

automate do\_dashboard() for backtest

```
In [1]: from IPython.core.display import display, HTML
from IPython.display import Image

from pathlib import Path
from datetime import datetime

from gwg_config import *
from gwg_db import DBUtils, DBConn
from gwg_sql import *
from gwg_ta import *
from gwg_quote import *
from gwg_helper import *

import warnings
warnings.filterwarnings("ignore")

from time import time

from gwg_chart import (
    rsi_score_trend_chart_one_v4,
    rsi_chart_v1, gen_chart_v6, # gen_chart_v5,
    _download_quote,
    chart_index_trend, plot_rsi_score_trend_v2, plot_rsi_score_trend_v4,
    plot_rsi_score_trend_v2A,
    chart_index_rsi_score_trend_v2, chart_index_rsi_score_trend_zoom,
    chart_ticker
)
```

```
/tmp/ipykernel_1452255/3050621922.py:1: DeprecationWarning: Importing display from IPython.core.display is deprecated since IPython 7.14, please import from IPython display
from IPython.core.display import display, HTML
```

```
In [2]: dates = []
for y in range(2005, 2024):    # 2005, 2024
    dates.append(f"{y}/02/02")
    dates.append(f"{y}/08/02")

dates.append("2024/08/03")
len(dates)
```

Out[2]: 38

```
In [12]: dates.append("2024/08/03")
```

```
In [3]: def _display_img(file_img, width=600, height=300):
    if file_img and Path(file_img).exists():
        # st.image(Image.open(file_img))
        display(Image(filename=file_img), metadata={'width': width, 'height'}
```

```
In [4]: CFG["RSI_SCORE_TREND_NDAYS"]
```

Out[4]: 300

```
In [5]: def gen_rsi_trend_chart(rsi_date
    , ndays_back = CFG["RSI_SCORE_TREND_NDAYS"]
    , use_existing = True # False #
):
    df_summary_rsi_score_bull = DBU_.summary_rsi_scores(rsi_date, "Bull", nc)
    df_summary_rsi_score_bear = DBU_.summary_rsi_scores(rsi_date, "Bear", nc)

    file_img = rsi_score_trend_chart_one_v4(df_summary_rsi_score_bull,
                                              df_summary_rsi_score_bear,
                                              rsi_date,
                                              ndays_back=ndays_back,
                                              use_existing=use_existing)
    _display_img(file_img)

for etf_ticker in ["SPY", "QQQ", "IWM"]:
    xx = CFG["SELECTED_ETFS"]["INDEX"].get(etf_ticker)
    list_name = xx.get("list_name")
    sector = xx.get("sector")
    file_img = chart_index_rsi_score_trend_zoom(etf_ticker,
                                                list_name, sector,
                                                trend_date=rsi_date,
                                                zoom_window=ndays_back,
                                                use_existing=use_existing)
    _display_img(file_img)
```

In [6]: DBU\_ = DBUtils()

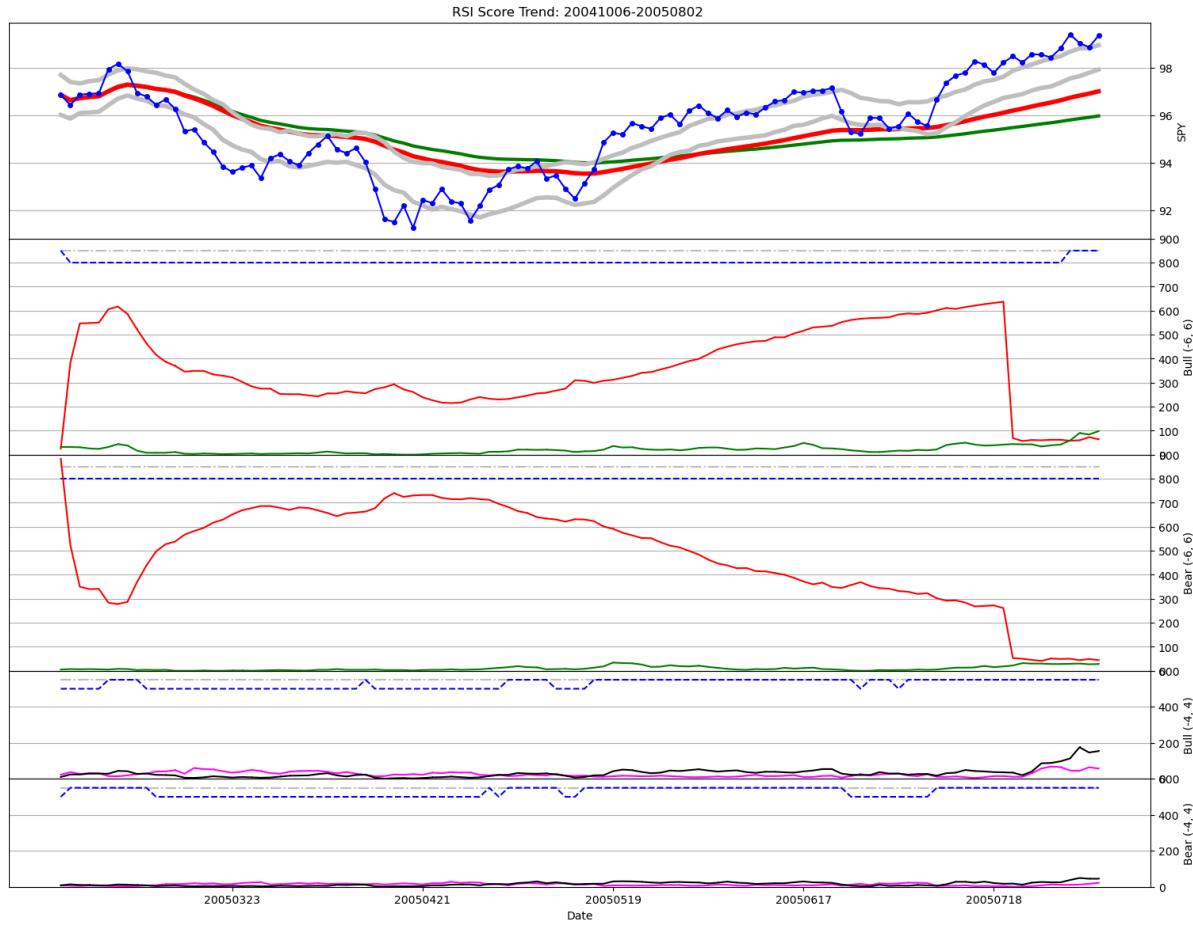
In [ ]:

```
In [7]: ts_start = time()
print(datetime.now())
```

2024-08-04 15:12:26.794851

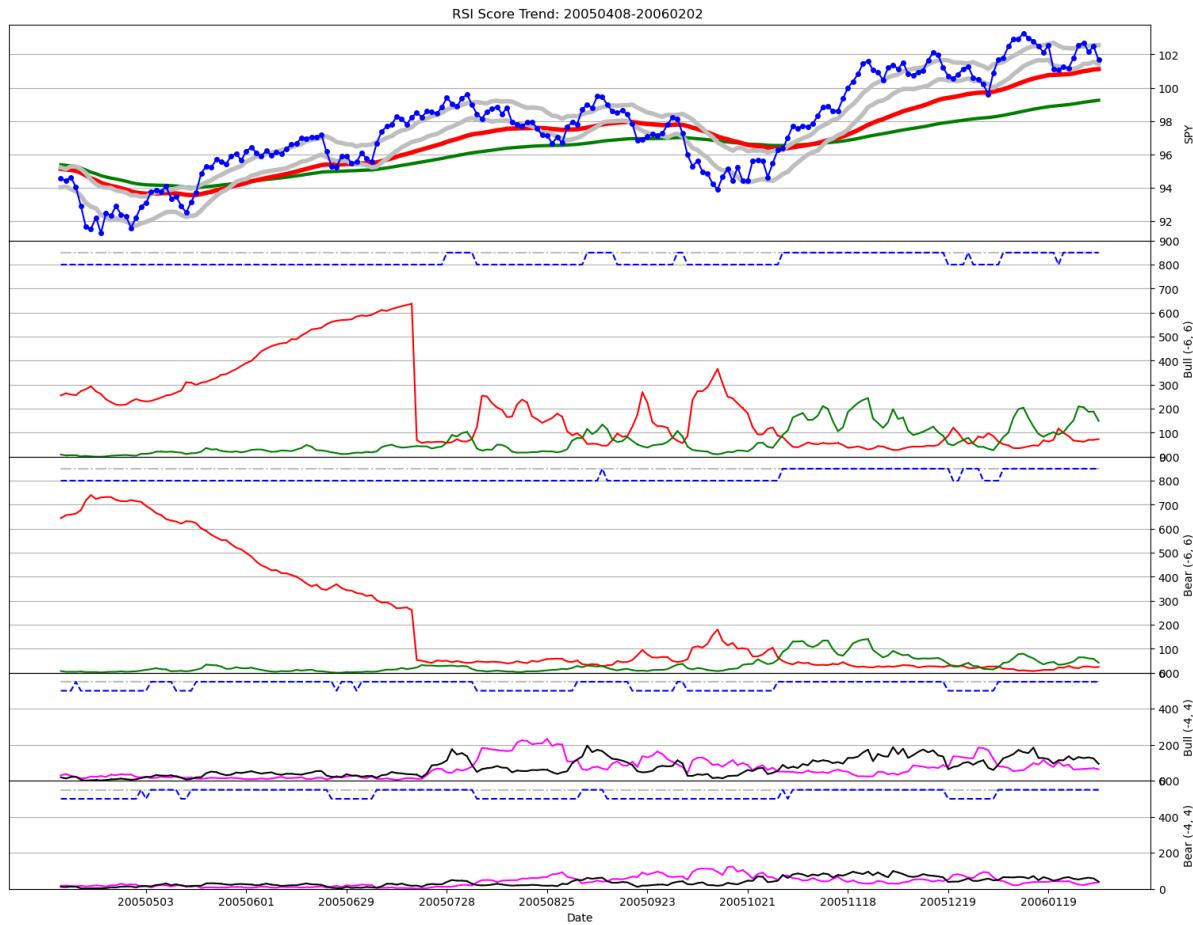
```
In [8]: for rsi_date in dates:
    try:
        # ts_1 = time()
        gen_rsi_trend_chart(rsi_date.replace("/", ""))
        # ts_2 = time()
        # print(f"gen_chart {rsi_date} done in {ts_2-ts_1:.1f} sec")
    except Exception as ex:
        print(f"[Failed] {rsi_date}: {str(ex)}")
        pass
```

[Failed] 2005/02/02:  
no numeric data to plot

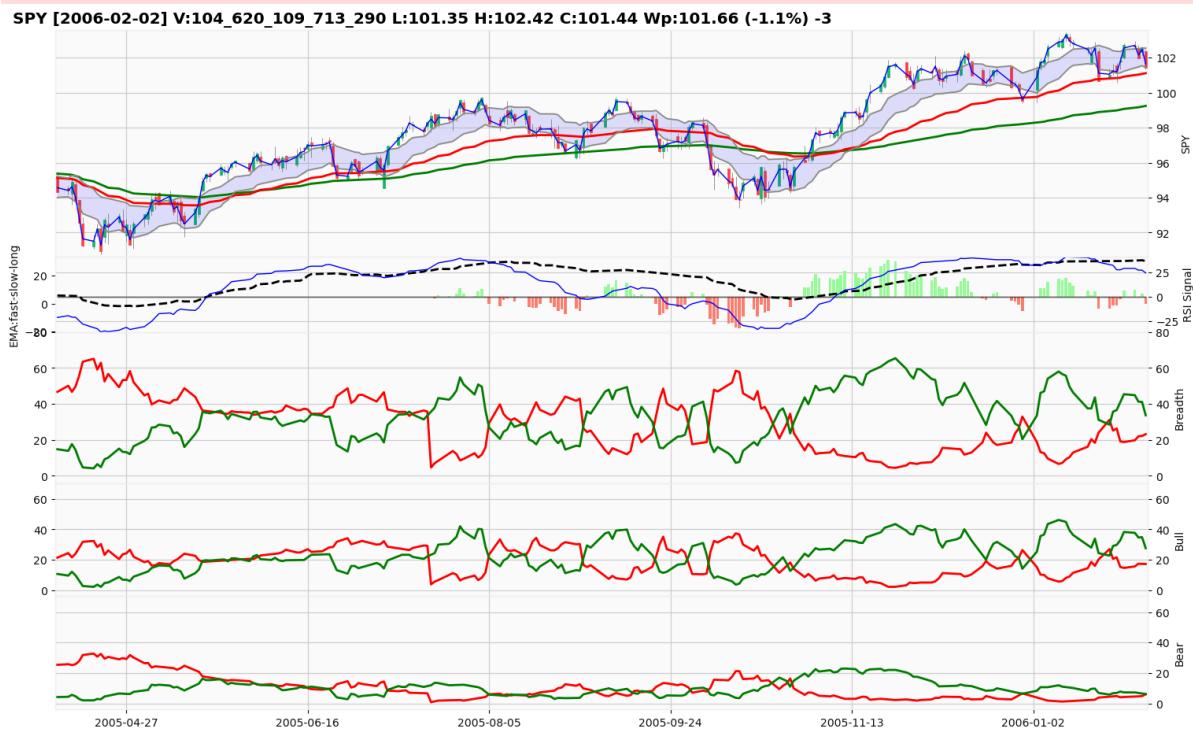


[Failed] 2005/08/02:

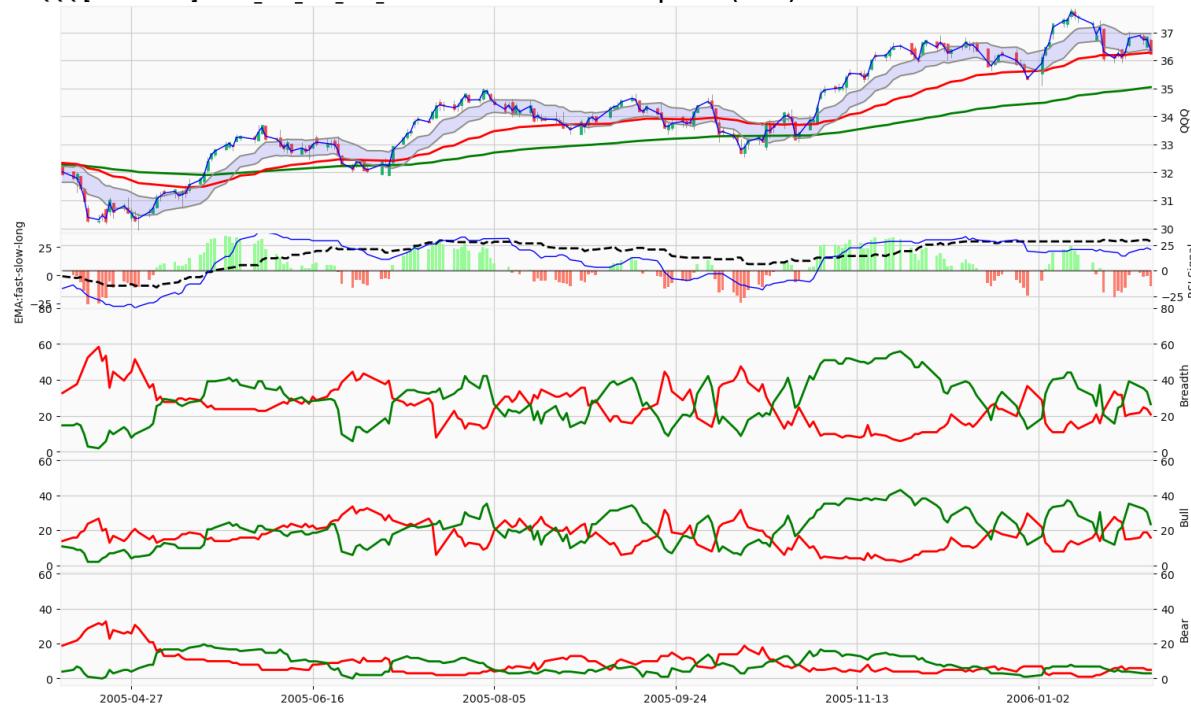
[ERROR] chart\_index\_trend(): insufficient data: SPY, Skip!



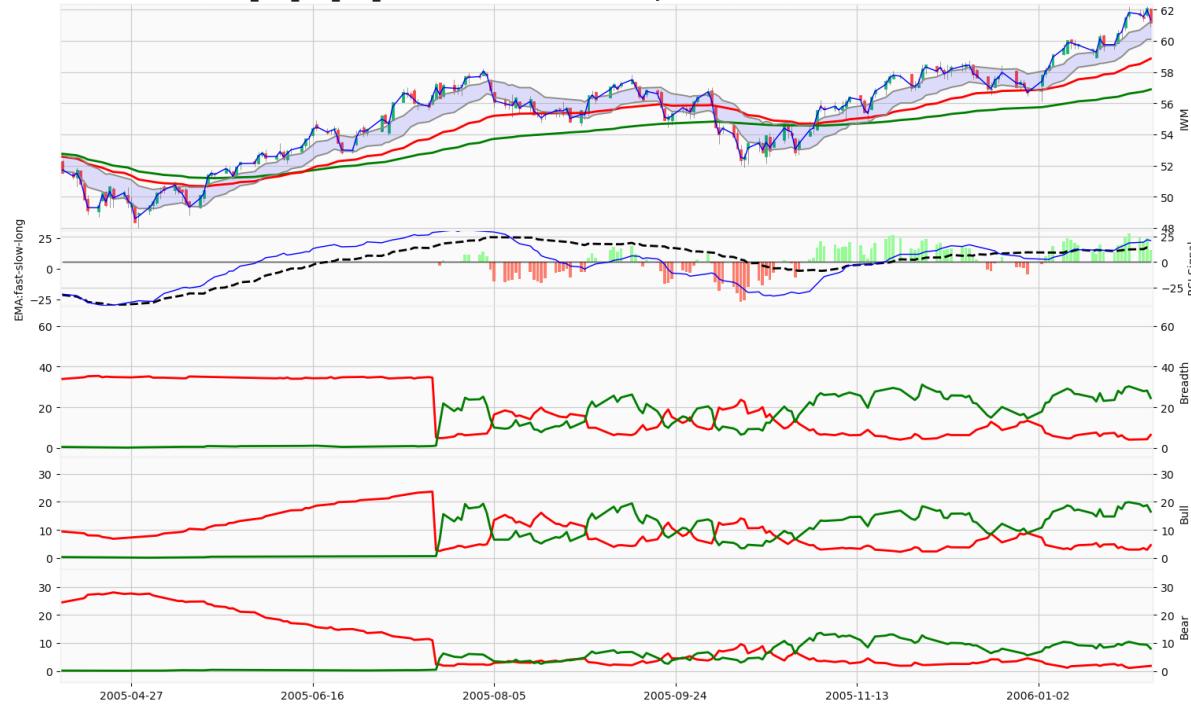
2024-08-04 15:12:29.042 WARNING streamlit.runtime.state.session\_state\_proxy:  
Session state does not function when running a script without `streamlit run`

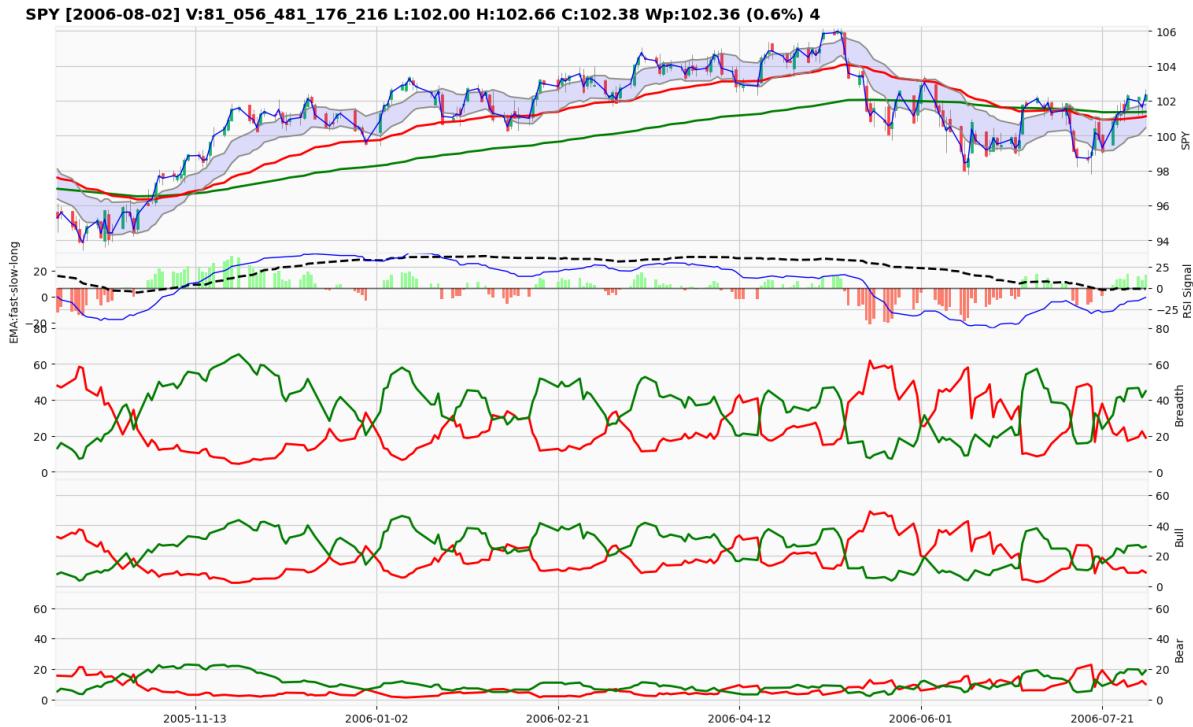
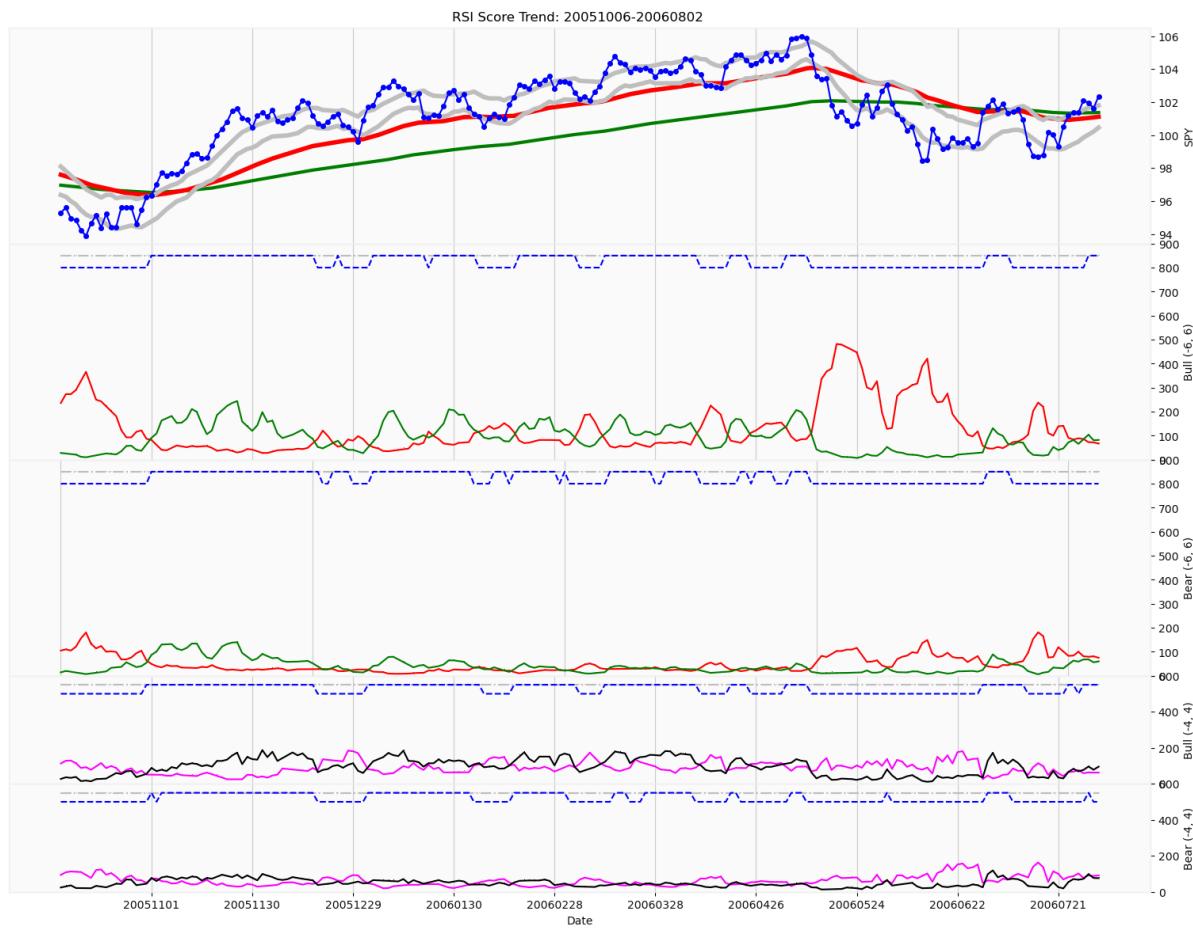


QQQ [2006-02-02] V:107\_558\_158\_478\_580 L:36.21 H:36.79 C:36.22 Wp:36.36 (-1.7%) -3

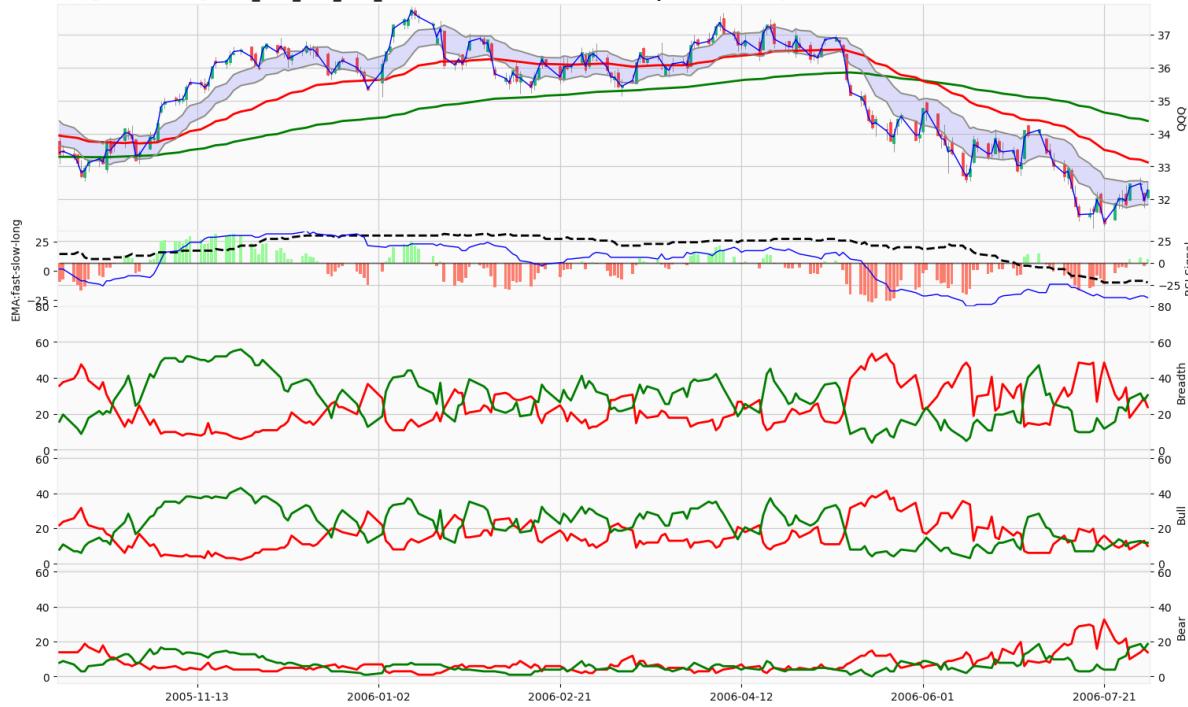


IWM [2006-02-02] V:61\_492\_780\_994\_499 L:60.89 H:62.10 C:61.18 Wp:61.34 (-1.5%) 2

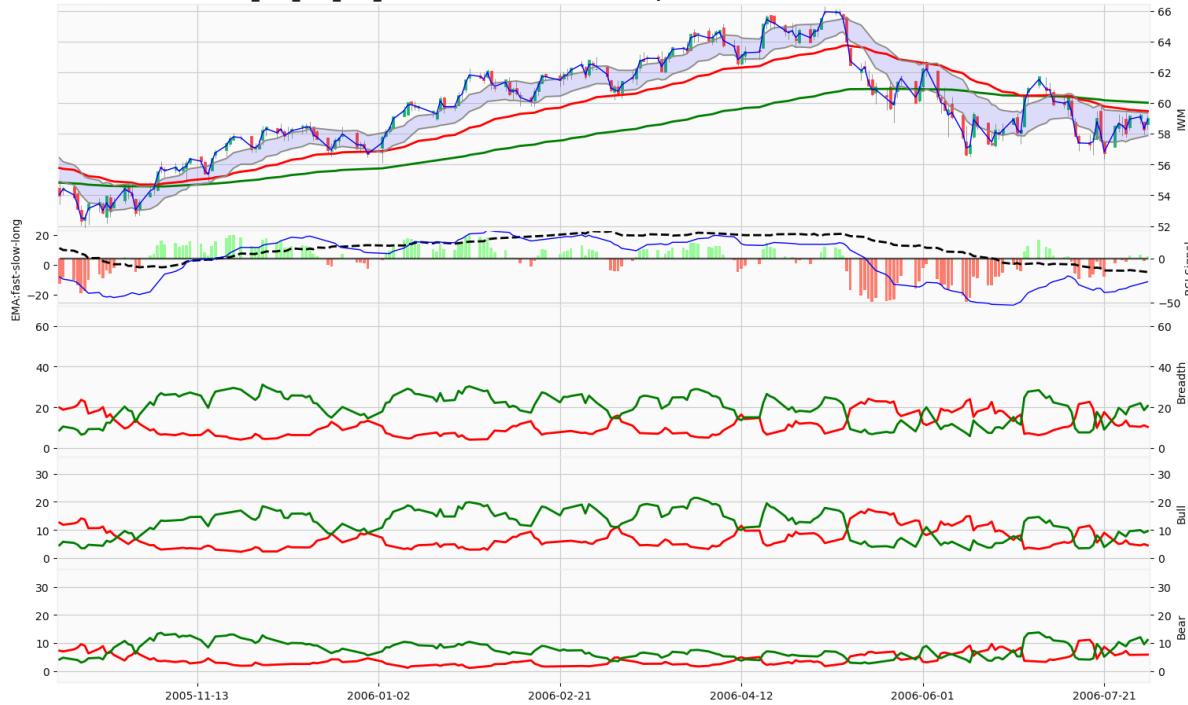


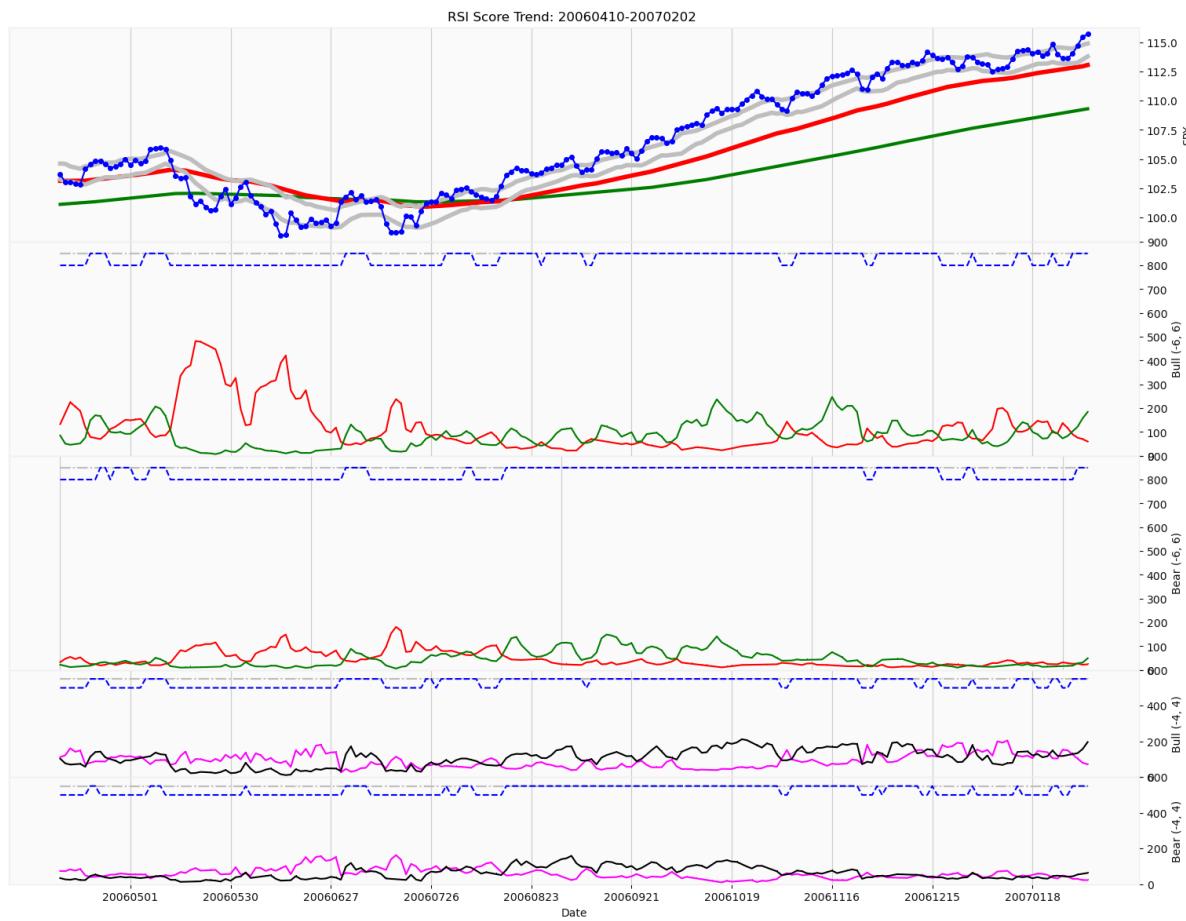


**QQQ [2006-08-02] V:99\_790\_386\_432\_614 L:32.06 H:32.51 C:32.30 Wp:32.29 (1.1%) 0**

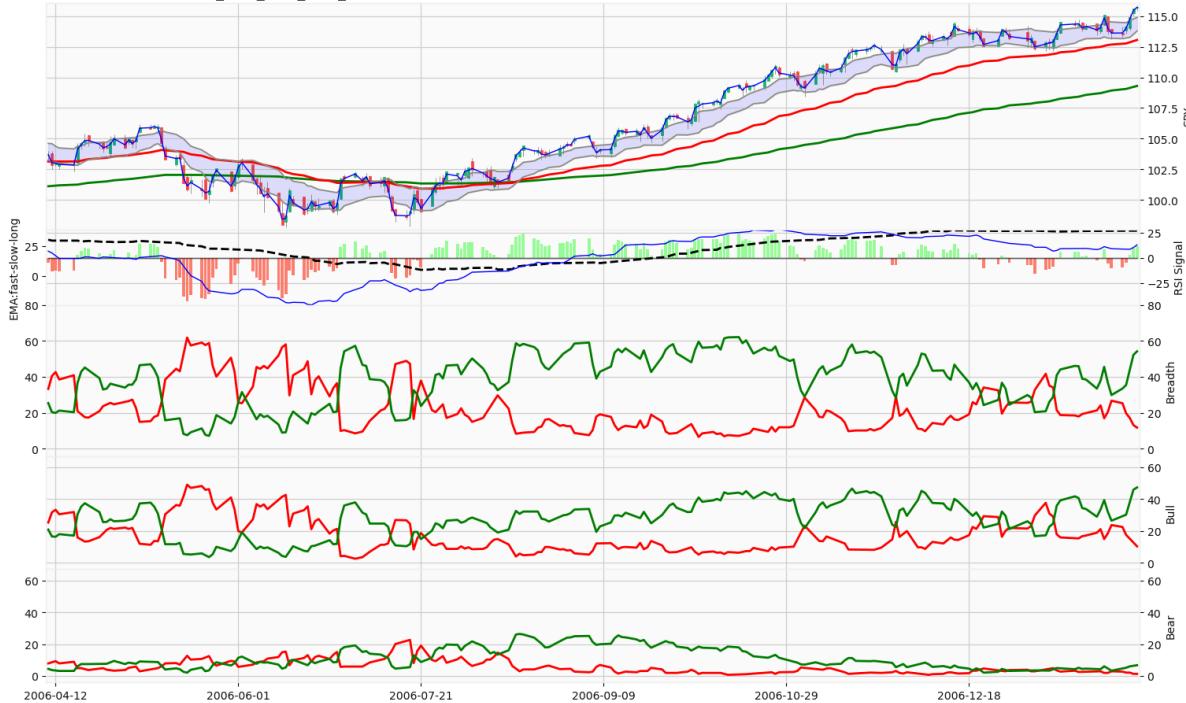


**IWM [2006-08-02] V:59\_005\_462\_484\_659 L:58.32 H:59.24 C:59.05 Wp:58.92 (1.3%) 0**

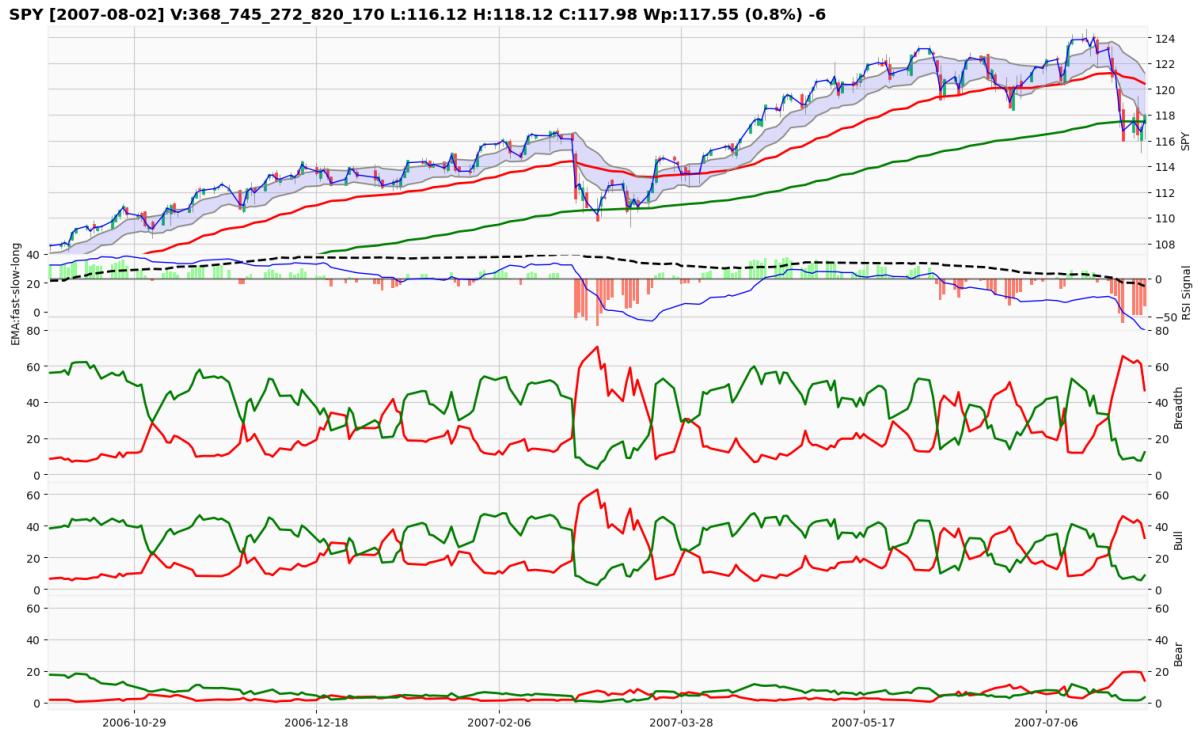
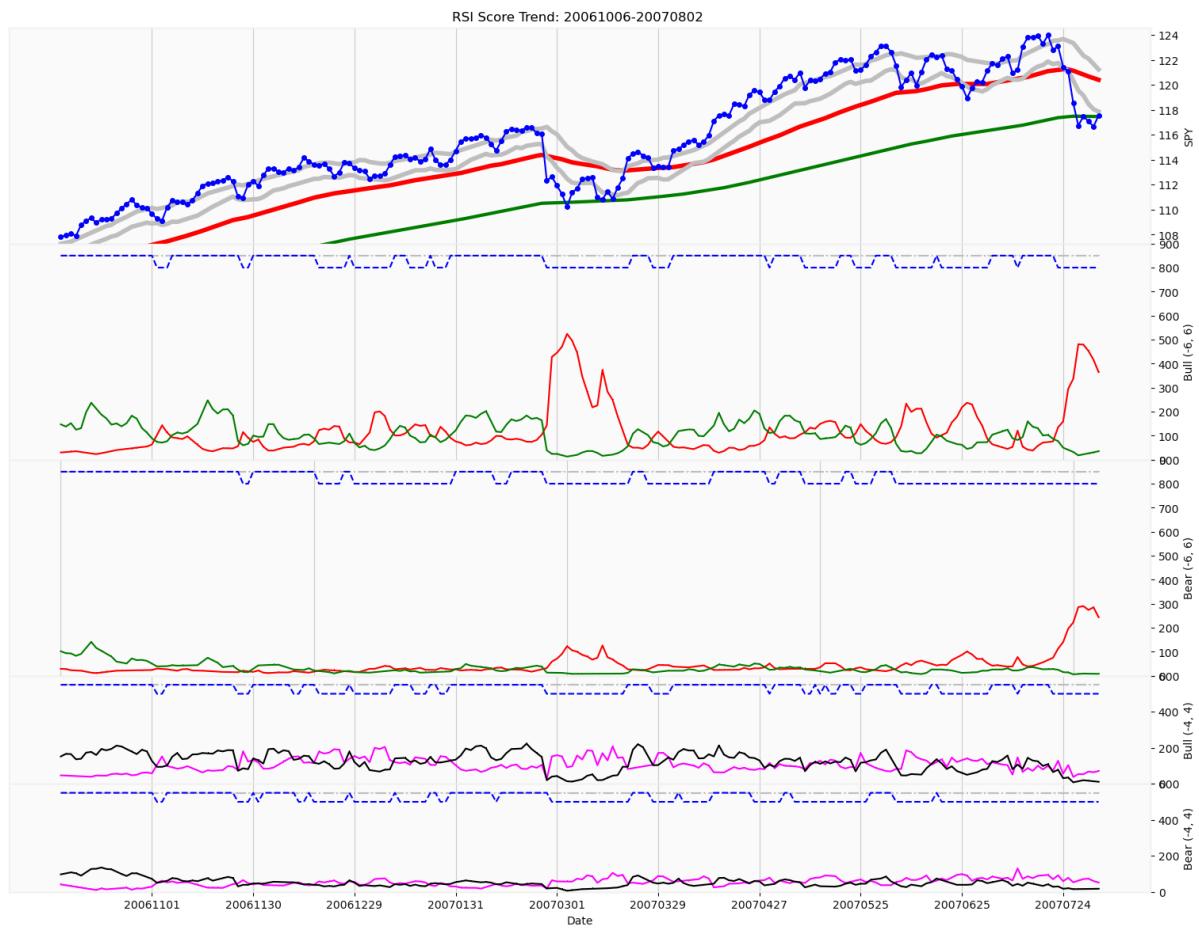


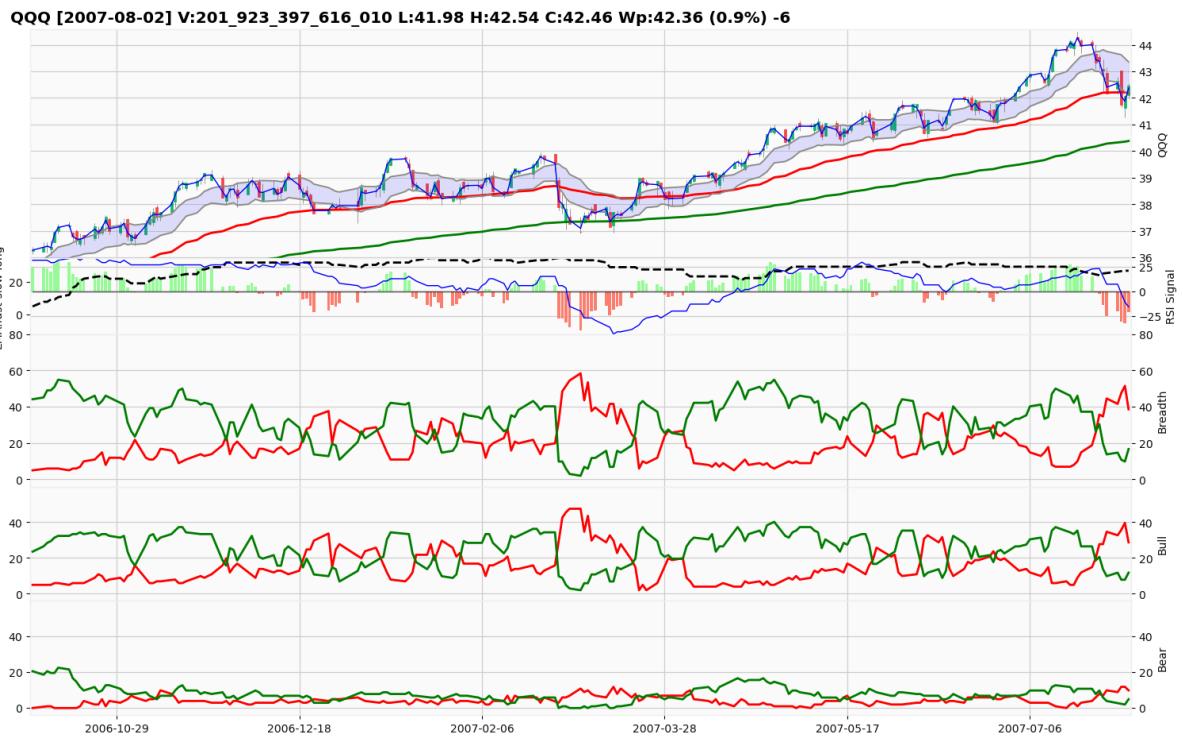


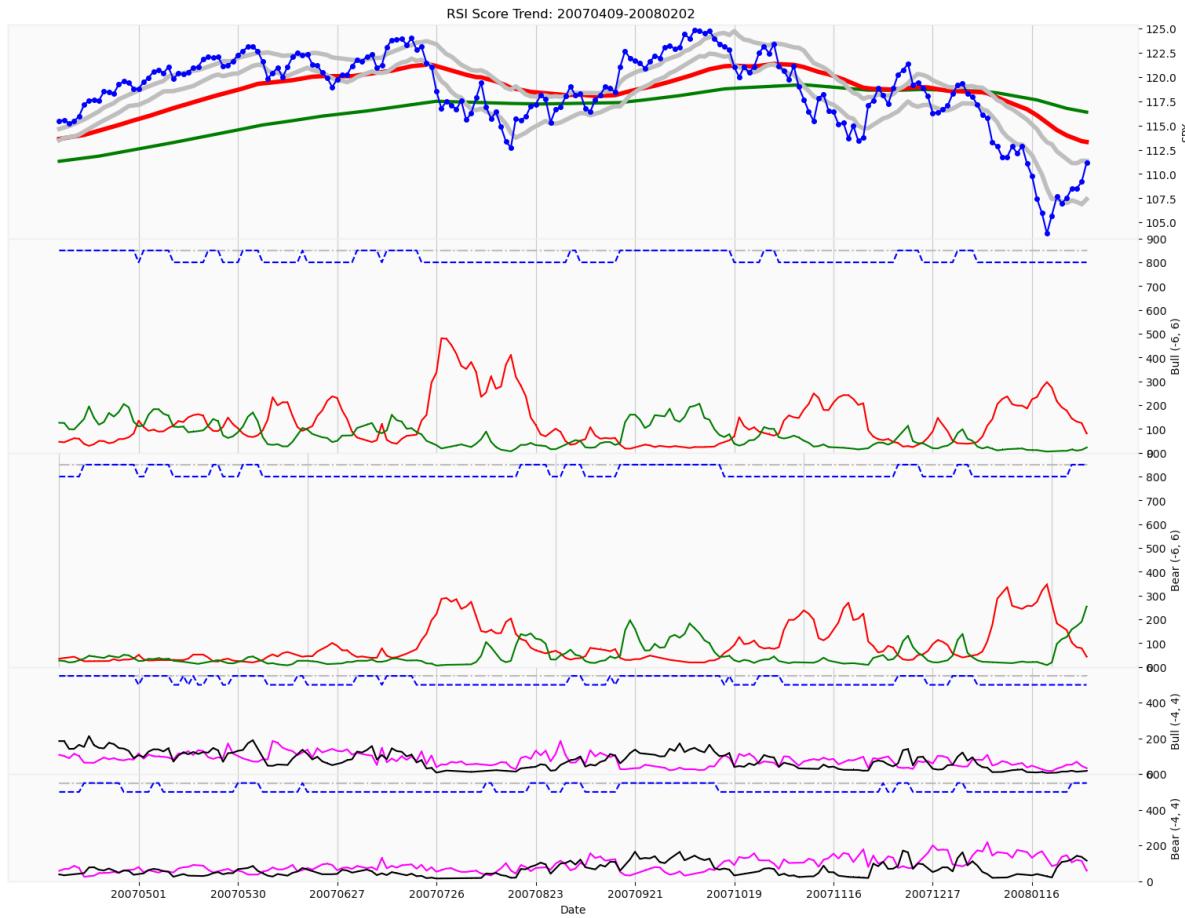
**SPY [2007-02-02] V:62\_059\_663\_495\_581 L:115.39 H:115.88 C:115.76 Wp:115.70 (0.1%) 4**



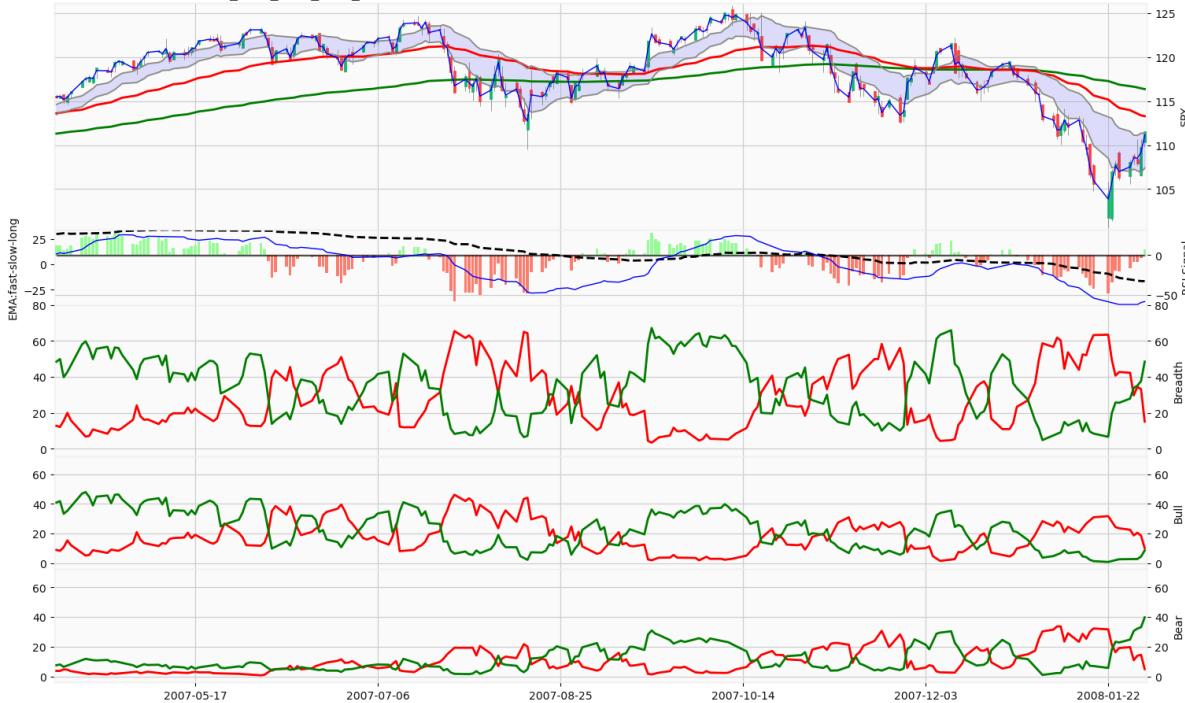


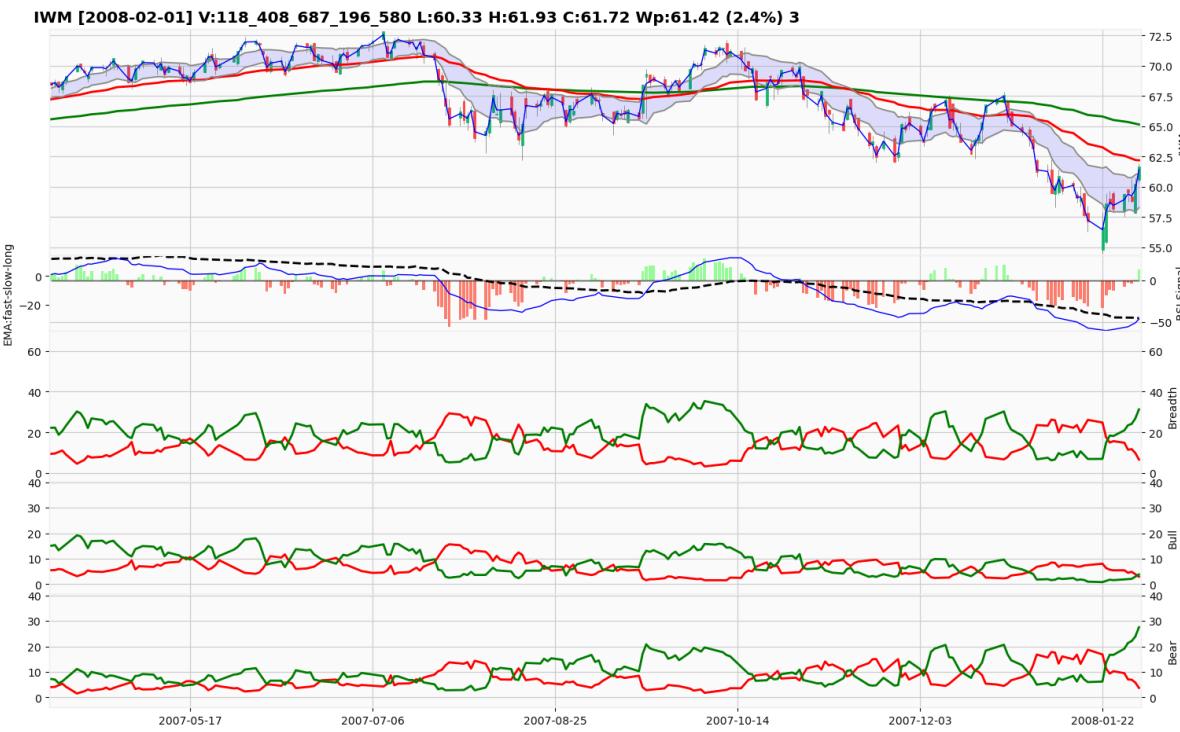
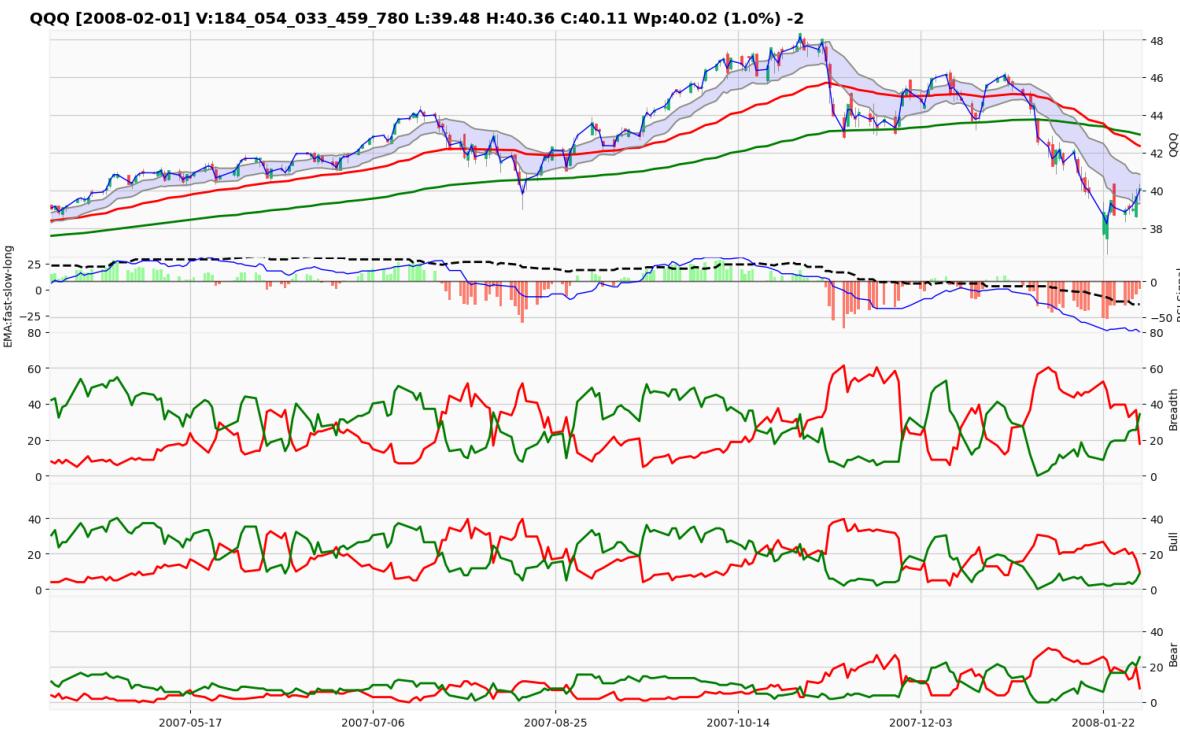


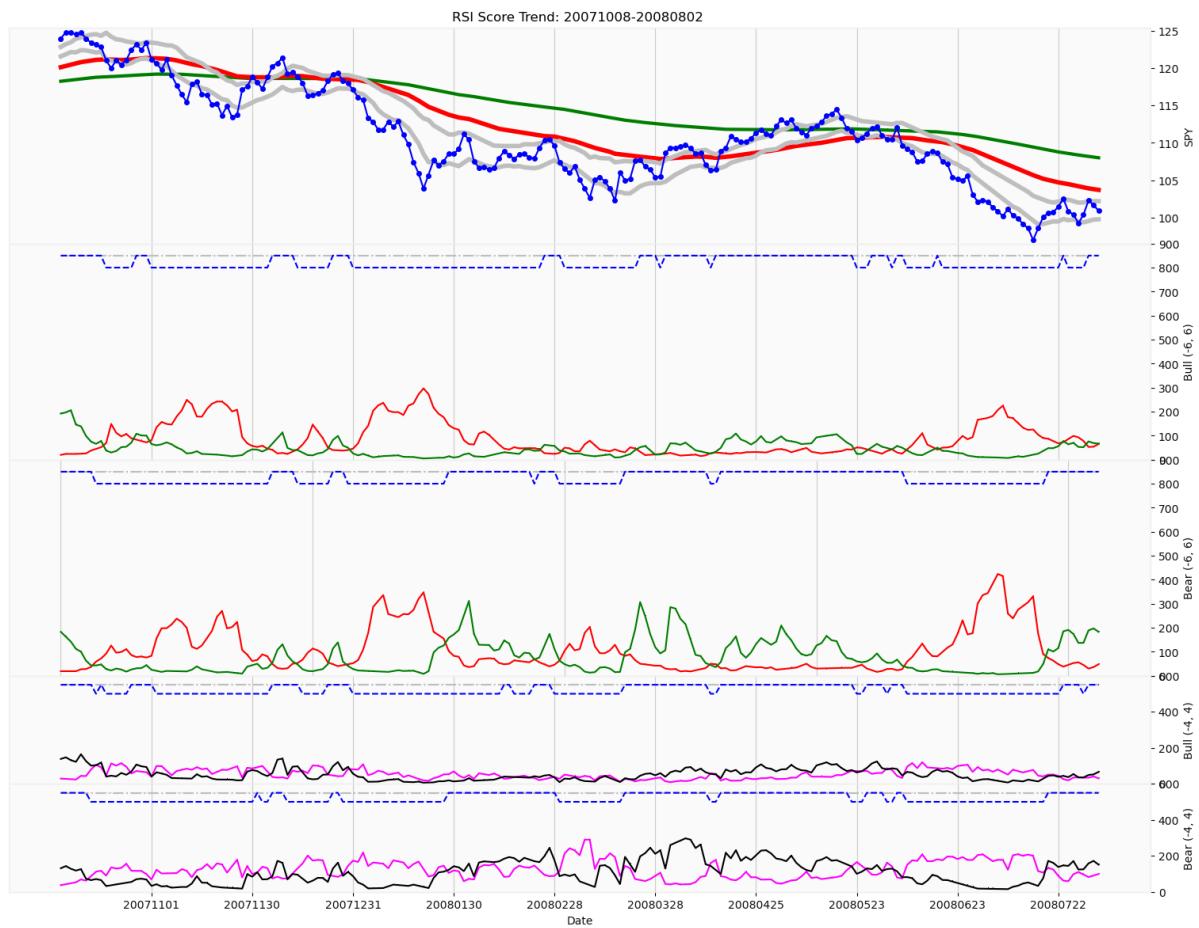




**SPY [2008-02-01] V:258\_763\_254\_058\_700 L:109.92 H:111.61 C:111.58 Wp:111.17 (1.6%) 3**



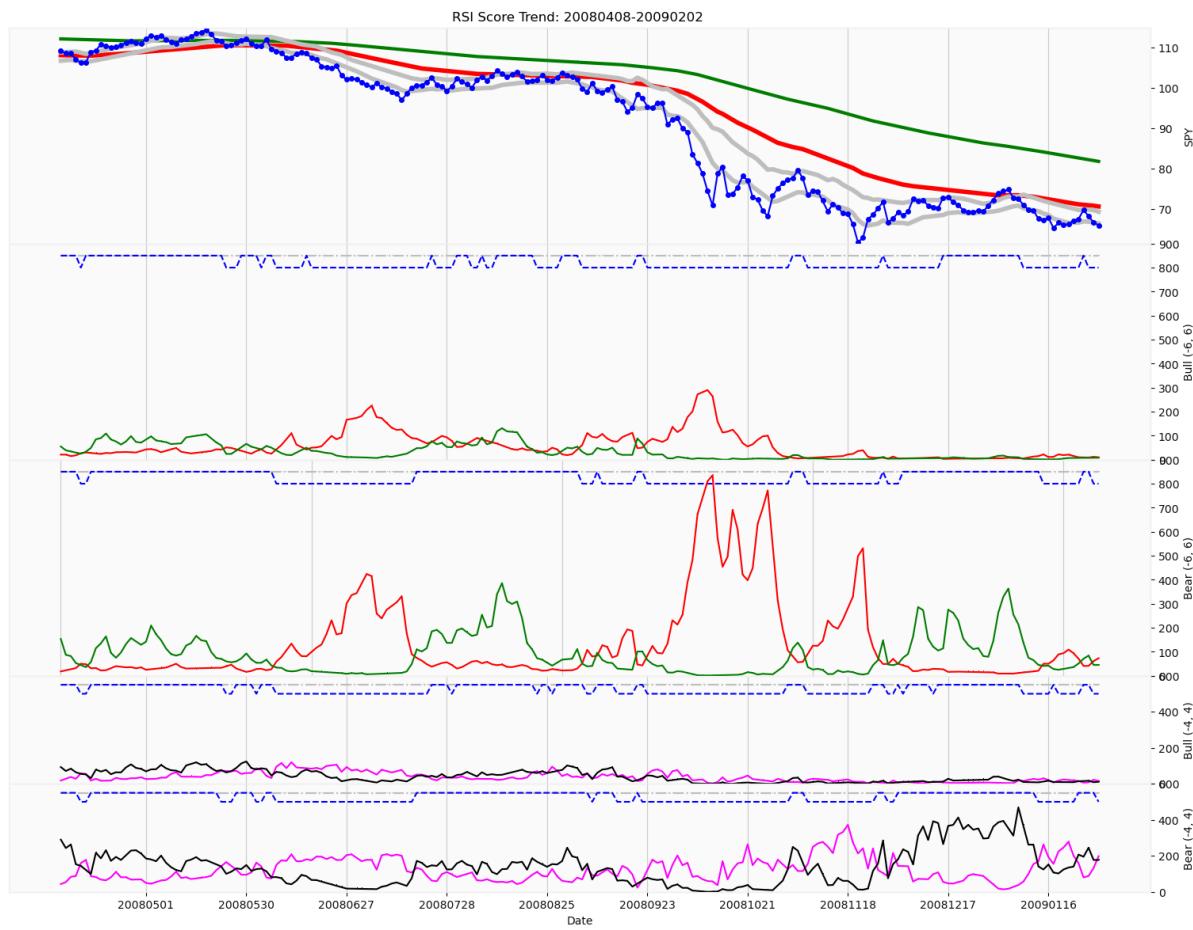




**SPY [2008-08-01] V:311\_130\_317\_382\_590 L:100.28 H:101.78 C:100.80 Wp:100.91 (-0.6%) 1**



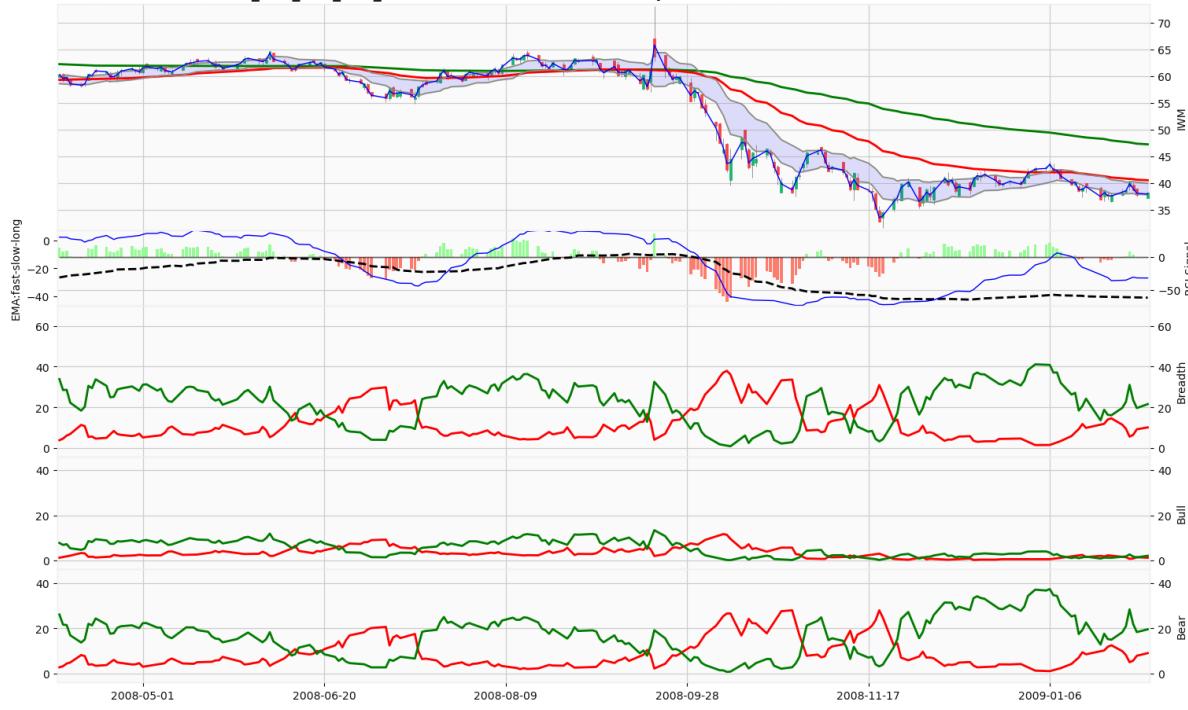


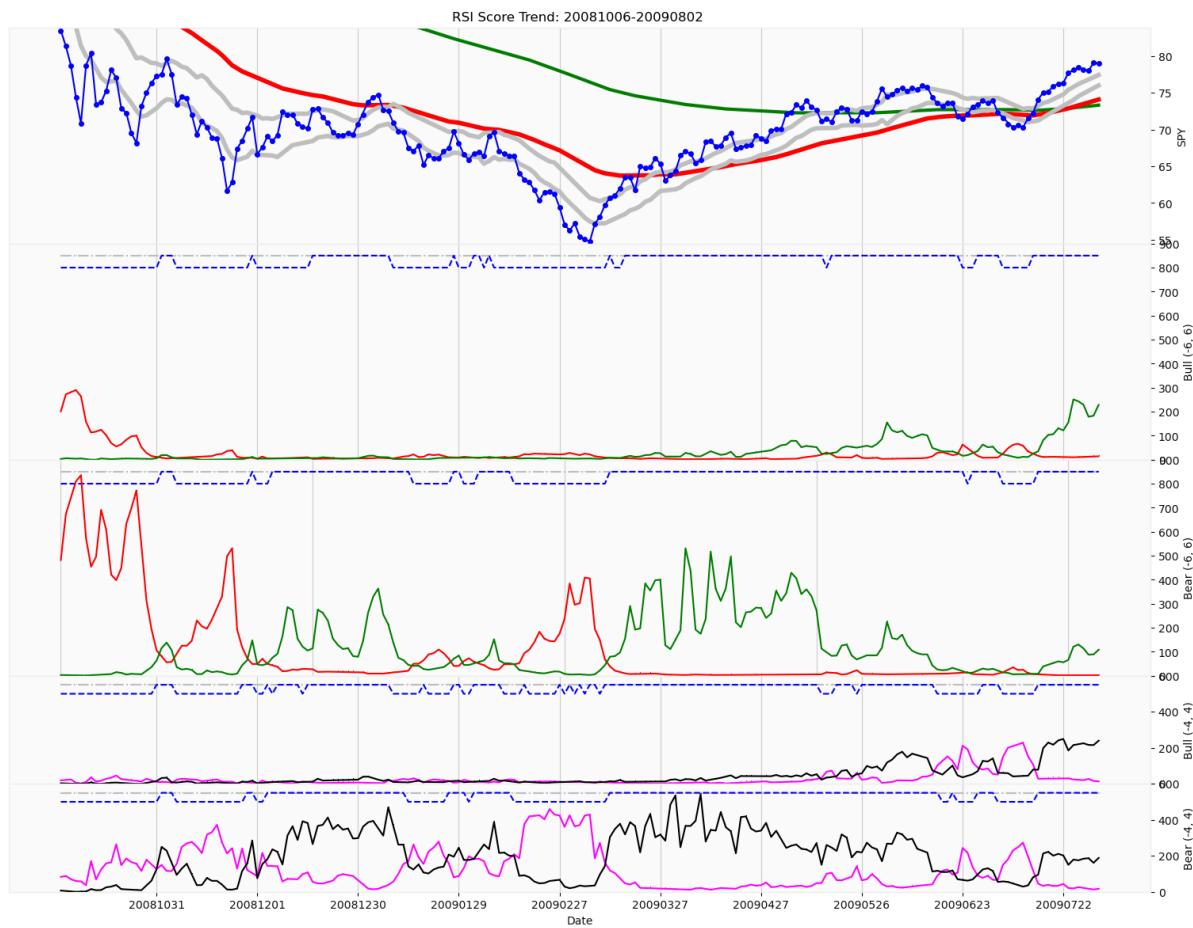


QQQ [2009-02-02] V:143\_582\_205\_005\_230 L:25.38 H:26.15 C:25.97 Wp:25.87 (1.2%) 0

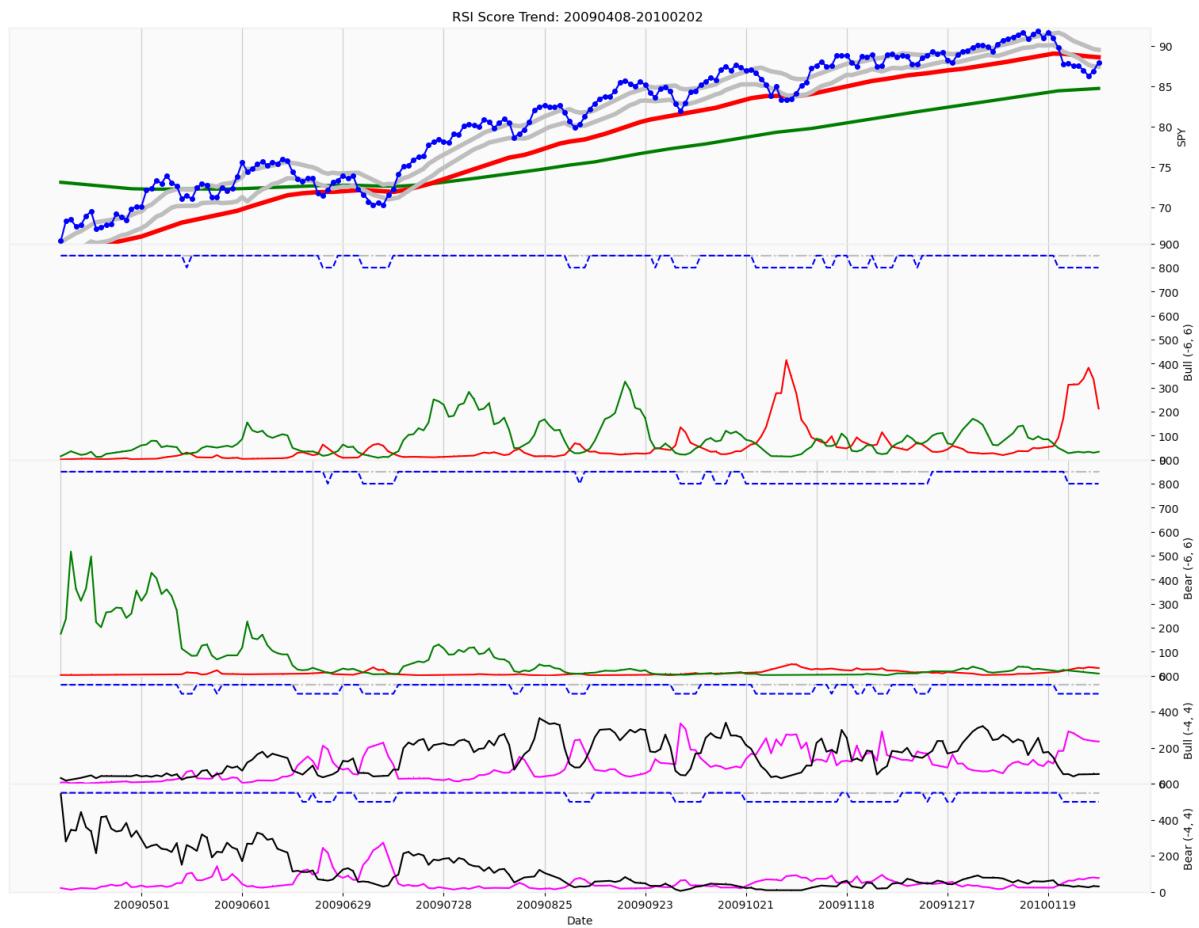


IWM [2009-02-02] V:74\_035\_602\_097\_755 L:37.08 H:38.36 C:38.10 Wp:37.91 (0.8%) 0

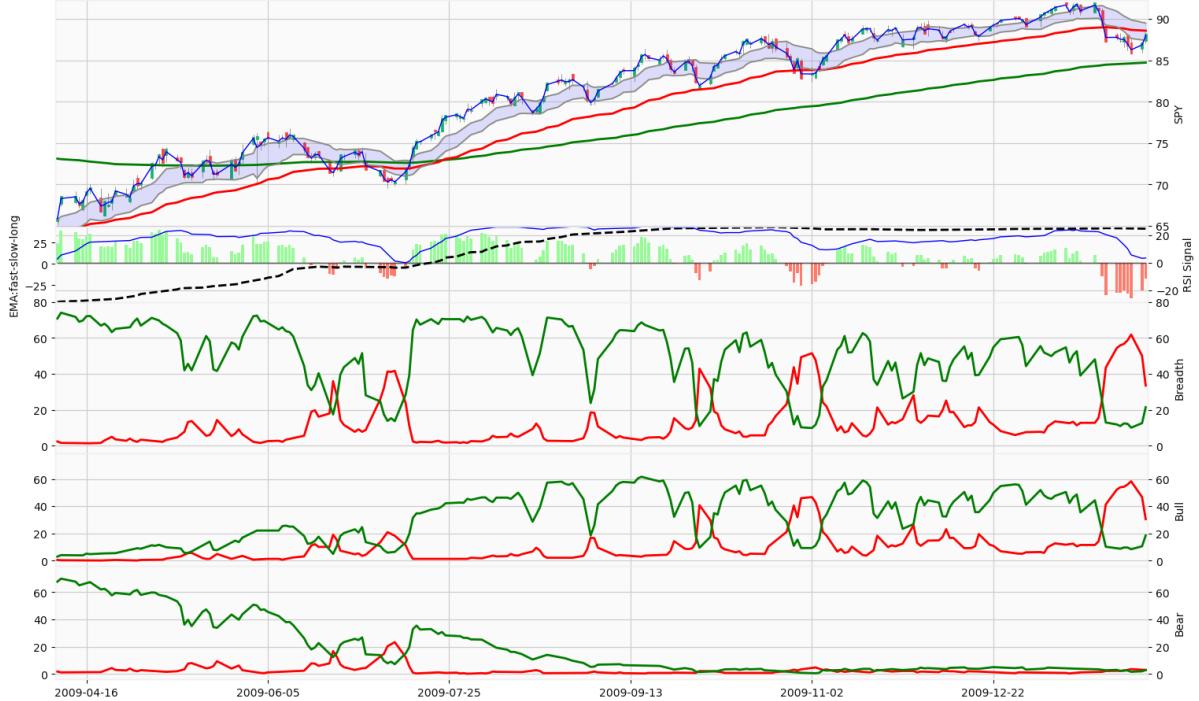




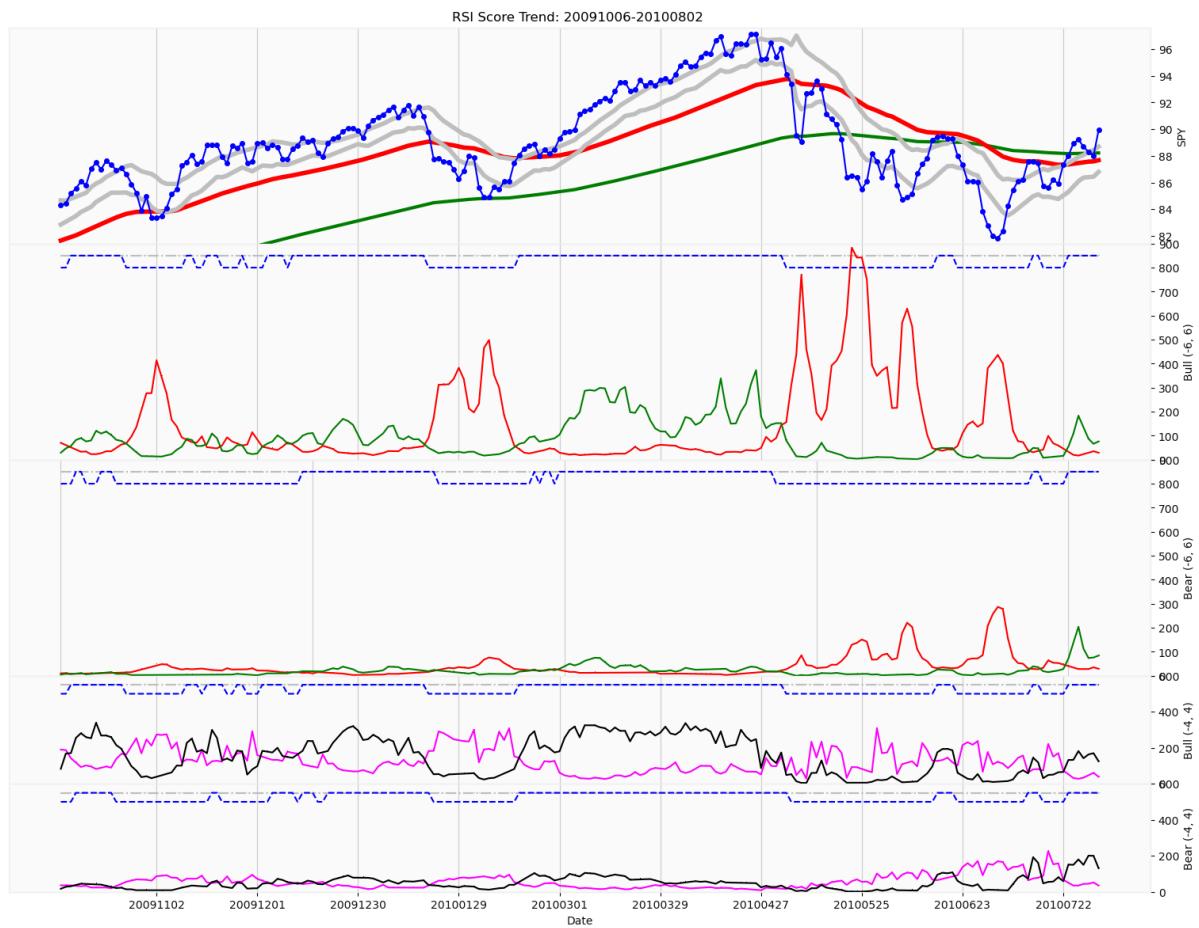




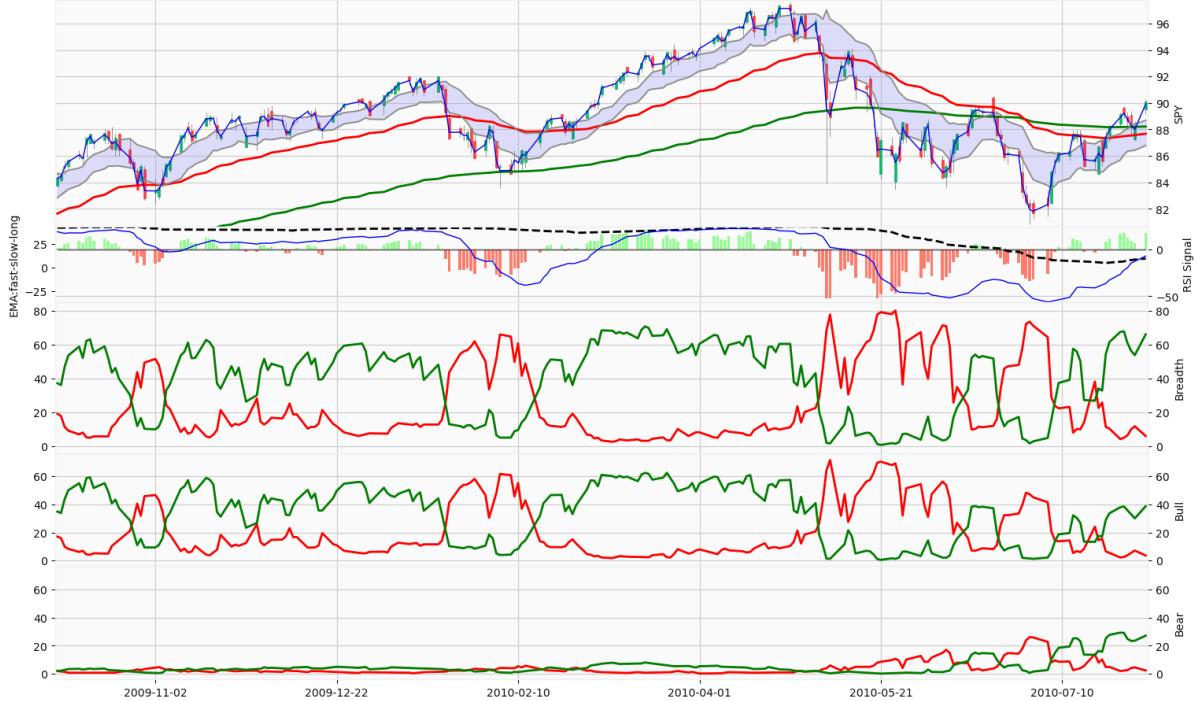
**SPY [2010-02-02] V:270\_628\_027\_093\_140 L:87.03 H:88.42 C:88.24 Wp:87.98 (1.2%) -2**

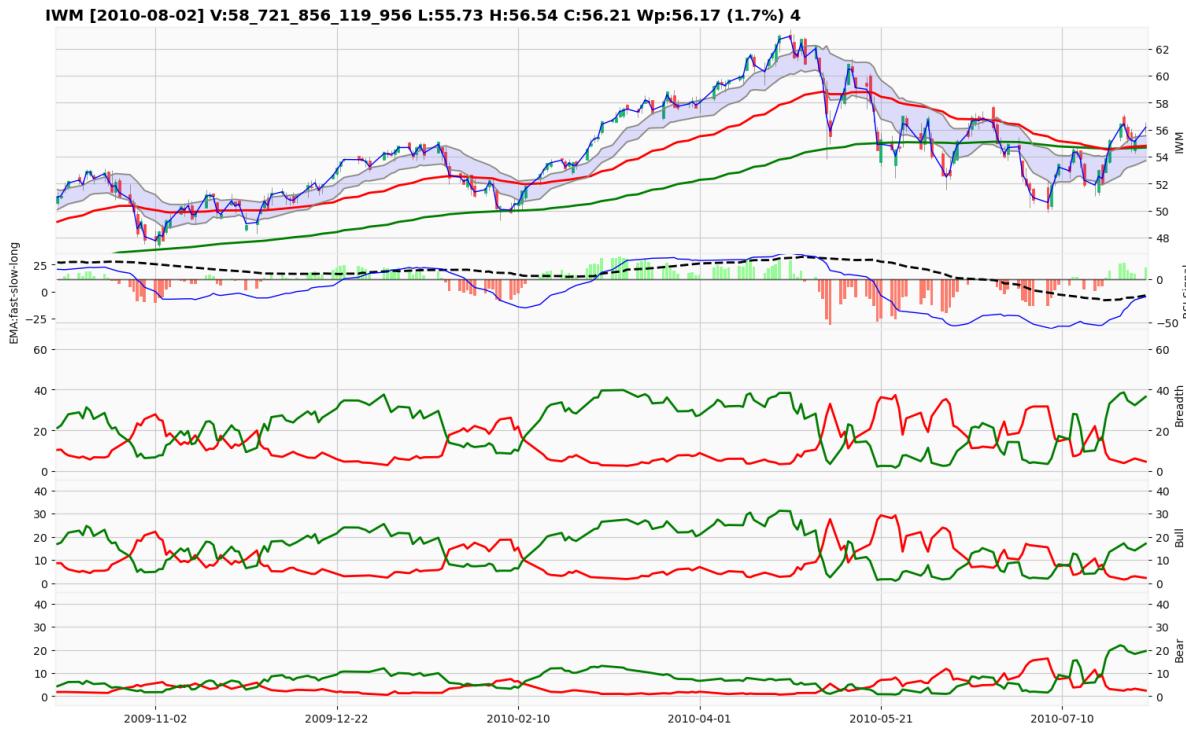
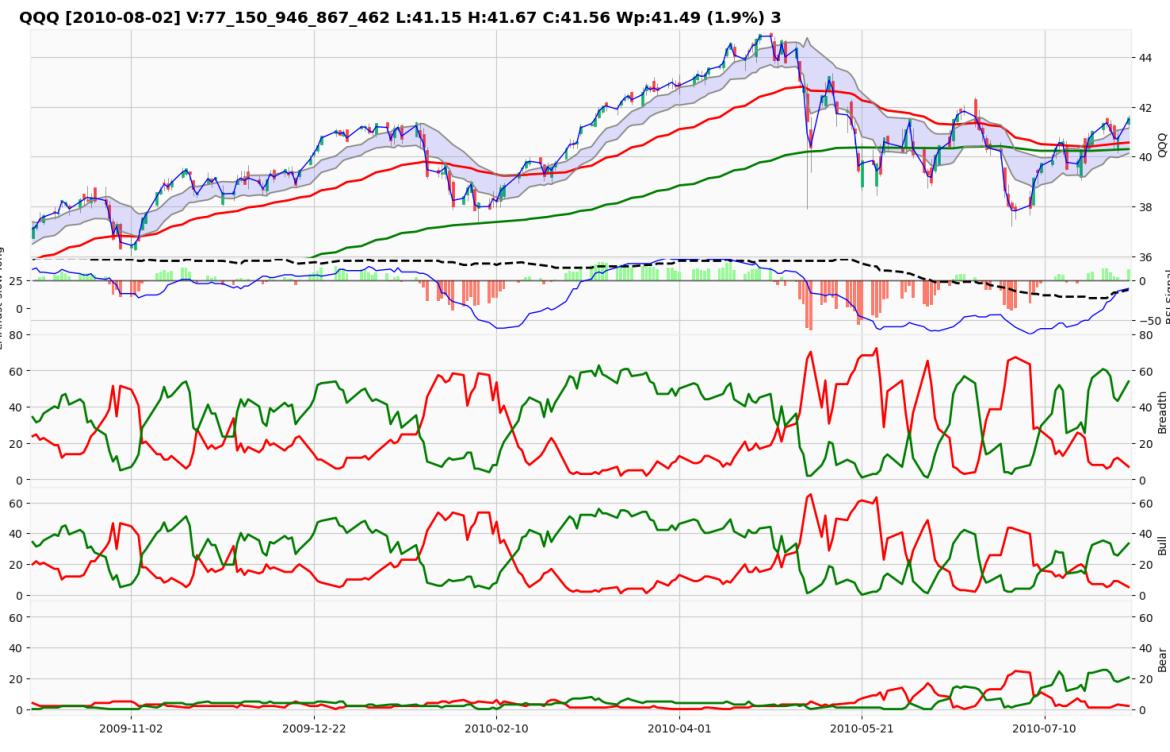


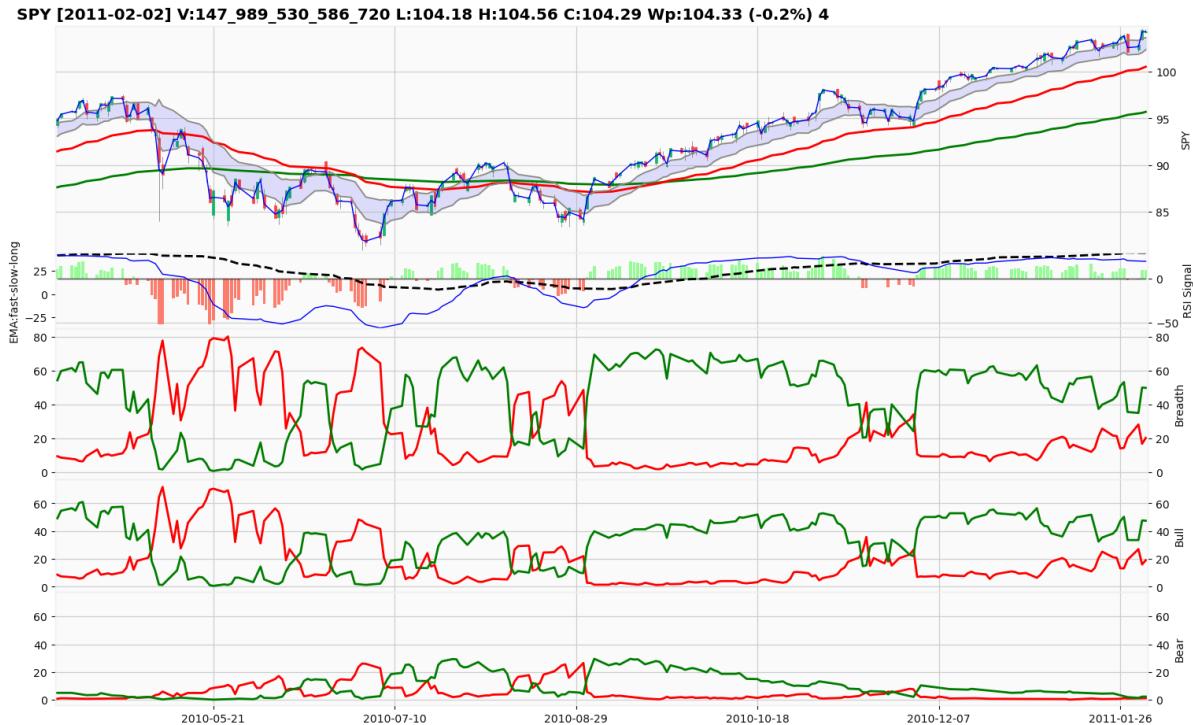
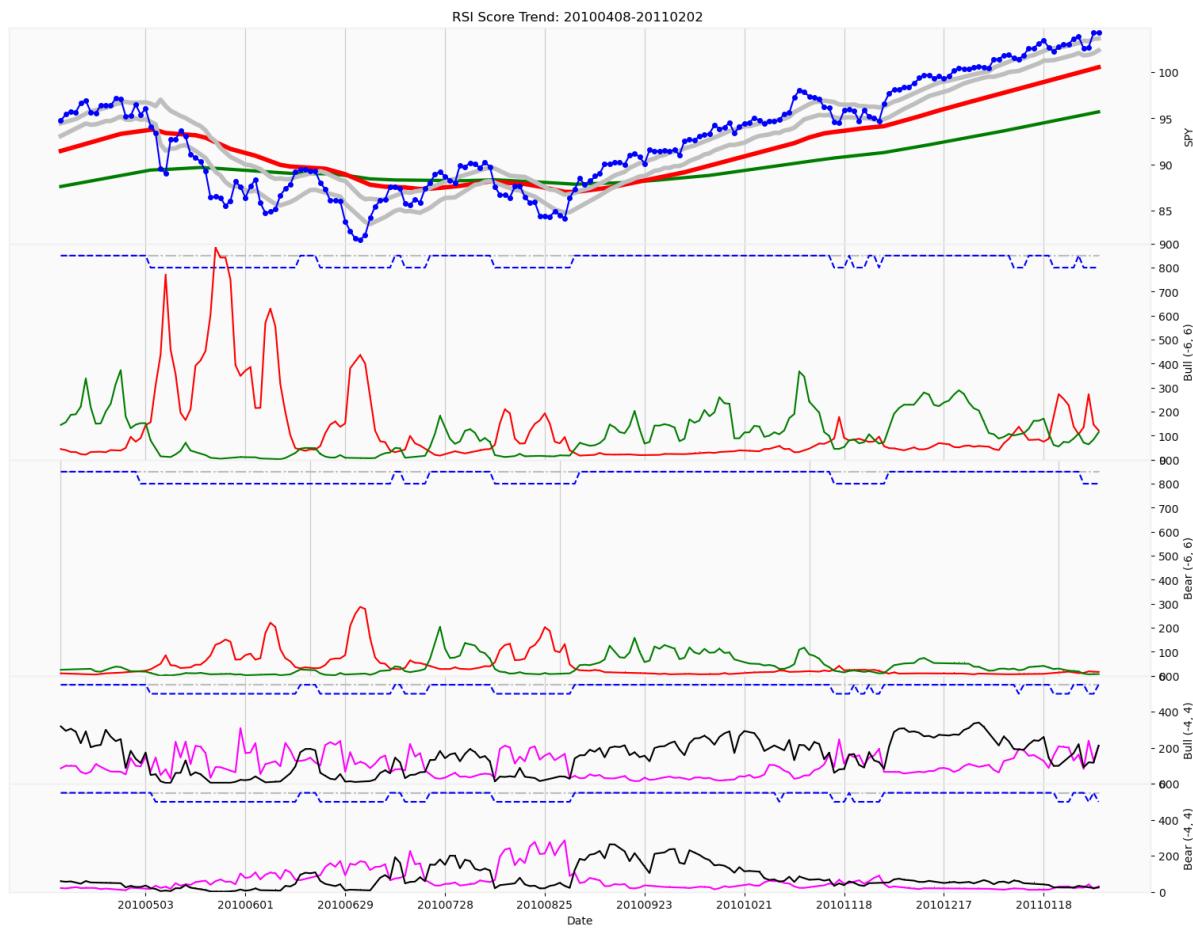


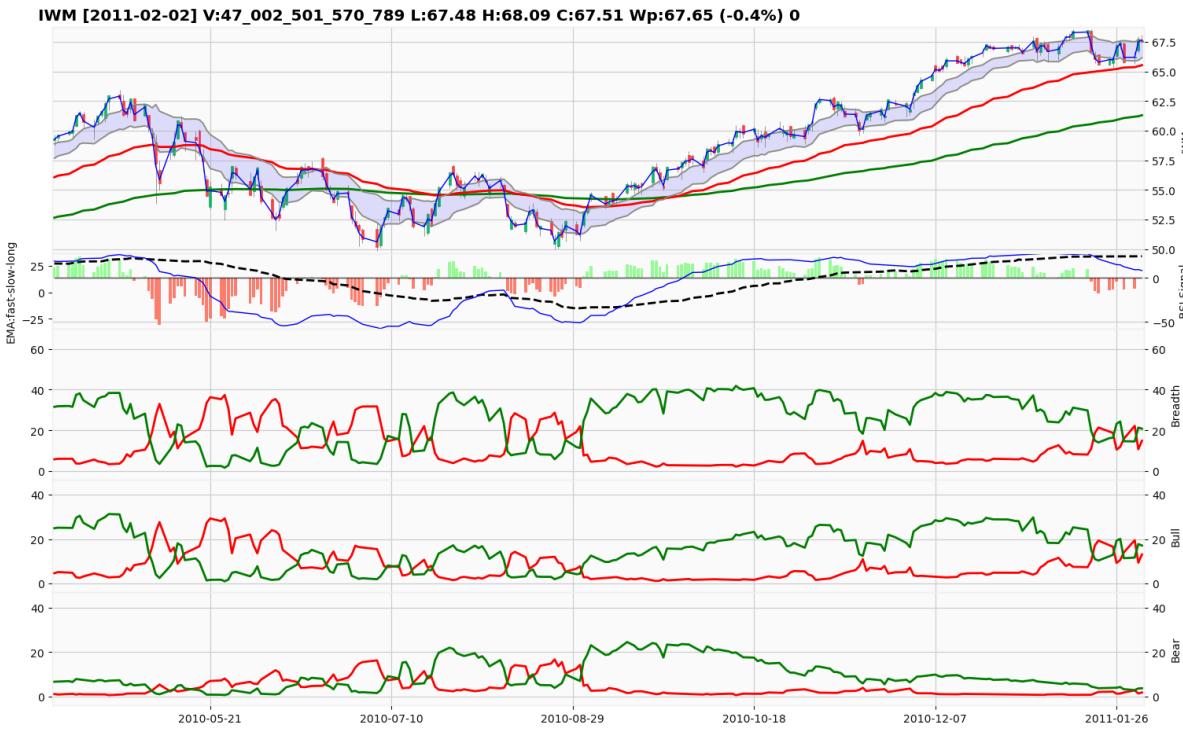
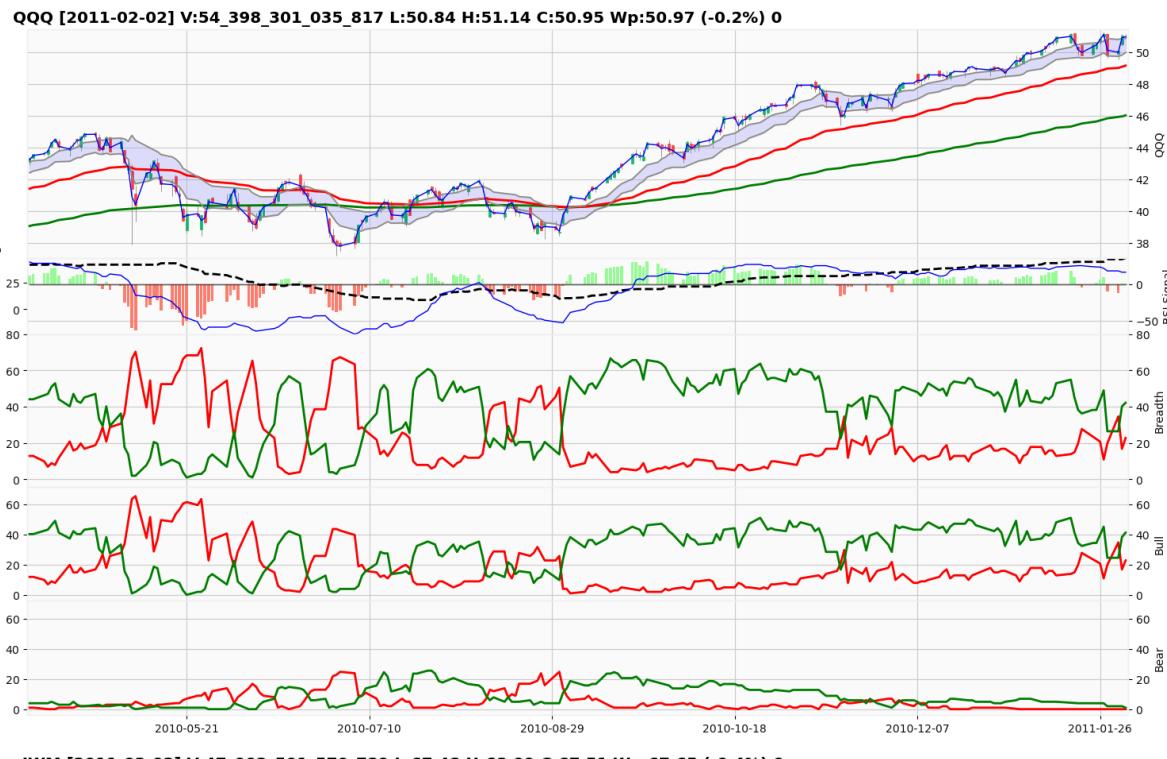


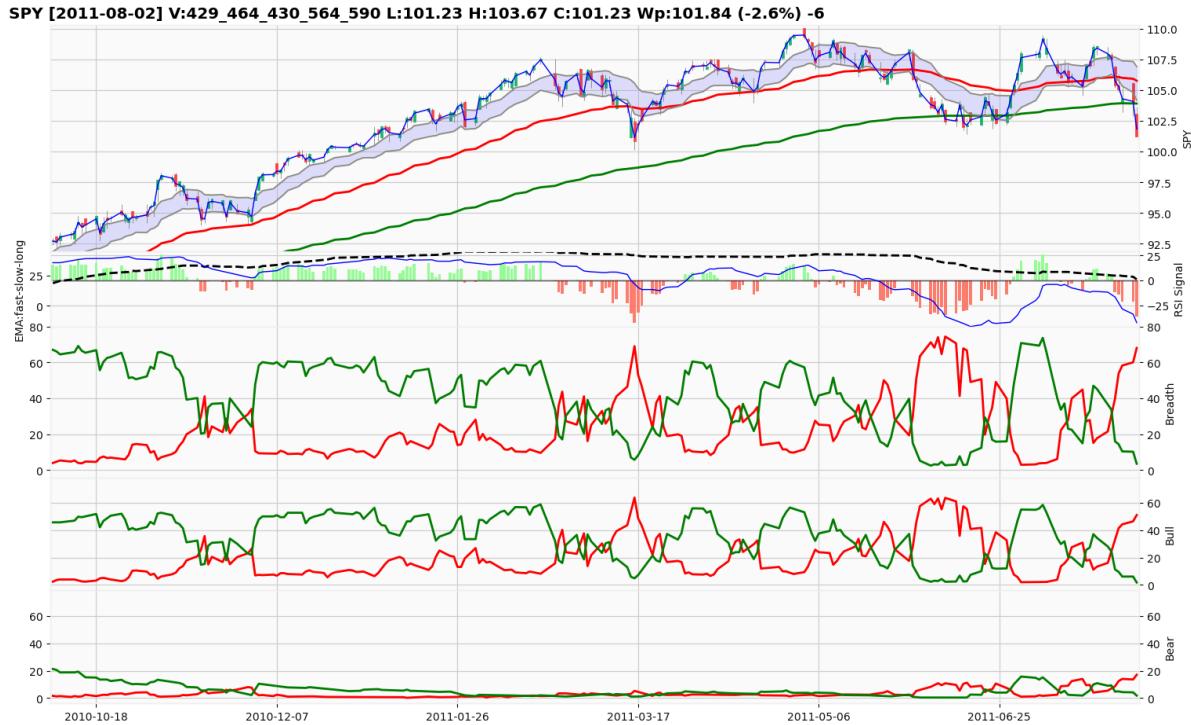
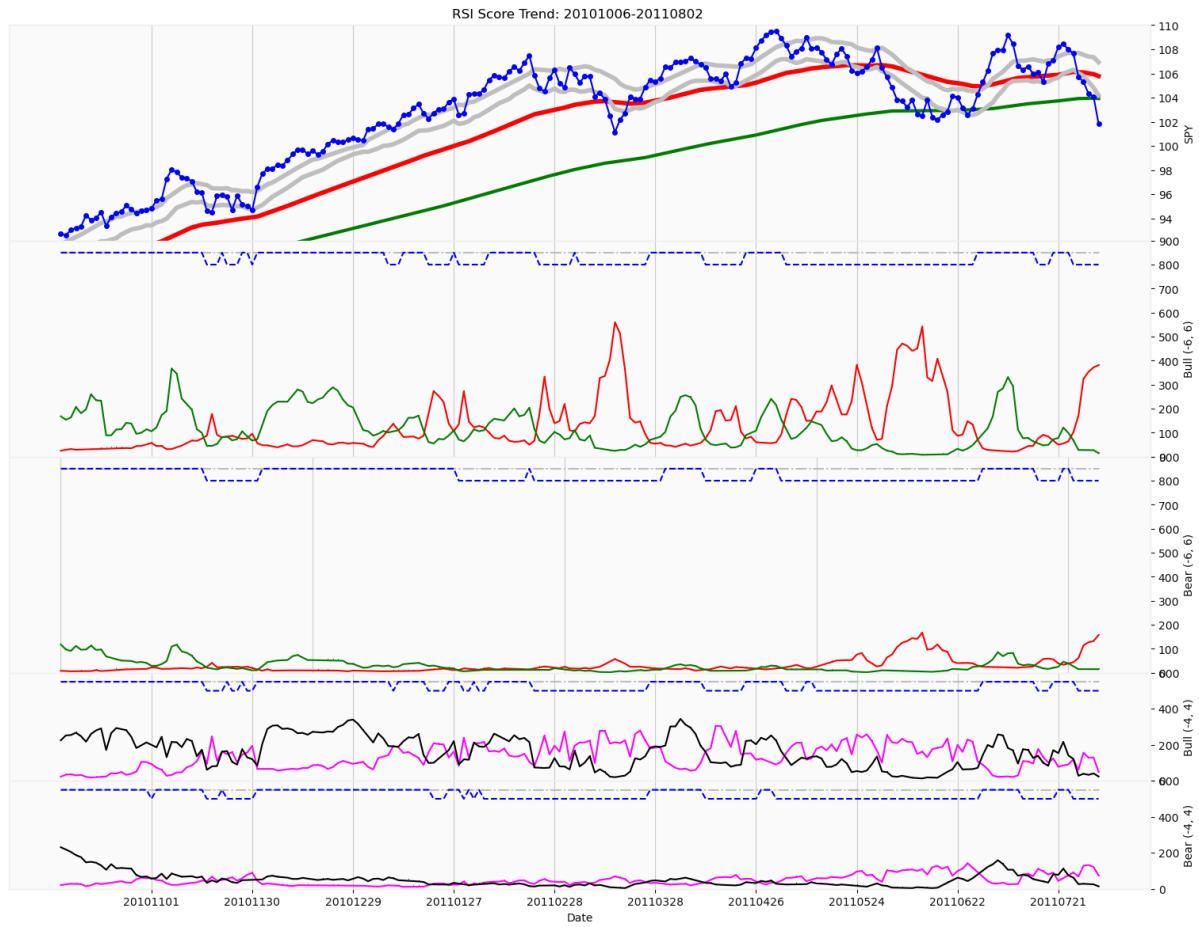
**SPY [2010-08-02] V:235\_518\_905\_208\_980 L:89.14 H:90.29 C:90.13 Wp:89.92 (2.2%) 5**

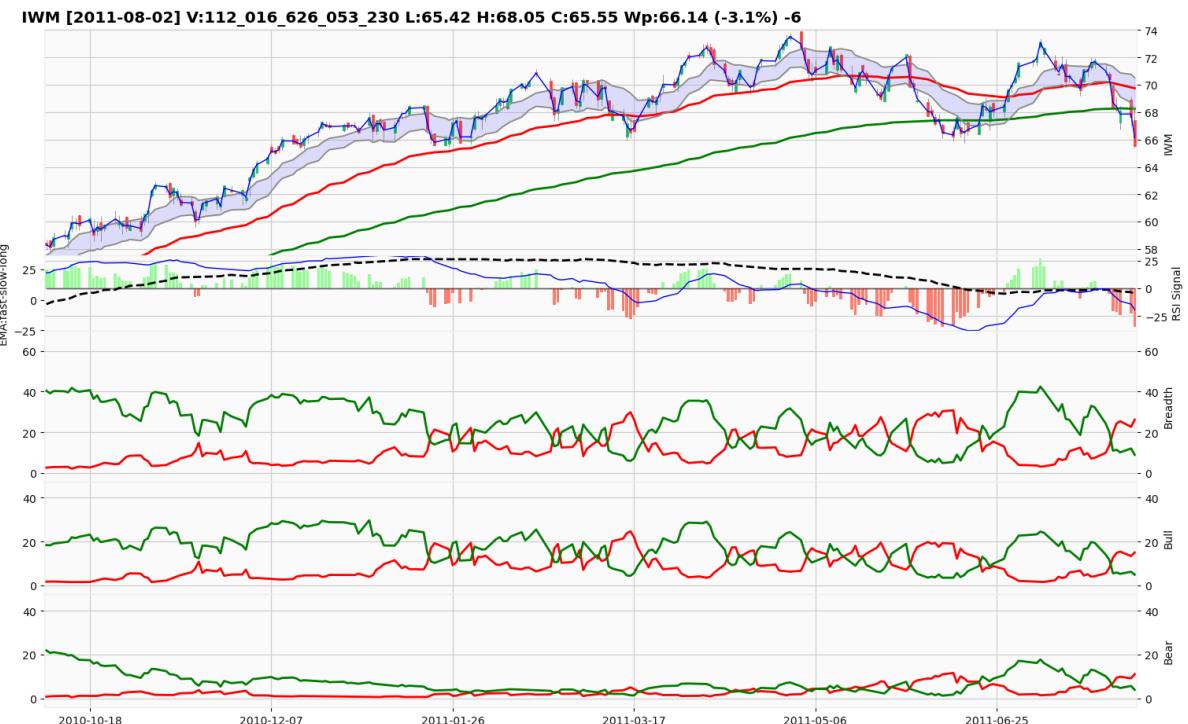
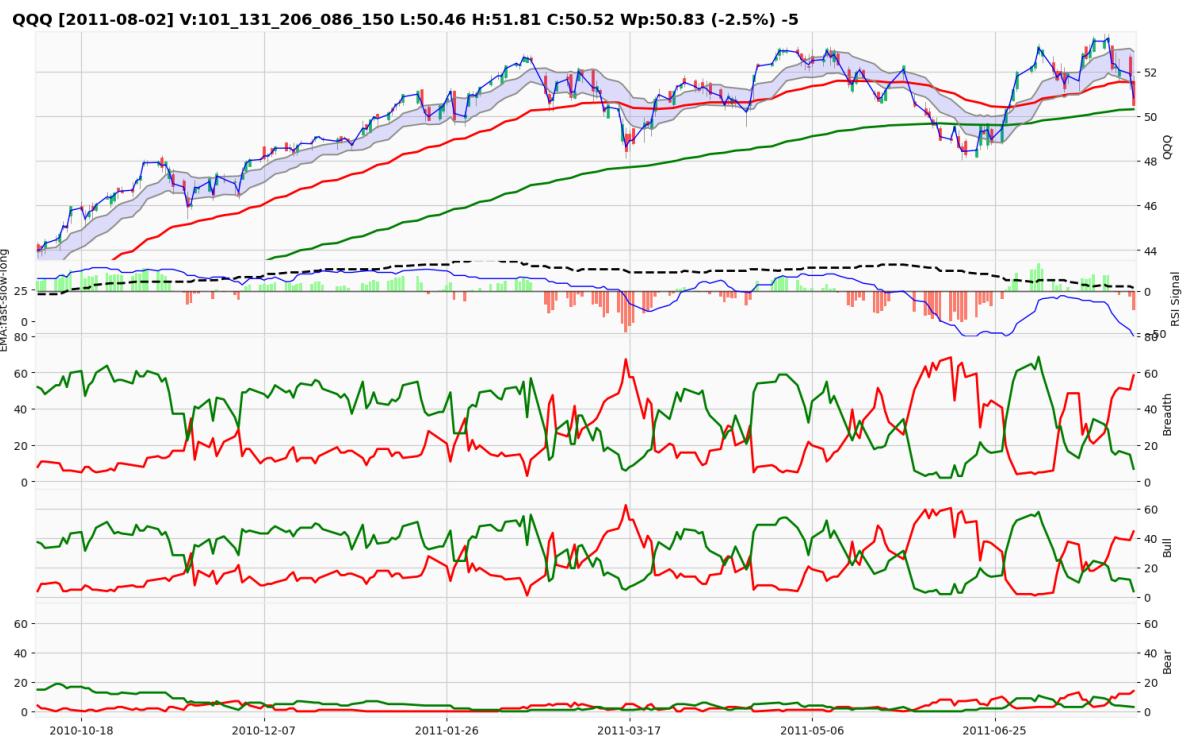


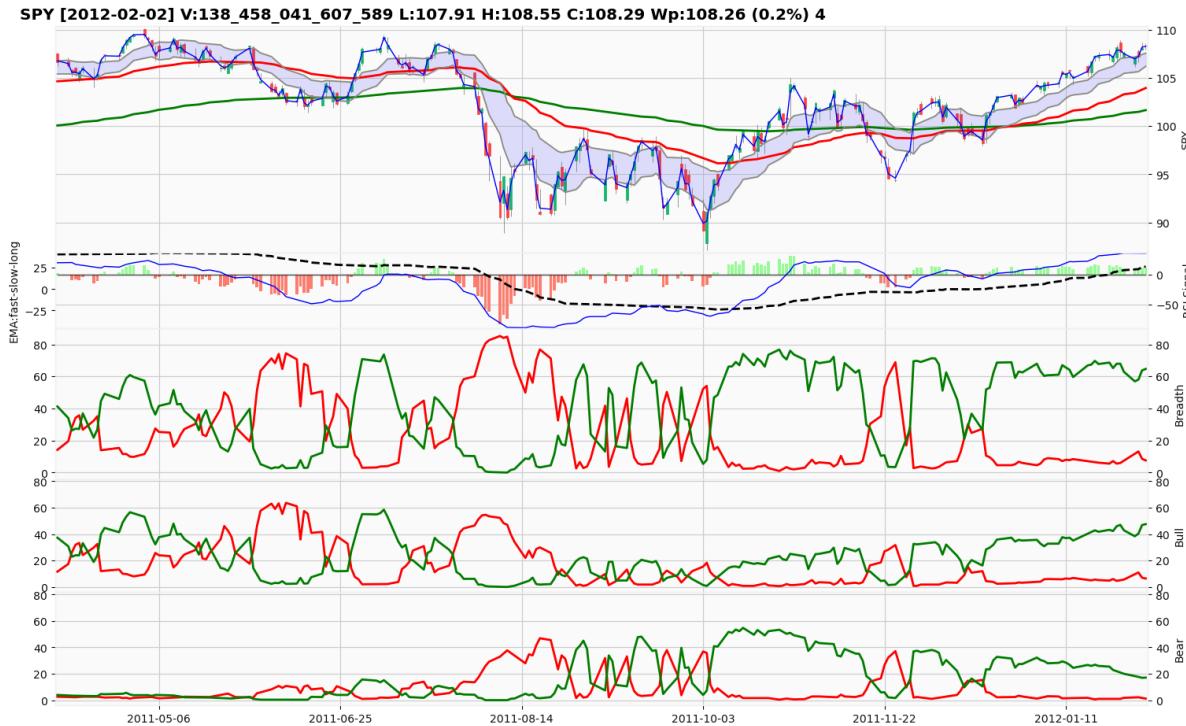
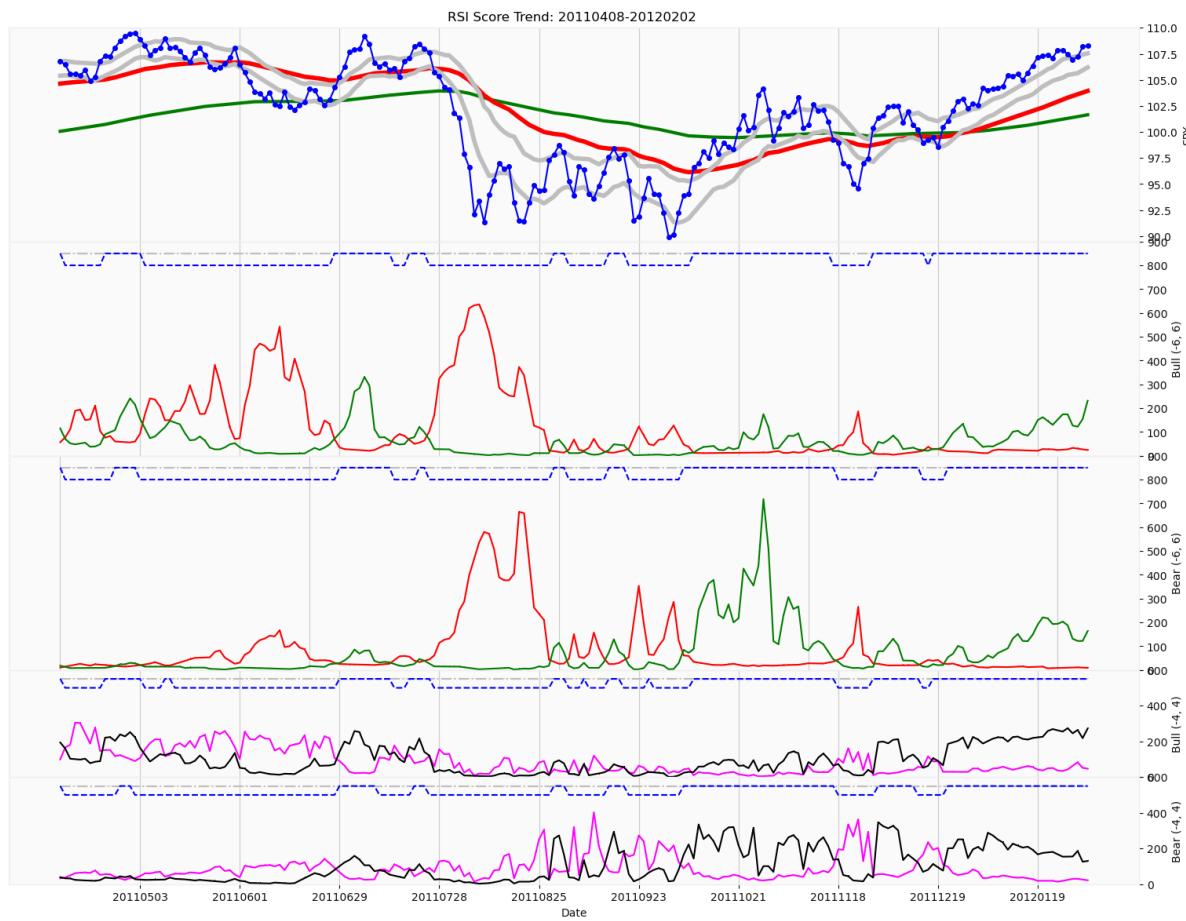


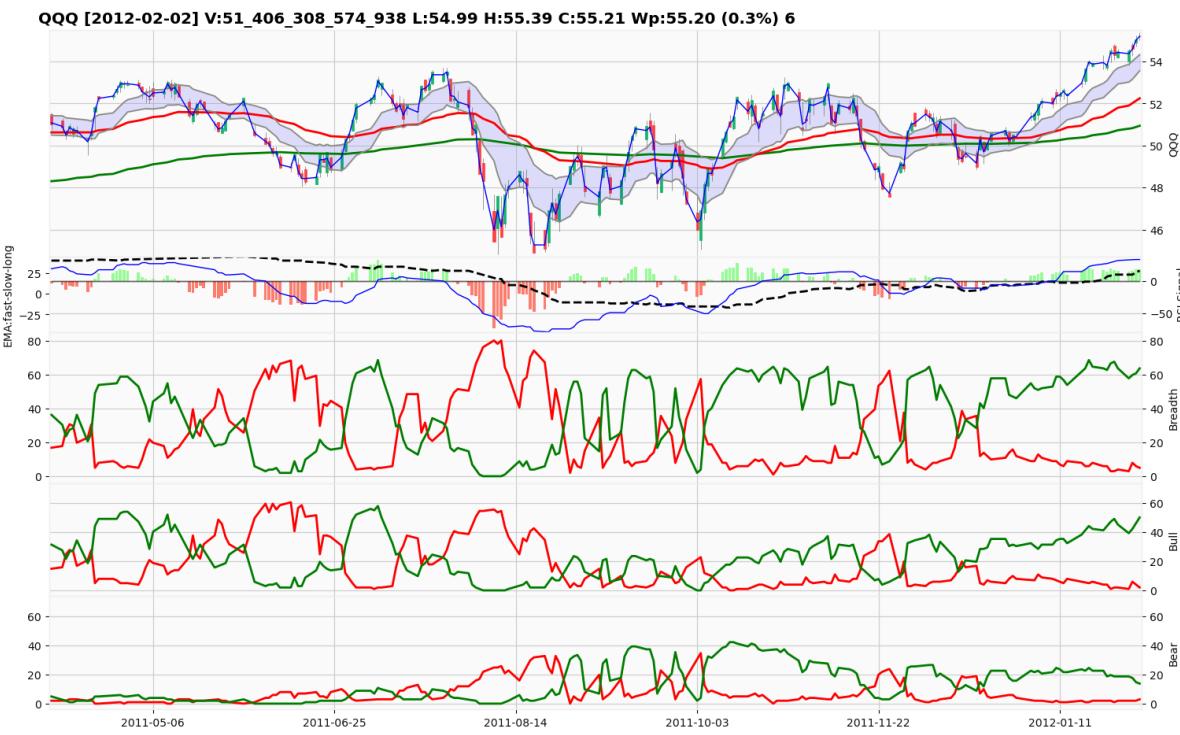


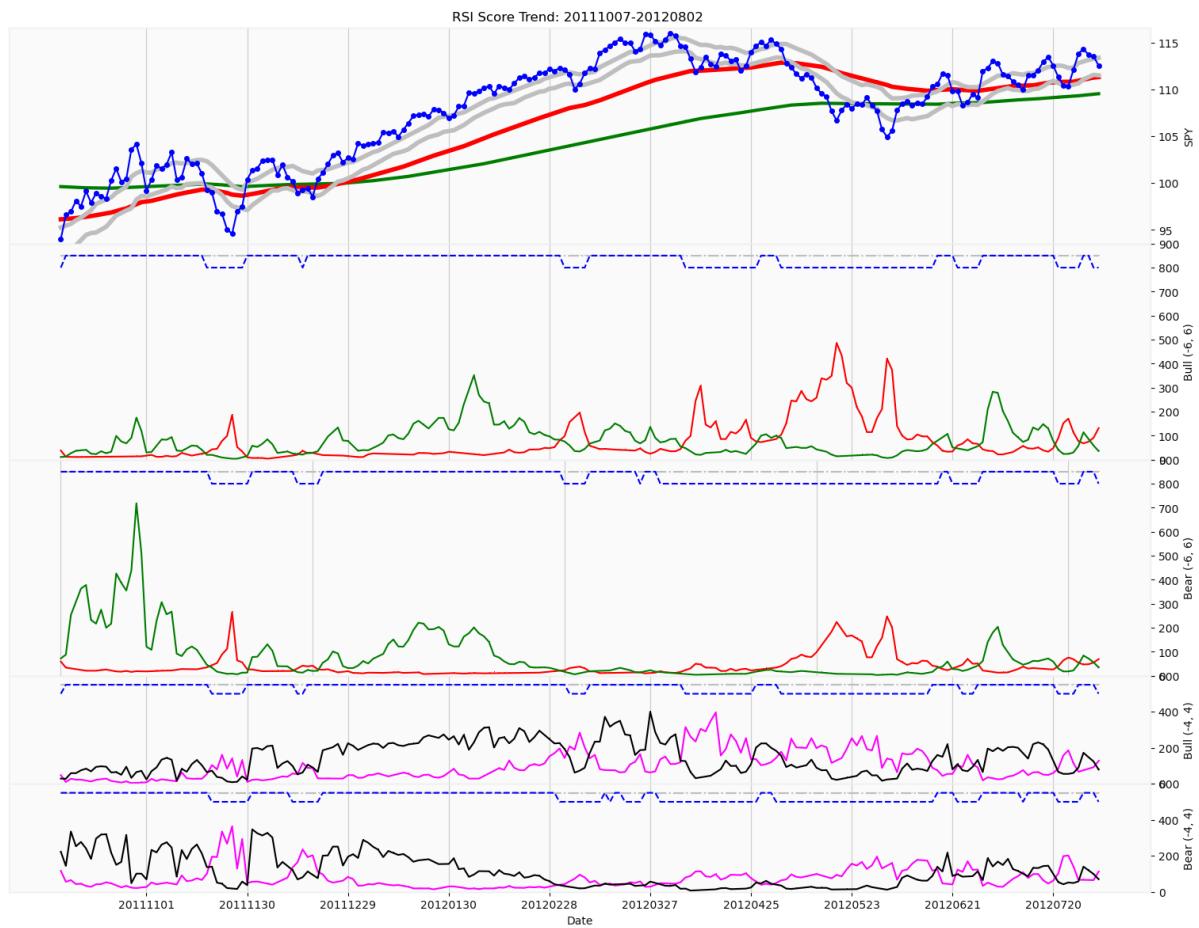




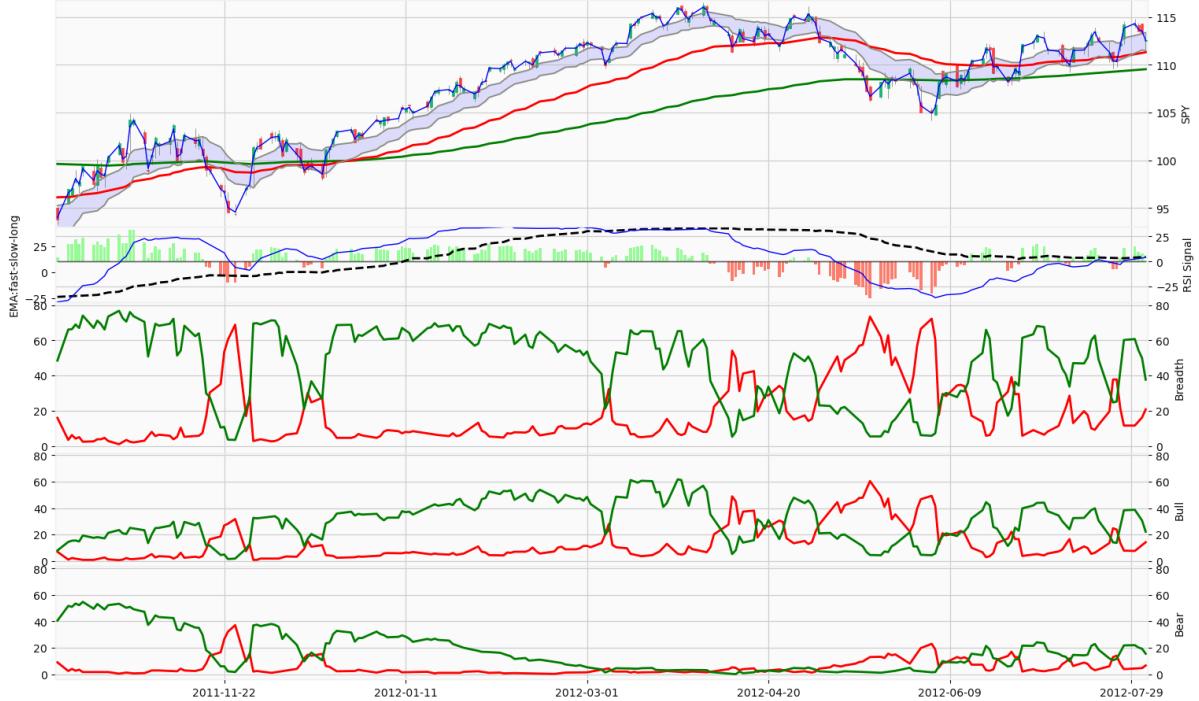


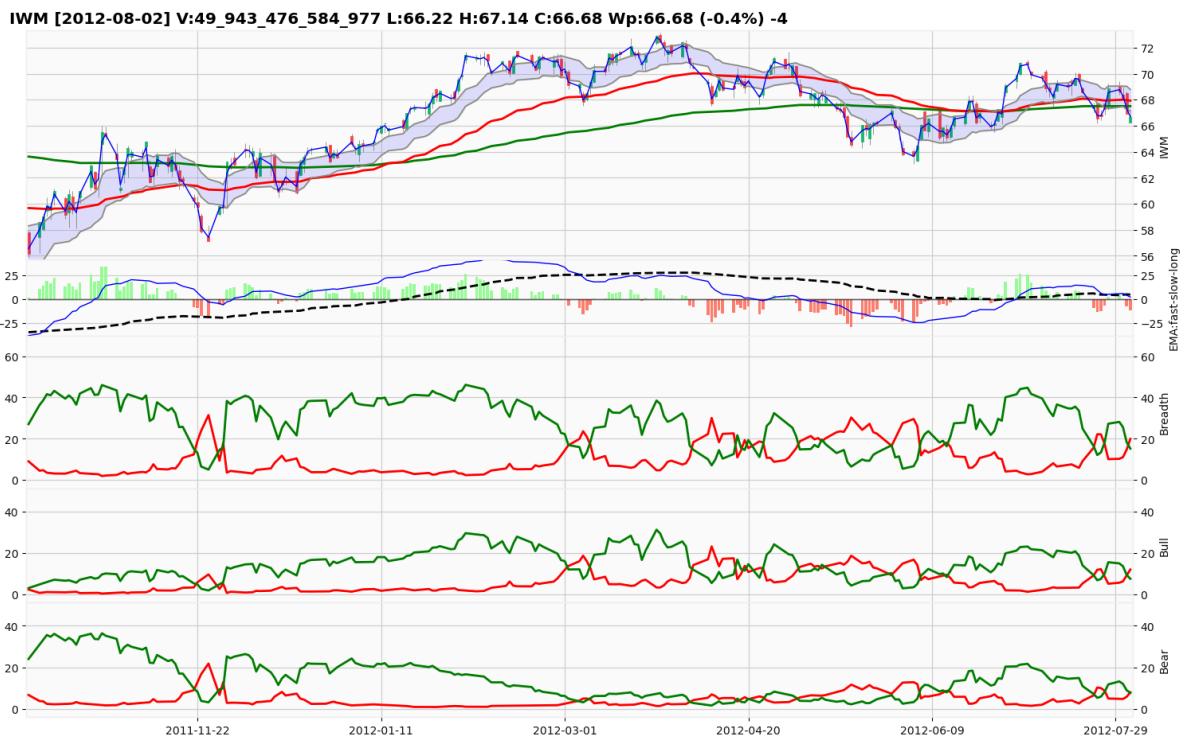
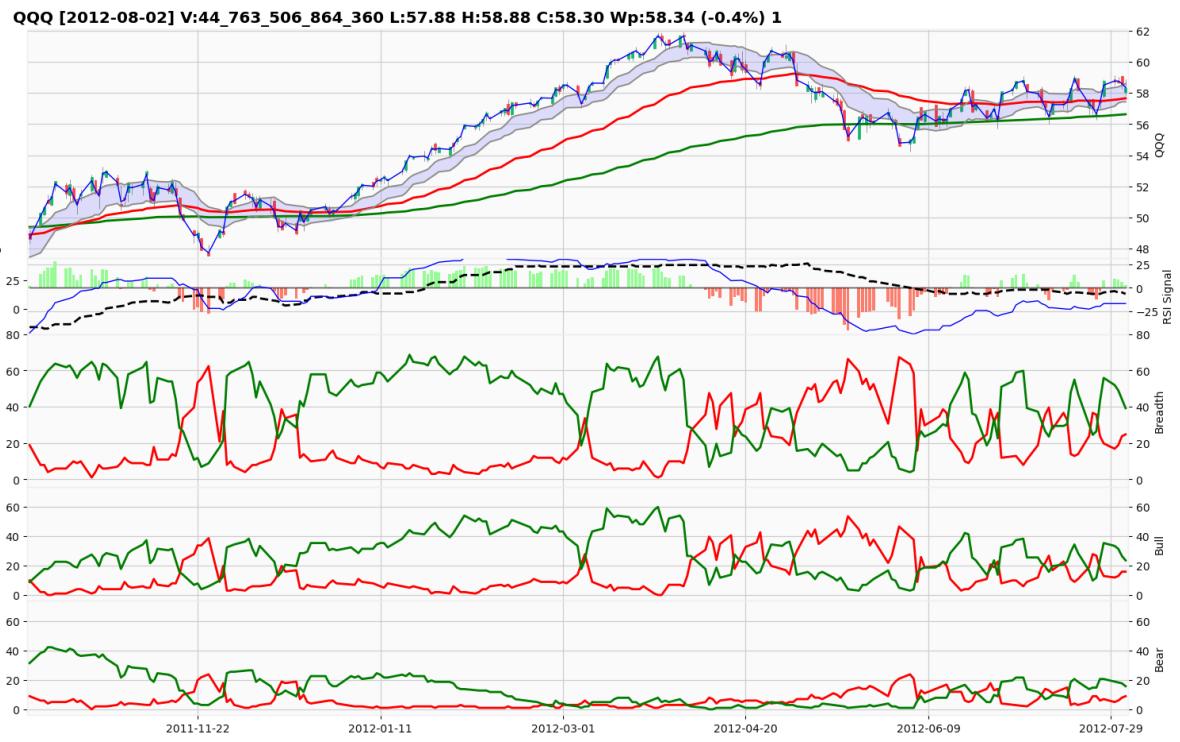


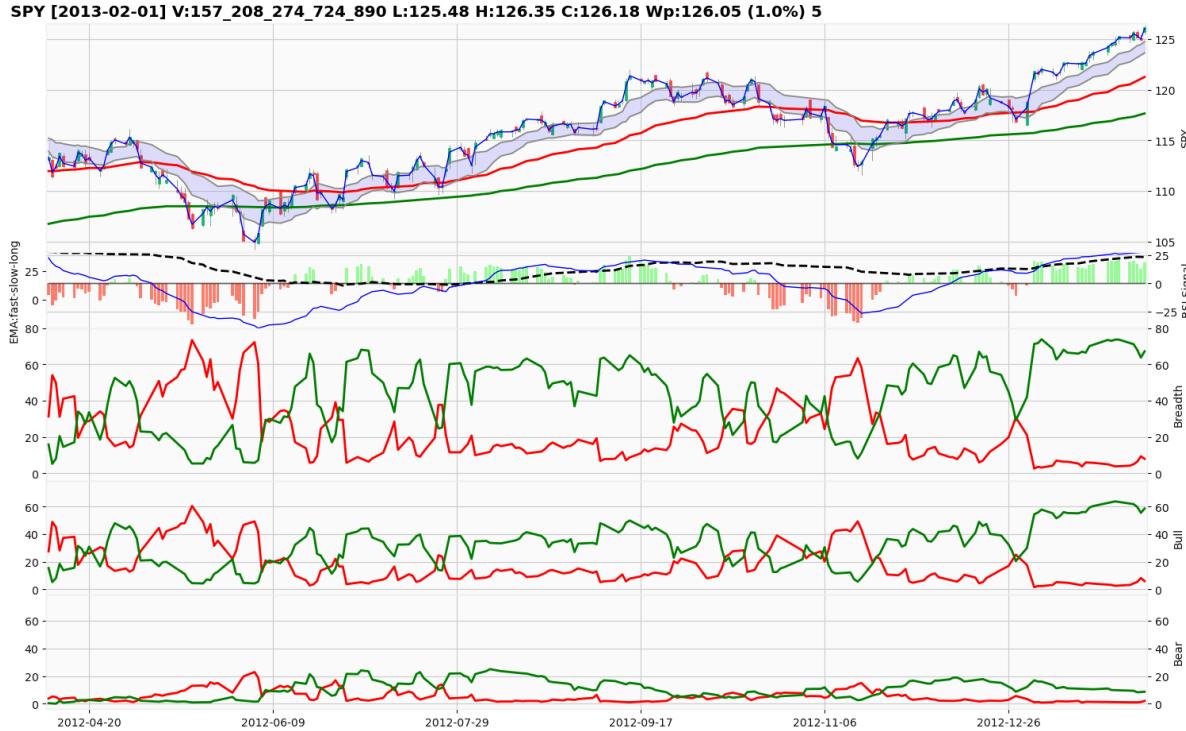
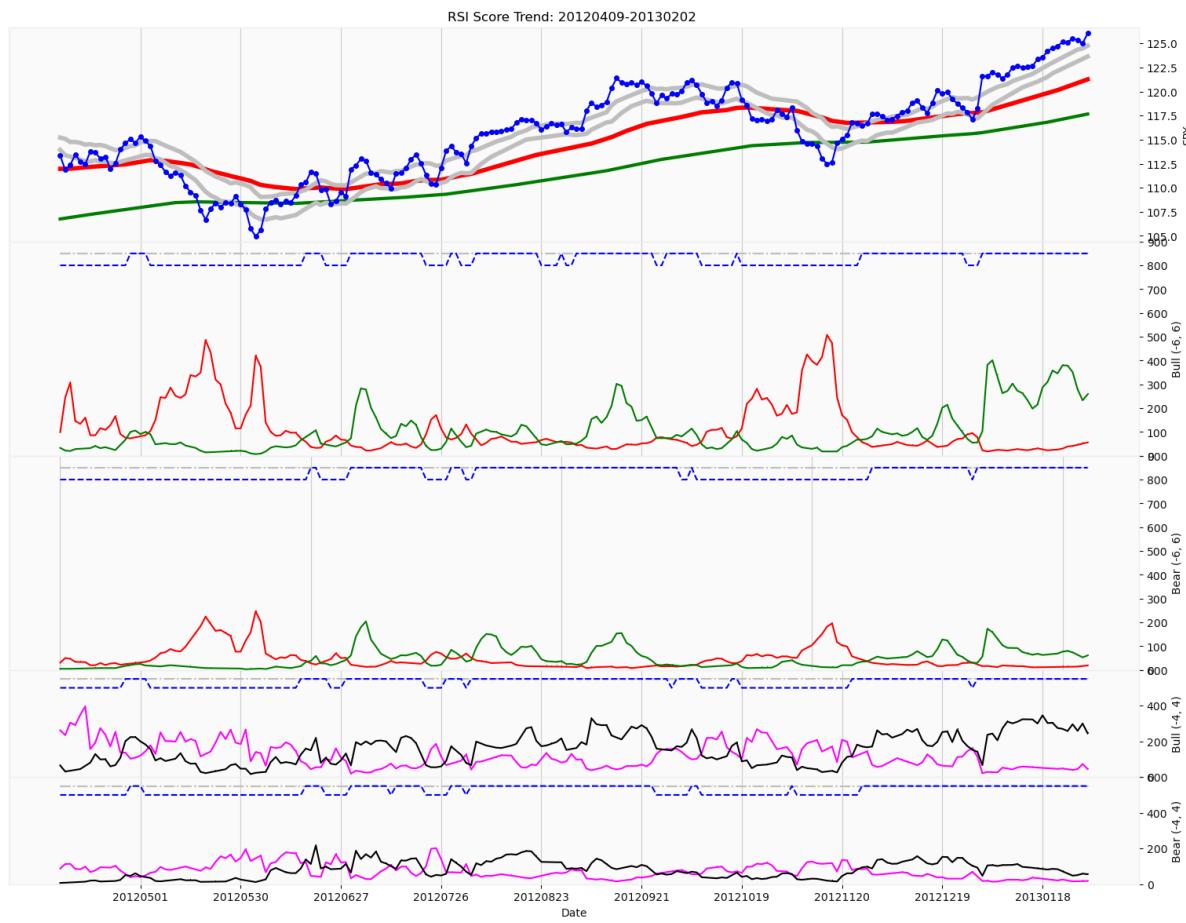


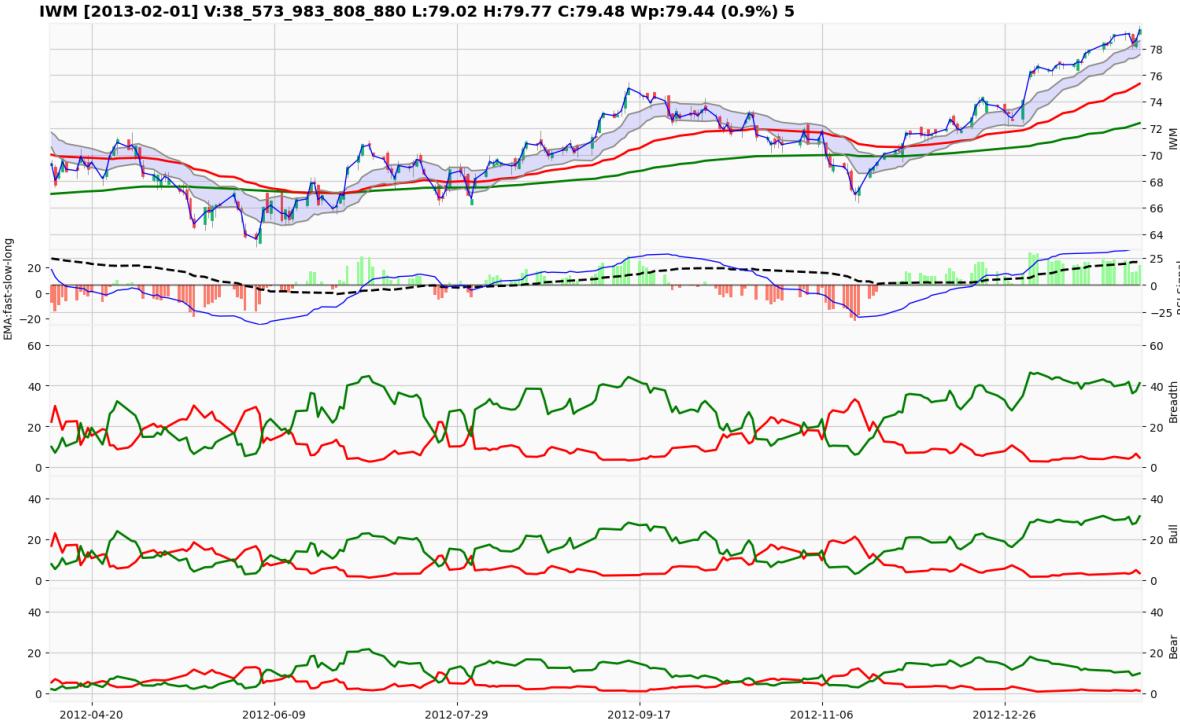
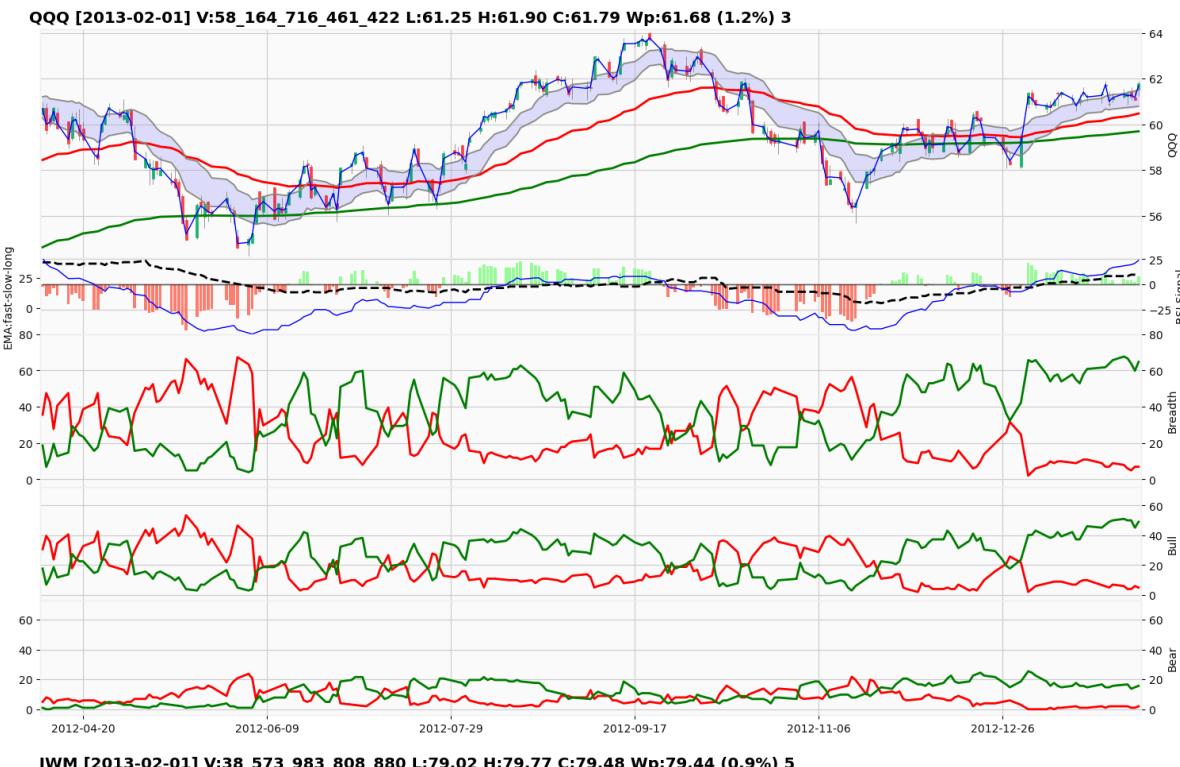


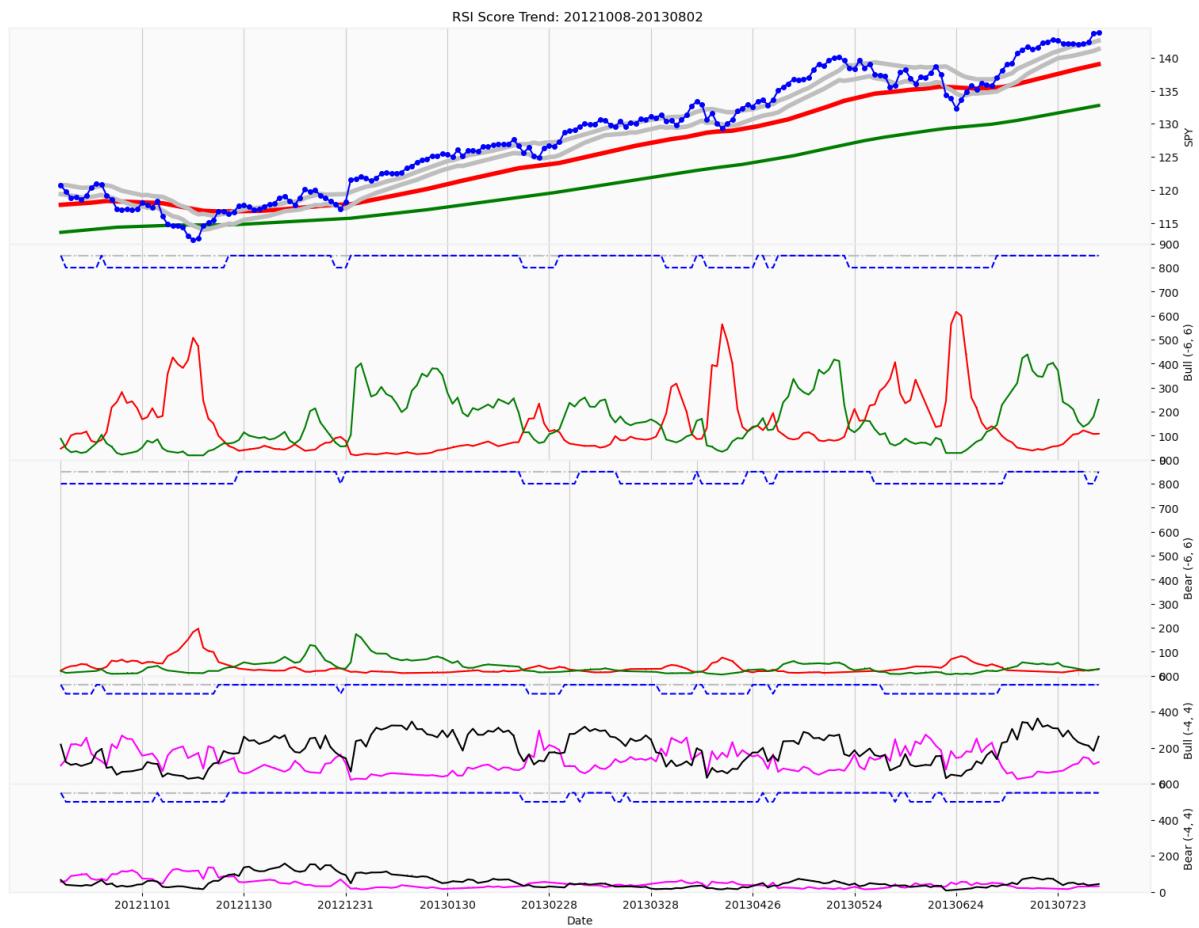
**SPY [2012-08-02] V:242\_109\_027\_547\_200 L:111.72 H:113.33 C:112.59 Wp:112.56 (-0.7%) 1**



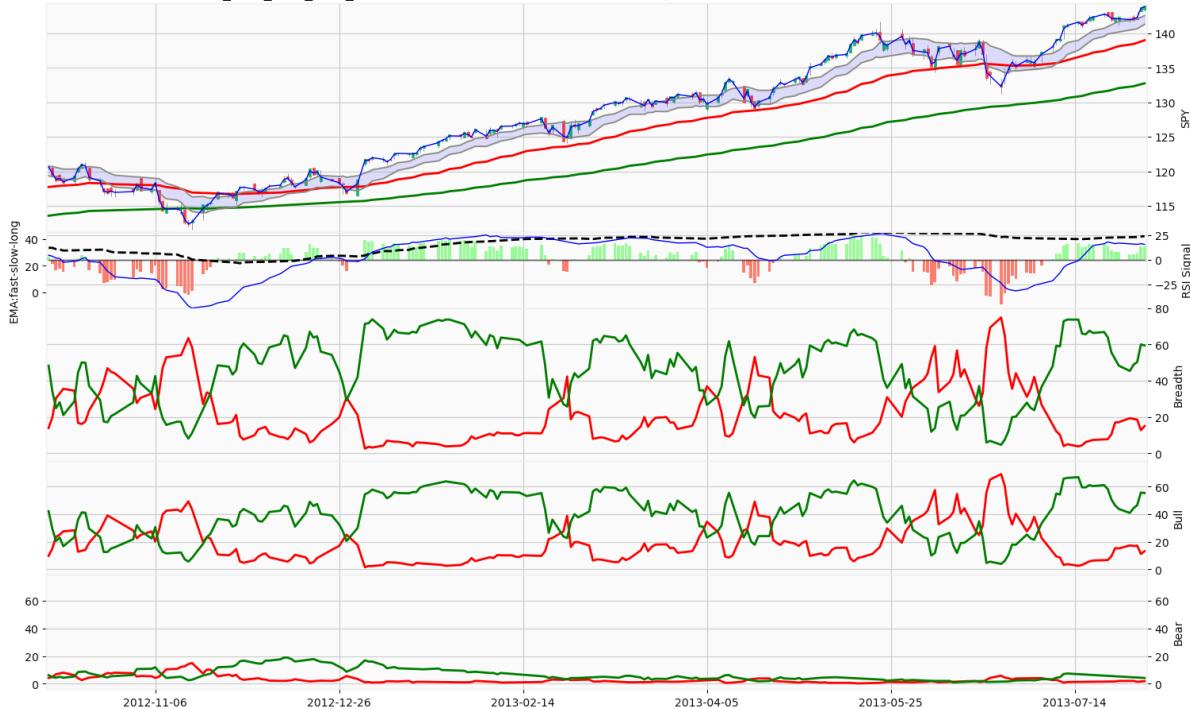


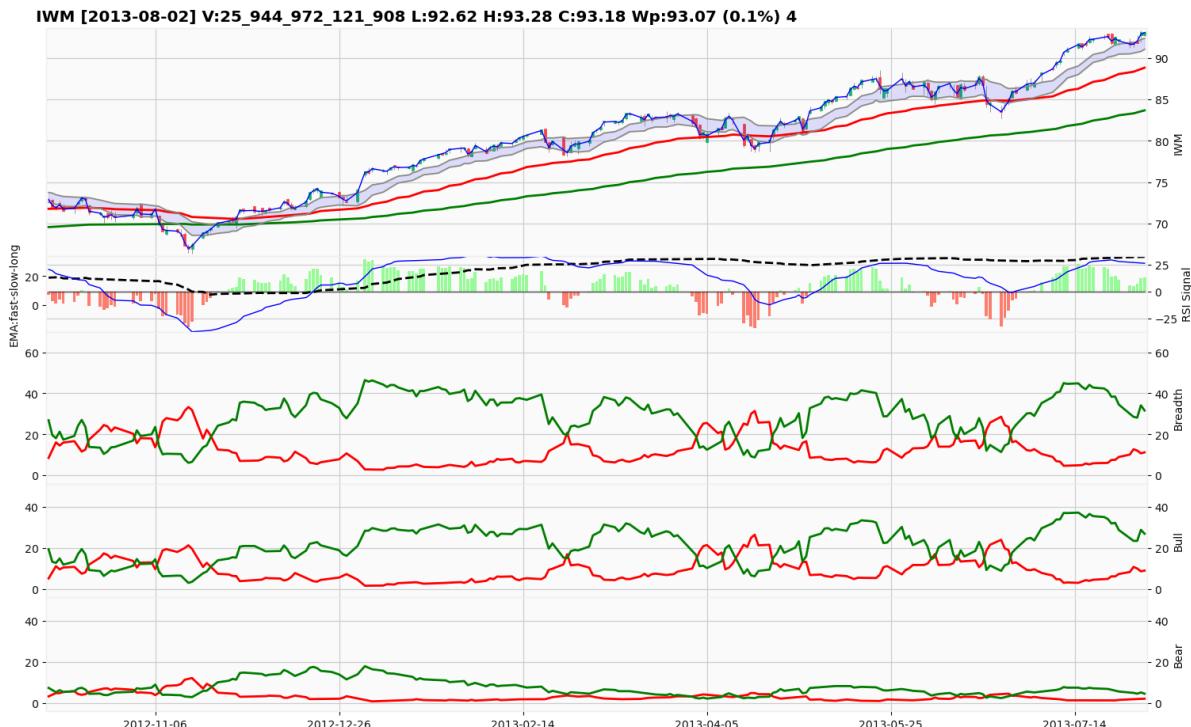
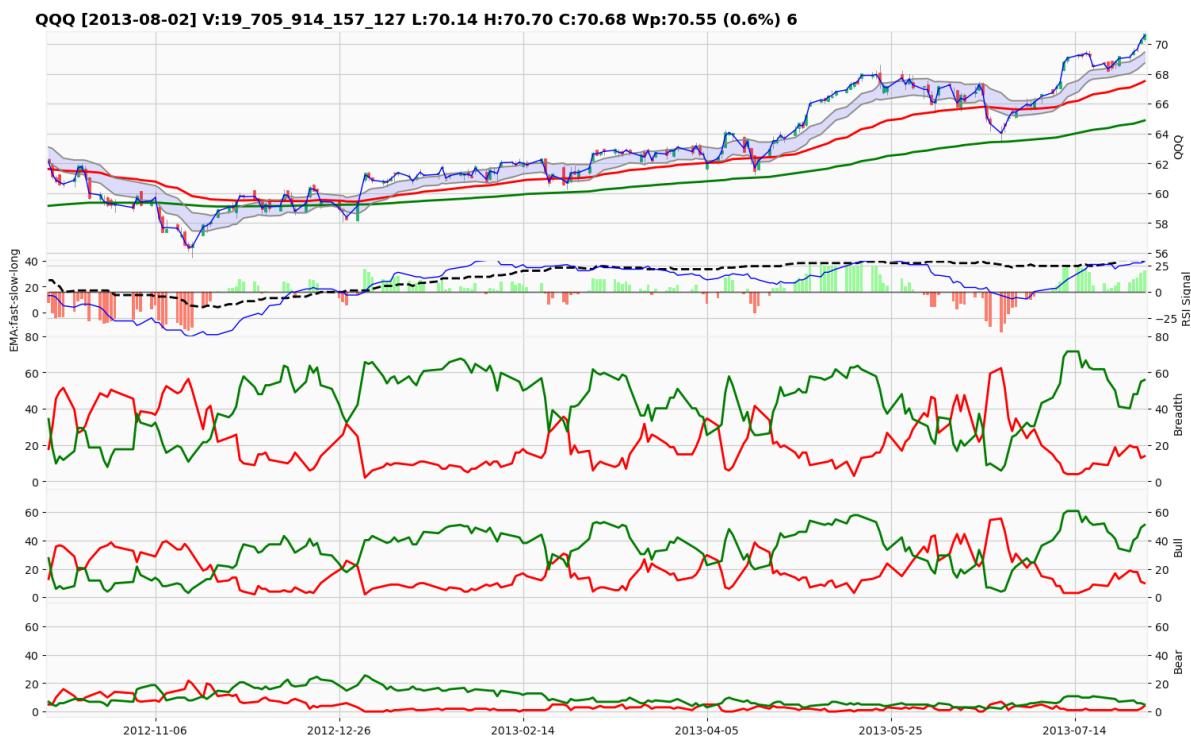


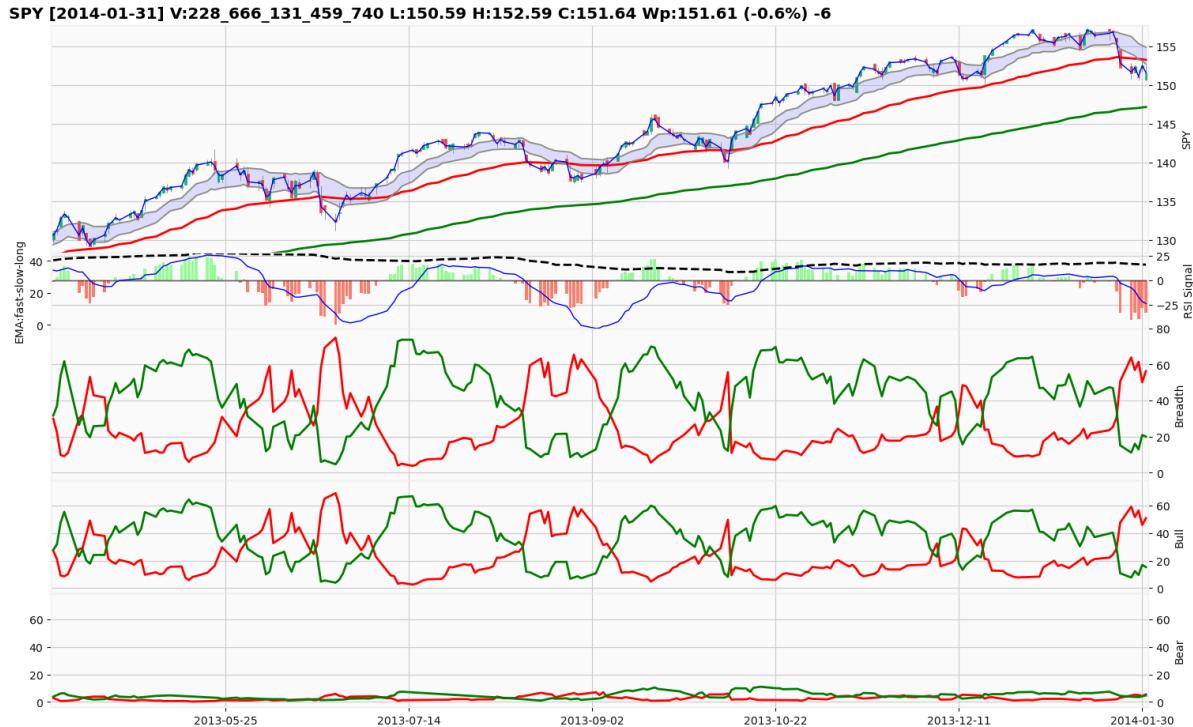
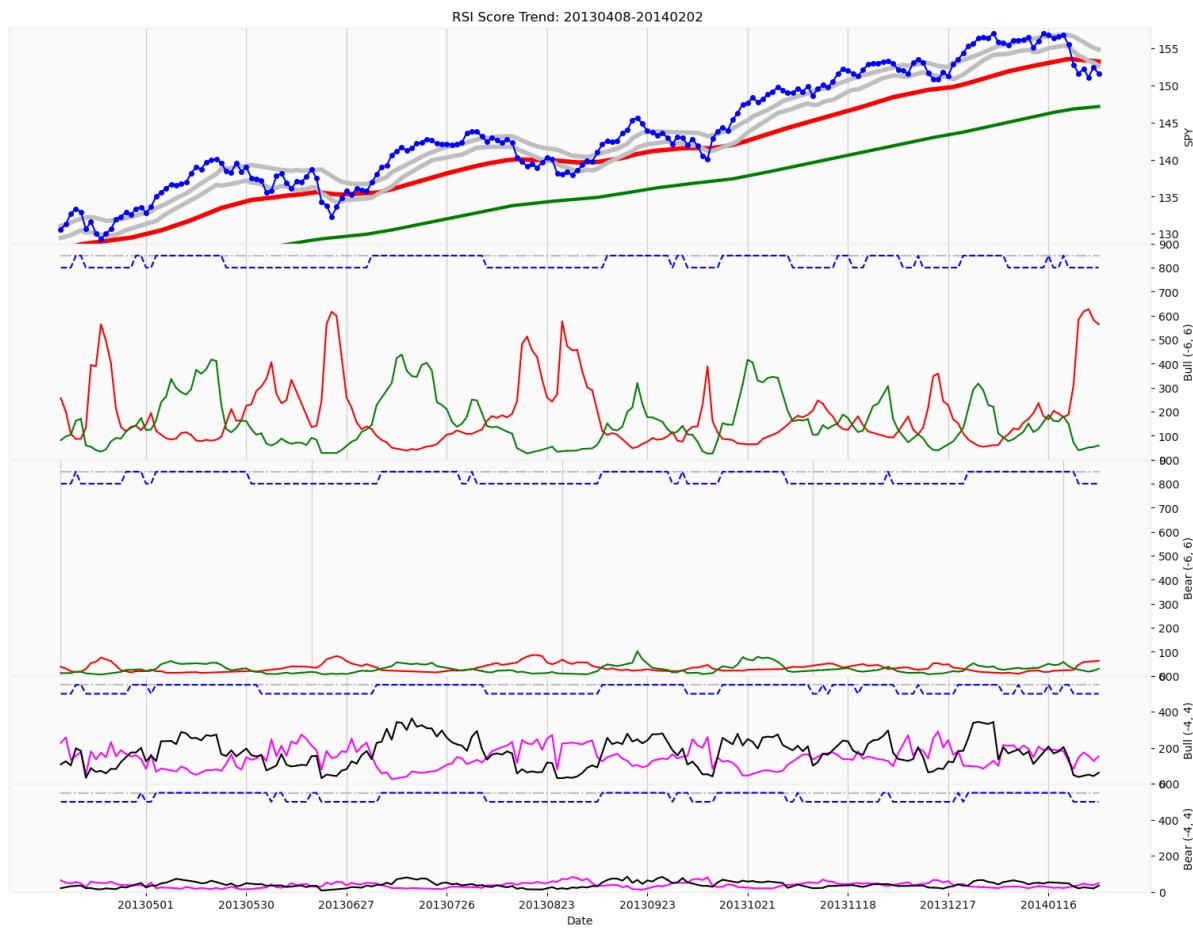




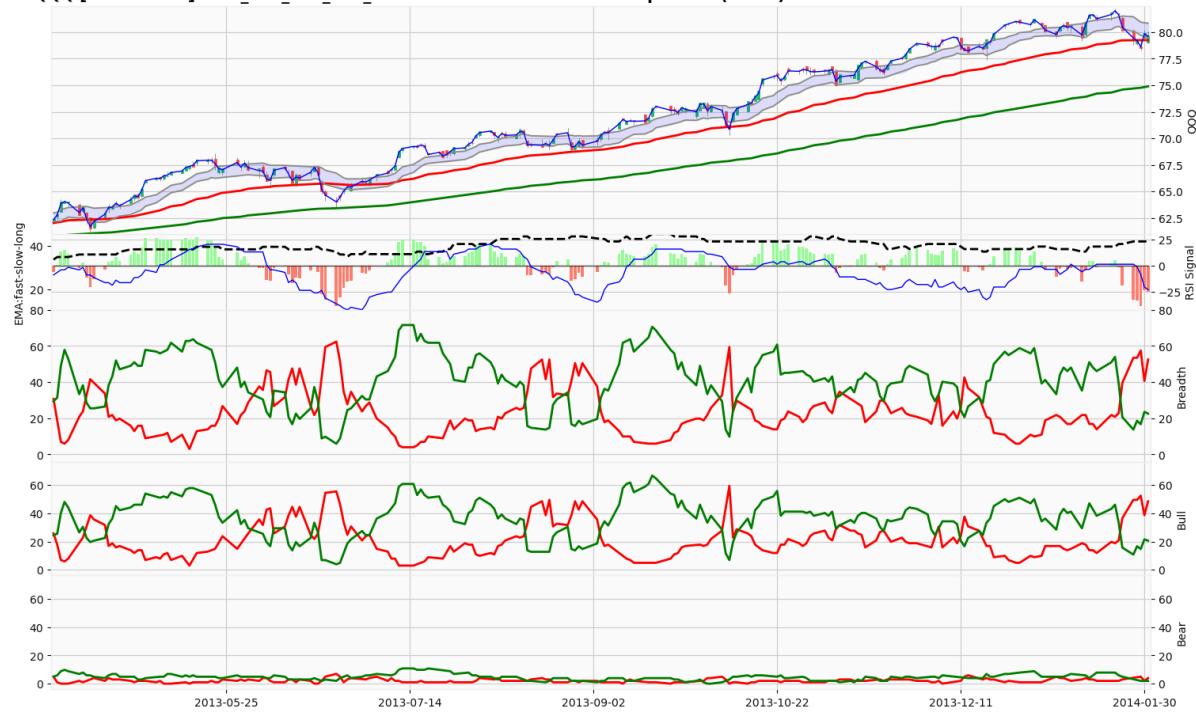
**SPY [2013-08-02] V:108\_088\_581\_301\_340 L:143.26 H:144.02 C:144.00 Wp:143.82 (0.2%) 4**



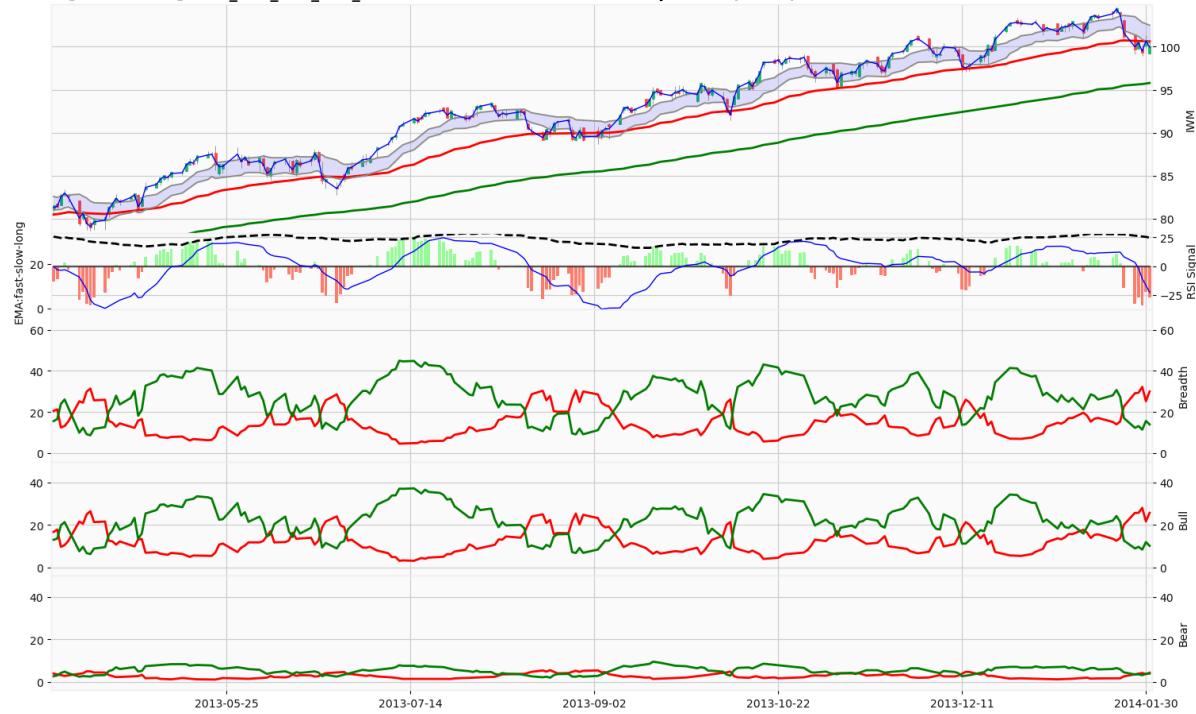


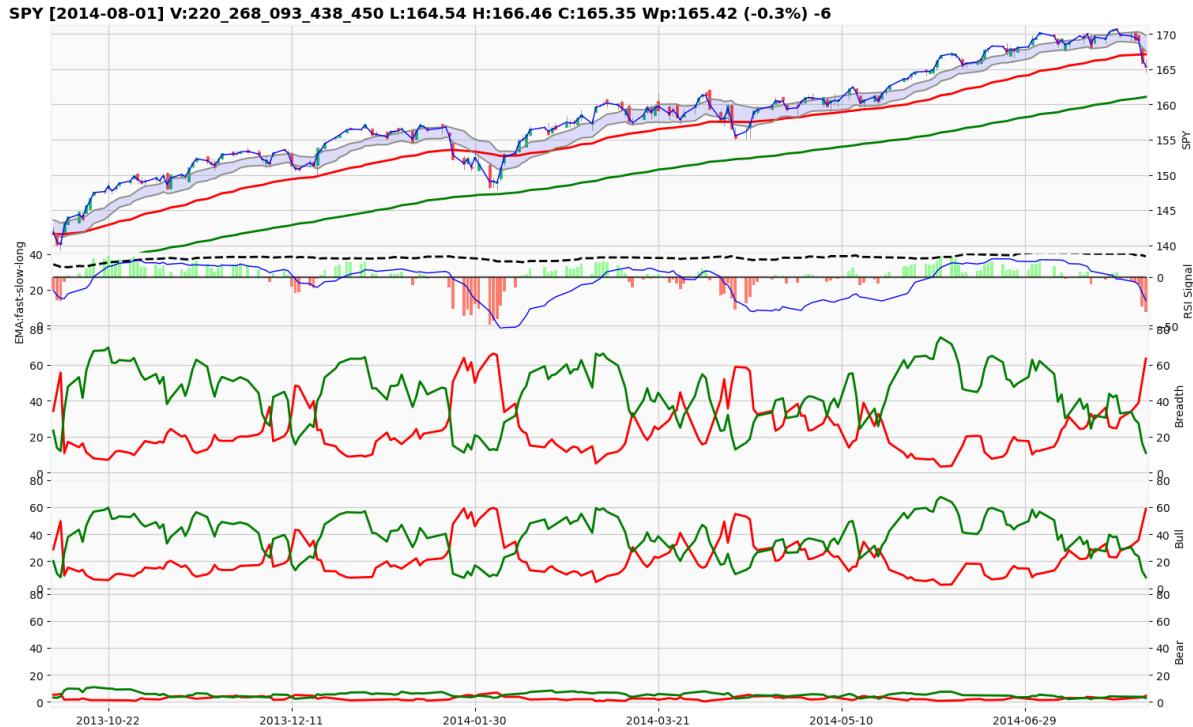
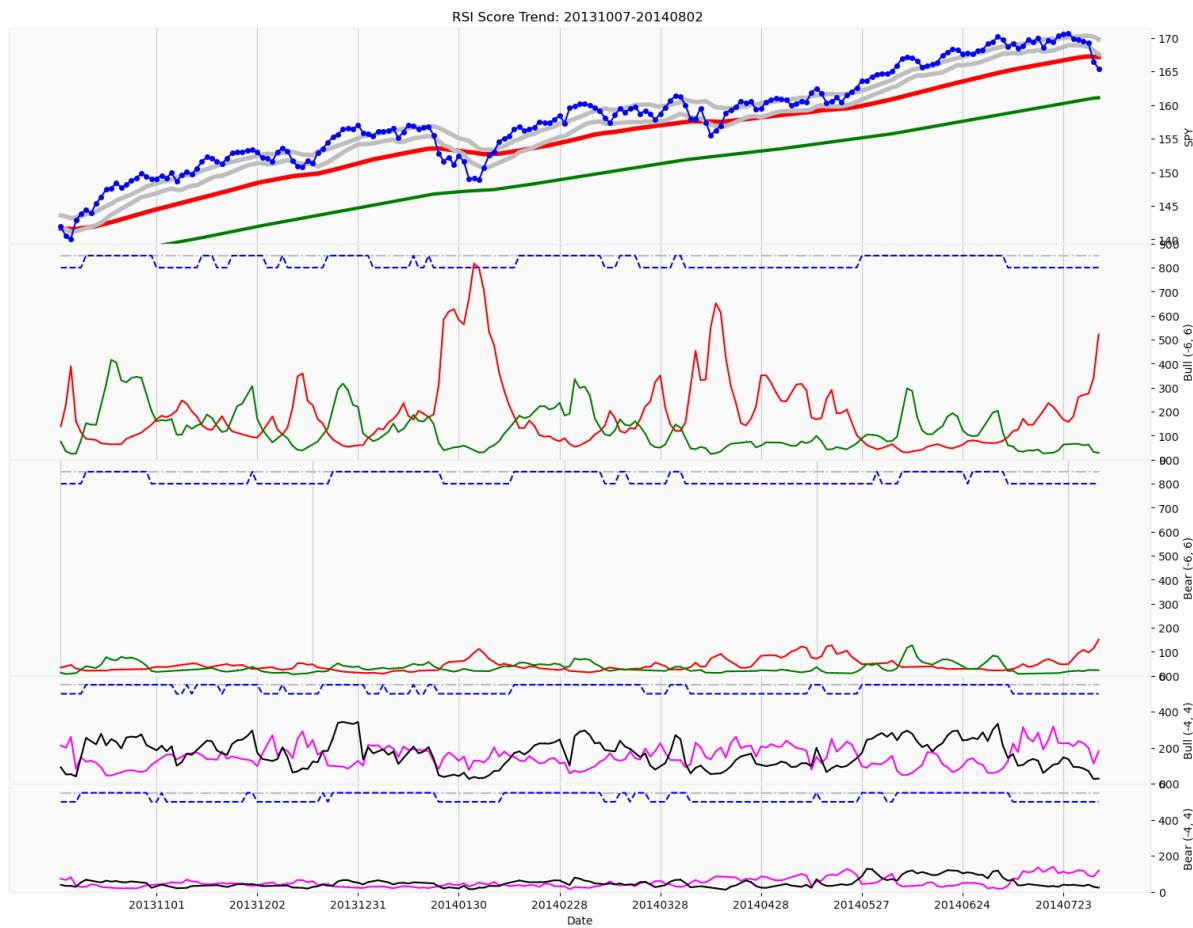


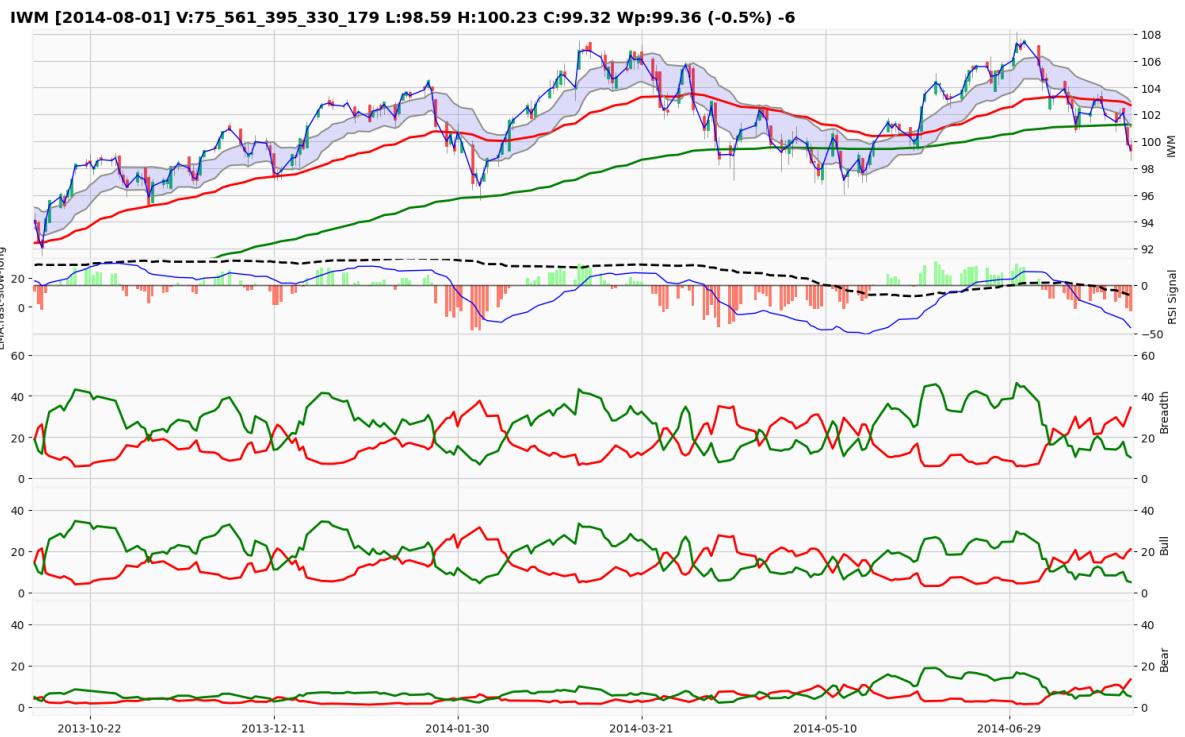
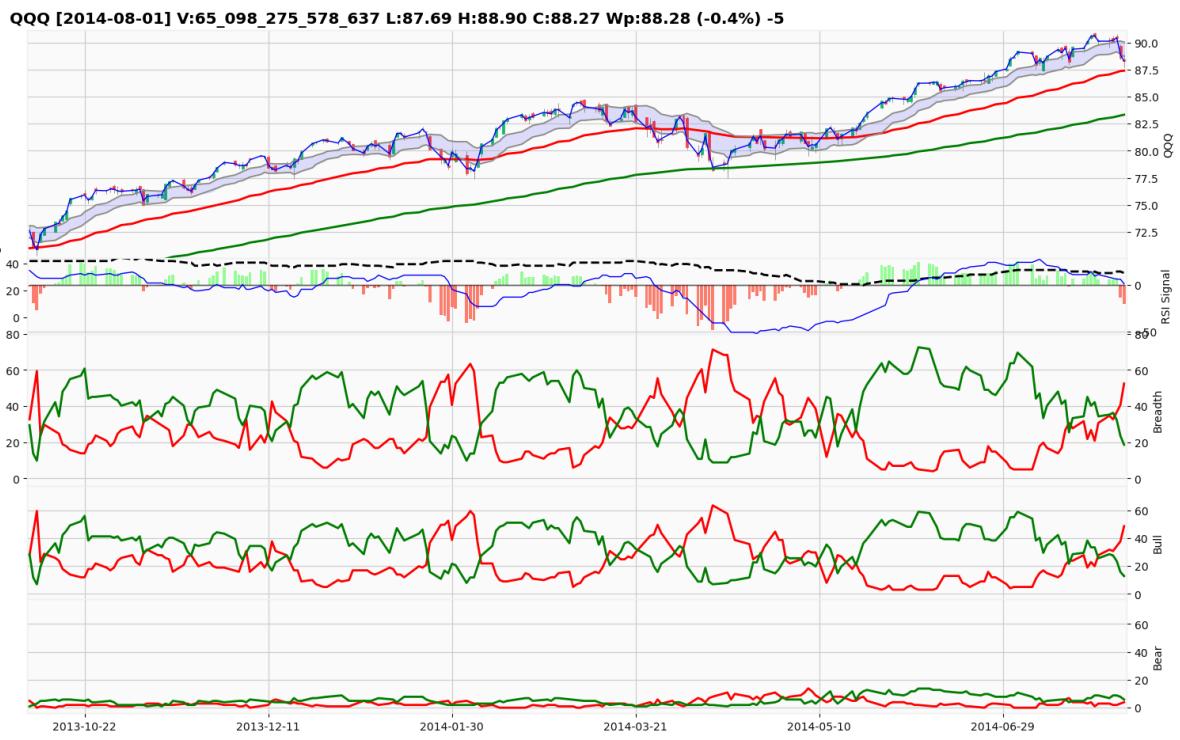
QQQ [2014-01-31] V:48\_912\_996\_412\_810 L:78.95 H:80.09 C:79.71 Wp:79.61 (-0.3%) -6

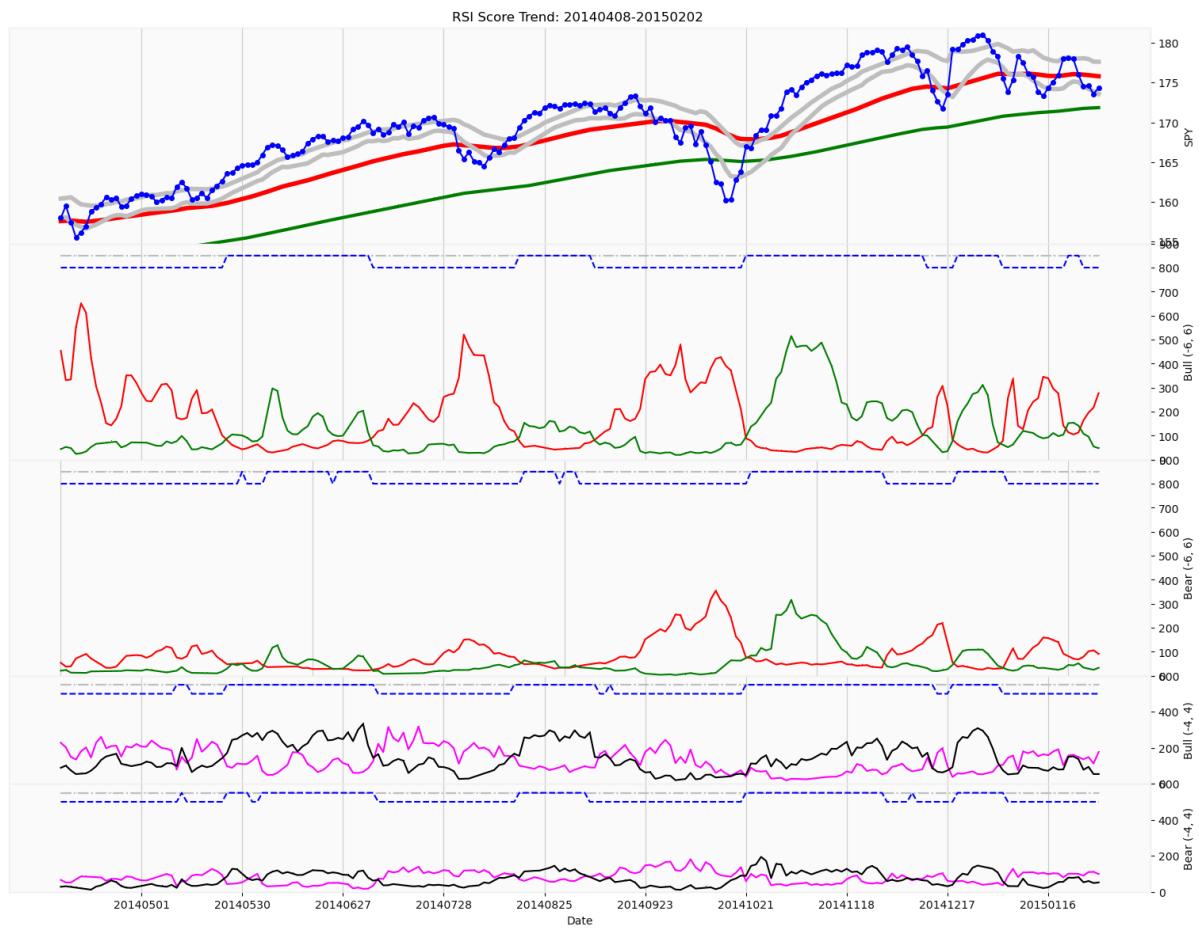


IWM [2014-01-31] V:55\_285\_854\_653\_083 L:99.13 H:100.73 C:100.01 Wp:99.97 (-0.7%) -6

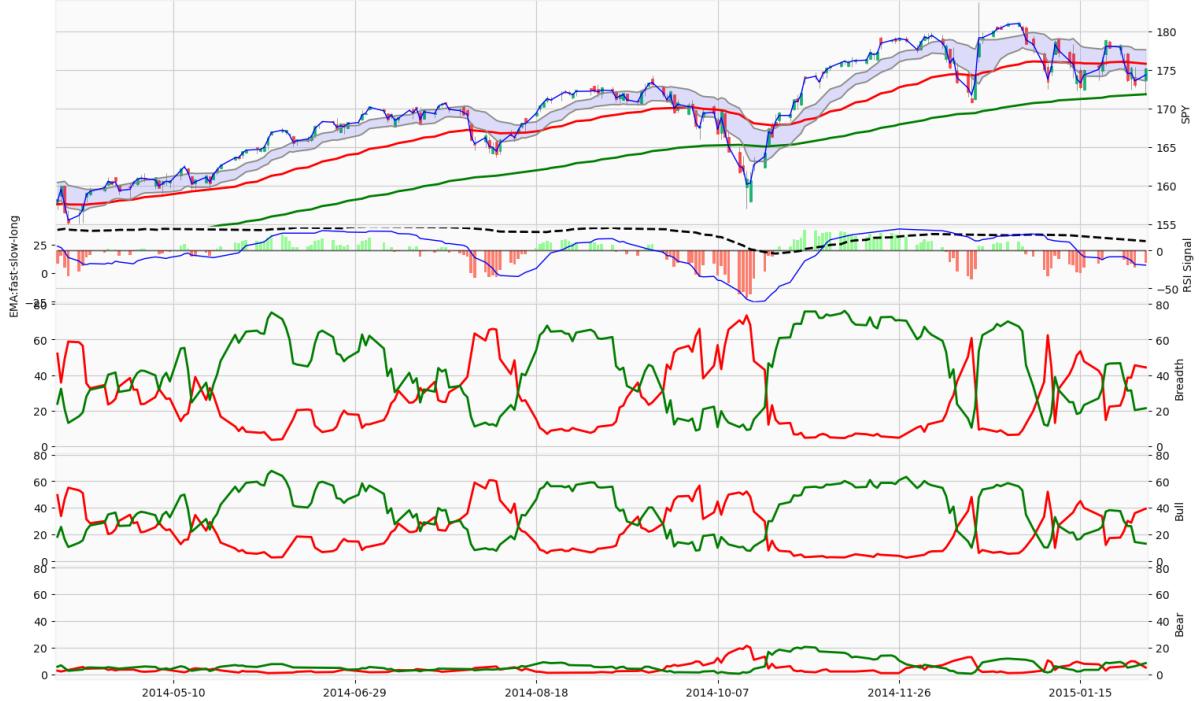


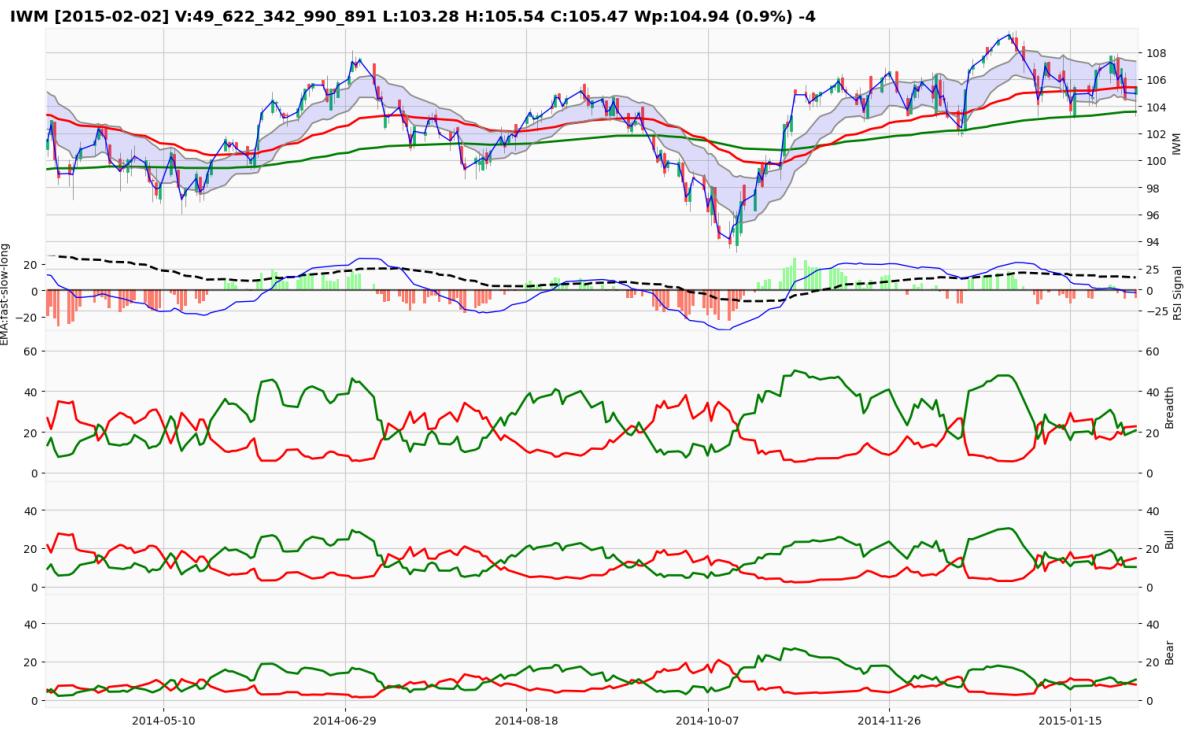
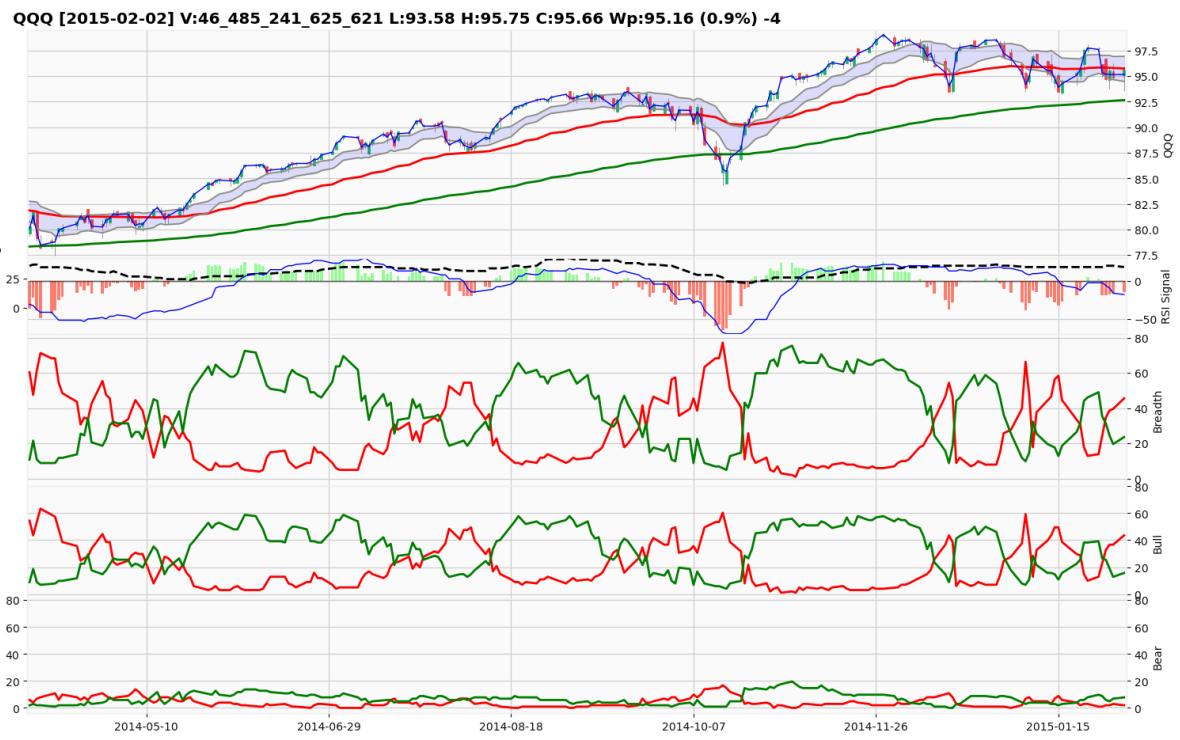


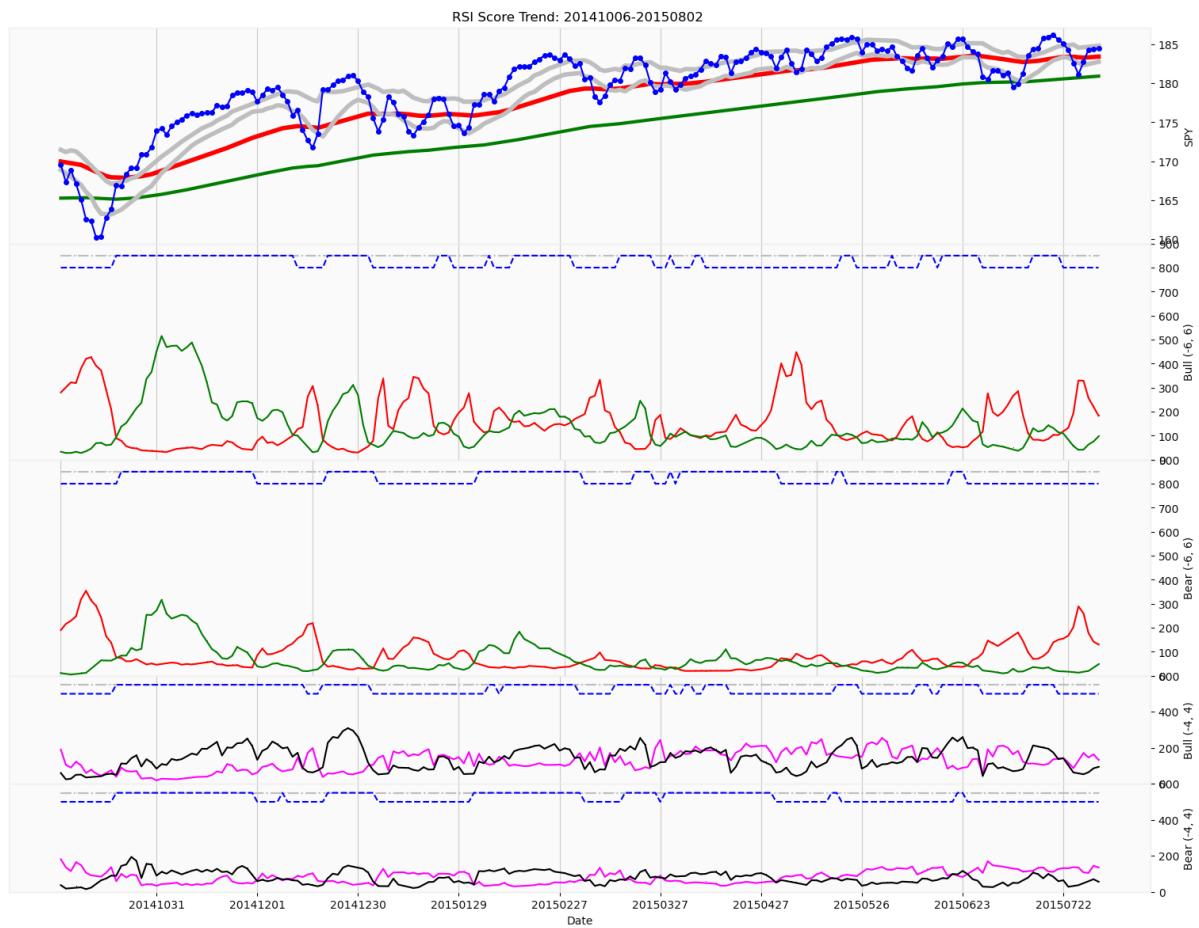




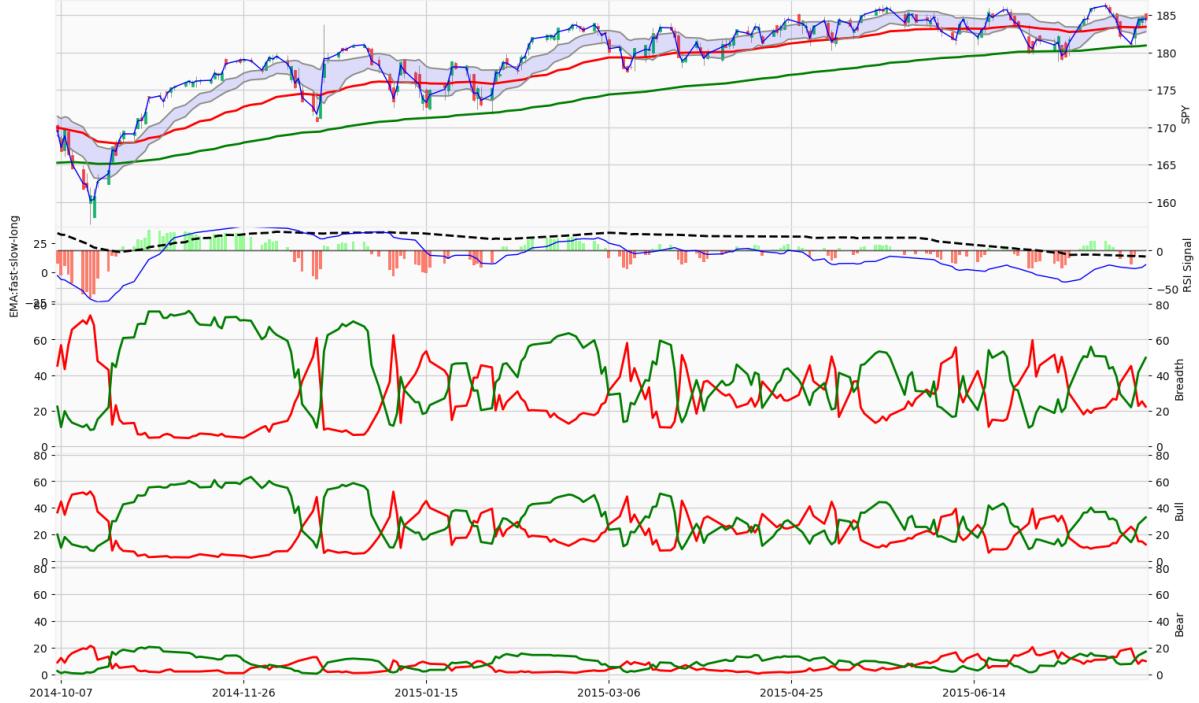
**SPY [2015-02-02] V:187\_667\_292\_680\_920 L:171.73 H:175.30 C:175.20 Wp:174.35 (1.2%) -2**

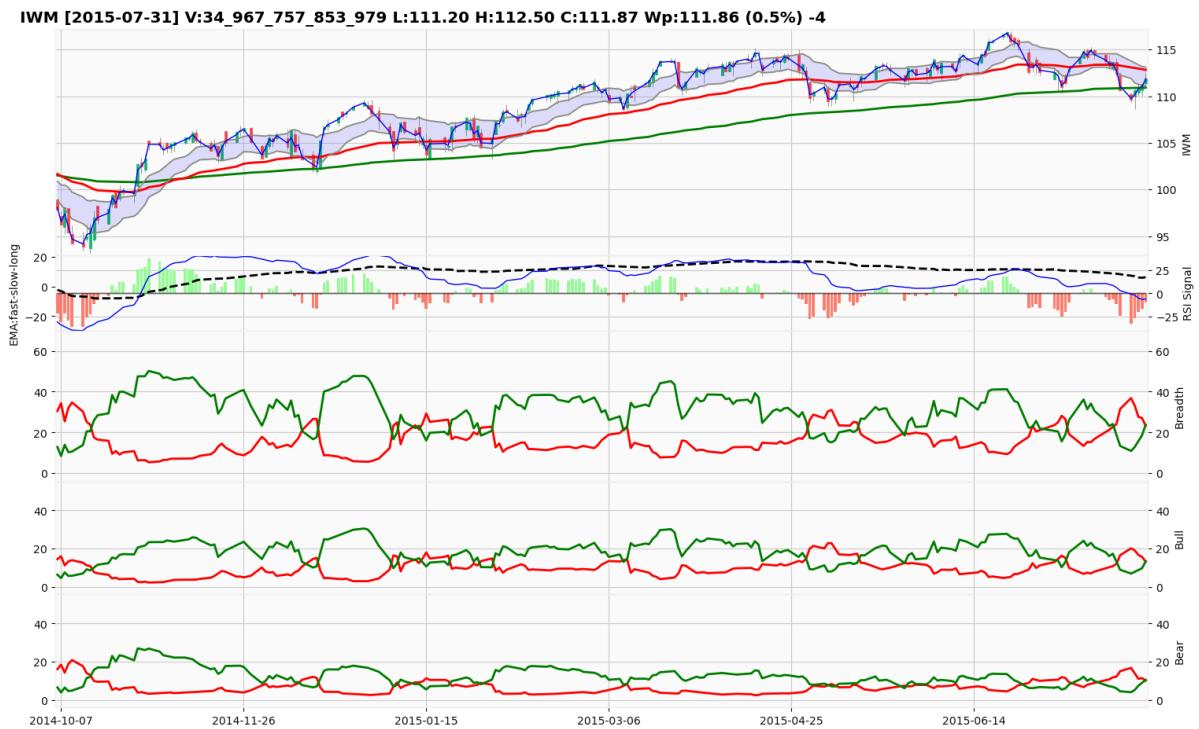


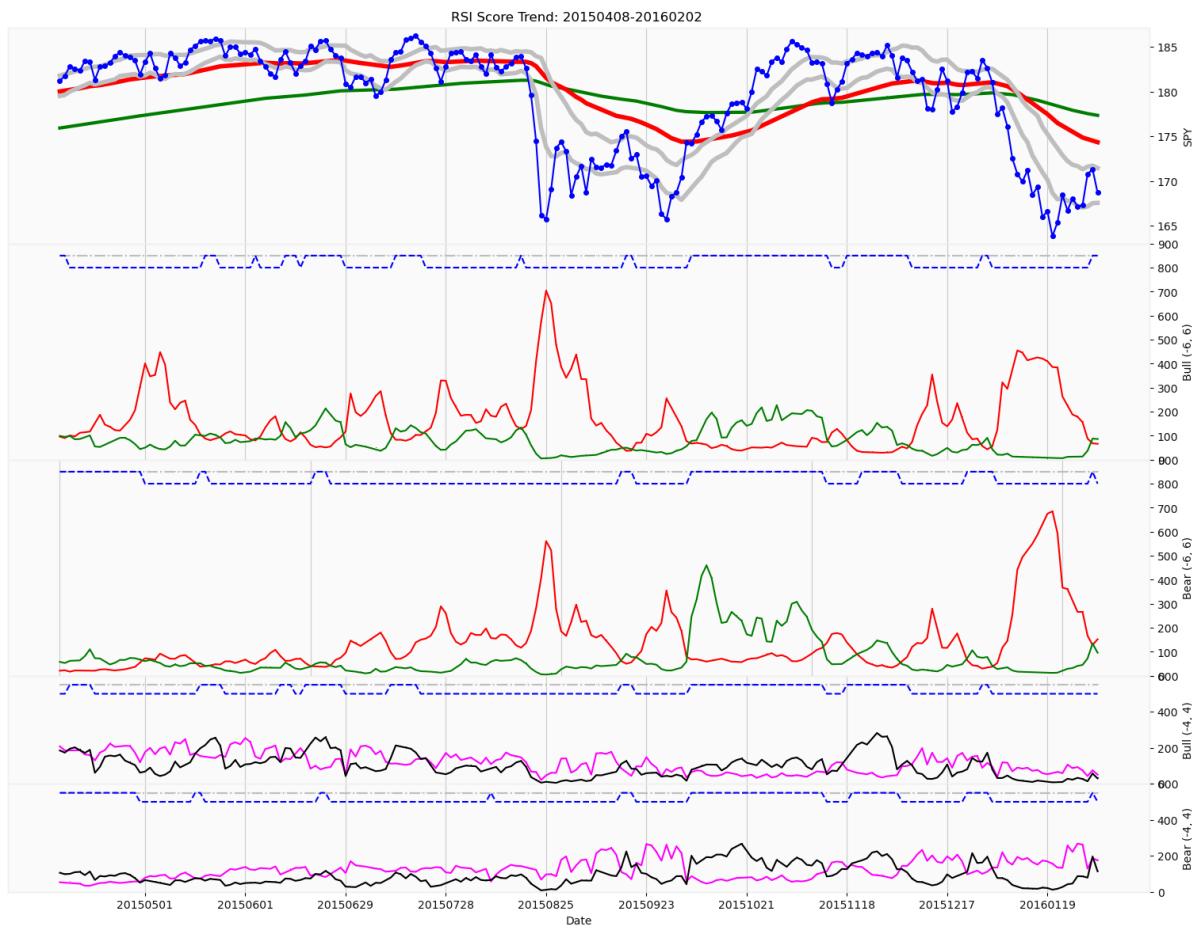


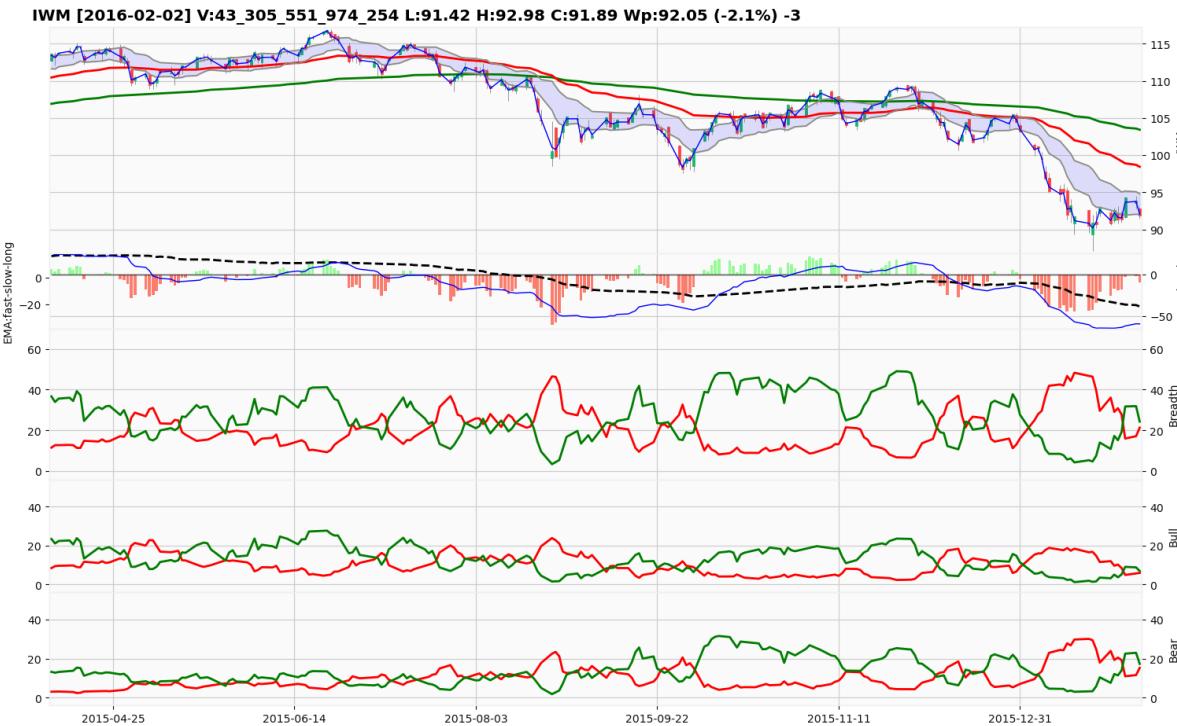
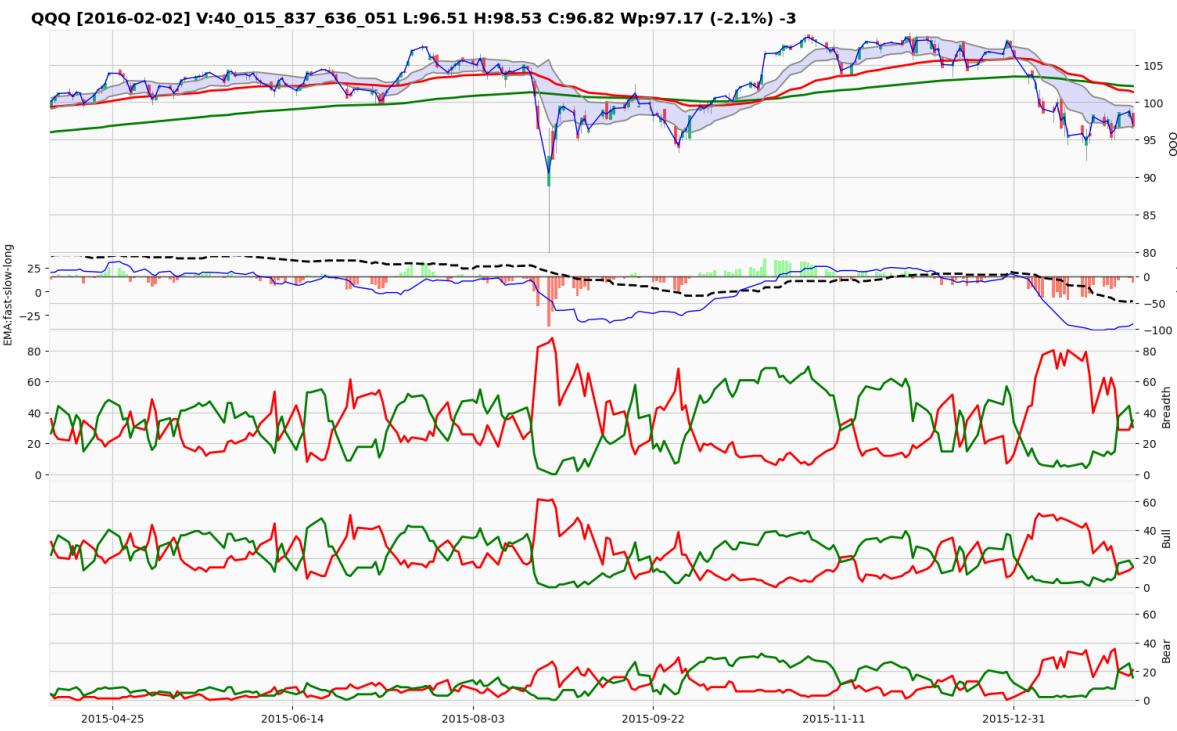


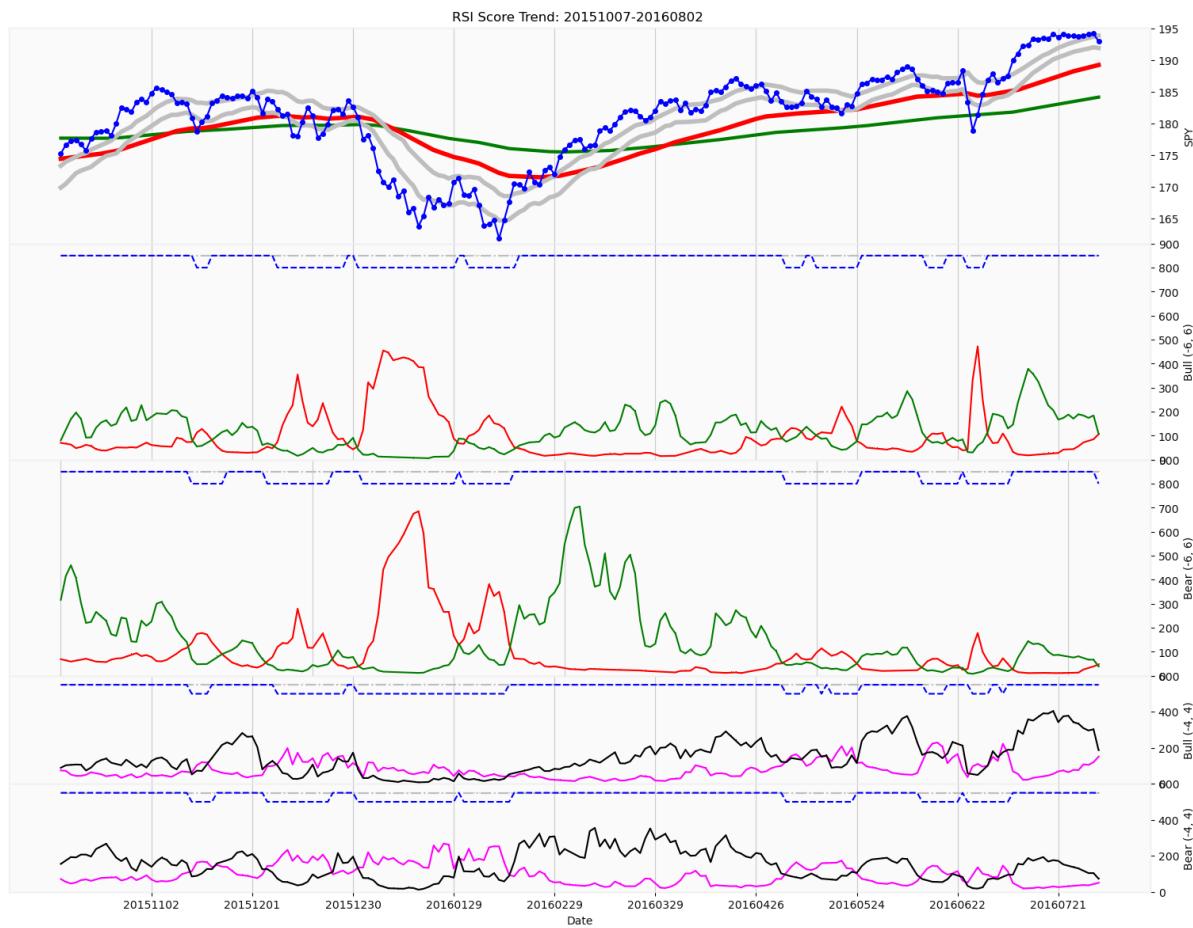
SPY [2015-07-31] V:117\_900\_857\_907\_220 L:184.05 H:185.22 C:184.37 Wp:184.50 (-0.2%) 0



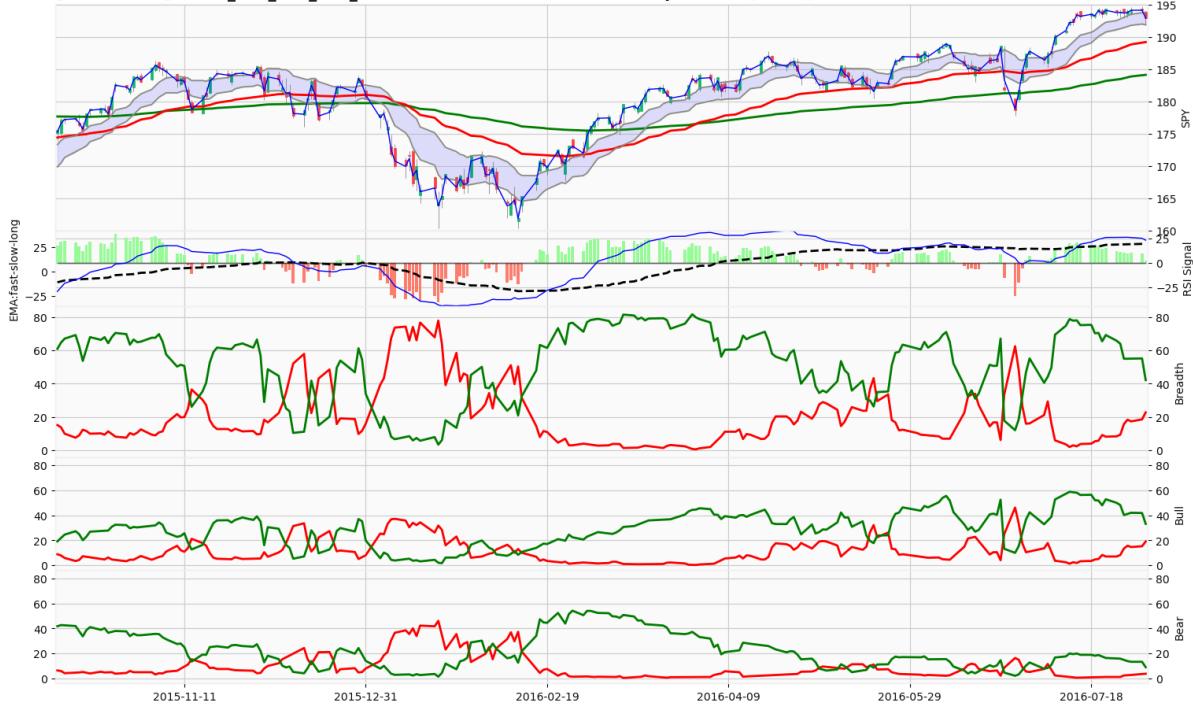








**SPY [2016-08-02] V:103\_035\_520\_389\_410 L:192.00 H:194.03 C:192.89 Wp:192.95 (-0.6%) 1**

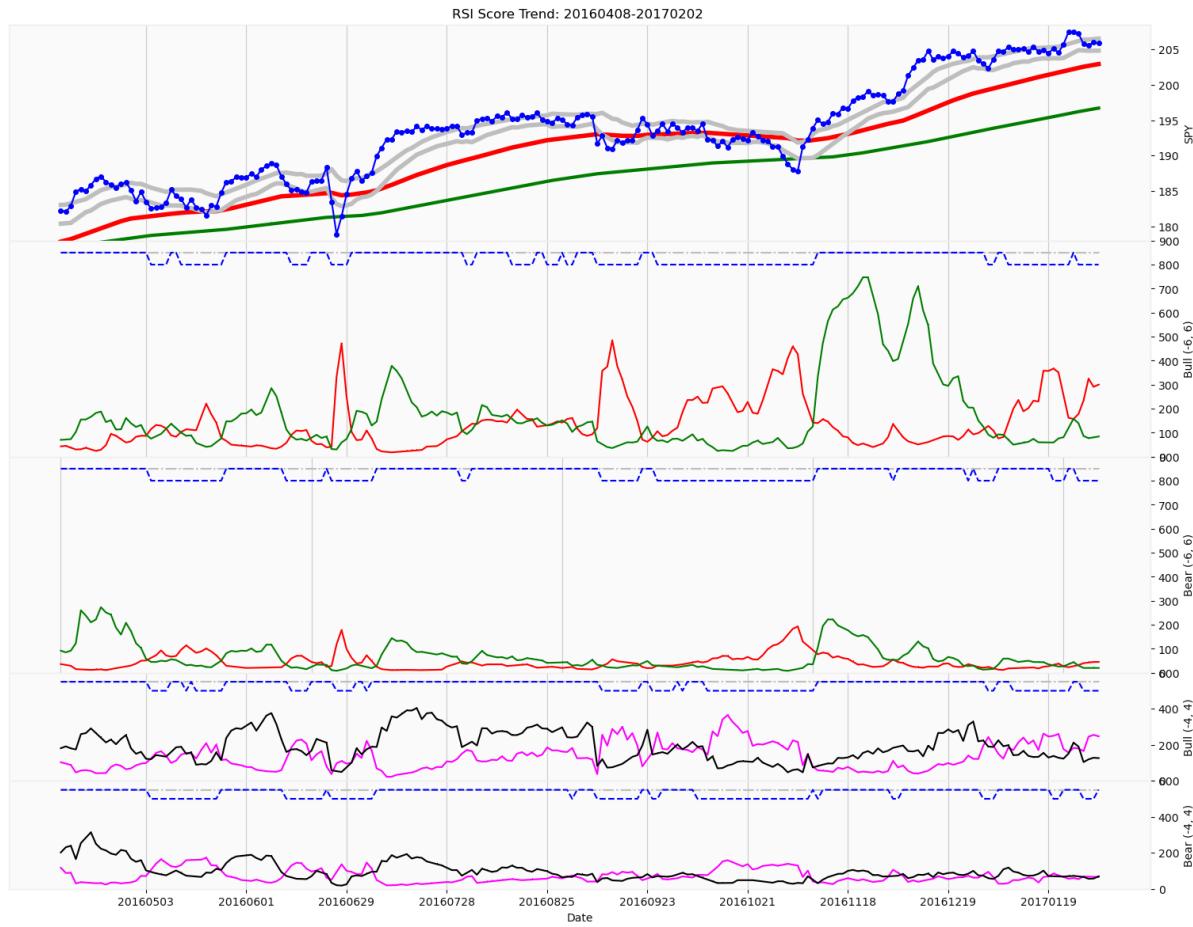


QQQ [2016-08-02] V:20\_214\_248\_294\_752 L:108.83 H:110.36 C:109.56 Wp:109.58 (-0.8%) 2

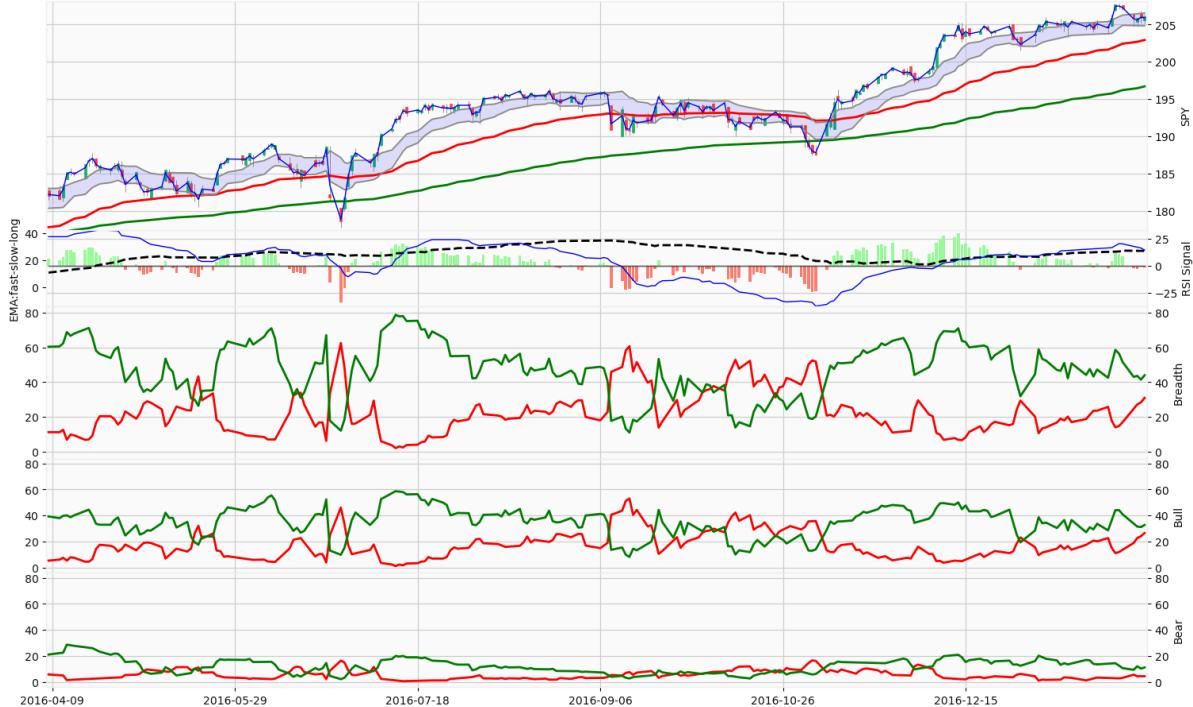


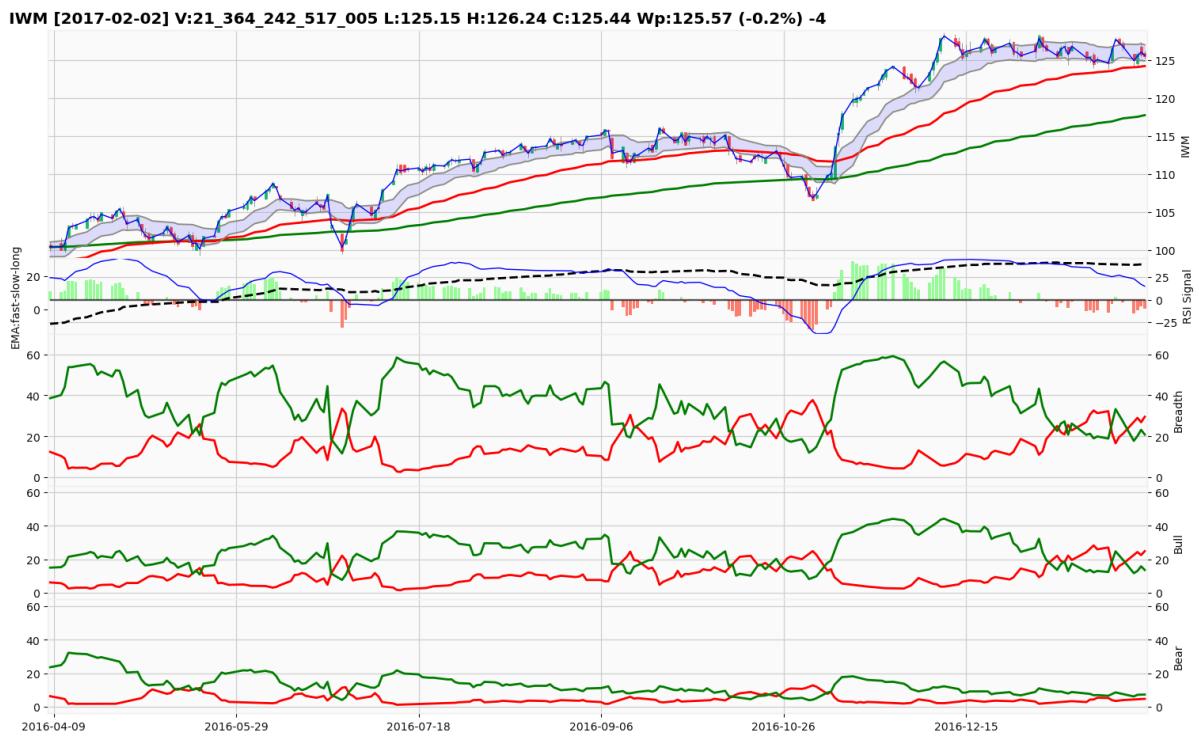
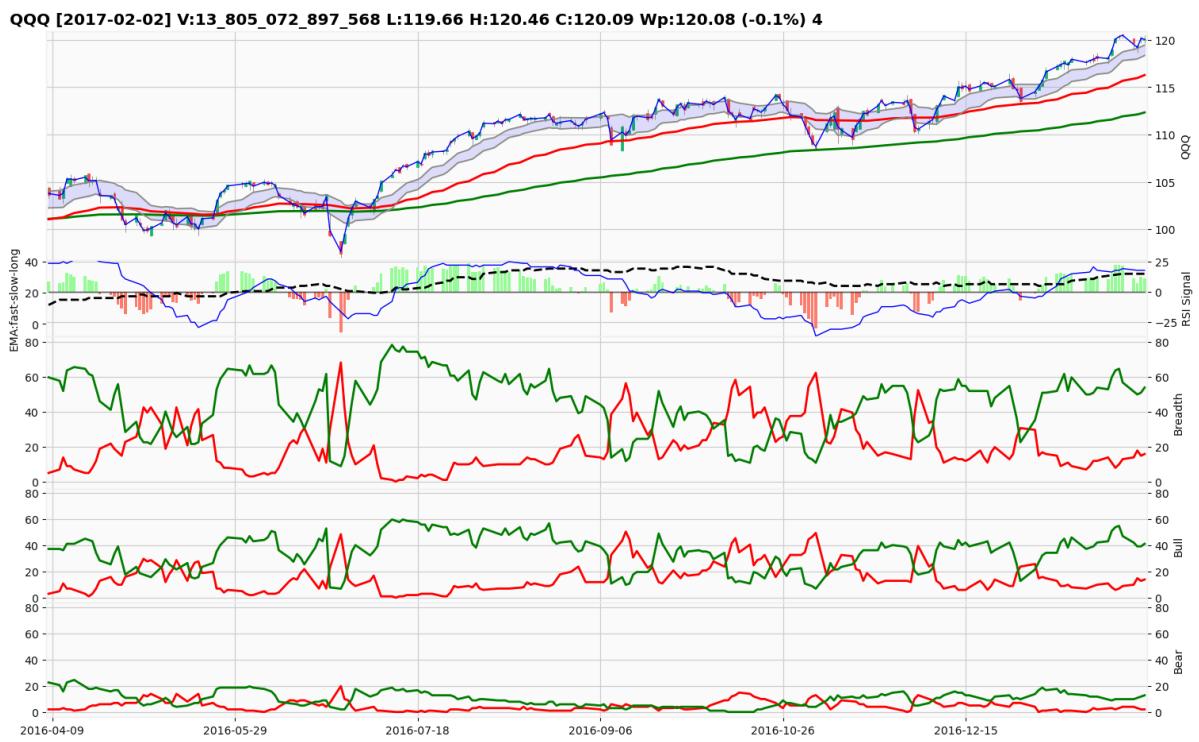
IWM [2016-08-02] V:31\_040\_772\_196\_608 L:110.05 H:112.00 C:110.41 Wp:110.72 (-1.4%) 1

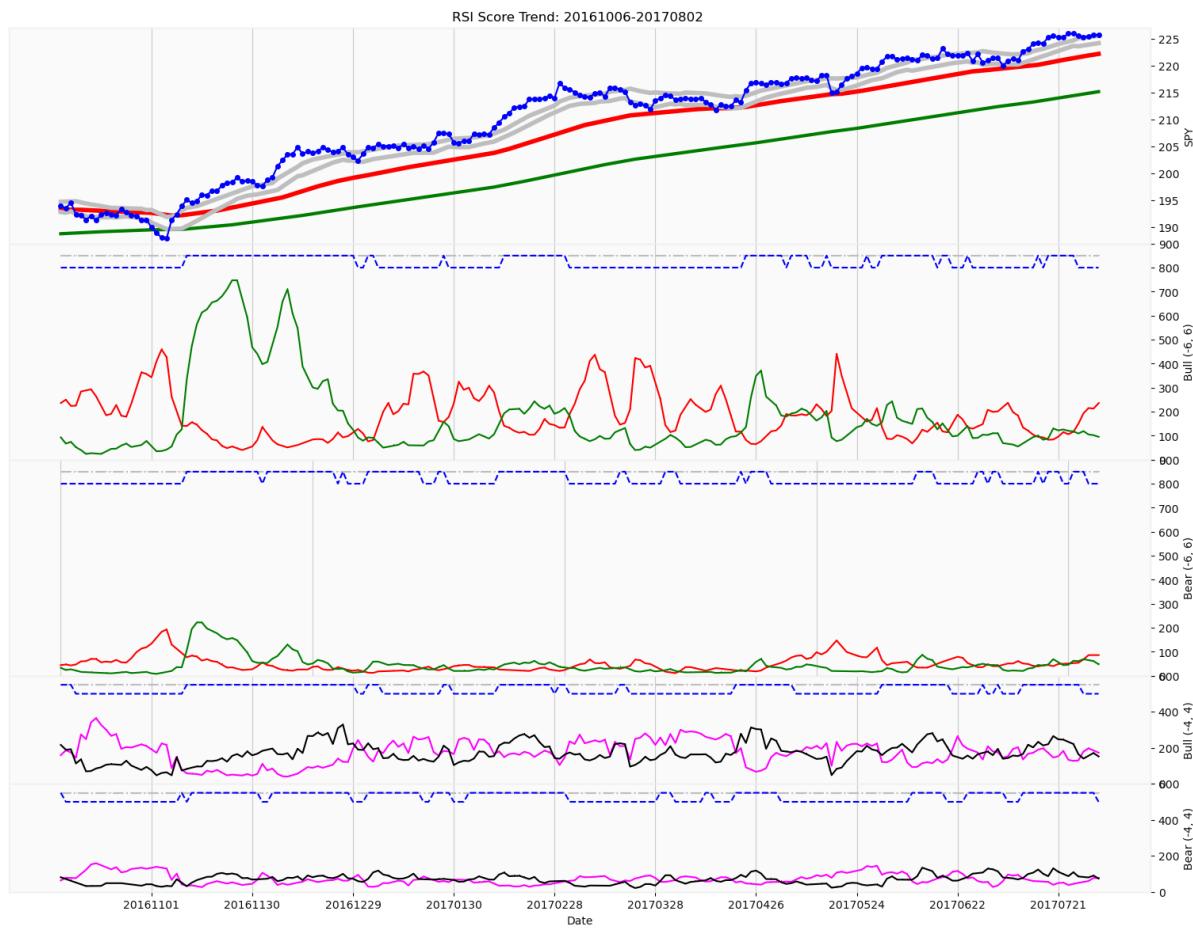




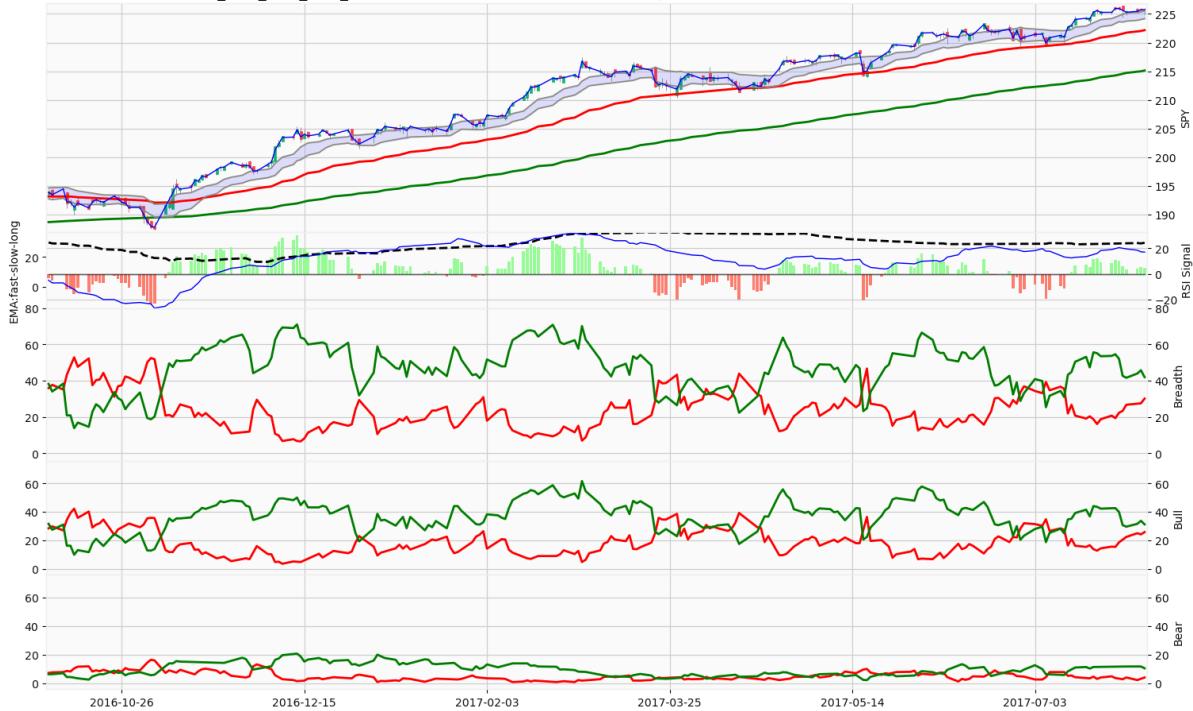
**SPY [2017-02-02] V:76\_831\_974\_247\_632 L:205.18 H:206.35 C:206.05 Wp:205.91 (0.1%) 0**

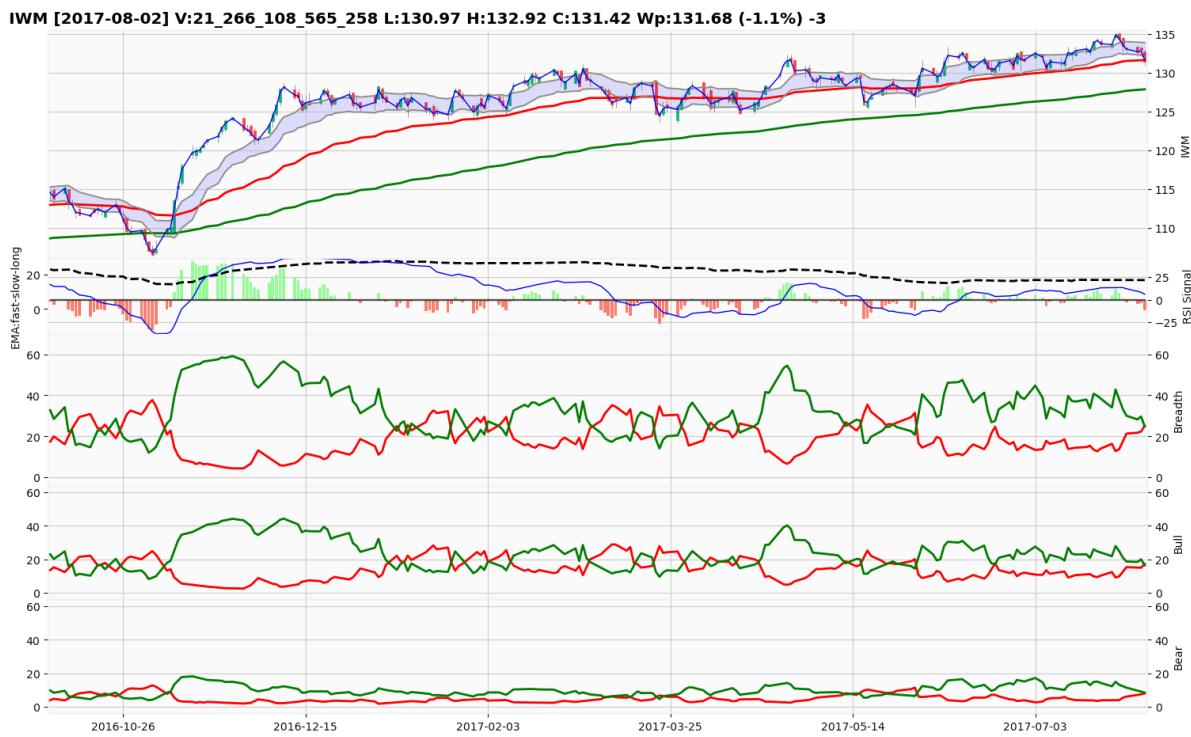
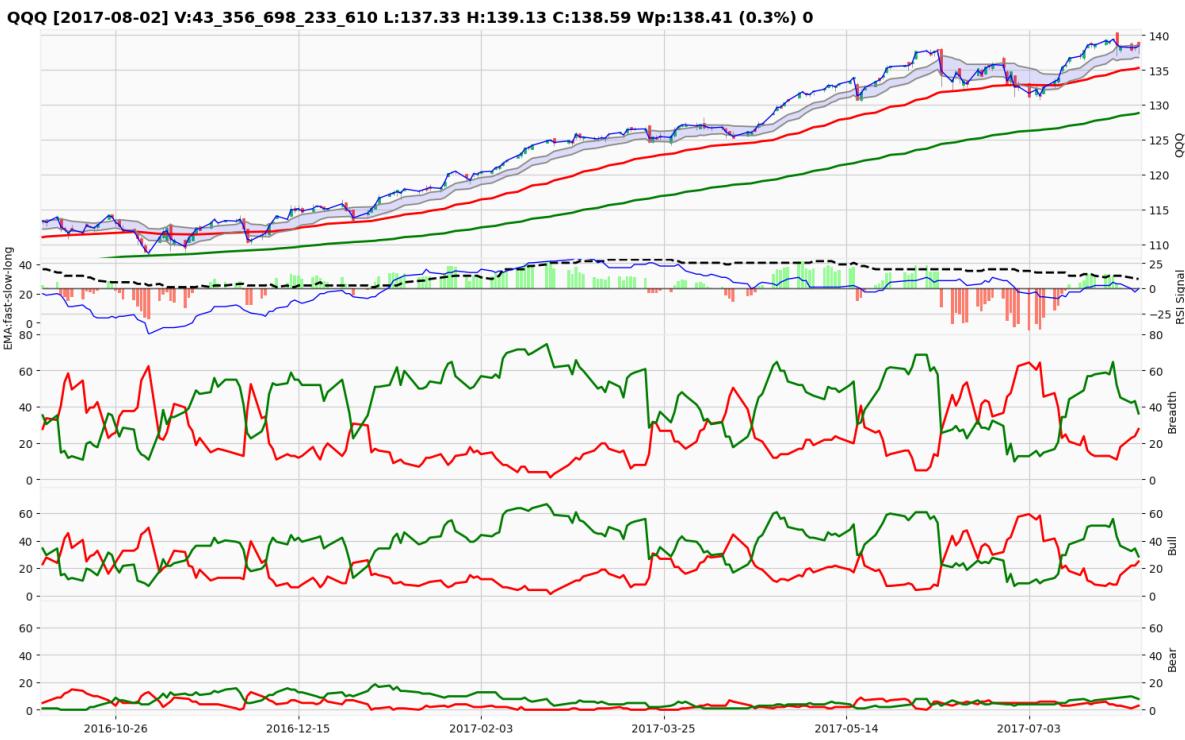


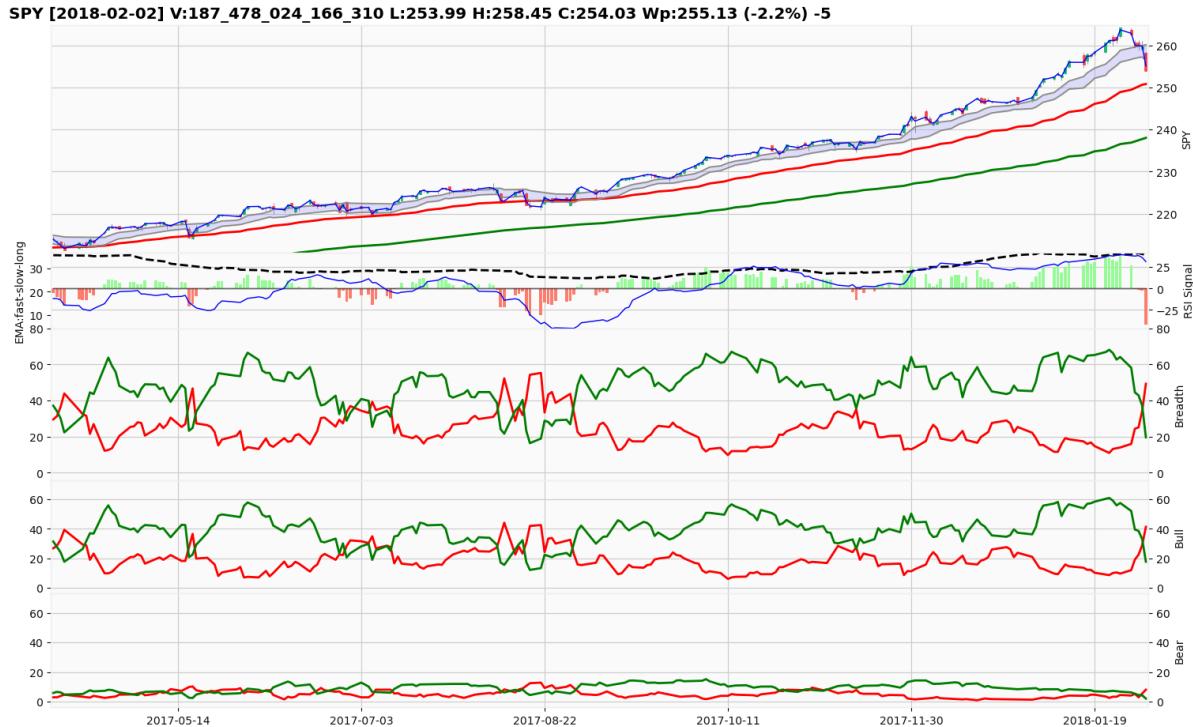
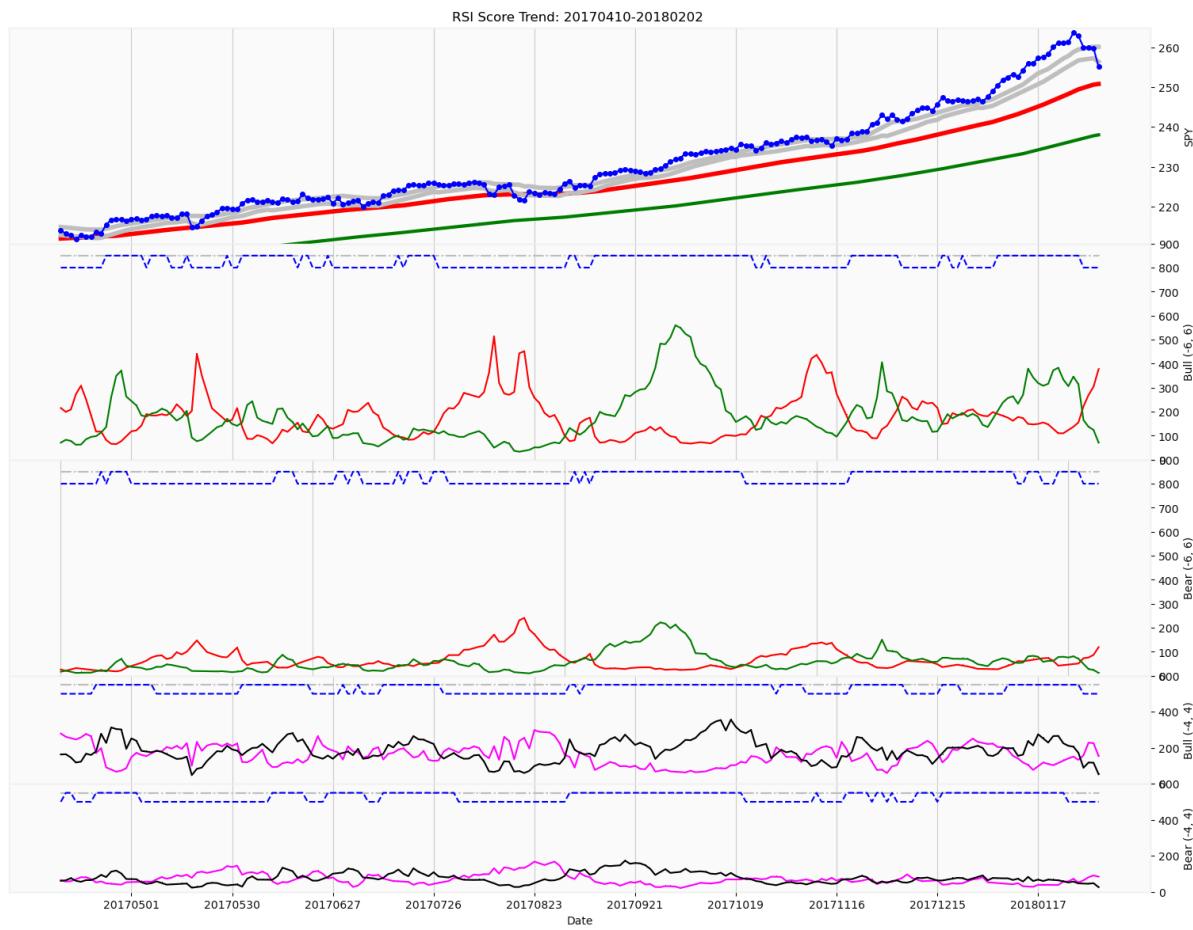


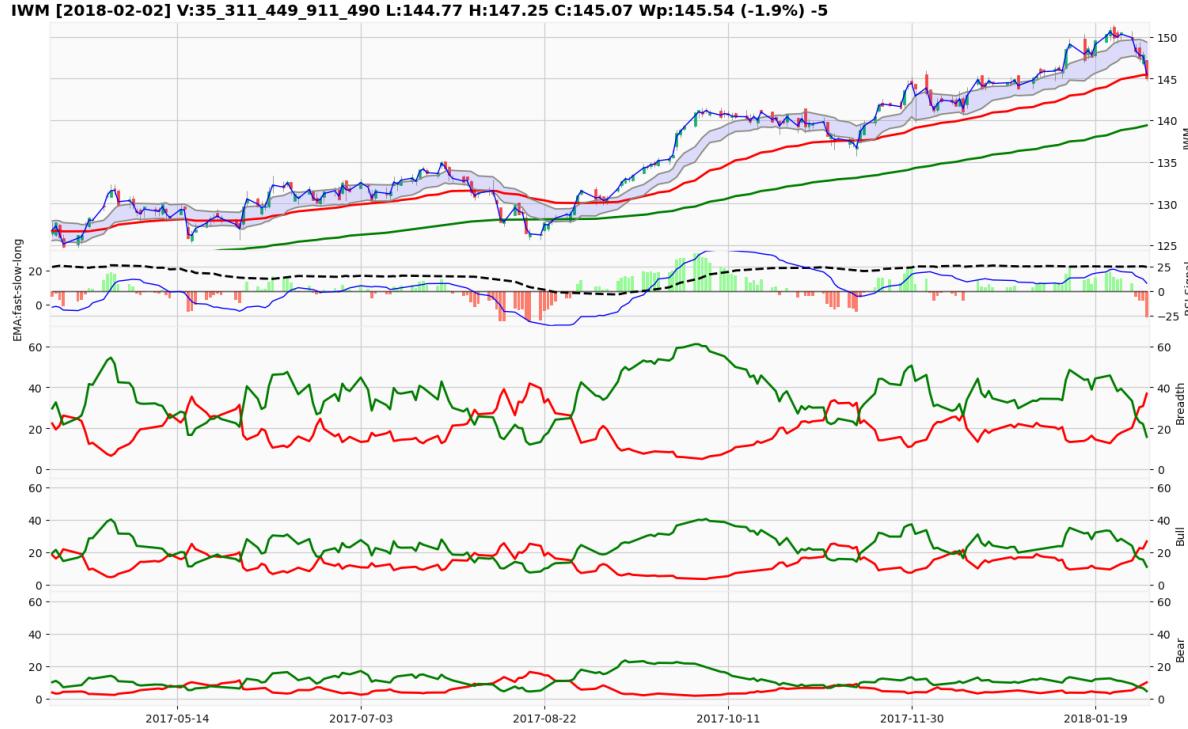
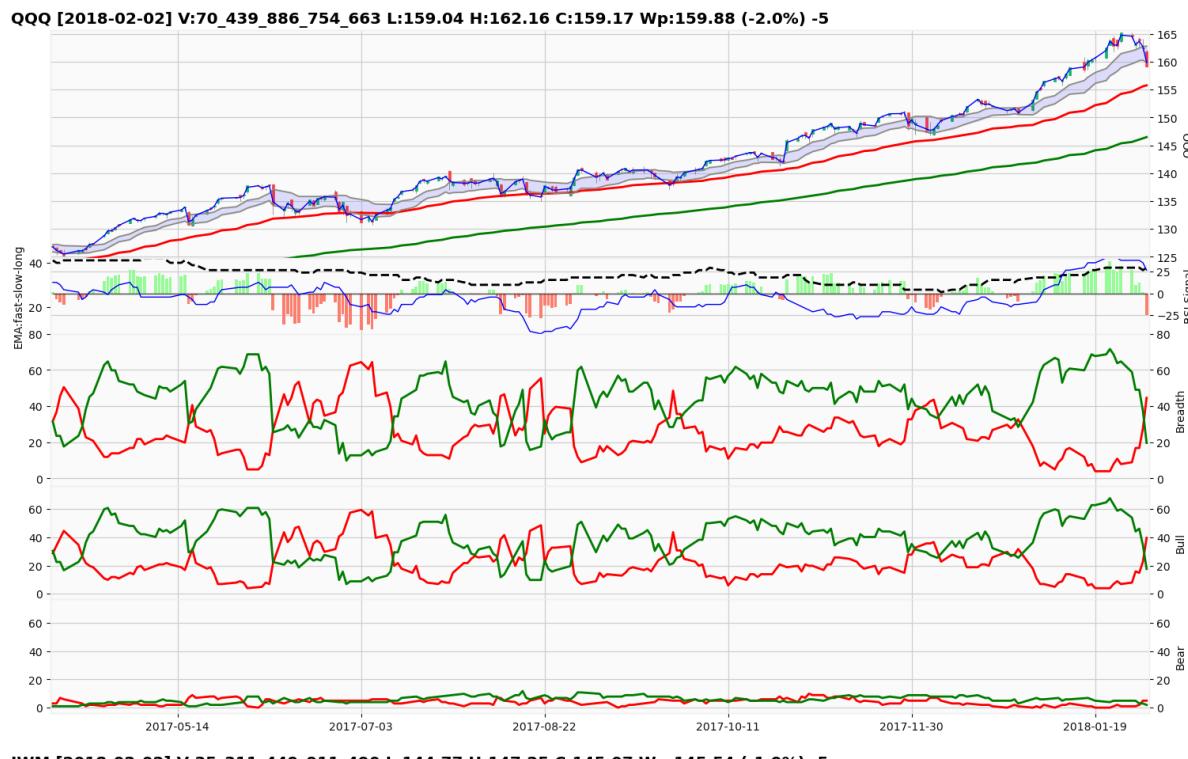


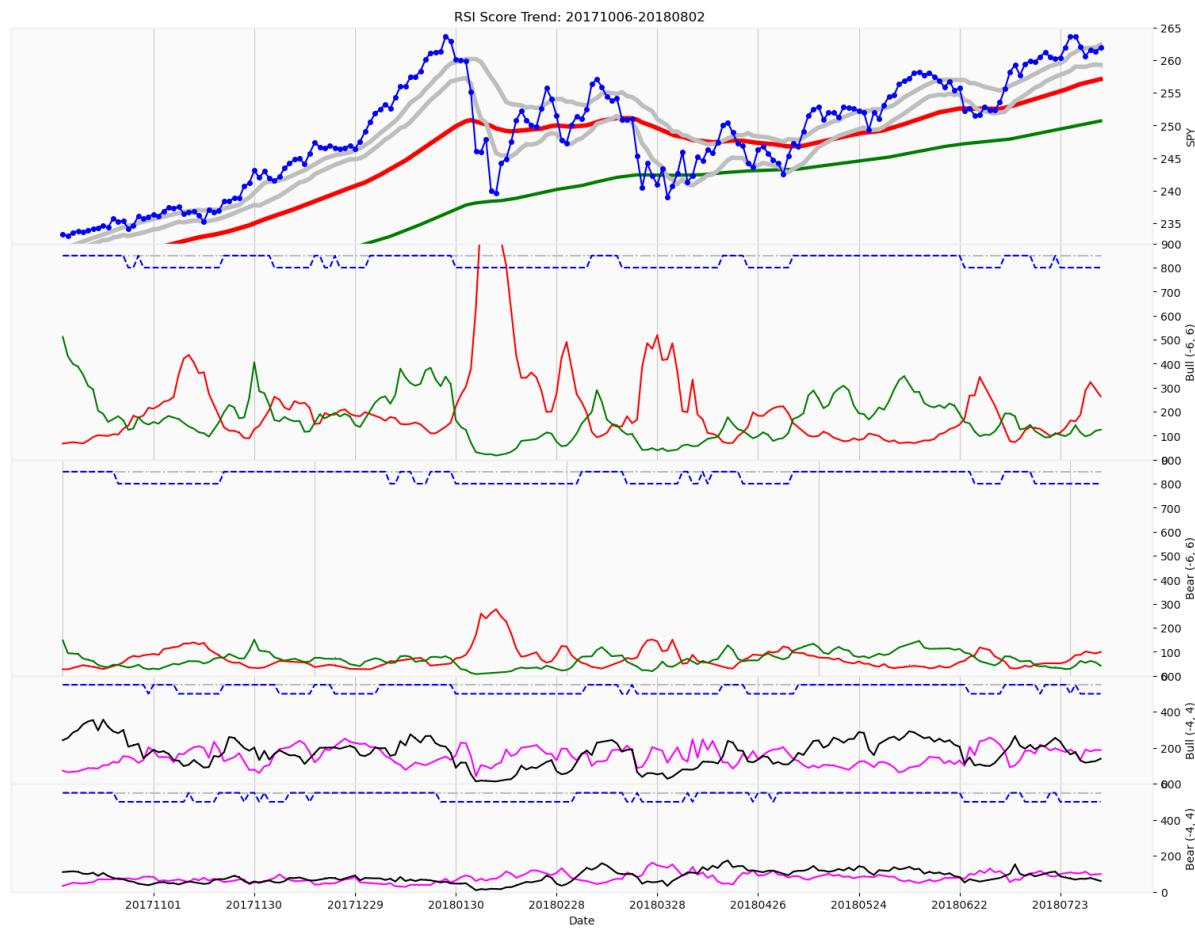
SPY [2017-08-02] V:51\_908\_700\_322\_055 L:224.95 H:226.08 C:225.93 Wp:225.72 (0.1%) 0





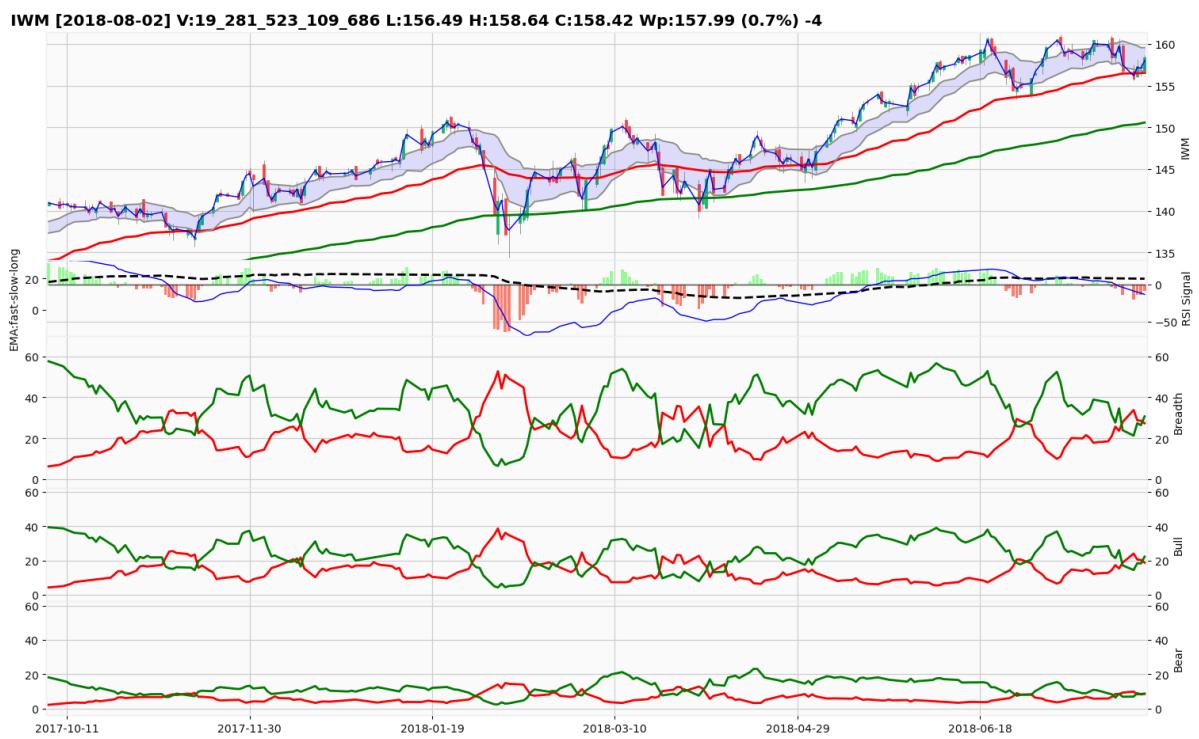
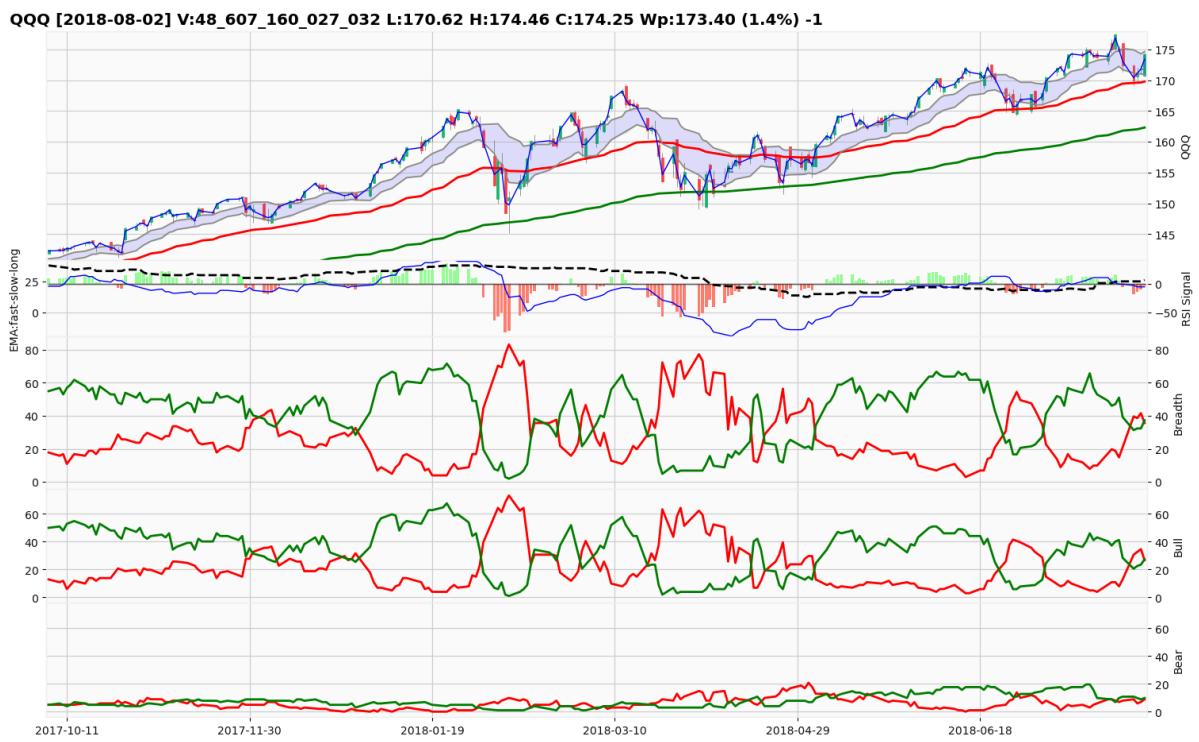


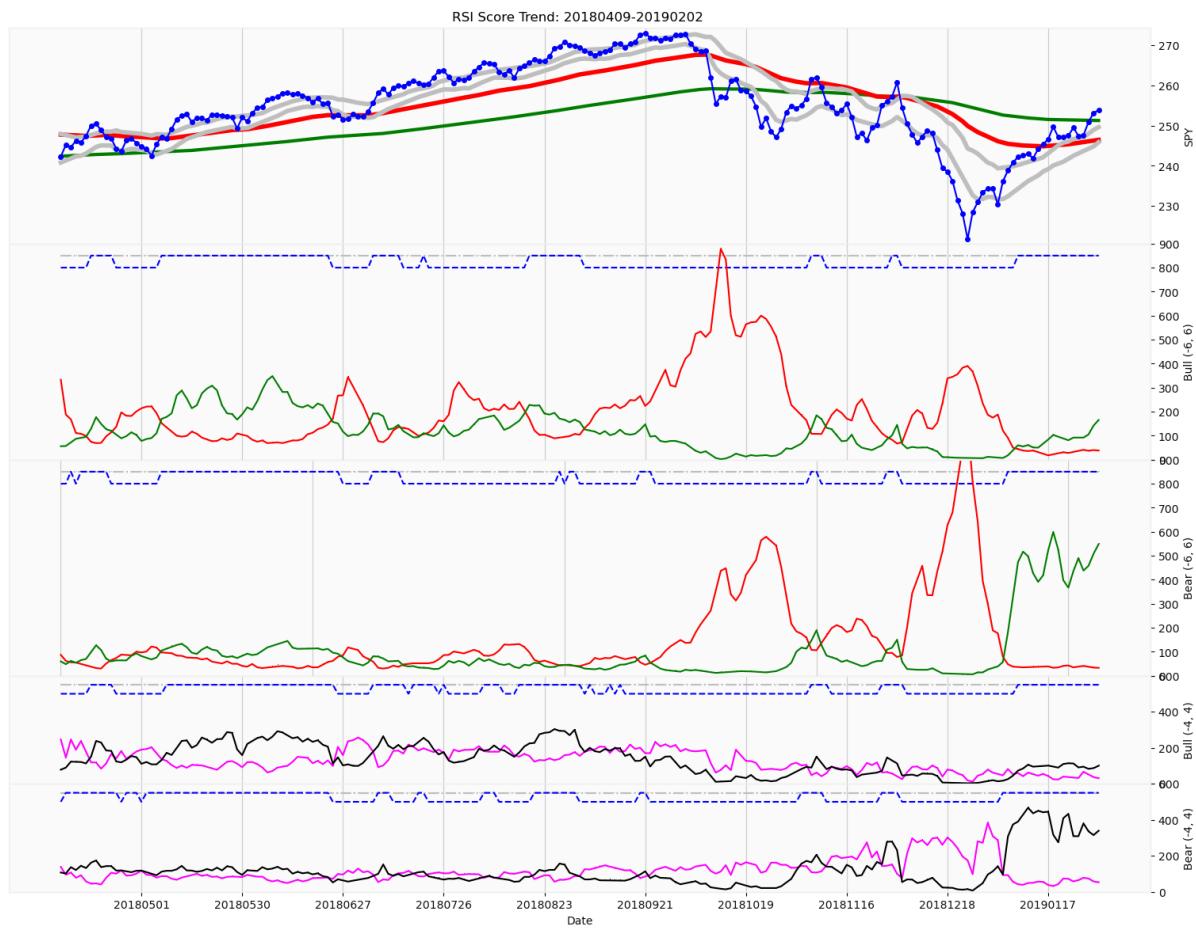




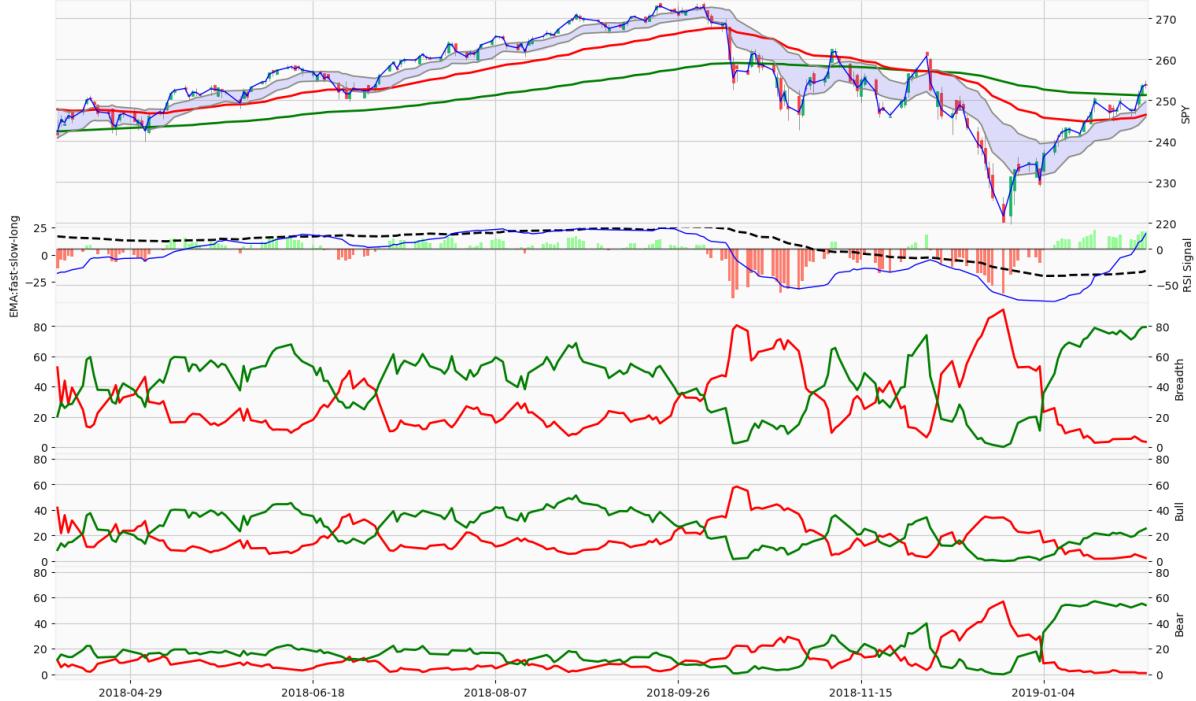
**SPY [2018-08-02] V:68\_190\_631\_187\_765 L:259.65 H:262.83 C:262.65 Wp:261.94 (0.5%) 0**



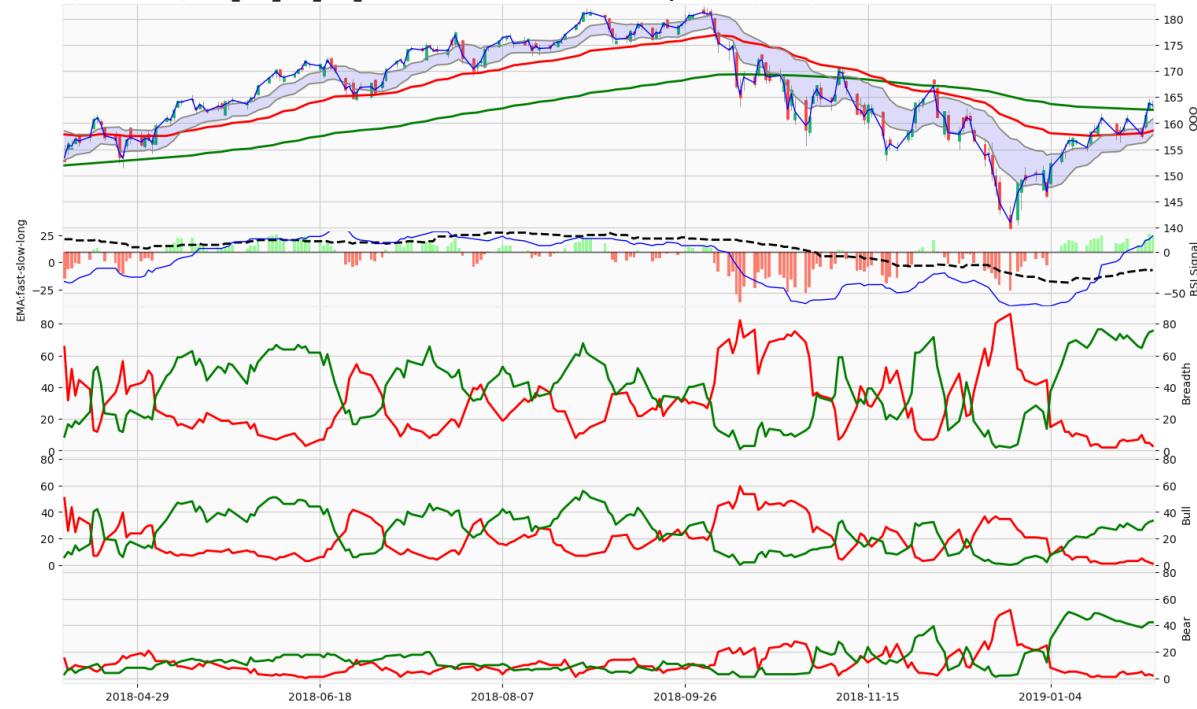




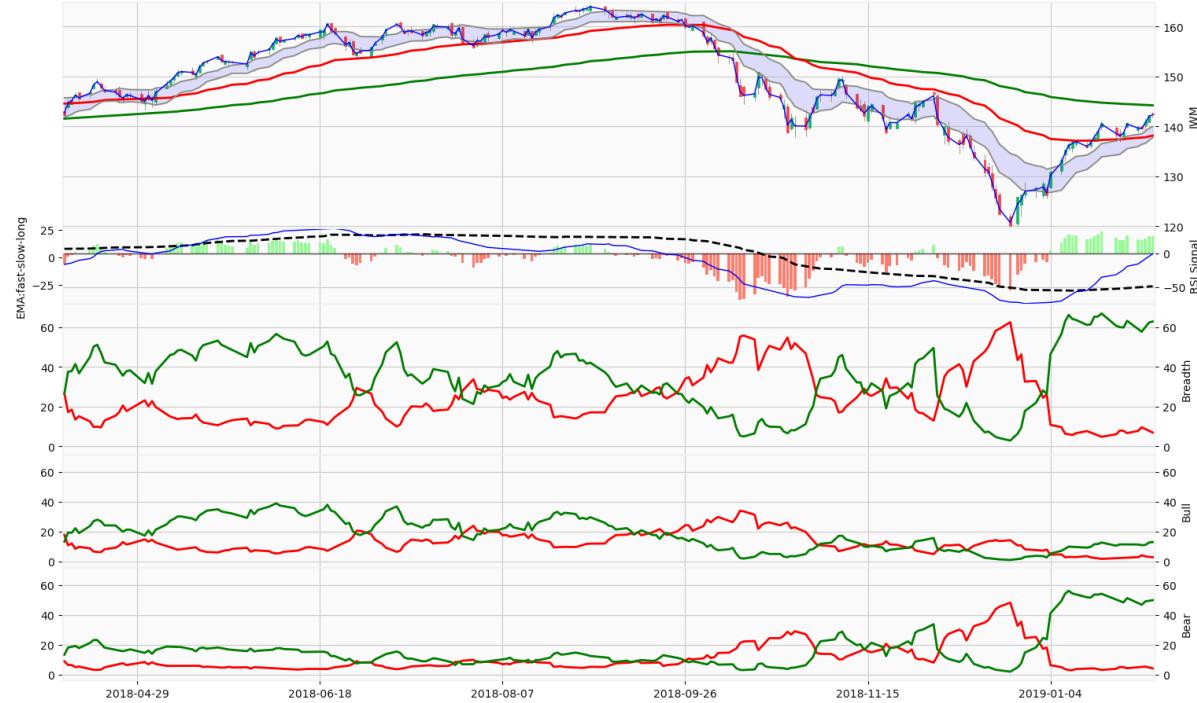
**SPY [2019-02-01] V:91\_276\_519\_088\_167 L:252.98 H:254.88 C:253.79 Wp:253.86 (0.1%) 6**

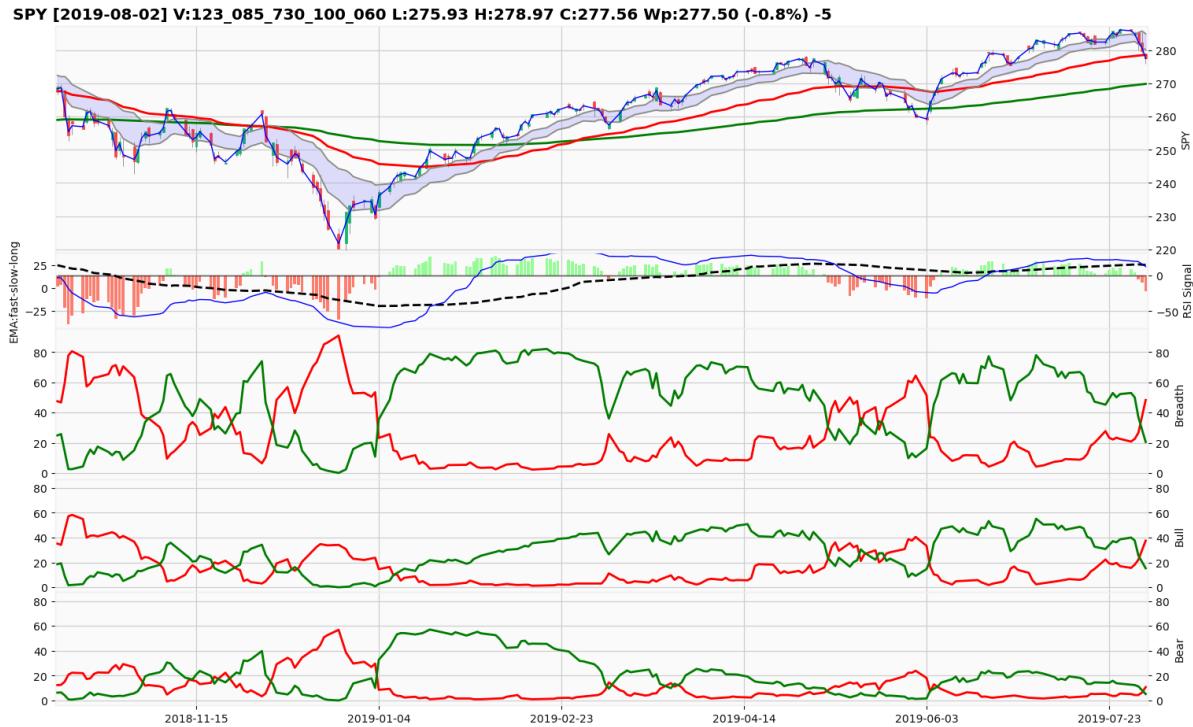
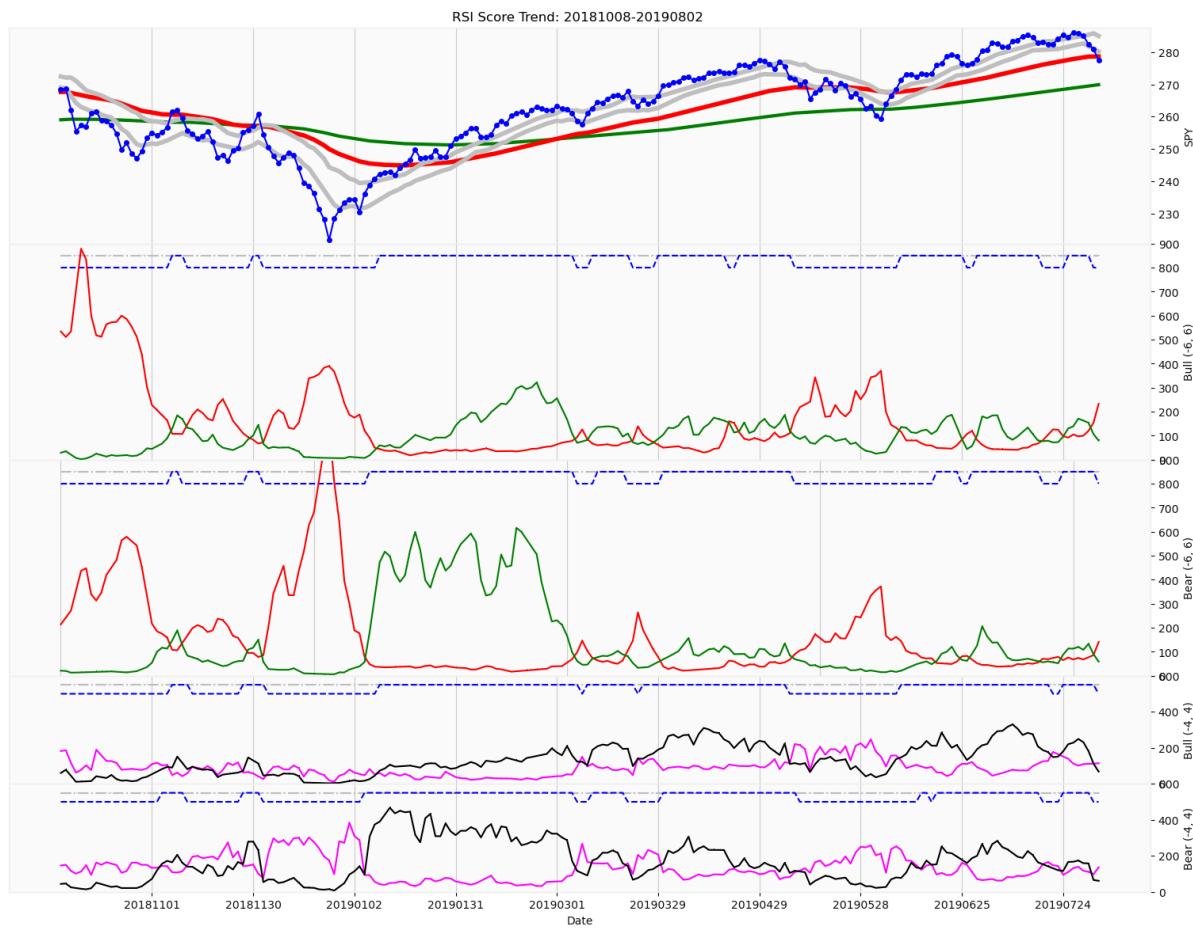


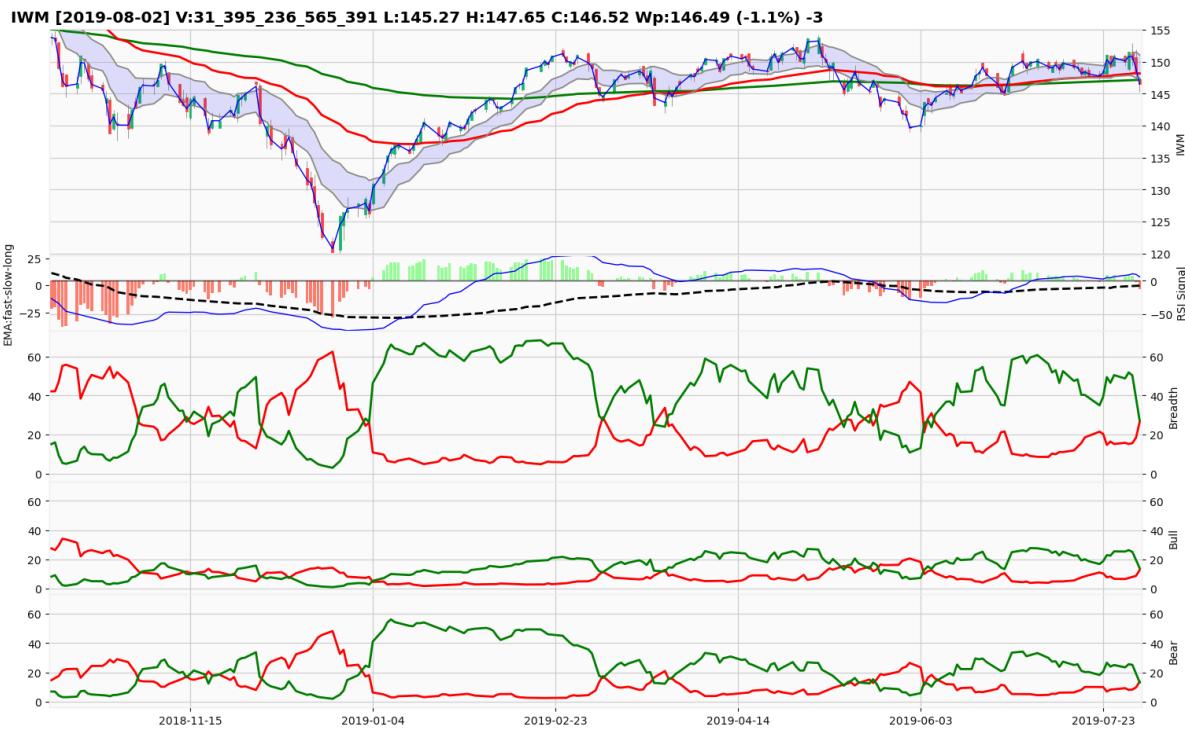
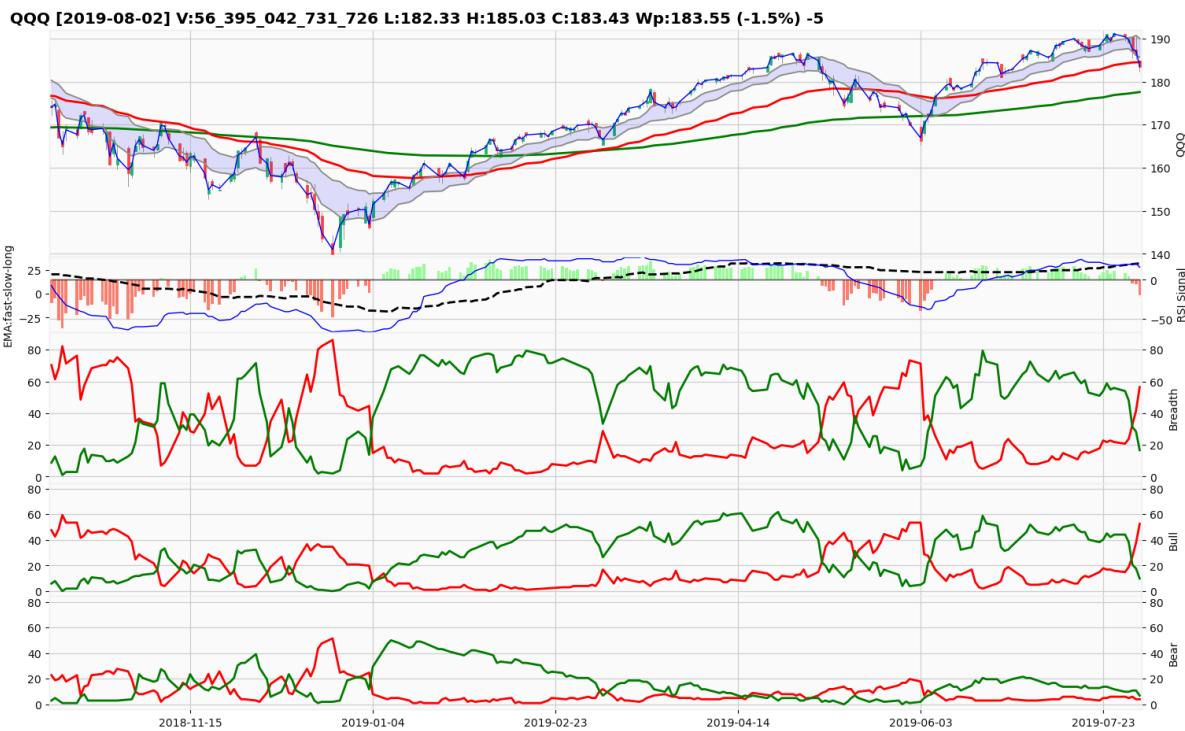
QQQ [2019-02-01] V:32\_963\_663\_171\_695 L:162.84 H:164.43 C:163.29 Wp:163.46 (-0.4%) 6

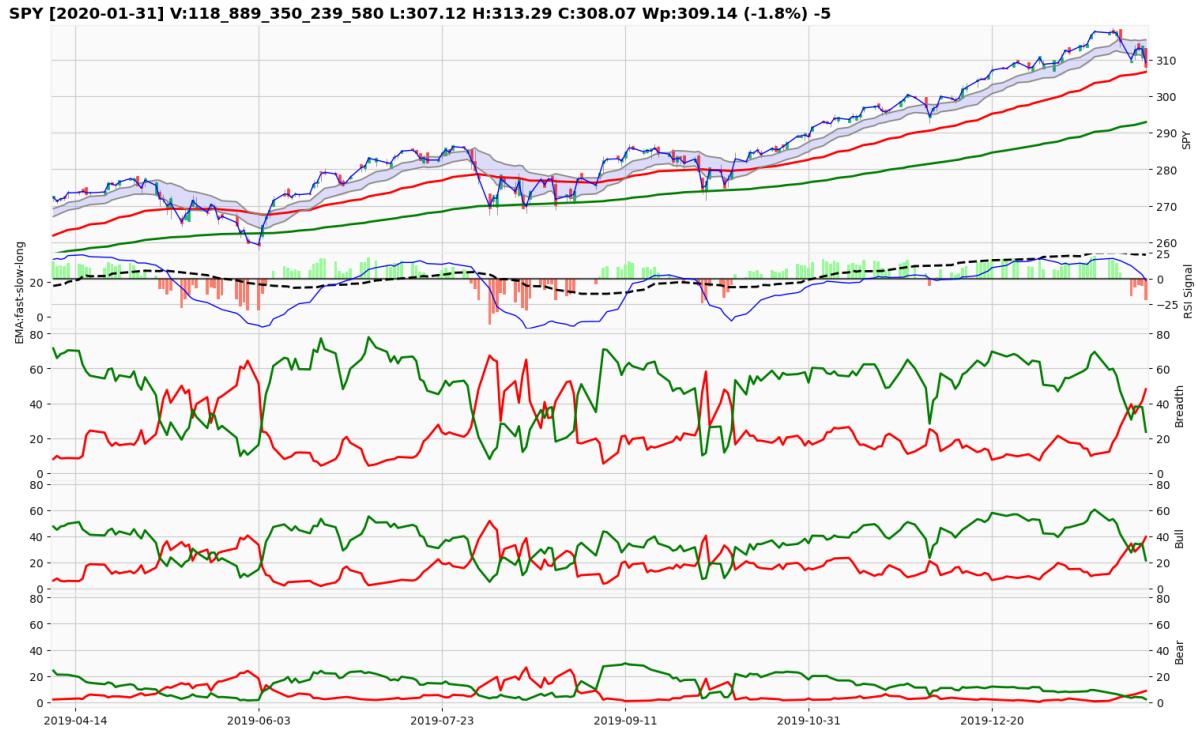
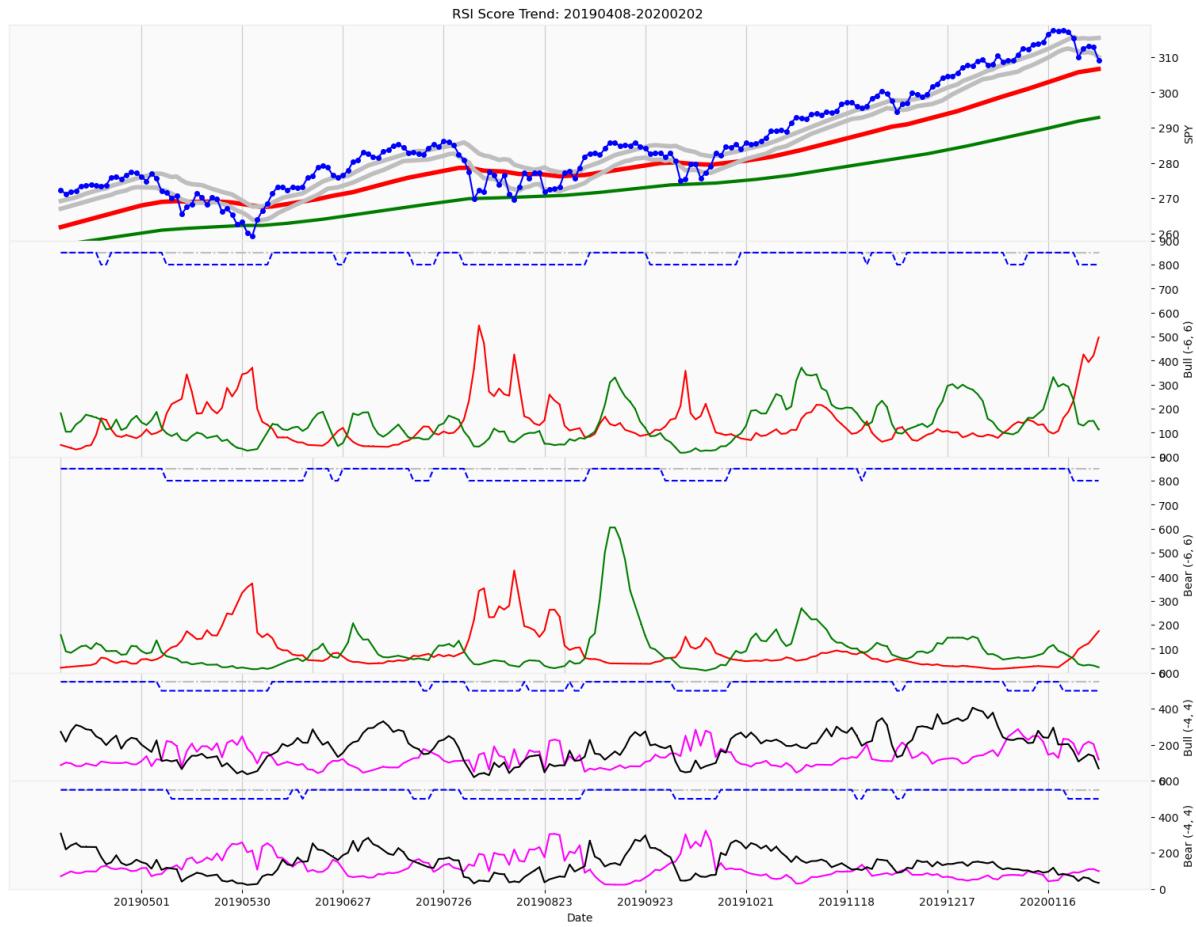


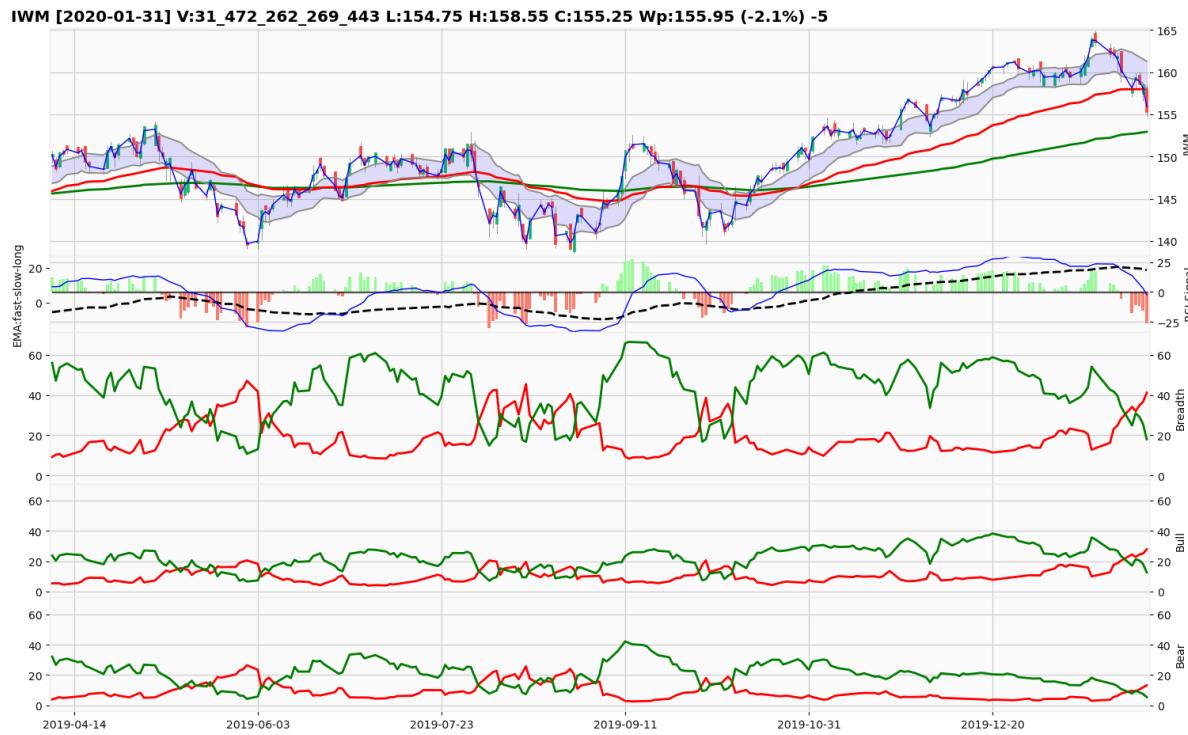
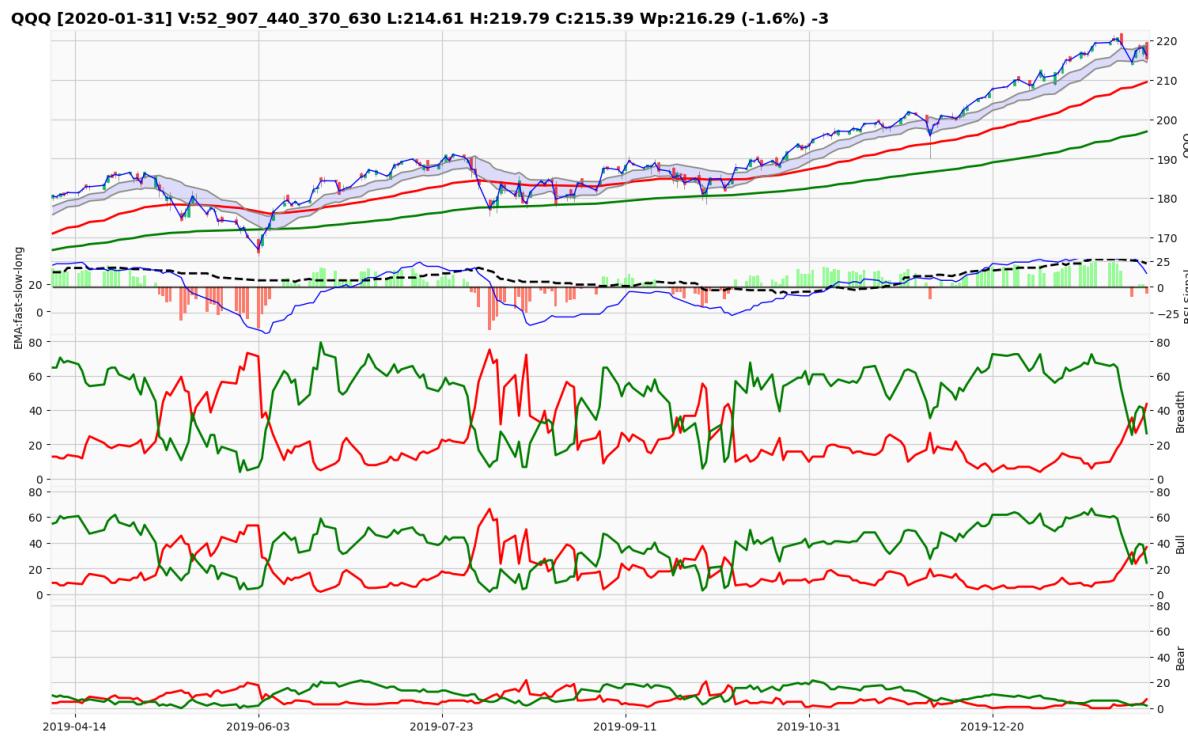
IWM [2019-02-01] V:17\_872\_936\_332\_597 L:141.80 H:142.86 C:142.38 Wp:142.36 (0.1%) 6

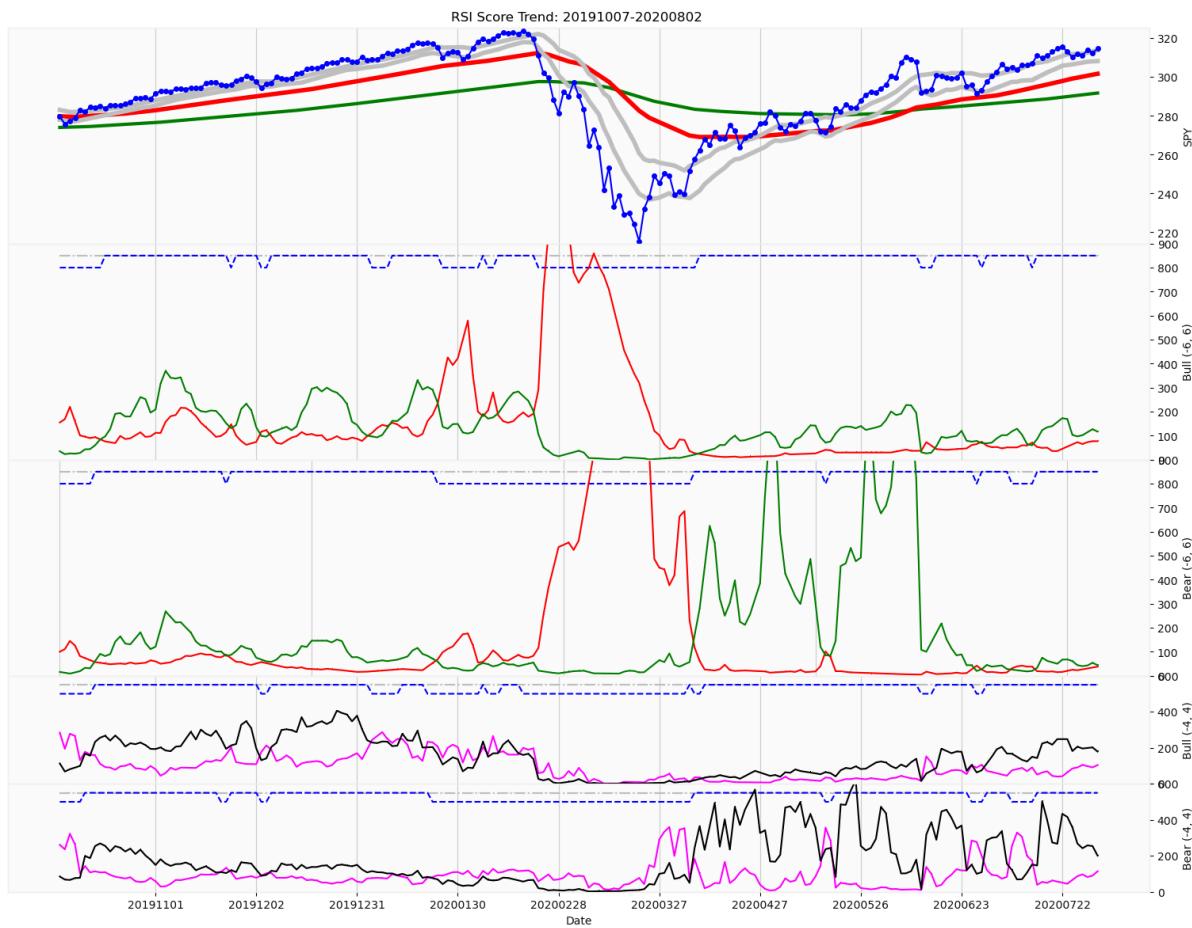




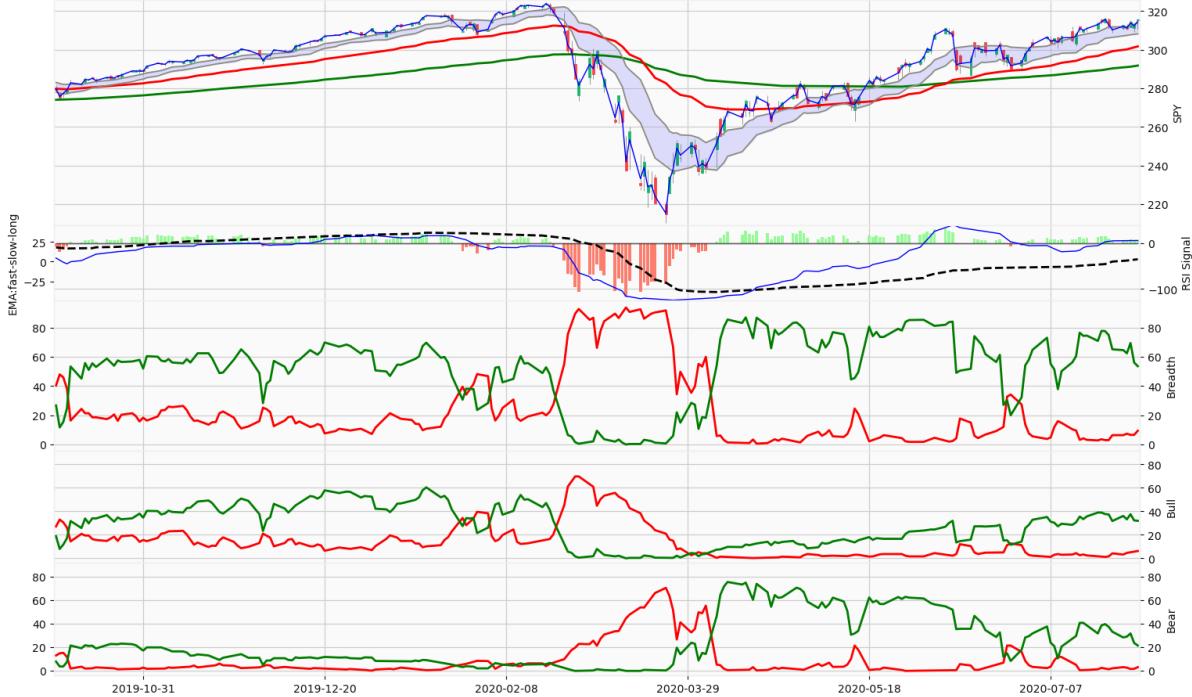


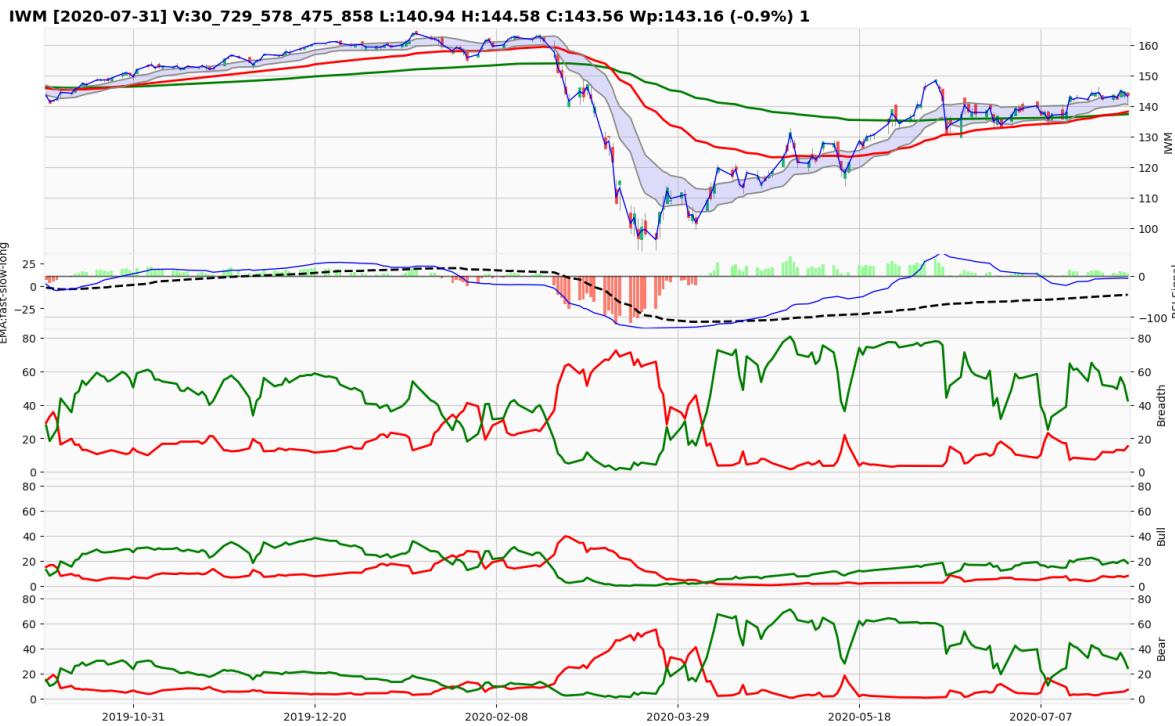
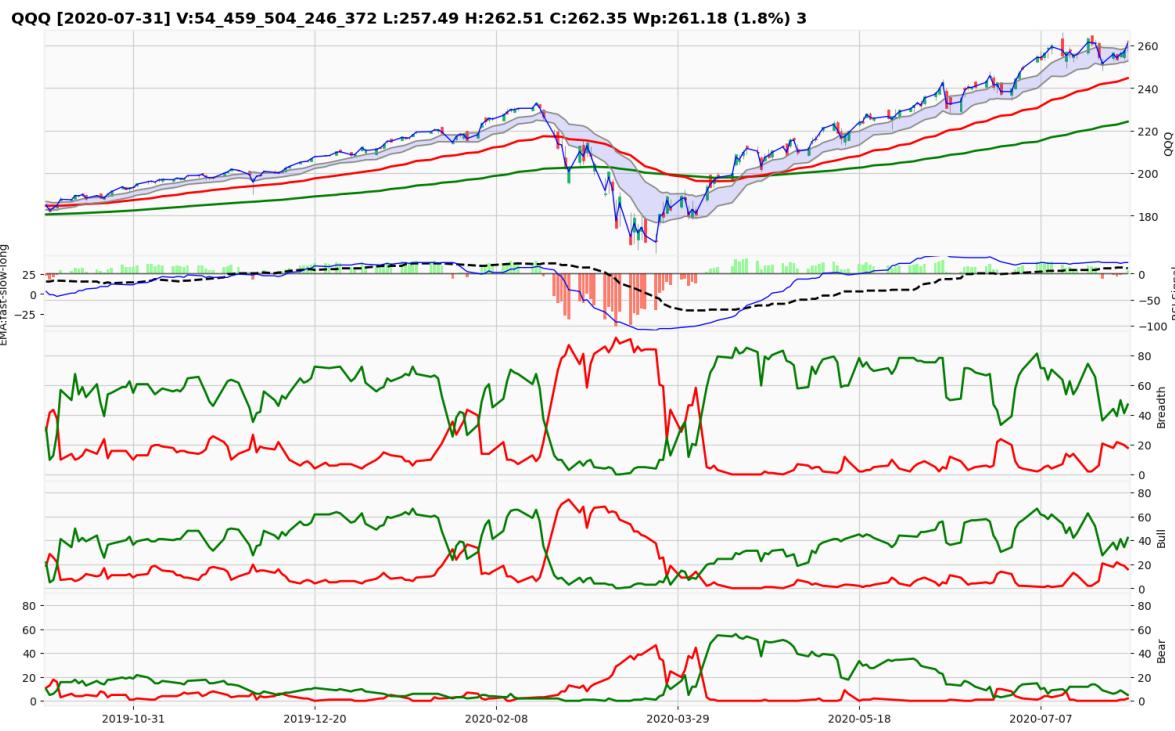


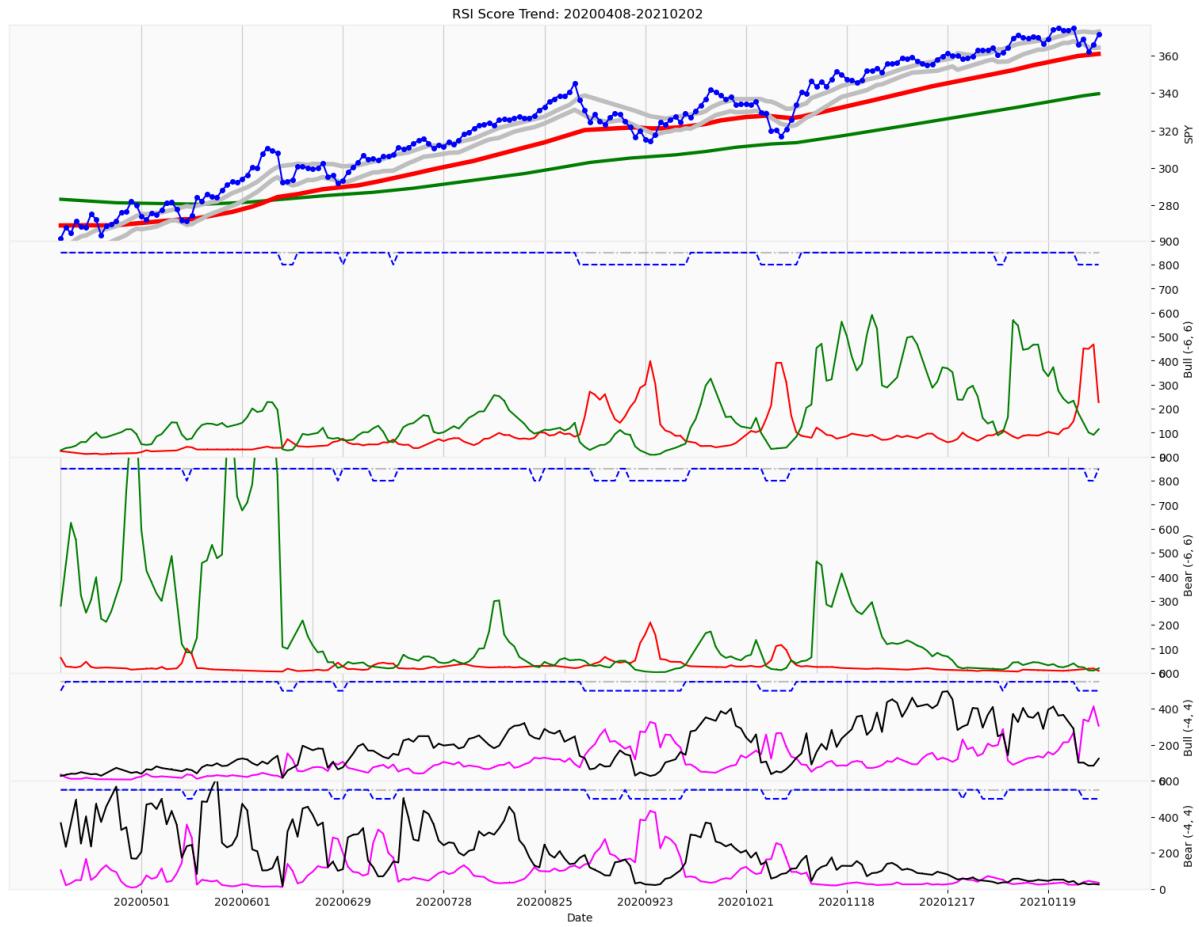




SPY [2020-07-31] V:88\_075\_496\_174\_167 L:310.88 H:316.01 C:315.90 Wp:314.67 (0.8%) 3



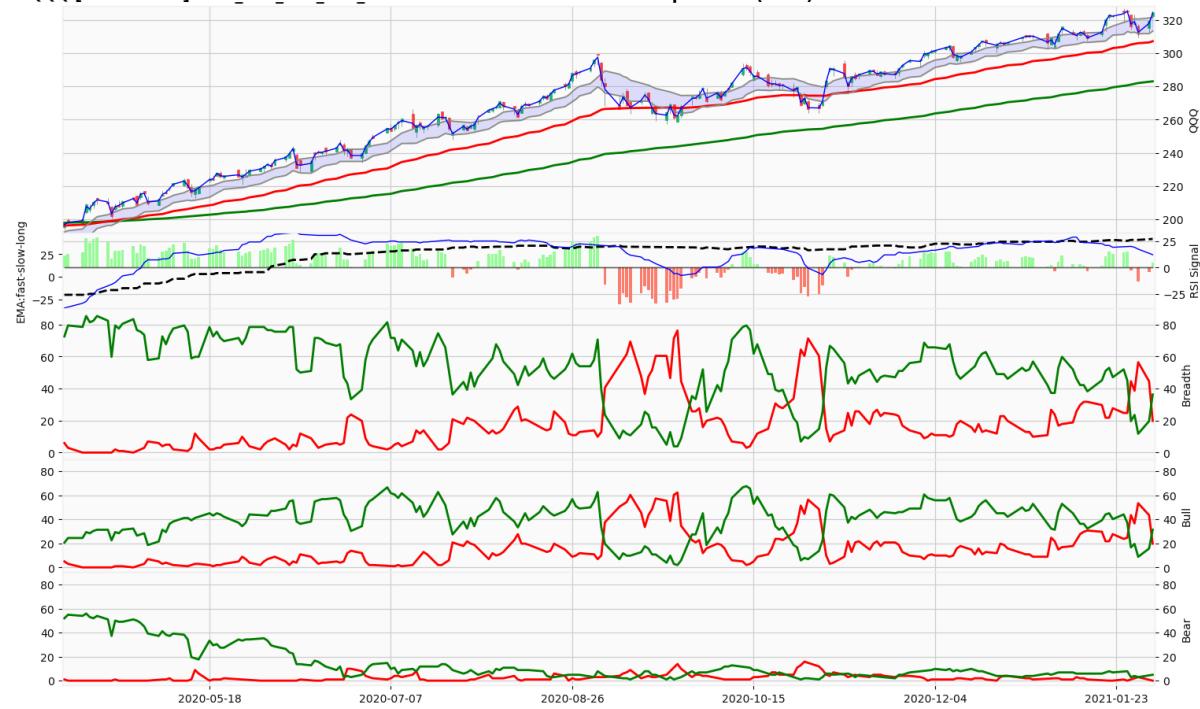




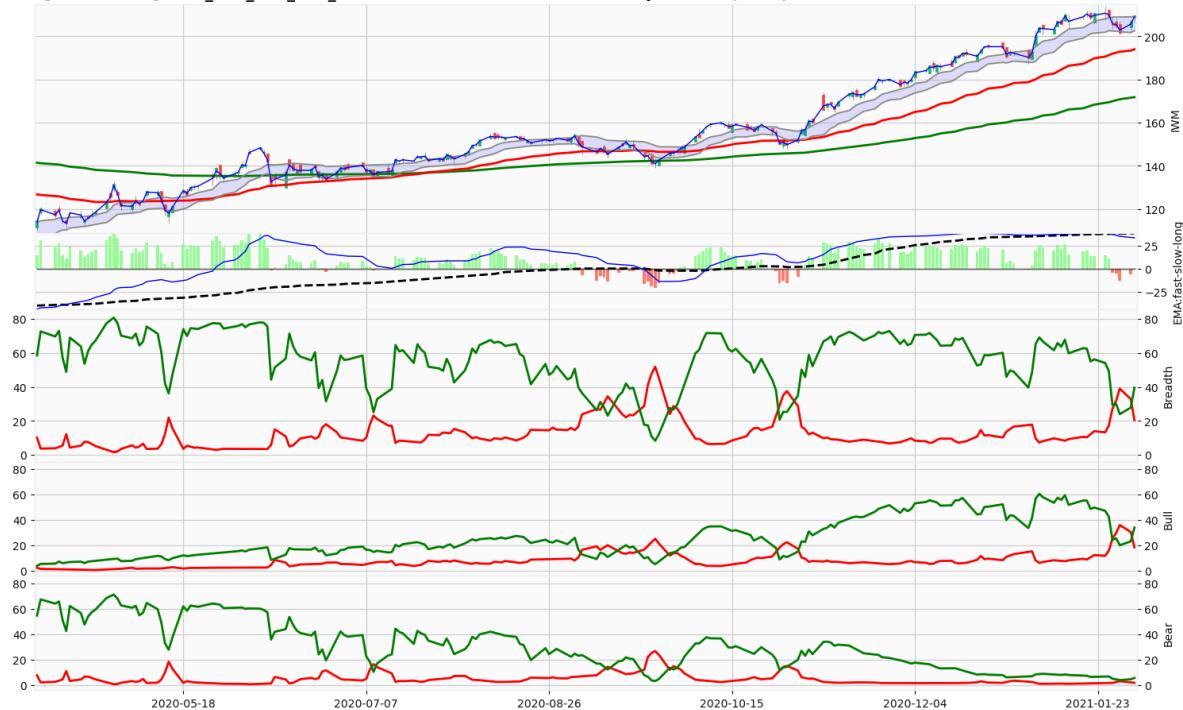
SPY [2021-02-02] V:66\_070\_022\_095\_394 L:367.09 H:373.83 C:372.20 Wp:371.33 (1.4%) -1

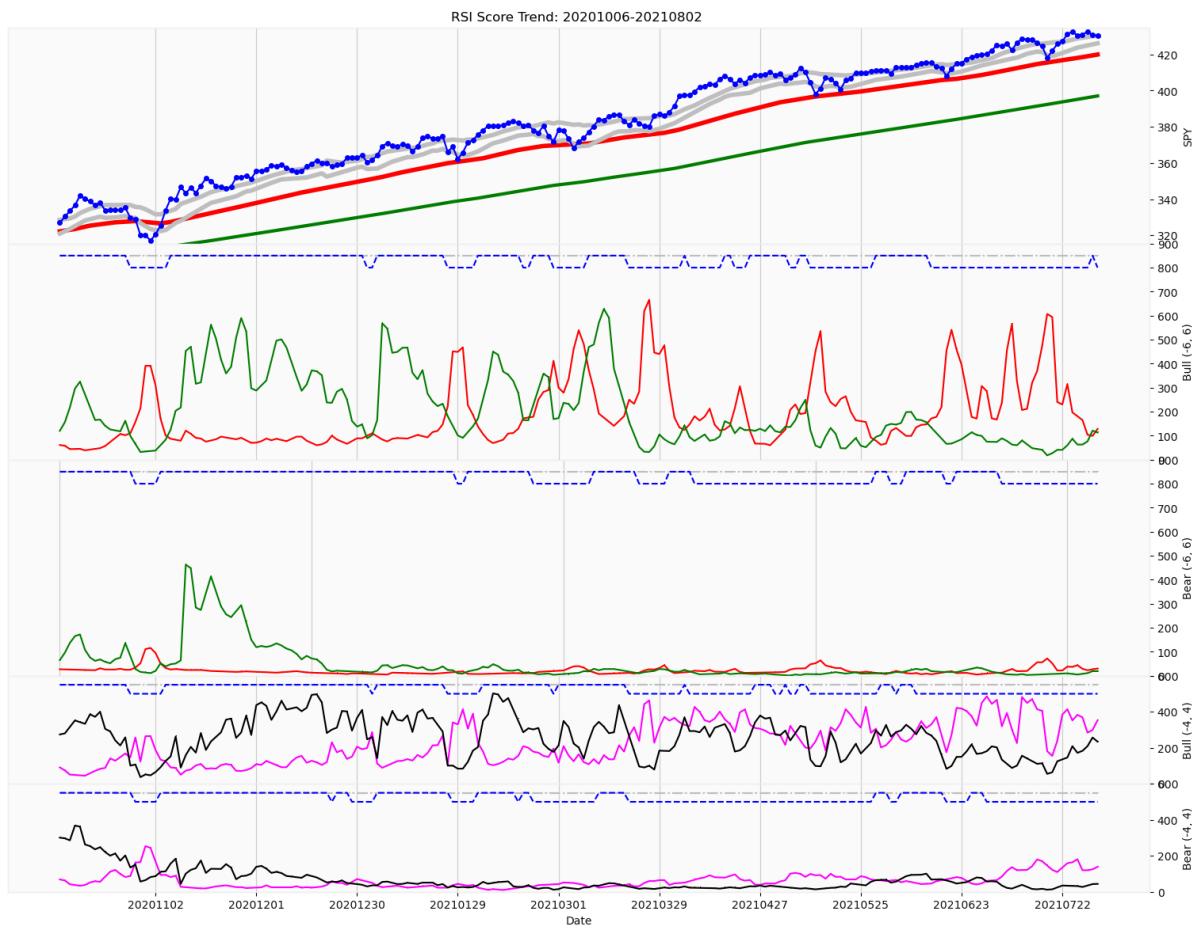


**QQQ [2021-02-02] V:34\_262\_567\_217\_895 L:322.15 H:325.69 C:324.50 Wp:324.21 (1.6%) 0**

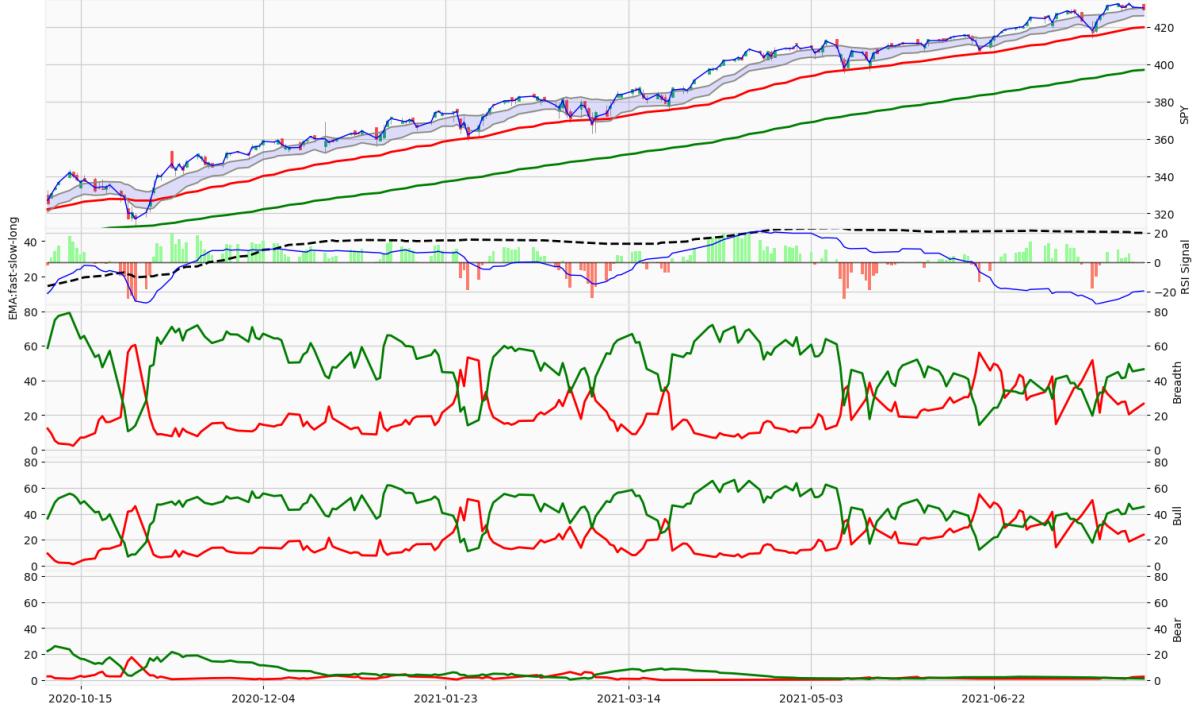


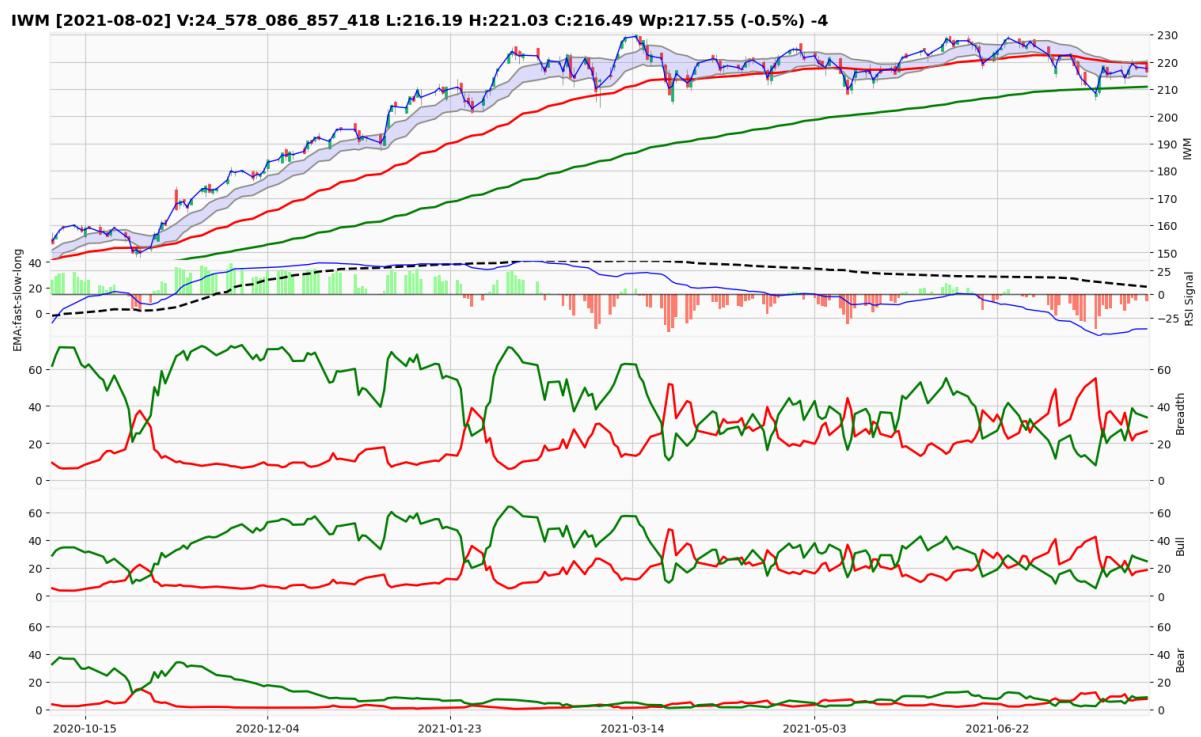
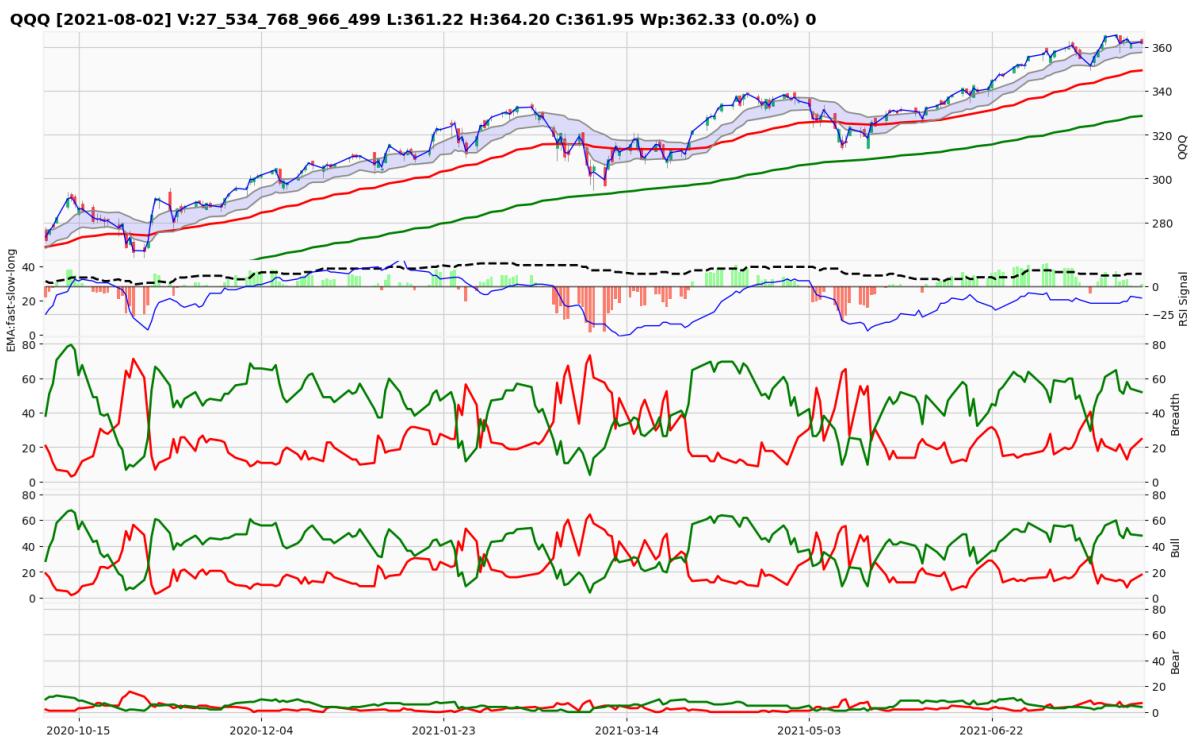
**IWM [2021-02-02] V:23\_214\_818\_565\_999 L:206.87 H:209.97 C:209.55 Wp:208.98 (1.4%) -1**

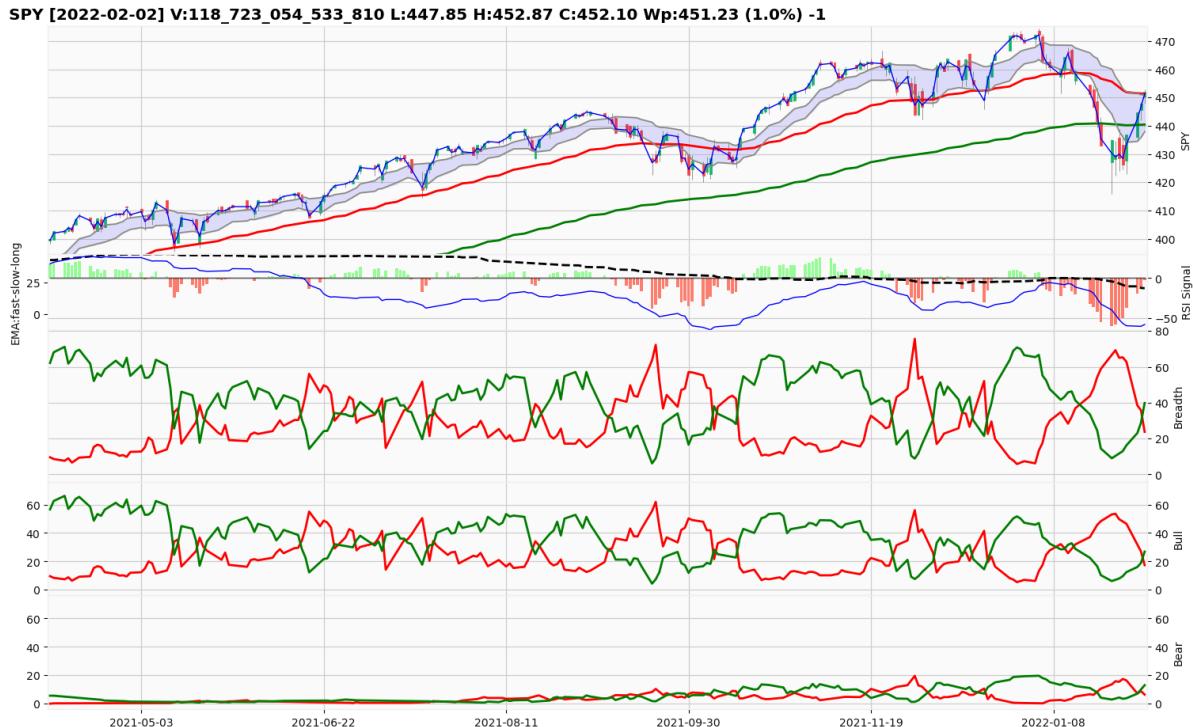
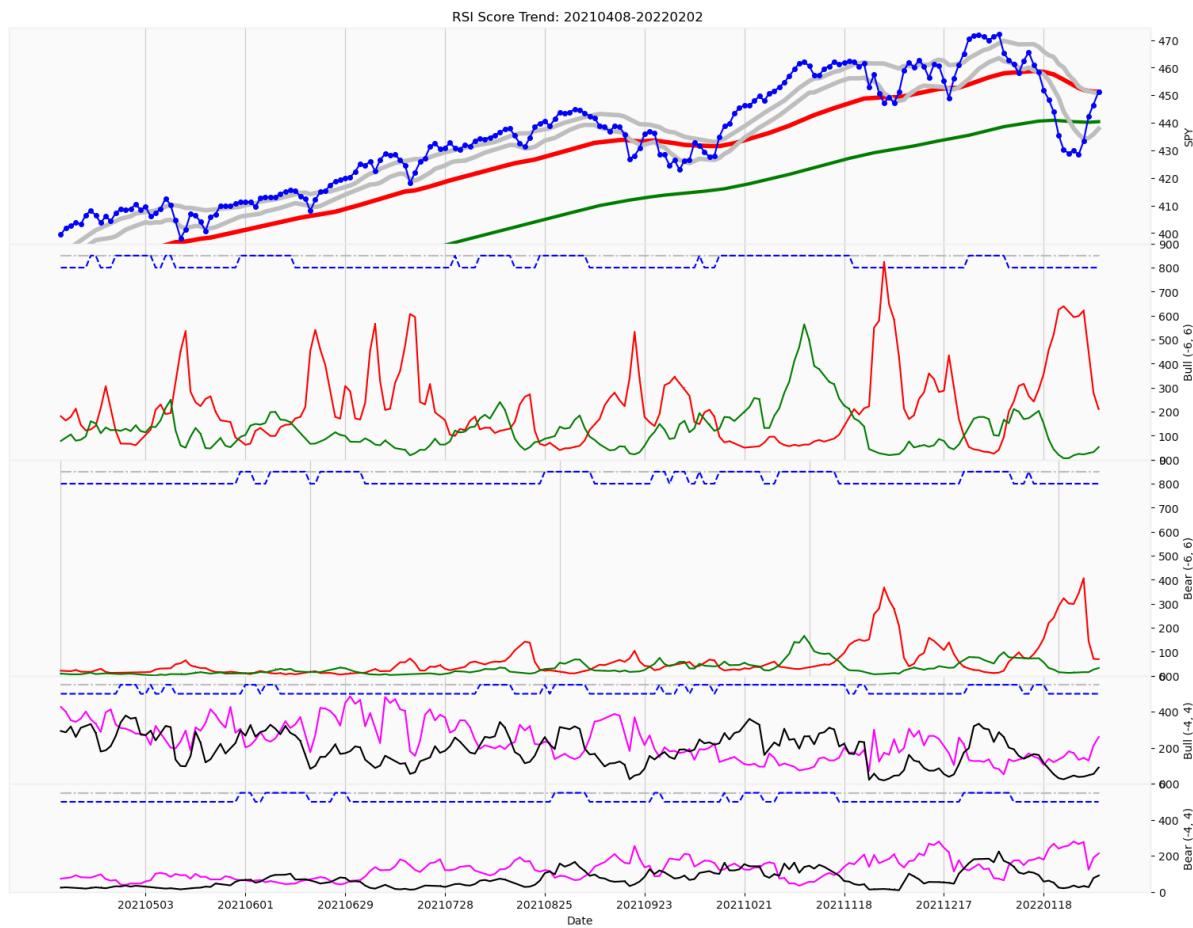




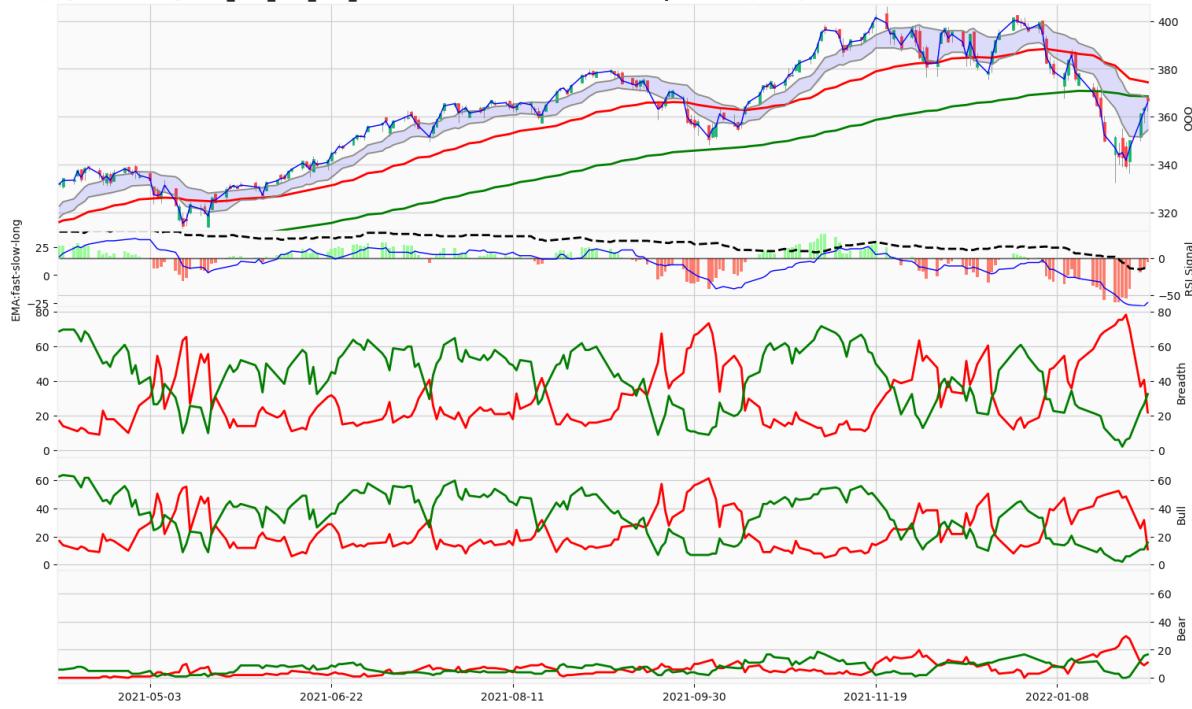
**SPY [2021-08-02] V:59\_866\_803\_003\_335 L:429.30 H:432.95 C:429.67 Wp:430.40 (-0.2%) 0**



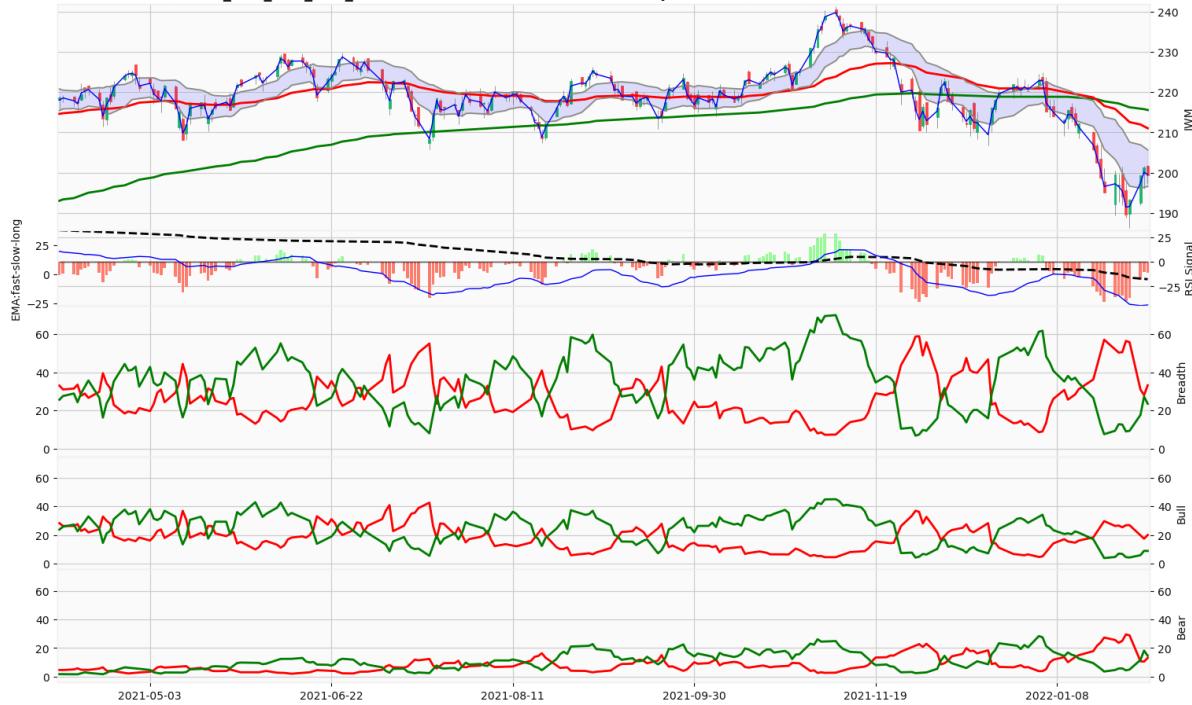


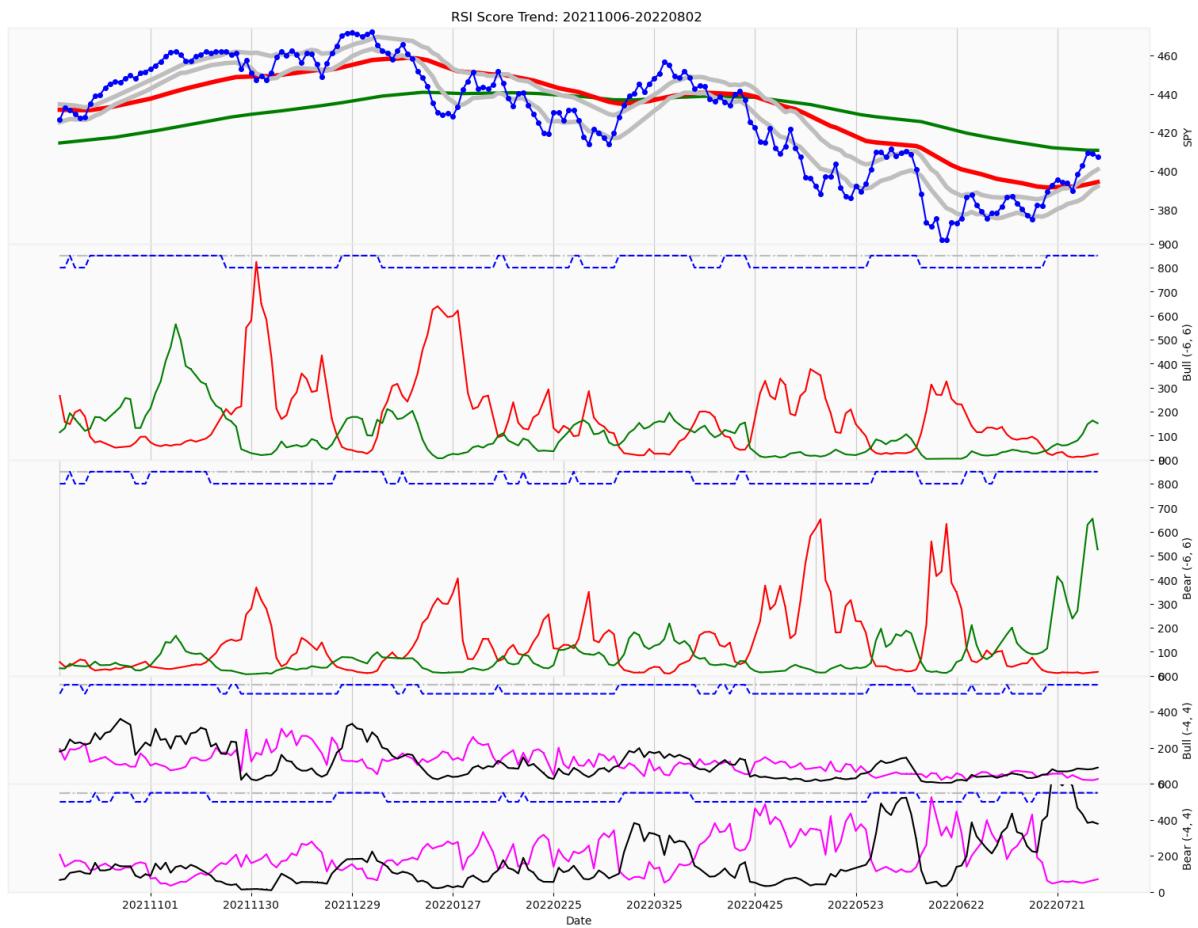


QQQ [2022-02-02] V:79\_167\_527\_212\_292 L:362.49 H:368.28 C:366.67 Wp:366.03 (0.8%) -4



IWM [2022-02-02] V:43\_195\_971\_496\_972 L:197.36 H:201.93 C:199.40 Wp:199.52 (-1.0%) -4





SPY [2022-08-02] V:63\_696\_657\_237\_268 L:405.15 H:411.31 C:406.39 Wp:407.31 (-0.7%) 6

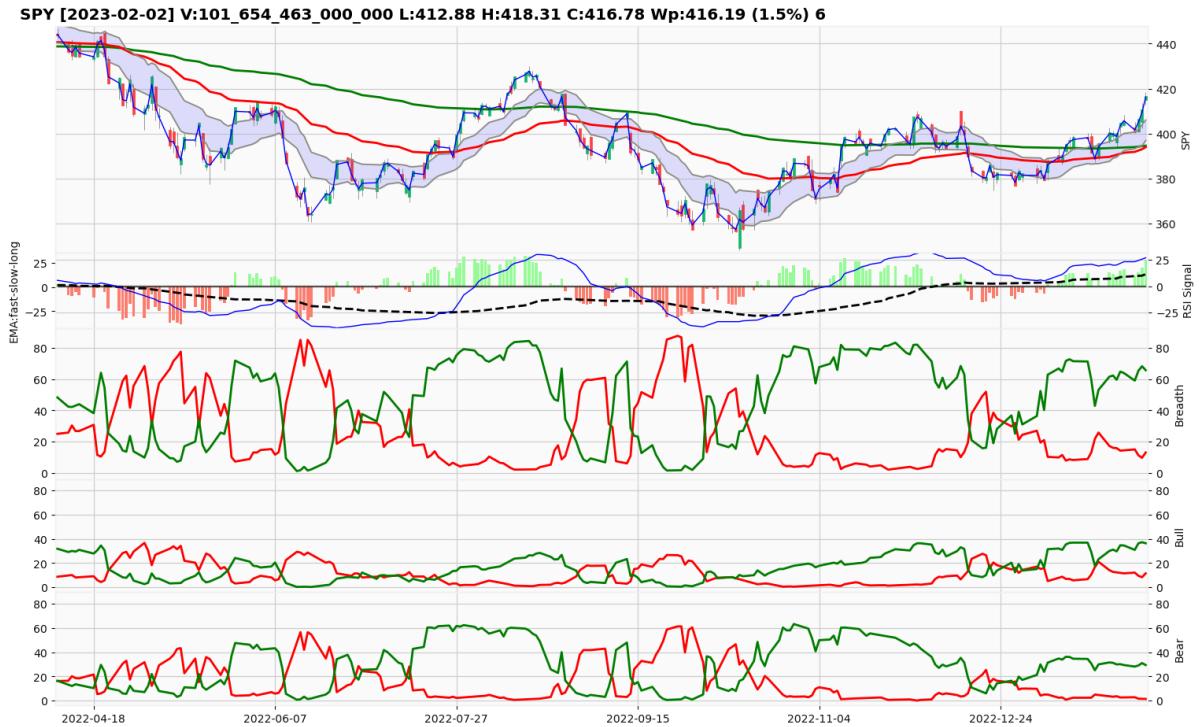
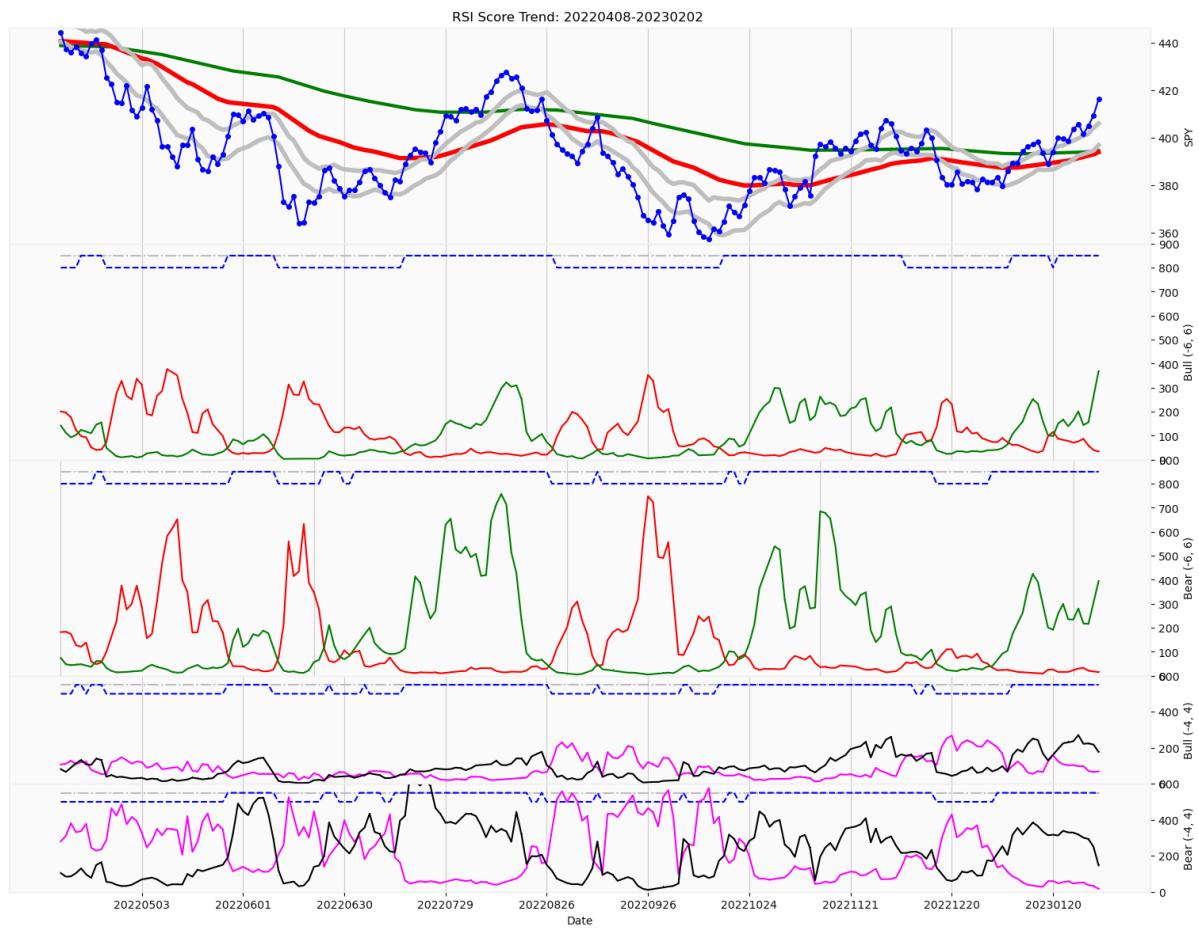


QQQ [2022-08-02] V:48\_764\_145\_668\_582 L:311.28 H:317.97 C:313.76 Wp:314.20 (-0.3%) 6

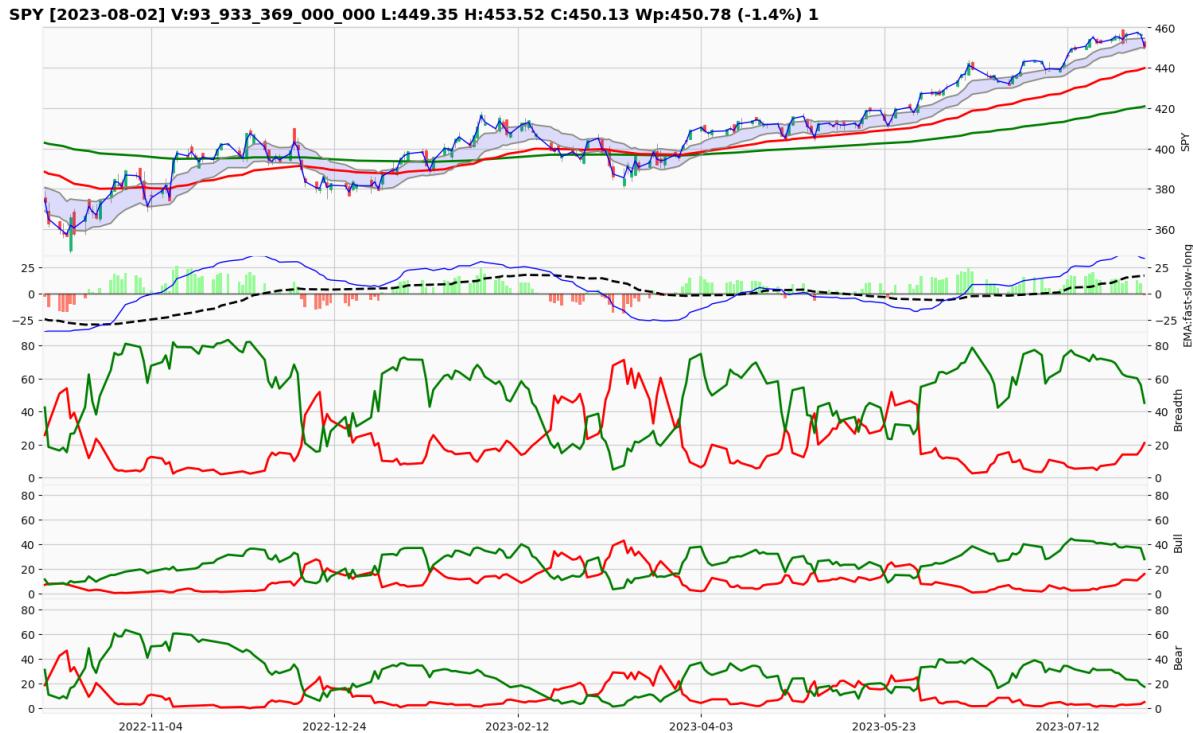
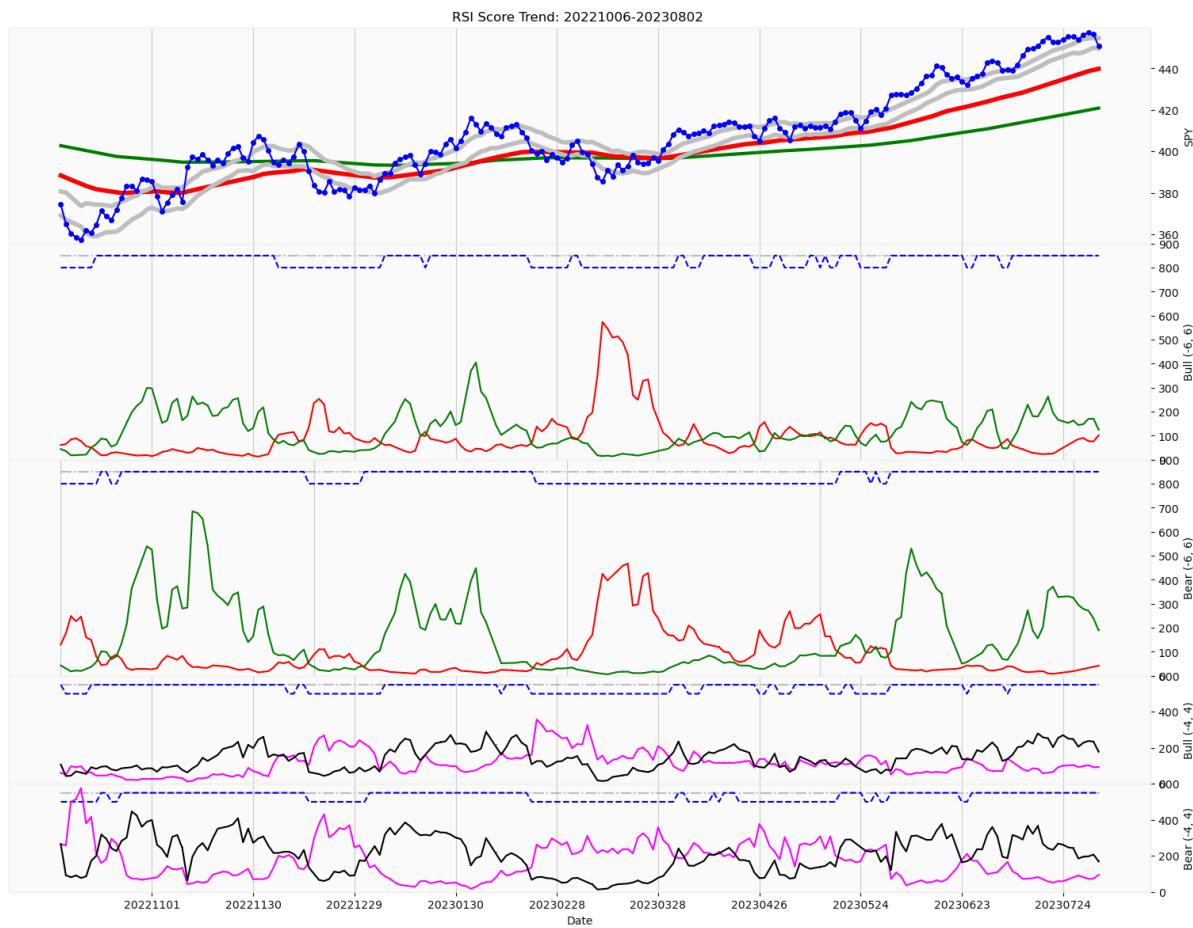


IWM [2022-08-02] V:17\_582\_953\_174\_212 L:184.53 H:188.05 C:186.04 Wp:186.16 (-0.1%) 6

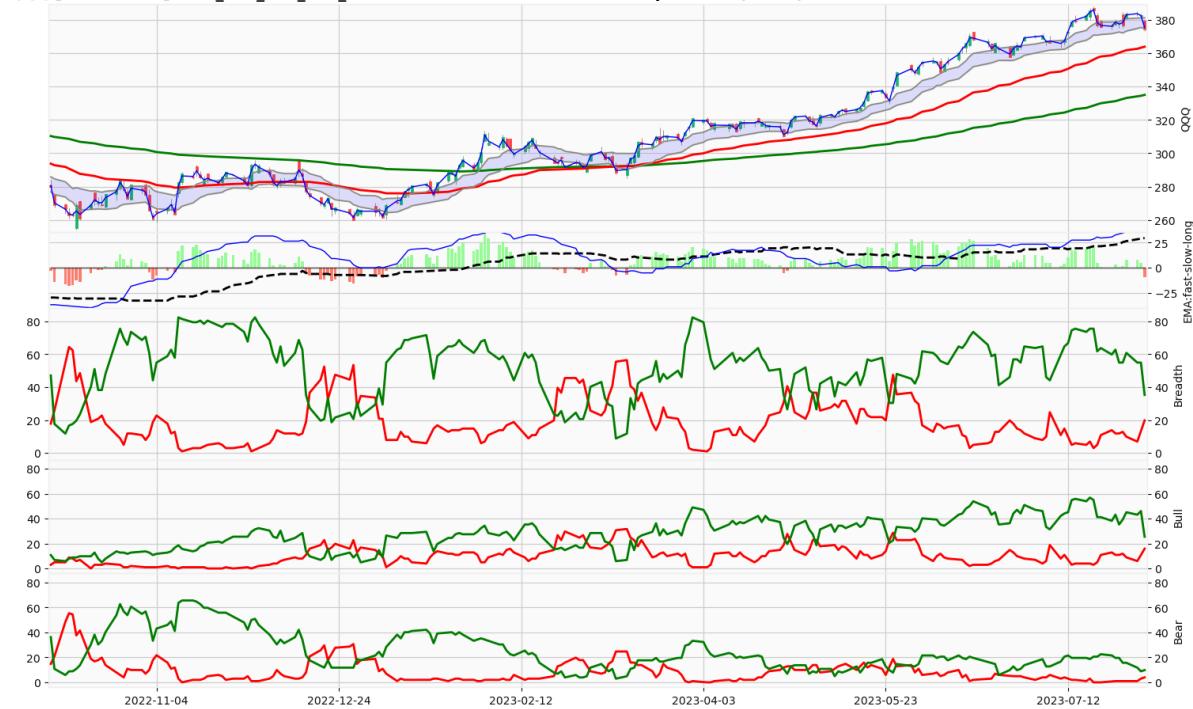








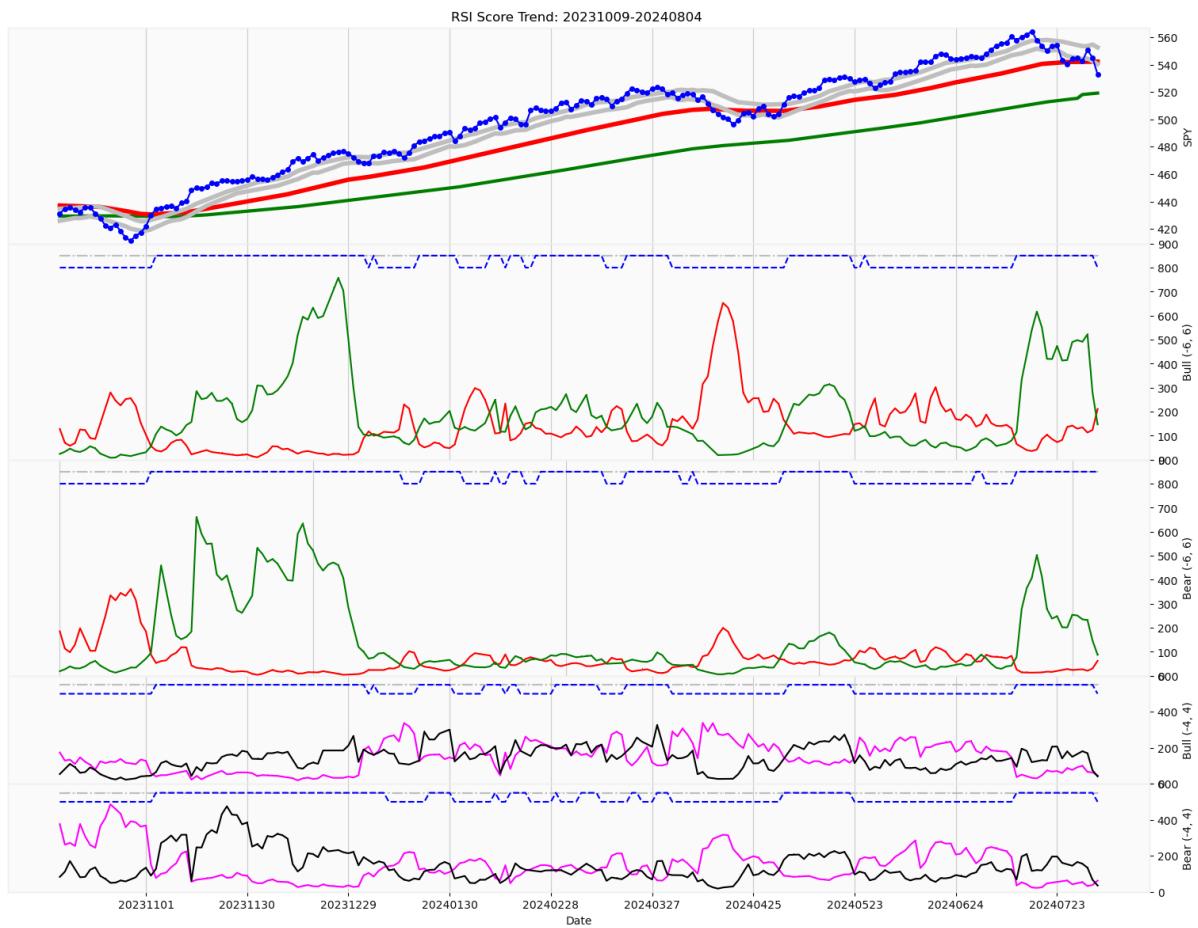
QQQ [2023-08-02] V:72\_687\_107\_000\_000 L:373.13 H:379.26 C:374.39 Wp:375.29 (-2.2%) -3



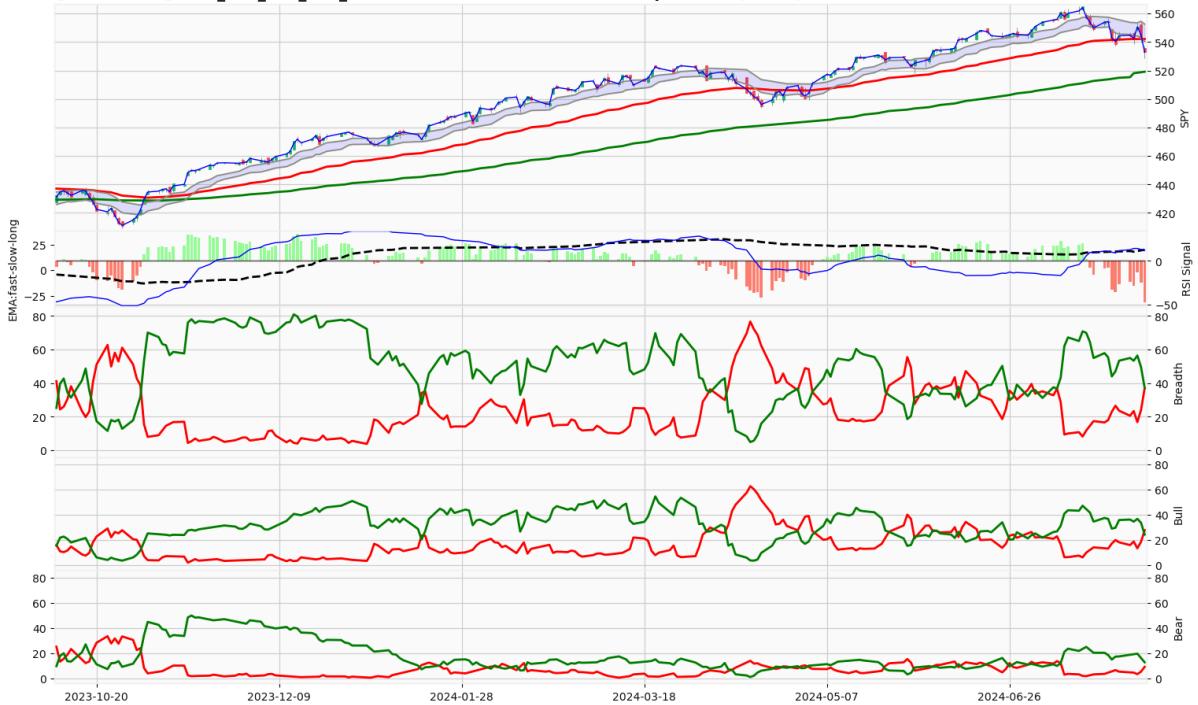
IWM [2023-08-02] V:28\_079\_650\_000\_000 L:194.09 H:196.09 C:195.11 Wp:195.10 (-1.4%) 1

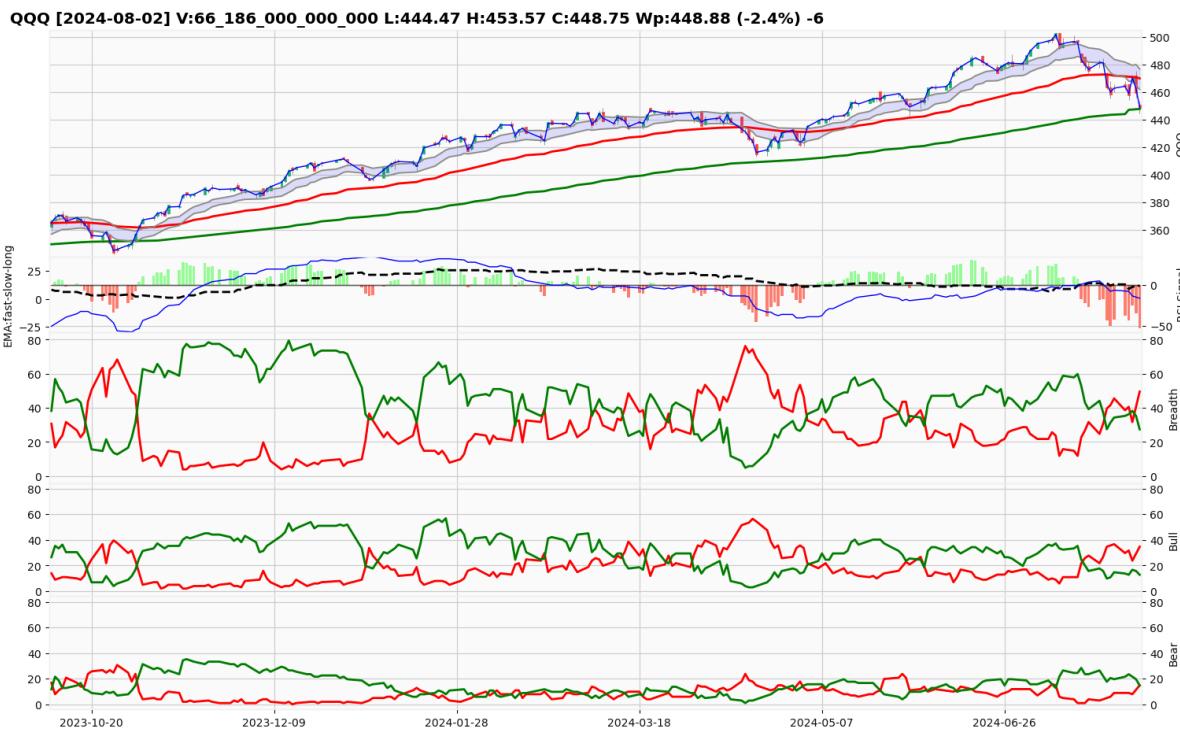


```
In [13]: rsi_date = "2024/08/04"
gen_rsi_trend_chart(rsi_date.replace("/", ""))
```



SPY [2024-08-02] V:82\_641\_600\_000\_000 L:528.60 H:536.99 C:532.90 Wp:532.85 (-1.9%) -6





```
In [9]: ts_stop = time()
print(datetime.now())
```

2024-08-04 15:14:34.088386

```
In [10]: time_per_date = (ts_stop-ts_start)/float(len(dates))
```

```
In [11]: print(f"sec per date: {time_per_date}")
```

sec per date: 3.349830056491651

In [ ]: