DOI: 10.2507/29th.daaam.proceedings.031

MICROCONTROLLER BASED CONTROL SYSTEM WITH FLEXIBLE CONFIGURATION

Petr Dostálek, Vladimír Vašek & Jan Dolinay

This Publication has to be referred as: Dostalek, P[etr]; Vasek, V[ladimir] & Dolinay, J[an] (2018). Microcontroller Based Control System with Flexible Configuration, Proceedings of the 29th DAAAM International Symposium, pp.0215-0224, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-20-4, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/29th.daaam.proceedings.031

Abstract

The paper is aimed at description of the developed microcontroller based control system with flexible configuration allowing very simple implementation in many control tasks. It consist of the central module equipped with 32bit microcontroller MK60DN512 by NXP Semiconductor and input / output modules for interfacing central module to the controlled process. Central module consists of two boards connected together – mainboard and microcontroller daughter board. Daughter board provides all the necessary microcontroller's signals on the two 2-row 40pin headers and complete Ethernet communication interface. Mainboard provides to the daughter board 5V stabilized power supply and all necessary peripherals – SDHC card slot, RS232 and RS485 communication interfaces. RS232 is used for communication with smart sensors for instance; RS485 is especially used for high speed interconnection with up to 15 expansion peripheral modules. Control system is programmable in C language using any compatible IDE – NXP Kinetis Design Studio, for example. For this purpose support program libraries including necessary routines for control and monitoring tasks were created.

Keywords: process control; microcontroller; embedded system; ethernet

1. Introduction

Present-day very rapid progress in electronics and computer science influences all areas of human activities. Production technology improvements of new microcontrollers lead to their miniaturization, increased central processor unit performance, decreased power consumption and price. Thus modern 8bit one-chip microcontrollers have enough computing power for process measurement, monitoring [1] and control tasks with utilization of fixed parameters controllers like PS or PSD. With some code optimization they are able to handle tasks from the origin of modern control methods such as adaptive control too [2]. However the second mentioned area is on the memory and computational power limit of most 8bit microcontrollers for more complex algorithms. New possibilities in this area are brought by modern 32bit ARM core based microcontrollers which can provide Ethernet, USB and other interfaces for communication and data storage enabling data presentation using simple web server, data logging to SD card and many other useful functionality.

Due to some limitations, mainly in main memory capacity and computational power in double precision floating point format, microcontrollers cannot fully substitute powerful industrial PCs or special programmable logic controllers, which can work with number of control loops simultaneously and realize highly sophisticated control algorithms. However, area of microcontroller usage is a bit different – in embedded systems, that is, in systems where the emphasis is on low price, compact dimensions, low power consumption, high reliability and immunity against environmental influences and other specific requirements.

This work presents hardware design of a modular control system intended for embedded applications. The main idea is to combine advantageous properties of modular programmable logic controllers (configuration variability perfectly meets application requirements) and embedded systems (freely programmable in C language, very low cost). To fulfil main requirements of high computational power, Ethernet communication capabilities and low cost of the whole control system 32bit Kinetis microcontroller MK60DN512 with ARM Cortex-M4 core by NXP Semiconductor was chosen as a main part of the central unit.

First section of the paper describes hardware design of all the components of the control system – central unit, digital input / output and analog input / output extension modules representing basic configuration of the system. Last chapter deals with software support of the device which is necessary for effective programming of control and monitoring applications.

2. Hardware design

Control system design is adapted to easy functionality modification on the basis of the specific application requirements – for example number of required digital and analog inputs / outputs, range and type of input signal, voltage and current capacity of digital outputs and so on. This is achieved by splitting the whole control system to the two main components:

- Central unit consisting of mainboard and microcontroller daughter board.
- Expansion modules which are connected with central unit using high speed RS485 communication bus.

Control system structure is depicted in the Figure 1.

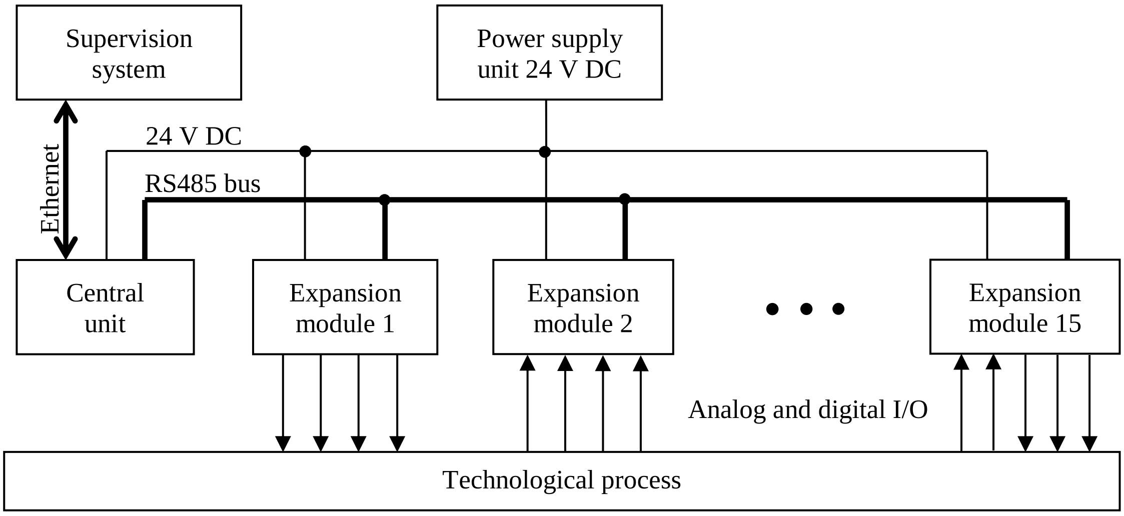

Fig. 1. Control system structure

2.1. Control system central unit

Central unit of the control system is based on low power 32bit microcontroller NXP Semiconductors MK60DN512 with ARM Cortex-M4 core with DSP instructions providing computational power of 1.25 Dhrystone MIPS per MHz. These microcontrollers based on 32bit Armv7E-M Harvard architecture are suitable for applications requiring fast and power efficient digital signal processing such as motor control, automotive applications, power control, industry automation and sound processing. MK60DN512 microcontrollers can operate with supply voltage of 1.71 to 3.6 V and wide range of ambient temperature from -40 to 105 °C. They have integrated on the chip 512 KiB of program FLASH memory, 128 KiB RAM memory and many peripherals such as:

- Two 16bit A/D converters
- 12bit D/A converter
- Three analog comparators
- 16bit timer subsystem with PWM, quadrature encoder, periodic interrupts generation
- Real-time clock

- Ethernet controller with MII and RMII interface to external PHY
- USB interface full/low speed
- Two CAN modules
- Three serial peripheral interfaces (SPI)
- Two I2C modules
- Five asynchronous serial communication interfaces (UART)
- Secure Digital Host controller (SDHC)
- 66 general purpose I/O pins [3]

Central unit consists of two separable printed circuit boards: microcontroller daughter board and mainboard connected together using two 2-row 40 pin headers. Block schematic of the unit is depicted in the Figure 2 and electronics schematic of microcontroller daughter board in the Figure 3.

Daughter board contain main microcontroller MK60DN512 connected via RMII interface with Ethernet PHY DP83848 by Texas Instruments supporting auto-MDIX feature for communication speed of 10/100 Mb/s [4]. Status of the interface – link good, speed and interface communication activity is indicated by three LEDs. Clock source for the PHY in RMII mode must provide 50 MHz reference clock signal with frequency tolerance and stability of ±50 ppm to the X1 pin. This requirement fulfils for example crystal clock oscillator ACHL-50.000MHz-EK which is currently used in the circuit. Clock output pin named 25MHz_OUT (in RMII mode provides 50 MHz) is connected to clock input pin EXTAL0 of the main microcontroller. TD+, TD- and RD+, RD- signals of the PHY PMD interface are connected to LAN magnetics module (transformer) B78476A8065A003 with 1:1 turn ratio and inductance of min. 350 µH followed by standard RJ-45 connector. Reference voltage of 2.5 V for A/D converter is provided by precision voltage reference REF03 by Analog Devices and connected to pin VREFH. Internal real-time clock is backed up when system power is down by lithium cell CR2032 with nominal voltage of 3 V. Microcontroller programming and application debugging can be done through 10 pin Kinetis debug port using compatible debug communication interface. Last part of the microcontroller daughter board is power supply circuit based on low drop linear voltage regulator AP1117D with output voltage of 3.3 V and 1 A current capability.

The purpose of the mainboard is to provide to microcontroller daughter board stabilized 5 V supply voltage and necessary communication interfaces for the whole system functionality. Power supply was designed with respect to minimization of power losses as a step-down DC/DC converter based on controller integrated circuit MC34063. On its output is generated stabilized DC voltage of 5 V which is in the next stage regulated to 3.3 V using linear voltage regulator AP1117D. Input supply voltage of the central module can be due to utilization of step-down converter in the wide range from 12 V to 40 V DC. Nominal supply voltage of control system is 24 V DC. Physical layer of the RS485 communication interface is implemented by RS485 transceiver SN65HVD10 operating at 3.3V supply voltage which supports communication speed up to 32 Mb/s. Bus connection pins A and B are on the chip internally protected against ESD and short circuit from -7 V to 12 V. Serial asynchronous interface RS232 utilizes standard voltage level converter MAX3232 which can operate with supply voltage in range from 3.0 V to 5.5 V. It supports communication speed up to 120 kb/s while maintaining RS232 output levels. Integrated dual charge pump for its function requires four external 100nF capacitors. Mainboard is also equipped with slot for SD or SDHC card which can be used for data storage purposes. Maximum number of expansion modules connected to central unit is 15.

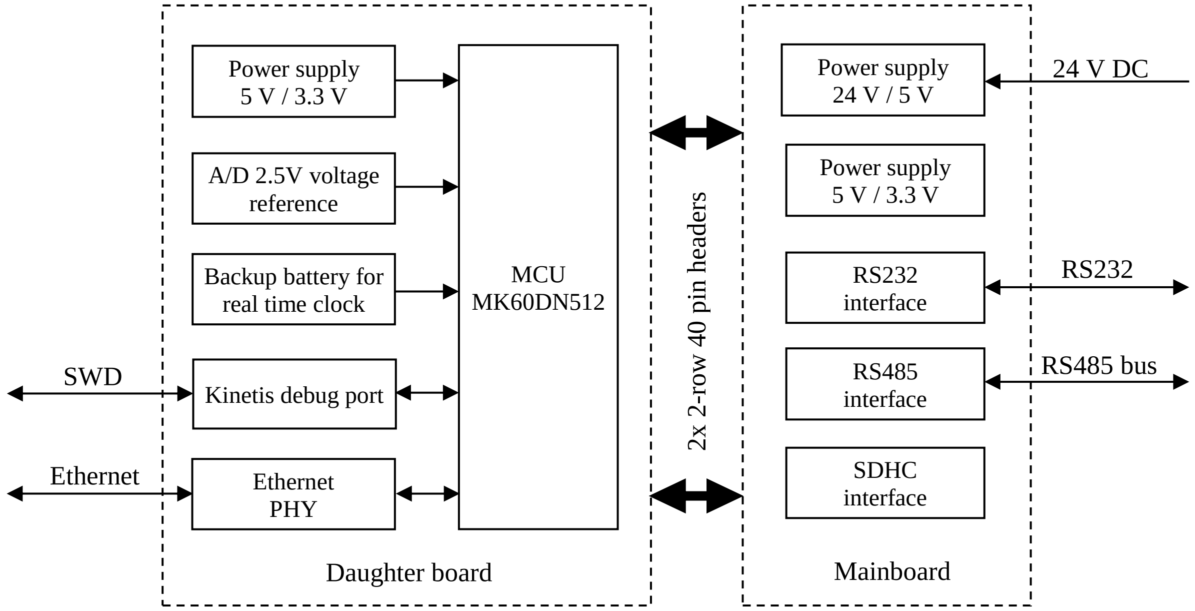

Fig. 2. Central unit block schematics

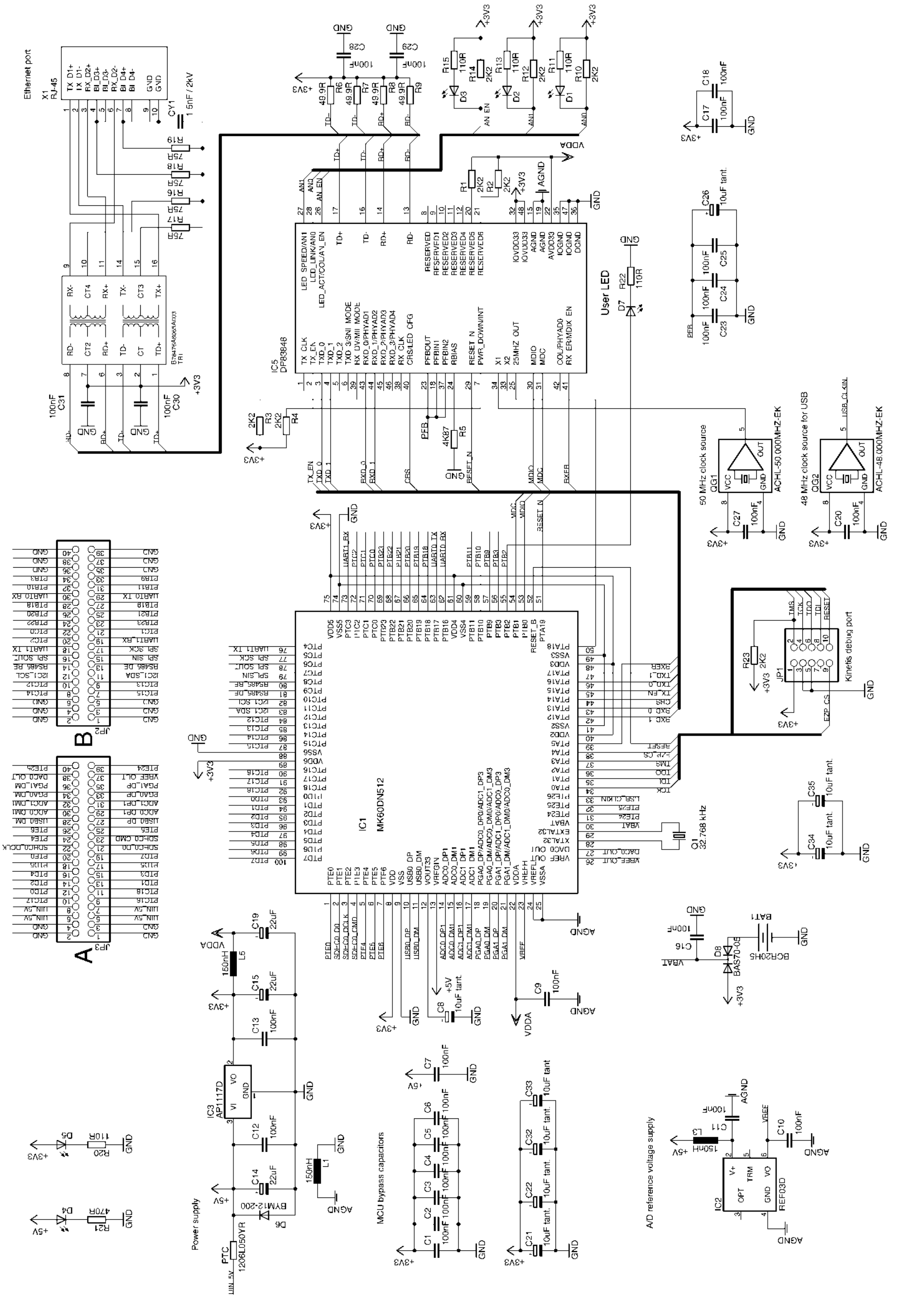

Fig. 3. Microcontroller daughter board schematics

2.2. Analog input / output expansion module

Analog input / output expansion module is based on 8bit microcontroller NXP MC9S08QE16 which was chosen due to integrated 12bit A/D converter [5]. Input unified voltage signal with range from 0 V to 10 V first enters to resistor divider which adapts it to 0 V to 3.1 V range suitable for A/D converter of the microcontroller. After voltage divider follows non-inverting amplifier with gain set to 1 and simple first order low-pass filter for high frequency ripple rejection. Then the signal enters to the corresponding input channel of the A/D converter. Analog outputs are implemented using two 2channel 12bit D/A converters Microchip Technology MCP4822 [6] which are connected with microcontroller via serial peripheral interface (SPI). Analog signal from the converter is amplified using non-inverted amplifier with gain of 4.9 to achieve unified output voltage signal in range from 0 V to 10 V. RS485 communication interface utilizes same bus transceiver type SN65HVD10 as central unit. Firmware of the microcontroller can be updated via 6pin BDM interface. Input supply voltage with nominal value of 24 V is converted to 5 V by DC-DC converter Mean Well SMU01N-05 with power of 1 W. Because the digital circuits work with 3.3 V it is necessary to lower 5 V output from converter with low-drop linear voltage regulator AP1117D. Module address is selected in binary form by DIP switch with four positions. Valid address range is from 1 to 15 (address 0 is reserved for central unit only). Schematic of the module is in the Figure 4.

Basic module properties are:

- 8 analog inputs with voltage range 0 to 10 V
- 4 analog outputs with voltage range 0 to 10 V
- 12bit resolution of A/D converter
- 12bit resolution of D/A converter
- RS485 communication interface for connection with central unit at speed of 625 kb/s
- 24 V DC supply voltage

Module implements 14 registers which are accessible from RS485 communication bus by central unit using specified binary communication protocol. Each register is 16bit length and can be accessed in read / write modes depending on its particular function. Complete register map is provided in the Table 1.

Address	Access	Description	Register HB	Register LB
0	R	Firmware version and module ID	Firmware version	Module ID
1	R	AI0 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
2	R	AI1 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
3	R	AI2 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
4	R	AI3 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
5	R	AI4 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
6	R	AI5 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
7	R	AI6 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
8	R	AI7 data register	A/D conversion result value 0 – 4095 represents input voltage 0 – 10 V	
9	RW	AO0 data register	Value for D/A conversion 0 – 4095 represents output voltage 0 – 10 V	
10	RW	AO1 data register	Value for D/A conversion 0 – 4095 represents output voltage 0 – 10 V	
11	RW	AO2 data register	Value for D/A conversion 0 – 4095 represents output voltage 0 – 10 V	
12	RW	AO3 data register	Value for D/A conversion 0 – 4095 represents output voltage 0 – 10 V	
13	RW	AI configuration register	Not used	Digital filter length (0, 2, 4, 8, 16 samples)

Table 1. Analog input / output module register map

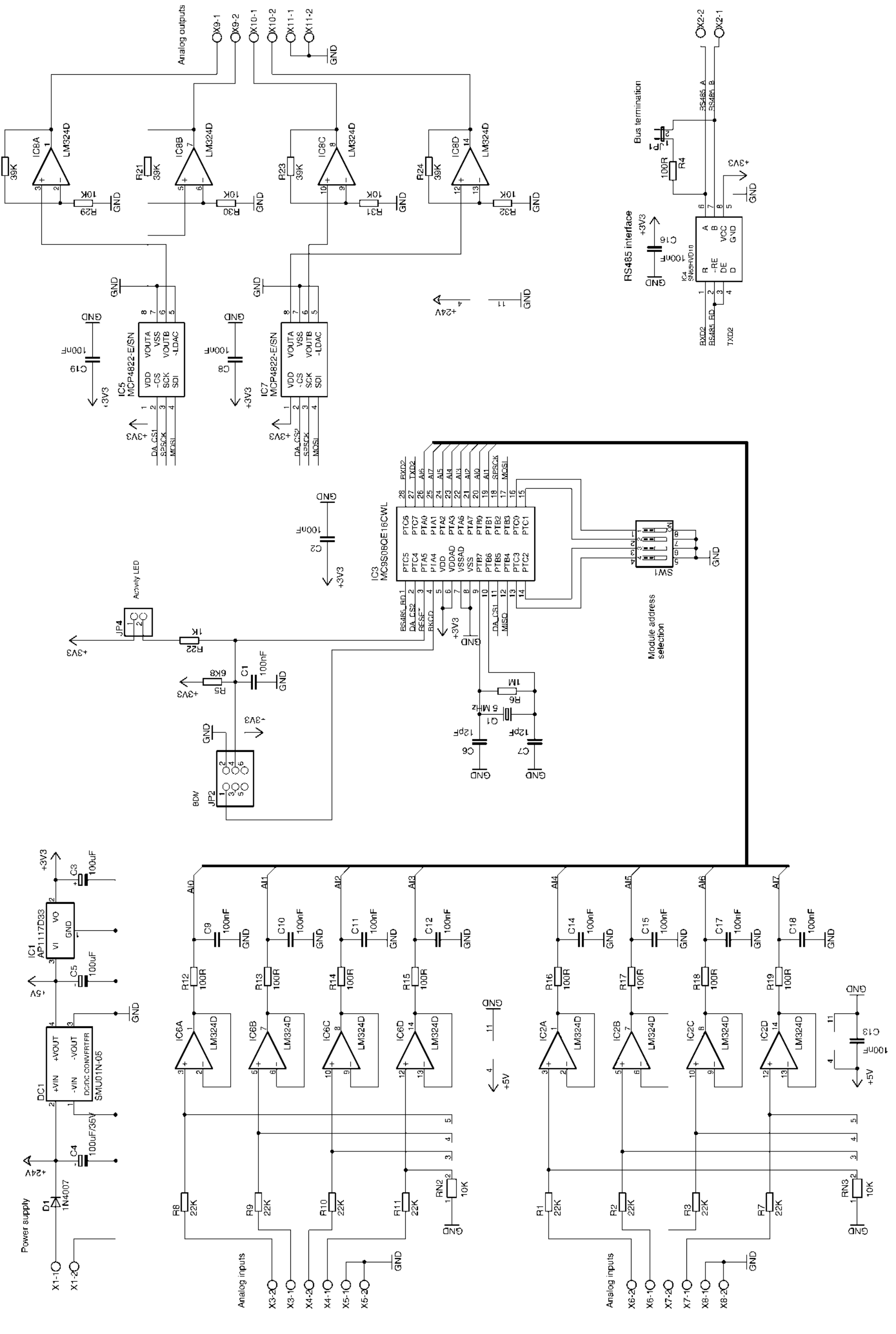

Fig. 4. Analog input / output module schematics

2.3. Digital input / output expansion module

Digital input / output expansion module is based on microcontroller MC9S08SH32 by NXP Semiconductor. It is 8bit microcontroller with 32 KB of program FLASH memory and 1 KB of RAM memory. On the chip the following peripherals are integrated: 10bit A/D converter, analog comparator, serial communication interface (SCI), serial peripheral interface (SPI), I2C, timer subsystem, real-time counter and others [7].

Digital inputs of the module are galvanically isolated by two 4channel optocouplers TLP283-4 with output electrooptical transistor. Input voltage for logical 1 should be in wide range from 3.3 V to 24 V providing compatibility with both TTL and relay logic. Digital outputs of the microcontroller are connected with integrated Darlington transistor array ULN2803A which makes possible directly switch loads with total current up to 500 mA – for example 24V relays. Maximum voltage which can be connected to the outputs is 50 V DC. Communication interface and power supply is designed identically as in analog I/O module. Schematic of the module is in the Figure 5.

Basic module properties are:

- 8 galvanically separated digital inputs
- 8 digital outputs with open collector (NPN type)
- PWM signal generation support on all digital outputs
- Pulse counting support on digital inputs
- RS485 communication interface for connection with central unit at speed of 625 kb/s
- Input voltage range 3.3 V to 24 V for logical 1 state
- Maximum output voltage of 50 V DC
- Total output current of 500 mA
- 24 V DC supply voltage

Module implements 13 registers which are accessible from RS485 communication bus by central unit using specified binary communication protocol. Each register is 16bit length and can be accessed in read / write modes depending on its particular function. Complete register map is provided in the Table 2.

Address	Access	Description	Register HB	Register LB
0	R	Firmware version and module ID	Firmware version	Module ID
1	RW	Digital outputs data register	Not used	DO port state
2	R	Digital inputs data register	Not used	DI port state
3	RW	PWM CH0 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
4	RW	PWM CH1 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
5	RW	PWM CH2 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
6	RW	PWM CH3 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
7	RW	PWM CH4 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
8	RW	PWM CH5 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
9	RW	PWM CH6 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
10	RW	PWM CH7 duty cycle (8bit resolution)	Not used	0 = 0% 255 = 100%
11	RW	PWM period in milliseconds	Allowed value range 500 – 65535 ms	
12	RW	PWM channel enable	Not used	0 in mask = disabled 1 in mask = enabled

Table 2. Digital input / output module register map

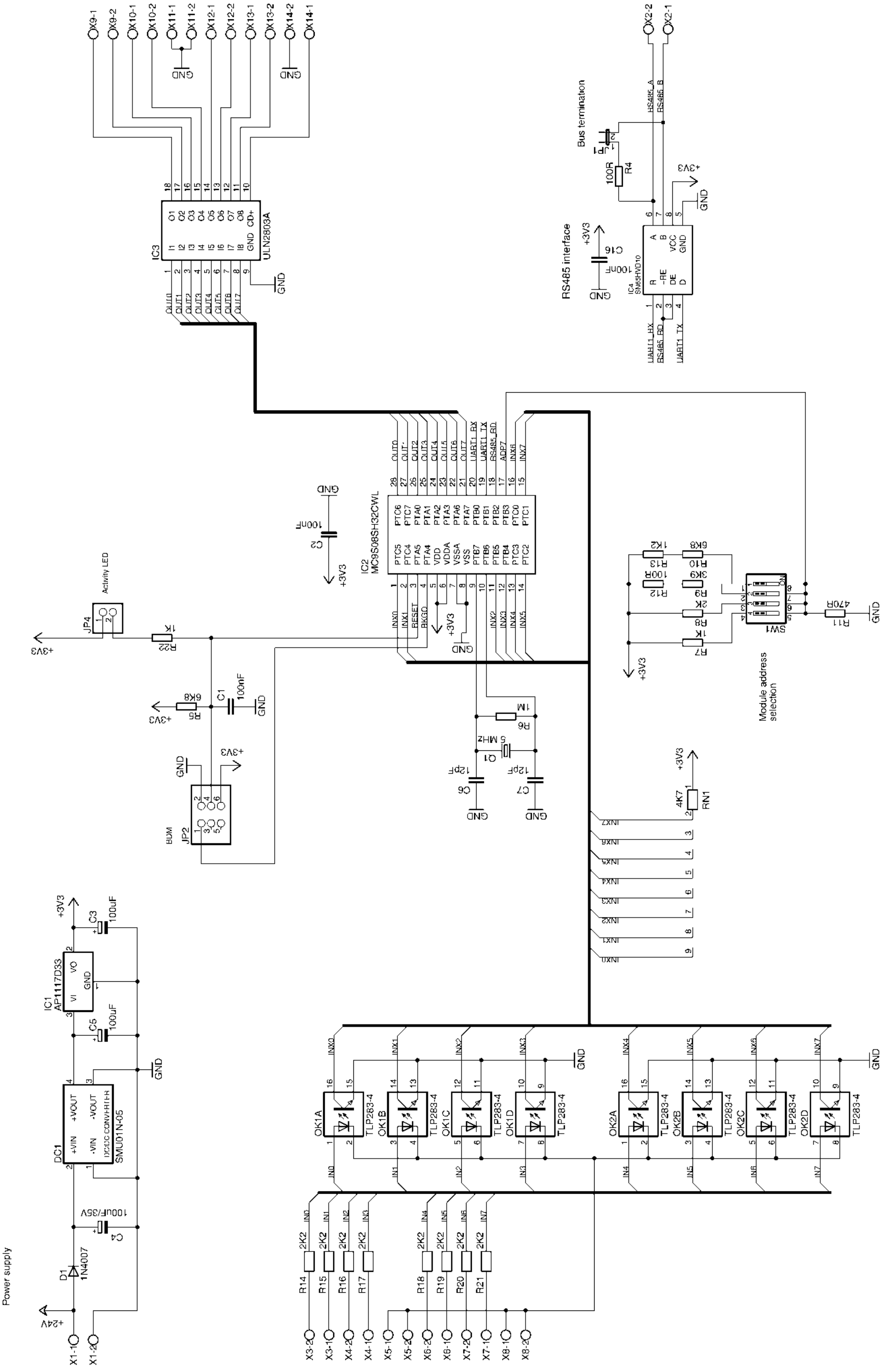

Fig. 5. Digital input / output module schematics

3. Communication protocol

Communication protocol between central unit and expansion modules connected on RS485 bus was designed mainly with respect to very fast response time. Due to this fact binary communication protocol was chosen. All data traffic on the bus is controlled by the central unit (master). Expansion peripheral modules (slaves) can transmit data only on request from the master. Each slave device is recognized by its address which must be unique on the bus. It can be set by the 4-position DIP switch in the range of 1 to 15. Master device has always assigned address 0 and it is not normally possible to change it by user. Protocol implements two special characters which controls frame processing. It is start character with code 0xAA and prefix character with code 0xFF. Prefix character is used for encoding values of 0xAA and 0xFF in the data section of the frame.

3.1. Master device frame description

Master device frame consists of five main parts: start character, module address, command, data and CRC8 check characters. Each frame must start with the special character 0xAA which is called start character. Its purpose is to synchronize all slave devices to listen to new communication frame. This character may not occur in other parts of the frame so special prefix code must be used if it is a part of the data or CRC8 section for example. Next follows 1 byte length modules address field which selects individual slave device for which is the frame intended. After that is command part with size of 1 byte indicating to slave device what operation to do. Depending on command type follows data and CRC8 fields which can be completely omitted when frame is without data. Structure of the frame transmitted by master device is depicted in the Figure 6.

Start character	**Module address**	**Command**	**Data**	**CRC8**
0xAA	1 byte <1; 15>	1 byte	N bytes (max. 15)	1 byte

Fig. 6. Master device frame structure

3.2. Slave device frame description

Slave device can respond to the master by two types of frames: command acknowledge response and error response. Both types of responses start with the special start character 0xAA followed by status byte indicating successful or unsuccessful command processing. In the case of error response status byte contain error code with reserved values in the range of 0xF0 to 0xFE so it can distinguish up to 15 error codes. Command acknowledge response has status byte value in the range of 0x10 to 0x2F where value of 0x10 indicates response without data and CRC8 part, value 0x11 means 1 byte of data with CRC8 and so on. Structure of both types of frames transmitted by slave device is depicted in the Figure 7.

Command acknowledge response			
Start character	**Status byte**	**Data**	**CRC8**
0xAA	1 byte	N bytes (max. 31)	1 byte

Error response	
Start character	**Status byte**
0xAA	1 byte

Fig. 7. Slave device frames structure

4. Control system software implementation

The software for the control system is based on the library created at our institute, named UCP [8]. This library was designed with three main requirements in mind: provide discrete controllers usable in many common microcontroller applications, easy to use programming interface and easily portable to different microcontrollers.

4.1. Controller modules

Controller modules are the fundamental part of the software. By module we mean a set of functions and the data used by those functions enclosed into a source and header file. From the logical point of view a module is one type of a controller, e.g. a discrete PID controller.

Typically it contains two C functions: one function which takes care of the initialization of the controller and one function which calculates the output of the controller in each step.

4.2. User application templates

The library also recommends a preferred way of using the controllers. This is accomplished by template files, which contain skeleton code of a control application. The user is advised to include these template files in his/her program and implement the application-specific and hardware-specific code as outlined in the templates. Example applications are also provided which show how to implement this code.

However, the user is not required to use this application template or any other particular style of programming. He is free to use any part of the library separately, e.g. just the controller module(s), which are C functions and therefore can be simply called from any C program.

4.3. Utility code

There is also utility code, which was developed during the development of the library for testing its functions. An example of such a code is a driver for serial communication interface (UART). The utility code can be used directly in user applications, or it may provide starting point and working examples of the code which will be very likely needed when using the library.

5. Conclusion

Paper presents design of a universal control system with flexible configuration intended for embedded applications demanding high computing power, wide communication possibilities and configuration flexibility while preserving low cost. These requirements are fulfilled by a system consisting of a central unit and expansion peripheral modules which number and type depends on target application. Developed prototype control system central unit is based on 32bit microcontroller NXP MK60DN512 with ARM Cortex-M4 core including DSP instructions. It is a part of a small daughter board including Ethernet communication interface and two 2-row 40 pin headers for connection with mainboard. The purpose of the mainboard is to provide to daughter board stabilized 5 V supply voltage and the required communication interfaces (RS232, RS485, SD card slot) for the whole system functionality. The RS485 interface is used for high speed interconnection with up to 15 expansion peripheral modules. Two types of expansion modules are now available: analog input / output module, digital input / output module which can cover most process control applications. Software development and debugging is simplified by our support program libraries named UCP which include various modules for control and monitoring tasks.

In the future development we will focus on expanding of number of available different types peripheral modules and from software point of view to communication software stack for Ethernet interface to improve visualisation and data processing capabilities with minimum impact to control unit computational performance.

6. Acknowledgments

This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic within the National Sustainability Programme project No. LO1303 (MSMT-7778/2014) and also by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

7. References

[1] Masic, A; Pikula, B & Bibic, D. (2017). Mobile Measurements of Particulate Matter Concentrations in Urban Area, Proceedings of the 28th DAAAM International Symposium, pp.0452-0456, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-11-2, ISSN 1726-9679, Vienna, Austria, DOI: 10.2507/ 28th.daaam.proceedings.063

[2] Dostálek, P.; Dolinay, J.; Vašek V. & Pekař L. (2010). Self-tuning Digital PID Controller Implemented on 8-bit Freescale Microcontroller. International Journal of Mathematical Models and Methods in Applied Science, Vol. 4, No. 4, 274-281, ISSN 1998-0140

[3] http://www.nxp.com, (2012). NXP Semiconductors, K60 Sub-Family Reference Manual, Accessed on: 2018-08-14

[4] http://www.ti.com, (2007). Texas Instruments, DP83848C/I/VYB/YB PHYTER™ QFP Single Port 10/100 Mb/s Ethernet Physical Layer Transceiver, Accessed on: 2018-08-17

[5] http://www.nxp.com, (2011). NXP Semiconductors, MC9S08QE16 Reference Manual, Accessed on: 2018-08-14

[6] http://www.microchip.com, (2010). Microchip Technology, 12-Bit Dual Voltage Output Digital-to-Analog Converter, Accessed on: 2018-08-17

[7] http://www.nxp.com, (2014). NXP Semiconductors, MC9S08SH32 MC9S08SH16 Data Sheet, Accessed on: 2018-08-14

[8] Dolinay, J.; Dostálek, P. & Vašek V. (2014). Program modules for control applications of microcontrollers. Proceeding of Latest Trends on Systems, Santorini Island, ISSN 1790-5117, ISBN 978-1-61804-244-6, pp. 488-491, Europment, Rhodes