



ENGLISH VERSION

**THE APOLLO II (TEMPORA)
DEVICE**

INSTALLATION MANUAL

Made by



**De Vunt 14
B-3220 Holsbeek (Belgium)**

**) ++32 (0)16 44 01 23 Fax ++32 (0)16 44 66 40
E-mail : export.team@clock-o-matic.com**

November 2002

TABLE OF CONTENTS

A. PRELIMINARY REMARKS	7
<i>Introduction</i>	8
<i>Glossary</i>	8
B. HARDWARE	10
1. <i>Different versions APOLLO II - device</i>	11
2. <i>Additional options for the APOLLO II (Tempora) device</i>	11
3. <i>Characteristics of the APOLLO II device (Tempora)</i>	12
4. <i>Changing the battery</i>	12
5. <i>standard</i>	13
6. <i>Number of software-items</i>	14
7. <i>Radio-synchronization</i>	14
C. SOFTWARE	15
A. General	16
1. <i>Front panel</i>	16
2. <i>Main menu</i>	16
3. <i>User menu</i>	17
4. <i>Function menu</i>	18
5. <i>Structure of the functions</i>	19
6. <i>Entering parameters</i>	20
7. <i>Power failure</i>	21
8. <i>Example</i>	21
9. <i>Diagram (shortened selection of characters)</i>	22
B. Programming the APOLLO II device	23
C. Parameters and description of the functions	26
1. <i>Set-up (function 14)</i>	26
1.1. <i>Contrast display (function 1401)</i>	26
1.2. <i>User code (function 1402)</i>	26
1.3. <i>Function code (function 1403)</i>	26
1.5. <i>Software (function 1405)</i>	27
1.6. <i>Load PIM (function 1406)</i>	27
2. <i>Relay (function 13)</i>	28

3. Clocks (function 12).....	29
3.1 Internal clock (function 1201)	29
3.1.1. Changing time (function 120101)	29
3.1.2. Changing date (function 120102).....	29
3.1.3. Time format (function 120103).....	30
3.1.4. Running speed (function 120104).....	30
3.1.5. Shift radio (function 120105).....	30
3.1.6 Daylight saving (function 120106)	30
3.2. Slave clock 1 (function 1202)	31
3.2.1. Time slave clock 1 (function 120201)	31
3.2.2. Type slave clock 1 (function 120202).....	31
3.3. Slave clock 2 (function 1203).....	32
4. Special outputs (function 11).....	32
4.1. Louver-hole (function 1101).....	32
4.2. Roman knell (function 1102)	32
4.3. MIDI (function 1103)	33
<i>This function is only effective when using the version Apollo II-SBSI</i>	33
4.4. Speech (function 1104).....	33
4.5. Serial output (function 1105).....	33
5. Hammers (function 10).....	34
6. Program blocks (function 09).....	35
7. Continuous outputs (function 08).....	36
8. Parameters for ringing (function 05).....	36
8.1. Individual ringing (functions 0501-0510).....	36
8.2. Joint ringing functions: ringing order (function 0511).....	37
9. Executions (function 01).....	38
9.1. Ringing (function 0101).....	38
9.2. Play sequence (function 0102).....	39
9.3. Play group (function 0103)	40
9.4. Switch on continuous (function 0104)	41
9.5. Program block (function 0105).....	41
9.6. Manual tolling (function 0106)	42
9.7. Pause (function 0107)	42
10. Sequences (function 06).....	43
10.1. Record sequences (function 0601)	44
Example 1 : Week mass (ringing 1 for 2 minutes)- sequence type	
1	44
10.2. Test sequence (function 0602).....	46
10.3. Save recording (function 0603)	47
Example 2 : stroke bell 1 (for hour stroke)	47
Example 3 :Angelus (strokes on bell 2 + ringing on bell 1)	49
10.4. Survey record sequence type 1.....	51
10.5. Names sequences (function 0605)	51
10.5. The use of pauses during the recording of a sequence	52
Example 4 : Celebration mass - sequence type 1	52
10.6. “STOP” function during “record sequence”.	52
Example 5: Manual tolling sequence type 1	52
Example 6: Change hands sequence type 3	54
10.7. Properties sequences (function 0606)	55
10.8. Delete sequence (function 0604).....	56
11. Programs (function 04).....	56
11.1. Single programs (function 0401)	57

11.1.1. Single ringing programs (function 040101)	57
11.1.2. Single sequence programs (function 040102)	58
11.1.3. Single group programs (function 040103)	59
11.1.4. Single continuous programs (function 040104)	60
11.1.5. Single block programs (function 040105).....	61
11.2. Permanent programs (function 0402).....	62
11.2.1. Permanent ringing programs (function 040201)	62
11.2.2. Easter programs	65
11.2.3 Similar program items.....	66
11.2.3.1. Permanent group programs (function 040203)	67
11.2.3.2. Permanent continuous programs (function 040204)	67
11.2.3.3. Permanent block programs (function 040205).....	68
11.2.3.4. Permanent sequence programs (function 040202)	68
11.3. Priorities of the executions.....	71
12. Groups (function 07).....	73
12.1. Names groups (function 0704).....	73
12.2. Content groups (function 0703)	74
12.3. Time between sequences (function 0702)	74
12.4. Active group (function 0701).....	74
13. User menu (function 02).....	75
13.1. User type (function 02xx01) : xx = 1 to 30	76
13.2. User action (function 02xx02) : xx = 1 to 30	76
13.3. Function key 1-0 (function 0231)	78
14. Inputs (function 03)	78

D. PROGRAM EXAMPLE 79

1. Preliminary remarks.....	80
2. Description of the installation	80
3. Program form	80
3.A. Hour stroke, half hour stroke + quarter strokes (sequences)	85
3.A.1 Hour stroke + half hour stroke	85
3.A.2 Quarter strokes	88
3.B. Hour stroke, half hour stroke + quarter strokes (programs)	88
3.B.1. Hour stroke programming (sequence program 1).....	88
3.B.2. Half hour stroke programming (sequence program 2).....	89
3.B.3. 1st and 3rd quarter stroke programming (sequence program 3 and 5)	89
3.B.4. 2nd and 4th quarter stroke programming (sequence program 4 and 6)	90
3.C. Angelus	91
3.C.1 Sequence	91
3.C.2. Programs	93
3.D. Week-, Saturday- and Sunday masses (programs)	94
3.E. Funeral, marriage (sequence)	95
3.E.1. Funeral.....	95
3.E.2. Marriage	96
3.E.3. Mourning tolling.....	97
3.F. Tower clock.....	98
4. Sequences to change programs (sequence type 2).....	103
4.1. Introduction.....	103
4.2. Example 1: Single ‘marriage’ program	106
4.3. Example 2: Single ‘mourning tolling’ program	109
4.4. Example 3 : ‘Permanent ringing program’ sequence	111
5. Automatic “holiday” - program.....	114
5.1. Introduction.....	114

E. SPECIAL APPLICATIONS..... 122

1. APOLLO II - serial.....	123
1.1. Connections	123
1.2. The connection of 2 serial power boards	125
1.3. Options – settings.....	126
2. APOLLO II-SBSI with MIDI IN/OUT interface.....	127
2.1. Preliminary remarks – connections (see also the plan at the end).....	127
2.2. Programming	129
2.3. Sequences for APOLLO II-SBSI (carillon computer).....	130
2.3.1. “Record melody”	130
2.3.2. “Test melody”	130
2.3.3. “Save melody”	131
2.3.4. “Play melody”	131
2.4. Programming melodies (carillon computer).....	132
2.4.1 Non-cyclic programming	132
2.4.2 Cyclic programming.....	133
2.4.2.1 Names groups (function 0704).....	134
2.4.2.2 Content groups (function 0703).....	134
2.4.2.2 Active group (function 0701).....	136
2.4.2.3 Programming the group changeover (function 0701).....	136
2.5 Sequences for groups (with cyclic programming).....	137
2.5.1 “Fill groups”	137
2.5.2 “Choose group”.....	138
3. Digital Bell Apollo II (Tempora)	138
3.1. Recording a song (Westminster or one of the 4 extra melodies) in a sequence	139
3.2. Recording sequences with “tolling” (digital sounds or hammers).....	141
3.3. The use of real ringing bells with the Digital Bell device	143

F. OPTIONS..... 144

1. Introduction.....	145
2. PIM-Module.....	145
3. MIDI-Interface.....	147
5. DTMF Interface + GSM.....	148
5.1. Requirements	148
5.2. Installation	150
5.3. Working of the system	152
5.4. Control-lamps	152

G. ANNEXE : DRAWINGS..... 153

APOLLO II DIMENSIONS	155
APOLLO II – 10 OUTPUTS.....	156
APOLLO II 16 OUTPUTS.....	157
APOLLO II-SBSI (MASTER CLOCK + OUTPUT BOARD)	158
APOLLO II-SBSI (master clock)	159
SERIAL APOLLO II WITH OUTPUT BOARD (6 REL + 4 SORTIES).....	160
SERIAL APOLLO II WITH OUTPUT BOARD (10 FET OUTPUTS)	161
TEMPORA - DIMENSIONS	163
TEMPORA – 6 OUTPUTS with optional print.....	164

PIM.....	165
AUDIO CONNEXION DIGITAL BELL TEMPORA (APOLLO II).....	167
CONNEXION OPTIONAL PRINT TEMPORA (INPUTS).....	168
CONNEXION OF THE APOLLO II – COMPUTER.....	169
CONNEXION EXTERNAL 24V POLARISED IMPULSES CONVERTOR.....	170
DECT – REMOTE CONTROL.....	171
CONNEXION FIXED MODEM (DTMF).....	172
MOBILE TELEPHONE – REMOTE CONTROL.....	173
GPS-RECEIVER.....	174
SERIAL CABLE– 3 POSSIBILITIES.....	175
MIDI CABLE.....	176

G. SURVEY OF ALL FUNCTIONS.....177

A. PRELIMINARY REMARKS

Introduction

The **APOLLO II** device is an electronic system designed for the automatic execution of bell functions. This comprises ringing, tolling, angelus, carillon melodies, etc. The APOLLO II device can also manage the control of the tower clock, the heating and/or the illumination, etc.

To perform these functions, the APOLLO II device is provided with a number of outputs (maximum 98). The starting up commands for the various executions can be given manually, pre-programmed, or by means of an entry (e.g. external key or remote control).

The APOLLO II device distinguishes 3 operating states:

The **main menu** provides all information concerning time, date and bell functions, which are active at that time.

In the second menu, the **user menu**, the final user can start, change and end different executions. It is also possible to switch continuous circuits and program blocks “ON/OFF”. When you are in the user menu, the numerical keys can be used for manual pealing or tolling if necessary.

The third and last menu, the **function menu**, allows the installer to define the complete installation. That way, the installer can use the function menu to program the different functions. That means that the user menu will differ from installation to installation. Both the function and the user menu can be secured by means of an **access code**.

The **TEMPORA** device offers the same possibilities as the APOLLO II device, but has fewer outputs. Both the APOLLO II and the TEMPORA device can optionally be provided with inputs without tension. Thus, the TEMPORA device has the same functions as the APOLLO II device. This instruction manual deals with both devices. Any difference between the two devices is specifically indicated.

Glossary

Internal memory: The APOLLO II device stores all its parameters and sequences in an internal memory. This memory is stored by means of a battery in the device.

External memory: The APOLLO II device has an optional external memory. This memory can be connected to the APOLLO II device by means of a **plug-in module PIM**. Depending on the position of the module, the data are then copied from the external memory in the module into the internal memory of the APOLLO II device. This external memory functions without battery – the data stay in the plug-in module until you overwrite them with other data.

Executions: The APOLLO II device distinguishes 7 types of executions. These include ringing, play sequence, play group, switch on continuous circuit, switch on program block, manual tolling and insert pauses.

Program item: By means of program items, all executions of the APOLLO II device can be

programmed. The program item determines, when what has to be executed. By means of a program block, the programs can be switched on and off by the final user

Program block: A program block can be considered as a switch (a condition) for a program item. If the execution of a program item depends on a program block, the program item will only be executed when this program block is active. **Program blocks enable you to switch on/off different executions, such as e.g. hour stroke, angelus and/or masses.**

Sequence: The APOLLO II device is completely based on the notion “sequence”. A sequence can be seen as a recording of chronological actions in the function menu. Because the function menu allows both the execution of applications (tolling, ringing, carillon melodies, continuous circuits) and the programming of the APOLLO II device, the execution of a sequence can lead to different results. That way, you can distinguish between sequences, which contain executions, such as tolling, ringing, angelus, sequences that make it possible to change parameters and sequences that provide the final user with the opportunity to modify the programs himself.

Record sequence: While recording a sequence, the different steps in the function menu are introduced in the **operating memory (buffer)**. Every time you record a new sequence, the operating memory is cleared in order to take in new steps. If you want to preserve the sequence on a permanent basis, you can store the sequence in the ***permanent memory*** by means of the function ‘*save sequence*’

Groups: In general, groups are used when the APOLLO II device serves as a carillon computer. When a group consists of different sequences, each containing one melody, it is possible to play an entire carillon concert by activating one group. Groups can also be used to play cyclical melodies. That way, you can compose a group with Christmas songs, summer songs, and autumn songs...

Bell functions: Bell functions include all bell-related actions, such as e.g. ringing, tolling, playing melodies, ...

User code: The user menu can be protected by means of a user code. This code consists of 4 digits. The code can be any number between 0000 and 9999. When the code is programmed (different from 0000), you will be asked to enter it when you want to proceed from the main menu to the user menu.

Function code: The function menu can be protected by means of a function code. Like the user code, this code consists of 4 digits. The code can be any number between 0000 and 9999. When you want to proceed from the main menu to the function menu, you first have to enter “1234” by means of the numerical keys. If the function code is set different from “0000”, you will be asked to enter the function code.

Function number: The functions of the APOLLO II device are composed via the principle of a tree diagram. That means that certain functions split up in subfunctions. These subfunctions in turn split up in several “branches”. The function number allows you to determine at which level the function is situated. For each lower level, the number grows 2 digits longer. In the function menu, the function number is always indicated above the description of the function.

B. HARDWARE

1. Different versions APOLLO II - device

There are several versions available of the APOLLO II device. These versions differ from each other in the number of outputs. The **standard version** disposes of **10 outputs**. The small version of this device has 4 outputs. It is called **TEMPORA**. The inputs on both devices are optional, low-voltage.

Versions	Apollo II-10	⊗	10 outputs
	Apollo II-16	⊗	16 outputs
	Tempora-4	⊗	4 outputs
	Tempora-6	⊗	6 outputs

Apollo II-SBSI ⇒ APOLLO II device with serial output, 2 mechanical relay outputs and 4 low-voltage inputs. This version is used for applications that require more than 16 outputs or **MIDI** (to record melodies yourself). The Apollo II-SBSI device is used in combination with a SBSI power board(s) (max. 4 in parallel), that is (are) located in the tower. To this power board, you can connect hammers, swinging motors, illumination, tower clock, ... The connection between the Apollo II-SBSI and the SBSI power board(s) is established by means of 4 wires (0,25 mm²) – **maximum 98 outputs, which can be used at will**.

APOLLO II – serial ⇒ APOLLO II device provided with serial outputs and 2 inputs. The APOLLO II – serial device is used in combination with the serial power board(s) (max. 2 in parallel), which are located in the tower. This power board is provided with 4 hammer outputs (sparkfree) and 6 relay outputs. The connection between the Apollo II – serial and the SBSI power board(s) is established by means of 4 wires – **maximum 20 outputs** of which 8 can only be used for 230V DC hammer coils.

The standard version functions with a power supply of 230V AC. On demand, we can also provide devices with 110 V AC.

2. Additional options for the APOLLO II (Tempora) device

MIDI-interface: This is only active in SBSI devices.

PIM (Programmable plug-In Module) : Module used to copy the contents of the internal memory of the APOLLO II device.

Radio Time Receiver (DCF-MSF-GPS-MASTER CLOCK SYNCHRONIZATION):
to obtain automatic radio synchronization of the device.

DTMF interface: In this option print, you can start to 10 different executions via the phone by using DTMF signals (the audio signal that is emitted when you press a telephone key). That way, you can start the executions of the APOLLO II device **via remote control**. The DTMF interface can be used with a fixed or a mobile modem.

PC-interface: to obtain data transfer between PC and the internal memory of the APOLLO II device. This communication can occur directly (with a cable) or by means of an additional medium such as a modem. The APOLLO II device transmits the contents of its display to the PC. These data are shown on the screen (e.g. the main menu). From then on, it is possible to program the parameters by means of the PC. It is also possible to transmit melodies from the PC to the APOLLO II device. That way, the APOLLO II device can be programmed by means of **remote control**.

Optional board (Apollo II): This board contains 4 low-voltage inputs and a slave clock connection (24V polarized, max. electric power 30mA).

Optional board (Tempora): This board contains 2 low-voltage inputs and a speech output (for connection with a “digital sound” option print).

3. Characteristics of the APOLLO II device (Tempora)

Power supply: 230VAC 50/60 Hz +5% / -10%
120VAC 60 Hz (upon request) +5% / -10%

Inputs:

Number:

- Apollo II : 4 (option print) or 10 (DTMF interface)
- Apollo II-SBSI : 4 (standard) or 10 (STMF interface)
- Apollo II serial : 2 (standard) or 10 (DTMF interface)
- Tempora : 2 (option print) or 10 (DTMF interface)

Tension: low-voltage (contact without tension) or via DTMF interface

Outputs:

Number:

- Apollo II: 10 or 16
- Apollo II-SBSI: max. 98
- Apollo II-serial: max. 20 of which 8 only for 230V DC hammers
- Tempora: 4 or 6

Max. output current: 3 A alternating current

Power: 6W

Housing: Polystyrene UL94-V0

Dimensions : Apollo II : 235mm x 255mm x 70mm
Tempora : 150mm x 215mm x 60mm

Weight: 1,6 kg / 3.52 pounds (Tempora : 0,85 kg)

4. Changing the battery

An internal timekeeper keeps track of the internal time of the APOLLO II device. It contains a battery that lasts up to **10 years**. The battery can be replaced as follows. Make a back-up copy of the functions by means of a PIM. **Disconnect the voltage**. Change the battery. Reconnect the device and charge the functions again through the PIM.

5. standard

The device complies with the following standards:

SECURITY STANDARD

EN 60950: This standard outlines the safety conditions of electrical information equipment. The APOLLO II device is provided with a short circuit protection. The APOLLO II device is also provided with a general switch. This switch is situated at the right-hand side and disconnects the device from power supply. **Always disconnect the device from power supply before opening it.** Within the device, there is a partition, which separates low voltage (12 VDC) from high voltage (230 VAC). **Therefore, it is very important to connect the wires for supply and for the supply circuits (all 230VAC wires) underneath. The connection wire for the Radio Time Receiver enters the device through top.** The connection terminals of the APOLLO II device cannot be used to connect wires with a cable section larger than 2,5 mm².

ELECTROMAGNETIC COMPATIBILITY

EN 50081-1 : Generic Emission Standard EN55022

EN 50082-2 : Generic Immunity Standard

EN 61000-4-2: Electrostatic Discharge (ESD)

EN 61000-4-3: Radiated Electromagnetic Field

EN 61000-4-4: Electrical Fast Transient/Burst

EN 61000-4-5: Surge Immunity

Measurements were carried out at : Technologisch Centrum v.z.w.
Laboratorium voor EMC
Oostende (Belgium)

Concerning this standard, it is important that the final user keeps some rules in mind. To prevent possible disturbances, it is important to keep the signal wires (such as the DCF-receiver connection) away from sources of disturbance like e.g. motors, computers, ... Make sure that the connecting wires of the DCF-antenna are separated from the power wires.

6. Number of software-items

Continuous circuits	10
Ringings	10
Hammers (C1 to DIS6)	64
Program blocks	24
Sequences	127
Groups	24
Sequences per group	96
User menu (items / pages)	30 / 10

Number of program items:

	single	permanent
ringing	10	90
sequences	10	100
groups	10	30
continuous circuits	10	50
blocks	10	80

7. Radio-synchronization

The synchronization takes place during the normal action (without any power cuts) between 03 and 05 am. When there is synchronization, the green led on the device will burn **continuously**. When you see that the green led on the device is burning, you know that the device has been synchronized during the night.

If you want to check how the device synchronizes, you can proceed according to the following instructions. Turn the device's supply voltage off and on. The APOLLO II device will now immediately try to synchronize. When the APOLLO II device receives no signals at all, the green led on the device will blink with the led on the antenna.

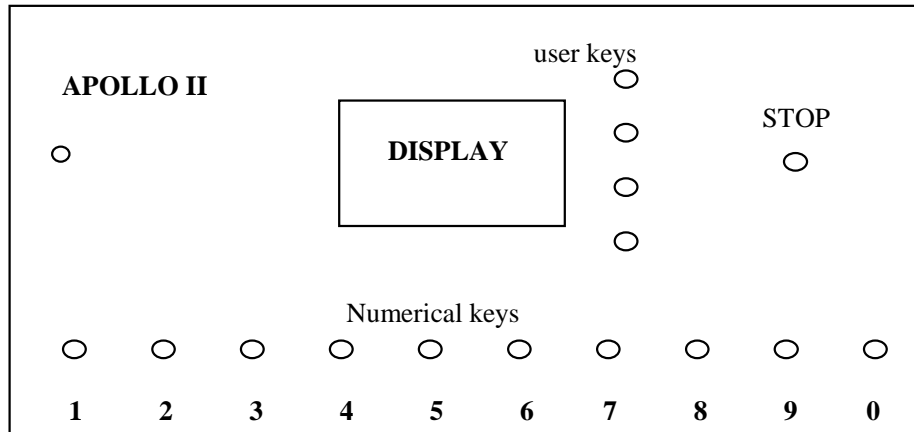
When the APOLLO II device receives synchronization signals, the green led will blink reverse to the led on the antenna. That means that the led on the device will go out, when the led on the antenna lights up and vice versa. Synchronization takes place on second 0. When the device has been synchronized, the green led on the device will burn **continuously**. When no synchronization has taken place, at 05 a.m. in the morning, the led will not burn until the device will try again to synchronize during the night.

The antenna should be installed at a distance of 80 to 150 cm from the device.

C. SOFTWARE

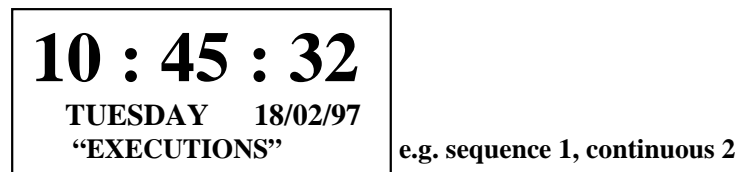
A. General

1. Front panel



2. Main menu

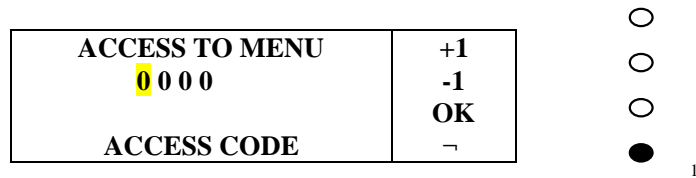
When starting up the APOLLO II device, the logo will appear. A few seconds later, you will proceed to the main menu. This menu provides information on time, date and current executions.



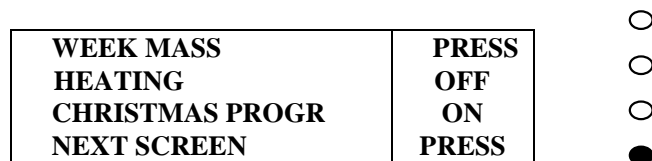
The current executions are indicated on the bottom line of the main menu, e.g. sequence 1, continuous 1. These executions are detected every 2 seconds. When different executions are active at the same time, the signal on the bottom line will switch executions every 2 seconds. Because the "ringing" executions are indicated by the led on the device, these "ringing" executions are not indicated in the display. Since the TEMPORA device is not equipped with any led, this device indicates the numbers of the active ringing executions in the display.

3. User menu

Press the bottom user key in the user menu to proceed from the main menu to the user menu,. If the user menu is secured by means of a **user code**, the system will ask you to enter this code first.



After entering the correct code and pressing “OK”, you can enter the user menu. The entered code will stay valid for about 10 minutes. It means that within this period of time, you can enter the user menu without having to enter the user code every time. Each time a key is pressed, this access time is refreshed to 10 minutes again. In the user menu, the final user can start different executions. These executions are put in the user menu by the installer.



In the user menu, the numerical keys can be used for manual tolling or for manual pealing. This function has to be programmed by the installer.

Every screen of the user menu contains max. 3 executions. By means of the 3 upper user keys, you can activate the programmed executions. Program blocks and continuous outputs automatically receive an ”ON/OFF” signal following their name to indicate whether they are active or not. By means of the bottom key, you will proceed to the next screen of the user menu (max. 10 pages). When no further executions are scheduled, you will return to the main menu.

In the user menu, the “STOP” key can be used to stop all actions with bells (bell functions). The continuous circuits and the program blocks are not stopped.

¹ ● = press here

4. Function menu

To proceed from the main menu to the function menu, you have to enter "**1234**". When a function code is introduced, the system will ask you to enter it first. Upon entering the function menu, the software and the hardware version of the APOLLO II device will appear in the display. In the function menu, a time-out of 30 minutes is applied. It means that you will automatically return to the main menu when you don't press any key for 30 minutes in this menu.

FUNCTION 01 EXECUTIONS	PRESS OFF
FUNCTION 02 USER MENU	ON PRESS

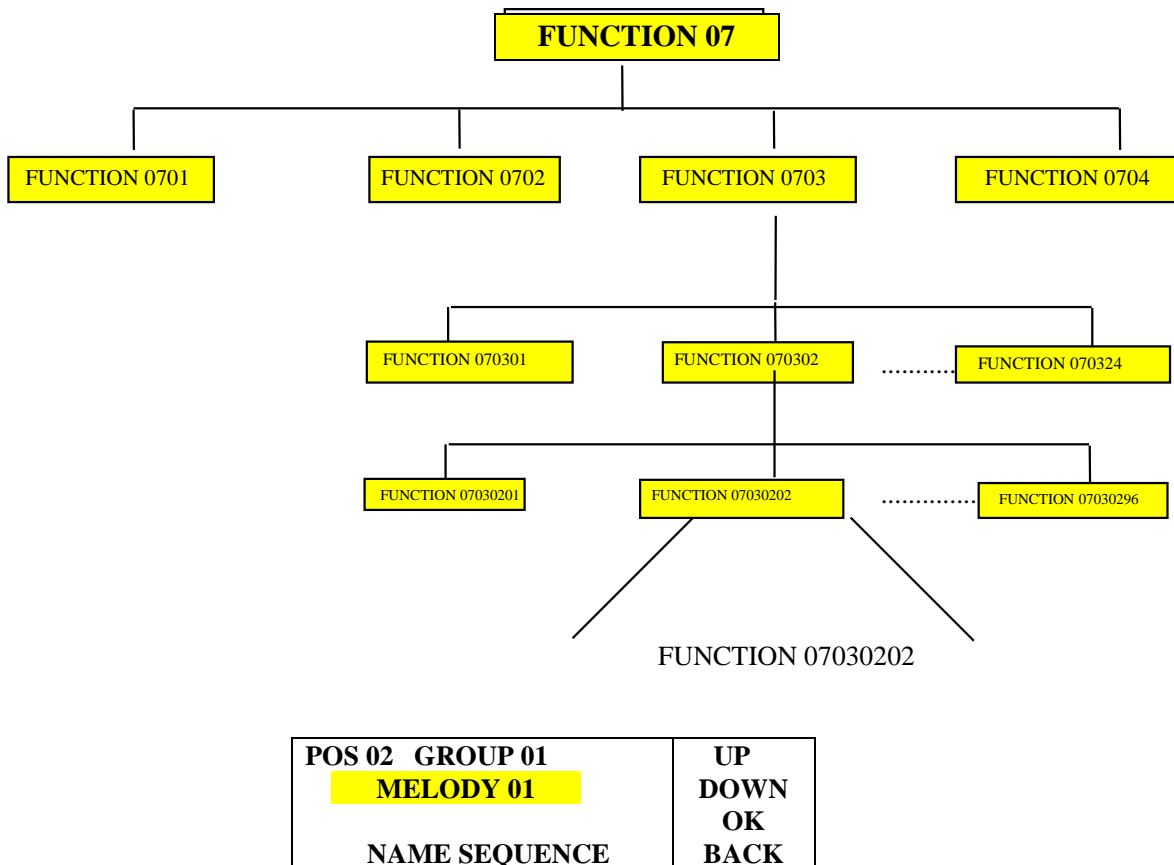
5. Structure of the functions

Each main function consists of different subfunctions. The level of a certain function can always be derived from the function number. For each lower level, the function number is elongated by 2 digits (e.g. function 070302 \Rightarrow level 3).

By pressing “UP” or “DOWN”, you can select the different subfunctions (e.g. function 0101, function 0103, ...) of a function. These subfunctions can also be selected by means of the numerical keys.

On pressing “OK”, you will proceed to the selected subfunction (**lit black**). When pressing “BACK”, you return to the previous level. When you have entered the last subfunction, you can introduce the parameters of this function.

E.g. function 07 is a main function. Function 07 (level 1) has 4 subfunctions. These are the functions 0701, 0702, 0703, 0704 (level 2). When we now proceed to e.g. function 0703, we notice that this subfunction too consists of several subfunctions: 070301,070302...070324 (level 3). When we proceed to function 070302, we notice that this function consists of 96 subfunctions: 07030201,07030202,07030296 (level 4). When we proceed to function 07030202, we finally enter a setup menu (last subfunction). Here, we can enter the required parameters.



6. Entering parameters

The function of the user keys (right-hand side) is always represented in the display.

In each final function, the description (bottom line) and the active parameter value will blink. The parameters can always be altered by means of the 2 upper user keys (at the right-hand side of the display) or by using the 10 numerical keys of the APOLLO II device. When the entered value exceeds the minimum and maximum values, it will be replaced by the minimum or the maximum value respectively. To enter the next parameter, press the “OK” key. To return to the previous parameter, press the “BACK” key.

To enter names (max. 16 characters), you can also use the keys “+1” and “-1” or the numerical keys. The different characters can be chosen quickly by means of (numerical) key combinations. The first key you press determines which series of 10 characters becomes active. This series appears on the bottom line of the display. The second key you press, determines which character from this series you select. On the next page, you will find a diagram of all key combinations. **When you are using a computer (= option) or a Workabout programming terminal, you can change the names by means of a keyboard.**

You can quit a final function in 2 different ways. All parameters you change are immediately saved. However, if you quit the function by means of the “BACK” key (when situated at the first parameter), all actions concerning this function will be cancelled. That is the case with ringing executions as well as with entering the correct time. By pressing the “OK” button (when you have come to the last parameter) to quit the function, all entered actions will be executed.

The functions to which actions are linked are:

- a) all types of **executions**
- b) introducing time, date and precision of the **internal clock**
- c) permanent program items, making use of the “Easterpr”-parameter
- d) internal memory
- e) saving and testing sequences

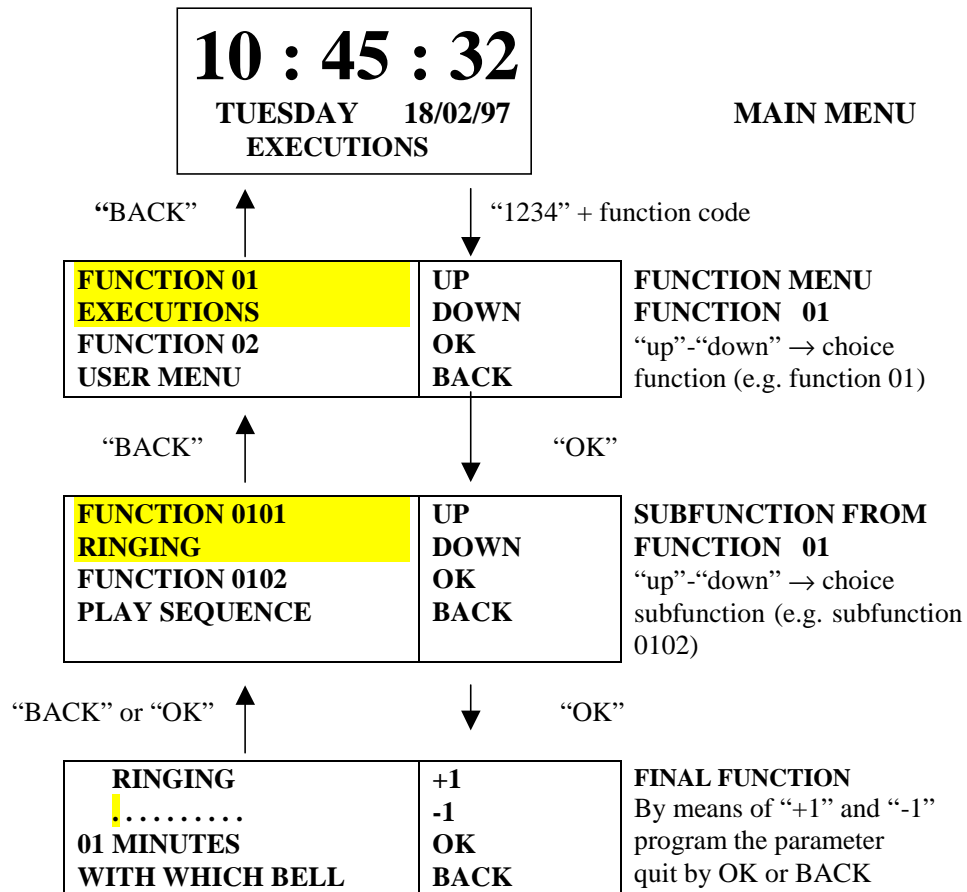
When we quit these functions with “BACK”, the entered action will not start.

7. Power failure

In case of power failure, all bell functions will be terminated internally. Bell functions are ringing, tolling and playing melodies with the bells. However, the continuous circuits and the program blocks will stay active internally. When power is restored, the continuous circuits and the program blocks will be switched on again. In that case, the duration of the power failure will be taken into account.

For example: continuous circuit switched on at 8h00 am for 4 hours ⇒ power failure at 10h00 am ⇒ The continuous circuit will stay active internally (even without power supply) ⇒ if power is restored at 11h00 am ⇒ the APOLLO II device will switch on the continuous circuit until 12h00 am.

8. Example



9. Diagram (shortened selection of characters)

A	K	U	a	k	u	l	.	á	â
---	---	---	---	---	---	---	---	---	---

A	B	C	D	E	F	G	H	I	J
K	L	M	N	O	P	Q	R	S	T
U	V	W	X	Y	Z	ã	æ	ø	
A	b	C	d	e	f	g	h	i	j
K	l	m	n	o	p	q	r	s	t
U	v	w	x	y	z	ã	ç	ñ	õ
1	2	3	4	5	6	7	8	9	0
.	,	?	!		+	-	=	<	>
Á	é	Í	ó	ú	à	è	ì	ò	ù
Â	ê	Î	ô	û	ä	ë	ï	ö	ü

e.g. X ⇒ first key 3, then key 4

e.g. 5 ⇒ first key 7, then key 5

e.g. c ⇒ first key 4, then key 3

e.g. blanco or erase character ⇒ first key 8, then key 5 or first key 3, then key 0

B. Programming the device

As a sequence can contain actions in every function of the function menu, we need to describe all other functions in this manual before explaining the function “record sequence” (function 06). The steps mentioned below are a guideline for programming the Apollo II (Tempora) device and for using this manual.

We advise you to read the description of the functions of the device at least once, before trying to program it. You will notice that recording sequences will be much easier than passing immediately to function 06.

PROGRAMMING AN APOLLO II DEVICE

STEP 01: defining relays (outputs)

Function 13: defining relays (outputs)

Example: Relays 1 till 4 : ringing bells
 Relays 5 till 8 : hammers
 Relays 9 and 10 : tower clock (1A and 1B)

STEP 02: adapting names of continuous outputs and program blocks

Function 08: adapting **names** of continuous outputs – e.g.: it is easier for the client to talk about ‘heating’ than about ‘continuous 01’. Examples of continuous outputs are: illumination, heating, carousel, doors, ...

Function 09: adapting **names** of program blocks – e.g.: it is easier for the client to talk about ‘auto. hour stroke’ than about ‘program block 01’. Examples of program blocks are: auto. hour stroke, auto. angelus, auto. ringing, ... A program block can be used as a switch to block certain programs. That way, the final user can temporarily block an automatic execution.

STEP 03: create sequences (if necessary) + programs

Programming continuous outputs (illumination, heating), ringing bells and groups can be done immediately in programs, which are provided for that purpose (**function 04:** continuous programs, ringing programs and group programs resp.). All the executions, for which you need a *sequence* (*hour stroke, angelus, funeral, tolling, bim bam, ...*), you can only program after having recorded the sequence (**function 06**).

Function 04: during the installation, only the permanent programs (function 0402) are used – if an execution can be programmed directly, you don’t have to create a sequence.

Function 06: to record sequences – sequences can be used in programs (for the automatic execution) and/or placed immediately in the user menu (that way, the final client can start the execution manually).

STEP 04: programming the user menu + inputs

Function 02: the installer determines what should appear in the user menu. That way, the final user will not find any redundant information in the user menu and the degree of difficulty for the final user will decrease dramatically.

Function 03: when you want to use the inputs (external switches, remote control, DTMF interface, ...) you can determine in function 03 what will be executed when a certain input is activated.

STEP 05: programming set-up parameters, like impulse duration of hammers, tower clock, ...

Function 05: ringing parameters

- Normal ringing or simulated ringing?
- Which hammer is installed on which bell?
- Ringing in and ringing out times: when you enter the ringing in and ringing out time for each bell, you obtain a lovely ringing pattern. The Apollo II device determines for itself at what moment the bells have to be activated in order to synchronize the time between the first clapper stroke of each bell.
- The interval when activating several bells?
- Order of ringing out – when you start ringing with the smallest bell, you can choose to switch off the smallest or the largest bell first.

Function 10: hammer parameters

- For each hammer, the impulse duration and the delay can be programmed

Function 11: special outputs

- Duration for opening the louver-holes
- Switch the Roman knell on/off
- Transpose a MIDI keyboard (only Apollo II-SBSI)
- Parameters for the SUBD9 (serial) output

Function 0606: properties sequences

- For each sequence, you can modify the playing speed. Attention: in doing this, all pauses (but also lengths), which are present in the sequence become shorter or longer.
- You can determine for each sequence, whether or not the louver-hole has to be opened.

Function 1201: parameter for the internal clock

- Programming date, time (not necessary when using an antenna)
- Function 120103: time format
- Function 120106: summer/winter time changeover

Function 1202 and 1203: parameters for tower (slave) clocks

- Programming the time for the tower (slave) clock (120201 and 120301)
- Determining the impulse duration for the tower clock (120202 and 120302)

Function 14: general settings

- Function 1401: contrast display
- Function 1402: user code
- Function 1403: function code

C. Parameters and description of the functions

1. Set-up (function 14)

1.1. Contrast display (function 1401)

The contrast of the display can be any value between 0 and 10. A higher value means more contrast (text darker), a lower value means less contrast.

1.2 User code (function 1402)

In this menu, you can enter the user code between “0000” and “9999”. Every time you want to enter the user menu from the main menu, you will be asked to enter this code. When the user code is set to “0000”, the user menu is considered unsecured. In the latter case, you can always get access to the user menu (you don’t have to enter the code).

1.3. Function code (function 1403)

The function code can be set to any value between “0000” and “9999”. From the main menu, you can enter the function menu by pressing “1234”. Then you will be asked to enter the function code, unless this code is set to “0000”. In that case, you can proceed directly to the function menu.

1.4. Internal memory (1404)

This function can alter the contents of the internal memory. There are eight different possibilities:

- “STOP EXECUTION”: All executions, program blocks and continuous are switched off.
- “CLEAR FUNCTIONS”: The parameters (e.g. programming relay, impulse duration hammers, names sequences, ...) of the internal memory will be deleted.
- “CLEAR SEQUENCES”: The sequences in the internal memory will be deleted. The names of the sequences are deleted as well.
- “CLEAR PROGRAMS”: All program items will be deleted.
- “LOAD MEMORY FROM PC→APOLLO”: To load the memory from your PC, you can start communication by means of this function. The device’s memory will be replaced by the memory from the PC. In that procedure, both the parameters and the sequences will be transferred (replaced).

“SEND MEMORY FROM APOLLO→PC”: This function allows you to transmit the internal memory of the APOLLO II device to the PC. There too, both parameters and sequences are transferred.

“LOAD PIM WITHOUT SOFTWARE”: The content of the PIM, that is connected to the device, is loaded. It is important NOT to cut off supply voltage to execute this function. The software that is always present in the PIM is not loaded. Only data that were determined when creating the PIM, are loaded. The PIM has to be connected to the APOLLO II device with the arrows pointing towards the device.

“VERIFY PIM”: The content of the PIM is compared to the data in the device. If there is a difference, the device signalises that something is different. Here, it is also important NOT to cut off supply voltage.

INTERNAL MEMORY CLEAR SEQUENCES	UP DOWN OK BACK
TRANSACTION	

1.5 Software (function 1405)

This function is aimed at executing some actions related to software. There are two options:

“SOFTWARE VERSION”: When you choose this option, the software and hardware version + possible options will appear on the display.

“LOAD SOFTWARE”: When you wish to replace the software of the APOLLO II device by means of a modem, this can be started by using this function.

SOFTWARE SOFTWARE VERSION	UP DOWN OK BACK
TRANSACTION	

1.6. Load PIM (function 1406)

You can use this function to determine what is loaded into the PIM (and what is transferred when this PIM is connected to another device).

First, the parameters in this function have to be set to the correct values. Then, the APOLLO II device has to be switched off. The PIM is connected to the device. When the APOLLO II device is switched on again, the data determined in this function are loaded into the PIM.

When this PIM is then connected to another device, the data determined in this function will be copied into that device.

There are several possibilities:

“**EVERYTHING**”: SEQUENCES, PARAMETERS as well as SOFTWARE are stored into the PIM.

“**SOFTWARE ONLY**”: Only the software will be stored into the PIM.

“**SOFTWARE + SEQ.**”: The software and sequences will be stored into the PIM. Only the sequences specified by the last two parameters in this function will be stored into the PIM (e.g. sequences 21 until 40). The names and speed of these sequences will be stored as well.

“**SOFT+SEQ+GROUP**”: The software, group contents and sequences will be stored into the PIM. Only the sequences specified by the last two parameters in this function will be loaded into the PIM (e.g. sequences 21 until 40). The names and speed of these sequences will be stored as well.

LOADING PIM SOFTWARE +SEQ. 021 040 TRANSACTION	UP DOWN OK BACK
---	--

2. Relay (function 13)

The number of relays of the clock is detected automatically. For each relay, there is a separate function (function 1302... 1316). When e.g. **function 1302** is activated, **relay 1** can be allocated. The different possibilities are:

00	NOT USED:	this relay is not used
01 to 10	RINGING 0...9:	1 of the 10 ringing relays
11 to 20	CONTINUOUS 1...10:	1 of the 10 continuous circuits
21	ROMAN KNELL:	to connect Roman knell
22	LOUVER HOLE:	to connect louver-hole
23 to 24	CLOCK 1A, 1B:	connections of 3-wire tower clock 1
25	CLOCK 1C:	extra connection for 4-wire tower clock 1
26	CATCH UP 1:	catching up of tower clock 1
27 to 28	CLOCK 2A, 2B:	connections of 3-wire tower clock 2
29	CLOCK 2C:	extra connection for 4-wire tower clock 2
30	CATCH UP2:	catching up of tower clock 2
31 to 91	HAMMER 1:C1...64:DIS6:	64 different hammers

RELAY 1 HAMM. 49 :C5 ALLOCATE RELAYS	UP DOWN ON BACK
---	--

The other relays can be assigned in the same way (function 1303 and further...).

By means of function “control relays” (function 1301), it is possible to switch off all relays

that control bell functions. Relays that control bell functions are relays used for swinging or controlling the hammers. This can be useful, when you wish to practice on the keyboard without using the bells. The led on the front panel and the texts in the display will still light up or appear. That way you can check whether a sequence is carried out correctly.

RELAYS BELLS ON SITUATION	UP DOWN ON BACK
--	--------------------------

Note:

* An APOLLO II with option print always contains 18 relays functions – the outputs for the 24V- polarised slave clock have to be programmed as follows:

- Function 1318: relays 17 -> clock 2A
- Function 1319: relays 18 -> clock 2B

* On an Apollo II-SBSI, the 2 relays on the interface are programmed as relays 97 and 98.

3. Clocks (function 12)

3.1 Internal clock (function 1201)

In this function, you can enter the time and the date that will appear in the main menu. Functions 120101 and 120102 expire when using an antenna.

3.1.1. Changing time (function 120101)

CHANGE TIME 17 03 CHANGE HOURS	+1 -1 OK BACK
---	------------------------

The internal time of the APOLLO II device is established here. Please remember that the time will only be saved when you quit function 120101 by means of “OK”.

3.1.2. Changing date (function 120102)

CHANGE DATE 10 APRIL 01 DAY OF THE MONTH	+1 -1 OK BACK
---	------------------------

Here you can enter the date of the APOLLO II device. The APOLLO II device automatically establishes the day (Monday ... Sunday). Please note that the year indication of the APOLLO II device goes from 1994 to 2093. For example, that means that 19 OCTOBER 72 refers to 19 OCTOBER 2072.

3.1.3. Time format (function 120103)

TIME FORMAT EUROPEAN SELECT CONTINENT	UP DOWN OK BACK
--	--

By selecting the continent, you can enter the time indication format.

EUROPEAN: 17:14:56
 TUESDAY 10/04/01

ASIAN: 05:14:56
 PM TUESDAY 01/04/10

AMERICAN: 05:14:56
 PM TUESDAY 04/10/01

3.1.4. Running speed (function 120104)

The running speed of the internal clock refers to calibration. When you notice that the clock is going too fast, you can lower the running speed. By raising the running speed, the clock will go faster. The standard value (middle value) is 31. The running speed is indicated in 5sec/month (if running speed > 31) or 10 sec/month (if running speed <31). Initial calibration takes place at the factory during testing. The running speed, entered during this test, is mentioned on the processor print.

RUNNING SPEED 31 IN 5 SEC / MONTH	+1 -1 OK BACK
--	--

3.1.5. Shift radio (function 120105)

When using a radio receiver, the functions 120101 to 120104 are not relevant. When, however, you use a radio receiver that was meant for another time zone than the zone you are using it in, this can be corrected in this function. E.g. DCF in England => shift radio = -1.00 HOUR

SHIFT RADIO 0.00 HR. ZONE VS. RADIO	UP DOWN OK BACK
--	--

3.1.6 Daylight saving (function 120106)

This function makes automatic summer/winter time changeover possible. We have to enter a few parameters, depending on the place where the device is installed.

DAYLIGHT SAVING FROM END MARCH UNTIL END OCT. MAKE SELECTION	UP DOWN OK BACK
---	--------------------------

There are a few possibilities:

- No automatic changeover
- From the end of March to the end of October (European)
- From the beginning of April to the end of October (American)
- From the end of October to the end of March (Australian)
- From the beginning of October to the end of March (Australian – Tasmanian)

3.2. Slave clock 1 (function 1202)

By using this function, you can enter the type of slave clock and its time.

3.2.1. Time slave clock 1 (function 120201)

TIME SLAVECLOCK1 17 03 CHANGE HOURS	UP DOWN OK BACK
---	--------------------------

When you enter this function, the slave clock (tower clock) will stop. You can check outside what time it indicates. You can enter this time value into this function. When you quit the function with “OK”, the slave clock will synchronize to the internal clock.

3.2.2. Type slave clock 1 (function 120202)

In this screen, all parameters concerning the type of slave clock can be entered.

The different parameters are:

- Type:
 “3-WIRE” (A, B, COM) **also for a 24V polarised slave clock!**
 “LAMBERT” (4-wire: A, B, C, COM) clock
 “2-WIRE (A, COM)
- Type clock:
 “12-HOUR”
 “24-HOUR”
- Type: moment, at which the APOLLO II sends a pulse to the clock
 “1/2 MIN PULSE”
 “MINUTE PULSE”
- Pulse duration s: The duration of the pulse that is sent to the slave clock in seconds.

TYPE SLAVECLOCK1 3-WIRE 24-HOUR ½ MIN PULSE 1 TYPE	UP DOWN OK BACK
---	--------------------------

3.3. Slave clock 2 (function 1203)

The parameters for slave clock 2 are entered in the same way as the parameters for slave clock 1.

These parameters can be found in the following functions:

Time slave clock 2 \Rightarrow function 120301 (see time slave clock 1)

Type slave clock 2 \Rightarrow function 120302 (see type slave clock 1)

4. Special outputs (function 11)

4.1. Louver-hole (function 1101)

The parameter in this screen determines the time, which is necessary to open the louver-hole. The actual output of the louver-hole is defined in function "control relays". The louver-hole is closed 50 seconds after finishing the execution (bell functions) or after the swinging (swinging). The parameter is indicated in seconds (0-59 seconds).

LOUVER-HOLE	+1
02	-1
DELAY (SECONDS)	OK
	BACK

Example: 2 seconds \Rightarrow 2 seconds before the program starts, the louver-hole opens – in other words, the execution starts exactly at the programmed time.

Note: in function 0606 (properties sequences), you can determine for each sequence whether or not the louver-hole has to be opened – Example: sequences where you must enter programs don't require opening the louver-hole.

4.2. Roman knell (function 1102)

The parameter for the activation of Roman knell is entered here. For Roman knell, bell 1 is rung (swinging 1), while tolling is realized on two other bells (hammer 2:CIS1 and hammer 3:D1). When there are other bells ringing besides bell 1, the option 'Roman knell' is blocked automatically.

The APOLLO II device only gives the command of the Roman knell. The pulses for the hammers have to be generated by another device, e.g. Movotron.

The time of delay to start ringing bell 1 (swinging 1), function 0501, is the same as for Roman knell.

ROMAN KNELL	UP
ON	DOWN
SITUATION	OK
	BACK

4.3. MIDI (function 1103)

This function is only effective when using the version Apollo II-SBSI.

The APOLLO II-SBSI device is able to receive and transmit MIDI-code.

Situation: When the MIDI option is switched “ON”, the MIDI OUT of the Apollo II-SBSI device is activated. For MIDI IN, the parameter does not have to be switched “ON”. The MIDI IN of the device is always active, regardless of the position of this parameter.

Midi Transpose: this parameter allows you to shift the bells on the electronic keyboard. That way, you can make sure that you always play in the middle of the keyboard. The MIDI transpose is indicated in half tones. If you want to shift an entire octave, you can do this by choosing – 12 or + 12.

MIDI	UP
ON 0	DOWN
SITUATION	OK
	BACK

Pay attention to the fact that the function is also present in the other devices (not Apollo II-SBSI), without having any influence (the MIDI doesn't come in or out).

4.4. Speech (function 1104)

This function only concerns the Digital Bell devices (DBTempora or DBAPOLLO II). When you want to start a melody in the Digital Bell device, you can do it in this function (of course, you can also use this function while recording a sequence). Next to choosing which melody you start, you can also store the duration of this melody (minutes, seconds) in the sequence. This can be important for the programming with ‘END’ (cf. function 040202).

SPEECH	UP
WESTMINSTER 1/4	DOWN
00 00	OK
START MELODY	BACK

4.5. Serial output (function 1105)

In this function, the specifications of the serial interface can be set. The serial interface is always connected via the SUBD9 connector (PIM connector). The serial interface can be used for:

- * the connection with the PC (APOLLO II communication center)
- * the connection of a modem for the remote control of the device
- * the synchronization of computer networks
- * the connection of the DTMF interface (for the DTMF remote control)

SERIAL OUTPUT	UP
19.200 03 VOICE	DOWN
REMOTE CONTROL	OK
BAUDRATE	BACK

Baudrate: to be able to communicate with the connected application, it is necessary that the same baudrate is connected to the APOLLO II device and to the application (modem, PC ...). Possibilities: 1.200 - 2.400 - 4.800 - 9.600 - 19.200 - 38.400 or 57.600

Ringers (only with telephone applications, such as DTMF or modem) – enter the number of rings before the APOLLO II picks up the phone.

Answer phone (only necessary with telephone applications with fixed modem) – in contrast with a DTMF interface, a fixed modem (with incorporated DTMF interface) can't detect if you phone via data (for programming via PC) or voice (for starting up an execution) via the telephone keys. In this function, the installer has to set it himself. It is possible to make sequences that guarantee the automatic switching of both systems.

Type of message : as already mentioned, the serial interface can be used for several applications. Choose the application you want to use it for in this parameter:

- Remote control: connection with the APOLLO II Comm. Center, Modem or DTMF interface
- NMEA message: for the synchronization of PC networks via the NMEA protocol
- Time message s: for the synchronization of PC networks and/or slave clocks – the message is emitted every second
- Time message m: for the synchronization of PC networks and/or slave clocks – the message is emitted every minute

5. Hammers (function 10)

In this function, the minimum pulse duration, the maximum pulse duration and the delay for each hammer are entered. Function 10 consists of 64 subfunctions (hammer 1:C1 to hammer 64:DIS6). In order to enter a parameter for a certain hammer, you have to select the hammer first.

FUNCTION 1001	UP
HAMMER 1:C1	DOWN
FUNCTION 1002	OK
HAMMER 2:CIS1	BACK

Select the right hammer, e.g. HAMMER 44:G4 (function 1008)

HAMMER 44:G4	UP
020 020 000	DOWN
MINIMUM 1:C1	OK
MIN. PULSE 1/100S	BACK

The min. and max. pulse duration are measured in 1/100 seconds. **In most cases, the two are set to the same value.** Max. = 127 = 1.27 seconds.

Because a large hammer reacts slower than a small hammer, you have to enter a delay for the smaller hammer. When we now toll with the two hammers at the same time, the two bells will sound together. That delay is also measured in 1/100 seconds. Max = 127 = 1,27 seconds.

To test the hammers, two extra parameters have been added. When we quit the function, we will always hear one stroke of the hammer, for which we have just entered the parameters. In our example this is hammer 44:G4.

“test hammer”: this parameter allows you to choose between testing at minimum or at maximum pulse duration when leaving the function.

“bells together” : this parameter allows you to activate a second hammer together with the hammer, which is programmed in this function. That way, you can easily introduce the delay of the smallest hammer. That is necessary for setting up the hammers in order to achieve a perfect timing for playing chords. The previous parameter (test hammer) also applies to the “bells together” hammer. That means that both hammers will be tested at “minimum” or at “maximum” pulse duration.

Procedure:

- 1) Make the largest and the smallest hammer strike (the bells) together. This can be done by means of the two testing parameters. Test both at minimum and at maximum pulse duration (if these are not the same – test hammers = “minimum” or “maximum”). To do so, open the function of the **smallest** hammer and enter the largest hammer in “bells together”. When you quit this function, the two bells will sound at the same time.
- 2) Increase the delay of the smallest hammer until both hammers strike at the same time.
- 3) Repeat this procedure with the second smallest hammer. Open the function of the second smallest hammer and enter the largest hammer in “bells together”. Increase the delay of the second smallest hammer until both hammers strike at the same time.
- 4) Repeat this procedure with all other hammers.

6. Program blocks (function 09)

This function is only used to ascribe names to the different program blocks. As there are 24 different program blocks available, the function 09 comprises 24 different subfunctions (0901-0924).

FUNCTION 0901	UP
PROGRAM BLOCK 01	DOWN
FUNCTION 0902	OK
PROGRAM BLOCK 02	BACK

Select the program block you want to ascribe a new name, e.g. program block 19 (function 0919). The bottom line shows the diagram 'AKUaku1.â' (see diagram page 15) from which you can choose a character. For example, when you want to select 'G', you have to press the 1st key (characters A to K). Next, you press the 7th key to select G.

PROGRAM BLOCK 19	UP
CHRISTMAS PROG.	DOWN
	OK
AKUaku1.â	BACK

7. Continuous outputs (function 08)

This function is identical to the previous function. The function is used to ascribe the names of the continuous outputs. As there are 10 different continuous outputs available, function 08 comprises 10 subfunctions (0801-0810).

FUNCTION 0801	UP
NAME CONTIN. 1	DOWN
FUNCTION 0802	OK
NAME CONTIN. 2	BACK

Select the continuous output that you want to give a new name, e.g. continuous output 7 (function 0807).

NAME CONTIN. 01	UP
HEATING	DOWN
	OK
AKUaku1.â	BACK

This procedure can be repeated for all continuous outputs.

8. Parameters for ringing (function 05)

By using this function, you can enter the parameters for each ringing bell. The actual programming or the execution of ringing bells is not done in this function. The ringing parameters are necessary for a correct ringing (example: 3 seconds after every first clapper stroke, another bell is added).

8.1. Individual ringing (functions 0501-0510)

The APOLLO II device has 10 ringing functions (ringing 0 - ringing 9). For each ringing function, there are separate parameters. For example: let us enter the parameters of ringing 3. To do that, we have to use function 0503. That way, you can enter into the following screen.

RINGING 3	+1
30 30 3:D1 060	-1
NORMAL RINGING	OK
TIME RING. IN S.	BACK

We distinguish 5 parameters :

- a) **Time ring. in s** : This parameter indicates the time between the moment the ringing command is given and the first clapper stroke. Technically, this is the time between the closing of the relay and the first clapper stroke. The parameter is indicated in seconds (0-99). This parameter is also used to establish the delay of the Roman knell option.
- b) **Time ring. out s** : This parameter indicates the time between the opening of a relay (end of command) and the last stroke of the clapper. The ringing out time is indicated in seconds (0-99). This parameter is also used to establish the speed reduction of the function 'louver-hole'. The duration of hammer blocking is **1.5 times** the ringing out time.
- c) **Pitch** : The APOLLO II device has to know which hammer should be blocked during the ringing function (here: ringing 3). Therefore, the hammer (pitch) has to be set up in this function. The duration of hammer blocking is 1.5 times the ringing out time. This parameter is also important for simulated ringing (see point e). In this function, you can determine which hammer is activated when key 3 is used during manual tolling.
- d) **Strokes/minute** : This parameter is only used for simulated ringing. It indicates the number of strokes per minute. For normal ringing, this parameter is ignored.
- e) **Type ringing**: There are 3 different types of ringing
 - 1) **normal ringing** : there is a continuous signal on the relay output
 - 2) **simulated ringing** : when there are no swinging bell motors, ringing can be simulated by means of hammers. For this option, you have to select the hammer to be used (see c) and the number of strokes/minute (see d). The parameters "time ring in", "time ring out" also apply to simulated ringing, but are mostly set to 0.
 - 3) **sim rot. ringing**: is almost identical to simulated ringing. The most important difference is that the time between two strokes is not always identical. There are always 2 strokes, following each other within a short period of time, with longer pauses between these pairs.

8.2. Joint ringing functions: ringing order (function 0511)

When several ringing actions are started up at the same time, the parameters in this menu will be checked. You have to enter 2 parameters.

RINGING ORDER	+1
02 SAME	-1
TIME BETWEEN SEC	OK
	BACK

- 1) **Time between sec**: This parameter determines how much time there is between the start of each ringing bell . However, this is only the case when the bells are started together in the ringing function. The bells are started up in the order in which they were entered. How to start ringing bells, is explained in the next chapters (function 0101).

E.g.: you start ringing 1, ringing 2 and ringing 3 and the parameter 'time between sec' is set to 2).

Notice that the first bell to start ringing, is bell 1 \Rightarrow 2 seconds later the first stroke of bell 2 is heard, another 2 seconds later the first stroke of bell 3 is heard.

Max. : 99

2) **Order ring. out**: Here you can determine how the ringing bells should stop. When you select "same", the ringing out order will be identical to the ringing in order. When you select "inverse", the bells will ring out in inverse order. The time between the last strokes is determined by "time between sec". For example, you ring with bells 1, 2 and 3 for 5 minutes. When you entered "same", bell nr 1 will stop first, followed by 2 and 3. In case you selected "inverse", which is the opposite order of the "ringing in order", bell nr 3 will stop first, followed by 2 and 1.

9. Executions (function 01)

Because these executions are usually used to record sequences, we will first give you an overview of the possible executions.

The APOLLO II device can execute 7 types of actions.

These 7 actions are : **ringing, play sequence, play group, switch on contin, program block, manual tolling, pause.**

9.1. Ringing (function 0101)

When you enter function 0101, you will see the following data on the screen:

RINGING	+1
2 4 3	-1
18 1/10 MINUTE	OK
WITH WHICH BELL	BACK

You can establish 3 parameters:

1) With which bells

In the second line of the display (2 4 3), you can select the bells with which you want to execute ringing. In this example, we have selected ringing 2, ringing 4 and ringing 3. You can also determine the ringing order. In our example, ringing bell 2 will start first, then ringing bell 4 and ringing bell 3. Please note that the ringing parameters (see chapter 8) also determine the closing and opening time of the relays.

2) Number (units of time)

Indicates the number of units the ringing action will take up, e.g. 05 minutes. The duration of the action always depends on the number and the type of unit (parameter 3). The number can

be set to any value between 1 and 59.

3) Unit

The different units in the “ringing” function are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours.**

e.g. 18 1/10 minute: the duration of the “ringing” action is $18 \times (60 \text{ seconds} : 10) = 18 \times 6 \text{ seconds} = 108 \text{ seconds}$. As the number is limited to 59, it was not possible to enter 108 seconds. Therefore, we had to enter the duration in 1/10 minutes.

9.2. Play sequence (function 0102)

This function is used to start sequences. The APOLLO II device has 5 internal channels that can be used to start sequences. That means that a different channel is used for each sequence that is started up. When all 5 channels are in use, the new sequence will not be started up anymore.

PLAY SEQUENCE	UP
SEQUENCE 001	DOWN
05 MINUTES	OK
NAME SEQUENCE	BACK

You can distinguish 3 parameters:

1) Name sequence

Here, you can select the sequence you want to play. You can start any of the 127 possible sequences.

Note for use with carillon:

It is also possible to start “seq. from group”. In that case, the next sequence of the active group will be started up (see chapter on groups). As in chapter “groups”, you can see that the APOLLO II device keeps a counter (active group). This counter keeps track of which sequence of the group will be started up next. E.g. parameter “active group” (see groups) = 3. That means that the third sequence of the “active group” will be played. When the time has passed, the sequence that is played on that particular moment will end. When a power failure occurs, the counter will not increase, so that the next melody from the group will start when the power is switched on again. The group can contain melodies for an automatic carillon concert or for cyclic melodies (each quarter a next melody from the group).

2) Number (units of time)

Indicates the number of units the sequence will be started, e.g. 01 hour stroke. The duration of the action always depends on the number and the type of unit (parameter 3). The number can be any value between 1 and 59.

3) Unit

The different units in the function “play sequence” are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours, hour stroke, next hour, times.**

The units **seconds, 1/10 minutes, minutes, 1/10 hours, hours** determine the time span of the sequence. When the sequence is finished, it will be repeated immediately (without any

pause). When the time has past, the sequence will be completed. After that, the sequence will not be repeated anymore.

The units **next hour, hour stroke and times** determine the number of times the sequence has to be played.

For example: it is now 17 h. 23 min. 40 seconds

Some possibilities:

01 hour stroke \Rightarrow 1 x 5 (hour stroke at 17 o'clock) times \Rightarrow the sequence is played 5 times

02 hour stroke \Rightarrow 2 x 5 (hour stroke at 17 o'clock) times \Rightarrow the sequence is played 10 times

02 next hour \Rightarrow 2 x 6 (hour stroke at 18 o'clock) \Rightarrow the sequence is played 12 times

02 times \Rightarrow the sequence is played 2 times

Remark: When you want to repeat the sequence a number of times, it will always be the same sequence that is repeated. That is also the case when the option "seq. from group" is selected. When you use the function **next time**, the next sequence from the group will be played (and possibly repeated).

9.3. Play group (function 0103)

This function is used to start groups (see chapter on groups).

PLAY GROUP	UP
GROUP 01	DOWN
01 TIMES	OK
NAME GROUP	BACK

You can distinguish 3 parameters:

1) Name group

Here you can select the group you want to play. You can start any of the 24 possible groups. Example: Christmas songs, spring melodies.

2) Number

Indicates the number of units the group will be started, e.g. 01 time. The duration of the action always depends on the number and the type of unit (parameter 3). The number can take up any value between 1 and 59.

3) Unit

The different units in the function "play group" are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours, times, active.**

The units **seconds, 1/10 minutes, minutes, 1/10 hours, hours** determine the time span of the group. When the unit "active" is selected, the selected group (here "group 01") will be activated. This will be explained in chapter 11.

The unit “**times**” determines how many times the group has to be played.
 02 times ⇒ the group will be played 2 times.

9.4. Switch on continuous (function 0104)

This function is used to switch on continuous circuits.

SWITCH ON CONTIN	UP
CONTINU 01	DOWN
02 WEEKS	OK
NAME CONTINUOUS	BACK

You can establish 3 parameters:

1) Name continuous

With this, you can select the continuous circuit that has to be switched on. You can select any of the 10 possible continuous circuits. Example: Heating, Illumination, ...

2) Number

Indicates the number of units the continuous circuit will be switched on, e.g. 01 time. The duration of the action always depends on the number and the type of unit (parameter 3). The number can take up any value between 0 and 59.

3) Unit

The different units in the function “switch on continuous” are: **1/10 sec., seconds, 1/10 minutes, minutes, 1/10 hours, hours, days, weeks, permanent, off.**

The units **1/10 seconds, seconds, 1/10 minutes, minutes, 1/10 hours, hours, days, weeks** determine the time span of the continuous circuit. The unit **permanent** is used to activate the continuous circuit permanently. In that case, the parameter “number” is not relevant. Choosing "**OFF**" turns off the continuous circuit.

9.5. Program block (function 0105)

This function is used to switch on the program blocks.

PROGRAM BLOCK	UP
PROGRAM BLOCK 01	DOWN
05 PERMANENT	OK
NAME PROGR.BLOCK	BACK

You can establish 3 parameters:

1) Name program block

With this, you can select the program block you want to switch on. You can select any of the 24 possible program blocks. Example: auto hour stroke, auto angelus...

2) Number

Indicates the number of units the program block will be switched on, e.g. 01 time. The duration of the action always depends on the number and the type of unit (parameter 3). The number can take up any value between 0 and 59.

3) Unit

The different units in the function “program block” are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours, days, weeks, permanent, off, hours susp., days susp.**

The units **seconds, 1/10 minutes, minutes, 1/10 hours, hours, days, weeks** determine the time span of the program block. The unit **permanent** is used to activate the program block permanently. In that case, the parameter “times” is not relevant. Choosing **"OFF"** turns the program block continuous off. With the units **hours susp.** and **days susp.**, you can block the program block for a while. **After the programmed time, the original value will be valid again.** You can use this for example to block the hour stroke at Easter for three days = 03 (AUTOM. HOURSTROKE 03 DAYS SUSP.).

9.6. Manual tolling (function 0106)

This function is used to strike the bells manually by means of the 10 numerical keys. For this function, it is necessary to allocate the 10 numerical keys to a certain hammer.

That is done in the function “ringing” (function 05). Each ringing circuit has its own pitch. It is this pitch that will determine with which hammer you strike when one of the 10 numerical keys is keyed in.

The pulse duration for each hammer can be entered in the function “hammers” (function 10).

When you have an Apollo II-SBSI device, the option MIDI also allows you to use an external keyboard to strike the bells. In that case, each key is automatically linked to the correct hammer. At the same time, you can send external MIDI codes to the clock. To quit this function, press the “OK” key.

MANUAL TOLLING PRESS KEY 1® 0 OR KEYBOARD	UP DOWN OK BACK
--	--

9.7. Pause (function 0107)

With this function you can insert a pause. The function is only used for recording sequences. That procedure will be discussed later.

PAUSE 05 1/10 MINUTES NUMBER	UP DOWN OK BACK
---	--

You can enter 2 parameters:

1) Number

Indicates the number of units the pause will last, e.g. 05 1/10 minutes. The duration of the action always depends on the number and the type of unit (parameter 3). The number can be any value between 1 and 59. The value 05 1/10 minutes corresponds to (5 x (60:10) seconds) = 30 seconds.

2) Unit

The different units in the function “pause” are : **1/100 seconds, 1/10 seconds, seconds, 1/10 minutes, minutes, 1/10 hours, 1/10 sec.hour, sec.hour st.**

The units **1/100 seconds, 1/10 seconds, seconds, 1/10 minutes, minutes, 1/10 hours, 1/10 sec.hour, sec.hour st.** determine how long the pause has to last. The options 1/10 sec.hour and “sec.hour st.” are important when you want to include a pause which depends on the “hour stroke”, e.g. when you want an hour stroke repetition which starts 20 seconds after the hour stroke.

10. Sequences (function 06)

A sequence can be seen as the recording of chronological actions in the function menu. That means that it is a kind of self-made function that consists of standard functions of the APOLLO II device. A sequence is used to start certain actions or change parameters of the APOLLO II device. The following explanation of the different sequence functions will enable you to fully understand this. The different actions for making new sequences are: record sequence, test sequence, save sequence and name sequence. In addition, it is also possible to change a few characteristics of the sequence. By using the last function (delete sequence) you can delete an existing sequence from the memory.

You can distinguish 3 different types of sequences:

* type 1: sequences with executions – these sequences **only** contain executions such as ringing, tolling, pauses... and can be used to start manual executions or in permanent programs, e.g. angelus, week mass, funeral...

* type 2: sequences with programs – these sequences are used to enter or change punctual or permanent programs, such as progr. week mass, progr. funeral.

* type 3: sequences with parameters - these sequences are used to change parameters, such as change time, change hands ...

Important to know is that the APOLLO II device does not make any distinction between these 3 types of sequences. They are all recorded in the same function – only the content and the procedure are different. Examples of type 2 and 3 of the sequences, are listed at the end of the manual – examples of type 1 sequences are described in the following pages.

10.1. Record sequences (function 0601)

Objective of the function: this function can be considered as one of the most important functions of the APOLLO II device. By starting up the recording of a sequence, the APOLLO II device will store all further actions you carry out until you stop the recording.

Contents of a recorded sequence: all actions that can be carried out in the **function menu** can be taken up in a new sequence (the one that is being recorded) during recording.

Remark: pay attention to the fact that there is a difference between recording the parameters by means of “+” and “-” and recording by using the numerical keys. The use of “+” and “-” can be used to raise (7,6,5,4,3...) or lower (3,4,5,6,7...).

Establishing the parameters by using the numerical keys allows you to enter an exact value for the parameter (e.g. always 5). If you are not sure, it is better to use the numerical keys. It does not often happen that you want to obtain a reduction or an augmentation.

Using the function: when you enter function 06, you will see the following data on the screen.

FUNCTION 0601 RECORD SEQUENCE	UP DOWN
FUNCTION 0602 TEST SEQUENCE	OK BACK

By selecting function 0601 (lit black) and pressing the “OK” key, you can enter the function menu.

FUNCTION 060101 EXECUTIONS	UP DOWN
FUNCTION 060102 USER MENU	OK BACK

All actions that are carried out at this instant will be recorded. Suppose you want to record the following sequence:

Example 1 : Week mass (ringing 1 for 2 minutes)- sequence type 1



As you can see, we want to record a small sequence with 2 minutes ringing on bell 1. The advantage of such a sequence is that it can be stored in the user menu or in the outputs. That way, the final user only has to press on 1 button to start this execution.

Procedure:

Start recording a sequence: as mentioned before, this is done by entering 0601 and pressing “OK”. We now enter the screen with the main functions.

FUNCTION 060101 EXECUTIONS	UP
FUNCTION 060102	DOWN
USER MENU	OK
	BACK

When you want to start an execution (ringing) in this sequence, you press “OK” again. The following is displayed:

FUNCTION 06010101 RINGING	UP
FUNCTION 06010102	DOWN
PLAY SEQUENCE	OK
	BACK

Now you can enter this function with “OK”, where the following is displayed:

RINGING	+1
!.....	-1
01 MINUTES	OK
WITH WHICH BELL	BACK

You enter the following data:

RINGING	+1
1.....	-1
02 MINUTES	OK
WITH WHICH BELL	BACK

Confirm with “OK”.

FUNCTION 06010101 RINGING	UP
FUNCTION 06010102	DOWN
PLAY SEQUENCE	OK
	BACK

At this moment, you have all the data required for this sequence. You finish the recording by pressing on the ‘BACK’ button until function 0601 is displayed again.

FUNCTION 0601 RECORD SEQUENCE	UP
FUNCTION 0602	DOWN
TEST SEQUENCE	OK
	BACK

Note: if you want the name of this sequence to appear on the screen for as long as the sequence lasts, you can enter a pause of 2 minutes after keying in the ringing and before ending the recording.

You can test this recording by means of function 0602 to verify if it meets the requirements.

10.2. Test sequence (function 0602)

When entering function 0602, the following data is displayed:

TEST SEQUENCE	UP
RECORD	DOWN
01 TIMES 0	OK
NAME SEQUENCE	BACK

Here you can distinguish 4 parameters.

1) Name sequence

Because we can also test existing sequences, we have to select the sequence we want to test. We want to test our ringing sequence. That is why we fix this parameter on "RECORD". That means that we want to test the last sequence we have recorded (ringing with bell 1 for 2 MINUTES).

2) Number

Indicates the number of units the test will take up, e.g. 01 time. The duration of the test always depends on the number and the type of unit (parameter 3). The number can take up any value between 1 and 59.

3) Unit

The different units in this function are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours, hour stroke, next hour, times.**

4) Deceleration (in seconds)

With this, we can check the recorded sequence in slow motion. That means that we can see what we have entered, e.g. how we have entered function 0106 and played our melody. The deceleration can be any value between 0 and 9. Number 5, for example, indicates that all our actions will be repeated with a pause of 5 seconds between each individual key. Please note that all pauses are indicated, but that they are not explicitly executed. For example: you have entered a pause of 1 hour. This pause will be indicated in the display, but the device will not wait for 1 hour. This gives a distorted view of the sequence. In order to see the exact timing, you should test the sequence at a deceleration of "0". Remark: **When the deceleration is not zero, the sequence will only be executed once, regardless of the values entered in the parameters "number" and "unit".**

Test the sequence you recorded. Try this with deceleration "3" and then again with deceleration "0". If you are satisfied with the result, you can save this sequence as one of the 127 possible sequences.

10.3. Save recording (function 0603)

When you enter function 0603, you will see the following data on the screen:

SAVE RECORDING SEQUENCE 001	UP DOWN OK BACK
NAME SEQUENCE	

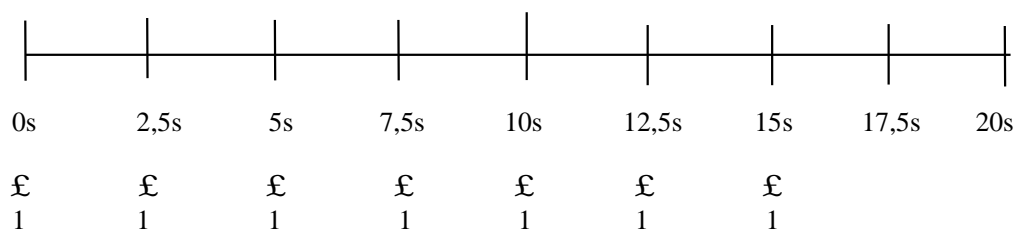
The recording can be stored in an empty sequence or in a sequence that is already used. When the sequence has already been used, its contents will be overwritten. The sequences can also be grouped, e.g. melodies (sequences 01 to 10), angelus (sequence 11), hour stroke (sequence 12), ... The APOLLO II device looks for the first empty sequence in its memory and shows this on the screen when we proceed to this function. As this is our first recording, this will be “sequence 001”. Of course, we can choose any other sequence to store our recording, but make sure not to overwrite any sequences you might still need. Press “OK” to store the sequence. Press “BACK” to quit the function without saving the sequence. To know the contents of all the sequences, you can change the name of “sequence 001”. This can be done in this function or in the next one.

After choosing in which sequence we want to store our recording, the APOLLO II device will ask you to enter a name for this sequence. In our example, we could enter a new name for our sequence. When you overwrite an existing sequence, you can quit the name unchanged by entering **16 blanks**. Because the memory needs to be rearranged after overwriting a sequence, this procedure will take about 20 seconds. During this time, all executions will be blocked.

Now this sequence has been saved, you can later use it in sequence programs, in the user menu, in the outputs...

Example 2 : stroke bell 1 (for hour stroke)

– The symbol £ indicates the stroke; the number underneath indicates the bell.



Note that this sequence is composed of a number of strokes with always the same pause (2,5 seconds) between them. Therefore, it is sufficient to record one stroke with one pause and then repeat this 7 times. We will follow this procedure.

Start the recording of a sequence: you do this, as mentioned before, by entering function 0601 and pressing “OK”. That leads you to the screen with the main functions.

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Now you can record the unique stroke in function 0106 (manual tolling). To do this, you have to enter function **06010106**. In doing this, you can see the following data on the screen:

MANUAL TOLLING	
PRESS KEY 1® 0	
OR KEYBOARD	OK
00 : 00 00 : 01.4	PA+OK

On the bottom line, you see two different counters. The left counter indicates the total duration of the sequence (which is still 0 at the moment). The recording of the melody starts when we press the first key. The right counter indicates the time span after the last stroke (or, when you haven't entered any strokes yet, the span after starting up the sequence).

Now you can enter your stroke by pressing on the right numerical key. Because you want to strike bell 1, you have to press here numerical key '1'. When you have entered the key, there are 2 possibilities to quit this function. By pressing "PA+OK", the pause between the stroke and the moment that you quit the function will be recorded. When you don't want to insert this pause, you press "OK". Because we want exactly 2,5 seconds between the strokes, we quit the function by means of "OK".

In both cases, you will see the following data on the screen.

FUNCTION 06010105	UP
PROGRAM BLOCK	DOWN
FUNCTION 06010106	OK
MANUAL TOLLING	BACK

Now you can enter the pause manually. To do this, move to function 06010107 (pause). You can do this by pressing "DOWN" one time or "UP"6 times.

With this function, you can enter pauses. A pause of 2,5 seconds can be entered as follows:

PAUSE	UP
25 1/10 SECONDS	DOWN
NUMBER	OK
	BACK

At this moment, you have entered everything that was required. Finish the recording by pressing the "BACK" key until you see function 0601.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

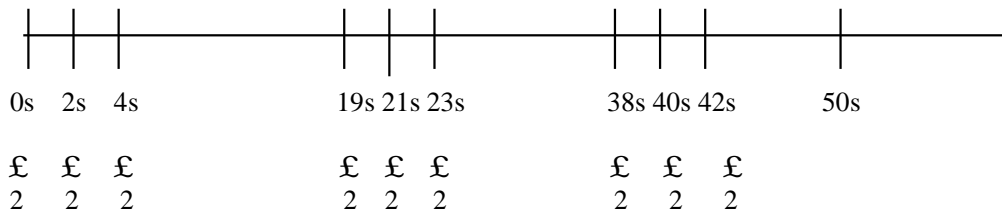
You can test the sequence (function 0602 – see 10.2). When you want to see the pause of this sequence, it's better to start the recording several times:

TEST SEQUENCE	UP
RECORD	DOWN
05 TIMES 0	OK
NAME SEQUENCE	BACK

When the recording is correct, you have to save it (function 0603 – see 10.3) before recording a new sequence.

Example 3 :Angelus (strokes on bell 2 + ringing on bell 1)

– The symbol £ indicates the stroke; the number underneath indicates the bell.



RINGING 1

The angelus consists of 3x3 strokes on bell 2, followed by one minute of ringing on bell 1.

Start the recording of a sequence: as mentioned before, this is done by entering function 0601 and pressing “OK”. That leads you to the screen with the main functions.

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Press on “OK” to enter the function “EXECUTIONS”. In the function “MANUAL TOLLING” (06010106), you can strike 9 times at once (with the exact timing for the angelus). In doing this, you can see the following data on the screen:

MANUAL TOLLING	
PRESS KEY 1® 0	
OR KEYBOARD	OK
00 : 00 00 : 01.4	PA+OK

Because you want to strike on bell 2, you have to press key “2” 9 times. Wait to press the next key until the desired pause after the previous stroke has been reached.

On the bottom line, you see two different counters. The left counter indicates the total duration of the sequence (at the moment it is still 0). The recording of the melody starts when we press the first key. The right counter indicates the time span after the last stroke (or, when you haven't entered any strokes, the span after starting up the sequence).

When you have entered all the strokes, there are 2 possibilities to quit this function. By pressing "PA+OK", the pause between the stroke and the moment that you quit the function will be recorded. When you do not want to insert a pause, you press "OK". We quit the function by 'OK' and will add the pause manually.

In both cases, you see the following data on the screen:

FUNCTION 06010105	UP
PROGRAM BLOCK	DOWN
FUNCTION 06010106	OK
MANUAL TOLLING	BACK

Now you can enter the pause manually. To do this, move to function 06010107 (pause). You can do this by pressing "DOWN" one time or "UP" 6 times.

With this function you can enter a pause. A pause of 8 seconds is required between the last stroke and the beginning of the ringing. Choose this function with the "UP" or "DOWN" key.

PAUSE	UP
8 SECONDS	DOWN
NUMBER	OK
	BACK

After having entered the pause, you can start the minute ringing with bell 1. You have to do this in function 06010101 (ringing).

RINGING	+1
1.....	-1
01 MINUTES	OK
WITH WHICH BELL	BACK

Now all the data required in this sequence has been entered. You end the recording by pressing on the 'BACK' key until you are in function 0601.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

You can test your sequence (function 0602 – see 10.2):

TEST SEQUENCE	UP
RECORD	DOWN
05 TIMES 0	OK
NAME SEQUENCE	BACK

When the recording is correct, you have to save it (function 0603 – see 10.3) before recording a new sequence.

10.4. Survey record sequence type 1

On the next pages you find a schematic survey on how to create new simple sequences (record, test, save).

Procedure:

Start the recording of a sequence. To do this, move to function 0601. In this sequence we only record executions. Therefore, we have to move to the function “executions”. Press the “OK” button twice (STEP 1A and STEP 1B).

2) Now we can establish all the executions step by step (STEP 2). You find examples, included under the headings ‘week mass’, ‘hour stroke’ and ‘angelus’ (from page 40). When you have entered all the executions, you have to finish the recording by means of the “BACK” key, until you see the function 0601 “record sequence” (STEP 3A and STEP 3B).

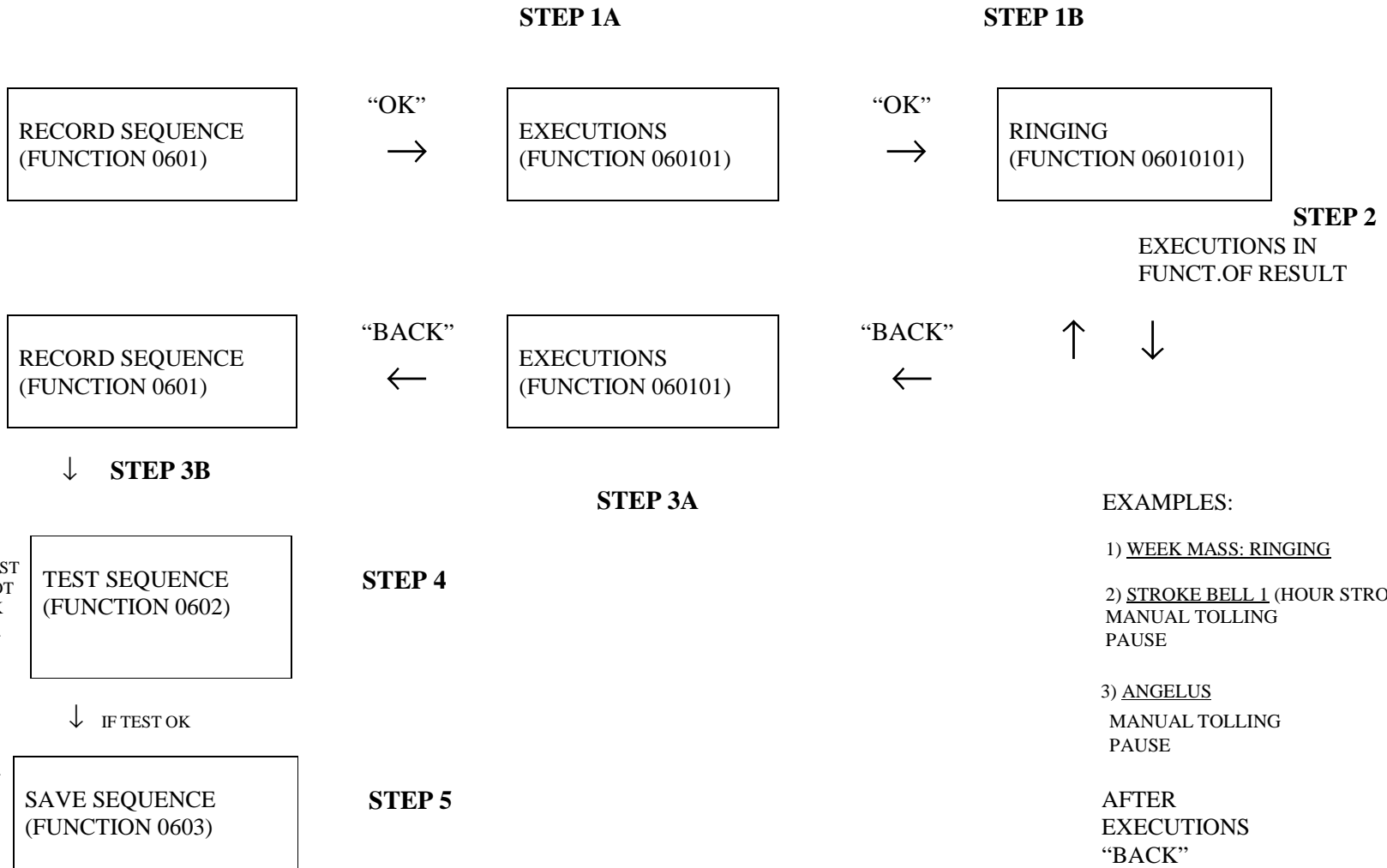
- 3) At this moment, you can test your recording in function 0602 – test sequence (STEP 4).
- When the test has passed correctly, you have to save the recording in function 0603 (“save sequence”). Afterwards, you can record a new sequence (STEP 1A and STEP 1B).
 - When the test has not passed correctly, you have to re record the sequence (STEP 1A and STEP 1B) again.

10.5. Names sequences (function 0605)

As mentioned before, you can also use this function to change the name of a sequence. Enter function 0605. Because there are 127 different sequences, this function consists of 127 subfunctions. In order to change the name of function 01, select function 060501. You will see the following data on the screen.

NAME SEQUENCE SEQUENCE 001 AKUaku1.â	UP DOWN OK BACK
---	--

Modify the name of your sequence in this item. We call it for example ‘week mass’.

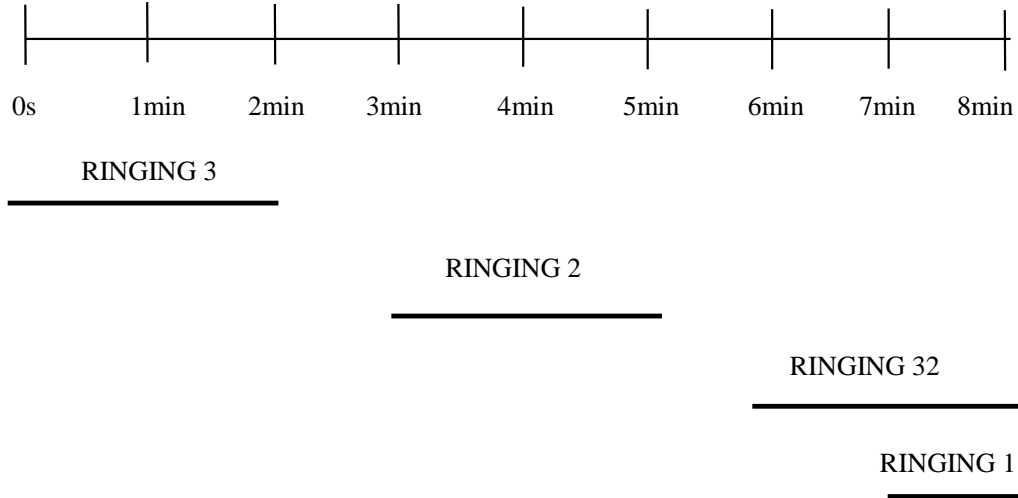


EXAMPLES:

- 1) WEEK MASS: RINGING
 - 2) STROKE BELL 1 (HOUR STROKE)
MANUAL TOLLING
PAUSE
 - 3) ANGELUS
MANUAL TOLLING
PAUSE
- AFTER
EXECUTIONS
“BACK”

10.5. The use of pauses during the recording of a sequence

Example 4 : Celebration mass - sequence type 1



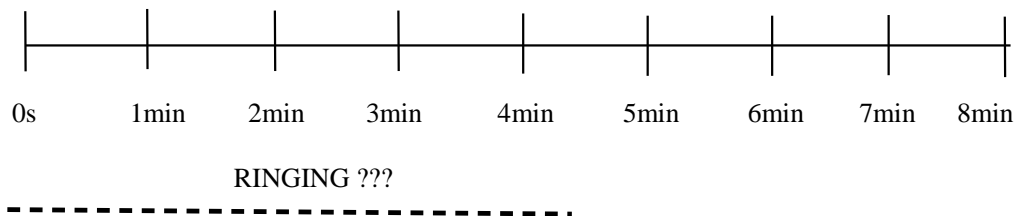
In the timetable you can see that the sequence consists of several ringing executions and pauses. To make sure that you can start two or more actions together (or before a previous execution is ended, like with the two last ringing executions), the introduction of pauses occurs as follows.

When you include a pause, you move further in time. You always have to include the pause between the time in which you find yourself and the time in which a next execution has to start. The following steps have to be completed:

- * ringing bell 3 for 2 minutes (ringing 3 02 minutes)
- * pause of 3 minutes (we are at minute 0, the next execution starts at min. 3)
- * ringing bell 2 for 2 minutes (ringing 2 02 minutes)
- * pause of 3 minutes (we are at minute 3, the next execution starts at min 6)
- * ringing bell 2 and 3 for 2 minutes (ringing 32 02 minutes)
- * pause of 1 minute (we are at minute 6, the next execution starts at min 7)
- * ringing bell 1 for 1 minute (ringing 1 01 minute)

10.6. “STOP” function during “record sequence”.

Example 5: Manual tolling sequence type 1



During the recording of a sequence, the “STOP” key gives the end user the opportunity to set a specific parameter himself. You can use this function to make a sequence, which allows the end user to choose the duration and the bells of a ringing execution.

Start the recording of a sequence: you do this, as mentioned before, by moving to function 0601 and pressing “OK”. That way, you enter the screen with the main functions.

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Because you want to start an execution (ringing) in this sequence, you have to press “OK” again. This is what you will see:

FUNCTION 06010101	UP
RINGING	DOWN
FUNCTION 06010102	OK
PLAY SEQUENCE	BACK

You can enter this function and see the following data appear on the screen:

RINGING	+1
!	-1
01 MINUTES	OK
WITH WHICH BELL	BACK

To give the client the opportunity to choose 3 bells, you press the “STOP” key 3 times. That way, the client will be asked to enter the parameter when the sequence is started.

RINGING	+1
.	-1
02 MINUTES	OK
WITH WHICH BELL	BACK

At that moment, you reach the fourth item. Since you only want to give the client the opportunity to choose 3 bells (there are for example only 3 in the bell tower), you affirm the other 7 choices with “OK”.

RINGING	+1
.	-1
01 MINUTES	OK
WITH WHICH BELL	BACK

You also want to let the client choose the ‘number’ of minutes. That is why you press “STOP” when ‘01’ is activated. Affirm the parameter ‘unit’ (minutes) with “OK”.

You have now entered everything you wanted to record in this sequence. Close the recording by means of the “BACK” key until you are back in the function 0601.

FUNCTION 0601 RECORD SEQUENCE	UP
FUNCTION 0602 TEST SEQUENCE	DOWN
	OK
	BACK

Now you can test your sequence (function 0602 – see 10.2) and save it (function 0603 – see 10.3).

Example 6: Change hands sequence type 3

The function of the “STOP” key during the recording of a sequence can also be used to make a sequence that change parameters (sequence type 3). The next example shows how to make a *sequence* with which you can correct the tower clock. The creation of this sequence is only necessary when you decide that the *end user* has to be able to do this on his own. Of course, it is better to let the installer do it directly in function 120201.

Start recording the sequence: as mentioned before, you do this by going to function 0601 and pressing “OK”. You enter the screen with the main functions.

FUNCTION 060101 EXECUTIONS	UP
FUNCTIONS 060102 USER MENU	DOWN
	OK
	BACK

No executions are started in this sequence (no type 1 sequence). The content of the function 120201 is asked. Therefore proceed to function 060112 ‘clocks’.

FUNCTION 060112 CLOCKS	UP
FUNCTION 060113 RELAYS	DOWN
	OK
	BACK

Press “OK” and choose ‘slave clock 1’ (function 060 11202).

FUNCTION 06011201 INTERNAL CLOCK	UP
FUNCTION 06011202 SLAVE CLOCK 1	DOWN
	OK
	BACK

Press “OK” and choose ‘time slave clock 1’ (function 06011202).

FUNCTION 06011202 TIME SLAVE CLOCK 1	UP
FUNCTION 06011202 TYPE SLAVE CLOCK 1	DOWN
	OK
	BACK

Press “OK” – e.g. following data may appear on the screen:

TIME SLAVE CLOCK 1 04 45 CHANGE HOURS	+1 -1 OK BACK
---	------------------------

To give the client the opportunity to enter the time of the slave clock, you have to press the “STOP” key twice. You proceed to the following screen:

FUNCTION 06011202 TIME SLAVE CLOCK 1 FUNCTION 06011202 TYPE SLAVE CLOCK 1	UP DOWN OK BACK
--	--------------------------

At this moment, you have recorded everything you wanted to include in this sequence. You can finish the recording by means of the “BACK” key until you are back in function 0601.

FUNCTION 0601 RECORD SEQUENCE FUNCTION 0602 TEST SEQUENCE	UP DOWN OK BACK
--	--------------------------

Now you can test your sequence (function 0602) and save it (function 0603). Of course, you can make a similar function for establishing the time and/or the date of the internal clock.

10.7. Properties sequences (function 0606)

*** Performance speed**

As mentioned before, you can use the sequence “stroke bell 1” (example) to program the hour stroke. If the melody has been recorded too slowly or too fast, you don’t necessarily have to record the melody again. Function 0606 allows you to change the properties of the sequences (performance speed and opening louver-hole) of each individual sequence. As there are 127 possible sequences, we first have to select the sequence of which we want to change the properties. Function 0606 consists of 127 subfunctions (functions 060601 - 060699). In order to change the performance speed of “stroke bell 1”, you have to enter the right subfunction and you will see following data appear on the screen:

SEQUENCE 001 100 % ON PERFORM. SPEED	UP DOWN OK BACK
--	--------------------------

Normally, the performance speed is 100%. That means that the pauses and the duration in time are rendered as fast as they have been recorded. If you want to play the sequence at a higher speed, you have to raise the value of the parameter. The parameter can take up any value between 20% and 500%. The performance speed also applies to all other actions in the sequence, such as ringing, pauses, continuous, ... However, when you start another sequence in this sequence (with ‘play sequence’), the performance speed will not be affected. Each sequence you start has its own performance speed.

*** Louver-hole**



When you have established a time that differs from 0 in function 1101 (louver-hole), the APOLLO II device will always open the louver-hole by the execution of a sequence or a program. However, this is not necessary and even disturbing for sequences, that only change parameters and/or programs. By means of the parameter “LOUVER-HOLE”, you can establish if you want to open (ON) the louver-hole or not (OFF) during the execution of the sequence. When the time is 0 in function 1101, the louver-hole can never be opened.

10.8. Delete sequence (function 0604)

This function allows you to delete one or more sequences. The first parameter indicates the first sequence to be deleted. When the second parameter is set at “NOT USED”, only this sequence will be deleted. When another sequence is entered in the second parameter, all sequences between the first and the last parameter will be deleted. When the second sequence has a smaller number than the first sequence, nothing will be deleted. Erasing a sequence takes about 10 seconds. Deleting of a continuous series of sequences in the memory also takes about 10 seconds.

DELETE SEQUENCE	UP
SEQUENCE 001	DOWN
NOT USED	OK
NAME SEQUENCE	BACK

11. Programs (function 04)

By means of programs, all executions of the APOLLO II device can be programmed in time. There are 2 types of programs:

Single programs (e.g. to program something **next** Monday or Tuesday): these programs are executed only once. Afterwards, they are deleted from the memory of the APOLLO II device. Normally, the single programs are only used during the creation of sequences that allow the client to enter single programs (sequence type 2).

Permanent programs (e.g. to program something **every** Monday or Tuesday): these are stored permanently in the device’s memory and can be executed several times if necessary. These programs are used to program the hour stroke, the angelus and the masses.

Number of program items:

	single	permanent
ringing	10	90
sequences	10	100
groups	10	30
continuous circuits	10	50
blocks	10	80

You can recognize an **empty program** by the parameter '**day of the week**'. Whenever this parameter is set to '**NEVER**', it's certain that the program is not used. To **erase** an existing program, you can always set the '**day of the week**' to '**NEVER**'.

11.1. Single programs (function 0401)

There are 5 types of single and permanent programs. These types are **ringing programs, sequence programs, group programs, continuous programs and block programs**.

For example: * a ringing program starts a ringing execution at a certain time.
 * a sequence program starts a sequence at a certain time.

11.1.1. Single ringing programs (function 040101)

A single ringing program allows you to start a certain ringing execution once. When this ringing program has to be started up immediately, the best way to do this is by means of executions - ringing (function 0101). If however, you want to start the ringing execution later, this can be programmed here. Move to function 040101. There are 10 single ringing programs available.

You will see the following data appear on the screen.

SG.RING.PROG. 01 01 MINUTES 00 00 NEVER	UP DOWN OK BACK
--	--------------------------

Select 1 of 10 possible ringing programs and press "OK". You will proceed to the following screen (the screen below is already completed as an example).

2 4 3 05 MINUTES 17 45 TUESDAY WITH WHICH BELL	UP DOWN OK BACK
---	--------------------------

Here you can establish 6 parameters :

1) With which bells

The first line of the screen (2 4 3), allows you to select the bells you want to ring. In this example, we have chosen ringing 2, ringing 4 and ringing 3. With that we can also determine the ringing order. Notice that bell number 2 will start ringing first, followed by bell number 4 and bell number 3.

2) Number

Indicates the number of units of time the ringing action will last, e.g. 05 minutes. The duration of the action always depends on the number and the type of unit (parameter 3). The number can be any value between 1 and 99.

3) Unit

The different units in a ringing program are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours.**

4) Start hours

By means of parameters 4 and 5 we can determine the starting up time of the ringing action. For example: we want to ring for 5 minutes at 17 hours 45 minutes. For that, we have to enter 17 in this parameter.

5) Start minutes

In our example we establish this parameter at 45. The program item will be executed at 17h45.

6) Day of the week

This parameter allows you to choose the day of the execution. Let us choose Tuesday.

We have programmed 5 minutes of ringing with bells number 2, 4 and 3. The ringing action will take place on the **next** Tuesday at 17h45.

Example :

Today is Monday, 14.45 h ⇒ execution tomorrow at 17.45 h.

Today is Tuesday, 14.45 h ⇒ execution today at 17.45 h.

Today is Tuesday, 18.45 h ⇒ execution next Tuesday at 17.45 h.

After the execution of this ringing action, the program is erased.

11.1.2. Single sequence programs (function 040102)

A single sequence program allows you to start a sequence program once. There are 10 single sequence programs available. You will proceed to the following screen (screen below is an example):

SG.SEQ.PROGR. 01	UP
WEEK MASS	DOWN
01 TIME	OK
16 35 WEDNESDAY	BACK

Choose 1 of 10 possible sequence programs and press “OK”. You will proceed to the following screen.

WEEK MASS	+1
01 TIME	-1
16 35 WEDNESDAY	OK
NAME SEQUENCE	BACK

Here you can establish 6 parameters:

1) Name sequence

Here you can choose which sequence you want to play. You can choose any of 127 possible sequences. However, it is not possible to start “seq. from group”.

2) Number

Indicates the number of units the execution will last, e.g. 01 time. The duration of the action always depends on the number and the type of unit (parameter 3). The number can take up any value between 1 and 99.

3) Unit

The different units in a sequence program are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours, hour stroke, next hour, times.**

4) Start hours

By means of parameters 4 and 5, we can determine the start time of the execution. Example: we want to play the sequence “week mass” at 16h35. Therefore, we have to enter 16 in this parameter.

5) Start minutes

In our example we have to enter 35 in this parameter. The program item will be executed at 16h35.

6) Day of the week

This parameter allows you to choose on what day of the week the APOLLO II device has to start the sequence. Let us choose Wednesday.

11.1.3. Single group programs (function 040103)

A single group program allows you to execute or activate (see chapter “groups”) a certain group (e.g. a concert with all the songs that are placed in the group) once. There are 10 single group programs available. You will proceed to the following screen.

SG.GROUP PROG. 1	UP
GROUP 01	DOWN
01 TIME	OK
00 00 NEVER	BACK

Choose 1 of 10 possible group programs and press “OK”. You will proceed to the following screen.

CHRISTMAS SONGS	UP
01 TIME	DOWN
16 35 MONDAY	OK
NAME GROUP	BACK

Here you can establish 6 parameters:

1) Name group

Under this heading, you can choose which group you want to play. You can choose one of 24 possible groups. Let us, for example, choose the group “Christmas songs”.

2) Number

Indicates the number of units the execution will last, e.g. 01 time. The duration of the action always depends on the number and the type of unit (parameter 3). The number can take up any value between 1 and 99.

3) Unit

The different units in a group program are: **seconds, 1/10 minutes, minutes, 1/10 hours, hours, times**. The unit “active” allows you to change the active group by means of a single group program (see chapter “groups”).

4) Start hours : See single sequence programs.

5) Start minutes : See single sequence programs.

6) Day of the week : See single sequence programs.

11.1.4. Single continuous programs (function 040104)

A single continuous program allows you to execute a certain continuous circuit once. There are 10 single continuous programs available. You will proceed to the following data on the screen:

SG.CONT.PROG. 01	UP
HEATING	DOWN
02 HOURS	OK
16 35 MONDAY	BACK

Select 1 of the 10 possible continuous programs and press “OK”. You will proceed to the following screen.

HEATING	+1
02 HOURS	-1
16 35 MONDAY	OK
NAME CONTINUOUS	BACK

Here you can establish 6 parameters:

1) Name continuous output

Under this heading you can choose which continuous circuit you want to switch on. You can choose any of the 10 possible continuous circuits.

2) Number

Indicates the number of units the execution will take up, e.g. 02 hours. The duration of the action always depends on the number and the type of unit (parameter 3). The number can take up any value between 0 and 99.

3) Unit

The different units in a continuous program are: **1/10 seconds, seconds, 1/10 minutes, minutes, 1/10 hours, hours, days, weeks, permanent, off.**

4) Start hours : See single sequence programs.

5) Start minutes : See single sequence programs.

6) Day of the week : See single sequence programs.

11.1.5. Single block programs (function 040105)

A single block program allows you to activate or deactivate a certain program block once. This can be used for example to block the hour stroke or the angelus during the Easter period. There are 10 single block programs available. You will proceed to the following screen.

SG.BLOCK PROG. 01	+1
PROGRAM BLOCK 01	-1
01 DAYS SUSP.	OK
00 15 NEVER	BACK

Select 1 of 10 possible block programs and press “OK”. You will proceed to the following screen.

AUTO HOUR STROKE	UP
01 DAYS SUSP.	DOWN
00 15 MONDAY	OK
NAME PROGR. BLOCK	BACK

Here you can establish 6 parameters:

1) Name program block

Under this heading you can choose which program block you want to switch on/off. You can choose any of the 10 possible block programs. Let us, for example, block the program block “auto hour stroke” for one day.

2) Number

Indicates the number of units the execution will take up, e.g. 01 “days susp.”. The duration of the action always depends on the number and the type of unit (parameter 3). The number can take up any value between 0 and 99.

3) Unit

The different units in a block program are: **1/10 seconds, seconds, 1/10 minutes, minutes, 1/10 hours, hours, days, weeks, permanent, off**. The parameters ‘hours susp.’ and ‘days susp.’ make sure that the program block is blocked for several hours or days (OFF). Afterwards, the program block will be placed in its original position.

4) Start hours : See single sequence programs.

5) Start minutes : See single sequence programs.

6) Day of the week : See single sequence programs.

11.2. Permanent programs (function 0402)

Like the single programs, the permanent programs can be divided into 5 categories. We will first discuss the permanent ringing programs, since the permanent continuous programs, group programs and block programs are practically identical to the permanent ringing programs. Because the permanent sequence programs are somewhat different from these four types, they will be discussed separately.

11.2.1. Permanent ringing programs (function 040201)

A permanent ringing program allows you to start a ringing execution **several times** (each program item only once a day, but on several days, e.g. **every** day of the weekend). There are 90 permanent ringing programs available. You will move to the following data on the screen:

RINGING PROGR.	1	
01	MINUTES	UP
	BLOCK-INDEPEND.	DOWN
	00 00 00	
	NEVER NORMAL	OK
	01 JANUAR	
	31 DECEMBER	BACK

Select one of the 90 possible ringing programs and press “OK”. You will move to the following screen (next screen is an example):

1 2 3	
05 MINUTES	UP
RINGING PROGR.	
17 45 23	DOWN
DAILY NORMAL	
01 JANUAR ..	OK
31 DECEMBER ..	
WITH WHICH BELL	BACK

You can establish the following parameters:

1) With which bell : see single ringing programs.

2) Number : see single ringing programs.

3) Unit : see single ringing programs.

4) Program block

On the third line you will find the parameter “program block”. This parameter allows you to make a certain execution dependent on the activation of a program block. In other words, the program block functions as a switch that you use to turn off certain programs (the programs which are dependent on this program block). Suppose you select program block “ringing program”. In that case, the ringing execution will only take place if the program block “ringing program” is activated (ON). The program block “ringing program” can be switched off for example during the Easter week. It is also possible to make the ringing program block - independent. That means that the programmed ringing execution will not depend on the activation of any program block. The disadvantage however is that the program cannot be switched off by the end user. Note that block-dependence can be considered as an additional condition for the execution of a program item. If the program item is block-dependent (not set at block-independent), it will not be executed unless the selected program block (here “ringing program”) is activated.

5) Start hours

The three following parameters determine the execution time. Here you can key in the starting up hour of the execution. Example: 17.

6) Start minutes

Here you can key in the starting up minutes of the execution. Example: 45.

7) Start seconds

Here you can key in the starting up seconds of the execution. Example: 00.

8) Day of the week

Because permanent program items can be executed several times, they have a few extra’s in comparison with single program items. It is possible to execute the program “daily”. It is also possible to do this on “week days” (Monday to Friday) or during the “weekend” (Saturday and Sunday). Further options are “MoTuThFr” (Monday, Tuesday, Thursday, Friday ⇒ for

schools) and “Mon→Sat” (all days except Sunday).

9) Easter program (SUMMER-WINTER PROGRAM)

This parameter can be used to make a program dependent on EASTER, Summer-or wintertime. There are four different options:

- 1) NORMAL: The parameter is ignored.
- 2) EASTER: Because Easter is not on the same date every year, the APOLLO II device has been provided with a program to calculate the date of Easter for every year. To remind the APOLLO II device that the execution of a certain program item has to take place on a day that depends of the Easter date, this parameter has to be set to “EASTER”. When you proceed to the correct date for one year, the APOLLO II device will calculate the correct date for each year.
- 3) SUMMER: When we choose the option SUMMER for this parameter, this program will only be executed during the SUMMER-period (from the last Sunday of March to the last Sunday of October). **If we do enter a year into the date, that will be regarded as an ‘Easter-dependent’ date.** Therefore, this date will be recalculated every year.
SUMMER = SUMMER + EASTER.
- 4) WINTER: When we proceed to the option WINTER for this parameter, this program will only be executed during the WINTER-period (from the last Sunday of October to the last Sunday of March). **If we do enter the year into the date, this will be regarded as an ‘Easter-dependent’ date.** Therefore, this date will be calculated every year. WINTER = WINTER + EASTER.

10) Date execution

The two bottom lines indicate the period in which the execution of the program item will take place. To explain this, we will give an example. There are 6 parameters in these lines.

01 (1)	JANUARY (2)	.. (3)
31 (4)	DECEMBER (5)	.. (6)

Parameters 1,2, 4 and 5 indicate the period of the year in which the program item has to be executed. Parameters 3 and 6 indicate the years in which the program item will be executed during this period. Remember that the APOLLO II device counts the years from 1994 to 2093. E.g. when you enter ‘23’, the execution will take place in 2023. ‘95’ means the year 1995. For example:

01	JANUARY	01
15	FEBRUARY	03

The program item will be executed for 3 years (2001,2002 and 2003) during the period from January 1st of 2001 to February 15th of 2003. In other words, in the following periods:

- 1) from 1 January 2001 to 15 February 2001
- 2) from 1 January 2002 to 15 February 2002
- 3) from 1 January 2003 to 15 February 2003

If both years are set at “..”, the program item will be executed every year from 1 January to

15 February.

Example:

- | | | | | |
|----|----|---------|------------------|---|
| 1) | 01 | MAY | .. | |
| | 15 | JULY | .. | ⇒ each year from 1 May to 31 July |
| 2) | 01 | JANUARY | 01 (2001) | ⇒ RIGHT |
| | 15 | JANUARY | 10 (2010) | ⇒ from 1 January to 15 January |
| 3) | 01 | JANUARY | 10 (2010) | (see NOTE 2) |
| | 15 | JANUARY | 05 (2005) | ⇒ WRONG (year numbers) – end year < start |
| 4) | 01 | JANUARY | .. | |
| | 15 | JANUARY | .. | ⇒ RIGHT |

Consider the next full example of a permanent ringing program:

1 3	
07 MINUTES	UP
SPRING PROGRAM	
17 45 00	DOWN
WEEKEND NORMAL	
01 MAY 01	OK
15 JUNE 04	
WITH WHICH BELL	BACK

This program item will have the following result:

If the program block “spring program” is active, the actions “ringing 1 and 3” will be started up for 7 minutes in the years 2001,2002,2003 and 2004, each day of the weekend in the period of 1 May to 15 June. When any of these conditions is not fulfilled, the execution will not take place. The conditions are:

- The selected program block (here spring program) is active (ON).
- The time matches the starting up time that was selected (17h 45m 00s).
- Today is the day (one of the days) we selected for the execution of the program item (in this example it has to be a weekend ⇒ so Saturday or Sunday).
- Today’s date lies within the selected periods (from May 1st to June 15th).
- We are in the year 2001, 2002, 2003 or 2004.

11.2.2. Easter programs

Programs that can be linked to the Easter period are called Easter programs. To make sure the APOLLO II device knows that the entered program is an Easter program, we have to set the parameter “Easter program” to “EASTER”. Even in the case of SUMMER-or WINTERTIME you can make a program dependent on Easter, depending if you enter the years or not. This was already discussed in 11.2.1. The dates we entered to in lines 5 and 6 will then be calculated automatically for each following year. Let us look at an example.

It is always necessary to enter the correct Easter date for one particular year. We know e.g. that Easter in 2002 was on Sunday 31 March. However, we want to start a ringing program

from the day before Easter until two days after Easter. Therefore, we enter the following parameters:

1 3	
07 MINUTES	UP
EASTER PROGRAM	
17 45 00	DOWN
DAILY EASTER	
30 MARCH 02	OK
02 APRIL 02	
WITH WHICH BELL	BACK

As the APOLLO II device knows that Easter 2002 is on Sunday 31 March, it can now calculate that we want to start the ringing execution from the day before Easter until two days after Easter. When we quit this function, the device will adapt these dates automatically to the actual year. Let us suppose that we are in the year 2003. The APOLLO II device knows that Easter 2003 is on Sunday 20 April. It also knows that we want to start the ringing executions from one day before Easter until two days after Easter. Therefore, the APOLLO II device will adapt the dates as follows:

1 3	
07 MINUTES	UP
EASTER PROGRAM	
17 45 00	DOWN
DAILY EASTER	
19 APRIL 03	OK
22 APRIL 03	
WITH WHICH BELL	BACK

The adaptation takes place when the device is in standby mode. This is the case when we are in the main menu.

EXAMPLE: When we want, for example, a program that is only played in SUMMER, but after EASTER, you can enter the following data:

1 3	
07 MINUTES	UP
EASTER PROGRAM	
17 45 00	DOWN
DAILY SUMMER	
31 MARCH 02	OK
31 DECEMBER ..	
WITH WHICH BELL	BACK

This program is active from EASTER (07/04/02) to the 31st of December (fixed, because we haven't entered a year), provided that it is SUMMER. Therefore, the first date 31/03/02 will be recalculated every year, because we have entered a year (02).

11.2.3 Similar program items

As mentioned before, there are 5 types of **permanent** program items. 3 out of 4 remaining types (besides the 4 ringing programs) are programmed in exactly the same way as the ringing programs. These 3 programs are **continuous programs, block programs and group programs**. For these types, we therefore refer to the other chapters.

11.2.3.1. Permanent group programs (function 040203)

A permanent group program allows you to start a group **several times** (each program item only once a day but several days). There are 30 permanent group programs available. You will proceed to the following screen

GROUP PROGRAM	1	
CHRISTMAS SONGS		UP
01 TIME		
BLOCK-INDEPEND.		DOWN
00 00 00		
NEVER NORMAL		OK
01 JANUARY ..		
31 DECEMBER ..		BACK

Select 1 out of 30 possible group programs and press “OK”. You will proceed to the following screen.

CHRISTMAS SONGS		UP
01 TIME		
AUTO CARILLON		DOWN
12 05 00		
DAILY NORMAL		
21 DECEMBER ..		OK
31 DECEMBER ..		
NAME GROUP		BACK

For the first 2 lines on the screen we refer to the single group programs (chapter 11.1.3). The 5 following lines are identical to the permanent ringing programs.

Here the group will be played once. That means that all sequences that are in the group, will be played one by one. Here too, the unit “active” allows you to activate another group. (see chapter “groups”)

11.2.3.2. Permanent continuous programs (function 040204)

A permanent continuous program allows you to switch on a continuous circuit **several times** (each program item only once a day but several days). There are 50 permanent continuous programs available. You will proceed to the following screen:

CONT. PROGR.	1	
HEATING		UP
02 HOURS		
AUTO HEATING		DOWN
18 45 00		
SATURDAY NORMAL		OK
01 JANUAR ..		
31 DECEMBER ..		BACK

Select 1 of the 50 possible continuous programs and press “OK”. You will proceed to the following screen.

HEATING	
02 HOURS	UP
AUTO HEATING	
18 45 00	DOWN
SATURDAY NORMAL	
01 JANUAR ..	OK
31 DECEMBER ..	
NAME CONTINUOUS	BACK

For the first 2 lines of the screen we refer to the single continuous programs (chapter 11.1.4). The 5 following lines are identical to the permanent ringing programs. For example: illumination, heating church, heating winter chapel, ...

11.2.3.3. Permanent block programs (function 040205)

A permanent block program allows you to start a program block **several times** (each program item only once a day but several days). There are 80 permanent block programs available. You will proceed to the following data on the screen:

BLOCK PROGR. 01	
AUTO CARILLON	UP
01 DAYS SUSP.	
BLOCK-INDEPEND.	DOWN
00 10 00	
DAILY EASTER	OK
28 MARCH 02	
30 MARCH 02	BACK

Choose 1 of 80 possible block programs and press “OK”. You will move to the following screen.

AUTO CARILLON	
01 DAYS SUSP.	UP
BLOCK-INDEPEND.	
00 10 00	DOWN
DAILY NORMAL	
28 MARCH 02	OK
28 MARCH 02	
NAME PROGR.BLOCK	BACK

For the first 2 lines of the screen we refer to the single block programs (chapter 11.1.5). The 5 following lines are identical to the permanent ringing programs.

11.2.3.4. Permanent sequence programs (function 040202)

The permanent sequence programs are somewhat different from the other permanent programs. Sequences can be **executed several times a day by means of one permanent sequence program**. That is not possible for other programs. For these items you have to program several program items in order to obtain several executions **a day**.

A permanent sequence program allows you to start a sequence **several times** (even several times a day). There are 100 permanent sequence programs available. You will proceed to the following screen.

SEQ.PROGRAM 001	
MELODY 01	UP
01 TIME	
BLOCK-INDEPEND	DOWN
BEGIN 00 00 00 00	
NEVER NORMAL	OK
01 JANUAR ..	
31 DECEMBER ..	BACK

Select 1 of 100 possible sequence programs and press “OK”. You will proceed to the following screen.

Example:

STROKE BELL 1		(1)
01 HOUR STROKE	UP	(2)
AUTOM. HOUR STROKE		(3)
BEGIN 08 22 00 00	DOWN	(4)
DAILY NORMAL		(5)
01 JANUARY ..	OK	(6)
31 DECEMBER ..		(7)
NAME SEQUENCE	BACK	

For the first 2 lines in the screen we refer to the single sequence programs (chapter 11.1.2). The **4th line** (in bold) on the screen is different from all other permanent programs. Here you have to set 5 parameters.

1) Begin start hour (08)

The parameter “begin start hour” indicates at what time the sequence will be started up every hour. In our example we have entered 08. That means that our melody will be played every hour from 8 am onwards. When we choose “..” here, this parameter will be ignored. In that case, only the parameter “end start hour” will be taken into account.

2) End start hour (22)

The parameter “end start hour” indicates until what time the sequence will be started up every hour. In our example we have entered 22. That means that our melody will be played for the last time at 22 pm. Actually, the parameters “begin start hour” and “end start hour” allow you to program night switch-off.

3) Start minutes

This parameter indicates at what precise minute the sequence has to be started up. In our example we enter 00.

4) Start seconds

This parameter indicates at what precise second the sequence has to be started up. In our example we enter 00.

5) Execution sequence

The possibilities are “BEGIN” and “END”. These allow you to indicate whether the selected sequence should **start** or **end** at the minute and the second we enter the previous parameters. We will give another example later on in this manual.

EXAMPLE: BEGIN 08 22 00 00

The sequence “stroke bell 01” will be **started up** at 00 minutes and 00 seconds. That is determined by the last two parameters together with the parameter “**BEGIN**”. The execution will normally take place every hour. However, the parameters “begin start hour” (08) and “end start hour” (22) allow you to determine from when until when this execution should take place.

The first time this happens is at **08 00 00** ⇒ 08h 00 min 00sec

The last time this happens is at **22 00 00** ⇒ 22h 00 min 00 sec

In other words, the sequence ‘hour stroke 1’ will be played at:

**08h 00m 00s ; 09h 00m 00s ; 10h 00m 00s ; 11h 00 00s ; 12h 00m 00s ;
13h 00m 00s ; 14h 00m 00s ; 15h 00m 00s ; 16h 00m 00s ; 17h 00m 00s ;
18h 00m 00s ; 19h 00m 00s ; 20h 00m 00s ; 21h 00m 00s ; 22h 00m 00s**

Let us look at another example. This example can be used to enter a 4/4 hour stroke just before the hour:

STROKE BELL	2	(1)
04 TIMES		UP (2)
AUTOM. HOUR STROKE		(3)
END 07 21 59 59		DOWN (4)
DAILY NORMAL		(5)
01 JANUARY ..		OK (6)
31 DECEMBER ..		(7)
NAME SEQUENCE		BACK

When we enter these values, the sequence “stroke bell 2” (that is played 4 times) will end at the time we entered without having to know how long this sequence will last. The APOLLO II device calculates itself when to start in order to finish at 59 min 59 sec. The sequence “stroke bell 2” will be started up at the following times:

entered time - sequence duration (let’s assume this is 4x2 SECONDS)

07h 59m 59s (entered time) - 0h 0m 8s (sequence duration x 4) = **07h 59m 51s;**
08 h 59m 51s; 09u 59m 51s ; 10h 59m 51s ; 11h 59m 51s ; 12h 59m 51s ;
13h 59m 51s ; 14u 59m 51s ; 15h 59m 51s ; 16h 59m 51s ; 17h 59m 51s ;
18h 59m 51s ; 19u 59m 51s ; 20h 59m 51s ; 21h 59m 51s

Other possibilities

* When you only want 1 execution (e.g. at 20 p.m. exactly), you have to enter the following:

BEGIN 20 20 00 00 or BEGIN . 20 00 00

* When you want to start a sequence **without night switch-off**, enter **00 23** for 'begin start hour' and 'end start hour'.

START **00 23** 00 00

In that case the sequence will be started up every hour (in this case at 00min 00sec). There is no night switch-off.

11.3. Priorities of the executions

To prevent that certain executions would take place at the same time, a priority table has been provided in the APOLLO II device. This table should prevent the simultaneous execution of e.g. the hour strokes and a melody. Executions that are started up **manually, all** take place, because the user can decide whether an execution must be started up or not.

For executions that are started up by means of program items, the following priorities are valid:

1) **Continuous circuits, program blocks and non-bell sequences are always started up.**

Non-bell sequences are sequences that contain no bell functions, such as tolling, swinging or manual tolling. A bell sequence is a sequence that comprises certain actions with the bells (tolling, ringing, playing melodies on bells). Please note that a sequence that contains no bell functions but starts another bell sequence will be considered as a non-bell sequence itself.

2) For the other types of actions the following priority table was established.

- 1) single ringing programs
- 2) single bell sequence programs
- 3) single group programs
- 4) permanent ringing programs
- 5) permanent bell sequence programs non-hour stroke (see note)
- 6) permanent group programs
- 7) permanent bell sequence programs hour stroke (see note)

Note: permanent bell sequence programs **hour stroke** are bell sequence programs, programmed to execute **several** times a day. In other words, these are the program items for which the parameters "begin start hour" and "end start hour" are different. Permanent bell sequence programs that are executed only **once** a day ("begin start hour" = "end start hour" of "begin start hour" = . .) are called permanent bell sequence programs **non-hour stroke**.

The priority table should be interpreted as follows. The APOLLO II device will constantly check the priority table from point 1 to point 7. When it comes across a program item that should start a certain action, the APOLLO II device will check if this is possible. To avoid any misunderstandings, we will call the **types** of program items in the priority table "**table programs**". These table programs are all program items, except for the continuous programs, the block programs and the non-bell sequence programs. There are two possible situations:

1) None of the table programs is active:

- a) One program item becomes valid: the action programmed in this program item will be started up.
- b) Several program items become valid at the same moment: here the priority table is important. The program item with the highest priority will be executed. The other program items will be ignored. Example: at 17h 00m 00s a single bell sequence program (priority 2) and a permanent group program (priority 6) should be started up (according to our program). The APOLLO II device checks the priority table. The device doesn't find valid ringing programs and goes to priority level 2. Those are the single bell sequences. The device will see that there is a valid single bell sequence program and it will start the corresponding bell sequence. Then, the APOLLO II device will quit the priority table. The permanent group program will be ignored.
- c) Several program items with the same priority are valid: e.g. at 17h 00m 00s, 3 permanent non-hour stroke sequence programs become valid, namely sequence programs 2, 7 and 13. The permanent sequence program with the lowest program number is started up, while the others are ignored. Result: the permanent sequence program 2 is started up while 7 and 13 are ignored.

2) One of the active table programs is already being executed:

When a table program becomes active during the execution of another table program, it will be ignored.

Exception: When the active program item has started up an action (e.g. a bell sequence) in which there is a pause (no active bell functions), the APOLLO II device will check whether it can fit the action of the second program item into this pause. If that is the case, the second program item will be executed. A pause is any length of time during which there are no executions of **bell functions** (no ringing, no tolling, no melodies). For example:

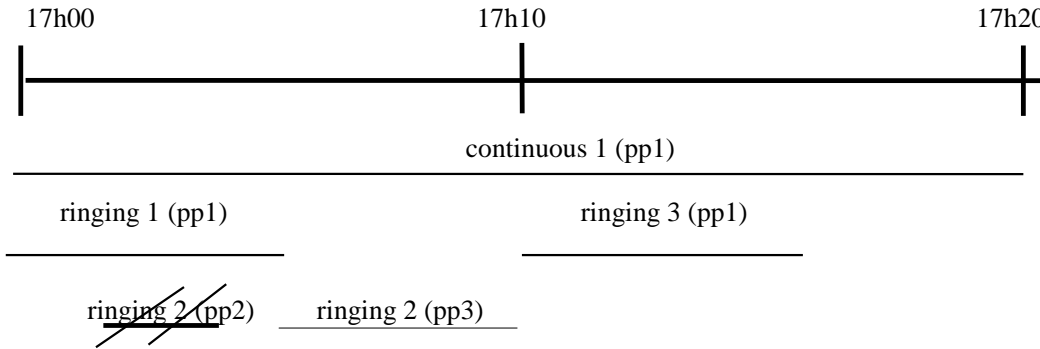
Program item 1: single bell sequence program: switch on continuous for 20 minutes at 17hrs 00m + ringing bell 1 for 5 minutes + pause of 10 minutes + ringing bell 3 for 5 minutes.

Program item 2: permanent ringing program: ringing bell 2 for 2 minutes
(start at 17h 02m)

Program item 3: permanent ringing program: ringing bell 1 for 4 minutes
(start at 17h 06)

Result:

- 1) Program item 1 will be started up.
- 2) Program item 2 will be ignored, because there is already another table program active (program item 1) and program item 2 does not fit into the pause of program item 1. There is another bell function active (ringing 1 of program item 1).
- 3) Program item 3 will be executed, because this program item fits into the pause of program item 1. Remember that continuous 1 of program item 1 is not a bell function → the time between 17h 05m and 17h 10m is therefore considered as a pause.



It is important not to program too many items at the same time. Try to vary the starting times of your program items to make sure that no items are ignored on the basis of the priority table. For example, when you enter a permanent program item that should start a sequence once a day, and you program this item at the time of the hour stroke, the hour stroke will not be executed. The “hour stroke sequence” has a lower priority (7) than the “non-hour stroke sequence” (5) and will be ignored. This can be avoided by starting up the non-hour stroke sequence after the hour stroke execution. Should you notice that a certain program item is not executed, it is therefore necessary to check the priority table.

12. Groups (function 07)

As mentioned before, a group can be seen as a series of sequences. One of the most common applications of groups is the playing of cyclic melodies. For this application, the APOLLO II device allows you to start the next sequence of a group in each program item. For example, when this program item is activated every quarter of an hour, the APOLLO II will always start the next sequence of the group. After playing the last sequence of the group, the device will repeat the series, starting up from the first melody. However, it is also possible to play the entire group by means of a program item (e.g. for carillon concerts).

Below, you find an example of a cyclic sequence program:

SEQ.PROGR.	1	
SEQ. FROM GROUP		UP
01 TIME		
AUTO MELODIES		DOWN
BEGIN 00 23 01 00		
DAILY NORMAL		OK
01 JANUARY ..		
31 DECEMBER ..		BACK

In this function you can determine the contents of the group (in other words the sequences which belong to this group), the way the group is played and the name of the group. Therefore, this function comprises of 4 subfunctions.

12.1. Names groups (function 0704)

In this function you can enter the names of the different groups (see program blocks &

continuous outputs). As there are 24 possible groups, function 0704 consists of 24 different subfunctions (070401-070424)

FUNCTION 070401	UP
NAME GROUP 1	DOWN
FUNCTION 070402	OK
NAME GROUP 2	BACK

Select the group you wish to give a different name, e.g. group 3 (function 0807)

NAME GROUP 03	UP
CHRISTMAS SONGS	DOWN
AKUaku1.â	OK
	BACK

Repeat this procedure for the other groups you want to rename.

12.2. Content groups (function 0703)

In this function you can determine the content of the groups. You can determine what sequences will be played and in what order. As there are 24 possible groups, this function has 24 subfunctions (select group). Each one of these 24 subfunctions consists of 96 end functions (each group may contain up to 96 sequences). When you enter an end function, you will see the following data on the screen. Let us, for example, program the second sequence of the 3rd group. We proceed to function 0703 (content groups) and we open function 070303 (contents group 3). Finally, we move to function 07030302 (content 2nd position in group 3). Here we can enter for example 'silent night'.

POS. 02 GROUP 03	UP
SILENT NIGHT	DOWN
NAME SEQUENCE	OK
	BACK

12.3. Time between sequences (function 0702)

When you want to play the whole group, it is important to enter the pause between each melody. This pause is indicated in seconds (0-99 seconds).

TIME BETWEEN SEQ	+1
05	-1
PAUSE IN SECONDS	OK
	BACK

12.4. Active group (function 0701)

This function determines which group will be used for playing "seq. from group". Let us, for example, select the group of Christmas songs. The activation of a group can be done in function 0103 (executions – play group), as mentioned before. The active position (here 02) determines which song will be played as the next "seq. from group". Each time a "seq. from group" is played, the counter will automatically raise the number by 1. After the last sequence has been played, the counter will return to 1. In order to 'reset' the group, the

counter should be set at 1. The “Resetting” of a group takes place automatically. Every night at 23h50 the counter (active position) will return to 1.

ACTIVE GROUP CHRISTMAS SONGS 02	UP DOWN OK BACK
SELECTION GROUP	

13. User menu (function 02)

This function allows the installer to select the different items he wants to put in the user menu. He can also program the numerical keys in such a way, that they can be used for manual pealing or tolling. That way, setting up the user menu becomes very straightforward. The user menu allows the user to start certain executions manually. The selection of executions in the user menu can be determined in this function. The user menu consists of 10 pages of 3 executions each. For example:

RINGING 123..... HEATING CHRISTMASPROGR NEXT SCREEN	PRESS® OFF ON PRESS®
--	---

The different executions can be activated manually by means of the user keys. In this example, the upper key will start “ringing 123.....”. The duration of ringing will be determined in function 02 (user menu). For this example, the 2 user keys in the middle are used to switch on the heating (a continuous circuit) and the Christmas program (a program block). By pressing the lower user key, you will proceed to the following screen. However, when a **fully empty screen** is entered (in function 02) you will return to the main menu.

As the user menu can render 30 different executions, function 02 consists of 30 subfunctions (function 0201-0230), each consisting of 2 subfunctions. The first subfunction is used to indicate for each item what **type** of execution you want to place in the user menu. Depending on the selected type, subfunction 2 will allow you to further define the **action**. Consider the following example.

Proceed to function 02. Here you can distinguish 30 different subfunctions.

FUNCTION 0201 USER MENU 1 FUNCTION 0202 USER MENU 2	UP DOWN OK BACK
--	--

Proceed to function 0201. Here we can define the first item of the user menu. Enter the following screen.

FUNCTION 020101	UP
USER TYPE 1	DOWN
FUNCTION 020102	OK
USER ACTION 2	BACK

13.1. User type (function 02xx01) : xx = 1 to 30

Because the user action depends on the user type, you first have to proceed to function 020101 (user type: in the user menu 01).

USER TYPE 1	UP
NOT USED	DOWN
SELECT USER MENU	OK
	BACK

Here you can change one parameter: the type of execution of this user item (user menu 01). The different options are:

- * **not used** : you don't want to use this user item. Attention: The first page is always shown. Further, you can fill out every screen at will. It is not necessary to fill out every line of the screen. Only when an empty screen is entered, you can return to the main menu, even if some lines are used.
- * **ringing** : to show ringing executions.
- * **sequence** : to show the execution of a sequence.
- * **group** : to show the execution of a group.
- * **continuous** : to show the execution of a continuous circuit.
- * **program block** : to show the execution of a program block.

13.2. User action (function 02xx02) : xx = 1 to 30

As mentioned before, function 02xx02 depends on the programmed user type. Therefore, we will discuss the different user actions in the following paragraphs.

A) user type = not used

The settings in this screen are not relevant.

B) user type = ringing

Proceed to the following screen for the user action:

USER ACTION 01	+1
2 4 3	-1
04 MINUTES	OK
WITH WHICH BELL	BACK

At this point we can determine again which bells we want to start and for how long. To do this, we refer to chapter “executions ⇒ ringing” (chapter 9.1.1). However, there is a big difference between starting up a ringing execution (function 0101) and the user action which we see here. When we choose “..” for the parameter “**number**” (this is the first parameter in the third line ⇒ here “04”), the user will be asked to enter this value (after having started up the execution in the user menu).

C) user type = sequence

Proceed to the following screen for the user action:

USER ACTION 01	+1
SEQUENCE 01	-1
01 HOUR STROKE	OK
NAME SEQUENCE	BACK

With this you can select the sequence you want to start along with the duration of the execution. To program this, we refer to chapter “executions P play sequence” (chapter 9.1.2). Here too, you can set the parameter “number” to “..” (see B).

D) user type = group

Proceed to the following screen for the user action:

USER ACTION 1	+1
GROUP 01	-1
01 TIME	OK
NAME GROUP	BACK

At this point you can select the group you want to start along with the duration of the execution. To program this, we refer to chapter “executions P play group” (chapter 9.1.3). Here too, you can set the parameter “number” to “..” (see B).

E) user type = continuous

You can proceed to the following screen for the user action:

USER ACTION 01	+1
CONTINUOUS 01	-1
01 HOUR	OK
NAME CONTINUOUS	BACK

You can select the continuous circuit you want to start along with the duration of the execution. To program that, we refer to the chapter “executions P switch on continuous” (chapter 9.1.4). Here too, you can set the parameter “number” to “..” (see B). For continuous programs and block programs there is always an “ON/OFF” signal in the user menu. For the other types there will be a “PRESS” signal.

F) user type = program block

Proceed to the following screen for the user action:

USER ACTION 01	+1
PROGRAMBLOCK1	-1
01 HOUR	OK
NAME PROGRAMBLOCK	BACK

At this point you can select the program block you want to start and the duration of the execution. To program that, we refer to the chapter “executions P program block” (chapter 9.1.5). Here too, you can set the parameter “number” at “..” (see B). For continuous programs and block programs there is always an “ON/OFF” signal in the user menu. For the other types there will be a “PRESS” signal.

13.3. Function key 1-0 (function 0231)

FUNCTION KEY 1-0	UP
RINGING	DOWN
05 MINUTES	OK
FUNCTION KEY1-0	BACK

Here, you can distinguish 3 different parameters:

“Function key 1-0”: this allows you to determine whether you want to use the numerical keys for **manual tolling, ringing, continuous outputs** or leave them “**not used**”. Please note that this function is **only active in the user menu**.

“Number (ringing)”: the number and the unit indicate how long the peal will last after pressing the numerical key once.

“unit (ringing)” : the number and the unit indicate how long the peal will last after pressing the numerical key once.

For example: Suppose the numerical keys are set for ringing and this for about 05 MINUTES. The numerical keys correspond to the respective ringing circuits. E.g. key “1” corresponds to “ringing 1”, key “2” corresponds to “ringing 2” etc. The parameter 05 MINUTES means that ringing will start for 5 minutes, when you press the numerical key once. When you press it twice, the ringing execution will take up 10 minutes. The indication “..” MINUTES means that the button is used as a ON / OFF switch.

14. Inputs (function 03)

Function 03 is identical to the user menu function (function 02). The only difference is that executions started in this function aren’t started in the user menu, but by means of an input. This input can, for example, be used to connect an external switch. Just like function 02, function 03 always contains an input **type** and an input **action**. As the external switch is located far from the APOLLO II device in most cases, it is not possible to set the parameter “number” to “..”. Make sure that you use push buttons for these inputs. Switches that remain “off” or “on” cannot be used.

D. PROGRAM EXAMPLE

1. Preliminary remarks

In this part, we will program an entire installation (3 swinging bell motors, 3 hammers + tower clock) from A to Z. When some techniques have been described earlier, we will refer to the part 'SOFTWARE'. If you are faced with any difficulties, we advise you to re-read the functions in question in the chapter 'SOFTWARE'.

2. Description of the installation

The installation is composed of 4 bells, of which the 3 smallest bells ring. All bells are provided with an electromagnetic hammer. There are also 4 dials (slave units) that need to be commanded by means of the APOLLO II device. Evidently, we can connect these 4 dials in parallel.

That means that we need a total of 3 (swinging bell motors) + 4 (hammers) + 2 (tower clock) = 9 outputs. In that case, we need an APOLLO II device with 10 outputs.

3. Program form

The next page shows you a program form with all the executions (programmed and/or manual).

To turn off any fixed programs (e.g. hour stroke), you have to provide a program block. When this program block is installed in the user menu, the customer can turn on/off the program.

In the last column of the program form, you can see which executions need a sequence. That is the case when:

* You have to program or start a **plural execution** manually – a single execution (e.g. just ring with one or more bells, start a group) can be programmed directly in the programs that are foreseen for that purpose (ringing-, continuous-, group-, block programs). When we talk about plural executions, e.g. 3x3 strokes + pause + ringing (angelus), you have to record a sequence before you can program the angelus.

* The customer wants to start something manually in the **user menu**, e.g. ringing for the funeral – despite of the fact that this is a single execution, we advise the use of a sequence in that case – that way, the customer does not have to choose the ringing bell and the duration himself, since all the information was already recorded in the sequence. This makes it a lot easier for the final user.

Execution	Manual	Entrance	Prog	Pr.Bloc	Bell	Duration	Moment	Day	Seq,
Angelus			x	x	3x3 (KL3) + L(KL2)	1 time	06.01	MO-SAT	x
			x	x	3x3 (KL3) + L(KL2)	1 time	12.01	MO-SUN	x
			x	x	3x3 (KL3) + L(KL2)	1 time	18.01	MO-SUN	x
hour stroke			x	x	KL1	hour stroke	8-22u	MO-SUN	x
half-hour stroke			x	x	KL1	1 stroke	8-22u	MO-SUN	x
1/4 hour stroke			x	x	KL2	1 stroke	8-22u	MO-SUN	x
2/4 hour stroke			x	x	KL2	2 strokes	8-22u	MO-SUN	x
3/4 hour stroke			x	x	KL2	3 strokes	8-22u	MO-SUN	x
4/4 hour stroke			x	x	KL2	4 strokes	8-22u	MO-SUN	x
Week mass			x	x	KL3+4	2 MIN	8.55u	MO-FR	
Sat. Mass			x	x	KL2+3+4	3 MIN	17u55	SAT	
Sun, Mass			x	x	KL 2+3+4	3 MIN	9u55	SUN	
Mourning (Tolling)	x	x			KL. 3121	5 MIN			x
Marriage (Ringing)	x	x			KL4+3+2	5 MIN			x
Funeral (Tolling)	x	x			KL3+2	5 MIN			x
Tower clock	x				to correct the hands of the tower clock				x

Manual: the customer wants to start this execution manually by means of the user menu.

Input: the customer wants to start this execution manually by means of an input (external switch, remote control or DTMF).

Prog (programmed): this execution has to be carried out automatically.

Pr. Block (Program block): the customer wants to switch on/off this program in the user menu – with this, he can switch off a fixed program temporarily, e.g. in the Easter period.

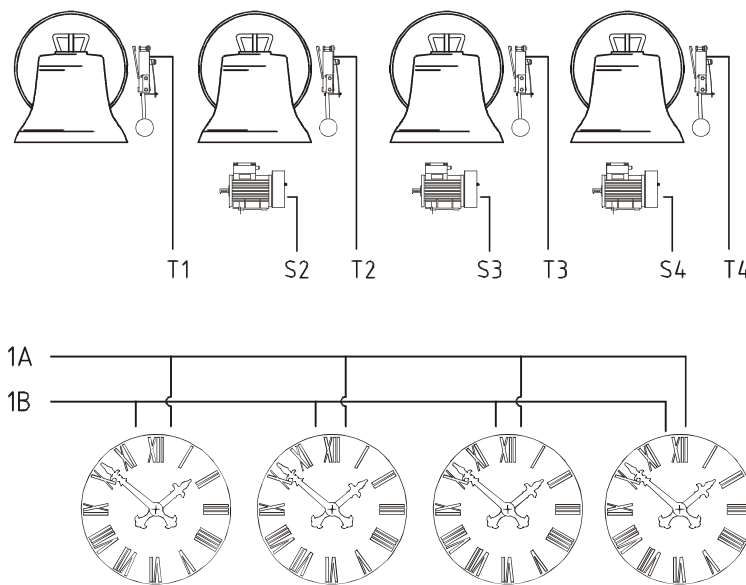
Seq. : indicates if this application needs the recording of a sequence.

On page 16 you can find a survey of the different steps during the programming of the APOLLO II device. During the realization of upper program form, we will follow this procedure.

STEP 1 : defining relays (outputs) – function 13

In function 13 (relays) we can define which outputs we want to use for the swinging bell motors, the hammers and the tower clock. Since we have 3 swinging bell motors, 4 hammers and 1 tower clock, we have to define 9 outputs here.

Make sure that, when you define a swinging bell circuit (ringing 1, ringing 2, ringing 3...) or a hammer circuit (hammer 1:C1, hammer 2:CIS1, hammer 3:D1, ...), you always use the number of the **bell**, as indicated in the following design.



Example:

When there are 4 bells, and only 3 of them have to ring, it is better to use the swinging bell circuits that correspond to the serial numbers of the bells.

In our case, we use RINGING 2, RINGING 3 and RINGING 4. That is important because it is taken as a standard that during RINGING 2, hammer 2:CIS1 is blocked (this hammer 2:CIS1 is also installed on bell 2).

When we would head the swinging bell motor for the second bell with RINGING 1, hammer 1:C1 and not hammer 2:CIS1 (that is installed on the second bell) would be blocked, during the execution of RINGING 1.

Establishment function 13

For our program example, we set up the following:

Function 1302 : Relay 01 : RINGING 2
Function 1303 : Relay 02 : RINGING 3
Function 1304 : Relay 03 : RINGING 4
Function 1305 : Relay 04 : NOT USED
Function 1306 : Relay 05 : HAMMER 1:C1
Function 1307 : Relay 06 : HAMMER 2:CIS1
Function 1308 : Relay 07 : HAMMER 3:D1
Function 1309 : Relay 08 : HAMMER 4:DIS1
Function 1310 : Relay 09 : CLOCK 1A
Function 1311 : Relay 10 : CLOCK 1B

STEP 02 : establishing names of continuous-outputs and program blocks

*** Function 09: PROGRAM BLOCKS**

A program block can be used as a switch to block certain programs. That way, the final user can temporarily block an automatic execution.

The program form shows that the customer wants to switch the hour stroke, the angelus and the masses on and off when necessary (e.g. Easter, disease...).

Since it is easier for the customer to talk about e.g. 'autom. hour stroke' than about 'program block 01', we can change the names of 4 program blocks in function 09.

These names will appear again later in other functions and screens.

Function 0901: change name in 'AUTOM. HOUR STROKE

Function 0902: change name in 'AUTOM. QUARTERS *

Function 0903: change name in 'AUTOM. ANGELUS'

Function 0904: change name in 'AUTOM. MASSES'

Observations:

- 1) The program block 'AUTOM. QUARTERS' is only necessary when the priest wants to switch on/off the quarter strokes (1/4 ; 2/4 ; 3/4 and 4/4 hour stroke) separately, without changing the hour stroke. The hour stroke itself can be commanded by means of the program block 'AUTOM. QUARTERS'. Because of that you can switch off the quarter strokes (AUTOM. QUARTERS = OFF) and switch on the hour stroke (AUTOM. QUARTERS = ON).
- 2) If necessary, you can use other program blocks to switch on/off the masses separately, e.g. 'AUTOM. SUN. MASS'. You can do this in functions 0905, 0906 up to 0924.

*** function 08 : CONTINUOUS-OUTPUTS**

Continuous outputs are used for heating, illumination, carousel, little doors, ... In our example, we don't need to command continuous outputs. That is why we don't need to change names in function 08.

STEP 03: create sequences (if necessary) + program

Programming continuous outputs (illumination, heating), ringing bells (masses, ...) and groups can be performed immediately in programs that are constructed for that purpose (**function 04**: continuous programs, ringing programs and group programs resp.).

All the executions for which you need a *sequence* (*hour stroke, angelus, funeral, tolling, bim bam, ...*) can only be programmed after having recorded the sequence (**function 06**).

Applied to our example, that means that the ringing programs (week mass, Saturday mass and Sunday mass) can be programmed directly. Concerning the other executions (angelus, hour stroke, mourning, marriage, funeral, tower clock), you have to record a sequence that you can program afterwards and/or install in the user menu. That is indicated by means of little crosses in the last column of the program form.

In this manual, we will first record the required sequences (function 0601) and then program them directly (function 04) for every application (hour stroke, angelus, masses...).

Of course you can also record **all** sequences first and then program them all at the same time. That way you do not have to switch all the time from function 06 to function 04. Still, that means though that you first have to consider what you want to put in the device.

Notice that the largest part of the following information was already discussed in "part C: SOFTWARE".

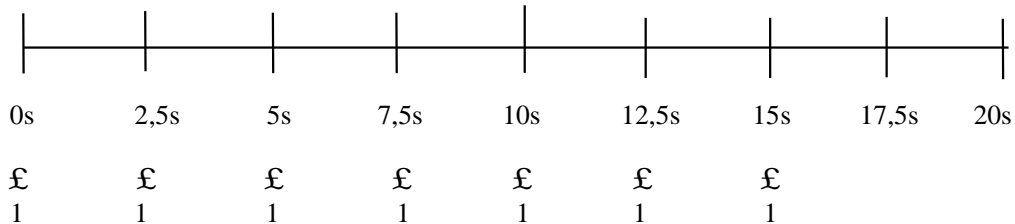
3.A. Hour stroke, half hour stroke + quarter strokes (sequences)

On the program form, notice that the hour stroke and the half hour stroke are played on bell 1, while the quarter strokes are played on bell 2. That means that you have to record 2 sequences before you can program the hour stroke, the half hour stroke and the quarter strokes.

3.A.1 Hour stroke + half hour stroke

Stroke bell 1

– The symbol £ indicates the stroke; the number underneath indicates the bell.



Notice that this sequence consists of a few strokes per minute with always the same pause (2,5 seconds) in between. Therefore, it is sufficient to record one stroke with one pause and then repeat this 7 times. That is what we will do.

Start the recording of a sequence: as mentioned before, you will do this by entering function 0601 and pressing "OK". That leads you to the screen with the main functions.

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Now you can record the unique stroke in function 0106 (manual tolling). For this, you have to enter function 06010106. In doing this, you can proceed to the following data on the screen:

MANUAL TOLLING	
PRESS KEY 1 @ 0	
OR KEYBOARD	OK
00 : 00 00 : 01.4	PA+OK

On the bottom line, you find two different counters. The left counter indicates the total duration of the sequence (which is still 0 at the moment). The recording of the melody starts when we press the first key. The right counter indicates the time span after the last stroke (or, when you haven't entered any strokes yet, the span after starting up the sequence).

Now you can key in a stroke by pressing on the right numerical key. When you want to strike bell 1, you have to press the numerical key '1'. When you have entered the key, there are 2 possibilities to quit this function. By pressing "PA+OK", the pause between the stroke and the moment that you quit the function will be recorded. When you don't want to insert a pause, you press "OK". Because you want exactly 2,5 seconds between the strokes, you quit the function by means of "OK".

In both cases, the following data will appear on the screen.

FUNCTION 06010105	
PROGRAM BLOCK	
FUNCTION 06010106	OK
MANUAL TOLLING	PA+OK

Now you can add the pause manually. Move to function 06010107 (pause). You can do this by pressing "DOWN" one time or "UP" 6 times.

With this function you can add a pause. A pause of 2,5 seconds can be entered as follows:

PAUSE	UP
25 1/10 SECONDS	DOWN
NUMBER	OK
	BACK

At this point you have recorded everything you wanted to record in this sequence. Finish the recording by pressing the key "BACK" until function 0601 appears.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

You can test the sequence. When you want to see the pause of this sequence, it's better to start the recording several times:

TEST SEQUENCE	UP
RECORD	DOWN
05 TIMES 0	OK
NAME SEQUENCE	BACK

When the recording is correct, you have to save it before recording a new sequence.

When you move to function 0603, the following data will appear:

SAVE RECORDING	UP
SEQUENCE 001	DOWN
NAME SEQUENCE	OK
	BACK

The recording can be saved in an empty sequence or in sequence that has already been used. When the sequence was already used, its contents will be overwritten. The sequences can also be grouped, e.g. melodies (sequences 01 to 10), angelus (sequence 11), hour stroke (sequence 12), ...

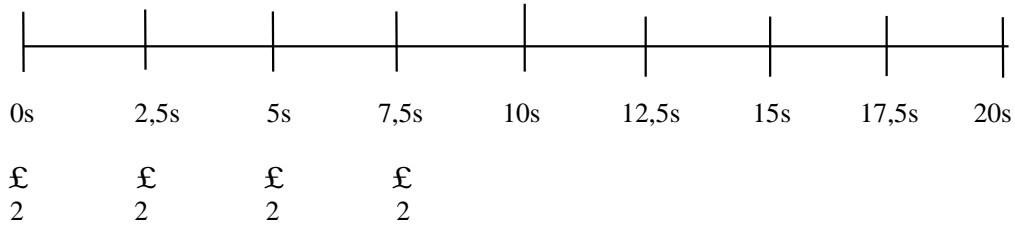
The APOLLO II device looks for the first empty sequence in its memory and shows this in the screen when we proceed to this function. As this is our first recording, this will become sequence 001. Of course, we can choose any other sequence to save our recording. But make sure not to overwrite any sequences you still need. Press "OK" to save the sequence. Press "BACK" to quit the function without saving the sequence.

To make sure that you remember the contents of a specific sequence, you can change the name of "sequence 001".

After choosing in which sequence we want to store our recording, the APOLLO II device will ask you to enter a name for this sequence. In our example, we can change the name of the sequence. Change the name into 'stroke bell 1'.

3.A.2 Quarter strokes

Stroke bell 2



For this sequence, you have to follow the same procedure as for the ‘stroke bell 1’ sequence. The only difference is that you have to press numeric key ‘2’ in stead of key ‘1’ in function 06010106 (manual tolling).

Store the sequence in ‘sequence 002’ and call it ‘stroke bell 2’.

3.B. Hour stroke, half hour stroke + quarter strokes (programs)

You have to move to function 040202 (permanent sequence programs) and choose an empty program (you can recognize an empty program by the signal ‘NEVER’ as a choice for the parameter ‘day of the week’).

In the next section, you find the programs that have to be entered to program the different hour strokes. For the desired programming (hour stroke, half hour stroke, 1st quarter, 2nd quarter, 3rd quarter, 4th quarter), you have to enter 6 different programs.

You will have to do this in the 6 permanent sequence programs (choose sequence programs 1 until 6 included).

3.B.1. Hour stroke programming (sequence program 1)

SEQ.PROGR.	1		
STROKE BELL 1		UP	(1)
01 HOUR STROKE		DOWN	(2)
AUTOM. HOUR STROKE		OK	(3)
BEGIN 08 22 00 00		BACK	(4)
DAILY NORMAL			(5)
01 JANUARY ..			(6)
31 DECEMBER ..			(7)

- (1) The sequence ‘STROKE BELL 1’ is programmed.
- (2) The special unit ‘HOUR STROKE’ makes sure that the APOLLO II device calculates how many strokes are needed for itself (e.g. 5 strokes at 17pm, 9 strokes at 21pm).
- (3) By means of the program block ‘AUTOM. HOUR STROKE’, the end customer will be able to switch the program ON and OFF.
- (4) 08 22 00 00: that means that the first execution takes place at **08 22 00 00 (8 am)** and the last execution at **08 22 00 00 (22 pm)**. The signal ‘BEGIN’ makes sure that the program always starts at **08 22 00 00 (00 minutes 00 seconds, so exactly on the hour)**.
- (5) DAILY: this program functions from Monday to Sunday – NORMAL: the program does not depend on Easter, summer- or wintertime.

- (6) + (7) The program is executed from the 01st of January until the 31st of December – so every Monday until Sunday (5) of the year (no special period has been programmed).

3.B.2. Half hour stroke programming (sequence program 2)

SEQ.PROGR.	2		
STROKE BELL 1		UP	(1)
01 TIMES			(2)
AUTOM. HOUR STROKE		DOWN	(3)
BEGIN 08 21 30 00			(4)
DAILY NORMAL		OK	(5)
01 JANUARY ..			(6)
31 DECEMBER ..		BACK	(7)

- (1) The sequence 'STROKE BELL 1' is programmed.
- (2) 1 stroke has to be given (the sequence, in which stroke 1 is recorded, is played 01 TIMES).
- (3) By means of the program block 'AUTOM. HOUR STROKE', the end customer will be able to switch the program ON and OFF.
- (4) 08 21 30 00: that means that the first execution happens at ~~08 21~~ **30 00** (8.30 am) and the last execution at ~~08 21 30 00~~ (~~21.30 pm~~). The signal 'BEGIN' makes sure that the program always starts at ~~08 21~~ **30 00** (30 minutes 00 seconds, so exactly on the half hour).
- (5) DAILY: this program functions from Monday to Sunday – NORMAL: the program does not depend on Easter, summer-or wintertime.
- (6) + (7) The program is executed from the 01st of January until 31st of December – so every Monday until Sunday (5) of the year (no special period has been programmed).

3.B.3. 1st and 3rd quarter stroke programming (sequence program 3 and 5)

SEQ.PROGR.	3		
STROKE BELL 2		UP	(1)
01 TIMES			(2)
AUTOM. QUARTERS		DOWN	(3)
BEGIN 08 21 15 00			(4)
DAILY NORMAL		OK	(5)
01 JANUARY ..			(6)
31 DECEMBER ..		BACK	(7)

- (3) By means of the program block 'AUTOM. QUARTERS', the end customer will be able to switch the program ON and OFF.
- (4) 08 21 15 00: that means that the first execution happens at ~~08 21~~ **15 00** (8.15 am) and the last execution at ~~08 21 15 00~~ (~~21.15 pm~~). The signal 'BEGIN' makes sure that the program always starts at ~~08 21~~ **15 00** (15 minutes 00 seconds, so exactly on the first quarter).

SEQ.PROGR.	5		
STROKE BELL 2		UP	(1)
03 TIMES			(2)
AUTOM. QUARTERS		DOWN	(3)
BEGIN 08 21 45 00			(4)
DAILY NORMAL		OK	(5)
01 JANUARY ..			(6)
31 DECEMBER ..		BACK	(7)

- (3) By means of the program block 'AUTOM. QUARTERS', the end customer will be able to switch the program ON and OFF.
- (4) 08 21 45 00: that means that the first execution happens at ~~08 21~~ **45 00** (8.45 am) and the last execution at ~~08 21 45 00~~ (21.45 pm). The signal 'BEGIN' makes sure that the program always starts at ~~08 21~~ **45 00** (15 minutes 00 seconds, so exactly on the third quarter).

3.B.4. 2nd and 4th quarter stroke programming (sequence program 4 and 6)

SEQ.PROGR.	4		
STROKE BELL 2		UP	(1)
02 TIMES			(2)
AUTOM. QUARTERS		DOWN	(3)
END 08 21 29 57			(4)
DAILY NORMAL		OK	(5)
01 JANUARY ..			(6)
31 DECEMBER ..		BACK	(7)

- (3) By means of the program block 'AUTOM. QUARTERS', the end customer will be able to switch the program ON and OFF.
- (4) 08 21 29 57: that means that the first execution ends at ~~08 21~~ **29 57** (8h 29m 57 s am) and the last execution at ~~08 21 29 57~~ (21h 29m 57s pm). The signal 'END' makes sure that the program always starts at ~~08 21~~ **29 57** (29 minutes 57 seconds, so before the half hour stroke – program 2 begins). That is important, because otherwise, the half hour stroke will not be executed. In case you don't have a half hour stroke, you can program with BEGIN 08 21 30 00.

SEQ.PROGR.	6		
STROKE BELL 2		UP	(1)
04 TIMES			(2)
AUTOM. QUARTERS		DOWN	(3)
END 07 21 59 57			(4)
DAILY NORMAL		OK	(5)
01 JANUARY ..			(6)
31 DECEMBER ..		BACK	(7)

- (3) By means of the program block 'AUTOM. QUARTERS', the end customer will be able to switch the program ON and OFF.
- (4) 07 21 59 57: that means that the first execution happens at ~~07 21~~ **59 57** (7h 59m 57s am) and the last execution at ~~07 21 59 57~~ (21h 59m 57s pm). The signal 'BEGIN'

makes sure that the program always starts at ~~07 21~~ **59 57** (59 minutes 57 seconds, so before the hour stroke – program 1 begins). That is important, because otherwise, the hour stroke will not be executed. In case you don't have an hour stroke, you can program with BEGIN 07 22 00 00.

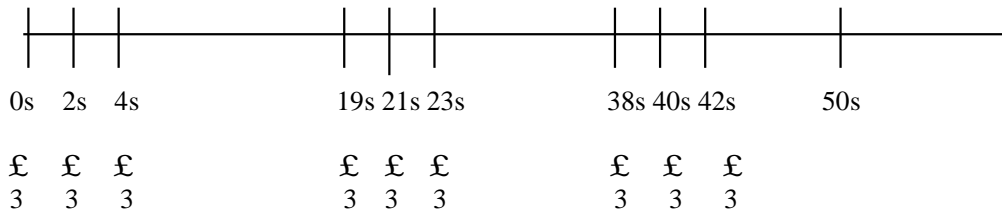
3.C. Angelus

3.C.1 Sequence

The angelus has to be programmed 3 times a day. Before we can program the angelus, we have to record the sequence 'angelus'.

Angelus (strokes on bell 3 + ringing on bell 2)

– The symbol £ indicates the stroke; the number underneath indicates the bell.



RINGING 2

The angelus consists of 3x3 strokes on bell 3, followed by one minute of ringing on bell 2.

Start the recording of a sequence: as mentioned before. Do this by entering function 0601 and pressing "OK". That leads you to the screen with the main functions.

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Press on "OK" to enter the function "EXECUTIONS". In the function "MANUAL TOLLING" (06010106), you can strike 9 times at once (with the exact timing for the angelus). In doing this, you can see the following data appear on the screen:

MANUAL TOLLING	
PRESS KEY 1® 0	
OR KEYBOARD	
00 : 00	00 : 01.4
	OK
	PA+OK

Because you want to give strokes on bell 3, you have to press key "3" 9 times. You have to wait before pressing the next key until the desired pause after the previous stroke has been reached.

On the bottom line, you see two different counters. The left counter indicates the total duration of the sequence (which is still 0 at the moment). The recording of the melody starts

when we press the first key. The right counter indicates the time span after the last stroke (or, when you haven't entered any strokes, the span after starting up the sequence).

When you have entered all the strokes, there are 2 possibilities to quit this function. By pressing "PA+OK", the pause between the stroke and the moment that you quit the function will be recorded. When you don't want to insert a pause, you press "OK". Because you want exactly 8 seconds between the strokes and the ringing, you quit the function by means of "OK".

In both cases, the following data will appear on the screen:

FUNCTION 06010105	UP
PROGRAM BLOCK	DOWN
FUNCTION 06010106	OK
MANUAL TOLLING	BACK

Now you can enter the pause manually. Move to function 06010107 (pause). You can do this by pressing "DOWN" one time or "UP" 6 times.

This function allows you to enter a pause. You need a pause of 8 seconds between the last stroke and the beginning of the ringing.

PAUSE	UP
8 SECONDS	DOWN
NUMBER	OK
	BACK

After having entered the pause, you can start the minute of ringing with bell 2. This can be done in function 06010101 (ringing).

RINGING	+1
2.....	-1
01 MINUTES	OK
WITH WHICH BELL	BACK

At this moment all the data required in this sequence has been entered. You end the recording by pressing the 'BACK' key until you are in function 0601.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

Now, you can test your sequence (function 0602):

TEST SEQUENCE	UP
RECORD	DOWN
05 TIMES 0	OK
NAME SEQUENCE	BACK

When the recording is correct, you have to save it (function 0603) before recording a new sequence.

3.C.2. Programs

Proceed to function 040202 (permanent sequence programs) and then choose an empty program (you can recognize an empty program by means of the signal 'NEVER' as value for the parameter 'day of the week').

In the following table you find the programs that have to be entered to program the angelus. Because the angelus has to be executed 3 times a day, we have to establish 3 programs (therefore choose sequence programs 7 until 9 included) :

SEQ.PROGR.	7		
ANGELUS		UP	(1)
01 TIMES		DOWN	(2)
AUTOM. ANGELUS		OK	(3)
BEGIN .. 06 01 00		BACK	(4)
MON-SAT NORMAL			(5)
01 JANUARY ..			(6)
31 DECEMBER ..			(7)

- (1) The sequence 'ANGELUS' is programmed
- (2) The angelus has to be played 01 TIMES at one precise moment.
- (3) By means of the program block 'AUTOM. ANGELUS', the end customer will be able to switch the program ON and OFF.
- (4) . . 06 01 00: that means that the angelus is executed at 06h 01m 00s am (the '. .' means that only the 3 following numbers are significant). Notice that we program the angelus 1 minute after the hour. If we would program the angelus on the exact hour (06 00 00), the hour stroke would be canceled. If we would program at 06 01 00, we are sure that the hour stroke has already been executed.
- (5) MON-SAT: this program functions from Monday to Saturday – NORMAL: the program does not depend on Easter, summer-or wintertime.
- (6) + (7) The program is executed from the 01st of January until 31st of December – so every Monday until Sunday (5) of the year (no special period has been programmed).

The two other programs are identical:

SEQ.PROGR.	8		
ANGELUS		UP	(1)
01 TIMES		DOWN	(2)
AUTOM. ANGELUS		OK	(3)
BEGIN .. 12 01 00		BACK	(4)
DAILY NORMAL			(5)
01 JANUARY ..			(6)
31 DECEMBER ..			(7)

SEQ.PROGR. 9		
ANGELUS	UP	(1)
01 TIMES		(2)
AUTOM. ANGELUS	DOWN	(3)
BEGIN .. 18 01 00		(4)
DAILY NORMAL	OK	(5)
01 JANUARY ..		(6)
31 DECEMBER ..	BACK	(7)

3.D. Week-, Saturday- and Sunday masses (programs)

The masses, described on the program form, can be programmed directly in the there fore provided permanent ringing programs intended for that purpose (function 040201). So we don't need to make sequences.

Beneath, you find the programs that have to be established for programming the masses. Because you have to establish 3 masses a week, you have to program 3 ringing programs (choose here fore ringing programs 1 until 3 included).

RINGING PROGR. 1		
3 4	UP	(1)
02 MINUTES		(2)
AUTOM. MASSES	DOWN	(3)
08 55 00		(4)
WEEK DAYS NORMAL	OK	(5)
01 JANUARY ..		(6)
31 DECEMBER ..	BACK	(7)

- (1) In this program, ringing bells 3 and 4 are programmed. The following order '34' determines that first bell 3 will start and then bell 4.
- (2) The duration of the ringing is 02 minutes.
- (3) By means of the program block 'AUTOM. MASSES', the end customer will be able to switch the program ON and OFF himself.
- (4) 08 55 00: that means that the ringing will start at 08h 55m 00s am.
- (5) WEEK DAYS: this program is executed every day of the week (Monday to Friday) – NORMAL: the program does not depend on Easter, summer-or wintertime.
- (6) + (7) The program is executed from the 01st of January until 31st of December – so every Monday until Sunday (5) of the year (a special period has not been programmed)

The Saturday and Sunday mass is programmed in the same way.

RINGING PROGR. 2		
2 3 4	UP	(1)
03 MINUTES		(2)
AUTOM. MASSES	DOWN	(3)
17 55 00		(4)
SATURDAY NORMAL	OK	(5)
01 JANUARY ..		(6)
31 DECEMBER ..	BACK	(7)

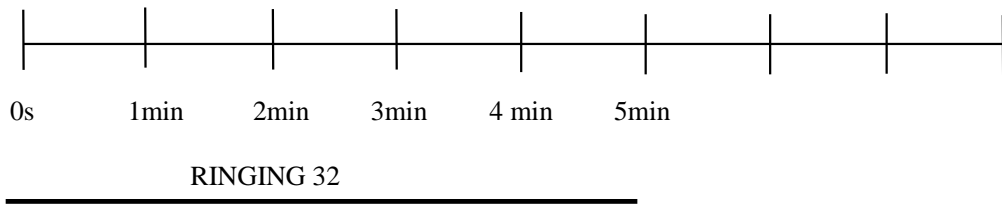
RINGING PROGR. 3 2 3 4 03 MINUTES	UP	(1)
AUTOM. MASSES 09 55 00	DOWN	(2)
SUNDAY NORMAL	OK	(3)
01 JANUARY ..		(4)
31 DECEMBER ..	BACK	(5)
		(6)
		(7)

3.E. Funeral, marriage (sequence)

These executions have to be started up manually. Therefore, we will record them in a sequence. This will make it easier for the end customer to search this sequence in the user menu and to start it up by pressing one key.

3.E.1. Funeral

Funeral (ringing 32 for 5 minutes)



Start the recording of the sequence (function 0601).

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

As an execution (ringing) is started up in this sequence, you have to press “OK” again. The following data will appear on the screen:

FUNCTION 06010101	UP
RINGING	DOWN
FUNCTION 06010102	OK
PLAY SEQUENCE	BACK

You can enter the function and look at the following data:

RINGING	+1
!	-1
01 MINUTES	OK
WITH WHICH BELL	BACK

Key in the following data:

RINGING 3 2 05 MINUTES WITH WHICH BELL	+1 -1 OK BACK
---	--

Confirm with 'OK'.

FUNCTION 06010101 RINGING FUNCTION 06010102 PLAY SEQUENCE	UP DOWN OK BACK
--	--

Because you want the text 'funeral' to appear in the main menu for 5 minutes (the duration of the ringing), you add a pause of 5 minutes in this sequence. Therefore, you have to move to function 'PAUSE' and press 'OK'.

FUNCTION 06010106 MANUAL TOLLING FUNCTION 06010107 PAUSE	UP DOWN OK BACK
---	--

You can now enter a pause of 5 minutes.

PAUSE 05 MINUTES NUMBER	UP DOWN OK BACK
--	--

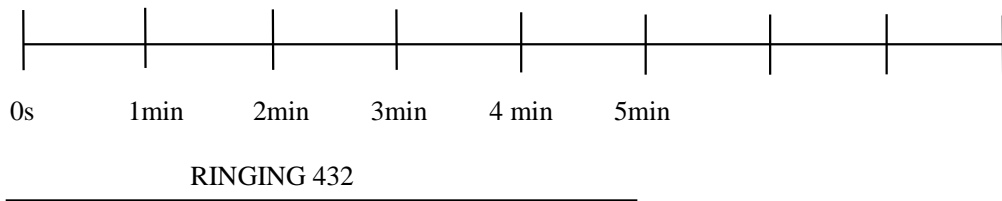
At this point you have recorded all data required in this sequence. Finish the recording by pressing the key "BACK" until you see function 0601.

FUNCTION 0601 RECORD SEQUENCE FUNCTION 0602 TEST SEQUENCE	UP DOWN OK BACK
--	--

You can test the sequence (function 0602) and save (function 0603) it. Call this sequence 'FUNERAL'. Later on you will place this sequence in the user menu.

3.E.2. Marriage

Marriage (ringing 432 for 5 minutes)

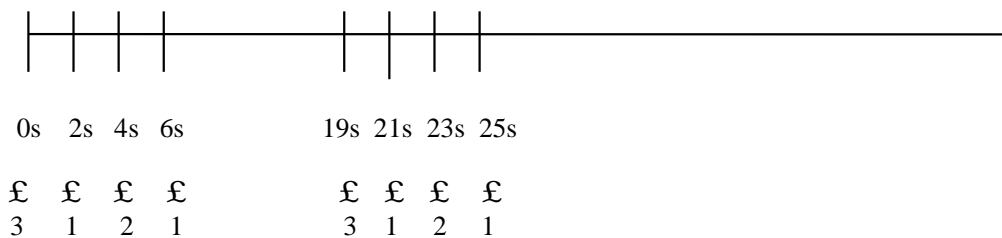


The sequence 'MARRIAGE' is identical to the sequence 'FUNERAL'. The only difference is that you have to choose 'ringing' the bells '432' in stead of '32' in the screen.

Record the sequence (function 0601), test it (function 0602) and store it afterwards in function 0603. Call this sequence 'MARRIAGE'. We will also place this sequence in the user menu.

3.E.3. Mourning tolling

– The symbol £ indicates the stroke; the number underneath indicates the bell.



We can record this sequence by entering 4 strokes and 4 pauses:

- * stroke on bell 3
- * pause of 2 seconds
- * stroke on bell 1
- * pause of 2 seconds
- * stroke on bell 2
- * pause of 2 seconds
- * stroke on bell 1
- * pause of 13 seconds

Start recording the sequence:

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Now you can record the first stroke in function 0106 (manual tolling). For that you have to move to function **06010106**. The following data will appear on the screen:

MANUAL TOLLING	
PRESS KEY 1® 0	
OR KEYBOARD	OK
00 : 00 00 : 01.4	PA+OK

Because you want to give a stroke on bell 3, you have to press key "3" 1 time and quit the function by means of 'OK'.

Now you can add a pause of 2 seconds. For that you have to move to function 06010107 (pause).

PAUSE 02 SECONDS NUMBER	UP DOWN OK BACK
--	--

Afterwards, you can go back to function 0106 (manual tolling), to establish the second stroke by pressing key '1' to record a stroke on bell 1. Then, add a new pause after this. Repeat this procedure for the last two strokes and pauses. Then go back to function 0601 by means of the 'BACK' key. Test (function 0602) and store (function 0603) the sequence. Call the sequence 'MOURNING TOLLING'. You will place it in the user menu.

3.F. Tower clock

Change hands sequence type 3

The function of the 'STOP' key during the recording of a sequence, can also be used to make sequences that change parameters (sequence type 3). The example shows us how to make a *sequence* that makes it possible to correct the tower clock. The creation of this sequence is only necessary when you decide that the *end customer* has to be able to do it on his own. Of course the installer can do this directly in function 120201.

Start recording the sequence.

FUNCTION 060101 EXECUTIONS FUNCTION 060102 USER MENU	UP DOWN OK BACK
---	--

In this sequence, no executions are started up, but the content of function 120201 is required. Move to function 060112 'clocks'.

FUNCTION 060112 CLOCKS FUNCTION 060113 RELAYS	UP DOWN OK BACK
--	--

Press "OK" and choose 'slave clock 1' (function 06011202).

FUNCTION 06011201 INTERNAL CLOCK FUNCTION 06011202 SLAVE CLOCK 1	UP DOWN OK BACK
---	--

Press "OK" and choose 'time slave clock 1' (function 06011202).

FUNCTION 06011202		UP
TIME SLAVE CLOCK 1		DOWN
FUNCTION 06011202		OK
TYPE SLAVE CLOCK 1		BACK

Press “OK” – the following data will appear:

TIME SLAVE CLOCK 1	+1
04 45	-1
CHANGE HOURS	OK
	BACK

Because you want to let the client enter the time of the slave clock himself, press the “STOP” key twice. You come back in the next screen:

FUNCTION 06011202		UP
TIME SLAVE CLOCK 1		DOWN
FUNCTION 06011202		OK
TYPE SLAVE CLOCK 1		BACK

You have recorded all data required in this sequence. Close the recording by means of the “BACK” key until you are back in function 0601.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

Now you can test our sequence (function 0602) and store it (function 0603). This sequence will be placed in the user menu.

STEP 04: Establishing the user menu + inputs

The user menu allows the customer to:

- 1) Switch program blocks on and off, such as e.g. ‘AUTOM. HOUR STROKE’. That way, the customer can block (cfr. 3.A.2) the programs in which these program blocks are used.
- 2) Start sequences manually, such as e.g. FUNERAL, MARRIAGE, and MOURNING TOLLING.

It is important to know how you want to organize the different lines in the user menu before you start to program.

In our example, the user menu could look like this:

Screen 1:

FUNERAL	PRESS⇒	User line 1
MARRIAGE	PRESS⇒	User line 2
MOURNING TOLLING	PRESS⇒	User line 3
1 NEXT SCREEN	PRESS⇒	

Screen 2:

AUTOM. HOUR STROKE	ON	User line 4
AUTOM. QUARTERS	ON	User line 5
AUTOM. ANGELUS	ON	User line 6
2 NEXT SCREEN	PRESS⇒	

Screen 3:

AUTOM. MASSES	PRESS⇒	User line 7
		User line 8
3 NEXT SCREEN	PRESS⇒	User line 9

Screen 4:

TOWER CLOCK	PRESS⇒	User line10
		User line11
4 NEXT SCREEN	PRESS⇒	User line12

Move to function 02. Here you can see 30 different subfunctions.

FUNCTION 0201	UP
USER MENU 01	DOWN
FUNCTION 0202	OK
USER MENU 01	BACK

Enter function 0201. At this point you can define the first item (user line 1) of the user menu. The following data will appear on the screen.

FUNCTION 020101	UP
USER TYPE 1	DOWN
FUNCTION 020102	OK
USER ACTION 2	BACK

You always have to establish the user type. Because you want to put 'FUNERAL' on the first line, you have to choose 'SEQUENCE' for user type 01. Enter function 020101 (user type). Remember that 'FUNERAL' is a 'SEQUENCE' that has already been recorded.

Afterwards, you can select in function 'user action 01' which sequence you want to put on the first user line. Enter function 020102 and key in the following:

USER ACTION 01	+1
FUNERAL	-1
01 TIMES	OK
NAME SEQUENCE	BACK

Afterwards, you can establish the next lines (e.g. user line 02). Go back to the screen below and choose function 0202 (user menu 2).

FUNCTION 0201	UP
USER MENU 1	DOWN
FUNCTION 0202	OK
USER MENU 2	BACK

Below, you find a list of how to establish function 02 to finish the example:

user line	appears as	user type	user action
1 (function 0201)	FUNERAL	sequence	"FUNERAL" 01 TIMES
2 (function 0202)	MARRIAGE	sequence	"MARRIAGE" 01 TIMES
3 (function 0203)	MOURNING TOLLING	sequence	"MOURNING TOLLING" 05 MINUTES *
4 (function 0204)	AUTOM. HOUR STROKE	program block	"AUTOM. HOUR STROKE" 01 PERMANENT
5 (function 0205)	AUTOM. QUARTERS	program block	"AUTOM. QUARTERS" 01 PERMANENT
6 (function 0206)	AUTOM. ANGELUS	program block	"AUTOM. ANGELUS" 01 PERMANENT
7 (function 0207)	AUTOM. MASSES	program block	"AUTOM. MASSES" 01 PERMANENT
8 (function 0208)		not used	
9 (function 0209)		not used	
10 (function 0210)	TOWER CLOCK	sequence	"TOWER CLOCK" 01 times
11 (function 0211)		not used	
12 (function 0212)		not used	

* The option '05 MINUTES' makes sure that the mourning tolling (sequence with 4 strokes + 4 pauses) is repeated for 05 minutes. If you don't know how long the customer wants the mourning to last, you can also choose for 'MOURNING TOLLING' .. MINUTES. The two '..' mean that the duration can be chosen by the end customer.

** The signal '01 PERMANENT' makes sure that the program block, e.g. AUTOM. HOUR STROKE functions as an ON/OFF switch. In pressing it, it will stay on PERMANENT until the moment you press it again. At that moment, it will stay off PERMANENTLY. Alternatively, you can program a program block, e.g. AUTOM. SUNDAY MASS' 01 WEEK. When the customer switches on this block, it will be switched off automatically after a week.

Function 03: when you want to use the inputs (external switch, remote control, DTMF interface, ...), you can determine in function 03 what is executed when a specific input is activated.

In this example, you have to program the following:

input	appears as	input type	input action
1 (function 0301)	FUNERAL	sequence	"FUNERAL" 01 TIMES
2 (function 0302)	MARRIAGE	sequence	"MARRIAGE" 01 TIMES
3 (function 0303)	MOURNING TOLLING	sequence	"MOURNING TOLLING" 05 MINUTES
4 (function 0304)	STOP	sequence	"STOP" 01 TIME ***

*** In order to give the customer the possibility to stop executions by means of an input, you can record and program a ‘STOP’ sequence.

This sequence is recorded as follows:

Function 0601 (record sequence)

Press ‘OK’

Press ‘STOP’

Press ‘BACK’, until you see function 0601 (record sequence).

Store the sequence (function 0603) and call it ‘STOP’.

STEP 05: programming set up parameters, like impulse duration hammers, tower clock

...

Function 0105: switch on/off program blocks

- You can switch the program blocks that are mentioned in the user menu on and off. To switch other program blocks on/off (e.g. your name or telephone number), you have to use function 0105 (executions – program blocks)
- Attention: to change the name of one of the program blocks, e.g. your name, you have to move to function 09 (program blocks).
- Don’t forget that you have to switch the blocks on permanently in function 0105, e.g. ‘YOUR NAME’ for ‘01 PERMANENT’

Function 05: ringing parameters

- Normal ringing or simulated ringing?
- Which hammer is installed on which bell?
- Ringing in and ringing out times: when you enter the ringing in and ringing out time for each bell, you obtain a lovely ringing pattern. The APOLLO II device itself determines at what moment the bells have to be activated in order to synchronize the first clapper stroke of each bell.
- The interval when activating several bells?
- Order of ringing out – when you start ringing with the smallest bell, you can choose to switch off the smallest or the largest bell first.

Function 10: hammer parameters

- For each hammer, the impulse duration and the delay can be programmed

Function 11: special outputs

- Duration for opening the louver-holes
- Switch on/off the Roman knell
- Transpose a MIDI keyboard (only Apollo II-SBSI)

- Parameters for the SUBD9 (serial) output

Function 0606: properties sequences

- You can modify the execution speed of each sequence,. Attention: in doing this, all durations (but also pauses) that are present in the sequence become shorter or longer.
- You can determine for each sequence whether or not the louver-hole has to be opened.

Function 1201: parameter for the internal clock

- Programming date, time (not necessary when using an antenna).
- Function 120103: time format.
- Function 120106: summer/winter time changeover .

Function 1202 and 1203: parameters for tower (slave) clocks

- Programming the time for the tower (slave) clock (120201 and 120301).
- Determining the impulse duration for the tower clock (120202 and 120302).

Function 14: general settings

- Function 1401: contrast display
- Function 1402: user code
- Function 1403: function code

We refer to the description in part C: SOFTWARE.

4. Sequences to change programs (sequence type 2)

4.1. Introduction

In this section, we give an example of how to record a sequence that offers the final customer the possibility to **change programs himself**. When the final customer never has to change the programs entered by you, the installer, you don't have to record these sequences.

While recording this sequence, you have to move to function 04 (programs). In this function, you have to establish all the parameters that you can fill out yourself. All the other parameters will be '*opened*' for the client by means of the 'STOP' key. The final customer can then set these parameters (e.g. days of the week) himself.

Example:

You want to create a sequence of which the final customer can choose the starting time of the first angelus program. The parameters **in bold** are the parameters that you can fill out yourself, because those never have to change. You have to 'open' the other parameters for the client, so that he can change these himself. In this case, it is only the hour, the minutes and the seconds.

SEQ.PROGR.	7	
ANGELUS		UP (1)
01 TIME		(2)
AUTO. ANGELUS		DOWN (3)
BEGIN .. 06 01 00		(4)
DAILY NORMAL		OK (5)
01 JANUARY ..		(6)
31 DECEMBER ..		BACK (7)

While recording **a sequence to change programs** you have to keep two important rules in mind:

RULE 1: always use the numerical keys to set the parameters yourself

When you set a parameter (entering the exact value), you have to do this by means of the **numerical keys** and not **only** by using the 'UP' and 'DOWN' keys. That is important because otherwise you would obtain a different value every time you run it.

When you proceed to a program, you see that it is programmed the way you left it. When you have e.g. programmed the angelus at 06.01 a.m., it means that this program has to stay present. When you proceed to the program another time, the angelus will still be programmed at 06.01 a.m.

A sequence where you only use 'UP' and 'DOWN' will increase or lower a parameter, but it will not result in exact (always the same) value:

Example:

In an empty program, the 'day of the week'-parameter has the value 'NEVER'. You want to establish 'DAILY' for our program. You can do this by pressing 'UP' 8 times.

NEVER	
MONDAY	1 x 'UP'
TUESDAY	2 x 'UP'
WEDNESDAY	3 x 'UP'
THURSDAY	4 x 'UP'
FRIDAY	5 x 'UP'
SATURDAY	6 x 'UP'
SUNDAY	7 x 'UP'
DAILY	8 x 'UP'
MON-FR	
WEEKEND	
MON -> SAT	
MOTUTHFR	

However, there is a problem: when you play the sequence a second time (e.g. with 'test sequence'), the 'day of the week'-parameter is already set at 'DAILY'. The sequence will increase this value again by means of 8 times 'UP' (we have recorded it like this).

NEVER	5 x 'UP'
MONDAY	6 x 'UP'
TUESDAY	7 x 'UP'
WEDNESDAY ←	8 x 'UP'
THURSDAY	
FRIDAY	
SATURDAY	
SUNDAY	
DAILY	
MON-FR	1 x 'UP'
WEEKEND	2 x 'UP'
MON -> SAT	3 x 'UP'
MOTUTHFR	4 x 'UP'

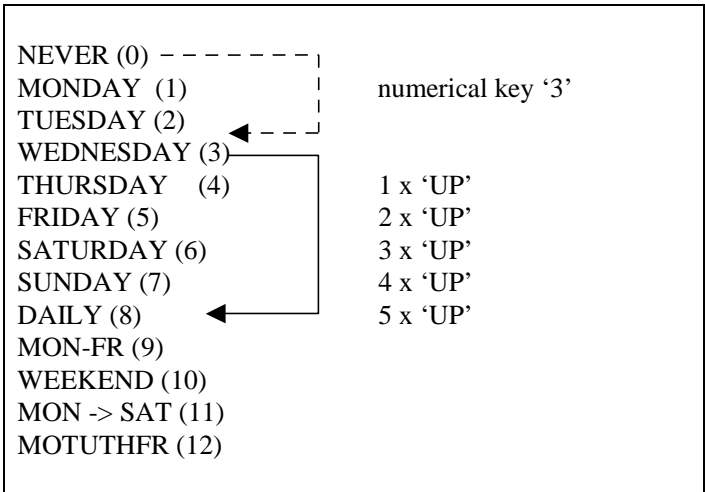
Because of the repeated increase the value is changed into 'WEDNESDAY'. That is wrong: you want it to be 'DAILY'.

Next time you run the sequence, you find another value (MON-> SAT) :

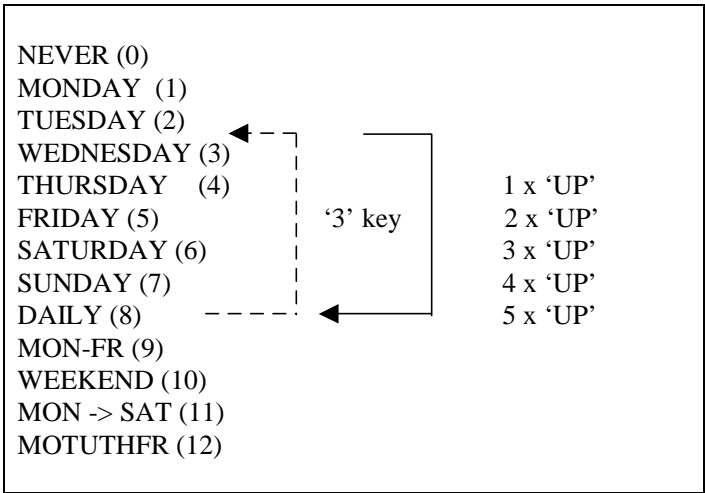
NEVER	
MONDAY	
TUESDAY	
WEDNESDAY	
THURSDAY	1 x 'UP'
FRIDAY	2 x 'UP'
SATURDAY	3 x 'UP'
SUNDAY	4 x 'UP'
DAILY	5 x 'UP'
MON-FR	6 x 'UP'
WEEKEND	7 x 'UP'
MON -> SAT ←	8 x 'UP'
MOTUTHFR	

Solution: By means of the numerical keys, you can determine a reference that always returns and that will always end with the same value. Every value has its own number (in this example, the numbers are indicated). When you don't know the numbers, you can always choose a number and press the 'UP' and/or 'DOWN' keys. Consider the same example:

In an empty program, the 'day of the week'-parameter has the value 'NEVER'. You want to establish 'DAILY' for your program. First of all, you have to press a numerical key (e.g. 3). That brings you to value 'Wednesday'. Then you have to press 'UP' another 5 times to obtain the value 'DAILY'.



When you play the sequence a second time (e.g. by means of 'test sequence'), the 'day of the week'-parameter will already be set at 'DAILY'. The parameter will go to value '3' (Wednesday) and then move to '5' positions 'UP' to 'DAILY'.



In other words, you always find the same value.

Always use a numerical key before you press the 'UP' and/or 'DOWN' keys when you record a sequence (when changing programs).

RULE 2: Set all the values that are known before pressing the first 'STOP'

4.2. Example 1: Single 'marriage' program

In this example you will record a sequence, which enables the final customer to program the sequence 'marriage', *which we have already recorded*. While you record this sequence, you have to adhere to the two rules mentioned on the previous pages.

Start recording the sequence (function 0601):

FUNCTION 060101 EXECUTIONS	UP
FUNCTION 060102	DOWN
USER MENU	OK
	BACK

Since you want to change a program in this sequence, you have to move to function 060104 (programs) and press 'OK'. The following data appears on the screen:

FUNCTION 06010401 SINGLE PROGR.	UP
FUNCTION 06010402	DOWN
PERMANENT PROGR.	OK
	BACK

You want to change a single program. Press 'OK' again.

FUNCTION 06010402 SG. RINGING PROGR.	UP
FUNCTION 06010402	DOWN
SEQUENCE PROGR.	OK
	BACK

Because you want to program the *sequence* 'marriage', you have to use a single sequence program. Choose 'sequence progr.' and press 'OK'.

SG. SEQ. PROGR. 01	UP
STROKE BELL 1	DOWN
01 TIMES	OK
00 00 NEVER	BACK

When you want to use the single sequence program nr. 1, press 'OK'. If this sequence was already used, you can choose another program by means of 'UP' and/or 'DOWN' and press 'OK'. The following data appears on the screen:

STROKE BELL 1	UP
01 TIMES	DOWN
00 00 NEVER	OK
NAME SEQUENCE	BACK

At this moment, you have to adhere to the two basic rules:

RULE 2: Set all the known values, before you press 'STOP'.

RULE 1: Set a known value by means of the numerical keys.

You have to set the program as follows (the parameters in bold are the parameters that you can establish ourselves, the parameters in *italic* are the parameters that you have to 'open' for the customer).

MARRIAGE	UP
01 TIMES	DOWN
<i>00 00 NEVER</i>	OK
<i>NAME SEQUENCE</i>	BACK

Notice that you can fill out the first 3 parameters yourself. You have to 'open' the hour, the minutes and the day for the customer. The first 3 parameters have to be set by means of the numerical keys (RULE 1), before you press the 'STOP' key to open the hour, the minutes and the day (RULE 2). Follow this procedure:

- **NAME SEQUENCE:** Press a numerical key (1 to 0 included). In doing this, you have set a reference for the 'name sequence'-parameter. At this moment, you can look for 'marriage' by means of the 'UP' and 'DOWN' keys. (if you know in which sequence 'marriage' is recorded, you can enter this directly). Press 'OK' to move to the next parameter.

Notice that '01 times' is already entered. However, it is safer to enter this a second time when you record the sequence. That way, you can be absolutely sure that what the client enters in the program is correct.

- **NUMBER:** Enter the '01' by pressing '0' first and then '1'. You proceed automatically to the next parameter (you can also just press a numerical key and then use 'UP' and 'DOWN' to establish 01).
- **UNIT:** Press a numerical key (e.g. 1). You will see that the 'unit'-parameter staggers from 'times' to 'seconds' and that you have proceeded to the hour parameter. Press 'BACK' to move to the 'unit' parameter. Because a reference for this parameter has already been entered, you can change the 'seconds' into 'times' by means of 'UP' and 'DOWN'. Press 'OK' to move to the next parameter.
- To open the 3 last parameters, you have to press the 'STOP' key 3 times. That way, the customer will be able to enter these parameters himself.

At this point you have recorded everything you wanted to record in this sequence. Finish the recording by pressing the key "BACK" until you see function 0601.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

You can test (function 0602) and store (function 0603) the sequence. Name this sequence 'SG. MARRIAGE PR.'. You can place this sequence in the user menu (type = sequence, action = example beneath).

USERACTION 17	+1
SG. MARRIAGE PR.	-1
01 TIMES	OK
NAME SEQUENCE	BACK

4.3. Example 2: Single ‘mourning tolling’ program

In this example you want to record a sequence, which enables the final customer to program the sequence ‘mourning tolling’, *which we have already recorded*. While you record this sequence, you have to stick to the two rules mentioned in the previous pages.

Start recording the sequence (function 0601):

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Since you want to change a program in this sequence, you have to move to function 060104 (programs) and press ‘OK’. You will see the following data on the screen:

FUNCTION 06010401	UP
SINGLE PROGR.	DOWN
FUNCTION 06010402	OK
PERMANENT PROGR.	BACK

You want to change a single program. Press ‘OK’ again.

FUNCTION 06010402	UP
SG. RINGING PROGR.	DOWN
FUNCTION 06010402	OK
SEQUENCE PROGR.	BACK

Because you want to program the *sequence* ‘mourning tolling’, you have to use a single sequence program. Choose ‘sequence progr.’ and press ‘OK’.

SG. SEQ. PROGR. 01	UP
STROKE BELL 1	DOWN
01 TIMES	OK
00 00 NEVER	BACK

Because the single sequence program nr. 1 is already used, you have to choose the single sequence program nr. 2 (to this end press ‘DOWN’ 1 time) and press ‘OK’. The following data will appear on the screen:

STROKE BELL 1	UP
01 TIMES	DOWN
00 00 NEVER	OK
NAME SEQUENCE	BACK

At this moment, you have to keep up the two basic rules:

RULE 2: Set all the known values, before you press ‘STOP’.

RULE 1: Set a known value by means of the numerical keys.

You have to fill out the program as follows (the parameters in bold are the parameters that you can set ourselves, the parameters in italic are the parameters that you have to 'open' for the customer).

MOURNING TOLLING	UP
<i>01 TIMES</i>	DOWN
<i>02 00 NEVER</i>	OK
<i>NAME SEQUENCE</i>	BACK

Notice that you can set the two parameters (name sequence and unit) yourself. Open the duration of the mourning tolling, the hour, the minutes and the day for the customer. The two parameters that you know have to be set by means of the numerical keys (RULE 1), before you press the 'STOP' key to open the hour, the minutes and the day (RULE 2). It goes as follows:

- **NAME SEQUENCE:** Press a numerical key (1 to 0 included). In doing this, you have entered a reference for the 'name sequence' parameter. At this moment, you can look for 'mourning tolling' by means of the 'UP' and 'DOWN' keys. (if you know in which sequence 'mourning tolling' is recorded, you can of course enter this directly). Press 'OK' to move to the next parameter.
- **NUMBER :** Confirm '01' temporarily by pressing 'OK'. You first have to set the 'unit' parameter to 'minutes', before opening the first parameter, taking into account rule 2.
- **UNIT :** Press a numerical key (e.g. 1). You will see that the 'unit'-parameter staggers from 'times' to 'seconds' and that you have proceeded to the hour parameter. Press 'BACK' to move back to the 'unit' parameter. Because a reference for this parameter is already entered, you can change the 'seconds' into 'minutes' by means of 'UP' and 'DOWN'. Press 'OK' to move to the next parameter.
- Now you can open the other parameters for the customer. For that purpose, return to the 'NUMBER' parameter (01) and press 'STOP'. Press 'OK' to pass to the hour parameter and press another 3 times 'STOP' to 'open' the hour, the minutes and the day for the customer.

Now you have recorded everything you wanted to record in this sequence. Finish the recording by pressing the "BACK" key until you see function 0601.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

You can test (function 0602) and store (function 0603) the sequence. Call this sequence 'SG. MOURNING PR.'. You can place this sequence in the user menu (type = sequence, action = example beneath).

USERACTION 18	+1
SG. MOURNING PR.	-1
1 TIMES	OK
NAME SEQUENCE	BACK

4.4. Example 3 : ‘Permanent ringing program’ sequence

In this example you want to record a sequence, which enables the final customer to adapt/modify/delete his own permanent ringing programs. While you record this sequence, you have to adhere to the two rules that we mentioned above.

Start recording the sequence (function 0601):

FUNCTION 060101	UP
EXECUTIONS	DOWN
FUNCTION 060102	OK
USER MENU	BACK

Since you want to change a program in this sequence, you have to move to function 060104 (programs) and press ‘OK’. The following data will appear on the screen:

FUNCTION 06010401	UP
SINGLE PROGR.	DOWN
FUNCTION 06010402	OK
PERMANENT PROGR.	BACK

You want to change a permanent program. So you have to choose ‘PERMANENT PROGR.’ and press ‘OK’.

FUNCTION 06010402	UP
RINGING PROGR.	DOWN
FUNCTION 06010402	OK
SEQUENCE PROGR.	BACK

Because you want the customer to enter the ringing programs, you have to use the permanent ringing programs. Press ‘OK’ again. The following data will appear on the screen:

RINGING. PROGR.	1	
.....	UP	(1)
01 MINUTES	DOWN	(2)
BLOCK INDEPEND.	OK	(3)
00 00 00	BACK	(4)
NEVER NORMAL		(5)
01 JANUARY ..		(6)
31 DECEMBER ..		(7)

At this moment, you have to keep the two basic rules in mind:

RULE 2: Set all the known values, before you press ‘STOP’.

RULE 1: Set a known value by means of the numerical keys.

You have to fill out the program as follows (the parameters in bold are the parameters that you can fill out yourself, the parameters in italic are the parameters that you have to ‘open’ for the customer).

RING. PROGR.	1	
3 4	UP	(1)
02 MINUTES		(2)
AUTO. MASSES	DOWN	(3)
08 55 00		(4)
<i>MON-FR</i> <i>NORMAL</i>	OK	(5)
01 <i>JANUARY</i> ..		(6)
31 <i>DECEMBER</i> ..	BACK	(7)

In this example, you will set the unit to ‘MINUTES’ and the program block to ‘AUTO. MASSES’. The other parameters will be ‘opened’ for the client.

Note: if the customer doesn’t want to program according to date, you can also confirm the date ‘01 JANUARY.. until 31 DECEMBER ..’ yourself.

The parameters that you know yourself have to be entered by means of the numerical keys (RULE 1), before you press the ‘STOP’ key to open the duration, the hour, the minutes and the day (RULE 2).

Exception to RULE 2 : if you want the customer to be able to enter several ringing programs, you can open the ringing program number by pressing ‘STOP’. If he only wants to enter one single program, you choose a program and press ‘OK’. You can give this ‘STOP’ before you have entered the other parameters.

Afterwards, you will end up in the first ‘with which bell’ parameter.

- ‘WITH WHICH BELL’: of these 10 parameters (10 points) you will open three later (we suppose that we only need 3 ringing bells), after having set the other parameters. First press ‘OK’ 10 times.
- ‘NUMBER’: For the time being, confirm ‘01’ with ‘OK’. Later, you will open this parameter for the client.
- ‘UNIT’: Press a numerical key (e.g. 1). You will see that the ‘unit’-parameter staggers from ‘times’ to ‘seconds’ and that you have to proceed to the ‘program block’ parameter. Press ‘BACK’ to move back to the ‘unit’ parameter. Because a reference for this parameter is already entered, you can change the ‘seconds’ into ‘minutes’ by means of ‘UP’ and ‘DOWN’. Press ‘OK’ to move to the next parameter.
- ‘PROGRAM BLOCK’: Press a numerical key (e.g. 1). Then use the ‘UP’ and ‘DOWN’ keys to change the parameter into ‘AUTO. MASSES’ (Note: the ‘AUTO. MASSES’ signal is only present after you have entered that name in function 09).
- Now you have entered all the parameters that are familiar. You can open all the other parameters for the client. First return to the first ‘WITH WHICH BELL’ parameter. Because you want the customer to choose maximum three bells, press ‘STOP’ 3 times. The other 7 points are confirmed with ‘OK’. The ‘number’ (01)

parameter can be opened by means of the 'STOP' key. The unit 'minutes' and the program block 'AUTO. MASSES' are confirmed with 'OK'. (press 2 x). Press 'STOP' another 11 times to open the other parameters for the customer as well.

At this point, you have recorded everything you wanted to record in this sequence. Finish the recording by pressing the key "BACK" until you see function 0601.

FUNCTION 0601	UP
RECORD SEQUENCE	DOWN
FUNCTION 0602	OK
TEST SEQUENCE	BACK

You can test (function 0602) and store (function 0603) the sequence. Call this sequence 'PERM. RINGING PR.'. You can place this sequence in the user menu (type = sequence, action = example beneath).

USERACTION 18	+1
PERM. RINGING PR.	-1
03 MINUTES	OK
NAME SEQUENCE	BACK

5. Automatic “holiday” - program

5.1. Introduction

Plan: the normal ringing programs are active the whole year long, (week- and weekend masses). These normal programs are blocked during holidays (and eventually also the day before the holiday) and replaced by special ‘holiday’ and ‘pre-holiday’ (the day before a holiday) programs.

We can program the above-mentioned summary by means of program blocks. When we make the several ringing programs dependent on these program blocks, we can easily program the holidays by switching the program blocks on and off.

EXAMPLE:

	1	2	3
	YEAR PROGRAM	PRE-HOLIDAY	HOLIDAY
RINGING (WHAT)	Ringing with bell 3 for 5 minutes	Ringing with bell 3 and 2 for 5 minutes	Ringing with bell 3, 2 and 1 for 5 minutes
TIME (WHAT)	1) 17.45 week day 2) 18.45 Saturday 3) 07.45 Sunday 4) 09.45 Sunday 5) 19.45 Sunday	1) 18.45 daily	1) 07.45 daily 2) 09.45 daily 3) 19.45 daily
HOLIDAY DATA (WHEN)		14/08 23/12 Easter Saturday ...	Ascension Day (15/08) Christmas (24/12) Easter ...

The table shows that there are 3 different ringing patterns in our example:

1. YEAR PROGRAM: during ‘the normal days’ of the year, bell 3 rings for 5 minutes at the indicated times.
2. PRE-HOLIDAY: on days before a holiday, bells 3 and 2 ring for 5 minutes (the YEAR PROGRAM have to be blocked).
3. HOLIDAY: on holidays, all 3 bells ring for 5 minutes (the YEAR PROGRAM have to be blocked).

To program this example, you have to program two different items, namely ‘WHAT’ and ‘WHEN’:

WHAT: has to be executed when a certain pattern is active. That means that you have to key in which ringing circuits have to be executed when a certain pattern is active.

For example: during ‘HOLIDAY’ bells 3, 2 and 1 have to ring for 5 minutes at 07h45, 09h45 and at 19h45.

The 'WHAT' part is entered by means of permanent ringing programs or permanent sequence programs (function 040201 or 040202).

WHEN: time when a pattern has to become active. In other words, when you have to switch the program blocks 'YEAR PROGRAM', 'PRE-HOLIDAY' and 'HOLIDAY' on or off.

For example: on the 14th of August, we have to turn on block 'PRE-HOLIDAY' and turn off the two other blocks. On the 15th of August (holiday), block 'HOLIDAY' has to be turned on and the other two have to be turned off. On the 16th of August, only the block 'YEAR PROGRAM' may be turned on.

The 'WHEN' part is entered by means of permanent program blocks (function 040205).

Detailed example:

STEP 01: change the names of the program blocks (function 09).

To key in our programs, you have to change the names of the 3 program blocks in 'YEAR PROGRAM', 'PRE-HOLIDAY' and 'HOLIDAY'. To do this, move to function 09.

Note: as mentioned before, a program block is a condition (a switch) for a program. When the program is 'OFF', the program will not be executed.

Example: choose the first three program blocks that are not used and change their names.

Function 0905 : change the name into 'YEAR PROGRAM'

Function 0906 : change the name into 'PRE-HOLIDAY'

Function 0907 : change the name into 'HOLIDAY'

STEP 02 : Enter the ringing programs (the 'WHAT' part)

	1	2	3
	YEAR PROGRAM	PRE-HOLIDAY	HOLIDAY
RINGING	Ringling with bell 3 for 5 minutes	Ringling with bell 3 and 2 for 5 minutes	Ringling with bell 3, 2 and 1 for 5 minutes
TIME	1) 17.45 week day 2) 18.45 Saturday 3) 07.45 Sunday 4) 09.45 Sunday 5) 19.45 Sunday	1) 18.45 daily	1) 07.45 daily 2) 09.45 daily 3) 19.45 daily

During the 'YEAR PROGRAM' pattern five weekly masses have to be entered. Below you find the permanent ringing programs (function 040202) where those 5 masses are programmed in.

RINGING PROG. 21	
3.....	UP
05 MINUTES	
YEAR PROGRAM	DOWN
17 45 00	
MON-FR NORMAL	OK
01 JANUARY ..	
31 DECEMBER ..	BACK

RINGING PROGR. 22	
3.....	UP
05 MINUTES	
YEAR PROGRAM	DOWN
18 45 00	
SATURDAY NORMAL	OK
01 JANUARY	
..	BACK
31 DECEMBER ..	

RINGING PROGR. 23	
3.....	UP
05 MINUTES	
YEAR PROGRAM	DOWN
07 45 00	
SUNDAY NORMAL	OK
01 JANUARY ..	
31 DECEMBER ..	BACK

RINGING PROGR. 24	
2.....	UP
05 MINUTES	
YEAR PROGRAM	DOWN
09 45 00	
SUNDAY NORMAL	OK
01 JANUARY ..	
31 DECEMBER ..	BACK

RINGING PROGR. 25	
3.....	UP
05 MINUTES	
YEAR PROGRAM	DOWN
19 45 00	
SUNDAY NORMAL	OK
01 JANUARY ..	
31 DECEMBER ..	BACK

Note that one mass has to be entered for the 'PRE-HOLIDAY' pattern.

Since the day before a holiday can be any day of the week (e.g. Easter is on a Sunday, Ascension Day is on a Thursday...), you have to enter 'DAILY'. The use of a program block guarantees that the program is only executed when the program block is turned 'ON'.

Below you find the permanent ringing program (function 040202), where you have programmed this mass:

RINGING PROGR.	26	
	3 2	UP
05	MINUTES	
	PRE-HOLIDAY	DOWN
	18 45 00	
	DAILY NORMAL	OK
01	JANUARY ..	
31	DECEMBER ..	BACK

Notice that 3 masses have to be entered for 'HOLIDAY'.

Since a holiday can be any day of the week (e.g. Easter is on a Sunday, Ascension Day is on a Thursday...), you have to enter 'DAILY'. The use of a program block guarantees that the program is only executed when the program block is turned 'ON'.

Underneath you find the permanent ringing program (function 040202), where you have programmed this mass:

RINGING PROGR.	27	
	3 2 1	UP
05	MINUTES	
	HOLIDAY	DOWN
	07 45 00	
	DAILY NORMAL	OK
01	JANUARY ..	
31	DECEMBER ..	BACK

RINGING PROGR.	28	
	3 2 1	UP
05	MINUTES	
	HOLIDAY	DOWN
	09 45 00	
	DAILY NORMAL	OK
01	JANUARY ..	
31	DECEMBER ..	BACK

RINGING PROGR.	29	
3 2 1		UP
05 MINUTES		
HOLIDAY		DOWN
19 45 00		
DAILY	NORMAL	OK
01 JANUARY	..	
31 DECEMBER	..	BACK

STEP 03 : enter program block programs (the ‘WHEN’ part)

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS (I)
ANNUAL MASS	PRE-HOLIDAY	HOLIDAY	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS (II)

- (I) a normal week without HOLIDAY
- (II) a week with a HOLIDAY on a week day (here e.g. on Wednesday)

The example above (II) shows that Wednesday is a holiday. You have to make sure that ‘Tuesday’ is a pre-holiday, ‘Wednesday’ a HOLIDAY and ‘Thursday’ a normal annual mass. To make it easier for the customer, we make sure that he only needs to program ‘pre-holiday’. The switch from *pre-holiday to HOLIDAY* (in our example, it is the night from Tuesday to Wednesday) and the switch from *HOLIDAY to YEAR PROGRAM* (in our example, it is the night from Wednesday to Thursday) will occur automatically.

Below you find a schematic representation of the bells that have to be turned ‘ON’ on the different days (we take the example above, where the holiday is on a Wednesday):

PROGR. BLOCK	MONDAY	TUESDAY	WEDNESDAY	THURSDAY
	Normal day	Day before holiday	holiday	Normal day
YEAR PROGRAM	ON	OUT	OUT	ON
FESTIVE SAT.	OUT	ON (PR)	OUT	OUT
HOLIDAY	OUT	OUT	ON	OUT

By means of the permanent program blocks mentioned below, the customer only has to program the pre-holiday.

The customer (or the installer) will turn the program block ‘pre-holiday’ ON for 1 day a few minutes after midnight (e.g. 00h02).

You have to follow that procedure for all the holidays that occur automatically. (1 program for each holiday) We give two examples, namely the 15th of August and the 30th of March (Ascension Day and Easter Saturday). You have to follow the same procedure to key in the other programs.

BLOCK PROGR.	01	
PRE-HOLIDAY		UP
01 DAYS		
BLOC INDEPENDENT		DOWN
00 02 00		
DAILY NORMAL		OK
15 AUGUST ..		
15 AUGUST ..		BACK

BLOCK PROGR.	02	
PRE-HOLIDAY		UP
01 DAYS		
BLOC INDEPENDENT		DOWN
00 02 00		
DAILY	EASTER	OK
30 MARCH	01	
30 MARCH	01	BACK

The next program block programs (nrs 3 until 4 included) have to be entered only once.

Right after pre-holiday is turned on (see upper examples), the 'YEAR PROGRAM' have to be turned off. A day before a holiday these YEAR PROGRAM may not be executed anymore. We can do this by blocking 'YEAR PROGRAM' for two days. After two days, the block "YEAR PROGRAM" will be automatically turned 'ON' again. For that we have to key in this example:

BLOCK PROGR.	03	
YEAR PROGRAM		UP
02 DAYS OFF		
PRE-HOLIDAY		DOWN
00 03 00		
DAILY	NORMAL	OK
01 JANUARY	..	
31 DECEMBER	..	BACK

ATTENTION:

This program has to be entered in (e.g. 00h03) **after** the pre-holiday programs (e.g. 00h02).

Now you can program the switch from 'pre-holiday' to 'HOLIDAY'. This has to be executed one day after. To do that, you have to enter in this program:

BLOCK PROGR.	04	
HOLIDAY		UP
01 DAYS		
PRE-HOLIDAY		DOWN
00 01 00		
DAILY	NORMAL	OK
01 JANUARY	..	
31 DECEMBER	..	BACK

ATTENTION:

This program has to be entered (e.g. 00h01) **before** the programming of the pre-holiday programs (e.g. 00h02).

SUMMARY:

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS (I)
ANNUAL MASS	PRE- HOLIDAY	HOLIDAY	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS	ANNUAL MASS (II)

- **MONDAY:** on a normal day, the program block 'YEAR PROGRAM' is turned ON. The other two program blocks 'PRE-HOLIDAY' and 'HOLIDAY' are turned OFF.
- **TUESDAY:** by means of a program block program (e.g. nr 1 or 2), the 'PRE-HOLIDAY' is turned on for one day at 00h02. After a while, at 00h03 exactly, the program block 'YEAR PROGRAM' will be blocked for two days. This occurs by means of program block 3.
- **WEDNESDAY:** by means of program block program 4, the 'HOLIDAY' is turned on for one day at 00h01. At 00h02, the 'Pre-holiday' is turned' off. This block was programmed for only one day.
- **THURSDAY:** at 00h01, the 'HOLIDAY' is turned off again (it was only programmed for one day). At 00h03, the program block 'YEAR PROGRAM' is turned ON again. This block was programmed for only two days.

STEP 04 : switch on 'YEAR PROGRAM'

Put the program block 'YEAR PROGRAM' in the user menu

Type = PROGRAM BLOCK,

Action = YEAR PROGRAM
01 PERMANENT

and switch the block ON.

E. SPECIAL APPLICATIONS

1. APOLLO II - serial

The serial system can be used when insufficient electric cables are available between the master clock and the control box in the tower.

The system is made up of 2 parts: the processor board (APOLLO II) in the sacristy and the power board in the tower. We advise to connect the two boards by means of a **4-wire telephonic cable**, or by 4 potential free cables.

The processor board is supplied by the power board to which you connect the 220V AC. You will always find 10 exits upon the power board (6 relay exits and 4 spark free hammer outputs). The incorporated transistor outputs make the use of hammer boxes redundant. You can connect 2 power boards in series, so that you obtain a maximum of 20 outputs.

1.1. Connections

1. Power board

KL2 : N – L: connections for 230 VAC

KL4 : NO1 NO2 NO3 NO4 C1-4 NO5 NO6 C5-6 L' L'

NO1 - NO4: to be used for tolling, continuous outputs, louver-hole, Roman knell, ...

C1-4: the desired tension for outputs NO1-NO4 can be connected to the connection C1-4. If 230VAC is desired, you can put a connection between L' and C1-4.

NO5 – NO6: when you have a **tower clock**, it is better to connect it to the **outputs N05 and N06** (important as regards the test functions!). When no tower clock has to be connected, you can use the outputs for another application.

C5-C6: the desired tension for outputs NO5-NO6 has to be connected to the connection C5-6. If 230V AC is desired, you can put a connection between L' and C5-6.

L'-L': the fused phase L leaves from here

KL3: M+ M1 M+ M2 M+ M3 M+ M4

This connector is used for direct connection of hammer coils.
The output tension is always rectified 230V.

KL5: TXD+ TXD- + -

The serial communication between the processor- and the power board is realized by means of a 4-wire telephonic cable. If desired, you can connect a second power board.

TXD+ en TXD-: communication

+ -: power supply (12V) for the processor board (APOLLO II)

ATTENTION: a wrong connection can lead to permanent damage.

If the distance between both boards is significant (e.g. 100 meters), it is better to use double wiring for the communication wires (TXD- and TXD+) (2 wires for connection). That way the resistance of the conductors (bigger section) will reduce.

2. Processor print APOLLO II

KL2: TXD + TXD- + -

The serial communication between the processor- and the power board is realized by means of a 4-wire telephonic cable.

ATTENTION: a wrong connection can lead to permanent damage.

KL3: IN1 IN2 + -

There are always 2 inputs on the serial board to start executions, programmed by the installer.

To start the input, you have to connect IN1 (or IN2) to the “-“ connection. Attention: the input is positively flank sensitive. That means that you always have to make sure that the connection between IN1 and “-“ is taken away before restarting the input.

KL1: RADIO

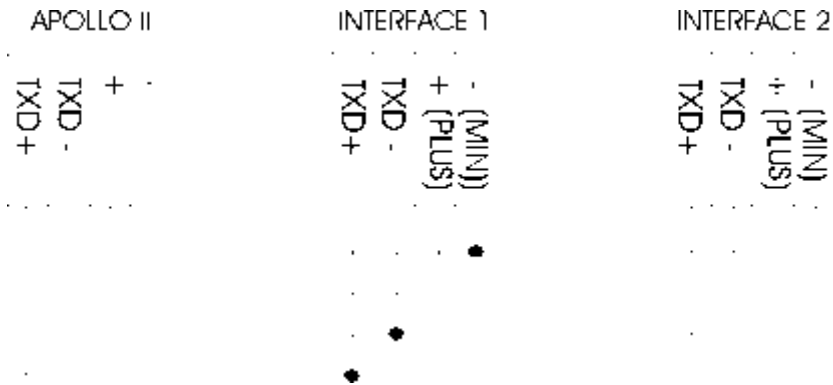
Provided for the connection with the radio synchronisation, e.g. DCF antenna.

1.2. The connection of 2 serial power boards

When you connect two serial power boards to the serial Apollo II device, the communication wires (TXD+ en TXD-) first have to go from the serial Apollo II device to the first serial power board, and then from the first to the second power board.

The connection from the '+12V' may only be made between the first serial and the Apollo II-serial device.

In that case, the terminator (Jumper J1) has to be installed onto the second power board. The jumper has to be removed from the first power board.



1.3. Options – settings

APOLLO II – Function 13:

In this function, you have to define the choice of the outputs. The 6 relay outputs are considered as relay 1-6, while the 4 hammer outputs are considered as relay 7-10. When two power boards in series are connected, we talk about the outputs 11 -16 (for the relay outputs) and outputs 17-20 (for the hammer outputs).

If you want to connect a tower clock, we advise you to do that on relay 5 and 6.

Example:

Relay 1: tolling 1
Relay 2: tolling 2
Relay 3: tolling 3
Relay 4: not used
Relay 5: clock 1A
Relay 6: clock 1B
Relay 7: hammer 1:C1
Relay 8: hammer 2:CIS1
Relay 9: hammer 3:D1
Relay 10: not used

Settings power board:

The power board has 2 jumpers (J1 and J2) and 1 switch with 3 dipswitches (S1).

Dipswitches S1

The dipswitches are used as test pins. When you place the dipswitch in the highest position (ON), a test takes place:

- 1: testing of the 4 first relays (normally used for tolling, continuous outputs, ...)
- 2: testing of the tower clock (connected on relays 5 and 6 !!)
- 3: testing of the 4 hammer outputs

In all 4 cases, the relays are connected and disconnected one by one. The test is finished when the dipswitch is placed back in the lower position. Verify at that moment if the tower clock is in the right position.

Jumper J2

Here you can choose the order number of the power board. If the jumper is not present (no connection), the power board is considered as number 1 (outputs have to be commanded by means of relays 1-10). When the jumper is present (connection is made), the power board is considered as number 2 (outputs have to be commanded by means of relays 11 -20).

Jumper J1 = TERMINATOR

This jumper is used to close the communication line. When you have only 1 power board, this jumper has to be present! In case you have 2 power boards in series, the jumper has to be installed on to the second board. You cannot install a jumper on to the first board.

2. APOLLO II-SBSI with MIDI IN/OUT interface

2.1. Preliminary remarks – connections (see also the plan at the end)

The APOLLO II-SBSI device is used for applications with more than 16 outputs or MIDI (to record melodies yourself). The APOLLO II device is used in combination with a (max. 4 in parallel) SBSI power board(s) that is situated in the tower. To this power board, you can connect hammers, ringing motors, illumination, tower clock...

The connection between the APOLLO II-SBSI and the SBSI power board(s) is made by means of 4 wires (0,25 mm²).

The programming of the APOLLO II-SBSI is done in the same way as the programming of every other APOLLO II/Tempora device. Since the APOLLO II-SBSI interface is provided with 2 mechanic relay outputs, and since you can command maximum 4 power boards with each 24 outputs, you have to program in total 98 relays in function 13. The two outputs on the APOLLO II-SBSI interface are commanded by means of function 1398 (relay 97 and function 1399 (relay 98).

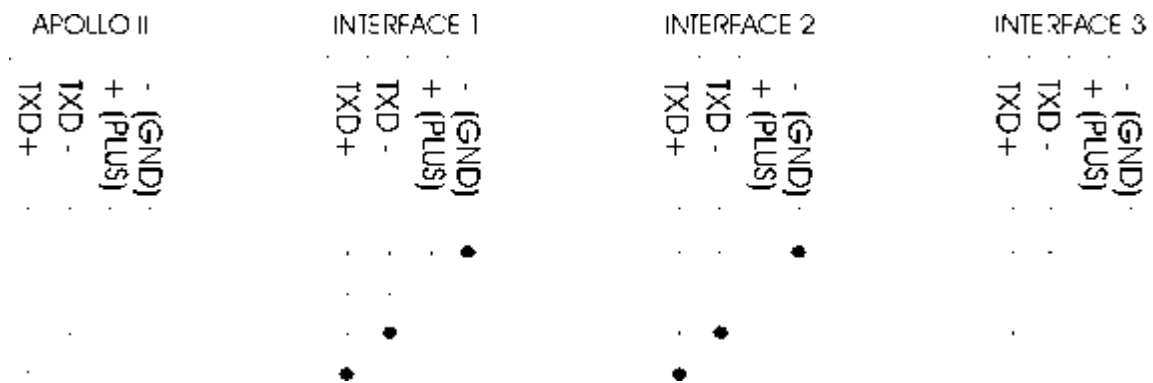
In order to distinguish the different SBSI power boards (which board has outputs 1-24? Which board has outputs 25-48?...), you first have to establish 2 dipswitches (SW1) in the SBSI-log board (top left side on the SBSI power board). The position of the dipswitch 3-8 is not important for the functioning of the whole.

DIPSWITCHES SW1: 1 2 3 4 5 6 7 8 9 0

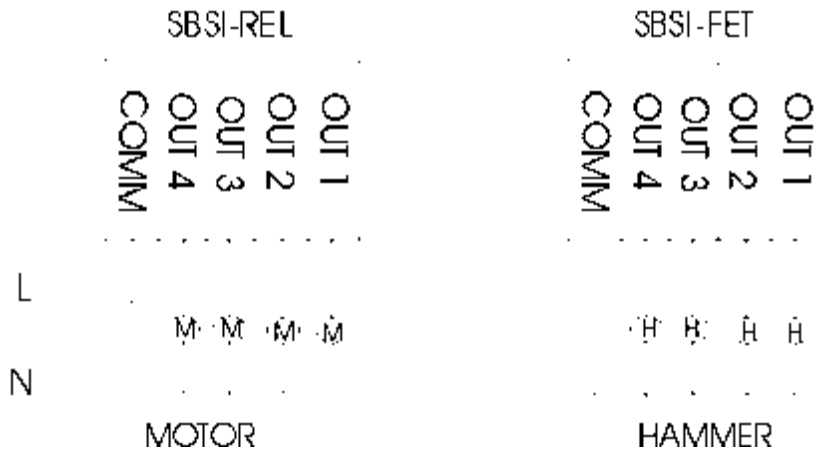
SBSI-power board 1 (outputs 1-24) : 0 0 0 0 0 0 0 0 0 0
SBSI-power board 2 (outputs 25-48) : 0 1 0 0 0 0 0 0 0 0
SBSI-power board 3 (outputs 49-72) : 1 0 0 0 0 0 0 0 0 0
SBSI-power board 4 (outputs 73-96) : 1 1 0 0 0 0 0 0 0 0

Remarks:

- When you want to use an existing SBSI power board (that was connected with an APOLLO I or MCC) with an Apollo II-SBSI device, you have to make sure that there is a **SBSI-LOG board with processor for the Apollo II.**
- When you connect more than one SBSI power boards to the Apollo II-SBSI device, the communication wires (TXD+ en TXD-) first have to go to the SBSI, then from the first to the second SBSI, then from the second to the third SBSI, etc. They may never be connected in star form. **The connection from the '+12V' may only be established between the first SBSI and the Apollo II-SBSI device.** On the last SBSI, two jumpers have to be placed (on the SBSI-LOG board). Those jumpers may not be installed on the other SBSI power boards.



- You can install up to 6 output boards on every SBSI power board. Every output board has 4 outputs. That way, up to 24 outputs can be provided for every power board. There are 2 types of output boards:
 - The 'SBSI-REL' board has 4 tensionless mechanic relays and is used for ringing motors, tower clock, louver-hole...
 - The 'SBSI-FET' board has 4 'solid-state' hammer outputs, to which you can immediately connect the hammer coil of the electromagnetic hammers.
- **SBSI-REL** : you have to put tension on the 'COMM'. You have to connect the devices that are fed by means of this tension, to the OUT connections. During the closure of the relays, the tension that is connected to the COMM, will be transmitted to the corresponding OUT.
- **SBSI-FET** : when a SBSI-FET print is entered, you create a tension of rectified 220V or rectified 110V (depending on the tension that is connected to the clip KL2) between the COMM and the OUT connection. In other words, the hammer has to be connected between the COMM and the OUT connection.



2.2. Programming

The APOLLO II-SBSI device is programmed like any other APOLLO II or Tempora device. Because the APOLLO II-SBSI, used as a carillon computer, requires some specific program techniques, we will discuss them below.

Function 13 (relay) : when you want to use an APOLLO II device to play melodies, it is important that the right tones are entered when defining the relay.

Example:

You have the following bell series:

C2 (do) – D2 (re) – DIS2 (re#) - E2 (mi) - F2 (fa) – FIS2 (fa#) – G2 (sol) – GIS2 (sol#)–
A2 (la) – AIS2 (la#) – B2 (si) – C3 (do)

Function 1302 : relay 1 = hammer 13:C2
 Function 1303 : relay 2 = hammer 15:D2
 Function 1304 : relay 3 = hammer 16:DIS2
 Function 1305 : relay 4 = hammer 17:E2
 Function 1306 : relay 5 = hammer 18:F2
 Function 1307 : relay 6 = hammer 19:FIS2
 Function 1308 : relay 7 = hammer 20:G2
 Function 1309 : relay 8 = hammer 21:GIS2
 Function 1310 : relay 9 = hammer 22:A2
 Function 1311 : relay 10 = hammer 23:AIS2
 Function 1312 : relay 11 = hammer 24:B2
 Function 1313 : relay 12 = hammer 25:C3

When you define the relay this way, you will hear the right bell (tone) each time you press a key on an electronic keyboard. The correct definition of the relay is also important when you send MIDI-files (melodies from the computer) or when you transfer melodies by means of a PIM module.

2.3. Sequences for APOLLO II-SBSI (carillon computer)

For the APOLLO II device, used as a carillon computer, some very specific sequences can be made. When we put these sequences in the user menu, the carillon computer becomes user-friendlier.

We advise to store all the sequences that don't contain any melodies as from sequence 100. That way, the first sequences (0 until 99 included) will be free for the customer to store melodies in.

2.3.1. "Record melody"

Purpose: This sequence can be used to give the customer the possibility to record melodies. He can do this by means of the numeric keys on the APOLLO II device or by means of an external keyboard (MIDI, only by APOLLO II-SBSI). Every time you start recording a melody, the previous recording will be erased.

Sequence (installer):

- Start recording a sequence (function 0601).
- Move to the function 'sequences' (function 060106) and press 'OK'.
- Select 'record sequence' (function 06010601) and press 'OK' – notice that we have started up recording a sequence twice.
- Enter function 'executions' by pressing 'OK'.
- Move to 'manual tolling' and press 'OK'.
- Press 'STOP' to let the customer practice the 'manual tolling' himself (= recording a melody).
- Press 'BACK' until function **0601** (record sequence) to close the recording of the sequence – notice that going back to *06010601* doesn't close the recording entirely.
- Test (function 0602) and store the sequence in function 0603 (e.g. sequence 100) and call it "record melody".
- Put the sequence in the user menu (function 02).

2.3.2. "Test melody"

Purpose: By means of this sequence, you can test the melody that you have recorded with 'record melody'. After this test, the customer can decide whether or not he wants to store the melody by means of 'store melody'.

Sequence (installer):

- Start recording a sequence (function 0601).
- Move to the function 'sequences' (function 060106) and press 'OK'.
- Move to 'test sequence' (function 06010602) and press 'OK'.
- Change the parameters as follows and press 'OK':

TEST SEQUENCE	UP
RECORD	DOWN
01 TIMES 0	OK
NAME SEQUENCE	BACK

- Press 'BACK' until function **0601** (record sequence) to finish recording the sequence.
- Test (function 0602) and store the sequence in function 0603 (e.g. sequence 101) and call it "test melody".
- Put the sequence in the user menu (function 02).

2.3.3. "Save melody"

Purpose: It is important that during recording a melody, the previous recording will always be erased. If you want to save a recording (save), you can do this by means of this sequence. The recording will be saved in the sequence-memory.

Record sequence (installer):

- Start recording a sequence (function 0601).
- Move to the function 'sequences' (function 060106) and press 'OK'.
- Move to 'save sequence' (function 06010603) and press 'OK'. You will see the following data on the screen:

SAVE SEQUENCE	UP
SEQUENCE 001	DOWN
	OK
NAME SEQUENCE	BACK

- Press 'STOP' to give the customer the opportunity to choose in which sequence he wants to save the recording. That way, he can overwrite existing melodies.
- Afterwards press the 'STOP'-key 16 times. That way, the customer will establish the title of the melody (sequence) himself.
- Press 'BACK' until function **0601** (record sequence) to finish recording the sequence.
- Test (function 0602) and save the sequence in function 0603 (e.g. sequence 102) and call it "save melody".
- Put the sequence in the user menu (function 02).

2.3.4. "Play melody"

Purpose: by means of this sequence, the customer will be able to start a song himself. The sequence will ask which melody has to be played.

Record sequence (installer):

- Start recording a sequence (function 0601).
- Move to the function 'sequences' (function 060106) and press 'OK'.
- Move to 'play sequence' (function 06010102) and press 'OK'. The following data appears on the screen:

PLAY SEQUENCE	UP
SEQUENCE 001	DOWN
01 TIMES	OK
NAME SEQUENCE	BACK

- Press 'STOP' to give the customer the opportunity to choose which sequence he wants to play. The 01 TIMES can be confirmed by pressing 'OK' twice.
- Press 'BACK' until function **0601** (record sequence) to finish recording the sequence.
- Test (function 0602) and save the sequence in function 0603 (e.g. sequence 103) and call it "play melody".
- Put the sequence in the user menu (function 02).

The 4 sequences mentioned above (record melody, test melody, save melody and play melody) are meant to offer the customer the possibility to record and play melodies himself.

2.4. Programming melodies (carillon computer)

Now that the customer can record, save and play melodies, you have to make sure that the songs can be programmed. To give the customer the possibility to switch the carillon on and off automatically, we are going to use a program block. Therefore, move to function 09 (program blocks) and look for an unused program block.

Example:

FUNCTION 0907: PROGRAM BLOCK 07 -> AUTOM. CARILLON

There are 2 ways to program the melodies (depending of the choice of the final customer):

2.4.1 Non-cyclic programming

With a non-cyclic program, the same melody is played at a particular moment (e.g. 12 o'clock). This kind of programming is used especially when the number of melodies during the day is limited (e.g. at 9 am, at 12 o'clock and at 19 pm) or when a limited number of melodies are recorded.

Example :

- At 9u01 (after the hour stroke!!) we want to play the 'Ave Maria '
- At 12u01 (after the hour stroke!!) we want to play the 'Town song'
- At 19u01 (after the hour stroke!!) we want to play the 'Adeste Fidelis'

After having recorded these 3 melodies, you can program them directly in the sequence programs. Take 3 empty sequence programs (in this example, you take programs 21 until 23 included) and change them as follows:

SEQ.PROGR.	021	
AVE MARIA		UP
01 TIMES		
AUTOM. CARILLON		DOWN
BEGIN.. 09 01 00		
DAILY NORMAL		OK
01 JANUARY ..		
31 DECEMBER ..		BACK

SEQ.PROGR.	022	
TOWN SONG		UP
01 TIMES		
AUTOM. CARILLON		DOWN
BEGIN .. 12 01 00		
DAILY	NORMAL	OK
01 JANUARY ..		
31 DECEMBER ..		BACK

SEQ.PROGR.	023	
ADESTE FIDELIS		UP
01 TIMES		
AUTOM. CARILLON		DOWN
BEGIN .. 19 01 00		
DAILY	NORMAL	OK
01 JANUARY ..		
31 DECEMBER ..		BACK

All the possibilities that the sequence programs offer (on date, during a specific period, summer/winter...) can be used to program the melodies.

2.4.2 Cyclic programming

Change the name of a program block in 'AUTOM. CARILLON' with function 09.

With cyclic programming, the different melodies are placed in groups. Every time a melody program becomes active, the next melody of the *active* group will be played. When the last melody from the *active* group is played, the first melody will be played again by starting up the next program.

A cyclic program looks like this (example):

SEQ.PROGR.	023	
SEQ. FROM GROUP		UP
01 TIMES		
AUTOM. CARILLON		DOWN
BEGIN 08 20 01 00		
DAILY	NORMAL	OK
01 JANUARY ..		
31 DECEMBER ..		BACK

'SEQ. FROM GROUP': as sequence, we choose 'seq. from group'. That way, the program will play the next melody in the active group. We have to divide the melodies into (different) groups.

Example:

Position	Christmas melodies	Carillon group 1	Carillon group 2
1	Jingle Bells	Town Song	Green Sleeves
2	Silent Night	Giant Song	Ave Maria
3	O pine tree	Waltzing Mathilda	Love Me Tender
4	Adeste Fidelis	Der Lindenbaum	Town Song
5	Three Kings	Funiculi, Funicula	Salve Regina
6		Ave Maria	
7			<i>Kalinka</i>

There are 24 different possible groups in the APOLLO II device. In our example, we have three groups: Christmas groups, Carillon group 1 and Carillon group 2.

Every group can contain 96 melodies (positions). The table shows that the group ‘Christmas melodies’ contains 5 melodies and that the ‘Carillon group 1’ and ‘Carillon group 2’ both contain 6 melodies. Notice that a song can be used in more than one group (e.g. ‘**Ave Maria**’ and ‘**Town Song**’).

It is important that the melodies are included one after the other in the group. As there is no melody on position 6 of the ‘carillon group 2’, the melody ‘*Kalinka*’ will not be played. After having played the melody ‘Salve Regina’ (position 5), the melody ‘Green Sleeves’ (position 1) will be played. If you want to play the melody ‘*Kalinka*’ as well, you have to place it in position 6.

The next pages describe how to program the upper table. As mentioned before, you first have to record and save all melodies,.

2.4.2.1 Names groups (function 0704)

In this function, you can change the names of the groups. Move to function 0704 (names groups) and press ‘OK’. You will see the next screen:

FUNCTION 070401	UP
NAME GROUP 1	DOWN
FUNCTION 070402	OK
NAME GROUP 2	BACK

Change the names as follows:

Function 070401 – NAME GROUP 1 => Christmas melodies

Function 070402 – NAME GROUP 2 => carillon group 1

Function 070403 – NAME GROUP 3 => carillon group 2

2.4.2.2 Content groups (function 0703)

Now you can ‘fill in’ the three groups with the recorded melodies. Go there for to function 0703. The following data will appear on the screen.

FUNCTION 0703	UP
CONTENT GROUPS	DOWN
FUNCTION 0704	OK
NAME GROUPS	BACK

When you press 'OK', you proceed to the following data on the screen:

FUNCTION 070301	UP
CHRISTMAS MELODIES	DOWN
FUNCTION 070302	OK
CARILLON GROUP 1	BACK

Because you first want to 'fill in' the group 'Christmas melodies', you have to choose function 070301.

FUNCTION 07030101	UP
POS 1 GROUP 01	DOWN
FUNCTION 07030102	OK
POS 2 GROUP 01	BACK

Now we can see 96 different places (positions) in the group. The table shows that you want to place the melody 'Jingle Bells' on position 1 of the group 'Christmas melodies'. So, you have to choose function 07030101 (pos. 1 group 01).

Change 'NOT USED' into 'Jingle Bells' and press 'OK' to confirm.

POS 1 GROUP 01	UP
JINGLE BELLS	DOWN
NAME SEQUENCE	OK
	BACK

After that, you have to move to 'pos. 2 group 01'.

FUNCTION 07030101	UP
POS 1 GROUP 01	DOWN
FUNCTION 07030102	OK
POS 2 GROUP 01	BACK

For this position, choose the melody 'Silent Night'.

POS 2 GROUP 01	UP
SILENT NIGHT	DOWN
NAME SEQUENCE	OK
	BACK

Repeat this procedure until you have divided all the melodies over the 3 groups, as indicated in the table.

2.4.2.2 Active group (function 0701)

Every time ‘seq. from group’ is started, the APOLLO II device will play the next melody from the **active** group. In the function ‘active group’, you can choose from which group the melodies have to be played. In other words, in this function, we determine which group is active. When we want, for example, to play melodies from the ‘Christmas Melodies’, we have to fill out the following data in function 0701:

ACTIVE GROUP	UP
CHRISTMAS MELODIES	DOWN
01	OK
SELECTION GROUP	BACK

The active place determines the position of the next melody that will be played. In the example above, the melody ‘Jingle Bells’ (position 1 in the group ‘Christmas melodies’) will be played. The next time, the melody ‘Silent Night’ will be played. When the last melody (‘Three Kings’) is played, the first melody (‘Jingle Bells’) will be repeated.

2.4.2.3 Programming the group changeover (function 0701)

Example:

The melodies from ‘Carillon group 1’ have to be played from the 08th of January until the 30th of June.

The melodies from ‘Carillon group 2’ have to be played from the 1st of July until the 17th of December.

The melodies from ‘Christmas melodies’ have to be played from the 18th of December until the 7th of January.

To create an automatic execution, you have to insert the next three permanent group programs (function 040203):

GROUP PROGR. 01	
CHRISTMAS SONGS	UP
01 ACTIVE	
BLOCK INDEPENDENT	DOWN
00 05 00	
DAILY NORMAL	OK
18 DECEMBER ..	
07 JANUARY ..	BACK

GROUP PROGR. 02	
CARILLON GROUP 1	UP
01 ACTIVE	
BLOCK INDEPENDENT	DOWN
00 05 00	
DAILY NORMAL	OK
08 JANUARY ..	
30 JUNE ..	BACK

GROUP PROGR. 03	
CARILLON GROUP 2	UP
01 ACTIVE	
BLOCK INDEPENDENT	DOWN
00 05 00	
DAILY NORMAL	OK
01 JULY ..	
17 DECEMBER ..	BACK

Remarks:

1. Notice that the unit 'ACTIVE' is used in these programs. Because of this the group becomes active when the program is executed.
2. In the example above, the group becomes active five minutes after midnight (before the first melody of the day is played) and this on the day of the changeover (e.g. 18 December).
3. If necessary, you can make the three programs dependent on a program block, e.g. 'AUTOM. GROUPS'. Label the program block (function 09) and change the parameter 'block independent' in 'AUTOM. GROUPS' in the 3 programs. That way, there will only be automatic changeover when 'AUTOM. GROUPS' is ON.

2.5 Sequences for groups (with cyclic programming)

When the final customer wants to 'fill in' the groups himself and when he wants to execute the group changeover manually, you can record the following sequences and include them in the user menu.

2.5.1 "Fill groups"

Record sequences (installer):

- Start recording a sequence (function 0601).
- Move to the function 'groups' (function 060107) and press 'OK'.
- Select 'content groups' (function 06010703) and press 'OK'. You will see the following data on the screen:

FUNCTION 06010703	UP
CHRISTMAS MELODIES	DOWN
FUNCTION 06010703	OK
CARILLON GROUP 1	BACK

- Press 'STOP' to let the customer choose himself in which group he wants to place a melody. You will see the following data on the screen:

FUNCTION 07030103	UP
POS 1 GROUP 01	DOWN
FUNCTION 07030103	OK
POS 2 GROUP 01	BACK

- Press 'STOP' to let the customer choose on which place in the group he

wants to include a melody. The following data will appear on the screen:

POS 01	GROUP01	UP
NOT USED		DOWN
NAME SEQUENCE		OK
		BACK

- Press 'STOP' to let the customer choose the melody.
- Press 'BACK' until function **0601** (record sequence) to close the recording of the sequence.
- Test (function 0602) and save the sequence in function 0603 (e.g. sequence 102) and call it "fill groups".
- Place the sequence in the user menu (function 02).

2.5.2 "Choose group"

Record sequence (installer):

- Start recording a sequence (function 0601)
- Move to the function 'groups' (function 060106) and press 'OK'
- Select 'active groups' (function 06010701) and press 'OK'. You will see the following data on the screen:

ACTIVE GROUP	UP
CHRISTMAS MELODIES	DOWN
01	OK
SELECTION GROUP	BACK

- Press 'STOP' 2 times to give the customer the opportunity to choose which group and which place he wants to activate. If you press 'STOP' once, the final customer can only choose the active group.
- Press 'BACK' until function **0601** (record sequence) to close the recording of the sequence.
- Test (function 0602) and save the sequence in function 0603 (e.g. sequence 103) and call it "choose groups".
- Include the sequence in the user menu (function 02).

3. Digital Bell Apollo II (Tempora)

The Digital Bell Apollo II (Tempora) device is the digital version of the APOLLO II (Tempora) device. It is used to play digital bell sounds by means of amplifier and loudspeakers.

Beside the incorporated digital sounds (tolling on 4 different bells, ringing with 1,2 or 3 bells, Westminster of Whittington), there's also the possibility of plugging in 4 extra digital recordings of maximum 1 minute into the device. The EPROM's, programmed by us, are plugged into the empty EPROM places (5 until 8 included) of the speech print.

The installer can use every digital sound in sequences, programs and in the user menu.

However, it is important to know how every bell sound is commanded from the function menu. Below you find an enumeration of the functions that are necessary to start the separated digital bell sounds. Notice that no relays have to be defined for the digital sounds, because those sounds only come out via the AUDIO exit. The function 'RELAYS' (function 13) only has to be used if you want to command illumination, amplifier or real bells (hammers). For this purpose, mechanic relays (4, 6, 10 or 16 – depending on the APOLLO II or Tempora device) are provided.

Digital sound	circuit
Ringling with 1 bell	ringing 1
Ringling with 2 bells	ringing 2
Ringling with 3 bells	ringing 3
Digital bell 1 (tolling)	Hammer 1:C1
Digital bell 2 (tolling)	Hammer 2:CIS1
Digital bell 3 (tolling)	Hammer 3:D1
Digital bell 4 (tolling)	Hammer 4:DIS1
Westminster 1.4 (1st quarter)	Melody 1 in function "speech" (1104) or Hammer 5:E1
Westminster 2.4 (2 nd quarter)	Melody 2 in function "speech" (1104) or Hammer 6:F1
Westminster 3.4 (3 rd quarter)	Melody 3 in function "speech" (1104) or Hammer 7:FIS1
Westminster 4.4 (4th quarter)	Melody 4 in function "speech" (1104) or Hammer 8:G1
Digital recording 1 (EPROM5)	Melody 5 in function "speech" (1104) or ringing 5
Digital recording 2 (EPROM5)	Melody 6 in function "speech" (1104) or ringing 6
Digital recording 3 (EPROM5)	Melody 7 in function "speech" (1104) or ringing 7
Digital recording 4 (EPROM5)	Melody 8 in function "speech" (1104) or ringing 8

The Digital Bell Tempora (APOLLO II) is programmed in the same way as a normal Tempora (APOLLO II). How you have to record the different sequences (hour stroke, angelus, ...), is explained in part D "Examples" or in part C "Software" under sequences. Below you will find an enumeration of tips and exceptions, used specifically with the Digital Bell version of the APOLLO II (Tempora).

3.1. Recording a song (Westminster or one of the 4 extra melodies) in a sequence

Before you can program a Westminster or a song in the Digital Bell Apollo II device, you have to record a sequence that contains this song. Follow this procedure:

Start the recording of a sequence: as mentioned before we will do this by entering function 0601 and pressing "OK". That brings you to the screen with the main functions.

FUNCTION 060101 EXECUTIONS	UP DOWN
FUNCTION 060102 USER MENU	OK BACK

The easiest way to record a digital melody in a sequence is by using function 1101 "speech". Therefore, move to function 060111 'special output'.

FUNCTION 060111 SPECIAL OUTPUT	UP
FUNCTION 060112	DOWN
CLOCKS	OK
	BACK

Press 'OK' and search function 06011104 'Speech'

FUNCTION 06011104 SPEECH	UP
FUNCTION 06011105	DOWN
SERIAL OUTPUT	OK
	BACK

Press 'OK'. You will see the following data on the screen:

SPEECH	UP
WESTMINSTER ¼	DOWN
00 00	OK
START MELODY	BACK

You have to fill out 3 parameters:

“Start melody” : Choose which melody you want to start (and eventually program) by means of this sequence. You have the choice between the 4 Westminster (1/4 ; 2/4 ; 3/4 and 4/4) and 4 extra melodies (Melody 005 until Melody 008 included).

“Duration minutes” : If you want to program with 'end' or if you want to be sure that the melody will not be interrupted by another program, you have to establish the duration of the melody that has to be played. This duration consists of a number of minutes and a number of seconds. In the first number, you have to enter the number of minutes. As a melody never passes one minute, you have to enter 0 or 1.

“Duration seconds” : By means of the second number, you enter how many seconds the melody lasts.

Example: You want to start the melody that is programmed in the 6th EPROM (= 2nd extra melody). This melody takes up 55 seconds. In that case, you key in:

SPEECH	UP
MELODY 006	DOWN
00 55	OK
START MELODY	BACK

You have entered all data required in this sequence. Close the recording by means of the "BACK" key until you are back in function 0601.

FUNCTION 0601 RECORD SEQUENCE	UP
FUNCTION 0602	DOWN
TEST SEQUENCE	OK
	BACK

Now you can test (function 0602) and save (function 0603) the sequence. You can put or program this sequence in the user menu.

Example program (function 040202):

SEQ.PROGR.	10	
SILENT NIGHT		UP
01 TIMES		
AUTOM. MELODY		DOWN
BEGIN .. 20 05 00		
DAILY NORMAL		OK
01 DECEMBER ..		
31 DECEMBER ..		BACK

3.2. Recording sequences with “tolling” (digital sounds or hammers)

During the recording of a sequence with tolling, such as e.g. ‘stroke bell 1’ (see part D for more explanation), you have to press the numerical keys to enter the strokes. That way you toll with the hammer, defined in the corresponding ringing function.

By pressing the numerical key ‘1’ during the recording of a sequence, you give a stroke that is defined in ‘ringing 1’ (function 0501). Normally, for ‘ringing 1’, this is hammer 1:C1, corresponding with the digital sound of the largest bell.

	RINGING 1		UP
30	30 1:C1 060		DOWN
	RINGING		OK
	TIME RING IN S.		BACK

By pressing numerical key ‘2’ during the recording of a sequence, you give a stroke that is defined in ‘ringing 2’ (function 0502). Normally, this is for ‘ringing 2’ hammer 2:CIS1, corresponding with the digital sound of the second largest bell.

	RINGING 2		UP
30	30 2:CIS1 060		DOWN
	RINGING		OK
	TIME RING IN S.		BACK

The standard value says also that:

- RINGING 3 -> 3:D1 (digital sound of the second smallest bell)
- RINGING 4 -> 4:DIS4 (digital sound of the smallest bell)
- RINGING 5 -> 5 : E1 (Westminster 1/4)
- RINGING 6 -> 6:F1 (Westminster 2/4)
- RINGING 7 -> 7:FIS1 (Westminster 3/4)
- RINGING 8 -> 8:G1 (Westminster 4/4)
- RINGING 9 -> 9:GIS1 (can be used as first real hammer)
- RINGING 0 -> 10:A1 (can be used as second real hammer)

Example:

You want to combine the sound of 4 digital bells with 3 real hammers. Notice that you need 7 numerical keys (APOLLO II) to do this. Afterwards, you can transmit the program to the Digital Bell Tempora device.

* Digital bell 1 :

- Record a new sequence (function 0601).
- Press numerical key '1' during the recording of the sequence and save the sequence – call it 'digital stroke 1'.
- Make sure that hammer 1:C1 is in function 0501 (ringing 1).

* Digital bell 2 :

- Record a new sequence (function 0601).
- Press numerical key '2' during the recording of the sequence and save the sequence – call it 'digital stroke 2'.
- Make sure that hammer 2:CIS1 is in function 0502 (ringing 2).

* Digital bell 3 :

- Record a new sequence (function 0601).
- Press numerical key '3' during the recording of the sequence and save the sequence – call it 'digital stroke 3'.
- Make sure that hammer 3:D1 is in function 0503 (ringing 3).

* Digital bell 4 :

- Record a new sequence (function 0601).
- Press numerical key '4' during the recording of the sequence and save the sequence – call it 'digital stroke 4'.
- Make sure that hammer 4:DIS1 is in function 0504 (ringing 4).

* first (largest) bell with hammer :

- Record a new sequence (function 0601).
- Press numerical key '5' during the recording of the sequence and save the sequence – call it 'stroke bell 1'.
- Make sure that hammer 9:GIS1 is in function 0505 (ringing 5).
- Program hammer 9:GIS1 in function 1302 (relay 01).

* second bell with hammer:

- Record a new sequence (function 0601).
- Press numerical key '6' during the recording of the sequence and save the sequence – call it 'stroke bell 2'.
- Make sure that hammer 10:A1 is in function 0506 (ringing 6).
- Program hammer 10:A1 in function 1303 (relay 02).

* third (smallest) bell with hammer :

- Record a new sequence (function 0601).
- Press numerical key '7' during the recording of the sequence and save the sequence – call it 'stroke bell 3'.
- Make sure that hammer 11:AIS1 is in function 0507 (ringing 7).
- Program hammer 11:AIS1 in function 1302 (relay 03).

Now you can use these sequences in programs, in the user menu and in the inputs like any

other sequence.

3.3. The use of real ringing bells with the Digital Bell device

If you want to command real ringing bells, you can do this by means of 'ringing 9' and 'ringing 0'. The other ringing circuits (ringing 1 until ringing 8 included) are used to start the digital sounds.

- Establish function 13 (relays):

EXAMPLE:

Function 1302: 'relay 1' = ringing 9

Function 1303: 'relay 2' = ringing 0

- Use 'ringing 9' and 'ringing 0' in your ringing programs and sequences for the command of these two ringing bells (relays).

Make sure that the right hammers are being blocked. You do this by establishing 'hammer 09:GIS1' in function 0509 and 'hammer 10:A1' in function 0510. You can use these hammers by pressing the numerical keys '9' (bell 1) and '0' (bell 2) during the recording of a sequence - see also point 3.2, previous pages.

F. OPTIONS

1. Introduction

For the APOLLO II device and the Tempora there are a few options available. They allow a faster and user-friendlier use of the contents of the APOLLO II device.

All options can be connected to the connector (SUBD9) at the left-hand side of the device.

The different options for the APOLLO II and Tempora are: Programmable Pluggable Module (PIM), MIDI interface, PC interface, Modem, Radio Time Receiver.

Both the APOLLO II and the Tempora device have been provided with optional inputs. The Tempora device can also be equipped with a speech (audio) output.

2. PIM-Module

The PIM (Programmable Insertable Module) contains an internal memory that can be used for storing the contents of the APOLLO II device's memory. The PIM module is mainly used for:

- 1) creating a back-up copy of the Apollo II device's memory in the PIM module
- 2) loading a software update from the PIM module into the Apollo II device
- 3) loading new sequences (e.g. melodies) from the PIM module into the Apollo II device
- 4) loading the complete contents of the PIM module into the Apollo II device (software update + sequences + parameters)

The PIM module consists of a print PCB in a small housing. It has two connectors. One connector is used to load data from the PIM into the Apollo II device, the other one to load the contents of the Apollo II device's memory into the PIM. The two arrows (red and green) on the PIM indicate the direction of data transfer. When the red arrow is situated at the side of the master clock, the device's software and possibly its memory too will be overwritten.

LOADING CONTENTS OF APOLLO II DEVICE INTO PIM

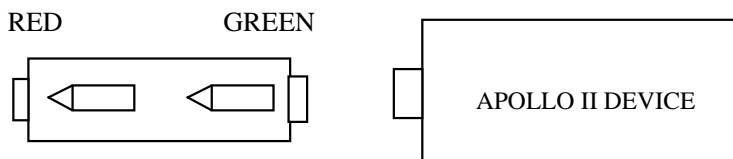
- 1) When using the PIM to make a back-up copy, you can determine which part of the memory of the APOLLO II device will be loaded into the PIM. When the contents of the PIM are loaded into the APOLLO II later on, only this part will be replaced in the Apollo II device. Therefore, it is important that the person who loads the PIM determines which part is to be replaced in the Apollo II device. This is done in the function "LOAD PIM" (function 1406). The different options for loading the PIM are:

- "ALL": SEQUENCES, PARAMETERS and SOFTWARE are stored in the PIM
- "SOFTWARE ONLY" : Only the software is stored in the PIM
- " SOFTWARE + SEQ." : The software and the sequences are stored in the PIM. Only the sequences, which are indicated by the last two parameters of this function, are stored in the PIM (e.g. sequences 21 up to 40). The names of the sequences as well as the speed of the sequences are stored in the PIM.
- "SOFT + SEQ. + GROUP" : The software, content of the groups and the sequences

are stored in the PIM. Only the sequences, which are indicated by the last two parameters of this function, are stored in the PIM (e.g. sequences 21 up to 40). The names of the sequences as well as the speed of the sequences are stored.

LOAD PIM	UP
ALL	DOWN
TRANSACTION	OK
	BACK

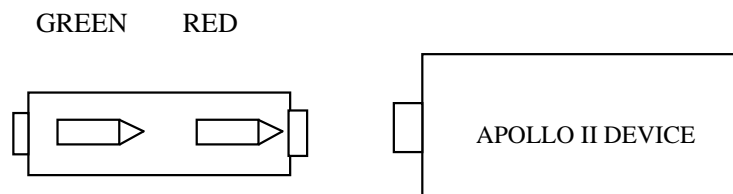
- 2) Switch the device off.
- 3) Connect the PIM to the APOLLO II device by means of the connector at the left-hand side of the APOLLO II device. To load the contents from the APOLLO II device into the PIM, make sure that the arrows on the PIM are pointing away from the APOLLO II device, indicating that the contents from this device are being copied. That means that the green end of the PIM is connected to the APOLLO II device.



- 4) Switch on the APOLLO II device. The contents of the APOLLO II device will be copied into the PIM module. The following messages will appear: CHECK PIM, ERASE PIM, PROGRAM PIM, and VERIFY PIM. When the PIM is loaded, the APOLLO II device will return to its main menu. When you are using a bad PIM, the message “PLEASE USE ANOTHER PIM” will appear on the screen of the APOLLO II device.
- 5) Now the PIM can be removed from the APOLLO II device. Switch off the device before doing so. Remove the PIM module from the APOLLO II device and switch on the device again.

LOAD CONTENTS OF PIM INTO APOLLO II DEVICE

- 1) As mentioned before, the person who loads the PIM can determine which part of the contents of the APOLLO II device will be loaded.
- 2) Switch off the device.
- 3) Connect the PIM to the APOLLO II device by means of the connector at the left-hand side of the APOLLO II device. To load the contents from the PIM into the APOLLO II device, make sure that the arrows on the PIM are pointing to the APOLLO II device, indicating that the contents from the PIM module are being copied. That means that the red end of the PIM is connected to the APOLLO II device.



4) Switch on the APOLLO II device. The contents of the PIM will be loaded into the Apollo II device. The following messages will appear: CONTROL PIM, ERASE EPROM (the software is being erased), PROGRAM EPROM (new software is being installed), CONTROL PIM, ERASE SEQUENCES, PROGR. SEQUENCES, LOAD FUNCTIONS.

5) Now the PIM can be removed from the APOLLO II device. Switch off the device before doing so. Remove the PIM module from the APOLLO II device and switch the device on again.

3. MIDI-Interface

This option allows you to connect an external keyboard to the APOLLO II device. It is also possible to connect a sampler in order to hear the melodies that have been programmed in the APOLLO II device through speakers. To use the MIDI interface, follow these instructions:

1. Switch off the device.
2. Connect the MIDI interface to the SUBD9 connector on the APOLLO II device.
3. Now you will see two wires, coming from the Apollo II device through the interface.
 - a) Connect wire "MIDI IN" to the "MIDI OUT" on your keyboard.
 - b) Connect wire "MIDI OUT" to the "MIDI IN" on your keyboard.
4. Switch on the device again. The APOLLO II device will automatically detect the MIDI interface. Make sure that function MIDI (function 1103) is switched "ON".

MIDI	+1
ON 0	-1
SITUATION	OK
	BACK

This function ensures that the MIDI signals from the APOLLO II device are sent to the keyboard. The MIDI signals from the keyboard always enter the APOLLO II device (no matter what MIDI parameter has been entered in function 1103). When you are in the main menu, you can go to the user menu by pressing a key on the keyboard. The note you enter will not be played. It is only necessary to proceed to the user menu. Please keep in mind that you have to enter the correct user code, when a user code has been entered in function 1402.

You can also transpose (shift) the keyboard in this function.

4. Radio Time Receiver

The APOLLO II device can be synchronized by means of a radio time receiver. There are several types of radio time receivers available: DCF, MSF, GPS and master clock synchronization. The following procedure is applied to synchronize the APOLLO II device by means of a DCF or MSF antenna.

- 1) Switch off the APOLLO II device.
- 2) Connect the antenna to the APOLLO II device as described in the enclosed manual.
- 3) Switch on again the APOLLO II device.
- 4) Now the APOLLO II device will try to synchronize.
- 5) There is a red LED on the antenna. This indicates the signal the antenna receives. Make

sure that the antenna is well placed so that this LED blinks regularly. The APOLLO II device has a green LED. This LED allows you to check whether the antenna has good reception and/or whether the APOLLO II device has been synchronized.

- a. The green LED (on the APOLLO II) and the red LED (on the antenna) blink simultaneously: the APOLLO II device is checking the received signal.
- b. The green LED and the red LED blink alternately (green on → red off + vice versa). The antenna and the APOLLO II device have good reception. The APOLLO II device will keep on trying this until 5 a.m. However, when synchronization is completed faster, the device will stop checking.
- c. When the APOLLO II device is synchronized, the green LED will keep on burning.
- d. When the APOLLO II device is not synchronized, the green LED is turned off (so after 5 a.m. if synchronization fails).
- e. **DCF and MSF synchronization occurs daily between 3 and 5 a.m.**
When synchronization is completed before 5 a.m. the APOLLO II device will stop searching and the green LED will stay on. When the APOLLO II device has failed to complete synchronization at 5 a.m. (so after trying for 2 hours) the green LED will go out. When you check the device during the day, you will immediately see if it has been synchronized. **If the green LED is on, this means that the device was synchronized during the night.** If the green LED is out, this means that there was no synchronization during the night. To check if there is good reception, you simply turn off and on the device. The APOLLO II device will then check the signal. Note that the APOLLO II device needs at least 2 minutes to synchronize.

Synchronization by means of a master clock is a continuous process. That is the case because these systems only have synchronization of the seconds. Therefore, there is no special synchronization period from 3 to 5 a.m.

5. DTMF Interface + GSM

The DTMF Interface for Mobile Telephone (Siemens M20) has 2 important functions:

1) DTMF Function: stands for Dual Tone Multi-Frequency. This is standard in telephony. It seems to it that the keys on the telephone are translated into 2 tones, one with high and one with a low frequency. These 2 tones are then translated into key strokes in the telephone exchange. With this, the **Customer** can **start** executions from a distance via his **own** mobile phone or via a fixed telephone.

2) PROG function: with this, the Installer can program the installation from a distance via the APOLLO II communication center (PC) or via the CAMPA Workabout. This function can be used in large installations (carillons, cathedrals...) or when the distances are long.

5.1. Requirements

Receiver (on the installation)

- * DTMF-interface with incorporated Siemens M20 modem
 - * SIM (mobile) – card (for the Siemens M20 modem)
- with DATACALL : when also the ‘PROG’ function is desired
without DATACALL : when only the ‘DTMF’ function is desired

Observations:

DATACALL = system that the mobile phone uses to send data as well as for having conversations (VOICECALL).

When you don't need DATACALL, you can mostly use a pre-paid SIM-card. This is the most economic solution, because there are no monthly subscription costs.

Sender (Client and/or Installer at home)

* for 'DTMF'-function : mobile (incl. SIM card) or fixed telephone for Client

* for 'PROG'-function : PC with APOLLO II Comm. Center – for the PC-modem, we advise to connect a mobile phone to a PC – there are several possibilities to do this (for example, a Nokia 6210 with DLR-3P cable)

APOLLO II or TEMPORA master clock with software version 2.16 or higher

5.2. Installation

Introduce the SIM card in M20

Put the SIM card (with or without DATACALL) in the Siemens M20 master clock – here for, you first open the slit in the M20 (black housing) with a sharp object (push on the button near the slit), then you put the card in the slit and push the whole back in the M20.

Connect the DTMF interface for GSM Modem

Connect the DTMF interface:

The DTMF interface is always provided with all intern connections. The installer only has to make the 3 following connections:

- 1) Antenna: the loose delivered antenna has to be screwed on the connector on the outside of the DTMF interface.
- 2) Data cable (SUBD9) to APOLLO II or TEMPORA: this cable has to be connected with the SUBD9-connector (connector on the left side of the APOLLO II or TEMPORA) of the APOLLO II or TEMPORA master clock.
- 3) Power supply: the DTMF interface is provided with a universal power supply (100-230V AC). This has to be connected to the electric mains. You will see now that the green led of the Siemens M20 is flashing. This means that the SIM-card has not yet received a PIN-code or hasn't found a telephone network.

Program APOLLO II / TEMPORA

Next functions have to be programmed in the APOLLO II or TEMPORA master clock – take care that the APOLLO II (TEMPORA) master clock is provided with software version 2.16 or higher.

Function 1105 : serial interface

SERIAL INTERFACE 19.200 01 DATA REMOTE CONTROL BAUDRATE	UP DOWN OK BACK
--	--

Baudrate: has to be set to '19.200'

Ringers: number of times the phone rings, before the APOLLO II (TEMPORA) picks up – choose for this application '01'

Answer phone: has to be set to 'DATA'

Type of message: has to be set to 'REMOTE CONTROL'

Function 1402 : user code

USER CODE 0000 2357 OFF PIN CODE	+1 -1 OK BACK
--	--

Access code: user code – to secure the user menu – when '0000' is asked, no code is asked.

PIN-code: after power failure, the APOLLO II has to send the PIN-code from the SIM-card to the M20 – here you have to put the PIN code of the **SIM card that is in the M20**, for example 2357.

Attention: entrance of a wrong code can block the SIM card.

PIN-CD.for DTMF: set here whether the DTMF-system (to start executions from a distance) has to be protected with a code or not (the code is the PIN-code from the M20 – cf. 'working of the system' for more information).

Function 03: inputs (for 'DTMF' function)

Here you can assign an execution to all of the 10 keys of the phone.

Example: you want to assign the sequence 'Week mass' to key '1'

Function 030101: input type 01 = Sequence
Function 030102: input action 01 = Week mass
 01 time

Tip: to be able to stop the executions from a distance, we advise to program always a 'STOP' sequence in the master clock (for example on key '0'). A 'STOP' sequence can be made as follows: function 0601 – press 'OK' – press 'STOP' (red key) – press 'BACK' – and save the sequence with function 0603.

Switch on and off APOLLO II / TEMPORA

Switch the APOLLO II (TEMPORA) master clock OFF and then ON. At that moment, the APOLLO II (TEMPORA) will detect the DTMF interface and send the PIN code to the M20 (adjusted in function 1402, second number). When the M20 has received access to the telephone network and has received the correct code, the green led on the M20 will light **continuously**. This is a requirement for the good functioning of the system. When you like to test the SIM card, you can put it in a normal mobile phone and give the PIN code.

5.3. Working of the system

After the installation of the DTMF interface, the client can start the executions, which were programmed by the installer in function 03 with his own mobile or fixed telephone.

There for, the client has to call to the telephone number of the M20. When the parameter '*PIN-CD.for DTMF*' is ON in function 1402, this code will be asked. You will recognize this by short quick tones (5 tones a second) in the receiver. Give the right 'PIN-code'.

When you have given the right code, the sound will pass from quick to slow tones (1 tone a second). You hear the same signal when you call and no code is asked (*PIN-CD.for DTMF* = OFF).

On this moment, you can start the different executions by the 10 telephone keys.

5.4. Control-lamps

Green led on M20

OFF : control the connections of the DTMF interface – take care that a SIM-card is placed in the M20 master clock.

flashes: the M20 waits for the PIN-code to initialize the SIM-card. Take care that you have switched OFF and ON the APOLLO II (TEMPORA) and that the right code has be given to function 1402.

ON : the correct PIN-code of the SIM-card is received in the M20. You can call now to the M20 to start the executions.

The 4 control lamps on the DTMF interface-print (LD1 – LD4)

LD 1 : indicates that the processor on the interface print works – print = OK

LD 2 : DTMF receive: flashes 1 time, every time a valid DTMF code was received, this means every time you press a key on the telephone, after having made the connection with the DTMF interface (only by starting executions on a distance – 'DTMF' function)

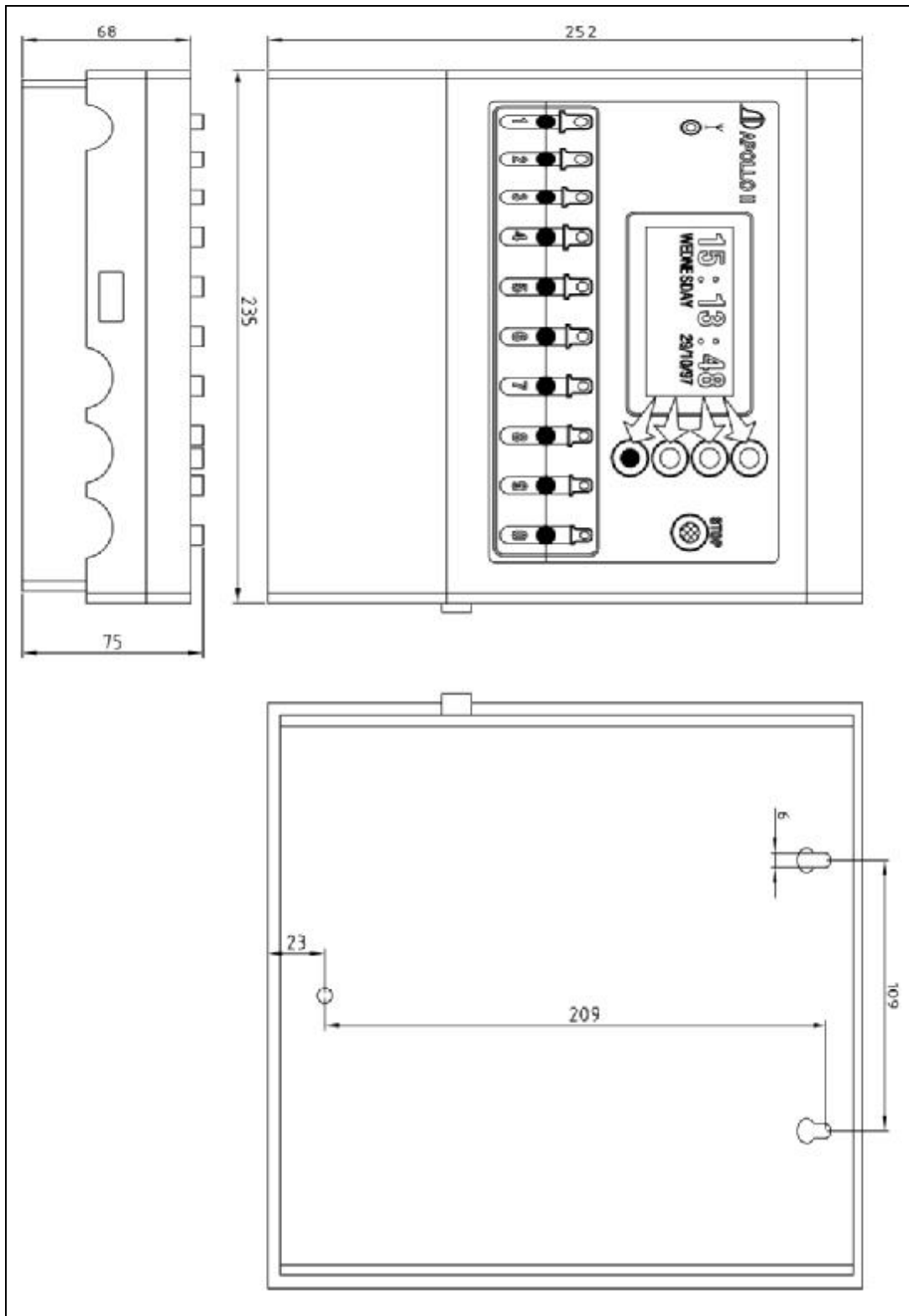
LD3 : ON when a PIN-code is asked (*PIN-CD.for DTMF* = ON).
OFF when no PIN-code is asked (*PIN-CD.for DTMF* = OFF) or when the correct code was already given via the phone.

LD4 : CD Carrier DETECT, this LED means that there is data communication (only by remote control – via PC and APOLLO II Comm. Center).

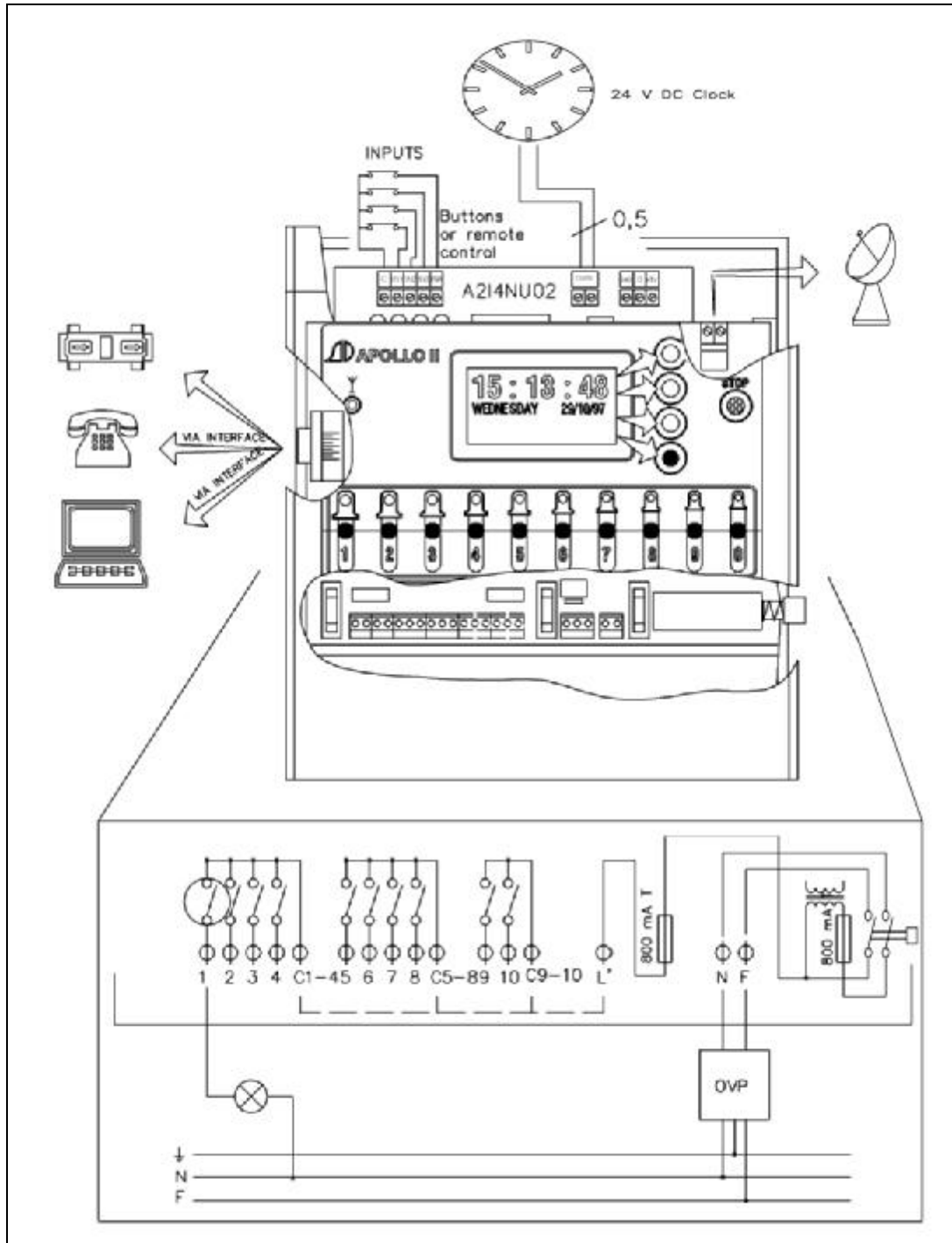
G. ANNEXE : DRAWINGS

DRAWINGS OF THE APOLLO II DEVICE

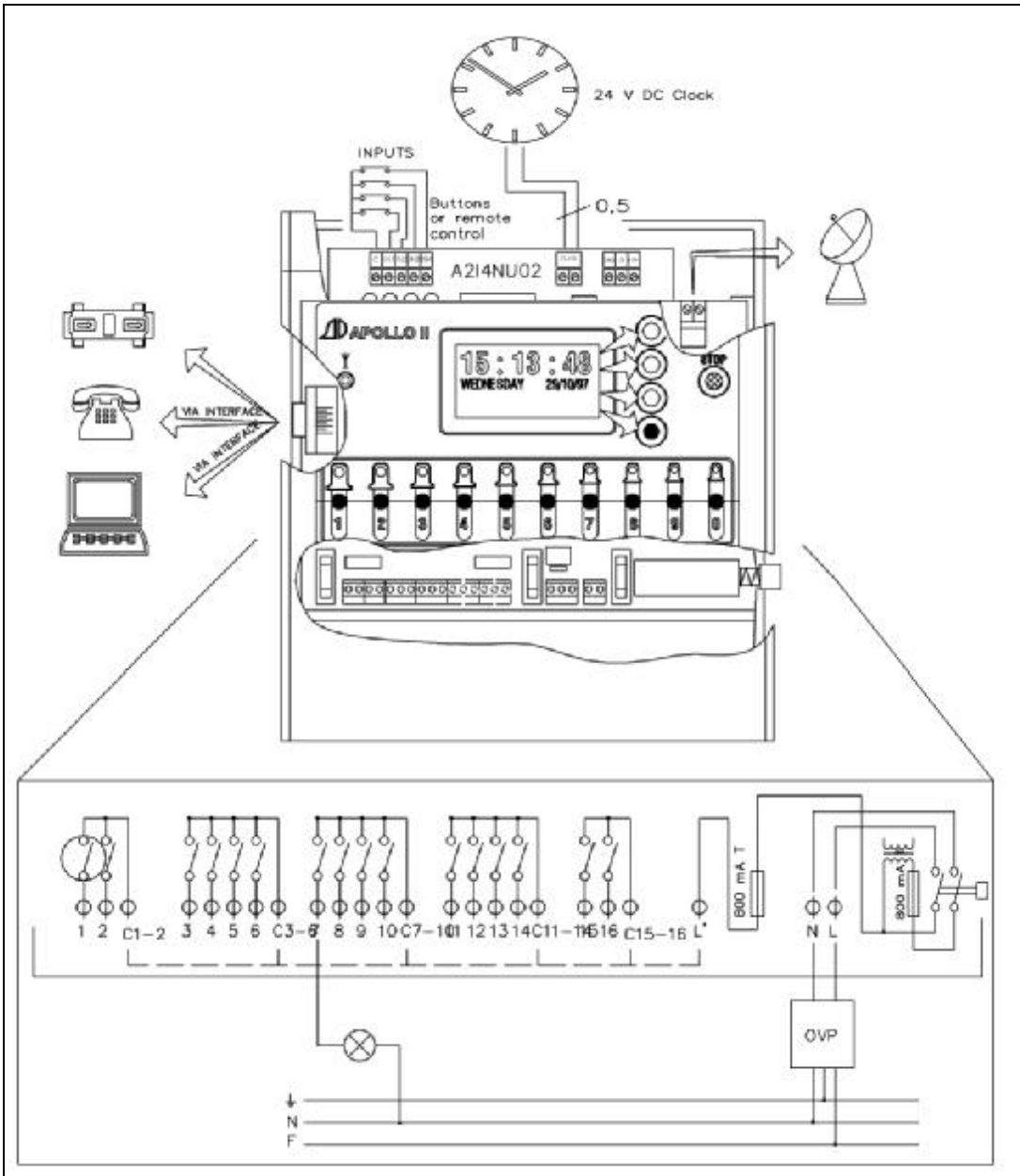
APOLLO II DIMENSIONS



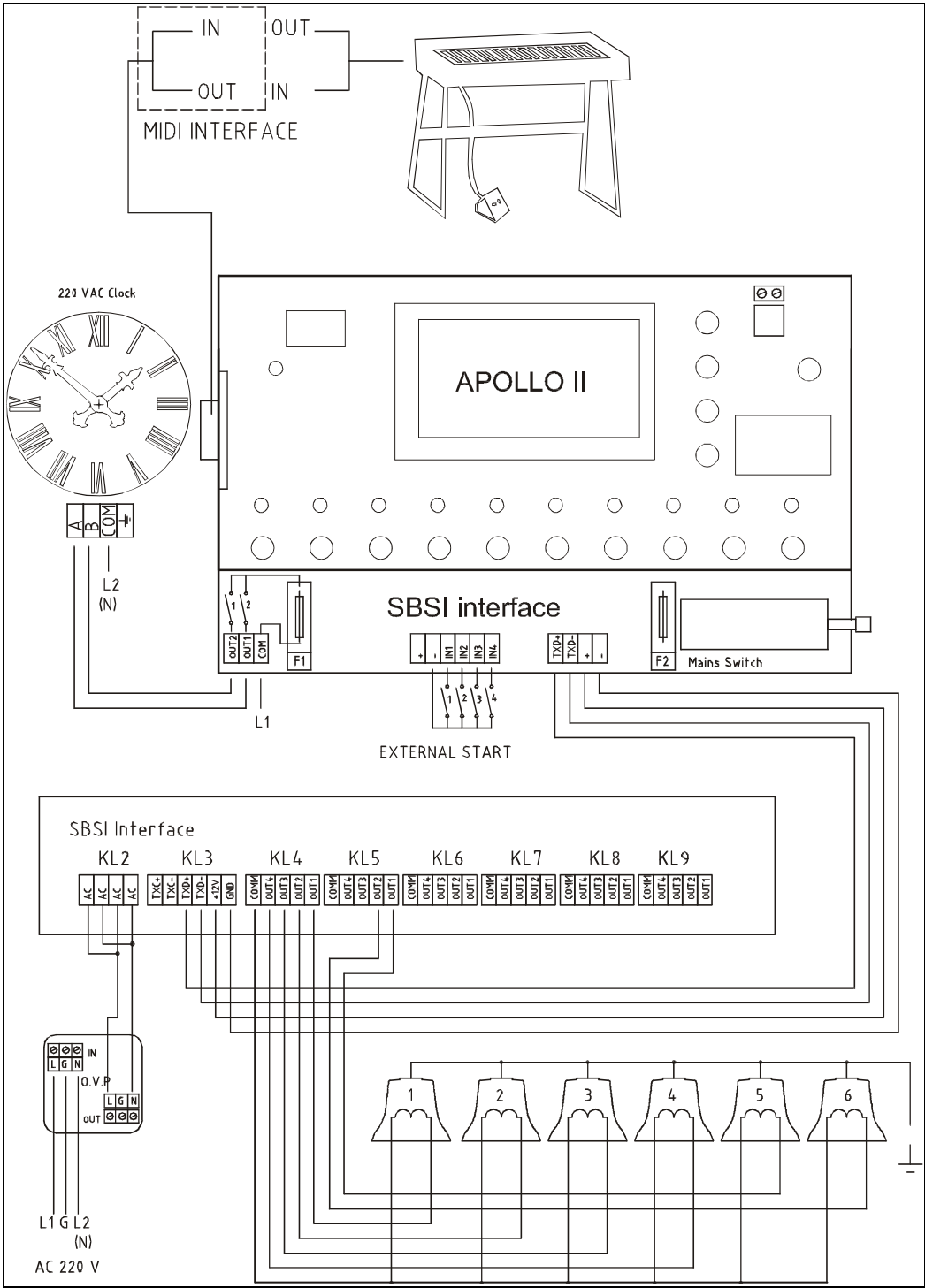
APOLLO II – 10 OUTPUTS



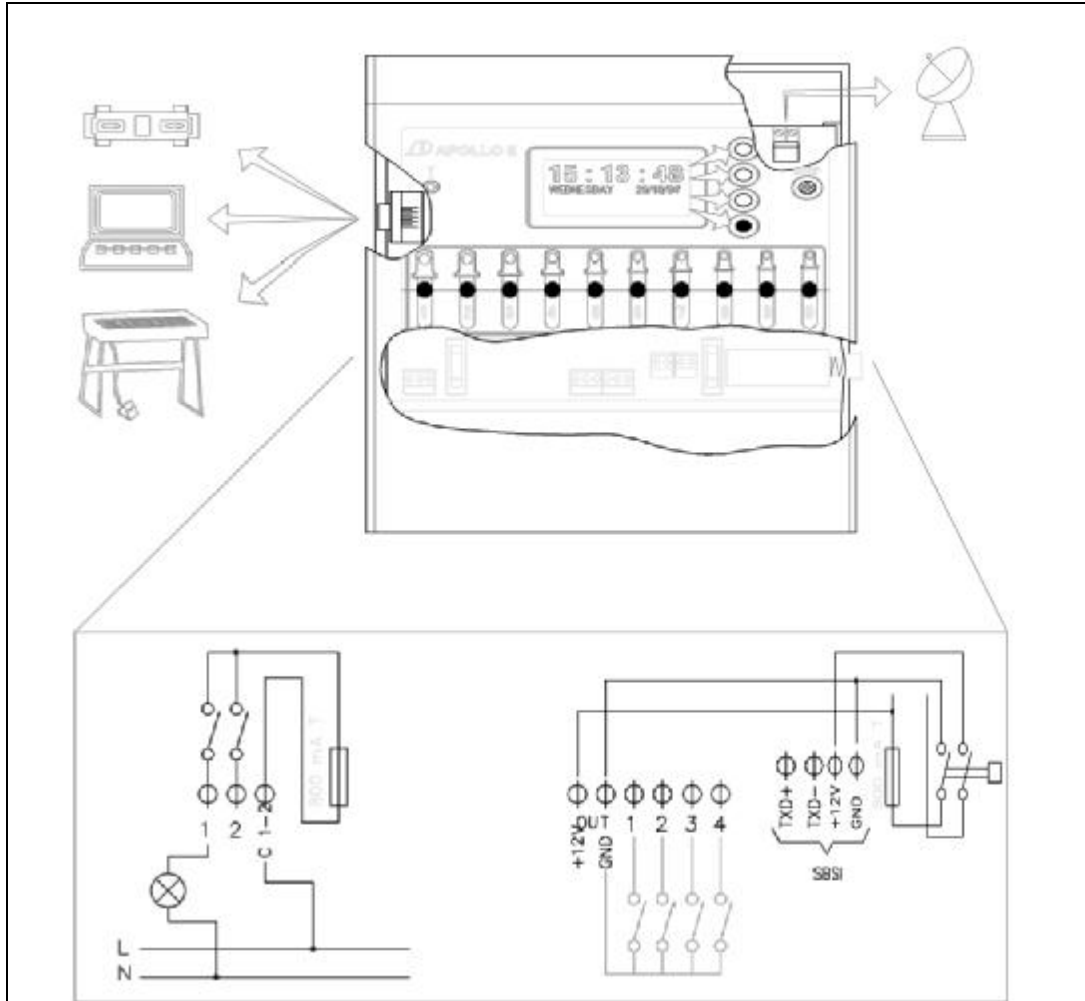
APOLLO II 16 OUTPUTS



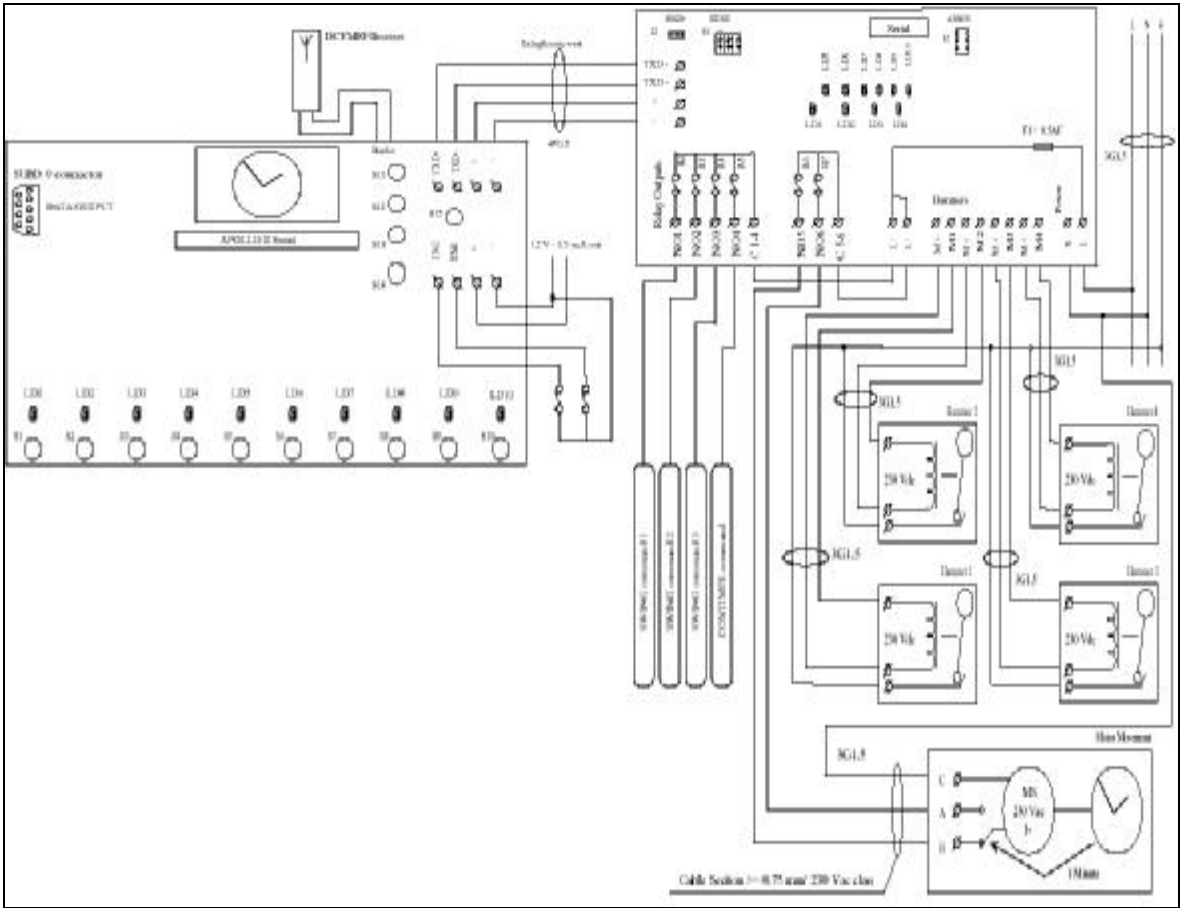
APOLLO II-SBSI (MASTER CLOCK + OUTPUT BOARD)



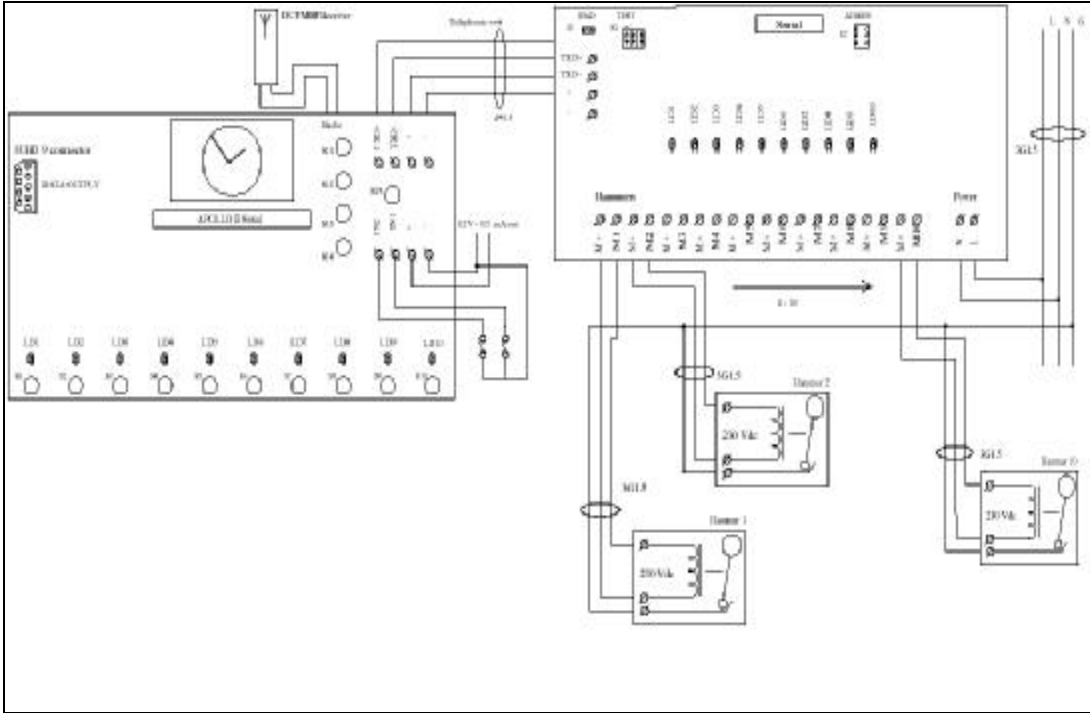
APOLLO II-SBSI (master clock)



SERIAL APOLLO II WITH OUTPUT BOARD (6 REL + 4 SORTIES)

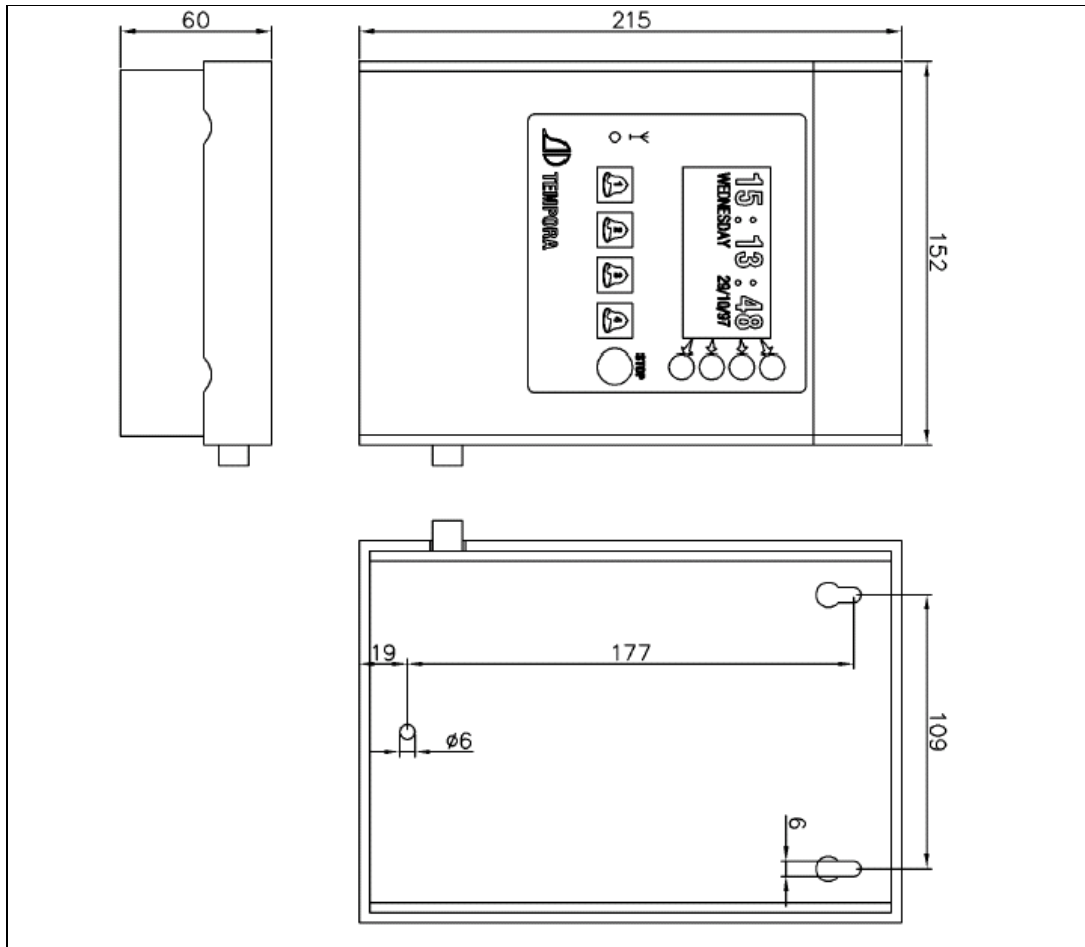


SERIAL APOLLO II WITH OUTPUT BOARD (10 FET OUTPUTS)

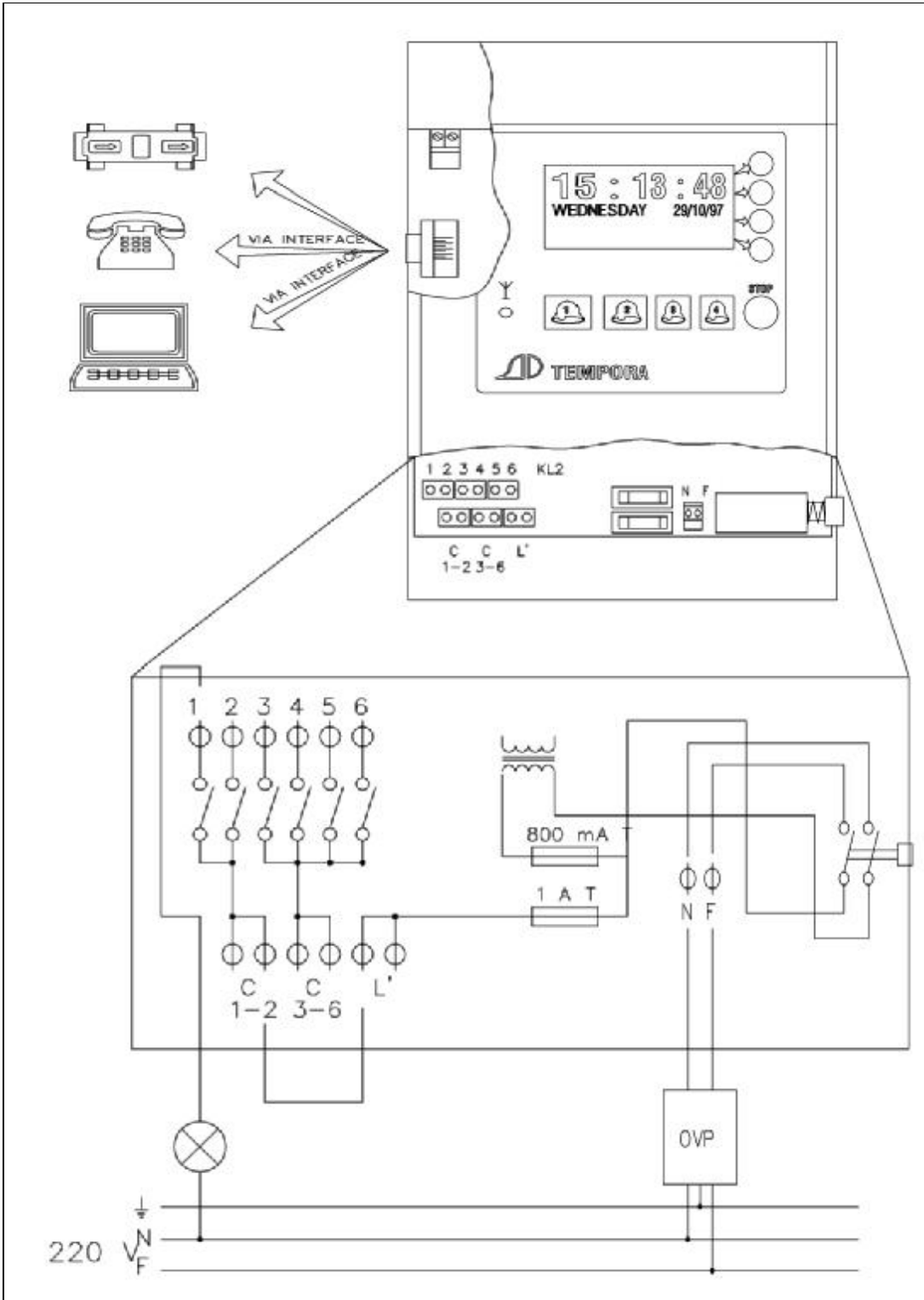


DRAWINGS OF THE TEMPORA DEVICE

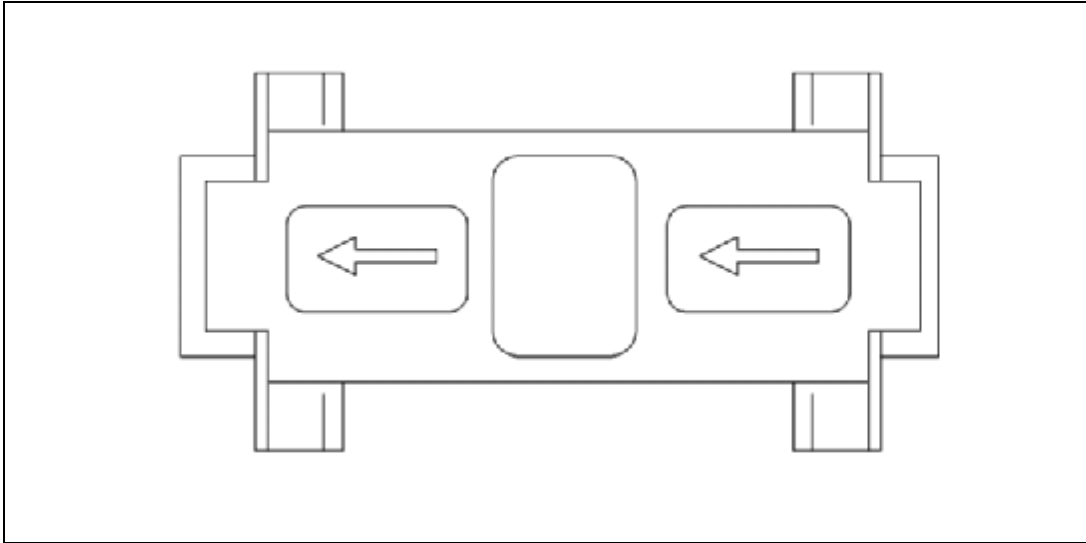
TEMPORA - DIMENSIONS



TEMPORA – 6 OUTPUTS with optional print

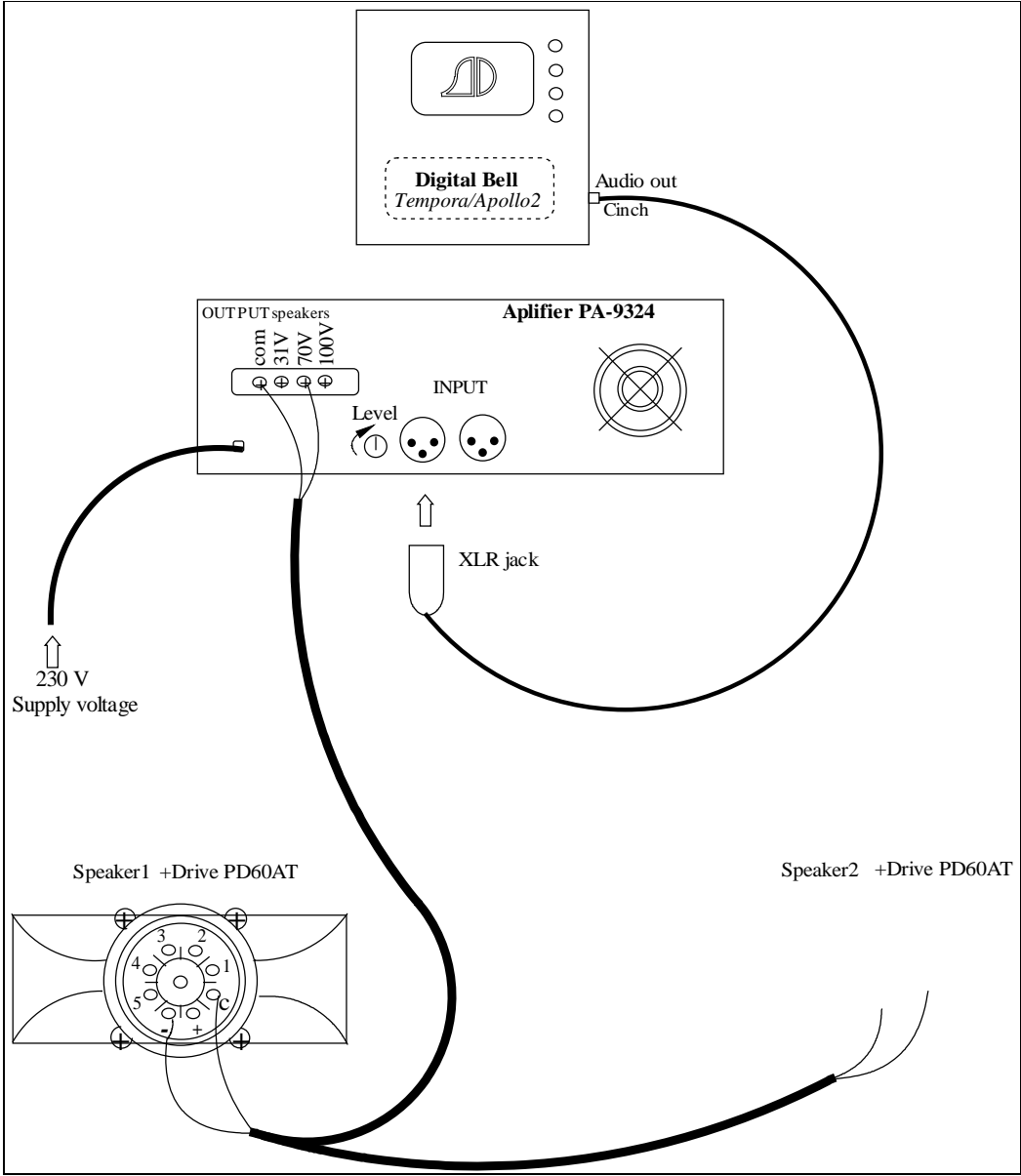


PIM

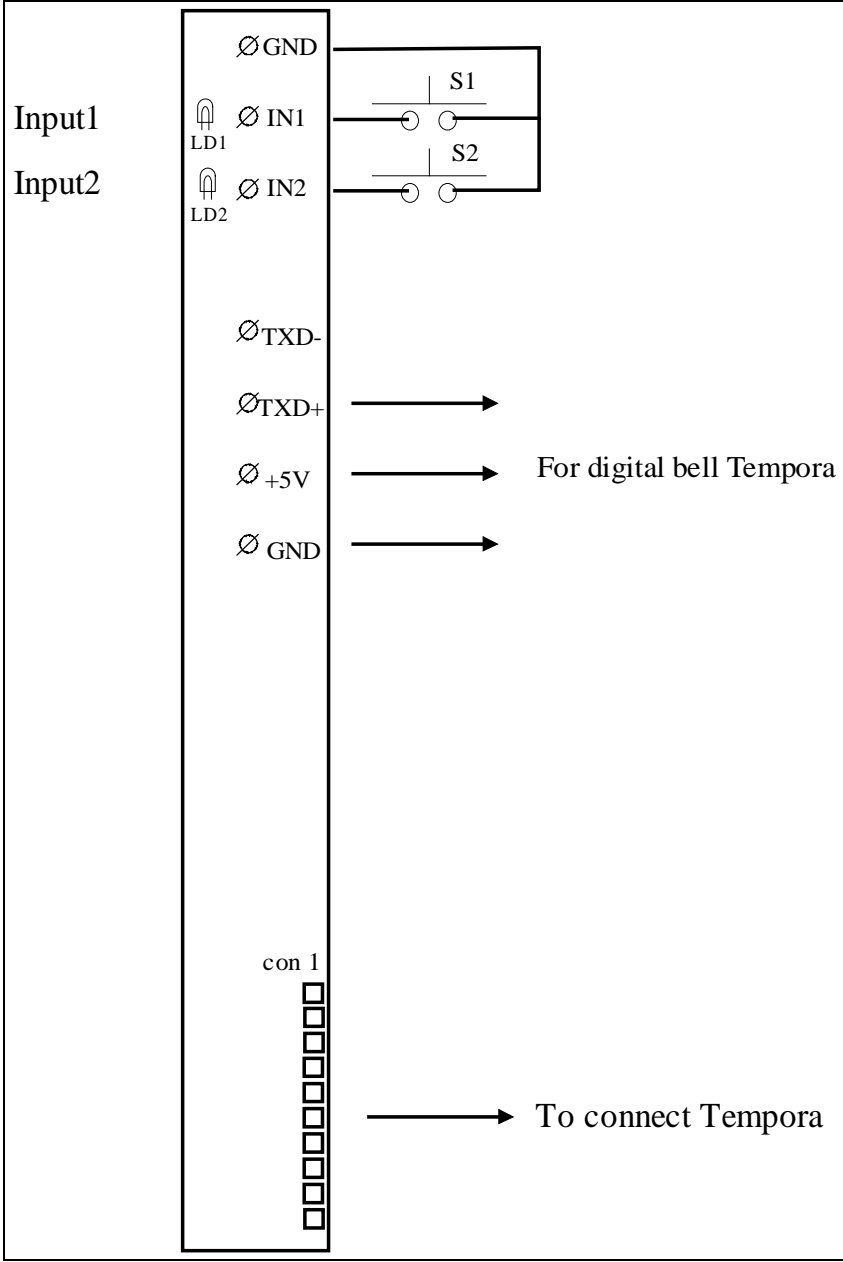


OPTIONS FOR THE APOLLO II AND THE TEMPORA DEVICE

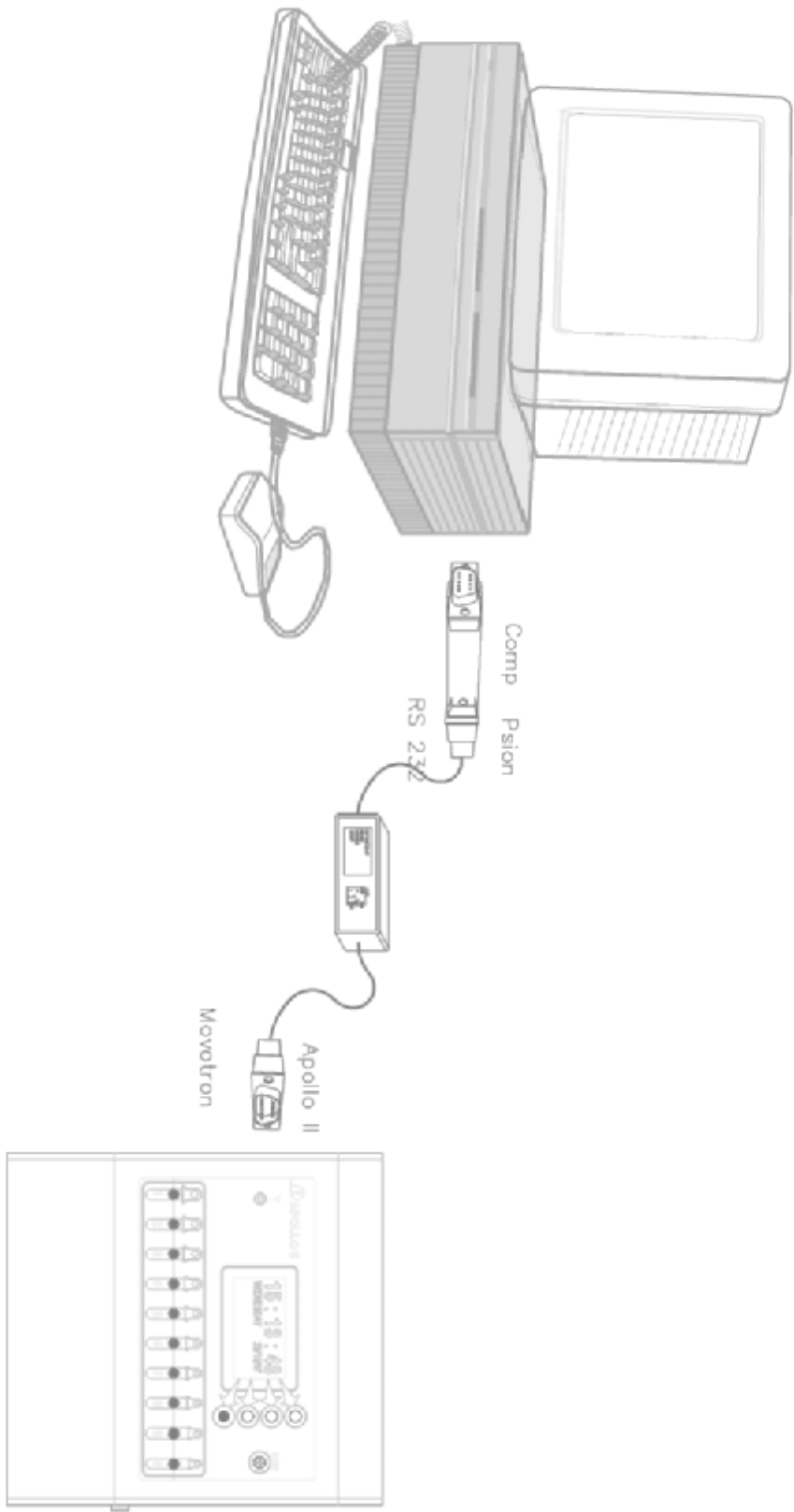
AUDIO CONNEXION DIGITAL BELL TEMPORA (APOLLO II)



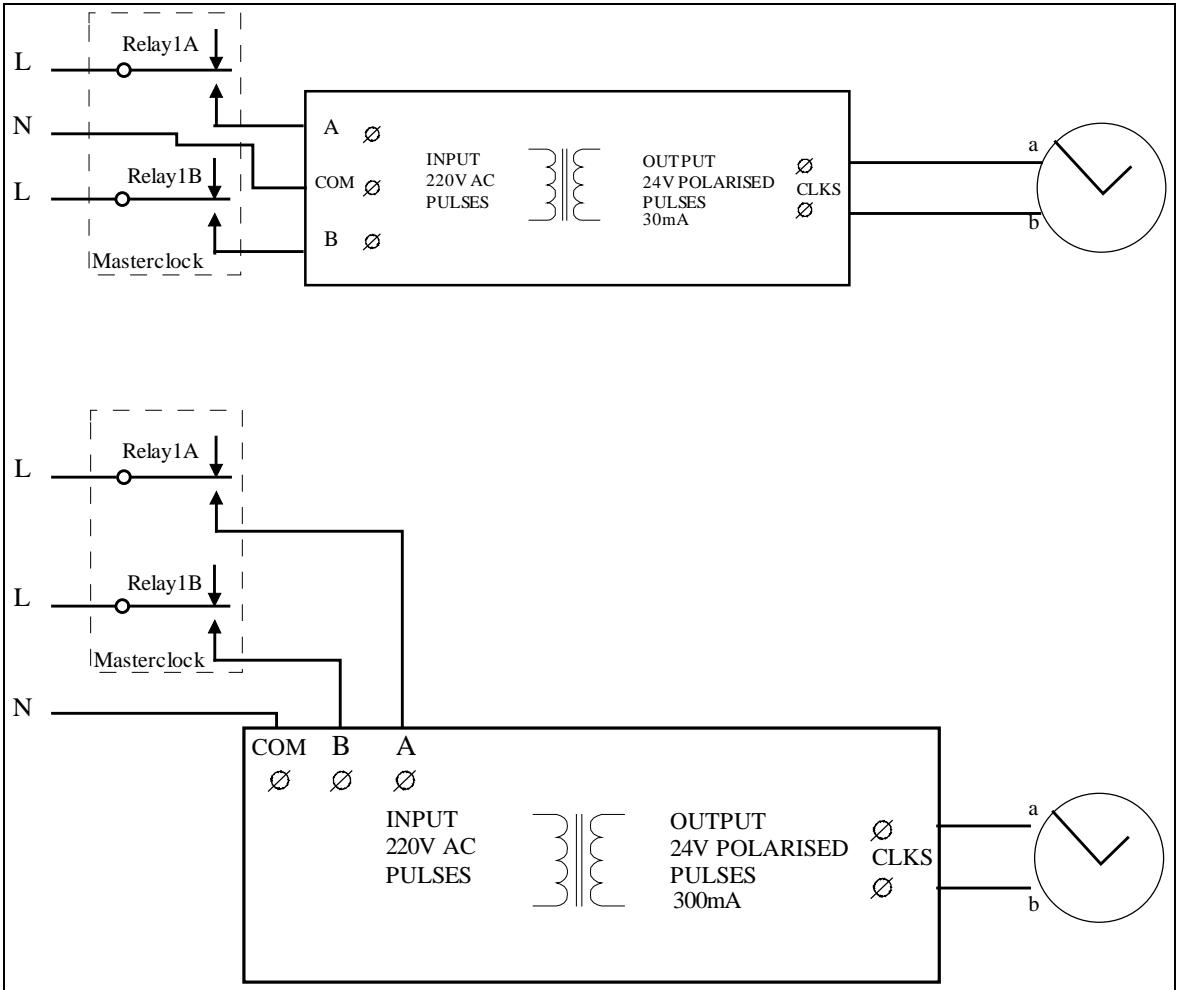
CONNEXION OPTIONAL PRINT TEMPORA (INPUTS)



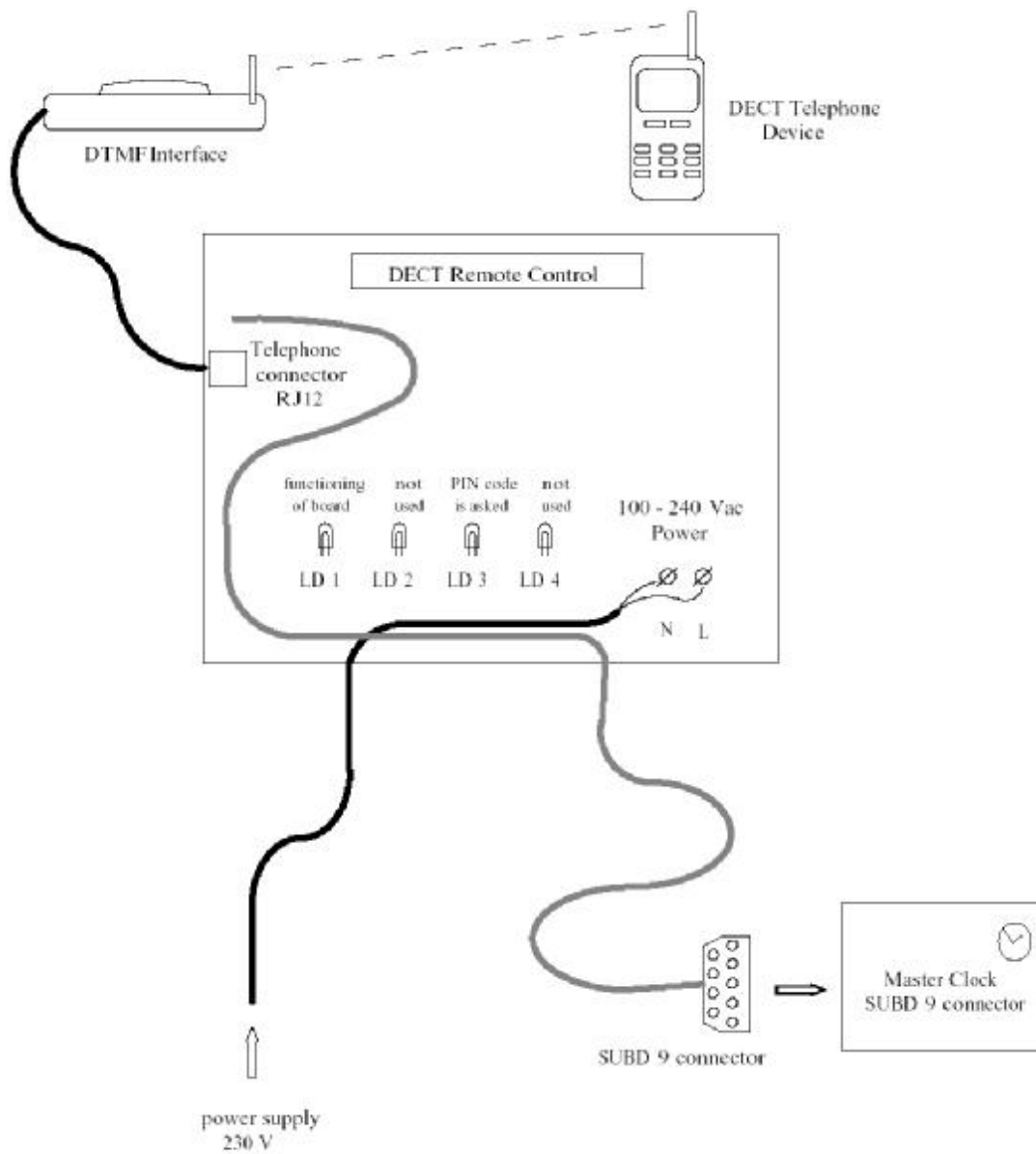
CONNEXION OF THE APOLLO II – COMPUTER



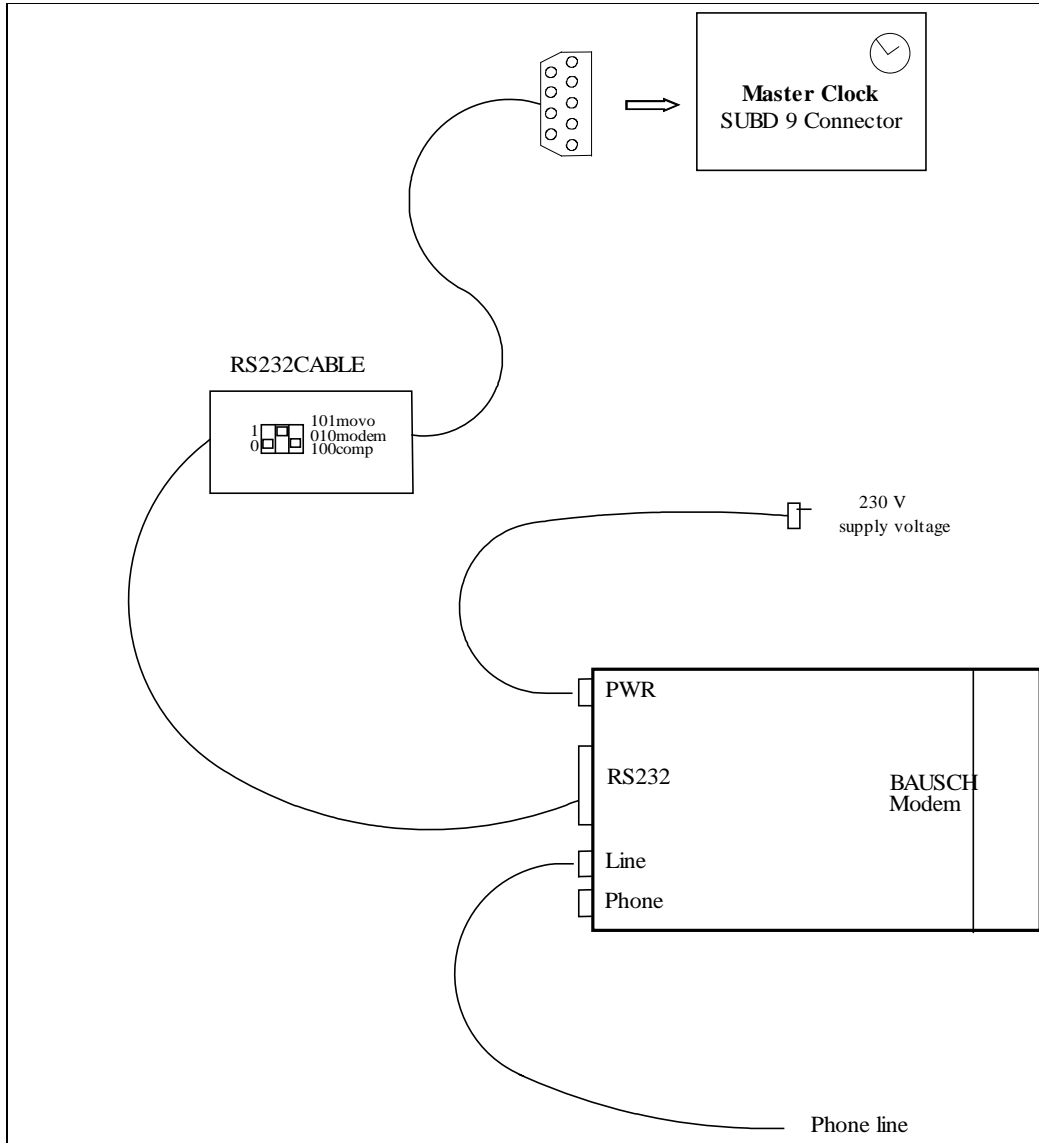
CONNEXION EXTERNAL 24V POLARISED IMPULSES CONVERTOR



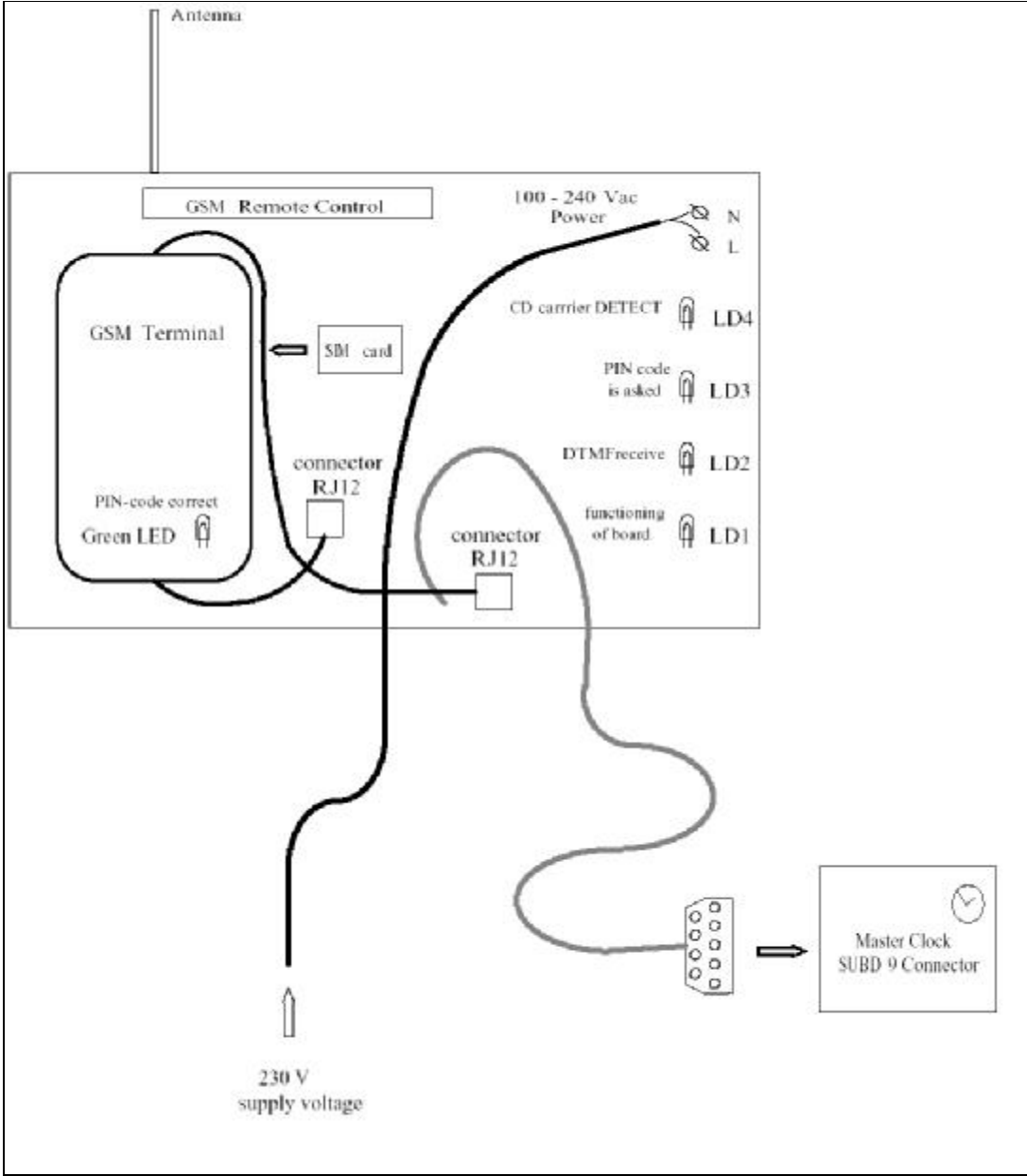
DECT - REMOTE CONTROL



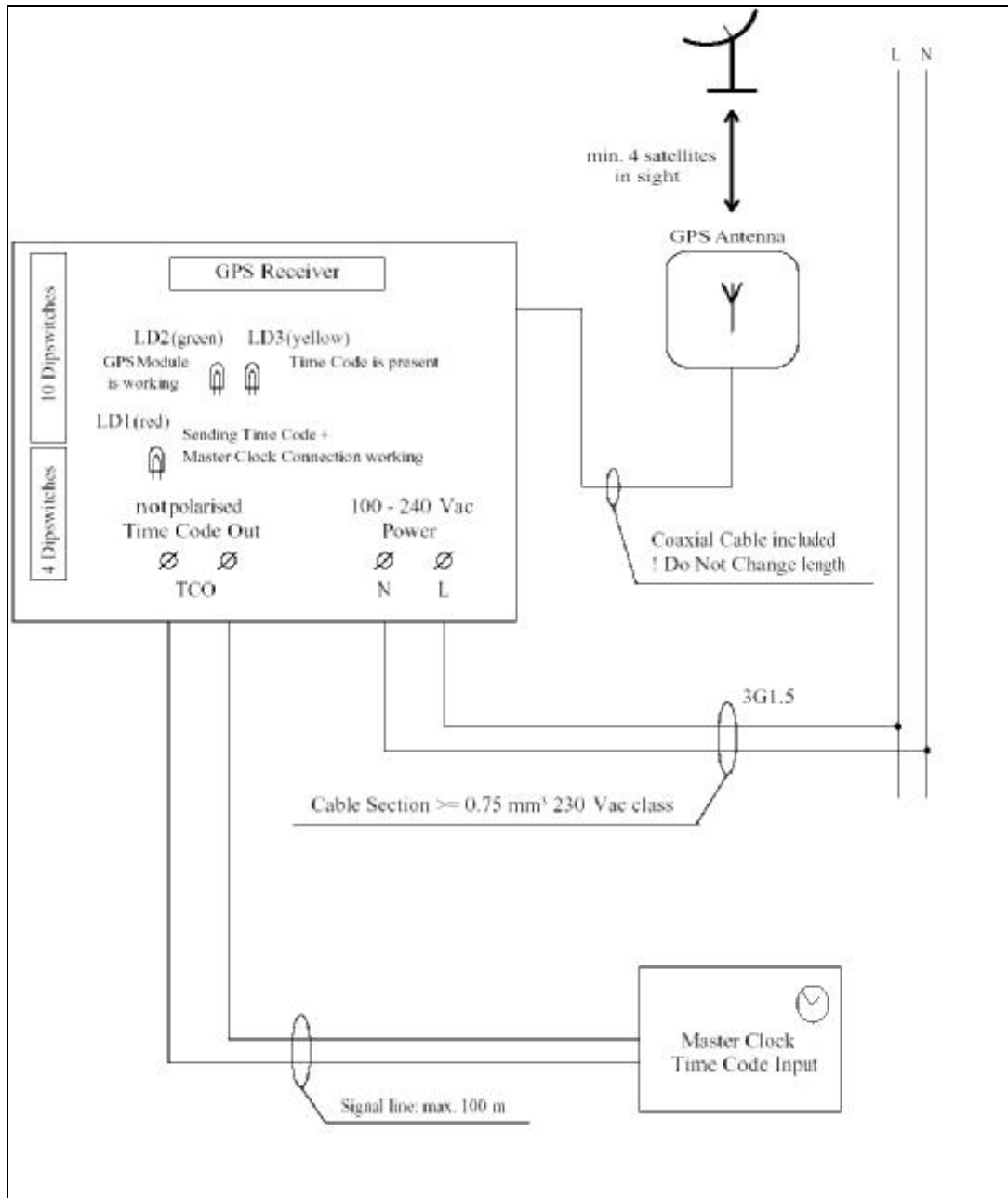
CONNEXION FIXED MODEM (DTMF)



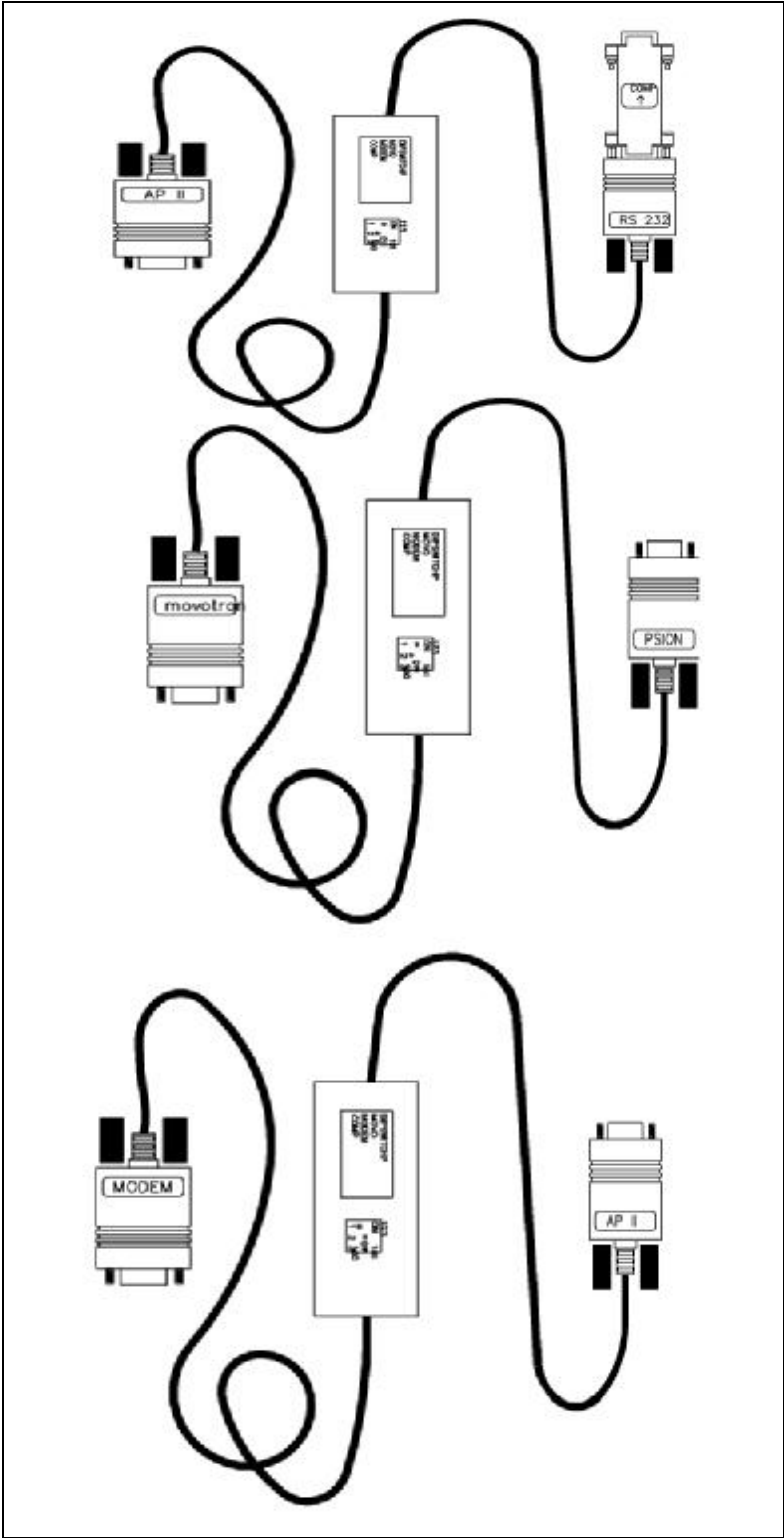
MOBILE TELEPHONE – REMOTE CONTROL



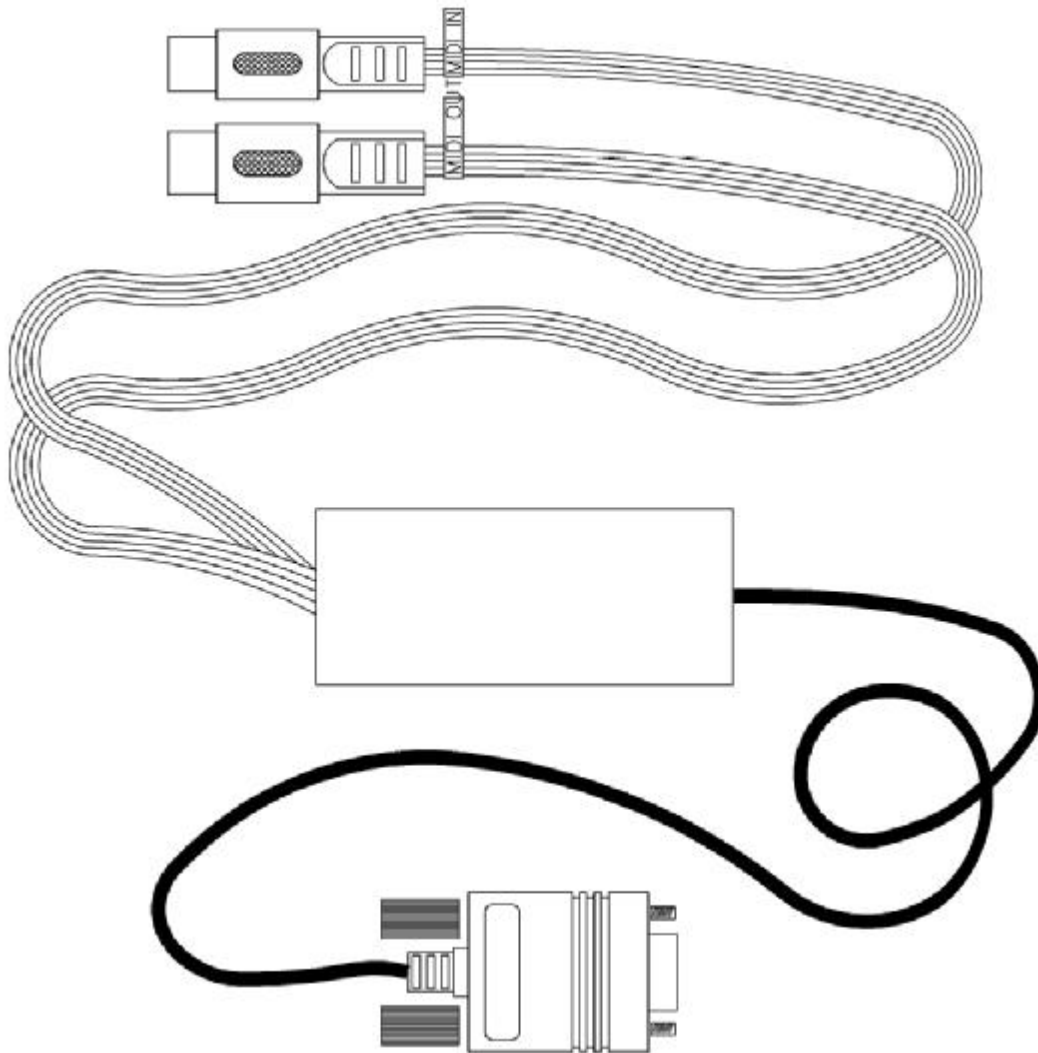
GPS-RECEIVER



SERIAL CABLE- 3 POSSIBILITIES



MIDI CABLE



G. SURVEY OF ALL FUNCTIONS

- 1) EXECUTIONS (function 01) => 1.1) RINGING (function 0101)
 1.2) PLAY SEQUENCE (function 0102)
CHAPTER 9 1.3) PLAY GROUP (function 0103)
 1.4) SWITCH ON CONTINUOUS (function 0104)
 1.5) PROGRAM BLOCKS (function 0105)
 1.6) MANUAL TOLLING (function 0106)
 1.7) PAUSE (function 0107)
- 2) USER MENU (function 02) => 2.1) USER MENU xx (xx = 1 to 30) (function 02xx)
 => 2.1.1) USER TYPE xx (function 02xx01)
 2.2.2) USER ACTION xx (function 02xx02)
CHAPTER 13 2.2) FUNCTION KEY 1-0 (function 0231)
- 3) INPUTS (function 03) => 3.1) INPUT xx (xx = 1 to (depending on type device)) (function 03xx) : automatic detection of number
 => 3.1.1) INPUT TYPE xx (function 03xx01)
 3.1.2) INPUT ACTION xx (function 03xx02)
CHAPTER 14
- 4) PROGRAMS (function 04) => 4.1) SINGLE PROGRAMS (function 0401)
 => 4.1.1) 10 SINGLE RINGING PROGRAMS (function 040101)
 4.1.2) 10 SINGLE SEQUENCE PROGRAMS (function 040102)
 4.1.3) 10 SINGLE GROUP PROGRAMS (function 040103)
 4.1.4) 10 SINGLE CONTINUOUS PROGRAMS (function 040104)
 4.1.5) 10 SINGLE BLOCK PROGRAMS (function 040105)
CHAPTER 12 => 4.2) PERMANENT PROGRAMS (function 0402)
 => 4.2.1) 90 PERMANENT RINGING PROGRAMS (function 040201)
 4.2.2) 100 PERMANENT SEQUENCE PROGRAMS (function 040202)
 4.2.3) 30 PERMANENT GROUP PROGRAMS (function 040203)
 4.2.4) 50 PERMANENT CONTINUOUS PROGRAMS (function 040204)
 4.2.5) 80 PERMANENT BLOCK PROGRAMS (function 040205)
- 5) RINGING PARAMETERS (function 05) => 5.1) RINGING xx (xx = 1 to 10) (function 05xx)
CHAPTER 8 5.2) RINGING ORDER (function 0511)
 !! Only the parameters are changed here. The ringing itself is started in executions ringing (function 0101).

- 6) SEQUENCES (function 06) => 6.1) RECORD SEQUENCE (function 0601)
 6.2) TEST SEQUENCE (function 0602)
CHAPTER 10 6.3) SAVE SEQUENCE (function 0603)
 6.4) ERASE SEQUENCE (function 0604)
 6.5) NAMES SEQUENCE (function 0605)
 6.6) PROPERTIES SEQUENCE (function 0606)
 => 6.6.1) SEQUENCE xx (xx = 1 to 127)
- 7) GROUPS (function 07) => 7.1) ACTIVE GROUP (function 0701)
 7.2) TIME BETWEEN SEQ (funktion 0702)
 7.3) CONTENTS GROUPS (function 0703)
 => 7.3.1) GROUP xx (xx = 1 to 24) (function 0703xx)
 => 7.3.1.1) GROUP xx SEQ yy (xx = 1 to 24)
 (yy = 1 to 96) (function 0703xxyy)
CHAPTER 11 7.4) NAMES GROUPS (function 0704)
 => 7.4.1) NAME GROUP xx (xx = 1 to 24) (function 0704xx)
- 8) CONTINUOUS OUTPUTS (function 08)=> 8.1) NAME CONTINUOUS xx (xx = 1 to 24) (function 08xx)
CHAPTER 7 !! Only the names of the continuous outputs are changed here
- 9) PROGRAM BLOCKS (function 09) => 9.1) NAME PROGRAM BLOCK xx (xx = 1 to 24) (function 09xx)
CHAPTER 6 !! Only the names of the program blocks are changed here
- 10) HAMMERS (function 10) => 10.1) PULSE DURATION <pitch> (pitch = C1 to C6) (function 1001-1061)
CHAPTER 5
 !! Only the tolling parameters are changed here. The actual tolling is started in executions-manual tolling (function 0106)
- 11) SPECIAL OUTPUTS (function 11) => 11.1) BELFRY WINDOW (function 1101)
 11.2) ROMAN KNELL (function 1102)
CHAPTER 4 11.3) MIDI (function 1103)
 11.4) SPEECH (function 1104)
 11.5) SERIAL OUTPUT (function 1105)
- 12) CLOCKS (function 12) => 12.1) INTERNAL CLOCK (function 1201)
 => 12.1.1) CHANGE TIME (function 120101)
 12.1.2) CHANGE DATE (function 120102)
 12.1.3) TIME FORMAT (function 120103)
 12.1.4) RUNNING SPEED (function 120104)
 12.1.5) SHIFT RADIO (function 120105)
 12.1.6) SUMMER TIME (function 120106)

CHAPTER 3

- 12.2) SLAVE CLOCK 1 (function 1202)
 - => 12.2.1) TIME SLAVE CLOCK 1 (function 120201)
 - 12.2.2) TYPE SLAVE CLOCK 1 (function 120202)
- 12.3) SLAVE CLOCK 2 (function 1203)
 - => 12.3.1) TIME SLAVE CLOCK 2 (function 120301)
 - 12.3.2) TYPE SLAVE CLOCK 2 (function 120302)

13) RELAYS (function 13)

- => 13.1) CONTROL RELAYS (function 1301)
- 13.2) RELAYS xx (xx = 2 to (depends on type of device)) (function 13xx) : automatic detection of number

CHAPTER 2

14) SET-UP (function 14)

- => 14.1) CONTRAST DISPLAY (function 1401)
- 14.2) USER CODE (function 1402)
- 14.3) FUNCTION CODE (function 1403)
- 14.4) INTERNAL MEMORY (function 1406)
- 14.5) SOFTWARE (function 1404)
- 14.6) LOADING PIM (function 1405)

CHAPTER 1