Discuss one thing you notice that is different between the two emails that might relate to the identification of spam.

The first email contains plain text, however the second emails is written in HTML syntax, and contains repeated phrases like '<html','<head','<body'. This HTML syntax may be repetitive causing for the false identification of spam in the second email.

0.0.1 Question 3

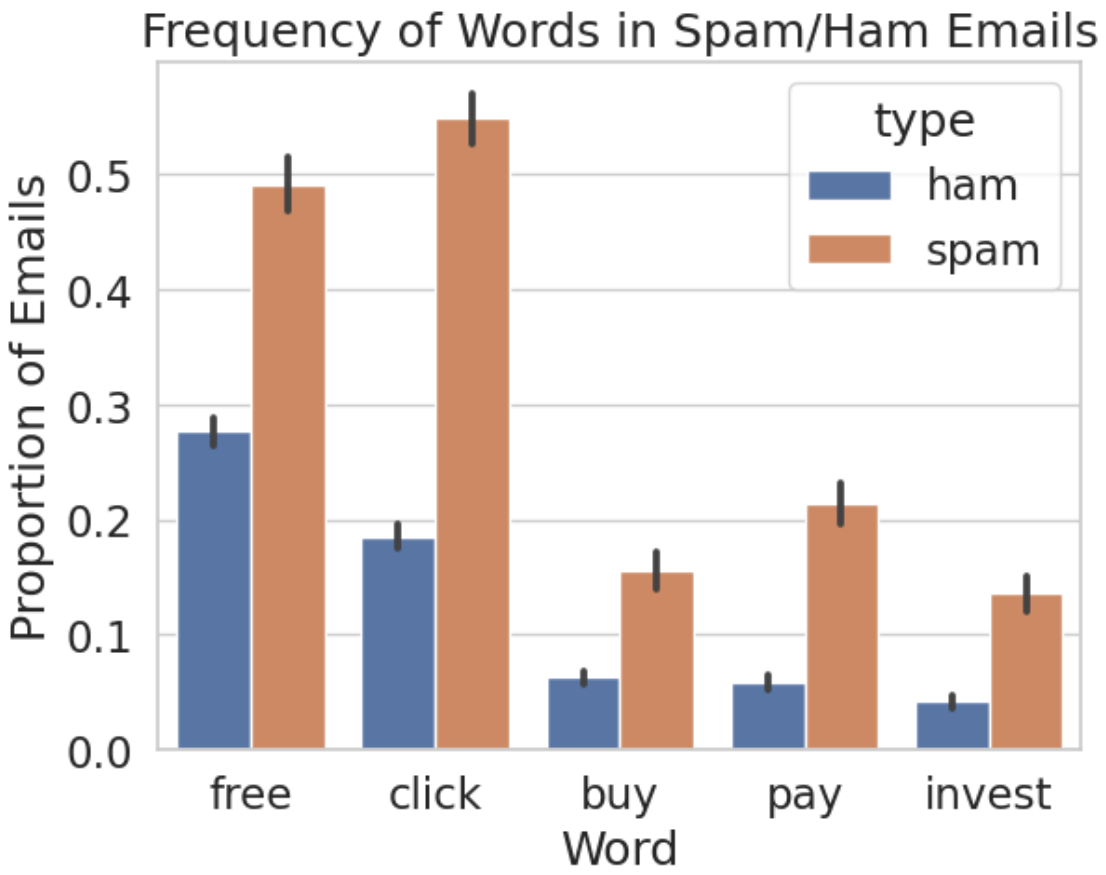
Create a bar chart like the one above comparing the proportion of spam and ham emails containing certain words. Choose a set of words that are different from the ones above, but also have different proportions for the two classes. Make sure to only consider emails from `train`.

In [12]:
```
train = train.reset_index(drop=True) # We must do this in order to preserve the ordering of em

lis = ["free","click","buy","pay", "invest"]
resid = words_in_texts(lis, train['email'])
x = pd.DataFrame(data=resid, columns=lis)
x["type"] = train["spam"]
x["type"] = x["type"].replace({0: "ham", 1:"spam"})

sns.barplot(data=x.melt("type"),y='value', x='variable', hue='type')
plt.title("Frequency of Words in Spam/Ham Emails")
plt.ylabel("Proportion of Emails")
plt.xlabel("Word")
```

Out[12]: Text(0.5, 0, 'Word')



0.0.2 Question 6c

Comment on the results from 6a and 6b. For **each** of FP, FN, accuracy, and recall, briefly explain why we see the result that we do.

Type your answer here, replacing this text.

0.0.3 Question 6e

Are there more false positives or false negatives when using the logistic regression classifier from Question 5?

From question 5, the classifier results in higher false negatives and a lower number of false positives from the Logistic Regression.

0.0.4 Question 6f

1. Our logistic regression classifier got 75.76% prediction accuracy (number of correct predictions / total). How does this compare with predicting 0 for every email?
2. Given the word features we gave you above, name one reason this classifier is performing poorly. Hint: Think about how prevalent these words are in the email set.
3. Which of these two classifiers would you prefer for a spam filter and why? Describe your reasoning and relate it to at least one of the evaluation metrics you have computed so far.

1) The logistic regression classifying a 75.76% prediction accuracy is more accurate than the zero_predictor which yielded 74.47%.

2) The selected words are not prevalent in the dataset, leaving prediction accuracy to be based on words that are considered 'poor' for the detection of spam in emails.

3) I prefer the Logistic Regression as a spam filter, since the zero_predictor only classifies emails as 'ham', which is harly useful for filtering any 'spam' at all. Logistic Regression also yields a higher prediction accuracy rate, while also filtering out 'spam' words.

0.1 Submission

Make sure you have run all cells in your notebook in order before running the cell below, so that all images/graphs appear in the output. The cell below will generate a zip file for you to submit. **Please save before exporting!**

```
In [25]: # Save your notebook first, then run this cell to export your submission.
         grader.export(run_tests=True)
```

```
Running your submission against local test cases…


Your submission received the following results when run against available test cases:

    q2 results: All test cases passed!

    q4 results: All test cases passed!

    q5 results: All test cases passed!

    q6a results: All test cases passed!

    q6b results: All test cases passed!
```

```
    q6d results: All test cases passed!
```

```
<IPython.core.display.HTML object>
```