HAL is Relatively Simple
(You must be kidding!)

I believe the arrangement of the sections in the documentation makes it feel complex.

I attempt to separate the basic operation of HAL from the more advanced (and confusing) sections of the documentation.
Just joining of Black Boxes
What is a Black Box?

Inputs:
- Food
- Air
- Water

Outputs:
- Life
- Energy
- CO₂
- S***t

Me
Black Box is called a “Component”
Connecting Wires are called “Signals”
Inputs & Outputs called Pins
“Parameters” are Values which need Storing

Constant Values inside a Component which need storing are called “Parameters”

Food | Life
Air  | Energy
Water| CO₂

Me

Age: 67  Male
Mass: 75 Married

Me

S***t
“Functions”

“Functions” are the engine(s) inside a Component, which do all the work.

“Functions” are created inside Components by the original author of the Component.

“Functions” read the input pins, do something with them, and set the output pins accordingly.

“Functions” often have the same name as the component (with an appended number).

There is often only one “Function” inside a Component.
Pin & Signal Names look Complicated
but “Any Name will Do”

Tom
Dick
Harry
TomAndHarryTogether
Toms.Mother.in.law
Toms.Mother.in.law.cat
Toms.Mother.in.law.cat.whisker
LinuxCNC pin names reflect heirarchy

They are just names
Any (non-stupid) Connection Between Pins with Signals is Possible
Components get updated:

Very Regularly
(Real Time Space)

Some Time
Any Time
(User Space)

When you have a moment to spare

Real Time Space

Clock Tic-Toc-Tic-Tok
The List of functions connected to the same clock is called a “thread”:
Revision:

- Black Box ---> Component
- Connecting Wire ---> Signal
- Connection Point ---> Pin
- Value stored inside component --> Parameter
- Engine inside a component -- > Function
- Update timing ---> Realtime / User space
- List of functions conn. to same clock --> Thread
HAL Configuration File

Understanding
Editing
Creating

Ordinary Text file

.filename .hal extension
Load Components

Real Time Space
loadrt <component name> <options>

User Space
loadusr [-W] <component name> <options>
Create the thread(s) with their clock(s)

```shell
loadrt motmod [base_period_nsec=<period>] [servo_period_nsec=<period>]

eg. loadrt motmod base_period_nsec=1000 servo_period_nsec=1000000
```

eg. loadrt motmod base_period_nsec=1000 servo_period_nsec=1000000
Add the Functions inside the components to their respective Threads

addf <function> <thread>

eg. addf encoder.update-counters base-thread
addf encoder.capture-position servo-thread
Join up Pins with Signals

```
net <signal name> <pin name> [<pin name>] etc.....
```

you may add => or <= to show ‘in’ to ‘out’ direction
but HAL does not look at them

net commands can join any number of signals / pins anywhere in HAL file
Set the pins & parameters

```
setp <pin | parameter> <value>
```

eg. `setp parport.0.pin-08-out TRUE`

```
setp siggen.0.amplitude 5
```

Set a Signal

sets <signal-name> <value>

eg. sets mysignal 1
Import Values from .ini File

```ini
[EMCMOT]
EMCMOT = motmod
COMM_TIMEOUT = 1.0
BASE_PERIOD = 100000
SERVO_PERIOD = 1000000

[HAL]
HALFILE = TestMill_hal
HALFILE = custom_hal
POSTGUI_HALFILE = postgui_call_list_hal

[HAL]
HALFILE = TestMill_hal
HALFILE = custom_hal
POSTGUI_HALFILE = postgui_call_list_hal
```
Understanding .hal file - 1

```
# Load the Run-Time Components

loadrt [KINS]KINEMATICS
loadrt sim_parport
loadrt stepgen step_type=0,0,0

# Add functions to the base-thread

addf sim-parport.0.read base-thread
addf stepgen.make-pulses base-thread
addf sim-parport.0.write base-thread
addf sim-parport.0.reset base-thread

# Add functions to the servo-thread

addf stepgen.capture-position servo-thread
addf motion-command-handler servo-thread
addf motion-controller servo-thread
addf stepgen.update-freq servo-thread
```

Understanding .hal file - 1
Understanding .hal file - 2

```
# Spindle stuff

net spindle-cmd-rpm  <= spindle.0.speed-out
net spindle-cmd-rpm-abs <= spindle.0.speed-out-abs
net spindle-cmd-rps   <= spindle.0.speed-out-rps
net spindle-cmd-rps-abs <= spindle.0.speed-out-rps-abs
net spindle-at-speed  => spindle.0.at-speed
net spindle-cw        <= spindle.0.forward

# sim-parport stuff

setp sim-parport.0.reset-time 5000

setp sim-parport.0.pin-01-out-reset 1
setp sim-parport.0.pin-02-out-reset 1
setp sim-parport.0.pin-03-out-reset 1
setp sim-parport.0.pin-04-out-reset 1
setp sim-parport.0.pin-05-out-reset 1
setp sim-parport.0.pin-06-out-reset 1
setp sim-parport.0.pin-07-out-reset 1
setp sim-parport.0.pin-08-out-reset 1
```

Understanding .hal file - 3

```bash
###:
# stepgen stuff X
###:
setp stepgen.0.position-scale [JOINT_0]SCALE
setp stepgen.0.steplen 1
setp stepgen.0.stepspace 0
setp stepgen.0.dirhold 35000
setp stepgen.0.dirsetup 35000
setp stepgen.0.maxaccel [JOINT_0]STEPGEN_MAXACCEL

net xpos joint.0.motor-pos-cmd => stepgen.0.position-cmd
net xposfb joint.0.motor-pos-fb <= stepgen.0.position-fb
net xstep <= stepgen.0.step
net xdir <= stepgen.0.dir
net x-ampenable joint.0.amp-enable-out => stepgen.0.enable

###:
# stepgen stuff Y
###:
setp stepgen.1.position-scale [JOINT_1]SCALE
setp stepgen.1.steplen 1
setp stepgen.1.stepspace 0
setp stepgen.1.dirhold 35000
setp stepgen.1.dirsetup 35000
setp stepgen.1.maxaccel [JOINT_1]STEPGEN_MAXACCEL

net ypos joint.1.motor-pos-cmd => stepgen.1.position-cmd
net yposfb joint.1.motor-pos-fb <= stepgen.1.position-fb
net ystep <= stepgen.1.step
net ydir <= stepgen.1.dir
net y-ampenable joint.1.amp-enable-out => stepgen.1.enable
```
Understanding .hal file - 4

```emacs
# e-stop stuff
*******
net estop-out <= iocontrol.0.user-enable-out
net estop-out => iocontrol.0.emc-enable-in

# manualtoolchange stuff
******************************
loadusr -W hal_manualtoolchange

net tool-change iocontrol.0.tool-change => hal_manualtoolchange.change
net tool-changed iocontrol.0.tool-changed <= hal_manualtoolchange.changed
net prepnum iocontrol.0.tool-prep-number => hal_manualtoolchange.number
net prepared iocontrol.0.tool-prepare => iocontrol.0.tool-prepared
```
Errors in .hal file

Debug file information:

/TestMill.hal:70: Signal name 'joint.0.motor-pos-cmd' must not be the same as a pin. Did you omit the signal name?
Creating .hal File:

Draw a Diagram!
Writing a Component

Next Time!