General Disclaimer

One or more of the Following Statements may affect this Document

MICROPROCESSOR CONTROL OF A WIND TURBINE GENERATOR

Arthur J. Gnecco and Gary T. Whitehead
National Aeronautics and Space Administration
Lewis Research Center
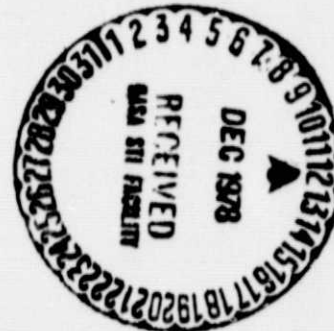
MICROPROCESSOR CONTROL OF
A WIND TURBINE GENERATOR

Arthur J. Gnecco and Gary T. Whitehead
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

MICROPROCESSOR CONTROL OF A WIND TURBINE GENERATOR

by Arthur J. Gnecco and Gary T. Whitehead

NASA-Lewis Research Center
Cleveland, Ohio

ABSTRACT

This paper describes a microprocessor based system used to control the un-attended operation of a wind turbine generator. The turbine and its microcomputer system are fully described with special emphasis on the wide variety of tasks performed by the microprocessor for the safe and efficient operation of the turbine. The flexibility, cost and reliability of the microprocessor were major factors in its selection.

E-9818

INTRODUCTION

As the United States starts to reexamine the feasibility of extracting power from the wind, it is found that this ancient art can benefit from the newest of technologies, the microprocessor. This paper describes a 100-kilowatt wind turbine generator designed by the National Aeronautics and Space Administration (NASA) Lewis Research Center for the Department of Energy (DOE). This wind turbine is located at the NASA Plum Brook Station in Sandusky, Ohio, and it is the prototype for four new 200-kW wind turbines to be operated by Utilities in the United States.

The first two of these are currently in place and operating in Clayton, N. M., and on Culebra Island, Puerto Rico. The other two machines will be installed on Block Island, R. I., and on the island of Ohau in Hawaii.

DESCRIPTION

a. General

The Plum Brook wind turbine is shown on figure 1. The turbine operates as a downwind machine, i. e. the wind passes through the tower before reaching the blades. This arrangement causes the blades to deflect away from the tower thereby reducing the amount of nacelle overhang required to prevent the blades from striking the tower. The turbine blades are 125 in diameter, operate at 40 rpm and produce 100 kW of electrical power in an 18 mph wind. A 45:1 gear train increases the rotor

speed to 1800 rpm, the operating speed of the synchronous alternator. When the wind turbine is up to operating speed, its alternator is synchronized and connected to the Ohio Edison distribution system. The power generated by the wind turbine supplements this commercial power system.

The 200 kW wind turbines are the same physical size as the Plum Brook machine but have larger alternators capable of producing 200 kW in a 22.5 mph wind.

b. Yaw (Azimuth) Control

A separate yaw control system, independent of the microprocessor, keeps the nacelle pointing into the wind. At low wind velocities wind direction is extremely variable so no attempt is made to have the nacelle track the wind below 6 mph.

The yaw controller is basically an analog voltage comparator. The comparator detects the difference between the actual nacelle direction and the actual wind direction and drives the nacelle yaw motors so as to make this difference zero. There is a dead band of $\pm 25^{\circ}$ and the wind direction signal is filtered so that the system will not react to wind gusts. The yaw motors are capable of driving the nacelle through 360° in 6 minutes. The circuitry is designed so that the nacelle always takes the shortest path when aligning itself with the wind.

c. Blade Pitch Angle Control

The amount of power extracted from the wind is regulated by varying the blade pitch angle. The system for positioning the blades is shown in figure 2.

The controller can be operated in any one of three modes. The controller mode is selected by the microprocessor.

In the POSITION mode, the pitch controller functions as a closed loop position control system driving the pitch angle until it corresponds to the position command (POSCMD) set point generated in the microprocessor.

In the AUTOMATIC/RPM mode, the pitch controller acts as a closed loop speed control system, adjusting the pitch angle until the rotational speed of the blade corresponds to the speed command (RPMCMD) set point generated in the microprocessor.

In the AUTOMATIC/KW mode, the pitch controller acts as a closed loop power control system which positions the blade pitch angle, as required, until the power output of the wind turbine matches the KW set point generated in the microprocessor.

The blade pitch angle is controlled by driving an electrohydraulic servo valve which controls the flow of hydraulic fluid to the pitch angle actuator. A hydraulic pump unit supplies the high pressure (1500 psig) hydraulic fluid. A signal from a rotary differential transformer (ZT), representing the actual pitch angle, is fed back to the input of the position amplifier where it is compared with the position command signal. The position command signal (POSCMD) comes from the microprocessor when the controller is in the POSITION mode or from the control amplifier when the controller is in either the AUTOMATIC/RPM or the AUTOMATIC/KW control mode.

A fail-safe solenoid valve in the hydraulic supply line and another in the return line are provided to bypass the servo valve should the controller, microprocessor or the servo valve itself fail. De-energizing the fail-safe solenoids will block the servo valve and drive the pitch actuator to the "feather" (no power) position.

For AUTOMATIC/RPM control, a RPM sensor detects the actual rotary speed of the blades and feeds a signal back to the input of the control amplifier. This feedback signal is compared against the speed command set point (RPMCMD) is generated by the microprocessor and an error signal is calculated. The controller amplifier uses the error signal to compute the position command signal that goes to the position amplifier. When the controller is in the speed control mode, the RPM compensation networks are switched across to the control amplifier.

For AUTOMATIC/KW control, a Power (kW) Sensor feeds a signal back to the input of the control amplifier where it is summed with the kW set point signal from the microprocessor. An error signal is derived and used to compute the position command signal for the closed loop position control system. When the controller is in the kW control mode, the kW compensation networks are automatically switched across the control amplifier.

d. Synchronization

An automatic synchronizer, independent of the microprocessor, is provided to connect the wind turbine to the power line smoothly. The microprocessor decides when to energize the synchronizer. The synchronizer compares the power line phase angle with the alternator phase angle and closes the connecting circuit breakers when the phase angles are matched and steady. As soon as synchronization is complete, a signal is returned to the microprocessor. The microprocessor now directs the pitch controller to switch from the speed control mode to the power control mode.

e. Safety

The safety system is a GO/NO-GO device that monitors critical parameters and triggers an EMERGENCY interrupt to shut down the wind turbine if limit set points are exceeded. The safety system is not dependent on microprocessor operation. When limits are exceeded, the safety system also de-energizes the fail-safe solenoid valves directly. The safety system tests for over-temperature, loss of hydraulic pressure, loss of hydraulic fluid, overspeed, excessive vibration and yaw error. As a further aid to safety there are fully independent redundant sensors located in the nacelle that can de-energize the fail-safe valves directly.

SOFTWARE

A simplified flow diagram of the program used to control the wind turbine is shown in figure 3.

a. Program Organization

The program is organized to perform the following seven major functions:

1. PROGRAM CONTROL pushbuttons are provided to enable the site operator or the remote dispatcher to start or stop the wind turbine or to initiate an emergency shutdown. Emergency shutdown first stops the wind turbine and then halts the microprocessor.

2. RESTART: This portion of the program tests wind conditions and monitors the control flags.

3. POSSTART: In POSSTART the program ramps the blade pitch angle from "feather" to "power" in order to get the rotor turning.

4. RPMSTART: In this phase, the microprocessor transfers the controller to AUTOMATIC/RPM control at 6 rpm and ramps to 40 rpm.

5. SYNCHRONIZE: The microprocessor synchronizes the wind turbine to the power line and transfers to load control with a 100 kW set point.

6. SHUTDOWN: The microprocessor shuts down the wind turbine by ramping the blade pitch angle back to "feather" on position control and breaking the connection to the grid at zero power.

7. OFF: The hydraulic pump unit is stopped, the fail-safe solenoid valves are de-energized and all control flags are re-initialized.

b. Program Description

The program is placed in operation by actuating the RESET pushbutton. The reset flag RSTFLAG is set (1), the halt flag HLFLAG is cleared (0) and since the blade pitch angle is at "feather," the microprocessor executes the SHUTDOWN: and OFF: portions of the program. The HFLAG test causes the program to go to RESTART: where the RSTFLAG test returns it to SHUTDOWN:. The program is now running and the interrupt pushbuttons are now usable.

Actuating the START pushbutton clears all flags including the RSTFLAG. This starts the program cycling through RESTART: where its first task is to reset the watchdog timer. This is an external timer, driven by the microprocessor, that must be continually reset or it will trigger the safety system and deenergize the fail-safe solenoid valves. The program is structured so as to always cycle through RESTART: hence the watchdog timer never times out unless the program hangs up or the microprocessor fails.

In RESTART: the emergency shutdown flag (EFLAG), the shutdown flag (SFLAG) and the dispatcher flag (DFLAG) are tested and if SFLAG = 1, BFLAG = 1 or DFLAG = 0, the wind turbine is shutdown. In this portion of the program a redundant wind velocity sensor, located on a nearby meteorological weather tower, is also tested. If the wind velocity as measured by this sensor is greater than 45 mph, the wind turbine is shutdown.

The program now tests the velocity of the wind to ascertain if conditions are such that the wind turbine can be started. A sensor mounted on top of the wind turbine measures the wind velocity. If the turbine is not running (RFLAG = 0), the wind must be between 13 to 30 mph for a startup cycle to begin. Once the wind turbine has started (RFLAG = 1), the wind must drop below 8 mph for 10 seconds or increase above 40 mph for 10 seconds before a shutdown cycle is initiated. This hysterisis and delay prevents excessive on/off cycling of the wind turbine when wind conditions are marginal.

If the wind velocity criteria for startup are met, the hydraulic pump is started and the hydraulic pump pressure is monitored until the 1500-psig operating pressure is reached. If operating pressure is not reached in 2 minutes, or if pressure is lost while the pump is running, an emergency shutdown is triggered. As soon as the hydraulic pressure reaches the operating range, the fail-safe solenoid valves are energized and the blade pitch angle actuator is released.

At this point the microprocessor is commanding the controller to be in the POSITION control mode and the position command, POSCMD, is zero (feather). The Controller mode test branches the program to POSSTART:.

In POSSTART: the program starts ramping the blade pitch angle away from "feather" toward "power." The ramp rate is controlled by an external timer so that it can be adjusted in the field without having to modify the software.

As the blade angle pitches toward "power" the blades begin to rotate and as soon as they reach 5 rpm, the controller is transferred to AUTOMATIC/RPM control. The controller mode test branches to the RPMSTART: phase of the program.

In the RPMSTART: portion of the program, the rpm command, RPMCMD, set point is slowly ramped from 6 rpm to 40 rpm, synchronous speed. The rate of the set point ramp is controlled by an external timer that can be field adjusted.

During RPM control, a RPM error detector, located in the controller, is monitored by the microprocessor to assure that the actual rotational speed of the wind turbine is tracking the RPMCMD signal. Excessive RPM error will set the stop flag (SFLAG) and cause a shutdown followed by another startup attempt. Repeated RPM errors will clear the dispatcher flag (DFLAG) which will cause a shutdown. This shutdown will not restart until the dispatcher actuates the SUPERVISOR START pushbutton to set the DFLAG.

As soon as the rotor achieves synchronous speed (40 rpm), synchronization is attempted.

In the SYNCHRONIZE: portion of the program, the microprocessor energizes the independent synchronizer and sets a 1 minute timer. Synchronization should occur within 1 minute if everything is working properly. If synchronization is not achieved, the SFLAG is set and the wind turbine is shutdown to try again. After three synchronization attempts the DFLAG is cleared and the wind turbine is shutdown until released by the dispatcher.

If synchronization is successful the microprocessor transfers the controller to the AUTOMATIC/KW mode. To assure that the synchronization attempt is successful, the microprocessor tests that the network circuit breakers have closed and have remained closed for 2 seconds. After synchronization is complete, if the network circuit breakers open, an emergency shutdown is initiated.

When the pitch controller is in the automatic power control (AUTOMATIC/KW) mode, the KW compensation networks are connected to the control amplifier. In the power control mode, the set point to the controller is 100 kW. If the wind velocity is low so that the wind turbine can not produce 100 kW, the controller will drive the blade pitch angle to full "power" and produce as much power as available. As soon as the wind starts exceeding 18 mph, the control system will start to move the blade pitch angle back toward "feather" to maintain an output of 100 kW.

The program continues to cycle through RESTART. where it monitors the wind and checks the shutdown flags. If the wind velocity drops off or becomes excessively high, the program branches to the SHUTDOWN: portion of the program.

In SHUTDOWN: the microprocessor sets the SFLAG, transfers the controller to the POSITION control mode and starts to ramp the blade pitch angle from a "power" position toward "feather." As soon as the power output of the wind turbine drops to zero, the network circuit breaker is opened to remove the wind turbine from the power line. As soon as the blade pitch angle reaches "feather," the program branches to OFF:.

In OFF: the microprocessor turns off the hydraulic pump unit, de-energizes the fail-safe solenoid valves and clears all flags except DFLAG and HFLAG. If a HFLAG is detected, program execution stops. After a HALT, the program has to be restarted by an operator at the site. If there is no HFLAG, the program branches back to RESTART: where wind velocities are monitored to determine when the wind turbine should start up again.

Whenever the program detects an emergency flag, EFLAG or branches to EMERGENCY:, the blade pitch angle is driven to "feather" directly by de-energizing the fail-safe solenoid valves. In EMERGENCY: the halt flag, HFLAG, is set so that the program will HALT.

HARDWARE

The microprocessor is configured for extensive communication with the outside world. The system consists of the following equipment:

(a) 16 A/D Converters, 12 bits, with a ± 10 volt dc input for sensing the following parameters - blade rpm, wind velocity (nacelle), controller tracking signal, power output, wind velocity (meteorological tower) and spares.

(b) 12 Discrete 120-V, 60-Hz inputs for the following parameters – hydraulic parameters - hydraulic pressure, RPM error, field circuit breaker, line circuit breaker, power station circuit breaker, supervisory control inputs and spares.

(c) 8 Discrete 120-V, 60-Hz, 2-amp outputs for operating the following equipment - hydraulic pump unit, fail-safe solenoid valves, field contractor, synchronizer and spares.

(d) 8 N.O. Relay Outputs, for controller mode control.

(e) 6 D/A Converters, ± 10 volts, for the following output signals - position command, rpm set point, load set point and spares.

(f) 2 BCD indicators used to display the last interrupt or the cause of the last shutdown.

(g) 8 Interrupts for start/stop control of the program.

(h) 256 Bytes of RAM.

(i) 2K Bytes of EPROM.

(j) 1 TTY Interface.

(k) 8 Timers; externally adjustable from 0 to 60 seconds for establishing set point ramp rates.

(l) 1 CPU, Intel 8080, 8 Bit parallel processor.

All system components are Control Logic M Series Modules except for timers and relay cards.

PROGRAM DEVELOPMENT

a. Software Support

The software programming of the microprocessor was simplified by the use of the high level Intel Programming Language for Microprocessors, PL/M. A PL/M compiler is available on the Lewis Research Center - IBM 360 central computer. The compiled program can be stored on either paper tape or EPROM. During the early phase of software debugging, paper tape was used to load the program in RAM. The program could then be easily modified by using the Control Logic Inc., Octal Debugging Technique (ODT) program to make changes via a teletype terminal. When the program was completely debugged, it was transferred to EPROM for actual operation of the wind turbine.

The PL/M compiler supplies a printout of the program in both assembly language and machine language. The printout is keyed to the original PL/M statements. This information is extremely useful for making small program changes.

b. I/O Simulator

Another aid to program development and checkout was the use of a hardware I/O simulator. A special simulator chassis was designed to emulate every input required by the microprocessor and display every output generated by the program. The simulator contained eight 10 volt potentiometers for supplying the analog signals to the A/D converters, 8 function switches for applying 120 V, 60 Hz to the discrete inputs, 8 lights for displaying the discrete outputs, 8 lights for displaying the relay outputs and four digital voltmeters for displaying the output of the D/A converters.

OPERATIONAL EXPERIENCE

a. Checkout

Early program debugging was performed with the microprocessor connected to the I/O Simulator. The simulator allowed exercising every portion of the program.

When the microprocessor was installed at the wind turbine site, the simulator was especially useful for testing those conditions that are normally uncontrollable, for example, low or high wind conditions could be simulated and the wind turbine forced to shutdown even when the wind velocity was in range. Using this technique all the wind turbine functions could be tested. The simulator was a great help in placing the wind turbine in operation.

Most of the early circuit problems were noise related. Some of the communication lines to the microprocessor are quite long and are in close proximity to power lines that carry a large current. The microprocessor responded too quickly to false signal caused by pickup or contact bounce. Most of the problems were alleviated by judicious placement of filters or by programming techniques that allowed transients to settle before the microprocessor decides on the response.

b. Flexibility

The flexibility of the microprocessor in developing the operational strategy for controlling a wind turbine was perhaps the most significant advantage of using a microprocessor. In the early design phase of the project the wind turbine was operated in many different modes: manual control, without a microprocessor; as a speed controlled machine feeding a load bank; and finally as a synchronous machine connected to both large and small power networks. By simply modifying the software it was possible to accommodate all these operating modes. The original design concept of providing a system heavy in I/O equipment proved valid. All the information that the microprocessor might need in order to control a wind turbine was wired to the microprocessor. Every output that an operator might have to manipulate was also connected to the microprocessor. The wind turbine could now be controlled in any way desired by simply reprogramming. If hard wired analog circuitry had been used every operating mode would have required different equipment and every control strategy change would have required circuit modifications, system rewiring and re-testing.

The following example shows how the flexibility of the microprocessor, helped solve problems that developed in the field. The original concept was to start the

wind turbine at a wind velocity of 13 mph and stop it at 8 mph. It was found by experience that at times the wind velocity tended to oscillate continually over this range causing excessive start/stop cycles of the wind turbine. To solve the problem various filters were applied to the wind velocity signal. It was not possible to find filters with time constants long enough to prevent excessive start/stop cycles and short enough to be sufficiently responsive for control. The solution was to program the microprocessor to test if the high or low winds has persisted for 10 seconds before deciding to shutdown. A relatively lightly-filtered signal was used for control. This approach was easily implemented and proved successful, resulting in improved performance.

c. Reliability

The microprocessor was found to be extremely reliable and consistent. Once a program change was made and its performance tested, the microprocessor could be depended upon to perform in exactly the same manner repeatedly. The machine language generated by PL/M always performed the job as expected, without any surprises. The logistics of writing, compiling, entering, debugging and running programs was not burdensome.

d. Program Modifications

The design of the program made changes easy to implement. The program was constructed as a continuous loop, always cycling back to RESTART. The program cycles asynchronously, that is, at its own speed without being keyed to a clock. As a result of this, software tasks can be lengthened or shortened without interferring with the scheduled start time of other tasks. It was therefore easy to add new tasks by simply inserting them at the appropriate point in the program. All other jobs would just move back automatically and the programmer did not have to concern himself with any timing problem. The longest path task took less than 6 milliseconds, far shorter than any critical time response required by the wind turbine.

e. Cost

The cost of the microprocessor system in relation to the entire project was insignificant. An equivalent hardwired analog circuit containing all the functions required would have been much more expensive and implementing changes would have been time consuming.

FUTURE APPLICATIONS

Since the microprocessor has proved so reliable, flexible and inexpensive it is now being considered to take over the duties of other analog controllers on the site. The functions of the yaw controller can be easily incorporated into the microprocessor and will be implemented in the near future.

The functions of the blade pitch angle controller could also be incorporated but since controller response is critical in this system, greater care would have to be exercised to assure that the pitch control task maintained the highest priority. Another approach would be a second microprocessor dedicated to pitch angle control. The tradeoffs are now being weighed.

Currently, larger (1800 kW) wind turbines are in design and the approach being used is similar to these smaller machines. The concept of using a computer for controlling all the startup and shutdown functions and a separate analog blade pitch angle controller has been retained. The yaw control functions are being assigned to the computer.

These large machines are presently designed around minicomputers rather than microprocessors. Minicomputers were selected because of their extensive software development support systems. These software development systems were considered necessary for programming and testing at wind turbines in remote locations. It is felt that when designs are finalized and proven, the mass produced versions of these wind turbines will turn to microprocessors because of their lower cost. Once the programs are developed and proven there will be no further need for the additional software support facilities on site.

CONCLUSIONS

The microprocessor has proved itself to be dependable, easy to program, responsive to the changing needs of the project, inexpensive and equal to the task it had to perform. If the job had to be done again the same approach would be used.
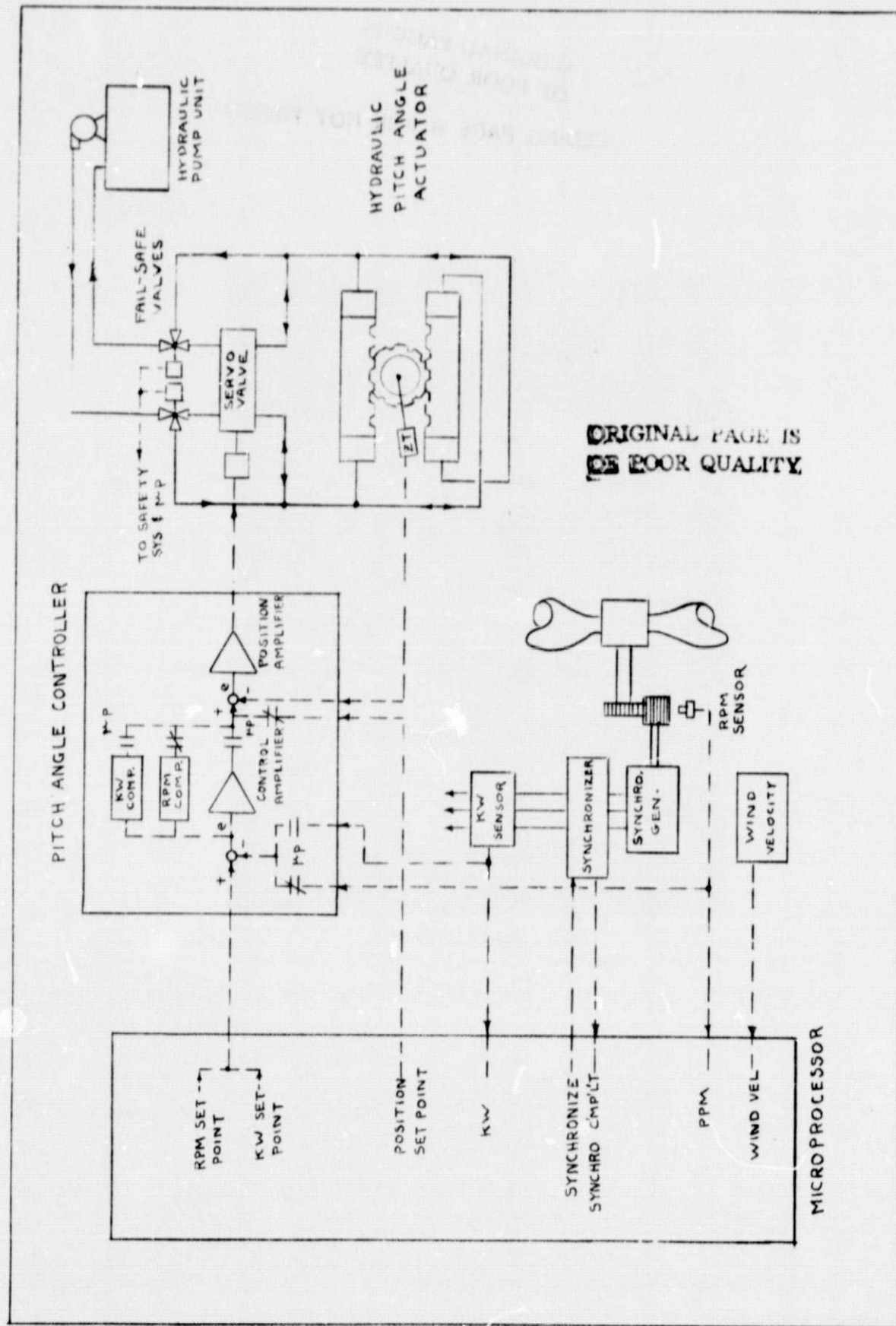
ACKNOWLEDGEMENT

Figure 1. - 100 KW Wind Turbine.

Figure 2. - Blade Pitch Angle Control System.

Figure 3. - Microprocessor Flow Diagram.