MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
NATIONAL TECHNICAL UNIVERSITY OF UKRAINE
“IGOR SIKORSKY KYIV POLYTECHNIC INSTITUTE”
DEPARTMENT OF BIOMEDICAL ENGINEERING

Shlykov V. V., Danilova V. A.

MEDICAL MICROPROCESSOR SYSTEMS

Workshop on discipline for students of specialties 163 "Biomedical Engineering"
and 152 "Metrology and information-measuring devices"

Kyiv
Igor Sikorsky Kyiv Polytechnic Institute
2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМ. ІГОРЯ СІКОРСЬКОГО»
КАФЕДРА БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

Шликов В.В., Данілова В.А.

МЕДИЧНІ МІКРОПРОЦЕСОРНІ СИСТЕМИ

Практикум з дисципліни для студентів спеціальностей 163 «Біомедична інженерія» та 152 «Метрологія та інформаційно-вимірювальна техніка»

Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для здобувачів ступеня/магістра
за освітньою програмою «Медична інженерія»
спеціальності 163 «Біомедична інженерія»

Ухвалено на засіданні
кафедри біомедичної інженерії ФБМІ
(протокол № 5 від « 24 » грудня 2019 р.)

Київ
«КПІ ім. Ігоря Сікорського»
2020

Рецензент	*Худецький І.Ю.*, д.м.н., проф., завідувач кафедри біобезпеки і здоров’я людини КПІ ім. Ігоря Сікорського, *Сидорець В.М.,* д.т.н., проф., провідний науковий співробітник відділу фізики газового розряду і техніки плазми № 056 Інституту електрозварювання ім.Є.О.Патона
Відповідальний редактор	*Зубчук В.І.*, к.т.н., доц., доцент кафедри біомедичної інженерії КПІ ім. Ігоря Сікорського

Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 8 від 09.04.2020 р.)
за поданням Вченої ради факультету біомедичної інженерії (протокол № 7 від 24.02.2020 р.)

Електронне мережне навчальне видання

Шликов Владислав Валентинович, д-р техн. наук, доц.
Данілова Валентина Анатоліївна, ст. викладач

МЕДИЧНІ МІКРОПРОЦЕСОРНІ СИСТЕМИ

Практикум з дисципліни для студентів спеціальностей 163 «Біомедична інженерія» та 152 «Метрологія та інформаційно-вимірювальна техніка»

Медичні мікропроцесорні системи: Практикум з дисципліни для студентів спеціальностей 163 «Біомедична інженерія» та 152 «Метрологія та інформаційно-вимірювальна техніка» [Електронний ресурс]: навч. посіб. для студ. спеціальності 163 «Біомедична інженерія» та 152 «Метрологія та інформаційно-вимірювальна техніка» / В.В. Шликов, В.А. Данілова; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,49 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 109 с.

Анотація: Лабораторний практикум з програмування та побудови медичних мікропроцесорних систем, який викладено у навчальному посібнику допомагає накопичувати й ефективно використовувати отриману інформацію з теоретичного курсу на всіх стадіях навчального процесу, що є важливим для підготовки магістрів та необхідною ланкою у науковому пізнанні практичних основ біомедичної електроніки.

Workshop on discipline for students of specialties 163 "Biomedical Engineering" and 152 "Metrology and information-measuring technique" / Shlykov V. V., Danilova V. A. – Kyiv, Igor Sikorsky Kyiv Polytechnic Institute, 2020. – 109 p.

Educational and methodical edition

MEDICAL MICROPROCESSOR SYSTEMS

Workshop on discipline for students of specialties 163 "Biomedical Engineering"
and 152 "Metrology and information-measuring devices"

Authors: ***Vladyslav Shlykov***, Doctor of Engineering, Associate Professor,
Valentyna Danilova, Senior Lecturer, Department of Biomedical Engineering

Editor-in-Chief: *Viktor Zubchuk*, Ph.D., Associate Professor

Reviewers: *Igor Khudetskyy*, Doctor of Medicine, Professor,
Volodymyr Sydorets, Doctor of Engineering, Professor

Edited by the authors

Підп. до друку Формат Папір друк. №3. Друк офс.
Ум. друк. арк. 0,93. Обл.-вид. арк. 1,0. Зам. № 000. Наклад 100 пр.

CONTENT

Introduction

The purpose of practical classes and laboratory work in the discipline "Medical microprocessor systems", performed using software in the programming environment of microprocessors Texas Instruments (Code Composer Studio) and digital microprocessors of the Texas Instruments DSK6400 family, models of electrical equipment using software and Instruments software LabVIEW 2010, is:

- studying the principles of construction of digital electronic components of microprocessor systems;
- modeling of control modes of microprocessor systems;
- acquisition of skills in working with the electrical equipment;
- acquisition of microprocessor-based control of skills;
- acquisition of skills of preparation, carrying out and documentation of results of researches of functioning of microprocessor systems.

As a result of laboratory work, the student must be able to formulate definitively the requirements for microprocessor system (MPS) parameters. When choosing a type of microcontroller (MC) for the design of motor vehicles it is necessary to learn to consider the following main characteristics:

- capabilities of peripherals;
- the amount of ROM programs and RAM data and the possibility of their increase;
- set of commands and methods of addressing;
- digit;
- performance;
- requirements for the power source and power consumption;
- possibility of delivery in different variants of constructive execution;
- cost in different versions;
- availability and availability of effective tools for programming and debugging MK.

Laboratory research assignments are designed for 2 academic hours. The results of the research and measurements should be documented and presented to the teacher at the end of the class. In the course of laboratory work, a report shall be drawn up, which shall include:

1. Title page of the report.

2. Theoretical information on the topic of the study.
3. ICS schemes for which research and measurements were carried out.
4. Software code that illustrates the processes studied.
5. Conclusions on the results of research and measurements.
6. Answers to control questions.

1. TMS320C62x / C64x / C67x Signal Processor general infromation

The Digital Signal Processor (DSP) TMS320C62x / C64x / C67x series consists of three main parts:

- processor (or core) C62x / C64x / C67x;
- peripheral devices;
- memory.

The computing kernel performs the mathematical processing by referring to the program stored in the programs memory and the data contained in the data memory. The program memory contains programs that the digital signal processor uses to process data. Data memory contains information and data that needs to be processed. The I / O subsystem has an interface that provides a range of functions for communicating with peripherals.

A block diagram for devices based on TMS320C62x / C64x / C67x processors is shown in fig. 1.1.

The flowchart includes digital devices consisting of program memory (Program RAM) and data memory (Data RAM) chips that can be configured as cache memory. The concurrency of the program code is determined during compilation and executed by the hardware of the processor, since no data dependencies are checked during execution. The program memory stores a 256-bit program that accesses eight 32-bit instructions each cycle. Peripherals include Direct Memory Access Controller (DMA), Shutdown Logic, External Memory Interface (EMIF), Serial Ports, Expansion Bus (EXB) or Host Port, and Timers.

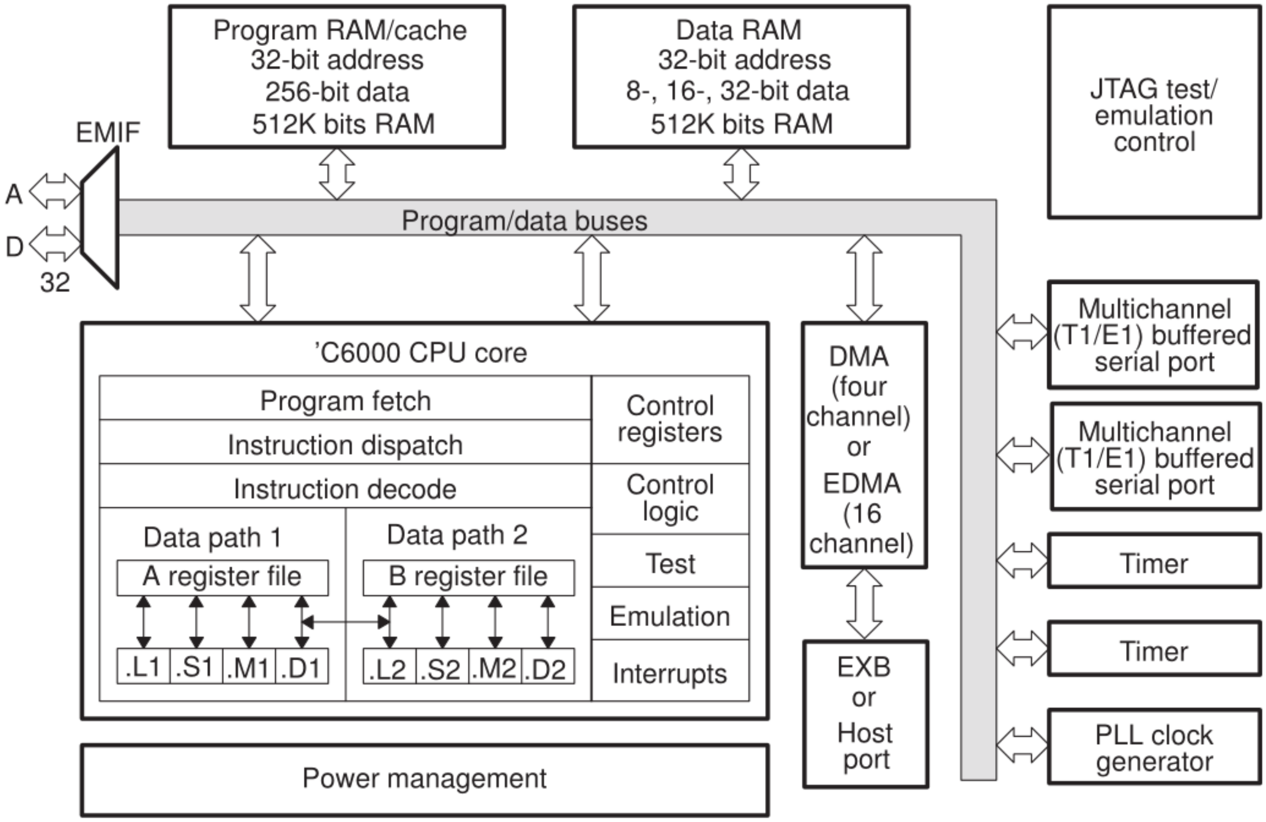


Fig. 1.1. Flowchart for devices based on TMS320C62x / C64x / C67x processors

The TMS320C6000 signal processor platform includes a memory chip for applications and data that can be used as a cache. In addition, the external memory interface (EMIF) can be used to enable external memory type "C6000". The total address memory range of the TMS320C6000 signal processors is 4 GB, which corresponds to a 32-bit internal address representation. Each memory card is divided into internal program memory, internal data memory, external memory space, and internal peripheral space.

The processor has two data paths (A and B) that provide processing. Each data path has four functional units (.L, .S, .M and .D) and a register file containing 16 and 32-bit registers. Functional units perform the operations of logic (Control Logic), moving, multiplying, forming the address and organizing interrupts (Interrupts). All instructions, except for the download and save operation, work in registers. Two data processing units (.D1 and .D2) are solely responsible for all data transfer operations between register files and memory.

Control Registers support both 32-bit and 40-bit fixed-point data. 32-bit data can be contained in any general purpose register, and 40-bit data is placed in two registers. Each function module directly reads and writes data to a register file within its own data path. That is, the units .L1, .S1, .D1 and .M1 write data to log A, and blocks .L2, .S2, .D2 and .M2 write data to log B.

Internal data memory consists of two blocks of eight 16-bit banks. This allows the processor to simultaneously load data with double precision in the same cycle as data access through DMA. External memory interface (EMIF) includes synchronous dynamic RAM (SDRAM), synchronous batch RAM (SBSRAM) and asynchronous memory, providing the processor with external memory. The EMIF interface allows reading of 8-bit and 16-bit memory, as well as support for downloading data from memory modules (EEPROM, EPROM and PROM), supports high-performance SDRAM interfaces.

Timers have two alarm modes, and can be synchronized internally or externally. Each timer has an input pin (TINP) and an output pin (TOUT). The TINP output can be used as a general purpose input, and the TOUT output can be used as a public output. The use of a second system bus (Program / Data Buses) for I / O devices ensures the separation of external devices between two ports (Serial Port).

1.1. TMS320C6455 digital signal processor

Digital Signal Processors (DSPs) TMS320C64x (including DSP TMS320C6455) are high-performance fixed point DSPs based on the TMS320C6000 platform. The C6455 platform is based on the third-generation VelociTI Long-Speech Instruction (VLIW) architecture developed by Texas Instruments, making this DSP an ideal choice for use in video and telecommunications equipment, image processing systems, medical applications, and video conferencing systems. The C64x DSP is programmable from the bottom up in code with its precursors, which are based on the C6000 DSP platform.

The block diagram of the DSP processor TMS320C6455 is presented in fig. 1.1.1.

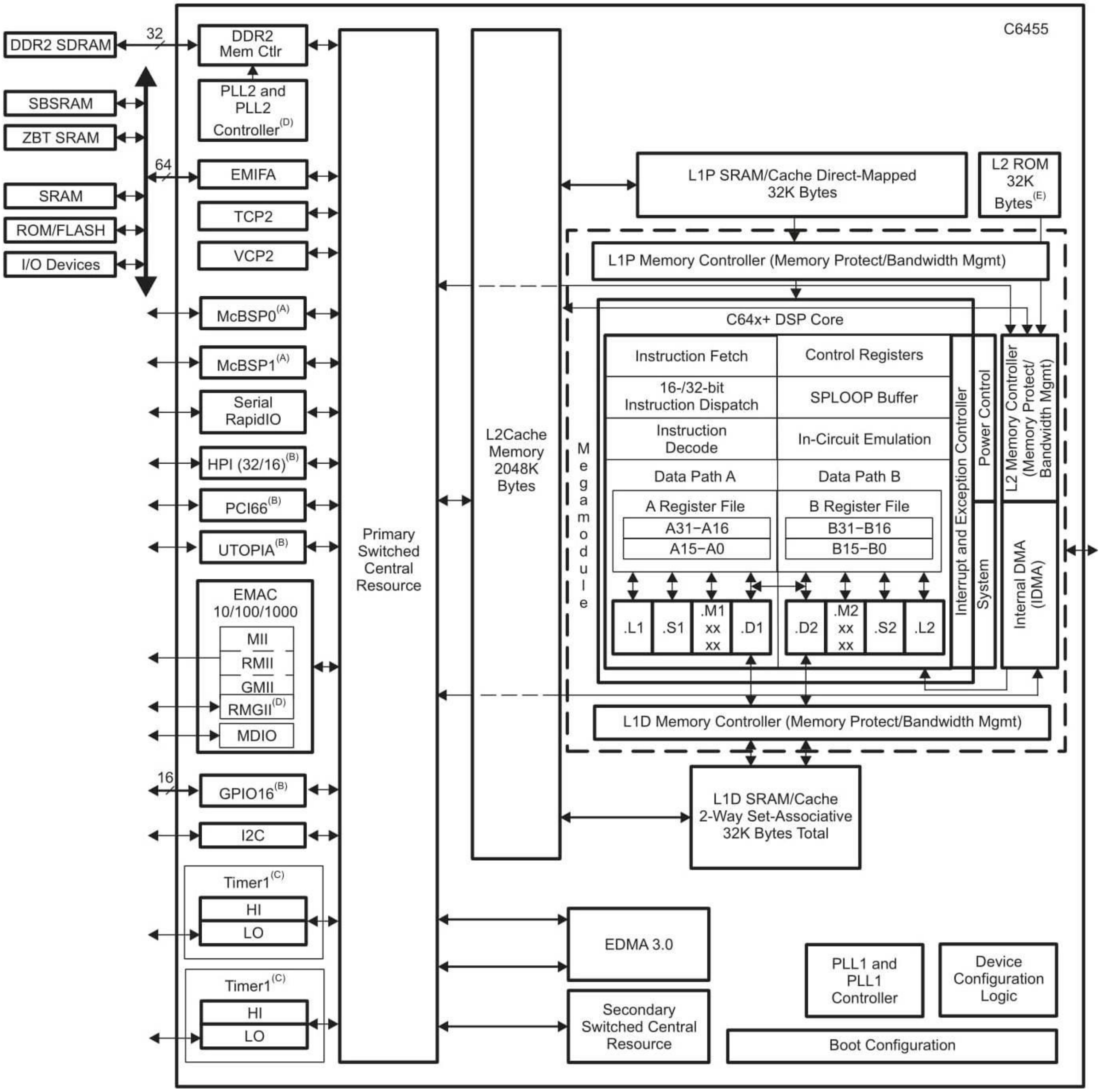


Fig. 1.1.1. Flowchart of DSP Processor TMS320C6455

The TMS320C64x processors are manufactured using 90 nm technology and clocked at 1 GHz, delivering 8,000 million instructions per second (or 8 billion 16-bit-per-second), making the C6455 a useful tool for solving complex digital processing tasks. The C6455 has the operational flexibility of high performance controllers and the processing capabilities of matrix processors.

Key Features and Features of the TMS320C6455 Processor:

1. High Performance Fixed Point DSP (C6455)

- duration of the instruction cycle 1.39; 1.17; 1 ns
- clock frequency 720 MHz, 850 MHz, 1 GHz
- eight 32-bit instructions / cycle
- 5760, 6800, 8000 million instructions per second
- 5760, 6800, 8000 million accumulations per second (16 digits)
- commercial temperature range (0°C ... + 90°C)

2. TMS320C64x DSP Core

- Specialized SPLOOP instruction
- compact instructions (32- / 16-bit)
- extension of the instruction set
- exception handling

3. TMS320C64x architecture with L1 / L2 megamodal memory

- 256 Kbps (32Kbyte) L1P application cache (spreadsheet organization)
- 256 Kbps L1D data cache (32 KB) (2-way associative set)
- 16Mbps (2048KB) L2 RAM / L2 Cache (Flexible RAM / Cache Location)
- Countdown timer

4. Advanced Viterbi Decoder Co-processor (VCP2)

- Support up to 694 AMR 7.95 Kbps.
- programmable code options

5. Advanced Turbo Decoder Co-processor (TCP2)

- Supports up to eight 2 Mbps 3GPP (6 iterations)
- programmable turbo code and decryption parameters

6. Support for Little Endian and Big Endian byte order

7. 64-bit / 133 MHz external memory interface

- direct connection to asynchronous memory (static RAM, flash memory and ROM ROM) and synchronous memory (SBSRAM and ZBT SRAM)
- support for connection to standard synchronous devices and specialized logic (FPGA, CPLD, ASIC)
- total addressable external memory space is 32 MB

8. 32-bit DDR2 memory controller (DDR2-500 SDRAM)

9. EDMA controller (64 independent channels)

10. Four 1 x RapidIO (or 4x) serial channels compatible with version 1.2

- transmission speeds of 1.25-, 2.5-, 3.125 Gb / s.
- messaging, DirectIO support, extension for error handling and channel load management
- I / O compatible with IEEE 1149.6

11. 32/16-bit (HPI) Host Port Interface

12. Master / slave PCI interface 32-fold. / 66 MHz / 3.3V compliant with PCI 2.3

13. One I2C bus

14. Two Multichannel Buffered Serial Ports (McBSP)

15. 10/100/1000 Mbps Ethernet MAC (EMAC) controller

- IEEE 802.3 compatibility
- support for multiple media-independent interfaces (MII, GMII, RMII and RGMII)
- 8 Separate Transmission Channels (TX) and 8 Separate Reception Channels (RX)

16. Two 64-bit general-purpose timers that can function as four 32-bit timers

17. Universal Testing and Work Interface for ATM (UTOPIA)

- UTOPIA Level 2 Slave ATM Controller
- 8-bit reception and transmission at up to 50MHz in one direction
- cell format up to 64 bytes, user-defined

18. General purpose 16 I / O lines

19. System PLL and PLL controller

20. Additional PLL and PLL controller for EMAC and DDR2 memory controller

21. Border scan according to? IEEE-1149.1 (JTAG)

22. Housing with matrix arrangement of spherical pins (BGA) 697-pin (suffix ZTZ), pitch of conclusions 0.8 mm

23. CMOS technology 0.09 microns with 7-level copper metallization

24. Power 3.3-, 1.8-, 1.5-, 1.2-I input-output, internal power 1.2V

CPU core. The C64x DSP core consists of 8 function blocks, two register files and two data channels. Similar to other C6000 DSPs, two of the eight function blocks are multiple devices or ".M" units. Each .M block in C64x performs four 16 x 16 multiplications per clock cycle. If you use a clock speed of 1 GHz, it means a performance of 8 billion 16-fold. multiplication-accumulation per second. In addition, each C64x kernel multiple device can calculate one 32 x 32 multiplication or four 8 x 8 multiplications per clock cycle.

Serial interface. The C6455 contains a RapidIO serial interface. This high-performance device dramatically improves system performance and reduces system costs when using software applications that require the installation of multiple DSPs, such as video or telecommunications equipment, medical equipment, or image processing systems.

Memory organization. DSP C6455 integrates a large amount of memory, which is organized as a two-tier system. Level 1 (L1) application memory and data memory is 32KB. This memory can be configured as a tabular RAM, cache, or some combination of the two. In Cache mode, L1 (L1P) applications are a cache with Direct Mapped Cache, and L1 (L1D) is a 2-channel associative cache access. Level 2 (L2) memory is shared as a memory of applications and data with a total size of 2 MB. L2 memory can also function as a tabular RAM, cache, or some combination of the two. The C64x also includes a 32-bit configuration port, an internal RAP controller, system components for reset / load control, interrupt / exception management, power reduction control, and a 32-bit timer.

Composition of peripherals. Peripherals include: I2C bus module; two multichannel buffered serial ports (McBSP); 8-bit UTOPIA slave port; two 64-bit general-purpose timers that can work as four 32-bit timers; User configurable 16 or 32 bit host port interface (HPI16 / HPI32); PCI interface; 16-pin General Purpose Input / Output (GPIO) port with programmable interrupt / event generation modes; a 10/100/1000 Ethernet (EMAC) media access controller, which is an effective interface between the C6455 processor core and the network; an I / O Management Unit (MDIO) (part of EMAC) that continuously polls all 32 MDIO addresses to search all physical devices in the system; external memory interface (64-bit EMIFA) for direct connection to synchronous and asynchronous memory; as well as a 32-bit DDR2 SDRAM interface.

Control ports. I2C ports allow the C6455 to easily manage peripherals and communicate with the host processor. In addition, a standard multichannel buffered serial port (McBSP) can be used to communicate with peripherals containing a serial SPI interface.

Advanced coprocessors. The C6455 contains two high-performance coprocessors - an advanced Viterbi decoder co-processor (VCP2) and an advanced turbo-decoder co-processor (TCP2), which significantly accelerate channel decryption. VCP2 runs at a DSP divided by 3 and can decode up to 694 adaptive multi-rate (AMR) voice channels at 7.95 Kbps (K = 9, R = 1/3). VCP2 supports fixed lengths K = 5, 6, 7, 8, 9, speeds R = 3/4, 1/2, 1/3, 1/4, as well as flexible polynomials, thus generating software or hardware results. TCP2 operates at a DSP frequency divided by 3, and can decode up to 50 channels at 384 Kbps or up to 8 turbo-coded channels at 2 Mbps (allowing 6 iterations). TCP2 implements the "Max * Log-Map" algorithm and is designed to support all the polynomials and speeds required by 3GPP and 3GPP2, as well as to fully program the parcel length and turbo-time seal. Decoding parameters are set as well, such as the number of iterations and the stop criteria. The connection between VCP2 / TCP2 and DSPC is organized using the EDMA controller.

1.2. Signal Processor Programming Tools

The TMS320C6000 programming tools have signal-specific IT instructions that are directly supported in C language. This set of instructions is intended to facilitate 16-bit operations on a 32-bit architecture when using Coding. IT provides C6000 compatibility with emulation systems that support hardware and software debugging of DSP systems using a JTAG emulation cable.

File model of digital processing. For the software modeling of digital processing algorithms on processors (DSP) TMS320C62x / C67x, the file model shown in Fig. 1.2.1.

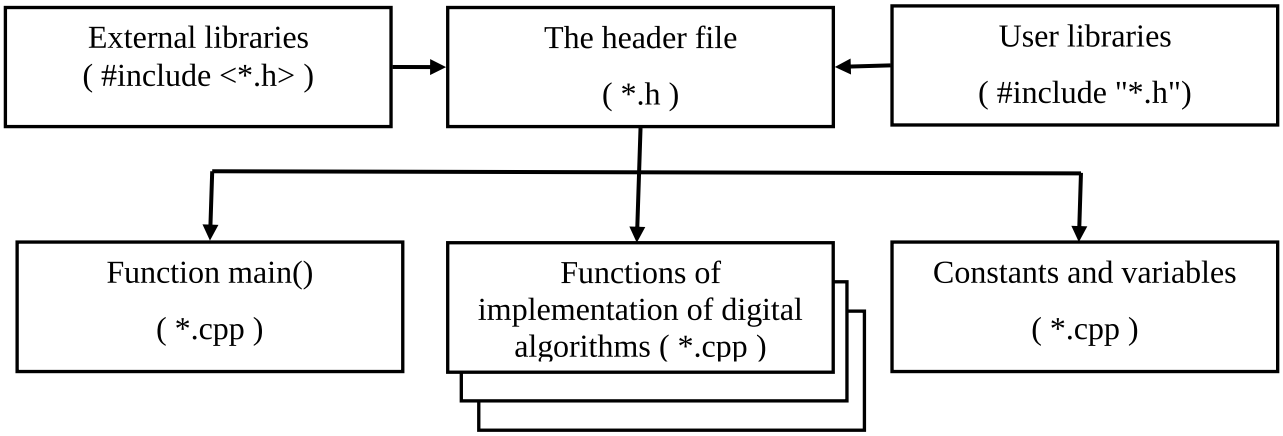


Fig. 1.2.1. File model for modeling digital processing algorithms

The essence of this approach is that the inputs and reactions to them are in files called test vectors. When developing a specific algorithm (for example, digital filtering), an influence is selected, the reaction to which is known in advance. When processing an input file using a software implementation of the algorithm, they obtain the output file and make sure that the calculated and obtained results.

The basic requirements for the file model are simple and are as follows:

1. There is only one header (* .h) in the code. This file connects all required external libraries, declares constants, arrays, functions, macros, user-defined types, etc .;

2. Creating a contextual structure of the algorithm, which includes all the necessary global pointers, constants, variables, etc .;

3. Absence of static and dynamic variables and constants. If necessary, they should not be global and should be included in the contextual structure;

4. Creating global variables, constants, structures, and arrays is done in one separate file (* .cpp). All global variables, constants, etc. must be declared in one header file;

5. Each special function must be placed in a separate file (* .cpp) and must be declared in the header;

6. Data exchange between special functions is done through a pointer to the contextual structure of the project;

7. Function code must be based on the C basic operators that meet the requirements of the eXpresDSP standard;

8. Functions that implement a digital processing algorithm should not be tied to a specific task. Communication with I / O peripherals is organized through the input and output buffers;

9. Arrays are processed through pointers to these arrays included in the context structure;

10. The main () function should not include program code that implements a digital signal processing algorithm. The function of this function is to fill the input buffer with data from a file, to call the main function of the digital processing algorithm and to write the result from the output buffer to the output file.

In accordance with the structure of the file model, the first step checks the compliance of the code in each file with the requirements of language C. In the second stage - the relationship between the parts of the program (functions, constants, arrays connected by external libraries, etc.) is established. The code collected in Code Composer Studio (files with extension * .h and * .cpp) must be converted into an executable module (file with extension * .exe).

The operation of creating an executable module is ensured by the compiler and includes two steps:

1. Broadcast each file with the program code (only files with extension * .cpp are broadcast, header files are automatically connected by the #include <*. H> directive) into object modules (files with extension * .obj);

2. Composition or linking of object files into executable module.

Design tools. DSP TMS320C6455 is supported by a complete set of design tools, including: a new C-compiler, an assembler optimizer to simplify programming and CPU time allocation, and the Texas Instruments (TI) Code Composer Studio debugger Windows for clarity of program source execution.

The software model for the implementation of digital signal processing algorithms must meet the requirements of the eXpresDSP standard implemented by the manufacturer of Texas Instruments signal processors. The model provides maximum portability of the developed software C-code between different families of digital signal processors**.**

1.3. Design tools for DSP Test Integration VIs

LabVIEW 2010 development system uses specialized Toolbox DSP Test Integration VIs for designing virtual devices using visual I / O data components using LabVIEW 2010 development system, which includes means for interfacing the TMS320C6455 DSP with the user interface in LabVIEW Figure 1.3.1.

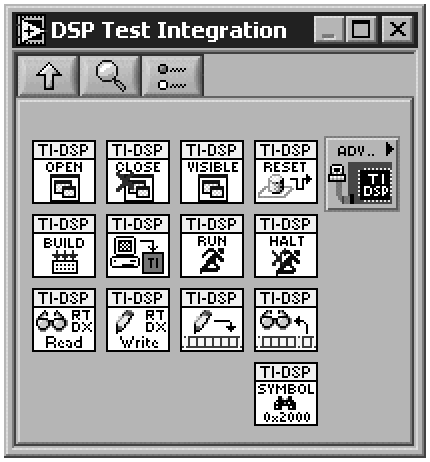
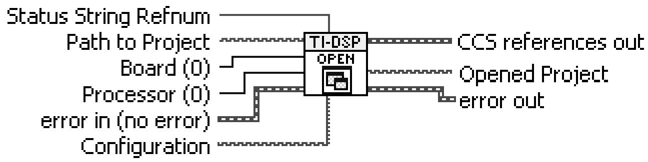
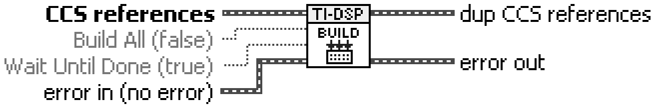
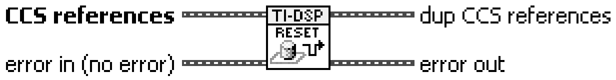
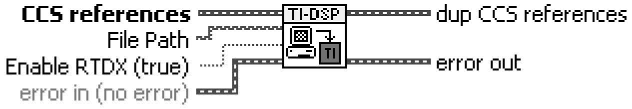


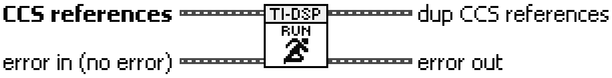
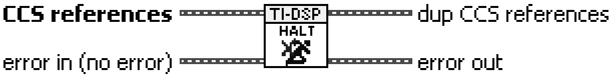
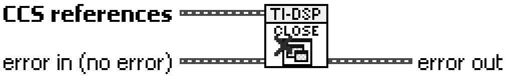
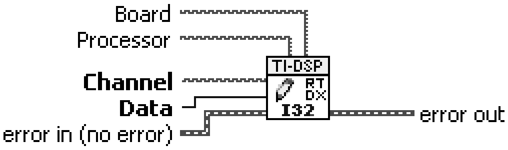
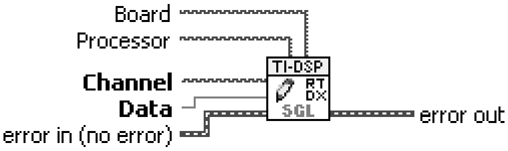
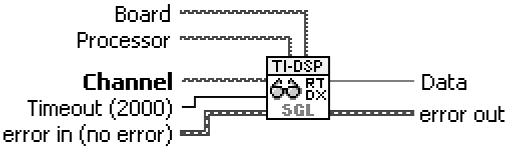
Fig. 1.3.1. Visual components of Toolbox DSP Test Integration VIs

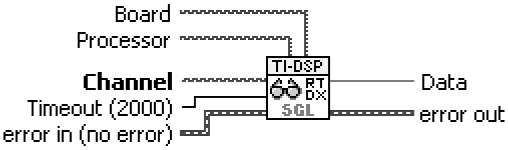
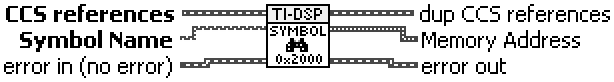
Using DSP Test Integration VIs components, LabVIEW virtual machine interacts with software in the Texas Instruments (TI) Code Composer Studio (CCS) application through dedicated RTDX channels and creates an integrated development environment (IDE) for equipment design and testing.

The list of components of DSP Test Integration VIs in LabVIEW is as follows: CCS Open Project, CCS Build, CCS Reset, CCS Download Code, CCS Run, CCS Halt, CCS Close Project, CCS RTDX Write, CCS RTDX Write SGL, CCS RTDX Read, CCS RTDX Read SGL, CCS Symbol to Memory Address. Assignment of DSP Test Integration VIs components to LabVIEW is presented in Table 1.1.

Tab. 1.1. Assigning DSP Test Integration VIs Components

Components of DSP Test Integration VIs	***Function***
CCS Open Project  	Running IDE Composer Studio code. If CCS is already open, the project opens without running another copy of CCS.
CCS Build  	Starts the project build operation in CCS IDE, re-creating the compiled .OUT file.
CCS Reset  	Stops the destination code on the TMS320C6455 processor and restores the CPU registers to their default values.
CCS Download Code  	Uploads a compiled .OUT file to the TMS320C6455 processor, or you can download a previously compiled .OUT file by specifying the path to it.

Components of DSP Test Integration VIs	*Function*
CCS Run  	Runs a .OUT file in the CCS IDE on the TMS320C6455 processor.
CCS Halt  	Stops a .OUT file running on the TMS320C6455 processor.
CCS Close Project  	Closes the open project file (* .pjt) and closes all CCS IDE links.
CCS RTDX Write  	Digital data recording in RTDX channel.
CCS RTDX Write SGL  	Digital data recording in the SGD RTDX channel.
CCS RTDX Read  	Read numeric data from RTDX.

Components of DSP Test Integration VIs	*Function*
CCS RTDX Read SGL 	Read numeric data from the RTDX channel in SGL format.
CCS Symbol to Memory Address 	Returns the symbol memory addresses for the destination code loaded on the TMS320C6455 processor.

LabSPIE's DSP Test Integration VIs Integration Toolkit includes library function identifiers in CCS and VI that use TI Real-Time Data Exchange ™ (RTDX ™) software technology to communicate with the DSP TMS320C6455 processor via RTDX channels. The toolkit is used to automate exchange with VI components and to work with CCS IDE (* .pjt) software programs. RTDX channels for component VI communications and VI tool memory allow programmatic communication of target code in CCS. The DSP integration tool with LabVIEW also includes LabVIEW Debugging Workbench tools for implementing RTDX ™ communication that can be used to interact with RTDX channels on RTDX-enabled developer platforms.

2. Medical devices with microprocessor control

At the stage of development of the structure of the controller the composition of the available and definitive hardware modules, protocols of exchange between modules, types of connectors are finally determined. Preliminary warning of the design of the controller is performed. The part of the software determines the composition and communication of the software modules, programming language. At the same stage, the choice of design and debugging tools is made.

The ability to redistribute functions between hardware and software at this stage exists, but it is limited by the characteristics of the MK already selected. It should be borne in mind that modern MKs are issued, as a rule, by a series of controllers that are compatible in software and design, but differ in their capabilities (amount of memory, a set of peripherals, etc.). This makes it possible to select the controller structure in order to find the most optimal implementation option.

After developing the hardware and software structure, further work on the vehicle structure can be divided into several steps. These stages include the development of a general schematic diagram, wiring of the board topology, layout layout and its autonomous debugging.

The stage of development of the control algorithm is the most responsible, since errors of this stage are usually detected only when testing finished products and lead to the need for expensive recycling of the entire device. The development of an algorithm is usually to choose one of several possible variants of the algorithms, which differ in the ratio of volume of software and hardware.

It should be borne in mind that the maximum use of hardware simplifies the development and provides a high speed controller as a whole, but is usually accompanied by an increase in cost and power consumption. This is due to the fact that the increase in the proportion of hardware is achieved either by choosing more sophisticated vehicles or by using specialized interface schemes. Both result in increased cost and energy consumption. Increasing the share of software can reduce the number of vehicles and the cost of hardware, but it leads to a decrease in performance, increase the required amount of internal memory MK, increase the development time and debugging software.

The content of the stages of software development, its translation and debugging on the models depends significantly on the system tools used. Currently, MK resources are sufficient to support high level programming. This allows you to take full advantage of structural programming, to develop software using individual modules.

2.1. Digital stethoscope

The purpose of the work is to design a digital device in LabVIEW with microprocessor control and to study the functions of arrays in the microprocessor programming environment Code Composer Studio.

Theoretical information

Digital stethoscope - a device for listening to the noise of the internal organs: lungs, bronchi, heart, blood vessels, etc. The block diagram of the stethoscope is shown in Fig. 2.1.1.

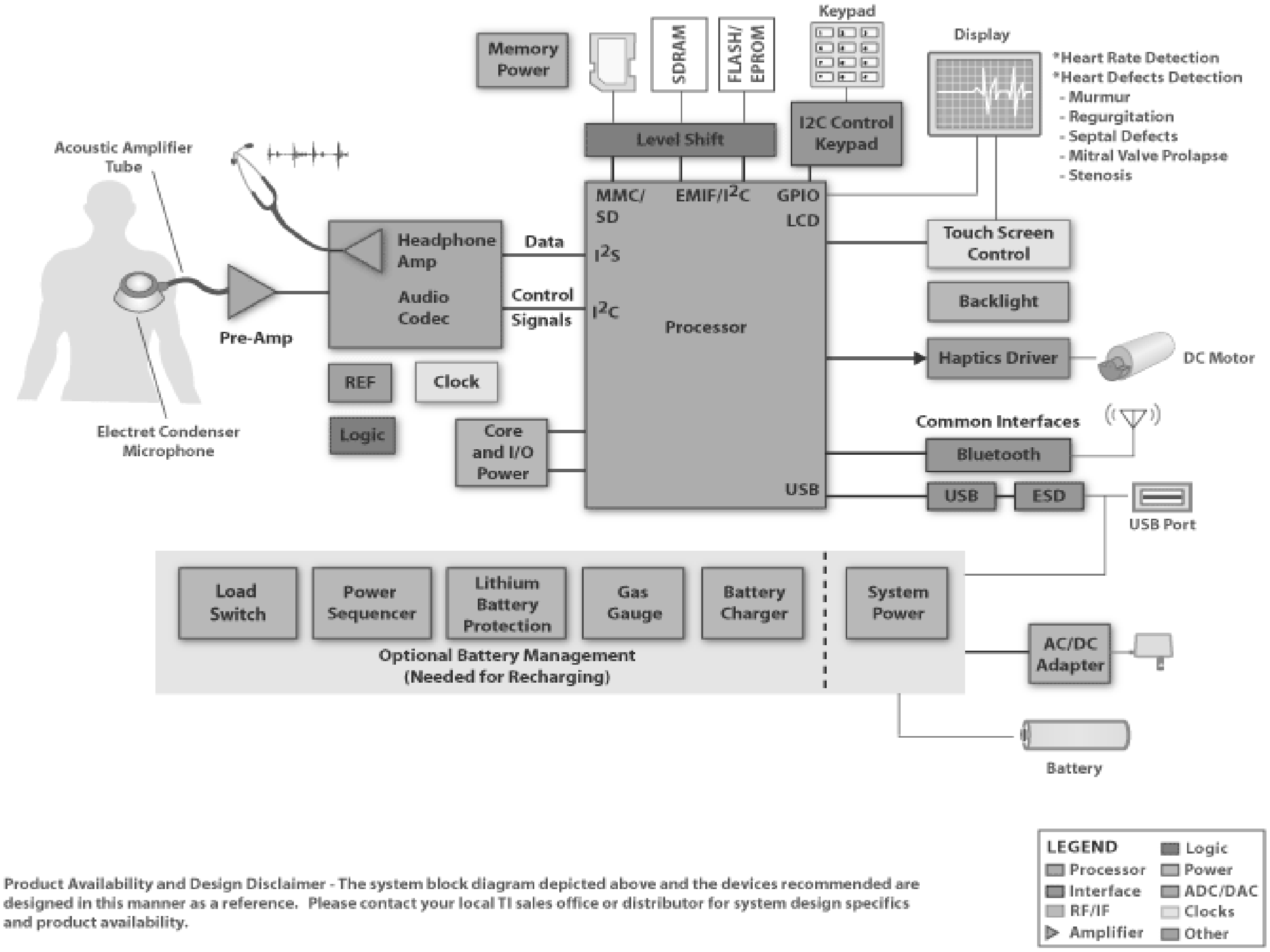


Fig. 2.1.1. Flowchart of the stethoscope

The flowchart of a digital stethoscope consists of several main elements:

- Microphone - sound sensor;

- AudioCodec (audio encoder-decoder), designed to amplify and convert the audio signal of the microphone into digital form, as well as to reverse the processor-processed signal to an analog view for listening to it in the headset of the stethoscope;

- Processor is designed to perform all the special functions of a digital stethoscope, including the implementation of algorithms for finding norms and heart defects.

For normal operation of the stethoscope, it includes such peripherals as MMC / SD memory cards with Level Shift for long-term data storage, Touch Screen Control Back Backlight for visual signal display, additional keyboard (Keypad) for controlling the stethoscope, Bluetooth and USB data communications to connect to your computer, and AC adapter with battery charger.

Modern capabilities of the digital stethoscope:

- sound amplification is 24 times stronger than a traditional acoustic stethoscope;
- system of suppression of unwanted noise of the environment and the organism of the patient (ANR) up to 75%;
- three filter modes for auscultation of the heart, lungs and other body sounds: unwanted bell (20 ... 200 Hz), diaphragm movements (100 ... 500 Hz) and noise filtering in extended mode (20 ... 1000 Hz);
- the ability to record, store and play sounds on six sound tracks at normal and half speed;
- transfer of recorded sounds through infrared port to another stethoscope or personal computer for further analysis of received audio data in the form of phonocardiograms and spectrograms;
- the ability to receive information in the form of phonocardiograms on the monitor of a personal computer.

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.1.2 and Fig. 2.1.3).

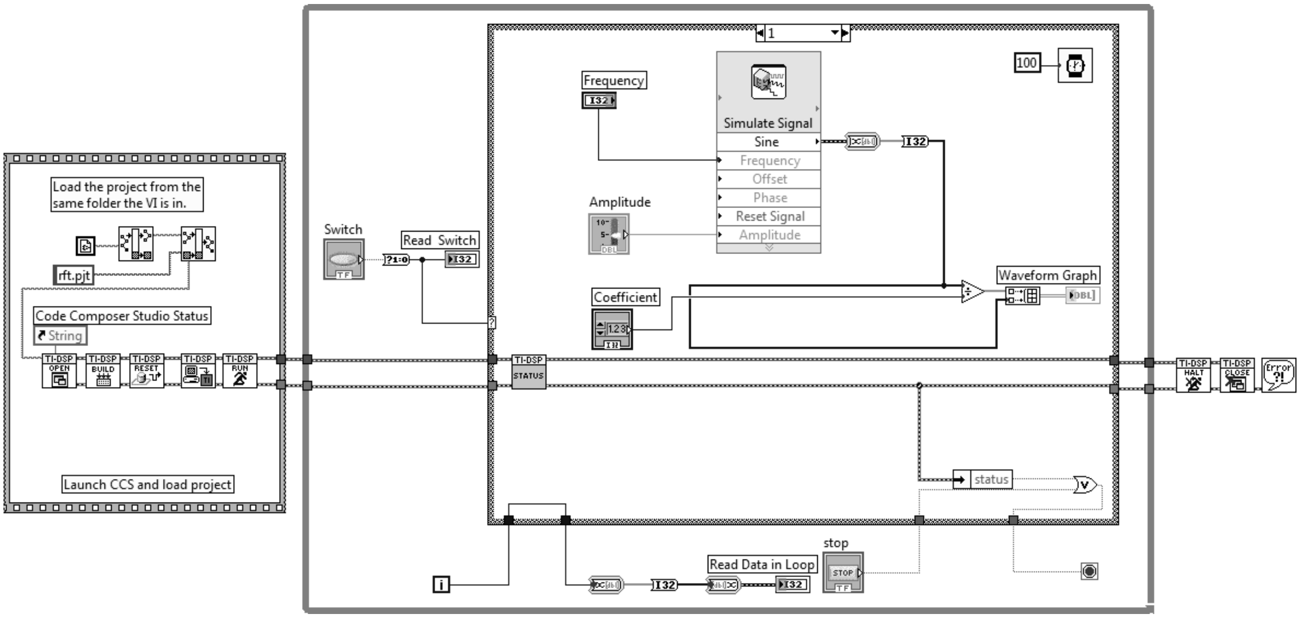
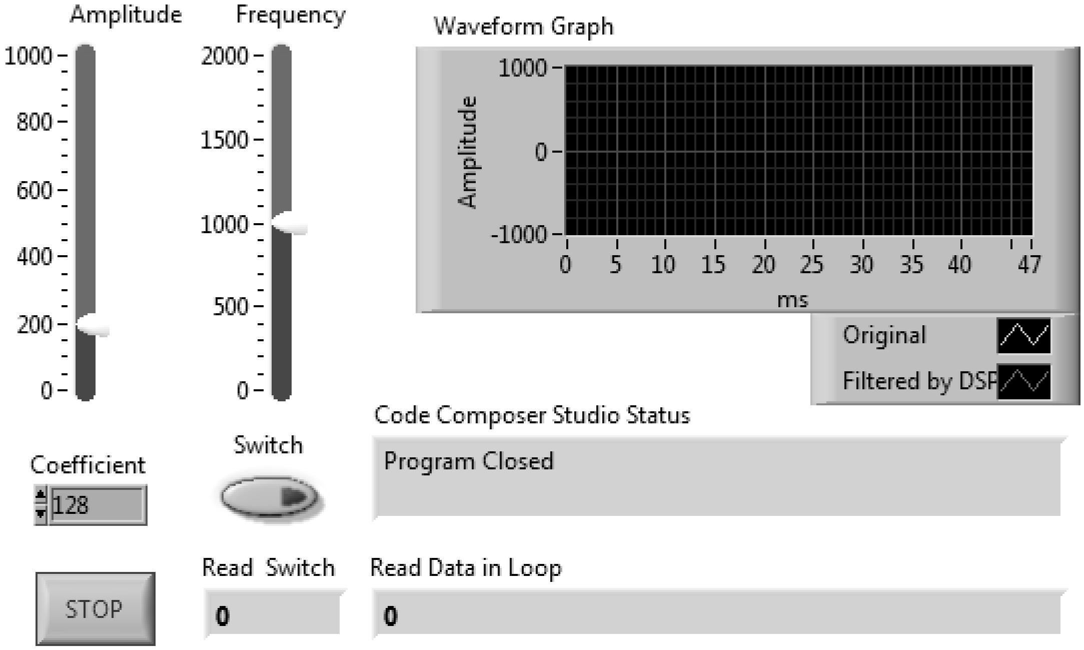


Fig. 2.1.2 Schematic of the virtual instrument in National Instruments LabVIEW 2010



Coefficient
128

Code Composer Studio Status
Program Closed

Read Switch
0

Read Data in Loop
0

Fig. 2.1.3 Appearance of the instrument control system

2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and functions of arrays in the Code Composer Studio microprocessor programming environment.
3. Document the data from the arrays obtained in the experiments for the report. To formulate

in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 1.1.2 and Fig. 1.1.3. Initialize data and variables in a work project in the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
        #define BUFFER_SIZE 32
        int i;
        int *ptr1, *ptr2;
        int array[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
                11, 12, 13, 14, 15, 16, 17, 18,
                19, 20, 21, 22, 23, 24, 25, 26,
                27, 28, 29, 30, 31, 32};
```

1. To use the method of working with arrays by reference in the programming environment of microprocessors Code Composer Studio:

```
// Implementing an array reference
        for(i = 0; i < 32; i++)
        {
        *ptr1 = array[i] + 1;
        *ptr2 = array[i] - 1;
                // Array processing function
                function (ptr2, ptr1);
        }
```

1. Implement the procedure of processing arrays with data in the programming environment of microprocessors Code Composer Studio:

```
// Array processing function
        void function (int c [BUFFER_SIZE], int d [BUFFER_SIZE])
        {
                int i;
```

```
                for(i = 0; i < 64; i++)
                {
                c[i] += d[i];
                d[i] += 1;
                }
        }
```

Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. What is the function of arrays?
2. What considerations is the Texas Instruments DSK6400 microcontroller?
3. Explain the method of working with arrays by reference.

2.2. Automated external defibrillators

The purpose of the work is to design a digital instrument in LabVIEW with microprocessor control and to use the CLK timer in the Code Composer Studio microprocessor programming environment.

Theoretical information

Automated External Defibrillator (AVD) is a sophisticated microprocessor device designed to monitor, evaluate and automatically treat patients with life-threatening heart rhythms.

AVDs record ECG signals from electrodes, perform rhythm analysis with the help of a special program (algorithm) for recognizing arrhythmias. After electrocardiographic rhythm registration, the automatic external defibrillator algorithm selects one of two possible situations: whether or not to run an electrical discharge. If the defibrillator is fully automatic, it automatically switches on or off the discharge; semi-automatic AVD - gives only advice in the form of a voice message and / or displays information on the monitor screen. An automatic external defibrillator can also report possible artifacts that occur on the electrocardiogram.

The main elements of a defibrillator are the following: high-voltage power supply, storage capacitor, additional inductor and electrodes for defibrillation. The defibrillator is arranged in such a way that the discharge of its capacitor at any amount of stored energy occurs within 7 ... 10 ms ("useful time" of the heart start) through the inductor, by which the initial high-voltage part of the discharge current is eliminated, and the duration of the discharge increases . As a result, the efficiency of the discharge is greatly increased and the harmful effects of excessively high currents passing through the heart are reduced. In addition, the presence of inductance in the circuit reduces the voltage on the object (ie, the chest and the heart) by 3-4 times compared to the charge voltage of the capacitor.

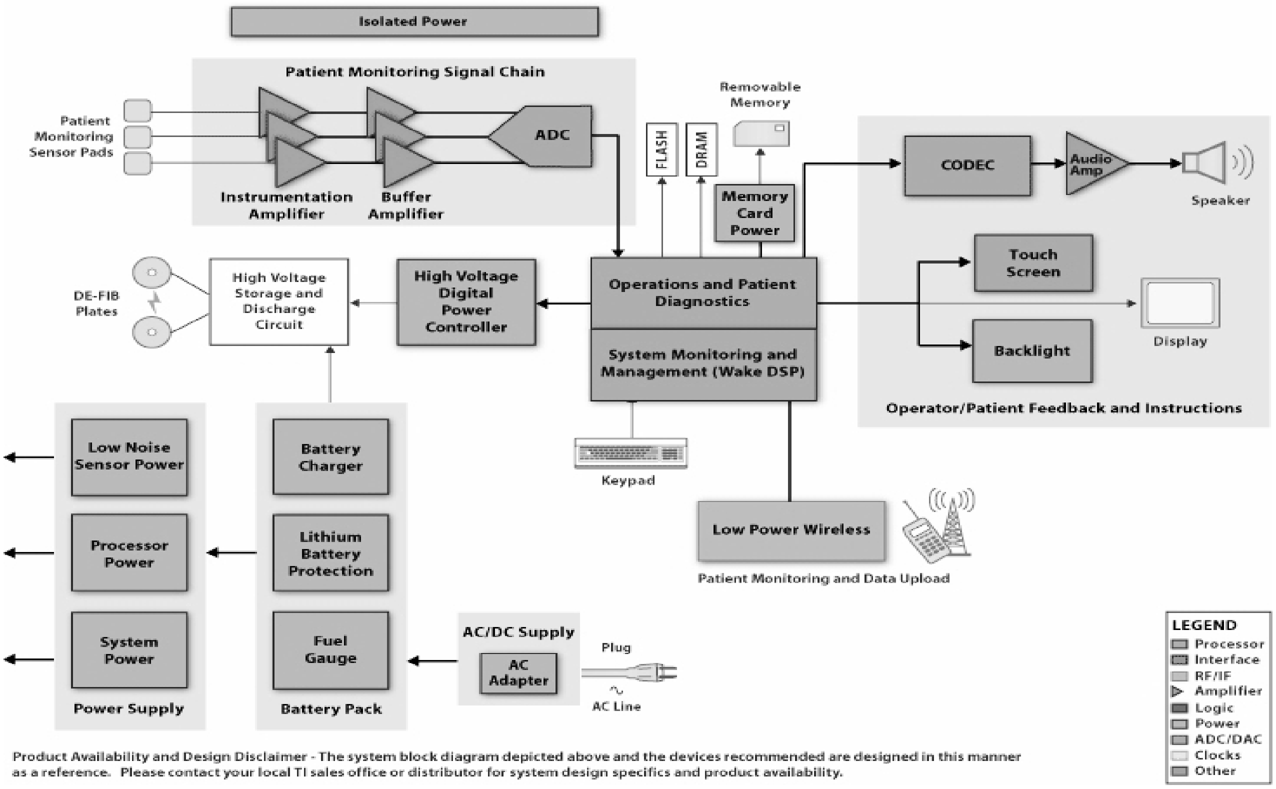
Many digital defibrillators have a heart rate audio recording function for post-event analysis. All digital AVDs have functions for storing and retrieving a patient's ECG model.

One of the AVD inputs can be connected to a sound recording microphone in the operating room. The microphone signal is used to compensate for external interference. The interface signals coming to the AVD inputs from the ECG electrodes mounted on the patient have a very small amplitude (<10 mV). Therefore, the use of an instrument amplifier is necessary. Instrumental amplifiers shall meet the following requirements: amplitude range of the input signals 0,1 ... 10 mV; high input impedance (> 5 MOm); low input leakage current (<1 μA); Frequency response constancy in the range 0.1 ... 100 Hz.

ECG and microphone inputs are digitized and processed by the DSP. Most AVDs use a 16-bit processor (MSP430 type) and therefore work well with the 16-bit ADC used for ECG digitization and voice input. The amplified ECG signal must have a minimum signal / noise ratio of 50 dB in the frequency band from 0.1 Hz to 100 Hz. Audio recordings typically have a bandwidth of 8 kHz and a minimum signal / noise ratio of 65 dB. The signals from the microphone input must also be amplified with a programmable maximum gain of 40 dB.

The AVD can synthesize audio output with adjustable output volume either on the headphone speaker or on an 8 ohm speaker. For AVD, the 16-bit TLV320AIC20 audio codec is best suited because it includes two channels: ADC, DAC, microphone amplifier, driver for headphones, and an 8 ohm loudspeaker with volume control controlled by separate DSPs.

The block diagram of the AVD is presented in Fig. 2.2.1.



Product Availability and Design Disclaimer - The system block diagram depicted above and the devices recommended are designed in this manner as a reference. Please contact your local TI sales office or distributor for system design specifics and product availability.

Fig. 2.2.1. Flowchart of automated external defibrillator

Modern features of automatic defibrillators:

- 98% - efficiency of the first category (caused in the first minutes of cardiac catastrophe);
- portability, protection against shocks, falls, vibrations - unlimited possibilities of using defibrillators in any extreme situations;
- constant readiness for use - no need for constant recharging of the device, system and indication of automatic daily self-control;
- biphasic discharge provides the most effective and least traumatic defibrillation;
- automatic selection of discharge power depending on the impedance of the tissues of the patient, its weight and height, that is, the maximum efficiency and the safest discharge that most effectively affects the heart are selected;
- simplicity and ease of use - the device is controlled by a single button;
- voice prompts for staff - minimizing errors and optimizing successful defibrillation;

- integrated electrodes - adhesive-based electrodes with low transient resistance allow for individually-effective defibrillation (staff hands are released to perform other additional manipulations necessary for successful resuscitation);

- electrodes are not polarized - no need to waste time to resolve the issue with the location of the electrodes, since the device itself specifies the location and sets the electrode polarization;

- the ability to visualize, archive, log all defibrillation events in digital format on a computer;

- possibility of full automatic self-defibrillation without the participation of personnel;

- safety of use, as it is impossible to make an unauthorized discharge, because in the presence of a normal heart rhythm, the defibrillator does not activate (block) the discharge button.

Work task

1. To collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.1.2 and Fig. 2.1.3).
2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and function of controlling the timing in the Code Composer Studio microprocessor programming environment.
3. Document the charts obtained in LabVIEW 2010 for the report. To formulate in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 2.1.2 and Fig. 2.1.3. Initialize data and variables in a work project in the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
        #include <clk.h>
        int i, ret;
        short *point;
        point = (short *) 0x80000000;
```

```
/* Hardware timer counts per microsecond */
Uns counts_per_us;
```

Use the Code Composer Studio microprocessor monitoring timer function:

```
// Log CLK timestamps        counts_per_us = CLK_countspms() / 1000;
        for(i=0;i<(int)counts_per_us;i++)
        {
                ret = ret_sum((short*)point[i],i);
        }
```

To implement procedure of accumulation of time counts of time in programming environment of microprocessors Code Composer Studio:

```
// The function of accumulation of time frames
int ret_sum (short array[restrict], int N)
{
         int count, sum;
        sum = 0;
        for(count=0 ; count < N ; count++)
                sum += array[count];
         return(sum);
}
```

Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. Explain the purpose of the CLK.h software library?
2. How are the NI Programming Function VI virtual interface components used?
3. What is the function of monitoring the timings?

2.3. Digital blood gas analyzer

The purpose of this work is to design a digital device in LabVIEW with microprocessor control and to study the functions of program timer control in the microprocessor programming environment of Code Composer Studio.

Theoretical information

A blood gas analyzer is a tool used to measure the partial pressure of oxygen, carbon dioxide, carbon monoxide and nitrogen in the blood.

The main subsystems of the blood gas analyzer are as follows:

- Analog interface. The signals from the sensors of the chemical analyzer after multiplexing are amplified and digitized for further processing. Sensor Amplifiers are selected to provide the same range of signals from sensors at the ADC input.
- The microcontrollerTMS320C55X performs the analysis of processes caused by gases in the blood and controls the memory interface and peripherals.
- LCD display for displaying data.
- Power management system. Converts battery voltage to voltage required for different function blocks.
- Analog feedback. Converts the signal from the digital output of the microcontroller to an analog signal for self-calibration after the input sensor. At self-control of the system of the microcontroller generates a signal of the given form and checks its passage on all channels.

A block diagram of a portable blood gas analyzer is presented in Fig. 2.3.1.

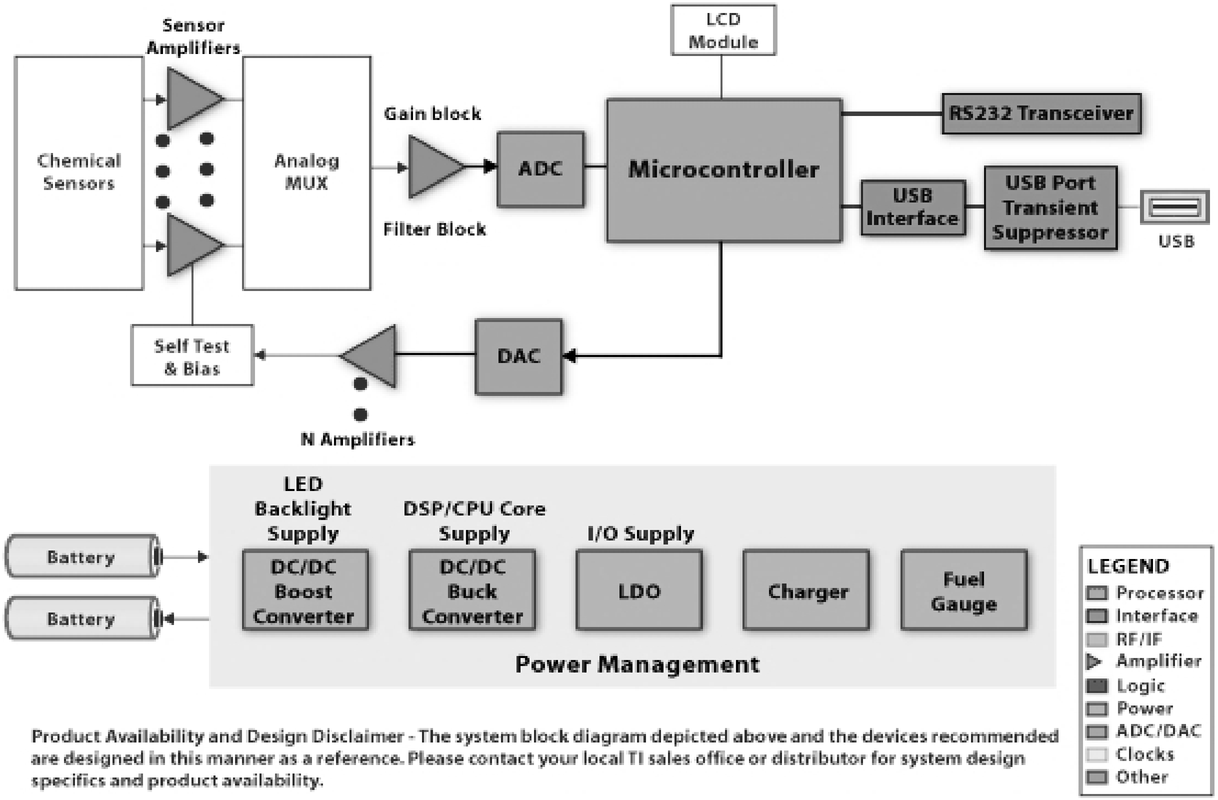
Current capabilities of the blood gas analyzer:

1. Basic working principles of measurement: optical fluorescence and optical reflected photometry. The fluorescent molecules that are part of the sensor specifically and quantitatively bind to the corresponding determining analyte. The sensor (sensor) of PO2 in the cassette has two functions: fluorescence determination of PO2 and photometric determination of total hemoglobin (tHb) and oxygen saturation (SO2).

The analyzer provides the ability to measure pH, PCO2, PO2, Na ++, K +, Ca ++, Cl-, glucose, hemoglobin, SO2. The total weight of the analyzer is less than 5.5 kg, which makes it easy to transport it to the desired place. It can run from its own battery (up to 8 hours) or from the network.

2. Innovative optical technology. Disposable cartridges stored at room temperature. The analysis is carried out by a non-contact method, which avoids regular maintenance of the electrodes, as in other analyzers.

3. Wide temperature range. The thermostat in the measuring unit warms up to 37 C. This allows you to work even in cold rooms, including operating rooms.



Product Availability and Design Disclaimer - The system block diagram depicted above and the devices recommended are designed in this manner as a reference. Please contact your local TI sales office or distributor for system design specifics and product availability.

Fig. 2.3.1. Flowchart of a portable blood gas analyzer

4. Measurement technology. The analyzer is a microprocessor tool that measures optical fluorescence from discrete sensors that represent optical electrodes. The disposable cassette contains everything you need to calibrate, measure, and collect waste. Specific calibration information is read into the analyzer using a barcode. The cartridge is then inserted into the measuring unit.

The analyzer heats the cartridge to 37.0 ± 0.1 ° C and performs a calibration test on the PCO2 and PO2 sensors by passing the calibration gas mixture along the sensors. The electrolyte channel and pH are calibrated using a buffer contained in the cassette. After calibration, the analyzer injects blood into the middle of the cartridge along the sensors. The intensity of the light beam is measured after equilibration with a blood sample.

5. Types of cartridges:

cassette B: pH, PCO2, tHb, SO2;

cassette E: pH, PCO2, tHb, SO2, Na +, K +;

cassette E-Ca: pH, PCO2, tHb, SO2, Na +, K +, Ca ++;

E-Cl cassette: pH, PCO2, tHb, SO2, Na +, K +, Cl–;

cassette E-Glu: pH, PCO2, tHb, SO2, Na +, K +, Glu.

6. Properties of cartridges. The cartridges are stored at room temperature, which eliminates possible storage and transportation problems. The cassette information and its calibration are stored in the cassette itself. Automatic aspiration of the sample, which does not allow the reagents to be used until the analyzer is used.

7. Performance of the blood gas analyzer:

Required blood volume, µl	125
Sample type	Whole blood, serum, plasma
Sampling	Automatic
Analysis time	<120 s
Operating temperature range, ° C	10 ... 32
Humidity, %	5 ... 95

8. Technical characteristics of the blood gas analyzer:

Measured parameters	*Measurement range*	*Resolution*
pH	6,6 ... 7,8	0,001
PCO_2	10 ... 200 mm. Hg Art.	0,1 mm. Hg Art.
PO_2	10 ... 700 mm. Hg Art.	0,1 mm. Hg Art.
Na+	100 ... 180 mmol / l	0,1 mmol / l
K+	0,8 ... 10 mmol / l	0,01 mmol / l
Cl–	50 ... 160 mmol / l	0,1 mmol / l
Ca++	0,2 ... 0,3 mmol / l	0,01 mmol / l
Glucose	30 ... 400 mg / dL	0,1 mg / dL
Glucose	1,7 ... 22 mmol / l	0,1 mmol / l
tHb	5 ... 25 g / dL	0,1 g / dL
SO_2	60 ... 100%	0,1%
Barometric pressure	300 ... 800 mm. Hg Art.	0,1 mm. Hg Art.

9. Calculated parameters of the blood gas analyzer:

Estimated parameters	*The range of values*
Actual hydrogen carbonate (HCO3)	1…200 mmol / l
Excess bases (BE)	–40…40 mmol / l
Additional cell fluid excess (BEecf)	–40…40 mmol / l
Actual Excess Grounds (BEacf)	–40…40 mmol / l
Buffer Base (BB)	0…100 mmol / l
Total CO2 (tCO2)	1…100 mmol / l
Standard Hydrocarbonate (st.HCO3)	2…200 mmol / l
Standard pH (st. PH)	6,5...8,0
Oxygen saturation (SO2)	0...100%
Oxygen content (O2ct)	0...56 mg / dL
Hematocrit (Hct)	15...75%
Concentration of hydrogen ions (CH +)	10...1000 mmol / l
Alveolar-arterial oxygen gradient (AaDO2)	0...800 mm. Hg Art.
Anion interval (AG)	3...30 mmol / l
Standardized Ionized Calcium (pH = 7.4) (NCa ++)	0,1...3,0 mmol / l
P_5O	15...35 mm. Hg Art.

10. Data management

The printer	built-in thermal
Screen	sensor
Contro;	By sensor screen
Interface,	RS-232, ACII or ASTM format
Memory	200 results
Quality control	30 days

11. Output parameters:

The temperature of the patient	14 ... 44°C
Hemoglobin	1 ... 26 g / dL
Type of hemoglobin	adult / fetal
The average concentration of red blood cells (ICS),%	29 ... 37
P_5O	15 ... 40 mm. Hg Art.
FIO_2	0,21 ... 1,0
RQ	0,7 ... 2,0
Gender of the patient	M / F
Operator ID	up to 10 digits
Patient ID	up to 10 digits

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010

(Fig. 2.1.2 and Fig. 2.1.3).

2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and functions of working with the program timer in the programming environment of microprocessors Code Composer Studio.
3. Document timer charts obtained in experiments for the report. To formulate in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 2.1.2 and Fig. 2.1.3. Initialize data and variables in a work project in the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
        #include <csl_irq.h>
        #include <csl_timer.h>
        /*Configure Structures */
        extern far TIMER_Config timerCfg0;
        extern far TIMER_Config timerCfg1;
        extern far TIMER_Handle hTimer0;
        extern far TIMER_Handle hTimer1;
        TIMER_Config timerCfg0 =
                { 0x00000000,     /*  Control Register (CTL)  */
                0x00000000,       /*  Period Register (PRD)   */
                0x00000000        /*  Counter Register (CNT)  */
        };
        TIMER_Config timerCfg1 =
                { 0x00000000,     /*  Control Register (CTL)  */
                0x0000FFFF,       /*  Period Register (PRD)   */
                0x00000000        /*  Counter Register (CNT)  */
        };
```

```
void swiFxn0(void);
void swiFxn1(void);
void ConfigureAllTimers(void);
```

2. To use methods of work with the program timer in the programming environment of microprocessors Code Composer Studio:

```
// Timer setting functions
        void swiFxn0(Void)
        {
                SWI_post(&SWI0);
        }
        void swiFxn1(Void)
        {
                SWI_post(&SWI1);
        }
        /* Configure All Timers */
        void ConfigureAllTimers(void)
        {
                /*  Handles  */
                TIMER_Handle hTimer0;
                TIMER_Handle hTimer1;
                /*  Timers  */
                hTimer0 = TIMER_open(TIMER_DEV0,TIMER_OPEN_RESET);
                TIMER_configArgs(hTimer0, 0x000002c0, 0x01000, 0x00000000);
                hTimer1 = TIMER_open(TIMER_DEV1,TIMER_OPEN_RESET);
                TIMER_configArgs(hTimer1, 0x000002c0, 0x00100, 0x00000000);
        }
```

3. To implement the procedure of accumulation of timer counts in the programming environment of microprocessors Code Composer Studio:

```
// Using a software timer
/* Run Timer */
```

```
ConfigureAllTimers();
/* SWI test Started */
SWI_post(&SWI0);
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

   Project (PJT) - example.pjt.

   Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. Explain the purpose of the CSL_Timer.h software library?
2. How are the NI Programming Function VI virtual interface components used?
3. What is the function of working with the program timer?

2.4. Computed tomograph

The purpose of the job is to design a microprocessor-controlled digital device in LabVIEW and to study the DMA mode serial port access function in Code Composer Studio microprocessor programming environment.

Theoretical information

Computed tomography (CT) is a medical x-ray diagnostic apparatus designed to reproduce a three-dimensional image of the internal organs of a person according to a large series of two-dimensional x-ray images taken around one axis of rotation. Compared to conventional radiographs, CT images of the exhibits have much better informativeness, clarity and contrast.

The block diagram of a computer tomograph is presented in Fig. 2.4.1.

Computed tomography consists of three main parts:

1. Gantry, which houses an X-ray tube and a ring of detectors. The tube, rotating around the patient, emits X-rays, which are perceived by detectors located around the circumference of the ring.

2. Multichannel processing unit consisting of multiplexers (AFE) and ADC (ADC). The inputs of the multiplexers receive information from detectors located in Gantry. In the ADC, the analog signal is converted to digital form.

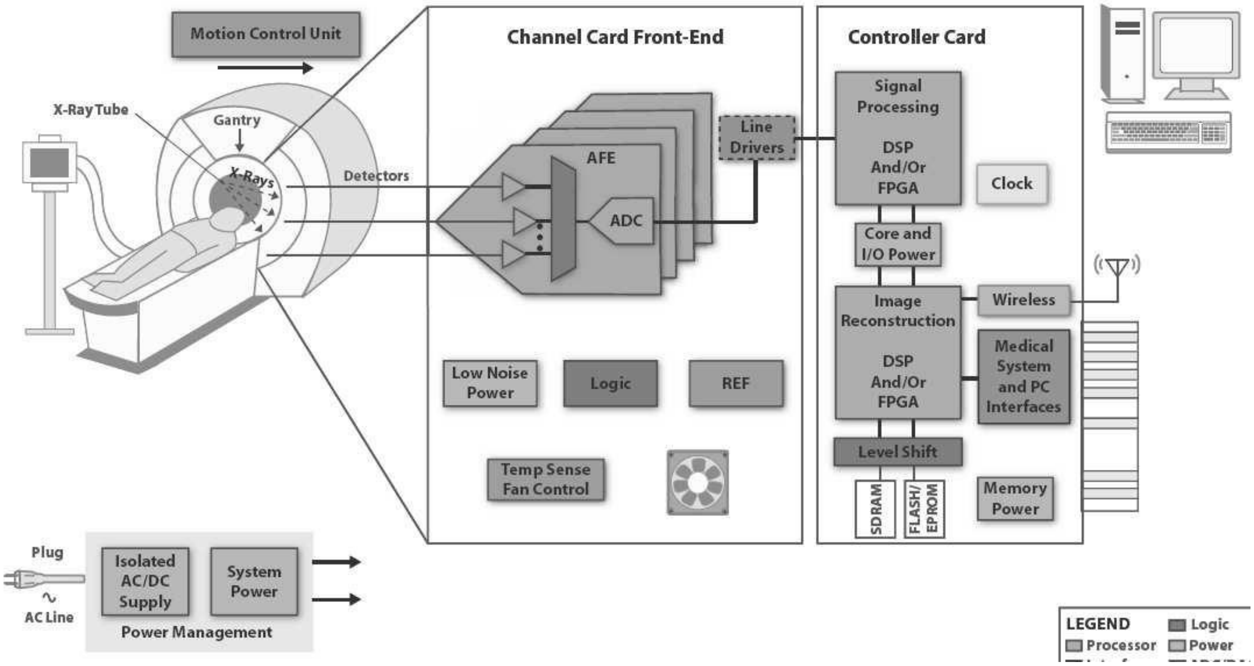


Fig. 2.4.1. Flowchart of a computer tomograph

3. The processor block consists of two DSP processors, one of which receives digital information from multiplexers (AFE) and the other performs image reconstruction in three-dimensional form.

To reproduce the image of human organs in the tomograph interfaces are implemented to connect a personal computer or any medical system. Patient information is stored in flash memory.

Spiral computed tomography.

Spiral scanning involves the simultaneous execution of two continuous actions: rotation of the X-ray source - an X-ray tube that generates radiation around the body of the patient, and translational motion of the table with the patient along the longitudinal axis of the scan through the aperture of the gentry. In this case, the trajectory of motion of the x-ray tube relative to the axis of direction of movement of the table with the patient's body is in the form of a spiral.

In contrast to sequential CT, the speed of movement of the table with the patient's body may take arbitrary values determined by the goals of the study. The higher the speed of the table, the greater the length of the scanning area. It is important that the length of the path that passes the table with the patient for one revolution of the x-ray tube can be 1.5 - 2 times the thickness of the tomographic layer without impairing the spatial resolution of the x-ray image.

Spiral scanning technology can significantly reduce the time spent on CT scans and significantly reduce the radiation load on the patient.

Multilayer Computed Tomography (MSCT).

Multilayer ("multispiral", "multisection" computed tomography - MSCT) was first introduced by ElscintCo. The fundamental difference between the MSCT tomographs and the spiral tomographs is that there are not more than one or more rows of detectors in the Henry circle. In order for the X-ray radiation to be simultaneously received by detectors located on different rows, a new bulk geometric shape of the beam is used. In addition, the number of rotations of the x-ray tube was increased from one to two per second. Thus, the four-spiral fifth-generation CT scanner is eight times faster than conventional fourth-generation CT scanner. Today, some modern clinics already use a 320-computed CT scanner. These tomographs were first introduced by Toshiba, which is the new revolution in the evolution of x-ray computed tomography. Not only do they capture images, they also allow the physiological processes that occur in the brain and heart to be observed in near real time. The peculiarity of such a system is the ability to scan the whole organ (heart, joints, brain, etc.) for one revolution of the x-ray tube, which significantly shortens the examination time, and also allows the scan of the heart even in patients with arrhythmias.

The advantages of MSCT over conventional spiral CT are the following:

- the improvement of temporal resolution is achieved by reducing the time of the study and the number of artifacts caused by involuntary movements of the internal organs and pulsations of large vessels;

- improving the spatial resolution along the longitudinal axis is due to the use of thin (1 ... 1.5 mm) and very thin (0.5 mm) sections;

- an increase in scanning speed is twice achieved by reducing the time of rotation of the x-ray tube, compared to a conventional spiral CT scan to 0.45 ... 0.50 s.

All the above-mentioned innovations in MSCT not only increase the spatial resolution, but thanks to specially developed image reconstruction algorithms can significantly reduce the number and size of artifacts (extraneous elements) in the CT image. The main advantage of MSCT compared to single-section SCT is the ability to obtain an isotropic image when scanning with a submillimeter cut thickness (0.5 mm). An isotropic image can be obtained if the boundaries of the voxel of the image matrix are equal, that is, the voxel takes the form of a cube. In this case, the spatial resolution in the transverse plane and along the longitudinal axis becomes the same.

Computed tomography with two sources of radiation.

DSCT (Dual Source Computed Tomography) is one of the logical continuations of MSCT technology. The fact is that in the study of the heart (CT coronary angiography) it is necessary to obtain images of objects that are in constant and fast motion, which requires a very short scan period. In MSCT, this was achieved by ECG synchronization and conventional X-ray tube rotation. But the minimum time required to record a fixed slice for MSCT at a tube rotation time of 0.33 s (or ≈ 3 rpm) is 173 ms, that is, the tube half-turn time. Such a temporal resolution is quite sufficient for the normal heart rate, and clinical studies have shown the effectiveness of MSCT at frequencies less than 65 beats per minute. and about 80 beats per minute.

The use of two x-ray tubes at an angle of 90 ° gives a temporal resolution equal to a quarter of the tube's rotation period (83 ms at a rotation time of 0.33 s). This allows you to get a heart image regardless of the frequency of contractions.

Also, this device has another significant advantage: each tube can operate in its own mode (at different values of kV voltage and mA current, respectively). This allows better differentiation of closely spaced objects of different density in the image. This is especially important when contrasting vessels and formations that are close to the bones or implanted metal structures. This effect is based on different absorption of radiation when changing its parameters in a mixture of blood with iodine-containing contrast agents with the invariance of this parameter in materials from hydroxyapatite (bone base) or metals. On the other hand, DSCTs are conventional MSCTs and have all their advantages.

Contrast enhancement.

To improve the differentiation of organs from each other, as well as normal and pathological structures, different methods of contrast enhancement are used (most often with the use of iodine-containing contrast agents).

The two main types of contrast drug administration are oral (a patient with a certain regimen drinks the drug solution) and intravenous (performed by medical staff). The main purpose of the first method is to contrast the hollow organs of the gastrointestinal tract; the second method allows to evaluate the nature of the accumulation of the contrast drug by tissues and organs through the circulatory system. In many cases, the techniques of intravenous contrast enhancement make it possible to clarify the nature of the pathological changes detected (including accurately indicating the presence of tumors) on the background of the surrounding soft tissues, as well as to visualize changes that are not detected in conventional CT examination.

Intravenous contrast can be performed in two ways: "manual" intravenous contrast and injection contrast. In the first method, the contrast is manually entered by a medical technician, the time and speed of administration are not regulated, the study begins after the introduction of the contrast agent. This method is used in studies on first-generation devices, but with MSCT "manual" administration of contrast agent is no longer consistent with the greatly increased capabilities of the method. When injecting contrast enhancement, the contrast agent is injected intravenously with a syringe injector with a set speed and time of substance delivery. The purpose of such contrast enhancement is to differentiate the contrast phases. Scan times vary across devices, at different contrast drug delivery rates, and across patients. On average, at a rate of administration of the drug 4-5 ml / s scan begins approximately 20 ... 30 seconds after the start of injection with contrast injector, while filling the arteries (arterial phase of contrast). After 40 ... 60 seconds, the device re-scans the same area for allocation of portal-venous phase, in which the contrast of veins is visualized. Also distinguish a delayed phase (180 s after the start of administration), in which the excretion of the contrast agent through the patient's urinary system is observed.

CT angiography and perfusion.

CT angiography allows to obtain a layered series of images of blood vessels; 3D-based computer-generated data builds a three-dimensional model of the circulatory system. Spiral CT angiography is one of the latest advances in x-ray computed tomography. The study is conducted on an outpatient basis. In the elbow vein is injected iodine-containing contrast drug in a volume of ~ 100 ml. At the time of introduction of the contrast drug make a series of x-ray scans of the study area.

CT perfusion. The method that allows to estimate the passage of blood through the tissues of the body, in particular is performed: brain perfusion; liver perfusion.

Work task

1. To collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.1.2 and Fig. 2.1.3).
2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and functions of working with the program timer in the programming environment of microprocessors Code Composer Studio.
3. Document the methods used to access the serial port in DMA mode for the report. To formulate in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 2.1.2 and Fig. 2.1.3. Initialize data and variables in a work project in the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
        #include <csl.h>
        #include <csl_dma.h>
        #include <csl_mcbsp.h>
        extern far MCBSP_Config hMcbsp1;
        /*  Configure Structures */
        MCBSP_Config mcbspCfg1 =
        {
                0x28000000, // opt
```

```
                        0x01900000, // src: DRR 1
                        0x00000001, // cnt
                        0x01900004, // dst: DXR 1
                        0x00000000, // idx
                        0x00000000 // rld
                };
                /*  Handles  */
                MCBSP_Handle hMcbsp;
                DMA_Handle hDma1;
```

2. Perform programmatic configuration of serial port access in DMA mode in Code Composer Studio microprocessor programming environment:

```
// Configure DMA access
    /* Once the serial port is initialized */
                void HostTargetComm(void)
                {
                        DMA_reset(INV);
                        hDma1 = DMA_open(DMA_CHA2,DMA_OPEN_RESET); // Ch. CHA2
                        DMA_configArgs(
                                hDma1, //Channel
                                0x0A000110u, //Primary Control Register (Peripherals pp4-9)
                                0x0000000Au, //Secondary Control Register
                                 (unsigned int) intermediate, //Source Address
                                0x01710000u, //Destination Address
                                0x00010080u); //Transfer Counter Register
                        DMA_start(hDma1);
                }
```

3. Implement the procedure for activating serial port access in DMA mode in the Code Composer Studio microprocessor programming environment:

```
// DMA serial access
  /* Initialize the library */
```

```
CSL_init();
hMcbsp = MCBSP_open(MCBSP_DEV1,MCBSP_OPEN_RESET);
MCBSP_config(hMcbsp,&mcbspCfg1);
HostTargetComm();
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. Explain the purpose of the CSL_DMA.h software library?
2. How are the NI Programming Function VI virtual interface components used?
3. What is the function of serial port operation?

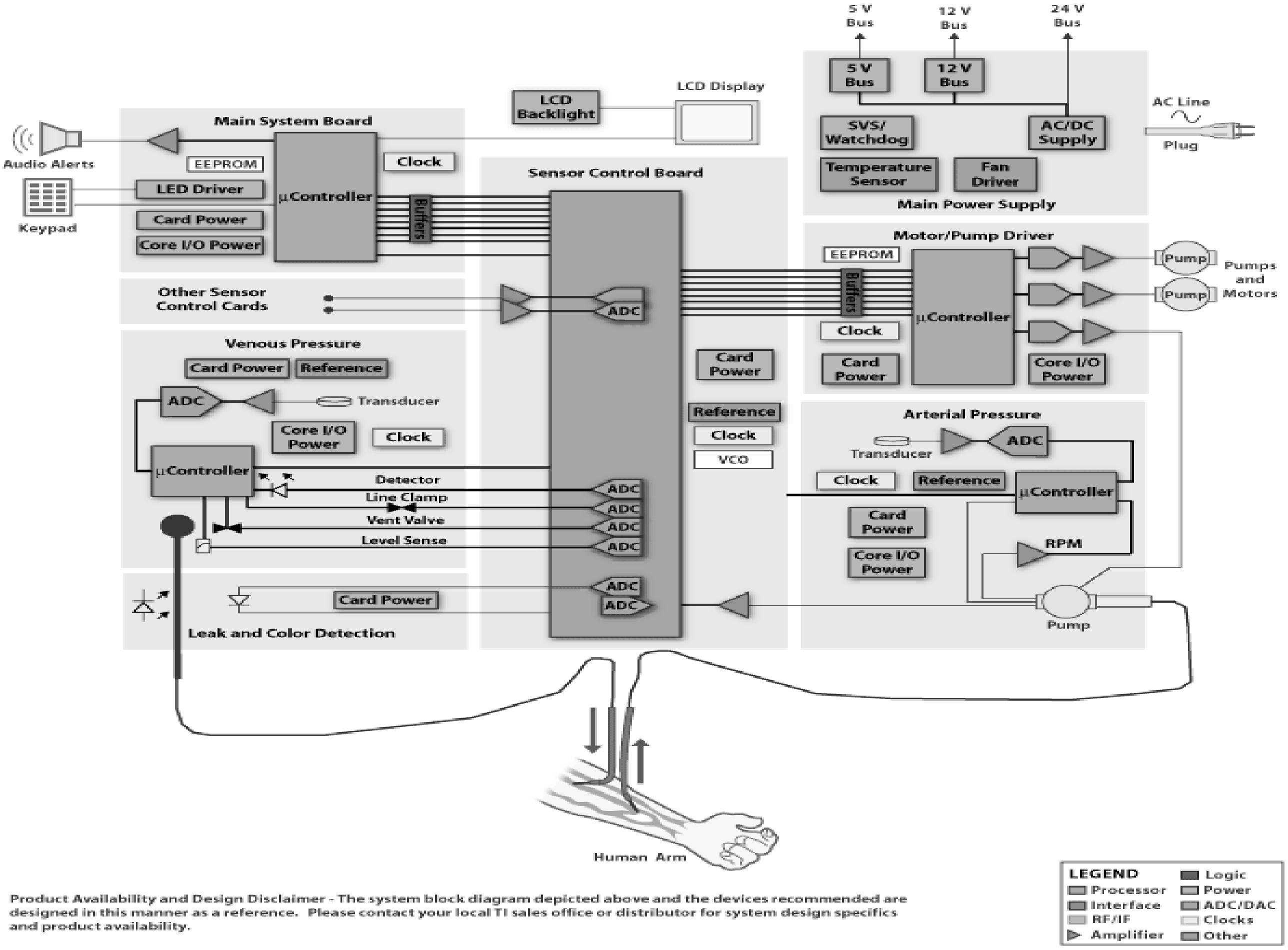
2.5. Dialysis equipment or artificial kidney

The purpose of the work is to design a digital instrument in LabVIEW with microprocessor control and to study the function of discrete Fourier transform in the microprocessor programming environment Code Composer Studio.

Theoretical information

In the device "artificial kidney" can be distinguished the following major nodes: dialyzer with bloodstream, blood pump, block electronic control of the parameters of blood and dialysis, the block of preparation and discharge dialysate.

The block diagram of the "artificial kidney" is presented in Fig. 2.5.1.



Product Availability and Design Disclaimer - The system block diagram depicted above and the devices recommended are designed in this manner as a reference. Please contact your local TI sales office or distributor for system design specifics and product availability.

Fig. 2.5.1. Artificial kidney block diagram

The flowchart includes four microcontroller-controlled units, a blood and color dialyser control unit, and an ADC-based Sensor Control Board:

- The Main System Board is designed for general control, keyboard input and alarm output.

- The blood pressure unit (Arterial Pressure) provides control of the blood pump, sets the speed depending on the results of blood pressure measurements obtained from the transducer (Transducer) and switches off the pump by alarm.

- Venous Pressure Unit provides measurement of venous pressure, control of venous clamp and drainage valve.

- Leakand Colordetection flow and color unit for electronic control of blood and dialysate parameters.

One of the main components of the apparatus "artificial kidney" is the dialyzer. It is intended for extracellular blood purification. Dialysis is a "selective diffusion" - the movement of substances from a high concentration to a lower one through a semipermeable membrane. This process is based on the properties of semipermeable membranes to pass particles and ions up to 500 A in size and to retain colloidal particles and macromolecules. Two solutions work in this process - dialysable (blood) and dialysate. There are single and reusable dialyzers. To date, a large number of modifications of dialysers have been created: plate, capillary, coil, spiral, etc. The dialyzer consists of three to four support plates and two sections, which form "dialysate cavities" and "blood cavities". Dialysate and blood cavities are separated from each other by translucent membranes. Dialysate is leaking into the cavity of dialysate, and blood flows through a thin layer of blood. Thus, dialysate washes the blood cavity.

The next node of the "artificial kidney" is a blood pump. It is used to pump blood through the arterial line into the dialyzer and to return it after purification into the patient's bloodstream through the vein.

A very important unit of the device is the electronic control unit for blood and dialysate parameters. It helps to control the integrity of the membrane. Even with its micro-breakthrough (micro-damage to the membrane) and when it enters the dialysate of a small amount of blood, electronic sensors are automatically triggered, the blood pump stops and the dialysate stops. Blood pressure in the apparatus-patient system and blood flow rate is also monitored.

The dialysate temperature, the concentration of the solution, the speed of the dialysate movement, the magnitude of the negative pressure in the dialyzer are also monitored. In the event of any system malfunction or deviation from the set parameters, an alarm and a light alarm sound immediately. Then hemodialysis is stopped until the defect is detected.

Work task

1. To collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.1.2 and Fig. 2.1.3).
2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and functions of working with the program timer in the programming environment of microprocessors Code Composer Studio.
3. Document the report on the use of the discrete Fourier transform function. To formulate in the

report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 2.1.2 and Fig. 2.1.3. Initialize data and variables in a work project in the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
    #include <math.h>
    struct COMPLEX
    {
        short real;
        short imag;
    };
    int i, j, k;
    int mx[128];
    struct COMPLEX x[128];
```

2. Prepare data that has a valid and imaginary part for applying a discrete Fourier transform in Code Composer Studio microprocessor programming environment:

```
// Initialize the data array
/* Change Input to k = N */
    k = 32;
    for(i=0;i<k;i++)
    {
        x[i].real=0x7FFF * (Byte)(i);
        x[i].imag=0;
    }
    DFT(k, x);
    for(j=0;j<k;j++)
    {
        mx[j] = (x[j].real*x[j].real + x[j].imag*x[j].imag);
```

```
                mx[j] = mx[j] << 1;
        }
```

3. Implement the discrete Fourier transform function in the Code Composer Studio microprocessor programming environment:

```
// Discrete Fourier transform function
        void DFT(int nx, struct COMPLEX fx[restrict])
        {
        int n, k;
        double arg;
        int Xr[1024];
        int Xi[1024];
        short Wr, Wi;
        const float PI = 3.14152;
        for(k=0; k<nx; k++){
                Xr[k] = 0;
                Xi[k] = 0;
                for(n=0; n<nx; n++){
                        arg =(2*PI*k*n)/nx;
                        Wr = (short)((double)32767.0 * cos(arg));
                        Wi = (short)((double)32767.0 * sin(arg));
                        Xr[k] = Xr[k] + fx[n].real * Wr + fx[n].imag * Wi;
                        Xi[k] = Xi[k] + fx[n].imag * Wr - fx[n].real * Wi;
                }
        }
        for (k=0;k<nx;k++){
                fx[k].real = (short)(Xr[k]>>15);
                fx[k].imag = (short)(Xi[k]>>15);
        }
}
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code

Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. Explain the purpose of the MATH.h software library?
2. How are the NI Programming Function VI virtual interface components used?
3. What is the function of a discrete Fourier transform?

2.6. CPAP positive airway pressure device

The purpose of the work is to design a digital device in LabVIEW with microprocessor control and to study mathematical functions embedded in the digital signal processor in Code Composer Studio microprocessor programming environment.

Theoretical information

A CPAP (Constant Positive Airway Pressure) device or constant positive airway pressure provides respiratory ventilation, which is used mainly for the treatment of apnea. During sleep, muscles tend to naturally relax, causing the upper respiratory tract to narrow. This reduces the amount of oxygen in the blood and causes awakening from sleep.

The block diagram of the CPAP device is presented in Fig. 2.6.1.

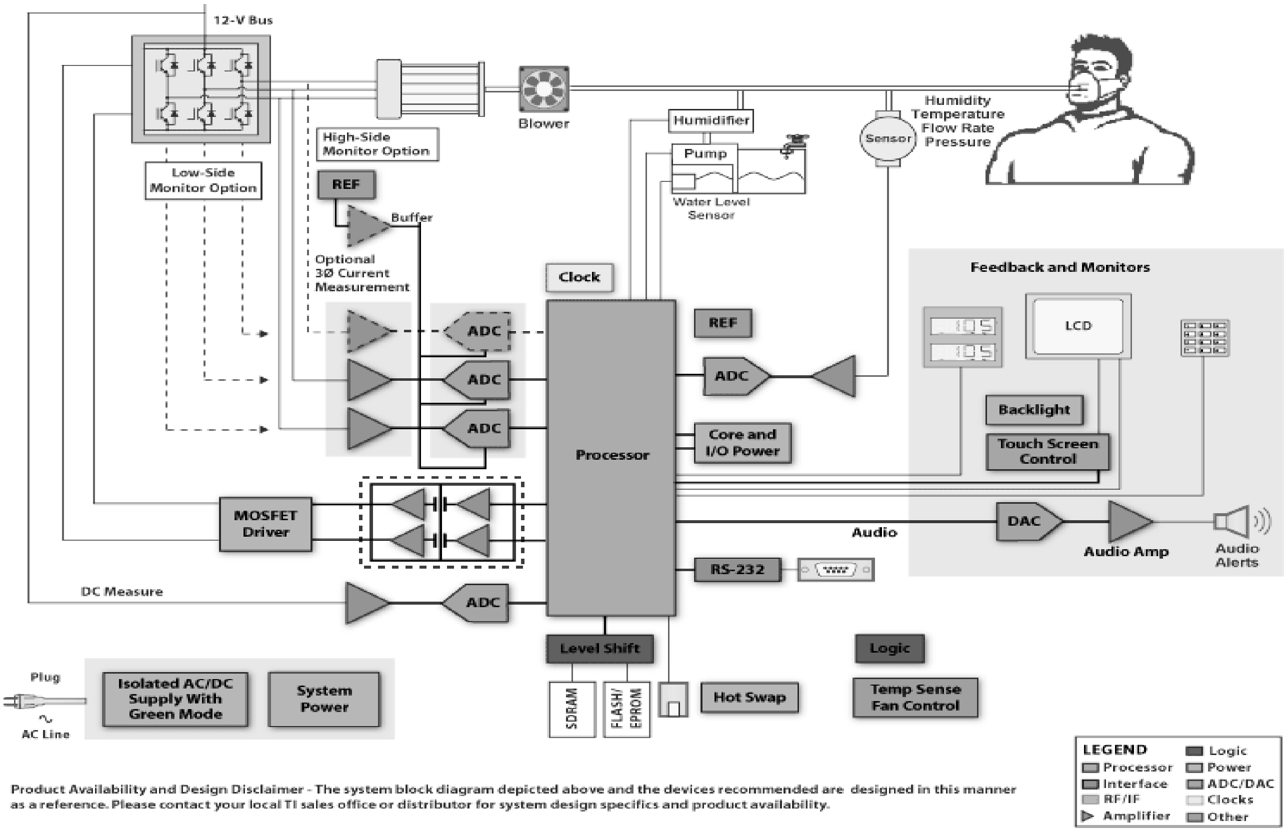


Fig. 2.6.1. Flowchart of CPAP device

The device has a sensor that real-time monitors the pressure in the respiratory circuit (in the tube), as well as low inertial low-noise engine. The microprocessor in the CPAP device performs several operations, including sampling from sensors of pressure, humidity, air flow, air flow temperature, as well as the calculation of the desired parameters: pressure, humidity, temperature in the respiratory tract and the formation of engine control signals, humidifier and heater.

When the pressure drops (on inhalation) the device accelerates the engine speed and maintains the proper medical pressure. As the pressure increases (exhale), the machine slows the engine speed, which also ensures pressure stability. In the event of an air leak from under the mask, the device determines the pressure drop, accelerates the engine speed and compensates for the outflow. Engine control accuracy is ensured by a level control system for all three power phases that maintains the required engine speed.

The device automatically adjusts the medical pressure depending on the presence or absence of apnea and snoring. In practice, it looks like this. The initial treatment pressure and its range are set. For example, the output pressure is 10 mm water column, the range 6 ... 14 mm water column. The unit starts operating at output pressure. If the patient does not have apnea or snoring, the device gradually lowers the pressure to the lower limit of the range. If these symptoms occur, the apparatus gradually increases the pressure until they disappear or to the upper limit of the range.

The feasibility of adjusting the pressure in real time is also due to the change in therapeutic pressure depending on the position of the body and stage of sleep. In deep sleep and in the back, much more pressure is required to open the airways compared to surface and side sleep, respectively.

An important element of the control system is the MOSFET drivers, which reliably and smoothly control the operation of the power switches of the DC motor phase switch. The drivers use high operating frequencies to provide a form of voltage conversion to the motor close to perfect. Drivers allow you to control the value of the voltage drop on the MOSFET transistors to smoothly switch off the transistor when the dangerous threshold is exceeded, to block for a predetermined time the circuit of protection at the active state of the transistor at the moments of switching.

Due to the high power of the built-in isolated AC / DC converters, drivers determine the extremely short switching time of the controlled transistors, which minimizes the dynamic losses in the converters. The drivers form the necessary "dead" time between the switching of the transistors of the upper and lower arms of the inverter. Such drivers are also available as separate monolithic modules.

Modern CPAP devices are compact (2-3 kg), low noise, allow to filter, humidify and heat the air, provide accurate dosing of pressure and compensate for possible leaks. In addition, most devices provide a function of gradually increasing the pressure to the working level within 5 - 45 minutes, which facilitates the process of falling asleep to the patient.

The modern CPAP machine records the following data:

- duration of use of the device during the night;
- parameters of medical pressure (minimum, average, maximum);
- graph of percentage distribution of different levels of medical pressure;
- hourly schedule of leaks and average pressure during the night;
- the presence of episodes of apnea, episodes of apnea with cardiac oscillations, hypopnoe, episodes of hypopnoe with flowlimation, snoring, flow limitation;

- percent distribution of respiratory cycles (normal, intermediate, flow cycles, false).

In addition, over the last 35 hours of treatment, it is possible to view detailed data in the form of graphs of the dynamics of medical pressure, air leakage from the mask and respiratory disorders in the patient. Registration of treatment parameters provides objective control over the patient's condition and the effectiveness of treatment.

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.1.2 and Fig. 2.1.3).
2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and functions of working with the program timer in the programming environment of microprocessors Code Composer Studio.
3. Document the mathematical functions of the digital signal processor built into the digital signal processor. To formulate in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions..

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 2.1.2 and Fig. 2.1.3. Initialize data and variables in a work project in the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
        #include <math.h>
        #define N 10
        int i, nw;
        short int *y[N], *w[N], *samples[N];
```

2. Prepare data that has a real and imaginary part for the application of the built-in mathematical function in the Code Composer Studio microprocessor programming environment:

```
// Initialize the data array
        /* Function generator */
        nw = N;
```

```
for(i = 0; i <= N; i++ ) {
        *y[i] = 0;
        *w[i] = i;
        *samples[i] = N*sin(i)/cos(N);
}
```

3. Use the integrated Fourier Mathematical Function of the TMS320C5505 Specialized Digital Signal Processor (DSP) Library in Code Composer Studio microprocessor programming environment:

```
// Built-in discrete Fourier transform function
        /*  void DSP_fft(const short *w, int nsamp, short *x, short *y);        */
        /*  nsamp = length of DSP_fft in complex samples                        */
        /*  x     = pointer to complex data input, time domain                  */
        /*  w     = pointer to complex twiddle factors                          */
        /*  y     = pointer to complex data output, frequency domain            */
        DSP_fft (*w, nw, *samples, *y);
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device..

Control questions

1. Explain the purpose of mathematical functions built into the microprocessor?
2. How are the NI Programming Function VI virtual interface components used?
3. What are the features of the fast discrete Fourier transform function?

2.7. Ddevice for artificial ventilation of the lungs

The purpose of the work is to design a digital device in LabVIEW with microprocessor control and to study the function of reading and writing data to RAM in the programming environment of microprocessors Code Composer Studio.

Theoretical information

The ventilator (IVL) is used for intensive care and resuscitation to increase the frequency and depth of breathing during respiratory failure by forming and regulating the flow of gas into the lungs. It is often necessary to first treat a patient with forced ventilation and then slowly recover the patient from forced ventilation in spontaneous ventilation mode. The mandatory mode of operation of the ventilator is the control of all aspects of breathing, such as respiratory volume, respiratory rate, air mixture pressure and oxygen concentration during breathing. In spontaneous mode, the ventilator must allow the patient to start breathing independently and to control breathing rate, flow rate and respiratory volume. In the short-term treatment of acute respiratory problems and in long-term therapy, the same concepts and components are used to treat patients with chronic respiratory diseases.

The block diagram of the IVL apparatus is presented in Fig. 2.7.1.

Pressure sensors play an important role in ensuring the normal operation of the equipment by converting physical quantities, such as airway pressure, which is converted into a differential signal. Accurate processing of these signals is very critical to the patient's life. Air and oxygen flow sensors generate signals used by the microprocessor control system, control valves to deliver the desired volume of air and oxygen flows. The airway pressure sensor generates the feedback signal necessary to maintain the desired positive end-of-breath pressure (PTK). Often the sensors themselves are characterized by a very large displacement and drift of characteristics.

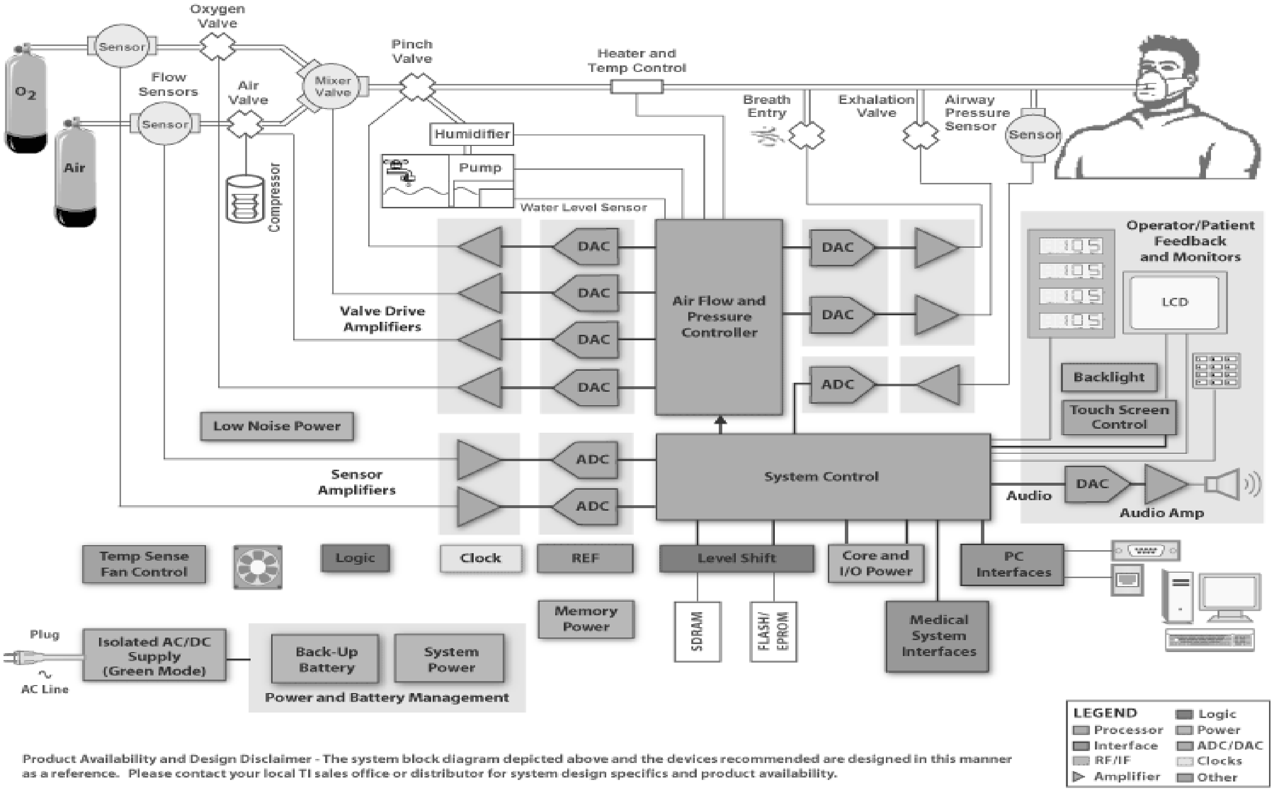


Fig. 2.7.1. Block diagram of the apparatus of artificial ventilation of the lungs

In this scheme, the microprocessor performs several operations, including sampling of air and oxygen pressure, calculation of the required airway pressure and total air flow rate on the inhalation, actuation of the air and oxygen valves for each individual inhalation cycle. These real-time operations do not require the use of high-speed, low-power, and high-signal-microprocessor integration.

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.1.2 and Fig. 2.1.3).
2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and functions of working with the program timer in the programming environment of microprocessors Code Composer Studio.
3. Document the report on the use of read and write data in RAM. To formulate in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 1.1.2 and Fig. 1.1.3. Initialize data and variables in a work project in the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
        #include <mem.h>
        #include <que.h>
        #include <sys.h>
        #define NUMMSGS     5       /* Number of messages */
        typedef struct MsgObj {
                QUE_Elem    elem;     /* First field for QUE */
                Char        val;          /* Message value */
        } MsgObj, *Msg;
        void reader();
        void writer();
```

2. Using the functions of recording and reading data from RAM in the programming environment of microprocessors Code Composer Studio:

```
// Initialize the data array
/*  The writer() must be called before reader() to ensure that the
        *  queue is non-empty for the reader.  */
        writer();
        reader();
```

3. Implement the functions of reading and writing data to the memory in the programming environment of microprocessors Code Composer Studio:

```
// Read and write functions to memory
void reader()
{
        Msg         msg;
        int         i;
        for (i=0; i < NUMMSGS; i++) {
```

```
                /* The queue should never be empty */
                if (QUE_empty(&queue)) {
                        SYS_abort("queue error\n");
                }
                /* Dequeue message */
                msg = QUE_get(&queue);
                /* Free msg */
                MEM_free(0, msg, sizeof(MsgObj));
        }
}
void writer()
{
        Msg         msg;
        int         i;
        for (i=0; i < NUMMSGS; i++) {
                /* Allocate msg */
                msg = MEM_alloc(0, sizeof(MsgObj), 0);
                if (msg == MEM_ILLEGAL) {
                SYS_abort("Memory allocation failed!\n");
                }
                /* Fill in value */
                msg->val = i + 'a';
                /* Enquire message */
                QUE_put(&queue, msg);
        }
}
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. Explain the purpose of the MEM.h software library?
2. How are the NI Programming Function VI virtual interface components used?
3. What is the function of reading and writing data to RAM?

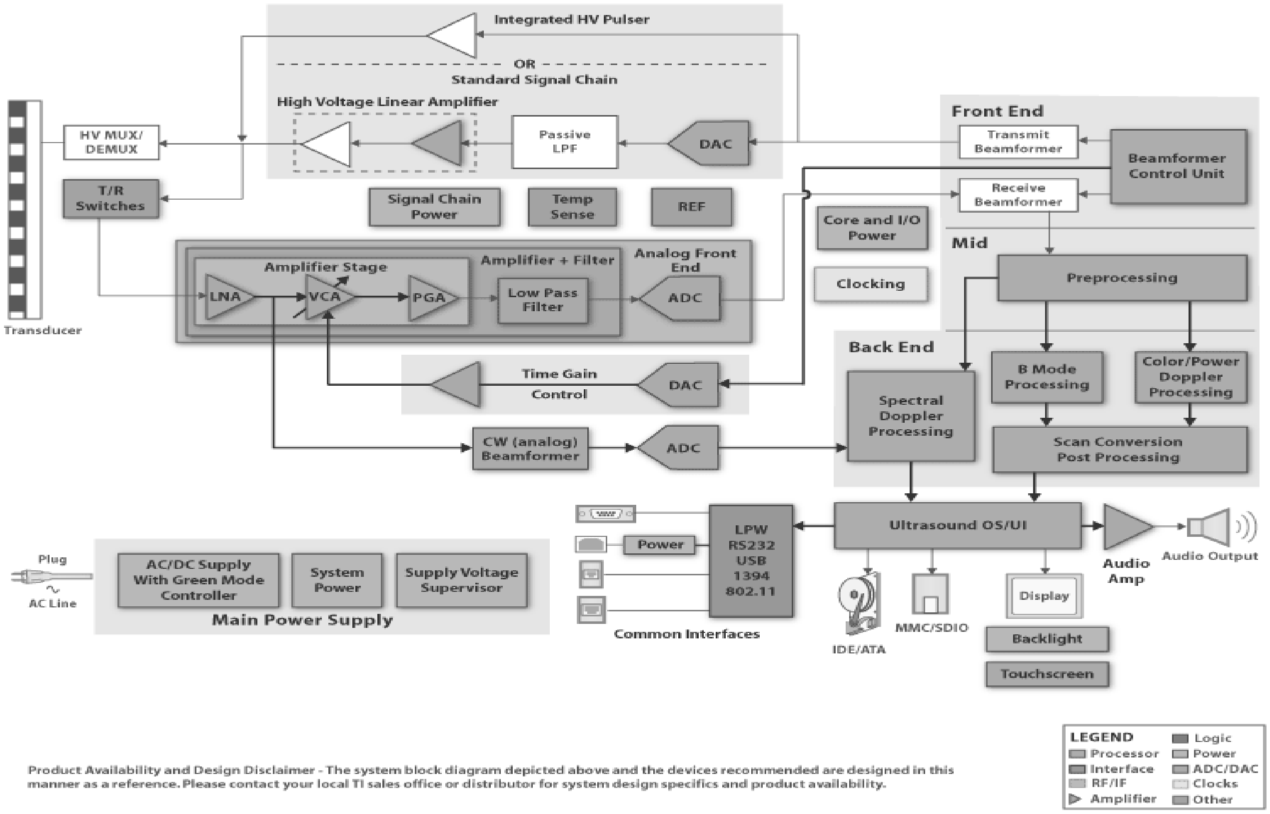
2.8. Ultrasonic systems

The purpose of the work is to design a digital device in LabVIEW with microprocessor control and to study software interrupts IRQ in Code Composer Studio microprocessor programming environment.

Theoretical information

Ultrasound systems, both medical and industrial, use focussed imaging techniques to achieve image quality far beyond what can be achieved using a single channel. Using an array of receivers, you can build a high-definition image with time shifting, scaling and changing the signal energy. The concepts of time shift and scaling are based on the reception of signals by an array of sensors, which makes it possible to focus the acoustic beam at a single point in the scanning area. In the future, as a result of receiving a reflected signal focused from different points, the image of the area irradiated by ultrasound is collected.

The block diagram of the ultrasonic scanner is presented in Fig. 2.8.1.



Product Availability and Design Disclaimer – The system block diagram depicted above and the devices recommended are designed in this manner as a reference. Please contact your local TI sales office or distributor for system design specifics and product availability.

Fig. 2.8.1. Flow chart of a modern ultrasound scanner

When scanning, simultaneously pulses from all sensor elements (from 8 to 512 sensors) are emitted. Impulses "illuminate" certain areas of the human body. After transmission, the sensors are switched to receive mode in parallel. The acoustic pulse in the form of a sound wave propagates through the body, usually in the range 1 ... 15 MHz. The signal decreases rapidly and decreases in proportion to the square of the distance. As the signal passes through areas with different acoustic characteristics of the human body, the energy of the wave front is reflected. The echo signal is reflected and detected by electronic means of the device. The signals reflected close to the surface will be very strong, while the signals reflected from the depths of the body will be very weak.

Due to the limitation on the power of ultrasonic energy that can be sent to the human body, the industry is developing extremely sensitive electronic receivers. Obtaining an informative signal from a focal point located close to the surface requires a slight amplification of the signal. This area is also called the near zone. At the same time, it is extremely difficult to receive signals from focal points located deep in the body, and such signals must be amplified a thousand times or more. This region is called the distant zone. In maximum gain mode, there is a system performance limit that depends on the noise sources in the receive channel. The main sources of noise are the converter-cable assembly and the low-noise receiver-amplifier. In the minimum gain mode, the performance limit is determined by the input signal.

A low pass filter is required between the amplifier and the ADC to smooth, filter and limit bandwidth noise. For this purpose, two-band or five-band filters are often used. Analog-to-digital converters (ADCs) typically have 10- and 12-bit bits. When processing digital data coming from the ADC, the number of interface lines can be reduced from 6144 to 1024 for a 512-channel system. This reduction is made on small and inexpensive microprocessor boards.

DSP processors are used to increase functionality and productivity, as well as for Doppler processing, 3D and even 4D imaging, as well as for implementation of various post-processing algorithms. The main requirements for imaging systems are high performance and high throughput. To provide intensive real-time processing of ultrasound information, processors operating at 1 GHz or higher are required, and communication with peripherals must be provided at 10 Gbit / s and full duplex bandwidth should be implemented.

Modern ultrasound scanners have very wide possibilities:

1. Plot scan modes: 2D gray scale, M-mode, color M-mode, color Doppler mapping, power Doppler, pulse-wave and constant-wave Doppler.

2. Extended frequency function:

- 4-5 frequencies in one sensor in 2D and color Doppler modes;
- Independent frequency selection in 2D and color Doppler modes;

3. Coherent image formation in 2D mode:

- 512 digital electronic channels;
- 57 344 coherent transceiver channels;
- multiple beamformers;
- storing and processing information about the phase and amplitude of the ultrasonic signal.

4. Dynamic range:

- clinically significant display range: 20 ... 100 dB with 1 dB adjustment step;
- dynamic system range over 160dB.

5. An ultrasonic workstation integrated into the system. DICOM 3.0 output, standard Ethernet / Internet output.

6. Built-in unit for loading and pharmacological stress echocardiography studies:

- the ability to install optimization programs and research protocols;
- dynamic recording and viewing of images in real time;
- capture the entire image field or selected area of interest;
- JPEG compression.

7. Complete package of cardiac calculations and measurements:

- full package of standard and additional fetometric parameters;
- a program for exploring twins with parallel display of measurement data and calculations;
- calculation of amniotic fluid index.

8. Package of vascular calculations:

- NTHI ™ native tissue harmonic technology;
- the use of harmonic components of echoes from tissues with contrast agents and without them;
- hardware and software for the operation of the internal heart sensor.

Work task

1. To collect the studied scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.1.2 and Fig. 2.1.3).
2. Explore the functionality of LabVIEW 2010 components for the NI Programming Function VI virtual interface. Describe the purpose and functions of working with the program timer in the programming environment of microprocessors Code Composer Studio.
3. Document a report on the use of IRQ program interrupts in real time. To formulate in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 2.1.2 and Fig. 2.1.3. Initialize data and variables in a work project in

the Code Composer Studio microprocessor programming environment:

```
// Initialize data and variables
        #include <csl_irq.h>
        #include <csl.h>
        #include <tsk.h>
        #include <clk.h>
        void task (Arg id_arg);
        void hi_pri_task (Arg id_arg);
```

2. Using real-time IRQ software interrupts in Code Composer Studio microprocessor programming environment:

```
// Initialize IRQ interrupts
        CSL_init();      // Initialize the library csl.h
        IRQ_globalEnable();          // Enable IRQ
        IRQ_enable(IRQ_EVT_RINT1);
        IRQ_globalDisable();         // Disable IRQ
```

3. Implement real-time IRQ program interrupts in Code Composer Studio microprocessor programming environment:

```
// Using IRQ program interrupts
void hi_pri_task (Arg id_arg)
{
        Int id = ArgToInt(id_arg);
        while (1) {
                SEM_pend(&sem, SYS_FOREVER);
        }
        IRQ_nmiEnable();
}
void task(Arg id_arg)
{
        int id = ArgToInt(id_arg);
        LgUns time;
        LgUns prevtime;
```

```
        while (1) {
                time = CLK_gethtime() / counts_per_us;
                if (time >= prevtime + 100) {
                        prevtime = time;
                }
                /* Pass through idle loop to pump data to the Real-Time
                * Analysis tools */
                TSK_disable();
                IDL_run();
                TSK_enable();
        }
        IRQ_map(IRQ_EVT_RINT1,15);
}
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. Explain the purpose of the CSL_IRQ.h software library?
2. How are the NI Programming Function VI virtual interface components used?
3. What is the function of real-time IRQ program interrupts?

2.9. Digital tonometer and pulse oximeter

The purpose of the work is to design digital data channels in LabVIEW for medical devices and microprocessor-controlled systems and to study its static and dynamic parameters.

Theoretical information

Digital tonometer - blood pressure monitor consists of a microcontroller, an LCD unit and an internal supercharger, a cuff, a pressure transducer with an amplifier and a valve controlled by an Air-Pressure Controller. The block diagram of the digital tonometer is presented in Fig. 2.9.1 [1].

When the Measurement mode is on, the microcontroller produces an LEDs signal and switches on the pump to pump air into the cuff. At the same time, the pressure measurement in the cuff is performed. From the pressure sensor, a voltage proportional to the air pressure in the cuff is fed to an analog-to-digital converter (ADC) located in the microcontroller. Upon reaching a certain pressure in the cuff, the microcontroller produces a pump stop signal. Air is released from the cuff through the valve, as a result, the pressure in the cuff decreases. The pressure sensor generates an output voltage that is proportional to the applied pressure. The output voltages of the pressure sensor are in the range 0 ... 40 mV. They must be amplified so that the voltage at the output of the DC amplifier is in the range 0 ... 5 V. Thus, the required amplifier with a gain of 125. Then the signal from the operational amplifier will be transmitted to the bandpass filter. The operational amplifier amplifies both the constant and the variable signal components. The filter is designed to allocate a signal in the frequency range 1 ... 4 Hz and attenuates any signal that is outside the bandwidth. The variable component of the filter output signal is used to capture systolic / diastolic pressure and heart rate in a patient. In the final stage, the signal is fed to the analog-to-digital converter and digitized. The microcontroller measures diastolic and systolic pressure as well as heart rate. The measured values are stored in an independent Flash memory and displayed on the LCD. The tonometer connects to the computer via a USB interface.

The audio speaker (Mono Speaker) is intended for sound indication of heart rate, as well as audible confirmation of switching the device on and off.

Modern features:

- Intellisense intelligent measurement technology, which is a double control of blood pressure during the measurement process (the device "listens" to the pressure both at the moment of air injection into the cuff and at the phase of air etching), rapid measurement due to the unmistakable choice of the discharge limit and acceleration in shoulder tonometers and due to the non-decompression method in wrist models, a detailed analysis of the pulse wave, the purpose of which is to select the fragment with the most stable pulsation;

- APS function, which is that the meter is provided with a position sensor capable of determining the optimum height at which the wrist should be positioned at heart level for accurate measurements;
- automatic cuff fixation on the arm;
- memory for two users of 84 results, as well as guest mode;
- calculation of the average of three consecutive measurements, average morning, average evening value;
- arrhythmia indicator;
- traffic indicator;
- high pressure indicator;
- indicator of the correct position of the body body;
- Morning hypertension monitoring function;
- notification by a voice signal of the result of measurement of pressure and pulse, as well as of problems that arise in violation of measurement conditions;
- the ability to connect to a computer and a printer.

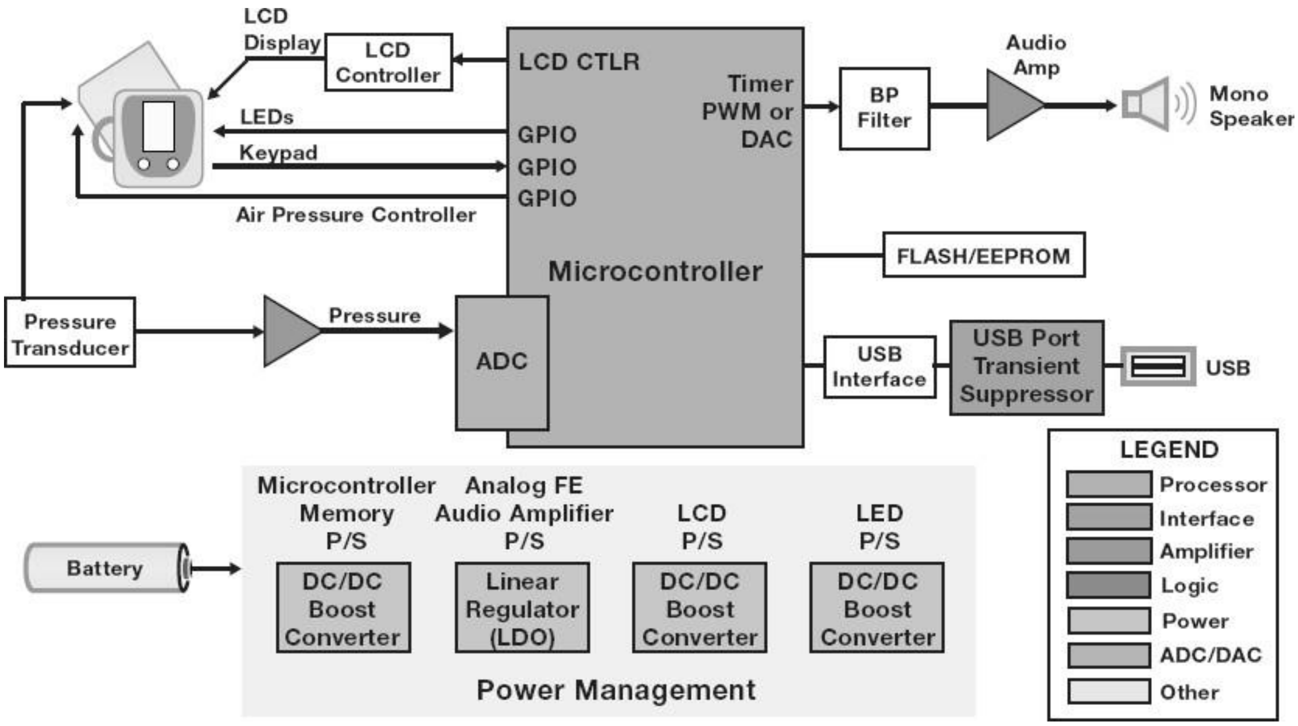


Fig. 2.9.1 Flowchart of a digital tonometer

The pulse oximeter is a modern control and diagnostic medical device designed to measure the saturation of hemoglobin of arterial capillary blood with oxygen (saturation). The cardiovascular system and the lungs of the person are constantly working for one purpose - to saturate the oxygenated arterial blood. There are a number of diseases that are associated with chronic oxygen deficiency (hypoxia), in which this indicator requires constant monitoring and reliable data, the failure of which significantly complicates treatment.

Being easy to use, the pulse oximeter is today an indispensable device for both single measurement and continuous monitoring (continuous monitoring) of the level of saturation and heart rate (heart rate). The method of obtaining data is non-invasive (which does not require blood sampling for the study), so the pulse oximeter measures the required values quickly, facilitating the task of the diagnostic doctor. The advantage of pulse oximetry is that it is not necessary to take blood from the arteries for research. In addition, the pulse oximeter allows to receive and process the data displayed on the device even if the patient is unconscious.

An individual pulse oximeter may be recommended for patients who have been shown to have outpatient oxygen therapy. The device in this case provides the most effective treatment, measuring and controlling the required indicators. The main advantages that make the pulse oximeter a competitive medical product are its affordable price and high accuracy of results. Modern pulse oximeters are shock-resistant, so accidents such as falls and shocks do not threaten its strength. The pulse oximeter is also practically insensitive to temperature variations. All this makes it possible to use the device in emergency medicine.

The pulse oximeter has a peripheral sensor, a microprocessor and a display that displays the pulse curve, its frequency and the saturation value. All the devices are equipped with an audible signal that proportionally reflects the level of saturation. A sensor equipped with two LEDs is most often applied to the finger, rarely to the ear lobe or nose. Measurement technology using a pulse oximeter is quite complex, but its main features can be distinguished.

The sensor has a light source. The light passing through the tissue capillaries to the photodetector is partially absorbed by the soft tissues and blood. The degree of absorption depends on how much blood hemoglobin is saturated with oxygen. The photodetector records the measurement of blood color depending on this indicator. The obtained data are displayed: the pulse oximeter saturates in 5 - 20 s. Pulse rate is calculated depending on the number of LED cycles and reliable signals per unit time.

The most important condition for the reliability of the data obtained is the complete immobility of the finger during the study. The norm is a saturation index of 95 ... 98%. The method of study using a pulse oximeter is highly informative. It assesses the functions of the respiratory organs, recognizes respiratory failure, in which the index of saturation becomes below 95%. Pulse oximetry is used by anesthetists in surgery, as well as in chronic obstructive pulmonary disease, sarcoidosis, tuberculosis, occupational pulmonary diseases. The pulse oximeter is a simple and safe device that requires no special training. The pulse oximetry method makes it possible to detect such dangerous conditions as hypoxia in a timely manner, which opens opportunities for timely implementation of appropriate measures to ensure prevention of dangerous complications, quality improvement and prolongation of life of patients.

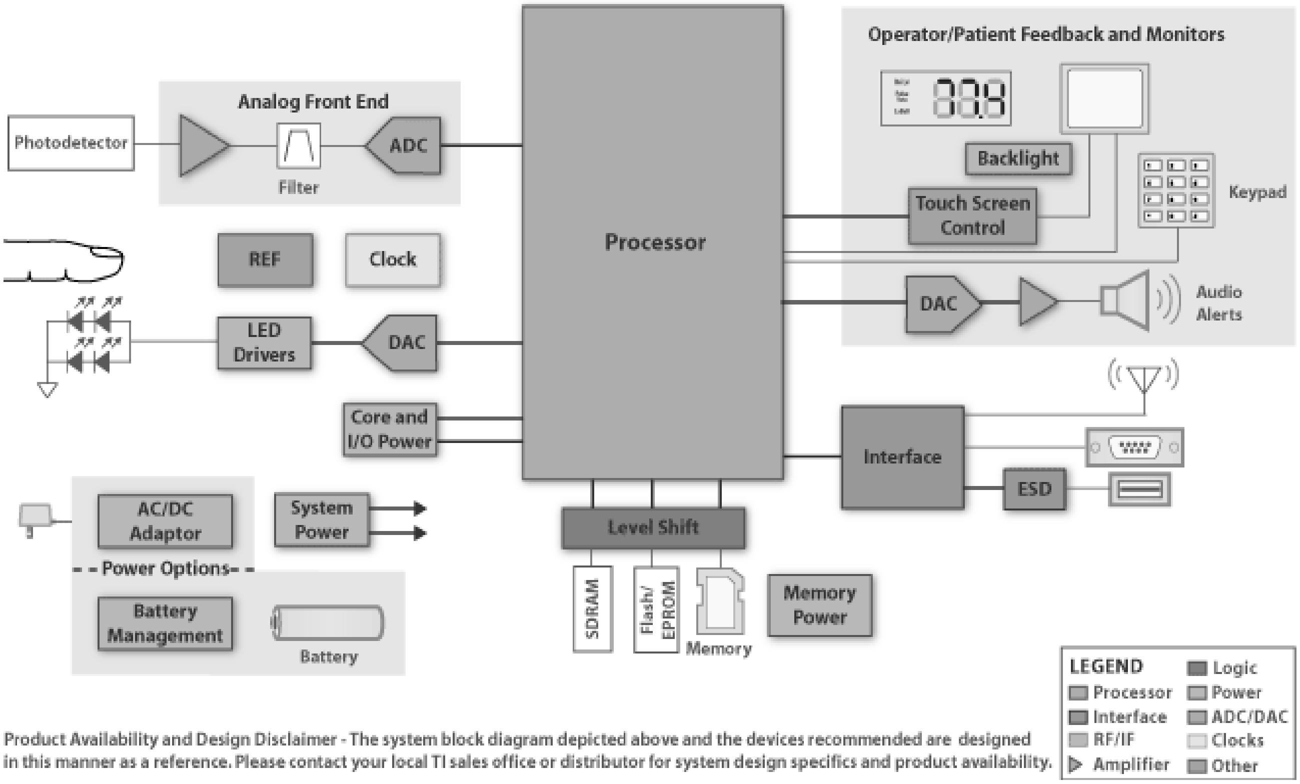
The undeniable advantages of a research method that uses a pulse oximeter over alternative methods that have recently been a priority in the field should be emphasized. Thus, the invasive method (blood sampling through a puncture of the skin), as well as CO-oximetry and multiple gas analyzes, do not give such accuracy as a pulse oximeter. The pulse oximeter is cheaper than the traditional CO-oximeter. The latter, in addition, provides for blood sampling, which is inconvenient, especially if the results are needed as soon as possible. This is most often required for patients experiencing respiratory failure.

A block diagram of a pulse oximeter using a microcontroller is shown in Fig. 2.9.2.

Such designs utilize microprocessors with a high degree of integration and with low power consumption, which is achieved by reducing the number of external components required in the design. In this case, the elements of the signal circuit, power management and display driver are integrated into the microcontroller.

The configuration of this circuit is such that an inverse amplifier with a variable resistor resistor is used in the feedback of the signal circuit. The resistance of the feedback resistor is controlled by the oscillations of the output with small changes in light intensity due to the high level of sensitivity of the circuit. Some signals can be both positive and negative. A dual-self-priming trans-impedance amplifier maintains a very close-to-zero output signal amplitude. A special resistor tied to a voltage of -5 V minimizes the error, since the output voltage becomes very close to 0 V.

For mid- to high-end pulse oximeter implementations, higher processor performance is required to provide higher precision analog low-current components. DSP's low power technology can eliminate signal distortions caused by other light sources or movement during data readings by recording only an informative signal. DSP technology allows accurate data to be obtained with very low signal strength through sophisticated algorithms. These additional processing capabilities are very useful for using a pulse oximeter to measure the absorption of additional wavelengths when detecting the saturation of other types of hemoglobin.



Product Availability and Design Disclaimer - The system block diagram depicted above and the devices recommended are designed in this manner as a reference. Please contact your local TI sales office or distributor for system design specifics and product availability.

Fig. 2.9.2. Structural diagram of the pulse oximeter

Measurement accuracy is ensured by the inclusion of an integrated trans-impedance amplifier, which has no thermal noise due to feedback and no stability problems commonly encountered in trans-impedance amplifiers using large feedback resistors. By using a single photodiode with two integral transimpedance amplifiers, dark currents and ambient light errors can be eliminated, as errors common to both amplifiers can be subtracted. In addition, these amplifiers allow you to synchronize the signal to an integer multiple of the AC frequency, providing extremely effective noise cancellation. The trans-impedance amplifier can be modified by manipulating the setting. In addition, high precision ADCs provide small sizes, excellent performance and single-body solutions for photodiodes that measure the signal.

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 2.9.3 and Fig. 2.9.4).

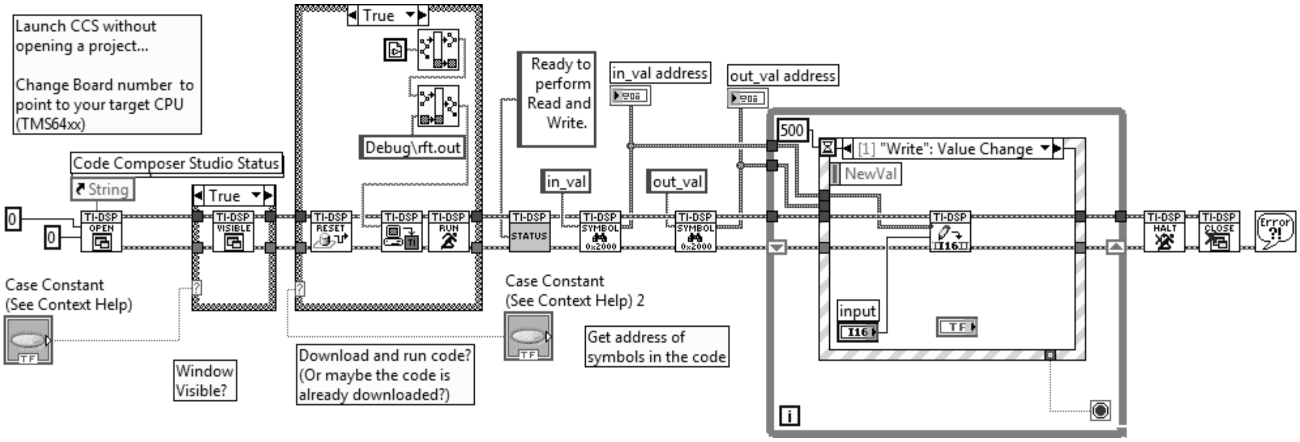
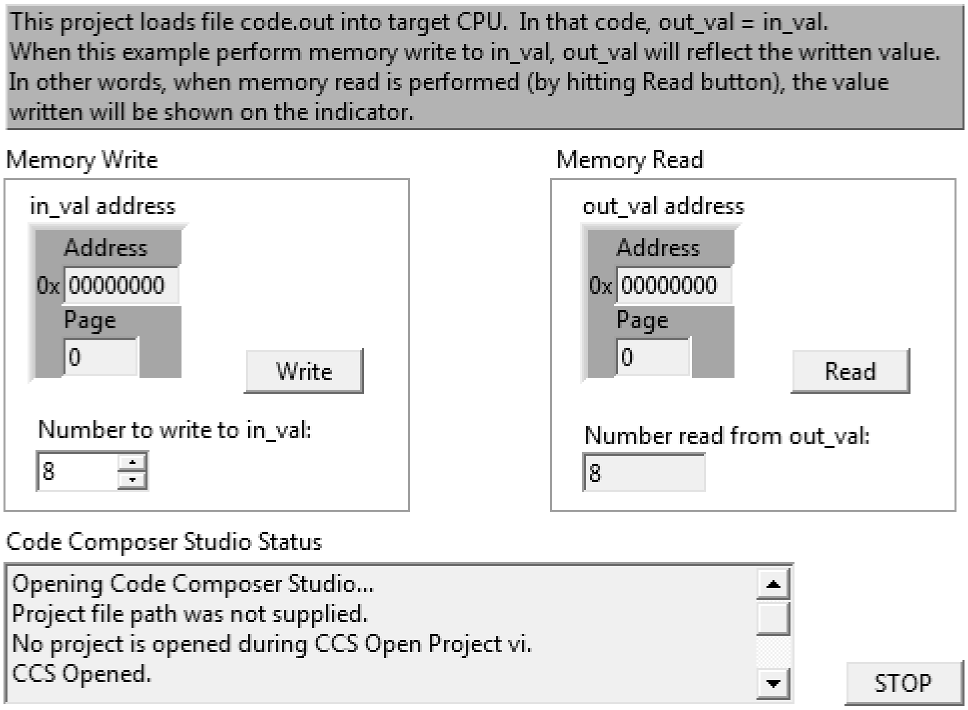


Fig. 2.9.3 National Instruments LabVIEW 2010 Virtual Device Diagram


This project loads file code.out into target CPU. In that code, out_val = in_val.
When this example perform memory write to in_val, out_val will reflect the written value.
In other words, when memory read is performed (by hitting Read button), the value written will be shown on the indicator.

Memory Write

in_val address: Address 0x 00000000, Page 0

Number to write to in_val: 8

Memory Read

out_val address: Address 0x 00000000, Page 0

Number read from out_val: 8

Code Composer Studio Status

```
Opening Code Composer Studio...
Project file path was not supplied.
No project is opened during CCS Open Project vi.
CCS Opened.
```

Fig. 2.9.4 Appearance of the instrument control system

2. Explore the LabVIEW DSP Test Integration Toolkit. Describe the purpose and functionality of the low-level RTDX and CCS IDE components for the DSP Test Integration Advanced VI virtual interface.
3. Document the frequency charts and characteristics recorded in the experiments for the report. To formulate in the report on the completed work the findings of the research and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 2.9.3 and 2.9.4. Set teacher-defined RTDX data transmission settings:

```
// Initialize variables and RTDX channels
        #define BUFFER_SIZE 48
        RTDX_CreateInputChannel(input);
        RTDX_CreateOutputChannel(output);
        int in_val[BUFFER_SIZE], out_val[BUFFER_SIZE];
```

2. To apply software methods of control of data channels with the help of components DSP Test Integration Advanced VI at RTDX level when developing software in microprocessor programming environment Code Composer Studio:

```
// Activate RTDX channels
        RTDX_enableInput(&input);
        RTDX_enableOutput(&output);
// Read-write RTDX channels
        RTDX_read(&input, in_val, sizeof(in_val));
        RTDX_write(&output, &out_val, sizeof(out_val));
```

3.Implement the procedure for overwriting data from the in_val array to the out_val array implemented in LabVIEW based on DSP Test Integration Advanced VI components that interact via channels (input, output) at RTDX level with software in the Code Composer Studio microprocessor programming environment:

```
    int in_val = 0;
    int out_val = 0;
    /* wait for new buffer */
        while(!RTDX_read(&input, in_val, sizeof(in_val)));
    // Loop of Project Running
        while (run)
        {
            out_val = in_val;
```

```
}
RTDX_write(&output,  &out_val, sizeof(out_val));
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. What is the function of APS in the design of the meter?
2. What are the components of DSP Test Integration Advanced VI?
3. What considerations is the Texas Instruments DSK6400 microcontroller?
4. Explain the principles for managing RTDX and CCS IDE data channels.
5. Explain the procedure for initializing variables and RTDX channels.
6. What are the parameters for RTDX activation?
7. Explain the CCS Open Project and CCS Close Project components.
8. Explain how CCS RTDX Read Array and CCS RTDX Write Array work.

3. Design of the ECE daily monitoring system

The main scope of daily monitoring is the assessment of cardiac arrhythmias and the detection of myocardial ischemia. Therefore, all characteristics of the treatment complex can be divided into parameters that affect the quality of arrhythmia analysis (allow you to correctly evaluate changes ST-T) and additional.

For the analysis of arrhythmias it is important to evaluate the form of QRS-complex (automatically and in dialogue with the doctor), which allows to distinguish confidently the complexes of ventricular origin and episodes of violation of ventricular conduction. Visual inspection of about 100 thousand QRS-complexes registered per day, the task is very time consuming and not real in practical work. The day-to-day monitoring system automatically detects arrhythmia and belongs to one of the doctor-prescribed classes. In the form of QRS-complex and RR interval, all arrhythmias are automatically divided into classes, the number of which in different systems can be from 6 to 60. It is obligatory to divide into ventricular and supraventricular arrhythmias, by the number of complexes in the episode of arrhythmia (single, paired, group, paroxysms), heart rate in the episode (rhythms, tachycardia) and the selection of pauses with an assessment of their duration. These classes are defined by all modern systems.

In addition, many systems separately distinguish atypical complexes with an interval equal to the basic rhythm (episodes of ventricular conduction, drainage complexes), asystole, early ventricular extrasystoles, polymorphic ventricular complexes. The number of arrhythmias of each selected class is calculated per day and presented to the doctor in the form of graphs, which allows to compare the number of arrhythmias with heart rate, ischemic ECG changes, with the patient's feelings, time of day. Most modern systems can calculate arrhythmias every minute of observation, making it easier to compare their numbers with short-term events such as physical activity or pain attacks. The most important characteristic is the ability to physically edit the detected classes of arrhythmias, that is, the ability to distinguish classes of rhythm disorders as required by the doctor. Combine or separate automatically selected classes, as well as make the diagnosis that most closely matches the detected arrhythmic phenomenon. A similar possibility is found in most foreign and domestic systems. Its absence significantly complicates the doctor's ability to form a correct conclusion for the detection of arrhythmias.

A relatively new feature of ECG monitoring systems is the estimation of rhythm changes, that is, the automatic detection of not only short-term arrhythmias, but also periods of change in the main rhythm of the heart (for example, the transition from sinus rhythm to nodal). This feature is useful for patients who have a change in the main rhythm over the course of observation, and can separately assess the arrhythmias that occur against different rhythms.

It is now mandatory to identify as important a parameter in the diagnosis of myocardial ischemia as the inclination of the ST segment. For accurate assessment of ischemic ECG changes, it is important to measure the accuracy of the ST segment offset. Ideally, when it is not worse than 10 ... 15 μV. Systems with an accuracy of ST offset measurement of 40 ... 50 μV can complicate the detection of a small offset, since the offset of the ST segment is already 100 μV. Not only the magnitude but also the slope of the ST segment (ascending, descending) is of clinical importance, so most modern systems allow this parameter to be measured. Measurement and presentation of information about the elimination and slope of the ST segment must occur simultaneously throughout all leads. Many systems also provide highlighting episodes of significant ST segment offset and calculate the characteristics of these episodes (number, duration, total duration, ST offset integral, threshold and maximum heart rate per episode). Modern systems allow, based on the variability of the ST segment during the observation period, to automatically identify episodes of ST bias that may be of diagnostic value. Upon confirmation by the physician of the ischemic nature of the detected ST-T intervals, the “threshold” and the maximum heart rate, the integral of the ST bias (the so-called “ischemia index”). Using these systems, the physician receives a large amount of meaningful information when examining patients with ischemic heart block (CHD) and eliminates the need to calculate these characteristics manually.

There is a great deal of interest recently in such a new capability of ECG daily monitoring systems as the estimation of RR interval variability. The analysis of the parameters of the "scattering" of the RR intervals allows to estimate the state of the autonomic tone (predominance of the sympathetic or parasympathetic parts). In addition, rhythm with low variability has been shown to be an independent prognostic adverse factor, especially in patients with myocardial infarction. Most systems that have this feature allow for both temporal and spectral analysis of RR intervals, both over the entire observation time and at the physician's chosen time interval.

The clinical significance of new features such as QT-interval variability analysis and the detection of late myocardial potential are currently being actively investigated. Some systems give the physician an additional opportunity - the analysis of the rhythm, which is registered by the stimulant used to detect defects in his work and disorders of stimulation. Determination of ECG parameters, such as heart rate and the magnitude of the shift of the ST segment, is carried out by modern systems for every 10 ... 60 min. observation. Some events in the patient's life (stress, pain, episodes of ischemic ECG changes) can be very short-lived - less than a minute, and the more detailed the dynamics of parameter changes, the better.

Of the features of systems that have emerged recently, the most common importance is the ability to form a clinical conclusion. Such modern systems allow the doctor to: print a monitoring report (heart rate, examples of arrhythmias with an assessment of their number and characteristics, examples of displacement of the ST segment), automatically "comment" registered in the patient of the violation, both in comparison with age standards, and according to their clinical the significance and their possible prognostic value. The presence of such clinical comments does not relieve the doctor of the need to think, but allows you to remember to pay attention to the diagnostically relevant information. The conclusion is illustrated by examples of ECG, up to the printing of the entire daily ECG, graphs of parameters and number of arrhythmias.

3.1. Flow chart of daily ECG monitoring

The purpose of the project is to design a digital storage device in LabVIEW for a microprocessor-controlled ECE daily monitoring system and to study its static and dynamic parameters.

Theoretical information

A generic block diagram of the analog part of a typical ECG block diagram is presented in Fig. 3.1.1. Input instrumentation amplifiers provide suppression (100 dB or more) of common-mode interference. Amplifiers with low input currents (less than 10 nA) are used to provide high input impedance (tens of IU). This is much greater than the impedance of the signal source, and therefore the effect of the final value of the impedance of the device is not taken into account. The instrumental amplifier gain several times provides a preliminary ECG gain. According to the requirements of GOST, a small gain is used to provide the input range (several hundred mV), which is required due to the polarization of the electrodes.

The ECG amplitude of a few mV is a small part of the input range. The use of HPF allows to isolate the variable component of the electrocardiographic signal for further amplification. The cutoff frequency of the HPF should be as low as possible to reduce the distortion of the low-frequency components of the ECG. Further, the electrocardiographic signal is amplified several hundred times to obtain an acceptable ADC permit, the input range being usually ± 5 mV. In the case of artifacts, patient movements, or external interference, the signal may go beyond measurement.

Installing a low frequency 0.05 Hz PVH takes several seconds during which data can be transmitted. Special circuit solutions are applied to ensure fast setup of the HPF.

After amplification, LPF limits the spectrum of the analog signal to half the sampling frequency to fulfill the conditions of the Kotelnikov theorem.

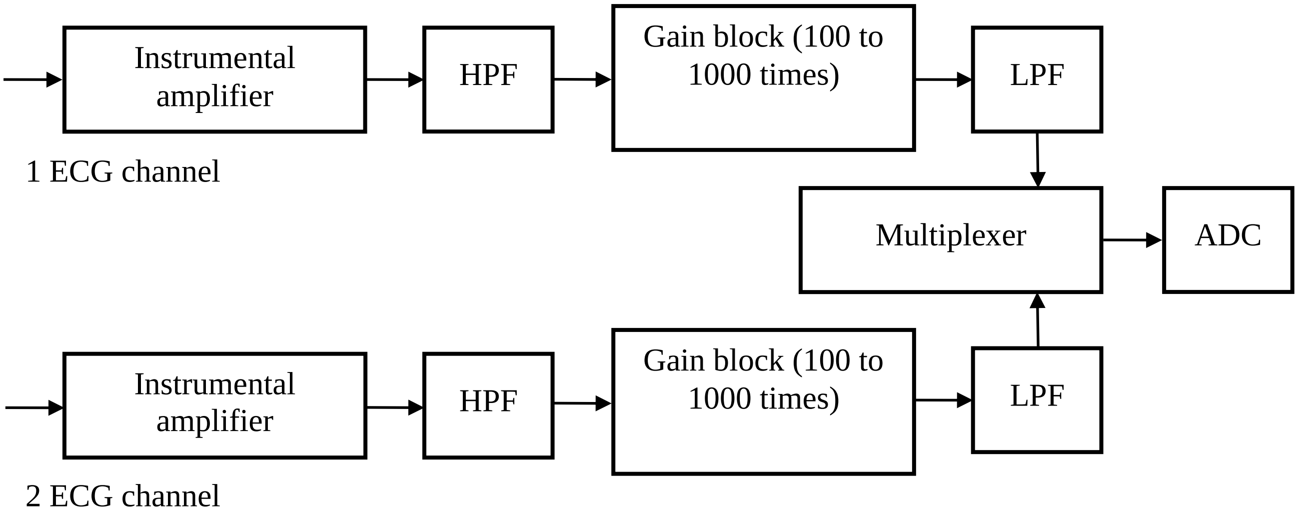


Fig. 3.1.1. Structural diagram of the analog part of a typical daily ECG monitoring

In the classical approach to the sampling process, the effective value of the quantization noise in the frequency band from 0 to Fs / 2 is Q / 3,47, where Q is the weight of the lower digit, and Fs is the frequency of passage of the initial samples. In this case, a large part of the quantization noise enters the working frequency band. To comply with the condition of the Kotelnikov theorem (bandwidth of the useful signal is less than Fs / 2) the LPF must have a high slope of the frequency response in bandwidth. This is necessary for the effective attenuation of high-frequency noise and interference that penetrates the working band as a result of interference with the harmonics of the sampling rate. In most cases, active LPF is used, the multiplexer at the input of the ADC selects the ECG channel. The conversion time of the ADC according to this scheme should be several tens of microseconds to ensure synchronization of the signal sampling. An additional electrode is also used which provides equilibrium potential for the analog part of the device. Through this electrode, a signal received from the conversion of one or more input potentials, designed to compensate for the high-amplitude common-mode interference, primarily from the power supply (50 or 60 Hz), may be fed to the patient's body.

We can distinguish the following main limitations and disadvantages inherent in the classical approach to the development of the analog-digital tract ECG:

- precision large capacitors with small leakage currents in the analog HDF are required;

- the analog signal may go beyond the ADC measurement range for artifacts, patient movements, or external interference (for example, the time of establishing a low-frequency 0.05 Hz HPF is more than three minutes);
- due to the low bit capacity of the ADC, an additional cascade of gain (hundreds of times) after the HPF is required;
- high-order analog lowpass is required to limit the frequency range of the ECG signal at signal sampling.

The advantage of ECG circuits with high-bit ADC

With sufficient bit rate and speed of ADC conversion, it is possible to abandon the classical scheme of constructing the structural scheme of the analog part of the ECG and go to the next structural scheme (Fig. 3.1.2).

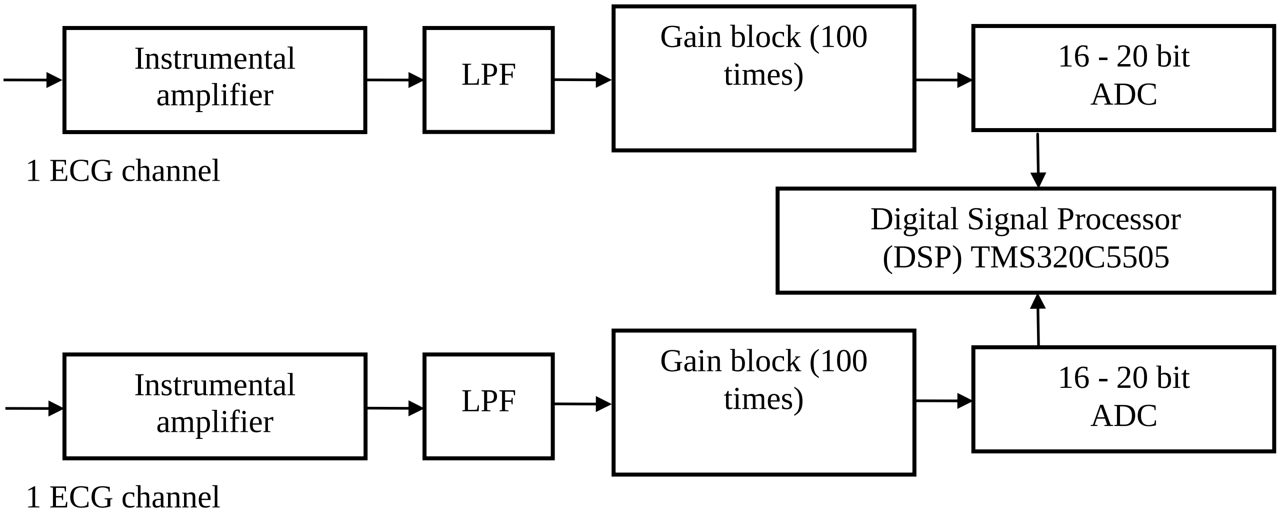


Fig. 3.1.2. Structural diagram of analog part of daily ECG monitoring at high-bit ADC

Figure 3.1.3 presents a block diagram of a daily ECG monitoring system with simultaneous ADC sampling.

It is still necessary to use instrumental or differential amplifiers with high coefficient of suppression of common-mode interference. However, due to the high bit rate and wide input range of the circuit, the HFD and additional signal amplification can not be used, but ECG signal sampling together with a constant component can be used. The permanent ECG component at the input of the device can be offset by the extra bits.

As a result, the number of electronic components to build such a circuit is sharply reduced. There is no need to use large capacitors with low leakage currents in the analog microwave, which significantly reduces the size. HPF is implemented digitally with the required cutoff frequency, depending on the task.

Stages of digital processing in the process of signal formation by electrocardiograms:

- elimination of the constant component drift (non-recursive high-pass filter, filter order 6 - 30);

- removal of high-frequency noise, tremor, removal of guidance with the frequency of the network (non-recursive low-pass filter, order 20 - 100, recursive second-order filter);

- decimation (decrease in sampling rate). This stage is due to the possibility of reducing the quantization noise and reducing the requirements for LPF in the analogue part of the daily ECG monitoring system. Therefore, a high sampling rate is not required. Decimation reduces the amount of memory required to store measurement data.

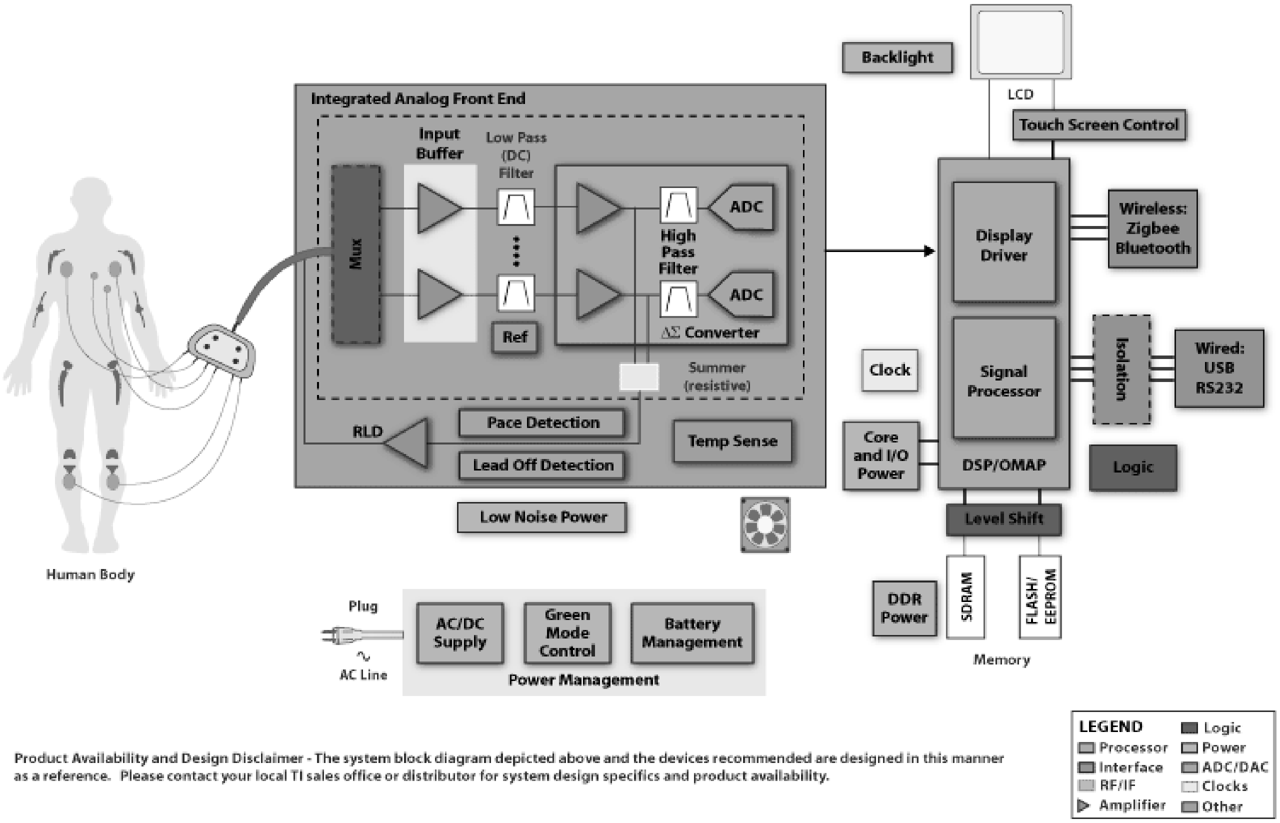


Fig. 3.1.3 Block diagram of daily ECG monitoring with simultaneous ADC sampling

In the case of resampling, the input signal has a quantization frequency of KFs, where K is the resampling ratio, Fs is the sampling rate according to the Kotelnikov theorem. Quantization noise in the frequency band from Fs / 2 to KFs / 2 is suppressed by a digital filter in the output stream, which leads to an improvement in the signal-to-noise ratio by a value of 10 × lg (K). In addition, by increasing the cutoff frequency of the LPF, a small non-uniformity of the frequency response and the frequency response of the digital filter and high linearity can be achieved. Moreover, the analog filter itself degenerates into a simple R-C link.

To improve the signal-to-noise ratio by 6 dB (1 bit), it is necessary to increase four times the oversampling factor. In order to keep the value of this factor within reasonable limits, it is possible to break the quantization noise spectrum so that its main part is between Fs / 2 and KFs / 2, and only a small part - by the segment [0 ... Fs / 2]. The sigma-delta modulator performs this function. After this distribution, the digital filter can easily reduce a significant part of the quantization noise energy, and the overall signal-to-noise ratio that determines the dynamic range increases significantly..

Choosing the type of multi-bit ADC for daily ECG monitoring

There are a large number of analog-to-digital conversion methods that differ significantly in potential accuracy, conversion speed, and complexity of hardware implementation. According to the method of transformation ADCs are divided into serial, parallel and serial-parallel. The disadvantage of sequential ADCs is the low noise immunity of conversion results. When using a sequential type of conversion, the noise immunity increases. Among the sequential type are the integrating type ADCs, in which the input signal is interpreted at a certain time interval, which in many cases suppresses interference at the conversion stage.

Sigma-Delta ADCs (formerly called ADCs with equilibrium or charge balance) refer to a subset of ADCs of integrating type. By their name, these converters are required to have two blocks in them: an adder and an integrator. The basic principle is to average the measurement results over a large time interval to reduce the noise introduced by the noise and increase the resolution. Comparison of the sigma-delta ADC with other integrating ADCs (single-stroke and multi-actuation integration) shows the following advantages: higher linearity of the sigma-delta ADC characteristics, since its integrator operates in a narrow dynamic range; much less linearity of the transient response of the amplifier (on which the integrator is built); much smaller capacitance of the ADC sigma delta integrator (tens of pF), and the capacitor can be made directly on the crystal of the integrated circuit; The sigma-delta ADC has virtually no external elements, which significantly reduces the area occupied by the ADC on the board and reduces the noise level. For example, a 24-bit AD7714 Sigma Delta ADC is made as a single-crystal integrated circuit in a 24-pin housing. Most integrated sigma delta ADCs have an advanced analog and digital part, a built-in controller. This allows you to implement modes of automatic zero setting and full scale self-calibration, save calibrated coefficients and transmit them at the request of an external processor. Sigma-Delta ADCs are widely used in measuring devices that require a large dynamic range at low speeds.

Fully integrated Analog Interface (AFE) is required for portable professional electrocardiograph (ECG) equipment as well as for monitoring patients in residential medical applications. The eight-channel 24-bit ADS1298 interface reduces component count and power consumption by up to 1 mW per channel and achieves the highest level of diagnostic accuracy.

ADS1298 includes:

- eight amplifiers with programmable gain and low noise (PGAs);
- eight high-resolution analog-to-digital converters, internal LPF and real-time sampling;
- integrated amplifier for the right leg electrode;
- integrated amplifiers for the Wilson Central Electrode (WCT) and Goldberg Electrodes (GCT);
- continuous monitoring of electrode connection;
- Built-in generator and reference source to reduce board space and reduce power consumption.

Key features and benefits of ADS1298:

- simplifies the design and saves space on the circuit board;
- 1mW / channel power is provided, reducing energy consumption by up to 95% over discrete components, increasing portability of equipment and improving patient mobility;
- the ECG circuit does not require additional filters included in the sigma-delta ADC scheme;
- ADS1298 has a noise level down to the 4-uVpp input, which is significantly better than the one installed in IEC60601-2-27 / 51, which provides more accurate measurements in portable devices and high-density ECG and ECG equipment;

- a complement to the analog interface is the low power consumption digital signal processor (DSP) family TMS320C5505.

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 3.1.4 and Fig. 3.1.5).

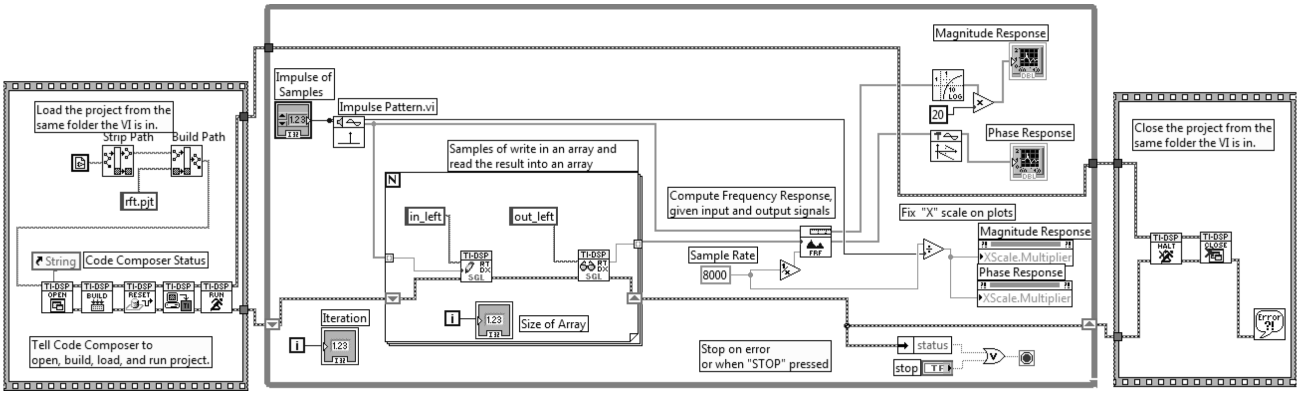
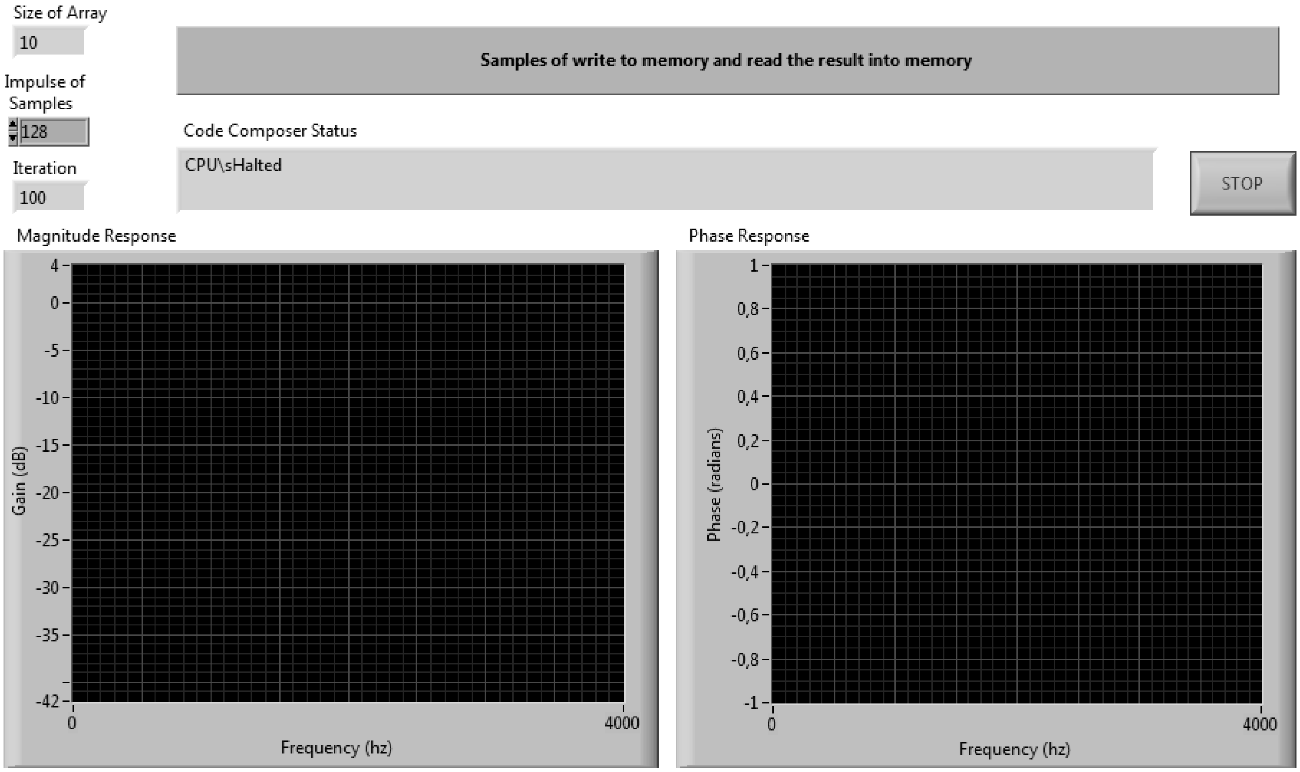


Fig. 3.1.4 Schematic of the virtual instrument in National Instruments LabVIEW 2010



Size of Array: 10

Impulse of Samples: 128

Iteration: 100

Samples of write to memory and read the result into memory

Code Composer Status: CPU\sHalted

Fig. 3.1.5 Appearance of the instrument control system

2. Explore the LabVIEW DSP Test Integration Toolkit. Describe the purpose and functionality of the low-level RTDX and CCS IDE components for the DSP Test Integration Advanced VI virtual interface.
3. Document the frequency charts and characteristics recorded in the experiments for the report. To formulate in the report on the completed work the findings of the research and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 3.1.4 and 3.1.5. Set teacher-defined RTDX data transmission settings:

```
// Initialize variables and RTDX channels
        #define BUFFER_SIZE 48
        RTDX_CreateInputChannel(in_left);    // input channels
        RTDX_CreateInputChannel(in_right);  // for coefficients
        RTDX_CreateOutputChannel(out_left);   // output channels
        RTDX_CreateOutputChannel(out_right); // output channels
        int input[BUFFER_SIZE], output[BUFFER_SIZE];
```

2. Use software methods for managing data channels using DSP Test Integration Advanced VI components at RTDX and CCS IDE level when developing software in Code Composer Studio microprocessor programming environment:

```
// Activate RTDX channels
        RTDX_enableInput(&in_left);
        RTDX_enableInput(&in_right);
        RTDX_enableOutput(&out_left);
        RTDX_enableOutput(&out_right);
// Read-write RTDX channels
        RTDX_read(&in_left, input, sizeof(input));
        RTDX_read(&out_left, input, sizeof(input));
        RTDX_write(&out_left,  &output, sizeof(output));
```

```
RTDX_write(&out_right, &output, sizeof(output));
```

3. Implement the procedure of reading / writing data from an array (output, input), representing the virtual memory of a virtual device in LabVIEW based on DSP Test Integration Advanced VI components that interact via channels (in_left, out_left) at RTDX level with the software in the Code Composer Studio microprocessor programming environment:

```
/* Wait for new buffer */
    while(!RTDX_read(&in_left, input, sizeof(input)));
/* See if there are new coefficients to be read */
    if (!RTDX_channelBusy(&in_right))
    RTDX_readNB(&in_right, &gFIRCoefficients, sizeof(gFIRCoefficients));
    ProcessData (output, input, 1);
    RTDX_write(&out_left,  &output, sizeof(output));
    RTDX_write(&out_right, &output, sizeof(output));
```

The processing of data read / write to an array (output, input) is performed in a ProcessData procedure (output, input, 1), which may be a function of tuning the digital filter coefficients:

```
int ProcessData (int *output, int *input, int gain)
{
    int i;
    double filtered;
    for(i=0; i<BUFFER_SIZE; i++) {
        filtered = FIRFilter(input[i]*gain, kTAPS, gFIRHistory, gFIRCoefficients);
        output[i] = (int)(filtered + 0.5); // round to nearest
    }
    return 0;
}
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. What are the advantages of the eight-channel 24-bit ADS1298 interface?
2. What are the components of DSP Test Integration Advanced VI?
3. What considerations is the Texas Instruments DSK6400 microcontroller?
4. Explain the principles for managing RTDX and CCS IDE data channels.
5. Explain the procedure for initializing variables and RTDX channels.
6. Explain the CCS Run Project and CCS Halt Project components.
7. Explain how CCS RTDX Write SGL and CCS RTDX Write SGL work.

4. Design of microprocessor meter based on PWM sensor

A block diagram of a meter using a microcontroller (MC), which provides the measurement of parameter Z of some physical value or object, is presented in Fig. 4.1.1. The meter consists of an analog part (sensor), a microcontroller and an indication system. The microcontroller in this case is implemented on the basis of a microprocessor system (MPS), including a microprocessor (MP) having memory circuits (ROMs) and interface devices, as well as the control circuit. The MK ROM is used to store the work program, the MK MK is used to store the received results temporarily.

The work of the meter is as follows.

MK generates start pulses with a specified duration and repetition period. When applying a startup pulse from the MC to the analog part, a voltage pulse is formed, the duration of which is proportional to the value of the measured parameter Z, and the amplitude corresponds to the level of logic unit for the selected series of microprocessor kit. MK starts polling the state of the analog part with a specified repetition period, which corresponds to the measurement accuracy. With each survey, the measurement result is incremented by one, and so on until the output of the analogue part has an impulse (or 0 appears). The result is recorded in the RAM at the address specified in the registers of general purpose (total memory) H, L of the microprocessor and is issued to the display system. When a new startup pulse is received, the process is repeated, the address specified in the HZ L, L, is increased by one.

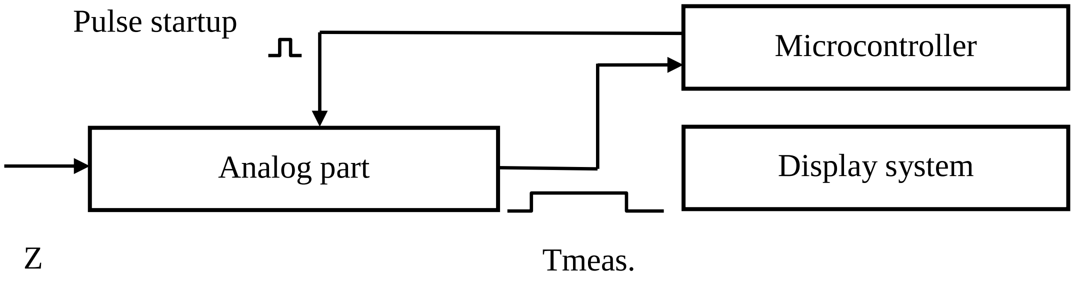


Figure 4.1.1. Structural diagram of the meter

Development of the concept of the microcontroller, as well as any other digital device, includes the choice of a series of used integrated circuits (ICs), the choice of specific ICs in the composition of the series according to functional purposes, ensuring the operation of the ICs in a given mode by direct filing of control signals or programming, expansion digit IC through circuit solutions and layout scheme.

To develop the schematic diagram of the microcontroller, you must perform the following points of the specification:

1. The rationale for choosing a series of microprocessor kit:

- calculate the maximum number of N polls of the analog part during the measurement:

$$\mathbf{N} = \frac{\mathbf{Z_{max}}}{\boldsymbol{\Delta}\mathbf{z}},$$

where Zmax is the maximum value of the measurement parameter, Δz is the measurement accuracy.

– calculate the analogue part survey period based on the maximum measurement time and the number of polls conducted during that time:

$$\mathbf{T} = \frac{\mathbf{T_{meas.max}}}{\mathbf{N}},$$

where $T_{meas.max}$ - maximum measurement time;

- calculate the maximum measurement time for the phase shift meter:

$$\mathbf{T_{meas.max}} = \frac{\mathbf{T_{in}}\boldsymbol{\varphi}_{\mathbf{max}}}{\mathbf{360}},$$

where Tin – input period; φmax - maximum value of phase shift.

- calculate the number of microprocessor clock cycles required to perform a measurement clock without taking into account the delay program, using the number of clock cycles for each command;

- calculate the required processor frequency:

$$\mathbf{F_{mp}} \geq \frac{M}{T}.$$

This uses the principle that the lower the processor frequency, the less power consumed, the cost and most likely the higher reliability.

- calculate the time of execution of the measurement cycle (without delay), taking into account the number M corresponding to the number of MP cycles, and the frequency Fmp for the selected MP;

- to calculate the duration of the delay program on the Tzat, taking into account the need to ensure the equality of the period T of the clock pulses of the meter and the time of execution of the measurement clock;

- calculate the required number of I / O ports to transmit data to the display system:

$$\mathbf{l = int\left(\frac{1}{8} int\left(log_2 \frac{Z_{max}}{\Delta z}\right)\right)},$$

where int (P) is the closest integer to P.

- calculate the bit of the data bus according to the formula:

$$\mathbf{m \geq int(log_2(N + 1))}.$$

The value of m is chosen equal to the nearest value of a series of numbers 8, 16, 32, 64;

calculate the bit width of the address bus n = 2 · m;

- to conclude on the possibility of using the selected microcontroller on the basis of comparison of data on MP [3] and received on calculations requirements.

1. Clarification of the required number of n bits of the bus address MP:

- determine the number of bus bits the address required to address to N1 lines of the program by the formula:

$$\mathbf{n \geq int(log_2(N_1 + 1))},$$

where N1 is the number of rows in the program layout.

- determine the number of bus bits the address required to address to K temporarily stored data by the formula:

$$\mathbf{n}_{\text{озу}} \geq \mathbf{int}(\mathbf{log_2}(\mathbf{K}+\mathbf{1})),$$

where K is the number of temporarily stored data.

- check the condition:

$$\mathbf{n} \geq \mathbf{int}(\mathbf{log_2}(\mathbf{N_1}+\mathbf{K}+\mathbf{1})),$$

where N1 is the number of rows in the program layout; K is the number of temporarily stored data.

If the condition is met, the number of bits of the address bus does not change. If the condition is not fulfilled, then it is necessary to select the MP with the bus address of the required bit and rework the justification for the development of the schematic diagram taking into account the characteristics of the MP.

Functional diagram of the microprocessor system is presented in Fig. 4.1.2.

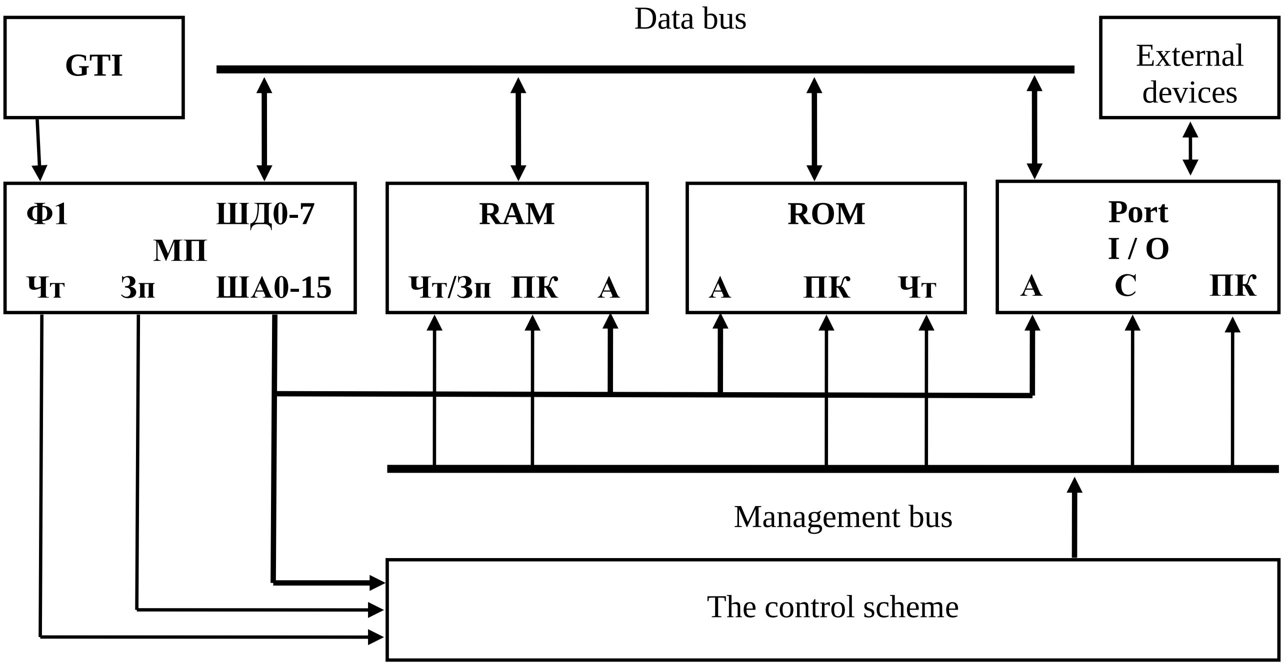


Fig. 4.1.2. Functional diagram of the vehicle

If the MP is to download data from the ROM, it first blocks its own data outputs, then generates the signals needed to connect the ROM to the data bus (PC). Then, the control circuit produces on their basis control signals to the inputs of the selection crystal PKPZU = 1, POPE = PKPORT = 0. As a result, information from the ROM outputs appears on the data bus and the MP reads the data.

The RAM and I / O port reads like this. When reading information from the RAM, the control signals are produced Th / Zn = 1, POPP = 1, PSPP = PSPORT = 0.

When reading information from the I / O port, control signals are output PCPORT = 1, PCPPP = PCPPP = 0.

To write data to the RAM or the I / O port, the MP first places the required data on the data bus and then generates the selection signals of the corresponding device.

Data comes through the MP. For example, to transfer data from an I / O port to the RAM, the MP first reads the data from the port and then writes it to the RAM. Because the data cannot be transmitted directly from the I / O port to the RAM, they must be stored temporarily in the MP.

Thus, the data bus is used to transfer data to the system. All devices use the same bus. Only two data circuits can be exchanged on the data bus at a time.

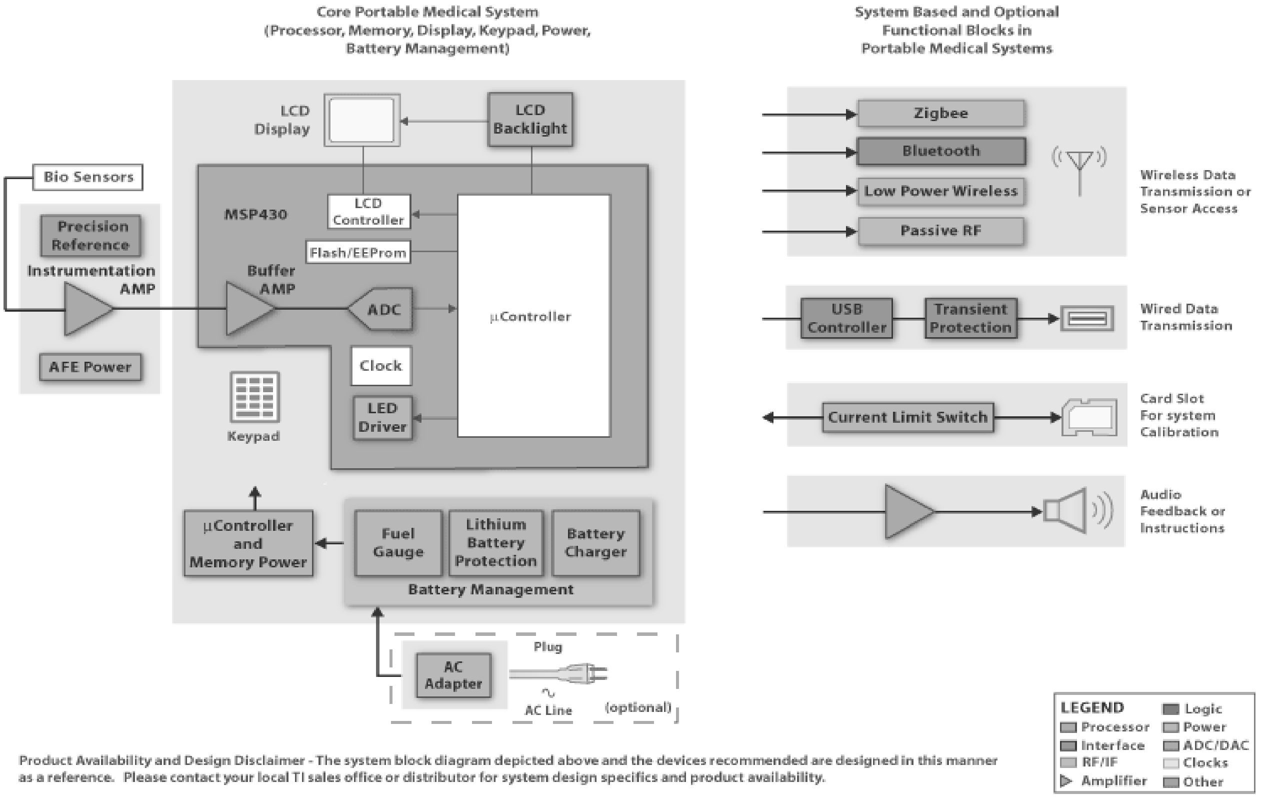
4.1. Patient monitor and portable medical meter

The purpose of the work is to design a digital filter for the patient monitor and the medical meter in LabVIEW with microprocessor control and to study its static and dynamic parameters.

Theoretical information

Portable medical meter. An example of a PWM microprocessor meter for monitoring multiple parameters of a biological object is a portable medical meter that includes: a blood glucose meter, a blood gas meter, a digital tonometer with a heart rate monitor, a digital thermometer. Its structural diagram is rearranged in fig. 4.1.3.

A microprocessor meter is a battery-operated, portable medical device that is designed to perform measurements using various biological sensors with a topology consisting of different blocks that differ in the method of probing, processing and displaying information and a set of functions. For the glucose meter, blood gas meter, digital heart rate monitor, digital thermometer, there are standard system units that are common to each device: power management unit, data and control unit, gain and ADC unit, display and element sensors.



Product Availability and Design Disclaimer - The system block diagram depicted above and the devices recommended are designed in this manner as a reference. Please contact your local TI sales office or distributor for system design specifics and product availability.

Fig. 4.1.3. Flowchart of a portable medical meter

In this unit, power consumption is a key factor in the need to extend battery life, as well as high accuracy with low response time. Requirements for wireless or wired communication, as well as for data processing, the addition of audio or voice prompts oblige manufacturers to use microcontrollers with adequate memory.

The main common subsystems are the following blocks:

1. Analog input sensor. The biosensors from the sensors in portable meters are rather slow and very low in amplitude. Therefore, prior amplification may be required for analog-to-digital conversion, which is performed discretely or in an integrated DAC in a microcontroller.

2. Microcontroller. The microcontroller performs the process of measuring the signal and managing the memory interface and peripherals. Since power consumption is critical, a wide range of low-power MSP430 MP products make it the ideal processor choice. Their high level of integration simplifies design and reduces the cost of the system as buffer amplifiers, data converters, LCD controllers, and the user interface.

3. Communication. Power, speed, and bandwidth are the three main considerations when

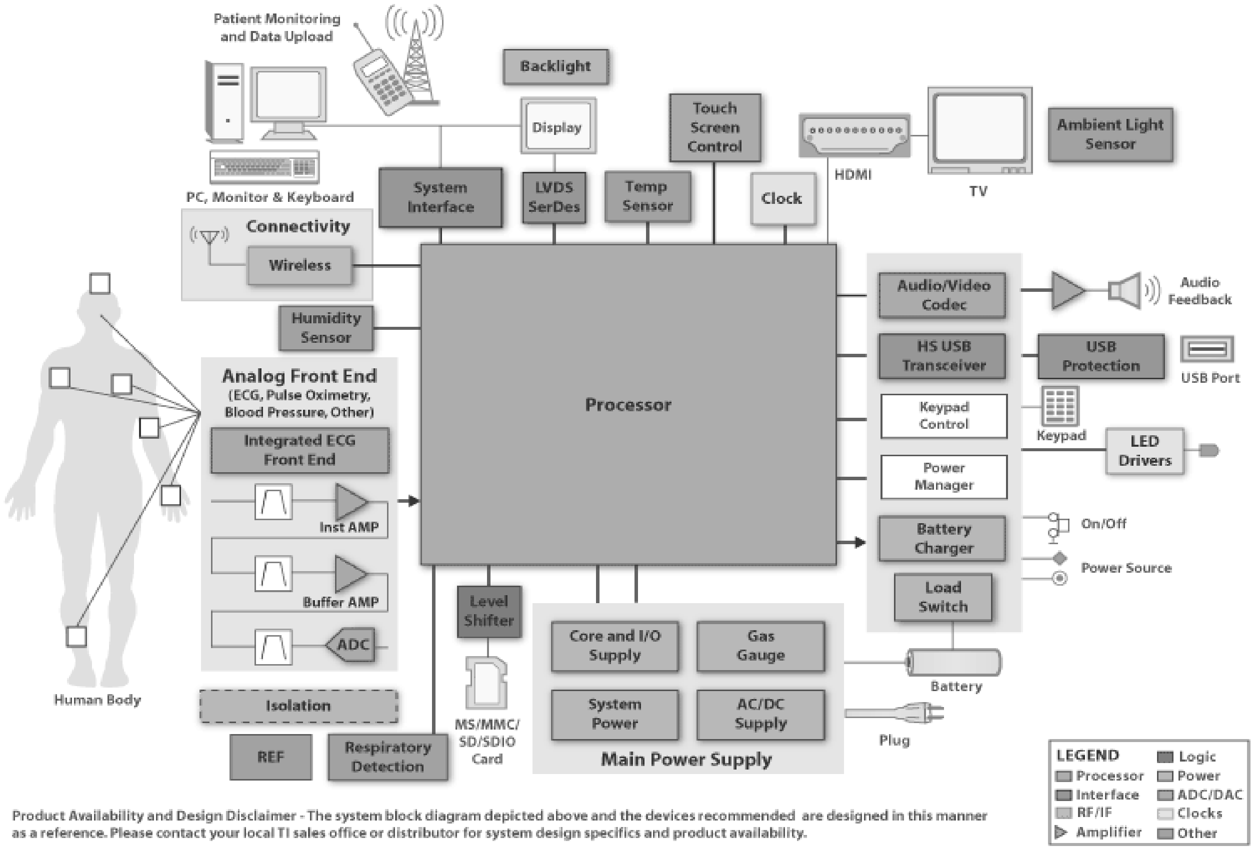
choosing a wireless interface. The Zigbee protocol provides global reach, moderate data rates and a lifecycle, as well as supporting multiple sensors in a single wide-range system. Bluetooth and Bluetooth Low Energy - These protocols provide a limited range with high data rates. The passive low-frequency interface is not only capable of providing a near-field wireless connection, but, depending on the power consumption of the system, is capable of feeding the entire system.

4. Consumption management. Consumption management decisions at the beginning of the design cycle allow you to determine the systemic level of trade-offs that are required to complete the tasks. Small portable medical devices can use disposable batteries, while large portable systems can use rechargeable batteries. Features such as dynamic power management allow the system to receive power regardless of the battery charge. This allows a device with a fully discharged battery to be used as soon as it is connected, without waiting for a voltage to recharge. There are also features such as battery authentication when system security and reliability are critical.

5. Amplifier. The amplifier amplifies the sound coming from either the PWM or DAC circuit, which can be used to notify users when the measurement results are readable. DAC is able to output voice instructions from the speech synthesizer program.

Bedside monitor. The patient monitor is an integral part of any intensive care unit and intensive care unit. Without a patient monitor, it is difficult to imagine a modern hospital. The patient's condition monitoring is subject to high requirements, so the higher the quality and accuracy of the patient's monitor measurement, the more effective the patient's treatment is. The patient monitor (bedside monitor) allows you to evaluate the vital signs, the effectiveness of the resuscitation measures and the correction of hospital therapy, so it is difficult to overestimate the importance of the patient's monitor in the diagnosis and assessment of serious patients.

The block diagram of the monitor is shown in fig. 4.1.4.



Product Availability and Design Disclaimer - The system block diagram depicted above and the devices recommended are designed in this manner as a reference. Please contact your local TI sales office or distributor for system design specifics and product availability.

Fig. 4.1.4. Flow chart of the patient monitor

The structure of the monitor consists of several basic elements:

- analog sensors - ECG-lead, pulse oximeter, blood pressure monitor, body temperature, etc .;

- OMAR processor, which is a programmable microcontroller with internal memory, built-in DAC and ADC, and consists of four main independent functional blocks - the processor core, graphics accelerator, image and video accelerator, and signal processor (DSP);

- For control of the monitor is assigned a touch screen with backlight (Backlight Touch Screen Control Display), additional access to an external keyboard (Keypad);

- additional memory in the form of a card reader (MS / MMC / SD Card) and a level converter (Level Shifter), designed to store patient information;

- Audio encoder (Audio / Video Codec) and audio speaker (Audio Feedback) are intended for sound signaling in case of various emergencies that occur with the equipment or in case of exceeding the patient's controlled parameters set by the limits of values;

- USB port together with USB transceiver designed to connect the monitor to a personal computer;
- LED-Drivers LEDs are designed to monitor various DC voltages generated by the monitor's power supply;
- with the help of wired (Ethernet) and wireless communication technologies (Bluetooth, Wi-Fi, Low Power Wireless) the monitor can be connected to any other hospital equipment, including narcosis-breathing apparatus, defibrillators, telemedicine equipment for remote monitoring of the patient;
- The Main Power Supply converts the 220V AC voltage to DC, and charges the battery and controls its charge level.

Portable monitors for medical meters were able to measure such parameters as blood pressure, glucose level, pulse rate, carbon dioxide level, and other biometric parameters. Today, monitors are not only portable but also flexible devices that can be adapted to different clinical applications, support different wired and wireless interfaces. Patient monitors monitor the dynamics of the patient's respiratory and cardiovascular systems, allow such tests as ECG, respiratory rate, non-invasive blood pressure measurement, degree of oxygen saturation, dual-channel body temperature measurement and other important biomedical parameters.

Regardless of the monitor being a measurer of one or more parameters, target capabilities, power consumption, and system versatility are often key requirements. At present, the monitor can be moved with the patient from the operating room to the intensive care unit, to the hospital ward. Universal patient monitors are portable devices, so they can be used both in the hospital and during transportation. Similarly, patient monitors can be used both in adult patients and in children and newborns. This is of paramount importance in the world of health today.

The most important features of today's patient monitors are mobility, ease of use and the ability to transmit data remotely. Mobility includes portability as well as the ability to interact with other medical devices such as anesthesia devices or defibrillators. The ease of use of monitors is achieved through the use of a touch screen that displays metrics, a multi-level menu of profiles that can be customized for the environment, and also taking into account the statistics of the natural movements of the patient. Data transmission is possible through both the RS232 serial port and through wireless protocols. Hospitals can support specific network infrastructures in all areas, but ambulances, homes, and in other cases may need support for different protocols. The increasing need to minimize health care costs has led to the relocation of inpatient care and monitoring outside the hospital. The provision of health care in densely populated rural and remote areas, in developing economies leads to the need for remote monitoring of patients using telemedicine technology.

Systems for the treatment and monitoring of patients present equipment that is strikingly similar to cellular telephone systems. OMAP technologies with built-in ARM and DSP processor core directly address these issues. The OMAP 3 processor performs sequential digital signal processing, measurement, and analysis of patient status monitoring. The powerful ARM processor operates at a high level of OS (HLOS), which makes new monitoring easy and provides ample setup and simple management. Identifying abnormal operating conditions and interacting with a central server is essential in ensuring timely healthcare. OMAP 3 has a large set of peripherals to support various connectivity options such as Bluetooth, WiFi, ZigBee and other new standards.

Main characteristics of ECG monitor:

- Assignments: 3, 5, 10, 12

3-lead: RA; LA; LL or R; L; F,

5-lead: RA; LA; RL; LL; V or R; L; N; F; C,

10-lead: RA; LA; RL; LL; V 1 - V6; or R; L; N; F; C, C1 - C6.

- Channels: ECG waves - 2 channels. Gain: ×0.25; ×0.5; ×1; ×2, auto.

- Heart rate measurement range: 20 ... 250 beats / min. HR accuracy: greater than ± 1% or ± 2 bpm Resolution: 1 beats / minute

- ST-segment analysis: -2.0 mV to 2.0 mV. ST-segment arrhythmias and displacements analysis, ST-segment map. QT / QT interval monitoring c.

- Filters. Diagnostic mode: 0.05 ... 100 Hz or 0.05 ... 150 Hz (for 12-channel). Monitoring mode: 0.5 ... 40 Hz. Surgery Mode: 1 ... 20 Hz. Protection: from defibrillator and ECHF interference. ECG waves: 9-channel ECG, 12-channel ECG.

- Input impedance:> 5 Mм.

- Breathing. Method: impedance. Measuring range: 0 ... 100 rpm

- Accuracy: ± 1 rpm

- SpO2 (pulse oximeter). Measuring range: 0 ~ 100%. Measurement accuracy: ± 1% (SpO2

90 - 100%); ± 2% (SpO2 70-89%). HR range: 0 ~ 250 beats / min Resolution: 1%. Accuracy: ± 1 beats / min

- Nellcor SpO2 module. Measuring range: 0 ~ 100%. Measurement accuracy: ± 2% (for adults and children); ± 3% (for newborns). Heart Rate Range: 0 ~ 254 bpm Resolution: 1 beats / minute Accuracy: ± 3 beats / min

- NIAD (non-invasive blood pressure). Measurement method: automatic and oscillometric. Types of measurement parameters: systolic, diastolic, average pressure. Measurement modes: manual / automatic. Auto Measurement Interval: 1 ... 480 min. Measurement time: User-defined. Unit of measurement: mm Hg. Art. NIAD range: 0 ~ 300 mmHg Art. Overpressure protection: Double overpressure protection.

- Pressure measuring range: Adults / children: systolic pressure 60 ~ 240 mmHg. art., diastolic pressure 30 ... 190 mm Hg. Art., the average pressure of 40 ... 210 mm Hg. Art. Newborns: systolic pressure 25 ... 135 mm Hg. art., diastolic pressure 12 ... 110 mm Hg. Art., the average pressure of 18 ... 120 mm Hg. Art. Pressure measurement accuracy: ± 5 mm Hg. Art.

- IAD (invasive blood pressure). Measuring range: 50 ~ 300 mm Hg Art. Channels: 2 channels. Measured parameters: ARP, PA, CVP, LAP, ICP, P1, P2. Accuracy: greater than ± 1% or ± 1 rpm

- Temperature. Measuring range: 20 ... 45 ° C. Resolution: 0.1 ° C. Measurement accuracy: ± 0,1 ° C (does not include sensor error). Sensor: standard is a leather temperature sensor. Alarm: 20 ... 45 ° C, automatic alarm recording.

- EtCO2 capnography. Method: study of absorption of IR radiation. Sampling rate: 50 ± 10 ml / min. (side). CO2 measurement range: 0 ... 150 mmHg. Art. CO2 resolution: 0.1 mmHg. Art. in the range of 0 ... 69 mm Hg. Art., 0.25 mm Hg. Art. in the range of 70 ... 150 mm Hg. Art. CO2 Accuracy: ± 2% in the range 0 ... 40 mm Hg. Art. ; ± 5% in the range of 41 ... 70 mm Hg. Art .; ± 8% in the range 71 ... 100 mm Hg. Art .; ± 10% in the range of 101 ... 150 mm Hg. centuries ..

- AwRR: ± 1 rpm Response time: <3 s, including transportation time. Numerical method: BTPS (Body Temperature Pressure Saturated). Approximate gas flow rate: 50 ml /min.

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010

(Fig. 4.1.5 and Fig. 4.1.6).

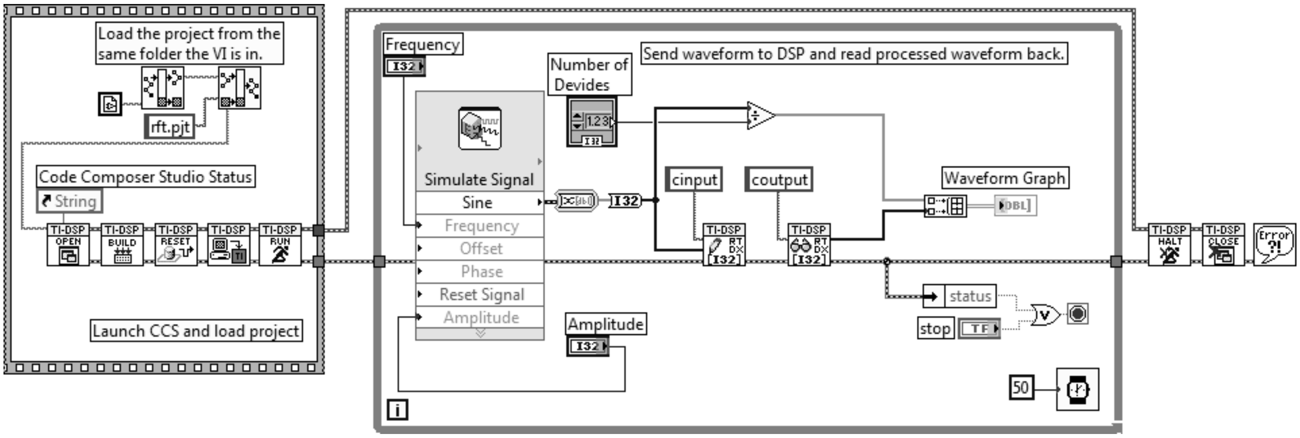
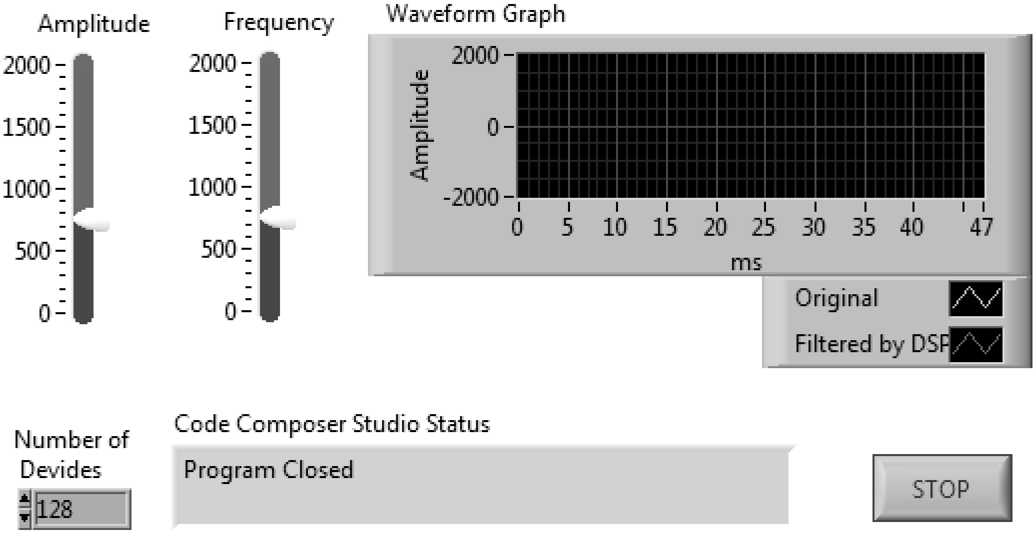


Fig. 4.1.5 Virtual Instrument Scheme in National Instruments LabVIEW 2010



Code Composer Studio Status
Program Closed

Fig. 4.1.6 Appearance of the instrument control system

2. Explore the LabVIEW DSP Test Integration Toolkit. Describe the purpose and functionality of the low-level RTDX and CCS IDE components for the DSP Test Integration Advanced VI virtual interface.
3. Document the frequency charts and characteristics recorded in the experiments for the report. To formulate in the report on the completed work the findings of the research and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 4.1.5 and 4.1.6. Set teacher-defined filter settings:

Amplitude (A) = 0.50V;

Frequency (F) = 1e+03Hz;

2. Use digital filter control software using DSP Test Integration Advanced VI RTDX and CCS IDE components when developing software in Code Composer Studio microprocessor programming environment:

```
// Initialize variables and RTDX channels
          #define BUFFER_SIZE 48
          RTDX_CreateInputChannel(cinput);
          RTDX_CreateOutputChannel(coutput);
          int input[BUFFER_SIZE], output[BUFFER_SIZE];
// Activation of RTDX channels
          RTDX_enableInput(&cinput);
          RTDX_enableOutput(&coutput);
// Read-write of RTDX channels
          RTDX_read(&cinput, input, sizeof(input));
          RTDX_write(&coutput, &output, sizeof(output));
```

3. Implement a digital filter (FIR Filter) with the ability to programmatically adjust the filter coefficients when developing software in the Code Composer Studio microprocessor programming environment:

```
// The Simple FIR Filter
double FIRFilter (double val, int nTaps, double* history, double* coefs)
{
  double temp, filtered_val, hist_elt;
  int     i;
  hist_elt = val;
  filtered_val = 0.0;
```

```
    for (i = 0; i<nTaps; i++)
    {
        temp = history[i];
        filtered_val += hist_elt * coefs[i];
        history[i] = hist_elt;
        hist_elt = temp;
    }
    return filtered_val;
}
```

4. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. What is the work of a biomedical signal meter?
2. What are the components of DSP Test Integration Advanced VI?
3. What considerations is the Texas Instruments DSK6400 microcontroller?
4. Explain the principles for managing RTDX and CCS IDE data channels.
5. Explain the procedure for initializing variables and RTDX channels.
6. What parameters depends on the activation of RTDX channels.
7. Explanation of the procedure for justifying the choice of a series of microprocessor kit.
8. Explain the components of CCS Build Project and CCS Download Code.

5. Telemedicine dynamic surveillance systems

Telemedicine is a method of providing health care services in an area where distance is a critical factor. Telemedicine is a fairly new direction at the intersection of several branches of medicine, telecommunications, information technology. It is undoubted that one of the main advantages of telemedicine is the ability to provide highly qualified help from specialists of leading medical centers in remote areas and significantly save the cost of patients.

Telemedicine consultations are carried out through the transmission of medical information via electronic communications channels. Consultations can be carried out both in "deferred" mode via e-mail - the cheapest and easiest way of transmitting medical information, and in real time on-line using communication channels and video equipment. Scheduled and emergency video consultations and video consultations are direct communication between the consulting physician and the attending physician, if necessary - with the participation of the patient. Moreover, a videoconference communication session can take place between two subscribers and between several subscribers in the so-called multipoint mode, that is, the most difficult cases can be discussed by a consultation of doctors from different medical centers.

The use of network camcorders allows you to organize the broadcast of surgery. For example, through standard means of the Internet it is possible to access camcorders installed in the telemedicine laboratory and operating NAS of Ukraine.

Mobile telemedicine complexes (portable, resuscitation-based, etc.) are being developed to work in the field of accidents. The modern mobile telemedicine complex should combine a powerful computer, easily connected with various medical equipment, means of near and far wireless communication, means of videoconferencing and means of IP-telephony.

Telemedicine dynamic surveillance systems are used to monitor patients with chronic diseases. The same systems can be used at industrial sites to monitor the health of workers (such as nuclear power plant operators). A new direction of development of remote biomonitoring is the integration of sensors in clothing, various medical accessories, mobile phones. For example, there is a vest with a set of biosensors that record the ECG, blood pressure and a number of other parameters, or a mobile phone with the ability to register the ECG and send them by GPRS to the medical center, as well as with the ability to determine the coordinates of the person in case of threat to the patient's life.

The availability of communications and Internet services allows you to develop a trend such as "home telemedicine". This is a remote provision of medical care to a patient outside the medical facility and undergoing home treatment. Special telemedicine equipment collects and transmits the patient's medical data from his home to a remote telemedicine center for further processing by specialists. An example is the monitoring system for patients with heart failure who require regular and frequent examinations, the cost of which is significantly reduced through the use of telemedicine. There are complexes that include sensors that measure body temperature, blood pressure, partial oxygen pressure, ECG, and breathing functions that are connected to a desktop monitor, which in turn automatically sends data to the control center. In addition, audiovisual contact with physicians during consultation or diagnostic procedures is possible.

Although telemedicine today remains primarily a remote diagnosis, its potential is much broader. Promising areas of telemedicine include telesurgery and remote sensing. They allow remote control of medical diagnostic equipment and remote treatment effects, surgical operations. Currently, some options for remote control are already coming into practice. An example is the management of network camcorders, which is effective in monitoring the status of patients in intensive care units and remote monitoring of surgeries. Another example of remote control is the management of a remote microscope, so that the consulting physician has the opportunity to conduct pathohistological or pathocytological studies in full, to view all available samples of material.

Experimental use of telecontrolled manipulators and remote control of them directly during operations (control of scalpel, laser, etc.). The most responsible and complex direction from the point of view of implementation of the direction at present are purely experimental techniques, the implementation of which requires the implementation of many technological innovations.

Prospects for telemedicine are related to the further miniaturization of control and measuring equipment, the introduction of smart technologies, robotics, the latest advances in computer science, applied aspects of nanotechnology.

In the structure of the telemedicine system (TMS) can be distinguished four types of elements, the interaction of which forms a telemedicine network:

1. Channel-forming environment - a set of hardware, software, storage media and technological solutions (protocols and standards) that provide the transmission of heterogeneous information in a geographically distributed environment;

2. Counseling center - medical institution, which has in the staff of highly qualified doctors in different fields of medicine and appropriate equipment for conducting remote consultations, consultations and medical-diagnostic procedures, as well as organizing training (conducting seminars, lectures) of doctors at remote TMS stations;

3. Dispatch point - a separate or functional structure of TMS that performs the functions of filtering requests for counseling, planning and providing consultations, organizing consultations, as well as collecting and disseminating information about the capabilities of counseling centers, as well as containing an administrative service that performs support functions network structure;

4. Remote points - specially equipped medical facilities, whose staff directly interacts with patients and performs a complex of medical, diagnostic, prophylactic and rehabilitation procedures.

If necessary, temporary cells are formed in the structure of the TMS - for example, a complex of remote medical units in battlefields or man-made disasters. Such stations are deployed and connected to the TMS in order to involve groups of experienced experts from leading centers in solving the operational problems that arise in such places. Patient consultations are possible around the clock due to time differences in different time zones.

In the structure of telemedicine hardware, you distinguish four basic components: multimedia information transmission infrastructure, general-purpose computer equipment, specialized computer equipment, specialized medical equipment.

The channel-forming environment of the TMS (multimedia information transmission infrastructure) is independent of the medium of information - these may be cable conductive structures, fiber optic channels, and satellite and radio channels. The equipment and communication channels allow the transmission of heterogeneous information - alphanumeric and graphic, audio and video, as well as digital and analog signals taken from the sensors and data transmitted to the control bodies of the diagnostic medical equipment. The final equipment provides the conversion and matching of signals, their conversion from one format to another, as well as their compression / decompression. It should be noted that modern TMSs for video conferencing can work effectively in various network topologies built on IP, ISDN, ATM, etc. Distributed application and archive servers serve as service providers. Multi-point video, scheduling consultations and distance learning and testing services are run on application servers. Archiving services provide long-term storage, cataloging and retrieval of large amounts of information. General-purpose computer equipment is used to organize doctor-consultant and physician jobs, centralized monitoring panels, and operating room equipment. It consists of computers of different architecture and purpose. In addition to computers, it includes a variety of peripherals - video codecs, camcorders, audio systems, various digitizers.

The composition of specialized computer equipment is determined on the basis of the needs of specific medical programs and may contain specialized scanners, controls, specialized systems for displaying video information, as well as devices for connecting computer and specialized medical equipment.

Diagnostic, medical and rehabilitation equipment can be connected to the TMS directly and through communication devices. If such a connection is impossible or impractical, information from such equipment may be converted into digital form using special equipment such as scanners, digitizers, etc.

For use in telemedicine networks, specialized medical equipment with visual or acoustic feedback from the physician and integrated network support is optimally suited. For cardiology, these can be angiographic installations and various ultrasound scans, in pulmonology - a bronchoscope, in gastroenterology - gastroscopes, in dermatology and endoscopy - dermatoscopes and video cameras with endoscopic nozzles. It can also be a generalist diagnostic equipment - ultrasound, NMR, microscopes, stethoscopes and other equipment.

Protecting stored and transmitted information, authorizing access to the TMS, and finally, ensuring the survivability of the network in different modes of operation (peacetime, emergency events, etc.) forms a complex of TMS security hardware and management solutions. Hardware and software cryptographic tools are used to protect the information stored in archives and transmitted through communication channels.

Authorization of doctors' access to TMS equipment is relevant both when conducting teleconsultations to confirm the authority of specialists and when working with terminals to prevent unauthorized access to medical data. Electronic signature tools are used to verify documents that record teleconsultation results, remote testing, and the like.

Access to TMS resources from external communications networks is ensured through the use of software systems - firewalls. The survivability of TMS is ensured both by the TMS topology, which has the structure of duplicate channels of different physical nature and intellectual switches, and measures for distributed archival storage of information.

5.1. Diagram of the diagnostic item

The purpose of the project is to design digital LPF and HDF filters in measuring channels for microprocessor-controlled telemedicine systems via built-in Ethernet and high-speed USB transceiver.

Theoretical information

One of the options for constructing a remote diagnostic point based on the microprocessor is shown in fig. 5.1.1.

System solutions for low-power diagnostic centers include the following key components:

1. Microprocessor. Microprocessor uses the LM3S9B9x, built on the basis of ARM Cortex M3 with built-in Ethernet and a high-speed USB transceiver. Low sleep power and fast interrupt processing are ideal options for such devices that use battery power. The microprocessor manufacturer provides all the necessary system drivers, so no special software build is required.

2. Communication. For a device to be certified, it must include at least one Personal Area Network (PAN) interface, such as USB and Bluetooth, and one Wide Area Network (WAN) interface such as Ethernet and ZigBee.

3. Audio. Integrated headphone jack and speaker amplifier deliver low power, low cost and small size.

4. Power management. The unit uses a special power supply and processor that provide low cost, low noise, low quiescent current for signal circuits and the processor. And a single cell lithium-ion battery can be used for a long time.

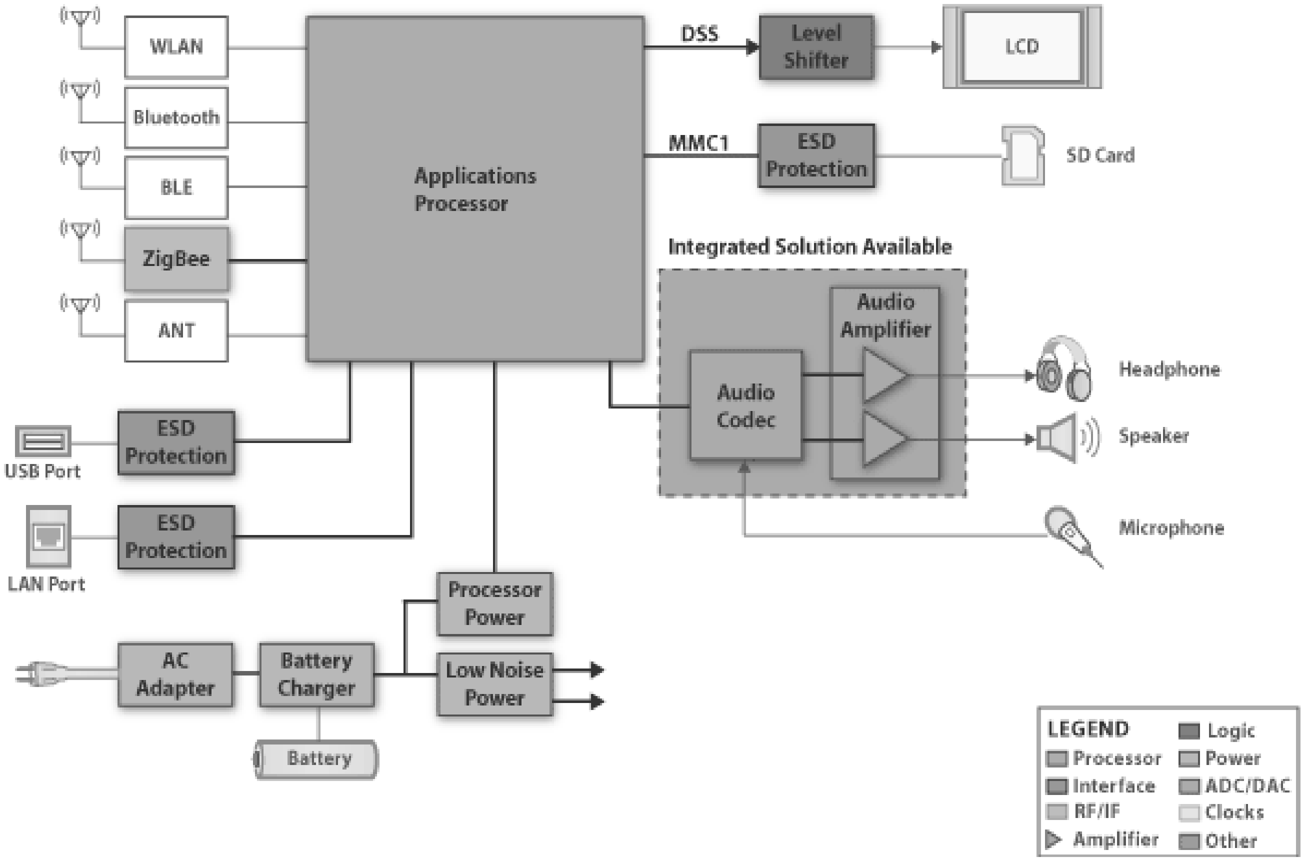


Fig. 5.1.1. Diagrammatic diagram of the diagnostic item

Figure 5.1.2 shows the use of a microcontroller to create a telemedicine consulting center with video output.

Such counseling centers should provide excellent images and sound. Therefore, system solutions include the following key components:

1. The processor. The OMAP3530 processor has an ARM Cortex A8 oriented architecture that provides audio and video interfaces to peripherals. It includes an HD video accelerator to enable high definition video streaming. For TMS systems that do not require HD video streaming, the AM3517, which does not support the HD video accelerator, should be used. Both processors are included in the latest Linux and Windows compatible codebases.

2. Communication. For a device to be certified, it must include at least one Personal Area Network (PAN) interface, such as USB and Bluetooth, and one Wide Area Network (WAN) interface such as Ethernet and ZigBee.

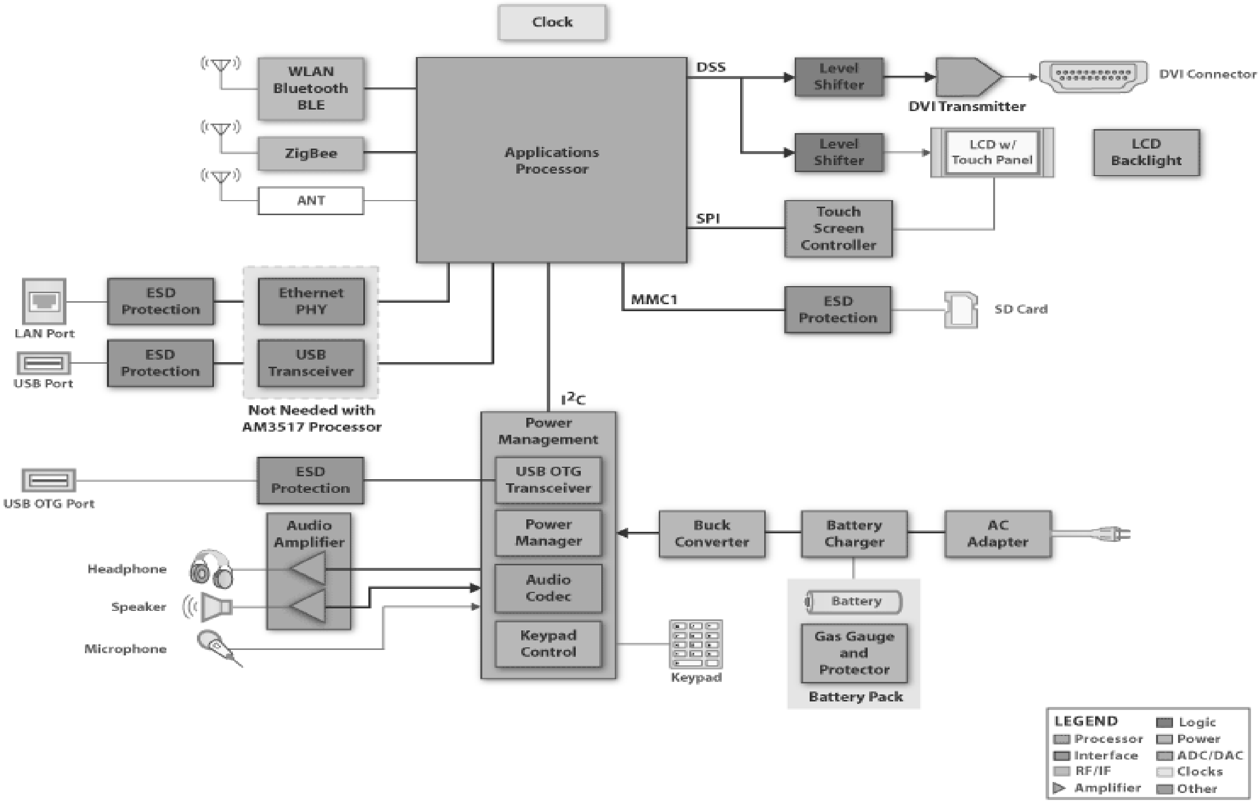


Fig. 5.1.2. Structural diagram of the telemedicine consultation center

3. Power management. The processor is designed to work with OMAP35x devices. It contains a converter, a low-level controller, a charging module, a digital filter audio module, Class D input and output amplifiers. There are also several additional features such as a high-speed USB transceiver.

4. Battery management. In order to support streaming video, wireless and a large screen, high capacity batteries are required with several elements connected in series and in parallel.

4. DVI. TFP410-DVI 1.0, a compatible digital transmitter that supports display resolution ranging from VGA to UXGA in 24 bits. Some of the benefits of this versatile interface include bus bit selection, customizable signal levels, and differential and unbalanced video flow.

Work task

1. Collect the tested scheme of the virtual instrument in National Instruments LabVIEW 2010 (Fig. 5.1.3 and Fig. 5.1.4).

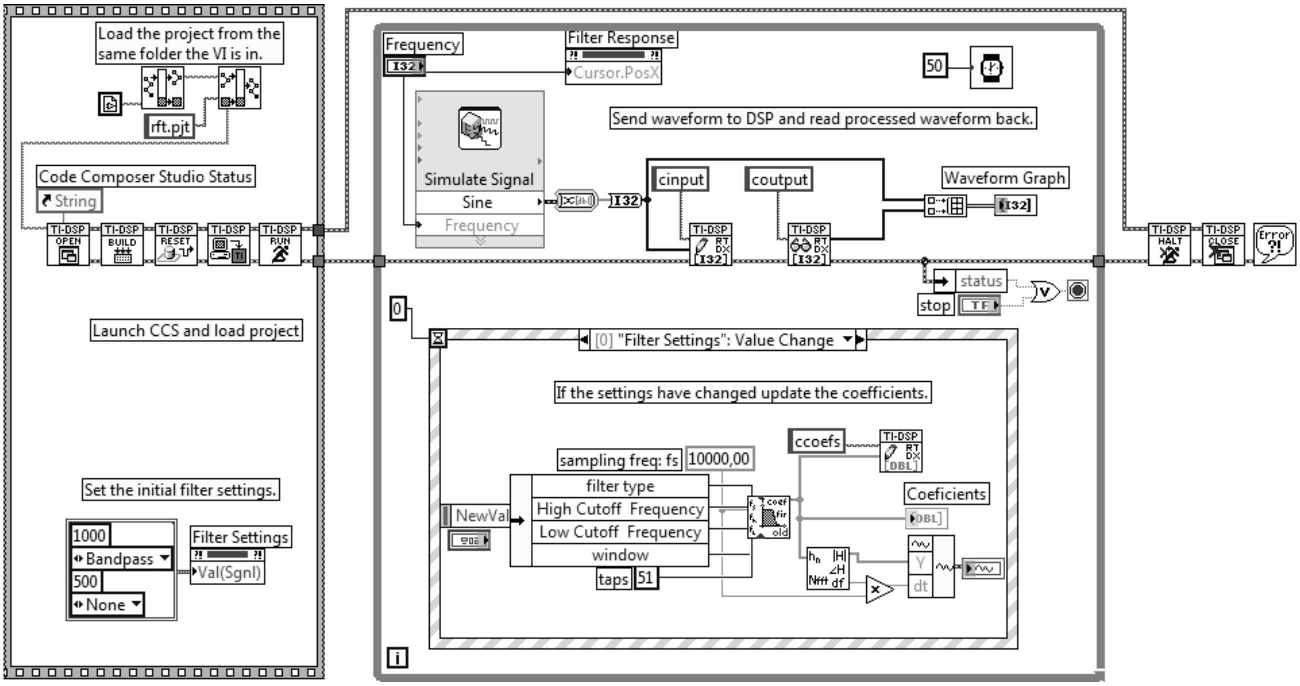
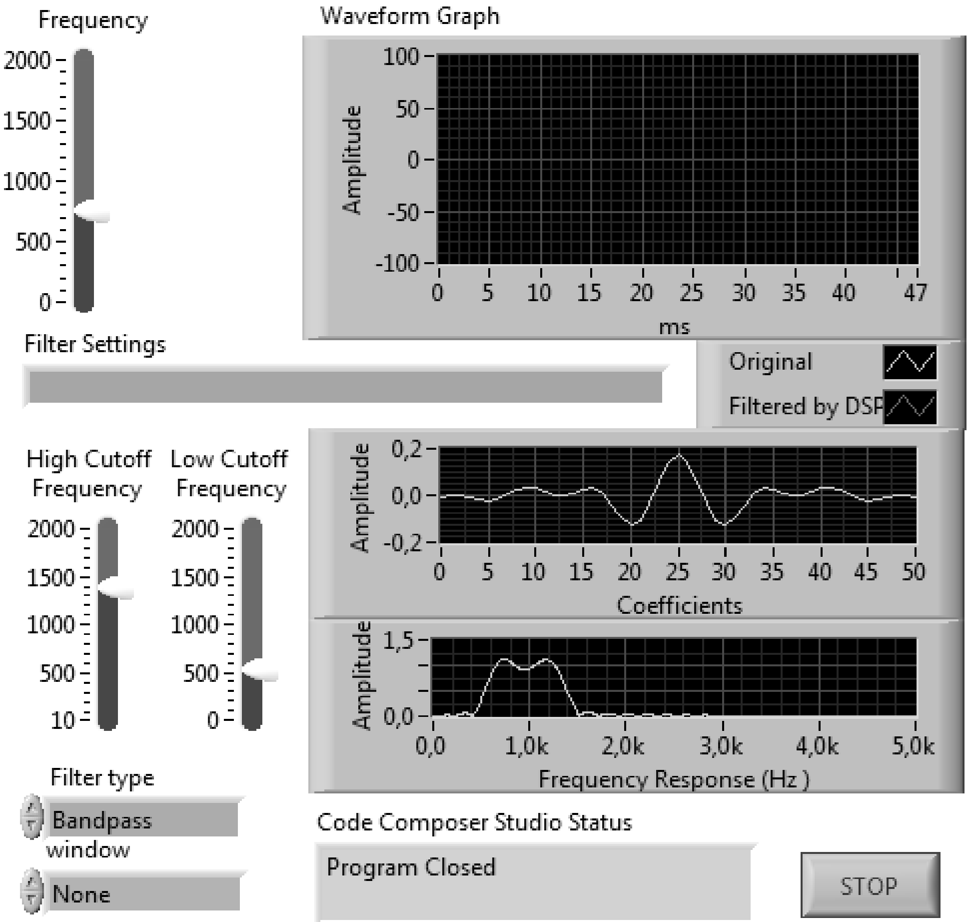


Fig. 5.1.3 Schematic of the virtual instrument in National Instruments LabVIEW 2010



Filter type: Bandpass

window: None

Code Composer Studio Status: Program Closed

Fig. 5.1.4 Appearance of the instrument control system

2. Explore the LabVIEW DSP Test Integration Toolkit. Describe the purpose and functionality of the low-level RTDX and CCS IDE components for the DSP Test Integration Advanced VI virtual interface.
3. Document the frequency charts and characteristics recorded in the experiments for the report. To formulate in the report on the completed work the conclusions of the research results and to prepare the answers to the control questions.

Methodical instructions

1. To perform this laboratory work, you must use the controls National Instruments LabVIEW 2010, shown in Fig. 5.1.3 and 5.1.4. Set teacher-defined filter settings:

Amplitude (A) = 0.50V;

Low Cut-off Frequency (LF) = 500Hz;

High Cut-off Frequency (HF) = 1e + 03Hz;

2. Use digital filter control software using DSP Test Integration Advanced VI RTDX and CCS IDE components when developing software in Code Composer Studio microprocessor programming environment:

```
// Initialize variables and RTDX channels
            #define BUFFER_SIZE 48
            RTDX_CreateInputChannel(cinput);
            RTDX_CreateInputChannel(ccoefs);
            RTDX_CreateOutputChannel(coutput);
            int input[BUFFER_SIZE], output[BUFFER_SIZE];
// Activation of RTDX channels
            RTDX_enableInput(&cinput);
            RTDX_enableInput(&ccoefs);
            RTDX_enableOutput(&coutput);
```

1. To implement reading / writing of data from digital channels with components of DSP Test Integration Advanced VI at RTDX and CCS IDE level in Code Composer Studio microprocessor programming environment:

```
int i;
double filtered;
// Read-write to RTDX feeds
for ( ; ; )
{
      /* Wait for new buffer */
      while(!RTDX_read(&cinput, input, sizeof(input)));
      /* See if there are new coefficients to be read */
      if (!RTDX_channelBusy(&ccoefs))
      RTDX_readNB(&ccoefs, &gFIRCoefficients, sizeof(gFIRCoefficients));
      for(i=0; i<BUFFER_SIZE; i++)
      {
```

```
            filtered = BandpassFilter (input[i]*gain, kTAPS, gFIRHistory, gFIRCoefficients);
            output[i] = (int)(filtered + 0.5); // round to nearest
            }
            RTDX_write(&coutput, &output, sizeof(output));
    }
```

2. Implement the digital bandwidth filter LPF and HPF in the programming environment of microprocessors Code Composer Studio:

```
// The simple Band-Pass Filter
double BandpassFilter (double val, int nTaps, double* history, double* coefs)
{
double temp, filtered_val, hist_elt;
int i;
hist_elt = val;
filtered_val = 0.0;
        for (i = 0; i<nTaps; i++)
        {
        temp = history[i];
        filtered_val += hist_elt * coefs[i];
        history[i] = hist_elt;
        hist_elt = temp;
        }
return filtered_val;
}
```

Or implement a Bessel digital bandpass filter set by the teacher of the filter order (n) in the Code Composer Studio microprocessor programming environment, whose coefficients (a, b) are calculated by the student independently:

```
// Bessel Filter Coefficients for n = 6
int npoints, index, j, k1, k2, xr, tr;
int n, n_index, n_dat, n_teil, n_fil, data;
float daten[256], koef[4][3], a[4][3], b[4][3];
```

```
float spei_x[4][3], spei_y[4][3], x[256], y[256], xtime[256];

void BesselFilter (void)
{
for (j=1;j<a_file;j++)
{
xtime[j]= j/(a_file);
x[j]= exp(-cos(j)/sin(a_file))+exp(cos(j)/sin(a_file));
}
/* n=6, F1=48Hz */
n = 6;
/* F2=1000Hz */
/* Coefficient a: */
a[1][0]=0.927303;  a[1][1]=1.854607;  a[1][2]=0.927303;
a[2][0]=0.941331;  a[2][1]=1.882663;  a[2][2]=0.941331;
a[3][0]=0.967707;  a[3][1]=1.935413;  a[3][2]=0.967707;
/* Coefficient b: */
b[1][0]=1.0;       b[1][1]=1.851753;   b[1][2]=0.857460;
b[2][0]=1.0;       b[2][1]=1.880051;   b[2][2]=0.885275;
b[3][0]=1.0;       b[3][1]=1.933302;   b[3][2]=0.937524;
for(n_teil=1;n_teil<(int)(n|2);n_teil++)
{
        for(n_fil=1;n_fil<2;n_fil++)
        {
        spei_x[n_teil][n_fil]=0;
        spei_y[n_teil][n_fil]=0;
        }
}
for(n_dat=1;n_dat<a_file;n_dat++)
{
```

```
    spei_x[1][0]=x[n_dat];
    for(n_teil=1;n_teil<(int)(n|2);n_teil++)
    {
    spei_y[n_teil][0]=spei_x[n_teil][0]*a[n_teil][0]+
        spei_x[n_teil][1]*a[n_teil][1]+
        spei_x[n_teil][2]*a[n_teil][2];
    spei_y[n_teil][0]=spei_y[n_teil][0]*b[n_teil][0]-
            spei_y[n_teil][1]*b[n_teil][1]-
            spei_y[n_teil][2]*b[n_teil][2];
    spei_x[n_teil][2]=spei_x[n_teil][1];
    spei_x[n_teil][1]=spei_x[n_teil][0];
    if(n_teil<(int)(n|2))
    {
    spei_x[n_teil][0]=-spei_y[n_teil][0]; }
    spei_y[n_teil][2]=spei_y[n_teil][1];
    spei_y[n_teil][1]=spei_y[n_teil][0];
    }
    y[n_dat]=spei_y[1][0];
    }
    for(n_index=1;n_index<a_file;n_index++)
    {
     x[n_index]=y[n_index-1];
    }
    for(index=30;index<a_file;index++)
    {
    time[index-30], x[index];
    }
}
```

3. Download software for Texas Instruments DSK6400 microcontroller developed in the Code Composer Studio programming environment and compiled into the project:

Project (PJT) - example.pjt.

Use the Waveform Graph tools to get the characteristics of a virtual device.

Control questions

1. 1. What types of elements can be distinguished within the telemedicine system?
2. 2. What are the components of DSP Test Integration Advanced VI?
3. 3. What considerations is the Texas Instruments DSK6400 microcontroller?
4. 4. Explain the principles for managing RTDX and CCS IDE data channels.
5. 5. Explain the procedure for initializing variables and RTDX channels.
6. 6. Explain the operation of the CCS RTDX Read Array DBL component.
7. 7. Features of using OMAP3530 and AM3517 processors for TMS?

Literature

1. www.ti.com

2. Tartakovsky, D.F. Metrology, standardization and technical means of measurement / D.F. Tartakovsky, A.S. Hawks. - M.: Higher School, 2002. - 206 p.

3. Pukhalsky, G.I. Designing microprocessor devices: a textbook for universities / GI. Puhalsky. - St. Petersburg. : Polytechnics, 2001. - 588 p.

4. Fomichev, AV Digital integrated circuits. Reference manual for information technology support for the design of radioelectronic systems / A.V. Fomichev, V.M. Stroev, AN Winds. - M.: Mechanical Engineering, 1999. - 123 p.

5. Ryabenky VM, Zhuikov V.Ya., Guliy VD Digital Circuitry: Educ. manual. - Lviv: New World 2000, 2009.-736 p.

6. Technology of production of electronic equipment / S.V. Frolov et al. - Tambov: Publishing House of the State Higher Education Institute of the TSTU, 2010. - 96 p.

7. OM Lebedev, OI Ladik. Digital technology. - K .: Polytechnic Publishing House, 2004.