DEVELOPMENT OF AN ELECTRONIC VEHICULAR TRAFFIC SIGNAL CONTROLLER

D. Anipa
Department of Electrical and Electronic Engineering,
College of Engineering,
KNUST, Kumasi

ABSTRACT

This paper presents the design, construction, and test of an electronic signal controller for urban vehicular traffic control. The design is based on a series of fixed-time signal plans for different time zones of the day and can accommodate sixty-four signal plans. Finite state machine concept was used in the design of the signal controller and the implementation carried out using Large Scale Integrated (LSI) devices. The controller is driven by a real-time clock and can be configured for a six-phase intersection.

Keywords: *Semi-adaptive Vehicular Traffic Control, Fixed-time Signal Plan, Digital System Design, Finite State Machine.*

INTRODUCTION

The rehabilitation of the Kumasi city roads has resulted in increased vehicular traffic in the metropolis. Vehicular traffic data have shown that the numerous intersections in the metropolitan road network have created bottlenecks to easy vehicular movement.

One way of improving the operational capacity of the intersections is to install traffic signals, which ensure safety and orderly flow of traffic as well as safety of pedestrians. Parameters that govern vehicular traffic flow depend, among others, on the inconsistencies of driver behavior and variations in traffic flow patterns. Traffic flow is therefore generally described as an uncertain, nonlinear, and time-varying dynamic system (Fu-Sheng and Petros, 1996; Akcelik and Besley, 2001). Its control, however, can be facilitated through the application of various technologies, including computers and smart sensors.

Signalization of an intersection generally requires good traffic data, optimal policy for the control of the signal, and appropriate hardware facility to implement the policy. The complexity of the hardware facility, and therefore its cost, depends on the control policy.

The SCOOT (Split Cycle Offset Optimization Technique) signal control system implements an adaptive control policy, where even short-term peaks in traffic demand call for new signal settings (DFT, 2003). It is particularly effective where traffic flows are unpredictable. Installation and maintenance of the computers and sensors that make SCOOT an intelligent closed-loop control system constitute a significant cost element in its deployment. Signal controllers that implement fixed-time policy do not respond to current traffic flow information and do often cause congestion during peak hours. They are described as open-loop non-adaptive control systems. Their installation and maintenance costs are, however, minimal.

A variant of the non-adaptive signal controller uses a series of fixed-time signal plans for different time zones of the day. Each signal plan has a predetermined signal setting. Optimal signal setting for each time zone is achievable if the mean rate at which traffic arrives is approximately constant (Salter, 1989). The controller cycles through the signal plans without taking into consideration disturbances in traffic flow. The control policy can be said to be semi-adaptive. In the absence of traffic disturbances, the performance of the semi-adaptive signal controller could approach that of the adaptive controller if optimal signal settings are used while installation and maintenance costs could be quite low compared to that of the fixed-time controller. Traffic data must, however, be collected seasonally in order to ensure optimal signal settings.

The Department of Urban Roads (DUR) carried out a comprehensive study of vehicular traffic flow in the Kumasi metropolis (DUR, 1995). The study focused on major intersections that lead traffic from the suburbs to the central business area of the metropolis. Results of the study show the occurrence of peak flows interspersed with 'shoulder' periods of reduced flow from 06:00 to 19:00 hours. Traffic flow outside this period was generally low. The general traffic flow pattern showed that the semi-adaptive control policy could be implemented to control traffic in the metropolis.

The work presented in this paper falls within the framework of developing an efficient and cost-effective hardware platform for the DUR to implement semi-adaptive traffic signal control policy.

The next section presents the development of the prototype electronic vehicular traffic signal controller. This is followed by results and discussions. Conclusions are presented in the last section.

METHODOLOGY

Signal controller design

An electronic traffic signal controller is basically a sequential machine whose operation can be modeled using finite state machine concept. A finite state machine (FSM) is a sequential system comprising states, inputs, and outputs. It models time as discrete instants at which input or output changes (Roth, 1995). Once a model is obtained in the form of a state diagram or state transition table, the next important issue is the choice of technology to implement the model. Technologies currently available include

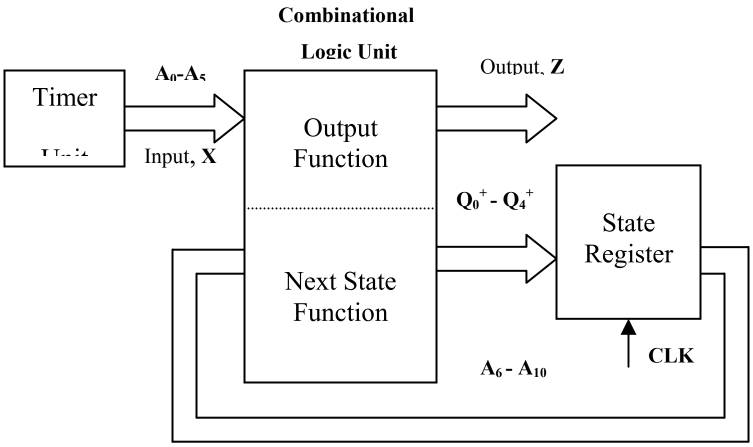
- Standard logic gates,
- Programmable Read Only Memories (PROMs),
- Complex Programmable Logic Devices (CPLDs),
- Field Programmable Gate Arrays (FPGAs), and
- Microprocessors/Micro-controllers.

Hardwired logic implementation using standard logic gates offers the least cost alternative but requires considerable design and circuit realization effort. In addition, it lacks flexibility when modifications to system operation are required. The microprocessor/micro-controller option, at the other extreme, offers the maximum flexibility since it operates under software control. However, the relative high cost of hardware/software development facilities and support chips required in its implementation makes it less attractive, particularly when the number of system states is reduced.

PROM implementation offers some degree of flexibility since the LSI device can readily be re-programmed to achieve system update. The implementation becomes more efficient when the number of system inputs is reduced and outputs large. However, the device size grows exponentially with increasing number of inputs.

A finite state machine that uses EPROM devices was designed based on the semi-adaptive signal control policy described above. The conceptual block diagram of the finite state machine is shown in Figure 1.

Figure 1. Conceptual block diagram of signal controller finite state machine



In this design, a timer unit driven by a crystal-controlled real-time clock delivers signals to the combinational logic unit in order to select a particular signal plan. A fixed-time signal plan assigns green interval to the different phases of an intersection for a pre-determined length of time during the *signal cycle*. A signal cycle comprises thirty-two *states*: S_0 – S_{31}, and is characterized by the *cycle time* (in seconds) and the duration of the green interval for each *phase* of the cycle. The duration of a state is one clock (CLK) period. Each signal plan is associated with a pre-determined clock period hence a unique cycle time.

For this state machine, the **Next state, Q^+** and the **output, Z,** are determined by the **Present state, Q** and the signal plan selection (input) signal, **X**. The relevant equations are written as

$$Q^+ = \boldsymbol{\delta}\,(Q, X), \text{ and}$$
$$Z \;= \boldsymbol{\lambda}\,(Q, X)$$

where **δ** and **λ** denote the next state and output functions respectively.

The symbolic state transition table for the signal controller state machine is shown in Table 1 for a selected signal plan, **X** = 01Hex. Each state, **S**, is coded on five bits (A_6 – A_{10}). The input, **X**, is coded on six bits (A_0 –A_5), giving a total number of sixty-four possible fixed-time signal plans. The output, **Z**, which is programmed to drive the traffic signal lamps, corresponds to the EPROM device 8-bit data word. Data bits D_0, D_1, D_2, and D_3 drive *Phase 1* pedestrian (P_1), green (G_1), amber (AM_1), and red (R_1) lamps respectively. Similarly, D_4, D_5, D_6, and D_7 drive *Phase 2* pedestrian (P_2), green (G_2), amber (AM_2), and red (R_2) lamps respectively.

Table 1: Symbolic State Transition Table

Present State, Q	**Next State, Q^+ Input X=01Hex**	**Output, Z**							
		D_7	**D_6**	**D_5**	**D_4**	**D_3**	**D_2**	**D_1**	**D_0**
		R_2	**AM_2**	**G_2**	**P_2**	**R_1**	**AM_1**	**G_1**	**P_1**
S_0	S_1	1	0	0	1	1	0	0	0
S_1	S_2	1	0	0	1	0	0	1	0
S_2	S_3	1	0	0	1	0	0	1	0
				…					
S_{17}	S_{18}	1	0	0	1	0	0	1	0
S_{18}	S_{19}	1	0	0	1	0	1	0	0
S_{19}	S_{20}	1	0	0	1	1	0	0	0
S_{20}	S_{21}	1	0	0	1	1	0	0	0
S_{21}	S_{22}	0	0	1	0	1	0	0	1
				…					
S_{28}	S_{29}	0	0	1	0	1	0	0	1
S_{29}	S_{30}	0	1	0	0	1	0	0	1
S_{30}	S_{31}	1	0	0	0	1	0	0	1

It is important to note that once a signal plan is selected for a particular time zone, the state machine sequences from state S_0 ($A_6 \ldots A_{10} = 00000_2$) to state S_{31} ($A_6 \ldots A_{10}=11111_2$) continuously. A change in the input variable, X, which signifies the selection of a new signal plan, is only recognized when state S_{31} transits to S_0. A change in the input variable, **X** is therefore not reflected in the output **Z** during a signal cycle. The finite state machine can thus be described as Moore sequential machine (Roth, 1995; Wakerly, 1994).

Practical realization and test

The signal controller was built around the Intel 2716(2kx8) EPROM device, which implements the combinational logic circuit. This LSI device is reliable in storing programmed data (Intel, 1986) and will ensure fail-safe operation of the signal controller, particularly against green-green conflict. A 5-bit binary counter, realized from two 4-bit 74LS193 synchronous counters, was used to implement the state register. This was possible because the 74LS193 chip integrates clocked D-type flip-flops whose outputs hold the **Present state** during a clock period and advances to the **Next state** when a clock transition occurs. No physical connections were therefore made between Q_0^+- Q_4^+ and the state register inputs as indicated in Figure 1. However, the output bits of the counter were connected to EPROM address lines $A_6 – A_{10}$. This approach presented a significant advantage: all the EPROM device outputs could be used to drive the traffic signal lamps without the need for additional output lines.

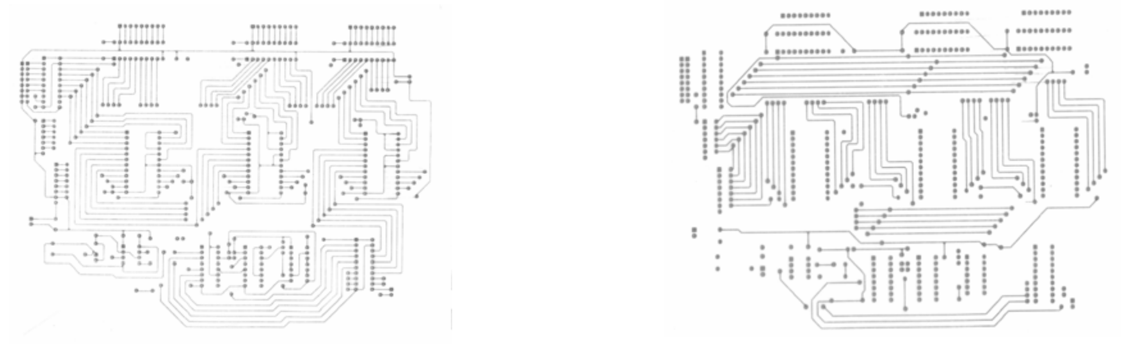
The 5-bit counter sequences through the thirty-two states ($S_0 – S_{31}$) to define one signal cycle. A reset pulse is generated whenever state S_{31} transits to state S_0. This starts a new cycle for the selected signal plan. The reset pulse also served as a gating or synchronizing signal for the signal plan selecting signals from the timer board, which were generated asynchronously vis-à-vis the state machine clock (CLK). The gating of a new input, X, marks the selection of a new signal plan for a particular time zone.

A printed-circuit-board (pcb) for the signal controller was designed with provision for two additional EPROM devices. The controller can therefore be configured for a maximum of six phases. The pcb layout developed using the Wintek Smartwork software is shown in Figure 2 for the finite state machine module. The signal lamps were driven by solid-state relays. The use of electromechanical relays, though cheaper, resulted in the malfunction of the controller due to electrical noise. Ordinary 60-watt incandescent bulbs were used for the signal lamps in order to reduce cost.

The timer board was battery-backed whereas the main controller board was not. In this way, input signals from the timer module were always on even during mains power failure when signal lamps were off.

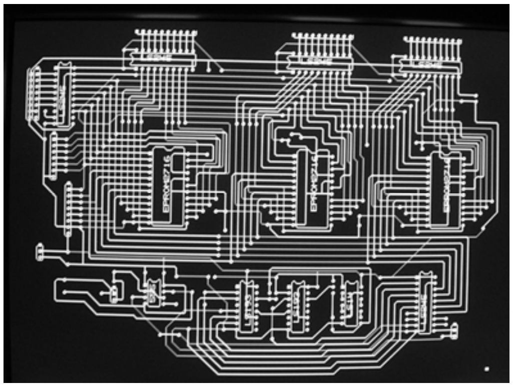
For test purposes, the controller was configured for an intersection in the metropolis called Texas, which is a two-phase intersection. This intersection was un-signalized and was beset with frequent vehicular collisions. Six signal plans (Table 2) were programmed into a single 2716 EPROM device using the Intel iUP-200A/201A Universal EPROM Programmer. The EPROM device could be reprogrammed when a signal plan update is required.

Fig. 2: PCB Layout



SOLDER LAYER

COMPONENT LAYER



COMPLETE LAYOUT

Table 2: Signal Plan for Texas Intersection

Signal Plan	1	2	3	4	5	6
Time Zone	06^{00}-07^{10}	07^{10}-10^{25}	10^{25}-14^{55}	14^{55}-20^{00}	20^{00}-22^{43}	22^{43}-06^{00}
Phase 1 Green interval in secs (Number of states)	54.0(18)	70.3(19)	66.6(18)	54.0(18)	54.0(18)	
Phase 2 Green interval in secs (Number of states)	24.0(8)	25.9(7)	29.6(8)	24.0(8)	24.0(8)	
Clock Period in secs.	3.0	3.7	3.7	3.0	3.0	1.0
Cycle time in secs.	96.0	118.4	118.4	96.0	96.0	32.0
% degree of saturation(avg)	0.72	0.93	0.93	0.80	0.80	

On power-on, the binary counter resets to the zero state and the controller enters **Present state**, S_0. The signal plan corresponding to the time of day is selected and red lamps R_1 and R_2 switched on to stop vehicular traffic flow on all approaches of the intersection. This state corresponds to the **Intergreen** period. The next clock transition sends the controller to state S_1 of the signal cycle when green lamp G_1 switches on to permit traffic flow; red lamp R_2 remains lit. After sequencing through the number of states assigned to the green period of the current phase, the controller switches on amber lamp AM_1 for one state and then enters another intergreen period for two additional states. The next phase of the cycle starts when green lamp G_2 switches on to permit traffic flow on the other approache(s) associated with the new phase. The controller repeats the cycle until a new signal plan is selected. The signal plan for the period 22:43 – 06:00 hours alternates the switching action of amber lamps AM_1 and AM_2. This is the **flashing amber** program. Since the timer unit constitutes a separate module, it can be unplugged from the controller board. The controller will operate independently on a single signal plan selected using a hardware jumper.

Field test was carried out at the Texas intersection after three months of laboratory test.

RESULTS AND DISCUSSION

The performance of the signal controller in improving operational capacity of an intersection was evaluated in terms of

Efficiency with which the controller hardware facility implemented the control algorithm (semi-adaptive control policy) and

Quality of traffic data from which optimal signal settings (cycle time and green intervals) were derived.

Controller hardware facility

A digital stopwatch was used to verify programmed cycle times and green intervals. The verification process was repeated every other week for three months in the laboratory and for an additional two months in the field. The time of day at which the crystal-controlled real-time clock initiated execution of the first signal plan of the day and the order in which signal plans were executed were also monitored over the period.

The maximum deviation in programmed cycle times and green intervals during the five- month test period was found to be less than 0.3%. This level of precision was expected since wire-wound precision resistors (0.1% tolerance and ±3ppm/^{o}C temperature coefficient) were used in realizing the clock (CLK) of the state machine.

The real-time clock experienced a delay of one second after four months in operation. As a result, the execution of the first signal plan was initiated at 06:01 am. The clock was simply reset to correct the time. Signal plans were executed consecutively as designed.

No green-green conflict in signal phases (simultaneous green intervals) was observed. Consequently, no vehicular collision occurred as a result of controller malfunction.

Effect of signal settings on traffic flow

The DUR used the SIGNAL85 off-line computer software package to determine optimal signal settings for the field test at Texas intersection. The values are shown in Table 2. Traffic flow observations carried out over the period showed approximately 15% reduction in queue length on the heavily trafficked Ahinsan approach. The minimal improvement in queue length reduction was the outcome of high level of saturation flow, which meant low practical reserved capacity of the intersection.

Cost analysis

The list of components (with their unit prices) used in realizing the prototype signal controller is shown in Table 3. Components were purchased from MAPLIN ELECTRONICS Ltd (Maplin, 1995). The finite state machine module, which is the core unit of the controller, cost about £30 while the timer unit cost about £20, giving a total of £50.

CONCLUSION

A prototype electronic signal controller was developed to implement semi-adaptive traffic control policy. The design was based on finite state machine concept and the hardware architecture permits the implementation of sixty-four signal plans per day and six phases at an intersection. A time-slice of 22.5 minutes per signal plan facilitates the capture of fine variations in traffic flow. The controller permits update of signal settings as well as signal phases by reprogramming the EPROM device(s). The application of PROM technology makes the controller less costly yet reliable compared with a microprocessor/microcontroller-based controller. Laboratory and field tests showed less than 0.3% deviation in programmed signal settings, which is acceptable for the control policy implemented.

Table 3: Cost Analysis of Signal Controller

Finite state machine module			
Component	**Unit price (£)**	**Quantity**	**Total (£)**
2716 EPROM device	5.097	3	15.291
74LS193 counter	0.672	2	1.344
NE555N timer	0.650	1	0.650
74LS245 octal bus transceiver	0.672	4	2.688
74LS374 clocked octal D latch	0.587	1	0.587
74LS14 Hex Schmitt-trigger inverter	0.289	1	0.289
Misc (resistors, capacitors, connectors, and pcb)			8.50
Total			29.349

Component	Timer unit Unit Price (£)	Qty	Total (£)
Real-time LCD clock module	4.460	1	4.460
NPN transistor 2N2222	0.20	1	0.20
UA2240CN timer	2.40	1	4.40
74LS164 shift register	0.757	2	1.514
74LS86 XOR gate	0.398	2	0.796
74LS14 Hex Schmitt-trigger inverter	0.289	1	0.289
74LS74 D-type FF	0.417	1	0.417
Misc (resistors, capacitors, connectors, and pcb)			7.20
Total			19.276

Field observations at one of the intersections in the metropolis showed a minimal 15% reduction in vehicular queue length for the programmed signal settings. The performance of the controller in terms of intersection capacity improvement therefore depends on efficient hardware platform as well as reliable traffic flow data and the determination of optimal signal settings.

ACKNOWLEDGEMENTS
The author would like to thank the following for their invaluable suggestions and assistance: Professor E.A Jackson and staff (academic and technical) of the Department of Electrical and Electronic Engineering, F.K Afukarr of the Building and Road Research Institute of the CSIR, Kome-Mensah and Amoo Gottfried of the DUR. The project was executed under DUR contract number KDRU/AR/TS/1/91.

REFERENCES

Fu-Sheng, H. and Petros, I. (1996). Traffic Flow Modeling and Control Using Artificial Neural Networks. IEEE Control Systems, 16(5), 16-26.

Akcelik, R. and Besley M. (2001). Micro-simulation and analytical methods for modeling urban traffic. Akcelik & Associates Pty Ltd. http://www.aatraffic.com

Department for Transport (2003). The SCOOT urban traffic control system. http://www. dft.gov.uk.

Salter, R.J (1989). Highway Traffic Analysis and Design, 2^e, Macmillan Education Ltd., Hampshire R G21 2XS, pages 291-297.

Department of Urban Roads, Kumasi (1995). Transportation Demand Studies Report. August, 1995.

Roth, C.H (1995). Fundamentals of Logic Design, 3^e, West Publishing Company, St. Paul, Minnesota, USA, pages 308 -311.

Wakerly, J.F. (1994). DIGITAL DESIGN *Principles & Practices*, 2^e, Prentice Hall, Englewood Cliffs, New Jersey, USA, pages 467 – 469.

Intel Components Quality/Reliability Handbook (1986). Santa Clara, USA.

MAPLIN ELECTRONIC Catalogue, (1995). Rayleigh, Essex SS6 8LU, UK.