MASTER

# Formal description of signalling systems and their interworking : signalling system no. 7 and R2, using CCS and the combination algorithm

Dijkhuis, I.J.W.G.

*Award date:*
1992

Eindhoven University of Technology
department of electrical engineering
digital systems, EB

# Formal description of signalling systems and their interworking

## signalling system no.7 and R2, using CCS and the combination algorithm

I.J.W.G. Dijkhuis

Masters thesis

February 1992

Supervised by prof.dr.ir. C.J. Koomen and prof.ir. J. de Stigter

# Summary

The role of modern telecommunications in modern society is growing fast. New services are introduced and the use of the current services offered is increasing. The telecommunications network must be able to handle this. Therefore the signalling system, the part that ensures telecommunications traffic to arrive at its right destination, must be renewed. This is done gradually, thus requiring interworking units between the old and the new system. In this report, Signalling System no.7 is dealt with.

Formal techniques, such as CCS, the calculus of communicating systems, can be used to describe signalling systems. Indication is made on how to translate particular aspects, e.g. timers, in a description.

This report will focus on the deriving of interworking of Signalling System no.7 to Signalling System R2.

The CCITT specifications of those systems, presented by means of SDL, are translated into CCS. After abstraction and refinement, these are combined to obtain an interworking unit according to the combination algorithm. The steps in this process and the assumptions made are mentioned.

CCS and the combination algorithm are good means to describe systems and derive interworking, though the set of expressions that results is large.

# Contents

# 1. Introduction

The objective of the graduation project described in this report is to enlarge the knowledge of Signalling System no.7 in the digital systems group and to study the application of formal techniques such as CCS to the description of signalling systems.

The role of modern telecommunications in modern society is growing fast. New services are introduced and the use of the current services offered is increasing. The telecommunications network must be able to handle this. Therefore the signalling system, the part that ensures telecommunications traffic to arrive at its right destination, must be renewed. This is done gradually, requiring interworking units between the old and the new system. Signalling system number 7 is being implemented worldwide and it is replacing system R2 in the Netherlands, France and Sweden.

This report deals with the interworking of Signalling System no.7 and Signalling System R2. It uses CCS, the calculus of communicating systems, to describe the systems formally and the interworking unit is derived according to the combination algorithm.

Chapter 2 gives an explanation of the principles of telecommunications and signalling.

Chapter 3 shows the operation of a signalling system and its signals in the description of Signalling System R2.

Chapter 4 describes signalling system no.7 and its telephone user part.

Chapter 5 treats interworking as the CCITT recommends it and the way it is derived in this report.

Chapter 6 is an outline of the theory and notational conventions of CCS, used in this report.

Chapter 7 shows how the preceding applies to the signalling systems no.7 and R2.

Chapter 8 deals with the actual interworking, using the combination algorithm.

Finally, conclusions and recommendations for further study are made in chapter 9.

# 2. Signalling in telecommunications networks

This chapter will deal with principles of telecommunications networks in general, signalling and call processing in particular. At the end, a brief overview of the evolution of signalling systems will be given.

## 2.1 Telecommunications networks

A telecommunications network consists of terminals, transmission links and switching centres. The telephone system is the most visible switching system, but other services are telex, telegraph, television, video conferencing, and mobile communications. These services will be integrated in the future, evolving into an *integrated services digital network, ISDN,* providing end-to-end digital connectivity to support a wide range of services, to which users have access by a limited set of standard multipurpose user-network interfaces.

There are different kinds of switching for the various sorts of telecommunications services [Hills,79], which has consequences for the signalling system.

*Circuit switching* sets up a circuit between two terminals which then interchange information directly. Examples are the telephone and telex network.

In *message switching*, or *store and forward switching*, the system stores messages and delivers them at their destinations at some later time, maximizing the utilisation of links by queueing messages and not depending on setting up a whole end-to-end connection. This kind of switching is applied to teleprinters.

*Packet switching* is a midway between the former types of switching. Information from the source, while being generated, is put in short packets. These packets are transmitted to their destination, possibly in such a rapid way that the connection seems to be an end-to-end circuit, but still with the advantages mentioned under message switching.

In *stored program control, SPC,* exchanges, the relationships between equipment and subscriber numbers are recorded in computer storage and therefore more easily alterable and accessible [Redmill,90]. The early systems combined SPC with electromechanical switching devices like crossbar and reed relais. Using semiconductor switches for telephony was hindered by worse transmission linearity and off-resistance than the electromechanical switches and the inability to handle the high voltages and ringing current required by the conventional telephone. The shift towards digital telephony, overcoming these disadvantages, was influenced by the introduction of digital transmission to the public switched networks and the development in integrated circuit technology.

Digital transmission of analogue signals uses *pulse code modulation, PCM,* shown in figure 1. In this scheme, analogue signals are sampled and these samples are quantized and translated into binairy words before the information is transferred over the digital network. At the destination, the binairy words are decoded and converted back into analogue signals. In telephony, the sampling rate is 8 kHz and the words consist of eight bits.

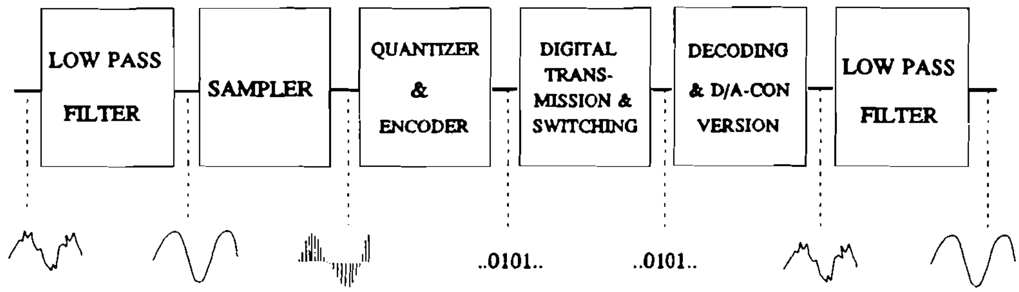



*figure 1. stages in the PCM process*

Channels are combined for transmission on a line, either by using frequency division multiplexing or time division multiplexing. In *frequency division multiplexing, FDM,* each channel is allocated a dedicated band of frequencies. *Time division multiplexing, TDM,* allocates dedicated periods of time, the so called time slots, to each channel.

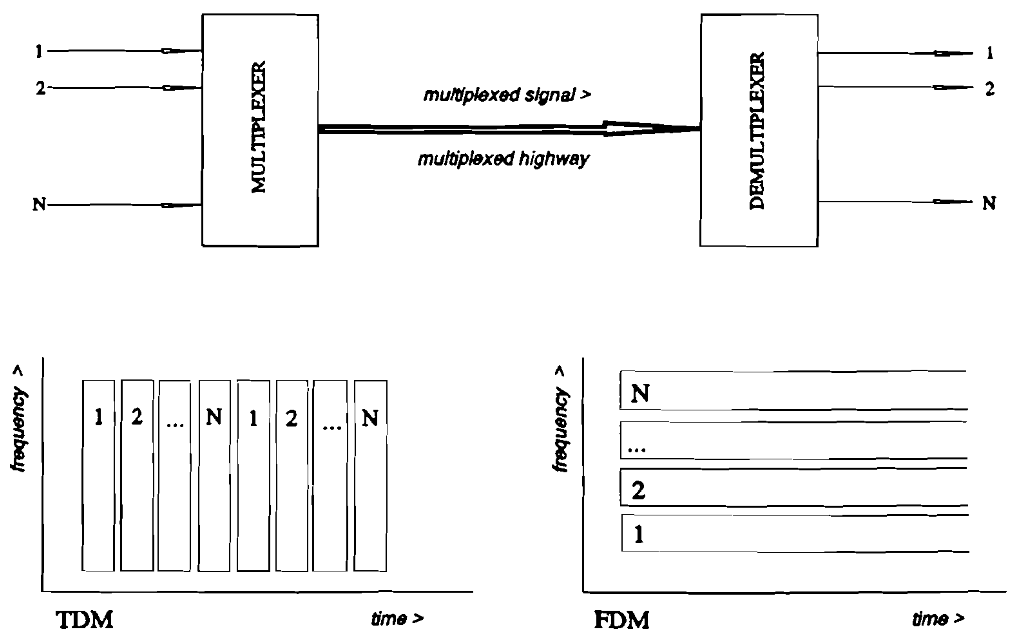



*figure 2. a multiplex system and its allocation of channels in TDM and FDM*

The introduction of digital exchanges saved analogue-digital conversion at the end of transmission lines and eliminated the need for multiplexing equipment for lines terminating at the exchange. The high voltage and DC-path requirements of the subscriber are handled in interface units at the periphery of the exchange.

Advantages of storage program control digital exchanges are:

- flexibility, in the long term the exchange can be upgraded without disruption of the service by incorporating new software or adding new hardware, and short term flexibility due to changing the exchange simply by changing the data, for instance rerouting to avoid congestion,
- subscriber facilities like call tracing and follow me,
- administration facilities that would be expensive or labour-intensive otherwise,
- high speed call set up,
- accommodation saving, relative to analogue exchanges,
- ease of maintenance,
- quality of connection, regarding noise, call loss and stability,
- potential for other services like data transmission,
- lower costs than the analogue ones and
- less installation time.

The standard CCITT frame format for a 30 channel PCM system contains 32 time slots, 30 for data, one for signalling and one for frame alignment, line-systems alarms and network-control signals.

Manufacturers in telecommunications used to work with own standards and specifications. This caused great difficulty to telecommunications administrations that use equipment of different suppliers. Therefore, the CCITT, Comité Consultatif International Télégraphique et Téléphonique or the international consultative committee for telephone and telegraph, issues recommendations on specifications for telecommunications. National telecommunications administration and the industry take part in the CCITT. The CCITT is affiliated with the International Telecommunication Union, ITU, that is an agency of the United Nations. In North America, the American National Standards Institute, ANSI, is the body in charge of protocol specifications for North American networks. It also co-ordinates submitting contributions to the CCITT study groups as well as adopting CCITT standards for North American networks [Modarressi,90].

## 2.2 Signalling

Signalling in telecommunications networks is the interchange of information between different functional parts of the telecommunications system in order to secure the right operation of the system [Welch,79]. Its structure can be compared with a language [Kroes,79]. Protocols define its grammar and its words are signals, that, like words, can have a different meaning in different circumstances. The letters of the signals are bits. Signalling comprises

- *subscriber line signalling*, the transfer of information from subscribers, calling and called, to the switching equipment,
- signalling within the exchanges, and
- the transfer of information over the network.

Signalling within the exchanges is independent of the chosen signalling system and will therefore not be considered.

A signalling system performs supervisory, selection and operational functions.

*Supervisory functions* detect or change the state or condition of elements of the network and the subscribers for instance seizure or clearing.

*Selection functions* deal with call connection set up.

*Operational functions* are necessary to optimize the use of the network and handle system facilities, management and administration. Typical features are information about congestion in the network, outages of elements of the network and call charging information.

A function is represented by one or more signals. Signals are classified in line signals and selection signals. Line or supervisory signals are the signals between supervisory units that the state of the line like seizure and clearing, and selection or address signals are the signals between control units such as registers in order to set up a connection. Most operational functions are represented by selection signals.

There are two signalling techniques, *on-speech path signalling*, meaning the signalling information is transmitted on the data path, and *common channel signalling* in which one channel carries the signalling information for a number of lines. On-speech signalling is relatively slow, limited in information capacity and possibility for non call related signalling information. It may also be inable to signal during the speech period and is expensive because of the per speech circuit signalling provision. Common channel signalling* is faster, can deal with a large number of signals, is able to handle signals during the speech period and is flexible to change or add signals to the system. However, because the signalling information follows another path, the data channels need to be checked. If the digital circuitry does not have some fault indication features, the signalling system must provide in this by performing a *continuity check*. Telephone networks used to be of the first kind, but because of the development of computer techniques in switching, common channel signalling will be applied in the future. The main advantage is cost saving, but on-speech signalling will still be used for some years.

There are three ways in which the common channel signalling links are associated with their dependent traffic links [Redmill,90]:

*-associated signalling,*

the messages relating to the traffic circuits connecting two exchanges are conveyed over signalling links directly connecting the two exchanges,

*-non associated signalling,*

the traffic circuits are directly routed between the two exchanges, while the signalling messages are through several signalling links between those two exchanges with different paths at other times, and

*-quasi associated signalling,*

the signalling messages follow a predetermined path through several links in tandem and the traffic is routed directly to the exchange.

* The abbreviation CCS will be reserved for the calculus of communicating systems.

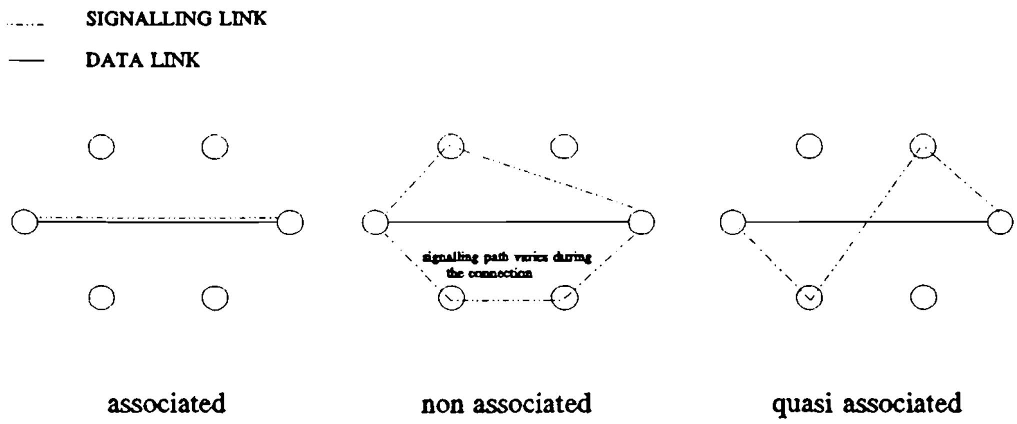



*figure 3. associating signalling with traffic links*

Common channel signalling is used in a number of applications. Signalling system no.7, described in chapter 4, is the modern internationally agreed standard for inter-exchange signalling in national and international networks. It should be noted that in advance of a fully specified system, national variants of the system were introduced.

There are two forms of *subscriber access signalling*: *primary rate access* from *private automatic branche exchanges, PABX*, and *basic access* from single exchange lines. There are CCITT specifications for both forms, but national versions were introduced before the CCITT standards. Common channel signalling is applied in inter-PABX signalling when PABXs are linked by leased lines using the digital private network signalling system DPNSS, for instance. Other, non-circuit related applications are control of intelligent networks, the transfer of location information in cellular telephone systems and the transfer of billing or network-management data.

## 2.3 Call processing

Call processing deals with processing the digits, setting up an appropiate path through the network, monitoring the call and finally, clearing it down [Redmill,90].

In case of *overlap operation* the reception and onward transmission at switching centres are overlapping processes. In *en bloc operation*, the address signals are assembled into one or more blocks for onward transmission, each block containing sufficient information to enable onward routing to be determined by the switching centres which receive it. In *digit-by-digit signalling*, call set-up is tried after reception of one digit.

The post dialling delay between the end of the dialling and the time the connection is set up is the most with non-overlapping en bloc and the least with overlapping digit-by-digit [Kroes,79].

*End-to-end signalling* is a method in which signals are passed from one end to the other of a multilink connection without signal processing at intermediate switching points.

In *link-by-link signalling* the signals are received and passed on at each intermediate switching point between the ends of a multilink connection.

In case of *compelled signalling*, every signal must be acknowledged in the opposite direction before another signal may be sent.

A call may come from a subscriber terminated on the exchange, an *originating call*, or from a distant exchange, an *incoming call*. An incoming call to a subscriber terminated at the exchange is a *terminating call*, to another exchange a *transit call*. An originating call may be to another exchange, an *outgoing call*, or to a subscriber terminated at the same exchange, a *terminating call*, or is a *facility call* when it sets up or cancels a customer facility.

| type of call | source | destination |
|---|---|---|
| originating | this exchange | |
| incoming | another exchange | |
| terminating | | this exchange |
| outgoing | this exchange | another exchange |
| facility | this exchange | |
| transit | another exchange | another exchange |

*table 1. types of call*

The main actions primarily depend on whether the call is outgoing or terminating. Minor variations depend on the call being originating or incoming. Functions to be carried out are digit analysis, digit translation, switch-path set-up, charging and enabling customer facilities.

### Digit analysis

For an incoming call, digit analysis is only required to determine whether it is terminating or transit since most of the analysis has been performed at the exchange where the call originated. Digits define the type of call being initiated like service calls, operater calls, emergency calls, local, non-local and international calls and calls concerning customer facilities. The signal is stored, a timer is set and the dial tone removed. If any further input signals have not been received once the timer has expired, the call can be aborted. For every call attempt, the *subscriber's class of service, COS, record* is checked. Specific action can be undertaken in case of a class of call that is barred, or in case of inadmissible digits.

### Digit translation

The purpose of digit translation is to access lists, by using the input digits, to determine a circuit on which the call can leave the exchange and to acertain the routing digits that must be sent to the next exchange in case of a non terminating call and the rate at which the call should be charged.

### Switch-path set-up

When digit analysis or translation has shown the call to be terminating on the exchange, the line is checked to be not engaged. If it is engaged, the busy tone must be sent to the subscriber. If not, the line is seized by setting the busy bit in the COS record. A path through the switch of the exchange will be set up and a ringing current will be sent to the called subscriber. If this is not possible, a busy tone will be sent. At his point, the call is 'waiting' and either

- an off-hook signal from the callee has not been received nor an on-hook signal from the calling subscriber for a certain time,
- the calling subscriber clears down, in which case the on-hook signal is sent or
- the off-hook signal of the called subscriber is received.

The ringing current is disconnected and in the first two possibilities the circuitry is cleared, in the last one transmission is provided on the switch path so that conversation can take place.

### Charging

Charging is carried out by a local exchange for originating calls and requires information like the duration of the call, the time of day, the day of the week and rates applicable.

### Customer facilities

Certain input digits set up, clear down or change facilities, which are offered to the subscriber. These facilities depend on the equipment and signalling system used. Examples of these facilities are short-code dialling, alarm call, and transfer of calls.

## 2.4 Evolution of signalling systems

Early methods of interoffice signalling involved dial pulse techniques as with step-by-step machines [Modarressi,90]. Common control switching systems introduced *single-frequency, SF,/multi-frequency, MF*, signalling techniques. In the SF method, the busy/idle state of the circuit is determined by a single-frequency tone, usually 2600 Hz, on the trunk when it was idle. The absence of that tone indicated an off-hook condition. Address information was passed by

MF tones. This signalling was *inband*, which means that it was conveyed in the same bandwith as speech. The call set-up times were long, 10 -20 seconds, and only limited information could be transferred, restricting the capabilities of the network. Also, tone signalling is subject to fraud with tone generators. System R2 of the next chapter uses MF signalling.

Electronic processing enabled common channel signalling. In 1976, AT&T introduced a common channel signalling system based on the CCITT recommendations for signalling system no.6 [CCITT,89]. Initially, it only provided for trunk signalling and routing was based on a permanent virtual network. Messages were routed by a band and label scheme, with 512 bands of 16 trunks to assign to signalling links. The incoming band and terminal were translated to an outgoing band and terminal for routing. The no.6 protocol is not layered to avoid the overhead of layered structures and thus meeting the efficiency needed to work at 2.4 kbit/s rates. Though more advanced than the previous signalling systems, drawbacks like large administrative efforts for banded routing, limited message lenghts and low speed links incited the CCITT to develop a new system for digital trunks: signalling system no.7.

# 3. Signalling system R2

Signalling System Regional 2 can be used for integrated international and national signalling. It is suitable for both automatic and semi-automatic working, for use on satellite links and 3 kHz spaced carrier links. System R2 is fully described in Blue Book fascicle VI.4 [CCITT,89].

There are two versions of line signalling: the analogue version for carrier systems which is outband and on-speech path and the digital version for link-by-link, channel associated PCM systems. This report will only deal with the digital version.

The interregister multifrequency selection signalling is the same for the digital and analogue version.

## 3.1 Line signalling

The digital version is link-by-link using two signalling channels in each direction of transmission per speech circuit. These channels are referred to as $a_f$ and $b_f$ for the forward direction and $a_b$ and $b_b$ for the backward direction. $a_f$ identifies the operating condition of the outgoing equipment and the condition of the calling subscriber's line. $b_f$ indicates a failure in the forward direction to the incoming switching equipment. $a_b$ gives the condition of the called subscriber's line, on or off hook. $b_b$ indicates the idle or seized state of the incoming switching equipment (table 2).

| $a_f$ | $b_f$ | $a_b$ | $b_b$ | state |
|---|---|---|---|---|
| 1 | 0 | 1 | 0 | idle/released |
| 0 | 0 | 1 | 0 | seized |
| 0 | 0 | 1 | 1 | seizure acknowledged |
| 0 | 0 | 0 | 1 | answered |
| 0 | 0 | 1 | 1 | clear back |
| 1 | 0 | 0 | 1 | clear forward (called party on hook) |
| 1 | 0 | 1 | 1 | clear forward (called party off hook) |

*table 2. signalling code on the PCM line under normal condition*

Forward line signals are:

-seizing,

sent at the beginning of the call to initiate transition of the circuit at the incoming end from the idle state to the seized state,

-clear forward,

sent to terminate the call or call attempt and to release in the incoming exchange and beyond it all switching units held on the call and

-forward transfer,

the operator on the outgoing end wants the help of an operator at the incoming international exchange. This signal is not standardized.

Backward line signals, sent to the outgoing exchange are:

-seizing-acknowledgement,

indicates transition of the equipment at the incoming end from the idle to seized state,

-answer,

indicating the called party has answered the call,

-clear-back,

indicating the called party has cleared,

-release-guard,

indicates the fully effective returning of the switching units at the incoming end of the circuit to the idle state in response to a clear-forward signal and

-blocking,

sent on an idle circuit to cause engaged conditions to be applied to this circuit, guarding it against subsequent seizure.

## 3.2 Interregister signalling

The interregister signalling is performed end-to-end using a 2-out-of-6 in-band multifrequency code with forward and backward compelled signalling. These frequencies do not interfere with other signals.

*End-to-end signalling* is a method for signalling between registers over two or more links in tandem without signal regeneration in intermediate exchanges, see figure 1.

The outgoing R2 register is at the outgoing end of a signalling system. It controls the call set-up over the whole signalling section. It sends forward interregister signals and receives backward interregister signals. This register receives information via preceeding links of the connection operating with another signalling system, or from a subscriber's line.

The incoming R2 register at the incoming end of a link receives forward interregister signals via the preceeding link or links and sends backward interregister signals.

Only the address information needed for routing the call through a transit exchange is transferred from the outgoing register to the incoming registers which come into action one after another.

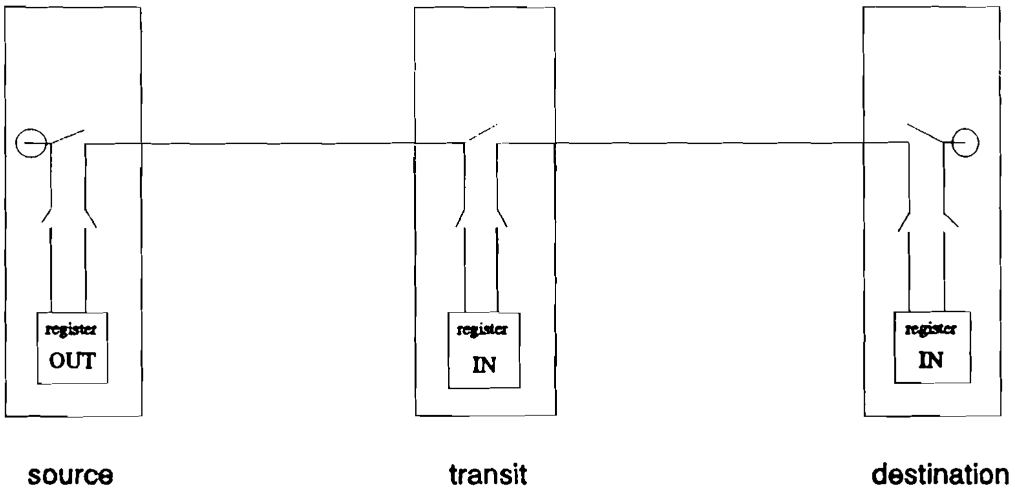



*figure 4. R2 end-to-end interregister signalling*

In the transit exchange the speech-path is immediately through-connected and the incoming register released. Then the outgoing register can exchange information directly with the incoming register of the next exchange. Such end-to-end signalling is advantageous as it reduces signalling equipment needed and minimizes the holding time of registers in transit exchanges.

Generally transmission conditions in a national network comply with the requirements specified for System R2 and as such allow end-to-end signalling over complete connections between local exchanges. If not, or where R2 is used in a satellite link, the overall multi-link connection is divided into end-to-end signalling sections.

A short post-dialling delay is achieved by using overlap operation. In overlapping interregister signalling the outgoing register starts call set-up as soon as it has received the minimum requisite information, also before the complete address information is received, i.e. the caller finishes dialling. En bloc register signalling requires complete reception of the address information.

System R2 compelled signalling operates as follows:

- on seizure of a link, the outgoing register automatically starts sending the first forward interregister signal,
- as soon as the incoming register recognizes this signal, it starts sending a backward interregister signal which has its own meaning and at the same time serves as an acknowledgement signal,
- as soon as the outgoing register recognizes the acknowledging signal, it stops sending the forward interregister signal,
- as soon as the incoming register recognizes the cessation of the forward interregister signal, it stops sending the backward interregister signal and
- as soon as the outgoing register recognizes the cessation of the acknowledging backward interregister signal it may, if necessary, start sending the appropriate next forward interregister signal.

Forward register signals are:

-address,

contains one element of information, a digit or a code, about the called or calling party's number or the end of pulsing,

-country-code and echo-suppressor indicators,

indicating the inclusion of the country-code in the address information, the need for an outgoing half-echo suppressor or an incoming half-echo suppressor,

-language or discriminating digit,

indicating what language should be used by the operator and the automatic working or any special characteristic of the call,

-test call indicator,

the call is originating from test equipment,

-nature of circuit indicators,

sent on request and using a second nature of some signals,

-end-of-pulsing,

no other address signal will follow or the transmission of code identifying the origin of the call is completed and

-calling party's category,

contains supplementary information concerning the nature of the call and its origin.

Backward register signals concern:

-requesting transmission of address signals,

-requesting information about the circuit,

-requesting information about the call or calling party,

-congestion signals,

-address-complete signals and

-the condition of the called subscriber's line.

Some other signals are reserved for use in the national network.

# 4. Signalling system no.7

Signalling system number 7 is an internationally standardized general purpose common channel signalling system that is optimized for operation in digital telecommunications networks in conjunction with stored program controlled exchanges. A general description and the specifications of the message transfer part is given in Blue Book VI.7 and the user parts are specified in fascicle VI.8 and VI.9 [CCITT,89]. The system's functional architecture consists of four levels and the relationship with the OSI model is indicated. Further the telephone user part is described by its signals. Recent and future developments concerning system no.7 end this chapter.

## 4.1 Features

Signalling system no.7* can meet present and future requirements of information transfer for inter-processor transactions within telecommunications networks for call control, remote control, and management and maintenance signalling. It provides a reliable means for transfer of information in correct sequence and without loss or duplication. It is suited for both circuit relates and non-circuit related signalling. The applications supported are the public switched telephone network, the integrated services digital network, interaction with network databases and service control points, the public land mobile network and operations administration and maintenance of networks.

The protocol was tailored to the telephone system originally, but its evolution has been influenced by the need to encompass more telecommunications services and to be aligned with the OSI reference model.

The development of system no.7 started in the mid seventies [Modarressi,90]. At that time, the layered approach for data transport was being developed and its value for signalling techniques was recognized. The advancement of technology diminished the importance of the inefficiency that was caused by the overhead of layered protocols and enabling flexibility in realizing complex functions. The system is influenced by bit-oriented protocols like higher-level data link control, HDLC. The first CCITT recommendations were published in 1980. Exapansions followed in 1984 and 1988. New recommendations are to be published this year.

The signalling system is optimized for operation over 64 kbit/s digital channels and it is carried in a dedicated time-slot within a PCM system. It is also suitable for operation over analogue channels and at lower speeds. The system can also be applied to point-to-point terrestial and

* The abbreviation SS7 denotes the signalling system implemented in the US which differs from the CCITT signalling system no.7. Therefore it will not be used.

satellite links. If required, it can be extended to point-to-multipoint operation.

Arrangements like error detection and correction, redundancy of signalling links and automatic diversion of signalling traffic in case of link failures provide reliable operation.

The combination of signalling points and their interconnecting signalling links form the CCITT signalling system no.7 signalling network. Signalling points are exchanges, operation, administration and maintenance centres, service control points and signalling transfer points, identified by a unique code known as a point code.

A sender/receiver system consists of three subsystems. The signalling control subsystem that receives information from the control system of the exchange, structures it to be sent in the appropriate formats, and queues the messages. It generates filler messages when there are no messages to be sent. Then the signalling termination subsystem assembles signalling units using the sequence numbers and check bits generated by the error control subsystem.

### Signalling network structure

Signalling system no.7 is specified for use in the associated and quasi associated mode, so the signalling route is pre-determined.

The *originating point* of a message is the signalling point at which the message is created. The *destination point* is the signalling point to which a message is destined. A *signal transfer point, STP,* is neither the source nor the destination of the message transferred.

The national and international network are considered to be structurally independent. A network consists of *signalling points* connected by *signalling links*. Signalling links encompass level 2 functions and provision is made for maintaining the correct message sequence. Signalling points provide signalling network functions at level 3 and user functions may be provided at level 4 if it also is an *originating point* or a *destination point*. A *signalling transfer point, STP,* only transfers messages from one signalling link to another. A particular signalling point may belong to both networks, but signalling points are allocated *signalling point codes* according to the rules of each network.

Destination point code, originating point code and the *signalling link selection, SLS,* are given in the message's label of figure 36. Routing is based on the DPC and SLS. Generally, more than one link can be used to route a message to a particular DPC. The selection of a particular link is made using the SLS field. This is called *load sharing* and its objective is to keep the load as evenly balanced as possible. Using the same SLS ensures messages taking the same path and thus arriving in sequence.

Routing is specified independent of the underlying signalling network structure, however to meet the availability requirements, the network must provide redundancy. Building components of the signalling network are *basic mesh networks*. These enable diversion in case of link failure.

Routing principles are:

- message routes pass through a minimum number of STPs,

- routing at each signalling point will not be affected by message routes used up to the concerning STP,
- if more then one route available, then the traffic should be load shared and
- related messages follow the same route to ensure the correct message sequence.

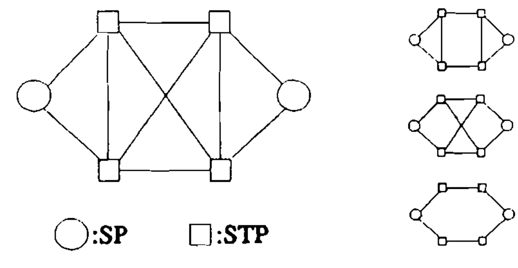



*figure 5. basic mesh network and its simlified versions*

## 4.2 Architecture

Signalling system no.7 is divided into four functional levels, not to be confused with the layers of the OSI reference model, as shown in figure 2. The user parts form the top level and the message transfer part, MTP, consists of the remaining three levels [CCITT,89],[Redmill,90].

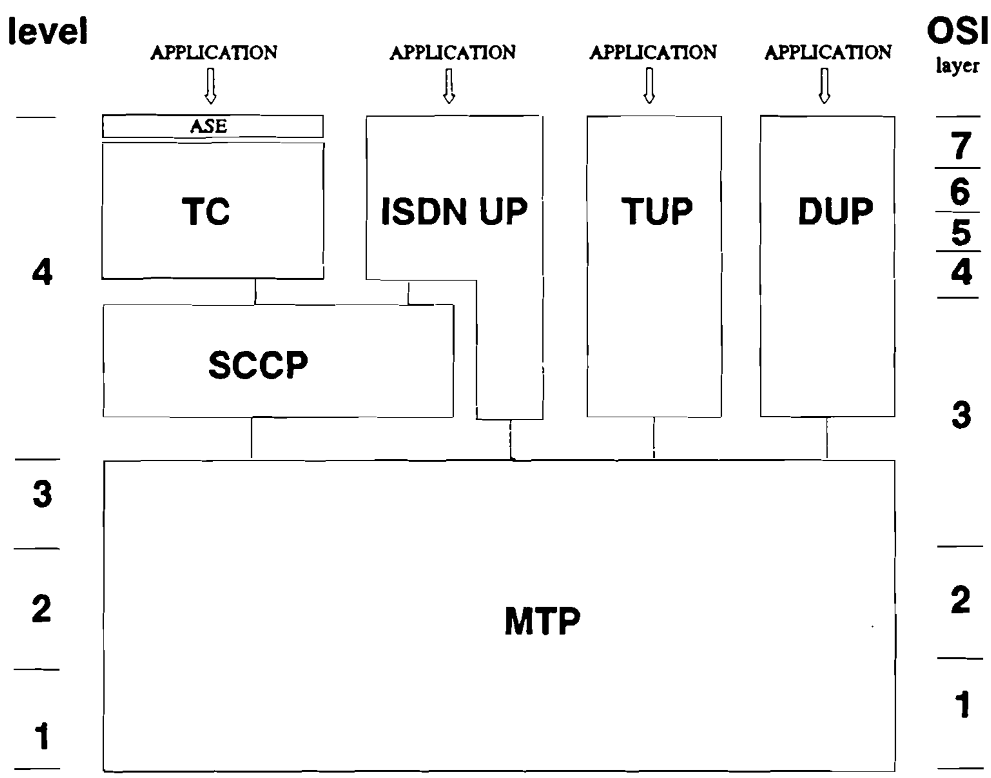



*figure 6. functional architecture of signalling system no.7*

### Message transfer part

Level 1, the *signalling data link functions*, of the MTP defines the physical, electrical and functional characteristics of a signalling link and the means to access it.

Level 2, the *signalling link functions*, defines the functions and procedures for and relating to the transfer of signalling messages over one individual signalling data link. These functions are error control, link initialisation, error-rate monitoring, flow control and delimitation of the signalling messages. *Error control* is achieved by appending a cyclic redundancy check code to the messages and requesting retransmission on errors according to the go-back-N technique [Tanenbaum,88]. For *link initialisation*, defined link status control messages are used to bring both ends of the signalling link into known states to each other. Error rate monitoring uses the 'leaky bucket' principle. The count of faulty messages is maintained and decremented at a fixed rate according to the number of good messages. If this count reaches a critical value, the link will be removed from service and this will be reported to level 3. *Flow control* prevents receiving nodes from being overloaded by witholding acknowledgement messages and informing the traffic source that the link concerning is in congestion. *Delimitation* marks the boundaries between messages. A signalling message delivered by the higher levels is transferred over the signalling link in variable length signal units. The signal unit comprises transfer control information in addition to the information content of the signalling message.

Level 3, the *signalling data link functions*, defines the transport functions and operations common to and independent of the operation of individual signalling links. These are *signalling message handling functions* directing the message to the proper signalling link or user part and *signalling network management functions* controlling the message routing and configuration of the signalling network facilities.

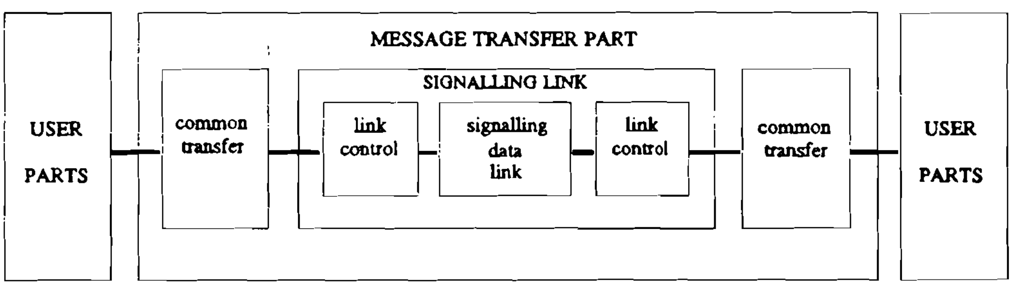



*figure 7. functions of the message transfer part*

### User parts

The user parts form level 4 of the signalling system. They define call-control signals to be used between communicating switching nodes.

The *signalling connection control part, SCCP*, comprises additional functions to the MTP to provide connectionless and connection-oriented network services to transfer circuit related and non-circuit related information. It enables the control of logical signalling connections in a Signalling system no.7 network and the transfer of signalling data units across the Signalling system no.7 network. The SCCP has a routing function in which the global title, for instance dialled digits, are translated into a signalling point code and a subsystem number. Another function is the management function to control the availability of the "subsystems" and broadcast this information to other nodes in the network which have a need to know the status of the subsystem. Users of the SCCP are the ISDN user part and the transaction capabilities, TC.

The *data user part, DUP*, defines the protocol to control interchange circuits used on data calls, and data call facility registration and cancellation.

The *telephone user part, TUP*, handles the necessary telephone signalling functions for use of Signalling system no.7 for international telephone call control signalling such as telephone signalling messages, their encoding and signalling procedures, and *cross-office performance*. The latter is to verify that the exchange does not exceed certain error rates. While the DUP still is not on an appreciable level of use, the TUP is in use in its international form between several countries. The *TUP+* is an enhanced version that supports limited ISDN-based services.

The *ISDN user part, ISDN-UP*, gives the ISDN network signalling messages, their encoding and signalling procedures, and cross-office performance. However, it deals with the basic services only. The ISDN-UP is suited for switched services and user facilities for voice and non-voice applications in the ISDN, dedicated telephone and circuit switched data networks and in analogue, and mixed analogue/digital networks. The combination with the SCCP provides end-to-end signalling.

The *transaction capabilities, TC*, consist of the transactions capabilities application part, TCAP, and the intermediate service part, ISP, that is for further study. TC provides services based on a connectionless network service which does not require ISP functions. Connection oriented TC services and the ISP layer 4-6 functions are for further study.

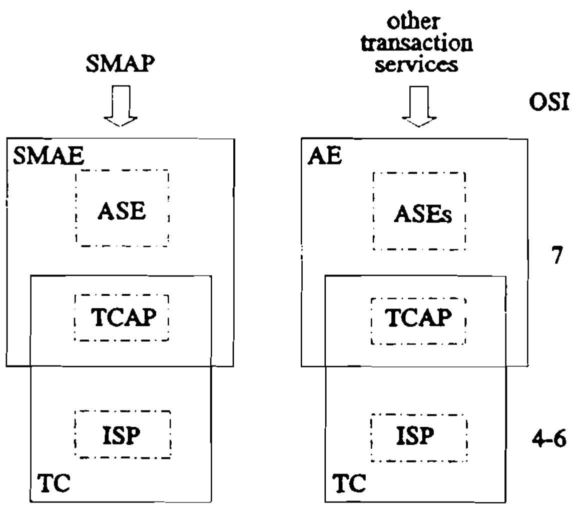


*figure 8. transaction capabilities and application entities*

The *application entity, AE*, is the combination of the TCAP and the *application service entity, ASE*. It represents the communication functions of the application process that are pertinent to inter-nodal communication using layer 7 application protocols. The mobile application part, MAP, is an example of an AE. Systems management is provided for by the *systems management application process, SMAP*. The aspect of the SMAP involving communication is the *systems management application entity, SMAE*. The *operation, maintenance and administration part, OMAP*, provides the application protocols to monitor, co-ordinate and control all the network resources that make communication based on signalling system no.7 possible.

An *application process* is a range of functions and features which support a specific network requirement, using one or more ASEs. It either co-ordinates specific aspects of network operation like ISDN call control and mobiles, or is an individual or supplementary service control function, e.g. closed user group.

## OSI reference model layering and signalling system no.7

The reference model of open systems interconnection is a well-defined structure for modelling the interconnection and exchange of information between users in a communications system. The functional model is divided into seven layers, not to be confused with the levels of Signalling System no.7 mentioned before. A layer receives service from an underlying layer and offers services to the layer above [Mitra,91].

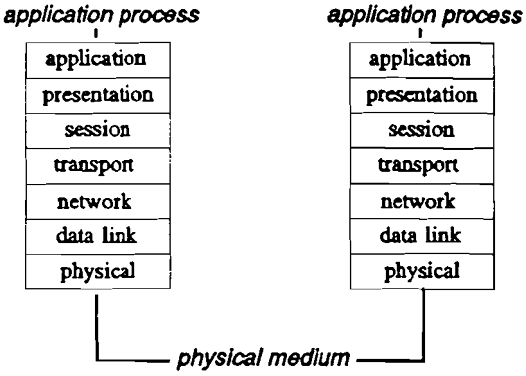



*figure 9. application processes communicating according to the OSI reference model*

1. The *physical layer* provides transparent transmission of a bitstream over a circuit. It furnishes the interface to the physical media and is responsible for relaying bits.

2. The *data link* layer overcomes the limitations inherent in the physical circuit and allows errors in transmission to be detected and recovered, thereby masking deficiences in transmission quality.

3. The *network layer* transfers data transparently by performing routing and relaying of data between end users. One or more of the subnetworks may interwork at the network layer to provide an end user to end user network service. A connectionless network provides for the transfer of data between end users, making no attempt to guarantee a relationship between two or more data messages from the same user.

4. The *transport layer* provides end user to end user transfer optimizing the use of resources according to the type and character of the communication, and relieves the user of any concern for the details of transfer. It operates end-to-end, enhancing the network layer when necessary to meet the quality of service objectives of the users.

5. The *session layer* co-ordinates the interaction within each association between communicating application processes. For example, full and half duplex dialogue.

6. The *presentation layer* transforms the syntax of the data to be transferred into a form that is recognizable by the communicating application processes.

7. The *application layer* specifies the nature of the communication required to satisfy the users' needs. It is the top layer of the OSI model and therefore the sole means for an application proces to access the OSI environment.

The MTP comprises OSI layers 1 and 2, and part of layer 3. The other part of layer 3 is provided for in the SCCP, TUP, DUP and ISDN-UP. Layer 4-7 define functions relating to end-to-end communication that are provided for in the user parts, see figure *.

## 4.3 Telephone user part

The telephone user part is described in the blue book fascicle VI.8 [CCITT,89]. The use of signalling system no.7 for telephone call control requires the application of the telephone user part functions as well as an appropriate set of the message transfer part functions. It is meant for international use and provides the same features as other CCITT signalling systems.

A link-by-link continuity check of the speech path is possible when the system is used with analogue telephone circuits or digital equipment without fault indications.

Signalling points are allocated unique codes.

Function of telephone messages and signals

The telephone signalling messages are grouped as follows:
-forward address messages,
    containing address information sent forward,
-forward set-up messages,
    further information for call set-up, subsequent to address messages,
-backward set-up request messages,
    requesting backwards further information for call set-up,
-successful backward set-up information messages,
    address complete and charging,
-unsuccessful backward set-up information messages,
    informing about the unsuccessful call set-up,
-call supervision messages,
-circuit supervision messages,
-circuit group supervision messages,
-circuit network management messages,
    used to control traffic flow to reduce congestion and
-automatic congestion control information messages,
    relating to the congestion status.

The service information contains the service indicator that identifies the user part the signalling message belongs to and the network indicator to distinguish national from international messages.

## 4.4 Prospects of signalling system no.7

Signalling system no.7 will be widespread deployed in public networks and their interconnection. DUP and TUP functions will be covered by the ISDN-UP protocol. After 1995, *broadband ISDN, BISDN*, capabilities will be emerged in the network [Modarressi,90]. A new user part will be designated, the *ISDN signalling control part, ISCP*. Its structure will be based as much as possible on the OSI reference model [Mitra,91], it will encompass both the network signalling protocol and the access signalling protocol for BISDN, and its functions and capabilities will be developed in the context of of a BISDN environment to provide for separation of bearer control from call control. It should support supplementary services and be applicable to intelligent network services. The first steps in this direction will follow in the forthcoming CCITT white book, to be published this year. For the further future, room to manoeuvre may be exhausted leading to a signalling system no.8, perhaps [Funke,90].

All this has consequences for the interworking of signalling system no.7 to R2. However the system that will be applied in the coming years will be based on the present recommendations that still contain the TUP. In addition, the following can be considered as an example for describing interworking systems formally and its use for deriving interworking.

# 5. Interworking of signalling systems

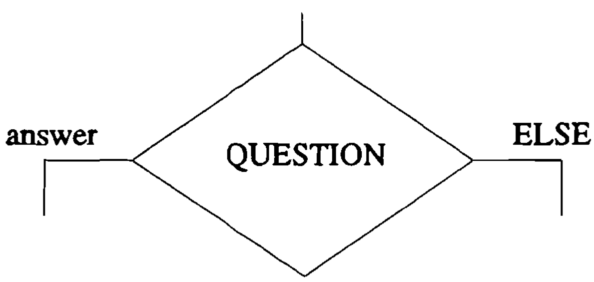
The signalling system R2 will gradually be replaced by signalling system no.7. This has for instance also been done in France [Craveur,87] and Sweden [Heidermark,84]. An interworking unit must be derived to enable operation of the new system with the old one. The CCITT recommendations on interworking are given in Blue Book VI.6 [CCITT,89]. These specifications are based on the CCITT specification and description language, SDL [Hogrefe,89].

## 5.1 Specification and description language, SDL

In SDL, processes are characterized by the following:

- a *process* is a finite state machine that can communicate with parallel existing processes and meanwhile also stores and manipulates data,
- a process is either in a transition or waiting for input,
- there is a queue for every process in which inputs are stored if the process is in a transition, the queue and process being independent of and parallel to each other and
- two inputs arriving at the same time are put randomly in the queue.

SDL has two syntax forms: *SDL graphical representation, SDL/GR* and *SDL phrase representation, SDL/PR*. Both are exact and semantically equivalent. The phrase representation can be compared with a computer programming language. Only the graphical representation will be dealt with since the CCITT recommendations are in SDL/GR.




DECISION question
(answer): ...
ELSE: ...
ENDDECISION;

SDL/GR

SDL/PR

*figure 10. a decision in SDL/GR and SDL/PR*

The description consists of actions and states [Koomen,91]. A *transition* is a sequence of actions that occurs when a process moves from one state into another. Actions are communication actions, decisions and tasks. Inputs and outputs, both from or to either the right or left hand side to denote the preceding or following process, are communication actions. In a state, the process is waiting for input. *Connectors* link parts of the SDL diagram. *Save* stores an input that is not yet expected, but this is not used in the interworking SDL* description. Inputs or outputs with a double line are *internal actions*. Examples of SDL diagrams can be found in apppendix C.

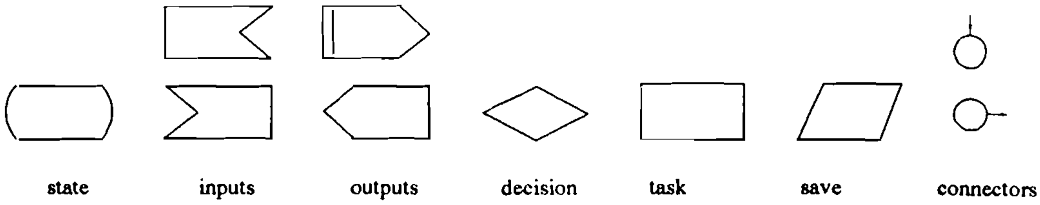



*figure 11. SDL symbols*

## 5.2 CCITT interworking

Interworking is defined to be the controlled transfer of signalling information across the interface** between different signalling systems where the significance of the transferred information is identical or where the significance is translated into a defined number and the performance of appropriate switching procedures in association with the transfer. It starts with the successful selection of the signalling system to interwork with and it continues until the connection is released. CCITT specifies interworking between its present standardized signalling systems no.4, 5, 6, 7(TUP), R1 and R2, also thirty combinations.

The interworking is divided into three functional blocks: the *incoming signalling system logic procedures****, the *interworking logic procedures* and the *outgoing signalling system logic procedures*, see figure 5. The incoming and outgoing procedures do not depend on the actual interworking combination. They cause actions such as the acknowledgement signal and the starting of time supervision that do not need interworking and generate interworking events in response to actions that do need interworking. These interworking events are:

- forward interworking telephone events, FITEs, that perform information transfer in the forward direction from an incoming signalling system to an outgoing signalling system,
- backward interworking telephone events, BITEs, which perform information transfer in the backward direction from an outgoing to an incoming signalling system and
- switching processing interface telephone events, SPITEs, describing the information flow at the functional interface between signalling and switching. These are internal actions.

* SDL will be used for SDL/GR

** In telecommunications, an interface is a point between (sub-)systems. In formal techniques, interfaces are systems that enable communication between (sub-)systems. In this report, the latter will be interworking units.

*** 'Procedure' is used in the same way as 'process' is in section 5.1.

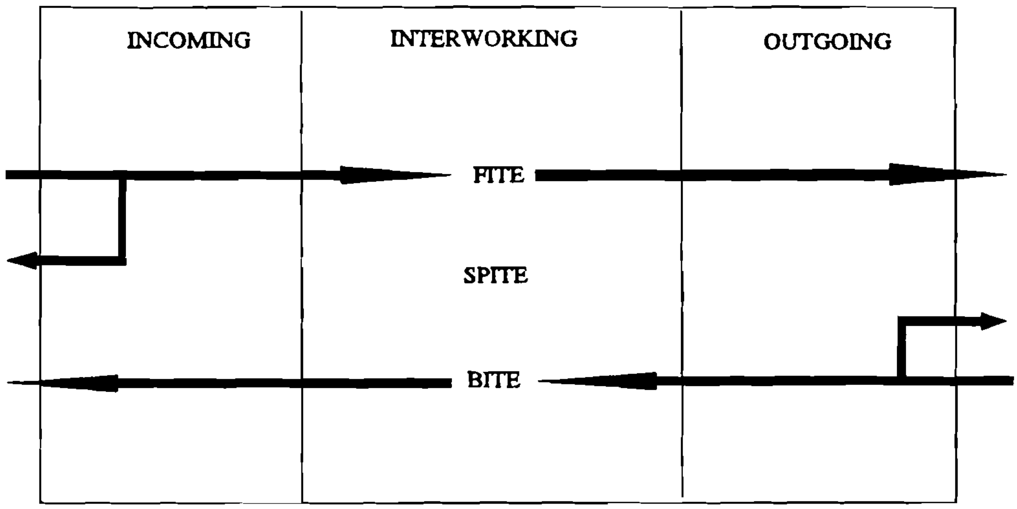



*figure 12. CCITT interworking of signalling system no.7 to R2*

The complete interworking between system no.7 and R2 is the combination of incoming no.7, interworking no.7 → R2 and outgoing R2 with incoming R2, interworking R2 → no.7 and outgoing R2, as shown in figure 6.

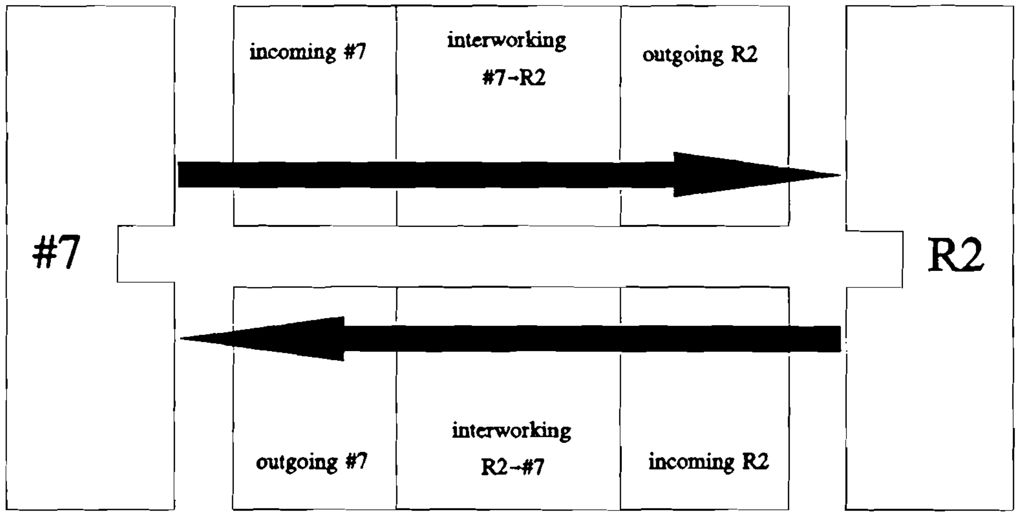



*figure 13. the interworking unit between signalling system no.7 and R2*

The FITEs, BITEs and SPITEs are listed in appendix D. Appendix C contains the SDL description incoming system no.7, interworking system no.7 to R2 and outgoing system R2.

## 5.3 Interworking by combining behaviours

This paragraph deals with interworking in a way that differs from that of the CCITT. It may reduce the complexity of the description, especially since only the interworking between signalling system no.7 and R2 is of interest, and not the aptitude of the description to other signalling sytems.

The behaviour of system no.7 and that of system R2 will be described. The interworking unit is the combination of the mirrored system no.7 behaviour and the mirrored R2 behaviour. The mirrored R2 behaviour can be seen as the response to R2 behaviour. The interworking from system no.7 to R2 and vice versa can be seperated, like in figure 13. The SDL diagrams for the incoming and outgoing system are then of use to describe the mirrored behaviour of system no7. and R2.

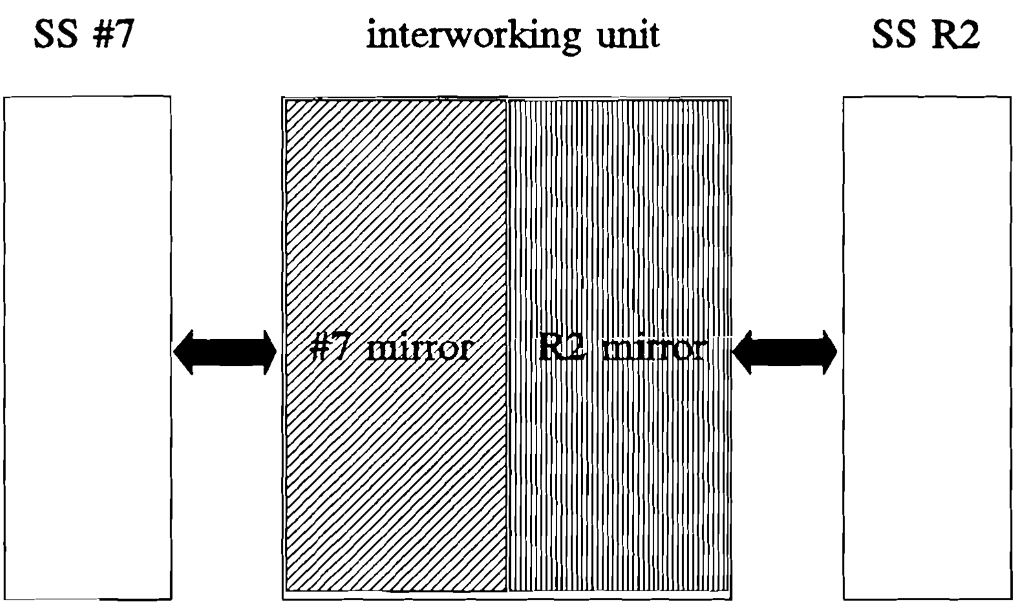



*figure 14. interworking by combining system no.7 and R2 behaviour*

The behaviours will be described in CCS and this formal description technique is treated in chapter 6. The behaviours will be combined according to the combination algorithm of section 6.4.

# 6. Calculus of communicating systems

The calculus of communicating systems, CCS, was developed by Robin Milner [Milner,80]. CCS will not be dealt with in detail here; the interested reader is referred to [Milner,89]. For the use of CCS in telecommunications, [Koomen,91] is recommended. This chapter is only about the theory and notational conventions needed for the derivation of the interworking unit.

The concurrent system is described by considering its behaviour that is externally observational. Of interest is the way the system interacts with the outside world, so its behaviour is defined by its entire capability of communication. Shortly, the behaviour of a system is exactly what is observable and to observe a system is exactly to communicate with it.

## 6.1 Definitions

*Agents* are the objects of behaviour. An agent's behaviour is defined in terms of actions they can perform. It is denoted by an algebraic expression, the *behaviour equation*, in which an agent equals one or a sequence of actions, separated by : and followed by the agent. A behaviour equation can be regarded as a state transition equation with the current state on the left-hand side and the state transitions followed by their resulting state on the right-hand side.

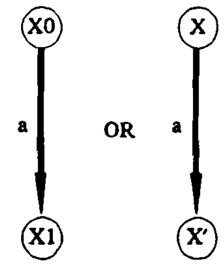


*figure 15. transition*

S is the agent for signalling system no.7 and R for signalling system R2. Their states are given by S1, S2, ... and R1, R2, ...

? is used for inputs and ! for outputs. Inputs from the preceding system are given by in1? and those from the following system by in2?. Each message can be seen as a separate input or output port.

For example, sending a clear forward to the

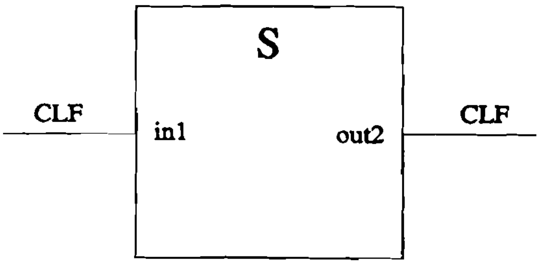


*figure 16. system S transferring message CLF*

next system equals performing out2!.CLF*. A system that would only pass a clear forward would be given by the following behaviour equation:

S = in1?.CLF: out2!.CLF: S

or in the *normal form*, which means that a new state is preceeded by only one action:

S0 = in1?.CLF: S1

S1 = out2!.CLF: S0

## Syntax**

*Action prefix*, X0 = $\alpha$: X1, states that behaviour X0 can perform action $\alpha$ and then show behaviour X1. Another notations are X = $\alpha$: X' and X $\xrightarrow{\alpha}$ X'. X' is an $\alpha$-derivative of X.

*Summation*: suppose X performing a leads to X' and Y' performing b to Y'. The summation of X and Y, X+Y, performing a then results in behaviour X'. Under action b, X+Y produces behaviour Y'.

The parallel *composition* X | Y can perform all the actions of X and Y. A complementary action, that is the input of X or Y and also the output of the other, is in this case an internal action of X | Y, replaced by $\tau$. (See *hiding*.) Suppose X = a: X' and Y = b: Y'. X | Y under a results in X' | Y and action b produces X | Y'. If a and b are complementary and occur synchronized, then the result is X' | Y'.

The *restriction (X | Y)\{a}* is the removal of all a and a-complementary actions from X | Y.

*Relabelling* X[b/a] is replacing label a by label b in the expression X. This is done to denote complementary actions.

Summation and composition are assumed to be associative and commutative [Milner,89].

## Hiding

A system may perform *internal actions*. Such an action is not externally observable, but a transition of the system due to this internal action, is. The action itself is not of importance, only the fact that an internal action has occurred is of interest. The action may be replaced by $\tau$, in this way simplifying the description, but also discarding

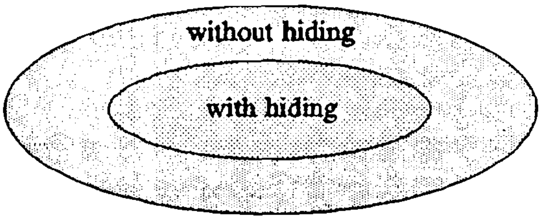



*figure 17. information contents of the description*

* At this stage, in1?.CLF is to be seen as a unique port. Communication is the reception of a synchronization pulse at this port. Actually, the message CLF is passed as a value at port in1 of S. This will be explaned under Synchronization and Value passing of section 6.2.

** CCS laws concerning this syntax are listed in section 6.3.

information about the system. The description of a system with use of hiding is that of a larger system than without hiding. $\tau$ is called internal or silent action.

For example, system X consists of two subsystems A and B that communicate with message b, given by:

| | | | | | |
|---|---|---|---|---|---|
| A0 | = | in?.a: A1 | B0 | = | in?.b: B1 |
| A1 | = | out!.b: A0 | B1 | = | out!.c: B0 |

| | | |
|---|---|---|
| X0 | = | in?.a: X1 |
| X1 | = | $\tau$: X2 |
| X2 | = | out!.c: X0 |

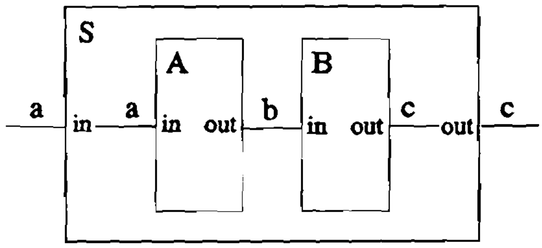



*figure 18. system X = (A | B)\{b}*

## Decision

Decision or *condition* is provided for by the IF-statement. Suppose system Y checks a variable x to be 0,1 or 2. The equation for state Y0 could be:

Y0 = if x=0 then Y1 + if x=1 then Y2 + if x=2 then Y3

or also possible:

Y0 = if x=0 then Y1 + if not x=0 then Y4

## IF reduction

The evaluation of a condition can be seen as an internal action and be replaced by $\tau$ and thereby hiding information, like in the system above:

Y0 = $\tau$: Y1 + $\tau$: Y2 + $\tau$: Y3

## $\tau$-laws

The following $\tau$-laws will be used:

a: $\tau$: X0 = a: X0

X0 + $\tau$: X0 = $\tau$: X0

## $\tau$-reduction

Applying $\tau$-reduction is the removal of all $\tau$'s. Again, this causes loss of information. The system Y above would yield:

$Y0 \approx^{*} Y1 + Y2 + Y3$

Y0 is the summation of the behaviour of Y1, Y2 and Y3, denoted by $Y_{0,1,2,3}$

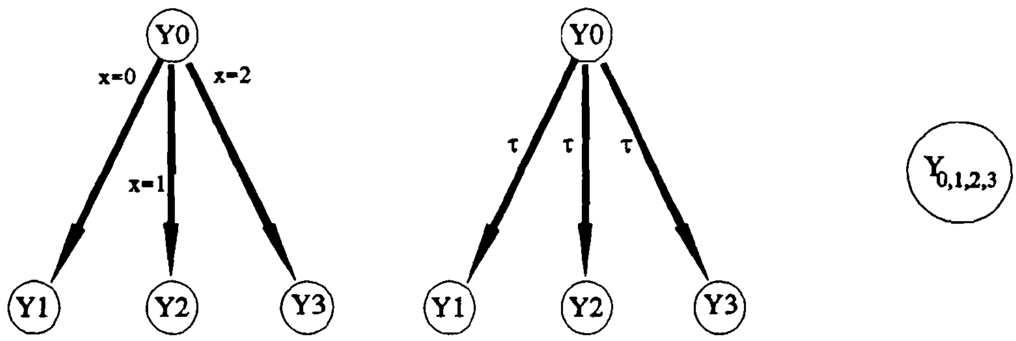



*figure 19. decision,...* *...followed by IF reduction...* *...and τ reduction*

#### Fairness

A system with $X0 = \tau\text{: } X0 + X1$ can make $\tau$-actions indefinitly and in this way never leave state X0. The *fairness rule* states that X will move to X1 after a finite amount of $\tau$-moves. Also, $X0 = \tau\text{: } X0 + X1$ must be replaced by $X0 = \tau\text{: } X1$. This rule may be used only in those cases in which there is additional information on the application that guarantees the rule is applicable.

## 6.2 Systems in CCS

This paragraph is about the means and techniques to describe systems in CCS. These are expansion in order to combine systems and value passing and queues to realize concurrency.

#### Expansion

Composing systems is to determine the parallel composition of their behaviours, restricted with respect to their connecting ports since only the externally behaviour is of interest.

In CCS, concurrency is modeled by *interleaving of actions*. This means that two actions can only map onto the same moment if they are complementary, also in case of communication. This yields

A1 | B0 = Aout!.b: A0 | B0 + Bin?.b: A1 | B1 + $\tau$: A0 | B1

for system X of figure *, because Aout!.b and Bin?.b are complementary actions. It takes all

* $\approx$ denotes observation equivalence, see section 6.3.

possible interleavings. After restriction, only

(A1 | B0)\{Aout!.b} = $\tau$: A0 | B1

remains. Calculating (A | B)\{Aout!.b} is known as expansion.

Let L be the set of complementary actions between systems $P_i$. The parallel composition of them, P = ($P_1$ | $P_2$ | ... | $P_n$)\L can be calculated as follows:

P = $\Sigma$ a: ($P_1$ | ... | $P_i$' | ... | $P_n$)\L

+ $\Sigma$ $\tau$: ($P_1$ | ... | $P_i$' | ... | $P_j$' | ... | $P_n$)\L

with:

$P_i$ = a: $P_i$' and a or its complement is not a member of L

$P_i$ = l: $P_i$' and $P_j$ = l': $P_j$'with l and its complement l' in L

This *expansion law* can be applied after relabeling connecting ports.

The algorithm for expansion contains four steps:

a. X0 = A0B0,
b. list every possible action of A and B that is permitted under the restriction, meaning not involving communication between A and B, followed by the resulting state,
c. add actions that do involve communication, denoted by $\tau$ and followed by the resultinf state and
d. repeat b. and c. for every resulting state that has not been evaluated yet.

For example, system X = (A | B)\{Aout!.b}:

a. X0 = A0B0 initial state,
b. A0B0 = Ain?.a: A1B0 this is the only action possible at this stage for A, and B is waiting until A is able to communicate, too,
c. A1B0 = $\tau$: A0B1 communication takes place and
b. A0B1 = Bout!.c: A0B0 the only state left to evaluate.

or:

X0 = (A0 | B0)\{Aout!.b}
= ((Ain?.a: Aout!.b: A0) | (Bin?.b: Bout!.c: B0))\{Aout!.b}
= Ain?.a: ((Aout!.b: A0) | (Bin?.b: Bout!.c: B0))\{Aout!.b}
= Ain?.a: $\tau$: (A0 | (Bout!.c: B0))\{Aout!.b}
= Ain?.a: $\tau$: Bout!.c: (A0 | B0)\{Aout!.b}
= Ain?.a: $\tau$: Bout!.c: X0

### Synchronization

CCS knows a version based on *pure synchronization* and and one with *value passing*. So far, the first is used.

In pure synchronization, communication only is the sending and receiving of a *synchronization* *pulse*, like sending an envelope without a letter. This seems not to be enough to describe systems whose future behaviour depends on the information received*. This dependence is expressed in decisions that have conditions based on input variables. These input variables occur as parameters in an input prefix. This is value passing communication. At first sight, value passing seems necessary and the most convenient way to describe systems in practice. However, values or messages on a single port can be translated into multiple ports that are purely synchronized. This means that the definition of the theory of CCS may be restricted to pure synchronization without value variables and value expressions. It can be applied to a value passing calculus by translating that calculus into the basic calculus. Value passing is useful, because a description using it, is smaller than one with purely synchronized ports.

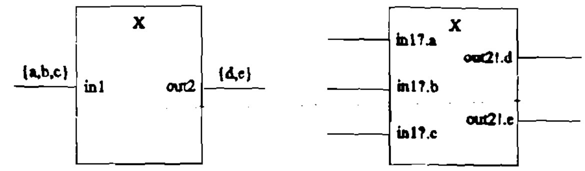



*figure 20. translating message values into multiple ports*

### Value passing

Suppose av is an input action in which the value x is transferred for variable v: $A \xrightarrow{av} A'(x/v)$. In this way, different states for different values of v are introduced. If a' is not an action in B, then $A \xrightarrow{av} A'$ will cause $A \mid B \xrightarrow{av} A'(x/v) \mid B$. Also, if A does not have an action complementary, to b, $B \xrightarrow{b} B$ implies $A \mid B \xrightarrow{b} A \mid B'$. In the case $B \xrightarrow{a'v} B'$, communication with value passing takes place: $A \mid B \xrightarrow{\tau} A'(x/v) \mid B'$.

In the description of signalling systems, in1 denotes the input from the preceding exchange and out1 the output that exchange. In2 and out2 are used for communication with the following exchange. Receiving a clear forward, CLF, is performing the action in1?.CLF. This can be regarded as a synchronization pulse on the port in1?.CLF of the system in the purely synchronized calculus.

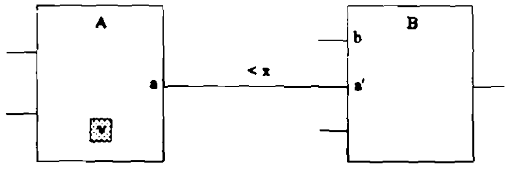



*figure 21. value passing*

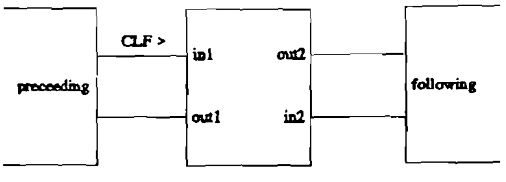



*figure 22. value passing between signalling systems*

### Queueing

In the foregoing, communication between two agents is assumed to occur simultaneously; the message is received at the same moment it is sent. In practice, the channel has a memory property and the sender proceeds, not awaiting the reception of the message at the other end.

* This is the fact for signalling systems in which decisions are made upon information, previously received.

This means that the sending and receiving are separated actions. Each input has a message queue. The delay is modelled to occur only in the input queue and not in the channel anymore. In the expansion of two parallel systems, first a $\tau$-action occurs for the sender to put the message in the queue, and then another $\tau$-action for the receiver to remove the message from the queue.

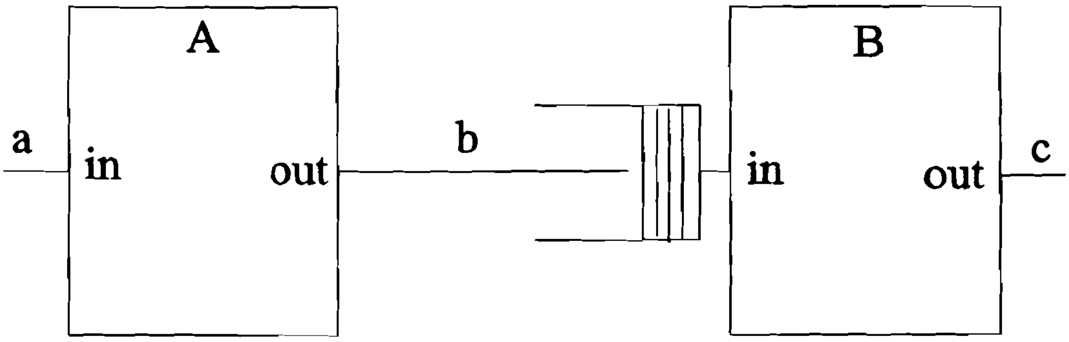



*figure 23. system X = (A | B)/{b} with buffering*

For example, the expansion of system X = (A | B)\{Aout!.b} now proceeds as follows:

| | | |
|---|---|---|
| a. | A0()B0() | initial state, |
| b. | A0()B0() = Ain?.a: A1()B0() | this is the only action possible at this state for A and B is waiting for A to be ready to communicate, too, |
| c1. | A1()B0() = $\tau$: A0()B0(b) | A puts the message b in the queue of B, |
| c2. | A0()B0(b) = $\tau$: A0()B1() | B takes the message from the queue, |
| b. | A0()B1() = Bout!.c: A0()B0() | the only state left to evaluate. |

### Timers

In practice, systems tend to show unwanted behaviour besides wanted behaviour. To be able to cope with this behaviour not wanted, strengthens the theory. Lossy channels are an important source of errors in communication. Redundancy may be added to the messages for the receiver to determine whether the information is correct or not. An example for this is the cyclic redundancy chech, CRC, in which a code, calculated from the bits of the message, is added to the message and checked against the result of the same calculation, but then performed by the receiver. In case of errors, retransmission can be asked by a negative acknowledgement or the absence of an acknowledgement informs the receiver that an error has occurred. The latter brings on the need to introduce time-outs.

The following is an example of how to do this. Figure 24 shows systems A and B with acknowledgement and time-out:

| | | | |
|---|---|---|---|
| A0 | = | in1?.a: A1 | |
| A1 | = | out2!.b: A2 | send message |
| | + | in2?.ACK: A0 | though timed-out, ACK may still arrive |
| A2 | = | in2?.ACK: A0 | correct received, start again at A0 |
| | + | $\tau$: A1 | time-out, so send again |

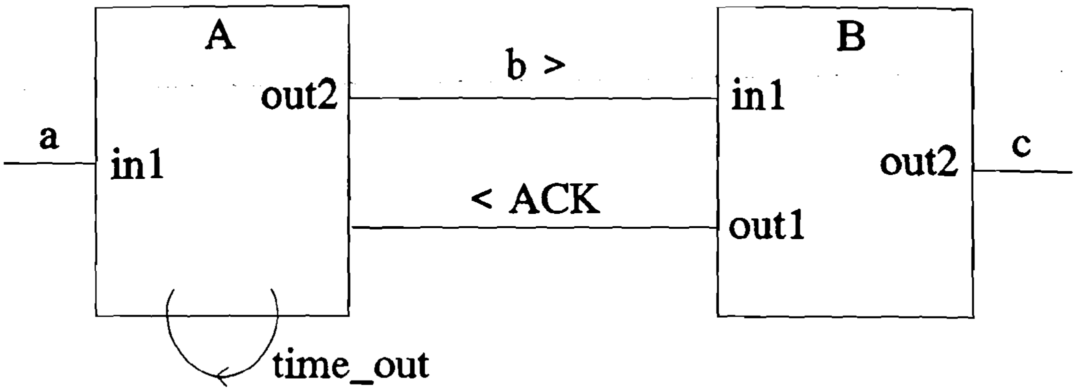



*figure 24. system X = (A | B)/{b,ACK} using acknowledgement and time-out*

| | | | |
|---|---|---|---|
| B0 | = | in1?.b: B1 | B receives b correctly and |
| B1 | = | out1!.ACK: B2 | confirms this with ACK |
| B2 | = | out2!.c: B0 | |

Expansion of X = (A | B)\{Aout!.b, Ain?.ACK} yields:

| | | |
|---|---|---|
| X0 | = A0B0 | initial state, |
| A0B0 | = Ain1?.a: A1B0 | this is the only action possible at this state for A and B is waiting until A is able to communicate, |
| A1B0 | = $\tau$: A2B1 | communication takes place A awaits ACK and B b |
| A2B1 | = $\tau$: A0B2 | ACK sent and received |
| | + $\tau$: A1B1 | internally timed-out |
| A1B1 | = $\tau$: A0B2 | both would like to send |
| A0B2 | = Ain1?.a: A1B2 | |
| | + Bout2!.c: A0B0 | |
| A1B2 | = Bout2!.c: A1B0 | A cannot send b and will not receive ACK |

This system is free of deadlock, which means there is not an equation of the form AB = nil. If deadlock should occur while expanding, the behaviour of A and B should be altered.

The time for a timer to expire must exceed the time for data to arrive or actions to occur. Assigning values to ACK referring to the message to be acknowledged prevents a wrong message from being acknowledged.

## 6.3 Formal verification

*Verification* is checking the consistency between a specification and a potential implementation. Formally: define a equivalence relation over behaviours and show that the specification S and the implementation I are in the same equivalence class. Agents are equivalent if an external observer cannot distinguish them.

Two agents are in *bisimulation* if one can simulate the behaviour of the other and vice versa. This is *bisimulation equivalence.*

Differences in equivalence

Replacing a description of a system by an equivalent one may change the information content of the description and its justification depends on the situation. It is important how equivalence is defined.

Consider systems A and B:

A = a: A + $\tau$: b: A

B = a: B + b: B

*figure 25. system A ≠ system B*

Assuming $\tau$: X = X would yield A = B. If a ≠ b then B may perform either a or b, whatever state it can reach (this is fact only B). But A may reach A' after $\tau$ and in that state, action b is possible and a is not. The occurence of $\tau$ is autonomous in the sense that it does not need external participation and therefore A is uncontrollable, or non deterministic, while B is perfectly controllable and deterministic.

The example shows that the fact that two agents can perform the same sequences of external actions does not prove them behaviourally equal.

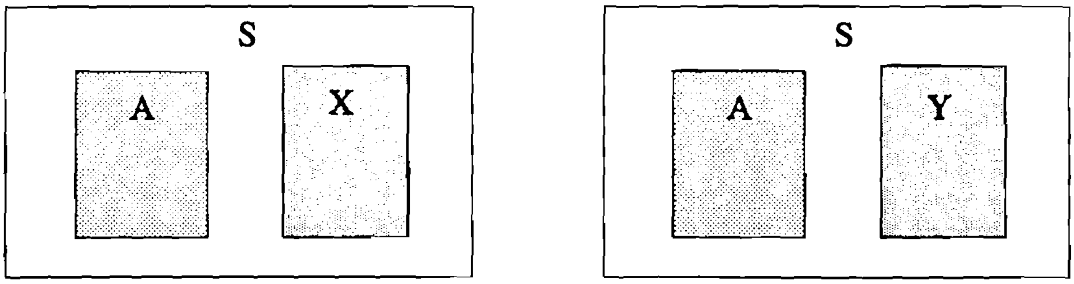



*figure 26. replacing X by Y requires X = Y*

If a part of a system must be replaced, the equivalence must be stronger than only being able to perform the same action sequences. For example, system X must be equal to system Y in figure 26. We will proceed with definitions of different types og equivalence. The quick reader can continue with the paragraph on CCS-laws.

### Strong equivalence

A *relation S* over agents is a *strong bisimulation* if (P,Q) is a member of *S* implies, for every action $\alpha$, that whenever P' is an $\alpha$-derivative of P, for some Q', Q' is an $\alpha$-derivative of Q, and (P',Q') also is a member of *S*.

Systems P and Q are strongly equivalent ($P \sim Q$) if and only if for every action $\alpha$, every $\alpha$-derivative of P is equivalent to some $\alpha$-derivate of Q and conversely.

Every $\alpha$-action of one agent must be matched by an $\alpha$-action of the other, even for $\tau$-actions.

### Observation equivalence

Here the requirements concerning $\alpha$-actions are relaxed. Every $\tau$-action is to be matched by zero or more $\tau$-actions.

X' is an *$\alpha$-descendant* of X if the action $\alpha$-is preceeded and followed by zero or more $\tau$-actions. For example X = $\tau$: $\tau$: $\alpha$: $\tau$: X'. Next, with all $\tau$-actions removed, it yields X = $\alpha$: X' or even X = X' if $\alpha = \tau$'.

A *relation S* over agents is a *weak bisimulation* if (P,Q) is a member of *S* implies, for every action $\alpha$, that whenever P' is an $\alpha$-derivative of P, for some Q', Q' is an $\alpha$-descendant of Q with all $\tau$-actions removed, and (P',Q') also is a member of *S*.

Systems P and Q are observation equivalent ($P \approx Q$) if and only if for every action $\alpha$ every $\alpha$-derivative of P is observation equivalent to some $\alpha$ descendant of Q, and similarly with P and Q interchanged.

Strong equivalence implies observation equivalence, observation equivalence or *bisimularity* only implies strong equivalence in case both the agents do not contain $\tau$-actions.

### Observation congruency

**P and Q are equal** or *observation congruent* ($P = Q$) if for every $\alpha$, whenever P' is an $\alpha$-derivative of P then for some Q' Q' is an $\alpha$-descendant of Q and P'and Q' are observationally equivalent and conversely. Note that Q' is an $\alpha$-descendant without the removal of $\tau$-actions. Each action of P or Q must be matched by at least one action of the other. This only applies to the first action of P and Q since P' and Q' only need to be observation equivalent and not observation congruent.

For example a $\approx$ $\tau$: a but a $\neq$ $\tau$: a. Also the systems of figure 25 are observationally equivalent, but not observationally congruent.

The following notation of equivalence types is used: $\equiv$ syntactically identical, $\sim$ strongly equivalent, $=$ equal or observationally congruent, and $\approx$ observationally equivalent.

## CCS laws

summation

$A + B = B + A$

$A + (B + C) = (A + B) + C$

$A + nil = A$

$A + A = A$

composition

$A \mid B = B \mid A$

$A \mid (B \mid C) = (A \mid B) \mid C$

$A \mid nil = A$

restriction

$nil \backslash a = nil$

$(A + B)\backslash a = A\backslash a + B\backslash a$

$(\alpha{:}\ A)\backslash a = nil$ $\{a=\alpha\}$

$= \alpha{:}\ (A\backslash a)$ $\{a \neq \alpha\}$

$A\backslash a = A$ {a is not one of the actions that constitute behaviour B}

$A\backslash a\backslash b = A\backslash b\backslash a$

$(A \mid B)\backslash a = (A\backslash a) \mid (B\backslash a)$ {a or a' only in A or only in B}

relabelling

$nil[b/a] = nil$

$(A + B)[b/a] = A[b/a] + B[b/a]$

$(A \mid B)[b/a] = (A[b/a]) \mid (B[b/a])$

$\tau$-laws

1. $\alpha{:}\ \tau{:}\ A = \alpha{:}\ A$

2. $A + \tau{:}\ A = \tau{:}\ A$

3. $\alpha{:}(A + B) + \alpha{:}\ B = \alpha{:}\ (A + \tau{:}\ B)$

4. $A + \tau{:}\ (A + B) = \tau{:}\ (A + B)$

5. $\tau{:}\ A \approx A$

fairness*

* See Fairness in § 6.1.

## 6.4 Combination algorithm

*Interface systems* are characterized by the following: given a set of systems, find a new system that interfaces with the other systems in a predefined way. Three types of interface synthesis are possible: synthesis by mirroring, synthesis by completing the specification and synthesis by interface derivation. The last one uses the combination algorithm.

### Mirroring

If systems A and S must communicate with each other, we want this to happen without the possibility of deadlock. Let C(A,S) $\equiv$ (A | S)\L and L are the complementary actions of A and S. Free of deadlock means C(A,S) = $\tau$: C(A,S).


*figure 27. given A, find S with C(A,S)=τ:C(A,S)*

M(A) is the *mirror image* of A: M(A) $\equiv$ A[$\alpha$'/$\alpha$] and is equal to behaviour A in which all actions are replaced by their complementary actions.

MT(A) is the *$\tau$-reduced mirror image* of A: MT(A) $\equiv$ A[$\alpha$'/$\alpha$, /$\tau$], the mirror image of A without $\tau$-actions.

MTU(A) is the $\tau$-reduced mirror image with *unique action occurrences* of A. An action that has different resulting states causes non-deterministic behaviour; this is not wanted. If an action occurs more than one time in an expression for a state, leading to different states, the occurrence is uniquely identified by a number. For instance, CLF_1 and CLF_2.

Synthesis by mirroring is illustrated in [Koomen,91], following the next steps:
1. calculate the behaviour of the system with which S should interface,
2. calculate the mirror image MTU and
3. refine the description, based on knowledge of the system.

### Mirror observation equivalence

Let CMTU(A) $\equiv$ C(A,MTU(A)). It can be shown that CMTU(A) = $\tau$: ($\tau$: CMTU(A) + $\tau$: nil) [Koomen,91]. Is there not sub-behaviour of the form $\tau$: nil then the system is deadlock-free, as stated above. Generalizing, if a system can perform $\tau$ in a certain state, then the mirror image is not allowed to do so. However, the roles of A and X at each step of the expansion CMTU(A) can be reserved; at each step one agent must be the MTU of the other. Thus C(A,X) = $\tau$: C(A,X) also if the roles of X and A are interchanged at each expansion step. In this case, A and X are *mirror observation equivalent* or A $\approx^m$ X.
A *relation S* over agents is a *weak mirror bisimulation* if (P,Q) is a member of *S* implies, for

every action $\alpha$, that

- whenever P' is an $\alpha$-derivative of P, for some Q', Q' is an $\alpha$' descendant of Q with all $\tau$-actions removed, and (P',Q') also is a member of *S*,
- and similarily with P and Q interchanged,
- whenever P' is a $\tau$-derivative of P, then there is not some Q', Q' is a $\tau$-descendant of Q and (P',Q) also is a member of *S*,
- and similarily with P and Q interchanged.

This type of equivalence is weaker than equality. P and Q only need to be able to follow one another, but they may not be able to perform a $\tau$-action at the same time. $\tau$-actions may only occur if there is not a $\tau$-action on the other side.

### Completing the specification

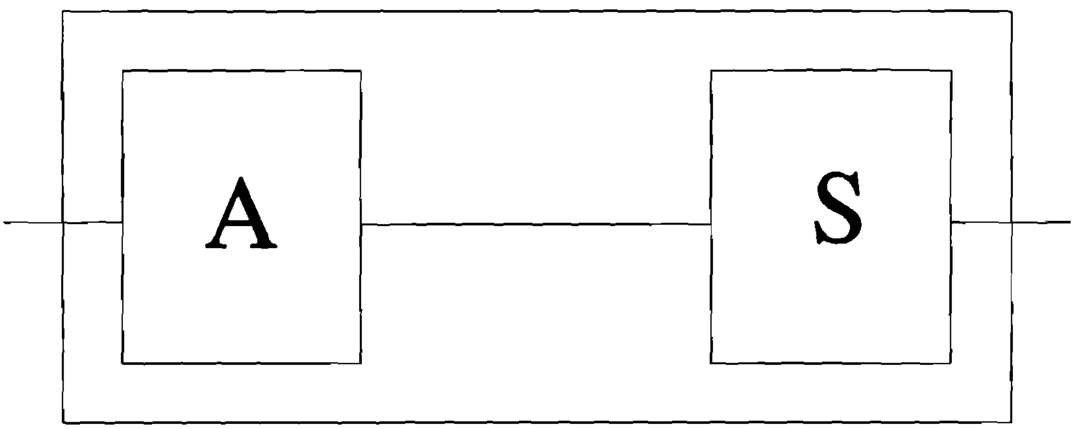



*figure 28. find S satisfying C(A,S)=SPEC*

If we have a specification SPEC, and an agent A which implements SPEC partially, the missing part S with the property C(A,S)=SPEC can be calculated. Without further proof is stated S'=C(A | SPEC'). 'denotes the complement or mirror of a system. Again, this is clarified in [Koomen,91]. The behaviour of A and C, the specification SPEC, is known. Expand (A | M(C))/L and take the mirror image of the equations resulting from the expansion. Replace the actions in this mirror image that are a member of L by their complements and rename the agents in this mirror image.

### Interface derivation: the combination algorithm

While synthesis by mirroring and completing the specification are slightly touched, the combination algorithm will be described fully. In this case, modules need to be linked by another module, thus enabling communication between those modules. The algorithm is apt for a number of systems to be linked together, but here only linking two is of interest. The requirement is C(A,B,S) = $\tau$: C(A,B,S).

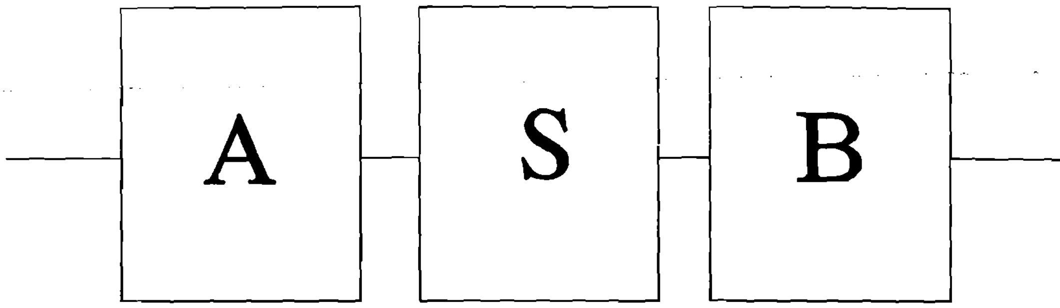



*figure 29. S to be derived between A and B*

This is satisfied if $C(S \mid A) \approx^{m} B$ and $C(S \mid B) \approx^{m} A$. MTU(A) and MTU(B) are the *partial specifications* of S.

In certain cases, messages received by S must be stored. This queueing is represented by adding a list of messages to the agent identifier of S.

Input queues can be taken into account according to queueing in section 6.2.

For now, assume the system having internal queues nor input queues and that the internal delay is negligible compared with the communication delay.

*Causal relations* need to be defined to maintain the correct sequence of occerences of actions that are related. A causal relation between actions $\alpha$ and $\beta$, $\alpha \rightarrow \beta$, means that $\alpha$ must have occurred before $\beta$ can. This relation is *left causal* if $\alpha$ is on the left-hand side. For instance, in signalling systems, B can transmit address information only after this has been received from side A.

The algorithm uses the sets Expand with the states of S to be evaluated, initially only S0 and Ready, the states that have already been evaluated. The initial state of S, S0 = MTU(A)0MTU(B)0. These sets are manipluated according to the following steps*:

1. write the behaviours of A and B in the normal form**,
2. calculate MTU(A) and MTU(B),
3. initialize Expand with S0,
4. define the causal relations,
5. select a state from Expand to be the current state,
6. remove this state from Expand,
7. add this state to Ready,
8. evaluate all the actions of the current state and their resulting states, reckoning with the causal relation previously defined,
9. add the states resulting from 8 to Expand,
10. if Expand is empty then go to 5 and

---

* Step 8 will be stated more formally later on.

** Hiding of internal actions, if-reduction and $\tau$-reduction may be applied. This abstraction will simplify the expressions but also reduce the information contents of the description.

11. reduce the obtained behaviour of S.

Example: system A and B would like to set up communication but they do not use the same protocol. Their behaviour is given by:

| | | | |
|---|---|---|---|
| 1. | A0 | = out2!.AM: A1 | send address message with address value |
| | A1 | = in2?.AF: A0 | address faulty received |
| | | + in2?.SD: A2 | right received, send data |
| | A2 | = out2!.CLF: A0 | end, clear forward |
| | B0 | = in1?.CR: B1 | receive connection request |
| | B1 | = out1!.CG: B2 | connection granted |
| | | + out1!.GNC: B0 | connection not granted |
| | B2 | = out1!.AR: B3 | address requested |
| | B3 | = in1?.AI: B4(AI) | address information |
| | B4(AI) | = $\tau$: B4 | value of AI is passed on |
| | B4 | = out1!.PS:B5 | proceed to send |
| | B5 | = in1?.EC: B0 | until end of connection |

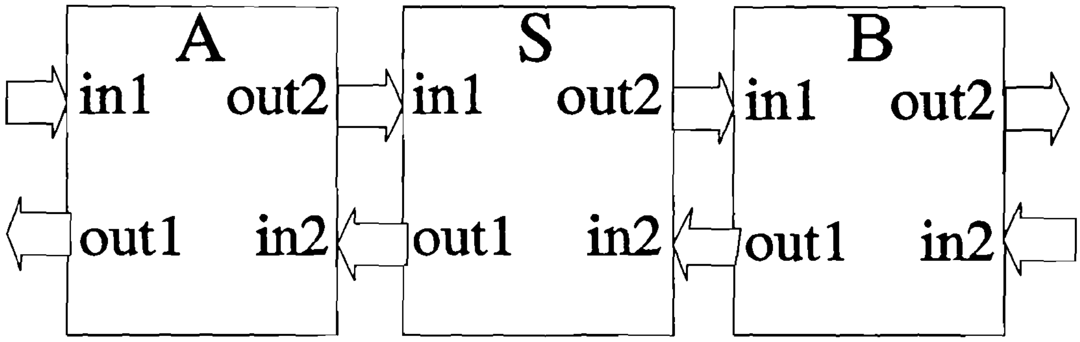



*figure 30. linking two systems with different protocols that want to set up communication*

$\tau$-reduction yields:

B3 = in1?.AI: B4

| | | |
|---|---|---|
| 2. | A0 | = in1?.AM: A1 |
| | A1 | = out1!.AF: A0 |
| | | + out1!.SD: A2 |
| | A2 | = in1?.CLF: A0 |
| | B0 | = out2!.CR: B1 |
| | B1 | = in2?.CG: B2 |
| | | + in2?.GNC: B0 |
| | B2 | = in2?.AR: B3 |
| | B3 | = out2!.AI: B4 |

B4 = in2?.PS:B5
B5 = out2!.EC: B0

3. S0 = A0B0

4. causal relations
AM → CR
AM → AI
CLF → EC

AF ← GNC
SD ← PC

5-10. A0B0 = in1?.AM: A1B0
A1B0 = out2!.CR: A1B1
+ out1!.AF: A0B0
A1B1 = in2?.CG: A1B2
+ in2?.GNC: A1B0
A1B2 = in2?.AR: A1B3
A1B3 = out2!.AI: A1B4
A1B4 = in2?.PS: A1B5
A1B5 = out2!.SD: A2B5
A2B5 = in1?.CLF: A0B5
A0B5 = out2!.EC: A0B0

11. S0 = in1?.AM: S1
S1 = out2!.CR: S2
+ out1!.AF: S0
S2 = in2?.CG: S3
+ in2?.GNC: S1
S3 = in2?.AR: S4
S4 = out2!.AI: S5
S5 = in2?.PS: S6
S6 = out2!.SD: S7
S7 = in1?.CLF: S8
S8 = out2!.EC: S0

The result can be verified by calculating C(S0,A0) and this must be mirror observationally equivalent to B0. The same goes for A0 and B0 interchanged.

The above is a simple example of interworking of protocols. Coping with possible arising faults and offering all kinds of services, makes signalling protocols rather complex. This may cause the need for timers, input buffering, value passing etcetera. Also the amount of abstraction, like hiding internal actions, influences the complexity of the system description.

### Synthesis rules

This paragraph is of less importance for the next chapters, but it completes the combination algorithm.

Introduce two *causality predicates*, L-causal($\alpha$) and R-causal($\beta$), that are true if $\alpha$ occurs on the left-hand side respectively $\beta$ on the right-hand side.

S(l) is the current, S'(l) the next state of system S with list l containing the messages received by S and for which the causality predicate is true. $S_i$ is a partial specification in this paragraph.

In(S) is the set of input actions possible in the current state of S. Storing them in l depends on the causal relations. Out(S) is the set of output actions possible in the current state of S. The actual occurrence of these outputs depends on the presence of their corresponding messages in l, and again this depends on the causal relations. List l has a head h(l) and a tail t(l).

The next rules are used in the combination algorithm.

| | |
|---|---|
| synth-1 | Suppose in(m) $\equiv \alpha$ in In(S), L-causal($\alpha$) and $S_i \xrightarrow{\alpha} S'_i$ then S(l) $\xrightarrow{\alpha}$ S'(l,m) |
| synth-2 | Suppose in(m) $\equiv \alpha$ in In(S), not L-causal($\alpha$) and $S_i \xrightarrow{\alpha} S'_i$ then S(l) $\xrightarrow{\alpha}$ S'(l) |
| synth-3 | Suppose out(h(l)) $\equiv \beta$ in Out(S), R-causal($\beta$) and $S_i \xrightarrow{\beta} S'_i$ then S(l) $\xrightarrow{\beta}$ S'(t(l)) |
| synth-4 | Suppose out(h(l)) $\equiv \beta$ in Out(S), not R-causal($\beta$) and $S_i \xrightarrow{\beta} S'_i$ then S(l) $\xrightarrow{\beta}$ S'(t(l)) |

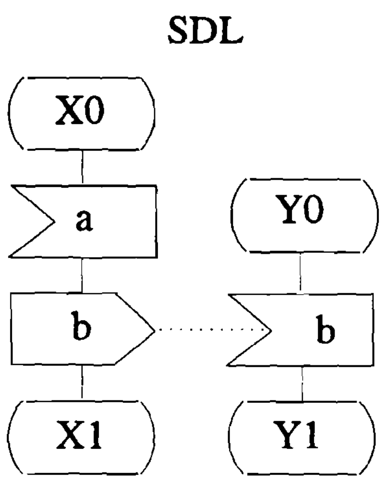
This causes a rephrasal of step 8 of the combination algorithm. With Expand, the states to be evaluated and Ready, the states that have already been evaluated, the algorithm is carried out by the following steps:

1. write the behaviours of A and B in the normal form,
2. calculate MTU(A) and MTU(B),
3. initialize Expand with S0,
4. define the causal relations,
5. select a state from Expand to be the current state,
6. remove this state from Expand,
7. add this state to Ready,
8. apply the synthesis rules for all input and output actions in the partial specification of S which are enabled, i.e. in In(S) or Out(S), to obtain a set of behaviour equations in the normal form that involve the enabled actions,
9. add the states resulting from 8 to Expand,
10. if Expand is empty then go to 5 and
11. reduce the obtained behaviour of S.

## 6.5 Linking CCS and SDL

In SDL, states must be followed by inputs. This is not necessary in CCS. CCS equations are put in the normal form, in which a new state is preceeded by only one action. Therefore sub states, indicated by a superscript number, are introduced. These are treated in the CCS description in the same way as normal states. They are not the same as states in SDL since in SDL states must be followed by inputs. A connector in SDL can also be represented by a sub state in CCS.

Tasks are internal actions and the figures 31 and 32 show how to treat input, output and decision.



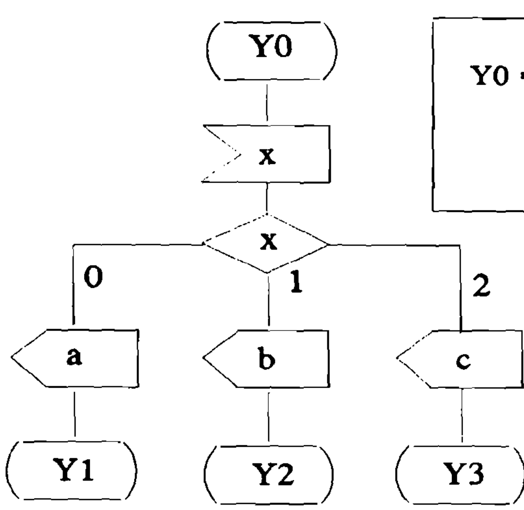

CCS

X0 = in1?.a: out2!.b: X1

Y0 = in1?.b: Y1

....... communication

*figure 31. communication in SDL and CCS*




Y0 = in1?.x: ( if x=0 then out1!.a: Y1
+ if x=1 then out1!.b: Y2
+ if x=2 then out1!.c: Y3 )

*figure 32. decision in SDL and CCS*

Actions in the SDL-diagrams may be abstracted, and queueing, timers, value passing may be provided for as described in this chapter.

# 7. Describing signalling systems no.7 and R2

Now we are ready to describe the signalling systems. The basis for this are the recommendations of the CCITT blue book on interworking [CCITT,89]. These are presented in SDL.

The subject is interworking from signalling system no.7 to R2. We need the logic procedures for incoming signalling system number 7 (TUP), for outgoing signalling system R2 and for interworking of signalling system no.7 to R2. These are given in appendix C.

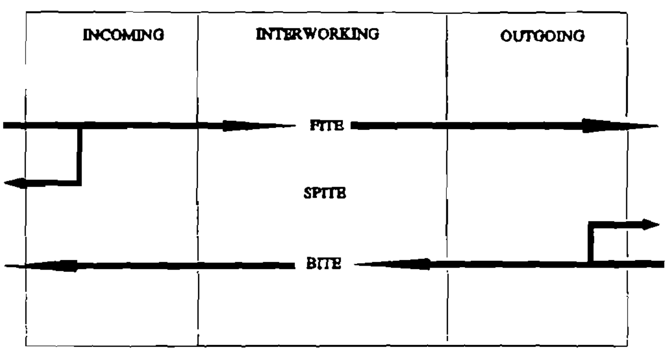
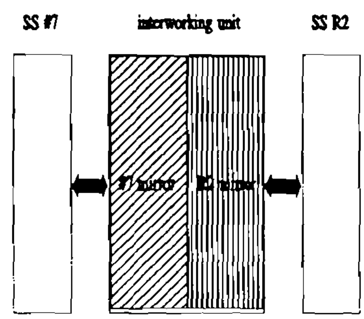



*figure 33. CCITT interworking no.7 → R2*




**incoming no.7 outgoing R2**

*figure 34. combining no.7 incoming and R2 outgoing behaviour*

As mentioned in chapter 5, the incoming no.7 description is the mirror image of the behaviour of system no.7 and so is the outgoing R2 description the mirror image of the R2 behaviour. S is the agent for mirrored no.7 behaviour, and R for mirrored R2 behaviour. States are numbered according to the SDL states, so S5 is the same as state 05 in the diagram. Derivatives are denoted with superscript numbers. Translating the SDL diagrams also yields step 1. of the combination algorithm.

We will now translate part of the incoming no.7 description into CCS as an example. In figure 35, the first part of that diagram is given. Note from the remaining of the diagram in the appendix, that connector 10 equals state S1 and connector 2 state S2.

This part receives the initial address message,IAM, or the initial address message with additional information. These consist of first the standard telephone label, consisting of destination and originating point code and circuit identification code, CIC, that indicates one speech circuit among those directly interconnecting destination and origin.

The following heading parts indicate the specific heading group in H0 and contains either a signal code or the format of more complex messages in H1. After the calling party's category,

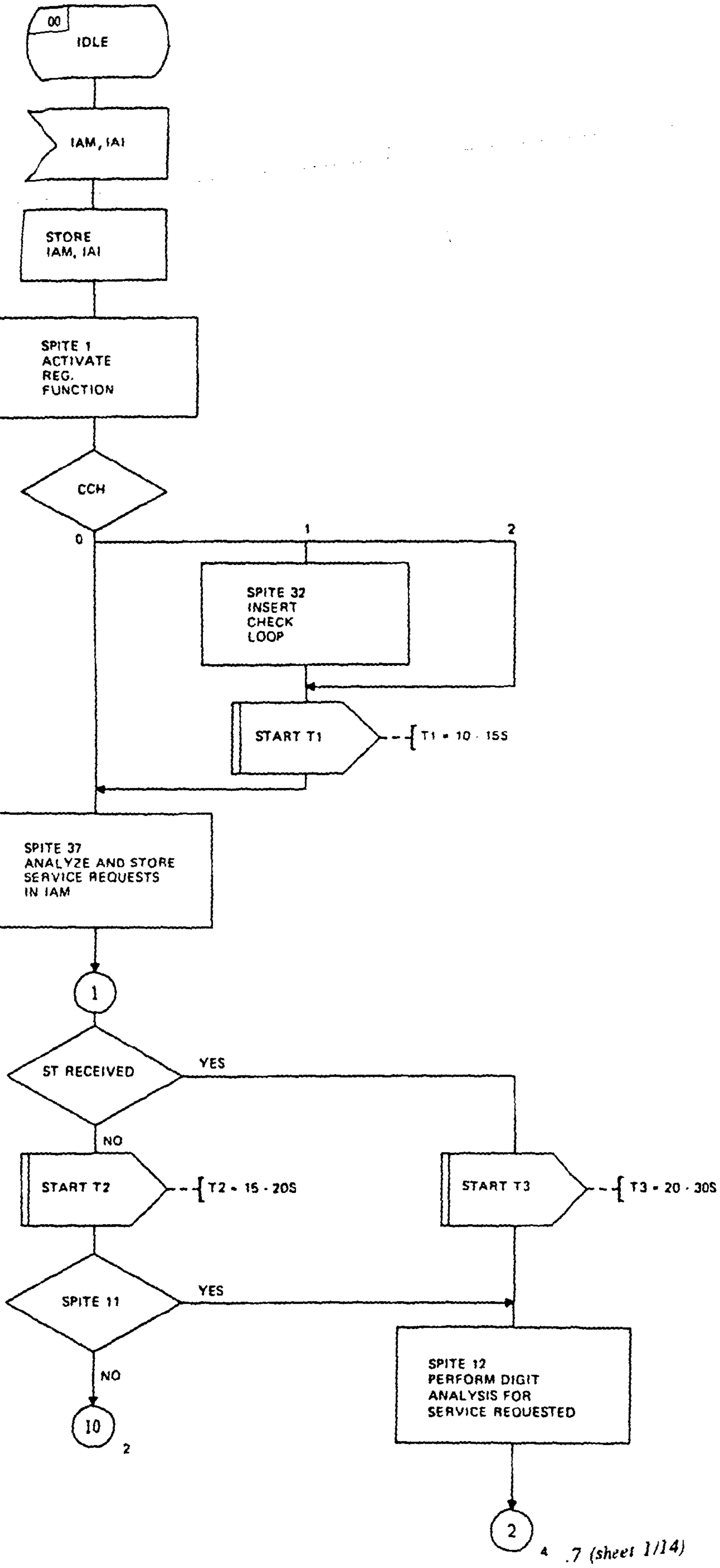



*figure 35. the first par... .7 (sheet 1/14)*

the message indicators give the nature of address and circuit etc. Address digits end the message.

Not all of the information given by IAM can be used by system R2. In stead of defining unique input ports for every possible different address message, value passing will be used. The address information in message IAM is stored internally. This can be explicitly denoted by S(IAM).

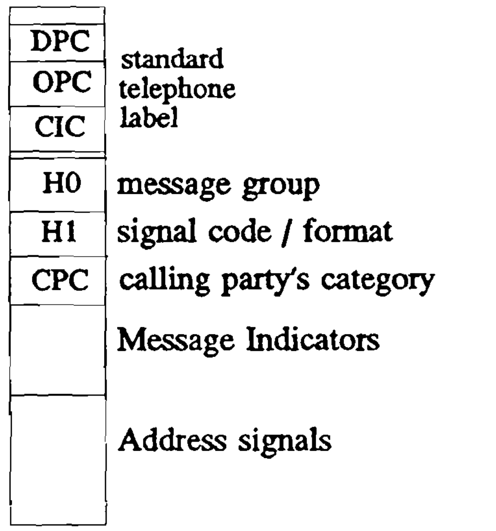



*figure 36. initial address message, IAM*

| | | |
|---|---|---|
| S0 | = in1?.IAM: $S0^1$(IAM) | information a stored |
| $S0^1$(IAM) | = SPITE_1: $S0^2$(IAM) | this is an internal action that activates the register function to keep a memory of all received signals |
| $S0^2$(IAM) | = if CCH=0 then $S0^3$(IAM) | continuity check not required |
| | + if CCH=1 then SPITE_32: $S0^4$(IAM) | or continuity check required on incoming circuit, followed by the insertion of the check loop |
| | + if CCH=2 then $S0^4$(IAM) | or continuity check performed on previous circuit |
| $S0^4$(IAM) | = start_T1: $S0^3$(IAM) | timer wait for continuity or continuity failure |
| $S0^3$(IAM) | = SPITE_37: $S0^5$(IAM) | analyse and store service requests |
| $S0^5$(IAM) | = if ST_received then start_T3: $S0^6$(IAM) | end of pulsing received, wait for clear forward |
| | + if not ST_rcvd then start_T2: $S0^7$(IAM) | or end of pulsing not received, wait for address complete |
| $S0^6$(IAM) | = SPITE_12: S2 | perform digit analysis for service requested |
| $S0^7$(IAM) | = if SPITE_11 then $S0^6$(IAM) | digit analysis shall be started |
| | + if not SPITE_11 then S1 | or digit analysis shall not be started |

Realizing that this is only a part of the first of 14 sheets, such a description for the whole system is elaborate. Combining it with the R2 behaviour in this way seems impossible.

At this stage, the purpose of the description must be discussed.

A description like the above seems to be enough to implement a system in practice. We will focus now on a description only concerning communication. In this case, the internal actions can be hidden. Further abstraction involves IF- and $\tau$-reduction.

Hiding internal actions and using the $\tau$-laws (1-4), the system would become:

$S0 \quad = \text{in1?.IAM}: S0^1$

$S0^1 \quad = \text{if ST\_received then } S2$

$\qquad + \text{if not ST\_received then } S0^2$

$S0^2 \quad = \text{if SPITE\_11 then } S2$

$\qquad + \text{if not SPITE\_11 then } S1$

Applying if reduction results in:

$S0 \quad = \text{in1?.IAM}: S0^1$

$S0^1 \quad = \tau: S2$

$\qquad + \tau: S0^2$

$S0^2 \quad = \tau: S2$

$\qquad + \tau: S1$

With $\tau$-reduction, and in doing so losing observation congruence:

$S0 \quad = \text{in1?.IAM}: S0^1$

$S0^1 \quad = S2$

$\qquad S0^2$

$S0^2 \quad = S2$

$\qquad S1$

or:

$S0 \quad = \text{in1?.IAM}: (S1+S2)$ or: $S0 \quad = \text{in1?.IAM}: S_{1,2}$

The description above has now been brought back to one equation.

The following will focus on communication. Hiding, if- and $\tau$-reduction will be applied to derive an interworking unit according to the combination algorithm. In this way, the interworking can be verified to be free of deadlock.

In the SDL-sheets of appendix C, the internal actions are struck out. The CCS-equations after this abstraction over internal actions are listed in appendix A.1.

The next stage is if- and $\tau$-reduction. The resulting equations are in appendix A.2.

As in the example, $\tau$-reduction causes the combination of states: (S1 + S2) $\equiv$ $S_{1,2}$. Care must be taken, cause only if S1 contains the S2 actions as well $\alpha$: S1 may be replaced by $S_{1,2}$.

The equations, thus obtained are given in appendix A.3. They are the first step in the application of the combination algorithm.

The *continuity check* procedures still are for further study in CCITT. They are considered to be internal actions and return to the state that invoked them. Since the TUP-functions will be covered by the ISCP in future, this further study is not likely to take place.

The abbreviations of no.7 signals are listed in appendix C. Their meaning is described in the CCITT blue book.

The R2 group A, B, I and II signals are listed in appendix E.

Messages that lead to the same resulting state are grouped together. This can be regarded as replacing them by a value that belongs to an action with their common label. For instance, the unsuccessfull messages that result in state S10 are replaced by UM.

# 8. Deriving interworking

The behaviours, obtained in the previous chapter are the $\tau$-reduced mirrors with unique action occurrences. S knows 15 states, and R 20. This means that 300 is upper limit for the number of states of the combined behaviour.

### Further abstraction

In order to simplify the behaviours, the next abstraction is made.

The GRQ/GSM protocol of system no.7 is neglected. The general request message, GRM, and general forward set-up information message do not have equivalent messages in system R2. The GRM, GMS, GRQ and GSM actions are deleted.

The R2 group A signals 1,2,5,7-14 are grouped together. These are backward requests that do not concern congestion, a changeover to group B signals nor a set-up speech condition. As a result, the R4 states, except $R4^3$, and $R3^1$ are combined to a new state R4 and state $R5^1$ can be omitted.

The resulting behaviours are given in appendix B.

### Causal relations

Certain actions on one side can only occur if a certain action on the other side has occurred. This is taken into account by defining causal relations. Defining the causal relations requires knowledge of the systems and their signals. An example is the R2 signal SEIZING. This message can only be sent if the address message IA on the other side has been received. An example of a signal that is not controlled by a causal relation is the R2 signal CLEAR_FORWARD. This signal is sent either when a CLF from the no.7 side has been received, or when an internal action gave reason to sent a CLEAR_FORWARD.

The causal relations are given in table 3. The IAM message contains the language and discriminating digits. The backward signals are the no.7 equivalents of the R2 signals.

| | | |
|---|---|---|
| FORWARD | | |
| IA | → | SEIZING |
| IA | → | LANGUAGE_DIGIT |
| BACKWARD | | |
| ADC | ← | A6 |
| AD | ← | A6 |
| AN | ← | ANSWER |
| CFC | ← | A4,15 |
| AF | ← | B6,7 |
| CBK | ← | CLEAR_BACK |

*table 3. causal relations*

**Initial state**

The initial state is the combination of the idle states of no.7 and R2. This is the state in which call set up starts, and the state to which the system returns when the connection is cleared.

Results

Starting from the initial state, possible actions are evaluated. If an action is possible, also depends on the causal relations. This is done for every state that can be reached.

In appendix B, the states are ordered by increasing numbers. The result is 88 states.

**Verification**

The interworking unit can be verified by expansion. In order to do this the following must be calculated:

- the expansion of the behaviour of the interworking unit and the SS #7 behaviour, this must be mirror-observationally equivalent to the R2 behaviour, and
- the expansion of the behaviour of the interworking unit and the R2 behaviour, this must be mirror-observationally equivalent to the SS #7 behaviour.

# 9. Conclusions

This report shows that complex systems such as signalling systems can be well described in CCS and can further be combined to derive interworking between them. The interworking unit has not been verified by expansion yet.

Starting with a large system, the descriptions in this report were simplyfied by certain abstractions; in this way losing certain properties of the system. Next, these certain properties can be added bacxk to the systems, e.g. internal decisions and timers.

CCS with its properties and features is a means to describe communicating systems.

Especially the interface derivation using the combination algorithm is important. The application to complex systems will yield a large set of expressions for the combined behaviour.

Using value passing as well as a carefull definition of causal relations can result in a considerable reduction.

The advantage of performing the comination manually lies in the fact that the designer is forced to study the behaviours thoroughly and thus be able to define the causal relations carefully.

It is important that the assumptions made and the abstraction done are described.

An important advantage over the SDL description is the possibilty to detect deadlock

CCS is not commonly used in telecommunications while two other calculi are: Lotos and SDL. Lotos is strongly based on CCS, so it may be useful to implement the techniqes described into Lotos and SDL. An advantage is that CCITT describes it signalling systems in SDL.

# References


CCITT,89 — CCITT, ITU: Blue book
fascicle VI.3 Specifications of signalling system no.6
fascicle VI.4 Specifications of signalling systems R1 and R2
fascicle VI.6 Interworking of signalling systems
fascicle VI.7 Specifications of signalling system no.7
fascicle VI.8 Specifications of signalling system no.7
1989, ITU, Geneva

Craveur,87 — Craveur, J., Algalarrondo, D., Tardieu, M., Monjoin, C.:CCITT no.7 common channel signalling in the French telecommunication network
1987, proceedings international switching symposium, ISS, IEEE

Funke,90 — Funke, G.: SS7 - A universal signalling system for today, tomorrow, ... the day after tomorrow?
1990, September, Philips telecommunication review, 48-3

Heidermark,84 — Heidermark, A., Borgström, O.: Field trial with signalling system number 7 in AXE 10
1984, proceedings international switching symposium, ISS, IEEE

Hills,79 — Hills, M.T.: Telecommunications switching principles
1979, George Allen & Unwin, London

Hogrefe,89 — Hogrefe, D.: Estelle, Lotos und SDL
1989, Springer Verlag, Berlin

Milner,80 — Milner, R.: A calculus of communicating systems
1980, Springer Verlag, Berlin

Milner,89 — Milner, R.: Communication and concurrency
1989, Prentice Hall, London

Mitra,91 — Mitra, N., Usiskin, S.D.: Relationship of the signaling system no.7 protocol architecture to the OSI reference model
1991, January, IEEE Network magazine

Modaressi,90 — Modaressi, A. R., Skoog, R.A: Signaling system no.7: A tutorial

| | |
|---|---|
| | 1990, July, IEEE Communications magazine |
| Koomen,91 | Koomen, C.J.: The design of communicating systems<br>1991, Kluwer academic publishers, Dordrecht |
| Kroes,79 | de Kroes, J.L., Lemstra: Electronische automatische telefonie<br>1979, TU Delft (lecture notes) |
| Redmill,90 | Redmill, F.J., Valdar, A.R.: SPC digital telephone exchanges<br>1990, Peter Peregrinus, London |
| Tanenbaum,88 | Tanenbaum, A.S.: Computer networks<br>1988, Prentice-Hall, London |
| Welch,79 | Welch, S.: Signalling in telecommunications networks<br>1979, IEE, London; Peter Peregrinus, New York |

# Index

# Appendix A. Behaviour equations for systems no.7 and R2

## Behaviours translated from SDL

### Incoming signalling system no.7

■message grouping

AF AFC, AFN or AFX

IA IAM or IAI

UE unsuccessful events: SEC, CGC, NNC, ADI, UNN, SSB, LOS, CFL or SST

UM unsuccessful messages: UNN, SST, SEC, CGC, ACB or DPN

SA SAM or SAO

■selection

STR ST (end of pulsing) received

SDA start digit analysis

■remarks

- Connector P1, procedure for continuity recheck incoming, is still for further study in CCITT. It will be assumed to be an internal action returning to state 01, wait for further digits (sheet 2/14).
- Connector P2, test call procedure to interworking, is also for further study and therefore neglected (sheet 6/14).
- Connector P3, out of service, is also neglected (sheet 11/14).
- A continuity check may be required (states S1, S2 and S5) but the continuity check indicator CCH does not affect any state transition.

$S0$ = in1?.IA: $S0^1$ + tau: $S1^2$

$S0^1$ = if STR then S2
+ if not STR and SDA then S2
+ if not STR and not SDA then S1

$S1$ = in1?.SA: $S0^1$
+ in1?.COT: S4
+ in1?.CCF: S1
{assume P1 returns to S1}
+ in1?.CLF: $S1^1$
+ tau: $S1^3$

$S1^1$ = out1!.RLG: S0

$S1^2$ = out1!.ADI: S10

$S1^3$ = out1!.CFL: S10

$S2$ = tau: S3
{assume P2 returns to S2, COT not received}
+ tau: ( if not STR then S1
+ if STR then $S1^2$ )

| | | |
|---|---|---|
| | + | in1?.SA: S2 |
| | + | tau: $S1^2$ |
| | + | tau: out1!.GRM: S2 |
| | + | in1?.CLF:$S1^1$ |
| | + | tau: $S1^3$ |
| | + | in1?.COT: S5 |
| | + | in1?.GMS: S2 |
| | + | tau: out1!.UM: S10 |
| S3 | = | in1?.COT: S6 |
| | + | in1?.CCF: P1 |
| | + | in1?.SA: S3 |
| | + | in1?.CLF: $S1^1$ |
| | + | tau: $S1^3$ |
| | + | tau: $S1^2$ |
| | + | tau: out1!.GRQ: S3 |
| | + | in1?.GSM: S3 |
| | + | tau: out1!.UE: S10 |
| S4 | = | in1?.SA: $S4^1$ |
| | + | in1?.CLF: $S1^1$ |
| $S4^1$ | = | if not STR and not SDA then S4 |
| | | + if STR then S5 |
| | | + if not STR and SDA then S5 |
| S5 | = | tau: S6 {assume P2 returns to S5, COT received} |
| | + | tau: ( if not STR then S4 + if STR then $S1^2$ ) |
| | + | in1?.SA: S5 |
| | + | in1?.CLF: $S1^1$ |
| | + | tau: $S1^2$ |
| | + | tau: $S1^3$ |
| | + | tau: out1!.UM: S10 |

| | | |
|---|---|---|
| S6 | = | in1?.SA: S6 |
| | + | tau: out1!.ADC: S7 |
| | + | tau: out1!.ADN: S8 |
| | + | tau: out1!.ADX: S8 |
| | + | tau: out1!.AF: S9 |
| | + | in1?.CLF: $S1^1$ |
| | + | tau: $S1^2$ |
| | + | tau: $S1^1$ |
| | + | in1?.GSM: S6 |
| | + | tau: out1!.GRQ: S6 |
| S7 | = | tau: S8 |
| | + | tau: out1!.CGC: S10 |
| | + | tau: out1!.NNC: S10 |
| | + | tau: $S1^3$ |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.GSM: S7 |
| S8 | = | tau: out1!.ANC: S11 |
| | + | tau: out1!.ANN: S11 |
| | + | tau: out1!.CGC: S10 |
| | + | tau: out1!.NNC: S10 |
| | + | tau: $S1^3$ |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.GSM: S8 |
| | + | in1?.FOT: S8 |
| S9 | = | tau: out1!.ANC: S11 |
| | + | tau: out1!.ANN: S11 |
| | + | tau: $S1^3$ |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.GSM: S9 |
| | + | in1?.FOT: S9 |
| S10 | = | in1?.CLF:$S1^1$ |
| | + | tau: out1!.CLF: S13 |

| | | |
|---|---|---|
| S11 | = | tau: out1!.CBK: S12 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S11 |
| S12 | + | tau: out1!.RAN: S11 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S12 |
| S13 | = | in1?.CLF: $S1^1$ |
| | + | tau: out1!.CFL: S13 |

**Outgoing signalling system R2**

| | | |
|---|---|---|
| R0 | = | tau: R1 |
| R1 | = | tau: R2 |
| | + | tau: R0 |
| R2 | = | tau: R3 |
| | + | tau: out2!.SEIZING: R4 |
| | + | tau: R0 |
| R3 | = | tau: out2!.SEIZING: R3$^1$ |
| | + | tau: R0 |
| R3$^1$ | = | out2!.LANGUAGE_DIGIT1: R4 |
| | + | out2!.LANGUAGE_DIGIT2: R4 |
| | + | out2!.LANGUAGE_DIGIT3: R4 |
| | + | out2!.LANGUAGE_DIGIT4: R4 |
| | + | out2!.LANGUAGE_DIGIT5: R4 |
| | + | out2!.DISCRIMINATING_DIGIT: R4 |
| R4 | = | in2?.A1: R4$^1$ |
| | + | in2?.A3: R4$^3$ |
| | + | in2?.A5: R4$^4$ |
| | + | in2?.A4: R9$^1$ |
| | + | in2?.A6: R7 |
| | + | in2?.A15: R9$^1$ |
| | + | in2?.A14: R4$^5$ |
| | + | in2?.A9: R4$^6$ |
| | + | in2?.A10: R4$^6$ |
| | + | in2?.A2: R4$^1$ |
| | + | in2?.A7: R4$^1$ |
| | + | in2?.A8: R4$^1$ |
| | + | in2?.A11: R4$^7$ |
| | + | in2?.A12: R3$^1$ |
| | + | out2!.A13: R4$^8$ |
| | + | tau: R9$^1$ |
| R4$^1$ | = | if Z-DIGIT_NEXT then R3$^1$ |
| | | + if not Z-DIGIT_NEXT then R4$^2$ |
| R4$^2$ | = | tau: R5 |
| | + | out2!.ADDRESS: R4 |
| R4$^3$ | = | out2!.II7: R6 |
| | + | out2!.II8: R6 |
| | + | out2!.II9: R6 |
| R4$^4$ | = | out2!.II7: R4 |
| | + | out2!.II8: R4 |
| | + | out2!.II9: R4 |
| R4$^5$ | = | tau: out2!.I14: R4 |
| | + | tau: R4$^1$ |
| R4$^6$ | = | out2!.I12: R4 |
| R4$^7$ | = | tau: out2!.I12: R4 |
| | + | tau: out2!.I14: R4 |
| R4$^8$ | = | tau: out2!.I14: R4 |
| | + | tau: out2!.I13: R4 |
| R5 | = | tau: out2!.ADDRESS: R4 |
| | + | in2?.A3: R5$^1$ |
| | + | in2?.A4: R9$^1$ |
| | + | in2?.A6: R7 |
| | + | in2?.A15: R9$^1$ |
| | + | tau: R9$^1$ |
| R5$^1$ | = | tau: out2!.II7: R6 |

| | | |
|---|---|---|
| | + | tau: out2!.II9: R6 |
| | + | tau: out2!.II8: R6 |
| R6 | = | in2?.B6: R7 |
| | + | in2?.B7: R7 |
| | + | in2?.B2: $R9^1$ |
| | + | in2?.B3: $R9^1$ |
| | + | in2?.B4: $R9^1$ |
| | + | in2?.B5: $R9^1$ |
| | + | in2?.B8: $R9^1$ |
| | + | tau: $R9^1$ |
| R7 | = | in2?.ANSWER: R8 |
| | + | tau: $R9^1$ |
| R8 | = | in2?.CLEAR_BACK: R9 |
| | + | tau: $R9^1$ |
| R9 | = | in2?.ANSWER: R8 |
| | + | tau: $R9^1$ |
| $R9^1$ | = | out2!.CLEAR_FORWARD: R10 |
| R10 | = | in2?.RELEASE_GUARD: R0 |

{State 6: signals B1, B9-B15 are spare for national use, also neglected}

## Behaviours after if and τ reduction

### Incoming signalling system no.7

| | | |
|---|---|---|
| S0 | = | in1?.IA: $S0^{1}$ |
| $S0^{1}$ | = | S2 |
| | + | S1 |
| S1 | = | in1?.SA: $S0^{1}$ |
| | + | in1?.COT: S4 |
| | + | in1?.CCF: S1 |
| | + | in1?.CLF: $S1^{1}$ |
| | + | $S1^{3}$ |
| | + | $S1^{2}$ |
| $S1^{1}$ | = | out1!.RLG: S0 |
| $S1^{2}$ | = | out1!.ADI: S10 |
| $S1^{3}$ | = | out1!.CFL: S10 |
| S2 | = | S3 |
| | + | S1 |
| | + | $S1^{2}$ |
| | + | in1?.SA: S2 |
| | + | $S1^{2}$ |
| | + | out1!.GRM: S2 |
| | + | in1?.CLF:$S1^{1}$ |
| | + | $S1^{3}$ |
| | + | in1?.COT: S5 |
| | + | in1?.GMS: S2 |
| | + | out1!.UM: S10 |
| S3 | = | in1?.COT: S6 |
| | + | in1?.SA: S3 |
| | + | in1?.CLF: $S1^{1}$ |
| | + | $S1^{3}$ |
| | + | $S1^{2}$ |
| | + | out1!.GRQ: S3 |
| | + | in1?.GSM: S3 |
| | + | out1!.UE: S10 |
| S4 | = | in1?.SA: $S4^{1}$ |
| | + | in1?.CLF: $S1^{1}$ |
| $S4^{1}$ | = | S4 |
| | + | S5 |
| S5 | = | S6 |
| | + | S4 |
| | + | $S1^{2}$ |
| | + | in1?.SA: S5 |
| | + | in1?.CLF: $S1^{1}$ |
| | + | $S1^{2}$ |
| | + | $S1^{3}$ |
| | + | out1!.UM: S10 |
| S6 | = | in1?.SA: S6 |
| | + | out1!.ADC: S7 |
| | + | out1!.ADN: S8 |
| | + | out1!.ADX: S8 |
| | + | out1!.AF: S9 |
| | + | in1?.CLF: $S1^{1}$ |
| | + | $S1^{2}$ |
| | + | $S1^{1}$ |
| | + | in1?.GSM: S6 |
| | + | out1!.GRQ: S6 |
| S7 | = | S8 |

| | | |
|---|---|---|
| | + | out1!.CGC: S10 |
| | + | out1!.NNC: S10 |
| | + | $S1^3$ |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.GSM: S7 |
| | | |
| S8 | = | out1!.ANC: S11 |
| | + | out1!.ANN: S11 |
| | + | out1!.CGC: S10 |
| | + | out1!.NNC: S10 |
| | + | $S1^3$ |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.GSM: S8 |
| | + | in1?.FOT: S8 |
| | | |
| S9 | = | out1!.ANC: S11 |
| | + | out1!.ANN: S11 |
| | + | $S1^3$ |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.GSM: S9 |
| | + | in1?.FOT: S9 |
| | | |
| S10 | = | in1?.CLF:$S1^1$ |
| | + | out1!.CLF: S13 |
| | | |
| S11 | = | out1!.CBK: S12 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S11 |
| | | |
| S12 | + | out1!.RAN: S11 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S12 |
| | | |
| S13 | = | in1?.CLF: $S1^1$ |
| | + | out1!.CFL: S13 |

**Outgoing signalling system R2**

$+\ R9^{1}$

R0 = R1

R1 = R2
+ R0

R2 = R3
+ out2!.SEIZING: R4
+ R0

R3 = out2!.SEIZING: $R3^{1}$
+ R0

$R3^{1}$ = out2!.LANGUAGE_DIGIT1: R4
+ out2!.LANGUAGE_DIGIT2: R4
+ out2!.LANGUAGE_DIGIT3: R4
+ out2!.LANGUAGE_DIGIT4: R4
+ out2!.LANGUAGE_DIGIT5: R4
+ out2!.DISCRIMINATING_DIGIT: R4

R4 = in2?.A1: $R4^{1}$
+ in2?.A3: $R4^{3}$
+ in2?.A5: $R4^{4}$
+ in2?.A4: $R9^{1}$
+ in2?.A6: R7
+ in2?.A15: $R9^{1}$
+ in2?.A14: $R4^{5}$
+ in2?.A9: $R4^{6}$
+ in2?.A10: $R4^{6}$
+ in2?.A2: $R4^{1}$
+ in2?.A7: $R4^{1}$
+ in2?.A8: $R4^{1}$
+ in2?.A11: $R4^{7}$
+ in2?.A12: $R3^{1}$
+ out2!.A13: $R4^{8}$

$R4^{1}$ = $R3^{1}$
+ $R4^{2}$

$R4^{2}$ = R5
+ out2!.ADDRESS: R4

$R4^{3}$ = out2!.I17: R6
+ out2!.I18: R6
+ out2!.I19: R6

$R4^{4}$ = out2!.I17: R4
+ out2!.I18: R4
+ out2!.I19: R4

$R4^{5}$ = out2!.I14: R4
+ $R4^{1}$

$R4^{6}$ = out2!.I12: R4

$R4^{7}$ = out2!.I12: R4
+ out2!.I14: R4

$R4^{8}$ = out2!.I14: R4
+ out2!.I13: R4

R5 = out2!.ADDRESS: R4
+ in2?.A3: $R5^{1}$
+ in2?.A4: $R9^{1}$
+ in2?.A6: R7
+ in2?.A15: $R9^{1}$
+ $R9^{1}$

$R5^{1}$ = out2!.I17: R6

| | | |
|---|---|---|
| | + | out2!.II9: R6 |
| | + | out2!.II8: R6 |
| R6 | = | in2?.B6: R7 |
| | + | in2?.B7: R7 |
| | + | in2?.B2: $R9^1$ |
| | + | in2?.B3: $R9^1$ |
| | + | in2?.B4: $R9^1$ |
| | + | in2?.B5: $R9^1$ |
| | + | in2?.B8: $R9^1$ |
| | + | $R9^1$ |
| R7 | = | in2?.ANSWER: R8 |
| | + | $R9^1$ |
| R8 | = | in2?.CLEAR_BACK: R9 |
| | + | $R9^1$ |
| R9 | = | in2?.ANSWER: R8 |
| | + | $R9^1$ |
| $R9^1$ | = | out2!.CLEAR_FORWARD: R10 |
| R10 | = | in2?.RELEASE_GUARD: R0 |

## Behaviours after refinement

**Incoming signalling system no.7**

| | | | | | |
|---|---|---|---|---|---|
| States: | S0 | $S1^1$ | S4 | S8 | S11 |
| | $S_{1,2,3}$ | S3 | S6 | S9 | S12 |
| | $S_{4,5,6}$ | $S_{7,8}$ | S10 | S13 | |

Unique actions: COT_1, COT_2, COT_3, SA and SA_2.

| | | |
|---|---|---|
| S0 | = | in1?.IA: $S_{1,2,3}$ |
| | | |
| $S_{1,2,3}$ | = | in1?.SA: $S_{1,2,3}$ |
| | + | in1?.COT_1: S4 |
| | + | in1?.CCF: $S_{1,2,3}$ |
| | + | in1?.CLF: $S1^1$ |
| | + | out1!.ADI: S10 |
| | + | out1!.CFL: S10 |
| | + | out1!.GRM: $S_{1,2,3}$ |
| | + | in1?.COT_2: $S_{4,5,6}$ |
| | + | in1?.GMS: $S_{1,2,3}$ |
| | + | out1!.UM: S10 |
| | + | in1?.COT_3: S6 |
| | + | out1!.GRQ: S3 |
| | + | in1?.GSM: S3 |
| | + | out1!.UE: S10 |
| | | |
| $S1^1$ | = | out1!.RLG: S0 |
| | | |
| S3 | = | in1?.COT_3: S6 |
| | + | in1?.SA: S3 |
| | + | in1?.CLF: $S1^1$ |
| | + | out1!.CFL: S10 |
| | + | out1!.ADI: S10 |
| | + | out1!.GRQ: S3 |
| | + | in1?.GSM: S3 |
| | + | out1!.UE: S10 |
| | | |
| S4 | = | in1?.SA: S4 |
| | + | in1?.CLF: $S1^1$ |
| | | |
| $S_{4,5,6}$ | = | in1?.SA: $S_{4,5,6}$ |
| | + | in1?.CLF: $S1^1$ |
| | + | out1!.RLG: S0 |
| | + | out1!.ADI: S10 |
| | + | out1!.UM: S10 |
| | + | in1?.SA_2: S6 |
| | + | out1!.ADC: $S_{7,8}$ |
| | + | out1!.ADN: S8 |
| | + | out1!.ADX: S8 |
| | + | out1!.AF: S9 |
| | + | in1?.GSM: S6 |
| | + | out1!.GRQ: S6 |
| | | |
| S6 | = | in1?.SA_2: S6 |
| | + | out1!.ADC: $S_{7,8}$ |
| | + | out1!.ADN: S8 |
| | + | out1!.ADX: S8 |
| | + | out1!.AF: S9 |
| | + | in1?.CLF: $S1^1$ |
| | + | out1!.RLG: S0 |
| | + | out1!.ADI: S10 |
| | + | in1?.GSM: S6 |
| | + | out1!.GRQ: S6 |

| | | | | |
|---|---|---|---|---|
| $S_{7,8}$ | = | out1!.ANC: S11 | + | out1!.CFL: S13 |
| $\{S7 = S_{7,8}\}$ | + | out1!.ANN: S11 | | |
| | + | out1!.CGC: S10 | | |
| | + | out1!.NNC: S10 | | |
| | + | out1!.CFL: S10 | | |
| | + | in1?.CLF: $S1^1$ | | |
| | + | in1?.FOT: S8 | | |
| | + | in1?.GSM: $S_{7,8}$ | | |
| | | | | |
| S8 | = | out1!.ANC: S11 | | |
| $\{S8 \neq S_{7,8}\}$ | + | out1!.ANN: S11 | | |
| | + | out1!.CGC: S10 | | |
| | + | out1!.NNC: S10 | | |
| | + | out1!.CFL: S10 | | |
| | + | in1?.CLF: $S1^1$ | | |
| | + | in1?.GSM: S8 | | |
| | + | in1?.FOT: S8 | | |
| | | | | |
| S9 | = | out1!.ANC: S11 | | |
| | + | out1!.ANN: S11 | | |
| | + | out1!.CFL: S10 | | |
| | + | in1?.CLF: $S1^1$ | | |
| | + | in1?.GSM: S9 | | |
| | + | in1?.FOT: S9 | | |
| | | | | |
| S10 | = | in1?.CLF:$S1^1$ | | |
| | + | out1!.CLF: S13 | | |
| | | | | |
| S11 | = | out1!.CBK: S12 | | |
| | + | in1?.CLF: $S1^1$ | | |
| | + | in1?.FOT: S11 | | |
| | | | | |
| S12 | + | out1!.RAN: S11 | | |
| | + | in1?.CLF: $S1^1$ | | |
| | + | in1?.FOT: S12 | | |
| | | | | |
| S13 | = | in1?.CLF: $S1^1$ | | |

**Outgoing signalling system R2**

States: $R_{0,1,2,3}$ $R3^1$ R4 $R4^1$ $R4^3$ $R4^4$
$R4^5$ $R4^6$ $R4^7$ $R4^8$ $R5^1$ R6
R7 R8 R9 R10

Unique actions: SEIZING_1 and SEIZING_2

$R_{0,1,2,3}$ = out2!.SEIZING_1: R4
+ out2!.SEIZING_2: $R3^1$

$R3^1$ = out2!.LANGUAGE_DIGIT1: R4
+ out2!.LANGUAGE_DIGIT2: R4
+ out2!.LANGUAGE_DIGIT3: R4
+ out2!.LANGUAGE_DIGIT4: R4
+ out2!.LANGUAGE_DIGIT5: R4
+ out2!.DISCRIMINATING_DIGIT: R4

R4 = in2?.A1: $R4^1$
+ in2?.A3: $R4^3$
+ in2?.A5: $R4^4$
+ in2?.A4: $R9^1$
+ in2?.A6: R7
+ in2?.A15: $R9^1$
+ in2?.A14: $R4^5$
+ in2?.A9: $R4^6$
+ in2?.A10: $R4^6$
+ in2?.A2: $R4^1$
+ in2?.A7: $R4^1$
+ in2?.A8: $R4^1$
+ in2?.A11: $R4^7$
+ in2?.A12: $R3^1$
+ out2!.A13: $R4^8$
+ out2!.CLEAR_FORWARD: R10

$R4^1$ = out2!.LANGUAGE_DIGIT1: R4
+ out2!.LANGUAGE_DIGIT2: R4
+ out2!.LANGUAGE_DIGIT3: R4
+ out2!.LANGUAGE_DIGIT4: R4
+ out2!.LANGUAGE_DIGIT5: R4
+ out2!.DISCRIMINATING_DIGIT: R4
+ out2!.ADDRESS: R4
+ in2?.A3: $R5^1$
+ in2?.A4: $R9^1$
+ in2?.A6: R7
+ in2?.A15: $R9^1$
+ out2!.CLEAR_FORWARD: R10

$R4^3$ = out2!.II7: R6
+ out2!.II8: R6
+ out2!.II9: R6

$R4^4$ = out2!.II7: R4
+ out2!.II8: R4
+ out2!.II9: R4

$R4^5$ = out2!.I14: R4
+ out2!.LANGUAGE_DIGIT1: R4
+ out2!.LANGUAGE_DIGIT2: R4
+ out2!.LANGUAGE_DIGIT3: R4
+ out2!.LANGUAGE_DIGIT4: R4

\+ out2!.LANGUAGE_DIGIT5: R4
\+ out2!.DISCRIMINATING_DIGIT: R4
\+ out2!.ADDRESS: R4
\+ in2?.A3: $R5^1$
\+ in2?.A4: $R9^1$
\+ in2?.A6: R7
\+ in2?.A15: $R9^1$
\+ out2!.CLEAR_FORWARD: R10

$R4^6$ = out2!.I12: R4

$R4^7$ = out2!.I12: R4
\+ out2!.I14: R4

$R4^8$ = out2!.I14: R4
\+ out2!.I13: R4

R5 = out2!.ADDRESS: R4
\+ in2?.A3: $R5^1$
\+ in2?.A4: $R9^1$
\+ in2?.A6: R7
\+ in2?.A15: $R9^1$
\+ out2!.CLEAR_FORWARD: R10

$R5^1$ = out2!.I17: R6
\+ out2!.I19: R6
\+ out2!.I18: R6

R6 = in2?.B6: R7
\+ in2?.B7: R7
\+ in2?.B2: $R9^1$
\+ in2?.B3: $R9^1$
\+ in2?.B4: $R9^1$
\+ in2?.B5: $R9^1$
\+ in2?.B8: $R9^1$
\+ out2!.CLEAR_FORWARD: R10

R7 = in2?.ANSWER: R8
\+ out2!.CLEAR_FORWARD: R10

R8 = in2?.CLEAR_BACK: R9
\+ out2!.CLEAR_FORWARD: R10

R9 = in2?.ANSWER: R8
\+ out2!.CLEAR_FORWARD: R10

$R9^1$ = out2!.CLEAR_FORWARD: R10

R10 = in2?.RELEASE_GUARD: $R_{0,1,2,3}$

# Appendix B. Combined behaviour

## Behaviours after further abstraction and refinement

### Incoming signalling system no.7

| | | | | | |
|---|---|---|---|---|---|
| ■States: | S0 | $S1^1$ | S4 | S8 | S11 |
| | $S_{1,2,3}$ | S3 | S6 | S9 | S12 |
| | $S_{4,5,6}$ | $S_{7,8}$ | S10 | S13 | |

■Unique actions: COT_1, COT_2, and COT_3

■Abstraction:

- no GRM/GMS nor GRQ/GSM cycle
- UM = UM, ADI, CFL, UE: unsuccessful messages leading to a CLF.

■Message grouping:

- CA = CFL, ADI: call failure/ adddress incomplete
- AD = ADN, ADX: addres complete; no charge/coin box
- AN = ANC, ANN: answer; charge/no charge
- CFC = CGC, NNC, CFL: call failure/ congestion

| | | |
|---|---|---|
| S0 | = | in1?.IA: $S_{1,2,3}$ |
| | | |
| $S_{1,2,3}$ | = | in1?.SA: $S_{1,2,3}$ |
| | + | in1?.COT_1: S4 |
| | + | in1?.CCF: $S_{1,2,3}$ |
| | + | in1?.CLF: $S1^1$ |
| | + | out1!.UM: S10 |
| | + | in1?.COT_2: $S_{4,5,6}$ |
| | + | in1?.COT_3: S6 |
| | | |
| $S1^1$ | = | out1!.RLG: S0 |
| | | |
| S4 | = | in1?.SA: S4 |
| | + | in1?.CLF: $S1^1$ |
| | | |
| $S_{4,5,6}$ | = | in1?.SA: $S_{4,5,6}$ |
| | + | in1?.CLF: $S1^1$ |
| | + | out1!.RLG: S0 |
| | + | out1!.UM: S10 |
| | + | out1!.ADC: $S_{7,8}$ |
| | + | out1!.AD: S8 |
| | + | out1!.AF: S9 |
| | | |
| S6 | = | in1?.SA_2: S6 |
| | + | out1!.ADC: $S_{7,8}$ |
| | + | out1!.AD: S8 |
| | + | out1!.AF: S9 |
| | + | in1?.CLF: $S1^1$ |
| | + | out1!.RLG: S0 |
| | + | out1!.ADI: S10 |
| | | |
| $S_{7,8}$ | = | out1!.AN: S11 |

| | | |
|---|---|---|
| | + | out1!.CFC: S10 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S8 |
| S8 | = | out1!.AN: S11 |
| {S8 $\neq S_{7,8}$} | + | out1!.UM: S10 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S8 |
| S9 | = | out1!.ANC: S11 |
| | + | out1!.CFL: S10 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S9 |
| S10 | = | in1?.CLF:$S1^1$ |
| | + | out1!.CLF: S13 |
| S11 | = | out1!.CBK: S12 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S11 |
| S12 | + | out1!.RAN: S11 |
| | + | in1?.CLF: $S1^1$ |
| | + | in1?.FOT: S12 |
| S13 | = | in1?.CLF: $S1^1$ |
| | + | out1!.CFL: S13 |

**Outgoing signalling system R2**

States: $R_{0,1,2,3}$ $R3^1$ R4 $R4^1$ $R4^3$ $R4^4$
$R4^5$ $R4^6$ $R4^7$ $R4^8$ $R5^1$ R6
R7 R8 R10

■Refinement

- LANGUAGE_DIGIT is either 1,2,3 4,5 or discriminating digit
- R9 = R7

■Abstraction:

- BACKWARD = A1,A2, A5, A7-A14: backward requests; group A messages not concerning congestion (A4,A15), a changeover to group B signals (A3), nor set up speech conditions (A6).

* this leads to combining states $R3^1$ and the R4 states, except $R4^3$, into state R4, and the omission of $R5^1$

$R_{0,1,2,3}$ = out2!.SEIZING: R4

R4 = in2?.BACKWARD: R4
+ in2?.A4,15: $R9^1$
+ in2?.A6: R7
+ in2?.A3: $R4^3$
+ out2!.LANGUAGE_DIGIT: R4
+ out2!.ADDRESS: R4
+ out2!.CLEAR_FORWARD: R10
+ out2!.II7,8,9: R6
+ out2!.I12,13,14: R4

$R4^3$ = out2!.II7,8,9: R6

$R5^1$ = out2!.II7,8,9: R6

R6 = in2?.B6,7: R7
+ in2?.B2,3,4,5,8: $R9^1$
+ out2!.CLEAR_FORWARD: R10

R7 = in2?.ANSWER: R8
+ out2!.CLEAR_FORWARD: R10

R8 = in2?.CLEAR_BACK: R7
+ out2!.CLEAR_FORWARD: R10

$R9^1$ = out2!.CLEAR_FORWARD: R10

R10 = in2?.RELEASE_GUARD: $R_{0,1,2,3}$

## Combining behaviours

■causal relations:

forward:

IA → SEIZING

IA → LANGUAGE_DIGIT_X

IA → DISCRIMINATING_DIGIT

backward:

ADC ← A6

AD ← A6

AN ← ANSWER

CFC ← A4,15

AF ← B6,7

CBK ← CLEAR_BACK {never as a result of an internal action}

SEIZING follows after the reception of the calling party's category CPCI (Fites 9-13+17-19) and this information is included in the IA message as well as the language or discriminating digit

■initial state $S0R_{0,1,2,3}$

$S0R_{0,1,2,3}$ = in1?.IA: $S_{1,2,3}R_{0,1,2,3}$

$S0R4$ = out2!.CLEAR_FORWARD: S0R10

$S0R4^3$ = out2!.II7,8,9: S0R6

{this was requested, so it must be sent before clearing continues at R6}

S0R6 = out2!.CLEAR_FORWARD: S0R10

S0R7 = out2!.CLEAR_FORWARD: S0R10

S0R8 = out2!.CLEAR_FORWARD: S0R10

$S0R9^1$ = out2!.CLEAR_FORWARD: S0R10

S0R10 = in2?.RELEASE_GUARD: $S0R_{0,1,2,3}$

$S_{1,2,3}R_{0,1,2,3}$ = in1?.SA: $S_{1,2,3}R_{0,1,2,3}$
\+ in1?.COT_1: $S4R_{0,1,2,3}$
\+ in1?.CCF: $S_{1,2,3}R_{0,1,2,3}$
\+ in1?.CLF: $S1^1R_{0,1,2,3}$
\+ in1?.COT_2: $S_{4,5,6}R_{0,1,2,3}$
\+ out1!.UM: $S10R_{0,1,2,3}$
\+ in1?.COT_3: $S6R_{0,1,2,3}$
\+ out2!.SEIZING: $S_{1,2,3}R4$

{IA has been received, so R can transit}

$S_{1,2,3}R4$ = in1?.SA: $S_{1,2,3}R4$
\+ in1?.COT_1: S4R4
\+ in1?.COT_2: $S_{4,5,6}R4$
\+ in1?.CCF: $S_{1,2,3}R4$

| | | |
|---|---|---|
| | + | in1?.CLF: $S1^{1}R4$ |
| | + | out1!.UM: S10R4 |
| | + | in1?.COT_3: S6R4 |
| | + | in2?.BACKWARD: $S_{1,2,3}R4$ |
| | + | in2?.A3: $S_{1,2,3}R4^{3}$ |
| | + | in2?.A4,15: $S_{1,2,3}R9^{1}$ |
| | + | in2?.A6: $S_{1,2,3}R7$ |
| | + | out2!.ADDRESS: $S_{1,2,3}R4$ |
| | + | out2!.LANGUAGE_DIGIT: $S_{1,2,3}R4$ |
| | + | out2!.I12,13,14: $S_{1,2,3}R4$ |
| | + | out2!.II7,8,9: $S_{1,2,3}R6$ |
| | + | out2!.CLEAR_FORWARD:$S_{1,2,3}R10$ |
| | | |
| $S_{1,2,3}R4^{3}$ | = | in1?.SA: $S_{1,2,3}R4^{3}$ |
| | + | in1?.COT_1: $S4R4^{3}$ |
| | + | in1?.COT_2: $S_{4,5,6}R4^{3}$ |
| | + | in1?.CCF: $S_{1,2,3}R4^{3}$ |
| | + | in1?.CLF: $S1^{1}R4^{3}$ |
| | + | out1!.UM: $S10R4^{3}$ |
| | + | in1?.COT_3: $S6R4^{3}$ |
| | + | out2!.II7,8,9: $S_{1,2,3}R6$ |
| | | |
| $S_{1,2,3}R6$ | = | in1?.SA: $S_{1,2,3}R6$ |
| | + | in1?.COT_1: S4R6 |
| | + | in1?.COT_2: $S_{4,5,6}R6$ |
| | + | in1?.CCF: $S_{1,2,3}R6$ |
| | + | in1?.CLF: $S1^{1}R6$ |
| | + | out1!.UM: S10R6 |
| | + | in1?.COT_3: S6R6 |
| | + | in2?.B6,7: $S_{1,2,3}R7$ |
| | + | in2?.B2,3,4,5,8: $S_{1,2,3}R9^{1}$ |
| | + | out2!.CLEAR_FORWARD:$S_{1,2,3}R10$ |
| | | |
| $S_{1,2,3}R7$ | = | in1?.SA: $S_{1,2,3}R47$ |
| | + | in1?.COT_1: S4R7 |
| | + | in1?.COT_2: $S_{4,5,6}R7$ |
| | + | in1?.CCF: $S_{1,2,3}R7$ |
| | + | in1?.CLF: $S1^{1}R7$ |
| | + | out1!.UM: S10R7 |
| | + | in1?.COT_3: S6R7 |
| | + | in2?.ANSWER: $S_{1,2,3}R8$ |
| | + | out2!.CLEAR_FORWARD:$S_{1,2,3}R10$ |
| | | |
| $S_{1,2,3}R8$ | = | in1?.SA: $S_{1,2,3}R8$ |
| | + | in1?.COT_1: S4R8 |
| | + | in1?.COT_2: $S_{4,5,6}R8$ |
| | + | in1?.CCF: $S_{1,2,3}R8$ |
| | + | in1?.CLF: $S1^{1}R8$ |
| | + | out1!.UM: S10R8 |
| | + | in1?.COT_3: S6R8 |
| | + | in2?.CLEAR_BACK: $S^{1,2,3}R7$ |
| | + | out2!.CLEAR_FORWARD:$S_{1,2,3}R10$ |
| | | |
| $S_{1,2,3}R9^{1}$ | = | in1?.SA: $S_{1,2,3}R9^{1}$ |
| | + | in1?.COT_1: $S4R9^{1}$ |
| | + | in1?.COT_2: $S_{4,5,6}R9^{1}$ |
| | + | in1?.CCF: $S_{1,2,3}R9^{1}$ |
| | + | in1?.CLF: $S1^{1}R9^{1}$ |
| | + | out1!.UM: $S10R9^{1}$ |
| | + | in1?.COT_3: $S6R9^{1}$ |
| | + | out2!.CLEAR_FORWARD:$S_{1,2,3}R10$ |
| | | |
| $S_{1,2,3}R10$ | = | in1?.SA: $S_{1,2,3}R10$ |
| | + | in1?.COT_1: S4R10 |
| | + | in1?.COT_2: $S_{4,5,6}R10$ |
| | + | in1?.CCF: $S_{1,2,3}R10$ |
| | + | in1?.CLF: $S1^{1}R10$ |
| | + | out1!.UM: S10R10 |
| | + | in1?.COT_3: S6R10 |
| | + | in2?.RELEASE_GUARD: $S_{1,2,3}R_{0,1,2,3}$ |
| | | |
| $S1^{1}R_{0,1,2,3}$ | = | out1!.RLG: $S0R_{0,1,2,3}$ |

{in this case, R: SEIZING will not be sent forward}

$S1'R4$ = out1!.RLG: $S0R4$
+ out2!.CLEAR_FORWARD: $S1'R10$

$S1'R4^3$ = out1!.RLG: $S0R4^3$

$S1'R8$ = out1!.RLG: $S0R8$
+ in2?.CLEAR_BACK: $S1'R7$
+ out2!.CLEAR_FORWARD: $S1'R10$

$S1'R9^1$ = out1!.RLG: $S0R9^1$
+ out2!.CLEAR_FORWARD: $S1'R10$

$S1'R10$ = out1!.RLG: $S0R10$
+ in2?.RLEASE_GUARD: $S1'R_{0,1,2,3}$

$S4R_{0,1,2,3}$ = in1?.SA: $S4R_{0,1,2,3}$
+ in1?.CLF: $S11R_{0,1,2,3}$

$S4R4$ = in1?.SA: $S4R4$
+ in1?.CLF: $S11R4$
+ in2?.BACKWARD: $S4R4$
+ in2?.A4,15: $S4R9^1$
+ in2?.A6: $S4R7$
+ in2?.A3: $S4R4^3$
+ out2!.LANGUAGE_DIGIT: $S4R4$
+ out2!.ADDRESS: $S4R4$
+ out2!.CLEAR_FORWARD: $S4R10$
+ out2!.II7,8,9: $S4R6$
+ out2!.I12,13,14: $S4R4$

$S4R4^3$ = in1?.SA: $S4R4^3$
+ in1?.CLF: $S11R4^3$
+ out2!.II7,8,9: $S4R6$

$S4R6$ = in1?.SA: $S4R6$
+ in1?.CLF: $S11R6$
+ out2!.CLEAR_FORWARD: $S4R10$
+ in2?.B6,7: $S4R7$
+ in2?.B2-5,8: $S4R9^1$

$S4R7$ = in1?.SA: $S4R7$
+ in1?.CLF: $S11R7$
+ in2?.ANSWER: $S4R8$
+ out2!.CLEAR_FORWARD: $S4R10$

$S4R8$ = in1?.SA: $S4R8$
+ in1?.CLF: $S11R8$
+ in2?.CLEAR_BACK: $S4R7$
+ out2!.CLEAR_FORWARD: $S4R10$

$S4R9^1$ = in1?.SA: $S4R9^1$
+ in1?.CLF: $S11R9^1$
+ out2!.CLEAR_FORWARD: $S4R10$

$S4R10$ = in1?.SA: $S4R10$
+ in1?.CLF: S11: $S11R10$
+ in2?.RELEASE_GUARD: $S4R_{0,1,2,3}$

$S_{4,5,6}R_{0,1,2,3}$ = in1?.SA: $S_{4,5,6}R_{0,1,2,3}$
+ in1?.CLF: $S1'R_{0,1,2,3}$
+ out1!.RLG: $S0R_{0,1,2,3}$
+ out1!.UM: $S10R_{0,1,2,3}$
+ out2!.SEIZING: $S_{4,5,6}R4$

$S_{4,5,6}R4$ = in1?.SA: $S_{4,5,6}R4$
+ in1?.CLF: $S1'R4$
+ out1!.RLG: $S0R4$
+ out1!.UM: $S10R4$
+ in2?.BACKWARD: $S_{4,5,6}R4$
+ in2?.A4,15: $S_{4,5,6}R9^1$
+ in2?.A6: $S_{4,5,6}R7$
+ in2?.A3: $S_{4,5,6}R4^3$
+ out2!.LANGUAGE_DIGIT: $S_{4,5,6}R4$

\+ out2!.ADDRESS: $S_{4,5,6}R4$
\+ out2!.CLEAR_FORWARD: $S_{4,5,6}R10$
\+ out2!.II7,8,9: $S_{4,5,6}R6$
\+ out2!.I12,13,14: $S_{4,5,6}R4$

$S_{4,5,6}R4^3$ = in1?.SA: $S_{4,5,6}R4^{34}$
\+ in1?.CLF: $S1^1R4^3$
\+ out1!.RLG: $S0R4^3$
\+ out1!.UM: $S10R4^3$
\+ out1!.ADC: $S_{7,8}R4^3$
\+ out1!.AD: $S8R4^3$
\+ out2!.II7,8,9: $S_{4,5,6}R6$

$S_{4,5,6}R6$ = in1?.SA: $S_{4,5,6}R6$
\+ in1?.CLF: $S1^1R6$
\+ out1!.RLG: S0R6
\+ out1!.UM: S10R6
\+ out1!.ADC: $S_{7,8}R6$
\+ out1!.AD: S8R6
\+ in2?.B6,7: $S_{4,5,6}R7$
\+ in2?.B2-5,8: $S_{4,5,6}R9^1$
\+ out2!.CLEAR_FORWARD: $S_{4,5,6}R10$

$S_{4,5,6}R7$ = in1?.SA: $S_{4,5,6}R7$
\+ in1?.CLF: $S1^1R7$
\+ out1!.RLG: S0R7
\+ out1!.UM: S10R7
\+ out1!.ADC: $S_{7,8}R7$
\+ out1!.AD: S8R7
\+ out1!.AF: S9R7
\+ in2?.ANSWER: $S_{4,5,6}R8$
\+ out2!.CLEAR_FORWARD: $S_{4,5,6}R10$

$S_{4,5,6}R8$ = in1?.SA: $S_{4,5,6}R8$
\+ in1?.CLF: $S1^1R8$
\+ out1!.RLG: S0R8
\+ out1!.UM: S10R8
\+ out1!.ADC: $S_{7,8}R8$
\+ out1!.AD: S8R8
\+ out1!.AF: S9R8
\+ in2?.CLEAR_BACK: $S_{4,5,6}R7$
\+ out2!.CLEAR_FORWARD: $S_{4,5,6}R10$

$S_{4,5,6}R9^1$ = in1?.SA: $S_{4,5,6}R9^1$
\+ in1?.CLF: $S1^1R9^1$
\+ out1!.RLG: $S0R9^1$
\+ out1!.UM: $S10R9^1$
\+ out1!.ADC: $S_{7,8}R9^1$
\+ out1!.AD: $S8R9^1$
\+ out1!.AF: $S9R9^1$
\+ out2!.CLEAR_FORWARD: $S_{4,5,6}R9^1$

$S_{4,5,6}R10$ = in1?.SA: $S_{4,5,6}R10$
\+ in1?.CLF: $S1^1R10$
\+ out1!.RLG: S0R10
\+ out1!.UM: S10R10
\+ out1!.ADC: $S_{7,8}R10$
\+ out1!.AD: S8R10
\+ out1!.AF: S9R10
\+ in2?.RELEASE_GUARD: $S_{4,5,6}R0$

$S6R_{0,1,2,3}$ = in1?.SA: $S6R_{0,1,2,3}$
\+ in1?.CLF: $S1^1R_{0,1,2,3}$
\+ out1!.RLG: $S0R_{0,1,2,3}$
\+ out1!.ADI: $S10R_{0,1,2,3}$
\+ out2!.SEIZING: S6R4

S6R4 = in1?.SA: S6R4
\+ in1?.CLF: $S1^1R4$

+ out1!.RLG: S0R4
+ out1!.ADI: S10R4
+ in2?.BACKWARD: S6R4
+ in2?.A4,15: $S6R9^1$
+ in2?.A6: S6R7
+ in2?.A3: $S6R4^3$
+ out2!.LANGUAGE_DIGIT: S6R4
+ out2!.ADDRESS: S6R4
+ out2!.CLEAR_FORWARD: S6R10
+ out2!.II7,8,9: S6R6
+ out2!.I12,13,14: S6R4

$S6R4^3$ = in1?.SA: $S6R4^3$
+ out1!.ADC: $S_{7,8}R4^3$

{ADC after A6}
+ out1!.AD: $S8R4^3$

{AD after A6}
+ in1?.CLF: $S1^1R4^3$
+ out1!.RLG: $S0R4^3$
+ out1!.ADI: $S10R4^3$
+ out2!.II7,8,9: S6R6

S6R6 = in1?.SA_2: S6R6
+ out1!.ADC: $S_{7,8}$
+ out1!.AD: S8R6
+ in1?.CLF: $S1^1R6$
+ out1!.RLG: S0R6
+ out1!.ADI: S10R6
+ out2!.CLEAR_FORWARD: S6R10
+ in2?.B6,7: S6R7
+ in2?.B2-5,8: $S6R9^1$

S6R7 = in1?.SA_2: S6R7
+ out1!.ADC: $S_{7,8}R7$
+ out1!.AD: S8R7
+ out1!.AF: S9R7

{AF may follow now because B6,7 may have been received}
+ in1?.CLF: $S1^1R7$
+ out1!.RLG: S0R7
+ out1!.ADI: S10R7
+ in2?.ANSWER: S6R8
+ out2!.CLEAR_FORWARD: S6R10

S6R8 = in1?.SA_2: S6R8
+ out1!.ADC: $S_{7,8}R8$
+ out1!.AD: S8R8
+ out1!.AF: S9R8
+ in1?.CLF: $S1^1R8$
+ out1!.RLG: S0R8
+ out1!.ADI: S10R8
+ in2?.CLEAR_BACK: S6R7
+ out2!.CLEAR_FORWARD: S6R10

$S6R9^1$ = in1?.SA_2: $S6R9^1$
+ out1!.ADC: $S_{7,8}R9^1$
+ out1!.AD: $S8R9^1$
+ out1!.AF: $S9R9^1$
+ in1?.CLF: $S1^1R9^1$
+ out1!.RLG: $S0R9^1$
+ out1!.ADI: $S10R9^1$
+ out2!.CLEAR_FORWARD: S6R10

S6R10 = in1?.SA_2: S6R10
+ out1!.ADC: $S_{7,8}R10$
+ out1!.AD: S8R10
+ out1!.AF: S9R10
+ in1?.CLF: $S1^1R10$
+ out1!.RLG: S0R10
+ out1!.ADI: S10R10
+ in2?.RELEASE_GUARD: $S6R_{0,1,2,3}$

$S_{7,8}R4^3$ = out1!.CFC: $S10R4^3$

{CFC follows after A4,15}

+ in1?.CLF: $S1^1R4^3$
+ in1?.FOT: $S8R4^3$
+ out2!.II7,8,9: $S_{7,8}R6$

$S_{7,8}R6$ = out1!.CFC: S10R6
+ in1?.CLF: $S1^1R6$
+ in1?.FOT: S8R6
+ in2?.B6,7: $S_{7,8}R7$
+ in2?.B2-5,8: $S_{7,8}R9^1$
+ out2!.CLEAR_FORWARD: $S_{7,8}R10$

$S_{7,8}R7$ = out1!.AN: S11R7
{AN after ANSWER}
+ out1!.CFC: S10R7
+ in1?.CLF: $S1^1R7$
+ in1?.FOT: S8R7
+ in2?.ANSWER: $S_{7,8}R8$
+ out2!.CLEAR_FORWARD: $S_{7,8}R10$

$S_{7,8}R8$ = out1!.AN: S11R8
+ out1!.CFC: S10R8
+ in1?.CLF: $S1^1R8$
+ in1?.FOT: S8R8
+ in2?.CLEAR_BACK: $S_{7,8}R7$
+ out2!.CLEAR_FORWARD: $S_{7,8}R10$

$S_{7,8}R9^1$ = out1!.AN: $S11R9^1$
+ out1!.CFC: $S10R9^1$
+ in1?.CLF: $S1^1R9^1$
+ in1?.FOT: $S8R9^1$
+ out2!.CLEAR_FORWARD: $S_{7,8}R10$

$S_{7,8}R10$ = out1!.AN: S11R10
+ out1!.CFC: S10R10
+ in1?.CLF: $S1^1R10$
+ in1?.FOT: S8R10
+ in2?.RELEASE_GUARD: $S_{7,8}R_{0,1,2,3}$

$S8R4^3$ = out1!.UM: $S10R4^3$
+ in1?.CLF: $S1^1R4^3$
+ in1?.FOT: $S8R4^3$
+ out2!.II7,8,9: S8R6

S8R6 = out1!.UM: S10R6
+ in1?.CLF: $S1^1R6$
+ in1?.FOT: S8R6
+ in2?.B6,7: S8R7
+ in2?.B2-5,8:$S8R9^1$
+ out2!.CLEAR_FORWARD: S8R10

S8R7 = out1!.UM: S10R7
+ in1?.CLF: $S1^1R7$
+ in1?.FOT: S8R7
+ in2?.ANSWER: S8R8
+ out2!.CLEAR_FORWARD: S8R10

S8R8 = out1!.AN: S11R8
{AN ← ANSWER}
+ out1!.UM: S10R8
+ in1?.CLF: $S1^1R8$
+ in1?.FOT: S8R8
+ in2?.CLEAR_BACK: S8R7
+ out2!.CLEAR_FORWARD: S8R10

$S8R9^1$ = out1!.AN: $S11R9^1$
+ out1!.UM: $S10R9^1$
+ in1?.CLF: $S1^1R9^1$
+ in1?.FOT: $S8R9^1$
+ out2!.CLEAR_FORWARD: S8R10

S8R10 = out1!.AN: S11R10
+ out1!.UM: S10R10
+ in1?.CLF: $S1^1R10$
+ in1?.FOT: S8R10
+ in2?.RELEASE_GUARD: $S8R_{0,1,2,3}$

S9R7 = out1!.CFL: S10R7
+ in1?.CLF: S1'R7
+ in1?.FOT: S9R7
+ in2?.in2?.ANSWER: S9R8

S9R8 = out1!.ANC: S11R8
{ANC ← ANSWER}
+ out1!.CFL: S10R8
+ in1?.CLF: S1'R8
+ in1?.FOT: S9R8
+ in2?.CLEAR_BACK: S9R7
+ out2!.CLEAR_FORWARD: S9R10

$S9R9^1$ = out1!.ANC: $S11R9^1$
+ out1!.CFL: $S10R9^1$
+ in1?.CLF: $S1'R9^1$
+ in1?.FOT: $S9R9^1$
+ out2!.CLEAR_FORWARD: $S9R9^1$

S9R10 = out1!.ANC: S11R10
+ out1!.CFL: S10R10
+ in1?.CLF: S1'R10
+ in1?.FOT: S9R10
+ in2?.RELEASE_GUARD: $S9R_{0,1,2,3}$

$S10R_{0,1,2,3}$ = in1?.CLF:$S1'R_{0,1,2,3}$
+ out1!.CLF: $S13R_{0,1,2,3}$

S10R4 = out2!.CLEAR_FORWARD: S10R10
+ in1?.CLF: S1'R4
+ out1!.CLF: S13R4

$S10R4^3$ = out2!.II7,8,9: S10R6
+ in1?.CLF: $S1'R4^3$
+ out1!.CLF: $S13R4^3$

S10R6 = out2!.CLEAR_FORWARD: S10R10
+ in1?.CLF:S1'R6
+ out1!.CLF: S13R6

S10R7 = out2!.CLEAR_FORWARD: S10R10
+ in1?.CLF:S1'R7
+ out1!.CLF: S13R7

S10R8 = in2?.CLEAR_BACK: S10R7
+ out2!.CLEAR_FORWARD: S10R10
+ in1?.CLF:S1'R10
+ out1!.CLF: S13R10

$S10R9^1$ = out2!.CLEAR_FORWARD: S10R10
+ in1?.CLF: $S1'R9^1$
+ out1!.CLF: $S13R9^1$

S10R10 = in2?.RELEASE_GUARD: $R_{0,1,2,3}$
+ in1?.CLF:S1'R10
+ out1!.CLF: S13R10

$S11R_{0,1,2,3}$ = out2!.SEIZING: S11R4
+ out1!.CBK: $S12R_{0,1,2,3}$
{in case a CLEAR_BACK has been received and R is back in the idle state}
+ in1?.CLF: $S1'R_{0,1,2,3}$
+ in1?.FOT: $S11R_{0,1,2,3}$

S11R4 = out2!.CLEAR_FORWARD: S11R10
+ in1?.CLF: S1'R4
+ in1?.FOT: S11R4

$S11R4^3$ = out2!.II7,8,9: S11R6

{R must transit to R6 before clearing can be carried out}

\+ in1?.CLF: $S1^1R4^3$

\+ in1?.FOT: $S11R4^3$

S11R6 = out2!.CLEAR_FORWARD: S11R10

\+ in1?.CLF: $S1^1R6$

\+ in1?.FOT: S11R6

S11R7 = out2!.CLEAR_FORWARD: S11R10

\+ in1?.CLF: $S1^1R7$

\+ in1?.FOT: S11R7

\+ out1!.CBK: S12R7

S11R8 = out2!.CLEAR_FORWARD: S11R10

= out1!.CBK: S12R8

\+ in1?.CLF: $S1^1R8$

\+ in1?.FOT: S11R8

$S11R9^1$ = out2!.CLEAR_FORWARD: S11R10

\+ out1!.CBK: $S12R9^1$

\+ in1?.CLF: $S1^1R9^1$

\+ in1?.FOT: $S11R9^1$

S11R10 = in2?.RELEASE_GUARD: $S11R_{0,1,2,3}$

\+ out1!.CBK: S12R10

\+ in1?.CLF: $S1^1R10$

\+ in1?.FOT: S11R10

S12R7 = in2?.ANSWER: S12R8

\+ out2!.CLEAR_FORWARD: S12R10

\+ out1!.RAN: S11R7

\+ in1?.CLF: $S1^1R7$

\+ in1?.FOT: S12R7

S12R8 = in2?.CLEAR_BACK: S12R7

\+ out2!.CLEAR_FORWARD: S12R10

\+ out1!.RAN: S11R8

\+ in1?.CLF: $S1^1R8$

\+ in1?.FOT: S12R8

$S12R9^1$ = out2!.CLEAR_FORWARD: S12R10

\+ out1!.RAN: $S11R9^1$

\+ in1?.CLF: $S1^1R9^1$

\+ in1?.FOT: $S12R9^1$

S12R10 = in2?.RELEASE_GUARD: $S12R_{0,1,2,3}$

\+ out1!.RAN: S11R10

\+ in1?.CLF: $S1^1R10$

\+ in1?.FOT: S12R10

S13R4 = out2!.CLEAR_FORWARD: S13R10

\+ in1?.CLF: $S1^1R4$

\+ out1!.CFL: S13R4

$S13R4^3$ = out2!.II7,8,9: S13R6

\+ in1?.CLF: $S1^1R4^3$

\+ out1!.CFL: $S13R4^3$

S13R6 = out2!.CLEAR_FORWARD: S13R10

\+ in1?.CLF: $S1^1R6$

\+ out1!.CFL: S13R6

S13R7 = in2?.ANSWER: S13R8

\+ out2!.CLEAR_FORWARD: S13R10

\+ in1?.CLF: S1'R7

\+ out1!.CFL: S13R7

S13R8 = in2?.CLEAR_BACK: S13R7

\+ out2!.CLEAR_FORWARD: S13R10

\+ in1?.CLF: S1'R8

\+ out1!.CFL: S13R8

S13R9' = out2!.CLEAR_FORWARD: S13R10

\+ in1?.CLF: S1'R9'

\+ out1!.CFL: S13R9'

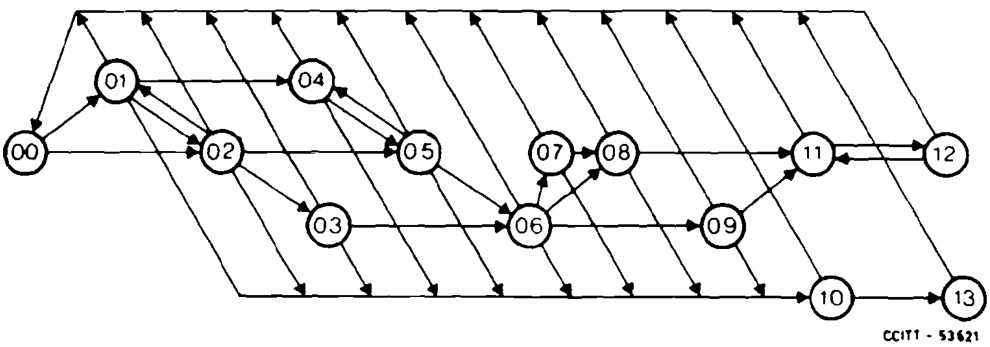
S13R10 = in2?.RELEASE_GUARD: $S13R_{0,1,2,3}$

\+ in1?.CLF: S1'R10

\+ out1!.CFL: S13R10

# Appendix C Incoming #7, outgoing R2 and interworking SDL procedures

## LOGIC PROCEDURES FOR INCOMING SIGNALLING SYSTEM No. 7 (TUP)




| *State number* | *State description* | *Sheet reference* | *Timers running* |
|---|---|---|---|
| 00 | Idle | 1, 13 | |
| 01 | Wait for further digits | 2 | $t_1$, $t_2$ |
| 02 | Wait for digit analysis | 4 | $t_1$, $t_2$ or $t_3$ |
| 03 | Wait for continuity check (COT) | 7 | $t_1$, $t_2$ or $t_3$ |
| 04 | Wait for further digits (COT received) | 3 | $t_2$ |
| 05 | Wait for digit analysis (COT received) | 3 | $t_2$ or $t_3$ |
| 06 | Wait for address complete (COT received) | 9 | $t_2$ or $t_3$ |
| 07 | Wait for register deactivation | 10 | |
| 08 | Address complete – wait for answer | 12 | |
| 09 | Address complete, subscriber free – wait for answer | 13 | |
| 10 | Call unsuccessful – wait for clear-forward | 11 | $t_4$ |
| 11 | Answered | 13 | |
| 12 | Clear-back | 13 | |
| 13 | Call failure wait for clear forward | 11 | |

FIGURE 1/Q.614

**State overview diagram for incoming Signalling System No. 7 (TUP)**

*Supervisory timers for incoming Signalling System No. 7 (TUP)*

| | | | | | |
|---|---|---|---|---|---|
| $t_1$ = 10 - 15 s | Recommendation Q.724, § 6.4.2, a) | $t_4$ = | 4 - 15 s | Recommendation Q.724, § 6.4.2, b) |
| $t_2$ = 15 - 20 s | Recommendation Q.724, § 1.7 | $t_5$ = | 1 min | Recommendation Q.724, § 6.4.2, b) |
| $t_3$ = 20 - 30 s | Recommendation Q.724, § 6.4.3 | | | |

*Procedures not shown*

The following procedures, not directly relevant to interworking, are not shown in the logic procedures:

- dual seizure,
- blocking and unblocking sequences,
- user part selection (see Note),
- confusion and message refusal signals,
- reset circuit procedures,
- test call procedures,
- out of service,
- national procedures.

*Signal abbreviations used*

The signal abbreviations used correspond to those of the Signalling System No. 7 specifications unless otherwise indicated on the same sheet.

The signal abbreviations used are listed below with their meanings:

| | |
|---|---|
| ACM | Address complete message |
| ADC | Address complete, charge |
| ADI | Address incomplete |
| ADN | Address complete, no charge |
| ADX | Address complete, coin box |
| AFC | Address complete, subscriber free, charge |
| AFN | Address complete, subscriber free, no charge |
| AFX | Address complete, subscriber free, coin box |
| ANC | Answer charge |
| ANN | Answer no charge |
| CBK | Clear-back signal |
| CCF | Continuity-failure signal |
| CCH | Continuity check indicator |
| CFL | Call failure |
| CGC | Circuit group congestion |
| CLF | Clear-forward signal |
| COT | Continuity |
| CPCI | Calling party category indicator |
| ESI | Echo suppressor indicator |
| FOT | Forward-transfer signal |
| IAM | Initial address message |
| LOS | Line-out-of-service |
| NCI | Nature of circuit indicator |
| NNC | National network congestion |
| NAI | Nature of address indicator |
| RAN | Reanswer signal |
| RLG | Release-guard signal |
| SAM | Subsequent address message |
| SAO | Subsequent address message with one address digit |
| SEC | Switching equipment congestion |
| SSB | Subscriber busy |
| SST | Send special information tone |
| UNN | Unallocated number |

*Note* – This SDL diagram relates only to the International Telephony User Part (TUP) specified for Signalling System No. 7 in Recommendations Q.721-Q.725. The selection of the TUP is assumed to have been made on a per message basis by the Level 3 message distribution process.

FIGURE 2/Q.614

**Notes to incoming Signalling System No. 7 (TUP)**

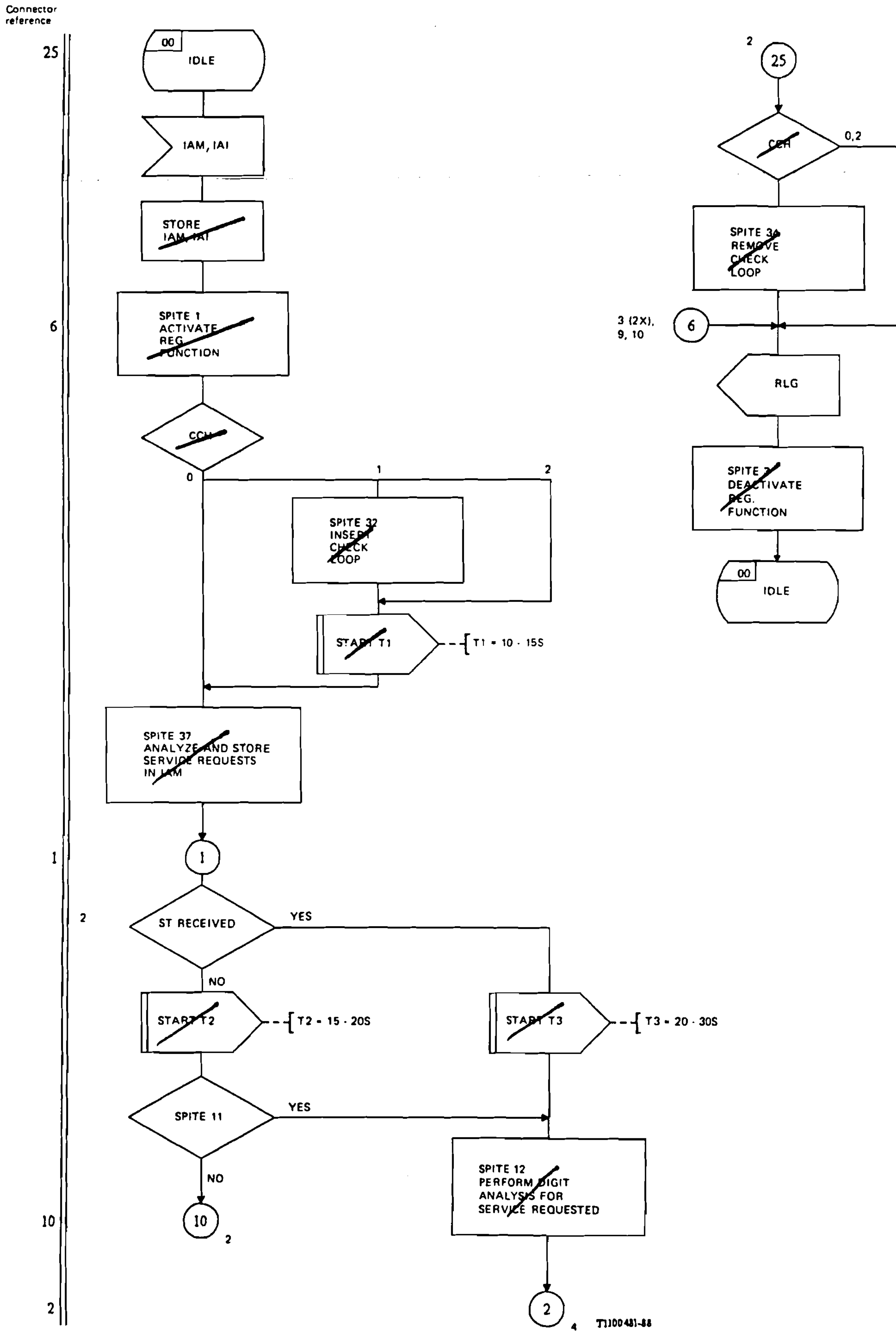



FIGURE 3/Q.614 (Sheet 1 of 14)

**Incoming Signalling System No. 7 (TUP)**

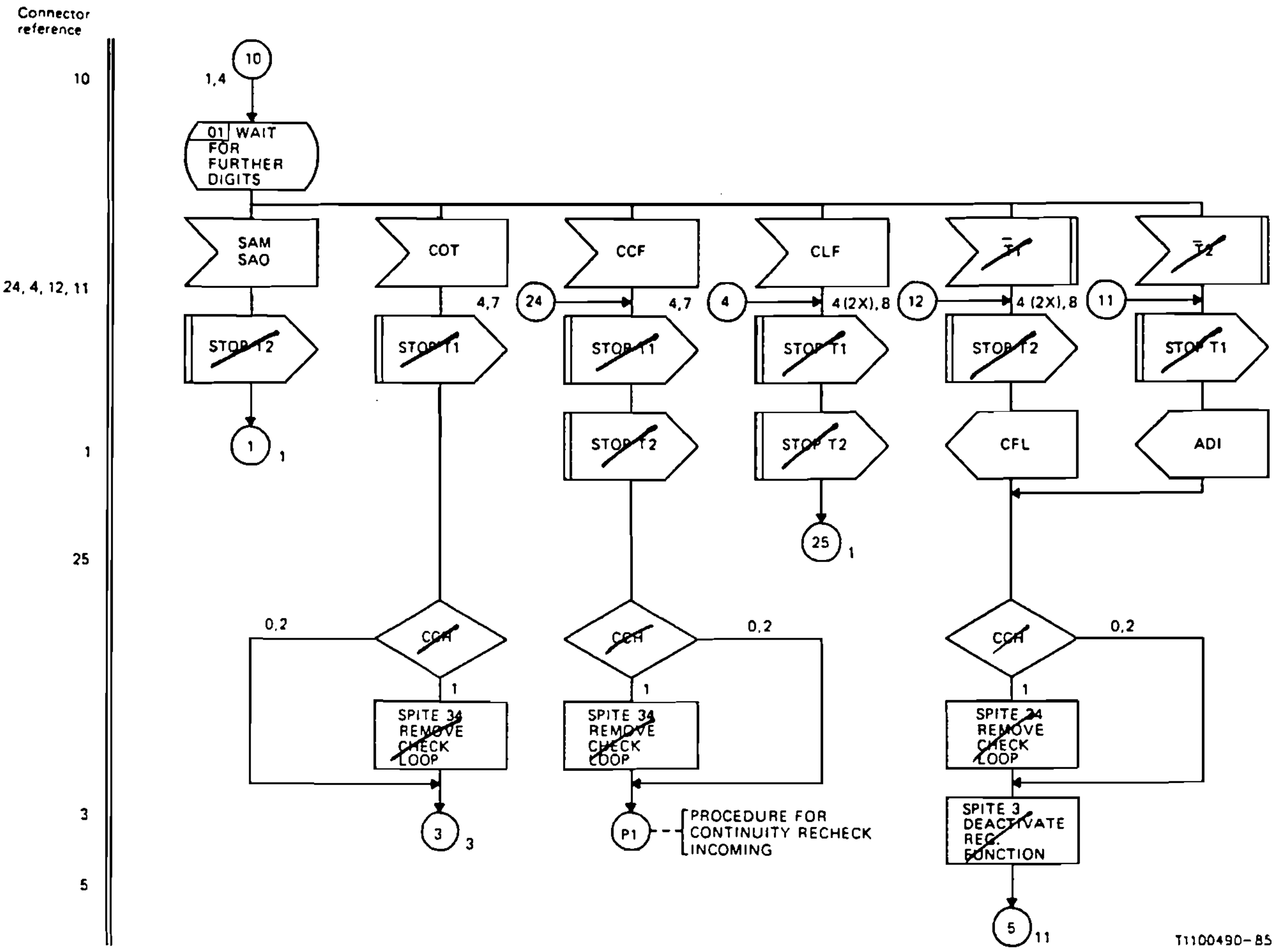



FIGURE 3/Q.614 (Sheet 2 of 14)

**Incoming Signalling System No. 7 (TUP)**

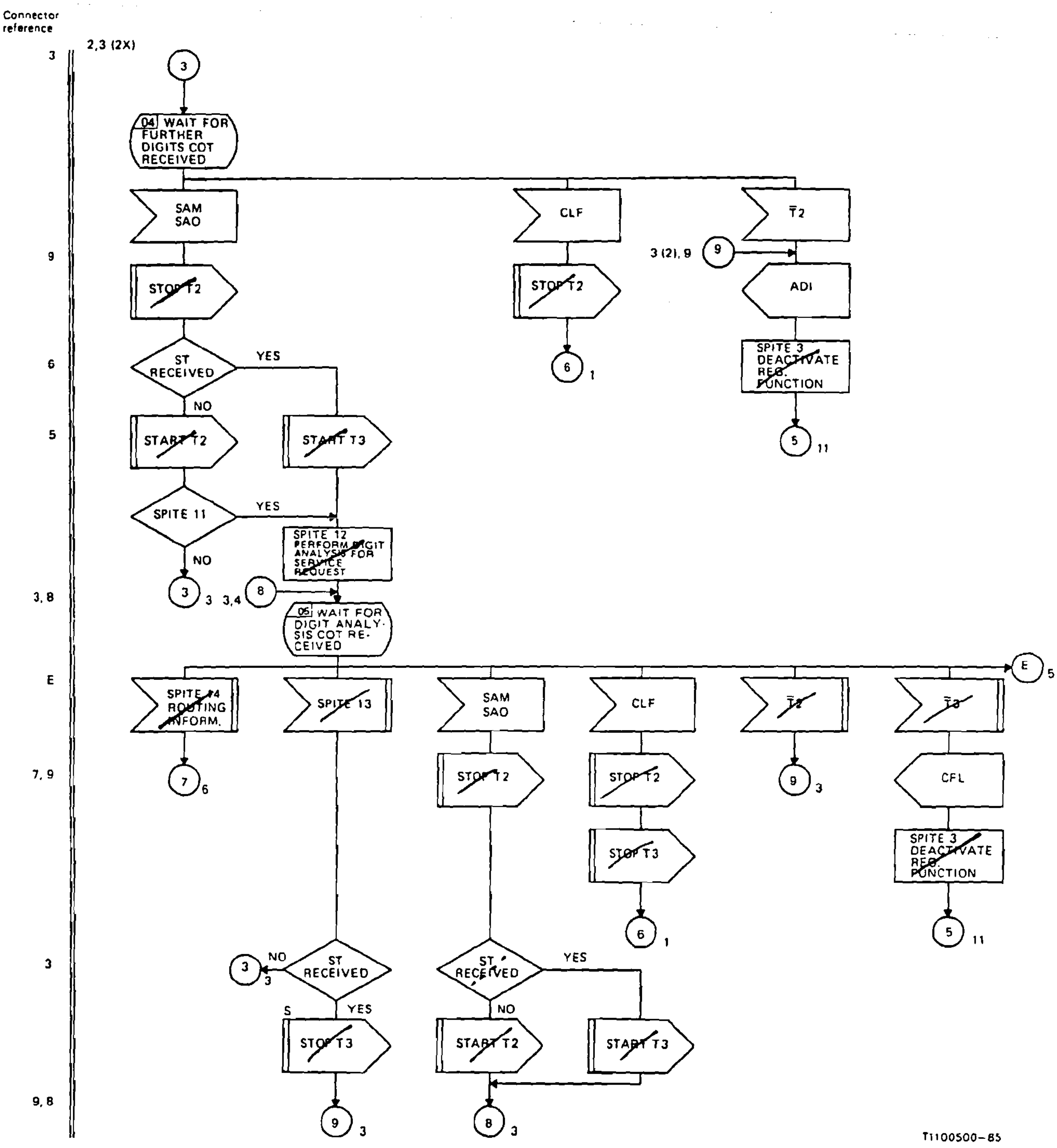
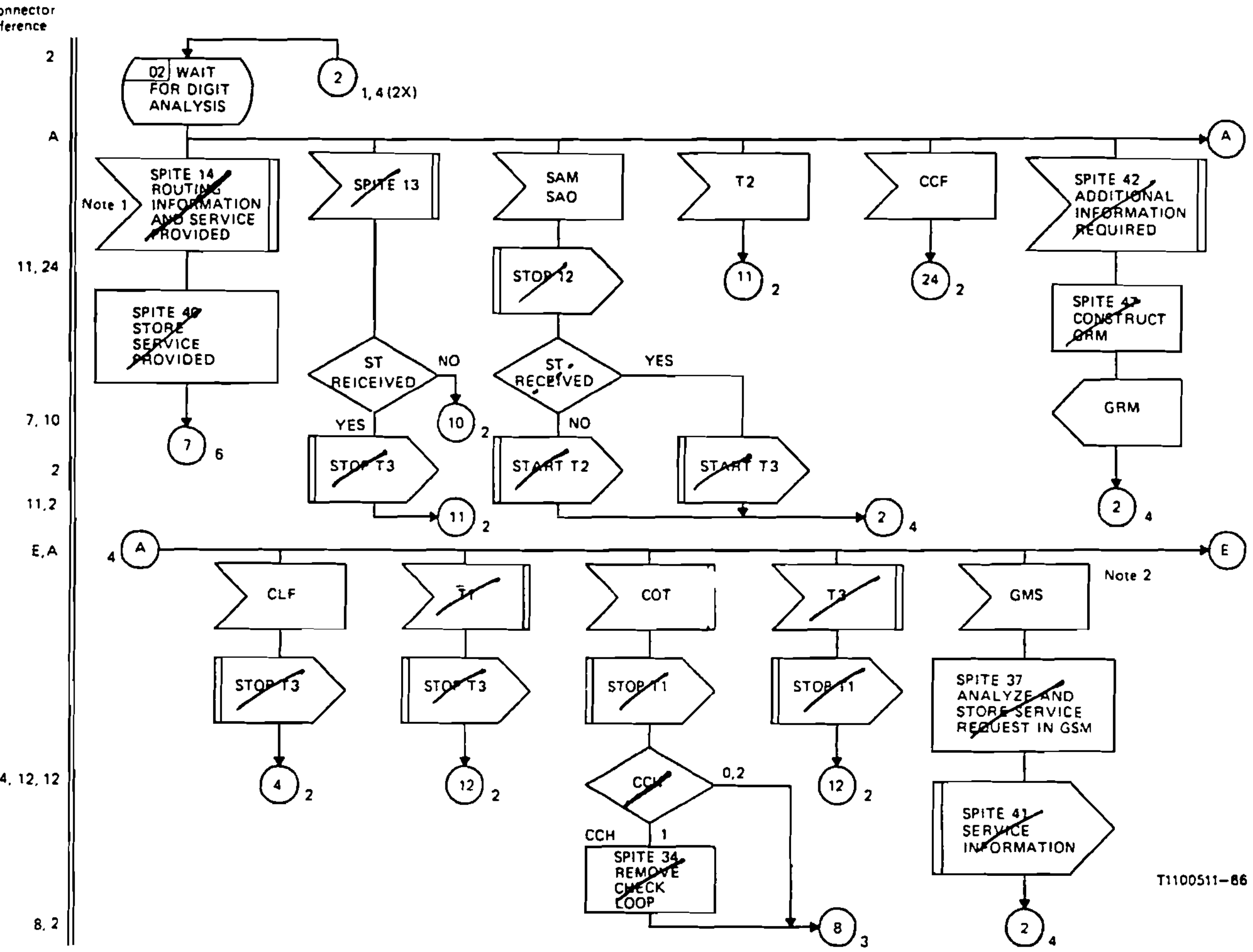



FIGURE 3/Q.614 (Sheet 3 of 14)

**Incoming Signalling System No. 7 (TUP)**

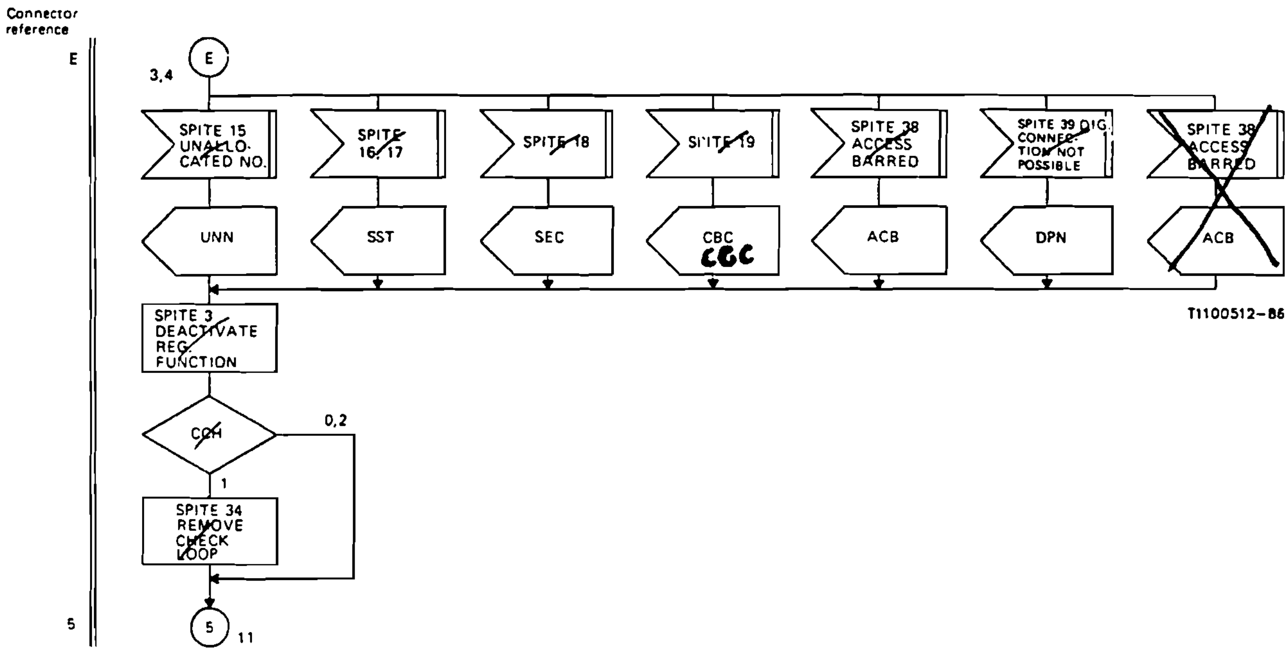



*Note 1* – It is assumed that if a transit exchange has requested additional information then routing information will not be received until after the requested information has been received.

*Note 2* – It is assumed that GRQ was sent.

FIGURE 3/Q.614 (Sheet 4 of 14)

**Incoming Signalling System No. 7 (TUP)**




*Note* – It is assumed that if a transit exchange has requested additional information then routing information will not be received until after the requested information has been received.

FIGURE 3/Q.614 (Sheet 5 of 14)

**Incoming Signalling System No. 7 (TUP)**

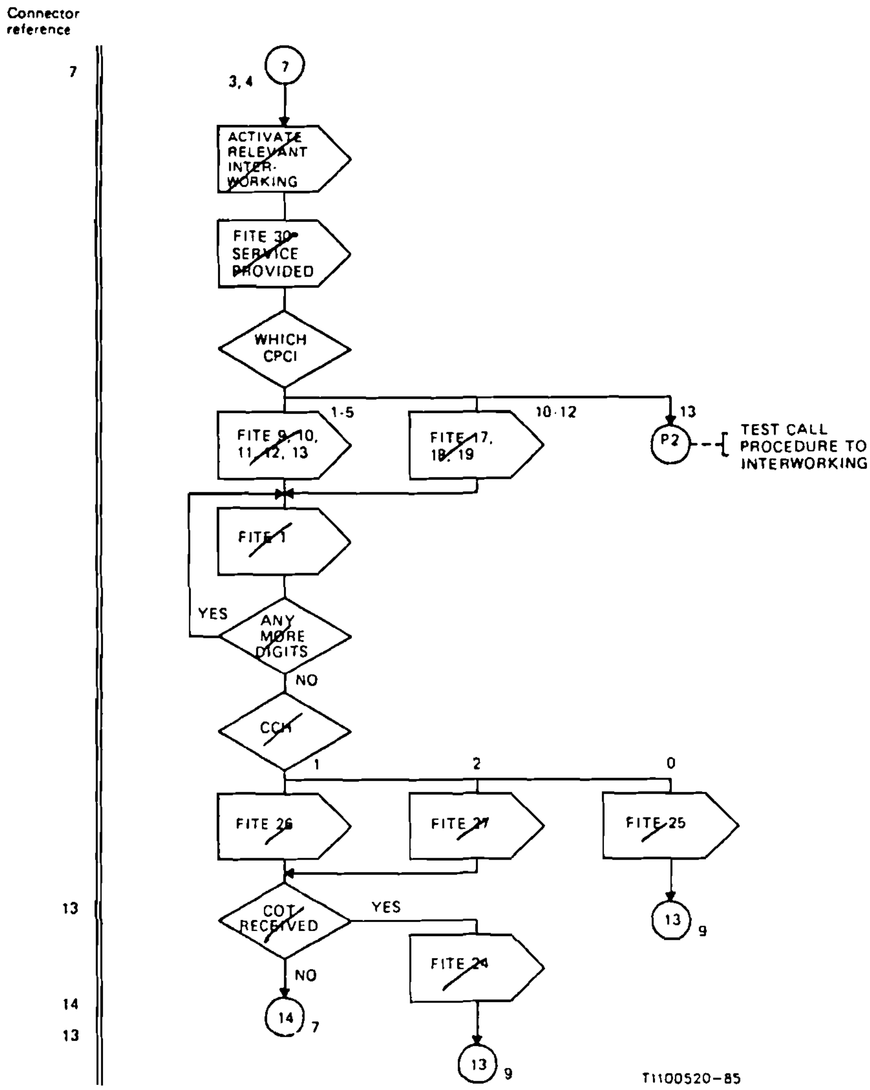
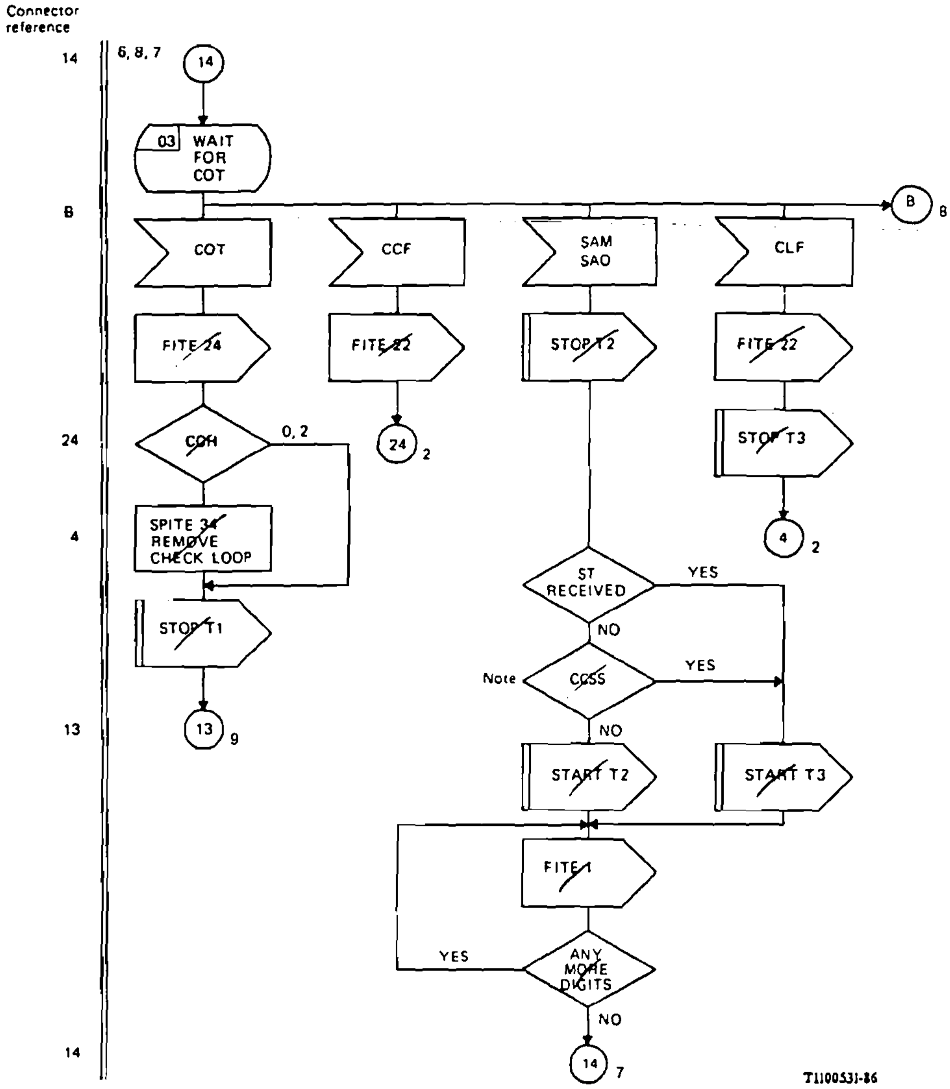



FIGURE 3/Q.614 (Sheet 6 of 14)

**Incoming Signalling System No. 7 (TUP)**




*Note* – Is outgoing link common channel signalling system?

FIGURE 3/Q.614 (Sheet 7 of 14)

**Incoming Signalling System No. 7 (TUP)**

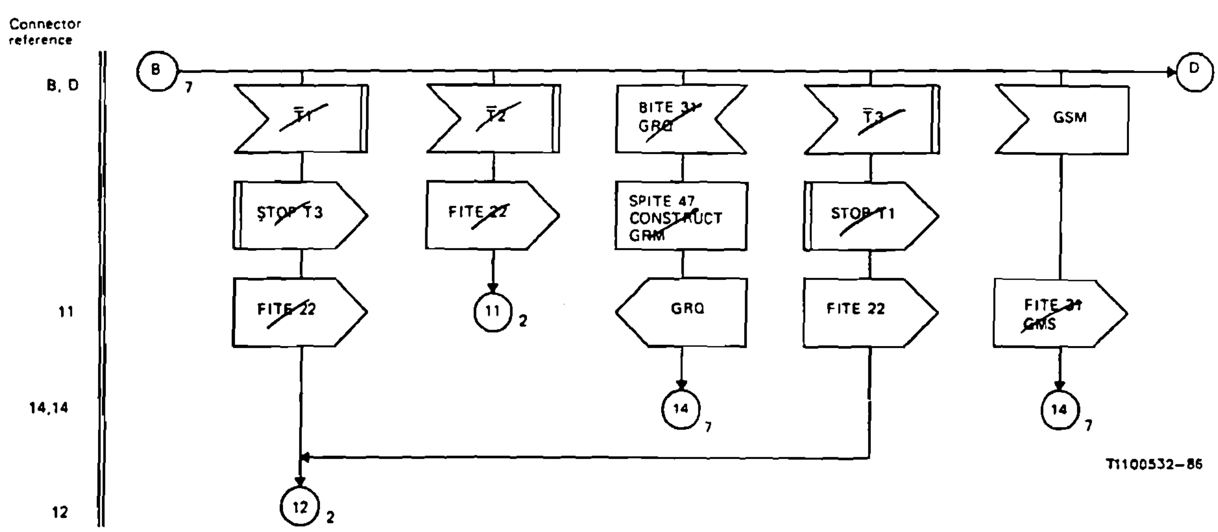
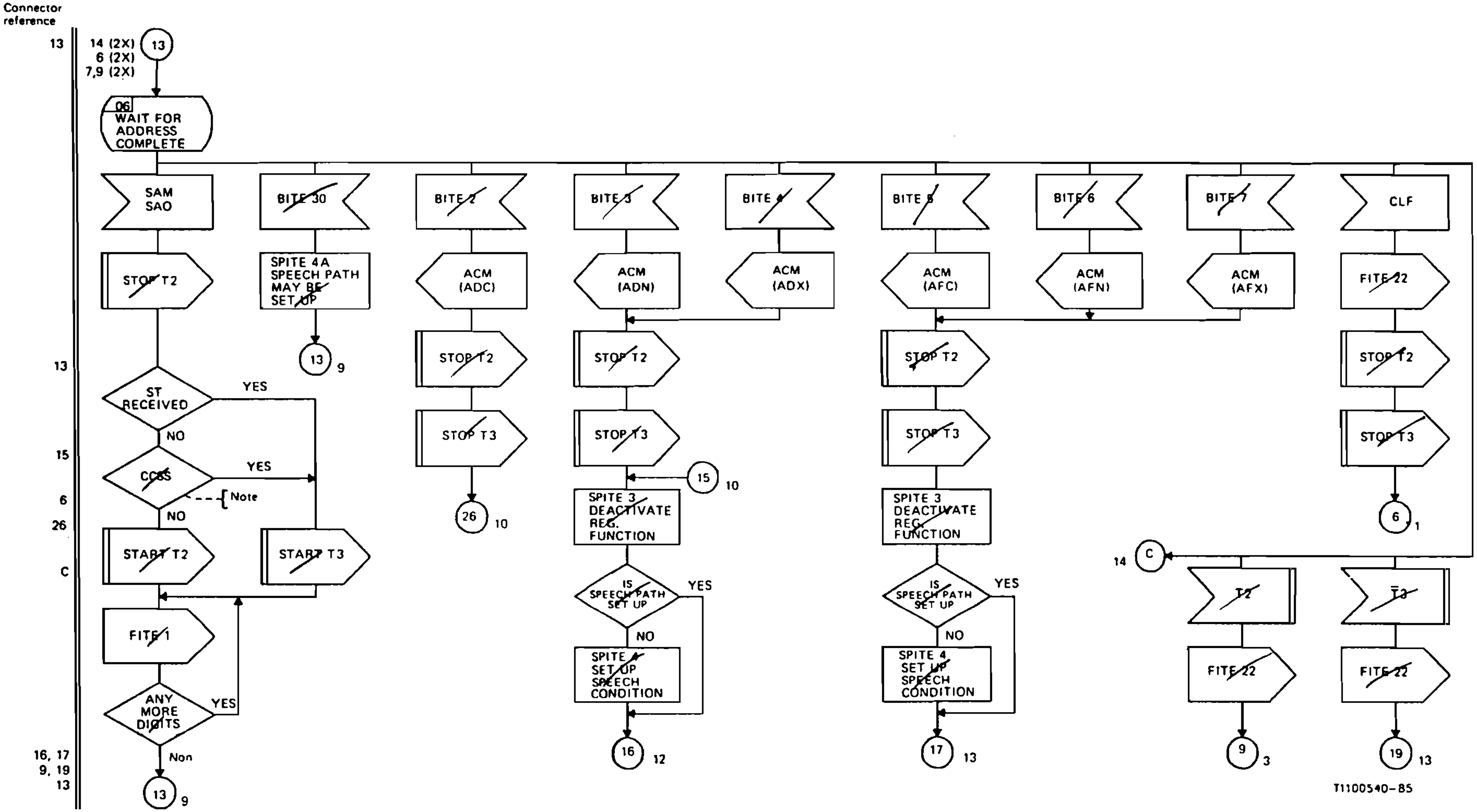



FIGURE 3/Q.614 (Sheet 8 of 14)

**Incoming Signalling System No. 7 (TUP)**




*Note* – Is outgoing link common channel signalling system?

FIGURE 3/Q.614 (Sheet 9 of 14)

**Incoming Signalling System No. 7 (TUP)**

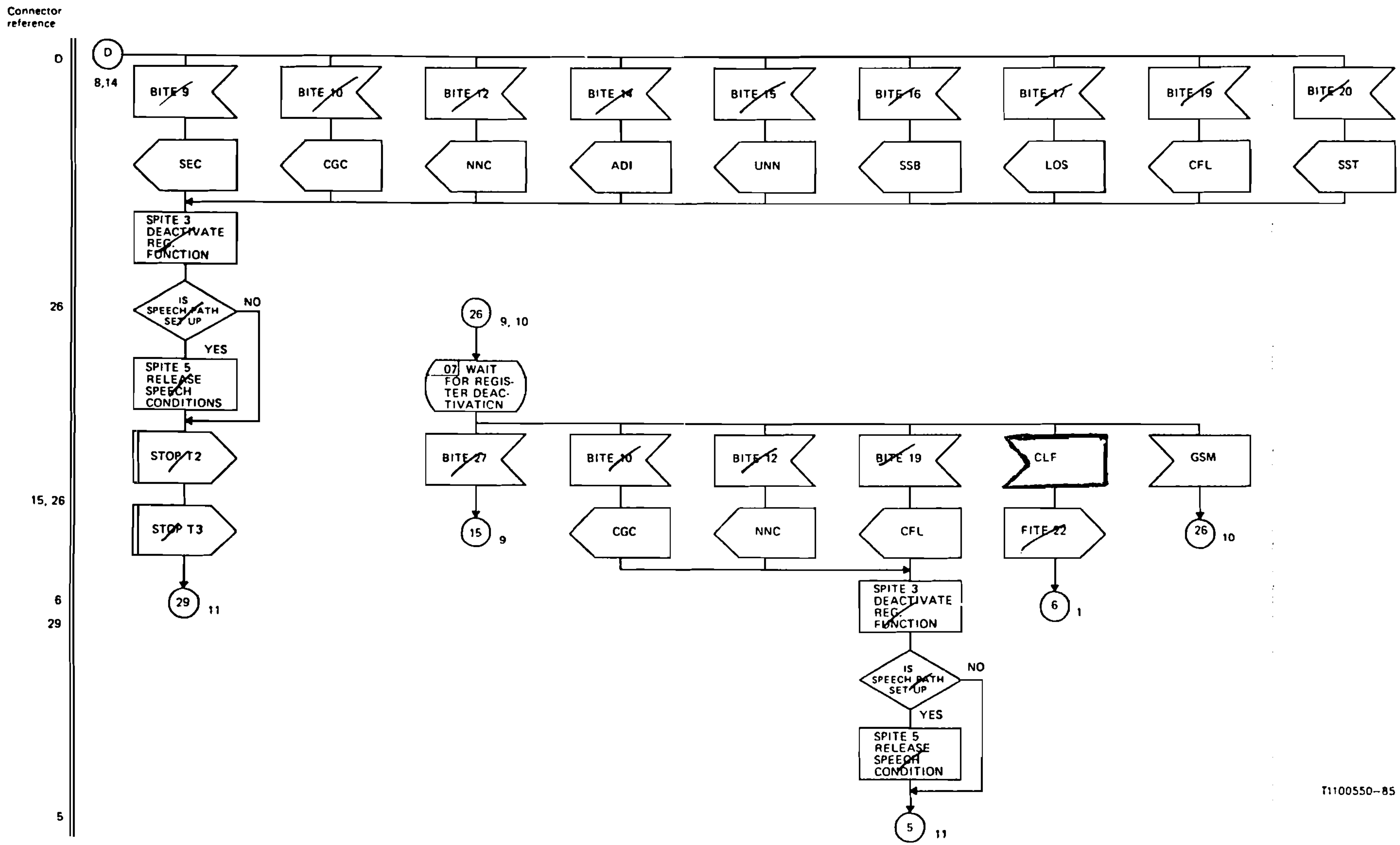



FIGURE 3/Q.614 (Sheet 10 of 14)

**Incoming Signalling System No. 7 (TUP)**

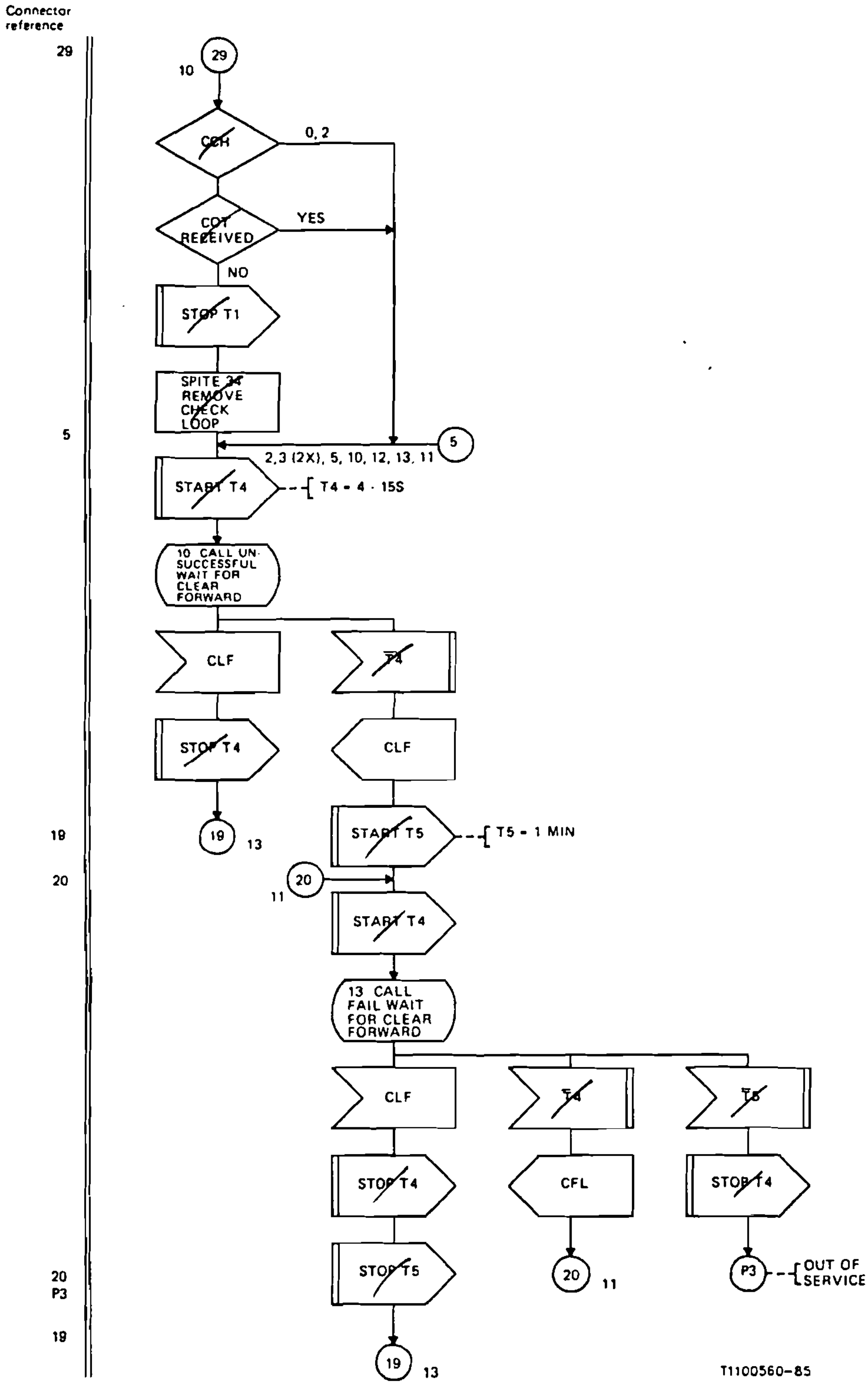



FIGURE 3/Q.614 (Sheet 11 of 14)

**Incoming Signalling System No. 7 (TUP)**

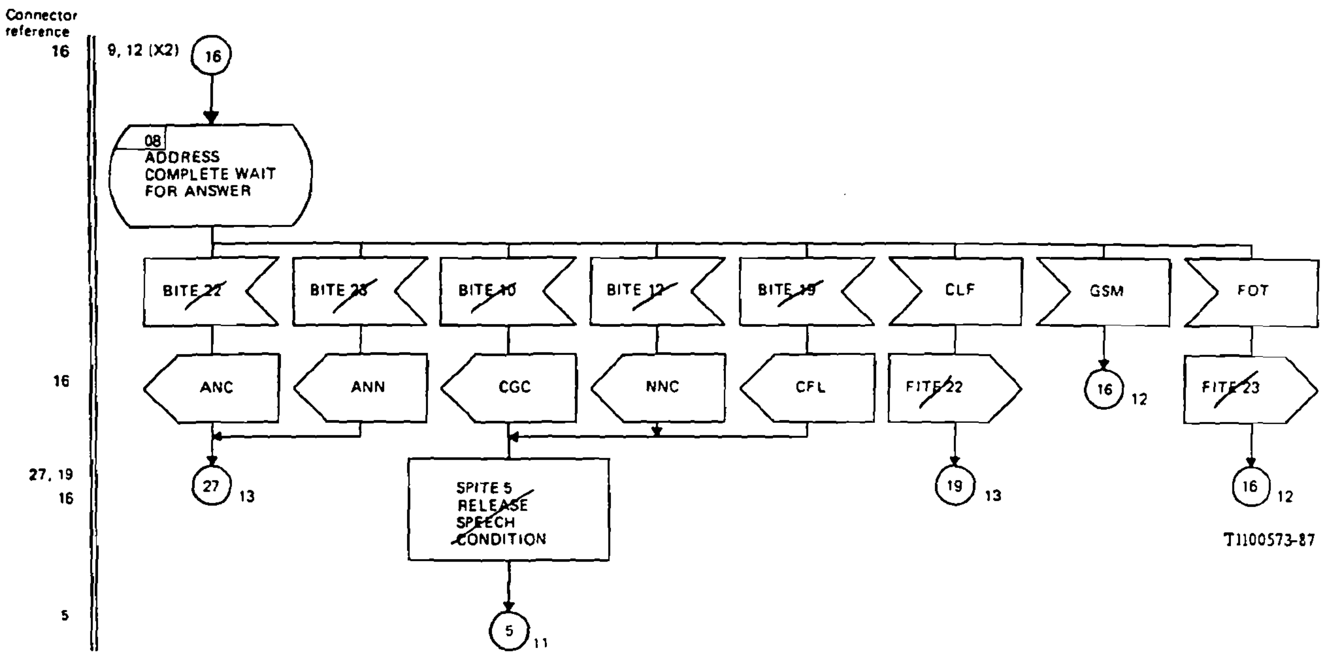



FIGURE 3/Q.614 (Sheet 12 of 14)

**Incoming Signalling System No. 7 (TUP)**

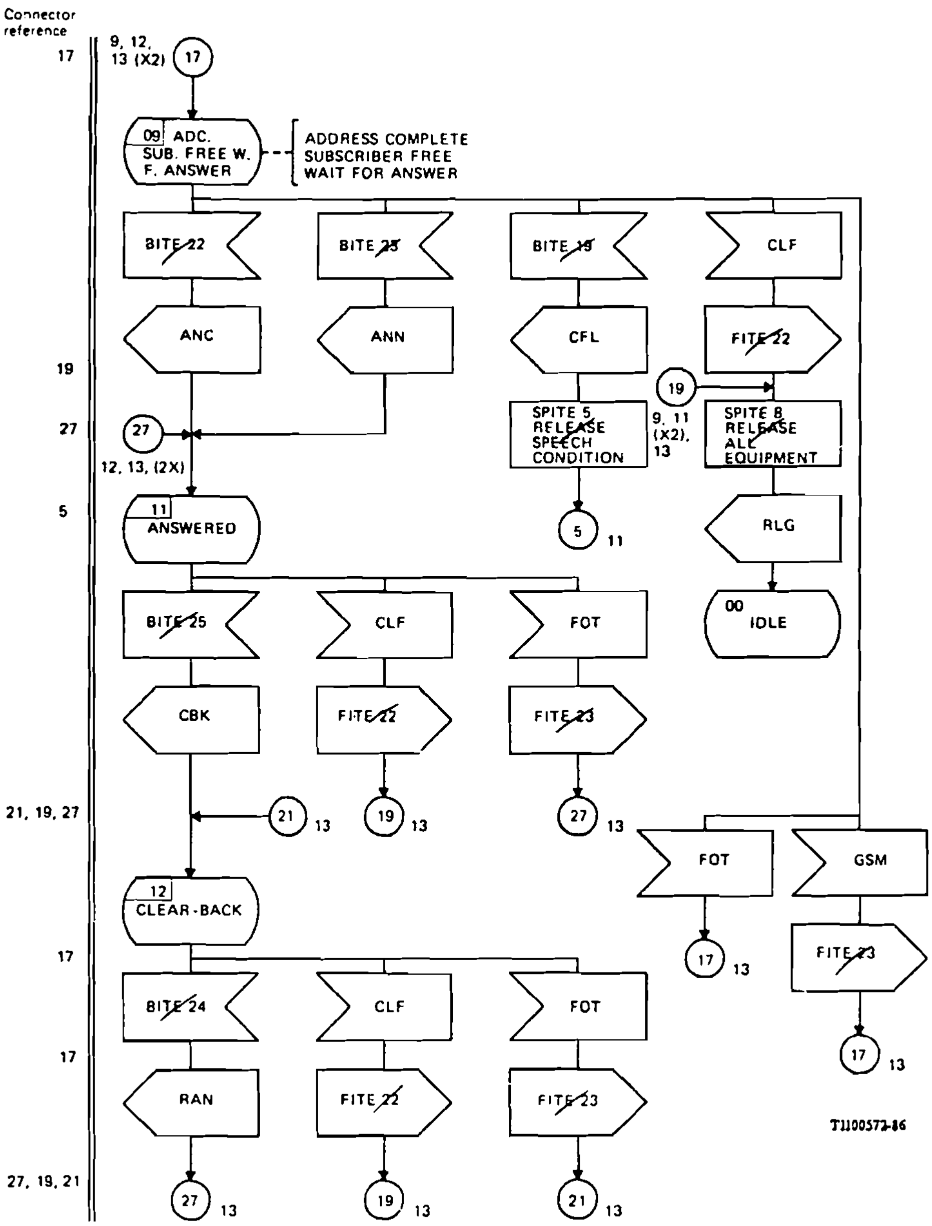



FIGURE 3/Q.614 (Sheet 13 of 14)

**Incoming Signalling System No. 7 (TUP)**

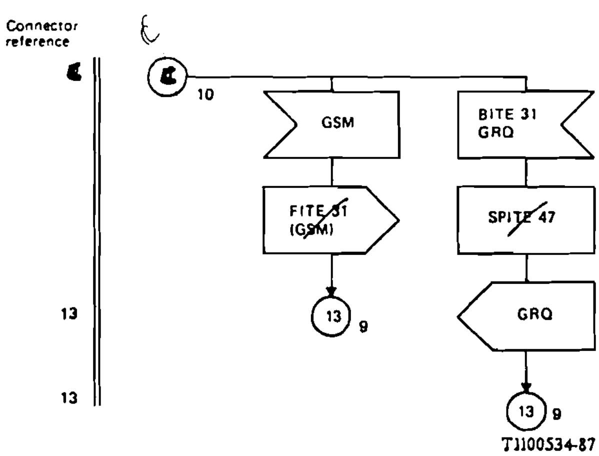
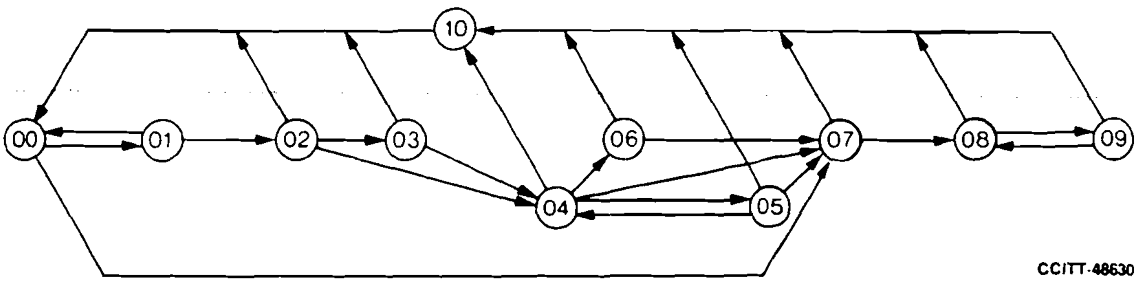



FIGURE 3/Q.614 (Sheet 14 of 14)

**Incoming Signalling System No. 7 (TUP)**

LOGIC PROCEDURES FOR OUTGOING SIGNALLING SYSTEM R2




| State number | State description | Sheet reference | Timers running |
|---|---|---|---|
| 00 | Idle | 1, 4 | |
| 01 | Wait for calling party's category (CPCI) | 1 | |
| 02 | Wait for country code indicator (CCI) | 1 | |
| 03 | Wait for echo suppressor indicator (ECI) | 1 | |
| 04 | Wait for backward signal | 2 | $t_1$ |
| 05 | Wait for address information | 3 | $t_2$ |
| 06 | Wait for Type B signal | 3 | $t_1$ |
| 07 | Wait for answer | 4 | |
| 08 | Answered | 4 | |
| 09 | Clear-back | 4 | |
| 10 | Clear-forward | 4 | |

FIGURE 1/Q.626

**State overview diagram for outgoing Signalling System R2**

*Supervisory timers for outgoing Signalling System R2*

| | |
|---|---|
| $t_1$ = 12-18 s | Recommendation Q.476, § 5.5.1.1 |
| $t_2$ > 24 s | Recommendation Q.476, § 5.5.1.2 |

*Procedures not shown*

The following procedures, not directly relevant to interworking, are not shown in the logic procedures:

- Interrupt control procedures (analogue version).
- Seizing acknowledgement (digital version).
- Transmission fault procedures (digital version).
- $T_1$ time-out and abnormal release sequence (analogue version).
- Optional forward transfer.
- Blocking and unblocking sequences.

FIGURE 2/Q.626

**Notes to outgoing Signalling System R2**

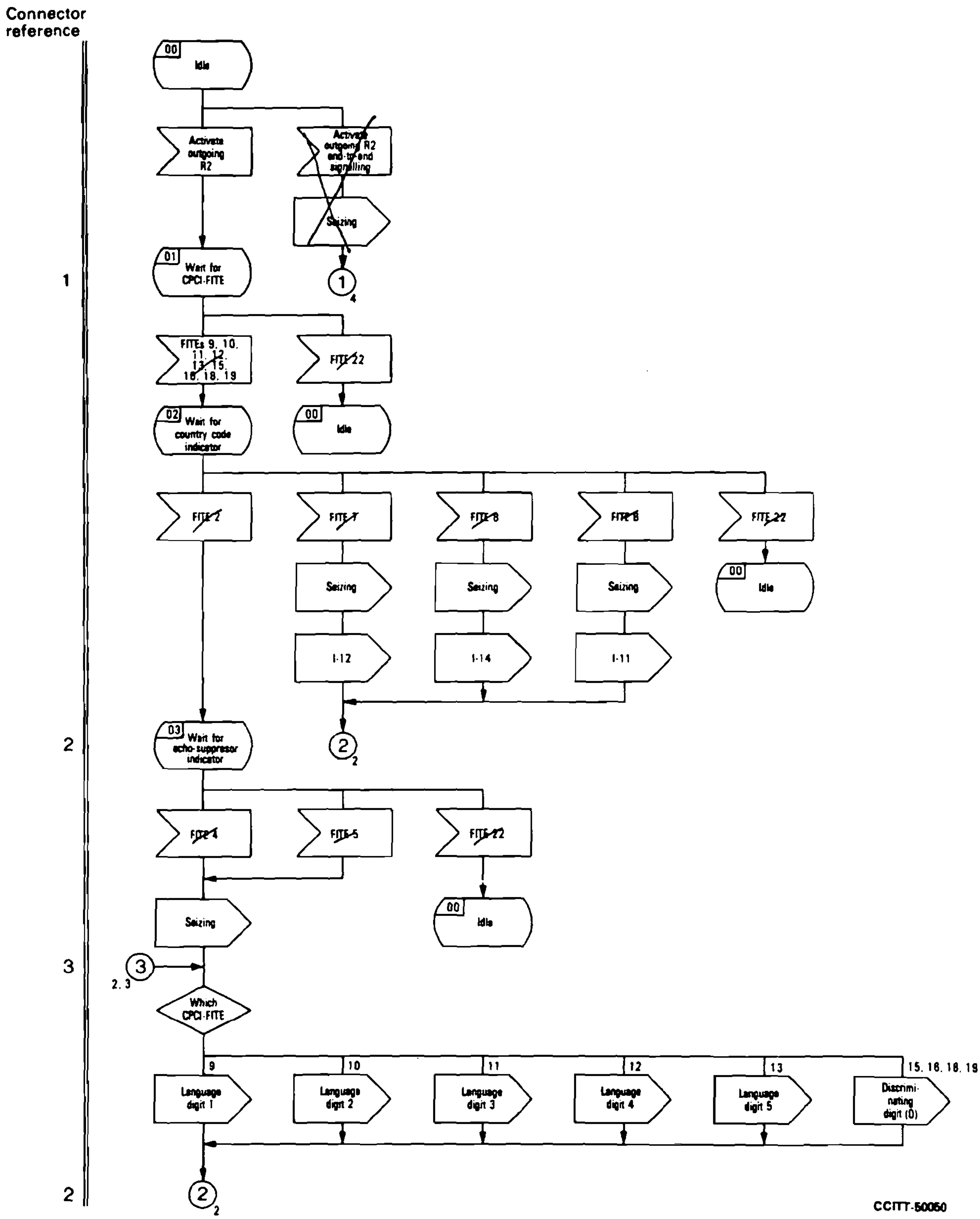



FIGURE 3/Q.626 (Sheet 1 of 4)
**Outgoing Signalling System R2**

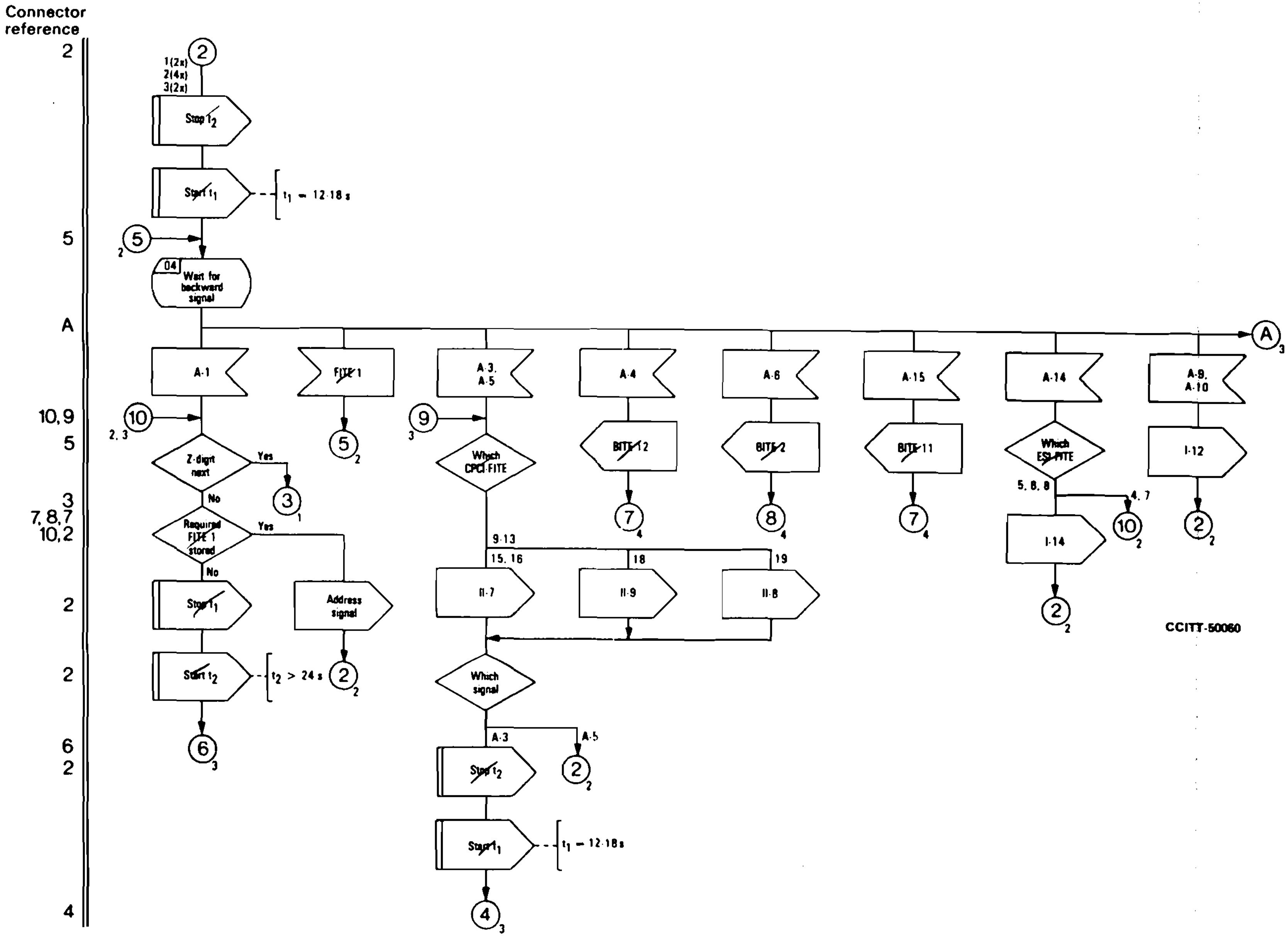



FIGURE 3/Q.626 (Sheet 2 of 4)

**Outgoing Signalling System R2**

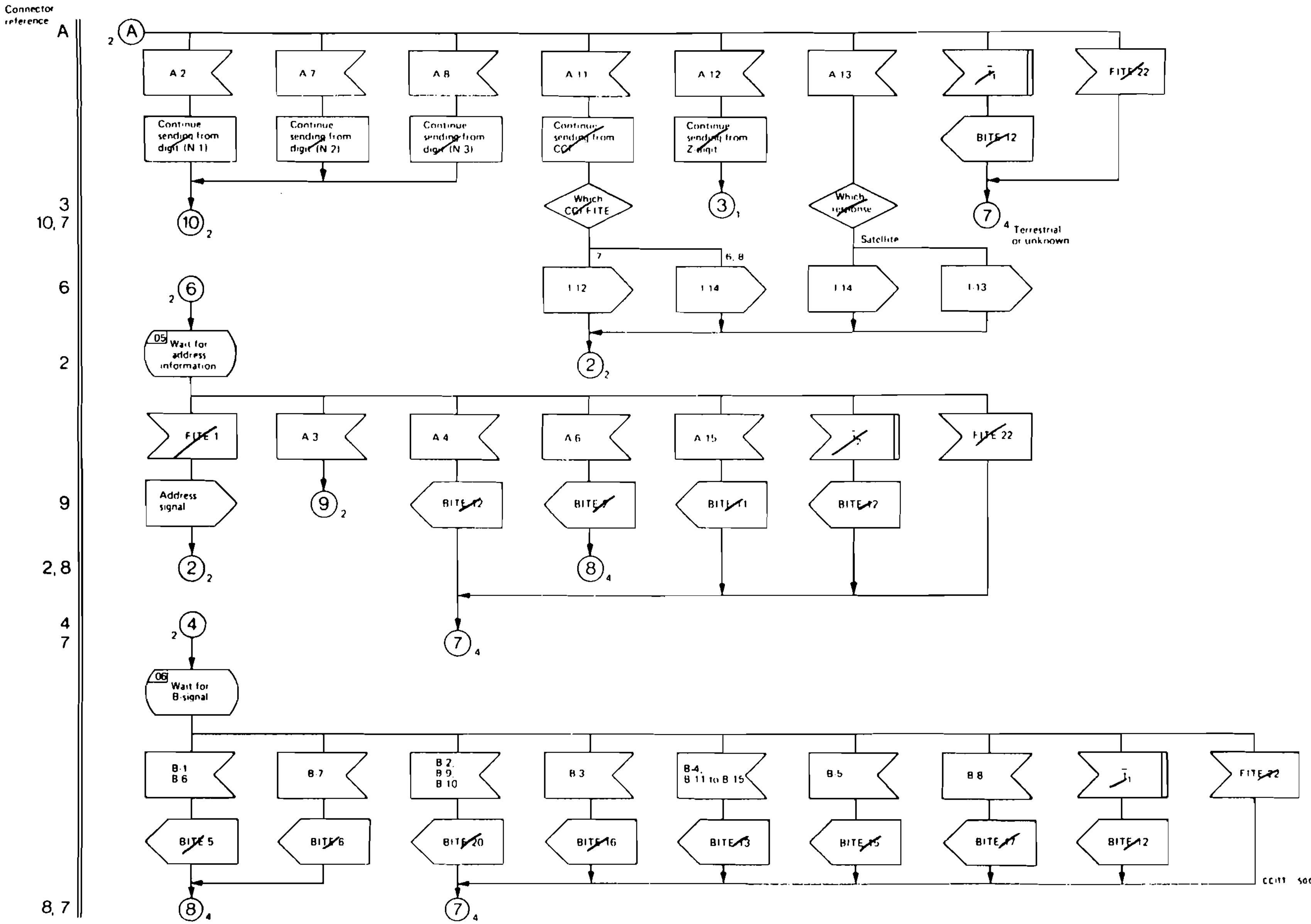



FIGURE 3/Q.626 (Sheet 3 of 4)

**Outgoing Signalling System R2**

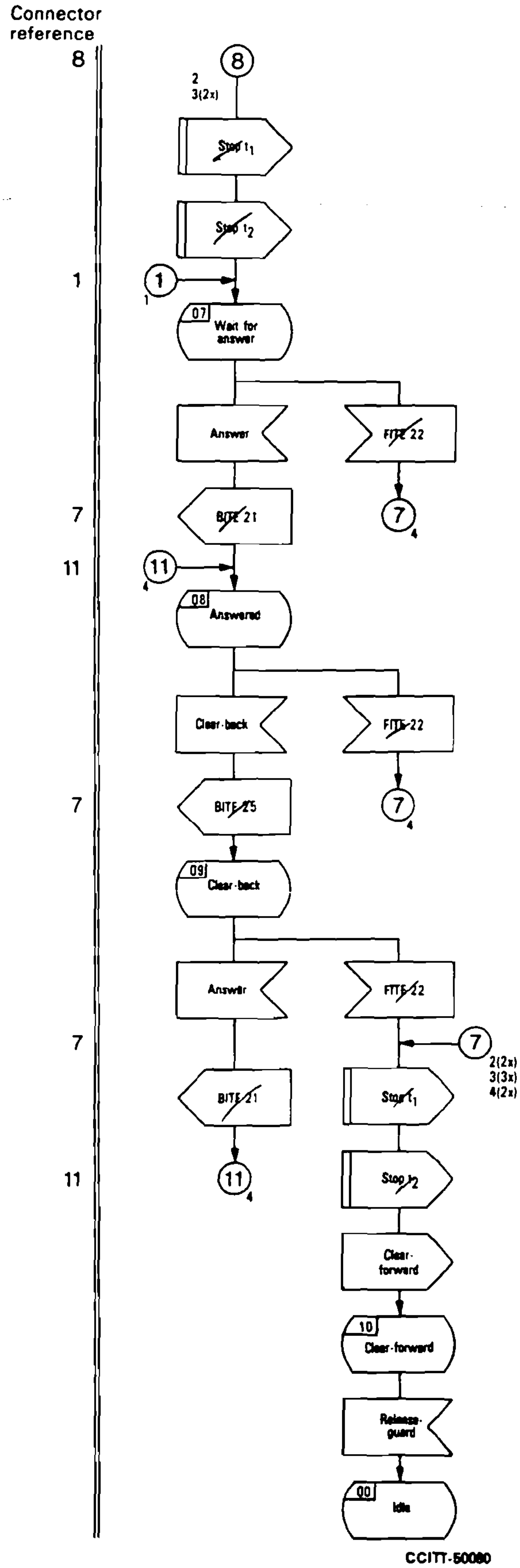
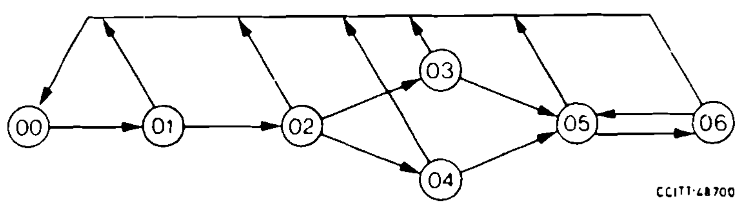



FIGURE 3/Q.626 (Sheet 4 of 4)

**Outgoing Signalling System R2**

LOGIC PROCEDURES FOR INTERWORKING OF SIGNALLING SYSTEM No. 7 (TUP) TO R2




| *State number* | *State description* | *Sheet reference* |
|---|---|---|
| 00 | Idle | 1, 2 |
| 01 | Wait for CPCI-FITF | 1 |
| 02 | Wait for address-complete | 2 |
| 03 | Wait for answer, charge | 2 |
| 04 | Wait for answer, no charge | 2 |
| 05 | Answered | 2 |
| 06 | Clear-back | 2 |

FIGURE 1/Q.666

**State overview diagram for interworking of Signalling System No. 7 (TUP) to R2**

FIGURE 2/Q.666

(Reserved for future use)

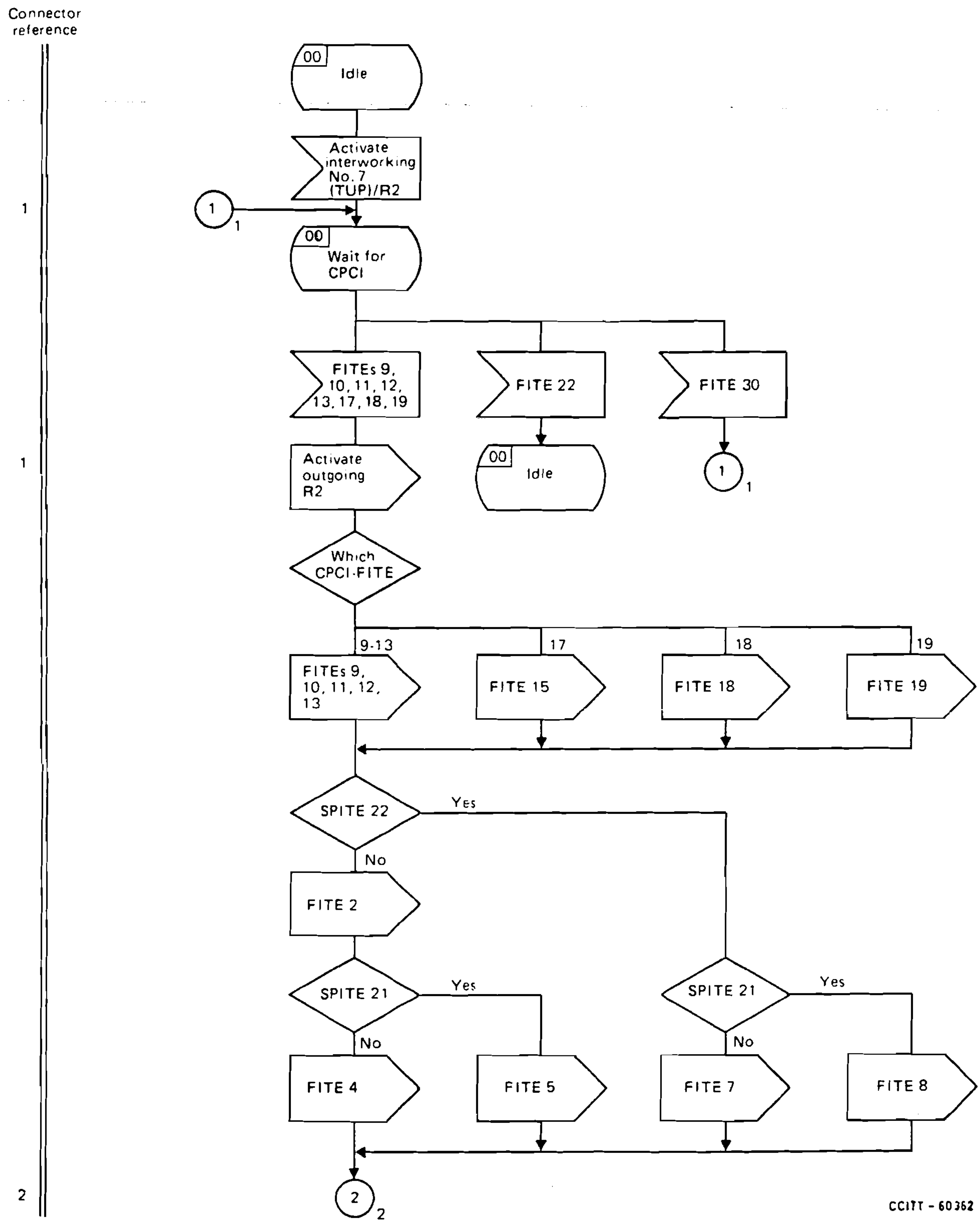



FIGURE 3/Q.666 (Sheet 1 of 2)

**Interworking of Signalling System No. 7 (TUP) to R2**

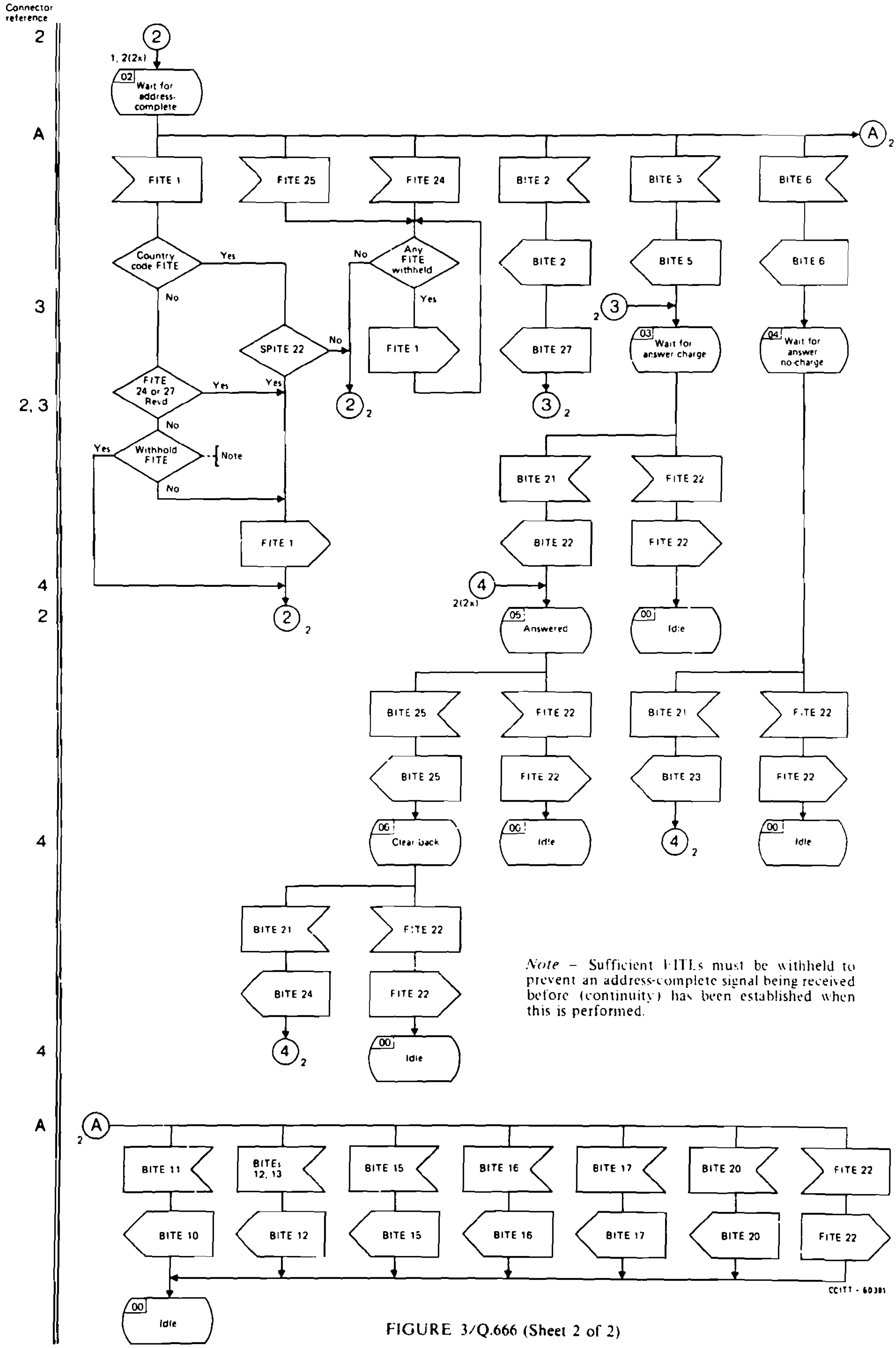



FIGURE 3/Q.666 (Sheet 2 of 2)

**Interworking of Signalling System No. 7 (TUP) to R2**

# Appendix D FITEs, BITEs and SPITEs

ANNEX A

(to Recommendations Q.601-Q.608)

**Lists and meanings of FITEs, BITEs and SPITEs.**
**Representation of information contents of signals**
**of the Signalling Systems.**

TABLE A-1

**List of forward interworking telephone events (FITEs)**

| No. | Forward interworking telephone events | Equivalent with signal of Signalling System | | | | | |
|---|---|---|---|---|---|---|---|
| | | No. 4 | No. 5 | No. 6 | No. 7 (TUP) | R1 | R2 |
| 1 | Digit 1, 2, . . . 9 or 0, code 11 or 12, end-of-pulsing (ST) signal | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | Country-code indicator, country code not included | 8 | 8 | 2 | 2 | | 18 |
| 3 | Country-code indicator, country code included | 9 | 9 | 3 | 3 | | |
| 4 | Echo-suppressor indicator, outgoing half-echo suppressor not included, incoming half-echo suppressor not required | | | 6 | 6 | | 19 |
| 5 | Echo-suppressor indicator, outgoing half-echo suppressor included, incoming half-echo suppressor required | 10 | | 7 | 7 | | 11 |
| 6 | Country-code indicator, country code included; echo-suppressor indicator, outgoing half-echo suppressor not included, outgoing half echo suppressor required | | | | | | 8 |
| 7 | Country-code indicator, country code included; echo-suppressor indicator, outgoing half-echo suppressor not included, no echo suppressor required | | | | | | 9 |
| 8 | Country-code indicator, country code included; echo-suppressor indicator, outgoing half-echo suppressor included, incoming half-echo suppressor required | | | | | | 10 |
| 9 | Calling-party's-category indicator, operator, language French | 2 | 2 | 8 | 8 | | 2 |
| 10 | Calling-party's-category indicator, operator, language English | 3 | 3 | 9 | 9 | | 3 |
| 11 | Calling-party's-category indicator, operator, language German | 4 | 4 | 10 | 10 | | 4 |
| 12 | Calling-party's-category indicator, operator, language Russian | 5 | 5 | 11 | 11 | | 5 |
| 13 | Calling-party's-category indicator, operator, language Spanish | 6 | 6 | 12 | 12 | | 6 |
| 14 | Calling-party's-category indicator, operator with forward-transfer facility | | | | | | 15 |
| 15 | Calling-party's-category indicator, subscriber | | | | | | 7 |
| 16 | Calling-party's-category indicator, subscriber or operator without forward-transfer facility | | | | | | 12 |

TABLE A-1 *(cont.)*

| No. | Forward interworking telephone events | Equivalent with signal of Signalling System | | | | | |
|---|---|---|---|---|---|---|---|
| | | No. 4 | No. 5 | No. 6 | No. 7 (TUP) | R1 | R2 |
| 17 | Calling-party's-category indicator, subscriber, ordinary call | 7 | 7 | 13 | 13 | | |
| 18 | Calling-party's-category indicator, subscriber, call with priority | | | 14 | 14 | | 14 |
| 19 | Calling-party's-category indicator, data call | | | 15 | 15 | | 13 |
| 20 | Nature-of-circuit indicator, no satellite circuit in the connection | | | 4 | 4 | | 20 |
| 21 | Nature-of-circuit indicator, one satellite circuit in the connection | | | 5 | 5 | | 21 |
| 22 | Clear-forward | 11 | 10 | 16 | 16 | 3 | 16 |
| 23 | Forward-transfer | 12 | 11 | 17 | 17 | 2 | |
| 24 | Continuity | | | 18 | 18 | | |
| 25 | Continuity-check indicator, continuity check not required | | | | 21 | | |
| 26 | Continuity-check indicator, continuity check required on this circuit | | | | 20 | | |
| 27 | Continuity-check indicator, continuity check being done on previous circuit | | | | 22 | | |
| 28 | Spare | | | | | | |
| 29 | Spare | | | | | | |
| 30 | Service information | | | | 23 | | |
| 31 | General set-up message (GSM) | | | | 24 | | |

TABLE A-2

**List of backward interworking telephone events (BITEs)**

| No. | Backward interworking telephone events | Equivalent with signal of Signalling System | | | | | |
|---|---|---|---|---|---|---|---|
| | | No. 4 | No. 5 | No. 6 | No. 7 (TUP) | R1 | R2 |
| 1 | Spare | | | | | | |
| 2 | Address-complete, charge | 1 | | 1 | 1 | | 2 |
| 3 | Address-complete, no charge | | | 2 | 2 | | |
| 4 | Address-complete, coin box | | | 3 | 3 | | |
| 5 | Address-complete, subscriber free, charge | | | 4 | 4 | | 8 and 13 |
| 6 | Address-complete, subscriber free, no charge | | | 5 | 5 | | 9 |
| 7 | Address-complete, subscriber free, coin box | | | 6 | 6 | | |
| 8 | Call unsuccessful | 2 | 1 | | | | |
| 9 | Call unsuccessful, switching-equipment congestion | | | 7 | 7 | | |
| 10 | Call unsuccessful, circuit-group congestion | | | 8 | 8 | | |
| 11 | Call unsuccessful, switching-equipment congestion or circuit group congestion | | | | | | 3 |
| 12 | Call unsuccessful, national-network congestion | | | 9 | 9 | | 1 |
| 13 | Call unsuccessful, address-complete, national network congestion | | | | | | 6 and 15 |
| 14 | Call unsuccessful, address incomplete | | | 10 | 10 | | |
| 15 | Call unsuccessful, (address-complete), unallocated number | | | 11 | 11 | | 7 and 14 |
| 16 | Call unsuccessful, address-complete, subscriber busy (elec.) | | | 12 | 12 | | 5 |
| 17 | Call unsuccessful, address-complete, line out of service | | | 13 | 13 | | 10 |
| 18 | Spare | | | | | | |
| 19 | Call unsuccessful, call-failure | | | 15 | 15 | | |
| 20 | Call unsuccessful, send special information tone | | | 14 | 14 | | 4 and 14 |
| 21 | Answer, subscriber free | | | | | | 11 |
| 22 | Answer, subscriber free, charge | 3 | 2 | 16 | 16 | 1 | |
| 23 | Answer, subscriber free, no charge | | | 17 | 17 | | |
| 24 | Answer, re-answer | | | 18 | 18 | | |
| 25 | Clear-back | 4 | 3 | 19 | 19 | 2 | 12 |

TABLE A-2 *(cont.)*

| No. | Backward interworking telephone events | Equivalent with signal of Signalling System | | | | | |
|---|---|---|---|---|---|---|---|
| | | No. 4 | No. 5 | No. 6 | No. 7 (TUP) | R1 | R2 |
| 26 | Artificial address complete is sent [a)] | | 4 | | | 3 | |
| 27 | Sending-finished; set up speech condition [a)] | | 5 | | | 4 | |
| 28 | Deactivate register function [a)] | | | | | | |
| 29 | Release incoming side [a)] | | | | | | |
| 30 | Switchthrough may be completed [a)] | | | | | | |
| 31 | General request message (GRQ) | | | | 20 | | |
| 32 | Spare | | | | | | |
| 33 | Spare | | | | | | |
| 34 | Spare | | | | | | |
| 35 | Call unsuccessful, access barred | | | | 21 | | |
| 36 | Call unsuccessful, digital patch not provided (DPN) | | | | 22 | | |

[a)] These signals do not necessarily correspond to a backward signal but correspond to logic events.

TABLE A-3

**List of switching processing interface telephone events (SPITEs)**

| No. | Designation | Symbol |
|---|---|---|
| 1 | Activate register function (physical register or equivalent function) | Task |
| 2 | Register function activated | Internal input |
| 3 | Deactivate register function | Task |
| 4 | Set up speech condition | Task |
| 4A | Speech path may be set up | Task |
| 5 | Release speech condition (of the speech path in the exchange) | Task |
| 6 | Return appropriate tone | Task |
| 7 | Disconnect tone | Task |
| 8 | Release all equipment (covers also disconnection of tones; exclusively used at the incoming procedures) | Task |
| 9 | Spare | |
| 10 | Spare | |
| 11 | Shall digit analysis be started? | Decision |
| 12 | Perform digit analysis | Task |
| 13 | Digit analysis cannot be completed (covers insufficient information, waiting for enough digits for routing, etc.) | Internal input |
| 14 | Routing information and service provided | Internal input |
| 15 | Unallocated number | Internal input |
| 16 | Unprovided routing (e.g. transit connection received at an exchange handling termination traffic only) | Internal input |
| 17 | Barred routing | Internal input |
| 18 | Switching equipment congestion | Internal input |
| 19 | Circuit group congestion | Internal input |
| 20 | Satellite link included? | Decision |
| 21 | Incoming half-echo suppressor to be included at distant end? | Decision |
| 22 | Transit connection following? (otherwise a terminal connection is following) | Decision |
| 23 | Has Z-digit been received? | Decision |
| 24 | Is this the Z-digit? | Decision |
| 31 | Perform continuity check at the outgoing end (covers all necessary switching procedures: – connecting of the transceiver – disabling of echo suppressors – sending check tone – automatic reattempts, where applicable) | Task |
| 32 | Insert check loop at the incoming end (including disabling of echo suppressors) | Task |
| 33 | Continuity check OK (covers also receiving of checktone and removal of the transceiver) | Internal input |
| 34 | Remove check loop at the incoming end (including enabling of echo suppressors) | Task |
| 35 | Ignore further register signals | Task |
| 36 | Is continuity check required on outgoing circuit? | Decision |

TABLE A-3 *(cont.)*

| No. | Designation | Symbol |
|---|---|---|
| 37 | Analyze and store information (received in GSM) | Task |
| 38 | Access barred | Internal input |
| 39 | Digital path not possible | Internal input |
| 40 | Store services provided | Task |
| 41 | Spare | |
| 42 | Additional information required | Internal input |
| 43 | Set IAM fields as defined by services | Task |
| 44 | Analyze information requested (applicable to GRQ) | Task |
| 45 | Is service allowed? (applicable to GRQ) | Decision |
| 46 | Is information available? (applicable to GRQ) | Decision |
| 47 | Construct request message (applicable to GRQ) | Task |
| 48 | Construct information message (applicable to GSM) | Task |
| 49 | Spare | |
| 50 | Spare | |

# Appendix E R2 group A,B, I and II signals

TABLE A-7

**Representation of the information contents – forward signals of Signalling System R2**

| Information elements \ Signals of Signalling System R2 | Address signals | I-1: Language digit: French | I-2: Language digit: English | I-3: Language digit: German | I-4: Language digit: Russian | I-5: Language digit: Spanish | I-10: Discriminating digit | I-11: Country-code indicator, outgoing half-echo suppressor required | I-12: Country-code indicator, no echo suppressor required | I-14: Country-code indicator, incoming half-echo suppressor required[e)] | I-14: Incoming half-echo suppressor required[e)] | II-7: Calling party's category, subscriber or operator without forward-transfer facility | II-8: Calling party's category, data transmission control | II-9: Calling party's category, subscriber with priority | II-10: Calling party's category operator with forward-transfer facility | Clear-forward signal | Forward-transfer signal[a)] | First digit I-1, I-2, ... I-10 | Reply to A-14, I-1, ... I-10[e)] | Reply to first A-13, I-13 | Reply to first A-13, I-14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Signal No. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
| Digit 1, 2, ... 9 or 0, code 11 or 12, | × | | | | | | | | | | | | | | | | | | × | | |
| end-of-pulsing (ST) signal | × | | | | | | | | | | | | | | | | | | | | |
| Country-code indicator | | | | | | | | × | × | × | | | | | | | | × | | | |
| Country-code not included | | | | | | | | | | | | | | | | | | × | | | |
| Country-code included | | | | | | | | × | × | × | | | | | | | | | | | |
| Nature-of-circuit indicator | | | | | | | | | | | | | | | | | | | | × | × |
| No satellite circuit in the connection | | | | | | | | | | | | | | | | | | | | × | |
| One satellite circuit in the connection | | | | | | | | | | | | | | | | | | | | | × |
| Echo-suppressor indicator | | | | | | | | × | × | × | × | | | | | | | | | | |
| Outgoing half-echo suppressor not included | | | | | | | | × | × | | | | | | | | | | | | |
| Outgoing half-echo suppressor included | | | | | | | | | | × | × | | | | | | | | | | |
| Outgoing half-echo suppressor required | | | | | | | | × | | | | | | | | | | | | | |
| No echo-suppressor required | | | | | | | | | × | | | | | | | | | | | | |
| Incoming half-echo suppressor required | | | | | | | | | | × | × | | | | | | | | | | |
| Calling party's category indicator | | × | × | × | × | × | × | | | | | × or | × | × | × | | | | | | |
| Operator | | × | × | × | × | × | | | | | | × | | | × | | | | | | |
| Subscriber | | | | | | | × | | | | | × | | × | | | | | | | |
| Data call | | | | | | | | | | | | | × | | | | | | | | |
| Ordinary call | | | | | | | | | | | | × | | | | | | | | | |
| Call with priority | | | | | | | | | | | | | | × | | | | | | | |
| Forward-transfer facility | | | | | | | | | | | | | | | × | | | | | | |
| No forward-transfer facility | | | | | | | | | | | | × | | | | | | | | | |
| Language: French | | × | | | | | | | | | | | | | | | | | | | |
| Language: English | | | × | | | | | | | | | | | | | | | | | | |
| Language: German | | | | × | | | | | | | | | | | | | | | | | |
| Language: Russian | | | | | × | | | | | | | | | | | | | | | | |
| Language: Spanish | | | | | | × | | | | | | | | | | | | | | | |
| Clear-forward | | | | | | | | | | | | | | | | × | | | | | |
| Forward-transfer | | | | | | | | | | | | | | | | | × | | | | |
| Continuity check indicator | | | | | | | | | | | | | | | | | | | | | |
| Continuity check required on this circuit | | | | | | | | | | | | | | | | | | | | | |
| Continuity check not required on this circuit | | | | | | | | | | | | | | | | | | | | | |
| Continuity check performed on previous circuit | | | | | | | | | | | | | | | | | | | | | |
| Continuity | | | | | | | | | | | | | | | | | | | | | |
| General setup information elements | | | | | | | | | | | | | | | | | | | | | |
| Service information elements | | | | | | | | | | | | | | | | | | | | | |
| Corresponds to signal No. ... of Signalling System No. 4 | 1 | 2 | 3 | 4 | 5 | 6 | [7] | 10[c)] +9[d)] | (9) | 10[d)] +9 | 10 | 7 | ⊗ | ⊗ | [7[f)]] | 11 | 12 | 8 | ⊗ | ⊗ | ⊗ |
| No. 5 | 1 | 2 | 3 | 4 | 5 | 6 | [7] | ⊗ | (9) | ⊗ | ⊗ | 7 | ⊗ | ⊗ | [7[f)]] | 10 | 11 | 8 | ⊗ | ⊗ | ⊗ |
| No. 6 | 1 | 8 | 9 | 10 | 11 | 12 | b) | 3 7[c)] | 3 +6 | 3 +7 | 7 | 13 | 15 | 14 | [13[f)]] | 16 | 17 | 2 | 6 | 4 | 5 |
| No. 7 | 1 | 8 | 9 | 10 | 11 | 12 | b) | 3[c)] 8 | 3 6[c)] | 3 7 | 7 | 13 | 15 | 14 | [13[f)]] | 16 | 17 | 2 | 6 | 4 | 5 |
| R1 | 1 | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | ⊗ | 3 | 2 | ⊗ | ⊗ | ⊗ | ⊗ |

T1107620-87

⊗ No equivalent signal

(No.) Loss of information

[No.] Additional information or change of information

No. Identical meaning of signals

a) See Recommendation Q.400, § 1.1.3. A forward-transfer signal does not normally form part of Signalling System R2.

b) See signals II-7, II-8 and II-9.

c) Translation of signal I-11, succeeding circuit must be equipped with outgoing half-echo suppressor.

d) Signal code 14 is available for use upon multilateral or bilateral agreement for echo-suppressor control.

e) In reply to signal A-14.

f) The I/O logic treats the signal II-10 as II-7, as long as no treatment of the forward-transfer signal is considered.

BB 374660

## TABLE A-11

**Representation of the information contents – backward signals of Signalling System R2**

| Information elements \ Signals of Signalling System R2 | A-4: Congestion on the national network | A-6: Address-complete, charge, set-up speech conditions | A-15: Congestion in an international exchange or at its output | B-2: Send special-information tone | B-3: Subscriber line busy | B-4: Congestion (encountered after change-over from A-signals to B-signals) | B-5: Unallocated number | B-6: Subscriber line free, charge | B-7: Subscriber line free, no charge | B-8: Subscriber line out of order | Answer signal | Clear back signal | B-1-B-6: International, subscriber line free, charge | B-9, B-10: International send special information tone | B-11-B-15 = B-4 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Signal No. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | |
| Address-complete | | × | | (×) | × | × | (×) | × | × | × | | | × | (×) | × | | | | | |
| Subscriber free | | | | | | | | × | × | | × | | × | | | | | | | |
| Coin box | | | | | | | | | | | | | | | | | | | | |
| Charge | | × | | | | | | × | | | | | × | | | | | | | |
| No charge | | | | | | | | | × | | | | | | | | | | | |
| Call unsuccessful | × | | × | × | × | × | × | | | × | | | | × | × | | | | | |
| Switching-equipment congestion | | | × or | | | | | | | | | | | | | | | | | |
| Circuit-group congestion | | | × | | | | | | | | | | | | | | | | | |
| National-network congestion | × | | | | | × | | | | | | | | | × | | | | | |
| Address-incomplete | | | | | | | | | | | | | | | | | | | | |
| Unallocated number | | | | | | | × | | | | | | | | | | | | | |
| Subscriber busy (elec.) | | | | | × | | | | | | | | | | | | | | | |
| Line-out-of-service | | | | | | | | | | × | | | | | | | | | | |
| Send special-information tone | | | | × | | | | | | | | | | × | | | | | | |
| Call failure | | | | | | | | | | | | | | | | | | | | |
| Answer | | | | | | | | | | | × | | | | | | | | | |
| Re-answer | | | | | | | | | | | | | | | | | | | | |
| Clear-back | | | | | | | | | | | | × | | | | | | | | |
| Artificial address-complete | | | | | | | | | | | | | | | | | | | | |
| Sending-finished | | | | | | | | | | | | | | | | | | | | |
| General setup request elements | | | | | | | | | | | | | | | | | | | | |
| Access barred | | | | | | | | | | | | | | | | | | | | |
| Digital path not provided | | | | | | | | | | | | | | | | | | | | |
| Corresponds to signals No. ... of Signalling System No. 4 | (2) | 1 | (2) | (a)) | (2) | (2) | (a)) | ⊗ | ⊗ | (a)) | (3) | 4 | ⊗ | (a)) | (2) | | | | | |
| No. 5 | (1) | ⊗ | (1) | (a)) | (1) | (1) | (a)) | ⊗ | ⊗ | (a)) | (2) | 3 | ⊗ | (a)) | (1) | | | | | |
| No. 6 | 9 | 1 | (8) | 14 | 12 | 9 | 11 | 4 | 5 | 13 | [16,17 or 18] | 19 | 4 | 14 | 9 | | | | | |
| No. 7 | 9 | 1 | (8) | 14 | 12 | 9 | 11 | 4 | 5 | 13 | [16,17 or 18] | 19 | 4 | 14 | 9 | | | | | |
| R1 | (b)) | ⊗ | (b)) | (b)) | (b)) | (b)) | (b)) | ⊗ | ⊗ | (b)) | 1 | 2 | ⊗ | (b)) | (b)) | | | | | |

T1107670-87

a) Special information tone.

b) Appropriate tone or announcement.

⊗ No equivalent signal

(No.) Loss of information

[No.] Additional information or change of information

No. Identical meaning of signals