

**Focus Microwaves Inc.**  
1603 St-Regis  
D.D.O., Quebec  
H9B 3H7 Canada  
Tel: 514-684-4554 Fax: 514-684-8581  
E-mail: [info@focus-microwaves.com](mailto:info@focus-microwaves.com)  
Website: <http://www.focus-microwaves.com>



## *Application Note 5*

---

### **User Defined GPIB Instrument Drivers for the CCMT System**

#### **Introduction**

The CCMT software, version 4.0, offers the possibility to the users to define their own instrument drivers and use them from within the Power or Noise measurement programmes of the CCMT software.

The CCMT software includes GPIB instrument drivers for over 50 popular instruments in the RF and microwave frequency area.

There are situations, however, where the introduction of new instruments, or the availability of non included instruments, require the possibility to define 'USER SPECIFIC' instrument drivers, without modifying the executable program or wait for an upgrade.

User specific instrument drivers can be defined for all instruments used in the Load Pull and Noise setup except for the Network Analysers.

Developing your own instrument drivers does not require programming knowledge and can be accomplished by following a few basic rules.

The CCMT Utilities software provides all required tools to make the development of the new drivers an easy task..

Once you have introduced your own driver, the instrument(s) you selected will be addressed, initialised, configured and triggered for reading and the response will be converted into meaningful data to be saved in a Load Pull file or processed for Noise Parameter calculations.

## What is an Instrument Driver?

An instrument driver is a set of programming commands that can be sent to a GPIB instrument in order to configure, trigger and read its response. These commands are sent in form of character strings (mnemonics) via the GPIB interface card and GPIB Driver program to the instruments.

For example the following text configures and reads the HP-438 power meter if sent as mnemonics code to the GPIB at address 13:

CLEAR 13	= Clears device 13
OUTPUT 13;RALG;TR3;AP;KB 98.0 EN	= Auto Range, Trigger, Channel A, Cal-Factor
OUTPUT 13;AP	= Switch to Channel A
ENTER 13	= Receive data from instrument

In an automatic operation these strings have been included in the source code of the program, compiled and linked and cannot be modified from the keyboard.

There exist though the possibility to retrieve these strings from files and send them to the GPIB. The CCMT user defined driver utility uses this approach to transfer the command strings to the instruments, without modifying the executable files.

## Components of an Instrument Driver

In order to talk and read an instrument it has first to be initialized, then configured, triggered and its response has to be converted to meaningful data according to its format. In addition the response of the instrument might need to be corrected for setup losses depending if it is power, frequency, voltage, current or else...

The output power in a load pull setup, for example must be multiplied with the setup loss in order to be transferred to the DUT reference port, whereas the frequency of an oscillator does not need to be modified.

We therefore divide the communication with an instrument in three steps:

- Step 1: Initialization
- Step 2: Configuration & Trigger
- Step 3: Reading and Processing the Response

The Initialization happens only once, when the measurement starts. It includes the range settings, entry of calibration factors, timing conditions etc.

The Configuration/Trigger happens each time the instrument is addressed. Only the parameters that change between measurements should be modified at this point. A trigger command should also be included, if the Initialization does not include a continuous trigger command.

Reading and Processing of the Response follows each Configuration session.

It includes information on how to convert the response-string to real numbers and if and how to correct the readings for setup losses. As an additional feature the Processing directives include the capability of user defined numerical correction factors (on top of the tuner/setup dependent losses).

These components of an instrument driver are all included in three ASCII (text) files (per selected instrument)

- **USERINI $x$ .PAR** for INI-tialization
- **USERCOM $x$ .PAR** for COM-mand
- **USERFOR $x$ .PAR** for FOR-mat

where  $x$  is the instrument type used ( $x = 2...9$ , see table 1)

## Selection of User Defined Instruments

The user can send unlimited number of GPIB commands to an unlimited number of GPIB instruments. It is obvious, however, that he cannot exceed the **maximum number** of addressable instruments (**15**) in the same setup and he should also avoid **GPIB address conflicts**.

At this point this is left to the discretion of the user.

As you can see, by invoking the **Instrument Selection** utility of the CCMT Shell, the following instruments can be programmed by the user (=USER DEFINED).

### Instrument Family associated to Index < $x$ > in USER files

Item	Instrument Family	Data Measured
$x = 2$	Output Power Meter	Power [dBm]
$x = 3$	Noise Analyzer	Noise Fig & Ass. Gain [linear]
$x = 4$	Main Signal Source	Set Power [dBm] and Freq [GHz]
$x = 5$	Frequency Counter	Frequency [Hz]
$x = 6$	Spectrum Analyzer	Intermod [dBc], Peak Power [dBm]
$x = 7,0$	DC Power Supply	7=Vds [Volt], 0=Ids [Ampere]
$x = 8$	Input Power Meter	Power [dBm] (connect to coupler)
$x = 9$	Second Signal Source	Set Power [dBm] and Freq [GHz]

Table 1. Selection of User defined Instruments

## Structure and Features of USER....PAR Files

### 1. Instrument(s) Initialization

File **USERINI $x$ .PAR** type ASCII, must be in C:\CCMT

<u>Format</u>		<u>Meaning</u>
#N		Address of 1st instrument
\$1	(optional)	<b>\$1</b> display GPIB commands as send
command1		<b>\$2</b> display GPIB commands and prompts
command2		<b>\$0</b> or omit: No display as send
...		
...		All commands send to Instrument address <b>N</b>
#M		Address of 2nd instrument etc.
\$1	(optional)	as above
command1		
command2		
...		
...		All commands send to Instrument address <b>M</b>
##		File termination.

NOTE: A **command** is a character string (MNEMONICS) as described in the GPIB programming section of each programmable instrument.

## ***2. Instrument(s) Configuration - before reading***

File **USERCOM $x$ .PAR** type ASCII, must be in C:\CCMT

<u>Format</u>		<u>Meaning</u>
#N		Address of 1st instrument
\$1	(optional)	<b>\$1</b> display GPIB commands as send
command1		<b>\$2</b> display GPIB commands and prompts
command2		<b>\$0</b> or omit: No display as send
...		
#M		Address of 2nd instrument etc.
\$1	(optional)	as above
command1		
command2		
...		
##		File Termination.

## ***3. Reading Format***

According to the **output format** of each GPIB programmable instrument (see instrument manual in the GPIB programming section or verify using the GPIB.EXE program - Utilities Menu, key

F3) the character string send by the instrument to the GPIB has to be interpreted by the CCMT program properly. There are different possibilities on how to read and convert the characters of this string into either integers or real (float) numbers. Some instruments also send message characters, which have to be discarded. The **format information** required by the CCMT program is included in the ASCII file **USERFORx.PAR**.

File **USERFORx.PAR** type ASCII, must be in C:\CCMT

<u>Format:</u>	<u>Meaning</u>	
%f,%f	Example of reading format.	See table 2
0,1	2 integer numbers i,j;	Operation see page 6, Loss corrections
5.2,4	2 float numbers (factors).	Each measured value will be multiplied with the corresponding factor. If these factors are set to 1 or omitted the measured values will not be affected. To be effective <b>both numbers must be present</b> even if one of them has to be = 1.0

## Reading Format Conventions

According to the **output character string** of the addressed instrument you have the following options.

Data Type	Format
Read a real (float) number	%f
Read an integer number	%d
Read a double precision number	%lf
Discart N characters	%*Nc

**Table 2: Reading format conventions**

**Examples:**

<u>Instrument Response</u>	<u>Use Reading Format</u>
1.403E0	%f (convert one float)
1.4023,423	%f,%d (convert one float, one integer)
PID 1.234E3	%%*3c%f (discard 3 characters, convert one float)
3.430044E49	%lf (read a double float; use if number >3.4E38)

**Remember:** Only the **last addressed instrument**, as defined in **USERCOMx.PAR** will be read.

**Loss Corrections**

The 2 integer numbers **i** and **j** in the file **USERFORx.PAR** give access to manipulations using the **Power Loss** of the active tuner at the actual tuner positions. If a quantity is measured at the **output of the setup** then **multiplying** it by the loss will transfer it to the DUT output reference plane. This happens if **i=1**. If **i=0** then the effect of the loss is ignored. If **i=-1** then the measured quantity is **divided** by the **Loss**. This is required if we measure a quantity **before** the setup (at the input coupler for example) and require the power available at the DUT input. The same operations are possible independently for the second measured quantity, by **manipulating** the integer **j** in the file **USERFORx.PAR**.

**Examples**

Example for configuring and reading the **hp436** Power Meter used here as an **OUTPUT POWER METER (x=2)**:

<b>USERINI2.PAR</b>	<b>USERCOM2.PAR</b>	<b>USERFOR2.PAR</b>
#13 (address)	#13	%%*3c%f (discard 3 char,read 1 float)
\$1 (display=yes)	\$0 (no display)	1,0 (multiply with loss)
TIME OUT 5	T (trigger)	2.0,1.0 (multiply power with 2.0)
REMOTE 13	## (end of file)	
CLEAR 13		
9D+ (auto range,dBm)		
## (end of file)		

Remark The HP436 is one of the instruments which responds with a character string such as **PID 1.403E0**.

Example for configuring and reading the **HP5351** Frequency Counter (x=5):

<b>USERINI5.PAR</b>	<b>USERCOM5.PAR</b>	<b>USERFOR5.PAR</b>
#3	#3	.*c%f
\$1	\$0	1.0,1.0
RESET	TRIGGER	1.0,1.0
RESOL,3	##	
SAMPLE,HOLD		
TIME OUT 5		
##		

### **Definition and Modifications of USER Factors**

The CCMT software provides the possibility of '**on line**' modification of the Factors defined in the **USERFORx.PAR** files. These factors can be used voluntarily to modify the values measured by the GPIB instrument.

This can be done either using **Alt-J** from the F-Keys menu, or, during automatic load pull measurements, by interrupting the routine using '**Esc**'.

The new USER FACTORS will be saved in the file **USERFORx.PAR** and the remaining of the reading routine will consider the new values.

In order to tell the software which instrument driver has to be modified a parameter file **USER\_SEL.PAR** has to be defined before hand. This file includes only one integer number **x**, as defined in table 1.

Example:

File: **USER\_SEL.PAR**

Content: **2**

tells the software that at **Alt-J** (or '**Esc**') the Factors included in the **USERFOR2.PAR** file (Output Power Meter) will be modified.

**Important:** Both USER Factors have to be entered at the prompt. If you press <Enter> the USER Factor does not Change. If you press 'Esc' at the prompt the operation is aborted. To review and modify the USER-Defined .PAR files use the PARAMETER FILE utility in the SETTINGS menu of the CCMT Shell.

This routine uses the MS-DOS ascii-editor program EDIT.EXE. If you prefer another editor, then give it the name EDIT.EXE and place it in C:\CCMT and it will be invoked instead of the MS-DOS editor.

## Determination of the Instrument Mnemonics

The CCMT software provides for direct communication from the keyboard with the GPIB connected instruments. This utility uses the Keyboard Control Programm **GPIB.EXE** which is part of the CCMT software.

This program can be invoked from the UTILITIES menu of the CCMT Shell (F3).

If the GPIB card has been properly installed and the GPIB driver loaded in the AUTOEXEC.BAT file, then GPIB.EXE will respond with the prompt **CMD>**

You can enter directly the GPIB commands from the keyboard using the following format (here example address is = 6).

```
CMD> OUTPUT 06;XYZ ABC EFG      <Enter>
```

```
CMD> OUTPUT 06;abc def         <Enter>
```

```
CMD> ENTER 06                  <Enter>
```

→RESPONSE OF INSTRUMENT

You can use the F3 key to recall the last send message in order to send it again or modify parts of it, to avoid retyping the whole string.

XYZ ABC EFG etc are considered to be legal mnemonics commands for the particular instrument as described in the GPIB section of its operation manual.

If there is a problem Error Messages will warn you, such as TIME OUT ERROR etc.

Once you have determined the mnemonics required to Initialize, Configure, Trigger and Read your instrument you exit the GPIB.EXE program by typing EXIT <Enter>.

Then you introduce, using the ASCII-editor program, the commands found in the files USERINix.PAR, USERCOMx.PAR and USERFORx.PAR as described before and you are ready to communicate with your new instrument.