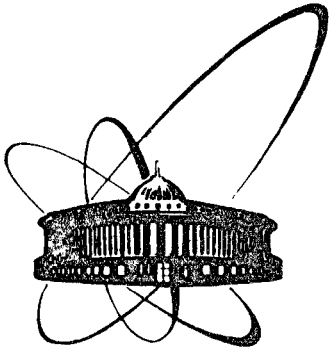


объединенный
институт
ядерных
исследований
дубна

E11-82-791

H.Leich

THE ARCHITECTURE OF AN ARITHMETICS MODULE FOR THE INTELLIGENT GRAPHICS TERMINAL

Submitted to the Conference on Computer Graphics´83
(ČSSR, June, 1983)

1982

1. INTRODUCTION

At present the design of graphics terminals is influenced by the trend towards performing complicated graphics functions as well as picture transformations directly on the terminal level, with minimal host computer influence. The appearance and the following mass production of low cost microprocessors affects more and more the design of graphics terminals. However widespread 8-bit microprocessors such as INTEL8080, MOTOROLA6800 and others have insufficient throughput so that their application in graphics terminals is restricted to storage tube types where image creation speed is not prevalent.

To provide high drawing speed, which is needed for high resolution stroke refresh displays, as well as a set of functions for object manipulation on the terminal level a multi-microprocessor system (MMPS) has been designed and a prototype of this system has been implemented. System tasks are partitioned into subtasks and statically allocated to the following modules of the MMPS: processor-monitor, display module, arithmetics module and common memory module [LEI81].

2 FUNCTIONS

The arithmetics module is the common resource in the MMPS, executing floating-point operations on display file data and calculating standard mathematical functions. It provides the computing power for performing graphics transformations within the terminal itself.

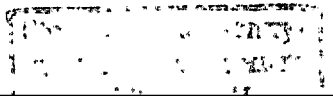
The following transformations may be executed on graphical objects:

- Displacement
- Scaling
- Rotation.

A window/viewport transformation, including clipping, may be performed, which transforms a selected part of the user coordinate system (window) into a predefined area on the display screen (viewport).

The functions mentioned above are implemented by special software which runs on the modules internal microprocessor [ALEI].

To execute a transformation, the arithmetics module must receive a function identifier (command code) and the starting



address of the display file area to be transformed from another module of the MMPS, which acts as a master with respect to the common resource. The arithmetics module autonomously loads the display file data, executes the desired transformations and stores the results in the common memory.

As the built-in functions of the arithmetics module may be called from any other module of the MMPS, special Test-and-Set (TAS) flags are implemented by hardware to resolve access conflicts. In an analogous manner, BUSY-flags support resource allocation.

3. IMPLEMENTATION

Figure 1 shows the overall structure of the arithmetics module. It consists of a CPU based on the microprocessor type 18080, a PROM of 8K byte and a 2K byte RAM, an interrupt system, an interface to the common bus of the MMPS and two special arithmetic processing units (APU). The two Am9511-APUs provide fixed and floating point arithmetic (ADD, SUB, MUL, DIV) as well as floating point trogonometric (SIN, COS, TAN,...) and mathematical operations like square root, logarithm a.o. on three data types [Am9511]:

- 16-bit-fixed
- 62-bit-fixed
- 32-bit-floating point.

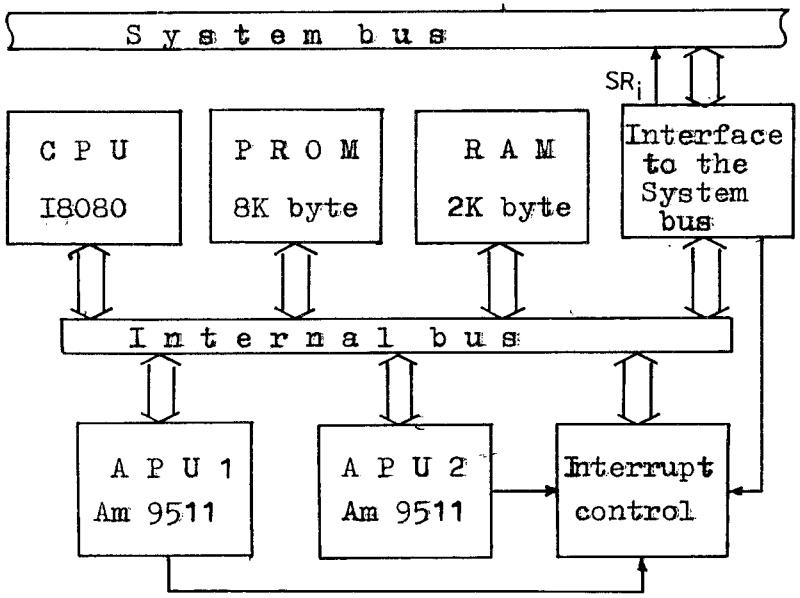


Fig. 1. The structure of the arithmetics module.

The CPU calculates the complex graphics transformations, including load and store of the operands, using the computing capabilities of the two APUs. For performing a transformation one APU computes the new x-coordinates of a graphic object while in parallel the other APU handles the y-coordinates. Transfers to and from the APUs are performed by the CPU using the memory map technique. Upon completion of a command, the APU issues an end of execution signal which is connected to the modules' interrupt system. Thus the CPU can prepare a task queue for each APU and upon receiving the appropriate interrupt the next command may be issued.

The interrupt system enhances the interrupt capabilities of the 18080-microprocessor providing eight channels for interrupt requests. It is built around a I8214 interrupt-controller.

The arithmetics module is connected to the system bus by means of special logic. This logic consists of interface drivers for address, data and transfer-control lines, of TAS- and BUSY-flags for synchronization of task execution and of a control memory block with a dual port interface.

As the module is a common resource within the MMPS, a mechanism for resource allocation should be provided. A hardware implemented TAS-flag is used to prevent access conflicts. If any module of the MMPS wants to use the built-in functions of the arithmetics module it should make sure that the module is not yet occupied. Doing this it must read the TAS-flag. If the flag is "ONE", the resource is yet allocated. If the answer is "ZERO", the resource is free, and at the end of the read flag operation the flag will be forced to "ONE" by hardware (see Fig. 2). The new master may start a task in the slave writing a message to the control memory block of the slave. When starting a task, a BUSY-flag will be set to "ONE" by hardware. Upon completion of the task the slave module clears the BUSY-flag. Reading the status information (BUSY) the master determines the end of operation and may immediately start further tasks (see Fig. 3). To release the resource the actual master must clear the TAS-flag of the slave (Fig. 4).

The communication between the arithmetics module and other modules of the MMPS, that may act as a master, is established by means of a control memory block which resides in the bus interface unit of the module. Access to this control memory is available from the system bus as well as from the modules internal bus. To prepare and to start a task it is necessary to put parameter information and a task identifier (a command) in the

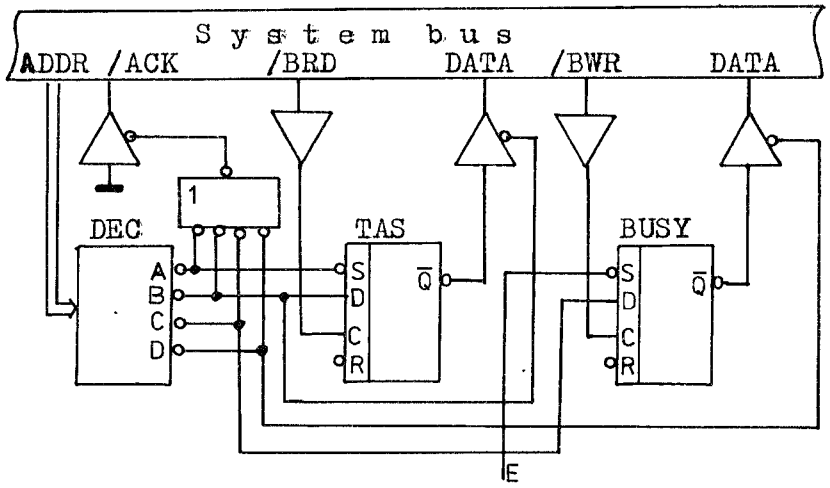


Fig. 2. TAS- and BUSY-flags. A: Clear TAS-flag (release the resource); B: Read TAS-flag; C: Start a task (sets the BUSY-flag); D: Read status of task-execution; E:Clear BUSY-flag (after completion of a task).

control memory. When the control memory is accessed an interrupt is generated by hardware and passed to the IR-controller thus informing the CPU about a new task.

For autonomously loading and storing data from and to the commom memory of the MMPS the arithmetics module is equipped with special logic. Everytime the CPU generates an address which lies within the address range of the system bus this logic sends out a request to the bus arbiter and turns off the ready-line of the CPU. Receiving "bus granted" the logic enables the bus drivers and acquires control of the system bus.When the addressed module sends his acknowledge-signal the ready-line to the CPU is turned on.

4. CONCLUSIONS

The module described provides a set of special graphics functions within a multi-microprocessor system designed for an intelligent graphics terminal. The functions are implemented by special software which runs on a universal microprocessor. To enhance the computational power of the microprocessor two special arithmetic processing units are incorporated in the system. Resource allocation as well as task synchronization are supported by flags implemented in hardware.

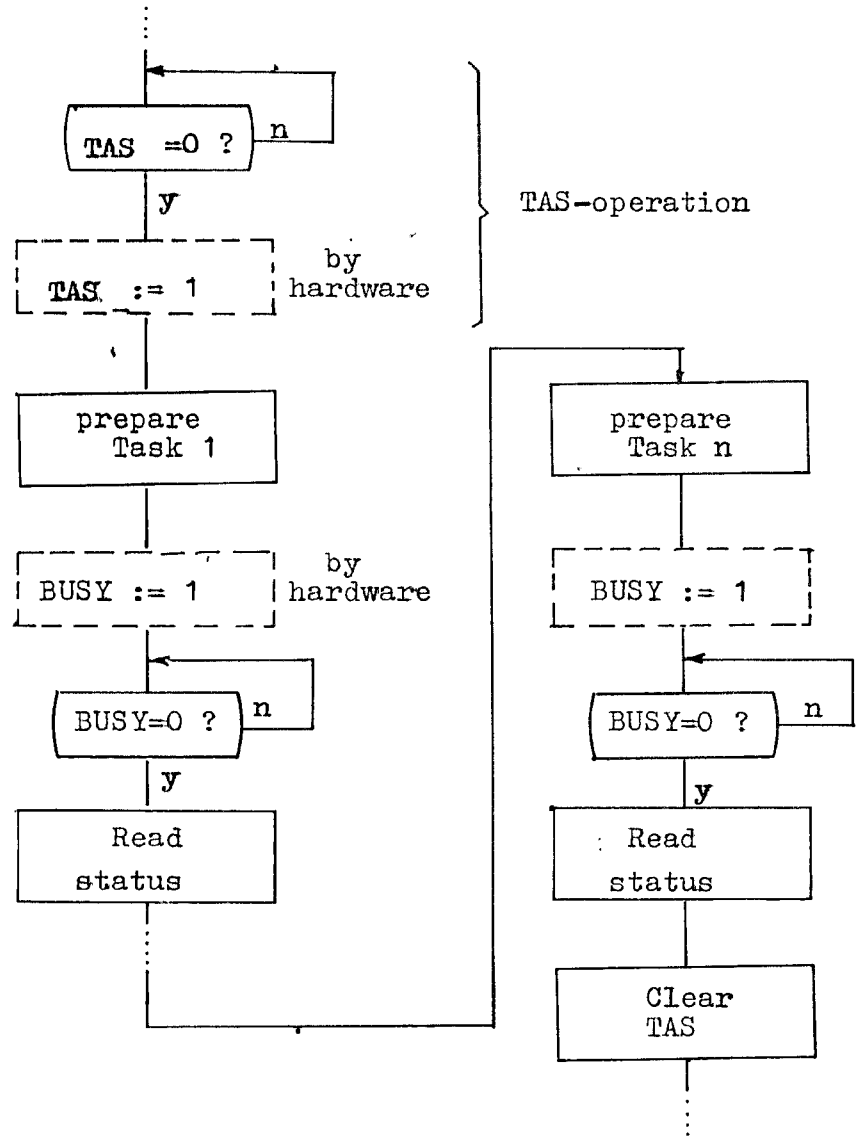
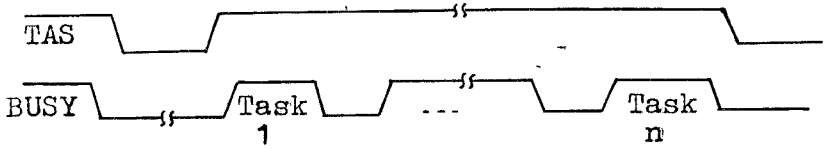


Fig. 3. Synchronization of task-execution.

The application of the arithmetics module is not restricted to particular intelligent graphics terminals but may be extended to other fields changing the modules' internal software.



TAS = 1: Resource allocated to any master
BUSY = 1: The module works on completion of a task

Fig. 4: Example of resource allocation.

REFERENCES:

[LEI81] H.Leich, F.V.Levchanovsky. The Structure of a Multi-Microprocessor System for an Intelligent Graphic Terminal. JINR, E1-81-297, Dubna, 1981.

[Am9511] Am9511 Arithmetic Processing Unit. Advanced Micro Devices, Inc., 1978.

[ALEI] A.Leich et al. The Implementation of Two-Dimensional Transformations on the Arithmetics Module of an Intelligent Graphics Terminal. To appear in: Almanac of Computer Graphics Conference, Bratislava, 1983.

Received by Publishing Department
on November 19, 1982.

SUBJECT CATEGORIES OF THE JINR PUBLICATIONS

Index	Subject
1.	High energy experimental physics
2.	High energy theoretical physics
3.	Low energy experimental physics
4.	Low energy theoretical physics
5.	Mathematics
6.	Nuclear spectroscopy and radiochemistry
7.	Heavy ion physics
8.	Cryogenics
9.	Accelerators
10.	Automatization of data processing
11.	Computing mathematics and technique
12.	Chemistry
13.	Experimental techniques and methods
14.	Solid state physics. Liquids
15.	Experimental physics of nuclear reactions at low energies
16.	Health physics. Shieldings
17.	Theory of condenced matter
18.	Applied researches
19.	Biophysics

Лайх Х. E11-82-791
Архитектура арифметического модуля
интеллектуального графического терминала

Интеллектуальный графический терминал, разработанный в ОИЯИ, содержит арифметический модуль /АМ/, предназначенный для автономного выполнения графических преобразований. Модуль состоит из универсального микропроцессора, который может анализировать и обрабатывать сложные структуры данных, и из двух специализированных быстродействующих арифметических процессоров для вычисления стандартных математических функций с разными форматами данных. Арифметический модуль является общим ресурсом внутри многопроцессорной архитектуры. Для решения конфликтных ситуаций при доступе к АМ и для распределения его ресурсасхематически реализованы флаги.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна 1982

Leich H. E11-82-791
The Architecture of an Arithmetics Module
for the Intelligent Graphics Terminal

The Intelligent Graphics Terminal designed at the JINR at Dubna incorporates an arithmetic module for autonomously performing graphics transformations. The module consists of an universal microprocessor, which is capable of analyzing and handling complex data structures, and two special high-speed arithmetic processors for calculations of standard mathematical functions on different data types.The arithmetics module is a common resource within the multiprocessor architecture. To resolve access conflicts and for resource allocation hardware primitives have been provided.

The investigation has been performed at the Laboratory of Cumputing Techniques and Automation, JINR.

Preprint of the Joint Institute for Nuclear Research. Dubna 1982