UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Pertama
Sidang Akademik 2008/2009

November 2008

EEM 352 – REKABENTUK MEKATRONIK II

Masa: 2 jam

Sila pastikan bahawa kertas peperiksaan ini mengandungi SEMBILAN muka surat dan TIGA muka surat LAMPIRAN yang bercetak sebelum anda memulakan peperiksaan ini.

Kertas soalan ini mengandungi EMPAT soalan dan dua bahagian iaitu Bahagian A dan Bahagian B.

Jawab TIGA soalan.

Gunakan buku jawapan yang berasingan supaya jawapan-jawapan bagi soalan-soalan Bahagian A adalah dalam satu buku jawapan dan Bahagian B dalam buku jawapan yang lain.

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.

Bahagian A
Section A

1. Rekabentuk satu pegenggam yang dikawal oleh satu servo. Lakukan analisa statik dan analisa kinematik serta nyatakan masalah yang mungkin berlaku atau kekangan dengan pembinaan pegengggam berdasarkan rekabentuk anda.

Design a simple gripper which is controlled by one servo. Perform static and kinematics analysis and state the possible problems or constraints with the constructed gripper based on your design.

Spesifikasi servo adalah seperti berikut:

The servo specifications are as follows.

Input signal = 5 V
Speed = 0° to 180° in 1.5 seconds
Weight = 45.0 gram
Torque = 3.40 kg cm

Dengan menggunakan PIC16F84 serta Pic Basic Pro, lukis skematik dan tulis program untuk mengawal pegenggam tersebut. Beri penjelasan bagi jawapan anda.

By using PIC16F84 and PicBasic Pro, draw the schematic and write the program to control the gripper. Provide explanation for your answers.

Diberikan di dalam **Lampiran 1** adalah susunatur pin bagi PIC16F84 serta penyataan PicBasic Pro.

*Given in **Appendix 1** the pin layout of PIC16F84 and PicBasic Pro statements.*

(100%)

2. Anda dikehendaki merekabentuk sistem keselamatan rumah menggunakan PIC16F84 dan PicBasic Pro. Gunakan proses rekabentuk berkaedah untuk menjawab soalan ini.

You are required to design a home security system using PIC16F84 and PicBasic Pro. Use a methodical design process in answering this question.

Antara keperluan:
Some requirements:

Di dalam rekabentuk anda, anda dikehendaki menggunakan pengesan pergerakan, buzzer, **wired-AND** pintu dan tingkap dan sekurang-kurangnya tiga keadaan operasi yang boleh dipilih oleh penghuni. Gunakan kreativiti anda untuk memperbaiki sistem keselamatan anda.

In your design, you are required to use motion detector, buzzer, ***wired-AND*** *doors and windows and at least three operating state of occupants setting/selection. Use your creativity to improve your security system.*

(100%)

Bahagian B
Section B

3. Soalan ini akan merangkumi bab berkenaan Isu-Isu Penyatupaduan sistem mekatronik.

This question will cover topics on Mechatronic System Integration Issues.

(a) Bagi pembangunan robot mobil, bincangkan tentang

For mobile robot development, discuss about

(i) Amalan terbaik penggantian bateri.
The best battery placement practises.

(ii) *Amalan terbaik pendawaian.*
The best wiring practises.

(45%)

(b) Anda akan menggerakkan halaju dan kedudukan motor dengan kaedah pemberian denyut PWM daripada mikropengawal ke peranti pensuisan.

You are to actuate the motor speed and position by means of giving PWM (pulse-width-modulation) pulse from the microcontroller to the switching device such as MOSFET.

(i) Bincangkan bagaimanakah anda melindungi MOSFET daripada bebanan pensuisan voltan dan arus yang tinggi, yang juga sekaligus dapat mengurangkan kehilangan kuasa. Gunakan lakaran bagi membantu penerangan anda.

Discuss how would you protect your MOSFET from switching stresses of high voltage and current which will in turn minimize the power losses. Use sketches to explain.

(15%)

...5/-

(ii) Bagaimanakah voltan diklipkan daripada sumber dan gat MOSFET supaya ciri-ciri keselamatan dapat disertakan dalam litar pemacu anda? Gunakan lakaran bagi membantu penerangan anda.

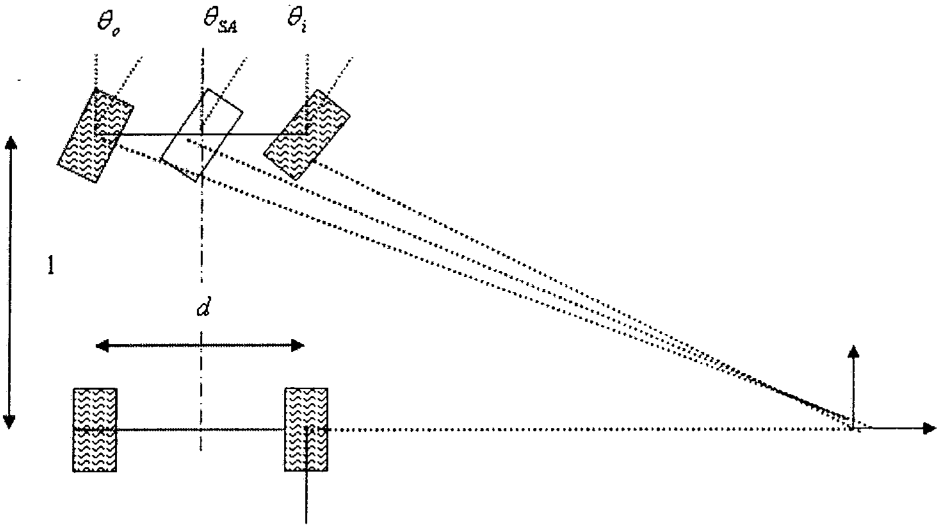
How to clip the voltage from the source and gate of the MOSFET so as to provide another safety feature of your driver circuit? Use sketches to explain.

(15%)

(c) Sebuah sistem pemacu perlu dipilih untuk rekabentuk robot bergerak seperti yang ditunjukkan dalam Rajah 1.

A drive system needs to be chosen for the mobile robot design shown in Figure 1.

Note that $\theta_i > \theta_0$



Rajah 1
Figure 1

...6/-

(i) Bincangkan tentang perbezaan di antara sistem pemacu holonomik dan bukan holonomik.

Discuss the difference between holonomic and nonholonomic drive systems.

(ii) Bincangkan sistem pemacu yang ditunjukkan dalam Rajah 1.

Discuss the drive system shown in Figure 1.

(iii) Tunjukkan pertalian antara pengasingan roda melintang dan sisi dengan sudut yang dibelokkan terhasil dari roda dalam dan roda luar.

Show the relationship between the longitudinal and lateral wheel separation with the angle turned produced by the inner and outer wheels.

(25%)

4. Soalan ini merangkumi bab berkenaan isu-isu pemodelan dan simulasi sistem pemacu robot bergerak.

This question covers topics on the modeling and simulation issues of mobile robot's drive system.

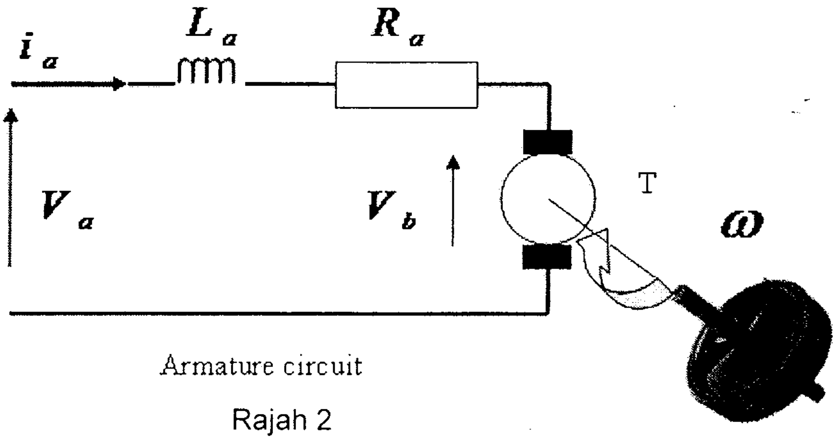
(a) Bincangkan apakah yang dimaksudkan oleh ketaklelurusan. Gunakan gambarajah bagi membantu penerangan anda.

Discuss what is meant by nonlinearity. Use diagrams to help your explanation.

(20%)

(b) Sebuah model dinamik motor DC seperti yang ditunjukkan dalam Rajah 2 diamati.

A dynamic model of DC motor as shown in Figure 2 is studied.



Rajah 2
Figure 2

Modelkan sistem yang ditunjukkan dalam Rajah 2 dalam perwakilan ruang keadaan mengaitkan voltan masukan ke motor servo DC dengan halaju sudut keluaran. Andaikan motor tersebut dikawal-angker, dengan rintangan dan kearuhan berada dalam keadaan bersiri dengan beban.

Model the system shown in Figure 2 in statespace representation relating the input voltage to a DC servo motor and the output angular velocity, assuming that the motor is armature controlled, having both resistance and inductance, in series with the load.

(30%)

(c) Anda akan memerhati kekangan kinematik roda sebagai langkah pertama kepada model kinematik robot.

You are to observe the wheel kinematic constraints as a first step to a kinematic model of the robot.

Tuliskan persamaan kekangan bagi setiap aturan roda robot mobil bergerak yang ditunjukkan.

Write the constraint equation for each of the wheel arrangement of mobile robot shown here. (50%)

Lampiran 2
Appendix 2

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}
NEXT {count}
	Repeatedly execute statements as count goes from start to end in fixed increment

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^{2}C serial EEPROM device
I2CWRITE data_pin, clock_pin, control,{ address,} [var1{, var2, . . .}]{, label}	Write bytes to an external I^{2}C 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)

Lampiran 3
Appendix 3

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 WEND	Execute code while condition is true
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