



Европейски съюз

Проектът се осъществява с финансовата подкрепа на
Оперативна Програма „Развитие на Човешките Ресурси” 2007 – 2013,
Съфинансиран от Европейския Социален Фонд на Европейския Съюз
Инвестира във вашето бъдеще!



ЛЯТНА ШКОЛА 2013

СПЕЦИАЛИЗИРАНИ ИНТЕЛИГЕНТНИ КОНТРОЛЕРИ

Гл. ас. инж. Емил Симеонов Петров

**ПОВИШАВАНЕ НА ЕФЕКТИВНОСТТА И КАЧЕСТВОТО НА
ОБУЧЕНИЕ И НА НАУЧНИЯ ПОТЕНЦИАЛ В ОБЛАСТТА НА
СИСТЕМНОТО ИНЖЕНЕРСТВО И РОБОТИКАТА**

Проект № BG051PO001-3.3.06-0002



**Българска Академия на Науките
Институт по Системно Инженерство и Роботика**



ГЛАВА ПЪРВА

СЪСТОЯНИЕ НА ПРОБЛЕМА. ЦЕЛ И ЗАДАЧИ НА ДИСЕРТАЦИОННАТА РАБОТА

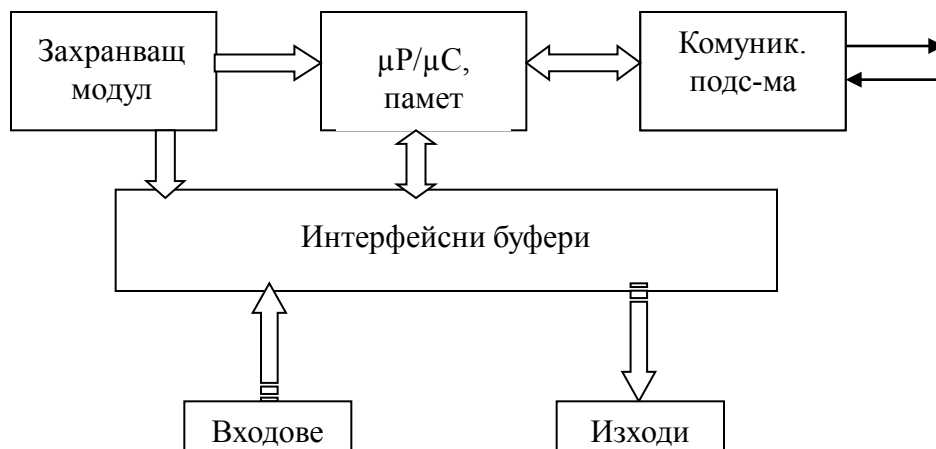
Системите за управление в повечето производствени отрасли са известни под общото наименование системи за управление на технологични процеси (Industrial Control Systems – ICS).

- Системи за събиране на данни, наблюдение и управление (Supervisory Control and Data Acquisition – SCADA);
- Разпределени системи за управление (Distributed Control Systems-DCS);
- Управляващи системни конфигурации, изградени с програмируеми логически контролери (Programmable Logic Controllers – PLC);
- Вградени системи (embedded systems).



Вградени системи. Архитектура

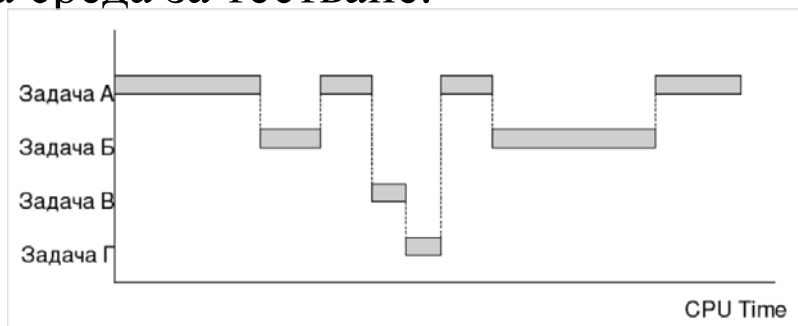
Вградените системи (ВС) са електронни цифрови системи, програмирани да изпълняват определени функции в реално време. Отличителна характеристика на вградените системи е ограниченият брой предварително дефинирани функции, които са проектирани да изпълняват.



Вградени системи. Програмно осигуряване

Основна характеристика на програмното осигуряване (ПО) на ВС е, че то трябва да работи в реално време. Основата на ПО за ВС е операционната система (ОС) за реално време.

- Администрира и предоставя достъп до ресурсите на централният процесор, паметта и периферните устройства;
- Предоставя правилата и примитиви за управление, синхронизация и комуникация между потребителските програми;
- Отговаря за ((квази)паралелното) изпълнение на множество задачи.
- Доставя на потребителя стандартизирани средства за достъп до вътрешни и външни за системата ресурси (файлови системи, комуникационни стекове и т.н.);
- Осигурява среда за тестване.



Вградени системи. Езици за програмиране

Езици от ниско ниво:

- Машинен език - трудоемък и изморителен
- Асемблер - машинно-зависим език от ниско ниво

Езици от високо ниво:

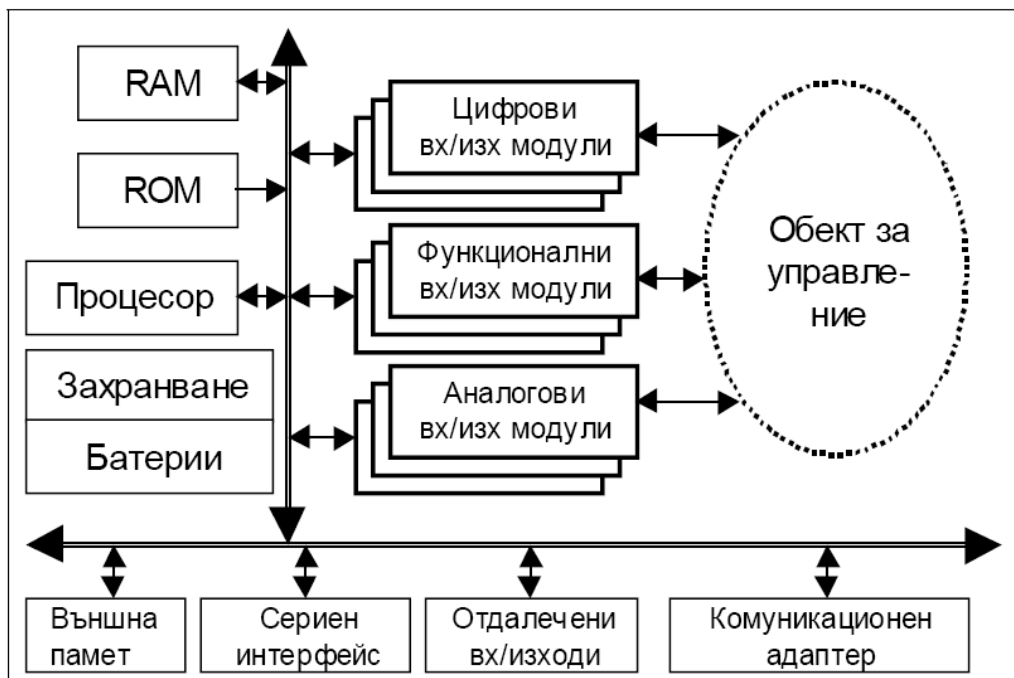
Езиците за програмиране от високо ниво позволяват значително да се съкрати времето за създаване на програмата, за сметка на увеличението на размера ѝ и намаляване на бързодействието ѝ.

- C – не предлага абстрактен модел на изчисленията, а реализира възможности, които са необходими в практиката на системното програмиране - средства за непосредствена работа с паметта, структурни конструкции на управлението и модулна организация на програмата. Всичко останало е отнесено в “runtime library”. По тези си характеристики езика C може да се сравни с макроасемблер.

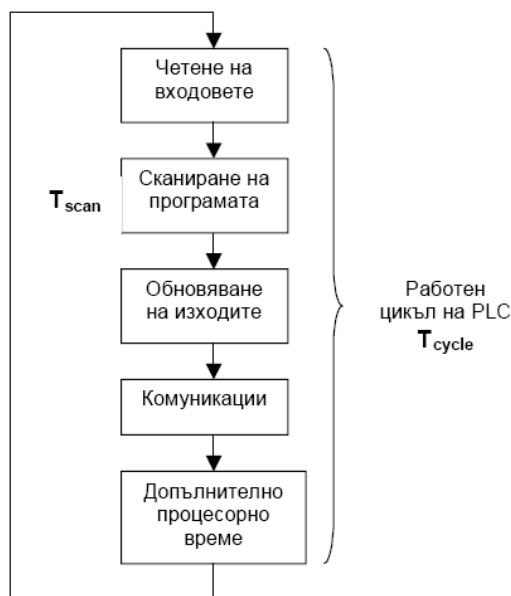
- Pascal - няма широко разпространение. Съществува търговска версия на Pascal - компилатор.

Програмируеми логически контролери. Архитектура

Програмируемите контролери (ПЛК) са микропроцесорни управляващи устройства, достъпни за програмиране от специалисти с обща инженерна подготовка и предназначени за управление на последователности, комбинаторни и непрекъснати процеси в промишлена среда и реално време.



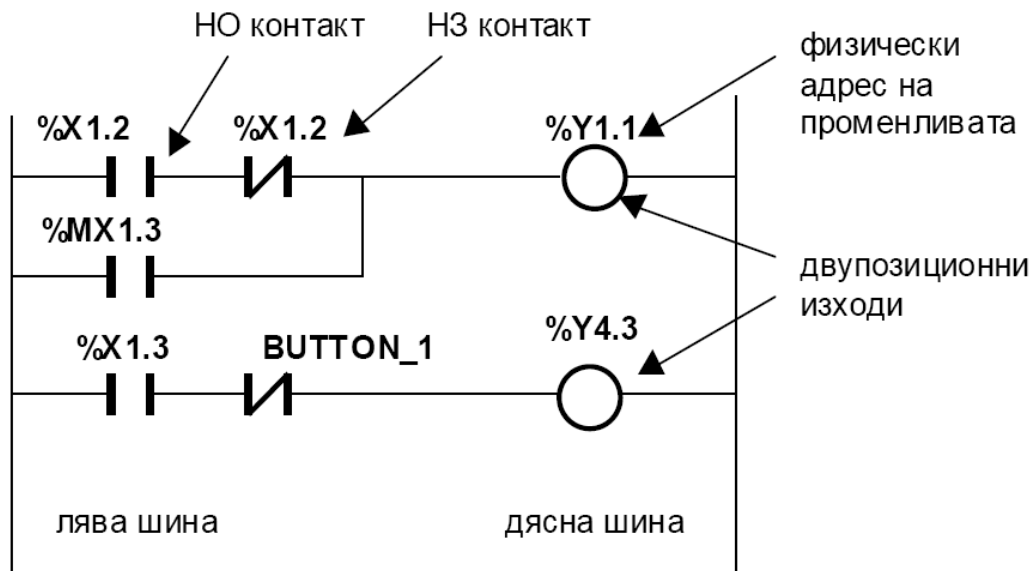
Програмируеми логически контролери. Организация на изчислителния процес



- Време на сканиране на програмата T_{scan} ;
- Общо време на цикъла T_{cycle} ;
- Време за реакция $T_{response}$ - интервал от време между промяната на един входен сигнал и промяната на свързан с него изход. Времето за реакция зависи от структурата и логиката на програмата.

Програмно осигуряване на ПЛК

- Базово ПО = СПО - програми за създаване, редактиране, транслиране на програми, изпълнение на приложните програми, самодиагностика и комуникация
- Приложното ПО – създава се от или по заявка от потребителя. От 1993 г. действащ стандарт за приложни програми на PLC е IEC 61131-3.

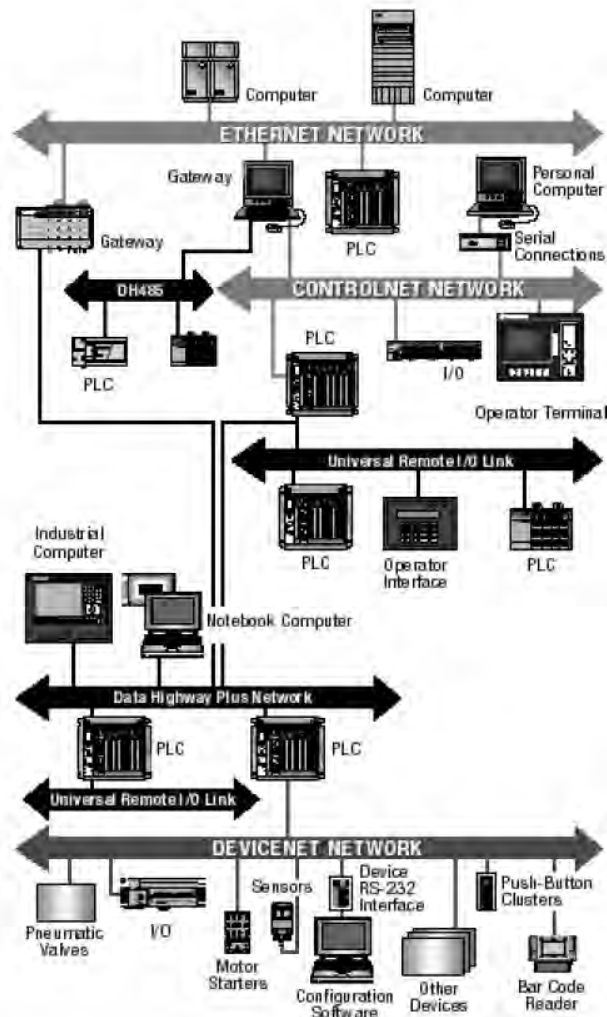


Промислени комуникационни мрежи

Ниво 1: Device Level - мрежи за комуникация на PLC с интелигентни крайни устройства - DeviceNet, LonWorks, Foundation Fieldbus, Profibus и др.

Ниво 2: Automation and Control Level. Изгражда се от PLC в локални мрежи с детерминистичен характер (например ControlNet).

Ниво 3: Information Level - за обмен на информация и управление на ниво предприятие, например Ethernet мрежа.



Изводи от обзора

- Основната причина за масовото приложение на ВС е тяхната ниска себестойност, която от своя страна се дължи на серийното им производство;
- При дребносерийно производство или при производството на уникални устройства, това предимство на ВС изчезва поради необходимостта от скъпоплатен висококвалифициран труд;
- Програмирането на ПЛК в конкретната промишлена среда от специалисти технолози е основно тяхно предимство;
- Недостатък на ПЛК е, че тяхната входно/изходна система е ориентирана към стандартизирани промишлени сигнали, както и че добавянето на допълнителен В/И интерфейс е модулно, т.е. кратно на 4, 8, 16 и т.н. вход/изхода или на отделни контролери, свързани в локална мрежа.

Актуалност на проблема: При разработката и изработката на уникални и/или дребносерийни изделия, както и при научни разработки и експерименти, проектирането и настройката на програмното осигуряване на контролерите отнема значително време. Предлаганите в дисертационния труд клас контролери заедно с методиката и средствата за изграждането на програмното им осигуряване ускоряват този процес.

Предмет на дисертационния труд: Предмет на дисертационния труд са малки контролерни системи, предназначени за управление на уникални и/или дребносерийни обекти.

Цел на дисертационния труд: Да се дефинира и определи клас малки контролерни системи – програмируеми специализирани контролери – като се разработят и апробират методика и средства за проектиране на програмното им осигуряване

Задачи на дисертационния труд:

- Изследване на организацията на изчислителния процес при ВС и ПЛК;
- Дефиниране на класа програмируеми специализирани контролери (ПСК);
- Организиране на програмното осигуряване в ПСК, като се отчитат информационните и управляващите аспекти на изчислителния процес, паралелизма му и работата в реално време;
- Създаване на методика за изграждане на системното програмно осигуряване на ПСК;
- Разработка на комуникационната подсистема на ПСК;
- Създаване на универсален език за потребителско програмиране на ПСК;
- Създаване на система за програмиране и настройка на приложното програмно осигуряване на ПСК;
- Експериментално прилагане на ПСК.

ГЛАВА ВТОРА

ПРОГРАМИРУЕМИ СПЕЦИАЛИЗИРАНИ КОНТРОЛЕРИ. МЕТОДИКА ЗА ПРОЕКТИРАНЕ НА СИСТЕМНОТО ПРОГРАМНО ОСИГУРЯВАНЕ

Дефиниция: Програмируемите специализирани контролери са малки (до 40 вход/изхода) информационно-управляващи системи с уникална апаратна част, стандартизирано системно програмно осигуряване, потребителско програмиране на процедурен език от високо ниво и възможност за многократно зареждане и настройка на технологичната програма.

- Малки контролерни системи с ограничени апаратни ресурси;
- По своята апаратна същност ПСК се идентифицират с ВС;
- По изграждането и същността на СПО те се идентифицират както с ВС, така и с ПЛК;
- По характера на разработка на приложната програма те се идентифицират с ПЛК;
- По начина на използване те се идентифицират повече като ВС, но и като ПЛК.

Структура на системното програмно осигуряване на ПСК

Модул първоначално установяване (МПУ)

Модул прекъсване (МП)

Модул главна програма (МГП)

Спомагателни програмни модули (СПМ)

Комуникационна подсистема на ПСК

Комуникационната подсистема е присъщо необходима на ПСК. Тя се използва за:

- Настройка на СПО – по RS232;
- Зареждане и настройка на ПП – по RS232;
- Връзка на ПСК с горно ниво, реализирано върху интелигентен терминал или персонален компютър, осигуряващо визуализация, архивиране, протоколиране и т.н на процеса. Комуникацията може да се осъществи както по RS232, така и по RS485;
- Връзка в по-голяма система с други ПСК и с горно ниво – комуникацията е по RS485.

Методика на изграждане на СПО за ПСК

Като принцип за изграждането на СПО на ПСК е прието модулното програмиране. Проектирането на СПО е разделено на пет логически обособени етапа, които се разработват последователно.

- **Етап 1** – въвеждат се основни модули в МПУ, МП и МГП – инициализация на портовете и USART, прекъсване по вътрешен таймер, въвеждане на системния протокол DPCP в МП, “празен цикъл” в МГП;
- **Етап 2** – добавят се необходимите драйвери и програмни модули за останалата апаратна част;
- **Етап 3** – подготвят се всички необходими вътрешносистемни информационни комуникации, така че да има еднозначно и обратимо съответствие между системни и потребителски данни. Извършва се окончателното разпределение на RAM паметта и се обособяват логически системната и потребителската ѝ част;
- **Етап 4** – подготвят се всички модули и пойнтери, осигуряващи зареждането и изпълнението на ПП;
- **Етап 5** - имплементира се мрежовия протокол DCCB и необходимите условия за избор на протокол.

Изводи

- Дефинираният клас програмируеми специализирани контролери съчетава голяма част от предимствата на вградените системи и програмируемите логически контролери;
 - ПСК са приложими, където от микро/малко интелигентно управляващо устройство се нуждаят уникални или дребносерийни обекти;
 - ЕМК са оптимални за изграждане на ПСК;
 - Опростената организация на изчислителния процес в ПСК повишава надеждността и съкращава времето на разработка;
 - Избраният език за разработка на системното програмно осигуряване – асемблер – предполага оптимално използване на ресурсите на ЕМК - максимално бързодействие и минимален обем памет;
 - Разработени и имплементирани са оригинални системен (за зареждане и настройка на потребителски програми), и мрежови (за връзка с други ПСК и/или горно ниво) комуникационни протоколи, описана е техниката на избор на протокол;
 - Предложена е методика от пет последователни етапа за изграждане на СПО за ПСК, необвързана с конкретна елементна база.

ГЛАВА ТРЕТА

РАЗРАБОТВАНЕ И НАСТРОЙКА НА ПОТРЕБИТЕЛСКО ПРОГРАМНО ОСИГУРЯВАНЕ ЗА ПРОГРАМИРУЕМИ СПЕЦИАЛИЗИРАНИ КОНТРОЛЕРИ

Универсален език за потребителско програмиране УЕПП

При процедурните езици програмата е поредица (списък, list) от инструкции, които се изпълняват последователно в определен ред, който зависи от решенията на конструкции от оператори за управление.

Типове данни

- Цифрови входове - **I** - от I1 до I24. Приемат стойност 0 или 1;
- Цифрови изходи - **O** - от O81 до O96. Приемат стойност 0 или 1;
- Логически променливи - **V** - от V1 до V128. Приемат стойност 0 или 1;
- Десетични променливи - **D** - от D1 до D104. Приемат целочислена положителна стойност от 0 до 9999;
- Времезадаващи променливи - **T** - от T1 до T24. Могат да реализират времезакъснения от 10мс до 99,99сек.;
- Константи - **C** – цели десетични числа от C0 до C9999;
- Етикети - **L** - от L0 до L511.

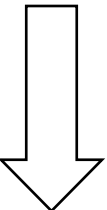
Специализирани променливи в УЕПП

Специализираните променливи в УЕПП служат за интерфейс между СПО и ПП.

Типове инструкции

- Актуализиране на изходи - код на операция **“OUT”**
- Четене на входове - код на операция **“INP”**
- Присвояване на стойност - код на операция **“LDU”**
- Логическо отрицание - код на операция **“NOT”**.
- Логическо "или"/"и" - код на операция **“OR”/“AND”**
- Събиране - код на операция **ADD**, форматите на инструкции са еднакви и за изваждане **SUB**, умножение **MUL** и деление **DIV**. Аритметичните операции се извършват с цели положителни числа
 - Увеличаване (намаляване) на десетична променлива с 1 - код на операция **INC (DEC)**
 - Безусловен преход - код на операция **BRA**
 - Условен преход по "препълване"=1(0) - код на оп. **BCS (BCC)**
 - Условен преход по нулева (ненулева) стойност на променлива - код на операция **BEQ (BNE)**
 - Поставяне на етикет - код на операция **LBL**

Транслатори на УЕПШ

LDU T15 C1000

**Pseudocode
cross-compiler**

0C OF 03 E8

program text

pseudocode (hex)


**Internal
interpreter**

bcf status,c *movwf indf2*
rlcf icb1,w *incf fsr2l*
addlw low (timvar-2) *movf icb3,w*
movwf fsr2l *movwf indf2*
movf icb2,w

**machine
instructions**

Система за зареждане и настройка на потребителски програми (СЗН)

DPCProgrammer - Opened COM1 port

File Tools Settings Help

Download Tools Start Pause Stop Debug Compile Help

```

B3 B5 B22
0 1 0

D4 D70 D101
4539 0 123

T4 T5 T12 T23
0 182 0 4245
    
```

Inputs state:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Outputs state:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Program: mt50xsm Mode: Running Status: 1 Scan Time: 1 msec. Prog. Size 8 % (461 bytes)

General view Boolean variables Decimal variables Timer variables Inputs state Outputs state

Detected controller DPC41H5, Soft.Rev. 1.H5

DPCProgrammer - Opened COM1 port

File Tools Settings Help

Download Tools Start Pause Stop Debug Compile Help

Select boolean variable(s) to be shown in the General View

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
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112
113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144
145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176
177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192
193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224
225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256

Program: mt50xsm Mode: Stopped Status: 0 Scan Time: 0 msec. Prog. Size 8 % (461 bytes)

General view Boolean variables Decimal variables Timer variables Inputs state Outputs state

Detected controller DPC41H5, Soft.Rev. 1.H5

DPCProgrammer - Opened COM1 port

File Tools Settings Help

Download Tools Start Pause Stop Debug Compile Help

Select decimal variable(s) to be shown in General View

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
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112
113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144
145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176
177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192
193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224
225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256

Program: mt50xsm Mode: Stopped Status: 0 Scan Time: 0 msec. Prog. Size 8 % (461 bytes)

General view Boolean variables Decimal variables Timer variables Inputs state Outputs state

Detected controller DPC41H5, Soft.Rev. 1.H5

DPCProgrammer - Opened COM1 port

File Tools Settings Help

Download Tools Start Pause Stop Debug Compile Help

Select timer variable(s) to be shown in General View

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
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112
113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144
145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176
177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192
193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224
225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256

Program: mt50xsm Mode: Stopped Status: 0 Scan Time: 0 msec. Prog. Size 8 % (461 bytes)

General view Boolean variables Decimal variables Timer variables Inputs state Outputs state

Detected controller DPC41H5, Soft.Rev. 1.H5

Изводи

- УЕПП може да бъде използван при разработката на ПСК, изградени върху различни ЕМК;
- Използването на специализирани променливи в УЕПП дава възможност на разработчика на конкретен ПСК да добавя нови такива, в зависимост от апаратната част и целевия клас на ПСК;
- Използването на език от високо ниво за програмиране на технологичната програма дава възможност на специалисти, работещи в различни области сами да съставят и настройват потребителските програми за ПСК;
- Първият модул от двумодулния транслятор на УЕПП дава възможност за генериране на процесорно независим код, което разширява неограничено елементната база използвана в ПСК;
- СЗН е процесорно независима;
- С предложения УЕПП могат да се кодират ПП по различни първични описания – блок-схеми на алгоритми, релейни схеми, логически уравнения, мрежи на Петри и т.н.

ГЛАВА ЧЕТВЪРТА. ПРИЛОЖИМОСТ НА СПЕЦИАЛИЗИРАНИТЕ ИНТЕЛИГЕНТНИ КОНТРОЛЕРИ

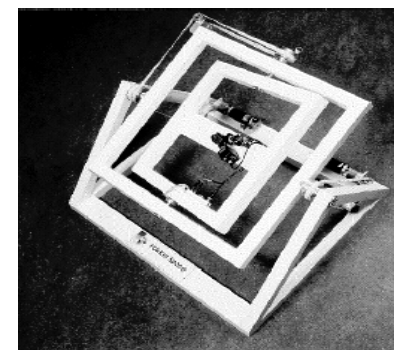
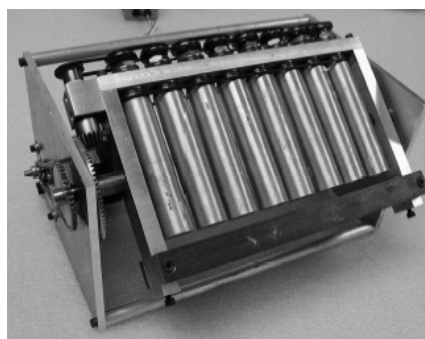
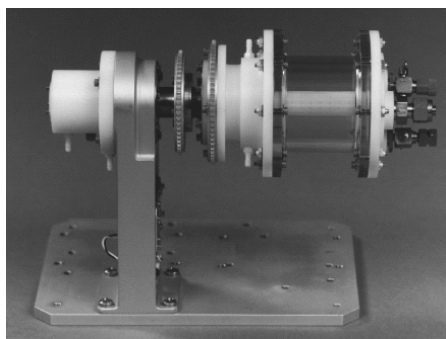
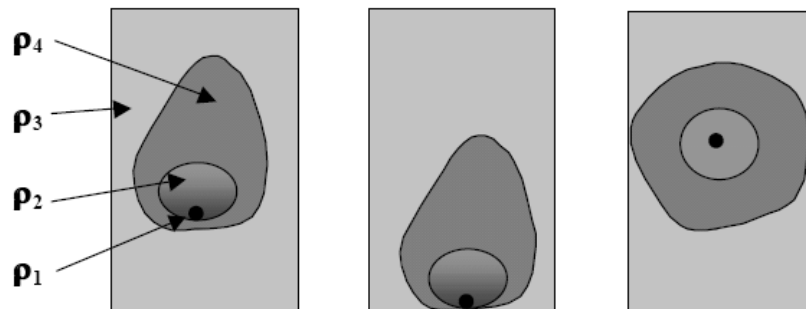
Експериментите са извършени върху различни ПСК както следва:

- ПСК за управление на ротационен биореактор – В рамките на договор с ФНИ-МОН: ТН-1514/05;
- ПСК за управление на енергийните потоци на безпилотен летателен апарат – в рамките на договор с ФНИ-МОМН: ТК-Х-1705/2007;
- ПСК за управление на стенд за изпитване на НЕРА филтри – В рамките на договор с ФНИ-МОМН: DINOV-09-0003/2009.

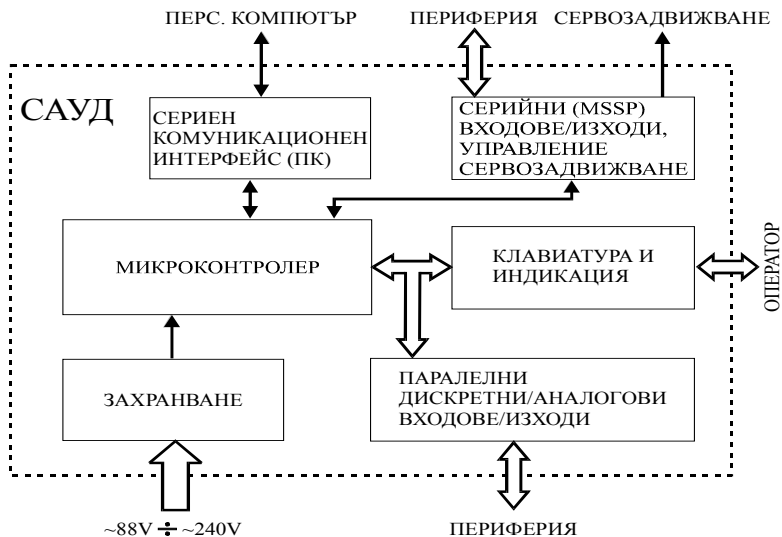
Данните от експерименталните изследвания са анализирани и обобщени.

ПСК за управление на ротационен биореактор

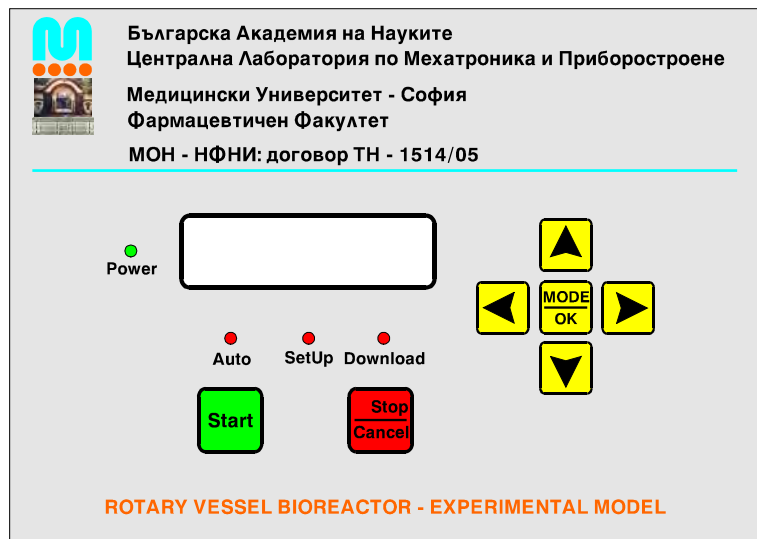
От използваните в тъканното инженерство различни видове биореактори голямо разпространение са получили биореакторите с въртящ се съд (БВС). Това се дължи на ниските механически напрежения върху отглежданите в БВС клетъчни и тъканни култури, тъй като повечето от тях са особено чувствителни към напреженията на срязване.



Идеен проект hardware



КБВС управлява БВС като позволява задаването от оператор и изпълнението на програма за израстване на клетъчни и тъканни култури. Контролерът може да работи автономно – без необходимост от горно ниво (персонален компютър).



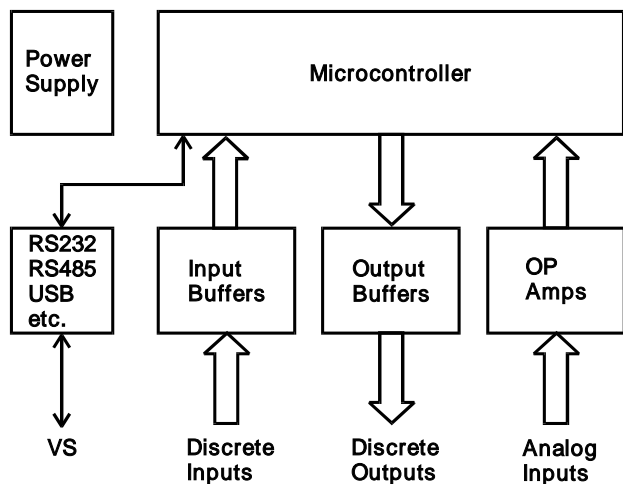
ПСК за управление на енергийните потоци на безпилотен летателен апарат

Безпилотният летателен апарат (БЛА) се задвижва от двутактов бензинов двигател, който се стартира от постояннотоков стартер/генератор, захранван от акумулатор. Целта е да се осигури достатъчна мощност за надеждното стартиране на двигателя по време на полет, като се добави и суперкондензаторна клетка. Предимствата на новата конфигурация са:

- Възможност за безшумен (ghost mode) режим на летене с електродвигател;
- Възможност за форсаж на двигателя, чрез използване на енергията, натрупана в суперкондензатора;
- Възможност за многократно стартиране във въздуха на бензиновия двигател при отрицателни температури благодарение на енергията натрупана в суперкондензатора.

За постигането на тези цели бе необходимо да се разработи контролер, управляващ енергийните потоци в БЛА (КУЕП) и осигуряващ оптимално зареждане и разреждане на акумулатора и суперкондензатора.

Идеен проект HW

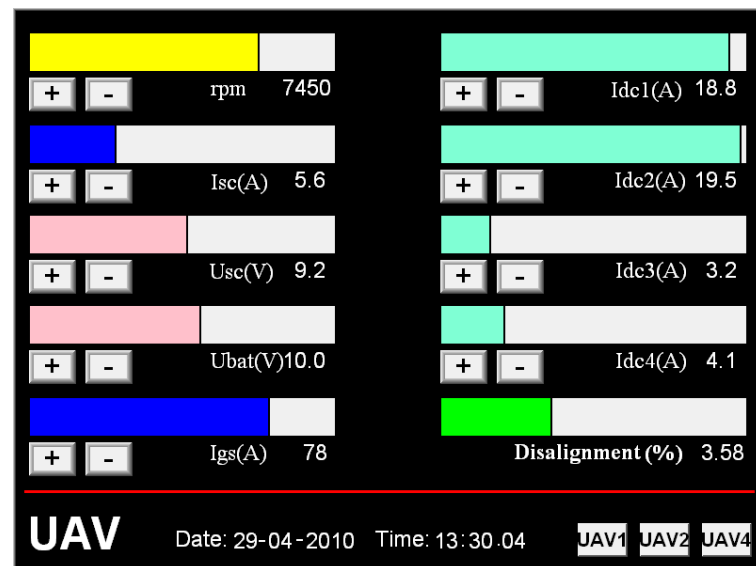
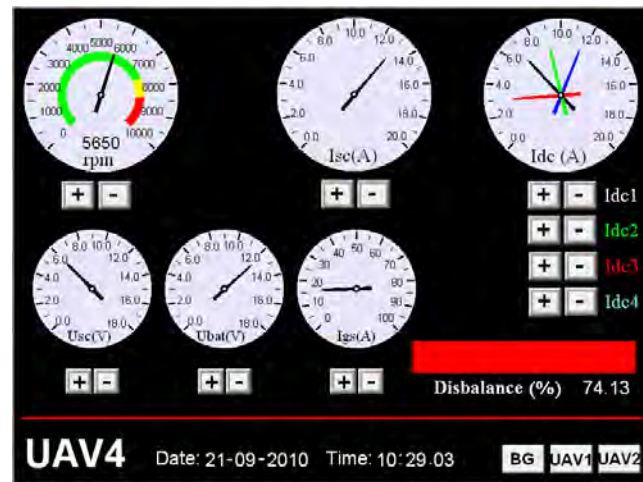
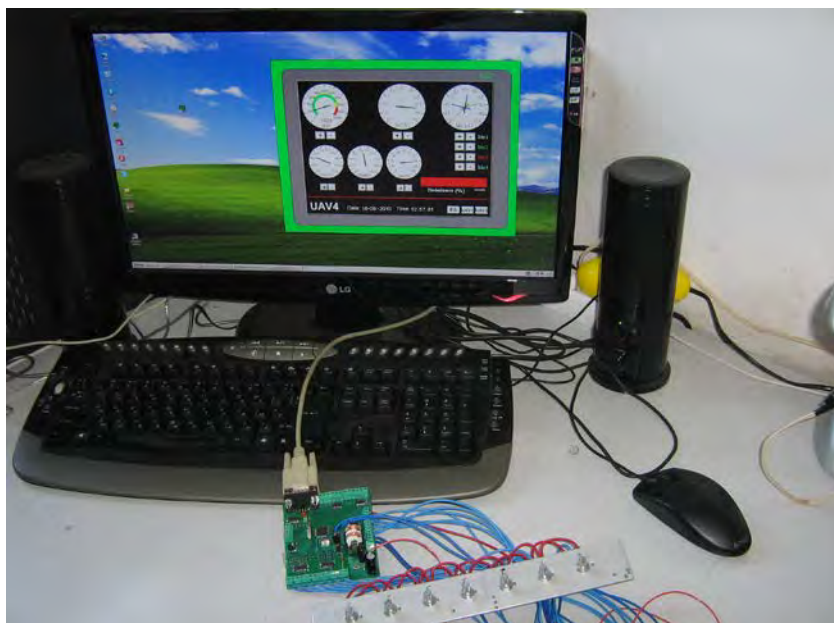


Name	Type	Level	Ref.
Supercapacitor charger ready	Input	TTL	RDYscc
Starter control unit ready	Input	TTL	RDYscu
Power switch ready	Input	TTL	RDYps
DC/DC converter A ready	Input	TTL	RDYdca
DC/DC converter B ready	Input	TTL	RDYdcb
Supercapacitor charger enable	Output	TTL	ENScc
Starter control unit enable	Output	TTL	ENscu
Power switch enable	Output	TTL	ENps
DC/DC converter A enable	Output	TTL	ENdca
DC/DC converter B enable	Output	TTL	ENdcb

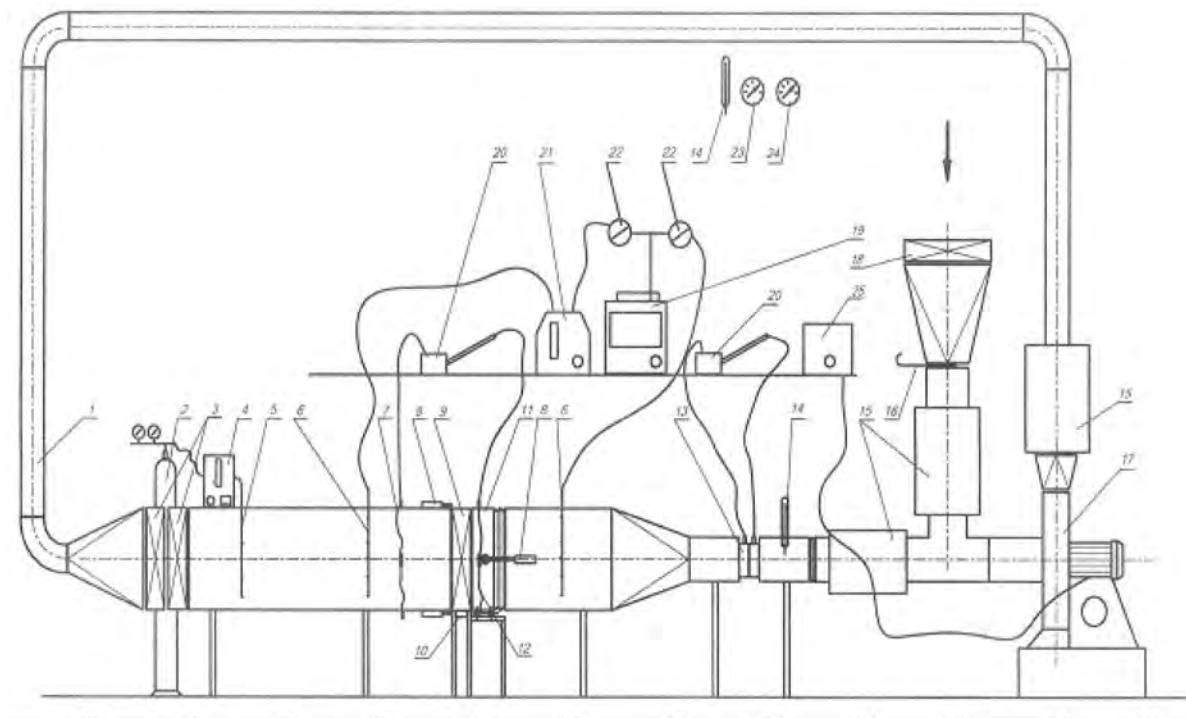
Name	Min. Value	Max. Value	Units	Reference
Engine revolutions per minute	0	7000	rpm	RPM
Supercapacitor voltage	0	16.8	V	Usc
Battery voltage	12	16.8	V	Ubat
Generator/starter current	0	100	A	Igs
Supercapacitor charge current	0	20	A	Isc
DC/DC converter A current	0	20	A	Idca
DC/DC converter B current	0	20	A	Idcb

Програмно осигуряване на КУЕП

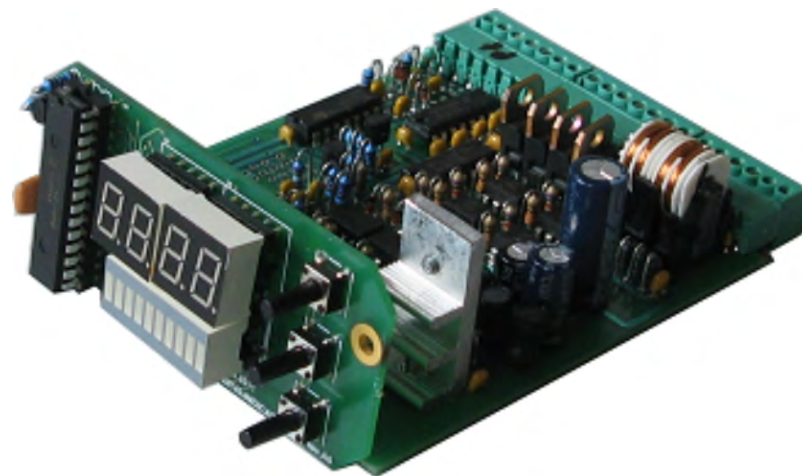
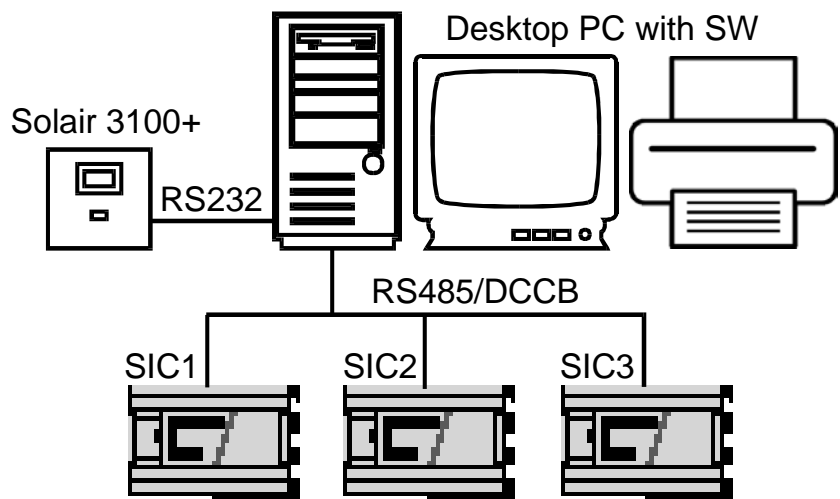
Системното програмно осигуряване на КУЕП е разработено като е използвана методиката, описана във втора глава. Вграден е системния протокол DPSP и мрежовия DSSB. Технологичната програма, е разработена на асемблер, а тестова програма за връзка с горно ниво – на УЕПП.



ПСК за управление на стенд за изпитване на НЕРА филтри



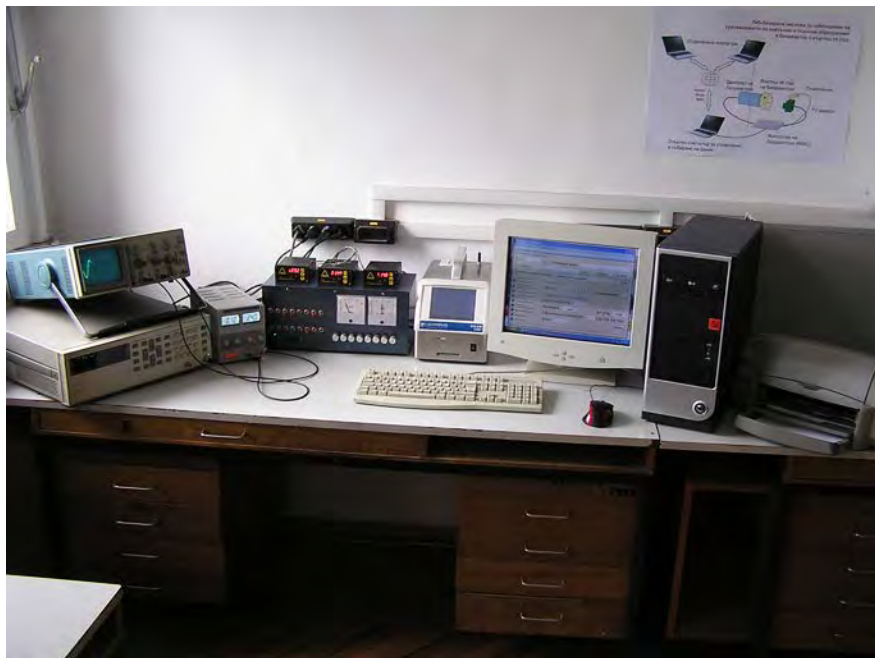
Апаратна част



- входове - 3 дискретни, оптоизолирани, 'npn', DC 24V/10ma;
- бърз брояч до 200kHz – 1 бр.;
- аналогови входове (напреженови, токови, Pt100) – 2 бр.;
- изходи - 4 дискретни, оптоизолирани, 'npn', DC 24V/0.5A;
- индикатори, управлявани от потребителя – четири седемсегментни;
- фолийна клавиатура с три бутона, достъпни за потребителя;
- комуникационен канал RS232/RS485 за програмиране, настройка и връзка в локална мрежа с други контролери и интелигентни терминали;

Програмно осигуряване на ПСК

Използвани са методиката и средствата описани в глави 2 и 3.



Специализиран софтуер на горното ниво – операторски панел

HEPA Filter Test Stand

File Edit View Tools Windows Help

Изпитван филтър

Код Размери [мм]

N серия Дата

Провел изпитанието

Състояние на стенда

- Сгъстен азот
- Монодисперсен аерозолен генератор тип MAGE
- Спирателен кран 1
- Лазерен брояч на частици
- Спирателен кран 2
- Колектор за статично налягане
- MM1 = 100.0 hPa
- MM2 = 2750.4 hPa
- T = 31.1°C
- Вентилатор

Тестове

Дебит Ефективност

T [°C] MM1 [hPa] MM2 [hPa]

Резултати от тестовете

Дебит Q [m³/h]

Пробиви ΔP [Pa]

Ефективност DEHS [%] EN 779; EN 1822

Клас

Status

Провеждане на тестове на HEPA филтри

Изводи

- Доказана е областта на приложение на ПСК;
- Апробирана е стандартизираната методика за изграждане на СПО на ПСК и използването на DPCP за настройката му;
- Показана е ефективността на разработените УЕПП и СЗН;
- Използван е апарата на специализираните променливи от УЕПП за реализиране на аритметични действия с плаваща запетая (IEEE 754) в ПСК;
- Представена е възможността за използване на интелигентен терминал или персонален компютър като горно ниво в йерархична система за управление, в чиито състав влизат ПСК с вграден мрежов протокол DCCB.

Обобщените резултати от експериментите потвърждават приложимостта на ПСК и методиката и средствата за изграждането на програмното им осигуряване

НАУЧНО-ПРИЛОЖНИ ПРИНОСИ

1. Изследвана и анализирана е организацията на изчислителния процес при вградените системи и програмируемите контролери;
2. Дефиниран е класа програмируеми специализирани контролери.
3. Създадена е необвързана с конкретна елементна база методика за изграждане на системното програмно осигуряване на ПСК;
4. Разработена е комуникационна подсистема на ПСК, съобразена със стандарта OSI – ISO;
5. Създаден е универсален език за потребителско програмиране на ПСК с двумодулен транслятор;
6. Разработена е оригинална среда за изграждане на потребителски програми за апаратнонезависими ПСК;
7. Различни ПСК са експериментирани в три обекта, като резултатите от експериментите показват приложимостта на предлаганите методика, език за програмиране, комуникации с горно ниво и система за зареждане и настройка, разработени в дисертационния труд.

Създадените методика и средства за програмиране на ПСК позволяват да се повиши ефективността на разработка на програмното осигуряване на управляващи устройства за уникални и дребносерийни обекти.

Основните идеи и постигнатите резултати в дисертационния труд са изнесени на 7 международни конференции и са публикувани в съответните сборници с доклади.



Благодаря за вниманието !