UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Pertama
Sidang Akademik 2010/2011

November 2010

EEM 352 – REKABENTUK MEKATRONIK II

Masa : 2 Jam

Sila pastikan bahawa kertas peperiksaan ini mengandungi TUJUH muka surat beserta Lampiran TIGA muka surat bercetak sebelum anda memulakan peperiksaan ini.

Kertas soalan ini mengandungi **EMPAT** soalan.

Jawab **TIGA** soalan.

Mulakan jawapan anda untuk setiap soalan pada muka surat yang baru.

Agihan markah bagi setiap soalan diberikan di sudut sebelah kanan soalan berkenaan.

Jawab semua soalan dalam Bahasa Malaysia atau Bahasa Inggeris atau kombinasi kedua-duanya.

[Sekiranya terdapat sebarang percanggahan pada soalan peperiksaan, versi Bahasa Inggeris hendaklah diguna pakai].

"In the event of any discrepancies, the English version shall be used".

1. Pilih salah satu daripada tajuk-tajuk di bawah dan seterusnya rekabentuk satu sistem berdasarkan mikropengawal. Rekabentuk anda mestilah mengandungi definasi masalah, pemilihan mikropengawal, pemilihan litar antaramuka, pemilihan bahasa pengaturcaraan, lukisan skematik dan carta-alir bagi program.

   Choose one of the following titles and then design a microcontroller-based system. Your design should consist of problem definition, microcontroller selection, interfacing circuit selection, programming language selection, schematic drawing and program flowchart.

   (a) Rumah Pintar
   Smart House

   (b) Kawalan Jauh Pintar untuk Kereta
   Smart Car Remote Control

   (c) Kereta Pintar
   Smart Car

   (d) Kawalan Jauh Pintar
   Smart Remote Control

   (100%)

2. (a) (i) Lukis gambarajah skematik sistem keselamatan bagi sebuah rumah yang menggunakan sistem berdasarkan kepada mikropengawal PIC16F84 (dilampirkan). Rujuk Jadual 2(a).

   Draw the schematic for a security system for a house using a system based on microcontroller PIC 16F84 (attached). Refer Table 2(a).

   (20%)

Jadual 2(a)
Table 2(a)

Peralatan	Kuantiti
Pintu (dipasang suis sebagai penderia)	2
Tingkap (dipasang suis sebagai penderia	2
Loceng (berbunyi apabila ada pencerobohan)	1
Pengesan pergerakan	1
Suis untuk dipilih oleh penghuni samada (2) Sedang tidur (3) Tiada di rumah (4) Ada di rumah	2

(ii) Bina satu carta alir bagi sistem keselamatan yang anda bina dalam (i).
Build a flow chart for the security system that you build in (i).

(10%)

(iii) Berdasarkan carta alir dalam (ii), tuliskan satu program lengkap dalam PicBasic.
Based on the flow chart you have drawn in (ii), write a complete program in PicBasic Pro.

(10%)

(b) Terangkan tentang penyediaan satu pin mikropengawal PIC sebagai masukan, keluaran dan ‘tarik-naik lemah’. Lukis gambarajah Blok bagi pin tersebut.

Explain the setting up of a pin of a PIC microcontroller as input, output and ‘weak pull-up’. Draw the block diagram of the pin.

(20%)

(c) Lukis litar antaramuka mikropengawal bagi 4 peranti masukan dan 6 peranti keluaran.

Draw the microcontroller interface circuits for 4 input devices and 6 output devices.

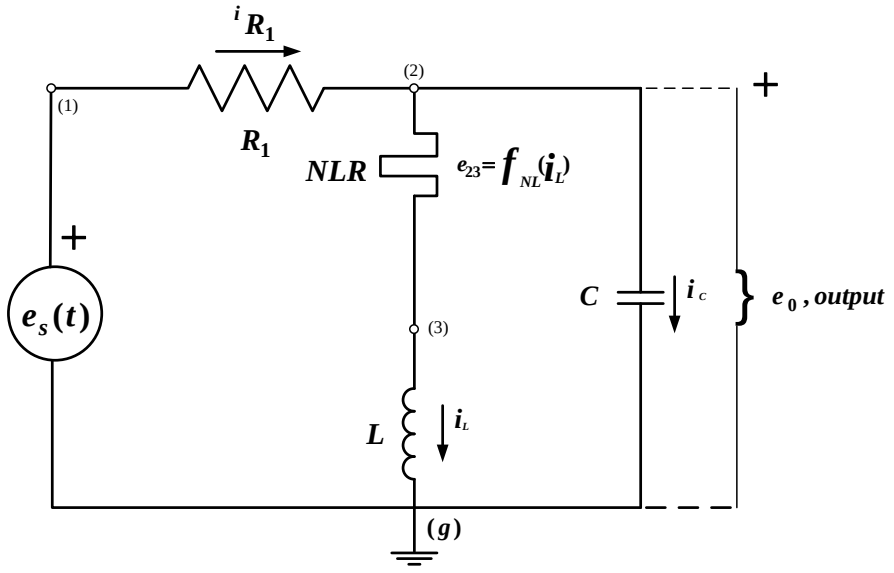
(20%)

(d) Terangkan sebab-sebab menggunakan mikropengawal bagi rekabentuk sistem mekatronik.

Explain the reasons for using microcontroller for mechatronic system design.

(20%)

3. Berdasarkan pada Rajah 3:
Based on Figure 3:



Rajah 3
Figure 3

…5/-

(a) Tuliskan persamaan-persamaan pembolehubah keadaan berdasarkan pada unsur-unsur tenaga simpanan *L* dan *C* untuk litar di atas.

Write the state-variable equations based on the energy storage elements L and C for the circuit shown above.

(30%)

(b) Leluruskan persamaan-persamaan ini untuk gangguan kecil dari semua pembolehubah.
Linearise these equations for small perturbation of all variables.

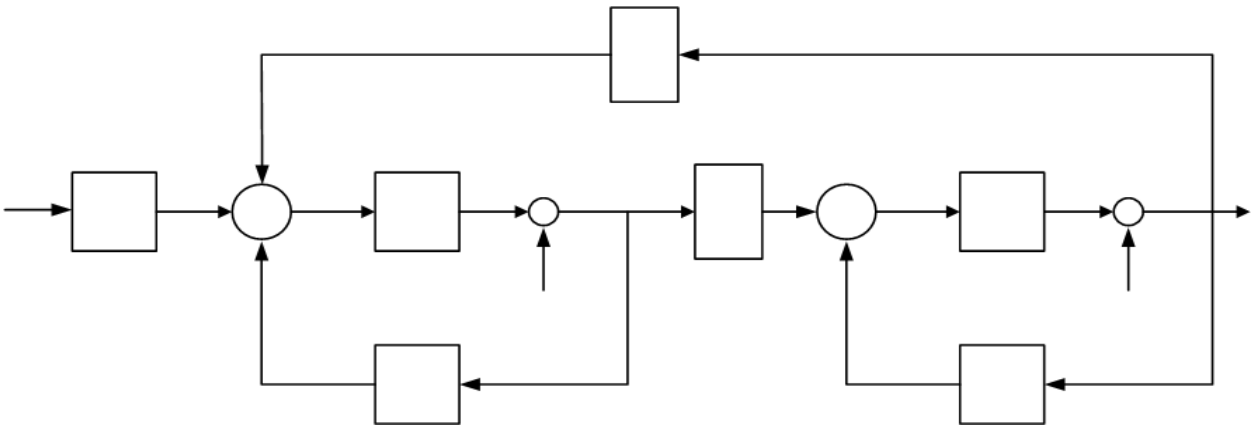
(40%)

(c) Gabungkan untuk menghilangkan pembolehubah yang tidak dikehendaki dan dapatkan persamaan kebezaan sistem masukan-keluaran yang mengaitkan antara keluaran kepada masukan .

Combine to eliminate the unwanted variable and obtain the input-output system differential equation relating the output to the input .

(20%)

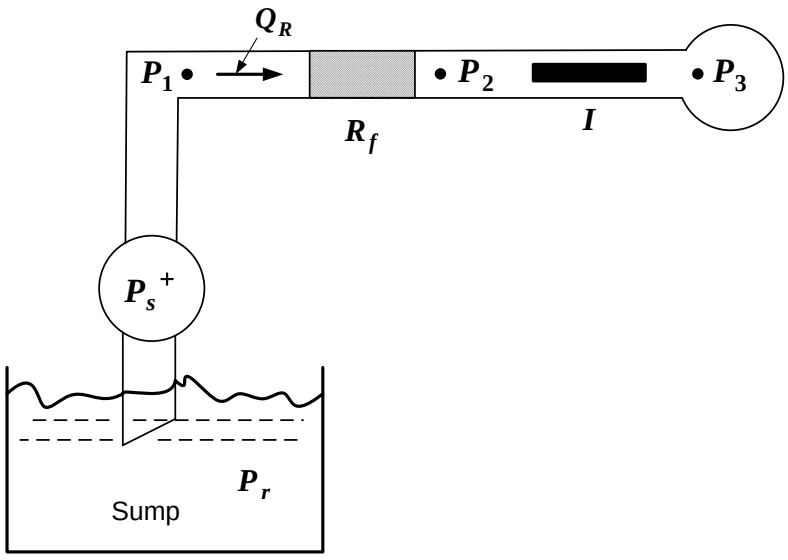
(d) Isikan gambarajah blok simulasi untuk sistem terlelurus.
Fill in the simulation block diagram for the linearised system.



(10%)

4. (a) Cari satu set persamaan pembolehubah keadaan dan kembangkan persamaan kebezaan masukan-keluaran yang mengaitkan tekanan keluaran kepada tekanan masukan untuk bendalir dalam Rajah 4.

Find a set of state-variable equations and develop the input-output differential equation relating the output pressure to the input pressure for the fluid shown in Figure 4.

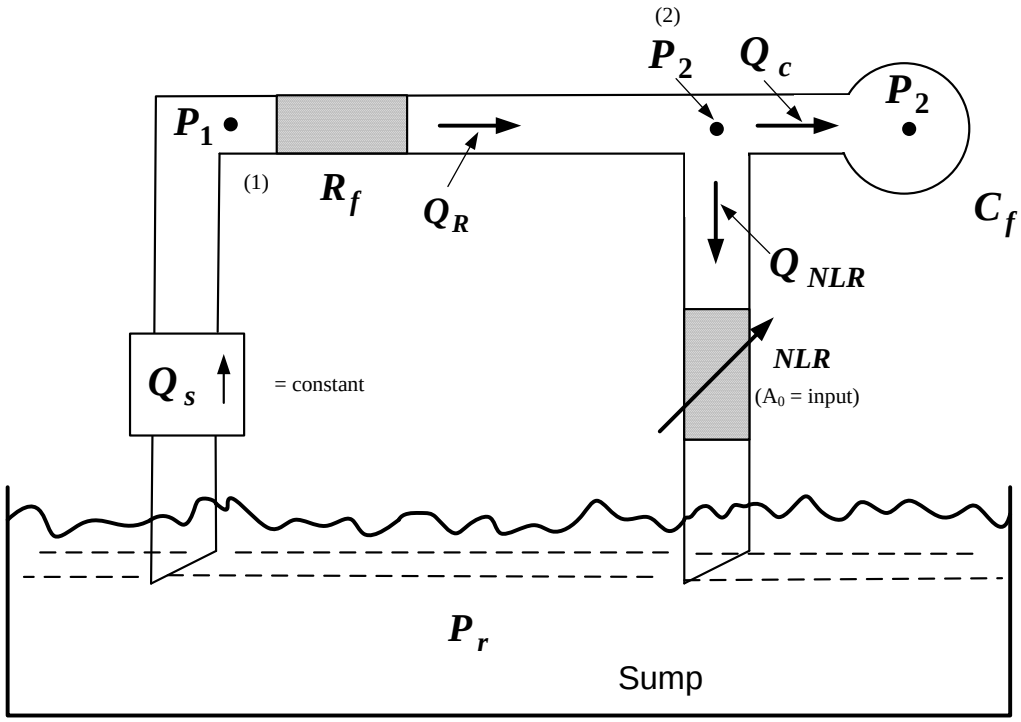


Rajah 4
Figure 4

(40%)

(b) Sebuah pembolehubah-orifis perintang tak lelurus sedang digunakan untuk memodulatkan kadar aliran dan mengawal tekanan dalam sistem kawalan bendalir mudah seperti di bawah. Persamaan aliran bagi orifis adalah . Kembangkan persamaan kebezaan masukan-keluaran yang mengaitkan perubahan kecil dalam tekanan keluaran kepada perubahan kecil dalam kawasan orifis apabila arus bekalan adalah tetap.

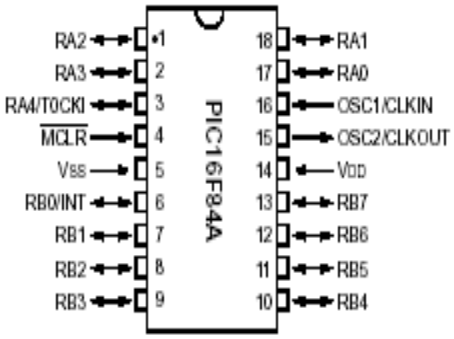
A variable-orifice nonlinear resistor is being used to modulate the flow rate and control the pressure in the simple fluid control system shown below. The flow equation for the orifice is . Develop the input-output differential equation relating small changes in the output pressure to small changes in the orifice area when the supply flow is constant.



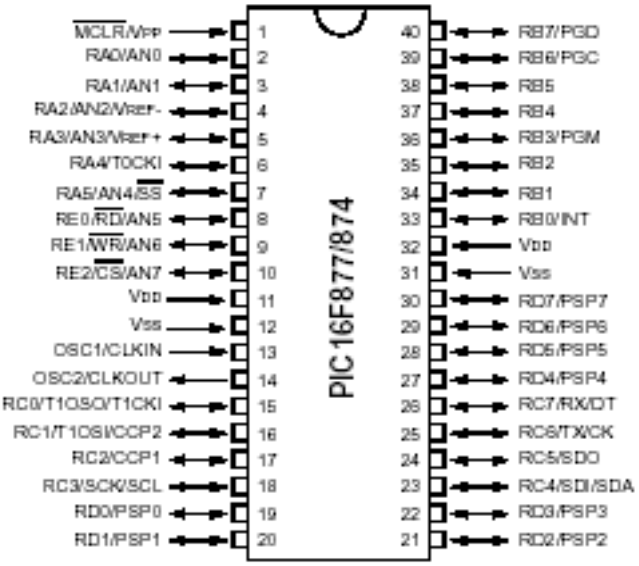
Rajah 5
Figure 5

(60%)

oooo0oooo



PIC16F84



PIC16F877

Table 7.5 PicBasic Pro statement summary

Statement	Description
@ assembly statement	Insert one line of assembly language code
ADCIN channel, var	Read the on-chip analog to digital converter (if there is one)
ASM . . . ENDASM	Insert an assembly language code section consisting of one or more statements
BRANCH index, [label1{, label2, . . .}]	Computed goto that jumps to a label based on index
BRANCHL index, [label1{, label2, . . .}]	Branch to a label that can be outside of the current page of code memory (for PICs with more than 2 k of program ROM)
BUTTON pin, down_state, auto_repeat_delay, auto_repeat_rate, countdown_variable, action_state, label	Read the state of a pin and perform debounce (by use of a delay) and autorepeat (if used within a loop)
CALL assembly_label	Call an assembly language subroutine
CLEAR	Zero all variables
CLEARWDT	Clear the watch-dog timer
COUNT pin, period, var	Count the number of pulses occurring on a pin during a period
DATA {@ location,} constant1{, constant2, . . .}	Define initial contents of the on-chip EEPROM (same as the EEPROM statement)
DEBUG item1{, item2, . . .}	Asynchronous serial output to a pin at a fixed baud rate
DEBUGIN {timeout, label,} [item1{,{item2, . . .}]	Asynchronous serial input from a pin at a fixed baud rate
DISABLE	Disable ON INTERRUPT and ON DEBUG processing
DISABLE DEBUG	Disable ON DEBUG processing
DISABLE INTERRUPT	Disable ON INTERRUPT processing
DTMFOUT pin, {on_ms, off_ms,} [tone1{, tone2, . . .}]	Produce touch tones on a pin
{EEPROM {@ location,} constant1{, constant2, . . .}}	Define initial contents of on-chip EEPROM (same as the DATA statement)
ENABLE	Enable ON INTERRUPT and ON DEBUG processing
ENABLE DEBUG	Enable ON DEBUG processing
ENABLE INTERRUPT	Enable ON INTERRUPT processing
END	Stop execution and enter low power mode
FOR count = start TO end {STEP {-} inc} {body statements}	Repeatedly execute statements as count goes from start to end in fixed increment
NEXT {count}	
FREQOUT pin, on_ms, freq1{, freq2}	Produce up to two frequencies on a pin
GOSUB label	Call a PicBasic subroutine at the specified label
GOTO label	Continue execution at the specified label
HIGH pin	Make pin output high
HSERIN {parity_label,} {time_out, label,} [item1{, item2, . . .}]	Hardware asynchronous serial input (if there is a hardware serial port)
HSEROUT [item1{, item2, . . .}]	Hardware asynchronous serial output (if there is a hardware serial port)
I2CREAD data_pin, clock_pin, control,{ address,} [var1{, var2, . . .}]{, label}	Read bytes from an external I^2C serial EEPROM device
I2CWRITE data_pin, clock_pin, control,{ address,} [var1{, var2, . . .}]{, label}	Write bytes to an external I^2C serial EEPROM device
IF log_comp THEN label	Conditionally jump to a label

IF log_comp THEN
true_statements
ELSE
false_statements
ENDIF
	Conditional execution of statements

INPUT pin	Make pin an input
LCDIN {address,} [var1{, var2, . . .}]	Read RAM on a liquid crystal display (LCD)
LCDOUT item1{, item2, . . .}	Display characters on LCD
{LET} var = value	Assignment statement (assigns a value to a variable)

Statement	Description
LOOKDOWN value, [const1{, const2, . . .}], var	Search constant table for a value
LOOKDOWN2 value, {test} [value1{, value2, . . .}], var	Search constant/variable table for a value
LOOKUP index, [const1{, const2, . . .}], var	Fetch constant value from a table
LOOKUP2 index, [value1{, value2, . . .}], var	Fetch constant/variable value from a table
LOW pin	Make pin output low
NAP period	Power down processor for a selected period of time
ON DEBUG GOTO label	Execute PicBasic debug subroutine at label after every statement if debug is enabled
ON INTERRUPT GOTO label	Execute PicBasic subroutine at label when an interrupt is detected
OUTPUT pin	Make pin an output
PAUSE period	Delay a given number of milliseconds
PAUSEUS period	Delay a given number of microseconds
{PEEK address, var}	Read byte from a register
{POKE address, var}	Write byte to a register
POT pin, scale, var	Read resistance of a potentiometer, or other variable resistance device, connected to a pin with a series capacitor to ground
PULSIN pin, state, var	Measure the width of a pulse on a pin
PULSOUT pin, period	Generate a pulse on a pin
PWM pin, duty, cycles	Output a pulse width modulated (PWM) pulse train to pin
RANDOM var	Generate a pseudo-random number
RCTIME pin, state, var	Measure pulse width on a pin
READ address, var	Read a byte from on-chip EEPROM
READCODE address, var	Read a word from code memory
RESUME {label}	Continue execution after interrupt handling
RETURN	Continue execution at the statement following last executed GOSUB
REVERSE pin	Make output pin an input or an input pin an output
SERIN pin, mode,{ timeout, label,} {[qual1, qual2, . . .],}{ item1{, item2, . . .}}	Asynchronous serial input (Basic Stamp 1 style)
SERIN2 data_pin{\flow_pin}, mode, {parity_label,} {timeout, label,} [item1{, item2, . . .}]	Asynchronous serial input (Basic Stamp 2 style)
SEROUT pin, mode, [item1{, item2, . . .}]	Asynchronous serial output (Basic Stamp 1 style)
SEROUT2 data_pin{\flow_pin}, mode, {pace,} {timeout, label,} [item1{, item2, . . .}]	Asynchronous serial output (Basic Stamp 2 style)
SHIFTIN data_pin, clock_pin, mode, [var1{\bits1} {, var2{\bits2}, . . .}]	Synchronous serial input
SHIFTOUT data_pin, clock_pin, mode, [var1{\bits1} {, var2{\bits2}, . . .}]	Synchronous serial output
SLEEP period	Power down the processor for a given number of seconds
SOUND pin, [note1, duration1{, note2, duration2, . . .}]	Generate a tone or white noise on a specified pin
STOP	Stop program execution
SWAP var1, var2	Exchange the values of two variables
TOGGLE pin	Change the state of an output pin
WHILE logical_comp statements	Execute code while condition is true
WEND	
WRITE address, value	Write a byte to on-chip EEPROM
WRITECODE address, value	Write a word to code memory
XIN data_pin, zero_pin, {timeout, label,} [var1{, var2, . . .}]]	Receive data from an external X-10 type device
XOUT data_pin, zero_pin, [house_code1\key_code1{\repeat1}{, house_code2\key_code2{\repeat2, . . .}]	Send data to an external X-10 type device

PicBasic Pro commands