1. 生成项目并安装模块

```
create-react-app zhufeng-connect
cd zhufeng-connect
cnpm i react-router-dom redux  react-redux connected-react-router -S
```

2. 跑通项目

src\index.js

```
import React from 'react';
import ReactDOM from 'react-dom';
import {Route,Link} from 'react-router-dom';
import Home from './components/Home';
import Counter from './components/Counter';
import { ConnectedRouter } from 'connected-react-router'
import history from './history';
import store from './store';
import { Provider } from 'react-redux'
ReactDOM.render(

    <>
      Home
      Counter

    </>

,document.getElementById('root'));
```

src\store\index.js

```
import { applyMiddleware, createStore } from 'redux'
import { routerMiddleware } from 'connected-react-router'
import history from '../history';
import reducers from './reducers';
const store = applyMiddleware(routerMiddleware(history))(createStore)(reducers);
window.store = store;
export default  store;
```

src\history.js

```
import { createHashHistory } from 'history'
let history = createHashHistory();
export default history;
```

src\store\reducers\index.js

```
import { combineReducers } from 'redux'
import { connectRouter } from 'connected-react-router'
import counter from './counter';
import history from '../../history';
export default  combineReducers({
  router: connectRouter(history),
  counter
})
```

src\store\reducers\counter.js

```
import * as types from '../action-types';
export default function (state={number:0}, action) {
    switch (action.type) {
        case types.INCREMENT:
            return {number:state.number+1};
        case types.DECREMENT:
            return {number:state.number-1};
        default:
            return state;
    }
}
```

src\store\action-types.js

```
export const INCREMENT = 'INCREMENT';
export const DECREMENT = 'DECREMENT';
```

src\store\actions\counter.js

```
import  * as types from '../action-types';
import { push } from 'connected-react-router';
export default {
    increment(){
        return {type:types.INCREMENT}
    },
    decrement(){
        return {type:types.DECREMENT}
    },
    go(path){
        return push(path);
    }
}
```

src\components\Home.js

```
import React,{Component} from 'react';
export default class Home extends Component{
    render() {
        return (
            <div>Homediv>
        )
    }
}
```

src\components\Counter.js

```
import React, { Component } from 'react'
import {connect} from 'react-redux';
import actions from '../store/actions/counter';
class Counter extends Component {
  render() {
    return (
      <>
          <p>{this.props.number}p>
          <button onClick={this.props.increment}>+button>
          <button onClick={this.props.decrement}>-button>
          <button onClick={()=>this.props.go('/')}>Homebutton>
      </>
    )
  }
}
export default connect(
    state=>state.counter,
    actions
)(Counter);
```

3.实现connected-react-router

src\connected-react-router\index.js

```
import push from './push';
import routerMiddleware from './routerMiddleware';
import ConnectedRouter from './ConnectedRouter';
import connectRouter from './connectRouter';
export {push,routerMiddleware,ConnectedRouter,connectRouter};
```

src\connected-react-router\constants.js

```
export const CALL_HISTORY_METHOD = '@@router/CALL_HISTORY_METHOD';
export const LOCATION_CHANGE = '@@router/LOCATION_CHANGE'
```

src\connected-react-router\push.js

```
import {CALL_HISTORY_METHOD} from './constants';
export default function (path) {
    return {
        type: CALL_HISTORY_METHOD,
        payload: {
            method: 'push',
            path
        }
    };
}
```

src\connected-react-router\routerMiddleware.js

```
import {CALL_HISTORY_METHOD} from './constants';
export default function routerMiddleware(history) {
    return function (store) {
      return function (next) {
        return function (action) {
          if (action.type !== CALL_HISTORY_METHOD) {
            return next(action);
          }
          let {method,path} = action.payload;
          history[method](path);
        };
      };
    };
  };
```

src\connected-react-router\ConnectedRouter.js

```
import React, { Component } from 'react'
import { Router } from 'react-router'
import { LOCATION_CHANGE } from './constants';
import { ReactReduxContext } from 'react-redux';
export default class ConnectedRouter extends Component {
  static contextType = ReactReduxContext
  componentDidMount() {
    this.unlisten = this.props.history.listen((location, action) => {
      this.context.store.dispatch({
        type: LOCATION_CHANGE,
        payload: {
          location,
          action
        }
      });
    });
  }
  componentWillUnmount() {
    this.unlisten();
  }
  render() {
    const { history, children } = this.props
    return (
      <Router history={history}>
        {children}
      Router>
    )
  }
}
```

src\connected-react-router\connectRouter.js

```
import {LOCATION_CHANGE} from './constants';
export default function (history) {
    let initState = {location:history.location,action:history.action};
    return function (state=initState, action) {
        if (action.type === LOCATION_CHANGE) {
            return {
                location: action.payload.location,
                action: action.payload.action
            }
        }
        return state;
    }
}
```

参考