DMA Changer - version 0v2

Background / General

After Christian had published his article "A new Atari STE bad DMA investigation" (http://www.chzsoft.de/site/hardware/new-atari-ste-bad-dma-investigation/), I decided at short notice to design a circuit that should be able to solve this problem as well with the help of a few components.

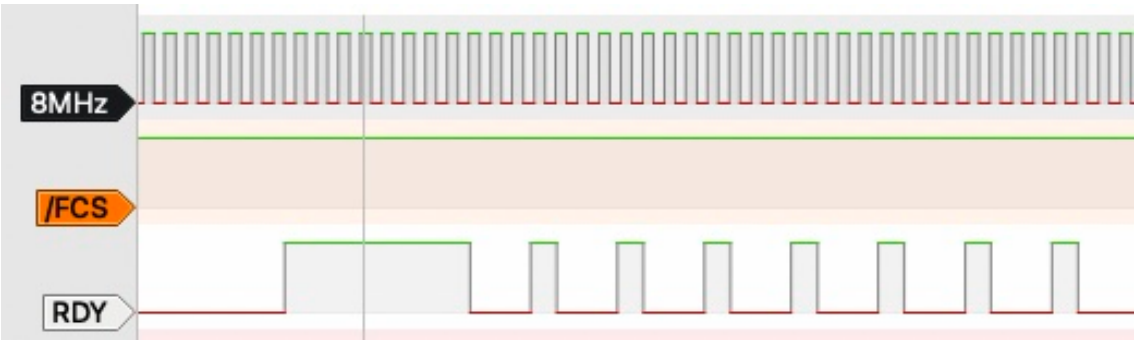
Unfortunately, the first version 0v1 did not have the desired success. In the meantime, I found the reason for this and fixed it. The present board with the version 0v2 works now in my two 1040 STE with altogether seven tested C025913-38 DMA ICs perfectly. Also, the use of a C100110-001 from IMP in the 1040 STE is possible. I have also tested this with the help of two such DMA ICs.

This document explains the technical facts and function of the previous and new circuit in detail.

The interaction of the two signals /FCS und RDY

The /FCS[1] and RDY signals are used for communication between the GSTMCU[2] and the DMA IC. The signal "Ready" (RDY) is designed as an open collector signal and can be pulled to LOW level by the GSTMCU as well as by the DMA IC. There are two accesses that can be distinguished.

In the first case, in the event of a DMA transfer, the DMA module automatically supplies data from ACSI interface[3] directly to the main memory. In this case the signal RDY is used as an "acknowledgement signal" for the correct transfer of the data into the memory. In this case the signal RDY is controlled by the GSTMCU. To differentiate between the two cases, here is the note that the signal /FCS always has a HIGH level in this operating mode.



Picture 1: DMA data transfer - /FCS remains at HIGH level

In the second case, the processor together with the GSTMCU controls access to the two DMA registers at addresses xFF8604 and xFF8606. In this case, the signal RDY is again used

[1] For a sequence, signals can have a HIGH or LOW level for the relevant sequence. If this signal is at LOW level for the considered process, the signal is preceded by a negation bar. In this case "/FCS". This means that the signal must have a LOW level, if it is considered to be logically active.

[2] In the normal ST it is the GLUE. In the STE it is the GSTMCU. The problem, described here, only occurs in interaction with the GSTMCU - not with the GLUE. The timing in the GSTMCU is clearly different from the GLUE at this point.

[3] “Data from ACSI interface" also means “data from floppy”. At this point no distinction is made as to where the data actually comes from.

as an acknowledgement signal. However, it is now used as a response from the DMA IC to the register requests from the processor and the GSTMCU. Again, the note, that in this operating mode the /FCS signal has a LOW level.

Picture 2: DMA register access – C025913-38 without modification of the signal RDY

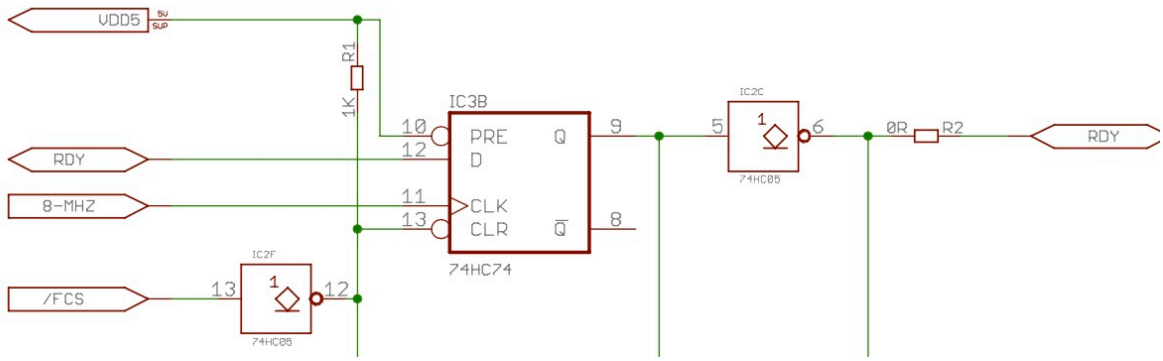
On the basis of the /FCS signal, these two cases can thus be distinguished.

Functionality of the old hardware version 0v1

In case of a DMA access, the RDY signal must not and should not be modified. In case of a register access the signal RDY must be shortened. Compare the article from Christian Zietz.[4] The DMA Changer version 0v1 (also version 0v2) shortens the signal RDY.

Picture 3: DMA register access – C025913-38 with modification of the signal RDY

The following circuit cutout provides for the shortening of the signal RDY:



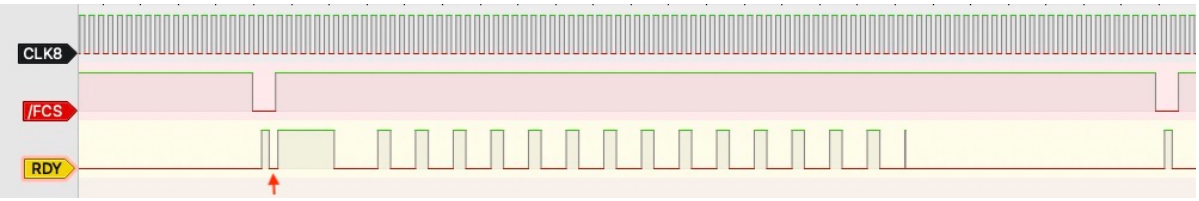
Picture 4: Circuit part for shortening the signal RDY for the C025913-38

To explain the circuit above: With the used flip-flop of type 74HC74 the output Q is set, if in case of a positive edge of the 8 MHz clock the signal RDY (D-input of the flip-flop) is HIGH. Via an inverter with an open collector output, this signal is given back to the input. The result

[4] http://www.chzsoft.de/site/hardware/new-atari-ste-bad-dma-investigation/

is, that the signal RDY is reset to LOW level. By means of the signal /FCS at the CLEAR input of the flip-flop, a shortening of the signal RDY is only performed during the time if the signal /FCS have a LOW level. This ensures during a DMA data transfer that the RDY signal isn't modified – as required.

When accessing the two DMA registers at address xFF8604 and xFF8606, there is a special case in which data is transferred immediately – when accessing the register. With the above described selection of the both cases (register access and DMA access) this leads to an interruption of the RDY signal (small red arrow in the following picture). This is undesirable.

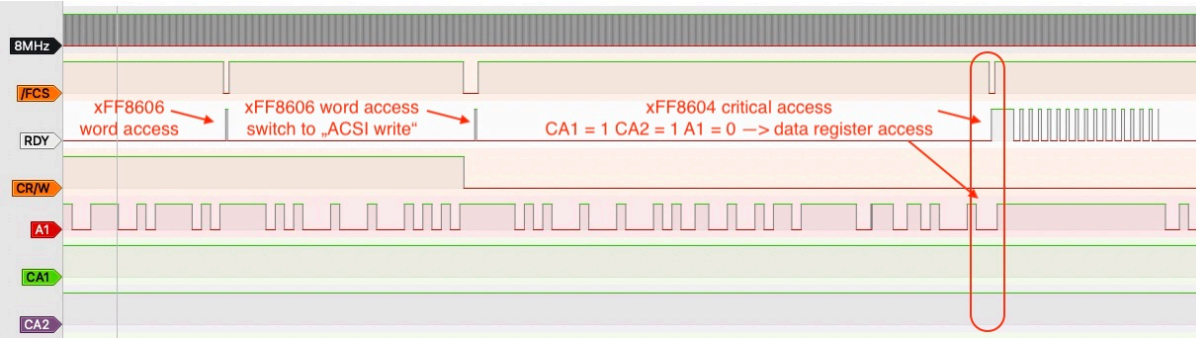


Picture 5: DMA register access with immediate data transmission and modification of the signal RDY

The interruption of the signal RDY leads to a "swallowing" of data. Only 15 packets are transmitted instead of the usual 16 packets.[5]

Functionality of the hardware version 0v2

Measurements showed that the special case listed above happens exactly when the data registers in the DMA IC are accessed. To understand this in detail, the following picture[6] will be helpful:



Picture 6: Access to the register FF8606 and FF8604:
Left: Change the status oft he DMAOUT bit
Mid: Load the Sector Count Register
Right: Write the first Command Byte to the DMA Port

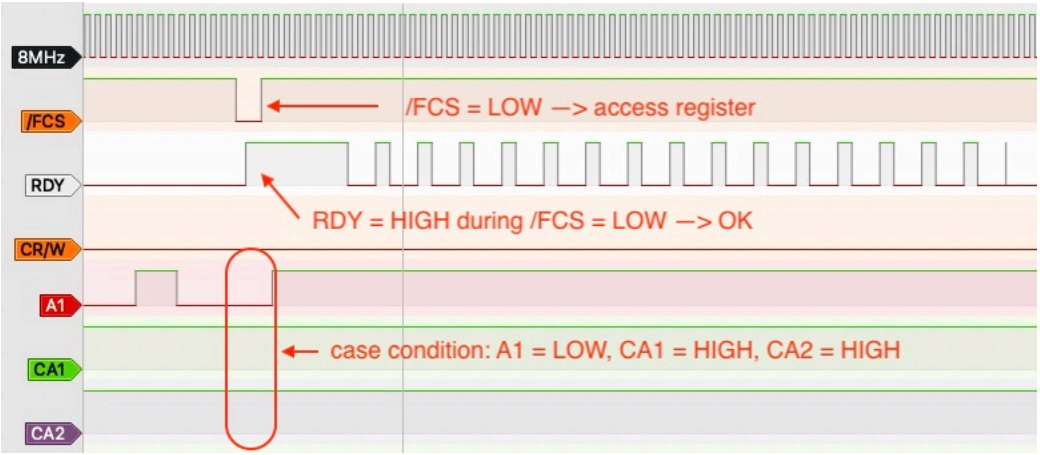
The combination CA1 = 1 and CA2 = 1 indicates that data is read from one of the DMA registers. Exactly, which register this is, cannot be determined from the outside because the state depends on bit 4 in register xFF8606. The above case shows an access to the sector count register. For further explanation, this case is to serve.

The usual practice is, that the sector count register very often accessed in write mode – rarely in read mode. However, both cases, writing and reading must be treated in the same

[5] If you recount the LOW phases in picture 5, you get 15 phases instead of the usual 16 phases. This fact was pointed out to me by Holger @pakman. Many thanks for this hint.
[6] At this point a thank you to Christian @czietz for the explanation of the sequence in picture 6.

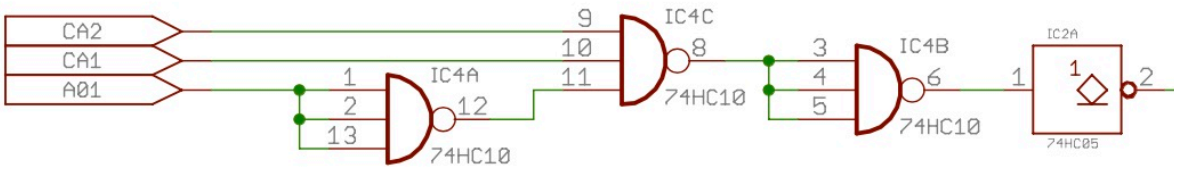
way. The write and read access, described above, occurs when the address line A1 = LOW[7] and the both signals CA1 and CA2 have HIGH level[8].

The access shown in Fig. 6 (the following image) once again in detail:



Picture 7: Detail for data transmission from the sector count register

The following sub circuit evaluates exactly the signals A1, CA1 and CA2. This signal decoding is added to the CLEAR input of the flip-flop.



Picture 8: Circuit cutout for decoding A1 = LOW and CA1 = CA2 = HIGH

With the help of this additional decoding the use of a C025913-38 in a 1040 STE now works without problems. The C025913-38 behaves with the two explained circuit parts so far like a C398739-001A. Further circuit parts are not necessary.

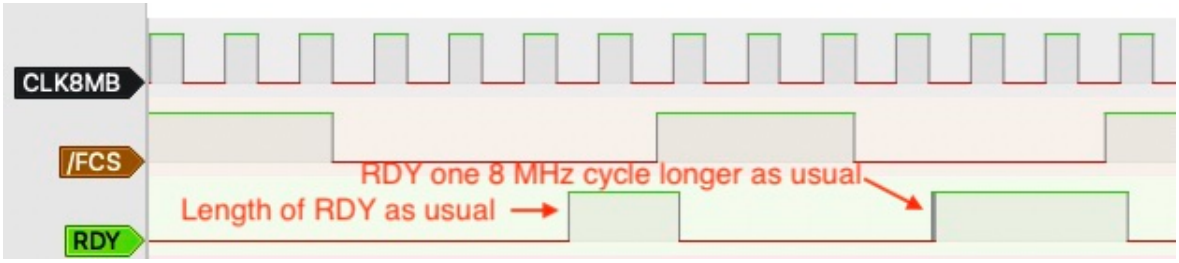
Another circuit part of the version 0v2

Besides the two DMA ICs C025913-38 and C398739-001A there is a third DMA IC which can be used in the ST but has not been used in the 1040 STE. The reason is, that the behavior on processor side is different from the two previously mentioned devices. It is the IMP IC C100110-001 and its timing doesn't harmonize with the output signals by the GSTMCU.

During the previous investigations I also looked at the timing behavior of the IMP DMA IC and saw an addition to the overall logic that also makes the operation of the IMP DMA IC in the 1040 STE possible.

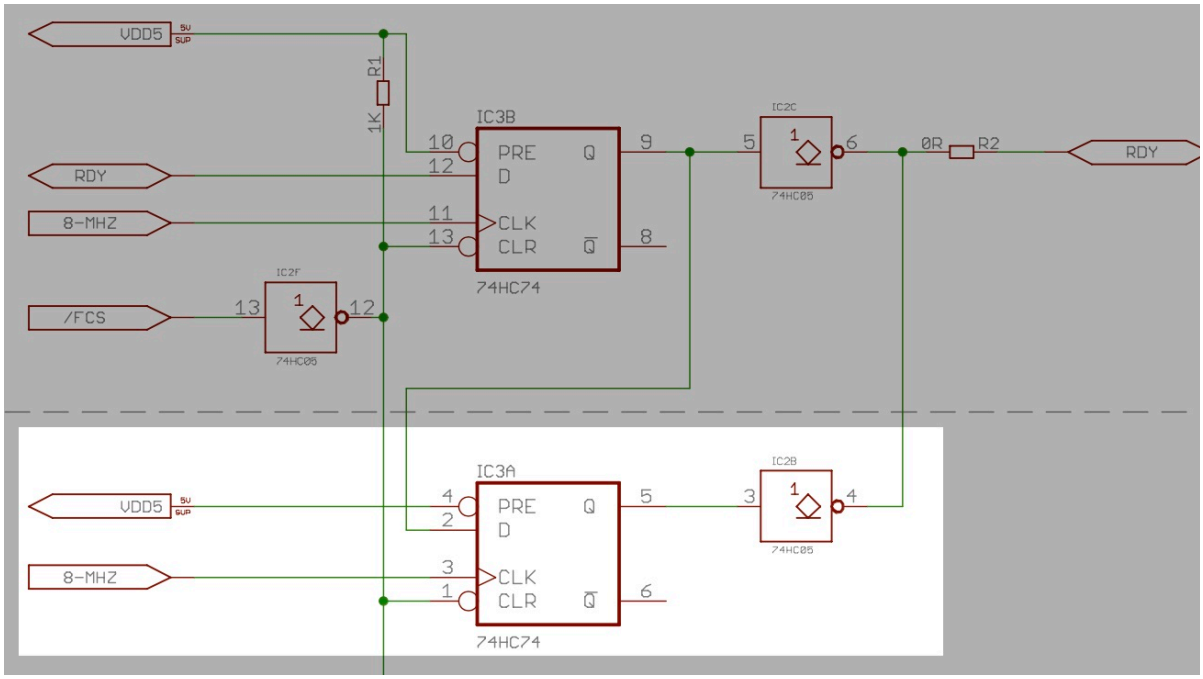
[7] A1 = LOW means an access to register xFF8604.
[8] CA1 = 1 and CA2 = 1 means data transfer in or out of the DMA register.

The following temporal behavior is different from the previously mentioned two devices:



Picture 9: Signal RDY can be one 8 MHz cycle longer for the IMP C100110-001 than for the other two DMA ICs.

There is also a solution for this case. With the help of the following addition (highlighted), part, this is possible:



Picture 10: Extension of the circuit for the DMA device C100110-001

To explain the above circuit part: The first flip-flop (IC3B) is set in case of a register access. Output Q (pin 9) is set to HIGH level. The output Q of IC3B is connected to the data input of IC3A. Thus, for the next 8 MHz clock the signal RDY is also shortened (output pin 5) and the downstream open collector inverter the signal RDY in case if the signal /FCS is still on LOW level.

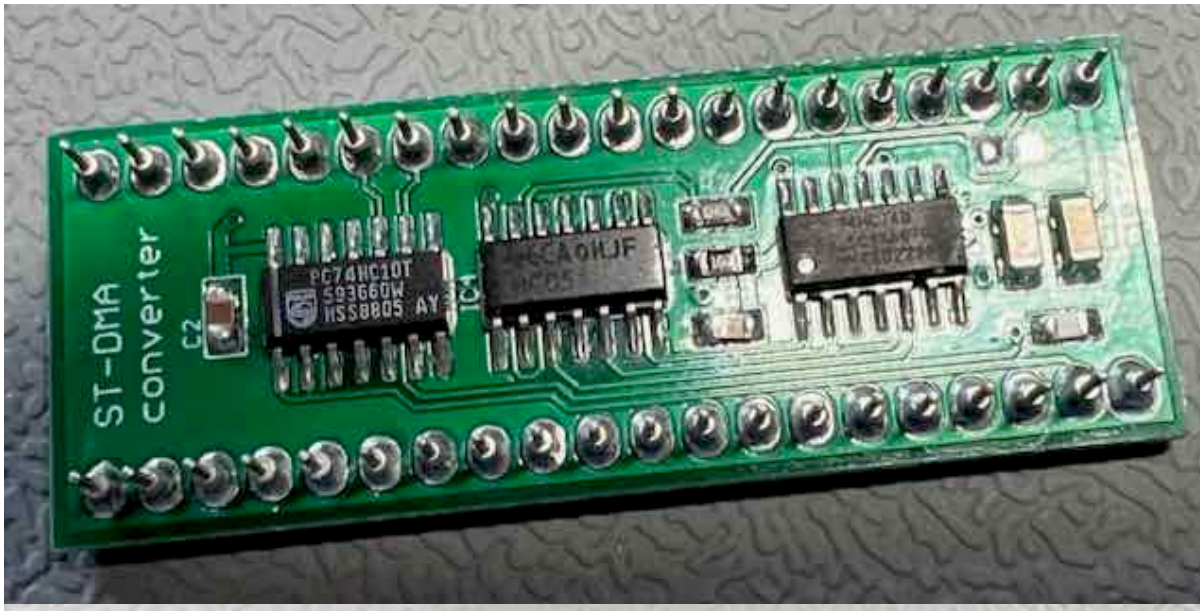
This circuit part is not active when a C025913-38 is used.

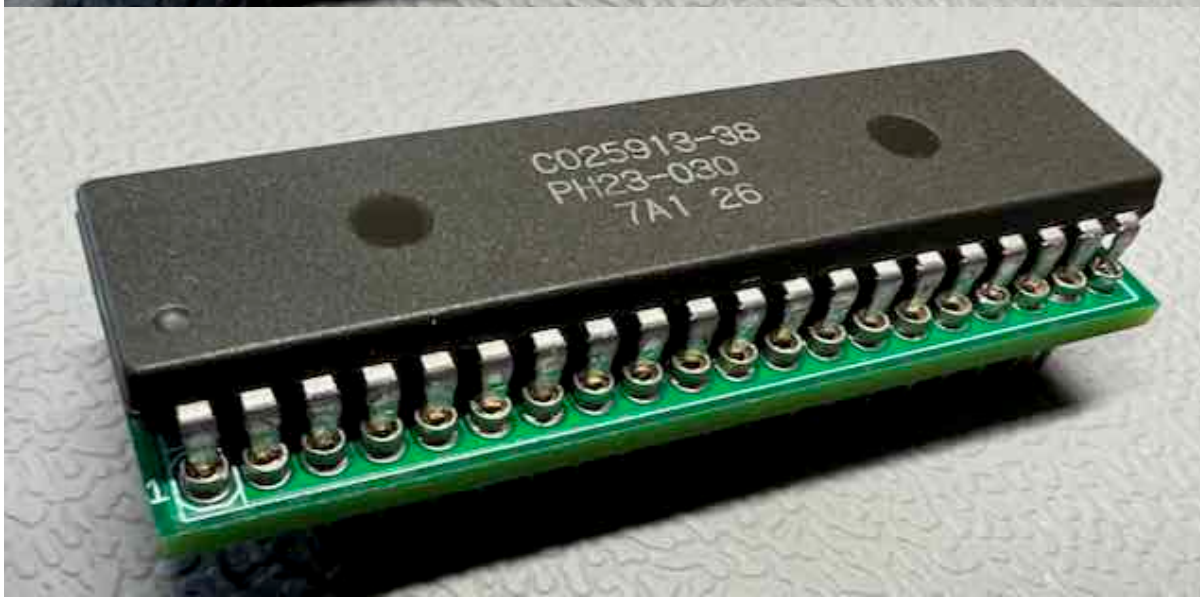
Mounting PCB

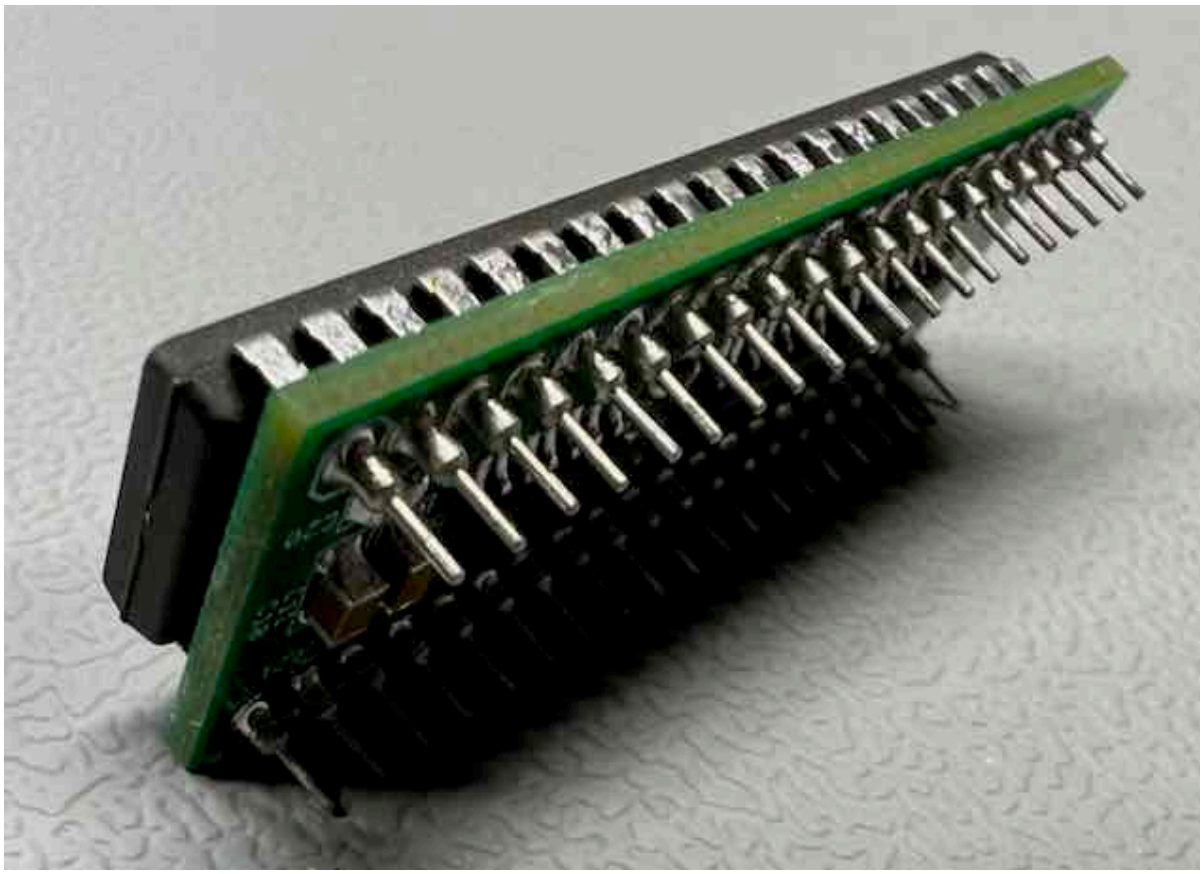
Mounting the PCB is very simple. It is recommended to equip the three ICs on the bottom side of the PCB first. Then the capacitors and the two resistors. Finally, the individual solder cups are inserted into the PCB from above and soldered to the underside. See the following pictures.

DMA Changer - version 0v2

Additionally, it should be mentioned, that I have equipped the ICs in my circuit with HC types. It should also be possible to use F-types. However, I have not carried out any further tests here.







Pictures of the PCB

Conclusion

The circuit of the DMA Changer version 0v2 works in my two 1040 STE without problems. It doesn't matter if you use a C025913-38 or a C100110-001. I tested this with my available DMA ICs[9]. Whether this leads in all computers to the desired success, must still be shown. Otherwise I have tested everything as far as it was possible for me to do so.

The complete schematic, assembly list and the necessary Gerber data can be found in the two attached ZIP files. If you like, you can order the circuit from a PCB manufacturer and assemble it yourself.

I hope for the best.

Best regards
Robert

[9] These are 7 ICs of type C025913-38 and 2 ICs of type C100110-001.