A darkroom exposure/process timer Part 2

Peter Ihnat

This article completes the construction and assembly of the project, explains how it works and how to use it.

HAVING TESTED your power supply/relay/buzzer board and got it working, and assembled the '662a and '662d boards to the front panel and completed their wiring up, next thing to tackle is the mains wiring.

Connect the mains input exactly as detailed in the wiring diagram here but, as mentioned previously, leave the mains connection to the relay until the very last.

At last, the *moment of truth*. Switch the device on. All the displays should come on briefly and then 00.00 should appear on the display. If all is well, UNPLUG THE UNIT FROM THE MAINS and finally wire the mains connections as required to the relay.

The remote unit can now be assembled if required. Two pushbuttons are simply soldered onto the pc board which then mounts into the bottom half of the small plastic case. If the recommended case is used then four bolts will self tap into the mounting pillars provided for this purpose. On the prototype, I spaced the board off these pillars by the thickness of three washers since the buttons are a bit too short to come through the top of the unit.

The stereo 3.5 mm jack plug can be connected to the unit by a suitable length of 3-core cable, for example figure-8 shielded cable. The length will depend on the distance from the enlarger to the wet area in your darkroom.

One last point of interest — keep all leads clear of the ceramic transducer. The particular one I used is a vibrating case type and loses volume if anything touches it.

Using it

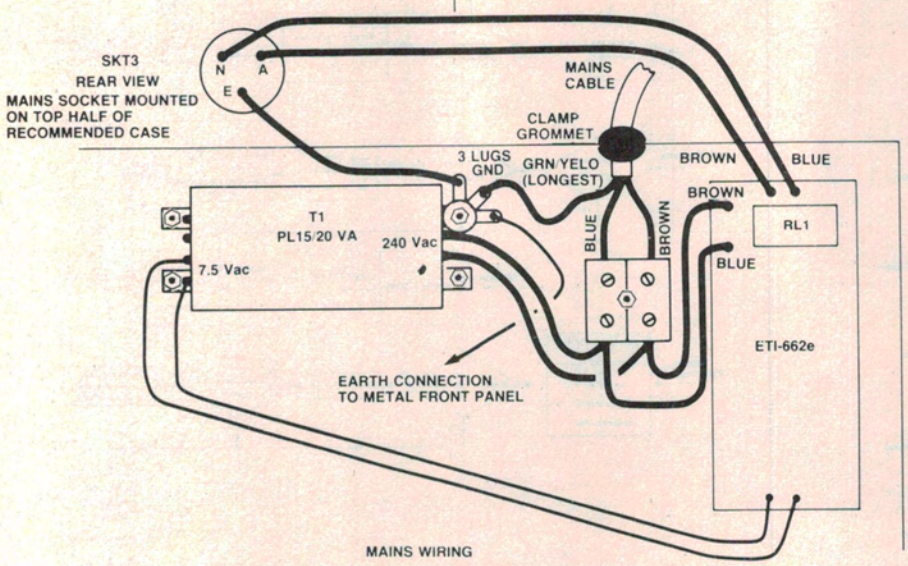
Firstly, plug your enlarger (or colour head power supply) into the output socket on the back of the unit. If your processing area is away from the enlarger, plug the remote unit in and position it somewhere convenient in that area. Now, let's program the unit.

Each timer in the unit must be programmed separately — actually, only the time currently being displayed can be changed. The SELECT button is used to select which timer is currently being displayed (the two LEDs below the main display indicate which). All you need to do is press MINS or SECS to set the required time — holding the button down causes the time to increment automatically. The FWD and BACK buttons allow each of the five exposure or ten process times to be accessed and programmed.

Let's look at a programming example. Assume we need 10, 7 and 5 seconds to be programmed as the three exposures to produce a certain print. To process it, the times are: four minutes developer, one minute for stop bath, four minutes in the fixer and 30 minutes wash.

Press SELECT a few times until the 'exposure' LED comes on.

Use SECS to set 00.10 (10 seconds) on the display.



MAINS WIRING

Press FWD (forward) to program the next time.

Use SECS to set 00.07 on the display. Press FWD.

Use SECs again to set 00.05 on the display.

The exposure timer is now ready for use. If you press BACK at any time then your previous entries can be checked or even modified. If the GO/STOP button is pressed, the enlarger will come on for the time period currently being displayed. Since we would like to start the exposure sequence with the first entered time, press exposure RESET twice (within 1/2 sec) and the first exposure time will be displayed.

Exposures may now commence and a press of ON/OFF will switch the enlarger on for the length of time being displayed. At the end of this time, the enlarger will switch off and the next programmed time will be displayed. After you have finished the three exposures, the timer automatically resets to the first programmed exposure time (because the next one hasn't been set and is therefore 00.00).

If during an exposure you press the ON/OFF button, the enlarger will switch off. Pressing the button again will result in the exposure continuing from where it stopped. If you want the particular exposure step to start from scratch, simply press RESET before pressing ON/OFF. Remember that pressing RESET twice within 1/2 second resets the entire exposure sequence to the first programmed time.

One very important function of an exposure timer is to allow the operator to switch the enlarger ON indefinitely so that a negative can be loaded and focussed. This is achieved by pressing ON/OFF twice within 1/2 second. To switch the enlarger OFF again, press either RESET or ON/OFF.

Now for the process timer. Press SELECT until the 'process' LED lights. Use MINS, SECS and FWD to set the following times — 04.00, 01.00, 04.00 and 30.00 as previously described. The RESET button here operates in a similar manner to the exposure RESET. The GO/STOP button is used to start and stop the timing process. Note that at the end of each step, the timer stops and needs to be restarted (by pressing GO/STOP on either the main or remote units).

If you want to use one of the built-in process sequences, simply press PROG and then either BACK, FWD, MINS or SECS. Table 1 lists which processes are available but to help you remember where they are, stick a small Scotchcal label on the front of the unit (see photograph). The processes ▶

can be used exactly as stored or you can modify any of the steps to suit your own processing methods (note that modifications are not permanent and are lost as soon as the unit is switched OFF).

Another press of SELECT blanks the 7-segment display and allows only the LED array to be displayed. If total darkness is required, another press of SELECT blanks the displays completely (the timers still work as described even though the displays are OFF). Since you may want to use this mode quite frequently, attach those little sticky pads (you know, the ones that glow in the dark) onto the tops of the four right-hand buttons. This makes them easy to find in the dark and stops you from pressing the wrong button.

SUMMARY OF KEYBOARD FUNC-TIONS

SELECT: Pressing this button places the timer in one of four modes. These are:

1. display off
2. display current exposure time
3. display current process time
4. display analogue process time (bargraph only). Changes to stored entries can only be made to the one currently being displayed.

SECS: Used to modify the 'seconds' part of the currently displayed time. A quick press increments it by one second. If held down, the time increments at an accelerated rate.

MINS: As for SECS except that the 'minutes' part of the currently displayed time is incremented.

FWD: This displays the next stored time which can be either one of five exposure or 10 process times, depending on the mode set by the SELECT button.

BACK: As for FWD except that the previous entry is displayed.

PROG: This is used to select one of the pre-programmed processes as listed in Table 1. When pressed, the display shows 'PROG'. Pressing one of BACK, FWD, MINS or SECS loads the corresponding processing times (any other button cancels this function).

GO/STOP: This pushbutton works in toggle fashion. When pressed, the current process time (whether displayed or not) starts to decrement. If pressed again before zero is reached, the timer stops. Another press continues the timing from where it stopped.

The internal buzzer sounds 15 seconds before the end of the current timing period to indicate that chemicals may be drained from the processing tank and those for the next step poured in. At the end of the time period, the buzzer sounds for one second and the next process time is loaded. Timing continues with a press of GO/STOP.

RESET (proc): When pressed, the current process time is reset to its original value. This is a useful function since accidents do happen and GO/STOP could be pressed before things are set to go. A single press of RESET sets the current time to its original value.

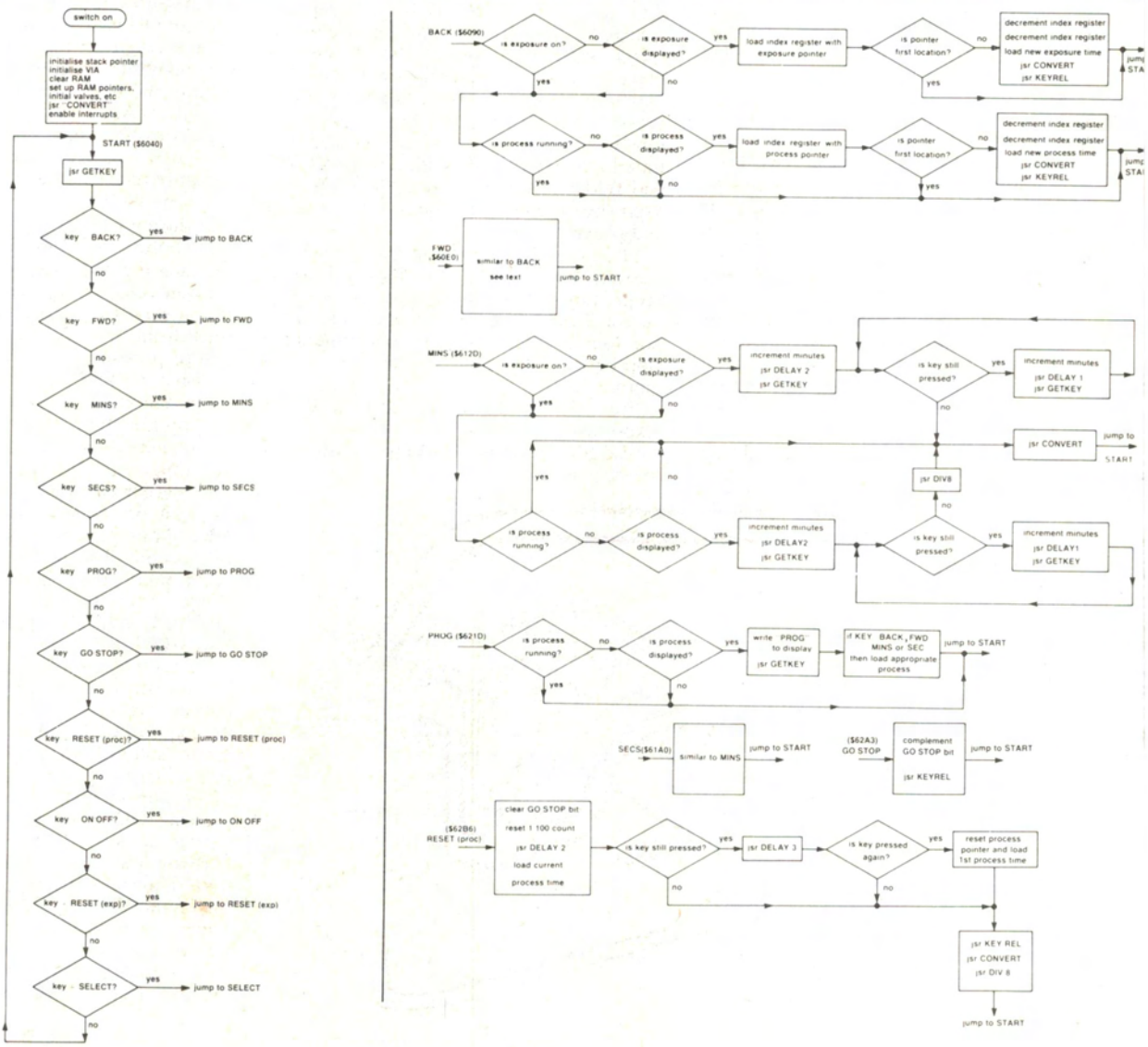
If the button is pressed twice within 0.5 second, the timer resets to the very first process time. This is used most frequently to initialise the timer after just entering all the steps for some process.

ON/OFF: Like the GO/STOP button, this works in toggle fashion. When pressed, the internal relay operates and applies power to the enlarger. The current exposure time (whether displayed or not) decrements and when zero is reached, the enlarger switches off. The timer can be interrupted by a press during the timing period resulting in the enlarger turning off. Another press continues timing.

When the enlarger needs to be switched on indefinitely for focussing purposes, press ON/OFF twice within 0.5 second. It can be switched off by pressing either RESET (exp) or ON/OFF. Note — due to the way the focus function is implemented, be careful when exposing prints. *One quick press* of the ON/OFF button is recommended since a sloppy press could leave the device in focus mode!

RESET (exp): As for RESET (proc) except the exposure timer is involved.

Figure 1.



EPROM PROGRAM

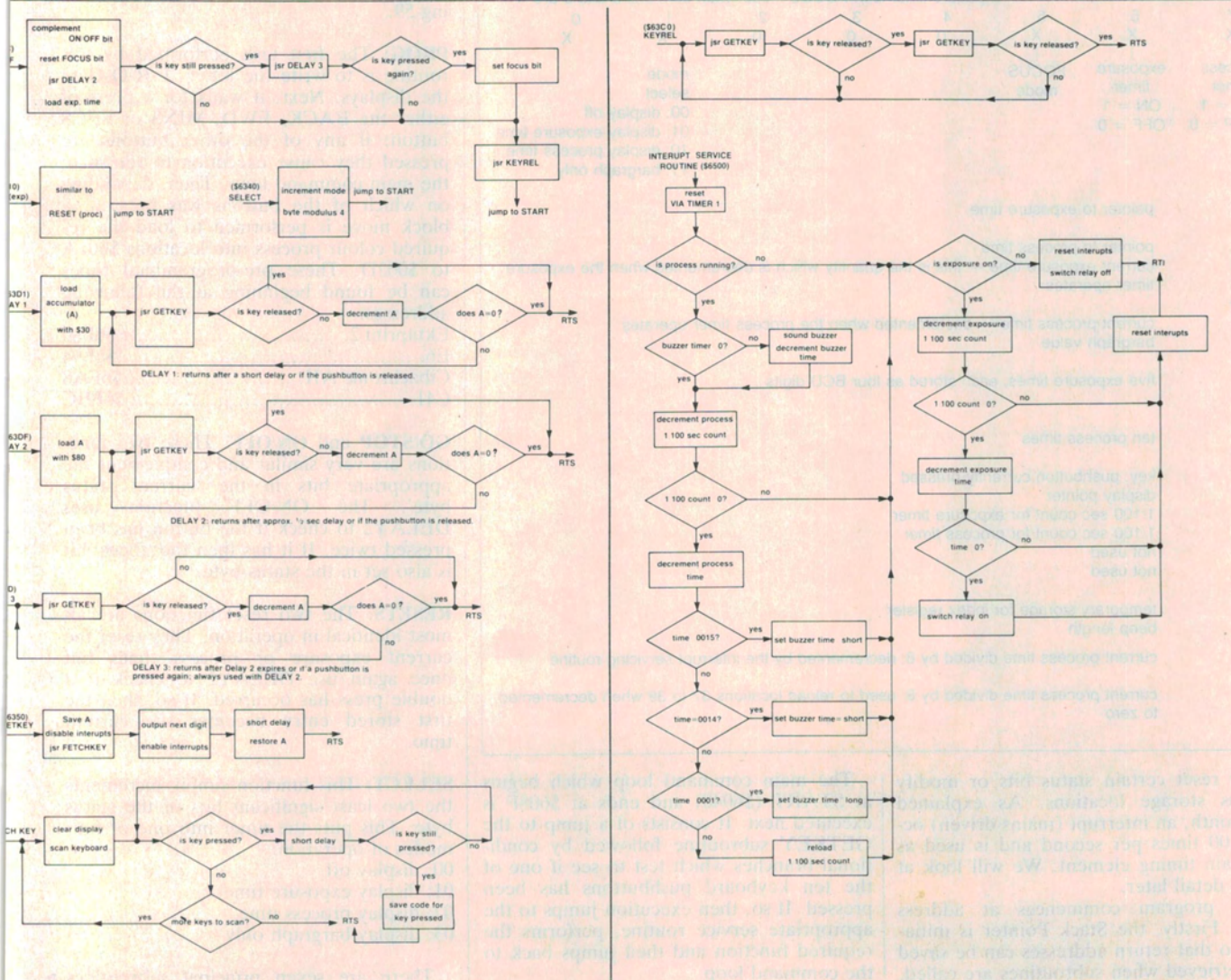
```
Addr  0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
6000: 8E 00 7F 86 FF B7 40 0C 86 3F B7 40 04 86 DD B7
6010: 40 03 7F 40 06 86 30 B7 40 0A 86 D0 B7 40 07 CE
6020: 00 7F 6F 00 09 26 FB 86 01 97 06 86 10 97 08 86
6030: 1A 97 0A 86 08 97 0F CE 64 64 DF 30 BD 64 00 0E
6040: BD 63 5D 20 2E C1 99 27 26 C1 AF 27 2B C1 B8 27
6050: 24 C1 AF 27 23 C1 B7 27 22 C1 BE 27 21 C1 BD 27
6060: 20 C1 7E 27 1F C1 7D 27 1E C1 77 27 1D 20 D1 7E
6070: 60 90 7E 60 E0 7E 61 2D 7E 61 A0 7E 62 1D 7E 62
6080: A3 7E 62 B6 7E 62 EC 7E 63 10 7E 63 40 01 01 01
6090: 96 06 85 40 26 1E 84 03 81 01 26 18 DE 07 8C 00
60A0: 10 27 0E 09 09 DF 07 EE 00 DF 0B BD 64 00 BD 63
60B0: C0 7E 60 40 96 06 85 80 26 F7 84 03 81 02 26 F1
60C0: DE 09 8C 00 1E 27 11 09 09 DF 09 EE 00 DF 0D BD
60D0: 64 90 BD 64 00 BD 63 C0 7E 60 40 01 01 01 01 01
60E0: 96 06 85 40 26 1E 84 03 81 01 26 18 DE 07 8C 00
60F0: 10 27 0E 08 08 DF 07 EE 00 DF 0B BD 64 00 BD 63
6100: C0 7E 60 40 96 06 85 80 26 F7 84 03 81 02 26 F1
6110: DE 09 8C 00 2C 27 11 08 08 DF 09 EE 00 DF 0D BD
6120: 64 90 BD 64 00 BD 63 C0 7E 60 40 01 01 96 06 85
6130: 40 26 32 84 03 81 01 26 2C 96 08 8B 01 19 97 08
6140: DE 09 8C 00 63 DF D6 2E 27 0F 96 08 8B 01 19 97
6150: 0B BD 64 00 BD 63 D1 20 ED DE 07 96 0B A7 00 BD
6160: 64 00 7E 60 40 96 06 85 80 26 F7 84 03 81 02 26
6170: F1 96 0D BB 01 19 97 0D BD 64 00 BD 63 DF D6 2E
6180: 27 0F 96 0D BB 01 19 97 0D BD 64 00 BD 63 D1 20
6190: ED DE 09 96 0D A7 00 BD 64 00 7E 60 40 01 01 01
61A0: 96 06 85 40 26 32 84 03 81 01 26 2C 96 0C BD 62
61B0: 14 97 0C BD 64 00 BD 63 DF D6 2E 27 0F 96 0C BD
61C0: 62 14 97 0C BD 64 00 BD 63 D1 20 ED DE 07 96 0C
61D0: A7 01 BD 64 00 7E 60 40 96 06 85 80 26 F7 84 03
61E0: 81 02 26 F1 96 0E BD 62 14 97 0E BD 64 00 BD 63
61F0: DF D6 2E 27 0F 96 0E BD 62 14 97 0E BD 64 00 BD
6200: 63 D1 20 ED DE 09 96 0E A7 01 BD 64 90 BD 64 00
6210: 7E 60 40 01 BB 01 19 81 60 26 01 4F 39 96 06 85
6220: 80 26 33 84 03 81 02 26 2D 7F 00 01 CE 5A D7 DF
6230: 02 CE 0A BF DF 04 BD 63 C0 BD 63 50 D6 2E 27 F9
6240: C1 99 27 1B C1 AF 27 19 C1 5F 27 1A C1 6F 27 1B
6250: 7F 00 02 BD 64 00 BD 63 C0 7E 60 40 CE 6F 92 20
6260: 0D CE 6F A6 20 0B CE 6F BA 20 03 CE 6F CE DF 34
6270: CE 00 2C DF 09 DE 34 A6 00 E6 01 09 09 DF 34 DE
6280: 09 A7 00 E7 01 BC 00 1A 27 06 09 09 DF 30 20 E5
6290: EE 00 DF 0B 7F 00 02 BD 64 90 BD 64 00 BD 63 C0
62A0: 7E 60 40 96 06 B8 B0 97 06 86 64 97 31 BD 64 90
62B0: BD 63 C0 7E 60 40 96 06 84 7F 97 06 86 64 97 31
62C0: BD 63 DF DE 09 EE 00 DF 0D 5D 27 0F BD 63 ED 5D
62D0: 27 09 CE 00 1A DF 09 EE 00 DF 0D BD 64 90 BD 64
62E0: 00 BD 63 C0 7E 60 40 01 01 01 01 96 06 B8 40
62F0: 84 C3 97 06 BD 63 DF 5D 27 10 BD 63 ED 5D 27 0A
6300: 96 06 8A 20 97 06 86 64 97 30 BD 63 C0 7E 60 40
6310: 01 96 06 84 83 97 06 86 64 97 30 BD 63 DF DE 09
6320: 00 DF 0B 5D 27 0F BD 63 ED 5D 27 09 CE 00 10 DF
6330: 03 EE 00 DF 0B BD 64 00 BD 63 C0 7E 60 40 01 01
6340: 96 06 4C 84 E3 97 06 BD 64 00 BD 63 C0 7E 60 40
6350: 30 37 7F 00 2E DF DE 2E DF 34 BD 63 90 BD 63 90
6360: B6 20 D6 2F 5D 27 04 44 5A 20 F9 B7 40 00 DE 34
6370: A6 00 B7 40 08 96 2F 4C 81 06 26 01 4F 97 2F 0E
6380: 86 FF 4A 26 FD 86 FF 4A 26 FD 33 32 39 01 01 01
6390: 7F 40 0B 86 20 BD 1D CA 3F C1 FF 27 13 C6 0F 5A
63A0: 26 FD BD 10 CA 3F C1 FF 27 EB F6 40 00 D7 2E 39
63B0: 44 26 E2 39 43 B7 40 00 F6 40 00 43 39 01 01 01
63C0: BD 63 50 7D 00 2E 26 FB BD 63 50 7D 00 2E 26 F0
63D0: 39 C6 10 BD 63 50 7D 00 2E 27 03 5A 26 F5 39 C6
63E0: 80 BD 63 50 7D 00 2E 27 03 5A 26 F5 39 BD 63 50
63F0: 7D 00 2E 26 03 5A 26 F5 39 01 01 01 01 01 01 01
6400: DE 02 8C 5A D7 26 01 39 96 06 84 03 27 31 81 02
6410: 27 0D 81 03 27 15 86 40 97 00 CE 00 08 20 07 86
6420: 04 97 00 CE 00 0D BD 64 4E 20 03 BD 64 42 CE 6F
6430: D0 96 0F 4D 27 04 4A 0B 20 F9 A6 00 97 01 39 4F
6440: 97 01 4F 97 00 97 02 97 03 97 04 97 05 39 DF 34
6450: A6 00 44 44 44 44 BD 64 81 97 05 DE 34 A6 00 84
6460: 0F BD 64 81 8A 20 97 04 DE 34 A6 01 44 44 44 44
6470: BD 64 81 97 02 DE 34 A6 01 84 0F BD 64 81 97 03
6480: 39 CE 6F E0 4D 27 04 4A 0B 20 F9 A6 00 39 01 01
6490: CE 00 00 DF 37 DF 39 DF 3B C6 06 01 01 96 0D 9B
64A0: 38 19 97 38 86 00 99 37 19 97 37 5A 26 EF C6 04
64B0: 78 00 38 79 00 37 5A 26 F7 96 0E 9B 38 19 97 38
64C0: 86 00 99 37 19 97 37 C6 03 0C 76 00 37 76 00 38
64D0: 76 00 39 CE 00 3F BD 11 DE 0D BD 0E 0B BD 0B 5A 26
64E0: E8 86 08 97 0F BD 64 00 39 A6 00 85 80 27 04 84
64F0: 7F 8B 50 85 08 27 04 84 F7 8B 05 A7 00 A7 03 39
6500: 86 30 B7 40 0A 96 06 85 80 26 03 7E 65 AC B6 40
6510: 03 7D 00 36 27 07 7A 00 36 B6 20 20 02 B4 DF B7
6520: 40 03 DE 0B 6C 00 00 27 5C CE 00 39 BD 66 20 81
6530: 99 26 0C 09 BD 66 20 81 99 26 04 09 BD 66 20 7D
6540: 00 39 26 15 7D 00 38 26 10 7D 00 37 26 0B 7A 00
6550: DF DE 3A DF 37 96 3C 97 39 7A 00 31 26 4E B6 64
6560: 97 31 CE 00 0D BD 66 38 DE 0D 8C 00 15 27 10 8C
6570: 00 14 27 0B 8C 00 01 26 33 B6 64 97 36 20 2D B6
6580: 32 97 36 20 27 96 06 84 63 97 06 DE 09 8C 00 2C
6590: 26 03 CE 00 18 08 08 A6 00 26 07 A6 01 26 03 CE
65A0: 00 1A DF 09 E3 00 DF 0D B6 08 97 0F 96 06 85 40
65B0: 27 4C 85 20 26 58 DE 0B 8C 00 00 27 16 7A 00 30
65C0: 26 4C 86 64 97 30 CE 00 0B BD 66 38 DE 0B 8C 00
65D0: 00 26 3B B6 40 03 84 FD B7 40 03 96 06 84 83 97
65E0: 06 DE 07 8C 00 18 26 03 CE 00 0E 08 08 A6 00 26
65F0: 07 A6 01 26 03 CE 00 10 DF 07 EE 00 DF 0B 96 06
6600: 84 83 97 06 B6 40 03 84 FD B7 40 03 20 08 B6 40
6610: 03 8A 02 B7 40 03 BD 64 00 B6 90 B7 40 0B 3B 01
6620: A6 00 4A 81 FF 26 04 86 99 20 09 16 C4 0F C1 0F
6630: 26 02 80 06 A7 00 39 01 A6 01 4A 16 81 FF 26 06
6640: 80 A6 A7 01 20 0B C4 0F C1 0F 26 02 80 06 A7 01
6650: 39 A6 00 4A 16 C4 0F C1 0F 26 02 80 06 A7 00 39
6660: 00 30 02 30 00 30 00 30 00 30 00 30 00 30 00 30
6670: 00 30 00 00 07 00 01 00 01 00 02 00 06 00 02 00
6680: 07 00 04 00 06 00 01 00 03 00 00 30 03 00 03 00
6690: 03 00 00 00 00 00 00 00 00 00 00 00 03 15 06 30
66A0: 03 15 06 30 03 15 01 30 00 00 00 00 00 00 00 00
66B0: 00 04 06 B6 B7 97 B7 F7 FF 00 00 00 00 00 00 00
66C0: BD C0 9B D3 C6 57 5F C1 DF C7 00 00 00 00 00 00
66D0: 50 2E 20 49 48 4E 41 54 65 00 60 00 60 00 60 00
```

Software operation

The ETI-662a General Purpose Microprocessor may be a versatile device but it can't do a thing unless it has been programmed for some intended task. The flow chart for the Darkroom Timer program is shown in Figure 1 and a hex listing is given elsewhere in this article. Unfortunately, a complete assembler listing can not be given but reference can be made to the hex listing while reading the following descriptions. Note also that the program can be shortened considerably if it was required to fit into a 1K or 2K-style EPROM. Luckily, the General Purpose microprocessor uses a 2732 (4K) EPROM which has more than enough room — besides, the program is easier to understand and follow in its present form.

The General Purpose microprocessor assumes that ROM is located between $6000 and $6FFF (the $ sign meaning hex) and the VIA's (Versatile Interface Adapter) registers are between $4000 and $400F inclusive (see ETI-662a project, April '84 for a discussion on memory decoding).

At switch-on, a hardware reset is performed and the microprocessor commences 'processing' by fetching the *address* of the first instruction from the ▶



last two locations in memory. These two locations are 6FFE and 6FFF and contain 60 and 00, respectively.

The complete program can be broken down into five sections comprising *initialisation, main command loop, pushbutton service* routines, *housekeeping* subroutines and *interrupt service* routine.

Basically, after switch-on the initialisation is performed and then the main command loop is entered. In this loop the displays are multiplexed and a check is made for a pushbutton press. The microprocessor exits the loop only briefly to respond to either a pushbutton press or an interrupt. In fact, the microprocessor spends most of its time in the command loop.

The pushbutton service routines simply set or reset certain status bits or modify various storage locations. As explained last month, an interrupt (mains driven) occurs 100 times per second and is used as the main timing element. We will look at this in detail later.

The program commences at address $6000. Firstly, the Stack Pointer is initialised so that return addresses can be saved and retrieved when subroutines are called. Secondly, the VIA is set-up so that I/O operations can be performed. This is accomplished by configuring Port A of the VIA as an 8-bit outport port and Port B as six output and two input lines.

The How It Works in last month's article describes the multiplexing procedure used for I/O. Control lines CA2 and CB2 are configured as outputs and are set low to ensure that the buzzer and relay are both off. Next, the conditions for interrupts are set-up (described later).

Finally, various memory locations are 'pre-loaded' with data which will be used for a variety of purposes. These locations are shown in the accompanying panel and are modified, tested, incremented or decremented as required by the main program.

The main command loop which begins at START ($6040) and ends at $608F is executed next. It consists of a jump to the GETKEY subroutine followed by conditional branches which test to see if one of the ten keyboard pushbuttons has been pressed. If so, then execution jumps to the appropriate service routine, performs the required function and then jumps back to the command loop.

Basically, the service routines operate as follows.

BACK: The current exposure or process time is normally loaded from the memory location pointed to by the exposure or process pointer. The BACK function simply decrements this pointer and loads the new time into the current time locations ($000B, 000C or $000D, 000E). Note that it also checks to see if it hasn't decremented past the first stored entry.

FWD: This is very similar to the BACK function except that the exposure or process pointer is incremented and a check is made to see if the last entry isn't passed.

MINS: This particular routine increments the 'minute' part of the currently displayed time making use of two delay subroutines. DELAY2 produces a delay of about half a second. If the button is still held after this, then the minutes increment at an increased rate which is determined by DELAY1, a relatively short delay subroutine. The flow diagram shows the actions of these delays.

SECS: As for MINS except seconds are incremented and reset to zero after passing 59.

PROG: The first task performed by this routine is to write the letters P-R-O-G to the displays. Next, it waits for a press of either the BACK, FWD, MINS or SECS button; if any of the other buttons are pressed they cause execution to return to the main command loop. Then, depending on which of the buttons was pressed, a block move is performed to load the required colour process into locations $001A to $002D. These pre-programmed times can be found beginning at the following addresses:

Ektaprint 2. .....$6F80
E6. .....$6F94
Cibachrome AII. $6FA8
C41. ..... $6FBC

GO/STOP and **ON/OFF:** These two functions are very similar and complement the appropriate bits in the 'current status byte'. The ON/OFF function uses DELAY2 to check if the button has been pressed twice. If it has then the 'focus' bit is also set in the status byte.

RESETS: The two reset functions are almost identical in operation. They reset the current exposure or process times but once again use DELAY2 to check if a double press has occurred. If so, then the first stored entry becomes the current time.

SELECT: This function simply increments the two least significant bits of the status byte. This puts the timer into one of four modes of operation:
00: display off
01: display exposure time
02: display process time
03: display bargraph only

There are seven principal subroutines ▶

'PRE-LOADED' MEMORY LOCATIONS

Address	Function
0000	display storage area: mode indication
0001	display storage area: bargraph segments
0002	display storage area: tens-of-seconds
0003	display storage area: seconds
0004	display storage area: minutes
0005	display storage area: tens-of-minutes
0006	current status byte: a '1' in a bit position indicates that the timer is running; the exposure timer is in FOCUS mode when both bits 6 and 5 are '1'.

bit: 7 (X) process timer GO = 1 STOP = 0 | 6 (X) exposure timer ON = 1 OFF = 0 | 5 (X) FOCUS mode | 4 (0) | 3 (0) | 2 (0) | 1 (X) | 0 (X) mode select 00: display off 01: display exposure time 10: display process time 11: bargraph only

0007	
0008	pointer to exposure time
0009	
000A	pointer to process time
000B	current exposure time — this is the quantity which is decremented when the exposure
000C	timer operates
000D	
000E	current process time — decremented when the process timer operates
000F	bargraph value
0010	
to	five exposure times, each stored as four BCD digits
0019	
001A	
to	ten process times
002D	
002E	key: pushbutton currently pressed
002F	display pointer
0030	1/100 sec count for exposure timer
0031	1/100 sec count for process timer
0032	not used
0033	not used
0034	
0035	temporary storage for index register
0036	beep length
0037	
0038	current process time divided by 8; decremented by the interrupt servicing routine
003A	
003B	current process time divided by 8; used to reload locations 37 to 39 when decremented
003C	to zero

which are used by many of the service routines. The GETKEY subroutine is probably the most important of these since it multiplexes the displays and scans the pushbuttons to determine if one has been pressed. The multiplexing procedure has been described in previous articles and will not be described here.

Depending on the pushbutton pressed, the subroutine returns with one of the following values in the KEY byte (location $002E). **BACK:** $9F, **FWD:** $AF, **MINS:** $5F, **PROG:** $B7, **GO/STOP:** $BE, **RESET (proc):** $BD, **ON/OFF:** $7E, **RESET (exp):** $7D, **SELECT:** $77, no pushbutton pressed: $00.

The key release (KEYREL) subroutine is one that is used at the end of most of the service routines. It prevents the microprocessor from returning to the main command loop until the currently pressed pushbutton is released, otherwise execution will return to the command loop and almost immediately jump to the keyboard service routine again, back to the command loop, etc until the button is released.

Three of the subroutines produce delays of varying degrees. DELAY1 simply produces a short delay (fraction of a second) and is used to set the automatic incrementing speed when either minutes or seconds are set. DELAY2, about half a second, is the delay used when checking to see if RESET or ON/OFF have been pressed twice in quick succession.

Note that a return-from-subroutine (RTS) is performed if either the required delay expires or if the currently pressed pushbutton is released. The two cases are distinguished by checking the KEY byte where a 00 indicates that the key has been released. DELAY 3 is used only after DELAY2 and basically continues DELAY 2 until it times-out or a pushbutton is pressed again (the double press of RESET or ON/OFF).

The remaining subroutines are **CONVERT** and **DIV8. CONVERT** is called at the end of programs which produce changes to the quantities being displayed; for example, after decrementing the displayed time or incrementing the seconds part of the time, etc. It separates the current time into its four individual digits and looks up the corresponding 7-segment data for each digit from the look-up table stored between $6FDO and $6FD9. This data is then stored in the correct place in the display storage area, ready to be multiplexed to the displays.

DIV8 is used to divide the current process time into eight equal intervals which define the bargraph 'step'. In actual fact, the subroutine multiplies the minutes by 60, adds the seconds, multiplies by 100 and divides the lot by eight. The multiplication by 100 is to convert the time into hundredths of a second so that it can be decremented each time an interrupt occurs.

The interrupt service routine performs the actual time keeping functions as well as operating the bargraph display and switching the relay and buzzer outputs. The 100 Hz pulses from the mains are fed into the VIA's CB1 control line which is configured as an interrupt input. Unfortunately, it isn't possible for this line to be an independent interrupt line (independent in this case meaning that reading or writing to port A and port B of the VIA does not clear any status bits which indicate that a pulse has occurred on the line).

Independent interrupts are possible on the CA2 and CB2 lines but these are already being used to control the buzzer and relay. The CB1 line is really a handshake line used when full handshaking is required in I/O operations.

So, in the present project, when a timing pulse is produced by the mains, the CB1 line sets an interrupt bit in register 13 of the VIA which in turn pulls the IRQ line to the microprocessor low indicating that an interrupt condition exists. However, if this occurs precisely when the GETKEY subroutine outputs to the display or inputs from the pushbuttons, the very act of inputting and outputting *resets the interrupt* condition resulting in the interrupt being ignored. I found this out the hard way when the prototype lost three seconds per minute — not good for a timer!

On closer inspection, I noticed that CB1 can be used as an external clock input to a shift register inside the VIA. This operates as follows: when a byte is loaded into the shift register, an internal modulo-8 counter is reset. Then the next eight pulses into CB1 move data around the register and after eight shifts, an interrupt is generated.

It seemed OK on the surface but when implemented the prototype still lost time — approximately one second per hour. It seems that the external clock pulse is ignored when it switches state simultaneously with the microprocessor E signal. This is not referred to in the 6522 data sheet and I am still investigating the problem.

To produce a reliable timing signal, I implemented two interrupt modes — the normal 'handshake' interrupt for CB1 described above and the Timer 1 one-shot mode. The latter is initialised when a number is loaded into the timer which then decrements at 1 MHz (frequency of the E signal). An interrupt is generated when it reaches zero.

If, however, the timer is reloaded before it times-out, then the interrupt is prevented. I used this 're-triggerability' property of the timer to act as a 'missing interrupt' detector. Basically, it operates as follows. The number which is loaded onto the timer produces a delay of just over 1/100 second and is initialised at the beginning of the interrupt service routine. Program execution continues normally until the 100 Hz signal produces the next interrupt.

Once again, the timer is initialised preventing it from interrupting as well. However, if the normal mains generated interrupt is missed, then the timer times-out and produces the interrupt *instead*, thus ensuring that accurate timing is maintained.

The rest of the interrupt routine merely decrements the current time and switches the two outputs as required. The flow diagram shows this clearly. ●

BACK: KODAK EKTAPRINT 2 (38°C)

1.	PREWASH	30 secs
2.	DEVELOP	2 min. 30 sec.
3.	STOP	30 secs
4.	WASH	30 secs
5.	BLIX	30 secs
6.	WASH	30 secs
7.	WASH	30 secs
8.	WASH	30 secs
9.	WASH	30 secs

FWD: KODAK E-6 (37.8°C)

1.	1ST DEVELOP	7 mins.
2.	WASH	1 min.
3.	WASH	1 min.
4.	REVERSAL	2 min.
5.	COLOUR DEV.	6 min.
6.	CONDITIONER	2 min.
7.	BLEACH	7 min.
8.	FIXER	4 min.
9.	WASH	6 min.
10.	STABILISER	1 min.

MINS: ILFORD CIBACHROME AII (24°C)

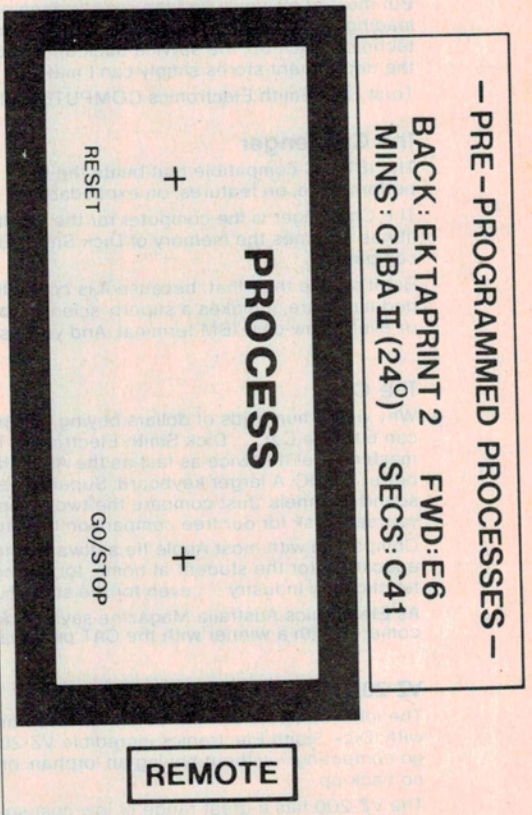
1.	DEVELOP	3 min.
2.	WASH	30 sec.
3.	BLEACH	3 min.
4.	FIX	3 min.
5.	WASH	3 min.

SECS: KODAK C-41

1.	DEVELOP	3 min. 15 sec.
2.	BLEACH	6 min. 30 sec.
3.	WASH	3 min. 15 sec.
4.	FIXER	6 min. 30 sec.
5.	WASH	3 min. 15 sec.
6.	STABILISER	1 min. 30 sec.

The above processes are preprogrammed in the darkroom timer. They can be accessed by pressing PROG, followed by either BACK, FWD, MINS or SECS, as appropriate.

TABLE 1.



NOTE: front panel artwork is on page 161.