



Cheetah Framework Specifications

Protocols

Overview

This document describes the protocols and data formats for external interfacing to the Cheetah Framework.

The Cheetah Framework interacts with a host server for exchange of configuration and status information. It also interacts with markets to receive market events and place orders.

The Cheetah Framework defines internal bus protocols for exchange of this data, and separate interfaces encode/decode this into UDP packets. These UDP packets may be further encoded and transmitted over a number of mediums including IP, TCP, and PCIe.

Protocol Messages

The following protocol formats are described:

- Logger Encoding
- Status message encoding:
 - Algo Module status
 - Router logs
 - Lookup bus logs
- Configuration and CMD encoding
- Market feed and Market order.

UDP Datagram Interface Protocol Specifications

The following describes the UDP datagrams used to interface with the system. Field widths are shown in (bytes) and [bits]. Refer to the individual framework specifications for further details of the datagram field interpretation.

Direction	Source Port	Destination Port	Message Type	Comment
Incoming	Host port	2000	Config/cmd	1 st byte indicates number of config items
Incoming	Host port	6000	Market Feed	1 st byte indicates number of mkt events
Outgoing	2001	Host Port	Status	1 st byte indicates type: x01 = Algo Module Status x02 = Configuration report x04 = HW Signature x05 = LKU audit log x06 = RTR audit log
Outgoing	2002	Host Port	Log	Logger event
Outgoing	5000	(configured)	Market Order	1 st byte indicates msg size

Incoming Datagrams (Host → Cheetah Framework)

CONFIG

Num items (1)	Item 1 (7)	Item 2 (7)	...	Item N (7)
---------------	------------	------------	-----	------------

Config Item

Algo Module ID (1)	Type [2]	Config item code [6]	Arg(1)	Data (4)
--------------------	----------	----------------------	--------	----------

config_type: 01=sys, 10=element, 11=cmd

Algo Module ID: may be 0x3F to indicate ALL

Market Feed

Num ticks (1)	Tick 1 (10)	Tick 2 (10)	...	Tick N (10)
---------------	-------------	-------------	-----	-------------

Tick

Type (1)	Instr Symbol (8)	Price (int) (4)	Price ndp (1)	Volume (int) (4)
----------	------------------	-----------------	---------------	------------------

type: 1=trade, 2=bid, 3=ask

Outgoing Datagrams

The following describes the UDP datagrams used to interface with the system. Field widths are shown in (bytes) and [bits]

LOGGER:

x03 (1)	Num logs (1)	Log 1 (14)	Log 2 (14)	...	Log N (14)
---------	--------------	------------	------------	-----	------------

Log

Src.id (1)	Src.port (1)	Instr id (1)	Tickref (1)	Timestamp (6)	Data (4)
------------	--------------	--------------	-------------	---------------	----------

STATUS:

x01 (1)	Num items (1)	Item 1 (8)	Item 2 (8)	...	Item N (8)
---------	---------------	------------	------------	-----	------------

Status Item:

Algo Mod id (1)	Algo Mod Type (1)	Basis State (1)	Internal State (1)	#events received uint (4)
-----------------	-------------------	-----------------	--------------------	---------------------------

HW Signature:

x04 (1)	HW Signature (8)	Microtick Period (ps) (2)	Maj/Min vers (2)
---------	------------------	---------------------------	------------------

LOOKUP AUDIT LOG:

x05 (1)	Num items (1)	Item 1 (17)	Item 2 (17)	...	Item N (17)
------------	------------------	----------------	----------------	-----	----------------

Lookup audit log Item:

timestamp (6)	Req Mod ID (1)	Instr id (1)	tickref (1)	Lookup op (1)	Rsp Mod ID (1)	Rsp time (1)	Result (1)	Data (4)
------------------	-------------------	-----------------	----------------	------------------	-------------------	-----------------	---------------	-------------

ROUTER AUDIT LOG:

x06 (1)	Num items (1)	Item 1 (20)	Item 2 (20)	...	Item N (20)
------------	------------------	----------------	----------------	-----	----------------

Router audit log Item:

Timestamp (6)	Src Mod ID (1)	Src Port (1)	Takers [32]	Instr id (1)	Tickref (1)	Data (4)	queue time (1)	Del time (1)
------------------	-------------------	--------------------	----------------	-----------------	----------------	-------------	----------------------	-----------------

Takers is a bitmap of the IDs of Algo Module that took the event

queue time is the number of clks that the event spent on the router internal queue

Del time is the number of clks that the Router took to deliver the event.

RAW MARKET ORDER:

Size = 10 (1)	Side = B S (1)	Instr Symbol (8)	Price (fp) (4)	Volume (int) (4)
------------------	-------------------	---------------------	-------------------	---------------------

Price is sp floating point

Volume is 32 bit unsigned int

Configuration Items

Algo Module Types

Type	Value
EL_TYP_ROUTER	x01
EL_TYP_EMA	X10
EL_TYP_MACD	x11
EL_TYP_P_EMA	x12
EL_TYP_TG1	x20
EL_TYP_LOG	x30
EL_TYP_LOGIC_4IP	x40

Type	Value
EL_TYP_ARITH_4IP	x41
EL_TYP_UDP_RAW_MKT	x50
EL_TYP_UDP_MFD_SYM	x52
EL_TYP_HLOC	x60
EL_TYP_ATR	x61
EL_TYP_ADX	x62
EL_TYP_RMO	x70

System Config Values (applies to all Algo Modules)

```
EL_C_SRC_ROUTE      : x"1"; -- xxxx xxxx xxxI xPSS (for src SS, prt P on ip I)
EL_C_CFG_DONE       : x"2"; -- config is done
EL_C_CFG_LKU_TRG    : x"3"; -- xxxx xxxx xxxx xNTT (set LKU trg entry N with target TT)
EL_C_CFG_OP_ENA     : x"4"; -- xxxx xxxx xxxx xxxE (set op enable bitmask)
EL_C_RESET_CNTRS    : x"5"; -- xxxx xxxx xxxx xxxx (reset CB counters)
```

Generic Commands (applies to all Algo Modules)

```
EL_C_RESET          : x"0";
```

EL Configs for SYSTEM CONTROL element

```
-- config
EL_C_LKU_AUDIT_THRESH : x"1"; -- set threshold for sending lku audit logs [data(7 downto0)]
EL_C_RTR_AUDIT_THRESH : x"2"; -- set threshold for sending rtr audit logs [data(7 downto0)]

-- commands
EL_C_STATUS_REQ       : x"1"; -- request status for el id [data(5..0)](all if el id = 0)
EL_C_CONFIG_REQ       : x"2"; -- request config for el id [data(5..0)]
EL_C_HWSIG_REQ        : x"3"; -- request hw signature
EL_C_CLKSYNC_REQ      : x"4"; -- request a clk_ts sync [data is clk reference]
EL_C_LKU_AUDIT_REQ     : x"5"; -- request immediate tx of lku audit data
EL_C_RTR_AUDIT_REQ     : x"6"; -- request immediate tx of rtr audit data
```

EL Configs for EMA

```
-- config
EL_EMA_C_ALPHA        : x"3"; -- set EMA alpha parameter
EL_EMA_C_LEN          : x"4"; -- set EMA length parameter
```

EL Configs for MACD

```
-- config
EL_MACD_C_FS_ALPHA    : x"3"; -- set EMA-Fast alpha parameter
EL_MACD_C_FS_LEN      : x"4"; -- set EMA-Fast length parameter
EL_MACD_C_SL_ALPHA    : x"5"; -- set EMA-Slow alpha parameter
EL_MACD_C_SL_LEN      : x"6"; -- set EMA-Slow length parameter
EL_MACD_C_SG_ALPHA    : x"7"; -- set EMA-Signal alpha parameter
EL_MACD_C_SG_LEN      : x"8"; -- set EMA-Signal length parameter
```

EL Configs for Traffic gen v1

```
-- config
EL_TG1_C_LEN          : x"4"; -- set pattern length
EL_TG1_C_GAP          : x"5"; -- set gap between generation points (#clks)
EL_TG1_C_IV           : x"6"; -- set initial value
EL_TG1_C_P1           : x"7"; -- set parameter p1

-- commands
EL_TG1_C_START        : x"3"; -- start the traffic generator
```

EL Configs for LOGIC_4IP block

```
-- config
EL_L4IP_C_OPS         : x"3"; -- operators
EL_L4IP_C_K1          : x"4"; -- constant K1
EL_L4IP_C_K2          : x"5"; -- constant K2
```

EL Configs for ARITH_4IP block

```
-- config
EL_A4IP_C_OPS      : x"3"; -- operators
EL_A4IP_C_K1       : x"4"; -- constant K1
EL_A4IP_C_K2       : x"5"; -- constant K2
```

EL Configs for HLOC

```
-- config
EL_HLOC_C_PERIOD   : x"3"; -- num ticks in period (int)
EL_HLOC_C_OP_METRIC : x"4"; -- metric to use for data in output events
EL_HLOC_C_OP_THROT : x"5"; -- min # clks between outputs

-- lookup
EL_HLOC_L_CURR_H    : x"00";      -- lookup current high
EL_HLOC_L_CURR_L    : x"01";      -- lookup current low
EL_HLOC_L_CURR_O    : x"02";      -- lookup current open
EL_HLOC_L_CURR_C    : x"03";      -- lookup current close
EL_HLOC_L_PREV_H    : x"04";      -- lookup previous (N) high
EL_HLOC_L_PREV_L    : x"05";      -- lookup previous (N) low
EL_HLOC_L_PREV_O    : x"06";      -- lookup previous (N) open
EL_HLOC_L_PREV_C    : x"07";      -- lookup previous (N) close
EL_HLOC_L_PRVM1_H   : x"08";      -- lookup previous (N-1) high
EL_HLOC_L_PRVM1_L   : x"09";      -- lookup previous (N-1) low
EL_HLOC_L_PRVM1_O   : x"0a";      -- lookup previous (N-1) open
EL_HLOC_L_PRVM1_C   : x"0b";      -- lookup previous (N-1) close
```

EL Configs for ATR

```
-- config
EL_ATR_C_ALPHA     : x"3"; -- set EMA alpha parameter
EL_ATR_C_LEN       : x"4"; -- set EMA length parameter
EL_ATR_C_IP_CN1    : x"5"; -- boolean. bit 0 indicates that evt has Close(N-1) data

-- lookup
EL_ATR_L_ATR       : x"10";      -- lookup current ATR value
```

EL Configs for ADX

```
-- config
EL_ADX_C_PDN_ALPHA : x"3"; -- set EMA alpha parameter for PDN EMA
EL_ADX_C_PDN_LEN   : x"4"; -- set EMA length parameter for PDN EMA
EL_ADX_C_NDN_ALPHA : x"5"; -- set EMA alpha parameter for NDN EMA
EL_ADX_C_NDN_LEN   : x"6"; -- set EMA length parameter for NDN EMA
EL_ADX_C_ADX_ALPHA : x"7"; -- set EMA alpha parameter for ADX EMA
EL_ADX_C_ADX_LEN   : x"8"; -- set EMA length parameter for ADX EMA

-- lookup
EL_ADX_L_ADX       : x"11";      -- lookup current ADX value
```

Configs for RAW MARKET FEED

```
-- config
EL_MDF_C_TRADE     : x"3"; -- trade evts o/p=0 data=V xxxx xxVO (V=1,vol V=0,price O=F,off)
EL_MDF_C_BID       : x"4"; -- trade evts o/p=0 data=V xxxx xxVO (V=1,vol V=0,price O=F,off)
EL_MDF_C_ASK       : x"5"; -- trade evts o/p=0 data=V xxxx xxVO (V=1,vol V=0,price O=F,off)
EL_MDF_C_ISYM_L    : x"6"; -- instrument symbol (left chars)
EL_MDF_C_ISYM_R    : x"7"; -- instrument symbol (right chars) (config L first)
EL_MDF_C_ISYM_ID   : x"8"; -- instrument symbol (ID) (config R first) xxID
EL_MDF_C_UDPPORT   : x"9"; -- UDP port to listen on xxxx PPPP
EL_MDF_C_MKTID     : x"a"; -- ID of this market xxxx xxID
```

```
-- lookup
EL_MDF_L_SYML      : x"12";      -- lookup left 4 chars of symbol
EL_MDF_L_SYMR      : x"13";      -- lookup right 4 chars of symbol
```

EL Configs for Basic Mkt Order

```
-- config
EL_RMO_C_UDPIP      : x"3"; -- set IP Addr of market order receiver
EL_RMO_C_UDPPORT    : x"4"; -- set UDP port of market order receiver
EL_RMO_C_MIN_VOL     : x"5"; -- min order volume (uint32)
EL_RMO_C_MAX_VOL     : x"6"; -- max order volume (uint32)
EL_RMO_C_MAX_POSN    : x"7"; -- max position (units held) (uint32)
```

References

- White Paper – Ultra Low-Latency Arbitrage
- Cheetah Framework Interface Specification
- Cheetah Cub specifications
- Cheetah Block Specifications
- Cheetah Host Specifications
- www.cheetah-solutions.com

Document Control

Date	Version	Change	Comment
19-Aug-2013	V0.2		Pre-Release
28-Feb-2014	V0.3	extended HW Signature format	Pre-Release