HNeT2005 Function Library Quick Reference V1.5
('C') pg1

Execution Control		
Function Prototype	**keywords**	**Description**
ulong CompactCell(ulong **hCellAsy**)	ALL	Eliminates any cortical memory, axons, synaptic connections, and cells that are unused
ulong ExecuteHnet(ulong **flag**, ulong **hCellAsy**)	LEARN / RECALL, ALL	Performs an execution cycle over a cell assembly
ulong ExecuteThread(ulong **flag,** ulong **threadID**)	LEARN / RECALL, ALL	Performs an execution cycle over the specified thread
ulong ExpandCell(ulong **flag**, ulong size, ulong hCell)	AXONAL_OUTPUT / STIMULUS_SYNAPSE / RESPONSE_SYNAPSE	Increases the number of axons, synapses, or cortical memory elements for the specified cell
ulong OptimizeMemory(ulong **flag**, float level, ulong hCell)	THRESHOLD / PERCENT / NUMBER, [OPTIMIZE_BYAXON / OPTIMIZE_NONZERO]	Optimizes cortical memory using a threshold , percent, or count value
ulong RegrowSynapse(ulong hCellSrc, ulong **flag**, ulong order, ulong **class**, ulong hCell)	STIMULUS_SYNAPSE / RESPONSE_SYNAPSE , NO_CONJUGATE / CONJUGATE / SPIN	Regrows synaptic connections based on internal flags set by OptimizeMemory, and SetConversionType

Cell Allocation		
Function Prototype	**keywords**	**Description**
ulong AllocAppendCell(ulong axons, ulong numcell, ulong *hCellPtr)		Allocates a cell that appends the axonal signals from multiple cells
ulong AllocExtractCell(ulong axonXdim, ulong axonYdim, ulong Xcord, ulong Ycord, ulong hCell)		Allocates a cell that extracts axonal signals from the specified source cell
ulong AllocGranuleCell(ulong axons, ulong **order**, ulong **class**, ulong hCell)	ALL, NO_CONJUGATE / CONJUGATE / SPIN	Allocates a cell that performs signal expansion through generation of complex products
ulong AllocPyramidalCell(ulong axons, ulong hCellResp, ulong numcell, ulong *hCellPtr)		Allocates a cell based on the neo-cortical model, used in conjunction with the stellate cell
ulong AllocPurkinjeCell(ulong axons, ulong hCellResp, ulong hCell)		Allocates a cell for storing cortical memory, based on the cerebellar model
ulong AllocReceptorCell(ulong axonXdim, ulong axonYdim)		Allocates a cell for buffering signals read into HNeT
ulong AllocStellateCell(ulong axons, ulong hCell)		Allocates a cell for storing cortical memory, based on the neo-cortical model
ulong AllocTemporalCell(ulong axons, ulong synapses, ulong timesteps, ulong hCell)		Allocates a cell that creates a temporal buffer for stimulus input signals

Cell Assembly Allocation		
Function Prototype	**keywords**	**Description**
ulong EndAssemblyCode()		Marks the end of a code section that allocates a cell assembly
ulong LoadConfig(char *filename)		Loads a cell assembly from a file onto the HNeT cell stack
ulong LoadConfigPtr(char *fileptr)		Loads a cell assembly from a memory pointer onto the HNeT cell stack
ulong SaveConfig(char *filename, ulong **hAssembly**)	ALL	Saves the designated cell assembly to a file
ulong SaveConfigPtr(char *fileptr, ulong **hAssembly**)	ALL	Saves the designated cell assembly to a memory pointer
ulong SizeConfig(ulong **hAssembly**)	ALL	Returns the size in bytes of the HNeT configuration file
ulong StartRecursionCode(ulong index)		Marks the code section that uses recurrent connections

Note : “ulong” defined as 32 bit unsigned interger

HNeT2005 Function Library Quick Reference V1.5
('C') pg2

Setting Cell Properties		
Function Prototype	**keywords**	**Description**
ulong CopyConversion(ulong srcCell, ulong destCell)		Copies conversion attributes from source cell to destination cell
ulong SetCellID(ulong IDnum, ulong hCell)		Assigns the specified user ID tag to the cell
ulong SetConversionType(ulong **flag**, ulong **options**, ulong hCell)	NONE / LINEAR / SIGMOID / HISTOGRAM[1,2] / FOURIER / WAVELET / etc, NORMALIZE / HAMMING to WELCH	Sets the signal conversion method and options (windowing function, normalization) that are applied to a cell
ulong SetExecutionState(ulong **flag**, ulong **hCellAsy**)	ENABLED / DISABLED, ALL	Sets the execution state (on/off) for cells
ulong SetLearnRate(float lrn_rate, ulong hCell)		Sets the learning rate applied to a cortical cell
ulong SetMemoryDecay(float threshold, ulong hCell)		Sets memory decay applied to a cortical cell
ulong SetQuantization(ulong **numcoef**, ulong **flag**, ulong hCell)	ALL, COMPLEX_MODE / REAL_MODE	Sets the degree of quantization applied when FOURIER or WAVELET conversions are applied
ulong SetScalarMode(ulong **flag**, ulong **hCellAsy**)	COMPLEX_MODE / REAL_MODE, ALL	Sets the scalar mode for the cell or assembly
ulong SetUserFunction(ulong index**,** ulong bytecount, void *parmstruct, ulong hCell)		Applies a user defined algorithm/function to the signals stored within the cell's axons
ulong SetWaveletCoef(ulong numcoef, float ***coefptr**, ulong hCell)	DAUB[4 to 20], COIF[8 to 20], SYMM[6 to 30], CUSTOM	Sets the "mother" function for the WAVELET conversion

Reading Cell Properties		
Function Prototype	**keywords**	**Description**
ulong GetAxonCount(ulong *axonXdim, ulong *axonYdim, ulong hCell)		Returns the number of axons for the specified cell
ulong GetAxonReference(ulong axon, ulong hCell)		Returns the number of synaptic connections that have been estblished to the specified axon
ulong GetCellID(ulong **flag**, ulong hCell)	CELL_TYPE / CELL_ID	Returns the cell type or user ID number
ulong GetConversionType(ulong *options, ulong hCell)	see SetConversionType	Returns the conversion method and options that are applied to a cell
ulong GetSynapseCount(ulong **flag**, ulong hCell)	STIMULUS_SYNAPSE RESPONSE_SYNAPSE	Returns the number of synaptic connections for receiving stimulus or conditioning response signals
ulong GetExecutionState(ulong hCell)		Returns the execution state (on/off) of a cell
float GetLearnRate(ulong hCell)		Returns the learning rate applied to a cortical cell
ulong GetMemoryCount(ulong **flag**, ulong hCell)	ALL / ENABLED / DISABLED	Returns the number of cortical memory values stored within a cortical cell
float GetMemoryDecay(ulong hCell)		Returns the memory decay threshold applied to a cortical cell
ulong GetMemoryReference(ulong synapse, ulong hCell)		Returns the number of enabled cortical memory elements that recieve input from the specified synapse
ulong GetQuantization(ulong *listptr, ulong *flag, ulong hCell)		Returns a list that specifies the quantization applied during signal conversion
ulong GetScalarMode(ulong hCell)		Returns the scalar mode for the cell
ulong GetUserFunction(ulong *bytecount, void *parmstruct, ulong hCell)		Returns the assigned user function. This function is applied to signals stored within the cell's axons
ulong GetWaveletCoef(float *coefptr, ulong hCell)		Returns the "mother" function used in the WAVELET conversion

HNeT2005 Function Library Quick Reference V1.5
('C') pg3

Signal Transfer		
Function Prototype	**keywords**	**Description**
ulong ClrSignal(ulong **hCellAsy**)	ALL	Sets the signals stored within a cell's axons to zero
ulong InputRawSignal(CPX *arrayptr, ulong hCell)		Transports signals from a memory pointer into the cell's axons (in real/imaginary format), no conversion is applied
ulong InputSignal(float *arrayptr, ulong hCell)		Transports floating point signals from a memory pointer into the cell's axons, with the assigned conversion applied
ulong InputSignalCpx(CPX *arrayptr, ulong hCell)		Transports CPX (complex) signals from a memory pointer into the cell's axons, with the assigned conversion applied
ulong OutputRawSignal(CPX *arrayptr, ulong hCell)		Transports signals from the cell's axons to a memory pointer (in real/imaginary format), no conversion is applied
ulong OutputSignal(CPX *arrayptr, ulong hCell)		Transports signals from the cell's axons to a memory pointer with conversion applied

Distribution Transfer		
Function Prototype	**keywords**	**Description**
ulong ClrDistribution(ulong **hCellAsy**)	ALL	Clears distribution values (μ, σ) that are applied to the conversion (via InputSignal/Cpx/Rdx or OutputSignal/Rdx)
ulong GetDistribution(DIST *arrayptr, ulong hCell)		Reads distribution values (μ, σ) out from a cell
ulong GetDistributionParm(ulong **flag**, ulong hCell)	STD_COEF / DISTRIBUTION_LAG	Reads parameters used to adjust the computed signal distribution
ulong GetHistogram(float *arrayptr, ulong axon, ulong hCell)		Reads the distribution histogram values out from a cell
ulong LrnDistribution(ulong **flag**, float *arrayptr, ulong hCell)	STD / HISTOGRAM	Updates distribution values (μ, σ) that are applied during signal conversion, using floating point sample signals
ulong LrnDistributionCpx(ulong **flag**, CPX *arrayptr, ulong hCell)	STD / HISTOGRAM	Updates distribution values (μ, σ) that are applied during signal conversion, using CPX (complex) sample signals
ulong SetDistribution(DIST *arrayptr, ulong hCell)		Sets distribution values (μ, σ) applied to signal conversion
ulong SetDistributionParm(ulong **flag,** float coef, ulong hCell)	STD_COEF, DISTRIBUTION_LAG	Sets parameters used to adjust the computed signal distributions
ulong SetHistogram(float *arrayptr, ulong axon, ulong hCell)		Sets the distribution histogram applied to the conversion performed during signal transfer

Cortical Memory Transfer		
Function Prototype	**keywords**	**Description**
ulong ClrMemory(ulong **flag**, ulong **hCellAsy**)	ALL / ENABLED / DISABLED, ALL	Clears cortical memory for the specified cell or cell assembly
ulong GetMemory(CPX *arrayptr, ulong hCell)		Reads cortical memory values (complex) from a cell to a pointer
ulong SetMemory(CPX *arrayptr, ulong hCell)		Sets cortical memory within a cell using values read from a pointer

Interconnecting Cells		
Function Prototype	**keywords**	**Description**
ulong ClrSynapse(ulong **flag**, ulong **index**, ulong hCell)	STIMULUS_SYNAPSE / RESPONSE_SYNAPSE, ALL	Disconnects synaptic connections into the cell pertaining to stimulus or conditioning response signals
ulong EnumGranuleProd(ulong size, ulong order, ulong **class**)	N0_CONJUGATE / CONJUGATE / SPIN	Returns the number of product terms possible for the specified input array size, product order and class
ulong GetGranuleSynapse(ulong axon, SYNAPSE *arrayptr, ulong hCell)		Reads the synaptic connections of the granule cell that feed their signals into the specified axon
ulong GetSynapse(ulong **flag**, ulong **index**, SYNAPSE *arrayptr, ulong hCell)	STIMULUS_SYNAPSE / RESPONSE_SYNAPSE, ALL	Reads synaptic connections of stimulus or conditioning response inputs from cells other than the granule
ulong SetGranuleSynapse(ulong **axon**, ulong *orderptr, SYNAPSE *arrayptr, ulong hCell)	ALL	Sets synaptic connections within a granule cell
ulong SetSynapse(ulong **flag**, ulong **index**, SYNAPSE *arrayptr, ulong hCell)	STIMULUS_SYNAPSE / RESPONSE_SYNAPSE, ALL	Sets synaptic connections of stimulus or conditioning response inputs for cells other than the granule

HNeT2005 Function Library Quick Reference V1.5
('C') pg4

Cell Removal		
Function Prototype	**keywords**	**Description**
ulong InitializeHnet(void)		Initializes HNeT kernel by removing all cells and related structures
ulong RemoveAssembly(ulong **hAssembly**)	ALL	Removes a cell assembly from the HNeT cell stack
ulong RemoveCell(ulong hCell)		Removes a cell from the HNeT cell stack
ulong RemoveThread(ulong threadID)		Removes cells that were allocated by the given process thread
ulong UninitializeHnet(void)		Releases the HNeT kernel and all associated storage from memory

Cell Sequencing	
Function Prototype	**Description**
ulong GetCellCount(void)	Returns the number of cells allocated by the current process
ulong HandleToSequence(ulong hCell)	Returns the execution sequence number for the specified cell
ulong MoveCell(ulong sequence, ulong hCell)	Sets the execution sequence number for the specified cell
ulong SequenceToHandle(ulong sequence)	Returns the cell handle for the specified execution sequence number

Assembly Utility Functions		
Function Prototype	**keywords**	**Description**
ulong EnumAssembly(ulong **index**)	COUNT	Returns the number of cell assemblies allocated by the current process
ulong GetAssemblyBase(ulong hAssembly)		Returns the lowest sequence number within the specified cell assembly
ulong GetAssemblyHandle(ulong hCell)		Returns handle of the cell assembly containing the specified cell
ulong GetAssemblySize(ulong hAssembly)		Returns the number of cells allocated within the specified assembly
ulong GetAssemblyThread(ulong hAssembly)		Returns thread ID# for program code that installed the assembly

Radix Functions		
Function Prototype	**keywords**	**Description**
ulong GetDistributionRdx(float *realMean, float *imagMean, float *Stdev, ulong axon, ulong hCell)		Reads an axon's distribution values (μ, σ) to a program variable
ulong GetMemoryRdx(float *real, float *imag, ulong index, ulong hCell)		Reads a cortical memory element to a program variable
ulong InputRawSignalRdx(float real, float imag, ulong axon, ulong hCell)		Transports a signal from a program variable into the specified axon without conversion
ulong InputSignalRdx(float reph, float imag, ulong axon, ulong hCell)		Transports a signal from a program variable into the specified axon and performs the assigned conversion
ulong LrnDistributionRdx(ulong **flag**, float reph, float imag, ulong axon, ulong hCell)	STD / HISTOGRAM	Updates an axon's distribution values (μ, σ) using a signal read from a program variable
ulong OutputRawSignalRdx(float *real, float *imag, ulong axon, ulong hCell)		Transports a signal from the specified axon to a program variable without conversion
ulong OutputSignalRdx(float *reph, float *imag, ulong axon, ulong hCell)		Transports a signal from the specified axon to a program variable and performs the assinged conversion
ulong SetDistributionRdx(float realMean, float imagMean, float Stdev, ulong axon, ulong hCell)		Sets an axon's distribution values (μ, σ) from a program variable
ulong SetMemoryRdx(float real, float imag, ulong index, ulong hCell)		Sets a cortical memory element from a program variable

Other Functions		
Function Prototype	**keywords**	**Description**
ulong EnumHnetThread(ulong **index**)	COUNT	Returns the number of program threads that have allocated cells
char* GetHnetVersion(void)		Returns a string pointer containing the HNeT version
char* GetMessageString(void)		Returns the error/warning message buffer
void InitializeRegrow(void)		Sets fixed starting point for the pseudo-random number generator used in selection of synaptic connections (RegrowSynapse)
ulong ResetMessageString(void)		Clears the error/warning message buffer
ulong SetMessageLevel(ulong flag)		Sets screening level for HNeT system error and warning messages