Chapter from the book *Electric Vehicles - Modelling and Simulations*
Downloaded from: http://www.intechopen.com/books/electric-vehicles-modelling-and-simulations

Interested in publishing with InTechOpen?
Contact us at book.department@intechopen.com
1. Introduction

As already stated for the reasons given in a previous chapter a good continuous time model, of low complexity, of a BLMD system is essential to adequately describe mathematically the PWM inverter switching process with dead time and subsequent binary waveform generation in terms of the switching instant occurrences for accurate computer aided design (CAD) of embedded BLMD model simulation in proposed electric vehicle (EV) propulsion systems. In this chapter a complete software model of the BLMD system as a set of difference equations representing subsystem functionality, the organization of these subsystem activities into flowchart form and the processing details of these modular activities as software function calls in C-language for simulation purposes (Guinee, 2003) is presented.

Furthermore in this the second chapter, concerning BLMD model fidelity for EV applications, BLMD model simulation accuracy for embedded EV CAD is next checked for a range of restraining shaft load torques via numerical simulation and then extensively compared and benchmarked for accuracy against theoretical estimates using known manufacturer’s catalogued specifications and motor drive constants (Guinee, 2003). Model simulation accuracy is further substantiated and validated through evaluation of the shaft velocity step response rise time when cross checked against (i) experimental test data and (ii) that evaluated from the catalogued performance index relating to the brushless motor dynamic factor (Guinee, 2003). Numerical simulation with outer velocity loop closure is used to demonstrate the accuracy of the completed BLMD reference model, based on established model confidence in torque control mode, in ASD configuration when compared with experimental test data.

In addition to the BLMD model structure presented in the previous chapter for actual drive emulation two innovative measures which relate to increased drive performance are also provided. These novel techniques (Guinee, 2003), which include
i. inverter dead time cancellation and
ii. motor stator winding impedance angle compensation,
are encapsulated within the BLMD model framework and simulated for validation purposes
and prediction of enhanced drive performance in EV systems. An approximate analysis is
given to support the approach taken and verify the performance outcome in each case.
In the first of these BLMD performance enhancements a novel compensation method has
already been presented in the first chapter to offset the torque reduction effects of inverter
delay during BLMD operation. This simple expedient relies on the zener diode clamping of
the triangular carrier voltage during the carrier waveform comparison with the modulating
current control signal in the comparator modulator to nullify power transistor turnon delay.
This approach obviates the need for separate compensation timing circuitry in each phase as
required in other schemes. The accuracy of this methodology is supported by current
feedback, EM torque generation and shaft velocity trace simulation when compared with
similar traces from the BLMD benchmark reference model with the effects of the inverter
based drive trigger delay neglected.

The second proposed innovative improvement, presented in this chapter, relates to the
progressive introduction of commutation phase lead with increased shaft speed as BLMD
impedance angle compensation which forces the impedance angle to the same value as the
internal power factor angle. This effect maintains zero load angle between the stator
winding terminal voltage and the back emf. It also results in rated load torque delivery at
lower shaft speeds with minimal rise time, overshoot and settling time in the generated
torque for a range of torque demand input values. This novel technique greatly enhances the
dynamic performance of the embedded BLMD prime mover in EV applications without
overstressing mechanical assembly components during periods of rapid acceleration and
deceleration. The incorporation of this novel impedance angle compensation technique thus
minimizes component wear-out such as gear boxes, transmission shafts and wheel velocity
joints and consequently enhances overall EV reliability improvement. BLMD simulation is
provided in torque control mode at rated torque load conditions, for the actual drive system
represented, with and without impedance angle compensation to gauge model performance
accuracy over a range of torque demand step input values.

2. BLMD model structure and program sequence of activities

The BLMD model structure is composed of interconnected subsystems with feedback as
shown in Figure 1, of varying complexity according to physical principles. Consequently it can
be described by a discrete time configuration of first order digital filter realizations for linear
elements cascaded with difference equations representing nonlinear PWM inverter behaviour
into a complete software model for simulation purposes as illustrated in Figure 2. The BLMD
model program is organized into a sequence of software activities, coded in C-language as
function calls, representing the functionality of various subsystem modules shown as the
flowchart in Figure 3. All subsystem output (o/p) variable quantities in the cascaded activity
chain are assumed to remain constant, once computed irrespective of feedback linkage,
throughout the remainder of the time step interval Δt_k based on the simulation sampling rates
(1/Δt_k) chosen from considerations given in section 3.1 of the previous chapter. The essential
features of the BLMD model program in Figure 3 can be explained by means of the linked
modular software configuration encoded as the functional block sequence in Figure 2 along
with the appropriate C-language code segments illustrated in Figure 4.
Extended Simulation of an Embedded Brushless Motor Drive (BLMD) System for Adjustable Speed Control Inclusive of a Novel Impedance Angle Compensation Technique...
Fig. 2-A. Software Functional Block Diagram (Guinee, 2003) of a BLMD System
Extended Simulation of an Embedded Brushless Motor Drive (BLMD) System for Adjustable Speed Control Inclusive of a Novel Impedance Angle Compensation Technique...
Fig. 2-C. Software Functional Block Diagram of a BLMD System
Fig. 3-A. Program Flow Diagram (Guinee, 2003) for BLMD Model Simulation
Fig. 3-B. Program Flow Diagram for BLMD Model Simulation
2.1 BLMD model simulation
Numerical simulation commences with the declaration of known BLMD system parameters followed by a declaration with initialization of variables and three phase (3Φ) arrays for global usage, over the program linked function call sequence, as outlined in the code blocks shown in Figure 4-1.

```
// Define BLMD System Parameters in Fig.5 ≈
// Define all filter constants:
K_T = 1.0; τ_T = 222μS; // Torque Demand Filter H_T
K_I = 1.0; τ_I = 100μS; // Current Demand Filter H_DI
K_F = 5.0; τ_F = 47μS; // Current Feedback Filter H_FB
K_W = K_I; // Current Feedback Factor
K_C = 19.5; τ_D = 225μS; τ_F = 1.5mS; // Current Controller G_I
H_v = 13.5x10^-3; α = √2; ω_n = 2x10^3 rad.s^-1; // Velocity Filter H_V
K_p = 1.0; τ_p = 28.6μS; // RC Delay
r_S = 0.75Ω; L_S = 1.94mH; p = 6 pole pairs; // Motor Winding
K_s = K_p = 0.3 // Motor torque & Back EMF Constants
J_m = 3 kg.cm^2; B_m = 2.14x10^-3 Nm.rad^-1; // Motor Dynamics
U_M = 310 Volts; V_th = 0; V_F = 10 Volts; // Voltage Constants
f_S = 5.7kHz; A_p = 6.9 Volts; // Carrier Waveform Constants

// Define Global Variables ≈
stepk // k^th iteration time step
delta_t // Fixed simulation Time Step Size Δt = (t_k - t_k-1)
delt // Variable simulation Time Step Size Δt_i
tdel // Carrier v_f(t-t_d) time delay t_d
osc // Carrier amplitude v_f
i_torq_dem // motor i/p torque demand T^k_d
f_torq_dem // Filtered torque demand T^k_f
mot_shaft_pos // Motor shaft position θ^k_m
mot_shaft_vel // Motor shaft velocity ω^k_m
tot_torq // Net drive torque T^k_T

// Define Global Array variables for j=1 to 3 ≈
ppsinkm[j] // 3-phase commutation psink[j]
tor_sink[j] // 3-phase torque demand T^k^m[j]
idenk[j] // 3-phase current demand i^k^m[j]
ifbk[j] // Filtered current feedback i^k_f[j]
jbk[j] // Current feedback i^k_b[j]
vmppwmk[j] // Current controller o/p v^k_c[j]
pwmtrk[j] // Modulator o/p v^k_m[j]
base_drk[j] // Basedrive o/p v^k_b[j]
bar_base_drk[j] // Complementary basedrive o/p v^k_b[j]
ph_voltk[j] // Stator winding Phase voltage v^k_p[j]
back_emfk[j] // Motor back EMF v^k_b[j]
ph_curtk[j] // Stator winding Phase current i^k_p[j]

// Function initializes all global variables and arrays to 0.0
init_vars();
```

Fig. 4-1. Declaration and Initialization (Guinee, 2003)
Fig. 5. Network Structure (Guinee, 1999) of a Typical Brushless Motor Drive System
All first order linear system discretization is accomplished by complex variable substitution of the Euler backward rectangular rule using the Z transform. The alternative filter discretization process using Tustin’s bilinear method (Franklin et al, 1980) or the trapezoidal integration rule (Balabanian, 1969) can also be used but with negligible observable differences at the step size $\Delta t$ chosen. Resultant digital filter implementation for simulation purposes is facilitated by transfer of the appropriate filter time constant and gain coefficients using the C-language ‘structure’ mechanism in the function call `setup_fo_filt()` illustrated in Figure 4-2.

```c
typedef struct
{
    double k;
    double n0;
    double n1;
    double d0;
    double d1;
} tf1;

/** First Order System Discretization Process discrete1() **/
/* Euler's Backward Integration Rule */
    double delt;
tf1 discrete1(tf1 temp)
{
    tf1 discrete;
    double at,bt,ts;
    double k,n0,n1,d0,d1;
    ts=delt;
    k=temp.k;
    n0=temp.n0; n1=temp.n1;
    d0=temp.d0; d1=temp.d1;
    at=ts*n0; bt=ts*d0;
    n0=at+n1; d0=bt+d1;
    if(n1!=0.0) {
        discrete.k=k*n1/d1;
        discrete.n0=n0/n1; discrete.n1 = -1.0;
        discrete.d0=d0/d1; discrete.d1 = -1.0;
    } else {
        discrete.k=k*n0/d1;
        discrete.n0=0/n1; discrete.n1 = -1.0;
        discrete.d0=d0/d1; discrete.d1 = -1.0;
    } return discrete;}

/** Tustin's Bilinear Method */
/* s ~ \frac{1}{T} (1 - z^{-1}) */
    tf1 discrete1(tf1 temp)
{
    tf1 discrete;
    double at,bt,ts;
    double k,n0,n1,d0,d1;
    ts=delt;
    k=temp.k;
    n0=temp.n0; n1=temp.n1;
    d0=temp.d0; d1=temp.d1;
    at=ts*n0; bt=ts*d0;
    n0=at+n1; d0=bt+d1;
    if(n1!=0.0) {
        discrete.k=k*n1/d1;
        discrete.n0=n0/n1; discrete.n1 = -1.0;
        discrete.d0=d0/d1; discrete.d1 = -1.0;
    } else {
        discrete.k=k*n0/d1;
        discrete.n0=0/n1; discrete.n1 = -1.0;
        discrete.d0=d0/d1; discrete.d1 = -1.0;
    } return discrete; }

Fig. 4-2. Linear Subsystem Discretization

www.intechopen.com
Before proceeding with model program execution in the $k^{th}$ time step ($t_{k-1} \rightarrow t_k$) all global variables and arrays are captured in the function call $capture\_filt\_out()$ for later reinstatement, during accurate resolution of the width modulated pulse edge transition time via the regula-falsi method, with $restore\_filt\_out()$ in Figure 4-3.

```c
void capture_filt_out(void)
{
    int j;
    cftorq_dem=ftorq_dem; cosc=osc;
    ctot_torq=tot_torq; cmot_shaft_vel=mot_shaft_vel;
    cmot_shaft_pos=mot_shaft_pos;
    for(j=1;j<=3;j++) {
        cidemk[j]=idemk[j]; ctor_sink[j]=tor_sink[j];
        cvmpwmk[j]=vmpwmk[j]; cpwmtrk[j]=pwmtrk[j];}
    return;
}
```

```c
//Function reinstates arrays and global variables
ftorq_dem = cftorq_dem; //etc. for all other variables
for (j=1;j<=3; j++) {
    idemk[j] = cidemk[j]; tor_sink[j]=ctor_sink[j];
    vmpwmk[j]=cvmpwmk[j]; pwmtrk[j]=cpwmtrk[j];
}
```

Fig. 4-3. Variable capture and restoration

The instruction code group $run\_to\_pwm\_sw()$ processes the sequence of BLMD software activities up to the comparator modulator o/p using the following list of function calls in Figures 4-1 and 4-4.

- $tdemf()$ filters the i/p torque demand signal $torq\_dem$ with o/p $ftorq\_dem$.
- $mot\_commutator()$ establishes the 3-phase reference $psink[j]$, from the computed shaft rotor displacement $\theta_{\text{m}}$, for 3 phase stator winding voltage commutation with modulated amplitude.
- $tor\_sink[j]$ based on the filtered torque demand $ftorq\_dem$.
- $idemf()$ filters the i/p torque related current command signal $tor\_sink[j]$ with o/p $idemk[j]$.
- $cur\_cont()$ The lag compensator ‘optimizes’ the current error as the difference between the current command $idemk[j]$ and the filtered stator winding current feedback $fifbk[j]$ in each phase of the $3\Phi$ current control loop. The o/p $vmpwmk[j]$ from each of the high gain controllers is amplitude limited to the saturation voltage levels $\pm V_z (~10v)$ by zener diodes.
- $pwm\_mod()$ produces a width modulated o/p pulse sequence $pwmtrk[j]$ for each phase in accordance with the amplitude comparison of the modulating control signal $vmpwmk[j]$ o/p and the triangular dither signal $osc$. The modulator has a gain $K_{\text{mod}}$ with complementary outputs, for basedrive operation, that are hard limited to $\pm V_z (~10v)$ by zener diodes.
A test is used to interrogate the output status of the simulated comparator modulator by monitoring any observational sign change in the output polarity ($\pm V_z$), which is indicative of a modulated pulse edge transition, after execution of the software code module $\text{pwm\_mod}(\ )$ as shown in Figure 4-5.

Fig. 4-5. Simulation of PWM

```
int j; 
\text{osc}=\text{dither}(); \text{// Oscillator amplitude}
for(\text{j}=1;\text{j}<3;\text{j}++) 
{ 
\text{modop} = \text{pwm\_mod\_sign*Kmod*(vmpwmk[j]- osc)};
\text{if\( \text{modop} \geq V_z+V_d \) modop=V_z+V_d; else 
\text{if\( \text{modop} \leq -(V_z+V_d) \) modop = -(V_z+V_d); 
P\text{PWM\_trim}[j]=\text{PWM\_trim}[j]; \text{PWM\_trim}[j]=\text{modop}; }
```

```
\text{PWM\_carrier\_waveform\_generation}
\text{double} \text{tdel, delta}_t; 
\text{long stepk;}
\text{double dither(\text{void}) \{ }
\text{double} \text{pslp, period, temp; period}=1/\text{Fd;}
\text{pslp}=4/\text{Ad*Fd}; \text{// slope of positive going ramp}
\text{temp}=\text{fmod(fabs(stepk*delta}_t+\text{tdel}),period); if(temp<period/2)
\text{return (pslp*temp-Ad); \text{// Pos. going ramp. }
\text{else return (Ad-(temp-period/2)*pslp); }
```

www.intechopen.com
The o/p status of the comparator modulator is examined by comparing the trapped value \( cpwmtkr[j][j] \) at the beginning of the time step \( t_{k-1} \) with the new o/p \( pwmtrk[j][j] \) at \( t_k \) for each phase \( j \) and signalling any change via the \( pwmtestflag \) in the function call \( test_pwm_xover() \) detailed in Figure 4-6. If a crossover event occurs during simulation then the transition interval \( \Delta t_x = (t_x-t_{k-1}) \), denoted by \( \text{min-time} \), is determined by the regula falsi method in (LXVII) in the previous chapter.

```c
void test_pwm_xover(int *flag) {
    int i, j, ref_sign, act_sign, sigfl;
    double min_time, tol;

    *flag=0; // Reset pwm_sw_flag
    tol=0.001; // Tolerance limit on the time resolution

    for(j=1;j<=3;j++) { // Examine all 3 \( \Phi \) for PWM X-over
        sigfl=0;
        if(cpwmtkr[j]<0.0) ref_sign = -1; else ref_sign=1;
        if(pwmtrk[j]<0.0) act_sign = -1; else act_sign=1;
        if(ref_sign!=act_sign) { // PWM Crossover Check
            min_time=delt*(cosc-cvmpwmk[j])/(vmpwmk[j] -cvmpwmk[j]-osc+cosc); // Expression (3.85)
            if(min_time<0.0) nrerror("PWM Switch_time calculation error");
            if(min_time<=(1-tol)*delt) { // switch-time \( \leq \Delta t = T \)
                if(*flag>=1) {
                    for(i=1;i<=(*flag);i++)
                        if(min_time>=sw_time[i]-tol*delt &&
                            min_time<=sw_time[i]+tol*delt) sigfl=1;
                }
                if(min_time<0.0) nrerror("PWM Switch_time calculation error");
            }
            else { ++(*flag); phase_flag[*flag]=j; sw_time[*flag]=min_time; }
        }
        else { ++(*flag); phase_flag[*flag]=j; sw_time[*flag]=min_time; }
    }
    return;
}
```

![Fig. 4-6. Search for PWM X-over](www.intechopen.com)
A check is made to see if this value occurs within the imposed tolerance limit ($tol \cdot delt$) at the end of the time step interval $\Delta t_k$, denoted by $delt$, in which it is discarded in the affirmative without test flag registration. If multiple crossover events occur within the simulation interval, corresponding to different phases, then all transition times with the appropriate phase tag number are logged in the respective $sw\_time[j]$ and $test\_flag[j]$ arrays along with the signaled transition count via the PWM test flag. A check is also performed for identical multiple switch transition times without an increase in test flag count. The test routine is completed by arranging the multiple switch times, with corresponding phase listing, in increasing order of magnitude for subsequent detailed PWM simulation in the function call $\text{interPWMsimulation()}$. Accurate internal simulation of a modulated pulse transition, indigenous to the time step, commences with restoration of the captured global variables preceding the time step and temporary storage of the original step size ($zeit$) and time delay ($t_{del}$) settings, relevant to the $\text{dither()}$ signal source, for later retrieval. The new time step ($delt$) is initially set to the smallest switching interval $\Delta t_X$ ($sw\_time[1]$) for discretization of all first order linear subsystems using the call function $\text{setupfofilt()}$ as per the C-code module in Figure 4-7.

```c
void interPWMsimulation(int *flag) {
    int i, ref_sign, act_sign;
    double zeit, t_del, tol=0.001;
    zeit=delt; t_del=tdel; // Retain original time step info.
    restore_filt_out();
    for(i=1;i<=(*flag);i++) {
        delt=sw_time[i]; // Adjust delt=(t_X-t_k) to X-over time t_X
        if(i>1) delt -= sw_time[i-1];
        tdel=t_del-zeit+sw_time[i];
        setup_fo_filt();
        run_to_pwmsw();
        if(pwmtrk[phase_flag[i]]<0.0) act_sign = -1;
        else act_sign=1;
        if(cpwmtrk[phase_flag[i]]<0.0) ref_sign = -1;
        else ref_sign=1;
        if(ref_sign==act_sign) pwmtrk[phase_flag[i]] *= -1.0;
        // Force PWM X-over
        run_post_pwmsw();
    } // adjust $\Delta t$ to complete interval ($t_k-t_X$)
    delt=zeit-sw_time[*flag];
    tdel=t_del; // Restore original timing to $V_{in}(t)$
    setup_fo_filt();
    run_to_pwmsw(); run_post_pwmsw();
    delt=zeit; // Restore original time step
    setup_fo_filt();
    return;
}
```

Fig. 4-7. PWM X-Over Simulation

The necessary delay offset $t_d$ ($tdel$) is determined by back tracking ($\Delta t_k-\Delta t_X$) from $t_k$ for proper time registration in the execution of the carrier function $\text{dither()}$ and rerun of the call sequence $\text{run_to_pwmsw()}$ followed by the block function call $\text{run_post_pwmsw()}$. The post PWM
Simulation call list contains the additional BLMD model basedrive switching features as an embedded layer in the nested base_drive() and associated switch event signalling test_drk_xover() program routines. The complete BLMD model program is subsequently exercised for other multiple switch time intervals $\Delta t^{n+1}$ with updated linear system discretization and adjusted delay offset. Termination of the remainder of the original time step simulation is accomplished by setting the integration interval delt equal to the time step residue $(\Delta t_{k}-\Delta t_{max})$ followed by the call sequences run_to_pwmsw() and run_post_pwmsw() and exiting to the main program with a reinstatement of original time settings.

Numerical BLMD model simulation proceeds to the next program step in the flowchart cycle shown in Figure 3, by processing the call sequence run_post_pwmsw(), with the execution of the switch event routine base_drive() associated with the basedrive turn-on/off as a consequence of the PWM process. The relevant global variables and arrays associated with this call sequence run are trapped by the command capture-drk_out() as a precursor to basedrive simulation. The ’lockout’ circuit routine, illustrated in Figure 4-8, consists of the integrating capacitor action when the PWM comparator o/p $v_{ksm}[j]$ exceeds $v_{k-1,l}[j]$ and charge dumping when $v_{ksm}[j] < v_{k-1,l}[j]$ as shown in Figure 3 for the basedrive BDJ with a similar microprogram description for complementary basedrive BDJ operation. The exponential trigger voltage growth on the timing capacitor, along with the basedrive voltage threshold $V_{th} (=0.0)$ setting determines the inverter turn-on time. The charge dump action by the shunt diode across the delay timing resistor is virtually instantaneous when the switched comparator PWM output $v_{ksm}[j]=K_{mod} \cdot (v_{k}[j]-v_{ktr})$ is hard limited to $-v_{z}$.

```c
void capture_drk_out(void)
{
    int j;
    for(j=1;j<=3;j++) {
        cbase_drk[j]=base_drk[j];
        cbar_base_drk[j]=bar_base_drk[j];
        cback_emfk[j]=back_emfk[j];
        cph_curk[j]=ph_curk[j];
        cph_voltk[j]=ph_voltk[j];
        cifbk[j]=ifbk[j];
        cfifbk[j]=fifbk[j];
    }
    return;
}
```

```c
void base_drive(void)
{
    int j;
    double barpwmtrk,barpwmtrkm1;
    for(j=1;j<=3;j++) {
        barpwmtrk = -pwmtrk[j]; barpwmtrkm1 = -pwmtrkm1[j];
        base_drkm1[j]=base_drk[j];
        bar_base_drkm1[j]=bar_base_drk[j];
        if(DEL==1) { // Basedrive BDJ delay activated
            if(pwmtrk[j]>base_drkm1[j])//BDJ capacitor charge-up
                base_drk[j]=(dtrdfilt.n0*pwmtrk[j]+dtrdfilt.n1*pwmtrkm1[j])*dtrdfilt.k/dtrdfilt.d0-
                dtrdfilt.d1*base_drkm1[j]/dtrdfilt.d0;
            else if(base_drkm1[j] >= (Vd+pwmtrk[j]))
                base_drk[j]=pwmtrk[j]+Vd; //BDJ capacitor discharge
        }
        else if(DEL==0) { // BDJ delay switched OFF
            base_drk[j]=pwmtrk[j];
        }
        else if(DEL==2) { // BDJ delay OFF
            base_drk[j]=pwmtrk[j];
            bar_base_drk[j]=barpwmtrk;
        }
    }
}
```

Fig. 4-8. Basedrive simulation
This effect results in swift basedrive turn-off with zero delay when referenced to the trailing edge of the PWM o/p. However, the capacitor discharge can be gradual, when the PWM o/p is soft switched \((v_s > v^k_{sm}\text{[j]} > -v_s)\), due to the limited magnitude of the product combination of modulator gain \(K_{oc} \approx -68\) and error response \(v^k_c\text{[j]}\) of the current loop controller which is implicitly dependent on the filtered current feedback response \(\ddot{\text{i}}_k\text{[j]}\) for fixed current demand \(\bar{\text{i}}_k\text{[j]}\). The gradual reduction in capacitor voltage protracts the basedrive switch-off time, when referenced to the initial point of the logic “1-to-0” transition, associated with the PWM trailing edge. This delay has to be accounted for in an accurate inverter software model description with a search of the basedrive turn-off in addition to the turn-on times associated with exponential voltage growth.

The BLMD program test function \(\text{test_drk_xover (\&\text{drk_test_flag})}\), which is shown in Figure 4-10 and is very similar to \(\text{test_pwm_xover ()}\) in code content, checks for basedrive on/off firing signal occurrence within a simulation time step interval. This search is complemented with the evaluation of associated multiple phase activation times \(\Delta t_i\) for both normal BDJ and complementary BDJ inverter drive modes of operation. These inverter trigger instants \(t_s\) are determined by piecewise linear approximation using (LXXVI) in the previous chapter, ranked in increasing order of magnitude and phase tagged via the global storage arrays \(\text{drk_sw_time}\text{[j]}\) and \(\text{drk_phase_flag}\text{[j]}\) for subsequent use in detailed basedrive simulation. Accurate simulation of the basedrive trigger timing signals for subsequent inverter operation is achieved using the software routine call \(\text{inter_base_drk_loop_sim (\&\text{drk_test_flag})}\) which is shown in Figure 4-11 and has similar execution features to \(\text{inter_pwm_simulation ()}\).

The function call begins with the reinstatement of the global arrays at the beginning \((t_{k,1})\) of the time step using \(\text{restore_drk_out ()}\), illustrated in Figure 4-9, and temporary storage \((\text{zeit})\) of the original step size \(\Delta t\). The routine proceeds with linear system discretization appropriate to and with execution of the \(\text{base_drive ()}\) function and the subsequent call sequence \(\text{run_post_drksw ()}\), listed in Figure 4-12, for progressive substitution of multiple differential switch times as the temporary variable \(\text{delt}\).

This simulation call is completed with restoration of the original time step size followed by first order system discretization with a return to the main BLMD program to begin the new time step \(t_k \rightarrow t_{k+1}\). The function call group \(\text{run_post_drksw ()}\), summoned during main program execution in the flowchart of Figure 3, processes the following sequence of modular software activities illustrated in Figures 4-12 and 4-13 pertaining to BLMD system electrodynamic operation with inverter interaction.

\(\text{pwm_inv ()}\) generates the 3Φ inverter output HT binary voltage \(\text{ph_htk}\text{[j]} = v^k_g\text{[j]}\) in response to the PWM basedrive gating signals \(v_{bj} \& \bar{v}_{bj}\) shown in Figure 1. The magnitude of the simulated complementary trigger signals \(v^k_i \& \bar{v}^k_i\) in relation to the basedrive BDJ threshold voltage \(V_{th}\) establish the conduction states \(S_j(k)\) for \(k \in \{0,1,2\}\) as per (LV) in the previous chapter, by means of the tristate switching indicator \(V_{Os}\) of the complementary power transistor pair \(T_j\) and \(T_j\) in each leg \(J\) of the 3Φ inverter shown in Figure 15 in the previous chapter. The tristate flag condition in conjunction with the sustained stator winding current flow \(\text{ph_curk}\text{[j]} = \ddot{\text{i}}_k\text{[j]}\) through the free wheeling shunt diodes establish the inverter o/p binary voltage as 0 or \(V_{dc}\). The neutral star point voltage \(V_{ng}\) of the stator winding is determined from (LVIII) in the previous chapter for subsequent evaluation of the phase voltages \(\text{ph_voltk}\text{[j]} = v^k_s\text{[j]}\) via (LIX) in the previous chapter.
Fig. 4-10. Search for Basedrive X-Over

```c
void test_drk_xover(int *flag) {
    int i,j,ref_sign,act_sign,sigfl;
    double min_time,tol;
    *flag=0;
    tol=0.001; // Switching time resolution
    for(j=1;j<=3;j++) {
        if(base_drk[j]<0.0) ref_sign = -1; else ref_sign=1;
        if(base_drk[j]<0.0) act_sign = -1; else act_sign=1;
        // Check basedrive switching times
        if(ref_sign!=act_sign) {
            min_time=cbase_drk[j]*delt/(cbase_drk[j]-base_drk[j]);
            if(min_time<=(1-tol)*delta_t) { // switch time < \Delta_t
                if(*flag>=1) {
                    for(i=1;i<=(*flag);i++) {
                        if(min_time>=drk_sw_time[i]-tol*delta_t &&
                            min_time<=drk_sw_time[i]+tol*delta_t) sigfl=1;
                    } else { ++(*flag);
                        drk_phase_flag[*flag]=j;
                        drk_sw_time[*flag]=min_time;
                    }
                } else { ++(*flag);
                    drk_phase_flag[*flag]=j;
                    drk_sw_time[*flag]=min_time;
                }
            } else { // Check complementary switch times
                if(ref_sign==act_sign) {
                    min_time=cbbar_base_drk[j]*delt/(cbbar_base_drk[j]-bar_base_drk[j]);
                    if(min_time<=(1-tol)*delta_t) { // switch time < \Delta_t
                        if(*flag>=1) {
                            for(i=1;i<=(*flag);i++) {
                                if(min_time>=drk_sw_time[i]-tol*delta_t &&
                                    min_time<=drk_sw_time[i]+tol*delta_t) sigfl=1;
                            } else { ++(*flag);
                                drk_phase_flag[*flag]=j;
                                drk_sw_time[*flag]=min_time;
                            }
                        } else { ++(*flag);
                            drk_phase_flag[*flag]=j;
                            drk_sw_time[*flag]=min_time;
                        }
                    }
                } else { ++(*flag);
                    drk_phase_flag[*flag]=j;
                    drk_sw_time[*flag]=min_time;
                }
            }
        }
        // Adjust switching times in increasing order
        for(i=1;i<=(*flag);i++) { min_time=1.0;
            for(j=i;j<=(*flag);j++) {
                if(drk_sw_time[j]<min_time) {
                    min_time=drk_sw_time[j]; ref_sign=j;
                }
            }
            if(i!=ref_sign) {
                SWAP(drk_sw_time[i],drk_sw_time[ref_sign],min_time);
                SWAP(drk_phase_flag[i],drk_phase_flag[ref_sign],act_sign);
            }
        }
        // Contd
    }
    return;
}
```

Fig. 4-10
Extended Simulation of an Embedded Brushless Motor Drive (BLMD) System for Adjustable Speed Control Inclusive of a Novel Impedance Angle Compensation Technique...

Fig. 4-9. Restore basedrive variables

```c
// Restore basedrive variables
void restore_drk_out(void) {
    int j;
    for(j=1;j<=3;j++)
        base_drk[j]=cbase_drk[j];
    bar_base_drk[j]=cbar_base_drk[j];
    back_emfk[j]=cback_emfk[j];
    ph_voltk[j]=cph_voltk[j];
    ph_curk[j]=cph_curk[j];
    ifbk[j]=cifbk[j];
    fifbk[j]=cfifbk[j];
}
```

Fig. 4-10. Inter base drive loop simulation

```c
// Simulate basedrive with accurate switching times
void inter_base_drk_loop_sim(int *flag) {
    int i;
    double zeit;
    zeit=delt; // Retain original time step info.
    restore_drk_out(); // Recover global variables
    for(i=1;i<=(*flag);i++) {
        delt=drk_sw_time[i]; // simulate to switching instant t
        if(i>1) delt -= drk_sw_time[i-1];
        setup_fo_filt();
        base_drive();
        run_post_drksw();
    }
    delt=zeit-drk_sw_time[*flag];
    setup_fo_filt(); // Complete time step interval Sim.
    base_drive();
    run_post_drksw();
    delt=zeit; // Restore original time step interval ∆t
    setup_fo_filt();
    return;
}
```

Fig. 4-11. BDJ X-over Simulation

```c
// Inverter O/P and Motor Phase Voltage Generation
double vng;
int test_flag;
void pwm_inv(void) {
    int j,vo; // Vo = inverter Tristate switching indicator
    test_flag=0; // Phase winding count with zero current flow
    vng=0.0; // Stator winding neutral voltage w.r.t. ground

    // Inverter Conduction State
    for(j=1;j<=3;j++) { flag_cur[j]=0;
        ph_htkm1[j]=ph_htk[j];
        ph_voltkm1[j]=ph_voltk[j];
        if(base_drk[j]>Vth && bar_base_drk[j]<Vth) vo=1;
        else if(base_drk[j]<Vth && bar_base_drk[j]>Vth) vo = -1;
        else if(base_drk[j]<Vth && bar_base_drk[j]<Vth) vo=0;
        else nrerror("NO PWM Inverter Control");
        if((vo==1) || (vo==0 && ph_curk[j]<0.0)) ph_htk[j]=Vdc;
        else if((vo == -1) || (vo==0 && ph_curk[j]>0.0)) ph_htk[j]=0.0;
        else { test_flag += 1; flag_cur[j]=j;}
    }

    // Motor Winding Phase Voltage Generation
    if((test_flag == 3) || (test_flag == 2))
        for(j=1;j<=3;j++) ph_voltk[j]=back_emfk[j];
    if(test_flag == 0) {
        for(j=1;j<=3;j++) vng += ph_htk[j]/3;
        for(j=1;j<=3;j++) ph_voltk[j]=ph_htk[j]-vng;
    }
    else if(test_flag==1) {
        for(j=1;j<=3;j++) if((flag_cur[j]) test_flag=j;
        ph_htk[test_flag]=back_emfk[test_flag];
        for(j=1;j<=3;j++) vng += ph_htk[j]/2.0;
        ph_htk[test_flag] += vng;
        for(j=1;j<=3;j++) ph_voltk[j]=ph_htk[j]-vng;
    }
    test_flag=0;
    return;
}
```

Fig. 4-12. Post Basedrive Operation
**Electric Vehicles – Modelling and Simulations**

Fig. 4-13. Post Basedrive Call Sequence For Inverter Fed Motor Operation

```c
// Motor Winding Simulation
void winding(void)
{
    int i;
    double diffk,diffkm1;
    if(test_flag == 0) {
        for(i=1;i<=3;i++) {
            ifbkm1[i]=ifbk[i]; ph_curkm1[i]=ph_curk[i];
            diffk=ph_voltk[i]-back_emfk[i];
            diffkm1=ph_voltkm1[i]-back_emfkm1[i];
            ph_curk[i]=(dmotwind.n0*diffk+dmotwind.n1*diffkm1)*dmotwind.k/dmotwind.d0 -dmotwind.d1*ph_curkm1[i]/dmotwind.d0;
            ifbk[i]=Kf*ph_curk[i];
        }
    } else {
        for(i=1;i<=3;i++) {
            ifbkm1[i]=ifbk[i]; ph_curkm1[i]=ph_curk[i];
            ph_curk[i]=ifbk[i]=0.0;
        }
    }
    return;
}

// Electromagnetic Torque Generation
void convert_torqunit(void)
{
    int i;
    double sum=0.0;
    tt_km1=tot_torq;
    for(i=1;i<=3;i++)
        sum += ppsink[i]*ph_curk[i]*Kt;
    tot_torq = sum-torq_load;
    return;
}

// Motor Shaft Velocity Evaluation
double mech_power, elec_power;
void mot_shaft_velocity(void)
{
    int i;
    elec_power=0.0;
    msv_km1=mot_shaft_vel;
    mot_shaft_vel=(dmotvel.n0*tot_torq+dmotvel.n1*tt_km1)*dmotvel.k/dmotvel.d0 -dmotvel.d1*msv_km1/dmotvel.d0;
    for(i=1;i<=3;i++) {
        back_emfkm1[i]=back_emfk[i];
        back_emfk[i]=Kt*ppsink[i]*mot_shaft_vel;
        elec_power += back_emfkm1[i]*ph_curk[i];
    }
    mech_power=tot_torq*mot_shaft_vel;
    return;
}

// Function call determines Motor Shaft Position
double msv_km1, mspos_km1;
void mot_shaft_posn(void)
{
    mspos_km1=mot_shaft_pos;
    mot_shaft_pos =((dmot_posn.n0*mot_shaft_vel +dmot_posn.n1*msv_km1)*dmot_posn.k -dmot_posn.d1*mspos_km1/dmot_posn.d0);
    return;
}
```
If one or more phase currents are zero during the condition $V_O = 0$ a test flag $tf$ is incremented by unity for each null occurrence and the relevant phases are tagged using the $test\_cur[j]$ array for later identification in the computation of the relevant motor winding phase voltages. A test flag count of 2 or 3 indicates that the phase voltages are identical to the back EMF voltages $back\_emf[k][j] = v_{ke}[j]$ from the previous simulation step using expressions (XXIII) and (LVII) in the previous chapter. If the test flag ($u$) value is unity the associated phase current $ph\_cur[k][test\_flag]$ is zero and the following relations result

$$
\begin{align*}
v_{jg} = v_{js} + v_{sg} &= r_s i_{js} + L_s \frac{di_{js}}{dt} + v_{ej} + v_{sg}; \forall j \neq u \\
v_{us} &= v_{eu}; \therefore i_{us} = 0 \\
v_{sg} = \frac{1}{2} \left[ \sum_{j \neq u} v_{jg} + v_{us} \right]; \therefore \sum_{j \neq u} i_{js} = 0 & \& \sum_{j} v_{ej} = 0
\end{align*}
$$

(l)

which enable the phase voltages to be calculated from the 3Φ inverter o/p.

$winding()$ determines the stator winding current $ph\_cur[k][j]$ from a knowledge of the phase voltages and motor parameters {$r_s, L_s$} encoded in the discrete filter representation of the winding electrical behaviour.

$convert\_torqunit()$ computes the developed electromagnetic torque, manifested in the winding current flow, via the 3Φ commutation vector $psink[k]$ and motor torque constant $K_t$ for dynamic operation.

$mot\_shaft\_vel()$ evaluates the rotor shaft velocity $mot\_shaft\_vel = \omega_m$ from the net torque $tot\_torq = \Gamma_{ke}$ available, with the effects of load torque retardation $torq\_load = \Gamma_{kl}$ considered, using a discrete filter representation of the rotor dynamics as per (IL) in the previous chapter with parameters {$J_m, J_l, B_m$}. The back emf can be determined via the motor voltage constant $K_e$, along with the mechanical ($mech\_power$) and electrical ($elec\_power$) power delivery, once the shaft velocity is known.

$mot\_shaft\_pos()$ evaluates the motor shaft position $mot\_shaft\_pos$ from the rotor velocity.

3. BLMD model simulation with restraining shaft load torque

The effect of a fixed applied shaft load $\Gamma_l$ on BLMD model behaviour can be monitored via its simulation characteristics, in torque control mode with and without impedance angle compensation considered, for a range of torque demand step input stimuli capable of matching the posed restraining torque. A suitable choice for the target load magnitude is based on the manufacturers continuous rated stall torque of 5Nm for the particular motor type specified in Table I of the previous chapter. The set of motor shaft velocity step response characteristics \{\omega_m | 4v \leq \Gamma_d \leq 9v, \Delta \Gamma = 1v\} for the indicated torque demand $\Gamma_d$ range is displayed in Figures 6 and 7 without impedance angle compensation.

The response time $T_{res}$ required for the motor drive to reach the maximum shaft velocity of $\omega_{rmax} = 419$ rads.sec$^{-1}$ in each case, corresponding to the rated output of 4000 rpm in Table I of the previous chapter, improves with increased motor shaft acceleration via the “dynamic factor” $[J_m \Gamma_l] / J_m$ as a consequence of its dependency on torque demand input. The corresponding generated torque step response family of characteristics
\[ \{ \Gamma_e \mid 4v \leq \Gamma_d \leq 9v \text{ } \& \text{ } \Delta \Gamma_d = 1v \} \] for the specified variation in torque demand inputs are shown in Figures 8 and 9.

![Fig. 6. Shaft Velocity Step Response](image1)

![Fig. 7. Shaft Velocity Step Response](image2)

![Fig. 8. Generated Torque Response](image3)

![Fig. 9. Generated Torque Response](image4)

All the torque response characteristics, with the exception of that at \( \Gamma_d = 4v \), exhibit overshoot before settling to the required value of \( \approx 5.3\text{Nm} \) to overhaul the fixed restraining load torque (5Nm) and frictional effects. The degree of overshoot increases in proportion to the torque demand i/p, as exhibited in Figure 10 for the average peak EM torque \( \Gamma_{ep} \).
responsible for overshoot $\Delta \Gamma_{ep}$ accompanied by a corresponding reduction in settling time as shown in Figure 11.

Fig. 10. Mutual Torque Characteristic

Fig. 11. Torque Settling Time

3.1 Theoretical consideration of motor accelerative dynamical performance
The reduction in settling time is paralleled by the shaft velocity response time improvement in reaching rated motor speed. It is evident from inspection of the velocity and torque simulation traces that a direct correlation exists between the EM torque settling time and motor shaft velocity response time as indicated in Table I.

| Torque Load $\Gamma_i = 5 \text{Nm}$ | “Inertial” Time Constant $\tau_m = J_m / B_m = 0.318$ | Tran. Gain (Fig. 10) $K_T = 1.28$ |
|----------------------------------|---------------------------------|---------------------------------|
| Torq-Dem $\Gamma_d$ (Volts) | Av. Peak EM Torq $\Gamma_{ep}$ (Nm) | Torque Overshoot $\Gamma_{ep} - \Gamma_i$ (Nm) | Torq. Settling Time $T_{setl}$ (sec) Figs. 8/9 | Shaft Velocity Rise Time $T_{r}$ (sec) Figs. 6/7 | Theoretical Rise Time $T_{r}$ (sec) Eqn. (IV) | Rise Time $T_{r}$ (sec) via Dyn-Fac. Eqn. (VI) |
| 5 | 6.2 | 1.2 | $\sim 0.13$ | $\sim 0.13$ | 0.131 | 0.107 |
| 6 | 7.45 | 2.45 | $\sim 0.06$ | $\sim 0.06$ | 0.057 | 0.0524 |
| 7 | 8.98 | 3.98 | $\sim 0.04$ | $\sim 0.037$ | 0.034 | 0.0323 |
| 8 | 10.29 | 5.29 | $\sim 0.03$ | $\sim 0.027$ | 0.025 | 0.0243 |
| 9 | 11.634 | 6.634 | $\sim 0.024$ | $\sim 0.022$ | 0.02 | 0.02 |

Table I. Correlation of EM Torque Settling Time with Shaft Velocity Response Time
The shaft velocity step response rise time, as defined in Figure 6, can be obtained directly from the solution of the transfer function (XCIX) from the previous chapter in the time
domain with a step input approximation for the average peak torque overshoot
\( \Delta \Gamma_{ep} = \Gamma_{ep} - \Gamma_r \) in Figure 8 as

\[
\omega_r(t) = \frac{\Delta \Gamma_{ep}}{B_m} \left[ 1 - e^{-t/\tau_m} \right] 
\]

(II)

with time constant

\[
\tau_m = \frac{J_m}{B_m} 
\]

(III)

The step response time, for the shaft velocity under load conditions to reach maximum speed \( \omega_{r,\text{max}} \), can be determined from (II) for different torque demand i/p and corresponding peak torque values as per the above Table I with

\[
T_r = \tau_m \ln \left[ \frac{\Delta \Gamma_{ep}}{\Delta \Gamma_{ep} - B_m \omega_{r,\text{max}}} \right] 
\]

(IV)

The estimated rise times are in excellent agreement with the approximate settling and response times obtained from the BLMD model simulation traces. An alternative crude estimate of the response time can be obtained from the motor “dynamic factor”

\[
\alpha = \frac{d\omega_r}{dt} = \frac{(\Gamma_{ep} - \Gamma_r)}{J_m} 
\]

(V)

for average peak torque endurance as the acceleration time

\[
T_\alpha = \frac{\omega_{r,\text{max}}}{\alpha} 
\]

(VI)

from standstill to maximum speed assuming a shaft velocity linear transient response which is valid for torque demand values in excess of 5 volts.
These response estimates, given in above Table I, are in good agreement with those already obtained except for that at $\Gamma_d = 5v$ where the rise time is longer with exponential speed ramp-up.

### 3.2 Torque demand BLMD model response - internal node simulation

The simulated back-EMF along with the stator impedance voltage drop are illustrated in Figures 12 and 13 for two relatively close values of torque demand i/p. In the former case the torque demand i/p of 4volts results in sufficient motor torque to meet the imposed shaft load constraint (5Nm) without reaching rated speed and saturation ($\pm 10v$) of the current compensator o/p trace shown in Figure 14. The corresponding reaction EMF exceeds the winding impedance voltage $V_Z$ and is almost in phase with the stator current, which is proportional to $V_Z$, at the particular low motor speed reached. The torque demand i/p of 5v in the latter case results in the onset of a clipped current controller o/p in Figure 15 due to saturation ($\pm 10$) at rated motor speed $\omega_{r_{\text{max}}}$.

The back-EMF generated at this speed greatly exceeds the winding impedance voltage, as in the former case, and leads the stator current necessary to surmount the torque load by the internal power factor (PF) angle ($\sim 27^\circ$) with a correspondingly low power factor ($\sim 0.7$). The stator winding currents corresponding to the inputs $\{v_9, v_5\} \in \Gamma_d$ are displayed in Figures 16 and 17 respectively which indicate the marked presence of peak clipping in the latter case with loss of spectral purity due to heavy saturation of the current controller o/p for $\Gamma_d > 5v$.

The simulated motive power characteristic with the steady state threshold value of $\sim 2.3kW$ necessary to sustain shaft motion, for $\Gamma_d = 5v$ with restraining load torque and friction losses is shown in Figure 18 at base speed $\omega_{r_{\text{max}}} \equiv 420 \text{ rad.sec}^{-1}$.

Fig. 14. Current Compensator o/p

Fig. 15. Current Compensator o/p

The back-EMF generated at this speed greatly exceeds the winding impedance voltage, as in the former case, and leads the stator current necessary to surmount the torque load by the internal power factor (PF) angle ($\sim 27^\circ$) with a correspondingly low power factor ($\sim 0.7$). The stator winding currents corresponding to the inputs $\{v_9, v_5\} \in \Gamma_d$ are displayed in Figures 16 and 17 respectively which indicate the marked presence of peak clipping in the latter case with loss of spectral purity due to heavy saturation of the current controller o/p for $\Gamma_d > 5v$.

The simulated motive power characteristic with the steady state threshold value of $\sim 2.3kW$ necessary to sustain shaft motion, for $\Gamma_d = 5v$ with restraining load torque and friction losses is shown in Figure 18 at base speed $\omega_{r_{\text{max}}} \equiv 420 \text{ rad.sec}^{-1}$.
This can be rationalized from the power budget required to sustain the load torque at rated speed via (LXXXVIII) in the previous chapter as

$$P_l = \Gamma_1 \omega r_{max} = (5)(420) = 2.1 \text{kW}$$

(VII)

The excess coupling field power required to surmount mechanical shaft friction losses is shown simulated in Figure 19 with a steady state estimate of ~200 watts.
Extended Simulation of an Embedded Brushless Motor Drive (BLMD) System for Adjustable Speed Control Inclusive of a Novel Impedance Angle Compensation Technique...

Stator Winding Phasor RMS Magnitude Estimation as per Figure 44 in previous chapter via BLMD Model Simulation

| Torq_Dem | Shaft_Vel | Elec_Power | Back_EMF | Imped_Vol | Ph_Cur |
|----------|-----------|------------|----------|-----------|--------|
| Volts    | \(\phi_{\text{max}}\) | Volts (XLVII) | \(V_e\) | \(V_Z\) | \(I_{\phi}\) |
| 5v       | 419.2     | 2301       | 94.82    | 44.87     | 9.09   |
| 6v       | 420.3     | 2305       | 95.24    | 48        | 9.7    |
| 7v       | 418.9     | 2298       | 94.78    | 50.89     | 10.32  |
| 8v       | 410.3     | 2251       | 92.05    | 52.42     | 10.84  |
| 9v       | 405.5     | 2224       | 91.5     | 53.66     | 11.23  |

Table II. Phase Angle Evaluation for BLMD Steady State Operation with \(\Gamma_d = 5\text{Nm}\)

The effect of shaft load on the BLMD model simulation characteristics for \(\Gamma_d > 5\text{v}\) is summarized in above Table II for steady state conditions with the aid of the general phasor diagram in Figure 42 of the previous chapter.

Fig. 20. Motor RMS Phasor Voltages  
Fig. 21. Stator Winding Phase Angles

www.intechopen.com
It is evident from the table that the back EMF has reached its peak rms value with the onset of maximum shaft velocity, for all values of \( \Gamma_d > 5V \), with

\[
V_{ej_{\text{max}}} = \frac{K_e}{\sqrt{2}} \omega_{r_{\text{max}}} = \left( \frac{0.315}{\sqrt{2}} \right) (420) = 93.6V
\]  

(VIII)

Furthermore the impedance voltage drop \( V_z \) in (XC) of the previous chapter is limited to a very small increase with torque demand current \( I_{dj} \) listed in Table II and is shown almost stabilized to a constant value in Figure 20. This voltage clamping effect, due to current compensator o/p saturation in response to tracking current feedback, is controlled to achieve the desired rms level of clipped current flow in the stator winding as shown in Figure 17 to satisfy torque load requirements. The rms winding current flow necessary at unity internal power factor to meet steady state torque load and friction demands at \( \sim 5.4 \) Nm in Figures 8 and 9 can be determined from (XLV) in the previous chapter as

\[
\bar{I}_{js} = \left( \frac{\sqrt{2}}{3} \right) \left( \frac{\Gamma_e}{K_e} \right) = \left( \frac{\sqrt{2}}{3} \right) \left( \frac{5.42}{0.315} \right) = 8.11 \text{ Amps}
\]  

(IX)

This is almost identical to the rms values obtained from BLMD model simulations in Table II, which are consistent with increased torque current demand, when internal power factor self adjustments are accounted for as in

\[
\bar{I}_{js} = I_{js} \cos \varphi_I \approx 8.1 \text{ Amps}
\]  

(X)

The internal power factor angles, listed in Table II and displayed in Figure 21, are deduced for \( \Gamma_d > 5v \) from the mechanical power transfer by substituting the rms quantities obtained from back EMF and winding current simulations in expression (XCII) of the previous chapter. These angles, which increase with torque demand i/p, can be alternatively calculated from the simulated winding current response using (X) with knowledge of \( \bar{I}_{js} \).

The tabulated angle estimates obtained statistically as the phase lag between the current and back EMF waveforms in Figure 13, for example, are in close agreement with those from (XCII) of the previous chapter. The motor winding impedance angle \( \phi_z \), which is fixed at rated machine speed \( \omega_{r_{\text{max}}} \), is determined from (LXXXIV) as \( \sim 81.2^\circ \) in Table II.

The rms winding voltage \( V_{js} \) is obtained in its pure spectral form, instead of the PWM version furnished by the current controlled inverter, upon application of (XCIII) to the known rms phasor quantities given in Table II for different values of \( \Gamma_d > 5V \).

Knowledge of the relevant phasor magnitudes with corresponding phase angles enable the load angle \( \beta_T \) to be determined from (XCV) of the previous chapter for given shaft load conditions. This is approximately fixed, at \( \sim 15^\circ \) as indicated in Table II with about \( 2^\circ \) variation, over the torque demand i/p range as shown in Figure 21. The resulting power factor angle \( \varphi \) listed in Table II increases with \( \varphi_I \), for fixed load angle over the torque demand i/p range as shown, in a way that is commensurate in (X) with motor current requirements towards sustaining shaft load torque with a decreasing power factor as illustrated in Figure 22.
3.3 BLMD model simulation with novel impedance angle compensation

The effect of motor impedance angle compensation (MIAC), manifested as commutation phase lead angle incorporated into the BLMD model in (XCVIII) of the last chapter as $p\theta_e - 2(j-1)\frac{\pi}{3} \rightarrow (p\theta_e + \varphi_e) - 2(j-1)\frac{\pi}{3}$ on the motor step response velocity and torque characteristics is illustrated in Figures 23 and 24 for the torque command i/p range $4\leq \Gamma_d \leq 9$ at step size intervals of $\Delta\Gamma_d = 1$ volt.

Fig. 23. Shaft Velocity with MIAC

Fig. 24. Torque Response with MIAC
The variation of peak torque overshoot with i/p demand, displayed as the mutual characteristic in Figure 25, is linear with a transfer gain that is lower than that without MIAC in Figure 10. Consequently the maximum peak torque delivery, for a given i/p demand to sustain shaft load requirements, is lower in amplitude and of shorter overshoot pulse duration as seen in Figure 24 when compared with that without MIAC in Figures 8 and 9. Furthermore the persistence of torque overshoot is lower with a much reduced settling time (<0.015 sec), in reaching steady state sustained load conditions in all cases albeit at lower acceleration and much smaller drive speeds, thereby exerting less mechanical stress on the drive shaft components and minimizing shaft flexure in EV propulsion applications.

![Simulated BLMD Mutual Torque Characteristic with Impedance Angle compensation](image1)

**Fig. 25. Mutual Torque with MIAC**

![Simulated Velocity - Torque Dependency with Impedance Angle Compensation](image2)

**Fig. 26. Torque - Velocity Transfer Curve**

![Impedance Angle Compensation with Commutation Phase Angle \( \phi_{\text{Advance}} \)](image3)

**Fig. 27. Impedance Angle Compensation**

![Motor Step Response with Impedance Angle Compensation](image4)

**Fig. 28. Phasor Voltages with MIAC**
The shaft velocity characteristics also indicate a much lower steady state motor run speed, with MIAC deployed, which never reaches velocity saturation $\omega_{r_{\text{max}}} = 419 \text{ rad.sec}^{-1}$ over the permissible torque demand i/p range of $-10V \leq \Gamma_d \leq 10V$. The relevant command torque to shaft velocity transfer characteristic is approximately linear as shown in Figure 26 which indicates a maximum motor operating speed of $\omega_{r_{\text{max}}} \approx 120 \text{ rad.sec}^{-1}$ with $\omega_{r_{\text{max}}} \leq 30\% \omega_{r_{\text{max}}}^*$ under rated load conditions (5Nm) for a maximum demand i/p of $\Gamma_{d_{\text{max}}} = 10V$. This speed reduction is singly due to the maintenance of an almost zero load angle $\beta_T$ shown in Figure 27, between the motor terminal $V_js$ and back EMF $V_{ej}$ rms voltage phasors in Figure 45 of the previous chapter, by commutation phase angle advance for optimal torque production as indicated from the BLMD simulation results in Table III. This phase compensation technique results in back EMF and winding impedance voltage $V_z$ phasors that appear approximately equal in magnitude over the allowable torque demand input range as shown in Figure 28. Furthermore the internal power factor angle $\phi_I$ is forced to adopt approximately the same value as the machine impedance angle $\phi_z$ as indicated in Table III, by the phase advance measure $\phi_\circ$ in the current commutation circuit, with a consequent collinear alignment of phasors $V_{ej}$ and $V_z$ in Figure 45. This collinear arrangement can only be sustained at a particular machine speed that is dependent on the torque demand i/p which determines the subsequent winding current flow and thus the necessary impedance angle for alignment. This reasoning can be deduced as follows by noting that for a given torque load $\Gamma_l$ the rms winding current flow is linear with torque demand i/p as per Table III and Figure 29.

### Stator Winding Phasor RMS Magnitude Estimation as per Figure 45 of the Previous Chapter via BLMD Model Simulation

| Torq_Dem $\Gamma_d$ Step i/p | Shaft_Vel $\omega_{\text{r_{max}}}$ rad.sec$^{-1}$ | Elec_Power $P_e$ (XLVII) in Prev. Chap. | Back_EMF $V_{ej}$ volts | Imped_Vol $V_z$ volts - (XC) in Prev. Chap. | Ph_Cur $I_{js}$ amps |
|-----------------------------|---------------------------------|--------------------------------------|-------------------------|--------------------------------------|--------------------------|
| 4v                          | 18.6                            | 94.44                                | 4.17                    | 6.06                                 | 7.76                     |
| 5v                          | 48.95                           | 257.2                                | 11.5                    | 9.23                                 | 9.7                      |
| 6v                          | 70.87                           | 363.67                               | 16.01                   | 13.05                                | 11.71                    |
| 7v                          | 87.9                            | 452.6                                | 19.95                   | 17.33                                | 13.66                    |
| 8v                          | 102.9                           | 531.2                                | 23.28                   | 22.18                                | 15.7                     |
| 9v                          | 116.3                           | 602.2                                | 26.3                    | 27.45                                | 17.74                    |

### Derived Phase Quantities as per Figure 42 of the Previous Chapter

| Torq_Dem $\Gamma_d$ volts | Int. PF Ang $\phi_I$ (XCII) in Prev. Chap. | Ph_Vol $V_{js}$ (XCIII) in Prev. Chap. | Imp_Ang $\phi_z$ (LXXXIV) in Prev. Chap. | Load_Ang $\beta_T$ (XCV) in Prev. Chap. | PFAng $\phi_i + \beta_T$ |
|---------------------------|---------------------------------------------|----------------------------------------|------------------------------------------|-----------------------------------------|--------------------------|
| 4                         | 13.75$^\circ$                              | 10.23                                  | 16.1$^\circ$                            | 1.39$^\circ$                            | 15.14$^\circ$            |
| 5                         | 36.13$^\circ$                              | 20.73                                  | 37.22$^\circ$                           | 0.51$^\circ$                            | 36.64$^\circ$            |
| 6                         | 49.71$^\circ$                              | 29.06                                  | 47.72$^\circ$                           | -1.0$^\circ$                            | 48.71$^\circ$            |
| 7                         | 56.38$^\circ$                              | 37.27                                  | 53.76$^\circ$                           | -1.22$^\circ$                           | 55.16$^\circ$            |
| 8                         | 61.02$^\circ$                              | 45.44                                  | 57.95$^\circ$                           | -1.5$^\circ$                            | 59.52$^\circ$            |
| 9                         | 64.52$^\circ$                              | 53.73                                  | 61.01$^\circ$                           | -1.79$^\circ$                           | 62.73$^\circ$            |

Table III. Phase Angle Evaluation at $\Gamma_l = 5Nm$ with Motor Impedance Angle Compensation
3.3.1 MIAC substantiation via theoretical analysis and validation

The internal power factor angle $\varphi_I$ can be determined theoretically for fixed winding current flow corresponding to a given torque demand i/p using (IX) and (X), assuming negligible dynamic friction at the shaft speeds concerned with $\Gamma_I \gg \Gamma_f$, as

$$\varphi_I = \cos^{-1}\left(\frac{\sqrt{2}}{3} \frac{\Gamma_I}{K_{\mu}}\right) \quad \text{(XI)}$$

The motor terminal voltage i/p $V_{js}$ in (XCIII) from previous chapter can be optimized with respect to the motor impedance angle $\varphi_z$, which is unknown, in terms of the rms phasor quantities $V_{ej}$, $V_z$ and the fixed internal power angle $\varphi_I$ from (XI) by letting

$$\frac{dV_{js}}{d\varphi_z} = 0 \Rightarrow \sin(\varphi_z - \varphi_I) = 0 \quad \text{(XII)}$$

This procedure results in the impedance angle $\varphi_z$ in terms of the known angle $\varphi_I$ as

$$\varphi_z = \varphi_I \quad \text{(XIII)}$$

with

$$\max V_{js} = \sqrt{V_{ej}^2 + V_z^2 + 2V_{ej}V_z} = (V_{ej} + V_z) \quad \text{(XIV)}$$

which is unknown as both $V_{ej}$ and $V_z$ depend on the motor shaft velocity $\omega_s$. The shaft velocity can now be determined from (LXXXIV) from previous chapter using expression (XIII) as

$$\omega_s = \left(\frac{P}{J_s}\right) \tan \varphi_I \quad \text{(XV)}$$
Theoretical Estimation of RMS Phasor Magnitudes

| Torq_Dem | Ph_Cur | Int_PF | Shaft_Vel | Back_EMF | Imp_Vol | Ph_Vol |
|----------|--------|--------|-----------|----------|---------|--------|
|           |        |        |           |          |         |        |
| $\Gamma_d$ | $I_{js}$ | $\varphi_i$ | $\omega_r$ | $V_{ej}$ | $V_{z}$ | $V_{js}$ |
| Step i/p  | Eqn. (XI) | Eqn. (XV) | Eqn. (VIII) | Eqn. (XC) | Eqn. (XIV) |
| 4v       | 7.76 A | 15.37° | 17.71 rad/s | 3.94v | 6.04v | 9.98v |
| 5v       | 9.70 A | 39.52° | 53.17 rad/s | 11.84v | 9.43v | 21.27v |
| 6v       | 11.71 A | 50.28° | 77.55 rad/s | 17.27v | 13.74v | 31.01v |
| 7v       | 13.66 A | 56.79° | 98.43 rad/s | 21.92v | 18.71v | 40.63v |
| 8v       | 15.70 A | 61.54° | 118.87 rad/s | 26.48v | 24.71v | 51.19v |
| 9v       | 17.74 A | 65.05° | 138.49 rad/s | 30.85v | 31.54v | 62.39v |

Table IV. Motor Impedance Angle Compensation

This value of $\omega_r$ can be used to theoretically generate the rms voltage phasors $V_{ej}$, $V_z$ and $V_{js}$ using expressions (VIII), (XC) and (XCVIII) in the previous chapter respectively from a knowledge of the motor winding phasor current $I_{js}$ as per Table IV over the i/p torque demand range $\Gamma_d \geq 4V$. The quantities obtained from BLMD simulations in Table III compare reasonably well with those derived in Table IV from theoretical considerations which reinforces model validation and confidence. The optimized internal power factor angle, which is almost identical to that in Table III, results in a zero load angle $\beta_T$ from (XCV) in the previous chapter due to the phasor collinearity and thus improved torque control via the PWM voltage supplied by the current controlled inverter. The power factor angle $\varphi$, internal power factor angle $\varphi_i$ and machine impedance angle $\varphi_z$ variations with torque demand i/p, which are displayed in Figure 27 using estimates extracted from BLMD model simulation in Table III for $\Gamma_d \geq 4V$, are almost congruent with a mismatched difference manifested as the negligible load angle ($\beta_T \approx 0$).

![Impedance Angle Compensation with Commutation Phase Angle $\varphi$ Advance](Fig. 30. Motor Power Factor)

![Motor Power Factor (PF) with Applied Shaft Load Torque](Fig. 31. Power Factor Comparison)
The internal power factor $\cos \varphi_I$ shows a gradual deterioration with increasing torque demand $i/p$ in Figure 30 as expected with the accompanying internal power factor angle $\varphi_I$ adjustment, from the mirrored motor current increase in Figure 29, constrained by a fixed shaft load in (X). Impedance angle compensation results in an improved motor power factor as shown in Figure 31 than that without MIAC over the torque demand $i/p$ range $4V \leq \Gamma_d \leq 6V$ necessary to meet load requirements $\Gamma_l$.

Motor speed reduction is also mirrored with a decrease of the shaft velocity step response rise time as shown Figure 32 with maximum values falling below the velocity time response floor of the uncompensated BLMD model. This results in constant motor speed operation, though small by comparison to that without phase angle advance, well below the rated value in torque control mode with smooth torque delivery to satisfy load requirements.

![Graph showing 10% to 90% Rise Time $t_r$ of Shaft Velocity Step Response with Motor Torque Loop Control](image)

**Fig. 32. Shaft Velocity Rise Times**

The simulated motor winding impedance and back EMF voltages for mid (5V) and full range (9V) torque demand input values, which result in developed torque capable of surmounting the fixed restraining shaft load (5Nm), are displayed in Figures 33 and 34. Both sets of characteristics exhibit comparable amplitudes appropriate to the level of torque demand $i/p$, with speed related motor current phase lags $\varphi_I$ as per Table III, that are much lower than those without MIAC in Figure 13. The impedance and back EMF voltages are interrelated which can be shown as follows by starting with expression (XC) for $V_z$ and using (IX) and (X) giving

$$V_z = |Z| |I| \cos \varphi = \left|Z\left(\frac{\alpha I}{V}ight)\right|\cos \varphi_I$$

(XVI)

This can be rewritten by using (LXXXIV) in the previous chapter with optimized value of $\varphi_I$ in (XIII) as
\[ V_z = \left( \frac{\sqrt{2}}{3} \middle( \frac{\Gamma_i}{K_r} \right) \left( r_s^2 + (p \omega_i L_s)^2 \right) \right) \]  
(XVII)

The shaft velocity \( \omega_r \), linking the back EMF, can be replaced in (XVII) by using (VIII) yielding

\[ V_z = K_1[1 + K_2 V_{ej}^2] \]  
(XVII)

where

\[ K_1 = r_s \left( \frac{\sqrt{2}}{3} \middle( \frac{\Gamma_i}{K_r} \right) \right) = 5.612 \]  
(XVIII)

and

\[ K_2 = \left( \frac{\sqrt{2} p L_s}{K_r r_s} \right)^2 = 4.855 \times 10^{-3} \]  
(XIX)

from substitution of parameters in Table I of the previous chapter and \( \Gamma_i = 5 \text{Nm} \). The impedance voltage in (XVII) is expressed as a quadratic equation in terms of the back EMF with points of equality corresponding to

\[ V_{ej} \in \{6.915 \text{v}, 29.79 \text{v}\} \left| V_z = V_{ej} \right\} \]  
(XX)

which are visible in Figure 28 as points of intersection of the two voltage traces.
These crossover points divide the rms $V_z$ amplitude variation along with $V_{ej}$ in Figure 28 into three distinct regions, over the usable torque demand i/p range as per Table IV, with

\[
V_z > V_{ej} \text{ for } V_{ej} < 6.9v \\
V_z < V_{ej} \text{ for } 6.9v < V_{ej} < 29.8v \\
V_z > V_{ej} \text{ for } V_{ej} > 29.8v
\]

\[(XXI)\]
These regions can also be inferred from the voltage amplitude traces in Figure 33 and 34 where $V_{ej}$ exceeds $V_z$ in the former case with $\Gamma_d = 5\nu$ and vice versa for $\Gamma_d = 9\nu$ in the latter diagram.

![Motor Step Response](image1)

*Motor Step Response*

**Impedance Angle Compensation Employed**

**Motor Shaft Load $\Gamma_L = 5\text{Nm}$**

**Torque Demand $\Gamma_d = 9\nu$**

**Time (sec)**

0 0.03 0.06

![Motor Step Response](image2)

*Motor Step Response*

**With Impedance Angle Compensation**

**Motor Shaft Load $\Gamma_L = 5\text{Nm}$**

**Torque Demand $\Gamma_d = 9\nu$**

**Time (sec)**

0 0.01 0.03 0.04 0.06

**Simulated Current Compensator o/p**

**Simulated Current Feedback $i_{fa}$**

Fig. 39. Motor Current Feedback

Fig. 40. Current Controller o/p

The simulated stator winding current along with current feedback response and current controller o/p are displayed in Figures 35, 36 and 37 respectively for $\Gamma_d = 5\nu$ without saturation related distortion in the current compensator o/p. The BLMD model simulation current characteristics corresponding $\Gamma_d = 9\nu$ are also shown in Figures 38, 39 and 40 without saturation effects.

4. BLMD reference model simulation in velocity control mode

In this section the BLMD reference model performance as an ASD emulator is examined and compared with experimental step response data for shaft inertial load conditions with $J_i \sim 3J_m$. Adjustable speed drive operation, with embedded inner PWM current control, is effected by closing the outer velocity loop via a two term PI term controller $G_v$ as shown in Figures 1 and 5. The analog velocity controller shown in Figure 41, which has an inbuilt velocity offset adjustment and speed gain control adjustment $K_e$ for the chosen BLMD system modelled here (Moog GmbH, 1989) has a transfer function

$$G_v(s) = K_e \left( K_p + \frac{K_i}{s} \right)$$  \hspace{1cm} (XXII)

with proportional and integral compensation gain settings $K_p$ and $K_i$ respectively. The inclusion of this outer loop velocity compensator, in addition to the inner torque control current loop, results in a complete holistic BLMD reference model that can now be used for ASD simulation and performance evaluation in embedded applications. Proportional and
integral control is easily incorporated in C-language routine during BLMD simulation of velocity closed loop operation as a digital filter code module via (LXV) of the previous chapter using the backward Euler method in (LXIV) of the previous chapter. The resulting ASD model was exercised at low and high shaft velocities corresponding to 36.4% and 73.6% of rated motor speed $n_o$ in Table 1 of the previous chapter and compared with experimental test data at critical internal nodes in Figure 1 for model validation and simulation accuracy.

![BLMD Velocity Controller Diagram](image)

**Fig. 41. BLMD Velocity Controller**

The current controller $G_j$ step response simulation traces for a velocity step command input $V_\omega$ of 2volts, corresponding to 36% of rated motor speed (~4000rpm), are exhibited in Figures 42 to 44 for linear pulsewidth modulator operation.

![BLMD Benchmark Model Reference Simulation for Closed Loop Velocity Control Operation](image)

**Fig. 42. ASD Reference Model Current Demand Comparison with Experimental Test Data**
Extended Simulation of an Embedded Brushless Motor Drive (BLMD) System for Adjustable Speed Control Inclusive of a Novel Impedance Angle Compensation Technique...

**Fig. 43. ASD Model Reference Current Feedback Comparison with Experimental Test Data**

**Fig. 44. ASD Model Current Controller o/p Comparison with Experimental Test Data**

The accuracy of these simulation traces, which capture the essence of the velocity transient step response $V_\omega$ overshoot in Figure 45, is characterized by a large waveform correlation coefficient of fit in Table V which provides a good indication of the model fidelity when matched with experimental data.
BLMD Benchmark Model Reference Simulation for Closed Loop Velocity Control Operation

Comparison of BLMD Shaft Velocity Step Response Simulation with Experimental Test Data for a 2 Volt Velocity Command Input

Medium Shaft Inertial Load Conditions: $J_{mml} = 12.3 \text{kg.cm}^2$
BLMD in Velocity Control mode with Step I/P Stimulus $V_{in} = 2$ volt
Motor Current Optimizer: MCO-402B

Data Acquisition:
Sample Size $N_d = 5000$; Sampling Rate $f_s = 12.5 \text{kHz}$

Fig. 45. ASD Model Reference Shaft Velocity Comparison with Experimental Test Data

| Waveform Correlation Analysis for Total Inertial Shaft Load $J_{Tot} = J_I + J_m = 12.3 \text{kg.cm}^2$ | ASD Waveform | Velocity Command i/p $V_{in} = 2V$ | Velocity Command i/p $V_{in} = 4V$
|------------------------------------------------|-----------------|-----------------|-----------------|
| Target Data Length $N_d = 3000$ | Data Sampling Rate: 12.5kHz | BLMD Simulation Time Step: 1μs | Current Demand $I_{dj}$ | 96.8% | 92.85% |
| Current Feedback (FC) $I_{fj}$ | 97.26% | 93.27% |
| Current Compensator output $V_{cj}$ | 59.81% | 45.46% |
| Motor Shaft Velocity output $V_{ar}$ | 99.8% | 99.68% |

Table V. ASD Model Trace Simulation Comparison with Experimental Test Data

The simulated current demand and feedback waveforms, which have high matching coefficients with test data, exhibit an amplitude modulated step response with velocity transient overshoot and ringing, before eventually setting to negligible constant amplitude traces with fixed frequency commensurate with reached shaft speed $\omega_r$ demanded ($V_{ar} \sim 2V$) in Figure 45.

The compensated velocity error output for 2Volts operation $V_{ar}$ in the BLMD network structure in Figure 5 is equivalent to the filtered torque demand $\Gamma_{df}$ as the velocity control effort $V_{voe}$ shown in Figure 46, applied to the inner closed loop for motor current control and BLMD output torque regulation. This optimized velocity error $V_{ar}$ in Figure 46 is a short duration pulse for reasons of fast BLMD shaft velocity risetime $T_{res}$ and short setting time $T_{setl}$ as required in high performance ASD industrial applications.
Fig. 46. ASD Reference Model Compensated Velocity Error Output

The presence of overshoot in the BLMD velocity step response in Figure 45 is due to the non-optimal tuning of the velocity controller PI parameters required to ensure stiff dynamical operation for the total drive shaft inertial load \( J_{\text{Tot}} = J_m + J_l \) in question.

Fig. 47. ASD Shaft Velocity Step Response Variation with Rotor Inertial Load \( J_{\text{Tot}} \)
Electric Vehicles – Modelling and Simulations

**Fig. 48. Variation of ASD Velocity Control Effort with Rotor Inertial Load $J_{Tot}$**

**Fig. 49. Variation of ASD Settling Time with Inertial Loading**
Examination of the ASD velocity step response trace simulations over a range of shaft inertial load multiples of the rotor value $J_m$ in Figure 47 reveal that the PI parameters have been optimized only at zero load with $J_{rot} = J_m$ for good drive dynamic transient performance with little overshoot. The effect of increased shaft inertia on the velocity control effort $V_{ue}$ in Figure 48, for a velocity command input of 2 volts, is a greater sustained oscillation accompanied by longer settling times $T_{set}$ manifested in the simulated ASD velocity step response as shown in Figure 49. This behaviour is mirrored by an increased overshoot, as defined in Figure 47, in the BLMD shaft velocity step response with shaft load inertia as shown in Figure 50.

The effect of increased load inertia on ASD dynamic performance also translates into slower rise times $T_{res}$ as shown in Figure 51 for a non optimally tuned velocity controller.

Further ASD step response simulation and comparison with experimental measurements in Figures 52 to 55, for a 4 volts velocity command input which corresponds to 74% of rated motor speed $n_0$, reveal good BLMD model accuracy.

Fig. 50. Variation of ASD Shaft Velocity Overshoot with Inertial Loading

---

**ASD Shaft Velocity Step Response Percentage Overshoot Variation with Rotor Shaft Inertial Loading for Different Command Velocities**

| Motor Shaft Load $J_l$ – Multiples of Rotor Inertia $J_m$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| ASD % Overshoot ($\Delta V_{\omega}/V_{\omega,r}$)100% | 10 | 18 | 26 | 34 | 42 | 50 |

- ASD Rotor Inertia: $J_m = 3.04$ kg.cm$^2$
- Motor Current Optimizer: *MCO – 402B*
- ASD Simulation Time Step Size: $\Delta t = 1\mu$s
- ASD Velocity Demand $V_\omega = 2$ Volts
- ASD Velocity Demand $V_\omega = 4$ Volts
Fig. 51. Variation of ASD Velocity Response Rise Time with Load Inertia

Fig. 52. BLMD Reference Model Current Demand Comparison with Experimental Test Data
**Extended Simulation of an Embedded Brushless Motor Drive (BLMD) System for Adjustable Speed Control Inclusive of a Novel Impedance Angle Compensation Technique**

Fig. 53. BLMD Model Current Feedback Comparison with Experimental Test Data

Fig. 54. BLMD Model Current Controller Output Comparison with Experimental Test Data
The quality of ASD simulation trace match with test data is indicated by the high value of the waveform correlation coefficients given in Table V.

Fig. 55. BLMD Model Shaft Velocity Comparison with Experimental Test Data

Fig. 56. BLMD Reference Model Compensated Velocity Error Output
The velocity loop derived torque command input stimulus $V_{\omega e}$ in Figure 56 has a pulse duration that is much shorter than the time constant ($\tau_{ml} \sim J_{tot}/B_m$) of the load dynamics, even though the pulse amplitude is of sufficient strength to force the shaft velocity to the value demanded ($V_{\omega e} \sim 4$ volts). The endurance $\tau$ of the velocity control effort in Figure 56, associated with pronounced PWM saturation in Figure 54, is a measure of the maximum sustained EM torque necessary to accelerate the total BLMD inertial load masses to the appropriate shaft velocity demanded by the ASD command setting $V_{\omega e}$. This velocity control pulse has amplitude that is clipped to a maximum saturation limit of 10 volts at the three phase current generator input, which limits the size of the torque loop input stimulus, in the derivation of the BLMD current command signals.

Examination of the family of characteristics pertaining to velocity control effort over a range of motor shaft inertial loads in Figure 57 indicate peak saturation over long pulse intervals $\tau^*$ proportional to the inertial masses as in Figure 58 to be accelerated to the required speed $V_{\omega r}$. This velocity error saturation is absent in the characteristics displayed in Figure 48 for 2 volt ASD operation and results in linear PWM operation with a BLMD acceleration torque delivery commensurate with the velocity effort.

---

**Fig. 57. Variation of Velocity Control Effort with Motor Shaft Inertial Load $J_{Tot}$**

- **BLMD Benchmark Reference Model Simulation for Closed Loop Velocity Control Operation**
- **ASD Velocity Controller Error Output Simulation for a 4 Volt Velocity Command Step Input at Various Shaft Inertial Loads $J_{Tot}$**

Medium Shaft Inertial Load Conditions: $J_{mmld} = 12.3 \text{kg.cm}$
- ASD Rotor Inertia: $J_m = 3.04 \text{kg.cm}^2$
- Motor Current Optimizer: MCO – 402B
- Simulation Time Step Size: $\Delta t = 1 \mu s$

**BLMD Reference Model Shaft Velocity Step Response for Various Inertial Loads $J_{Tot}$**

- $J_{Tot} = 8 J_m$
- $J_{Tot} = 7 J_m$
- $J_{Tot} = 6 J_m$
- $J_{Tot} = 5 J_m$
- $J_{Tot} = 4 J_m$
- $J_{Tot} = 3 J_m$
- $J_{Tot} = 2 J_m$
- $J_{Tot} = J_m$

**Time (sec)**
**Electric Vehicles – Modelling and Simulations**

### Fig. 58. Variation of Saturated Velocity Error Pulse Duration with Load Inertia

*ASD Saturated Velocity Error Pulse Duration $\tau$ Variation with Rotor Shaft Inertial Loading for a 4Volt Velocity Command*

**Motor Shaft Load $J_l$ – Multiples of Rotor Inertia $J_m$**

- **ASD Rotor Inertia:** $J_m = 3.04\text{kg}\cdot\text{cm}^2$
- **Motor Current Optimizer:** *MCO – 402B*
- **ASD Simulation Time Step Size:** $\Delta t = 1\mu\text{s}$
- **ASD Velocity Demand $V_{\omega} = 4\text{Volts}$**

---

**BLMD Benchmark Model Reference Simulation for Closed Loop Velocity Control Operation**

ASD Shaft Velocity Output Simulation $V_{\omega r}$ for a Velocity Command Step Input $V_{\omega} = 4\text{ Volt}$ at Various Shaft Inertial Loads $J_{Tot}$

- **Medium Shaft Inertial Load Conditions:** $J_{med} = 12.3\text{kg}\cdot\text{cm}^2$
- **ASD Rotor Inertia:** $J_m = 3.04\text{kg}\cdot\text{cm}^2$
- **Motor Current Optimizer:** *MCO – 402B*
- **Simulation Time Step Size:** $\Delta t = 1\mu\text{s}$

**BLMD Reference Model Shaft Velocity Step Response for Various Inertial Loads $J_{Tot}$**

- $J_{Tot} = 8J_m$
- $J_{Tot} = 7J_m$
- $J_{Tot} = 6J_m$
- $J_{Tot} = 5J_m$
- $J_{Tot} = 4J_m$
- $J_{Tot} = 3J_m$
- $J_{Tot} = 2J_m$
- $J_{Tot} = J_m$

---

**Fig. 59. ASD Velocity Step Response Variation with Inertial Shaft Load**

*BLMD Reference Model Shaft Velocity Step Response for Various Inertial Loads $J_{Tot}$*

**Shaft Velocity O/P $V_{\omega r}$ (Volts)**

- **Medium Shaft Inertial Load Conditions:** $J_{med} = 12.3\text{kg}\cdot\text{cm}^2$
- **ASD Rotor Inertia:** $J_m = 3.04\text{kg}\cdot\text{cm}^2$
- **Motor Current Optimizer:** *MCO – 402B*
- **Simulation Time Step Size:** $\Delta t = 1\mu\text{s}$

---

www.intechopen.com
The variation of the simulated ASD velocity step response overshoot and settling time with shaft inertial load for a 4volt velocity command corresponding to 75% of rated shaft speed in Figures 50 and 49 respectively appear to be lower than those for 2 volt ASD operation at 36% rated speed. This is due to the saturation effect of the velocity error in ASD torque generation, which limits the peak amplitude swing of the oscillatory step response on velocity overshoot in Figure 59. However the rise time is longer in this instance as indicated through ASD simulation in Figure 51 for higher velocity command input (~4V) and increases almost linearly with shaft inertial load. The bigger the load inertia being handled during normal ASD operation the greater the corrective action required in the control effort to limit shaft velocity overshoot with inertial deceleration and settling time with reached velocity experienced with a non optimal speed controller during large step changes in velocity command input. Unless the inertial load is known apriori in this scenario the variable PI parameters cannot be optimally selected for fast risetime and minimum overshoot except through offline manual tuning approximation procedures during the installation and commissioning phase (Moog GmbH, 1988) of the embedded drive in industrial applications. Inertial load parameter extraction in new ASD industrial applications, using the BLMD reference model with step response testing in an offline identification strategy for autotuning purposes, is difficult without knowledge of the initial PI settings of the velocity controller for zero load conditions before drive hookup to the embedded application. In this instance the procedure of accurate inertia parameter extraction using the ASD model in any identification strategy is complicated by the control action of outer velocity loop closure on the wideband inner torque loop when the variable PI settings are unknown. The ASD parameter identification problem is this case has to incorporate evaluation of the existing PI term settings in addition to the inertial load parameter in the new application in order to optimally design the embedded drive velocity controller. However if the drive is configured in torque control mode, thereby eliminating the velocity controller during step response testing in the commissioning phase, inertial load parameters are more amenable to extraction using low torque command input stimuli (<1V) for linear BLMD behaviour with small perturbation of the drive dynamics. The identification problem in this case has to focus only on the extraction of inertial load parameter \( J_{tot} \) and also on the friction coefficient \( B_m \) if required.

5. Conclusions

Further BLMD simulation for various load torque settings, based on model confidence, yield results that compare favourably with those obtained from theoretical considerations using catalogued data for the actual drive concerned. The introduction of stator impedance angle compensation at high shaft speed results in improved motor power factor and better BLMD steady state performance. This is verified theoretically and illustrated through model simulation. Detailed BLMD simulation, configuration as an ASD with velocity feedback, is provided at internal observation nodes and checked against measured data at low and high command speed settings for confirmation of model accuracy and validation purposes.

6. Acknowledgment

The author wishes to acknowledge

i. Eolas – The Irish Science and Technology Agency – for research funding.

ii. Moog Ireland Ltd for brushless motor drive equipment for research purposes.
7. References

Balabanian, N. & Bickart, T.A.; Electrical Network Theory, 1969, J.Wiley & Sons
Franklin, G.F. & Powell, J.D.; Digital Control Of Dynamic Systems, Addison Wesley, 1980
Guinee, R.A.; (2003). Modelling, Simulation, and Parameter Identification of a Permanent Magnet Brushless Motor Drive System, Ph. D. Thesis, NUI – University College Cork.
Guinee, R.A. & Lyden, C.; (1999). Accurate Modelling And Simulation Of A DC Brushless Motor Drive System For High Performance Industrial Applications, IEEE ISCAS'99 - IEEE International Symposium on Circuits and Systems, May/June 1999, Orlando, Florida
Moog GmbH; (1988). Moog Brushless Technology:Brushless Servodrives User Manual, D310.01.03 En/De/It 01.88, Moog GmbH, D-7030 Böblingen, Germany.
Moog GmbH; (1989). Moog Brushless Technology User Manual:D31X-XXX Motors,T158-01X Controllers,T157-001 Power Supply, , D-7030 Böblingen.
In this book, modeling and simulation of electric vehicles and their components have been emphasized chapter by chapter with valuable contribution of many researchers who work on both technical and regulatory sides of the field. Mathematical models for electrical vehicles and their components were introduced and merged together to make this book a guide for industry, academia and policy makers.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Richard A. Guinee (2011). Extended Simulation of an Embedded Brushless Motor Drive (BLMD) System for Adjustable Speed Control Inclusive of a Novel Impedance Angle Compensation Technique for Improved Torque Control in Electric Vehicle Propulsion Systems, Electric Vehicles - Modelling and Simulations, Dr. Seref Soylu (Ed.), ISBN: 978-953-307-477-1, InTech, Available from: http://www.intechopen.com/books/electric-vehicles-modelling-and-simulations/extended-simulation-of-an-embedded-brushless-motor-drive-blmd-system-for-adjustable-speed-control-in