UART Debugger CLI

Overview

As the name suggests, the UART Debugger CLI is an executable program that has the capability to initiate transactions to Dawnstar's IPs through its UART debug interface. This software can grant access to:

1. Common channel (reset and enable)
2. Configuration channel
3. Signal channel

Each executable is bound to an instance of an IP (ex: instance named impl0 will be binded to an executable named `impl0`). For simplicity sake, from this point on, the executable will be represented as `uad`.

Command Line Interface

The general format of the CLI is as follows.

```
./uad <channel> <--flags>
```

The user needs to specify a channel and some flags that are specific to the chosen channel.

Common Channel

```
./uad com --action <action>
```

Options for the action flag are:

1. reset, resets the IP by asserting then de-asserting the reset signal on the IP
2. enable, enables the IP by asserting the enable signal on the IP
3. disable, disables the IP by de-asserting the enable signal on the IP

Configuration Channel

```
./uad cfg --address <address>
./uad cfg --address <address> --data <data>
```

The software can read or write registers in the IP via its configuration channel. The user needs to specify the register's address to read its content. If the user includes the data flag, then that data will be written to the register. Hexadecimal or decimal encoding is used for the value of address and data.

Signal Channel

```
./uad sig --data <data>
```

The software can drive IPs with the signal channel (ex: FIR Filter IP). The command above will set the signal input with the data set by the user, drive the signal clock, and print out the output signal afterwards. Hexadecimal or decimal encoding is used for the signal input and output.

Examples

We’ll be using the FIR Filter IP to show how this software can be used to reset the IP, access its registers, and drive its input signal.

Common Channel: Resetting the IP

```
./uad com --action reset
```

Configuration Channel: Reading and writing to the CSR

```
./uad cfg --address 0x0
0x00000000
./uad cfg --address 0x00000000 -–data 0xDEADBEEF
0xDEADBEEF
```

Signal Channel: Driving the input signal

```
./uad sig --data 0xC0
0x20
./uad sig --data 0xC0
0x28
./uad sig --data 0xC0
0x30
```