2x} - F_{3x} = 0 \longrightarrow (5)$$
$$J_2 \ddot{\beta} - F_{3x} b \sin(\beta) - F_{3y} b \cos(\beta) = 0 \longrightarrow (6)$$

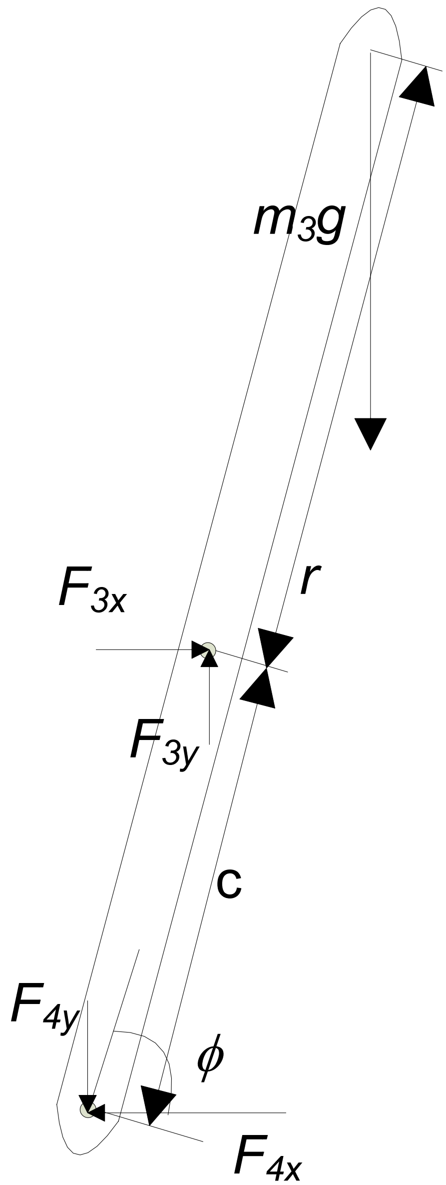


Figure D6: Free Body Diagram of Rocker Mechanism in the Four Bar Rocker Mechanism

The equations of motions based on Figure D6 are as follows:

$$F_{3y} + m_3 g - F_{4y} = 0 \longrightarrow (7)$$

$$F_{3x} - F_{4x} = 0 \longrightarrow (8)$$

$$J_3 \ddot{\Phi} - F_{3y} \cos(\phi) + F_{3x} c \sin(\phi) = 0 \longrightarrow (9)$$

From equations (2), (5) and (8),

$$F_{1x} = F_{2x} = F_{3x} = F_{4x} \longrightarrow (10)$$

From Equation (4)

$$F_{3y} = F_{2y} + m_2 g$$

From Equation(7)

$$F_{4y} = F_{3y} + m_3 g$$
$$or$$
$$F_{4y} = F_{3y} + (m_2 + m_3) g$$

Also From Figure D5:

$$J_2 \ddot{\beta} - (F_2 y + m_2 g) b\cos(\beta) - F_{2x} b\sin(\beta) = 0 \longrightarrow (11)$$

Also From Figure D6:

$$J_3 \ddot{\phi} - (F_2 y + m_2 g) c\cos(\phi) + F_{2x} c\sin(\phi) = 0 \longrightarrow (12)$$

From Equation (11)

$$F_{2y} b\cos(\beta) + F_{2x} b\sin(\beta) = J_2 \ddot{\beta} - m_2 bg \longrightarrow (13)$$

$$Let$$
$$c_1 = c\sin(\phi), c_2 = c\cos(\phi), c_3 = J_3 \ddot{\beta} - m_2 cg, c_4 = b\sin(\beta), c_5 = b\cos(\beta), c_6 = J_2 \ddot{\beta} - m_2 bg$$

Equations (12) and (13) can be rewritten as

$$c_6 = c_5 F_2 y + c_4 F_{2x} \longrightarrow (14)$$

$$c_3 = c_2 F_2 y - c_1 F_{2x} \longrightarrow (15)$$

From equations(14) and (15)

$$F_{2y} = \frac{c_3 c_4 + c_6 c_1}{c_2 c_4 + c_5 c_1} \longrightarrow (16)$$

$$F_{2x} = \frac{c_6 c_2 - c_3 c_5}{c_4 c_2 - c_1 c_5} \longrightarrow (17)$$

The different values subtended by the connecting bar and the rocker mechanisms are given by:

$$\phi = 2\tan^{-1}\left(\frac{-B - \sqrt{B^2 - 4AC}}{2A}\right) \longrightarrow (18)$$

$$\beta = 2\tan^{-1}\left(\frac{-E - \sqrt{E^2 - 4DF}}{2D}\right) \longrightarrow (19)$$

The angular velocities of the connecting bar and the rocker mechanisms are given by:

$$\omega_2 = \left(\frac{-a\omega_1 \sin(\phi - \theta)}{b\sin(\phi - \beta)}\right) \longrightarrow (20)$$

$$\omega_3 = \left(\frac{-a\omega_1 \sin(\beta - \theta)}{b\sin(\phi - \beta)}\right) \longrightarrow (21)$$

The angular acceleration of the connecting bar and the rocker mechanisms are given by:

$$\alpha_2 = (-a\alpha_1 \sin(\phi - \theta) + a\omega_1^2 \cos(\phi - \theta) + b\omega_2^2 \cos(\phi - \beta) - \frac{c\omega_3^2}{b\sin(\phi - \beta)} \longrightarrow (21)$$

$$\alpha_3 = (-a\alpha_1 \sin(\beta - \theta) + a\omega_1^2 \cos(\beta - \theta) + b\omega_2^2 - \frac{\omega_3^2 \cos(\phi - \beta)}{\sin(\phi - \beta)} \longrightarrow (22)$$

Where,

$$A = (1 - k_2)\cos(\theta) + k_3 - k_1$$
$$B = -2\sin(\theta)$$
$$C = k_1 + k_3 - (1 + k_2)\cos(\theta)$$
$$D = (k_4 + 1)\cos(\theta) + k_5 - k_1$$
$$E = -2\sin(\theta)$$
$$F = (k_4 - 1)\cos(\theta) + k_5 - k_1$$

Where,

Let

$$k_1 = \frac{d}{a}$$

$$k_2 = \frac{d}{c}$$

$$k_3 = a^2 - b^2 + c^2 - \frac{d^2}{2ac}$$

$$k_4 = \frac{d}{b}$$

$$k_5 = c^2 - a^2 - b^2 - \frac{d^2}{2ab}$$

Derivation of Equations (18) through (22) is described in “Theory of Machines” by R.S.Khurmi & J.K.Gupta(1999).

Based on the above stated Equations a matlab program is used to obtain the forces acting on the mechanism and consequently the input torque is derived. The Following plots are obtained.The following assumptions have been made in the program in order to obtain the graphs shown in figure D7:

- The Frictional Components of the system are neglected.
- The following values are assumed

  m_1 = 0.003kg, m_2 = 0.003kg, m_3 = 0.04kg, a = 5cm, d=10cm,r=15.3cm, x = 10º, y = 25 º(masses have been calculated based on rough estimates of dimensions of various members)
- The mass moment of inertias J_1, J_2, J_3 are given by

  $J_1 = m_1 a^2/3$, $J_2 = m_2 b^2/3$, $J3 = m^3(c+r)^2/3$

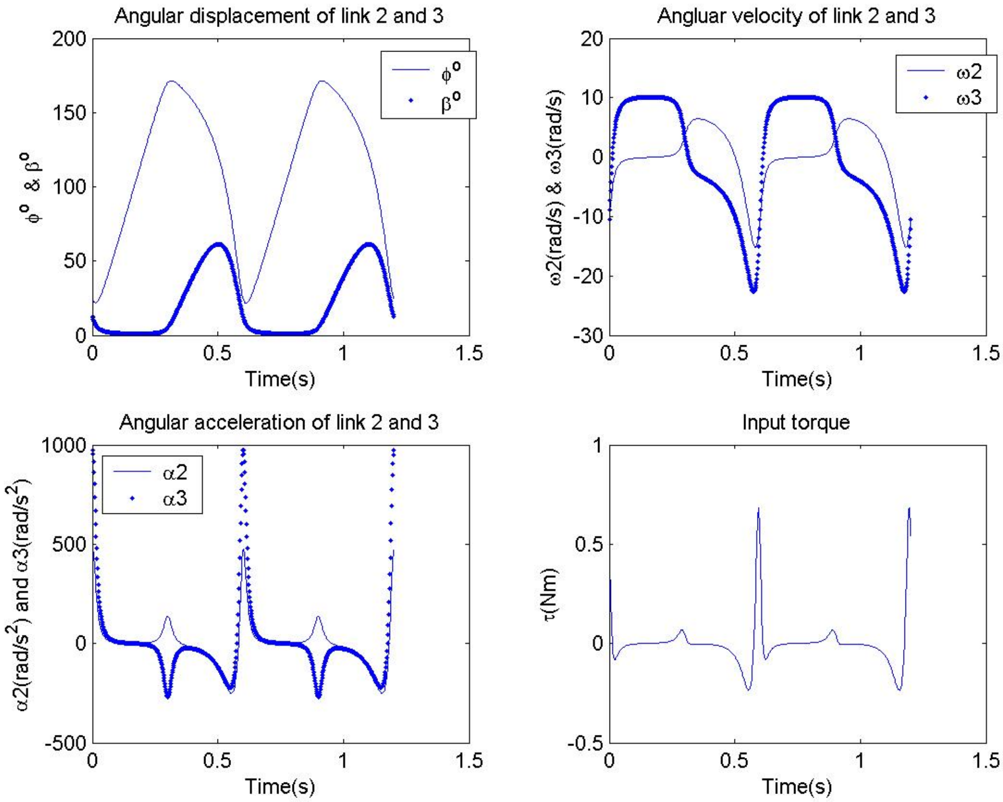


Figure D7: Various plots describing the kinematics of the Four Bar Rocker Mechanism

The input torque graph shows that the input torque required to drive the mechanism is around 0.7Nm but the torque that will be exerted on the motor is significant and hence an H-bridge circuit is essential to make sure that there is some braking involved in the mechanism.

The lengths of the other two members are calculated as *b*=9.9391cm and *c*=5.1804.

Based on the information obtained from the simulation, a dc gearmotor with a stall torque of 1.57Nm which is significantly higher than 0.7Nm of torque required to drive the four bar mechanism, providing a factor of safety of 2.25

which is assumed to overcome frictional losses and other losses which were not considered.

The matlab program used to obtain the above plot is given below:

```
clc
clear
warning off MATLAB:divideByZero
%Defining the constants used in the program
g = 9.8;
a1 = 10/100;
d1 = 5/100;
x = 10;
y =25;
N=100;
%The following code segment is used to calculate the lengths of the unknown members
%of the four bar mechanism
x = pi*x/180;
y = pi*y/180;
b1 = 2*d1*(cos(x)+cos(pi-y)) -2*a1*(cos(pi-y)-cos(x));
b1 = 4*a1*d1/b1;
c1 = sqrt(power(a1+d1,2)+power(b1,2) - 2*(a1+d1)*b1*cos(x));
a = d1;
b = c1;
c = b1;
d = a1;
%printing the values obtained to the screen
a = a*100
b = b*100
c = c*100
d = d*100
a = a/100;
b = b/100;
c = c/100;
d = d/100;
x = x*180/pi
y = y*180/pi
x = x*pi/180;
y = y*pi/180;
%calculate the angular velocity and the mass moment of inertias
i=0;
omega1 = (2*pi*N)/60;
alpha1 = 0;
m1 = 0.1;
m2 = 0.1;
m3 = 0.4;
```

```
J1 = m1*power(a,2)/3;
J2 = m2*power(b,2)/3;
J3 = m3*power(c+0.2,2)/3;
%Simulating the model for two revolutions for every single degree
for j=0:1:2*360
i=i+1;
theta(i) = j*pi/180;
k1 = d/a;
k2 = d/c;
k3 = (power(a,2)-power(b,2)+power(c,2)+power(d,2))/(2*a*c);
k4 = d/b;
k5 = (power(c,2)-power(a,2)-power(b,2)-power(d,2))/(2*a*b);
A = (1-k2)*cos(theta(i)) + k3 - k1;
B = -2*sin(theta(i));
C = k1+k3-(1+k2)*cos(theta(i));
D = (k4+1)*cos(theta(i))+k5-k1;
E = -2*sin(theta(i));
F = (k4-1)*cos(theta(i))+k5+k1;
phi(i) = 2*atan((-B-sqrt(power(B,2)-4*A*C))/(2*A));
beta(i) = 2*atan((-E-sqrt(power(E,2)-4*D*F))/(2*D));
omega2(i) = (-a*omega1*sin(phi(i)-theta(i))/(b*sin(phi(i)-beta(i))));
omega3(i) = (-a*omega1*sin(beta(i)-theta(i))/(c*sin(phi(i)-beta(i))));
alpha2(i) = (-a*alpha1*sin(phi(i)-theta(i))+a*power(omega1,2)*cos(phi(i)-
theta(i))+b*power(omega2(i),2)*cos(phi(i)-beta(i))-c*power(omega3(i),2))/(b*sin(phi(i)-
beta(i)));
alpha3(i) = (-a*alpha1*sin(beta(i)-theta(i))+a*power(omega1,2)*cos(beta(i)-
theta(i))+b*power(omega2(i),2) - c*power(omega3(i),2)*cos(phi(i)-
beta(i)))/(c*sin(phi(i)-beta(i)));
c1 = c*sin(pi - phi(i));
c2 = c*cos(pi - phi(i));
c3 = J3*alpha3(i) - c*m2*g;
c4 = b*sin(beta(i));
c5 = b*cos(beta(i));
c6 = J2*alpha2(i) - m2*g*b;
F2y(i) = (c3*c4 + c6*c1)/(c2*c4 + c5*c1);
F2x(i) = (c6*c2 - c3*c5)/(c1*c5 + c4*c2);
torque(i) = +F2x(i)*a*sin(theta(i))+F2y(i)*a*cos(theta(i));
time(i) = theta(i)/omega1;
phi(i) = phi(i)*180/pi;
theta(i) = theta(i)*180/pi;
beta(i) = beta(i)*180/pi;
end
%plotting the output to a plot screen in order for the data to be analyzed
subplot(2,2,1);
plot(time,phi);
xlabel('Time(s)');
```

```
ylabel('\phi^o  & \beta^o');
title('Angular displacement of link 2 and 3');
legend('\phi^o','\beta^o',2);
hold on;
plot(time,beta,'.');
subplot(2,2,2);
plot(time,omega2);
xlabel('Time(s)');
ylabel('\omega2(rad/s) & \omega3(rad/s)');
title('Angluar velocity of link 2 and 3');
legend('\omega2','\omega3',2);
hold on;
plot(time,omega3,'.');
subplot(2,2,3);
plot(time,alpha2);
hold on;
xlabel('Time(s)');
ylabel('\alpha2(rad/s^2) and \alpha3(rad/s^2)');
title('Angular acceleration of link 2 and 3');
legend('\alpha2','\alpha3',2);
plot(time,alpha3,'.');
subplot(2,2,4);
plot(time,torque);
xlabel('Time(s)');
ylabel('\tau(Nm)');
title('Input torque');
```

APPENDIX E

Programs Developed for Various Microcontrollers

Programming Language for all Programs is CCS© PCW Compiler
PCM Version 3.207

The following code is used to control microcontroller-1 which is intended to interface with the computer:-

```
#include<16F877.h>
#use delay(clock = 20000000)
#use rs232(baud = 19200, parity = N, xmit = PIN_C6, rcv = PIN_C7)
void main(void)
{
 output_low(PIN_C3);//disabling all microcontrollers to ready status
 output_low(PIN_C2);//disabling all microcontrollers to ready status
 output_low(PIN_C1);//disabling all microcontrollers to ready status
 delay_ms(2000);//giving time for other microcontrollers to get ready
 while(1)//getting into an infinite loop to process requests
 {
  output_high(PIN_C2);//Master signal to signal other microcontrollers
                      //to get ready
  while(input(PIN_B6)==1)//check for feedback from napkin position
                      //indicator
  {
   output_low(PIN_C1);//signalling error to BIN microcontroller that
                      //the bin is empty
   while(input(PIN_C0)==0)//wait for bin microcontroller to bring back
                      //bin to base
    printf("Out of napkin, bringing the napkin bin back to base\r");
   printf("\r\n\r\nNapkin base at bottom, Reload and then press any
                      key\r\n");
   getch();//get confirmation from user
  }
  output_high(PIN_C1);//signalling microcontrollers to do their
                      //functions
  output_high(PIN_C2);//signalling microcontrollers to do their
                      //functions
  output_high(PIN_C3);//signalling microcontrollers to do their
                      //functions
  delay_ms(1000);//giving time for them to respond
  output_low(PIN_C3);//signalling microcontrollers to stop once napkin
                      //is pulled out
  printf("press any key when napkin is out\r\n");
  getch();//get feedback from user to get into next process.
 }
}
```

The following code is used to program microcontroller – 2 which is used to control the napkin bin mechanism:-

```
#include<16F876.h>
#use delay(clock = 20000000)
#use rs232(baud = 19200, parity = N, xmit = PIN_C6, rcv = PIN_C7)
unsigned long int timervalue = 255 - (1*20000/4),flagx = 0,flagy = 0;
int flbackward = 0,armupflag = 0,armdownflag = 0,binticks =
0,flhighlowbin=0,binspeedflag = 1,binticksvalue=0;
int flforward = 0, forward = 0, backward = 0;
#int_TIMER1 //interrupt set on timer 1 to keep the motors running in
            //multiples of milliseconds
TIMER1_isr()
{
 if(input(PIN_B6) == 0)
 {
  output_low(PIN_B2); //stop main motor
  output_low(PIN_B3); //stop roller motor
  disable_interrupts(int_TIMER1);//disabling the timer once the bin
                                  //tray has reached the bottom
  disable_interrupts(GLOBAL);//disabling global interrupts
  output_low(PIN_B2); //stop main motor
  output_low(PIN_B3); //stop roller motor
  while(input(PIN_B6) == 0);//wait for error to be reset...
  enable_interrupts(int_TIMER1);//enable timer interrupts
                            //to run the bin motors backward
  enable_interrupts(GLOBAL);//enabling the global interrupts
  output_high(PIN_B2);//start main motor
 }
 set_timer1(timervalue); //set the timer-1 to preset value so that the
                        //interrupt is called every millisecond
 flagy = 1;//triping the flag in order for the main program to know
            //that a millisecond has passed.
if(binticks == binticksvalue)//check for no. of ticks needed to run the
                              //motors
 {
  if(backward == 1) //Check if bin motors are supposed to run backward
  if(flhighlowbin == 0)//check if pulse has to be made high
  {
   output_high(PIN_C4);//in order to run the bin motors backward
   output_high(PIN_C3);//make the pulse high
   flhighlowbin = 1;//set information flag indicating that the pulse
                    //has been made high
  }
  else//if pulse has to be made low
  {
   output_high(PIN_C4);//setting backward direction
   output_low(PIN_C3);//make the pulse low for motors to go through one
                    //step
   flhighlowbin = 0;//set information flag indicating that the pulse
                    //has been made low
  }
  if(forward == 1)//check if bin motors are supposed to run forward
  if(flhighlowbin == 0)//check if pulse has to be made high
  {
   output_low(PIN_C4);//setting the pulse low for motor to run forward
```

```
    output_high(PIN_C3);//make the pulse high
    flhighlowbin = 1;//set information flag indicating that the pulse
                     //has been made high
   }
   else
   {
    output_low(PIN_C4);//setting the pulse low for motors to run forward
    output_low(PIN_C3);//make the pulse low for motor to go through one
                     //step
    flhighlowbin = 0;//set information flag indicating that the pulse
                     //has been made low
   }
   binticks = 0;//resetting the counter for number of ticks to go for
                //the next cycle
  }
  else//if insufficient no. of ticks to run motor then
   binticks = binticks + 1;//increment tick count
}
void main(void)
{
 char ch;
 int flag,delay=2,flag1=0,flag2=0;
 setup_timer_1(T1_INTERNAL | T1_DIV_BY_1);//setup timer1 to be used in
                                          //timer interrupt
 setup_timer_2(T2_DIV_BY_16, 255, 16);//setup timer2 for PWM
 set_timer1(timervalue);//initializing time to the preset value so that
                        //the trigger occours in 1ms.
 delay_ms(3000);
 while(1)
 {
  delay_ms(2000);//initialization time for hardware to get ready....
  output_low(PIN_B2);//stop main motor
  output_low(PIN_B3);//stop roller motor
  while(input(PIN_B6)==0);//wait for error to be resorted by the main
                          //PIC
  output_high(PIN_B2);//start main motor

  if(input(PIN_B5)==0)//Get information on availability of napkin
  {
   printf("checking for napkin\r\n");
   backward = 1; //set backward flag high
   forward = 0;// set forward motion flag low
   enable_interrupts(int_TIMER1);//enable timer interrupts to run the
                                 //bin motors backward
   enable_interrupts(GLOBAL);//enabling the global interrupts
   binticksvalue = 4;
   while(input(PIN_A0)==0);//run motor backwards until the switch at
                           //the bottom is activated
   disable_interrupts(int_TIMER1);//disabling the timer once the bin
                                  //tray has reached the bottom
   disable_interrupts(GLOBAL);//disabling global interrupts
   output_high(PIN_B4);//give information to master PIC that the bin
                       //bottom has reached bottom
   output_low(PIN_B3);//stop the bin roller motors
   output_low(PIN_B2);//stop main motor
   delay_ms(1500);// give some reaction time for the main PIC.
   while(input(PIN_B5)==0); //wait for the bin to be reloaded
```

```
 forward = 1;
 backward = 0;
 output_low(PIN_B4);//say that the bin control PIC is ready to do its
                    //function
 output_low(PIN_B2);//start running the main motor
}
else//if napkin is available
{
 forward = 1;
 backward = 0;
}
while(input(PIN_A3) == 0)//As long as napkin is out..
{
 output_high(PIN_B4);//report error to main PIC
 output_low(PIN_B2);//stop main motor
 output_low(PIN_B3);//stop roller motor
 while(input(PIN_B6)==1);//wait for main PIC to signal error
 while(input(PIN_B6)==0);//wait for error to be resorted by the main
                          //PIC
 output_high(PIn_B2);//start main motor
}
if(input(PIN_B7) == 0)
{
 disable_interrupts(int_TIMER1);//setup timer1 interrupt when it
                                //overflows
 disable_interrupts(GLOBAL);//enabling the GLOBAL interrupts
 output_low(PIN_B2);
 output_low(PIN_B3);
 while(input(PIN_B7) == 0);//wait for main PIC to signal requirement
                          //for napkin..
}
enable_interrupts(int_TIMER1);//setup timer1 interrupt when it
                            //overflows
enable_interrupts(GLOBAL);//enabling the GLOBAL interrupts
output_high(PIN_B2);
while(1)
{
 printf("Into running loop %d %d
         %d\r\n",binticksvalue,forward,backward);
 if(input(PIN_B6) == 1) // make this 1 during runtime...... //master
                          //error signal from 16F877 PIC
   output_high(PIN_B2);//start running the main motor
 else //if master error received then
 {
  disable_interrupts(int_TIMER1);//stop bin motors
  disable_interrupts(GLOBAL);//stop bin motors
  output_low(PIN_B2); //stop main motor
  output_low(PIN_B3); //stop roller motor
  while(input(PIN_B6) == 0);//wait for error to be reset...
  enable_interrupts(int_TIMER1);//setup timer1 interrupt when it
                                //overflows
  enable_interrupts(GLOBAL);//enabling the GLOBAL interrupts
  output_high(PIN_B2);//start main motor
 }
 if(input(PIN_A4) == 0 && flag1 == 0)//check distance between roller
                                      //and bin base
 {
```

```
  binticksvalue = 4; // set ticks accordingly
  output_high(PIN_B3);//run the bin roller motor
  flag1 = 1;//say that the flag has been triggered
 }
 else if(flag1 == 1)//if once triggered then continue with the speed
                    //reduction process
 {
  output_high(PIN_B3);//keep the bin roller motor running
  if(flagy == 1)//check if 1ms has passed..
  {
   flagx = flagx+1;//increment the timer value
   if(flagx == 100)//if time has passed value
   {
    flagx = 0;//reset local count
   }
   flagy = 0;//reset the flag value to find next increment
  }
 }
 else//if base is at lower level
 {
  binticksvalue = 4;
  output_low(PIN_B3);//stop bin roller motor
 }
 if(input(PIN_A3) == 0)//napkin out...
 {
  printf("Napkin Out\r\n");
  disable_interrupts(GLOBAL);//stop running the bin motors
  disable_interrupts(int_TIMER1);//stop running the bin motors
  flag1 = 0;//reset flag value for next cycle
  delay_ms(10);//give time delay
  output_high(PIN_B3);//keep running the bin roller motor
  delay_ms(850);//wait for sufficient napking length to be out
  output_low(PIN_B3);//stop roller motor
  forward = 0;//stop forward motion
  backward = 1;//start backward motion
  binticksvalue = 2;//setting the tick value for speed of bin motors
  enable_interrupts(int_TIMER1);//enabling bin motors
  enable_interrupts(GLOBAL);//enabling bin motor
  delay_ms(1500);//wait for bin to go down a little bit
  disable_interrupts(GLOBAL);//stop bin motors
  disable_interrupts(int_TIMER1);//stop bin motors
  break;//break out of the loop and go for arm motion
 }
}
setup_ccp2(CCP_PWM);//setup PWM to run the arm motor upwards
output_high(PIN_C0);//direction to move the arm upwards
delay_us(10);
set_pwm2_duty(700);//setting duty to limit motor speed

                                 // This sets the time the pulse is
                                // high each cycle.
                                // the high time will be:
                                //  if value is LONG INT:
                                //    value*(1/clock)*t2div
                                //  if value is INT:
                                //    value*4*(1/clock)*t2div
                                // WARNING:  A value too high or low
```

```
                                    //         will prevent the output
                                    //           from changing.

   while(input(PIN_A2)==0);//wait for arm to arrive at the top
   setup_ccp2(CCP_OFF);//turn off the arm motion
   delay_ms(12000);//wait for napkin to be pulled out
   output_low(PIN_C0);//run the arm downwards
   delay_us(10);
   setup_ccp2(CCP_PWM);//setup PWM to run the motor downwards
   set_pwm2_duty(256);//setting duty to limit speed
   while(input(PIN_A1)==0);//wait for arm to come down
   setup_ccp2(CCP_OFF);//stop the arm motion
   output_low(PIN_C0);//stop arm motion completely
   output_low(PIN_C1);//stop arm motion completely
  }
 }
```

The following code is used to control microcontroller – 3 which is used to control bin roller, conveyor system and power flap control circuitry controlled by microcontroller – 4:

```
#include<16F877.h>
#use delay(clock = 20000000)
#use rs232(baud = 19200, parity = N, xmit = PIN_C6, rcv = PIN_C7)
int DELAY = 6; //DEFINE DELAY IN TERMS OF MICROSECONDS
int flhighlowbin=0,flhighlowmain=0; //FLAG FOR INFORMATION ON WHETHER
                                    //TO MAKE THE PIN B6 LOW OR HIGH
int flbinmotor=0,flmainmotor=0,flagticksforbin = 0,flagticksformain =
0,flactivesol = 0;
long solcount1=0,solcount2=0,solcount3 = 0;
unsigned long int timeout = 0, timeout2;
int flin = 0, flout = 0, dropaction = 0, counttimeout2=0;
unsigned long int timervalue = 65535 - (1*20000/4);
long dropsolenoid = 0;
#int_TIMER1//interrupt service routine to operate verious mechanisms
TIMER1_isr()
{
 set_timer1(timervalue);//setting timer value in order to cause
                        //interrupt every 1 millsecond
  if(flagticksforbin == 6)//check if 6 milliseconds have gone by
 {
  if(flbinmotor)//if bin motor needs to be actuated
  if(flhighlowbin == 0)//check flags in order to make the pin high to
                        //drive bin roller motor
  {
   output_high(PIN_B6);
   flhighlowbin = 1;//setting flag in order to make pin low for next
                        //cycle
  }
  else
  {
   output_low(PIN_B6);
   flhighlowbin = 0;//setting flag in order to make pin high for next
                        //cycle
  }
  flagticksforbin = 0;//resetting the counter for entire bin motor
                        //cycle
 }
 else//if count has not been reached then increment counter
  flagticksforbin = flagticksforbin + 1;
 if(flagticksformain == 4)//check if 4 milliseconds have gone by to
                        //drive main motor
 {
  if(flmainmotor)//if main motor needs to be acutated
  if(flhighlowmain == 0)//check flags in order to make the pin high to
                        //drive main motor
  {
   output_high(PIN_B4);
   flhighlowmain = 1;//setting flag in order to make pin low for next
                        //cycle
  }
  else
  {
```

```
   output_low(PIN_B4);
   flhighlowmain = 0;//setting flag in order to make pin high for next
                  //cycle
  }
  flagticksformain = 0;//resetting the counter for entire main motor
                        //cycle
 }
 else//if counter has not reached then increment them
  flagticksformain = flagticksformain + 1;
 if(flactivesol == 1)//if folding mechanism needs to be actuated
 {
  flin = 1;//setting flag to indicate that the process has been
            //triggered
  dropaction = 1;//actuate the silverware drop mechanism
  if(solcount1<700)//check if napkin has come between fingers of
                  //folding mechanism
   solcount1 += 1;
  else//hold napkin within the fingers
   output_high(PIN_B2);
 }
 else if(flactivesol == 0 && flin == 1)//if folding process is complete
 {
  output_low(PIN_B2);//release napkin
  if(solcount2<300)//counter triggered to hold finger in intermediate
                  //position
   solcount2 += 1;
  else//move finger to intermediate position
  {
   output_high(PIN_B2);
   flin = 0;//indicate that the process has been completed
   flout = 1;//trigger the next process
  }
  if(solcount2 >= 300)//lock finger in intermediate position
   output_high(PIN_B3);
 }
 else if(flactivesol == 0 && flout == 1)
 {
  if(solcount3 < 1500)//set counter to bring finger to home position
   solcount3 += 1;
  else
  {
   flout = 0;//set flag to bring folding mechanism-1 to home position
   solcount1 = 0;
   solcount2 = 0;
   solcount3 = 0;
  }
 }
 else//bring back finger to home position
 {
  output_low(PIN_B2);
  output_low(PIN_B3);
 }
 if(dropaction == 1)//if silverware needs to be dropped on to the
                        //napkin
 {
  if(dropsolenoid < 2600)//counter for providing napkin to move
                        //underneath the drop mechanism
```

```
    dropsolenoid += 1;
   if(dropsolenoid >2500)//bring mechanism in holding position
   {
    output_high(PIN_D3);
    dropaction =0;//indicatate that the process is complete
   }
   if(dropsolenoid >1500 && dropsolenoid<2500)//drop solenoid
    output_low(PIN_D3);
  }
  else//resetting entire process for dropping silverware
   dropsolenoid = 0;
}
void main(void)
{
 int flagset = 0;
 printf("Timervalue = %lu\r\n",timervalue);
 delay_ms(2000);//setup time
 setup_timer_1(T1_INTERNAL | T1_DIV_BY_1);
 set_timer1(timervalue);
 enable_interrupts(int_TIMER1);
 enable_interrupts(GLOBAL);
 output_low(PIN_B3);//initializing solenoid
 output_high(PIN_B0);//providing power for four bar mechanisms
 output_high(PIN_B1); //providing power for four bar mechanisms
 output_high(PIN_D3); //initializing solenoid
 while(1)
 {
  if(input(PIN_C0) == 1 )//if bin motor is to be actuated
   flbinmotor = 1;
  else
   flbinmotor = 0;
  if(input(PIN_A4) == 0 || input(PIN_A3) == 0)//check if main motor is
                                              //to be actuated
   flmainmotor = 1;
  else
   flmainmotor = 0;
  if(!input(PIN_A0))//if silverware drop mechanism is to be actuated
   flactivesol = 1;
  else
   flactivesol = 0;
 }
}
```

The following code is used to program microcontroller – 4 used to control the folding mechanisms:

```
#include<16F876a.h>
#use delay(clock = 20000000)
#use rs232(baud = 19200, parity = N, xmit = PIN_C6, rcv = PIN_C7)
unsigned long int timervalue = 65535 - (1*20000/4);
int flapright=0, flapleft=0, flapcenteractive = 0, flapcenter = 0;
int flaprightend = 0,rightprocessend = 0, flapleftend = 0, process = 1;
unsigned long int luntticks=0,delayticks = 0,delayticks1 = 0;
unsigned long int process1ticks = 0, process2ticks = 0;
unsigned long int processfingerbottom = 0, processfingertop =
0,countfingertemp = 0;
int flag3 = 0, process3count = 0,finger = 0;
int top = 0, bottom = 0, counter = 1;
unsigned long int afterarmup = 0, process2ticksa=0, process1ticksa=0;
int bringarmup = 0,flagprocess1=0, flagprocess2=0;
#int_TIMER2//Timer interrupt configured to cause interrupt every 500us
TIMER2_isr()
{
 if(bottom == 1)//if finger is to mover to bottom position
 {
  if(processfingerbottom <= 4000)//Giving 4s for finger to move to
                                    //bottom position
  {
   if(counter <=5)//Generation of pulse width equivalent to 2.5ms out
                  //of 6ms to drive servo motor
   {
    output_high(PIN_C1);
    counter += 1;
   }
   else
   {
    output_low(PIN_C1);
    counter += 1;
   }
   if(counter == 11)//Resetting value to produce continuous pulse
                  //pattern
    counter = 1;
   processfingerbottom += 1;
  }
  else
   output_low(PIN_C1); //Stop the servo motor after 4s
 }
 else
 {
  if(top == 0)
   output_low(PIN_C1);//setting output pin low in case it is set high
                         //in the process
  processfingerbottom = 0;//reset 4s counter value
 }
 if(top == 1)//if finger has to be moved towards the top position
 {
  if(processfingertop <= 4000) //Counter value check for 4s to move
                              //finger towards top position
  {
```

```
    if(counter == 1) // Generate pulse with a pulse width of 0.5ms out
                         //of 4ms to drive servo motor
    {
     output_high(PIN_C1);
     counter += 1;
    }
    else
    {
     output_low(PIN_C1);
     counter += 1;
    }
    if(counter == 7)//resetting counter to generate continuous pulse
     counter = 1;
    processfingertop += 1; //incrementing counter for 4s time interval
                         //to ensure finger reaches the top position
   }
   else
   output_low(PIN_C1);//to ensure that the pulse width is made low once
                   //finger is moved to desired position
  }
  else
  {
   if(bottom == 0)
    output_low(PIN_C1);
   processfingertop = 0;//resetting counter for the 4s time interval
  }
 }
 #int_TIMER1//timer interrupt used to cause a timer interrupt for every
             //1ms
 TIMER1_isr()
 {
  set_timer1(timervalue);//setting timer to a preset value to ensure
                         //interrupt is caused every 1ms
  if(countfingertemp <= 8100)
   countfingertemp  += 1;
  else
   countfingertemp = 0;
  if(flapright == 1 && process == 1)//To actuate flap on the right side
                                     //to place right fold
  {
   if(process1ticksa <= 700)//check if sufficient time has elapsed
   {
    process1ticksa += 1;//increment counter
    output_low(PIN_C0);
   }
   else if(input(PIN_A3) && flagprocess1 == 0)//check if flaps must be
                                              //actuated
   {
    output_low(PIN_B2);//slowing down motor to acheive required speed
    delay_us(200);
    output_high(PIN_B2);
    output_high(PIN_C0);//signal Microcontroller-3 to stop conveyor
                        //system
   }
   else if(process1ticks<=0)
   {
    output_low(PIN_C0);//signal Microcontroller-3 to start conveyor
```

```
                     //system again
  process1ticks += 1;
  flagprocess1 = 1;
 }
 else//resetting flags and O/P pins
 {
  flagprocess1 = 0;
  process1ticks = 0;
  output_low(PIN_B2);
  output_low(PIN_C0);
  flapright = 0;
  flapleft = 1;
  flaprightend = 1;
  process = 2;
  process1ticksa = 0;
  process2ticksa = 0;
 }
}
if(flapleft == 1 && process == 2)//if left flap must be actuated
{
 if(process2ticksa<10)
 {
  process2ticksa += 1;
  output_low(PIN_C0);//run conveyor system
 }
 else if(input(PIN_A0) && flagprocess2 == 0)
 {
  output_low(PIN_B0);//slow down motor used to actuate left flap
  delay_us(200);
  output_high(PIN_B0);
  output_high(PIN_C0);//signal Microcontroller-3 to stop conveyor
                     //system
 }
 else if(process2ticks<=1)
 {
  output_low(PIN_C0);//signal Microcontroller-3 to start conveyor
                     //systme again
  process2ticks += 1;
  flagprocess2 = 1;
 }
 else//resetting flags and O/P pins
 {
  flagprocess2 = 0;
  process2ticks = 0;
  process2ticksa = 0;
  output_low(PIN_B0);
  output_low(PIN_C0);
  flapleft = 0;
  flapcenter = 0;
  flapleftend = 1;
  process = 3;
 }
}
if(process == 3)//if Folding Mechanism-3 must be actuated
{
 if(process3count == 0 && !input(PIN_A2))//check if finger must be
                                         //actuated
```

```
 finger = 1;
if(finger == 1)//ask finger to move to bottom position to place first
                //fold
{
  top = 0;
  bottom = 1;
  finger = 2;
}
if(process3count > 0 && finger == 2)//check if more than one fold has
                                        //been placed
{
 if(!input(PIN_C3))//take carriage outside
 {
  output_low(PIN_B5);
  output_high(PIN_B4);
 }
 else//stop carriage
 {
  output_low(PIN_B4);
  top = 0;
  bottom = 0;
  afterarmup = 0;
  finger = 3;
 }
}
if(finger == 3)//if finger has to be taken back to home position
{
 top = 1;
 bottom = 0;
 if(afterarmup < 2000)
  afterarmup += 1;
 else
  finger = 4;
}
if(finger == 4)//if finger mechanism has to be taken to home position
{
 top = 0;
 bottom = 0;
 if(!input(PIN_C2))//bring carriage to home position
 {
  output_low(PIN_B4);
  output_high(PIN_B5);
 }
 else
 {
  finger = 0;//finger mechanism entire reset
  output_low(PIN_B5);
 }
}
}
if(flapcenter == 1 && process == 3)//if flap is to be actuatede
{
 if(input(PIN_A4) && flag3 == 0 && delayticks>300)//actuate motor to
                                        //actuate flap mechanism
 {
  delayticks1 += 1;
  if(process3count == 0)
```

```
    output_high(PIN_C0);
   output_high(PIN_B6);
  }
  else if(countticks > 50)//give sufficient time to stop stop motor
  {
   delayticks1 = 0;
   output_low(PIN_B6);
   output_low(PIN_C0);
   flapcenter = 0;
   if(process3count>10)//set flags to actuate side flaps
   {
    process = 1;
    process3count = 0;
    flapright = 0;
    delayticks = 0;
    output_low(PIN_C0);//keep running conveyor system
   }
   else
   {
    process3count += 1;
    output_low(PIN_C0);
   }
   countticks = 0;
   flag3 = 0;
  }
  else if((delayticks<=300) && input(PIN_A4) && flag3 == 0)
   delayticks+=1;
  else
  {
   countticks += 1;
   flag3 = 1;
  }
 }
}
void main(void)
{
 int flagz = 0;
 delay_ms(2000);
 setup_timer_1(T1_INTERNAL|T1_DIV_BY_1);
 setup_timer_2(T2_DIV_BY_16, 155, 1);
 enable_interrupts(int_TIMER2);//enable timer interrupts to run the bin
                              //motors backward
 enable_interrupts(int_TIMER1);
 enable_interrupts(GLOBAL);
 while(1)
 {
  if(!input(PIN_A5))
     flapright = 1;//if flaps must be actuated
  if(!input(PIN_A1) && flapleftend == 1)
   flapcenter = 1;//if center flap must be actuated
 }
}
```

The following code is used to control microcontroller – 5 which is used to obtain feedback from the bin motors used to traverse the napkin bin tray:

```
#include <16F876.h>
#use delay(clock=20000000)
#fuses HS,NOWDT
#use rs232(baud=115200,parity=N,xmit=PIN_C6,rcv=PIN_C7)
void main(void)
{
 int
a,b,c,aa,bb,cc,flag=1,val1,val2,val3,val1a,val2a,val3a,aval,bval,cval;
 long counta=0,countb=0,countc=0;
 int16 value1=0,value2=0,value3=0;
 int i=0;
 set_tris_b(0xff);//define all pins in port as inputs
 set_tris_c(0xff);//define all pins in port as inputs
 bb = 0;
 cc = 0;
 aa = 0;
 b = input_b();//get input from port B
 c = input_c();//get input from port C
 a = b;
 output_low(PIN_A0);//signal no error
 output_high(PIN_A1);//singal no error
 for(i=0;i<4;i++)//formatting data
 {
  shift_right(&b,1,0);
  shift_left(&a,1,0);
  shift_left(&c,1,0);
 }
 for(i=0;i<4;i++)//formatting data
 {
  shift_right(&a,1,0);
  shift_right(&c,1,0);
 }
 aa=a;
 bb=b;
 cc=c;
 while(1)//infinite loop
 {
  /*get input continuously*/
  b = input_b();
  c = input_c();
  a = b;
  for(i=0;i<4;i++)
  {
   shift_right(&b,1,0);
   shift_left(&a,1,0);
   shift_left(&c,1,0);
  }
  for(i=0;i<4;i++)
  {
   shift_right(&a,1,0);
   shift_right(&c,1,0);
  }
 aval = aa - a;//get incremented value
```

```
bval = bb - b;//get incremented value
cval = cc - c;//get incremented value
aa = a;
bb = b;
cc = c;
value1 = value1+aval;//maintain position details for encoder1
value2 = value2+bval;//maintain position details for encoder2
value3 = value3+cval;//maintain position details for encoder3
if(value1>400)//if value goes beyod limit
{
 value1 = 0;//reset counter for encoder1
 value2 = 0;//reset counter for encoder2
 value3 = 0;//reset counter for encoder3
}
 if(aval>0)// if value must be incremented for encoder1
{
 counta++;
}
else if(aval<0)// if value must be decremented for encoder1
{
 counta--;
}
if(bval>0)// if value must be incremented for encoder2
{
 countb++;
}
if(bval<0)// if value must be decremented for encoder2
{
 countb--;
}

if(cval>0)// if value must be incremented for encoder3
{
 countc++;
}
else if(cval<0)// if value must be decremented for encoder2
{
 countc--;
}
if(counta>50)//check if counters must be resetted
{
 counta = countb= countc=0;
}
if(counta>=countb)//check error between encoder1 and encoder2
val1 = counta - countb;
else
val1 = countb - counta;
if(countb>=countc)//check error between encoder2 and encoder3
val2 = countb - countc;
else
val2 = countc - countb;
if(counta>=countc)//check error between encoder1 and encoder3
val3 = counta - countc;
else
val3 = countc - counta;
if(val1>1||val2>1||val3>1)//check if there is error
{
```

```
  output_high(PIN_A0);//signal +ve error
  output_low(PIN_A1);//signal -ve error
  }
 }
}
```

The following code is used to power the infrared LED's which is used in the feedback circuitry:

```
#include <16F876a.h>
#use delay(clock=20000000)
#use rs232(baud=19200, xmit=PIN_C6, rcv=PIN_C7)



void main() {
   long int value = 256;//duty for PWM
   short flag;
   set_tris_a(0xff);
   setup_timer_2(T2_DIV_BY_1, 127, 1);//setup timer
   setup_ccp2(CCP_PWM);//make mode as PWM
   set_pwm2_duty(51);//set duty
   while(1);//run into infinite loop in order to have continuous PWM
}
```

APPENDIX F

SPECIFICATION SHEET FOR DC GEARMOTOR USED IN FOLDING MECHANISM-2 AND FOLDING MECHANISM-3

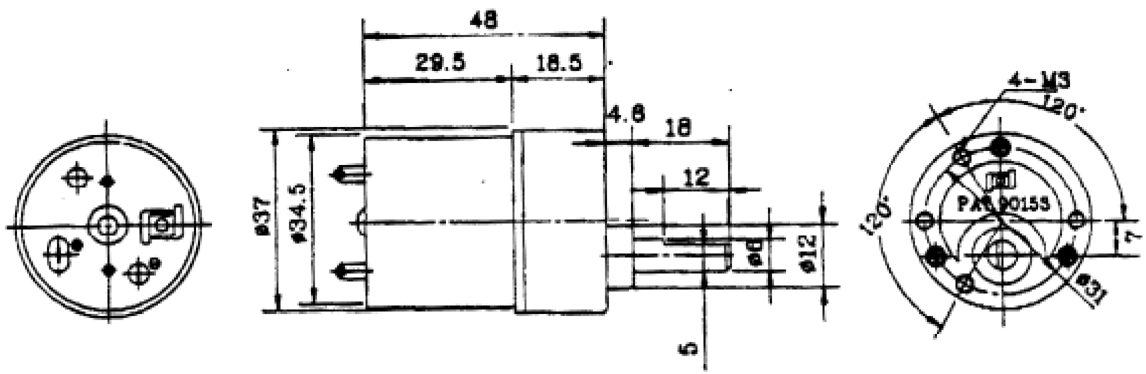
SPECIFICATION SHEET

DATE: MAY 05,98'
NO. : SS98050503
PAGE: 3/3

TYPE: HN-GH35GMA TYPE
MODEL NO. : HN-GH12-1926Y
CUSTOMER PART NO.:

Jameco Part Number 152910

III. OUTTER DIMENSIONS :



IV. GEAR RATIO : 100:1

TYPE	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
RATIO	6	10	18	30	50	60	75	90	100	150	180	300	500	810	900	1000	1500	3000
DIM "L" MM	18.5	18.5	18.5	18.5	18.5	18.5	18.5	18.5	18.5	24	24	24	24	24	24	24	26	26
EFF %	81	81	73	73	66	66	66	66	66	59	59	59	59	53	53	53	48	48
ALLOWABLE TORQUE kg-cm	0.7	0.7	1.0	1.3	2.3	3	4	4	5	6	6	6	6	6	6	6	6	6

SPECIFICATION SHEET

DATE: MAY 05,98'
NO. : SS98050503
PAGE: 1/3

TYPE: HN-GH35GMA TYPE
MODEL NO. : HN-GH12-1926Y
CUSTOMER PART NO.:

I. SPECIFICATIONS A :

1. RATED VOLTAGE : DC 12V
2. DC VOLTACE OPERATING RANGE : DC 4.5-12V
3. TORQUE(LOAD) AT MAXIMUM EFFICIENCY : OVER 2.37 Kg-cm
4. STALL TORQUE : OVER 11.80 Kg-cm
5. NO LOAD SPEED: 44 RPM
6. SPEED AT MAXIMUM EFFICIENCY(UNDER LOAD AT 2.37 Kg-cm) : 35 RPM
7. NO LOAD CURRENT : MAX. 48 mA
8. CURRENT AT MAXIMUM EFFICIENCY(UNDER LOAD AT 2.37 Kg-cm) : MAX. 172 mA
9. SHAFT END PLAY : 0..8 M/M
10. STARTED TORQUE : OVER 11.80 Kg-cm
11. INSULATION RESISTANCE : 10 M OHM
12. DIELECTRIC STRENGTH : 300V DC
13. MAXIMUM PERCENT EFFICIENCY : 41%
14. DIRECTION OF ROTATION : "CCW" WHEN VIEWED FROM OUTPUT SHAFT SIDE.
15. GEAR RATIO: 100:1

APPENDIX G

SPECIFICATION SHEET FOR DC GEARMOTOR USED IN FINGER MECHANISM

SPECIFICATION SHEET

DATE: JUN.07,99'.
NO.: HN99060708
PAGE: 2/2

CUSTOMER NAME:
TYPE: HN-GH35GMB TYPE
MODEL NO.: HN-GH12-1640Y
CUSTOMER PART NO.:

Jameco Part Number 164785

III. SPECIFICATIONS :

1. RATED VOLTAGE: DC 12V

2. DC VOLTAGE OPERATING RANGE: DC 4.5-24V

3. TORQUE (LOAD) AT MAXIMUM EFFICIENCY : 180 g-cm

4. STALL TORQUE: 760 g-cm

5. NO LOAD SPEED: 300 RPM +/-10%

6. SPEED AT MAXIMUN EFFICIENCY (UNDER LOAD AT 180g-cm): 220 RPM +/- 10%

7. NO LOAD CURRENT: MAX. 45 mA

8. CURRENT AT MAX. EFFICIENCY (UNDER LOAD AT 180 g-cm): MAX. 120 mA

9. SHAFT END PLAY: 0.8 m/m

10. STARTED TORQUE: 760 g-cm

11. INSULATION RESISTANCE: 10M OHM

12. DIELECTRIC STRENGTH: 300V DC

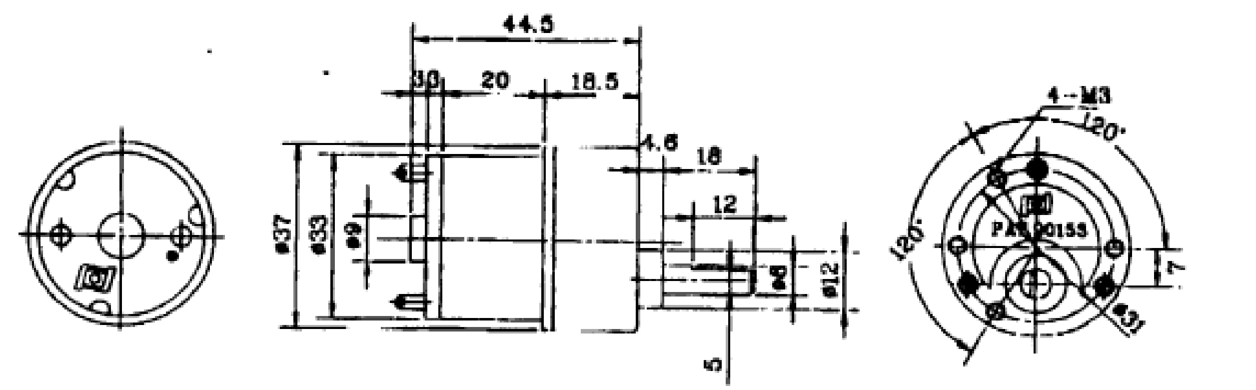
13. GEAR RATIO : 10:1

SPECIFICATION SHEET

DATE: JUN.07,99'.
NO.: HN99060708
PAGE: 1/2

CUSTOMER NAME:
TYPE: HN-GH35GMB TYPE
MODEL: HN-GH12-1640Y
CUSTOMER PART NO.:

I. OUTER DIMENSIONS :



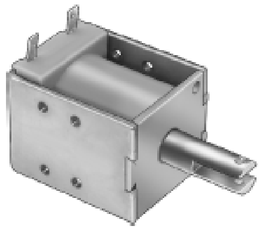
II.GEAR RATIO : 10:1

TYPE	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
RATIO	6	10	18	30	50	60	75	90	100	150	180	300	500	810	900	1000	1500	3000
DIM "L" MM	18.5	18.5	18.5	18.5	18.5	18.5	18 5	18.5	18.5	24	24	24	24	24	24	24	26	26
EFF %	81	81	73	73	66	66	66	66	66	59	59	59	59	53	53	53	48	48
ALLOWABLE TORQUE kg-cm	0.7	0.7	1.0	1.3	2.3	3	4	4	5	6	6	6	6	6	6	6	6	6

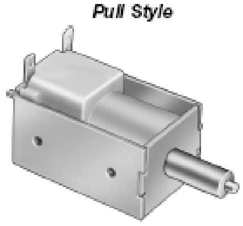
APPENDIX H

DATASHEET FOR SOLENOIDS USED IN FOLDING MECHANISM-1

More About Linear Solenoids



Pull Style

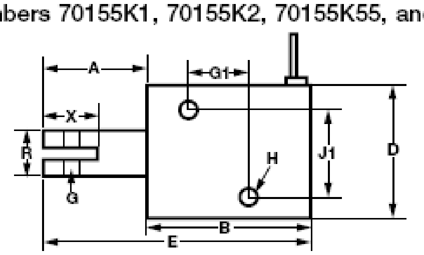


Push Style

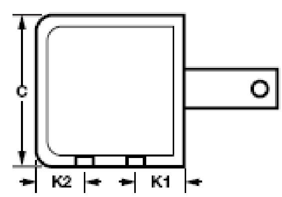
MCM Part No.	Input Voltage	Max. Stroke	Force, oz. @ 1/8" Stroke	Power Rating (Watts)	Coil Resistance (Ohms) 12V Model	24V Model	120 VAC Model
Pull Style—Intermittent Duty							
70155K1	DC	1/2"	20	11.5	12.5	50.1	—
70155K3	DC	1/2"	43	19.0	7.6	30.3	—
70155K5	DC	1"	120	38.0	3.8	15.2	—
70155K72	120 VAC	1/2"	14	22	—	—	232.1
70156K47	120 VAC	7/8"	24	40	—	—	85
70155K41	120 VAC	7/8"	45	60	—	—	32.2
Pull Style—Continuous Duty							
70155K2	DC	1/2"	11	5.0	28.8	115.0	—
70155K4	DC	1/2"	29	8.0	18.0	72.0	—
70155K6	DC	1"	76	11.0	13.1	52.4	—
70155K55	120 VAC	1/2"	12	8	—	—	400
70155K48	120 VAC	7/8"	12	12	—	—	200
70155K42	120 VAC	7/8"	14	10	—	—	133
Push Style—Intermittent Duty							
70155K11	DC	1/2"	42	19.0	7.6	30.3	—
70155K13	DC	1"	40	30.0	7.4	30.3	—
70155K65	120 VAC	1/2"	14	22	—	—	220
70155K61	120 VAC	7/8"	20	40	—	—	85
Push Style—Continuous Duty							
70155K12	DC	1/2"	16	8.0	18.0	72.0	—
70155K14	DC	1"	20	29.0	30.3	117.0	—
70155K66	120 VAC	1/2"	9	8	—	—	400
70155K62	120 VAC	7/8"	10	12	—	—	200

Pull Style

For part numbers 70155K1, 70155K2, 70155K55, and 70155K72.



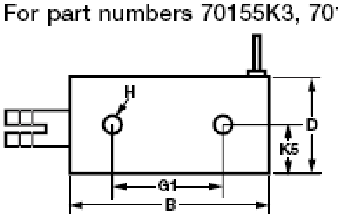
Side View



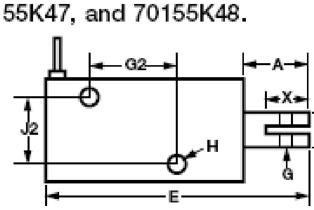
Top View

Retracted Rod Lg. (A)	Frame Lg. (B)	Frame Wd. (C)	Frame Ht. (D)	Overall Lg., Rod Retracted (E)	Rod Cutaway Depth (X)	Rod Hole Dia. (G)	Rod Dia. (R)	Mounting Hole Size (H)	Mounting Holes, Center-to-Center (G1)	(J1)	Mounting Holes, Center-to-Frame Edge (K1)	(K2)
.612"	1.13"	1.19"	.94"	1.74"	.41"	.096"	.31"	6-32	.406"	.625"	.41"	.31"

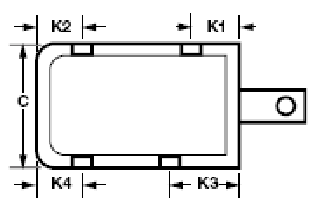
For part numbers 70155K3, 70155K4, 70155K47, and 70155K48.



Side View



Side View



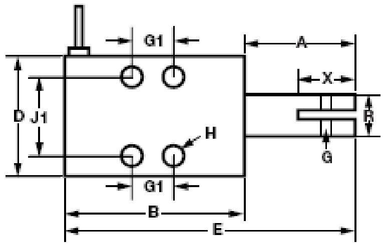
Top View

Retracted Rod Lg. (A)	Frame Lg. (B)	Frame Wd. (C)	Frame Ht. (D)	Overall Lg., Rod Retracted (E)	Rod Cutaway Depth (X)	Rod Hole Dia. (G)	Rod Dia. (R)	Mounting Hole Size (H)	Mounting Holes, Center-to-Center (G1)	(G2)	(J2)	Mounting Holes, Center-to-Frame Edge (K1)	(K2)	(K3)	(K4)	(K5)
.62"	1.846"	1.189"	.94"	2.466"	.41"	.096"	.31"	6-32	1"	.81"	.63"	.42"	.426"	.63"	.41"	.47"

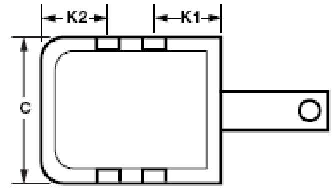
More About Linear Solenoids

Pull Style *(Cont.)*

For part numbers 70155K5, 70155K6, 70155K41, and 70155K42.



Side View

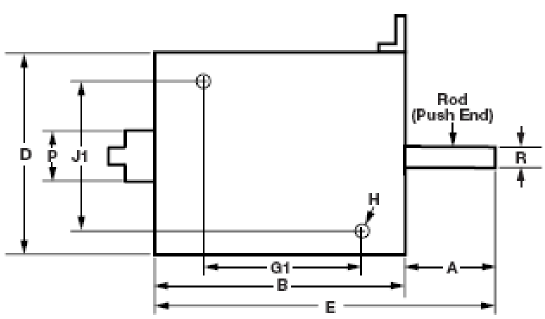


Top View

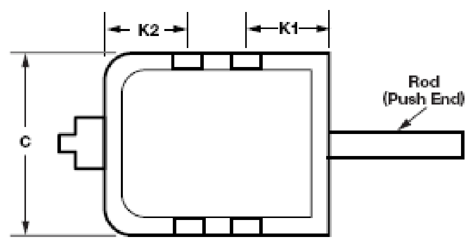
Retracted Rod Lg. (A)	Frame Lg. (B)	Frame Wd. (C)	Frame Ht. (D)	Overall Lg., Rod Retracted (E)	Rod Cutaway Depth (X)	Rod Hole Dia. (G)	Rod Dia. (R)	Mounting Hole Size (H)	Mounting Holes, Center-to-Center (G1)	Mounting Holes, Center-to-Center (J1)	Mounting Holes, Center-to-Frame Edge (K1)	Mounting Holes, Center-to-Frame Edge (K2)
1.165"	2"	1.63"	1.44"	3.165"	.69"	.128"	.437"	8-32	.5"	.936"	.87"	.63"

Push Style

For part numbers 70155K65 and 70155K66.



Side View



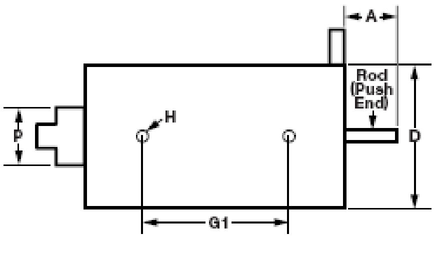
Top View

Extended Push End Lg. (A)	Frame Lg. (B)	Frame Wd. (C)	Frame Ht. (D)	Overall Lg., Push End Extended (E)	Rod (Push End) Dia. (R)	(P)	Mounting Hole Size (H)	Mounting Holes, Center-to-Center (G1)	Mounting Holes, Center-to-Center (J1)	Mounting Holes, Center-to-Frame Edge (K1)	Mounting Holes, Center-to-Frame Edge (K2)
.5"	1.13"	1.19"	.94"	1.63"	.093"	.31"	6-32	.406"	.625"	.32"	.41"

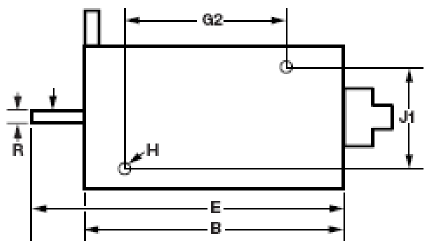
More About Linear Solenoids

Push Style *(Cont.)*

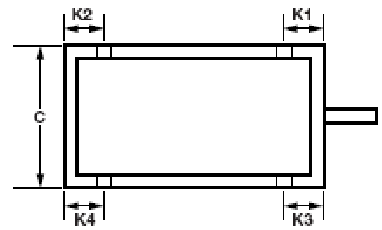
For part numbers 70155K11, 70155K12, 70155K61, and 70155K62.



Side View



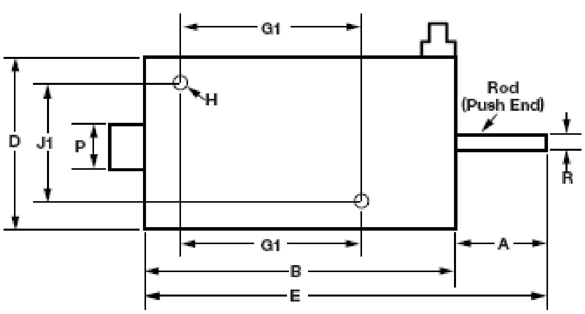
Side View



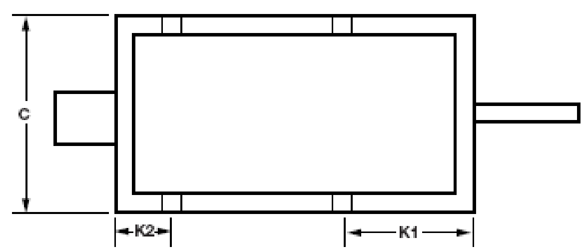
Top View

Extended Push End Lg. (A)	Frame Lg. (B)	Frame Wd. (C)	Frame Ht. (D)	Overall Lg., Push End Extended (E)	Rod (Push End) Dia. (R)	(P)	Mounting Hole Size (H)	Mounting Holes, Center-to-Center (G1)	Mounting Holes, Center-to-Center (G2)	Mounting Holes, Center-to-Center (J1)	Mounting Holes, Center-to-Frame Edge (K1)	Mounting Holes, Center-to-Frame Edge (K2)	Mounting Holes, Center-to-Frame Edge (K3)	Mounting Holes, Center-to-Frame Edge (K4)
1"	1.846"	1.19"	.94"	2.846"	.093"	.312"	6-32	1"	.812"	.631"	.404"	.63"	.42"	.42"

For part numbers 70155K13 and 70155K14.



Side View



Top View

Extended Push End Lg. (A)	Frame Lg. (B)	Frame Wd. (C)	Frame Ht. (D)	Overall Lg., Push End Extended (E)	Rod (Push End) Dia. (R)	(P)	Mounting Hole Size (H)	Mounting Holes, Center-to-Center (G1)	Mounting Holes, Center-to-Center (J1)	Mounting Holes, Center-to-Frame Edge (K1)	Mounting Holes, Center-to-Frame Edge (K2)
1"	1.735"	1"	.84"	2.735"	.125"	.31"	8-32	.5"	.437"	.77"	.465"

APPENDIX I

DATASHEET FOR MICROCONTROLLERS PIC-16F876 AND PIC-16F877

PIC16F87X

28/40-Pin 8-Bit CMOS FLASH Microcontrollers

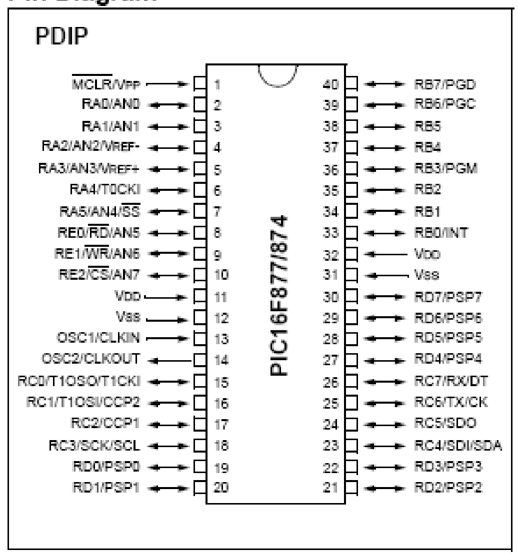
Devices Included in this Data Sheet:

- PIC16F873
- PIC16F874
- PIC16F876
- PIC16F877

Microcontroller Core Features:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory,
  Up to 368 x 8 bytes of Data Memory (RAM)
  Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
  Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
  - < 0.6 mA typical @ 3V, 4 MHz
  - 20 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

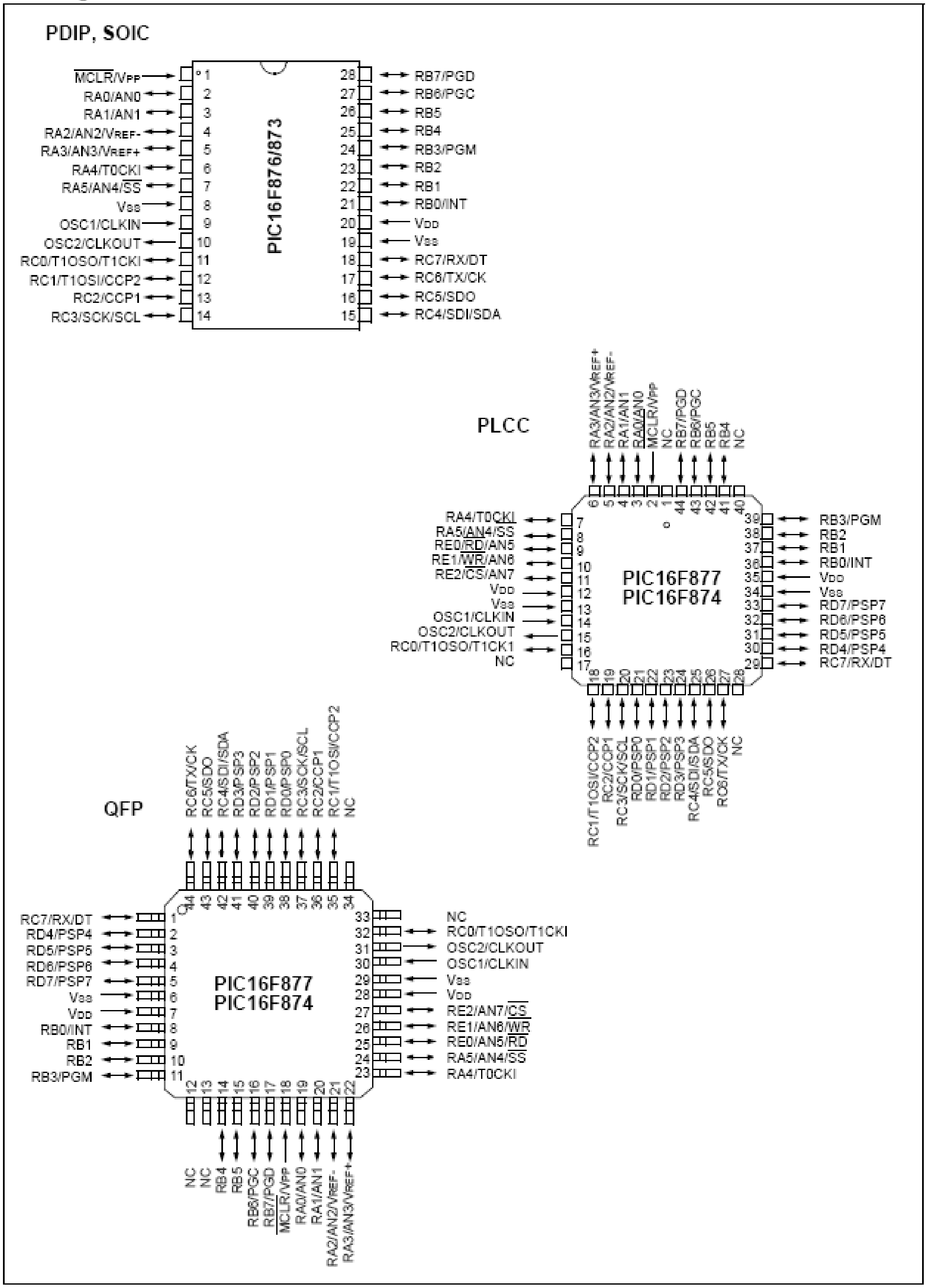
Pin Diagram



Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during SLEEP via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master mode) and I^{2}C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external $\overline{RD}$, $\overline{WR}$ and $\overline{CS}$ controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

Pin Diagrams



APPENDIX J

DATASHEET FOR HIGH SIDE SWITCH IPS 511

International Rectifier

Data Sheet No.PD 60155H

IPS511/IPS511S

FULLY PROTECTED HIGH SIDE POWER MOSFET SWITCH

Features

- Over temperature protection (with auto-restart)
- Short-circuit protection (current limit)
- Active clamp
- E.S.D protection
- Status feedback
- Open load detection
- Logic ground isolated from power ground

Description

The IPS511/IPS511S are fully protected five terminal high side switches with built in short circuit, over-temperature, ESD protection, inductive load capability and diagnostic feedback. The output current is controlled when it reaches I_{lim} value. The current limitation is activated until the thermal protection acts. The over-temperature protection turns off the high side switch if the junction temperature exceeds Tshutdown. It will automatically restart after the junction has cooled 7°C below Tshutdown. A diagnostic pin is provided for status feedback of short-circuit, over-temperature and open load detection. The double level shifter circuitry allows large offsets between the logic ground and the load ground.

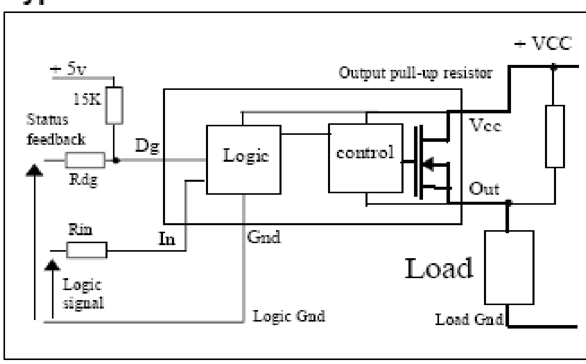
Product Summary

$R_{ds(on)}$	135mΩ (max)
V_{clamp}	50V
I_{Limit}	5A
$V_{open\ load}$	3V

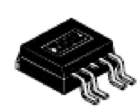
Truth Table

Op. Conditions	In	Out	Dg
Normal	H	H	H
Normal	L	L	L
Open load	H	H	H
Open load	L	H	H
Over current	H	L (limiting)	L
Over current	L	L	L
Over-temperature	H	L (cycling)	L
Over-temperature	L	L	L

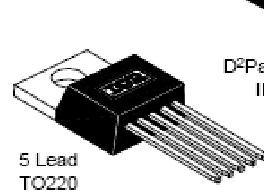
Typical Connection



Packages



5 Lead
D²Pak (SMD220
IPS511S

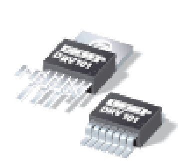


5 Lead
TO220
IPS511

APPENDIX K

DATASHEET FOR PWM/SOLENOID DRIVER

DRV101



SBVS008A – JANUARY 1998 – REVISED OCTOBER 2003

PWM SOLENOID/VALVE DRIVER

FEATURES

- HIGH OUTPUT DRIVE: 2.3A
- WIDE SUPPLY RANGE: +9V to +60V
- COMPLETE FUNCTION
  PWM Output
  Internal 24kHz Oscillator
  Digital Control Input
  Adjustable Delay and Duty Cycle
  Over/Under Current Indicator
- FULLY PROTECTED
  Thermal Shutdown with Indicator
  Internal Current Limit
- PACKAGES: 7-Lead TO-220 and 7-Lead Surface-Mount DDPAK

APPLICATIONS

- ELECTROMECHANICAL DRIVER:

Solenoids	Positioners
Actuators	High Power Relays/Contactors
Valves	Clutch/Brake

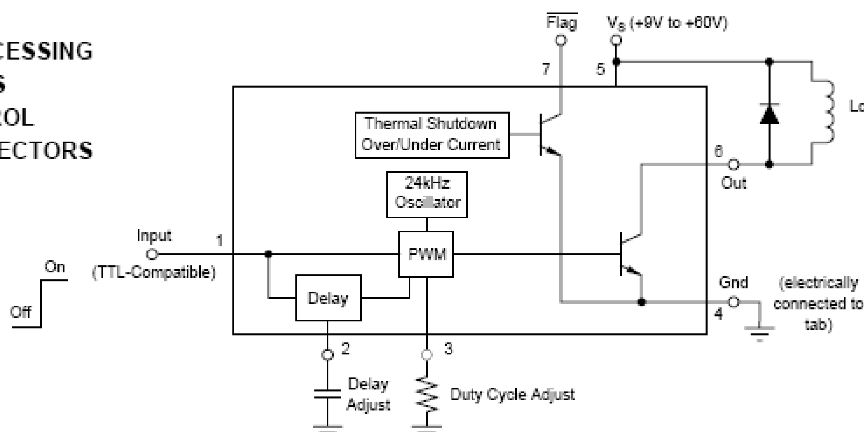
- FLUID AND GAS FLOW SYSTEMS
- INDUSTRIAL CONTROL
- FACTORY AUTOMATION
- PART HANDLERS
- PHOTOGRAPHIC PROCESSING
- ELECTRICAL HEATERS
- MOTOR SPEED CONTROL
- SOLENOID/COIL PROTECTORS
- MEDICAL ANALYZERS

DESCRIPTION

The DRV101 is a low-side power switch employing a pulse-width modulated (PWM) output. Its rugged design is optimized for driving electromechanical devices such as valves, solenoids, relays, actuators, and positioners. The DRV101 is also ideal for driving thermal devices such as heaters and lamps. PWM operation conserves power and reduces heat rise, resulting in higher reliability. In addition, adjustable PWM allows fine control of the power delivered to the load. Time from dc output to PWM output is externally adjustable.

The DRV101 can be set to provide a strong initial closure, automatically switching to a "soft" hold mode for power savings. Duty cycle can be controlled by a resistor, analog voltage, or digital-to-analog converter for versatility. A flag output indicates thermal shutdown and over/under current limit. A wide supply range allows use with a variety of actuators.

The DRV101 is available in 7-lead staggered TO-220 package and a 7-lead surface-mount DDPAK plastic power package. It is specified over the extended industrial temperature range, –40°C to +85°C.

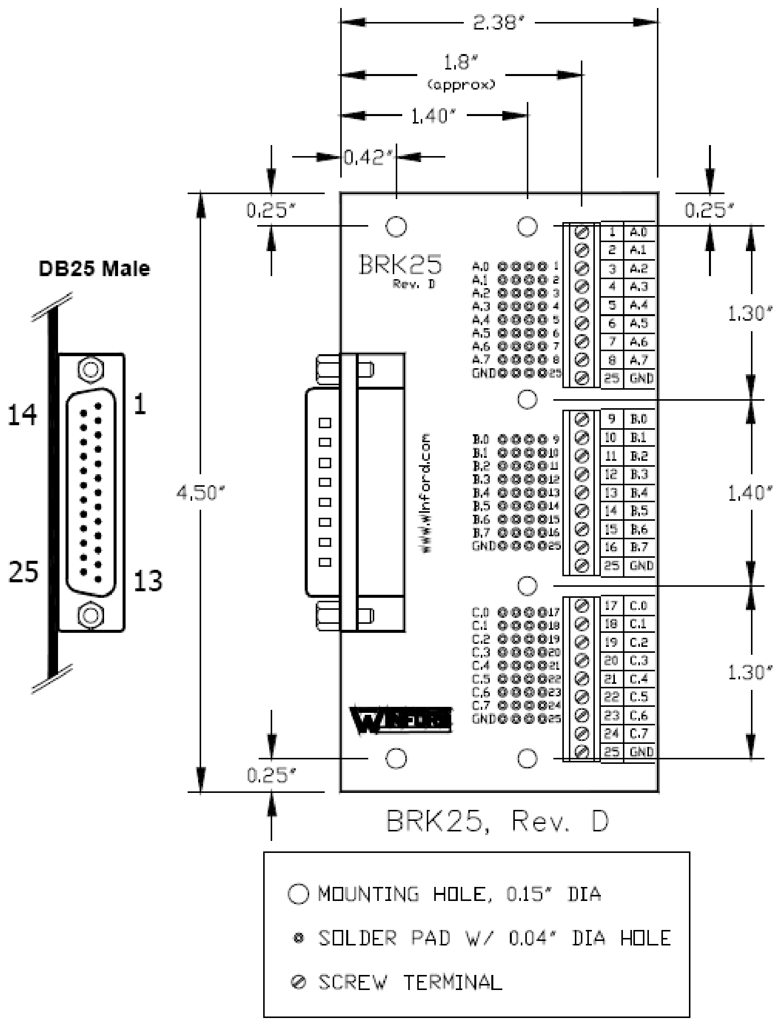


APPENDIX L

DATASHEET FOR DB-25 BREAKOUT BOARD USED TO OBTAIN FEEDBACK

WINFORD ENGINEERING, LLC
4169 Four Mile Road • Bay City, MI 48706
www.winford.com • sales@winford.com
1-877-634-2673 • FAX: 1-989-671-2941

BRK25 Rev D Physical Layout



WINFORD

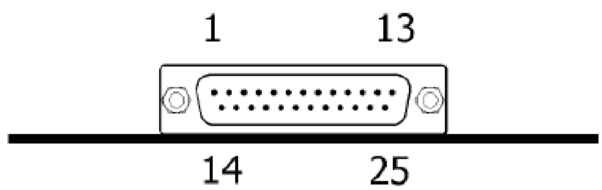
WINFORD ENGINEERING
4169 Four Mile Road • Bay City, MI 48706

www.winford.com • sales@winford.com

1-877-634-2673 • FAX: 1-989-671-2941

BRK25 Connection Reference

Male DB-25 Pinout



DB-25 Pin	*Winford Port / Bit **	*Standard Printer Port*
1	A.0	/Strobe
2	A.1	Data 0
3	A.2	Data 1
4	A.3	Data 2
5	A.4	Data 3
6	A.5	Data 4
7	A.6	Data 5
8	A.7	Data 6
9	B.0	Data 7
10	B.1	/Acknowledge
11	B.2	Busy
12	B.3	Paper End
13	B.4	Select

DB-25 Pin	*Winford Port / Bit **	*Standard Printer Port*
14	B.5	/Autofeed
15	B.6	/Error
16	B.7	/Initialize
17	C.0	/Select In
18	C.1	Ground
19	C.2	Ground
20	C.3	Ground
21	C.4	Ground
22	C.5	Ground
23	C.6	Ground
24	C.7	Ground
25**	GND	Ground

* The "Winford Port / Bit" column lists the I/O Port and bit number assignment on Winford Engineering's products. For example, A.0 indicates bit 0 of the first I/O port on the DB25 connector.

** Pin 25 is brought out to three screw terminals for convenience

APPENDIX M

DATASHEET FOR STEPPER MOTOR USED TO DRIVE BIN TRAY CONTAINING NAPKINS

Series 43000 Size 17 Linear Actuator

Salient Characteristics

Size 17: 43 mm (1.7") Hybrid Linear Actuator (1.8° Step Angle)						
Part No.	Captive	43H4(X)-V			43H6(X)-V	
	Non-captive	43F4(X)-V			43F6(X)-V	
	External Lin.	E43H4(X)-V			E43H6(X)-V	
Wiring		Bipolar			Unipolar**	
Operating voltage		2.33 VDC	5 VDC	12 VDC	5 VDC	12 VDC
Current/phase		1.5 A	700 mA	290 mA	700 mA	290 mA
Resistance/phase		1.56 Ω	7.2 Ω	41.5 Ω	7.2 Ω	41.5 Ω
Inductance/phase		1.9 mH	8.7 mH	54.0 mH	4.4 mH	27.0 mH
Power consumption		7 W				
Rotor inertia		37 gcm²				
Temperature rise		135°F Rise (75°C Rise)				
Weight		8.5 oz (241 g)				
Insulation resistance		20 MΩ				

** Unipolar drive gives approximately 30% less thrust than bipolar drive.

Linear Travel / Step					
Screw Ø .218" (5.54 mm) inches	mm	**Order Code I.D.**	Screw Ø .250" (6.35 mm) inches	mm	**Order Code I.D.**
.00012	.0030*	N	.00015625	.0039*	P
.00024	.0060*	K	.0003125	.0079*	A
.00048	.0121*	J	.000625	.0158*	B
.00096	.0243*	Q	.00125	.0317*	C
.00192	.0487*	R			

*Values truncated

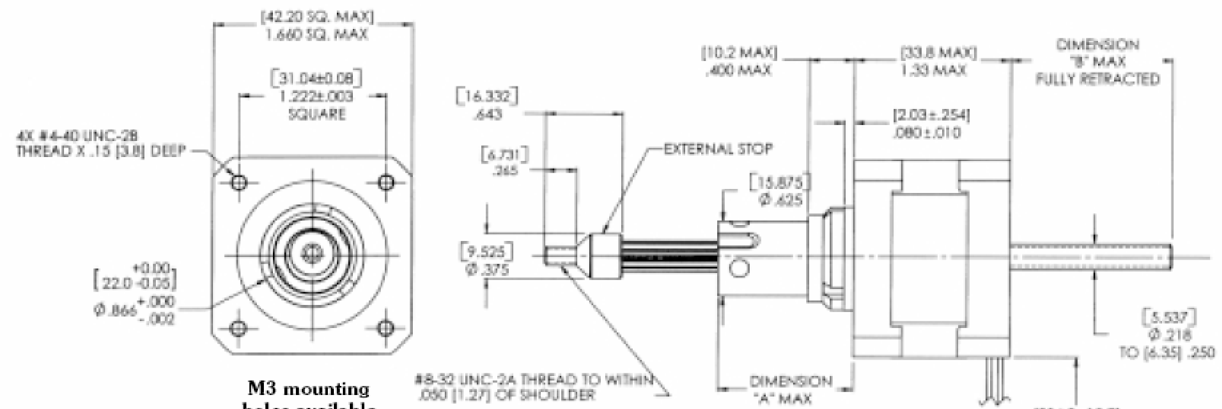
Standard motors are Class B rated for maximum temperature of 130°C. Also available, motors with high temperature capability windings up to 155°C.

Special drive considerations may be necessary when leaving shaft fully extended or fully retracted.

NOTE: See page 21 to identify product code information before placing order.

Series 43000 Size 17 Dimensional Drawings

Captive Shaft



M3 mounting holes available on request.

STROKE	DIM. "A"	DIM. "B"
0.500 (12.7)	0.78 (19.8)	0.16 (4.1)
0.750 (19.1)	1.03 (26.2)	0.41 (10.4)
1.00 (25.4)	1.28 (32.5)	0.66 (16.8)
1.250 (31.8)	1.53 (38.9)	0.91 (23.1)
1.500 (38.1)	1.78 (45.2)	1.16 (29.5)
2.00 (50.8)	2.28 (57.9)	1.66 (42.2)
2.500 (63.5)	2.78 (70.6)	2.16 (54.9)

Identifying Part Numbers for Orders

A standard HSI motor part number consists of 7 digits – XXXXX-VV.

- The **first and second** digits indicate the motor's series or size (in mm).
  Example: 35000 Series = 35 mm square
- The **third** digit or letter indicates the motor's step angle *Example: F = 1.8° non-captive; H = 1.8° captive or external linear; J = .9° non-captive; K = .9° captive or external linear*
  (NOTE: External linear actuators have the prefix "E" to distinguish them from the captive actuators; High Temperature actuators require the prefix "T")
- The **fourth** digit indicates the number of leads. (4 leads for bipolar; 6 leads for unipolar).
- The **fifth** digit indicates the travel per step (see chart) *Example: N = .00012" (.003048 mm)*
- The **sixth and seventh** digits indicate the motor's voltage. Standard voltages are 5 (05) and 12 (12) volt. We also have low inductance coils available for chopper drives. Custom voltages are also available.

SCREW LENGTH OPTIONS: For non-captive shaft and external linear motors various screw lengths are available to accommodate any travel requirement.

For assistance with a part number or a custom design, please us at call 203.756.7441.

Travel Per Step Code Letter or Digit	Linear Travel Per Step: Inches	Linear Travel Per Step: Millimeters	21000 Series Size 8 21 mm: Ø .138" (3.50) Screw	28000 Series Size 11 28 mm: Ø .1875" (4.76) Screw	35000 Series Size 14 35 mm: Ø .218" (5.54) Screw	35000 Series Size 14 35 mm: Ø .250" (6.35) Screw	43000 Series Size 17 43 mm: Ø .218" (5.54) Screw	43000 Series Size 17 43 mm: Ø .250" (6.35) Screw	57000 Series Size 23 57 mm: Ø .375" (9.53) Screw	87000 Series Size 34 87 mm: Ø .625" (15.88) Screw
U	0.00006	0.0015*	●				●			
V	0.000078*	0.00198*						●		
AA	0.000098*	0.0025	●							
N	0.00012	0.0030*	●		●		●			
7	0.000125	0.0031*		●						
P	0.00015625	0.0039*				●		●		
AB	0.00019*	0.005	●							
K	0.00024	0.0060*	●		●		●			
9	0.00025	0.0063*		●						
A	0.0003125	0.0079*				●		●	●	
AC	0.00039*	0.01	●							
S	0.0004167	0.0105*							●	
J	0.00048	0.0121*	●		●		●			
3	0.0005	0.0127		●					●	●
B	0.000625	0.0158*				●		●		●
AD	0.00078*	0.02	●							
T	0.0008333	0.0211*							●	
Q	0.00096	0.0243*			●		●			
1	0.001	0.0254		●					●	
C	0.00125	0.0317*				●		●		●
AE	0.00157*	0.04	●							
R	0.00192	0.0487*			●		●			
2	0.002	0.0508		●					●	
Y	0.0025	0.0635								●
Z	0.005	0.127								●

*Values truncated

APPENDIX N

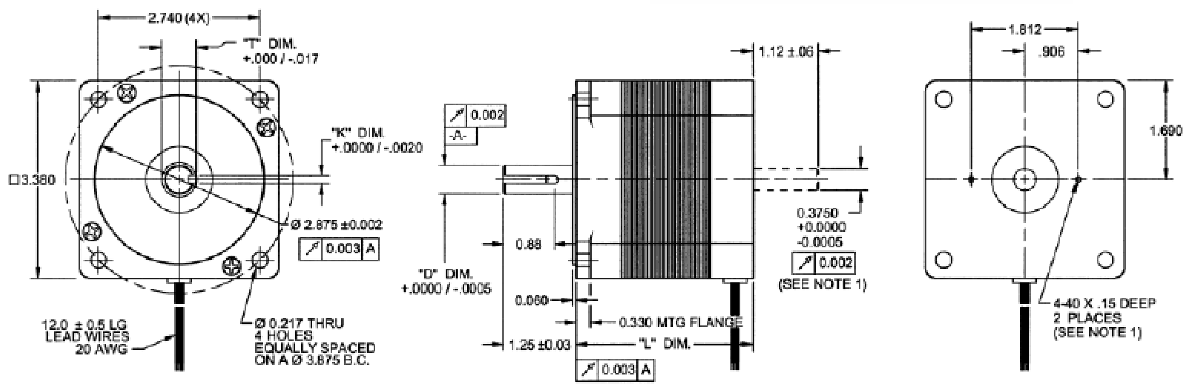
DATASHEET FOR STEPPER MOTOR USED TO DRIVE CONVEYOR SYSTEM

34Y Series - High Torque Step Motors

FEATURES

- NEMA Size 34
- 1.8° Step Angle
- High Torque - Up to 1700 oz-in
- High Step Accuracy and Resolution
- Low Vibration and Noise
- Can be Customized for
    - -Winding Current
    - -Shaft Options
    - -Cables and Connectors

DIMENSIONS



(All units are inches)

Note 1: Double shaft option only.

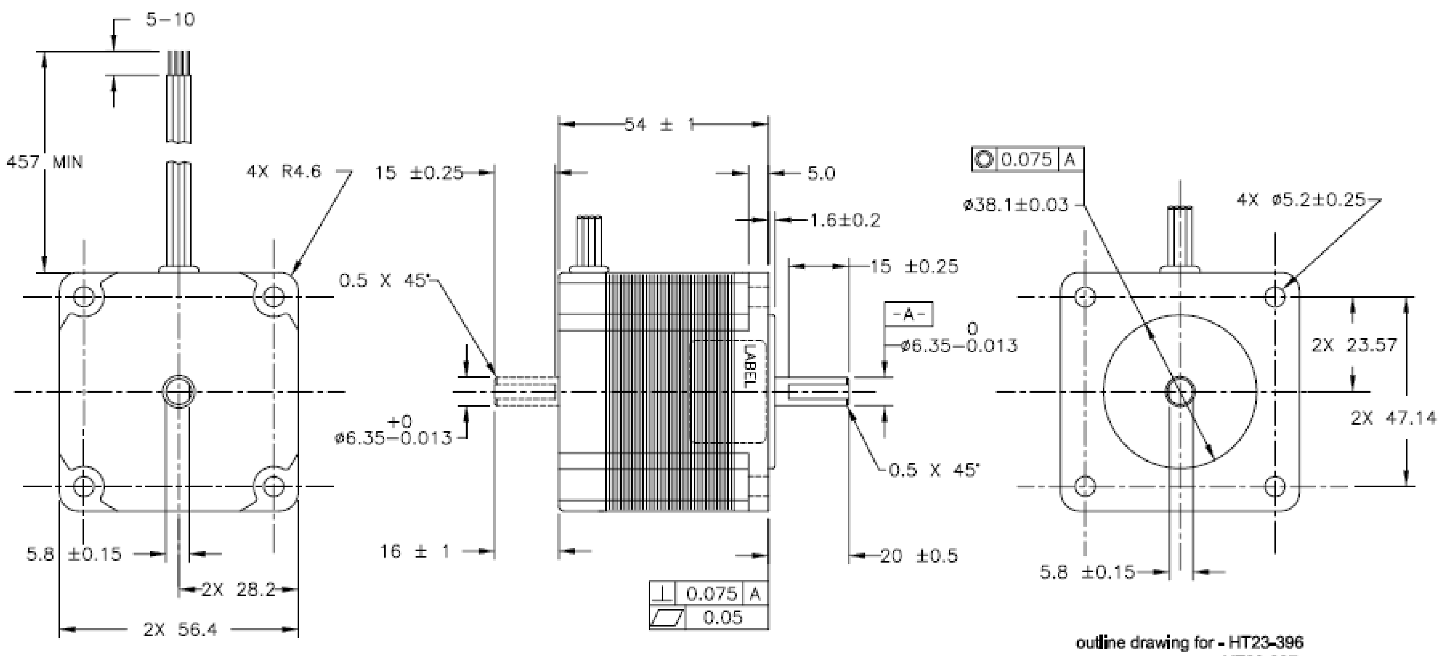
SPECIFICATIONS

The 34Y Series High Torque Step Motors offer a great value without sacrificing quality. These motors were designed to offer the highest possible torque while minimizing vibration and audible noise. A broad line of motor windings and stack lengths are available off-the-shelf, or the motors can be customized to fit your machine requirements. The standard 8-lead motors can be connected in all possible configurations: series, unipolar, or parallel, to allow the maximum flexibility for your application. Anaheim Automation can also customize the winding to perfectly match your voltage, current, and maximum operating speed.

Model #	NEMA Size	Bipolar Torque (oz-in)	Series Current (A)	Unipolar Current (A)	Parallel Current (A)	Unipolar Inductance (mH)	Rotor Inertia (oz-in-sec^2)	Shaft Diameter (in)	# of Lead Wires	Weight (lbs)	L Length (in)
34Y004S-LW8	34	590	1.4	2.0	2.8	11.3	0.014	0.5	8	3.8	2.56
34Y104S-LW8	34	650	1.4	2.0	2.8	14.2	0.020	0.5	8	5.1	3.15
34Y112S-LW8	34	650	4.3	6.1	8.6	1.7	0.020	0.5	8	5.1	3.15
34Y207S-LW8	34	1200	2.5	3.5	5.0	8.3	0.038	0.5	8	8.4	4.65
34Y214S-LW8	34	1200	5.0	7.1	10.0	2.0	0.038	0.5	8	8.4	4.65
34Y307S-LW8	34	1700	2.5	3.5	5.0	12.5	0.057	0.625	8	12.0	6.14
34Y314S-LW8	34	1700	5.0	7.0	9.9	3.5	0.057	0.625	8	12.0	6.14

Notes: LW8 is for 8 leadwires, other leadwire options are available. All Shafts have a flat unless otherwise noted. The 7th character "S" denotes a single shaft, use "D" for double shaft. Double shafts include encoder mounting provisions. Custom leadwires, cables, connectors, and windings are available upon request.

Notes
1.For double shaft version add D to end of part Number e.g. HT23-397D



outline drawing for - HT23-396
- HT23-397
- HT23-398

TOLERANCES
DECIMALS: MM (INCH)
X.XXX= ±0.01 (.005)
X.XX = ±0.10 (.010)
X.X = ±0.20 (.020)
ANGLES:
MACH. = ±.5°
CHAM. = ±5°
COMPUTER DATA BASE DRAWING

THIRD ANGLE PROJECTION

APPROVALS	DATE
DRAWN	
CHECKED	
APPROVED	

APPLIED MOTION PRODUCTS, INC.

STEP MOTOR OUTLINE

B DWG NO. 23HT54 REV

SCALE: FULL SHEET 1 OF 1

APPENDIX O

DATASHEET FOR SERVO MOTOR USED IN FINGER MECHANISM USED BY FOLDING MECHANISM-3

599 Menlo Drive, Suite 100
Rocklin, California 95765, USA
Office: (916) 624-8333
Fax: (916) 624-8003

General: info@parallaxinc.com
Technical: support@parallax.com
Web Site: www.parallax.com
Educational: http://www.parallax.com/html_pages/edu/

Standard Servo (#900-00005)

General Information

The Parallax standard servo is ideal for robotics and basic movement projects. These servos will allow a movement range of 0 to 180 degrees. The Parallax servo output gear shaft is a standard Futaba configuration. The servo is manufactured by Futaba specifically for Parallax.

Technical Specifications

> Power 6vdc max

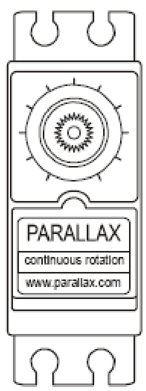
> Speed 0 deg to 180 deg in 1.5 seconds on average

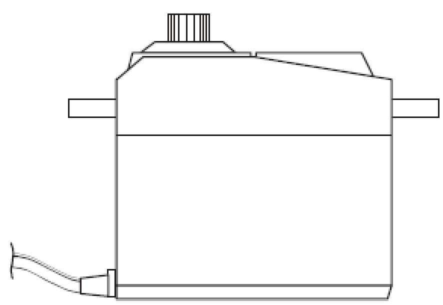
> Weight 45.0 grams/1.59oz

> Torque 3.40 kg-cm/47oz-in

> Size mm (L x W x H) 40.5x20.0x38.0

> Size in (L x W x H) 1.60x.79x1.50





Motor Control from a BASIC Stamp

Parallax (www.parallax.com) publishes many circuits and examples to control servos. Most of these examples are available for download from our web site. On www.parallaxinc.com type in "servo" and you'll find example codes below.

APPENDIX P

DATASHEET FOR STEPPER MOTOR DRIVER UNIT

CK1406 - BI-POLAR STEPPER MOTOR DRIVER

This kit will drive a bi-polar stepper motor driver using externally supplied 5V levels for steping and direction. These usually come from software running in a computer.

Google 'Stepper Motor Software' and will see a range of controller freeware available. Also go to

http://www.metalworking.com/
http://www.kellyware.com/
http://www.thegallows.com/stepster.htm

Construction. Follow the overlay on the PCB. Add the lowest height components first – the resistors and diodes. Note there are seven links to add to the board. For the six short links use offcuts from the resistor legs. For the longer link we have provided a 1 ½" of tinned copper wire. There are 4 pins you can place in the DIR and STEP positions if you wish to use the pins.

We have assumed that the operation of stepper motors is known to you.

Operation. Attach a bi-polar stepper motor to the TO MOTOR terminals. Power to the Kit can be the same or different depending on the stepper motor being driven.

The **DIR**ection and **STEP** inputs are opto-isolated by the 4N25 IC's.

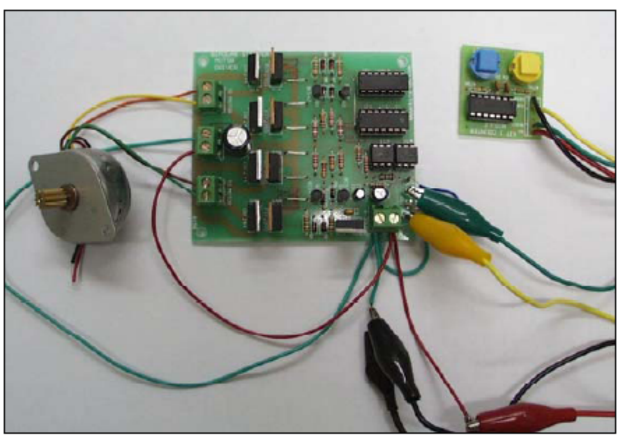
DIRection. This is controlled by a 0V or 5V applied between the pins.

STEP. A 5V to 0V transistion between the pins will step the motor one position. The step direction will be according to the voltage applied to the DIRection pins.

COMPONENTS		
Resistors 5%, 1/4W, carbon		
150R brown green brown	R9 R10 R15 R16	4
1K brown black red	R1 R2	2
2K2 red red red	R7 R8 R13 R14	4
10K brown black orange	R3 R4	2
12K brown red orange	R5 R6 R11 R12	4
100uF/35V ecap	C1	1
10uF mini ecap	C2	1
100uF/63V ecap	C4	1
1N4148 diode	D1 – D8	8
104 mono	C3	1
Pins		4
4013 IC	IC1	1
4030 IC	IC2	1
4N25 IC	IC3 IC4	2
7805 IC	IC5	1
BC547B transistors	Q1 Q2 Q3 Q4	4
IRFZ44	Q1B Q2B Q3B Q4B	4
MTP2955	Q1A Q2A Q3A Q4A	4
6 pin IC socket		2
14 pin IC socket		2
2 pole terminal block		4
tinned copper wire		1 1/2"
K158 PCB		1

All the power inputs were connected together. The CLOCK was connected to STEP, and the RESET was connected to DIRection. Pushing the CLOCK button then advanced the motor one notch. Pressing CLOCK with the RESET button also depressed and pressed down advanced the motor one notch the other way.

Software does the same thing but using a PC.



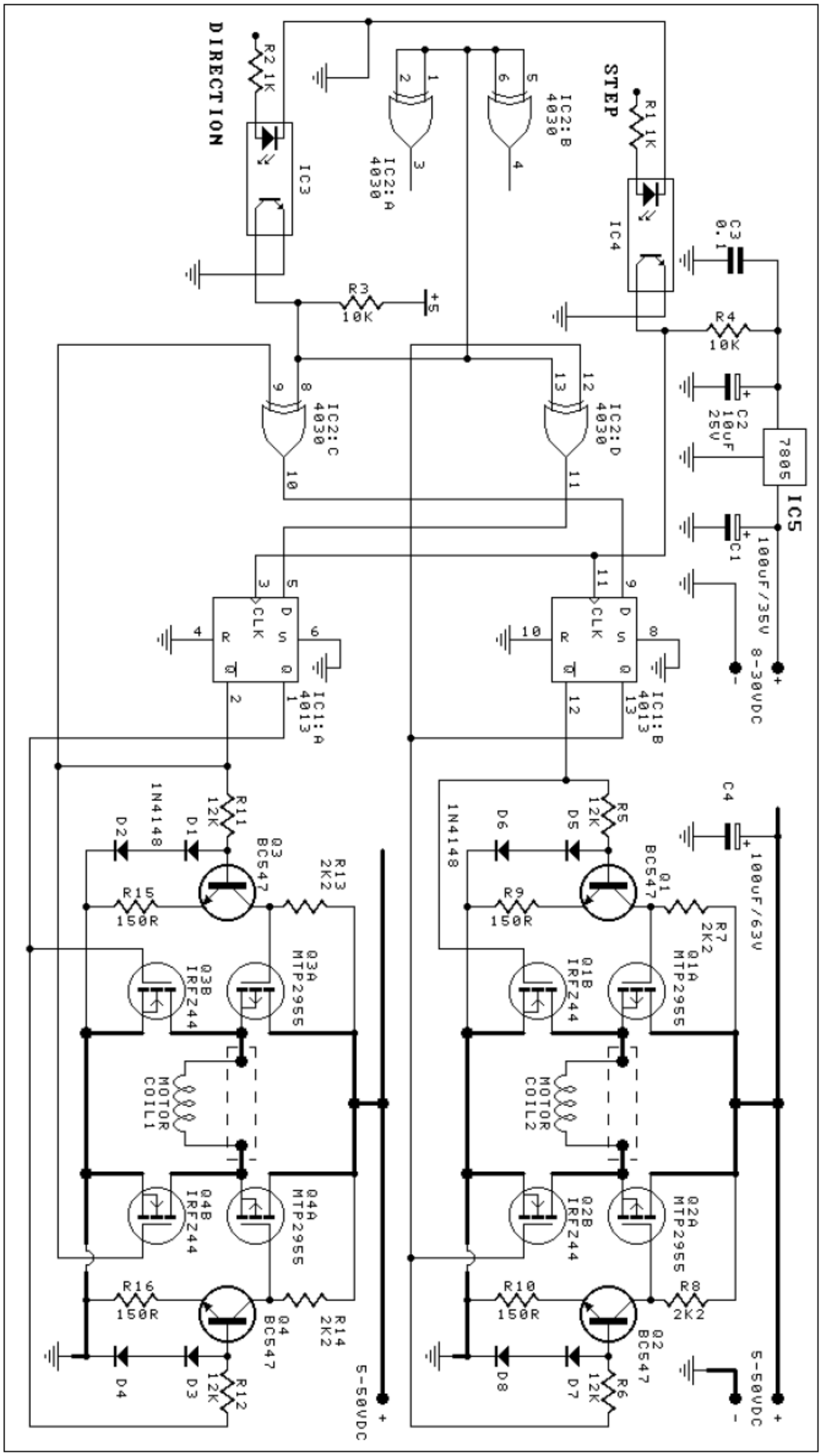
This kit is based on a kit from Oatley Electronics, Australia. It is adapted here with permission.

(Documentation march 30, 2003.)

- - - - - - - - - - - - - - - - -

The unipolar stepper motor is connected as a bipolar motor (the 2 center wires of the 6 wire motor are unused.) 9V was used. The STEP and DIRection negative input pins were tied together and connected to system ground.

CK1406 - BI-POLAR STEPPER MOTOR DRIVER

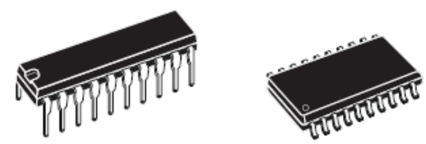


APPENDIX Q

DATASHEET USED TO AMPLIFY PWM SIGNAL USED BY THE IR SENSOR UNIT

STEPPER MOTOR CONTROLLERS

- NORMAL/WAVE DRIVE
- HALF/FULL STEP MODES
- CLOCKWISE/ANTICLOCKWISE DIRECTION
- SWITCHMODE LOAD CURRENT REGULATION
- PROGRAMMABLE LOAD CURRENT
- FEW EXTERNAL COMPONENTS
- RESET INPUT & HOME OUTPUT
- ENABLE INPUT



DIP20 SO20

ORDERING NUMBERS : L297/1 (DIP20)
L297D (SO20)

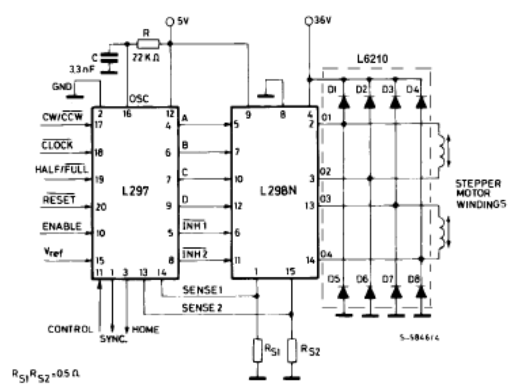
DESCRIPTION

The L297 Stepper Motor Controller IC generates four phase drive signals for two phase bipolar and four phase unipolar step motors in microcomputer-controlled applications. The motor can be driven in half step, normal and wawe drive modes and on-chip PWM chopper circuits permit switch-mode control of the current in the windings. A feature of this device is that it requires only clock, direction and mode input signals. Since the phase are generated internally the burden on the microprocessor, and the programmer, is greatly reduced. Mounted in DIP20 and SO20 packages, the L297 can be used with monolithic bridge drives such as the L298N or L293E, or with discrete transistors and darlingtons.

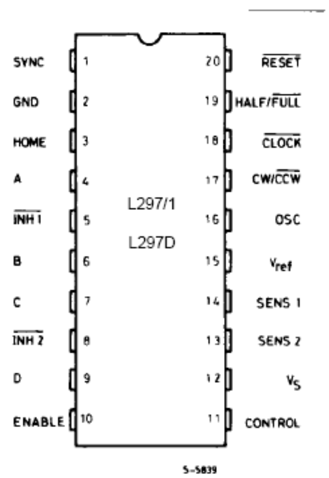
ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_s	Supply voltage	10	V
V_i	Input signals	7	V
P_{tot}	Total power dissipation (T_{amb} = 70°C)	1	W
T_{stg}, T_j	Storage and junction temperature	-40 to + 150	°C

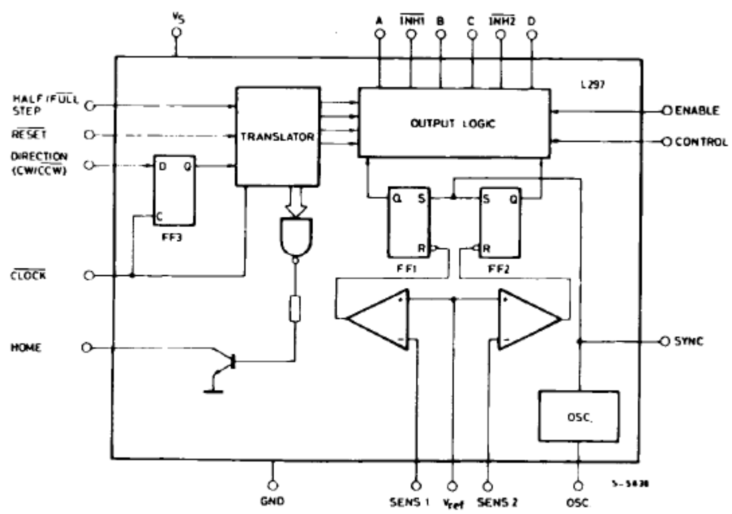
TWO PHASE BIPOLAR STEPPER MOTOR CONTROL CIRCUIT



PIN CONNECTION (Top view)



BLOCK DIAGRAM (L297/1 - L297D)



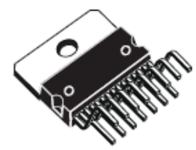
APPENDIX R

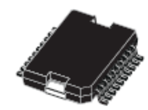
DATASHEET FOR H-BRIDGE DRIVER UNITS USED BY MICROCONTROLLER-4

L298

DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)


Multiwatt15

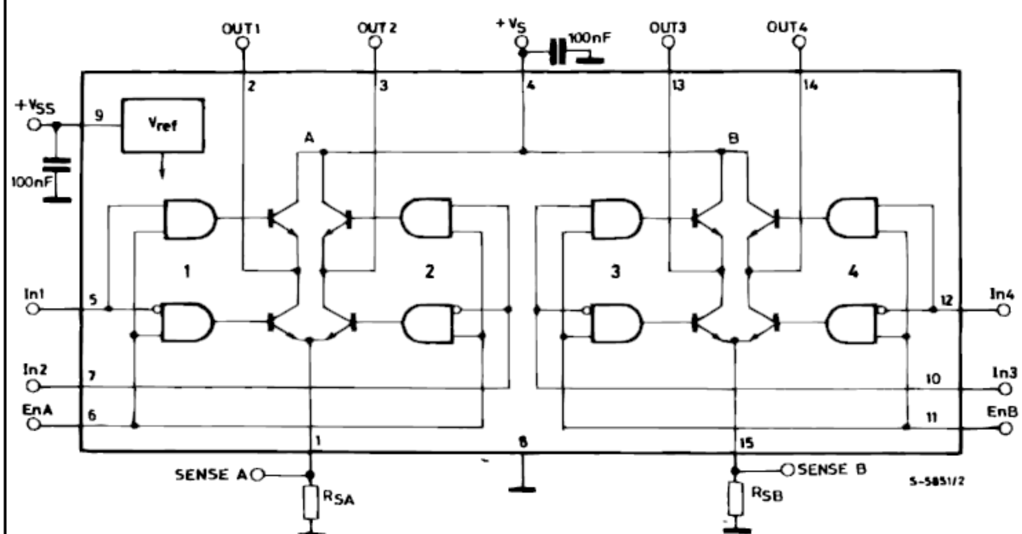

PowerSO20

ORDERING NUMBERS : L298N (Multiwatt Vert.)
L298HN (Multiwatt Horiz.)
L298P (PowerSO20)

DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

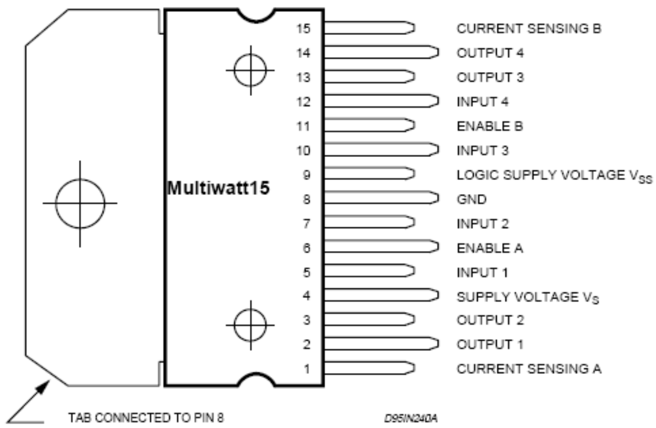
BLOCK DIAGRAM

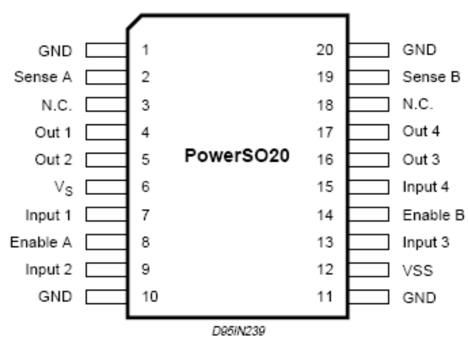


ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_S	Power Supply	50	V
V_{SS}	Logic Supply Voltage	7	V
V_I, V_{en}	Input and Enable Voltage	–0.3 to 7	V
I_O	Peak Output Current (each Channel) – Non Repetitive (t = 100µs) –Repetitive (80% on –20% off; t_{on} = 10ms) –DC Operation	 3 2.5 2	 A A A
V_{sens}	Sensing Voltage	–1 to 2.3	V
P_{tot}	Total Power Dissipation (T_{case} = 75°C)	25	W
T_{op}	Junction Operating Temperature	–25 to 130	°C
T_{stg}, T_j	Storage and Junction Temperature	–40 to 150	°C

PIN CONNECTIONS (top view)





THERMAL DATA

Symbol	Parameter		PowerSO20	Multiwatt15	Unit
$R_{th\ j\text{-}case}$	Thermal Resistance Junction-case	Max.	–	3	°C/W
$R_{th\ j\text{-}amb}$	Thermal Resistance Junction-ambient	Max.	13 (*)	35	°C/W

(*) Mounted on aluminum substrate

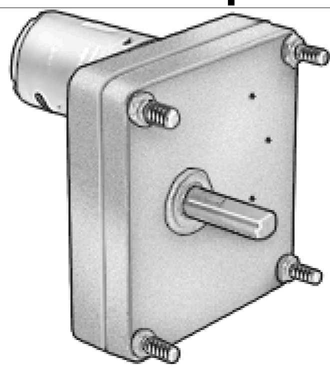
APPENDIX S

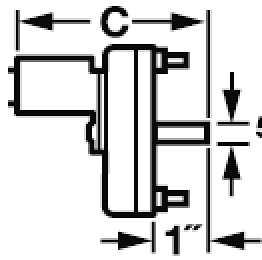
DATASHEET FOR DC GEARMOTORS USED TO POWER ARM MECHANISM USED BY THE NAPKIN PICKUP MECHANISM

DC Gearmotors, Motors & Motor Mounts

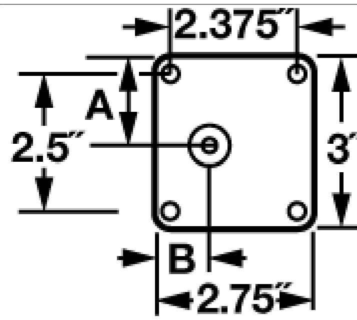
For information about electric motors, see page 910 . For speed controllers, see page 829 .

Subfractional-hp DC Gearmotors





Side View



Front View

With less than 1/100 hp, these permanent magnet gearmotors are ideal for use in small spaces such as business machines, appliances, and valve actuators. A gearmotor consists of a motor and fan matched with a geared speed reducer to lessen speed while increasing torque. Shaft has a flat to accept set screws for easy connection to your equipment.

Motors are brush style and have two terminals for electrical connection. Rotation is clockwise when facing the shaft end. To reverse rotation, switch the lead wires (follow the included wiring diagram). For speed controllers, see 7729K on page 829 . Housing is die cast zinc with bronze sleeve bearings. Gears are iron and Delrin. Motor face has four 10-32 threaded mounting studs.

rpm	Torque, in.-lbs.	(A)	(B)	(C)	Full Load Amps		Each
12 VDC							
.6	50	1.43"	0.91"	3.45"	0.12	6409K11	$45.76
1.3	50	1.43"	0.91"	3.65"	0.45	6409K12	45.76
4	40	1.43"	0.91"	3.45"	0.68	6409K13	42.03
8	40	1.43"	0.91"	4.17"	0.73	6409K14	42.03
12	40	1.43"	0.91"	3.65"	1.3	6409K15	42.03
16	25	1.43"	0.91"	3.65"	1.4	6409K16	37.42
25	20	1.43"	0.91"	3.65"	1.3	6409K17	37.42
50	10	2.33"	1.38"	3.65"	1.2	6409K18	37.42
24 VDC							
1.2	50	1.43"	0.91"	3.45"	0.14	6409K21	45.76
4	50	1.43"	0.91"	3.45"	0.38	6409K22	42.03
8	34	1.43"	0.91"	3.65"	0.45	6409K23	42.03
12	24	1.43"	0.91"	3.65"	0.42	6409K24	42.03
17	17	1.43"	0.91"	3.65"	0.45	6409K25	37.42
25	50	1.43"	0.91"	4.17"	1.08	6409K26	37.42
47	28	2.33"	1.37"	4.15"	1.08	6409K27	37.42

APPENDIX T

DATASHEET FOR LS 7084 USED BY MICROCONTROLLER - 4

LSI/CSI

LS7083/7084

LSI Computer Systems, Inc. 1235 Walt Whitman Road, Melville, NY 11747 (631) 271-0400 FAX (631) 271-0405

QUADRATURE CLOCK CONVERTER

January 2003

FEATURES:
- x1 and x4 mode selection
- Up to 16MHz output clock frequency
- Programmable output clock pulse width
- On-chip filtering of inputs for optical or magnetic encoder applications.
- TTL and CMOS compatible I/Os
- +4.5V to +10V operation (V_{DD} - V_{SS})
- LS7083, LS7084 (DIP); LS7083-S, LS7084-S (SOIC) - See Figure 1

DESCRIPTION:
The LS7083 and LS7084 are CMOS quadrature clock converters. Quadrature clocks derived from optical or magnetic encoders, when applied to the A and B inputs of the LS7083/ LS7084, are converted to strings of Up Clocks and Down Clocks (LS7083) or to a Clock and an Up/Down direction control (LS7084). These outputs can be interfaced directly with standard Up/Down counters for direction and position sensing of the encoder.

INPUT/OUTPUT DESCRIPTION:

RBIAS (Pin 1)
Input for external component connection. A resistor connected between this input and Vss adjusts the output clock pulse width (Tow). For proper operation, the output clock pulse width must be less than or equal to the A, B pulse separation ($T_{OW} \leq T_{PS}$).

V_{DD} (Pin 2)
Supply Voltage positive terminal.

V_{SS} (Pin 3)
Supply Voltage negative terminal.

A (Pin 4)
Quadrature Clock Input A. This input has a filter circuit to validate input logic level and eliminate encoder dither.

B (Pin 5)
Quadrature Clock Input B. This input has a filter circuit identical to input A.

x4/$\overline{x1}$ (Pin 6)
This input selects between x1 and x4 modes of operation. A high-level selects x4 mode and a low-level selects the x1 mode. In x4 mode, an output pulse is generated for every transition at either A or B input. In x1 mode, an output pulse is generated in one combined A/B input cycle.
(See Figure 2.)

PIN ASSIGNMENT - TOP VIEW

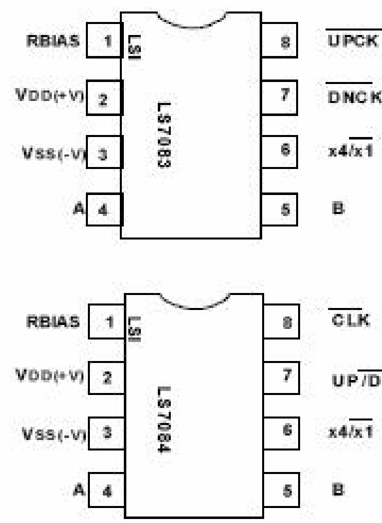


FIGURE 1

LS7083 - $\overline{DNCK}$ (Pin 7)
In LS7083, this is the DOWN Clock Output. This output consists of low-going pulses generated when A input lags the B input.

LS7084 - UP/$\overline{DN}$ (Pin 7)
In LS7084, this is the count direction indication output. When A input leads the B input, the UP/$\overline{DN}$ output goes high indicating that the count direction is UP. When A input lags the B input, UP/$\overline{DN}$ output goes low, indicating that the count direction is DOWN.

LS7083 - $\overline{UPCK}$ (Pin 8)
In LS7083, this is the UP Clock output. This output consists of low-going pulses generated when A input leads the B input.

LS7084 - $\overline{CLK}$ (Pin 8)
In LS7084, this is the combined UP Clock and DOWN Clock output. The count direction at any instant is indicated by the UP/$\overline{DN}$ output (Pin 7).

NOTE: For the LS7084, the timing of $\overline{CLK}$ and UP/$\overline{DN}$ requires that the counter interfacing with LS7084 counts on the rising edge of the $\overline{CLK}$ pulses.

7083/84-012703-1

APPENDIX U

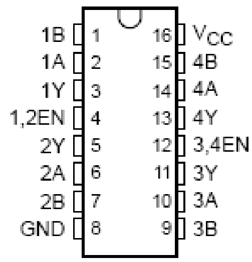
DATASHEET FOR LINE DRIVER MC3486P USED BY MICROCONTROLLER-4

MC3486
QUADRUPLE DIFFERENTIAL LINE RECEIVER WITH 3-STATE OUTPUTS

SLLS097B – JUNE 1980 – REVISED MAY 1995

- Meets or Exceeds the Requirements of ANSI Standards EIA/TIA-422-B and EIA/TIA-423-B and ITU Recommendations V.10 and V.11
- 3-State, TTL-Compatible Outputs
- Fast Transition Times
- Operates From Single 5-V Supply
- Designed to Be Interchangeable With Motorola™ MC3486

D OR N PACKAGE
(TOP VIEW)



description

The MC3486 is a monolithic quadruple differential line receiver designed to meet the specifications of ANSI Standards EIA/TIA-422-B and EIA/TIA-423-B and ITU Recommendations V.10 and V.11. The MC3486 offers four independent differential-input line receivers that have TTL-compatible outputs. The outputs utilize 3-state circuitry to provide a high-impedance state at any output when the appropriate output enable is at a low logic level.

The MC3486 is designed for optimum performance when used with the MC3487 quadruple differential line driver. It is supplied in a 16-pin package and operates from a single 5-V supply.

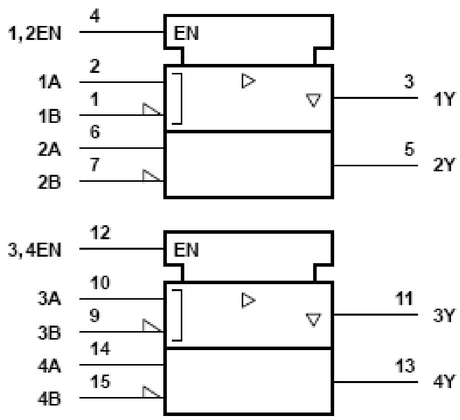
The MC3486 is characterized for operation from 0°C to 70°C.

FUNCTION TABLE
(each receiver)

DIFFERENTIAL INPUTS A–B	ENABLE	OUTPUT Y
$V_{ID} \le 0.2$ V	H	H
-0.2 V < V_{ID} < 0.2 V	H	?
$V_{ID} \le -0.2$ V	H	L
Irrelevant	L	Z
Open	H	?

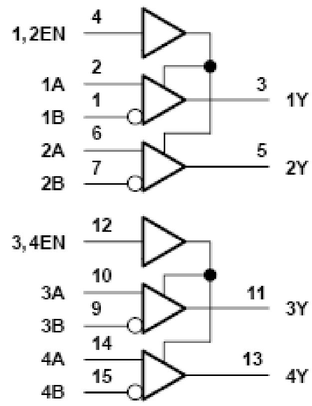
H = high level, L = low level, Z = high impedance (off), ? = indeterminate

logic symbol†

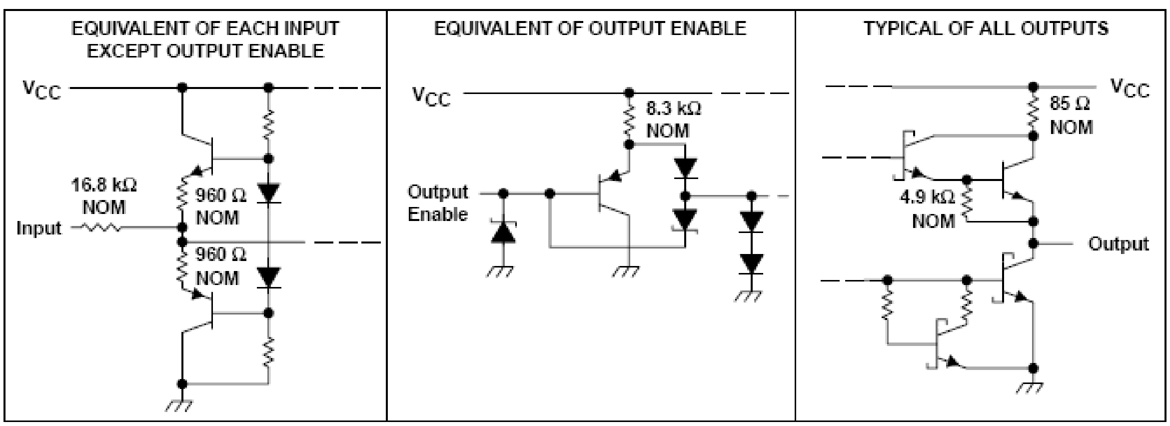


† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

logic diagram (positive logic)



schematics of inputs and outputs



APPENDIX V

DATASHEET FOR CD4516 COUNTER USED BY MICROCONTROLLER - 4

Jameco Part Number 13549

Texas Instruments

CD4510B, CD4516B Types

CMOS Presettable Up/Down Counters

High-Voltage Types (20-Volt Rating)

CD4510B – – – BCD Type

CD4516B – – – Binary Type

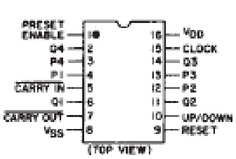
■ CD4510B Presettable BCD Up/Down Counter and the CD4516 Presettable Binary Up/Down Counter consist of four synchronously clocked D-type flip-flops (with a gating structure to provide T-type flip-flop capability) connected as counters. These counters can be cleared by a high level on the RESET line, and can be preset to any binary number present on the jam inputs by a high level on the PRESET ENABLE line. The CD4510B will count out of non-BCD counter states in a maximum of two clock pulses in the up mode, and a maximum of four clock pulses in the down mode.

If the CARRY-IN input is held low, the counter advances up or down on each positive-going clock transition. Synchronous cascading is accomplished by connecting all clock inputs in parallel and connecting the CARRY-OUT of a less significant stage to the CARRY-IN of a more significant stage.

The CD4510B and CD4516B can be cascaded in the ripple mode by connecting the CARRY-OUT to the clock of the next stage. If the UP/DOWN input changes during a terminal count, the CARRY-OUT must be gated with the clock, and the UP/DOWN input must change while the clock is high. This method provides a clean clock signal to the subsequent counting stage. (See Fig. 15).

These devices are similar to types MC14510 and MC14516.

The CD4510B and CD4516B Series types are supplied in 16-lead hermetic dual-in-line ceramic packages (D and F suffixes), 16-lead dual-in-line plastic packages (E suffix), and in chip form (H suffix).

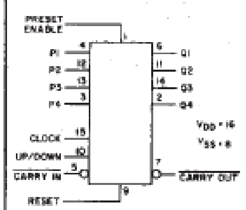


92CS-27015

CD4510B, CD4516B TERMINAL ASSIGNMENT

Features:

- **Medium-speed operation – f_{CL} = 8 MHz typ. at 10 V**
- **Synchronous internal carry propagation**
- **Reset and Preset capability**
- **100% tested for quiescent current at 20 V**
- **5-V, 10-V, and 15-V parametric ratings**
- **Standardized symmetrical output characteristics**
- **Maximum input current of 1 µA at 18 V over full package temperature range; 100 nA at 18 V and 25°C**
- **Noise margin (full package-temperature range): 1 V at V_{DD} = 5 V; 2 V at V_{DD} = 10 V; 2.5 V at V_{DD} = 15 V**
- **Meets all requirements of JEDEC Tentative Standard No. 13B, "Standard Specifications for Description of 'B' Series CMOS Devices"**



CD4510B, CD4516B FUNCTIONAL DIAGRAM

92CS-24824

Applications:

- **Up/Down difference counting**
- **Multistage synchronous counting**
- **Multistage ripple counting**
- **Synchronous frequency dividers**

3

COMMERCIAL CMOS HIGH VOLTAGE ICs

OPERATING CONDITIONS AT T_A = 25°C, Unless Otherwise Specified

For maximum reliability, nominal operating conditions should be selected so that operation is always within the following ranges.

Characteristic	V_{DD}	Min.	Max.	Units
Supply Voltage Range (At T_A = Full Package-Temperature Range)		3	18	V
Clock Pulse Width, t_W	5 10 15	150 75 60	– – –	ns
Clock Input Frequency, f_{CL}	5 10 15	– – –	2 4 5.5	MHz
Preset Enable or Reset Removal Time•	5 10 15	150 80 60	– – –	ns
Clock Rise and Fall Time, t_rCL, t_fCL *	5 10 15	– – –	15 5 5	µs
Carry-In Setup Time, t_S	5 10 15	130 60 45	– – –	ns
Up-Down Setup Time, t_S	5 10 15	360 160 110	– – –	ns
Preset Enable or Reset Pulse Width, t_W	5 10 15	220 100 75	– – –	ns

•Time required after the falling edge of the reset or preset enable inputs before the rising edge of the clock will trigger the counter (similar to setup time).

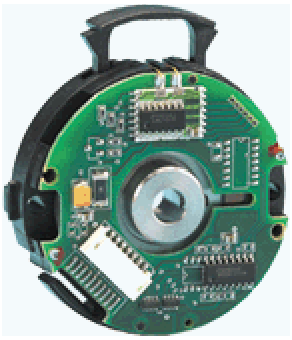
*If more than one unit is cascaded in the parallel clocked application, t_rCL should be made less than or equal to the sum of the fixed propagation delay at 15 pF and the transition time of the carry output driving stage for the estimated capacitive load.

APPENDIX W

DATASHEET FOR RM21 ENCODER USED BY MICROCONTROLLER - 4

RM21 Modular 2.1 Inch Encoder

Features:



- Self-aligning. Self-centering. Self-gapping.
- Frequency response up to 200KHz (all channels).
- A,B, and Index with complements.
- Positive lock on cover.
- Wide range of standard **no cost** options including: hub bore diameters, terminations, output formats and thru-hole cover.
- Robust .060" [1,53mm] chrome on glass code disk.

- Meets NEMA ICS-16 standard.

General Description:

The RM21 Series is a highly reliable, low cost bearingless (modular) encoder built using SMT technology.

The 1-2-3 set-up simplifies troublesome features of typical modular encoders. With a single push on the slide/lock mechanism, the RM21 is aligned, centered and gapped for maximum performance.

No adjustments are required and no mechanical rubbing exists once installed. The cover has an integral positive locking mechanism to eliminate mounting hardware. Sensors are set up differentially under a single LED light source.

Specifications:

(Subject to change without notice)[Metric measurements in square brackets]

MECHANICAL:

Outer Diameter:	**2.1 inches [53,34mm]**
Height:	**.84 inches [21,34mm]**
Moment of Inertia:	**17.3 X 10E-05 oz-in sec^2 (max) [12,22 g-cm^2]**
Weight:	**2.0 oz max [57g]**
Slew Speed:	**Resolution dependent**
Base & Cover Material:	**Glass filled polycarbonate**
Code Disc Material:	**Chrome on glass (.060")[1,53mm]**
Hub Material:	**Stainless steel**
Max End Play:	**+.010 -.040 inches [+0,25mm -1,02mm](non operational) +.005 -.040 inches**

[+0,12mm -1,02mm] (2500 count & above)

ELECTRICAL:

Output:	Square-wave, two-channel quadrature with index (gating options available)
Input Power:	5 VDC ±5%
Output Format:	TI AM26LS31CD or LM2901
Output Frequency:	200 KHz
Flutter:	1% max

ENVIRONMENTAL:

Operating Temp:	-10°C to +100°C
Storage Temp:	-30°C to +110°C
Shock:	50 G's for 11ms duration
Vibration:	5-2000 Hz @ 10 G's
IP Rating:	IP 40
Humidity:	90% relative (non condensing)

RESOLUTIONS:

Range:	5 to 2048 PPR (Pulses per Revolution)
Currently available:	5, 10, 40, 50, 60, 75, 100, 104,120 125, 150, 180, 192, 200, 250, 254 256, 288, 300, 360, 400, 500, 508 512, 576, 600, 625, 635, 720, 800 900, 1000, 1024, 1140, 1250, 1270 1500, 1800, 1885, 2000, 2048

VO output option is not available in resolutions below 250 PPR.

HUB SIZE: (shown in inches)

Specify	Hub Bore +.0004/-.0000 [+0,010/-0,000]	
3/16	.1873	[4,757mm]
1/4	.2498	[6,345mm]
1/4+	.2501	[6,353mm]
5/16	.3123	[7,932mm]
3/8	.3748	[9,520mm]
3/8+	.3751	[9,528mm]
1/2	.4998	[12,695mm]
1/2+	.5001	[12,703mm]
4MM	.1575	[4,000mm]
5MM	.1969	[5,000mm]
6MM	.2362	[6,000mm]
8MM	.3150	[8,000mm]
10MM	.3937	[10,000mm]
12MM	.4724	[12,000mm]

Encoders ordered with metric shaft size will receive metric mounting hardware.

TERMINATION OPTIONS:

Round Cable

Pin	CA18 Function	Wire Color
1	+VCC	Red
2	GND	Blk
3	CH A	Wht
4	CH A NOT	Yel
5	CH B	Grn
6	CH B NOT	Blu
7	INDEX	Orn
8	INDEX NOT	Brn
	SHIELD	Drain

Ribbon Cable

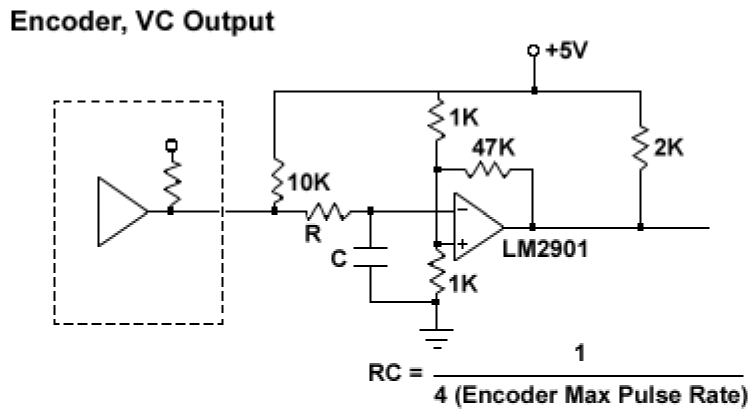
Pin	R24 Function	R24 (VC/VO) Function
1	CH A	CH A
2	VCC	+5 VDC
3	GND	GND
4	N/A	N/A
5	CH A NOT	N/A
6	CH A	N/A
7	CH B NOT	N/A
8	CH B	CH B
9	INDEX NOT	N/A
10	INDEX	INDEX

Standard shielded round cable (CA18) is supplied with connector P/N AMP 103971-7 or equivalent. This connector mates with any .025 [0,635mm] sq. non-polarized single row header (.100 [2,54mm] centers) or may be intalled into single row latching shroud (AMP P/N 103680-5).

OUTPUT INTERFACES:

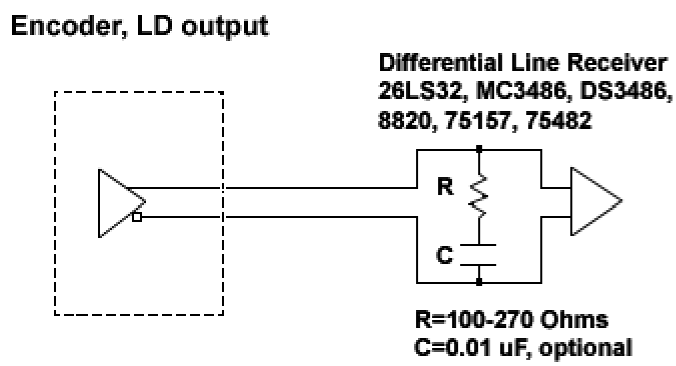
VC (VOLTAGE COMPARATOR)
VO (VC W/OPEN COLLECTOR)

LM2901 (VC with pullup resistor)
No complementary signals available
5 VDC power input TTL compatible
Output sinks 6mA
Nominal power requirements 75mA for VC and 125mA for VO option



LD (LINE DRIVER)

TI AM26LS31CD typical output
26LS32 typical line receiver suggested for user interface
Output sinks and sources 20mA
Nominal power requirements 150mA

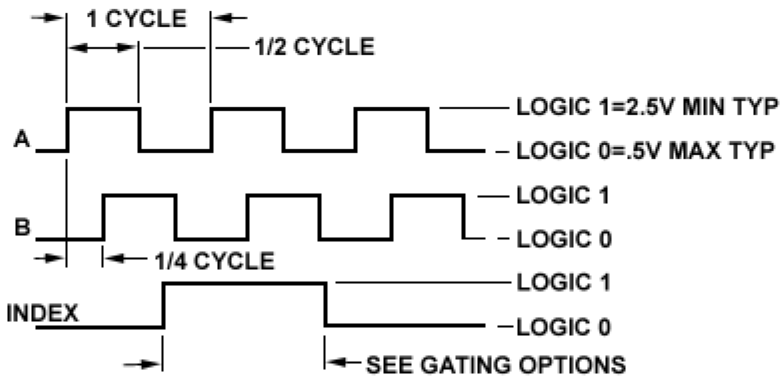


GATING OPTIONS:

Specify	Gating Option	Index Width
0	Ungated	360° ± 20%
1	Gated W/A&B	90° ± 45°

Gating option 0 (Ungated) is standard. Ungated index is random with channels A and B.

Output Configuration:



Minimum transition to transition edge separation of A and B is 45° over operating temperature and frequency. Complementary outputs are not shown for simplicity. CCW viewing PC board surface.

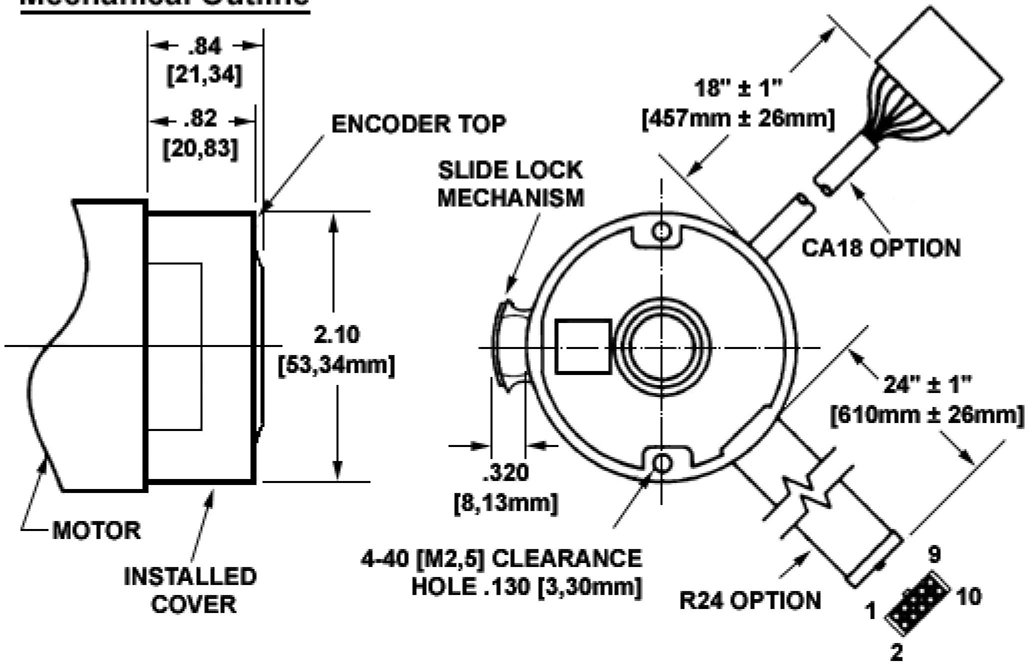
CLOSURE OPTIONS:

C 4-40 [M2,5] Mtg. No hole Cover
H 4-40 [M2,5] Mtg. Cover with hole

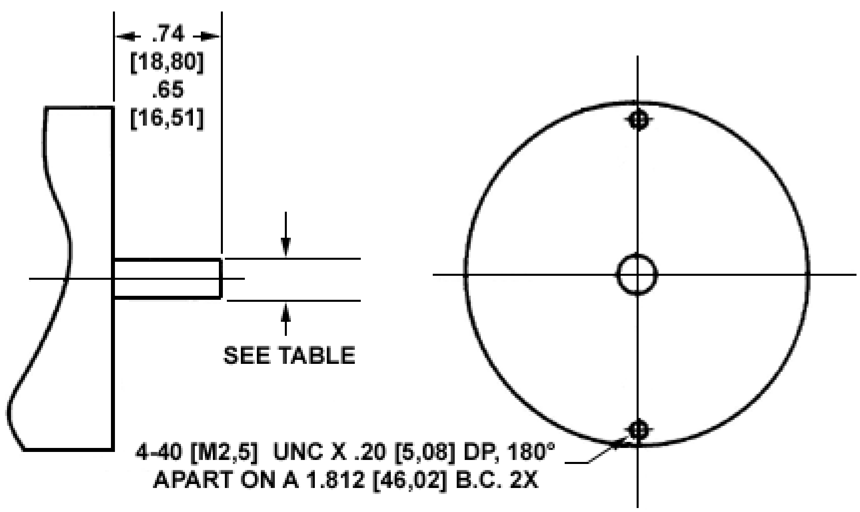
Shaft length longer than .75" [19,05mm] requires a through-hole cover. A mylar seal is provided with through-hole cover.

Dimensions:

Mechanical Outline



Mounting Requirements



APPENDIX X

USER MANUAL FOR NMIH-0050 H-BRIDGE DRIVER USED BY MICROCONTROLLER-2

The NMIH-0050 H-Bridge

Features:

- **5 A continuous, 6 A peak current**
- **Supply voltages from 5.3V up to 40V**
- **Terminal block for power / motor**
- **Onboard LEDs for motor operation/direction**
- **Onboard LED for motor supply**
- **Single row header for inputs**
- **Onboard LEDs for input indication**
- **Onboard LED for digital supply 5V**
- **Very low RDS ON - typically 200 mOhm @ 25 °C per switch**
- **Internal freewheeling diodes**
- **No crossover current**
- **Undervoltage lockout with hysteresis**
- **Overtemperature protection with hysteresis**
- **Output short circuit protected**
- **Error Flag signal indicator for fault conditions**
- **Onboard LED for Error Flag for fault detection**
- **CMOS/TTL compatible inputs with hysteresis**
- **Wide temperature range; - 40 °C < Tj < 150 °C**
- **Operating frequency up to 33 kHz**

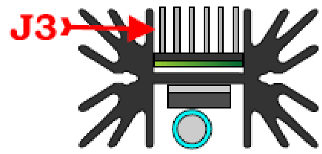
Benefits:

- **Visual indication of status: power, inputs, and drive outputs**
- **Compact size**
- **Good thermal distribution**
- **Internally protected**
- **Simple connections**
- **Easy interface**
- **Easy mounting options w/brackets**

The NMIH-0050 Full H-Bridge consists of a dual half h-bridge, screw terminal and pin connectors, single input inversion, and LED status indicators. It is usable for forward-reverse-stop control of many DC servomotors using TTL/CMOS compatible inputs to interface directly with most microcontrollers. It can also be used for phase control of steppers or brushless DC motors. With current handling to 5 A continuous, 6 A peak, it allows more flexibility in use with many small to mid range DC motors. Voltage supply range is from 5.3 to 40 V for the NMIH-0050. Switching characteristics allow PWM frequencies up to 33 KHz. Undervoltage protection, as well as Error Flag generation for shorted outputs, over temperature and over current are available for use.

Connections for the NMIH-0050

The NMIH-0050 has TTL/CMOS compatible inputs. Looking at the module, screw terminals away from you, board with LED's up, the 0.1 spaced pin connections are the digital signals from the controlling processor. This is J3.

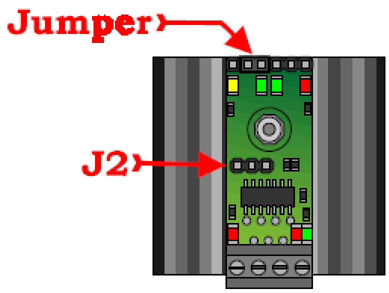


From left to right, these signals are: Vdd, Gnd, IN1, IN2, IN1', and EF.

Vdd	Gnd	IN1	IN2	IN1'	EF

Vdd is a logic supply pin, accepting a voltage range of 3 to 6 volts (5 volt nominal) for powering the LED indicator functions on the module. A LED near the Vdd pin serves as a power indicator. Gnd is a common ground pin, and the return point for the Vdd power signal as well as the ground reference for the other digital control signals. IN1 is the primary input to switch one half of the h-bridge, or both halves using Locked Anti-Phase mode from the microcontroller. IN2 is the secondary input to switch the other half of the h-bridge. Pins IN1 and IN2 also have input indicators to show how the module is being switched by a microcontroller. IN1' is an output pin, the inverted state of the input pin IN1. It is place next to IN2 so, if jumpered, a single signal on IN1 can control the whole H-bridge in Locked Anti-Phase mode. A jumper between IN2 and IN1' drives the module from a single PWM on IN1. Without this jumper, using both IN1 and IN2 pins, Sign Magnitude control can be used. EF is the error flag, an output from the module that your microcontroller can read to detect faults. The EF pin also has a visual LED indicator on the module just below that pin, to show if an error condition has been generated.

A 3-pin .1" connector is provided near the middle of the module as an alternative signal input point. It is J2.

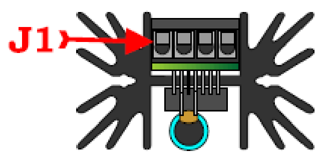


Viewed with screw terminal down, and looking down at the board, it's pins are from left to right: IN1, Vdd, and GND.

IN1	Vdd	Gnd

With a jumper between the upper IN2 and IN1' pins on the connector at the edge, J3, this center connector, J2, is a useful place to connect for Locked Anti-Phase control. RC servo pin out are compatible with this center connector. Several microcontrollers, such as the IsoPod™, have this pin out order. Hence this intended as a back up connection point with a different pin out the J3, useful for connecting to PWM outputs wired for RC Servo connections.

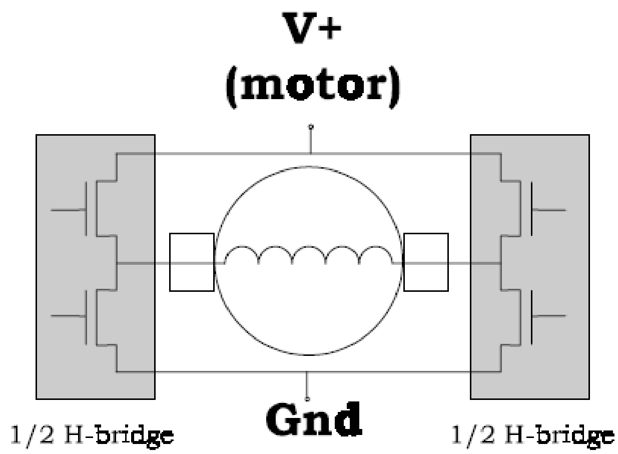
Reversing the board, look at it with J1, the 0.35mm spaced screw terminals toward you, board side up.



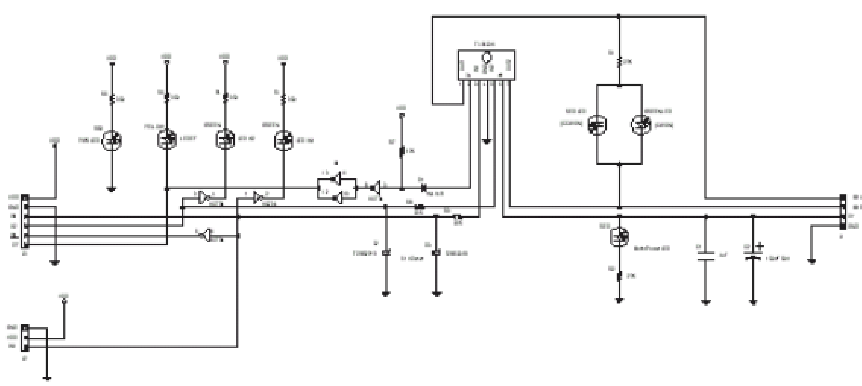
The connector layouts are: GND, V+, OUT2, OUT1.

Gnd	V+ (motor)	OUT2	OUT1

The left two screw terminals are for the power (or battery) motor supply. GND is the common ground for the motor supply, the driver chip tab, and the heatsink, and a common connection to the digital side ground. Vmotor is the positive voltage motor supply of up to 40 volts for the motor. A power indicator LED here above the V+ supply pin to indicate battery power coming into the module. If you reverse the battery and motor connections, you will cause the error flag LED to intermittently light and have the motor turning at normal speed with IN1 high, slow speed in the same direction with IN1 low. This could damage your module, but typically it should still function once motor power and motor outputs are connected correctly.



OUT2 is one half of the H-bridge, the output connection to one side of the motor. This half of the H-bridge corresponds to the input IN2 signal. OUT1 is the other half of the H-bridge, the output connection to one side of the motor. This half of the H-bridge corresponds to the input IN1 input. Above the OUT1 output, a pair of LEDs indicates the direction the motor is being driven.



Functional Truth Table

IN1	IN2	OUT1	OUT2	Comments
L	L	L	L	Brake; both low side transistors turned-ON
L	H	L	H	Motor turns counterclockwise
H	L	H	L	Motor turns clockwise
H	H	H	H	Brake; both high side transistors turned-ON

Notes for Output Stage

Symbol	Value
L	Low side transistor is turned-ON High side transistor is turned-OFF
H	High side transistor is turned-ON Low side transistor is turned-OFF

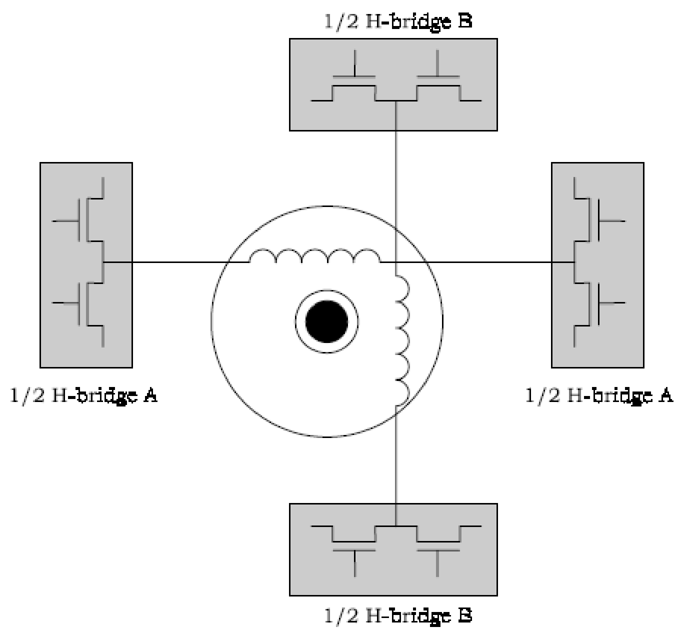
Control of a DC motor with the NMIH-0050

The inputs IN1 and IN2 drive the corresponding OUT1 and OUT2 outputs according to the above table:

The module can be driven directly with 2 pins from the controller to the IN1 and IN2 inputs to follow the above table. In Sign Magnitude control, the two pins must be used. IN1 must be switched High with PWM applied to IN2 for forward rotation, or IN2 switched High with PWM applied to IN1 for reverse rotation. The module can be driven directly with a single pin at IN1, if the IN2 input pin is jumpered to the IN1' output pin. The motor is driven constantly forward based on input IN1 being driven high. Or, the motor is driven constantly backward if input IN1 being driven low. In a similar mode, Locked Anti-Phase drives PWM this input, IN1 and the IN2 signal jumpered to IN1' gets a complementary PWM. A 50% duty cycle PWM signal produces a stopped motor, a net braking effect. Higher duty cycle signals produce motor drive in one direction, and lower duty cycle PWM signals drive in the other direction.

Bipolar Stepper Motor Control

Using 2 NMIH-0050 modules, bipolar steppers can be sequenced by a microcontroller. Connecting one modules OUT1 and OUT2 outputs to the A coil and a second modules OUT1 and OUT2 to the B coil, and then driving the two pair of IN1, IN2 inputs, each coil of the stepper can be sequenced according to the desired motion for the stepper. Complementary driving of the IN1 and IN2 will apply power to the stepper coil. Same signal driving of IN1 and IN2 will remove power. Therefore, having a varying PWM on one of the pair and grounding the other can be used as a chopper circuit to control power applied. By using Locked Anti-Phase, micro stepping of some bipolar stepper motors should be possible as well.



Circuit Comments

There is no fuse protection on this product. Any fuse protection desired must be provided externally.

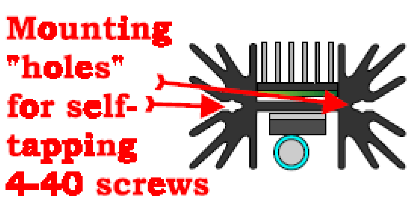
The 5.1V Zeners to ground in series with a 20K resistor to the inputs on the board provide protection for the input lines of a controller from the motor supply forcing back into the drive inputs if the internal chip circuitry should catastrophically fail.

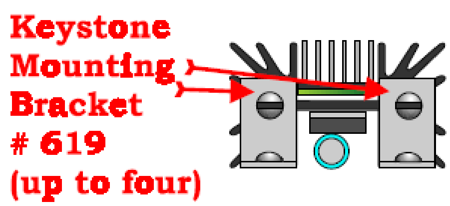
Use of 0.1 uf caps on each outgoing line at the motor terminals is recommended. The capacitors are for limiting noise from the motor itself. A much less desirable placement would be at the connector of the H-bridge, which lets the RFI travel through the connections, using them as antennas. Terminate them at the source, the motor, if possible. Typical applications is for one directly across the two motor posts, and another each from the posts to the motor case.

Mounting the NMIH-0050

Grooves provided in the heat sink of the NMIH-0050 make excellent mounting points. They are sized to accept a 4-40 screw. Self-tapping screws can easily be driven into the soft aluminum heat sink by hand to hold the heat sink to mounting brackets, which can then be attached to any flat surface. Up to two on each end of the heat sink.

For detailed mechanicals, and thermal characteristics of heat sink used in the NMIH-0050 is from Wakefield Engineering. See:
http://www.wakefield.com/pdf/Board_Level_Heat_Sink.pdf at bottom of page 16 of the pdf file (labeled page 45 in text), Part # 647-15ABP. This heat sink can be secured to any flat surface by using Keystone Mounting bracket, item Cat. # 619. See:
http://www.keyelco.com/kec/standpro/specpage/spec36.htm.





VITA

Arul Selvam Simon Jeyapalan

Candidate for the Degree of

Master of Science

Thesis: AUTOMATED WRAPPING OF SILVERWARE IN A NAPKIN

Major Field: Mechanical Engineering

Biographical:

Personal Data: Born in Pettai, Tamilnadu, India, on June 21st 1980, the son of Jeyapalan Selvanayagam.A and Dulsybai.J.

Education: Received Bachelor of Engineering Degree in Mechanical Engineering from Government College Of Engineering at Salem under Madras University at May 2001. Completed the requirements for the Master of Science Degree at Oklahoma State University in December 2005.

Experience: Graduate Research Assistant, Department of Mechanical And Aerospace Engineering, Oklahoma State University from January 2004 to May 2004, and August 2004 to December 2005.

Name: Arul Selvam Simon Jeyapalan Date of Degree: December, 2005

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: AUTOMATED WRAPPING OF SILVERWARE IN A NAPKIN

Pages in Study: 223 Candidate for the Degree of Master of Science

Major Field: Mechanical Engineering

Scope and Method of Study:

This Study investigates the wrapping of silverware in commercial dishwashing applications. The purpose of this research is to design and implement an automated silverware wrapping device that is capable of automatically picking cloth napkins from a napkin bin and wrapping the napkin in an appealing manner with a set of silverware comprised of a knife, fork and spoon. The major challenges encountered were in identifying the right techniques used to handle napkins and wrap napkins to get an appealing wrap. The method was primarily experimental.

Findings and Conclusions:

New and innovative techniques were used to singulate napkins from a stack and napkins, place silverware in the napkin and roll the napkin in an appealing manner. An effective control unit, used to control the entire device was implemented. The sliding friction between the napkins in a napkin stack was found to affect the performance of the entire device. Experimental results established a processing efficiency of 68% and a processing efficiency of 100% when napkins were fed manually.

ADVISER’S APPROVAL: DR. LAWRENCE L HOBEROCK