A VECTOR CONTROL SIMULATION FOR INDUCTION MOTOR DRIVES USING SIMNON PROGRAM

Min-Huei Kim
Yeungnam College of Science & Technology
Dept. of Electrical Engineering
1737 Taemyeung-dong, Namgu, Teagu 705-037. Korea
Phone +82-53-650-9263, Fax +82-53-624-4736

Jong-Moo Kim
Korea Electrotechnology Research Institute
Power Electronics Research division
P.O. BOX 20 Changwon, 641-120 Korea
Phone +82-551-280-1483, Fax +82-551-280-1406

ABSTRACT - This paper describes the simulation approaches employed for a vector control system of induction motor drive using SIMNON for windows program. SIMNON program tool can solve differential and difference equations for nonlinear dynamical control system. One powerful feature is its ability of allowing integration of individual program modules after each individual module is programed and tested independently. This particular feature is exploited here for an SVPWM inverter drive by real-time modeling and simulation. The suggested programs are provided a simple and complete simulation for induction motor vector drive system.

1. INTRODUCTION

This paper describes the simulation approaches employed for a induction motor drive system by space vector pulse width modulation (SVPWM) using SIMNON simulation program [1]-[5]. SIMNON that can be help many researcher to solve differential and difference equations is a tool for nonlinear dynamics simulation. It features state space representation for dynamic systems, hybrid linear and nonlinear, continuous and discrete, fast and slow time constants, and interactive graphics [6]. The program can also describe more complicated nonlinear systems, which are interconnections of subsystems. The subsystems can be continuous time systems described by differential equations, i.e., $dx/dt = f(x, u, t)$, or discrete time systems described by difference equations, i.e., $x(t_{k+1}) = f(x(t_k), u(t_k), t_k)$. Moreover, a complex system modeling and simulation can be broken down into several subsystems for individual testing first and later for system integration. In particularly, the new window version of SIMNON/PCW2.0 includes a feature for real-time simulation which offers with a DLL(Dynamic Linked Library) and Matlab function capable of translating a linear, time-invariant dynamic system into its SIMNON equivalent. This particular feature is exploited here for a generally vector control drive simulation of induction motor using SVPWM inverter. Fig.1 shows the typically block diagram of the proposed induction motor vector control system. For ease of simulation programming, the system is divided into several modules: (1) Induction motor modeling, (2) Vector control system, (3) Modeling of SVPWM inverter, (4) Combined drive system.

There are many papers describing the simulation and modeling for each module. In the [7][8], SIMNON Program was illustrated for the simulation of high frequency resonant DC link inverter induction motor, and of AC/DC/AC converter that thyristor-rectifier modeling with harmonic equation and DC link modeling by PWM inverter. A induction motor drive system using SVPWM inverter that typically contains switching logic circuits, control circuitry, an induction machine modeling, presents comprehensive features in the field of simulation for non-linear dynamic system with up-to-date user interface [9][10].

In this paper, the proposed a vector control system for induction motor drive by SVPWM inverter was then real-time modeled and simulated using SIMNON Program. Fig.2 shows the simulation block diagram of the drive system. The suggested program lists are provided a simple and complete simulation in a induction motor drive system.

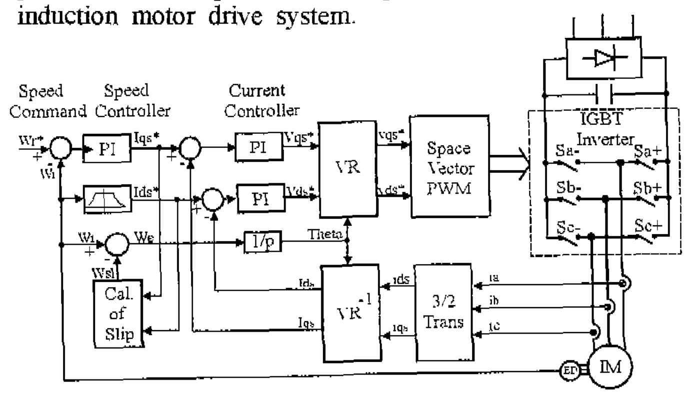

Fig. 1. A schematic diagram of vector control system

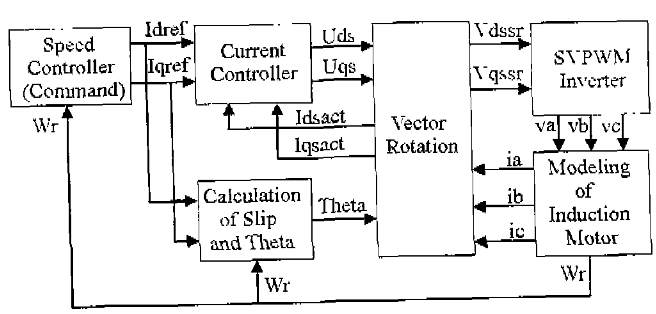

Fig. 2. Simulation block diagram of the drive system

2. INDUCTION MOTOR MODELING

Induction motor modeling are well developed in [1][2]. SIMNON simulation of induction motor drive using d-q model in synchronous rotating frame is illustrated in [3]-[5]. The voltage and motion equations of the induction motor in the reference frame rotating are expressed as the (1)-(3).

$$\begin{bmatrix} V_{ds} \\ V_{qs} \\ V_{dr} \\ V_{qr} \end{bmatrix} = \begin{bmatrix} R_s+L_sP & -\omega_e L_s & MP & -\omega_e M) \\ \omega_e L_s & R_s+L_sP & \omega_e M & MP \\ MP & \omega_{sl} M & R_r+L_rP & -\omega_{sl} L_r \\ \omega_{sl} M & MP & \omega_{sl} L_r & R_r+L_rP \end{bmatrix} \begin{bmatrix} I_{ds} \\ I_{qs} \\ I_{dr} \\ I_{qr} \end{bmatrix} \quad (1)$$

$$T_e = (3/2)(p/2)M(I_{ds}I_{qr} - I_{dr}I_{qs}) = J\frac{d}{dt}\omega_m + B_m\omega_m + T_L \quad (2)$$

$$\omega_{sl} = \omega_e - \omega_r = (1/T_r)(I_{qs}/I_{ds}) \quad (3)$$

Where, ω_e, ω_r and ω_{sl} is flux, rotor and slip angular speed, M is mutual inductance per phase, P is differential operator, p is number of poles, J is initial moment of motor, Tr is rotor time constant. The input voltages for ac machine drive are three phase voltage (v_a, v_b, v_c), generally the outputs are three phase current values (i_a, i_b, i_c) and rotor speed (ω_r). In order to obtain the easy modeling of dynamic systems in the simulation program, the state space differential equations model of induction machine can be derived from the machine model in the d-q stationary reference frame, as modified by (1). The Program List 1 shows a modeling of induction motor (2.2kw, 220V) with included two coordinate transformation(voltage 3/2 and current 2/3) using the Continuous System. The step size, time interval for computation, to solve the differential equations is to set 1 usec. The load torque T_L is in proportion to mechanical rotor speed ω_m, and considered fluctuating load for observing a response characteristics of operating transient state.

Program List 1. Modeling of Induction motor

```
CONTINUOUS SYSTEM IMACH
"Modeling Induction motor (2.2Kw, 220V)
INPUT  Va Vb Vc
OUTPUT Ia Ib Ic wr
STATE Iqs  Ids  Iqr  Idr  x  w
DER  DIqs DIds DIqr DIdr Dx Dw
TIME t
pi = 3.141593
" ----- Voltage Transformations to 3/2 -----
Vqss = (2/3)*0.5*sqrt(3)*(vb-vc)   "va
Vdss = (2/3)*(va-0.5*vb-0.5*vc)    "(vc-vb)/sqrt(3)
Vds = Vqss*sinref + Vdss*cosref
Vqs = Vqss*cosref - Vdss*sinref
" ----- State-Space Differential Equations of IM--------
Diqs = (Lr*(Vqs-Rs*iqs)-fse*ids+Lm*Rr*iqr-Lrm*w*idr)/kk
Dids = (fse*iqs+Lr*(Vds-Rs*ids)+Lrm*w*iqr+Lm*Rr*idr)/kk
Diqr = (Lm*(Rs*Iqs-Vqs)+w*Lsm*ids-iqr*Rr*Ls+fre*idr)/kk
Didr = (-w*Lsm*Iqs+Lm*(ids*Rs-Vds)-fre*iqr-Ls*Rr*idr)/kk
dw = -(Bm/J)*w + (Te-TL)/J*pole/2
" ----- Relationships of Induction Machine ----
wsl = (1/Tr)*(Iqs/Ids)    "Slip calculation
we = wsl + wr
fse = lsr*we-wsl*lmm
fre = we*lmm-wsl*lsr
lrm = Lr*Lm
lsm = Ls*Lm
lsr = Ls*Lr
lmm = Lm*Lm
kk = lsr-lmm
"------ Generation of Flux Angular Reference Speed -----
Dx = we
thet = mod(x, 2*pi)
cosref = cos(thet)
sinref = sin(thet)
"---- Electromechanical Equations -------
Te = 0.75*pole*Lm*(Idr*Iqs-Iqr*Ids)
TL = KL*Wm        "Proportion of mechanical rotor speed
KL = 0.05 +0.01*sqw(t/0.012)  "Fluctuation of load torque
wr = w                         "Electrical rotor speed
wm = pole*wr/2                 "Mechanical rotor speed
rpm = 60*wr/pole*pi            "Calculation of RPM
"------ Inverse Current Transformations to 2/3 ----
Idss = ids*cosref -iqs*sinref
Iqss = iqs*cosref +ids*sinref
ia = idss                          "iqss
ib = 0.5*(-idss +sqrt(3)*iqss ) "0.5*(-iqss -sqrt(3)*idss)
ic = 0.5*(-idss -sqrt(3)*iqss)    "0.5*(-iqss +sqrt(3)*idss)
"--- Motor parameters and Initial values ---------
Rs = 0.7138 "Ohm    / Rr = 0.7732 "Ohm
Ls = 0.079156 "H    / Lr = 0.079156 "H
Lm = 0.07501 "H     / Tr = 0.102
J  = 0.025          "Jm=0.015, Jg=0.010
Bm = 0.003          "[Kg m^2/sec]
pole = 4  / Iqs:0 / Ids:0.00001 / Iqr:0 / Idr:0 / x:0 / w:0
END
```

3. VECTOR CONTROL SYSTEM

The Field Orientated Control (Vector Control) consists in controlling the components of the motor stator currents, represented by a vector, in a rotating reference frame d-q aligned with the rotor flux. The control system requires the dynamic model equations of the induction motor and returns to the instantaneous currents and voltages in order to calculate and control the variables. This control is based on projections which transform a three-phase time and speed dependent system into a two coordinate time invariant system. The vector controlled machines need two constants as input reference, the torque component and the flux component. By maintaining the amplitude of the rotor flux at fixed value we have a linear relationship between torque and torque component (I_{qs}). We can then control the torque by controlling the torque component of stator current vector. The simply model as the fig.1 is utilizing to the simulation.

The Program List 2 shows a speed controller by the Discrete System for the more ease setting sampling time. The reference speed command ω_{ref} is to set a square wave, amplitude 63 rad/sec and frequency 1.0 sec. There are using PI controller and speed sampling period is to set 1 msec. The Program List 3 shows PI current controller by the Discrete System. The current sampling time interval is to set 10 usec such as similarly a implementation system. The Program List 4 show the calculation of slip ω_{sl} and reference rotor flux angular speed ω_e by the Continuous System. The Program List 5 shows the vector rotation due to coordinate 2/3 reference voltage and 3/2 actual current inverse transformation by the Continuous System.

Program List 2. Speed Controller

```
DISCRETE SYSTEM ISC  "Speed Controller
INPUT wr
```

```
OUTPUT idref iqref wref
STATE y
NEW  ny
TIME t
TSAMP ts
pi = 3.14159
ts = t + h
"----- Speed command -------------
wref = if t< 0.05 then 0 else comd   "Starting time at 0.05Sec
comd = 63 *sqw(t/1.0)    "Wref=2*pi*f, at 10 Hz wref=63 [rad/sec]
" ----- Speed loop in electrical [rad/s] -----
Ew = wref - wr
ny = Ew
wd = kps*Ew + wcs*y
"----- Limiter implementation ----------------------
Iqref = IF wd<-Iqlim THEN -Iqlim ELSE IF wd>Iqlim THEN
Iqlim ELSE wd
Idref = if t<0.05 then 0 else 4.2  "Value of No-Load current
"----- Parameters and Inintial Valued ------------
Iqlim : 20   "This limit comes from Israted*sqrt(2)
Kps : 5     "P-Gain
Wcs : 8     "I-Gain
h: 1e-3     "Speed Sampling time : 1 msec
END
```

Program List 3. Current Controller

```
DISCRETE SYSTEM ICC  "Current Controller
INPUT Idref Iqref Idsact Iqsact
OUTPUT Uds Uqs
STATE x1 x2
NEW   nx1 nx2
TIME t
TSAMP ts
ts = t + h
" ------- Computer voltage errors and Control Signals ----
Eid = Idref - Idsact
Eiq = Iqref - Iqsact
Zero = ko
nx1 = Eid
nx2 = Eiq
Vd = Kpd*Eid + Wcd*X1
Vq = Kpq*Eiq + Wcq*X2
Uds =IF Vd<-vdlim THEN -vdlim ELSE IF Vd>vdlim THEN vdlim
ELSE Vd
Uqs =IF Vq<-vqlim THEN -vqlim ELSE IF Vq>vqlim THEN vqlim
ELSE Vq
" ----------- Parameters and Initial values ---------------
vdlim : 220   "Voltage Limit in d-axes
vqlim : 220   "Voltage Limit in q-axes
Kpd : 5      "P-Gain in d-axes
Wcd : 8      "I-Gain in d-axes
Kpq : 5      "P-gain in q-axes
Wcq : 8      "I-Gain in q-axes
X1: 0 / X2: 0 / ko: 0
h: 1e-5      "Current Sampling time : 10 usec
END
```

Program List 4. Slip Calculation

```
CONTINUOUS SYSTEM IWSL  "Slip and Theta Calculation
INPUT Ids Iqs Wr
OUTPUT We theta fdr
STATE x fd
DER   dx dfd
TIME t
pi = 3.141519265
dfd = -(1/Tr)*fd +(M/Tr)*ids    "Calculation of rotor flux
wsl  = (M*Iqs)/(Tr*(err+fd))    "Wsl = (1/Tr)*(Iqs/Ids)
we = Wr + Wsl
fdr = fd
"----- Generation of Angular Reference  ------
Dx = we
thetaa = mod(x, 2*pi)
theta = thetaa
"----- Parameters and Initial values--------
M: 0.07501 / Tr: 0.1024 / err: 0.00001 / fd: 0.31489 / x: 0 / END
```

Program List 5. Vector Rotation

```
CONTINUOUS SYSTEM IVR  "Vcvtor Rotation
INPUT ia ib ic vds vqs theta
OUTPUT Vdssr Vqssr Ids iqs
TIME t
thet = Theta
cosref = cos(thet)
sinref = sin(thet)
"----- Current 3/2 Trnasformation and Inverse VR ---
idss = (2/3)*(ia -0.5*ib -0.5*ic)
iqss = (2/3)*0.5*sqrt(3)*(ib -ic)
Ids = iqss*sinref + idss*cosref
Iqs = iqss*cosref - idss*sinref
"----- Voltage 2/3 Transformations and VR ---
vdssr = Vds*cosref - Vqs*sinref
vqssr = Vqs*cosref + Vds*sinref
END
```

4. MODELING OF SVPWM INVERTER

Pulse width modulation (PWM) technique is used to generate the required voltage or current to feed the motor or phase signals. A space vector PWM method is increasingly used for AC drives with the condition of the desired phase voltages that the harmonic current is as small as possible, and the maximum output voltage based on the space vector theory is 1.155 times as large as the conventional sinusoidal modulation [9]. Taking into consideration the two constraints, three of the switches must always be ON and three always OFF, the upper and the lower switches of same leg are driven with two complementary pulsed signals, there are eight possible combinations for the switch commands. These eight combinations determine eight phase voltage configurations. The diagram in the fig.3 depicts these combinations.

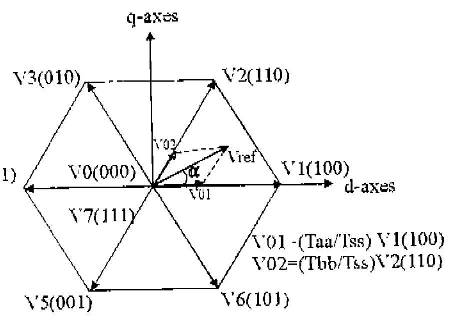

Fig. 3. Space vectors switching pattern for SVPWM

The vectors divide the plan into six sectors. Depending on the sector that the voltage reference V_{ref} is in, two adjacent vectors are chosen. The two vectors are time weighted in sample period T_{ss} to produce the desired output voltage. The complex reference voltage is processed as a whole. The reference vector V_{ref} is sampled at the fixed clock frequency $2F_{ss}$ The sampled value $V_{ref}(T_{ss})$ is then used to solve the equations. Where V_{01} and V_{02} are the two switching-state vectors adjacent in space to the reference vector. The solutions of equation are the respective on-durations T_{11}, T_{22} and T_{00} of the switching-state vectors V_{01}, V_{02} and V_{00}, respectively. The time intervals T_{11} and T_{22}, which define the switching instants, are simply computed in real time from the respective sampled value $V_{ref}(T_{ss})$ using the geometrical relationships.

Assuming that the reference vector V_{ref} is in the #I sector, we have the situation where T_{01} and T_{02} are times during which the vectors V_{01}, V_{02} are applied, and T_{00} the time during which the zero vector are applied. When the reference voltage and sample periods are known, it is possible to determine the uncertainties T_{11}, T_{22} and T_{00} in the (4), where the angle alpha in these equations is the phase angle between the reference vector V_{ref} and V_{01}.

$$V_{ref} = V_{01} + V_{02}$$

$$= (1/T_{ss})(T_{11}V(100) + T_{22}V(110) + T_{00}V(000))$$

$$T_{ss} = 1/F_{ss} = T_{11} + T_{22} + T_{00}$$

$$T_{11} = |V_{ref}|(T_{ss})\sqrt{3}\ \mathrm{Sin}(\pi/3 - \alpha)$$

$$T_{22} = |V_{ref}|(T_{ss})\sqrt{3}\ \mathrm{Sin}\ \alpha \qquad (4)$$

The input for the SVPWM are the reference vector components (V_{dsf}, V_{qsf}) and the outputs are the times to apply each of the relevant sector limiting vectors. The binary representations of two adjacent basic vectors differ in only one bit, so that only one of the upper switches when the switching pattern moves from one vector to the adjacent one. Under these constraints the locus of the reference vector is the inside of a hexagon whose vertices are formed by the tips of the eight vectors. The generated SVPWM waveforms are symmetrical with respect to the middle of each PWM period. The fig.4 shows three phase waveforms of switching pattern (S_a, S_b, S_c) in the example presented above. A section method of the sector (I-VI) and switching-state vectors V_{01}, V_{02} can be chosen and calculated by the table 1. The output phase voltages of SVPWM inverter can be calculated by the (5). The Program List 6 is shown the generating three phase voltage v_a, v_b and v_c into SVPWM inverter by the Discrete System. The time interval for computation is using 1 usec. The fig.5 shows the expanded operating waveforms of SVPWM inverter by the Program List 6: (1) Selecting sector expanded m0, (2) Reference fixed time for current sampling, $2*T_{ss}$ (= 20 usec), (3) Supplied voltage vectors V_{01}, V_{02} and V_{00} during time interval T_{aa}, T_{bb} and T_{cc}, (4) Sa+ switching ON time, (5) Sb+ switching ON time, (6) Sc+ switching ON time.

$$\begin{bmatrix} Va \\ Vb \\ Vc \end{bmatrix} = \frac{1}{3} Vdc \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} Sa \\ Sb \\ Sc \end{bmatrix} \qquad (5)$$

Table 1. Condition of Selected Sector

Sector	Condition	V_{01}	V_{02}
I	Vds>0, Vqs>0, aVds> aVqs3	Vds-Vqs3	2Vqs3
II	Vqs>0, aVds< aVqs3	Vds+Vqs3	Vds+2Vqs3
III	Vds<0, Vqs>0, aVds> aVqs3	2Vqs3	-Vds-Vqs3
IV	Vds<0, Vqs<0, aVds>aVqs3	-Vd+Vqs3	-2Vqs3
V	Vqs<0, aVds< aVqs3	-Vds-Vqs3	-2Vqs3
VI	Vds>0, Vqs<0, aVds> aVqs3	-2Vqs3	d+Vqs3
aVds = | Vds |, aVqs3 = | Vqs |/√3, Vqs3 = Vqs/√3			

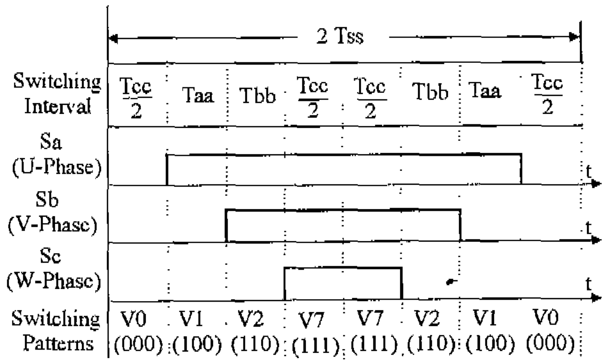

Fig. 4. A symmetric SVPWM switching pattern.

Program List 6. SVPWM Inverter

```
DISCRETE SYSTEM SVPWM  "Space Vector PWM Inverter
INPUT Vdssr Vqssr
OUTPUT Vaa Vbb Vcc
Time t
Tsamp ts
ts = t + h
pi = 3.14259
Vdsf = Vdssr
Vqsf = Vqssr
aVqs3 = abs(vqsf)/sqrt(3)
vqs3 = Vqsf/sqrt(3)
"------ Selectional Sector # (Calculation M) -----
S11 = if (vdsf>0) and (Vqsf>0) and (abs(vdsf)>avqs3) then 1 else 0
S22 = if (Vqsf>0) and (abs(vdsf)<avqs3) then 2 else 0
S33 = if (vdsf<0) and (Vqsf>0) and (abs(vdsf)>avqs3) then 3 else 0
S44 = if (vdsf<0) and (Vqsf<0) and (abs(vdsf)>avqs3) then 4 else 0
S55 = if (Vqsf<0) and (abs(vdsf)<avqs3) then 5 else 0
S66 = if (vdsf>0) and (Vqsf<0) and (abs(vdsf)>avqs3) then 6 else 0
m0 = S11 + S22 + S33 + S44 + S55 + S66
"----- Switching ON-OFF Time calculation -----
Gamma = ATAN2(Vqsf,Vdsf)
Avdsf = sqr(vdsf)
Avqsf = sqr(vqsf)
Avdq = Avdsf +Avqsf
Vampl = sqrt (Avdq)
Tss = 1 /(fss)
Tsss = 2*Tss
Trep = mod(t, Tsss)
Alp = gamma -(m0-1)*pi/3
alpha =if alp>0 then alp else (if alp <0 then (2*pi+alp) else 0)
Vamp2 = (Tss*sqrt(3) *Vampl/Vdc)
Taa = Vamp2 *sin(pi/3-alpha)
Tbb = Vamp2 *sin(alpha)
Tcc = Tss-(Taa+Tbb)
Taa1 = Tss*Taa/(Taa+Tbb)  "Calculation for overmodulation
Tbb1 = Tss*Tbb/(Taa+Tbb)
T11 = if Tcc>0 then Taa else (if Tcc<0 then Taa1 else 0)
T22 = if Tcc>0 then Tbb else (if Tcc<0 then Tbb1 else 0)
T00 = if Tcc>0 then Tcc else 0
"----- Switching Patterns -----
TTa = T00/2
TTb = T00/2 + T11
TTc = T00/2 + T22
TTd = T00/2 + T11 + T22
TTe = T00 + T11 + T22
Ka1 = if Trep< TTa then 1 else 0
Ka2 = if Trep< TTb then 1 else 0
Ka3 = if Trep< TTd then 1 else 0
ka4 = if Trep< TTe then 1 else 0
Ka5 = if Trep< (TTe+TTa) then 1 else 0
Ka6 = if Trep< (TTe+TTc) then 1 else 0
Ka7 = if Trep< (TTe+TTd) then 1 else 0
Ka8 = if Trep< (TTe+TTe) then 1 else 0
Ka9 = if Trep< TTc Then 1 else 0
Ka10 =if Trep< (TTe+TTb) then 1 else 0
"----- Table of V01 & V02 ------
V11 = Vdsf - Vqs3
V12 = 2*Vqs3
V21 = Vdsf + Vqs3
V22 = -vdsf + Vqs3
```

```
V31 = 2*Vqs3
V32 = -vdsf - Vqs3
V41 = -Vdsf + Vqs3
V42 = -2*Vqs3
V51 = -Vdsf - Vqs3
V52 = vdsf - Vqs3
V61 = -2*Vqs3
V62 = Vdsf + Vqs3
V77 = 0
"----- Calculation of va, vb & vc by V01 V02 -----
KT11 = (not Ka1 and Ka2) or (not Ka6 and ka7)
KT22 = (not Ka5 and Ka6) or (not Ka2 and Ka3)
KT33 = ka1 or (not Ka3 and Ka5) or (not ka7 and ka8)
Ya1 = not ka1 and ka7
yb1 = not ka2 and ka6
yc1 = not ka3 and ka5
Vaa11 = if (S11>0 and KT11 >0) then 1 else -1  "V11
Vaa12 = if (S11>0 and KT22 >0) then 1 else -1  "V12
Vaa2 =  if (S22>0 and KT11 >0) then 1 else -1  "V21
Vaa5 =  if (S55>0 and KT22 >0) then 1 else -1  "V52
Vaa61 = if (S66>0 and KT11 >0) then 1 else -1  "V61
Vaa62 = if (S66>0 and KT22 >0) then 1 else -1  "V62
Va = Vaa11 or Vaa12 or vaa2 or Vaa5 or Vaa61 or Vaa62
Vbb1 =  if (S11>0 and KT22 >0) then 1 else -1  "V12
Vbb21 = if (S22>0 and KT11 >0) then 1 else -1  "V21
Vbb22 = if (S22>0 and KT22 >0) then 1 else -1  "V22
Vbb31 = if (S33>0 and KT11 >0) then 1 else -1  "V31
Vbb32 = if (S33>0 and KT22 >0) then 1 else -1  "V32
Vbb4 =  if (S44>0 and KT11 >0) then 1 else -1  "V41
Vb = Vbb1 or Vbb21 or Vbb22 or Vbb31 or Vbb32 or Vbb4
Vcc3 =  if (S33>0 and KT22 >0) then 1 else -1  "V32
Vcc41 = if (S44>0 and KT11 >0) then 1 else -1  "V41
Vcc42 = if (S44>0 and KT22 >0) then 1 else -1  "V42
Vcc51 = if (S55>0 and KT11 >0) then 1 else -1  "V51
Vcc52 = if (S55>0 and KT22 >0) then 1 else -1  "V52
Vcc6 =  if (S66>0 and KT11 >0) then 1 else -1  "V61
Vc = Vcc3 or Vcc41 or Vcc42 or Vcc51 or Vcc52 or Vcc6
vaa = (Vdc/3)*(2*va -vb -vc)   "Phase Voltage
Vbb = (vdc/3)*(-va +2*vb -vc)
Vcc = (vdc/3)*(-va -vb +2*vc)
"----- Parameter values -----
h : 1e-6      "Step size (Interval time) 1 usec
fss = 10000   "Current sampling fequency. 10KHz
Vdc = 310     " Bus Voltage
END
```

5. COMPLETE DRIVE SIMULATION

Many systems can often viewed as interconnections of subsystems, which are represented by differential and difference equations. To represent subsystems declarations of inputs and outputs are added to the system descriptions. The system descriptions are otherwise the same as the Continuous System and the Discrete System described previously in the Program Lists. The interconnections between the systems are described by a Connecting System as a Macro. The Program List 7 shows the Connecting System for this simulation, where "/" symbols mean "starting the statement at the next line".

After the individual module is tested and proven to be working, it is not necessary to reorganize each module for combined simulation. SIMNON program provides a feature that allows individual program modules to be connected together and run under a Macro. The following Marco shows the linking among all the modules presented. As shown in the Macro, parameters for different simulation scenario can be defined here as well. In addition, a Macro defining programs to be run, variables to be stored, simulation interval, integration step size, plots to be displayed, etc. can be written accordingly. The Program list 8 shows Marco program of this simulation in order to

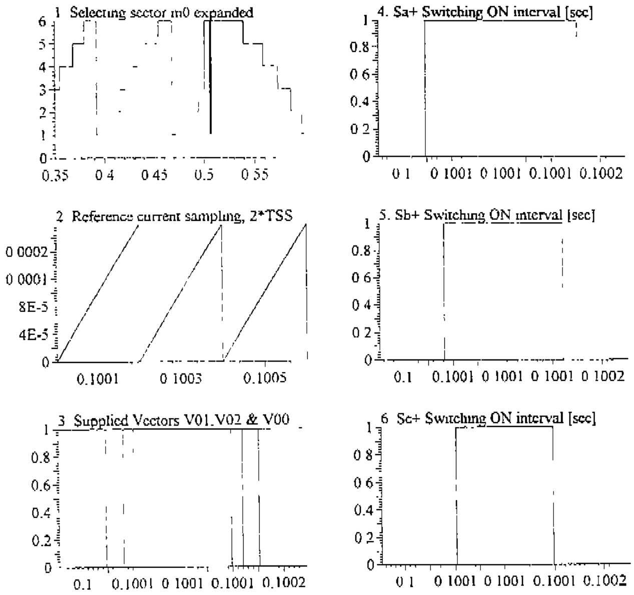

Fig. 5. Operating result of SVPWM

obtain the operating results as the fig.5 and fig.6. The fig.6 shows the simulation results for the vector control drive system during a transient state: (1) Response characteristics of speed control, W_{ref} and W_r [rad/sec], (2) Fluctuation load torque waveform, T_L [kg-m], (3) Response waveforms of rotor flux and slip angular speed, W_e and W_{sl} [rad/sec], (4) Computing Theta for vector rotation, (5) Command and actual current at d-axes, I_{dref} and I_{dact}, (6) Command and actual current at q-axes, I_{qref} and I_{qact}, (7) Output voltage waveform of A-phase, v_a, (8) Output current waveform of A-phase, i_a, (9) Expanded three phase current waveforms, i_a, i_b and i_c, (10) Expanded v_a waveform, (11) Expanded v_b waveform, (12) Expanded v_c waveform.

Program List 7. Connection ICONN

```
CONNECTING SYSTEM ICONN  "IPCE98 Paper Connecting
wr[isc] = wr[imach] / idref[icc] = idref[isc] / iqref[icc] = iqref[isc]
Idsact[icc] = Ids[ivr] / Iqsact[icc] = Iqs[ivr] / fdr[icc] = fdr[iwsl]
we[icc] = we[iwsl]  / Ids[iwsl] = Idref[isc] / Iqs[iwsl] = Iqref[isc]
wr[iwsl] = wr[imach] / theta[ivr] = theta[iwsl] / Vds[ivr] = Uds[icc]
Vqs[ivr] = Uqs[icc] / ia[ivr] = Ia[imach] / ib[ivr] = Ib[imach]
ic[ivr] = Ic[imach] / Vdssr[svpwm] = Vdssr[ivr]
Vqssr[svpwm] = Vqssr[ivr] / va[imach] = Vaa[svpwm]
vb[imach] = Vbb[svpwm] / vc[imach] = Vcc[svpwm] / END
```

Program List 8. Macro ICPE

```
MACRO ICPE   " ICPE'98 Paper
"Author:       Min-Huei Kim
"Created:      98-07-01
SYST  ISC ICC IVR IWSL SVPWM IMACH ICONN
STORE wref[isc] wr[Imach] Tl[imach] Zero[icc]
STORE ia[Imach] ib[Imach] ic[Imach] -add
STORE Idref[icc] Iqref[icc] Idsact[icc] Iqsact[icc] -add
STORE we[iwsl] wsl[iwsl] theta[iwsl] Trep[svpwm] m0[svpwm] -add
STORE Vaa[svpwm] Vbb[svpwm] Vcc[svpwm] -add
STORE va1[svpwm] yb1[svpwm] yc1[svpwm] -add
STORE kt11[svpwm] kt22[svpwm] kt33[svpwm] -add
split 3 2
simu 0 0.8 1e-6    "Step size 1 usec
                   "Simulation time is from 0 to 0.8 sec
disp (lp) -state   "make the Simnon.log File
axes H 0 0.8 V -80 80 / show Wref Wr
```

```
text '1. Speed Control, wref & wr [rad/sec]'
axes H 0 0.8 V -8 8 / show TL
text '2. Fluctuation Load Torque TL [Kg-m]'
ashow we wsl / text '3. Rotor flux and slip speed We & Wsl'
ashow 0.3 0.6 Theta / text '4. Expanded Theta '
ashow Idref Idsact zero / text '5. d-axes current, Idref & Idsact [A]'
ashow Iqref Iqsact / text '6. q-axes current, Iqref & Iqsact [A] '
Newplot / split 3 2
ashow vaa / text '7. Voltage wave of A-phase, va ' / ashow ia
text '8. Current wave of A-phase, ia ' / ashow 0.4 0.6 ia ib ic /
text '9. Expanded 3-phase Current ia,ib & ic' / ashow 0.4 0.6 vaa
text '10. Expanded voltage of A-phase, va' / ashow 0.4 0.6 vbb
text '11. Expanded Voltage of B-phase, vb' / ashow 0.4 0.6 vcc
text '12. Expanded voltage of C-phase, vc'
Newplot / split 3 2
ashow 0.35 0.6 m0 zero / text '1. Selecting sector m0 expanded'
ashow 0.1 0.1004 Trep / text '2. Reference current sampling, 2*TSS'
ashow 0.1 0.1002 KT11 KT22 KT33
text '3. Supplied Vectors V01,V02 & V00 '
ashow 0.1 0.1002 ya1 / text '4. Sa+ Switching ON interval [sec] '
ashow 0.1 0.1002 yb1 / text '5. Sb+ Switching ON interval [sec] '
ashow 0.1 0.1002 Yc1 / text '6. Sc+ Switching ON interval [sec] '
END
```

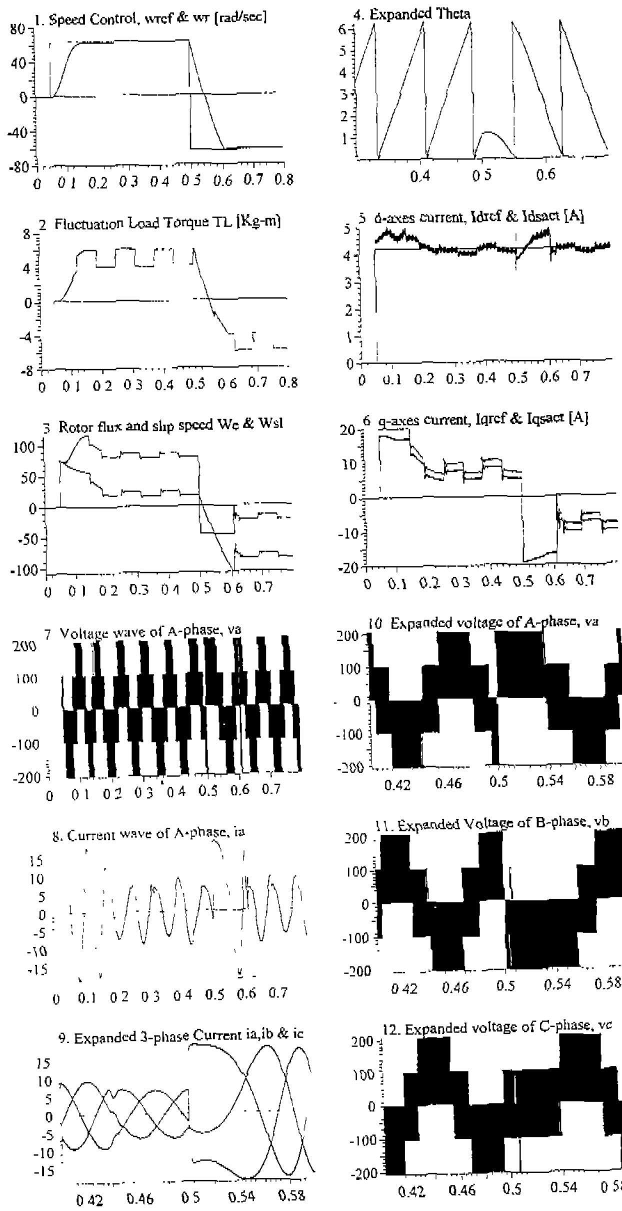

Fig. 6. Simulation results of the drive system

6. CONCLUSION

In conclusion, this paper provides a simple and complete simulation program of SVPWM inverter for induction motor vector drive system. The setup using SIMNON is an excellent tool for studying and understanding the dynamics involved with the non-linear drive system. The drive system considered is not only suitable for performance evaluation, but also suitable for dynamic motor control system design. Any complicated control algorithm can be easily incorporated into the system as a separate module for further study of speed, current and torque control in a vector control drive system for induction motor. The suggested program lists are provided a simple and complete real-time modeling and simulation using SIMNON Program in a induction motor vector drive system. It will be expected in some simulation that more complicated induction motor drive system technology.

7. REFERENCES

[1] B. K. Bose, "Power Electronics and AC Drives, Prentice Hall", 1986

[2] D.W. Novotny and T.A. Lipo, Vector control and Dynamics of AC drive, Oxford Press, 1996.

[3] Min-Huei Kim, James C. Hung, "Vector Control System for Induction Motor without Speed Sensor at very Low Speed ", IECON95, pp524-529, 1995

[4] Bimal K. Bose, Min-Huei Kim and M.K. Kankam, "High frequency ac vs. dc distribution system for next generation hybrid electric vehicle", IECON '96, pp706-712, 1996.

[5] Min-Huei Kim, "Advanced High Frequency AC Propulsion System for Next Generation Electric Hybrid Vehicle", JKIPE, Vol.2, No.1, pp 12-24, 1997.

[6] SSPA Maritime Consulting AB, SIMNON for window Version 2.0, Sweden, 1995.

[7] Fong K. Mak, "AC/AC/AC Converter Drives Simulation Using Simnon", IECON95, pp525-530, 1995.

[8] J. S. Lai and Bimal K. Bose, "A PC-Based Simulation and DSP-Based Control of an Improved high Frequency resonant DC Link Inverter Induction motor Drive", IECON'90, pp882-888, 1990.

[9] Bimal K. Bose, "Power Electronics and Variable Frequency Drives", IEEE Press, pp 138-208 1997.

[10] Robert L. woods and Kent L. Lawrence, Modeling and Simulation of Dynamic systems, Prentice-Hall, 1997.