Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434

RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

Real-time implementation of model predictive control on a 16-bit microcontroller for speed control of a DC motor

Mujtaba Jaffery[1], Sohaib Aslam[2], Moinuddin Ghauri[3*], M. Shahzad Khuram[3], Sikandar Rafiq[3], Subhan Khan[1]**

[1]Department of Electrical Engineering, COMSATS Institute of Engineering and Technology, Lahore, Pakistan
[2]Department of Electrical Engineering, Superior University, Lahore, Pakistan
[3]Department of Chemical Engineering, COMSATS Institute of Information Technology , Lahore, Pakistan

*Corresponding authors

Abstract: Model Predictive Control (MPC) is an optimization based control technique and handles process operational constraints more effectively in comparison to Proportional Integral and Derivative Controller (PID). This paper presents real-time implementation of MPC on a 16-bit PIC microcontroller for speed control of a DC motor in constrained environment. Although the computational burden of MPC is a key hindrance but Laguerre functions have been used to reduce the computational burden of classical MPC algorithm and make it feasible for a simple microcontroller implementation. The simulations of MPC for speed control of a DC motor are presented in four modes without constraints, with input constraints, with output constraints, and with both input and output constraints. In all four modes, MPC gives better performance. Finally, MPC is real time implemented on a 16-bit PIC microcontroller for speed control of DC motor in presence of constraints and then its performance is further evaluated in presence of magnetic brake as a load torque and again MPC shows good results to convince the idea of implementing MPC on a cheaper microcontroller for single input and single output (SISO) application.to the knowledge.

Keywords: Optimization, DC motor, Microcontroller, SISO, Laguerre Functions.

1 Introduction

Automatic Control System is playing a key role in all areas of commercial ventures like quality control of made items, automatic mechanical production system, machine instruments, control space engineering, transportation frameworks, power frameworks, robotics and numerous more. Latest use of modern control theory includes in non-engineering systems as biomedical, biological, economic and socio-economic systems. The advancement of control software and technology has given a chance to utilize more refined control algorithms for industrial systems. Because of this, Proportional Integral (PI) and Proportional Integral Derivative (PID) have gained extensive application due to its compatibility to both hardware and software implementation [1]. PID controller has been the most broadly used controller because of its simplicity of usage [2] PID represents more than 90% of automatic industrial systems. PID compares the actual value with the desired value, and then uses the compared error to compute the new input to minimize this error and to achieve the desired steady state [3]. However, since most control processes operate at constraints, therefore, it must be ensured that the input and output constraints are not violated [4]. But PID does not has the ability to handle input and output constraints and also tuning parameters of PID have to be adjusted repeatedly to get optimal performance for any change occurring in dynamics of a system. PID has also limitations in handling multivariable processes subject to constraints [5,6]. Solution to these confinements is MPC that can deal with the process operational limitations more effectively. Different recent publications also indicate the growing market for advance controllers and few vendors offer turn-key products [7-9] MPC portrays human conduct by selecting control activities which may lead to best anticipated results over some constrained Skyline and continuously upgrade choices as new perceptions are accessible [10,11]. MPC acts as a closed loop optimal control method and explicitly use the process model to handle input and output constraints and also works more effectively in a dynamic environment [12-14]. MPC is often seen as more effective control technique and there is a great need to actively implement MPC in different control processes to get more sophisticated results. MPC is generally used to regulate slow dynamical systems with high bandwidth applications such as process control in the petrochemical, pulp and paper industries where sampling times can be on the order of seconds or minutes [15-17]. The major hindrance in implementing MPC on embedded system is the computational ability needed for the solution of an online recursive optimization problem which is used to calculate an optimized control input to satisfy both input and output constraints [14]. Development of advanced embedded systems with increased computational power and faster optimization algorithms has motivated researchers to implement MPC for higher bandwidth applications i.e. robotics, aerospace, electrical power and automotive [15]. Recent research work has investigated the performance of MPC in fast devices such as PLC and FPGA [15-17]. In [15] MPC algorithm is implemented on FPGA which takes advantage of slack parallel computational channels of FPGA to become suitable for systems with fast sampling times. [16] has optimized MPC algorithm and implemented on PLC. In [17] MPC has been implemented on PLC by transforming general MPC in to an efficient algorithm with less computational burden by using Laguerre functions. But the disadvantage associated with PLCs and FPGAs is their high- power consumption, long compilation times and high cost. Although, the significant requirement for embedded systems application is the small power and cost-effective solution [14]. [18] has proposed fast interior-point implementation of MPC without constraints on very fast but expensive embedded devices i-e Intel Atom with 32-bit processor, ARM Cortex A9. [19] and [20] has implemented MPC on a high performance 32-bit STM 32 microcontroller for SISO application in absence of input and output constraints. Similarly, [21] has implemented MPC on high speed CortexA7 series processor for a simple process and in presence of only input constraints and even it has tested the results only in Hardware in Loop (HIL) mode. [22] has used Laguerre functions to make MPC efficient and presented only simulations for DC drive control. Therefore, based on the need of fast dynamical systems with maximum power efficiency and cost-effective solutions, novelty of this research work is the improvement in computational burden of MPC by using Laguerre functions and then its implementation in real-time on a simple cost effective 16-bit microcontroller for speed control of DC motor in the presence of both input and output`` constraints. Also, the robustness of controller is tested by applying load torque at the output of DC motor. DC motor has outstanding speed control characteristics therefore, played an important role as a drive configuration in many applications for a wide range of torques and speeds and it also provides a test bed to researchers for testing different control algorithms [23-25]. In section 2 Laguerre functions are discussed. Section 3 presents implementation of MPC. Simulations based analysis is done in section 4. Hardware implementation and real-time results are shown in section 5. Finally, conclusion is summarized in section 7.

2. Laguerre Functions

When The optimized control vector ΔU of MPC is represented by:

$$\Delta U = [\Delta u(k_i)\ \Delta u(k_i + 1) \dots \dots \Delta u(k_i + N_C - 1)]^T \quad (1)$$

Where, N_C is the number of elements in control vector and is known as control horizon. At any given time k_i, the elements of optimized vector can be presented in the form of discrete δ-functions with ΔU is the coefficient

$$\Delta u(k_i + i) = [\delta(i)\ \delta(i-1) \dots\dots \delta(i - N_c + 1)]\Delta U \quad (2)$$

Where, δ(i) = 1 if i = 0 and δ(i) = 0 if i ≠ 0 so the function δ is the pulse operator and δ(i-d) shifts the center of pulse onward with increase in index d so above relation can be used to capture the trajectory of control vector. It is evident that $\Delta u(k_i + i)$ for i=0, 1,….N_C-1 can be characterized by a discrete polynomial function so discrete Laguerre functions can be used to realize the relation shown in Eq. (1) .

2.1 Discrete-Time Laguerre Networks

The Laguerre functions are orthonormal in nature and discrete time Laguerre network is generated by discretization of a continuous-time Laguerre network. The set of continuous time Laguerre functions for any value of p > 0 are defined below [26]:

$$l_1(t) = \sqrt{2p} \times e^{-pt}$$

$$l_2(t) = \sqrt{2p}(-2pt + 1)e^{-pt}$$

$$l_i(t) = \sqrt{2p}\,\frac{e^{pt}}{(i-1)!}\frac{d^{i-1}}{dt^{i-1}}\left[t^{i-1}e^{-2pt}\right] \quad (3)$$

Where, p is the time scaling constant of Laguerre functions. The z-transforms of discrete-time Laguerre functions are shown below:

$$r_1(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}}$$

$$r_2(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}}\frac{z^{-1}-a}{1-az^{-1}}$$

$$r_N(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}}\left(\frac{z^{-1}-a}{1-az^{-1}}\right)^{N-1} \quad (4)$$

Where, a is the user define pole of discrete-time Laguerre function and its value ranges from 0 to 1. The discrete time Laguerre functions can be easily realized using state space approximation of network

$$r_k(z) = r_{k-1}(z)\frac{z^{-1}-a}{1-az^{-1}} \quad (5)$$

Let denote the inverse z-transform of Γ_1(z,a) by l_1(k), Γ_2(z,a) by l_2(k) and similarly Γ_N(z, a) by l_N (k) and this forms a set of Laguerre functions in discrete domain shown in a vector form as

$$L(k) = [l_1(k) l_2(k) \dots\dots l_N(k)]^T \quad (6)$$

Taking advantage of Eq. (4) the set of discrete time Laguerre functions validates the following difference Eq. (7)

$$L(k+1) = A_l L(k) \quad (7)$$

Where, A_l is (N×N) matrix and it is a function of parameters a and $\beta = (1-a^2)$. By analysing the Laguerre equations and using the following z transform properties given in Table 1

Table 1. Properties of Z Transform

Sr. No.	Z-Domain	Discrete-time
1	X(z)	X(k)
2	X(z)H(z)	$\sum_{m=0}^{\infty} x(m)h(k-m)$
3	$\lim_{z\to 1} X(z)$	X(0)

One can obtain the initial condition given below [24].

$$L(0)^T = \sqrt{\beta}[1 - a \;\; a^2 - a^3 \dots (-1)^{N-1}(a)^{N-1}] \quad (8)$$

For N=5 L(0) and A_l will become

$$A_l = \begin{bmatrix} a & 0 & 0 & 0 & 0 \\ \beta & a & 0 & 0 & 0 \\ -a\beta & \beta & a & 0 & 0 \\ a^2\beta & -a\beta & \beta & a & 0 \\ -a^3\beta & a^2\beta & -a\beta & \beta & a \end{bmatrix}; L(0) = \sqrt{\beta}\begin{bmatrix} 1 \\ -a \\ a^2 \\ -a^3 \\ a^4 \end{bmatrix}$$

3 Implementation of MPC

Implementation of MPC for speed control of DC motor can be achieved by applying the System Model shown in Figure 1 [27]. This System Model illustrates the inter connection of different subsystems to realize a control system. The first subsystem is the input transducer which converts the input to the form compatible with the controller. Here, input is the desired speed for the DC motor and output is the actual speed. The controller estimates the control signal to drive the plant i-e DC motor at the desired speed. The output is given back to

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

the controller as feedback via output transducer. It measures the output and transforms it to a compatible form for the controller.

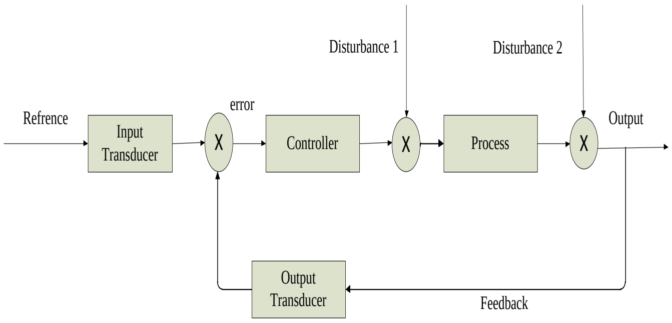


Figure. 1. System Model.

3.1 Model Predictive Controller

MPC for the speed control of DC motor is implemented by following the sequence of steps as shown in MPC flow diagram in Figure 2.

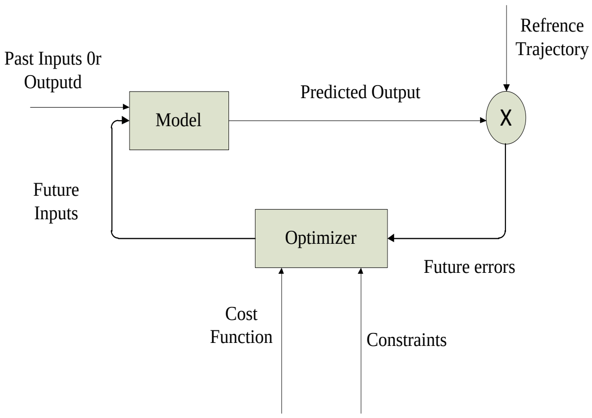


Figure. 2. MPC Flow Diagram

The MPC flow diagram consists of two blocks one is system model and other is optimizer. The initial inputs and outputs in terms of voltage and speed are given to the system model and it will give the predicted output speed. Output will be compared with the reference speed and the future errors will be given to the optimizer. The optimizer in presence of cost function and constraints optimizes the future control signal. The control signal will be further applied as an input to the system model to predict the output and this procedure will remain continue until the desired steady state is achieved.

3.2 System Model

Perfection of any controller is based on how efficiently the dynamics of a system have been modeled [28]. DC motor is not only frequently used in industrial motion control systems but also provide a standard platform for researchers to test various control algorithms [29]. There are four major types of DC motor. They are Shunt DC motor, Series DC motor, Permanent Magnet DC motor (PMDC) and Compound DC motor. PMDC is used in the research work. The schematic of PMDC is shown in Figure 3. V_m is the applied voltage, Rm is the armature resistance and L_m is the armature inductance. V_b is the back emf generated by the motor. Motor torque, load torque, moment of inertia and speed are T_m, T_L, J_m and ω_m respectively. The mathematical model of DC motor is obtained by applying electrical and mechanical equations to the electromechanical system i-e DC motor and then putting the values of internal parameters of DC motor in the transfer function to get exact model to be used for MPC. The internal parameters of DC motor are shown in Table 2. [30].

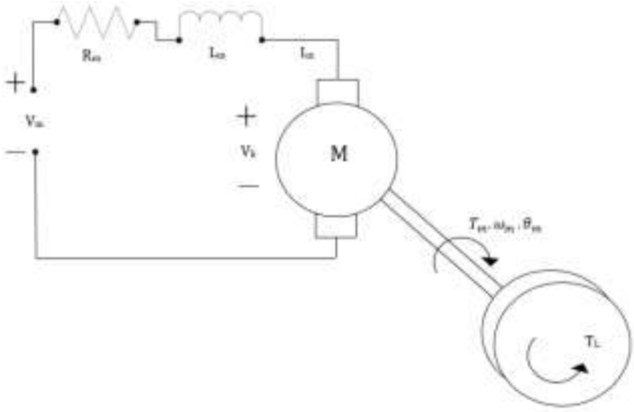


Figure. 3. Schematic Representation

Apply KVL to the loop in above Figure 3 to get electrical equation for the system

$$V_m = R_m I_m + L_m \frac{di(t)}{dt} + k_m \omega_m \quad (9)$$

Where $V_b = k_\omega$ and k_m= Motor torque constant. Motor is an electromechanical system can be represented by the Eq. (10).

$$\frac{J d\omega_m(t)}{t} = T_m - T_L - B_m \omega_m \quad (10)$$

Here T_L is assumed zero to simplify transfer function modeling, B_m is damping friction and $T_m = k_m I_m$

After applying Laplace transform to Eq. (9) and (10), simultaneously solve them to get the transfer function model as shown in Eq. (11).

$$\frac{\omega_m}{V_m} = \frac{k_m}{[(L_m J_m)S^2 + (R_m J_m + B_m L_m)S + (k_m^2 + R_m B_m)]} \quad (11)$$

Substituting DC motor parameters from Table 2 in Eq. 11, the transfer function becomes

$$\frac{19.607}{[(0.00014)S^2 + (0.0049)S + 1]} \quad (12)$$

Table 2. Internal Parameters of DC Motor [8]

Sr. No.	Parameter	Value
1	Moment of inertia	$J_m = 0.00025\ N_m/rad/s^2$

2	Damping factor	B_m = 0.0001 N_m/rad/s
3	Armature resistance	R_m = 0.5 Ω
4	Armature inductance	L_m = 1.5 mH
5	Torque constant	K_m = 0.05 N_m/A
6	Rated voltage	V_m = 12 V
7	Rated speed	2400 RPM

This transfer function is discretized by some sampling interval. The sampling interval of a control system is calculated by using a rule of thumb given below [31]

$$\frac{1}{T} > 30 \times BW\ of\ System$$

Where, BW is the Bandwidth of the system and for second order system it can be calculated by Eq. (13) [26].

$$BW = \omega_n \sqrt{1 - 2\zeta^2 + \sqrt{2 - 4\xi^2 + 4\xi^4}} \quad (13)$$

Where, ω_n is natural frequency and ζ is the damping ratio of the system so values of these two variables can be calculated by comparing Eqs. 12 and 14

$$G(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (14)$$

The calculated values of ξ and ω_n are 0.206 and 84.515 respectively. By putting the values in Eq. (13) BW is 127.296 Hz and it is further used to calculate the sampling time of 261µs.

$$x_m(k+1) = \begin{bmatrix} 1.9908 & -0.9912 & 0.0045 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} x_m(k) + \begin{bmatrix} 0.0045 \\ 0 \\ 1 \end{bmatrix} u(k) \quad (15)$$

$$y(k) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x_m \quad (16)$$

The Eq. (15) and (16) represent the state-space model that will be used as a system model in Figure 2 for implementation of MPC.

3.3 Optimizer

MPC is an optimization based technique that optimizes control signal to keep the process output at the set point and within the predetermined system constraints [32].

$$\Delta U(k_i) = L(k)^T \eta \quad (17)$$

Where $L_i k^T$the Lagurre function and η is the parameter optimized at every sampling instant by the function shown in Eq. (18).

$$\eta = -(\sum_{m=1}^{N_p} \emptyset(m) Q \emptyset m^T + R_L)^{-1} (\sum_{m=1}^{Np} \emptyset(m) Q A^m) x(k_i) \quad (18)$$

Where, Q is the weight matrix, N_p is the prediction horizon and R_L is a diagonal matrix in the form the $R_L = r_w \times I_{NCxNC}$ ($r_w \geq 0$). r_w is a tuning parameter for desired closed loop performance. For the case that $r_w = 0$, the cost function is interpreted as the situation where there is no attention paid to how large the ΔU might be and goal would be solely to make the error $(Rs - Y)^T (Rs - Y)$ as small as possible. For the case of large r_w, the cost function is interpreted as the situation where the magnitude of ΔU is of concern which results in cautiously reduction of error $(Rs - Y)^T (Rs - Y)$ while ∅(m) is expressed by the following Eq. (19) and its number of rows is identical to the number of rows in η

$$\emptyset(m) = \sum_{i=0}^{m-1} A^{m-i-1} B_L(i) \quad (19)$$

Where m=2, 3, 4…..,N_p

4 Simulations based Performance Analysis

The MPC is being analyzed in four modes first is without constraints, the second is with input constraints, third is with output constraints and fourth is both with input and output constraints

4.1 Model Predictive Controller

The first step in implementation of MPC is to acquire state space model (SSM) of the DC motor and introduce the design parameters, reference point and constraints. Then MPC optimizes the control vector ΔU for output constraints after the input constraints are ensured to be within the predetermined range. Finally, that control signal is applied to the SSM of DC motor to predict the output and this procedure continues until desired set point is achieved. Flow chart of MPC algorithm is shown in Figure 4 [27].

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

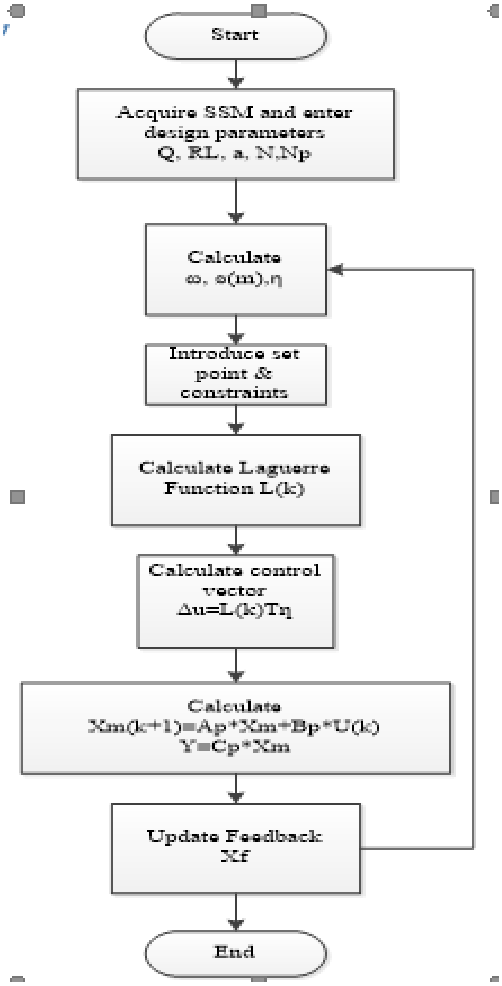


Figure. 4. Flow Chart of MPC Algorithm

4.2 MPC Analysis

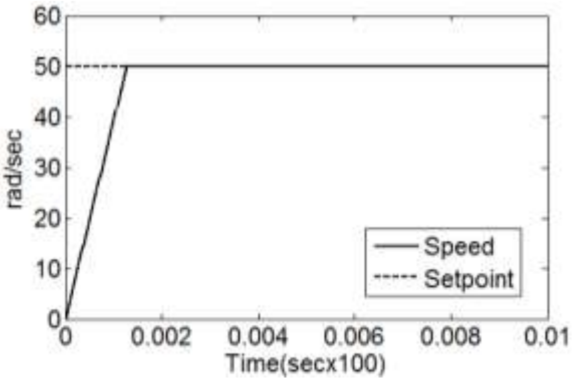
Simulations based MPC analysis is done in four modes. These modes are discussed below:

Mode 1: MPC implementation without constraints
Mode 2: MPC implementation with input constraints
Mode 3: MPC implementation with output constraints
Mode 4: MPC implementation with both input and output constraints
The MPC parameters used in simulations are N_p = 46, N_c = 2, a = 0.7 and performance parameter N=3

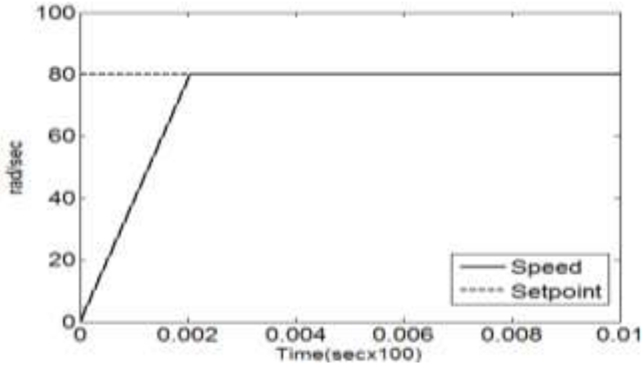
4.2.1 Mode 1: MPC Without Constraints

The MPC is simulated without constraints for two cases. First with set-point 50 rad/sec and second with set-point 80 rad/sec. The results are shown in Figure 5

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434



(a) Step response at 50 rad/sec



(b) Step response at 80 rad/sec

Figure. 5. Step response without constraints

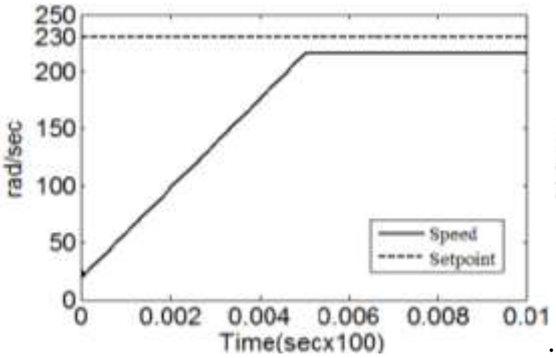
From above figure it is clear that MPC control the outputs at the desired speeds effectively

4.2.2 Mode 2: MPC With Input Constraints

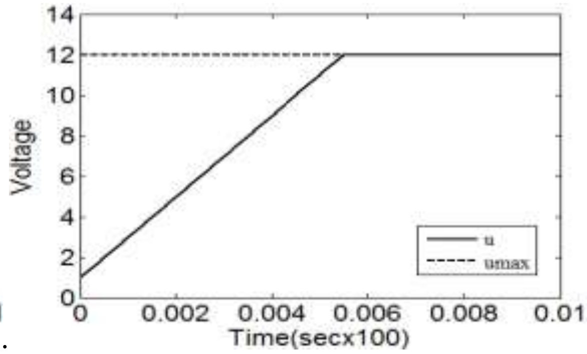
As maximum rated voltage of DC motor is 12V and to keep the motor out of dead-zone minimum 1V must be applied at motor. Therefore, two cases for input constraints imposed on control signal are presented. They are given below:

Case1: $U^{min} = 1V, U^{max} = 12V$ and Case 2: $U^{min} = 1V, U^{max} = 10V$

The simulation results for this mode are shown in Figure 6

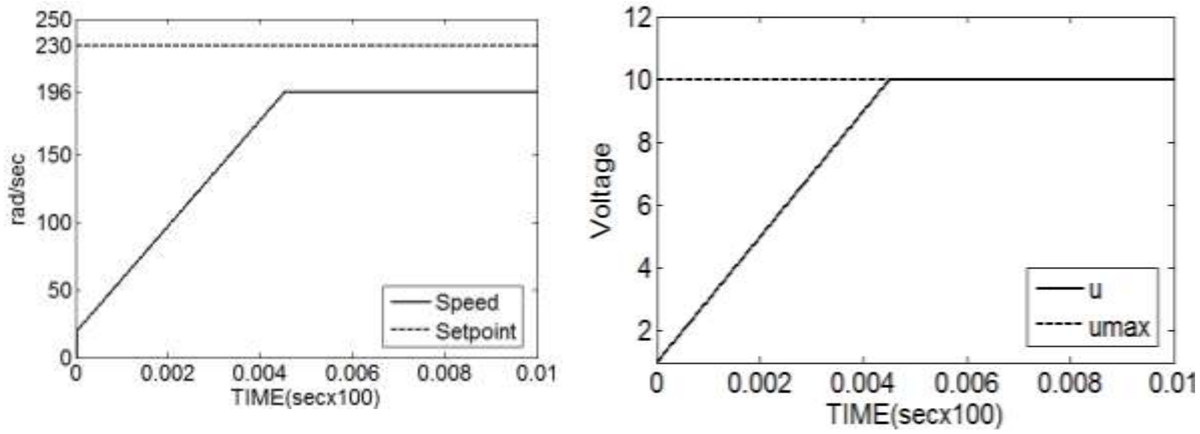


(a) Case 1: Output



(b) Case 1: Input

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434



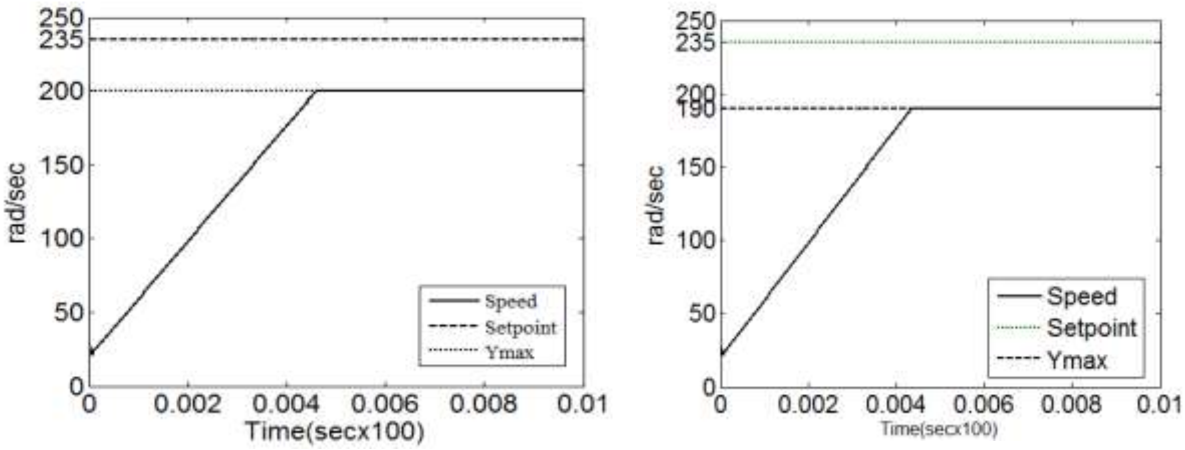
(a) Case 2: Output (a) Case 2: Input

Figure. 6. Step response with Input constraints

From Figure 6 it is clear that in both cases the input constraints are not violated. In first case due to input constraint applied at 12V the maximum speed limits to 225 rad/sec and in second case the input constraint was at 10V and maximum speed limits to 196 rad/sec.

4.2.3 Mode 3: MPC With Output Constraints

As the maximum rated speed is 235 rad/sec so the output constraint is implemented for two cases at 200 rad/sec and 190rad/sec. the simulation results are shown in Figure 7



(a) Case 1: Output (b) Case 2: Output

Figure. 7. Step response with Output constraints

From Figure 7 it is evident that in both cases speed does not violate the output constraint and MPC effectively handles the output constraints

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434
RESEARCH ARTICLE -ENGINEERNG TECHNOLOGY

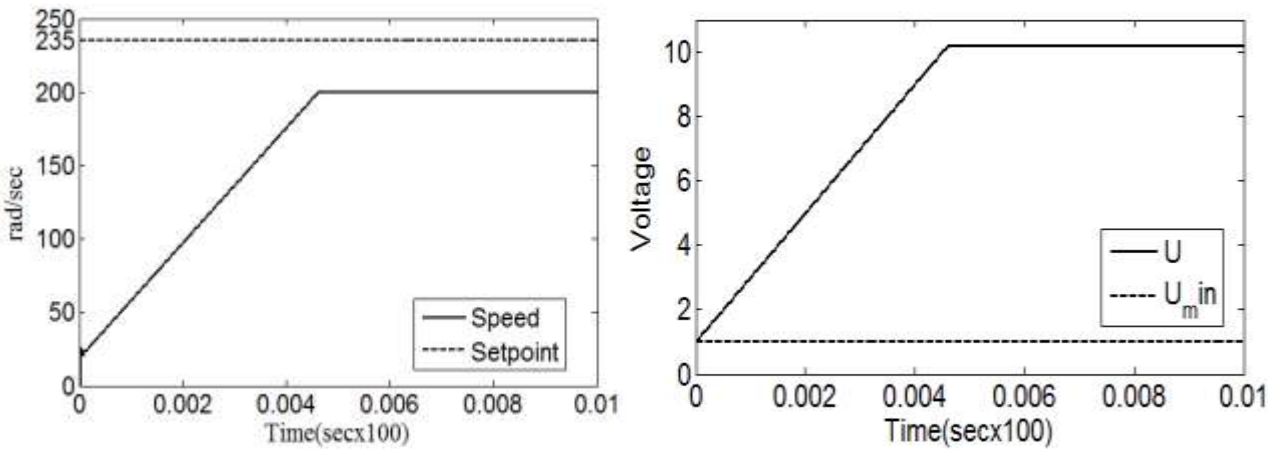
4.2.4 Mode 4: MPC with both Input and Output Constraints

To understand the handling of both input and output constraints simultaneously by MPC the input constraint is applied at minimum of applied voltage and output constraint is applied at rated speed 200 rad/sec while considering the set-point at 235 rad/sec. The constraints imposed on both input and output is given below:

$$U^{min} = 1V$$

$$Y^{max} = 200\ rad/s$$

The simulation results for this mode are shown in Figure 8



(a) Output (b) Input

Figure.8. Step response with both Input and Output constraints

From Figure 8 it is clear that both the input and output constraints are simultaneously handled effectively by MPC and none of the constraints are violated.

5 Hardware Implementation of MPC

Use The hardware implementation consists of microcontroller, motor driver, sensor, user interface, display unit, power supply and a DC motor. The microcontroller receives the actual speed of the DC motor as a feedback from optical encoder and then MPC generates PWM pulses to the motor driver for speed control of DC motor.

5.1 Block Diagram

The block diagram of the MPC speed controller is shown in Figure 9. It constitutes of an input transducer to transform the desired speed reference to a form compatible with the controller, PIC microcontroller generates required PWM for speed control using MPC algorithm. Motor driver regulates voltage at terminals of DC motor. Output transducer or sensor measures speed and convert it into the form used by the controller and give the result to the controller for estimation of PWM using MPC and display unit shows the result of actual speed and the applied PWM.

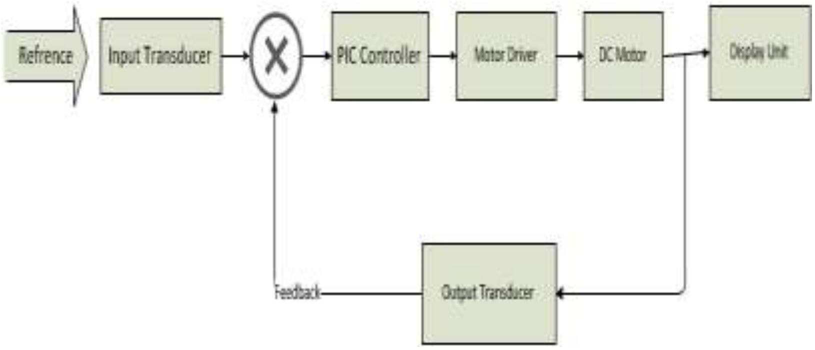


Figure.9. Block Diagram of Hardware Implementation

5.1.1 Input Transducer/User Interface

Keypad is used as a user interface unit. It is used to provide a reference or desired value of speed to be achieved by the motor in the form compatible to the microcontroller. It also provides options to select four different modes for speed control of DC motor in constrained environment.

5.1.2 Microcontroller

DSPIC30F4011 is a general purpose 16-bit digital controller and it provides precise control by taking advantage of combining the calculation capability of digital signal processor and controlling capability of PIC microcontroller. It has high performance modified RISC architecture. The controller has a C compiler optimized instruction set mechanism with dynamic addressing modes [33]. It is a low power consumption device with wide range of operating voltage from 2.5V - 5.5V. Controller is used to generate PWM pulses to regulate the motor speed by getting feedback from encoder

5.1.3 Motor Driver

Controller generates 5V PWM pulses and is applied at the driver circuit. Driver circuit regulates voltage at the DC motor [32]. The Driver circuit used in this project is Dual H Bridge IC L293D. Motor is connected between pins 3 and 6 of L293D. A separate supply for running motor is applied at pin No. 8 of L293D. VCC is applied at pin No.4 and 16 while pin No 5 is grounded [33].

5.1.4 DC Motor

DC Motors have numerous industrial and domestic applications and can be selected in various sizes and rates as per applications. The DC motor used in the research work is WLC-C25MD with built in optical encoder. The specifications of this motor are shown in Table 3

Table 3. Specifications of DC Motor

Sr. No.	Specification	Rated Value
1	Voltage	12 V
2	Speed	1200 RPM
3	Power	1W

5.1.5 Display Unit

LCD is used to show real time results of speed. The LCD used is 16*2 display LCD module. It consists of 16 rows and 2 columns of 5*8 dot matrices. It has 16 pins.

5.1.6 Output Transducer/Sensor

Optical encoder is used to measure the real speed of motor and converts it into a form compatible to microcontroller. This information is given back to the controller for the optimization of control signal to generate accurate PWM to drive the motor at desired speed.

5.2 Pulse Width Modulation (PWM)

The output from microcontroller to motor driver is PWM. MPC optimizes the PWM duty cycle to control the power supplied to DC motor to get the desired speed. In this research work DSPIC30F4011 microcontroller is used which has a PWM window from 0 to 8000. 0 means fully “Off” 0% duty cycle while 8000 means fully “On” 100%duty cycle and any value between them is a proportion of this range.

5.3 Results

Hardware results are shown here in four modes. The first mode is speed control without constraints, second is with input constraints, third is with output constraints and the last is with both input and output constraints. The parameters and constraints applied in all four modes are shown in Table 4.

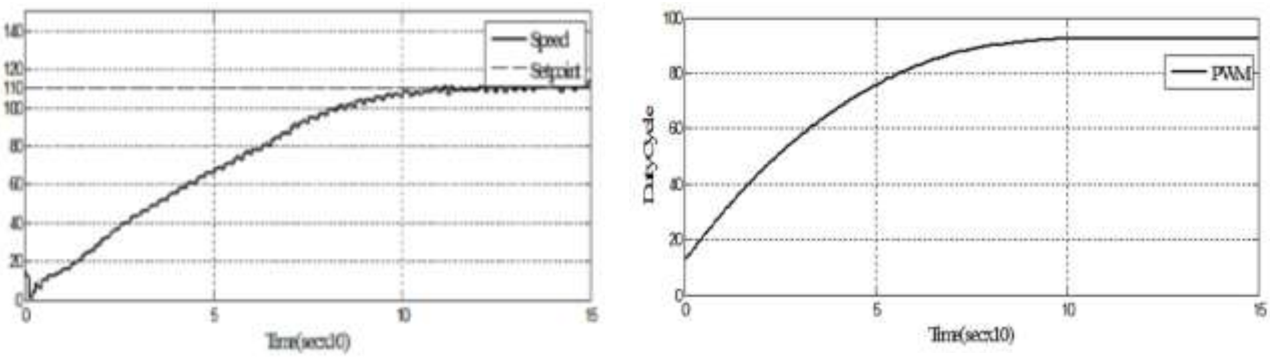
Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434

Table 4. Parameters and Constraints used in Hardware Implementation

Mode	Prediction Horizon (N_p)	Control Horizon (N_c)	Performance Parameter (R)	Laguerre Pole (a)	Constraints Input (U^{min}),(U^{max})		Constraints Output (Y^{max})
1	46	2	0.3	0.7	Nil	Nil	Nil
2	46	2	0.3	0.7	12.5 PWM	80 PWM	Nil
3	46	2	0.3	0.7	Nil	Nil	100 rad/sec
4	46	2	0.3	0.7	12.5 PWM	Nil	100rad/sec

5.3.1 Mode 1: Without Constraints

In this mode the MPC has been implemented on DSPIC30F4011 controller without any constraints imposed to evaluate MPC for simple step response. The real-time results are shown in Figure 10.



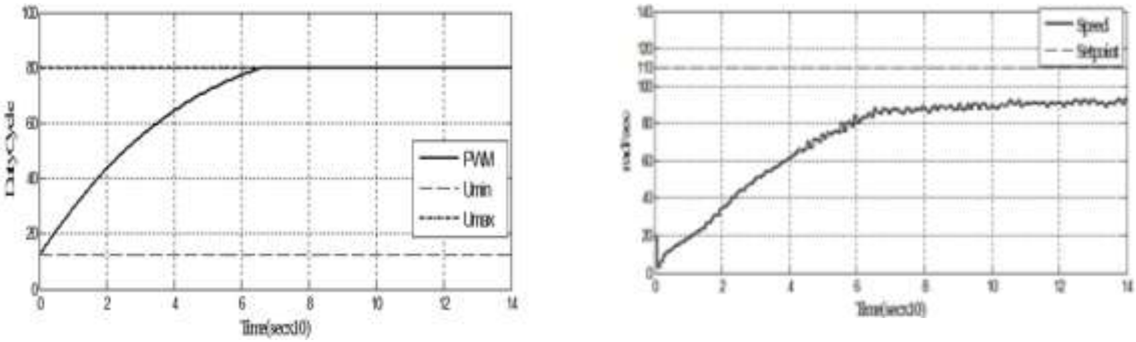
(a) Output (b) Input

Figure.10. Speed Control Without Constraints

5.3.2 Mode 2: With Input Constraints

In this mode the MPC has been implemented onDSPIC30F4011 controller for speed control of DC motor with input constraints imposed. As DC motor has maximum rated voltage of 12V so PWM window should not be more than 8000 which results in 100% duty cycle so to drive motor within the saturation limit the maximum allowable duty cycle is 80%. While to keep the motor state out of dead-zone minimum 1V is required which results in 12.5% duty cycle so constraints in this mode are U^{min}=12.5 and U^{max}=80 .The real time results are shown in Figure 11

Journal of Engineering Technology
Volume 6, Issue 1, Jan. 2018, PP. 415-434



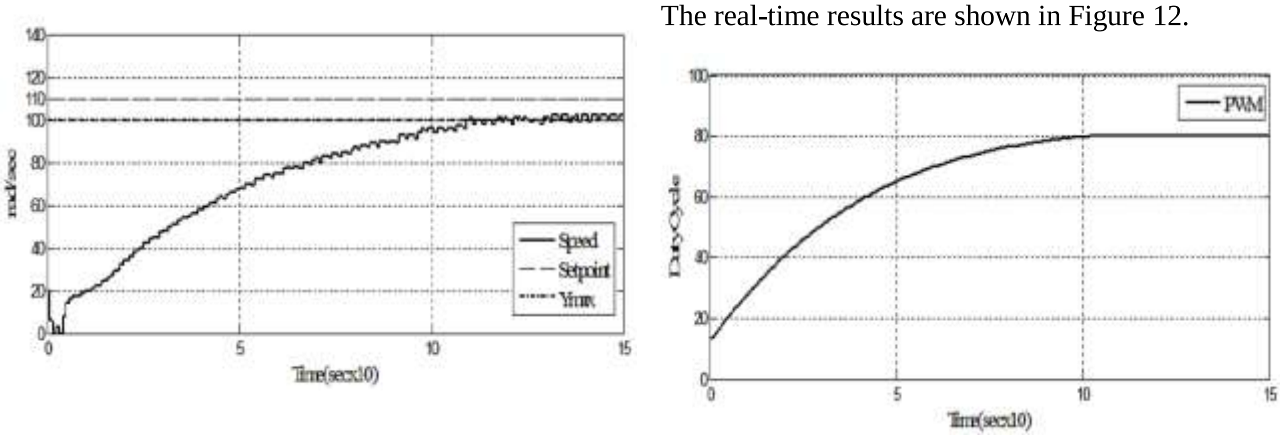
(a) Output (b) Input

Figure.11. Speed Control with Input Constraints

The real-time results have shown that the input constraints are not violated and MPC maintains the applied PWM with in the applied constraints.

5.3.3 Mode 3: With Output Constraints

Maximum rated speed of DC motor is 120 rad/sec so to keep the speed within the rated value constraints on output speed are imposed which is Y^{max} = 100rad/sec. The real-time results are shown in Figure 12.



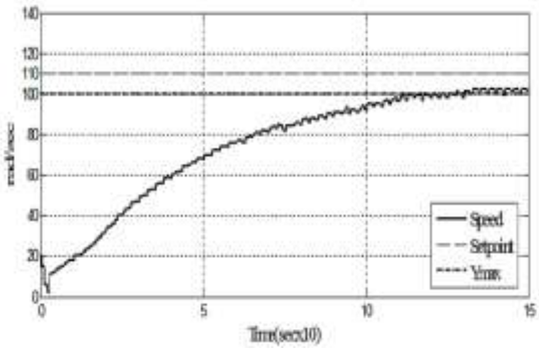
(a) Output (b) Input

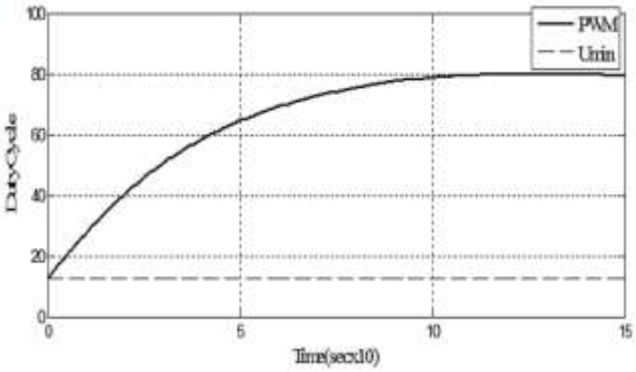
Figure.12. Speed Control with Output Constraints

The results from the Figure 12 have shown that the output constraint is not violated and MPC maintains output in Pre-determined range

5.3.4 Mode 4: With both Input and Output Constraints

In this mode the constraints are imposed on both PWM and speed to evaluate performance of MPC in terms of constraints handling. The constraints imposed are U^{min} =12.5 PWM and Y^{max} = 100rad/sec. The real time results are shown in Figure 13.





(a) Output

(b) Input

Figure.13. Speed Control with both Input and Output Constraints

The results from Figure 13 have shown that none of the input or output constraints are violated and MPC maintains the output in pre-determined range effectively.

5.3.5 MPC Optimization in Presence of Load Torque

MPC has been evaluated in terms of load torque by using magnetic break. The set point considered is 110 rad/sec. The results are shown in Figure 14

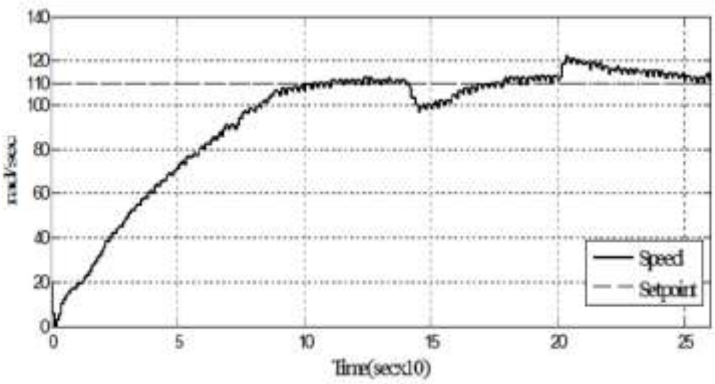


Figure.14. MPC Optimization in Presence of Load Torque

The results have shown that when load torque is induced at output through magnetic brake the speed first dips at 97 rad/sec but MPC optimizes the control signal to maintain the desired speed and when induced load is removed the speed increases sharply and reaches to 122 rad/sec but again MPC optimizes the control signal to maintain the desired speed.

6. Conclusion

In this research work, real-time implementation of MPC on a simple 16-bit PIC microcontroller for speed control of a DC motor is done. As classical MPC has high computational burden and not appropriate for implementation on an ordinary microcontroller so at first, Laguerre functions are used to reduce computational burden and to improve the feasibility for embedded implementation. The simulations of MPC have been analyzed in four modes i.e. without constraints, with input constraints, with output constraints and with both input and output constraints. In all four modes MPC effectively regulates speed without violating predetermined constraints. Finally, MPC is real-time implemented on a 16-bit PIC microcontroller for speed control of a DC motor and tested in presence of input constraints, output constraints and load torque. Again, MPC performs better and effectively regulates speed in presence of constraints and load torque. Moreover, this paper advocates the idea of implementing MPC on a simple microcontroller for SISO application. In future prospect this high efficient MPC can be further evaluated by implementation on 16-bit microcontroller for Multiple Input and Multiple Output (MIMO) system and on 8-bit microcontroller for SISO system.

References

[1] K. R. Das, D. Das, J. Das, "Optimal tuning of PID controller using GWO algorithm for speed control in DC motor" IEEE International conference on soft Computing Techniques and Implementation, pp. 108-112, June. 2016.

[2] K. H. ANG, G. Hong, Y. Li, "PID Control System Analysis, Design and Technology" IEEE Trans. Control Systems, vol. 13, pp. 559-576, July. 2005

[3] S.W. Khubulkar, A.S. Chopade, A.S. Junghare, "Design and Tuning of Fractional order PID controller for speed control of Permanent Magnet Brushless DC motor" IEEE International Conference on Control, Measurement and Instrumentation, pp. 326-330, Jan. 2016.

[4] S.Aslam, S.Hannan. W.Zafar, U.Sajjad, "Temperature control of Water-Bath system in presence of constraints by using MPC", International Journal of Advanced and Applied Sciences, vol. 3, pp. 62-68, 2016.

[5] K. S. Holkar, L.M. Waghmere, K.K. Wagh, "An Overview of Model Predictive Control", International Journal of Control and Automation, vol. 3, pp.47-64, December, 2010

[6] A.A. Ahmed, "Fast-Speed Drives for Permanent Magnet Synchronous Motor Based on Model Predictive Control" IEEE International Conference on Vehicle Power and Propulsion, pp. 978-983, December. 2015.

[7] J. Richal et, ". Industrial application of Predictive functional control", In: Nonlinear Model Predictive Control, Software and Applications. Loughborough, U.K, 2007

[8] Y.Z. Zhu, XU, J. Zhao, K. Han, J. Ojan, W. Li, "Development and Application of an Integrated MPC Technology", In Proceedings of the 17th IFAC World Congress, Korea, pp.6962-6967, 2008

[9] J. Rossister, R.H. Bishop. Model based Predictive Control a Practical approach. CRC Press, 2005.

[10] D.E. Seborg, D.A. Mellichamp. Process , Dynamics and Control. WILEY Publisher, 2010

[11] J. Maciejowski. Predictive Control with Constraints. Pearson Education, 2001

[12] M.V. Kathore, V. Balakrisham, M. Morari, "Robust Constrained Model Predictive Control using Linear Matrix Inequalities" Automatica, vol. 32, pp. 1361-1379, October, 1996

[13] A.A. Kherji, F. Bouani, M. Ksouri, "A microcontroller Implementation of Model Predictive Control" World Academy of Science, Engineering and Technology, vol. 56, pp. 662-665, 2011

[14] A.K. Abes, F. Bouani, M. Kasouri, “A microcontroller Implementation of Constrained Model Predictive Control” World Academy of Science, Engineering and Technology, vol 5, pp. 662-665, 2011
[15] L. Juan, G.A.C. Jarez. C.K. Eric, “Parallel MPC for Real-Time FPGA-based Implementation”, . In: Preceedings of the 18th IFAC World Conference. Milano, Italy, pp. 1338-1343, 2011.
[16] S.R. Arief, S. Raphael, “Model Predictive Control on a Programmable Logic Controller for DC motor Speed Control”, International Conference on Electrical Engineering and Informatics, pp. 1-4, 2011
[17] G. Valencia, J.A. Rossister, “.Novel programmable logic controller implementation of a predictive controller based on Laguerre functions and multiparametric solutions” IET Control Theory and Applications, vol. 6, pp. 1003-1014, 2011
[18] G.Frison, D. Kwarme, L.Imsland and J.B.Jorgenson, “Efficient Implementation of solvers for linear Model Predictive Controller on Embedded Devices”, IEEE International Conference on Control Applications. Antibes, France, pp. 1954-1959, 2014
[19] Malouchee, I. Kherji, F. Bouani, “Automatic Model Predictive Control Implementation in a Higher Performance Microcontroller” . In : Proceedings of 12th International Conference on Systems, Signals and Devices, Tunisia, pp. 1-6, 2015.
[20] Malouchee, I. Kherji, F. Bouani, “Automatic Model Predictive Control Implementation in a Higher Performance Microcontroller” . In : Proceedings of 12th International Conference on Systems, Signals and Devices, Tunisia, pp. 1-6, 2015.
[21] M.Short, F.Abugchem, “A microcontroller-Based Adaptive Model Predictive Control Platform for Process Control applications”, Electronics, vol. 6, pp. 1-17, 2017.
[22] K.A. Holkar, L.M. Waghmere, “Discrete Model Predictive Control for DC drive using Orthonormal Basis function” UKACC International Conference on Control. Coventry, U.K, pp.1-6, 2010
[23] S. Aslam, S. Hannan, W. Zafar, U. Sajjad, “Implementation of PID on PIC24F series Microcontroller for speed control of a DC motor using MPLAB and Proteus” Advances in Science and Technology Research Journal, vol. 10, pp. 40-50, September, 2016.
[24] S.Sahoo, B.Subhudi, G. Pranda, “Optimal Speed control of DC motor using linear quadratic regulator and model Predictive Control, July. 2016.
[25] S.K.Suman, V.K.Giri, “Speed control of DC motor using optimization techniques based PID controller” IEEE International Conference on Engineering and Technology, March. 2016.
[26] L. Wang. Model Predictive Control system design and implementation using MATLAB. Springer, 2009
[27] N.S. Nise. Control System Engineering 6th Edition. John Wiley and Sons, 2010.
[28] S. Mohatty, “Artificial Neural Network based System Identification and Model Predictive Control of a Flotation Column” Journal of Process Control, vol.19, pp. 991-999, 2009
[29] N. Pajapati, V.P. Patel, “Explicit Model Predictive Control of fast dynamic system” Journal of Research in Engineering and Tchnology, vol. 2, 692-695, 2013
[30] B. Hamed, M. Al-Mobaid, “Fuzzy PID Controllers using FPGA technique for real time DC motor speed Control” International Journal of Intelligent Control, vol. 2, pp. 233-240, 2011
[31] C. Mark. Discrete Systems. Teutorial Sheet. 2-30, 2014
[32] K. Kraig. Scond Order Dynamic Systems. Lecture Notes. 1-39, 2011

[33] S. Priyan, P. Salvaraj, "A Wireless Speed Control of AC Drive System" International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering., vol. 3, pp. 448-456, 2014.

[34] S.K. Arvind, T.A. Arun, "Speed Control of DC motor using PIC 16F877A Microcontroller" Multidisciplinary Journal of Research in Engineering and Technology, pp 223-234, 2008.