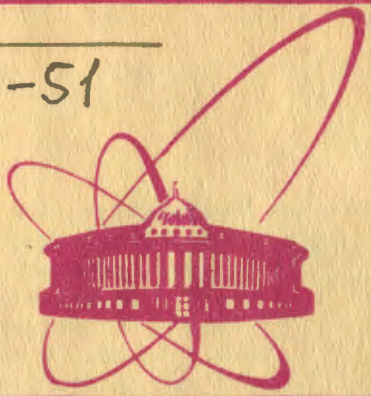

объединенный
институт
ядерных
исследований
дубна

E11-81-297

H.Leich, F.V.Levchanovsky

THE STRUCTURE OF A MULTI-MICROPROCESSOR SYSTEM FOR AN INTELLIGENT GRAPHIC TERMINAL

Submitted to 7th Symposium
on Microprocessors and Microprogramming,
"Euromicro-81", Paris, France, 8-10 Sept., 1981

1981

The trend towards performing more complicated graphic picture processing and display file transformation directly on the terminal level, with minimal host computer interference dominates more and more the design of graphic terminals. This increases the terminal's local abilities and decreases the load of the host computer. At the same time enlarging the functions of the terminal leads to the inevitable rise of cost and, naturally limits the terminal's application fields.

Resently designed intelligent graphic terminals based on either special processors, or minicomputers execute the following functions:

- data exchange between terminal and host computer (realization of a communication protocol and interpretation of high level input language);
- floating-point data handling;
- display file management;
- picture generation on a CRT display screen;
- image transformation (displacement, scaling, rotation, etc.);
- user-terminal dialogue support;
- peripheral driving;

Microprocessors (uP) are used in graphic-terminal design since 1975. However wide-spread universal uP's such as INTEL 8080, MOTOROLA 6800 and others have comparativly low throughput and their application in the graphic terminal field is practically restricted to storage tube display where image creation speed is not so prevalent.

The above mentioned uP's have insufficient throughput for stroke refresh displays providing high image quality and picture transformation in time and space.

The structure of the multi-microprocessor system (MMPS) for an intelligent graphic terminal is intended for real-time picture processing both in autonomous mode and on-line mode connected to a host computer. System tasks are partitioned into subtasks based on functional principles, which are statically allocated to the following modules of the MMPS: processor-monitor, display module, arithmetic module and common random-access memory module. In module building blocks universal 8-bit uP's, arithmetic uP's, processing units and bitslice uP's have been used.

The processor-monitor module is designed to perform: the user-terminal dialogue, the display file creation, the system turn-on initialization, the handling of interrups from interface controllers and system modules and the system faults diagnostic.

The main task of the display module is reading display file instructions from common RAM, constructing graphic images on CRT screen of stroke-refresh display and handling signals from light-pen or trackball. The monitor processor starts the display module by transmitting the first address of a display file block to be executed.

The display module handles two types of instructions-graphic (to draw point, vector, symbol) and control (jump, call, etc.) instructions. Display primitives like point, vector, symbol are generated by special hardware built on bit-slice uP's. Vectors are drawn based on the constant-rate method. Symbols are generated by means of piece-linear approximation.

The arithmetic module executes floating-point operations on display file data, calculates simple mathematical functions and performs image - and data format fransformations.

The 48K byte memory module provides the common system resource.

All systems modules are connected to a common bus structure working in semisynchronous mode. In this mode bus time is divided into fixed, equal-width time slots (50-100 ns) where various number of time slots may be assigned to a given intermodule-communication depending on the cycle times of the involved modules. This method seems to us the most reasonable to connect a small number of modules with different speed characteristics via a common bus, as the timing characteristics of an asynchronous bus are almost maintained, with the advantage of a less complex hardware realization.

Access conflicts on the common bus are handled by a bus-arbiter, which accepts requests and resolves contention in parallel for at most eight input lines.

Communication between the various modules of the MMPS is established by means of "mailbox-memories". This "mailboxes" do not reside usually in the common memory module but are implemented in each module, with the exception of the monitor, as blocks of special dual-port control memory. Access to these blocks is available from the common bus as well as from the modules internal bus. This allows to decrease by a half the number of accesses to common bus in case of message exchanges between modules.

Private memory of up to 15K byte is used in module implementation to significantly reduce the number of accesses to common resources (memory, common bus).

To realize the above-mentioned principles of the 2^{16} words-wide address space of the system is partitioned in the following manner:

Ø – 15K	– private memory of each module;
15K – 16K	– adress space for dual-port control memory blocks;
16K – 64K	– common memory address space.

All modules of the system containing processors, except the monitor, may act as master as well as slave in the process of module communication. As the monitor module has to perform system monitoring it only may function as master. Therefore the monitor may gain exclusive control of the bus via a hardware implemented bus-lock function.

The structure of the proposed MMPS is not restricted to particular intelligent graphic terminal design but may also be used for other applications.

Received by Publishing Department
on April 30 1981.