Functional Level Comparision

This notebook computes the opus and weighted opus score for the following:

1. Electric Hook Type TD
    - vs. all types of Electric Hand
    - vs. Electric single grip hand
    - vs. body-powered TD
    - vs. Electric multi-articulating hand
2. Motion Control ETD
    - vs. electric hooks
3. Multi-articulating hands
    - vs. Electric single grip hand
    - vs. Body-powered TD

In [1]:
```
import pandas as pd
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt
from driverCleanData import cleanData
%matplotlib inline
[sht,_] = cleanData();
```

```
There are 0 duplicates
```

Electric Hook TD

vs. all types of Electric Hand

In [2]:
```
from vectorSubsets import electricHooks, electricHandMA, electricHandSG
print("Electric hooks are: {0}".format(electricHooks))
print()
print("Electric hands - multi articulating are {0}".format(electricHandMA))
print()
print("Electric hands - single grip are {0}".format(electricHandSG))
```

```
Electric hooks are: ['Motion Control (MC) ETD', 'Otto Bock\xa0Greifer', 'Otto Bock Axon
 Hook']

Electric hands - multi articulating are ['bebionic Hand', 'i-limb Hand', 'Michelangelo
 Hand']

Electric hands - single grip are ['Motion Control (MC) Hand', 'Otto Bock Hand']
```

In [3]:
```
sht["opusEHo"] = sht.opusScore[sht["PrimaryTerminalDevice"].isin(electricHooks)]
sht["opusAllEHa"] = sht.opusScore[sht["PrimaryTerminalDevice"].isin(electricHandMA or el
ectricHandSG)].dropna()
```

Summary Stats

First up the electric hooks

In [4]:
```
sht.opusEHo.dropna().describe()
```

Out[4]:
```
count    125.000000
mean      57.856000
std        9.165771
min       33.000000
25%       52.000000
50%       55.000000
75%       63.000000
max       80.000000
Name: opusEHo, dtype: float64
```

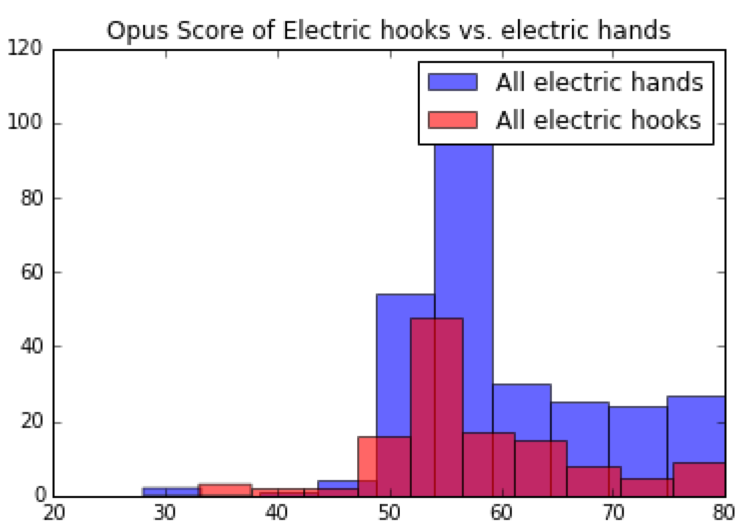
Now for the electric hands

In [5]:
```
sht.opusAllEHa.dropna().describe()
```

Out[5]:
```
count    272.000000
mean      60.095588
std        9.065061
min       28.000000
25%       54.000000
50%       57.000000
75%       66.000000
max       80.000000
Name: opusAllEHa, dtype: float64
```

Histogram

In [6]:
```
plt.hist(sht.opusAllEHa.dropna(), alpha=0.6, color='b', label = "All electric hands")
plt.hist(sht.opusEHo.dropna(), alpha=0.6, color='r', label = "All electric hooks")
plt.title("Opus Score of Electric hooks vs. electric hands")
plt.legend()
plt.show()
```



Looks like we might have some significance, however the limited sample of electric hooks will limit our conclusions.

In [7]:
```
stats.ttest_ind(sht.opusAllEHa.dropna(), sht.opusEHo.dropna())
```

Out[7]:
```
Ttest_indResult(statistic=2.2783682103618665, pvalue=0.023237915723183421)
```

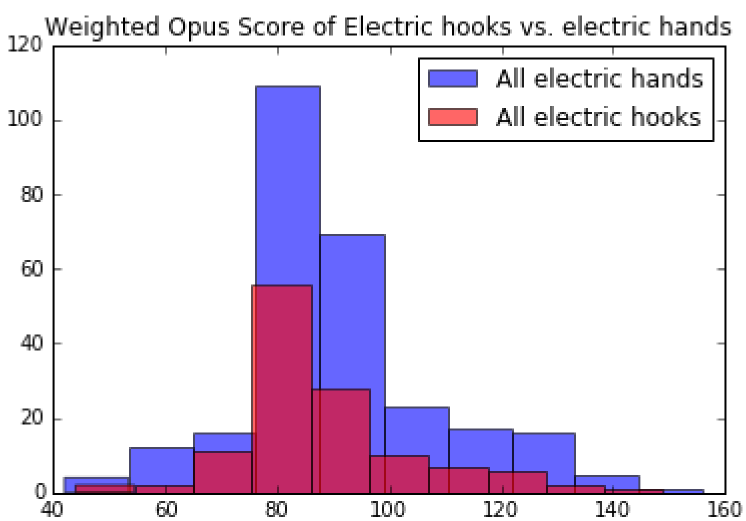
The higher mean score on Opus of electric hooks compared to electric hands is significant.

Weighted Opus

In [8]:
```
sht["opusEHoWeighted"] = sht.weightedOpusScore[sht["PrimaryTerminalDevice"].isin(electri
cHooks)]
sht["opusAllEHaWeighted"] = sht.weightedOpusScore[sht["PrimaryTerminalDevice"].isin(elec
tricHandMA or electricHandSG)].dropna()
```

Histogram

In [9]:
```
plt.hist(sht.opusAllEHaWeighted.dropna(), alpha=0.6, color='b', label = "All electric ha
nds")
plt.hist(sht.opusEHoWeighted.dropna(), alpha=0.6, color='r', label = "All electric hook
s")
plt.title("Weighted Opus Score of Electric hooks vs. electric hands")
plt.legend()
plt.show()
```



Looks good, let's take a look at summary stats

In [10]:
```
print(sht.opusAllEHaWeighted.dropna().describe())
print()
print(sht.opusEHoWeighted.dropna().describe())
```

```
count    272.000000
mean      90.926471
std       17.718643
min       42.000000
25%       82.000000
50%       87.000000
75%       97.000000
max      156.000000
Name: opusAllEHaWeighted, dtype: float64

count    125.000000
mean      87.760000
std       16.280505
min       44.000000
25%       79.000000
50%       83.000000
75%       93.000000
max      149.000000
Name: opusEHoWeighted, dtype: float64
```

In [11]:
```
stats.ttest_ind(sht.opusAllEHaWeighted.dropna(), sht.opusEHoWeighted.dropna())
```

Out[11]:
```
Ttest_indResult(statistic=1.6957981432595295, pvalue=0.090712077279887657)
```

We don't have significance at the 0.05 level

vs. Electric single grip hand

In [44]:
```
sht["opusElecSG"] = sht.opusScore[sht["PrimaryTerminalDevice"].isin(electricHandSG)]
print("The types of devices we are considering as electric hands are {0}".format(electri
cHandSG))
print()
print("The mean of the electric hooks opus is {0}".format(sht.opusEHo.dropna().mean()))
print()
print("The summary stats for the single grip are")
print(sht.opusElecSG.dropna().describe())

# create weighted for later use
sht["opusElecSGWeighted"] = sht.weightedOpusScore[sht["PrimaryTerminalDevice"].isin(elec
tricHandSG)]
```

```
The types of devices we are considering as electric hands are ['Motion Control (MC) Han
d', 'Otto Bock Hand']

The mean of the electric hooks opus is 57.856

The summary stats for the single grip are
count    600.000000
mean      60.125000
std        8.600348
min       33.000000
25%       53.000000
50%       58.500000
75%       67.000000
max       80.000000
Name: opusElecSG, dtype: float64
```

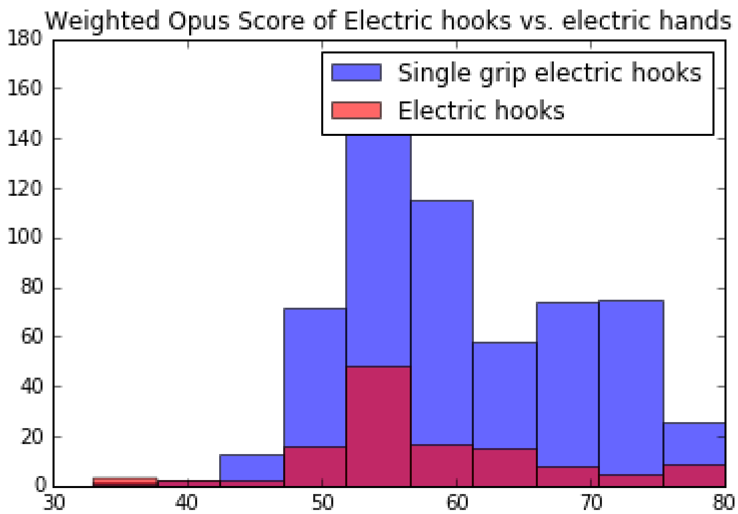
In [13]:
```
stats.ttest_ind(sht.opusEHo.dropna(), sht.opusElecSG.dropna())
```

Out[13]:
```
Ttest_indResult(statistic=-2.6526514753560089, pvalue=0.0081614428637592634)
```

The higher mean score for electric hooks compared to single grip is signficant.

Histograms

In [14]:
```
plt.hist(sht.opusElecSG.dropna(), color='b', alpha=0.6, label = "Single grip electric ho
oks")
plt.hist(sht.opusEHo.dropna(), color='r', alpha=0.6, label = "Electric hooks")
plt.title("Weighted Opus Score of Electric hooks vs. electric hands")
plt.legend()
plt.show()
```



vs. Body Powered Hooks

In [15]:
```
from vectorSubsets import bodyPoweredHooks
sht["opusBPHook"] = sht.opusScore[sht["PrimaryTerminalDevice"].isin(bodyPoweredHooks)]
sht.opusBPHook.dropna().describe()
```

Out[15]:
```
count    181.000000
mean      59.751381
std        9.112815
min       20.000000
25%       54.000000
50%       57.000000
75%       66.000000
max       80.000000
Name: opusBPHook, dtype: float64
```

In [16]:
```
stats.ttest_ind(sht.opusEHo.dropna(), sht.opusBPHook.dropna())
```

Out[16]:
```
Ttest_indResult(statistic=-1.7842168094090507, pvalue=0.075385702215169312)
```

Can't reject the null at alpha=0.05

Weighted Opus

In [45]:
```
sht["opusBPHookWeighted"] = sht.weightedOpusScore[sht["PrimaryTerminalDevice"].isin(body
PoweredHooks)]
print("The mean of w.o. for body poweres is
{0}".format(sht.opusBPHookWeighted.dropna().mean()))
print("The mean for w.o. for electric hooks is
{0}".format(sht.opusEHoWeighted.dropna().mean()))
print()
print(stats.ttest_ind(sht.opusBPHookWeighted.dropna(), sht.opusEHoWeighted.dropna()))
```

```
The mean of w.o. for body poweres is 89.40883977900552
The mean for w.o. for electric hooks is 87.76

Ttest_indResult(statistic=0.82694366104746109, pvalue=0.40891759933790106)
```

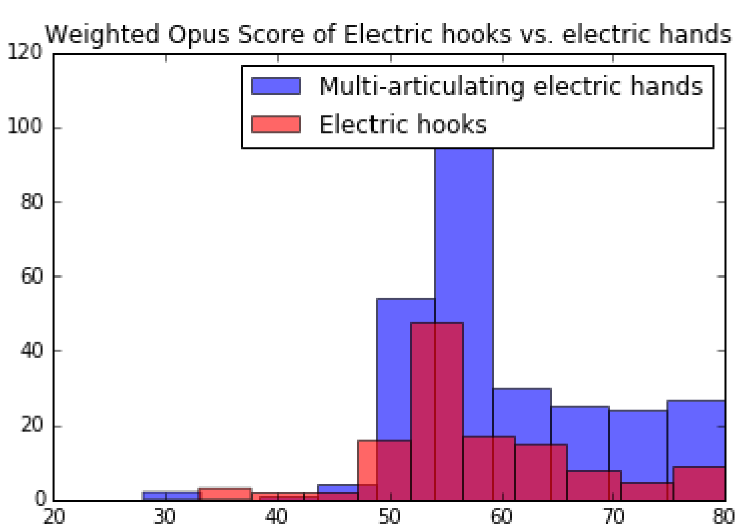
Very simiar to standard opus

vs. Multi-articulating Hands

In [46]:
```
from vectorSubsets import electricHandMA
sht["opusMA"] = sht.opusScore[sht["PrimaryTerminalDevice"].isin(electricHandMA)]
print("The summary stats for multi-articulating are:")
print(sht.opusMA.dropna().describe())
```

```
The summary stats for multi-articulating are:
count    272.000000
mean      60.095588
std        9.065061
min       28.000000
25%       54.000000
50%       57.000000
75%       66.000000
max       80.000000
Name: opusMA, dtype: float64
```

In [19]:
```
plt.hist(sht.opusMA.dropna(), alpha=0.6, color='b', label = "Multi-articulating electric
 hands")
plt.hist(sht.opusEHo.dropna(), alpha=0.6, color='r', label = "Electric hooks")
plt.title("Weighted Opus Score of Electric hooks vs. electric hands")
plt.legend()
plt.show()
```



In [20]:
```
stats.ttest_ind(sht.opusEHo.dropna(),sht.opusMA.dropna())
```

Out[20]:
```
Ttest_indResult(statistic=-2.2783682103618665, pvalue=0.023237915723183421)
```

The higher mean score of the multi-articulating is significant compared to electric hooks.

Weighted Opus

In [21]:
```
sht["opusMAWeighted"] = sht.weightedOpusScore[sht["PrimaryTerminalDevice"].isin(electric
HandMA)]
print("The average on the w.o. for the electric hooks is
{0}".format(sht.opusEHoWeighted.dropna().mean()))
print("The average on the w.o. for the multi-articulating hands is {0}".format(sht.opusM
AWeighted.dropna().mean()))
print()
print(stats.ttest_ind(sht.opusMAWeighted.dropna(), sht.opusEHoWeighted.dropna()))
```

```
The average on the w.o. for the electric hooks is 87.76
The average on the w.o. for the multi-articulating hands is 90.92647058823529

Ttest_indResult(statistic=1.6957981432595295, pvalue=0.090712077279887657)
```

Not significant

Motion Control ETD vs. All Electric Hooks

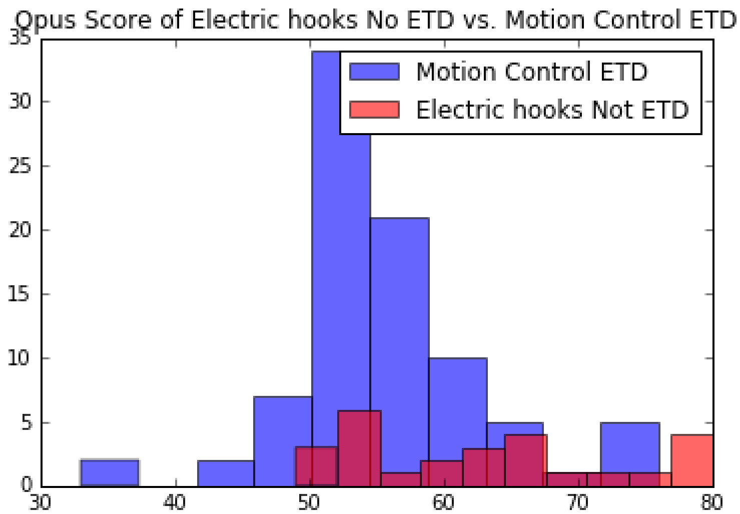
In [22]:
```
sht["opusETD"] = sht.opusScore[sht["PrimaryTerminalDevice"]=="Motion Control (MC) ETD"]
from vectorSubsets import electricHooksNoETD
sht["opusElecNoETD"] = sht.opusScore[sht["PrimaryTerminalDevice"].isin(electricHooksNoET
D)]
print("The electric hooks we are considering are: {0}".format(electricHooksNoETD))
print("The number of participants are {0}".format(sht.opusElecNoETD.dropna().count()))
print()
print("The average score on opus for the ETD is
{0}".format(sht.opusETD.dropna().mean()))
print("The average score on opus for the electric hooks is
{0}".format(sht.opusElecNoETD.dropna().mean()))
print()
print(stats.ttest_ind(sht.opusETD.dropna(), sht.opusElecNoETD.dropna()))
```

```
The electric hooks we are considering are: ['Otto Bock Greifer', 'Otto Bock Axon Hook']
The number of participants are 26

The average score on opus for the ETD is 55.5632183908046
The average score on opus for the electric hooks is 62.92307692307692

Ttest_indResult(statistic=-4.1105410376607354, pvalue=7.5894527963986009e-05)
```

This result is significant and unfortunately unfavorable for the ETD. this could change though if the sample for non etd electric hook increased.

Let's look at the histogram

In [23]:
```
plt.hist(sht.opusETD.dropna(), alpha=0.6, color='b', label = "Motion Control ETD")
plt.hist(sht.opusElecNoETD.dropna(), alpha=0.6, color='r', label = "Electric hooks Not E
TD")
plt.title("Opus Score of Electric hooks No ETD vs. Motion Control ETD")
plt.legend()
plt.show()
```



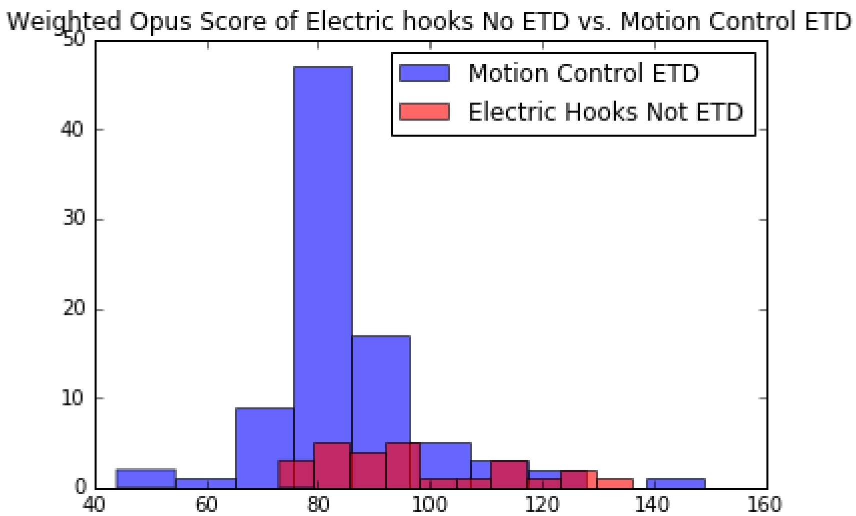
Weighted Opus

In [24]:
```
sht["opusETDWeighted"] = sht.weightedOpusScore[sht["PrimaryTerminalDevice"]=="Motion Con
trol (MC) ETD"]
sht["opusElecNoETDWeighted"] = sht.weightedOpusScore[sht["PrimaryTerminalDevice"].isin(e
lectricHooksNoETD)]
print("The electric hooks we are considering are: {0}".format(electricHooksNoETD))
print("The number of participants are {0}".format(sht.opusElecNoETDWeighted.dropna().cou
nt()))
print()
print("The average score on w.o. for the ETD is
{0}".format(sht.opusETDWeighted.dropna().mean()))
print("The average score on w.o. for the electric hooks is {0}".format(sht.opusElecNoETD
Weighted.dropna().mean()))
print()
print(stats.ttest_ind(sht.opusETDWeighted.dropna(), sht.opusElecNoETDWeighted.dropna()))
```

```
The electric hooks we are considering are: ['Otto Bock Greifer', 'Otto Bock Axon Hook']
The number of participants are 26

The average score on w.o. for the ETD is 83.9080459770115
The average score on w.o. for the electric hooks is 97.26923076923077

Ttest_indResult(statistic=-4.0056075954782493, pvalue=0.00011219423506405462)
```

Again, we have significance at alpha=0.05, an unfortunate result for the ETD.

In [48]:
```
plt.hist(sht.opusETDWeighted.dropna(), alpha=0.6, color='b', label = "Motion Control ET
D")
plt.hist(sht.opusElecNoETDWeighted.dropna(), alpha=0.6, color='r', label = "Electric Hoo
ks Not ETD")
plt.title("Weighted Opus Score of Electric hooks No ETD vs. Motion Control ETD")
plt.legend()
plt.show()
```



Multi-articulating hands

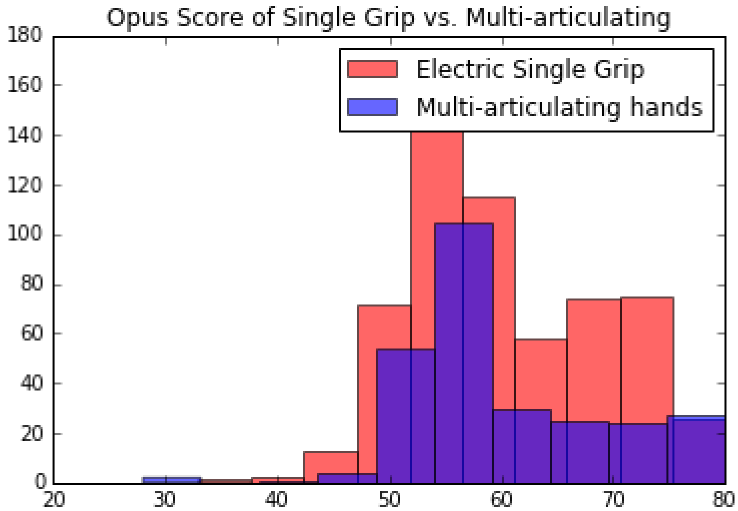
vs. Single Grip hands

In [49]:
```
print("The hands in the multi-articulating group are {0}".format(electricHandMA))
print()
print("The hands in the single grip group are {0}".format(electricHandSG))
print()
print("The average score on opus for the multi-articulating is {0}".format(sht.opusMA.dr
opna().mean()))
print("The average score on opus for the single grip is {0}".format(sht.opusElecSG.dropn
a().mean()))
print()
print(stats.ttest_ind(sht.opusMA.dropna(), sht.opusElecSG.dropna()))
```

```
The hands in the multi-articulating group are ['bebionic Hand', 'i-limb Hand', 'Michela
ngelo Hand']

The hands in the single grip group are ['Motion Control (MC) Hand', 'Otto Bock Hand']

The average score on opus for the multi-articulating is 60.095588235294116
The average score on opus for the single grip is 60.125

Ttest_indResult(statistic=-0.045996681584298454, pvalue=0.96332345043036749)
```

The higher score on the single grip hands is not significant.

In [27]:
```
plt.hist(sht.opusElecSG.dropna(), alpha=0.6, color='r', label = "Electric Single Grip")
plt.hist(sht.opusMA.dropna(), alpha=0.6, color='b', label = "Multi-articulating hands")
plt.title("Opus Score of Single Grip vs. Multi-articulating")
plt.legend()
plt.show()
```



Weighted Opus

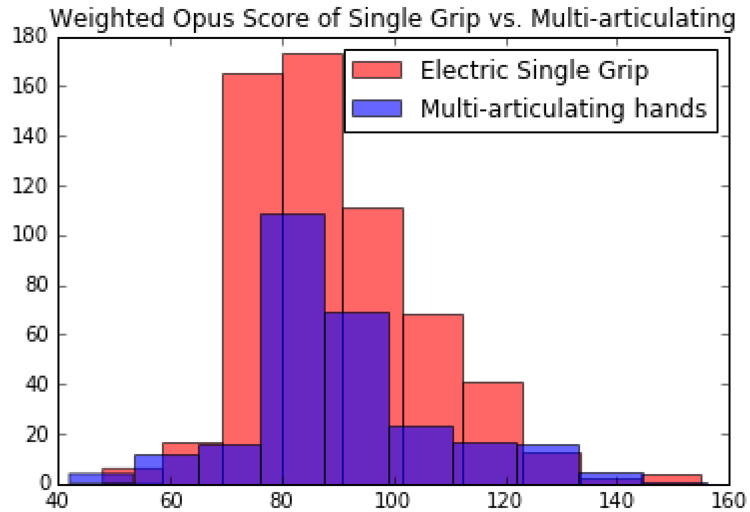
In [28]:
```
print("The average score on opus for the multi-articulating is {0}".format(sht.opusMAWei
ghted.dropna().mean()))
print("The average score on opus for the single grip is {0}".format(sht.opusElecSGWeight
ed.dropna().mean()))
print()
print(stats.ttest_ind(sht.opusMA.dropna(), sht.opusElecSGWeighted.dropna()))
```

```
The average score on opus for the multi-articulating is 90.92647058823529
The average score on opus for the single grip is 89.795

Ttest_indResult(statistic=-28.853133340570007, pvalue=5.6819650830696448e-129)
```

The high score for single grip is definetly significant! Interesting the standard opus is not significant, yet the wieghted is.

Let's look at the histogram

In [29]:
```
plt.hist(sht.opusElecSGWeighted.dropna(), alpha=0.6, color='r', label = "Electric Single Grip")
plt.hist(sht.opusMAWeighted.dropna(), alpha=0.6, color='b', label = "Multi-articulating hands")
plt.title("Weighted Opus Score of Single Grip vs. Multi-articulating")
plt.legend()
plt.show()
```



Let's look at what the single grip users value vs. multi-articulating

In [51]:
```
from vectorSubsets import importanceFull
sgSht = sht[sht.PrimaryTerminalDevice.isin(electricHandSG)]
print("The single grips users value:")
print(sgSht[importanceFull].mean().sort_values(ascending=False).head())
print()
maSht = sht[sht.PrimaryTerminalDevice.isin(electricHandMA)]
print("The multi-articulating users value:")
print(maSht[importanceFull].mean().sort_values(ascending=False).head())
```

```
The single grips users value:
ImportanceUsingKeyboard                    1.681667
ImportanceUsingMobile                      1.645000
ImportanceButtoningButtons                 1.625000
ImportanceStirring                         1.621667
ImportancePuttingToothpasteToothbrush      1.573333
dtype: float64

The multi-articulating users value:
ImportanceBrushingTeeth                    1.735294
ImportancePuttingToothpasteToothbrush      1.702206
ImportanceUsingKeyboard                    1.694853
ImportanceUsingMobile                      1.691176
ImportanceBrushingHair                     1.639706
dtype: float64
```

They have similar values, they just rank them differently. This idea could turn out to be what we use to help us identify a key ETD demographic

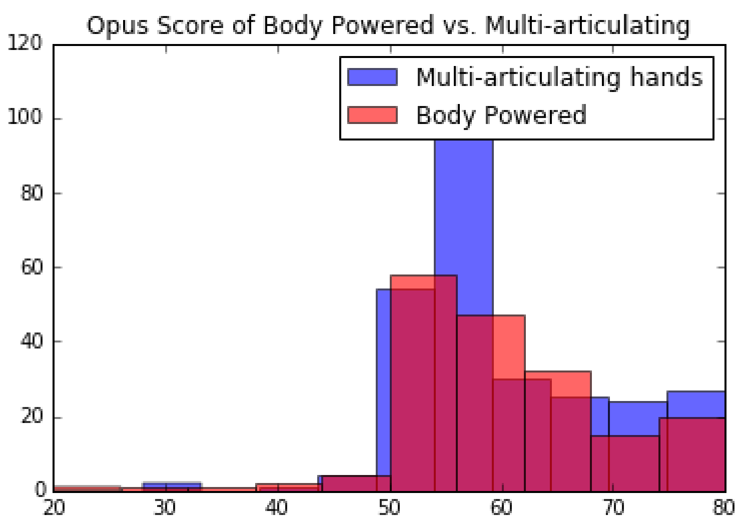
vs. Body Powered

In [30]:
```
print("The hands in the multi-articulating group are {0}".format(electricHandMA))
print()
print("The hands in the body powered group are {0}".format(bodyPoweredHooks))
print()
print("The average score on opus for the multi-articulating is {0}".format(sht.opusMA.dr
opna().mean()))
print("The average score on opus for body powered is
{0}".format(sht.opusBPHook.dropna().mean()))
print()
print(stats.ttest_ind(sht.opusMA.dropna(), sht.opusBPHook.dropna()))
```

```
The hands in the multi-articulating group are ['bebionic Hand', 'i-limb Hand', 'Michela
ngelo Hand']

The hands in the body powered group are ['TRS Hook', 'Hosmer Hook', 'Body-powered Han
d']

The average score on opus for the multi-articulating is 60.095588235294116
The average score on opus for body powered is 59.751381215469614

Ttest_indResult(statistic=0.39501177719799657, pvalue=0.69302094817310467)
```

The higher score on opus by the body powered hooks is not significant.

In [32]:
```
plt.hist(sht.opusMA.dropna(), alpha=0.6, color='b', label = "Multi-articulating hands")
plt.hist(sht.opusBPHook.dropna(), alpha=0.6, color='r', label = "Body Powered")
plt.title("Opus Score of Body Powered vs. Multi-articulating")
plt.legend()
plt.show()
```

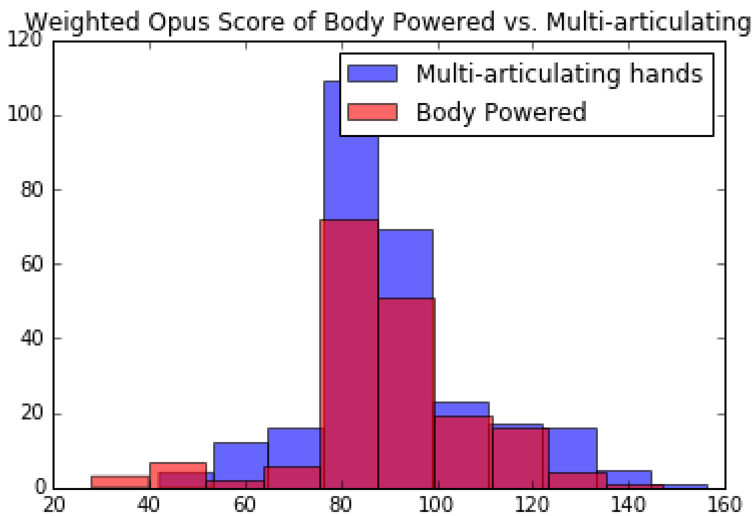


Weighted Opus

In [52]:
```
print("The average score on w.o. for the multi-articulating is {0}".format(sht.opusMAWei
ghted.dropna().mean()))
print("The average score on w.o. for the body powered is {0}".format(sht.opusElecSGWeigh
ted.dropna().mean()))
print()
print(stats.ttest_ind(sht.opusMAWeighted.dropna(), sht.opusElecSGWeighted.dropna()))
print("Not significant")
```

```
The average score on w.o. for the multi-articulating is 90.92647058823529
The average score on w.o. for the body powered is 89.795

Ttest_indResult(statistic=0.94116905248244731, pvalue=0.34687948287985171)
Not significant
```

In [34]:
```
plt.hist(sht.opusMAWeighted.dropna(), alpha=0.6, color='b', label = "Multi-articulating
 hands")
plt.hist(sht.opusBPHookWeighted.dropna(), alpha=0.6, color='r', label = "Body Powered")
plt.title("Weighted Opus Score of Body Powered vs. Multi-articulating")
plt.legend()
plt.show()
```



Check t-test analysis

The results from t-tests can sometimes be too positive, anova and Tukey's test accomadate for multiple comparisons. So the below should be a validation of what was observed above.

ANOVA

Use anova to make sure there is a difference in the means of at least one of the opus scores between the body powered, multi-articulating, single grip, and electric hooks

In [35]:
```
print("The body powered hooks are {0}".format(bodyPoweredHooks))
print()
print("The multi-articulating hands are {0}".format(electricHandMA))
print()
print("The single grip hands are {0}".format(electricHandSG))
print()
print("The electric hooks are {0}".format(electricHooks))
```

```
The body powered hooks are ['TRS Hook', 'Hosmer Hook', 'Body-powered Hand']

The multi-articulating hands are ['bebionic Hand', 'i-limb Hand', 'Michelangelo Hand']

The single grip hands are ['Motion Control (MC) Hand', 'Otto Bock Hand']

The electric hooks are ['Motion Control (MC) ETD', 'Otto Bock\xa0Greifer', 'Otto Bock A
xon Hook']
```

Now we'll run anova to check that there is a difference among the mean scores on opus of these 4 groups.

In [36]:
```
stats.f_oneway(sht.opusBPHook.dropna(), sht.opusMA.dropna(), sht.opusElecSG.dropna(), sh
t.opusEHo.dropna())
```

Out[36]:
```
F_onewayResult(statistic=2.3813920817159544, pvalue=0.06800169658072347)
```

We have don't have signficance at 0.05. But we do at 0.1. Let's look at Turkeys test, find out where.

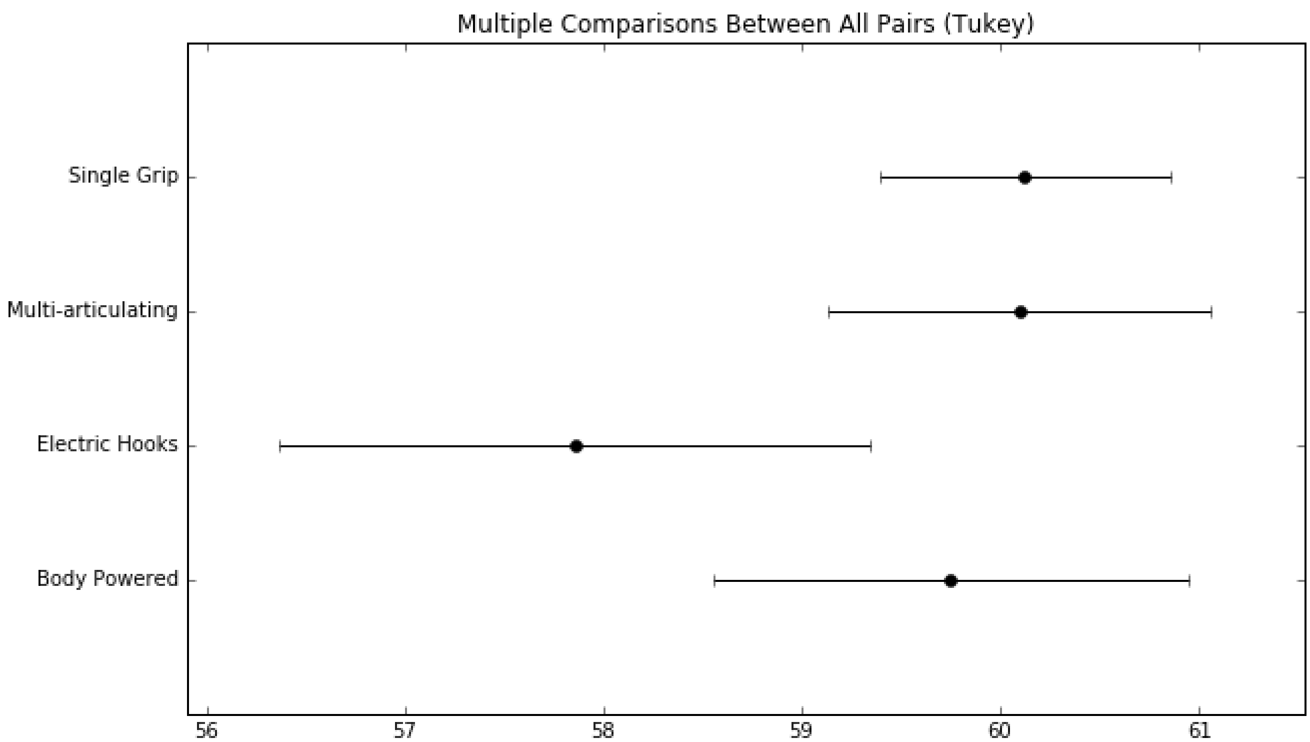
In [54]:
```
opusDf = pd.concat([sht['PrimaryTerminalDevice'], sht['opusScore']], axis=1, keys=['devi
ceCategory','opusScore'])
opusDf["deviceCategory"] = opusDf["deviceCategory"].replace([bodyPoweredHooks],"Body Pow
ered")
opusDf["deviceCategory"] = opusDf["deviceCategory"].replace([electricHandMA], "Multi-art
iculating")
opusDf["deviceCategory"] = opusDf["deviceCategory"].replace([electricHandSG], "Single Gr
ip")
opusDf["deviceCategory"] = opusDf["deviceCategory"].replace([electricHooks], "Electric H
ooks")
opusDf["deviceCategory"] = opusDf["deviceCategory"].replace('Otto.*', "Electric Hooks",
regex=True)
opusDf["deviceCategory"] = opusDf["deviceCategory"].replace('Other.*', np.nan, regex=Tru
e)
opusDf["deviceCategory"] = opusDf["deviceCategory"].replace('Pass.*', np.nan,
regex=True)
#groups = opusDf.groupby(opusDf['deviceCategory']).groups
#opusDf['deviceCategory'] = str(opusDf['deviceCategory'])
opusDf = opusDf.dropna()
df = opusDf.as_matrix()
dC = df[:,0].astype(str)
oS = df[:,1].astype(float)
```

In [55]:
```
from ipykernel import kernelapp as app
from statsmodels.stats.multicomp import pairwise_tukeyhsd
tukey = pairwise_tukeyhsd(endog = oS,
                          groups = dC,
                          alpha = 0.05)
tukey.plot_simultaneous()    # Plot group confidence intervals
plt.vlines(x=49.57,ymin=-0.5,ymax=4.5, color="red")

tukey.summary()
```

Out[55]:

Multiple Comparison of Means - Tukey HSD,FWER=0.05

group1	group2	meandiff	lower	upper	reject
Body Powered	Electric Hooks	-1.8954	-4.5431	0.7524	False
Body Powered	Multi-articulating	0.3442	-1.8397	2.5281	False
Body Powered	Single Grip	0.3736	-1.5571	2.3044	False
Electric Hooks	Multi-articulating	2.2396	-0.2206	4.6998	False
Electric Hooks	Single Grip	2.269	0.0305	4.5075	True
Multi-articulating	Single Grip	0.0294	-1.6348	1.6936	False



At a significance level of 0.05 we have a difference between electric hooks and single grips. Which was observed with ttest. Therefore we can conclude that there is a significant difference in the means between the single grips and the hooks. However we do not have significance between hooks and multi-articulating at 95% confidence interval as we saw in the ttest.

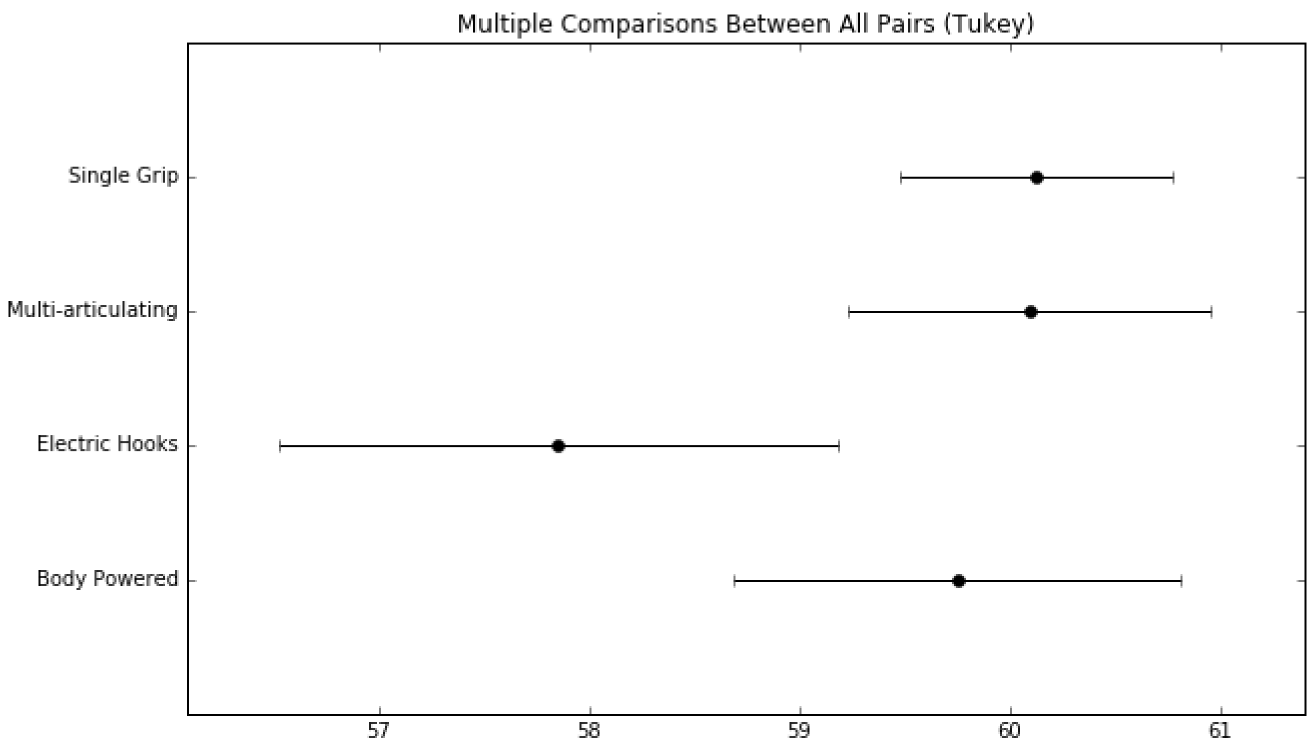
What if we broadened the significance level to 0.1

In [56]:
```
from ipykernel import kernelapp as app
from statsmodels.stats.multicomp import pairwise_tukeyhsd
tukey = pairwise_tukeyhsd(endog = oS,
                          groups = dC,
                          alpha = 0.1)
tukey.plot_simultaneous()    # Plot group confidence intervals
plt.vlines(x=49.57,ymin=-0.5,ymax=4.5, color="red")

tukey.summary()
```

Out[56]:

Multiple Comparison of Means - Tukey HSD,FWER=0.10

group1	group2	meandiff	lower	upper	reject
Body Powered	Electric Hooks	-1.8954	-4.2561	0.4653	False
Body Powered	Multi-articulating	0.3442	-1.6029	2.2914	False
Body Powered	Single Grip	0.3736	-1.3478	2.095	False
Electric Hooks	Multi-articulating	2.2396	0.0461	4.4331	True
Electric Hooks	Single Grip	2.269	0.2732	4.2648	True
Multi-articulating	Single Grip	0.0294	-1.4544	1.5132	False



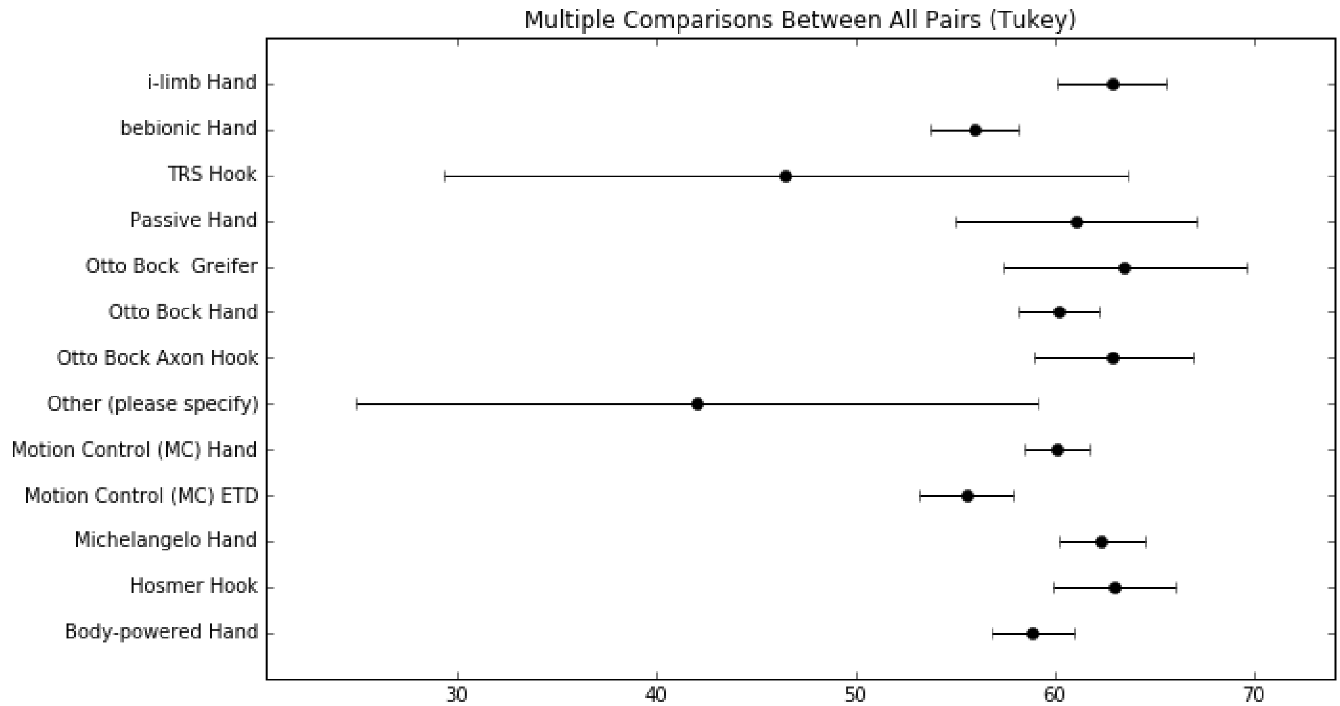
If we look at a 90% confidence interval, then we have significance between electric hooks vs single grip and electric hooks vs. multi-articulating. With electric hooks underperforming on opus.

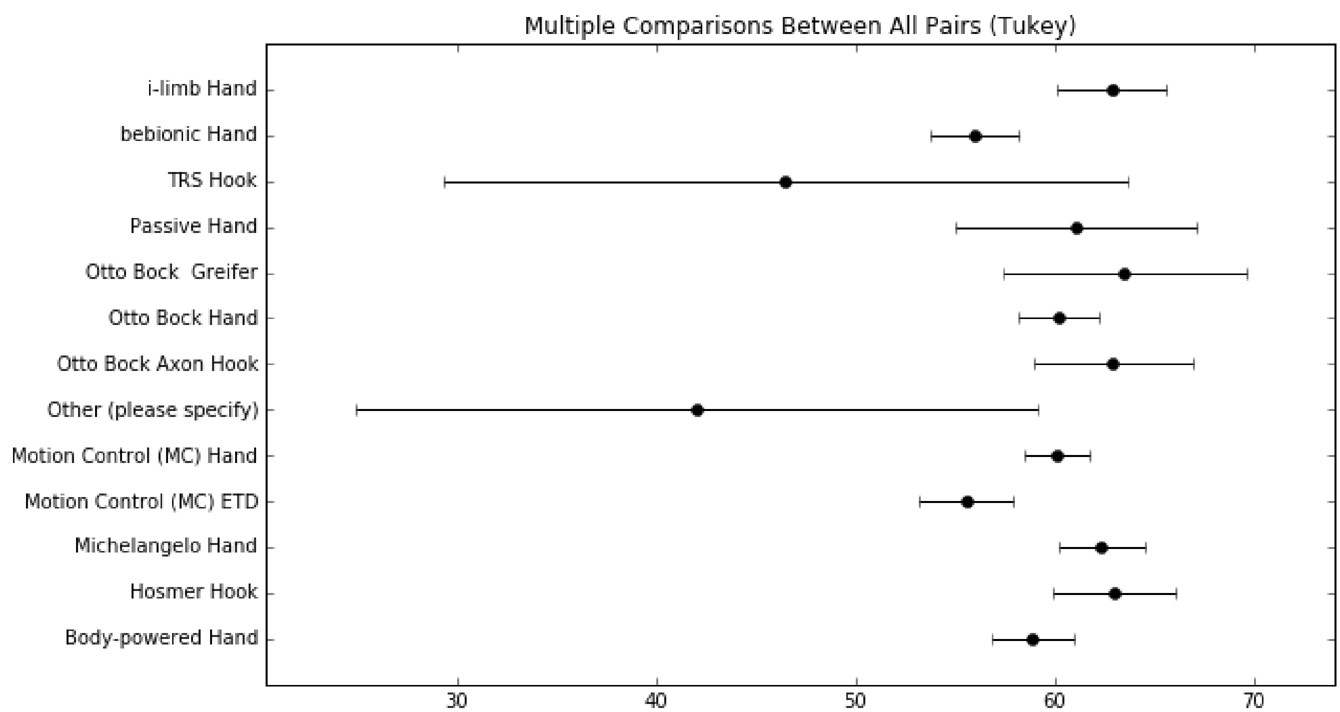
Significance among brands

I'm curious if we'll get something by looking at the specific brands

In [39]:
```
opusDf = pd.concat([sht['PrimaryTerminalDevice'], sht['opusScore']], axis=1, keys=['devi
ceCategory','opusScore'])
opusDf = opusDf.dropna()
df = opusDf.as_matrix()
dC = df[:,0].astype(str)
oS = df[:,1].astype(float)
tukey = pairwise_tukeyhsd(endog = oS,
                          groups = dC,
                          alpha = 0.05)
tukey.plot_simultaneous()    # Plot group confidence intervals
#tukey.summary()
```

Out[39]:





The TRS Hook and "other" are mucking up the scale, let's remove those

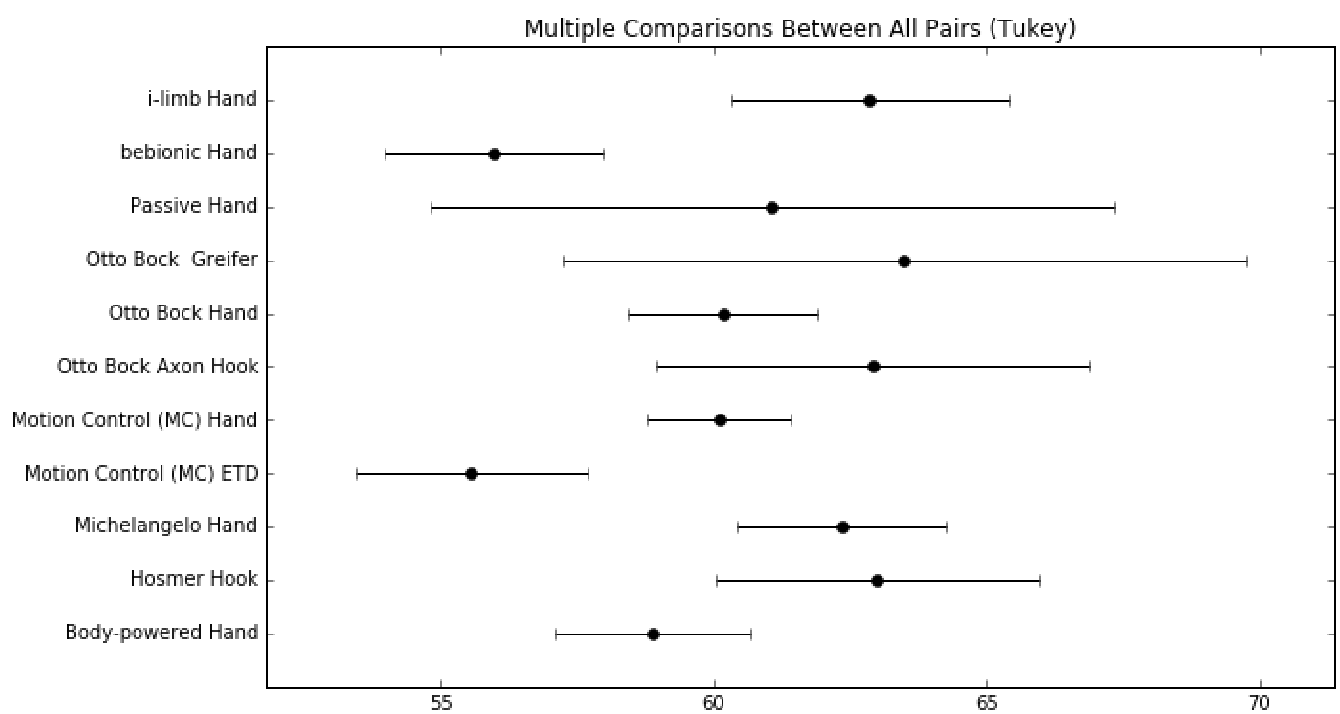
In [40]:
```
opusDf = pd.concat([sht['PrimaryTerminalDevice'], sht['opusScore']], axis=1, keys=['prim
aryTD','opusScore'])
opusDf["primaryTD"] = opusDf["primaryTD"].replace('Other.*', np.nan, regex=True)
opusDf["primaryTD"] = opusDf["primaryTD"].replace('TRS.*', np.nan, regex=True)
opusDf = opusDf.dropna()
df = opusDf.as_matrix()
dC = df[:,0].astype(str)
oS = df[:,1].astype(float)
tukey = pairwise_tukeyhsd(endog = oS,
                         groups = dC,
                         alpha = 0.05)
tukey.plot_simultaneous()    # Plot group confidence intervals
tukey.summary()
```

Out[40]:

Multiple Comparison of Means - Tukey HSD,FWER=0.05

group1	group2	meandiff	lower	upper	reject
Body-powered Hand	Hosmer Hook	4.1111	-0.7064	8.9286	False
Body-powered Hand	Michelangelo Hand	3.4683	-0.0788	7.0153	False
Body-powered Hand	Motion Control (MC) ETD	-3.3257	-7.1411	0.4897	False
Body-powered Hand	Motion Control (MC) Hand	1.2208	-1.4984	3.9399	False
Body-powered Hand	Otto Bock Axon Hook	4.0342	-1.9094	9.9778	False
Body-powered Hand	Otto Bock Hand	1.2847	-2.0399	4.6094	False
Body-powered Hand	Otto Bock Greifer	4.6111	-3.7486	12.9708	False
Body-powered Hand	Passive Hand	2.1944	-6.1653	10.5542	False
Body-powered Hand	bebionic Hand	-2.9186	-6.5697	0.7325	False
Body-powered Hand	i-limb Hand	3.9755	-0.3556	8.3066	False
Hosmer Hook	Michelangelo Hand	-0.6429	-5.5805	4.2948	False
Hosmer Hook	Motion Control (MC) ETD	-7.4368	-12.5706	-2.303	True
Hosmer Hook	Motion Control (MC) Hand	-2.8904	-7.2713	1.4906	False
Hosmer Hook	Otto Bock Axon Hook	-0.0769	-6.9417	6.7879	False
Hosmer Hook	Otto Bock Hand	-2.8264	-7.6068	1.954	False
Hosmer Hook	Otto Bock Greifer	0.5	-8.5379	9.5379	False
Hosmer Hook	Passive Hand	-1.9167	-10.9546	7.1212	False
Hosmer Hook	bebionic Hand	-7.0297	-12.0426	-2.0168	True
Hosmer Hook	i-limb Hand	-0.1356	-5.6635	5.3923	False
Michelangelo Hand	Motion Control (MC) ETD	-6.7939	-10.7599	-2.828	True
Michelangelo Hand	Motion Control (MC) Hand	-2.2475	-5.1742	0.6792	False
Michelangelo Hand	Otto Bock Axon Hook	0.5659	-5.4754	6.6073	False
Michelangelo Hand	Otto Bock Hand	-2.1835	-5.6799	1.3129	False
Michelangelo Hand	Otto Bock Greifer	1.1429	-7.2867	9.5724	False
Michelangelo Hand	Passive Hand	-1.2738	-9.7033	7.1557	False
Michelangelo Hand	bebionic Hand	-6.3868	-10.195	-2.5787	True
Michelangelo Hand	i-limb Hand	0.5073	-3.957	4.9716	False
Motion Control (MC) ETD	Motion Control (MC) Hand	4.5464	1.2997	7.7932	True
Motion Control (MC) ETD	Otto Bock Axon Hook	7.3599	1.1571	13.5626	True
Motion Control (MC) ETD	Otto Bock Hand	4.6104	0.842	8.3788	True
Motion Control (MC) ETD	Otto Bock Greifer	7.9368	-0.6091	16.4827	False
Motion Control (MC) ETD	Passive Hand	5.5201	-3.0258	14.066	False
Motion Control (MC) ETD	bebionic Hand	0.4071	-3.6522	4.4664	False

Motion Control (MC) ETD	i-limb Hand	7.3012	2.6208	11.9816	True
Motion Control (MC) Hand	Otto Bock Axon Hook	2.8134	-2.7822	8.409	False
Motion Control (MC) Hand	Otto Bock Hand	0.064	-2.5888	2.7168	False
Motion Control (MC) Hand	Otto Bock Greifer	3.3904	-4.7256	11.5063	False
Motion Control (MC) Hand	Passive Hand	0.9737	-7.1423	9.0897	False
Motion Control (MC) Hand	bebionic Hand	-4.1394	-7.1913	-1.0874	True
Motion Control (MC) Hand	i-limb Hand	2.7548	-1.0848	6.5944	False
Otto Bock Axon Hook	Otto Bock Hand	-2.7495	-8.663	3.1641	False
Otto Bock Axon Hook	Otto Bock Greifer	0.5769	-9.1082	10.2621	False
Otto Bock Axon Hook	Passive Hand	-1.8397	-11.5249	7.8454	False
Otto Bock Axon Hook	bebionic Hand	-6.9528	-13.0558	-0.8498	True
Otto Bock Axon Hook	i-limb Hand	-0.0587	-6.5913	6.474	False
Otto Bock Hand	Otto Bock Greifer	3.3264	-5.012	11.6648	False
Otto Bock Hand	Passive Hand	0.9097	-7.4287	9.2481	False
Otto Bock Hand	bebionic Hand	-4.2033	-7.8052	-0.6014	True
Otto Bock Hand	i-limb Hand	2.6908	-1.599	6.9805	False
Otto Bock Greifer	Passive Hand	-2.4167	-13.7463	8.913	False
Otto Bock Greifer	bebionic Hand	-7.5297	-16.0035	0.9441	False
Otto Bock Greifer	i-limb Hand	-0.6356	-9.4239	8.1527	False
Passive Hand	bebionic Hand	-5.113	-13.5868	3.3608	False
Passive Hand	i-limb Hand	1.7811	-7.0072	10.5693	False
bebionic Hand	i-limb Hand	6.8941	2.3467	11.4415	True



In [41]:
```
groupbyPrimaryD = sht["opusScore"].groupby(sht["PrimaryTerminalDevice"])
groupbyPrimaryD.mean()
```

Out[41]:
```
PrimaryTerminalDevice
Body-powered Hand           58.888889
Hosmer Hook                 63.000000
Michelangelo Hand           62.357143
Motion Control (MC) ETD     55.563218
Motion Control (MC) Hand    60.109649
Other (please specify)      42.000000
Otto Bock Axon Hook         62.923077
Otto Bock Hand              60.173611
Otto Bock Greifer           63.500000
Passive Hand                61.083333
TRS Hook                    46.500000
bebionic Hand               55.970297
i-limb Hand                 62.864407
Name: opusScore, dtype: float64
```