

## References

- [1]. Mason, M.T.. "Mechanics and planning of manipulator pushing operations". International Journal of Robotics Research, vol5(3): 53-71, Fall. 1986.
- [2]. R. C. Brost. "Automatic grasp planning in the presence of uncertainty". International Journal of Robotics Research, Vol.7, No.1, February 1988.
- [3]. M. Mani and W. Wilson. "A programmable orienting system for parts". North American Manufacturing Research Institute Conference XIII, 1985
- [4]. Peshkin M.A., Sanderson A.C., "The motion of a pushed, sliding work piece", IEEE J. on Robotic. & Automation., vol.4(6):569-598, 1988.
- [5]. M. A. Peshkin and A. C. Sanderson. "Planning robotic manipulation strategies for work pieces that slide". IEEE Journal of Robotics and Automation, Vol.4, No.5, October 1988.
- [6]. Akella Srinivas and M.T. Mason. Posing polygonal objects in the plane by pushing. In IEEE International Conference on Robotics and Automation, pages 2255-2262, 1992.
- [7]. Akella S., Mason M.T., " Parts orienting by push-aligning", *IEEE Int'l Conf. on Rob. & Autom. ICRA '95.*, Nagoya, Japan vol.1:414-420, May 1995.
- [8]. Lynch, Kevin M., Hitoshi Maekawa, and Kazuo Tanie. "Manipulation and active sensing by pushing using tactile feedback". Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 416-421, 1992.
- [9]. Agarwal, Pankaj K., Jean-Claude Latombe, Prabhakar Raghavan and Rajeev Motwani. "Nonholonomic path planning for pushing a disk among obstacles". Proceedings of IEEE International Conference on Robotics and Automation, 1997.
- [10]. Takagi, Seiji and Yoshikuni Okawa. "Rule- based control of a mobile robot for the push-a-box Operation". Proceedings of IEEE/RSJ International Workshop on Intelligent Robots and System, 1991.
- [11]. Okawa, Yoshikuni and Ken Yokoyama. Control of a mobile robot for the push-a-box Operation. In proceedings of IEEE International Conference on Robotics and Automation, 1992.

- [12]. Lynch, Kevin M. Nonprehensile robotic manipulation: controllability and planning. PhD thesis, Robotics Institute, Carnegie Mello University, Pittsburg, Pennsylvania, 1996.
- [13]. Seiichiro Katsura, Kouhei Irie, and Kiyoshi Ohishi, Wideband Force Control by Position-Acceleration Integrated Disturbance Observer, IEEE Transactions on Industrial Electronics, VOL. 55, NO. 4, April 2008
- [14]. S. Katsura, Y. Matsumoto, and K. Ohnishi, "Analysis and experimental validation of force bandwidth for force control," *IEEE Trans. Ind. Electron.*, vol. 53, no. 3, pp. 922–928, Jun. 2006.
- [15]. S. Katsura, Y. Matsumoto, and K. Ohnishi, "Modeling of force sensing and validation of disturbance observer for force control," *IEEE Transaction on Industrial Electronics.*, vol. 54, no. 1, pp. 530–538, Feb. 2007.
- [16]. Microchip Technology Inc., PIC16F87XA datasheet(DS30582B), pp2, 2003
- [17]. Fairchild Semiconductor Corporation, DM74LS08, Quad 2 input and gate datasheet(DS006347), 2000
- [18]. Peshkin, Michael A., and Arthur C. Sanderson. The motion of a pushed, sliding workpiece. *IEEE Journal of Robotics and Automation*, 4(6), 569-598, 1988a.
- [19]. Peshkin, Michael A. and Arthur C. Sanderson. Planning robotic manipulation strategies for work pieces that slide. *IEEE Journal of Robotics and Automation*, 4(5), 524- 531, 1988b
- [20]. Mathththew T. Mason, *Mechanics of Robotics Manipulation*, Prentice-hall of India, pp 121-141, 2005
- [21]. SGS THOMPSON microelectronics corporation, HCF4017 datasheet, June 1989.
- [22]. A.M.H.S. Abeykoon, Kouhei Ohnishi, "Estimation of Optimal Slip for Traction Force Improvement of a Mobile Manipulator", *Proceedings of the International Conference on Information and Automation*, Colombo, Sri Lanka, December, 2005..
- [23]. <http://www.oshonsoft.com/>
- [24]. Wang, Yu and M.T. Mason. Two-Dimensional Rigid-Body Collisions With Friction. *Journal of Applied Mechanics* Vol.60, June, pp. 566. 1993.
- [25]. De Wit, C.C., B. Siciliano, and G. Bstin. *Theory of Robot control*. Springer-Verlag, London. 1996.

## 6.1 Appendix A: Source code for testing robot

The source code written for the PIC BASIC compiler is given below

```
Hseropen 9600
Dim duty1 As Word
Dim duty2 As Word
Dim duty1_temp As Word
Dim duty2_temp As Word
Dim preinput1 As Bit
Dim preinput2 As Bit
Dim x As Bit
Dim pulse_1 As Word
Dim pulse_2 As Word
Dim pulse As Word
Dim drive As Byte
Dim duty_st As Word
Dim duty_th As Word
```



University of Moratuwa, Sri Lanka.  
Electronic Theses & Dissertations  
[www.lib.mrt.ac.lk](http://www.lib.mrt.ac.lk)

```
duty_st = 375
duty_th = 385
duty1 = duty_st
duty2 = duty_st
```

```
pulse_1 = 0
pulse_2 = 0
trisb.4 = 1
trisb.5 = 1
```

```
INTCON.GIE = 1 'enable all un-masked interrupts
INTCON.PEIE = 1 'enable peripheral un-masked interrupts
```

```
INTCON.rbie = 1 'ENABLE PORTB CHANGE INTERRUPT
```

```
PIE1.rcie = 1
```

```
PWMon 1, 1
```

```
PWMon 2, 1
```

```
pr2 = 249
```

```
PWMduty 1, duty1
```

```
PWMduty 2, duty2
```

```
preinput1 = portb.4
```

```
preinput2 = portb.5
```

```
drive = "F"
```

```
loop0:
```

```
    If pulse_2 <= 5 Then
```

```
        duty2 = duty2 + 1
```

```
        PWMduty 2, duty2
```

```
        WaitMs 100
```

```
        Goto loop0
```

```
    Else
```

```
        PWMduty 2, duty_st
```

```
        pulse_2 = 0
```

```
    Endif
```

```
main:
```

```
    If drive = "F" Then
```

```
        loop1:
```

```
        If pulse_1 <= 10 Then
```

```
            duty1 = duty1 + 1
```

```
            PWMduty 1, duty1
```

```
            WaitMs 100
```

```
            Goto loop1
```

```
        Endif
```

```

If pulse_1 > 10 Then
    PWMduty 1, duty_st
    Hserout "R", #duty1, " "
    duty1_temp = duty1
    duty1 = duty_st
    pulse_1 = 0
Endif

```

```

loop2:
If pulse_2 <= 10 Then
    duty2 = duty2 + 1
    PWMduty 2, duty2
    WaitMs 100
    Goto loop2
Endif

```

```

If pulse_2 > 10 Then
    PWMduty 2, duty_st
    Hserout "L", #duty2, CrLf
    duty2_temp = duty2
    duty2 = duty_st
    pulse_2 = 0
Endif

```

```

If duty1_temp > duty_th And duty2_temp > duty_th Then
    If duty1_temp = duty2_temp Then
        loop3:
            If pulse_1 <= 5 Then
                duty1 = duty1 + 1
                PWMduty 1, duty1
                WaitMs 100
                Goto loop3
            Endif
        Endif
    Endif
Endif

```

```

                Else
                    pulse_1 = 0
                Endif

                PWMduty 1, 400
                PWMduty 2, 400
                WaitMs 3000
            Endif
        Endif
    Else

        PWMduty 1, duty_st
        PWMduty 2, duty_st

    Endif
    Goto main
End

```



```

On Interrupt
Save System
If PIR1.rcif = 1 Then
    Hserin drive
Endif

If INTCON.rbif = 1 Then
    x = Not preinput1
    If portb.4 = x Then
        pulse_1 = pulse_1 + 1
        preinput1 = portb.4
    Endif
    x = Not preinput2
    If portb.5 = x Then
        pulse_2 = pulse_2 + 1
    Endif
Endif

```

```
preinput2 = portb.5
```

```
Endif
```

```
INTCON.rbif = 0
```

```
Endif
```

```
Resume
```



University of Moratuwa, Sri Lanka.  
Electronic Theses & Dissertations  
[www.lib.mrt.ac.lk](http://www.lib.mrt.ac.lk)