EUROPEAN LABORATORY FOR PARTICLE PHYSICS
CERN – SL Division

CERN/SL/91–52 (AP)

SIXTRACK

Version 1.1

Single Particle Tracking Code Treating Transverse Motion with Synchrotron Oscillations in a Symplectic Manner

User's Reference Manual

F. Schmidt

Abstract

The aim of SIXTRACK is to carry two nearby particles through those complicated structures over large number of turns taking into account the full six–dimensional phase space including synchrotron oscillations in a symplectic manner. It allows to predict the long–term dynamic aperture which is defined as the border between regular and chaotic motion. This border can be found by studying the evolution of the distance of phase space of two initially nearby particles. Parameters of interest like non–linear detuning and smear are determined via a post–processing of the tracking data. An analysis of the first order resonances can be done and correction schemes for several of those resonances can be calculated. Moreover there is the feature to calculate a one–turn map of such complicated structures as the LHC to very high order, using the differential algebra techniques of M. Berz. This map allows a subsequent theoretical analysis like normal form procedures which are provided by E. Forest.

Version 1.1 allows now the calculation of these maps in the five– (four transverse dimensions plus the dependence on the relative momentum deviation) and the full six–dimensional case. The postprocessing prozedure has also been extended to the six–dimensional case. The linear elements are usually treated as thick elements in SIXTRACK, there is now also a thin–lens version which allows an additional speedup by a factor of two. All versions, including the programs to analyze maps, are available in a scalar mode for Apollo, Cray, IBM and VAX and also in a vector mode for Cray and IBM. As a new feature a decoupling subroutine has been added. Moreover a common header of output data and the format of these data has been found for MAD and SIXTRACK tracking data. This allows the analysis of tracking data from MAD with the SIXTRACK postprocessor.

Geneva, Switzerland
24 September 1992

Contents

List of Tables

Chapter 1. Introduction

The Single Particle Tracking Code SIXTRACK is optimized to carry two particles [1] through an accelerator structure over a large number of turns. It is an offspring of RACETRACK [1] written by Albin Wrulich and its input structure has been changed as little as possible so that slightly modified RACETRACK input files or those of other offsprings like FASTRAC [2] can be read in.

The main features of SIXTRACK are:

1. Treatment of the full six–dimensional motion including synchrotron motion in a symplectic manner [3]. The energy can be ramped at the same time considering the relativistic change of the velocity [4].
2. Detection of the onset of chaotic motion and thereby the long–term dynamic aperture by evaluating the Lyapunov exponent.
3. Postprocessing procedure allowing
   - calculation of the Lyapunov exponent
   - calculation of the average phase advance per turn
   - FFT analysis
   - resonance analysis
   - calculation of the average, maximum and minimum values of the Courant Synder emittance and the invariants of linearly coupled motion
   - calculation of smear
   - plotting using the CERN packages HBOOK, HPLOT and HIGZ [5, 6, 7]
4. Calculation of first–order resonances and of correction schemes for the resonances [8].
5. Calculation of the one–turn map using the differential algebra techniques of M.Berz [9].
6. A vectorized version is available, where the two particles, the number of amplitudes, the different relative momentum deviations and several seeds for the random distribution of multipole errors can be treated in parallel [10].
7. Operational improvements:
   - free format input
   - optimisation of the calculation of multipole kicks
   - improved treatment of random errors
   - each binary data–file has a header describing the history of the run (Appendix D)

In the following the general features of the program, the format of source code, storage requirements and speed are briefly summarized. As stated above, the input part of SIXTRACK was modified without loosing the possibility to read input files of RACETRACK or its offsprings. This applies with minor modification to the machine geometry described in (chapter 2). In most cases, however, SIXTRACK input can not be read by other programs as SIXTRACK allows free format input, which is a considerable improvement compared to the fixed format input of the other

[1] Two particles are needed for the detection of chaotic behavior.

programs. The input is still line oriented. Each line of 80 characters is treated as one string of input in which a certain sequence of numbers and character strings is expected to be found. The numbers and character strings must be separated by at least one blank, floating point numbers can be given in any format, but must be destinguished from integer numbers. Omitted values at the end of an input line will keep their default values (B.1), and lines with a slash '/' in the first column will be ignored by the program.

The program is kept in Patchy [11] format. This has considerable advantages:

- the source code can be kept in binary format, which saves a lot of storage space
- for program development the binary (PAM) file is transformed into a CARDS file, where blocks of COMMON and PARAMETER statements need be defined at the beginning of the program only
- the CARDS file can be transformed into a FORTRAN file for compilation or back into binary format
- machine specific features can be selected with flags, with different versions of SIXTRACK for Apollo,Cray,IBM and VAX

The organization of the source file in this way makes it very easy to change the size of arrays, which may be necessary when very large accelerator structures are to be investigated. At the time being the LHC is studied with roughly 2500 multipoles, so that the program needs 8 Mbytes. On top of that the differential algebra package needs another 12 Mb (in the 4d version) to allow the calculation of a one–turn map of the accelerator to 10th order. This can be handled by shifting large common blocks above the 16 Mb limit which can be done by going from the CMS to the CMSXA system on the VM at CERN [12].

At the CERN IBM one turn of the two particles in the full LHC lattice with all multipole errors in the dipoles and correction elements takes approximately 134 ms, so that 10 000 turns with 22 minutes of cpu time is roughly the upper limit that can be reached without going to dedicated emulators that can run 24 hours a day. There is, however, a vectorized version of the program, which allows a speedup by approximately a factor of 4.5 when 60 particles are treated in parallel [10]. The CPU–time on the IBM (for the Cray these numbers are smaller by a factor of 3 due to vectorization) needed to determine with the differential algebra option the one–turn map with four and six parameters (four– and six–dimensional transverse motion) of the same LHC lattice is given in the table 1.1 below as a function of the order to which the map is calculated.

Table 1.1: CPU Time for One–turn Map Calculation

	4 dimensions	6 dimensions
order	cpu time (s)	cpu time (s)
1	17	27
3	37	101
5	94	475
7	252	1996
9	655	8017

Per order roughly 1.6, 2.0 times more cpu time is needed in the four– and six–dimensinonal cases respectively (see [13]).

For detailed questions concerning rounding errors, calculation of the Lyapunov exponent and determination of the long–term dynamic aperture, see [14].

In chapter 2, the input structure of SIXTRACK is discussed in detail. To facilitate the use of the program, a set of appendices are added, giving a list of keywords (Appendix A), a list of default values (Appendix B), the input and output files (Appendix C), a description of the data structure of the binary data–files (Appendix D) and tracking examples (Appendix E).

Chapter 2. Input Structure

The idea of RACETRACK input is to use a sequence of input blocks, each block with a specific keyword in the first line, the keyword 'NEXT' in the last line and the input data in the lines in between. The keyword 'ENDE' ends this sequence, and all blocks after this keyword are ignored. This system makes it easy to read input and allows easy change and addition of input blocks. It was therefore also used in SIXTRACK.

The fixed format input of RACETRACK was however not acceptable as it demands precise knowledge of the correct position of each item in a line of input and also definite format and precision of numerical values. A free format input was introduced to overcome these difficulties.

2.1 General Input

2.1.1 Program Version

Description The *Program Version* input block determines if all of the input will be in the input file # 3 or if the geometry part of the machine (see 2.2) will be in a separate file # 2. The latter option is useful if tracking parameters are changed but the geometry part of the input is left as it is. The geometry part can be produced directly from a MAD [15] input file (2.1.5).

Keyword FREE or GEOM

Number of data lines 0

Format *keyword comment title*

keyword The first four characters of the first line of the input file # 3 are reserved for the keyword (FREE for free format input with all input in file # 3; GEOM if the geometry part is in file # 2)

comment Following the first four characters, 8 characters are reserved for comments

title The next 60 characters are interpreted as the title of the output file # 6

2.1.2 Print Selection

Description Use of the *Print Selection* input block causes the printing of the input data to the output file # 6. It is advisable to always use this input block to have a complete protocol of the tracking run.

Keyword PRIN

Number of data lines 0

2.1.3 Comment Line

Description An additional comment can be specified with this block. It will be written to the binary data files (Appendix D) and will appear in the postprocessing output as well.

Keyword COMM

Number of data lines 1

Format A string of up to 80 characters.

2.1.4 Iteration Errors

Description For the processing procedures, the number of iterations and the precision to which the processing is to be performed are chosen with the *Iteration Errors* input block. If the input block is left out, default values will be used.

Keyword ITER

Number of data lines 1 to 4

Format Each data line holds three values as in table 2.1.

Table 2.1: Iteration Errors

data line	integer	double	default value	number of iterations for	demanded precision of	variations of
1	ITCO		50	closed orbit calculation		
		DMA	1e-12		closed orbit displacements	
		DMAP	1e-15		derivative of closed orbit displacements	
2	ITQV		10	Q adjustment		
		DKQ	1e-10			quadrupole strengths
		DQQ	1e-10		tunes	
3	ITCRO		10	chromaticity correction		
		DSM0	1e-10			sextupole strengths
		DECH	1e-10		chromaticity correction	
4		DE0	1e-9			momentum spread for chromaticity calculation
		DED	1e-9			momentum spread for evaluation of dispersion
		DSI	1e-9		desired orbit rms value; compensation of resonance width	

2.1.5 MAD – SIXTRACK Conversion

Description The program *MAD8SIX* uses a MAD [15] output file produced with the SURVEY option. This file is read from unit # 2 and a SIXTRACK geometry file is produced on unit # 3, which can be read with the GEOM option of the *Program Version* input file (2.1.1). No extra input file is needed.

2.2 Machine Geometry

2.2.1 Single Elements

Description The *Single Elements* input block defines the name and type of linear and nonlinear elements, the inverse bending radius or multipole strength respectively, and the strength and length of the elements. Linear and nonlinear elements are distinguished by length; linear elements have a nonzero length and nonlinear elements have zero length. Both kinds of elements can appear in the input block in arbitrary order. The input line has a different format for linear and nonlinear elements. Moreover, the multipoles, being a set of nonlinear elements, are treated in a special way. The maximum number of elements is set as a parameter (see Appendix B.2).

Keyword SING

Number of data lines variable

Format See the following three sections.

2.2.1.1 Linear Elements

Description Each linear single element has a name, type, inverse bending radius, focussing and a nonzero length.

Format *name type* ϱ^{-1} K *length*

name May contain up to eight characters

type As shown in the table 2.2

ϱ^{-1} Inverse bending radius in m^{-1}

K Focussing strength in m^{-2}

length Magnet length in meters

Table 2.2: Different Types of Linear Elements

type	ϱ^{-1}	K	description
0	0	0	drift length magnet
1	X	0	horizontal (rectangular) bending
2	0	X	quadrupole (− focussing, + defocussing)
3	X	0	horizontal (sector) bending
4	X	0	vertical (sector) bending
5	X	0	vertical (rectangular) bending
6	X	X	horizontal combined function magnet
7	X	X	vertical combined function magnet
8	X	0	edge focussing

Remarks

1. For the horizontal plane the bending radius is defined to be negative ($\varrho < 0$). This is different from other programs like MAD [15].
2. $K < 0$ corresponds to a horizontal focussing quadrupole.
3. For the length of an edge focussing element (type=8) the same value must be used as for the corresponding bending magnet. A sector bending magnet is transformed into a rectangular magnet with an edge focussing element of positive length on either side, while for the opposite transformation a negative length is required.
4. It is important to note that the splitting of a rectangular magnet, which is sometimes necessary if multipole errors are to be introduced, does change the linear optics. It is therefore advisable to replace the rectangular magnet with a sector magnet, which can be split without affecting the linear optics, and make an overall transformation into a rectangular magnet via edge focussing elements. Do not forget to use the total length of dipole as the length of the edge focussing element.

2.2.1.2 Nonlinear Elements

Format *name type* K_n*–strength rms–strength length*

name May contain up to eight characters

type As shown in table 2.3

K_n*–strength* Average multipole strength

rms–strength Random multipole strength

length Must be = 0

Table 2.3: Different Types of Nonlinear Elements

type	strength	description
0	–	observation point (for instance for aperture limitations)
1 –1	$b_1(rad \cdot m^0)$ a_1	horizontal bending kick vertical bending kick
2 –2	$b_2(rad \cdot m^{-1})$ a_2	normal quadrupole kick skew quadrupole kick
⋮		
10 –10	$b_{10}(rad \cdot m^{-9})$ a_{10}	normal 20th pole skew 20th pole

Remarks

1. Because the horizontal bending magnet is defined to have a negative bending radius, the sign for normal elements is different from other programs like MAD, while skew elements have the same sign.

2. Again contrary to other programs the factor $(n-1)!$ is already included in the multipole strength, which is defined as follows:
   - for normal elements $b_n(SIXTRACK) = \frac{-1}{(n-1)!} L_{element} b_n(MAD)$
   - for skew elements $a_n(SIXTRACK) = \frac{1}{(n-1)!} L_{element} a_n(MAD)$
3. Unlike in RACETRACK, the horizontal and vertical displacements do not fit into the 80 character input lines of SIXTRACK. They have to be introduced in a separate *Displacements of Elements* input block (see 2.2.4).

2.2.1.3 Multipole Blocks

Description A set of normal, normal–rms, skew and skew–rms errors can be combined effectively. The actual values for the strengths have to be given in a separate *Multipole Coefficient* input block (see 2.3.1) which must have the same name.

Format *name type cstr cref length*

name May contain up to eight characters

type Must be = 11

cstr The bending strength given in the *Multipole Coefficient* input block (2.3.1) is multiplied with this factor.

cref The reference radius given in the *Multipole Coefficient* input block (2.3.1) will be multiplied by this factor. If it is zero the multipole block will be ignored.

length Must be = 0

Remark The definition of the multipole strength in a block will be given in (2.3.1).

2.2.2 Block Definitions

Description In four–dimensional transverse tracking, the linear elements between nonlinear elements can be combined to a single linear block to save computing time.

Keyword BLOC

Number of data lines variable but at least one

Format

- first data line: *mper msym(1)* ... *msym(mper)* (integers)
- from second data line on: *block–name* {*element–name*}

mper Number of superperiods. The following set of blocks is considered a *superperiod.* The accelerator consists of *mper* superperiods.

msym(i) ± 1 for each superperiod. If *msym(i)*=1, the *i*'th superperiod will be built up in the order in which linear elements appear in the blocks below. If *msym(i)*=–1, the superperiod will be built up in reverse order.

block–name The name of the block with up to eight characters

element–name The element names have to appear as a linear element in the list of 'single elements' (2.2.1.1). If one line is too short to contain all the elements of a block, a line with additional elements to the same block can be added. At least 5 (five) blanks must appear at the beginning of the extra line so that names of blocks and names of linear elements in a block can be distinguished.

Remarks

1. When synchrotron oscillation is introduced, the linear elements can no longer be lumped into one block, because in that case even a drift length magnet is a nonlinear element with respect to the longitudinal plane. However, the block structure is still kept to make use of the speed–up in case one can restrict the studies to the four–dimensional case.
2. The maximum number of blocks and the maximum number of entries in each block are defined as parameters (Appendix B.2).
3. In the vectorized version, the inversion of a superperiod ($msym(i) = -1$) is not allowed.

2.2.3 Structure Input

Description The model of the accelerator is put together by constructing a sequence of blocks of linear elements, nonlinear elements, observation points, and possibly a cavity with the keyword 'CAV'. This name does not appear in the list of single elements (2.2.1). Instead, its parameters are given in the *Synchrotron Oscillations* input block (2.6.3).

Format { *structure–element* | CAV | GO }

structure–element Structure elements must appear as nonlinear and observation elements in the single element list or in the list of blocks of the *Block Definition* input block (2.2.2).

CAV A cavity can be introduced by a keyword 'CAV'. This element does not appear in the single element list (2.2.1).

GO Starting point: the keyword 'GO' denotes where the tracking is started and where the tracked coordinates are recorded at each turn.

Remark Repetition of parts of the structure is indicated by parentheses with a multiplying factor N in front of them. If the left parenthesis '(' occurs in a line of input, the factor N is expected to be found in the preceding characters. If the characters are blank, N is set to 1. The right parenthesis ')' signals the end of the sequence to be repeated.

2.2.4 Displacement of Elements

Description This block allows to displace nonlinear elements in horizontal and vertical positions. With the rms values of the horizontal and vertical displacements it is possible to achieve a displacement that is different from element to element.

To simulate a measured closed orbit at the position of nonlinear elements, it is convenient to use the *Displacement of Elements* input block instead of trying to produce a closed orbit by dipole kicks.

Keyword DISP

Number of data lines variable

Format *xd xdrms zd zdrms*

xd Horizontal displacement

xdrms Rms of horizontal displacement

zd Vertical displacement

zdrms Rms of vertical displacement

Remark In RACETRACK the displacements had been included in the *Single Element* input block (2.2.1). In SIXTRACK they must be given in the separate *Displacement of Elements* input block because of the limited length of one line of input.

2.3 Special Elements

One advantage of SIXTRACK, that has been adopted from RACETRACK, is that it easily allows to define elements for a specific purpose. The special elements implemented till now are found in this section.

2.3.1 Multipole Coefficients

Description Sets of normal and skew multipoles of up to tenth order, each with an rms value, can be combined with this block. The multipole kick is calculated using a Horner scheme which saves considerably in computation time. Moreover, using the multipole block reduces the number of elements in the single element list (2.2.1).

Keyword MULT

Number of data lines 2 to 10

Format

- first data line: *name* R_0 δ_0
- data lines 2 to 11: B_n $rms{-}B_n$ A_n $rms{-}A_n$

name Name of the multipole block which must appear in the list of single elements (2.2.1.3).

R_0 Reference radius (in mm) at which the magnet errors are calculated. This makes it convenient to use values from field measurements.

δ_0 Bending strength of the dipole (in mrad). Field errors of line 2–11 are taken to be relative to the bending strength.

Remarks

1. The B_n and A_n are related to the b_n, a_n of the single nonlinear element (2.2.1.2) in the following way:
$$b_n = \delta_0 B_n R_0^{1-n} 10^{3n-6}; a_n = \delta_0 A_n R_0^{1-n} 10^{3n-6}$$
2. The sign convention and the factorial ($n!$) are treated as for the single nonlinear elements in (2.2.1.2).
3. Multipoles of different names can be set to be equal using the 'ORG' input block.

2.3.2 Aperture Limitations

Description This input data block is used to introduce additional collimators or aperture limitations in the machine. Each nonlinear element can be used for this purpose. Rectangular or elliptical shapes of the aperture limitations are allowed. On top of that there is a general (rectangular) aperture check at each non–zero length element. The general aperture values are chosen to be large enough (B.1) to define the short–term dynamic aperture.

Keyword LIMI

Number of data lines variable

Format *name type–of–limitation xaper zaper*

name The name of any nonlinear (zero length) element in the *Single Element* input block (2.2.1.2) except multipole blocks (2.2.1.3).

type–of–limitation Two types of aperture limitations are allowed:
'RE' for a rectangular aperture shape, i.e.

$$x_i < xaper, z_i < zaper$$

'EL' for an elliptical aperture shape, i.e.

$$\frac{x_i^2}{xaper^2} + \frac{z_i^2}{zaper^2} < 1$$

xaper Aperture in the horizontal plane in mm

zaper Aperture in the vertical plane in mm

2.3.3 Power Supply Ripple

Description If power supply ripple is to be considered this input data block can be used. A nonlinear quadrupole is expected as a ripple element (type=2 and zero length in the single element list (2.2.1.2)), but in principle other nonlinear elements are also allowed. Ripple depth, ripple frequency and starting phase of the ripple frequency are the input parameters.

Keyword RIPP

Number of data lines variable

Format *name ripple–depth ripple–frequency start–phase*

name Name of the nonlinear element in the 'single element' block (2.2.1.2)

ripple–depth Maximum kick strength of the ripple element, a quadrupole kick is usually expected

ripple–frequency Given in number of turns (a real value is allowed) of one ripple period

start–phase Initial phase of the ripple element

2.4 Organizing Tasks

In this section the input data blocks are described, which are used to organize the input structure.

2.4.1 Random Fluctuation Starting Number

Description If besides mean values for the multipole errors (Gaussian) random errors should be considered this input data structure is used to set the start value for the random generator.

Keyword FLUC

Number of data lines 1

Format *izu0 nmac* (integers)

izu0 Start value for the random number generator

nmac In the vectorized version the number of different starting seeds can be varied. Each seed is calculated as $k \times izu0$ where k runs from 1 to *nmac* which can not exceed 5 to save storage space (see list of parameters in Appendix B.2).

Remarks

1. The RANECU random generator [16] is used as it produces machine independent sequences of random numbers.
2. If the starting point has to be changed or another nonlinear element is to be inserted, this can be done without changing the once chosen random distribution of errors by using the *Organization of Random Numbers* input block.

2.4.2 Organization of Random Numbers

Description Working on a lattice for an accelerator often requires to introduce new nonlinear elements. In those cases simply introducing this new element means that the previously chosen random distribution of the errors will be changed and with it often the linear parameters. This input data block is mainly used to avoid this problem by reserving extra random numbers for the new elements. It also allows to change the observation point without affecting the machine. The random values of different nonlinear elements including blocks of multipoles can be set to be equal to allow to vary the number of nonlinear kicks in one magnet which clearly should have the same random distribution for each multipolar kick. Finally multipole sets with different name can be made equal with this input data block.

Keyword ORGA

Number of data lines variable

Format *ele1 ele2 ele3*
The data lines can be set in three different ways:

1. Ele1 = 'name' where name $\neq$ MULT
   Ele2 = ignored
   Ele3 = ignored
   The nonlinear element or multipole set will have its own set of random numbers.

2. Ele1 = 'name1' where name1 $\neq$ MULT
   Ele2 = 'name2'
   Ele3 = ignored
   The nonlinear element or multipole block Ele1 has the same random number set as those of Ele2, if it follows Ele2 as the first nonlinear element in the structure list (2.2.3).

3. Ele1 = 'MULT'
   Ele2 = 'name2'
   Ele3 = 'name3'
   The multipole set 'name3' is set to the values of the set 'name2'. random errors are not influenced in this case.

Remarks

1. A simple change of the starting point, by placing a 'GO' somewhere in structure, used to change the machine optics as the random numbers were shifted, too. Simply calling this block even without a data line, will always fix the sequence of random numbers to start at the first multipole in the structure.

2. This input data block must follow the definition of the multipole block, otherwise multipoles cannot be set equal (option 3).

3. Do not use the keyword 'MULT' in the single element list (2.2.1).

2.4.3 Combination of Elements

Description It is often necessary to use several families of magnetic elements with a certain ratio R of magnetic strength to perform corrections like tune adjustment (2.5.2), chromaticity correction (2.5.3) or resonance compensation (2.5.8). The *Combination of Elements* input block allows such a combination of elements. The maximum number of elements is defined by the parameter NCOM (see Appendix B.2).

Keyword COMB

Number of data lines variable

Format *e0 R1 e1 . . . Rn en*

e0 reference element which appears in the input of the processing prozedure

e1, . . . , en elements to be combined with *e0*

Rj ratio of the magnetic strength of element *ej* to that of element *e0*

2.5 Processing

This section comprises all the input blocks that do some kind of pre– or postprocessing.

2.5.1 Linear Optics Calculation

Description The linear optics calculation input block is used to make a printout of all linear parameters (magnet lengths, β and α functions, tunes, dispersion and closed orbit) in the horizontal and vertical planes at the end of each element or linear block. The number of elements or blocks can be chosen.

Keyword LINE

Number of data lines 1

Format *mode number–of–blocks*

mode 'ELEMENT' for a printout after each single element (2.2.1); 'BLOCK' for a printout after each structure block (2.2.2)

number–of–blocks (integer) The number of the blocks in the structure to which the linear parameter will be printed. If this number is set to zero or is larger than the number of blocks, the complete structure will be calculated.

Remark If the 'BLOCK' option has been used, the tunes may be wrong by a multiple of 1/2.

2.5.2 Tune Variation

Description This input block initializes a tune adjustment with zero length quadrupoles. This is normally done with two families of focussing and defocussing quadrupoles. It may be necessary however to have a fixed phase advance between certain positions in the machine. This can be done with this block by splitting the corresponding family into two subfamilies which then are adjusted to give the desired phase advance.

Keyword TUNE

Number of data lines 2 or 4

Format

- data lines 1 and 2: *name1* Qx and *name2* Qy respectively
- data lines 3 and 4, optional: *name3* ΔQ and *name4* *name5* respectively

name1, name2 Names of focussing and defocussing quadrupole families respectively (in the single element list (2.2.1.1)

Qx, Qy (floats) Horizontal and vertical tune *including* the integer part

name3 Name of the second subfamily, where the first subfamily is one of the above (*name1* or *name2*) This second subfamily replaces the elements of the first subfamily between the positions marked by *name4* and *name5.*

ΔQ Extra phase advance *including* the integer part (horizontal or vertical depending on the first subfamily) between the positions in the machine marked by *name4* and *name5*

name4, name5 Two markers in the machine for the phase advance ΔQ with the elements of the second subfamily between them

Remark The integer has to be included as the full phase advance around the machine is calculated by the program.

2.5.3 Chromaticity Correction

Description The chromaticity is corrected or adjusted to desired values with two sextupole family using this input block.

Keyword CHRO

Number of data lines 2

Format *name* ξ

name Names (in the single element list (2.2.1.2) of the two sextupole families

ξ Desired values of the chromaticity

2.5.4 Orbit Adjustment

Description Due to dipole errors in a real accelerator a closed orbit different from the beam axis is unavoidable. Even after careful adjustment one always will be left over with some random deviation of the closed orbit around the zero position. This input data block allows the simulation of such a random distributed closed orbit. A closed orbit is introduced by nonzero strengths of b_1 and a_1 components of the multipole block (2.3.1), horizontal and vertical dipole kicks (2.2.1.2) or displacements of nonlinear elements (2.2.4). The first two types can be used for adjusting the orbit. For that purpose they have to be denoted by *'COR= '*. The resulting closed orbit is recorded at positions of elements that are denoted with *'MON= '* (or position *'MON'*) in the structure input (2.2.3). The correction dipoles are then scaled properly to give the desired rms–value of the closed orbit deviation.

Keyword ORBI

Number of data lines variable but at least 1

Format

- first data line: *sigmax sigmaz*
- other data lines: *'COR= 'namec* or *'MON= 'namem*

sigmax, sigmaz Desired rms–values of the randomly distributed closed orbit

'COR= 'namec Correction element of name *namec*

'MON= 'namem Monitor for the closed orbit of name*namem*

Remarks

1. Elements can have only one extra functionality: a correction element can not be a monitor at the same time.
2. The *'COR= '* and *'MON= '* must be separated from the following name by at least one space.

2.5.5 Decoupling of Motion in the Transverse Planes

Description Skew–quadrupole components in the lattice create a linear coupling between the transverse planes of motion. A decoupling can be achieved with this block using four independent families of skew–quadrupoles, which cancel the off–diagonal parts of the transfer map. As these skew–quadrupoles also influence the tunes an adjustment of the tunes is performed at the same time.

Keyword DECO

Number of data lines 3

Format

- first data line: *name1,name2,name3,name4*
- data lines 2 and 3: *name5 Qx* and *name6 Qy* respectively

name1,2,3,4 Names of the four skew–quadrupole families

name5, name6 Names of focussing and defocussing quadrupole families respectively (in the single element list (2.2.1.1)

Qx, Qy (floats) Horizontal and vertical tune *including* the integer part

Remark A decoupling can also be achieved by compensating skew–resonances (2.5.8). The two approaches, however, are not always equivalent. In the resonance approach the zeroth harmonic is compensated, whilst a decoupling also takes into account the higher–order terms.

2.5.6 Subresonance Calculation

Description First order resonance widths of multipoles from second to ninth order are calculated following the approach of Guignard [8]. This includes resonances, which are a multiple of two lower than the order of the multipole.

Keyword SUBR

Number of data lines 1

Format *n1 n2 Qx Qy Ax Ay Ip length*

n1, n2 (integers) Lowest and highest order of the resonance

Qx, Qy Horizontal and vertical tune including the integer part

Ax, Ay Horizontal and vertical amplitudes in mm

Ip (integer) Is a switch to change the nearest distance to the resonance $e = nxQx + nyQy$. In cases of structure resonances a change of p by one unit may be useful.

- $ip = 0$: e is unchanged
- $ip = 1$: $(e \pm 1) = nxQx + nyQy - (p \pm 1)$

length Length of the accelerator in meters

2.5.7 Search for Optimum Places to Compensate Resonances

Description To be able to compensate a specific resonance one has to know how a correcting multipole affects the cosine and sine like terms of the resonance width at a given position in the ring. This input data block can be used to find best places for the compensation of up to three different resonances, by calculating the contribution to the resonance width for a variable number of positions. For each position the effect of a fixed and small change of magnetic strength on those resonance widths is tested.

Keyword SEAR

Number of data lines variable but at least 2

Format

- data line 1: *Qx Qy Ax Ay length*
- data line 2: *npos n ny1 ny2 ny3 ip1 ip2 ip3* (integers)
- data lines from 3 on: *name1, . . . , namen*

Qx, Qy Horizontal and vertical tune including the integer part

Ax, Ay Horizontal and vertical amplitudes in mm

length Length of the accelerator in m

npos Number of positions to be checked

n Order of the resonance

ny1, ny2, ny3 Define three resonances of order n via : $nxQx + nyQy = p$ with $|nx| + |ny| = n$

ip1,ip2,ip3 The distance to a resonance is changed by an integer ip for each of the three resonances: $e = nxQx + nyQy - (p + ip)$.

namei i'th name of a multipole of order n , which has to appear in the single element list (2.2.1.2)

2.5.8 Resonance Compensation

Description The input block allows the compensation of up to three different resonances of order n simultaneously the chromaticity and the tunes can be adjusted. For mostly academic interest there is also the possibility to consider subresonances which come from multipoles which are a multiple of 2 larger than the resonance order n . However it must be stated that the subresonances depend differently on the amplitude compared to resonances where the order of the resonances is the same as that of the multipoles.

Keyword RESO

Number of data lines 6

Format

- data line 1: *nr n ny1 ny2 ny3 ip1 ip2 ip3* (integers)
- data line 2: *nrs ns1 ns2 ns3* (integers)
- data line 3: *length Qx Qy Ax Ay*
- data line 4: *name1, . . . , name6*
- data line 5: *nch name7 name8*
- data line 6: *nq name9 name10 Qx0 Qy0*

nr Number of resonances (0 to 3)

n Order of the resonance, which is limited to *nrco* = 5 (see list of parameters in Appendix B.2). regular: $3 \leq n \leq nrco$; skew: $2 \leq n \leq nrco$

ny1, ny2, ny3 Define three resonances of order n via : $nxQx + nyQy = p$ with $|nx| + |ny| = n$

ip1, ip2, ip3 The distance to the resonance e can be changed by an integer value: $e = nxQx + nyQy - (p + ip)$.

nrs Number of subresonances (0 to 3)

ns1, ns2, ns3 Order of the multipole with $ns \leq 9$ and $(ns - n)/2 \in \underset{\sim}{N}$

length Length of the machine in meters

Qx, Qy Horizontal and vertical tune including the integer part

Ax, Ay Horizontal and vertical amplitudes in mm

name1, ..., name6 Names (2.2.1.2) of the correction multipoles for the first, second and third resonance

nch (integer) Switch for the chromaticity correction (0 = off, 1 = on)

name7, name8 Names (2.2.1.2) of the families of sextupoles to correct the chromaticity

nq (integer) Switch for the tune adjustment (0 = off, 1 = on)

name9, name10 Names (2.2.1.1) of the families of quadrupoles to adjust the tune

Qx0, Qy0 Desired tune values including the integer part

2.5.9 Differential Algebra

Description This input block initiates the calculation of a one turn map using the Differential Algebra package of M.Berz [9]. For this purpose a special module is needed, which has to be provided with extra memory space by shifting large common blocks above the 16 Mb limit [12]. The use of this block inhibits postprocessing. The same differential algebra tools allow a subsequent normal form analysis (see [17]), which is, for the time being, a stand–alone program.

Keyword DIFF.

Number of data lines 1 or 2

Format

- data line 1: *n nv p nsix ncor*
- data line 2: *name(1),...,name(ncor)*

n (integer) Order of the map

nv (integer) Number of the variables (2 to 6). *nv* = 2,4,6 : two– and four–dimensional transverse motion and full six–dimensional phase space respectively. *nv* = 5 : four–dimensional transverse motion plus the relative momentum deviation as a parameter.

p Precision needed by the DA package, usually set to *p* = 1e–38

nsix (integer) switch to calculate a 5x6 instead of a 6x6 map. This saves computational time and memory space, as the machine can be treated up to the cavity as five–dimensional (constant momentum). The cavity should therefore be placed at the end of the structure.

- *nsix* = 0 : 6x6 map
- *nsix* = 1 : 5x6 map

ncor (integer) Number of zero–length elements to be additional parameters besides the transverse and/or longitudinal coordinates (i.e. two–, four–, five– or six–dimensional phase space).

name(i) (char) *Ncor* names (2.2.1.2) of zero–length elements (e.g dipole kicks, quadrupole kicks, sextupoles kicks etc.).

Remark

- If a DA map is needed at some longitudinal location one just has to introduce an element denoted 'DAMAP' at that place in the structure, 'DAMAP' has also to appear as a marker (zero length, element type = 0) in the single element list (2.2.1.2). This extra map is written to file # 19.
- The method is not restricted to the number of variables which are presently used. It only needs minor changes to the program to add other parameters; for instance if the dependence on a certain magnet strength is needed, this gradient can be added as an additional parameter.

2.5.10 Postprocessing

Description It has been seen in the past that the tracking data hold a large amount of information which should be extracted for a thorough understanding of the nonlinear motion. It is therefore necessary to store the tracking data turn by turn and postprocess it after the tracking has been finished. The following quantities are calculated:

1. **Lyapunov exponent analysis** This allows to decide if the motion is of regular or chaotic nature, and, in the later case, that the particle will ultimately be lost. This is done with the following procedure:

   (a) Start the analysis where the distance in phase space of the two particles reaches its minimum.

   (b) Study the increase in a double logarithmic scale so that the slope in a regular case is always one, while a exponential increase stays exponential when we have chaos.

   (c) Average the distance in phase space to reduce local fluctuations, as we a re interested in a long range effect.

   (d) Make a weighted linear fit with an increasing number of averaged values of distance in phase space, so that an exponential increase results in a slope that is larger than one and is increasing. (The weighting stresses the importance of values at large turn numbers).

2. **Analysis of the tunes** This is done either by the averaged phase advance method leading to very precise values of the horizontal and vertical tunes. A FFT analysis is also done. With the second method one can evaluate the relative strength of resonances, rather than achieve a precise tune measurement. In both cases the nearby resonances are determined.

3. **Smear** The smear of the horizontal and vertical emittances and the sum of the emittances are calculated in case of linearly coupled and uncoupled motion.

4. **Nonlinear Invariants** A rough estimate of the nonlinear invariants are given.

5. **Plotting** The processed tracking data can be plotted in different ways:

   (a) The distance of phase space as a function of amplitude

   (b) Phase space plots

   (c) Stroboscoped phase space

   (d) FFT amplitudes

6. **Summary** The postprocessing results for a complete tracking session with varying initial parameters are summarized in a table at the end of the run.

Keyword POST

Number of data lines 4

Format

- data line 1: *comment title*
- data line 2: *na nstart nstop iw dx dy iskip imad cma1 cma2* (general parameters)
- data line 3: *Qx0 Qy0 ix iy ir dr if df* (parameters for the tune calculation)
- data line 4: *kwtype iout istop idist icoor istrob ifft nprint idafi* (integer parameters for the plotting)

na (integer) Averaging interval of the values of the distance in phase space. Typically a tenth of the total turn number should be used as this interval.

nstart, nstop (integers) Start and stop turn number for the analysis of the postprocessing.

iw (integer) Switch for the weighting of the slope calculation of the distance in phase space (0 = off, 1 = on)

dx, dy Horizontal and vertical angle interval in radians that is used to stroboscope phase space. This stroboscoping of one of the two phase space projections is done by restricting the angle in the other phase space respectively to lie inside ± *dx* or ± *dy*.

iskip (integer) This parameter allows to reduce the number of data to be processed: only each *iskip* sample of data will be used.

imad, cma1, cma2 (integer,floats) These parameters are useful when MAD data shall be analysed (*imad* set to one). To improve the Lyapunov analysis the off–momentum and the path–length difference can be scaled with *cma1* and *cma2* respectively (see also 2.6.3).

Qx0, Qy0 (floats) Values of the horizontal and vertical tune respectively (integer part) to be added to the averaged phase advance and to the Q values of the FFT analysis.

ix, iy (integers) The tunes from the average phase advance are difficult to be calculated when this phase advance is strongly changing from turn to turn and when the tune is close to 0.5, as then the phase may become negative leading to a deviation of one unit. This problem can partly be overcome by setting these switches in the following way:

- tune close to an integer: $ix,\ iy = 1$
- tune close to half an integer: $ix,\ iy = 0$

ir, dr (integer,float) For the calculated tune values from the average phase advance method and the FFT–routine the closest resonances are searched up to *ir*'th order and inside a maximum distance to the resonance *dr*, so that $nxQx + nyQy < dr$ and $nx + ny \leq ir$.

if, df (integer,float) For the FFT analysis the tune interval can be chosen with *if*. To find resonances with the FFT spectrum, all peaks below a fraction *df* of the maximum peak are accepted.

- $if = 0 : 0 \leq Q \leq 1$
- $if = 1 : 0 \leq Q \leq 0.5$

- *if* $= 2 : 0.5 \leq Q \leq 1$

kwtype Terminal type, e.g. 7878 for the Pericom graphic terminals. For details, consult the HPLOT manual [6].

iout Switch to direct the plotting output.

- *iout* $= 0$: Terminal
- *iout* $= -1$: GKS–file
- *iout* $= 1$: Terminal and GKS–file

istop Switch to stop after each plot (0 = no stop, 1 = stop after each plot).

idist, icoor, istrob, ifft Switches (0 = off, 1= on) to select the different plots. If all values are set to zero, the HBOOK/HPLOT routine will not be called.

- *idist* = 1 : plot of distance in phase space
- *icoor* = 1 : a set of plots of projections of the six–dimensional phase space and the energy E versus the turn number
- *istrob* = 1 : plot of the stroboscoped phase space projection by restricting the phase in the other phase space projection
- *ifft* = 1 : plots of the horizontal and vertical FFT spectrum

nprint Switch to stop the printing of the postprocessing output (0 = printing off, 1 = printing on).

idafi Number of datafiles to be processed (units : from 90 to (90–idafi+1)).

Remarks

1. The postprocessing can be done in two ways :
   (a) directly following a tracking run by adding this input block to the input blocks of the tracking
   (b) as a later run where the tracking parameter file (unit # 3) consists of only the *Program Version* input block 2.1.1 (using the *FREE* option) and of this input block specifying the postprocessing parameters followed by *ENDE* as usual
2. The HBOOK/HPLOT routines are only used at the start of the main program for initialization and termination. The actual plots are done in the postprocessing subroutine. The routines are activated only if at least one of the plotting parameters (*idist, icoor, istrob, ifft*) is set to one.

2.6 Initial Conditions for Tracking

Description For the study of nonlinear system the choice of initial conditions is of crucial importance. The input structure for the initial conditions was therefore organized in such a way as to allow for maximum flexibility. SIXTRACK is optimized to reach the largest possible number of turns. In order to derive the Lyapunov exponent and thereby to distinguish between regular and chaotic motion, the particle has a close by companion particle. Moreover, experience has shown that varying only the amplitude while keeping the phases constant is sufficient to understand the nonlinear dynamics, as a subsequent detailed postprocessing allows to find the dependence of the parameter of interest on these phases.

2.6.1 Tracking Parameters

Description All tracking parameters are defined with this input block, the initial coordinates are generally set here, too. A fine tuning of the initial condition is done with Initial Coordinates block (2.6.2) and the parameters for the synchrotron oscillation are given in block (2.6.3)

Keyword TRAC

Number of data lines 3

Format

- data line 1: *numf numb na astart aend nvar nmo*
- data line 2: *icplx icply idcon irew iclo6* (integers)
- data line 3: *nbot nramp nob nor not ncop ntwin* (integers)

numf (integer) Number of turns in the forward direction

numb (integer) Number of turns in the backward direction

na (integer) Number of amplitude variations

astart, aend (floats) Start and end amplitude in the horizontal phase space plane for the amplitude variations. The vertical amplitude is calculated using the ratio between the horizontal and vertical emittance set in the *Initial Coordinates* block (2.6.2), where the initial phase in phase space are also set.

nvar (integer) Switch for the type of amplitude variation. In case $na = 1$ the amplitude nstart is used.

- $nvar = 0$: amplitudes are varied between the amplitudes *astart* and *aend* with equal increments:
$$delta = (aend - astart)/(na - 1)$$
- $nvar = 1$: amplitude variation to find an estimate for the short term dynamic aperture. The amplitude is increased or decremented corresponding to stable motion or particle loss respectively. The change of amplitude is reduced each iteration $i \leq (na - 1)$ to:
$$delta = (aend - astart)/2^i$$

nmo (integer) Number of variations of the relative momentum deviation. The maximum value of the relative momentum deviation $\frac{\Delta p}{p}$ is taken from that of the first particle in the *Initial Coordinates* block (2.6.2). The variation will be between $\pm\frac{\Delta p}{p}(max)$ in steps of $\frac{\Delta p}{p}(max)$ / (*nmo–1*).

icplx, icply A tracking where one of the transversal motion planes shall be ignored is only possible when all coupling terms are switched off. The part of the coupling that is due to closed orbit and other effects can be turned off with these switches.

- *icplx, icply* $= 1$: coupling on
- *icplx, icply* $= 0$: coupling to the horizontal and vertical motion plane respectively switched off

idcon Usually the closed orbit is added to the initial coordinates. This can be turned off using *idcon*, for instance when a run is to be prolonged.

- *idcon* = 0 : closed orbit added
- *idcon* = 1 : initial coordinates unchanged
- *idcon* = 2 : prolongation of a run, taken the initial coordinates from unit # 13

irew To reduce the amount of tracking data after each amplitude and relative momentum deviation iteration the binary output units 90 and lower (see Appendix C) are rewound. This is always done when the postprocessing is activated (2.5.10). For certain applications it may be useful to store all data. The switch *irew* allows for that.

- *irew* = 0 : unit 90 (and lower) rewound
- *irew* = 1 : all data on unit 90 (and lower)

iclo6 This switch allows to calculate the 6d closed orbit using the differential algebra package.

- *iclo6* = 0 : switched off
- *iclo6* = 1 : calculated
- *iclo6* = 2 : calculated and added to the initial coordinates (2.6.2).

nbot Number of turns at flat bottom, useful for energy ramping.

nramp Number of turns for the energy ramping. *numf–nramp* gives the number of turns on the flat top. For constant energy with *nbot* = *nramp* = 0 the particles are considered to be on the flat top.

nob Every *nob*'th turn the coordinates will be written on unit 90 (and lower) in the flat bottom part of the tracking.

nor Every *nor*'th turn the coordinates in the ramping region will be written on unit 90 (and lower).

not Every *not*'th turn at the flat top a write out of the coordinates on unit 90 (and lower) will occur. For constant energy this number controls the amount of data on unit 90 (and lower), as the particles are considered on the flat top.

ncop In cases of very long runs it is sometimes useful to save all coordinates for a prolongation of a run after a possible crash of the computer. Every *ncop*'th turn the coordinates are written to unit 6.

ntwin For the analysis of the Lyapunov exponent it is usually sufficient to store the calculated distance of phase space together with the coordinate of the first particle (*ntwin* set to one). If storage space is no problem, one can store the coordinates of both particles (*ntwin* set to two). This allows a subsequent refined Lyapunov analysis using differential–algebra and Lie–algebra techniques ([19]).

Remarks

1. This input data block is usually combined with the *Initial Coordinates* input block (2.6.2) to allow a flexible choice of the initial coordinates for the tracking.
2. For a prolongation of a run the following parameters have to be set :

- in this input block : idcon = 1
- in the *Initial coordinates* input block :
  (a) npa = 0
  (b) take the end coordinates of the previous run as the initial coordinates (including all digits) for the new run.

3. A new feature is installed for a prolongation of a run (it works for the scalar and vector version) by using *idcon = 2* and reading the initial data from unit # 13. The end coordinates are now written on unit # 12 after each run. Intermediate coordinates are also written on unit # 12 in case the turnnumber *ncop* is exceeded in the run. The user takes responsability to transfer the required data from unit # 12 to unit # 13 if a prolongation is requested.

2.6.2 Initial Coordinates

Description The *Initial Coordinates* input block is meant to manipulate how the initial coordinates are organized, which are generally set in the tracking parameter block (2.6.1). Number of particles, initial phase, ratio of the horizontal and vertical emittances and increments of 2 × 6 coordinates of the two particles, the reference energy and the starting energy for the two particles.

Keyword INIT

Number of data lines 16

Format

- first data line: *npa ph0 phd re*
- data lines 2 to 16: *15 initial coordinates in table* 2.4

npa (integer) Number of particles

- *npa* = 0 : Amplitude values of tracking parameter block (2.6.1) are ignored and coordinates of data line 2–16 are taken. *npa* is set internally to 2 for tracking with two particles. This is necessary in case a run is to be prolonged.
- *npa* = 1 : Tracking of one particle, twin particle ignored
- *npa* = 2 : Tracking the two twin particles

ph0 Starting phase of the initial coordinate in the horizontal and vertical phase space projections

phd Phase difference between first and second particles

re Emittance ratio of horizontal and vertical motion

Table 2.4: Initial Coordinates of the 2 Particles

data line	contents
2	x coordinate of particle 1
3	x' coordinate of particle 1
4	y coordinate of particle 1
5	y' coordinate of particle 1
6	path length difference σ of particle 1
7	$\frac{\Delta p}{p}$ of particle 1
8	x coordinate of particle 2
9	x' coordinate of particle 2
10	y coordinate of particle 2
11	y' coordinate of particle 2
12	path length difference σ of particle 2
13	$\frac{\Delta p}{p}$ of particle 2
14	energy of the reference particle
15	energy of particle 1
16	energy of particle 2

Remarks

1. These 15 coordinates are taken as the initial coordinates if *npa* is set to zero (see above). If *npa* is 1 or 2 these coordinates are added to the initial coordinates generally defined in the tracking parameter block (2.6.1). This procedure seems complicated but it allows freely to define the initial difference between the two twin particles. It also allows in case a tracking run should be prolonged to continue with precisely the same coordinates. This is important as small difference may lead to largely different results.
2. The reference particle is the particle in the centre of the bucket which performs no synchrotron oscillations.
3. The energy of the first and second particles is given explicitly, again to make possible a continuation that leads precisely to the same results as if the run would not have been interrupted.
4. There is a refined way of prolonging a run, see the *Tracking Parameters* input block (2.6.1).

2.6.3 Synchrotron Oscillation

Description The parameters needed for treating the synchrotron oscillation in a symplectic manner are given in the *Synchrotron Oscillation* input block.

Keyword SYNC

Number of data lines 2

Format

- first data line: *harm comp v0 phas length pma itran*
- second data line: *dpscor sigcor*

harm Harmonic number

comp Momentum compaction factor, used here only to calculate the linear synchrotron tune Q_S .

v0 Circumference voltage in megavolts

phas Acceleration phase in degrees

length Length of the accelerator in meters

pma rest mass of the particle in MeV/c^2

itran (integer) Transition energy switch

- $itran = 0$ for no synchrotron oscillation (energy ramping still possible)
- $itran = 1$ for above transition energy
- $itran = -1$ for below transition energy

dpscor Scaling factor for relative momentum deviation

sigcor Scaling of the path length difference σ

In order to evaluate in a simple manner the Lyapunov exponent from the evolution in time of the distance in phase space, the components of the distance must all be of roughly the same order of magnitude. This is not true for the coordinates of the synchrotron oscillation: the relative momentum deviations tend to be small while the path length difference usually reaches large values. The two scaling factors can be used to adjust for that.

Conclusions and Acknowledgements

Programs with large input structures like SIXTRACK tend to be far from perfect, even though a cumbersome chase for program bugs and a lot of polishing on the input structure has been performed.

As stated above, the input structure of RACETRACK has been kept to keep its flexibility. It allows easy addition (in some hours) of further input blocks like for instance beam–beam kicks or higher order chromaticity correction a la PATRAC [18].

Finally I hope to receive plenty of comments and suggestions to further improve the program.

Acknowledgement

I would like to thank my colleagues at DESY and CERN to help to find nasty bugs and for a thorough check of the program. I would like to thank Mikko Vaenttinen who helped to vectorize the program. He also did most of the typing of the manuscript. Moreover i want to express my gratitude to F. Zimmermann from DESY who helped to finish the differential–algebra part in endless night sessions.

Appendix A. List of Keywords

Table A.1: List of Keywords

#	Keyword	Inputdatablock		Short Description	§	Page
		Title	**# of Datalines**			
1	BLOC	Blockdefinition	variable + 1	Blocks of linear Elements	2.2.2	8
2	BLOCK			Linear Parameters for each Structure Element	2.5.1	14
3	CAV			Cavity in the Structure Input Block	2.2.3	9
4	CHRO	Chromaticity Correction	2	Correcting Chromaticity with Sextupoles	2.5.3	15
5	'COR= '			Specifies an Orbit Correction Dipole	2.5.4	15
6	COMB	Combination of Elements	variable	Combining different Elements for a Correction	2.4.3	13
7	COMM	Comment Line	1	Additional Comments	2.1.3	4
8	DAMAP			Location for a Printout of a DA map	2.5.9	19
9	DECO	Decoupling	3	Compensation of linear Coupling	2.5.5	16
10	DIFF	Differential Algebra	1	Calculating a One–turn Map with Differential Algebra	2.5.9	19
11	DISP	Displacement of Elements	variable	Displacing Nonlinear Elements	2.2.4	9
12	EL			Elliptical Aperture Limitation	2.3.2	11
13	ELEMENT			Linear Parameters after each Single Element	2.5.1	14
14	ENDE			End of SIXTRACK Input Structure		
15	FLUC	Random Fluctuation Starting Number	1	Seed for the Random Generator	2.4.1	12
16	FREE	1^{st} Program Version	0	Free Format Input from one File	2.1.1	4
17	GEOM	2^{nd} Program Version	0	Input of Machine Geometry in extra File	2.1.1	4
18	GO			Start of Tracking in the Structure Input	2.2.3	9
19	INIT	Initial Coordinates	16	Setting up of the initial Coordinates	2.6.2	25
20	ITER	Iteration Errors	4	# of Iterations and Precision for Correction Routines	2.1.4	5
21	LIMI	Aperture Limitation	variable	Collimators that stop the Program when being hit	2.3.2	11
22	LINE	Linear Optics	1	Calculation of all linear Parameters of the Accelerator	2.5.1	14

Table A.2: List of Keywords (continued)

#	Keyword	Inputdatablock		Short Description	§	Page
		Title	# of Datalines			
23	MON			Orbit Observation Position in the Structure Input	2.2.3	9
24	'MON= '			Orbit Observation position in the Orbit Adjustment Input	2.5.4	15
25	MULT	Multipole skew Coefficients	max. 11	Multipole Coefficients normal and up to 10^{th} order	2.3.2	11
				Combination of different Multipoles in the ORGA Input Block	2.4.2	12
26	NEXT			Last Line of each Input Data Block	2.5.4	15
27	ORBI	Orbit Adjustment	variable	Adjusting Orbit to desired Sigma Values	2.5.4	15
28	ORGA	Organization of Random Numbers	variable + 1	Arranging Random Errors and Multipole sets	2.4.2	12
29	POST	Postprocessing	3	Postprocessing of the Tracking Data	2.5.10	20
30	PRIN	Printout Selection	0	Initiates the Printing of the Input Data	2.1.2	4
31	RE			Rectangular Aperture Limitation	2.3.2	11
32	RESO	Resonance Compensation	6	Compensation of up to 3 different Resonances	2.5.8	18
33	RIPP	Power Supply Ripple	variable	Invokes a sinusoidal Tune Variation	2.3.3	11
34	SEAR	Search for Resonance Compensation Positions	variable	Evaluating longitudinal Positions for a Resonance Compensation	2.5.7	17
35	SING	Single Elements	variable	Magnet Parameters of Single Elements	2.2.1	6
36	STRU	Structure Input	variable	Structure of linear Blocks and nonlinear Elements	2.2.3	9
37	SUBR	Subresonance Calculation	1	Calculation of 1^{th} Order Resonances up to 9^{th} Multipole Order	2.5.6	17
38	SYNC	Synchrotron Oscillations	2	Parameters concerning Synchrotrons Oscillation	2.6.3	26
39	TRAC	Tracking Parameters	3	All major Tracking Parameters for the transversal Motion Plane	2.6.1	23
40	TUNE	Tune Variation	2 or 4	Adjusting the horizontal and vertical Tunes	2.5.2	14

Appendix B. List of Default Values

B.1 Default Tracking Parameters

Some of the parameters for tracking are set to non–zero values. This is done for instance to avoid as much as possible program errors such as division by zero due to an erroneous input. The default values for the *Iteration Errors* (2.1.4) see table 2.1.

Table B.1: Default Tracking Parameters

#	**Description**	**Value**	§	**Page**
1	General Aperture Limitations (horizontal and vertical)	1000 mm	2.3.2	11
2	Starting in the Accelerator Structure at Element Number	1	2.2.3	9
3	Number of Turns in the forward Direction	1	2.6.1	23
4	Initial horizontal Amplitude	.001 mm		
5	Horizontal and vertical Phase Space Coupling Switches on	1		
6	Flat Bottom, Ramping and Flat Top Printout after Turn Number	1		
7	Printout of Coordinates (file 6) after Turn Number	10000		
8	Kinetic Energy [MeV] of the Reference Particle	10^{-6}	2.6.2	25
9	Harmonic Number	1	2.6.3	26
10	Momentum Compaction Factor	.001		
11	Length of the Machine	1 km		
12	Mass of the Particle (Proton)	938.2796 MeV/c^2		
13	Momentum Correction Factor for Distance in Phase Space	1		
14	Pathlength Correction Factor for Distance in Phase Space	1		
15	Averaging Turn Interval for Postprocessing	1	2.5.10	20

B.2 Default Size Parameters

For large machines the arrays holding the machine parameters might have to be increased. The size of each of the dimensions of the arrays is therefore defined as a parameter. The default values are adjusted as to stay below 8 Mb in the scalar version and below 16 Mb in the vectorized version.

Table B.2: Default Size Parameters

#	**Description**	**Value**	**Name**	§	**Page**
1	Maximum Number of Coordinates used in the Correction Routines	6	MPA		
2	Number of Single Elements	320	NELE	2.2.1	6
3	Number of Blocks of linear Elements	100	NBLO	2.2.2	8
4	Number of linear Elements per Block	100	NELB		
5	Total Number of Elements in the Structure	6000	NBLZ	2.2.3	9
6	Number of Accelerator Superperiods	16	NPER		
7	Total Number of Random Values	80000	NZFZ	2.4.1	12
8	Number of Random Values for the basic Set of nonlinear Elements	70000	NRAN		
9	Number of Random Values for inserted nonlinear Elements	10000		2.4.2	12
10	Number of Random Values for each Inserted Non–Linear Element Number of nonlinear Elements that can be inserted	500 20	MRAN		
11	Limit Number of Particles for Vectorization	64	NPART		
12	Number of different Random Distributions for Vectorization	5	NMAC	2.4.1	12
13	Maximum Number of Elements for combined Tasks	100	NCOM	2.4.3	13
14	Maximum Resonance Compensation Order	5	NRCO	2.4.3	13
15	Total Number of Data for Processing	20000	NPOS	2.5.10	20
16	Number of Intervals for Calculation of Lyapunov–Exponents	100	NLYA		
17	Number of Data for FFT–Analysis	16384	NFFT		
18	Number of Intervals for Calculation of Invariants	1000	NINV		
19	Number of Data for Plotting	20000	NPLO		

Appendix C. Input and Output Files

The program uses a couple of files for its input and output prozedures.

Table C.1: List of Input and Output Files

File Unit	Input	Output	File Type	Contents
2	X		Ascii	Geometry and strength Parameters (but see 2.1.1)
3	X		Ascii	Tracking Parameters (but see 2.1.1)
6		X	Ascii	Input Parameters and Analysis of Data
10	X	X	Ascii	Summary of Postprocessing (auxiliary)
11	X	X	Binary	Instead of internal read on the Cray (auxiliary)
12		X	Ascii	End Coordinates of both Particles
13	X		Ascii	Start Coordinates for a Prolongation
14		X	Ascii	Horizontal FFT Spectrum for detailed Analysis; Format: (2 X F10.6)
15		X	Ascii	Vertical FFT Spectrum for detailed Analysis; Format: (2 X F10.6)
18		X	Ascii	One–Turn Map with Differential Algebra (2.5.9)
20		X	Metafile	GKS–Metafile of selected Plots (2.5.10)
90–k		X	Binary	Tracking Data 0 <= k <= 31

Appendix D. Data Structure of Binary Data–Files

A common data structure for the programs MAD and SIXTRACK is agreed on. Besides some minor differences this allows a straightforward postprocessing of data from either program. Each binary data–file has a header which holds a description of the run with comments, tracking parameters and 50 additional parameters for future purposes, six of which are already specified in SIXTRACK.

Table D.1: Header of the Binary Data–Files

Data Type	Bytes	Description
Character	80	General title of the run
Character	80	Additional title
Character	8	Date
Character	8	Time
Character	8	Program name
Integer	4	First particle in the file
Integer	4	Last particle in the file
Integer	4	Total number of particles
Integer	4	Code for dimensionality of phase space 1,2,4 are hor., vert. and longitudinal respectively
Integer	4	Projected number of turns
Float	8	Horizontal Tune
Float	8	Vertical Tune
Float	8	Longitudinal Tune
Float	6 * 8	Dispersion vector
Float	36 * 8	Six–dimensional transfer map
—– 50 additional parameters —–		
Float	8	Maximum number of different seeds (IBM set to one; Cray up to 5)
Float	8	Actual seed number
Float	8	Starting value of the seed
Float	8	Number of turns in the reverse direction (IBM only)
Float	8	Correction–factor for the Lyapunov (path–length difference)
Float	8	Correction–factor for the Lyapunov (relative momentum deviation)
Float	44 * 8	Dummies

Following this header the tracking data are written in n samples of nine numbers preceded by the turn number. In the MAD format the number of samples n is not restricted, whilst SIXTRACK writes only up to two samples for the two particles for the Lyapunov–exponent method. Up to 64 particles (two per file) can be treated in the vectorized version of SIXTRACK.

Table D.2: Format of the Binary Data

Data Type	**Bytes**	**Description**
Integer	4	Turn number
— One or two samples of 9 values are following —		
Integer	4	Particle number
Float	8	Angular distance in phase space (<= 1)
Float	8	x (mm)
Float	8	x’ (mrad)
Float	8	z (mm)
Float	8	z’ (mrad)
Float	8	Path–length (mm)
Float	8	Relative momentum deviation
Float	8	Energy (MeV)

Some of the postprocessing data are written in ascii-format on file # 10. This can be used for instance for plotting purposes. Each time the post- processing routine is called 60 double precision numbers (some of them still dummy) are added to the file.

Table D.3: Postprocessing Data

#	**Description**
1	Turn number
2	Turn number of a particle loss
3	Horizontal tune
4	Vertical tune
5	Horizontal betafunction
6	Vertical betafunction
7	Horizontal amplitude
8	Vertical amplitude
9	Relative momentum deviation
10	Final distance in phase space
11	Final slope of # 10
12	Horizontal detuning
13	Spread of horizontal detuning
14	Vertical detuning
15	Spread of vertical detuning
16	Horizontal factor to nearest resonance
17	Vertical factor to nearest resonance
18	Order of nearest resonance
19	Horizontal smear
20	Vertical smear
21	Transverse smear
22	Dummy
23	Dummy
24	Starting seed for random generator
25	Synchrotron tune
26-60	Dummy

Appendix E. Tracking Examples

A simple tracking example is shown with its input file (E.1), its output file (E.2) and some corresponding plots in (E.3).

E.1 Input Example

For the description of the different input blocks see chapter 2.

```
FREE FORMAT   TITLE: EXAMPLE
PRINTOUT OF INPUT PARAMETERS--------------------------------------------
NEXT--------------------------------------------------------------------
SINGLE ELEMENTS---------------------------------------------------------
B      0  0.0000000     0.000000     50.0000
QD2    2  0.0000000     0.009536     0.77000
QF2    2  0.0000000    -0.009536     0.77000
MU    11  1.0000000     1.000000     0.00000
SEX    3  0.0500000     0.000000     0.00000
NEXT--------------------------------------------------------------------
BLOCK DEFINITIONS-------------------------------------------------------
  1  1
 B1  QD2 B QF2
 B2  QF2 B QD2
NEXT--------------------------------------------------------------------
STRUCTURE INPUT---------------------------------------------------------
          MU  B1 SEX B2
NEXT--------------------------------------------------------------------
MULTIPOLE COEFFICIENTS--------------------------------------------------
MU          10.0    3.5765
        0.0000     0.0000    0.0000    0.0000
        0.0000     0.0000    0.0000    0.0000
        0.405E-3   0.0000    0.0000    0.0000
        -.5E-5     0.0000    0.0000    0.0000
        -.56E-4    0.0000    0.0000    0.0000
        0.0000     0.0000    0.0000    0.0000
        0.3E-5     0.0000    0.0000    0.0000
        0.0000     0.0000    0.0000    0.0000
        -.1E-5     0.0000    0.0000    0.0000
NEXT--------------------------------------------------------------------
TRACKING PARAMETERS-----------------------------------------------------
 10000   0   2    11.0    11.5   0    1
   1        1        0   0
      0         0         1         1         1    50000 2
NEXT--------------------------------------------------------------------
INITIAL COORDINATES-----------------------------------------------------
        2   0.       0.        1.
                      0.
                      0.
                      0.
                      0.
                      0.
                      0.
                      0.
                      0.000001
                      0.
                      0.
                      0.
                      0.
                 450000.
                 450000.
                 450000.
NEXT--------------------------------------------------------------------
ITERATION-ACCURACY------------------------------------------------------
       50 1D-14 1D-15
    10 1D-10 1D-10
    10 1D-5 1D-6
    1D-8  1D-12 1D-10
NEXT--------------------------------------------------------------------
POSTPROCESSING----------------------------------------------------------
EXAMPLE
  1000    0 0  1    .08 .08 1
   0.    0.  1  1  20  .005   1 .10
  7878   1   0    1 1 1 1
NEXT
ENDE====================================================================
```

E.2 Output Example

The preprocessing part is shown first.

```
-----------------------------------------------------------------------------------------------------------------------------------

         OOOOOOOOOOOOOOOOOOOOO
         OO                 OO
         OO  PREPROCESSING  OO
         OO                 OO
         OOOOOOOOOOOOOOOOOOOOO

-----------------------------------------------------------------------------------------------------------------------------------

    ---- ENTRY CLORB ----/DPP= 0.00000 /CLOX/   0.00000   0.00000 /CLOZ/   0.00000   0.00000 /ITERAT.=  2/ ACCURACY= 0.000000D+00
    ---- ENTRY CLORB ----/DPP= 0.00000 /CLOX/   0.00000   0.00000 /CLOZ/   0.00000   0.00000 /ITERAT.=  2/ ACCURACY= 0.000000D+00
    ---- ENTRY ORBIT ----/NO MONITORS SPECIFIED
    ---- ENTRY CLORB ----/DPP= 0.00000 /CLOX/   0.00000   0.00000 /CLOZ/   0.00000   0.00000 /ITERAT.=  2/ ACCURACY= 0.000000D+00
    ---- ENTRY CLORB ----/DPP= 0.00000 /CLOX/   0.00000   0.00000 /CLOZ/   0.00000   0.00000 /ITERAT.=  2/ ACCURACY= 0.000000D+00

-----------------------------------------------------------------------------------------------------------------------------------

         REL. MOMENTUM DEVIATION=   0.00000
       ========================================

         TRACKING FOR CONSTANT MOMENTUM DEVIATION

               ------ NO ACCELERATION ------

                   TUNE           CLO            CLOP           BET0          ALF0
           X    0.1222386779 0.00000000E+00 0.00000000E+00   92.957545511   0.0000000000
           Z    0.1222386779 0.00000000E+00 0.00000000E+00  203.581213058   0.0000000000
```

Followed by the initial coordinates and the final coordinates for a regular (right side) and chaotic (left side) case.

```
     ---- INITIAL COORD. OF TWIN-TRAJECTORIES

            11.500000000000000000000000000000000               11.000000000000000000000000000000000
            -0.000000000000000004942488456867054               -0.000000000000000004727597654394574
            17.018620942597692600000000000000000               16.278680901615182100000000000000000
             0.000000000000000001669894918209964                0.000000000000000001597290791331270
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
            11.500000000000000000000000000000000               11.000000000000000000000000000000000
             0.000000999999999995057306000000000                0.000000999999999995272240000000000
            17.018620942597692600000000000000000               16.278680901615182100000000000000000
             0.000000000000000001669894918209964                0.000000000000000001597290791331270
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
        450000.000000000000000000000000000000000           450000.000000000000000000000000000000000
        449999.999999999971000000000000000000000           449999.999999999971000000000000000000000
        449999.999999999971000000000000000000000           449999.999999999971000000000000000000000

-------------------------------------------------------------------------------------------------------------

          OOOOOOOOOOOOOOOO
          OO            OO
          OO  TRACKING  OO
          OO            OO
          OOOOOOOOOOOOOOOO

-------------------------------------------------------------------------------------------------------------

     NUMBER OF REVOLUTION       10000

            -2.487686155718602960000000000000000                6.413505251156681690000000000000000
             0.075756093328203377000000000000000               -0.089292907637211219000000000000000
           -15.969507084566773900000000000000000               -6.601359419085703450000000000000000
             0.081264812099892794200000000000000                0.081356939244856263400000000000000
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
            10.716992116821544100000000000000000                6.407742545437783880000000000000000
            -0.079677572866025711500000000000000               -0.089306143607324223200000000000000
             5.973841771908771080000000000000000               -6.608681175154261120000000000000000
            -0.075407708725221220300000000000000                0.081353966395825996800000000000000
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
             0.000000000000000000000000000000000                0.000000000000000000000000000000000
        450000.000000000000000000000000000000000           450000.000000000000000000000000000000000
        449999.999999999971000000000000000000000           449999.999999999971000000000000000000000
        449999.999999999971000000000000000000000           449999.999999999971000000000000000000000

          ******************************
          **** ALL PARTICLES STABLE ****
          ******************************
```

Finally part of the postprocessing for the two particles are shown (chaotic on the left and regular on the right respectively) and a summary of the postprocessing is given.

```
-----------------------------------------------------------------------------------------------------------------------------

          OOOOOOOOOOOOOOOOOOOOOO
          OO                  OO
          OO  POSTPROCESSING  OO
          OO                  OO
          OOOOOOOOOOOOOOOOOOOOOO

-----------------------------------------------------------------------------------------------------------------------------

          ANALYSING THE INCREASE OF THE DISTANCE IN PHASE-SPACE
          -----------------------------------------------------

 TURNS        DISTANCE           SLOPE        RESIDUAL       TURNS        DISTANCE           SLOPE        RESIDUAL
-------------------------------------------------------------------------------------------------------------------------
  2000    0.2253779764D-03    0.4554898739    0.0000000000      2000    0.9092427661D-04    0.3871969581    0.0000000000
  3000    0.1081182799D-02    1.3754730225    0.3947801590      3000    0.2666317995D-03    0.9350422025    0.1414830685
  4000    0.6399160375D-02    3.0341444016    2.1449279785      4000    0.3394974411D-03    0.9019818306    0.0706214905
  5000    0.4981834333D-02    2.2218360901    2.3199357986      5000    0.5020486718D-03    1.0834455490    0.0927543640
  6000    0.8028940085D-01    4.1418428421   10.5150909424      6000    0.5503330639D-03    0.9862068892    0.0943765640
  7000    0.3407768847D+00    6.2231464386   25.0305175781      7000    0.7196859601D-03    1.0560836792    0.0917263031
  8000    0.4788764947D+00    6.7520313263   22.5305938721      8000    0.7402482154D-03    0.9828781486    0.1097971201
  9000    0.4507363285D+00    6.2743434906   21.2371520996      9000    0.9506629146D-03    1.0507984161    0.1310729980
 10000    0.6438836450D+00    5.8023786545   21.1673889160     10000    0.9737567472D-03    1.0118942261    0.1297397614
--------------------------------------------------------------------------------------------------------------------------


      AVERAGED PHASE-ADVANCE
      ----------------------

 X-PHASE :    0.1175695955   +/_   0.0003315228                  X-PHASE :    0.1189145300   +/_   0.0001260177
 Z-PHASE :    0.1196627812   +/_   0.0001993022                  Z-PHASE :    0.1211689637   +/_   0.0001953892

 S-PHASE :    0.0000000000   +/_   0.0000000000                  S-PHASE :    0.0000000000   +/_   0.0000000000

START-QX :   0.1222386779   CHANGE IN X : -.4669082376D-02      START-QX :   0.1222386779   CHANGE IN X : -.3324147915D-02
START-QZ :   0.1222386779   CHANGE IN Z : -.2575896685D-02      START-QZ :   0.1222386779   CHANGE IN Z : -.1069714140D-02

      THE AVERAGED PHASE-ADVANCES ARE CLOSER THEN 0.5000D-02 TO THE FOLLOWING RESONANCES UP TO  20 ORDER
      --------------------------------------------------------------------------------------------------

      NX * QX   +   NZ * QZ   -      P      =      DELTA          NX * QX   +   NZ * QZ   -      P      =      DELTA
      ----------------------------------------------------        ----------------------------------------------------
         1           -1           0.0         -.2093D-02           1           -1           0.0         -.2254D-02
         2           -2           0.0         -.4186D-02           2           -2           0.0         -.4509D-02
        14            3           2.0         0.4963D-02
        15            2           2.0         0.2869D-02
        16            1           2.0         0.7763D-03
        17            0           2.0         -.1317D-02
        18           -1           2.0         -.3410D-02




-----------------------------------------------------------------------------------------------------------------------------
         SUMMARY OF THE POSTPROCESSING

-----------------------------------------------------------------------------------------------------------------------------
|        |           |           |            |          |NORMALIZED| SLOPE  |             |          |                     |
|  TURN  |   LINEAR  |   BETA-   | AMPLITUDES | MOMENTUM |PHASESPACE| OF THE |  NONLINEAR  |  NEAREST |       SMEAR OF      |
| NUMBER |   TUNES   | FUNCTIONS |            | DEVIATION| DISTANCE |DISTANCE|  DETUNING   | RESONANCE|    THE EMITTANCES   |
-----------------------------------------------------------------------------------------------------------------------------
|        |           |     [M]   |     [MM]   |          |          |        |             |     |ORD.|    [%]  |      [%]  |
-----------------------------------------------------------------------------------------------------------------------------
|   10000|X   0.12224|X   92.9575|X  11.000000|0.0000D+00|0.9738D-03|  1.0119|X -.33241D-02|X   1|  2 |X  26.423|X+Z   5.664|
|        |Z   0.12224|Z  203.5812|Z  16.278681|          |          |        |+/- 0.126D-03|Z  -1|    |Z  34.450|           |
|        |QS 0.000000|           |            |          |          |        |Z -.10697D-02|     |    |         |           |
|        |           |           |            |          |          |        |+/- 0.195D-03|     |    |         |           |
-----------------------------------------------------------------------------------------------------------------------------
|   10000|X   0.12224|X   92.9575|X  11.500000|0.0000D+00|0.6439D+00|  5.8024|X -.46691D-02|X  16| 17 |X  45.228|X+Z   6.683|
|        |Z   0.12224|Z  203.5812|Z  17.018621|          |          |        |+/- 0.332D-03|Z   1|    |Z  33.745|           |
|        |QS 0.000000|           |            |          |          |        |Z -.25759D-02|     |    |         |           |
|        |           |           |            |          |          |        |+/- 0.199D-03|     |    |         |           |
-----------------------------------------------------------------------------------------------------------------------------
```

E.3 Plot Example

In figure E.1 a typical example of the evolution of the distance in phase space is shown of a regular and chaotic particle. Figure E.2 and figure E.3 show the corresponding horizontal phase space and the physical phase space projections respectively. An example of the stroboscoped phase space is shown in figure E.4, where the motion in the chaotic case is beyond a 'separatrix' in the four–dimensional phase space. Even in the FFT (figure E.5) one can see the effect of chaotic behavior: it leads to a widening of the lines of the spectrum.

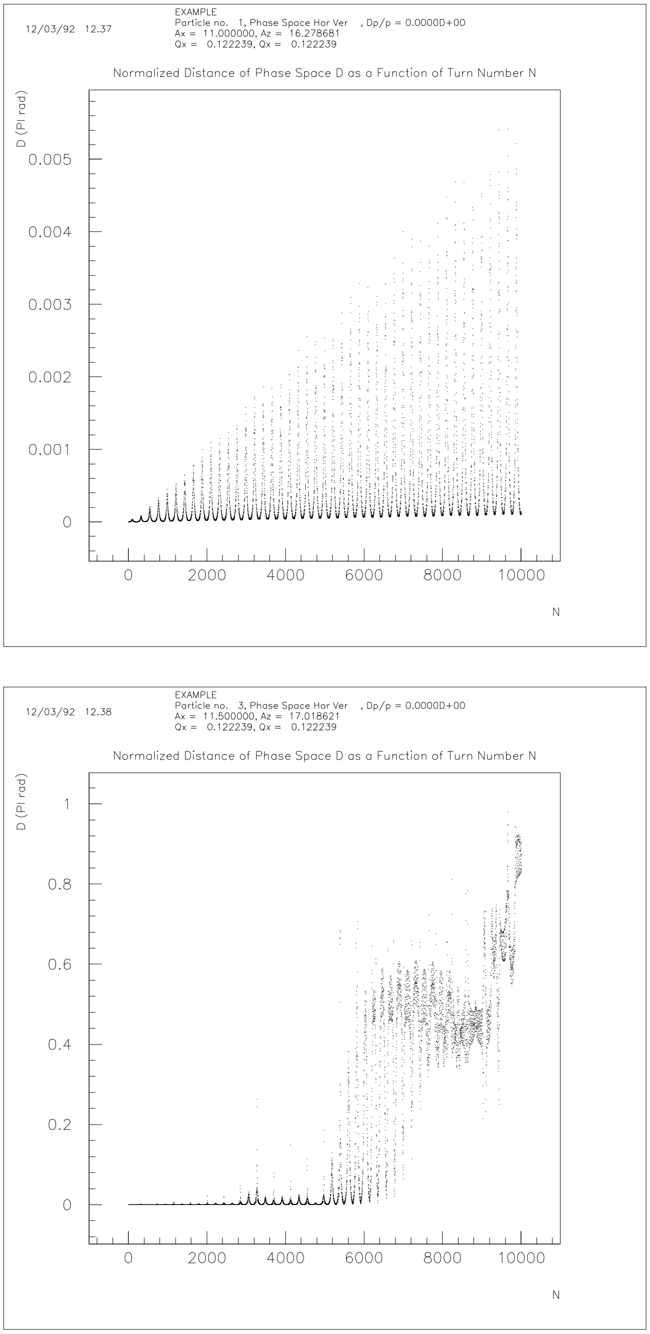

Figure E.1: Evolution of the Distance of Phase Space for Regular (upper part) and Chaotic (lower part) Motion

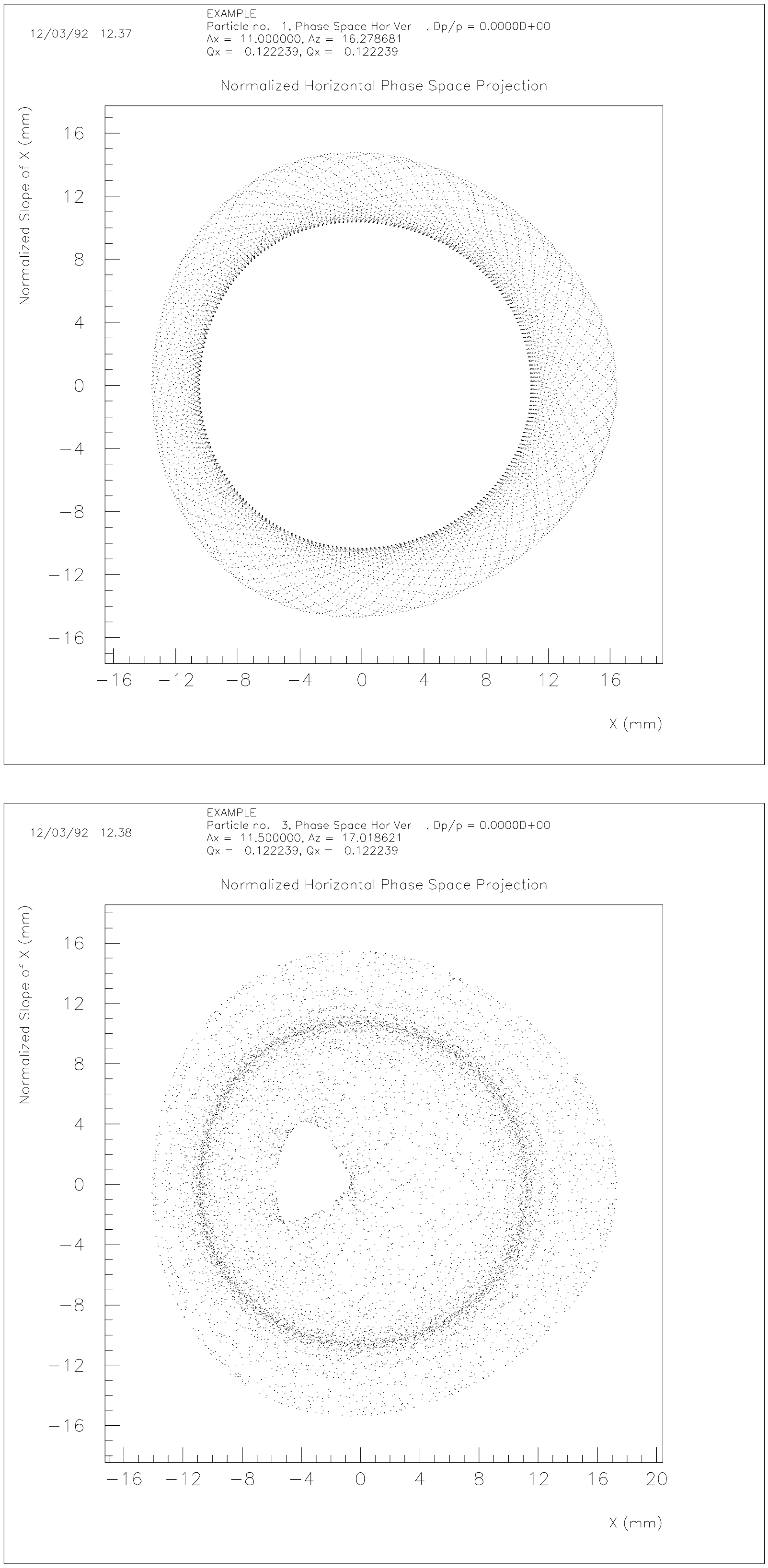

Figure E.2: Horizontal Phase Space Projections for the Regular (upper part) and the Chaotic (lower part) Cases

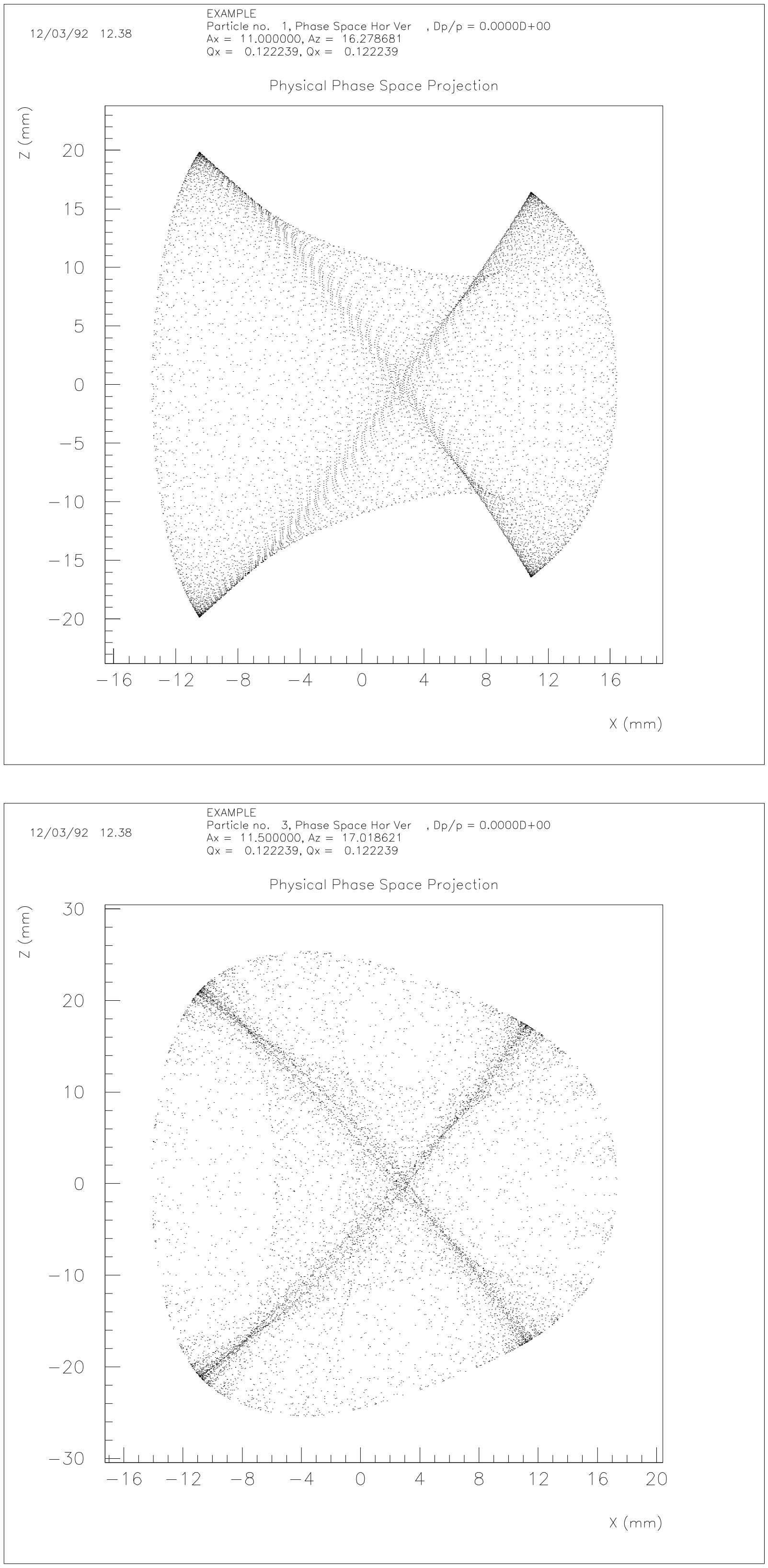

Figure E.3: Physical Phase Space Projections for the Regular (upper part) and the Chaotic (lower part) Cases

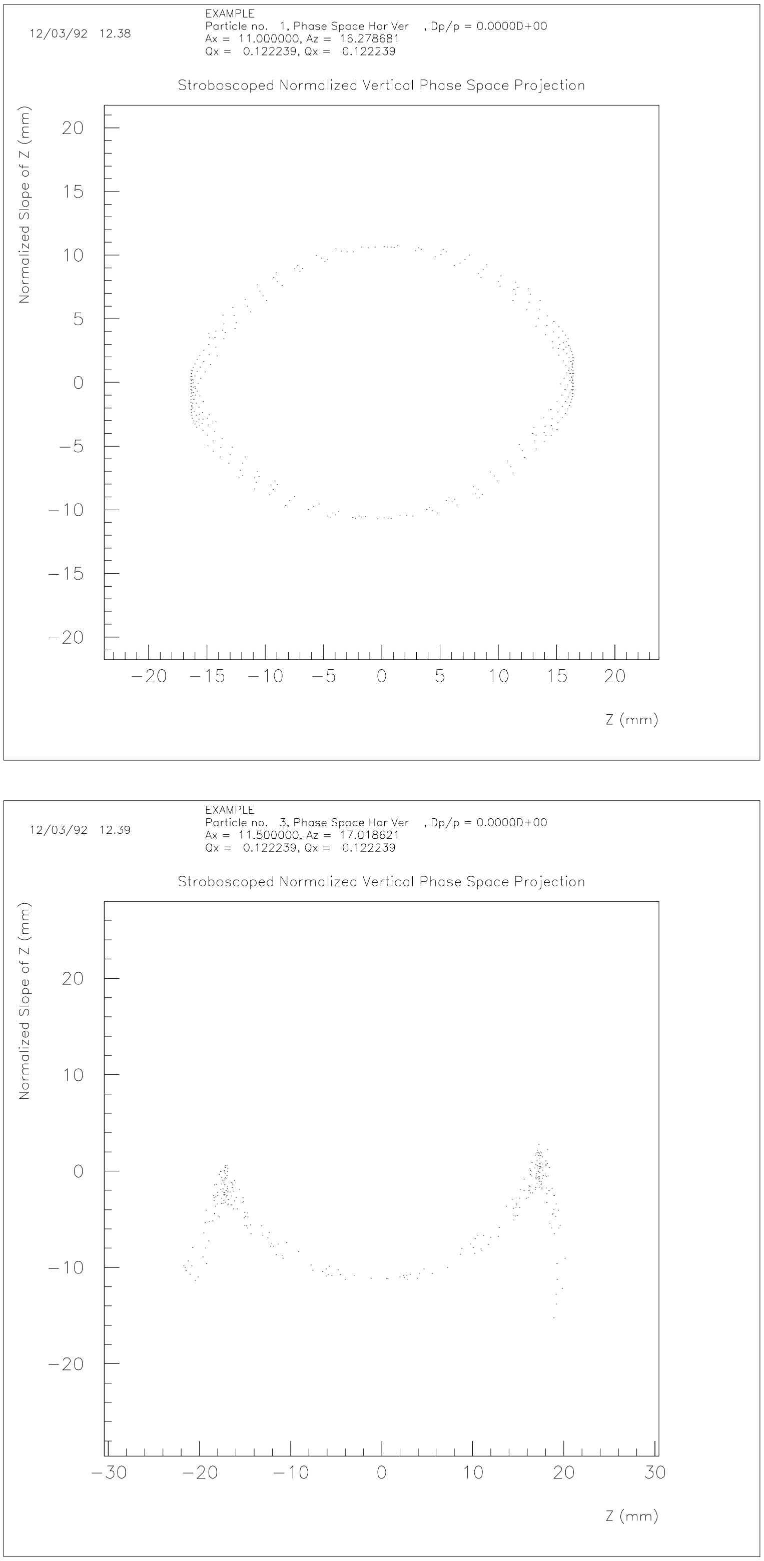

Figure E.4: Stroboscoped Vertical Phase Space Projections for the Regular (upper part) and the Chaotic (lower part) Cases
The regular motion stays inside a 'separatrix' with two unstable fixpoints visible, while the chaotic motion is clearly outside this 'separatrix'.

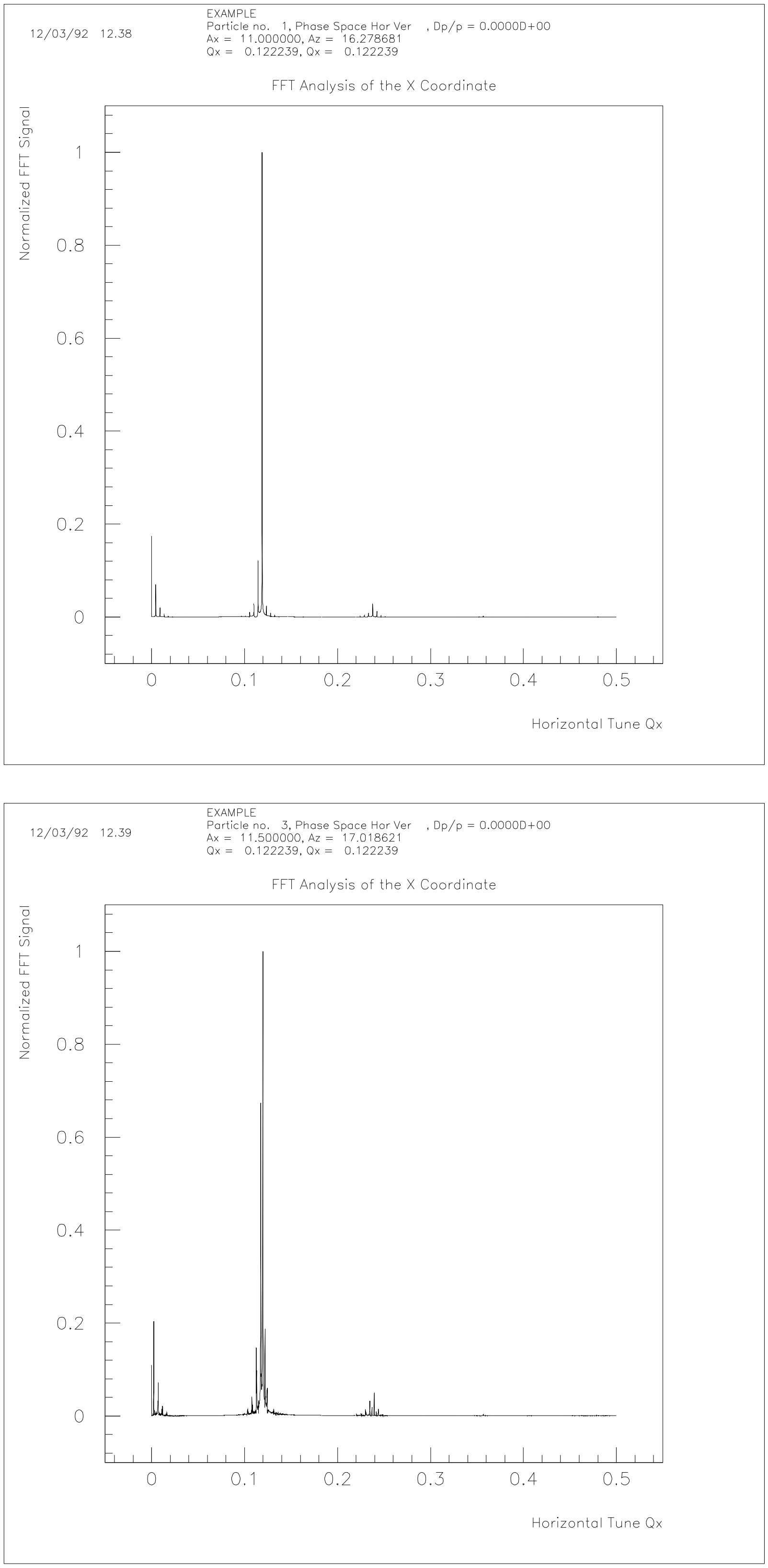

Figure E.5: Horizontal FFT–Analysis for the Regular (upper part) and the Chaotic (lower part) Cases

Bibliography

[1] A. Wrulich, *RACETRACK, A computer code for the simulation of non–linear motion in accelerators*, DESY 84–026 (1984).

[2] B. Leemann and E. Forest, *Brief description of the tracking codes FASTRAC and THINTRAC*, SSC Note SSC–133.

[3] G. Ripken, *Non–linear canonical equations of coupled synchro–betatron motion and their solution within the framework of a non–linear 6–dimensional (symplectic) tracking program for ultra–relativistic protons*, DESY 85–084 (1985).

[4] D.P. Barber, G. Ripken and F. Schmidt, *A non–linear canonical formalism for the coupled synchro–betatron motion of protons with arbitrary energy*, DESY 87–036 (1987).

[5] R. Brun and D. Lienart, *HBOOK User Guide*, CERN Program Library Y250 (1987).

[6] R. Brun and N.C. Somon, *HPLOT User Guide*, CERN Program Library Y251 (1988).

[7] R. Bock, R. Brun, O. Couet, N.C. Somon, C.E. Vandoni and P. Zanarini, *HIGZ User Guide*, CERN Program Library Q120.

[8] G. Guignard, *A general treatment of resonances in accelerators*, CERN 78–11 (1978).

[9] M. Berz, *Differential algebra description of beam dynamics to very high orders*, Particle Accelerators, 1989, Vol. 24, pp. 109–124.

[10] F. Schmidt and M. Vaenttinen, *Vectorization of the single particle tracking program SIXTRACK*, CERN SL/AP Note 90–20 (1990).

[11] H. J. Klein and J. Zoll, *PATCHY, Reference Manual, Version 4.13*, CERN Program Library (1988).

[12] H.Renshall, private communication.

[13] J. Irwin, private communication.

[14] F. Schmidt, *Untersuchungen zur dynamischen Akzeptanz von Protonenbeschleunigern und ihre Begrenzung durch chaotische Bewegung*, DESY HERA 88–02, (1988).

[15] H. Grote and F.C. Iselin, *The MAD program (Methodical Accelerator Design), Version 8.1, User's Reference Manual*, CERN/SL/90–13 (AP).

[16] F. James, *A review of pseudorandom number generators*, to be published in Computer Physics Communication.

[17] M. Berz, E. Forest and J. Irwin, *Normal form methods for complicated periodic systems: a complete solution using differential algebra and Lie operators*, Particle Accelerators, 1989, Vol. 24, pp. 91–107.

[18] A. Hilaire and A. Warman, *A program for single particle tracking*, CERN SPS/88–8 (AMS).

[19] F. Schmidt,F. Willeke and F. Zimmermann, *Comparison of methods to determine long–term stability in proton storage rings*, 1991, Particle Accelerators, Vol. 35, pp. 249–256.