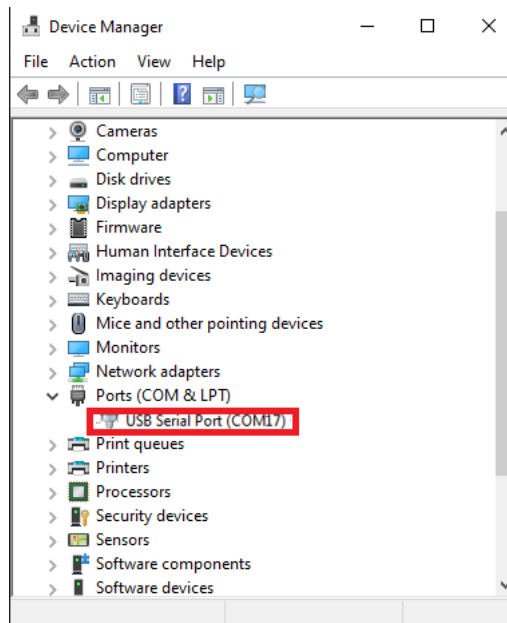# Amtery RF Synthesizer Programming Guide

Content

1. Overview

1.1. Scope

This programming guide is intended for customers to create their own programs to control Amtery RF synthesizers.

1.2. Getting started

After installing Amtery synthesizer driver, plug in an Amtery RF synthesizer into an USB port, Windows device manager will show "USB Serial Port" in Ports (COM & LPT). If "USB Serial Port" doesn't pop up, disable anti-virus software, unplug and plug in again.

1.3. Programming methods

There are three methods to program Amtery RF synthesizer:

A. DLL

B. ASCII command

C. LabVIEW API

1.4. Installed content

DLL, documents and examples are installed in three sub folders of "C:\Program Files (x86)\Amtery\Amtery Synthesizer":
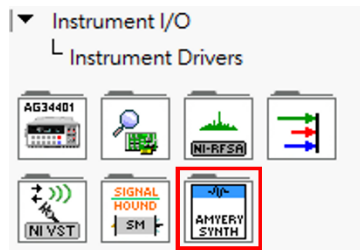
A. A 32-bit (x86) DLL file, amtery_synthesizer.dll, and header file, amtery_synthesizer.h, are in "Bin".

B. Programing Guide are in "Documents".

C. Examples are in "Programming Examples".

    i. DLL examples for C/C++ are in "Programming Examples\Cpp Examples".

    ii. LabVIEW API examples are in "Programming Examples\LabVIEW Examples".

iii.   C/C++ ASCII command examples are in "Programming Examples\ASCII command Cpp Examples".

iv.   LabVIEW ASCII command examples are in "Programming Examples\ASCII command LabVIEW Examples\Examples".

The 64-bit (x64) DLL is in "C:\Program Files (x86)\Amtery\Amtery Synthesizer\Bin\x64". C/C++ API DLL and LabVIEW API DLL examples are written with 32-bit (x86) DLL. If a 64-bit Windows is used, replace the DLL with 64-bit (x64) DLL.

Steps to install LabVIEW driver:

A.   Find folder "C:\Program Files (x86)\Amtery\Amtery Synthesizer\Programming Examples\ASCII command LabVIEW Examples".

B.   Manually copy the folder "ASCII command LabVIEW Examples" to "\LabVIEW\instr.lib". For example, if a user has installed LabVIEW 2023 32-bit, the destination folder is "C:\Program Files (x86)\National Instruments\LabVIEW 2023\instr.lib".

C.   "Amtery Synthesizer" will appear in "LabVIEW instrument I/O -> instrument drivers" palette.

2. DLL

The DLL file amtery_synthesizer.dll contains API to control the synthesizer.

2.1. API List

A) Reset Device

Command Syntax	Description
int amtSynthReset(char* portName)	Reset the device.

Arguments	Description
portName	Virtual COM port number of the device.
return	0: Success. -1: Fail.

B) Query Device Information

Command Syntax	Description
int amtSynthGetDeviceInfo(char* portName, char* deviceInfo)	Read the device information.

Arguments	Description
portName	Virtual COM port number of the device.
deviceInfo	Device information. The return string is "model name" + "firmware version" + "SN". Ex: Amtery SYN0020-USB ver1.3, SN:E0EE15FB1C00
return	0: Success. -1: Fail.

C) Start Continuous Wave

Command Syntax	Description
int amtSynthStartCW(char* portName, int channel, double frequency, double power)	Set the continuous wave parameters and start generation.

Arguments	Description
portName	Virtual COM port number of the device.
channel	Specify the channel number. Input "0" to control all channels.
frequency	Output frequency in MHz.
power	Output power level in dBm.
return	0: Success. -1: Fail.

D) Start Pulse Modulation

Command Syntax	Description
int amtSynthStartPulseMod(char* portName, int channel, int source, double frequency, double power, double pulseWidth, double repetitionRate)	Set the pulse modulation parameters and start generation.

Arguments	Description
portName	Virtual COM port number of the device.
channel	Specify the channel number. Input "0" to control all channels.
source	Source of pulse: 0: Use internal pulse. 1: Use external pulse.
frequency	Output frequency in MHz.
power	Output power in dBm.
pulseWidth	Pulse width in µs. Ignored if source is external.
repetitionRate	Repetition rate in kHz. Ignored if source is external.
return	0: Success. -1: Fail.

E)   Start Frequency Sweep

Command Syntax	Description
int amtSynthStartFreqSweep (char* portName, int channel, double startFrequency, double stopFrequency, double power, double step, int time)	Set frequency sweep parameters and start generation.

Arguments	Description
portName	Virtual COM port number of the device.
channel	Specify the channel number. Input "0" to control all channels.
startFrequency	The start frequency of the sweep in MHz.
stopFrequency	The stop frequency of the sweep in MHz.
power	Output power in dBm.
step	The incremental frequency change in MHz.
time	The dwelling time at each frequency in ms.
return	0: Success. -1: Fail.

F)   Start Power Sweep

Command Syntax	Description
int amtSynthStartPwrSweep (char* portName, int channel, double startPower, double stopPower, double step, int time)	Set power sweep parameters and start generation.

Arguments	Description
portName	Virtual COM port number of the device.
channel	Specify the channel number. Input "0" to control all channels.
frequency	Output frequency in MHz.
startPower	The start power of the sweep in dBm.
stopPower	The stop power of the sweep in dBm.
step	The incremental power change in dB.
time	The dwelling time at each power value in ms.
return	0: Success. -1: Fail.

G)   Get Generation Status

Command Syntax	Description
int amtSynthGetGenStatus(char* portName, int	Get the generation status.

channel)	

Arguments	Description
portName	Virtual COM port number of the device.
channel	Specify the channel number.
return	0: Stopped. 1: Generating.

H) Stop Generation

Command Syntax	Description
int amtSynthStop(char* portName, int channel)	Stop generation.

Arguments	Description
portName	Virtual COM port number of the device.
channel	Specify the channel number. Input "0" to stop all channels.
return	0: Success. -1: Fail.

I) Set Reference Clock

Command Syntax	Description
int amtSynthSetRefClk(char* portName, int source, double frequency)	Set the reference clock.

Arguments	Description
portName	Virtual COM port number of the device.
source	Clock source: 0: Use internal reference clock. 1: Use external reference clock.
frequency	Frequency of reference clock in MHz. Ignored if source is internal.
return	0: Success. -1: Fail.

J) Get Reference Clock

Command Syntax	Description
int amtSynthGetRefClk(char* portName, int source, double* frequency)	Get the reference clock.

Arguments	Description
portName	Virtual COM port number of the device.
source	Clock source: 0: Internal reference clock.

	1: External reference clock.
frequency	Frequency of reference clock in MHz.
return	0: Success.
	-1: Fail.

3. ASCII Commands for RF Synthesizer Control

ASCII commands are case sensitive. ASCII commands can be used in common programing languages or serial port communication tools such as "PuTTY".

3.1. Using ASCII Commands

A terminal character 0x0D "carriage return" is needed at the end of all commands sent. Amtery RF synthesizer ASCII commands are SCPI-like but not standard SCPI, users cannot cascade multiple commands. Every single command needs to be sent separately.

3.2. ASCII Commands List

A terminal character 0x0D "carriage return" is needed at the end of all commands sent.

A) Reset device

Command Syntax	Description	Example
*RST	Reset the device.	*RST

Arguments	Description
n/a	n/a

Return String	Example
"Status" or error information	"0": Success. Negative value: Fail.

B) Query device information

Command Syntax	Description	Example
*IDN?	Read device information.	*IDN?

Arguments	Description
n/a	n/a

Return String	Example
"Model name" + "Firmware version" + "SN".	"Amtery SYN0020-USB ver1.3, SN:E0EE15FB1C00."

C) Start Continuous Wave

Command Syntax	Description	Example
CW <CH> <FREQ> <PWR>	Set the continuous wave parameters and start generation.	CW 1 1000 -10

Arguments	Description
CH	Specify the channel number. Input "0" to control all channels.
FREQ	Output frequency in MHz.

PWR	Output power level in dBm.
Return String	**Example**
"Status" or error information	"0": Success. Negative value: Fail.

D)   Start Pulse Modulation

Command Syntax	Description	Example
PWM <CH> <SOURCE> <FREQ> <PWR> <WIDTH> <RATE>	Set the pulse modulation parameters and start generation.	PWM 1 0 1000 0 50 10

Arguments	Description
CH	Specify the channel number. Input "0" to control all channels.
SOURCE	Source of pulse: 0: Use internal pulse. 1: Use external pulse.
FREQ	Output frequency in MHz.
PWR	Output power in dBm.
WIDTH	Pulse width in µs. Ignored if source is external.
RATE	Repetition rate in kHz. Ignored if source is external.
Return String	**Example**
"Status" or error information.	"0": Success. Negative value: Fail.

E)   Start Frequency Sweep

Command Syntax	Description	Example
SW_FREQ <CH> <STARTFREQ> <STOPFREQ> <PWR> <STEP> <TIME>	Set frequency sweep parameters and start generation.	SW_FREQ 1 100 1000 100 -10 20

Arguments	Description
CH	Specify the channel number. Input "0" to control all channels.
STARTFREQ	The start frequency of the sweep in MHz.
STOPFREQ	The stop frequency of the sweep in MHz.
STEP	The incremental frequency change in MHz.
PWR	Output power in dBm.
TIME	The dwelling time at each frequency in ms.

Return String	Example
"Status" or error information.	"0": Success. Negative value: Fail.

F) Start Power Sweep

Command Syntax	Description	Example
SW_PWR <CH> <FREQ> <STARTPWR> <STOPPWR> <STEP> <TIME>	Set power sweep parameters and start generation.	SW_PWR 1 1000 -5 0 1 20

Arguments	Description
CH	Specify the channel number. Input "0" to control all channels.
FREQ	Output frequency in MHz.
STARTPWR	The start power of the sweep in dBm
STOPPWR	The stop power of the sweep in dBm.
STEP	The incremental power change in dB.
TIME	The dwelling time at each power value in ms.

Return String	Example
"Status" or error information.	"0": Success. Negative value: Fail.

G) Get Generation Status

Command Syntax	Description	Example
RF? <CH>	Get the generation status.	RF? 1

Arguments	Description
CH	Specify the channel number.

Return String	Example
"Status" or error information.	0: Stopped. 1: Generating.

H) Stop Generation

Command Syntax	Description	Example
STOP <CH>	Stop generation.	STOP 1

Arguments	Description
CH	Specify the channel number. Input "0" to control all channels.

Return String	Example

"Status" or error information.	0: Success.
	Negative value: Fail.

I)   Set Reference Clock

Command Syntax	Description	Example
REF <SOURCE> <FREQ>	Set the reference clock.	REF 0

Arguments	Description
SOURCE	Clock source:
	0: Use internal reference clock.
	1: Use external reference clock.
FREQ	Frequency of reference clock in MHz. Ignored if source is internal.

Return String	Example
"Status" or error information.	"0": Success.
	Negative value: Fail.

J)   Get Reference Clock

Command Syntax	Description	Example
REF?	Get the reference clock.	REF?

Arguments	Description
n/a	n/a

Return String	Example
"Clock Source", "Frequency".	"0, 100.0"
Clock source:	
0: Internal reference clock.	
1: External reference clock.	