Static and dynamic RAMs are playing an increasingly important role in home computers these days. Data stored in static memory can be preserved for relatively long periods, (providing the power supply is not switched off). Where dynamic RAMs are concerned, long-term storage is a little more complicated, as all the data has to be renewed ('refreshed') at regular intervals to prevent it from being lost. Now that prices have dropped dramatically, dynamic RAMs are gaining the upper hand,

dynamic RAM card

and an FET switch. The voltage across the capacitor determines whether the data contents are 'high' or 'low'. A dynamic RAM memory cell takes up far less room than its static counterpart. Theoretically speaking, this would mean that the former is able to provide a much greater memory capacity on an identical chip surface area. But there is more to dynamic RAMs than meets the eye, for one of their main disadvantages is that a great deal more is involved in making a memory cell fully operative

dynamic RAM card

16 K in 8 ICs

Dynamic RAMs are so economical these days that it is worthwhile to use them instead of static RAMs, despite the additional control electronics required. Eight ICs can store up to 16 K and still leave plenty of room on the Eurocard for the control logic. Further advantages include low current consumption and high-speed access times. Computer owners who are running out of memory and space will welcome this opportunity to extend their RAM facilities. even though static RAMs have always been preferred in the past.

Dynamic vs static

Static RAMs have an advantage in that they are very easy to operate. The required circuitry is already incorporated inside the IC, so very few external components are called for. Life is also made easier for the operator by the fact that no timing problems are involved as long as IC types are selected for the right speed to cope with the application in hand.

A static RAM memory cell consists of a sort of set/reset flipflop, which contains at least 5 or 6 transistors. As readers can imagine, a complete RAM IC has an immensely complex structure.

A dynamic RAM, on the other hand, is based on capacitances rather than flipflops. Each cell consists of a capacitor than a capacitor and an FET switch. As a result of a slight leakage current in each capacitive unit, the voltage level across the capacitor slowly drops in value. Thus, in order to prevent the data stored in the capacitors from being lost, their charge must be refreshed from time to time. This calls for an additional control circuit and very precise timing for the operation to pass off smoothly. That is not the only problem. An awful lot of memory cells can be integrated on a single chip and so addressing is rather complicated. Dynamic RAM manufacturers have tried to solve this by using a multiplexed address bus, (yet another addition to the circuitry).

Nevertheless, dynamic RAMs are so cheap nowadays, (compared to their static rivals), that even the extra components required do not affect the overall cost. Although they consume very



82017 - 1

Figure 1. The pin assignment of the dynamic RAM IC 4116. Note that it requires three supply voltages, and that its current consumption is fairly low.



82017 - 2

Figure 2. Block diagram of the 4116. Its memory capacity is distributed among 128 columns. Sense amplifiers are located between the rows.

dynamic RAM card

little current, the dynamic RAMs used here do need three separate supply voltages.

All things considered, if the same results can be obtained for less money, there is no reason why readers shouldn't use dynamic RAMs.

The structure and operation of a dynamic RAM chip

The design described in this article is centred around the inexpensive 4116 IC, which is available from various manufacturers. The IC encompasses 16384 x 1 bits. 8 ICs therefore provide an 8 bit wide 16K memory. The IC series has access times, ranging from 150... 300 ns according to the figure indicated after the type number.

The 4116 memory is arranged in an array of 128 columns and 128 rows (128 x 128 = 16,384). To decode 1 of the 16,384 cell locations within the 4116, 14 address bits are required, seven per column and seven per row. An integrated clock, the Row Address Strobe (RAS) latches the 7 row address bits into the chip and a second clock, the Column Address Strobe (CAS), subsequently latches the 7 column address bits into the chip. In other words, the 7 address inputs are multiplexed. The pin assignment for the 4116 is shown in figure 7. A negative pulse at the RAS input 'reads in' data in the form of a row address into the address inputs and a negative pulse at the CAS input reads in the data as a column address. As the memory is only one bit wide, only one data input and one data output are required (Din and Dout). The logic state of the WRITE input determines whether a bit is to be read out or written in. The remaining four pins constitute the supply connections: VDD, VCC, VBB and VSS (+12 V, +5 V, -5 V and 0 V, respectively).

To come back to the internal structure of the IC, 128 sense amplifiers are situated in the middle of the 128 rows with the task of topping up the capacitors during a 'refresh cycle'. In addition, they transfer data to and from the memory locations. A sense amplifier is a flipflop, each input of which is connected to half a column. Each column has its own sense amplifier which detects the charge passing through an addressed row and amplifies the signal produced. The boosted signal is a full logic level, either 'high' or 'low' and is fed back to the column line, causing the original (amplified) logic level to be restored in the capacitor. The sense amplifier now contains the same data as the read (and immediately rewritten) capacitor. Thus, as soon as the row is accessed, all the logic levels stored in the capacitors belonging to that row are refreshed. To give you an idea of the capacitance level involved: a 4116 storage capacitor has a value of about 0.04 pF!

The order in which the different signals have to be applied is as follows:



WRITE CYCLE (EARLY WRITE)





Figure 3. Time sequence charts for reading, writing and refreshing the 4116. No specific times are given here, because they depend on the speed of the host processor. All times are in the nanosecond range.

4-30 - elektor april 1982



dynamic RAM card

Data is read out of a memory location, a seven bit address being stored at the address inputs beforehand. Then a pulse is generated at the RAS input. The row address must be available for a certain amount of time, after which the seven bit column address can be produced. This is followed by a pulse at the CAS input. The column address must also be present for a certain minimum period. An internal output buffer then sends the logic level of the selected address bit to the data output. During this procedure the WRITE input must be high. Virtually the same principle applies to write operations, only now the data input is initially provided with a logic level and the WRITE input goes low. The time sequence chart in figure 3 illustrates these events in the form of a graph.





Figure 5. The dynamic RAM circuit diagram. Using wire links the card may be adapted to different microprocessor systems.



Figure 6. The waveforms of the main signals in the circuit diagram. The abbreviations are also indicated in the circuit diagram together with the corresponding lines.

The refresh cycle

As mentioned earlier, using capacitors to store digital information has a number of advantages, but there is also another side to the coin. Slowly but surely, this type of capacitor loses its charge and the stored logic level. This is why it is necessary to refresh the level from time to time. In the case of the 4116 this must be done every two milliseconds, which is quite reasonable, considering its low cell capacitance.

Fortunately, the refresh operation is relatively straightforward, thanks to the structure of the IC, in which the 'sense' amplifiers are situated in the direct vicinity of the cells. The sense amplifiers, as has already been seen, boost the logic levels in the memory cells. When a row address is read in after the computer has generated a RAS pulse the entire row of 128 bits is read into the sense amplifiers. At the same time, the logic levels are amplified and written back into the 128 row capacitances. In other words, once a row address and an RAS pulse have been produced, the 128 bits are refreshed. As long as this method ensures that the whole operation is executed within 2 milliseconds, data stored inside the IC will remain intact

Of course, the refresh cycle may be shorter than 2 milliseconds, if necessary. This particular RAM card was designed to be used with the Junior Computer, or a similar microprocessor, with a clock frequency of 1 MHz. This means the refresh cycle for 128 rows takes 128 μ s.

The timing

Figure 3 contains the time sequence charts for the read, write and refresh cycles, respectively. The diagrams clearly show the order in which the various signals must be provided. Different times are involved and this will have to be taken into account. No specific values are indicated, as they vary somewhat per IC type and manufacturer.

The power supply

Special attention should be paid to the power supply of the dynamic RAM card. The average current consumption rate for the three supply voltages is fairly low. The highest peak is reached upon either edge of the RAS and/or CAS input. An example of this is shown in figure 4, where relatively high current peaks occur during the rising and falling edges of the signal. Up to 100 mA may be attained (per IC)!

Obviously, this calls for certain protective measures. Rather than provide the power supply with a high current capability it is best to buffer the power supply by placing capacitors around the RAMs.

The circuit diagram

Figure 5 shows the complete circuit diagram of a 16K dynamic RAM card.

IC12...IC19 constitute the 16K x 8 bit dynamic memory. The data inputs of the ICs are directly connected to the data pins of the connector (on the lefthand side of the drawing). The data outputs are connected to the data lines by way of tri-state buffers. Address lines $A\emptyset$... A13 are linked to IC9 and IC10, which each contain four multiplexers (with two inputs and one output). These multiplex the fourteen address lines in two groups of seven. The address lines are linked to the address inputs of the RAMs by way of the tri-state buffers N11...N17.

Address lines A12... A15 are connected to the address decoder IC11. This enables data to be stored in any address range by mounting wire links between the outputs of IC11 and gates N27 and N28.

IC6 serves to refresh the memory blocks regularly, as it acts as a seven-bit counter. The outputs of the IC are also linked to the address inputs of the RAMs by way of tri-state buffers (N20...N26). The refresh cycle takes place during the period that the processor is not using the address bus. The clock input of the counter IC6 and the

elektor april 1982 - 4-31

control inputs of the tri-state buffers, N20... N26, are connected to the clock \emptyset 1 of the processor system by way of gates N1... N3. During a certain part of the clock signal, memory is not accessed. The wire links shown are needed if the circuit is used with the Junior Computer. For the sake of clarity we will describe the rest of the circuit diagram with reference to the Junior Computer and then explain how it may be modified for use with other microprocessor systems.

In the case of the Junior Computer, memory is not accessed with the positive-going transition of Ø1 and so this can be used to refresh the stored information. The pulse diagram in figure 6 illustrates this. During each positive edge of Ø1 the contents of the counter are incremented by one. Buffers N11...N17 are disabled, as they are controlled by the output of N2 (which is inverted with respect to the output of N3). The buffers N20 . . . N26 then send an address to the address inputs of the RAMs. A delay is enforced, with the aid of MMV1 and MMV2, to allow a negative pulse to be provided at the RAS inputs of the RAMs shortly after the rising edge of the clock signal. That is sufficient to refresh a complete row. Since one row is refreshed per positivegoing clock pulse, the counter is reset 128 clock periods. After this after period all the rows will have been refreshed. Thus, a full refresh cycle lasts 128 μ s (at a clock frequency of 1 MHz). Addresses are read in and out on the negative edge of the Ø1 clock. This requires a certain amount of 'timing logic' with carefully calculated values (in nanoseconds) to be sure that the positive and negative edges reach the RAMs (and the multiplexer) in the right order. Three pulse 'delays' consisting of N4...N10, R1...R3 and C3...C5 are included for the purpose.

When an address is accessed in the RAM address range, the output of NAND gate N29 is pulled low by the address decoder. The clock signal is then sent to N7 and N9 by way of N31, which is also connected to Ø1. The falling edge of the clock is delayed by the R1/C3 combination and is fed to the RAS inputs via a couple of gates (see figure 6). This means that the first seven address bits are read into the RAMs. After this, the multiplexer must be activated, which is achieved by delaying the falling RAS edge through R3 and C5. Once the following seven address bits have been accessed, a falling edge may be produced at the CAS inputs. The latter edge is derived from the falling clock edge by way of the R2/C4 delay unit. The \overline{WE} inputs are directly linked to the corresponding connector pin.

That covers the main signals. A couple of gates and a flipflop are shown in the top left-hand corner of the circuit diagram. These simply serve to adapt the various signals to make them 'digestible' for processors other than the 6502. 7





Figure 7. The component overlay and the copper tracking pattern of the dynamic RAM printed circuit board. The wire links should be mounted according to the different μ P specifications in table 1.

Parts list

```
Resistors:
R1... R3 = 270 Ω
R4,R5 = 2k2
R6... R8 = 390 Ω
```

Capacitors: C1 = 33 p C2 = 100 p C3 = 68 p C4 = 470 p C5 = 120 p $C6 \dots C21 = 1 \mu/16 \text{ V}$ tantalum

Semiconductors: IC1,IC2 = 74LS14 IC3 = 74LS221 IC4,IC5,IC20 = 74LS244 IC6 = 74LS393 IC7 = 74LS08 IC8 = 74LS32 IC9,IC10 = 74LS157 IC11 = 74LS154 IC12 ... IC19 = 4116 (250 ns) IC21 = 74LS15 IC22 = 74LS74 (see text)

Miscellaneous:

1 x 64-pin DIN 41612, male connector

Construction

The printed circuit board for the dynamic RAM card is shown in figure 7. Since timing is very important, care must be taken when mounting the components. Readers are advised to abide by the indicated component values, especially with regard to the resistors and capacitors. The easiest method is to use the Elektor printed circuit board.

How the wire links are positioned depends on which processor is in use. Table 1 shows the requirements for the Junior Computer, the Z80 and the 8085. IC22 can be omitted for the 6502 and the Z80, as flipflop FF1 has also been left out. As far as the 8085 is concerned, things are different again. Unlike the Z80, the 8085 does not produce a refresh signal. Instead, this is generated by SØ, SI and INTA (which indicate the opcode fetch status). During the period that the processor needs to detect the code, the RAM is not being used and so a refresh cycle may take place. In the 8085 (multiplexed) addresses are accessed by way of a data bus. Since the dynamic RAM card is only suitable for a non-multiplexed bus, however, the data bus will have to be demultiplexed elsewhere in the 8085 system.

The connections between points V, W, X and Y and the outputs of IC11 define the address range. Each output of the IC represents an address range of 4K. The memory array is shown in table 2. A total of 16K therefore requires four outputs of IC11 to be linked to points V...Y. This enables data to be stored in blocks of 4K practically anywhere within the memory range.

Operators must remember one important aspect: the same code may not be used twice for A12 and A13 (see the last column in table 2), because the two address lines are both linked to the address inputs of the RAMs. This means that the wire links must be mounted in such a manner that the following combinations of A13 are stored in consecutive memory blocks:

13	A12
Ø	Ø
Ø	1
1	Ø
1	1

A

It can easily be deduced which combinations are feasible. A valid combination would, for instance, be blocks 8000, 9000, A000 and B000. But 0000, 4000, 8000 and C000 are totally out of the question, because A13 and A12 would be 00 for all blocks.

If the card is used in combination with the Junior Computer, the required supply voltages will already be available. The other processor systems will have to produce the required voltages using integrated voltage regulators. Plenty of power supplies meeting the requirements can be found in previous Elektor issues.

Testing the circuit

Before connecting up the supply voltages, it is a good idea to check all the solder joints thoroughly. Then the card may be plugged into the bus of the μ P system.

It doesn't really matter in which order the supply voltages are connected, although the manufacturer recommends constructors to start with the -5 V line. This provides an extra safety margin in the event of an overload (which is unlikely to happen if a good power supply is used).

If all is well the memory should function normally as soon as the power supply is switched on. As the memory locations are invisible to the naked eye, the best way to test the system is to read data in and out and compare the results. A special test program has been written for the purpose and is shown in table 3. This can also be used to test other types of RAM. Once the program has been entered, the start address and the end address of the memory range being tested must be stored at locations 0000 (= ADL) and 0001 (= ADH) and at locations 0002 (= ADL) and 0003 (= ADH), respectively. The program is then initialised at address 0004 and 00 is written into the memory range. The program checks whether ØØ is in fact stored at the first address of the rangeunder-test. If so, Ø1, Ø2, Ø4, Ø8, 10, 20, 40 and 80 are written into the address in succession and read out again at once. As a result, every bit in the address will have been high once. Subsequently, FF is stored at this address to track down any addressing errors. For if there is an found in a different address. The mistake is detected when FF is read out somewhere along the line.

The above procedure is applied to every single address until the program reaches the end of the test range. Then the entire test program is repeated (it also tests the operator's patience!) starting with the storage of \emptyset . This time the range is examined back to front. Again, this is necessary to be able to trace any addressing errors that might have cropped up.

If everything passed off without a hitch, address ØØØØ will appear on the display at the end of the program, followed by the low order address byte of the entered start address. If on the other hand an error was detected, the address at which it was found is shown on the display together with its (erroneous) contents. Restart the program at address ØØØA in order to carry on with the test.

Table 1		
6502	Z-80	8085
1-1'	1-1'	1-1'
A-B	2-2'	2-2'
C-D	J2	3-3'
E-F	J3	4-4'
G-H	J4	5-5'
J8	J5	J1
J9	JG	J2
	J9	J4
IC22 is omitte	d	J6
	IC22 is omitted	J9
		J10

Table 1. This indicates which links are required on the printed circuit board when using the 6502, the Z80 or the 8085.

Table 2

output	address				
IC5	4 Kbyte-block	A15	A14	A13	A12
Ø	0000 ØFFF	Ø	Ø	ø	Ø
1	10001FFF	Ø	Ø	Ø	1
2	2000 2FFF	Ø	Ø	1	Ø
3	3000 3FFF	Ø	Ø	1	1
4	4000 4FFF	Ø	1	Ø	Ø
5	5000 5FFF	Ø	1	Ø	1
6	6000 6FFF	Ø	1	1	Ø
7	7000 7FFF	Ø	1	1	1
8	8000 8FFF	1	Ø	Ø	Ø
9	9000 9FFF	1	Ø	Ø	1
A	A000 AFFF	1	Ø	1	Ø
в	BØØØ BFFF	1	Ø	1	1
C	CØØØ CFFF	1	1	Ø	Ø
D	DØØØ DFFF	1	1	Ø	1
E	EØØØ EFFF	1	1	1	Ø
F	FOOD FEEE	1	1	1	1

Table 2. The address range can be defined by linking the outputs of IC5 to points V, W, X and Y. Each connection provides 4K bytes, so that four connections are needed for a total of 16K.

dynamic RAM card

	0010:	0004					ORG	\$0004	
	0030:					*** RA	M TEST	PROGRA	M ***
	0060:					DEFINI	TIONS		
	0080:	0004				BEG	*	\$0000	BEGIN OF MEMORY
	0100:	0004				END	:	\$0002 \$00F6	END OF MEMORY CURRENT ADDRESS POINTER
	0120:	0004				POINT		\$00FA	MONITOR'S ADDRESS POINTER
	0130:	0004				PATTER	*	\$00E5	CURRENT TEST PATTERN
	0140:	0004				MONITO	•	51010	
	0160:								
	0170:	0004	20	45	00	RAMIST	JSR	CUREEG	FILL WORKSPACE WITH SOU CUR = BEG
	0190:					1000000			
	0200:	A 000	20	84	00	TSTA	JSR	WALK	WALKING BIT ROUTINE BRANCH TE MEMORY CELL IS DEFECT
	0220:	000F	A9	FF			LDAIM	SFF	TEST PATTERN FOR DOUBLE ADDRESSING
	0230:	0011	91	E6	00		STAIY	CUR	THERE AND CHECK CUP
	0250:	0016	80	F2	00		BCS	TSTA	TEST FINISHED?
	0260:	0018	20	45	00		JSK	WRZERO	FILL WORKSPACE WITH \$00
	0270:	0018	86	E6			STX	CUR	CHECK FROM BOTTOM TO TOP
	0290:	001F	A6	03			LDX	END	+01
	0300:	0021	86	E7			STX	CUR	+01
	0320:	0023	20	84	00	TSTB	JSR	WALK	
	0330:	0026	DO	12			BNE	TSTC	BRANCH IF MEMORY CELL IS DEFECT
	0350:	002A	91	E6			STAIY	CUR	TEOT TRITERA TOR DODDEE ROOKEDOING
	0360:	0020	20	60	00		JSR	DECCHK	DECREMENT AND CHECK CUR
	0380:	0031	A9	00			LDAIM	\$00	DISPLAY "0000 XX" IF
	0390:	0033	85	FA			STA	POINT	MEMORY IS O.K.
	0400:	0035	40	FB 1D	10		JMP	MONITO	+01
	0420:				3.20		0.000		
	0430:	003A	A5	E6		TSTC	LDA	CUR	DISPLAY THE ADDRESS OF
	0440:	003C	A5	E7			LDA	CUR	+01
	0460:	0040	85	FB			STA	POINT	+01
	0470:	0042	40	10	10		JMP	MONITO	
	0490:								
	0500:					SUBROU	TINES		
	0520:	-							
	0530:	0045	20	54	00	WRZERO	JSR	CURBEG \$00	FILL THE MEMORY BETWEEN BEG & END WITH \$00
	0550:	0040	nu	00			20120		
	0560:	004A	A9	00		WRZ	LDAIM	\$00	
	0570:	004C	91 20	E6	00		STAIY	CUR	
	0590:	0051	80	F7	00		BCS	WRZ	
	0600:	0053	60				RTS		
	0620:	0054	A6	00		CURBEG	LDX	BEG	CUR = BEG
	0630:	0056	86	E6			STX	CUR	101
	0650:	0058 005A	86	E7			STX	CUR	+01
	0660:	005c	60	1178-15			RTS	19292026	205.32
	0670:	0050	F6	F.6		INCCHK	INC	CUR	CIIR = CIIR+D1
-	0690:	005F	DO	02		Incent	BNE	IA	
	0700:	0061	E6	E7			INC	CUR	+01
	0720:	0063	38			IA	SEC		C=O IF CUR >END
	0730:	0064	A5	02			LDA	END	
	0750:	0068	A5	03			LDA	END	+01
	0760:	006A	E5	E7			SBC	CUR	+01
	0780:	0060	60				RTS		
	0790:	0060	38			DECCHK	SEC		CUR = CUR - 01
	0800:	006E	A5	E6			LDA	CUR	
	0820:	0072	85	E6			STA	CUR	
	0830:	0074	A5	E7			LDA	CUR	+01
	0840:	0076	85	E7			SBCIM	CUR	+01
	0860:	007A	38	-			SEC		C=D IF CUR < BEG
	0870:	007B	A5	E6			LDA	CUR	
	0890:	007F	AS	E7			LDA	CUR	+01
	0900:	0081	E5	01			SBC	BEG	+01
	0920:	0085	00				RIS		
	0930:	0084	A9	01		WALK	LDAIM	\$01	INIT. PATTERN
	0940:	0086	85	E5			LDYTM	\$00	
	0960:	008A	81	E6			LDAIY	CUR	IS STILL \$00 IN THE CELL
	0970:	0080	DO	OF			BNE	WALKB	IF NOT, THEN BRANCH
	0990:	UUBE	AC	00			LUXIM	908	WARNING DIE COUNTER
	1000:	0090	A 5	E5		WALKA	LDA	PATTER	CURR. PATTERN INTO ACCU
	1020:	0094	DI	EÓ			CMPIY	CUR	DOES IT MATCH?
	1030:	0096	00	05			BNE	WALKB	IF NOT, THEN BRANCH
	1040:	0098	06	ES			ASL	PATTER	WALKING BITS!
	1060:	009B	DO	F3			BNE	WALKA	
	1070:	0005	60			UALYS	DTC		
	1090:	0070	00			BALKB			
	1100:								