Repairing HP9800 Desktop Calculators – Part 5

Tony Duell (ard@p850ug1.demon.co.uk)

Storage System Theory of Operation

Since the HP9800 machines use volatile semiconductor RAM, a magnetic backing storage system was fitted to all models so that programs and data could be saved and reloaded. The HP9810 and HP9820 have a magnetic card reader, the HP9830 has a digital cassette tape drive.

1 HP9810/HP9820 Magnetic card reader

The magnetic card reader records 3 data tracks and a clock (also known as 'timing' or 'strobe') track in each direction on a magnetic card. The system consists of a motorised mechanism to pull the card past the 4-coil head and the **Card Reader Interface** PCB which links the mechanism to the processor **I/O Interface** PCB.

1.1 Card reader mechanism

The card reader mechanism consists of a 24V DC permanent-magnet motor with an integral gearbox, the output shaft of which is coupled to the card feed roller via a belt drive. This roller draws the magnetic card past a read/write head. Four optical sensors, each consisting of a filament lamp and a photodetector located so that the card blocks the light beam as it passes detect the leading and trailing edges of the card and also the presence or absence of a notch in the leading edge of the card, used to write-protect it.

1.2 Card reader interface

The **Card Reader Interface** PC is quite simple because most of the timing is controlled by the processor firmware..

There are 2 outputs from the **I/O interface PCB** used to select the card reader. **CROut** is asserted by U19a on this PCB when **I/OCmd(3)** and **Q(1)** are both high, that is when an 'STC' machine is executed with internal select code 2. **CRSOut** is asserted by U1c when **Q(1)** and **RdInst** are high. The latter signal is asserted by U8d on the **Clock** PCB when **Q(9)** is low and **I/OCmd(0)** is asserted. So **CRSOut** is asserted when an 'STF' instruction with select code 2 is executed. These 2 signals write the low 6 bits of the processor I/O register into a 6 bit output port built form D-type flip-flops on the **Card Reader Interface** PCB.

Bit 5 of the I/O register is loaded into U5a by **CROut**. The output of this flip-flop is buffered by a 2-transistor circuit which connects the positive side of the motor to the +12V supply when the flip-flop is set. Since the negative side of the motor is connected to the -12V rail, 24V appears across the motor which thus rotates.

The card sensor photodetector signals are amplified by 2-transistor circuits and fed to **DI(3)**...**DI(6)**. These transistors are cut off by **SenseEn** when the motor is not running. 2 of the sensor outputs are diode-ANDed to give the **RdEn** signal which enables the read amplifiers.

Bit 4 of the I/O register is latched in U8a, again by **CROut**. The output of this flip-flop, **WrD**, disables the write drivers when it is set. It appears to be set throughout read operations but cleared during write operations.

The 4 write circuits are essentially the same, so only track A will be described. Bit 0 of the I/O register is copied into U8b, which drives the inverter U7c. When U8b is set, the output of U7c is low, saturating the transistor and making the voltage on its collector close to +12V. The coupling diode is reverse-biased, so no current flows in the head winding. When U8b is set, the transistor is cut off, and a current flows through the head from the =12V supply rail via the coupling diode and 2.7k resistor. U9e clamps the head drive signal to ground, preventing writing during read operations.

There is one difference for the 'S' clock write channel. The output flip-flop, U5b, is loaded by **CRSOut** rather than **CROut**. This makes it easier to delay the clock pulse with respect to the data pulses.

The 4 read amplifiers are, again, identical. The signal from the head is amplified by an operational amplifier such as U4, and fed to a discrete-transistor amplifier and schmitt trigger circuit. The latter is enabled by **RdEn** from the card sensor circuitry when there is a card in the reader track.

The output of the clock channel is fed to a 2-transistor circuit which produces a positive-going pulse on **CRRdClk** for a magnetic transition of either polarity on the card. This signal loads the outputs of the 3 data channels, suitably buffered, into the read flip-flops such as U10b. These can be read into the processor I/O register via the associated open-collector inverters, such as U9c. When there is no card in the reader, **RdEn** being deasserted asserts **ClrRd**, forcing all the read flip-flops to be set and thus the outputs of the inverters to be in the high-impedance state. Thus other devices can drive the i/O register inputs.

CRRdClk/ is fed to the **I/O Interface** PCB where it sets the SR flip-flop U4c, U4d. This causes the first gate of U9 to assert **I/O flag** when a card reader I/O instruction is executed. This is used to indicate the presence of new data in the read flip-flops. This flip-flop is cleared by U2d when **I/OStb** is asserted at the end of an I/O instruction with bit 9 set (this is detected by U2a on the **Clock** PCB along with **Q(1)** being set, that is select code 2.

2 Cassette drive

The digital cassette drive fitted to the HP9830 uses digital-grade tape in cassettes similar to audio Compact Cassette. Unlike normal audio cassette recorders, though, the tape is transported simply by turning the spools, there is no capstan and pinch roller. Although this means the tape does not travel at constant speed, the encoding system used on the HP9830 in not sensitive to this.

The entire width of the tape is used at once, the cassettes are not turned over like audio cassettes. Two tracks are recorded on the tape. A pulse on one track indicates a logic 0 bit. A pulse on the other track indicates a logic 1 bit. And a pulse on the 2 tracks simultaneously is used as a start-of-byte marker. This is known as 'Bit-Mark-Sequence' or 'BMS' in some documentation

An optical EOT sensor detects the clear leader tape in the cassette. It consists of a small filament lamp and light-dependent resistor (LDR) mounted so that light from the bulb is reflected off the cassette shell and onto the LDR when the clear leader (rather than opaque tape) passes in front of the sensor assembly.

2.1 Motor control

The tape drive is fitted with a pair of permanent-magnet DEC motors, one for each tape spool. The motors are mounted on a metal beam, pivoted at the mid-point. This beam is rocked by a pair of solenoids, and this engages the motor spindle with a rubber-tyred wheel attached to each tape spindle.

The **Motor Control** PCB selects a particular motor and its associated solenoid to control the tape direction and regulates the speed of that motor. The direction control system takes in 2 signals from the **Tape Controller** PCB. **MotorOn/** goes low to start tape motion, causing the output of U4c to go high, enabling U4b and U4d. The **Dir** direction control signal, inverted by U4a thus causes the output of one of those gates to go low., turning on the associated PNP transistor. This drives a pair of NPN transistors, one of which energises the solenoid, the other grounds the negative side of the selected motor.

The motor speed controller senses the current through the selected motor by measuring the voltage drop across a 10Ω resistor in series with it (this is the resistor connected between TP2 and TP3 on the **Motor Control**) PCB. This voltage is amplified by U1 and compared with a reference level by U2. The output of U2 controls the motor pass transistor, Q9 on the **PSU** PCB (it is located there simply because it needs to be fitted on a heatsink), thus controlling the motor current and therefore the motor speed.

The **Speed** input from the **Tape Controller** PCB is applied to U3a, the output of which varies the reference voltage level to U2. When **Speed** is low. the reference voltage is increased, so the motor turns faster.

The output of the LDR in the optical EOT sensor is converted to a digital signal by U1 on the **Tape Drive Connector** PCB acting as a comparator. When the tape leader is detected, **EOT** goes low, causing the output of U1a on the **Tape Interface** PCB to go high. If the motor is running, monostable U6 is triggered, and when this times out, U5b is set (since the output of U4b must be 1 when **MotorOn/** is low). This asserts **SpeedClamp** via U4c.

When **EOT** goes high again (tape, rather than leader in front of the sensor), U5b is directly cleared, deasserting **SpeedClamp**. U5b is also set when there is no tape in the drive (**CassSw** is low) or during power-on initialisation (**ClrMark**, from U17f on the **Tape Controller** PCB is low).

Back on the **Motor Control** PCB, **SpeedClamp** causes U3b to clamp the motor speed control line to the voltage of its associated zener diode. This prevents the motor running too fast and causing damage when the tape stalls.

2.2 Low level read/write system

This section translates between the conventional clock and data signals and the BMS format described earlier. It also interface to the tape head itself.

For a write operation, the output of U5b on the **Tape Controller** PCB goes high, asserting **Write**. On the **Read/Write** PCB, U4c ANDs this with the output of the write-protect switch (**WpSw/**), preventing writing on protected cassettes. **WrEn** enables the write encoder circuit.

During writing, if **WrData** is high when **WrClk** goes high, the output of U8b goes low, and forces **WA** high via U7a. Similarly, if **WrData** is low (making the output of U6a high) when **WrClk** goes high, the output of U8c goes low and that of U7b, **WB** goes high. Finally, when **WrMark** is asserted, the output of U8a goes low, asserting both **WA** and **WB**. In this way the incoming data is converted to the BMS format.

The head drivers, located on the **Tape Head** PCB are identical in operation, so only the 'A' track will be described. **WA** is inverted by U1 and thus the outputs of U1b and U1c are in antiphase. This provides the signal to the head winding. The power to U1 is controlled by a transistor on the **Read/Write** PCB which is turned on via U6e during writing, there is no write current in the head at other times.

The 2 read amplifiers are identical, so as before I will only describe the 'A' channel. The output from the head winding is amplified by U2a on the **Tape Head** PCB and fed, as an analogue signal (**RA**) to the comparator U1b on the **Read/Write** PCB. The threshold of this comparator is determined by its current state via the 2 transistor circuit connected to its output, thus providing hysteresis and also by the state of the **Speed** signal via U6f, U4d and the associated transistors. When the tape is running fast for searching, the comparator threshold is increased, improving the noise immunity. The output of U1b is fed to U4a, which is enabled during reading by U6e. The **TTLRA** and **TTLRB** signals at the outputs of U4a and U4b are logic signals that indicate when pulses have occurred on the tape tracks.

When there is a pulse on both tracks at the same time, the output of U7c goes low, clearing U3a. This asserts **RdMark**.

When a pulse occurs on track 'A' and no mark has been detected, U7d clears U5b, thus making **RdData** high. When a pulse occurs on track 'B' or a mark has been detected, U5c sets U5b, making **RdData** high.

A pulse on either (or both) tracks makes the output of U5a pulse low. This triggers the monostable U2. During reading, when no mark has been detected, the output of U5d is high, and thus the output of U6b low. When U2 is triggered, therefor, its Q/ goes low causing **RdClk** to pulse low via U5b and U6c.

Thus the BMS data on the tape is separated into clock, data and mark signals.

2.3 Tape control

The **CO(n)** peripheral select lines are decoded on th **Tape Interface** PCB, and when select code 1010_2 is detected, the output of U13a goes high. If **CEO** is also high, U11c asserts **StatEn** enabling the gates in U7, which pass various status signals to the **SI(n)** inputs to the I/O register. If **CEO** is low, **TapeSel** is asserted by U11b, and, this, after being buffered by U11a, enables the data output gates in U8 and U9 so the tape data can be read into the I/O register (the source of this data will be explained shortly).

On the **Tape Controller** PCB, **TapeSel** triggers monostable U10. U10 enables decoder U8, which decodes **SO(0)**...**SO(2)** and asserts **RunCmd/** for any of the 4 read commands or the write command. This latches the **SO(n)** lines in the command latch U1. The 2 least-significant bits of U1are fed to the **Motor Control** PCB via U6a and U6b, to control the tape speed and direction.

RunCmd also sets the run flip-flop U4b, which asserts **MotorOn** via U6c, starting the tape motion. The run flip-flop is cleared at the end of the tape (**SpeedClamp** goes high, thus clocking a 0 into U4b via U16a and U17e), or when **Stop** is asserted by U12c. This occurs at power-on (from U17f), when a stop command is given to the interface (**StopCmd/** is set low by U8), or when there is no tape in the drive.

Manual rewinding is controlled by U4a. This is set by the 'Rewind' key on the keyboard, and its output causes the output of U5c to go high if there is no tape operation in progress. This forces all 3 motor control signals low, causing the tape to rewind at high speed. U4a is cleared either when a tape command is started (**RunCmd** clocks a 0 into U4a), or when the tape ends or when one of the 'stop' conditions mentioned earlier occurs.

2.4 Tape data register

Data is stored on the tape in 9-bit characters, each character is preceded by a mark. Of the 9 bits, 8 of them are transfered to or from the least significant byte of the processor I/O register. The centre bit is, however, loaded from bit 3 of the command latch U1 on the **Tape Controller** PCB and indicates the file marker character. The centre bit of the nine is used for this since it is in the same position relative to the marks no matter which way the tape is being read, enabling file markers to be found during both forward and reverse scans of the tape.

The data register consists of the D-type flip-flop U20b and the two 4-bit registers U19 and U18, cascaded in the usual manner. U15a and U15b provide a parallel load facility for the most significant bit.

The timing of the data transfer is controlled by the 10-state counter U3. U7c asserts **WrMarkEn/** when this counter gets to its final state of 1001_2. Data bits are transfered in the 9 states when **WrMarkEn/** is not asserted.

Although these 2 sections are common to both reading and writing data, it is simpler to consider their operation and that of the associated control in these 2 cases separately

2.4.1 Tape writing

When the tape system is selected by the CPU during a write command, **LdSR** is asserted by U5a, thus transferring data from the processor I/O register to the tape data register.

Since **write** is asserted by U5b during write commands, the write clock oscillator U2 and U9 is enabled, The reset-to-9 input of U3 is enabled (and the reset-to-0 disabled via U17d) so when the write operation is started, U11b is set by the rising edge of **write**, forcing U3 to state 1001_2 and asserting **WrMarkEn/**. This enables U13b via U14f, casing the first **RawWrClk** pulse to assert **WrMark** and thus write a mark on the tape. **WrMark** also clears U11b via U14d, thus enabling the counter U3 to count.

While **WrMarkEn/** is not asserted, U7d is enabled, and the shift register is clocked by the write clock oscillator. U7d also provides the **WrClk** signal to the data encoder described above. The data input to the encoder is connected to the least significant output bit of the tape data register, Thus the 9 bits in the shift register are transfered to the tame, then **WrMarkEn/** is asserted to write the marker for the start of the next 9 bit word, and the sequence repeats.

2.4.2 Tape reading

When the tape is read, the write clock oscillator is disabled since **Write** is low. The output of U7d is forced high, and thus U7b passes the read clock signal, **RdClk** from the data decoder to the shift register. When a data bit is detected on the tape, it is shifted into the data register, The bit counter U3 is also clocked by **RdClk**, and is set to 0 when U20a is set by a mark being read from the tape. **WrMarkEn** is asserted by U7c when 9 bits have been read from the tape.

The parallel outputs of the data register are transfered to the processor I/O register via the data output gates in U8 and U9 on the **Tape Interface** PCB, enabled by U11a

2.5 Tape flag and interrupt

The tape flag signal is produced by U12b on the **Tape Controller** PCB. When the tape system is writing the output of U6d is forced low, forcing the output of U12a high. Therefore the flag signal is controlled by U15c. Its output goes low, causing a flag, when **WrMark** is asserted, indicating that a new word should be loaded into the shift register. U11a is set at the start of a write operation, inhibiting the flag signal for the mark that is then generated.

During reading, U15c's output is always high. The flag is generated by U12a, normally when 9 bits have been read from the tape, since U7a's output is forced high by the **Bit4** output of the command latch being low. However, by setting this bit, the tape flag will only be asserted when **Bit4Out/** from the data registers is asserted after 9 bits have been read, that is when a file marker has been detected. As mentioned above, this is independent of the direction of tape movement.

The tape flag signal sets U5a on the **Tape Interface** PCB, thus asserting **BusI/ORd/**. This causes the output of U7c on the **I/O interface** PCB to go high, triggering the monostable U8 and copying the data from the tape data register into the I/O register. U8 also sets the flip-flop U13a, asserting the processor I/O flag via the second gate in U9.

During a scan for file markers (that is, **Bit4** is high and therefore so is **BuffBit4**), the tape system can interrupt the processor. When the tape system is not selected by the processor and **BuffBit4** is high, the reset input of U14a is deasserted. This flip-flop is therefore set by a rising edge at the output of U2c,which can occur in 3 ways : if the tape runs to the clear leader, and thus **BuffSC** is asserted causing the output of U1c to go low; if the cassette is removed, causing the output of U1b to go low; or if the output of U2b goes low, which occurs when the file marker character is found. The last occurs when bits 2...5 of the tape data register are all set and a tape flag occurs, but since **Bit4** must be high to enable the interrupt system in the first place, the centre bit (**Bit4Out/** of the shift register must also be asserted to enable **TapeFlag/** as described above.

Interrupts other than the file marker being found cause the output of U3a to go low, setting U5a and causing **BusI/ORd/** to be asserted, which has the effects described earlier.

When the interrupt occurs and U14a is set, **KeyDn** is asserted via U3c, interrupting the processor. U3b pulls **SI(1)** low, thus enabling the processor to determine the source of the interrupt.

The output of U14a, **StopTape/** causes the output of U12c on the **Tape Controller** PCB to go low, thus clearing the run flip-flop U4b and stopping the tape.

2.6 Beeper

The HP9830 incorporates a fixed-frequency, fixed-duration beeper which is described here simply because it is located on the **Tape Interface** PCB.

The otherwise-redundant **CROut** signal (used for card reader output in the HP9810 and HP9820 and described in that section) triggers the monostable U12 when a beep is required. This enables U15c, passing the output of the oscillator U15a, U15b to the loudspeaker driver transistor via U15d. Thus the speaker is driven. The positive side of the speaker is connected to the output of an emitter follower, the base of which is connected to a 22μF capacitor. When U12 is not triggered this capacitor is charged via the diode and 470Ω resistor. When U12 is triggered and the beep is produced, this capacitor discharges via the 3.9kΩ resistor thus gradually reducing the voltage applied to the loudspeaker and giving a decaying amplitude envelope.

Power Supply Theory of Operation

The power supply provides the following DC voltages from the AC mains input :

+5V : Logic power supply

+12V : Supply for the ROM selection inputs, and for the analogue parts of the storage system.

-12V : Used to provide the ROM bias supply and also for the analogue parts of the storage system

+16V : Supply for the PMOS RAMs

+20V or +19.5V : Actually referenced to the +16V supply, this is the bias rail for the PMOS RAMs.

+24V : This is only present in the HP9810 and HP9820 machines, and is used for the printer motor and thermal printhead.

The power supply also provides the **Init/** signal which sets the processor microcode program counter to 1717 when the power is first turned on.

The power supply consists of the mains input circuitry mounted mostly on the rear panel of the machine., This feeds a double-wound transformer which supplies lower voltage AC to the PSU PCB(s). The outputs of the transformer are rectified and smoothed, and then regulated down to the voltages given above. In all cases, the +5V regulator is s switch-mode design, while all other voltages are produced by linear regulators.

Although the HP9830 power supply is physically different from that in the other 2 machines, the circuitry is very similar and thus all the power supplies will be described together. The main physical difference is that in the HP9810 and HP9820, the transformer secondary windings are wired to a pair of edge connectors mounted in the base of the machine. Two of the power supply PCBs plug both into those connectors and into connectors on the main backplane, These PCBs connect to the appropriate transformer secondary connections on the case-mounted connectors and provide the DC voltage to the main backplane. One of these PCBs also passes AC to the third PSU PCB which only plugs into the main backplane. In the HP9830, the transformer secondary windings are connected to a free edge connector which is plugged into the rear edge of the main backplane, The AC outputs of the transformer are fed over the backplane to the single PSU PCB, the regulated outputs from which are again fed over the main backplane.

1 Mains input and transformer

AC mains enters the machine via a standard IEC60320 C14 plug[1]. The live (phase) wire is fused at 6A, and the supply is connected to 2 or 3 (depending on the machine) IEC sockets (EN60320-2-2 Sheet F type) which can be used to power peripheral devices. Mains is also fed through an interference filter circuit (which may either be an encapsulated module or discrete components) and then to an in-line connector for the mains switch fitted to the keyboard.

When the switch is turned on, the mains is then applied to the primary of the power transformer via a further fuse (1A for 220V or 240V mains, 2A for 110V or 120V mains). When the switch it turned off, a 1MΩ resistor is connected across the output of the mains filter, ensuring the capacitors are discharged when the mains plug is removed and preventing (admittedly very minor) electric shocks from the exposed plug pins.

The mains transformer has 2 primary windings, each 120V with a tap at

[1]This is a stanard mains input connector as used on many other devices, Miud you, if you have trouble finding where to feed the mains in I think you will have even greater trouble repairing the processor.

the 100V position, A pair of slide switches on the rear of the machine connects these windings for different mains voltages as follows :

100V : Mains is supplied to the two 100V sections of the primary windings connected in parallel

120V : The two 120V primary windings are connected in parallel across the mains

220V : One entire 120V winding and the 100V section of the other winding are connected in series. Mains is connected to the ends of this combination.

240V : Both primary windings are connected in series across the mains.

The machine's cooling fan has a 120V AC (shaded-pole) motor and is connected across one of the 120V primary windings. For 120V mains, it ends up being simply connected to the mains, for other mains voltages the transformer primary windings act as an autotransformer to supply 120V to the fan.

Whether the mains transformer has one multiply-tapped secondary winding, or several with the centre taps joined together is something that I have no determined. Electrically it makes no difference. The center-tap of the winding is connected to the system ground rail via the PSU PCB, symmetrical outputs voltages from the transformer are bi-phase rectified and smoothed by the PSU circuitry.

2 Regulators

The regulators are based round a type 723 IC, which contains a reference voltage source and an error amplifier. The operation of a linear regulator circuit is to compare a voltage derived from the appropriate output voltage with a known reference voltage using the error amplifier and use the output of the latter to control a pass transistor which in turn controls that voltage output of the supply. The +12V supply uses the internal reference of the associated 723, all other supplies use the +12V supply as their reference.

The DC inputs to the regulator circuits are produced by bi-phase rectifiers and smoothing capacitors.

All the voltage outputs apart from +19V have crowbar circuits. Each crowbar circuit consists of an SCR (thyristor) triggered by a zener diode. If the power supply voltage exceeds that of the zener diode, the SCR receives gate current, turns on, and thus shorts that power supply rail to ground, protecting the rest of the machine.

2.0.1 A couple of RPL programs

I investigated the operation of the regulators using my HP49G calculator, and found the following RPL programs to be helpful :

'PAR' calculates the equivalent resistance of 2 resistors in parallel :

```
PAR : \<< INV SWAP INV + INV \>>
```

'PDIV' calculates the voltage ratio for a potential divider :

```
PDIV : \<< SWAP OVER + / \>>
```

2.1 +12V regulator

Since the +12V supply is used as a reference for most of the other voltage rails, I shall begin with this one. It is based round U1 on the **+/-12V PSU** PCB in the HP9810 and HP9820, or U2 on the **PSU** PCB in the HP9830. The output of this regulator IC controls an emitter-follower pass transistor which in turn controls the voltage of the +12V output The **+12Sense** signal is connected to the +12V rail on the backplane, and measures the voltage there. This means that the voltage drop across the PCB edge connector is irrelevant, the voltage at the load is what is measured.

The operation of the 723 in this circuit is to try to make the voltages on its 2 inputs equal. The voltage on **In+** is easy to determine. It's simply the reference voltage, **Vref** from that IC, and the data sheet gives a typical value of 7.15V for this.

The voltage at **In-** is determined by the output voltage and a potential divider. The first thing to do is to calculate the division ratio :

```
499. 788. 11500. PAR PDIV
```

Now, due to the operation of the 722, the output voltage multiplied by that ratio will be the reference voltage. So :

```
7.15 SWAP /
```

will give the output voltage, which is very close to 12V. The 11.5kΩ resistor may have a different value in different machines to compensate for slight changes in the reference voltage of the 723.

The current sensing inputs of the 723, **IL** and **IS** are connected across the sense resistor (1Ω in the HP9810 and HP9820, 0.75Ω in the HP9830). If too much current is drawn from the power supply, the increased voltage drop across this resistor will cause the 723 to remove the drive from the pass transistor, shutting the supply down.

2.2 -12V regulator

This regulator, being designed to give a negative output voltage, is possibly the most difficult to understand. The pass transistor is a PNP device and is biased on by the resistor connected between its collector and base. The output of the 723 (U2 on the **+/-12V PSU** PCB in the HP9810 and HP9820 or U4 on the **PSU** PCB in the HP9830) reduces the absolute vale of the the base voltage of this transistor, reducing the output voltage. As a result of this, the 'sense' voltage is applied to **In+** and the 'reference' voltage to **In-**. As before, **=12Sense** is connected to the -12V rail on the backplane.

The next thing to do is to calculate those two voltages. Let the output voltage be 'V' Then the voltage on **In+** can be calculated by :

```
4530. DUP PDIV 'V' *
```

Or you could notice that a potential divider with equal resistors divides the voltage in half.

The voltage on **In-** is first calculated relative to the +12V rail and then 12 is added to the result to make it relative to the ground rail :

```
3010. 9090. PDIV 'V' 12. - * 12. +
```

The operation of the 723 is to make those equal, so set them equal on the calculator and isolate 'V' :

```
= 'V' ISOL
```

This gives an output voltage close to -12V as expected.

In this negative regulator, the internal current limit system of the 723 cannot be used. Instead, the B-E junction of a PNP transistor is connected across the current sense resistor. When the voltage drop across this resistor exceeds 0.7V, the transistor is biased on, and removes the base drive from the pass transistor.

2.3 +16V regulator

The +16V regulator, based around U1 on the **+16V/+20V/+24V PSU** PCB in the HP9810 and HP9820 or U3 on the **PSU** PCB in the HP9830 is very similar to the +12V regulator. The operation of the 723, its pass transistor and the current limiter is identical.

In the case of the HP9810 and HP9820, the 'reference' voltage at **In+** is given by :

```
681. 953. PDIV 12 *
```

The operation of the 723 means that this is equal to the output voltage divided by the potential divider circuit connected to the **In-** input. This division ratio is calculated by

```
887. 698. PDIV
```

And of course

```
INV *
```

gives the output voltage, which is close to 16V

For the resistor values used in the HP9830, the output voltage can be calculated by :

```
499. 698. PDIV 12 * 887 698 PDIV INV *
```

which is again close to 16V

2.4 +20V (or +19.5V) regulator

The (very low current) memory bias supply is provided by a simple zener diode circuit 'on top of' the +16V rail. In the HP9810 or HP9820, the current is provided by a resistor from the +24V supply. In the HP9830, a constant current source based round Q10 on the **PSU** PCB is used for this purpose.

2.5 +24V regulator (HP9810 and HP9820 only)

This circuit is again similar to the +12V regulator. The **In+** input is held at +16V, and therefore the output voltage, determined by the potential divider on the **In-** input is given by :

```
16. 806. 1620. PDIV /
```

which is close to 24V. The current limiting circuit of this regulator senses the base current of the pass transistor, dropped across the 4.7Ω resistor, and is not particularly effective. A short-circuit from the +24V rail to ground often burns out the 0.51Ω series resistor

2.6 +5V regulator

The +5V supply uses a switch-mode regulator. The control transistor (known as the chopper) is either saturated or cut off and therefore dissipates little power. When the output voltage is too low (compared to the reference voltage), the 723 (U1 on the **+5V PSU** PCB in the HP9810 or HP9820 or on the **PSU** PCB in the HP9830) turns on the chopper, applying the raw input voltage to the load via the series inductors. Due to the inductance of the latter, the output voltage does not rise instantaneously, but relatively slowly as the magnetic field around the inductor builds up. When the output voltage exceeds the reference voltage, the 723 cuts off the chopper transistor. The magnetic field around the inductor begins to collapse, and supplies power to the load via the flyback diode. Thus the output voltage has a small ripple (essentially determined by the hysteresis of the 723 error amplifier) around the required voltage.

The **+5Sense** signal is connected to **In-** on the 723, and thus the output voltage is equal to the reference voltage, set by the potential divider on **In+** which is given by :

which is close to 5V. There is no overcurrent protection in the +5V regulator, but if the output is shorted to ground (possibly via the crowbar circuit) then the NPN transistor connected to the **Comp** input of the 723 will be turned on. This will clamp **Comp** to ground, disabling the drive to the chopper transistor and thus shutting the regulator down.

3 Init signal

The **Init/** initialisation signal used to set all bits of the microcode program counter is produced by a monostable circuit based round U2 on the **+5V PSU** PCB in the HP9810 and HP9280 or U5 on the **PSU** PCB in the HP9830. This is triggered by the resistor-capacitor network on its input shortly after power-on and during its time period it asserts **Init/** via the NPN transistor buffer connected to its output.

That's it!. We have now covered every major section of the machine. Now we are ready to start the actual repair.