```
int start()
{
  int limit = Bars - IndicatorCounted();
  int i = 0;

  //転換線の値の計算
  for(i = limit - 1; i >= 0; i--)
      {
        Change[i] = (High[iHighest(NULL,0,MODE_HIGH,Change_Period,i)] +
                     Low[iLowest(NULL,0,MODE_LOW,Change_Period,i)]) / 2;
      }

  //基準線の値の計算
  for(i = limit - 1; i >= 0; i--)
      {
        Standard[i] = (High[iHighest(NULL,0,MODE_HIGH,Standard_Period,i)] +
                       Low[iLowest(NULL,0,MODE_LOW,Standard_Period,i)]) / 2;
      }

  //雲の値の計算
  for(i = limit - 1; i >= 0; i--)
      {
        Span_A_L[i] = (Change[i] + Standard[i]) / 2;
        Span_B_L[i] = (High[iHighest(NULL,0,MODE_HIGH,Span_B_Period,i)] +
                       Low[iLowest(NULL,0,MODE_LOW,Span_B_Period,i)]) / 2;
        Span_A_H[i] = (Change[i] + Standard[i]) / 2;
        Span_B_H[i] = (High[iHighest(NULL,0,MODE_HIGH,Span_B_Period,i)] +
                       Low[iLowest(NULL,0,MODE_LOW,Span_B_Period,i)]) / 2;
      }

  //遅行線の値の計算
  for(i = limit - 1; i >= 0; i--)
      {
        Delay[i] = Close[i];
      }

  return(0);
}
```