

**Pengaturcaraan Ingatan Setempat
Di Dalam Sistem Pengoperasian DOS**

**Aziz B. Romli
Azmi B. Mahmood**

Institut Sains Komputer,
Universiti Teknologi Malaysia
Jalan Semarak, 54100, K.Lumpur

Abstrak

Persekitaran sistem operasi pelbagai-kerja merupakan suatu kemudahan dalam melaksanakan beberapa aplikasi secara serentak. Walau bagaimanapun tidak semua komputer berupaya menyediakan persekitaran tersebut, contohnya pada komputer IBM PC. Oleh itu aturcara setempat merupakan salah satu cara dalam menyelesaikan masalah operasi pelbagai-kerja. Kertas kerja ini membincangkan bagaimana hendak membangunkan aturcara setempat dengan menyatakan masalah-masalah yang akan dihadapi serta kaedah untuk mengatasinya.

Katakunci : Aturcara setempat, kunci 'hotkey', DOS, pelbagai-kerja

Abstract

Multitasking operating system is a facility in order to execute several applications simultaneously. But not all computers (such as IBM PC) can provide multitasking capability because of hardware limitations. Therefore resident program is the next best solution to overcome multitasking limitation. This paper discussed about the development of resident program with the problems that will be facing and how to solve it.

Keyword : Resident Program , 'hotkey', DOS, multitasking

1.0 Pendahuluan

Aturcara ingatan setempat atau lebih dikenali sebagai aturcara "Terminate and Stay Resident" atau TSR mulai popular pada pertengahan 80'an. Kaedah ingatan setempat ini berkembang hanya di dalam sistem pengoperasian DOS yang boleh didapati pada komputer keluaran IBM dan yang serasi sahaja. Ini kerana sistem pengoperasian DOS tidak dapat membenarkan beberapa perisian beroperasi dalam masa yang sama atau lebih dikenali sebagai operasi pelbagai-kerja (multi-tasking). Oleh itu, untuk mengatasi masalah yang dihadapi oleh DOS, kaedah ingatan setempat diperkenalkan. Aturcara ingatan setempat bertindak dengan memuatkan aturcaranya sendiri secara latarbelakang (background) kemudian memulangkan kawalan kepada DOS [Michael J. Young, 1989]. Semasa aturcara ingatan setempat berada di latarbelakang, ia tidak melakukan sebarang operasi sehingga suatu keadaan menghidupkannya semula seperti menekan kekunci yang telah dipilih (hotkey). Ini akan menyebabkan aturcara setempat menyampuk aturcara aplikasi yang sedang beroperasi dan aturcara setempat akan terus memberi servisnya tersendiri. Contoh aturcara ingatan setempat yang sering di dapati ialah :-

- a) **SIDEKICK** - utiliti yang menyediakan buku catatan, alat pengira, diari dan sebagainya.

Pengaturcaraan Ingatan Setempat Di Dalam Sistem Pengoperasian DOS

- b) **CLOCK.COM** - untuk memaparkan masa di skrin komputer secara berterusan dengan sampukan yang buat pada aturcara .
- c) **PRINT.COM** - mencetak fail-fail dan operasi dilakukan secara latarbelakang.
- d) **Aturcara virus** - aturcara yang selalunya bertujuan untuk membuat kerosakan pada data .

2.0 Kebaikan Aturcara Ingatan Setempat

Kaedah ingatan setempat ini mempunyai kebaikannya tersendiri iaitu dari segi;

- i) **Cepat** - Penukaran dari satu aplikasi ke aplikasi yang lain dengan pantas tanpa perlu menamatkan suatu aplikasi apabila hendak mengguna aplikasi yang lain. Contohnya apabila aplikasi pemproses perkataan yang sedang digunakan dengan utiliti ingatan setempat seperti SIDEKICK yang berada di latarbelakang. Utiliti SIDEKICK dapat dihidupkan dengan menekan kunci 'hotkey' nya sendiri dan terus menyampuk pemproses perkataan disamping memberi khidmatnya.
- ii) **Pemindahan Data** - Dengan adanya aturcara ingatan setempat, pemindahan data di antara aturcara-aturcara yang tidak sama dapat dilaksanakan. Contohnya untuk mengambil imej skrin grafik supaya datanya dapat diguna oleh aplikasi lain.

3.0 Perkara-perkara Yang Perlu Difikirkan Apabila Menulis Aturcara Ingatan Setempat.

Boleh dikatakan dalam menghasilkan aturcara ingatan setempat, banyak cabaran yang perlu dihadapi kerana ia melibatkan pengetahuan yang luas tentang bagaimana sistem pengoperasian bergerak dan khidmat sampukan yang ada di dalamnya. Ini supaya aturcara ingatan setempat dapat beroperasi sebagaimana aturcara biasa bekerja. Antara perkara yang perlu diberi perhatian adalah:-

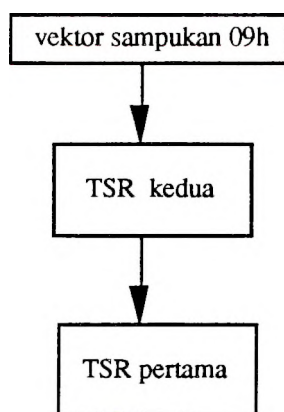
3.1 Kewujudan aturcara ingatan setempat bersama dengan aturcara ingatan setempat yang lain

Oleh kerana setiap aturcara ingatan setempat mempunyai khidmat atau tugasnya sendiri , maka kadangkala timbulnya konflik di antara aturcara-aturcara ingatan setempat yang telah dimuatkan dalam ingatan. Contoh berlakunya keadaan sedemikian ialah apabila sumber seperti perkakasan dan perisian sering dikongsi antara satu sama lain. Perkara-perkara yang menyebabkan berlakunya konflik ialah:-

3.1.1 Kekunci 'Hotkey'

Kebanyakan utiliti ingatan setempat mempunyai kekunci pilihannya tersendiri supaya apabila kekunci tersebut ditekan, ia akan dibawa ke hadapan untuk memulakan khidmatnya kepada pengguna. Jadi pemilihan kekunci yang unik adalah perlu bagi setiap aturcara ingatan setempat supaya tidak berlakunya kekeliruan apabila mengguna aplikasi ingatan setempat. Contoh keadaan yang boleh dilihat ialah apabila ingin menghidupkan utiliti SIDEKICK , biasanya kekunci <ALT> <CONTROL> diambil sebagai kekunci 'hotkey' bagi utiliti tersebut. Sekiranya aturcara ingatan setempat yang hendak dibangunkan nanti menggunakan kekunci yang sama, maka salah satu daripada dua utiliti ingatan setempat akan memberi servisnya sendiri Ini bergantung kepada utiliti yang mana dimuatkan terkemudian, kerana vektor sampukan papan kekunci akan menyimpan alamat aturcara setempat yang terakhir dimuatkan dalam ingatan[Rajah 3.1]. Oleh itu adalah lebih baik mencari kekunci terpilih yang masih belum digunakan oleh sebarang aplikasi komersial di pasaran.

Pengaturcaraan Ingatan Setempat Di Dalam Sistem Pengoperasian DOS



Rajah 3.1 TSR kedua akan dihidupkan dahulu sekiranya kunci terpilih adalah sama dengan TSR pertama

3.1.2 Kekurangan ingatan

Apabila aturcara ingatan setempat dilaksanakan kawasan ingatan yang diambilnya akan berkekalan sehingga komputer dimatikan. Oleh itu apabila beberapa aturcara ingatan setempat dimuatkan terus dalam sistem pengoperasian DOS, sumber ingatan mula menjadi berkurangan dan kesukaran akan dihadapi apabila hendak melaksanakan aturcara yang besar seperti pemproses perkataan ataupun perisian pangkalan data. Bagi mengatasi kekurangan ingatan yang dihadapi ini, beberapa garis panduan untuk memastikan aturcara ingatan setempat beroperasi dengan baiknya iaitu :-

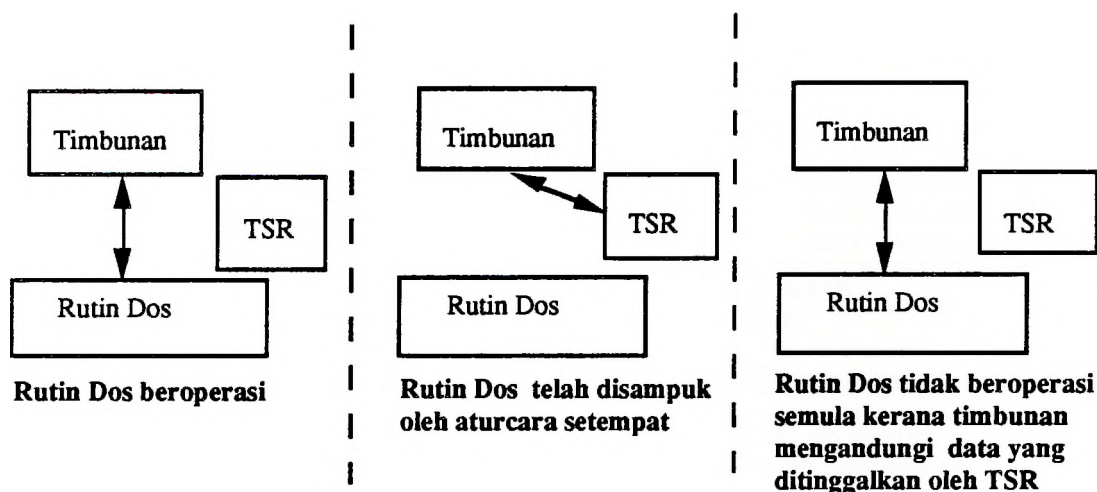
- i) Membenarkan pengguna memilih saiz ingatan yang diperlukan oleh aturcara ingatan setempat. Contohnya SIDEKICK PLUS membolehkan pengguna memilih gabungan utiliti-utiliti tertentu seperti gabungan 'notepad' dengan kalender , gabungan kalender dengan pengira , ataupun sebagainya.
- ii) Penggunaan fail cakera di mana kod atau data boleh disimpan pada storan sekunder. Walau bagaimanapun, kaedah ini bergantung kepada kepantasan storan sekunder apabila membaca atau menulis fail data yang bekerja sebagai fail sementara.
- iii) Penulisan aturcara dalam bahasa himpunan memberikan aturcara yang padat dan kecil, dengan itu ruang yang diguna adalah kecil.
- iv) Penggunaan 'expanded memory' sekiranya ada supaya penyimpanan data boleh dibuat pada kawasan ingatan tersebut.

3.2 Penggunaan sampukan DOS

Apabila aturcara ingatan setempat mengguna khidmat sampukan DOS, masalah mulai timbul apabila berlakunya kerosakan maklumat pada timbunan dalaman DOS (Dos internal stack) . Ini berpunca daripada;

a) Rutin DOS yang tidak kemasukan semula (non-reentrant)

Oleh kerana aturcara ingatan setempat boleh menyampuk rutin DOS pada bila-bila masa, maka kerosakan ke atas timbunan DOS akan berlaku. Keadaan ini berlaku apabila aturcara setempat memanggil sampukan DOS yang semestinya mengguna timbunan yang sama bagi menyimpan data. Aturcara setempat tersebut masih berjalan dengan baik, tetapi sebaliknya masalah timbul apabila kawalan dipulangkan pada titik rutin yang telah disampuk tadi. Rutin DOS akan mengguna semula data daripada timbunan tadi dan akhirnya mengakibatkan kerosakkan pada sistem operasi [Lihat Rajah 3.2]. Ini boleh dikawal dengan menggunakan 'bendera indos' yang bertindak sebagai pembilang untuk mengetahui berapa kali kod DOS dipanggil secara rekursi.



Rajah 3.2 Keadaan DOS Yang Tidak Boleh Melakukan Kemasukan Semula

b) Prefiks Segmen Aturcara (PSP-Program Segment Prefix)

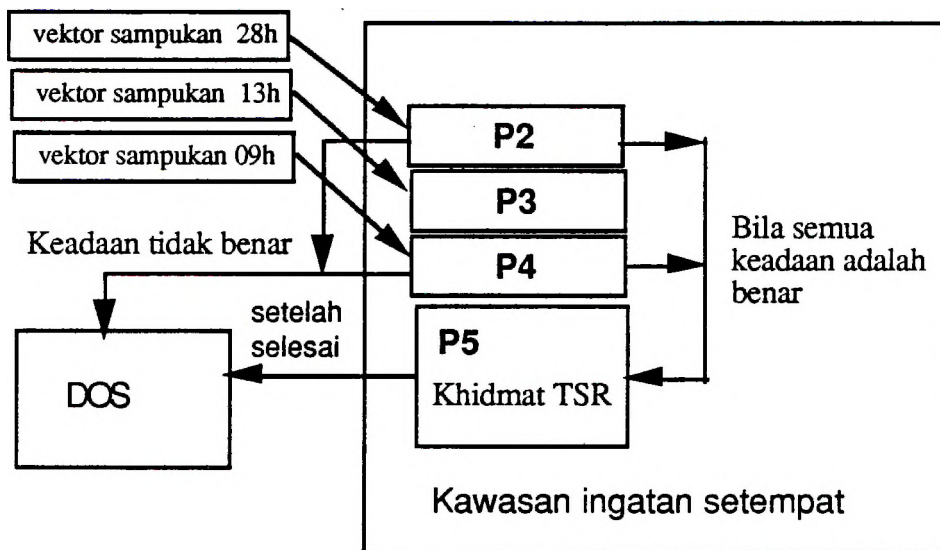
PSP merupakan antaramuka bagi DOS dan aturcara yang dilaksanakan dengan menyimpan maklumat alamat aturcara, supaya DOS dapat menyelenggara aturcara yang sedang terlaksana [Steven Holzner, 1987]. Apabila aturcara setempat menyampuk aturcara yang sedang terlaksana, DOS akan terus mengguna PSP yang asal iaitu daripada aturcara yang telah disampuk tadi. Oleh itu aturcara setempat perlu menukar rekod PSP kepada aturcaranya. Ini untuk mengelakkan aturcara setempat daripada mengguna petunjuk fail aturcara yang tersampuk tadi.

c) Kawasan Pemindahan Data (DTA - Disk Transfer Area)

Sama juga dengan keadaan seperti PSP, aturcara setempat perlu mengubah maklumat DTA kepada kawasan pemindahan data yang disediakan. Ini untuk mengelakkan berlakunya kekeliruan apabila aturcara yang telah disampuk tadi mengguna semula kawasan pemindahan data yang telah diisi semasa aturcara setempat dilaksanakan. DTA amat penting dalam pemindahan fail data dan mencari fail dengan mengguna sampukan DOS 4Eh. Dengan itu kawalan terhadap masalah ini perlu diambil perhatian.

4.0 Pembangunan Aturcara Ingatan Setempat

Semasa hendak membangunkan aturcara ingatan setempat perlu diketahui khidmat sampukan yang telah disediakan oleh DOS dan BIOS (Basic Input-Output System), tujuannya adalah untuk memudahkan lagi penggunaan sumber-sumber melalui khidmat tersebut. Pemahaman terhadap vektor sampukan juga adalah perlu dalam aturcara sampukan ini dan amat penting di dalam penukaran vektor sampukan kepada aturcara setempat yang hendak dilaksanakan (Lihat rajah 4.1).



Rajah 4.1 Setelah aturcara setempat dimuatkan

Di bawah merupakan kod pseudo yang menerangkan bagaimana aturcara setempat dibangunkan dengan khidmat sampukan DOS yang utama iaitu sampukan 31 heksadesimal.

4.1 Kod Pseudo Untuk Membangunkan Aturcara Setempat

P1: Prosedur penyediaan awalan bagi pemuatan aturcara setempat

- simpan alamat segmen timbunan (stack segment) dan penunjuk timbunan (stack pointer).
- simpan alamat DTA dengan khidmat sampukan DOS 2Fh.
- simpan penunjuk kepada bendera indos dengan sampukan DOS 34h.
- simpan vektor-vektor asal sampukan dengan khidmat sampukan 35h iaitu:-
 - vektor sampukan 28h (untuk mengetahui keadaan melahu) ke V28
 - vektor sampukan 13h (sampukan untuk pemacu cakera) ke V13.
 - vektor sampukan 09h (sampukan untuk papan kekunci) ke V09
- setkan vektor-vektor sampukan yang baru dengan khidmat sampukan 25h iaitu:-
 - vektor sampukan 28h kepada alamat P2.
 - vektor sampukan 13h kepada alamat P3.
 - vektor sampukan 28h kepada alamat P4.
- simpan vektor PSP dengan sampukan 51h.
- tamatkan aturcara dengan kod masih kekal dalam ingatan komputer melalui sampukan 31h.

P2: Prosedur sampukan 28h untuk mengesan keadaan komputer melahu (idle state)

- pergi ke vektor asal sampukan 28h yang terdapat pada V28
- uji bendera adakah khidmat aturcara setempat ini sibuk,
 - jika sibuk, keluar dari prosedur ini.
 - jika tidak, pergi ke langkah berikut.
- uji bendera pemacu cakera adakah,
 - jika sibuk, keluar dari prosedur ini.
 - jika tidak, pergi ke langkah berikut.
- uji 'hotkey',
 - jika sama, pergi ke P7
 - jika tidak, keluar dari prosedur ini.

P5: Prosedur sampukan 13h untuk pemacu cakera

- set bendera pemacu cakera kepada sibuk
- buat operasi pemacu seperti membaca atau memasukan data ke disket.
- set bendera pemacu cakera kepada melahu.

P6:Prosedur sampukan 09h untuk papan kekunci

- pergi ke rutin sampukan yang asal di V09.
- uji bendera khidmat aturcara setempat dan bendera pemacu cakera adakah sibuk,
 - jika ya, keluar dari prosedur ini.
 - jika tidak, pergi ke langkah seterusnya.
- uji kekunci 'hotkey',
 - jika sama, pergi ke langkah berikutnya.
 - jika tidak, keluar dari prosedur ini
- uji bendera indos,
 - jika kosong (kod DOS belum dimasuki), maka pergi ke langkah berikut.
 - jika tidak (kod DOS telah diguna berulang kali), keluar dari prosedur ini.
- pergi ke P7

P7:Prosedur Khidmat aturcara setempat

- set bendera khidmat aturcara kepada sibuk
- simpan timbunan segmen dan penunjuk timbunan.
- set segmen timbunan kepada alamat segmen aturcara ini.
- set penunjuk timbunan kepada kawasan data yang telah disediakan iaitu di D1
- simpan semua kandungan daftar-daftar ke dalam timbunan.
- simpan alamat DTA dengan sampukan 2Fh.
- set DTA kepada kawasan yang telah disediakan iaitu di D2.
- buat khidmat aturcara setempat sebenar seperti menyimpan fail atau memaparkan masa.
- setkan semula vektor asal DTA .
- pulangkan semula kandungan daftar-daftar.
- setkan semula segmen asal timbunan dan penunjuk timbunan.
- set bendera khidmat aturcara kepada melahu, menandakan perkhidmatan aturcara setempat telah selesai.
- pulang ke titik asal aturcara yang telah disampuk sebelumnya.

Petuniuk :-

D1 : Kawasan data sebagai timbunan untuk sampukan DOS

D2 : Kawasan data untuk DTA contohnya dengan saiz 512 bait

P - prosedur aturcara

V - Pembolehkan untuk menyimpan vektor sampukan yang asal.

D - Kawasan data

bendera - status

5.0 Kesimpulan

Pada masa sekarang dengan adanya perisian WINDOWS MICROSOFT kebolehan untuk melaksanakan beberapa aplikasi dalam waktu yang sama dapat dilaksanakan. Walau bagaimanapun perisian tersebut memerlukan keupayaan senibina pemproses yang tinggi, contohnya INTEL 80386. Oleh itu aturcara setempat masih diperlukan lagi pada komputer IBM yang serasi dalam mengatasi kekurangan persekitaran pelbagai-kerja.

Rujukan

Steven Holzner,1987 : Advanced Assembly Language on the IBM PC, A Brady Book, Prentice Hall Press,New York.

Michael J. Young,1989: MS DOS Advanced Programming, Sybex/Tech Publications.