DIGITAL TELEPHONE SWITCH: SOFTWARE

J. G. Viveros Talavera, J. P. Abarca Reyna & E. Noriega Solano

Universidad Autónoma Metropolitana – Azcapotzalco
División de Ciencias Básicas, Departamento de Electrónica
Av. San Pablo No. 180, Col. Reynosa Tamaulipas, C.P. 02200, México, D.F.
Tel: 5318-9032 Email: vtjg@correo.azc.uam.mx

ABSTRACT

This paper describes the design of the program of control for a Digital Phone Switch (PBAX) with a maximum capacity of one hundred and twenty-eight lines (extensions and trunks). The control program was designed using object-oriented programming and concurrent programming techniques.

RESUMEN

En el presente trabajo se describe el diseño del programa de control para un Conmutador Telefónico Digital (PBX) con capacidad máxima de ciento veintiocho líneas (extensiones y troncales). El programa de control se diseño con programación orientada a objetos y técnicas de programación concurrente.

1. INTRODUCTION

A concurrent programming model was used for designing the control program. In linear programming, a consecutive sequence is followed using a series of functions or sub-routines that constitute a particular process within the program, therefore, if any specific external action needs to be taken for these sub-routines to continue running, none of the following routines may be run until the running of the sub-routine in question has concluded.

We used a control program with a concurrent programming model. The program will not stop during the process of detecting digits dialed by extension one, but it will detect whether a digit has been dialed or not, either by activating a flag or saving the information available at the time in a log or stack; it may continue checking other processes to detect any change in other devices. When the program has checked its constituent processes, it will then begin checking again from the first process, and so forth.

2. STRUCTURE OF THE CLASSES

Given the necessities of the PABX, we are required of software that is: flexible, modular, reusable, concurrent and that it is expandable. The object oriented programming (OOP) it fulfills all the previous requirements. With the OOP a structure modular of the control program is obtained. The hierarchy of classes that is used for the program of control in the PABX is shown in Fig. 1.
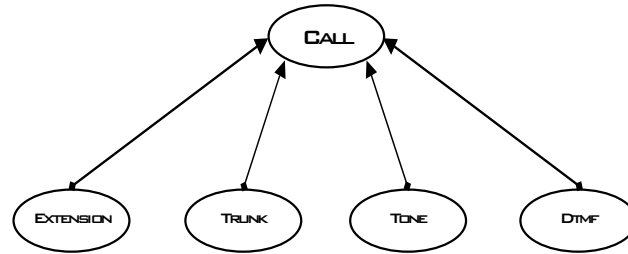
Figure 1. Hierarchy of classes for the program of control.

In the following paragraphs the functions and data members more important of each class are described.

2.1 Functions Member of the Superclass "CALL"

Manda_DC (parameter 1, parameter 2, parameter 3). This function enabled or disable the time slots of the TSAC, the parameters that it uses are: parameter 1 (PTO), the port corresponds the TSAC that is wanted to program, parameter 2 (DC), word of 8 bits necessary to program the TSAC and parameter 3 (FSXRn), time slot of the frame that is enabled or disables.

Switching (parameter 1, parameter 2, parameter 3). In this function the switch matrix is programmed to make the exchange of time slots, the parameters that it uses are: parameter 1 (addr_mem), address data memory location in that is loaded the information of the time slot input, parameter 2 (ran), input time slot and parameter 3 (hab), this parameter be can take two value '1' (enable) and '0' (disable).

Taking advantage of the inheritance property these two functions are used by the subclasses: Extension, Trunk, Tone and DTMF.

2.2 Functions and Data Member of the Class "Extension"

Tsac (parameter 1). In this function the necessary parameters are calculated for the programming of the TSAC and it requires of alone a parameter: parameter 1, can take two you value '1' (enable) and '0' (disable). Starting from this information the three parameters required by the function Manda_DC are calculated, calculated these parameters the function Manda_DC is called.

Matrix (parameter 1, parameter 2). In this function the necessary parameters for the exchange of time slots are calculated, the parameters that it uses are: parameter 1 (Ran), time slot with which leaves to carry out the exchange, that is to say, settles down the connection and parameter 2 (Hab), this parameter can take two value '1' (enable) and '0' (disable).

Ring (parameter 1). In this function ringing signal is sent to subscriber 'B', and it requires of alone a parameter: parameter 1 that is the subscriber number to which is gone to send ringing signal.

Lib_Ext ( ). In this function the subscriber resources are liberated, like they are: the time slots, the variables and the flags (in the case of the time slots they are disabled and in the case of the variable and the flags are reestablished). The data member of the class Extension are: Slot, is the value of the time slot that corresponds it to transmit and to receive. DC, word of configuration of the TSAC. Extn, binary number that corresponds to the extension. Dtmf_Asig, DTMF decoder that has assigned the subscriber (1, 2 or 3). Abon_A, store subscriber number of 'A' if it is 'B'. Abon_B, store subscriber number of 'B' if it is 'A'. Flag_Ext, byte that contains the necessary flags for the connection of the call, the Fig. 2 show the meaning of each contained bit in this variable.
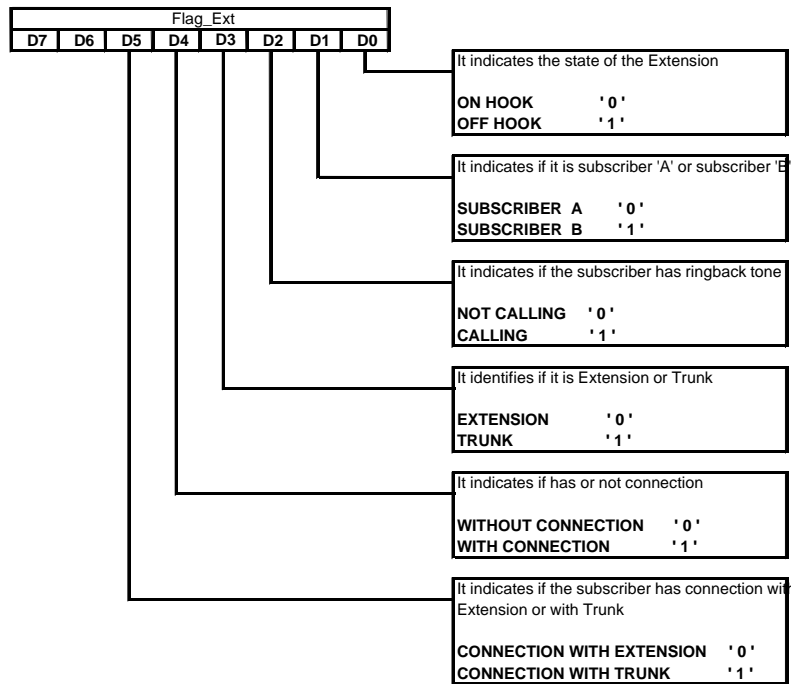
176

Figure 2. Used flags for the object Extension.

2.3 Functions and Data Member of the Class "Trunk"

The class Trunk has three important functions member: *Tsac* (parameter 1), *Matrix* (parameter 1, parameter 2) and *Lib_Tk* ( ). (Tsac and Matrix are the same as in the class "Extension").
Lib_Tk ( ). In this function the trunk resources are liberated, like they are: the slots, the variables and the flags (in the case of the slots they are disabled and in the case of the variable and the flags are reestablished). The data member of the class Trunk are: *Slot*, is the value of the time slot that corresponds it to transmit and to receive. *DC*, word of configuration of the TSAC. *Tkn*, binary number that corresponds to the trunk. *Abon_A*, store subscriber number of 'A' if it is 'B'. *Abon_B*, store subscriber number of 'B' if it is 'A'. *Flag_Tk*, byte that contains the necessary flags for the connection of the call. The Fig. 3 show the meaning of each contained bit in this variable.

2.4 Functions and Dates Member of the Class "Tone"

The class Tone has two important functions member: *Tsac* (parameter 1, parameter 2) and *Matrix* (parameter 1, parameter 2). (Matrix is the same as in the class "Extension"). *Tsac (parameter 1, parameter 2).* In this function the necessary parameters for the programming of the TSAC are calculated, the parameters that it uses are: *parameter 1*, can take three values: 1 (dial tone), 2 (ringback tone) and 3 (busy tone), *parameter 2*, can take two values: '1' (enable) and '0' (disable). Once calculated the three parameters the function *Manda_DC* is called. The data member more important of the class Tone is: *sem_llam*, semaphore to control the disable of ringback tone.
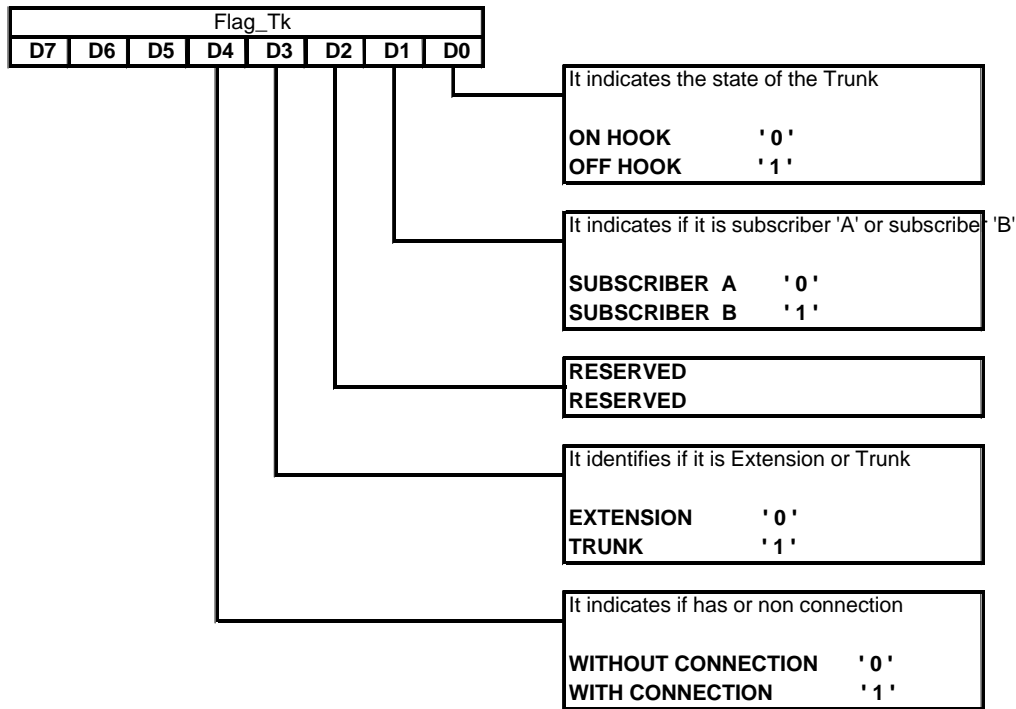
177

Figure 3. Used flags for the object Trunk.

2.5 Functions and Data Member of the Class "Dtmf"

The class Dtmf has three important functions member: *Tsac* (parameter 1, parameter 2), *Matrix* (parameter 1, parameter 2) and *Lib_Dtmf* ( ). (*Tsac* is the same as in the class "Tone" and *Matrix* is the same as in the class "Extension").
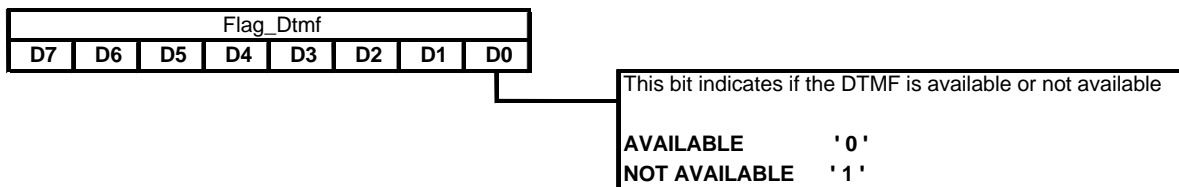

Figure 4. Used flags for the object DTMF.

Lib_Dtmf ( ). In this function the resources of the DTMF are liberated, like they are: the time slot (it is disabled) and the variables and flags take initial values, the DTMF is available for another subscriber.

The members data more important of the class Dtmf are: *Flag_Std*, flag that conserves the state previous of the DTMF. *Flag_Dtmf*, byte that contains the flag of available or not available. The Fig. 4 shows the meaning of each contained bit in this variable.

3. STRUCTURE OF THE PROGRAM OF CONTROL

The Figure 5(a) shows the process that is followed when a subscriber wants to carry out a call, the Fig. 5(b) shown the process for an incoming call. The stage more critic of the whole control program it is the reception of digits. This process requires special attention, because all the digits should be received in real time, if for some flaw of the program a digit is not received the call can get lost. The 95% of the success of

the call depends on the reception of the digits that the subscriber dials. Of the analysis of the Figure 5 it is determined that they exist three types of processes basically: PLP (Process of Low Priority), PNI (Process Not Interrupt) and PPI (Process Priority Interrupt).

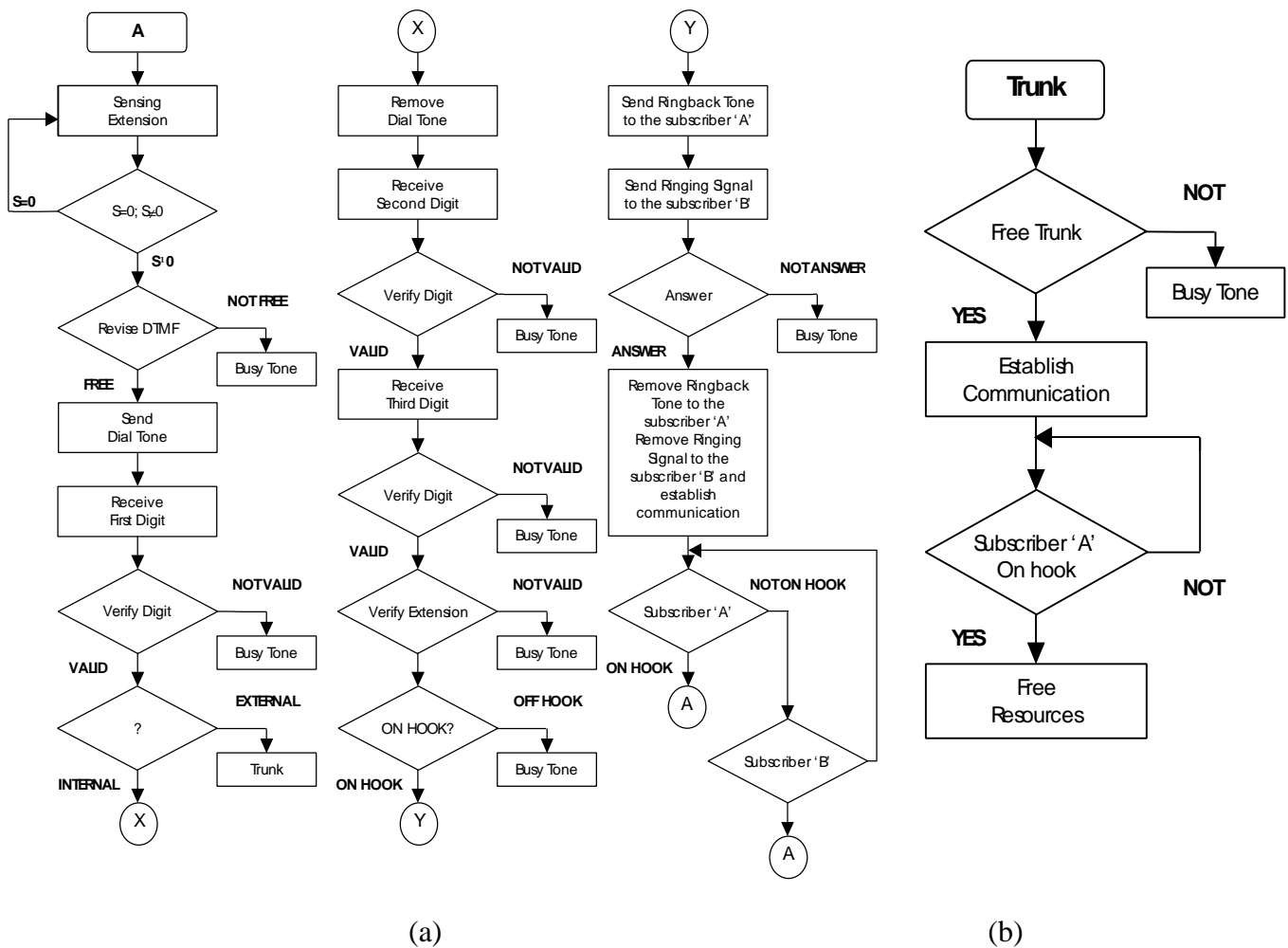(a)   (b)

Figure 5. Flow diagrams (a) internal call y (b) incoming call.

The Figure 6 show the structure of the control program for the PABX in terms of the three types of processes mentioned previously (PLP, PNI and PPI).
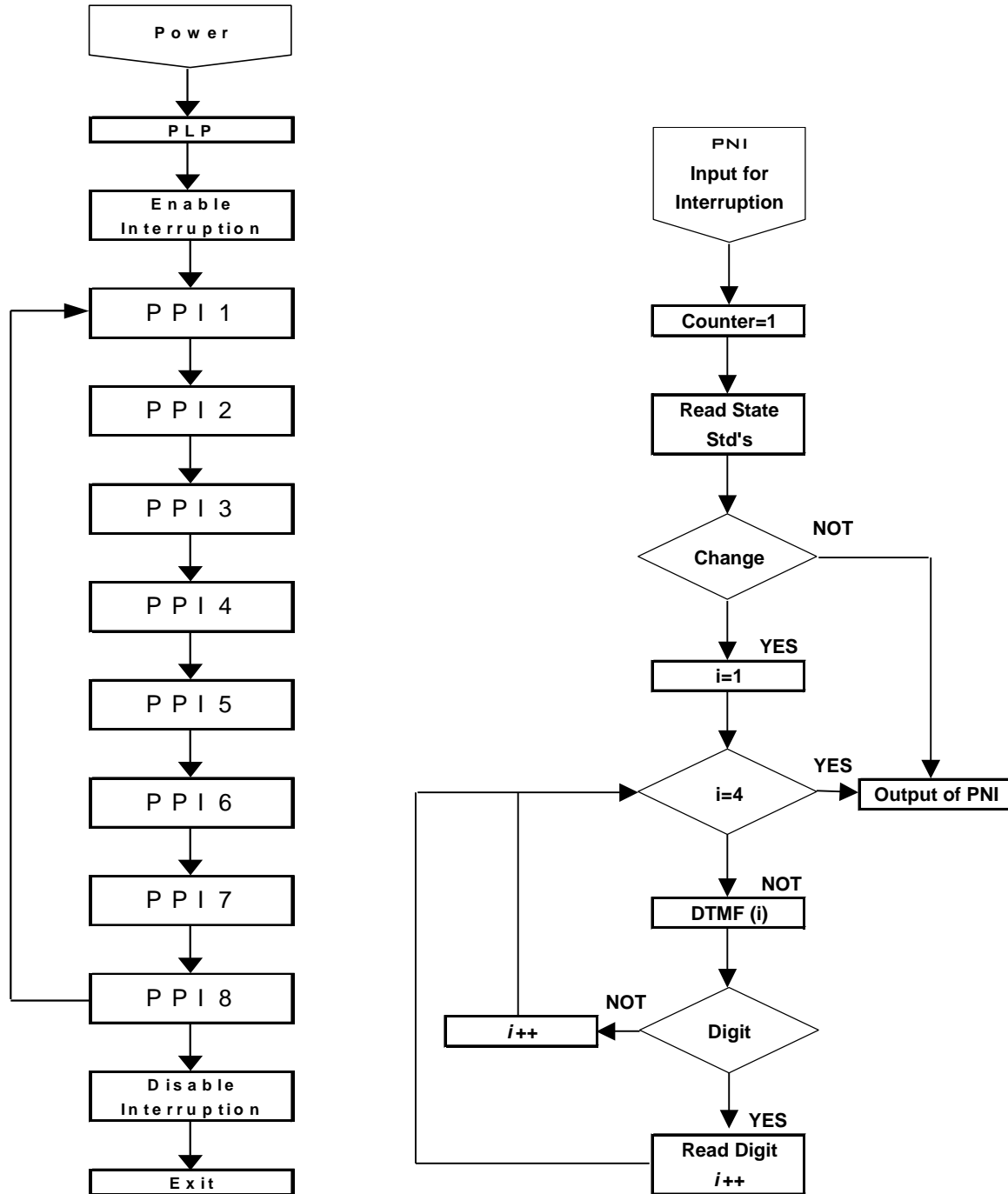
179

Figure 6. Diagram of Flow in Function of PLP, PNI and PPI.

Three forms exist of enabling a process. The first one, for a present event in the PABX (PEP), this event can be: on-hook or off-hook of a subscriber, incoming call for a trunk or reception of a valid digit. The second, for wait list (WL), that is to say, the process revises it or its associate lists continually and if has an element assists him, if the list this empty the process doesn't make anything, and the third are for interruption each 55 milliseconds. The Table I contains the function of each one of the lists used for the control program, the used lists are of type FIFO. It is important to keep in mind that the recovery of an element eliminates it of the list, and if it is not stored in another place it will be destroyed.

180

Table I. Used Lists for the Program of Control.

NAME	FUNCTION
Que_Std	Save the change of the **Std** of each DTMF.
Que_DTMF[3]	Array of three list, each one store the digit received by each **DTMF.**
Que_Dig [3]	Array of three lists, each one stores the process of dialing in which is the subscriber. Digit 1, digit 2 or digit 3.
Que_Tmp[3]	Array of three temporary lists.
Que_POT_IM	Lists that store subscribers to put dial tone.
Que_QT_IM	Lists that store subscribers to remove dial tone
Que_CT_LLAM	Lists that store subscribers with ringback tone.
Que_Tmp_LLAM	Temporary list.
Que_QT_LLAM	Lists that store subscribers to remove ringback tone.
Que_POT_OC	Lists that store subscribers to put busy tone.
Que_QT_OC	Lists that store subscribers to remove busy tone.
Que_QS_TIM	Lists that store subscribers to remove ringing signal
Que_Abon_A	Lists that store subscriber "A" for to connect with subscriber "B".
Que_Abon_B	Lists that store subscriber "B" for to connect with subscriber "A".

3.1 Process of Low Priority (PLP)

This type of processes is executed alone once to the beginning of the program of control and with the disabled interruption, this process include: Process of Cleaning, reestablishes the system of the PABX, the functions that carries out are: to clean the lists, to initialize variables, to reestablish flags, to disable the slots, to disable any connection, to put CS' of the TSACs in '1', to disable ringing signal of the subscribers and to reestablish the relays of trunks.

3.2 Process Not Interruption (PNI)

This process is also known as Interruption Service Routine (ISR); it is the only process of the program that may not be interrupted and is enabled by interruption. The 0x08 interruption is used, a software interruption, corresponding to the PC internal clock with a frequency of 18.1Hz and that lasts 55 msec.; this means that this process is run every 55 msec. The functions it carries out are: reading the Std pin of the 3 DTMFs, decoded in the 0x325 port and reading the digit and storing it in the list of the DTMF that requested the change. Figure 7 shows how usage time is distributed among the process. The NIP does not have a constant run time, it may use the entire 55msec. period or use just 5 msec. The time remains until the next interruption is shared between the IPP that, likewise the NIP, it does not have a constant run time. If an IPP has too much work, it may use all the remaining time left by the NIP, and continues running in the next cycle, and so on, until it finishes its work. Nevertheless, it may happen that the processes have little or no work at all, in which case the eight processes may be run in a single cycle. (Figure 6)
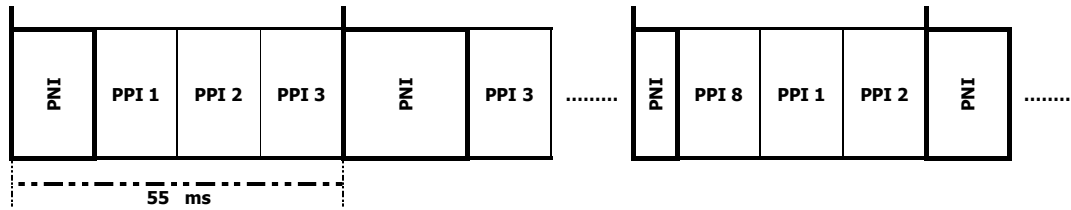
Figure 7. Time of Consumption between PNI and the PPI.

3.3 Process Priority Interrupt (PPI)

It is in this type of processes where are carried out all the control actions and commutation necessary to establish a call, and are these processes the only where the interrupt is enable. These it is the processes that is executed when the PNI leaves time before the following interruption, the PPI this formed by eight subprocces and they are:

IPP 1 (Analysis of digits). This process is enabled by wait list (WL) and has the function of analyzing the digits dialed by the subscriber, if the lists of this process are empty doesn't carry out any operation. If Que_Dig[0] has an element at least (Extension Subscriber 'A') it is taken out of the list and it is verified if already dial first digit, if it has not dialed and the temporized has not expired (7 seconds) enters again, if the temporized already expired enters in Que_POT_OC, and the DTMF is liberated. If first digit is already dial is taken out of the list DTMF[n] corresponding, puts on in zero temporized and the digit is analyzed. If this between 1 and 4 is an internal call, if it is this the first digit dialed subscriber 'A' enters in Que_QT_IM and in Que_Dig[1], and temporized is activated, for any other digit that is not 1 subscriber 'A' enters in Que_POT_OC, and the DTMF is liberated.

If it is 9 it is an external call it is revised if there is available trunk. If there is not available trunk subscriber 'A' enters in Que_POT_OC and the DTMF is liberated. If there is trunk subscriber 'A' enters (Extension) in Que_Abon_A, subscriber 'B' (Trunk) in Que_Abon_B, and the DTMF is liberated. Any other dialed digit is considered garbage and subscriber 'A' enters in Que_POT_OC, and the DTMF is liberated. If Que_Dig[1] has an element at least (Extension Subscriber 'A') it is taken out of the list and it is verified if already dialed second digit, if it has not dialed and the temporized has not expired (1 minute) enters again, if the temporized already expired enters in Que_POT_OC and the DTMF is liberated. If dial second digit is already taken out of the list DTMF[n] corresponding, puts on in zero temporized and the digit is analyzed, if it is valid digit subscriber 'A' enters in Que_Dig[2], and temporized is activated, if digit is not valid subscriber 'A' enters in Que_POT_OC and the DTMF is liberated.

If Que_Dig[2] has an element at least (Subscriber 'A' Extension) it is taken out of the list and it is verified if already mark third digit, if it has not marked and the temporized has not expired (1 minute) enters again, if the temporized already expired enters in Que_POT_OC, and the DTMF is liberated. If mark third digit is already taken out of the list DTMF[n] corresponding, puts on in zero temporized and the digit is analyzed. For any digit not valid subscriber 'A' enters in Que_POT_OC, and the DTMF is liberated. In the event of third valid digit the DTMF is liberated and it is verified that subscriber 'B' this on-hook and nobody him this calling, if this on-hook and somebody him this calling subscriber 'A' enters in Que_POT_OC. If subscriber 'B' is on-hook and nobody him this calling, flag of *subscriber 'B'* in subscriber 'B' and flag of *calling* in subscriber 'A' are activated, ringing signal is send to subscriber 'B', ringback tone to subscriber 'A', and subscriber 'A' enters in Que_CT_LLAM, and temporized is activated.

IPP 2 (Detection of state change in subscribers). This process is enabled by PEP, the function of this process is to revise the state of the subscribers (on-hook or off-hook), if the state of the subscribers is the same one that that of the previous cycle is not carried out any operation, if there is some change in the subscribers regarding the previous cycle the necessary operations they should be carried out. Two event types are those that can be presented in this process and to enable it: the on-hook or the off-hook of a subscriber.

Off hook. The first thing is to lift the flag *off hook* of the subscriber and to verify if it is subscriber 'B' or subscriber 'A'. If is subscriber 'B' it is verified with who goes to establish the connection, with trunk or with extension. If it is with trunk, subscriber 'B' enters in Que_QS_TIM, and in Que_Abon_B. If the connection is with extension, subscriber 'B' enters in Que_QS_TIM, and subscriber 'A' in Que_QT_LLAM. If is subscriber 'A' it is verified if there is available DTMF, if there is not enters in Que_POT_OC. If there is available DTMF, DTMF is reserve and subscriber 'A' enters in Que_POT_IM.

On hook. The first thing is to verify if has connection with extension or with trunk. If doesn't have connection their *on hook* flag is active and the resources are liberated. If has connection it is verified if it is subscriber 'B' or subscriber 'A'. If is subscriber 'B', has connection with extension or with trunk. If is extension whit who has connection the flags of *connection* of subscriber 'A' and subscriber 'B' are disable, resources of subscriber 'B' are liberate and subscriber 'A' enters in Que_POT_OC. If connection is whit trunk, relay is closed and resources of subscriber 'B' (Extension) and subscriber 'A' (Trunk) are liberate. If is subscriber 'A', has connection with extension or with trunk. If the connection is whit extension put *connection* flag off of subscriber 'A' and subscriber 'B', liberate resources from subscriber 'A', and 'B' enters in Que_POT_OC. If the connection is whit trunk close relay and liberate resources of subscriber 'A' (Extension) and subscriber 'B' (Trunk).

IPP 3 (Control for the dial tone). This is a process that is enabled by WL, the function of this process it is to put and to remove dial tone. Two list Que_POT_IM, and Que_QT_IM are revised, if none of the two lists contains elements is not carried out any operation in the process. If the list Que_POT_IM has an element at least (Extension Subscriber 'A') it is taken out of the list and they are carried out the following connections: subscriber 'A' whit DTMF and subscriber 'A' whit dial tone, furthermore the mentioned subscriber 'A' enters in Que_Dig[0], and temporized is activated. If the list Que_QT_IM has an element at least (Extension Subscriber 'A') it is taken out of the list and the connection is liberated among dial tone and subscriber 'A'.

IPP 4 (Surveillance of trunks). This is a process that is enabled by PEP and it doesn't use any list type. The function of this process is to revise the state of the trunks, if an incoming call exists and the state of they trunks it is the same one that that of the previous cycle is not carried out any operation, if there is some change in some of they trunks regarding the previous cycle, it is necessary to carry out some functions. When an incoming call the is detected, first it is verify if there is a free extension to assist the call, if there is not it is let that the public switching telephone network temporize the call and send busy tone having once expired the temporized. If there is extension to assist the call the flags of *off hook* of the trunk, *subscriber 'B'* of extension and *connection with trunk* of extension are activated.

IPP 5 (Control for disable ringing signal). This is a process that is enabled by WL, this process has the function of disable ringing signal. The list Que_QS_TIM is revised, if there is not element is not carried out any operation. If the list has an element at least it is taken out and it is verified if has connection with trunk or with extension. If has connection with trunk disable ringing signal of subscriber 'B', enters subscriber 'A' (Trunk) in Que_Abon_A and relay opens up. If has connection with extension disable ringing signal of subscriber 'B', and enters subscriber 'A' (Extension) in Que_Abon_A.

IPP 6 (Control for ringback tone). This is a process that is enabled by WL, the function of this process it is to remove ringback tone. The list Que_QT_LLAM is revised, if there is not element is not carried out any operation. If the list Que_QT_LLAM has an element at least (Subscriber 'A' Extension) it is taken out of the list and disable the flag off *calling*.

IPP 7 (Control for busy tone). This is a process that is enabled by WL, the function of this process it is to put and to remove busy tone. Two list Que_POT_OC and Que_QT_OC are revised, if none of the two lists contains elements is not carried out any operation in the process. If the list Que_POT_OC has an element at least (Extension Subscriber 'A' or 'B') it is taken out of the list and send it busy tone. If the list

Que_QT_OC has an element at least (Extension Subscriber 'A' or 'B') it is taken out of the list and the connection among busy tone and the extension is liberated.

IPP 8 (Establishment of connection). This is a process that is enabled by WL, the function of this process it is to establish the connection among the subscribers 'A' and 'B'. Two list Que_Abon_A and Que_Abon_B are revised, if none of the two lists contains elements is not carried out any operation in the process. If the list Que_Abon_A has an element at least (Subscriber 'A' Extension or Trunk) it is taken out of the list and is carried out the commutation among having subscriber 'A' and subscriber 'B'. If the list Que_Abon_B has an element at least (Subscriber 'B' Extension or Trunk) it is taken out of the list and is carried out the commutation among having subscriber 'B' and subscriber 'A'.

4. RESULTS

The designed program fulfills the objective. While doesn't register an incoming call or that some subscriber off-hook, the program it is while waiting for some of these two events. It was it carried out the following tests for the PABX. Connection between two extensions, Connection among an extension and a trunk, Connection among a trunk and an extension (incoming call) and the standard temporizations were check. An extreme case is presented when there is not available receiver of digits for a subscriber or in case to an incoming call an extension cannot be assigned to be busy. In anyone of these two cases the call gets lost.

5. CONCLUSIONS

Thanks to the OOP the control program is to modulate. This represents an advantage since an application you can subdivide in parts more small, this way the program has it in different files, like they are files of interfaces (.h) and implementation files (.cpp). Thanks to the concurrence several processes in a parallel way can be assisted, for example we can receive digits of the three receivers in a simultaneous way without losing information. Another advantage that offers the software is the optimization of the memory in time of execution, this is, if the program is in wait state any object that occupies memory doesn't exist, alone if change is detect in some extension or trunk, the corresponding object is believed to assist the call. Lastly the software was design so that in a future they can be added user's facilities.

6. REFERENCES

[1]   Abarca Reyna J. P., Noriega Solano E., *Conmutador Telefónico Digital*, Proyecto Terminal de Ingeniería Electrónica, UAM-A, 2001.

[2]   Booch Grandy, *Análisis y Diseño Orientado a Objetos***,** (Addison Wesley México), (1998).

[3]   Ceballos Francisco Javier, *Enciclopedia del Lenguaje C*, (Alfaomega, España) (1997).

[4]   Joyanes Aguilar Luis, *Programación Orientada a Objetos***, (**McGraw-Hill, España), (1996).

[5]   References in Internet: www.borland.com, www.programadores.com.

[6]   Schild Herbert, *Turbo C/C++, Manual de Referencia***,** (McGraw-Hill, México), (1992).

Authors Biography

Jaime Pedro Abarca Reyna

He was born in Mexico City, he studied electronic engineering at Universidad Autónoma Metropolitana –UAM- (Autonomous Metropolitan University). He currently studies a Ph.D in Telecomunications at CINVESTAV in Mexico City. He is mainly interested in sistems of fixed and mobile telephony.

Edmundo Noriega Solano

He was born in Mexico City. He studied electronic engineering at Universidad Autónoma Metropolitana –UAM- (Autonomous Metropolitan University). He got his bachelor´s degree with the creation and desing of a PBX digital. Currently he works as teacher-investigator in the electronics deparment at UAM- Azcapotzalco. He is mainly interested in sistems of fixed on mobile telephony.

José Guadalupe Viveros Talavera

He studied his bachelor degree in phisics in the faculty of sciences at Universidad Nacional Autónoma de México (National Autonomous University of Mexico). He got a master´s degree in instrumentation in electronics and telecomunications at Centro de la Investigación Científica y Estudios Superiores de Ensenada, B. C., México (Scientific Research and Higher Education of Ensenada, B. C., Mexico). He has worked in several teaching and researching institutions as well as in private telecomunications firms. He currently works as teacher-researcher in the electronics department in UAM-Azcapotzalco. His professional interes centers in fixed and mobile telephony.