WHAT IS CLAIMED IS:

1. A method of decoding a bitstream of encoded data to reconstruct a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

decoding the encoded data based on the determined context to reconstruct a bit value, and

updating the context based on that reconstructed bit value,

wherein the reconstructed bit values form the decoded significance map, wherein the transform unit is sized 4x4, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
4,  5,  2,  3,
6,  6,  7,  7,
8,  8,  7,
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

2. The method claimed in claim 1, wherein determining includes selecting the partition set from among a plurality of partition sets based upon text type and transform unit size, and wherein the text type is luma and the transform unit size is 4x4.

3. A method of decoding a bitstream of encoded data to reconstruct a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

decoding the encoded data based on the determined context to reconstruct a bit value, and

updating the context based on that reconstructed bit value, wherein the reconstructed bit values form the decoded significance map, wherein the

transform unit is sized 8x8, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
 0,  1,  2,  2,  3,  3,  4,  4,
 1,  1,  2,  2,  3,  3,  4,  4,
 5,  5,  6,  6,  7,  7,  4,  4,
 5,  5,  6,  6,  7,  7,  4,  4,
 8,  8,  9,  9,  6,  7, 10, 10,
 8,  8,  9,  9,  9, 10, 10, 10,
11, 11, 11, 11, 10, 10, 10, 10,
11, 11, 11, 11, 10, 10, 10
```

and wherein the above integers represent the contexts assigned to the bit positions of a 8x8 block significance map.

4. The method claimed in claim 3, wherein determining includes selecting the partition set from among a plurality of partition sets based upon text type and transform unit size, and wherein the text type is luma and the transform unit size is 8x8.

5. A method of decoding a bitstream of encoded data to reconstruct a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

decoding the encoded data based on the determined context to reconstruct a bit value, and

updating the context based on that reconstructed bit value, wherein the reconstructed bit values form the decoded significance map, wherein the transform unit is sized 4x4, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
1,  1,  2,  3,
4,  4,  5,  5,
3,  3,  5
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

6. A method of decoding a bitstream of encoded data to reconstruct a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

decoding the encoded data based on the determined context to reconstruct a bit value, and

updating the context based on that reconstructed bit value, wherein the reconstructed bit values form the decoded significance map, wherein the transform unit is sized 8x8, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  0,  1,  1,  2,  2,  3,  3,
0,  0,  1,  1,  2,  2,  3,  3,
1,  1,  1,  1,  2,  2,  3,  3,
1,  1,  1,  1,  2,  2,  3,  3,
2,  2,  2,  2,  1,  2,  3,  3,
2,  2,  2,  2,  2,  3,  3,  3,
3,  3,  3,  3,  3,  3,  3,  3,
3,  3,  3,  3,  3,  3,  3
```

and wherein the above integers represent the contexts assigned to the bit positions of a 8x8 block significance map.

7. A decoder for decoding a bitstream of encoded data to reconstruct a significance map for a transform unit, the decoder comprising:

a processor;

a memory; and

a decoding application stored in memory and containing instructions for configuring perform the method claimed in any one of claims 1 to 6.

8. A method for encoding a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

encoding a bit value at that bit position based on the determined context to generate encoded data, and

updating the context based on that bit value,

wherein the encoded data forms an encoded significance map, wherein the transform

unit is sized 4x4, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
4,  5,  2,  3,
6,  6,  7,  7,
8,  8,  7,
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

9. A method for encoding a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

encoding a bit value at that bit position based on the determined context to generate encoded data, and

updating the context based on that bit value,

wherein the encoded data forms an encoded significance map, wherein the transform unit is sized 8x8, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
 0,  1,  2,  2,  3,  3,  4,  4,
 1,  1,  2,  2,  3,  3,  4,  4,
 5,  5,  6,  6,  7,  7,  4,  4,
 5,  5,  6,  6,  7,  7,  4,  4,
 8,  8,  9,  9,  6,  7, 10, 10,
 8,  8,  9,  9,  9, 10, 10, 10,
11, 11, 11, 11, 10, 10, 10, 10,
11, 11, 11, 11, 10, 10, 10
```

and wherein the above integers represent the contexts assigned to the bit positions of a 8x8 block significance map.

10. A method for encoding a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

encoding a bit value at that bit position based on the determined context to generate encoded data, and

updating the context based on that bit value,

wherein the encoded data forms an encoded significance map, wherein the transform unit is sized 4x4, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
1,  1,  2,  3,
4,  4,  5,  5,
3,  3,  5
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

11. A method for encoding a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

encoding a bit value at that bit position based on the determined context to generate encoded data, and

updating the context based on that bit value,

wherein the encoded data forms an encoded significance map, wherein the transform unit is sized 8x8, and wherein the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  0,  1,  1,  2,  2,  3,  3,
0,  0,  1,  1,  2,  2,  3,  3,
1,  1,  1,  1,  2,  2,  3,  3,
1,  1,  1,  1,  2,  2,  3,  3,
2,  2,  2,  2,  1,  2,  3,  3,
2,  2,  2,  2,  2,  3,  3,  3,
3,  3,  3,  3,  3,  3,  3,  3,
3,  3,  3,  3,  3,  3,  3
```

and wherein the above integers represent the contexts assigned to the bit positions of a 8x8 block significance map.

12. An encoder for encoding a significance map for a transform unit, the encoder comprising:

a processor;

a memory storing the significance map; and

an encoding application stored in memory and containing instructions for configuring the processor to perform the method claimed in any one of claims 8 to 11.

13. A non-transitory processor-readable medium storing processor-executable instructions which, when executed, configures one or more processors to perform the method claimed in any one of claims 1 to 6 and 8 to 11.

WHAT IS CLAIMED IS:

1. A method of decoding a bitstream of encoded data to reconstruct a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

decoding the encoded data based on the determined context to reconstruct a bit value, wherein each bit value is a flag indicating whether a corresponding position in the transform unit contains a non-zero coefficient, and

updating the context based on that reconstructed bit value,

wherein the reconstructed bit values form the decoded significance map, and wherein the partition set assigns contexts to bit positions

characterised in that

the transform unit has a size of 4x4 and that the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
4,  5,  2,  3,
6,  6,  7,  7,
8,  8,  7,
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

2. The method claimed in claim 1, wherein determining includes selecting the partition set from among a plurality of partition sets based upon text type, and wherein the text type is luma.

3. A method of decoding a bitstream of encoded data to reconstruct a significance map for a

transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

decoding the encoded data based on the determined context to reconstruct a bit value, wherein each bit value is a flag indicating whether a corresponding position in the transform unit contains a non-zero coefficient, and

updating the context based on that reconstructed bit value, wherein the reconstructed bit values form the decoded significance map, and wherein the partition set assigns contexts to bit positions

characterised in that

the transform unit has a size of 4x4 and that the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
1,  1,  2,  3,
4,  4,  5,  5,
3,  3,  5
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

4. A decoder for decoding a bitstream of encoded data to reconstruct a significance map for a transform unit, the decoder comprising:

a processor;

a memory; and

a decoding application stored in memory and containing instructions for configuring perform the method claimed in any one of claims 1 to 3.

5. A method for encoding a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

encoding a bit value at that bit position based on the determined context to generate encoded data, wherein each bit value is a flag indicating whether a corresponding position in the transform unit contains a non-zero coefficient, and

updating the context based on that bit value,

wherein the encoded data forms an encoded significance map, and wherein the partition set assigns contexts to bit positions

characterised in that

the transform unit has a size of 4x4 and that the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
4,  5,  2,  3,
6,  6,  7,  7,
8,  8,  7,
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

6. A method for encoding a significance map for a transform unit, the method comprising:

for each bit position in the significance map,

determining a context for that bit position based upon a partition set,

encoding a bit value at that bit position based on the determined context to generate encoded data, wherein each bit value is a flag indicating whether a

corresponding positing in the transform unit contains a non-zero coefficient, and

updating the context based on that bit value,

wherein the encoded data forms an encoded significance map, and wherein the partition set assigns contexts to bit positions

characterised in that

the transform unit has a size of 4x4 and that the partition set assigns contexts to bit positions in accordance with a block-based mapping given by:

```
0,  1,  2,  3,
1,  1,  2,  3,
4,  4,  5,  5,
3,  3,  5
```

and wherein the above integers represent the contexts assigned to the bit positions of a 4x4 block significance map.

7. An encoder for encoding a significance map for a transform unit, the encoder comprising:

a processor;

a memory storing the significance map; and

an encoding application stored in memory and containing instructions for configuring the processor to perform the method claimed in claim 5 or claim 6.

8. A non-transitory processor-readable medium storing processor-executable instructions which, when executed, configures one or more processors to perform the method claimed in any one of claims 1 to 3, 5, and 6.