International Journal of General
Engineering and Technology (IJGET)
ISSN 2278-9928
Vol. 2, Issue 1, Feb 2013, 15-24

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

A SOFTWARE BASED ELECTRIC ACTUATOR CONTROL SYSTEM WITH ROBOTIC ARM - A LEARNING AID FOR UNDERGRADUATE STUDENTS

MOHAMMAD SAIDUR RAHMAN[1] & SAJAL CHANDRA BANIK[2]

[1]Student, Department of Mechanical Engineering, Chittagong University of Engineering & Technology, Chittagong, Bangladesh

[2]Associate Professor, Department of Mechanical Engineering, Chittagong University of Engineering & Technology, Chittagong, Bangladesh

ABSTRACT

The main objective of this work is to develop a hardware and software based learning tool to control any electric actuator such as stepper and servo motor. Later on, a control scheme is discusses for an industrial robotic arm by using these motors with the software system which can be used by the undergraduate students to learn about electro-mechanical system. The software is used to control actuators and the robotic arm through computer by using serial communication protocol. The robotic can be used to pick objects with acceptable perfection.

KEYWORDS: Mechatronics Systems, Robotic Arm control, Stepper & Servo Motor Control, Serial Communication

INTRODUCTION

Electric actuator acts as muscles of robotic systems. Actuator can move or rotate the links/joints to change the configuration of robots. It has enough power to accelerate or decelerate the links. It can easily carry any payload. High accuracy, light weight, and ease of maintenance are the most important benefits of electric actuator [1]. “Actuation systems are the elements of control systems which are responsible for transforming the output of a microprocessor or control into a controlling action on a machine or device” [2]. From this statement, it is very much clear to us that actuators are very much important in the *Mechatronics systems*. There is a wide application of electric actuators mainly used in industrial automation such as process industries, chemical industries, automotive industries ,etc. Figure 1 shows the basic schematic diagram of the proposed work.

Figure 1: Basic Schematic of the System

From this work, software based learning tool will help to understand easily to the undergraduate students for learning aid. This hardware system receives the command signal from a control panel which is a microcontroller based circuit. By this way it helps to learn about the interfacing of microcontroller with industrial actuator such as stepper, servo motor, etc. and it is accomplished by dedicated software using *USART* serial communication system. Which also gives a better understanding of interfacing with a computer system. The robotic arm has been made for the best performance of object picking associated with this proposed system, which is done by CACS (Computerized Actuator Control System) software developed in Visual Basic 6.0.

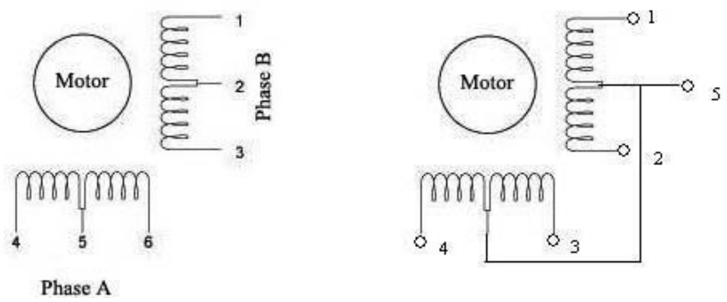
ELECTRIC ACTUATOR CONTROL

There are many types of actuators used in the sector of industrial automation such as hydraulic actuators, pneumatic type actuators and electric motors. Hydraulic and Pneumatic type actuators were popular for heavy and large type of robots in the past. But the most commonly used robotic actuators are stepper and servo motors. So, the discussion will be only about the controlling of robotic actuators using the software.

Stepper Motor Control

The stepper motor is an electro-mechanical device which is rotated in an equal step angel. The typical step angel is 0.9^0 and 1.8^0. It receives electronic pulses in a proper sequence and then converts it to a mechanical output by rotating its rotor shaft. The sequential electronic pulses can be given by a simple microprocessor based control circuit. There are number of forms of stepper motor, such as Variable Reluctance Stepper, Permanent Magnet Stepper and Hybrid Stepper [2].

Here, only the controlling of unipolar stepper motors is discussed with its function. These motors are usually used in the robotics, *CNC* machining systems, industrial production lines and also in the precise controls [3], [4]. Unipolar stepper motors are generally designed to work with only one power source. It has four coils and may have five, six or eight wire connections [5]. Figure 2 shows various stepper connections.



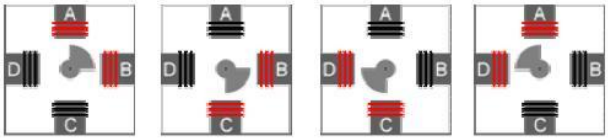
(a) 6 Wires Connection (b) 5 Wires Connection

Figure 2: Example of Various Stepper Motor Wire Configuration

The basic approach to configure a stepper motor is as followed:

- *First*, checking the number of wires coming out from it.
- *Secondly*, measuring the resistance between the leads to find out the common wire and pulse wires.
- *Thirdly*, giving a manual pulses to the pulse wires of the motor and continue the checking to find the sequence of these leads by trial and error.
- *Finally*, connecting the common wire to the power line, and then connect the sequential wires to the driver circuit, which is normally a microcontroller based control circuit.

In this work, all the stepper motors have 1.8 degree/step. It can be run in full or half torque mode. The stepper motor has no commutator but has a series of coils. Instead, there is a five or six wires coming out of the motor; one wire for each coil (usually four) can be connected in a ground and one or two common wires can be connected in a positive power connection. In order to get the motor to turn, power must be applied to one coil after another in the proper sequence and it can be done by using a simple driver circuit which can be controlled from a microcontroller. To get the full torque, two coils must be always turning on at any instant. Figure 3 shows the sequential signal of the stepper motor driving for full torque.



(a) (b) (c) (d)

Figure 3: Basic Principal Operation of Stepper Motor

Microprocessor Control of Stepper Motor

A microprocessor is a digital device. It only deals with digital inputs and outputs. Digital devices like microcontrollers handle numbers by bits. A one-bit of information can be represented in two states of (0 or 1) [1]. So, to control the motion of the stepper motor we have to use a control circuit. The control circuit is constructed to receive the signal from the software and send it to the microcontroller unit (usually known as MCU), and then it will create sequential pulses to drive the stepper motor as per instructions via the driver circuit. Table 1 & 2 shows the instruction of the pulse sequence for full and half torque.

Table 1: Pulse Instruction of MCU for Full Torque

	Pulse Instruction Sequence for Full Torque			
	bit 1	*bit 2*	*bit 3*	*bit 4*
Pulse 1	**1**	0	0	**1**
Pulse 2	**1**	**1**	0	0
Pulse 3	0	**1**	**1**	0
Pulse 4	0	0	**1**	**1**

Table 2: Pulse Instruction of MCU for Half Torque

	Pulse Instruction Sequence for Half Torque			
	bit 1	*bit 2*	*bit 3*	*bit 4*
Pulse 1	0	0	0	**1**
Pulse 2	0	0	**1**	0
Pulse 3	0	**1**	0	0
Pulse 4	**1**	0	0	0

This control system needs a driver circuit, which is a *NPN* transistor based circuit to switch sequentially turn on the wires of the stepper motor. A BD135 power transistor is used to make this driver circuit. The every *bit* of Table 1 & 2 represents the every transistor. Figure 4 shows the diagram of a simple stepper motor driver for 5 wires. 4 wires of the stepper motor must be connected to this driver circuit output port and the remaining common wire must be connected with the positive pole of the power supply. A Graphical User Interface (GUI) program sends signals through the computer's *RS-232* communication protocol to the MCU, which is then decoded to drive the motor as per required instructions. The signal code will be generated only by the user operator from the software system.

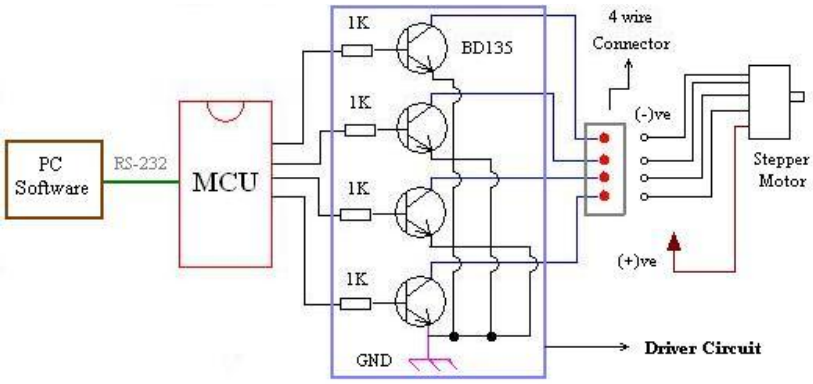


Figure 4: Stepper Motor Driver Circuit for 5 Wires

Servo Motor Control

A servo motor is a DC, AC, brushless or even stepper, motor with feedback that can be controlled to move at a desired speed (and consequently, torque), for a desired angle of rotation. To do this, a feedback device sends signals to the controller circuit of the servo motor reporting its angular position and velocity [1]. Servo motors are usually used in such applications such as robotics, CNC machinery or in automated manufacturing. In the proposed hardware system, one servo motor *SG-5010* [8] is used as for learning aid. For large industrial servo motors, *Variable Frequency Driver* (VFD) is often used to control their speed. But, here MCU is used to generate Pulse Width Modulation (PWM) signal to control its angular position. PWM is a process of controlling devices by varying the length of the pulse in a given time period. To do this, the voltage on the output port of the microprocessor is turned on and off repeatedly, many times a second [1]. Duty Cycle describes the proportion of 'on' time to the regular interval or 'period' of time; a low duty cycle corresponds to low power, because the power is off for most of the time. Figure 5 shows the position control of the servo motors

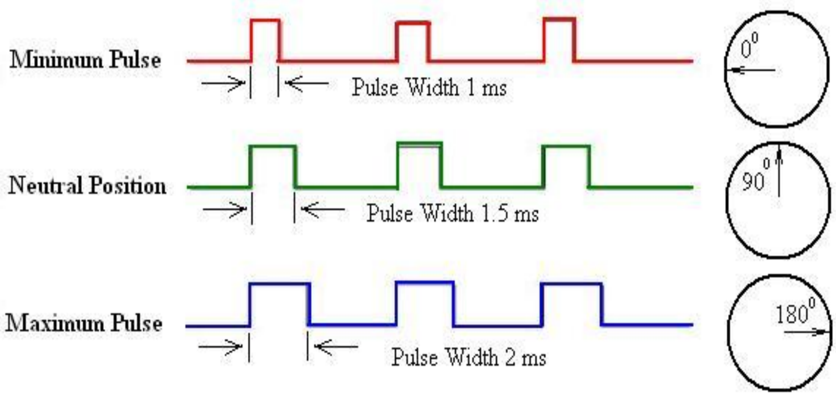


Figure 5: PWM Signal for Servo Motor Position Control

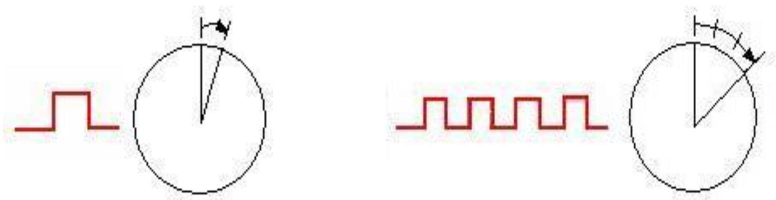
The duty cycle is usually expressed in percent, 100% being fully on. The duty cycle (D) is calculated by the following expression:

$$D = \frac{t_{on}}{T} \tag{1}$$

where, t_{on} = on/high state and T = period of the function. In these servos the time period of oscillation is 20ms and the on-time varies from 1ms for 0 degrees to 2ms for 180 degrees.

GENERATION OF COMMAND CODE

The command code will be written in decimal format but the Universal Synchronous Asynchronous Receiver/Transmitter (*USART*) system sends them in *ASCII* format sequentially from PC to the microcontroller unit. The data is received and transmitted in the form of *ASCII* which represents the decimal value within 0 to 255, which satisfied 8 bit memory system. It is very easy to send or receive these decimal values to work as like as Numerical Control (NC) system [6]. A memory stack/array matrix is used to send the signal code into the systems control board. The data of the array stack is then prepared by the software according to the users given command which is the function of some parameters including direction of rotation, torque (full/half), number of rotations and the degree of angle of the stepper or servo motor. The data is stored in the array using Last In First Method (LIFO). In this research work, open-loop control design is used and stepper motors are mainly used for simple point-to point positioning tasks if they are in open-loop control system [7]. Stepper motor converts a train of pulses into shaft revolutions. So, each pulse equals to one rotary increment, which is a portion of one complete rotation. Figure 6 clarifies this process.



(a) One Pulse Equals One Step **(b) Pulse Count Equals Step Count**

Figure 6: Stepper Motor Rotation System Based on Pulses

Initially, a user must have to give some parametric data to the software on the basis of stepper or servo motor principle. These information will store in a matrix array, which is then send to the MCU based control board via the *USART* communication system. Let's consider, 1.8^0 is the per step rotation of the stepper motor and 360^0 is the possible full rotation for one revolution. Now, we can write a mathematical relation about some motor parameters shown in Eqn (2) & Eqn (3).

Degree of Rotation, $D = d \times N$ (2)

$$N = \frac{D}{d} \tag{3}$$

Where $\boldsymbol{d}$ is the per step rotation (1.8^0/step) and $\boldsymbol{N}$ is the number of step required for rotation of the motor. Suppose, 189^0 rotation is needed, then the no. of steps, **N** = 105 (using Eqn 3). Hence, the software will only send the value of $\boldsymbol{N}$ and other informations relating with the given data and the program will arrange the informations in a 6×1 matrix/array as shown in Table 3 & 4.

Table 3: Code Generation Sequence

No.	Data Elements	ASCII Char./ Value	Description
1	5	'ENQ' = 5	Data 1
2	0	'NUL' = 0	Data 2
3	1	'SOH' = 1	Data 3
4	1	'SOH' = 1	Full Torque
5	2	'STX' = 2	Counter Clokwise
6	240	'ð' = 240	Specific Code for Stepper

Table 4: Example of Data Formation in MCU

N (no. of steps) = 105	**Data 3**	**Data 2**	**Data 1**
	1 x 100 +	0 x 10 +	5 x 1

The full torque is '1' in that memory stack, which is a specified value for full or half torque mode. In this same manner, for the memory stack no. of 5 & 6 some specified values are also given in the software algorithm and as well as in the microcontroller. The MCU will control the signal by receiving and following this specification to drive the electric actuator and the robotic arm. The flowchart of the software algorithm is shown in Figure 7.

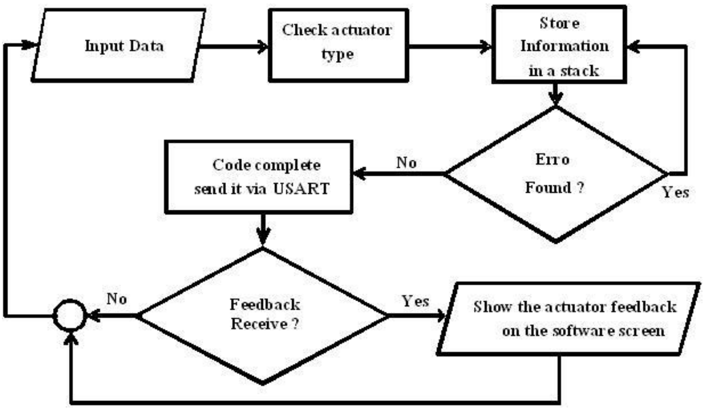


Figure 7: Flow Chart of the Software Algorithm

HARDWARE SETTINGS

The very basic construction of this system is made by a hardware consisting of a control board, and the software system to control the actuators and the robotic arm. A serial communication system of computers *RS-232* protocol is also used to demonstrate the hardware system.

System Modelling

The basement hardware of this system is made by plywood. Others hardware are one stepper motor, one servo motor [8], one lead screw, horizontal & circular scales, some gears & rivets for the gripper and the arm links of two degree of freedom robotic arm. This robotic arm is constructed with two servo motors and some aluminum plates. Linear and circular scale is used for visual indication. Figure 8 shows the schematic of the system model.

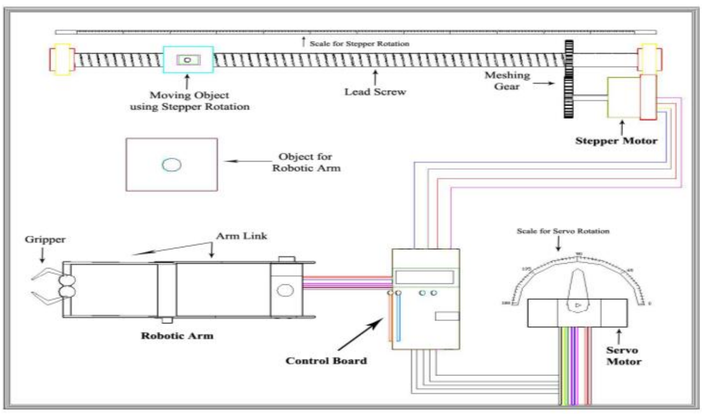


Figure 8: Hardware Schematic of the Training Board System

Control Board Contstruction

The control board is constructed to receive and process the signal. The circuit is controlled by a MCU. The *USART* serial data communication system is used to establish the communication between the PC and MCU.

The control circuit consists of a microcontroller PIC16F877A, a crystal 8 MHz, MAX232, voltage regulator 7805, electrolytic capacitor 1μF, 100μF, power transistor BD135, Resistors 1K Ω (0.25 watt), 2.2Ω(10watt), LED, 16x2 character LCD and 12V power supply. All electronic equipments are soldered in the PCB board. A MAX-232 IC is used for the serial communication.

The pin number 7, 8, 9 & 10 of max232 is sequentially connected with 2 and 3 pin of RS-232 port, and RC7 & RC6 of the MCU. While 8 *bit* data reaches from the computer, it is then stored in a '*mem*' register inside microcontroller. Then the microcontroller will send the value of '*mem*' register through PORTD to drive the stepper motor. The Servo motor is controlled using PWM signal through PORTC of RC2 and the other two servos for the robotic arm are controlled from RC0 & RC1 pin. Figure 9 shows the schematic diagram of the control board.

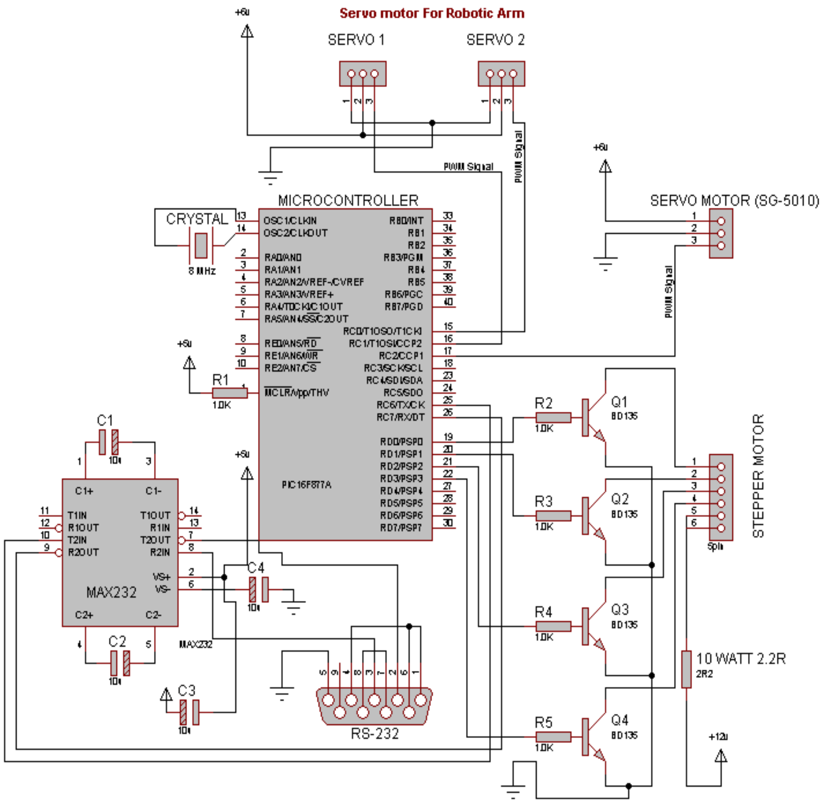


Figure 9: Electric Schematic Diagram of the Control Board

The USART Communication System

A Universal Synchronous Asynchronous Receiver & Transmitter is abbreviated as *USART*. This system is usually used in conjunction with communication standards such as RS-232 protocol. The *Universal* designation is used, which indicates that the data format and transmission speed can be configurable and also the electronic signal levels or the method. 8 bit asynchronous data transmission system is used here. The Baud rates for serial data transmission systems are generally 300, 1200, 2400, 9600,192000, 38400, 115200 [9]. Here in this project, 1200 baud rate is used. Figure 10 shows the data transmission system.

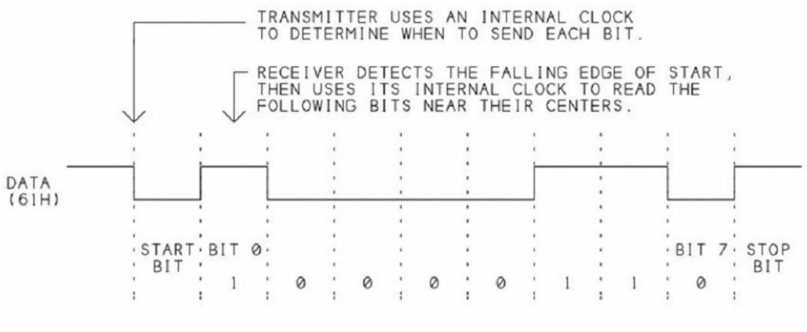


Figure 10: Basic Asynchronous Data Transmission System [9]

MECHATRONICS SYSTEMS & APPLICATIONS

A wide range of mechatronic applications is used in the field of industrial and factory automation. Today, mechatronics system is a combination of advanced technology from engineering, in particular sensors, actuator control, adaptive systems with the computer technology. Improved productivity, greater accuracy of product, reduces the personal requirements of the engineering, assistance in inspection of complicated parts; these are the wonderful benefits of mechatronics systems.

Robots and the robotic systems are one of the best applications of mechatronics. Product industries & factories are using this type of systems, because they need an efficient, controlled & adaptive system for their product manufacturing. Machine loading operation, pick and place operations, welding, painting, inspection, sampling, assembly operations, safety control etc. are the most popular applications of mechatronics system [1]. CNC machining systems are the important mechatronic systems for the advanced manufacturing [10].

Learning tool is an important part for the better understanding of the mehatronics systems. This type learning aid helps to improve the technical skills as developed in this research work.

CONCLUSIONS

This work has constructed an Electro-Mechanical system and the software supports for user friendly controlling. This system is fully software controlled and it sends the instructions continuously to the controller circuit. The system works with a good level of perfection and the software performs the operations as instructed. But, sometimes there are some backlash errors, tooth errors because of the roughness of the lead screw. It is under consideration for further development. The response time very fast and the commands are executed precisely. It can receive & follow the prescribed command very smoothly.

REFERENCES

1. S. B. Niku, Introduction to Robotics Analysis, Systems, Applications, ISBN-81-203-2379-3.
2. W. Bolton, Mechatronics, Electronic control system in mechanical & electrical engineering, ISBN 81-7758-284-4
3. Z. Yagun and C. LONG, "A design of elevator positioning control system model", IEEE Intl. Conf. on neural networks and signal processing, pp. 535-538, June 7-11, 2008
4. K. R. Rajagopal, M. Krishnaswamy and B. Singh, "An Improved High- Resolution Hybrid Stepper Motor for Solar-Array Drive of Indian Remote-Sensing Satellite", IEEE transaction on industrial applications, Vol. 33, No.4, pp. 906-913, 1997.

5. S .G. ABEYRATNE and U. I. DAYARATNE, “A New A Power Conversion Strategy for a Unipolar Stepper Motor Drive”, IEEE transaction on industrial electronics, pp. 213-217, 2010.

6. N. K. Mehta, ‘Machine Tool Design and Numerical Control’, ISBN No. 0-07-462237, pp 382-384. 1996

7. S. F. Rezeka, N. M. Elsodany, and N.A.Maharem “Fuzzy Gain Scheduling Control of a Stepper Motor Driving a Flexible Rotor”, European Journal of Scientific Research, Vol. 39, No.1, pp.50-63, 2010

8. http://www.giantcod.co.uk/sg5010-towerpro-servo-torque-p-205.html (accessed in December 2012)

9. J. Axelson, “Serial Port Complete Programming for RS-232 1 and Rs 485 Links and Networks, 2000-Lakeview Research, USA. ISBN-0-9650819-7-4, pp. 43- 45.

10. J. Balic, “Intelligent CAD/CAM system for CNC programming – An Overview”, Advances in Production Engineering & Management 1, ISSN 1854-6250, pp. 13-22, 2006