

# Development of a Speed Stabilizer for Rapid Synchronization of Mini-Hydro Generator

J.P. Karunadasa and D.G. Subasinghe

**Abstract:** Several causes may affect a Mini-Hydro generator to shut down whilst in operation. They can be the earth faults in the electricity lines (mainly way tree leaves touching the transmission lines), lightning, planned and unplanned interruptions in that particular area connecting to the load bus of the distribution network and for maintenance of the plant itself. To resume power export to the grid after each shutdown will require generator to synchronize with the grid supply. Since the generator Rotor-Turbine assembly is a heavy massed system and the viscous damping is very low, stabilizing of rotor speed at the reference set point is a time taking exercise. The time taken for synchronization will typically be in the range of 2-10 minutes depending on the plant design. This paper describes the work of authors on the development of a speed stabilizer to minimize synchronization time of a chosen Mini-Hydro generator plant. This involves the modeling of the entire plant, identifying of damping strategies, designing of power electronic switching circuit and associated closed loop control, MATLAB simulation for parameter fine-tuning and final site testing. The designed system introduces a controlled resistive load on the generator during the synchronization transient to expedite settling. Test results confirmed the reduction of synchronization time as expected.

**Keywords:** Generator Synchronization, Damping, Resistive Load

## 1. Introduction

Several causes may affect a Mini-Hydro generator to shut down whilst in operation. They can be outlined as, earth faults in the electricity lines (mainly way tree leaves touching the transmission lines), lightning, planned and unplanned interruptions in that particular area of the distribution network and for maintenance of the plant itself.

To resume power export to the grid after a plant shutdown, it is necessary to synchronize the generator with the grid supply, either by automatic controls or by manual means. In either process, the Hydro Turbine ramps up to near synchronous speed and during this period the Input valve (wicket gates for Francis and Kaplan turbines, runner blades for Kaplan turbines, or Nozzle jets for Pelton Turbines) starts to open in steps so that the hydro turbine starts receiving energy to gradually accelerate the speed. The speed acceleration has limitations, determined by the Penstock characteristics. The ramping period considered here is the time taken by the generator to ramp up the speed from stationary state to about 95% of the rated speed (or 95% of rated frequency, which is 47.5Hz). At this point, the synchronization process begins where it gives biasing signals to raise or lower the speed of the rotor to synchronize with the bus frequency.

Synchronization time is the time between the end of ramping and the point of closing the generator breaker after synchronization (at 50 Hz).

The time taken for rotor speed ramping will typically be in the range of 3-6 minutes and then generator synchronizing would take another 2-10 minutes depending on the plant design.

### 1.1 Loss of Energy Production during Synchronization

Most of the Mini-Hydro installations are run-of-the-river type (not storage type) and therefore the amount of power generated at a given time depends on the flow level of the stream and availability of the water in the fore bay tank. Therefore, the loss of energy production during a downtime is not recoverable later by increasing the generator load factor.

*Eng. (Dr.) J. P. Karunasena, B.Sc. Eng (Moratuwa), M. Sc. (Manchester), Ph.D. (Manchester), C. Eng., MIE (Sri Lanka), Head, Dept. of Electrical Engineering, University of Moratuwa.*

*Eng. D. G. Subasinghe, B.Sc. Eng (Hons) (Peradeniya), M. Sc. (Moratuwa), C. Eng., MIE (Sri Lanka), MIET, Manager Power Generation, tp1 (Distributor for Cummins Inc., U.S.A.)*



The loss of energy production due to each plant shutdown is substantial when accumulated for a year. Even though the Ramping Period is constrained by the design of the plant itself, the minimization of synchronization period is a clear alternative to minimize the total downtime. Further as per the figure 1.1 the synchronization period is generally longer than the ramping period. Therefore, there is a reasonable potential to minimize the start up downtime by approximately 25-50% by optimizing the synchronization time.

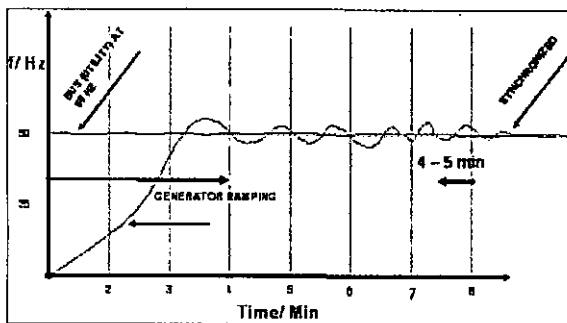


Figure 1.1 - Graph of a typical Generator speed Vs Time during Synchronization.

The work described in this paper is about the development of a speed stabilizer that can dampen the rotor speed appropriately to stabilize it at a reference input within a desired time. The prototype stabilizer constructed for the chosen Mini-Hydro plant produced good results proving the concept and showing a reduction of synchronizing time. Mathematical modeling and MATLAB simulation formed the basis of estimating controller parameters. Manufacturer's literature gave values of some other parameters.

The speed stabilizer developed in this project is a PI controller programmed in a PIC 16F877A with a power electronic circuit to switch a resistive load bank. The resistive load is switched on as the rotor speed tends to exceed bus speed and thereby imparts an electrical braking effect on the rotor, which in turn stabilizes the rotor speed.

## 2. Methodology

### 2.1 Model of the Mini-Hydro Generator

The main details of the identified project for the study are as follows:

Capacity: 1MW, 415V, 50Hz, 750RPM, 8 pole, Synchronous Generator.

Turbine: Francis Turbine, with 12 Numbers Wicket Gates

Head: 100m

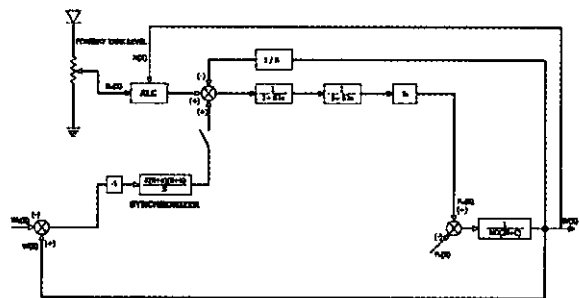


Figure 2.1 - Model of the chosen Mini-Hydro Generator

Figure 2.1 gives the mathematical model [1] of the system.

- J = Moment of Inertia of the rotor assembly, kgm<sup>2</sup>
- C = Viscous damping coefficient, kgm<sup>2</sup>/sec
- $\omega_s$  = Grid frequency converted to speed, rad/sec
- $\omega$  = Speed of the rotor, rad/sec
- $\omega_0$  = Near-synchronous speed at which system modeled, rad/sec
- R = Governor speed-droop, %
- T<sub>G</sub> = Governor Time Constant, sec
- T<sub>H</sub> = Hydro turbine Time Constant, sec
- K<sub>G</sub> = Governor gain
- P<sub>V</sub> = Change of Inlet valve position
- K<sub>T</sub> = Turbine Gain
- K<sub>a,b</sub> = PID parameters of the present synchronizer
- P<sub>m</sub> = Mechanical power by turbine, W
- P<sub>L</sub> = Power supplied to the load (grid), W

## 2.2 Model of the Mini-Hydro Generator during Synchronization

During the synchronization process the function of speed droop and automatic loading control (ALC) are disabled and the Synchronizer will govern the speed control of the system.

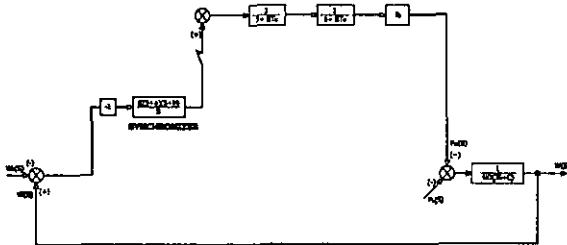


Figure 2.2 - Model of the Mini-Hydro generator during synchronization.

As shown in Figure 2.2, parameters involved in this model are  $K$ ,  $a$ ,  $b$ ,  $TG$ ,  $TH$ ,  $KT$ ,  $P_m$ ,  $\omega$ ,  $\omega_s$ ,  $J$  and  $C$  which are either derived or estimated using computational and experimental methods. At the summation block,  $\omega$  and  $\omega_s$  are measured quantities which are sensed from generator frequency and grid frequency respectively. Manufacture's datasheets gave the values of Moment of inertia  $J$  and time constant values  $TG$  and  $TH$ . Parameter  $KT$  depends on the value of  $P_m$  and it is estimated against the maximum possible value of  $P_m$ . PID values are derived using MATLAB simulation.

### 2.3 Estimation of value for C

Value of  $C$  is unknown and estimated experimentally, with reference to free deceleration test results given in Figure 2.3.

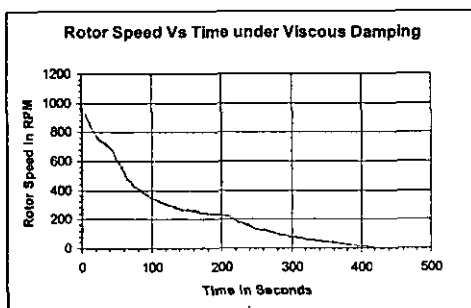


Figure 2.3 - Rotor Speed Vs Time when rotor spinning freely

When the Turbine gate is closed,  $T_m = 0$ . Then, for the freely decelerating rotor,

$$C = \frac{J \ln(\dot{\theta})}{\omega_s t}$$

Where,  $\dot{\theta}$  is measured speed of Rotor in rad/s. Estimated value for  $C = 0.8 \text{ Kgm}^2/\text{sec}$

### 2.4 Estimation of $P_m$

Since  $P_m$  deals with absolute power values of maximum 1MW, it is required to find out an estimated values for  $P_m$  and  $K_T$  during the synchronization process.

Maximum value of  $P_m$  during synchronization estimated as follows:

$J$ :  $250 \text{ kgm}^2$   
 Time span: 15 sec  
 Speed change: From 712.5 to 750.0 RPM  
 (74.613 to 78.540 rad/s)

$$P_m = \frac{250 * (78.54^2 - 74.613^2)}{15}$$

$$P_m = 10,024 \text{ W}$$

### 2.5 Estimation of $K$ , $a$ & $b$ using Matlab

Estimation of the PID values of the synchronizer takes the present system's settling time as a design parameter. The MATLAB program uses an iterative method [3] to compute the optimum values for the parameters. Appendix 1 shows the MATLAB program used to find PID values of the synchronizer.

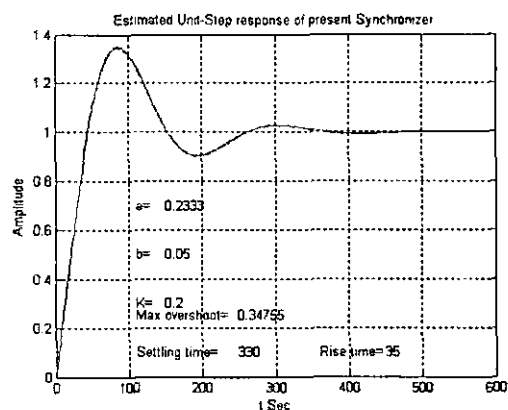


Figure 2.4 - Unit-Step response of present Mini-Hydro Generator.



## 2.6 Model of the Mini-Hydro Generator with 'Resistive Load' in position during Synchronization

In this scenario, power equilibrium [2] of the hydro-turbine generator is as follows.

$$\omega(s) = \frac{P_m(s) - P_e(s)}{\omega_0 (Js + C)}$$

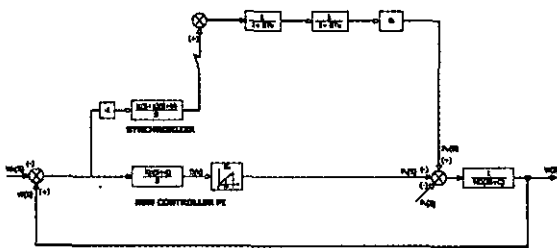


Figure 2.5 - Model of Mini-Hydro Generator with 'Artificial Load' during synchronization

## 2.7 New PI controller to switch Resistive Load

The function of the new controller is to switch a resistive load in proportional to actuating error  $e(s)$  (Proportional control) to incorporate damping to the system and thereby reduce the settling time. Moreover, it is required to settle the system exactly at reference input level, which is at the grid frequency (Integrative Control) in order to support present synchronizer for rapid synchronizing. In this context, high sensitive error-correction control action responding to rate of change of error (derivative control) is not appropriate due to two main reasons.

1. Derivative term could amplify disturbances of input or noise as the PID is not well tuned. This can prompt oscillations or the system can become unstable.
2. PID controller will permit fast changes of larger values of resistive load,  $P_e(s)$  on the generator which may disturb the voltage matching process of AVR.

Thus, a PI controller is more appropriate than a PID controller for switching the resistive load,  $P_e(s)$  for this application.

## 2.8 MATLAB Program to Estimate $K_p$ , $d$

In order to estimate the PI values of the proposed new controller, the desired settling time of the new system and Maximum overshoot will

be design parameters. The program used in MATLAB to estimate PI parameters [3] are given in Appendix 2.

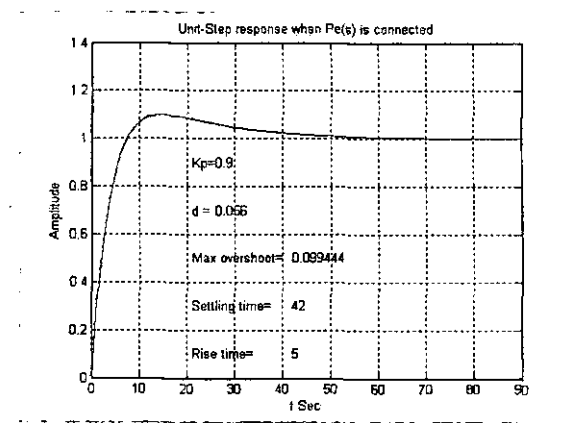


Figure 2.6 - Unit-Step response of the Mini-Hydro Generator with Artificial load.

## 3.0 Switching Circuit

The new PI controller generates control signal according to the error(s) during synchronization. Figure 2.6 shows the response for a unit step input of the system. The PI controller programmed in 'PIC' 16F877A generates a PWM signal and the Duty Factor is varied corresponding to PI Controllers' output control signal. The switching frequency of the PWM signal is pre-set in the processor at 5 kHz. The magnitude of PWM pulses from the processor will be at 0-4.8 Volts that feeds the gate of IGBT [5] via a 'Gate Driver' circuit. Therefore, IGBT switches the load in a controlled manner.

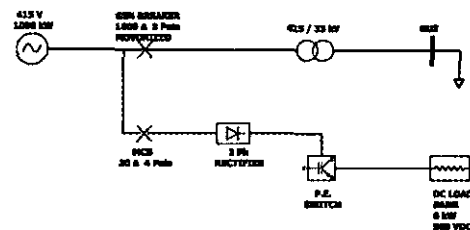


Figure 3.1 - Line Diagram of the Switching Circuit.

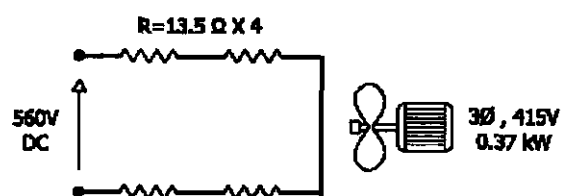


Figure 3.2 - Circuit of the DC Load Bank

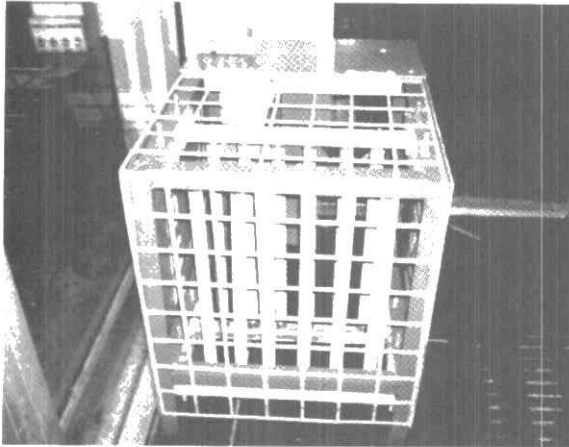


Figure 3.3 - load bank, connected through a 40A/ 4P MCB

Figure 3.1 shows the line diagram of the switching circuit.

AC input Voltage = 415 V rms  
 DC Bus Voltage = 560 V DC  
 Max. DC load current at Duty Factor 1.0

$$= \frac{6000}{560} = 10.375A$$

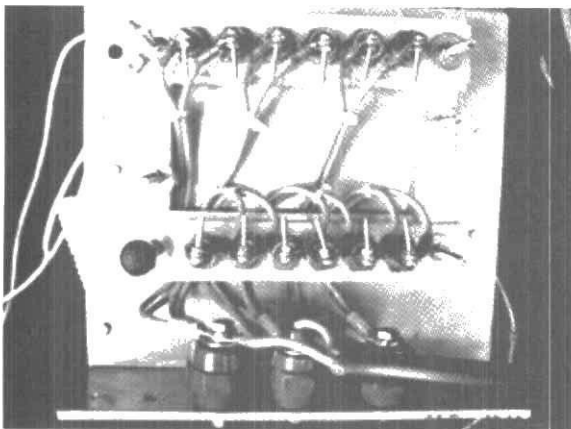


Figure 3.4 - A picture of 3 Phase Rectifier

### 3.2 IGBT Gate Driver Circuit

The maximum calculated Voltage and Current across the semi conductor switch in order to switch 6 kW resistive load will be 560 V DC and 10.7 A respectively. The proposed frequency of the switching would be in the range of 1 kHz-5 kHz. Figure 3.5 shows a summary of device capabilities [6]. Based on power capabilities, switching speed, and considering gate controllability by means of a voltage signal, an IGBT is decided and Toshiba-GT50J101 [11] selected.

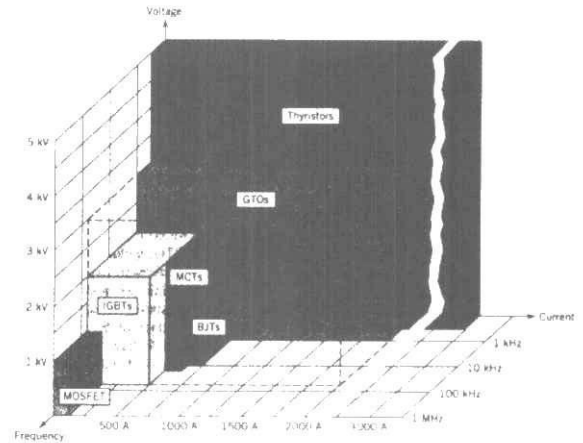


Figure 3.5 - Summaery of semiconductor device capabilities. Source: Ned Mohan [6]

PWM output signal from PIC16F877A [10] (pin # 16) is isolated through the Optocoupler PC817 [12] (see Figure 3.6) and connected to the gate at 15 Volts after amplification [8] through a switching transistor D313 [13].

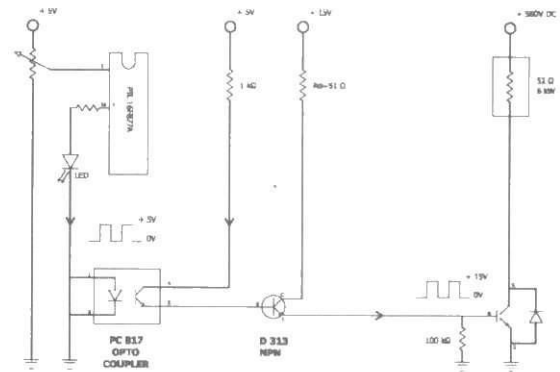


Figure 3.6 - IGBT Gate Driver Circuit

### 3.3 Speed Sensing Circuit

As per the model shown in the Figure 2.5, it is required to sense the electrical speed of the grid supply  $\omega_s$  and Generator supply  $\omega$  at the summation point to generate speed error signale(s).

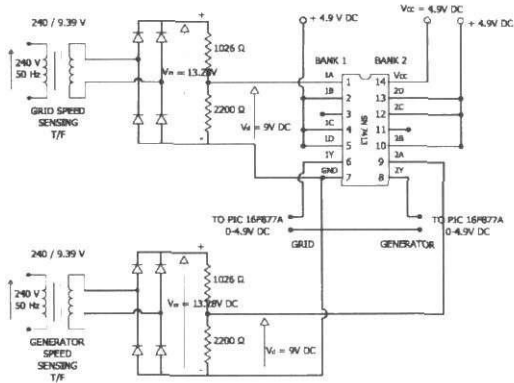


Figure 3.7 - Diagram of Speed Sensing Circuit

As shown in Figure 3.7 two single-phase step-down transformers sense the cycle speed of both sources. The full wave rectified positive half Sine wave signal [9] with 9 V DC Maximum Voltage feeds Pin 1 (1A) of the Schmitt Trigger [7]. At the Schmitt Trigger, high-level reference voltage is preset at 4.9V DC at Pin 14. All other NAND gate pins 1B, 1C, 1D are given logic 1 (4.9V DC). Thus, full wave rectified signal at Pin1 is converted to Square wave pulses at Pin 6 (1Y).

Two Schmitt Trigger process the signals from grid supply and generator supply. The two square wave signals derived from them feed Pins 2 and 3 (PORT A, RA0 and RA1) of the Microcontroller.

Table 3.1 - Truth Table of 4 input NANDgate with Hysteresis.

1A (pin 1)	1B(pin 2)	1C(pin 4)	1D(pin 5)	1Y(pin 6)	Reference (pin 14)	GND (pin 7)
Full wave rectified sine wave input						
$\geq 4.9V \Rightarrow 1$	1	1	1	0	4.9V	0V
$0.0V \Rightarrow 0$	1	1	1	1	4.9V	0V

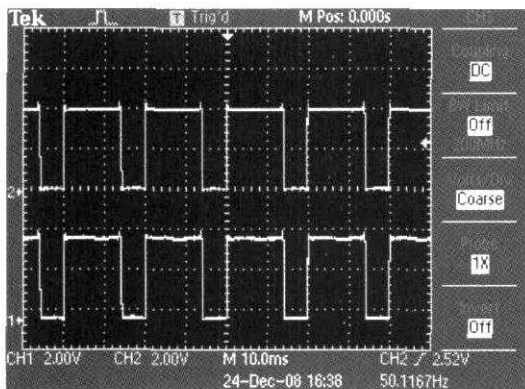


Figure 3.8 - Square wave signal output from Schmitt Trigger, converted from Sinusoidal voltage sources.

### 3.4 Implementation of PI controller

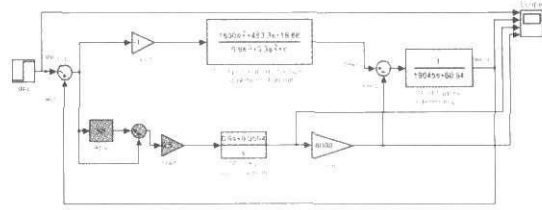


Figure 3.9 - Model of continuous PI controller.

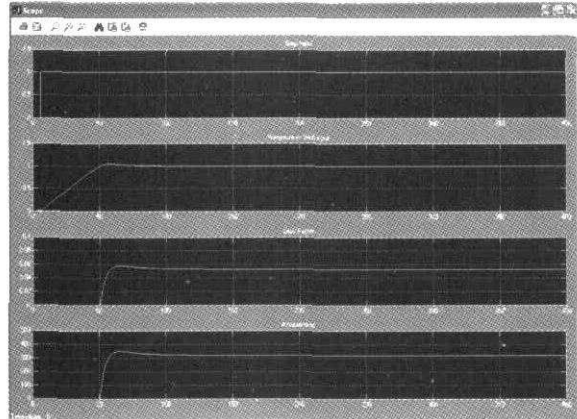


Figure 3.10 - Results of continuous PI controller simulation.

Digital implementation of PI controller [4] assumes a sampling time  $T_s$  of 0.04 seconds (40ms) which is determined by the time taken by CPU to detect a speed error between two digital pulse signals (from grid and generator sources). MATLAB code that converts continuous PI controller to digital PI controller is as follows:

% tf of the new PI controller in S plane

$K_p=0.9;$

$d=0.056;$

$num=[0 K_p K_p*d];$

$den=[0 1 0];$

$PI\_TFc=tf(num,den)$

$PI\_TFd=c2d(PI\_TFc,0.04,'zoh')$

Result: Discrete-time Transfer function of PI

$$Controller = \frac{0.9z + 0.899}{z - 1}$$

Sampling time: 0.04 sec

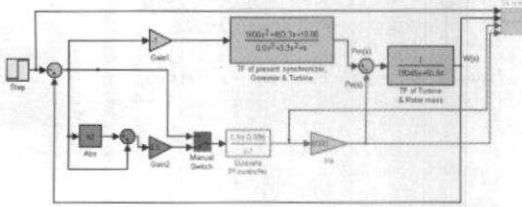


Figure 3.11 Model of the system with discrete PI controller.

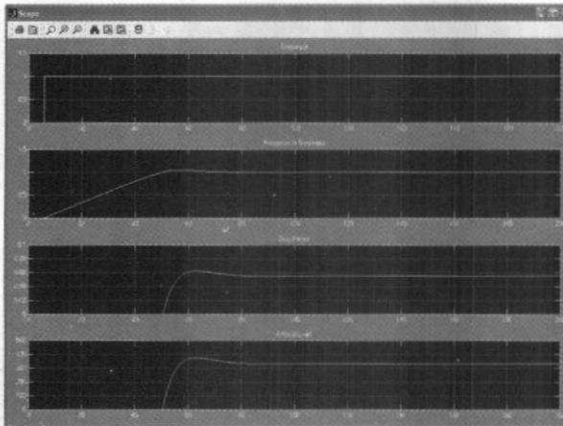


Figure 3.12 Results of discrete PI controller simulation for a step input

The response curves of the discrete PI controller obtained in Figure 3.12 exhibits a small stair case shape in response curves (when zooming). However when compared with those of continuous PI controller responses in Figure 3.10, both set of results are identical. Therefore, in the implementation of PI control algorithms in microcontroller, the system behaves like continuous because the synchronization is a slow process, and the sampling time 40ms is very small compared to settling time in the range of 40-125 seconds.

## 4.0 Results and Conclusion

### 4.1 Testing at Site

The PI controller was set up at the site with power analyzer (model Fluke1735) to take measure the frequency Vs time readings during synchronization. First, the generator started and synchronized in a normal operation without PI controller, and recorded the frequency Vs time readings (see Figure 4.1).

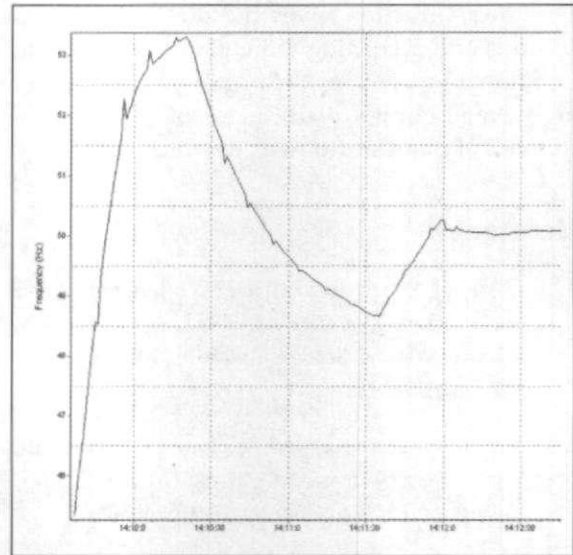


Figure 4.1 - Frequency Vs Time during synchronization under normal operation.

Then tested with the speed stabilizer and recorded the frequency Vs time during synchronization (see Figure 4.2).

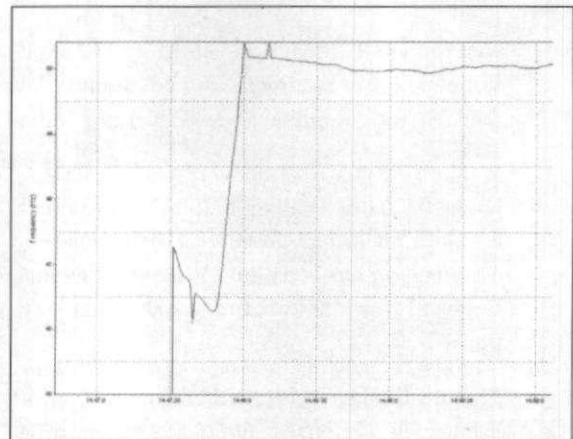


Figure 4.2 - Frequency Vs Time during synchronization with speed stabilizer

In this testing synchronizing time was 48 sec.

### 4.2 Conclusion

Test results indicate that the synchronization time after the introduction of speed stabilizer comes down to 48 seconds from the original 142 seconds. This is a significant achievement as far as the goals of the projects are concerned. The frequency overshoot is also now low, down from 3.5 Hz to 0.5 Hz.

The time reduction saves the downtime by 94 seconds and accordingly the plant can generate additional units of 360kWhrs per year assuming an average annual plant factor of 30% and 46 number of plant shutdowns per year.

## References

1. Allen J. Wood and Bruce F. Wollenberg, *Power Generation Operation and Control*, 2nd Edition, John Wiley & Sons, Singapore, 2005, Chapter 9, pp. 328-362.
2. John J. Grainger and William D. Stevenson, Jr., *Power System Analysis*, McGraw-Hill, Inc., Singapore, 1994, Chapter 16, pp. 695-707.
3. Katsuhiko Ogata, *Modern Control Engineering*, 4th Edition., Prentice Hall of India Pvt. Ltd, New Delhi, 2006.
4. Katsuhiko Ogata, *Discrete-Time Control Systems*, 2nd ed., Prentice Hall of India Pvt. Ltd, New Delhi, 2005.
5. Cyril W. Lander, *Power Electronics*, 3rd ed., McGraw-Hill Book Co., Singapore, 1993.
6. Ned Mohan, Tore M. Undeland and William P. Robbins, *Power Electronics*, 3rd ed., John Wiley & Sons Inc., Replika Press Pvt. Ltd., India, 2006.
7. James W. Dally, William F. Riley and Kenneth G. McConnell, *Instrumentation for Engineering Measurement*, 2nd ed., John Wiley & Sons Inc., Replika Press Pvt. Ltd., India, 2006, Chapter 6, pp. 162-205.
8. Thomas C. Hayes and Paul Horowitz, *Student manual for The Art of Electronics*, Cambridge University Press, Gopsons Papers Limited, India, 2002, Chapter 2, pp. 82-140.
9. Website, <http://www.allaboutcircuits.com/vol2/chpt12/6.html>, accessed on 4/06/2008.
10. Data sheet # DS39582B for PIC16F87XA 28/40 pin 8-bit CMOS flash Microcontrollers, Microchip Technology Inc., U.S.A, 2003.
11. Data sheet # GT50J101 for 600V, 50A IGBT, Toshiba, Japan.
12. Data sheet # PC817, Photocoupler, Sharp, Japan.
13. Datasheet # UTC D313, NPN Switching Transistor, ynfddwUnisonic Technologies Co., Ltd.

## Appendix 1 - MATLAB program used to estimate PID parameters of synchronizer

```
% Estimate values for PID [K(s+a)*(s+b)/s] of
present synchronizer in the Governor and Turbine
model
j=250;
Ww=76.18;
C=0.8;
t=0:1:600;
Tg=0.3;
Th=3.0;
a=0.2333;
Kt=8000;
for K=0.2:-0.01:0.01; %starts the inner loop to vary
the a values
for b=0.6:-0.001:0.001; %starts the inner loop to
vary the a values
num1=[K (a+b)*K a*b*K];
den1=[0 1 0];
tf1=tf(num1,den1);
num2=[0 0 1];
den2=[0 Tg 1];
tf2=tf(num2,den2);
num3=[0 0 1];
den3=[0 Th 1];
tf3=tf(num3,den3);
num4=[0 0 1];
den4=[0 Ww*J Ww*C];
tf4=tf(num4,den4);
tf5=tf1*tf2*tf3*tf4*Kt;
sys=feedback(tf5,1);
y=step(sys,t);
m=max(y);
n=min(y);
if m<1.35 & m>0.99;
break; % breaks the inner loop
end
if m<1.35 & m>0.99;
break; % breaks the inner loop
end
r1=1;while y(r1)<0.1,r1=r1+1;end;
r2=1;while y(r2)<0.9,r2=r2+1;end;
Rise_time=(r2-r1)*1
s=601; while y(s)>0.98 & y(s)<1.02; s=s-1;end;
settling_time=(s-1)*1
max_overshoot=m-1
plot(t,y);
grid;
title('Estimated Unit-Step response of present
Synchronizer')
xlabel('t Sec')
ylabel('Amplitude')
aa=num2str(a); %string value of 'a' to be printed on
the plot
bb=num2str(b); %string value of 'b' to be printed
on the plot
kk=num2str(K); %string value of 'K' to be printed
on the plot
mm=num2str(max_overshoot); %string value of
max_overshoot to be printed on the plot
rr=num2str(Rise_time); %string value of Rise_time
to be printed on the plot
ss=num2str(settling_time); %string value of
settling_time to be printed on the plot
text(110,0.7,'a='),text(150,0.7,aa)
text(110,0.5,'b='),text(150,0.5,bb)
text(110,0.3,'K='),text(150,0.3,kk)
text(110,0.25,'Max overshoot='),text(250,0.25,mm)
text(360,0.1,'Rise time='),text(450,0.1,rr)
text(110,0.1,'Settling time='),text(250,0.1,ss)
```





```
sol=[sys]
sol=[K;b];
a=a
b=b
K=K
TF_of_Present=tf1*tf2*tf3*Kt
```

### Results obtained from Matlab Program

```
Rise_time = 35
settling_time = 330
max_overshoot = 0.3475
```

Total system Transfer function:

$$\frac{\omega_r(s)}{\omega(s)} = \frac{1600 s^2 + 453.3 s + 18.66}{1.714e004 s^4 + 6.29e004 s^3 + 2.085e004 s^2 + 514.2 s + 18.66}$$

```
a = 0.2333
b = 0.0500
K = 0.2000
```

Transfer function of Synchronizer, governor and Turbine system:

$$\frac{1600 s^2 + 453.3 s + 18.66}{0.9 s^3 + 3.3 s^2 + s}$$

### Appendix 2 - Program in MATLAB to estimate PI parameters of proposed controller

% Estimate parameters for new  $[K(s+d)/s]$  controller (PI) when  $Pe(s)$  is connected

```
J=250;
Wo=76.18;
C=0.8;
t=0:1:90;
Tg=0.3;
Th=3;
K=0.2;
a=0.2333;
b=0.05;
Kt=8000;
Ke=6000;
for Kp=0.9:-0.1:0.1; %starts the inner loop to vary the 'Kp' values
for d=0.1:-0.001:0.001; %starts the outer loop to vary the 'd' values
num1=[K (a+b)*K a*b*K];
den1=[0 1 0];
tf1=tf(num1,den1);
num2=[0 0 1];
den2=[0 Tg 1];
tf2=tf(num2,den2);
num3=[0 0 1];
den3=[0 Th 1];
tf3=tf(num3,den3);
tf4=tf1*tf2*tf3*Kt;
%tf of the new PI controller
num4=[0 Kp Kp*d];
den4=[0 1 0];
tf5=tf(num4,den4);
tf6=tf5*Ke;
Gc=parallel(tf4,tf6);
num5=[0 0 1];
den5=[0 Wo*J Wo*C];
Gp=tf(num5,den5);
sys=feedback(Gc*Gp,1);
y=step(sys,t);
m=max(y);
```

```
if m<1.1 & m>0.99;
break; % breaks the inner loop
end
end
if m<1.1 & m>0.99;
break; % breaks the inner loop
end
end
r1=1;while y(r1)<0.1,r1=r1+1;end;
r2=1;while y(r2)<0.9,r2=r2+1;end;
Rise_time=(r2-r1)*1
s=91; while y(s)>0.98 & y(s)<1.02; s=s-1;end;
Settling_time=(s-1)*1
max_overshoot=m-1
plot(t,y);
grid;
title('Unit-Step response when Pe(s) is connected')
xlabel('t Sec')
ylabel('Amplitude')
kk=num2str(Kp); %string value of Kp to be printed on the plot
dd=num2str(d); %string value of d to be printed on the plot
mm=num2str(max_overshoot); %string value of max_overshoot to be printed on the plot
ss=num2str(Settling_time); %string value of Settling_time to be printed on the plot
rr=num2str(Rise_time); %string value of Rise_time to be printed on the plot
text(21,0.9,'Kp='),text(26,0.9,kk)
text(21,0.7,'d ='),text(26,0.7,dd)
text(21,0.5,'Max overshoot='),text(42,0.5,mm)
text(21,0.3,'Settling time='),text(42,0.3,ss)
text(21,0.1,'Rise time='),text(42,0.1,rr)
sol=[sys]
sol=[Kp;d]
Kp=Kp
d=d
tf4
TF_of_PI=tf(num4,den4)
PIc2d=c2d(tf5,0.04,'zoh')
```

### Results obtained from Matlab Program

```
Rise_time = 5
settling_time = 42
max_overshoot = 0.0994
Transfer function:
```

$$\frac{\omega_r(s)}{\omega(s)} = \frac{4860 s^4 + 1.969e004 s^3 + 6851 s^2 + 321.1 s}{1.714e004 s^5 + 6.776e004 s^4 + 3.894e004 s^3 + 6912 s^2 + 321.1 s}$$

```
Kp = 0.9000
D = 0.0560
```

Transfer function of PI Controller:

$$\frac{0.9 s + 0.0504}{s}$$

