Burster: This is a cell with two compartments, and acts as an endogenous burster, i.e., a pacemaker, working without any input like the neurons that control your heart muscles.

Set up model

In [1]:

```
%matplotlib inline
#import mpld3
#mpld3.enable_notebook()
from neuron import h
import matplotlib.pyplot as plt

h.load_file('stdrun.hoc')
```

Out[1]: 1.0

In [2]:

```
#Create the soma and axon section and define the default parameters
soma = h.Section(name='soma')
soma.diam = 1e3 #micrometers
soma.L = 31.83098861837907e3 #micrometers
soma.cm = 9e-3 #uF/cm2
soma.Ra = 10000 # ohm-cm
soma.insert('leak')
soma.insert('cat')
soma.insert('cas')
soma.insert('kdr')
soma.insert('kA')
soma.insert('kca')
soma.insert('capool')
soma.el_leak = -50# mV
soma.glbar_leak = 0.000045e-3 # 1/Rm (siemens/cm2)
soma.eca=120 #mV
soma.gcatbar_cat =  0.0552e-3 # (siemens/cm2)
soma.gcasbar_cas= 0.055e-3 # (siemens/cm2)
soma.ek = -80 # (mV)
soma.gkdrbar_kdr = 1.890e-3# (siemens/cm2)
soma.gkAbar_kA = 0.200e-3# (siemens/cm2)
soma.gkcabar_kca = 0.500e-3# (siemens/cm2)
soma.taucas_capool = 303 # (ms)
soma.cainf_capool = 500e-6# (mM)
soma.casi = soma.cainf_capool

axon = h.Section(name='axon')
axon.diam = 1e3 #micrometers
axon.L =31.83098861837907e3 # micrometers
axon.cm = 1.5e-3 #uF/cm2
axon.Ra = 10000 # ohm-cm
axon.insert('leak')
axon.insert('na')
axon.insert('kdr')
axon.glbar_leak= 0.0000018e-3 # 1/Rm (siemens/cm2)
axon.el_leak = -60 # mV
axon.ena = 50# (mV)
axon.gnabar_na = 0.300e-3# (siemens/cm2)
axon.ek =-80# (mV)
axon.gkdrbar_kdr = 0.0525e-3# (siemens/cm2)
```

In [3]:

```
axon.connect(soma(1),0)
h.topology()
```

```
|-|       soma(0-1)
   `|       axon(0-1)
```

Out[3]: 1.0

I inject parameters

In [4]:
```
#Inject current in the middle of the soma
stim = h.IClamp(soma(0.5))
stim.delay = 200.0 #delay in ms
stim.dur = 600.0 #duration in ms
stim.amp = 0 #amplitude in nA
```

Simulation Parameters

In [5]:
```
v_init = h.v_init= -50
tstop = h.tstop = 1000   # how long to run the simulation in ms
h.dt = 0.05 # time step (resolution) of the simulation in ms

# Define vectors for recording variables
#soma membr. potential
s_v_vec = h.Vector();t_vec = h.Vector()
# soma channel currents
s_il_vec = h.Vector(); s_ikd_vec = h.Vector()
icas_vec = h.Vector(); icat_vec = h.Vector()
ia_vec = h.Vector(); ikca_vec = h.Vector()
# Axon membrane potential
a_v_vec = h.Vector(); a_il_vec = h.Vector()
# Axon channel currents
ina_vec = h.Vector(); a_ikd_vec = h.Vector()
# Ca pool concentration
casi_vec = h.Vector()

# record the voltage (_ref_v) and time (_ref_t) into the vectors we just crea
t_vec.record(h._ref_t); s_v_vec.record(soma(0.5)._ref_v)
s_il_vec.record(soma(0.5)._ref_il_leak); s_ikd_vec.record(soma(0.5)._ref_ikd_
icas_vec.record(soma(0.5)._ref_icas_cas); icat_vec.record(soma(0.5)._ref_icat
ia_vec.record(soma(0.5)._ref_ia_kA); ikca_vec.record(soma(0.5)._ref_ikca_kca)
a_v_vec.record(axon(0.5)._ref_v); a_il_vec.record(axon(0.5)._ref_il_leak)
ina_vec.record(axon(0.5)._ref_ina); a_ikd_vec.record(axon(0.5)._ref_ikd_kdr)
casi_vec.record(soma(0.5)._ref_casi)

# run the simulation!
h.run()
```

```
Changed dt
```

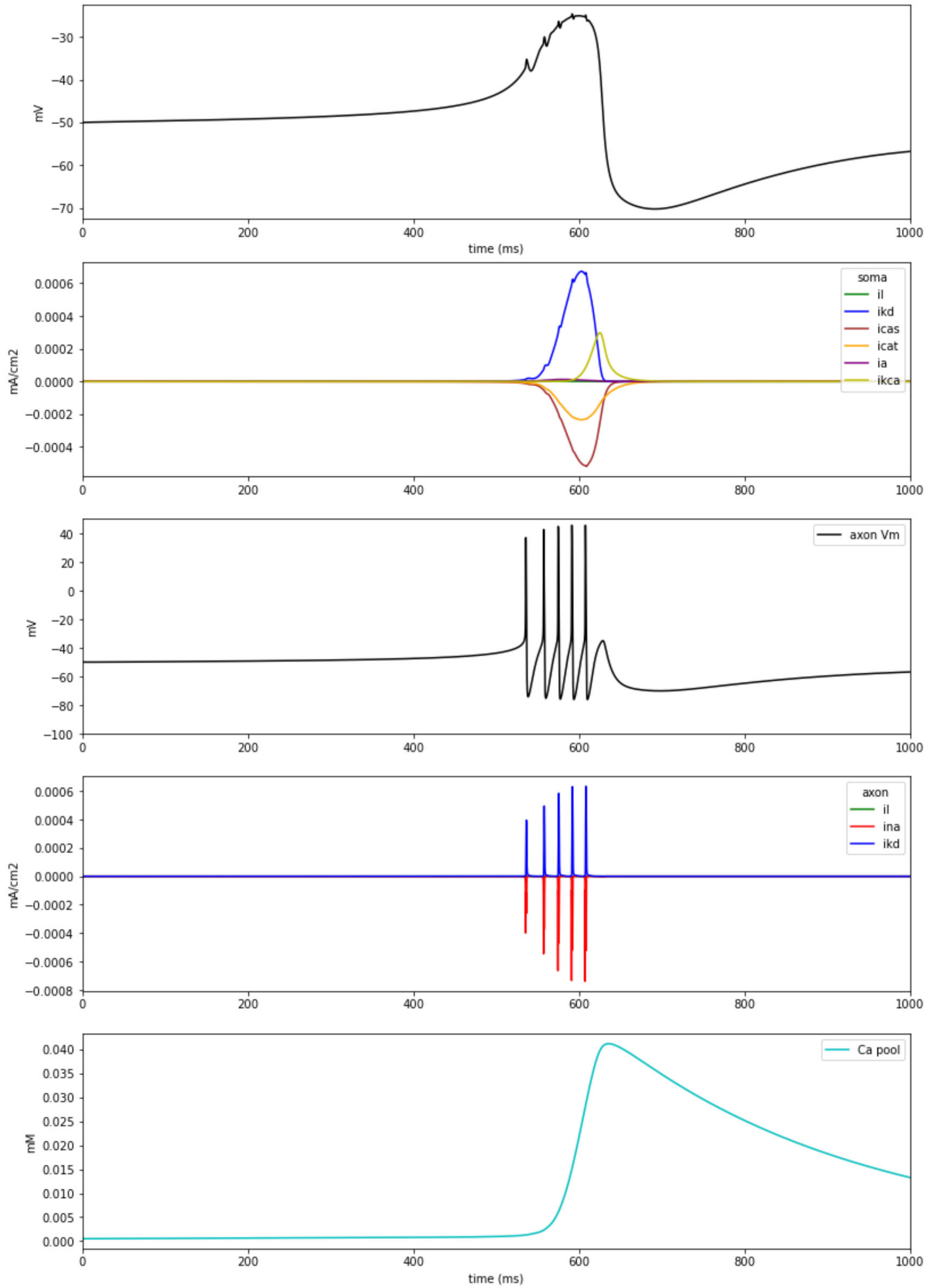
Out[5]:
```
0.0
```

Show Plots

In [6]:

```
plt.figure(figsize=(13,20))
# Soma membrane potential
plt.subplot(5,1,1)
plt.plot(t_vec, s_v_vec,'k')
plt.xlim(0, tstop)
plt.xlabel('time (ms)')
plt.ylabel('mV')
# Soma channel currents
plt.subplot(5,1,2)
plt.plot(t_vec,s_il_vec,'g'); plt.plot(t_vec,s_ikd_vec,'b')
plt.plot(t_vec,icas_vec,'brown'); plt.plot(t_vec,icat_vec,'orange')
plt.plot(t_vec,ia_vec,'purple'); plt.plot(t_vec,ikca_vec,'y')
plt.xlim(0,h.tstop)
plt.ylabel('mA/cm2'); plt.legend(['il','ikd','icas','icat','ia','ikca'],title
# Axon membrane potential
plt.subplot(5,1,3)
plt.plot(t_vec,a_v_vec ,'k')
plt.xlim(0,h.tstop); plt.ylim(-100,50)
plt.ylabel('mV'); plt.legend(['axon Vm'],loc=1)
# Axon channel currents
plt.subplot(5,1,4)
plt.plot(t_vec,a_il_vec ,'g')
plt.plot(t_vec,ina_vec ,'r')
plt.plot(t_vec,a_ikd_vec ,'b')
plt.xlim(0,h.tstop);
plt.ylabel('mA/cm2'); plt.legend(['il','ina','ikd'],title='axon',loc=1)
# Ca pool concentration
plt.subplot(5,1,5)
plt.plot(t_vec,casi_vec,'c')
plt.xlim(0,h.tstop); plt.xlabel('time (ms)')
plt.ylabel('mM'); plt.legend(['Ca pool'],loc=1)

plt.show()
```


mV
time (ms)
soma
il
ikd
icas
icat
ia
ikca
mA/cm2
axon Vm
mV
axon
il
ina
ikd
mA/cm2
Ca pool
mM
time (ms)