Instruction: Mechano-Informatics (Subject)

Date : H23 (2011), August 23, 9:00 – 11:30

Answers should be written either in Japanese or English.

1) Do not open this problem booklet until the start of the examination is announced.
2) Among four problems provided, choose and answer two.
3) If you find missing, misplaced, and/or unclearly printed pages in the problem booklet, notify the examiner.
4) Two answer sheets are provided. Check the number of them, and if you find excess or deficiency, notify the examiner. You must use a separate sheet for each problem. When you run short of space to write your answer on the front side of the answer sheet, you may use the back side by clearly stating so in the front side.
5) In the designated blanks at the top of each answer sheet, write examination name "Mechano-Informatics (Subject)", "Master" or "Doctor" , your applicant number, and the problem number. Omission in filling up these blanks may void your test score.
6) An answer sheet is regarded as invalid if you write marks and/or symbols unrelated to the answer.
7) Submit both answer sheets even if they are blank.
8) Use the blank pages in the problem booklet for your draft.
9) Fill in the blank below with your applicant number, and submit this booklet. Also submit the Japanese booklet with your applicant number in the corresponding blank.

Applicant number:

MEMO

(Do not detach this page)

MEMO

(Do not detach this page)

Problem 1

Figure 1 shows a hydraulic system with two pistons and two bar links on the horizontal plane. When P_{in}, the tip position of the link A moves, P_{out}, the tip position of the link B moves by this mechanism. Nodes b and d are pin-joints. Node d is connected to the fixed end and nodes c and e are the slider joints. Each length of ab, bc, cd and de is l, and pistons are moved only in y axis. From an oil-pump, high pressurized oil is provided sufficiently and the pushed oil is discharged to oil-tanks. Here as shown in Fig. 2, consider P_{out} moves by y_B when P_{in} moves by y_A. Define the amount of the oil per unit time from piston A to B as q. There are two proportional relationships as follows: q is proportional to the valve's opening displacement of piston A, z. q is also proportional to the moving speed of piston B. Their proportional constants are k_1 and k_2, respectively. The frictions and masses are negligible, and the displacements of y_A, y_B and z are small. Answer the following questions.

(1) Represent q by y_B. Also represent q by z.

(2) Represent z with y_A and y_B by using the relationship of the displacement between the link A and the link B.

(3) Derive the transfer function G of the input y_A to the output y_B.

(4) Draw the Bode diagram of G when k_1=63 and k_2=0.5.

(5) Describe the characteristics of this mechanism and give an example of the applications of the mechanism.

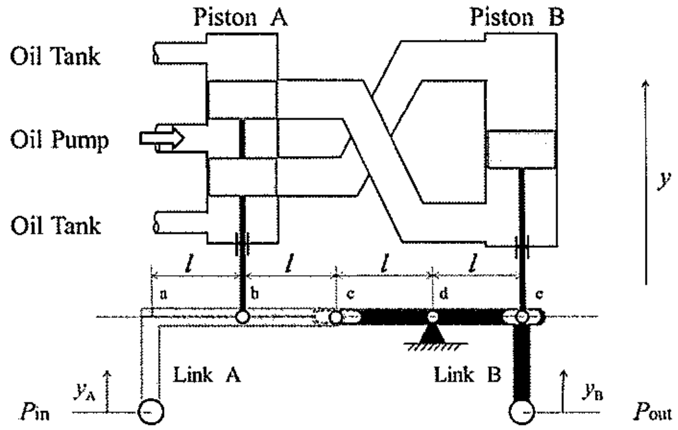

Fig. 1

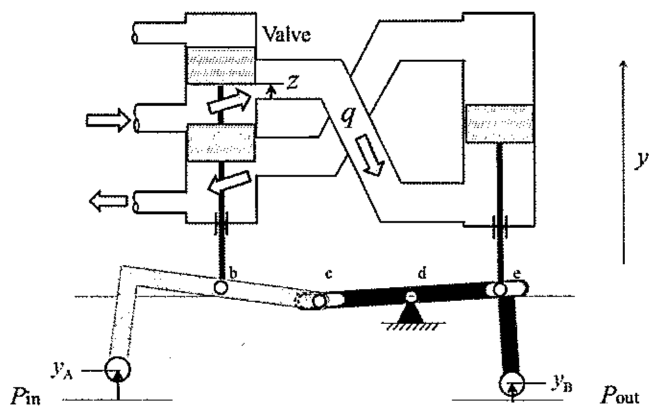

Fig. 2

MEMO

(Do not detach this page)

Problem 2

P. 1. Consider a circuit with an operational amplifier as shown in Fig.1. A voltage of V_r is applied to the series resistor which consists of n resistors of R and one resistor of $2R$. Each of n nodes between the resistors is connected to a switch through a resistor of $2R$. Let the voltage of each node be $V_i(0 \le i \le n-1)$, and the state of the corresponding switch be $S_i(0 \le i \le n-1)$. The switch is connected to a lower contact in case of $S_i = 0$, whereas to a upper contact in case of $S_i = 1$. Figure 1 shows the case all the switches are connected to lower contacts. Consider characteristics of the amplifier are ideal.

(1) V_i is constant in either case of $S_i = 0$ or $S_i = 1$. Explain the reason(s).

(2) Show the relationship between V_0 and V_1.

(3) Express V_{out} using V_r and S_i.

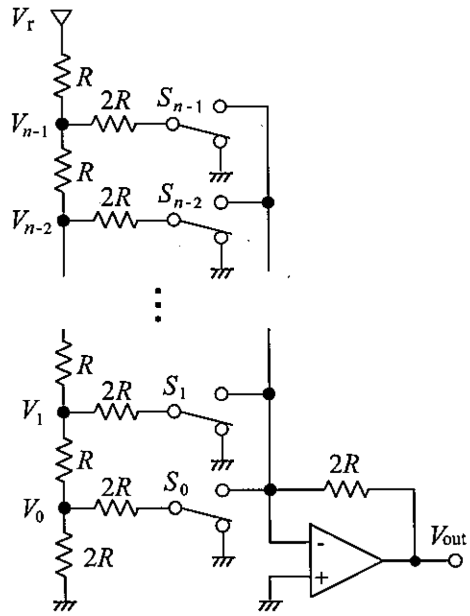

Fig. 1

P. 2. Consider a system that controls angular velocity of a DC motor. Note that the input voltage to the motor is $v(t)$, the angular velocity is $\omega(t)$, the resistance of the motor coil is R, and the total inertial moment of the motor including the load is J. The torque generated by the motor is proportional to the coil current, and let this proportional constant be K_a. The counter electromotive force proportional to the angular velocity is generated during the motor rotation, and let this proportional constant be K_e. Assume that the counter electromotive force induced by change of the motor coil current and the viscous resistance regarding the rotation are negligibly small.

(1) The model of the DC motor can be expressed by $G(s) = \dfrac{\Omega(s)}{V(s)} = \dfrac{K_m}{T_m s + 1}$, where $V(s)$ and $\Omega(s)$ are Laplace transformations of $v(t)$ and $\omega(t)$, respectively. Derive K_m and T_m.

(2) The feedback system shown in Fig. 2 is constructed to make the angular velocity follow the target. Note that K_p is a proportional gain. Show change of the time constant compared to the one without the feedback system. Also show steady-state error to a step input. K_m and T_m can be used if necessary.

(3) The feedback system shown in Fig. 3 is constructed to make the angular velocity follow the target. Note that K_i is an integral gain and K_v is a proportional gain. Show the condition for the fastest response without overshoot to a step input. K_m and T_m can be used if necessary.

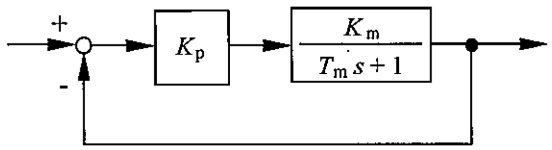

Fig. 2

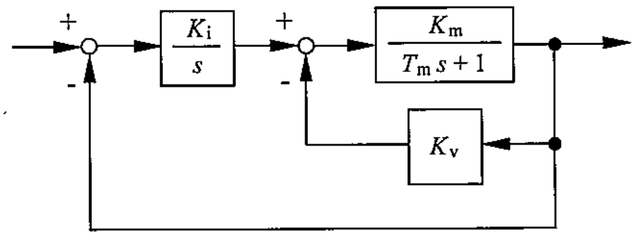

Fig. 3

Problem 3

P. 1. Describe the following terms in about three lines. You may use figures and tables if needed.

(1) Half-Adder and Full-Adder

(2) Exclusive OR

P. 2. Let $X = \{x_i\}_{i=1}^{m}$ and $Y = \{y_i\}_{i=1}^{n}$ be discrete random variables, having probability distributions $P(x_i) = P(X = x_i)$ and $P(y_i) = P(Y = y_i)$, respectively. The entropy, the conditional entropy, the joint entropy, and the mutual information are defined as follows:

The entropy of X is $$H(X) = -\sum_{i=1}^{m} P(x_i)\log_2 P(x_i).$$

The conditional entropy of X given Y is $$H(X \mid Y) = -\sum_{i=1}^{m}\sum_{j=1}^{n} P(x_i, y_j)\log_2 P(x_i \mid y_j).$$

The joint entropy of X and Y is $$H(X, Y) = -\sum_{i=1}^{m}\sum_{j=1}^{n} P(x_i, y_j)\log_2 P(x_i, y_j).$$

The mutual information between X and Y is $$I(X;Y) = \sum_{i=1}^{m}\sum_{j=1}^{n} P(x_i, y_j)\log_2 \frac{P(x_i, y_j)}{P(x_i)P(y_j)}.$$

Answer the following questions.

(1) Consider binary discrete random variables X and Y taking 0 or 1 for values, with the joint distribution shown in Table 1. Assuming $\log_2 3 = 1.6$, find $H(X)$, $H(Y)$ and $I(X;Y)$, respectively.

Table 1

		Y = 0	Y = 1
X	0	1/2	1/4
	1	0	1/4

(2) $P(X)$ that maximizes $H(X)$ can be obtained as a stationary point of the following Lagrange function L:

$$L = -\sum_{i=1}^{m} P(x_i)\log_2 P(x_i) + \lambda\left(1 - \sum_{i=1}^{m} P(x_i)\right),$$

where λ is a Lagrange multiplier. Find the probability distribution $P(X)$ that maximizes $H(X)$, and describe the process to derive the distribution.

(3) Show that $H(X, Y) = H(X \mid Y) + H(Y)$. Note that the following equations are satisfied:

$$P(x_i) = \sum_{j=1}^{n} P(x_i, y_j) \quad \text{and} \quad P(x_i, y_j) = P(x_i \mid y_j)P(y_j).$$

(4) Show that $I(X;Y) = H(X) + H(Y) - H(X, Y)$.

(5) Consider discrete random variables $X_1, X_2, \cdots, X_T$, which form a Markov chain in this order $X_1 \to X_2 \to \cdots \to X_T$. More specifically, the joint distribution of those variables $P(X_1, X_2, \cdots, X_T)$ can be written as

$$P(X_1, X_2, \cdots, X_T) = P(X_1)P(X_2 \mid X_1)\cdots P(X_T \mid X_{T-1}).$$

Reduce $I(X_1; X_2, X_3, \cdots, X_T)$ to its simplest form, and describe the process to derive the form.

MEMO

(Do not detach this page)

Problem 4

P. 1. Describe the following terms in about three lines. You may use figures.

(1) Compiler and Interpreter

(2) Superpipeline and Superscalar

(3) Hard real-time and Soft real-time

(4) Imperative programming and Declarative programming

P. 2. About list manipulation functions, answer the following questions.

(1) Explain the FIFO and the FILO of list manipulations in about two lines.

(2) File *queue.h* shown in Program 1 is an implementation of the FIFO. Describe the output of the Program 2.

(3) Implement the FILO by changing an `add_queue` function of file *queue.h*. Describe the changed `add_queue` function, which is referred to as *queue2.h* .

P. 3. Program 3 is a graph search algorithm using the FIFO. Answer the following questions.

(1) A graph that has N nodes can be represented as an N-*by*-N matrix where an entry in the i-th row and the j-th column is 1 when there exists an arc from the node i to the node j, otherwise 0. Draw a graph structure that corresponds to the variable `adj` in the function `main` of Program 3.

(2) Describe the output of Program 3 and explain the behavior. You may use figures.

(3) Search algorithm using the FILO is achieved by including the file *queue2.h* instead of the file *queue.h* in Program 3. Describe the output and explain the behavior of this program. You may use figures.

(4) Search algorithm using the FILO is able to be implemented recursively. Fill in the blank in Program 4.

(5) Describe the time complexity and the memory requirement of programs (2), (3) and (4), respectively. Discuss the advantages and disadvantages of these programs. Suppose that every node has b successors, m is the maximum depth of the graph and d is the depth of the goal node.

```c
#include <stdio.h>
#include <stdlib.h>

typedef struct queue{
  int item;
  struct queue *next;
} queue_t;

queue_t *head;

void add_queue(int element) {
  queue_t *t = head;
  while (t->next != NULL) t = t->next;
  queue_t *q =
 (queue_t *)malloc(sizeof(queue_t));
  q->item = element;
  q->next = NULL;
  t->next = q;
}

int remove_queue() {
  int element;
  queue_t *q = head->next;
  head->next = q->next;
  element = q->item;
  free(q);
  return element;
}

void print_queue() {
  queue_t *q = head->next;
  while (q != NULL) {
    printf("%d ", q->item);
    q = q->next;
  }
  printf("¥n");
}
```

Program 1 (*queue.h*)

```c
#include "queue.h"

int main() {
  head = (queue_t *)malloc(sizeof(queue_t));
  head->item = -1;  head->next = NULL;
  add_queue(1);  add_queue(2);
  remove_queue();add_queue(3);
  remove_queue();add_queue(4);
  print_queue();
  return 0;
}
```

Program 2

```c
#include "queue.h"
#define N 7

int tree_search(int adj[N][N], int s, int g)
{
  int j, count = 0;
  add_queue(s);
  while ( head->next != NULL ) {
    printf("%d: ", count++);
    print_queue();
    int element = remove_queue();
    if (element == g) { return 1; }
    for ( j = 1; j < N; j++ )
      if ( adj[element][j] == 1 )
         add_queue(j);
  }
  return 0;
}

int main() {
  static int adj[N][N] =
    {{0, 1, 1, 0, 0, 0, 0},
     {0, 0, 0, 1, 0, 0, 0},
     {0, 0, 0, 0, 1, 1, 0},
     {0, 0, 0, 0, 0, 0, 0},
     {0, 0, 0, 0, 0, 0, 1},
     {0, 0, 0, 0, 0, 0, 0},
     {0, 0, 0, 0, 0, 0, 0}};
  head = (queue_t *)malloc(sizeof(queue_t));
  head->item = -1;
  head->next = NULL;
  tree_search(adj, 0, 6);
  return 0;
}
```

Program 3

```c
int tree_search(int adj[N][N], int s, int g) {
  int j;  printf("%d ", s);

  [                                        ]

   return 0;
}
```

Program 4

MEMO

(Do not detach this page)

MEMO

(Do not detach this page)