Report for Base64 decoder

Base64 is a group of binary-to-text encoding scheme. By applying this scheme one could represent binary data in an ASCII string. It is usually used in cases when binary-format data needed to be transferred over media that mostly deals with text data, e.g. emails attachments etc.

In this project, we implement a Base64 decoder, i.e. it takes a byte sequence of 8-bit-padded ASCII characters and outputs the "original" texts. The base64_decoder.c is the "main" file containing the program entry and "utilities.h" file contains a number of helper functions used. Several functions for input and output purpose is also used from the libio library set.

In our implementation, the input data are (almost) evenly partitioned into two parts, and we create two independent decode runners for them respectively. Each of them will be responsible for reading encoded data and store the decoded data on their data section. We also set up a watcher responsible for handling output work. As soon as the decoded data has been allocated, these watchers will notice the data update and proceed with work.

The data for testing purpose is encoded from some paragraph of words. And the code has been tested on data sets of different size, from hundreds of characters to ten thousands of characters.

Our overall design contains much space for improvement.
First most apparently the data could endure more partitions instead of just two, especially for large data. The degree of parallelism is the key to the program performance. Also, in this version of implementation the data input part is done in traditional C way. However it could be improved by setting independent input runners reading in encoded data, and different (layers of) watchers will be responsible for further operations such as decoding and outputting etc.