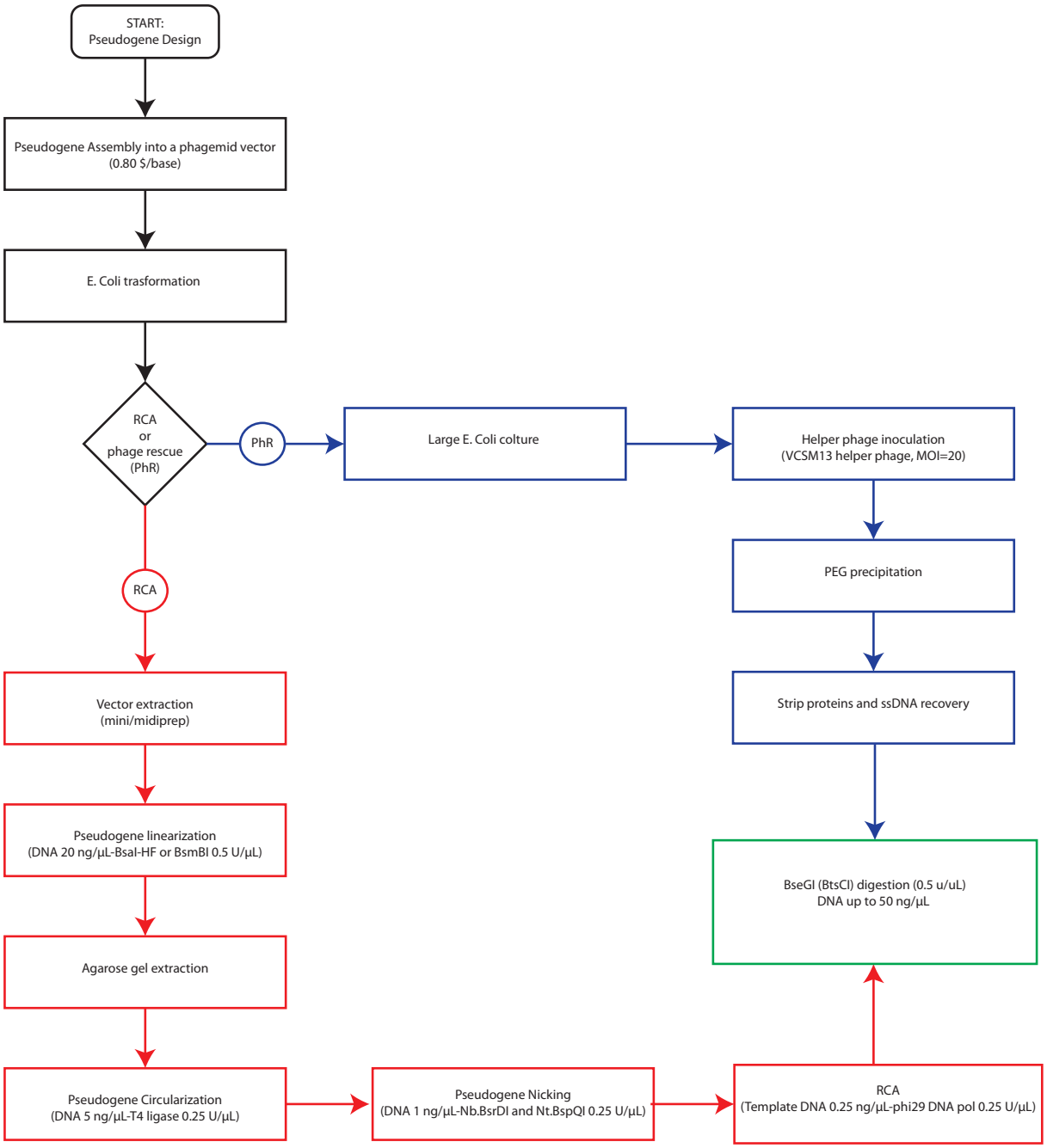
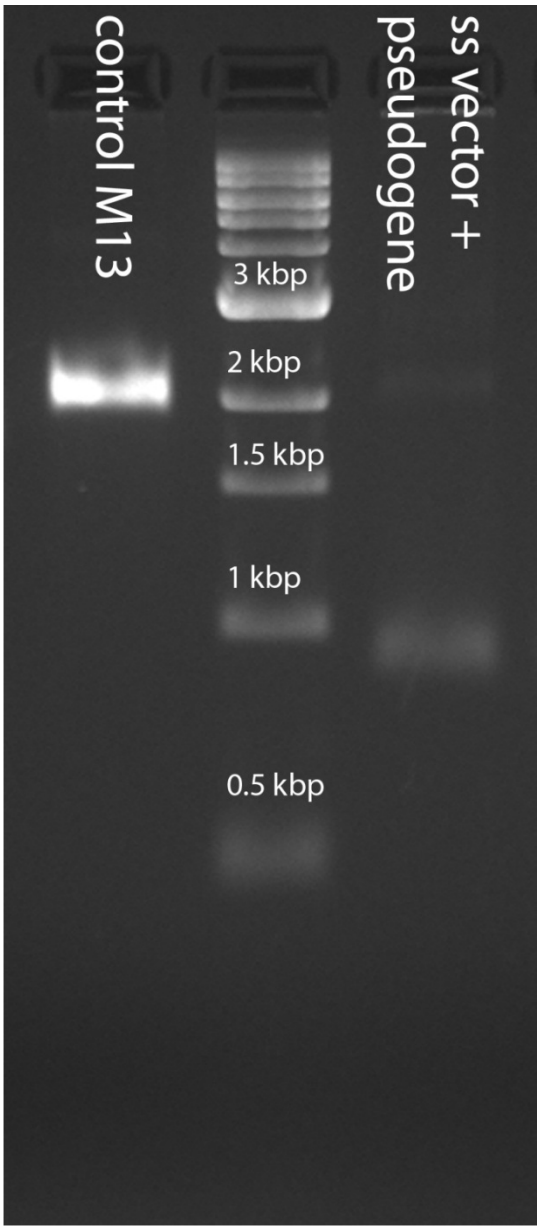
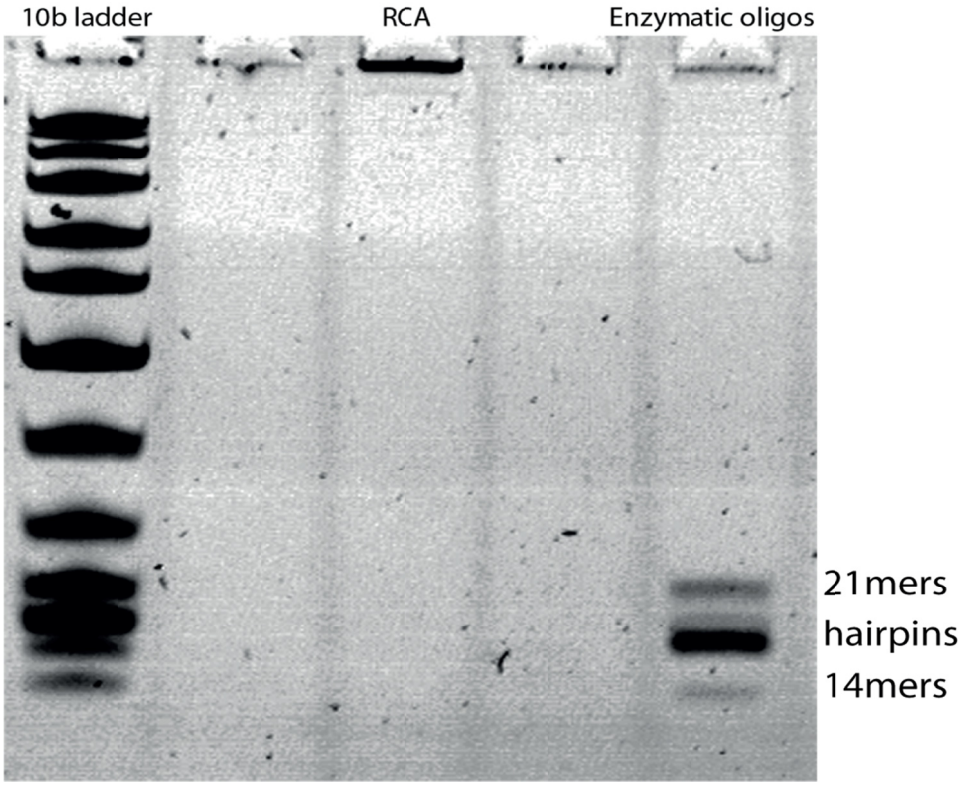


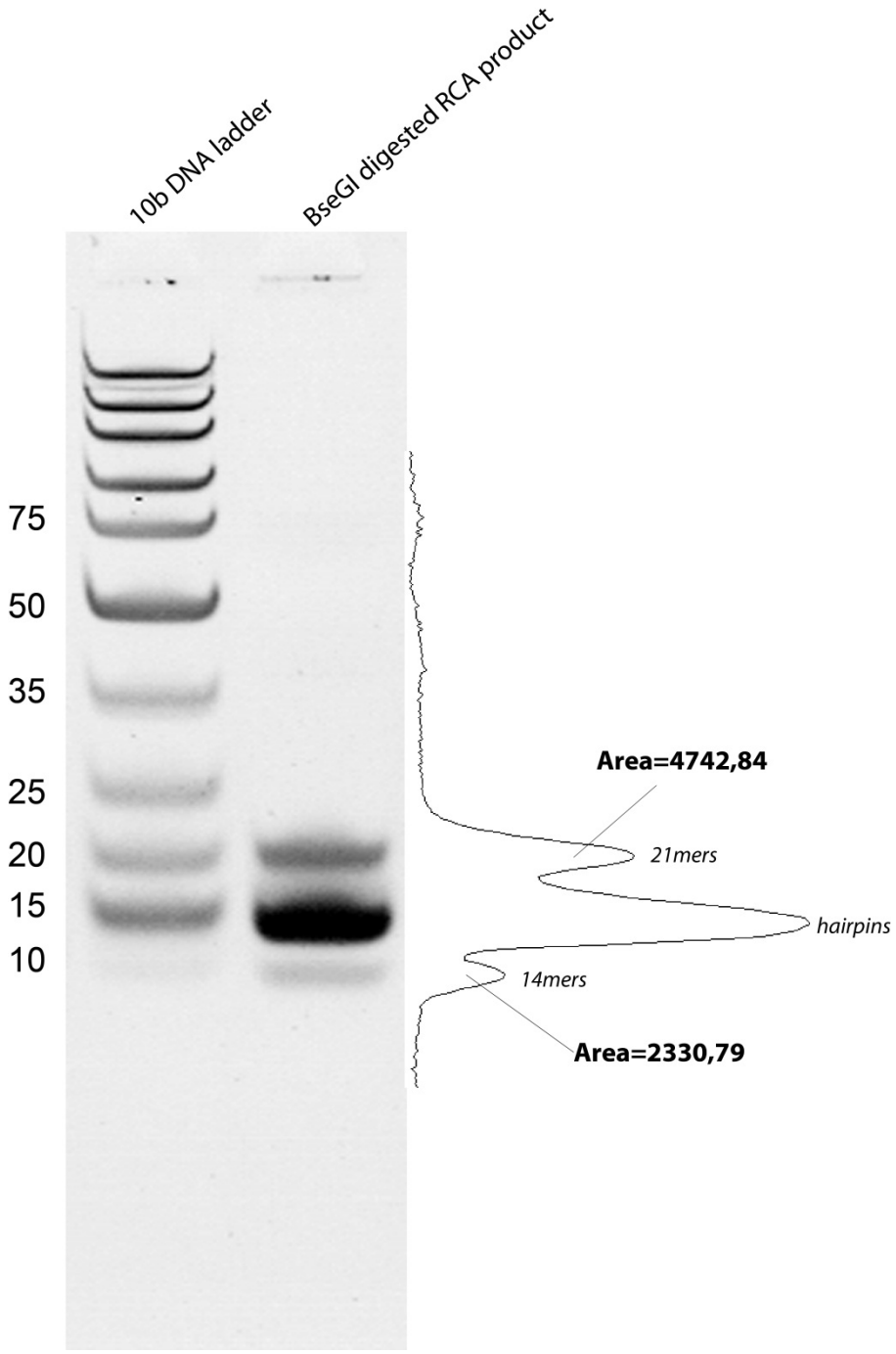
Supplementary Figure 1: Workflow for the technique.



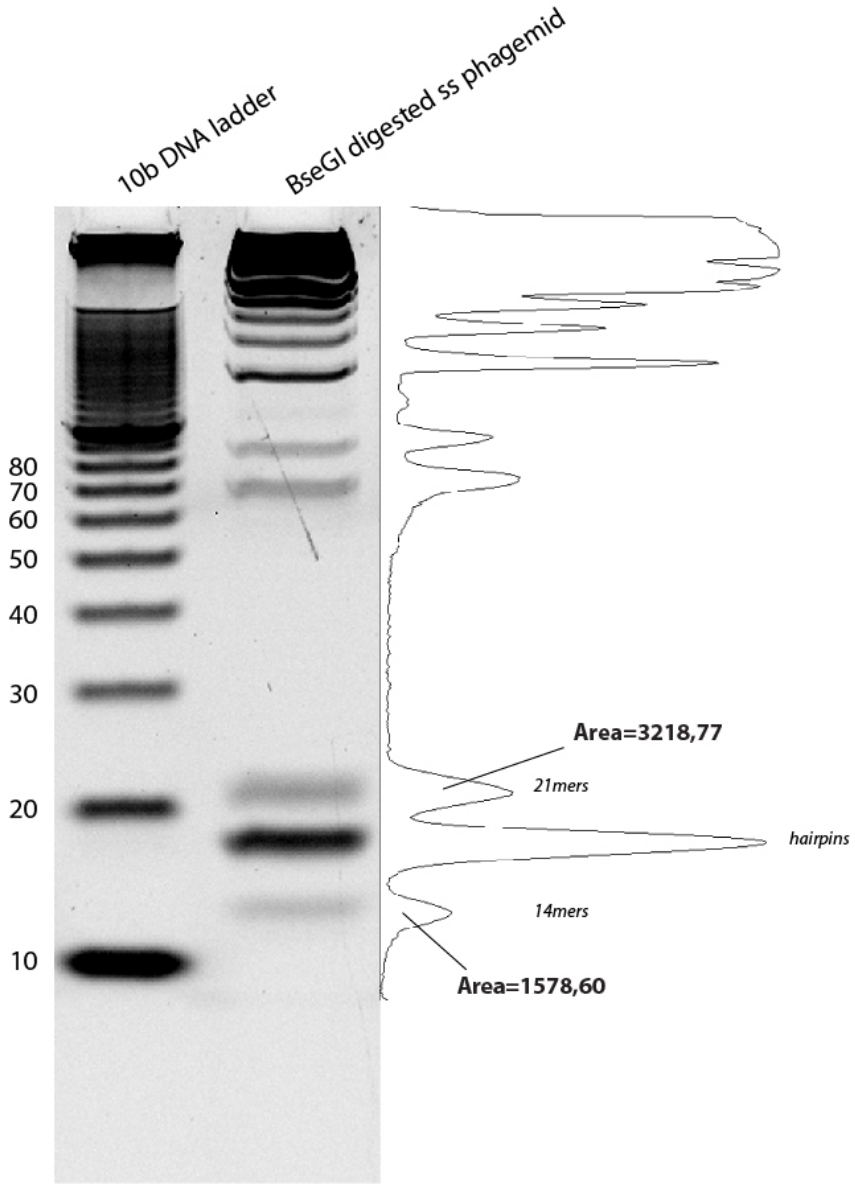
Supplementary Figure 2: Agarose gel electrophoresis of single stranded pBluescript SK II(-) containing CP recovered from phage particles(third lane). The recovered ss phagemid is usually 20 times more than the VCSM13 helper phage DNA.



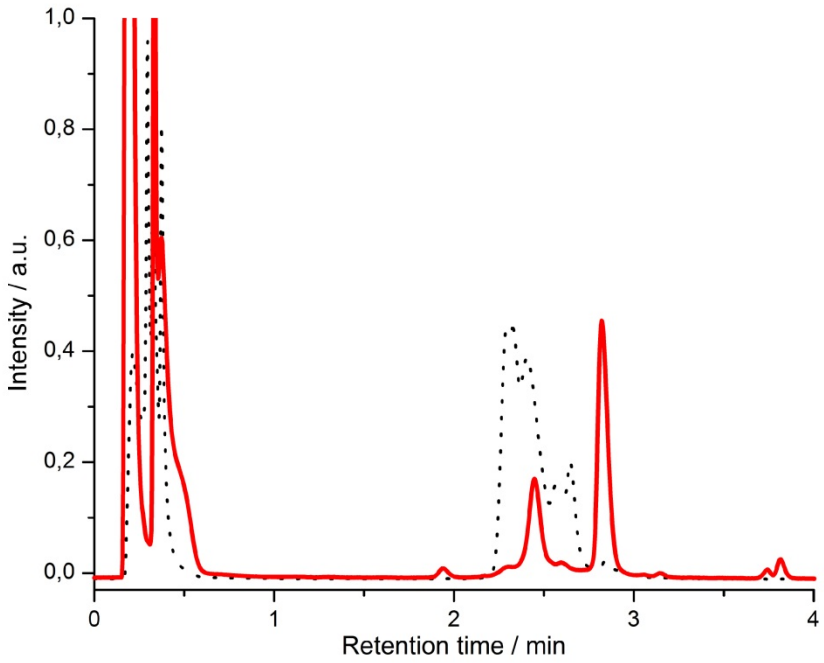
Supplementary Figure 3: Full picture of gel shown in fig. 2.



Supplementary Figure 4: Intensity evaluation of CP oligos from RCA product BseGI digestion: Polyacrylammide gel of the BseGI digested RCA product has been analyzed by using ImageJ. The areas corresponding to the intensities of the bands show that the 21mers and the 14mers are in a ratio of 1:2. This value confirms the stoichiometry of the oligonucleotides (four 21mers for a total of 84 nucleobases, three 14mers for a total of 42 nucleobases).

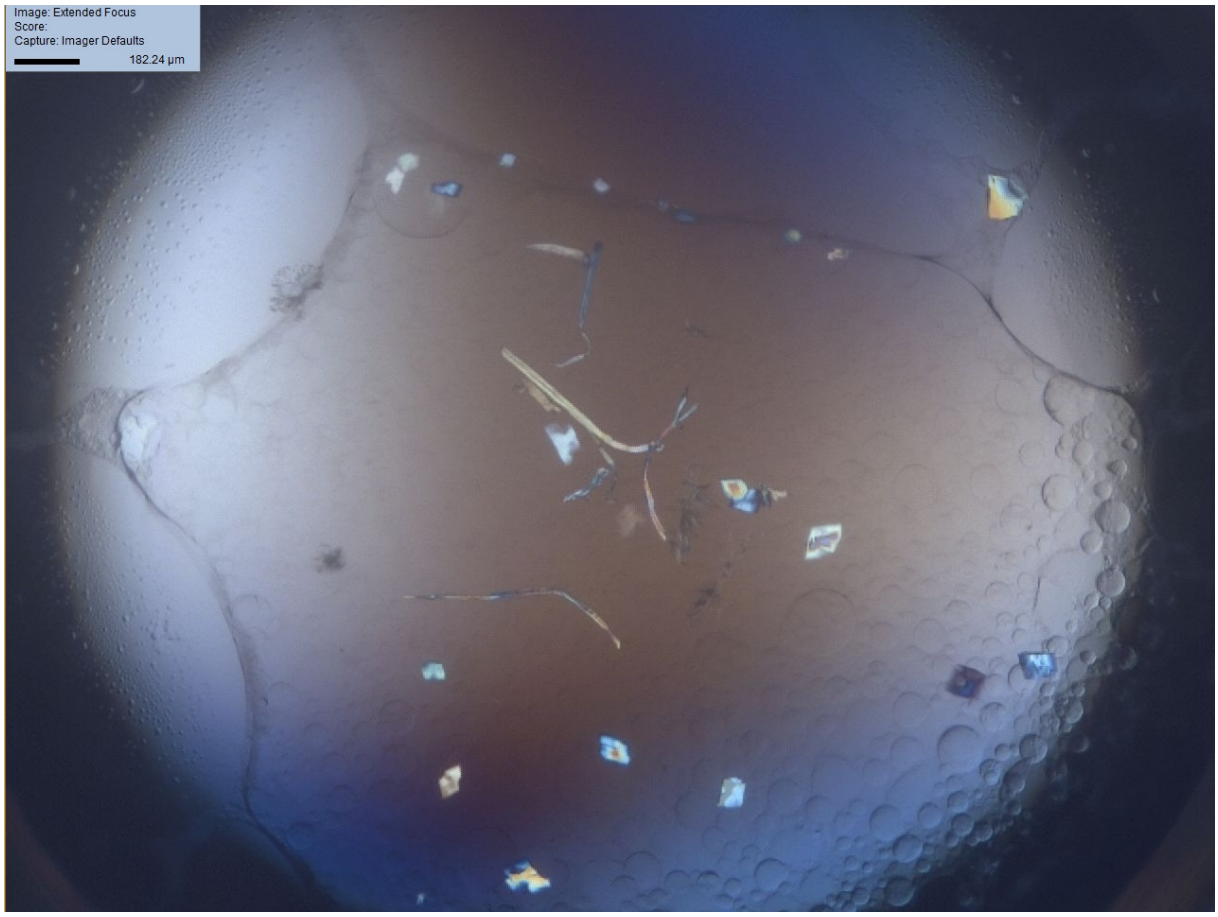


Supplementary Figure 5: Intensity evaluation of CP oligos from ss phagemid BseGI digestion: Polyacrylammide gel of the BseGI digested ss phagemid has been analyzed by using ImageJ. The areas corresponding to the intensities of the bands show that the 21mers and the 14mers are in a ratio of 1:2. This value confirms the stoichiometry of the oligonucleotides (four 21mers for a total of 84 nucleobases, three 14mers for a total of 42 nucleobases).

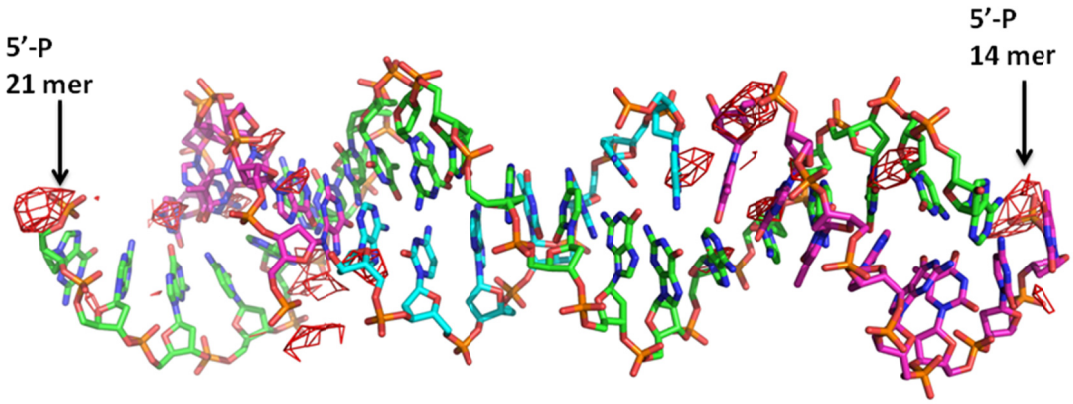


Supplementary Figure 6: Full picture of HPLC shown in Figure 3a. Hairpins have a low retention into the column and flow through it. The synthetic oligonucleotide mixture shows impurities which run at low retention time.

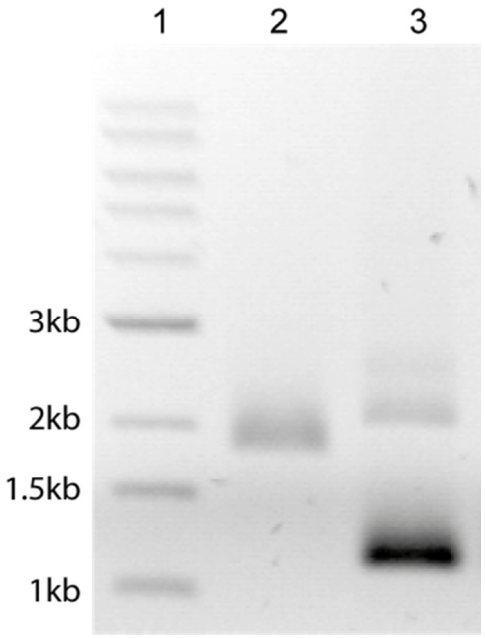
Nature Methods: doi:10.1038/nmeth.2503



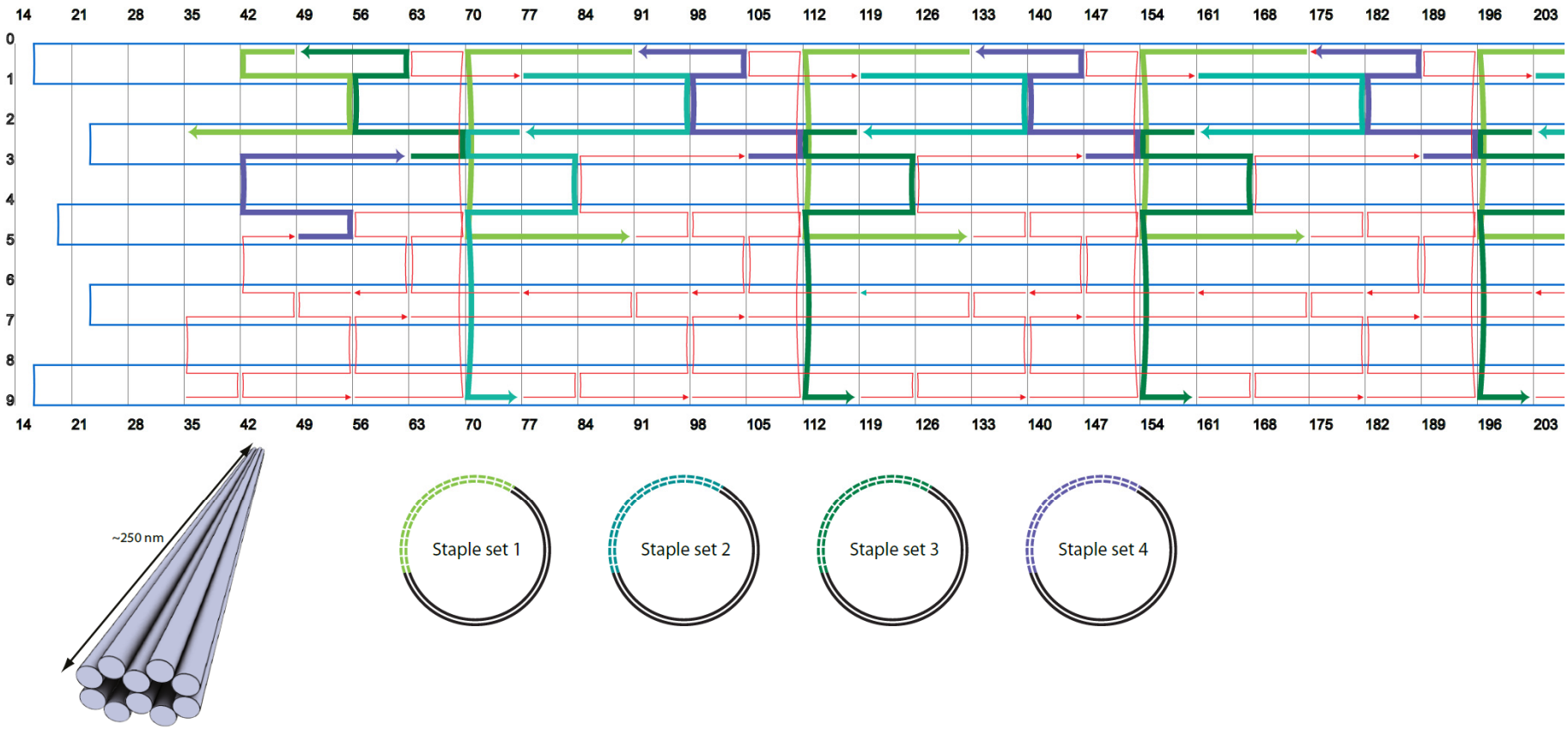
Supplementary Figure 7: Optical image of a drop with tensegrity triangles. The rhombohedral shape and the scale are visible.



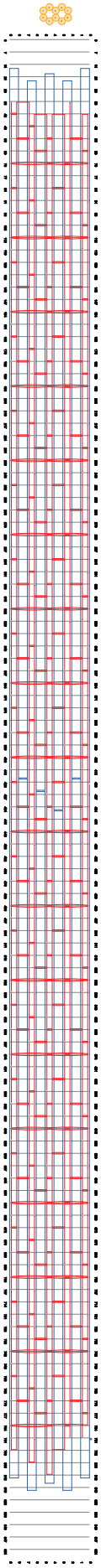
Supplementary Figure 8: Initial difference map contoured at 3 sigma covering the final model. The DNA crystal is built up by three oligonucleotides in a 3:3:1 ratio of 14:21:21-mers colored in magenta:green:cyan respectively (Figure 3B). This picture shows the asymmetric unit selected as in 3GBI (Seeman et al). Initial refinement of the Seeman 3GBI structure resulted in this initial difference density map here contoured at 3 sigma covering the entire model. This initial difference map highlighted the non-modeled 5-prime phosphate groups present in the MOSIC generated DNA oligonucleotides and absent from the 3GBI oligos and model. The 5-prime phosphates were subsequently modeled and the picture shows our final model (4B8D).



Supplementary Figure 9: Agarose gel electrophoresis of single stranded pBluescript SK II(-) containing staple set 1 recovered from phage particles (lane 3) compared to 1 kb ladder (lane 1) and single-stranded M13 (lane 2). The recovered ssDNA phagemid is usually 20 times more than the VCSM13 helper phage DNA.



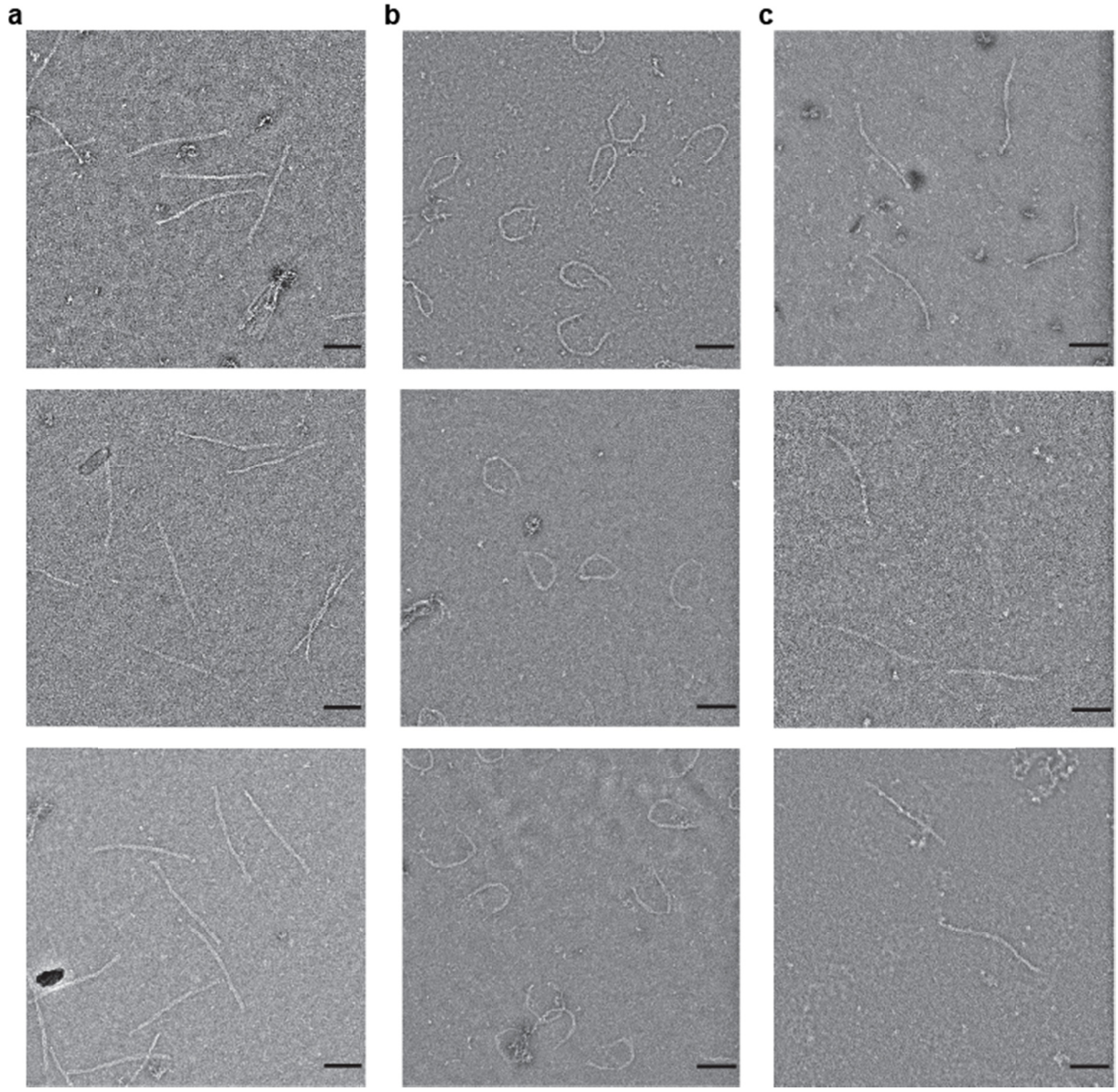
Supplementary Figure 10: Schematic representation and partial caDNAno design of 10 HB with designation of the 4 staple set pseudogenes. Design diagram truncated to the right. Full design in S13.



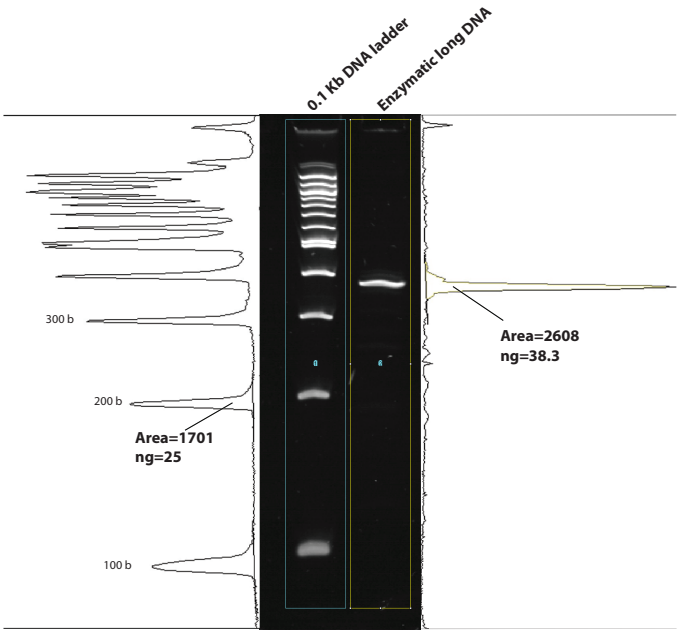
Start (helix{base})	End (helix{base})	Sequence	NTs	Staple set
0[48]	2[35]	TTTTATAGTAGCAATACTTCTTTATCATCATATTCCTGATTA	42	1
0[90]	5[90]	GGATTTTAGACAGGAACGGTATAATTCTGTCCAGACGACGAC	42	1
0[132]	5[132]	CGTTAGAATCAGAGCGGGAGCATGCAGAACGCGCCTGTTTAT	42	1
0[174]	5[174]	GTACTATGGTTGCTTTGACGAGCACGAAAAATAATATCCCATC	42	1
0[216]	5[216]	AACCACCACACCCGCCGCGCTACCAATCAATAATCGGCTGTC	42	1
0[258]	5[258]	GGGCGCTAGGGCGCTGGCAAGGGGTATTAAACCAAGTACCGC	42	1
0[300]	5[300]	GGCGAACGTGGCGAGAAAGGAAGGGTTTTTATTTTCATCCTAGG	42	1
0[342]	5[342]	CCCTAAAGGGAGCCCCCGATTTCAAGCAAATCAGATATAGAAG	42	1
0[384]	5[384]	AAGTTTTTGGGGTCGAGGTGCGCGAGGCGTTTTAGCGAACC	42	1
0[426]	5[426]	CTATCAGGGCGATGGCCCACTGAAGCCTTAAATCAAGATTCC	42	1
0[468]	5[468]	ATTAAAGAACGTGGACTCCAAGTCAATAACCTCTTTAGCTAT	42	1
0[510]	5[510]	GATAGGGTTGAGTGTTGTTCCTGAAAAGGTGGCATCAATTCT	42	1
0[552]	5[552]	CGAAATCGGCAAAATCCCTTAATCCAATAAATCATACAGGCA	42	1
0[594]	5[594]	TGGTTTGCCCCAGCAGGCGAAAAGCCAATAAAGCCTCAGAGCA	42	1
0[636]	5[636]	TCACCGCCTGGCCCTGAGAGAGAAAAACATTATCAACCTGTAA	42	1
0[678]	5[678]	TTTTTCTTTTCACCAGTGAGACATTTTCAACGCAAGGATAAAA	42	1
0[720]	5[720]	ACGCGCGGGGAGAGGCGGTTTTTTAAATGCAATGCCTGAGTA	42	1
0[762]	5[762]	GTCGGCAAACCTGTCGTCCCAAGGGTAGAAAGCCGGAGAC	42	1
1[77]	2[77]	TGCCTGAGTAGAGAGACTCAATTATCATTTGCGGAACAAAG	42	2
2[76]	9[76]	AAACCACGGAAGGGTTAGAACCTCAACAGTAGGGCTCTGAAA	42	2
1[119]	2[119]	ATCCAGAACAATATTACCGCCTCCTTTGCCCGAACGTTATTA	42	2
1[161]	2[161]	GCTCATGGAATACCTACATTGAAGTATTAGACTTTACAAAC	42	2
1[203]	2[203]	TGGATTATTTACATTGGCAGAAATAGATTAGAGCCGTCAATA	42	2
1[245]	2[245]	AATAAAGGGACATTCTGGCCTGAGGAAGGTTATCTAAAATA	42	2
1[287]	2[287]	ACCTGAAAGCGTAAGAATACGATCAATATCTGGTCAGTTGGC	42	2
1[329]	2[329]	ATGGCTATTAGTCTTTAATGCTCTAAAGCATCACCTTGCTGA	42	2
1[371]	2[371]	ATCGCCATTAAAAATACCGAACCTGCAACAGTGCCACGCTGA	42	2
1[413]	2[413]	AACAGTCAGCGGGGTCATTGCCTGGAGGTGTCCAGCAAGGTG	42	2
1[455]	2[455]	GCCGGGCGCGGTTGCGGTATGGCAGCACCGTCGGTGGTGCCA	42	2
1[497]	2[497]	CGGCTGCTAATGGGTAAAGGTGTGGTGCTGGTCTGCTCAGCA	42	2
1[539]	2[539]	CCCTTACACTGGTGTGTTCAGACATCCTCATAACGGAACGTG	42	2
1[581]	2[581]	TGCCGGGTTACCTGCAGCCAGGTCCGTTTTTTCGTCTCGTCG	42	2
1[623]	2[623]	CAGACGATCCAGCGCAGTGTCGGCTAAAGTTAAACGATCCTG	42	2
1[665]	2[665]	TGGTGCTGCGGCCAGAATGCGAATCCCGTAAAAAAGCCGCA	42	2
1[707]	2[707]	ACCGGGGGTTTCTGCCAGCACTTGCCGCCAGCAGTTGGGCGG	42	2
1[749]	2[749]	GTGAGCCTCCTCACAGTTGAGAGAGACGCAGAAACAGCGGAT	42	2
3[63]	0[49]	GAATAATCAGAAGGAGCCGAATTCATTAATCCTGAGAAGTCT	42	3
2[118]	9[118]	ATTTTAAAACAGAAATAAAGAATCATATGCGTTATAAGGATT	42	3
2[160]	9[160]	AATTCCAAGATGAATATACAGATAAACACCGGAATAAGTGCC	42	3
2[202]	9[202]	GATAATAAATAACGGATTCGCTTTGAAATACCGACGTGTATC	42	3
2[244]	9[244]	TCTTTAGATCCCGCAGAGCCATATATTTTAGTTAAACCGCC	42	3
2[286]	9[286]	AAATCAAGAAGATGATGAAAAAATCCAATCGCAACATTTTC	42	3
2[328]	9[328]	ACCTCAAATTTAACAATTTCACGGCTTAGGTTGGGAACACTG	42	3
2[370]	9[370]	GAGCCAGTACATAAATCAATATTTATCAAAATCATCATTCCA	42	3
2[412]	9[412]	AGGCGGTAATCCTCCTATTAGCGATAGCTTAGATTTTTGTC	42	3
2[454]	9[454]	TCCCACGATGGGCGCATCGTAGCGGATTGACCGTAGGGATTT	42	3
2[496]	9[496]	GCAACCGACGACAGTATCGGCCTGAGCGAGTAACAAAATAGAA	42	3
2[538]	9[538]	CCGGACTTCCGGCAACCGCTTCCGCCTCGGCCTTCCTCAGCTT	42	3
2[580]	9[580]	CTGGCAGTCGCCATTCAGGCTAGCTCATTTTTAATTAATTG	42	3
2[622]	9[622]	ATTGCCGCGGGCCTCTTCGCTGTTAATATTTTGTTTTAAACA	42	3
2[664]	9[664]	CAGGCCGGCTCGAGGCCATTAAAAACAGGAAGATACCATCG	42	3
2[706]	9[706]	TTGTGTATCACACCTTGTAAATCTCAATCATATCTGCAGGG	42	3
2[748]	9[748]	CAAACTTGTGGAGCCGCCACGGAGCAAACAAGAGAGCAGCGA	42	3
2[790]	9[790]	AGGGATAGGCGAAACGTACAGGAGAGATCTACAAAGAGGCTT	42	3
3[105]	0[91]	CACGTAAAAGTTTGAGTAACAACTATCGGAGGCCGATTAAAG	42	4
5[49]	3[62]	AAGTACCCCATATTTAACAACGATTGTTTGGATTATACTCT	42	4
3[147]	0[133]	TAACGTCCAACTCGTATTAAAAGCCATTTAACGTGCTTTCCT	42	4
3[189]	0[175]	GAGAAACCATTTGAGGATTTATTGACGCCCGCTACAGGGCGC	42	4
3[231]	0[217]	TTACAAAGAGCACTAACAACTTTCACCAGTCACGCTGCGCGT	42	4
3[273]	0[259]	GAGCAAACAGTTGAAAGGAATAACAGAGAAGCGAAAGGAGC	42	4
3[315]	0[301]	TAATTACATATCAAACCCTCATGGCACATGACGGGGAAAGCC	42	4
3[357]	0[343]	GAAACAGCAGCAAATGAAAAACGCAACTGCACTAAATCGGAA	42	4
3[399]	0[385]	TTCTGTACAGTATTAACACCCCGAACACCATCACCCAAATC	42	4
3[441]	0[427]	GGTGTAGCAACCAGCTTACGGAGGCGCTGGGCGAAAAACCGT	42	4
3[483]	0[469]	GGGGACGCAAGAATGCCAACGAGCCGGGAACAAGAGTCCACT	42	4
3[525]	0[511]	CCAGCTTTGTAGAACGTCAGCTTCTTTGAAAGAATAGCCCGA	42	4
3[567]	0[553]	GCGCCATCCTCCGGCCAGAGCCAAATCGTTTGATGGTGGTTC	42	4
3[609]	0[595]	ATCGGTGTTCCGGCAAACGGCGGTGCCCAAGCGGTCCACGC	42	4
3[651]	0[637]	GGGATGTCCTTTAGTGATCAAACTGCGCCAGCTGATTGCCCT	42	4
3[693]	0[679]	TTCCCAGCATCGACATAAAAAGCGGGCCGGGCGCCAGGGTGG	42	4
3[735]	0[721]	TTCAGAGAAATTTCTGCTCATGCCTGCCTAATGAATCGGCCA	42	4
3[777]	0[763]	AACAATCGCTCTCACGGAAAAGATCCCCCTGCCCGCTTTCCA	42	4
5[91]	3[104]	AATAAACGTATAAAGCCAACGCTACCATATCAAAATTATTTG	42	
6[55]	9[55]	GATAGCCAAAAGAAGTGTACTGGTAATAGCCCCTCGCCTATT	42	
5[133]	3[146]	CAACAATAAAGCCTGTTTAGTAATTGCGTAGATTTTCAGGTT	42	
5[175]	3[188]	CTAATTTGGCGTTAAATAAGATAACAGTACCTTTTACATCGG	42	
5[217]	3[230]	TTTCCTTCCTAAATTTAATGGCTGATTGCTTTGAATACCAAG	42	
5[259]	3[272]	ACTCATCGAAAACTTTTCAAAATTATTCATTTCAATTCTTGC	42	
5[301]	3[314]	AATCATTGTAAATGCTGATGCCAAACATCAAGAAAACAAAAT	42	
5[343]	3[356]	GCTTATCACCTTTTTAACCTCTTTGAATTACCTTTTTTAATG	42	
5[385]	3[398]	TCCCGACGAGTCAATAGTGAATATGTGAGTGAATAACCTTGC	42	
5[427]	3[440]	ATTAGATTCCTTGAAAACATAATAATTTTCCCTTAGACGTT	42	
5[469]	3[482]	ATTTTCACGTGGGAACAAACGACCGTGCATCTGCCAGTTGA	42	
5[511]	3[524]	ACTAATAATCAACATTAAATGCTCAGGAAGATCGCACTCCAG	42	
5[553]	3[566]	AGGCAAAATCAAAAATAATTCTGGTGCCGGAAACCAGGCAAA	42	
5[595]	3[608]	TAAAGCTATTTTGTTAAATCGCGCAACTGTTGGAAGGGCG	42	
5[637]	3[650]	TACTTTTTTTAAATTGTAAACATTACGCCAGCTGGCGAAAGG	42	
5[679]	3[692]	ATTTTTAATCAGAAAAGCCCCAAGTTGGGTAACGCCAGGGTT	42	
5[721]	3[734]	ATCGTAATCGTAAAACTAGCAACGACGGCCAGTGCCAAGCT	42	
5[763]	3[776]	AGTCAAATGCCTGAGAGTCTGGGAACGGATAACCTCACCGGA	42	
6[76]	1[76]	CCTTTTTAAGAAAAGGTAAAGCGCCAGAGTAATAACATCACT	42	
6[97]	9[97]	CAATAGCTAGCAAACCGTTCCAGTAAGCGTATTAAGAGGCTG	42	
6[118]	1[118]	GAGCAAGAAACAATTCAGCTATAAACAGGCCTTGCTGGTAAT	42	
6[139]	9[139]	TTGAGTTAAAGAAATCCTCATTAAAGCCGCGGGGTTTTGCTC	42	
6[160]	1[160]	CAGAGAGATAACCCTCCTGAAGCACGTAGCAACAGGAAAAAC	42	
6[181]	9[181]	AGGGTAAAATCAATGGCAGGTCAGACGAGGGTTGATATAAGT	42	
6[202]	1[202]	GAACACCCTGAACATGTACGAATAATGCCTCAATCCTCTAAA	42	
6[223]	9[223]	CATTAGAAAGGGCGGAACCACCACCAGATCAGGAGGTTTAGT	42	
6[244]	1[244]	TAACATAAAAACAGCAAGAACTGTAGCGGTCACACGACCAGT	42	
6[265]	9[265]	TAGCAGCGACGGAAGAACCGCCACCCTCGAACCGCCACCCTC	42	
6[286]	1[286]	GTTTAACGTCAAAAGCAAGCAGGGAAGATAGAACCCTTCTG	42	
6[307]	9[307]	CCAAATACTTGAGCGAGCCACCACCGGAGCAAGCCCAATAGG	42	
6[328]	1[328]	AGCCATATTATTTACCAATAGTAGAGCTGACAATATTTTGA	42	
6[349]	9[349]	TTTCCCACACCATTTAGCGTTGCCATTCACCAGTACAAAC	42	
6[370]	1[370]	GAGCGTCTTTCCAGCTAAGAACCGTAAAGATAGCCCTAAAAC	42	
6[391]	9[391]	CTGAATCCATCGATGTAGCGCGTTTCCCCTCATAGTTAGC	42	
6[412]	1[412]	GCACCCAGCTACAAAGGTTTTACGTGAACCAGCAGAAGATAA	42	
6[433]	9[433]	TTAGTTTCAGAACGTAAGCCTGCCCTGCAGACGTTACGTAAA	42	
7[63]	6[77]	GATTAAGACTCCTTATTACGCAGTATGTTATCTTACCGAAGC	42	
6[454]	1[454]	CAATTCTGCGAACGGCAAATGCGTCAAATTCGCACTCAATCC	42	
6[475]	9[475]	TTCCATACTTTAATTCATTACCCAAATCCAACTTTCAACAGT	42	
6[496]	1[496]	GTACGGTGTCTGCACGCAGCAGTTTGCTCACTGTTGCCCTG	42	
6[517]	9[517]	TGTTTTACTCATTAGGCTGACCTTCATCACTAAAGGAATTGC	42	
6[538]	1[538]	ATATAATGCTGTAGCATTAACTAAATCACTCGTCATAAACAT	42	
6[559]	9[559]	GGCTTAGAATAAAAAAAGAGGACAGATGTCCAAAAAAAGGC	42	
6[580]	1[580]	TAAGAGCTCATTTTCAAAATTAATCCTGTTAACGGCATCAGA	42	
6[601]	9[601]	ACCTTTAATTCATCCGCAGACGGTCAATTTATCAGCTTGCTT	42	
6[622]	1[622]	AACAGGTCAGGATTTTGTACCGTTGCAGGGTGCCCCTGCAT	42	
6[643]	9[643]	ACCAGACAGATACACGAAATCCGCGACCACCGATAGTTGCGC	42	
6[664]	1[664]	TTAATTCGAGCTTCAAGCCTTCGGGCAAGCCTGTGCACTCTG	42	
6[685]	9[685]	AAAGACTAACACTAAAAGTACAACGGAGATAACCGATATATT	42	
6[706]	1[706]	AAAAGATTAAGAGGCATATATGCGTATTGTTTTCACGGTCAT	42	
6[727]	9[727]	GAAGCAAAAAATAGACACTCATCTTTGAGGCCGCTTTTGCGG	42	
6[748]	1[748]	TTACCCTGACTATTATTCAAAGCTGCATTGTTCTTCGCGTCC	42	
6[769]	9[769]	CATAAATGGTAATAAGGCACCAACCTAACATCGGAACGAGGG	42	
6[790]	1[790]	AGTTCAGAAAACGACAATATGGCGCTCAGGGTACCGAGCTCG	42	
9[35]	5[48]	AGTTAATAGTTTTAATAACGGAATACCCGACAAAGAATATA	42	
7[105]	6[119]	AATACATACATAAAGGTGGCAACATATAAAGCCAATAATAA	42	
7[147]	6[161]	ACACCACGGAATAAGTTTATTTTGTCACTTGAGGGCTAATAT	42	
7[189]	6[203]	TCATATGGTTTACCAGCGCCAAAGACAACGGGAGAATTAACT	42	
7[231]	6[245]	ACCGATTGAGGGAGGGAAGGTAAATATTCTTTACAGAGAGAA	42	
7[273]	6[287]	ATTAAAGGTGAATTATCACCGTCACCGAAGAAACGATTTTTT	42	
7[315]	6[329]	GAATTAGAGCCAGCAAAATCACCAGTAGGTTACAAAATAAAC	42	
7[357]	6[371]	GCAAGGCCGGAAACGTCACCAATGAAACTTACCAACGCTAAC	42	
7[399]	6[413]	CCGTAATCAGTAGCGACAGGAGAAACACGAAGTTGCTATTTT	42	
7[441]	6[455]	AATTGGGCTTGAGATGGTTTAATTTCAATAACAGTTGATTCC	42	
7[483]	6[497]	AATTACCTTATGCGATTTTAAGAACTGGAATAGTGCAACTAAA	42	
7[525]	6[539]	CAGGACGTTGGGAAGAAAATCTACGTTAGCTTAATTGCTGA	42	
7[567]	6[581]	ACGGAACAACATTATTACAGGTAGAAAGATTGCTCCTTTTGA	42	
7[609]	6[623]	ATTTAGGAATACCACATTCAACTAATGCCGGAAGCAAACTCC	42	
7[651]	6[665]	AAAAGGAATTACGAGGCATAGTAAGAGCTCAAATATCGGTT	42	
7[693]	6[707]	CCTCGTTTACCAGACGACGATAAAACCAGCGGATTGCATCA	42	
7[735]	6[749]	CTTTTGCAAAAGAAGTTTTGCCAGAGGGCAAAAATCAGGTCT	42	
7[777]	6[791]	GTTTAGACTGGATAGCGTCCAATACTGCCAAATGCTTTAAAC	42	
9[56]	6[56]	TCGGAACCTATTATTTAATTGAGAATCGGACAAAGCTAAGCA	42	
9[77]	7[62]	CATGAAAGTCATACATGGCTTTTGATGATACAGGACTGGCAT	42	
9[98]	6[98]	AGACTCCTCAAGAGACAAATTCTTACCAAACATGTGAAATAG	42	
9[119]	7[104]	AGGATTAAGAATCGAAAGCGCAGTCTCTGAATTTACGTAGAA	42	
9[140]	6[140]	AGTACCAGGCGGATCATAATTACTAGAAAGATAAGACAAGAA	42	
9[161]	7[146]	GTCGAGATTGGCCTTGATATTCACAAACAAATAAACGCAAAG	42	
9[182]	6[182]	ATAGCCCGGAATAGCGTGTGATAAATAAACGAGCAAAGTCAG	42	
9[203]	7[188]	ACCGTACGCCGCCGCCAGCATTGACAGGAGGTTGAAGAAAAT	42	
9[224]	6[224]	ACCGCCACCCTCAGATTCATCTCTGAATCATTCGGAAGCG	42	
9[245]	7[230]	ACCCTCAAGAGCCACCACCCTCAGAGCCGCCACCAACATTCA	42	
9[266]	6[266]	ACGAGCCACCACCCTGACAAAGAACGCGAGAGAACAATGAAAA	42	
9[287]	7[272]	AGGGATAACCGCCTCCCTCAGAGCCGCCACCCTCAATTATTC	42	
9[308]	6[308]	AACCCATGTACCGTTTATATAACTATATACGCGCTCCCAAT	42	
9[329]	7[314]	ACGTTTCGCTTTTCATAATCAAAATCACCGGAACCACATTTGG	42	
9[350]	6[350]	TACAACGCCTGTAGAGGTCTGAGAGACTCGGTATTAGCCTAA	42	
9[371]	7[356]	CAGACAGATCGGCATTTTCGGTCATAGCCCCCTTAACATTA	42	
9[392]	6[392]	GTAACGATCTAAAGTAAGACGCTGAGAATTGCGGGTTTTATC	42	
9[413]	7[398]	GTCTTTCACAATCAAGTTTGCCTTTAGCGTCAGACAGCAGCA	42	
9[434]	6[434]	TGAATTTTCTGTATATGGGATAGGTCAAACATTTCAGTAGAT	42	
9[455]	7[440]	TGCTAAAAACGTAACAAAGCTGCTCATTCAGTGAAAGTAGTA	42	
9[476]	6[476]	TTCAGCGGAGTGAGCCGTCGGATTCTCTTTGGGGAGTTTCA	42	
9[497]	7[482]	AGGAACAAAGAGTAATCTTGACAAGAACCGGATATCATTGTG	42	
9[518]	6[518]	GAATAATAATTTTTTGTAGCCAGCTTTCGTAGTAGCTCAACA	42	
9[539]	7[524]	GAAAATCAACGGTCACAGACCAGGCGCATAGGCTTACCAGT	42	
9[560]	6[560]	TCCAAAAGGAGCCTCCAATAGGAACGCCGAATTAGTCGCGAT	42	
9[581]	7[566]	TATCGGTCATAAGGGAACCGAACTGACCAACTTTGCGAACTA	42	
9[602]	6[602]	TCGAGGTGAATTTCAAAATTCGCATTAAAAATCGGAGAGAGT	42	
9[623]	7[608]	GCTTGATTGCTCCATGTACTTAGCCGCGACGAGTTGAG	42	
9[644]	6[644]	CGACAATGACAACATGTATAAGCAAATAGCGGGAGAAAGCGA	42	
9[665]	7[650]	CCCACGCATTTGTATCATCGCCTGATAAATTGTGTTAAGCGC	42	
9[686]	6[686]	CGGTCGCTGAGCGTTACCCCGGTTGATAGAACCCTAAGCCCG	42	
9[707]	7[692]	AGTTAAACCCCCAGCGATTATACCAAGCGCGAAACTCATAAC	42	
9[728]	6[728]	GATCGTCACCCTCAATCGATGAACGGTAGGTAAAGATAGTCA	42	
9[749]	7[734]	AAGACAGAACGAAAGAGGCAAAAGAATACACTAAACGAGAGG	42	
9[770]	6[770]	TAGCAACGGCTACAGGCTATCAGCTCATTCACATGAATGAC	42	
9[791]	7[776]	TGAGGACACGGGTAAAATACGTAATGCCACTACGAGTAAAAT	42	

Supplementary Figure 11: Full design of 10HB used in demonstration of staple production. Distribution of staple-sets is explained in S12.

(Not optimized for printing, zoom in pdf viewer)



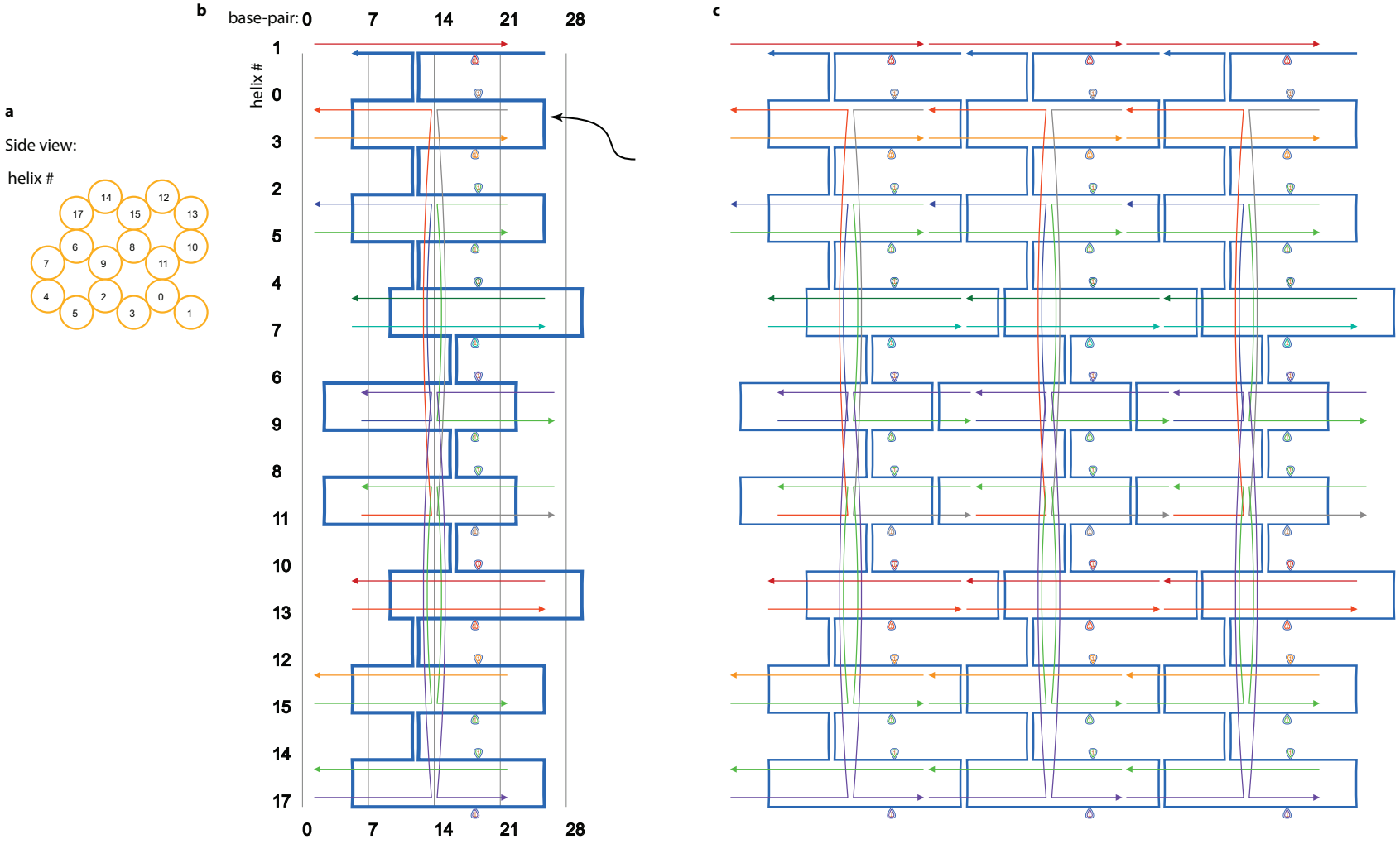
Supplementary Figure 12: TEM images of 10 helix bundle **a:** after addition of 180 synthetic staple strands (full set). **b:** lacking 72 staple strands **c:** after addition of lacking 72 staple strands produced by MOSIC, scale bars 100 nm.



Long ODN MOSIC reactions	Consumables	units/µg of DNA	$/unit	$/µg of DNA
Plasmid extraction	PLASMID PURIFICATION MINI KIT (Omega Bio-Tek)	1 test kit rection	0.74	0.039
Linearization	GEL EXTRACTION KIT (Omega Bio-Tek)	1 test kit rection	1.9	0.100
Linearization	BsaI-HF (NEB)	1.41 units	0.05	0.071
Ligation	T4 ligase (Fermentas)	0.26 units	0.14	0.036
Nicking	Nb.BsrDI/Nt.BspQI (NEB)	2.6 units	0.05	0.260
Rolling Circle Amplification	phi29 DNA polymerase (Fermentas)	10.5 units	0.15	1.575
Rolling Circle Amplification	T4 gene 32 protein (NEB)	2.1 µg	0.568	1.193
Rolling Circle Amplification	dNTPs mix (Fermentas)	21 nmol	0.0076	0.160
Final digestion	BseGI (Fermentas)	42 units	0.035	0.147
Plastic[1]	tubes, pipettes tips (VWR), C18 cartridges (Waters)	/	/	0.462
Manual labor[2]	2,5 hrs of hands-on work	/	40 $/hr	5.250
total cost ($/µg of DNA)				**9.923**
Synthetic ODN cost ($/µg of DNA)				**up to 295.2**

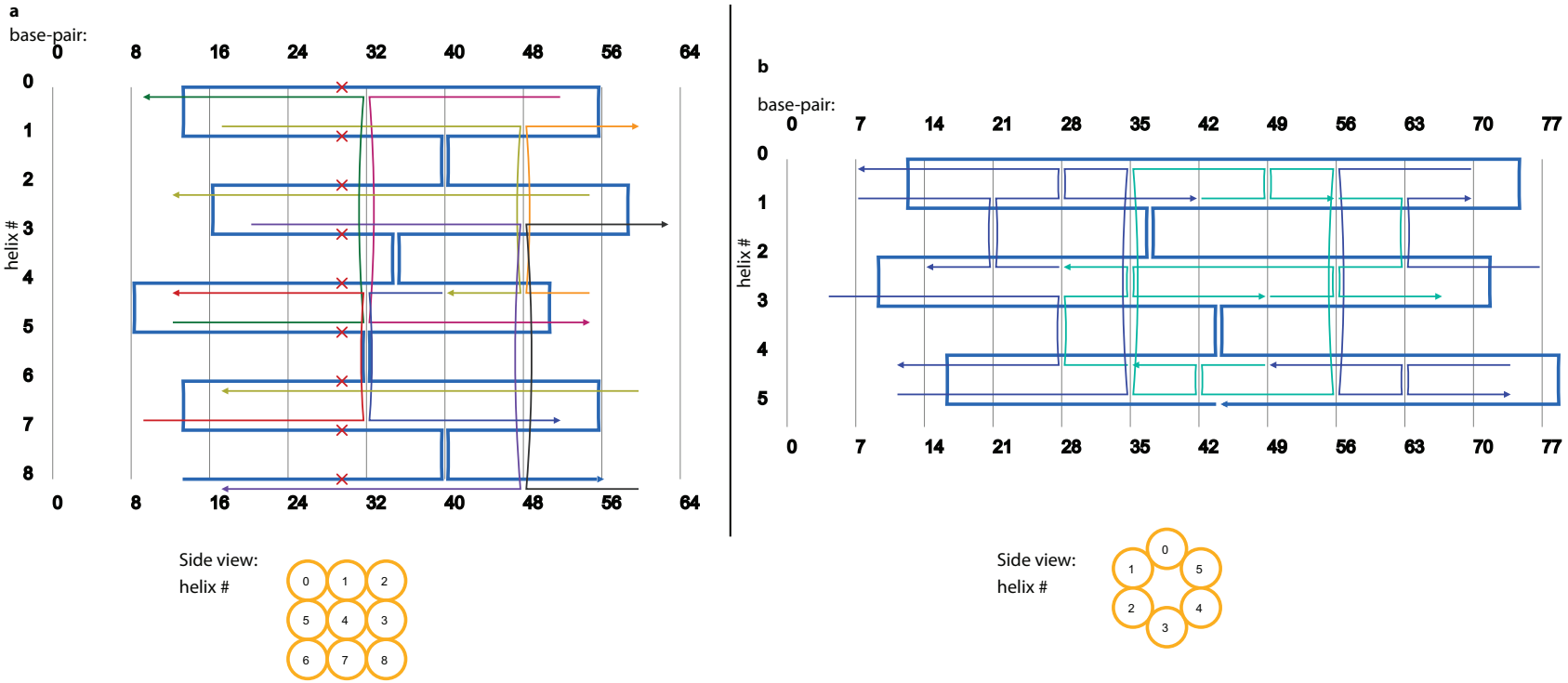
[1] 15 mL centrifuge tube (1), PCR tubes (5), 1,5 mL tubes (5), 0.1-200 µl pipette tips, Sep-Pak C18 cartridges
[2] Excluding incubation times, approximate cost of post-doc in Sweden per hour, including social fees and indirect costs

Supplementary Figure 13: Yield and cost calculation of the 378nt long ODN produced by the MOSIC method. **Top**: The intensities of the 0.1 Kb DNA ladder (lane 1) were measured by imageJ and used as standard for approximately measuring the final yield obtained from BseGI digestion of RCA product (lane 2). **Bottom**: Cost calculation for total *variable* or running costs (not including start-up costs for initial gene-synthesis of about 0.8 USD/base) per µg of long ODN produced by the MOSIC method. The total cost at 9.92 USD/µg is 15-30 times lower compared with a quote for the same sequence synthetically produced at 1476 USD / 5-10 µg from a commercial vendor. Enzyme and dNTP costs are based on list prices for small, standard research level, quantities.

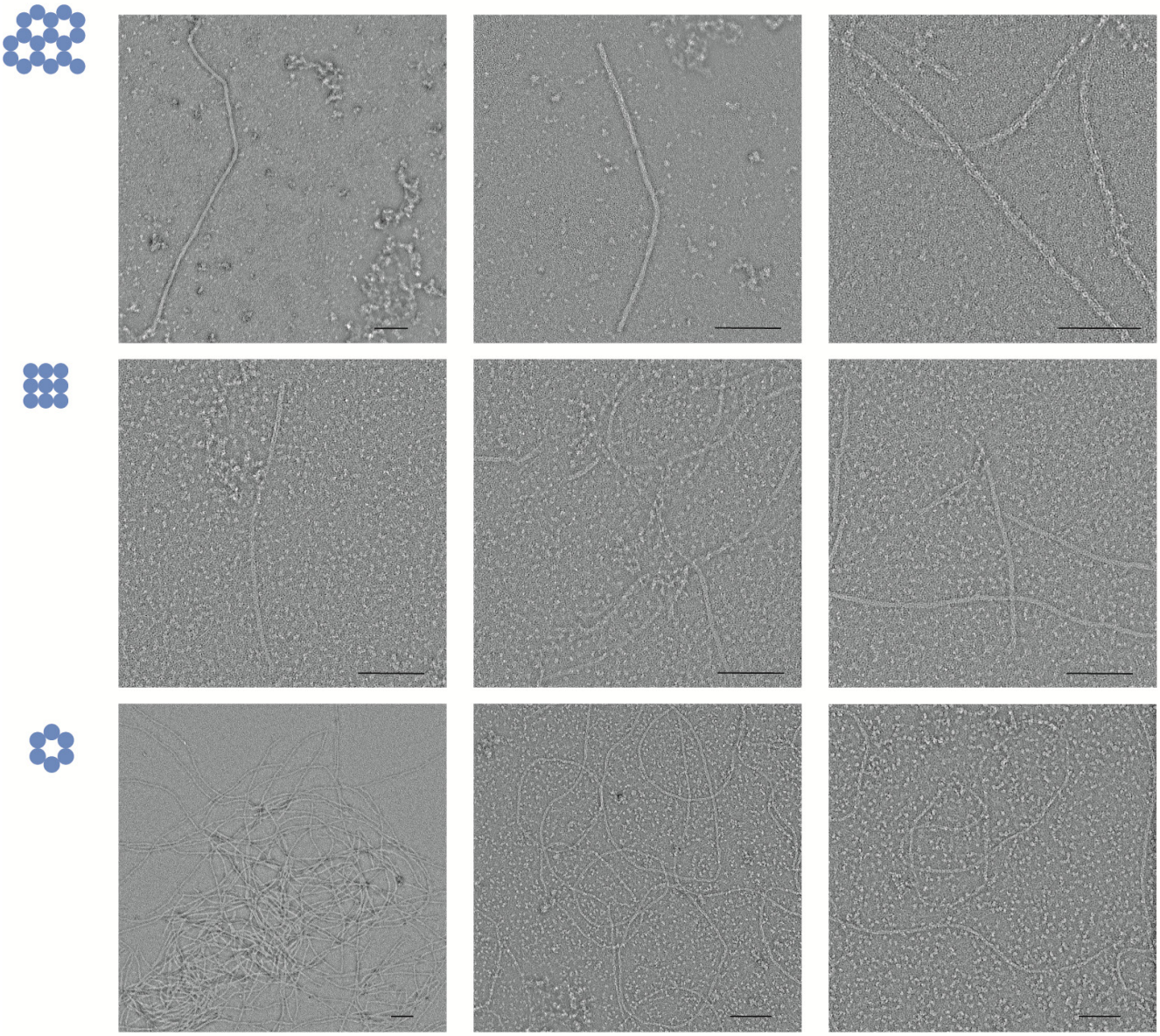


Supplementary Figure 14: Design of the 17 helix bundle (17HB) ultra-small DNA-origami polymerizing brick. **a** Side view showing the arrangement of helices. **b** DNA origami design diagram. The 378 bases long MOSIC oligo (black arrow) is used as a scaffold. **c** Polymerization. As multiple ultra-small origamis encounter each other, the prodtruding 4 bases of each staple oligo acts as a sticky ends leading to the formation of linear, periodic, 17HB nano-tube polymers.

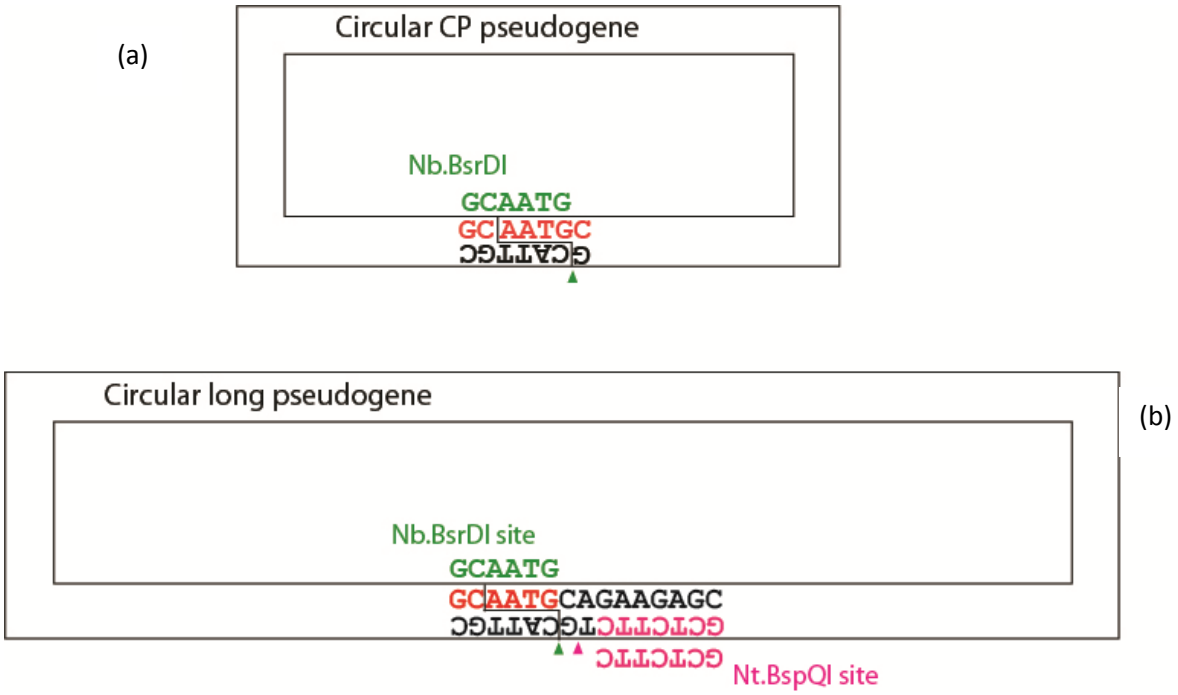
Nature Methods: doi:10.1038/nmeth.2503



Suppelementary Figure 15: Designs of the 9-, **a**, and 6- , **b**, helix-bundle (9HB and 6HB) ultra-small DNA-origami polymerizing bricks folded from the 378 nt long MOSIC oligo. Similar polymerization schemes as the ones in **S14-c** were used to form nanotubes.



Supplementary Figure 16: TEM images of 17-, (**top**), 9-, (**middle**), 6-, (**bottom**), helix bundle ultra small DNA origami polymers folded by using the 378 nt long MOSIC oligo as scaffold, scale bars 100 nm.



Supplementary Figure 17: Schematic representation of the circular crystal pseudogene (**a**) and long ODN pseudogene (**b**) to highlight the nicking sites after ligation.

CGTCTCACATTGCATAATTCATCCGCGCGAAGCGCGGATGAAGAGCAGCCTGTACGGACATCACATCCGCGCG
AAGCGCGGATGTGGAGCAGCCTGTACGGACATCACATCCGCGCGAAGCGCGGATGTGTCTGATGTGGCTGCC
ATCCGCGCGAAGCGCGGATGGCACACCGTACACCGTACACCGTCATCCGCGCGAAGCGCGGATGACTCTGAT
GTGGCTGCCATCCGCGCGAAGCGCGGATGGCGAGCAGCCTGTACGGACATCACATCCGCGCGAAGCGCGGAT
GTGTCTGATGTGGCTGCCATCCGCGCGAAGCGCGGATGGCATAATTGCATTGGAGACG

Suppementary Table 1: Complete sequence of crystal pseudogene (CP). The pseudogene includes the hairpin sequences (highlighted in grey), and the BsmBI restriction site (highlighted in turquoise).

	17 Helix bundle ultra-small origami staples	NTs
1	CCGGTTTCGGGCCGTATTGACG	22
2	GCGGTATTATAGGGTGCACTAAA	23
3	TATGGCACGGATAAAGTTCTGC	22
4	ATGAGCACTGATACATCGAGTAT	23
5	TTCCTTTGTTTCGAAGAACGTT	22
6	TGAGAGTTTTCGCCCATTTCCT	22
7	AAGATTTCTAATCAACAGCGGT	22
8	ATATACTGGATCATGTCGCCCTT	23
9	CCGCTCATTTACCCCTAAATG	21
10	CTGAGAGTGGGTGATGCCTTCCT	23
11	TGCTTCAATCGAAATGTTGGC	21
12	GAAGATCAGTTATATTGATGCT	22
13	AAGAAAACGCTGGTGAAAGTAA	22
14	TCACCCAGAAAAGGAAGATTGC	22
15	GTTTGTATGAGTTCAATAACC	21
16	TTTTGCGGCATTATTCAACCTT	22
17	ATTCCATTTCCGTACATTCAA	21
	9 Helix-bundle ultra-small origami staples	
1	ATTCCCTTTTTTGCGGCATTCAAATATGTATCCGCTCATCCTT	43
2	CCAGAAACGCTGGTGAAAGTAAAACATTTCACAATAAC	38
3	AGAGTATGAGTATTCAAGATGCTGAAGATCAGTTGGGTAGGA	42
4	GAGTGGGTTACATCTAAATGCTTCAATGGGGAAATGCGGTATTATCCCGTATTGACG	57
5	CCTGAGAATATGTGGCGTGCACGGAACC	28
6	AGATGAGCGTGTCGCTCAC	19
7	GAGAGTTTTCGCCCCGAAGTTGCCTTCCTGTTTTTGCCCTT	41
8	ATGATTTGTTTATTTTTCTAAATACATTAACGTTTTCCAATG	42
9	CCTAGCACTTTTAAAGTTCTGCCTGGATCTCAACAGCGGTA	41
10	CCGGCACTTTTCAATATTGAAAAGCAC	27
	6 Helix-bundle ultra-small origami staples	
1	GAGACAAAAGGAAGAGTATGAGTATTGATTTCCAATGCTCAT	42
2	ATCCGATGGCACTTTTCGGGGGCTCACCGTCGCCC	35
3	TTTCCGTCAGAAACGCTGGTGTGACGCCGAGCACT	35
4	TTTAAAGAGAACGTACTGGAT	21
5	GCCTTCCTGTTTTTAAATGTGACATTCAAATATGT	35
6	TTATTCCATTGAAATAACCCTGATA	25
7	CTCAACAACGAGTGAAGATGCTGAAGATCAATTTT	35
8	TTTTTCTAAATCACGGAACCCC	22
9	GCGGCGTTGGGTGCGCGGTAA	21
10	AATGCTTGAGAGTTTTCGCCCCGATTCTGCTATGTGGTTTA	41
11	GATCCTTCAATAATCTTTTTT	21
12	TATTTGCGCGGTATTATCCCGTATAAAGTAAGGTTACATCCAACA	45

378 nt long ODN pseudogene (the final 378 nt ODN sequence is marked in yellow):

GGTCTCACATTGCATAATTAACATCCGCGGAACGCGGATGTTCCGGCGTCAATACGGGATAATACCGC
GCCACATAGCAGAACTTTAAAAGTGCTCATCATTGGAAAACGTTCTTCGGGGCGAAAACTCTCAAGGA
TCTTACCGCTGTTGAGATCCAGTTCGATGTAACCCACTCGTGCACCCAACTGATCTTCAGCATCTTTT
ACTTTCACCAGCGTTTCTGGGTGAGCAAAAACAGGAAGGCAAAATGCCGCAAAAAAGGGAATAAGGGC
GACACGGAAATGTTGAATACTCATACTCTTCCTTTTTCAATATTATTGAAGCATTTATCAGGGTTATT
GTCTCATGAGCGGATACATATTTGAATGTATTTAGAAAAATAAACAAATAGGGGTTCCGCGCACATTT
CCCCGAAAAGTGCATCCGCGGAACGCGGATGCAGCTCTTCTGCATTGGAGACC

Supplementary Table 2 Sequences used in production and verification of the 378 nt long ODN.

Supplementary Table 3: Data collection and refinement statistics (molecular replacement)

	DNA-x001
Data collection	
Space group	H3 (146)
Cell dimensions	
a, *b*, *c* (Å)	106.44, 106.44, 95.15
α, β, γ (°)	90, 90,120
Resolution (Å)	42.3-4.79(5.05-4.79) *
R_{merge}	0.052(0.801)
$I / \sigma I$	16.3(2.3)
Completeness (%)	99.9(100)
Redundancy	5.8(5.8)
Refinement	
Resolution (Å)	42.3-4.79
No. reflections	1980
R_{work} / R_{free}	0.185 / 0.205
No. atoms	
Nucleic acid	863
Ligand/ion	0
Water	0
B-factors	(Ask for input)
Nucleic acid	20
Ligand/ion	0
Water	0
R.m.s. deviations	
Bond lengths (Å)	0.01
Bond angles (°)	2.46

*Number of xtals for each structure should be noted in footnote. *Values in parentheses are for highest-resolution shell.

[AU: Equations defining various *R*-values are standard and hence are no longer defined in the footnotes.]
[AU: Ramachandran statistics should be in Methods section at the end of Refinement subsection.]
[AU: Wavelength of data collection, temperature and beamline should all be in Methods section.]

```python
#!/usr/bin/env python
# encoding: utf-8

# Python standard libraries
import string
import os
import sys

'''
Created by Bjorn Hogberg in 2012
Provides the geneAssembler class that performs the
low-level gene assembly of oligos with hairpins between.
'''

class newGene():

    def __init__(self,initSequence=[]):
        self.sequence = initSequence

    def complement(self):
        return self.complement_to(self.sequence)

    '''
    def reverse(self,sequence):
        return sequence[::-1]
    '''

    def complement_to(self,sequence):
        '''Returns the wc-complement to the sequence'''
        comp = {
                'a':'T', 'A':'T', 'c':'G', 'C':'G',
                'g':'C', 'G':'C', 't':'A', 'T':'A'}
        temp_c = ''
        for letter in sequence:
            if letter in comp:
                temp_c = temp_c + comp[letter]
        return temp_c[::-1]

    def appendSequence(self,sequenceToAppend=''):
        '''Appends sequence to the pseudogene'''
        if len(sequenceToAppend)==0:
            print 'Warning: appended sequence empty'
        self.sequence=self.sequence+list(sequenceToAppend)


    # This variable contains the loopside sequence and the
    # actual GAA loop:
    #
    #
    #      loopside|restriction site
    #    A         |            | NN  |              -
    #   / \ G C G C - G G  A T G - N N  S t a p l e  N  /
    #  A    . . . .  . .  . . .  . .
    #   \ / C G C G - C C  T A C - S t a p l e  N-1 - - -
    #    G
    #
    #
    #
    LoopsideSeq = 'GCGCGAAGCGC'

    def appendHairpin(self):
        # Store the last and next-to-last bases to
        # create NN
        #                             Cut
        #      loopside|restriction site |
        #    A         |           | NN  |              -
        #  / \ C G C - G G  A T G - N N  S t a p l e  N  /
        # A    . . .   . .  . . .  . .
        #  \ / G C G - C C  T A C - S t a p l e  N-1 - - -
        #    G                   |
        #                       Cut
        #
        #
        NN=self.complement_to(self.sequence[-2:])
        self.appendSequence('CATCC')
        self.appendSequence(self.LoopsideSeq)
        self.appendSequence('GGATG')
        self.appendSequence(NN)


def cleanSeq(input):
    '''Reads a sequence and stores only bases'''
    upper = {
        'a':'A', 'A':'A', 'c':'C', 'C':'C',
        'g':'G', 'G':'G', 't':'T', 'T':'T'
            }
    tempSeq = ''
    for letter in input:
        if letter in upper:
            tempSeq = tempSeq + upper[letter]
    return tempSeq

#
# READ THE LIST OF OLIGOS FROM THE FILE PASSED AS ARGUMENT
#
fileName=sys.argv[1]
print 'Opening oligo file {0}'.format(fileName)
inFile=file(fileName,'r')
oligos=[]
i=0
for line in inFile:
    i+=1
    oligos.append(cleanSeq(line))
    if oligos[i-1].count('GGATG')>0:
        print 'Warning! Sequence GGATG found'
        print 'in oligo on line {0}'.format(i)
    if oligos[i-1].count('CATCC')>0:
        print 'Warning! Sequence CATCC found'
        print 'in oligo on line {0}'.format(i)
inFile.close()


'''
Restriction enzyme recognition sequence for the
circularization protocol, in this example HindIII.
Needs to be replaced if sequence is present in the oligos.
'''
circRestrict='AAGCTT'

'''
Nicking enzyme recognition sequence. Sequence of the
strand that is nicked. In this example, Nb_BsrDI.
Needs to be replaced if sequence is present in the oligos.
'''
nickEnzyme='CATTGC'

#
# INITIALIZE THE newGene CLASS AND START ADDING THE
# OLIGOS TO BUILD UP THE PSEUDOGENE
#
pgene=newGene()

pgene.appendSequence(circRestrict)
pgene.appendHairpin()

for oligo in oligos:
    pgene.appendSequence(oligo)
    pgene.appendHairpin()

pgene.appendSequence(nickEnzyme)
pgene.appendSequence(circRestrict)

if ''.join(pgene.sequence).count(circRestrict) > 2 or \
   ''.join(pgene.sequence).count(
            pgene.complement_to(circRestrict)) > 2:
    print 'Warning! The Circularization restriction site'
    print 'seems to be present within the oligo region.'

if ''.join(pgene.sequence).count(nickEnzyme) > 1 or \
   ''.join(pgene.sequence).count(
                pgene.complement_to(nickEnzyme)) > 1:
    print 'Warning! The nicking site seems to be present'
    print 'within the oligo region.'


outFile=file('pseudogene_output.txt', 'w')
outFile.write(''.join(pgene.sequence))
outFile.close()
print 'Pseudogene written successfully.'
```

Supplementary Note 1 Python code used to assemble a pseudogene out of a list of oligonucleotides in a text file.