

The Web Site to Remember National Semiconductor's Series 32000 Family

Home	Chips	Systems (1)	Systems (2)	M32632	Software
Miscellaneous	Imprint				

TITAN6

In April 2020 due to the Corona crisis the distributor Arrows gave away Quartus evaluation licenses for people working in home office. This license enables work with the full version of Quartus for two months. During that time I was in short work and decided to spent some effort for my second Stratix II board.

In 2014 I thought it would become TITAN5. But TITAN5 is now part of TRIPUTER. So it became TITAN6.

TITAN6 has two distinctive features. First it runs the fastest version of M32632 at 75 MHz and second it has two main memories. All my other FPGA systems have only one memory. (The Cyclone V GX Starter Kit has a tiny SRAM - it is too small to be useful.)

Hardware

Figure 1 shows the final system with two small add-on cards on the right side. The upper one has a real-time-clock, an SD card socket and two option pins. The lower one has a DVI interface chip, two PS2 ports and a connector for a Dallas DS18S20 temperature sensor. The temperature sensor is mounted on top of the heatsink for the FPGA.

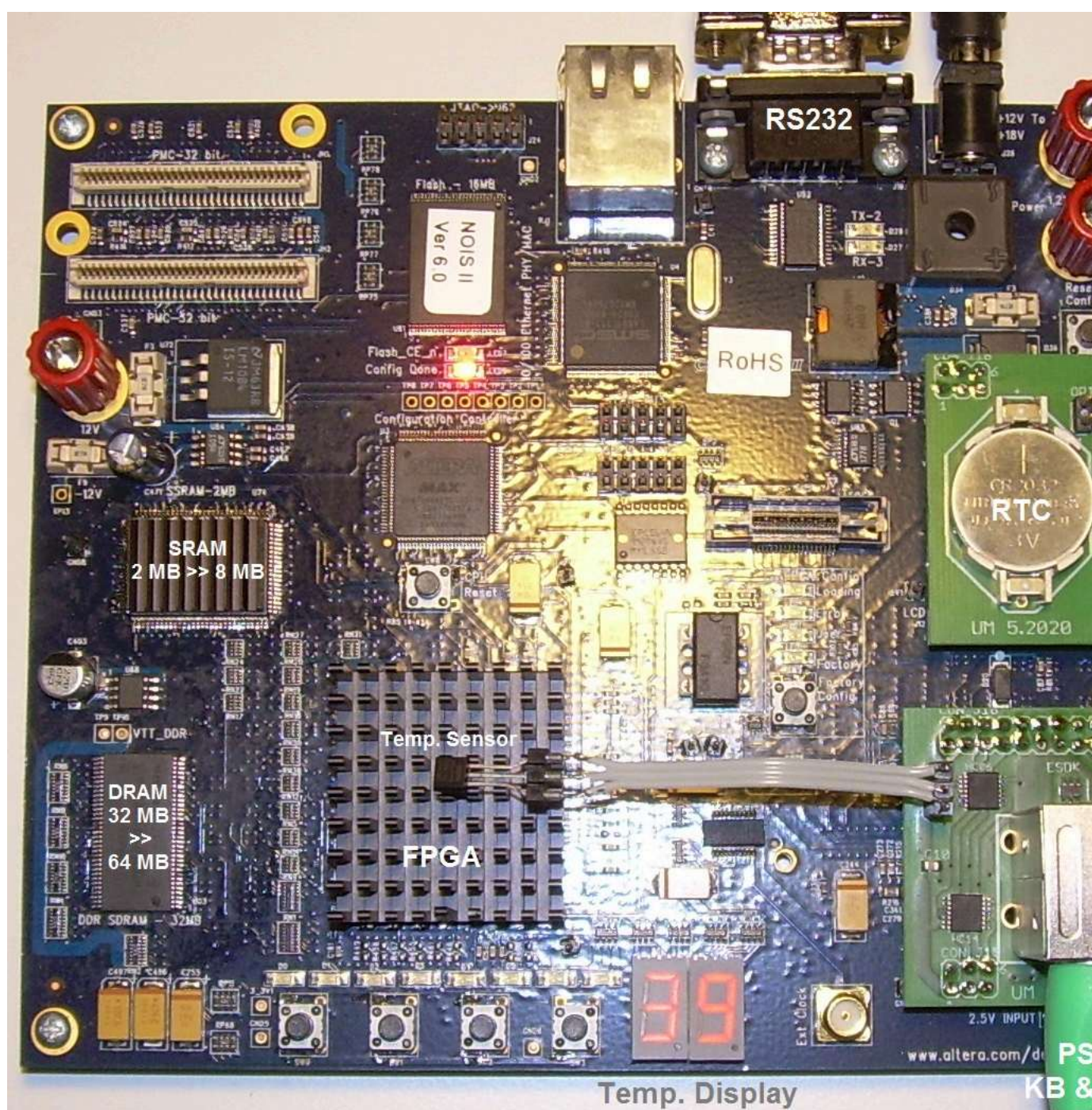


Fig. 1. TITAN6 uses two small PCBs for more capabilities. This enables TITAN6 to work as a standalone computer.

A nice feature of this board are two RAMs which are connected separately to the FPGA. One is a fast synchronous SRAM of 2 MB, 32 bit data bus, and the other is a DDR-SDRAM of 32 MB, 16 bit data bus. Even better the designer of the board routed additional address wires between the SRAM and the FPGA. A change of the DRAM from 32 MB to 64 MB requires no extra wires because the added column address bit is multiplexed with an existing row address bit.

I had always planned to upgrade the board. I bought already in 2015 some SRAMs of 4 MB. 64 MB DRAMs were available from earlier projects. Now it was time to do the upgrade.

I'm not perfect in soldering. TQFP packages with 0.65 mm pitch are a nightmare. Therefore I let all parts, which don't have 2.54 mm pitch, solder at a small service company.

I tested the upgraded parts with the M32632 running at 55 MHz. The frequency factor between memory and CPU was 3 and this tested the RAMs at 165 MHz. The test was pass. During this time I was hoping for a CPU frequency of 80 MHz with a frequency factor of 2.

Later the day I was looking in the schematic of the board. Nothing special in mind I looked at the SRAM and noticed an address wire called A22. My design was not using it. Could it be that an even bigger SRAM can be used? Yes, it can. I simply have overlooked it for years. But if it is your favorite hobby then you do crazy things and I ordered the next day two 8 MB SRAMs at Mouser. They are very expensive and they are available since 2005...

Now I was really happy about the board. I never used a fast SRAM as main memory and these SRAMs are really fast, much faster than DRAMs. But it is a different story if you look from a system perspective on this topic. A 16 bytes cache refill needs 5 CPU clocks from the SRAM and 8 CPU clocks from the DRAM. What does that mean for the system performance? The following table shows some benchmark results:

Program	SRAM	DRAM	Difference	DRAM + Graphics	Difference	NS32532 @25MHz
Pascal Compiler	1.087 sec.	1.151 sec.	+ 5.9 %	no change		5.428 sec.
Assembler	1.966 sec.	2.031 sec.	+ 3.3 %	no change		8.882 sec.
JPEG Decoder	11.297 sec.	11.506 sec.	+ 1.85 %	11.576 sec.	+ 2.5 %	94.090 sec.

Bigger programs like the SRAM more. But the speed increase is very limited. I have expected something like this from former experiences. If you have enough cache then the speed of main memory is not important (bad news for DDR5 in PCs).

It is a surprise that if the graphics is active there is no noticeable speed decrease. Anyway 3 % of the available bandwidth of the DRAM is used for graphics. Obviously the programs do not access the memory very often.

Two main memories showed another effect: ever since the beginning of the CPU development in 2009 there was a common data bus for the instruction cache and the data cache. For TITAN6 I had to modify the toplevel schematic to use two data busses. The instruction cache and the data cache can now be loaded in parallel - another factor for higher performance (not yet tested).

Very disappointing was the fight for CPU speed. It was not possible in two months to bring the CPU up to 80 MHz. Even at 75 MHz most of the time the compilation process failed. The last one I made before the license expired resulted in 75.19 MHz - minimum target achieved.

The graphics features are limited. TITAN6 only provides 640 by 480 pixels at 8 bit colors (VGA) and a text mode with 80 characters by 40 lines.

The temperature of the FPGA heatsink is around 42 degree celisus. I put a small heatsink on the SRAM too. If you want a long lifetime of semiconductor devices keep them cool.

In 2019 I built an "Apfelmännchen" machine (or Mandelbrot engine) for fun. I wanted to see 1 Gflop/s in action. I run it on TRIPUTER. Later I took the modul out because it used a lot of ressources and increased the compile time. TITAN6 offered the opportunity to include this modul permanently. The modul contains two pipelines. Each of it does seven single precision floating-point operations in parallel at 75 MHz. The combined performance is 1.05 Gflop/s. This is enough for a walk nearly in real-time through the Mandelbrot landscape in VGA resolution.

At the end only half of the FPGA resources were required to implement TITAN6. The other half can be used in the future for more features - if I get a license...



Fig. 2. The head of the system before it disappeared under the heatsink: the Stratix II FPGA. It is nice that the manufacturer went for the fastest speed grade.

Software

The operating system of TITAN6 uses one memory pool containing both the SRAM and the DRAM. They are used differently as long as the pool is well sorted. The SRAM is used for program and data. The DRAM is used for a RAM disk and for the graphics. If a large program uses more than 8 MB it will get access to the DRAM. After the program has terminated, the memory pool may no longer be well sorted.

The central work space of TITAN6 is a RAM disk. Its size is variable up to 60 MB. This is more than enough memory for people who started to work with computers in the 1980's.

The SD card has a simple filesystem. No bit map for free blocks is used because of the limited write capability of flash devices. Each file on the SD card has one descriptor block for name, size and date followed by the data blocks. Files are stored in multiples of 32 kB. This is most of the time enough space if a file gets bigger. To read a file the OS starts searching it always at block 0. The card is fast enough to make hundreds of descriptor accesses in less than a second. Files on the SD card must be copied to the RAM disk before they can be used.

A task for TITAN6

TITAN6 is the fastest implementation of the NS32K architecture. As so it must get a task which makes use of its power. Such a task exists and it is emulation of a PC. Of course not a modern one but something based on 80386 or 80486 processors running in real mode.

The background is that during the time when this processors were popular nice games were written in software only, i.e. Doom. Doom is not nice and not my favorite but some flight simulators have impressive graphics. There are no such games for NS32K (and nobody will write them). The only way to play such games is to emulate the PC. This is what I will try to do with TITAN6.

Next chapter: [Project C7](#)