State-of-the-art feature project

Electronics Australia
JULY 1982

CAR COMPUTER PART ONE

With this Car Computer you will have immediate moment-to-moment feedback on the effect of your driving habits on fuel consumption. You will be able to drive your car at optimum efficiency for all driving conditions and make a worthwhile contribution to energy conservation.

by LEO SIMPSON and JOHN CLARKE

At present, there are only a few cars which can be purchased with a dashboard computer and there is only one car computer, that we know of, which can be fitted as an accessory. That is about to change, particularly now that "Electronics Australia" has designed this computer to suit locally available cars and components. That means this computer is equally suited to measuring the fuel consumption of gas guzzlers and sippers – the economical four-cylinder cars which are becoming ever more popular.

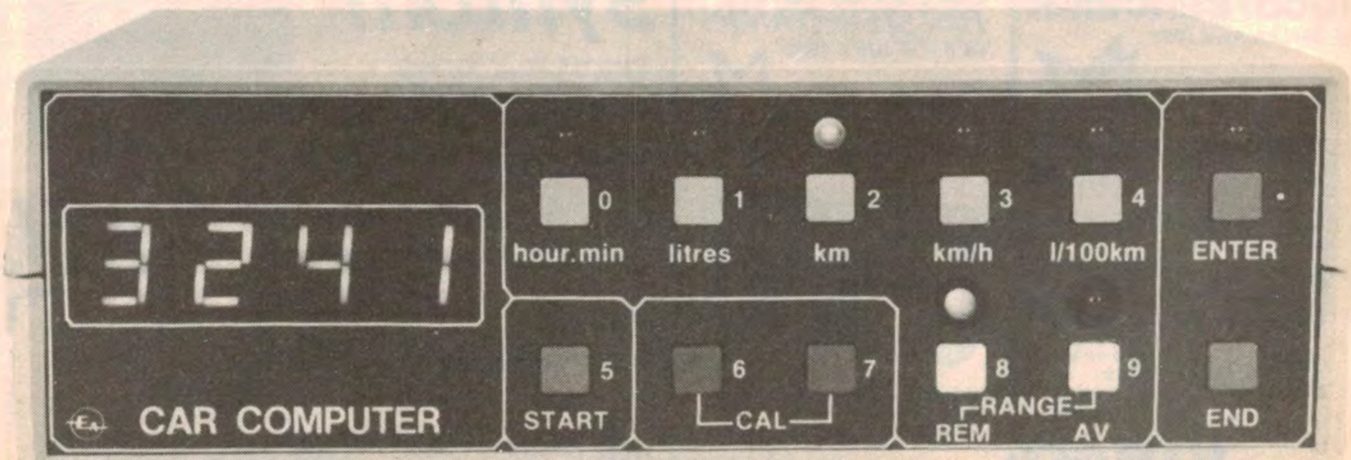
The EA Car Computer uses a microprocessor and other supporting integrated circuits to keep track of three parameters: *time, distance* and *petrol flow*. To keep track of time, the car computer has its own crystal controlled clock. To keep track of distance, there is a sensor which monitors the number of revolutions of the drive-shaft or speedometer cable. And to keep track of petrol flow there is a fluid flow sensor which can measure flow rates down to as little as one litre per hour! This rate of flow is roughly equivalent to that from a fast dripping tap! It is necessary to be able to measure this very low rate if accurate fuel consumption of small four-cylinder cars is to be recorded.

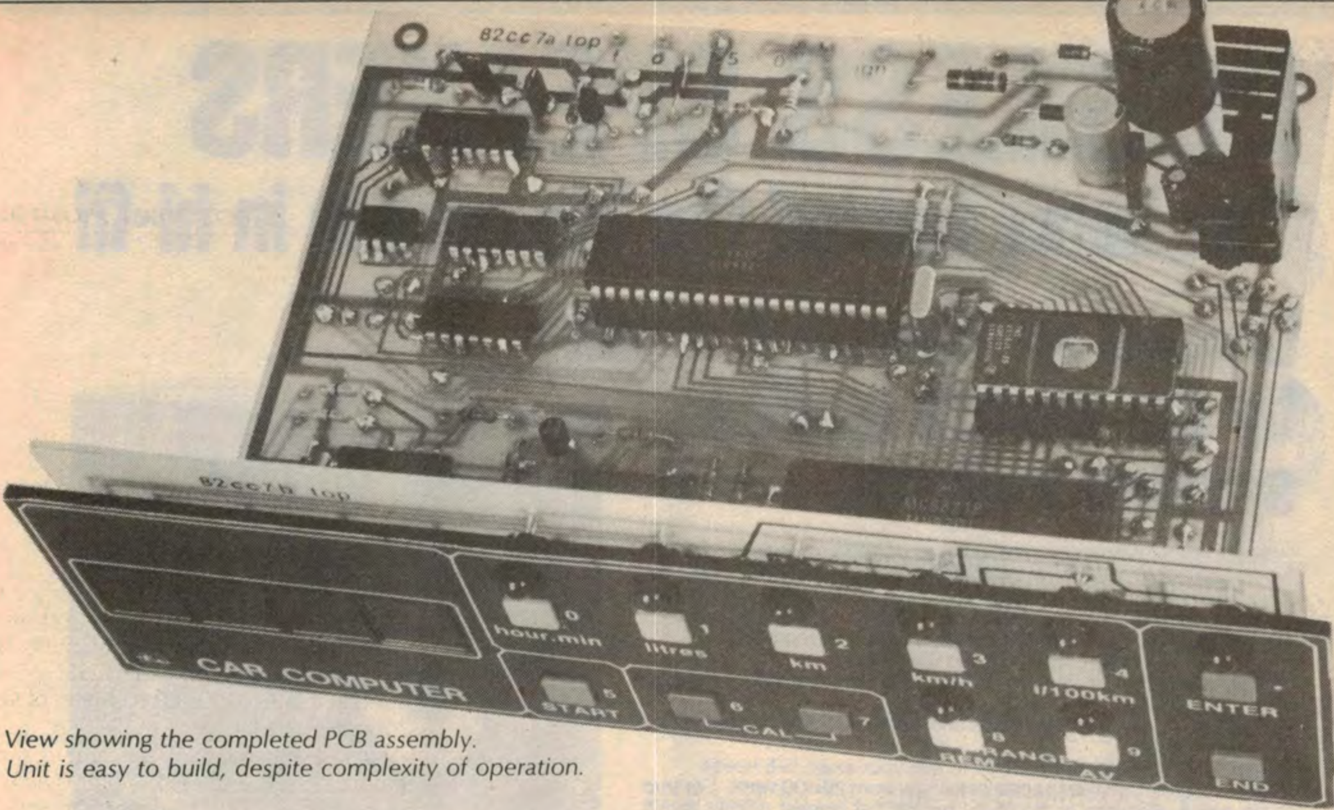
Depending on the type used, the fuel sensor can be one of two forms. One is a miniature turbine with a vane which interrupts a beam of light to a phototransistor. The other type uses a ball running in a circular race to also interrupt a beam of light to a phototransistor.

There are also two types of sensor for distance. One uses a coil placed close to magnets on the spinning driveshaft. The other uses a vane attached to the speedo cable to interrupt a beam of light to a phototransistor.

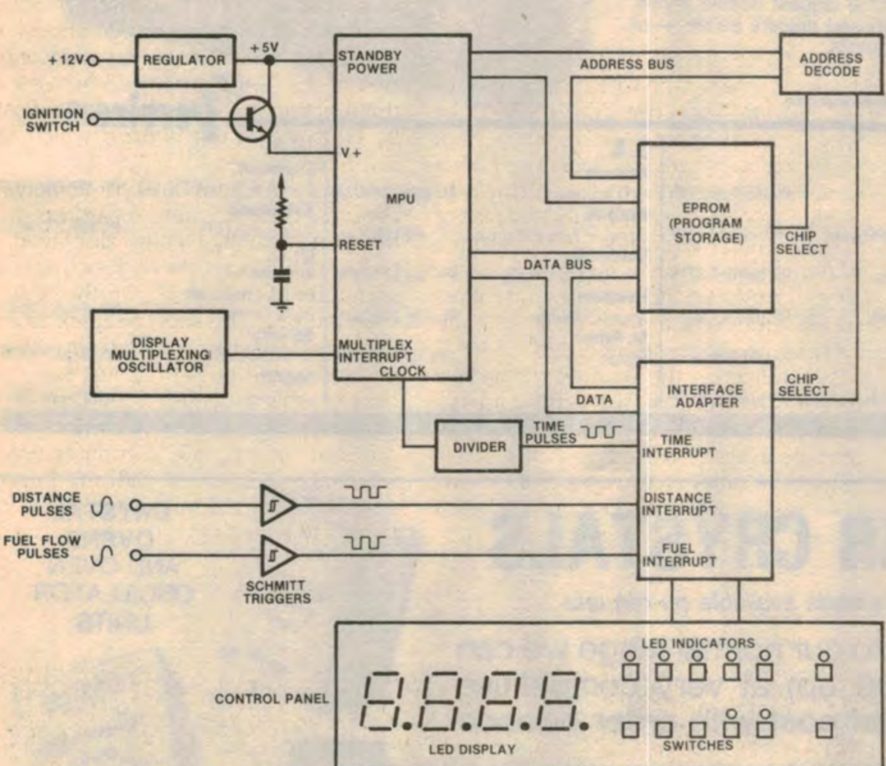
By keeping track of the above three parameters, the Car Computer is able to give readings of 11 separate functions from its four-digit seven-segment display. These are called up by pressing one or two of the 12 colour-keyed buttons and the function being displayed is indicated by LEDs above the buttons which have just been pushed.

All the functions are continually updated by the computer, regardless of the function actually being displayed. In order to display these functions, the computer is initiallised at the start of each journey. This sets the clock to zero. The length of the journey to be taken is entered and if petrol has just been purchased, this is entered in too. If the journey to be taken is identical to the previous journey the computer will be automatically initiated with this information, since its memory circuits are permanently energised.





View showing the completed PCB assembly.
Unit is easy to build, despite complexity of operation.



This block diagram shows the general concept of the Car Computer.

Functions

Twelve separate values can be displayed, as follows:

- hour. min – elapsed time. This is the time in hours and minutes since the start of the journey. The reading is updated every minute and the decimal point after the hours digit flashes at a one second rate.

This is called up by pressing button "0".

- hour. min REM – time remaining. This is the time in hours and minutes which will be required to complete the journey at the present average speed recorded since the start of the journey. Again, the decimal point flashes at a one second rate and the reading is updated with every kilometre covered or every minute. This is called up by pressing buttons "0" and "8".
- litres – fuel used. This is the amount of fuel consumed since the start of the journey. This is displayed in litres with 0.1 litre resolution. The reading is updated for every 0.1 litres of fuel consumed. This is called uyp by pressing button "1".
- litres REM – fuel remaining. This, as you might expect, is the amount of fuel remaining in the tank, not allowing for losses by leakage or evaporation. Again it is displayed in litres with 0.1 litre resolution and updated for every 0.1 litres of fuel consumed. This is called up by pressing buttons "1" and "8".
- litres REM RANGE – capacity of fuel tank in litres. This is called up by pressing buttons "1" and "9".
- km – distance travelled. This is the distance travelled since the "START" button was pressed. This can record a maximum trip length up to 9999 kilometres over a period of several days, weeks or months, as this information is stored whether the ignition is on or not. The reading is updated with every kilometre travelled and is called up by pressing button "2".
- km. REM – distançe remaining of journey. Updated every kilometre travelled and called up by pressing buttons "2" and "8".
- km REM RANGE – distance that can

CAR COMPUTER

be travelled in kilometres. This is based on the number of litres left in the fuel tank and on the average fuel consumption since the beginning of the journey. This is updated for every kilometre travelled or for every 0.1 litre of fuel used. Called up by pressing buttons "2" and "9".

- km/h – speed in kilometres per hour. This is updated every one second and is called up by pressing button "3".
- km/h AV – average speed. This is based on the elapsed time of the journey and distance travelled, since pushing the Start button. This is updated every minute or every kilometre travelled.
- l/100km – instantaneous fuel consumption. Gives the fuel consumption for every one or eight-second period, depending on the fuel sensor used. Press button "4".
- l/100km AV – average fuel consumption for journey, based on fuel used so far and distance travelled. Updated every kilometre travelled or 0.1 litres used. Press buttons "4" and "9".

Data entry

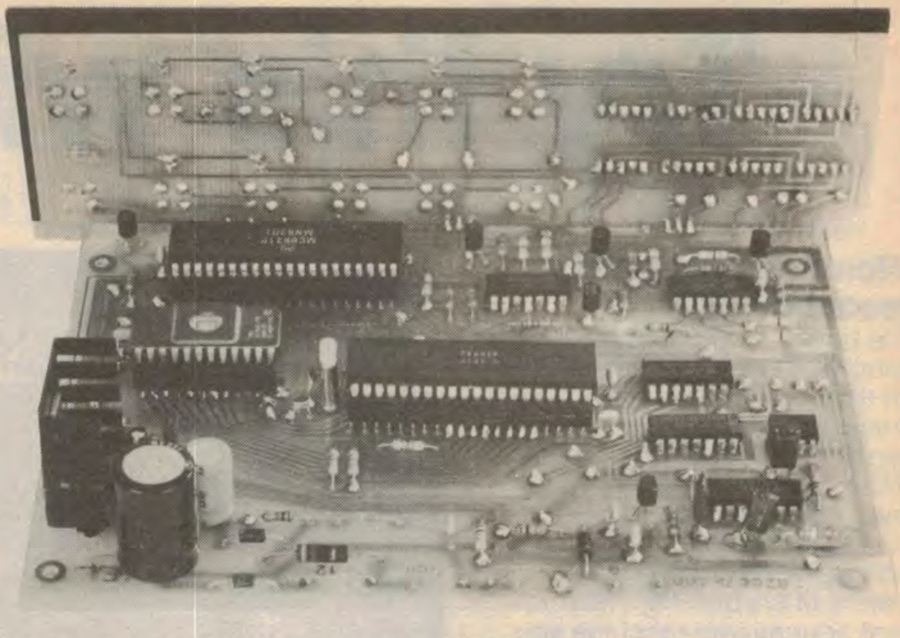
Data is entered into the Car Computer by pushing three buttons, START, ENTER and END. When the START button is pressed the computer displays "rEdY" and zeros the following functions: elapsed time, fuel used and distance travelled. It also enters in the previous journey, ie, km REM.

To change data in the computer, such as the amount of fuel in the tank, the ENTER button is pressed and this changes the function of all buttons (except END) to numerical data entry. The ENTER button itself is the decimal point.

When you have the correct data shown on the display, pushing the END button loads it and reverts all the buttons to their normal functions. If you have made an error in your data values, such as not entering the decimal point for the fuel quantity, the data will not be entered when you press END and the display will show "F. Err" which signifies an error in the fuel quantity. Brilliant, isn't it?

Calibration

Two buttons are provided for calibrating the sensors. Button "6" calibrates the fuel sensor (using the ENTER procedure briefly described above). Here the user enters the manufacturer's stated number of pulses per litre.



All the circuitry is accommodated on two double-sided printed circuit boards which are soldered together at right angles. Apart from connections to the sensors and battery, there is no discrete wiring.

Button "7" calibrates the distance sensor and this is done by trial and error between kilometre posts during an on-road test. Thus the Car Computer is not subject to the vagaries of normal car speedometers. Note though that the Car Computer does not take into account the varying effects of tyre slip – this can only be accounted for by using a fifth wheel.

Presentation

The Car Computer is housed in a compact and smart cabinet which will look well on or in the dash or console of any car. The front panel is black with labelling in white for easy legibility. The LED readouts have integral red filters for ease of visibility in high ambient light.

Inside there is almost no wiring at all with all the circuitry accommodated on two double-sided PC boards. A vertical board accommodates the LED readouts, eight LEDS and 12 pushbuttons while the larger horizontal board accommodates the integrated circuits.

All the connections from the Car Computer to the car battery and external sensors are made via a quickly detachable multi-way plug and socket. In fact, if you were so inclined, it would be possible to transfer the Computer from one vehicle to another, provided each vehicle was fitted with sensors.

Hardware

The total semiconductor complement is really quite small, as can be seen from the accompanying photographs. Apart from the previously noted four LED readouts and eight LEDs, there are three major integrated circuits and six minor, one 5V regulator, seven transistors and four diodes.

The block diagram shows the general concept of the Car Computer. The eight-bit microprocessor is the Motorola 6802 which is a variant of the well-known 6800 which has 72 instructions and six different addressing modes (see the series on "How to Program in Machine Language" beginning March 1982). The 6802 has all the facilities of the 6800 and has a built-in clock and a divide-by-four circuit to allow an external 4MHz clock to be used. In our particular case, the clock runs at 3.579MHz. Also incorporated into the 6802 is 128 bytes of RAM and the first 32 bytes of this memory may be operated in a low power mode to prevent loss of data when normal power is off (power down).

Teamed with the 6802 processor is the 6821 peripheral interface adapter which has two 8-bit bidirectional data buses and four control lines. This device scans the front panel push-buttons and the fuel and distance sensors for input signals and drives the LED readout in multiplex mode.

The machine language program for the Car Computer is stored in a 2716 2048-byte EPROM (Electrically Programmable Read Only Memory).

Next month we shall give the circuit and software description plus details of construction. Don't miss out on your copy of the August issue.

CAR COMPUTER

CAR COMPUTER PART TWO

Electronics Australia
AUGUST 1982

Although the operation of our microprocessor-based Car Computer is quite involved, most of the complexity is concealed within three major integrated circuits. Construction is therefore relatively simple. This month we give the circuit and software details and describe construction.

by JOHN CLARKE

By far the most important integrated circuit used in the Car Computer is IC1, a Motorola 6802 8-bit microprocessor (MPU). This microprocessor can perform all the instructions of the well-known 6800 microprocessor and contains 128 bytes of volatile Random Access Memory (RAM), which can be used for data storage when running a program. Additional to this is the advantage that the first 32 bytes (8-bits wide) of RAM is separately powered, enabling important information to be retained when power to the main processor is switched off.

IC2, the Peripheral Interface Adaptor (PIA), supports the external hardware devices. In our circuit, IC2 is used to drive (write) the LED display, read the function select switches and detect the distance, fuel and time pulses. Two 8-bit ports, PA0 to PA7 and PB0 to PB7, are available and can be programmed as inputs (read) or outputs (write). On each port are two extra lines, CA1 and CA2 and CB1 and CB2 respectively. CA1 and CB1 are inputs and CA2 and CB2 can be programmed as either inputs or outputs.

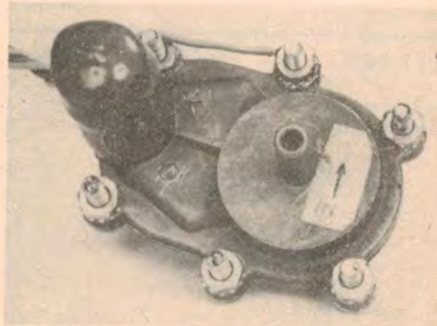
IC3 is a 2K byte Electrically Programmable Read Only Memory (EPROM), which holds the program for the Car Computer. This memory is non-volatile, which means that switching off the power to the IC will not erase the memory. The program remains stored indefinitely (unless erased by ultra-violet light).

A common 8-bit data bus interconnects IC1, IC2 and IC3. This provides two-way (Read/Write) communication between these devices. An address bus (A0 to A10) connects IC1 and IC3 and this is used to access all the EPROM locations.

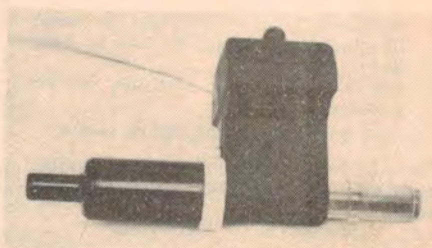
Simple address decoding for IC2 and IC3 is performed by two NAND gates, IC6a and IC6b. IC3 is accessible to IC1 when both the VMA and A14 lines are high, and IC6a brings $\overline{CE}$ of IC3 low. The VMA line, or Valid Memory Address line, indicates that a valid address is on the bus. The memory locations for the EPROM are from hexadecimal number 6000 to 67FF.

All the registers of IC2 are accessible by high, low combinations of RS0 and RS1, which are connected to A0 and A2 respectively. However, these registers cannot be selected unless the Chip Select lines CS0, CS1 and $\overline{CS2}$ are true. CS1 is permanently held high and CS0 is connected to the $\overline{CE}$ if IC3. When A15 and the VMA are both high, IC6b brings $\overline{CS2}$ low. Providing that IC3 is not selected with A14 high, then IC2 is selected. We used addresses from 8004 to 8007 to access the PIA.

One point of interest here is why the $\overline{CE}$ of IC3 has been connected to CS0 of IC2 to prevent both ICs being selected at the same time. The only time that this conflict could occur is when both A14 and A15 are high. Why not simply avoid addresses at and above C000 in the program? To understand this, it is necessary to further discuss the operation of IC1.



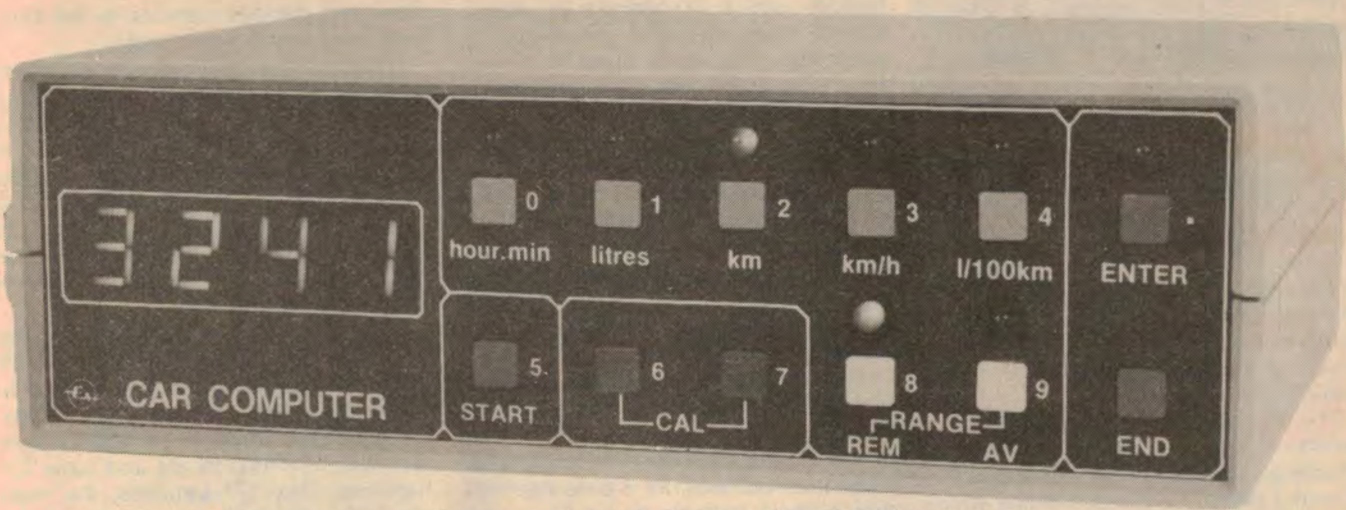
The Prince fuel flow sensor delivers 130 pulses per 0.1 litres of fuel flow.



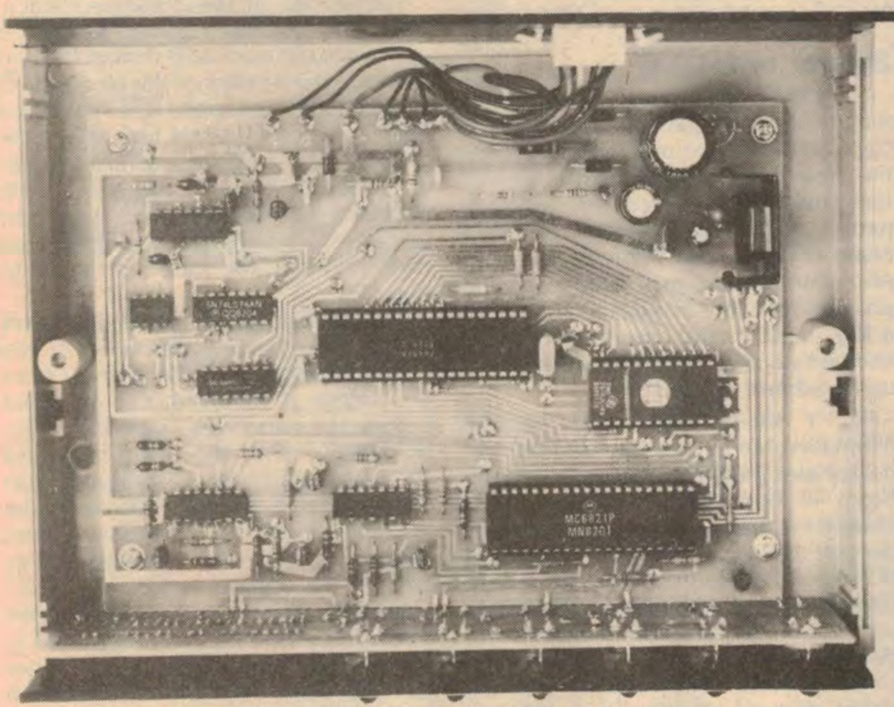
The alternative Moray fuel flow sensor delivers 1500 pulses per 0.1 litres, and is the unit we recommend.

Three programming levels are used in the Car Computer: RESET, Non Maskable Interrupt (NMI), and Interrupt Request (IRQ). The initiation of each of these programs is determined by the voltage levels (or edge triggering in the case of the NMI) on the respective hardware pins of IC1, pins 40, 6 and 4.

We shall discuss how the voltage levels at these pins are controlled and why we have used these levels of programs at a later stage. At present it is sufficient to say that to access the start address in the EPROM for these programs, IC1 looks at addresses between FFF8 and FFFF. This means that both A14 and A15 will be



Just 3241km to go! Car Computer should mate well with the interior of most modern cars.



View inside the assembled Car Computer. IC sockets are mandatory for the three main ICs, optional for others. Note clip-on heatsink fitted to the 7805 regulator.

high. Since we only want to access IC3 and not IC2, IC2 is disabled by the $\overline{CE}$ to CSO connection.

The advantage of using these program levels is that each has its own priority. At first power on, the Car Computer runs the "background" program initiated by the power-on RESET. This program continues to run until interrupted by either the $\overline{NMI}$ or $\overline{IRQ}$. If the $\overline{IRQ}$ pin were to go low, the processor finishes the current instructions and begins the IRQ program routine. Once this program is complete, the processor continues with the background program as though no interrupt had occurred.

If an $\overline{NMI}$ occurs during an IRQ routine, the processor immediately carries out the NMI routine, then reverts to IRQ and finally resumes the background program. The converse is not true, however – ie, if an $\overline{IRQ}$ occurs during NMI, the IRQ program is not run until the completion of the NMI routine.

Display multiplexing

This first priority of the NMI is put to good use by using it to multiplex the display. The display consists of four common anode 7-segment display digits and eight individual LEDs. The cathode of each display digit segment is connected to the corresponding cathode on the adjacent display, while the anodes of the function LEDs are also common. In order to light the display segments and the LEDs, it is necessary to switch on driver transistors Q1-Q5 at the anodes.

Each display is lit in turn and this is repeated at such a fast rate that the eye perceives a continuous display free from flicker. PA0 to PA7 on the PIA are used to send the correct segment of the display low at the appropriate time, while PB3 to PB7 scan the common anodes of the display digits. Note that 7404 inverting buffers, IC8 and IC9, are used to drive the cathodes of the display and the bases of the anode driving transistors. These buffers are necessary because the PIA output lines are incapable of supplying the necessary current.

Note also that the key switches are tied in matrix form to the digit scanning PIA outputs. If any switch is closed, this is read as a high signal on the PB0 to PB2 lines which are programmed as inputs. The rate at which these keys are scanned, and consequently the scanning rate of the display, is determined by the Schmitt trigger oscillator IC5a, which runs at close to 600Hz.

IC6c gates IC5a, allowing $\overline{NMI}$ to only occur when both CB2 and the IC5a oscillator output are high. Normally CB2 is high; however, during the initial stages of the RESET program (background program), initialisation must be completed before the NMI is allowed to proceed. The initialisation includes setting up the PIA lines as inputs or outputs and for interrupt inputs. This can

CAR COMPUTER

be seen on the flowchart program beginning with RESET (Flowchart 1). Setting the CB2 line high occurs after the "B" connection point.

Flowchart 1 also shows what happens when the NMI routine begins. Firstly, leading zeros in the display memory are suppressed. Following this, a check is made to see if a switch is closed and the next display digit is lit.

The IRQ performs three functions, and is interrupted when either a time pulse, distance pulse or fuel pulse occurs. So that the processor can determine which of the three inputs actually caused the interrupt (after all the processor only has one $\overline{IRQ}$ pin), it is necessary to send the signals via the PIA. The PIA sets flag bits within the PIA registers which correspond to the input causing the interrupt. The hardware outputs, $\overline{IRQA}$ and $\overline{IRQB}$, then interrupt the $\overline{IRQ}$ pin of the processor.

CA1 and CA2 on the A half of the PIA are used to detect the pulses from the distance and fuel sensors respectively. CB1 on the B half detects the time interrupt.

When negative edges at CA1, CA2 or CB1 occur, the $\overline{IRQA}$ and $\overline{IRQB}$ signals, which are tied together, go low and trigger the $\overline{IRQ}$ input of IC1. The $\overline{IRQ}$ line only goes high when all interrupt flags within the PIA registers are cleared.

The time interrupt is derived from a 3.58MHz crystal connected directly to the crystal input pins (pins 38 and 39) of IC1. Inside IC1 is a divide by four circuit which provides the E clock (pin 37), and

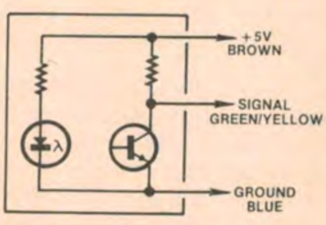


Fig. 1: MORAY FUEL SENSOR AND SPEEDOMETER CABLE DISTANCE SENSOR

this is used for timing the entire program. The resulting 894.9kHz E clock pulse is buffered by IC6d and fed to the input of IC7. IC7, an MM5369 divider, is designed to provide a 60Hz signal when operated from a 3.58MHz crystal. Since we are operating it from a crystal-controlled source of one quarter this frequency, the output at pin 1 has a frequency of 15Hz.

The 15Hz signal is connected to the data input (pin 12) of IC4b, a 74LS74 D flipflop, and is transferred to the Q output when the clock input, pin 11, goes high. This clock signal is derived from IC6c and is also connected to the $\overline{NMI}$ of IC1. The NMI routine occurs on the negative edge of the $\overline{NMI}$ clock and is completed well before the $\overline{NMI}$ clock goes high, which is the only time that the $\overline{IRQB}$ time signal can interrupt the $\overline{IRQ}$ of the processor. This arrangement is necessary since the NMI routine contains an instruction which will clear the $\overline{IRQB}$ interrupt; the very instruction which reads the key switches.

Flowchart 2 shows the logic operations which occur on an $\overline{IRQ}$. Firstly, a check is made to see if a time interrupt has occurred and, if so, the time is updated. If the interrupt was not a time interrupt then a check is made to see if it was a distance or fuel interrupt, or both. The corresponding distance and/or fuel reading is then incremented.

Sensor operation

As indicated last month, two different fuel flow sensors can be used with the Car Computer. The unit represented on the main circuit diagram is the "Prince" fuel flow sensor and consists of a ball running in a circular race to interrupt a beam of light from a small bulb to a phototransistor. Collector current for the phototransistor is derived via a 4.7kΩ resistor and the ouput signal filtered by a .01μF capacitor and squared up by Schmitt trigger IC5e.

Fig. 1 shows the circuit for the alternative Moray fuel flow sensor. This device uses multiple vanes to interrupt a beam of light between a LED and a phototransistor, an arrangement which delivers 11½ times more pulses per litre than the Prince sensor. The only change necessary to accommodate the Moray sensor is that the 4.7kΩ resistor be deleted from circuit.

The distance sensor shown on the circuit diagram consists of a coil and rotating magnet assembly. As the magnets rotate, they induce a voltage in the coil. This signal is half-wave rectified by a 1N4002 diode and filtered with a 0.1μF capacitor and 100kΩ resistor. A BC549 transistor provides the necessary gain and, after further filtering by a 0.1μF capacitor, the resulting waveform is squared up by Schmitt trigger IC5d.

The alternative speedometer cable sensor uses the same circuit configuration as the Moray fuel sensor (Fig. 1). In this case, however, the distance sensor signal is applied directly to pin 11 of IC5d and the diode, 0.1μF capacitor, and 100kΩ and 56kΩ resistors deleted. The .01μF capacitor should be left in circuit.

We'll have more to say about the fuel flow and distance sensors in the third (and final) article next month.

Power for the Car Computer is derived from the 12V car battery. A diode and 1000μF capacitor filter the battery voltage and a 7805 three-terminal regulator supplies +5V directly to the Vcc standby of IC1. Thus, power is permanently supplied to the first 32 bytes of RAM. The 10μF tantalum and 10μF electrolytic capacitors ensure stability of the regulator.

Transistor Q7 is used to switch the power to the main circuit on and off under the control of the ignition switch. When the ignition switch is turned on, current flows through an 82Ω resistor and series 1N4002 diode and turns Q7 hard on. Since Q7 saturates, the main circuit is effectively connected to the +5V output of the three terminal regulator.

With power on, the crystal oscillator in IC1 starts and the resulting $\overline{E}$ signal is applied to the clock of D flipflop IC4a. At the first positive edge of this clock, the $\overline{Q}$ output connected to the $\overline{RESET}$ and RAM Enable (RE) of IC1 is set low. When the 1MΩ resistor charges the 0.1uF capacitor at the input of Schmitt trigger IC5c, the output of IC5c goes low and, at the next positive clock transition, the $\overline{RESET}$ goes high, allowing the RESET program to begin.

When the ignition is turned off, power to the 82Ω resistor is disconnected and the current driving the base of Q7 from this source is removed. Q7 does not cease conduction immediately, however, due to the 100μF capacitor connected to its base. This capacitor can only discharge through Q7, since the associated 1N4002 diode is now reverse biased. During this discharge time, power is still applied to the circuit.

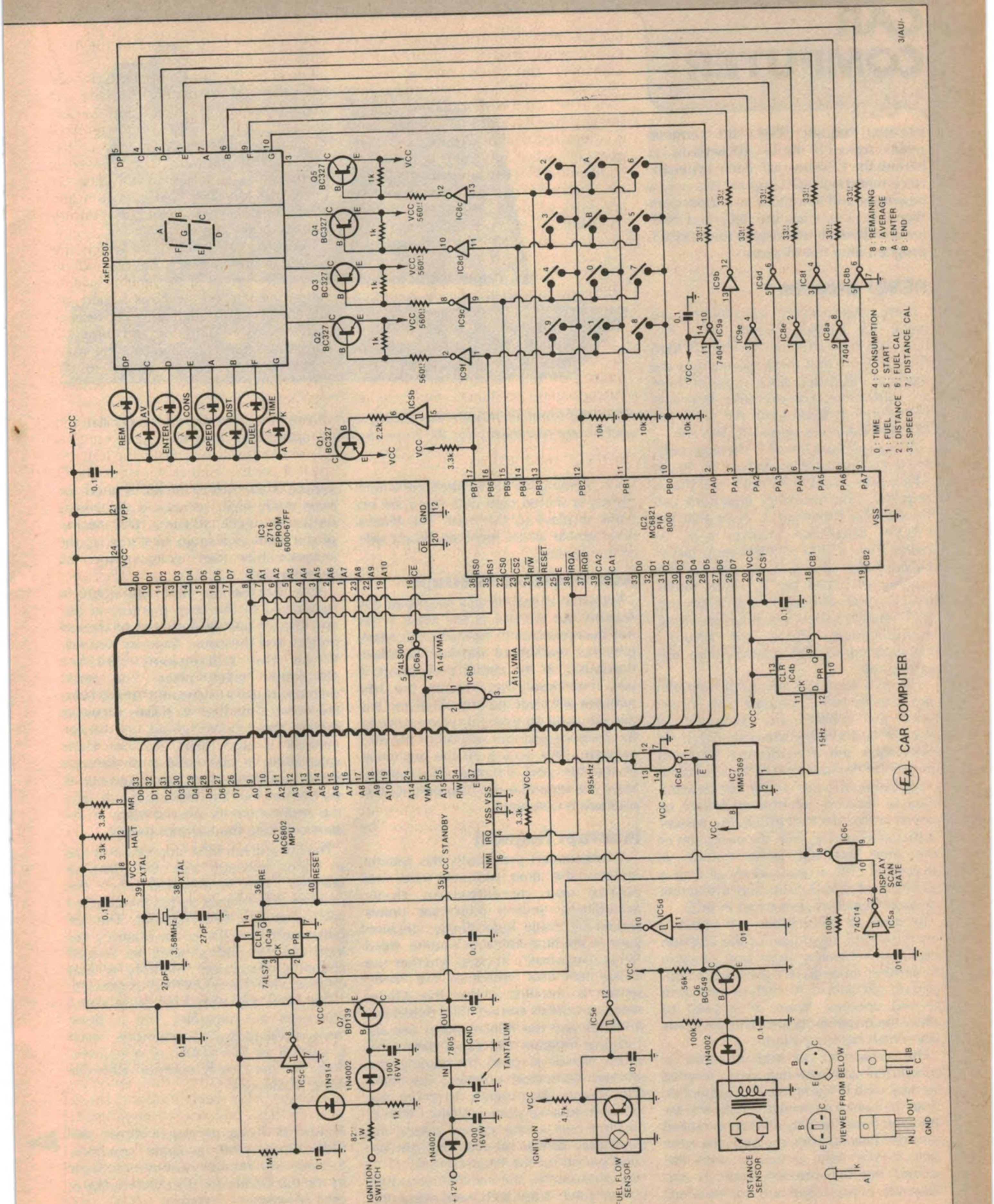
However, the 0.1μF capacitor at the input of IC5c is rapidly discharged at the moment of switch off via a series 1N4148 diode and 1kΩ resistor to ground. This sets the output of IC5c high and, at the next positive edge of the $\overline{E}$ pulse (IC1, pin 37), both RE and $\overline{RESET}$ go low.

This power down sequence ensures that memory in the first 32K bytes of RAM (which are permanently powered) is not corrupted at this critical stage.

The software

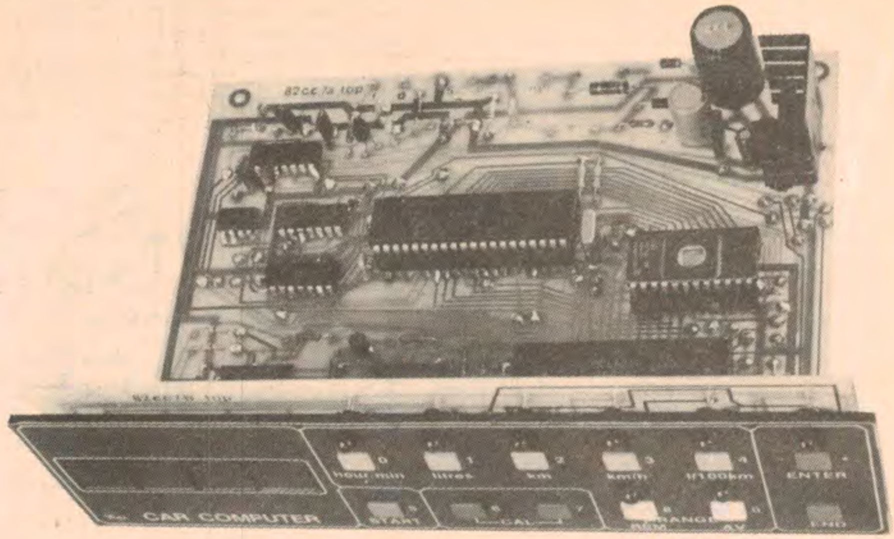
Although we do not intend to completely describe the software, flowcharts have been included to explain the basic concepts of the program. A few clarifying points, however, will help in tracing through these flowcharts.

As already mentioned, there are three programs for the Car Computer: RESET, Non Maskable Interrupt (NMI) and



CAR COMPUTER

Interrupt Request (IRQ). The "terminal point", shown in the key of symbols on Flowchart 1, starts off each program. Note that the RESET program has only a beginning and continues in a loop from then on. It is only the IRQ and NMI programs which return back to the RESET program after completion.



Repeated from last month, this photograph shows the completed PCB assembly. IC sockets are mandatory for the three main ICs, optional for others.

RESET program

Every one second, calculations are made for l/100km, km/h, km/h AV, hour.min REM, l/100km AV and km REM RANGE. Note that these calculations are made in this order, since some of them are interactive. For example, hour.min REM depends upon km/h AV while km REM RANGE relies upon l/100km AV.

All calculations involve division only. Multiplications are in factors of 10, in which case a simple left shift is all that is required. The necessary equations can be seen on Flowchart 1. Note that the l/100km calculation involves a × 10 rather than × 100 multiplication factor because the litre CAL is actually the number of pulses per 0.1 litres rather than per litre. Although the l/100km and km/h functions show equations involving multiplication, these are actually manipulated so that only divisions are carried out.

The two equations involving hour.min are actually more complex than shown since the minutes are converted to decimal hours in the case of the km/h AV calculation and from decimal hours to minutes in the hour.min REM calculation.

For those who are wondering how the data is handled, whether in binary or binary coded decimal (BCD), the answer is that all counting, with the exception of the time interrupt pulse counts, is in packed BCD. Consequently all data is stored in the BCD form and the actual division routine is performed in BCD.

Because all calculations are updated every second, regardless of the function displayed, changing from one function to another immediately provides an up-to-date calculation. At each of these one second updates, "Mask 2" is used to allow the program to skip over the 20ms key switch debounce time.

The remainder of the program is concerned with entering data, reading the key switches and initialising the PIA. Note that when entering the data, the far left-hand (most significant) digit is loaded with the first number pressed, the next digit to the right is loaded with the second number pressed, and so on. However, if four digits are not entered, upon pressing the END switch the whole display is shifted right until there are no blank displays to the right. The blanks now appear at the most significant side of the display.

Non Maskable Interrupt

We have already briefly described this routine, but the use of the Mask 1 has not been explained. Basically, this Mask prevents reading of the key switches. Previously, in the circuit description, it was mentioned that reading the key switches will clear the time $\overline{IRQB}$ and it is this we want to avoid if, when running the distance and fuel section of the IRQ program, $\overline{NMI}$ occurs. At the beginning of the fuel and distance IRQ routine, Mask 1 is set and is not cleared until this program is complete.

Interrupt Request

As mentioned previously, this routine updates the time, distance and fuel pulses and operates on these accordingly. Several important points should be made here. Firstly, decision logic in the time interrupt routine, titled "litres CAL small", decides whether the Moray fuel flow sensor or the Prince sensor is installed. Since the Moray sensor produces around 1500 pulses per 0.1 litres and the Prince sensor around 130, the program can easily determine which sensor is used by reading the entered calibration number. The logic then directs the program to count the fuel and distance pulses for the l/100km function over a one second period for the Moray sensor or over an eight second period for the Prince sensor.

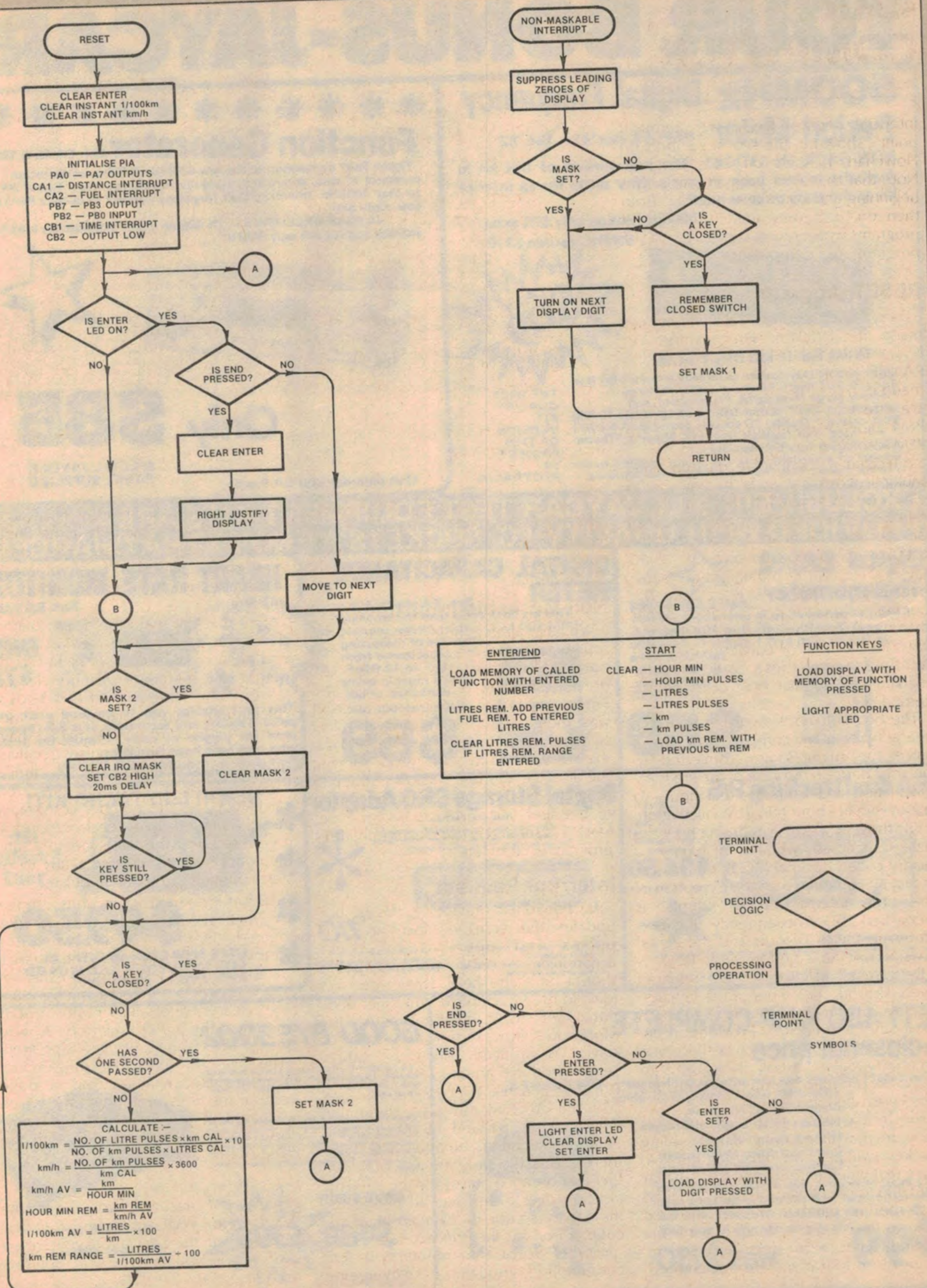
In other words, the instantaneous fuel consumption is updated once every one second if the Moray sensor is used, or once every eight seconds if the Prince sensor is used. (Clearly, the Moray sensor is the one to go for if you regard instantaneous fuel consumption as important.)

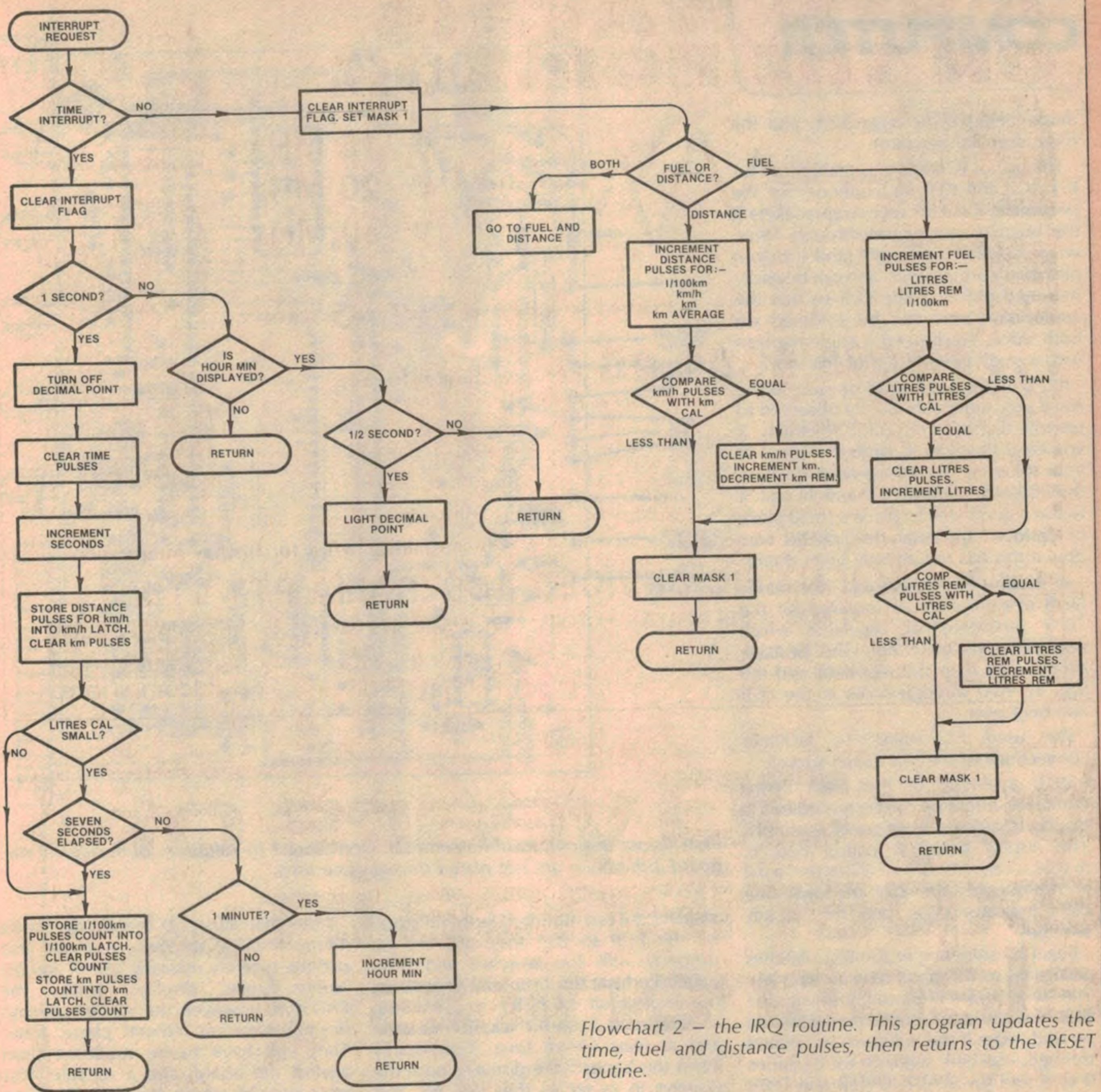
Note that the fuel flow pulse count is transferred to the latch memory at the end of each one second or eight second period, and the count memory cleared. When the subsequent l/100km calculation takes place, the latch memory is used to ensure that we have the correct number of pulses received during the count period. A similar method is also used for the km/h calculation; ie, the number of distance pulses counted over each second is transferred into the latch memory, and the count memory cleared ready to re-start counting the distance pulses.

The km and km REM functions also use a count memory for the distance covered. 1km is recorded when the number of km pulses equals the distance calibration (km CAL) number. The km pulses stored in the count memory are then cleared and allowed to re-start counting. The count memory is also cleared when the START key is pressed.

The litres and litres REM pulse count memories are separate. This is done because the litres pulse memory, which is cleared at the START of a journey, does not necessarily coincide with the filling of the tank.

Flowchart 1 (facing page) shows the RESET and NMI program routines, together with the calculations performed by the Car Computer. The NMI routine is used to update the display.


RESET
CLEAR ENTER
CLEAR INSTANT 1/100km
CLEAR INSTANT km/h
INITIALISE PIA
PA0 — PA7 OUTPUTS
CA1 — DISTANCE INTERRUPT
CA2 — FUEL INTERRUPT
PB7 — PB3 OUTPUT
PB2 — PB0 INPUT
CB1 — TIME INTERRUPT
CB2 — OUTPUT LOW
A
IS ENTER LED ON?
YES
NO
IS END PRESSED?
NO
YES
CLEAR ENTER
RIGHT JUSTIFY DISPLAY
MOVE TO NEXT DIGIT
B
IS MASK 2 SET?
YES
NO
CLEAR IRQ MASK
SET CB2 HIGH
20ms DELAY
CLEAR MASK 2
IS KEY STILL PRESSED?
YES
NO
IS A KEY CLOSED?
YES
NO
HAS ONE SECOND PASSED?
YES
NO
SET MASK 2
A
CALCULATE :—
l/100km = NO. OF LITRE PULSES × km CAL / NO. OF km PULSES × LITRES CAL × 10
km/h = NO. OF km PULSES / km CAL × 3600
km/h AV = km / HOUR MIN
HOUR MIN REM = km REM / km/h AV
l/100km AV = LITRES / km × 100
km REM RANGE = LITRES / l/100km AV ÷ 100
IS END PRESSED?
NO
YES
A
IS ENTER PRESSED?
NO
YES
LIGHT ENTER LED
CLEAR DISPLAY
SET ENTER
A
IS ENTER SET?
NO
YES
LOAD DISPLAY WITH DIGIT PRESSED
A
A
NON-MASKABLE INTERRUPT
SUPPRESS LEADING ZEROES OF DISPLAY
IS MASK SET?
NO
YES
IS A KEY CLOSED?
NO
YES
TURN ON NEXT DISPLAY DIGIT
REMEMBER CLOSED SWITCH
SET MASK 1
RETURN
B
ENTER/END
LOAD MEMORY OF CALLED FUNCTION WITH ENTERED NUMBER
LITRES REM. ADD PREVIOUS FUEL REM. TO ENTERED LITRES
CLEAR LITRES REM. PULSES IF LITRES REM. RANGE ENTERED
START
CLEAR — HOUR MIN
— HOUR MIN PULSES
— LITRES
— LITRES PULSES
— km
— km PULSES
— LOAD km REM. WITH PREVIOUS km REM
FUNCTION KEYS
LOAD DISPLAY WITH MEMORY OF FUNCTION PRESSED
LIGHT APPROPRIATE LED
B
TERMINAL POINT
DECISION LOGIC
PROCESSING OPERATION
TERMINAL POINT
SYMBOLS



Flowchart 2 – the IRQ routine. This program updates the time, fuel and distance pulses, then returns to the RESET routine.

Construction

Fortunately, construction of the EA Car Computer is a lot easier than understanding how it works. All the circuitry is accommodated on two double-sided printed circuit boards (PCBs) which are soldered together at right angles to virtually eliminate internal wiring. The completed PCB assembly is mounted in a standard Pac-tec case and fitted with a silver-on-black front panel that should mate well with the interior of most modern cars.

We understand that PCBs with plated-through holes will be available for this project, and these are well worthwhile as they simplify construction considerably. If the holes are not plated-through, you will have to solder the pads on the component sides of the PCBs as well as on the reverse sides. In this case, components such as IC. sockets (wire-wrap type) and capacitors will have to sit slightly proud of the PCB so that you can gain access to the leads.

In addition, if the holes are not plated-through, you will have to insert and solder a large number of pin-throughs. These pin-throughs consist simply of a short length of tinned copper wire soldered in and then cropped close to the board. They must be inserted first, since some are beneath ICs.

Before starting construction, very carefully inspect the two PCBs for possible shorts between tracks or breaks in the copper pattern. A few minutes careful checking here could save a lot of frustration later on. Check also that the edge bus on the main PCB runs right up to the edge; if not, file the edge until it does.

The way in which it all goes together is fairly obvious from the photographs and diagrams. Start by assembling the main PCB (code 82cc7a, 171 x 123mm) according to the parts overlay diagram, making sure that all polarised components are correctly oriented. These include the ICs, transistors,

CAR COMPUTER

diodes, electrolytic capacitors, and the three terminal regulator.

The use of IC sockets is mandatory for IC1, IC2 and IC3, and optional for the remaining ICs. Use wire-wrap sockets if the board is not plated through. Wire-wrap sockets have longer (and stronger) pins than normal types, and can be easily mounted proud of the PCB so that the appropriate pins can be soldered on both sides. You'll need a soldering iron with a small pointed tip for this work.

IC5 (74C14) is a CMOS device, so the usual precautions should be observed to prevent damage from static electricity. If you elect to solder it, earth the barrel of your soldering iron to the earth track on the PCB and solder pins 7 and 14 first. It is also a good idea to place a small piece of opaque tape over the EPROM window if this has not already been done.

A small clip-on heatsink, Thermaloy 6038 or equivalent, is required for the 7805 three-terminal regulator which normally runs quite hot. The heatsink simply clips over the regulator and the lugs inserted through holes in the PCB and bent over.

We used PC stakes to facilitate connections to the rear panel socket.

With assembly of the main board complete, attention can be turned to the display board and front panel assembly. The display board is coded 82cc7b, measures 191 x 57mm and accommodates the LED readouts, the eight indicator LEDs, and the 12 key switches.

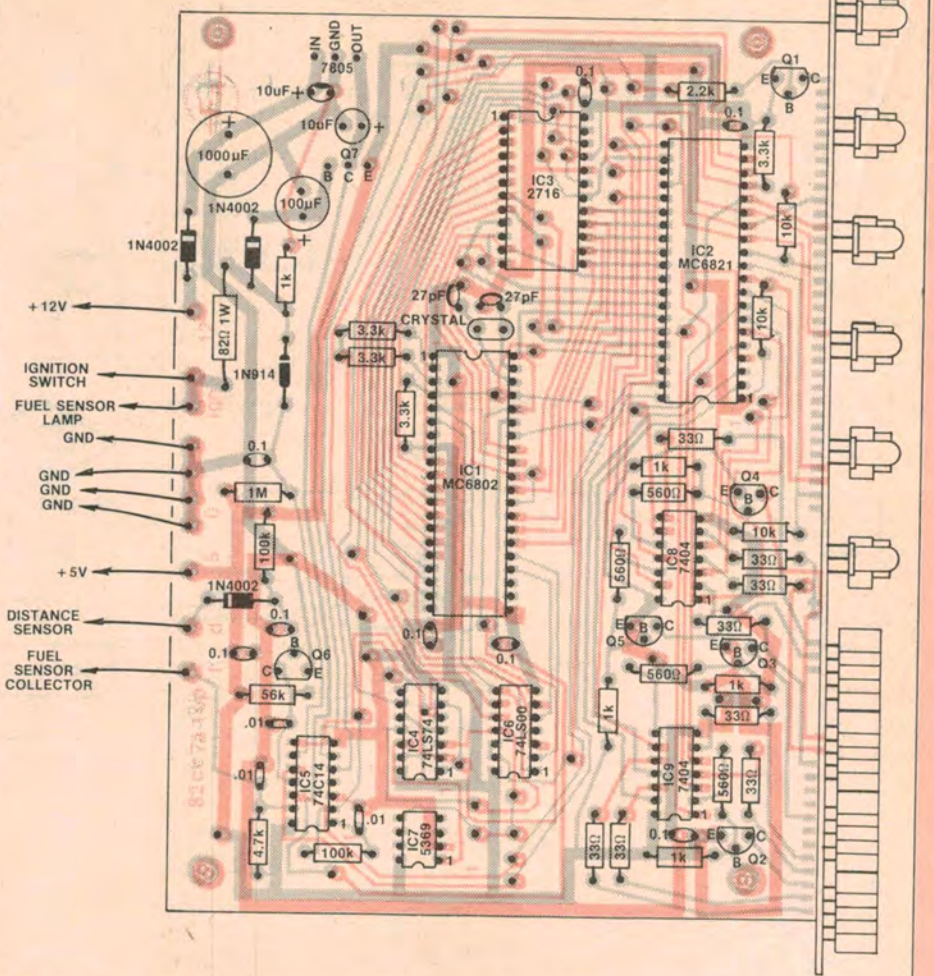
Begin by soldering in the key switches according to the parts overlay diagram. The switches are mounted flush with the PCB and the appropriate pins soldered on both sides if the holes are not plated through. Use blue switches for positions 0 to 4, red for START, ENTER and END, green for positions 6 and 7, and white for positions 8 and 9.

The four FND507 LED displays are next and must be oriented so that the ribbed edge of each display is at the top. The displays are not mounted flush but stood off the PCB so that they will line up properly with the switches and front panel. Perhaps the best way of locating the displays off the PCB is to use a strip of cardboard 1.5mm thick, 10mm wide and at least 65mm long. Temporarily insert this beneath the displays, push the displays in as far as they will go, and solder.

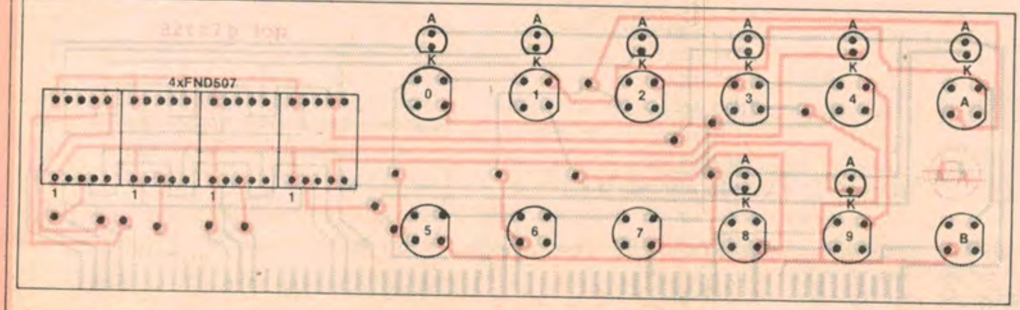
As with the key switches, some of the pins will have to be soldered on both sides of the PCB if the holes are not plated through.

Note that although the circuit shows 13mm FND507 displays, you can also use the recently released 15mm Stanley "super bright" displays. Unlike the FND507s, however, the Stanley displays do not have an integral plastic filter. They will have to be mounted flush against the board, and a suitable filter inset into the cutout. You will also have to make the cutout slightly larger.

Stanley displays are distributed by A&R Soanar and are available in three colours: NKR163 red, NKG163 green, and NKY163 yellow. Note that, in this

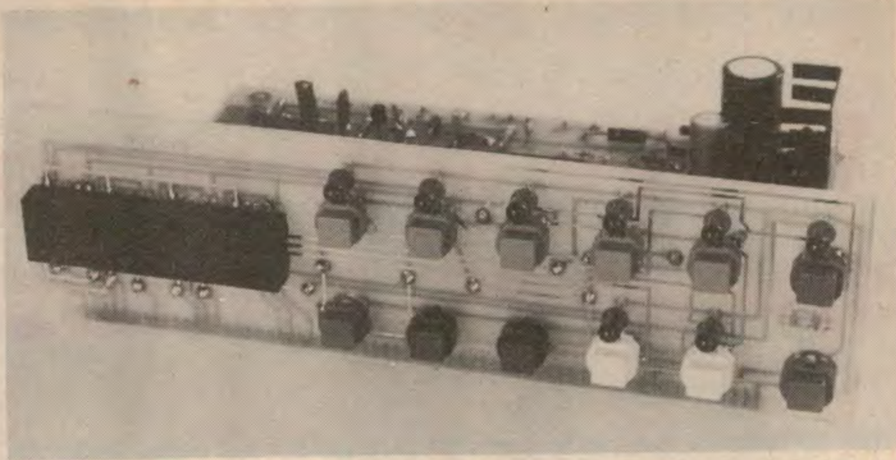


Parts layout diagram for the main PCB. Don't forget to solder on both sides of the board if the holes are not plated through (see text).



Left: parts overlay diagram for the display PCB. Make sure that you mount the four FND507 displays the right way up.

CAR COMPUTER



View showing the display PCB assembly before fitting the front panel. Note that the FND507 displays are mounted proud of the board (see text).

application, they can only be used with a plated-through PCB.

The Scotchcal label should now be carefully affixed to the smooth side of the front panel, and holes drilled and filed to shape to take the displays, LEDs and switches. Proceed carefully with this step, periodically offering the front panel to the display board so that you can judge how much progress has been made. The job is admittedly tedious, but requires care to ensure a neat finish.

In some kits, however, this work will not be necessary. At least one retailer will be supplying silk-screened and pre-punched front panels to ensure a "snazzy" job!

At this point, mount the LED bezels on the front panel and snap the eight LEDs into position. Orient the LEDs so that the anode leads are at the top, then insert all the LED leads through their respective mounting holes by carefully offering the front panel to the display PCB. Position the front panel so that it sits flush with the front surface of the seven-segment displays, then solder each LED in turn.

Check that all switches operate correctly and make any necessary adjustments before trimming the LED leads. The switches should sit about 1.5mm proud of the front panel.

As before, you will have to solder both sides of the PCB if the holes are not plated through. Install the LEDs as described above, then remove the front panel (by pushing the LEDs out of the bezels) to gain access to the two pads on the component side. Re-install the front panel when you have finished soldering.

The display PCB can now be soldered to the main PCB. To do this, slide the front panel/display PCB assembly into the retaining slot at the front of the case, and screw the main PCB to the four moulded standoffs on the base. Check that the two edge buses line up, then solder the six mating bus pads together. This done, remove the PCB assembly from the case and solder the bus pads on the reverse side.

With assembly of the PCBs complete, go over your work and check that all components are in the correct position and that all the pads have been soldered. Before actually inserting the three main ICs into their sockets, it is best to check voltages around the circuit.

Apply between 10 and 15 volts DC to the +12V and ignition terminals, and check the supply voltages on all ICs and IC sockets with a multimeter. If all is correct, disconnect power and insert the ICs. When power is reconnected the word "rEdY" should appear if the START switch is pressed. Pressing other buttons should turn on the appropriate indicator LED and bring up various numbers on the display.

Next month, we will tell you how to fit the sensors to the vehicle and describe how the Car Computer is operated.

PARTS LIST

1 Pac-tec case, 205 x 159 x 65mm
1 double-sided PCB, code 82cc7a, 171 x 123mm
1 double-sided PCB, code 82cc7b, 191 x 57mm
1 12-way Utilux line plug socket and panel plug
1 TO-220 clip-on heatsink, Thermaloy 6038 or equivalent
1 Scotchcal front panel, 192 x 59mm
12 Isostat key switches, 5 blue, 3 red, 2 green, 2 white
1 3.58MHz crystal
2 40-pin DIL sockets (see text)
1 24-pin DIL socket (see text)

SEMICONDUCTORS
1 MC6802 microprocessor
1 MC6821 PIA
1 74LS74 dual D flipflop
1 74LS00 quad NAND gate
2 7404 hex inverters
1 74C14 hex Schmitt trigger
1 MM5369 divider, 60Hz version
1 2716 2K EPROM with EA Car Computer program
1 7805, LM340T 5V regulator
5 BC327 PNP transistors
1 BC549 NPN transistor
1 BD139 NPN transistor
3 1N4002 1A silicon diodes
1 1N4148, 1N914 small signal diode
4 FND507 common anode displays, or equivalent
8 5mm red LEDs plus matching bezels

CAPACITORS
1 1000μF/16VW PC mounting electrolytic
1 100μF/16VW PC mounting electrolytic
1 10μF/16VW PC mounting electrolytic
1 10μF/16VW tantalum or low leakage electrolytic
8 0.1μF monolithic
3 .01μF metallised polyester
2 27pF miniature ceramic

RESISTORS (¼W, 5% unless stated)
1 × 1MΩ, 2 × 100kΩ, 1 × 56kΩ, 3 × 10kΩ, 1 × 4.7kΩ, 4 × 3.3kΩ, 1 × 2.2kΩ, 5 × 1kΩ, 4 × 560Ω, 8 × 33Ω, 1 × 82Ω 1W.

SENSORS (see text next month)
1 fuel flow sensor, Prince or Moray
1 distance sensor, Compucruise or Pimac
1 length of brass rod, 5mm diameter x 20mm long
1 T-junction piece to suit
1 length of fuel line hose plus clamps to suit

MISCELLANEOUS
Hook-up wire, solder, PC stakes, screws, nuts, etc

NOTE: Components specified are those used in the prototype. In general components with higher ratings can be used providing they are physically compatible.

Sensors, operation & EPROM program

Electronics Australia
SEPTEMBER 1982

CAR COMPUTER PART THREE

Our final article this month tells you how to fit the fuel flow and distance sensors to the vehicle, and describes how the Car Computer is operated. Also included is a listing of the EPROM program.

by JOHN CLARKE and GREG SWAIN

Before actually describing how the sensors are fitted to the vehicle, we should first point out that the Prince fuel flow sensor will not now be available with this kit. We learned of this situation shortly after the August issue went to press, and by that stage it was too late to make any alterations.

This means that kit suppliers will only be supplying the Moray fuel flow sensor, which is all to the good. As explained last month, the Moray sensor is the preferred unit as it allows a one second update time for instantaneous fuel consumption. The Prince unit, by comparison, was only good for eight second update times.

The way in which the sensors are fitted will be largely self-evident from the accompanying diagrams. Fitting is straightforward, although dirty hands and bruised knuckles are par for the course when working with any motor vehicle. We suggest that constructors read this article carefully, as there are a number of important guidelines that must be followed.

Fuel flow sensor

The fuel flow sensor is fitted in the fuel line between the fuel pump and the carburettor – preferably after the fuel filter (see Fig. 1). It should ideally be mounted vertically, but if it has to be mounted horizontally it should be fitted with the fuel passage above the detector housing. The arrow on the side of the sensor must point in the direction of flow; ie, towards the carburettor.

In cars which use flexible hosing between the fuel pump and carburettor, it is simply necessary to disconnect the hose at the pump end (assuming a mechanical fuel pump with integral filter) and insert the flow sensor in the line. If a metal fuel line is used, it will have to have a suitable length cut out of it. Remove the fuel line from the vehicle before cutting so that the ends can be satisfactorily deburred and any filing cleaned away.

Use only genuine fuel line hose to make the interconnections to the sensor. Do not use nylon tubing or any other type of hose not designed for petrol, as it will become hard and prone to leakage within a short period of time. Secure each connection with a hose clamp to prevent leakage.

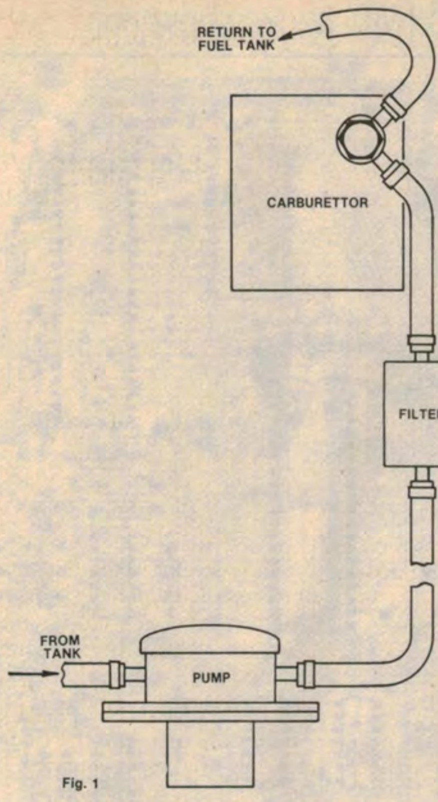
Because of its light weight, the Moray fuel flow sensor can usually be mounted suspended in the fuel line without the need for a supporting bracket. Avoid close proximity to ignition leads, and keep it well away from the exhaust manifold to avoid the possibility of vapour lock. You should also ensure that the sensor is correctly oriented, that all clamp connections are tight, and that the hoses are not kinked or stressed.

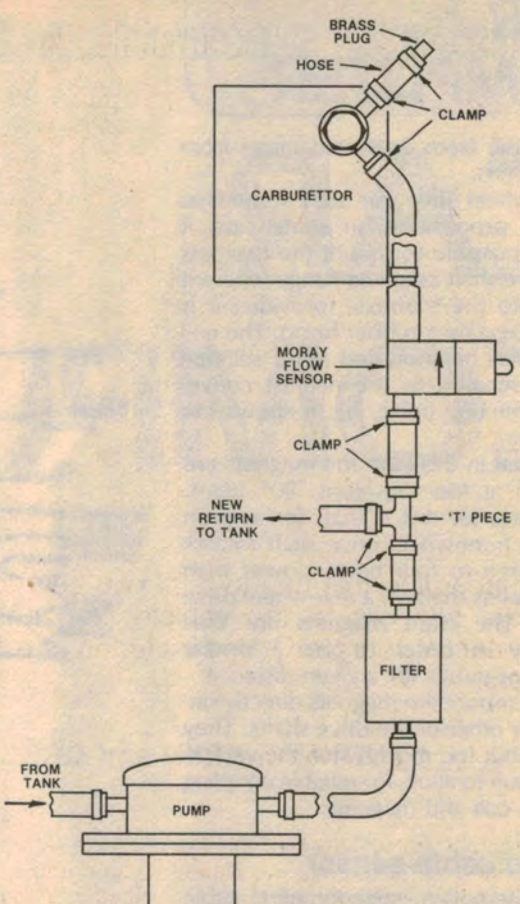
On vehicles fitted with a recirculating fuel system, the flow sensor should be fitted as shown in Fig. 2. A car with a recirculating fuel system is one in which petrol is not only pumped to the carburettor, but is pumped back to the fuel tank as well. If your car has a second pipe connected to the fuel line union at the carburettor, and which runs back to the tank, then it has a recirculating fuel system (see Fig. 1).

Fairly obviously, any fuel returned to the tank must bypass the fuel flow sensor otherwise we would get an incorrect reading of fuel usage. This problem is overcome by fitting a T-junction into the fuel line between the pump and the flow meter. The original return is then blocked off using a suitable brass plug secured by a clamp.

Make sure that the stem of the T-junction is used for the return to the

Car Computer can be mounted at any convenient location on or under the dashboard. Unit is easy to operate and calibrate. (see text).



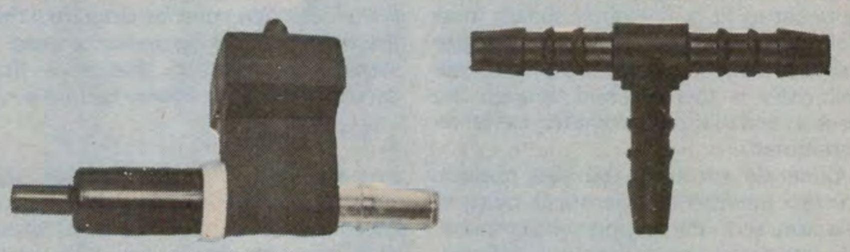


Figs. 1 & 2: fitting the fuel flow sensor to a car with a recirculating fuel system. Mount the sensor vertically, with the arrow pointing in the direction of the flow.

tank, as this contains a restricting orifice which limits the return flow. On some cars, a T-junction is fitted between the pump and carburettor as standard, in which case the flow meter should be fitted after the T-junction.

WARNING: some cars, such as the Holden Commodore, are fitted with a vapour recovery system employing a charcoal cannister. This is NOT the same as a recirculating fuel system, and on no account should you tamper with the vapour recovery line between the charcoal cannister and the carburettor. If in doubt, consult your local dealer.

Unfortunately, the Moray fuel flow sensor cannot be used with fuel injection or diesel engines.



Moray fuel flow sensor (left) and matching T-piece (right). The T-piece is used only in cars which have recirculating fuel systems.

Distance sensors

As stated last month, two different distance sensors can be fitted: (1) a magnetic pick-up using a coil and rotating magnets, or (2) a speedometer cable sensor. Generally, the magnetic pick-up system will suit most rear-wheel drive cars with a front-mounted engine, as the tail shaft is an ideal position for the magnets. It may also be possible to find a suitable position for the magnets on some front-wheel drive cars.

Figs. 3 and 4 show how the magnetic pick-up sensor is installed in rear-wheel and front-wheel drive cars respectively. In the case of a rear-wheel drive car the sensor should be mounted as close to the gearbox as possible, where vertical movements of the tail shaft are minimal. The magnets are secured to the tail shaft using tie wire and epoxy adhesive.

We used four 15mm-dia round magnets in all, two mounted side-by-side at each position to compensate for any longitudinal movement of the tail shaft (see Fig. 3). Some kit suppliers, however, will be supplying 25mm-long bar magnets, in which case only one magnet will be required at each position.

The coil was mounted on an L-shaped bracket made from aluminium and secured to the underside of the car using self-tapping screws. This bracket should be positioned so that there is a 10mm gap between the end of the coil and the magnets when they are directly opposite each other. Be careful not to damage any wiring cables running along the floor when drilling the mounting holes for the bracket.

Wiring to the coil can be run along the underside of the car, with the leads secured at various points as convenient. Do not connect the earth lead to chassis at the coil mounting position. Instead, we suggest that both the earth and signal leads be run as a twisted pair all the way back to the rear panel socket on the computer. Plastic tubing can be used to

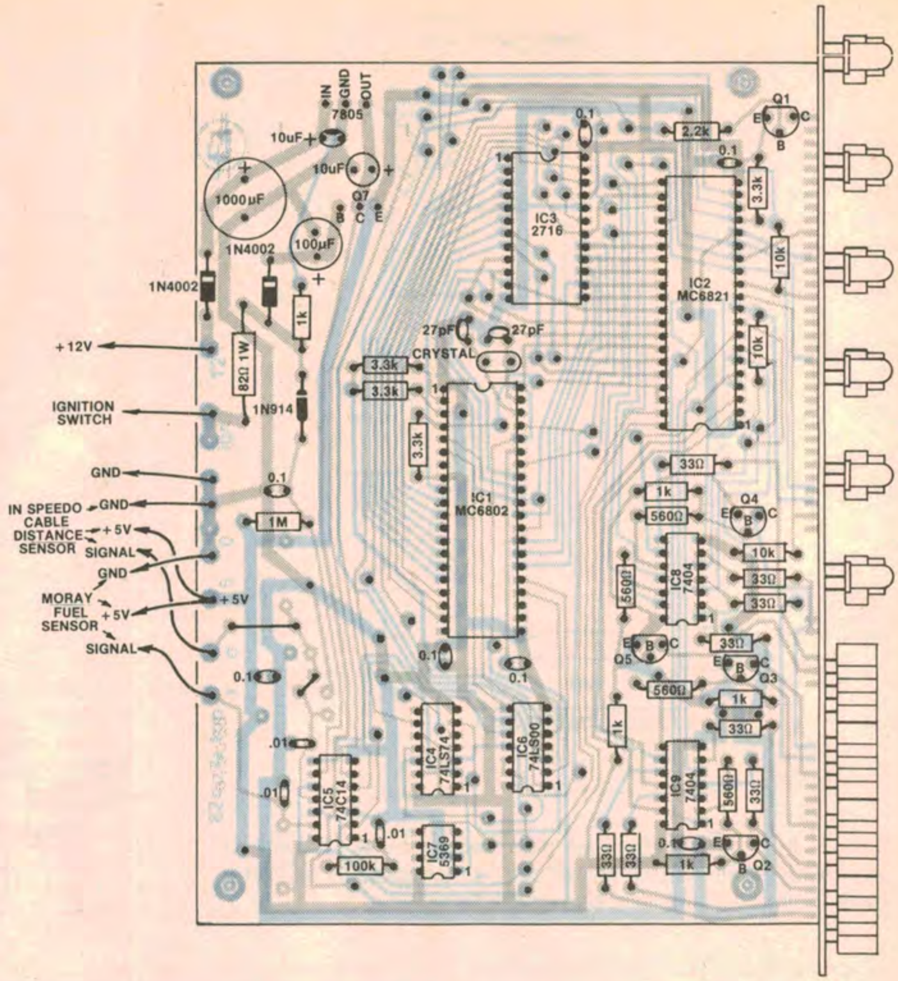
CAR COMPUTER

protect the leads against damage from flying stones.

Front-wheel drive cars are a somewhat different proposition. In some cars, it may be possible to mount the magnets on a drive-shaft coupling flange where it bolts onto the transaxle (provided it is not covered by a rubber boot). The coil could then be mounted on a suitable bracket secured to the nearest convenient mounting point. Fig. 4 shows the basic idea.

Note that in this case the magnets are mounted at four positions, 90° apart. The reason for this is that, for a given speed, a front-wheel drive shaft rotates about three to four times slower than the propeller shaft on a rear-wheel drive vehicle. The extra magnets are thus necessary in order to get a similar number of pulses for a given distance.

Do not mount the magnets directly onto one or other of the drive shafts. They move about too much when the vehicle is in motion to allow for reliable coupling between coil and magnets.



Follow this parts overlay diagram if the speedometer cable distance sensor is used. If the magnetic pick-up sensor is used, the parts overlay diagram in the August issue should be followed, but note that the 4.7kΩ resistor must be deleted to accommodate the Moray fuel flow sensor.

Speedo cable sensor

The alternative speedometer cable sensor can be used with both front and rear-wheel drive cars, but is mainly applicable to front-wheel drive cars where the drive shaft flanges are not accessible. In order to fit it, the outer sheath must be removed from the speedometer cable and cut at a suitable point. The inner cable is then pushed through the sensor and the speedometer cable reassembled.

Generally speaking, the best position for the speedometer sensor is close to the firewall in the engine compartment. The installation procedure is as follows:

- Mark the appropriate position with white chalk, then remove the speedometer cable from the vehicle;
- Remove the retaining circlip and withdraw the inner cable;
- Using a hacksaw, cut out and discard a 15mm section of the outer sheath at the marked position;
- Push the inner cable through the sensor and refit the two sheath sections by clamping the ends in the slotted end tubes. Note that the inner cable should be a force fit into the sensor, otherwise the slotted disc inside the sensor will not rotate;
- Check that the inner cable is free to rotate, then re-install the speedometer cable in the vehicle.

As supplied, the speedometer cable sensor can be fitted to most speedometer cables. Readers should note, however, that it cannot be used in vehicles fitted with an electronic speedometer – eg XD and XE Ford Falcons – for the simple reason that such vehicles do not have a speedometer cable!

If you do elect to fit the speedometer cable sensor, then the PCB parts layout diagram accompanying this article should be followed. If the magnetic pick-up sensor is used, then the overlay diagram in the August issue should be followed but don't forget to delete the 4.7kΩ resistor in order to accommodate the Moray fuel flow sensor.

Once the sensors have been installed, the wiring to the Car Computer can be completed. Fig. 5 shows the recommended wiring to the 12-pin Utilux socket, as viewed from the front. The leads from the sensors are passed through the firewall, and terminated to the appropriate mating pins on the matching plug. Make sure that you get these connections right, otherwise the circuitry could be damaged.

The +12V from the ignition switch and the permanent +12V supply can be obtained from the fusebox. Check the voltages available with a multimeter before actually connecting the leads, and make the connections to the fused side. The ground connection can be made at any suitable chassis point.

The Car Computer itself can be mounted at any convenient location on or under the dashboard, using a suitable U-shaped bracket. We'll leave it to readers to work out mounting details to suit individual model cars.

Operation

At first switch on, all memory locations have random numbers located in them. Press the START key to clear hour.min,

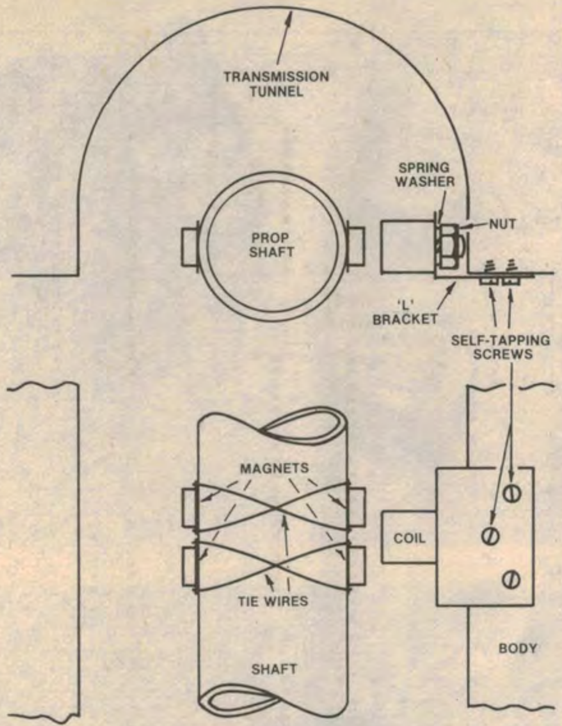


Fig. 3

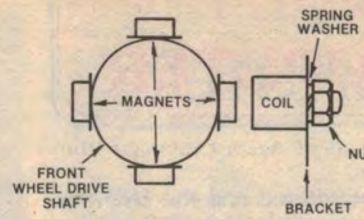


Fig. 4

Fig. 3 (left) shows how the magnetic pick-up sensor is installed in a rear-wheel drive car while Fig. 4 (above) shows the installation for a front-wheel drive car (see text).

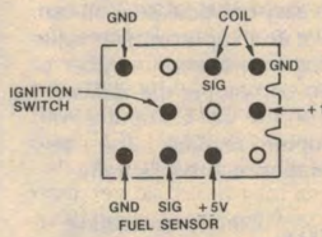


Fig. 5

Fig. 5: recommended wiring for the 12-pin Utilux socket, as viewed from the front. An additional connection to +5V will be required if the speedometer cable sensor is used.

litres, km, and their associated pulse count memories. Data to be entered initially includes the maximum fuel tank capacity, and the litres CAL and km CAL calibration numbers. The fuel remaining (litres REM) and journey length (km REM) figures must also be entered initially and, subsequently, each time fuel is added to the tank or a "new" journey undertaken.

Let's go through the procedure step-by-step:

- **Litres CAL.** The number to be entered here is the number of pulses that the fuel sensor provides for 0.1 litres of fuel flow. Each Moray sensor is precalibrated by the manufacturer, and a small tag on the sensor lead provides a clue to the calibration number to be used. On one side of the tag is a number for litres, and on the other a number for gallons. All you have to do is multiply the litre reading by 25.6 to get the litre CAL number for that particular sensor. For example, our Moray sensor has the number 67; multiplying this by 25.6 gives a calibration number of 1715 pulses per 0.1 litres.

To enter this number, first press the litres CAL key switch (key switch 6). The display should show a random number. Now press the ENTER key switch – the display will clear and the ENTER LED will light. Finally, enter the calculated calibration number and press the END key to extinguish the ENTER LED. For example, to enter 1715 press 6, ENTER, 1, 7, 1, 5, END.

The display will now show the calibration number just entered. If you are not convinced, press another function key and then return to the litres CAL key. The previously entered number will again be displayed. Note that although a decimal point can be entered into the calibration number, this will be totally ignored by the computer. After the END key is pressed, the decimal point will disappear.

It is a good idea to write the calibration number down and keep it in a safe place as it will have to be re-entered each time the battery is disconnected from the vehicle. The same applies to the distance calibration number (see below) and any other enterable data.

- **km CAL.** This is the number of pulses received from the distance sensor in one kilometre. If you are using a magnetic pick-up sensor mounted on the tail shaft, an approximate distance calibration number can be calculated. Measure the wheel diameter (the horizontal distance) in mm, multiply by 3.14 (π) and divide the result into 1 million. This gives the number of wheel rotations per kilometre. Multiplying this value by the differential ratio of the particular car and then by two (two magnets on the shaft) will then provide an approximate number for the distance calibration.

The actual number will be in the region of 4000.

To enter the distance calibration number, first press the distance CAL key switch (key switch 7) followed by the ENTER button. The calibration number

The speedometer cable distance sensor can be used on both front and rear-wheel drive cars, but not on cars which have an electronic speedometer (eg, XD, XE Ford Falcon).

can now be entered and the END key pressed to extinguish the ENTER LED (eg to enter 4000 press 7, ENTER, 4, 0, 0, 0, END).

The alternative speedometer cable sensor provides a similar number of pulses per kilometre, although the actual value is not as easy to calculate. You can, however, arrive at an approximate value by first entering a calibration number of 4000 and then comparing the indicated km/h reading on the Car Computer with the speedometer reading. The new distance calibration number is then:

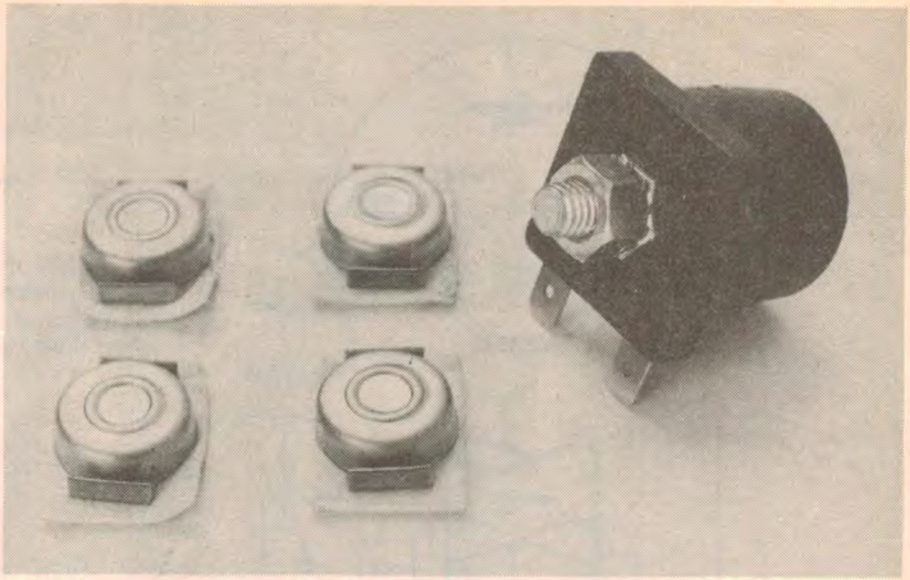
$$\text{km CAL} = 4000 \times \frac{\text{Computer reading}}{\text{Speedometer reading}}$$

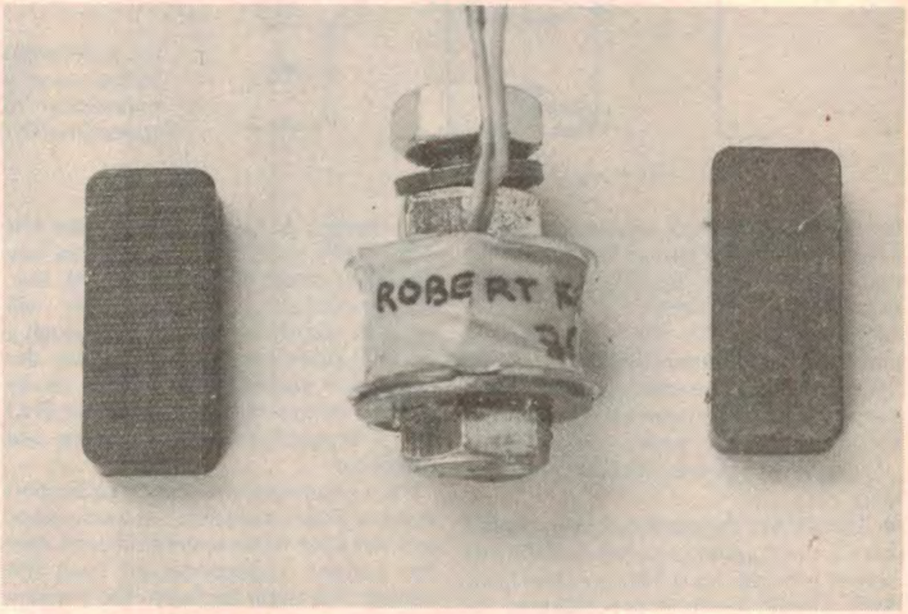
Note that the maximum calibration number that can be fed in is 9999. If the calculated calibration number is greater than this, you will have to dismantle the sensor and remove half the number of vanes (either that, or use the magnetic pick-up).

The exact distance calibration number (both types of sensors) can now be found using one of two methods. First, a direct comparison can be made between the km reading on the Car Computer and the car's odometer. Zero the odometer trip meter, press the START button on the computer, and drive the car for 15-20km (longer journeys will give a more accurate result). The true calibration number is now calculated by multiplying the old calibration number by the computer reading and dividing by the odometer reading.

A more accurate method involves checking the indicated distance on the Car Computer against the kilometre marking posts located alongside major highways. If you do use this method, don't be surprised if the km and km/h readings on the Car Computer differ from the values indicated on the speedometer. Car speedometers are not particularly accurate devices!

● **km REM.** Press the km and REM keys and the associated LEDs will light, displaying the km REM value. Initially, a random number will be displayed. Once entered, the km REM value will be reloaded into the km REM memory each time the START key is pressed unless another distance is entered. This value will decrease by 1km for every kilometre travelled until zero is reached, after which the display will count down from 9999km.





Two versions of the magnetic pick-up sensor currently available from kit suppliers. See Figs. 3 & 4 and text for installation details.

To enter data, press km, REM and ENTER, enter the expected distance, and press the END key. For example, to enter a journey length of 483km press km, REM, ENTER, 4, 8, 3, END. Note that entering a decimal point will cause a "d.Err" to be displayed, indicating that the value will have to be entered again.

If the estimate for the distance of the journey is subsequently found to be incorrect, the km REM value can be changed at any time. The hour.min REM is automatically corrected for the new value.

● **litres REM RANGE.** This value is the maximum capacity of the fuel tank to the nearest 0.1 litres, as specified in the vehicle handbook. The value stored is actually provided for convenience when the tank is filled, since it can be directly entered into the litres REM memory. Press litres, AV and ENTER, enter the tank capacity, and then press END. Note that the entry must be made to the nearest 0.1 litres, otherwise "F.Err" will appear on the display.

● **litres REM.** This is the actual amount of fuel in the tank, so new data must be entered each time fuel is added. Initially, a random number will appear and you will have to determine and enter the amount in the tank. The easiest method is to fill the tank and then enter the tank capacity. This is done by pressing litres and AV (to display the tank capacity), followed by ENTER and END. Do not enter any numbers. The litres REM RANGE value will now be stored in the

CAR COMPUTER

litres REM memory, and this can be verified by pressing the litres and REM keys.

If the fuel tank is not completely filled, the extra fuel must be entered and added to the litres REM memory. To do this, press litres, REM and ENTER, enter the amount of fuel added to the tank, and press END. The entered value will be automatically added to the previously displayed litres REM reading. Once again, fuel must be entered to the nearest 0.1 litres otherwise "F.Err" will be displayed and the entered fuel will be ignored.

Error correction

Mistakes made in entering data, other than litres REM, are easily corrected by pressing END and ENTER, then re-entering the correct value. An incorrect litres REM entry is a little trickier, however, since the entered value is actually added to the previous value. Basically, you've got three choices:

If the decimal point has not been entered when the mistake is realised, press the END key to display "F.Err". This will prevent the entered value from being added to the previous value. All you have to do now is re-enter the correct value.

If the decimal point has been entered but the END key not pressed, the situation can be retrieved by typing in several leading zeroes followed by the correct number. Entering is cyclic, so any digits re-entered will overwrite the previously entered digits.

If, however, the decimal point and END keys have both been pressed, display litres REM RANGE, enter the previous litres REM value (assuming that you can remember what it was), and transfer this value to litres REM (see above). The fuel added is now re-entered in litres REM, and the correct fuel tank capacity re-entered in litres REM RANGE. If you are unable to remember what the previous litres REM value was, then you will have to fill the tank completely and transfer the litres REM RANGE value to litres REM.

Troubleshooting

If your Car Computer does not work, don't rush out and buy new ICs. Provided that they've been installed correctly and that the power supply is correct, the ICs are not likely to be at fault. Instead, go over the project carefully and check that you've inserted all pin-throughs (for

CAR COMPUTER EPROM PROGRAM

```
0000 86 C9 B7 80 05 86 FF B7 80 04 B6 80 05 8A 04 B7
0010 80 05 F6 80 04 86 F1 B7 80 07 86 F8 B7 80 06 B6
0020 80 07 8A 04 B7 80 07 7F 00 20 96 1E 81 00 26 03
0030 7E 62 22 81 01 26 03 7E 63 A8 81 02 26 03 7E 62
0040 71 81 03 26 03 7E 63 62 81 04 26 03 7E 63 85 81
0050 07 26 03 7E 62 FB 81 06 26 D6 20 F7 7F 00 54 96
0060 54 81 00 26 2C 96 20 81 01 27 F4 96 1E 81 07 27
0070 EE 81 06 27 EA 81 05 27 E6 96 2C 81 3B 27 E0 96
0080 06 81 01 26 04 97 54 20 48 BD 65 ED 7C 00 5B 20
0090 CE CE 61 2A 96 54 5F A1 00 27 0B 08 5C 8C 61 36
00A0 27 02 20 F3 20 B6 C1 0B 27 20 96 20 81 01 27 28
00B0 C1 0A 27 10 C1 08 27 17 C1 09 27 13 D7 1E 7F 00
00C0 1F 7E 60 2A C6 01 D7 20 20 F7 7F 00 20 20 F2 D7
00D0 1F DE 26 08 DF 26 20 E9 A6 0C 20 2E 96 54 81 01
00E0 27 15 CE 09 00 09 26 FD 0E 96 23 B1 00 5A 27 07
00F0 86 FD B7 80 07 20 F1 DE 26 9C 28 27 06 09 DF 26
0100 7E 60 5C 08 08 08 DF 26 20 F6 81 00 27 06 DE 26
0110 A7 00 20 C8 DE 26 08 D6 29 5C 5C 5C D1 27 27 BC
0120 A6 00 8A 80 A7 00 DF 26 20 B2 22 42 0C 14 24 11
0130 09 21 41 44 0A 12 7E 44 3D 6D 47 6B 7B 4C 7F 6F
0140 00 97 21 DF 37 D6 37 C4 F0 8D 23 D7 2D D6 37 C4
0150 0F 8D 20 D7 2C D6 38 C4 F0 8D 13 96 21 81 20 26
0160 02 CA 80 D7 2B D6 38 C4 0F 8D 08 D7 2A 39 0C 56
0170 56 56 56 D7 60 CE 61 36 4F 91 60 27 08 4C 08 81
0180 0A 27 05 20 F4 E6 00 39 C6 7F 39 FF FF FF 96 2A
0190 84 7F 7F 00 21 8D 4E DE 37 E7 00 96 2B 85 80 26
01A0 25 8D 3B DE 37 EA 00 E7 00 96 2C 85 80 26 1F 8D
01B0 34 DE 37 09 E7 00 96 2D 85 80 26 1A 8D 20 DE 37
01C0 09 EA 00 E7 00 39 C6 20 D7 21 84 7F 20 D3 C6 40
01D0 D7 21 84 7F 20 D9 C6 80 D7 21 84 7F 20 DE 8D 05
01E0 58 58 58 58 39 5F CE 61 36 81 00 27 08 A1 00 27
01F0 04 08 5C 20 F8 39 CE 00 2F DF 26 CE 00 2A DF 28
0200 7E 60 DC 86 00 09 0C 66 00 08 66 00 4C 81 04 26
0210 F4 39 96 2E 8A 20 97 2E CE 00 00 DF 2A DF 2C 7E
0220 60 DC D6 1E C1 00 27 28 7F 00 2E DE 00 DF 12 CE
0230 00 00 DF 04 DF 08 DF 0E DF 06 DF 0A DF 10 86 1A
0240 97 2D 86 3B 97 2C 86 75 97 2B 86 67 97 2A 20 15
0250 D6 1F C1 08 27 15 C1 09 27 0B DE 04 86 01 97 2E
0260 86 00 BD 61 41 7F 00 20 7E 61 F6 86 81 DE 02 20
0270 ED D6 1F C1 00 27 31 C1 09 27 3A 96 20 81 01 27
0280 24 96 2E 81 A4 27 32 20 28 7F 00 20 D6 1F C1 08
0290 27 0B C1 09 27 0B 86 04 97 2E 7E 61 F6 86 84 20
02A0 F7 86 C4 20 F3 7E 62 12 DE 0E 86 00 BD 61 41 20
02B0 D8 DE 12 20 F5 DE 33 20 F1 86 13 97 38 7F 00 37
02C0 BD 61 8E CE 00 13 86 00 91 2C 27 0A 91 2B 27 09
02D0 91 2A 27 08 20 09 BD 62 03 BD 62 03 BD 62 03 96
02E0 21 81 00 27 10 86 F5 97 2D 86 3B 97 2C 86 11 97
02F0 2B 97 2A 20 94 DE 12 DF 00 20 B6 D6 20 C1 01 27
0300 0E 96 2E 85 20 26 1E 20 09 7F 00 2E 7E 61 F6 7E
0310 62 12 96 1E 81 06 27 09 DE 1A 86 00 BD 61 41 20
0320 E8 DE 1C 20 F5 7F 00 37 96 1E 81 06 27 0E 86 1B
0330 97 38 BD 61 8E CE 00 00 DF 10 20 0C 86 1D 97 38
0340 BD 61 8E CE 00 1D 20 03 CE 00 1B 86 00 91 2B 27
0350 06 91 2A 27 05 20 06 BD 62 03 BD 62 03 7F 00 2E
0360 20 B0 D6 1F C1 00 27 06 C1 09 27 13 20 0E 86 08
0370 DE 31 97 2E 86 00 BD 61 41 7F 00 20 7E 61 F6 86
0380 48 DE 14 20 ED 7F 00 20 D6 1F C1 00 27 06 C1 09
0390 27 10 20 0B C6 10 DE 35 86 20 D7 2E BD 61 41 7E
03A0 61 F6 C6 50 DE 52 20 F0 D6 1F C1 00 27 3F C1 08
03B0 27 0E 96 20 81 01 27 32 96 2E 81 E2 27 40 20 3A
03C0 96 20 81 01 27 24 96 2E 81 A2 27 74 20 28 7F 00
03D0 20 D5 1F C1 00 27 0B C1 08 27 0B 86 C2 97 2E 7E
03E0 61 F6 86 02 20 F7 86 82 20 F3 7E 62 12 DE 08 86
03F0 20 BD 61 41 20 D8 DE 0C 20 F5 DE 16 20 F1 7F 00
```

continued on page 69

CAR COMPUTER

boards without plated-through holes) that you've soldered both sides of the PCB as required, that all components are correctly oriented, and that none of the IC pins have been bent under the body of the IC.

Check also that the sensor connections are correct, and that there are no broken or shorted tracks on the PCB. If any of the digits fail to light, check driver transistors Q2 to Q5 and check for continuity between each digit driver output and pin 3 of the corresponding display. Similarly, if any segments fail to light, check for breaks in the segment drive lines between the PIA and the inverters (ICs 8 and 9), between the inverters and the displays, and between the displays themselves.

Detailed troubleshooting will generally require the use of a CRO, although the simple checks outlined above will usually be quite sufficient. For what it's worth, we built up three versions of the Car Computer without encountering any problems.

In use, calculated values such as hours.min REM, km REM RANGE, km/h AV and litres/100km AV will not give sensible readings until several kilometres have been travelled after pressing the START key. The litres/100km reading may also appear to be unusual when the vehicle is braked. As the vehicle slows the fuel consumption initially falls and then dramatically increases just before the vehicle is brought to a stop. This is because, at very low speeds, the distance travelled between fuel pulses progressively decreases while the fuel flow rate remains fairly constant.

The limiting case is obviously at standstill, when fuel is used but no distance is travelled. The litre/100km value then becomes infinite, however the Computer will not display this since no distance pulses are received.

"Hold it! I ordered two MICRO farad!" (Radio Electronics).

continued from page 68

```
0400 18 7F 00 19 DE 2C 8C 00 00 27 2F 86 51 97 38 7F
0410 00 37 BD 61 8E 96 21 81 80 27 0A 81 40 27 0C 81
0420 00 27 4A 20 0C CE 00 51 BD 62 03 CE 00 51 BD 62
0430 03 7F 00 21 DE 50 DF 16 20 C0 DE 16 DF 0C 20 B6
0440 86 51 97 38 7F 00 37 BD 61 8E 96 21 81 80 27 0C
0450 81 40 27 0E 81 00 27 15 20 0E 20 98 CE 00 51 BD
0460 62 03 CE 00 51 BD 62 03 7F 00 21 20 10 86 9B 97
0470 2D 86 3B 97 2C 86 11 97 2B 97 2A 20 DD 96 51 9B
0480 0D 19 97 0D 96 50 99 0C 19 97 0C 20 B1 86 00 97
0490 48 97 47 97 46 97 45 97 42 97 41 97 3E 86 04 C6
04A0 03 CE 00 41 8D 27 26 F7 86 08 97 49 8D 52 8D 6E
04B0 24 05 7C 00 48 20 F7 8D 47 7A 00 49 27 18 8D 5E
04C0 86 04 C6 08 CE 00 49 8D 04 26 F7 20 DF 0C 09 69
04D0 00 5A 26 FA 4A 39 86 04 C6 03 CE 00 3E 66 00 08
04E0 5A 26 FA 4A 26 F2 C6 02 D7 49 CE 00 47 09 A6 00
04F0 E6 02 A7 02 E7 00 7A 00 49 26 F2 DE 47 DF 43 39
0500 C6 03 CE 00 3E 86 99 A0 00 A7 00 08 5A 26 F6 0D
0510 C6 03 09 86 00 A9 00 19 A7 00 5A 26 F5 39 C6 03
0520 CE 00 41 0C 09 A6 00 A9 03 19 A7 03 5A 26 F5 39
0530 DE 05 DF 43 7F 00 44 CE 00 60 DF 3F BD 64 8D DE
0540 04 DF 3F 96 44 97 40 DE 0E DF 43 BD 64 8D DE 44
0550 DF 14 39 DE 12 DF 43 DE 14 DF 3F BD 64 8D 96 44
0560 97 02 DE 45 DF 43 CE 01 67 DF 3F BD 64 8D 96 44
0570 97 03 39 DE 08 DF 43 DE 0E DF 3F BD 64 8D DE 44
0580 DF 52 39 DE 0C DF 43 DE 52 DF 3F BD 64 8D DE 44
0590 DF 33 39 DE 1A DF 43 CE 36 00 DF 3F BD 64 8D DE
05A0 44 DF 57 DE 57 DF 3F DE 55 DF 43 BD 64 8D DE 44
05B0 DF 31 39 DE 1C DF 43 DE 1A DF 3F BD 64 8D DE 45
05C0 DF 57 DE 5E DF 43 DE 5C DF 3F BD 64 8D 96 1C 85
05D0 10 27 13 DE 44 8D 06 DE 45 20 0F DE 45 DF 43 DE
05E0 57 DF 3F 7E 64 8D 8D F3 DE 44 DF 35 39 96 06 81
05F0 09 27 1A 81 02 27 19 96 5B 81 01 27 16 81 02 27
0600 15 81 03 27 14 81 04 23 13 7F 00 5B 39 7E 65 B3
0610 7E 65 93 7E 65 30 7E 65 53 7E 65 73 7E 65 83 B6
0620 80 07 F6 80 06 85 80 27 04 BD 66 D8 3B 86 01 97
0630 59 B6 80 05 F6 80 04 8A 3F 81 BF 27 09 81 7F 27
0640 0A 81 FF 27 0B 3B BD 66 59 20 0A BD 66 A3 20 05
0650 BD 66 59 20 F6 7F 00 59 3B CE 00 30 8D 35 CE 00
0660 11 8D 30 CE 00 3A 8D 2B DE 10 9C 1A 27 01 39 7F
0670 00 10 7F 00 11 CE 00 0F 8D 19 CE 00 13 8D 01 39
0680 8D 06 25 03 09 8D 01 39 86 99 80 01 0D A9 00 19
0690 A7 00 39 A6 00 8B 01 19 A7 00 09 A6 00 89 00 19
06A0 A7 00 39 CE 00 0B 8D EB CE 00 19 8D E6 CE 00 3C
06B0 8D E1 DE 0A 9C 1C 27 07 DE 18 9C 1C 27 0E 39 7F
06C0 00 0A 7F 00 0B CE 00 09 8D C9 20 EC CE 00 0D 7F
06D0 00 18 7F 00 19 8D A9 39 96 06 4C 81 0F 27 07 97
06E0 06 81 08 27 04 39 7F 00 06 D6 1E C1 00 27 34 81
06F0 08 27 F2 96 07 4C 97 07 DE 39 DF 55 7F 00 39 7F
0700 00 3A 81 3C 27 2F D6 1C C5 10 26 07 84 07 81 07
0710 27 01 39 DE 3B DF 5E DE 2F DF 5C CE 00 00 DF 3B
0720 DF 2F 39 D6 2C 81 08 27 06 C4 7F D7 2C 20 C0 CA
0730 80 D7 2C 20 BA 7F 00 07 CE 00 05 A6 00 8D 01 39
0740 81 59 27 06 8B 01 19 A7 00 39 6F 00 09 A6 00 20
0750 F3 CE 00 2D A6 00 81 7E 27 0E 81 FE 27 0A 81 80
0760 27 06 81 00 27 02 20 0C 84 80 A7 00 09 8C 00 2A
0770 27 F4 20 E0 86 00 B7 80 04 96 22 DE 24 B7 80 06
0780 D6 59 C1 00 26 0B D6 54 C1 00 26 05 B1 80 06 26
0790 28 E6 00 F7 80 04 08 8C 00 2F 27 07 48 DF 24 97
07A0 22 20 09 CE 00 2A 86 08 DF 24 97 22 96 59 81 00
07B0 20 01 3B B6 80 06 97 5A 3B B6 80 06 97 54 96 22
07C0 97 23 3B 8E 00 7F 7F 00 2E 7F 00 54 86 08 97 22
07D0 97 23 CE 00 2A DF 24 7F 00 59 CE 00 00 DF 5C DF
07E0 5E DF 2F DF 55 DF 3B DF 39 DF 31 DF 35 DF 52 DF
07F0 5A 7E 60 00 FF FF FF FF 66 1F FF FF 67 51 67 C3
```