|  |  |  |  |
| --- | --- | --- | --- |
| Name | Code | Direction | Description |
| START | 0x01 | Out | Start sending commands |
| DONE | 0x02 | In | Done with commands |
| KILL | 0x03 | Out | Terminate peripheral |
| Requests | | | |
| PRINT | 0x20 | In | Send string to stderr |
| SETM | 0x21 | In | Set listener for memory address |
| SETP | 0x22 | In | Set listener on port |
| SETPN | 0x23 | In | Set listener on pin |
| CLAIM | 0x24 | In | Protect Memory Address for future writing |
| CLAIMP | 0x2B | In | Claim port |
| CLAIMPN | 0x2C | In | Claim pin |
| WRITEM | 0x25 | In | Write memory address |
| WRITEP | 0x26 | In | Write port |
| WRITEPN | 0x27 | In | Write pin |
| SYNC | 0x28 | In | Start synchronous transaction |
| UNSYNC | 029 | In | End synchronous transaction |
| CLK | 0x2A | In | Get the clock period |
| Notifications | | | |
| DATAM | 0x40 | Out | Memory write notification |
| DATAP | 0x41 | Out | Port write notification |
| DATAPN | 0x42 | Out | Pin write notification |
| Acknowledgements and Errors | | | |
| ACK | 0xA0 | Out | Request acknowledged |
| ACK1 | 0xA1 | Out | Acknowledgement with one data byte |
| ACKN | 0xA2 | Out | Acknowledgement with specified number of bytes |
| BAD | 0x80 | Out | Unknown request code |
| FORMAT | 0x81 | Out | Improperly formatted request |
| RANGE | 0x82 | Out | Requested address/port does not exist |
| TAKEN | 0x83 | Out | The address requested was taken |
| VALUE | 0x84 | Out | The value is invalid for the location |

The first communication between the simulator and a peripheral must be the simulator sending a START command. this command has been sent, the peripheral may send any number of request commands. A request command will always be followed by an acknowledgement. Additional requests should not be made until the acknowledgement is received. After the last request has been made, the DONE command should be sent. This communication will take place before the simulation starts.

After listener requests are made, the simulator can send a notification at any time in the future doing simulation.

After the initial request period, no additional requests may be made, unless a CLAIM command is made during this period. If a CLAIM command was made, then a WRITE or SYNC request can be sent at any time.

If a SYNC request is made, the simulator will respond with the current cycle count, and will send another START command every cycle. When this command is sent, the peripheral can send any number of WRITE commands, or an UNSYNC, followed by a DONE command. If an UNSYNC is sent, normal operation will continue.