4

1. I would encode each of the four possible bit patterns to ±8v, ±6v, ±4v, and ±2v respectively. The encoding would follow the AMI principle of transitioning between different bit patterns. The following table details this further:

|  |  |
| --- | --- |
| 00 | ±2V |
| 01 | ±4V |
| 10 | ±6V |
| 11 | ±8V |

1. This scheme does guarantee transitions
2. It is DC balanced since every representation has both a positive and a negative voltage level it corresponds to. Since it follows the AMI idea of transitioning, this will average out to 0V in the end.
3. The scheme already is DC balanced and should give the greatest throughput given the constraints.
4. This transmits 2 bits per microsecond.
5. This encoding scheme guarantees transitions since it is based of AMI.
6. This coding scheme can tolerate noise that is at most half the difference between each voltage level. Since each level is 2v apart,

3.

a) (insert graph here)

b) Every instance, the receiver samples early by an additional 0.05 microseconds on top of the lag that it had already accumulated. By the 10th sample, the receiver is off by

c) (insert graph here)

d) The voltage is already high for the first bit that lasts from 0-1.05 microseconds. Therefore, a sharp noise at 0.4 microseconds would not register as a transition and would also not affect samplings beyond that point.

e) The voltage is originally low at 2.4 microseconds. Therefore, a sharp noise at that point would register as a transition. This would greatly decrease the next few frame wait lengths before the algorithm would begin lengthening it back.