WATSON EDWARDS (Pty) Ltd

Report to the

WATER RESEARCH COMMISSION

on

BALANCING TANK CONTROL APPLICATION

by

A J Mather

WRC Report No. 621/1/97
ISBN 1 86845 353 7

NOTE

The computer program and manual,
as well as answers to any queries
regarding its application, are obtainable
from the author stationed at the Sandton Office
of Watson Edwards (Pty) Ltd.

Telephone (011) 803 7884
Fax (011) 803 7430
E-mail: wevsjhb@wevs.co.za

ACKNOWLEDGEMENTS

The research in this report emanated from a project funded by the Water Research Commission and entitled :

"A Balancing Tank Control Application for Wastewater Treatment Plants".

The financing of the project by the Water Research Commission and Watson Edwards (Pty) Ltd is gratefully acknowledged.

This project was only possible with the co-operation of many individuals and institutions. The author therefore wishes to record his sincere thanks to the following :

Officials of the Wastewater Department of the Greater Johannesburg Transitional Metropolitan Council for their assistance, advice and use of their facilities.

Members of the Control Engineering Department of the University of the Witwatersrand for their advice and use of their facilities.

Colleagues at Watson Edwards (Pty) Ltd for their assistance and support and especially to Shane Smalle who was responsible for some of the software development.

Staff at the Goudkoppies and Northern Wastewater Treatment Works.

Table of Contents

TITLE PAGE 1-1

ACKNOWLEDGEMENTS 1-2

TABLE OF CONTENTS 1-3

LIST OF FIGURES 1-4

EXECUTIVE SUMMARY 1-5
PROBLEM STATEMENT
RESEARCH APPROACH
PREVIOUS RESEARCH INTO BALANCING TANK CONTROL
PROPOSED SOLUTION
FUZZY CONTROLLERS
FUZZY LOGIC BALANCING TANK CONTROLLER
RESEARCH RESULTS
IMPLEMENTED SOLUTION
SYSTEM CONFIGURATION
DESCRIPTION OF THE MODIFIED CONTROL ALGORITHM
OPERATOR INTERFACE
IMPLEMENTATION RESULTS
CONCLUSION

INTRODUCTION 1
PROBLEM STATEMENT 1
RESEARCH APPROACH 1

LITERATURE SURVEY 3
LITERATURE SURVEY OBJECTIVES 3
PREVIOUS RESEARCH INTO BALANCING TANK CONTROL 3

PROPOSED SOLUTION 6
SOLUTION OBJECTIVES 6
SYSTEM CONFIGURATION 6
AN INTRODUCTION TO FUZZY LOGIC 6
FUZZY SETS 7
LINGUISTIC HEDGES 7
FUZZY CONTROLLERS 7
FUZZY LOGIC BALANCING TANK CONTROLLER 8
STEP 1 - DEFINITION OF ANTECEDENTS (INPUTS) 9
STEP 2 - DEFINITION OF FUZZY SETS 9
STEP 3 - DEFINITION OF CONSEQUENT - OUTPUT 9
STEP 4 - ESTABLISH INFERENCE RULES 9
STEP 5 - SELECT DEFUZZIFICATION METHOD 10
STEP 6 - RUN SIMULATION 10
RESEARCH RESULTS 10

IMPLEMENTED SOLUTION 11
SYSTEM CONFIGURATION 11
PREVIOUS SHORTCOMINGS 11
DESCRIPTION OF THE MODIFIED CONTROL ALGORITHM 11
OPERATOR INTERFACE 12
IMPLEMENTATION RESULTS 13

CONCLUSION 14

REFERENCES 15

APPENDIX A 16

APPENDIX B 17

LIST OF FIGURES

Fig 1 : Simplified Process and Instrumentation diagram of the Balancing Tank

Fig 2 : System Configuration diagram of the control system

Fig 3 : The structure of a fuzzy controller

Fig 4 : Modified balancing tank control algorithm configuration screen

Fig 5 : Modified balancing tank control algorithm graphical Mimic status display screen

Fig 6 : Modified balancing tank control algorithm outlet valve control screen

Fig 7 : Modified balancing tank control algorithm results at Northern Works.

Fig 8 : Modified balancing tank control algorithm results at Northern Works.

Fig 9 : Modified balancing tank control algorithm results at Northern Works.

Fig 10 : Comparative results of mathematical model controller vs Neural network controller

Fig 11 : Comparative results of human operator vs Neural Network controller

Fig 12 : Graphic results of the fuzzy logic controller

FINAL REPORT TO THE WATER RESEARCH COMMISSION
ON A RESEARCH PROJECT INTO THE DEVELOPMENT
OF A BALANCING TANK CONTROL ALGORITHM

EXECUTIVE SUMMARY

PROBLEM STATEMENT

The inflow of wastewater or sewage into a treatment works is characterised by large variations in both flow and concentration (of pollutants). In the absence of any load equalisation upstream of the bioreactors, the oxygen demand in the bioreactors would undergo large fluctuations. These large fluctuations would make control of the oxygen supply to the reactors extremely difficult. As a result of these difficulties, it has become common practice to construct a balancing or equalisation tank upstream of the bioreactors. It is necessary to optimise the use of the tank so that it provides a fairly constant outflow to the bioreactor whilst at the same time not running dry or overflowing. The control function is non-linear and therefore a simple 3 term PID loop controller (Proportional, Integral, derivative) cannot be used in isolation. There is also an inverse relationship between volumetric inflow and pollutant load as generally the load decreases as the volumetric flow into the Works increases. The nature of the problem necessitates some form of "intelligent" control strategy which requires the incorporation of extensive computational facilities.

RESEARCH APPROACH

The aim of the research project is to develop a controller which under the cyclic inputs of flow and load, determines the appropriate outflow rate at any time such that the flow and/or load will be optimally equalised. The controller has to operate within the following constraints :

* The equalisation tank volume available
* The tank should neither overflow nor be drawn down to a level below the minimum
* The inverse relationship between volumetric flow and pollutant load
* Daily and seasonal variations in the inflow pattern

The intention was to establish whether a suitable controller could be developed utilising "fuzzy logic" to modify the set point for a standard PID controller.

PREVIOUS RESEARCH INTO BALANCING TANK CONTROL

Early research by MacInnes, Middleton and Adomowski (1978) recognised that a major factor contributing to a reduction in the efficiency of any equalisation facility is the limitations inherent in manual operation. To overcome this problem these authors have proposed a control strategy based on a stochastic approach, for determining the required tank outflow rate at intervals over the day. Their approach makes use of a mathematical simulation of the real-time behaviour of an in-line equalisation process. The process is viewed as a dynamic system producing stochastic outflows from stochastic inflows.

The flow rate time series is applied at regular intervals to forecast the mean flow rate for the subsequent 24 hours. This forecast then becomes the tank outflow rate for the subsequent 3 hour period.

MacInnes, et al, encountered operational difficulties with the tank emptying and overflowing. This was because the procedure did not take into account the current tank level.

Later research by Dold, Buhr and Marais (1982) adopted the approach of predicting influent flow rate and concentration patterns over the ensuing 24 hour period, then computing the outflow profile (for the ensuing 24 hour cycle) that gives the least error in terms of some flow and load optimisation criteria.

The resulting equalisation algorithm given below was then incorporated into a control strategy for real time, continuous operation. This strategy involved the prediction of expected influent patterns, based on historical data. It also incorporated the differences between actual and historical inflow rates so as to update the historical data.

The equalisation algorithm was :

$E_t = E_e + E_{lm} + E_s$

Where : E_t is the total error used as the objective function in the optimisation procedure.

: $E_e = \propto E_f + (1 - \propto) E_{lm}$

: E_{lm} is the penalty error for overflowing or running dry of the tank.

: E_s is the penalty error to constrain the rate of change of the tank outflow rate.

This algorithm was successfully implemented, however it has fallen into disuse due to problems with the interface hardware, and difficulties experienced in "tuning" the adjustable parameters. Results obtained with this algorithm are shown in Appendix A.

The latest research undertaken at Rand Afrikaans University by Shaw and Midlane, and Shaw and Mather, utilised the pattern recognition properties of neural networks to predict output patterns based on recognised inflow patterns. Results obtained were favourable, however "training" of the neural networks was complex. Further developments are likely in this area as more advanced software becomes available for training the networks. Results obtained during simulation tests are shown in Appendix A.

PROPOSED SOLUTION - USING FUZZY LOGIC

The objective of this research project was to establish whether a suitable balancing tank controller could be developed utilising "fuzzy logic" to modify the setpoint for a standard PID controller.

The "Omron fuzzy inference" software package was to be used to design and test the controller. Once developed, the fuzzy logic controller software would be permanently installed on an Omron fuzzy inference board in a personal computer. A standard supervisory control and data acquisition (SCADA) software package was used to provide a data communications driver between the "fuzzy controller" and the field devices. It also provided an operator interface.

FUZZY CONTROLLERS

In a fuzzy controller, linguistic variables are used in fuzzy associations, or rules which connect a linguistically defined condition (or input) with a linguistically defined consequence (or output). The form of such a rule is :

IF < Condition > Then < Consequence

There are a number of such rules which each produce a response.

From the combined fuzzy outputs contributed by each rule, a single value is chosen by "defuzzifying" the combined output. This is done by taking the centroid of the fuzzy responses.

FUZZY LOGIC BALANCING TANK CONTROLLER

The objectives of our controller were as follows :

- Utilise on-line inputs of inlet flow, tank level and outlet flow
- Utilise historical inflow patterns
- Calculate an outlet flow setpoint which can be used by a standard PID loop controller to control the outlet flow control valve
- The rate of change of the outlet flow had to be minimised
- The balancing tank should not be allowed to overflow or run dry

RESEARCH RESULTS

The simulation test results were very disappointing. These are shown graphically in Appendix A. The control achieved using our fuzzy logic controller could not match that achieved using the mathematical models used by Dold, et al, or those achieved by Shaw using Neural networks.

The major failure with this controller was that it could not achieve the objective of minimising the rate of change of tank outflow and it was not adaptive or self correcting.

As a result of these poor simulation results, the author decided to abandon the use of fuzzy logic for the development of this controller and to rather develop an adaptive controller based on iterative mathematical calculations.

IMPLEMENTED SOLUTION

SYSTEM CONFIGURATION

The overall system configuration is the same as shown in Fig. 1. Physically the control system configuration for the modified controller is the same as for the fuzzy controller but with the following differences :

* The fuzzy logic inference board running the fuzzy logic software is no longer required.

* Instead the control algorithm runs as a PASCAL subroutine in the supervisory computer and is called by the SCADA software package. This routine is run every 30 minutes and the resulting tank outflow setpoint is passed via the SCADA system to the PID control loop in the PLC, which in turn controls the balancing tank outlet control valve.

DESCRIPTION OF THE MODIFIED CONTROL ALGORITHM

As mentioned previously the modified control algorithm was written as a Pascal subroutine which could be called by the plant SCADA system. Although it was written to interface with the Turbolink/Multilink version 8.1 SCADA system, it could easily be modified to run with any SCADA system.

The modified algorithm, like the Dold algorithm utilises historical inflow data to assist in predicting the required out-put. 48 flowrates taken at half hourly intervals are stored to give a typical 24 hour weekday inflow pattern. Another 48 flowrates are stored to give a typical 24 hour weekend inflow pattern.

The algorithm is "called" every half hour by the SCADA system and it goes through the following steps.

Step 1 : Check current day and time from SCADA system.

Step 2 : Check current tank level.

Step 3 : Using the current tank level and the anticipated inflow for the next 24 hours and the current average outflow, calculate the change in tank level.

Step 4 : If with the current average outflow you will exceed any of the preset tank level limits, then adjust the outflow depending on which limit is exceeded. The change in level is then re-calculated. This is an iterative process until the optimum outflow is obtained where no tank levels are exceeded for the 24 hour period. Because it starts at the current outflow, the rate of change of outflow is minimised.

Step 5 : Calculate difference between actual inflow and historical inflow. Add or subtract a percentage of this error to the outflow setpoint. This percentage is configurable from the SCADA.

Step 6 : If the actual level is approaching one of the limits, then the algorithm will run every 5 minutes in order to prevent the tank level exceeding the limits.

Step 7 : At midnight the days 48 actual inflow values for the day, are compared to the 48 predicted values. The predicted values are then updated by taking the average of 95% of the predicted values and 5% of the actual values. In this way any long term changes in the inflow pattern such as seasonal variations would be accounted for by the algorithm.

Installation and operating instructions for the running of the balancing tank algorithm are given in Appendix B :

OPERATOR INTERFACE

The Turbolink SCADA package is used to provide an operator interface to the modified balancing tank control algorithm.
Three graphical mimic control screens are available to the operator to configure the algorithm.

CONCLUSION

From the research conducted we can draw the following conclusions :

* Fuzzy Logic cannot be used to develop a balancing tank controller which is easy to set up and configure and which provides stable control. The main problem associated with the fuzzy logic controller which was developed, was the fact that it was not self adapting. In other words it would not be self correcting.

* It is possible to develop a controller based on iterative mathematical calculations, using historical data as a reference. The modified controller was developed to run as a Pascal subroutine, and was interfaced with a standard SCADA package which provided a user friendly operator interface. The modified controller has been successfully installed at 4 wastewater treatment works operated by the Greater Johannesburg Metropolitan Council.

* Future research work may focus on developing a controller based on neural networks which have proved to have excellent pattern recognition characteristics. Early work in this regard showed promising results, however the complexities associated with "training" the neural networks proved to be a limiting factor. Recent developments in neural network training software may however have paved the way for further work in this field.

1

INTRODUCTION

PROBLEM STATEMENT

The inflow of wastewater or sewage into a treatment works is characterised by large variations in both flow and concentration (of pollutants). In the absence of any load equalisation upstream of the bioreactors, the oxygen demand in the bioreactors would undergo large fluctuations. These large fluctuations would make control of the oxygen supply to the reactors extremely difficult. In plants utilising surface aeration where control is achieved by switching on and off of the aerators, control is not only difficult but would be harmful to the electrical and mechanical equipment and would lead to significant increases in electrical power consumption. As a result of these difficulties, it has become common practice to construct a balancing or equalisation tank upstream of the bioreactors. This tank is used as buffer tank to dampen out the effect on the bioreactors, of large changes in inflow to the plant. It is necessary to optimise the use of the tank so that it provides a fairly constant outflow to the bioreactor whilst at the same time not running dry or overflowing. The control function is non-linear and therefore a simple 3 term PID loop controller (Proportional, Integral, Derivative) cannot be used in isolation. There is also an inverse relationship between volumetric inflow and pollutant load as generally the load decreases as the volumetric flow into the Works increases. The nature of the problem necessitates some form of "intelligent" control strategy which requires the incorporation of extensive computational facilities.

RESEARCH APPROACH

The aim of the research project is to develop a controller which under the cyclic inputs of flow and load, determines the appropriate outflow rate at any time such that the flow and/or load will be optimally equalised. The controller has to operate within the following constraints :

* The equalisation tank volume available
* The tank should neither overflow nor be drawn down to a level below the minimum
* The inverse relationship between volumetric flow and pollutant load
* Daily and seasonal variations in the inflow pattern

The intention was to establish whether a suitable controller could be developed utilising "fuzzy logic" to modify the set point for a standard PID controller.

The Omron "Fuzzy" inference software package was to be used to design and test the controller. Once the development was completed, the fuzzy logic controller software would be installed on an Omron fuzzy inference board which would in turn be installed in a personal computer. A communications driver software package would then have to be developed to enable the PC based fuzzy logic controller to communicate with a Programmable Logic Controller (PLC) which would serve as the interface between the field devices and the Controller.

Historical inflow data from Goudkoppies Wastewater Treatment works was used for simulation testing of the Controller and on site tests were done at Northern Works and Goudkoppies Works.

2

LITERATURE SURVEY

LITERATURE SURVEY OBJECTIVES

The 2 primary objectives of the literature survey were to establish the following :

I) What previous research had been carried out into balancing tank control
ii) Whether "fuzzy logic" would be an appropriate form of control and whether it had been used previously for similar control applications.

PREVIOUS RESEARCH INTO BALANCING TANK CONTROL

Early research by MacInnes, Middleton and Adomowski (1978) recognised that a major factor contributing to a reduction in the efficiency of any equalisation facility is the limitations inherent in manual operation. To overcome this problem these authors have proposed a control strategy based on a stochastic approach, for determining the required tank outflow rate at intervals over the day. Their approach makes use of a mathematical simulation of the real-time behaviour of an in-line equalisation process. The process is viewed as a dynamic system producing stochastic outflows from stochastic inflows. The influent flow rate time series, Z (t), is assumed to be represented by a linearly additive model such that

$$Z(t) = Z_T(t) + Z_P(t) + Z_S(t)$$

Where

$Z_T(t)$ = deterministic trend component
$Z_P(t)$ = deterministic cyclic component
$Z_S(t)$ = stochastic component

The flow rate time series is applied at regular intervals to forecast the mean flow rate for the subsequent 24 hours. This forecast then becomes the tank outflow rate for the subsequent 3 hour period.

MacInnes, et al, encountered operational difficulties with the tank emptying and overflowing. This was because the procedure estimates a future flow at any time based solely on a background of statistical history of the influent flow and does not take into account the tank level at that point.

From their research one can conclude that any control algorithm must determine the required outflow rate based on a forecast of the influent flow pattern and the tank volume at the time of the forecast.

Later research by Dold, Buhr and Marais (1982) adopted the approach of predicting influent flow rate and concentration patterns over the ensuing 24 hour period, then computing the outflow profile (for the ensuing 24 hour cycle) that gives the least error in terms of some flow and load optimisation criterion. The optimal condition is identified by minimising an empirical error function that expresses the integrated daily deviation of both flow and load rates from their respective mean values. The relative importance of flow as against load equalisation was varied through applying a weighting factor, $\propto$, to the errors for flow. E_f, and load E_d, respectively.
This approach differed from that of MacInnes, Middleton and Adomowski in that the approach here was to accept a given tank volume and then to control the outflow rate to give the minimum deviation from the mean. This approach therefore makes allowance for variability in the daily cyclic influent pattern. Therefore even if the available volume is too small to allow complete equalisation, that volume is utilised optimally.
Dold, et al, added two penalty errors to their equalisation equation to cater for the following constraints :

I) That the tank level can at no time exceed specified upper and lower volume limits.
ii) "Spikiness" in the outflow profile could develop when the tank was near full or empty.

The resulting equalisation algorithm was as follows :

$E_t = E_e + E_{lm} + E_s$

Where : E_t is the total error used as the objective function in the optimisation procedure.

: $E_e = \propto E_f + (1 - \propto) E_{lm}$

: E_{lm} is the penalty error that increases rapidly as the tank hold up attains values outside of the specified limits.

: E_s is the penalty error to constrain the rate of change of the tank outflow rate.

This equalisation algorithm was then incorporated into a control strategy for real-time, continuous operation. This involved the prediction, at any point in time, of the expected influent patterns for the ensuing 24 hour cycle. The prediction is based primarily on historical inflow and concentration data, but also incorporates differences between actual and historical inflow rates for the period prior to the prediction.

For application of the Control Strategy, the day is divided into half-hour intervals. At the beginning of an interval, the expected influent patterns for the ensuing 24 hour cycle are set up and utilised by the equalisation algorithm to compute the optimal simulated tank outflow profile for the 24 hours ahead. The outflow value determined for the first interval in the 24 hour cycle is then applied as the actual output for the duration of that interval. By repeating this procedure at the start of each control interval, performance of the equalisation tank is continuously optimised.
An important aspect of the control strategy is that the algorithm differentiates between influent patterns for weekdays and weekend days and between summer days and winter days.

This algorithm was successfully implemented at Goudkoppies Wastewater Treatment Works, however, it has fallen into disuse due to problems with the interface hardware and difficulties experienced in "tuning" the adjustable parameters. Graphical results obtained with this algorithm are shown in Appendix A.

The latest research into equalisation tank control was undertaken at Rand Afrikaans University by Shaw and Midlane, and Shaw and Mather.
This research was aimed at using the pattern recognition properties of neural networks to recognise changes in inflow patterns and to predict the required outflow pattern.
Although the controller which was developed was never implemented on site, test results achieved during simulation were very promising and compared favourably with those achieved by the "Dold" algorithm.
The neural network controller had to be "trained" utilising representative data sets of typical inflow, outflow and level values. "Training" of the controller proved to be a relatively complex process. There have been recent developments which virtually automate the "training" of neural networks. There may therefore be merit in undertaking further research into the use of neural networks for balancing tank control.

Graphical results obtained during simulation tests with neural networks are shown in Appendix A.

3

PROPOSED SOLUTION

SOLUTION OBJECTIVES

The objective of this research project was to establish whether a suitable balancing tank controller could be developed utilising "fuzzy logic" to modify the setpoint for a standard PID controller.

SYSTEM CONFIGURATION

The "Omron Fuzzy inference" software package was to be used to design and test the controller. Once developed, the fuzzy logic controller software would be installed on an Omron fuzzy inference board which in turn would be installed in a personal computer. The "turbolink supervisory control and data acquisition" (SCADA) package was used to provide a data communications driver for communications between the PC based controller and a programmable logic controller which monitors and controls the field devices.

Field instrumentation and control devices consisted of :
I) an ultrasonic level transmitter for monitoring the tank level.
ii) An inlet flow meter.
iii) An outlet flow meter.
iv) An outlet control valve.

Fig 1 Shows a simplified process and instrumentation diagram of the balancing tank.

Fig 2 Shows a system configuration diagram of the control system.

AN INTRODUCTION TO FUZZY LOGIC

Classical control theory relies on the requirement that the plant to be controlled is capable of being described in a rigid mathematical form. Where the plant is a highly non-linear system, difficulties arise in designing an appropriate controller using linear system theory. Although a plant may be highly non-linear, the human operator still manages to control the system. It would therefore seem logical to design a controller that is based on the experience of the human operator. To obtain an accurate description of the control strategy of an operator, the following problems have to be overcome.

LEGEND:

FIT 01	– INLET FLOW METER	FC 03	– FLOW CONTROLLER (PLC – PID CONTROLLER)
LIT 02	– LEVEL TRANSMITTER	FCV 03	– FLOW CONTROL VALVE
FIT 03	– OUTLET FLOW METER	B.T CONTROLLER	– BALANCING TANK CONTROLLER

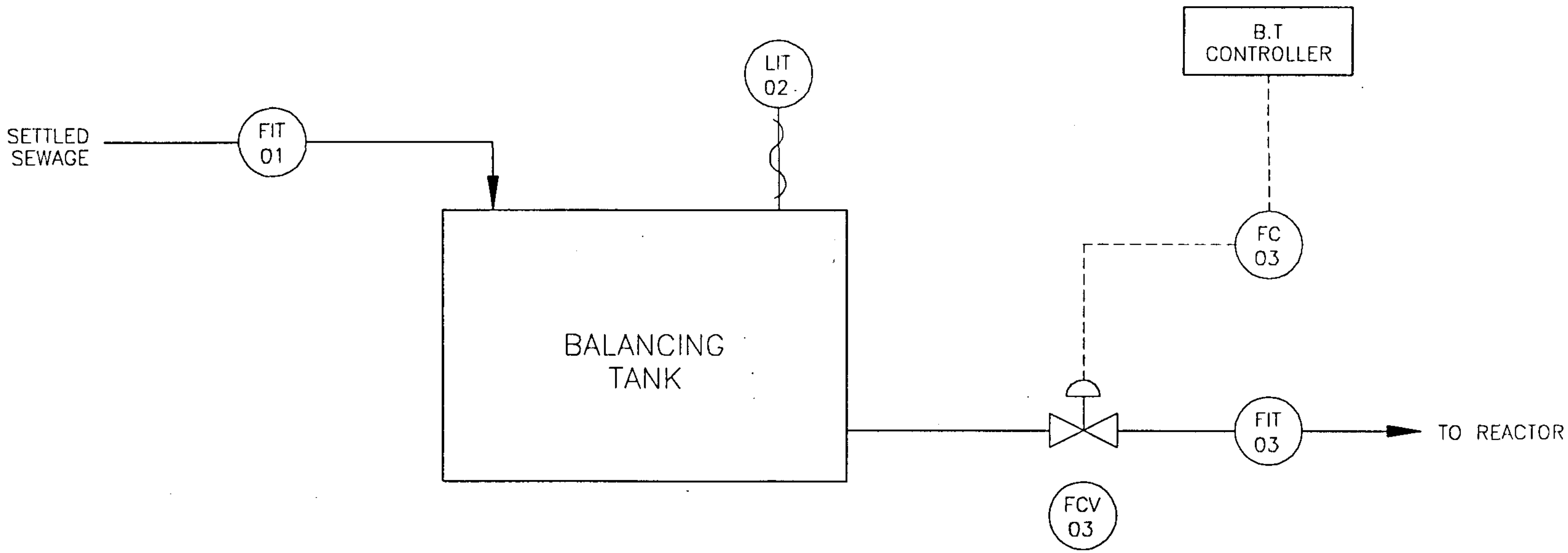


FIG. 1

SIMPLIFIED PROCESS AND INSTRUMENTATION
DIAGRAM OF THE BALANCING TANK

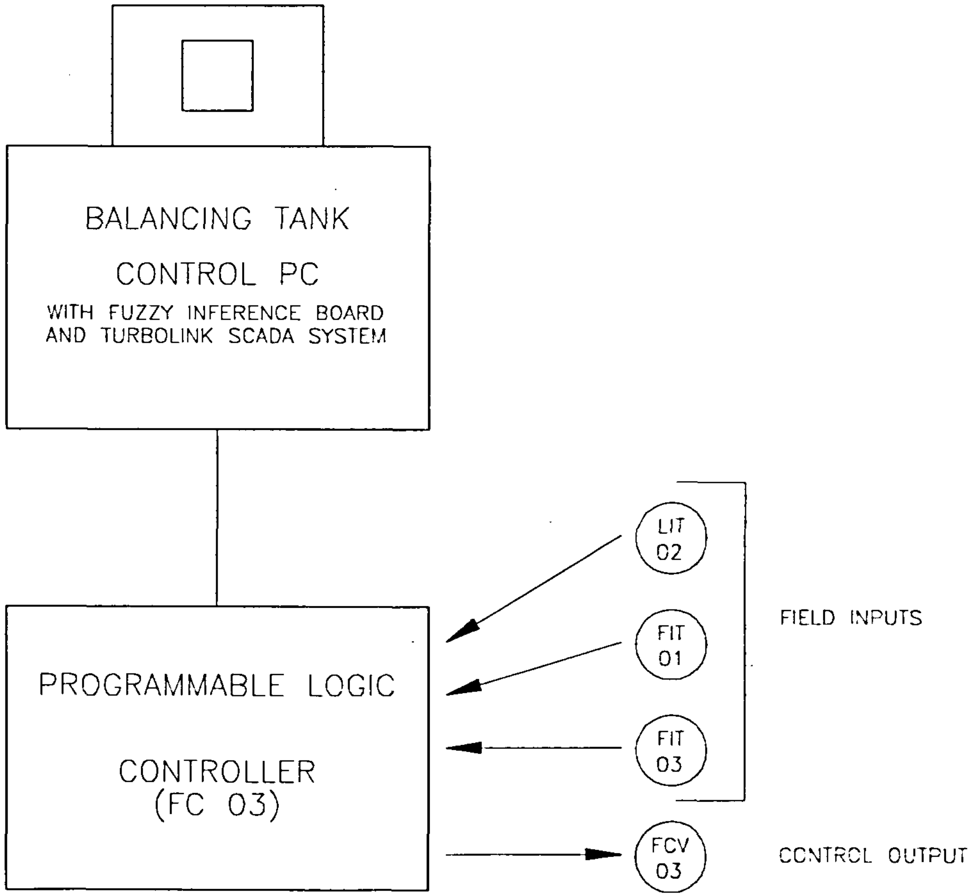


FIG. 2

SYSTEM CONFIGURATION DIAGRAM
OF THE CONTROL SYSTEM

a) The control actions of a human are often erratic and difficult to interpret.
b) The human operator responds not only to single measurements, but to complex patterns of measurements and observations are then categorised subjectively and used as a basis for control decisions.

Zadeh states that "as the complexity of a system increases, our ability to make precise and yet significant statements about its behaviour diminishes until a threshold is reached beyond which precision and significance become almost mutually exclusive characteristics". In other words, the closer a real-world problem is examined, the "fuzzier" becomes its solution. The problem in pattern recognition is often attributable to over-precision.

By using fuzzy feature definition instead of sharp thresholds improvement in performance can often be achieved. This has led to the development of "fuzzy sets" which are the basis of any fuzzy logic controller.

FUZZY SETS

In classical set theory there is a distinct difference between elements which belong to a set and those that do not. The set can be defined in terms of a membership function, μ, that can take values of either 0 or 1. If the variable is β then, If μ (β) = 0; β is not a member of the set.
If μ (β) = 1; β is a member of the set
This type of set is referred to a Crisp Set.

In fuzzy set theory, a fuzzy set A of a universe of discourse U is characterised by a membership function. μA (u), which assigns to each element u $\in$ U a number μA (u) in the interval 0 to 1, that represents the grade of membership in A, i.e.

A = ((u, μA (u))/u $\in$ U)

For example if one takes a linguistic variable such as age. A person who is 45 years old belongs neither to the young or old crisp sets but has a degree of membership in both the young and old, fuzzy sets.

LINGUISTIC HEDGES

One of the many attractions of using fuzzy set theory in control engineering problems is that qualitative expressions such as very small and rather big may be used. In our balancing tank application we can have a tank which is nearly full or rather empty. Fuzzy logic therefore is a way of treating vagueness in a way that can be handled by computers. It offers a methodology, firmly grounded in mathematical theory, for the handling of qualitative, inexact, imprecise information in a systematic and rigorous way.

Humans tend to summarise specific sensory variables (such as for example, distance) into imprecise linguistic values (a vague, imprecise, fuzzy description by language) to a variable yields "fuzzified values of the original variables.

FUZZY CONTROLLERS

In a fuzzy controller, the linguistic variables are used in fuzzy associations, or rules which connect a linguistically defined condition (or input) with a linguistically defined consequence (or output). The form of such a rule is :

IF < Condition > Then <Consequence>

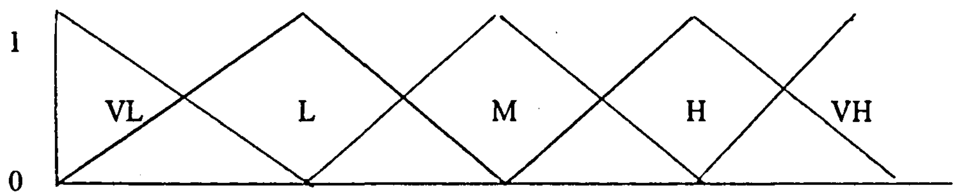
The fuzziness may be defined graphically in the form of a fuzzy distribution over the horizontal axis, and a corresponding membership function whose value can be any fraction from 0 to 1 plotted on the vertical axis.

There are a number of such rules which each produce a response. When given an input all of the fuzzy rules are activated in parallel, but to a different degree (or weighting). The response of each rule is weighted according to the degree of membership of its input variables.

From the combined fuzzy outputs contributed by each rule, a single value is chosen by "defuzzifying" the combined output. This is done by taking the centroid of the fuzzy responses.

The steps in building a fuzzy controller are therefore as follows :

1. Define the input variables (also known as conditions or antecedents).
2. Define the membership functions for each variable such as :



Where VL = Very Low
L = Low
M = Medium
H = High
VH = Very High

(This is called fuzzifying the inputs)

3. Define the "inference rules and weight the control action part of each rule.

4. Combine all rule outputs to the same controller output. (This output is still a fuzzy value).

5. Defuzzify the output to a crisp value. Usually by taking the centroid value.

The structure of a fuzzy controller is shown in Fig 3.

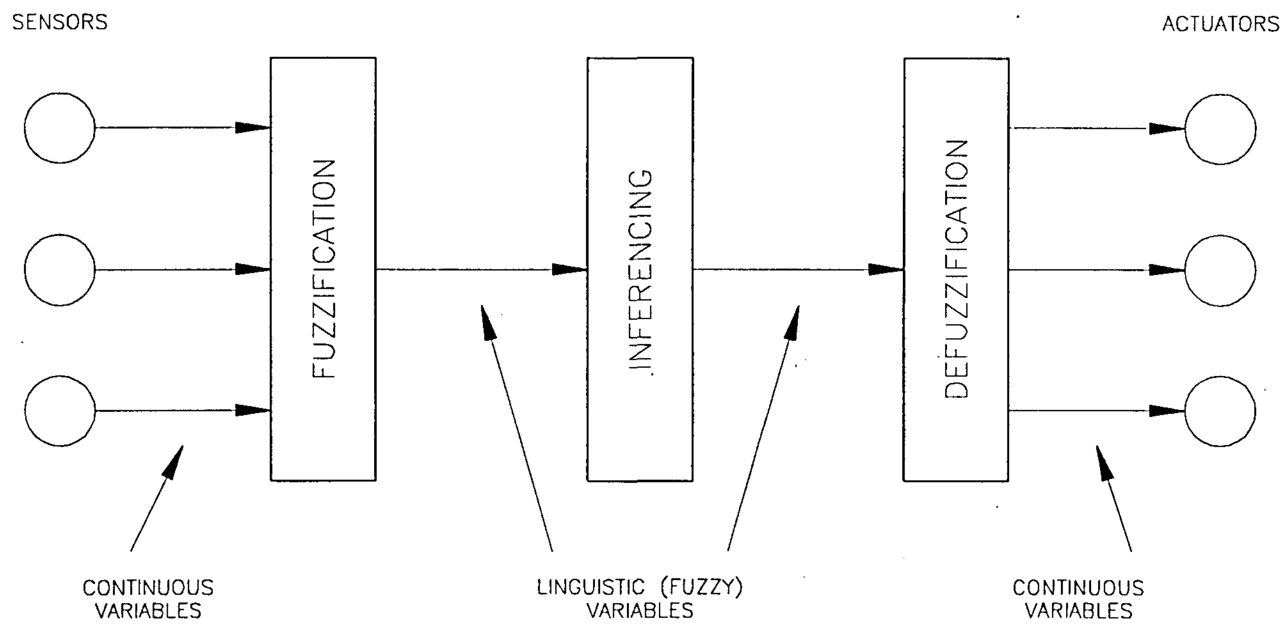


FIG. 3

THE STRUCTURE OF A FUZZY CONTROLLER

FUZZY LOGIC BALANCING TANK CONTROLLER

In designing our fuzzy logic balancing tank controller, numerous combinations of inputs, membership functions and rules were tried. Only the combination which yielded the best results will be described here.

The objectives of our controller were as follows :

- Utilise on-line inputs of inlet flow, tank level and outlet flow.
- Utilise historical inflow patterns.
- Calculate an outlet flow setpoint which can be used by a standard PID loop controller to control the outlet flow control valve.
- The rate of change of the outlet flow had to be minimised.
- The balancing tank should not be allowed to overflow or run dry.

STEP 1 - DEFINITION OF ANTECEDENTS (INPUTS)

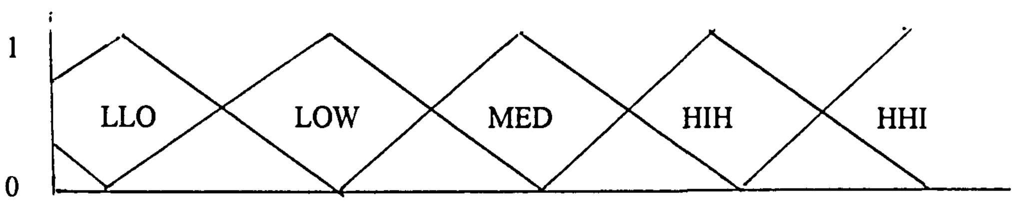
The following real inputs were defined.

Tank Level - (LEV)
Current Inflow - (INF)
Current Outflow - (OUT)
In addition the expected Inflow - (EIN) was obtained from a table of 48 values of historical half hourly inflow readings over a period of 24 hours.

STEP 2 - DEFINITION OF FUZZY SETS

The fuzzy sets were defined for each variable.

These all took the following form :



Where	LLO	=	Low Low
	LOW	=	Low
	MED	=	Medium
	HIH	=	High
	HHI	=	High High

STEP 3 - DEFINITION OF CONSEQUENT - OUTPUT

The consequent was defined as the tank outflow rate or (TOU).

STEP 4 - ESTABLISH INFERENCE RULES

The following rules were defined :

RULE NO.	RULE			WEIGHTING
1	IF LEV	=	HHI THEN SET TOU = HHI	1
2.	IF LEV	=	HIH AND INF = HIH AND EIN = HIH THEN SET TOU = HHI	1
3.	IF LEV	=	HIH AND INF = HIH THEN SET TOU = HIH	1
4.	IF LEV	=	LLO THEN SET TOU = LLO	1
5.	IF LEV	=	LOW AND INF = HIH AND EIN = HIH THEN SET TOU = MED	0,9
6.	IF LEV	=	LOW AND INF = LOW AND EIN = HIH THEN SET TOU = LOW	0,9
7.	IF LEV	=	LOW AND INF = LOW AND EIN = LOW THEN SET TOU = LLO	1
8.	IF LEV	=	MED AND INF = HIH AND EIN = HIH THEN SET TOU = HIH	0,9
9.	IF LEV	=	MED AND INF = LOW AND EIN = LOW THEN SET TOU = LOW	0,9
10.	IF LEV	=	MED AND INF = MED AND EIN = MED THEN SET TOU = MED	0,9
11.	IF LEV	=	MED AND INF = HIH AND EIN = HIH THEN SET TOU = HIH	0,9
12.	IF LEV	=	HIH AND INF = HHI AND EIN = HIH THEN SET TOU = HHI	1
13.	IF LEV	=	HIH AND INF = HIH AND EIN = HHI THEN SET TOU = HHI	1
14.	IF LEV	=	HIH AND INF = HIH AND EIN = HIH THEN SET TOU = HHI	1
15.	IF LEV	=	LOW AND INF = LLO AND EIN = LLO THEN SET TOU = LLO	1
16.	IF LEV	=	LOW AND INF = LOW AND EIN = LLO THEN SET TOU = LLO	1
17.	IF LEV	=	MED AND INF = HHI AND EIN = HHI THEN SET TOU = HIH	0,9
18.	IF LEV	=	MED AND INF = HIH AND EIN = HIH THEN SET TOU = HIH	0,9

STEP 5 - SELECT DEFUZZIFICATION METHOD

A defuzzification option had to be selected from the following options :

Centre of Gravity
Max Height Left
Max Height Right

For our controller the Centre of Gravity method was selected.

STEP 6 - RUN SIMULATION

The various controller combinations were tested using the simulation facility available with the FS - 10AT software. Simulation inflow data from Goudkoppies Wastewater Treatment Plant was used for the simulation. The resulting tank level after each outflow adjustment had to be separately calculated and then fed back as an input to the simulation.

RESEARCH RESULTS

The simulation test results were very disappointing. These are shown graphically in Appendix A. The control achieved using our fuzzy logic controller could not match that achieved using the mathematical models used by Dold et al or those achieved by Shaw using Neural networks.

The major failure with this controller was that it could not achieve the objective of minimising the <u>rate of change</u> of tank outflow and it was not adaptive or self correcting.

As a result of these poor simulation results, the author decided to abandon the use of fuzzy logic for the development of this controller and to rather develop an adaptive controller based on iterative mathematical calculations.

This modified controller is detailed in the next chapter.

4

IMPLEMENTED SOLUTION

SYSTEM CONFIGURATION

The overall system configuration is the same as shown in Fig 1. Physically the control system configuration for the modified controller is the same as shown in Fig 2 but with the following differences :

* The fuzzy logic inference board running the fuzzy logic software is no longer required.
* Instead the control algorithm runs as a PASCAL subroutine in the supervisory computer and is called by the SCADA software package. This routine is run every 30 minutes and the resulting tank outflow setpoint is passed via the SCADA system to the PID control loop in the PLC, which in turn controls the balancing tank outlet control valve.

PREVIOUS SHORTCOMINGS

The modified algorithm was designed to overcome the following shortcomings experienced with previous balancing tank control systems.

* The "Dold" algorithm achieved very good results, however it has a number of "tuning" constants which were difficult to tune on site.
* The neural network controller proposed by Midlane and Shaw had to be "trained" for each application using typical data sets. This training process was complex and time consuming. The simulation results achieved from this controller were however extremely good. Modern developments in neural network training software would make it worthwhile to re-investigate the neural network solution.
* The fuzzy logic controller was unstable and not self adapting.

DESCRIPTION OF THE MODIFIED CONTROL ALGORITHM

As mentioned previously the modified control algorithm was written as a Pascal subroutine which could be called by the plant SCADA system. Although it was written to interface with the Turbolink/Multilink version 8.1 SCADA system, it could easily be modified to run with any SCADA system.

The modified algorithm, like the Dold algorithm utilises historical inflow data to assist in predicting the required out put. 48 flowrates taken at half hourly intervals are stored to give a typical 24 hour weekday inflow pattern. Another 48 flowrates are stored to give a typical 24 hour weekend inflow pattern.

The algorithm is "called" every half hour by the SCADA system and it goes through the following steps.

Step 1 : Check current day and time from SCADA system.

Step 2 : Check current tank level.

Step 3 : Using the current tank level and the anticipated inflow for the next 24 hours and the current average outflow, calculate the change in tank level.

Step 4 : If with the current average outflow you will exceed any of the preset tank level limits, then adjust the outflow depending on which limit is exceeded. The change in level is then re-calculated. This is an iterative process until the optimum outflow is obtained where no tank levels are exceeded for the 24 hour period. Because it starts at the current outflow, the rate of change of outflow is minimised.

Step 5 : Calculate difference between actual inflow and historical inflow. Add or subtract a percentage of this error to the outflow setpoint. This percentage is configurable from the SCADA.

Step 6 : If the actual level is approaching one of the limits, then the algorithm will run every 5 minutes in order to prevent the tank level exceeding the limits.

Step 7 : At midnight the days 48 actual inflow values for the day, are compared to the 48 predicted values. The predicted values are then updated by taking the average of 95% of the predicted values and 5% of the actual values. In this way any long term changes in the inflow pattern such as seasonal variations would be accounted for by the algorithm.

Installation and operating instructions for the running of the balancing tank algorithm are given in Appendix B :

OPERATOR INTERFACE

The Turbolink SCADA package is used to provide an operator interface to the modified balancing tank control algorithm.
Three graphical mimic control screens are available to the operator to configure the algorithm.

Fig 4 : shows the configuration screen where the algorithm can be turned on or off, and the maximum and minimum tank levels can be set. The recommended outflow calculated, is shown on this screen. The Pascal sub routine also has a configuration section where the following parameters used by the algorithm can be entered as described in Appendix B.

* Tank volume
* Top limit
* Bottom limit
* 48 weekday inflow values
* 48 weekend inflow values
* Number of outlet control valves

Fig 5 : is a graphical mimic status display screen which shows the values of all analog variables and the status of all drives. This screen has to be customised for the particular balancing tank where the algorithm is installed.

Fig 6 : is a control screen where the parameters for the PID control loop for the outlet control valve are set up. The setpoint displayed on this screen is the setpoint passed down from the control algorithm.

IMPLEMENTATION RESULTS

The modified balancing tank controller as described in section 4 has been successfully installed and commissioned at the following wastewater treatment plants :

* Northern Works Unit 4
* Olifantsvlei Unit 3
* Goudkoppies Unit 1
* Northern Works Unit 3

The controller has achieved all the objectives set.

* It is easy to set up and configure
* It maintains a relatively constant outflow pattern and limits the rate of change of the outlet flow.
* It prevents the tank from either overflowing or running dry.

Figures 7, 8 and 9 show results obtained at Northern Works Unit 4. The graphs show the tank inflow, the tank level and the tank outflow over a 3 day period.

It can be seen from the graphs that the controller is operating successfully.

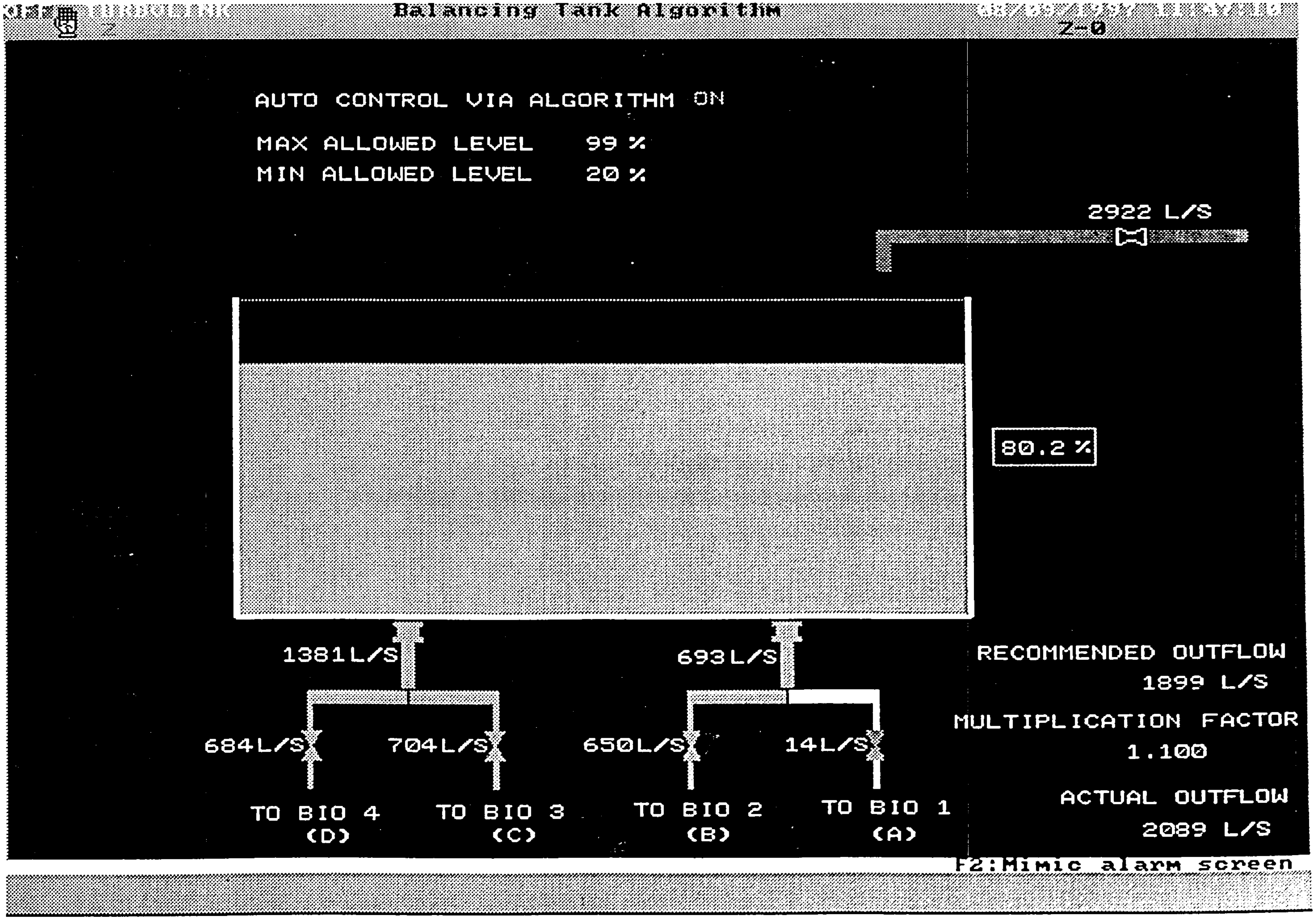


FIG 4 : CONFIGURATION SCREEN

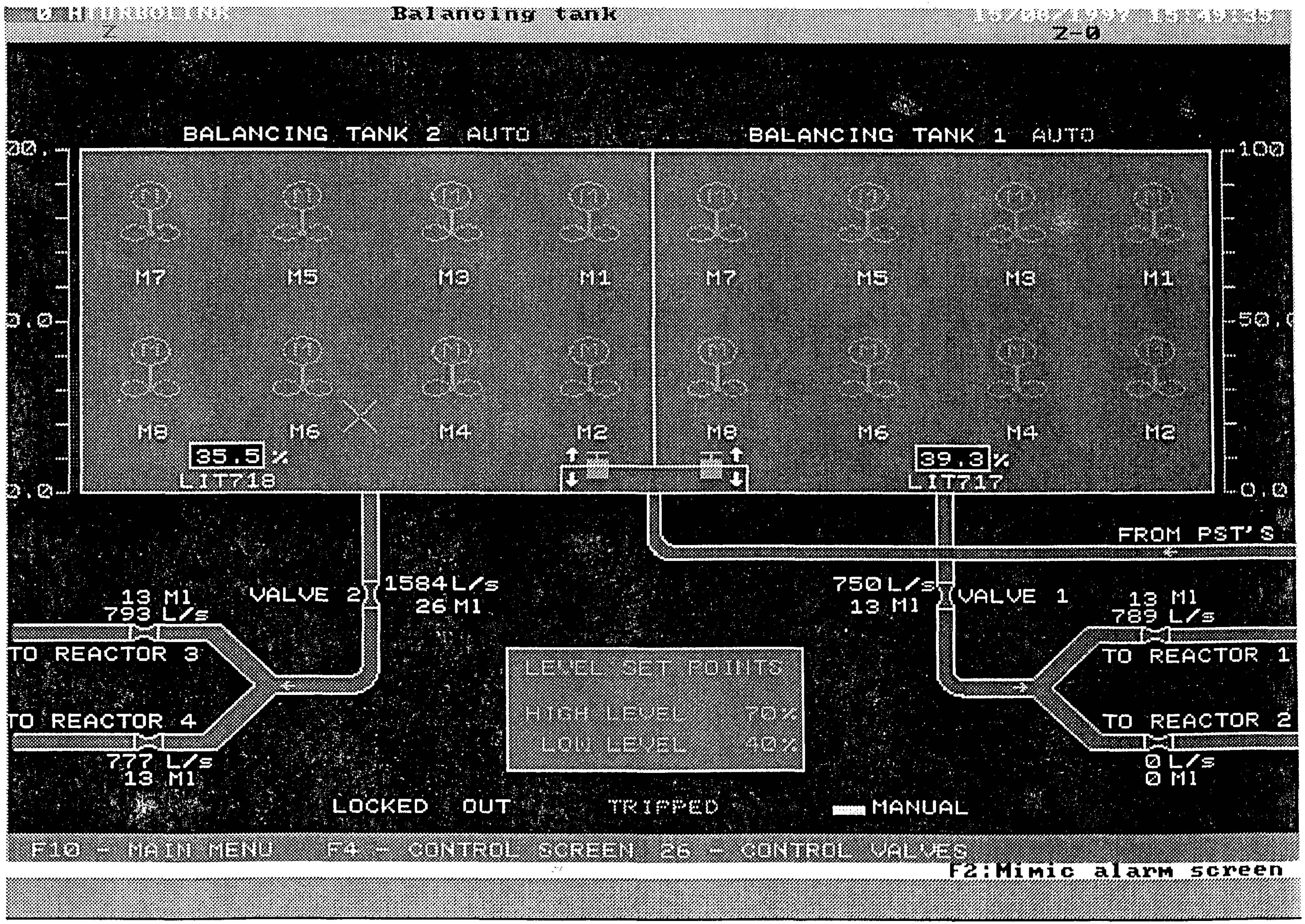


FIG 5 : STATUS DISPLAY SCREEN

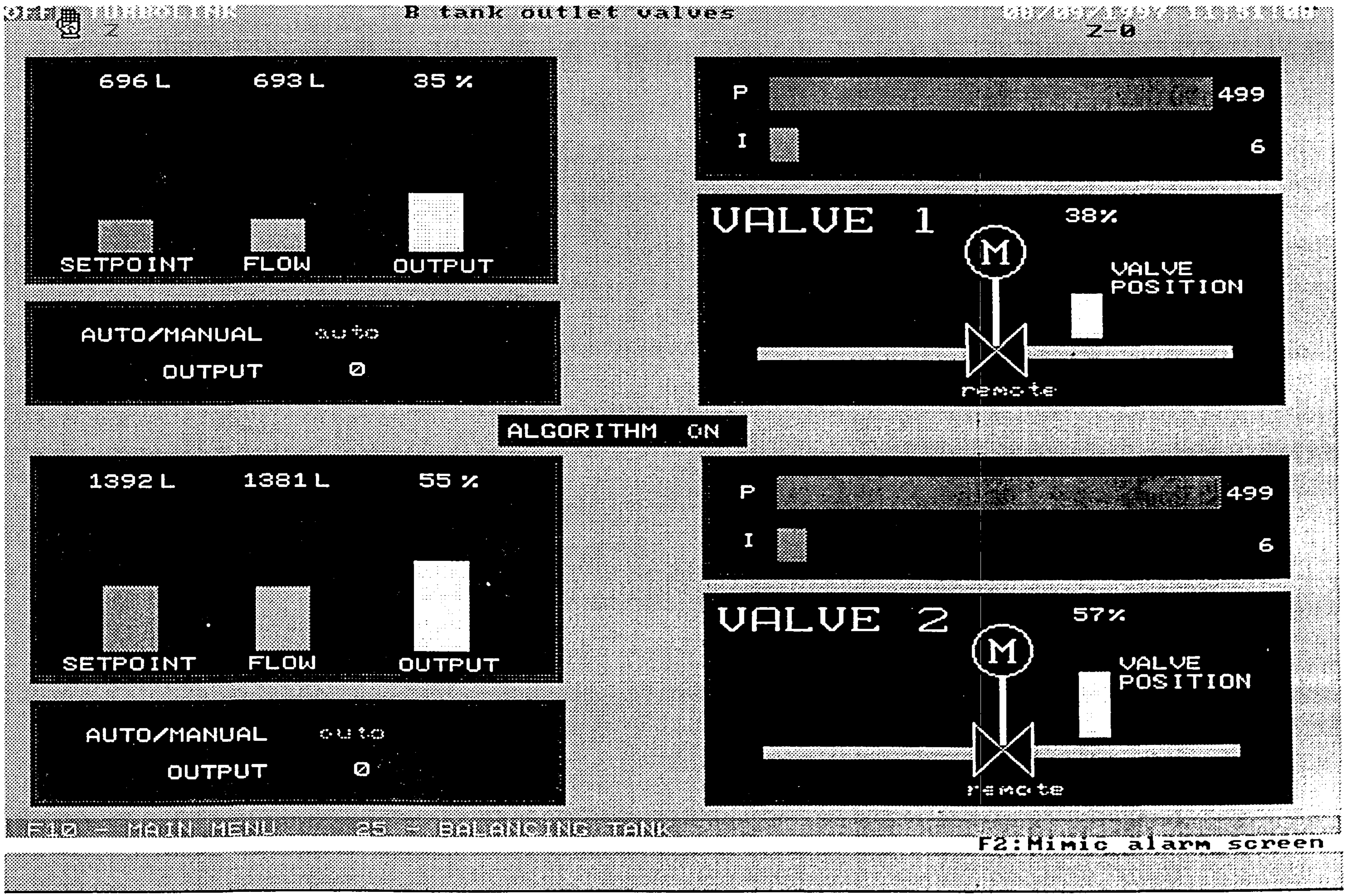


FIG 6 : OUTLET VALVE CONTROL SCREEN

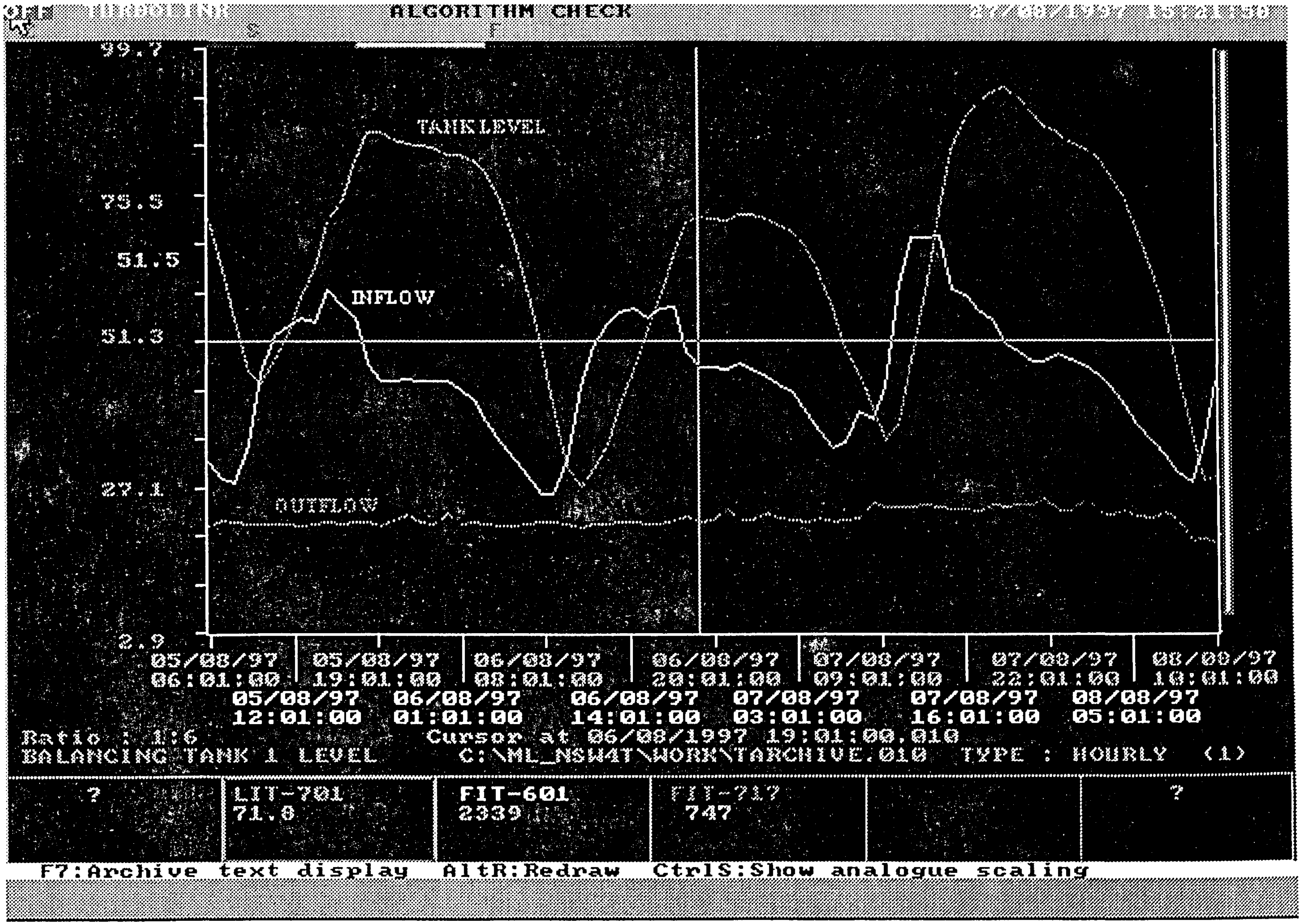


FIG 7 : RESULTS AT NORTHERN WORKS

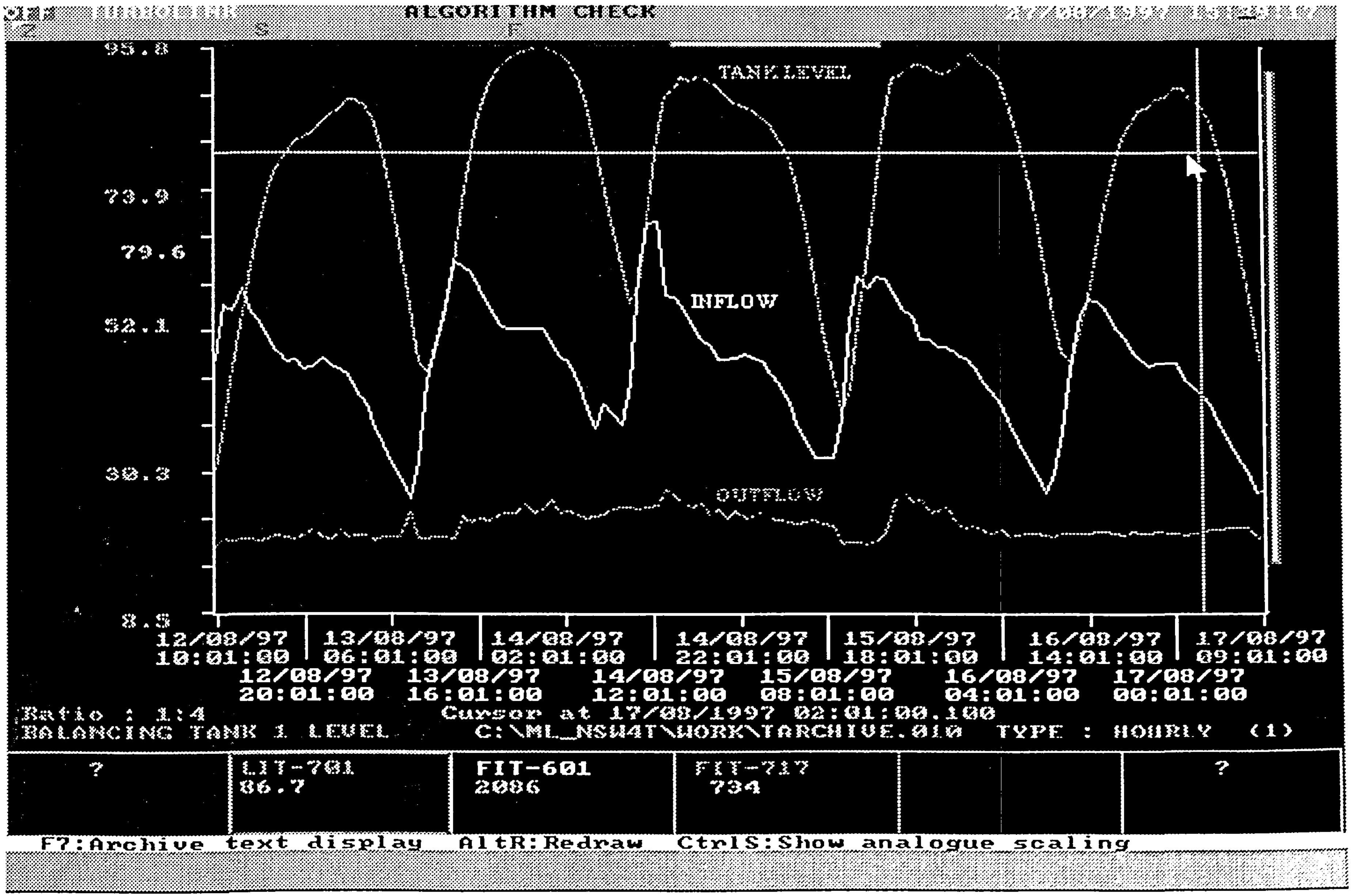


FIG 8 : RESULTS AT NORTHERN WORKS

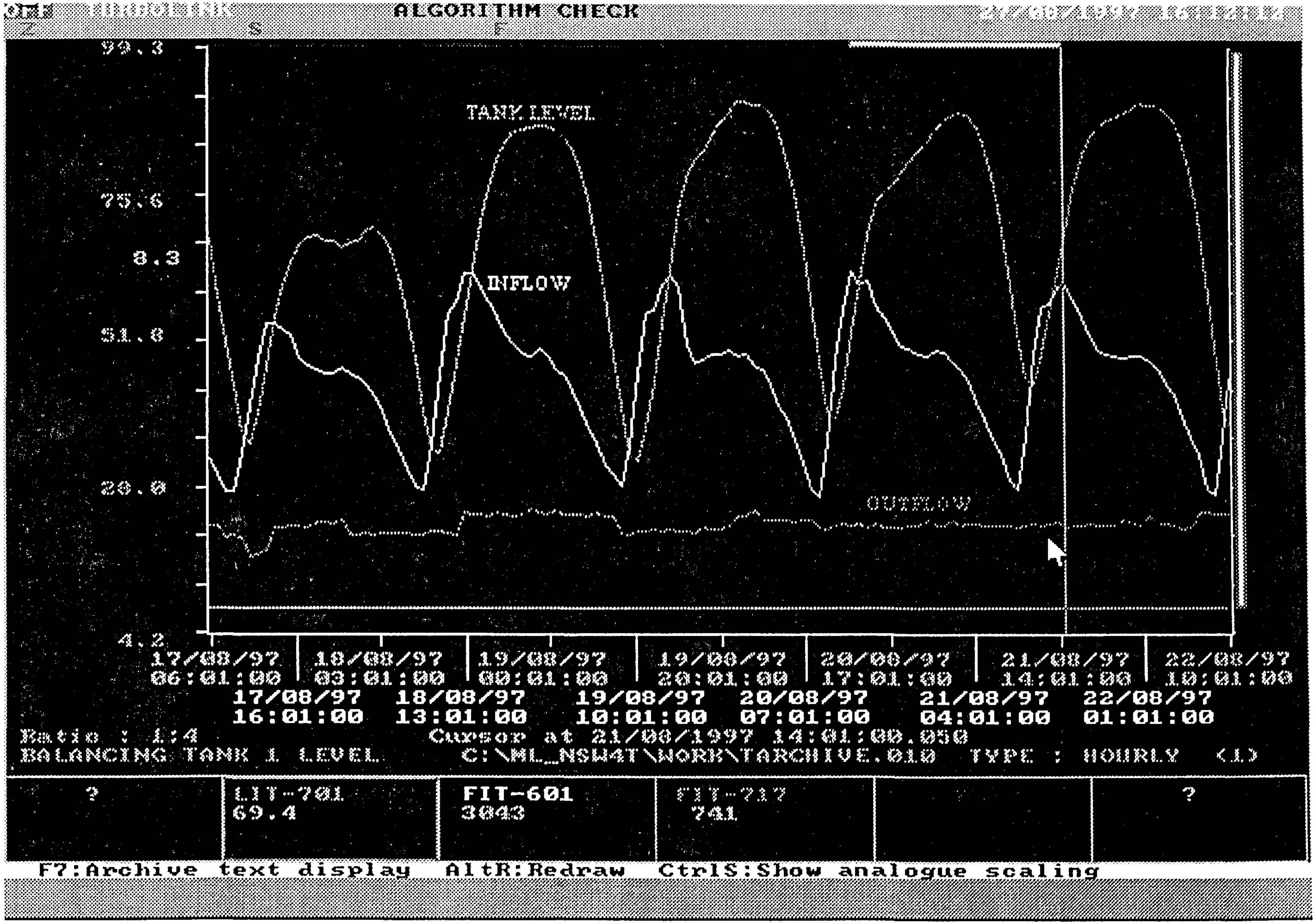


FIG 9 : RESULTS AT NORTHERN WORKS

5

CONCLUSION

From the research conducted we can draw the following conclusions :

* Fuzzy Logic cannot be used to develop a balancing tank controller which is easy to set up and configure and which provides stable control. The main problem associated with the fuzzy logic controller which was developed, was the fact that it was not self adapting. In other words it would not be self correcting.

* It is possible to develop a controller based on iterative mathematical calculations, using historical data as a reference. The modified controller was developed to run as a Pascal subroutine, and was interfaced with a standard SCADA package which provided a user friendly operator interface. The modified controller has been successfully installed at 4 wastewater treatment works operated by the Greater Johannesburg Metropolitan Council.

* Future research work may focus on developing a controller based on neural networks which have proved to have excellent pattern recognition characteristics. Early work in this regard showed promising results, however the complexities associated with "training" the neural networks proved to be a limiting factor. Recent developments in neural network training software may however have paved the way for further work in this field.

REFERENCES

1. Shaw I.S. (1992) Fuzzy Logic - An introduction, Elektron, pp 8-11, July 1992

2. Zadeh, L.A. (1973) Outline of a new approach to the analysis of complete systems and decision processes, IEEE Trans. on Systems, Man and Cybernetics, SMC-3, no. 1, pp 28-44, January 1973

3. Sutton R & Towill D.R. (1985) An Introduction to the use of fuzzy sets in the implementation of Control Algorithms, Journal of the Institution of Electronic and Radio Engineers, Vol. 55, No. 10, pp 357-367, October 1985.

4. Dold P.L.; Buhr H.O., Marais G.V.R. (1982) Design and Control of Equalisation Tanks, Water Research Commission research report no. W42, February 1982.

5. Omron Corporation, (1991), FS-10AT Fuzzy Inference Software User's Manual.

6. Froese T. (1994), Applying Fuzzy Logic and neural networks in modern process control systems, Chemical Processing S.A., pp 40-42, February 1994

7. Kosko B, Neural Networks and Fuzzy Systems : A Dynamic Systems Approach to Machine Intelligence. Eaglewood Cliffs, N J : Prentice - Hall, 1992.

8. Mather A.J., Shaw I.S. (1993), alternative method for the control of balancing tank at a Wastewater Treatment Plant, Wat. Sci. Tech. Vol 28, No. 11-12 pp 523 - 530, 1993

APPENDIX A

RESULTS ACHIEVED WITH BALANCING TANK CONTROLLERS

FIG 10	:	Comparative results of mathematical model based (Dold Algorithm) controller vs. Neural network controller.
FIG 11	:	Comparative results of human operator vs. Neural network controller.
TABLE 1	:	Fuzzy Logic Controller - rule definition.
TABLE 2	:	Fuzzy Logic Controller - signal and label information.
FIG 12	:	Graphic results of the fuzzy logic controller

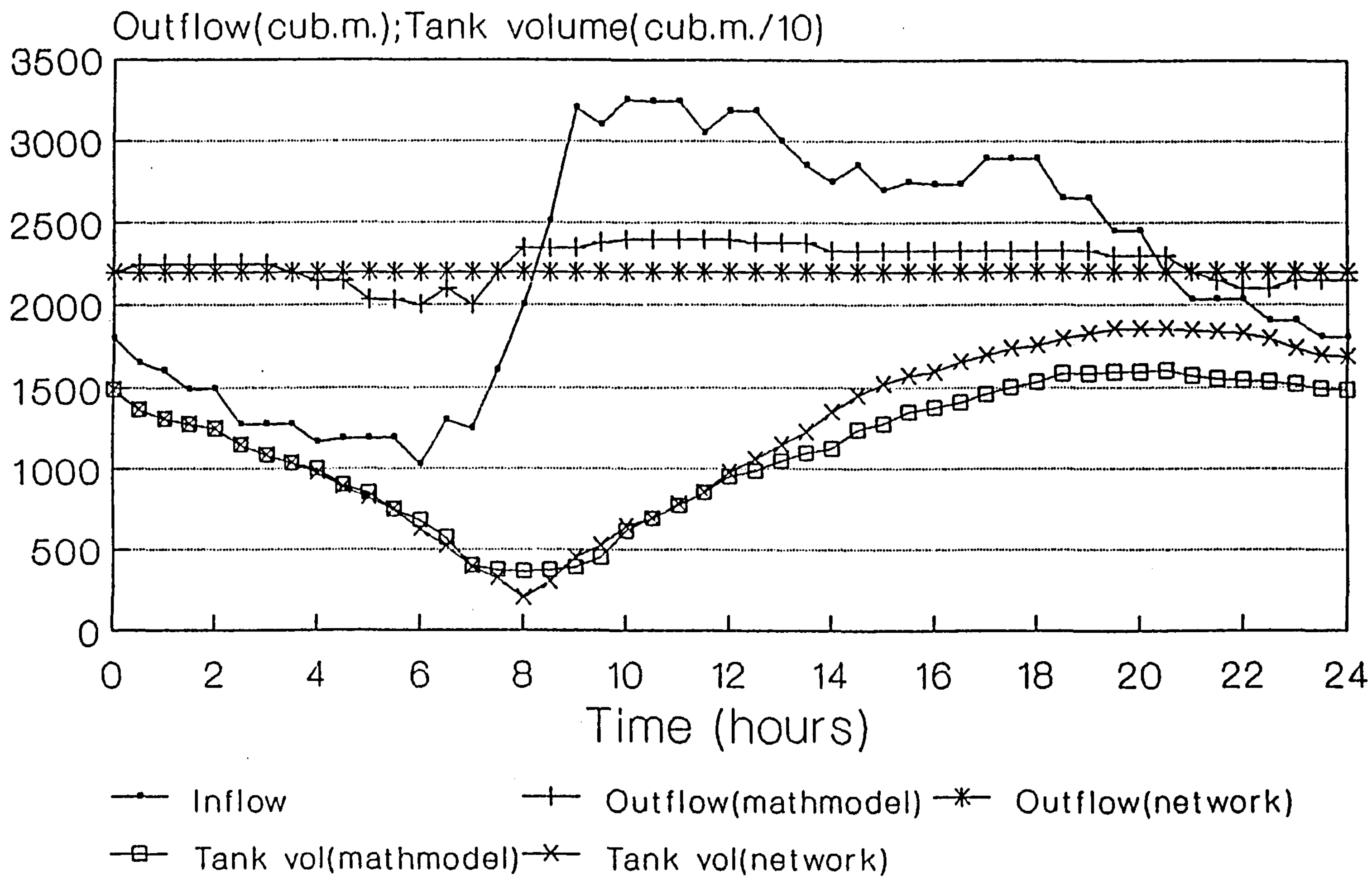


FIG 10 : COMPARITIVE RESULTS OF MATHEMATICAL MODEL CONTROLLER VS NEURAL NETWORK CONTROLLER

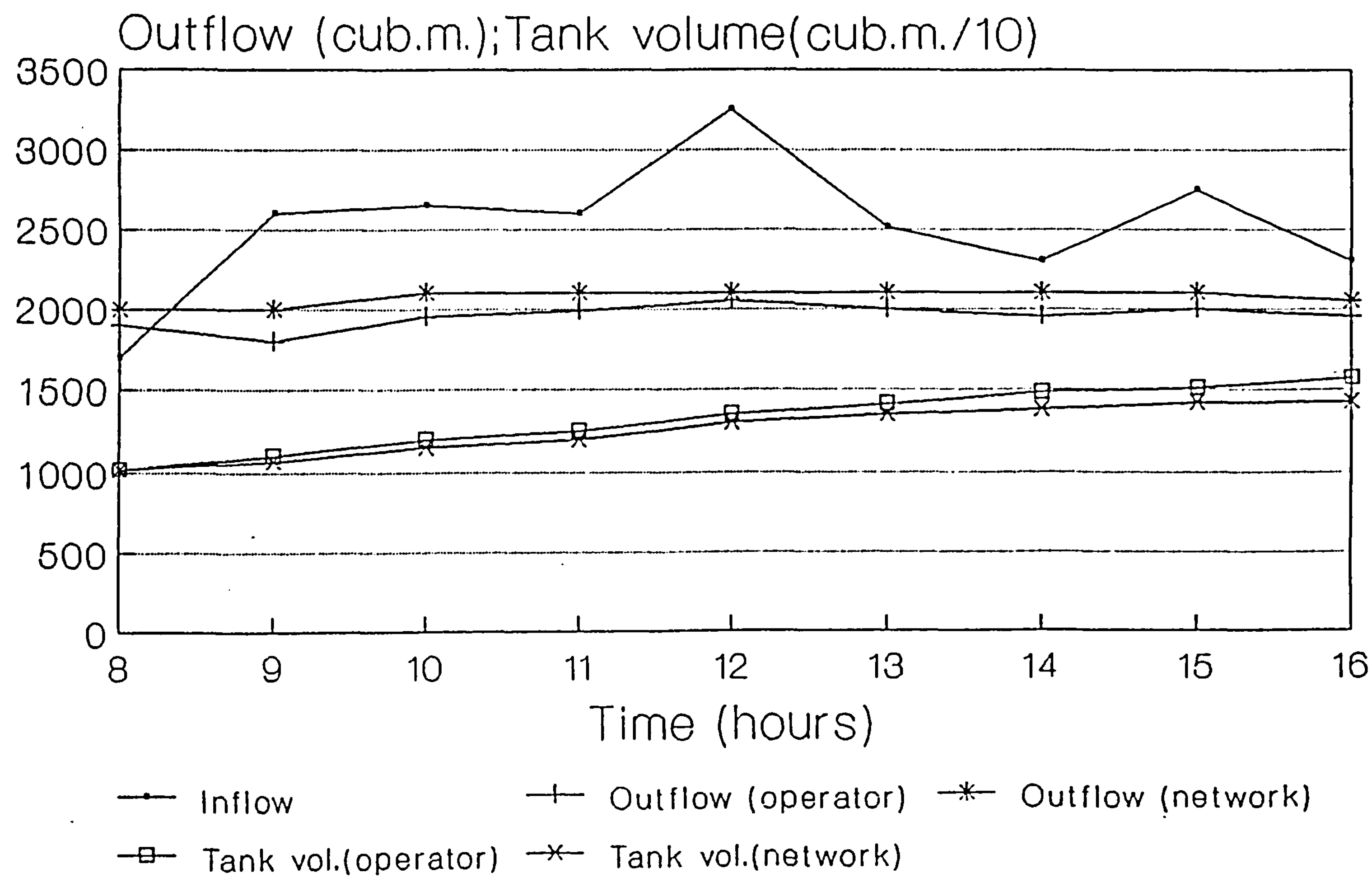


FIG 11 : COMPARITIVE RESULTS OF HUMAN OPERATOR VS NEURAL NETWORK CONTROLLER

FILE NAME : TANK2 DATE: 96/03/21

COMMENT: BALANCING TANK TEST

ANTECEDENT					CONSEQUENT			
RULE NO.	LEV	INF	EIN			OUT		WEIGHT (* = 0)
1	HHI					HHI		1
2	HIH	HIH	HIH			HHI		1
3	HIH	HIH				HIH		1
4	LLO					LLO		1
5	LOW	HIH	HIH			MED		0.9
6	LOW	LOW	HIH			LOW		0.9
7	LOW	LOW	LOW			LLO		1
8	MED	HIH	HIH			HIH		0.9
9	MED	LOW	LOW			LOW		0.9
10	MED	MED	MED			MED		0.9
11	MED	HIH	HIH			HIH		0.9
12	HIH	HHI	HIH			HHI		1
13	HIH	HIH	HHI			HHI		1
14	HIH	HIH	HIH			HHI		1
15	LOW	LLO	LLO			LLO		1
16	LOW	LOW	LLO			LLO		1
17	MED	HHI	HHI			HIH		0.9
18	MED	HIH	HIH			HIH		0.9

TABLE 1 : FUZZY LOGIC CONTROLLER - RULE DEFINITION

SIGNAL AND LABEL INFORMATION

SIGNAL	LABELS				
LEV	HHI	HIH	MED	LOW	LLO
INF	HHI	HIH	MED	LOW	LLO
EIN	HHI	HIH	MED	LOW	LLO
OUT	HHI	HIH	MED	LOW	LLO

TABLE 2 : FUZZY LOGIC CONTROLLER - SIGNAL & LABEL INFORMATION

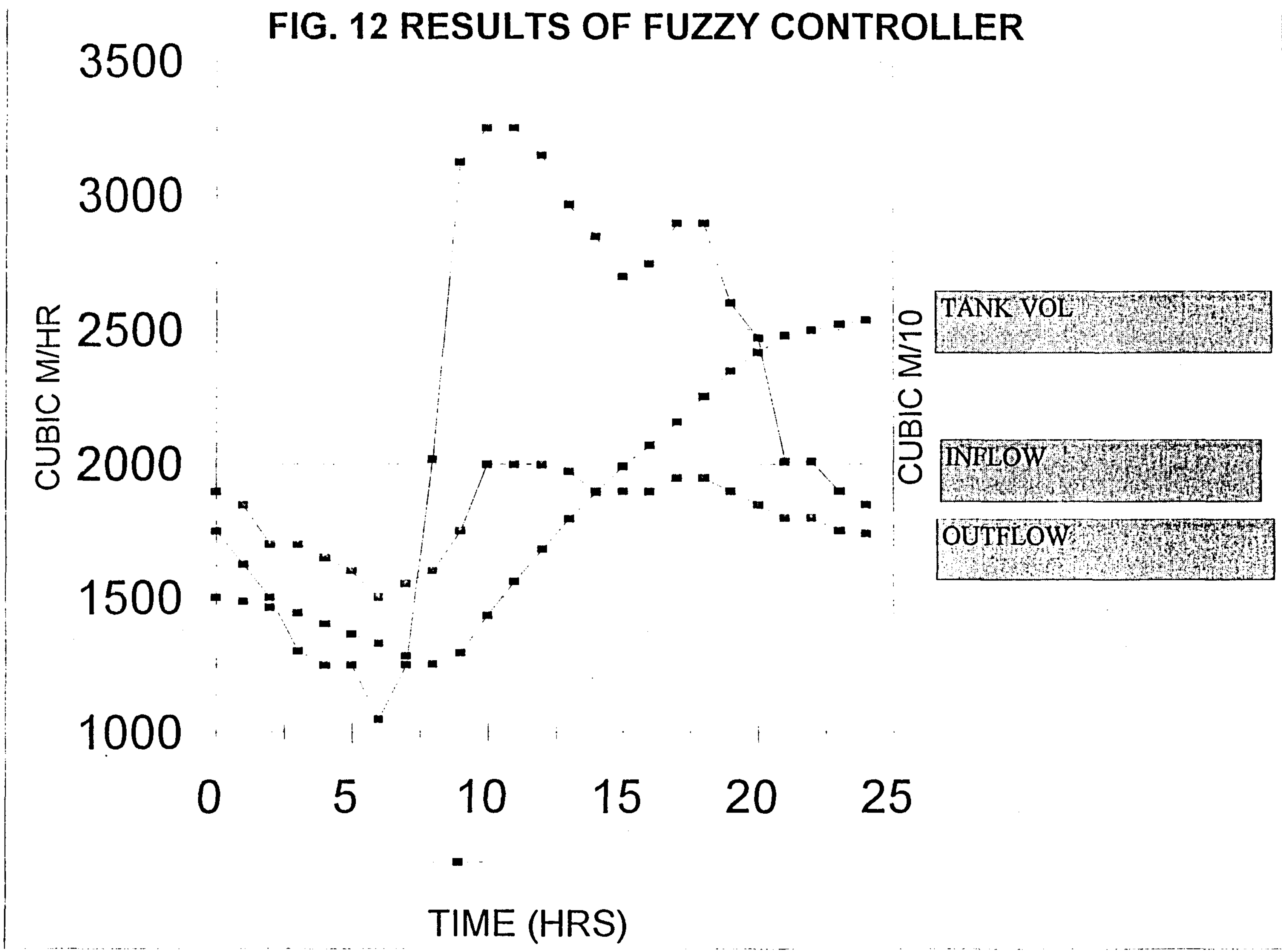


FIG. 12 RESULTS OF FUZZY CONTROLLER
3500
3000
2500
2000
1500
1000
CUBIC M/HR
CUBIC M/10
0
5
10
15
20
25
TANK VOL
INFLOW
OUTFLOW
TIME (HRS)

APPENDIX B

Installation and Operating Instructions for the Modified Balancing Tank Controller.

BALANCING TANK CONTROL ALGORITHM

Installation Instructions:

1. This algorithm only works with TurboLink/MultiLink 8.1
2. The disk contains the following files:

ROOT	Readme.doc	
	TP	Bal.exe
		Holiday.dat
		Apidone.exe
	DATA	Analdb81.prn
		Dbtx81.exe

3. After installing TurboLink/MultiLink from the disks run the package. Exit the package and create the directory TP from the TurboLink/Multilink directory and copy the files in the TP directory.

4. Copy the files from the DATA directory to the TurboLink/MultiLink directory and run DBTX81.EXE. At the prompt convert the analogue text to database (F6).

5. Create a directory called BAL_DATA and copy the file HOLIDAY.DAT to this new directory. Failure to do this will result in a runtime error.

6. Run BAL.EXE with the required interrupt eg BAL 90. Configure the comms definition to scan device 73 (API) and use the corresponding interrupt.

7. Rerun TurboLink/MultiLink. Ensure that the Tank Volume, Upper and Lower limits as well as all the Historical data is not zero.

8. Running ApiDone and specifying the interrupt will purge the TSR from memory (eg ApiDone 90).

Definition

Volume	:Tank volume in Mega Litres
Top Limit	:Maximum upper limit the tank is to be controlled to
Bot Limit	:Minimum lower limit the tank is to be controlled to
WDay????	:Historical flow for a weekday at time ????
WEnd????	:Historical flow for a weekend at time ????
Flow_In	:Actual flow in as received from a field device
Level	:Actual tank level as received from a field device
Flow Out	:Command from the algorithm. This is the optimum required outflow
Flow Ave	:Average flow expected over the next 24 hours. Useful to see how the algorithm is operating

IMPLEMENTATION OF THE BALANCING TANK ALGORITHM

GOUDKOPPIES WASTEWATER TREATMENT WORKS

BALANCING TANKS

25/06/97
1.0

SCADA DATABASE NO.	DESCRIPTION	SETPOINTS	UNITS
0	VOLUME (OF BALANCING TANK)	?	CU METRES
1	MAX LIMIT (MAX ALLOWABLE LIMIT BEFORE EMERGENCY PROCEDURE)	80	%
2	MINIMUM LIMIT	20	%
3	WEEKDAY EXPECTED INFLOW AT 00:00	1200	RANGE 0-4095
4	WEEKDAY EXPECTED INFLOW AT 00:30	1200	RANGE 0-4095
5	WEEKDAY EXPECTED INFLOW AT 01:00	1200	RANGE 0-4095
6	WEEKDAY EXPECTED INFLOW AT 01:30	1200	RANGE 0-4095
7	WEEKDAY EXPECTED INFLOW AT 02:00	1200	RANGE 0-4095
8	WEEKDAY EXPECTED INFLOW AT 02:30	1200	RANGE 0-4095
9	WEEKDAY EXPECTED INFLOW AT 03:00	1200	RANGE 0-4095
10	WEEKDAY EXPECTED INFLOW AT 03:30	1200	RANGE 0-4095
11	WEEKDAY EXPECTED INFLOW AT 04:00	1200	RANGE 0-4095
12	WEEKDAY EXPECTED INFLOW AT 04:30	1200	RANGE 0-4095
13	WEEKDAY EXPECTED INFLOW AT 05:00	1200	RANGE 0-4095
14	WEEKDAY EXPECTED INFLOW AT 05:30	1200	RANGE 0-4095
15	WEEKDAY EXPECTED INFLOW AT 06:00	1200	RANGE 0-4095
16	WEEKDAY EXPECTED INFLOW AT 06:30	1200	RANGE 0-4095
17	WEEKDAY EXPECTED INFLOW AT 07:00	1200	RANGE 0-4095
18	WEEKDAY EXPECTED INFLOW AT 07:30	1200	RANGE 0-4095
19	WEEKDAY EXPECTED INFLOW AT 08:00	1200	RANGE 0-4095
20	WEEKDAY EXPECTED INFLOW AT 08:30	1200	RANGE 0-4095
21	WEEKDAY EXPECTED INFLOW AT 09:00	1200	RANGE 0-4095
22	WEEKDAY EXPECTED INFLOW AT 09:30	1200	RANGE 0-4095
23	WEEKDAY EXPECTED INFLOW AT 10:00	1200	RANGE 0-4095
24	WEEKDAY EXPECTED INFLOW AT 10:30	1200	RANGE 0-4095
25	WEEKDAY EXPECTED INFLOW AT 11:00	1200	RANGE 0-4095
26	WEEKDAY EXPECTED INFLOW AT 11:30	1200	RANGE 0-4095
27	WEEKDAY EXPECTED INFLOW AT 12:00	1200	RANGE 0-4095
28	WEEKDAY EXPECTED INFLOW AT 12:30	1200	RANGE 0-4095
29	WEEKDAY EXPECTED INFLOW AT 13:00	1200	RANGE 0-4095
30	WEEKDAY EXPECTED INFLOW AT 13:30	1200	RANGE 0-4095
31	WEEKDAY EXPECTED INFLOW AT 14:00	1200	RANGE 0-4095
32	WEEKDAY EXPECTED INFLOW AT 14:30	1200	RANGE 0-4095
33	WEEKDAY EXPECTED INFLOW AT 15:00	1200	RANGE 0-4095
34	WEEKDAY EXPECTED INFLOW AT 15:30	1200	RANGE 0-4095
35	WEEKDAY EXPECTED INFLOW AT 16:00	1200	RANGE 0-4095
36	WEEKDAY EXPECTED INFLOW AT 16:30	1200	RANGE 0-4095
37	WEEKDAY EXPECTED INFLOW AT 17:00	1200	RANGE 0-4095
38	WEEKDAY EXPECTED INFLOW AT 17:30	1200	RANGE 0-4095
39	WEEKDAY EXPECTED INFLOW AT 18:00	1200	RANGE 0-4095
40	WEEKDAY EXPECTED INFLOW AT 18:30	1200	RANGE 0-4095
41	WEEKDAY EXPECTED INFLOW AT 19:00	1200	RANGE 0-4095
42	WEEKDAY EXPECTED INFLOW AT 19:30	1200	RANGE 0-4095
43	WEEKDAY EXPECTED INFLOW AT 20:00	1200	RANGE 0-4095
44	WEEKDAY EXPECTED INFLOW AT 20:30	1200	RANGE 0-4095
45	WEEKDAY EXPECTED INFLOW AT 21:00	1200	RANGE 0-4095
46	WEEKDAY EXPECTED INFLOW AT 21:30	1200	RANGE 0-4095
47	WEEKDAY EXPECTED INFLOW AT 22:00	1200	RANGE 0-4095
48	WEEKDAY EXPECTED INFLOW AT 22:30	1200	RANGE 0-4095
49	WEEKDAY EXPECTED INFLOW AT 23:00	1200	RANGE 0-4095
50	WEEKDAY EXPECTED INFLOW AT 23:30	1200	RANGE 0-4095
51	WEEKEND EXPECTED INFLOW AT 00:00	1200	RANGE 0-4095
52	WEEKEND EXPECTED INFLOW AT 00:30	1200	RANGE 0-4095
53	WEEKEND EXPECTED INFLOW AT 01:00	1200	RANGE 0-4095
54	WEEKEND EXPECTED INFLOW AT 01:30	1200	RANGE 0-4095
55	WEEKEND EXPECTED INFLOW AT 02:00	1200	RANGE 0-4095
56	WEEKEND EXPECTED INFLOW AT 02:30	1200	RANGE 0-4095
57	WEEKEND EXPECTED INFLOW AT 03:00	1200	RANGE 0-4095
58	WEEKEND EXPECTED INFLOW AT 03:30	1200	RANGE 0-4095
59	WEEKEND EXPECTED INFLOW AT 04:00	1200	RANGE 0-4095
60	WEEKEND EXPECTED INFLOW AT 04:30	1200	RANGE 0-4095
61	WEEKEND EXPECTED INFLOW AT 05:00	1200	RANGE 0-4095
62	WEEKEND EXPECTED INFLOW AT 05:30	1200	RANGE 0-4095
63	WEEKEND EXPECTED INFLOW AT 06:00	1200	RANGE 0-4095
64	WEEKEND EXPECTED INFLOW AT 06:30	1200	RANGE 0-4095
65	WEEKEND EXPECTED INFLOW AT 07:00	1200	RANGE 0-4095
66	WEEKEND EXPECTED INFLOW AT 07:30	1200	RANGE 0-4095

67	WEEKEND EXPECTED INFLOW AT 08:00	1200	RANGE 0-4095
68	WEEKEND EXPECTED INFLOW AT 08:30	1200	RANGE 0-4095
69	WEEKEND EXPECTED INFLOW AT 09:00	1200	RANGE 0-4095
70	WEEKEND EXPECTED INFLOW AT 09:30	1200	RANGE 0-4095
71	WEEKEND EXPECTED INFLOW AT 10:00	1200	RANGE 0-4095
72	WEEKEND EXPECTED INFLOW AT 10:30	1200	RANGE 0-4095
73	WEEKEND EXPECTED INFLOW AT 11:00	1200	RANGE 0-4095
74	WEEKEND EXPECTED INFLOW AT 11:30	1200	RANGE 0-4095
75	WEEKEND EXPECTED INFLOW AT 12:00	1200	RANGE 0-4095
76	WEEKEND EXPECTED INFLOW AT 12:30	1200	RANGE 0-4095
77	WEEKEND EXPECTED INFLOW AT 13:00	1200	RANGE 0-4095
78	WEEKEND EXPECTED INFLOW AT 13:30	1200	RANGE 0-4095
79	WEEKEND EXPECTED INFLOW AT 14:00	1200	RANGE 0-4095
80	WEEKEND EXPECTED INFLOW AT 14:30	1200	RANGE 0-4095
81	WEEKEND EXPECTED INFLOW AT 15:00	1200	RANGE 0-4095
82	WEEKEND EXPECTED INFLOW AT 15:30	1200	RANGE 0-4095
83	WEEKEND EXPECTED INFLOW AT 16:00	1200	RANGE 0-4095
84	WEEKEND EXPECTED INFLOW AT 16:30	1200	RANGE 0-4095
85	WEEKEND EXPECTED INFLOW AT 17:00	1200	RANGE 0-4095
86	WEEKEND EXPECTED INFLOW AT 17:30	1200	RANGE 0-4095
87	WEEKEND EXPECTED INFLOW AT 18:00	1200	RANGE 0-4095
88	WEEKEND EXPECTED INFLOW AT 18:30	1200	RANGE 0-4095
89	WEEKEND EXPECTED INFLOW AT 19:00	1200	RANGE 0-4095
90	WEEKEND EXPECTED INFLOW AT 19:30	1200	RANGE 0-4095
91	WEEKEND EXPECTED INFLOW AT 20:00	1200	RANGE 0-4095
92	WEEKEND EXPECTED INFLOW AT 20:30	1200	RANGE 0-4095
93	WEEKEND EXPECTED INFLOW AT 21:00	1200	RANGE 0-4095
94	WEEKEND EXPECTED INFLOW AT 21:30	1200	RANGE 0-4095
95	WEEKEND EXPECTED INFLOW AT 22:00	1200	RANGE 0-4095
96	WEEKEND EXPECTED INFLOW AT 22:30	1200	RANGE 0-4095
97	WEEKEND EXPECTED INFLOW AT 23:00	1200	RANGE 0-4095
98	WEEKEND EXPECTED INFLOW AT 23:30	1200	RANGE 0-4095
99	SPARE		
100	SPARE		
101	BALANCING TANK INFLOW		RANGE 0-4095
102	BALANCING TANK LEVEL		%
103	SPARE		
104	SPARE		
105	SPARE		
106	SPARE		
107	SPARE		
108	SPARE		
109	SPARE		
110	OUTFLOW FROM BALANCING TANK		RANGE 0-4095
111	AVERAGE OUTFLOW FROM BALANCING TANK		RANGE 0-4095
112	SPARE		
113	SPARE		
114	SPARE		
115	SPARE		
116	SPARE		
117	SPARE		
118	SPARE		
119	SPARE		
120	SPARE		
121	SPARE		
122	SPARE		
123	SPARE		
124	SPARE		
125	SPARE		
126	SPARE		
127	SPARE		

COMMUNICATIONS DEFINITION CONFIGURATION

THE FOLLOWING COMMS BLOCK SHOULD BE INSERTED INTO THE COMMUNICATIONS DEFINITION SCREEN

Blk No	Enable Tag	Bk Typ	DB Strt	DB End			DT	Dbg Scn	Intr No	Route	Result Tag
0		DIN	7000	7000	----	----	73	1	90	---	

THIS BLOCK ENABLES THE SCADA TO COMMUNICATE WITH THE API DRIVER.

A SUBDIRECTORY CALLED 'TP' SHOULD BE CREATED OFF THE TLINK/MLINK DIRECTORY FROM WHERE YOUR TURBOLINK/MUL-T-LINK SYSTEM IS RUNNING. THE 'TP' DIRECTORY MUST CONTAIN THE BALANCING ALGORITHM PROGRAM (BAL.EXE) AS WELL AS ALL THE API DRIVER SOFWARE FILES.

A DIRECTORY CALLED 'BAL_DATA' SHOULD BE CREATED OFF THE ROOT DIRECTORY AND SHOULD CONTAIN A FILE CALLED 'HOLIDAY.DAT'. THIS FILE CONTAINS THE DATES FOR ALL THE PUBLIC HOLIDAYS FOR THE YEAR AND SHOULD APPEAR AS FOLLOWS:

1 1
21 3
28 3
31 3
27 4
28 4
1 5
16 6
8 8
24 9
16 12
25 12
26 12